



Blood Tide: Design of a 3D Computer Game

A Major Qualifying Project Report
submitted to the Faculty of the
Worcester Polytechnic Institute
in partial fulfillment of the requirements for the
Degree of Bachelor of Science
on May 4, 2010

Submitted By: Joseph Alea, Garret Doe, Sarah Gilkey, Chris Willams, Frank Williams

Advised By: Professor Jennifer deWinter and Professor Robert Lindeman

Abstract

Blood Tide: Design of a 3D Computer Game

By

Joseph Alea, Garret Doe, Sarah Gilkey, Chris Williams, Frank Williams

This report discusses the results of the implementation of the game *Blood Tide* for a joint Interactive Media and Game Development and Professional Writing Major Qualifying Project. *Blood Tide* is a multiplayer, online, action, real-time strategy game that combines the strategic planning of a real-time strategy game with the fast paced combat of an action game. This report will detail the game's vision, the artistic and technical designs, the gameplay features, the successes and shortcomings of the project, and research on effective strategies for providing in-game help. This report will begin by introducing the game features of *Blood Tide*, then it will discuss the implementation of the game, and finally it will discuss the results. In addition to the results of the implementation, this report includes a post mortem from the project team that discusses lessons that were learned from the hurdles and successes of this project.

Acknowledgements

We would like to acknowledge Robert Lindeman and Jennifer deWinter for their assistance and guidance as our advisors.

Additionally, we would like to express our gratitude toward Sean Beck, who helped us during the implementation phase of the project.

We would also like to recognize the following people for giving us their time throughout the project:

ChanLing Beswick and Colin Cowles for dedicating time writing original music for our game.

Elliot Borenstein, Drew Hickox, Ryan Bedell, and Lukas Wong-Achron, the makers of *Blinding Silence*, for their help and suggestions during the development process.

ChanLing Beswick, Steve Esposito, J. Forrest Hogeboom, Nick Holovnia, Shanna Infantino, Dan Lampke, Amanda Moreno, Pat Nietupski, and Mike Oshetsky for taking the time to test in-game help and provide valuable feedback on our game and its help.

Ryan Bednar, Jordan Fields, Gabe, Wyatt Gray, Carolyn Hardy, Kirk Lanciani, Keegan Leitz, Graham Pentheny, Justin Perron, and Bryan Rickard for their invaluable help and feedback during testing.

Table of Contents

Abstract	ii
Acknowledgements	iii
List of Figures	x
List of Tables	xiii
List of Equations.....	xiv
Chapter 1: Introduction	1
Chapter 2: Gameplay Vision.....	3
2.1. Gameplay	3
2.2. Victory	4
2.3. Artistic Design	4
2.3.1. Art Pipeline	5
2.4. Audio Design	6
2.4.1. Game Music	6
2.4.2. Sound Effects.....	7
2.5. Technical Design	8
2.5.1. Hardware Requirements	8
2.5.2. Software Requirements	8
2.5.3. Entity.....	9
2.5.4. Structures.....	14
2.5.5. Character.....	14
2.5.6. Player Controller	15
2.5.7. Unit Controller	15
2.6. Hint System	17
Chapter 3: Units	19

3.1. Human Units	20
3.1.1. Scuba Troop.....	21
3.1.2. Submarine.....	22
3.1.3. Bomb Ship.....	25
3.2. Piscivian Units	26
3.2.1. Merman.....	27
3.2.2. Manta Ray.....	28
3.2.3. Turtle.....	30
3.3. Unit Sounds	31
3.4. Steering	34
3.4.1. Seeking.....	35
3.4.2. Avoidance.....	35
3.4.3. Interpolation.....	36
3.5. Unit Abilities.....	37
3.6. Commander Types	38
3.6.1. Tactical-Commander	38
3.6.2. Field-Commander	39
Chapter 4: Structures	44
4.1. Artistic Vision	44
4.2. Headquarters	45
4.2.1. Headquarters Controller.....	46
4.2.2. Scuba Troop Headquarters.....	47
4.2.3. Submarine Headquarters	48
4.2.4. Bomb Ship Headquarters	50
4.2.5. Merman Headquarters	51
4.2.6. Manta Ray Headquarters	53

4.2.7.	Turtle Headquarters	54
4.3.	Buildings	55
4.3.1.	Plot Controller	57
4.3.2.	Defense Turret.....	58
4.3.3.	Greenhouse Dome.....	59
4.3.4.	Training Bunker	60
4.3.5.	Sea Wasp Tower	61
4.3.6.	Kelp Garden	62
4.3.7.	Training Grounds.....	63
Chapter 5:	Environment	65
5.1.	Map.....	67
5.2.	Fog of War.....	69
5.3.	Environment Models	70
5.3.1.	Rusted Bomber Airplane	70
5.3.2.	Unexploded Mine.....	70
5.3.3.	Pirate Ship	71
5.3.4.	Fish.....	72
5.3.5.	Coral Reefs	73
5.3.6.	Kelp	74
5.3.7.	Ruins	75
Chapter 6:	User Interface	76
6.1.	Menu System	76
6.1.1.	Main Menu.....	77
6.1.2.	Game Lobby.....	78
6.2.	RTS UI.....	79
6.2.1.	Building UI	81

6.2.2.	RTS Camera	82
6.2.3.	RTS Controls.....	83
6.3.	Field-Commander UI.....	83
6.3.1.	Action Controls.....	85
Chapter 7:	In-Game Help – Overview and Methodology.....	86
7.1.	Literature Review of Game Help.....	87
7.2.	Genre Analysis of In-Game Help	89
7.2.1.	Objective 1: Prepared a Coding Sheet.....	90
7.2.2.	Objective 2: Tested Games and Recorded Presence of Game Help	91
7.2.3.	Objective 3: Analyzed Our Results.....	93
Chapter 8:	In-Game Help – Results and Implementation	95
8.1.	Overall Results of Methodology	95
8.2.	Tutorial Level.....	97
8.2.1.	Results for Tutorial Level	98
8.2.2.	Implementation of Tutorial Level.....	100
8.3.	Mentor Character	100
8.3.1.	Results for Mentor Character	102
8.3.2.	Implementation of Mentor Characters.....	103
8.4.	Pop-Up Windows	105
8.4.1.	Results for Pop-Up Windows	106
8.4.2.	Implementation of Pop-Up Windows	107
8.5.	User Interface Help	111
8.5.1.	Results for User Interface Help.....	112
8.5.2.	Implementation of User Interface Help	113
8.6.	Other Character Help.....	116
8.6.1.	Results for Other Character Help	117

8.7.	Dialog Tree Help	118
8.7.1.	Results for Dialog Tree Help	119
8.8.	In-Ear Guide	121
8.9.	Synopsis	122
Chapter 9:	Testing for <i>Blood Tide</i>	124
9.1.	In House Testing	124
9.1.1.	Results from In-House Testing	125
9.2.	Ad Hoc Testing	125
9.2.1.	Results from Ad hoc Testing	126
9.3.	Game Help Testing.....	127
9.3.1.	Objective 1: Prepared a Coding Sheet.....	128
9.3.2.	Objective 2: Tested Games and Recorded Presence of Game Help	129
9.3.3.	Objective 3: Analyzed Our Results.....	130
Chapter 10:	Post Mortem.....	134
10.1.	Technical Reflection	135
10.2.	Artistic Reflection	138
10.3.	Game Help Reflection	139
References	141
Appendix A: Art Asset List		
Appendix B: Art Timeline		
Appendix C: Technical Timeline		
Appendix D: Written Guide Forms		
Appendix E: Coding Sheet		
Appendix F: Coding Sheet for <i>Baten Kaitos</i>		
Appendix G: Coding Sheet for <i>Final Fantasy X</i>		
Appendix H: Coding Sheet for <i>Paper Mario</i>		

Appendix I: Coding Sheet for *Rogue Galaxy*

Appendix J: Coding Sheet for *Star Ocean*

Appendix K: Example Application of Weighted Total

Appendix L: Results Summary

Appendix M: Tutorial Script

Appendix N: Implemented Loading Screens

Appendix O: Menu Navigation Test Plan

Appendix P: Input Functionality Test Plan

Appendix Q: Tutorial Functionality Test Plan

Appendix R: RTS Mode Unit Functionality Test Plan

Appendix S: RTS Mode Building Functionality Test Plan

Appendix T: RTS Mode Interface Functionality Test Plan

Appendix U: Action Mode Interface Functionality Test Plan

Appendix V: Multiplayer Functionality Test Plan

Appendix W: Map Functionality Test Plan

Appendix X: Artistic Functionality Test Plan

Appendix Y: Overall Help Functionality Test Plan

Appendix Z: Ad Hoc Testing Result Database

Appendix AA: *Blood Tide* Help Effectiveness Test Plan

Appendix AB: *Blood Tide* Help Effectiveness Coding Sheet Results for Playing Skipping Tutorial

Appendix AC: *Blood Tide* Help Effectiveness Coding Sheet Results for Playing Tutorial

List of Figures

Figure 1: Hydropolis Hotel (Hauser)	5
Figure 2: Ambient Source	7
Figure 3: Entity Hierarchy	9
Figure 4: Stats Definition.....	10
Figure 5: Collisions Enabled Results	12
Figure 6: Collisions Disabled Results	13
Figure 7: Goal System	16
Figure 8: Help Window.....	18
Figure 9: Scuba Diver (“Scuba Diving 03 Jpg”).....	22
Figure 10: Scuba Troop In-Game Model	22
Figure 11: Submarine Reference (Lee).....	24
Figure 12: Submarine In-Game Model	24
Figure 13: Bomb Ship Reference (DaChief)	26
Figure 14: Bomb Ship In-Game Model	26
Figure 15: Statue of Poseidon (“Poseidon / Neptune God of the Sea”).....	28
Figure 16: Merman In-Game Model	28
Figure 17: Manta Ray Reference (de Pastino).....	29
Figure 18: Manta Ray In-Game Model	29
Figure 19: Turtle Reference (Sangster).....	31
Figure 20: Turtle In-Game Model.....	31
Figure 21: Seeking	35
Figure 22: Avoidance	36
Figure 23: Interpolation	37
Figure 24: Ability Definition	38
Figure 25: Commander Reselect Pane.....	42
Figure 26: Ghost Effect	43
Figure 27: Hydropolis Hotel (Hauser).....	45
Figure 28: Headquarters Controller	47
Figure 29: The Hemisfèric(David K.)	48
Figure 30: Scuba Troop Headquarters In-Game Model	48
Figure 31: WWII Submarine Pen (“The submarine pens at St Nazaire”)	49
Figure 32: Submarine Headquarters In-Game Model	49
Figure 33: National Grand Theater of China in Beijing (Business Week).....	51
Figure 34: WWII Dock (“PT 658 at Vancouver dock”).....	51
Figure 35: Bomb Ship In-Game Model.....	51
Figure 36: Sand Castle Tower (Susan)	52
Figure 37: Merman In-Game HQ.....	52
Figure 38: Manta Ray HQ Reference (“Coral Reef”)	54
Figure 39: Manta Ray HQ In-Game Model.....	54
Figure 40: Outdoor Garden(gift ideas).....	55
Figure 41: Turtle HQ In-Game Model	55
Figure 42: Plot Controller	57
Figure 43: Implemented Plot Model.....	58
Figure 44: Plot Controller	58
Figure 45: WWII Turret (Kuzmin).....	59

Figure 46: Human Turret In-Game Model	59
Figure 47: Geodesic Greenhouse Dome (Colin).....	60
Figure 48: Greenhouse Dome In-Game Model.....	60
Figure 49: WWII Battle Bunker (Teskem)	61
Figure 50: Training Bunker In-Game Model	61
Figure 51: Sea Wasp Tower Reference (Andrei)	62
Figure 52: Sea Wasp Tower In-Game Model	62
Figure 53: Greek Garden(Boharsik)	63
Figure 54: Kelp Garden In-Game Model	63
Figure 55: Greek Temple (Davies)	64
Figure 56: Training Grounds In-Game Model.....	64
Figure 57: Human Side of Game World	66
Figure 58: Piscivian Side of Game World.....	66
Figure 59: Game Map.....	68
Figure 60: Example of Fog of War	69
Figure 61: Rusted Bomber Airplane.....	70
Figure 62: Underwater Mine	71
Figure 63: Pirate Ship	72
Figure 64: Green Fish School	73
Figure 65: Ocean Sunfish.....	73
Figure 66: Teardrop Butterfly Fish	73
Figure 67: Blue Tang	73
Figure 68: Variation of a Coral Reef	74
Figure 69: Kelp	74
Figure 70: Variation of Ruins	75
Figure 71: Selection of Multiple Units	76
Figure 72: Menu Item	77
Figure 73: Design for Main Menu	78
Figure 74: Implemented Main Menu.....	78
Figure 75: Game Lobby	79
Figure 76: RTS UI.....	80
Figure 77: Building UI.....	81
Figure 78: Camera Blending.....	82
Figure 79: Field-Commander UI.....	84
Figure 80: Scribblenauts - Tutorial Level (Slisenberger)	98
Figure 81: Paper Mario - Prologue Introduction Screen (Chuggaacronoy).....	99
Figure 82: Warcraft 3 - Uther advising Arthas	101
Figure 83: Paper Mario: Mentor - Professor Frankly (Helmerts10)	102
Figure 84: Implemented Mentor for Piscivian Faction - General Crabbly.....	104
Figure 85: Implemented Mentor Character for Human Faction - TACTIS.....	104
Figure 86: Mario Kart Wii - Wrong Way Pop-Up (Hmyaa).....	105
Figure 87: Rogue Galaxy - Pop-Up Tip (“Rogue Galaxy”)	106
Figure 88: Implemented Loading Screen for Piscivian Faction	108
Figure 89: Implemented Loading Screen for the Human Faction	108
Figure 90: Implemented Help Window	109
Figure 91: Controls Window.....	110
Figure 92: Command and Conquer - Red Alert 2 Mini Map (Steeven)	111
Figure 93: Rogue Galaxy: User Interface Mini-map (“Rogue Galaxy”)	112

Figure 94: Implemented Tactical-Commander Interface	114
Figure 95: Implemented Building Interface.....	115
Figure 96: Implemented Field-Commander Interface.....	116
Figure 97: Okami - Amaterasu listening to Sakuya (Metallian)	117
Figure 98: Baten Kaitos - Dialog Tree Tutorial.....	120
Figure 99: Halo 3 - Master Chief and Cortana.....	122
Figure 100: <i>Kingtom Hearts 2</i> - Jiminy's Journal (Haynes).....	123

List of Tables

Table 1: Unit Strength and Weakness Chart	20
Table 2: Scuba Troop Unit Profile.....	21
Table 3: Submarine Unit Profile	23
Table 4: Bomb Ship Unit Profile	25
Table 5: Merman Unit Profile.....	27
Table 6: Manta Ray Unit Profile	29
Table 7: Turtle Unit Profile.....	30
Table 8: Sound Effects for Human Units	32
Table 9: Sound Effects for Piscivian Units	33
Table 10: General Unit Sounds	34
Table 11: Human Faction Field-Commander Abilities	40
Table 12: Piscivian Field-Commander Abilities.....	41
Table 13: Headquarters Stats	45
Table 14: Building Stats	56
Table 15: Average Skills Learned for tested games.....	96
Table 16: Tutorial Level Results	100
Table 17: Mentor Character Results.....	103
Table 18: Pop-Up Window Results	107
Table 19: User Interface Results	113
Table 20: Other Character Help Results	118
Table 21: Dialog Tree Tutorial Results	120

List of Equations

Equation 1: Weighted Total	94
----------------------------------	----

Chapter 1: Introduction

We developed a multiplayer game called *Blood Tide*. *Blood Tide* has an innovative gameplay mechanic that gives the player the ability to command troops and strategize across the entire battlefield in addition to fighting the enemy on the field in hand-to-hand combat. Our game allows players to play as one of two factions pitted against each other in an underwater battle.

During development, we created designs for the technical and artistic assets needed for the game. During this phase, we also analyzed other genres of games that incorporated in-game help in order to implement in-game help into *Blood Tide*. This design phase allowed us to create a structure, which we would follow for the implementation phase. During the implementation, we focused on creating all of the technical and artistic components of the game, from the core to the desired. We also focused on incorporating help into the game during this phase. The technical implementation for *Blood Tide* focused on creating core game features, constructing user interface elements to assist the player in controlling the game, and optimizing the game to run smoothly. The vision for the art in *Blood Tide* was to create a unique artistic style and feel for each of the two factions that would complement the aquatic theme of the game. Each art asset that we created corresponded to the overarching theme. The goal of the in-game help was to teach the player about all of the game mechanics in *Blood Tide* without breaking immersion. Once the implementation phase was completed, we conducted functional testing using test plans and ad hoc testing to locate and fix bugs in our game. We focused on fixing the most critical errors, such as crashing, to create a complete, functional, and well-balanced game.

Chapter two will describe the overall game vision for *Blood Tide*. This chapter will detail all of the artistic, audio, and technical designs, as well as our plan to research and incorporate in-game help into *Blood Tide*. Chapter three will discuss all of the units in the game, including the two modes of play that are featured in our game. Chapter four will outline all of the structures in *Blood Tide* such as headquarters and buildings. Chapter five will describe the environment in the game world. Chapter six will discuss the user interfaces used for both command roles. Chapter seven describes our literature

review of in-game help, the methodology we used to test in-game help. Then, chapter eight, as a continuation of chapter seven, is the results of our methodology, including the definition of each form of help. Chapter nine outlines our testing plan and the results of our testing phase. Finally, chapter ten is post mortem, serving as a reflection on the project as a whole.

Chapter 2: Gameplay Vision

War on Earth has rendered its surface uninhabitable for the human race. In order to preserve themselves, humans have moved to the depths of the ocean and established new colonies. However, the Humans' fight for survival is threatening the delicate ecosystem, which an ancient aquatic-human race, known as Piscivian, has survived within for millennia. The limited underwater resources make it impossible to share the sea, thus the two factions are gearing up for battle – a battle that will determine which race will thrive and which will become a memory.

Blood Tide is an online multiplayer game that pits two players against each other. Each player takes control of either the Human or Piscivian faction. Players are able to play two unique roles: tactical-commander and field-commander. As the tactical-commander, the player controls units and constructs buildings, like a real-time strategy (RTS) game. As the field-commander, the player takes the role of one unit on the field and battles alongside the other units. This role plays like an action game with a third-person chase camera. The player can switch between the two roles to better utilize the functions of the tactical-commander and field-commander.

In this chapter, we will discuss the gameplay vision for *Blood Tide*, the victory condition of the game, and the specific artistic, audio, technical, and game help designs for the game.

2.1. Gameplay

At the start of a game session, each faction has three headquarters. These headquarters act as spawning points for different types of units. The units can then be controlled by the player in tactical-commander mode in a similar way to other RTS games. In addition to spawning units, the headquarters can be further specialized by building farms, training grounds, and guard towers around them. Headquarters and buildings can be attacked by enemy units and field-commanders. When a faction loses all three of its headquarters, that faction loses the game.

2.2. Victory

Currently the only way to win a game in *Blood Tide* is by eliminating enemy headquarters; however, we built the victory controller, which decides when the game is won, to support other types of victories. Creating the victory controllers in this manner was useful because we could easily create new victory conditions in the future should we choose to. The logic for deciding when a player has won is part of the C++ code, but to enable a specific victory condition, such as destroying all enemy headquarters, a level designer only needs to place a node with the victory controller on it in the world editor. If there were more types of victory conditions other than the one that we implemented, the level designer would have to specify the condition that the controller uses.

2.3. Artistic Design

The art in *Blood Tide* uses an aquatic theme with a combination of realistic and fantastic styles while keeping a serious mood during the game. The Piscivian and Human factions each have their own art styles that are based on different values and cultures. The art style in *Blood Tide* is the medium used to show the differences in the factions and adds color to the game.

The art style for the Human faction blends a World War II industrial style with the underwater architecture and design of the Hydropolis Hotel— a planned underwater hotel constructed off Jumeira Beach in Dubai, as shown in Figure 1, that was scheduled to open by the end of 2009 (Hauser). We were inspired by the familiarity of World War II weaponry and military bases seen in other war games. However, we wanted to blend a futuristic architecture that would represent a humanized underwater city, because we felt that futuristic architecture was more appropriate for a time period in which Earth's surface is uninhabitable.

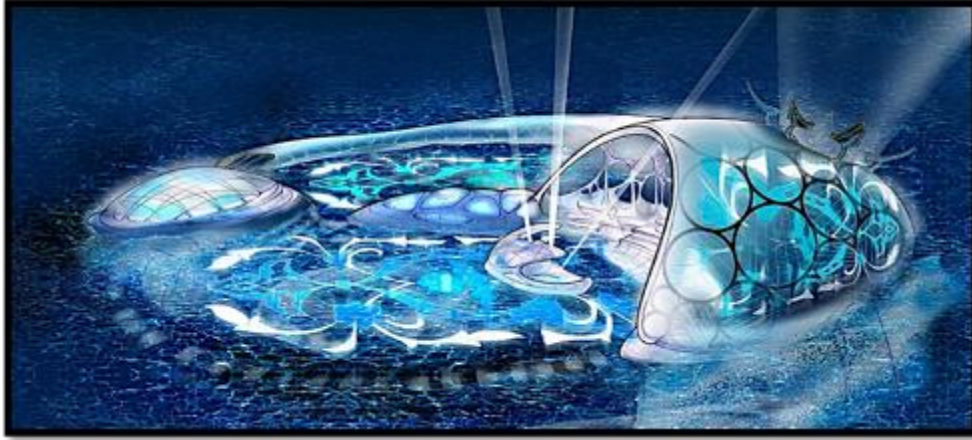


Figure 1: Hydropolis Hotel (Hauser)

The art style for the Piscivian faction blends Doric Greek architecture with a Hawaiian white-sand beach theme. We were inspired by the feel and familiarity of ancient Doric Greek architecture, but we wanted to blend the Hawaiian white-sand beach theme to create a more unique art style. Additionally, we wanted the Piscivian's art style to harmonize with *Blood Tide's* story because the art style can greatly impact the mood of a game. Correlating the story and art style helps the player understand each faction and the environment that they are fighting for; therefore, the Piscivian art style represents a lush, natural, and delicate ecosystem that the Humans are threatening.

Several different types of art assets are included in our game: menu and UI art, unit models, building models, environment models, and audio. For a complete list of every art asset, see Appendix A: Art Asset List. In addition to this, we created a timeline to accommodate the creation of all of the 2D and 3D assets, which is detailed in Appendix B: Art Timeline.

2.3.1. Art Pipeline

One of the major challenges we faced was to get the artwork, like models and animations, and content, like unit stats and names, into the game in a robust and scalable way. We decided early on that it would be better to have a system where anyone could get new content to work in the game, not just the programmers or people who know C++. This allowed our entire team to be able to quickly change the content that is in the game. We decided that the simplest approach would be to use string tables, which

are formatted text files, to completely define all of our in-game content. These string tables contain a pseudo scripting language that gets parsed for the information needed by the game. The main string tables that we used were for buildings, commanders, game, input, messages, music, projectiles, units, worlds, and game scripts. Each of these string tables encapsulates some portion of the game logic and is named to describe the section that it deals with. In addition to these string tables, we have also modified the world editor to be able to define level-specific content, which cannot be done using the string tables, such as the art assets and functionality for headquarters.

The content pipeline begins when the game first loads. All of the string tables are loaded and stored in a cache that can be quickly accessed. Whenever another game object is created and wants to use information from a string table, it accesses the string table manager, which is globally available to all game systems. Once an object obtains the correct string table, it can pull out whatever information it requires. For example, units extract information about their models, animations, and stats from the string tables.

2.4. Audio Design

We have two varieties of audio in our game – music and sound effects. We outsourced custom game music, and we collected free sound effects and adjusted them in Audacity.

2.4.1. Game Music

Blood Tide features two musical tracks: a Piscivian theme song and a Human theme song. Originally, we were going to have three songs: one song for the main menu, one song when playing as the tactical-commander, and one song when playing the field-commander. However, we felt that each faction should have its own theme music to further differentiate the culture of the two factions. The differentiation between the two factions was more important than the differentiation between the two roles because it helps establish our game vision in terms of the look-and-feel for each faction.

The game music was inspired by *The Battle* by Harry Gregson-Williams, used in the movie *Narnia: The Lion the Witch and the Wardrobe*. This music piece consists of mostly strings and horns to give the player a strong sense of being in battle and heighten the dramatic tension during the game.

2.4.2. Sound Effects

To make the sounds work in-game, we used built in sound mechanisms from the C4 Engine and modified them to match our needs. The C4 Engine already supports 2D and 3D sounds, so we did not need to create our own sound mechanisms.

For background music, we used an AmbientSource node that is placed in the world editor. This node dictates whether or not there will be music and at what volume the music will play, as shown in Figure 2. The music string table dictates the songs that will play and is different for each team. In addition to the music, the user interface noises are also controlled by string tables. The events that trigger these sounds are done in the C++ code, but the specific sounds that play are specified in the string tables.

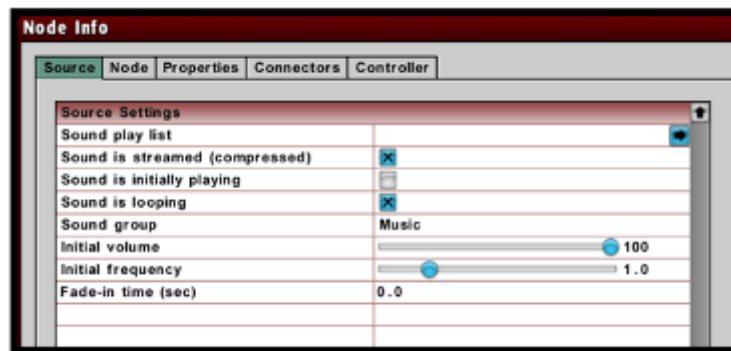


Figure 2: Ambient Source

For 3D sounds that are location specific, we used OmniSource nodes from the C4 Engine. This sound source is attached to the world where the entity that generated the sound is located. Since the sound is 3D, it becomes louder when the player moves closer to the source's position. The sound is automatically deleted from the world when it finishes playing.

2.5. Technical Design

This section describes the high-level technical design for *Blood Tide*. It details our technical goals for *Blood Tide* and some of the major abstractions we created that affect most of the other areas of the game. The implementation of the technical assets was scheduled in Appendix C: Technical Timeline.

2.5.1. Hardware Requirements

Although we have not tested our game on a large number of computers with different hardware, we created some specifications to approximate the hardware that we have actually tested on. Besides testing, some of our hardware requirements come from the requirements needed to run the C4 Engine. The following list details the hardware requirements needed to play *Blood Tide*:

- Operating System: Windows XP/Vista/7
- Graphics Card: Nvidia GeForce 6600 or higher, ATI Radeon X1300 or higher
- RAM: 2 GB
- Processor: Intel Pentium D processor or better
- Hard-disk Space: 1.5 GB
- Internet Connection: 300kb/s or better

2.5.2. Software Requirements

Blood Tide met the software requirements we set forth in the design phase. Because the game runs smoothly, we have readjusted our requirements. The following list details the extent of processing that *Blood Tide* can achieve:

- Players: 2 over a network
- Video frames per second: 60
- Units in world: Maximum of 60

Originally we planned for the game to run at 30 frames per second minimum with 50 units in the game world. At maximum, we have run the game with 70 units in the game world at 100 frames per second; however, in order to account for computers

that run at our minimum requirements, we decided that 60 frames per second with 60 units in the game world was a good balance of pushing our game's limits and making it run smoothly on older machines.

2.5.3. Entity

An entity is anything in our game world that performs actions and is important to gameplay. For example, units, buildings, headquarters, and field commanders are all entities. Early on in the implementation process, we realized that the unit and commander controllers had a lot in common. Therefore, we made both commanders and units share traits, such as stats and animations. During development, we decided that buildings, plots, and headquarters could also be categorized as entities because they have stats, models, and can collide with other entities. A visual depiction of the entity base class and its relation to other game components is shown in Figure 3

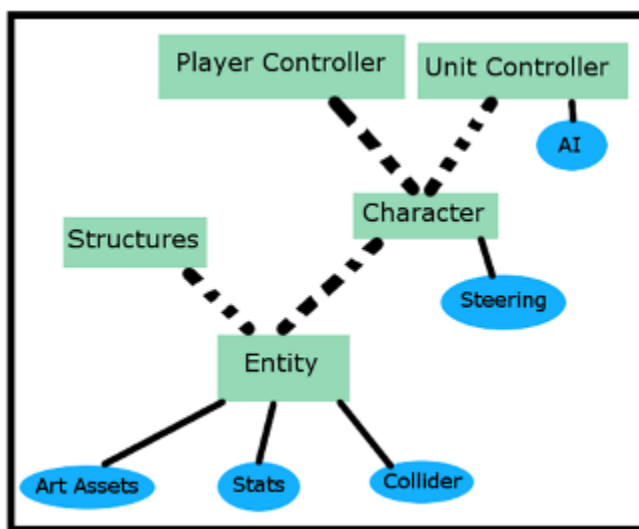


Figure 3: Entity Hierarchy

This entity base class is the crux of our gameplay logic because it encompasses all of the things that act in our game. The entity class allows us to express all of the gameplay logic that we want in a way that is easily modified. The graphic above shows the three pieces that define an entity: art assets, stats, and colliders.

2.5.3.1. Entity Art Assets

All entities have art assets because entities are objects in the game world and need to be visually represented. Specifically, all entities have 3D model information. Some entities have information about the icons that are associated with them, if the entity is a unit, structure, or field-commander, or animation information for a unit or field-commander. These art assets are described more in-depth in Chapter 3: Units and Chapter 4: Structures.

2.5.3.2. Entity Stats

Each entity has a stats controller, which contains all of the stats that we designed, such as an entity's health and strength. Different stats get used depending on which entity the stats module is connected to. For example, headquarters do not use movement speed, while units do. Figure 4 shows a stats definition for one of the game units.

```
'STAT' "Unit Stats"  
{  
    'MHP' "250"  
    'HP' "250"  
    'STR' "30"  
    'ARMR' "20"  
    'MSPD' "15"  
}
```

Figure 4: Stats Definition

The stats controllers are specified in the string tables, or sometimes in the world editor for entities that are not specified in string tables, such as headquarters. The stats shown in the figure are maximum health, starting health, strength, armor, and movement speed.

2.5.3.3. Entity Colliders

The first time we integrated our art assets into the game, we realized the game could not meet our software requirements. Our initial technical requirements were designed to allow up to 50 units in the world and a minimum of 30 frames per second. The original physics system that was used in *Blood Tide* came from preexisting systems in the C4 Engine. To see whether or not our models had too many polygons or if the

collision code in the C4 Engine couldn't support our game, we ran the Sleepy profiler while playing the game. The Sleepy profiler is a program that runs in the background and records information about which portions of our game are taking up the most time.

We first ran the profiler while using the C4 Engine collision system. To test out the collisions, we placed eight submarines in the game. Our average rate of frames per second during this test was approximately 20. By inspecting the results of our profiling in Figure 5, we saw that two of the top five processes running were related to collision detection. In just our code, seven of the top eleven functions were related to collision detection. The problem of the poor performance was clearly due to collisions, especially obvious when there were eight objects in the world that weren't moving. At first we were confused because collisions are supposed to occur between moving objects; however, we discovered that the C4 Engine checks all of the geometries in the game world for collisions against every other geometry every engine cycle, or a single iteration of its main loop. The main loop continually cycles through code that updates the game, draws the graphics to the screen, and continually checks for collisions. Instead of checking for collisions while units are in motion, the C4 Engine handles collisions after all units have moved.

Name	Exclu...	Inclusive	% Exclusive	% Inclusive	Module
C4::GeometryObject::ClipParallelogramToCollision...	2183	2183	20.800	20.800	BloodTide
DrvPresentBuffers	721	1713	6.870	16.322	nvoglv32
KiFastSystemCallRet	674	674	6.422	6.422	ntdll
RtlCompareMemoryUlong	628	628	5.984	5.984	ntdll
C4::Collision::CollideSweptSegmentAndCylinder	592	1963	5.641	18.704	BloodTide
sqrt	439	439	4.183	4.183	BloodTide
C4::MemoryMgr::InitMemory	435	435	4.145	4.145	BloodTide
C4::GeometryObject::DetectCapsuleVertexOctreeC...	384	2345	3.659	22.344	BloodTide
C4::GeometryObject::DetectEdgeOctreeCollision	272	623	2.592	5.936	BloodTide
C4::GeometryObject::ClipSegmentToCollisionBou...	204	204	1.944	1.944	BloodTide
C4::operator*	204	204	1.944	1.944	BloodTide
C4::GeometryObject::DetectVertexOctreeCollision	169	344	1.610	3.278	BloodTide
C4::InverseMag	162	162	1.544	1.544	BloodTide
C4::operator*	161	161	1.534	1.534	BloodTide
C4::GeometryObject::DetectCapsuleEdgeOctreeCo...	159	2808	1.515	26.756	BloodTide
C4::Image::DecompressImageRLE_RGBA32	140	140	1.334	1.334	BloodTide
C4::operator-	133	133	1.267	1.267	BloodTide
C4::operator*	127	127	1.210	1.210	BloodTide
C4::SquaredMag	124	124	1.182	1.182	BloodTide
RtlFillMemoryUlong	122	122	1.162	1.162	ntdll
_checkTOS_withFB	96	96	0.915	0.915	BloodTide
C4::Vector3D::Normalize	87	104	0.829	0.991	BloodTide
C4::operator*	86	86	0.819	0.819	BloodTide
C4::Math::ClipSegmentAgainstPlane	81	242	0.772	2.306	BloodTide
C4::operator+	80	80	0.762	0.762	BloodTide
C4::Collision::CollideSweptSegmentAndSphere	71	140	0.677	1.334	BloodTide

Figure 5: Collisions Enabled Results

To test this theory, we disabled collisions on the submarine models. Once we did this, we noticed that our frame rate on average was 100. Our profiling results in Figure 6 show that only three functions were related to collision and were no longer the top resource users. One function of note is ClipParallelogramToCollision, which attempts to check if a polygon on some geometry is colliding with another polygon. In our first profiling run, this function was the top resource user, making 2183 exclusive calls and 2183 inclusive calls. Exclusive calls exclude all functions outside of GeometryObject, which is the class it belongs to. Inclusive calls include all functions called while this function is on the stack. In our second profiling run, this function made 128 exclusive calls and 128 inclusive calls. This one function was only a small piece of the problem, but many of the collision functions were using far less CPU cycles.

Name	Exclu...	Inclusive	% Exclusive	% Inclusive	Module
KiFastSystemCallRet	2634	2634	40.856	40.856	ntdll
C4::MemoryMgr::InitMemory	256	256	3.971	3.971	BloodTide
C4::World::DetectNodeCapsuleCollision	229	737	3.552	11.432	BloodTide
C4::operator*	224	224	3.474	3.474	BloodTide
RtlCompareMemoryUlong	174	174	2.699	2.699	ntdll
C4::World::DetectNodeCollision	171	417	2.652	6.468	BloodTide
C4::operator-	163	163	2.528	2.528	BloodTide
C4::GeometryObject::ClipParallelogramToCollision...	128	128	1.985	1.985	BloodTide
[6991bcf6]	127	127	1.970	1.970	nvoglv32
C4::Image::DecompressImageRLE_RGBA32	109	109	1.691	1.691	BloodTide
NtYieldExecution	82	82	1.272	1.272	ntdll
C4::Skybox::ProcessRegion	72	72	1.117	1.117	BloodTide
SwitchToThread	59	59	0.915	0.915	kernel32
RtlFillMemoryUlong	51	51	0.791	0.791	ntdll
C4::operator*	51	51	0.791	0.791	BloodTide
C4::operator*	50	63	0.776	0.977	BloodTide
C4::GraphicsMgr::DrawRenderList	43	1830	0.667	28.385	BloodTide
C4::Dot3D	40	40	0.620	0.620	BloodTide
C4::UpdatableTree<C4::Node>::Invalidate	39	51	0.605	0.791	BloodTide
sqrt	38	38	0.589	0.589	BloodTide
C4::Inverse	37	42	0.574	0.651	BloodTide
C4::operator*	35	44	0.543	0.682	BloodTide
C4::operator*	33	33	0.512	0.512	BloodTide
C4::GraphicsMgr::DrawVelocityList	30	134	0.465	2.078	BloodTide
C4::Node::CalculateWorldTransform	30	97	0.465	1.505	BloodTide
DrvCopyContext	29	404	0.450	6.266	nvoglv32

Figure 6: Collisions Disabled Results

In order to meet our software requirements, we were forced to rethink how our collisions would work. We decided that, instead of using precise collisions and handling collisions after movement, we would use a simpler approach that we termed “soft collisions” and would handle collisions in real time. First, we removed C4’s collision code and made our own colliders to help determine collisions. Our colliders store an entity’s position and determine if another entity is colliding with it. Instead of checking for collisions on all geometries, the function checks collisions against all colliders, which reduces the number of calculations far less colliders are present in the game world than there are geometries. Our colliders were modeled after the C4 colliders, but use simpler calculations. The collision calculations are not very precise because the colliders use 3D

cylinders and 2D circles to check collisions instead of checking each geometry in a model. Also, our collision system does not attempt to prevent and correct collisions – that is left to the steering component, as discussed in section 3.4. Instead, our collision system tells other systems, like the artificial intelligence system, if a collider is in a collision state.

2.5.4. Structures

Various types of structures stem from the entity base class. These types are plots, buildings, and headquarters. They all have models and stats, but make no use of animations because they do not move around in the game world. We will discuss the building types more in-depth in Chapter 4: Structures.

2.5.5. Character

A character is defined as an entity in the game world that can perform abilities and move around and can either be a player or a unit, as discussed in Chapter 3: Units. Abilities can be both offensive and defensive. The important difference between a character and an entity is that a character also has a steering module in addition to the stats, collision, and asset modules found on entities.

2.5.5.1. Character Steering

Characters have steering modules that contain the logic for movement, allowing characters to have various different steering behaviors. These behaviors are seeking a target, avoiding other entities, interpolating positions, moving without facing the direction of movement, and staying separated from other entities. Each character can be defined to have different behaviors, or the artificial intelligence can set the behaviors. This module is discussed more in detail in section 3.4.

2.5.6. Player Controller

The player controller extends the character base class, but it contains player-centric information. A player controller, or the field-commander, is controlled by the player using the keyboard or moving the mouse. In addition to this, it also allows the field-commander to gain ranks and unlock new abilities, unlike its unit counterpart. This controller is discussed further in section 3.6.2.

2.5.7. Unit Controller

The unit controller has similarities to the player controller, but is governed by artificial intelligence instead of player actions. For example, instead of a player pressing a button to shoot a torpedo, the AI decides when and in which direction to shoot. Most of the information needed to define a unit is contained in string tables, as part of the art pipeline.

2.5.7.1. Artificial Intelligence

We designed the artificial intelligence to have three main systems: movement, path-finding, and unit attacks. Path-finding was removed from the game because there were no static objects to avoid, so we did not need to implement a system that will find a path around obstacles. Unit attacks were implemented as part of our ability system instead of part of the artificial intelligence system to allow us to reuse the ability system, which is also used by field commanders and buildings. While movement was mostly unchanged from the initial design, it became fundamentally a different component from the artificial intelligence. Movement is now part of the steering component, discussed in section 3.4, which allowed us to have entities move without requiring that they also have artificial intelligence.

The artificial intelligence component that was implemented models entity goals, or desired world states. The two types of goals that we created are primitive goals and composite goals. Primitive goals try to make entities perform actions or wait for a desired world state. Composite goals are similar to primitive goals, but have sub-goals, which can be primitive or composite goals themselves, and are evaluated with the parent

goal. The goal system made it easy to create new entity behaviors, requiring little coding to implement. Also, complex behaviors can be modeled by composing simpler goals.

The goals system works by allowing each goal to perform an enter procedure, an exit procedure, and an update procedure. The responsibility for updating a goal is given to the entity that has that goal. Goals automatically handle calling their enter and exit procedures and are responsible for updating their own sub goals, as shown in Figure 7.

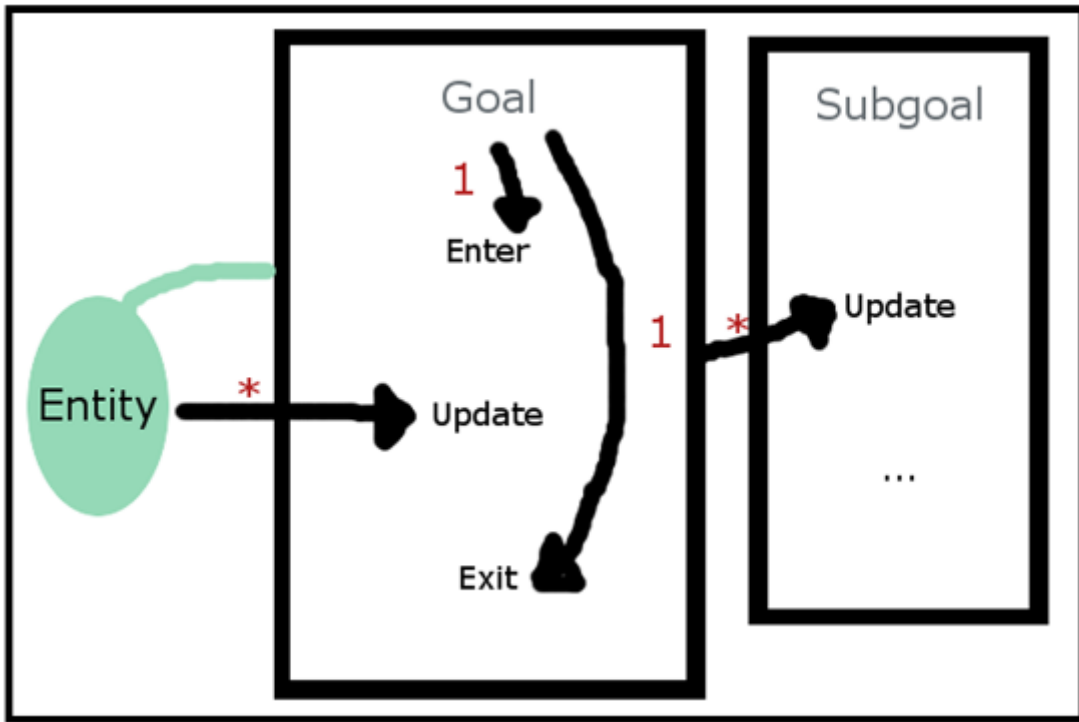


Figure 7: Goal System

The arrows in the figure represent a call to a procedure, while asterisks represent an infinite amount of calls. In the example above, the arrow from entity to goal's update procedure can be called many times, which makes sense because goals need to be constantly evaluated. A goal calls its sub-goals' update procedures many times in the same way that the entity calls its goals' update procedures. The arrows from goal to the enter and exit procedures represent the goal calling its own procedures only once because a goal is only entered and exited once.

2.6. Hint System

The hint system began as a small feature meant to display some help messages. During development, we decided that we wanted a more advanced in-game help system, which meant that we needed to expand the hint system. The forms of help, including the forms we implemented into our game, are described in Chapter 7: In-Game Help – Overview and Methodology and Chapter 8: In-Game Help – Results and Implementation.

The main element of the hint system is the help window, depicted in Figure 8, which displays all of the help messages to the player. Some messages are based upon player inactivity; for example, if the user has not built a building in three minutes, the game will remind the player to build something. Because the construction of buildings in an integral of any RTS game, including *Blood Tide*, we did not want players who were unfamiliar to the game to be at a disadvantage during online play. Similarly, we wanted to remind players about other important gameplay aspects of our game, so we created other forms of player inactivity in addition to buildings, like reminding the player how to choose a field-commander and how to select and order units.

Other messages are event based; for example, when a commander dies, the game will send an alert to notify the player, because the player might not be watching the commander at the time of its death. We were concerned that it could be frustrating to not know when the field-commander dies or a headquarters is destroyed, so the help system gives the player this information. These message types are specified in the string tables and have additional sounds that can be played in conjunction with the messages. The sounds are optional because the message window is primarily textual, but for important events we wanted to also play a sound to alert the player that an event is more significant than other events.



Figure 8: Help Window

Another help element is the loading screen tip. While the game is loading, a random tip is displayed for the player's chosen faction and features one of the three units a team can command. The loading screen tips also remind the player of the map's objective. The player is reminded of the objective because the goal of the game is to complete said objective before the opponent. Also, the randomized tips are useful for giving the player additional information about specific units without requiring players to read a game manual.

We also included help in separate windows: a controls window, which displays all of the available controls for the player, and a window that appears one time after installation, informing players of a tutorial level for new players. Players need to know the controls in order to play the game, so having a window that can be toggled at will allowed us to give the player this information without forcing the player to have a physical copy of the controls or memorize the controls.

Chapter 3: Units

There are six different units in *Blood Tide* with three for each faction. The Human faction's units are the scuba troop, submarine, and bomb ship. The Piscivian faction's units are the merman, manta ray, and turtle. There are five stats that define a unit:

- **Health** refers to the amount of life that a unit has and is represented by a numerical value in the user interface. Each time an enemy ability collides with a unit, that unit will lose health. When the health has a value of zero or below, then the unit dies and is removed from the map. The higher a unit's health the more times the unit can be hit by enemy abilities.
- **Strength** is a numerical value that affects how much damage a unit's ability does to an enemy unit. The higher a unit's strength, the more health an enemy unit loses when struck with the unit's ability.
- **Armor** negates damage done to a unit from an enemy ability. Armor lowers the amount of damage normally done by an ability, so the unit will lose fewer health points when attacked.
- **Movement Speed** refers to the speed that units are able to move around a map. A unit that has higher speed will take less time to move to a target location.
- **Rate of Attack** is the speed of which a unit can use abilities. The units with a higher rate of attack are able to use their ability faster than units with a low rate of attack.

Each has different strengths and weaknesses, as detailed in Table 1. In general, the units of the Human faction are stronger while the units of the Piscivian faction are more agile.

Table 1: Unit Strength and Weakness Chart

Unit	Faction	Health	Strength	Armor	Movement Speed	Rate of Attack
Scuba Troop	Human	★★★	★★★★★	★★	★★★	★★★★
Submarine	Human	★★	★★★	★★★★★	★★★★★	★★
Bomb Ship	Human	★★★★★	★★★★★	★★★	★	★
Merman	Piscivian	★★★	★★★	★★★	★★★	★★★★
Manta Ray	Piscivian	★★	★★★★★	★	★★★★★	★★★★
Turtle	Piscivian	★★★★★	★★	★★★★★	★★	★★

The stars shown next to each field in the stats category represent the value of that particular statistic—more stars mean a higher value. The translation of stars into numerical values was done through implementation and testing to find a proper balance between units and factions.

While stats define the different strengths and weaknesses of a unit, each unit also specializes in either a melee ability or a ranged ability. Units that specialize in a melee ability must be near an enemy to attack, while units that specialized in a ranged ability can attack an enemy units from afar.

We created variation in the unit stats because we based our units on traditional RPG classes. We classified the merman and the scuba troop as a class similar to a basic soldier from other RTS games. The basic soldier units are balanced in attack strength and armor. The manta ray and the submarine resemble an assassin class, with the ability to be quick and stealthy and strong, but lack armor. The turtle and the bomb ship are a mobile siege unit, with great strength and defense but move slowly.

In this chapter we will first discuss the differences between Human units and Piscivian units, unit sound effects, how unit movement works, and then how the different commander roles function.

3.1. Human Units


We were aiming for a futuristic, creative art style that borrows familiarity from World War II to create a sense of modern realism for all Human units. All of the Human

models have a red, black, and gray color scheme to reflect their faction’s artistic style. Additionally, this color scheme is reflected in the icons for the Human faction. Red icons were created for the each of the units in the Human faction and their ability. These icons would be displayed in the tactical-commander interface discussed in Chapter 6: User Interface.

3.1.1. Scuba Troop

The scuba troop is similar to the basic soldier class in other RTS games and functions as the most well balanced unit for the Human faction; however, the scuba trades some extra attack for a loss in armor as shown in Table 2. The scuba troop uses his hand blade ability, which is a basic melee attack, to strike enemies at a close range.

Table 2: Scuba Troop Unit Profile

Unit	Stats	Ability	Ability Description
Scuba Troop 	Health: ★★★ Strength: ★★★★★ Armor: ★★ Movement Speed: ★★★ Rate of Attack: ★★★	Hand Blade 	Hand Blade - basic melee attack with spinning propeller blade(<i>Melee</i>)

The scuba troop was designed to look similar to traditional scuba divers in terms of equipment, as shown in Figure 9. We chose the scuba troop unit because scuba diving is a very common and accessible way for modern people to interact with the underwater world. However, as scuba diving is generally a peaceful endeavor, we gave this unit hand blades in order to enhance their fighting ability. We also included the scuba troop to encompass the faction’s name - “Humans.” The scuba troop is the only unit on the Human faction that portrays human-life. Because we wanted players to relate to the Human race and their need to survive, the scuba troop was a mandatory unit to both humanize the Human faction and maintain familiarity with a modern audience.

The scuba troop we created for the game, as shown in Figure 10, maintained its original design as a scuba diver from Figure 9, including the two hand blades that acted

as a propulsion system and a method of attack. His color scheme corresponds to the factions color scheme as shown in the black and red wetsuit.



Figure 9: Scuba Diver (“Scuba Diving 03 Jpg”)





Figure 10: Scuba Troop In-Game Model

The biggest changes from the original image to the created model can be seen in the wet suit and the location of the equipment. Scuba divers in warm waters like Hawaii traditionally wear more skin-exposed wetsuits, as shown in Figure 9; however, because the scuba troop is a soldier, we felt that the entire body should be covered by a wetsuit. Also, we chose to lessen the equipment that a normal scuba diver uses because we wanted to simplify the character and stream-line him for battle. We did not want large and obstructive gear to hinder this unit’s fighting ability.

3.1.2. Submarine

The submarine is similar to assassin class in other RTS games and functions as the fastest unit for the Human faction; however, the submarine is not like the typical assassin unit of being fast moving, having high strength, and having low armor. Instead the submarine trades higher strength for some extra armor as shown in Table 3. The submarine strikes enemies from afar using its torpedo.

Table 3: Submarine Unit Profile

Unit	Stats	Ability	Ability Description
Submarine 	Health: ★★ Strength: ★★★ Armor: ★★★★★ Movement Speed: ★★★★★ Rate of Attack: ★★	Torpedo 	Torpedo - fires a torpedo (<i>Ranged</i>)

Submarines were designed to be small personal vessels with a body that resembles WWII submarines as shown in Figure 11. We chose submarines as one of the Human units because these machines were built for underwater battling capabilities; therefore, the structure was left mostly unchanged. In addition, submarines already reflect the WWII theme, so only minor alterations were needed to make the ship fit with the futuristic theme. To incorporate the futuristic theme, we planned to wrap the submarine with a clear metal coating sealed together with titanium rings, which is similar to the glass architecture seen on the Hydropolis Hotel. However, as can be seen in Figure 12, the implemented model for the submarine was changed from the initial design.



Figure 11: Submarine Reference (Lee)



Figure 12: Submarine In-Game Model



While we planned on incorporating a futuristic theme into the submarine's design, only the WWII art style was implemented into the model. This change was due to a problem with outsourcing, which is further discussed in Chapter 10: Post Mortem. The outsourced creator of the submarine had a different artistic style, and the project's art team did not have access to the original model; therefore, the art team was unable to make adjustments in order to harmonize the submarine with the Human faction's art style.

A similar problem occurred with the size of the model. Instead of enlarging the tiny submarine to accurately correlate to its actual size, the outsourced creator justified the size of the submarine by deeming it remote controlled. Because the art team did not have access to the original model, we were unable to resize the submarine.

3.1.3. Bomb Ship

Bomb ships are slow moving units that boasts high strength and armor, which reflects the siege tank class from other RTS games. The bomb ship has both the highest health and highest strength in the game but also has the lowest movement speed and rate of attack as shown in Table 4. The bomb ship’s stats coincide with the overall vision of the Humans specializing in strength over speed. The bomb ship fires an explosive cannon ball at enemies from afar.

Table 4: Bomb Ship Unit Profile

Unit	Stats	Ability	Ability Description
	Health: ★★★★★ Strength: ★★★★★★ Armor: ★★★ Movement Speed: ★ Rate of Attack: ★		Cannon - fires a an explosive cannon ball (<i>Ranged</i>)

Bomb ships were designed to resemble aircraft carriers during World War II, similar to Figure 13, and borrowed the flat top and immense size of aircraft carriers. The bomb ship would have protruding titanium spikes at the front of the ship, which resembles the crown-like structure on top of the Hydropolis Hotel. Additionally, bomb ships would have several shiny fins on the port, starboard, and aft similar to architecture seen on the Hydropolis Hotel. The bomb ship was originally designed to have a blue and white exterior to suit the aquatic theme and for camouflage; however, the unit was given a red and black exterior to harmonize with the overarching color scheme for the Human faction as shown in Figure 14. We chose to create a new ship because, as this game is set in the future, we did not want to rely solely on modern technology. The bomb ship was designed to be the most futuristic of the three Human units and house the strongest artillery.



Figure 13: Bomb Ship Reference (DaChief)



Figure 14: Bomb Ship In-Game Model

The bomb ship model maintained the initial design of resembling an aircraft carrier. The bomb ship keeps consistent with the aircraft carrier top, as well as the port holes around the “oar” holes.



3.2. Piscivian Units

We were aiming for realism and used images and videos of an underwater environment to create realistic models for all the Piscivian units. All of these models have a blue-green color scheme to reflect their faction’s artistic style. The color scheme was also used to create blue icons for the Piscivian units and abilities. These icons would be displayed in the tactical-commander interface discussed in Chapter 6: User Interface.

3.2.1. Merman

The merman acts as the basic soldier for the Piscivian faction, similar to the scuba troop for the Human faction. The merman is the most balanced unit in the game and has no strengths or weaknesses as shown in Table 5. The merman uses his trident ability, which is a basic melee attack, to strike enemies at a close range.

Table 5: Merman Unit Profile

Unit	Stats	Ability	Ability Description
	Health: ★★★ Strength: ★★★ Armor: ★★★ Movement Speed: ★★★ Rate of Attack: ★★★		Trident - basic melee attack with trident (<i>Melee</i>)

The merman unit was designed to resemble the Greek god Poseidon - an attractive, muscular, bearded man with a fish tail seen in the statue in Figure 15. To incorporate the overarching Hawaiian beach theme, the merman unit's fish tail resembles the Picasso triggerfish, which is the state fish of Hawaii. The human half of the merman was originally designed to have short brown hair and a short full beard; however, due to engine constraints animating hair was out of the scope of this project. Therefore, our implemented merman has a shaved head and no beard.

The merman in our game, as shown in Figure 16, has the human body like a swimmer with broad shoulders and thin hips to reflect the amount of time spent in the water. In order to keep the model simple and reduce the polygon count, we did not build geometry for muscles. Instead, we chose to paint the muscles on the texture to give the illusion of muscles.



Figure 15: Statue of Poseidon (“Poseidon / Neptune God of the Sea”)



Figure 16: Merman In-Game Model



As you can see in Figure 16, the transition from human to fish was done abruptly. The merman model resembles our original design of having a half human, half fish model, similar to fictitious depictions of Poseidon, seen in Figure 15. However, we removed the fins protruding from the arms, because we wanted equal differentiation between human and fish in the model. We chose this appearance because of its familiarity to a general audience. Additionally, the merman’s trident was kept similar to the original concept except we included two additional sharp edges to add an intimidating feel to the trident.

3.2.2. Manta Ray

The manta ray is the counterpart to the submarine and resembles the assassin class in other RTS games. The manta ray functions as both the fastest unit in the game

and the strongest unit in the Piscivian faction; however, the manta ray has the worst health and armor in the game as shown in Table 6. The manta ray uses its tail whip ability to strike enemies at close range.

Table 6: Manta Ray Unit Profile

Unit	Stats	Ability	Ability Description
Manta Ray 	Health: ★★ Strength: ★★★★★ Armor: ★ Movement Speed: ★★★★★★ Rate of Attack: ★★★	Tail Whip 	Tail Whip - the manta will spin around whipping surrounding units with its tail (<i>Melee</i>)

The manta ray unit has a shiny black top and a cream underside that resemble manta rays commonly seen off the coast of Hawaii as shown in Figure 17. We used images and video with manta rays to make this unit realistic and move correctly.

We chose a manta ray unit to take advantage of this creature’s fierce look, speed, and deadly potential against humans. While manta rays are usually peaceful animals and use their tail to stealthily move through the water instead of for combat, we wanted to represent manta rays in the future when threatened and forced to fight to save their race. The implemented manta ray model is shown in Figure 18.



Figure 17: Manta Ray Reference (de Pastino)

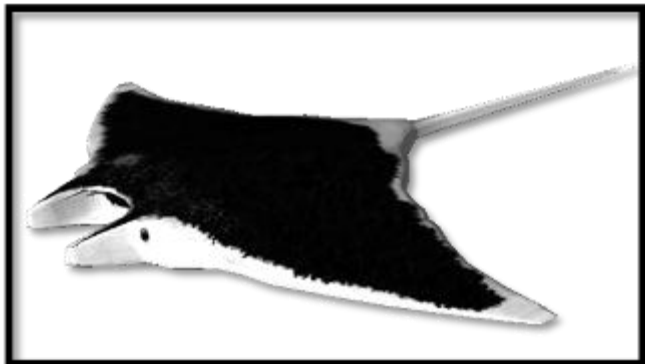


Figure 18: Manta Ray In-Game Model

The manta ray model closely corresponds to the original design; however, unlike the reference, the model’s texture deviates slightly from a real manta ray. Most specifically, the eyes are slightly different; in the model, the eyes point more forward

than the original image. Despite the small changes from the original design, the size and fluidity of a real manta ray was captured in the animations.

3.2.3. Turtle

Turtles are slow moving units that boasts high health and armor, which reflects the siege tank class from other RTS games. The turtle's complement on the Human faction is the bomb ship. While the bomb ship specializes in higher attack, the turtle has low attack but higher movement speed and armor as shown in Table 7. The turtle uses its wave cannon, which has the largest attack range of all unit abilities but must take time to recharge after fired.

Table 7: Turtle Unit Profile

Unit	Stats	Ability	Ability Description
	Health: ★★★★★ Strength: ★★ Armor: ★★★★★ Movement Speed: ★★ Rate of Attack: ★★		Wave Cannon - fires a wave with a long cool-down (<i>Ranged</i>)

The turtle unit looks similar to the common green sea turtle found off the coast of Hawaii as shown in Figure 19. We chose the turtle unit because of their strong and sturdy appearance and their longevity. Because turtles are usually peaceful animals we needed to give the turtle more attack versatility as a unit. Therefore, we included a conch shell cannon on the turtle's shell that fires wave lasers. This weapon also changes the turtle from a docile creature to a warrior, as shown in Figure 20.



Figure 19: Turtle Reference (Sangster)



Figure 20: Turtle In-Game Model

The turtle model reflects the original design for the turtle, as shown in the reference image in Figure 20. We simulated the shell and skin of a green sea turtle from real photographs and created textures for the turtle model. Other than the addition of the shell cannon, the model kept consistent with both the design and the real creature, both in model and animations.

3.3. Unit Sounds

Each unit type will have three different sound effects – movement sound, attack sound, and accept order sound. These sound effects are detailed for the Human faction in Table 8 and the Piscivian faction in Table 9. The movement and attack sounds can be faintly heard when playing as the tactical-commander; however, while playing as the field-commander, the sounds are heard when in range of the source. The accept order sound effects are only heard while in the role of the tactical-commander.

Table 8: Sound Effects for Human Units

Human Faction Sounds			
Units	Move SFX	Attack SFX	Accept Order SFX
Scuba Troop	This sound effect is used whenever the scuba troop is moving around the map. A faint buzzing sound of propellers is played as the scuba troop moves around the map. Additionally, the player hears bubbles from the air tank when using a scuba troop as a field-commander.	The attack sound effect is used when the scuba troop unit is attacking an enemy unit or building. The sound of a razor blade cutting wood plays when the scuba troop attacks. A sound of rushing water will follow.	The scuba troop signals that he has received orders by playing the accepting orders sound effect. A heavy breathing noise, similar to the sound Darth Vader makes, sounds when the scuba troop accepts an order. This will follow by bubbles from the air tank.
Submarine	This sound effect is used whenever the submarine pilots around the map. A faint buzzing sound of propellers plays as the submarine moves around the map.	There will be no sound for when the torpedo is fired, because a torpedo breaks the speed of sound. However, there will be a small explosion sound that is played after a target is hit.	The submarine signals that it has received orders by playing the accepting orders sound effect. A submarine sonar sound plays when the submarine accepts an order.
Bomb Ship	This sound effect is used whenever the bomb ship is maneuvering around the map. A swishing noise and water rushing noise sounds when the bomb ship moves its metal fins to move through the water.	The attack sound effect is used when the bomb ship unit is attacking an enemy unit or building. A cannon sound plays when the bomb ship attacks. An explosion sounds when the bomb ship hits the target.	The bomb ship signals that it has received orders by playing the accepting orders sound effect. A water rushing sound will play to show the bomb ship has accepted the order.

Table 9: Sound Effects for Piscivian Units

Piscivian Faction Unit Sounds			
Units	Move SFX	Attack SFX	Accept Order SFX
Merman	This sound effect is used whenever the merman is swimming around the map. This sounds effect will be similar to a fish swimming through water or a water rush sound.	The attack sound effect is used when the merman unit is attacking an enemy unit or building. This sounds like a swish followed by a rushing water sound.	The merman signals that he has received orders by playing the accepting orders sound effect. It will play a small song-like choir chant.
Manta Ray	This sound effect is used whenever the manta ray is moving around the map. A water rushing sound effect is played when the manta ray flaps its wing-like fins.	The attack sound effect is used when the manta ray unit is attacking an enemy unit or building. This sounds like a swish followed by a rushing water sound.	The manta ray signals that it has received orders by playing the accepting orders sound effect. Bubbles followed by rushing water will designate that the manta ray has accepted an order.
Turtle	This sound effect is used whenever the turtle is swimming around the map. A sound of water rushing is played when the turtle flaps its fins.	The attack sound effect is used when the turtle unit is attacking an enemy unit or building. An inhaling noise will play to charge the cannon and will follow with a swish with an echo.	The turtle signals that it has received orders by playing the accepting orders sound effect. A turtle cry sound effect is heard to show the turtle has accepted an order.

In addition, several sounds such as death and spawn will be used for both factions. These sounds are detailed in Table 10.

Table 10: General Unit Sounds

Name of Sound	Use of Sound	Description of Sound
Unit Spawn	The unit spawn sound will be used when a wave of units have spawned.	This will sound similar to a bell chime.
Field-Commander Spawn	The field-commander spawn sound will be used when the field-commander has spawned for the first time or has re-spawned after death.	This will sound like a choir arpeggio.
Organic Unit Death	The organic unit death sound will be used when any non-mechanical NPC unit dies. This sound effect will be used for the Merman, Manta Ray, Turtle, and Scuba Troop units.	A burst of bubbles sound is played when an organic unit dies.
Mechanical Unit Death	The mechanical unit death sound will be used when any non-organic NPC unit dies. This sound effect will be used for the Submarine and Bomb Ship units.	This will sound like a small underwater explosion.

3.4. Steering

Units use a steering component to decide how to move around in the environment. Steering works by calculating forces that are applied to a moving entity. These forces are dictated by various behaviors.

Each steering object has an internal target, which it uses to generate the forces on a unit. These forces determine the new position of a unit. Once the steering component determines that a unit is not on a target, it goes through the list of behaviors and applies forces to both entity location and entity heading. The forces that are applied are behavior specific, such as moving forward for a seek behavior. In general, if a unit is not facing the target, it will not move until it has turned to face the target.

We have programmed a few behaviors, but the main actions are seeking, avoiding, and interpolating. These behaviors will be discussed in detail in the following sections.

3.4.1. Seeking

The first behavior is seeking. This is the main behavior and is used by every unit that needs to move. Seeking works by moving a unit along a vector to some target, as shown in Figure 21.

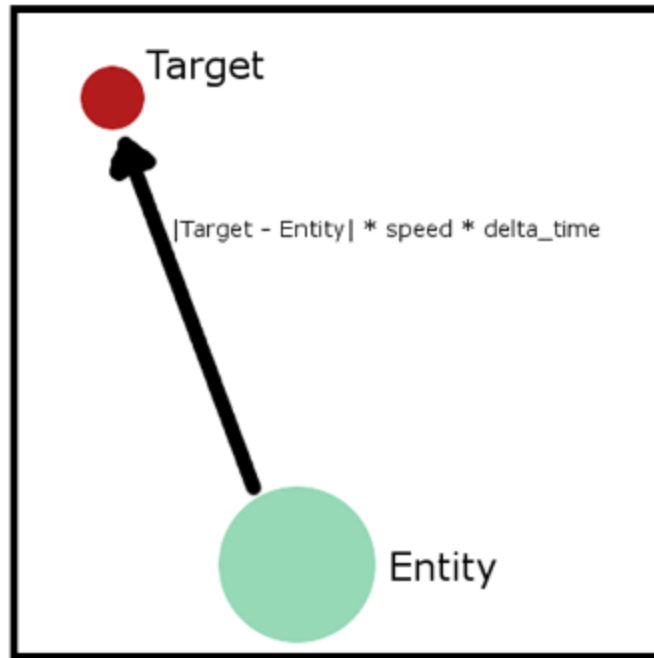


Figure 21: Seeking

The figure shows how seeking calculates an entity's motion. When an entity, such as a unit, is given a target, the distance that the unit moves along the vector is dictated by the entity's speed and the time that has passed since the last update, or delta time. This behavior is used as the main method of travel.

3.4.2. Avoidance

Since there are many units that might be moving around a map, they need to avoid each other so they do not collide. The avoidance behavior guides units away from each other. When a unit determines where it will move to, the steering component will check if there are any other units in front of it. If there are, an intermediate target will be generated so that this unit will avoid the obstacle, as shown in the example in Figure 22.

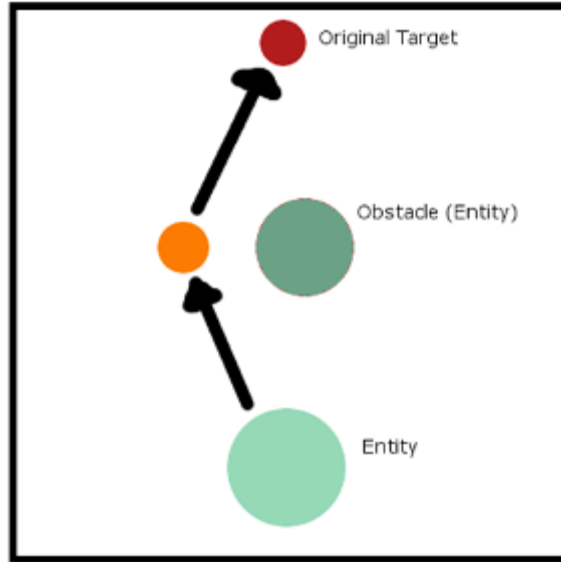


Figure 22: Avoidance

The figure shows what would happen if an entity, or unit, detected an obstacle in its path. Instead of traveling directly to the red original target, the unit will instead travel to the orange intermediate target before moving on to the original target.

3.4.3. Interpolation

The last behavior is interpolation. This behavior moves units to new positions based on predictions of where the unit should be. The interpolation behavior is only used on the client machine. Our client-server architecture dictates that all important calculations must be done on the server, meaning that the server needs to send position updates to the clients. Clients do not use artificial intelligence or steering because the server is in charge of keeping track of locations. The client continually predicts locations while it waits for server updates as shown in Figure 23.

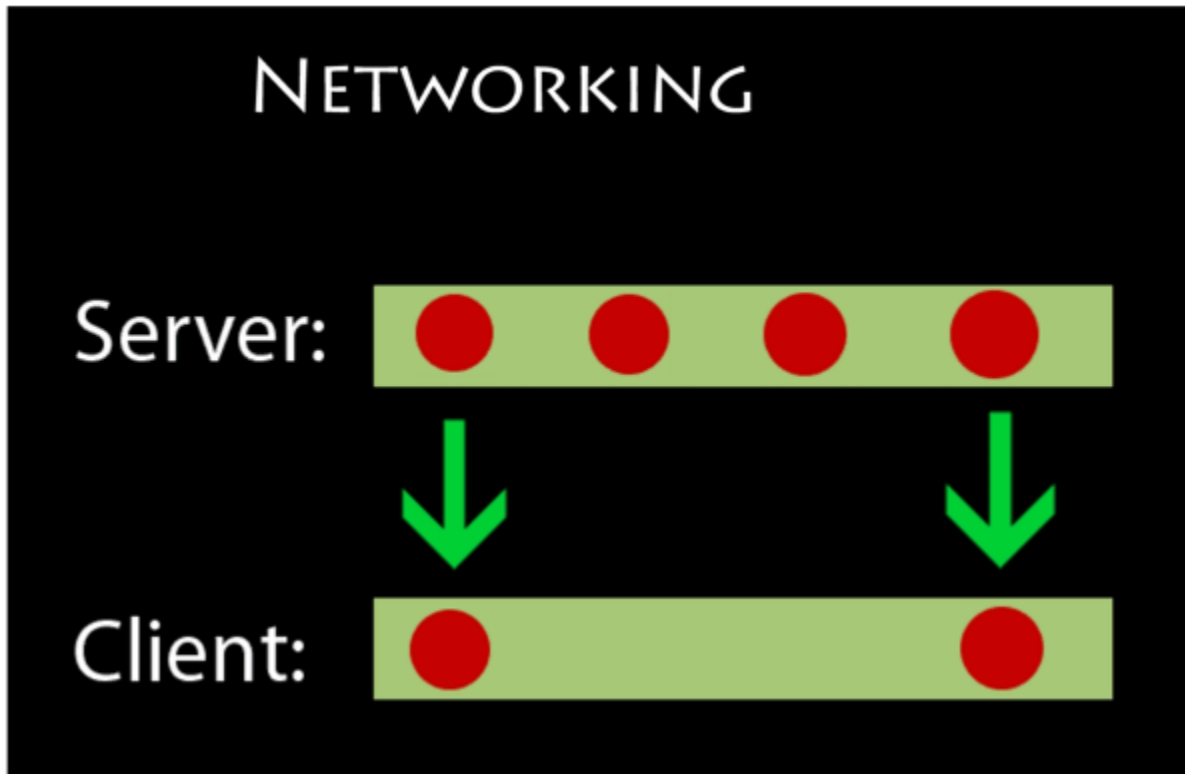


Figure 23: Interpolation

Clients have all of the same information as the server because the string tables are shared. This means that clients can perform movements similar to seek. This could possibly lead to the client having different position information than the server if the client's machine is running faster or slower. To avoid this, the server sends updates every 125 milliseconds, which replace the client's information with the accurate server information.

3.5. Unit Abilities

All units can perform attacks and defensive abilities. When a unit is in range to attack an enemy unit, it automatically carries out the action of attacking. To do this, the unit has to play an attacking animation, the damage of the attack must be calculated, and both players must be informed whether damage has been dealt.

Abilities are generic actions that units can perform, like melee attacks, shooting projectiles, and healing. All of the functionality for these abilities is specified in the

string tables, as shown in Figure 24. The figure shows a complete ability definition that gets parsed and used by bomb ships.

```
'abil' "Cannon"
{
    'TYPE' "PROJECTILE"
    'PROJ' "CANN"
    'DELA' "4000"
    'COOL' "8000"
    'STAT' "REGULAR"
    'DMG' "80"
    'ANIM' "animation/units/bombship/bombship_attack"
    'ICON' "texture/ui/icons/weapons/cannon_icon"
}
```

Figure 24: Ability Definition

Since models, animations, and abilities are all defined in the string tables, creating game units is very easy. If we wanted to add a new type of unit, we could mix and match how the unit looks and how the unit behaves based on the available functionality by giving the unit different kinds of abilities in the string table. For example, we could make the merman shoot cannonballs, like the one in Figure 24, instead of using the trident. Since the actions we have built into the game encompass many actions that similar games have, we have a wide range of behaviors that we can express.

3.6. Commander Types

The game starts the player in the tactical-commander role, allowing the control the faction's units and gives the player the ability to construct buildings. In this role, the player can select the race of the field-commander. The field-commander can gain levels and unlock two new abilities that the basic units do not have. The field-commander cannot be controlled while in tactical-commander mode, but the player can switch between the two roles at any time during the game.

3.6.1. Tactical-Commander

The tactical-commander is played similar to an RTS game with a bird's eye view. When the game begins, the player starts in the tactical-commander role. As the tactical-

commander, the player has the ability to issue orders to units and construct buildings. The orders that can be issued are: move to position, attack to position, or attack enemy unit. From this role the player can enter the field-commander role and will be returned to this role if the player's field-commander is killed.

In addition to issuing orders, the player can also build three types of buildings, which are described in section 4.3. The tactical-commander must strategically construct buildings to strengthen the faction's units and protect his or her bases.

3.6.2. Field-Commander

The field-commander is played like an action game with a third-person chase camera. At the start of the game session, the player has the ability to choose which type of field-commander to play as. Each headquarters will spawn a different type of field-commander that coincides with the regular units that spawn from that headquarters. If a player wishes to change to a different field-commander type, the player must move the field-commander back to any friendly headquarters.

All units and the field-commander will start with one attack. For instance, both a unit and field-commander submarine will start off with the torpedo ability; however, only the field-commander is capable of unlocking the next two abilities for the unit type. These abilities are unlocked for the field-commander upon promotion, which are stealth and barrage for the submarine commander. When a training facility is built, both the field-commander and the normal units of that type will be upgraded. The detailed list of field-commander abilities for the Human faction can be seen in Table 11, and the detailed list of field-commander abilities for the Piscivian faction can be seen in Table 12. Unlike units only specializing in either melee and ranged abilities, a field-commander can also have a support ability, which gives a passive bonus to the unit to aid in battle, or an area of effect (AOE) ability, which is capable of hitting multiple targets at once.

Table 11: Human Faction Field-Commander Abilities

























Unit	Rank	Ability	Icon
 Scuba Troop	1	Hand Blade - basic melee attack with spinning propeller blade (<i>Melee</i>)	
	2	Toxic Dagger - fires a hand blade that poisons the enemy (<i>Ranged</i>)	
	3	Blade Counter – blocks an attack and counters with the hand blade (<i>Support</i>)	
 Submarine	1	Torpedo - fires a torpedo (<i>Ranged</i>)	
	2	Stealth – hides the unit for a period of time (<i>Support</i>)	
	3	Barrage - fires several torpedoes in quick succession (<i>Ranged</i>)	
 Bomb Ship	1	Cannon - fires a an explosive cannon ball (<i>Ranged</i>)	
	2	Depth Charge - plant a bomb that will explode and stun enemies (<i>AOE</i>)	
	3	Recovery Units – an AOE healing move that moderately heals allies in range. (<i>Support</i>)	

Table 12: Piscivian Field-Commander Abilities

Unit	Rank	Ability	Icon
 <p>Merman</p>	1	Trident - basic melee attack with trident (<i>Melee</i>)	
	2	Laser Trident - a magic beam of light fires from trident and stuns enemies (<i>Ranged</i>)	
	3	Serenade - a AOE tribal song that gives large healing bonuses (<i>Support</i>)	
 <p>Manta Ray</p>	1	Tail Whip - the manta will spin around whipping surrounding units with its tail (<i>Melee</i>)	
	2	Riptide - hides from enemy site for a limited time and increases speed to nearby allies (<i>Support</i>)	
	3	Whirlpool - creates an underwater tornado that destroys surrounding enemies (<i>AOE</i>)	
 <p>Turtle</p>	1	Wave Cannon - fires a wave with a long cool-down (<i>Ranged</i>)	
	2	Shell Barrier - nullifies all damage done to the turtle for a short period of time (<i>Support</i>)	
	3	Bite - a fast, strong bite attack that pierces through enemy defense (<i>Melee</i>)	

The field-commander will be able to gain ranks through promotions gained with experience in battle. In order to get promoted, the field-commander must gain a certain number of combat points, which are gained by killing enemy units, headquarters, buildings, or the opposing field-commander. There are two additional ranks for the field-commander to gain. When the field-commander is promoted, he or she will become stronger and gain a new ability or attack. When the field-commander dies, the next field-commander chosen will spawn at rank one without any active upgrades.

3.6.2.1. Field-Commander Spawn

The field-commander spawn component controls where and when the field-commander will spawn on the map. This component also keeps track of time periods when a field-commander is not allowed to spawn, such as the time after a field-commander dies.

There are two key functions to the field-commander spawn component. First, when the player can spawn a commander, the commander spawn pane shows on the screen, as shown in Figure 25.



Figure 25: Commander Reselect Pane

This pane displays buttons for each type of commander the player can spawn. In addition to this, when the player moves a commander back to a base, the pane will show up again, to allow the player to pick a new commander without losing ranks.

3.6.2.2. Shaders

Originally, we planned to use a shader to create an underwater effect; instead, we used a modified version of fog to simulate being in the ocean. However, we did use a shader for the stealth effect that is used by the submarine and manta ray field-commanders in their second ability, as seen for the submarine in Figure 26.

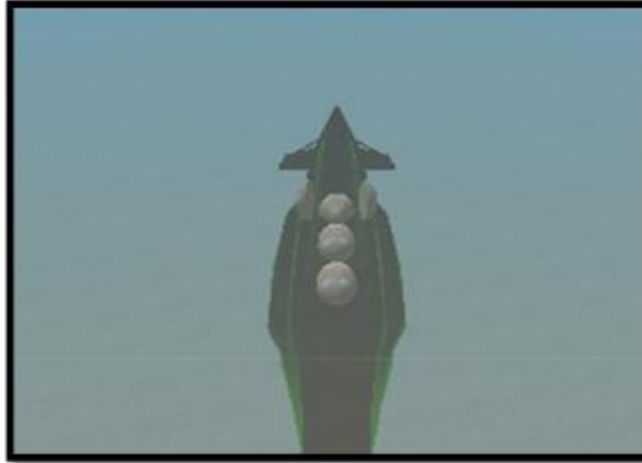


Figure 26: Ghost Effect

In order to make our stealth shader, we created a ghostly effect by modifying a Fresnel shader. This shader is then applied to a model. When we want to apply the shader to one of the units, we load the model with the shader, copy the shader, and then apply it to the model we want it on.

Chapter 4: Structures

Each of the units for the Piscivian and Human factions has its own headquarters from which it spawns. Additionally, the player may build defense towers, farms, and training facilities to improve defense at a headquarters, decrease the time between unit spawns, and increase the quality of units, respectively.

Structures are an integral part of the gameplay design, as the headquarters are the spawning points for the units. The destruction of the opponent's headquarters is the victory condition for the game. Also, the constructible buildings allow the player to customize his or her play style and add an additional layer of strategy to the game.

This chapter will discuss the artistic vision for each faction's structures, the headquarters, and buildings that can be constructed around the headquarters. This chapter discusses both the artistic and technical components that were involved in the creation of the headquarters and buildings.

4.1. Artistic Vision

The art theme of the Human faction's structures was influenced by the familiar architecture of World War II and blended with the futuristic, aquatic city feel of the Hyropolis Hotel, as shown in Figure 27. This art style combines metal and a thick layer of indestructible glass, like Armor Screen® hurricane protection glass, to create mechanical structures. The art theme of the Human faction's structures uses a combination of red, black, and grey materials.

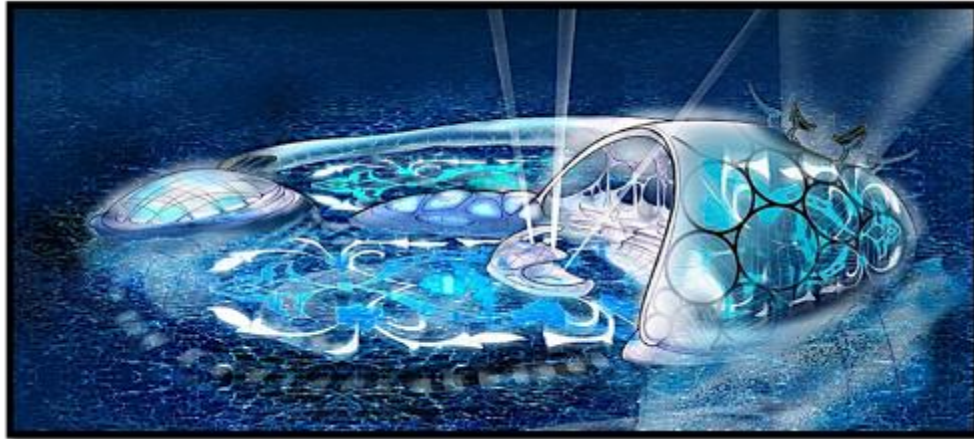



Figure 27: Hydropolis Hotel (Hauser)






The art theme of the Piscivian faction’s structures was influenced by ancient Doric Greek architecture blended with a Hawaiian white-sand beach theme. This art style combined Greek architecture and natural oceanic materials to create organic and Greek architecture. The art theme of the Piscivian faction’s structures includes a combination of blue, green, and white colors.

4.2. Headquarters

At the start of a game session, each faction begins with three headquarters. Each headquarters spawns a certain type of unit. Headquarters can only take a certain number of damage before they are destroyed and will no longer spawn units, thus removing that type of unit from the faction’s future reinforcements. The headquarters cannot be rebuilt, so the player must protect each headquarters to continue providing the faction with the same number and variety of units. Table 13 details all of the headquarters, the structure’s stats, and the icon associated with the structure.

Table 13: Headquarters Stats

Unit Headquarters	Stats	Icon
Scuba Troop HQ	Health: 800 Spawn Time: 30 seconds	

Submarine HQ	Health: 800 Spawn Time: 30 seconds	
Bomb Ship HQ	Health: 800 Spawn Time: 30 seconds	
Merman HQ	Health: 800 Spawn Time: 30 seconds	
Manta Ray HQ	Health: 800 Spawn Time: 30 seconds	
Turtle HQ	Health: 800 Spawn Time: 30 seconds	

All headquarters have two stats: health and spawn time. Health describes how much damage the structure can take before the headquarters is destroyed. The spawn time relates to how much time is between each wave of new units appearing. The icon for each headquarters is used in the interface when the structure is selected. The outline color of the icons reflects the faction that the headquarters belongs: red for the Humans, blue for the Piscivians.

4.2.1. Headquarters Controller

The headquarters controller is responsible for the status of each headquarters, as well as spawning units and field-commanders at the headquarters. The headquarters controller is placed on a node in the world editor. All of the necessary information for unit spawning, including the unit type to spawn and the interval to spawn the units, is

specified in the world editor, pictured in Figure 28. The type of units to spawn corresponds to entries in the units string table.

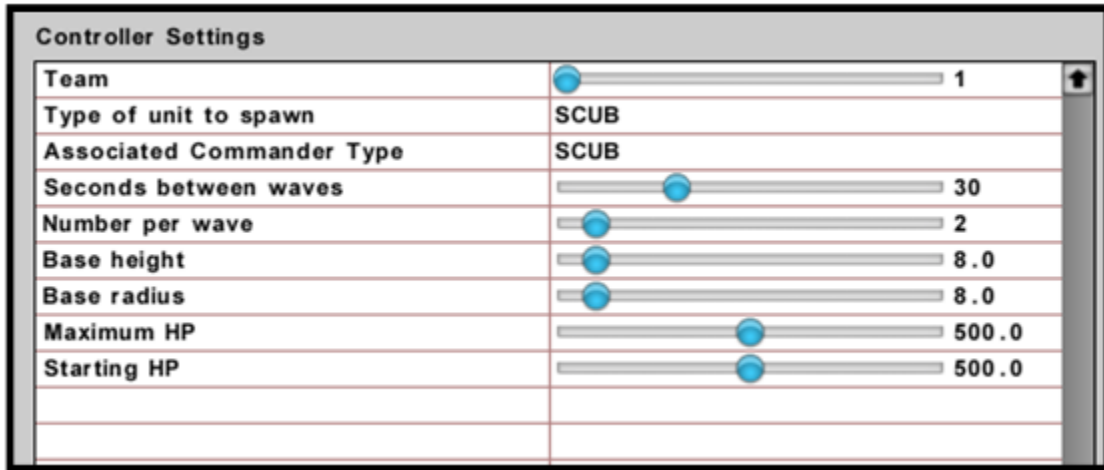


Figure 28: Headquarters Controller

We chose to implement the headquarters controller in the world editor to allow for custom maps, such as the tutorial level. This also allows for other users to create new structures to use as new types of headquarters.

4.2.2. Scuba Troop Headquarters

The scuba troop headquarters was designed to be an underwater barracks that resembles the Hydropolis Hotel, as shown in Figure 27. The headquarters was planned to be comprised of a spherical building, like the Hemisfèric in Valencia, Italy in Figure 29; however, the materials would feature an all-glass structure with minimal metal reinforcements.

This building was intended to look like an army barracks, because the scuba troop's class is similar to a modern military soldier. However, as traditional American military barracks have a simple structure, we decided to completely change the architecture to fit with the futuristic theme of the game. The implemented model is shown in Figure 30.



Figure 29: The Hemisfèric(David K.)

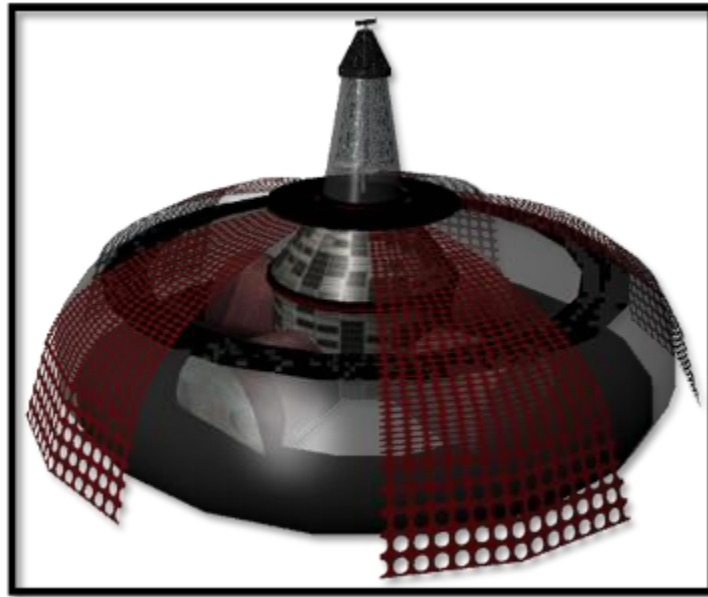


Figure 30: Scuba Troop Headquarters In-Game Model

While some of the elements of the original design remained, like the circular theme, the scuba troop headquarters was created to have a more military station feel. This included adding hatch compartments and a water turbine to power the headquarters. Many of the changes from the original design were made due to changes in the unit and building's function. Also, the fan and bubbles were created to appear to provide power to the structure.

4.2.3. Submarine Headquarters

The submarine headquarters was designed to resemble a World War II submarine pen with the long tunnels in which submarines are docked, like in Figure 31.

The submarine headquarters was initially planned to be a circular structure constructed out of shiny metals and glass.

Because submarines were found in WWII, we wanted to create a headquarters similar to their actual docking station. However, as the submarines in the game are not as large as the ones in the American Navy, we chose to create a smaller headquarters to be comparable in size to the other bases. In addition, the change in materials from concrete to highly polished metal and glass is consistent with our futuristic theme, as shown in Figure 32.



Figure 31: WWII Submarine Pen (“The submarine pens at St Nazaire”)



Figure 32: Submarine Headquarters In-Game Model

Instead of the original design, the implemented model took the feel of a submarine factory. We kept the original materials for the structure by incorporating shiny metals. This structure was drastically changed from its original design due to outsourcing, specifically the differing artistic styles of the original creator.

4.2.4. Bomb Ship Headquarters

The bomb ship headquarters was initially designed to resemble a long, glass, titanium structure, similar to the Hydropolis Hotel, in Figure 27, and the National Grand Theater of China in Beijing, in Figure 33. The building was planned to have docking stations on which the bomb ships will spawn. The docking stations, as shown in Figure 34, look similar to harbors used in World War II, but with the futuristic architecture similar to the Hydropolis Hotel.

The bomb ship headquarters was designed to be a combination of several different architectures in order to create a structure that was both futuristic and functional and serve both the futuristic and WWII themes. The National Grand Theater was chosen as a reference because of its round, glass design, which is similar to the Hydropolis Hotel. Also, because the bomb ship unit is a unique design, we felt that the bomb ship headquarters should be unique as well.

During implementation, this structure was changed to resemble the main lobby of the Hydropolis Hotel, with port holes on the sides to offer light into the structure. Also, two hatches face the rear of the structure to allow for access. Water turbines, which resemble the bomb ship's back fins, were added to the top of the structure to provide a power source, as shown in Figure 35.

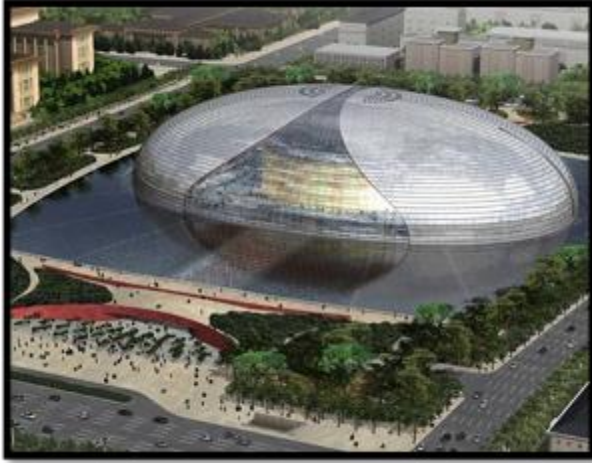


Figure 33: National Grand Theater of China in Beijing (Business Week)



Figure 34: WWII Dock (“PT 658 at Vancouver dock”)

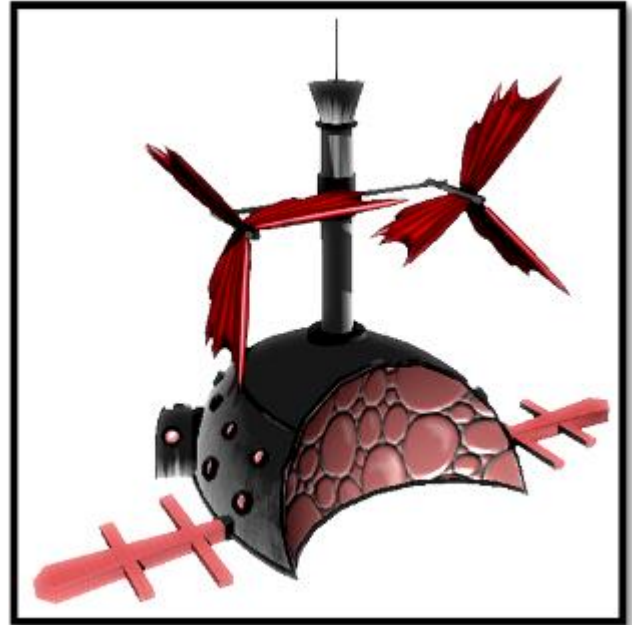


Figure 35: Bomb Ship In-Game Model

The structure of the building was changed slightly to give the building more visual appeal and to better reflect the art vision for the faction. Additionally, we added water power turbines to reflect the changes to the scuba troop base: to provide power to the structure. The water turbines on the bomb ship headquarters resemble modern water power generators, using the fins from the bomb ship unit.

4.2.5. Merman Headquarters

The merman headquarters was designed to be a tall tower that resembles a sandcastle made of white sand, similar to the one in Figure 36. Shells and starfish were embedded on the sides of the headquarters. At the top of the structure is a triangular

roof noticeable in Doric Greek architecture. Additionally, the merman headquarters is surrounded by several Doric Greek pillars.

We chose this design for the merman headquarters to convert a typical depiction of a mermaid's home in order to fit with our theme. We chose to create one large tower instead of a large array of towers because we wanted each of the headquarters to have a similar size in order to give the player the feeling that each race's headquarters have same importance, as shown in Figure 37.



Figure 36: Sand Castle Tower (Susan)



Figure 37: Merman In-Game HQ

The main structure of the headquarters was kept, though simplified to reflect a single tower rather than the castle in the reference image. The door and window design, as well as the walls, which were added from the original concept to make the headquarters appear more protected, were borrowed from the reference image.

4.2.6. Manta Ray Headquarters

The manta ray headquarters was originally designed to model a coral reef with various types of plant life and organisms living on it, as in Figure 38. The coral reef was planned to be colorful with different types of life and would rise sharply out from the water so that it would stand out from the natural environment around the Piscivian faction.

We chose this design for the manta ray headquarters because manta rays are commonly found around coral reefs to eat and be cleaned by smaller fish. Because manta rays give birth to live babies rather than eggs, this creature has no natural home; therefore, we decided the coral reef would be the most realistic habitat for the manta rays.

However, once this model was put into the game, it appeared insignificant in comparison to the environment. Therefore, we redesigned our model to include more of a Doric Greek element. The new manta ray headquarters includes a circular formation of Doric pillars, similar to a gazebo, which holds up a second floor of pillars, as shown in Figure 39.



Figure 38: Manta Ray HQ Reference (“Coral Reef”)



Figure 39: Manta Ray HQ In-Game Model

We created this design to mirror the merman’s tall, towering headquarters. This change also made the headquarters more prominent and stood out better against the environment. Even though the design was changed away from the coral reef, we wanted to include a variety of aquatic plant life, like kelp, feather caulerpa, brain coral, honeycomb coral, and sea fans. Additionally, a giant chocolate starfish clings on the side of the structure to give the headquarters additional life.

4.2.7. Turtle Headquarters

The turtle headquarters was originally designed to resemble an open underwater garden containing many clam shells. The garden would have been surrounded by Doric pillars that were tangled in kelp and other sea grasses. The design was similar to modern gardens, as can be seen in Figure 40. While no real underwater gardens exist, we modeled our design for the turtle headquarters off of a garden because they often feature plant life and manmade structures together. Gardens were also popular in ancient Greece, complementing our artistic theme.

The turtle headquarters was initially modeled as designed; however, when this was modeled it looked insignificant in comparison to the other two headquarters, and was too similar to the Piscivian farm. Therefore, we decided to remodel the turtle headquarters to resemble a Doric temple, as shown in Figure 41.



Figure 40: Outdoor Garden(gift ideas)



Figure 41: Turtle HQ In-Game Model

As the model for the Piscivan Training Grounds was changed, we felt that we could use its design for the new turtle headquarters. Doric temples are a very familiar Greek structure; we wanted to include this style of building in the Piscivian faction. However, as the shell roof was already being used, we chose to construct a traditional temple roof. To keep the white-sand beach theme, we created sand piles inside the temple to allow plant life to grow in the building.







4.3. Buildings

Surrounding the faction's headquarters is a number of plots on which the player may construct buildings. The only resource that buildings require is time, because we did not want the player to focus on resource management along with the other responsibilities. We decided that only one building could be constructed at a time so the player would have to strategically manage the order that buildings are constructed. Once

the building is complete, the player is notified and can begin constructing a new building. If a building is destroyed, the advantage that it once granted is removed and the plot becomes available for building again. However, if the headquarters that the plots are associated with is destroyed, then all the plots will be unavailable for construction for the remainder of the game session.

A building can either be a defensive structure or a passive structure that provides a bonus for the units that spawn from that headquarters. The types of buildings that the player can construct are shown in Table 14.

Table 14: Building Stats

Building Type	Description	Stats	Icons (Human / Piscivian)	
Guard Tower	A defensive structure that automatically attacks any enemy units in range.	Health: 400 Damage: 100 Range: 45 m		
Farm	A passive structure that spawns additional units at headquarters.	Health: 300		
Training Facility	A passive structure that boosts the strength of existing and future units that reside at the headquarters.	Health: 300		

All buildings have one stat for health, describing how much damage the structure can take before the building is destroyed. The damage of a structure relates to the

number of damage that is dealt to enemy units. The range describes the distance that the building can detect and attack enemy units. The icon for each building is used in the interface when the structure is selected or in the building interface. The outline color of the icons relates to the faction that the building belongs: red for the Humans, blue for the Piscivians.

4.3.1. Plot Controller

The plot controller component is responsible for each plot that surrounds a headquarters. It keeps track of what building is built upon it and acts as a mediator between buildings and headquarters. A plot controller is connected to one of the headquarters controllers, as depicted in Figure 42.

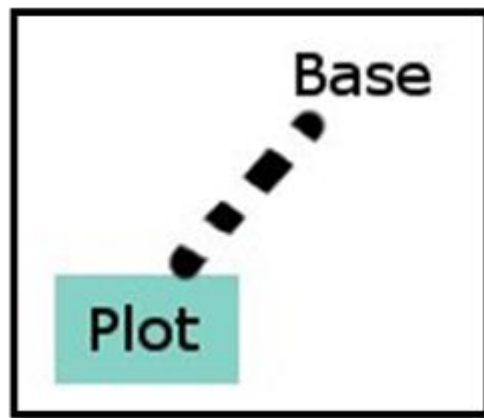


Figure 42: Plot Controller

While no building has been constructed, the empty plot model is shown in world, as shown in Figure 43 surrounding scuba troop headquarters. Once the user selects the option to build any of the buildings, the plot model disappears and the building model replaces the plot. Once the building is destroyed, the plot reappears. The buildings that can be built on a plot are specified in the world editor, as shown in Figure 44.

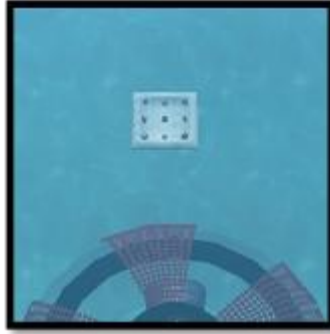


Figure 43: Implemented Plot Model

Controller Settings	
Building at start of game	
Possible Building type (1)	PGRD
Possible Building type (2)	PFRM
Possible Building type (3)	
Possible Building type (4)	

Figure 44: Plot Controller

These buildings correspond to entries in the buildings string table. We chose to implement buildings in the world editor, instead of in the engine code, to allow for custom maps to be created by other users, similar to the headquarters controller as discussed in section 4.2.1.

4.3.2. Defense Turret

The Human defense turret was designed to look similar to a World War II turret, as in Figure 45. As opposed to the traditional design, the Human turret was planned to have six barrels designed to shoot small torpedoes. We chose to copy the design of a WWII turret to correspond to our artistic theme. However, as the game is set in the future, we felt that changes to the turret were important in order to create a more powerful and futuristic weapon that would be comparable to the sea wasp tower, the Piscivian's defensive structure. The final game model is shown in Figure 46.



Figure 45: WWII Turret (Kuzmin)



Figure 46: Human Turret In-Game Model

During implementation, the turret was changed to resemble a four-way turret, with each gun having eight barrels for firing, colored in red, black, and grey metals to reflect the factions' art style. Because the player only has a small number of plots on which to build, we wanted the player to feel that this structure would be able to protect the headquarters adequately.

4.3.3. Greenhouse Dome

The Human farm was designed to resemble the hemispheric glass dome structure of a geodesic dome greenhouse in Figure 47, but changed slightly to resemble the lobby

of the Hydropolis Hotel, which has greenery and trees inside of it. The farm contains a fielded green area and a giant tree in the center of the dome. We chose to create an underwater greenhouse dome as the Human farm because this design is similar to some modern greenhouses. The dome shape, similar to the Hydropolis Hotel, is futuristic looking, which enhances our artistic vision for the Human faction, as shown in Figure 48.



Figure 47: Geodesic Greenhouse Dome (Colin)

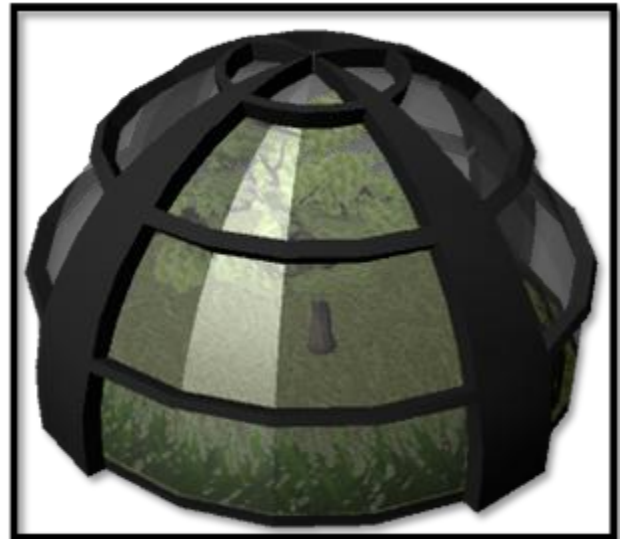


Figure 48: Greenhouse Dome In-Game Model

Instead of the geometric structure in the reference image, we modeled circular, protective rings around the glass dome. This design was used in other Human structures, such as the scuba troop headquarters, which has rims around all of the glass structures.

4.3.4. Training Bunker

The Human training facility, the training bunker, was designed to resemble a battle bunker used during World War II, as shown in Figure 49. However, instead of the cement exterior, the bunker would be made from a clear metal coating and titanium architecture. The Human training bunker was designed after WWII bunkers because the structure is a familiar design and also has a very safe and secure feel. The change in

materials from concrete to metal was done to further enhance the futuristic feel of the faction, as shown in Figure 50.



Figure 49: WWII Battle Bunker (Teskem)

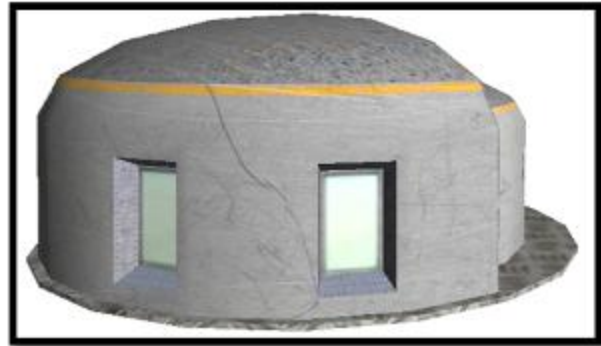


Figure 50: Training Bunker In-Game Model

This model was changed slightly due to issues with outsourcing and differing artistic styles. The exterior of the bunker reflects the Human faction's futuristic feel that the structure itself did not, by using the metal coating and glass windows. In addition, the worn textures give a similar used feeling as the bunker in the reference picture.

4.3.5. Sea Wasp Tower

The Piscivian defense tower was designed to resemble a tall Doric pillar with a giant jellyfish on top, which fires electric beams at enemy forces, as shown in Figure 51. We chose to name the tower after the box jellyfish, nicknamed sea wasp, because is considered a deadly jellyfish found in parts of Hawaii. The jellyfish's fatal stinging ability and long stingers are the reasons we chose it for the Piscivian's defense building. Figure 52 shows the implemented model of the sea wasp tower.



Figure 51: Sea Wasp Tower Reference (Andrei)



Figure 52: Sea Wasp Tower In-Game Model

While the design of the building did not change, the visual representation of the jellyfish was changed because of engine restraints. We found it difficult to create transparencies in the C4 Engine, especially when using the fog shader. Therefore, we changed the model slightly to have a pearly white texture. Also, we were hoping to make the tentacles sway with the water currents; however, we had difficulty getting this aspect to work for this building.

4.3.6. Kelp Garden

The Piscivian farm, the kelp garden, was designed to be a sun starfish that is held up by several Doric pillars, similar in structure to Figure 53. Instead of a path underneath the structure, as in the image, aquatic plants and sea grasses would grow, similar to those that are found in the Hawaiian region. This building was planned to feature a variety of colorful plant life to compliment the Picivian's lush and delicate ecosystem. Also, the choice of garden's sun starfish "roof" was because of their commonality on coral reefs and their diet of plants and plankton. The final kelp garden model is shown in Figure 54.



Figure 53: Greek Garden(Boharsik)



Figure 54: Kelp Garden In-Game Model

Originally, a sun starfish was going to be placed on top of the pillars. However, the current model features a common five-armed starfish. We chose to change the species of starfish because the sun starfish is less familiar to the audience. Also, the many arms made the building too complex and hard to visually understand from the tactical-commander role's distance.

4.3.7. Training Grounds

The Piscivian training facility, the training grounds, was designed to resemble a Doric temple, as shown in Figure 55, and would have a giant shell on the top of the building. Training equipment such as swords and dummies was planned to be inside the structure. We created the training grounds to mimic a Doric temple to reinforce our ancient Greek theme and the architecture's familiarity and its use of the Doric pillars. The temple was changed, however, to include a seashell top to incorporate our beach theme as well. We also added various training equipment to show the building's purpose, as shown in Figure 56.



Figure 55: Greek Temple (Davies)



Figure 56: Training Grounds In-Game Model

The Piscivian training grounds structure was modeled using Doric pillars to support a shell roof. Unlike the original design in Figure 55, this building is surrounded on three sides by rock formations. Also, the back columns were shrunken, which creates a clamshell feel to the building. This structure was changed from its original design due to outsourcing, specifically the differing artistic styles of the original creator. Similar to the problems with the submarine unit, we were not able to make changes to the original structure to make it better resemble the original design.

Chapter 5: Environment

The battle between the Humans and the Piscivians occurs on the ocean floor of Earth. The level contains aquatic scenery including coral reefs, bubbles, kelp, underwater ruins, and schools of fish. The scenic environment further immerses the player in the underwater setting that we envisioned for *Blood Tide*. In addition, the environment is used to create natural boundaries around the battleground because we want the battlegrounds to be a set size. Two types of boundaries were used: ocean drop-offs and underwater cliffs. The environment drastically changes around each of the faction's headquarters, as shown in



Figure 57 and Figure 58.



Figure 57: Human Side of Game World



Figure 58: Piscivian Side of Game World

The environment is important for capturing the artistic feel of each faction and reinforcing the story of our game. The map was created to portray a barren, war-torn territory on the Human side of the map, as shown in Figure 57, and a lush ecosystem on the Piscivian side of the map, and seen in Figure 58.

This chapter will discuss the game map, the fog of war used to simulate the underwater feel of the game, and the specific 3D models created to enhance our game's artistic vision and complement the game story.

5.1. Map

The map was designed to be a realistic depiction of the ocean floor. We used rock walls and drop-offs to surround the play area. The map is divided into two sides; the Piscivian faction on one end and the Human faction on the opposite end. At the Piscivian end of the map, the plant life is large and abundant. However, the plants gradually become less frequent near the Human faction headquarters and becoming more barren, with remnants of humanity's past life on the surface. This division was created to reinforce the Piscivian's delicate ecosystem that the Human Faction is endangering, as shown in Figure 59 .

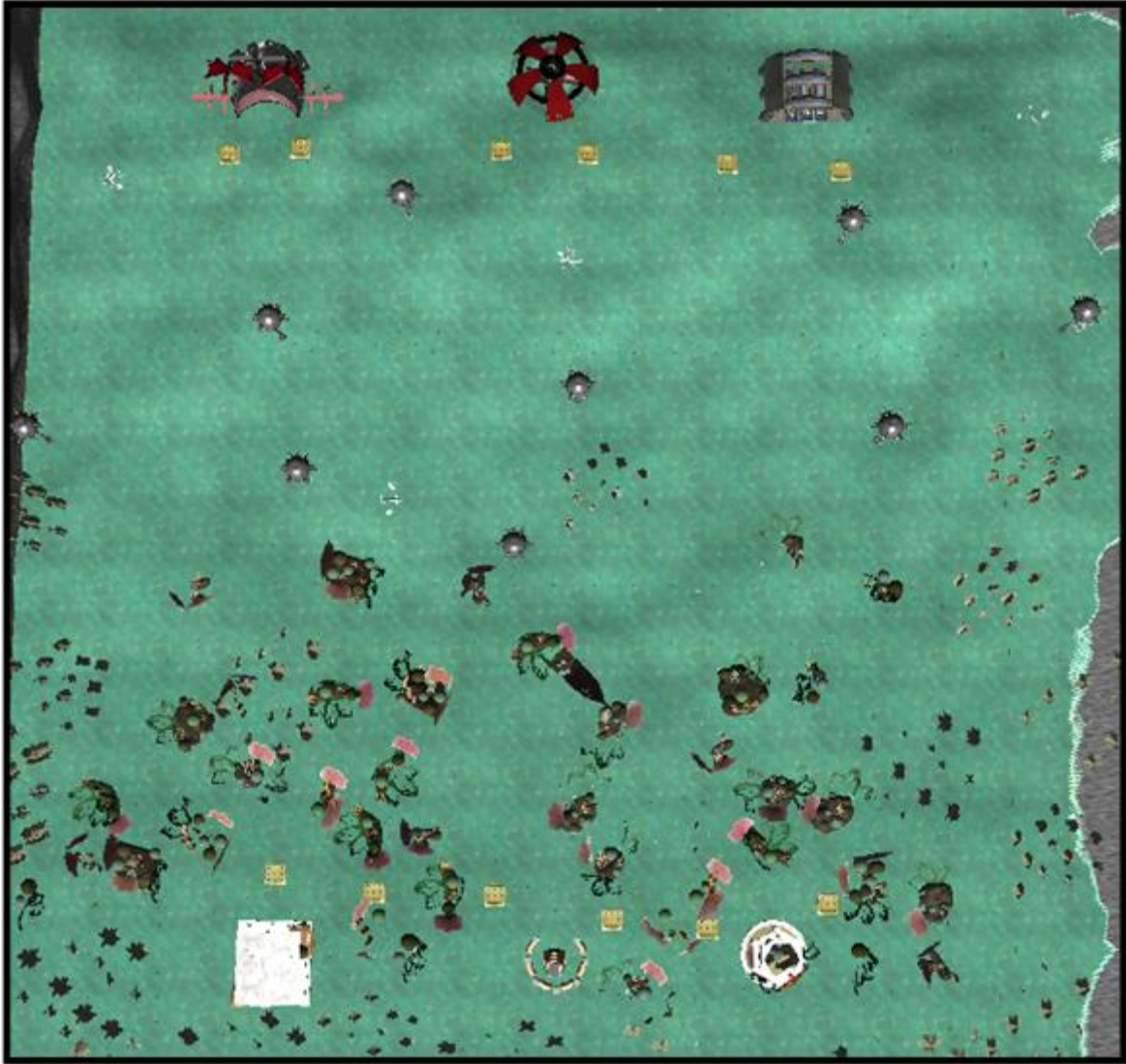


Figure 59: Game Map

As seen in the above image, the divide in the game map is gradual between the two faction's territories. In this image, which has removed the blue underwater fog, as discussed in the section below, the Human headquarters are located at the top of the image, while the Piscivian headquarters are located at the bottom of the image.

The size of the map is large enough to allow over one hundred units, but small enough so the player does not need to explore the world to find battles. This was done to force the player to be in constant conflict during the game to heighten excitement and gives the player a stronger sense of being in battle.

5.2. Fog of War

Initially, we intended to create a fog of war system that hides or shows enemy units depending on whether or not they can be seen by one of the player's units. However, because of the incompatibility of the two cameras in the RTS and action modes, we decided to get rid of this component. The goal of fog of war is to simulate what ground units can see, because RTS cameras allow players to pan around the map, which is not very realistic. Since we wanted to emphasize the field-commander role in *Blood Tide*, we decided to orient the camera to function best in this mode.

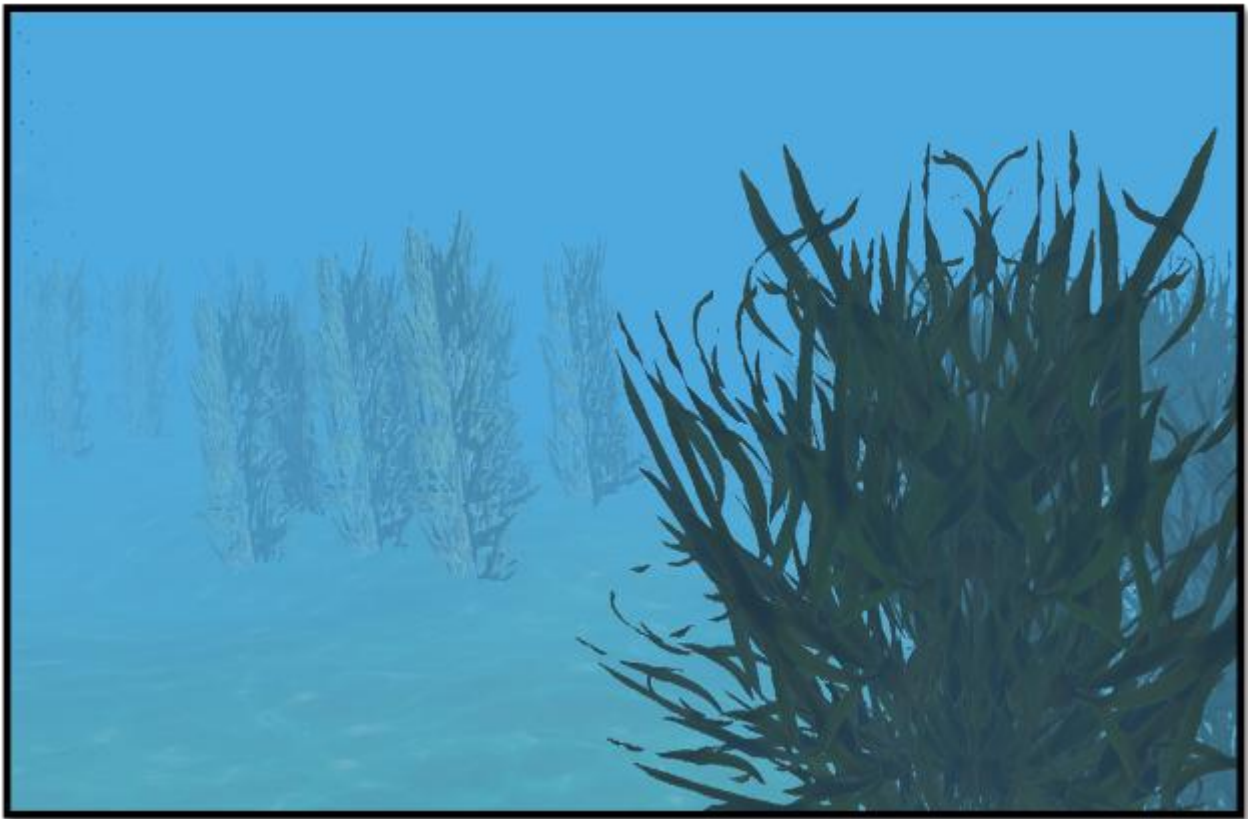


Figure 60: Example of Fog of War

Instead of hiding units at distances where the environment can still be seen, we decided to visually represent declining vision by simulating water - sight fades to blue at large distances, as shown in Figure 60. Since our game allows the player to see what the units would see, using art to create the fog of war allows the game to retain a sense of locality without arbitrarily making things invisible. This effect greatly enhances the feeling of being underwater. We used the fog system that is already built into the C4

Engine to do this. Instead of using the standard grey as the fog color, we chose shades of blue.

5.3. Environment Models

3D models were created for the environment to enhance the game vision and story. These models were categorized depending on which faction they most closely relate to. In general, the Human faction's environment relates to war and destruction, while the Piscivian faction's environment relates to nature and life.

5.3.1. Rusted Bomber Airplane

The game map features three variations of rusted bomber airplanes that have crashed underwater during World War II, as shown in Figure 61. The Human side of the map displays several crash landings.

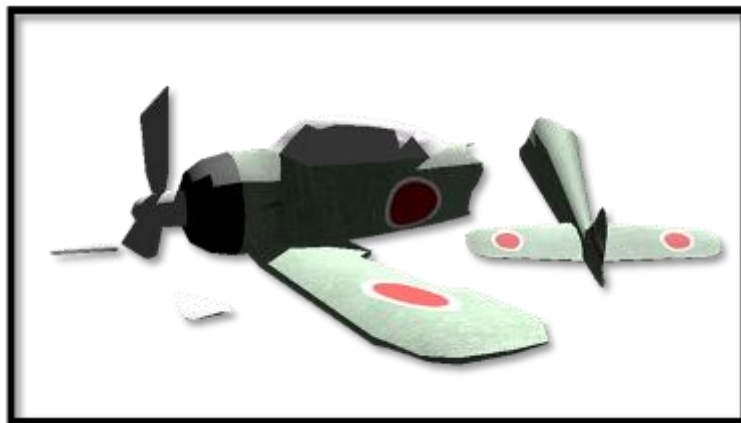


Figure 61: Rusted Bomber Airplane

5.3.2. Unexploded Mine

Many unexploded mines are scattered around the Human side of the map. They float above the plane that the units battle on so the mines do not appear to be hazardous to the player and are solely environment, seen in Figure 62.



Figure 62: Underwater Mine

5.3.3. Pirate Ship

One pirate ship is located near the center of the map lodged in some coral reefs. This model represents the division from the Human to Piscivian territory. The pirate ship has torn sails and a broken mast, shown in Figure 63.

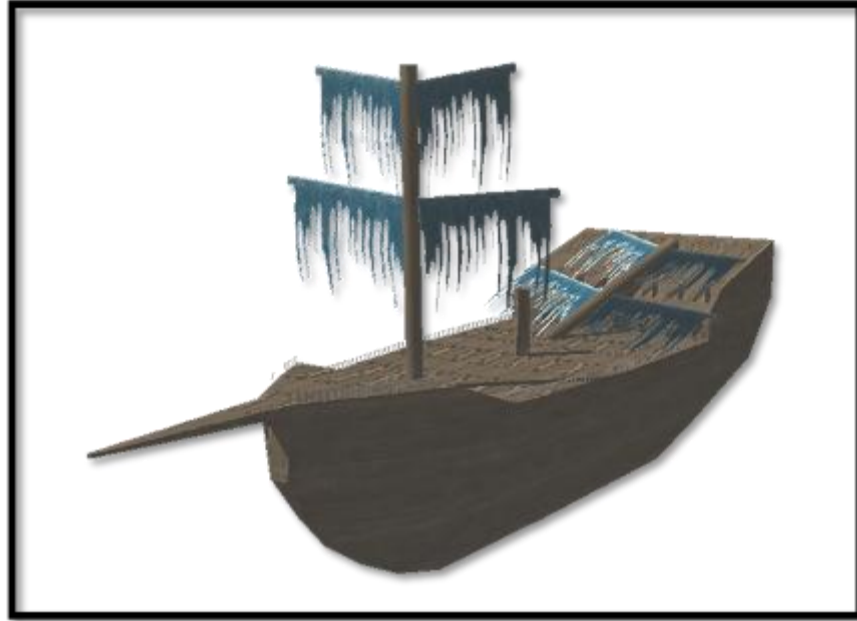


Figure 63: Pirate Ship

5.3.4. Fish

We created many different types of fish that move around the map during gameplay. The schools of fish come in a variety of colors: blue, green, and yellow. Figure 64 shows an example of the green variation of a school of fish. Only one ocean sunfish, in Figure 65, is present on the map because, in actuality, this fish is very large and we did not want to create distraction for the player. We also included two common reef fish: the teardrop butterfly fish, seen in Figure 66, and the blue tang, seen in Figure 67.



Figure 64: Green Fish School



Figure 65: Ocean Sunfish



Figure 66: Teardrop Butterfly Fish



Figure 67: Blue Tang

5.3.5. Coral Reefs

We created eight variations of coral reefs. Each of the coral reefs has a different rock formation topped with aquatic plant life common in reefs. We included giant sea fans, red sea fans, brain coral, honeycomb coral, and many variations and sizes of sea sponges, seen in Figure 68.



Figure 68: Variation of a Coral Reef

5.3.6. Kelp

The kelp was mostly positioned around the turtle headquarters and near the map boundaries on the Piscivian side of the map. We created three variations of the kelp and formed them into clusters to create an illusion of a forest of kelp, shown in Figure 69.



Figure 69: Kelp

5.3.7. Ruins

We created three variations of ruins that were reminiscent of Doric columns. The ruins were scattered around the Piscivian side of the map, as shown in Figure 70.



Figure 70: Variation of Ruins

Chapter 6: User Interface

The user interface consists of components that display information about the state of the game world, such as the positions and stats of units and structures. The user interface also allows players to interact with these game entities. For example, using the mouse cursor, the player can select and issue orders to units, construct buildings, and spawn a commander. Figure 71 shows one aspect of the user interface.

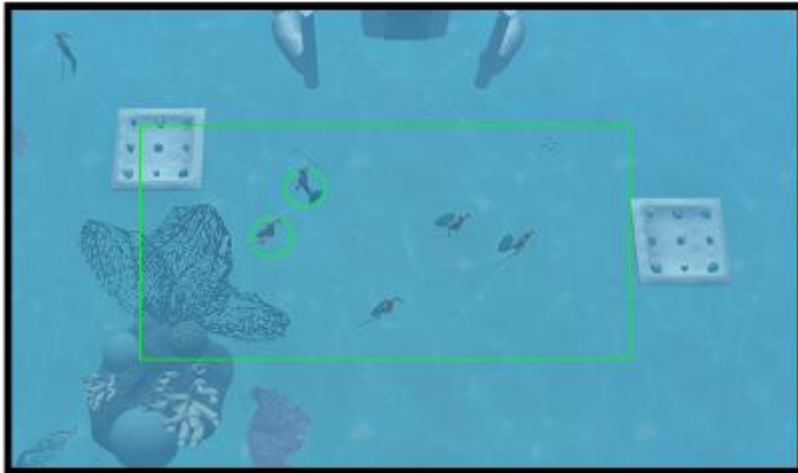


Figure 71: Selection of Multiple Units

This image demonstrates the selection of multiple units with a selection box. This box is created by holding down the left mouse button and dragging the edge of the box to the desired dimensions. The two mermen with circles underneath them are already selected.

This chapter will discuss the interface for the menu, the tactical-commander role, and the field-commander role. Some of the interface designs were influenced by the research done on in-game help, as can be seen in Chapter 8: In-Game Help – Results and Implementation.

6.1. Menu System

The menu system was implemented by modifying the existing C4 menu system. Visually we changed the colors of the menus to match our color scheme, as shown in Figure 72, and we replaced the C4 logo with the *Blood Tide* logo. In addition, we

removed all of the options that did not apply to our game, such as the ability to customize the player's in-game avatar.



Figure 72: Menu Item

The menu section will discuss the artistic and technical aspects of the menu system, specifically the main menu and game lobby.

6.1.1. Main Menu

The main menu for our game originally consisted of the game title, our team name, and the option to begin a multiplayer battle using either the Human faction or Piscivian faction. The color scheme was initially chosen to represent Human and Piscivian faction, using red and blue respectively. The menu conveys to the player the serious feeling of *Blood Tide* by representing death with the appearance of blood in water.

The main menu design completely changed from the original design, as shown in Figure 73. We decided, instead of a static image, to create a dynamic world that could move like a virtual aquarium. This interface was designed to be more interesting for the viewer, as the fish move around and the floor appears to reflect the sun, as shown in Figure 74.



Figure 73: Design for Main Menu



Figure 74: Implemented Main Menu

While many things in the implementation changed, the core goals of the main menu stayed the same. The underwater feel was not only kept, but strongly enhanced by the new design. Also, the new design still focuses on our game's title and has a simple interface, although the new design is slightly more complicated than the original.

6.1.2. Game Lobby

The game lobby consists of three sections: an area for players on the Human faction, which is shown in the area labeled "Humans" on the left; an area for the Piscivian faction, which is shown in the area labeled "Piscivians" on the right; and a third area for players who have not yet joined a team, which is labeled as "Undecided," as shown in Figure 75.



Figure 75: Game Lobby

When a player first connects to a game, the player's name appears in the undecided area. The player can then click on any slot with the name "Open" to move to that slot. Once the player has moved to a different slot, that player's name will be removed from the old slot and added to the new one. The host of the game has extra options available to help with game creation. He can choose to close slots, preventing users from occupying them, and also kick players from the game, as shown in the undecided area of Figure 75. Once players are in both the Human and Piscivian team sections, the host can click the start button to launch the game.

6.2. RTS UI

The RTS UI consists mostly of elements that provide information to the player about his units, buildings, commander, and in-game events. In addition to displaying

information, it also allows the user to communicate with other players and interact with the game world. To best give the player the feeling they command the entire army, we built the RTS UI for use in the tactical-commander role.

To implement these elements, we used a polling technique, meaning that each element polls the game data that it corresponds to. We chose to use polling instead of using an event-driven approach because it separated the UI elements from the game elements. The UI components were relatively easy to build in isolation to other game systems. This interface is pictured in as pictured in Figure 76.



Figure 76: RTS UI

The RTS UI consists of the:

- **Help window:** Used in the tutorial to provide instructions to the player. Also provides tips for the player and messages for game events, such as when a base is being attacked by enemies.
- **Chat window:** Allows communication between players.
- **Field-Commander selection:** Shows the list of the commanders that the player can play as.

- **Unit stats window:** Provides detailed information about the current selected game entity. If the current selected entity is a unit, its health, stats and abilities will be displayed in this box. If the current selected entity is a headquarters, it will display the headquarters' name and current health.
- **Mini-map:** Displays the positions of units, headquarters, buildings and commanders in the game world. The player can also move the camera and issue orders to units through the mini-map, as if they were clicking on the actual point in the world.

6.2.1. Building UI

The building UI component consists of a pop-up UI element that displays the building choices for a plot of land, as shown in Figure 77. When a player selects a plot of land (right-clicks on it), an interface pops up and displays all of the buildings that can be constructed on it. The user can then click on one of the buildings to begin construction of that building.



Figure 77: Building UI

6.2.2. RTS Camera

The RTS camera provides the player with a birds-eye view of the game world. It can be moved around the game world several ways: by moving the cursor to the edges of the screen, by using the arrows on the keyboard, and by clicking on the mini-map.

Because the two game modes use different types of cameras and the location of the commander and the RTS camera are variable, a camera blending feature was introduced to smooth the transition between these two modes. The blending works in three steps, as shown in Figure 78.

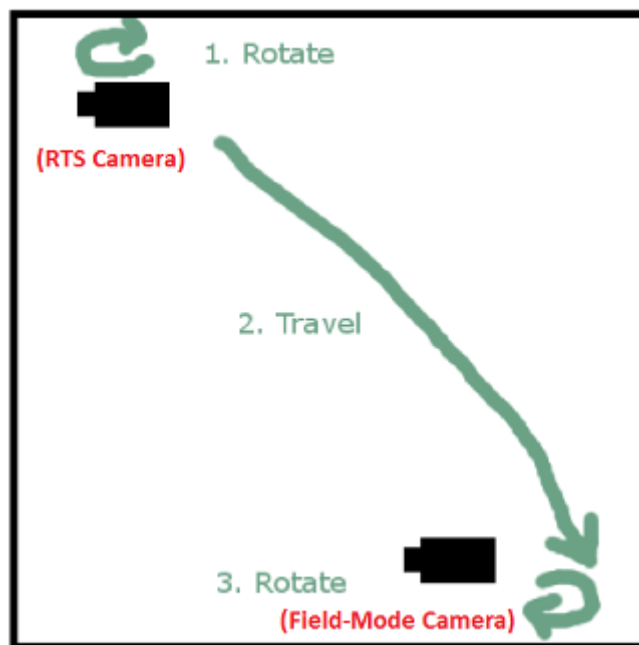


Figure 78: Camera Blending

As can be seen in the image, the camera must first spin around on the z-axis until it faces its new location. Then, the camera travels along a straight path to the new location. Finally, the camera spins again until it orients itself correctly in respect to the field-commander.

6.2.3. RTS Controls

The main actions we implemented as part of the RTS controls are the ability to select entities, spawn a commander, issue orders to units, and construct buildings.

The control scheme for the RTS role is:

- Left clicking on an entity selects it.
- Dragging a box around units with left click selects all of them.
- Right clicking on an enemy unit with units selected attacks the enemy.
- Right clicking somewhere in the world with units selected moves them to that location.
- Pressing A then clicking on a position in the world issues selected units to move to a location and to attack any enemies along the way.
- Pressing S stops the units.
- Pressing T changes to the field-commander role.
- Holding right mouse button on a plot and releasing on a building starts construction of that building.
- Pressing V selects all units.
- Pressing control and 1, 2, or 3, creates a group of units.
- Pressing 1, 2, or 3 selects a previously saved group of units.

6.3. Field-Commander UI

The action UI component provides the player with information pertaining to the field-commander, such as the field-commander's health, stats, and rank, as pictured in Figure 79. It also carries over other elements from the RTS UI: the mini-map, chat window, and help window.



Figure 79: Field-Commander UI

The field-commander UI consists of the:

- **Help window:** Used in the tutorial to provide instructions to the player. Also provides tips for the player and messages for game events, such as when a base is being attacked by enemies.
- **Chat window:** Allows communication between players.
- **Field-Commander stats, including:**
 - **Commander rank pane:** Contains one star for each of the three ranks. Unlocked ranks are yellow.
 - **Commander health pane:** Displays the current and maximum health of the commander.
 - **Commander abilities pane:** Displays each of the three abilities for a commander. Locked abilities are grayed out completely. Abilities that are on cooldown are partially grayed out, indicating the percentage of the cooldown that has expired.

- **Mini-map:** Displays the positions of units, bases, building and commanders in the game world. The player can also move the camera and issue orders to units through the mini-map, as if they were clicking on the actual point in the world.

6.3.1. Action Controls

The controls in field-mode provide functionality for the player to move the commander, aim, and perform abilities.

The control scheme for the Action role is:

- Moving the mouse looks around.
- Pressing W moves forward.
- Pressing S moves backward.
- Pressing A strafes left.
- Pressing D strafes right.
- Pressing 1, 2, or 3 performs abilities.
- Pressing T goes into the tactical commander role.
- Pressing G makes all friendly units follow the commander.

Chapter 7: In-Game Help – Overview and Methodology

RTS games, like *Blood Tide*, predominantly use media such as printed strategy guides and internet walkthroughs to provide help to players. The problem with these forms is that they break the player's immersion in the game world; the player must change his or her focus from the game to reading a guide or searching the internet to find help.

A game's success can be credited to how well it teaches the player game skills. If the player cannot figure out how to continue with the game, he or she will quit playing. Game help is not a primary focus during development because programming, audio capturing, and art asset creation takes precedence for the release date. Each game can have millions of dollars invested into the development, so a game's failure to sell can be catastrophic to a game company's future. Many high-budget games focus on incorporating in-game help to keep the player immersed in the game world.

In order to determine an effective way to apply in-game help to *Blood Tide*, more analysis and information of effective and ineffective uses of in-game help was needed. To achieve this goal we completed the following objectives:

- 1) We identified the various forms of game help and found commonalities between game help genres.
- 2) We determined what elements of a game are required for successful game help by categorizing the types of help and what game elements are needed for the help to work effectively.
- 3) We created an effective method of providing in-game help for *Blood Tide* by mapping when help is needed during the game. This was determined through testing.

Section 7.1 will be a literature review where we discuss immersion as it relates to providing help to the player. Section 7.2 will discuss the method we used to evaluate the different forms of in-game help. Chapter 8: In-Game Help – Results and Implementation will discuss our results from the genre analysis, explore the results from our research of the forms of in-game help that are most effective, determine which forms of help to employ, and discuss how the in-game help was implemented into *Blood Tide*.

7.1. Literature Review of Game Help

The concept of immersion was first introduced by Johan Huizinga, a widely cited cultural theorist on “Play Theory.” He discusses play as a fundamental part of human life. While his book *Homo Ludens: A Study of the Play-Element in Culture* (1955), does not discuss play in computer games, the nature of play in general can be applied to computer game theory. In his book *Homo Ludens*, he first coins the term “magic circle” to describe the realm of the game world in which play occurs. Huizinga deemed this realm “magic” because of the thin illusion that it creates to separate the play-world from reality. This circle is involved with how immersed the player is in a game; it makes the player feel involved with this newly created world, characters, and story. Once the player leaves the magic circle, when looking for external help for example, he or she is no longer immersed in the game world and tends to have a difficult time re-entering. Printed strategy guides and user-made guides are examples of forms of help that do break the magic circle. However, these forms provide game help in different ways than in-game help, and therefore are important to the understanding of how help is provided to the player. These written guide forms are discussed more in depth in Appendix D: Written Guide Forms.

Printed strategy guides contain a large number of original concept art and more in-depth character development than is shown in the game. Print guides are often released earlier than the game to offer players a sneak peak at the game and increase excitement for the game title. Because printed guides are official and endorsed by the game development company, they generally provide the player with hints and tips rather than direct commands.

Unlike printed guides, user-made guides often are step-by-step walkthroughs that thoroughly detail exactly what the player needs to do during the game. User-made guides can also be more difficult to use because they typically do not use images or text formatting; however, this form of help is free and readily available when a player gets frustrated. Forums, user-hosted or a part of the game’s official website, are a discussion about the game and the various tactics that players have used. Forums allow a player to ask questions about a specific problem they are having; however, unlike user-made or printed guides, the help is not instantaneous.

The advantage of in-game help is that players do not have to leave the magic circle; offering real time feedback on the player's actions. The longer the player is immersed in the world of the game, the longer the player will play the game. Role-Playing Games (RPG) have a long history of interweaving in-game help into the narrative because of the genre's unique game mechanics and complex gameplay. For example, in the game *Tales of the Abyss*, an RPG made by Namco, the player is forced into a sparring lesson with his mentor during the beginning of the game, or in the game *Kingdom Hearts*, an action RPG made by Square Enix, the player learns game mechanics through a search for pieces to build a raft. While some game genres have utilized in-game help, other genres have more difficulty when seamlessly blending hints into games because of their lack of a strong narrative and detailed character development. Because RPG's tend to use very unique game mechanics, they rely on in-game help to teach the player, which has forced this genre to explore the different ways of using help during the narrative. For this reason, we chose to study in-game help in RPG's, to make *Blood Tide* more enjoyable for the player. When the help is implemented effectively, the player easily learns how to play the game, making it fun and therefore attributing to the game's success.

In order to determine the effectiveness of in-game help we first need to understand the important factors to create effective in-game help and identify the forms of game help, as discussed in Chapter 8: In-Game Help – Results and Implementation. The advantage of in-game help is that players receive help while staying immersed within the game world. In-game help offers real time feedback on the player's actions.

According to Peter Bickford, a major goal for a game developer is to keep the player immersed in the game for as long as possible (qtd. in Federoff, 8). The amount of time spent in the game world will increase the player's attraction to the game's series, making sequels more valuable. Sequels are cheaper to make because a lot of the code and art assets can be re-used, saving on resources and labor needed to produce the game.

The presence as immersion is a concept that has been studied by psychologists analyzing virtual reality. The most convincing form of virtual reality is when the senses are immersed in the virtual world (Biocca & Delaney, 57 - 124). This perpetual immersion occurs when the player's senses are in tune with the virtual world of

computer games. The more senses that are submerged into a virtual world, the more of an immersive experience is given to the player (Biocca & Delaney, 57 - 124). Computer games have relied on submerging the senses of sight, sound, and most recently touch through rumble controller technology. Computer games lack the compatibility to incorporate the senses of taste and smell, which has been used in other media like motion simulators and IMAX theaters.

An "illusion of nonmediation" is the state in which a person fails to perceive or acknowledge the existence of a medium within their reality (Lombard & Ditton). In this state the person responds as if the medium did not exist (Lombard & Ditton). While Lombard and Ditton use "nonmediation" to refer to media that are experienced naturally, the concept relates to immersion and can be applied to computer games. An "illusion of nonmediation" occurs when a person strongly identifies with this "new" reality (Lombard & Ditton). When the player is immersed into the realm of the game world, the game becomes a new reality for as long as the player stays within the magic circle.

In-game help utilizes the concept of immersion because it keeps the player in the magic circle as they learn. Instead of forcing the player to stop play to solve their problems, in-game help is designed to be there when the player is having problems achieving his or her goal. However, other forms of help are still utilized by players and are the basis from which in-game help was built from and therefore are still relevant.

7.2. Genre Analysis of In-Game Help

The goal of this study was to analyze methods of in-game help used in RPG's and to assess the commonalities of effective in-game help. To achieve our goal, we completed the following objectives:

1. We prepared a coding sheet of which to test five games.
2. We tested five different games and recorded the presence of in-game help.
3. We analyzed our results using generic criticism to determine the effectiveness of in-game help.

This section will discuss the method we used to achieve our goal.

7.2.1. Objective 1: Prepared a Coding Sheet

To test the prominence and effectiveness of in-game help, we tested the first thirty-minutes of gameplay for five different console games in the RPG genre. We chose to study the first thirty-minutes of gameplay because the introductions of games contain the most concentrated in-game help.

We analyzed the in-game help for five games that were well-received by the public. We only analyzed well-received games for two reasons: (1) Well-received games have a higher probability of having effective in-game help than a poorly-received game, and (2) well-received games and poorly-received game may have differences in the quality of in-game help. The games we chose were *Final Fantasy X*, *Star Ocean: Till the End of Time*, *Rogue Galaxy*, *Paper Mario: The Thousand Year-Old Door*, and *Batman Kaitos: Eternal Wings and the Lost Ocean*. These games ranged from 81 percent to 91 percent according to *GameFAQs* Rankings and ranged from 8.7 to 9.5 out of ten on *IGN* score in December 2009.

To play these games, we chose people who have played a console RPG, action, or adventure game in the last six months because these people should have an understanding of this genre's basic game mechanics. We did not want the players to be concerned with essential game skills such as movement and accessing the menus. Additionally, we wanted to eliminate the variable of the player's intuition during testing.

We determined the presence and use of various forms of in-game help using a coding sheet we developed (Appendix E: Coding Sheet). The coding sheet consisted of nine different categories that correspond to popular methods of providing help to the player: general help, tutorial level, mentor character, other character help, in-ear guide, dialog tree help, pop-up windows, synopsis, and user interface help. We chose these genres of help based on the types of help we have personally encountered, as well as those we have discovered through our research. To find common trends of in-game help, we looked for specific forms of help in each of the categories:

- **General Help:** We studied what types of help the game used and if the help was skip-able. We also looked at whether the game skills were taught all at once or gradually. By looking at the types of help and how each form was used, we will be able to draw conclusions about common trends of in-game help.

- **Tutorial Level:** We analyzed if the game help is presented during a separate level in the game. A tutorial level is defined as a separate level training in-game or a defined “Prologue” or “Introduction” to the game. The tutorial level is sometimes broken in to several smaller levels, which is another form of game help.
- **Mentor Characters:** We examined whether a wiser or more experienced in-game character advice to the player or taught the player the controls of the game.
- **Pop-Up Windows:** We considered if windows appear on screen to give tips or hints to the player. These pop-up windows can appear as full screen window tutorials with instructions about how to do a specific action. We also analyzed whether these windows hindered the player’s ability to continue with the story or could appear as a small box on the side of the screen.
- **User Interface Help:** We wanted to analyze the presence of a map or mini-map to show the player where to go, as well as the symbols used to show significant places on the map. Also, we studied whether an in-game handbook reminded the player of skills learned.
- **Other Character Help:** We analyzed whether other characters besides the mentor or non-playable characters (NPC’s) gave the player tips about how to play the game. We also analyzed if the player’s goals or objectives were learned through cutscenes or in the game world with other characters.
- **Dialog Tree Help:** We studied if the game provided an interactive window in which the player can initiate tutorial information about different topics according to want. We also noted whether the help was given through conversation or window.
- **In-Ear Guides:** We studied if a character guides the player, not as a physical character but as someone who could see the entire game field.
- **Synopsis:** We also studied whether a synopsis gave the player a summary of what has just happened.

7.2.2. Objective 2: Tested Games and Recorded Presence of Game Help

During each test, we designated one person to be the player, one person to be the observer, and one person to be the timer. We tested each game twice to ensure reliable analysis of the effectiveness of the game help.

- **The Player** played the chosen game from the beginning for 30 minutes of play time or until the player received a game over if insufficient time remained in the 30 minutes of play to progress further in the narrative.
- **The Observer** watched the player and analyzed the game's help using the coding sheet. At the end of the play-through, the observer asked the player to rate their confidence that they have learned all of the necessary skills in order to finish the game using a scale of one to five: one meaning the player has learned nothing and five meaning the player is very confident they could continue playing with ease. Also, a section of the coding sheet allowed for any notes the observer wanted to add about the play-through that could be reflected with the coding sheet.
- **The Timer** was responsible for recording the amount of time the player spent interacting with the game world, as opposed to watching cutscenes¹.

Each play through of each of the tested games was broken down as follows:

1. The observer recorded who the player, observer, and timer were for the play-through.
2. The observer recorded the game title, game genre, publisher, and year sections.
3. The player started game.
4. The timer began the timer once the player takes control of the game to count the gameplay time.

NOTE: The timer must pause the gameplay timer when the player was in a cutscene. This did not count when a player initiates a conversation with another character, only when control is completely removed from the player.

¹ A cutscene, often called a cinematic or in-game movie, is when all control is removed from the player and a short video clip is shown. Often, cutscenes use voice acting instead of dialog boxes. We do not define a cutscene as when a player initiates a conversation with another game character or object.

5. The observer watched while the player played the game, filling out the different categories of the coding sheet accordingly.
6. At end of tutorial, the timer stopped the timer and recorded total time that the player was in control of the game world.
7. The observer asked the player to rate how confident they felt that they had learned the game's skills.

7.2.3. Objective 3: Analyzed Our Results

We conducted our analysis using generic criticism to determine the effectiveness of the different types of in-game help. We compiled the data we received through our tests to find the common trends used in all of the tested games. We analyzed what forms of help were most common and which were most effective. Then, we wrote an analysis about what we found through this method for future games to create more effective in-game help. We compared which games had each form of help, in addition to how effective the player felt the in-game help was at teaching them the game skills. We concluded with our reflection about our process and what we believe should be done in the future when analyzing in-game help.

To determine the effectiveness for each form of game help, we determined a formula that calculates a “weighted total” for each question on the coding sheet. We compared the weighted total for each type of help to the average of the weighted totals. If a type of help's weighted total is above this average, we considered it to be an effective form of help. If a type of help's weighted total is below the average, we considered it to be an ineffective form of help. If a type of help's weighted total was very close to the average, we were not able to conclude the effectiveness unless our observations note otherwise. The formula for calculating the weighted total is:

Equation 1: Weighted Total

$$\text{Weighted Total} = \left[\sum_{g=1}^5 (\text{Response}_g \times \text{Average Skill Learned}_g) \right] / (\text{Sum of "Yes"})$$

Where: g = index of the game

Response = list of integers, either 0 or 1

Average Skill Learned = list of numbers from 1 through 5

Sum of "Yes" = integer from 1 through 5

For a case example of how we applied our weighted total formula to determine the effectiveness of help refer to Appendix K: Example of Weighted Total.

Chapter 8: In-Game Help – Results and Implementation

In Chapter 7: In-Game Help – Overview and Methodology, we discussed our goal for incorporating an effective form of in-game help into *Blood Tide*. Additionally, we discussed our method for testing the effectiveness of in-game help for RPG's.

This chapter will act as a continuation of Chapter 7 but will focus on the categories of help and the results of testing. In addition, sections 8.2 through 8.5 will address the four forms of in-game help that we thought would be most effective for *Blood Tide*. We decided that tutorial levels, mentor characters, pop-up windows, and user interface help would be the best options to teach the player how to play our game.

In section 9.3, we will discuss the method that we used for determining the effectiveness of our in-game help in *Blood Tide* to establish whether the implemented in-game help was clear and successful. The testing chapter will also discuss the final results from testing our in-game help on players.

8.1. Overall Results of Methodology

Based on our research using generic criticism, as discussed in Chapter 7: In-Game Help – Overview and Methodology, we have concluded that a common trend is to include in-game help. Additionally, we have determined that allowing the player to skip the in-game help hinders the ability for the player to learn. Breaking immersion to teach the player the game skills was useful because the player was forced to learn how to progress in the game. In this section we will discuss the categories of in-help game, our findings about the common trends and which categories were most useful for teaching the player game skills.

Through our method, we determined how effectively each game was able to teach the player important game skills. The player's confidence that he or she would be able to continue playing the game was rated on a scale of one to five: one would mean the player has learned no game skills and five would mean the player had fully learned all of the game skills. Both of the player's confidences for the two play-throughs were averaged to create the average skills learned. The games and their help effectiveness are rated below:

Table 15: Average Skills Learned for tested games

Game Title	Average Skills Learned
Baten Kaitos: Eternal Wings and the Lost Ocean	1.75
Final Fantasy X	4.5
Paper Mario: The Thousand-Year Old Door	4.5
Rogue Galaxy	5
Star Ocean: Till the End of Time	3

Rogue Galaxy (Appendix I), *Final Fantasy X* (Appendix G), and *Paper Mario: The Thousand-Year Old Door* (Appendix H) were considered easy to learn while *Star Ocean: Till the End of Time* (Appendix J) and *Baten Kaitos: Eternal Wings and the Lost Ocean* (Appendix F) were more confusing to the player.

The average of the skills learned for the five tested games is 3.75. A type of help that had a weighted total above this number was considered to be an effective way to teach players, while a type of help with a weighted total below this number was considered to be ineffective at teaching game skills. The complete list of the results can be found in Appendix L: Results Summary.

Each game we tested used multiple forms of help, ranging from three types to six. The games that were rated the highest for effectiveness of in-game help used more types of help than the games that were rated lower. We can conclude that using multiple forms of help gave the player the most opportunity to learn and made the game more effective at teaching game skills. We were able to test all of the different forms of help we defined earlier, except for in-ear guides and synopsis. All of the other types appeared at least once in testing; therefore, we were able to draw a conclusion about them. Because we did not test any games that utilized in-ear guides and synopsis as a form of help, we cannot make any conclusions about these genres of game help.

The common trend was to force the player into the tutorial, which was beneficial for providing help to the player. Enabling an option to skip the tutorial prevented the player from learning all the necessary skills. We are unable to make a definite conclusion about the effectiveness of providing game help upfront versus gradually

through the game because we found no definitive group of games that included one or the other. However, we believe that gradually providing help to the player is more effective than presenting all help upfront, because the player is not overwhelmed with information and is able to master skills before learning new ones.

The general conclusions we made about help closely reflect our implications to *Blood Tide*. Since we found that in-game help is a common trend and beneficial to learning game mechanics when used in multiple forms, we applied several of the effective forms of help to our game: tutorial levels, mentor character, pop-up windows, and user interface.

8.2. Tutorial Level

Ernest Adams describes a tutorial level as “early levels that teach the player how to play” (375). In-game tutorials have evolved from the game instructional booklets that are included in the game’s packaging. Game instructional booklets provided the player with written instructions of game controls and techniques. Opposed to the booklet, the purpose of the tutorial is to help the player learn relevant game skills in a more interactive setting. Tutorials allow the player to learn and interact within the magic circle, and therefore provide the player with a more immersive experience than the written booklets.

Tutorial levels are usually presented at the beginning of a game, but sometimes will be presented at specific points during the gameplay when the player learns a new skill. The tutorials have a common element of educating the player on specific skills and allowing the player to interact with the game to master the skills. Tutorial levels, unlike the other forms of in-game help, are separate from normal gameplay. Sometimes these levels are cleverly hidden in the narrative, while other games are obvious in their deviation from the game. One example of a tutorial is shown in Figure 80.



Figure 80: Scribblenauts - Tutorial Level (Slisenberger)

As seen in the image, an example of a defined tutorial is in the game *Scribblenauts*, a puzzle game developed by 5th Cell. A tutorial level is found in the “University” where the player receives simple objectives to learn game mechanics. The tutorial is called “University” and level number is 0-1. Also, the player is being directly told how to move in the game world in the image above. Conversely, in the game *Need for Speed: Underground*, a racing game created by EA Black Box, the tutorial level in which the player is taught how to drive is disguised as a street race.

8.2.1. Results for Tutorial Level

We concluded that a common trend in the games we studied was to include a tutorial level, but not reveal itself as a tutorial level. Tutorial levels were effective in providing help to the player because they gave the player a small and easy goal to complete at the beginning of the game. Four out of the five games we tested made use of defined level for learning. While the game mechanics were more confusing in *Baten Kaitos: Eternal Wings and the Lost Ocean*, the player was given a linear path to progress through the game. In this game, the player had to investigate the woods near the local village because villagers claimed it to be haunted. *Star Ocean: Till the End of*

Time did not provide the player with any goal or direction. Because of the lack of definition, all players wandered around the large hotel aimlessly searching for guidance for about 15 minutes.

Rogue Galaxy, *Final Fantasy X*, and *Baten Kaitos: Eternal Wings and Lost Ocean* had beginning levels that were not defined as tutorial levels but acted as one. *Paper Mario: The Thousand Year-Old Door* had a defined tutorial level and acknowledged itself as one, as can be seen in Figure 81. Only *Star Ocean: Till the End of Time* did not have a tutorial level.

While it was hard to quantify because the levels were not defined, we believe that *Final Fantasy X* and *Baten Kaitos: Eternal Wings and Lost Ocean* had sub levels to their tutorial because there multiple areas in which the player learns basic game mechanics. We were not able to fully determine the effectiveness of sub-levels in the tutorial because the player often did not reached them in our 30 minute tests. However, including multiple sub-levels did not seem to hinder the learning process.



Figure 81: Paper Mario - Prologue Introduction Screen (Chuggaacronoy)

Out of the games that we played only *Paper Mario: The Thousand Year-Old Door* had acknowledged that the tutorial level was specifically for teaching, labeled “Prologue;” therefore, the lack of data makes it difficult to draw conclusions. However, we believe that acknowledgment of a tutorial level does not hinder or help the effectiveness of the tutorial. The full results for the tutorial level can be seen in Table 16.

Table 16: Tutorial Level Results

Tutorial Level	Total Games with Form of Help	Weighted Total
Is there a defined level for learning?	4	3.94
Are there multiple sub-levels?	2	3.13
Does it acknowledge that it is help?	1	4.50

8.2.2. Implementation of Tutorial Level

In order to create an effective tutorial, we first outlined the various pieces of *Blood Tide* that the player would need to learn to successfully play the game. After this outline was created, we began to link these pieces together to create a gradual increase in difficulty and complexity over the course of the tutorial. Finally, we created a narrative to weave all of the tasks that would be taught to the player into one script.

To create a tutorial that allowed the player to experience both factions equally, we decided to write two dialogs for the tutorial level. The order in which events were presented in the tutorial was kept the same for both tutorials; however, the separate scripts were written to reflect each of the faction’s mentor characters. These mentor characters, as discussed in section 8.3, are the player’s guide through the tutorial. We chose to write two scripts to allow the player to gain familiarity with the faction of their choice, or even both factions if the player chose to play the tutorial twice, before they began online play. The entire script can be found in Appendix M: Tutorial Script.

8.3. Mentor Character

Mentor characters are often used in games that have a strong narrative, such as action, adventure, and RPG games. Ernest Adams discusses the mentor character’s role in terms of the common archetype in storytelling, to teach the player initial skills and give advice through the beginning of the game story (149). They often drive the player’s objectives in the beginning of the game, fill in the main character’s backstory, and give the player limited feedback on his or her actions. The player is frequently reliant on the

mentor character in the first several missions in a game (Isbister, 238). Two British professors who study AI in games believe the benefit of mentor characters is that they provide “guidance targeted to individual users” and “have the potential to make introductory experiences much more pleasant than they might otherwise be” (Charles, 2). One of the drawbacks for using mentor character is that the development team must anticipate the situations in which the player may have trouble with the game because the mentor character must be written into the story during the beginning of the narrative. An example of a mentor character is shown in Figure 82.



Figure 82: Warcraft 3 - Uther advising Arthas

For example, the game *Jak & Daxter*, a platformer released by Naughty Dog in 2001, introduced Samos the Sage, the master of Green Eco, who helped the player learn the basic controls of the game. The game *Warcraft III*, an RTS released by Blizzard in 2002, uses the main character’s uncle, Uther, to give the player advice, directions as to building their first headquarter, and describing his first missions. However, after the initial mission explanation, the mentor character runs around the headquarters, defending it from enemy troops for the rest of the mission and does not help the player complete his or her objective.

8.3.1. Results for Mentor Character

The common trend was to not use a mentor character to provide help. Even though the presence of a mentor character was not as common, we have concluded that using a mentor character to provide the player help was effective in teaching the game skills, as shown in Figure 83.

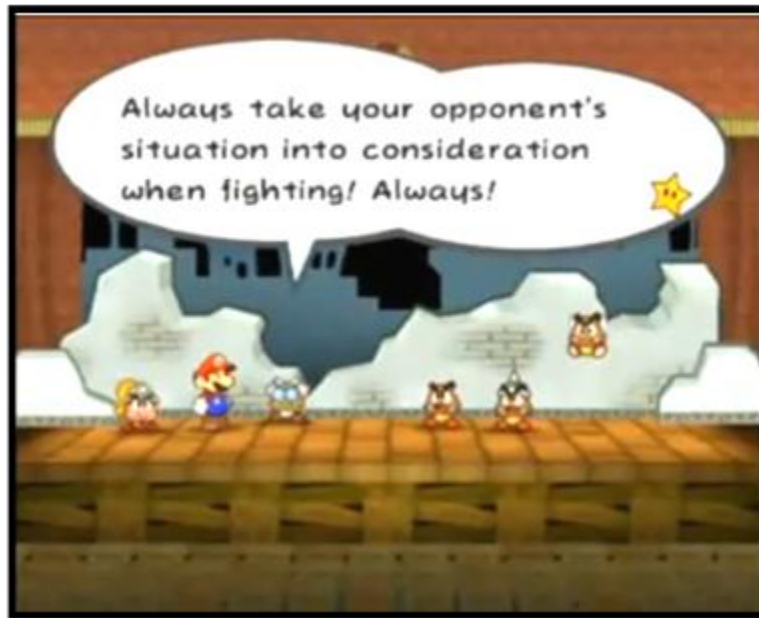


Figure 83: Paper Mario: Mentor - Professor Frankly (Helmerts10)

Paper Mario: The Thousand Year-Old Door and *Final Fantasy X* were both highly rated games in providing help to the player and utilized in-game help via mentor characters. In *Paper Mario: The Thousand Year-Old Door*, Professor Frankly acted as the mentor character by accompanying the player through the prologue. When in battle Professor Frankly gave the player hints to defeat the enemies, as can be seen in the image above. Additionally, Professor Frankly explained the basic battle mechanics to the player. In *Final Fantasy X*, Auron, a more experienced fighter, acted as the mentor character by giving the player tips on how to win the first few battles in the game. The full results of the mentor character are shown in Table 17.

Table 17: Mentor Character Results

Mentor Characters	Total Games with Form of Help	Weighted Total
Is there a mentor character?	2	4.50

Because this form of help was so well received, we decided to include this form of help in *Blood Tide*. The mentor character was implemented alongside the tutorial level to provide the player guidance. We chose to create a mentor character because it made sense for a more experienced figure to help the new battle commander, or player. However, as the factions are vastly different, we created two different mentors.

8.3.2. Implementation of Mentor Characters

As the factions are vastly different, we created two different mentors, one to guide each faction – a battle experienced scholar for the Piscivans and an artificial intelligence for the Humans. Each of these mentor characters was designed to give the player help during the tutorial.

The Piscivian mentor, General Crabbly (Figure 84), is a retired commander who was once the tactical and field-commander of the crab race. Since the crabs can no longer fight, General Crabbly is asked to teach the player how to be an effective battle commander.



Figure 84: Implemented Mentor for Piscivian Faction - General Crabbly

The Human mentor, TACTIS (Figure 85), an acronym for Tactical Advisory Computer Training Interactive Simulator, has data on every Earth war and can make flawless battle calculations. However, because humans have never fought undersea life before, TACTIS is unable to predict how the Piscivian faction will act.

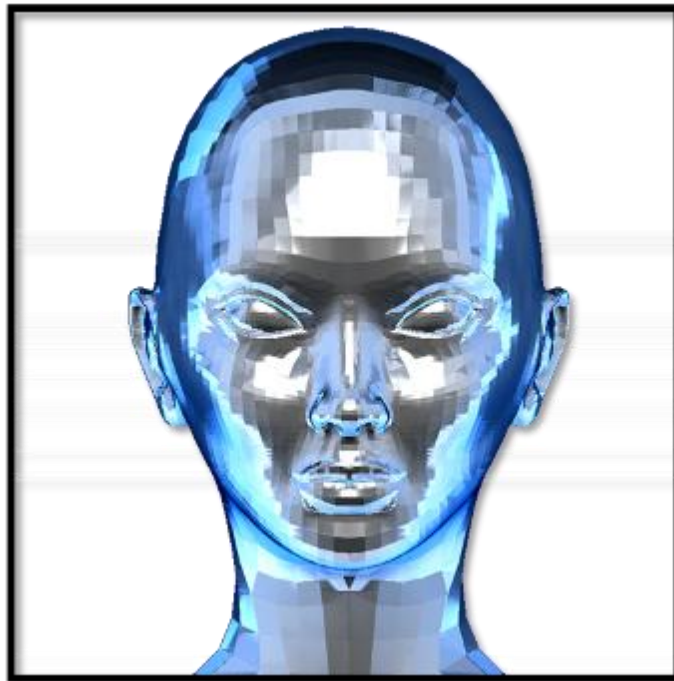


Figure 85: Implemented Mentor Character for Human Faction - TACTIS

8.4. Pop-Up Windows

Help in windows has evolved from early instances in arcade games, when all the controls were listed in text before starting the game, to a pop-up window of text used in current game help. Pop-up windows will display quick messages or images on screen that will alert the player about a current in-game problem. Pop-up windows are a term used throughout Ernest Adam's book, *Fundamentals of Game Design*, to describe different forms of windows that are displayed to the player. Pop-up windows are used in a wide variety of game genres and recently have been utilized for the game consoles when the controller is low on batteries or is not synchronized.

Unlike other forms of in-game help, pop-up help is known for not majorly interrupting gameplay because it requires minimal instruction or dialogue. Therefore, pop-up windows will keep the player in the magic circle and will not break the immersive experience; however, pop-up windows can be seen as annoying when overused, thus breaking the magic circle.

Game developers will usually apply pop-up windows to situations in which the player would be stuck. Pop-up windows are very easy to implement from a game design perspective because they do not need to fit in any sort of narrative or interface element. An example of one form of pop-up window is shown in Figure 86.



Figure 86: Mario Kart Wii - Wrong Way Pop-Up (Hmyaa)

The Mario Kart series, driving games developed by Nintendo, use pop-up tips to alert the player when he or she is driving the wrong way and to notify the player when they have completed a lap. If the player is driving the wrong way, Lakitu will fly in front

of the avatar with a sign indicating that the player is driving the wrong way. This pop-up tip has used both symbolism (u-turn sign) and text (“Wrong Way”) to convey its message to the player.

8.4.1. Results for Pop-Up Windows

We have concluded that the use of pop-up windows works exceptionally well in providing the player help. The common trend for games was to include pop-up windows to help the player. One example of a pop-up tip encountered during our tests is shown in Figure 87.



Figure 87: Rogue Galaxy - Pop-Up Tip (“Rogue Galaxy”)

The games that were rated highly all used pop-up windows to teach the player. *Paper Mario: The Thousand Year-Old Door* presented pop-up windows in battle, that were not obtrusive to gameplay, to explain how to perform abilities. *Rogue Galaxy* used pop-up windows that blocked the screen but gave detailed instructions on how to play and pop-up tips, where the player is being told how to throw a held object. *Final Fantasy X* used large pop-up windows that interrupted gameplay to give the player important information during battle. The games that were rated lower did not make use

of pop-up window tutorials. We concluded that all forms of pop-up windows were helpful to the player. The full results for the pop-up window are shown in Table 18.

Table 18: Pop-Up Window Results

Pop-Up Window	Total Games with Form of Help	Weighted Total
Are there pop up tips?	2	4.75
Is there a window based tutorial?	2	4.75
Are they obtrusive to the gameplay?	2	4.75

8.4.2. Implementation of Pop-Up Windows

We decided to use pop-up windows in the design of our in-game help for *Blood Tide*, because pop-ups were commonly used and are very effective in providing help to the player. During the implementation of the pop-up help, we created four different forms of this help. We produced loading screens for the two factions, designed help to appear in a special help window, created a game controls window, and implemented a pop-up window to suggest to players to play the tutorial.

8.4.2.1. Loading Screen

The loading screens were divided in design for each of the two factions to fit with their overall artistic design. The Piscivian faction features marble and sand textures with a Greek font on a stone tablet (Figure 88). The Human faction's loading screen features metallic textures, military themes, as shown in both the letter and the file folder, and a font that resembles typewriters used in WWII to relay messages (Figure 89).

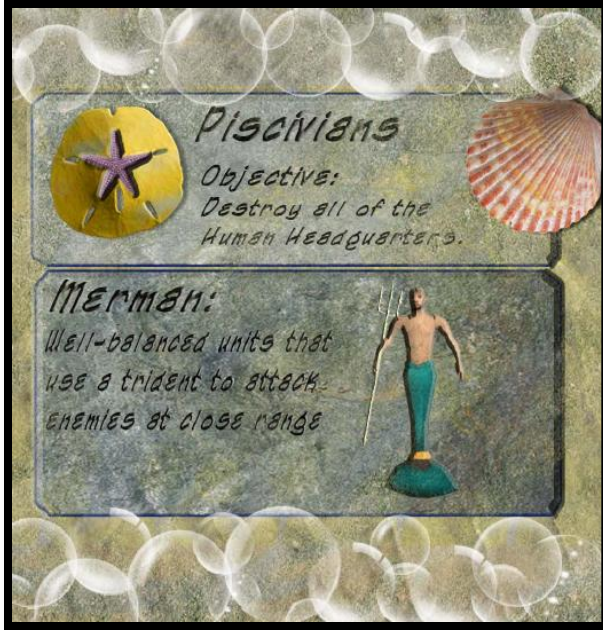


Figure 88: Implemented Loading Screen for Piscivian Faction

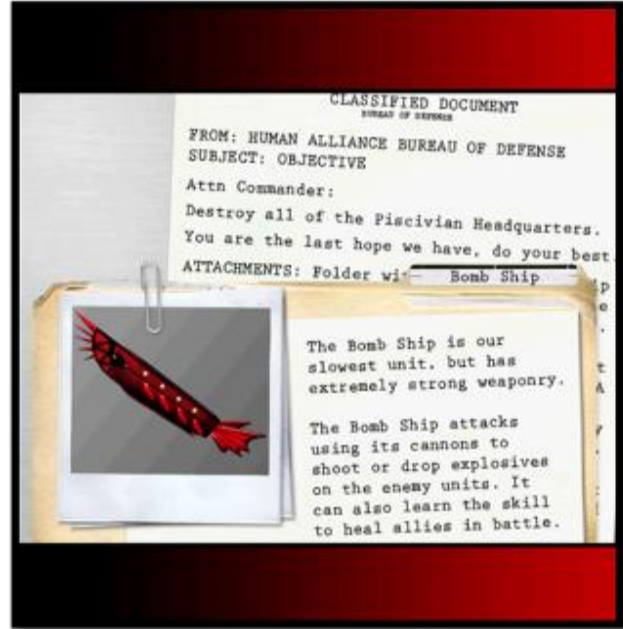


Figure 89: Implemented Loading Screen for the Human Faction

As you will note in both of the loading screens, the image is divided with the faction name and objective at the top, with a unit image and description on the bottom. This template was applied to create three different loading screens for each of the faction's unit types; the top section stayed consistent throughout the three screens, while the bottom was changed to match the appropriate unit. When the player loads the game, each faction will randomly choose to show one of the three unit loading screens. A full list of all of the Appendix N: Implemented Loading Screens.

8.4.2.2. Help Window

Originally, we intended to provide pop-up tips through the chat window on the bottom left of the screen; however, we needed to change this because we did not want the help to mix with potential dialogue between players. The chat window is an important element for communication in online-multiplayer games; therefore, we needed to find another location to present help to the players.

We implemented a help window, positioned in the top left of the screen, to give the player help. The help window will inform players of all the tips that are given to help the player. For instance, the help window will tell the player where to find the game

controls window, will remind the player when a building is finished constructing, and will remind the player how to play as the field-commander if the player has not utilized this role. Figure 90 shows the help window.

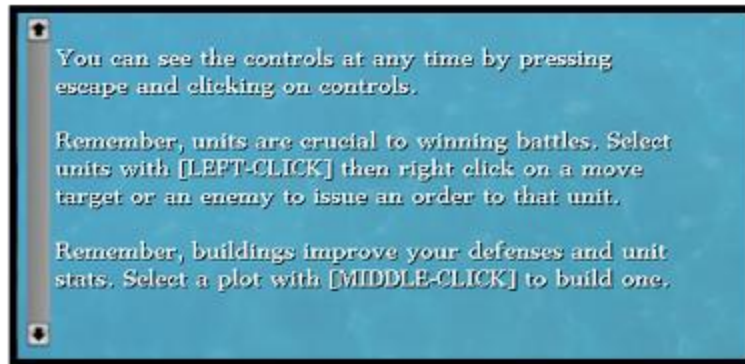


Figure 90: Implemented Help Window

The help window is somewhat small, allowing only three lines at one time. We chose white font color because of the contrast to the dark blue of the water.

8.4.2.3. Game Controls Window

We initially did not plan on having a game controls window, and we were instead going to present the game controls through the tutorial or help window. However, we were concerned that the help window could be flooded with help, and knowledge of the game controls would be lost. Therefore, we created a game controls window, which is located under the main menu, so the player could have a quick visual that informs them of the game controls. The help window will immediately inform the player on where to locate this help. The controls window is pictured in Figure 91.

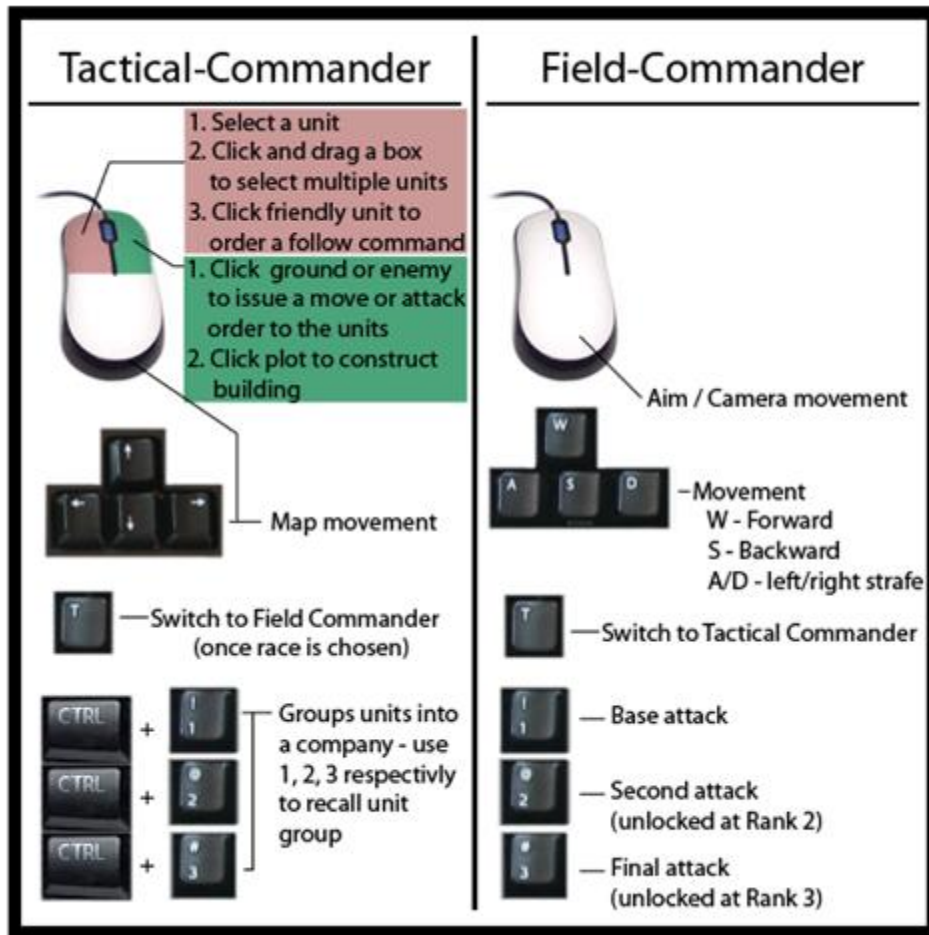


Figure 91: Controls Window

We split the controls into the tactical-commander and field-commander roles because we felt that it would be easier for the player to understand while providing clarity as to the difference between the two control schemes. We also felt that including images of a mouse and keyboard were helpful for the player to understand our meaning, rather than the traditional list of key bindings, as found in other games.

8.4.2.4. “Play the Tutorial”

When a player first installs *Blood Tide*, a pop-up window will appear once the game loads to suggest that the player try the tutorial level before beginning online play. This pop-up window directs players that are new to RTS games or want a better understanding of the game’s objectives and controls to play the tutorial. We did not

want to force players who are confident with RTS games or players who prefer learning a game through trial and error into the tutorial.

8.5. User Interface Help

Game developers can also add many things to the player's interface to help keep them focused on the goal and aware of their surroundings. Many games in all genres that have large worlds make use of maps in the menu system and mini maps in the corner of the screen to tell the player about his or her location. Maps and mini-maps often use a code of symbols to signify the player, other character, buildings, shops, and the objective location. A form of user interface help is shown in Figure 92.



Figure 92: Command and Conquer - Red Alert 2 Mini Map (Steeven)

In the game *Command and Conquer: Red Alert 2*, a real-time strategy game developed by Westwood Studios in 2000, the player has a mini-map in the top-right corner of the screen to show his or her buildings and units, as well as the enemy's base. This interface also shows the player how much money they have in the bank, seen above

the mini-map, and how close the player is to the maximum number of units, shown in the bar to the left of window displaying all of the trainable units.

8.5.1. Results for User Interface Help

We concluded that the user interface was extremely effective in guiding the player to the next objective but also provided the player with spatial reference. The use of mini-maps and icons was a common trend for the games. Icons used on the map to help guide the player were extremely helpful; however, the icons were not explained and were generic, seen in many other RPG's. A player who is not familiar with these icons may be lost. More research would need to be done on whether these icons are intuitive or an accepted part of the game genre. Figure 93 shows an example of one the interface help encountered in *Rogue Galaxy*.



Figure 93: Rogue Galaxy: User Interface Mini-map (“Rogue Galaxy”)

Rogue Galaxy and *Final Fantasy X* used a mini-map system that included icons noting save points, doors, and shops. Additionally, a red arrow is used to point the player in the direction of their next objective. *Star Ocean: Till the End of Time* used icons to note doors and people; however, the level design was so complex that it would

have benefited from the arrow to direct the player. *Paper Mario: The Thousand Year-Old Door* and *Baten Kaitos: Eternal Wings and the Lost Ocean* did not use a mini-map or icons; however, both games had a linear game world so the player was not confused of the destination. Table 19 shows the full results for user interface help.

Table 19: User Interface Results

User Interface Help	Total Games with Form of Help	Weighted Total
Is there a mini-map or map?	3	4.17
Are there icons/symbols that guide player?	3	4.17

8.5.2. Implementation of User Interface Help

The user interface help was very important to *Blood Tide*; because of the complexity of our game, the player needs to manage a large amount of information at all times. The interface was designed and implemented to be as unobtrusive to gameplay as possible, while still giving the player all of the information he or she needs. As with some of the other pieces of help, we created two different styles for the interface to reflect the feel of each faction.

8.5.2.1. Tactical-Commander Interface

The tactical-commander interface has four key components present, as can be seen in the Human faction's interface in Figure 94. This interface is also described in section 6.2.



Figure 94: Implemented Tactical-Commander Interface

As highlighted in the figure, the player has access to the help window, the chat window, the unit stats, and the mini-map. All these interface components, with the exception of the chat window, provide help to the player. The help window will provide the player with general help pop-ups and reminders. The unit stats window displays all of the units selected or detailed information about one selected entity. Additionally, if the player has not selected a field commander, a transparent field-commander selection triangle appears above the chat window in the bottom left side of the screen.

8.5.2.2. Building Interface

The building interface will display all the buildings the player can construct and a description of the building functionality when a player right clicks on a given plot, as shown in Figure 95. Icons were created to represent each of the faction's buildings, as can be seen in Section 4.3. The help window will inform the player when the building has finished constructing and new construction may begin.



Figure 95: Implemented Building Interface

8.5.2.3. Field-Commander Interface

As in the tactical-commander interface, four key interface components are present in the field-commander interface, as can be seen for the Piscivian faction in Figure 96. Most of these components are the same, only the field-commander selection triangle has been removed and unit stats pane has been changed. Instead, the field-commander stats pane displays the health, rank, and abilities available to the player. This interface is also discussed in section 6.3.



Figure 96: Implemented Field-Commander Interface

8.6. Other Character Help

Help through other characters is an extension of the mentor character. Ernest Adams notes the usefulness of non-playable characters (NPC's) to give the player information (460). While these NPC's usually aid the player, sometimes a playable character or even the main character can give the player help. Therefore, the category was expanded to include all characters other than the mentor.

Other characters have a variety of uses to help the player or develop the narrative. They can be used to remind the player of his or her goals, warn the player about what to expect in the near future, remind the player about events that just occurred, give the player back story about the location or local people, remind the player of learned skills, or give the player tips about hidden items or skills. Other characters will provide help in conversations, either while in the game world or during cutscenes. An example of other character help is shown in Figure 97.



Figure 97: Okami - Amaterasu listening to Sakuya (Metallian)

Character help can be found in almost all genres of games, but are most common in action, adventure, and RPGs. For example, the game *Professor Layton and the Diabolical Box*, an adventure/puzzle game made by Level-5 in 2009, uses cutscenes whenever an important plot point occurs like the beginning of the story or the revealing of the criminal. In the game *Okami*, an action-adventure game developed by Clover Studio in 2006, the protagonist Amaterasu is a wolf and not able to talk with other characters, but can listen to what the other characters say as they talk to themselves. From listening, the player learns about what is going on in the village and how to help the people.

8.6.1. Results for Other Character Help

We concluded that using other character to provide help to the player was effective if supplemented with other forms of help; this form of help is ineffective when used alone. The common trend was to give the player tips using other characters. *Baten Kaitos: Eternal Wings and the Lost Ocean* relied solely on NPC's during gameplay to initiate the dialog tree help. Players skipped this optional in-game help and were confused during battle. *Paper Mario: The Thousand Year-Old Door* and *Final Fantasy X* included help through NPC's on map as a supplement to other help, but these tips

were optional. In *Rogue Galaxy*, the main character, Jaster, would often give audio tips that indicated the player’s next objective while exploring the map.

Using other characters in cutscenes to guide the player to their next objective was a common trend and useful; however, cutscenes cannot be the only form of help given to the player, because the player a) gets frustrated with the length of the cutscene without direction and b) can forget the exact direction in the cutscene and can never replay it again. In *Star Ocean: Till the End of Time* the player only received directional guidance through lengthy cutscenes and this was notably frustrating to the player. The highly-rated games for in-game help all used other characters in cutscenes to provide help, but relied on other genres of in-game help to guide the player. Table 20 shows the results for other character help.

Table 20: Other Character Help Results

Other Character Help	Total Games with Form of Help	Weighted Total
Is help provided by other characters optional?	2	3.13
Are tips given by NPCs in the game world?	2	3.13
Is help provided in cutscenes with NPCs (goals, ex)?	4	4.25

8.7. Dialog Tree Help

Ernest Adams discusses dialog trees in relation to conversations in which the player has an option of response, with each choice offering a different reply (460). While he does not name this type of help specifically, dialog tree help works in the same manner as the conversation dialog tree he describes.

Dialog tree help will display a window that the player can interact with. Usually dialog tree help has several categories of help, such as movement, fighting, guarding, or item usage, in which the player can select from. For instance, if a player wanted to learn how to fight he or she would select the “how to fight” option to learn the game mechanics of fighting.

Dialog tree help is completely optional. The player has the option of learning about all categories of help, some of the help, or none of the help. Players are given the choice of how much to learn during a dialog tree help. Dialog tree help minimally breaks the immersive experience because the player has power over help.

In *Fire Emblem: Path of Radiance*, a tactical RPG developed by Intelligent Systems in 2005, the player has the option of going to a tutorial zone during the Prologue: Mercenaries. When in the tutorial the player is presented with an interactive window that gives the option to learn about relevant game mechanics. The player can select a variety of topics such as movement, direct combat, shove, or leveling up. These mini-tutorials are completely optional and the player has the option to skip them. In the case where the player accidentally skipped a topic, *Fire Emblem: Path of Radiance* allows the player to relearn game skills under the “Guide” menu.

8.7.1. Results for Dialog Tree Help

We concluded that dialog tree help was not an effective way to teach the player. While, in theory, giving the player the ability to choose what they want to learn is useful, we found that this form of help was boring to the player and always skipped. The common trend was to not use dialog tree help. Figure 98 shows one of the dialog tree help windows encountered during our test.



Figure 98: Baten Kaitos - Dialog Tree Tutorial (Sora7keys)

In *Baten Kaitos: Eternal Wings and the Lost Ocean*, the player was able to talk with NPC's in the world, which would bring up dialog tree help to allow the player to find out about different game mechanics. However, players briefly read only a few of the topics and did not desire to read more. They felt that they would rather learn the game skill through trial and error than reading. In *Star Ocean: Till the End of Time*, the player is given dialog tree help when he or she enters the virtual reality battle simulator. Similar to the dialog tree help in *Baten Kaitos: Eternal Wings and the Lost Ocean*, the player chose not to read all, if any, topics offered, preferring to try to play on their own. We noticed that players were very lost in both of these games, even when able to initiate the dialog tree help again. The results for the dialog tree help is shown in Table 21.

Table 21: Dialog Tree Help Results

Dialog Tree Help	Total Games with Form of Help	Weighted Total
Is help provided through a dialog box the player can interact with?	2	2.38
Is it conversation-based?	1	1.75

We decided not to include this form of help into *Blood Tide* because this form of help scored low on our effectiveness scale. As our game is a multiplayer online game, it would be difficult for the player to read this form of help during the game. Additionally, because this form of help was not as interesting to the player, we felt that we should focus our efforts on creating other forms of help for our game that will keep the player engaged.

8.8. In-Ear Guide

An in-ear guide is an extension of the “other character” category of help. This term is not an industry standard, but a description of the function of help that it provides. In-ear guides were separated from other character help because, during the period when this character is giving help, the in-ear guide is not a physical game character, nor can the player communicate with this character at will.

An in-ear guide is a character in the game narrative that typically can see the big picture of the mission and provides a relay of information about what is happening around the player. In a war themed game, the in-ear guide is generally a higher ranked officer who does not take an active role in the fighting. These characters often give the player objectives during a mission and advice about fighting enemies. This form of game help gives the player feedback about their actions in real-time, so the player always knows what to do and what is coming. An example of an in-ear guide is shown in Figure 99.



Figure 99: Halo 3 - Master Chief and Cortana (Cam)

In-ear guides are most often found in action games or first person shooters (FPS). For example, in the game *Mass Effect*, an action-RPG developed by BioWare in 2007, the player is guided by his or her ship captain, Joker. Joker is the voice of their ship, the Normandy, while the player is ashore, and often acts as the player's eyes and ears when the player is off the ship. In the *Halo 3*, an FPS developed by Bungie in 2007, the player has both Cortana, an AI cloned from a praised scientist, and Sergeant Johnson, an equally ranked officer in the UNSC Marines, who both feed the player information during missions.

8.9. Synopsis

Ernest Adams defines synopsis as a summary of the game narrative. Although he does not talk about a synopsis as a form of help, other games, such as the *Pokémon* series and *Tales of the Abyss*, have utilized a storyline summary to aid the player. Therefore, the term "synopsis" will be used to describe this form of game help.

The synopsis reminds the player of the previous game story and gives hints on the next objective the player must complete in a game. In-game guides are a sub category of the synopsis and provide the player with a database of abilities the player has mastered or important characters the player has encountered during play. Synopses tend to be used in games with heavy story and many cutscenes, such as RPG's, but also can be seen in game with lots of different characters or abilities. The synopsis acts as a way of reviewing the information that was given in previous cutscenes. This can be

effective because help or direction given by NPC's during cutscenes cannot be replayed if unclear.

The synopsis works similar to the printed strategy guide in that new story information and character background information can be provided outside the immediate gameplay. However, the synopsis does not break the magic circle because the player does not need to leave the game world to obtain information. Additionally, synopses will not accidentally spoil any future story because only events that have occurred during the game are recorded. An example of synopsis help is shown in Figure 100.



Figure 100: *Kingdom Hearts 2* - Jiminy's Journal (Haynes)

Kingdom Hearts 2, an RPG developed by Square Enix in 2006, has a synopsis in the form of a journal kept by Jiminy Cricket and will keep track of all events and characters encountered during play. This journal both acts as a way to learn more plot related elements that are unclear during the game, such as character relationships, and acts as a reminder of quest statuses in each world.

Chapter 9: Testing for *Blood Tide*

Testing is an integral part of the development of any game to find broken features, locate game-crashing bugs, discover which areas need improvement, and gain new ideas for features that could have been overlooked but would vastly improve the game's playability. It also helps to assure that the player is experiencing the game the way that our design intended; the game aligns as closely as possible with our game vision.

To thoroughly test *Blood Tide*, we created 11 test plans to use for in-house testing (Appendix O: Menu Navigation Test Plan through Appendix Y: Overall Help Functionality Test Plan) that tested the functionality of various parts of the game, conducted ad hoc testing to verify the overall game functionality and playability on a number of players, and finally tested the effectiveness of our in-game help by following the method presented in Chapter 8: In-Game Help – Results and Implementation and applying the coding sheet seen in Appendix Z: Ad Hoc Testing Results. Each section of this chapter focuses on one of these three different testing methods.

9.1. In House Testing

The first session that we conducted was an in-house testing session. In-house testing focused on assuring that the absolute critical features of gameplay functioned correctly and aligned with our initial game vision. For the in-house testing session, we strictly followed the test plans in order to locate errors with the ability to easily reproduce any bugs. We followed a strict procedure for the test plans and passed each step of the procedure if the expected result matched with the actual result experienced. For example, the step with the player actions, “select the human faction and press start,” we expected that the loading screen should appear, the game should start, and the player should be controlling the human faction. If all of these conditions were true, then the step was successful and we moved on to the next step of the testing procedure. However, if the actual result did not match the defined expected result, that step failed and so we did not continue with the rest of the test plan. When a failure occurred, we

wrote a comment indicating the actual result and took a screenshot, if applicable, to provide a reference for easier bug fixing.

9.1.1. Results from In-House Testing

In-house testing proved valuable for discovering and documenting many reproducible bugs. The RTS Mode Unit Functionality Test Plan, as shown in Appendix R: RTS Mode Unit Functionality Test Plan, is an example of a test plan that successfully lead to the finding and documenting of a bug. We discovered that certain units would spin around in circles when certain orders were given, such as attacking an enemy headquarters. This bug was high priority because it interfered with players being able to control their units accurately, diminishing the level of fun the player would have while playing as the tactical-commander. We were able to fix this bug for our final release.

However, not all the test plans were useful for documenting bugs. We found that the Artistic Functionality Test Plan, as shown in Appendix X: Artistic Functionality Test Plan, was unnecessary for documenting aesthetic bugs. While the test plan provided guidance on which art assets needed to be accounted for, we found that discovering visual problems was more easily done through simple ad hoc testing. For example, some of our testers found that three of our models were not properly displaying their textures: the pirate ship, the honeycomb coral, and the Human farm. Instead, they appeared entirely blue due to the technical aspects of the textures interfering with the blue fog that shrouded the map. With broken textures like these, the environment was less immersive since our game vision outlined a natural undersea world. We were able to fix this bug for the final release of *Blood Tide*.

9.2. Ad Hoc Testing

The second session that we conducted was an ad hoc testing session. Ad hoc testing was important for gathering feedback from fresh eyes that had never seen our game before, allowing for some very valuable comments. For ad hoc testing, we recruited volunteers from the Interactive Media and Game Development (IMGD) major as well as other friends with previous experience playing video games to test *Blood Tide*

without following any structured test plan. *Blood Tide* was installed on four computers in the IMGD lab, which we reserved for testing.

The testers were matched up to play against another tester and were told only about the general theme and gameplay mechanics. We tried to avoid giving the testers any more information than this on purpose, so that any confusion on their part about any game features or concepts would manifest naturally. They were given a notebook to record their comments and suggestions about the game that arose during play. Performing testing with people outside of the development team and familiar to games would better simulate our target audience for *Blood Tide*. These comments were collected and compiled into one file, seen in Appendix Z: Ad Hoc Testing Results Database. Each piece of feedback was rated on a scale of priority so that the time spent fixing bugs and adding new features would be as constructive as possible.

9.2.1. Results from Ad hoc Testing

The ad hoc testing was probably the most effective form of testing that we conducted, because it encompassed forms of testing on all levels of our game experience. Players were able to locate bugs that we found through functional testing, and in addition, players gave suggestions on how the help could be improved. During our first ad hoc testing session, five different testers played through at least one entire match with another tester. We received over 50 bugs and suggestions, which can be seen in Appendix Z: Ad Hoc Testing Results Database. The bugs were rated based on two factors: how often the bug appeared, and how critical the bug was to our game vision. Knowing we would not have time to fix every single issue, this ranking system allowed for us to spend our time most efficiently by focusing on the most critical things first. Fortunately, several critical bugs that were vital to the game's completion were found through this form of testing and we were able to fix all of them for the final release of *Blood Tide*.

The highest priority bugs were those that crashed the game. To give the players the most fulfilling experience, they had to at least be able to play through an entire match of *Blood Tide* from beginning to end. Ad hoc testing revealed a few bugs that

crashed the game or caused a disruption of the network connection, so these bugs were the ones we spent time fixing first.

Other categories of bugs were technical, artistic, help-related, balancing, or other. A technical bug is defined as a gameplay mechanic that functions incorrectly, like the player clicking on a base with the intention to view details about the base but the user interface does not display this information. Technical bugs were also defined as a form of feedback about a technical feature rather than an error. For example, many players complained that when they clicked on the mini-map, it did not move the player's camera to that point, as expected. We realized we had overlooked this feature when designing the game vision, so to improve the game vision we added this feature in. Artistic bugs can be errors, such as art assets displaying incorrectly in game, but they were more often in the form of feedback describing a particular artistic feature. For example, some players were confused about the floating mines and whether they were an object related to gameplay or simply part of the environment. Help-related bugs were those in which the feedback given to the player was not as clear as it should have been. Balancing bugs concerned the issue of both factions being relatively equal in power and whether that meant some units, bases, buildings, or commanders were overpowered or underpowered in some particular manner. Finally, bugs or feedback that did not fit into the other categories were classified as other.

While the testers found many bugs that hindered gameplay, they also came up with good ideas concerning features that *Blood Tide* lacked that would otherwise improve our overall game vision. We rated these ideas along with all the rest of the bugs, once again desiring to add some of them in but knowing that we would not have time to implement every one.

9.3. Game Help Testing

The third session that we conducted applies to the in-game help created for *Blood Tide*. We moderated the testing of six people who are familiar with video games to evaluate the game help in *Blood Tide*. We created a specific method for this particular testing session to provide a clear plan on how to evaluate the effectiveness of the in-game help. Out of the six people who tested the help in *Blood Tide*, three of them were

instructed to play the tutorial before playing the main game and the other three were instructed to skip the tutorial and play the main game. We wanted to assure that the player is able to learn the necessary skills needed to play *Blood Tide* without the tutorial. Because we had two separate scenarios for testing the help in *Blood Tide*, we obtained two different sets of results and comments from players.

The goal of this study is to determine whether the in-game help that we implemented into *Blood Tide* was effective in teaching the player important game skills. We followed a similar process used in our genre analysis to achieve this goal:

1. We prepared a coding sheet to test the help in *Blood Tide*.
2. We conducted a test session and surveyed subjects to rate the effectiveness of help in *Blood Tide*.
3. We analyzed our results using generic criticism to determine whether we succeeded in providing effective help to the player.

This section will discuss the method we used to achieve our goal. The section is broken down into three subsections and will discuss the results from testing.

9.3.1. Objective 1: Prepared a Coding Sheet

To test the effectiveness of in-game help in *Blood Tide*, we prepared a coding sheet (Appendix AA: *Blood Tide* Help Effectiveness Test Plan). The coding sheet contains four different categories, which corresponded to the forms of help that we implemented in *Blood Tide*: tutorial level, mentor character, pop-up windows, and user interface help. Each category specifically focuses on the whether the help was obstructive to gameplay and whether the help was clearly stated; the combination of which will allow us to assess the effectiveness of our help. We felt that if the player was able to clearly understand our help without it hindering gameplay, then we can deem our help effective. If we are successful in our implementation of help, we would prove the results from our genre analysis study could be applied to other game genres.

The obstruction of in-game help refers to help provided to the player that could hinder the gameplay experience. We looked for whether the player feels that too much help was being presented at once, which can be risky in breaking the immersion of the game. When control is removed from the player for an extensive period of time, the

player may begin to lose interest in the game. The obstruction of in-game help was a topic that we never considered while we were testing the presence of in-game help and observing which form of help was most preferred, because we were not interested in whether the help hindered the gameplay. Now that we are testing the effectiveness of our help in *Blood Tide*, we wanted to assess whether our implemented help was hindering the player's game experience.

The second subcategory of our coding sheet was for testing the clarity of our in-game help. We will be looking for whether the help clearly demonstrates the necessary skills needed to win *Blood Tide*. The clarity of the help will measure through survey how well the player learns movement within the game, the controls for playing field-commander, the controls for playing tactical-commander, building controls, unit strengths and weaknesses, and the victory condition.

9.3.2. Objective 2: Tested Games and Recorded Presence of Game Help

We selected several people of different backgrounds from our college to play *Blood Tide*. We chose people who had some familiarity with computer games because we wanted people to have an understanding of basic computer mechanics and game conventions. We did not want people to be concerned with mouse movement and navigation through menus. Additionally, we wanted to eliminate the variable of player's intuition during testing.

We assigned half of the testers to first play *Blood Tide*'s tutorial before playing an online game, and we assigned the remaining half of the testers to begin playing the main game without playing the tutorial. We wanted to assure that the player is able to learn the necessary skills needed to play *Blood Tide* without need of the tutorial; therefore, we applied our coding sheet for both these situations.

During testing of the help in *Blood Tide*, we designated several people to play the game while we took on the role of a mediator.

- **The Player** either began with playing the tutorial before playing the main game or skipped the tutorial and began with the main game. The player played the main *Blood Tide* game for one complete session, usually lasting at least 10 minutes, before answering the questions on the coding sheet.

- **The Mediator** was responsible for observing the in-game help session. The mediator helped all the players open the *Blood Tide* executable, told the players whether to begin with the tutorial or skip the tutorial, and then kept quiet unless a player was frustrated with understanding the game controls. If the mediator clarified any of the help in the game, he or she was responsible for writing down which help needed clarification. After all players finished testing *Blood Tide*, the mediator was responsible distributing the coding sheet and explaining each of the categories on the coding sheet to insure accurate data. After all players have finished evaluating the help in *Blood Tide* by following the coding sheet, the mediator was responsible for collecting the coding sheets.

9.3.3. Objective 3: Analyzed Our Results

After applying our coding sheet, in Appendix AA: *Blood Tide* Help Effectiveness Test Plan, and compiling the data we received through our test, we analyzed which forms of help that we implemented into *Blood Tide* were the most effective in teaching the player. Then, similar to our initial goal, we wrote an analysis about what we have found through this method for future games to create more effective in-game help.

To determine the effectiveness for each form of game help, we chose not to create a new formula as we did in the first methodology. We felt that an average of the results was the best way to determine the effectiveness for a general player. We then compared the results to our original results for the type of help as we found in Appendix L: Results Summary. If the average for the implemented help was above the average found in the original test, we considered the help to effectively teach the player. If the implemented help average was below the original average, we considered the help ineffective. If the implemented help's average was very close to the original test's average, we were not able to draw conclusions about the effectiveness of the help, unless our observations note otherwise.

This section will be broken down into two parts. First we will discuss the results and observations from players who played the tutorial before playing the main game. Second we will discuss the results and observations from players who skipped the tutorial and played the main game instead. Lastly, we discuss which elements of the

implemented help worked in section 10.3, because we believe an evaluation of what worked is important in testing.

9.3.3.1. Results and Observations from Players Skipping Tutorial

We told three players to disregard the suggestion to play the tutorial and begin play of a main *Blood Tide* game. All players were familiar with playing video games; however, one of the players was new to the RTS genre. Therefore, we received mixed reviews regarding the clarity of the help.

Overall, the testers generally agreed that our help was not obtrusive to gameplay, as can be seen in Appendix AB: *Blood Tide* Help Effectiveness Coding Sheet Results for Playing Skipping Tutorial. However, throughout our study on game help, we have concluded that some level of obtrusion is needed to effectively provide help to the player. When reviewing the results and comments for testing game help, we felt that players were ignoring important pieces of help in the help window. Because players often ignored important help available to them, the clarity of our help was rated lower. Players new to *Blood Tide* wanted to be forcefully given the information. Therefore, in order to notify the player that new help is available, we added sound chimes whenever important tips appeared.

The players also agreed that the clarity of the help given was not perfect. For instance, the control scheme was not intuitive for players and, while the control scheme is not directly correlated to the help provided, the player's intuition took precedence over the displayed help. When the help window informed the player to construct a building by middle mouse clicking on a plot, we noticed that the players would continue to left click a plot instead of middle click, bang on the keyboard and say "it isn't working." Even though the help window specifically instructed the player to middle mouse click on the plot, the player's intuition was to left click. Our study on game help never took into consideration that a player's intuition would take precedence over the provided help. We feel that this topic would be a new avenue for a future study.

Additionally, there were two specific, noticeable problems with the clarity of our help. First, in the example of help informing players on how to construct buildings on plots, the player was confused about what a plot actually was. Terminology specific to

our game is never defined outside of the tutorial and, while all players eventually found the plot, the game never indicated where buildings could be built. Secondly, players liked the loading screen help, but wished the screen would last longer to obtain all the information.

We fixed some of the clarity related issues with the help by increasing the time that the player can see the loading screen before the game begins. Additionally, as discussed in section 9.2, we altered the control scheme for *Blood Tide* to better reflect player's intuition. In addition to several people suggesting to change the control scheme during ad hoc testing, we felt that the unintuitive control scheme hindered the help and therefore, the controls were changed.

9.3.3.2. Results and Observations from Players Playing Tutorial

We told three players to play the tutorial before beginning the main game. All players were familiar with playing video games. The results of the tests were very positive towards the help.

Overall, players agreed that the tutorial help was not obtrusive to the gameplay, as shown in Appendix AC: *Blood Tide* Help Effectiveness Coding Sheet results for Playing Tutorial. Unlike in the main game, as discussed in the section 9.3.3.1 above, the player was expecting to be given help; therefore, noticed the help window immediately and began following the directions that it gave the player.

The testers differed as to the perceived clarity of help. After the initial round of testing, from which the tests for players beginning the game without playing the tutorial was ran, many updates were made to better clarify messages or problems with help windows. After these fixes were complete, the players found the new control scheme and help windows easier to understand. Also, as descriptions are much more detailed in the tutorial level, players were able to figure out how to perform important game actions with less frustration than the previous test.

While some balancing issues still remained in the tutorial level, the tutorial designed so that the player would always win if the person waited to spawn more units before attacking again. The initial enemy force was large enough to deter players from ignoring the tutorial messages and defeating the enemy base at the start of the level, but

was still too difficult to defeat when the player reached the final stage of the tutorial. The balance problem was changed after these tests to give the player less difficulty. However, because of time constraints, we were unable to fully implement a tutorial level that allowed the player to command all of the faction's units. As discussed in Chapter 10: Post Mortem, if we had more time to work on the project, we would have implemented a more robust tutorial like we had originally planned.

Chapter 10: Post Mortem

This chapter will serve as a postmortem for *Blood Tide*, reflecting on the project's successes and failures through the game development cycle. We also hope that, through the lessons we learned from our project, we may share our advice to future game project teams.

Looking at the project as a whole, we felt that the game accomplished and exceeded our main goals for *Blood Tide*; we were able to successfully combine two very different game genres into one game in one complete development cycle. Also, the in-game help that was implemented was somewhat successful at teaching the player how to navigate through the game world in both commander roles, even when the player did not play the tutorial level. We are also very pleased that we did not have to deviate from our original game design during the implementation phase. Our development team was able to create a task schedule and stay on schedule without much deviation - we hit every project milestone on time due to carefully managing tasks.

The largest hindrance during our development cycle was our team's lack of a defined project lead. When questions arose as to changing the original design, there was no single person that the team could look to for a final decision. Decisions were ultimately left to the group as a whole, which seemed less efficient than having a project leader make decisions.

One of the most beneficial decisions that we made during our design phase was to extend WPI's traditional three term MQP schedule into a fourth term. This allowed our team extra time to complete all of our desired artistic and technical assets, while still allowing time for testing and help integration. Having this extra term meant that we were able to build the game in a way in which it can be easily modified and extended in the future, instead of making a game that has no flexibility and barely holds together. Additionally, having a clearly defined design enabled the team to focus on implementation without having to revisit the game design when unexpected situations arose. We also benefited from having a very talented and dedicated team, and we believe that we would not have been able to complete a large project the size of *Blood Tide* if the team members were not as accountable for their work.

10.1. Technical Reflection

Overall, we are very pleased with the technical implementation in *Blood Tide*. Our project was ambitious because we combined two genres into one game and attempted to make it playable over the internet. All of these major goals were met. In addition to this, our game runs with more units in the game world and at a higher frame rate than we estimated in the technical design in section 2.5.2. To reiterate, we made an RTS with 70 units on screen at one time, all of the RTS features players expect – which is also networked over the internet, a new physics system, an action mode with action controls and RPG elements, robust artificial intelligence, single player capabilities with a tutorial level, a content pipeline that enables non-programmers to completely modify almost everything in the game, custom shaders, many small details like clean menus and loading screens, and our game has no memory leaks and runs at over 60 frames per second – we could not be happier with the final product.

Although we feel our game is a success, there are a few areas where we ran into problems. The first area involves some of our planned components. Before we even modeled entities in our game, we created optimization components. When we finally got to the implementation of the components that were to be optimized, we realized that they no longer needed to be optimized or that the optimizations were for features that were changed or removed.

One of our largest mistakes in the original design was to not fully think about how units fit into the environment. This affected our planned unit movement, path-finding, node-graphs, and collision. The way we envisioned the units moving was the same as if they were on land, like most other RTS games, where many obstacles are present that units need to avoid. The node-graph and path-finding components were designed to be able to get units to avoid static obstacles. Since our units in *Blood Tide* float above the sea floor, there are no static obstacles. These components that we thought were critical were not needed – so they were never implemented.

Much like the path-finding, our movement was to be based on navigating around static obstacles. One of the largest challenges we faced was to get the units to move when they were in large groups because we did not properly think about how to get units to avoid dynamic obstacles. This added a lot of development time, because the most

important feature in our game is being able to move units and have them attack enemies. If units just sit around and spin or start moving in the wrong direction, the game is frustrating and unplayable. We eventually fixed this by adding heuristics that help units manage avoiding each other when in large groups.

Originally, a zoning component was expected to be a critical component to our game. We figured that by adding zones automatically to the game, we would not have to hand-craft the zones and the game would run much smoother. When we implemented this feature, the game ran a whole lot worse. As it turns out, outdoor environments are not handled well by zones because zones are meant to separate areas that cannot see each other. In an outdoor environment, all of the zones can be seen from any point, so we added extra calculations to the game with no added benefits. Since this component made the game worse, we decided to scrap it entirely.

We did not think about the relationship between the two modes of play and the environment well enough during the design phase. Since we melded two genres together, this caused a few problems. We did not think that an RTS feature could affect action game play and vice versa. The fog of war system was most affected by this. The goal of fog of war is to simulate sight from a bird's eye view. Since our game also has an action mode where the player can see from a view actually in the world, we did not need to simulate what the sight would be like, because the player could already see the entire map. If we were to still create the fog of war system, enemy units would disappear at certain distances, but the environment around the units would not, which does not make sense. Since the most unique feature in our game is the ability to play in field-commander mode, we decided to use the field-commander mode sight restrictions instead of the RTS ones to emphasize field-commander mode. We compensated for removing the fog of war by making the world seem underwater, which naturally has limited sight because everything eventually fades to blue.

Our other problematic areas involved underestimating some components. One of these components is the user interface. Originally we wanted a minimalistic interface, but quickly realized that RTS games often have many user interface elements for good reason. We were forced to spend a few weeks implementing many elements that we had neglected in our original design. Our game now gives the player a lot more information

than we had anticipated, but based on our feedback from testing, the extra effort made the game much better.

One of our major successes was the implementation of our content pipeline. This component was originally planned as a way to import art assets into the game, but quickly grew to involve hints, game logic, game scripts, and more. Our system ended up being very powerful for quickly changing the game. For example, if we want to add a new unit to the game, we only need to copy approximately 30 lines of text and change some parameters around, such as the unit's name, its stats, and what abilities it has. Much of the information specified in the text files is functionally maintained in the C++ code, but we have provided enough functionality to our text files that we can express a wide range of possible new units with minimal effort.

Another major success was our networking code. Our game can have up to 70 units on screen at once. If all of these units are moving and attacking, we need to constantly be sending position updates, heading updates, stat updates, animation updates, and ability updates (such as firing a torpedo) for most of the units. Not only do we have to send a high volume of updates, but the networking must work smoothly because, if it does not, the game is unplayable. Not only did we properly synchronize everything that is happening in the game, but we made sure that we were doing it efficiently. Under normal circumstances, the server is sending 0.1 kilobytes per second (Kbps) of data. Under medium loads the server sends between 2 and 6 Kbps. We have not seen server loads exceed 15 Kbps, which is a very good amount for games. Since an average game lasts ten minutes (based on our testing), this means that our game sends approximately 3 MB of data per session.

Our biggest success in the implementation of *Blood Tide* involves the physics system, detailed in section 2.5.3.3. The C4 Engine's built in physics system could not handle as many units as we had hoped. Though it added development time, creating our own physics system saved our project from drastic changes. We were able to create a system that was simple, but suited our needs, in approximately two weeks. Before we implemented this physics system, we could only support 8 units in the game, which ran at approximately 20 frames per second. With our new physics system, we can have somewhere on the order of 70 units in the game and still maintain 60 frames per second.

10.2. Artistic Reflection

We feel that we successfully captured the underwater feel of the game, as well as the individuality of each faction. RTS games typically are very challenging artistically because of their wide variety of units, structures, and animations; therefore, we are very proud of the amount of content we were able to implement in the game. We were especially proud of our artistic assets, because they are visually appealing for both modes of play, even though they look drastically different in each commander role. The ability for all of the art assets to be discernable from a distance but also appear to be detailed up close was challenging, and we are very pleased with the results of our work. Although the art team encountered some problems during the implementation phase, we were able to create all of the art assets that were defined in the artistic vision on time, as we outlined in the design phase.

One problem that we faced during the project was the need for a third artist. We had originally planned and hoped to find a third artist for our team, but we were unable to find another member who had the time and experience available to commit to a four-term MQP. We were able to have some help by outsourcing art during the most intensive implementation phase; however, as this person was not a member of the MQP team, the commitment to the original artistic vision was not the same. This led to several of the art pieces not reflecting the artistic vision and were inconsistent with the remaining models. Because our goal was to have a consistent theme for each faction, we chose to re-do art pieces to better fit the defined artistic theme. However, constantly changing art assets that were expected to be complete took time away from the creation of other assets that needed to be implemented, including help. In spite of the challenges with this person, we were able to completely redo all of the artistic assets that did not fit with the game's artistic style, so we are pleased with how the artistic vision was implemented.

We also faced a few challenges when using the C4 Engine. We had originally planned for the environment, especially the plant life, to move with the undersea currents. However, we were unable to find a way to make this function properly in the game and were forced to cut this feature. We had also planned to create all of the terrain in the engine. However, we were forced to create a mesh in Maya because we were

unable to make the terrain in the world editor texture correctly, nor could we easily simulate light reflecting on the ocean floor using the engine's built-in terrain. In addition, we faced challenges with the pipeline to get artistic assets into the game, specifically for the animations. We found, after almost two terms of problems, that the simple action of deleting groups enabled all of the animations to work properly for the merman and scuba troop units. This group problem also interfered with the way that textures interacted with the blue fog, distorting the original colors.

Overall, we believe that the choice to create two separate, clearly defined artistic themes for each of the factions was very successful in creating a unique feel for both players and an immersive environment. This also helped to focus all of the art team's aesthetic decisions during implementation, for example, when creating the game environment. Our schedule also was very beneficial in keeping the project on track but was also flexible enough to allow for shifting as other course loads became more or less intensive.

10.3. Game Help Reflection

Overall, we are very pleased with the implementation of help into *Blood Tide*. The research into game help proved to be very helpful in our design and we believe that our test results reflect what we have observed in other games as well as the tested games. Therefore, we believe that we have created a comprehensive listing and explanation of the various types of in-game help that can be used for future games. Our help was also able to teach players effectively how to play the game. Even when the players did not play the tutorial and were thrust into online play, they were still able to learn the game within five minutes of play, once the player learned how to utilize the help window. We are very proud of our successes in teaching game skills to players who have never played *Blood Tide*.

During implementation, we were faced with challenges in time and limitations in the game engine. The art and tech teams were forced to focus time the implementation of the game assets, which were more important in creating a working prototype than the implementation of the help. Therefore, the implementation of all of the help was pushed into the very last term. Everything was able to be completed as planned; however, it

would have been beneficial to have a team of people to work on art and tech assets and a second team to work on help implementation. Also, as help is not native to the C4 Engine, the tech team needed to spend additional time creating scripts for a separate help message window and a single player tutorial level.

Through our testing, we found that the most effective help was some form of pop-up window. We believe this was helpful for the player because it gave solutions in the middle of play, especially in battle when players are under pressure to work quickly. Not giving the player the option to skip the help was also very effective. When given the option, the player always skipped the help, intentionally or not, leading to confusion about how to proceed. We have learned that the most effective games applied multiple forms of in-game help to maximize the player's chance to learn the game skills without constant repetition.

Once the help was implemented, both the tutorial and the main game help were able to effectively teach the player important game skills. While the help that was provided ended up being unobtrusive to play, players wished that the help had been more obtrusive so that it would be more noticeable. As expected, teaching the player through the tutorial proved to be more helpful than putting the player directly into online combat. We believe that our in-game help was useful and aided to the player experience. However, we did not have the time during the implementation phase to correct all of the problems and incorporate the suggestions we received through the testing process. While the in-game help for *Blood Tide* could have been improved, we were successful in aiding the player and enhancing the player's experience in the game, so we feel that this portion of the project was successful.

In future work, we would like to focus more time on games that used in-ear guides and synopsis; both of these forms of in-game help were not as prevalent in the games we chose to study. Several other forms of help require more testing to fully analyze their usefulness to the player. If we were to continue this study, we would also have chosen more games to study for all consoles, as we only studied games made for the PlayStation2 and GameCube. While our tests were a beginning to analyzing in-game help, more testing is needed to definitively decide the best forms of help.

References

- Aareseth, Espen. *Playing Research: Methodological approaches to game analysis*. 23 August 2003. 29 September 2009
<<http://www.spilforskning.dk/gameapproaches/GameApproaches2.pdf>>.
- Aarseth, Espen. "Genre trouble: Narrativism and the art of simulation." Harrigan, Pat and Noah (ed.) Wardrip-Fruin. *First Person*. Cambridge: MIT Press, 2005. 45-55.
- Adams, Ernest. *The Fundamentals of Game Design*. 2nd ed. Berkely, California: New Riders. 2009.
- Andrei, Mihai. "The deadliest creature in the world." 3 August 2009. *ZME Science*. 1 April 2010. <<http://www.zmescience.com/science/oceanography/the-deadliest-creature-in-the-world-000030/>>.
- Bethke, Erik. *Game Development and Production*. Plano: WordWare Publishing Inc., 2003.
- Biocca, Frank and Mark Levy. *Communication in the age of virtual reality*. Hillsdale, New Jersey: Lawrence Erlbaum Associates. 1995. Pages 57 – 124.
- Boharsik, Darr. "Greek Garden." *Camera Artist*. 1 April 2010.
<<http://cameraartist.com/index.php?showimage=113>>.
- Business Week*. "International Design: National Grand Theater, Beijing." 1 April 2010.
<http://images.businessweek.com/ss/05/12/china_wonders/source/11.htm>.
- Cam. "Halo 3 campaign walkthrough - The Storm part 2." 28 July 2008. *YouTube*. 2 October 2009 <<http://www.youtube.com/watch?v=z-zVTTReEcI&feature=Playlist&p=DBB4C4C09C1A3EFA&index=9>>.
- Charles, Darryl and Livingstone, Daniel. "Intelligent Interfaces for Digital Games." 2004. *Association for the Advancement of Artificial Intelligence*. 2 October 2009
<<https://www.aaai.org/Papers/Workshops/2004/WS-04-04/WS04-04-002.pdf>>.
- Chuggaacroonoy. "Paper Mario: The Thousand-Year Door - Prologue - Episode 1." 27 February 2009. *YouTube*. 13 December 2009.
<<http://www.youtube.com/watch?v=7ebdqIkZxko>>.

Colin. "Low cost geodesic dome greenhouse kit." *Geo-Dome*. 1 April 2010. <http://www.geo-dome.co.uk/article.asp?uname=geo_polydome>.

"Coral Reef." *NaturePhoto-CZ*. 1 April 2010. <<http://www.naturfoto-cz.de/coral-reef:coral-photo-2236.html>>.

DaCheif. "USS Enterprise – Famed WWII Airvraft Carrier." *StrangeMilitary*. 6 April 2010. <<http://www.strangemilitary.com/content/item/103414.html>>.

David K. "If it's Tuesday, This Must be Belgium – 8/26/08." 1 April 2010. Online image. <<http://travel.gather.com/viewArticle.action?articleId=281474977430034>>.

Davies, Sarah. "Arkagas (Agrigento), Sicily: Doric Temple of Concordia." 7 February 2005. *University of Texas*. 1 April 2010. <<http://www.utexas.edu/courses/introtogreece/lect8/img9tempconc.html>>.

de Pastino, Blake. "Photo In the News: First Giant Manta Ray Born in Captivity Dies." 21 June 2007. *National Geographic*. 2 April 2010. <<http://news.nationalgeographic.com/news/2007/06/070618-manta-ray.html>>.

Emrich, Alan. "The Decline of the Official Computer Game Strategy Guide." *Alan Emrich Homepage*. 13 December 2009. <http://www.alanemrich.com/Writing_Archive_pages/decline.htm>.

Federoff, Melissa A. "Heuristics and Usability Guidelines for the Creation and Evaluation of Fun in Video Games." 2002. *Melissa Federoff*. 13 December 2009. <http://melissafederoff.com/heuristics_usability_games.html>

Game FAQs: Kingdom Hearts. 1 October 2009 <<http://www.gamefaqs.com/console/ps2/game/516587.html>>.

gift ideas. "Gazebo Roof Dome for Wedding Outdoor Garden." 26 January 2010. *Garden*. 1 April 2010. <<http://garden.onsugar.com/Gazebo-Roof-Dome-Wedding-Outdoor-Garden-7172760>>.

Hauser, Joachim. "Hydropolis Underwater Hotel, Dubai, United Arab Emirates." *Design Build Network*. 12 October 2009. <<http://www.designbuild-network.com/projects/Hydropolis/Hydropolis4.html>>

Haynes, Jeff. "Kingdom Hearts II." 28 March 2006. *IGN*. 12 April 2010. <<http://ps2.ign.com/articles/698/698697p1.html>>.

- Helmert10. "Paper Mario the Thousand Year Door Co-Op Walkthrough Part 2 – Professor Frankly." 7 February 2009. *YouTube*. 13 December 2008.
<<http://www.youtube.com/watch?v=8F2r9IDxmhM>>.
- Hmyaa. "mkwii Wrong way go back." 27 July 2009. *YouTube*. 13 December 2009.
<<http://www.youtube.com/watch?v=3ZhNBr3hkVY>>.
- Isbister, Katherine. *Better game characters by design: a psychological approach*. San Francisco: Elsevier, 2006.
- "Jak & Daxter Playthrough 1 - Intro." January 2009. *Google Video*. 1 October 2009
<http://video.google.com/videoplay?docid=-8576991548060581735&ei=_SvISsfHO4LIqQKpz7yZDQ&q=jak+and+daxter+intro&hl=en&client=firefox-a#>.
- Jeff "Joker" Moreau. 2 October 2009
<http://masseffect.wikia.com/wiki/Jeff_%22Joker%22_Moreau>.
- Kirbx. "Fire Emblem: Chapter 22 Treasure Map." 2 June 2008. Online image. *GameFAQs*. 13 December 2009.
<<http://www.gamefaqs.com/portable/gbadvance/file/468480/52991>>.
- Kuzmin, Vitaly V. *Flamber.ru*. 1 April 2010.
<http://flamber.ru/files/photos/1211922554/1234813347_f.jpg>.
- Lee, Jeannette J. "WWII submarine found 65 years after sinking." 24 August 2007. *NavyTimes*. 17 April 2010.
<http://www.navytimes.com/news/2007/08/ap_subfound_070824/>.
- Lombard, Matthew and Theresa Ditton. "At the Heart of it All: The Concept of Presence." September 1997. *Jcmb*. 13 December 2009.
<<http://jcmc.indiana.edu/vol3/issue2/lombard.html>>
- McElroy, Matt M. "Review, Dread: The First Book of Pandemonium." 29 January 2008. *Flames Rising*. 1 October 2009. <<http://www.flamesrising.com/dread-rpg-review/>>.
- Metallian. "Okami: Amaterusa listening to Sakuya." 31 December 2006. Online image. *Directional Pad*. 13 December 2009.
<<http://www.directionalpad.com/articles/reviews/okami/>>.

- “Rogue Galaxy.” 13 December 2009.
 <<http://shop.gameplay.co.uk/webstore/productpage.asp?productcode=PT2573>>.
- “PT 658 at Vancouver dock.” August 2005. *Save the PT Boat*. 4 April 2010.
 <<http://www.savetheptboatinc.com/WWII0192.JPG>>.
- “Poseidon / Neptune God of the Sea.” *The Plate Lady*. 24 April 2010.
 <<http://www.theplatelady.com/figurines3/poseidon-6140.jpg>>.
- Sangster, Laura. “Make a Difference – Caribbean Conservation Corperation.” 14 August 2008. *Caribbean Travel News*. 2 April 2010.
 <<http://caribbeantravelnews.blogspot.com/2008/08/make-difference-caribbean-conservation.html>>.
- “Scuba Diving 03 Jpg.” *Scuba Diver News*. 15 April 2010. <<http://www.scuba-diver-news.com/images/view/8758/scuba-diving-03-jpg.html>>.
- Slisenberger. “Scribblenauts – World 0 and World 1 TAS.” 20 September 2009.
YouTube. 13 December 2009.
 <<http://www.youtube.com/watch?v=FIoUAYplI4U>>.
- Sora7keys. “Baten Kaitos Eternal Wings and the Lost Ocean walkthrough p2.” 7 July 2008. *YouTube*. 13 December 2009.
 <<http://www.youtube.com/watch?v=ENtYh1ZOlK8>>.
- Steeven. “Command & Conquer: Red Alert 2.” 31 March, 2009. Online image. *NeoGAF*.
 13 December 2009.
 <<http://www.neogaf.com/forum/showthread.php?t=356965&page=6>>.
- Susan. “Do You Suffer From Sand Castle Management? Take this Litmus test.” 12 November 2009. *Susan Ireland & Associates*. 1 April 2010.
 <<http://susanireland.wordpress.com/2009/11/19/do-you-suffer-from-sand-castle-management-take-this-litmus-test/>>.
- Sun, Chuen-Tsai, Holin Lin and Cheng-Hong Ho. "Game Tips as Gifts: Social Interactions and Rational." November 2003. *Digital Games Research Association*. 1 October 2009. <<http://www.digra.org/dl/db/05150.35057.pdf>>.
- “The submairne pens at St Nazaire.” *South Brittany Cottage*. 1 April 2010.
 <<http://www.southbrittanycottage.com/photogallery3.htm>>.

Teskem. "165-monterosso bunker." 5 November 2002. *Webshots*. 1 April 2010.
<<http://travel.webshots.com/photo/1054351246014992534UsDvFX>>.

Ward, Kip. *Banjo-Kazooie: Prima's Official Strategy Guide*. Rocklin, California: Prima Publishing, 1998. pg 81.

Appendix A: Art Asset List

Table 22: Models and Textures Asset List

Models and Textures Asset List		
Units:	Merman Commander Texture	Will be used with Merman unit model
	Scuba Commander Texture	Will be used with Scuba Troop unit model
	Turtle Commander Texture	Will be used with Turtle unit model
	Submarine Commander Texture	Will be used with Submarine unit model
	Manta Ray Commander Texture	Will be used with Manta Ray unit model
	Bomb Ship Commander Texture	Will be used with Bomb Ship unit model
	Mermen Unit	
	Scuba Troop Unit	
	Turtle Unit	
	Submarine Unit	
	Manta Ray Unit	
	Bomb Ship Unit	
	Buildings:	Mermen Headquarters
Turtle Headquarters		
Manta Ray Headquarters		
Piscivian Farm		
Piscivian Training		
Piscivian Guard Tower		
Scuba Diver Headquarters		
Submarine Headquarters		
Bomb Ship Headquarters		
Human Farm		
Human Training		

Environment:	Human Guard Tower
	Destroyed Headquarters/Building
	Empty Plot
	Coral Reef
	Sea Weed
	School of Fish
	Rocks
	Sea Shells
	Sea Sponge
	Sunken Subs(WWII)
	Unexploded Bombs (WWII)
	Ruins
	Starfish
	Jelly Fish

Table 23: Animations Asset List

Animations Asset List	
Merman	Attack 1
	Attack 2
	Attack 3
	Move
	Death
Scuba Troop	Attack 1
	Attack 2
	Attack 3
	Move
	Death
Turtle	Attack 1
	Attack 2
	Attack 3
	Move
	Death
Submarine	Attack 1
	Attack 2
	Attack 3
	Move
	Death
Manta Ray	Attack 1
	Attack 2
	Attack 3
	Move
	Death
Bomb Ship	Attack 1
	Attack 2

	Attack 3
	Move
	Death

Table 24: Music Asset List

Music Asset List	
Song	Menu Song
	Command Mode Song
	Field Mode Song
SFX	Organic Unit Death
	Mechanical Unit Death
	Building Created
	Building Destroyed
	Unit Spawn
	Field-Commander
	Spawn
	Merman Attack
	Merman Move
	Merman Accept Orders
	Scuba Troop Attack
	Scuba Move
	Scuba Accept Orders
	Turtle Attack
	Turtle Move
	Turtle Accept Orders
	Submarine Attack
	Sub Move
	Sub Accept Orders
	Manta Ray Attack
	Manta Move
	Manta Accept Orders
	Bomb Ship Attack
	Bomb Move
	Bomb Accept Orders

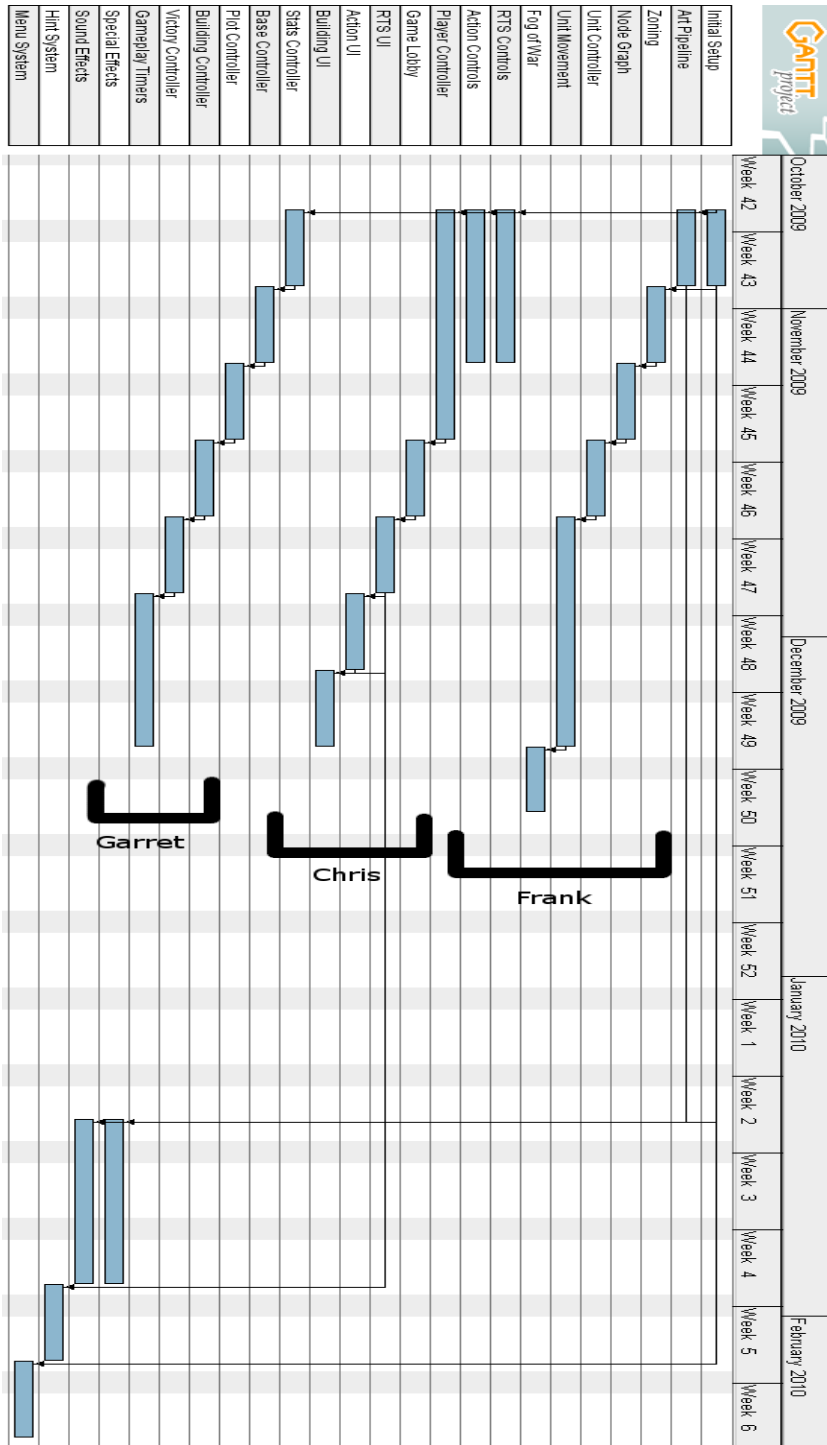
Turret Tower Attack

Jelly Tower Attack

Appendix B: Art Timeline

	Joseph Alea	Sarah Gilkey	Sean Beck
Week 1	Merman, Scuba Headquarters	Scuba Troop, Merman headquarters	Submarine, Sub Headquarters, Sub Animations
Week 2	Animations: Scubatrop, Merman Commander Tex.	Animations: Merman, Scuba Commander Tex.	Manta Ray, Manta Headquarters
Week 3	Turtle and Headquarters, Human Guard	Animations: Manta Ray, Pisc. Guard Tower, Empty Plot	Bombship & Bombship Headquarters
Week 4	Animations: Turtle & Bombship	Remaining environment, Destroyed Headquarters, Map	Human Farm & Univ, Pisc. Farm & Univ
Week 5	Menu art - human	Menu art - piscivian	Sub Commander Tex, Manta Commander Tex, Bombship Commander Tex
Week 6	In Game music, SFX - sub, turtle	Main menu music, SFX - manta, bomb	SFX - merman, scuba troop, general, buildings
Week 7	Finish Remaining Assets	Finish Remaining Assets	Finish Remaining Assets

Appendix C: Technical Timeline



Appendix D: Written Guide Forms

There are many reasons for a player to want to use a game guide. Some players want to get an advantage in the game, while some cannot solve a particular problem on their own. Sun, Lin, and Ho, three researchers from the University of Taiwan, found that “most players only use walkthroughs as general references, preferring to read them only when they get very confused or frustrated” (Sun et al., pg 3). Espen Aareseth, a game scholar who has written books and papers about computer games and co-founded an international journal of computer game research, wrote that, when playing *Myst*, “players are often stuck on one of the puzzles, and have no choice but either to buy the solutions book, download a ‘walkthrough’ guide from the internet, or give up” (“Genre,” pg 7). In an earlier paper titled *Playing Research: Methodological Approaches to Game Analysis*, he says that cheats and walkthroughs can be necessary to understanding a game, but can also take away the player’s enjoyment (pg 6). These guides can be integral parts of playing the game because of their complete instructions for all of a game’s puzzles, skills, and recipes needed through the narrative (Sun et al., pg 3). The biggest potential problem with guides is that “walkthroughs and other types of cheats can easily ruin the game;” they are called spoilers for a reason (Aareseth, “Playing,” pg 7). Aareseth believes that cheating makes players lose the flavor of the game, even though cheating is an extremely common practice among gamers (“Playing,” pg 5). Written guide forms can be an essential for players who need assistance. Game help needs further definition to aid game developers in understanding the process of teaching the player important game skills, even though these form of help may break the immersive experience.

Through our research, we have identified three different types of guides: printed guides, online guides, and in-game help. While each of these types has strengths and weaknesses, we believe that in-game guides are a very good way to provide help to the player without breaking the immersion of the game. Each of the different types of guides uses different tactics to provide help, each with its own strengths and weaknesses; therefore, all three media continue to co-exist. In-game guides and their strengths will be discussed in the section titled “In-Game Help.”

Printed walkthroughs found a niche in the game help realm because they contain game stories, screenshots, maps, and playing methods (Sun et al, pg 3). These printed strategy guides first began in the late 1980's with Japanese games like *Earthbound* in 1989. Soon after, the early 1990's brought printed guide companies such as Prima Games and BradyGames into America. Printed guides are typically written in conjunction with the creation of the game. This can create problems with continuity between the game and the guide. As Erik Bethke notes in his book *Game Development and Production*, "even the ultra-high-profile game Gran Turismo 3 for the PS2 contains many discrepancies in the pricing of various upgrades between the U.S. version of the game and the U.S. strategy guide" (Bethke, pg 60). As someone who works in the computer game industry, Bethke also found that the strategy guide writer for the game *Star Trek: Starfleet Commander - Orion Pirates* had a lot of difficulty getting information from the developers who were busy trying to make the final push to finishing the game (pg 60). By the time the developers got back to the guide writer, several chapters of the book had been removed from the final version to save on paper (Bethke, pg 60).

The most obvious drawback to printed strategy guides is the price. Many game players do not want to pay the 15 to 20 dollars that a printed guide typically costs. Also, printed guides are not as easily accessible as an online or in-game guide. The other big drawback of printed guides is that they break the player's immersion in the game. The player is forced to pause the game in order to read the correct section of the book.

One of the strengths of printed strategy guides is they generally have big graphics that easily show maps and treasure, as can be seen in the Banjo-Kazooie strategy guide's map of Click Clock Wood (Figure 101). As the researchers from Taiwan stated, printed guides can also be collectors' items for avid gamers (Sun et al, 2003, pg 3). Printed strategy guides have been used for all genres of games and across all platforms. This form of help is generally made for well established titles or highly anticipated game releases; often being printed before the game is actually released (Emrich, paragraph 7).



Figure 101: Banjo-Kazooie: Click Clock Wood Map

User Made Guides

Unlike printed strategy guides that are professionally made and printed, online guides are the complete opposite. User-made guides are generally created by “aggressive players of new games,” often compiled within days of the new game’s release (Sun et al, pg 3). Aareseth believes that, during the player’s strive to master almost any type of game, he or she will socialize to solve problems or find new solutions to puzzles (“Playing,” pg 3). He further says that “a complex game, such as Civilization, Deus Ex or GTA3 may be won in a matter of days or weeks, but due to the openness of the simulation and the collective ingenuity of players, the potential for new discoveries is endless” (“Playing,” pg 3). Online help also allows players to ask questions about their specific problems to a wide range of players. The posting of questions is good for the

player, because he or she faces less of a risk of spoiling the game. The questions cover a wide range of topics such as “I don’t know what to do next” or “I want to get extra treasure. How?” (Sun et al, pg 3). With the popularity of video sharing websites like YouTube, players also are able to upload videos onto the internet for other players to watch. This can be a helpful visual aid for players to find solutions to specific problems that are difficult to describe with text alone.

The problem with online guides is that anyone can post them. This can lead to a wide variety in the quality of content; some players will put a lot of time and effort into polishing their guide whereas others will try to write their guide from memory. The problem with video walkthroughs is the general lack of explanation as to what the video’s poster is doing and why they are doing it. Another problem is that this type of guide can frequently lead to spoilers and dissatisfaction. Aareseth noticed this during his own playing of *Morrowind*:

He was curious about finding strategy to defeat a boss, which caused him to look online. However, he learned that during all of his hours of playing the game, he had not started playing the intended “main quest.” Therefore, instead of quickly being able to defeat the boss, he found himself completely drawn out of the magic of the game with the overwhelming amount of new knowledge. After a week of playing, he suddenly stopped permanently. (Aareseth, “Playing,” pg 6).

The reason online guides are commonly used among gamers is that this form of help is free and can be easily accessed if the player has the internet. Also, the player can get specific help for his or her problems rather than only being able to read a printed page. Online guides are made for a variety of different genres, but are most common for adventure and RPG games as these genres typically have many side quests, hidden items, character skills, weapons, and bosses. For example, on a section on GameFAQs.com about the game *Kingdom Hearts*, an action/RPG released in 2002, there are 17 general FAQs, 33 in-depth FAQs, 4 FAQs translated into foreign languages, and 1 FAQ specifically for game secrets (Game FAQs). These online guides can sometimes include images like maps, but the quality, like in the walkthroughs themselves, can vary. Figure 102 shows a map for a campaign in *Warcraft 3* that is poorly made because the player cannot discern the terrain and buildings, while Figure

103 shows a map from *Fire Emblem* that is nicely detailed with enemies and treasure locations clearly drawn.



Figure 102: Warcraft III: Human Campaign - Mission 1 Map

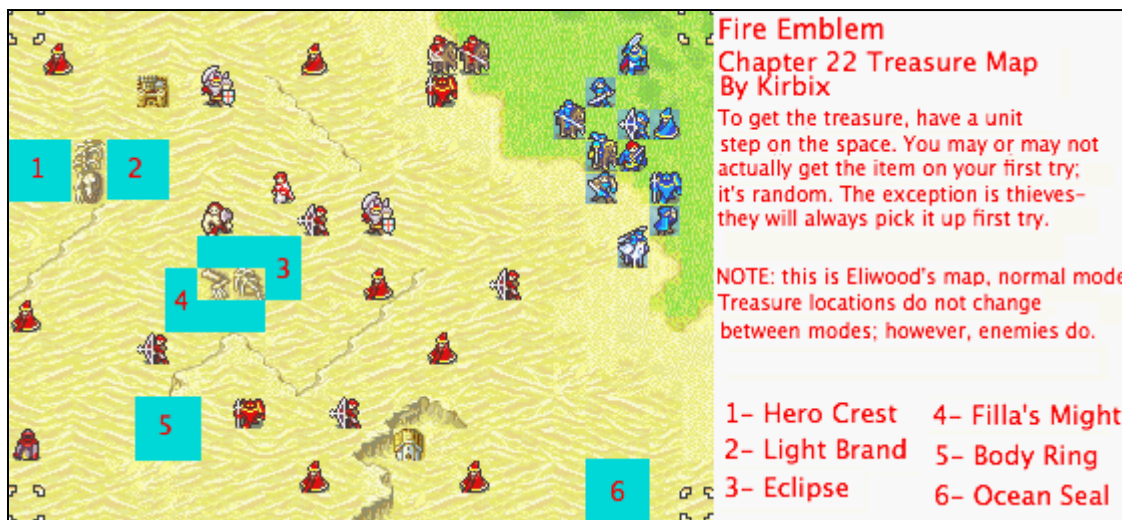


Figure 103: Fire Emblem: Chapter 22 Map

Appendix E: Coding Sheet

Play Through 1:

Player: _____ Observer: _____ Timer: _____

Game Title: _____ Publisher: _____

Game Genre: _____ Year Released: _____

Total Time Taken to Learn Game: _____ Total Types of Help Used: _____

Confidence that skills are learned (*1 – not confident, 5 – extremely confident*): _____

General	Yes	No
Is there in-game help?		
Is there a tutorial level ?		
Is help provided by a mentor character ?		
Is help provided by other characters ?		
Is help provided by an in-ear guide ?		
Is there a dialog tree help ?		
Is there any pop-up windows ?		
Is there a synopsis ?		
Are there UI elements (ex. minimap icons)?		
Can you skip the help?		
Does it teach you everything upfront? (vs. gradual)		
Does it acknowledge that it is a game (“Press A”)?		

Tutorial Level	Yes	No
Is there a defined level for learning?		
Are there multiple sub-levels?		
Does it acknowledge that it is help?		

Mentor Characters	Yes	No
Is there a mentor character?		

Other Character Help	Yes	No
Is help provided by other characters optional?		
Are tips given by NPCs in the game world?		
Is help provided in cutscenes with NPCs (goals, ex)?		

In-Ear Guides	Yes	No
Is there an in-ear guide to give help?		
Is it a mentor character?		
Is it a more experienced character?		

Dialog Tree Help	Yes	No
Is help provided through a dialog box the player can interact with?		
Is it conversation-based?		

Pop-Up Windows	Yes	No
Are there pop up tips?		
Is there a window based tutorial?		
Are they obtrusive to the gameplay?		

Synopsis	Yes	No
Is there a synopsis to tell you what has happened?		
Is there an in-game handbook with all learned moves?		

User Interface Help	Yes	No
Is there a mini-map or map?		
Are there icons/symbols that guide player?		

Notes:

Play Through 2

Player: _____ Observer: _____ Timer: _____

Total Time Taken to Learn Game: _____

Confidence that skills are learned (*1 – not confident, 5 – extremely confident*): _____

Notes:

Appendix F: Coding Sheet for *Baten Kaitos*

Play Through 1

Player: Joseph Alea Observer: ChanLing Beswick Timer: Sarah Gilke

Game Title: Baten Kaitos: Eternal Wings and Lost Ocean Publisher: Namco

Game Genre: Role Playing Game Year Released: 2004

Total Time Taken to Learn Game: 30 minutes Total Types of Help Used: 3

Confidence that skills are learned (*1 – not confident, 5 – extremely confident*): 1.5

General	Yes	No
Is there in-game help?	X	
Is there a tutorial level ?	X	
Is help provided by a mentor character ?		X
Is help provided by other characters ?	X	
Is help provided by an in-ear guide ?		X
Is there a dialog tree help ?	X	
Is there any pop-up windows ?		X
Is there a synopsis ?		X
Are there UI elements (ex. minimap icons)?		X
Can you skip the help?	X	
Does it teach you everything upfront? (vs. gradual)	X	
Does it acknowledge that it is a game (“Press A”)?		X

Tutorial Level	Yes	No
Is there a defined level for learning?	X	
Are there multiple sub-levels?	X	
Does it acknowledge that it is a tutorial?		X

Mentor Characters	Yes	No
--------------------------	------------	-----------

Is there a mentor character?		X
------------------------------	--	---

Other Character Help	Yes	No
Is help provided by other characters optional?	X	
Are tips given by NPCs in the game world?	X	
Is help provided in cutscenes with NPCs (goals, ex)?		X

In-Ear Guides	Yes	No
Is there an in-ear guide to give help?		X
Is it a mentor character?		X
Is it a more experienced character?		X

Dialog Tree Help	Yes	No
Is help provided through a dialog box the player can interact with?	X	
Is it conversation-based?	X	

Pop-Up Windows	Yes	No
Are there pop up tips?		X
Is there a window based tutorial?		X
Are they obtrusive to the gameplay?		X

Synopsis	Yes	No
Is there a synopsis to tell you what has happened?		X
Is there an in-game handbook with all learned moves?		X

User Interface Help	Yes	No
Is there a mini-map or map?		X
Are there icons/symbols that guide player?		X

Notes:

The player did not talk with the NPC's, therefore did not learn through the tutorial as the game designed. However, he believed that with more practice he would have been able to understand better.

Player had trouble targeting self vs. monsters.

Play Through 2: Baten Kaitos

Player: Forrest Hogeboom Observer: Joseph Alea Timer: Sarah Gilkey

Total Time Taken to Learn Game: 34 minutes

Confidence that skills are learned (*1 – not confident, 5 – extremely confident*): 2

Notes:

Player did not talk with NPCs; instead he proceeded into the second dungeon. Player received a game over in 13 minutes of play during battle when his character died. Player had trouble targeting self vs. monsters. However, the player was asked if he wanted advice on targeting, but either did not see it or did not want to choose it.

Appendix G: Coding Sheet for *Final Fantasy X*

Play Through 1

Player: Steve Esposito Observer: Joseph Alea Timer: Joseph Alea

Game Title: Final Fantasy X Publisher: Square (now Square Enix)

Game Genre: Role Playing Game Year Released: 2001

Total Time Taken to Learn Game: 30 minutes Total Types of Help Used: 6

Confidence that skills are learned (*1 – not confident, 5 – extremely confident*): 5

General	Yes	No
Is there in-game help?	X	
Is there a tutorial level ?	X	
Is help provided by a mentor character ?	X	
Is help provided by other characters ?	X	
Is help provided by an in-ear guide ?		X
Is there a dialog tree help ?		X
Is there any pop-up windows ?	X	
Is there a synopsis ?	X	
Are there UI elements (ex. minimap icons)?	X	
Can you skip the help?		X
Does it teach you everything upfront? (vs. gradual)		X
Does it acknowledge that it is a game (“Press A”)?	X	

Tutorial Level	Yes	No
Is there a defined level for learning?	X	
Are there multiple sub-levels?	X	
Does it acknowledge that it is a tutorial?		X

Mentor Characters	Yes	No
Is there a mentor character?	X	

Other Character Help	Yes	No
Is help provided by other characters optional?		X
Are tips given by NPCs in the game world?		X
Is help provided in cutscenes with NPCs (goals, ex)?	X	

In-Ear Guides	Yes	No
Is there an in-ear guide to give help?		X
Is it a mentor character?		X
Is it a more experienced character?		X

Dialog Tree Help	Yes	No
Is help provided through a dialog box the player can interact with?		X
Is it conversation-based?		X

Pop-Up Windows	Yes	No
Are there pop up tips?		X
Is there a window based tutorial?	X	
Are they obtrusive to the gameplay?	X	

Synopsis	Yes	No
Is there a synopsis to tell you what has happened?		X
Is there an in-game handbook with all learned moves?	X	

User Interface Help	Yes	No
Is there a mini-map or map?	X	
Are there icons/symbols that guide player?	X	

Notes:

In battle, the characters will sometimes give the player hints about how to fight effectively.

There is a *Help* icon in the main menu to teach the player moves. However, the player in our test did not use this feature.

Play Through 2

Player: ChanLing Beswick Observer: Joseph Alea Timer: Joseph Alea

Total Time Taken to Learn Game: 30 minutes

Confidence that skills are learned (*1 – not confident, 5 – extremely confident*): 4

Notes:

Player had difficulty figuring out the “sphere grid”, but was able to resolve the problem through testing.

Appendix H: Coding Sheet for *Paper Mario*

Play Through 1

Player: Sarah Gilkey Observer: ChanLing Beswick Timer: Joseph Alea

Game Title: Paper Mario: The Thousand-Year Door Publisher: Intelligent Systems

Game Genre: Role Playing Game Year Released: 2004

Total Time Taken to Learn Game: within 30 minutes Total Types of Help Used: 4

Confidence that skills are learned (1 – not confident, 5 – extremely confident): 5

General	Yes	No
Is there in-game help?	X	
Is there a tutorial level ?	X	
Is help provided by a mentor character ?	X	
Is help provided by other characters ?	X	
Is help provided by an in-ear guide ?		X
Is there a dialog tree help ?		X
Is there any pop-up windows ?	X	
Is there a synopsis ?		X
Are there UI elements (ex. minimap icons)?		X
Can you skip the help?		X*
Does it teach you everything upfront? (vs. gradual)	X	
Does it acknowledge that it is a game (“Press A”)?	X	

Tutorial Level	Yes	No
Is there a defined level for learning?	X	
Are there multiple sub-levels?		X
Does it acknowledge that it is a tutorial?	X	

Mentor Characters	Yes	No
Is there a mentor character?	X	

Other Character Help	Yes	No
Is help provided by other characters optional?	X	
Are tips given by NPCs in the game world?	X	
Is help provided in cutscenes with NPCs (goals, ex)?	X	

In-Ear Guides	Yes	No
Is there an in-ear guide to give help?		X
Is it a mentor character?		X
Is it a more experienced character?		X

Dialog Tree Help	Yes	No
Is help provided through a dialog box the player can interact with?		X
Is it conversation-based?		X

Pop-Up Windows	Yes	No
Are there pop up tips?	X	
Is there a window based tutorial?		X
Are they obtrusive to the gameplay?		X

Synopsis	Yes	No
Is there a synopsis to tell you what has happened?		X
Is there an in-game handbook with all learned moves?		X

User Interface Help	Yes	No
Is there a mini-map or map?		X
Are there icons/symbols that guide player?		X

Notes:

* There were a few instances where the mentor asked if the player wanted to skip a piece of tutorial

Play Through 2: Paper Mario

Player: Pat Nietupski Observer: Sarah Gilkey Timer: Joseph Alea

Total Time Taken to Learn Game: 30 minutes

Confidence that skills are learned (*1 – not confident, 5 – extremely confident*): 4

Notes:

The player believed that it was overall easy, but the game was slightly confusing and required trial and error.

Appendix I: Coding Sheet for *Rogue Galaxy*

Play Through 1

Player: ChanLing Beswick Observer: Sarah Gilkey Timer: Joseph Alea

Game Title: Rogue Galaxy Publisher: Level-5

Game Genre: Role Playing Game Year Released: 2007

Total Time Taken to Learn Game: 20 minutes Total Types of Help Used: 5

Confidence that skills are learned (1 – not confident, 5 – extremely confident): 5

General	Yes	No
Is there in-game help?	X	
Is there a tutorial level ?	X	
Is help provided by a mentor character ?		X
Is help provided by other characters ?	X	
Is help provided by an in-ear guide ?		X
Is there a dialog tree help ?		X
Is there any pop-up windows ?	X	
Is there a synopsis ?	X	
Are there UI elements (ex. minimap icons)?	X	
Can you skip the help?		X*
Does it teach you everything upfront? (vs. gradual)		X**
Does it acknowledge that it is a game (“Press A”)?	X	

Tutorial Level	Yes	No
Is there a defined level for learning?	X	
Are there multiple sub-levels?		X
Does it acknowledge that it is a tutorial?		X

Mentor Characters	Yes	No
Is there a mentor character?		X

Other Character Help	Yes	No
Is help provided by other characters optional?		X***
Are tips given by NPCs in the game world?		X
Is help provided in cutscenes with NPCs (goals, ex)?	X	

In-Ear Guides	Yes	No
Is there an in-ear guide to give help?		X
Is it a mentor character?		X
Is it a more experienced character?		X

Dialog Tree Help	Yes	No
Is help provided through a dialog box the player can interact with?		X
Is it conversation-based?		X

Pop-Up Windows	Yes	No
Are there pop up tips?	X	
Is there a window based tutorial?	X	
Are they obtrusive to the gameplay?	X	

Synopsis	Yes	No
Is there a synopsis to tell you what has happened?	X	
Is there an in-game handbook with all learned moves?		X

User Interface Help	Yes	No
Is there a mini-map or map?	X	
Are there icons/symbols that guide player?	X	

Notes:

- * The windows are forced on-screen, but the player can immediately close the window without reading it.
- ** The game broke up the different techniques in different battles. However, the battles were all back to back.
- *** The tips were audio clips from the player's character and the accompanying party member telling you where to go. It could be muted.

Play Through 2: Rogue Galaxy

Player: Steve Esposito Observer: Joseph Alea Timer: Sarah Gilkey

Total Time Taken to Learn Game: 25 minutes

Confidence that skills are learned (*1 – not confident, 5 – extremely confident*): 5

Notes:

Appendix J: Coding Sheet for *Star Ocean*

Play Through 1

Player: ChanLing Beswick Observer: Sarah Gilkey Timer: Joseph Alea

Game Title: Star Ocean Publisher: Square Enix

Game Genre: Role Playing Game Year Released: 2004

Total Time Taken to Learn Game: 30 minutes Total Types of Help Used: 3

Confidence that skills are learned (*1 – not confident, 5 – extremely confident*): 3

General	Yes	No
Is there in-game help?	X	
Is there a tutorial level ?		X
Is help provided by a mentor character ?		X
Is help provided by other characters ?	X	
Is help provided by an in-ear guide ?		X
Is there a dialog tree help ?	X	
Is there any pop-up windows ?		X
Is there a synopsis ?		X
Are there UI elements (ex. minimap icons)?	X	
Can you skip the help?	X	
Does it teach you everything upfront? (vs. gradual)	X	
Does it acknowledge that it is a game (“Press A”)?	X*	

Tutorial Level	Yes	No
Is there a defined level for learning?		X
Are there multiple sub-levels?		X
Does it acknowledge that it is help?		X

Mentor Characters	Yes	No
Is there a mentor character?		X

Other Character Help	Yes	No
Is help provided by other characters optional?		X
Are tips given by NPCs in the game world?		X
Is help provided in cutscenes with NPCs (goals, ex)?	X	

In-Ear Guides	Yes	No
Is there an in-ear guide to give help?		X
Is it a mentor character?		X
Is it a more experienced character?		X

Dialog Tree Help	Yes	No
Is help provided through a dialog box the player can interact with?	X	
Is it conversation-based?		X

Pop-Up Windows	Yes	No
Are there pop up tips?		X
Is there a window based tutorial?		X
Are they obtrusive to the gameplay?		X

Synopsis	Yes	No
Is there a synopsis to tell you what has happened?		X
Is there an in-game handbook with all learned moves?		X

User Interface Help	Yes	No
Is there a mini-map or map?	X	
Are there icons/symbols that guide player?	X**	

Notes:

* It was a “battle tutorial,” but presented itself in a virtual reality game, so it stayed immersive

**While there are icons and symbols that show the player entranced and exits, this game lacks an arrow that show direction for destination.

The player accidentally skipped the game’s intended tutorial. He has completed the first battle in the virtual reality game and believed that the game was asking if he would like to play again. However, the game was actually about to break down how to fight.

Game goals were very unclear; player spent a long time wandering around the map. There was also a large amount of places the player could explore that were not necessary to the progression of the game.

Play Through 2: Star Ocean

Player: Steve Esposito Observer: Joseph Alea Timer: Joseph Alea

Total Time Taken to Learn Game: 25
minutes

Confidence that skills are learned (*1 – not confident, 5 – extremely confident*):
3

Notes:

The player did complete the designed tutorial, but found that very difficult to progress.

Appendix K: Example Application of Weighted Total

We will provide a case example to explain how we applied our weighted total equation, discussed in 7.2 Genre Analysis of In-Game Help, to determine the effectiveness of help. To analyze whether giving the player the ability to skip the tutorial was effective or ineffective, we took used the following data:

Table 25: Example results for weighted total equation

Game Title	Average Skills Learned (ASL)	Ability to Skip (AS)	ASL x AS
Baten Kaitos	1.75	Yes (1)	1.75
Final Fantasy X	4.5	No (0)	0
Paper Mario	4.5	No (0)	0
Rogue Galaxy	5	No (0)	0
Star Ocean	3	Yes (1)	3
		Total	4.75

To begin the calculation, we converted our analysis for the ability to skip into either a 1 if the response was “Yes” or 0 if the response was “No.” After, we multiplied the average skills learned and the ability to skip for each game; we summed the results together. This step can be represented in the column “ASL x AS” in the above table, resulting in a sum of 4.75. Finally, we divided that number by the total number of “Yes” responses, or 2 in this example. This gives us the final result of 2.38 for the weighted total. We compared this number to the other weighted totals to determine the effective and ineffective forms of in-game help.

Appendix L: Results Summary

	Baten Kaitos		Final Fantasy X		Paper Mario		Rogue Galaxy		Star Ocean		Total	Weighted Totals
Skills Learned Average	1.75		4.5		4.5		5		3			
Playthrough 1	1.5		5		5		5		3			
Playthrough 2	2		4		4		5		3			
General	<i>Yes</i>	<i>No</i>	<i>Yes</i>	<i>No</i>	<i>Yes</i>	<i>No</i>	<i>Yes</i>	<i>No</i>	<i>Yes</i>	<i>No</i>		
Is there in-game help?	1		1		1		1		1		5	3.75
Is there a tutorial level?	1		1		1		1			1	4	3.94
Is help provided by a mentor character?		1	1		1			1		1	2	4.50
Is help provided by other characters?	1		1		1		1		1		5	3.75
Is help provided by an in-ear guide?		1		1		1		1		1	0	0.00
Is there a dialog tree tutorial?	1			1		1		1	1		2	2.38
Is there any pop-up windows?		1	1		1		1			1	3	4.67
Is there a synopsis?		1	1			1		1		1	2	4.75
Are there UI elements (ex. minimap icons)?		1	1			1		1		1	3	4.17
Can you skip the help?	1			1		1		1		1	2	2.38

Does it teach you everything upfront? (vs. gradual)	1			1		1			1		3	3.08
Does it acknowledge that it is a game ("Press A")?		1	1		1		1		1		4	4.25
Total Types of Help	3		6		4		5		3			

	Baten Kaitos		Final Fantast X		Paper Mario		Rogue Galaxy		Star Ocean		Total	Weighted Totals
Tutorial Level	<i>Yes</i>	<i>No</i>	<i>Yes</i>	<i>No</i>	<i>Yes</i>	<i>No</i>	<i>Yes</i>	<i>No</i>	<i>Yes</i>	<i>No</i>		
Is there a defined level for learning?	1		1		1		1			1	4	3.94
Are there multiple sub-levels?	1		1			1		1		1	2	3.13
Does it acknowledge that it is help?		1		1	1			1		1	1	4.50
Mentor Characters	<i>Yes</i>	<i>No</i>	<i>Yes</i>	<i>No</i>	<i>Yes</i>	<i>No</i>	<i>Yes</i>	<i>No</i>	<i>Yes</i>	<i>No</i>		
Is there a mentor character?		1	1		1			1		1	2	4.50
Other Character Help	<i>Yes</i>	<i>No</i>	<i>Yes</i>	<i>No</i>	<i>Yes</i>	<i>No</i>	<i>Yes</i>	<i>No</i>	<i>Yes</i>	<i>No</i>		
Is help provided by other characters optional?	1			1	1			1		1	2	3.13
Are tips given by NPCs in the game world?	1			1	1			1		1	2	3.13
Is help provided in cutscenes with NPCs (goals, ex)?		1	1		1		1		1		4	4.25

In-Ear Guides	Yes	No	Yes	No	Yes	No	Yes	No	Yes	No		
Is there an in-ear guide to give help?		1		1		1		1		1	0	0.00
Is it a mentor character?		1		1		1		1		1	0	0.00
Is it a more experienced character?		1		1		1		1		1	0	0.00

	Baten Kaitos		Final Fantast X		Paper Mario		Rogue Galaxy		Star Ocean		Total	Weighted Totals
Dialog Tree Tutorial	Yes	No	Yes	No	Yes	No	Yes	No	Yes	No		
Is help provided through a dialog box the player can interact with?	1			1		1		1	1		2	2.38
Is it conversation-based?	1			1		1		1		1	1	1.75

Pop-Up Window	Yes	No	Yes	No	Yes	No	Yes	No	Yes	No		
Are there pop up tips?		1		1	1		1			1	2	4.75
Is there a window based tutorial?		1	1			1		1		1	2	4.75
Are they obtrusive to the gameplay?		1	1			1		1		1	2	4.75

Synopsis	Yes	No	Yes	No	Yes	No	Yes	No	Yes	No		
Is there a synopsis to tell you what has happened?		1		1		1		1		1	1	5.00
Is there an in-game handbook with all learned moves?		1	1			1		1		1	1	4.50

	Baten Kaitos		Final Fantasy X		Paper Mario		Rogue Galaxy		Star Ocean		Total	Weighted Total
User Interface Help	<i>Yes</i>	<i>No</i>	<i>Yes</i>	<i>No</i>	<i>Yes</i>	<i>No</i>	<i>Yes</i>	<i>No</i>	<i>Yes</i>	<i>No</i>		
Is there a mini-map or map?		1	1			1	1		1		3	4.17
Are there icons/symbols that guide player?		1	1			1		1	1		3	4.17

Appendix M: Tutorial Script

Piscivian Mentor: General Crabbly

General Crabbly is an old, wise, grouchy crab. He is a scholar who is an expert on the art of war. He was once the commander of the Crab people; however, the crab's endangerment has led to their inability to participate in battle. Therefore, the turtle race replaced the crabs in the Piscivian Alliance's primary defense. General Crabbly is bitter about his race's lack of participation and upset that he is too old to actively participate in battle, but is willing to help train you before this epic battle commences.

Human Mentor: Computer – TACTIS

(Tactical advisory computer training interactive simulator)

The Tactical Advisory Computer Training Interactive Simulator (TACTIS) is the most advanced training simulator known to man. It possesses knowledge from every battle ever fought on Earth and can flawlessly make calculations about success in battle. TACTIS is unable to express emotion, believing that using math and logical deduction is the only way to solve problems. However, TACTIS has no data about the Piscivian Alliance and races that fight for it. TACTIS was chosen to teach you about battle.

SCRIPT

When reading the script:

- Text in black is dialog given by the mentor
- Text in parenthesis and blue is actions to be done in the game code
- Text in green is actions that need to be performed by the player to continue the script.

Action/Step	Human	Piscivian
Introduction	Salutations commander, I am the Tactical Advisory Computer Training Interactive System, known as TACTIS. I have been programmed to train you for battle.	Attention! I am General Crabbly. The Council of the Piscivian Alliance has charged me with your commander training. Unfortunately, I am too old to command our forces

	<p>Let us begin.</p> <p>Simulation loading. Loading complete. Simulation commencing.</p>	<p>anymore, so we are forced to put our trust in you.</p> <p>Anyway, just do what I say and we will win this war. Understand? Alright, let's get this over with.</p>
<p>One unit spawns – view it (using map movement)</p>	<p>(hide all HQ, units, and disable buildings) (enable scuba HQ)</p> <p>Welcome to the simulation world. If, at any point during this simulation, you do not remember what I have said, you may scroll back in this window.</p> <p>As the tactical commander, it is your responsibility to command troops. Currently, you have access to one Headquarters. This HQ is barracks for the scuba troop. The scuba troop is known for strong attack power, but can be defeated by its weak armor. You must understand your unit's strengths and weaknesses if you want more than a 65% chance of success.</p> <p>(scuba unit spawn)</p> <p>Please move to the HQ and select the scuba troop.</p>	<p>(hide all HQ, units, and disable buildings) (enable merman HQ)</p> <p>Here is a model of the battle field. Don't be fooled by the realism, everything is made out of sand. I am going to explain this role down to the very basics, but if you forget what I have said, refer to the scribe's notes.</p> <p>As the tactical commander, you give orders to all of our troops. For now, I will only give you access to the Merman race and their HQ. The Mermen are known for their balanced training in attack and defense, as well as their ability to heal their allies.</p> <p>(merman unit spawn)</p> <p>Please move to the HQ and select the merman.</p>
<p>Move unit anywhere (using move command)</p>	<p>Now that the scuba troop is selected, click on the open ground to move the unit. Notice that the cursor changes to a green bull's-eye and an arrow when the troop is able to move to a location.</p>	<p>With the merman awaiting orders, select a location on the ground to issue a move command. Notice that your cursor changes to a green bull's-eye and an arrow when the troop is able to move to a location.</p>
<p>Enemy approaches – defend HQ</p>	<p>(spawn and move 1 merman to HQ)</p>	<p>(spawn and move 1 scuba troop to HQ)</p>

(using attack command)	Alert! Attention commander, there is an enemy mermen approaching the scuba troop HQ. With the scuba troop selected, click on the enemy unit to attack.	Pay attention commander! There is an enemy scuba troop approaching the Mermen's HQ. With the merman selected, click on the enemy unit to issue an attack order.
Build guard tower (building guard tower)	<p>(enable only guard tower)</p> <p>I would suggest building a guard tower to give you an advantage in defense of your HQ from enemies. Select an open plot of land and choose the turret icon to begin construction.</p> <p>Notice that you can only build on one plot at any one time. Also, when the building is destroyed the benefits are lost. However, you are able to rebuild on the empty plot.</p> <p>I will alert you when the construction is complete.</p>	<p>(enable only guard tower)</p> <p>If this were my war to command, I would utilize buildings like the Jellyfish Tower to guard the HQ. To build, select an open plot of land and choose the Jellyfish Tower icon to begin construction.</p> <p>I hope you noticed that you can only build on one plot at any one time. If you ignore your building and let it get destroyed, the benefits to the HQ are lost, but you are able to rebuild on the empty plot.</p> <p>Since I will be observing your progress, I will message you when the construction is complete.</p>
One enemy approach & dies (see power of guard tower)	<p>Construction complete.</p> <p>(spawn and move 1 manta ray to HQ)</p> <p>Alert! An enemy manta ray approaches your HQ. Please allow me to direct the turret to eliminate the threat.</p>	<p>Your Jellyfish Tower is finished.</p> <p>(spawn and move 1 sub to HQ)</p> <p>Look alive, commander! An enemy sub is approaching your HQ. Watch the Jellyfish Tower eliminate the threat.</p>
Build farm (building farm)	<p>(enable only farm)</p> <p>Farms allow your HQ to train an additional unit for each troop deployment. Select an empty plot and</p>	<p>(enable only farm)</p> <p>You do know that farms enable a race's HQ to train an additional unit for each troop deployment, correct?</p>

	<p>build a farm.</p>	<p>To build a farm, select an empty plot and choose the farm icon.</p>
<p>Watch all units spawn & all other HQs appear (see power of farm)</p>	<p>Construction complete.</p> <p>(show ally HQ and spawn all units)</p> <p>You now have control of all three HQ. Notice that the scuba troop HQ has deployed an additional unit.</p> <p>The submarine unit is classified as fast moving, highly armored, and not durable. The bomb ship is classified as slow moving, slow rate of fire, strong attack power, and durable.</p> <p>Also, units will continue to be deployed at a constant rate during the game as long as HQ stands.</p>	<p>The farm is constructed, commander.</p> <p>(show ally HQ and spawn all units)</p> <p>You have been making fine progress so far, so I will give you control of all the alliance's HQ. I hope you noticed that the merman HQ has deployed an additional unit, thanks to the farm building.</p> <p>The manta ray is fast moving but cannot hold up under constant attackers. The turtle is slow moving and attacking, but is a very powerful attacker.</p> <p>Also, units will continue to be deployed at a constant rate during the game as long as HQ stands.</p>
<p>Build armory (building armory)</p>	<p>(enable only armory)</p> <p>To ensure your victory, you may require the utilization of the armory. The armory strengthens all of the currently deployed and training units. Select an available plot and construct the armory.</p>	<p>(enable only armory)</p> <p>When I was a commander, I made sure to build armories to strengthen all of the troops who are currently deployed and for all future deployments. Select an available plot and construct your armory.</p>
<p>See stat increases (see power of armory)</p>	<p>Construction complete.</p> <p>Now, go to the HQ where you built the armory and select a unit. See that, on top of the unit's icon, there is a star. This means that the unit has more power than</p>	<p>The armory is finished, commander.</p> <p>Now, go to the HQ where you built the armory and select a unit. See that, on top of the unit's icon, there is a star. This means that the</p>

	before.	unit has more power than before.
Choose unit to be FC (move to FC mode)	(FC spawn select appear) Now that you are familiar with the Tactical Commander role, it is time to train you in the Field Commander role. This role allows you to remotely control one of the units through telepathic technology. Select one of your allied units to be your field commander race.	(FC spawn select appear) Alright, commander, you seem to have a grasp on the duties of the Tactical Commander. Now, I will teach you how to fight on the battle ground as a Field Commander. For this position, you will need to utilize the Piscivian power of spirit voyage to take the body of a new race and fight with it. Select one of your allied units to be your field commander race.
Movement in world (move in FC)	Press the “t” key to enter your field commander body. The field commander is superior to the units in this faction. It is statistically improbable to succeed without utilizing your role as Field Commander. Familiarize yourself with movement in this environment.	Press the “t” key to enter your field commander body. When I was at my prime, I was the most feared field commander, far superior to any other troop. You cannot ignore your field commander duties if you want to win this war. Take a minute to get used to moving around in your new body, commander.
One enemy approaches FC – attack (attack in FC)	(spawn and move 1 turtle to FC) Alert! An enemy is in your proximity. You are capable of defeating this unit with minimal output required. Attack the unit by clicking on the enemy or clicking the available skill in the pane below.	(spawn/move 1 bomb ship to FC) There is an enemy bomb ship en route to your position, commander. You should have no trouble with this straggler, show no mercy. To attack, click on the enemy or click the available skill in the pane below.
Level up to 2 in FC (see new abilities)	Defeating enemies increases your experience and you to learn additional abilities. Your field commander	With each enemy killed your field commander gains battle experience, giving you access to new skills to help in the

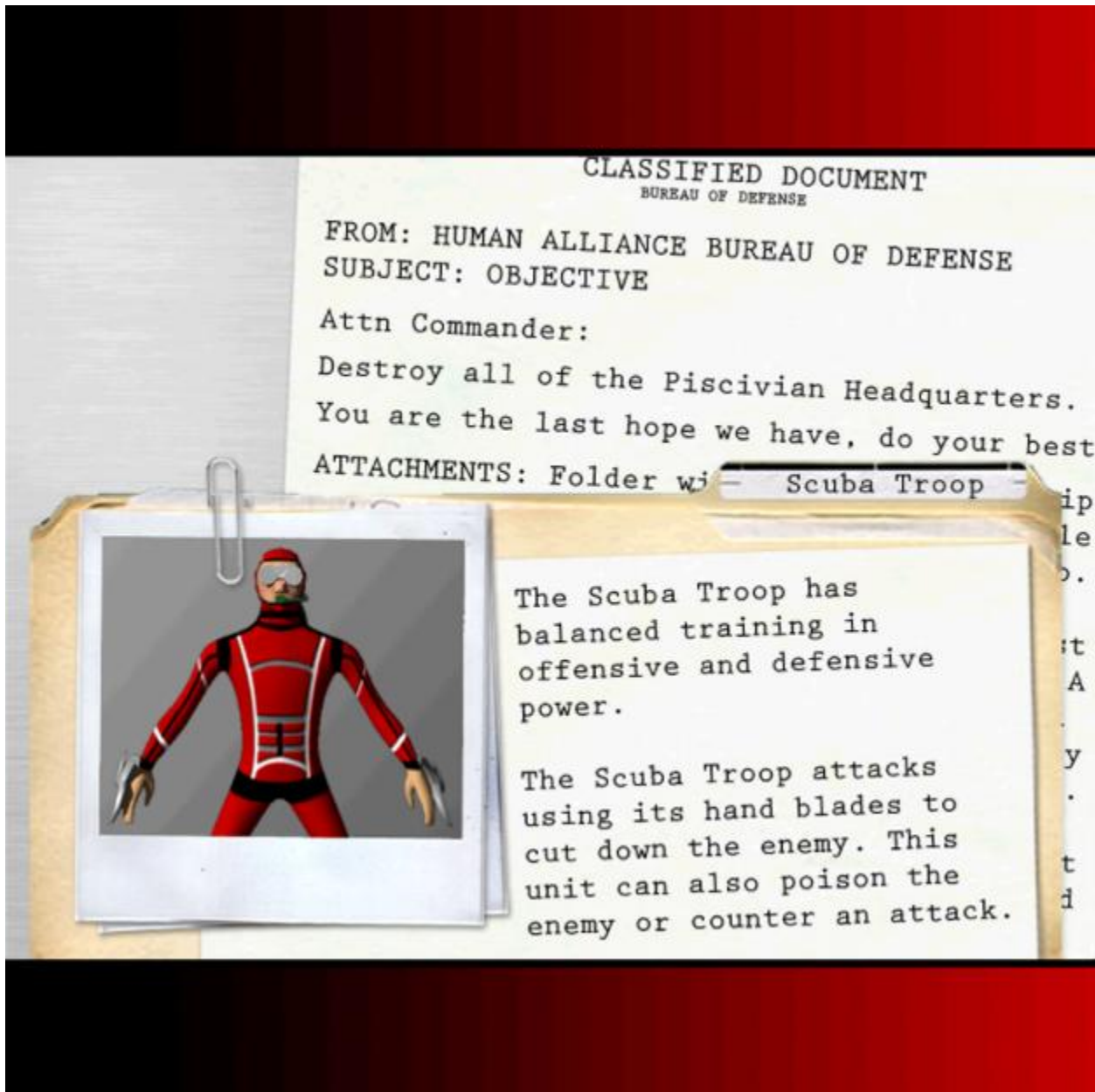
	gained the level 2 ability.	war.
One enemy approaches (ability use in FC)	(spawn and move 1 merman to FC) Alert! A second enemy is approaching your coordinates. Use your new ability to defeat the enemy.	(spawn and move 1 scuba to FC) Another enemy is approaching your location, commander. Use your new skill and eliminate the threat.
Level up to 3 in FC (see new abilities)	You have gained more experience from the previous battle and have learned your third and final ability.	You have gained another skill with your experience commander. Now, you will be unstoppable with this final ability.
Two enemies approach (ability use in FC)	(spawn and move 2 mantas to FC) Alert! There are two units moving to your position at a speed of 15 knots. Use a combination of your three abilities to destroy the enemy.	(spawn and move 2 subs to FC) Attention, commander, there are 2 enemy subs on a direct course for you. Use a combination of your three abilities to destroy the enemy.
Show how to switch FC (switching FCs)	You are able to change your Field Commander's race at any time by moving to any HQ. When changing FC races, you retain the rank you held previously. If you desire to switch races, do so now. If not, move back to your FC's home base.	You can switch your Field Commander's race by approaching any HQ. Your spirit retains the rank you held during a switch. If you want to change races before we assault the enemy HQ, do it now.
Switch out of FC (switching modes)	Now, switch back into your Tactical Commander role using "t". You can switch back into the Field Commander role again by clicking on the FC pane at any time. You can also view a real-time feed of your FC using the pane on the right side of the interface.	Now, switch back into your Tactical Commander role using "t". If you need to move back to the field, you can switch by clicking on the FC pane. You can also view a real-time feed of your FC using the pane on the right side of the interface.
Tell how FC dies	Be aware, when your Field Commander is killed in battle, you lose the rank you	Keep an eye on your FC. If he is killed, your spirit will lose a critical amount of

	have achieved. After a period of time to reset the equipment, you are able to select a new FC race.	energy. It will lose all of the experience and skills that were gained and will need time to rest before it can assume a new body.
Attack a base as TC (commanding units to attack)	Now, defeat an enemy HQ. <i>As the tactical commander, select your units and launch an attack on an enemy base.</i>	I think you are ready to move to the offensive. <i>As the tactical commander, command your units to launch an attack on one of the enemy's HQ.</i>
Attack a base as FC (attacking buildings as an FC)	Now, once you order your units, change to the FC role and destroy the second enemy HQ.	Now, let's try it as the FC. After you have ordered your troops to attack, <i>go onto the field and destroy your second HQ.</i>
Destroy final base (freeplay - player choice)	<i>There is one HQ left, destroy it.</i>	What are you waiting for? <i>Destroy our enemy's last outpost by any means necessary.</i>
Win!	Congratulations on completing the simulation. Our victory was inevitable. Simulation complete. Terminating Program.	Well, well. Maybe you do have a chance of winning this war after all. By your leave, commander, and good luck out there.

Appendix N: Implemented Loading Screens

Human Faction

Scuba Troop



Submarine

CLASSIFIED DOCUMENT
BUREAU OF DEFENSE

FROM: HUMAN ALLIANCE BUREAU OF DEFENSE
SUBJECT: OBJECTIVE

Attn Commander:

Destroy all of the Piscivian Headquarters.
You are the last hope we have, do your best.

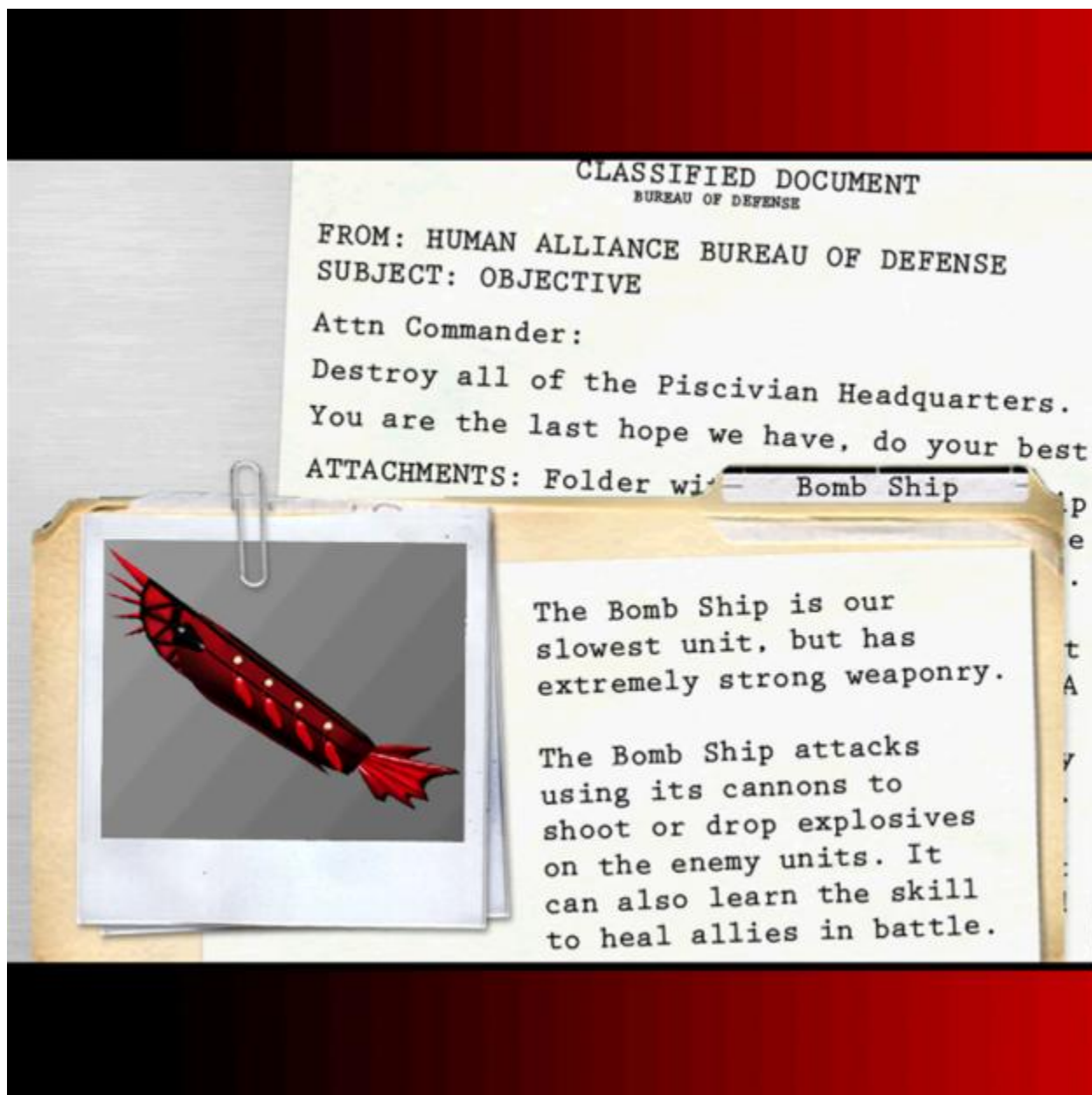
ATTACHMENTS: Folder with Submarine



The Submarine is our
fastest unit, capable of
speeds up to 40 knots.

The Submarine shoots its
torpedo to attack over
long distances. It also
utilizes a stealth skill
to sneak through enemy
lines.

Bomb Ship



CLASSIFIED DOCUMENT
BUREAU OF DEFENSE

FROM: HUMAN ALLIANCE BUREAU OF DEFENSE
SUBJECT: OBJECTIVE

Attn Commander:
Destroy all of the Piscivian Headquarters.
You are the last hope we have, do your best.

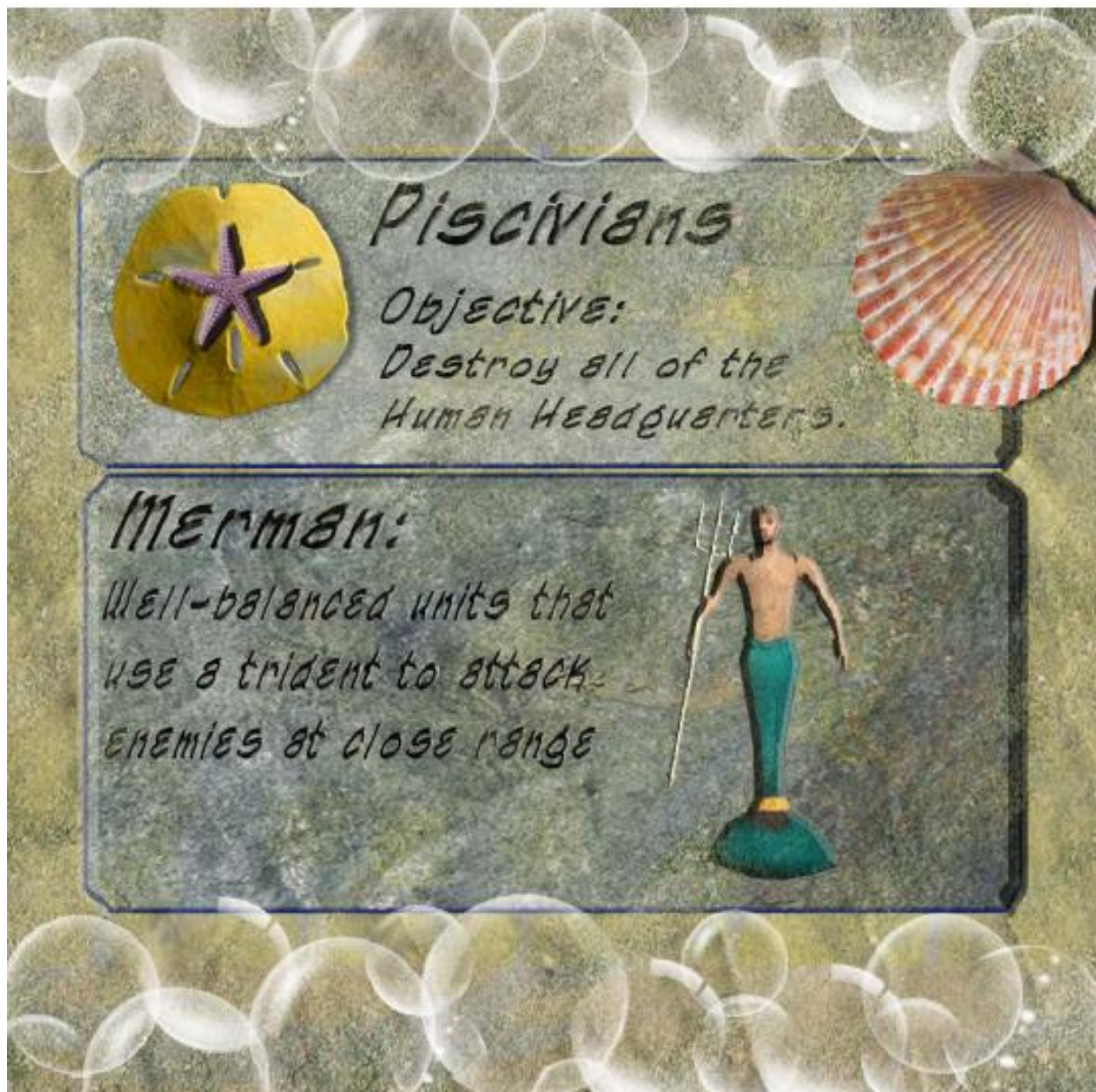
ATTACHMENTS: Folder with Bomb Ship



The Bomb Ship is our slowest unit, but has extremely strong weaponry.

The Bomb Ship attacks using its cannons to shoot or drop explosives on the enemy units. It can also learn the skill to heal allies in battle.

Merman



Manta Ray



Turtle



Appendix O: Menu Navigation Test Plan

The purpose of this test plan is to determine the functionality of menu navigation.

Step	Expected Result	Pass	Fail	Comment
1. Run BloodTide.exe	Game successfully loads and you are brought to the main menu.			
2. Verify layout of main menu.	Verify that the menu matches Figure 104: Menu - Main			



Figure 104: Menu - Main

3. Click "Quit"	Game successfully closes.			
-----------------	---------------------------	--	--	--

Reload game and return to main menu				
4. Click "Player Settings"	Verify that the resulting screen matches Figure 105: Menu - Player Settings			



Figure 105: Menu - Player Settings

5. Click "Cancel"	Verify that you are taken back to the main menu screen.			
6. Click "Player Settings" and click "OK"	Verify that you are taken back to the main menu screen.			
7. Click "Graphics Settings"	Verify that the resulting screen matches Figure 106: Menu - Graphics Settings.			

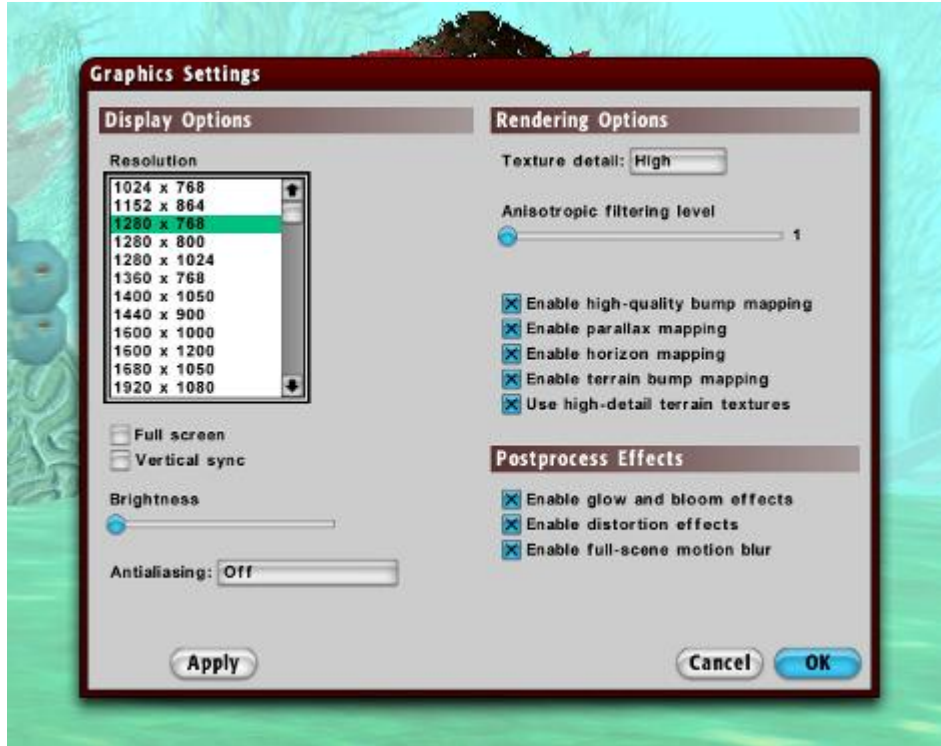


Figure 106: Menu - Graphics Settings

8. Click “Cancel”	Verify that you are taken back to the main menu screen.			
9. Click “Graphics Settings” and click “Apply”	Verify that you remain on the Graphics Settings screen.			
10. Click “OK”	Verify that you are taken back to the main menu screen.			
11. Click “Audio Settings”	Verify that the resulting screen matches Figure 107: Menu - Audio Settings			



Figure 107: Menu - Audio Settings

12. Click "Cancel"	Verify that you are taken back to the main menu screen.			
13. Click "Audio Settings" and click "OK"	Verify that you are taken back to the main menu screen.			
14. Click "Controls"	Verify that the resulting screen matches Figure 108: Menu - Controls			

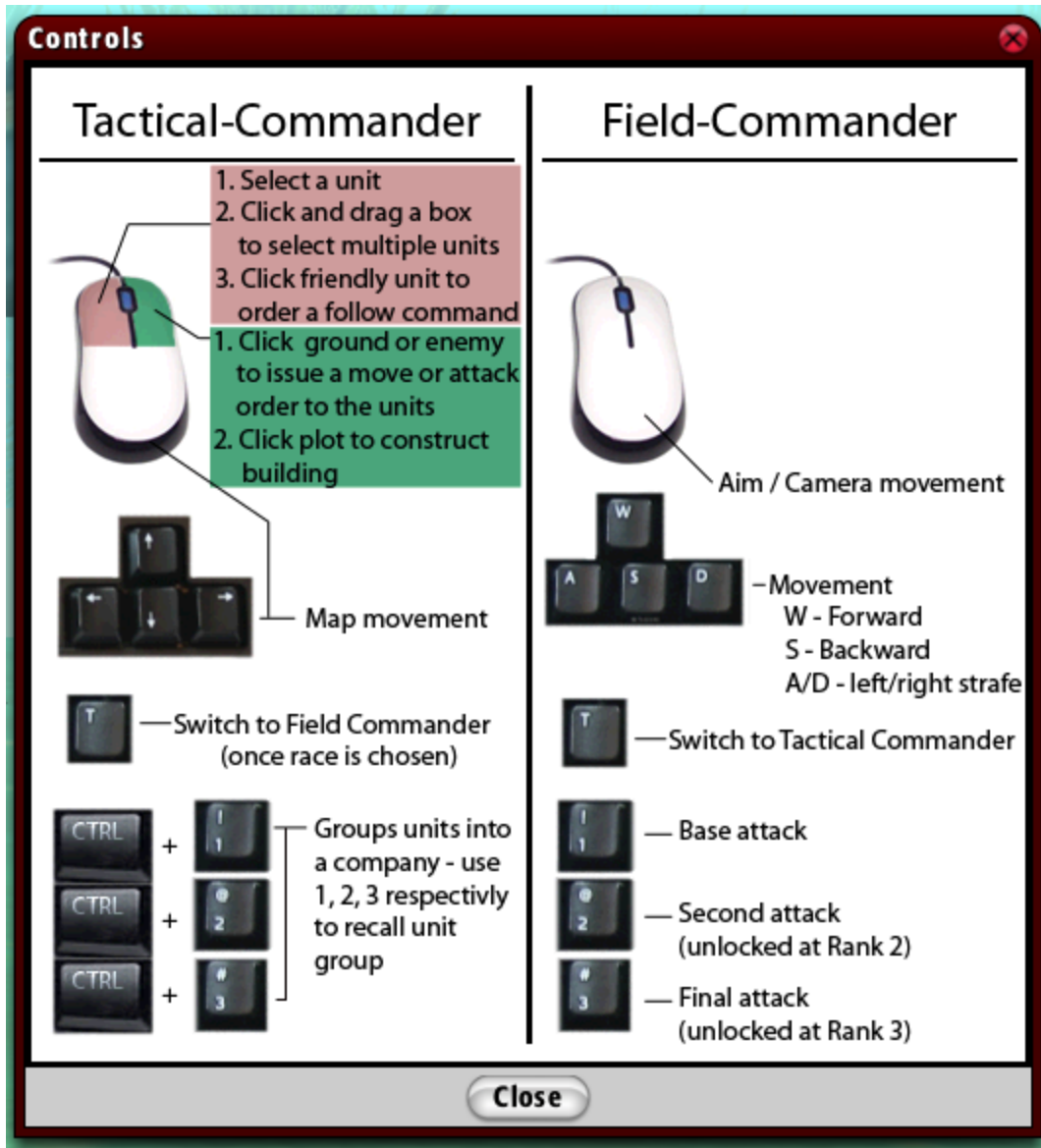


Figure 108: Menu - Controls

15. Click the "X"	Verify that you are taken back to the main menu screen.			
16. Click "Host Game"	Verify that the resulting screen matches Figure 109: Menu - Host Game			



Figure 109: Menu - Host Game

<p>17. Click “Cancel”</p>	<p>Verify that you are taken back to the main menu screen.</p>			
<p>18. Click “Host Game” and click “Host”</p>	<p>Verify that you are taken back to the faction selection screen matches Figure 110: Menu - Faction Selection</p>			



Figure 110: Menu - Faction Selection

<p>19. Click “Start” without selecting a faction</p>	<p>Verify that the start button is disabled and game does not start.</p>			
<p>20. Click “Cancel”</p>	<p>Verify that you are taken back to the main menu screen.</p>			
<p>21. Click “Host Game”, then “Host”, then select the Human faction, and click “Start”</p>	<p>Verify that the start button is enabled and the loading screen appears. Then, the game should start. Verify that you are the human faction.</p>			
<p>22. Press ESC when in the game.</p>	<p>Verify that the in-game menu matches Figure 111: Menu - In-Game Menu</p>			

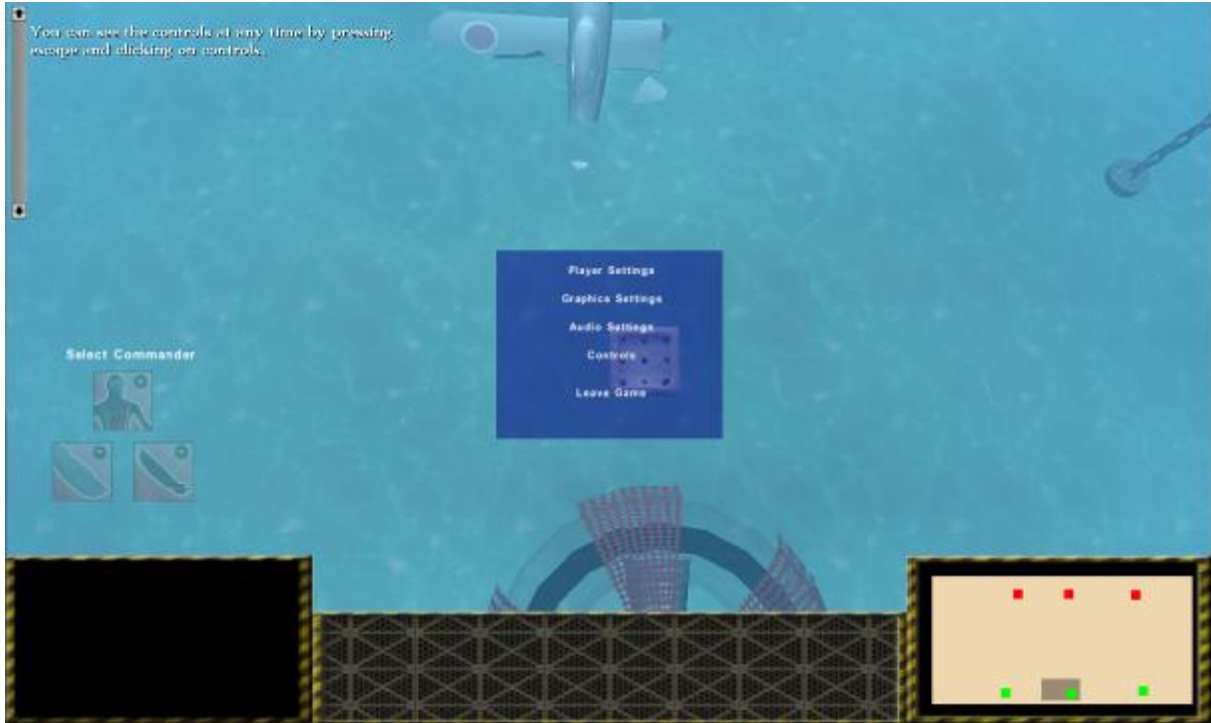


Figure 111: Menu - In-Game Menu

<p>23. Click “Exit Current Game”</p>	<p>Verify that the game ends and you are taken to the main menu screen.</p>			
<p>24. Click “Host Game”, then “Host”, then select the Piscivian faction and click “Start”</p>	<p>Verify that the start button is enabled and the loading screen appears. Then, the game should start. Verify that you are the Piscivian faction.</p>			
<p>25. Press ESC when in the game.</p>	<p>Verify that the menu appears.</p>			
<p>26. Click “Exit Current Game”</p>	<p>Verify that the game ends and you are taken to the main menu screen.</p>			
<p>27. Click “Join Game”</p>	<p>Verify that you are taken to the join game screen (Figure 112: Menu - Join Game). Verify that there are no available games to join.</p>			

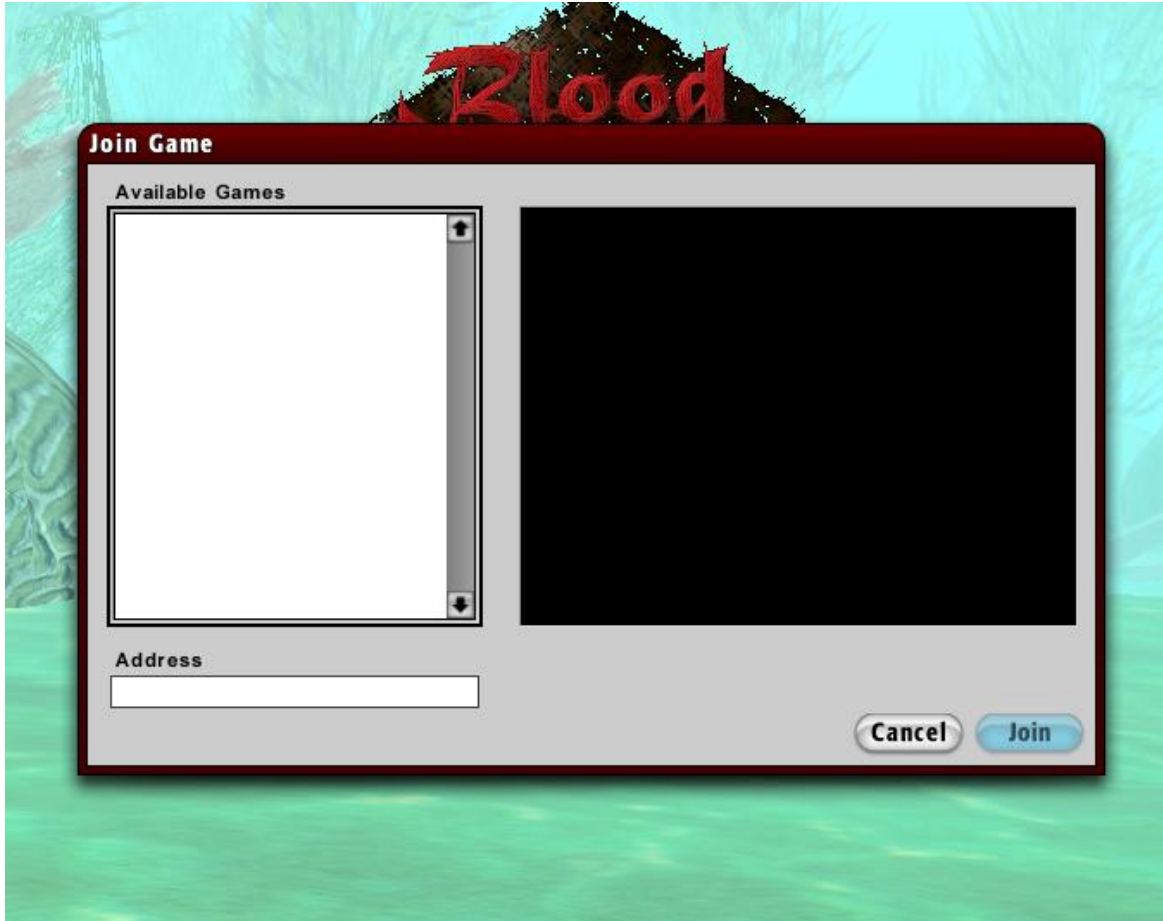


Figure 112: Menu - Join Game

<p>28. Click “Join” without selecting a game</p>	<p>Verify that the join button is disabled and game does not start.</p>			
<p>29. Click “Cancel”</p>	<p>Verify that you are taken back to the main menu screen.</p>			
<p>30. Click “Quit”</p>	<p>Verify that the game closes.</p>			

Appendix P: Input Functionality Test Plan

The purpose of this test plan is to determine the functionality of the game controls with the keyboard and mouse.

Step	Expected Result	Pass	Fail	Comment
1. Run BloodTide.exe	Game successfully loads. You should see the main menu screen.			
2. Click "Host Game"	Verify that you are taken to the world select screen			
3. Click "Host"	Verify you are taken to the faction select screen			
4. Select the Human faction and click "Start"	Verify that the game begins and you start as the human team			
5. Press the up arrow	Verify that the camera moves up			
6. Press the down arrow	Verify that the camera moves down			
7. Press the left arrow	Verify that the camera moves left			
8. Press the right arrow	Verify that the camera moves right			
9. Move the mouse	Verify that the camera moves with the mouse			
10. Click the left mouse button and drag to select a group of units	Verify that their icons appear in the bottom of the interface			
11. Click the right mouse button on the empty ground	Verify that the units move to the location but do not attack			
12. Click the right mouse button on an enemy unit	Verify that the units move to the location and attack any nearby enemies			
13. Press ENTER	Verify that the chat window appears			

14. Press ENTER again	Verify that the chat window disappears			
15. Press ESC	Verify that the menu appears			
16. Press ESC again	Verify that the menu disappears			
17. Select a field commander race	Verify that the commander selection triangle disappears			
18. Press 'T'	Verify that you switch to Field Commander mode			
19. Move the mouse	Verify that the camera view moves in conjunction with the mouse movement			
20. Press 'W'	Verify that the unit moves up			
21. Press 'S'	Verify that the unit moves down			
22. Press 'A'	Verify that the unit moves left			
23. Press 'D'	Verify that the unit moves right			
24. Move to an enemy unit	Verify that movement works as expected			
25. Press '1' at enemy unit	Verify that the rank 1 skill activates correctly and unit dies			
26. Press '2' at enemy unit	Verify that rank 2 skill activates correctly and unit dies			
27. Press '3' at enemy unit	Verify that rank 3 activate correctly and unit dies			
28. Press ENTER	Verify that the chat window opens			
29. Press ENTER again	Verify that the chat window closes			
30. Press ESC	Verify that the menu opens			

31. Press ESC again	Verify that the menu closes			
32. Press ESC, then select "Quit	Verify the menu opens, then game quits successfully			

Appendix Q: Tutorial Functionality Test Plan

The purpose of this test plan is to determine the functionality of the tutorial level. This test plan will verify the script, the dialog, and the correct behavior corresponding to the text.

Step	Expected Result	Pass	Fail	Comment
1. Run BloodTide.exe	Game successfully loads. You should see the main menu screen.			
2. Click “Host Game”, then select the “Tutorial – Human” map and click “Host”	Verify that you are taken to the faction selection screen (Figure 113: Menu - Faction Selection Screen)			



Figure 113: Menu - Faction Selection Screen

3. Select the human faction, then click “Start”	Verify that the game begins and you start as the human team			
4. Press the ESC key	Verify that the menu			

	opens			
5. Press ESC again	Verify that the menu disappears			
6. Wait until first dialog box opens	<p>Verify that the mentor is TACTIS and that his first statement is:</p> <p><i>Salutations commander, I am the Tactical Advisory Computer Training Interactive System, known as TACTIS. I have been programmed to train you for battle. Let us begin.</i></p> <p><i>Simulation loading. Loading complete. Simulation commencing.</i></p>			
7. Wait until world has loaded	Verify that all of the units, buildings, and HQs are disabled except for the scuba troop.			
8. Wait until second dialog to appear	<p>Verify that a one scuba troop has spawned.</p> <p>Verify that TACTIS says:</p> <p><i>Welcome to the simulation world. If, at any point during this simulation, you do not remember what I have said, you may scroll back in this window.</i></p> <p><i>As the tactical commander, it is your responsibility to command troops. Currently, you have access to one Headquarters. This HQ is barracks for the scuba troop. The scuba troop is known for</i></p>			

	<p><i>strong attack power, but can be defeated by its weak armor. You must understand your unit's strengths and weaknesses if you want more than a 65% chance of success.</i></p> <p><i>Please move to the HQ and select the scuba troop.</i></p>			
9. Select the scuba troop unit	<p>Verify that TACTIS says: <i>Now that the scuba troop is selected, click on the open ground to move the unit. Notice that the cursor changes to a green bull's-eye and an arrow when the troop is able to move to a location.</i></p>			
10. Move the scuba troop unit to any location	<p>Verify that a merman unit spawns and moves toward the scuba HQ.</p> <p>Verify that TACTIS says: <i>Alert! Attention commander, there is an enemy merman approaching the scuba troop HQ. With the scuba troop selected, click on the enemy unit to attack.</i></p>			
11. Order scuba troop to attack merman	<p>Verify that merman is able to be destroyed.</p> <p>Verify that the guard tower building is enabled.</p> <p>Verify that TACTIS says: <i>I would suggest building a guard tower to give you an advantage in defense of your HQ from enemies. Select an open</i></p>			

	<p><i>plot of land and choose the turret icon to begin construction.</i></p> <p><i>Notice that you can only build on one plot at any one time. Also, when the building is destroyed the benefits are lost. However, you are able to rebuild on the empty plot.</i></p> <p><i>I will alert you when the construction is complete.</i></p>			
12. Build a guard tower on any open plot	<p>Verify that the building is completed after a time.</p> <p>Verify that a manta ray spawns and moves to the scuba HQ.</p> <p>Verify that TACTIS says: <i>Construction complete.</i></p> <p><i>Alert! An enemy manta ray approaches your HQ. Please allow me to direct the turret to eliminate the threat.</i></p>			
13. Wait for manta ray to be destroyed	<p>Verify that the farm building is enabled.</p> <p>Verify that TACTIS says: <i>Farms allow your HQ to train an additional unit for each troop deployment. Select an empty plot and build a farm.</i></p>			
14. Build a farm on any open plot	<p>Verify that the bomb ship and sub bases are enabled.</p> <p>Verify that all units spawn, with one extra scuba troop.</p>			

	<p>Verify that the building is completed after a time.</p> <p>Verify that TACTIS says: <i>Construction complete.</i></p> <p><i>You now have control of all three HQ. Notice that the scuba troop HQ has deployed an additional unit.</i></p> <p><i>The submarine unit is classified as fast moving, highly armored, and not durable. The bomb ship is classified as slow moving, slow rate of fire, strong attack power, and durable.</i></p> <p><i>Also, units will continue to be deployed at a constant rate during the game as long as HQ stands.</i></p>			
15. Wait for next dialog to appear	<p>Verify that armory building is enabled.</p> <p>Verify that TACTIS says: <i>To ensure your victory, you may require the utilization of the armory. The armory strengthens all of the currently deployed and training units. Select an available plot and construct the armory.</i></p>			
16. Build an armory on any open plot	<p>Verify that the building is completed after a time.</p> <p>Verify that TACTIS says: <i>Construction complete.</i></p> <p><i>Now go to the HQ where you built the armory and</i></p>			

	<i>select a unit. See that, on top of the unit's icon, there is a star. This means that the unit has more power than before.</i>			
17. Wait for the commander selection triangle to appear.	Verify that TACTIS says: <i>Now that you are familiar with the Tactical Commander role, it is time to train you in the Field Commander role. This role allows you to remotely control one of the units through telepathic technology. Select one of your allied units to be your field commander race.</i>			
18. Select a commander race.	Verify that TACTIS says: <i>Press the 'T' key to enter your field commander body.</i> <i>The field commander is superior to the units in this faction. It is statistically improbable to succeed without utilizing your role as Field Commander.</i> <i>Familiarize yourself with movement in this environment.</i>			
19. Press 'T' to enter field commander mode, then move around world	Verify that a turtle unit spawns and moves towards your FC after a short time. Verify that TACTIS says: <i>Alert! An enemy is in your proximity. You are capable of defeating this unit with minimal output required. Attack the unit by clicking on the enemy or clicking the available skill in the pane below.</i>			
20. Defeat the turtle unit	Verify that the second rank ability is			

	<p>unlocked.</p> <p>Verify that TACTIS says: <i>Defeating enemies increases your experience and you to learn additional abilities. Your field commander gained the level 2 ability.</i></p>			
21. Wait for next dialog box to appear	<p>Verify that a merman unit spawns and moves towards your FC.</p> <p>Verify that TACTIS says: <i>Alert! A second enemy is approaching your coordinates. Use your new ability to defeat the enemy.</i></p>			
22. Defeat the merman unit	<p>Verify that the second rank ability is unlocked.</p> <p>Verify that TACTIS says: <i>You have gained more experience from the previous battle and have learned your third and final ability.</i></p>			
23. Wait for next dialog box to appear	<p>Verify that two manta rays unit spawns and moves towards your FC.</p> <p>Verify that TACTIS says: <i>Alert! There are two units moving to your position at a speed of 15 knots. Use a combination of your three abilities to destroy the enemies.</i></p>			
24. Defeat manta ray units	<p>Verify that TACTIS says: <i>You are able to change your Field Commander's race at any time by</i></p>			

	<i>moving to any HQ. When changing FC races, you retain the rank you held previously. If you desire to switch races, do so now. If not, move back to your FC's home base.</i>			
25. Switch field commander races	<p>Verify that the FC rank remains the same and the same number of skills is unlocked.</p> <p>Verify that TACTIS says: <i>Now, switch back into your Tactical Commander role using 'T'. You can switch back into the Field Commander role again by clicking on the FC pane at any time. You can also view a real-time feed of your FC using the pane on the right side of the interface.</i></p>			
26. Wait for next dialog to appear	<p>Verify that TACTIS says: <i>Be aware, when your Field Commander is killed in battle, you lose the rank you have achieved. After a period of time to reset the equipment, you are able to select a new FC race.</i></p>			
27. Wait for next dialog to appear	<p>Verify that TACTIS says: <i>Now, defeat an enemy HQ. As the tactical commander, select your units and launch an attack on an enemy base.</i></p>			
28. Select a group of your faction's units and order an attack on one of the enemy's HQs	<p>Verify that the HQ gets destroyed.</p> <p>Verify that TACTIS says: <i>Now, once you order your units, change to the FC role and destroy the second enemy HQ.</i></p>			

<p>29. Press 'T' to enter FC mode, then move to attack the second HQ</p>	<p>Verify that the HQ gets destroyed.</p> <p>Verify that TACTIS says: <i>There is one HQ left, destroy it.</i></p>			
<p>30. Order units to attack final HQ.</p>	<p>Verify that the HQ gets destroyed.</p> <p>Verify that TACTIS says: <i>Congratulations on completing the simulation. Our victory was inevitable.</i></p> <p><i>Simulation complete. Terminating Program.</i></p> <p>Verify that the game returns to the main menu once the dialog box closes.</p>			
<p>Begin a new tutorial and go to the faction selection screen</p>				
<p>31. Select the Piscivian faction, then click "Start"</p>	<p>Verify that the game begins and you start as the Piscivian team</p>			
<p>32. Wait until first dialog box opens</p>	<p>Verify that the mentor is Professor Crabbly and that his first statement is:</p> <p><i>Attention! I am General Crabbly. The Council of the Piscivian Alliance has charged me with your commander training. Unfortunately, I am too old to command our forces anymore, so we are forced to put our trust in you.</i></p> <p><i>Anyway, just do what I say and we will win this war. Understand? Alright, let's get this over with.</i></p>			
<p>33. Wait until world has loaded</p>	<p>Verify that all of the units, buildings, and HQs are disabled</p>			

	except for the merman.			
34. Wait until second dialog to appear	<p>Verify that a one merman has spawned.</p> <p>Verify that Crabbly says: <i>Here is a model of the battle field. Don't be fooled by the realism, everything is made out of sand. I am going to explain this role down to the very basics, but if you forget what I have said, refer to the scribe's notes.</i></p> <p><i>As the tactical commander, you give orders to all of our troops. For now, I will only give you access to the Merman race and their HQ. The Mermen are known for their balanced training in attack and defense, as well as their ability to heal their allies.</i></p> <p><i>Please move to the HQ and select the merman.</i></p>			
35. Select the merman unit	<p>Verify that Crabbly says: <i>With the merman awaiting orders, select a location on the ground to issue a move command. Notice that your cursor changes to a green bull's-eye and an arrow when the troop is able to move to a location.</i></p>			
36. Move the merman unit to any location	<p>Verify that a scuba troop unit spawns and moves toward the merman HQ.</p> <p>Verify that Crabbly says: <i>Pay attention commander! There is an</i></p>			

	<p><i>enemy scuba troop approaching the Mermen's HQ. With the merman selected, click on the enemy unit to issue an attack order.</i></p>			
<p>37. Order merman to attack scuba troop</p>	<p>Verify that scuba troop is able to be destroyed.</p> <p>Verify that the guard tower building is enabled.</p> <p>Verify that Crabbly says: <i>If this were my war to command, I would utilize buildings like the Jellyfish Tower to guard the HQ. To build, select an open plot of land and choose the Jellyfish Tower icon to begin construction.</i></p> <p><i>I hope you noticed that you can only build on one plot at any one time. If you ignore your building and let it get destroyed, the benefits to the HQ are lost, but you are able to rebuild on the empty plot.</i></p> <p><i>Since I will be observing your progress, I will message you when the construction is complete.</i></p>			
<p>38. Build a guard tower on any open plot</p>	<p>Verify that the building is completed after a time.</p> <p>Verify that a sub spawns and moves to the scuba HQ.</p> <p>Verify that Crabbly says: <i>Your Jellyfish Tower is finished.</i></p>			

	<p><i>Look alive, commander! An enemy sub is approaching your HQ. Watch the Jellyfish Tower eliminate the threat.</i></p>			
39. Wait for sub to be destroyed	<p>Verify that the farm building is enabled.</p> <p>Verify that Crabbly says: <i>You do know that farms enable a race's HQ to train an additional unit for each troop deployment, correct? To build a farm, select an empty plot and choose the farm icon.</i></p>			
40. Build a farm on any open plot	<p>Verify that the manta ray and turtle bases are enabled.</p> <p>Verify that all units spawn, with one extra merman.</p> <p>Verify that the building is completed after a time.</p> <p>Verify that Crabbly says: <i>The farm is constructed, commander.</i></p> <p><i>You have been making fine progress so far, so I will give you control of all the alliance's HQ. I hope you noticed that the merman HQ has deployed an additional unit, thanks to the farm building.</i></p> <p><i>The manta ray is fast moving but cannot hold up under constant attackers. The turtle is slow moving and</i></p>			

	<p><i>attacking, but is a very powerful attacker.</i></p> <p><i>Also, units will continue to be deployed at a constant rate during the game as long as HQ stands.</i></p>			
41. Wait for next dialog to appear	<p>Verify that armory building is enabled.</p> <p>Verify that Crabbly says: <i>When I was a commander, I made sure to build armories to strengthen all of the troops who are currently deployed and for all future deployments. Select an available plot and construct your armory.</i></p>			
42. Build an armory on any open plot	<p>Verify that the building is completed after a time.</p> <p>Verify that Crabbly says: <i>The armory is complete.</i></p> <p><i>Now go to the HQ where you built the armory and select a unit. See that, on top of the unit's icon, there is a star. This means that the unit has more power than before.</i></p>			
43. Wait for the commander selection triangle to appear.	<p>Verify that Crabbly says: <i>Alright, commander, you seem to have a grasp on the duties of the Tactical Commander. Now, I will teach you how to fight on the battle ground as a Field Commander. For this position, you will need to utilize the Piscivian power of spirit voyage to take the body of a new race and fight with it. Select one of</i></p>			

	<i>your allied units to be your field commander race.</i>			
44. Select a commander race.	<p>Verify that Crabbly says: <i>Press the ‘T’ key to enter your field commander body.</i></p> <p><i>When I was at my prime, I was the most feared field commander, far superior to any other troop. You cannot ignore your field commander duties if you want to win this war.</i></p> <p><i>Take a minute to get used to moving around in your new body, commander.</i></p>			
45. Press ‘T’ to enter field commander mode, then move around world	<p>Verify that a bomb ship unit spawns and moves towards your FC after a short time.</p> <p>Verify that Crabbly says: <i>There is an enemy bomb ship en route to your position, commander. You should have no trouble with this straggler, show no mercy. To attack, click on the enemy or click the available skill in the pane below.</i></p>			
46. Defeat the bomb ship unit	<p>Verify that the second rank ability is unlocked.</p> <p>Verify that Crabbly says: <i>With each enemy killed your field commander gains battle experience, giving you access to new skills to help in the war.</i></p>			
47. Wait for next dialog box to	Verify that a scuba troop unit spawns			

appear	<p>and moves towards your FC.</p> <p>Verify that Crabbly says: <i>Another enemy is approaching your location, commander. Use your new skill and eliminate the threat.</i></p>			
48. Defeat the scuba troop unit	<p>Verify that the second rank ability is unlocked.</p> <p>Verify that Crabbly says: <i>You have gained another skill with your experience commander. Now, you will be unstoppable with this final ability.</i></p>			
49. Wait for next dialog box to appear	<p>Verify that two subs unit spawns and moves towards your FC.</p> <p>Verify that Crabbly says: <i>Attention, commander, there are 2 enemy subs on a direct course for you. Use a combination of your three abilities to destroy the enemy.</i></p>			
50. Defeat subs units	<p>Verify that Crabbly says: <i>You can switch your Field Commander's race by approaching any HQ. Your spirit retains the rank you held during a switch. If you want to change races before we assault the enemy HQ, do it now. If not, move back to your FC's home base.</i></p>			
51. Switch field commander races	<p>Verify that the FC rank remains the same and the same</p>			

	<p>number of skills is unlocked.</p> <p>Verify that Crabbly says: <i>Now, switch back into your Tactical Commander role using 'T'. You can switch back into the Field Commander role again by clicking on the FC pane at any time. You can also view a real-time feed of your FC using the pane on the right side of the interface.</i></p>			
52. Wait for next dialog to appear	<p>Verify that Crabbly says: <i>Keep an eye on your FC. If he is killed, your spirit will lose a critical amount of energy. It will lose all of the experience and skills that were gained and will need time to rest before it can assume a new body.</i></p>			
53. Wait for next dialog to appear	<p>Verify that Crabbly says: <i>I think you are ready to move to the offensive. As the tactical commander, command your units to launch an attack on one of the enemy's HQ.</i></p>			
54. Select a group of your faction's units and order an attack on one of the enemy's HQs	<p>Verify that the HQ gets destroyed.</p> <p>Verify that Crabbly says: <i>Now, let's try it as the FC. After you have ordered your troops to attack, go onto the field and destroy your second HQ.</i></p>			
55. Press 'T' to enter FC mode, then move to attack the second HQ	<p>Verify that the HQ gets destroyed.</p> <p>Verify that Crabbly says:</p>			

	<i>What are you waiting for? Destroy our enemy's last outpost by any means necessary.</i>			
56. Order units to attack final HQ.	<p>Verify that the HQ gets destroyed.</p> <p>Verify that Crabbly says: <i>Well, well. Maybe you do have a chance of winning this war after all.</i></p> <p><i>By your leave, commander, and good luck out there.</i></p> <p>Verify that the game returns to the main menu once the dialog box closes.</p>			
57. Click "Quit"	Verify that game closes successfully.			

Appendix R: RTS Mode Unit Functionality Test Plan

The purpose of this test plan is to determine the functionality of the units in the RTS mode of the game. This test plan will verify the important aspects of RTS mode including unit spawning, attacking, death, movement, and the win/lose condition.

Step	Expected Result	Pass	Fail	Comment
1. Run BloodTide.exe	Game successfully loads. You should see the main menu screen.			
2. Click “Host Game” and click “Host”	Verify that you are taken to the faction selection screen			



Figure 114: Menu - Faction Selection

3. Select the Human faction, and click “Start”	Verify the loading screen appears. Then, the game should start. Verify that you are playing as the Human			
--	--	--	--	--

	faction.			
4. Move so the screen is positioned above the Scuba Troop HQ (Figure 115: Scuba Troop HQ - no unit)	Verify that scuba troops are spawning from the Scuba Troop HQ (Figure 116: Scuba Troop HQ - with unit)			

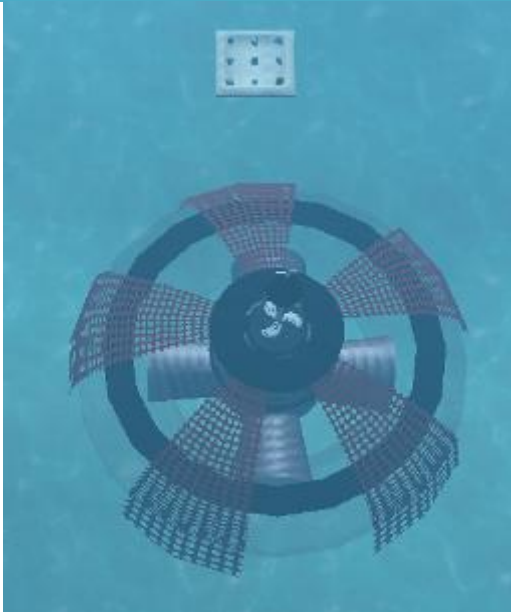


Figure 115: Scuba Troop HQ - no unit

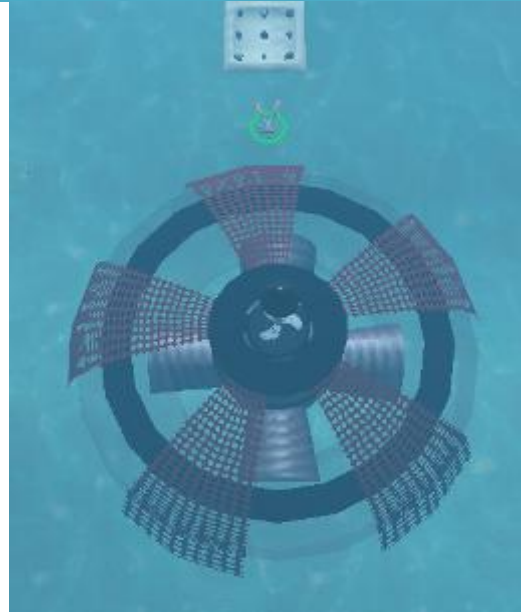


Figure 116: Scuba Troop HQ - with unit

5. Move so the screen is positioned above the Submarine HQ (Figure 117: Submarine HQ - no unit)	Verify that submarines are spawning from the Submarine HQ (Figure 118: Submarine HQ - with unit).			
---	---	--	--	--

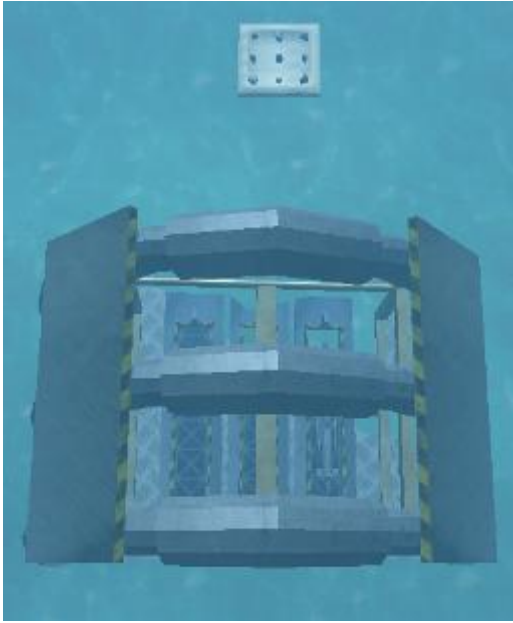


Figure 117: Submarine HQ - no unit



Figure 118: Submarine HQ - with unit

<p>6. Move so the screen is positioned above the Bomb Ship HQ (Figure 119: Bomb Ship HQ - no unit)</p>	<p>Verify that bomb ships are spawning from the Bomb Ship HQ (Figure 120: Bomb Ship HQ - with unit)</p>			
--	---	--	--	--

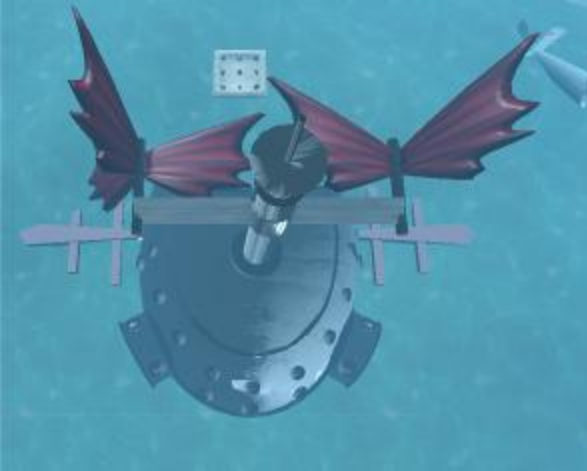


Figure 119: Bomb Ship HQ - no unit



Figure 120: Bomb Ship HQ - with unit

<p>7. Move the screen back to the Scuba Troop HQ.</p> <p>Left click and hold to drag select over all the scuba troop units.</p>	<p>You should now have all the scuba troops selected. Verify that the number of scuba troops you have selected corresponds to the number of scuba troop icons now visible in the UI.</p>			
<p>8. Click anywhere on the map.</p>	<p>Verify that your selected scuba troops move to that designated location.</p> <p>Your scuba troops should still be selected.</p>			
<p>9. Move the screen so that a Piscivian unit (merman, turtle, or manta ray) is visible.</p> <p>Left click on an enemy unit.</p>	<p>Verify that your selected scuba troops move to the enemy target and attack.</p> <p>Verify that the scuba troops attack close range using their hand propellers.</p> <p>Verify that the Piscivian unit dies.</p>			
<p>10. Continue fighting until one of your</p>	<p>Verify that the scuba troop's health bar</p>			

scuba troops die.	can decrease and that the unit can die.			
11. Click on any of the Piscivian headquarters.	Verify that the scuba troops go attack the headquarters.			
12. Continue attacking a headquarters until it is destroyed.	Verify that the scuba troop is capable of destroying a HQ.			
13. Right click on the mouse.	Verify that you have deselected your scuba troops.			
14. Move the screen back to the Submarine HQ. Left Click and hold to drag select over some submarine units.	You should now have some submarines selected. Verify that the number of submarines you have selected corresponds to the number of submarine icons now visible in the UI.			
15. Right click anywhere on the map.	Verify that your selected submarines move to that designated location. Your submarines should still be selected.			
16. Move the screen so that a Piscivian unit (merman, turtle, or manta ray) is visible. Left click on an enemy unit.	Verify that your selected submarines move to the enemy target and attack. Verify that the submarines attack from afar using their torpedo. Verify that the Piscivian unit dies.			
17. Continue fighting until one of your submarines die.	Verify that the submarine's health bar can decrease and that the unit can die.			

18. Click on any of the remaining two Piscivian headquarters.	Verify that the submarines go attack the headquarters.			
19. Continue attacking a headquarters until it is destroyed.	Verify that the submarine is capable of destroying a HQ.			
20. Right click on a piece of unoccupied ground.	Verify that you have deselected your submarines.			
21. Move the screen back to the Bomb Ship HQ. Left Click and hold to drag select over some bomb ship units.	You should now have some bomb ships selected. Verify that the number of bomb ships you have selected corresponds to the number of bomb ship icons now visible in the UI.			
22. Right click anywhere on the map.	Verify that your selected bomb ships move to that designated location. Your bomb ships should still be selected.			
23. Move the screen so that a Piscivian unit (merman, turtle, or manta ray) is visible. Left click on an enemy unit.	Verify that your selected bomb ships move to the enemy target and attack. Verify that the bomb ships attack from afar using their cannon ball. Verify that the Piscivian unit dies.			
24. Continue fighting until one of your bomb ships die.	Verify that the bomb ship's health bar can decrease and that the unit can die.			

25. Click on the last remaining Piscivian headquarters.	Verify that the bomb ships go attack the headquarters.			
26. Continue attacking a headquarters until it is destroyed.	Verify that the bomb ship is capable of destroying a HQ. Verify that the winning screen loads for the Humans.			
27. Click Okay	Verify that you are brought back to the main menu.			
Begin a new game and go to the faction selection screen				
28. Select the Piscivian faction, and click "Start"	Verify the loading screen appears. Then, the game should start. Verify that you are playing as the Piscivian faction.			
29. Move so the screen is positioned above the Merman HQ (Figure 121: Merman HQ - no unit)	Verify that mermen are spawning from the Mermen HQ (Figure 122: Merman HQ - with unit)			



Figure 121: Merman HQ - no unit



Figure 122: Merman HQ - with unit

<p>30. Move so the screen is positioned above the Manta Ray HQ (Figure 123: Manta Ray HQ - no unit)</p>	<p>Verify that manta rays are spawning from the Manta Ray HQ (Figure 124: Manta Ray HQ - with unit)</p>			
---	---	--	--	--



Figure 123: Manta Ray HQ - no unit



Figure 124: Manta Ray HQ - with unit

<p>31. Move so the screen is positioned above the Turtle HQ (Figure 125: Turtle HQ - no unit)</p>	<p>Verify that turtles are spawning from the Turtle HQ (Figure 126: Turtle HQ - with unit)</p>			
---	--	--	--	--



Figure 125: Turtle HQ - no unit



Figure 126: Turtle HQ - with unit

<p>32. Move the screen back to the Mermen HQ.</p> <p>Left click and hold to drag select over all the mermen units.</p>	<p>You should now have all the mermen selected. Verify that the number of mermen you have selected corresponds to the number of mermen icons now visible in the UI.</p>			
<p>33. Right click anywhere on the map.</p>	<p>Verify that your selected mermen move to that designated location.</p> <p>Your mermen should still be selected.</p>			
<p>34. Move the screen so that a Human unit (scuba troop, submarine, or bomb ship) is visible.</p>	<p>Verify that your selected mermen move to the enemy target and attack.</p> <p>Verify that the mermen attack close</p>			

Left click on an enemy unit.	range using their trident. Verify that the Human unit dies.			
35. Continue fighting until one of your mermen die.	Verify that the merman's health bar can decrease and that the unit can die.			
36. Click on any of the Human headquarters.	Verify that the mermen go attack the headquarters.			
37. Continue attacking a headquarters until it is destroyed.	Verify that the mermen are capable of destroying a HQ.			
38. Right click on the mouse.	Verify that you have deselected your mermen.			
39. Move the screen back to the Manta Ray HQ. Left Click and hold to drag select over some manta ray units.	You should now have some manta rays selected. Verify that the number of manta rays you have selected corresponds to the number of manta ray icons now visible in the UI.			
40. Right click anywhere on the map.	Verify that your selected manta rays move to that designated location. Your manta rays should still be selected.			
41. Move the screen so that a Human unit (scuba troop, submarine, or bomb ship) is visible. Left click on an enemy unit.	Verify that your selected manta rays move to the enemy target and attack. Verify that the manta rays attack close range using their tail whip.			

	Verify that the Human unit dies.			
42. Continue fighting until one of your manta rays die.	Verify that the manta ray's health bar can decrease and that the unit can die.			
43. Click on any of the remaining two Human headquarters.	Verify that the manta rays go attack the headquarters.			
44. Continue attacking a headquarters until it is destroyed.	Verify that the manta ray is capable of destroying a HQ.			
45. Right click on an empty piece of land.	Verify that you have deselected the manta rays.			
46. Move the screen back to the Turtle HQ. Left click and hold to drag select over some turtles units.	You should now have some turtles selected. Verify that the number of turtles you have selected corresponds to the number of turtle icons now visible in the UI.			
47. Right click anywhere on the map.	Verify that your selected turtles move to that designated location. Your turtles should still be selected.			
48. Move the screen so that a Human unit (scuba troop, submarine, or bomb ship) is visible. Left Click on a unit.	Verify that your selected turtles move to the enemy target and attack. Verify that the turtles attack from afar using their wave cannon. Verify that the			

	Piscivian unit dies.			
49. Continue fighting until one of your turtles die.	Verify that the turtle's health bar can decrease and that the unit can die.			
50. Click on the final Human headquarters.	Verify that the turtles go attack the headquarters.			
51. Continue attacking a headquarters until it is destroyed.	Verify that the turtle is capable of destroying a HQ. Verify that the winning screen loads for the Piscivians.			
52. Click Okay	Verify that you are brought back to the main menu.			

Appendix S: RTS Mode Building Functionality Test Plan

The purpose of this test plan is to determine the functionality of the buildings in the RTS mode of the game. This test plan will verify the important aspects of RTS mode including building timer and proper functioning of the buildings. You will be required to use two computers during this test plan.

Step	Expected Result	Pass	Fail	Comment
1. Run BloodTide.exe	Game successfully loads. You should see the main menu screen.			
2. Click “Host Game” and click “Host”	Verify that you are taken to the faction selection screen (Figure 127: Menu - Faction Selection)			



Figure 127: Menu - Faction Selection

3. Select human faction. Then, load the game on a second	Verify that the “loading screen” appears with game objective and tip for			
--	--	--	--	--

computer, join your game and select Piscivian faction. Click start	the appropriate faction.			
Switch to the Human faction once the game loads				
4. Middle mouse click on one of the plots around the Scuba Troop HQ and build a turret	Verify that the building begins construction			
5. Select a new plot around the Scuba Troop HQ and try to build a turret, farm, and armory	Verify that you cannot construct a new building			
6. Select a new plot around the Bomb Ship HQ and try to build a turret, farm, and armory	Verify that you cannot construct a new building			
7. Select a new plot around the Sub HQ and try to build a turret, farm, and armory	Verify that you cannot construct a new building			
8. Wait for the turret to finish construction. Then, select a new plot and build a second turret around the Scuba Troop HQ. When the building starts construction, time how long the construction takes to complete.	Verify that the turret takes 45 seconds to build.			
9. Select one of the plots around the Bomb Ship HQ and build a turret	Verify that the building begins construction			
10. Select a new plot around the Bomb	Verify that you cannot construct a			

Ship HQ and try to build a turret, farm, and armory	new building			
11. Select a new plot around the Scuba Troop HQ and try to build a turret, farm, and armory	Verify that you cannot construct a new building			
12. Select a new plot around the Sub HQ and try to build a turret, farm, and armory	Verify that you cannot construct a new building			
13. Wait for the turret to finish construction.	Verify that the building is completed successfully.			
14. Select one of the plots around the Sub HQ and build a turret	Verify that the building begins construction			
15. Select a new plot around the Sub HQ and try to build a turret, farm, and armory	Verify that you cannot construct a new building			
16. Select a new plot around the Scuba Troop HQ and try to build a turret, farm, and armory	Verify that you cannot construct a new building			
17. Select a new plot around the Bomb Ship HQ and try to build a turret, farm, and armory	Verify that you cannot construct a new building			
Switch to the Piscivian faction				
18. Move a group of units to any turret building	Verify that the turret attacks and kills the units			
Switch to the Human faction				
19. Select one of the plots around the Bomb Ship HQ and build a farm	Verify that the building begins construction			
20. Select a new plot	Verify that you			

around the Bomb Ship HQ and try to build a turret, farm, and armory	cannot construct a new building			
21. Select a new plot around the Scuba Troop HQ and try to build a turret, farm, and armory	Verify that you cannot construct a new building			
22. Select a new plot around the Sub HQ and try to build a turret, farm, and armory	Verify that you cannot construct a new building			
23. Wait for the farm to finish construction. Then, select a new plot and build a second farm around the Bomb Ship HQ. When the building starts construction, time how long the construction takes to complete.	Verify that the farm takes 45 seconds to build.			
24. Select one of the plots around the Scuba Troop HQ and build a farm	Verify that the building begins construction			
25. Select a new plot around the Scuba Troop HQ and try to build a turret, farm, and armory	Verify that you cannot construct a new building			
26. Select a new plot around the Bomb Ship HQ and try to build a turret, farm, and armory	Verify that you cannot construct a new building			
27. Select a new plot around the Sub HQ and try to build a turret,	Verify that you cannot construct a new building			

farm, and armory				
28. Wait for the farm to finish construction.	Verify that the building is completed successfully.			
29. Select one of the plots around the Sub HQ and build a farm	Verify that the building begins construction			
30. Select a new plot around the Sub HQ and try to build a turret, farm, and armory	Verify that you cannot construct a new building			
31. Select a new plot around the Scuba Troop HQ and try to build a turret, farm, and armory	Verify that you cannot construct a new building			
32. Select a new plot around the Bomb Ship HQ and try to build a turret, farm, and armory	Verify that you cannot construct a new building			
33. Wait until units spawn	Verify that three units appear around the Bomb Ship HQ and two units appear around the Scuba Troop and Sub HQ			
34. Select one of the plots around the Sub HQ and build a armory	Verify that the building begins construction			
35. Select a new plot around Sub HQ and try to build a turret, farm, and armory	Verify that you cannot construct a new building			
36. Select a new plot around the Scuba Troop HQ and try to build a turret, farm, and armory	Verify that you cannot construct a new building			
37. Select a new plot around the Bomb	Verify that you cannot construct a			

Ship HQ and try to build a turret, farm, and armory	new building			
38. Wait for the armory to finish construction. Then, select a new plot and build a second farm around the Sub HQ. When the building starts construction, time how long the construction takes to complete.	Verify that the farm takes 45 seconds to build.			
39. Select one of the plots around the Scuba Troop HQ and build a armory	Verify that the building begins construction			
40. Select a new plot around the Scuba Troop HQ and try to build a turret, farm, and armory	Verify that you cannot construct a new building			
41. Select a new plot around the Bomb Ship HQ and try to build a turret, farm, and armory	Verify that you cannot construct a new building			
42. Select a new plot around the Sub HQ and try to build a turret, farm, and armory	Verify that you cannot construct a new building			
43. Wait for the armory to finish construction.	Verify that the building is completed successfully.			
44. Select one of the plots around the Bomb Ship HQ and build a armory	Verify that the building begins construction			

45. Select a new plot around the Bomb Ship HQ and try to build a turret, farm, and armory	Verify that you cannot construct a new building			
46. Select a new plot around the Scuba Troop HQ and try to build a turret, farm, and armory	Verify that you cannot construct a new building			
47. Select a new plot around the Sub HQ and try to build a turret, farm, and armory	Verify that you cannot construct a new building			
48. Select a Scuba Troop unit	Verify that the Scuba Troop's icon has a blue arrow in the corner			
49. Select a Bomb Ship unit	Verify that the Bomb Ship's icon has a blue arrow in the corner			
50. Select a Sub unit	Verify that the Sub's icon has a green arrow in the corner			
Switch to the Piscivian faction				
51. Select one of the plots around the Merman HQ and build a jellyfish tower	Verify that the building begins construction			
52. Select a new plot around the Merman HQ and try to build a jellyfish tower, farm, and armory	Verify that you cannot construct a new building			
53. Select a new plot around the Turtle HQ and try to build a jellyfish tower, farm, and armory	Verify that you cannot construct a new building			
54. Select a new plot around the Manta Ray HQ and try	Verify that you cannot construct a new building			

to build a jellyfish tower, farm, and armory				
55. Wait for the jellyfish tower to finish construction. Then, select a new plot and build a second jellyfish tower around the Merman HQ. When the building starts construction, time how long the construction takes to complete.	Verify that the turret takes 45 seconds to build.			
56. Select one of the plots around the Turtle HQ and build a jellyfish tower	Verify that the building begins construction			
57. Select a new plot around the Turtle HQ and try to build a jellyfish tower, farm, and armory	Verify that you cannot construct a new building			
58. Select a new plot around the Merman HQ and try to build a jellyfish tower, farm, and armory	Verify that you cannot construct a new building			
59. Select a new plot around the Manta Ray HQ and try to build a jellyfish tower, farm, and armory	Verify that you cannot construct a new building			
60. Wait for the jellyfish tower to finish construction.	Verify that the building is completed successfully.			

61. Select one of the plots around the Manta Ray HQ and build a jellyfish tower	Verify that the building begins construction			
62. Select a new plot around the Manta Ray HQ and try to build a jellyfish tower, farm, and armory	Verify that you cannot construct a new building			
63. Select a new plot around the Merman HQ and try to build a jellyfish tower, farm, and armory	Verify that you cannot construct a new building			
64. Select a new plot around the Turtle HQ and try to build a jellyfish tower, farm, and armory	Verify that you cannot construct a new building			
Switch to the Human faction				
65. Move a group of units to any jellyfish tower building	Verify that the jellyfish tower attacks and kills the units			
Switch to the Piscivian faction				
66. Select one of the plots around the Turtle HQ and build a farm	Verify that the building begins construction			
67. Select a new plot around the Turtle HQ and try to build a jellyfish tower, farm, and armory	Verify that you cannot construct a new building			
68. Select a new plot around the Merman HQ and try to build a jellyfish tower, farm, and armory	Verify that you cannot construct a new building			
69. Select a new plot	Verify that you			

around the Manta Ray HQ and try to build a jellyfish tower, farm, and armory	cannot construct a new building			
70. Wait for the farm to finish construction. Then, select a new plot and build a second farm around the Turtle HQ. When the building starts construction, time how long the construction takes to complete.	Verify that the farm takes 45 seconds to build.			
71. Select one of the plots around the Merman HQ and build a farm	Verify that the building begins construction			
72. Select a new plot around the Merman HQ and try to build a jellyfish tower, farm, and armory	Verify that you cannot construct a new building			
73. Select a new plot around the Turtle HQ and try to build a jellyfish tower, farm, and armory	Verify that you cannot construct a new building			
74. Select a new plot around the Manta Ray HQ and try to build a jellyfish tower, farm, and armory	Verify that you cannot construct a new building			
75. Wait for the farm to finish construction.	Verify that the building is completed successfully.			
76. Select one of the	Verify that the			

plots around the Manta Ray HQ and build a farm	building begins construction			
77. Select a new plot around the Manta Ray HQ and try to build a jellyfish tower, farm, and armory	Verify that you cannot construct a new building			
78. Select a new plot around the Merman HQ and try to build a jellyfish tower, farm, and armory	Verify that you cannot construct a new building			
79. Select a new plot around the Turtle HQ and try to build a jellyfish tower, farm, and armory	Verify that you cannot construct a new building			
80. Wait until units spawn	Verify that three units appear around the Turtle HQ and two units appear around the Merman and Manta Ray HQ			
81. Select one of the plots around the Manta Ray HQ and build a armory	Verify that the building begins construction			
82. Select a new plot around Manta Ray HQ and try to build a jellyfish tower, farm, and armory	Verify that you cannot construct a new building			
83. Select a new plot around the Merman HQ and try to build a jellyfish tower, farm, and armory	Verify that you cannot construct a new building			
84. Select a new plot around the Turtle	Verify that you cannot construct a			

HQ and try to build a jellyfish tower, farm, and armory	new building			
85. Wait for the armory to finish construction. Then, select a new plot and build a second farm around the Manta Ray HQ. When the building starts construction, time how long the construction takes to complete.	Verify that the farm takes 45 seconds to build.			
86. Select one of the plots around the Merman HQ and build an armory	Verify that the building begins construction			
87. Select a new plot around the Merman HQ and try to build a jellyfish tower, farm, and armory	Verify that you cannot construct a new building			
88. Select a new plot around the Turtle HQ and try to build a jellyfish tower, farm, and armory	Verify that you cannot construct a new building			
89. Select a new plot around the Manta Ray HQ and try to build a jellyfish tower, farm, and armory	Verify that you cannot construct a new building			
90. Wait for the jellyfish tower to finish construction.	Verify that the building is completed successfully.			
91. Select one of the plots around the	Verify that the building begins			

Turtle HQ and build a farm	construction			
92. Select a new plot around the Turtle HQ and try to build a jellyfish tower, farm, and armory	Verify that you cannot construct a new building			
93. Select a new plot around the Merman HQ and try to build a jellyfish tower, farm, and armory	Verify that you cannot construct a new building			
94. Select a new plot around the Manta Ray HQ and try to build a jellyfish tower, farm, and armory	Verify that you cannot construct a new building			
95. Select a Merman unit	Verify that the Merman's icon has a blue arrow in the corner			
96. Select a Turtle unit	Verify that the Turtle's icon has a blue arrow in the corner			
97. Select a Manta Ray unit	Verify that the Manta Ray's icon has a green arrow in the corner			
98. Send units to destroy all of the Human turrets, farms, and armories	Verify that each building is capable of being destroyed.			
Switch to the Human faction				
99. As the Human faction, send units to destroy all of the Piscivian buildings	Verify that each building is capable of being destroyed.			

Appendix T: RTS Mode Interface Functionality Test Plan

The purpose of this test plan is to determine the functionality of the user interface in the RTS mode of the game. This test plan will verify the important aspects of RTS mode including the chat window, unit information pane, mini-map, and commander view pane. You will be required to use two computers during this test plan.

Step	Expected Result	Pass	Fail	Comment
1. Run BloodTide.exe on 'Computer 1'	Verify that the game runs and you are at the main menu.			
2. Click "Host Game," then "Host"	Verify you are taken to the faction selection screen			
3. Select Piscivan faction.	Verify that your player name is moved from the "undecided" column and is listed under the Piscivan faction.			
4. Run BloodTide.exe on 'Computer 2'	Verify that the game runs and you are at the main menu.			
5. Join your created game and select the Human faction	Verify that your player name is moved from the "undecided" column and is listed under the Human faction.			
6. Click "Start" on both of the computers	Verify that each game displays the loading screen for the appropriate faction, then each game loads correctly			
7. Press ENTER to open the chat window on both computers	Verify that the chat window appears			
8. As the Piscivan faction (Computer 1),	Verify that the message appears for the Human faction			

send a message in the chat window				
9. As the Human faction (Computer 2), send a message in the chat window	Verify that the message appears for the Piscivian faction			
Switch to the Piscivian faction (Computer 1)				
10. Wait for units to spawn	Verify that the interface looks like Figure 128: Human Interface			



Figure 128: Human Interface

11. Select one unit	Verify that the unit is highlighted and an icon appears in the bottom-middle of the interface			
12. De-select the unit	Verify that the unit is no longer highlighted and the icon disappears			
13. Select a large group of units by	Verify that the units are highlighted and			

left clicking and dragging a box around the units	an icon appears in the bottom-middle of the interface, scaling the unit icons if needed so they all fit on screen			
14. De-select the units	Verify that the units are no longer highlighted and the icons disappear			
15. Wait for units to spawn	Verify that small dots appear on the mini-map for each unit that spawns, as shown in Figure 129: Human Mini-Map - no units and Figure 130: Human Mini-Map - with units.			



Figure 129: Human Mini-Map - no units



Figure 130: Human Mini-Map - with units

<p>16.Select a group of units and issue an order to move</p>	<p>Verify the dots on the map move in correspondence with the order issues</p>			
<p>17.Select a group of units and issue an order to move near an enemy base, wait for the units to be killed</p>	<p>Verify the dots on the map move in correspondence with the order issues and disappear when are killed</p>			
<p>18.Mouse over and off of the the commander selection triangle</p>	<p>Verify that when the mouse is not over the triangle, it looks like Figure 131: Human Commander Selection.</p> <p>Verify that when the mouse is over the triangle, it looks like Figure 132: Human Commander Selection - Hover.</p>			



Figure 131: Human Commander Selection



Figure 132: Human Commander Selection - Hover

<p>19. Select any commander race</p>	<p>Verify that the commander selection triangle disappears and the commander view appears over the mini-map</p>			
<p>20. Press “t” to enter field-commander view</p>	<p>Verify that the commander view disappears</p>			
<p>21. Move the field commander to a new location, then press “t” to enter tactical-commander view</p>	<p>Verify that the commander view reappears and shows the new location of the commander</p>			
<p>Switch to the Human faction (Computer 2)</p>				
<p>22. Select a unit and issue an order to attack the Piscivian field-commander</p>	<p>Verify that the units move</p>			
<p>Switch to the Piscivian faction (Computer 1)</p>				
<p>23. Wait for the enemy units to attack the field commander</p>	<p>Verify that the enemy units appear on the mini-map and in the commander view</p>			
<p>Switch to the Human faction (Computer 2)</p>				
<p>24. Wait for units to spawn</p>	<p>Verify that the interface looks like Figure 133: Piscivian Interface.</p>			



Figure 133: Piscivian Interface

<p>25. Select one unit</p>	<p>Verify that the unit is highlighted and an icon appears in the bottom-middle of the interface</p>			
<p>26. De-select the unit</p>	<p>Verify that the unit is no longer highlighted and the icon disappears</p>			
<p>27. Select a large group of units by left clicking and dragging a box around the units</p>	<p>Verify that the units are highlighted and an icon appears in the bottom-middle of the interface, scaling the unit icons if needed so they all fit on screen</p>			
<p>28. De-select the units</p>	<p>Verify that the units are no longer highlighted and the icons disappear</p>			
<p>29. Wait for units to spawn</p>	<p>Verify that small dots appear on the mini-map for each unit that spawns, as</p>			

	shown in Figure 134: Piscivian Mini-Map - no units and Figure 135: Piscivian Mini-Map - with units			
--	--	--	--	--



Figure 134: Piscivian Mini-Map - no units



Figure 135: Piscivian Mini-Map - with units

<p>30. Select a group of units and issue an order to move</p>	<p>Verify the dots on the map move in correspondence with the order issues</p>			
<p>31. Select a group of units and issue an order to move near an enemy base, wait for the units to be killed</p>	<p>Verify the dots on the map move in correspondence with the order issues and disappear when are killed</p>			
<p>32. Mouse over and off of the commander selection triangle</p>	<p>Verify that when the mouse is not over the triangle, it looks like Figure 136: Piscivian Commander Selection.</p> <p>Verify that when the mouse is over the</p>			

	triangle, it looks like Figure 138: Piscivian Commander Selection - Hover.			
--	--	--	--	--



Figure 136: Piscivian Commander Selection



Figure 137: Piscivian Commander Selection - hover

33. Figure 138: Piscivian Commander Selection - Hover	Select any commander race	Verify that the commander selection triangle disappears and the commander view appears over the mini-map			
34. Press "t" to enter field-commander view		Verify that the commander view disappears			
35. Move the field		Verify that the			

commander to a new location, then press “t” to enter tactical-commander view	commander view reappears and shows the new location of the commander			
Switch to the Piscivian faction (Computer 2)				
36. Select a unit and issue an order to attack the Human field-commander	Verify that the units move			
Switch to the Human faction (Computer 1)				
37. Wait for the enemy units to attack the field commander	Verify that the enemy units appear on the mini-map and in the commander view			

Appendix U: Action Mode Interface Functionality Test Plan

The purpose of this test plan is to determine the functionality of the user interface in the Action mode of the game. This test plan will verify the important aspects of Action mode including the chat window, the commander information pane, and the mini-map. You will be required to use two computers during this test plan.

Step	Expected Result	Pass	Fail	Comment
1. Run BloodTide.exe on 'Computer 1'	Verify that the game runs and you are at the main menu.			
2. Click "Host Game," then "Host"	Verify you are taken to the faction selection screen			
3. Select Piscivan faction.	Verify that your player name is moved from the "undecided" column and is listed under the Piscivan faction.			
4. Run BloodTide.exe on 'Computer 2'	Verify that the game runs and you are at the main menu.			
5. Join your created game and select the Human faction	Verify that your player name is moved from the "undecided" column and is listed under the Human faction.			
6. Click "Start" on both of the computers	Verify that each game displays the loading screen for the appropriate faction, then each game loads correctly			
7. Select a field commander race and press "t" to enter field-commander mode on both computers	Verify that the view switches to field commander mode Verify that the correct field-commander race icon appears in the			

	commander info pane			
8. Press ENTER to open the chat window on both computers	Verify that the chat window appears			
9. As the Piscivian faction (Computer 1), send a message in the chat window	Verify that the message appears for the Human faction			
10. As the Human faction (Computer 2), send a message in the chat window	Verify that the message appears for the Piscivian faction			
Switch to the Piscivian faction (Computer 1)				
11. Wait for units to spawn	<p>Verify that the interface looks like Figure 139: Piscivian Field Commander Interface.</p> <p>Verify that small dots appear on the mini-map for each unit that spawns, as shown in Figure 140: Piscivian Field Commander Interface - units.</p>			

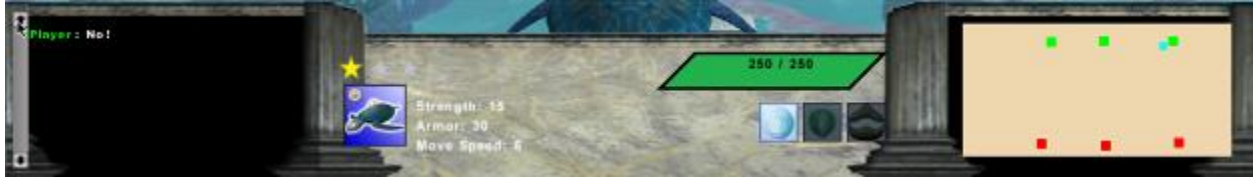


Figure 139: Piscivian Field Commander Interface



Figure 140: Piscivian Field Commander Interface - units

<p>12. Switch to tactical commander mode, then select a group of units and issue an order to move, then re-enter field-commander mode</p>	<p>Verify the dots on the mini-map move in correspondence with the order issues</p>			
<p>13. In tactical-commander mode, select a group of units and issue an order to move near an enemy base</p> <p>Re-enter field-commander mode, then wait for the units to be killed</p>	<p>Verify the dots on the map move in correspondence with the order issues and disappear when are killed</p>			
<p>14. Move the field commander to an enemy HQ, then let yourself be attacked</p>	<p>Verify that your health bar decreases accordingly</p>			
<p>15. Attack the enemy units until you gain a level</p>	<p>Verify that a second star appears above the health bar and the second skill is not grayed out</p>			

16. Attack the enemy units until you gain another level	Verify that a third star appears above the health bar and the third skill is not grayed out			
Switch to the Human faction (Computer 2)				
17. Wait for units to spawn	<p>Verify that the interface looks like Figure 141: Human Field Commander Interface</p> <p>Verify that small dots appear on the mini-map for each unit that spawns, as shown in Figure 142: Human Field Commander Interface - units</p>			

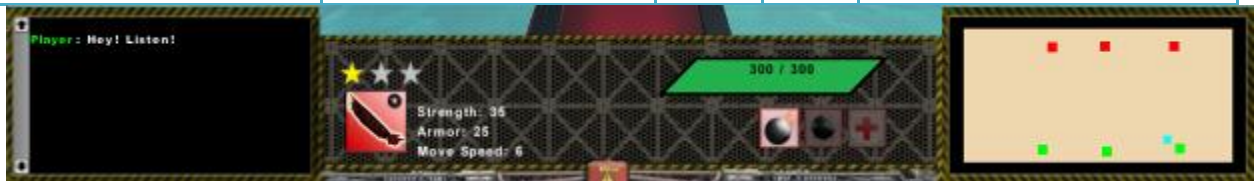


Figure 141: Human Field Commander Interface

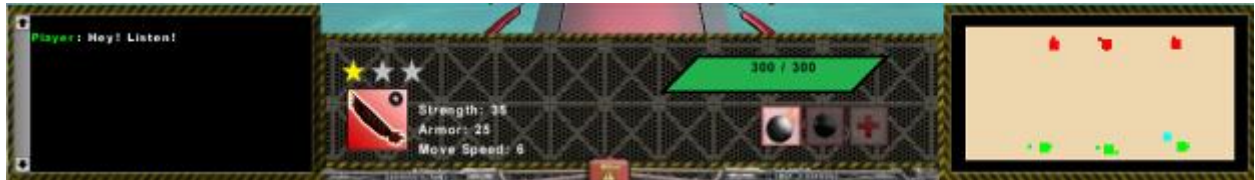


Figure 142: Human Field Commander Interface - units

18. Switch to tactical commander mode, then select a group of units and issue an order to move, then re-enter field-commander mode	Verify the dots on the mini-map move in correspondence with the order issues			
19. In tactical-commander mode, select a	Verify the dots on the map move in correspondence with			

<p>group of units and issue an order to move near an enemy base</p> <p>Re-enter field-commander mode, then wait for the units to be killed</p>	<p>the order issues and disappear when are killed</p>			
<p>20. Move the field commander to an enemy HQ, then let yourself be attacked</p>	<p>Verify that your health bar decreases accordingly</p>			
<p>21. Attack the enemy units until you gain a level</p>	<p>Verify that a second star appears above the health bar and the second skill is not grayed out</p>			
<p>22. Attack the enemy units until you gain another level</p>	<p>Verify that a third star appears above the health bar and the third skill is not grayed out</p>			

Appendix V: Multiplayer Functionality Test Plan

The purpose of this test plan is to determine the functionality of multiplayer gameplay. This test plan will test hosting and joining abilities for the player. You will be required to use two computers during this test plan.

Step	Expected Result	Pass	Fail	Comment
1. Run BloodTide.exe on 'Computer 1'	Verify that the game runs and you are at the main menu.			
2. Click "Host Game," then "Host"	Verify you are taken to the faction selection screen			
3. Select Piscivan faction.	Verify that your player name is moved from the "undecided" column and is listed under the Piscivian faction.			
4. Run BloodTide.exe on 'Computer 2'	Verify that the game runs and you are at the main menu.			
5. Join your created game and select the Human faction	Verify that your player name is moved from the "undecided" column and is listed under the Human faction.			
6. Click "Start" on both of the computers	Verify that each game displays the loading screen for the appropriate faction, then each game loads correctly			
7. Wait for units to spawn	Verify that the mini-map updates appropriately for both computers			
Switch to the Piscivian faction (Computer 1)				
8. Move the camera over the Scuba Troop HQ	Verify that, on Computer 2, the HQ model is correct and			

	scuba troop unit spawns correctly			
9. Move the camera over the Submarine HQ	Verify that, on Computer 2, the HQ model is correct and sub unit spawns correctly			
10. Move the camera over the Bomb Ship HQ	Verify that, on Computer 2, the HQ model is correct and bomb ship unit spawns correctly			
11. Select a plot around the Manta Ray HQ and construct a guard tower	Verify that, on Computer 2, the building plays the build effect and the building appears when finished			
12. Select a plot around the Merman HQ and construct a farm	Verify that, on Computer 2, the building plays the build effect and the correct building model appears when finished			
13. Select a plot around the Turtle HQ and construct an armory	Verify that, on Computer 2, the building plays the build effect and the correct building model appears when finished			
Switch to the Human faction (Computer 2)				
14. Move the camera over the Merman HQ	Verify that, on Computer 1, the HQ model is correct and merman unit spawns correctly			
15. Move the camera over the Manta Ray HQ	Verify that, on Computer 1, the HQ model is correct and manta ray unit spawns correctly			
16. Move the camera over the Turtle HQ	Verify that, on Computer 1, the HQ model is correct and			

	turtle unit spawns correctly			
17. Select one bomb ship unit and issue a move command	Verify that, on Computer 1, the bomb ship unit moves correctly			
18. Select a group of bomb ship units and issue a move command	Verify that, on Computer 1, the bomb ship units all move correctly			
19. Select one bomb ship unit and issue an attack command on an enemy turtle	Verify that, on Computer 1, the bomb ship unit moves and attacks correctly Verify that the enemy unit dies correctly			
20. Select a group of bomb ship units and issue an attack command on the enemy armory	Verify that, on Computer 1, the bomb ship units all move and attack correctly Verify that the enemy building dies correctly			
21. Select one scuba troop unit and issue a move command	Verify that, on Computer 1, the scuba troop unit moves correctly			
22. Select a group of scuba troop units and issue a move command	Verify that, on Computer 1, the scuba troop units all move correctly			
23. Select one scuba troop unit and issue an attack command on an enemy merman	Verify that, on Computer 1, the scuba troop unit moves and attacks correctly Verify that the enemy unit dies correctly			
24. Select a group	Verify that, on			

of scuba troop units and issue an attack command on the enemy farm	Computer 1, the scuba troop units all move and attack correctly Verify that the enemy building dies correctly			
25. Select one sub unit and issue a move command	Verify that, on Computer 1, the sub unit moves correctly			
26. Select a group of sub units and issue a move command	Verify that, on Computer 1, the sub units all move correctly			
27. Select one sub unit and issue an attack command on an enemy manta ray	Verify that, on Computer 1, the sub unit moves and attacks correctly Verify that the enemy unit dies correctly			
28. Select a group of sub units and issue an attack command on an enemy guard tower	Verify that, on Computer 1, the sub units all move and attack correctly Verify that the enemy building dies correctly			
29. Select a plot around the Sub HQ and construct a guard tower	Verify that, on Computer 1, the building plays the build effect and the building appears when finished			
30. Select a plot around the Scuba Troop HQ and construct a farm	Verify that, on Computer 1, the building plays the build effect and the correct building model appears when finished			
31. Select a plot around the Bomb	Verify that, on Computer 1, the			

Ship HQ and construct an armory	building plays the build effect and the correct building model appears when finished			
Switch to the Piscivian faction (Computer 1)				
32. Select one turtle unit and issue a move command	Verify that, on Computer 2, the turtle unit moves correctly			
33. Select a group of turtle units and issue a move command	Verify that, on Computer 2, the turtle units all move correctly			
34. Select one turtle unit and issue an attack command on an enemy bomb ship	Verify that, on Computer 2, the turtle unit moves and attacks correctly Verify that the enemy unit dies correctly			
35. Select a group of turtle units and issue an attack command on the enemy armory	Verify that, on Computer 2, the turtle units all move and attack correctly Verify that the enemy building dies correctly			
36. Select one merman unit and issue a move command	Verify that, on Computer 2, the merman unit moves correctly			
37. Select a group of merman units and issue a move command	Verify that, on Computer 2, the merman units all move correctly			
38. Select one merman unit and issue an attack command on an enemy scuba troop	Verify that, on Computer 2, the merman unit moves and attacks correctly Verify that the enemy unit dies correctly			

<p>39. Select a group of merman units and issue an attack command on the enemy farm</p>	<p>Verify that, on Computer 2, the merman units all move and attack correctly</p> <p>Verify that the enemy building dies correctly</p>			
<p>40. Select one manta ray unit and issue a move command</p>	<p>Verify that, on Computer 2, the manta ray unit moves correctly</p>			
<p>41. Select a group of manta ray units and issue a move command</p>	<p>Verify that, on Computer 2, the manta ray units all move correctly</p>			
<p>42. Select one manta ray unit and issue an attack command on an enemy sub</p>	<p>Verify that, on Computer 2, the manta ray unit moves and attacks correctly</p> <p>Verify that the enemy unit dies correctly</p>			
<p>43. Select a group of manta ray units and issue an attack command on the enemy guard tower</p>	<p>Verify that, on Computer 2, the manta ray units all move and attack correctly</p> <p>Verify that the enemy building dies correctly</p>			
<p>44. Select the merman as the Field Commander race and press “T” to enter mode</p>	<p>Verify that, on Computer 2, the Field Commander spawns and appears on the mini-map</p>			
<p>45. Use the first ability on an enemy unit</p>	<p>Verify that, on Computer 2, the correct animation plays and the correct damage is dealt</p>			

<p>46. Use the second ability on an enemy unit</p> <p><i>(unlock if not already unlocked)</i></p>	<p>Verify that, on Computer 2, the correct animation plays, the correct projectile appears, and the correct damage is dealt</p>			
<p>47. Use the third ability on an enemy unit</p> <p><i>(unlock if not already unlocked)</i></p>	<p>Verify that, on Computer 2, the correct animation plays and the correct amount is healed</p>			
<p>48. Allow the field commander to be killed</p>	<p>Verify that, on Computer 2, the correct death animation plays and the model disappears</p>			
<p>49. Select the manta ray as the Field Commander race and press “T” to enter mode</p>	<p>Verify that, on Computer 2, the Field Commander spawns and appears on the mini-map</p>			
<p>50. Use the first ability on an enemy unit</p>	<p>Verify that, on Computer 2, the correct animation plays and the correct damage is dealt</p>			
<p>51. Use the second ability on an enemy unit</p> <p><i>(unlock if not already unlocked)</i></p>	<p>Verify that, on Computer 2, the model becomes invisible until it attacks or the time runs out</p>			
<p>52. Use the third ability on an enemy unit</p> <p><i>(unlock if not already unlocked)</i></p>	<p>Verify that, on Computer 2, the correct animation plays, the correct projectile appears, and the correct damage is dealt</p>			
<p>53. Allow the field</p>	<p>Verify that, on</p>			

commander to be killed	Computer 2, the correct death animation plays and the model disappears			
54. Select the turtle as the Field Commander race and press “T” to enter mode	Verify that, on Computer 2, the Field Commander spawns and appears on the mini-map			
55. Use the first ability on an enemy unit	Verify that, on Computer 2, the correct animation plays, the correct projectile appears, and the correct damage is dealt			
56. Use the second ability on an enemy unit <i>(unlock if not already unlocked)</i>	Verify that, on Computer 2, the correct animation plays and damage is not taken for an amount of time			
57. Use the third ability on an enemy unit <i>(unlock if not already unlocked)</i>	Verify that, on Computer 2, the correct animation plays and the correct damage is dealt			
58. Allow the field commander to be killed	Verify that, on Computer 2, the correct death animation plays and the model disappears			
Switch to the Human faction (Computer 2)				
59. Select the scuba troop as the Field Commander race and press “T” to enter mode	Verify that, on Computer 1, the Field Commander spawns and appears on the mini-map			
60. Use the first ability on an	Verify that, on Computer 1, the			

enemy unit	correct animation plays and the correct damage is dealt			
61. Use the second ability on an enemy unit <i>(unlock if not already unlocked)</i>	Verify that, on Computer 1, the correct animation plays and the correct damage is dealt over time			
62. Use the third ability on an enemy unit <i>(unlock if not already unlocked)</i>	Verify that, on Computer 1, the correct animation plays and the correct amount of damage is reflected to the attacking unit			
63. Allow the field commander to be killed	Verify that, on Computer 1, the correct death animation plays and the model disappears			
64. Select the sub as the Field Commander race and press “T” to enter mode	Verify that, on Computer 1, the Field Commander spawns and appears on the mini-map			
65. Use the first ability on an enemy unit	Verify that, on Computer 1, the correct animation plays, the correct projectile appears, and the correct damage is dealt			
66. Use the second ability on an enemy unit <i>(unlock if not already unlocked)</i>	Verify that, on Computer 1, the model becomes invisible until it attacks or the time runs out			
67. Use the third ability on an enemy unit	Verify that, on Computer 1, the correct animation plays, the correct			

<i>(unlock if not already unlocked)</i>	projectile appears, and the correct damage is dealt			
68. Allow the field commander to be killed	Verify that, on Computer 1, the correct death animation plays and the model disappears			
69. Select the bomb ship as the Field Commander race and press “T” to enter mode	Verify that, on Computer 1, the Field Commander spawns and appears on the mini-map			
70. Use the first ability on an enemy unit	Verify that, on Computer 1, the correct animation plays, the correct projectile appears, and the correct damage is dealt			
71. Use the second ability on an enemy unit <i>(unlock if not already unlocked)</i>	Verify that, on Computer 1, the correct animation plays, the correct projectile appears, the correct damage is dealt, and all nearby units are stunned for a period of time			
72. Use the third ability on an enemy unit <i>(unlock if not already unlocked)</i>	Verify that, on Computer 1, the correct animation plays and the correct amount is healed			
73. Allow the field commander to be killed	Verify that, on Computer 1, the correct death animation plays and the model disappears			
74. Select a group	Verify that, on			

of units and destroy the Manta Ray HQ	Computer 1, the building model disappears			
75. Select a group of units and destroy the Merman HQ	Verify that, on Computer 1, the building model disappears			
76. Select a group of units and destroy the Turtle HQ	Verify that, on Computer 1, the building model disappears			
Switch to the Piscivian faction (Computer 1)				
77. Select a group of units and destroy the Submarine HQ	Verify that, on Computer 2, the building model disappears			
78. Select a group of units and destroy the Scuba Troop HQ	Verify that, on Computer 2, the building model disappears			
79. Select a group of units and destroy the Bomb Ship HQ	Verify that, on Computer 2, the building model disappears			

Appendix W: Map Functionality Test Plan

The purpose of this test plan is to determine the functionality of the game map. This test plan will test map boundaries, unit collisions with buildings, and camera.

Step	Expected Result	Pass	Fail	Comment
1. Run BloodTide.exe	Game successfully loads. You should see the main menu screen.			
2. Click “Host Game” and click “Host”	Verify that you are taken to the faction selection screen (Figure 109)			
3. Select the Human faction, and click “Start”	Verify the loading screen appears for the Human faction. Then, the game should start. Verify that you are playing as the Human faction.			
4. Verify the initial position of the camera.	See Figure 143 for picture of the initial camera position for the Human faction.			
5. Press the Esc key and select “Exit Current Game”	Verify that you are brought back to the main menu.			
Host a new game				
6. Select the Piscivian faction, and click “Start”	Verify the loading screen appears for the Piscivian Faction. Then, the game should start. Verify that you are playing as the Piscivian faction.			
7. Verify the initial position of the camera.	See Figure 144 for picture of the initial camera position for the Piscivian faction.			
8. Wait for some	Verify that you have			

units to spawn, then Left Click and hold to drag select a group of units	selected a group of units			
9. Attempt to order the units to move outside of the map area.	Verify that you cannot issue an order outside the map boundary.			
10. Move your selected units as close to the map border as possible	Your selected units should move to your designated location. Verify that the units do not go outside the map border			
11. Select the manta ray field-commander and press “T” on your keyboard to enter the field commander role.	You should now be playing as the manta ray field commander			
12. Move along the border of the entire map.	Insure that you do not go outside the map boundaries.			
13. Press the Esc key and select “Exit Current Game”	Verify that you are brought back to the main menu.			

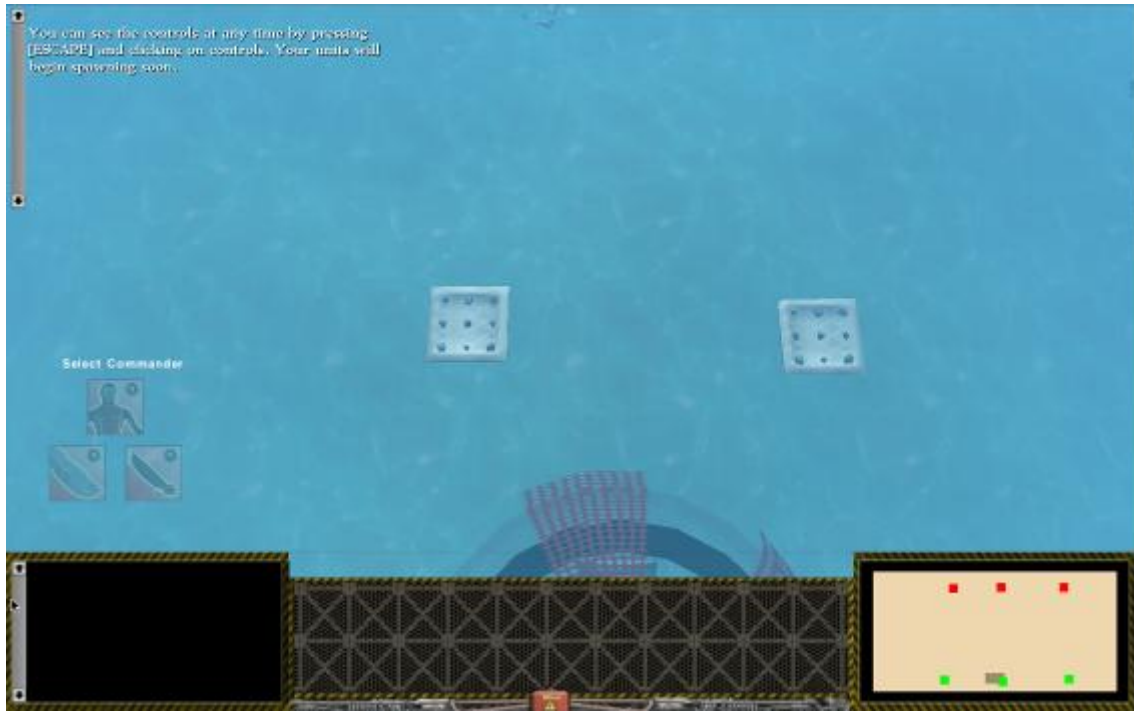


Figure 143: Human Starting Location



Figure 144: Piscivian Starting Location

Appendix X: Artistic Functionality Test Plan

The purpose of this test plan is to determine the functionality of the artistic assets. This test plan will test correct model use, correct animations, and correct icons.

Step	Expected Result	Pass	Fail	Comment
<ul style="list-style-type: none">• Run BloodTide.exe	Verify that pop-up displays asking user to play tutorial if first time.			
<ul style="list-style-type: none">• Click “Host Game”, then “Host”	Verify you are taken to the faction selection screen			
<ul style="list-style-type: none">• Select the human faction, then click “Start”	Verify that the “loading screen” appears with the game objective and tips for one of the three units			
<ul style="list-style-type: none">• Exit the game, and re-start the until all three of the unit tips appear	Verify that all three loading screens appear in the correct orientation			
<ul style="list-style-type: none">• Mouse over the Scuba Troop HQ	Verify that the Scuba Troop HQ looks like Figure 145			

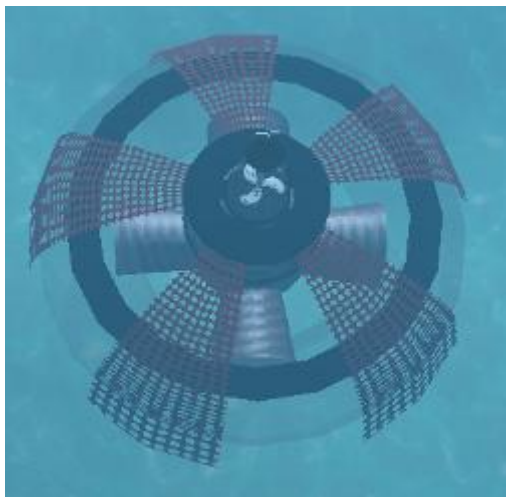


Figure 145: Scuba Troop HQ

<ul style="list-style-type: none"> • Mouse over the Sub HQ 	Verify that the Sub HQ looks like Figure 146: Submarine HQ			
---	--	--	--	--



Figure 146: Submarine HQ

<ul style="list-style-type: none"> • Mouse over the Bomb Ship HQ 	Verify that the Bomb Ship HQ looks like Figure 147: Bomb Ship HQ			
---	--	--	--	--

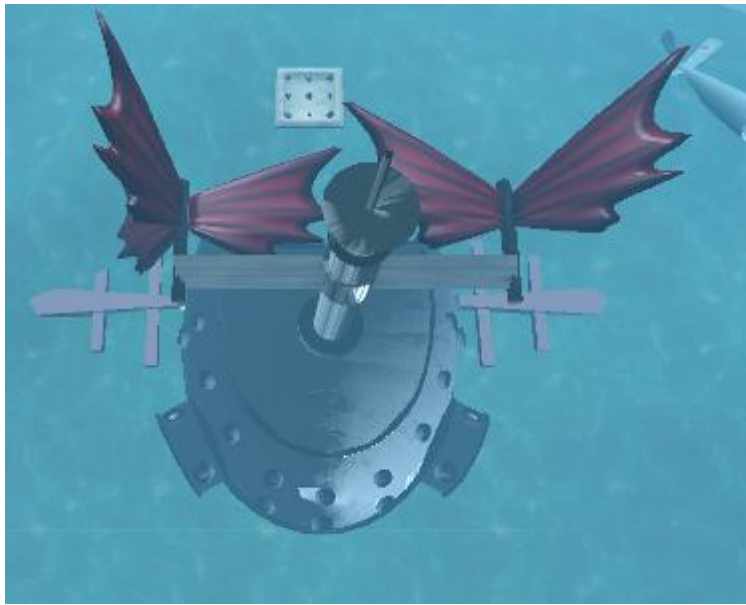


Figure 147: Bomb Ship HQ

<ul style="list-style-type: none"> • Wait until units spawn 	Verify that the Scuba Troop unit looks like Figure 148, the Sub unit looks like Figure 149, and the Bomb Ship unit looks like Figure 150			
--	--	--	--	--



Figure 148: Scuba Troop



Figure 149: Submarine



Figure 150: Bomb Ship

<ul style="list-style-type: none"> • Select a Scuba Troop Unit 	Verify that the icon that appears looks like Figure 151			
---	---	--	--	--



Figure 151: Scuba Troop Icon

<ul style="list-style-type: none"> • Select a Sub Unit 	Verify that the icon that appears looks			
---	---	--	--	--

	like Figure 152			
--	-----------------	--	--	--



Figure 152: Submarine Icon

<ul style="list-style-type: none"> Select a Bomb Ship Unit 	Verify that the icon that appears looks like Figure 153			
---	---	--	--	--



Figure 153: Bomb Ship Icon

<ul style="list-style-type: none"> Select an open plot and build a turret around the Scuba Troop HQ 	Verify that the empty plot disappears once the building begins, then a dust cloud appears until the building is completed, which looks like Figure 154			
--	--	--	--	--



Figure 154: Turret at Scuba Troop HQ

<ul style="list-style-type: none"> Select an open plot and build a 	Verify that the empty plot disappears once the building begins,			
---	---	--	--	--

farm around the Bomb Ship HQ	then a dust cloud appears until the building is completed, which looks like Figure 155			
------------------------------	--	--	--	--



Figure 155: Farm at Bomb Ship HQ

<ul style="list-style-type: none"> Select an open plot and build an armory around the Sub HQ 	Verify that the empty plot disappears once the building begins, then a dust cloud appears until the building is completed, which looks like Figure 156			
---	--	--	--	--

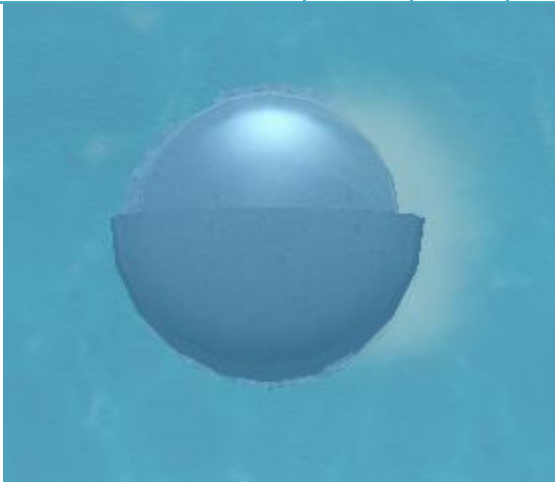


Figure 156: Armory at Submarine HQ

<ul style="list-style-type: none"> Select the Scuba Troop as the field commander race 	Verify that your unit icon is correct and you have only one			
--	---	--	--	--

and enter FC mode	skill unlocked, with the other two grayed out, as in Figure 157			
-------------------	---	--	--	--




Figure 157: Scuba Troop Field Commander Interface - 1st Rank

<ul style="list-style-type: none"> Move to the enemy HQ and attack units until you level up 	Verify that, as you gain levels, the new skills unlock, as in Figure 158			
--	--	--	--	--




Figure 158: Scuba Troop Field Commander Interface - 3rd Rank

<ul style="list-style-type: none"> Use the first skill 	Verify that the correct animation plays			
<ul style="list-style-type: none"> Use the second skill 	Verify that the correct animation plays			
<ul style="list-style-type: none"> Use the third skill 	Verify that the correct animation plays			
<ul style="list-style-type: none"> Move the field commander to the sub base to switch FC races 	Verify that your skill level is saved and all three levels are unlocked, as in Figure 159			



Figure 159: Sub Field Commander Interface - 3rd Rank

<ul style="list-style-type: none"> Use the first skill 	Verify that the correct animation plays			
<ul style="list-style-type: none"> Use the second 	Verify that the			

skill	correct animation plays			
• Use the third skill	Verify that the correct animation plays			
• Move the field commander to the bomb ship base to switch FC races	Verify that your skill level is saved and all three levels are unlocked, as in Figure 160			



Figure 160: Bomb Ship Field Commander Interface - 3rd Rank

• Use the first skill	Verify that the correct animation plays			
• Use the second skill	Verify that the correct animation plays			
• Use the third skill	Verify that the correct animation plays			
• Attack a Piscivian building	Verify that the destroyed building becomes an empty plot again			
• Attack a Piscivian HQ	Verify that the destroyed HQ looks like Figure 161			



Figure 161: Destroyed Piscivian HQ

Exit game, then host a new game				
<ul style="list-style-type: none"> Select the Piscivian faction, then click "Start" 	Verify that the correct faction's loading screen appears			
<ul style="list-style-type: none"> Exit the game and re-start until all three of the unit tips appear 	Verify that all three loading screens appear in the correct orientation			
<ul style="list-style-type: none"> Mouse over the Merman HQ 	Verify that the Merman HQ looks like Figure 162			



Figure 162: Merman HQ

<ul style="list-style-type: none"> • Mouse over the Manta Ray HQ 	Verify that the Manta Ray HQ looks like Figure 163			
---	--	--	--	--



Figure 163: Manta Ray HQ

<ul style="list-style-type: none"> • Mouse over the Turtle HQ 	Verify that the Turtle HQ looks like Figure 164			
--	---	--	--	--



Figure 164: Turtle HQ

<ul style="list-style-type: none">• Wait until units spawn	Verify that the Merman unit looks like Figure 165, the Manta Ray unit looks like Figure 166, and the Bomb Ship unit looks like Figure 167			
--	---	--	--	--



Figure 165: Merman



Figure 166: Manta Ray



Figure 167: Turtle

<ul style="list-style-type: none"> Select a Merman Unit 	Verify that the icon that appears looks like Figure 168			
--	---	--	--	--



Figure 168: Merman Icon

<ul style="list-style-type: none"> Select a Manta Ray Unit 	Verify that the icon that appears looks like Figure 169			
---	---	--	--	--



Figure 169: Manta Ray Icon

<ul style="list-style-type: none"> Select a Turtle Unit 	Verify that the icon that appears looks like Figure 170			
--	---	--	--	--



Figure 170: Turtle Icon

<ul style="list-style-type: none"> Select an open plot and build a jelly tower around the Merman HQ 	Verify that the empty plot disappears once the building begins, then a dust cloud appears until the building is completed, which looks like Figure 171			
--	--	--	--	--



Figure 171: Jelly Tower at Merman HQ

<ul style="list-style-type: none"> Select an open plot and build a farm around the Turtle HQ 	<p>Verify that the empty plot disappears once the building begins, then a dust cloud appears until the building is completed, which looks like Figure 172</p>			
---	---	--	--	--



Figure 172: Farm at Turtle HQ

<ul style="list-style-type: none"> Select an open plot and build an armory around the Manta Ray HQ 	<p>Verify that the empty plot disappears once the building begins, then a dust cloud appears until the building is completed, which looks like Figure 173</p>			
---	---	--	--	--



Figure 173: Armory at Manta Ray HQ

<ul style="list-style-type: none"> Select the Merman as the field commander race and enter FC mode 	<p>Verify that your unit icon is correct and you have only one skill unlocked, with the other two grayed out, as in Figure 174</p>			
---	--	--	--	--



Figure 174: Merman Field Commander Interface - 1st Rank

<ul style="list-style-type: none"> Move to the enemy HQ and attack units until you level up 	<p>Verify that, as you gain levels, the new skills unlock, as in Figure 175</p>			
--	---	--	--	--



Figure 175: Merman Field Commander Interface - 3rd Rank

<ul style="list-style-type: none"> Use the first skill 	<p>Verify that the correct animation plays</p>			
<ul style="list-style-type: none"> Use the second skill 	<p>Verify that the correct animation plays</p>			
<ul style="list-style-type: none"> Use the third skill 	<p>Verify that the correct animation plays</p>			

<ul style="list-style-type: none"> Move the field commander to the Manta Ray base to switch FC races 	Verify that your skill level is saved and all three levels are unlocked, as in Figure 176			
---	---	--	--	--



Figure 176: Manta Ray Field Commander Interface - 3rd Rank

<ul style="list-style-type: none"> Use the first skill 	Verify that the correct animation plays			
<ul style="list-style-type: none"> Use the second skill 	Verify that the correct animation plays			
<ul style="list-style-type: none"> Use the third skill 	Verify that the correct animation plays			
<ul style="list-style-type: none"> Move the field commander to the turtle base to switch FC races 	Verify that your skill level is saved and all three levels are unlocked, as in Figure 177			

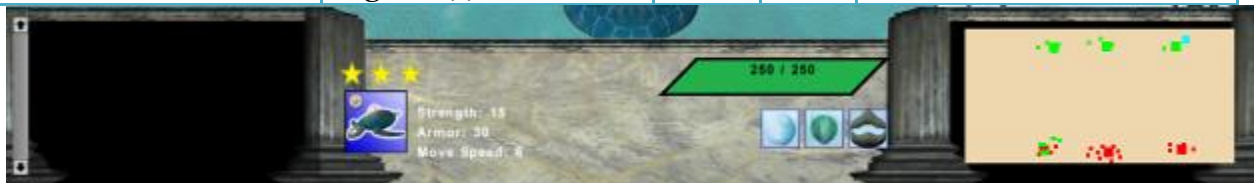


Figure 177: Turtle Field Commander Interface - 3rd Level

<ul style="list-style-type: none"> Use the first skill 				
<ul style="list-style-type: none"> Use the second skill 	Verify that the correct animation plays			
<ul style="list-style-type: none"> Use the third skill 	Verify that the correct animation plays			
<ul style="list-style-type: none"> Attack a Human building 	Verify that the destroyed building becomes an empty plot again			
<ul style="list-style-type: none"> Attack a Human HQ 	Verify that the destroyed HQ looks			

like Figure 178



Figure 178: Destroyed Human HQ

Appendix Y: Overall Help Functionality Test Plan

The purpose of this test plan is to determine the functionality of the in-game help. This test plan will test effectiveness of the in-game help. This test plan should be given to people who have not played *Blood Tide* and observed by team members.

Step	Expected Result	Pass	Fail	Comment
1. Install <i>Blood Tide</i>	Verify that game installs correctly			
2. Run BloodTide.exe	Verify that pop-up displays asking user to play tutorial if first time.			
3. Click “Host Game”, then “Host”	Verify you are taken to the faction selection screen			
4. Select human faction. Then, load the game on a second computer, join your game and select Piscivian faction. Click start	Verify that the “loading screen” appears with game objective and tip for the appropriate faction.			
Switch to the Human faction once the game loads				
5. Press ENTER to open chat window	Verify that chat window opens			
6. Wait until a message appears	Verify that a message appears reminding the player to move idle units			
7. Continue to wait until the next unique message appears	Confirm that a message appears reminding the player to construct a building			
8. Continue to wait until the next unique message	Verify that a message appears reminding the			

appears	player to choose a field commander race			
9. Hover over an open plot around any HQ	Verify that the plot highlights to indicate it is able to be built upon			
10. Select an open plot	Verify that three icons appear for each of the available buildings			
11. Hover over the three building icons	Verify that a pop-up window appears with appropriate information on each building			
12. Select a building and start construction	Verify that the building starts constructing and a message appears confirming the choice			
13. Wait until building is complete	Verify that a message appears telling the player that the building is complete			
Switch to the Piscivian faction				
14. Select a unit and issue an order to attack the human building	Verify that a message appears alerting the Human faction that their building is under attack			
15. Issue an order to attack a human HQ	Verify that a message appears alerting the Human faction that their base is under attack			
Switch to the Human faction				
16. Select a group of units by left clicking and dragging white the button is help to make a box	Verify that the group of units are highlighted when they are selected			

around the units.				
17. Hover over one of the units	Verify that a pop-up window appears with information on the unit			
18. Issue a move order to the units	Verify that a message appears acknowledging that the order was received			
19. Wait for units to reach location	Verify that a message appears verifying that the order was fulfilled			
20. Issue an attack order against enemy units	Verify that a message appears acknowledging that the order was received			
21. Wait for units to start attacking and are being attacked in return	Verify that a message appears alerting the player that their units are under attack			
22. Order the units to stop attacking and let them be killed	Verify that a message appears notifying the player that their units have died			
23. Choose a field commander race	Verify that the commander selection disappears and no messages appear to remind you to select a field commander			
24. Enter field commander mode and approach one of the bases for a race other than your field commander	Verify that a message appears asking if you want to switch your field commander race			
25. Switch into tactical	Verify that you can see a view of your			

commander role	commander in the interface			
Switch to the Piscivian faction				
26. Order units to attack the Human field commander	Verify that a message appears alerting the Human faction that their commander is under attack			
27. Wait until the Human field commander is killed	Verify that a message appears alerting the Human faction that their commander has been killed			
28. Press ENTER to open chat window	Verify that chat window opens			
29. Wait until a message appears	Verify that a message appears reminding the player to move idle units			
30. Continue to wait until next message appears	Confirm that a message appears reminding the player to construct a building			
31. Continue to wait until next message appears	Verify that a message appears reminding the player to choose a field commander race			
32. Hover over an open plot around any base	Verify that the plot highlights to indicate it is able to be built upon			
33. Select an open plot	Verify that three icons appear for each of the available buildings			
34. Hover over the three icons	Verify that a pop-up window appears with information on each building			

35. Select a building and start construction	Verify that the building starts constructing and a message appears confirming the choice			
36. Wait until building is complete	Verify that a message appears telling the player that the building is complete			
Switch to the Human faction				
37. Select a unit and issue an order to attack the Piscivian building	Verify that a message appears alerting the Piscivian faction that their building is under attack			
38. Issue an order to attack a Piscivian HQ	Verify that a message appears alerting the Human faction that their base is under attack			
Switch to the Piscivian faction				
39. Select a group of units by left clicking and dragging white the button is help to make a box around the units.	Verify that the group of units are highlighted when they are selected			
40. Hover over one of the units	Verify that a pop-up window appears with information on the unit			
41. Issue a move order to the units	Verify that a message appears acknowledging that the order was received			
42. Wait for units to reach location	Verify that a message appears verifying that the order was fulfilled			
43. Issue an attack order against	Verify that a message appears			

enemy units	acknowledging that the order was received			
44. Wait for units to start attacking and are being attacked in return	Verify that a message appears alerting the player that their units are under attack			
45. Order the units to stop attacking and let them be killed	Verify that a message appears notifying the player that their units have died			
46. Choose a field commander race	Verify that the commander selection disappears and no messages appear to remind you to select a field commander			
47. Enter field commander mode and approach one of the bases for a race other than your field commander	Verify that a message appears asking if you want to switch your field commander race			
48. Switch into tactical commander role	Verify that you can see a view of your commander in the interface			
Switch to the Human faction				
49. Order units to attack the Piscivian field commander	Verify that a message appears alerting the Piscivian faction that their commander is under attack			
50. Wait until the Piscivian field commander is killed	Verify that a message appears alerting the player that their commander has died			

Appendix Z: Ad Hoc Testing Result Database

Bug Description	Number of Suggestions	Category	Type	Priority
Bombship Depth charge crashed game	4	Technical	Bug	Critical
Units ordered to attack a building crashes the game	1	Technical	Bug	Critical
Cannot click on bases or buildings to display name and health	3	Technical	Bug	High
Need to be clearer to player that he can only build one thing at a time	1	Help	Desired	High
Units wandered around when clicking to attack the enemy base	1	Technical	Bug	High
Remove building options that do not do anything on the building menu	1	Technical	Bug	High
Client cannot select a new field commander when the first is killed	6	Technical	Bug	High
Clicking on the help area minimizes the game	1	Technical	Bug	High
Building Piscivian guard tower prevents the building of other buildings	1	Technical	Bug	High

More obvious when units are attacking/being attacked	9	Technical/Art	Desired	High
Middle click on plots is not intuitive	4	Technical	Desired	High
More obvious where projectile is going to land	1	Technical	Desired	High
Tooltips for unit abilities and unit stats	3	Help	Desired	High
Commander is dead...need more feedback for when/if he is going to respawn	2	Help/Technical	Desired	High
Honeycomb Coral, Pirate Ship, and Human Farm show up blue	3	Art	Bug	High
Building creation not synced on the server	1	Technical	Bug	High
Buildings and plots are not destroyed when a headquarters is destroyed	1	Technical	Bug	Medium
The client should not be able to Start the game	1	Technical	Bug	Medium
More obvious if commander is hitting with abilities	4	Technical	Desired	Medium
There should be a health bar above the units	4	Technical	Desired	Medium

Control scheme is different when in different modes (USE WASD or Arrows, not both)	6	Technical/Help	Desired	Medium
Would like sound to play when message appears or window to flash	1	Help/Technical	Desired	Medium
Clearer distinction between units and their archetypes	2	Help	Desired	Medium
Unclear how to make units move/attack	1	Technical/Help	Desired	Medium
Music should be louder in the game	2	Art/Technical	Bug	Medium
Building help should stop if all buildings are full	1	Technical/Help	Bug	Medium
Not intuitive that bases will automatically spawn units	1	Help	Desired	Medium
Attack animation does not play every time unit attacks	1	Technical/Art	Bug	Medium
Turtle and Bomb ship are too slow	4	Balancing	Bug	Medium
Piscivians seem to have advantage	2	Balancing	Bug	Medium
Option menu appears during loading screen if previously opened	1	Technical	Bug	Low
Gray box on mini-map does not represent the position of the camera	1	Technical	Bug	Low

At zero health commander is “dead” but model does not disappear	1	Technical	Bug	Low
Remove little Xs on commander health bar	1	Technical	Bug	Low
Cannot move camera using the mini-map	6	Technical	Desired	Low
Move commander from RTS mode, annoying to switch back and forth to “save” commander	2	Technical	Desired	Low
Hard to find which unit is the commander	2	Technical/Art	Desired	Low
Button to select all units (moving entire army is difficult)	2	Technical	Desired	Low
Way for units to follow the commander	1	Technical	Desired	Low
Make map more interesting with obstacles	1	Art	Desired	Low
Unknown that ESC brings up the menu	1	Help	Desired	Low
Stats are very small text...also hard to read on the human side	1	Art	Bug	Low
LARGER Start/Exit buttons in game lobby	1	Art	Bug	Low
Reposition bases to not be in a straight line...map is boring	1	Art	Desired	Low

Indicator about how long it will take to spawn next wave of units	1	Technical/Help	Desired	Low
Want way to group units into teams with CTRL+1, CTRL+2	1	Technical	Desired	Low
Confusing whether large floating mines do anything	1	Art	Bug	Low
Japanese planes too large	1	Art	Bug	Low
Units blend too much with the environment (red rings around enemy)	1	Technical	Desired	Low
Loading screen goes by too quickly to read all that text...simpler versions	3	Art/Help	Bug	Low
Sub base disappeared (in two test, once for host and once for client) <- noticed after field commander died	2	Technical	Bug	Unable to Reproduce
Kicking players in game lobby does not work	1	Technical	Bug	Unable to reproduce

Appendix AA: *Blood Tide* Help Effectiveness Test Plan

Player: _____ Moderator: _____

Overall confidence that skills are learned for tactical-commander role
(1 – not confident, 5 – extremely confident): _____

Overall confidence that skills are learned for field-commander role
(1 – not confident, 5 – extremely confident): _____

Overall confidence that skills are learned to play *Blood Tide*
(1 – not confident, 5 – extremely confident): _____

Tutorial Level

Tutorial Level Help		Rating from 1 to 5 (1 – lowest, 5 – highest)
<i>Tutorial Help</i>	Help does not hinder gameplay	
	Help clearly demonstrates game skills	

Mentor Characters

Mentor Character Help		Rating from 1 to 5 (1 – lowest, 5 – highest)
<i>Mentor Help during Tutorial</i>	Help does not hinder gameplay	
	Help clearly demonstrates game skills	

Pop-Up Messages

Pop-Up Message Help		Rating from 1 to 5 (1 – lowest, 5 – highest)
<i>Loading Screen</i>	Help does not hinder gameplay	
	Help clearly demonstrates game skills	
<i>Help Window</i>	Help does not hinder gameplay	
	Help clearly demonstrates game skills	
<i>Game Controls Window</i>	Help does not hinder gameplay	
	Help clearly demonstrates game skills	

Interface Help

Field-Commander Interface		Rating from 1 to 5 (1 – lowest, 5 – highest)
<i>Help Window</i>	Help does not hinder gameplay	
	Help clearly demonstrates game skills	
<i>Commander Status Window</i>	Help does not hinder gameplay	
	Help clearly demonstrates game skills	
<i>Mini-Map Window</i>	Help does not hinder gameplay	
	Help clearly demonstrates game skills	

Tactical-Commander Interface		Rating from 1 to 5 (1 – lowest, 5 – highest)
<i>Chat Window</i>	Help does not hinder gameplay	
	Help clearly demonstrates game skills	
<i>Unit Selection Window</i>	Help does not hinder gameplay	
	Help clearly demonstrates game skills	
<i>Mini-Map Window</i>	Help does not hinder gameplay	
	Help clearly demonstrates game skills	

Building Interface		Rating from 1 to 5 (1 – lowest, 5 – highest)
<i>Total Building Interface</i>	Help does not hinder gameplay	
	Help clearly demonstrates game skills	

Appendix AB: *Blood Tide* Help Effectiveness Coding Sheet Results for Playing Skipping Tutorial

Player: ChanLing Beswick Moderator: Sarah Gilkey

Overall confidence that skills are learned for tactical-commander role
(1 – not confident, 5 – extremely confident): 4

Overall confidence that skills are learned for field-commander role
(1 – not confident, 5 – extremely confident): 4

Overall confidence that skills are learned to play *Blood Tide*
(1 – not confident, 5 – extremely confident): 4

Tutorial Level

- N/A for this test

Mentor Characters

- N/A for this test

Pop-Up Messages

Pop-Up Message Help		Rating from 1 to 5 (1 – lowest, 5 – highest)
<i>Loading Screen</i>	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	1
<i>Help Window Pop-ups</i>	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	5
<i>Game Controls Window</i>	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	4

Field-Commander Interface (3rd person chase camera)		Rating from 1 to 5 (1 – lowest, 5 – highest)
<i>Help Window</i>	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	5
<i>Commander Status Window</i>	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	5
<i>Mini-Map Window</i>	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	5

Tactical-Commander Interface (birds-eye camera)		Rating from 1 to 5 (1 – lowest, 5 – highest)
<i>Help Window</i>	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	5
<i>Unit Selection Window</i>	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	5
<i>Mini-Map Window</i>	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	5

Building Interface		Rating from 1 to 5 (1 – lowest, 5 – highest)
<i>Total Building Interface</i>	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	3

Notes:

1. Controls are wrong, backwards should be “s”, not “d” as shown.
2. Turtles are difficult to beat.
3. Field motion a little slow, units may be dead before you can catch up.
4. Sound too soft.
5. Being able to move with your tactical units in first person view is very cool.

Player: Dan Lampke Moderator: Sarah Gilkey

Overall confidence that skills are learned for tactical-commander role
(1 – not confident, 5 – extremely confident): 3.5

Overall confidence that skills are learned for field-commander role
(1 – not confident, 5 – extremely confident): 2

Overall confidence that skills are learned to play *Blood Tide*
(1 – not confident, 5 – extremely confident): 3

Tutorial Level

- N/A for this test

Mentor Characters

- N/A for this test

Pop-Up Messages

Pop-Up Message Help		Rating from 1 to 5 (1 – lowest, 5 – highest)
Loading Screen	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	3
Help Window	Help does not hinder gameplay	4
	Help clearly demonstrates game skills	2
Game Controls Window	Help does not hinder gameplay	4
	Help clearly demonstrates game skills	4

Interface Help

Field-Commander Interface		Rating from 1 to 5 (1 – lowest, 5 – highest)
Help Window	Help does not hinder gameplay	4
	Help clearly demonstrates game skills	2
Commander Status Window	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	4
Mini-Map Window	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	5

Tactical-Commander Interface		Rating from 1 to 5 (1 – lowest, 5 – highest)
Chat Window	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	N/A
Unit Selection Window	Help does not hinder gameplay	4
	Help clearly demonstrates game skills	N/A
Mini-Map Window	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	N/A

Building Interface		Rating from 1 to 5 (1 – lowest, 5 – highest)
Total Building Interface	Help does not hinder gameplay	3
	Help clearly demonstrates game skills	2

Notes

- Help Window does not hinder, but it's not extremely intuitive to see any help that does pop up during action. A notification sound, or some sort of visual mechanism may alert the player that there is a new piece of advice.

Player: Amanda Moreno Moderator: Joseph Alea

Overall confidence that skills are learned for tactical-commander role
(1 – not confident, 5 – extremely confident): 1

Overall confidence that skills are learned for field-commander role
(1 – not confident, 5 – extremely confident): 3

Overall confidence that skills are learned to play *Blood Tide*
(1 – not confident, 5 – extremely confident): 2

Tutorial Level

- N/A for this test

Mentor Characters

- N/A for this test

Pop-Up Messages

Pop-Up Message Help		Rating from 1 to 5 (1 – lowest, 5 – highest)
Loading Screen	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	3-5
Help Window	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	3
Game Controls Window	Help does not hinder gameplay	4
	Help clearly demonstrates game skills	4

Field-Commander Interface		Rating from 1 to 5 (1 – lowest, 5 – highest)
<i>Help Window</i>	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	5
<i>Commander Status Window</i>	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	5
<i>Mini-Map Window</i>	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	3

Tactical-Commander Interface		Rating from 1 to 5 (1 – lowest, 5 – highest)
<i>Chat Window</i>	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	2
<i>Unit Selection Window</i>	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	5
<i>Mini-Map Window</i>	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	3

Building Interface		Rating from 1 to 5 (1 – lowest, 5 – highest)
<i>Total Building Interface</i>	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	2

Notes

- Loading screen should have a “Next” button or equivalent when player is done reading it/taking it in. Also, could the objective be more bold. Maybe make it surrounded by a bolded box/border. A different color would be helpful.
- Help window explained fairly clearly, but the controls were confusing/counter-intuitive.

- It would be helpful if the controls window was easier to find. Maybe a drop down menu. Also, short keys would be very nice! For example, pressing escape would exit out of the help window. Additionally, after I clicked the “X” button (to exit out of window), I was taken to a new window. I was very confused about how to get back to the game. It would be nice to have a “Return to Game” option.
- Number pad should be able to substituted for the normal number keys.
- Can there be instructions to clarify the minimap. Where it shows who you are, who your enemy is, etc?
- Choosing commander should be more obvious since it is very necessary. Perhaps make person decide before being allowed to do anything else (centered window)
- Can it please explain what the building interface does? I was confused about what a plot is. I may have been quicker if it showed me a visual image or highlighted the ones on the map.

Appendix AC: *Blood Tide* Help Effectiveness Coding Sheet Results for Playing Tutorial

Player: Mike Oshetsky Moderator: Sarah Gilkey

Overall confidence that skills are learned for tactical-commander role
(1 – not confident, 5 – extremely confident): 4

Overall confidence that skills are learned for field-commander role
(1 – not confident, 5 – extremely confident): 4

Overall confidence that skills are learned to play *Blood Tide*
(1 – not confident, 5 – extremely confident): 4

Tutorial Level

Tutorial Level Help		Rating from 1 to 5 (1 – lowest, 5 – highest)
Tutorial Help	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	5

Mentor Characters

Mentor Character Help		Rating from 1 to 5 (1 – lowest, 5 – highest)
Mentor Help during Tutorial	Help does not hinder gameplay	4
	Help clearly demonstrates game skills	4

Pop-Up Messages

Pop-Up Message Help		Rating from 1 to 5 (1 – lowest, 5 – highest)
Loading Screen	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	4
Help Window	Help does not hinder gameplay	4
	Help clearly demonstrates game skills	4
Game Controls Window	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	5

Interface Help

Field-Commander Interface		Rating from 1 to 5 (1 – lowest, 5 – highest)
Help Window	Help does not hinder gameplay	4
	Help clearly demonstrates game skills	4
Commander Status Window	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	5
Mini-Map Window	Help does not hinder gameplay	3
	Help clearly demonstrates game skills	5

Tactical-Commander Interface		Rating from 1 to 5 (1 – lowest, 5 – highest)
Chat Window	Help does not hinder gameplay	4
	Help clearly demonstrates game skills	4
Unit Selection Window	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	5
Mini-Map Window	Help does not hinder gameplay	4
	Help clearly demonstrates game skills	4

Building Interface		Rating from 1 to 5 (1 – lowest, 5 – highest)
Total Building Interface	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	5

Notes:

- Turtles are too difficult to defeat with only scuba troops
- The control scheme switching between arrow and WASD keys is confusing

Player: Nike Holovnia Moderator: Joseph Alea

Overall confidence that skills are learned for tactical-commander role
(1 – not confident, 5 – extremely confident): 5

Overall confidence that skills are learned for field-commander role
(1 – not confident, 5 – extremely confident): 3

Overall confidence that skills are learned to play *Blood Tide*
(1 – not confident, 5 – extremely confident): 4

Tutorial Level

Tutorial Level Help		Rating from 1 to 5 (1 – lowest, 5 – highest)
Tutorial Help	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	4

Mentor Characters

Mentor Character Help		Rating from 1 to 5 (1 – lowest, 5 – highest)
Mentor Help during Tutorial	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	3

Pop-Up Messages

Pop-Up Message Help		Rating from 1 to 5 (1 – lowest, 5 – highest)
Loading Screen	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	5
Help Window	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	4
Game Controls Window	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	5

Field-Commander Interface		Rating from 1 to 5 (1 – lowest, 5 – highest)
<i>Help Window</i>	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	3
<i>Commander Status Window</i>	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	4
<i>Mini-Map Window</i>	Help does not hinder gameplay	4
	Help clearly demonstrates game skills	3

Tactical-Commander Interface		Rating from 1 to 5 (1 – lowest, 5 – highest)
<i>Chat Window</i>	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	5
<i>Unit Selection Window</i>	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	5
<i>Mini-Map Window</i>	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	5

Building Interface		Rating from 1 to 5 (1 – lowest, 5 – highest)
<i>Total Building Interface</i>	Help does not hinder gameplay	4
	Help clearly demonstrates game skills	4

Notes:

- It would be nice to be able to command all of the units
- Also, it would be nice to be able to construct all of the buildings
- It was unclear at the beginning that units would spawn automatically

Player: Shanna Infantino Moderator: Sarah Gilkey

Overall confidence that skills are learned for tactical-commander role
(1 – not confident, 5 – extremely confident): 5

Overall confidence that skills are learned for field-commander role
(1 – not confident, 5 – extremely confident): 4

Overall confidence that skills are learned to play *Blood Tide*
(1 – not confident, 5 – extremely confident): 4

Tutorial Level

Tutorial Level Help		Rating from 1 to 5 (1 – lowest, 5 – highest)
Tutorial Help	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	4

Mentor Characters

Mentor Character Help		Rating from 1 to 5 (1 – lowest, 5 – highest)
Mentor Help during Tutorial	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	5

Pop-Up Messages

Pop-Up Message Help		Rating from 1 to 5 (1 – lowest, 5 – highest)
Loading Screen	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	4
Help Window	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	5
Game Controls Window	Help does not hinder gameplay	3
	Help clearly demonstrates game skills	3

Field-Commander Interface		Rating from 1 to 5 (1 – lowest, 5 – highest)
<i>Help Window</i>	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	4
<i>Commander Status Window</i>	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	4
<i>Mini-Map Window</i>	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	5

Tactical-Commander Interface		Rating from 1 to 5 (1 – lowest, 5 – highest)
<i>Chat Window</i>	Help does not hinder gameplay	4
	Help clearly demonstrates game skills	4
<i>Unit Selection Window</i>	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	5
<i>Mini-Map Window</i>	Help does not hinder gameplay	4
	Help clearly demonstrates game skills	4

Building Interface		Rating from 1 to 5 (1 – lowest, 5 – highest)
<i>Total Building Interface</i>	Help does not hinder gameplay	5
	Help clearly demonstrates game skills	3

Notes:

- I wasn't sure if the building was actually doing anything
- It took me a while to understand the controls window and new messages kept appearing in the help window