**CIS319 – Aim to Learn**

**Software Requirements Specification***Adapted from Long Software*

**Version 3.2**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Authors** |
| 6/21/2017 | 1.0 | Initial requirements discussed from initial planning. | Justin Mariner, Emir Kaynak, Rebekah Marsh |
| 6/28/2017 | 1.1 | Fixed main title and added age range. |
| 7/3/2017 | 2.0 | Lowered age range to elementary school |
| 7/13/2017 | 3.0 | Added info on “how to play” menu, and the ship invulnerability effect |
| 7/13/2017 | 3.1 | Added project leader-specific questions to the end of the document. | Justin Mariner |
| 7/18/2017 | 3.2 | Mentioned new in-game tutorial that has been added |
|  |  |  |  |

# Executive Summary

This document is the Requirements Specification document for the CIS319 Case Study for an educational game. It provides detailed descriptions of the software, user, and hardware interfaces of the system, and includes a detailed description of the user interface for the system.

The intention of the game is to allow the users to learn basic general education subjects through an involved and engaging video game. Students will be able to improve both general education, through the content of the video game; and coordination and reaction time, through the video game itself.

The objective of the Software Requirements Specification is to provide a summation of the findings thus far in the development stage of the project. It will be treated as a working document and provides a detailed outline of the system from the client's perspective.

# Table of Contents

1. Project Description ........................................................................ Page 4
   1. Purpose
   2. Scope
2. Overall Game Description ............................................................... Page 4
   1. Product Perspective
   2. User Characteristics
   3. Constraints
   4. Assumptions and Dependencies
3. Specific Requirements ................................................................... Page 6
   1. Usability
   2. Reliability
   3. Online User Documentation and Help System Requirements
   4. Interfaces

## 1. Project Description

The Three Softapeers custom software development services. We can handle all the developmental aspects of this case study including specification, architecture & design, implementation, testing, maintenance and training. We always involve our clients in each and every aspect of the software development process by providing prototypes and encouraging feedback. This helps ensure that our clients are highly satisfied with the end product.

### 1.1 Purpose

This document is a proposal for requirement specifications for the CIS319 Case Study and Request for Proposal posted on June 19, 2017. The Three Softapeers is very excited about the idea of the game and prospects of working on its development.

### 1.2 Scope

The requirements specified in this document will be used for designing all the aspects and components of the game and is a working document. The document will be updated and maintained by the project leader as the requirements grow and change over the design and development process.

## 2. Overall Game Description

### 2.1 Product Perspective

#### 2.1.2 User Interfaces

The interface for the students will be entertaining and engaging. The function of the buttons will be easy to understand and simple to use. Menus will be interactive and easily accessible throughout the game. Once the game is in playing mode, everything a player/student needs will be clearly visible on the screen and easily accessible. Students will the find the most basic functions of the game, such as educational exercises, fun to play.

#### 2.1.3 Hardware Interfaces

* The product is required to operate on both Macintosh and Windows systems. The mouse will only be used for menu selections requiring one mouse button, so the product will run the same on either system.
* The keyboard will also play an integral role in the student's interaction with the game. Player movement will be controlled with the keyboard alone; the mouse will play no part in gameplay.
* Since the game will have a large amount of text on the screen, it is best viewed on a high-resolution monitor of at least a resolution of 1280x720. The amount of content will also be limited to ensure that the total size of the game remains under 80mb.

#### 2.1.4 Operations

The game will provide the following minimal operations:

* Teach the subjects of Math, Science, and History to high school and college students in an entertaining and engaging manner.
* Provide functionality for progress tracking.
* Provide user interface and controls for the targeted audience.
* Provide difficulty levels to cater the skill level of the users.

### 2.2 User Characteristics

* Game is targeting students in elementary school, ranging in ages from 6 to 12.
* These students will definitely have access to a computer while at their school or on campus, and will likely have computer access at home as well.
* It is safe to assume these students will have sufficient computer literacy to play and interact with the game.
* High school and college students will have had the required general education to answer the questions posed by the game.

### 2.3 Constraints

The following constraints are specified in the RFP:

* Platform independence is necessary since each school may have a different OS.
* Because a lot of schools are using older systems this game should run on a system with these requirements:
  + **Windows 7 and newer**: Pentium 60, 16 MB ram, 80 MB HDD space, Mouse, SVGA video card, 2x, or better, speed CD-ROM, DirectX compatible sound card.
  + **Mac OS 7.6 and up:** 16 MB ram, 80 MB HDD space, Mouse, SVGA video card, 2x, or better, speed CD-ROM, Monitor Requires 640x480, 256 color
* Testing will be necessary, but as the designers are also the intended audience, much of the testing will be done internally. Outside users will be found to test the game as well.
* The designers will communicate with high school and college students in order to ensure the product is as engaging as possible for their level of education.

### 2.4 Assumptions and Dependencies

In order to finish the project on time, the documentation must be reviewed and signed off within one week of the delivery of our deliverables. The following are the deliverables deadlines:

|  |  |
| --- | --- |
| June 21, 1pm | Requirements document submitted |
| June 26, 1pm | Use case & interactive diagrams, and storyboard, submitted |
| June 26 | Review the Design among the Project Team and Divide Tasks |
| June 28 | Create the system |
| July 10 | Continue work on the system |
| July 12 | Continue work and begin testing |
| July 17 | Test more and debug |
| July 19, 9am | Final submissions and presentation |

## 3. Specific Requirements

In this section, we will specify detailed requirements for the game. Our designers and programmers will design and build the game based on these requirements. Throughout section 3, requirements are indicated as functional or non-functional by the symbol (F) or (NF) respectively in the requirement heading.

### 3.1 Usability

This section addresses the software usability requirements for the students and teachers laid out in UCSE’s RFP.

*3.1.2 (NF) The system will provide in-game tutorials and help.*

The system is intended for use in classes, so the game will need to provide support for students that need help. A single teacher is not be able to help all students all the time.

*3.1.3 (F) Maximum time from launching the game until it is playable will be 5 minutes.*

After the application is launched, it will take fewer than 5 minutes for the player to load their data and begin playing the game from where they last left off. This will help to ensure that children do not lose interest.

*3.1.4 (NF) Familiar user interface provided for high school and college students.*

The user interfaces for the game itself (including in-game menus) will be similar to those found in typical GUI applications. “Similar” here refers to how the menu is accessed, its appearance and how it reacts to input.

### 3.2 Reliability

*3.2.1 (NF) The software will be able to run 99% of the time when launched.*

There is a potential for errors relating to the state of the operating system that could prevent the game from launching (for example not enough resources available, etc.). The chance of such an occurrence is at most 1%.

#### 3.2.2 (NF) Expected system uptime will be 95%

The game itself will be able to run for at least 3 consecutive hours, 95% of the time.

#### 3.2.4 (F) The system will not be prone to errors caused by unexpected input

The software will be able to handle all sorts of input and be immune to side effects caused by undesirable inputs (such as buffer overflow), which could potentially create security holes in the system.

#### 3.2.5 (NF) The system will maintain network security

The software will employ appropriate network security protocols to ensure that it doesn’t create network security problems.

### 3.3 Online User Documentation and Help System Requirements

The game will have no form on online documentation; all instructions and information will be provided from within the game itself. The game will also contain an interactive tutorial that directs the user through the basic controls and gameplay.

### 3.4 Interfaces

#### 3.7.1 (F) User Interfaces

* The first screen the user will be shown is the main menu screen, which will contain several buttons: Start Game, Options, How to Play, and Quit Game.
  + Start Game and Quit Game will function as their names imply. Quitting will prompt the user for confirmation of their action.
  + “How to Play” is also self-explanitory - it will show information describing how the game is played
  + Options will open a secondary menu containing various game configuration options, including:
    - Volume sliders for main volume, sound effects, and background music
    - Option to set resolution to various pre-set window sizes.
    - Some other various gameplay configuration options.
* The gameplay screen, shown by pressing Start Game, will have the following specifications:
  + Dark, solid background
  + Contrasting (white) text color
  + Current question the player must answer printed at the top of the window, very clearly.
  + Many possible answers to the posed question, falling downward from the top of the screen towards the player.
  + The player’s ship that can move along the bottom of the screen
  + A visual effect around the ship to indicate momentary invulnerability.
* When first starting the game, the user will be prompted with a gameplay world where they must shoot at the difficulty and subject options that they want to play at. This simply acts as a more engaging menu and is only used for simple selections.
* Once in game, there will also be a pause menu that is accessed by pressing the escape key. Opening this menu will halt all game movement as long as it is active.
  + This menu will also contain the Options menu, with some options possibly marked as unavailable since the user is currently in-game.
  + There will also be a Main Menu button, which will, after prompting the user for confirmation, return the user to the initial menu.

#### 3.7.2 (NF) Hardware Interfaces

The gaming software does not require any additional or specialized hardware in order to operate. Existing hardware such a keyboard and mouse will be the only hardware required for input to the game.

#### 3.7.3 (NF) Software Interfaces

The game will access the file system of the machine it is running on in order to access local data and configuration files. It will not require use of any network connections.

## 4. Project Leader Extra Questions

### What each group member is working on

Justin - Code

Rebekah - Sound and data

Emir - Graphics and animations

### The Lens of The Team

#### (1) Is this the right team for the project? Why?

Yes, this team is right for this project because each has their own strengths that work well together, and each knows the extents their strengths, which has been used to determine the extensivity of this project.

#### (2) Is the team communicating objectively?

Yes, all team members have always had the goal of creating a complete game as the first thought when making decisions and communicating.

#### (3) Is the team communicating clearly?

Yes, team members are able to communicate opinions and ideas with clarity.

#### (4) Is the team comfortable with each other?

Yes, all team members have been comfortable working with each other, even taking time during the July 4th break to come in and work on the project.

#### (5) Is there an air of trust and respect amongst the team?

Yes, each team member trusts each other’s opinions and respects the work each other does on this project.

#### (6) Is the team ultimately able to unify around decisions?

Yes, group members have always been able to unify around the original idea of keeping this project simple enough to fully complete within the deadline.

### 

### The Lens of Fun

#### (1) What parts of the game are fun?

The shooting and dodging parts of the gameplay are pretty fun since there is a slight element of skill involved.

#### (2) What parts need to be more fun?

The actual flow of the overall gameplay, such as progressing between rounds, levels, and difficulty, could use some work to be made more fun.