**CIS319 - Educational Game**

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

**Version 1.0**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# 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 …

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. Overall Game Description
   1. Add subtitles and page numbers
   2. Add subtitles and page numbers
   3. Add subtitles and page numbers
2. Specific Requirements
   1. Add subtitles and page numbers
   2. Add subtitles and page numbers
   3. Add subtitles and page numbers

## 1. Overall Game Description

(Your team name here) 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 (your team name) 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 **(replace example content below)**

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 fun to play, from character creation to the educational exercises.

#### 2.1.3 Hardware Interfaces **(replace example content below)**

The product is required to operate on both Macintosh and Windows systems. As such, the game should be able to adjust to one button mouse input or two button mouse input depending on the system it’s running on.

The keyboard will also play an integral role in the student's interaction with the game. Answers to some questions will have to be typed in, and the character movement will be accomplished using either the mouse or the keyboard.

It will be possible to transfer the students results to a server with any standard networking hardware.

The graphical content will be at most 256 colors at a resolution of 640x480. This will allow for the game to be played on older computers commonly found in elementary schools. The amount of graphical content will also be limited to ensure that the total size of the game remains under the 80mb limit.

#### 2.1.4 Operations **(replace example content below)**

The game will provide the following minimal operations ***(replace example content below):***

* Teach Math and English to grade levels 1 & 2 in an entertaining and engaging manner.
* Provide functionality for teachers to track and evaluate student progress.
* Provide user interface and controls for the targeted audience.
* Provide difficulty levels to cater the skill level of the users.

### 2.2 User Characteristics (*replace example content below)*

This game is targeted directly towards children attending Grades 1 and 2. These grades typically correspond to ages ranging from 6 to 8 years. At this stage in school, children are expected to have rudimentary reading and math skills. Children in this age range tend to have very short attention spans, and great effort must be put in to maintaining their interest. They also tend to be very curious, and may inadvertently cause problems as a result.

Most children probably have had access to computers at home, but this cannot be guaranteed, so some children may not have the computer skills necessary to operate the game without guidance. However, they are generally very good at following patterns and can be taught fairly easily.

Starting with ages between 6-8 years, girls and boys show very different characteristics from each other. They will have very different interests and sources of entertainment: a factor that cannot be ignored. Also, girls tend to be more mentally advanced and developed than boys.

Because the game is intended for in school use, teachers will be administrating the game, and must also be considered. Teachers represent a much more varied age range and background than children, but it can be assumed that they will have basic computer skills, such as word and spreadsheet processing, e-mail and web browsing.

### 2.3 Constraints *(replace example content below)*

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
* It will be necessary to test the game on children to ensure that it is entertaining and easy to use.
* The designers will coordinate with teachers, parents, children entertainers, child psychologists and other educators in the development of the software. This is to ensure that the subject matter and educational material is appropriate for the students.

### 2.4 Assumptions and Dependencies *(replace example content below)*

We assume the following responsibilities from the client during the game development process:

* Provide testing methods and help in developing the test cases with our test engineers.
* Upon the completion of the development process, organize meetings and workshops with the target user groups to test the software.

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: **(Add dates – what will be delivered)**

|  |  |
| --- | --- |
| January 26 | Meeting to discuss RFP |
| January 30 | RS 1.0 |
| February 6 | RS 1.1 |
| February 16 | Prototype Demo |
| March 2 | Requirements Negotiations |
| March 6 | RS 2.0 |
| March 10 | RS 2.1 |
| April 3 | RS 3.0 |
| March 30 – April 6 | Final Demo |

## 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.

*3.1.1 (F) The system will test basic computer abilities prior to beginning the game, and provide the necessary tutorials.*

Basic computer abilities include mouse and keyboard input, and understanding computer displays and feedback (alerts, menus etc). The system will provide a short test to determine the computer skills that the student is lacking, and will then provide tutorials in required areas.

*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 children.*

The user interfaces for the game itself (including load screens, and in-game menus) will be similar to those found in other children’s games. “Similar” here refers to how the menu is accessed, its appearance and how it reacts to input. Menus will also use animations and colors to attract and maintain attention to important items. The game interface will be tested alongside other children’s games to ensure that the interface is usable.

#### 3.1.5 (NF) Familiar user interface provided for teachers

The interface for queries will be similar to a web-browser, with regards to navigation and file access. Navigation will use links to follow as well as forward and back buttons. Queries will be made using html-style forms and downloadable files. Responses to system querieswill be exported to Microsoft Office (i.e. Word and Excel) formatted documents, providing a familiar interface for data manipulation.

*3.1.6 (NF) Long Stretch Software will work with SME’s to ensure that the system will be*

#### usable by all parties

We will collaborate with teachers, child psychologists and other educators, as well as children to ensure that the system will be useable for people of all experience levels.

### 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. The back-end database should be able to support up to 100 connections of the game clients, 95% of the time.

#### 3.2.3 (NF) Online backup will be provided by the system

The software will employ mechanisms to ensure that there is no data loss if the game crashes.

#### 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 cause 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 *(replace example content below)*

Online documentation will be available for the game upon completion and delivery. The game will also come with built in help files and tutorials for game play.

### 3.4 Interfaces (*replace example content below)*

#### 3.7.1 (F) User Interfaces

The first screen is the login screen. It will feature a username and password string entry, and submit and exit buttons. The user will be notified by a dialogue box in case of an incorrect entry.

Once the user has signed on, the main menu screen will be presented. The main menu will have three buttons: New Game; Load Game; and, Quit Game. All buttons in the preceding and subsequent menus will have a similar style: energetic, entertaining, but also easy to understand and use. The buttons will provide the following functionality:

* The function of any Quit Game button will be the closing of the game.
* The Load Game button will load the user to the position in the game they were before their last sign off.
* The New Game button will open the new game menu, where the character is selected. On this screen, there will be an image of whatever character is selected; as well as a scrolling method to switch characters, *and* *continue* and *back* buttons for progression through the menus.

Once a character is selected and *continue* is pressed, the user will choose his or her difficulty level and the game will commence.

The game-play screen will have one bar along the bottom. The style of the bar is dependent on what character was chosen by the user: one style for boys and one for girls.

There will be three basic functions along the bottom bar: character, skills, and map.

* The character function will show the player their character at that point in time.
* The skills section will show the player what their skills are at the three main educational types: math, English, and problem solving.
* The map function will bring up a map of the game area, showing the player where they have been before, what areas they have completed the exercises or lessons in, and what the areas are in any direction from them.

A fourth in-game menu is accessed by pressing the ESC button. In this menu are some basic game-play options such as text scroll speed and text size, and the Quit Game button which exits the game, while also saving the current state of the game.

#### 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 gaming, administration and database components of the software will communicate securely via a local network. All connections will be adequately encrypted, while still meeting the