

Table of Contents

1. Introduction	1
1.1 Purpose	1
1.2 Document Conventions	1
1.3 Intended Audience and Reading Suggestions	1
1.4 Product Scope	1
1.5 References	1
2. Overall Description	2
2.1 Product Perspective	2
2.2 Product Functions	2
2.3 User Classes and Characteristics	2
2.4 Operating Environment	3
2.5 Design and Implementation Constraints	3
2.6 User Documentation	3
2.7 Assumptions and Dependencies	4
3. External Interface Requirements	4
3.1 User Interfaces	4
3.2 Hardware Interfaces	4
3.3 Software Interfaces	4
3.4 Communications Interfaces	5
4. System Features	5
4.1 Character Management	5
4.2 Interactive Map	19
4.3 Interactive NPCs	23
5. Other Nonfunctional Requirements	26
5.1 Performance Requirements	26
5.2 Safety Requirements	26
5.3 Security Requirements	26
5.4 Software Quality Attributes	26
5.5 Business Rules	27
6. Other Requirements	27

Revision History

Name	Date	Reason For Changes	Version
SRS for 3D Educational Game for Dyslexic Students	1/5/2023	Initial Version of Document	1.0

1. Introduction

1.1 Purpose

The purpose of this document is to collect, analyze, and define high-level needs and features of the Interactive Educational Game. It focuses on the capabilities needed by the stakeholders, and the target users, and why these needs exist. This document is used to define how the features will be implemented in the final game using the means of requirements of each feature. The details of how the Interactive Educational Game fulfills these needs will be detailed in the use-case and supplementary specifications.

1.2 Document Conventions

Nouns with unclear are shown in bold and have added definition in glossary. Priority for each requirement is given in the requirement description. Low level requirements have additional numbering to them, e.g. for requirement 1.1 further low level requirements will be as 1.1.1, and so on.

1.3 Intended Audience and Reading Suggestions

This Document is intended for the game developers and designers to understand what they are required to design and implement. This is done through detailed descriptions of the game itself and also what external interfaces requirements are needed to be followed. The features that this game will possess have also been specified in this document so that the developers know what things that are to be implemented.

1.4 Product Scope

This Vision document applies to the Interactive Educational Game (IEG), which will be developed by DBT Development Team. The Game will support the ability to connect to the Internet to allow Multiplayer with people from all over the world while also having the ability to play it without any internet connection.

1.5 References

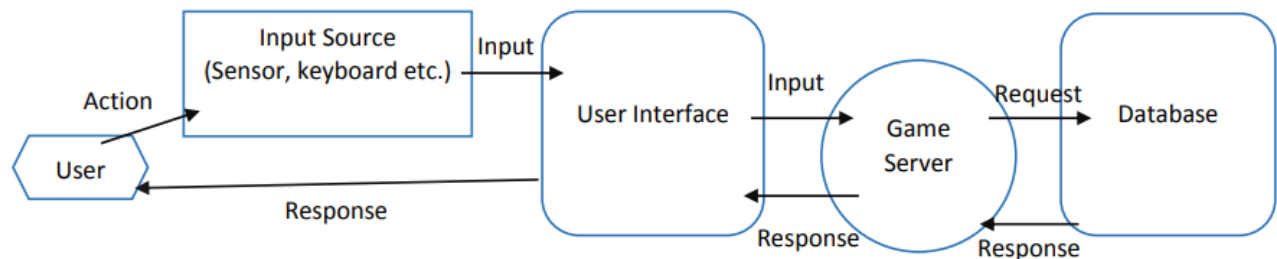
RE Vision Document 1.0, IEEE STD 830-1998, IEEE Recommended Practice for Software Requirements Specifications.

2. Overall Description

2.1 Product Perspective

The product is a single player focused, user interactive, linear 3D video game with varying RPG elements. It follows the player through an immersive story through which they discover more about themselves as they follow a linear quest chain through the game. The game uses researched methodologies and psychological principles with accessibility settings to help teach dyslexia students. The game provides a semi-multiplayer experience to allow players to use the game's tools for interactive education. Also the game will serve the purpose of a standard grading tool for such students, where teachers and guardians can view students' progress.

The system can be visualized by the following context diagram:



2.2 Product Functions

The following are the majors' functions:

- The game will let the user customize their account details.
- The game will have a character creation and management mode which allows the user to create new custom playable avatars(characters).
- The game will have multiple maps and multiple levels.
- The game will allow both Single Player as well as Multiplayer game modes.
- The game will have evolutionary NPCs for enhanced interaction.

2.3 User Classes and Characteristics

a) Student: Primary End user

The student will be the one playing the game, with all of the available features based on which Edition he/she has been provided with (Expansion bundle or Basic). The student can acquire and complete Quests, learn different courses in an engaging and fun environment.

b) Teacher: End user

The teacher will be given access to the compiled analytics of all of the students that are signed up with the teacher. This data will include information like story progress tracking, active time, stall time, overall competence based on time taken on each level, memory retention and more. This system provides a standard for grading the students based on the information provided to the teacher, which will allow them to grade the students based on their progress.

c) Guardian: End user

The Guardian will be able to access their child's complete server record and the data based on their activity along with it. This will allow them to vividly understand the child's progression.

d) General Gamer: End user

The casual gamer that may not be diagnosed with Dyslexia can also take full advantage of the Game's fun, stylized and engaging method of education, as a means to both learn and enjoy.

e) Developers:

The developers will be developing the System/Game, ensuring that all of the features that have been acquired and integrated. They will be able to receive data based on flaws in the system, as a record generated by the system itself. This is so that they may know what to patch and where to patch it, in the next updates.

f) Institutes:

Institutes will be allowed special access to data management and progression systems. Also access to beta patches to give feedback on whether the changes are acceptable or not.

g) Market Analyst:

The Market Analyst will ensure that there is going to be a market demand for the product's features and unique take on education for the uniquely-abled.

2.4 Operating Environment

The game is capable of operating in all types of operating systems however for windows it will only work with Windows 11. The game is 3D hence why it requires GPU to operate efficiently and only graphics card GTX 960 and above are capable of running it along with the Graphics card it also requires a minimum of 8gb RAM and minimum CPU i5 8th Gen. The games also require the Internet to function.

2.5 Design and Implementation Constraints

Unity3D will be used for graphics.

2.6 User Documentation

TBD.

2.7 Assumptions and Dependencies

After the testing phase, we will decide the minimum requirements and oldest versions to be supported then released on market. The game is dependent on the availability of Internet Connection. Any connection loss from a client will result in loss in game and other users will be warned.

3. External Interface Requirements

3.1 User Interfaces

Characteristics of User Interfaces:

- ❖ The Main Screen will consist of multiple options such as,
 - New Game
 - Continue
 - Settings
 - Level Selection
 - Account Management
 - Character Management
 - Exit
- ❖ All of these options will be buttons which will allow the user to navigate through them by clicking on them.
- ❖ The game interface will have different levels and difficulties and each of them will have a different interface.
- ❖ The game interface will have Multiplayer and Single player game modes with different interface according to the playable level
- ❖ The game interface will have a pause menu which will allow the user to stop the game and return to the menu or go to settings and change different settings.
- ❖ The user will also be able to customize the characters and also can edit their profile/account as well.
- ❖ The game's interface will have a loading screen which will display helpful tips and instructions to play the game.
- ❖ The game interface will be in English with later update of different selective languages.

3.2 Hardware Interfaces

The hardware interface is composed of keyboards, mouse, personal computers and game servers.

3.3 Software Interfaces

The software interfaces required for our games are as specified below:

Unity3D: Unity3D is the game engine that we will use to develop our game. The version that we have specifically used to develop the game is Unity 2022.1.13f1. We have specifically used Unity

as it gives the programmers complete control over the graphics and game environment. It is also the easiest to use game engine with plenty of manuals and tutorials available.

MySQL: It is the database management system that we have used to store and retrieve the game data. We have used the MySQL community server 8.0.31 to store the game data. We have specifically used this as MySQL is a commonly and widely Database management system with plenty of online documentation and support available online.

OS: Our game has also built-in support for all types of operating systems, but it is built on the Windows operating system.

3.4 Communications Interfaces

Communication between the client application and the server is provided by **UDP** (User Datagram Protocol).

4. System Features

4.1 Character Management

4.1.1 Character List

4.1.1.1 Description and Priority

The character list is the complete list of all the default characters as well as the custom characters created by the user.

Priority=High.

4.1.1.2 Functional Requirements

4.1.1.2.1 The game shall display all the available players characters in the form of a list.

4.1.1.2.2 The player shall be allowed to select any 1 of the characters available in the list.

4.1.1.2.2.1 The game shall display the character description of the selected character in a text box.

4.1.1.2.2.2 The game shall display a preview of the character model of the selected model.

4.1.1.2.3 The game shall allow the player to search for a specific character by entering the character name within the search bar.

4.1.1.2.4 The game shall let the user preview the character model by hovering over the character name within the list.

4.1.2 Inventory

4.1.2.1 Description and Priority

The Inventory is the list that is displayed alongside the character list, and it contains all types of equipment, weapons and potions unlocked or obtained by the player and it also shows which equipment is equipped by which character.

Priority=High.

4.1.2.2 Functional Requirements

4.1.2.2.1 The game shall group all of the equipment into categories.

4.1.2.2.2 The game shall display all of the categories as tabs to the user.

4.1.2.2.3 The user shall view the equipment by selecting the categories shown by the game.

4.1.2.2.3.1 If the user selects the helmets category, then the game shall display all the helmets unlocked by the player by retrieving the data from the database.

4.1.2.2.3.1.1 The game shall display Two columns to the user.

4.1.2.2.3.1.1.1 The game shall highlight the first column as "Equipped Helmets"

4.1.2.2.3.1.1.1.1 The game shall display all the "equipped" helmets in the first column.

4.1.2.2.3.1.1.1.2 The game shall display the name of the character that has equipped each helmet alongside each item.

4.1.2.2.3.1.1.1.3 If there are no "equipped helmets, then the game shall display "No Helmets Available".

4.1.2.2.3.1.1.2 The game shall highlight the second column as "Unequipped Helmets".

4.1.2.2.3.1.1.2.1 The game shall display all the "unequipped" helmets in the second column.

4.1.2.2.3.1.1.2.2 If there are no "unequipped helmets, then the game shall display "No Helmets Available"

4.1.2.2.3.2 If the user selects the body category, then the game shall display all the body armor unlocked by the player by retrieving the data from the database.

4.1.2.2.3.2.1 The game shall display Two columns to the user.

4.1.2.2.3.2.1.1 The game shall highlight the first column as “Equipped Armor”.

4.1.2.2.3.2.1.1.1 The game shall display all the “equipped” armor inside the first column.

4.1.2.2.3.2.1.1.2 The game shall display the name of the character that has equipped each armor alongside every item.

4.1.2.2.3.2.1.1.3 If there are no “equipped helmets, then the game shall display “No Helmets Available”.

4.1.2.2.3.2.1.2 The game shall highlight the second column as “Unequipped Armor”.

4.1.2.2.3.2.1.2.1 The game shall display all the “unequipped” armor inside the second column.

4.1.2.2.3.2.1.2.2 If there is no “unequipped Armor, then the game shall display “No Armor Available”.

4.1.2.2.3.3 If the User selects the boots category, then the game shall display all the boots unlocked by the player by retrieving data from the database.

4.1.2.2.3.3.1 The game shall display two columns to the user.

4.1.2.2.3.3.1.1 The game shall highlight the first column as “Equipped Boots”.

4.1.2.2.3.3.1.1.1 The game shall display all the “equipped” boots inside the first column.

4.1.2.2.3.3.1.1.2 The game shall display the name of the character that has equipped each boot alongside each item.

4.1.2.2.3.3.1.1.3 If there are no “equipped helmets, then the game shall display “No Helmets Available”.

4.1.2.2.3.3.1.2 The game shall highlight the second column as “Unequipped Boots”.

4.1.2.2.3.3.1.2.1 The game shall display all the “unequipped” boots within this column.

4.1.2.2.3.3.1.2.2 If there are no “unequipped boots, then the game shall display “No Boots Available”.

4.1.2.2.3.4 If the user selects the weapons category, then the game shall display all the weapons unlocked by the player by retrieving data from the database.

4.1.2.2.3.4.1 The game shall display two columns to the user.

4.1.2.2.3.4.1.1 The game shall highlight the first column as ‘Equipped Weapons”.

4.1.2.2.3.4.1.1.1 The game shall display all the “equipped” weapons inside the first column.

4.1.2.2.3.4.1.1.2 The game shall display the name of the character that has equipped that item.

4.1.2.2.3.4.1.1.3 If there are no “equipped helmets, then the game shall display “No Helmets Available”.

4.1.2.2.3.4.1.2 The game shall highlight the second column as “Unequipped Weapons”.

4.1.2.2.3.4.1.2.1 The game shall display all the “unequipped” weapons inside the second column.

4.1.2.2.3.4.1.2.2 If there are no “unequipped weapons, then the game shall display “No Weapons Available”.

4.1.3 Create Characters:

The create characters feature is a sub feature of character management that allows the user to create additional characters that they can play with within the game.

Priority=High.

4.1.3.1 Select Body Features

4.1.3.1.1 Description and Priority

This feature allows the user to update and customize all the bodily features of the character from the given options within the menu.

Priority=High.

4.1.3.1.2 Functional Requirements

4.1.3.1.2.1 The game shall allow the player to select the gender of character model between male and female.

4.1.3.1.2.1.1 If the player selects the male gender, then the game shall display the male character model on the screen, to let the user select from.

4.1.3.1.2.1.2 If the player selects the female gender, then the game shall display the female character model.

4.1.3.1.2.2 The game shall display the options to edit the character models hair.

4.1.3.1.2.2.1 The game shall display the different styles of hair available for the character models within the game, to let the user to select from.

4.1.3.1.2.2.1.1 The user shall select the hair style from among displayed by the game.

4.1.3.1.2.2.2 The game shall allow the change the color of the hair of the character model.

4.1.3.1.2.3 The game shall display all the options to edit the characters facial features.

4.1.3.1.2.3.1 The game shall display all the options to edit the characters eyes.

4.1.3.1.2.3.1.1 The game shall display all available eye styles within the game, to let the user to select from.

4.1.3.1.2.3.1.1.1 The user shall select the eye style from among displayed by the game.

4.1.3.1.2.3.1.2 The game shall allow the player to change the color of the eye.

4.1.3.1.2.3.2 The game shall display all the options to edit the ears of the character.

4.1.3.1.2.3.2.1 The game shall display all available ears style within the game to let the user to select from.

4.1.3.1.2.3.2.1.1 The user shall select the ear style from among the styles displayed by the game.

4.1.3.1.2.3.2.2 The game shall allow the player to change the ears color.

4.1.3.1.2.4 The game shall allow the player to adjust the height of the character models.

4.1.3.1.2.4.1 The game shall limit the maximum height of the character to "7 Feet".

4.1.3.1.2.4.2 The game shall limit the minimum height of the character to "4 Feet".

4.1.3.1.2.5 The game shall allow the player to adjust the weight of the character model.

4.1.3.1.2.5.1 The game shall limit the maximum weight of the character to "150 KG".

4.1.3.1.2.5.2 The game shall limit the minimum weight of the character to "50 KG".

4.1.3.1.2.6 The game shall have an undo button to retrieve the previous step in the event accidentally edits a part or does like the change.

4.1.3.1.2.7 The game shall allow the user to proceed to the next step through the continue button.

4.1.3.1.2.8 If the user presses the continue button, then the game shall load the next step of the character creation.

4.1.3.1.2.8.1 If the user presses the continue button, then the game shall use the recent state of the character model for the finalized character.

4.1.3.2 Select Equipment

4.1.3.2.1 Description and Priority

The user can customize the equipment of the character they are creating but they can only use equipment that is not being used by any other character.

Priority=High.

4.1.3.2.2 Functional Requirements

4.1.3.2.2.1 The game shall allow the user to customize the equipment of the character they are creating.

4.1.3.2.2.2 The game shall display an inventory list of all the “unequipped” equipment.

4.1.3.2.2.3 The game shall display the inventory by grouping all equipment into categories.

4.1.3.2.2.4 The game shall allow the user to select any of the categories.

4.1.3.2.2.4.1 If the user selects the Helmets category, then the game shall display all the unequipped helmets in a list.

4.1.3.2.2.4.1.1 If there are no unequipped helmets available to display, then the game shall display “No helmets Available”.

4.1.3.2.2.4.1.2 If at least 1 helmet is available, then the game shall allow the user to select 1 helmet to be equipped onto the character.

4.1.3.2.2.4.1.2.1 If a user selects a helmet, then the game shall display the character wearing that helmet.

4.1.3.2.2.4.2 If the user selects the Armor Category, then the game shall display all the unequipped armor in a list.

4.1.3.2.2.4.2.1 If there is no unequipped armor available to display, then the game shall display “No Armor Available”.

4.1.3.2.2.4.2.2 If at least 1 armor is available, then the game shall allow the user to select 1 armor to be equipped onto the character.

4.1.3.2.2.4.2.2.1 If a user selects an armor, then the game shall display the character wearing that armor.

4.1.3.2.2.4.3 If the user selects the Boots Category, then the game shall display all the unequipped boots in a list.

4.1.3.2.2.4.3.1 If there are no unequipped boots available to display, then the game shall display “No Boots Available”.

4.1.3.2.2.4.3.2 If at least 1 Boot is available, then the game shall allow the user to select 1 boot to be equipped onto the character.

4.1.3.2.2.4.3.2.1 If a user selects a boot, then the game shall display the character wearing that boot.

4.1.3.2.2.4.4 If the user selects the Weapons Category, then the game shall display all the unequipped weapons in a list.

4.1.3.2.2.4.4.1 If there are no unequipped weapons available to display, then the game shall display “No Weapon Available”.

4.1.3.2.2.4.4.2 If at least 1 weapon is available, then the game shall allow the user to select 1 weapon to be equipped onto the character.

4.1.3.2.2.4.4.2.1 If a user selects a weapon, then the game shall display the character wearing that weapon.

4.1.3.2.2.5 The game shall allow the user to move to the next by the continue button.

4.1.3.2.2.6 If the user presses the continue button, then the game shall load the next step of character creation.

4.1.3.2.2.6.1 If the user presses the continue button and no item is selected, then the game shall turn the equipment status of the character to “free”.

4.1.3.2.2.6.2 If the user presses the continue button and at least 1 item is selected, then the game shall set the character equipment status for all selected items to “occupied”.

4.1.3.3 Save Progress

4.1.3.3.1 Description and Priority

This feature allows the user to save the character before it is completed and can exit the character creation menu and resume the character creation from the exact point they exited.

Priority=High.

4.1.3.3.2 Functional Requirements

4.1.3.3.2.1 The game shall save the character creation progress at each change the player makes.

4.1.3.3.2.2 If the player tries to exit the character creation mode, then the game shall prompt the user to save the progress.

4.1.3.3.2.2.1 If the user saves their progress, then the game shall display and allow the user to select the unfinished character.

4.1.3.3.2.2.1.1 If the user selects the unfinished character, then the game shall retrieve the player's progress from the database and the player can continue building their character.

4.1.3.4 Character Information

4.1.3.4.1 Description and Priority

The user can give custom names and age along with a short background description about the characters they are creating.

Priority=High.

4.1.3.4.2 Functional Requirements

4.1.3.4.2.1 The game shall allow the player to customize the name for the character they are creating.

4.1.3.4.2.2 The game shall allow the player to specify the age of the character they are creating.

4.1.3.4.2.3 The game shall allow the player to give the character a background story or a life description.

4.1.3.4.2.4 The game shall let the user finish creating the character at this process by the Finish button.

4.1.3.4.2.5 If the user presses the finish button, then the game shall save the data inside the database.

4.1.3.4.2.5.1 If the user presses the finish button and any of the data boxes are empty, then the game shall prompt the user to fill those boxes before continuing.

4.1.4 Edit Characters

4.1.4.1 Description and Priority

The user can select previously made characters and edit the clothes and body features of any selected characters.

Priority=Medium.

4.1.4.2 Functional Requirements

4.1.4.2.1 If the player clicks the edit character buttons, then the game displays the list of all the characters created.

4.1.4.2.2 The game shall ask the player to select 1 of the characters from the displayed list.

4.1.4.2.3 The game shall retrieve the data of the character model and then display the selected character model.

4.1.4.2.4 The game shall allow the user to edit the character information and the equipment loadout as well.

4.1.4.2.4.1 If the user selects the edit information option, then the game shall display all the information of the character.

4.1.4.2.4.1.1 The game shall allow the user to change the name of the character.

4.1.4.2.4.1.2 The game shall allow the user to change the age of the character.

4.1.4.2.4.1.3 The game shall allow the user to change the description of the character.

4.1.4.2.4.2 If the user selects the edit equipment option, then the game shall display all 4 options to edit the equipment to the user.

4.1.4.2.4.2.1 If the user selects the helmet option, then the game shall display all the unequipped helmets for the user.

4.1.4.2.4.2.1.1 The game shall display two columns to the user.

4.1.4.2.4.2.1.1.1 The game shall highlight the first column as Equipped Helmet.

4.1.4.2.4.2.1.1.2 The game shall display the equipped helmet of the character within the first column.

4.1.4.2.4.2.1.1.3 If the character has no equipped helmet, then the game shall display “No Helmet Equipped” in the first column.

4.1.4.2.4.2.1.1.4 The game shall highlight the second column as “Available Helmets”.

4.1.4.2.4.2.1.1.5 The game shall display all “unequipped” helmets within the second column as a list.

4.1.4.2.4.2.1.1.6 If there are no unequipped helmets, then the game shall display “No Helmets Available”.

4.1.4.2.4.2.1.1.7 If at least 1 unequipped helmet is available and a helmet is already equipped by the character, then the user shall replace the current helmet by selecting 1 helmet from the second column.

4.1.4.2.4.2.1.1.8 If at least 1 unequipped helmet is available and no helmet is equipped by the character, then the user shall equip the helmet by selecting 1 helmet from the second column.

4.1.4.2.4.2.2 If the user selects the Armor option, then the game shall display all the unequipped armor for the user.

4.1.4.2.4.2.2.1 The game shall display two columns to the user.

4.1.4.2.4.2.2.1.1 The game shall highlight the first column as Equipped Armor.

4.1.4.2.4.2.2.1.2 The game shall display the equipped armor of the character within the first column.

4.1.4.2.4.2.2.1.3 If the character has no equipped armor, then the game shall display “No armor Equipped” in the first column.

4.1.4.2.4.2.2.1.4 The game shall highlight the second column as “Available Armor”.

4.1.4.2.4.2.2.1.5 The game shall display all “unequipped” armor within the second column as a list.

4.1.4.2.4.2.2.1.6 If there is no unequipped armor, then the game shall display “No Armor Available”.

4.1.4.2.4.2.2.1.7 If at least 1 unequipped armor is available and an armor is already equipped by the character, then the user shall replace the current armor by selecting 1 armor from the second column.

4.1.4.2.4.2.2.1.8 If at least 1 unequipped armor is available and no armor is equipped by the character, then the user shall equip the armor by selecting 1 armor from the second column.

4.1.4.2.4.2.3 If the user selects the boot option, then the game shall display all the unequipped boots for the user.

4.1.4.2.4.2.3.1 The game shall display two columns to the user.

4.1.4.2.4.2.3.1.1 The game shall highlight the first column as Equipped Boot.

4.1.4.2.4.2.3.1.2 The game shall display the equipped boot of the character within the first column.

4.1.4.2.4.2.3.1.3 If the character has no equipped boot, then the game shall display “No Boot Equipped” in the first column.

4.1.4.2.4.2.3.1.4 The game shall highlight the second column as “Available Boots”.

4.1.4.2.4.2.3.1.5 The game shall display all “unequipped” boots within the second column as a list.

4.1.4.2.4.2.3.1.6 If there are no unequipped boots, then the game shall display “No Boots Available”.

4.1.4.2.4.2.3.1.7 If at least 1 unequipped boot is available and a boot is already equipped by the character, then the user shall replace the current boot by selecting 1 boot from the second column.

4.1.4.2.4.2.3.1.8 If at least 1 unequipped boot is available and no boot is equipped by the character, then the user shall equip the boot by selecting 1 boot from the second column.

4.1.4.2.4.2.4 If the user selects the weapon option, then the game shall display all the unequipped weapons for the user.

4.1.4.2.4.2.4.1 The game shall display two columns to the user.

4.1.4.2.4.2.4.1.1 The game shall highlight the first column as Equipped Weapon.

4.1.4.2.4.2.4.1.2 The game shall display the equipped weapon of the character within the first column.

4.1.4.2.4.2.4.1.3 If the character has no equipped weapon, then the game shall display “No Weapon Equipped” in the first column.

4.1.4.2.4.2.4.1.4 The game shall highlight the second column as “Available Weapon”.

4.1.4.2.4.2.4.1.5 The game shall display all “unequipped” weapons within the second column as a list.

4.1.4.2.4.2.4.1.6 If there are no unequipped weapons, then the game shall display “No Weapon Available”.

4.1.4.2.4.2.4.1.7 If at least 1 unequipped weapon is available and a weapon is already equipped by the character, then the user shall replace the current weapon by selecting 1 weapon from the second column.

4.1.4.2.4.2.4.1.8 If at least 1 unequipped weapon is available and no weapon is equipped by the character, then the user shall equip the weapon by selecting 1 weapon from the second column.

4.1.4.2.4.3 The game shall allow the user to confirm these changes by pressing confirm.

4.1.4.2.4.4 If the User presses the confirm button, then the game shall save the changes within the database and the game shall load the previous page.

4.1.4.2.5 The game shall allow the user to save all the changes by pressing Confirm Changes.

4.1.4.2.6 If the user presses the confirm changes, then the game shall overwrite the character data with the new data in the database.

4.1.4.2.7 After the game saves the changes, the game shall close the Edit Character window and return to the Character management window.

4.1.5 Delete Characters

4.1.5.1 Description and Priority

The user can select one or more and delete them from the character list making these characters unavailable to be used as playable characters within the game levels.

Priority=High.

4.1.5.2 Functional Requirements

4.1.5.2.1 If the player selects the delete characters option, then the game shall display the list of all available characters.

4.1.5.2.2 The game shall prompt the user to select at least 1 character.

4.1.5.2.3 If a minimum of 1 character is selected, then the game shall unlock the continue option for the user.

4.1.5.2.3.1 When the user clicks the continue option, the game shall open a dialogue box and ask the player to confirm deletion or cancel selection.

4.1.5.2.3.1.1 If the player presses confirm deletion, then the game shall delete all selected characters from the database.

4.1.5.2.3.1.2 If the player presses cancel selection, then the game shall close the dialogue box and unselect all the previously made selection.

4.1.5.2.4 If a deletion has been made, then the game shall refresh the character list and remove all deleted characters.

4.1.5.2.5 The game shall set the weapons and equipment of the deleted character to free, so the user can equip those things onto other characters.

4.1.5.2.6 The game shall display those equipment and weapons of the deleted character as "Equipable" inside the Inventory.

4.2 Interactive Map

4.2.1 The system shall open the main-Map window upon pressing the M-key.

4.2.1.1 If the main-Map window is already open, the system shall close the main-Map window upon pressing the M-key.

4.2.2 The system shall open the mini-Map window upon pressing the Tab-key.

4.2.2.1 If the mini-Map window is already open, the system shall close the mini-Map window upon pressing the Tab=key.

4.2.2.2 The mini-Map shall have a base opacity of 35% (thirty-five percent).

4.2.2.3 The mini-Map shall have a base location at the top-right corner of the screen.

4.2.2.4 The mini-Map can have its location changed by right-click (Mouse-2) and dragging it towards a different location on the screen.

4.2.2.5 The mini-Map shall only mimic icons, markers and text from the main-Map.

4.2.3 The main-Map shall display the outline of the Continent the Player is concurrently existing in, a marker indicating the Player's current position and names of Explored Areas.

4.2.3.1 If the Player has an active Quest, the main-Map shall display a Quest Marker.

4.2.3.2 If the Player has more than 1 (>1) active Quests, the main-Map shall display the Quests in accordance with the Quest-number shown by the Quest-Journal.

4.2.3.3 If the Player has not actively visited an Area, the main-Map shall display the area with three-question-marks ("???").

4.2.3.4 If the Player has actively visited an Area, the main-Map shall display the respective name of the Area.

4.2.3.5 The main-Map shall display the current Continent name at the top of the window.

4.2.4 The main-Map shall have a default Zoom-level of 100% (one-hundred percent).

4.2.4.1 The main-Map shall Zoom-In by using the mouse scroll-wheel up (Mouse-3) in increments of 5% (5 percent) Zoom-In added to the scale, up till 200% (two-hundred percent) Zoom level.

4.2.4.2 The main-Map shall Zoom-Out by using the mouse scroll-wheel down (Mouse-3) in decrements of 5% (5 percent) Zoom-Out subtracted to the scale, up till 40% (forty percent) Zoom level.

4.2.4.3 The main-Map should have a Reset-Zoom button in the bottom left corner of the main-Map window which resets the Zoom level to 100% (one-hundred percent).

4.2.4.4 The main-Map window shall have a Zoom Level slider on the left side of the window which can be used by left-clicking (Mouse-1) the slider and dragging it.

4.2.5 The main-Map window shall have a Search bar at the top right corner of the window.

4.2.5.1 The Search bar shall show an I-beam pointer ("I") when left clicked (Mouse-1) to allow typing.

4.2.5.2 The Search bar shall show suggestions in a drop-down menu when the user types in the Search bar.

4.2.6 The main-Map shall have a "View My Location" button in the bottom left corner of the window which shall center the main-Map onto the player and reset Zoom level to 100% (one-hundred percent).

4.2.7 The main-Map shall have a "World Map" button in the bottom left corner of the window which shall show the World Map.

4.2.7.1 The World Map shall have the names of all the Continents written above their respective locations.

4.2.7.2 The World Map can be Zoomed in to show a closer view of the Continents

4.2.7.3 The system shall move the main-Map from World Map to Continent Map (with Continent name at the top) when the respective Continent name is left-clicked (Mouse 1).

4.2.7.4 The World Map shall show the Player marker next to the name of the Continent that they currently reside in.

4.2.8 The main-Map shall have a "Legends" button that shall show a drop down menu for Legends NPCs which can be left-clicked (Mouse 1) to toggle them on and off.

4.2.8.1 If an item from the Legends drop-down menu has been unselected (turned off) the main-Map shall remove all respective markers from that Continent and the item shall be gray-scaled.

4.2.8.2 If an item from the Legends drop-down menu has been selected (turned on) the main-Map shall display all respective markers in that Continent and the item shall be colored.

4.2.9 The system shall allow the player to use a Triport when left-clicking (Mouse-1) the Triport icon on the main-Map

4.2.9.1 The system shall display a dialogue box for confirming Triport travel when interacted.

4.2.10 The main-Map shall show a Tooltip when hovering over a Legend marker.

4.2.11. The system shall allow the player to Auto-Move when pressing Alt-key (Left Alt or Right Alt) and right-clicking (Mouse-2) a location on the main-Map.

4.2.11.1 The system shall allow up to 4 (four) Auto-Move Anchor locations at a time.

4.2.11.2 The system shall create a numbering between 1-4 (one to four) when placing Anchors and the main-Map shall display the Anchors respectively with indicators.

4.2.11.3 The system shall block an Auto-Move Anchor if selecting out-of-bounds.

4.2.11.4 When the Player character reaches an Anchor the system shall move the Player towards the next Anchor.

4.2.11.4.1 If the Player has no more Anchors to approach the system shall display the message "Auto-Movement has ended".

4.2.11.4.2 The system shall increment the number of Anchors the Player has when one Anchor has been reached and decrement the Anchor counting on the main-Map.

4.2.11.4.3 The Player can place another Anchor if there are less than 4 (<4) Anchors placed on the Map.

4.2.12 The system shall generate a field of set render distance (12 Chunks) in which the Player can perform Movement and Attack actions

4.2.12.1 The system shall use Entity Culling to improve game performance.

4.2.13 The system shall generate Obstacles which inhibit the Player's movement

4.2.13.1 If the Obstacle encountered is a Border, the system shall remove any horizontal velocity the Player exhibits towards the Border.

4.2.13.2 If the Obstacle encountered is a Rock, NPC, Collectible (or similar small objects) the system shall reduce horizontal velocity of the Player by -0.3 m/s (negative zero point three meters per second) and increase the Obstacle's horizontal velocity by 0.6 m/s (zero point six meters per second).

4.2.13.3 If the Obstacle encountered is a DeathZone, the system shall kill the Player and respawn them to their most recent Triport.

4.2.14 The system shall generate NPC Zones in fixed locations of the Map.

4.2.14.1 If the NPC Zone is a Hostile NPC Zone, the system shall spawn 3-6 (three to six) Hostile NPCs depending on the difficulty setting.

4.2.14.2 If the NPC Zone is a Friendly NPC Zone, the system shall spawn 1 (one) Blacksmith, 1 Weapons Dealer, 1 Potion Merchant, 1 Medic, 1 Cook and 1 Gear Honer by default.

4.2.14.3 If the NPC Zone is a Neutral NPC Zone, the system shall spawn 3-6 (three to six) Neutral NPCs by random.

4.2.14.4 The system shall spawn 0-2 (zero to two) Quest NPCs by random in Neutral and Friendly NPC Zones.

4.2.15 The system shall have a fixed Day/Night Cycle lasting the duration of the gameplay session

4.2.15.1 When the Player begins a session, the Day/Night Cycle shall begin from 7:00 (7 o'clock) from the in-game timer.

4.2.15.2 The Day/Night Cycle shall increment 1 (one) in-game minute every 1 real-life second.

4.2.15.2.1 Day Cycle shall last between 4:31 AM - 7:30 PM in-game time.

4.2.15.2.2 Night Cycle shall last between 7:31 Pm – 4:30 PM in-game time.

4.2.15.3 The system shall generate Day-Time Effects while Day is active during the Day/Night Cycle.

4.2.15.4 The system shall generate Night-Time Effects while Night is active during the Day/Night Cycle.

4.2.16 The system shall generate Water bodies in random chunks.

4.2.16.1 If the Player enters a Water body the system shall reduce their horizontal movement by –1.5m/s (negative one point five meters per second).

4.2.16.2 If the Player's character has half of its hitbox within a water body, the system shall change the Player character into Swimming mode.

4.2.16.2.1 If the Player's character hitbox collides with Ground under a water body they shall exit Swimming Mode and return to regular stance.

4.2.16.3 If the Water body is 1/6 (one sixth) chunk deep the Player should be able to use Fishing skills in the Water body.

4.2.16.4 If the Water body is 1/6 (one sixth) chunk deep the system shall spawn a Neutral NPC Zone.

4.2.16.5 If the Water body is $\geq 6 \times 6$ (greater than or equal to six by six) chunks, the system shall generate a Water dungeon with a random chance between 0 – 0.4 (0 = never, 1 = always).

4.2.17 The system shall generate Mountains, Hills and Plateaus in random chunks.

4.2.17.1 The system shall generate Hills within 3 (three) chunks of Neutral NPC Zones.

4.2.17.2 The system shall generate Mountains in northern regions of the Map.

4.2.17.3 The system shall generate Plateaus in southern regions of the Map.

4.2.18 The system shall generate Buildings, Huts and Campfires within 3 (three) chunks of near NPC Zones.

4.2.18.1 The system shall generate 4-5 (four to five) Buildings near Friendly NPC Zones.

4.2.18.2 The system shall generate 0-2 (zero to two) Campfires near Friendly NPC Zones and Hostile NPC Zones.

4.2.18.3 The system shall generate 2-3 (two to three) Huts near Hostile NPC Zones.

4.3 Interactive NPCs

4.3.1 If an NPC falls under the category of “Companion” then the System shall provide the player with the option to view their inventory items.

4.3.2 If the player unlocks the perk “Born Leader” then the System shall allow the player to command NPCs that fall under the category of “Companion” i.e., attack a target, interact with an item, stop movement.

4.3.3 If the player interacts with an NPC, then the NPC shall provide dialogue to the player.

4.3.3.1 If the NPC that is being interacted with is a “non-companion NPC” then the dialogue shall be a one sentence greeting or remark.

4.3.3.2 If the NPC that is being interacted with is a “companion NPC” then the dialogue shall be “branching”.

4.3.3.2.1 The Player can choose options in Dialogue by using the left-click (Mouse-1) on given options.

4.3.3.2.2 If the NPC is a Quest NPC, the system shall provide the Player with an option to accept the Quest.

4.3.3.2.3 If the Player has completed at least 1 (one) Quest for an NPC, they can become a Companion.

4.3.4 If an NPC is of the category “Companion” and the dialogue for an acquaintance has been completed then the NPC shall follow the player around.

4.3.4.1 Companion NPCs shall attack Hostile NPCs that attack the Player.

4.3.4.2 Companion NPCs shall remark on the Player’s actions by random when they kill a Hostile NPC.

4.3.4.2.1 Remarks can be of fixed pool of voice-lines.

4.3.4.3 Companion NPCs shall die when they reach 0 (zero) hit-points.

4.3.5 If an NPC is of the category “Quest-Giver” then the NPC shall provide the player with specific “quests”.

4.3.6 If an NPC is of the category “Companion” and the dialogue for an acquaintance has been completed then the NPC shall allow the player to store their items in the NPCs inventory.

- 4.3.6.1** The NPC's inventory shall open within 1 second of clicking the assigned button.
- 4.3.6.2** The Player should be able to heal the companion by placing Health Potions in the Companions Inventory.
- 4.3.6.3** The Player can not place loot more than the given space in a Companion's Inventory.
- 4.3.6.4** If the Companion has full Inventory, the Companion shall have reduced horizontal movement of -1.2m/s (negative one point two meters per second).
- 4.3.6.5** The Companion shall be able to use a Weapon that the Player places inside their Inventory.
- 4.3.7** If the player makes a decision and/or travels to a new location then the "Companion" NPC shall react with dialogue e.g., a remark.
- 4.3.8** If two NPCs come within 2 grid-spaces (Unity 3d engine measurement) then the NPCs shall form dialogue with each other.
- 4.3.9** If the player and an "enemy NPC" come within 15 grid-spaces (Unity 3d engine measurement) then the player will enter "combat state" with the NPC.
 - 4.3.9.1** If the Player has equipped a bow and arrow then the System shall grant the NPC a 40% chance to dodge incoming projectiles.
 - 4.3.9.2** If the Player has equipped a heavy weapon then the System shall grant the NPC a 65% chance of equipping a shield.
 - 4.3.9.3** If the Player has equipped Frost Magic then the System shall grant the NPC a 65% chance of equipping a "Frost-Repelling" Item.
 - 4.3.9.4** If the Player has equipped Fire Magic then the System shall grant the NPC a 65% chance of equipping a "Fire-Repelling" Item.
 - 4.3.9.5** If the Player has equipped Shock Magic then the System shall grant the NPC a 65% chance of equipping a "Shock-Repelling" Item.
- 4.3.10** The system shall spawn NPCs in NPC Zones.
 - 4.3.10.1** The system shall spawn Friendly NPCs in Friendly NPC Zones.
 - 4.3.10.1.1** If the "Crowd Density" setting is set to low then the system shall spawn 10 Friendly NPCs.
 - 4.3.10.1.2** If the "Crowd Density" setting is set to medium then the system shall spawn 15 Friendly NPCs.
 - 4.3.10.1.3** If the "Crowd Density" setting is set to high then the system shall spawn 20 Friendly NPCs.
 - 4.3.10.2** The system shall spawn Neutral NPCs in Neutral NPC Zones.
 - 4.3.10.2.1** If the "Crowd Density" setting is set to low then the system shall spawn 10 Neutral NPCs.
 - 4.3.10.2.2** If the "Crowd Density" setting is set to medium then the system shall spawn 15 Neutral NPCs.
 - 4.3.10.2.3** If the "Crowd Density" setting is set to high then the system shall spawn 20 Neutral NPCs.
 - 4.3.10.3** The system shall spawn Hostile NPCs in Hostile NPC Zones.
 - 4.3.10.3.1** If the "Crowd Density" setting is set to low then the system shall spawn 10 Hostile NPCs.

4.3.10.3.2 If the “Crowd Density” setting is set to medium then the system shall spawn 15 Hostile NPCs.

4.3.10.3.3 If the “Crowd Density” setting is set to high then the system shall spawn 20 Hostile NPCs.

4.3.10.4 The system shall spawn Quest NPCs in Friendly and Neutral NPC Zones.

4.3.10.4.1 If the “Crowd Density” setting is set to low then the system shall spawn 10 Quest NPCs.

4.3.10.4.2 If the “Crowd Density” setting is set to medium then the system shall spawn 15 Quest NPCs.

4.3.10.4.3 If the “Crowd Density” setting is set to high then the system shall spawn 20 Quest NPCs.

4.3.11 The system shall allow the Player to attack non-Friendly NPCs.

4.3.11.1 If the Player uses a weapon, the Player shall enter Combat Mode.

4.3.11.2 If the Player is within 15 grids of an “Enemy NPC” (Unity 3D engine measurement) the system shall enter the Player into Combat Mode.

4.3.11.3 If the NPC is a Neutral NPC, the NPC shall begin moving away from the Player’s position.

4.3.11.4 If the NPC is a Hostile NPC, the PC shall begin attacking the Player.

4.3.12 The system shall reward the Player with loot when successfully killing an NPC.

4.3.12.1 If the killed NPC was a Neutral NPC, the system shall reward the Player with 15 (fifteen) EXP points, 5 (five) Coins and 0-2 (zero to two) items from the NPC loot table.

4.3.12.2 If the killed NPC was a Hostile NPC, the system shall reward the Player with 30 (thirty) EXP points, 10 (ten) Coins and 0-3 (zero to three) items from the NPC loot table.

4.3.13 If the Player dies from Hostile NPCs, the Hostile NPCs shall steal 10 (ten) Coins from the Player.

4.3.13.1 The system shall not despawn Hostile NPCs that have Player-dropped Coins.

4.3.13.2 The system shall drop the same number of coins the Hostile NPC picked up from the Player when killed by the Player.

4.3.14 If the Player dies (Health at 0) then the game shall revive player at closest Friendly NPC Zone/Neutral NPC Zone.

4.3.14.1 If the Player has acquired the perk “Hard to kill” then the system shall restore Player’s health to maximum upon only the FIRST death.

4.3.15 The system shall respawn enemies at respective NPC Zones every 1 (one) in-game hour (every 60 seconds real-time).

4.3.15.1 If 1 (one) NPC exists at an NPC Zone, the respawn timer shall activate to add 2-5 (two to five) more NPCs depending on Difficulty.

4.3.15.2 If more than 1 (>1) NPCs exist at an NPC Zone, the system shall not initiate the respawn timer.

4.3.16 The system shall not allow NPCs to move away more than 4 (four) chunks from their NPC Zone.

4.3.16.1 Hostile NPCs chasing the Player shall stop chase and return to their NPC Zone when exceeding 4 (four) chunks.

4.3.17 All NPCs shall have 2 (two) voice-lines when Idle, 2 voice-lines when attacking and 1 (one) Death voice-line.

4.3.17.1 NPC idle voice-lines shall activate only when the Player is within 3 grid-spaces (Unity 3d engine measurement).

4.3.17.2 NPC death voice-line shall activate when NPC hit-points reach 0 (zero).

5. Other Nonfunctional Requirements

5.1 Performance Requirements

5.1.1 System should support 1000 users concurrently.

5.1.2 Game users must have at least 3MB of internet connection speed.

5.1.3 Response time for any request should not be more than 2 seconds.

5.1.4 Delay between players should not exceed 4 seconds.

5.1.5 System should run on 60 fps under standard specification setting.

5.1.6 System displays analytics within 5 seconds on average, to Teacher or Guardian request.

5.2 Safety Requirements

5.2.1 System shall follow IEC 61511 for Functional Safety.

5.3 Security Requirements

5.3.1 System shall require an email activation to complete the registration.

5.3.2 System shall keep encrypted passwords instead of original passwords.

5.3.3 System shall use **special algorithms** to prevent users from cheating or wall hacking.

5.4 Software Quality Attributes

Reliability

5.4.1 The Mean-time-to-repair shall be at most 1 hour

5.4.2 The Mean-time-to-failure shall be at least 24 hours.

5.4.3 In case of any system failure, System shall display an error message in the main screen to inform the user about the cause of failure.

Availability

5.4.4 System should be available 24 hours per day, 7 days per week.

Maintainability

5.4.5 Documentation should be supplied for all modules of the system.

5.4.6 Requirement and change management should be used in the development phase.

5.4.7 Configuration items should be changed under version control system and linked to change requests to provide traceability.

Portability

5.4.8 Game should run on all platforms.

5.5 Business Rules

5.5.1 Students must not have access to teacher/guardian functionality.

5.5.2 System must provide game mode only from student access.

6. Other Requirements

TBD.

Appendix A: Glossary

Player – User controlled entity.

Chunk – 16x16 Grid blocks in Unity 3D.

Continent – Location made up of several Areas.

Area – Location with a fixed name and position in the world with fixed NPCs and structures.

key – Keyboard-inputted command.

Mouse – Mouse-inputted command.

Quest Journal – A window that shows the Player's currently tracked Quest(s).

Triport – A collectible that allows the Player to quickly travel between Areas.

Legends – Includes game-elements such as (but not limited to): Main Quest(s), Side Quest(s), Sudden Quest(s), Area(s), Triports, Dungeons, NPCs and Collectibles.

Tooltip – A brief description of an item.

Auto-Move – When pressing Alt+Mouse1 the system creates an Anchor. The Player's character will follow this anchor by moving on its own without the movement keys (default to WASD).

Movement Actions – used through W, A, S, D, Space, Shift, etc. to change velocity of the Player character.

Attack Actions – used through Mouse-1, Mouse-2, Mouse-3 and dragging the mouse to exhibit Player attack motions.

Entity Culling – Reducing rendered entities in an area by removing those which are not visible to the Player.

Obstacles – Any item, entity or object that has a hitbox that inhibits the Player's movement. (such as (but not limited to): DeathZones, Borders, Rocks, Collectibles, Triports, NPCs, Trees, Ground, Mountains, Buildings).

Difficulty Setting – Standard difficulty settings are given to the Player upon starting the game namely: Easy, Medium, Medium-core, Hardcore. Can be changed anytime in the settings.

Time Effects – Includes (but not limited to): Color gradient, Hue, Gamma, Contrast, Shadows, Shaders, Dynamic Lighting.

NPC Non-Player-Controlled character.

Quest A task or a series of multiple tasks that will be assigned to the Player's Quest Journal either by the game or by "Quest NPCs". Upon their completion, Player is rewarded.

NPC Zones Zones covering 1 chunk in the game where NPCs can spawn.

Combat Mode A state in which the player can choose from their weapons to eliminate the "Hostile NPC(s)" that they are currently engaged with.

Friendly NPC An NPC that can be made a follower and companion after a favor has been completed for them. They help the player along on their journey through the quests of the game. The Player can not attack this NPC and this NPC cannot attack the Player.

Neutral NPC An NPC that is only present to fill in the game world so as to add life to the game, this NPC cannot follow the player under any circumstances.

Hostile NPC An NPC that will present the player with a quest.

Quest NPC An NPC that can engage in combat with the player. The NPC can attack the Player and the Player can attack the NPC.

Dialogue This event opens a window and the Player is unable to move. The Player can then read and listen to NPCs talking.

Loot These are any items that are dropped from killing NPCs, destroying breakable objects or opening chests. (e.g., Coins, Bags, Weapons, Armours, Clothes).

Loot Tables Each NPC is specified with a loot table. Neutral NPCs have the highest loot table because of several variants such as cows, chickens, horses, frogs, fish. Each has its own loot table e.g., cow has three items in its loot table: leather, beef, hoofs.

Crowd Density A setting in the games Configuration menu that dictates how many NPCs will be spawned by the system corresponding to the respective “NPC Zone”.

Frost-Repelling Items Items that repel/negate the effects of Frost-based magic e.g. Frost negating enchanted Apparel items, Frost Ward spell.

Fire-Repelling Items Items that repel/negate the effects of Fire-based magic e.g. Fire negating enchanted Apparel items, Fire Ward spell.

Shock-Repelling Items Items that repel/negate the effects of Shock-based magic e.g. Shock negating enchanted Apparel items, Shock Ward spell.

Appendix C: To Be Determined List

- Section 2.6
- Section 6