# **Software Requirements Specification**

for

## Hearthstone Lite

Version 1.0 approved

Prepared by Logan McDowell, Picard Yang

WIU - CS491 Project

Created: 2/28/2020

Table of Contents

Table of Contents......................................................................................................... ii

Revision History........................................................................................................... ii

1. Introduction............................................................................................................ 1

1.1 Purpose............................................................................................................... 1

1.2 Document Conventions....................................................................................... 1

1.3 Intended Audience and Reading Suggestions.................................................... 1

1.4 Project Scope...................................................................................................... 2

1.5 References.......................................................................................................... 2

2. Overall Description................................................................................................ 2

2.1 Product Perspective............................................................................................ 2

2.2 Product Features................................................................................................. 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........................................................................ 3

3. System Features ......................................................................... 3

3.1 Main Menu ......................................................................... 3

3.2 Program Customization Options......................................................................... 4

3.3 Deck Customization ......................................................................... 5

3.4 Gameplay ......................................................................... 6

3.5 AI Interaction ......................................................................... 7

3.6 Music/Sound ......................................................................... 8

4. External Interface Requirements.......................................................................... 9

4.1 User Interfaces................................................................................................... 9

4.2 Hardware Interfaces........................................................................................... 10

4.3 Software Interfaces............................................................................................ 10

4.4 Communications Interfaces............................................................................... 11

5. Other Nonfunctional Requirements..................................................................... 11

5.1 Performance Requirements.............................................................................. 12

5.2 Safety Requirements........................................................................................ 12

5.3 Security Requirements..................................................................................... 13

5.4 Software Quality Attributes............................................................................... 13

6. Other Requirements........................................................................................... 13

Appendix A: Glossary............................................................................................... 13

Appendix B: Analysis Models................................................................................... 14

Appendix C: Issues List............................................................................................ 15

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Logan M,Picard Y | 3/1/2020 | Completion of Origin | 1.0 |
| Logan McDowell | 4/7/2020 | Requirements Update, Section 4,5 Revision Requirement, Issues Update, Diagrams Added | 1.1 |
| Picard Yang | 4/15/2020 | Sec 4 & 5 Requirements Update.  (5 is still updating) | 1.2 |

# **1.** **Introduction**

## **1.1** **Purpose**

Version 1.0 of project “HearthStone Lite".

The software is a game created to mimic a ‘lite’ version of Blizzard’s Hearthstone Card Game, available on PC and App-Stores. This document describes the software itself and the software SDKs required to run it.

## **1.2** **Document Conventions**

Priority of Requirements inherit parent statements when shown in a bulletin format. Child elements (sub-statements) inherit priority of the parent, as they are detailed elements within the parent.

## **1.3** **Intended Audience and Reading Suggestions**

Intended for developers, testers, management, and documentation purposes of the software project. This is to enable an organized and stable approach to understanding the requirements, implementations, and limitations of the product. General understanding may be gauged from overview summaries, with increased understanding being granted by visiting the more specific elements within the document.

## **1.4** **Project Scope**

The project itself encompasses a ‘game’ environment. The software has menus as a game would, customization of card decks, and gameplay environments. The purpose is for the player to battle against an AI using a preselected deck of cards to defeat them.

## **1.5** **References**

The program is based off of Blizzard’s Hearthstone: <https://playhearthstone.com/en-us/>,  
using design formats similar to that of the original.

# **2.** **Overall Description**

## **2.1** **Product Perspective**

This program is developed with a small-scale interpretation of Blizzard’s Hearthstone game in mind, with basic gameplay elements similar in implementation to that. The design of gameplay elements, such as card decals and abilities, will be original. The game itself consists of a player selecting from or customizing a deck to then play on a ‘playing field’ against an AI. The opponents use cards to attack/defend until one of them runs out of health points, resulting in the other opponent winning the match.

## **2.2** **Product Features**

* Main Menu
* Program Customization Options
* Deck Customization
* Gameplay (Card Battle on a playing field)
* AI interaction
* Music

## **2.3** **User Classes and Characteristics**

As this is a game, only basic knowledge of computer environments is required in order to navigate and experience the software. Most if not all elements are fairly straightforward in naming and design convention to alleviate basic questions on how to operate them. It functions as a card game where you draw cards and play them.

## **2.4** **Operating Environment**

The program runs in the Java runtime environment SDK, on any platform that may support the SDK and mouse/keyboard input. The program will be represented via Java GUI.

## **2.5** **Design and Implementation Constraints**

Limitations for the project consist mainly upon creative and intelligence-operation implementation. The AI within the program will require considerable effort to implement, with different gameplay strategies developed based on dynamic variables. The creative elements, consisting of card design factors and actual decals, will require extra development time to ensure an originality and uniqueness of cards.

## **2.6** **User Documentation**

* Gameplay Manual
  + In PDF
* On-line help
  + In online PDF/HTML

Documents to be created later in development

## **2.7** **Assumptions and Dependencies**

Java runtime environment SDK and Java GUI software are required for this program to function.

# **3.** **System Features**

## **3.1** **Main Menu**

3.1.1 Description and Priority

The main menu has modules for the main menu of the game. Which are play a game (with AI), Deck Customizer, Options, and Exit game.

Priority - 6: The risk factor for these modules is high because these are the core modules of this project.

3.1.2 Stimulus/Response Sequences

The play button navigates the user to the play game screen. The deck customizer can be created and modified through the player. Control the sound effect from option.

3.1.3 Functional Requirements

**FR01- Play:** The play feature shall redirect to start the game.

**FR02- Deck Customizer:** Selection of cards to choose from to build a personal deck of limited size.

**FR03- Option:** Control the sound effect

**FR04- Exit:** Exit the game

## **3.2 Program Customization Options**

3.2.1 Description and Priority

The program customization has various options in the game. It includes resolution, quality, sound, and other options. The cost for these options is TBD “To Be Decided”.

Priority - 4: These modules are medium because these are the moderate features for the application.

3.2.2 Stimulus/Response Sequences

The user interacts with the screen resolution to select the image quality for the game. The user allows selecting the master sound volume, sound in background checkbox and music volume. The user navigates to the other feature screen to do modification of control settings.

3.2.3 Functional Requirements

**FR01- Resolution Quality:** The resolution feature shall enable the player to select resolution.

**FR02- Sound:** The sound feature shall allow selecting sound volume options which include master sound volume, background sound checkbox and music volume.

**FR03- Other Options:** The other option shall allow the players to select games controls options in the other options menu.

**Error Conditions:**

The system shall generate an error message when the user tries to use unsupported resolution for the screen.

## **3.3** **Deck Customization**

3.3.1 Description and Priority

The deck size is TBD. The decks can be created and modified through the player. Each deck has a deck slot that the player can take during customization.

Priority - 7: It is the highly important factor in game play

3.3.2 Stimulus/Response Sequences

Choose cards from the card list which are creature cards & spell cards, and customize the deck.

3.3.3 Functional Requirements

**FR01- Card chosen:** Cards have a power-cost to play onto the field. (Mana Cost-TBD) The player chooses the creature & spell card.

**FR02- Deck:** Player has a selection of cards to choose from to build a personal deck of limited size.

**Error Conditions:**

The cost of the card must not exceed the limited cost. (Total cost TBD)

## **3.4** **Gameplay (Card Battle on a playing field)**

3.4.1 Description and Priority

The gameplay of card battle on a playing field played with AI.

Priority - 9: One of most important as it’s a core module.

3.4.2 Stimulus/Response Sequences

The gameplay is turn-based for players and AI to play. Players can choose the attack and defense by creature cards.

3.4.3 Functional Requirements

**FR01- Player:** Use the deck which is already customized and chosen from player, play with AI with turn-based playing.

**Error Conditions:**

For the gameplay, a possible error could be that of the AI having an issue, like not being able to decide what to do within a specified timeframe.

## **3.5 AI interaction**

3.5.1 Description and Priority

The Artificial Intelligence framework chooses a deck from a selection of default ones. The AI chooses attack or defense depending on the player’s cards and programmed strategies.

Priority - 9: It is a core module and most important. It is expected to show the most errors in gameplay, and it is the hardest part.

3.5.2 Stimulus/Response Sequences

AI framework which includes collectible cards and chooses decks. The AI interacts with the player user to attack the player efficiently, with the player making moves against the AI, and vice-versa.

3.5.3 Functional Requirements

**FR01- Player:** Player vs. AI through their cards they choose

**Error Conditions:**

AI Decision Making and AI Timeout.

## **3.6** **Music**

3.6.1 Description and Priority

The game attracts the user’s attention through ear attractive music ringtones. The music module includes ringtone, sound volume, background music mode and music volume which interacts with the user.

Priority - 4: It is the one of the ways players will be more interested in gameplay, but it doesn’t really affect gameplay on the other hand.

3.6.2 Stimulus/Response Sequences

The user interacts with the game and it entertains the player through the music. The users navigate to the sound volume, background music mode and sound volume options.

3.6.3 Functional Requirements

**FR01- Background Music:** The background music feature shall interact with the users during playing the game.

**FR02- Sound Effect:** The sound feature shall allow selecting sound volume options which include master sound volume, music volume, and ringtone.

# **4.** **External Interface Requirements**

## **4.1** **User Interfaces**

Hearthstone Lite uses 2D GUI from Netbeans. Every screens have their own background images including buttons.

## **4.2** **Hardware Interfaces**

Hearthstone Lite using keyboard inputs for text data, along with using the mouse to select items and options in the program such as buttons.

## **4.3** **Software Interfaces**

The software interface includes Java GUI, it allows objects in the application to interact with each other.

## **4.4** **Communications Interfaces**

Hearthstone Lite is actually not like the online game, so it doesn’t have any interfaces for communicate.

# **5.** **Other Nonfunctional Requirements**

## **5.1** **Performance Requirements**

## Hearthstone Lite utilized the Java GUI system, require to use Java to run the program.

## **5.2** **Safety Requirements**

The application should be safe and secure which can prevent loss, damage, and harm for the product. The application should not cause any stress or tension on the user which can cause loss or damage. The application should be safety friendly to fulfill the needs of the users in a good manner.

## **5.3** **Security Requirements**

The application should be secured which cannot be accessible by unauthorized users. It should maintain the privacy of the users through maintaining authorized session access. The user authentication requirements include the login module for the application. The weak security and phishing attack can break the security of the product. The application should support encryption and SSL certifications to make it secure.

## **5.4** **Software Quality Attributes**

Software quality attributes include nonfunctional properties. The non-functional properties required for this product include availability, correctness, flexibility, maintainability, portability, reliability, reusability, testability, efficiency and usability. These quality attributes help to make the product accurate, safe and reliable.

# **6.** **Other Requirements**

There currently no other requirements outside of the previously documented items.

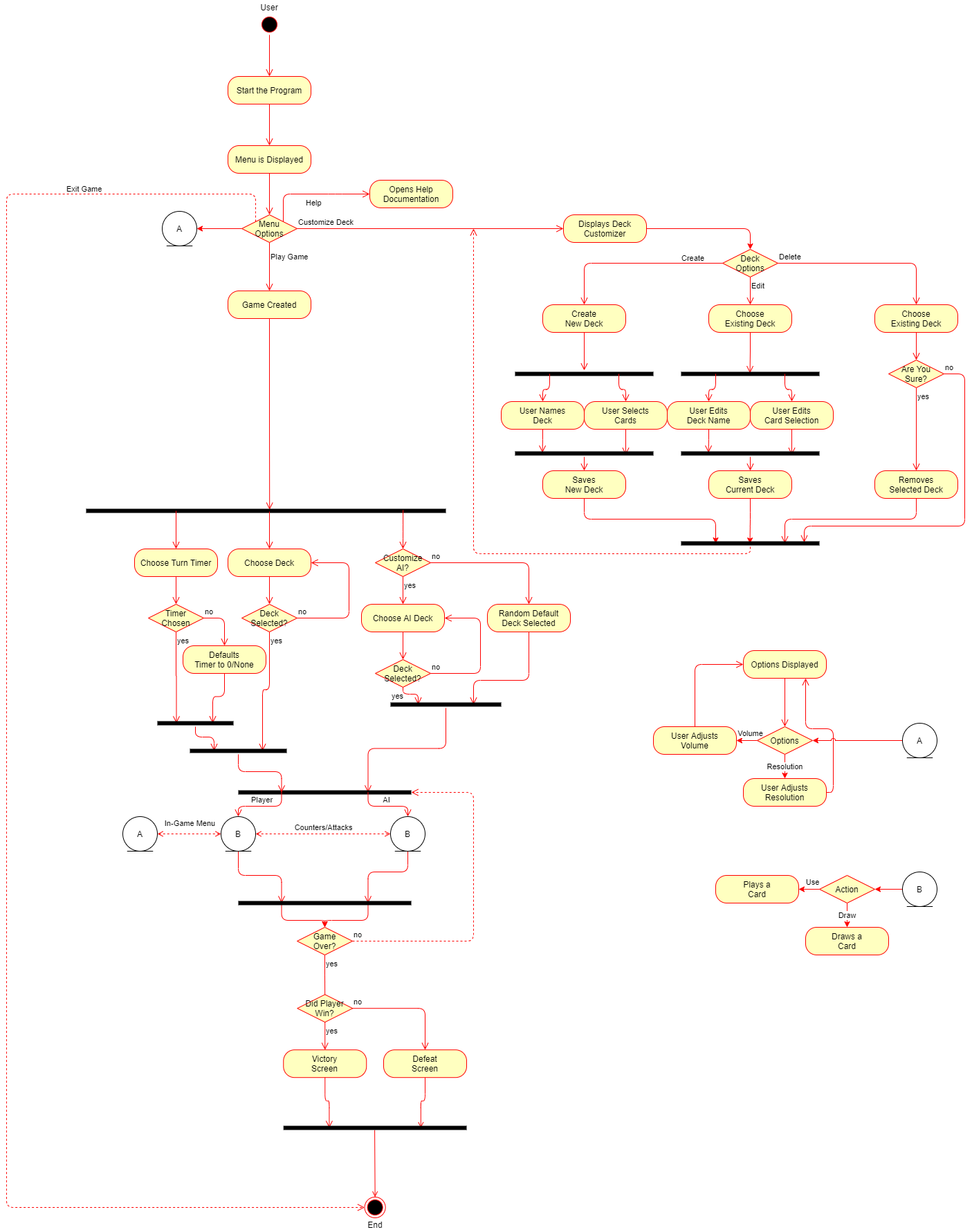
This is subject to change with the progress of development.

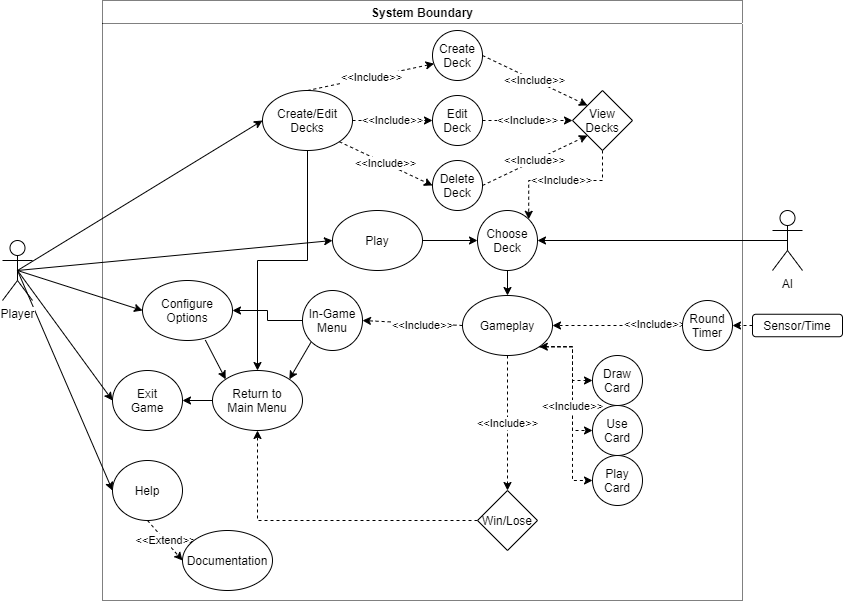
### Appendix A: Glossary

SDK: Software Development Kit

TBD: To Be Decided (information to be provided further in development)

### Appendix B: Analysis Models





A close up of text on a white background

Description automatically generatedA close up of a map

Description automatically generated

### Appendix C: Issues List

* TBD: Deck Size
* TBD: Card Cost
* Software Intensity of AI
* AI Decision Making
* Sound Selection
* Database
* Deck/Card Design