# SOFTWARE REQUIREMENTS SPECIFICATION

## SFRWENG 3XA3

## February 12, 2022

**Safwan Hossain** **Eamon Earl** **Tyler Magarelli**

**hossam18** **earle2** **magarelt**

# Contents

# Revision History

Table 0.1: **Revision History**

| p3cmp2cmX **Date** | **Version** | **Notes** |
|---|---|---|
| Date 1 | 1.0 | Notes |
| Date 2 | 1.1 | Notes |

# 1 Introduction

## 1.1 Purpose

Poker is a world renowned card game in which players must use their understanding of probability, wit and deceptive strategy to best their opponents and win money. While poker is very commonly played face-to-face on a table, the goal of this project is to make a variety of poker game-types easily accessible from anywhere, without betting, and for all ages. The current software we begin with sets the framework for randomizing poker hands and analyzing them, and by expanding upon this foundation we will be able to develop a complete poker experience.

## 1.2 Intended Audience

This SRS is intended for the project managers, developers, and users.

## 1.3 Stakeholders

### The Customer / User

The primary stakeholder to consider during the development process is the subgroup that will be directly using and interfacing with the finished product; the customer. Our main goal for this relationship is to cater to them a full poker experience, and an easy way to engage in this neglected sport with no financial cash flow required from them for the purposes of betting. We will also be providing online play, via private online lobbies, so that the customer can play with their friends. Random match-making could be a later implementation to further improve the user experience, but this goal will lay outside the scope of the project, and will be discussed later on in this document.

### The Publisher

For the purpose of this project, the publisher is considered to be the Department of Computing and Software at McMaster University, wherein their input will be queried and provided throughout the documentation and development process, and will most notably affect this SRS at revision 1, which will be finalized by March 18th.

### The Developer

As our project will be created in a Java environment, the software developers are provided the correct environment to be able to easily adapt and tailor the software towards a simple yet effective interface with high cohesion and low coupling, while providing the user an easy way to play whenever desired with very little initial setup. The developers are also responsible for testing the program and internally managing the project.

## 1.4 Project Constraints

The scope of this project will, to some degree, be inherently described by the constraints, and as such they will be listed and discussed first.

Description: Our software will be able to run on any machine that is capable of running JVM on their system.

Rationale: Any semi-modern local computing device has the capability to run java programs, and as such we will be keeping our demographic of eligible customers as large as possible.

Fit Criterion: Our program will be entirely enclosed within java software and architecture, including the server implementation. Our UI will be text based and represented in the console.

Description: Our program will not involve gambling using real-world currency.

Rationale: Including this feature would conflict with the constraints specified by our publishers, and would greatly extend the scope of this project, to a degree that would require far more resources to complete.

Fit Criterion: Poker chips will be rewarded equally to all players for any game, and will merely be represented in data; we wish to focus on the practical and mental benefits of the game of poker, not the financial deficits.

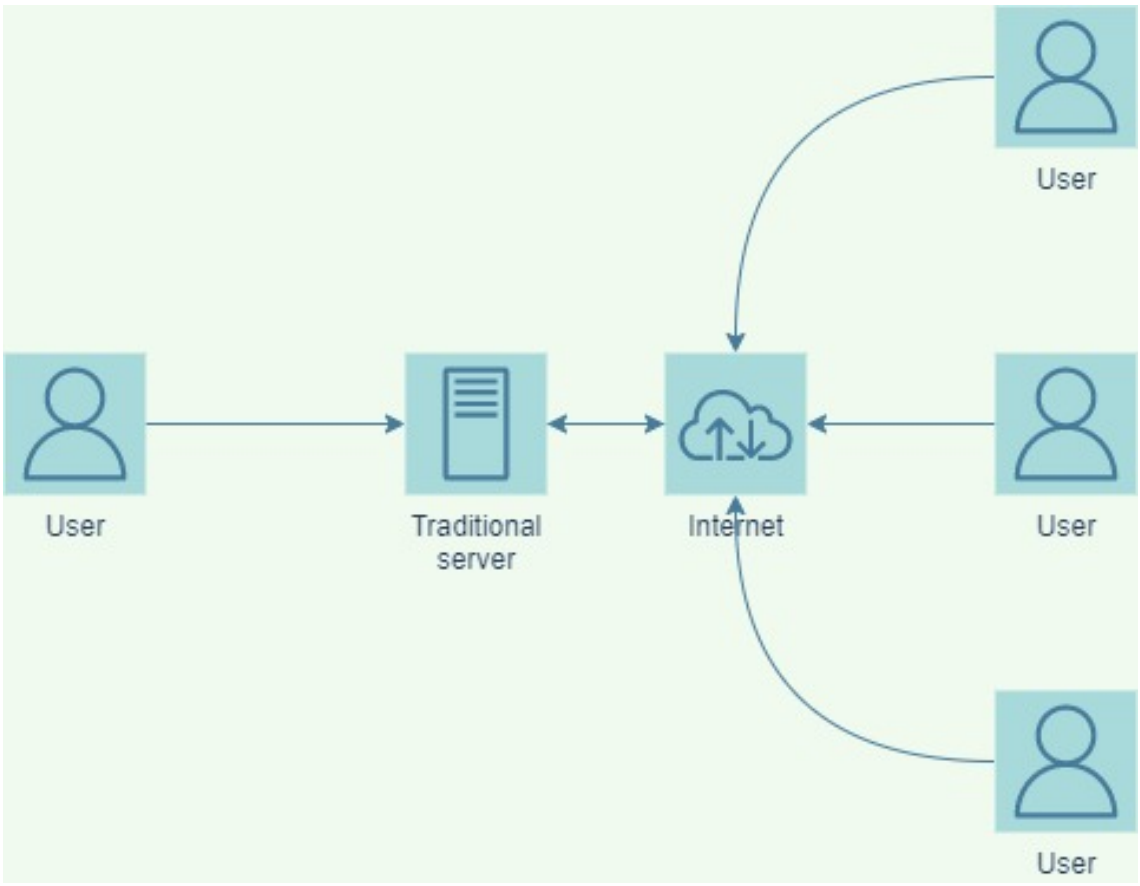
## 1.5 Project Scope

The scope of this project is contained to being held within java framework, and is limited by the deadlines and relatively tight schedules of the students who take the roles of developers. We will implement one main poker game mode, and do so in such a way that adding more in future iterations of the project would be straightforward, and would not require reconfiguration of the existing architecture, merely additional controllers; thus designing for generality. As we will be implementing "Texas Hold'em" as the initial game mode, the program will be inherently a multiplayer experience, with offline local multiplayer and online multiplayer as well. The servers for online multiplayer will be implemented with low fidelity and bandwidth, as it is known by the project managers that at this point we are not designing for a large player base, and the project is not due to hit markets during the current project life cycle.

As previously discussed, using real currency for the betting process is outside of the scope of this project.

## 1.6 Implementation

### 1.6.1 Environment



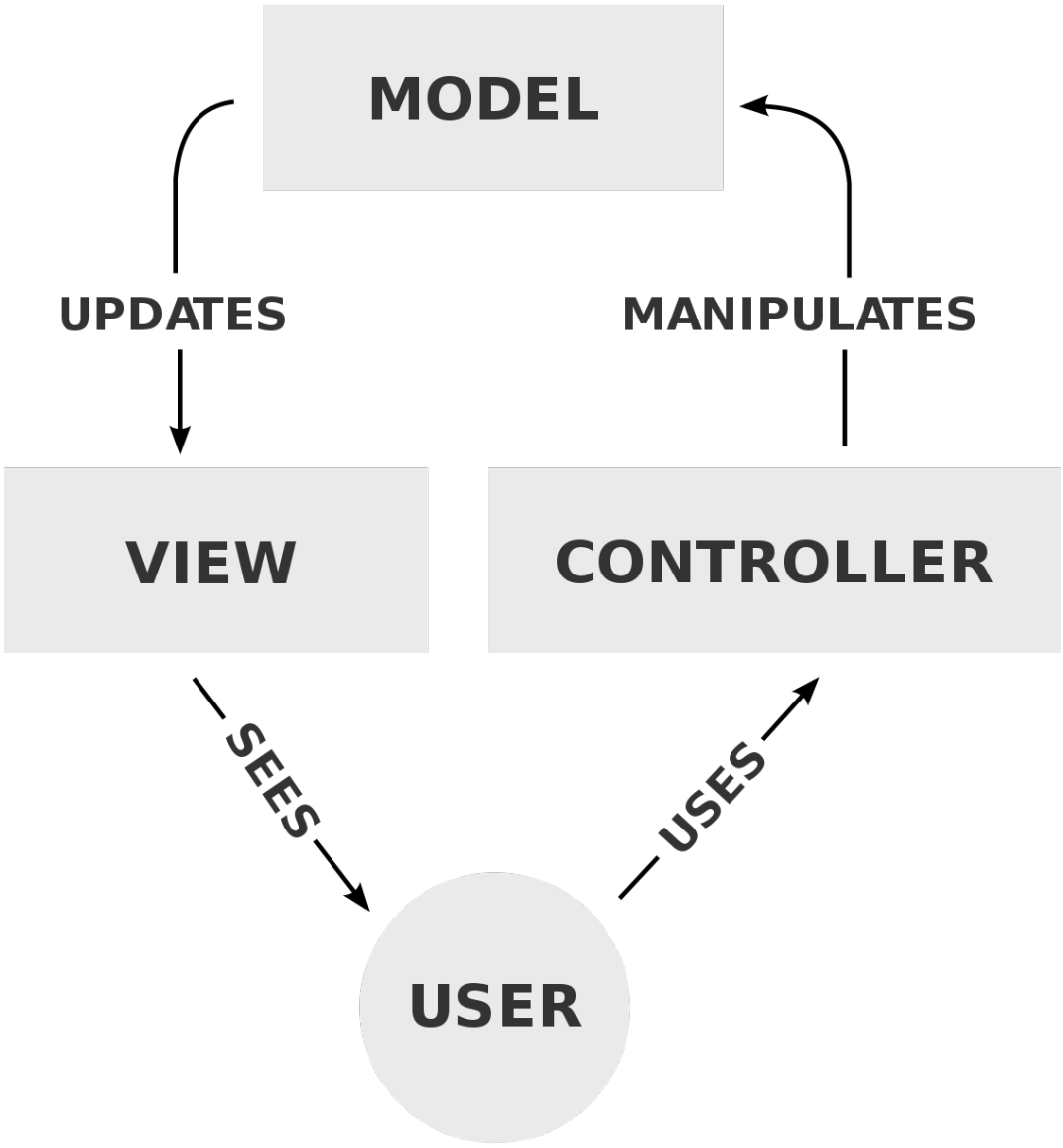

Server Architecture

Above, we see the general architecture of the online aspect of the game, where each user will access a shared server via their respective internet connections. Once again, this server will be low fidelity and will be created entirely within Java.

### 1.6.2 Architecture

We jave decided on the Model-View-Controller (MVC) architecture, as it is well geared towards playable games. It also gears itself towards designing for change in the sense of the generality required to implement other game modes in the future; they would use

easily use the same model, as they are all played with one deck of cards and one hand per player, and would only require a specific controller to script the game (and possibly some additional methods in the view). This allows future developers to leave the entirety of the data untouched - an exceptionally valuable trait for developing pre-existing software.




MVC Architecture -
`https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller`

### 1.6.3 Risks

Some technical risks that need to be weighed out are the implications of running a low fidelity java server off of one of the developer's local machines. This will be something that needs to be tested during the proof of concept such that we can see if our intended number of concurrent online players is feasible under such conditions, else we may wish to consider some off-the-shelf solutions for our server.

Some of these off the shelf solutions might include:

1. Hostinger - a cheap cloud based virtual server with varying payment plans for different requirements
2. InterServer - a fairly priced server for hosting powerful Java applications (could be more sophisticated than we need)
3. Hostwinds - requires more knowledgeable server developer, but grants more in-depth control and customization, with great service almost internationally

# 2 System Features

"IICT WEBSITE" is a result processing web software. So the main art of this product is to enter data of results and publish.

## 2.1 Description and Priority

"IICT WEBSITE" has features that are main and also some are sub. But all the feature is necessary for this software.
The features with priority up to down -

1. Result Creation : This is the goal feature of this software. It is been operated by teachers. They input the results of the students.
2. Result Re-view : This is done by director.
3. Result Publish : It is basically notice. This is mainly done by director or sometimes by staffs commanded by director.
4. Teacher : Teacher takes the courses and create the results.
5. Student : Student takes the courses of a program.
6. Staff : Their only work is to post notice.

## 2.2 Functional Requirements

FR1. The user shall be able to generate a unique alphanumeric code that will allow other users to join their lobby.

FR2. The user shall be able to join lobbies by entering in a valid alphanumeric code.

FR3. The user shall be able to perform at most one of the following actions during their turn: fold, check, call, bet, raise.

FR4. The system shall automatically fold any player that does not make a move within the allotted period of time during their turn.

FR5. A game shall have a specified maximum amount of players.

FR6. A game shall have a specified minimum amount of players.

FR7. The system shall be able to identify the correct card ranking for a player's hand.

FR8. The system shall carry a virtual standard deck of cards, consisting of 52 unique cards.

FR9. The system shall shuffle its deck of cards before every game.

FR10. At the start of the game, the system shall give a random player the role of the dealer. Note that this is just a title and the system will deal the cards, not the player.

FR11. Each round, the first player to the immediate left of the dealer shall be given the role of small blind.

FR12. Each round, the first player to the immediate left of the small blind shall be given the role of big blind.

FR13. The system shall take half of a specified amount of money from the small blind and add it to the money pool in the beginning of the round.

FR14. The system shall take a specified amount of money from the big blind as it did from the small blind and add it to the money pool in the beginning of the round.

FR15. The system shall give each player on the table 2 cards in the beginning of the first round.

FR16. The player to the immediate left of the big blind shall be given the first turn.

FR17. The system shall allow at most one user to have a turn any time during a game.

FR18. After a player is done their turn, the player to the immediate left of them shall be given a turn.

FR19. Once every player has either bet the same amount or decided to fold the round ends.

FR20. After the first round, the system shall present three cards from its deck face up to all of the players.

FR21. After the second round, the system shall present one card from its deck face up to all of the players.

FR22. After the third round, the system shall present one card from its deck face up to all of the players.

FR23. After the fourth round, each player shall be forced to present their cards face up to all the other players.

FR24. During the fifth round, the player with the highest hand ranking will win the pot.

FR25. If the highest hand ranking belongs to more than one player, then the pot will be split among the highest ranking players.

FR26. After the fifth round, all cards will be collected and the game will start again from round one.

FR27. The system shall give each player a specified amount of money when they join.

FR28. If a player loses all their money they will be kicked from the table.

FR29. If a player does not have enough money to call a bet they will be allowed to go all in.

FR30. If a player does not have enough money to pay for big or small blind, they will be allowed to go all in.

# 3 Other Nonfunctional Requirements

## 3.1 Performance Requirements

### 3.1.1 Speed and Latency Requirements

PR1. Any interface between a user and the system shall have a maximum response time of 500ms.

PR2. The maximum time to boot the game on any system shall be less than 2 minutes.

PR3. The maximum time to attempt connection to an open lobby will be 1 minute, and the attempt will be force quit if the connection is not made by then.

### 3.1.2 Safety-Critical Requirements

N/A.

### 3.1.3 Precision or Accuracy Requirements

N/A.

### 3.1.4 Reliability and Availability Requirements

PR1. The system shall run 7 days a week, 24 hours a day.

### 3.1.5 Robustness or Fault-Tolerance Requirements

PR1. The system shall be designed to operate on a highly robust level.

### 3.1.6 Capacity Requirements

PR1. The system will handle at most 100 users at a given time.

PR2. Each lobby / local game can have at most 6 individual players.

### 3.1.7 Scalability or Extensibility Requirements

PR1. The system shall be able to update through software updates that are issued by the developers.

### 3.1.8 Longevity Requirements

PR1. The product will be developed under the heuristics of designing for change via generality.

## 3.2 Security Requirements

### 3.2.1 Access Requirements

SR1. The alphanumeric codes for lobby creation and access will be encrypted / generated by some means such that unwanted players cannot easily join.

### 3.2.2 Integrity Requirements

SR1.

### 3.2.3 Privacy Requirements

SR1. The lobby creators will have the option to kick players out of a lobby before the game commences.

### 3.2.4 Audit Requirements

SR1.

### 3.2.5 Immunity Requirements

SR1.

## 3.3 Usability Requirements

### 3.3.1 Ease of Use Requirements

1.

### 3.3.2 Personalization and Internationalization Requirements

1.

### 3.3.3 Learning Requirements

1.

### 3.3.4 Understandability and Politeness Requirements

1.

### 3.3.5 Accessibility Requirements

1.

## 3.4 Installability Requirements

"IICT WEBSITE" is for storing students results and publish those results of particular programs.
Basically save working time and pressure.

# 4 Other Requirements

"IICT WEBSITE" needs maintenance as it is a long process software. It will need refactoring and further the requirements can be changed as the field is changing frequently.