# **Project Document**

# Three Card Poker Game Application

## Version 1.3

## Last revised: 11/19/2017

## CPSC 4900 FA17

## Prepared by:

UTC Poker Club

*James Hare - Project Manager*

*Loro Dumo - System Analysis Manager*

*Javier Martinez - Quality Assurance Manager*

*Richard Kimsey - Development Manager*

*Daniel Velasquez - Application Architect*

Three Card Poker Game Application

SECTION 1 - INTRODUCTION Page 2

1.1 PURPOSE

1.2 SCOPE

1.3 TERMINOLOGY

1.4 REFERENCES

SECTION 2 – MANAGEMENT Page 4

2.1 GENERAL DESCRIPTION

2.2 SOFTWARE VALIDATION

2.3 CONFIGURATION MANAGEMENT PLAN

2.4 PROJECT SCHEDULE

SECTION 3 - REQUIREMENTS Page 6

3.1 FUNCTIONAL REQUIREMENTS

3.1.1 Functional Capabilities

3.1.1 System Inputs

3.1.2 System Outputs

3.2 NON FUNCTIONAL REQUIREMENTS

3.2.1 Performance Considerations

3.2.2 User Interface

3.2.3 Language

3.2.8 Operating Environment

SECTION 4 – DESIGN Page 8

4.1 Configuration Design

4.2 Technical Design

4.3 SYSTEMS ARCHITECTURE

SECTION 5 - IMPLEMENTATION STRATEGY Page 12

SECTION 6 - USER DOCUMENTATION Page 13

6.1 INSTALLATION GUIDE

6.2 USER'S GUIDE

SECTION 7 - TEST AND VALIDATION Page 14

SECTION 1 - INTRODUCTION

The UTC Poker Club team was founded to develop a Three Hand Poker Gaming Application to address consumer interest in computer-based card games. The Three Hand Poker product delivered in the first iteration will consist of a windows console application which operates without the need for an Internet connection. The startup menu will have options to play other popular poker games in a casino environment. However, only Three Card Poker will be implemented in the first iteration. The group of target users for this application indicated that their preference is for a simple game interface without complicated graphics that will be played during waiting periods and the UTC Poker Club team has developed the applications requirements with these requirements in mind.

1.1 PURPOSE

The purpose of this document is to capture and record the basic information that will be needed to correctly define and plan the Three Card Poker application. This document will expand upon the scope and specification of the project and state what the project is planning to achieve. It also lists the names of the development team along with their roles and responsibilities. This document also specifies and defines irregular terminology that both users and the client may not be familiar with. It will also specify both functional and nonfunctional requirements of the project. It will lay out a design plan for the development team and the client to review. Finally, it will specify user documentation, including installation and user guides along with software testing and validation.

1.2 SCOPE

The initial implementation will consist of a fully playable Three Hand Poker game. The game platform will also be able to hold multiple different poker games from the start up menu but those will not be included in the initial iteration of the product. The game will be playable on a Windows Desktop operating system in the form of an executable windows console application. The game will be limited to the Windows Desktop operating system and will not be made available on other operating systems or mobile applications.

1.3 TERMINOLOGY

This document makes use of acronyms and terms that may not be understood by the reader. Therefore, the following provides definitions for terms relevant to this document [1].

ANTE: A small bet all players are required to make before a hand is dealt. An ante is similar to a blind, but everyone has to contribute it before a hand commences. Antes give the pot a value right off the bat.

ALL-IN: A bet that places all of a player's chips into the pot.

BLIND: The generic term for either the big blind or the small blind. If you are one of the blinds, you are sitting either immediately to the left of the dealer button (small blind position) or one position farther left (big blind).

BURN: The discarding of the top card before each betting round. In the case that there is a distinguishing mark on the top card, the burn card keeps the next card to be dealt concealed before it comes out. That way no unfair information is being intentionally or unintentionally conveyed.

CALL: To contribute the minimum amount of money to the pot necessary to continue playing a hand.

CHECK To pass on betting. If there's no action (bet) to you, there's nothing to call. If you don't want to bet, you can just "check." If there's subsequent action from your fellow players in the betting round, then the action will come back to you to either call, fold or raise.

COMMUNITY CARDS: Cards that are dealt face up in the center of the table, available for all players to use in making a hand.

DEALER: The player who shuffles the deck and deals the cards.

FLUSH: A poker hand consisting of three (3) cards of the same suit.

FOLD: To give up by placing your cards face down on the table, losing whatever you have bet so far. You only fold when you think your hand is too weak to compete against the other players.

HEADS-UP: Playing a pot or tournament against only one other player.

LIMIT: A structure of the game in which bets and raises are capped at a fixed amount.

NO-LIMIT: A structure of the game in which players can bet their entire stack. There's a minimum to what you can bet, but not a maximum.

PAIR: Two cards of the same rank.

POCKET CARDS: The cards in your hand that are not part of the community cards.

RAISE: To wager more than the minimum required to call, forcing other players to put in more money as well.

STRAIGHT: A hand consisting of three (3) cards in sequence but not in suit.

STRAIGHT FLUSH: A hand consisting of three (3) cards in sequence and the same suit.

1.4 REFERENCES

[1] Speak poker as well as you play it. (n.d.). Retrieved November 08, 2017, from http://www.wsop.com/poker-terms

SECTION 2 – MANAGEMENT

2.1 GENERAL DESCRIPTION

The following is a list naming the key people involved in the development of the project, along with their roles and responsibilities:

James Hare (Project Manager) is responsible for leading the development process and maintaining the project schedule. Also responsible for organizing and leading all Scrum sessions as well as overseeing compilation and delivery of the project document. Finally, responsible for reporting project status to upper management.

Daniel Velasquez (Application Architect) is responsible for designing the overall vision for the solution. Will lead in the development and maintenance of the System Architecture Specification. Finally, is responsible for coordinating with traceability of design and development.

Richard Kimsey (Development Manager) is responsible for leading the development and maintenance of the System Design Specification. Will also coordinate developer resources. Finally, is responsible for code production and delivery.

Javier Martinez (Quality Assurance Manager) is responsible for leading in developing and maintaining the Test Plan and Test Specifications. Is also responsible for coordinating text activities. Finally, is responsible for the validation of developed software.

Loro Dumo (System Analysis Manager) is responsible for organizing and leading requirements elicitation activities. Is also responsible for leading in the development and maintenance of the Requirements Specification. Finally, is responsible for coordinating traceability of requirements and design.

2.2 SOFTWARE VALIDATION

Through the use of software validation, we will be able to assess whether or not we are developing the right product. Software Validation decisions will be made by the Quality Assurance Manager, who will also coordinate text activities, develop and maintain test plans and test specification. Software will be validated by using a given set of test cases as defined in section seven (7) of this document, which will reflect the expected use of the system. If the system performs in the correct manner, the software will meet the requirements.

2.3 CONFIGURATION MANAGEMENT PLAN

Configuration Management decisions will be made by the Project Manager. Configuration implementation and maintenance will be handled by the Application Architect. Configuration management tasks will include Visual Studio IDE configuration and necessary component installations on the developer’s devices. The Application Architect will also be responsible for source control configuration and version control through the GitHub client. Software Validation tools will be configured and managed by the Quality Assurance Manager.

2.4 PROJECT SCHEDULE

The Three Hand Poker project schedule will be documented in the form of a Gantt Chart. The Project’s start date is September 26, 2017, and the schedule is to be organized by a series of sprints excluding the kick off tasks undertaken prior to Sprint 1. The Project Manager will be responsible for project activity scheduling, documenting task progress, and updating task dates. An updated tracking of each sprint, including completed tasks, will be documented in the form of a Sprint Report.

SECTION 3 - REQUIREMENTS

3.1 BUSINESS REQUIREMENT SPECIFICATION (BRS)

The user requirements for this project is a single player poker game that allows the user to play against the robot or handler. The user should be allowed to draw a set of cards and be able to play against the computer. With the ability to fold or draw a hand that will allow the user to either win or lose to the handler. User just needs a system that allows them to run the program that will let them play the poker game. The user is allowed to bet on the amount that they have. The user should be allowed the option to bet based on the hand of cards they have to play. User wants a simple interface that is easy to operate.

3.2 FUNCTIONAL REQUIREMENT SPECIFICATION (FRS)

The system requirements for this project is a Microsoft Windows operating system that will allow them to run the program using Windows Visual Studio. The system or program needs a user to execute the program and run the poker game to be able to play. The system or poker game allows the user to draw or bet on each hand that the user elects to play. The system needs the users wager or bet amount to play its hand. The system needs to know what kind of poker hand the user has to win the wager amount that is bet.

3.2.1 Functional Capabilities

The delivered product will host many popular card games played in a casino environment. The main game of focus will be Three Card Poker. The object of this game is to make the best poker hand possible using only three (3) cards, with each user playing against an Artificial Intelligence (AI) dealer. When gameplay begins, the user must place an ante wager in anticipation of having a winning hand. The player is then dealt three cards for the deck. After looking at his three cards, known as his hand, the player will either fold or make a bet that he has better hands than the dealer. If the player folds then the round is over and the dealer collects the ante wager. If the player opts to make a

bet, the players cards will be compared to the cards of the dealer to determine who has the better hand. Each hand will be based on the following hierarchy of poker hands: Straight Flush (pays 40 to 1), Three of a Kind (pays 30 to 1), Straight (pays 6 to 1), Flush (pays 3 to 1) and Pair (pays 2 to 1).

3.2.2 System Inputs

The user inputs whether how much the user wants to bet against the dealer’s hand. The amount of betting the user can input is based on how much the user has in their balance.

3.2.3 System Outputs

The system outputs whether or not the user beat the dealer or not once the user hits [ENTER]. If the user betted against the dealer and wins. The system will give the user double what the user doubled.

3.3 NON-FUNCTIONAL REQUIREMENTS

The system keeps track of each bet or wager that is placed. The balance that the user has either decreases or increases as the game goes on and also it does the same for the dealer. The system keeps track of each responsive move made in the game. The game is only able to be used on one platform.

3.3.1 Performance Considerations

The applications performance is based on the user input. The user is required to take action by using the keyboard by taking any action the application will respond to the user in a reasonable time frame. Their isn't any constraints that would not allow the user to interact with the system or be able to play game. The application should be able to run on a system that has sufficient requirements that allow it to function correctly.

3.3.2 User Interface

The interface in Three Card Poker is straightforward. The only controls that the user needs to know is to play is [ENTER] and to fold is [SPACEBAR]. The system will automatically bet 5 dollars but it allows the user to set what the user actually wants to bet by using the arrow keys.

3.3.3 Language

The Three Card Poker application will be written in Visual C# (2015) and will implement the ASP.NET Framework. It will also make use of Application Insight tools and Web Tools provided by the Visual Studio Package.

3.3.8 Operating Environment

The Three Card Poker application will operate using the Microsoft Windows Operating System environment. In a future sprint, it will be developed to meet the needs of iOS and Android environments for use on cell phones and tablets.

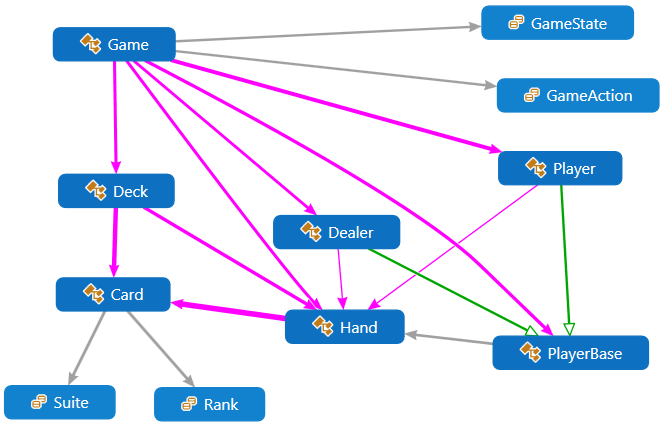
SECTION 4 – DESIGN

4.1 SYSTEMS ARCHITECTURE

High Level Architecture for Three Card Poker include Library and User Interface:



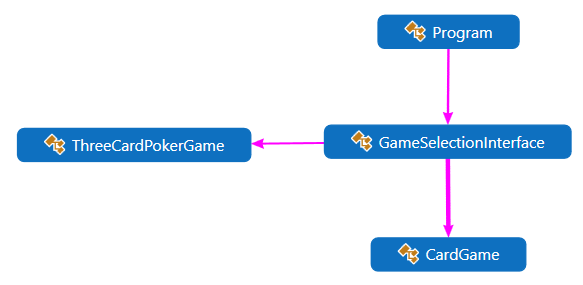
Architecture and dependencies for << Three Card Poker Library >>:



High Level Architecture for Three Card Poker(Continued):

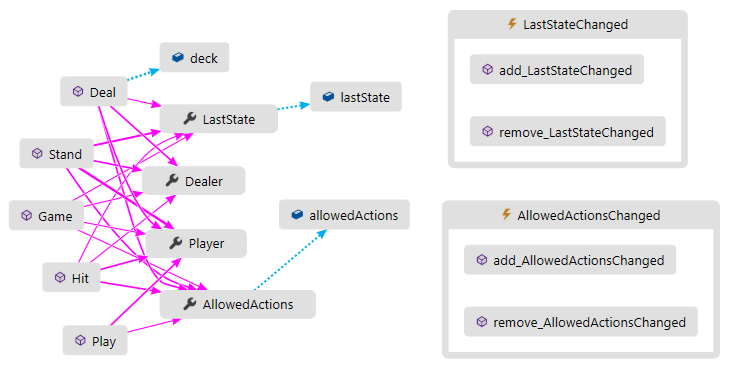


Architecture and dependencies for Game Selection User Interface:

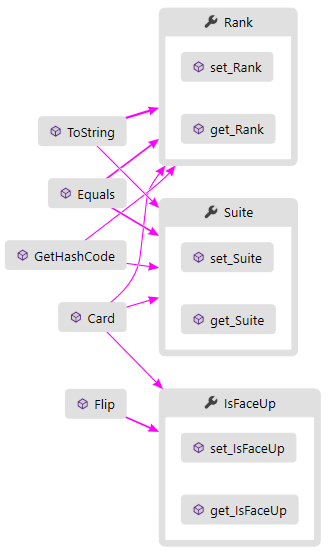


4.2 Configuration Design

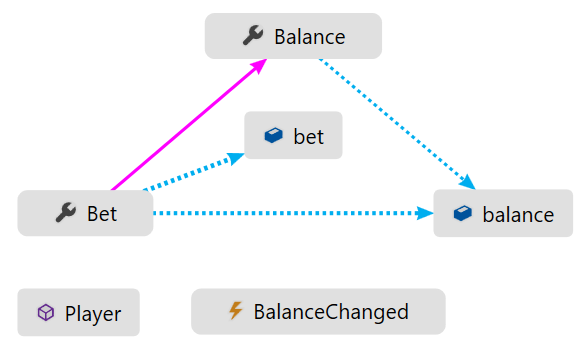
Game Behavior Configuration Design:



Card Behavior Configuration Design:

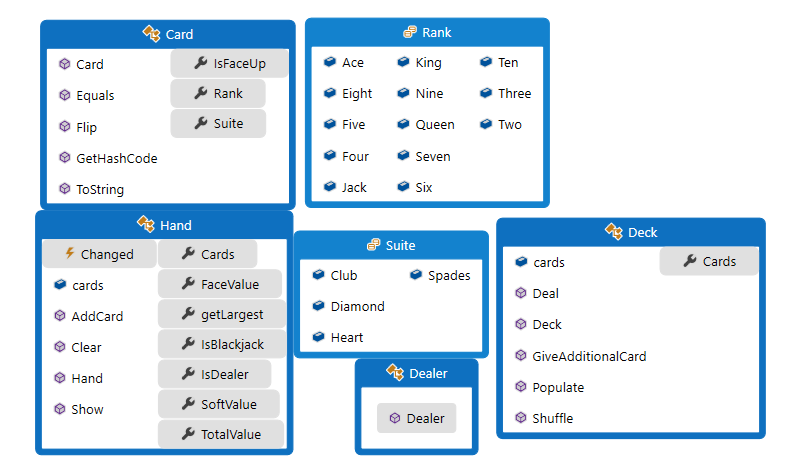


Player Behaviour Configuration Design:

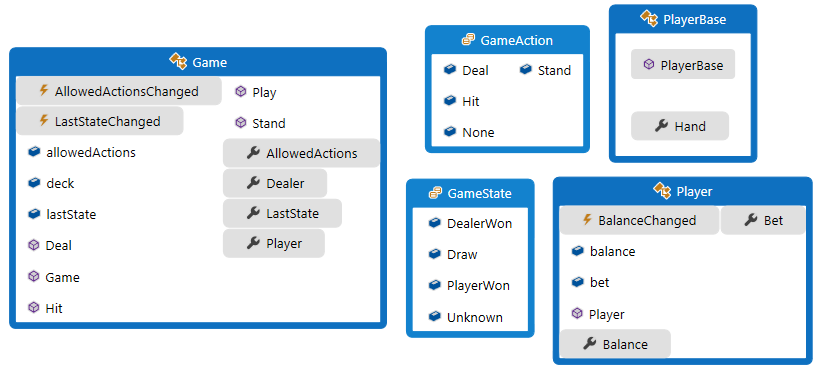


4.3 Technical Design

Technical Design Pattern for Card Objects and Dealer Objects:



Technical Design Patterns for Player Objects and Game State Objects:



SECTION 5 - IMPLEMENTATION STRATEGY

Implementation and design are interleaved activities so that the level of detail and creativity in the design is determined by the limitations of our windows console app as well as time limitations. The strategy we are using to develop the software revolves around being aware of implementation issues such as reuse, configuration management, and host-target. Most of our implementation will be developed using a host-target where we will be developing the software on one machine and use a different machine to execute that software. We are also searching for open source code so we have more ideas to build upon when we make changes and improve our software. A configuration manager will be in charge of keeping track and managing the changes we make to the software. Our implementation is planned so that the first week of the Sprint developers will work on their user stories and the second week of the Sprint will be for Software Quality testing. If the User Stories pass SQA they will move on to a peer review done by another developer on the team who has not involve with the code change. If the User Stories are found without defect they will be merged onto our production environment and the test case documents used to test them will be archived for later inspection if needed. This rigorous quality assurance process is to ensure a quality product is achieved in the end.

SECTION 6 - USER DOCUMENTATION

6.1 INSTALLATION GUIDE

The solution will be given as a windows executable application and can be opened on any Windows environment. The user may install the application on their system by dragging and dropping the application to their machine.

6.2 USER'S GUIDE

To open the application, double click the icon with mouse.

User will be given four different game options. To move to game, use up and down arrows. To add a game to favorites press the F key. To toggle between all games and favorites list, press the spacebar. To play a game press the enter key. To go back to main menu, click the x in upper right corner to close the current game window.

When playing three card poker, the user can choose to play their hand or fold. If you choose to fold, you lose your initial ante of $50. If you play, you must match their $50 ante but will have a chance at winning. This means you will lose $100 overall if you lose, but will win $100 if you win the match.

SECTION 7 - TEST AND VALIDATION

Test Plan for our system to make sure it meets the requirements of our users and to document the improvements of our systems.

Test cases

* Make sure that the dealer deals you three cards.
  + Out of 30 tries, it was 100 percent that the dealer deal the player 3 cards.
  + In those 30 tries, the dealer did not deal less than 3 cards or more than 3 cards.
* Make sure that all hands that are greater than all hands worse wins. You do not want a hand with 2 of hearts, a jack of diamonds, and a 7 of clubs beating out a three of a kind Ace.
  + Out of 30 tries, the cards understand their values on which cards are greater value than the other cards. The hands in the cards also showed their value on which hand was greater compared to the dealer. It understood that a hand that had a pair was greater than one high card.
* The program can shut on and off again without crashing
  + During the 30 tries, the game could successfully to turn on and off again.
* Max and Min bid does not exceed the bid placed or places a lower bid
  + During the 30 tries, I could bid and match bid to the other players. It did not allow you to bid lower the bid place, but you were able to raise when asked.