GambleMania
for UTM CSCI 352

Collin Winstead, Diego Arriaga

Abstract

This app allows the user to play three standard casino games: blackjack, three-card poker, and roulette. The player will play against an AI dealer in the blackjack and three-card poker games. In roulette, the program will randomly generate values from the roulette wheel. None of the games in our app will not require any real money to play; all money is virtual and has no monetary value. Users will have their own accounts which will contain login information and the account's balance. This information will be managed with a database.

1. Introduction

Our project is called GambleMania. It will allow you to play the typical casino games of blackjack, three-card poker, and roulette. The player's balance will be saved in a database along with their login information. We decided to do this project because we thought it would be fun and require a good bit of research and work to finish. As for why we chose the games previously mentioned, it's because they are all fairly simple games to learn and play (probably by design). Because they are fairly simple, they should be relatively easy to program. Our target audience is adults 18 years old and older because the games featured in out app are all typically played in casinos. We expect our audience to use our app simply for fun or to practice blackjack or three-card poker. We did not include roulette in this list because roulette is based solely on randomness, so you can't really practice it.

1.1. Background

Minimally, we expect the user to know how to play blackjack, three-card poker, and roulette. However, we could implement a tutorial as a stretch goal to remove this requirement.

1.2. Impacts

Although our project is very small and limited, if distributed the project could possibly have a cultural impact by getting more people into the games of blackjack, three-card poker, and roulette. It could also potentially be used for data research and game theory by simulating thousands or millions of blackjack, three-card poker, and roulette games.

1.3. Challenges

We foresee the most challenging part of this project as the way the AI dealer will work in the blackjack and three-card poker games. These games have certain rules that need to be checked. For example, in the blackjack game, the dealer must stand on 17. In three-card poker, the player's hand must be checked against the dealer's hand. We think these kinds of obstacles will be our biggest hurdles.

2. Scope

This project will be completed when we have implemented a navigable main menu and the games of blackjack, three-card poker, and roulette. The main menu should minimally allow the user to navigate between games by clicking a button (one for each game) and return to desktop by clicking an exit button. The user must be able to play an accurate game of blackjack with the rules of dealer stands on 17 and blackjack pays 3:2. The user must be able to play an accurate game of three-card poker. A fully functioning roulette table must be implemented. Finally, each of the 56 individual cards in a standard deck of playing cards must be designed.

As mentioned earlier in this paper, there are a few stretch goals we can potentially meet should we be able to do so:

1) Create a tutorial for each game for users who do not know how to play.
2) Add more games to the app.
3) Make the main menu and game environments look better.
4) Create a Texas Hold'em style poker game that can be played over a network connection.

Use Case ID	Use Case Name	Primary Actor	Complexity	Priority
1	Play Blackjack	User	Med	1
2	Play Three-Card Poker	User	Med	1
3	Play Roulette	User	Med	1
4	Watch a tutorial	User	Easy	2
5	Navigate the Main Menu	User	Easy	1
6	Exit the Program	User	Easy	1

TABLE 1. USE CASES

2.1. Requirements

As mentioned in the Scope section, there are many basic requirements that need to be completed in order for this project to be done. There are also a few stretch goals that we could implement should we get done early. The basis of the basic requirements stems from the rules of play for each of the three basic games we will be implementing. There are also requirements that involve the UI design, some of which are simply required for a functional app and others are for a good-looking app. The stretch requirements are things that would directly make the app better, but are not required for the app to function. In the following subsections, we will describe these requirements in greater detail.

2.1.1. Functional.

- A button that exits the program in the main menu and in each of the games.
- Buttons to navigate from the main menu to each of the games.
- A back button in each game that navigates the user back to the main menu.
- Game logic for each game.
- A Microsoft Access database that stores user login information and user balances.

2.1.2. Non-Functional.

- Tutorials for each game
- Visual/animated backgrounds for the main menu and each game.

2.2. Use Cases

This section defines and ranks the use cases for our project. A table of use cases is shown in Table 1.

2.3. Interface Mockups

Figure 1. Login/sign-up GUI mockup

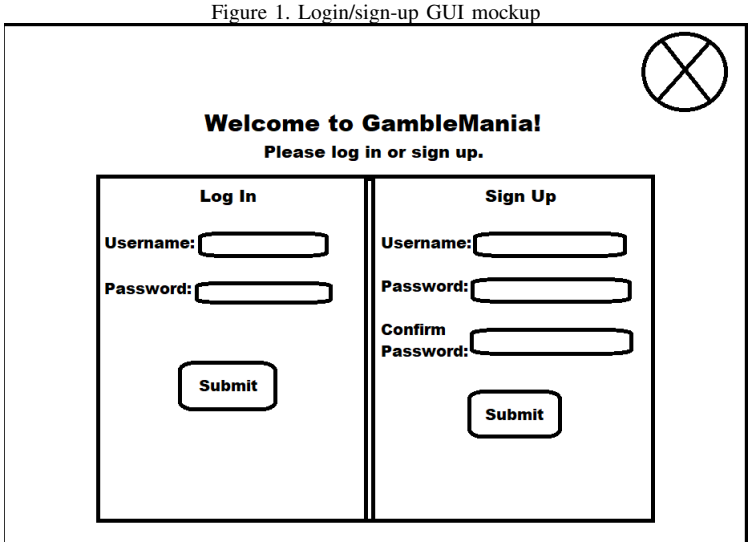

Figure 2. Main menu GUI mockup

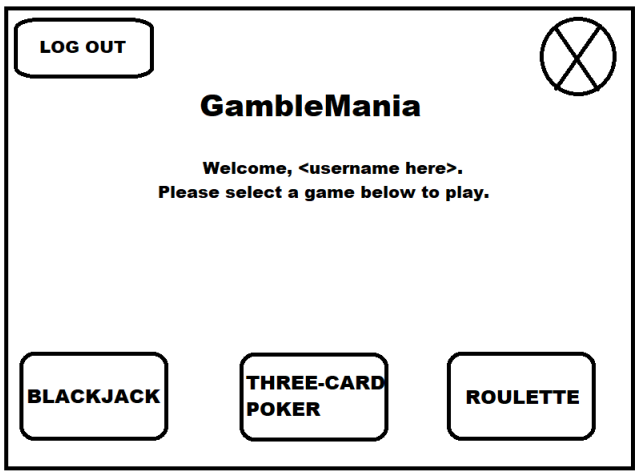

Figure 3. Blackjack GUI mockup

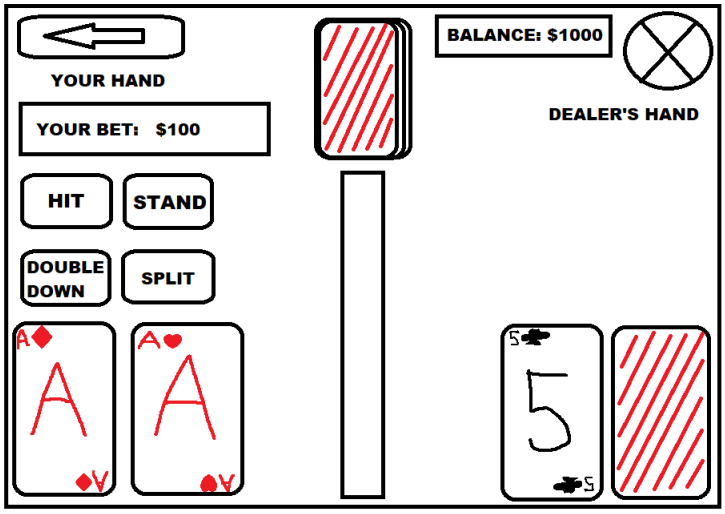

Figure 4. Three-card poker GUI mockup

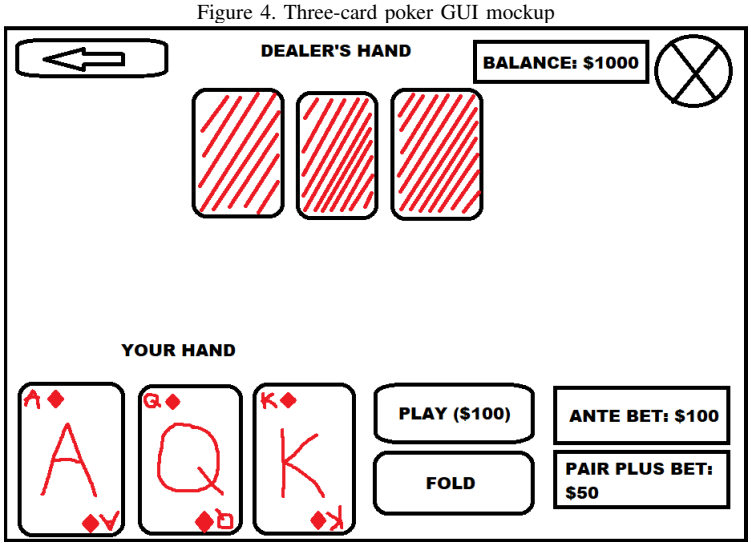

Figure 5. Roulette GUI mockup