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	Log In	User	Easy	1
2	Sign Up	User	Med	1
3	Play Blackjack	User	Easy	3
4	Play Three-Card Poker	User	Easy	4
5	Play Roulette	User	Hard	5
6	Watch a Tutorial	User	Easy	6
7	Navigate the Main Menu	User	Easy	2
8	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 non-functional requirements are based on how the user will interact with our app. These requirements are not necessarily required for the app to function, but are important for the users of our app.

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.
- User login data and balance is hashed.
- The app should be responsive and perform well without any major dips in framerate.
- When the user's balance is modified, it should be updated in the database as soon as possible.
- The app should be flexible enough so that it is possible to scale it up in the future.

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.

2.2.1. Use Case 1: Log In. See Figure 1
1) User enters their username and password into the provided text fields.
2) User clicks submit.
3) If the username and password are found in the database, the user proceeds to the main menu. Otherwise, the user must keep entering the username and password until the username and password matches.

2.2.2. Use Case 2: Sign Up. See Figure 1
1) User enters the username and password for their new account.
2) User clicks submit.
3) Account is registered to the database and the user proceeds to the main menu with the new account. The new account will have a starting balance of $1000.

2.2.3. Use Case 3: Play Blackjack. See Figure 3
1) User places an ante bet.
2) User and dealer are dealt two randomly chosen cards.
3) User is shown a maximum of four options: Hit, Stand, Double Down, and Split. Double Down is only shown on the first turn. Split is only shown on the first turn when the user has two cards of the same face.
4) If the user chooses the Hit option, the user will be dealt another card.
5) If the user chooses the Stand option, the dealer will begin their turn.

6) If the user chooses the Double Down option, the user will place a bet with the same value as the ante bet and be dealt another card. After this, the dealer's turn begins.

7) If the user chooses the Split option, the user's two cards will be split into separate hands. Each hand will be dealt another card, giving the user two hands with two cards each. The user can play each hand separately.

8) After the user's turn is over, the dealer's turn begins. The dealer uses the same options the user has.

9) After the dealer's turn is over the user's hand and the dealer's hand are compared. The hand with the closest point value to 21 is the winner.

10) After the game is over, the user may enter another ante bet or exit to the main menu.

2.2.4. Use Case 4: Play Three-Card Poker. See Figure 4

1) User places an ante bet and optionally a pair plus bet.
2) User and dealer are dealt three cards.
3) User is shown two options: Play and Fold.
4) If the user chooses the Play option, the user will make a bet that has the same value as the ante bet. After this, the user's turn will be over.
5) If the user chooses the Fold option, the user will cede their hand to the dealer and lose automatically.
6) After the user's turn is over, the dealer's turn begins. The dealer will reveal their cards and compare its hand with the user's.
7) If the dealer's hand is not better than Queen high or the user's hand is better, the user wins. Otherwise, if the dealer has the better hand, the dealer wins.
8) After the game is over, the user may enter another ante bet/pair plus bet or exit to the main menu.

2.2.5. Use Case 5: Play Roulette. See Figure 5

1) User places a bet of their choice on any valid space on the board.
2) Once the user is done betting, the user may click the "Spin" button to spin the wheel.
3) Once the ball in the roulette wheel rests in a slot, that slot determines what values win.
4) The user is paid out according to what bet they made and where the ball landed.
5) The user may choose to place another bet or exit to the main menu once the payout - or lack thereof - is done.

2.2.6. Use Case 6: Watch a Tutorial.

1) In the main menu, the user may click the button labeled "Watch a Tutorial."
2) Once in the tutorial screen, the user may select any of the tutorials associated with the three games in the app.
3) Once the user clicks on a tutorial, they will be shown a video that teaches the user how to play the game.
4) Once the user is done watching the tutorial, they may go back to the tutorials screen.
5) From the tutorials screen, the user may choose to watch another tutorial or exit to the main menu.

2.2.7. Use Case 7: Navigate the Main Menu. See Figure 2

1) The user can click the "Blackjack" button to go to the blackjack game screen.
2) The user can click the "Three-Card Poker" button to go to the three-card poker game screen.
3) The user can click the "Roulette" button to go to the roulette game screen.
4) The user can click the "Log Out" button to log out of their account and go back to the log in/sign up screen.
5) The user can click the "X" button at the top right corner of the screen to exit the program.

2.2.8. Use Case 8: Exit the Program. See Figure 2

1) The user can click the "X" button at the top right corner of the main menu screen to exit the program.

2.3. Interface Mockups

Figure 1. Login/sign-up GUI mockup

Welcome to GambleMania!

Please log in or sign up.

Log In

Username: [   ]

Password: [   ]

Submit

Sign Up

Username: [   ]

Password: [   ]

Confirm
Password: [   ]

Submit

Figure 2. Main menu GUI mockup

LOG OUT

GambleMania

Welcome, <username here>.
Please select a game below to play.

BLACKJACK

THREE-CARD
POKER

ROULETTE

Figure 3. Blackjack GUI mockup

DEALER'S HAND

5♣
5

BALANCE: $1000

HIT | DOUBLE DOWN | STAND | SPLIT | YOUR BET: $100

YOUR HAND

A♦ A
A♥ A
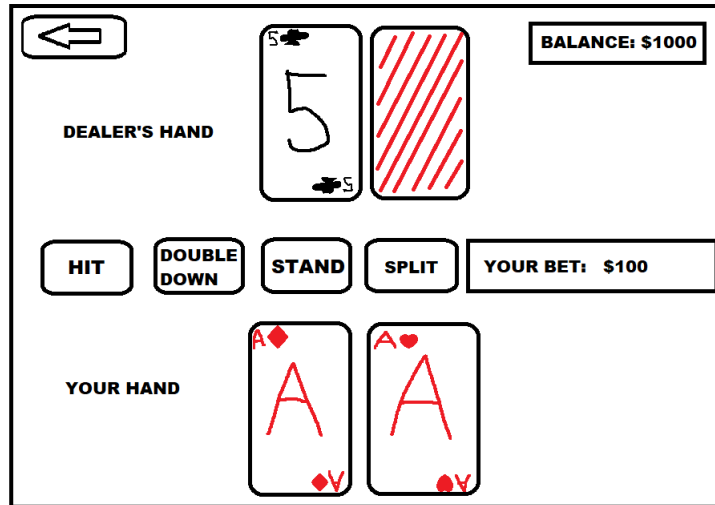
Figure 4. Three-card poker GUI mockup

DEALER'S HAND

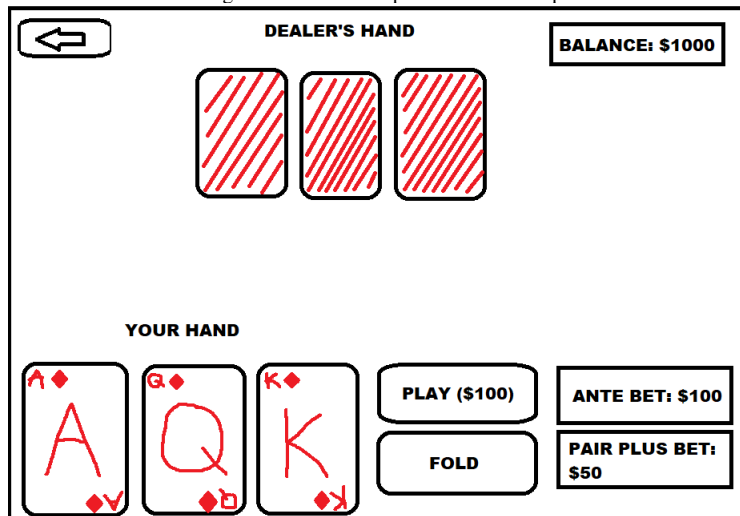BALANCE: $1000

YOUR HAND

A♦ A
Q♦ Q
K♦ K

PLAY ($100)

FOLD

ANTE BET: $100

PAIR PLUS BET: $50

Figure 5. Roulette GUI mockup