

# Black Jack

## CS482 Software Engineering Team Assignment 2

Jenna Borowy, Emma Heiser, Chris Plowman, Callie Walker  
Client: Dr. Nguyen Ho

2024-10-09

## Contents

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Introduction TBD</b>                                       | <b>3</b>  |
| <b>2</b> | <b>Requirements</b>   | <b>3</b>  |
| 2.1      | User Roles . . . . .  | 3         |
| 2.1.1    | Guest User . . . . .  | 3         |
| 2.1.2    | Account User . . . . .  | 3         |
| 2.1.3    | Table Host . . . . .  | 4         |
| 2.1.4    | Admin . . . . .   | 4         |
| 2.2      | Functional Requirement User Stories and Tasks [TBD] . . . . . | 4         |
| 2.3      | Non-Functional Requirements [TBD] . . . . .                   | 12        |
| 2.3.1    | Category: User Interface . . . . .                            | 12        |
| 2.3.2    | Category: Security . . . . .                                  | 12        |
| 2.3.3    | Category: Performance . . . . .                               | 12        |
| <b>3</b> | <b>Iterations [TBD]</b>                                       | <b>12</b> |
| 3.1      | Iterations 1 . . . . .  | 13        |
| 3.1.1    | Plan . . . . .  | 13        |
| 3.1.2    | Activities . . . . .  | 13        |
| 3.1.3    | Retrospective . . . . .                                       | 13        |
| <b>4</b> | <b>Final System Architecture and Design</b>                   | <b>13</b> |
| 4.1      | System Architecture . . . . .                                 | 13        |
| 4.2      | Coding Standards . . . . .                                    | 13        |
| 4.2.1    | Comments . . . . .  | 13        |
| 4.2.2    | Headers . . . . .   | 13        |
| 4.2.3    | Variables . . . . .   | 14        |
| 4.2.4    | Database . . . . .  | 14        |
| 4.3      | Technologies Used . . . . .                                   | 15        |
| 4.3.1    | Languages . . . . .   | 15        |
| 4.4      | UI Mockups . . . . .  | 16        |
| <b>5</b> | <b>Testing</b>  | <b>17</b> |
| <b>6</b> | <b>Reflections on the Project [TBD]</b>                       | <b>17</b> |

## Abstract

Our project is a Black Jack app for our client Dr. Nguyen Ho. Our client's company is in the market for a profitable Black Jack app that supports online social aspects such as friends, messaging and table creation.

# 1 Introduction TBD

[[ No more than one page that provides a short project description and then sets the stage for the rest of the document. ]]

## 2 Requirements

### 2.1 User Roles

1. Guest User
2. Account User
3. Table Host
4. Admin

#### 2.1.1 Guest User

A Guest User is an user that wishes to play anonymously, or without signing in. A Guest will have limited actions compared to that of other user types, such as an Account User. A Guest will be able to view the tutorial, join games and tables, play games and bet. However, Guests will not be able to add friends, have access to a friends list, they will not keep winnings after they play, they will not be able to make a table or game, and they will not be able to message Admins.

#### 2.1.2 Account User

An Account User is a user who is logged in with an account and has the ability to add friends, remove friends, message friends and view a friends list. Additionally, an account user can create tables and games (which for that table, makes them a Host), they keep earnings from games and can view their account history.

### 2.1.3 Table Host

A Table Host is a specialized Account User. The only time an Account User is promoted to a Table Host is when that user decides to create a table from the lobby. At that time the Account User gains special permissions for that table such as being able to set min/max bet, max number of seats, betting increments etc. Once the Table Host leaves that table, it then goes back to being a regular Account User.

### 2.1.4 Admin

An Admin Account can do everything that every other user type can. Additionally, an Admin can change the credentials and reset the password of an Account User. Admin Accounts also have a special inbox for messages and recommendations from Account Users.

## 2.2 Functional Requirement User Stories and Tasks [TBD]

These are our user stories organized by user type then priority:

### Guest-only Actions:

#### 1. : S3: Create Account (2pts)

- (a) Priority: High
- (b) **I want** to create an account **As a guest So that** I can play with an account.
- (c) **Definition of Done:** I have an account.
- (d) **Depends On:** Im a guest user.
- (e) Tasks: S3: Create Account: Make a new account

#### 2. : S7: Play As Guest (2pts)

- (a) Priority: High

- (b) **I want** to play as a guest **As a** guest user **So that** I can play anonymously
- (c) **Definition of Done:** I'm playing as a guest.
- (d) **Depends On:** user wants to play anonymously.
- (e) Tasks: S7: Play as guest: user wants to play as a guest

#### **Admin-only Actions:**

##### 1. : **S24: Reset Password (2pts)**

- (a) Priority: High
- (b) **I want** to reset a user's password **As an** Admin account **So that** I can reset a user's password.
- (c) **Definition of Done:** User has a new password.
- (d) **Depends On:** (S1,S3)
- (e) Tasks: S24: Reset Password: reset a user's password

##### 2. : **S25: Modify Profile (2pts)**

- (a) Priority: Medium
- (b) **I want** to edit a user's profile **As an** Admin account **So that** I can update a profile
- (c) **Definition of Done:** User's profile is updated.
- (d) **Depends On:** User has an account (S3)
- (e) Tasks: S25: Modify Profile: Admin edits a user's profile.

#### **Account-User-Only Actions**

##### 1. : **S1: Login (2pts)**

- (a) Priority: High

- (b) **I want** to login to my account **As an** Account-User **So that** I can play with my account.
- (c) **Definition of Done:** User is logged into account.
- (d) **Depends On:** (S3)
- (e) Tasks: S1: Login: login to account

## 2. : S2: Logout (1pt)

- (a) Priority: High
- (b) **I want** to logout of account **As an** Account-user **So that** I can logout of my account.
- (c) **Definition of Done:** User has logged out of account.
- (d) **Depends On:** user is logged in (S1,S3)
- (e) Tasks: S2: Logout: logout of account

## 3. : S5: Add Friend (1pt)

- (a) Priority: High
- (b) **I want** to add a friend **As an** Account-user **So that** I can have that user registered as a friend.
- (c) **Definition of Done:** Friend shows up in list.
- (d) **Depends On:** (S1,S3)
- (e) Tasks: S5: Add Friend: Add friend to friends list

## 4. : S6: Remove Friend (1pt)

- (a) Priority: High
- (b) **I want** to remove a friend from friends list **As an** Account-user **So that** I can remove a friend from my list.

- (c) **Definition of Done:** Friend is in list.
- (d) **Depends On:** User is in friends list (S5)
- (e) Tasks: S6: Remove Friend: Remove friend from list

#### 5. : S4: View Friends List (1pt)

- (a) Priority: High
- (b) **I want** to view my friends list **As an** Account-user **So that** I can see my friends list
- (c) **Definition of Done:** I can see my friends list on screen
- (d) **Depends On:** user has friends added (S5)
- (e) Tasks: S4: View Friends List: open and view friends list

#### 6. : S8: Create Table (2pts)

- (a) Priority: High
- (b) **I want** to create a table for blackjack **As an** Account-user **So that** I can make a new table for blackjack
- (c) **Definition of Done:** There is a new table in the lobby
- (d) **Depends On:** User is in lobby
- (e) Tasks: S8: Create Table: create a new table for blackjack

#### 7. : S23: Send Message to Friends (2pts)

- (a) Priority: Medium
- (b) **I want** to message friends **As an** Account-user **So that** I can chat with my friends
- (c) **Definition of Done:** The chat is sent and my friend can see it

- (d) **Depends On:** user has a friend (S5)
- (e) Tasks: S23: Send Message to Friend: send a message to friend

8. : **S17: Send Chat (2pts)**

- (a) Priority: Low
- (b) **I want** to reset a user's password **As an** Account-user **So that** I can send a chat to the table.
- (c) **Definition of Done:** The chat is sent to the table.
- (d) **Depends On:** User is at table (S10)
- (e) Tasks: S17: Send Chat: user sends chat to table

9. : **S22: Message Admin (2pts)**

- (a) Priority: Low
- (b) **I want** to send an admin a message **As an** Account-user **So that** I can contact an admin
- (c) **Definition of Done:** The message is sent and the admin can see it
- (d) **Depends On:** (S1)
- (e) Tasks: S22: Message Admin: send message to admin

10. : **S19: View Game History (2pts)**

- (a) Priority: Low
- (b) **I want** to view my game history **As an** Account-user **So that** I can see my wins and losses
- (c) **Definition of Done:** I see the number of games I've won and lost.
- (d) **Depends On:** User has played at least 1 game and is in lobby (S12)

- (e) Tasks: S19: View Game History: see games won and loss

**Any-User Actions:****1. : S21: View Tutorial (1pt)**

- (a) Priority: High
- (b) **I want** To read the tutorial **As** Any user **So that** I know how to play the game
- (c) **Definition of Done:** user is at the tutorial page
- (d) **Depends On:** user is in lobby
- (e) Tasks: S21: View Tutorial: redirected to a page on how to play blackjack

**2. : S20: Request Chips (2pts)**

- (a) Priority: High
- (b) **I want** to request chips **As** Any user **So that** I can get chips when I've ran out
- (c) **Definition of Done:** I get chips
- (d) **Depends On:** User is out of chips
- (e) Tasks: S20: Request Chips: get chips when I've run out

**3. : S10: Join Table (2pts)**

- (a) Priority: High
- (b) **I want** to join a table **As** Any user **So that** I can be in a table for blackjack
- (c) **Definition of Done:** when Im in the blackjack table
- (d) **Depends On:** user is in lobby
- (e) Tasks: S10: Join Table: user joins a blackjack table

**4. : S12: Join Game (2pts)**

- (a) Priority: High
- (b) **I want** to join a game **As** Any user **So that**
- (c) **Definition of Done:** user joins the next game of blackjack
- (d) **Depends On:** user is at table (S10)
- (e) Tasks: S12: Join Game : join a game of blackjack at a table

5. : **S13: Stand (1pt)**

- (a) Priority: High
- (b) **I want** to stop receiving cards **As** Any user **So that** I can keep my point value where it is
- (c) **Definition of Done:** I stop receiving cards
- (d) **Depends On:** Player is in a game (S10,S12)
- (e) Tasks: S13: Stand : stop receiving cards during game

6. : **S14: Hit (1pt)**

- (a) Priority: High
- (b) **I want** to receive another card **As** Any user **So that** I can try and get more points in blackjack
- (c) **Definition of Done:** I get another card
- (d) **Depends On:** user is in a game
- (e) Tasks: S14: Hit: user gets another card

7. : **S11: Leave Table (2pts)**

- (a) Priority: Medium
- (b) **I want** to leave the blackjack table **As** Any user **So that** I will no longer be at that table

(c) **Definition of Done:** user is no longer at that table

(d) **Depends On:** user is at a table (S10)

(e) Tasks: S11: Leave table : user leaves the blackjack table

8. : **S9: View Table info (2pts)**

(a) Priority: Medium

(b) **I want** to view the table info **As** Any user **So that** I can see the specific details of a table

(c) **Definition of Done:** the table's details are shown

(d) **Depends On:** user is in lobby

(e) Tasks: S9: View Table info: view table's information

9. : **S15: Change Ace value (1pt)**

(a) Priority: Medium

(b) **I want** to change the ace value in my hand **As** Any user **So that** My ace can equal 1 or 11

(c) **Definition of Done:** When the value of my hand is updated correctly

(d) **Depends On:** user has an ace and is in game (S10,12)

(e) Tasks: S15: Change Ace Value: make the ace in my hand 1 or 11

10. : **S16: Open chat (1pt)**

(a) Priority: Low

(b) **I want** to see the table chat **As** Any user **So that** I can see the table's chat

(c) **Definition of Done:** table chat appears on screen

(d) **Depends On:** user is at table (S10)

### 3 ITERATIONS [TBD]

---

(e) Tasks: S16: Open chat : view table's chat

11. : **S18: Close chat(1pt)**

(a) Priority: Low

(b) **I want** to close the table's chat **As** Any user **So that** I don't see the table chat anymore

(c) **Definition of Done:** user no longer sees table chat

(d) **Depends On:** chat is open (S16)

(e) Tasks: S18: Close chat: close the table's chat screen

## 2.3 Non-Functional Requirements [TBD]

[[ Complete the following template for each non-functional requirement.

**ID: Title**

1. Description:

2. Affects:

Categorize the non-functional requirements using subsections using as many categories as your need. Please put a blank line between requirements.]]

### 2.3.1 Category: User Interface

### 2.3.2 Category: Security

### 2.3.3 Category: Performance

## 3 Iterations [TBD]

[[ The requirements analysis identifies **what** your client wants. With each iteration describe the **how**. Include user stories attempted, data structures introduced, and any key design decisions.

Copy this layout for each iteration. ]]

### 3.1 Iterations 1

#### 3.1.1 Plan

#### 3.1.2 Activities

#### 3.1.3 Retrospective

## 4 Final System Architecture and Design

### 4.1 System Architecture

Based on our project's requirements and the components involved, we have decided to use the MVC architecture for our project. We believe that the MVC architecture will best let us integrate the major parts of our project (Web UI, functional logic, and database) effectively.

### 4.2 Coding Standards

#### 4.2.1 Comments

Comments (outside of function headers) will always be above the line of code it pertains to, never on the same line. We will aim to keep comments clear, concise and in plain-english.

#### 4.2.2 Headers

- Function Headers

For functions, we will have consistent headers (in plain-english) including:

- Input: what the function receives as input.
- Output: what the function produces.
- Purpose: what the function does.

- Reliance: function that directly relies on this function, and function that this function directly relies on if any. (for testing purposes)

- File Headers

For file headers we will have a brief description of what the file is and what it contains.

### 4.2.3 Variables

Following the same standard across all of our code, all variables will be Pascal case (camel casing with first capital letter) and singular. This guarantees consistency in all variable nomenclature.

### 4.2.4 Database

For our database, our table names will be Pascal case and singular. The primary keys in our database will be camel case followed by 'ID'. The foreign keys will also be camel case with a descriptor followed by 'FK'.

Here is an ER diagram of our database:

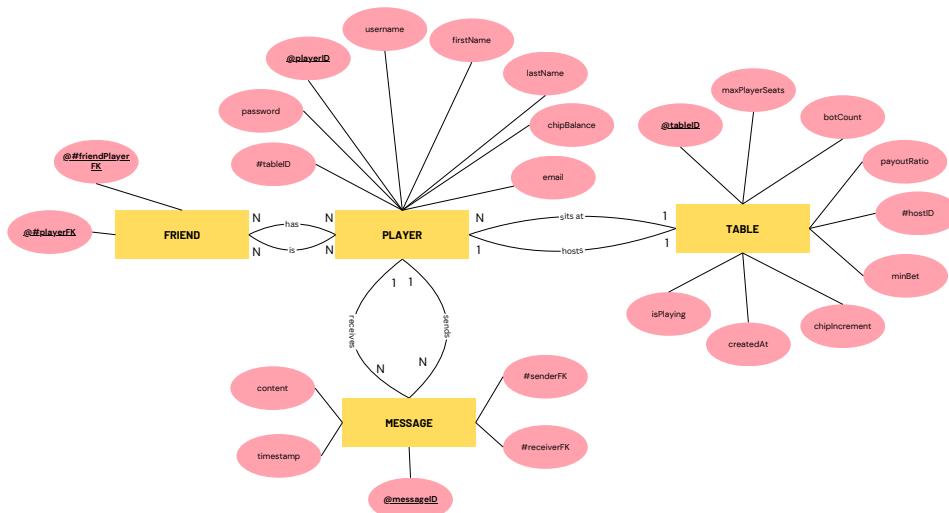


Figure 1: ERD for our database tables.

### 4.3 Technologies Used

### 4.3.1 Languages

For the language to run our functional logic we have chose Java as it integrates well with the other languages in our project. For our database components we will be using MySQL along with MySQL community server and MySQL workbench. For our web UI we will be using React along with Next JS for our server.

Here is our UML class diagram that outlines the major classes and relations in our software:

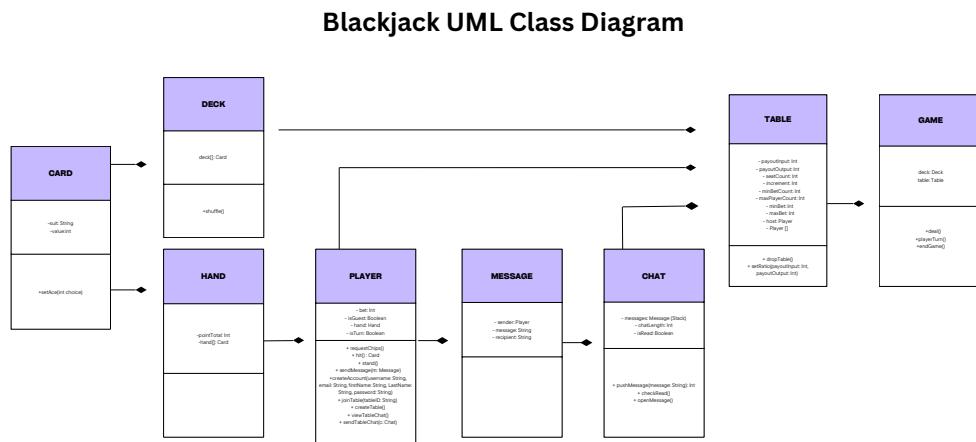


Figure 2: Caption

## 4.4 UI Mockups

Here are some of the initial UI designs for our project:



Figure 3: This is the UI mockup for the logo of the app (illus. Emma Heiser)

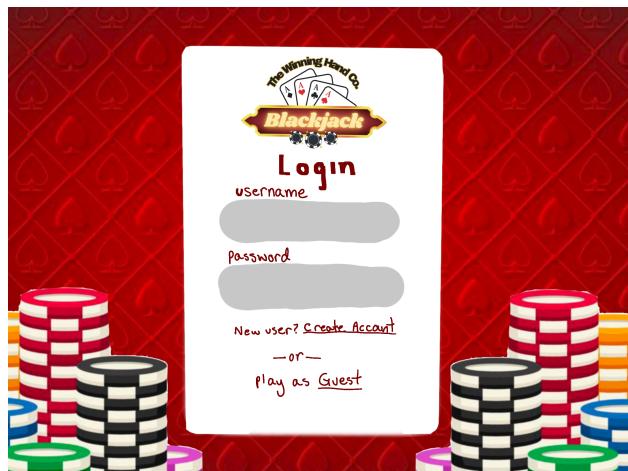


Figure 4: This is a UI mockup of the login screen (illus. Emma Heiser)

## 6 REFLECTIONS ON THE PROJECT [TBD]

---



Figure 5: This is the UI mockup of creating a table (illus. Emma Heiser)

## 5 Testing

For our testing we will be using JUnit. We will be focused on writing good tests rather than achieving a set percentage required to commit. We believe that this is a reliable way to ensure that good software is being developed. However, if we *had* to give a numerical value, we decided that 75% coverage would be sufficient to commit.

## 6 Reflections on the Project [TBD]