SW Engineering CSC648

Blackjack Game development

Section 01 Team 05

Seng Maw **–** Team leader & Scrum Master

Pritam Gautam **–** Frontend lead

Sneha Shrestha **–** Frontend lead

Lane Maimone **–** Backend lead

Nathaniel Miller **–** Backend lead

Tyler Hsieh **–** Backend lead & GitHub Master

Milestone 1

09.27.2022

|  |
| --- |
| **Executive Summary** |
| **Personas and User stories** |
| **Data Definitions** |
| **Functional requirements** |
| **Non-functional requirements** |
| **Competitive analysis** |
| **High-level system requirements** |

Green= Completed, Red= In progress

1. **Executive summary :**

Card games have been in human society since the 9th century, and it has been a fun activity for humans to release stress and bond relationships because it is easy to learn and accessible. By recent research on the effect of card games, it tends to help with brain functionality such as improve memory and reduce the risks of dementia. Among different card games, Blackjack is one of the most popular games because it is based on simple math where the player’s decisions matter in the game. It is also the one game that can earn a living. The thrill that comes from the possibility of counting games adds up other potentials.

For this project, we are developing an online Blackjack game that targets the audience of age 13+. The proposed functionality would be to feel as close to a real blackjack game as possible and be able to access from all around the world.  We are aiming to provide the multiplayer function and open to future extensible function as online casinos. The player will be able to see the other players as icons when playing in multiplayer mode which ensures the players to feel like they are in real casinos.

Our team is composed of six members, young and passionate computer science students who are excited to learn along the developing journey and very enthusiastic about card games strategy and user interfaces. All of our team members are senior students who can’t wait to fly out to the real world and show what we can offer to the development industries.

1. **Personas and User Stories**

Name: Ryan (Retired Senior)

Ryan is a retired senior citizen who likes to play online games. He usually likes to play card games like Blackjack. He does not like to gamble money but likes to have competition among players so that he can enjoy playing it. Every time he browses to play online games, he must buy coins which he does not like to do. He would love to play where he does not have to buy coins but have a leader board where players can get their rank based on the free coins they win or lose and have alternative ways to get those coins if the player loses it all.

|  |  |
| --- | --- |
| Shape  Description automatically generated | **Name: Ryan (Retired Senior)**  **Age: 65**  **Location: San Francisco**  **About Him:**  Ryan is a retired senior citizen who likes to play online games. He usually likes to play card games like Blackjack. |
| **Goals:**  Want to have fun playing games.  Do not want to spend money on a card game.  Want to play a game that is more competitive. | **Motivation:**  Want to play an online card game.  Want to play for free but still have the same excitement as playing by betting money.    He wants to see the leaderboard of the game. |

Name: Paul (Online gamer)

Paul is 25 years old online gamer. Out of all the online card games, he loves to play Blackjack game. Most of the time when he goes online and tries to play games, he finds himself playing with bots. He feels like those boots are made in the way that they can beat humans easily. He gets frustrated to play with bots because he thinks it's not fair to play with bots with the human mind. He would love to play with his friends online.

|  |  |
| --- | --- |
| Icon  Description automatically generated | **Name: Paul (Online gamer)**  **Age: 25**  **Location: Berkeley**  **About Him:**  Paul is an online gamer who loves playing card games online. |
| **Goals:**  Want to have fun playing games with friends.  Do not want to encounter bots while playing online. | **Motivation:**  Want to play an online card game.  Get experience like playing with a real person. |

Name: Karen (Gambling Fein)

|  |  |
| --- | --- |
| Icon  Description automatically generated | **Name: Karen (Gambling Fein)**  **Age: 71**  **Location: Livermore**  **About Her:**  Karen is a grandmother who has retired from gambling as it was once a huge time consumer for her. She however has been looking for a casual game to play online. |
| **Goals:**  Want to reconnect to her passion of online gambling without the consequences of losing real money.  Wants something that will be easy to use as she isn’t tech savvy. | **Motivation:**  Fun games that her and her friends can share together online.  Games that have low risk and have some sort of competitiveness to them. |

**3. Data Definitions**

1. Object structure/class for cards:

* structure/class will have variables for number attributes depending on the card, etc: King has a value of 10 and 4 has a value of 4. Ace will have a value of 1 or 11 and follow an if/else tree to handle the different scores that could be associated with an ace.
* Card class may also have color variables and face variables for display purposes.
* Number variable is int.
* cardStatus is Int variable for if the card is in play, discard, or in the deck.
* All other variables are string.

1. Stack/Array of card objects with a max size that is a multiple of 52, depending on game mode, etc: Deck[52], Deck[104]
2. int turnNum to count turn
3. Object structure/class for players/dealer: the player struct/class will be the parent of the dealer.
   * player has: vector<card> Hand[] for cards in hand,
   * int totalValue for total value of Hand[],
   * int Score for player's Score that's stored in database,
   * int playerStatus for if player is ingame, or out of the game.
   * int Credits for player credits.
   * the dealer does not have credits or score, and is an AI.

**E**.   **Functions**:

* + player.hit(),
  + player.pass(),
  + endTurn(),
  + startGame(),
  + checkScore(),
  + Deck.shuffle(),
  + deal(),
  + endGame(),
  + playerOut(),
  + updatePlayerRank()

**4. List of Functional requirements**

1. **Create Account**

When the player first joins the site they will be prompted to sign in/create an account/play as a guest. Players who are registered with the website will be able to participate in the leaderboard and have special promotions.

1. **Join Game (single player or multiplayer)**

Single players will be hosted on their local machine (local browser). The multiplayer will be hosted off of the web server. Multiplayer mode will allow players to join either random lobbies or queue up with their friends.

1. **Difficulty**

Three modes of difficulty will be incorporated with the initial release. - Easy mode will be single deck blackjack. This mode will have a minimum and maximum bet that will be much smaller than the other modes. Medium mode will be 5-8 decks of cards but will have a higher minimum and maximum bet than easy mode.Hard mode will also have 5-8 decks of cards, will have the highest minimum and maximum bet (players could potentially bet their entire pot), and the dealer will hit on a soft 17.

1. **Start/End Round of Blackjack**

This function will be necessary because it will need to take into account all players' bets with the database as well as update their leaderboard position at the beginning and end of the round. Start round will either create the deck or update the deck from the previous round (depending on how many cards have been played).

1. **Shuffle/Split Deck**

This requirement allows for the instances of the cards to be created as well as shuffle them like a real deck of cards for the fairness of the game. Splitting the deck is a key feature of blackjack in real life so it is important for us all for random deck splitting.

1. **Update Leaderboard**

Updating the leaderboard will show players where they stand locally, regionally, and globally. Top ten players will be displayed for each of these leagues. The whole point of our game is to have players compete against each other, not just the house like in traditional blackjack.

**5. List of non-functional requirements**

* Compatibility with most browser including Chrome and firefox in mobile version as well.
* For our Blackjack game we decided to  use Amazon AWS using SQL to store our data.
* Graphical interface with buttons (hit, stand, split, double, and then clickable chips to raise/lower a individual players bet).
* All the front end and the backend should be in a separate folder in the GitHub repository.

**6**. **Competitive Analysis**

Competitive Features**:**Game engine, Themed art style, Web and Mobile compatibility within browser

|  |  |
| --- | --- |
| **Competitor Features** | **Our Features** |
| Visual effects are done through the browser resulting in low frames per second | By using a game engine in the back end, the game will have better overall performance |
| There is an assumption that the user went to the site knowing how to play blackjack | Creating a tutorial/how-to popup on the homepage will help new visitors |
| Typical unattractive casino assets | A themed art style to make the casino for our blackjack unique |
| Number of decks but no difficulty settings for players | Difficulty setting will be based on how many decks of cards are selected and whether the dealer hits on a certain hand. |
| One-on-one blackjack. Player vs NPC Dealer | One-on-one against a dealer as well as a multiplayer feature so players can play at the same table with/against each other. |

**7. High level requirements**

Cloud Server – AWS

Web Server - Apache

Operating System – Ubuntu 22.4 Database – MySql

Front-end technology – Javascript, React  Back end – C++

Framework – Rogueengine.io

**8. Team**:

**Seng Maw - Team Leader & Scrum Master**

**Pritam Gautam - Front end developer**

**Sneha Shrestha - Front end developer**

**Tyler Hsieh – Back-end developer**

**Lane Maimone – Back-end developer**

**Nathaniel Miller – Back-end developer**

**Study group plan – (** Seng – Executive summary, Pritam – Personas and Stories, Tyler – Data Definitions, Lane – List of functional requirements, Sneha – List of non-Functional requirements, Nate – Competitive analysis and High-level requirements )

**9.Checklist:**

|  |  |
| --- | --- |
| Team found a time slot to meet outside of the class. | **DONE** |
| Scrum Master shares meeting minutes with everyone after each meeting. | **DONE** |
| Github master chosen | **DONE** |
| Everyone sets up their local development environment from the team’s git repo. | **DONE** |
| Team decided and agreed together on using the listed SW tools and deployment server | **DONE** |
| Team ready and able to use the chosen back/front-end frameworks. | **DONE** |
| Team lead ensured that all team members read the final M1 and agree/understand it before submission |  |