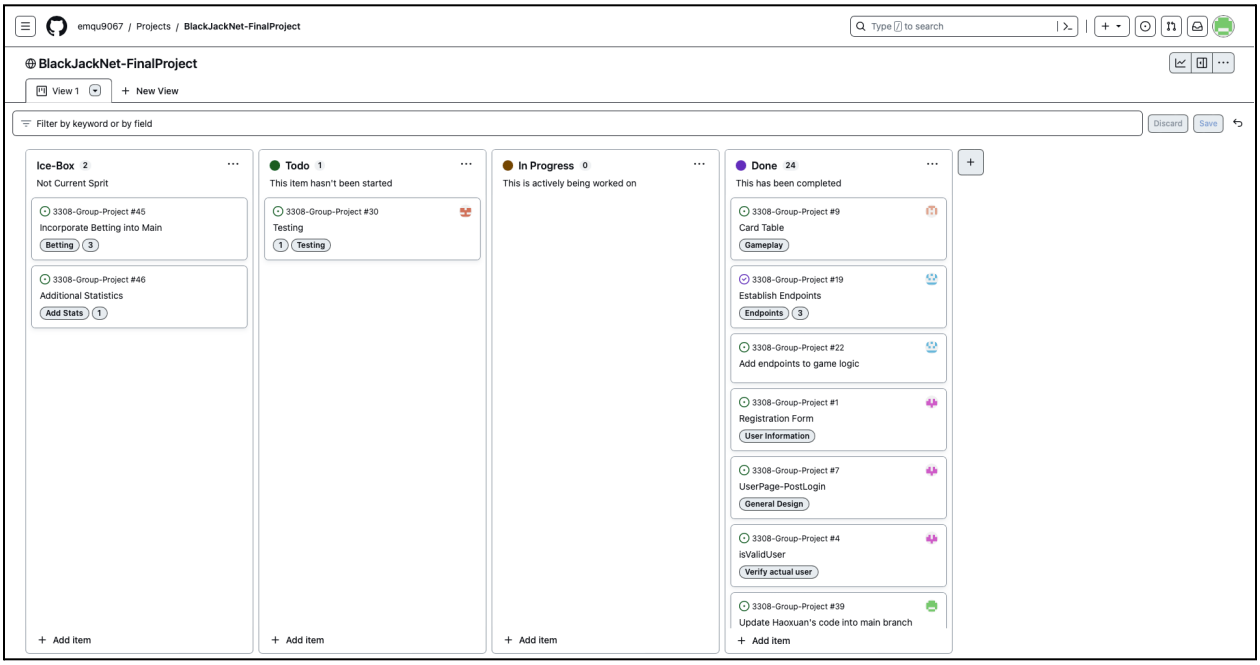
Project: Blackjack.Net

Members		
Name	*Email*	*GitHub Username*
Aiden Macdonald	aima2745@colorado.edu	aima2745
Alexis Cooper	peco6819@colorado.edu	AlexisCooper1
Emanuel Quintana	emqu9067@colorado.edu	emqu9067
Haoxuan Ding	hadi8111@colorado.edu	A-m-e-n-g
Zach Jordan	zajo2067@colorado.edu	ZachJordan

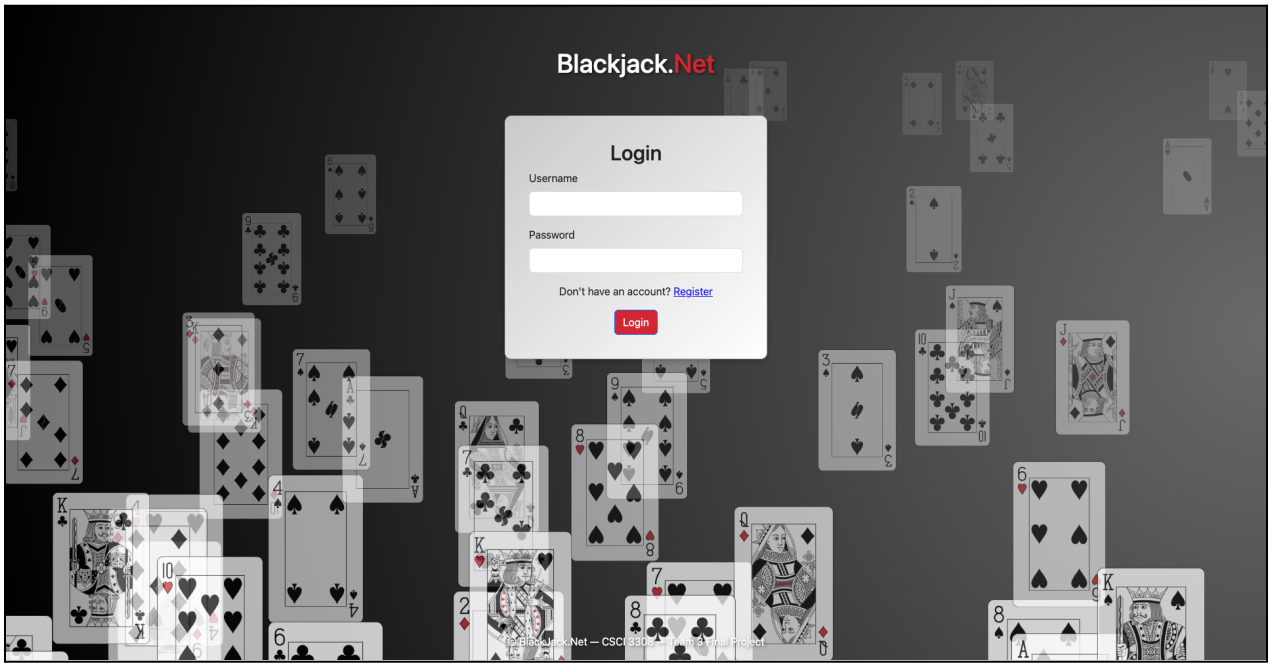
Project Description:

Our blackjack website offers users an immersive and exciting online gaming experience where they can play the classic casino game of blackjack against an automated dealer. The application combines sleek design, smooth gameplay, and realistic graphics to create an engaging platform for both novice and experienced players. Our application follows all essential mechanics of a traditional blackjack game. Users receive their initial cards and make strategic decisions by hitting and standing, to maximize their chances of winning. The automated dealer follows standard blackjack rules—adding an element of challenge and unpredictability to each hand. Track you statistics over time to improve your win percentage, and ensure you're not being dealt bad hands by tracking your most common cards. By bringing the thrill of blackjack to the virtual world, our application offers users the convenience of playing their favorite game anytime and anywhere. Whether it's to unwind during a break, or sharpen their blackjack skills, our application provides hours of entertainment and a valuable opportunity for players to test their strategies and luck.

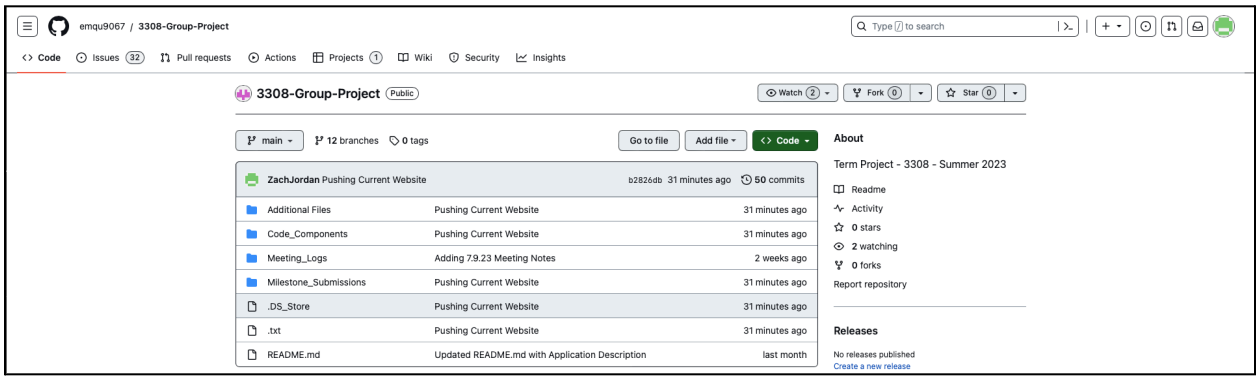
GitHub Project Board: [Link]



Video: [Link]



Version Control System: [Link]

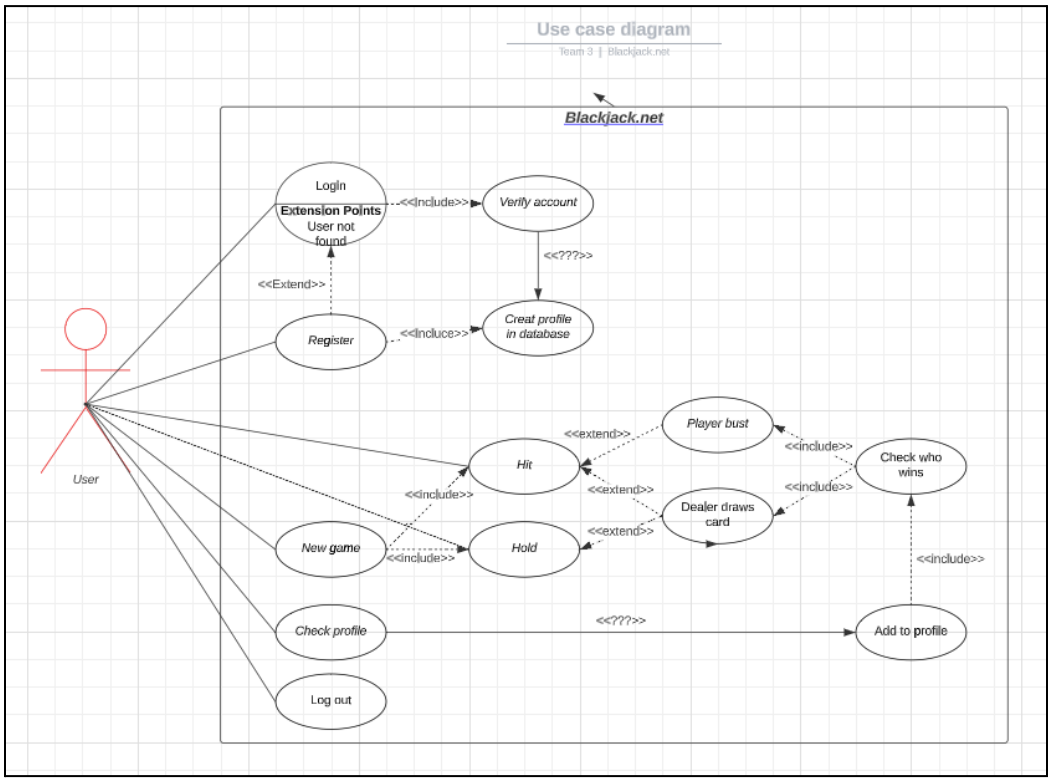


Contributions:

Member	Personal Contributions
Aiden Macdonald	Initial API routes for: - Login - Logout Contributed to: - Testing - Initial use-case diagram - Reviewing major pull requests - Application description
Alexis Cooper	Most API routes and database interaction for ● Registration ● Betting ● Table functions (hand database, card database) ● Sessions
Emanuel Quintana	Betting Implementation: Not Deployed * Partial Page Implementations to represent three stages in a blackjack game ● To starts the game ● To place a bet - updates the database in real time ● To play the game and decide to keep playing or not *Navigated using API calls embedded in buttons *If you would like to check this out, clone the betting branch in our GitHub and run docker compose up from the src folder
Haoxuan Ding	The initial gameplay.js framework A few refinements ● Auto Bust(lose game) ● Play again function User interface creation (not deployed)

Zach Jordan	- Reworked member's code from branches into standardized format - All UI Work (all CSS, login page animation, and related Javascript) - Designed and Implemented Gameplay Mechanics - Refined Login/Logout/Registration Pages & Respective APIs - Incorporated Password Hashing - Designed Database Schema - Created Database and Server - Deployed Docker Image to Azure - All of Presentation Slides (Excluding the tool logos) - Entirety of Profile Page, along with the respective endpoints - Found/Implemented playing card SVGs - Organized meetings, tracked attendance, and took notes (notes for first 2 weeks only) - Updated GitHub with revised README

Use Case Diagram:



Test Cases:

Task 1: Create an account

Description: In this task, you will need to create a new account on our platform. To do this, you will be required to provide some basic information such as your name, email address, and a secure password. Once you have entered the necessary details, click on the

"Register" button to complete the process. Please take note of the username and password you used as you will need them for future logins.

Task 2: Login to Account
Description: For this task, you will need to log in to your existing account. Click on the "Login" button and enter the email address and password that you used during the account creation. If the information provided is correct, you will be successfully logged in and directed to the main table page.

Task 3: Play 3 games of Blackjack
Description: In this task, you will play three games of Blackjack, a popular card game. Once you are logged in, locate the "Play" section or tab in your dashboard. Click on it to access the Blackjack game interface. Follow the instructions to make decisions during the game. After completing three games, you can proceed to the next task.

Task 4: Check your playing statistics
Description: For this task, you will need to check your playing statistics for Blackjack. Go to the Profile tab, and look for a section that displays your gaming or playing history. It might be labeled as "Statistics," "Gaming History," or something similar. Click on it to access your Blackjack statistics, including the number of games played, win/loss ratio, and any other relevant information.

Test Results/Observations:

1: No issues. User created an account with a real email and was redirected to the login page. Consistent with use case. User then tried to do SQL injection which was kinda funny but it didn't work. No changes recommended.

2: User was redirected to the login page and assumed that registration had failed - recommended change: registration page should automatically log in a player and redirect them to the table.

3: User played several games of blackjack successfully. The dealer immediately drew 21 on the user's first game, which was a massive skill issue. "Play Again" button worked as intended, and session continually updated. Consistent with use case, no changes recommended.

4: User checked statistics and found them to be accurate. Enjoyed the UI. Consistent with use case, no changes recommended.

Azure Deployment: [Link]