

Project Overview

Group Members: Junhui Zhong, Yuvraj Gupta, Daniel Mitra, Kataliya Sungkamee

Game: Texas hold 'em (Poker game)

Description: Our project features one of most popular poker game Texa hold 'em, allowing users to create a room to play the game at a minimum 2 players and up to maximum 10 players.

Features:

Core Features (Required)

- **User Authentication** – Secure sign-up and login system using JWT.
- **Database Integration** – PostgreSQL database with Prisma ORM for managing users, game data, and scores.
- **Express.js Backend API** – RESTful routes built with Node.js and Express.js to handle all game logic and data flow.
- **Frontend** – Built with React + TypeScript + TailwindCSS for clean, dynamic UI.
- **Render Deployment** – Full-stack deployment on Render for a live, production-ready website.

Features:

Game-Specific / Application Features

- **Global Lobby** – Displays active game rooms and lets users create or join matches.
- **Real-Time Gameplay** – Live interactions using Socket.io for multiplayer sync and game updates.
- **Chip System** – Virtual chips for betting, raising, and tracking total winnings.
- **Dealer Logic** – Automatic card dealing, turn rotation, and pot calculation.
- **Game Room Chat** – In-room text chat to communicate with other players.
- **Game Results Screen** – Shows hand rankings, pot winners, and updated balances after each round.

Features:

Optional Enhancements

- **Leaderboard** – Displays top players by total chips or win streak.
- **Game Rules** – We need decide the winner based on their cards.
- **Private Tables** – Create password-protected poker rooms for friends.
- **Responsive Frontend** – mobile-friendly UI.
- **Dark/Light Mode Toggle** – Smooth theme switching using Tailwind.
- **Animation Effects** – Framer Motion for smooth transitions and gameplay feedback.
- **Error Handling UI** – Friendly alert messages for invalid inputs or connection issues.

Technologies

Backend Stack

- **Render** — Deployment platform for hosting the app
- **Node.js + Express.js** — Backend server and routing
- **PostgreSQL** — Relational database for storing user and game data

Frontend Stack

- **React (with Vite)** — Client-side framework
- **TypeScript** — Adds static typing for safer and cleaner code
- **TailwindCSS** — Utility-first styling for fast and responsive UI design

Supporting Tools

- **Prisma ORM** — Database modeling and query management
- **Axios / Fetch API** — For communicating between frontend and backend
- **Socket.io** — Enables live updates and player interaction
- **JWT / Passport.js** — secure user login and authentication

Texas hold 'em

[Sign in](#) / [Sign up](#)

POKER



Team Z-j

Welcome!

Already a member? [Sign in](#)



By creating an account, I agree to the [Terms of Service](#) and [Privacy Policy](#).

Sign-up

Ready to Play?

Not a member? [Sign up](#)

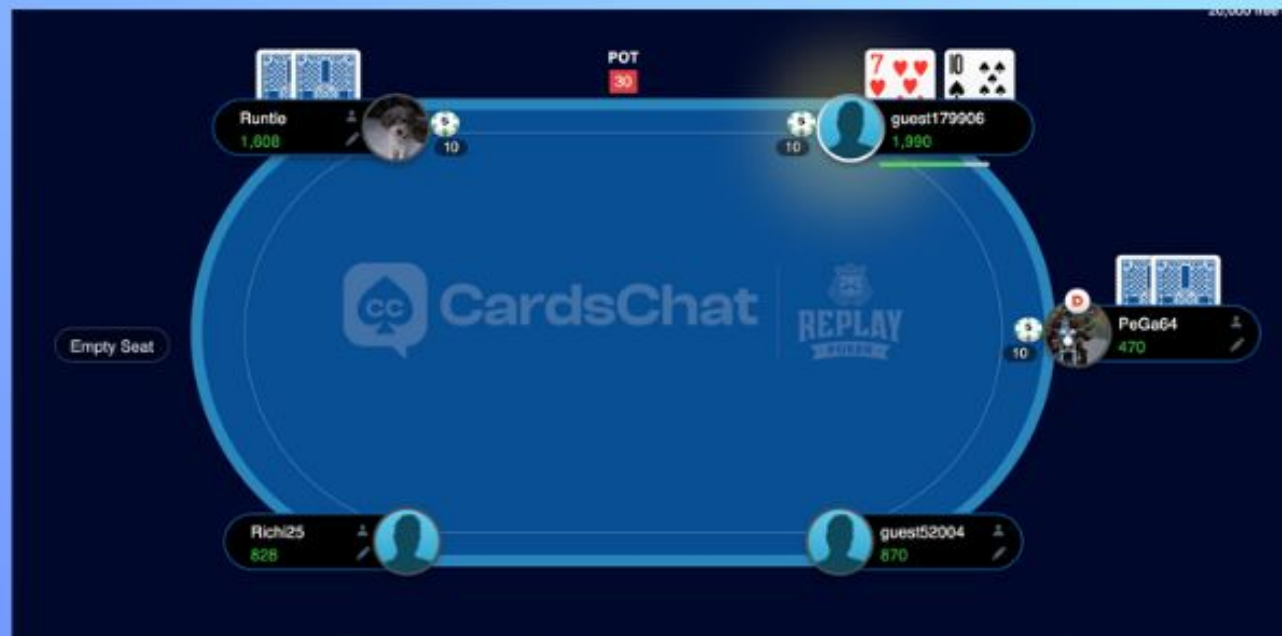
☐ Remember me

Log in

Signed-in Landing



Game Room



Game Results

#1. Username T0kens

#1. Username T0kens

#3. Username T0kens

#4. Username T0kens

Play Again

Return to Lobby

Wireframes

TODO: draw a picture of major screens,

- Show key UI elements (buttons, input fields, labels, etc.)
- Indicate **user interactions** (e.g., “click → navigates to Game Room”)
- Include short captions or arrows showing behavior

Required Screens

1. Signed-out Landing Page
2. Sign Up
3. Log In
4. Signed-in Landing (global lobby)
5. Game Room
6. Any additional screens your game requires (e.g., results screen, settings)