

GameHub

Product Owner: Benito Reyes

Scrum Master: BranDon Brown

Developers: Benito Reyes, BranDon Brown, Mekhi Green, Ramsay Burls

Sprint Goal

Enhance gameplay and user experience with AI single-player and suggestions, mobile and Safari improvements, retro theme, light/dark mode, profile & settings pages, and easier room joining.

Stories Delivered

As a user, I want to play against AI in Single Player so that I can have fun alone without friends.

→ The AI in Single Player has different levels of difficulty for a variety of

As a user, I want to join a game without typing in a code so I can join friends easily.

→ Players can join open matches directly through lobby selections with or without entering room codes.

As a user, I want to see a game menu for Drop 4 so that I can choose between game modes.

→ A main menu allows users to view and select available Drop 4 game modes before starting a match.

As a user, I want to navigate from the GameHub menu to Drop 4 so that I can easily access and start the game.

→ Selecting Drop 4 from the GameHub menu launches the game while preserving session data.

Updated Design Artifacts

Sprint 2

3. Players take turns dropping pieces on the board.
4. Moves are broadcast to the opponent via Socket.IO.
5. Win condition is checked locally and announced.

Chat Flow

1. Backend generates StreamChat token using `socket.id` as `userId`.
2. Token and userId are sent to frontend via `chat-auth` event.
3. Frontend connects to StreamChat using `connectUser()`.
4. A `gaming` channel is created or joined.
5. Players can send and receive messages in real-time.

Security & Environment

- `.env` file stores API secrets (never committed)
- StreamChat token is generated server-side only
- Frontend receives only the token and userId through requesting get from backend
- passwords are encrypted

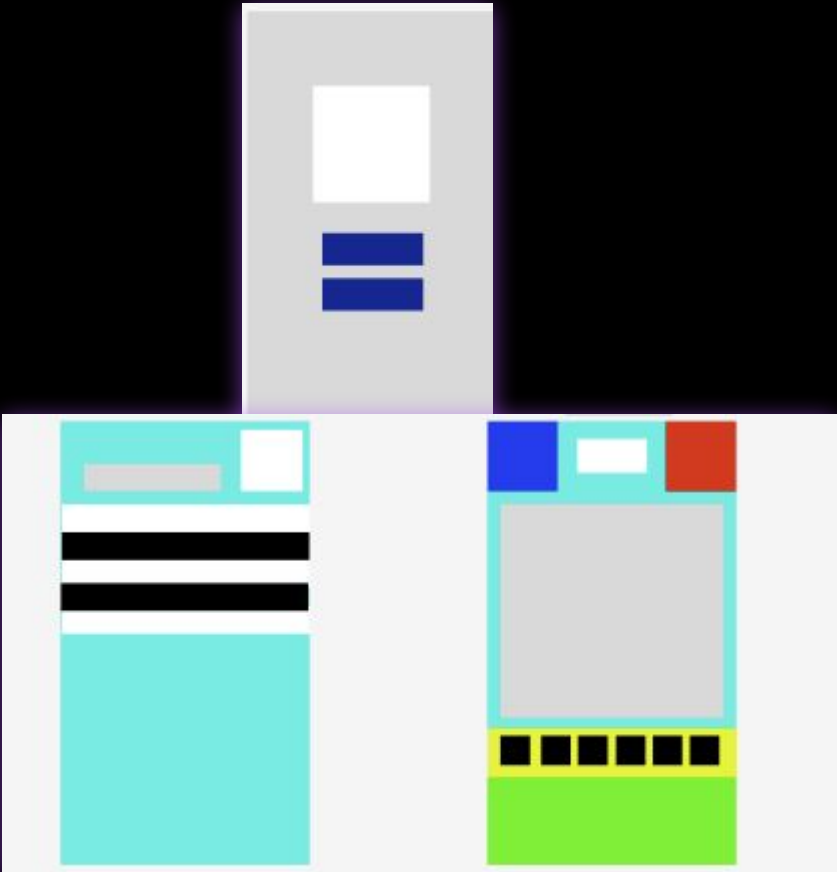
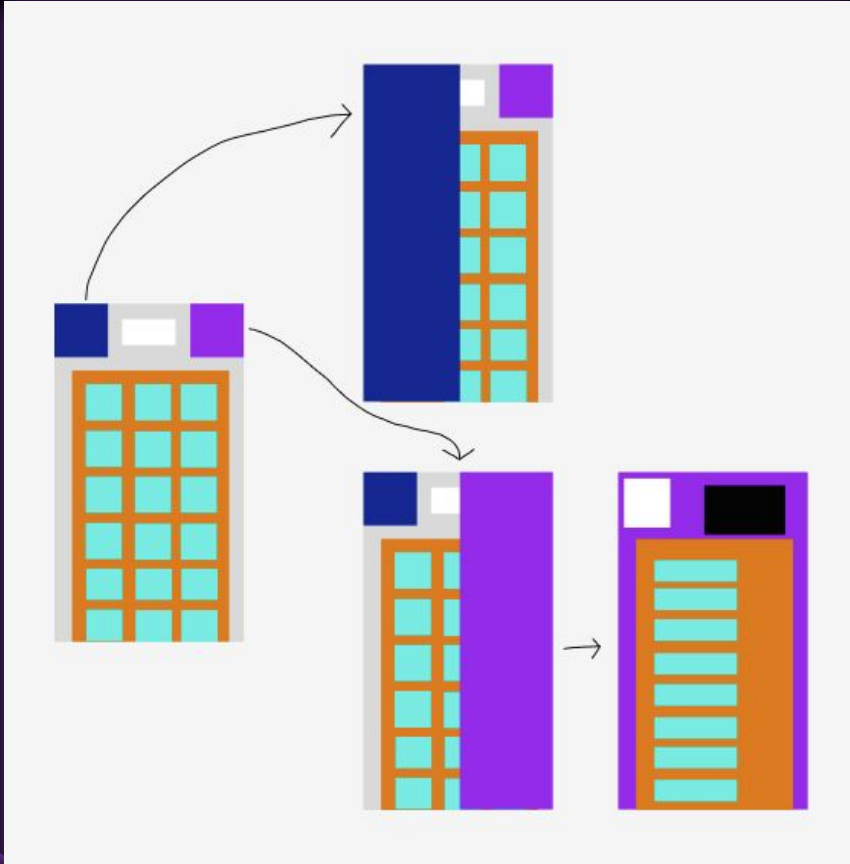
Dependencies

- `express` - initial HTTP server
- `socket.io` - real-time game communication
- `stream-chat` - chat SDK (client + server)
- `dotenv` - environment variable management
- `Node` - another server for browser testing
- `bcrypt` - used to encrypt password
- `cookie` - used to set userId and token cookies
- `neon` - the postgresql database being used
- `uuid` used to generate random userId's and roomId's
- `vite` - used to package streamchat bundle for browser/html usage
- `prisma` - used to query the database more easily

Sprint 3

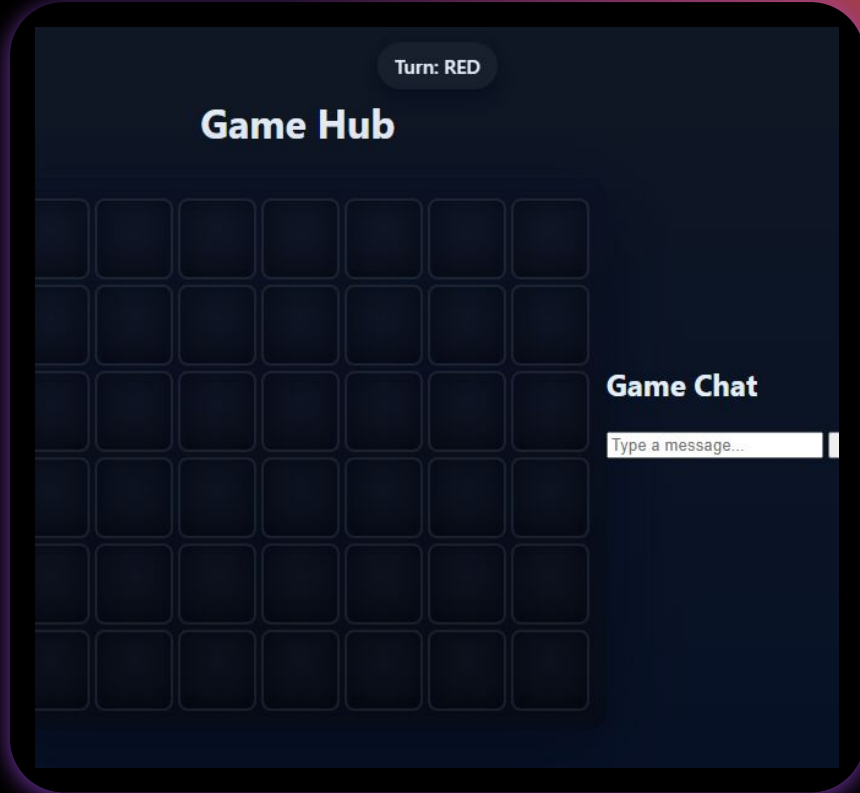
Architecturally nothing changed but there has been database clean up in its place (deleting rooms when the last users leaves it, etc)

Figma

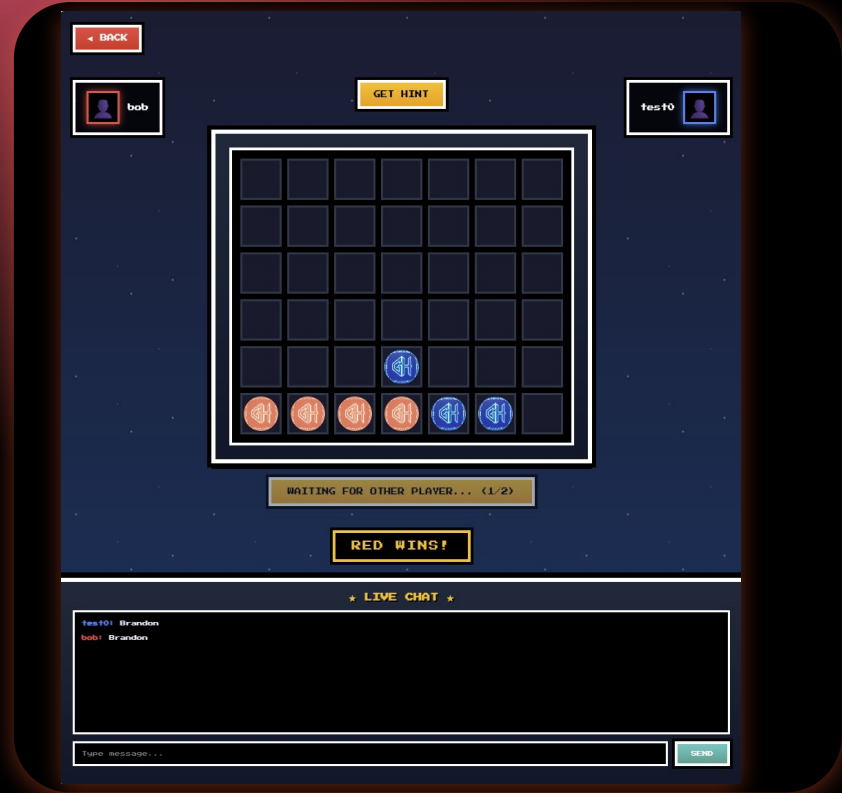


Updated Design Artifacts

Sprint 2



Sprint 3



Demo Link

[https://gamehub-ikpr.
onrender.com](https://gamehub-ikpr.onrender.com)



THANK YOU