Brodie Baxter
Rocco Zeno
Jake Naholnik

List of Frameworks/ Architecture

- **Socket.io** - We need to avoid using HTTP send/response for messages as this method would not be able to handle multiple players using the same server to play multiple games. Therefore we need a way to handle bidirectional and even-obased communication between our servers and users through web sockets. Socket.io will provide the services needed to achieve this goal with a high amount of scalability. The socket.io library consists of two parts, which are: a client-side library that runs in the browser, and a server-side library for Node.js.
- **ReactJS** -This library will be used to build our interface for our application. ReactJS is a flexible framework that will allow us to build our own components that are managed in their own state so that we can design our application closer to the ideas that we have in mind. React will also efficiently update and render the right components according to how our data changes in the game. Therefore, React will be how we interface ourbackend to our frontend.
- **AWS Cognito** - Cognito will be used to track everything that is connected to the player's account. This includes the log in system and storing everything attached to the players account such as: images, player status, and any other objects. AWS Cognito will handle the authentication and authorization system. Therefore Cognito will also cover security concerns such as proper encoding and encryption methods to securely store usernames and password information.
- **AWS CloudFront** - This service will be the network that we use to distribute our application. We will use an EC2 instance for our Web Application server. An EC2 instance is a scalable virtual machine that will provide all the CPU, storage, and memory that is needed for our application. The server will run on linux.
- **Terraform** - This is an open source infrastructure as code language that we will be using to build our VPC to create and modify AWS resources as needed. Using Terraform to manage our AWS infrastructure means we can create small changes without worrying about breaking the network due to its ability to save backups of prior commits.

Robust Diagram 1

Use Case: Create Account

BASIC COURSE:

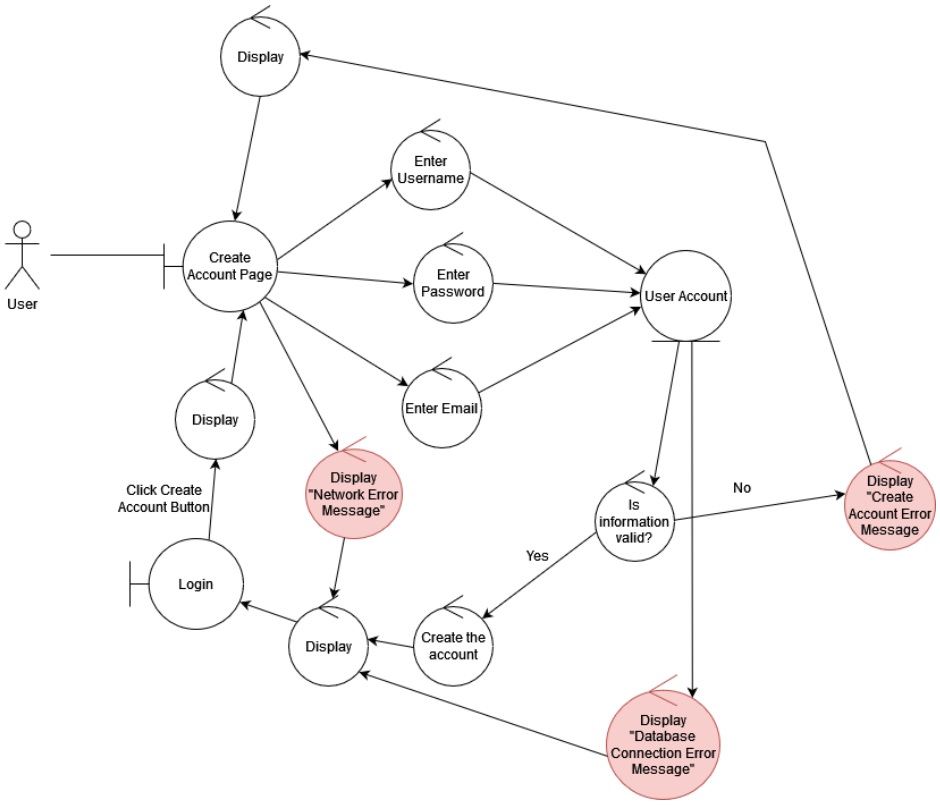
On the Login Page, the User clicks the Create Account Button and the system displays the Create Account Page. The user enters their username, password, email, and phone number into the fields and clicks the Create Account Button. The system checks this information in the database to see if it is valid. The system then displays the Home Page.

ALTERNATE COURSES:

Account existing error: The system shows a Create Account Error Message displaying the username, password, email, and/or phone number already exist for an account.

Network error: The system shows a Network Error Message and redirects the user to the Login Page.Home

Database connection error: The system shows a Database Connection Error Message and redirects the user to the Login Page.



Robust Diagram 2

Use Case: Create Private Lobby

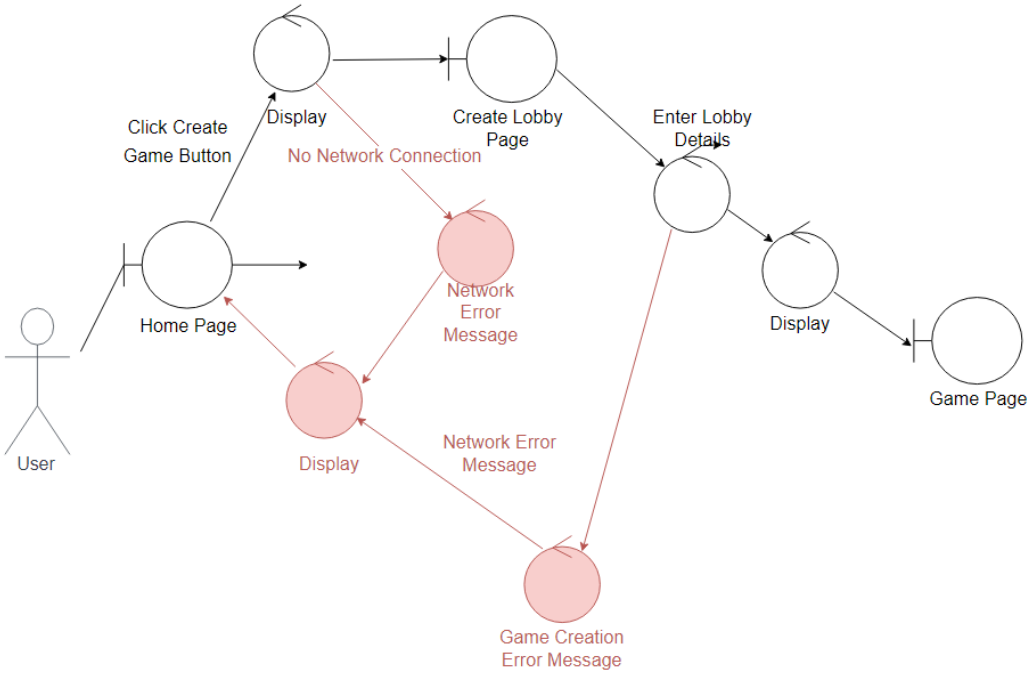
BASIC COURSE:

On the Home Page, the user clicks the Create Game Button, then the system displays the Create Lobby Page. The user enters the information for the game they are trying to create. The system then takes the user to the Game Page where the game is played.

ALTERNATE COURSES:

Network error: The system shows a Network Error Message and redirects the user to the Home Page.

Game Creation error: The system shows a Game Creation Error Message and redirects the user to the Home Page.



Sequence Diagram 1

Use Case: Create Account

BASIC COURSE:

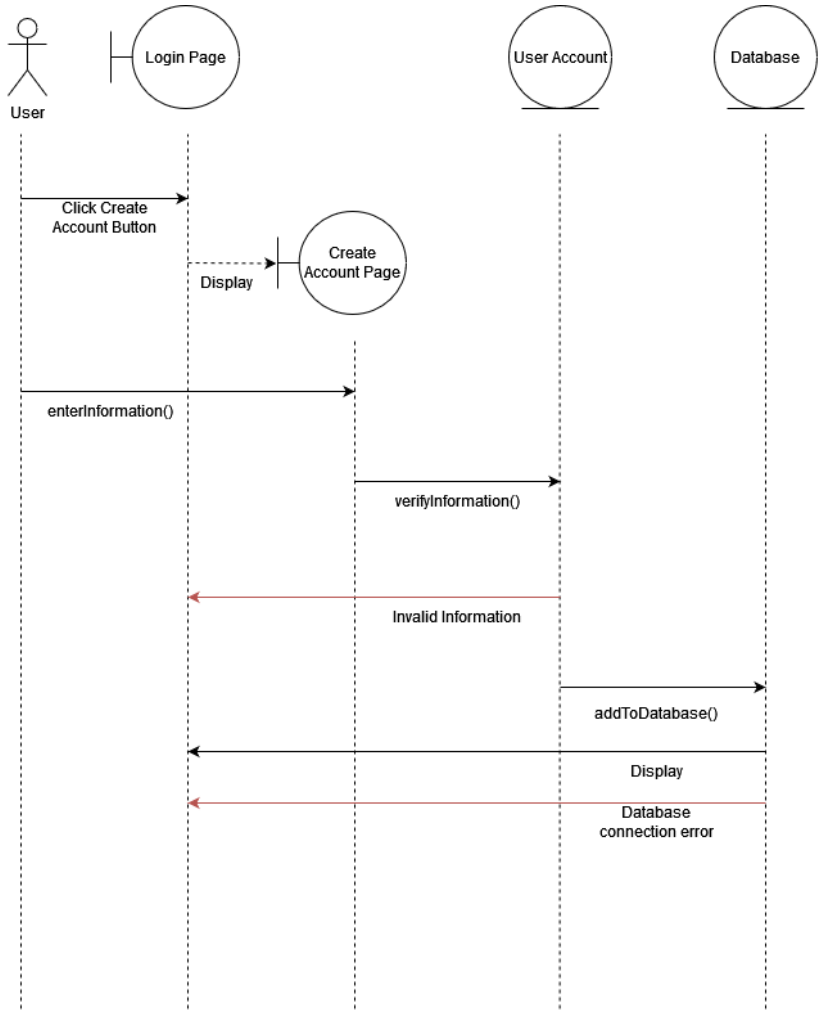
On the Login Page, the User clicks the Create Account Button and the system displays the Create Account Page. The user enters their username, password, email, and phone number into the fields and clicks the Create Account Button. The system checks this information in the database to see if it is valid. The system then displays the Home Page.

ALTERNATE COURSES:

Account existing error: The system shows a Create Account Error Message displaying the username, password, email, and/or phone number already exist for an account.

Network error: The system shows a Network Error Message and redirects the user to the Login Page.Home

Database connection error: The system shows a Database Connection Error Message and redirects the user to the Login Page.



Sequence Diagram 2

Use Case: Create Private Lobby

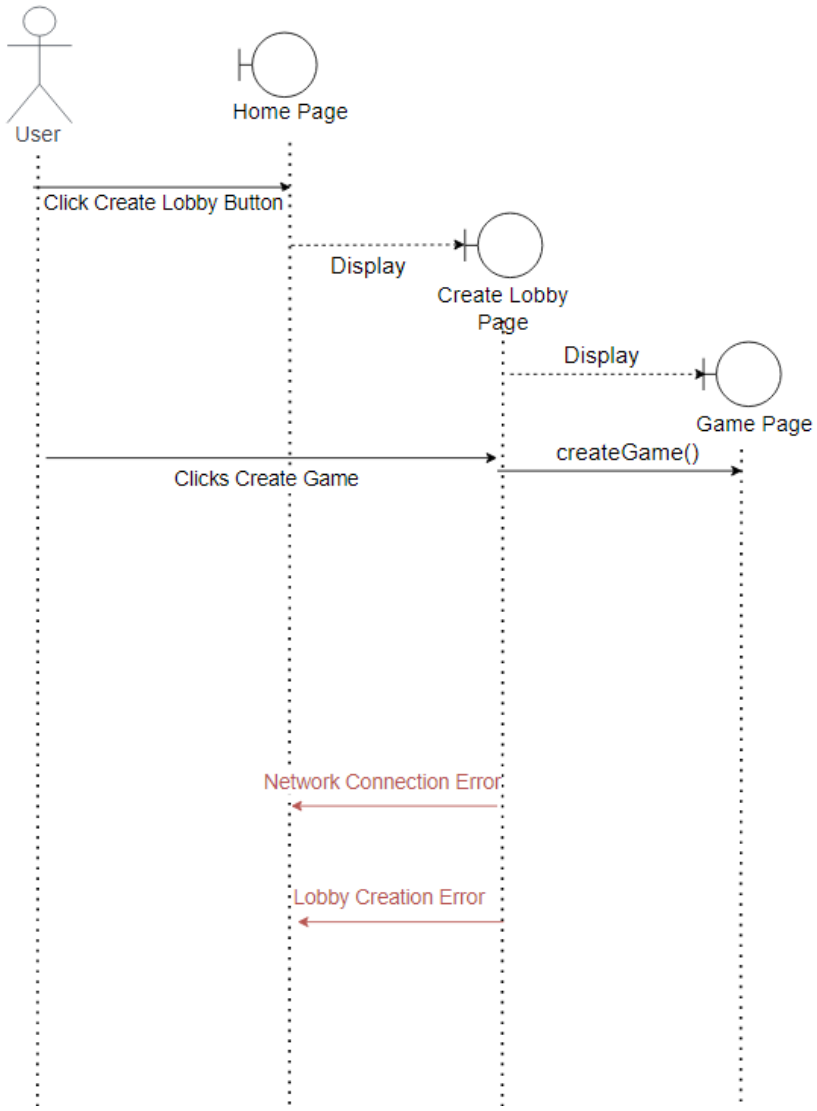
BASIC COURSE:

On the Home Page, the user clicks the Create Game Button, then the system displays the Create Lobby Page. The user enters the information for the game they are trying to create. The system then takes the user to the Game Page where the game is played.

ALTERNATE COURSES:

Network error: The system shows a Network Error Message and redirects the user to the Home Page.

Game Creation error: The system shows a Game Creation Error Message and redirects the user to the Home Page.



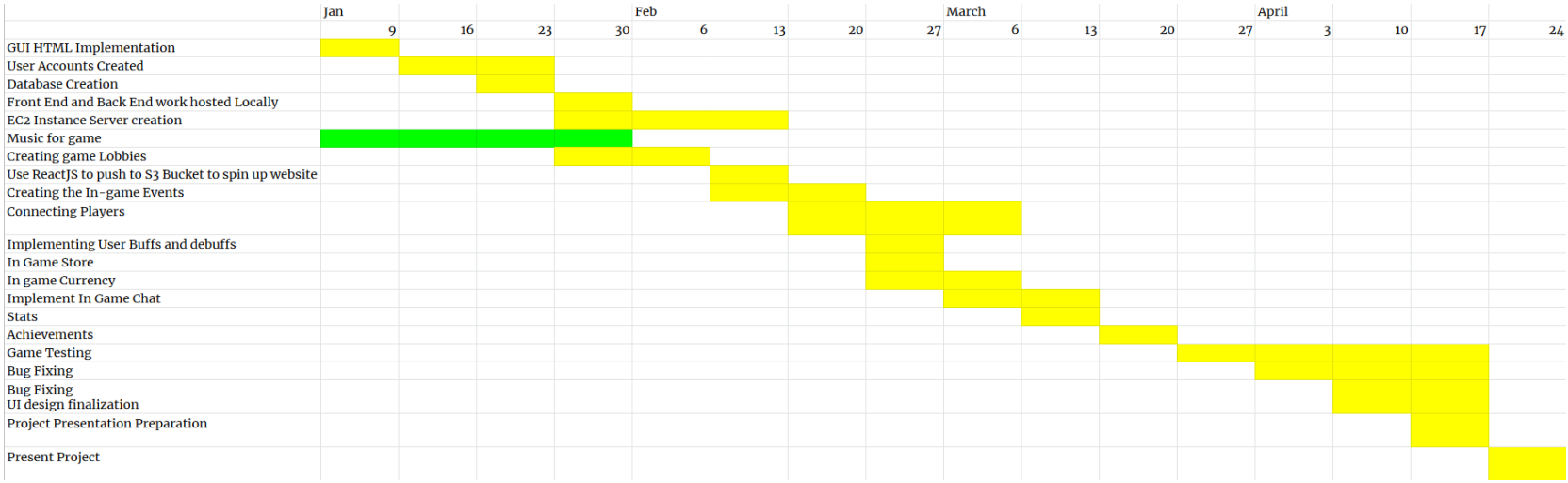
Weekly Software Process Schedule

Month	Day	Development Plan	Test and Demonstration Plan
Jan	9	GUI HTML Implementation - Log in page, Sign in Page, Home Page, Game Page created	- Users can click buttons to switch screens and browse them
	16	User Accounts Created - AWS Cognitio will be implemented for user accounts Database Creation - DB will be created to store user accounts	- Users will be able to create their account and sign up
	23	Front End and Back End work hosted Locally - Use ReactJS to create locally ran app	- Users will be able to connect to the app locally
	30	EC2 Instance Server creation - EC2 configuration for AWS Music for game - Game will have music during home screen and login screen	- Show EC2 Configuration - Users can listen to the music in the different screens
Feb	6	Creating game Lobbies - Joining and creating other users game lobbies	- Users will be able to create their own lobbies or join others lobbies
	13	Use ReactJS to push to S3 Bucket to spin up website	- Users can connect to the website now instead of

		- Instead of locally ran, app will be hosted on a web server	having to connect locally
	20	Creating the In-game Events - Adds the events users can come across during the game Connecting Players - Creating functionality to joining and creating a game	- Users can view the events in the game screen - Users can now create and join a game instead of locally
	27	Implementing User Buffs and debuffs - In game events Causing users to get buffs and debuffs	- Users can view buffs and debuffs on their player during gameplay
March	6	In Game Store - Multiple stores throughout the game for users to buy items In game Currency - Our Crypto currency esque system for users to buy items	- Users can view multiple stores with items to be purchased during gameplay
	13	Implement In Game Chat - Users chat for game events to be logged	- Users can chat with one another - Users can read game logs
	20	Stats - Game Stats like win total and death total Achievements - Achievements as in win a game or lose a number of games	- Users can view their stats on the home page - Users can view their achievements on the home page
	27	Game Testing - Having users test our	- Users can test and play the whole game

		game to find bugs	
April	3	Bug Fixing - If users find bugs during testing, we will fix them	- Users can report bugs found from gameplay to be fixed
	10	Bug Fixing UI design finalization	- Users can report bug found from gameplay to be fixed and suggest UI improvements
	17	Project Presentation Preparation	- Users can play the full game
	24	Present Project	- Present project

Gantt Chart


Jan
Feb
March
April
9
16
23
30
6
13
20
27
6
13
20
27
3
10
17
24
GUI HTML Implementation
User Accounts Created
Database Creation
Front End and Back End work hosted Locally
EC2 Instance Server creation
Music for game
Creating game Lobbies
Use ReactJS to push to S3 Bucket to spin up website
Creating the In-game Events
Connecting Players
Implementing User Buffs and debuffs
In Game Store
In game Currency
Implement In Game Chat
Stats
Achievements
Game Testing
Bug Fixing
Bug Fixing
UI design finalization
Project Presentation Preparation
Present Project