Project Milestone 4

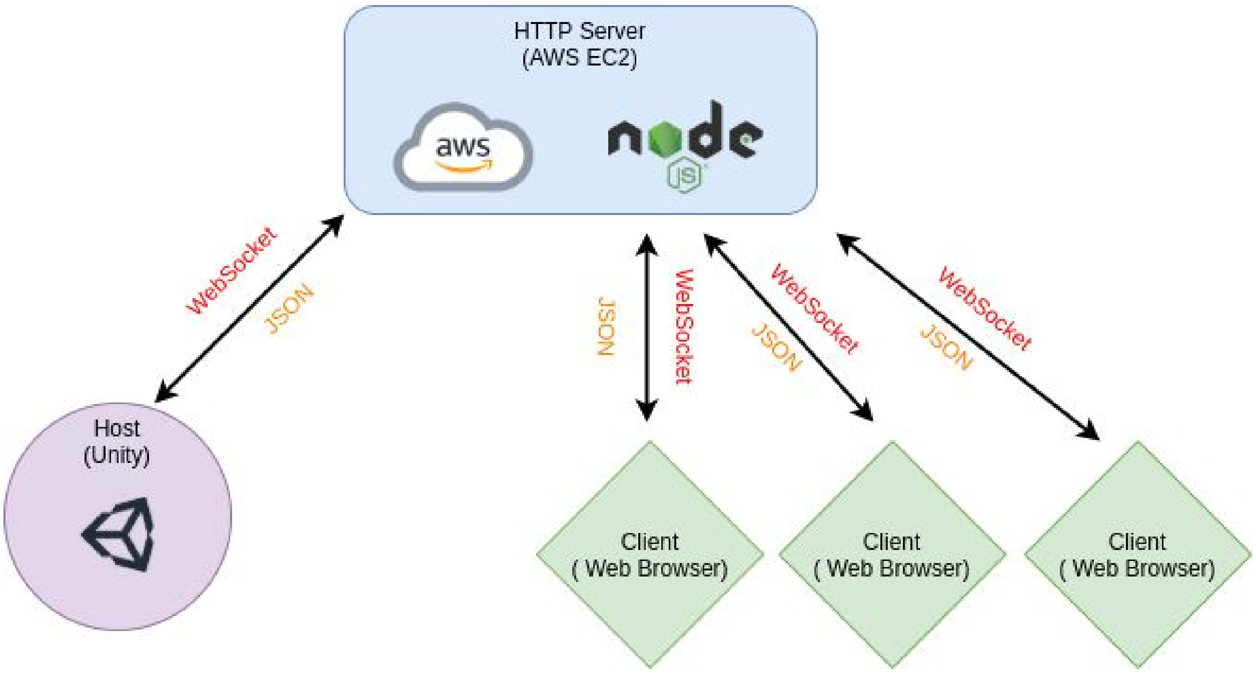
Group Name and Number: 203-5 Trouble Twist

Group Members:
Jake Tracy - jakewoodtracy
James Bohn - jbohn3353
Alexey Yermakov - yyexela
Conor Simmons - conorsim
James Ryan - jary7635
Blake Peery - BlakeP32

Revised Project Features List:
(Highest to lowest priority)

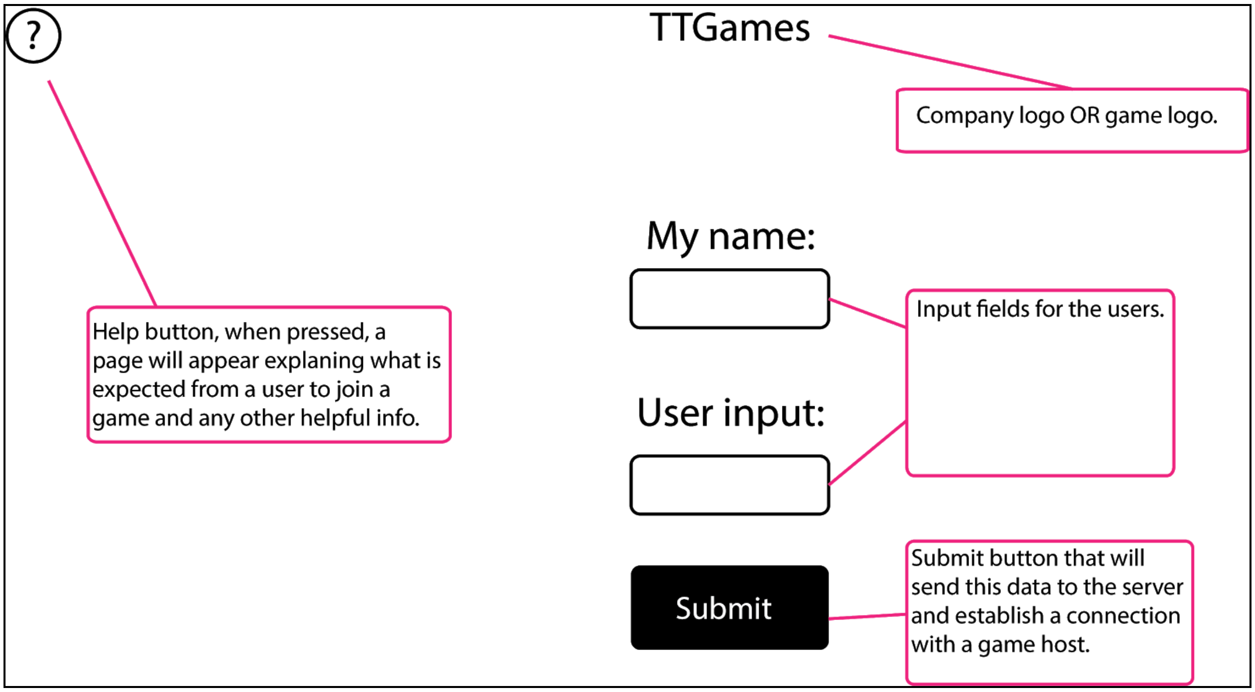
1. Game executable and application
   - The main point of contact for a user to create a game session. The Unity executable will be used to create a game lobby with a unique ID which clients can use to join the lobby on their browsers.
2. AWS server to manage traffic between players and hosts
   - Initiate websocket connections with hosts and clients. Subsequent requests between clients and hosts within their respective lobbies will route through the server.
3. Connect your phone to the host game via a web browser
   - The main point of contact for a user to join an existing game session. The users will have to supply a game ID and a username which isn't yet in the lobby they are joining.
4. Choose a nickname when joining a game
   - Users will have the option to customize their identity by choosing a nickname which isn't yet present in the lobby they are joining.
5. Question and prompt style game with multiple choice answers on famous quotes
   - The core MVP game mechanic, players will be shown famous quotes with multiple choice options for for who said it, with the game showing who was correct at the end of every question

Architecture Design:



Front-End Design:

Wireframe 1: Website view upon load.



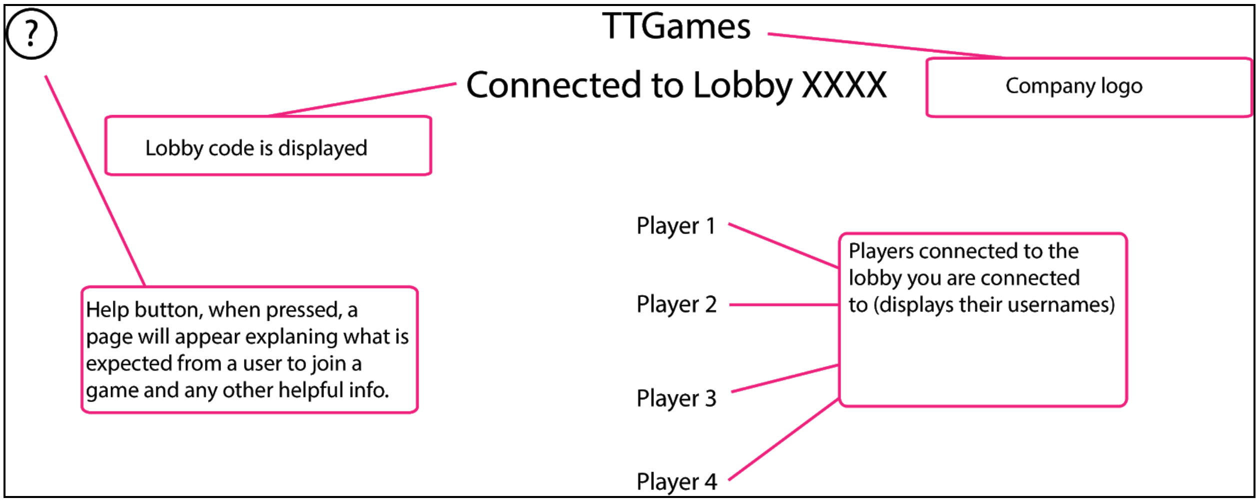
Company logo OR game logo.

Help button, when pressed, a page will appear explaning what is expected from a user to join a game and any other helpful info.

Input fields for the users.

Submit button that will send this data to the server and establish a connection with a game host.

Wireframe 2: Website view upon successful login.



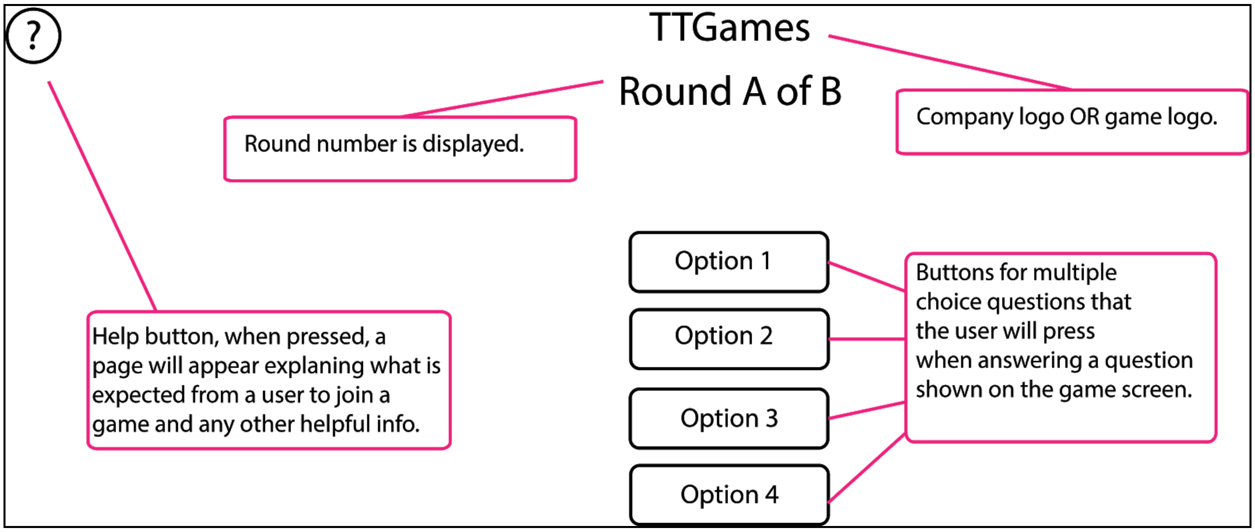
Company logo

Lobby code is displayed

Help button, when pressed, a page will appear explaning what is expected from a user to join a game and any other helpful info.

Players connected to the lobby you are connected to (displays their usernames)

Wireframe 1: Website view on game start.



Company logo OR game logo.

Round number is displayed.

Help button, when pressed, a page will appear explaning what is expected from a user to join a game and any other helpful info.

Buttons for multiple choice questions that the user will press when answering a question shown on the game screen.

Web Service Design:

- We are using the WebSocket API in our website to communicate with our AWS server. The data being passed to and from our server through the API is in the form of JSON objects, which contain information regarding joining a lobby, playing the game, and any error messages.

Database Design:

- We will not be using a database for this project.
- We will not have registration/log capabilities because our program is designed to be able to run by anyone. If you have access to the Unity executable and/or website where the

game client is hosted, you can play the game and send messages to/from the server. The server contains back-end message-checking code in order to ensure that if anyone sends a message that isn't formatted correctly, the server disconnects that user.

Individual Contributions:

James Bohn:

- Server architecture and message handling, web client interface with server communications.
- https://github.com/CSCI-3308-CU-Boulder/203_5_F20/commit/3360ec9fb67bebba83e44b5334ee9a2c80f1e1e1

Alexey Yermakov:

- HTML user interface (static files and some JS), initial server development (Creation of AWS account, EC2 instance, EBS storage, EC2 and EBS integration, server set-up), meeting management, initial websocket communication development between server and clients
- Link to latest Github Push: https://github.com/CSCI-3308-CU-Boulder/203_5_F20/commit/b1843c2c815d8ec44f3ef0d7487ce329e9d1789d

Jake Tracy:

- Initial Unity development, project documentation, wireframe layout of Unity application, initial development of logic mapping of Unity application
- Link to latest Github Push: https://github.com/CSCI-3308-CU-Boulder/203_5_F20/blob/master/Unity/Quotes%20and%20Answers%20Link.txt

James Ryan:

- Initial Unity development, project/milestone documentation, initial development of logic mapping of Unity application, development of multiple choice game format for questions in Unity and file input/output utilizing a document of quotes
- Link to latest Github Push: https://github.com/CSCI-3308-CU-Boulder/203_5_F20/tree/UnityDemo/Multiple_choice

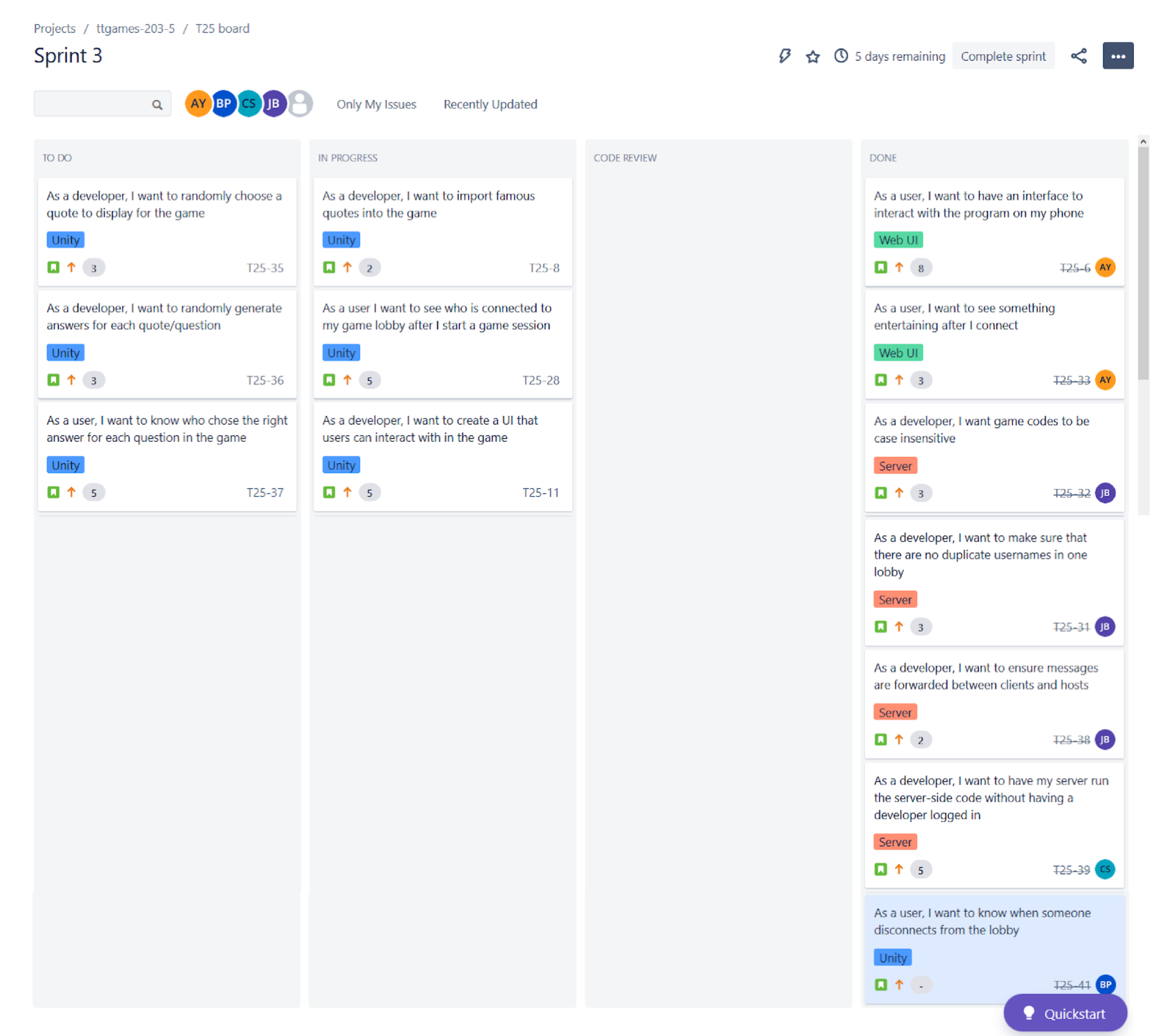
Conor Simmons:

- Initial server development, exploratory research and ideation to pivot to a different architecture (AWS), established C# websocket communication, set up the node.js server to run perpetually
- https://github.com/CSCI-3308-CU-Boulder/203_5_F20/commit/1a248f027b58230e63df2f2e6cdea89cb3b291d5

Blake Peery:

- Unity game host lobby, Unity-side communication, Unity UI, created a document with quotes for our unity game.
- https://github.com/CSCI-3308-CU-Boulder/203_5_F20/commit/901282f81bbc60972238d3767b560e27ea3cfe13

Link to Jira:
https://csci-3308-fa20-203-5.atlassian.net/secure/RapidBoard.jspa?rapidView=1&projectKey=T25&view=planning.nodetail&selectedIssue=T25-8&epics=visible&issueLimit=100&atlOrigin=eyJpIjoiZjRmN2I2OGZkYzI3NDM4NDhlMDg3YjQ5M2I1ZWNjNTEiLCJwIjoiaiJ9



Projects / ttgames-203-5 / T25 board

Sprint 3

5 days remaining · Complete sprint

AY BP CS JB · Only My Issues · Recently Updated

TO DO

As a developer, I want to randomly choose a quote to display for the game
Unity
↑ 3 · T25-35

As a developer, I want to randomly generate answers for each quote/question
Unity
↑ 3 · T25-36

As a user, I want to know who chose the right answer for each question in the game
Unity
↑ 5 · T25-37

IN PROGRESS

As a developer, I want to import famous quotes into the game
Unity
↑ 2 · T25-8

As a user I want to see who is connected to my game lobby after I start a game session
Unity
↑ 5 · T25-28

As a developer, I want to create a UI that users can interact with in the game
Unity
↑ 5 · T25-11

CODE REVIEW

DONE

As a user, I want to have an interface to interact with the program on my phone
Web UI
↑ 8 · ~~T25-6~~ · AY

As a user, I want to see something entertaining after I connect
Web UI
↑ 3 · ~~T25-33~~ · AY

As a developer, I want game codes to be case insensitive
Server
↑ 3 · ~~T25-32~~ · JB

As a developer, I want to make sure that there are no duplicate usernames in one lobby
Server
↑ 3 · ~~T25-31~~ · JB

As a developer, I want to ensure messages are forwarded between clients and hosts
Server
↑ 2 · ~~T25-38~~ · JB

As a developer, I want to have my server run the server-side code without having a developer logged in
Server
↑ 5 · ~~T25-39~~ · CS

As a user, I want to know when someone disconnects from the lobby
Unity
↑ - · ~~T25-41~~ · BP

Quickstart

Challenges:

- One challenge is determining whether the game itself is fun, since we have been spending all our time planning and developing this game and its format but have not actually tested it in length. As a risk mitigation plan, if we find that the game is not fun, we have allowed a bit of time to make minor changes to make the game more fun; however, we are relatively confident that this game will be fun and will be an enjoyable experience to play.
- Another challenge is finishing development of the game itself, since it is not yet completed at the time of this writing. Finishing the game design in Unity will complete our initial plan for the software, as it will tie in with all the other components of our software design. As a risk mitigation plan, if the game itself is not able to be finished in Unity, we at least have several other features completed and have some sort of software to show for our efforts. However, we are confident that we will overcome this issue since we are so close and the majority of the work has already been completed.
- A third challenge is user input test cases and edge cases, since it is important for us to account for all possible user behaviors and interactions with our software. As a risk mitigation plan, our software has its intended functionality and can properly run even if not all these test and edge cases are addressed. We will continue to test to identify and address each test case in turn, and we are confident that we will make more progress with regards to this challenge.