1. Team Number:
111-2

2. Team Name:
Entertainment 720

3. Team Members:
- Sol Atencio
- Aidan Baack
- Nicholas Lescanic
- Adam Spiers
- Jaykob Velasquez
- Yicheng Yi

4. Application Name:
UpRight

5. Application Description:

The application will be a web page where a user sets an alarm. When the alarm goes off, they will be required to play a Wheel of Fortune/Hangman type game in order to turn off the sound. When the phone is unlocked, relevant information such as the weather and tweets could appear on the screen. A user will be able to login to the website, and they will then be able to keep track of how quickly they turn off the alarm. They can also choose what they want to display after they win the game.

There will also be additional statistics such as how few incorrect guesses they have and how often they snooze the alarm. The hope is that the alarm will work well on a phone because this is where most people set alarms, but it will begin as a web page. This will ensure that it works on desktop initially, and it can then be modified to also work on mobile devices. This will also avoid having to separately code applications for Android and iOS.

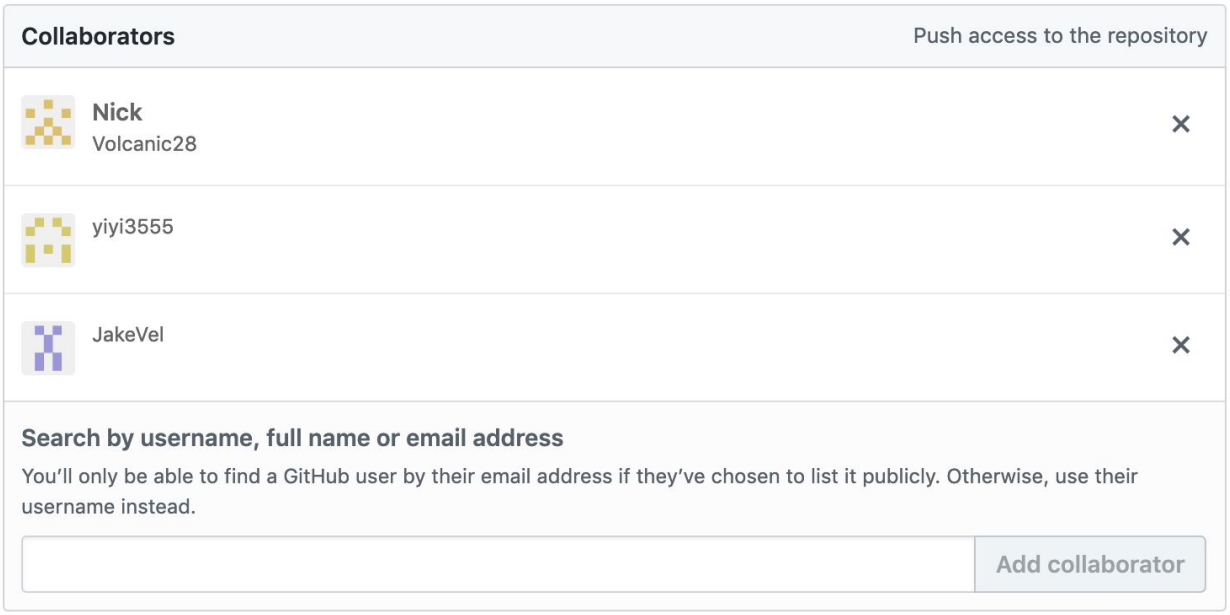
6. Vision Statement:

We seek to develop the necessary skills for creating a modern web application and help people boost their productivity by forcing them to wake up with an enjoyable and engaging game.

7. Version Control:

- Meeting Logs: https://github.com/aebaack/csci3308_meetings
- Milestones: https://github.com/aebaack/csci3308_milestones
- Project: https://github.com/aebaack/csci3308_project

Several group members were not in lab and have not responded to any group messages, so we are missing two collaborators from the repositories. All three of the repositories look like this:



8. Development Method:

We will follow an agile methodology of development because we are still learning a lot of the technology, so we want to allow for a great deal of flexibility during development. Agile is an iterative development methodology with a big focus on customer value. It is very flexible because plans are not heavily set in stone, allowing for experience during production to guide the process. Instead of waterfall, where the project specifics are planned at the beginning and cannot be modified later on, agile is open to change.

Our first development goal will be working on the database and the login/password for the users. From there, we will determine which components are most important for the alarm and the game. For our group, we will have a weekly scrum meeting during our usual meeting time. There will be no exact team leader, so everyone is free to share

their ideas for the project. We will consider each idea and discuss our plan as a group. We will use project management software where we can keep track of important changes and what has been completed. The most relevant issues will be at the top, and we will determine which issues are important at the beginning of every weekly scrum. We will use waffle.io for a dashboard for project management.

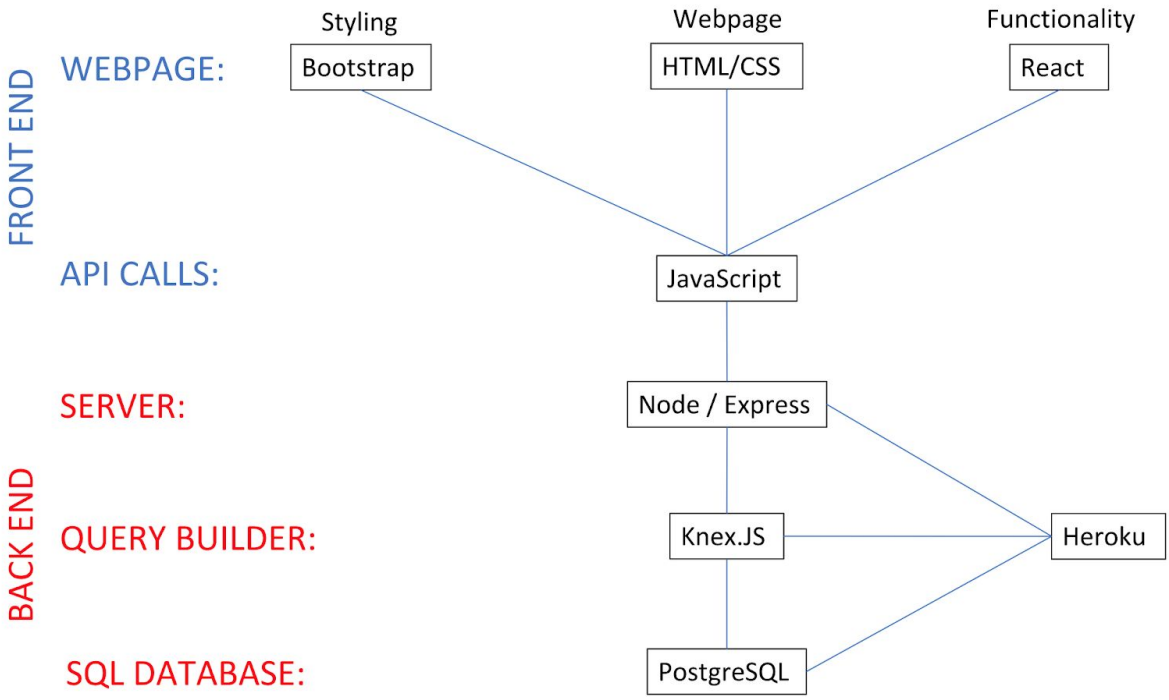
9. Communication Plan:

We currently have a GroupMe setup so that every member can communicate in one place. We also have a Google Drive for creating documents related to milestones and other deliverables. Each repository for the class is on GitHub, and every member is added as a collaborator. For project management, we will be using waffle.io, which tracks project features as issues on a GitHub repository. We all have access to the repositories, so we will all be able to view the project management dashboard.

10. Proposed Architecture Plan:

For the front end of our application, we will be using HTML and CSS for creating a website. We will use Bootstrap for styling the website. As for the functionality of the page, we are planning on using React, which means that we will be using JavaScript. Since we will gain JS experience through programming the front end, we are also planning on using it for the backend with Node.js.

The database will be managed with PostgreSQL, and queries will be made through Node.js using Knex.js, which is a query builder for JavaScript. The server will use Express. The server and database will be hosted on Heroku. In summary, our data will be saved in an SQL database using PostgreSQL, and the data can be retrieved by our node server using Knex.js. Our server will rely on Express in order to respond to any calls from the webpage that will be the front-end of our application. The server will be hosted in Heroku. This application will use React and JS for the functionality, and Bootstrap/HTML/CSS will be used for creating the actual webpage.



11. Meeting Plan:

We will be meeting every Thursday from 5-7 PM at the MATH library.