Teatime

December 10, 2019

By Jack O’Hare

COMP 424

Table of Contents

Participants

Teatime was a solo project and solely developed by Jack O'Hare. He completed the design, backend development, cloud services, testing, and any other components to this project.

Abstract

The idea behind teatime was to create a utility that assisted in the art of brewing tea. Brewing tea is a much more complicated process than many people think, and there are many variables that go into the process of making tea. These variables are different for most kinds of tea, and the goal with this project was to have this information in one place. This simplifies the process of making tea and allows you to do it as effectively as possible.

Narrative

The goals for this project were to implement the core principles shown in the abstract. Simplicity was vital in the design and function of this application. Tea is meant to be a relaxing event and it was important to keep it that way when developing this web app. The main goal of this project was to provide the core information required to understand and brew the tea, while keeping it unobtrusive and to not overwhelm the user. A timer function was an essential part of the project, as it is an essential part of making tea. A custom timer was also desired, as some people like their tea brewed at different strengths and like to make adjustments of their own. When doing this, the suggested time is still available to the user as a base to go off of. Water temperature is also a very important part of brewing tea, so that was a required part to add to the app. To cater to the widest user base possible, a Fahrenheit/ Centigrade conversion

tool was to be implemented. Finally, a general description is wanted so when waiting for their tea to brew, a user can read a little about what they are drinking.

Design Considerations

The main design for this project was to follow a “dashboard” layout. All the core functionality was to be kept on one page so it could all be accessed at once when making tea. Simplicity was the biggest concept in mind when creating the layout of the page. A grid layout was used to position the timer, tea selection box, description, and the rest of the controls. This allowed for a clean and uniform positioning of the various components into tiles. The color scheme was decided to be a combination of greys to keep the design simple. Other secondary colors were tried, but found to be distracting to the overall design, so they were mostly omitted.

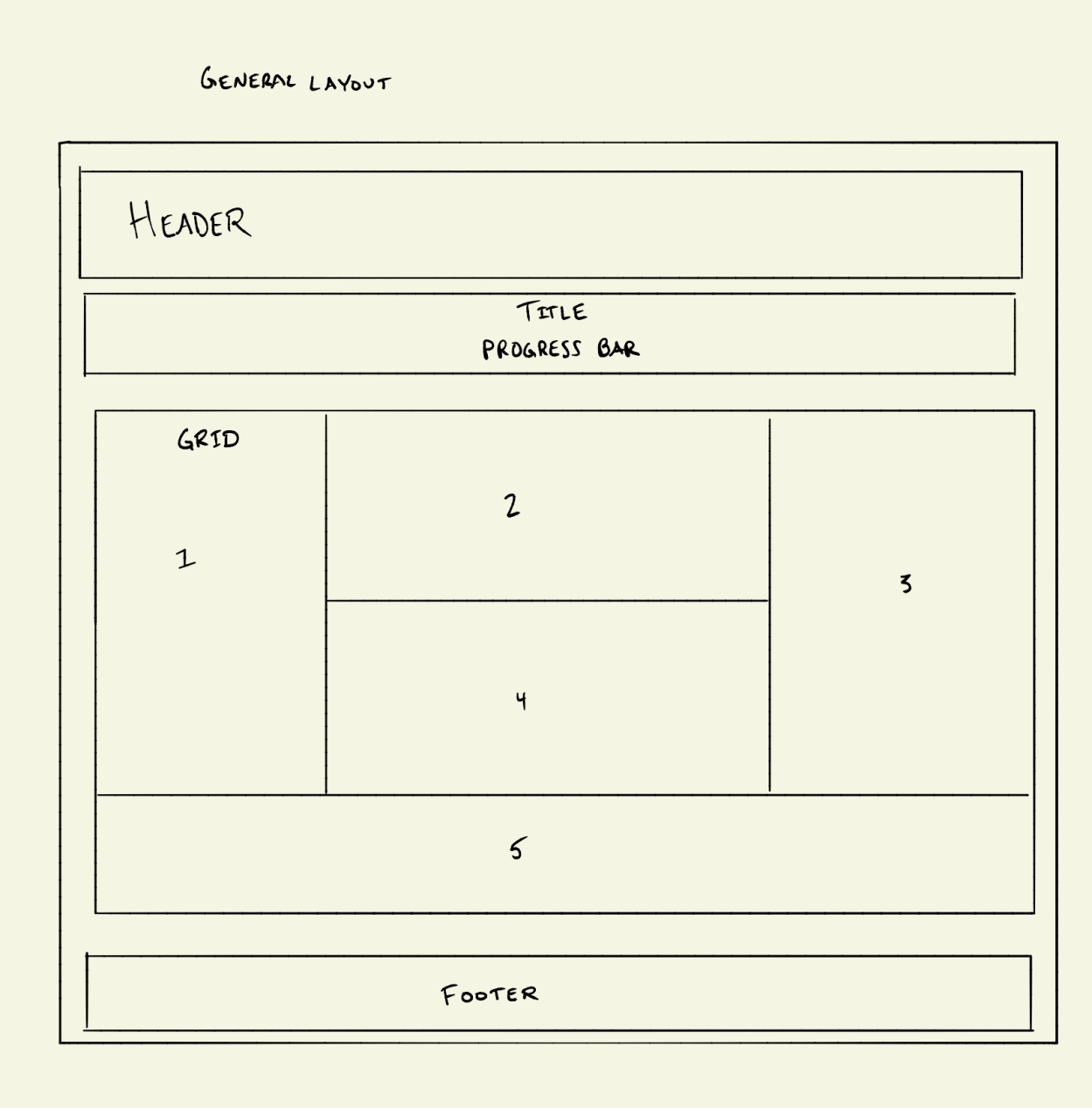
The timer function is essential to the applications function and uses the SetInterval function to countdown the timer.

Firebase real-time database was chosen for database services. Firebase’s simple structure is ideal for this app because of it’s read-only nature. There is need for one object, the Tea. The object’s attributes include teaName, teaDescription, steepTime, temp, and seconds. This is all returned as a json string which is scraped off the URL and converted to a JavaScript object.
For web hosting, Microsoft azure was originally used, but the app service was migrated to firebase last minute. The reason for this will be detailed in the *Restrictions, Limitations, and Constrains* section.

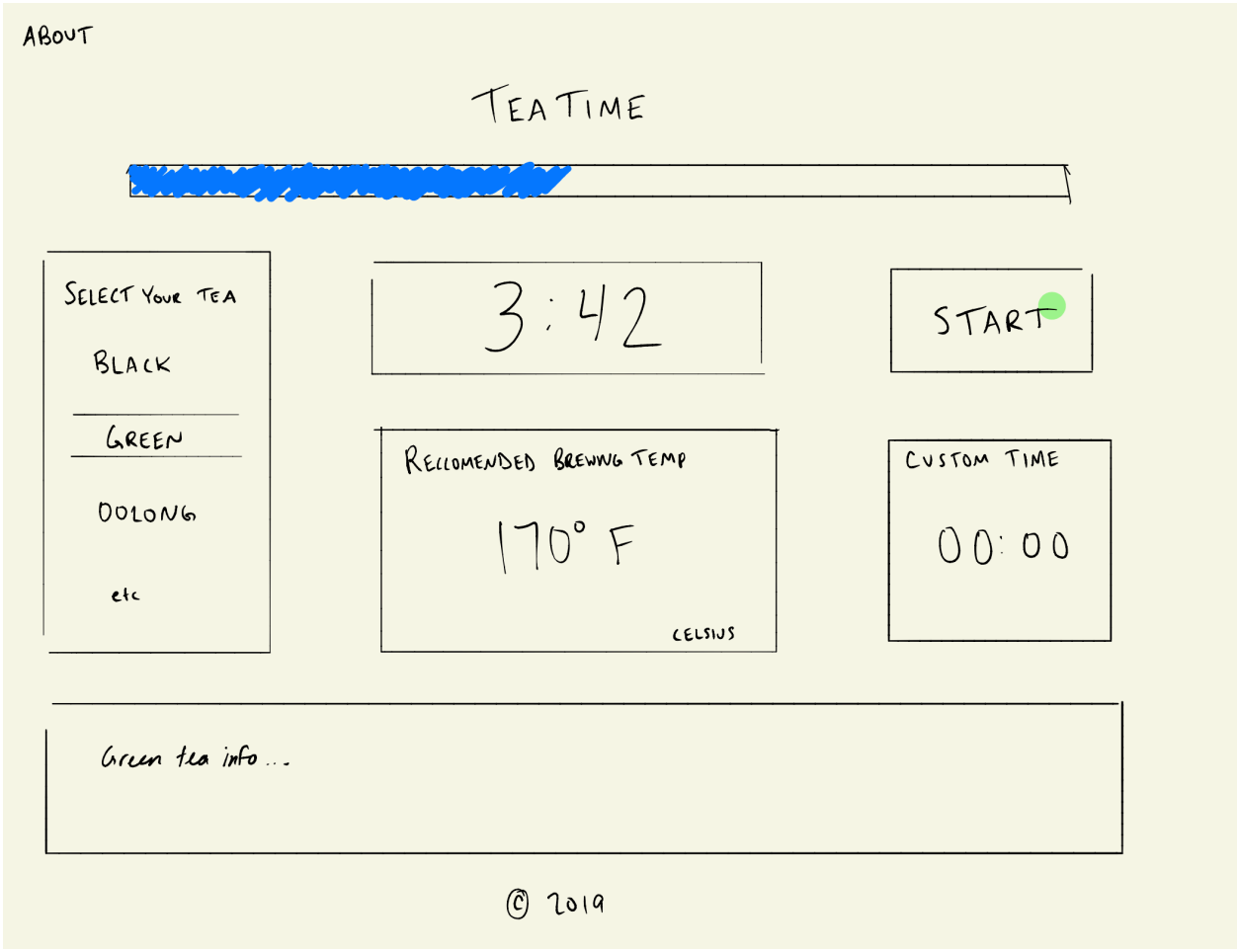
Design and Specification

The initial mockups of this application have stayed pretty consistent to the final design of the application. The progress bar was left off of the final design because it seemed unnecessary when the timer countdown was a better way to show the amount of time left.

Mockup of Layout



Drawings of design and interface with realistic data



Testing

Usability testing was used to locate bugs within this app. The following issues were found during the testing phase of this application:

- Timer malfunctions on object switch
    - When a “tea” object is switched when the timer is already running, the timer would begin to malfunction and switch off showing different timers every second and the timer could not be stopped. This was remedied by putting in a clearTimer function at the beginning of the tea selection change event.
- Firebase data failing to load
    - Data from firebase will not load if the json string is not formulated properly. This occurs when there are characters in the “description” string that are not supported. Such characters that break the database connection are quotations. To remedy this, the data put into the database is made sure to be supported characters only.
- Sound not playing online
    - The sound effect on timer completion is functional when running on local server, but issues with it playing on the hosted site are still a problem.

Restrictions, Limitations, and Constraints

The intended hosting platform of this application was Microsoft Azure. While that was the case for the majority of this application’s development process, within the last update of the

application, problems arose when publishing the application to the server. Many CSS elements failed to load and the connection with firebase would break when accessed from azure. For these reasons, hosting was switched to firebase. With the application hosted on firebase, the only issue that has been found is the lack of sound playing on timer completion.

Conclusion

This project was completed to the developer's satisfaction and has stayed true to the original design and concepts from the start. This was a fun and satisfying project to work on and a greater knowledge of JavaScript and the true core of web development has been improved greatly.

The project can be located online at

- **https://teatime-6336b.firebaseapp.com/index.html**

and the code at

- https://github.com/johare1/TeaTime.git