Brock University

Computer Science Department

COSC 4P02 – Software Engineering 2

Progress Report 1

# HabitForge

www.habitforge.ca

Authors:

Abhi Patel – 6897334 – ap19wf@brocku.ca
Ameen Khawaja – 6935688 – ak19nu@brocku.ca
Zakir Raza – 6834923 – zr19zt@brock.ca
Nico McFarlane – 7001811 – nm20lw@brocku.ca
Rafael Bocsa – 7035801 – rb20qf@brocku.ca
Ahmed Yaser – 7063944 – am20gy@brocku.ca

Professor: Dr. Naser Ezzati-Jivan

## Introduction

The HabitForge team is so excited to showcase the immense progress we have made as a team in the past 2 months, we have covered so much ground and are proud of what we could achieve as a team in such a short span of time. We would also like to announce that we have officially hosted our software on www.habitforge.ca

This site will be updated at every major feature set we introduce, and we cannot wait to ship the final product. In this document we will breakdown what was achieved in each sprint and who it was done by. We hope this gives you a great amount of transparency into our work habits and how we operate as a team.

## Setting up the codebase [Abhi]

Created and set up the codebase and code system that will be used for the development process, this included creating the next app, creating the github repo, setting up Firebase for things like authentication, hosting, cloud storage (Firestore) and cloud functions. This was done before the sprints began as it gave us a good jumping off point for all future planning and gave everyone in the team to clone the repo and explore the codebase to understand what will be expected of them when the time comes for development.

Sprint 1 - 2

## UI Design [Abhi]

Designed and created the UI elements and user flows for the entire software. This included landing page, login screens, dashboard, leaderboards, creating habits, creating habits with the use of audio, and group habits and inviting users. All while keeping responsiveness in mind and putting user experience at the top of the hierarchy. These designs were created using Figma and the Figma file can be viewed using the following link:
https://www.figma.com/file/v9AKPTPlfN2CNPt8IJrauR/HabitForge?type=design&node-id=0%3A1&mode=design&t=eh39iGVUN2nRyEkB-1

I have tried creating a simple and modern design language to simplify the habit forming and tracking flow. Since UI designs were a large task, it was split into both sprint 1 and 2.

Sprint 1

## Cool Down Brainstorm [Zakir]

Researched how we can implement a habit cool down system. The goal of this system is to basically limit the number of habits a user can have depending on their level. For example, if they are level 1, then they can only do two habits. As their level goes up, the number of habits can increase.

To implement this, we can use the points system, so whenever users complete a habit, points get added to the level and then when certain number of points are collected, the user can level up. To control the amount habits users can have per level. We can use an array and increase that array size each level.

## Audio Use [Ahmed]

Researched how different usages of sound indicate different achievements and moods as well as found audio libraries such as SoundBible and FreeSound that offer a wide range of ambient noises for UX design purposes. Implemented storing the ambient sounds in firebase as well as Spotify integration in React to give users the freedom to listen to music within the app in itself.

## Brainstorm for data showcase (Graphs) [Raf]

Researched ways of displaying habit info to users using user data from firebase storage. Representing data in forms of graphs using tools such as chartJS and Plotly make this part of development easy seeing that they integrate with React quite seamlessly. Implementation is as simple as importing the correct folders and using the corresponding React component with the proper chart details and data. Now that we have a way of displaying habit data, now we just have to decide the type of chart/graphs we wish to use.

## System Notifications [Nico]

Implemented system notifications by integrating a notifications API which will be able to prompt the users for permission to display browser notifications. When granted permission, users will receive a notification with the custom message with the Habit Forge icon. The button is enabled in the dashboard which will allow users to manually trigger notifications. This is just a temporary button as we wanted to trigger notifications manually but for the final product, as the notifications will be triggered based on cloud functions. This implementation allows users to stay engaged with any updates for their habits as well as improving the overall user experience and interaction with the notifications.

## Speech to Text [Ameen]

With the current implementation of Speech-to-Text that is done using the React Speech Recognition API, it successfully allows the user to click a button on the webpage and start talking. While they talk, the text is shown in live-time on the webpage and the user can see what they have said. The words the user has spoken is presented to the user in the form of a transcript. I will be working on further improving this in the next sprint, as our ultimate end goal is to have the speech recognition text extract only the important information the user has said, and then form the important words as habits. Also, in future sprints I will implement Polyfill, which is going to assist in making the speech-to-text compatible different types of browsers other than Chrome and Mozilla.

## Backend System (Database) [Abhi]

Designed the data structure that will be used for the product, this included things such as how habits will be stored, how group habits will be structured, how users will be structured and how their notifications will be handled. These designs were just made on notes and will be used as a blueprint when the time comes to connect the frontend to the backend. The backend is hosted through Firebase Cloud Firestore.

Sprint 2

## Landing Page [Zakir, Ahmed]

Worked collectively to implement an interactive landing page of the website using react, next, and tailwind. Both of us worked on one machine and committed through one person (Zakir). Implemented responsiveness to cater for users who use smaller devices such as phones and tablets. Consequently, it will allow users to access their accounts and manage habits from anywhere and any device. This landing page can be viewed at www.habitforge.ca

## Other Sign-In Methods [Raf]

Implemented other sign-in methods using firebase authentication and desired sign-in provider developer details. Created GitHub HabitForge app with App ID and Secret Key used for firebase GitHub authentication. Implemented signInWithGithub method for getting provider details and user authentication all with a great frontend look with the help of Tailwind CSS.

## Speech to Text API Setup [Ameen]

Implemented React Speech Recognition API which is essentially Mozilla’s Web Speech API which is free and built into the browser. This was the most trivial approach that allows for a free way of accessing a speech-to-text API without needing to pay per usage. This approach was done using Typescript.

## In App Notifications [Abhi]

Implemented how in-app notifications will be sent and shown to users. These notifications will include such things like group habit invites, group habit tracking updates, group updates and any updates sent by the HabitForge team. We can also use these notifications as a communication point to the user about new updates and features added.

Sprint 3

## Sprint 3 [Feb 19 – Mar 1]

Sprint 3 will create the foundation for the habit creating, and tracking feature, which is the most important feature for our software, which is why the sprint only has one deliverable, other than the progress report. We are hoping to have a fully working frontend for the habit creating and tracking flow. This will include the dashboard with its empty states and filled in states for habits. The focus for this is to ensure that the frontend has no glitches across any screen size from the smallest of phones to the biggest of monitors. All while adding any micro interactions possible that can improve the user experience.

Futures & Progress Conclusions

## Future

Once sprint 3 is completed we will be at a very good position to move forward with the other features of our software, these will include speech to text, audio feedback, leaderboards, ensuring the habit tracking system cannot be falsified. We will also be creating the group habits and allowing for users to join group habits. A reward earning and leveling system will also be introduced to gate keep users just signed up to create immense number of habits and overload the system. As a user continues tracking habits they will go up in levels and will unlock greater habit creating amount. We will also be ensuring every feature is thoroughly tested before it is shipped to the main product.

## Any Hiccups?

We have so far faced no hiccups or problems while working in a team or working individually. Everyone has met their deadlines, and the work has been flowing smoothly. We also all appreciate the transparency and communication we have as a team. We hope to continue this for the future.

## Conclusions

All in all, we are set for HabitForge to reach the moon! 🚀