Development Plan
Housemates

Team #9, Housemates
Justin Dang - dangj15
Harris Hamid - hamidh1
Fady Morcos - morocof2
Rizwan Ahsan - ahsanm7
Sheikh Afsar - afsars

Table 1: Revision History

Date	**Developer(s)**	**Change**
9/25/23	All	Revision 0
11/1/23	Justin Dang	Update to POC demo
4/4/24	Justin Dang	Revision 1

This document provides the development plan for the Housemates app. The housemates app will allow for its users to better communicate with their housemates. Additionally the app will have a cost management and chore management system to allow for splitting of chores/costs amongst housemates.

1 Team Meeting Plan

The team will have a weekly progress check (1 hour on length) on either Monday afternoon during tutorial time (2:30 - 4:30 PM) or Friday afternoon (around 4:00 PM). The purpose of these meetings is to discuss what group member has done for the project in the last week. Additional meetings can be scheduled throughout the week if necessary. At the end of each meeting a discussion of what needs to be accomplished before the next meeting will occur. The details of each meeting will be recorded as an issue on GitHub.

2 Team Communication Plan

The team will primarily communicate through Microsoft Teams group chat. Discussions regarding specifics of documentation or source code may occur on

an issue on GitHub. Group members can also be contacted by their McMaster email if necessary. All group members are expected to be readily available on these communication channels.

3 Team Member Roles

These roles are subject to change throughout the course of the development of Housemates. All team members are expected to contribute to each of the three subsystems of Housemates (Bill Management, Task Management, Scheduling).

Justin Dang

Scrum Master

- Make sure group is on track to finishing deliverables
- Schedule meetings
- Create meeting agendas

Harris Hamid

Team Liaison

- Handle external communications
- Inform group of any new announcements from Instructor
- Handle any internal communication issues in the group

Fady Morcos

Database Engineer

- Setup database for Housemates
- Design database schema for Housemates
- Troubleshoot issues with the database for Housemates

Rizwan Ahsan

UI/UX Design Expert

- Design the simple and intuitive UI of Housemates
- Ensure that the visual design of Housemates is appealing to users
- Create surveys to get feedback on user experience for Housemates

Sheikh Afsar

DevOps Engineer

- Set up the CI of Housemates
- Set up project structure for Housemates
- If possible implement CD of Housemates

4 Workflow Plan

For the duration of the project, our team will adhere to established guidelines to track and manage all of the code and document modifications. We will also leverage GitHub Actions as our CI tool to automatically build, run tests within PRs and any commits, which determine whether it passes or fails. The workflow is as follows:

- Open and describe a GitHub issue
- Create a new branch to work on the feature/bug
- Work on this feature/bug branch
- Commit changes to new branch
- Create a Pull request for changes
- CI pipeline is triggered
- Review and Approve PR if no test failures
- Merge changes with main branch
- Close GitHub issue

Issue Tags

- Bill Management: Issue deals with bill management system of Housemates
- Task Management: Issue deals with task management system of Housemates
- Scheduling: Issue deals with scheduling system of Housemates
- Account: Issue deals with account system of Housemates
- Miscellaneous: Miscellaneous issue that deals with Housemates that do not fit with the above system tags.
- Documentation: Issue deals with documentation of Housemates
- Deliverable: Deliverable for SFWRENG 4G06
- Review-Team06: Document review from team 6
- Meeting: Team meeting details

5 Proof of Concept Demonstration Plan

One of the main risks in this project is in the client-server interactions that are required for the Housemates application. If the client-server interactions fail then the Housemates application will not be able to work correctly. Another risk is that we won't be able to create one of the main features that are outlined in the problem statement document. As such for this POC demo we are going to focus on showcasing a part of the bill splitting feature of the Housemates application. In the POC demo we will demonstrate a client connecting to a server to split an expense with their group of housemates. This will show both that we can successfully have client server interactions and that one of the major features of the application works.

6 Technology

- MongoDB: NoSQL database to store user information.
- Firebase: Backend database for authentication.
- React: JavaScript library for front-end development.
- Express: JavaScript framework for back-end development.
- Node: JavaScript run-time environment for server.
- JavaScript: Core programming language of the internet.
- VSCode: Convenient, good quality IDE to be used for front and back end development.
- GitHub: To be used for version control, CI, and collaboration.
- ESLint: Static code analysis tool used to find issues in JavaScript code.

7 Coding Standard

The coding standard we will be following will be similar to Java style coding. This means that all variable and function names will be written in CamelCase and will have neat accurate comments describing functionality. ESLint will be the linter the used to enforce this coding standard.

8 Project Scheduling

Deliverable	Deadline
Problem Statement, POC Plan, Development Plan	September 25
Requirements Document Revision 0	October 4
Hazard Analysis 0	October 20
V&V Plan Revision 0	November 3
Proof of Concept Demonstration	November 13–24
Design Document Revision 0	January 17
Revision 0 Demonstration	February 5–February 16
V&V Report Revision 0	March 6
Final Demonstration (Revision 1)	March 24
Final Documentation (Revision 1)	April 4
EXPO Demonstration	April 9