# Sprint 3 Plan

Product: Remember Me

Team Name: RememberMe

Revision: 05/5/2021

High Level Goal(s): For Sprint 3, we want to update our login page and store information into Firebase. We also want to create an ordered profile list with accessible, updated information.

User Stories:

- As a user, when I change devices I would like to be able to transfer data easily - ( 5 )
    - Save data into firebase based on individual accounts

- As a user, I would like my data stored reliably in an account - ( 7 )
    - Data: basic information such as name and notes about the individual

Team Roles:

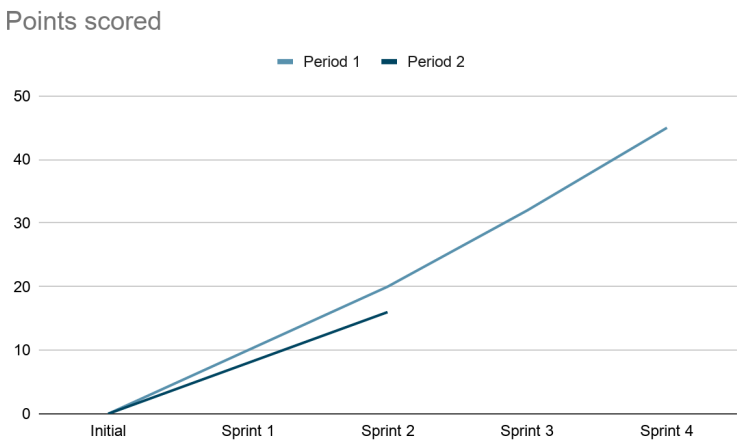
- Dillon Lee: Developer, Researcher
- Sarah Liang: Scrum Master, Developer, Researcher
- Gareth Samadhana: Researcher, Developer, Data manager
- Gavril Tango: Researcher, Developer, Data manager
- Grant Fu: Product Owner, Researcher, Task manager

Initial Task Assignment:

- Dillon Lee: Implement ability to Categorize journal entries by person
    - Initial Task: Utilize Framework and multiple navigable pages for different categories
    - Implement Photo Storage (Backlog)
- Sarah Liang: Improve interface for ease of use
    - Initial Task: Fix react-native bugs and make the interface more neat. Add profiles to contact information
    - Implement Photo Storage (Backlog)
- Gareth Samadhana: Shape up backend storage to match the architecture of front end

    - Initial Task: React-Native and Firebase
    - Match up with Front End (photos and text)
    - Give Firebase ability to create accounts
- Gavril Tango: Shape up backend storage to match the architecture of front end
    - Initial Task: React-Native and Firebase
    - Match up with Front End (photos and text)
    - Give Firebase ability to create accounts
- Grant Fu:
    - Finalize Connection between React-Native-fetch-blob and Python Flask Server
    - Finalize communication messages and content between RN and Flask
    - Take facial “detection” and turn into “recognition” (library included for python face-recognition)

Initial burnup chart:




Initial Scrum Board [Sprint 3]:

| User Stories | Tasks (Not Started) | Tasks (In Progress) | Tasks (Completed) |
|---|---|---|---|
| Store diverse data such as text, contact info, photos, etc …**(Backlog) (4)** | - Store the photo/data that has been taken | - We have the data types we just need to store it | -Initial obtaining of data such as photos, etc. |
| Efficiently catalog data(Backlog) (4)<br><br>And(+)<br><br>Categorize data by person(s) (Backlog)(6) | 1-Following new tutorial methods for a better version<br>2-Connect with backend (includes integrating backend categorization with front end display) | -Modify existing Interface(Dillon and Sarah)<br>- Default organization of entries by date<br>-Connecting backend to front end | 1-Login/logout into backend database<br>2-We have notes/ entries stored just not in organization we intended<br>3-backend side is able to organize by person, (currently incompatible with frontend) |
| As a user, I would like my data stored reliably in an account (7) | 1-reliable data storage completed but must be connected with frontend input/output<br>2-connect with frontend input to store photos and other media | 1-Connecting reliable backend to the frontend input/output | 1-Firebase data storage through queries is functional |

| | | | |
|---|---|---|---|
| As a user,when I change devices I would like to be able to transfer data easily - ( 5 ) | -Connect with a front end login/logout page | | -Log in/logout functions complete despite differing devices |
| Recognize person faces | 1)Finish utilizing face_recognition library<br>2)finalize Flask and RN fetch communication (2-way) | | 1)RN post request to Flask server<br>2)Python detect faces |
| Finding journal entries by person recognized | 1)combine file sharing/ file access of photos with backend/frontend<br>2)resolve any compatibility errors.<br>3)If possible, have both Flask/similar service run on concurrently with Firebase | | |

Scrum Times:

- Monday 2-3PM
- Wednesday: 4-5 PM
- Thursday: 5-6 PM (TA present)