Milestone 3
Abhijit Ravichandran, Chan Wei Fong Max
Team 1803

Level of achievement: Apollo 11

The NUS social app to connect you with like-minded people.
GitHub link: https://github.com/rabhijit/ChilliPadi/tree/final-orbital

1) Dating
    - Meet other NUS students of your preferred gender via a card-based swipe-and-match system
    - Matching with other students initiates a conversation for both of you to interact

2) 'Open jios'
    - Find other people in NUS who have similar interests to you to attend events together with and explore common interests
    - Create open invitations to attend certain events
    - Find other people in NUS studying the same course or module to study together with and discuss school work.
    - Create group chats to discuss the details of the event and open jio.

Motivation

NUS is a place that thrives with events and opportunities for learning and exploration in a social environment. However, many of us are either unaware of such opportunities, or face the social barrier of not knowing anyone of similar interests to attend such events together with.

Moreover, for the majority of NUS students who don't stay on campus or missed out on orientation camps, it's hard to interact and mingle with other students, particularly those of different faculties.

Our app aims to break down these barriers and enhance the social and academic experiences of NUS students by providing them with a comprehensive NUS-centric social platform for mingling with and finding people of similar interests.

Aim
To facilitate bringing NUS students together for a more close-knit and vibrant student experience.

User stories:
1) As a student and frisbee enthusiast who stays at home and commutes to school daily, I would like to meet more fellow casual frisbee players in NUS to enjoy a game every now and then and destress.
2) As a student who missed out on most faculty orientation camps due to army commitments, I do not have many friends taking the same modules as I do, and I would like to meet other students who do take these modules to study together and motivate each other.
3) As a year 4 student who has decided to settle down and start thinking for the future, I would like a medium to meet someone to share my life with.

Scope of project
We will be creating an Android mobile app, using the React Native framework, Firestore database management system and Firebase authentication and file storage system, that provides a medium for swipe-based matching and dating, and an interface to create and accept invitations for NUS-based events and other social events. Users will use their NUSNET ID as a unique identifier to log-in and create accounts.
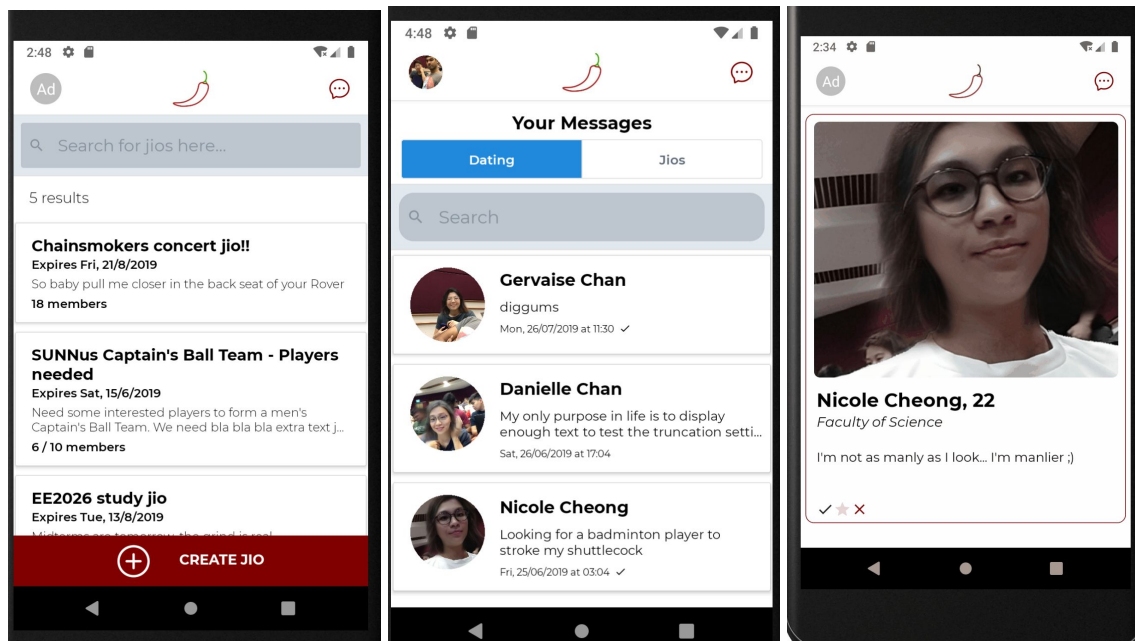
1. Dating
   - Matches will be facilitated if two users have both swiped right on each other. The match will initiate a conversation between the two users
   - The match will allow both users to view each other's profiles, and provide them both with the opportunity to initiate a conversation
   - This will motivate users to arrange for interaction outside our dating app

2. Event invitations
   - Users will be able to create 'open jios' for events that they want to go to, which allows other users to join their 'jio' to attend that event
   - Jio creators can list details of the event they wish to attend, along with maximum sizes for their jios, if necessary
   - Alternatively, users can just join existing jios for events
   - Users can also create or join 'study jios' which will be sorted according to modules / courses
   - Once a user joins a jio, he can enter the jio's chat room where users can openly discuss more about the event and its details.

3. Messaging

- Users will be able to view their dating messages and jio group chats in a unified messaging page
- The messaging page allows you to select whether you want to view your dating messages, or your group chats


From left to right: the jio system, messaging system and dating system

Unique features: what makes us different from other platforms?
- Dating apps (e.g. Tinder)
  - Unique to NUS students only
  - Students will have the option to limit matches to their faculty
  - Conversations will have a time limit (i.e. conversations will expire after 7 days) to facilitate more rapid interaction
- Event management apps (e.g. Eventbrite)
  - Such apps only display a very limited number of events on their website (many events happening in Singapore are not displayed)
  - Does not facilitate the 'open jios' or open invitations to join an event together with people of similar interests - lacking the 'social invitation' aspect of our app

Chilli Padi: the app
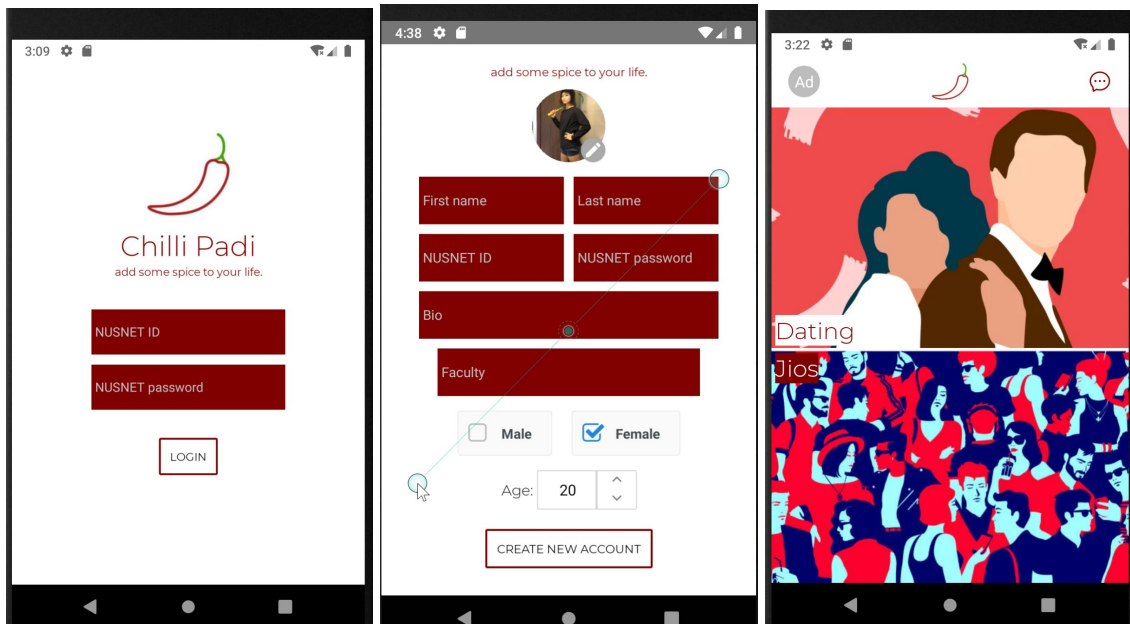
The login page, account creation page and home page, which allows you to select between swiping for matches, looking for jios and seeing your messages

Implementation of features
- Account login, creation and authentication
    - Each person's account is defined by their unique NUSNET ID and password
    - Once a user types in their account details, account authentication and creation is handled through the Firebase Authentication system
    - A user's account stores their full name, NUSNET details, a short bio, their faculty, age and gender
    - Users will be able to view all other user's details, save for their NUSNET details
- Dating: swiping and matching with other users
    - When you swipe right on another user, that user's NUSNET ID is stored under your profile details in the Firestore database
    - That user's profile details are also checked to see if he/she has swiped right on you. If both conditions are fulfilled, a match is created
    - The match gives you the option to view the other user's profile, message the other user or continue swiping
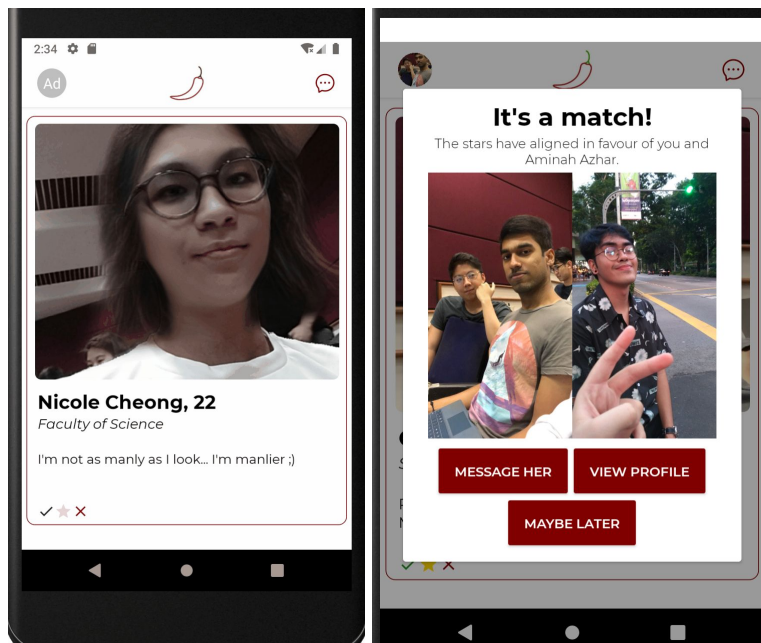
The dating system - swiping and matching

- <u>Jios</u>: creating open invitations and discussing event details with members
    - A user who wants to create a 'jio' can type out the details of the event and publish his open jio
    - The details of the jio will be uploaded to the collection of jios on the Firestore database
    - When other users then look for jios, the app will draw and display all these existing jios from the database. Clicking on the jio will open more details, and an option to join the jio
    - Once a user joins the jio, it opens up an option to join the jio's group chat and discuss jio details
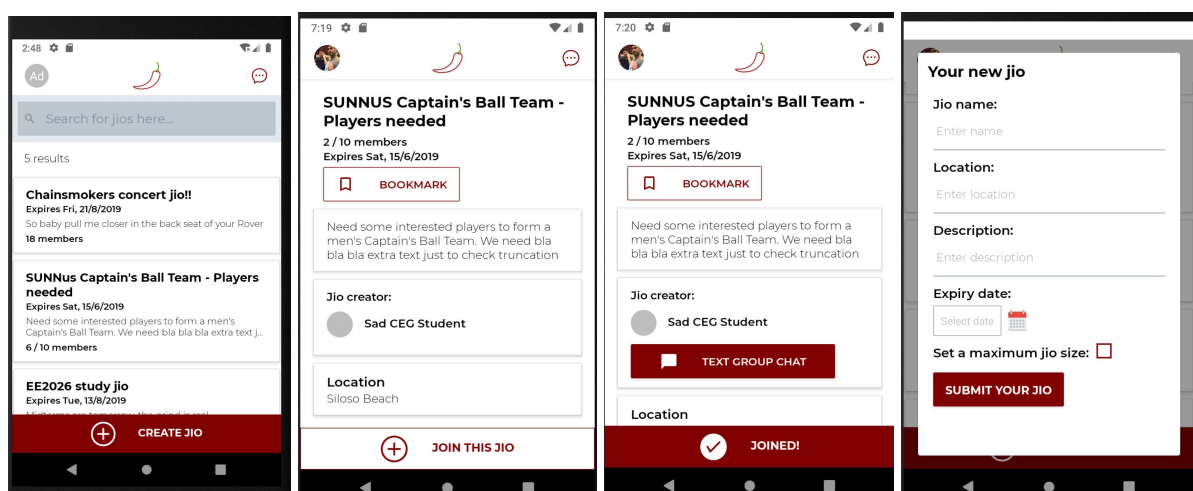
The jio system - looking for jios, joining jios and creating jios

- <u>Messaging</u>: split into Dating messages and Jio chats

- Depending on the field selected by the user, the Messaging page will either draw all the Dating chats, or all the Jio chats from the Firestore database
- The last message in each conversation will be displayed, with a tick to indicate who the last sender was
- Opening up a conversation will allow you to view all messages, and send a new message into that chat
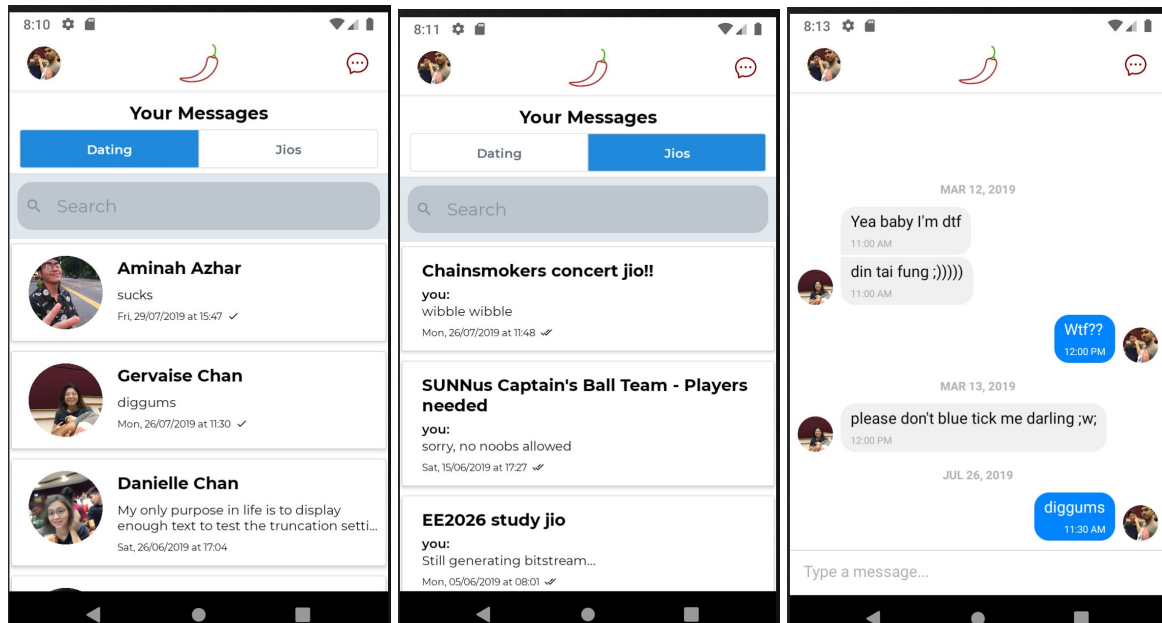
The messaging system - the dating chats, jio chats and an individual dating chat

Testing and debugging done so far
- We have deployed the app on various Android devices, both real devices and Android Studio emulators, to ensure there are no visual oddities or glitches across devices of different screen sizes, Android versions, etc
- We have tested the messaging, jio creation and dating (swiping) system using multiple devices simultaneously (e.g. my partner swipes on me using an account on his phone, and I swipe on him using an account on my phone to test the matching system -- or we message each other using the app installed on our phones)
- The app was tested across a maximum of 4 devices across different Internet networks of varying qualities, to test the speed of updates to the dating, messaging and jio systems

Difficulties faced
- Setting up React Native itself was a complicated affair. Quite a significant number of time was spent fixing issues such as failed installation of React Native modules or installed modules being unable to be linked to our project
    - We also often received strange errors such as apparently not having sufficient storage to install our app onto the emulated Android device, even though there was clearly enough space
    - However, we eventually got accustomed to seeing the recurring React Native errors and learnt to debug and fix the standard errors

- Setting up our back-end solution gave us great difficulty. Being new to back-end development, we struggled to understand the many new concepts behind database management.
    - Initially, we chose Realm Database as our database solution, but we found Realm to be extremely restricted in terms of feature and lacking in terms of ease-of-use and features. We eventually chose Firebase as the best database solution that was much more featured and usable than Realm, but this transition and struggling with Realm cost us a lot (almost a week) of time.
- Testing our app across various devices and making our app look good on different devices (of different screen sizes and Android versions) also caused us some difficulty

Program flow
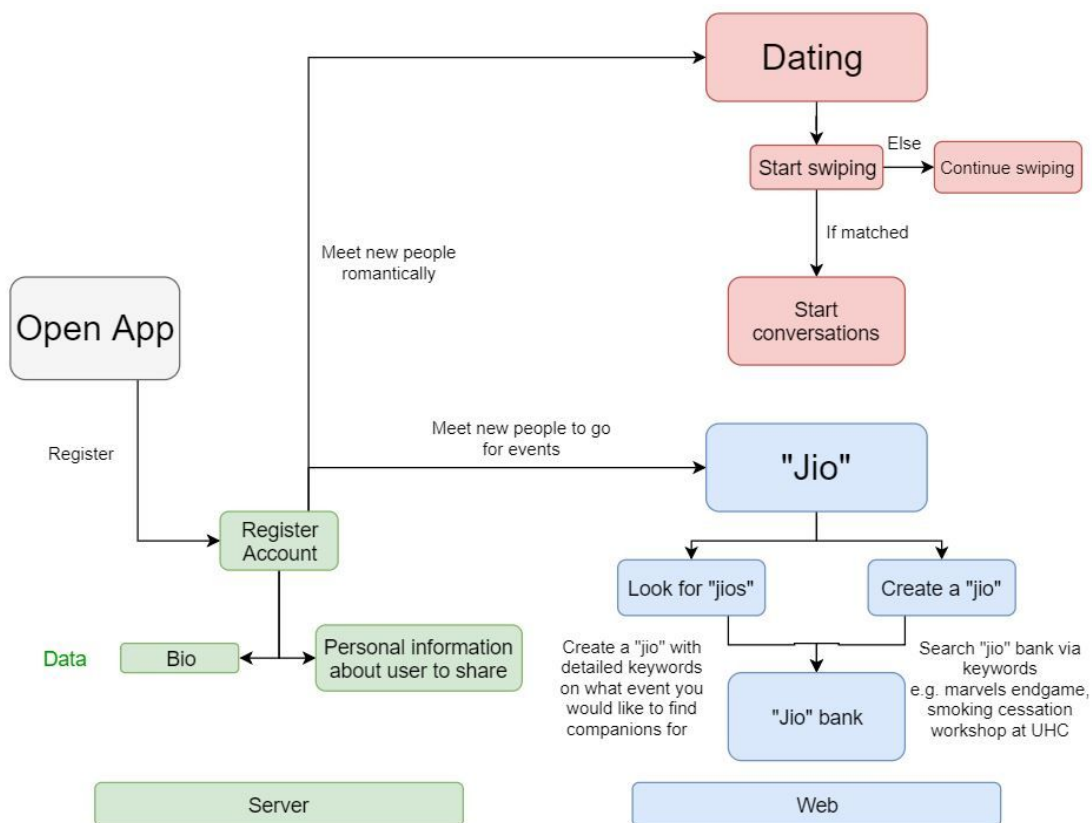
Project log

S/N	Task	Date	Orbitee 1 Abhijit (hrs)	Orbitee 2 Max (hrs)	Remarks
1	Liftoff, Day 1	13/5/19	8	8	Decided on using React Native for app

					development
2	Liftoff, Day 2	14/5/19	8	8	Attended ReactJS workshop and Git workshop
3	Meeting with advisor	14/5/19	1	1	Discussed project ideas and possible frameworks for implementation
4	Team meeting	15/5/19	6	6	Further discussion and solidification of project ideas
5	Team meeting	16/5/19	6	6	Creation of project structure, establishment of version control
6	Programming at home	20/5/19	5	5	Started creating basic framework of app (app home page)
7	Team meeting	25/5/19	10	10	Worked on Milestone 1: readme and poster
8	Working at home	26/5/19	1	1	Final edits for Milestone 1
9	Programming at home	31/5/19	5	5	Creating basic framework of app (app menus: events and jios) (50 here)
10	Programming together at NUS	3/6/19	5	5	Finishing up on the events and jios menu
11	Debugging at home	5/6/19	2	2	Fixing errors with npm manager and React Native modules
12	Team meeting	7/6/19	5	5	Drew out and designed screens for remaining app functions (messaging, dating, login page)

13	Studying at home	8/6/19	1	1	Reading up on useful React Native modules for our app
14	Programming together at NUS	10/6/19	5	5	Finished front-end of login page, designed relevant error messages, tested with dummy data
15	Programming together at NUS	12/6/19	5	5	Started work on messaging system: finished front-end for Jio messaging
16	Programming together at NUS	14/6/19	5	5	Finished up on messaging system: finished front-end for both Jio and Dating messages
17	Studying at home	15/6/19	2	2	Reading up on how to implement a card-swiping system for Android
18	Studying at home	16/6/19	2	2	Continued study on implementing a swiping system in React Native
19	Programming together at NUS	17/6/19	5	5	Started work on dating system: swiping front-end
20	Debugging at home	18/6/19	5	5	Resolving errors and bugs in swiping system
21	Debugging and studying at NUS	19/6/19	5	5	Finished up fixing errors in swiping system: started looking at Database solutions for our app
22	Studying at home	20/6/19	4	4	Choosing between and trying out various database systems (Firebase, Realm, etc)

23	Studying together at NUS	21/6/19	8		8	Decided on Realm for database, started studying and implementing into our app
24	Programming at home	23/6/19	4		4	Successfully implemented Realm database for Jio management
25	Team meeting	29/6/19	8		8	1 week break was taken due to reservist; worked on README and video for Milestone 2
26	Working at home	30/6/19	2		2	Further edits to Milestone 2
27	Programming at home	2/7/19	5		5	Transitioning from local Realm database to Realm Cloud
28	Programming at home	3/7/19	5		5	Dealing with difficulties in Realm Cloud
29	Team meeting	15/7/19	8		8	Long break was taken due to overseas travel; Discussed further features to implement, continued transition to Realm cloud
30	Programming at home	16/7/19	5		5	Realm Cloud plans abandoned due to extended difficulties and limited abilities of Realm Cloud; began transitioning to Firebase
31	Studying at home	17/7/19	5		5	Studying up on Firebase and Firestore
32	Programming at NUS	19/7/19	8		8	Successfully

					transitioned all data from Realm to Firebase
33	Programming at NUS	20/7/19	8	8	Began integrating our React Native code with Firebase database; settled Firebase authentication
34	Programming at home	22/7/19	8	8	Finished integrating Jios with Firestore database
35	Programming at home	23/7/19	8	8	Started integrating messaging system with Firestore database
36	Programming at home	25/7/19	8	8	Finished integrating Messaging with Firestore; started on integrating Dating/Swiping system with Firestore
37	Programming at home	26/7/19	8	8	Finished integrating Dating with Firestore; started fixing up issues and front-end glitches; testing system on multiple devices
38	Programming at home	27/7/19	8	8	Finishing up on front-end fixes; continued testing across various devices
39	Working at home	28/7/19	5	5	Started work on Milestone 3 README and poster
40	Team meeting at NUS	29/7/19	10	10	Finished up on Milestone 3 requirements; celebrated end of Orbital! (for now)

Total hours: 444
Orbitee 1 (Abhijit): 222
Orbitee 2 (Max): 222