M1: Requirements Specification

Study Buddy

Nima Nasiri, 36969475
Joe Puthenkulam, 49190135
Yuyi Wang, 60661196
Yiyi Yan, 40800161

CPEN 321, 2019W
October 1st, 2019

PART 1: Status Update **3**
Progress made since M0 3
Plans until M2 3
Decisions and Changes in Project Scope 3
Individual Contribution 4
PART 2: Project Requirements **5**
Use Case Diagrams: 5
Study Buddy: Use Case Diagram 5
JobToGo: Use Case Diagram 6
Non-functional Requirements 6
StudyBuddy: Non-functional Requirements 6
JobToGo: Non-functional Requirements 7
Sequence Diagram 7
Objects explanation 7
Screen Sketches 9

PART 1: Status Update

Progress made since M0

- Front-end: Developed flow diagrams for main functionality, account setup, creating groups, chat functionality and “Meet up now!” bonus feature. Set up the Facebook login SDK for the login page.
- Back-end: Decided on tech stack and how back-end services are going to communicate with each other.

Plans until M2

- Design app logo and icon
- Draw out all pages of the app (main page, settings page etc.)
- Learn how to implement at least 3 major functionalities:
    - Learn and implement real time location sharing using Google Maps API
    - Implement Calendar setup using Google Calendar API
    - Implement Create group functionality.
- Finalize main flow between the pages and features.
- Decide on adding local calendar from device and in app editing features.

Decisions and Changes in Project Scope

- Included location sharing as basic functionality.
- Added accept/decline invitation (suggestion from TA) before accepting newly created group.
- “Meet up Now!” feature: Have “last active location”(current location depending on location sharing setting) of users visible on the map, so their friends can set up an immediate meeting by clicking on their icon.
- Decided on main functionality swipe flow of App, will be like Snapchat; where leftmost page is for active chats, left page for friends list and “Meet up Now!” feature, right page for Calendar, rightmost page for Settings.

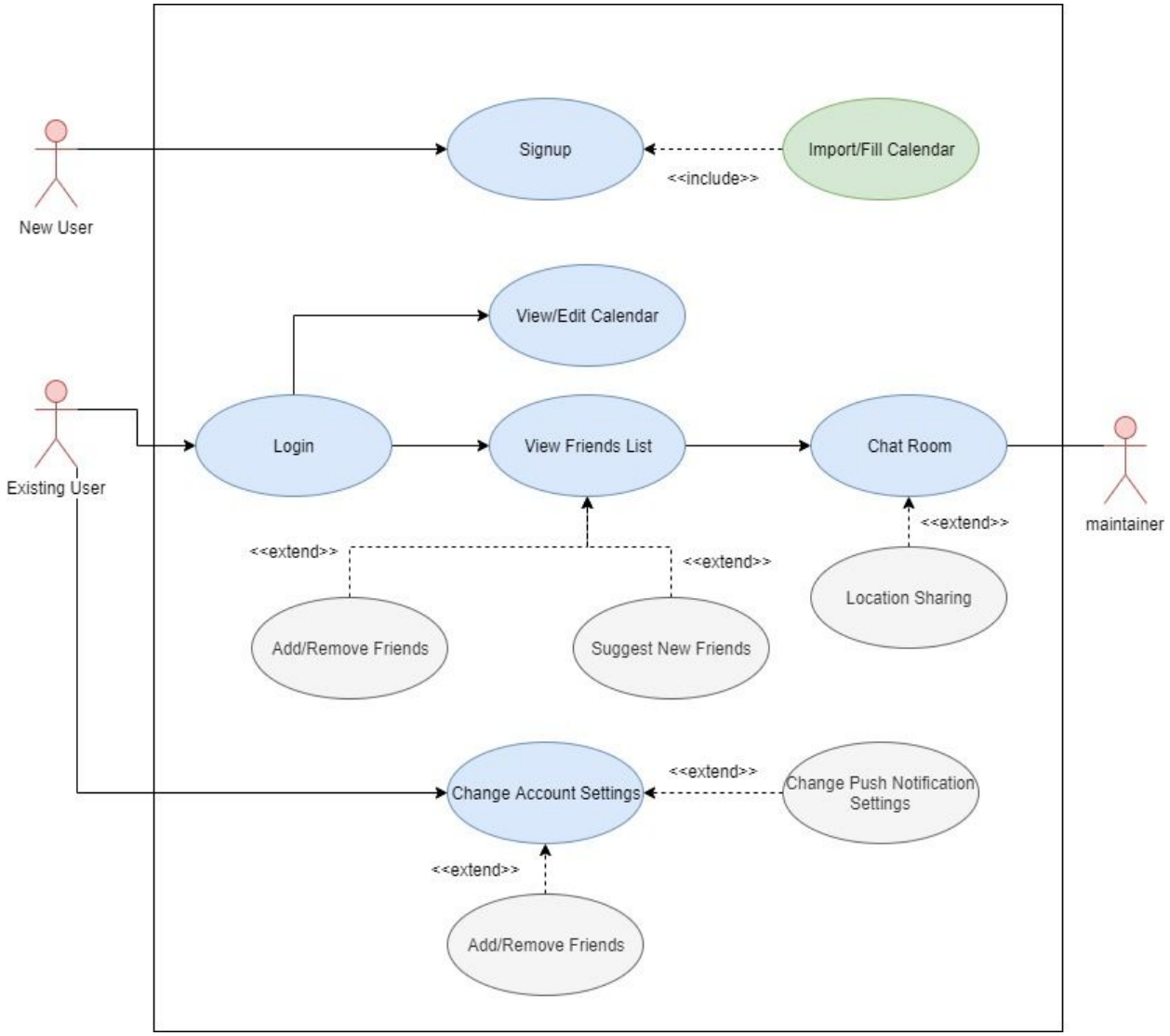
Individual Contribution

- Joe: Drawing flow diagrams of functionality and learning Android Studio, Google Calendar API and Facebook login API
- Nima: Learning NodeJS and MongoDB in addition to the backend tutorial, major contributions in the design and flow of the app
- Yuyi: Setup the Facebook SDK for the login page, learning Android Studio
- Yiyi: set up AWS, and dockerize backend. Learning NodeJS and MongoDB. Made screen prototype sketches.

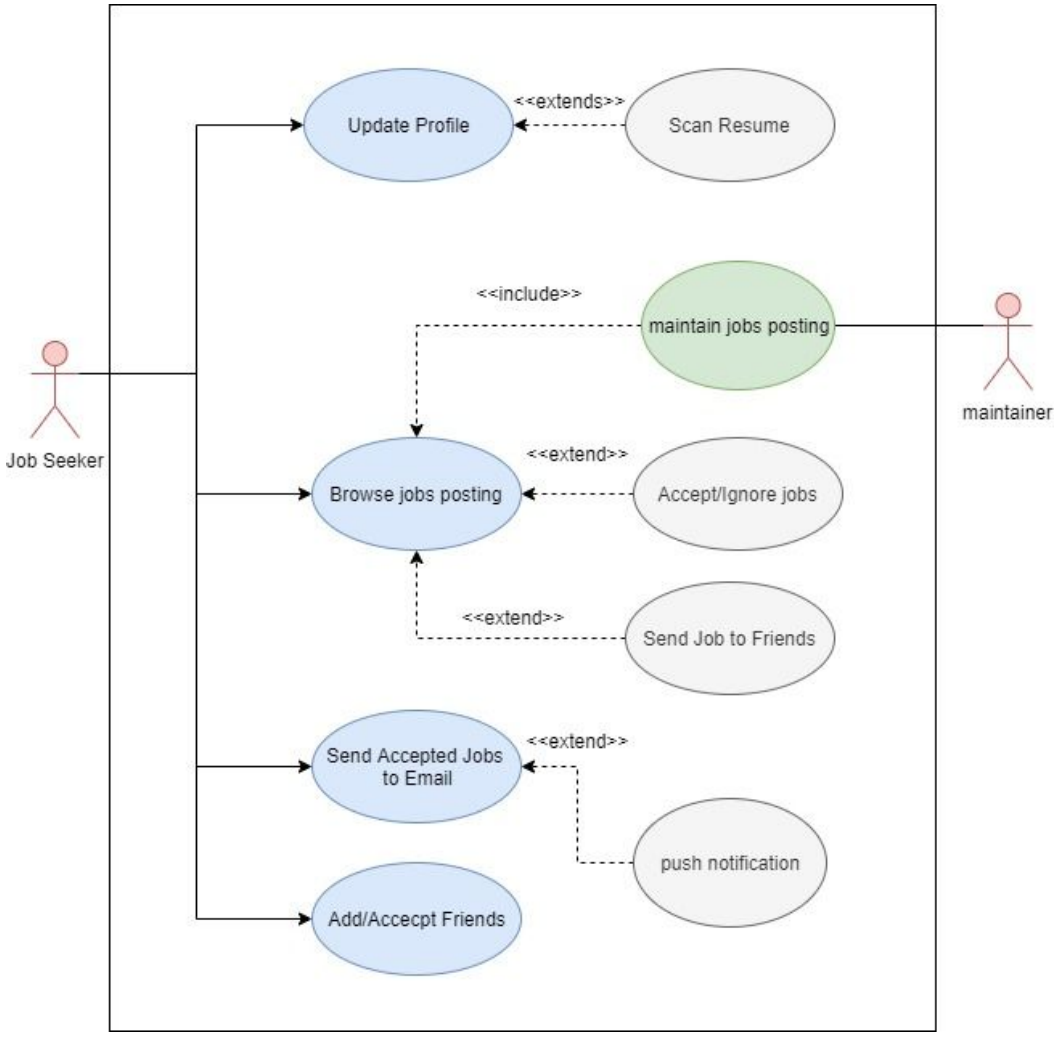
PART 2: Project Requirements

Use Case Diagrams:

Study Buddy: Use Case Diagram



JobToGo: Use Case Diagram



Non-functional Requirements

StudyBuddy: Non-functional Requirements

- Messages received within 500 milliseconds
- Pop ups and page switches execute within 100 milliseconds
- Push Notifications received within 30 seconds of starting a “free” block
- Friends list changes updated after refreshing page (leaving and coming back to it)
- Filtering friends takes effect after typing each letter within 100 milliseconds

JobToGo: Non-functional Requirements

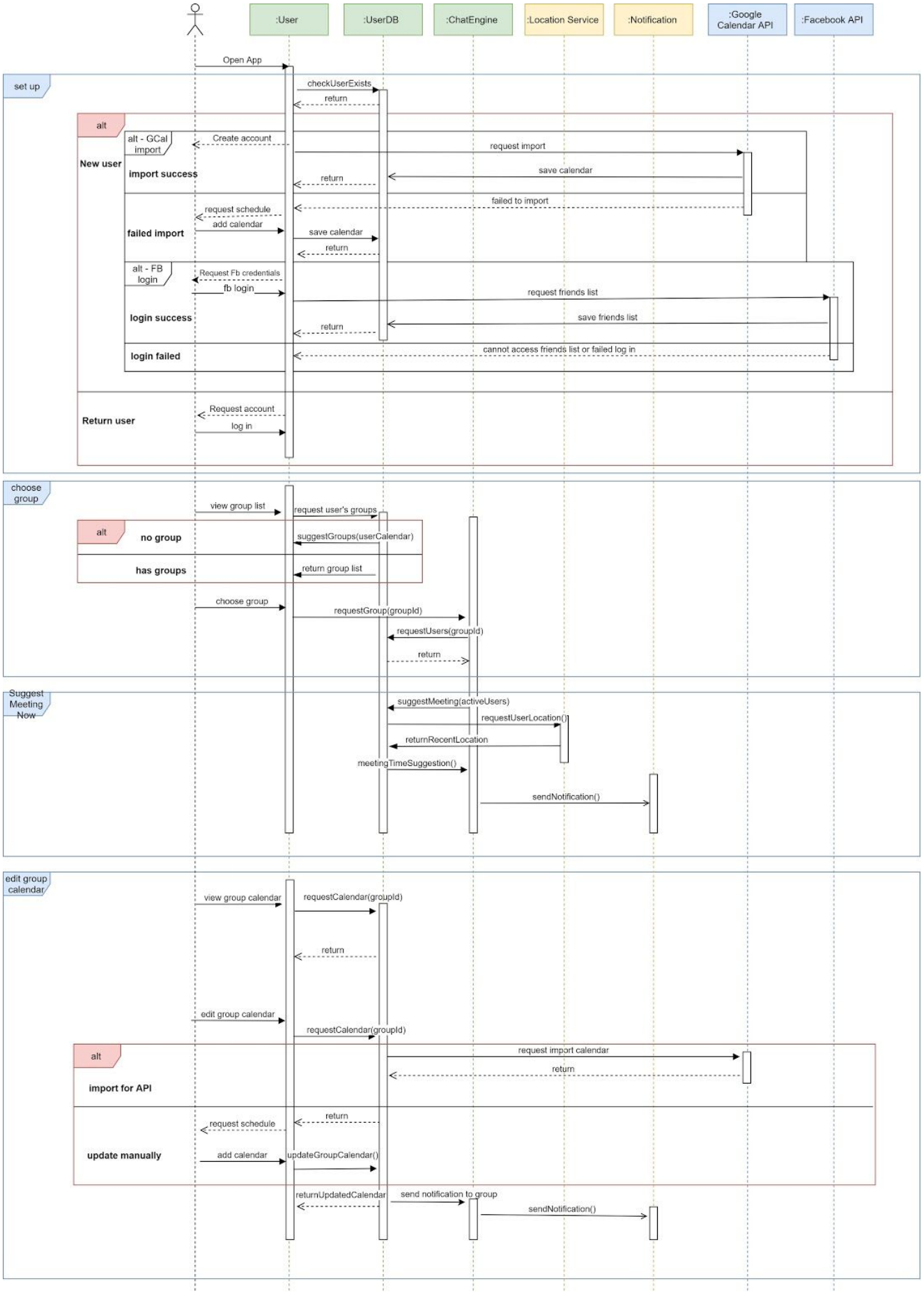
- Swiping is instant, takes at most 100 milliseconds after swiping away from one job to see the next one
- Friend receives push notification of a job sent to them, or friend request within 30 minutes of it being sent
- Email that contains all accepted jobs reaches inbox within 30 seconds of it being sent in-app
- Takes at most 10 seconds after a resume scan is complete to have the information filled into the app

Sequence Diagram

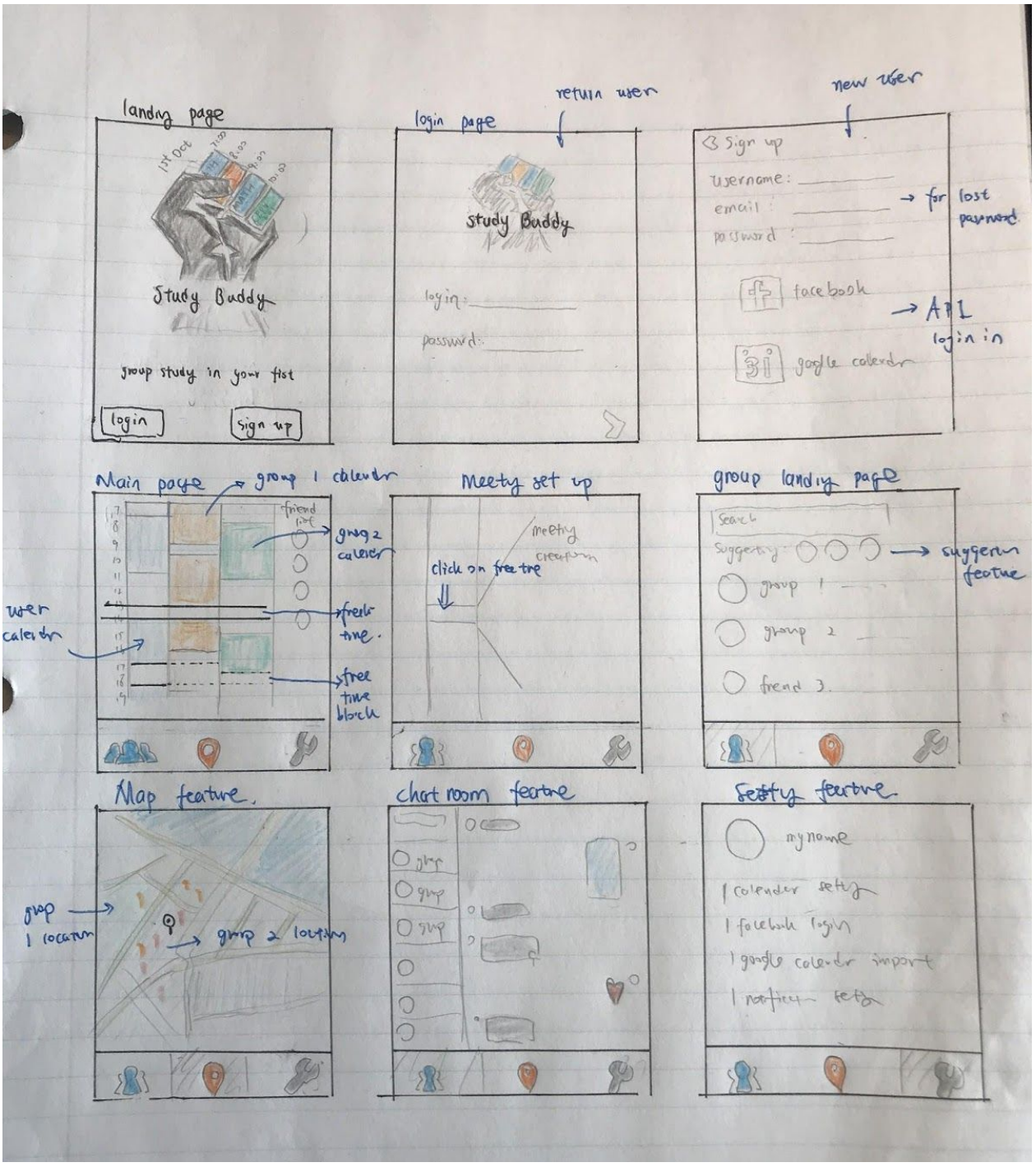
Objects explanation

- **User**: the app representation of user, it's also the android app that users will interact with
- **UserDB**: Database in the backend server where all user related data are stored. It includes user's calendars, the groups that users have, all data that are related to groups (such as group's calendar, group's users and meeting events)
- **ChatEngine**: a chatroom service that gives users real-time chatting service (used socketIO), with chat bot to give meeting event suggestions to the group, as well as new people to add to group suggestion.
- **Location Service**: Android native service that returns the current location of the user based on setting.
- **Notification**: real-time notification calls to the Android phone.
- **Google Calendar API**: external API that leads to google calendars login and return ical files of logged in users.
- **Facebook API**: external API that leads to facebook login, returns the friend list of the users.

Note all three sequence diagrams have been combined into one image*


:User
:UserDB
:ChatEngine
:Location Service
:Notification
:Google Calendar API
:Facebook API
Open App
set up
checkUserExists
return
alt
alt - GCal import
Create account
request import
New user
import success
save calendar
return
failed to import
request schedule
failed import
add calendar
save calendar
return
alt - FB login
Request Fb credentials
fb login
request friends list
login success
save friends list
return
login failed
cannot access friends list or failed log in
Return user
Request account
log in
choose group
view group list
request user's groups
alt
no group
suggestGroups(userCalendar)
has groups
return group list
choose group
requestGroup(groupId)
requestUsers(groupId)
return
Suggest Meeting Now
suggestMeeting(activeUsers)
requestUserLocation()
returnRecentLocation
meetingTimeSuggestion()
sendNotification()
edit group calendar
view group calendar
requestCalendar(groupId)
return
edit group calendar
requestCalendar(groupId)
alt
request import calendar
return
import for API
return
request schedule
update manually
add calendar
updateGroupCalendar()
returnUpdatedCalendar
send notification to group
sendNotification()

Screen Sketches



Note The flow of the system is from top left to button right. The return features are not included in the sketch, but will be similar to the small arrows in the login page.*