When2Meet For Mobile User Manual & Technical Manual

Table of Contents:

General Information

<u>Group 8</u>

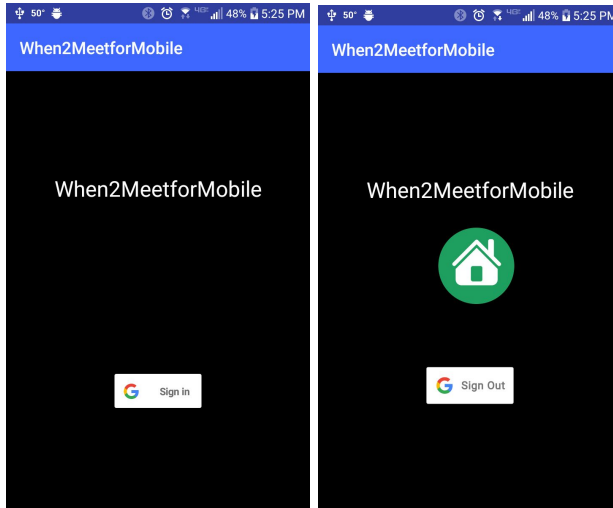Evelyn Bailey (ebail@seas.upenn.edu)

Sang Lee (eunsang@sas.upenn.edu)

Diana Marsala (marsalad@sas.upenn.edu)

Saniyah Shaikh (saniyah@seas.upenn.edu)

Screens

Main Screen

The 1st screen shows when the user first opens the app, or has logged out from a previous session. The sign in button opens a pop up window, which allows the user to login using their Google sign in credentials.

The 2nd screen shows after the user has successfully logged in. The sign out button signs the user out of the app. The green icon with a house moves the user to the Home Screen.

Home Screen
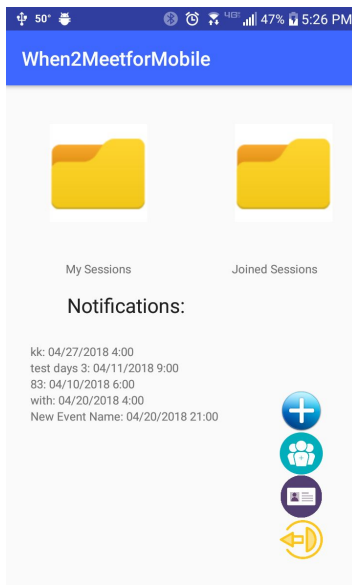
By clicking on the My Sessions folder, you will reach the My Sessions Screen, which lists all the sessions you have created.

By clicking on the Joined Sessions folder, you will reach the Joined Sessions Screen, which lists all the session you are in that you did not create.

By clicking on the plus icon, you will reach the Create Session page, which enables the user to create a session with a set of dates and maximum start and end times for each day.

By clicking the icon with three people, you will reach the Join With Code Page, which enables the user to input a code to join a created session.

By clicking on the purple Profile icon, you will reach the Profile Screen, where you can change your profile picture, name, phone number, and short description.

By clicking on the yellow icon, you will logout of the app and return to the Main Screen with the option to login again.

Join With Code Screen
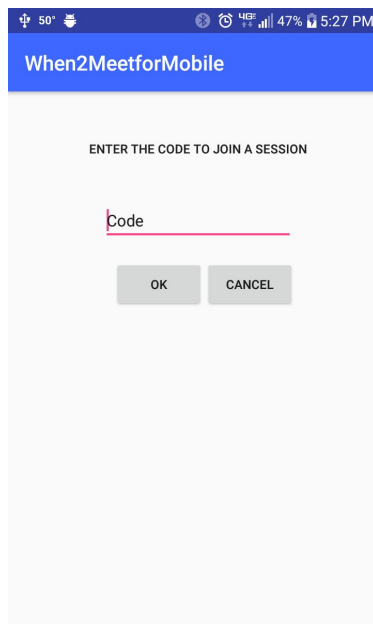
By inputting the code unique to the meeting, the user can join an existing meeting session. The code can be obtained by a text invite from the original meeting owner.

By clicking Ok, the user adds their account to the specified meeting and goes back to the home screen.

By clicking cancel, the user terminates the process and goes back to the home screen.

Profile Screen

This screen displays the option to update the user's profile. The "Username" field updates the user's display name, the number field updates the user's phone number, the "Description" field updates the user's short bio, and the image updates the user's photo. When clicked, the image button selects images from the user's photo gallery. Submit updates the database based on the current fields, and cancel returns the user to the home screen.

Create Session Screen


On this screen, the user can create a session for a meeting. The user can choose between specific dates and days in a week. When the user chooses days in a week, it will record the subsequent week in the calendar. For time, the user can choose between 12 am to 11 pm. The

code for the meeting is displayed at the bottom. By clicking Create, you create a new meeting session. You will be brought back to the home screen.
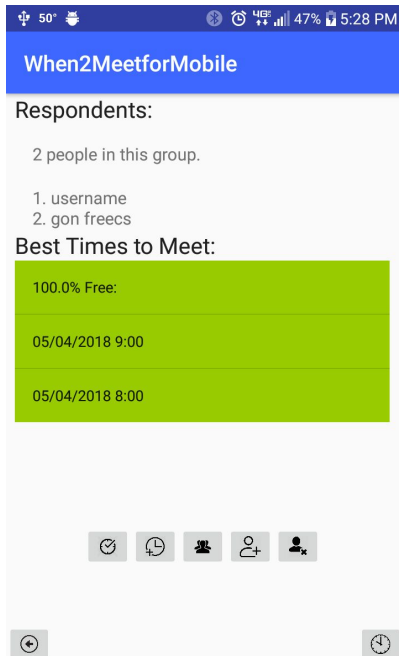
My Sessions Screen
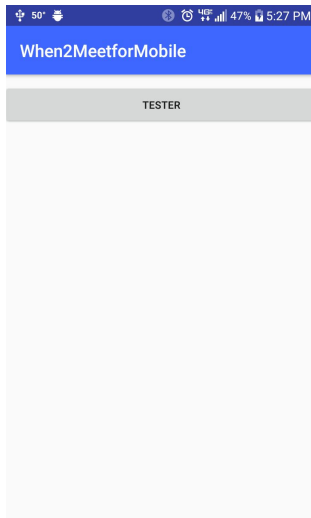
This screen displays one button for each meeting for which you are the owner. The button contains the name of the meeting.

By clicking on a button representing a meeting on this screen, you will be taken to the My Session Display Screen for that meeting.

My Session Display Screen

For a specific meeting, this shows the number of people in the group, lists all of the respondents by user name, and lists the best times to meet.

As exemplified on this page, the best times for the session will be sorted by the percentage of people that are free to meet at that particular time. If there are no times for which at least 50% of respondents are free, the time is not listed here.

In addition to this information, since you own the meeting, this screen also has some special administration options located in the bar below the listing of best times. These include buttons, from left to right, for Select Final Time, Add More Times, View User List, Add Users, and Remove Users.

By clicking the Select Final Time button (radio button with a check inside), one of the best times for this meeting will be selected, the meeting will be deleted, and a notification will be sent to everyone who was in the meeting with the details of the meeting time. This does not take you to a new screen

By clicking the Add More Times button (clock with a plus sign on the bottom left side), you will be taken to a screen which allows you to add additional options of days and times to the meeting, which happens on the Add Times Screen.

By clicking the View Users button (silhouettes of three people together), you will be allowed to view all the users in this meeting, which happens on the View Users Screen.

By clicking the Add Users button (white silhouette of a person with an plus on the bottom right side), you will be allowed to add users from the meeting, which happens on the Add Users Screen.

By clicking the Remove Users button (silhouette of a person with an X on the bottom right side), you will be allowed to remove users from the meeting, which happens on the Remove Users Screen.

At the bottom right of the screen, the clock bottom takes you to the Enter Times Screen for this meeting, which will allow you to put in your availability. At the bottom left of the screen, there is a back button which returns you to the My Sessions Screen.

Add Times Screen

This screen allows you to add more times to the selected meeting. At the top of the screen, you will see the current dates and times of the session. Then, the buttons on the calendar section will each represent a date that can be added to the meeting; the dates already in the meeting are not included. The earliest and latest times for the meeting can also be changed, and the options in the spinner for earliest time will be early that the currently stored minimum, while the options in the spinner for latest time will be later than the currently stored maximum.

After clicking "Add Times" to submit the data to the database, the user is sent back to the My Session Display Screen for that meeting.

View Users Screen
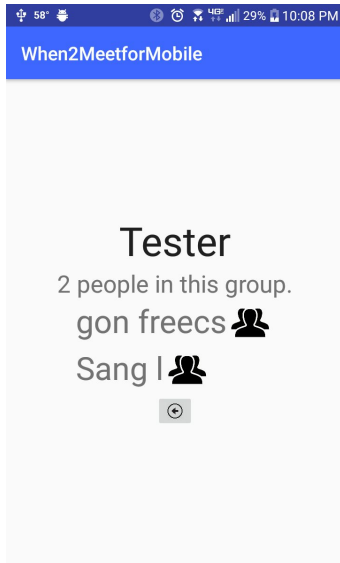
This screen shows the users in the selected meeting. The Name of the meeting is on top. It displays the number of people in the meeting group. Each user's user name and their profile photos are displayed. The back button will bring you back to the session display screen.

Add More Users Screen
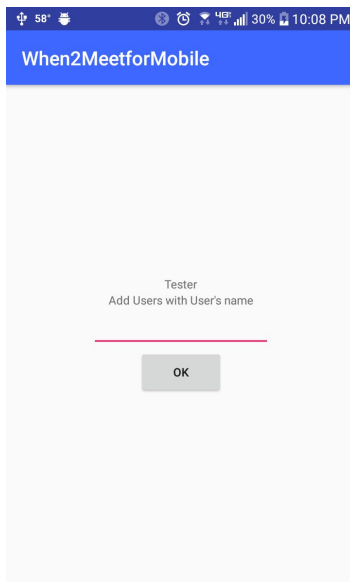
The user can add more users to a meeting using the user's Google Name. The user can add one or more users at a time. The user cannot add themselves. The screen shows the meeting name at the top.

Remove Users Screen

This screen allows you to select users to remove from the meeting by using the checkboxes to the left of each listed user. You will not be listed on this page since you are the owner of the meeting and it would not make sense to remove the owner from a meeting.

Once you click "Remove Users", the selected users will be removed from the database for that meeting, and you will be sent back to the My Sessions Display Screen for that specific meeting.
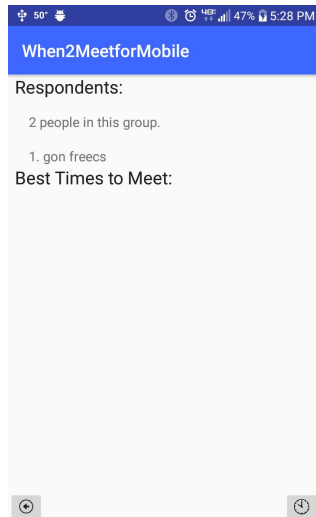
Joined Sessions Screen
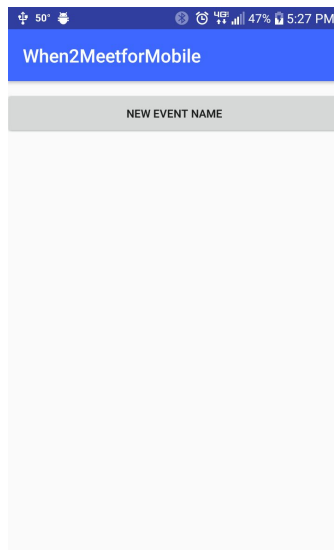
This screen displays one button for each meeting that you are in, so long as you are not the owner of that meeting. The button contains the name of the meeting.

By clicking on a button representing a meeting on this screen, you will be taken to the Joined Session Display Screen for that meeting.

Joined Session Display Screen

For a specific meeting, this shows the number of people in the group, lists all of the respondents by user name, and lists the best times to meet.

At the bottom right of the screen, the clock button takes you to the Enter Times Screen for this meeting, which will allow you to put in your availability. At the bottom left of the screen, there is a back button which returns you to the Joined Sessions Screen.

Enter Times Screen
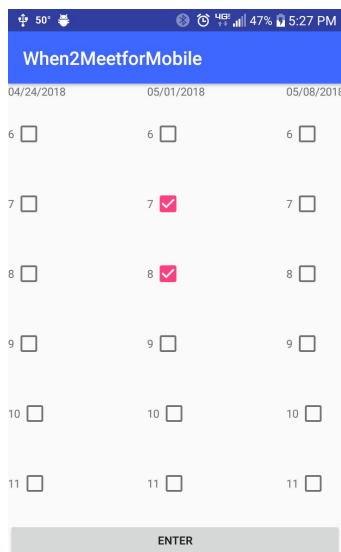
The Enter Times Screen lists all the dates in the selected session and lists buttons for each hour within the allowed availability for each day. (The screen scrolls both vertically and

horizontally to fit the days.) Clicking Enter updates the user's availability for that session, displays that on the Joined Session Display screen, and returns you to that page.

When2Meet For Mobile Technical Documentation

Table of Contents:

General Information

Group 8
Evelyn Bailey (ebail@seas.upenn.edu)
Sang Lee (eunsang@sas.upenn.edu)
Diana Marsala (marsalad@sas.upenn.edu)
Saniyah Shaikh (saniyah@seas.upenn.edu)

Helper Classes

User

This class stores the name (associated with the Google account), userName (display name), phoneNumber, description, and image (base 64 encoded string) associated with each user. It is used to store all users in the database.

InternalUser

This class stores the times that this user is available as a List of Strings called myTimes. Each String is of the format "dd/mm/yyyy h" and represents a full hour of time when the user is free. The hours are represented on the 24 hour clock. It is used to save the times each user is free in a Meeting object.
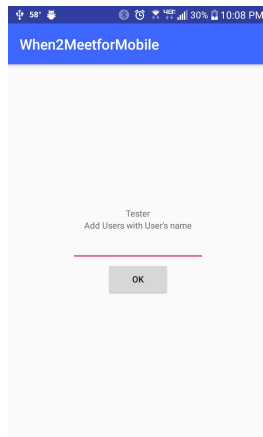
Meeting

This class stores the users (Map from user id to InternalUser object), dates (List of Strings in format dd/mm/yyyy), high_time, low_time, name, and owner (as a user id), as well as an indicator showing that the meeting is in days of the week versus dates (isDays). It is used to save and read meetings to and from the database. It also has functions that allow you to add or remove a user, check if a user is in the meeting but not as the owner, and get a map of the times for which a certain number of respondents are available.

DayData

This helper class contains too maps - one from the short string representing a day of the week to the integer for that day in Calendar (dayToDate), and one from the integer for a day in Calendar to the short string representing that day of the week (dateToDay). There are functions to retrieve the String day from the int date, get the int date from the String day, and get the String day from a String representing the date (ie "04/21/2018" -> "sat").

Activities

Add More Users Activity

This Activity accesses the Meeting in the database with the Meeting's ID passed from the previous activity's intent. The user enters the google name of existing users in the database. You can input one or more users. If the user enters their name, they will not be added again. Using this name, the program finds the User object in the database in the User document and then updates the Database by adding the User object into the Meeting object in the Meeting database document.

Sources: AddMoreUsersActivity.java, Meeting.java, activity_addmoreusers.xml

Add Times Activity
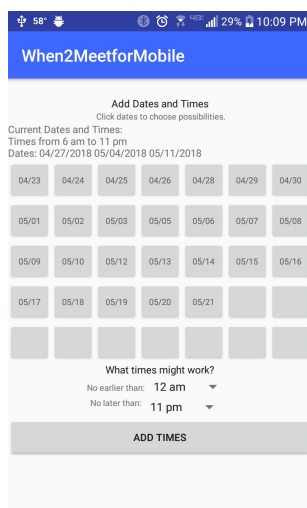
This Activity allows the user to add additional dates or times to a meeting that they have created. It will read the meeting into a Meeting object, then initialize all the calendar buttons or the days of the week based on what days / dates are already in the meeting, and also display what is already in the meeting in a TextView. When ever one of the buttons is clicked, it will either add or remove the associated date to a list, and then once the user submits the changes, the Meeting object gets updated with the new dates and the low and high times are changed, and then it is rewritten to the database.

Sources: AddTimesActivity.java, Meeting.java, DayData.java, activity_add_times.xml

Create Session Activity

Create Session Activity will create a new Meeting Object with dates, time, name and owner user's ID. and add it to the database. At this moment, the Meeting will only have the owner as its User. The Meeting Object will be added to the Meetings document in the database. This Meeting will have a unique code.

Sources: CreateSessionActivity.java, activity_createsession.xml, Meeting.java, User.java

Enter Times Activity
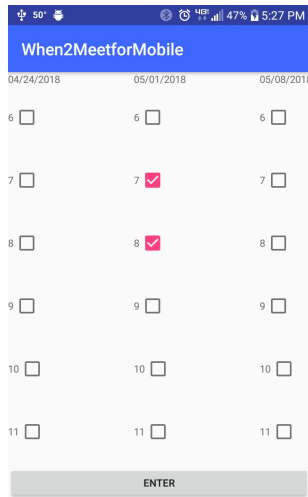
The EnterTimesActivity gets the meeting the current user has selected, and displays the available dates and the hours of availability between the high and low time for each day (loadChoice). When the user clicks enter, EnterTimesActivity stores the list of times in the database by creating an InternalUser object with the user's ID and List<String> of all the times the user selected.

Sources: EnterTimesActivity.java, activity_enter_times.xml, Meeting.java, User.java
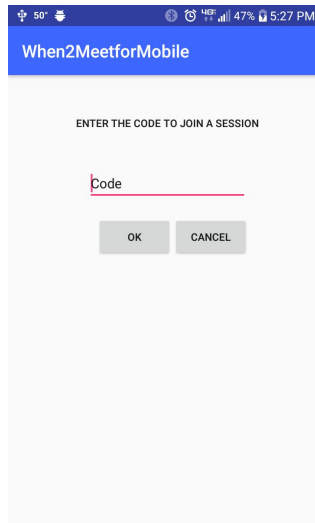
Home Screen Activity
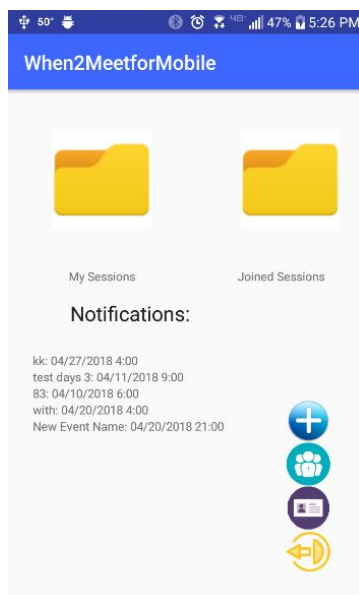
Displays the My Session folder (sessions the user has created), the Joined Sessions folder (sessions the user has joined and not created), and from top to bottom, the Create Session

button, Join With Code Button, Profile button, and logout button. It also displays notifications from the database for completed sessions.

Sources: HomeScreenActivity.java, activity_home.xml, Meeting.java

Join With Code Activity

When you click Ok, the activity will return the code entered by the user to the HomeScreenActivity.java. In HomeScreenActivity.java, the entered code will be used to update the database. The Meeting with the code will be found in the database and the Meeting object will be updated with the current User's unique ID number.

Sources: JoinWithCodeActivity.java, activity_joingingwithcode.xml, Meeting.java, User.java, HomeScreenActivity.java, activity_home.xml

Main Activity
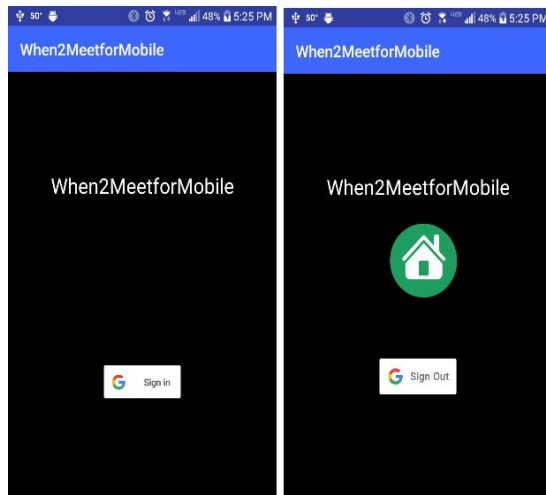
The 1st screen shows when the user first opens the app, or has logged out from a previous session. The sign in button opens a pop up window using the a Google Auth call, which allows the user to login using their Google sign in credentials. Once the user is logged in, the sign out button signs the user out of the app. The green icon with a house opens the Home Screen Activity.

Sources: MainActivity.java, activity_main.xml

Profile Activity
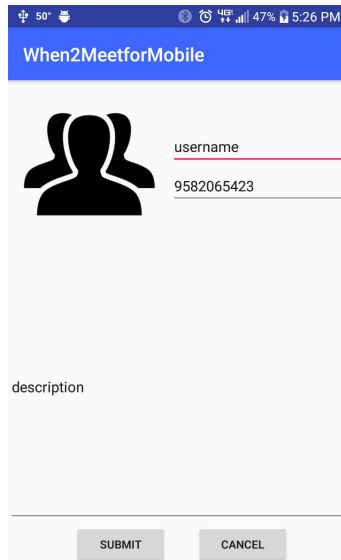
Uses the current user's ID to autofill in the Username, PhoneNumber, and Description fields with the current data or displays default labels if these fields are empty. The image displayed is the last image uploaded or the default image if none have been selected. Once the user submits, it stores the text in each TextView in the corresponding field in the User object. The image is compressed and then encoded as a Base 64 String.

Sources: ProfileActivity.java, activity_profile.xml, User.java

Remove Users Activity
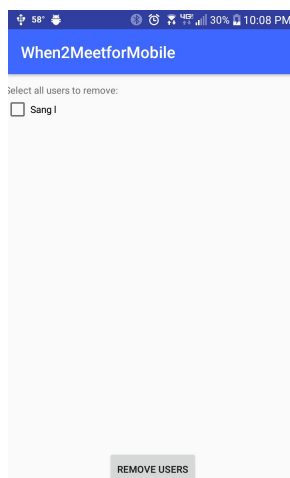
This Activity allows the owner of a session to remove user(s) from the session. After reading the data into a Meeting object using the meeting id passed to the Intent, it creates a checkbox for each user that is not the owner that displays their username and adds or removes their associated user id to a list. When the user hits the button to submit, the database is updated by removing the users in the list from the Meeting object and then overwriting the meeting in the database.

Sources: RemoveUsersActivity.java, Meeting.java, activity_remove_users.xml

Sessions Activity
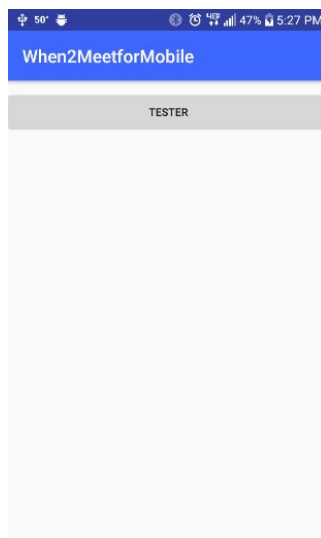
This Activity displays a list of meetings based on which button was clicked from the Home Screen. If the My Sessions folder was selected, then populateCreated() is called to create a button for every meeting that the current user owns. If the Joined Sessions folder was selected, then populateJoined() is called to create a button for every meeting that the current user has joined but does not own. Each button is given an onClickListener that will pass the meeting id, user id, and whether or not the user owns the meeting to the Session Display Activity.

Sources: SessionsActivity.java, Meeting.java, activity_sessions.xml

Session Display Activity
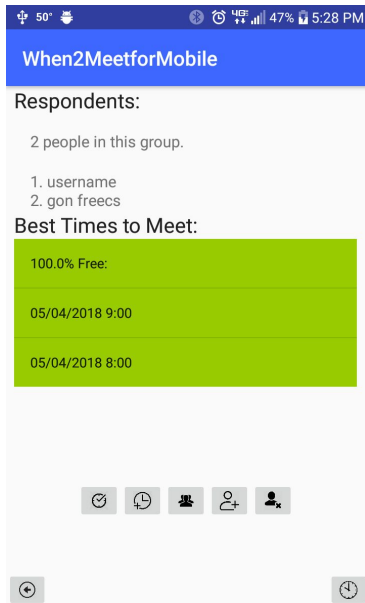
This Activity gets and displays the information associated with the given meeting by retrieving the meeting from the database, getting its users and displaying the user names for those who have responded, and getting the best times and displaying them, and displays the administrative buttons if this meeting is being viewed by its owner. It also handles the selection of the final time for the meeting, and passes meeting and user information to the activities that handle the administrative buttons if this is a created session.

Sources: SessionDisplayActivity.java, Meeting.java, InternalUser.java, activity_session_display.xml

View User Activity
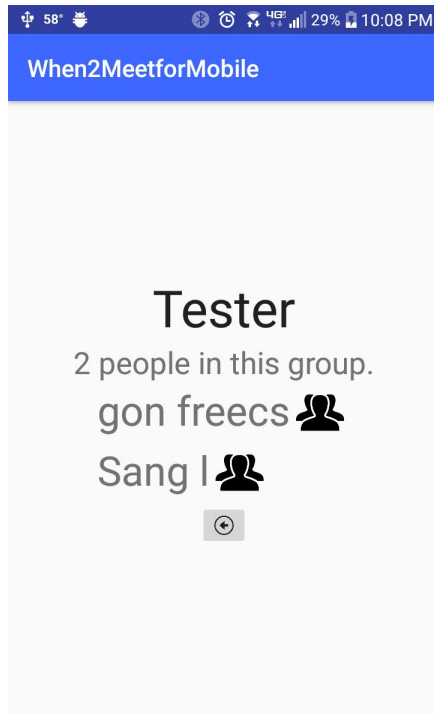
The activity accesses the current Meeting object from the database. After getting the ID numbers of each Internal User listed in the Meeting, the program access the Users document in the database to find the user names and the profile pic of each user in the Meeting object.

Sources: ViewUserActivity.java, activity_view_users.xml, Meeting.java, InternalUser.java

Database Layout

We used Firebase as our database to store meetings, users, and notifications. Data is written and read from the database by the use of the Meeting, InternalUser, and User objects.

Meetings

Stores a list of all Meeting objects that users have created as documents. Each meeting contains a list of dates specified for that meeting; high_time, which specifies the latest time a user can list their availability for each day; low_time, which specifies the earliest time a user can list their availability for each day; name, the name of meeting; owner, the ID of the user that created the meeting; a list of InternalUsers, which are users who have joined this session and contains a list of available times each user has selected for this meeting.

Notifs

Stores a list of each user ID of all users and lists all the notifications for that user. The notification lists the meetings that have selected a final time.

Users

Stores a list of all the users by ID. Each user has a name, which is the same as the Google account name; username, which is displayed in app and can be changed via the profile; image, which contains a base 64 encoded string of an image; phone number, which is the user's number; and a description, which is a short bio for the user.

Access

This Gmail account has access to the database:

Name: When2 Meet
Email: group8.when2meet@gmail.com
Password: 350When2Meet-G8!

Go to https://console.firebase.google.com/u/0/project/when2meetformobile/overview to view the database once you are logged in with this information.