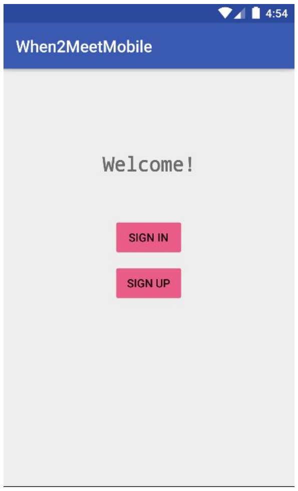
USER MANUAL

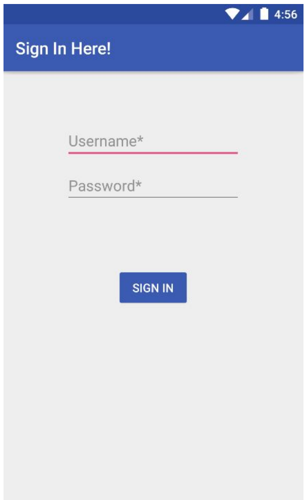
Launch Screen

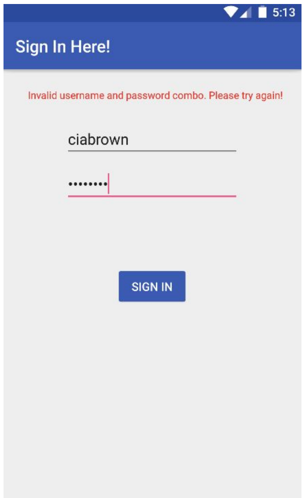
When the user opens the app and is not already logged in, they will be directed to the launch screen. They will see two buttons: Sign In and Sign Up. The user should tap Sign In if they already have an account, as they will be directed to the sign in page. If the user does not already have an account, they should tap Sign Up. They will be directed to the sign up page where they can create an account.



Sign In Screen

On the sign in screen, the user will see a field for “username” and “password”. The user should type in their username and associated password in the correct fields, and then click sign in. If the credentials they entered were accurate, the user will be redirected to the main menu page. However, if the username/password combination was incorrect, an error message will be displayed when the sign in button is clicked. Likewise, if the user leaves either field blank when they click sign in, an error message will appear to fill in both fields.

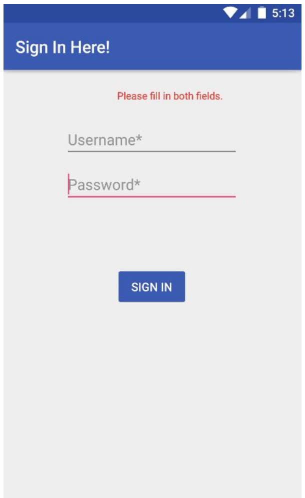




Invalid username and password combo. Please try again!

ciabrown

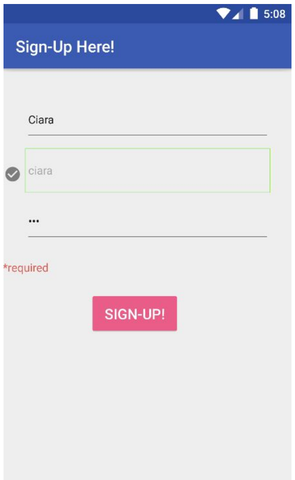
••••••••

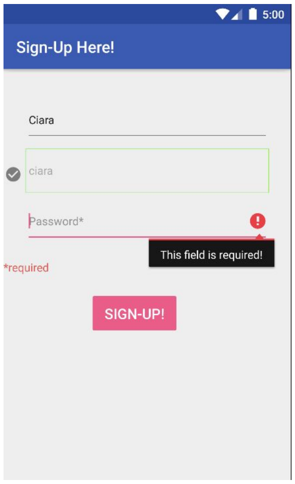


Please fill in both fields.

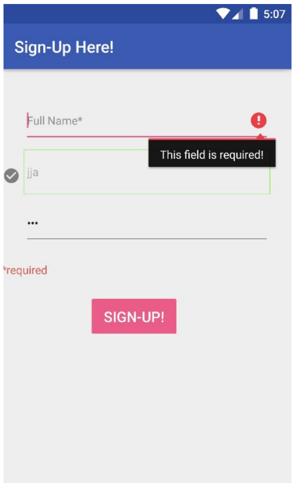
Sign Up Screen

On the sign up screen, the user will see fields to enter a full name, username, and password. The user must fill in all three fields before they click the sign up button. As the user types in a username, the text field will be outlined in green and a checkmark will display if the username is available. If the username is unavailable, the text field will be outlined in red and an “x” will appear next to the text field. If the password field is empty, it will appear red and an alert will display that indicates the password field is required. If the full name field is empty, it will appear red and an alert will display that indicates the full name field is required. Upon clicking sign up, if all fields are filled in and the username is available, the user will be logged in and directed to the main menu. If the user clicks sign up and one or more of the fields is empty, the user will remain on the sign up page.

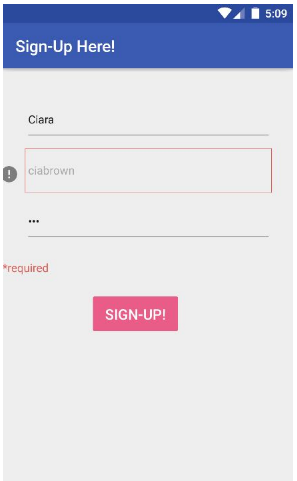




This field is required!

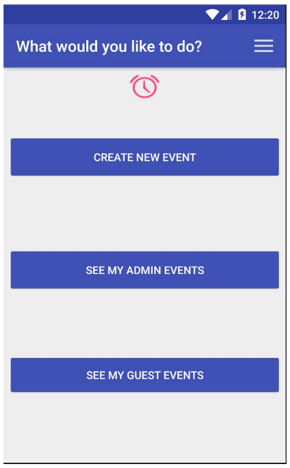


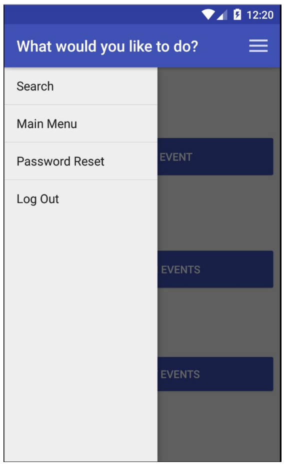
This field is required!

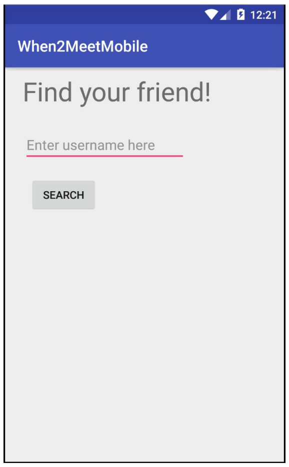


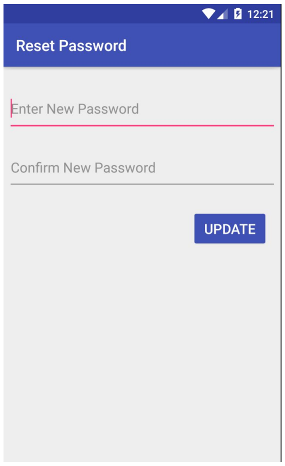
Main Menu Screen

On the main menu screen, there are three large buttons: Create a new event, See My Guest Events, and See My Admin Events. Clicking Create a New Event directs the user to the create event page to begin to create a new event. Clicking See My Guest Events takes users to the See My Events page, which in this case is filled with events that the user was invited to and has accepted. Clicking See My Admin Events takes users to the See My Event page, which is filled with events that the user created and/or has been given admin rights to. The Main Menu also has a hamburger menu, which can be accessed by clicking the hamburger icon in the top right corner of the screen or by dragging from the left side of the screen. The hamburger menu can be closed by clicking the hamburger button or by clicking anywhere on the app beside the hamburger menu itself. The hamburger menu has four buttons: Search, Main Menu, Password Reset, and Log Out. By clicking on the Search button, the user will be directed to the Search page. On the Search page, the user can type in a username and click enter, and a textview will appear indicating whether there is an account associated with that username. The Main Menu will return the user to the Main Menu upon being clicked. The Reset Password button will take the user to a page, where the user can enter and set a new password. The Log Out button will log out the user from his/her account and send the user to the Sign-In page.









Create Event Page

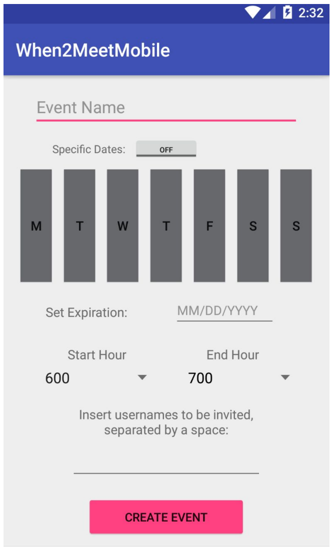
The Create Event page is the page by which the user creates an event. The user is required to fill in all event attributes, beginning with event name. By clicking the text field, the user may type in his/her desired event name.

Below the Event Name field is the date selection for the availability poll. This poll is to gauge people's specific time availabilities. The "specific dates" toggle allows the user to decide if he/she wants the event to have specific date options or day-of-the-week options for the poll. If the user toggles on specific dates, the Date Range Picker page is launched (see Date Range Picker Page). Upon return from successful date range selection, the previous area holding the days of the week rectangles will be populated with two text fields that specify the user's selected start and end date for the availability poll. If the user chooses to keep the specific dates toggle off, then he/she may click on the desired days-of-the-week boxes, one for each day of the week, that the user wants as options on the availability poll. Each click on a rectangle toggles the selection of that specific day of the week. A green-colored rectangle indicates a successfully selected day of the week.

Below is a text field for the expiration date, to the right of “Set expiration:”. The user must type in a correctly-formatted (follows the pattern given in the hint MM/DD/YYYY) expiration date for the event. After this date, invited guests may no longer RSVP or fill out availability polls. The expiration date must be after the event and must be a valid date.

Below the expiration date are the start and end date dropdown spinners. If the user clicks on the box with “600”, indicating the auto start time, there will be a dropdown of options for start hours, given in military time. To set the start time, simply click the desired time, and the spinner will close out. The same follows for end time selection, although the user will not be able to select end times that occur prior to the selected start time.

Finally, the bottommost text field allows the event creator to invite guests. To invite a specific guest, the user must type in the username of the desired guest in the text field. Multiple guests may be invited, so long as their usernames are separated by a space. If the user tries to submit the create event form with a username that is not associated with any user currently, the user will be notified of the error in a pop-up dialog and asked to correct the form. To finalize and create the event after filling out all required fields, the user must click “Create Event.”



Event Name

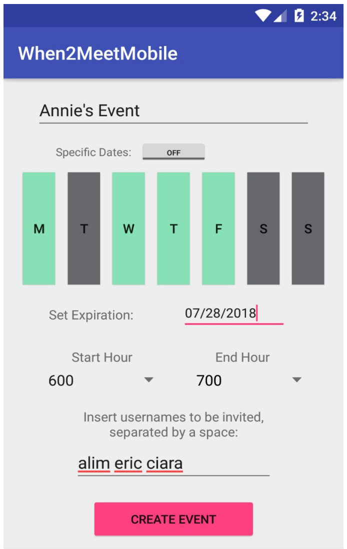
Specific Dates: OFF

Set Expiration: MM/DD/YYYY

Start Hour: 600

End Hour: 700

Insert usernames to be invited, separated by a space:



Annie's Event

Specific Dates: OFF

Set Expiration: 07/28/2018

Start Hour: 600

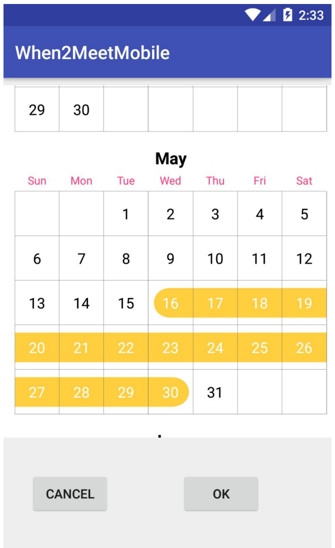
End Hour: 700

Insert usernames to be invited, separated by a space:

alim eric ciara

Date Range Picker Page

The Date Range Picker page is accessed through the Create Event page, and is launched when the user toggles on the specific dates toggle. The date range picker is a calendar view of each month, starting from the present day that the user is creating the event. The user then clicks a date. This will be set as the start date. The user then can select the end date by clicking a date that occurs after the selected start date. Consequently, all the days in between the selected start and end date will be highlighted yellow, indicating the date range the user has selected. If the user would like to select a different range, simply click a new date, and this will be set as the start date. If the user has selected a start date and attempts to select an earlier date than this start date, this secondly selected date will become the new selected start date. If the user clicks the cancel button, he/she will be redirected to the event page, with the specific dates toggle off. If the user has selected a range of dates from the Date Range Picker, he/she may finalize this selection by clicking the “Ok” button, which will redirect the user to the Create Event page.



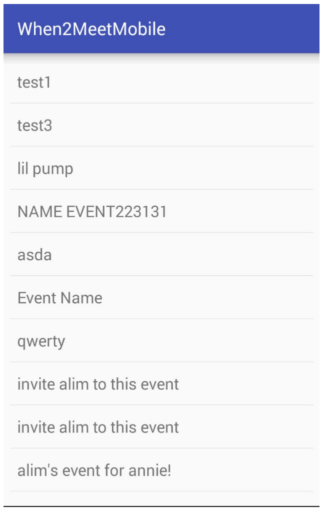
29	30					

May

Sun	Mon	Tue	Wed	Thu	Fri	Sat
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

See My Admin Events Page

The See My Admin Events page gives a list of all events in which the user is an admin. The list is given in the form of links displaying the names of the events. By clicking on these links, the user is redirected to the event page specific to that event, and that event page will be either an admin or an owner event page depending on whether the user is an admin or an owner of that event. If the user is not an admin in any events, the list will remain unpopulated.



When2MeetMobile

test1

test3

lil pump

NAME EVENT223131

asda

Event Name

qwerty

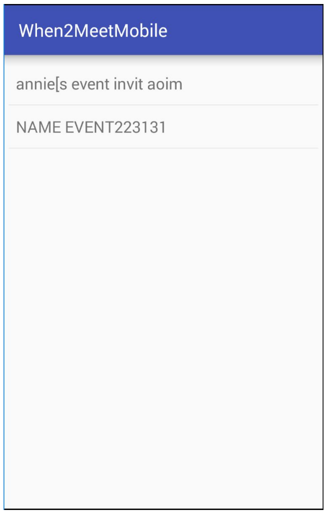
invite alim to this event

invite alim to this event

alim's event for annie!

See My Guest Events Page

The See My Admin Events page gives a list of all events in which the user is a guest who has accepted the invitation to join an event. The list is given in the form of links displaying the names of the events. By clicking on these links, the user is redirected to the event page specific to that event. If the user is not an admin in any events, the list will remain unpopulated.



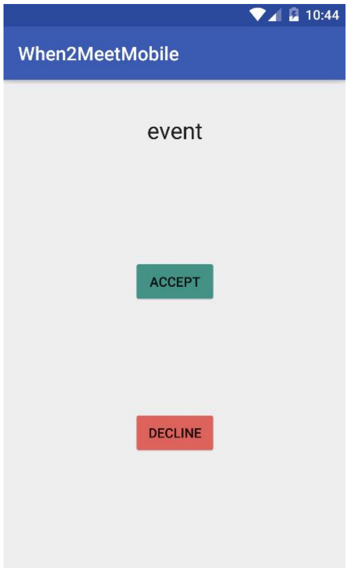
When2MeetMobile

annie[s event invit aoim

NAME EVENT223131

Invite Page

The invite page is an intermediate page to which the user is brought on clicking a notification which states that they have been invited to the event. If they click accept, they are added to the event's accepted list and the event is added to their guest events, then redirected to the event page. If they reject, then the event is removed from their invitations, they will not be added to it, and they are then redirected to the main menu.



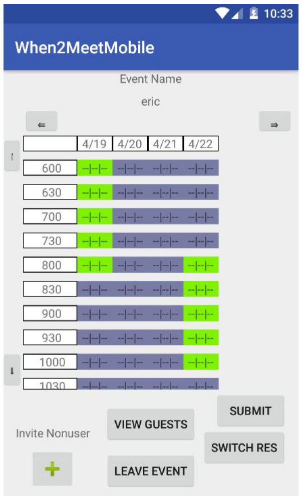
Guest Event Page

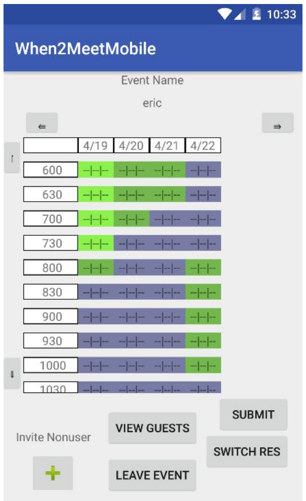
The event page represents an event in which the user is a guest. On loading, the user can see the event name at the top center of the screen, with the event owner directly below it. The user also has access an up arrow button, a down arrow button, a left arrow button, a right arrow button, a button to invite a non-user, a view guests button, a leave event button, a switch res button, and a submit button. At the center of the screen is a grid with the top columns labeled with the dates of the event if it has specific dates, or the days of the week if not. The far left column is populated with military times within the time range where the user may respond. By clicking the cells below the date labels and to the right of the time labels, the user may respond. If the user clicks a cell corresponding to a time or date outside of the accepted range, the click is ignored. If the click is on a cell within the active range, the cell's status is flipped. If the user had not previously submitted a response selecting the cell, the cell will initially be blue. On selecting an active cell, the cell will become green, otherwise it will be blue. The user's prior responses are loaded into the grid when the page is loaded initially, and all cells matching the user's responses will be made green. After altering their response, a user may click submit to submit the changes to their responses. By clicking the switch res button, the user may switch between the two views of the page. The first page, as mentioned above, displays the user's own input. The second displays all responses to an event. The cells are colored based on the percent of participants that have activated those cells. All inactive cells or cells without respondents are blue. The cells with respondents become increasingly dark with more users being available during the time they represent. When in this view, a user may click a cell within the accepted time range to have a popup appear that displays the names of all the users that have responded in that time slot. Note that responses made in the first view will not dynamically affect the second view's population, and so the page must be reloaded after the user submits to have the second view be accurate. In the event where there are more dates than there are labels in the top rows, the left and right buttons may be used to switch between the visible dates. The right arrow shifts the date or day of the week to the further along, while the left moves it back. In the event that the user is either at the end of the range or there are not enough dates to need to move around, the buttons will not do anything. Similarly, the up and down arrow buttons will shift the visible time range backwards or forwards respectively. If they are at the end of the range for the times, or there are not enough times to overflow the display, the buttons will have no effect.

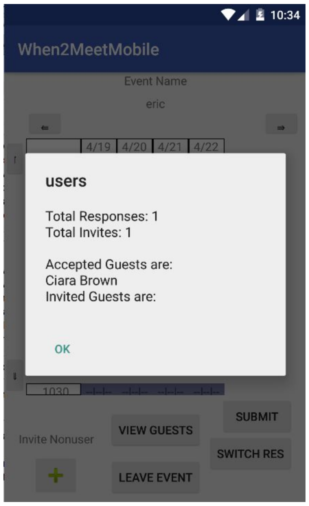
By clicking the view guests button, a popup appears which shows both the users that have accepted the invitation to the event and all the guests who have been invited but have yet to accept.

By clicking the leave event button, the user is removed from the event in its entirety. Their responses are deleted and they are no longer in the event. When a user leaves an event, a notification is sent to the event owner.

When the user clicks the invite nonuser button, a popup is brought up, asking the user for an email, and then sending an email to that address inviting the recipient to download the app.



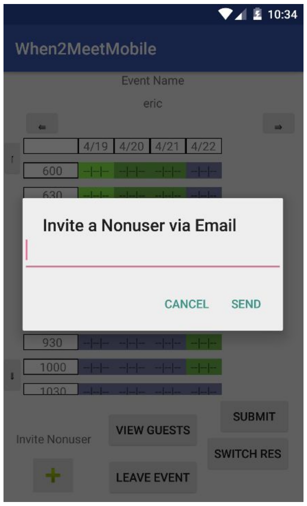




users

Total Responses: 1
Total Invites: 1

Accepted Guests are:
Ciara Brown
Invited Guests are:

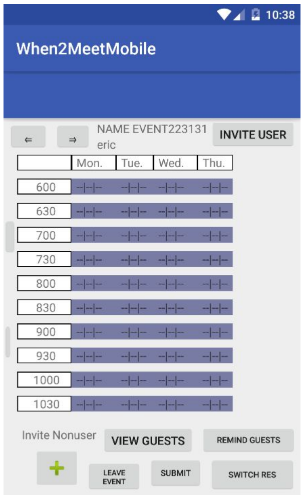


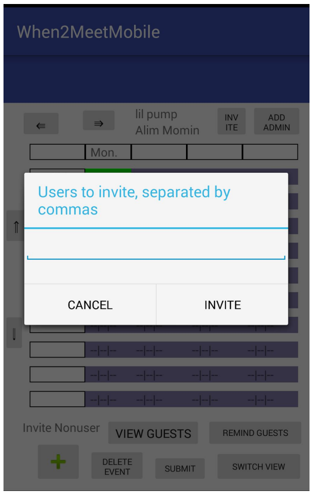
Invite a Nonuser via Email

Admin Event Page

The Admin event page is identical to the Guest event page with the exception of the presence of the Invite user button and the remind guests button. The invite user button occupies the top right of the screen, and so the right arrow button is now on the left of the event name. When the user clicks the invite user button, a popup appears asking for a username. By entering the name of a valid user who has not yet been invited and clicking submit, the user is able to invite another user to that event. If the user wants to invite multiple users, it can be done by separating the usernames with commas. The user can also cancel this operation by hitting cancel.

By clicking the remind guests button, a notification is sent to all guests reminding them to finalize their responses.

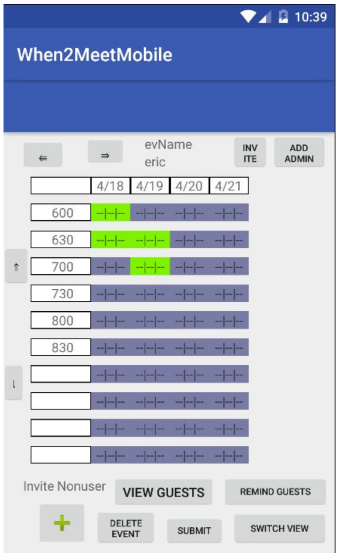


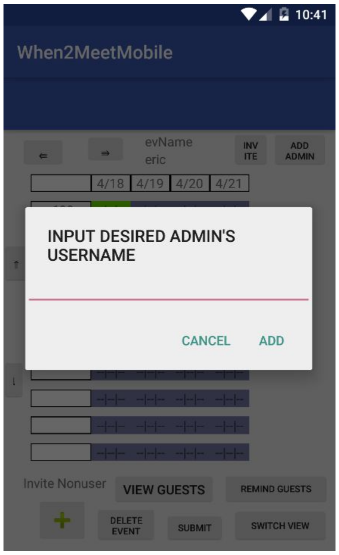


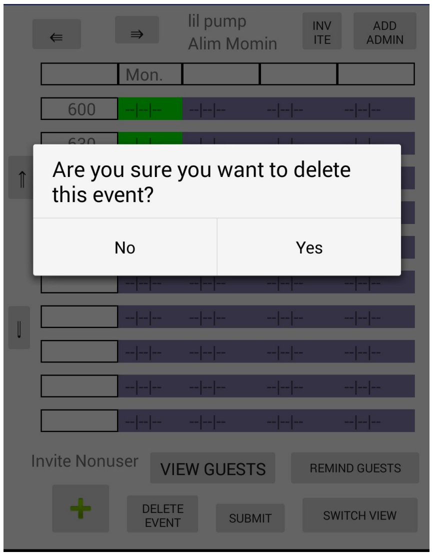
Owner Event Page

The Owner event page is identical to the Admin event page with the exception of the presence of the add admin button and the replacement of the leave event button with the delete event button. The add admin button sits to the left of the invite user button. On clicking it, a popup appears asking the user for a username. By entering the username of the desired user, the owner of that event gives the target user administrative powers in the event. Note that this requires that the desired admin has already accepted an invitation to the event, otherwise the user will be notified that the target action cannot occur. This action may be canceled with the cancel button.

Clicking the delete event button brings up a popup asking if the user is sure that they want to delete the event. By clicking yes, the event is deleted in its entirety and a notification is sent to all participants that the event has been deleted. Clicking no removes the popup and has no other effect.



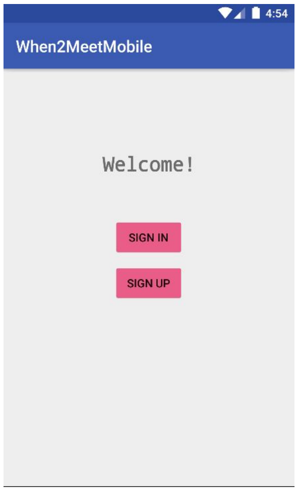

INPUT DESIRED ADMIN'S USERNAME


Are you sure you want to delete this event?

TECHNICAL DOCUMENTATION

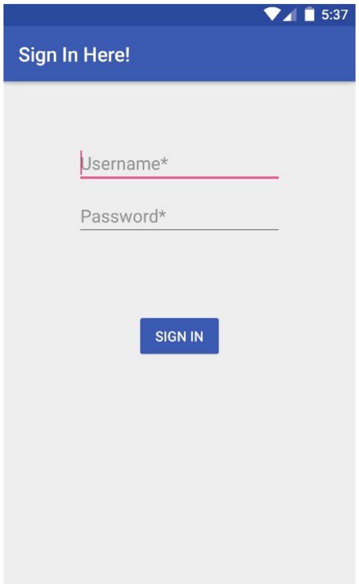
Launch Screen

The launch screen uses the LaunchScreen.java class. It allows for the app to switch intents to the sign in and sign up screens based on which button the user clicks. It will appear if the user of the app is not logged in.



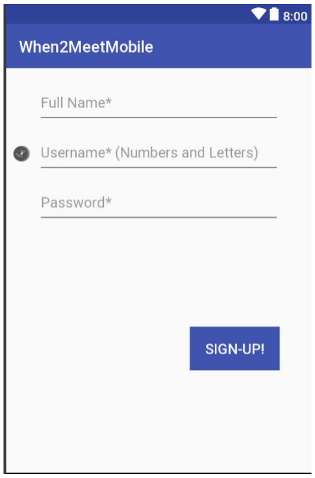
Sign In Page

The sign in page uses the SignInActivity.java class. It allows for the app to accept a user's input username and password. If the username and password combination is correct, the app will log in the user and direct them to the main menu.



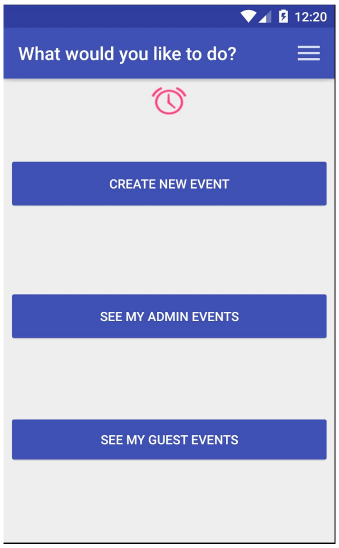
Sign Up Page

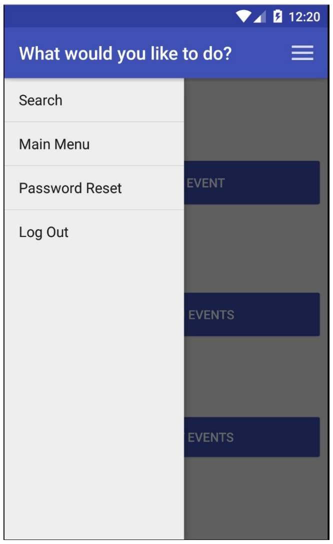
The Sign Up page is associated with SignUp.java. SignUp.java takes in the user-specified name, password, and username and creates an account for the user based on that information. If any of the parameters are invalid, SignUp.java handles how to notify the user (i.e. if the username is taken, the box is outlined in red).



Main Menu

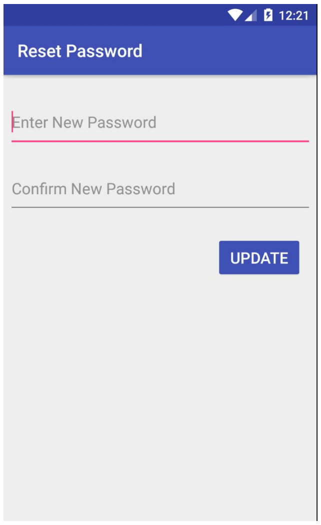
The Main Menu page is associated with MainMenu.java. MainMenu.java defines what each button on the Main Menu should do when clicked on. MainMenu.java also fills the hamburger menu.





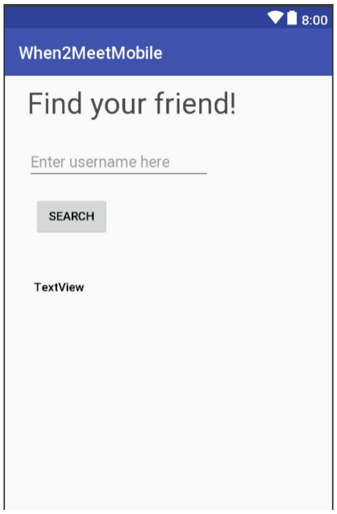
Reset Password

The Reset Password page is associated with ResetPassword.java. ResetPassword handles the post request to update the user’s password with the inputted string.



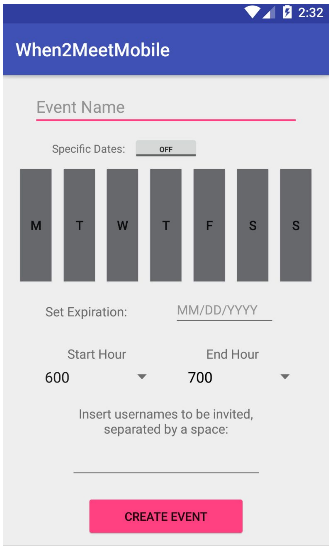
Search

The Search page is associated with SearchActivity.java. SearchActivity.java handles the request that detects whether the user-specified username is associated with an account or not.



Create Event Page

The Create Event page uses the classes CreateEvent.java and DateRangePicker.java.
The CreateEvent.java class defines, sets, and makes visible the event creation attributes: event name, expiration date, specific dates toggle, start/end hour, and invited users.



Event Name

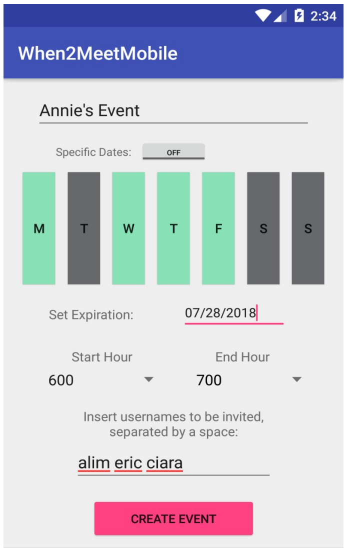
Specific Dates: OFF

Set Expiration: MM/DD/YYYY

Start Hour 600

End Hour 700

Insert usernames to be invited, separated by a space:



Annie's Event

Specific Dates: OFF

Set Expiration: 07/28/2018

Start Hour 600

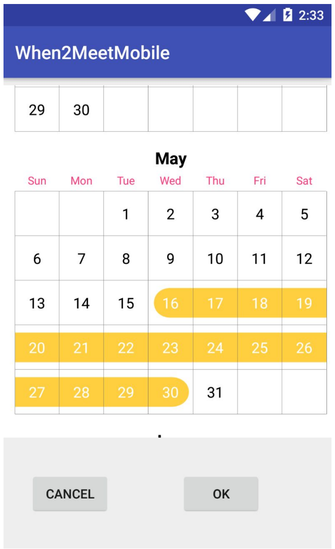
End Hour 700

Insert usernames to be invited, separated by a space:

alim eric ciara

Date Range Picker Page

The DateRangePicker.java class allows the user to pick a range of dates from a calendar view for his/her event. The class will store the start and end date of the range and prepare the intent with this information, when the user returns back to the Create Event page. This class makes use of the following github repo: https://github.com/savvisingh/DateRangePicker



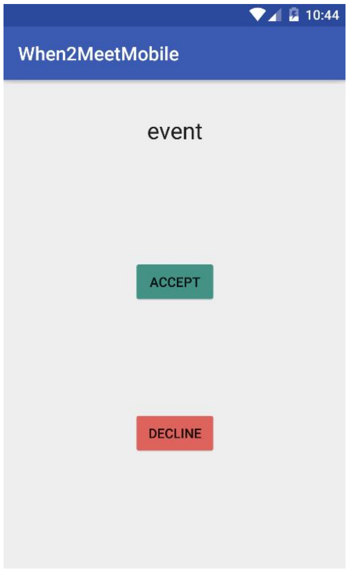
29	30					

May

Sun	Mon	Tue	Wed	Thu	Fri	Sat
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

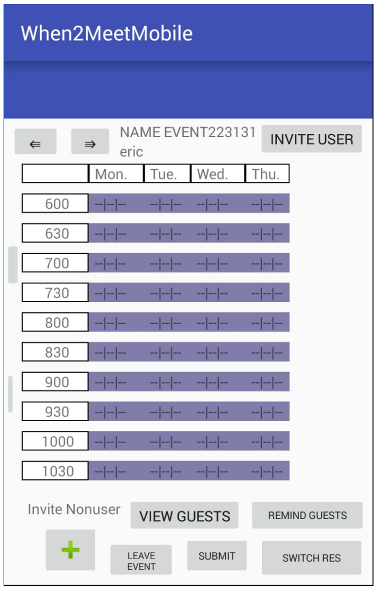
Invite Page

The Invite Page uses the class InvitePage.java. This class defines the on-click behavior for the accept/reject buttons. The accept button on-click redirects the user to the specific event page associated with the invite, and the reject button redirects the user to the Main Menu.

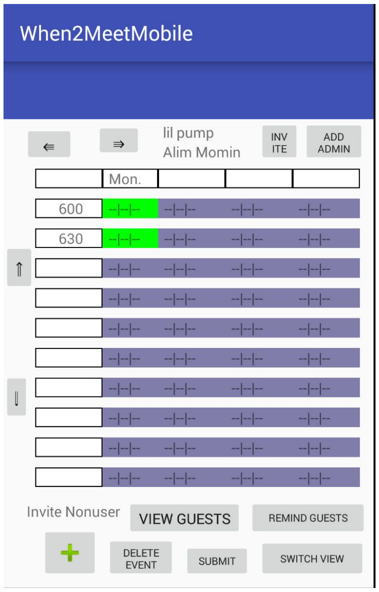


Event Page

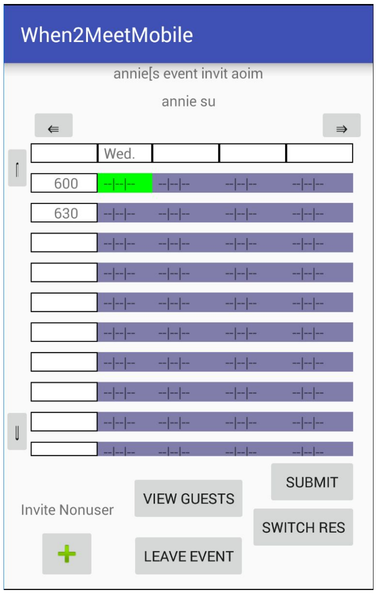
EventPage.java is associated with the admin page, owner page, and guest page of an event. It is responsible for all functionality associated with the buttons on the pages, as well as taking in, processing, and submitting the data for responses. This primarily entials populating and then submitting the data on the calendar in the center of the screen.



Admin Page



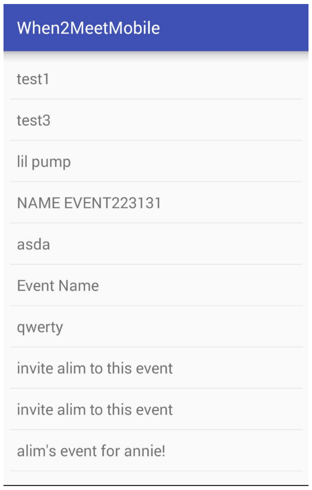
Owner Page



Guest Page

See My Events

SeeMyEvents.java is associated with the see my admin events page and see my guests events page. It is responsible for retrieving the list of events from the database, populating the list, and linking each entry with the event id to which the user will be directed upon clicking the link.



When2MeetMobile

test1

test3

lil pump

NAME EVENT223131

asda

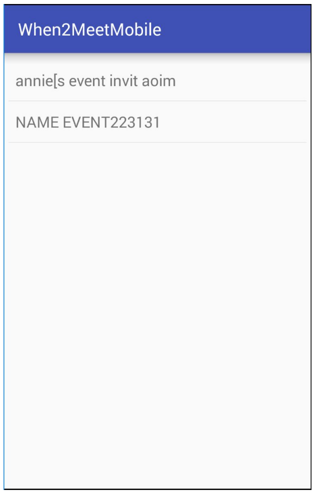
Event Name

qwerty

invite alim to this event

invite alim to this event

alim's event for annie!



When2MeetMobile

annie[s event invit aoim

NAME EVENT223131

Event Schema

The event scheme is defined in eventSchema.js. The event model is the db object associated with created events. This schema stores the following fields: name of the event, owner (person who created the event, can invite guests and delete the event), admins (who have faculty to invite guests, but cannot delete the event), identifier (a unique integer-valued identifier is associated with each event in the database), the specificDates boolean value, date range, time range, expiration date, a list of guests who have accepted the event invites, an array of guests who have been invited to the event and not yet RSVP’d, an array of guests who have rejected the event invite, and an array of accepted guest responses which store their time availabilities that they selected on the availability poll.

User Schema

The user schema is defined in userSchema.js. The user model is the db object associated with a user with a When2Meet account. The user schema stores the name, username, and password of the user. It also stores arrays of event IDs for the different events the user is in. The user schema has an ownedEvents field, which stores the event IDs for the events that the user owns. The schema has a participantEvents field, which stores the event IDs for the events the user was invited to and has accepted. There is also an invitedEvents field for events that the user has been invited to but has not accepted or rejected yet. The schema also stores messages from an event sent to that user in an array.

Index.js

The file index.js serves as the server for the app. It handles all HTTP requests that come from the app, responding with appropriate information based on the state of the application and the database. The route for ‘/inviteNonUser’ uses the external library nodemailer, which allows the app to send emails from the account when2meetmobile@gmail.com. The method has the password saved under the field ‘pass’ in the auth object. The password to the email account is cis350g7.

DBDriver.js

DBDriver.js contains and exports the DBDriver class, as well as connecting to the database. The DBDriver class is responsible for all interactions with the MongoDB database and the app’s backend logic. This includes all management of users and events, done via interactions with the user schema and event schema. The class is as asynchronous as possible to make it as efficient as possible.