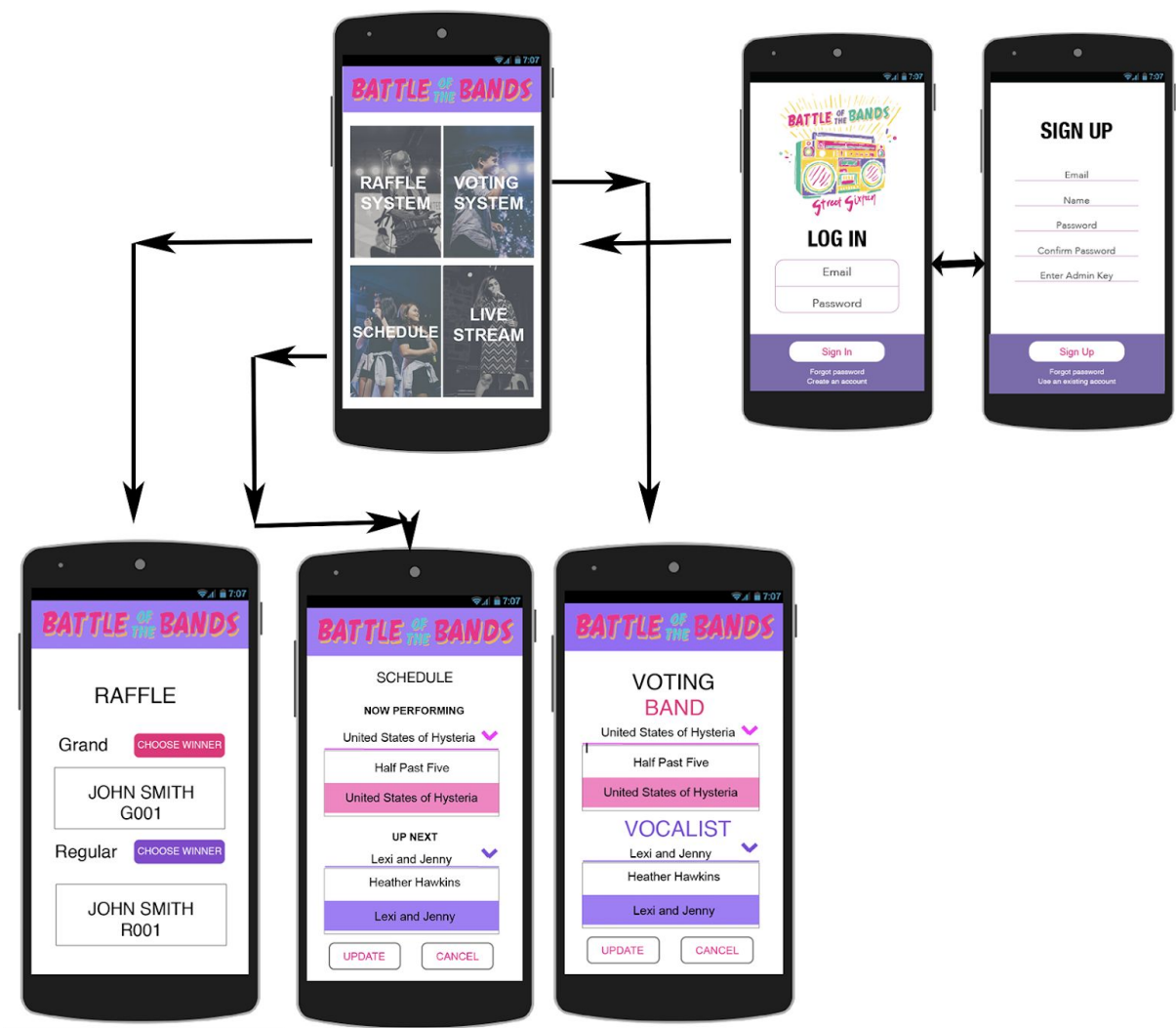


Criterion B: Design

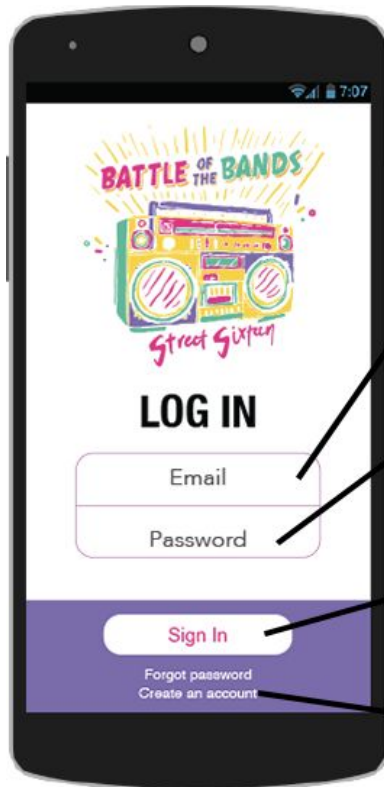
DESIGN PROTOTYPE:



Prototype Colour Codes:	PINK COLOURS: TEXT: #FE0089 BUTTONS: #FA3AB2 DROP DOWN: #F873bb	PURPLE COLOURS: TEXT: #7D17B7 BUTTONS: #8B47B2 DROP DOWN: #a479fb	CYAN COLOURS TEXT: #04E4AD
All current and future designs will be made using Adobe Photoshop			

## Classes

### LOGIN ACTIVITY

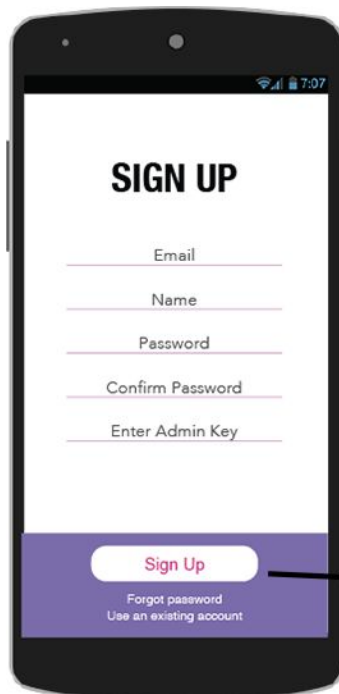


email will be taken as a string to be converted into JSON object

Password will be taken as string and converted to JSON object. Using android's built-in XML, the password will be hidden when the user types his/her password

once button is clicked, the email and password is checked with the database; if access is correct, the window will change to the main menu

Once the sign up text is clicked, the window will change to the signup window.

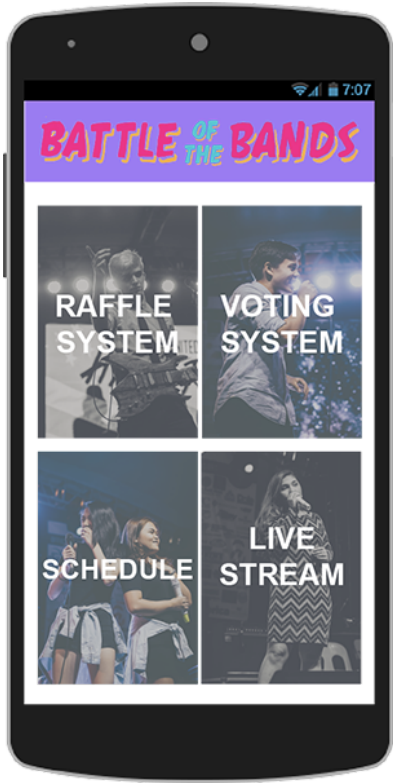


all fields will received user input in string format which will be converted into a JSON array to be sent to the database for insertion

This variable will determine whether the user can log in to the main page or not

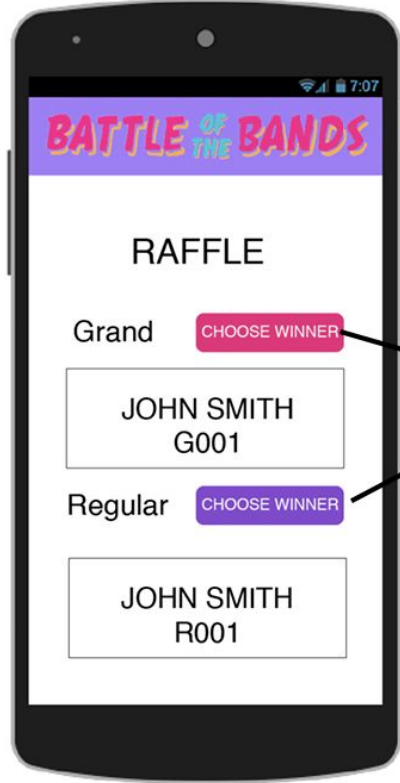
Once this button is pressed, the user is taken back to the login page

## MAIN ACTIVITY

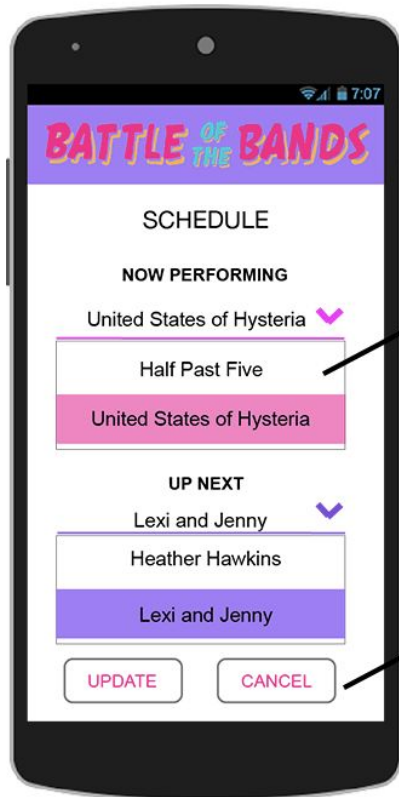


All photographs are buttons, which, when clicked, will change layouts to its respective functions.

\*\*\* Live stream was not included in the success criteria but it is just a link to the live stream of our school.

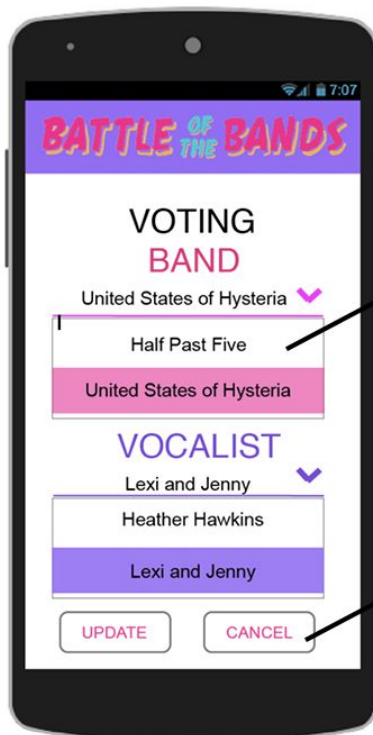


This will get an HTTP request to bottle, which will generate a random number, and check with the database. The database will return the name of the user to be featured on the boxes



An android widget (spinner) will be used to create a drop down menu. This will list the performers from the database

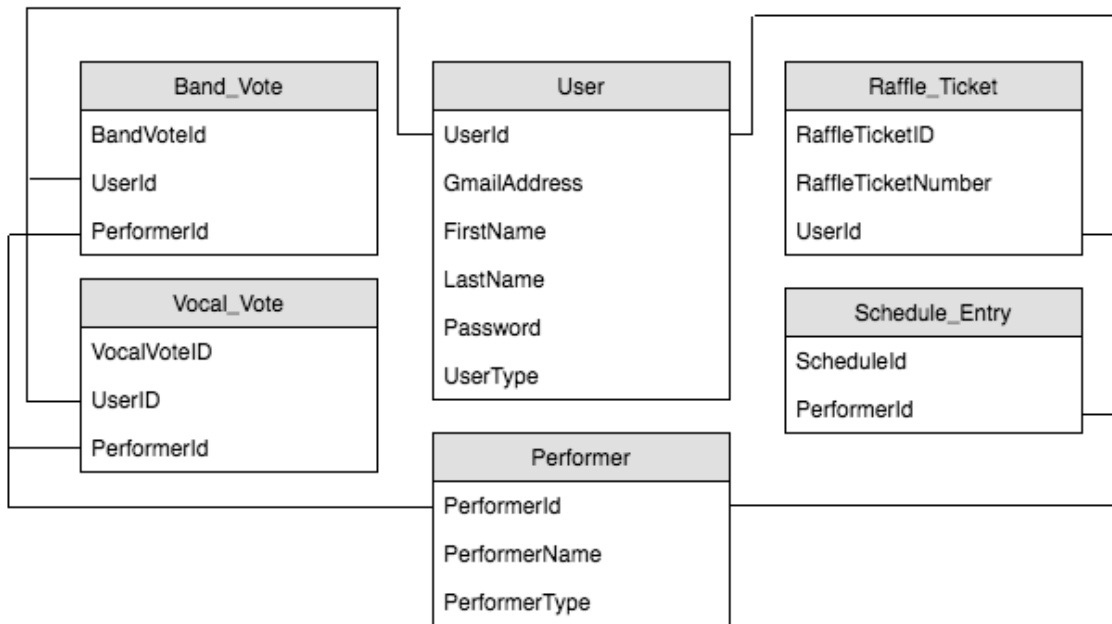
The update button will set the chosen performers to be playing and up next



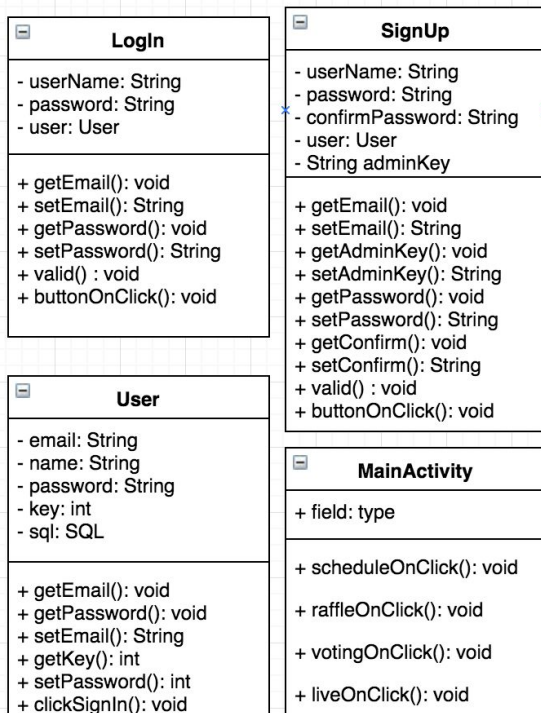
An android widget (spinner) will be used to create a drop down menu. This will list the performers from the database

The update button will set the chosen performers to be the vote of admin user

## Entity Diagram



## UML diagram:



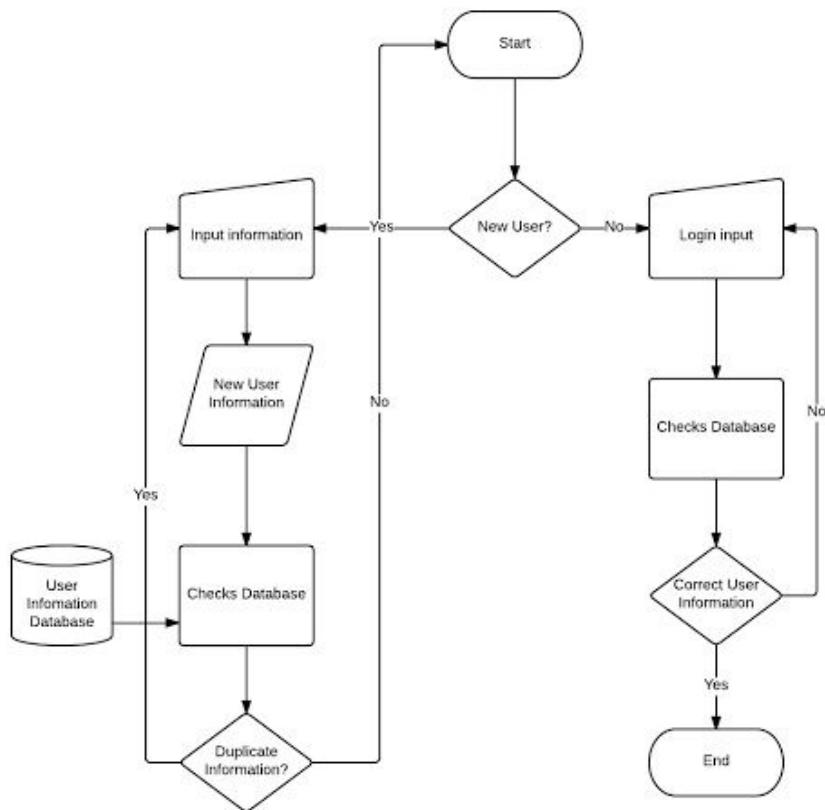
File Format:

I will be using JSON (javascript object notation) file format to parse data from the client (android) to the server (postgresql) and vice versa.

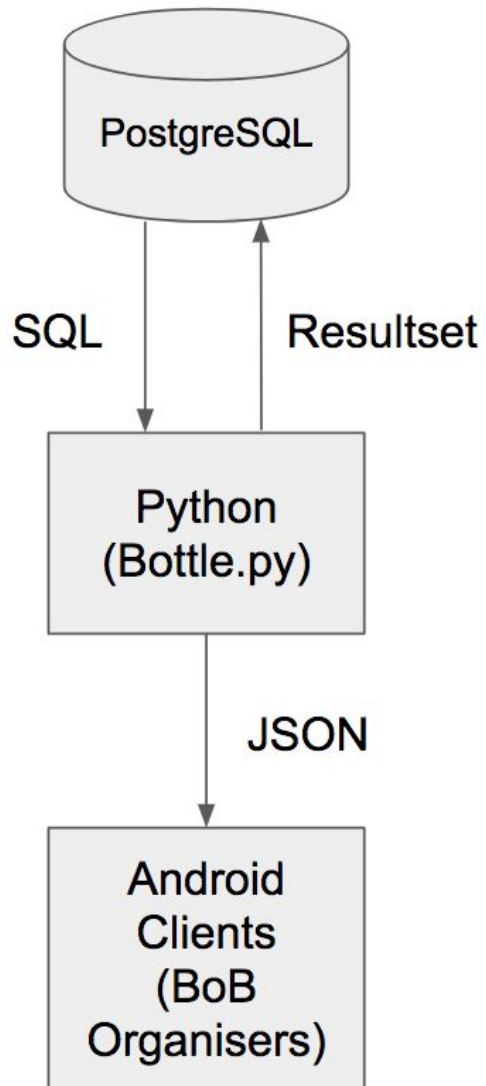
**Flowcharts:**

**Login Activity:**

This checks whether the user is a new user or an existing user and also validates the information with the database.

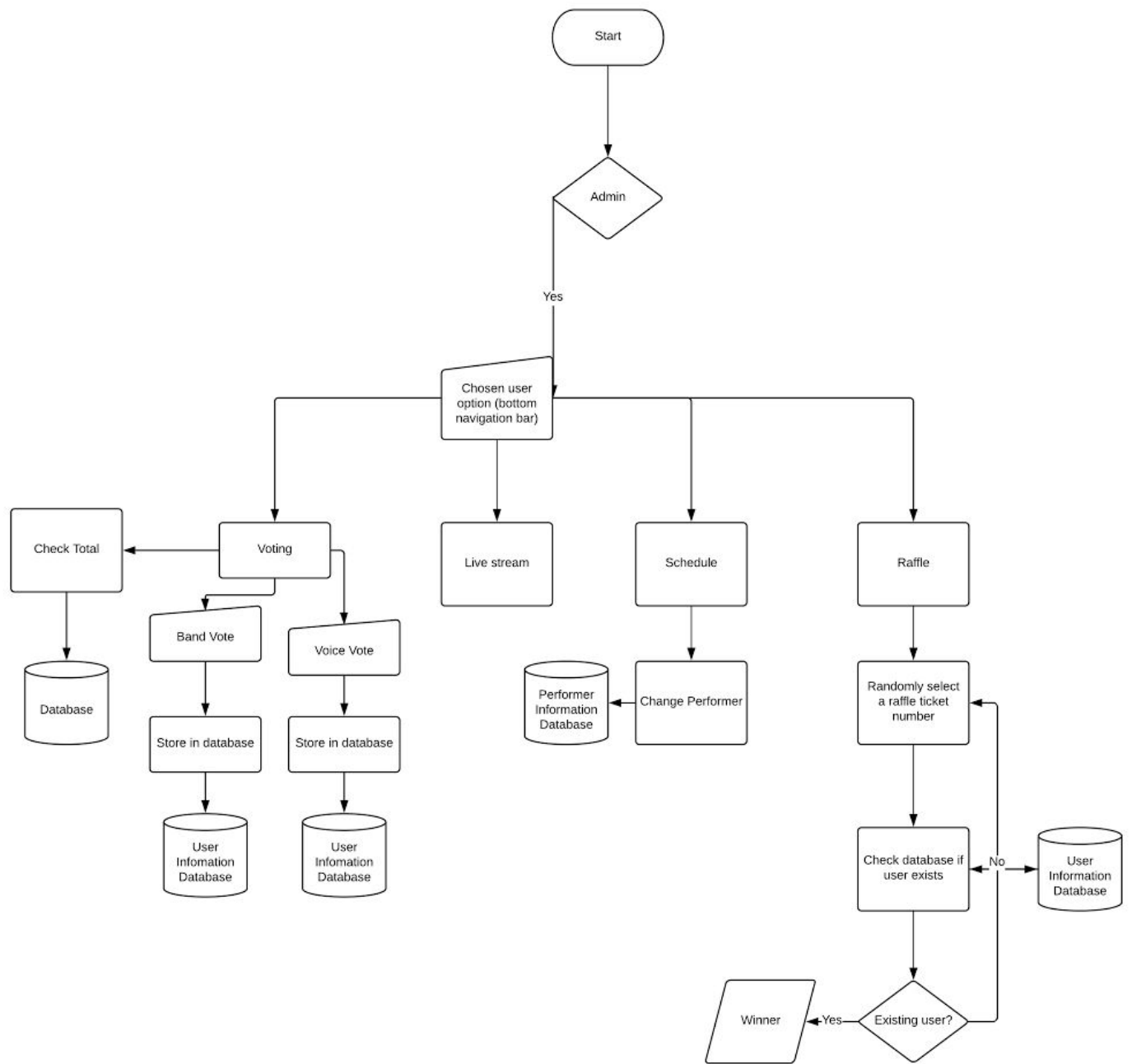


**Database:** This diagram shows the interaction between the android clients and the database



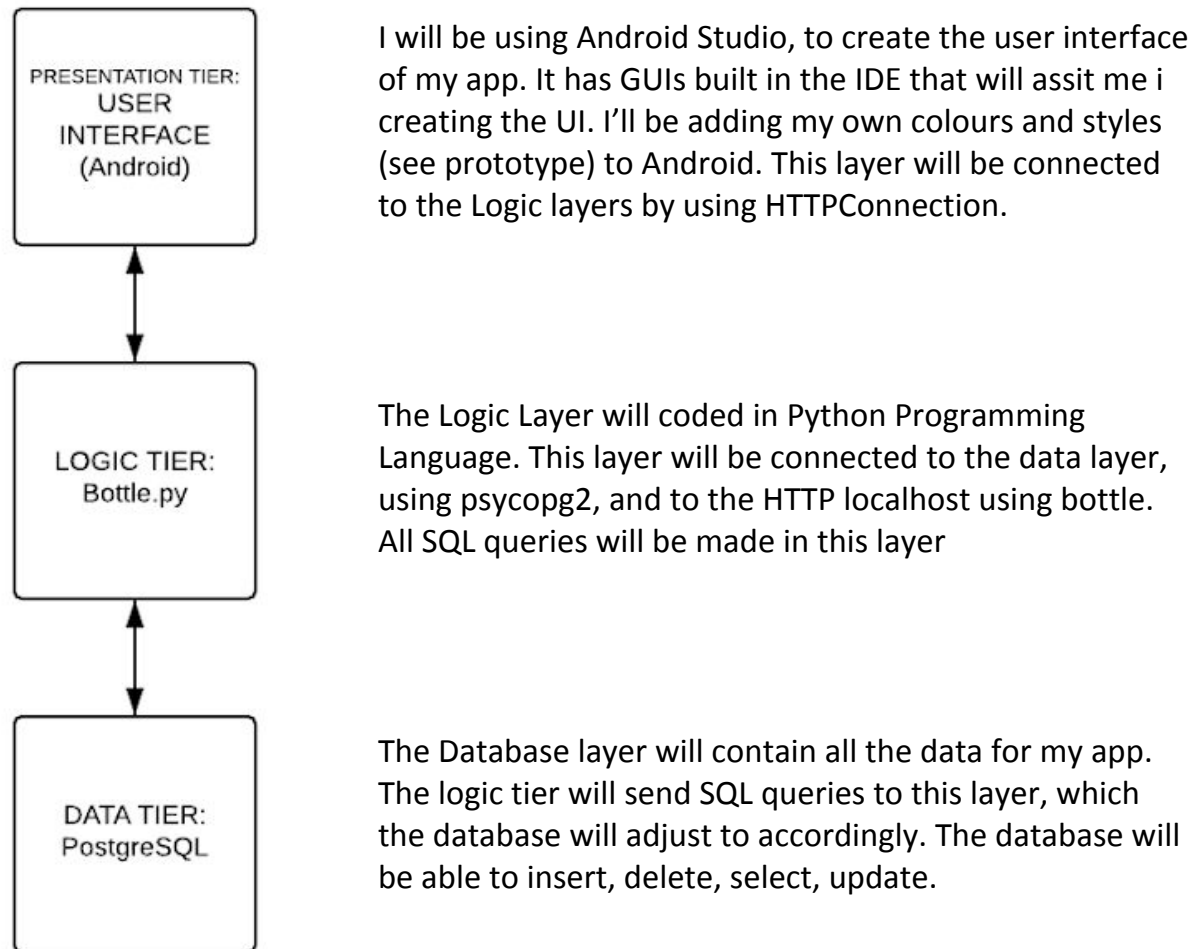
#### **Main Activity**

**Admin:** This shows the user functions of admin users.





## Database Architecture:



## Testing Plan:

	Success criterion	Testing
1	<b>User signup:</b> Users are able to sign up as either an admin or an attendee. Admins are given full access to the app.	When program opens, user login page is shown. Users must be able to click the signup button and be redirected to a new window where they can sign up.
2	<b>User login:</b> When the app is launched, a user login page will appear. Users can login with	App must be able to authenticate users by verifying the email and password input. The app must say whether the inputted email and password is incorrect.

	their email and password as an administrator. The data is checked with the database	
3	<b>Users can switch between windows:</b> Users should be able to use a Navigation Bar to switch between raffle, voting, and schedule	After logging in, check each bottom navigation button if each button corresponds with a different fragment and can be done more than once.
4	<b>Users can update schedule information:</b> By clicking a button to send a request to the database that changes the schedule information	Check update button to ensure information is fetched from the database.
5	<b>App can select a winner:</b> Administrators can press a button that randomly selects a winner for the raffle.	Check bottom navigation (clicking raffle icon) leading to raffle fragment. Check if app shows user information. Click choose winner button. Check if winner is selected from the database.
6	<b>Users can have raffle numbers:</b> Users will have their raffle numbers linked to their user profile on the database	The admins must be able to click a button, which changes the winner for both Grand Raffle and Regular Raffle winners.
7	<b>Users can vote:</b> Admins can vote for their favourite performers by choosing from a drop down menu.	Check bottom navigation (clicking vote icon) leading voting fragment. Click dropdown menu for band and vocalist voting. Press the submit button. Check database for recorded vote.

Extensibility:

Programming Style	Explanation
Proper Use of Variable Names	The variables used in both python and java must be appropriate to its function as a variable
Proper Use of Capitalisation	The code must use camel casing for variables and classes
Comments	The code should be easily understood by anyone by reading the comments that explain the function of most lines of code.
Component	Explanation
Encapsulation	Variables will be declared private and will be accessed using the accessor methods "get" and "set"

File input/output	Instead of hard-coding, developers can change data
PostgreSQL	Developers can easily use postgresQL because it is a standard industry database