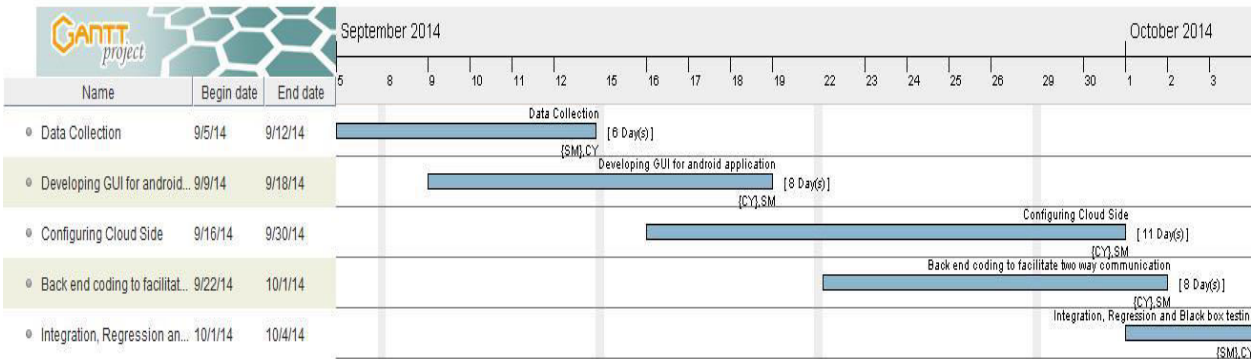
Cloud assisted android application for ASU students

Chaitanya Yaddanapudi Sarvani Mallajosyula

Tasks and timelines mentioned as per the proposal (by midterm submission):

Sno.	Task	Status	Date of Completion
1	Data Collection	Completed	09-13-2014
2	Developing GUI	Completed	09-19-2014
3	Cloud Side	On Track	09-30-2014

Gantt chart as per the proposal:



Task 1 and Task 3: Data Collection & Cloud Side

First task is to collect data related to bus timings and the bus routes. We will obtain this information from Orbit and Valley Metro bus service and update it in our database.

Summary:

As mentioned in the first task we have successfully completed the task of data collection. We used the Orbit schedule of bus arrivals and departures from different bus stations. Collected the details of Valley Metro buses similarly and updated these entries in the form of tables in the data base. Now our data base is ready to retrieve the information from the data base.

Steps followed to set up Database:

1. Using Ubuntu 14.04 Virtual Machine through ASU Vlab. Installed MySQL, PHP, Apache2 collectively known as LAMP software on Ubuntu via the terminal.

2. Tables have been created for each bus route. A detailed list of bus routes have been given in the flow chart given below.

From figure 1 given below, we can see that the Buses can be classified as ORBIT (free bus service) and Valley Metro (paid Bus service). Each Bus type has various routes.

For example, ORBIT-EARTH has two bus routes, EAST bound and WEST bound. They operate at different timings in both the directions.

Two separate tables have been created for EARTH east and EARTH west.

i.e. Table - Eartheast; Earthwest

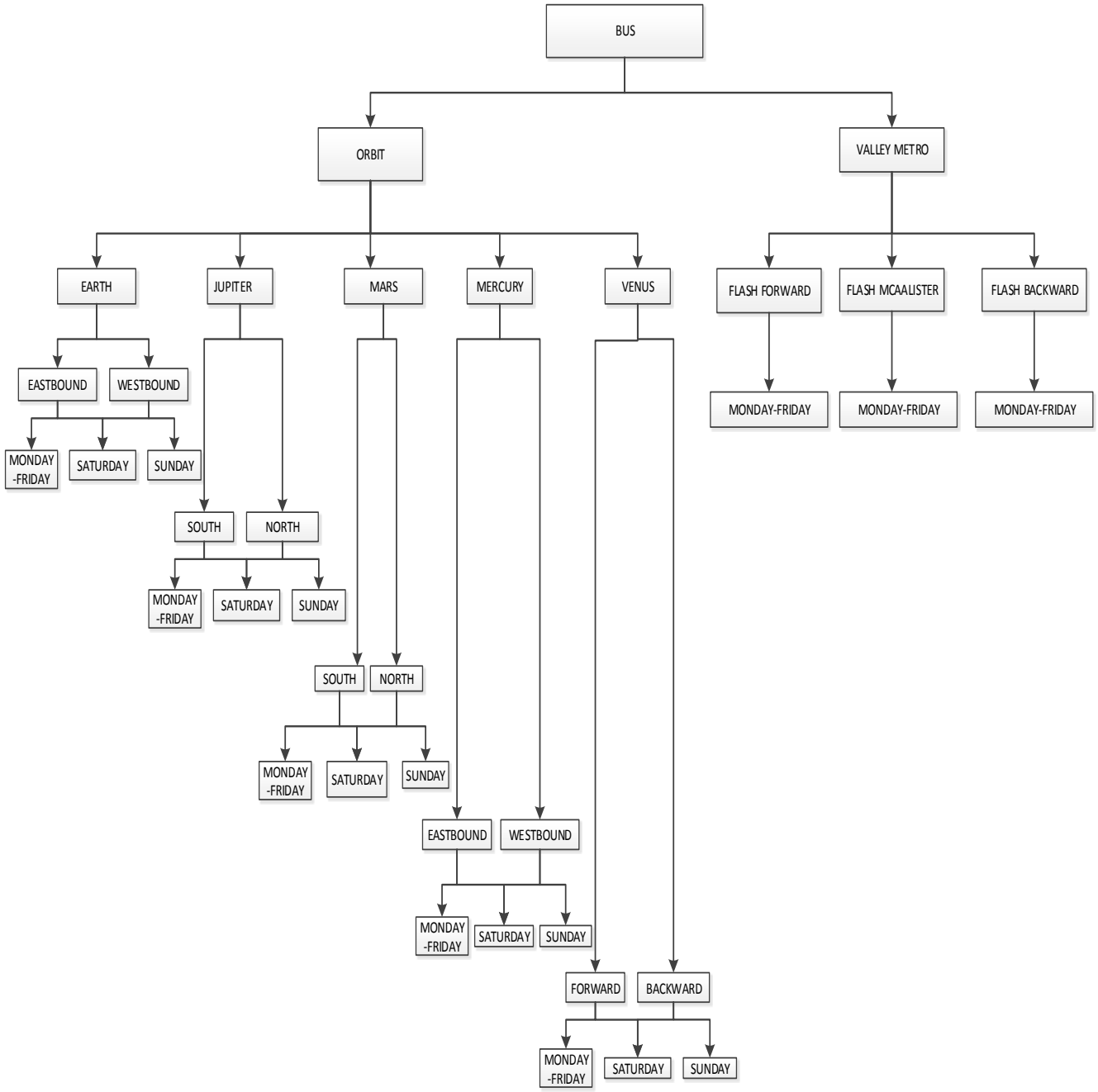
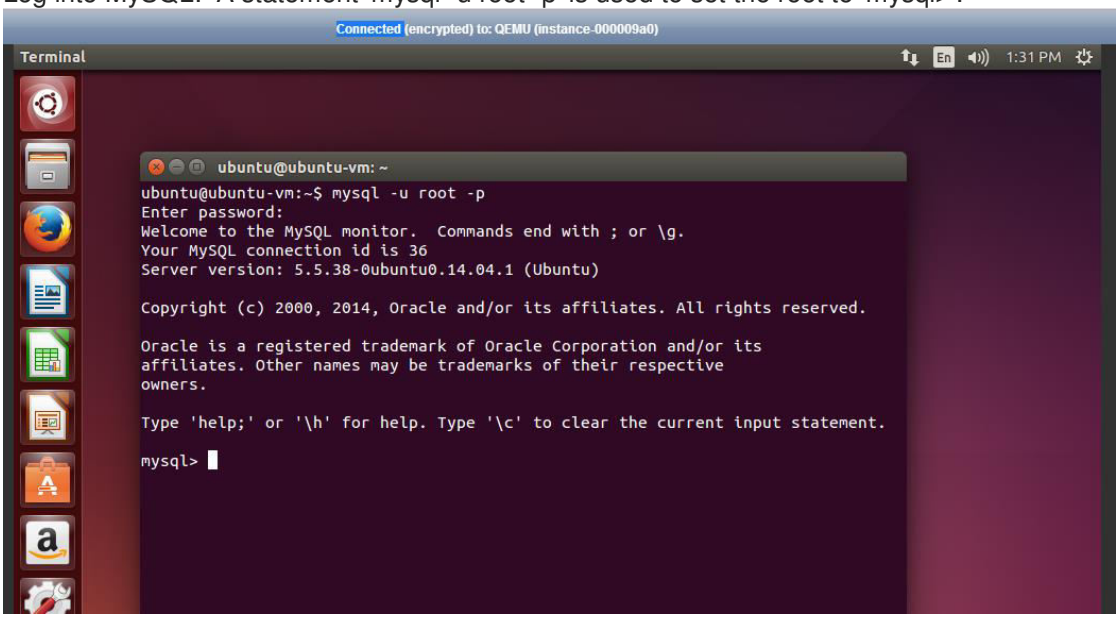


Figure 1: In depth bus routes of Orbit and Valley Metro.

Steps to create, check and configure database tables in MySQL:

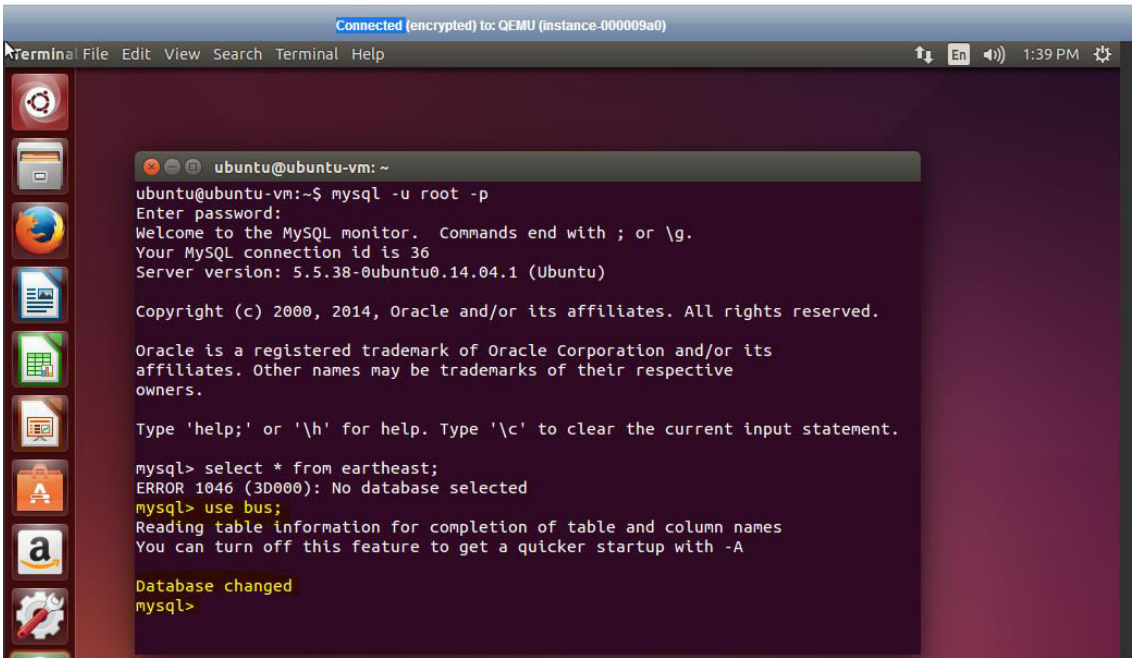
1. Log into MySQL: A statement 'mysql -u root -p' is used to set the root to 'mysql>'.



2. Creation of Database: Database by name 'BUS' has been created.

   Syntax: CREATE DATABASE Bus;

3. Check if the Database has been created:
   Syntax: USE Bus;
   Can be used to switch to Database.



Tables:

EARTHEAST
EARTHWEST
JUPITERSOUTH
JUPITERNORTH
MARSSOUTH
MARSNORTH
MERCURYEAST
MERCURYWEST
VENUSFORWARD
VENUSBACK

Figure 2: List of Tables

4. Creation of Table: All the tables mentioned in Figure2 have been create in Ubuntu with respective columns.

5. Inserting values in tables: Bus timings have been inserted into corresponding tables. Reference: http://routes.valleymetro.org/timetables/6/route_list .

   Syntax: INSERT INTO table_name (columns) VALUES (values);

6. Checking the values in a table: Select * from table_name can be used to get all the rows in a particular table.

Example of EARTHEAST table.

TTC	C&C	S&C	NTC	TM
06:58 AM	06:06 AM	06:16 AM	06:20 AM	06:33 AM
06:13 AM	06:21 AM	06:31 AM	06:35 AM	06:48 AM
06:28 AM	06:36 AM	06:46 AM	06:50 AM	07:03 AM
06:43 AM	06:51 AM	07:01 AM	07:05 AM	07:18 AM

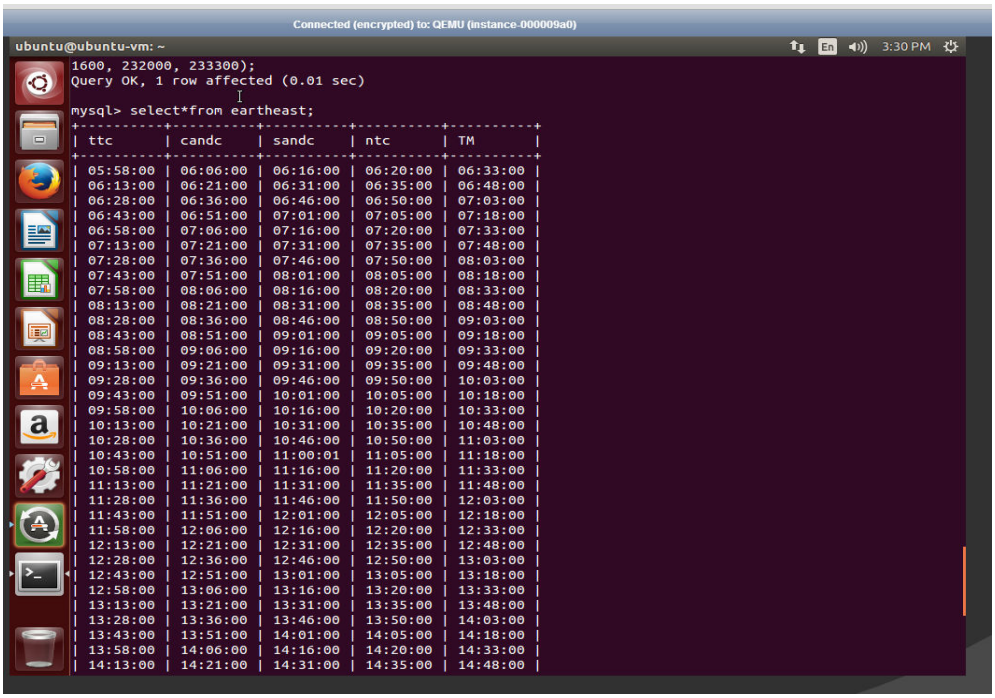
Figure 3: Table – EARTHEAST with four rows as an example.

PHP is used by Android SDK to fetch the required data from MySQL in the VM on cloud (Ubuntu in VLAB).

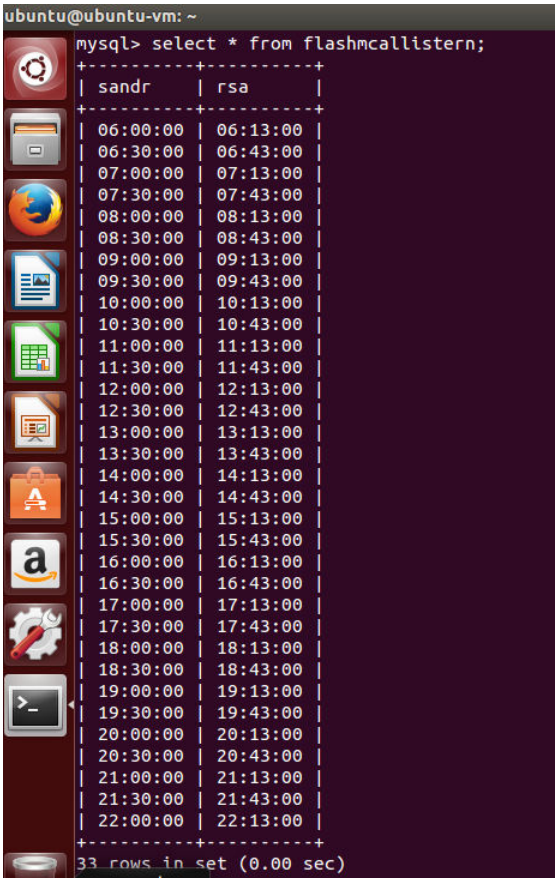
When user makes selection on Android application, a JASON class will be class which in turn calls the PHP on the cloud side and the required and relevant data is fetched from MySQL.

Examples of tables created in MySQL:

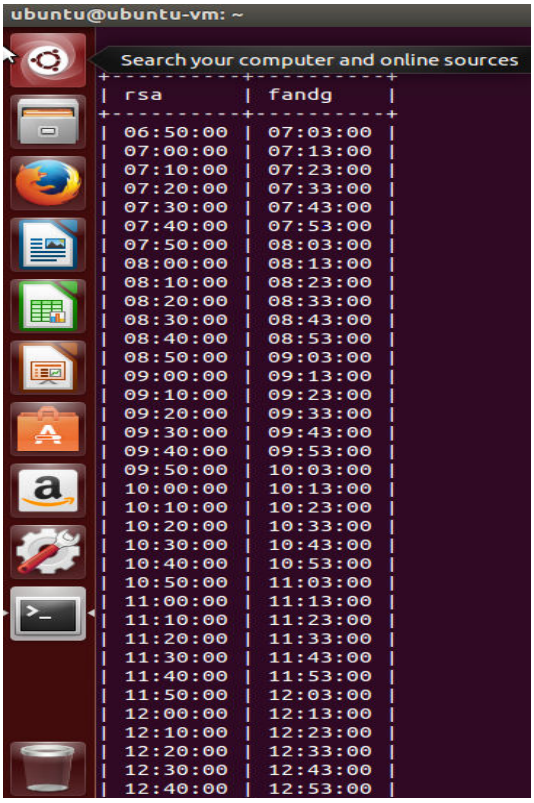
EarthEast:



Mcallister:

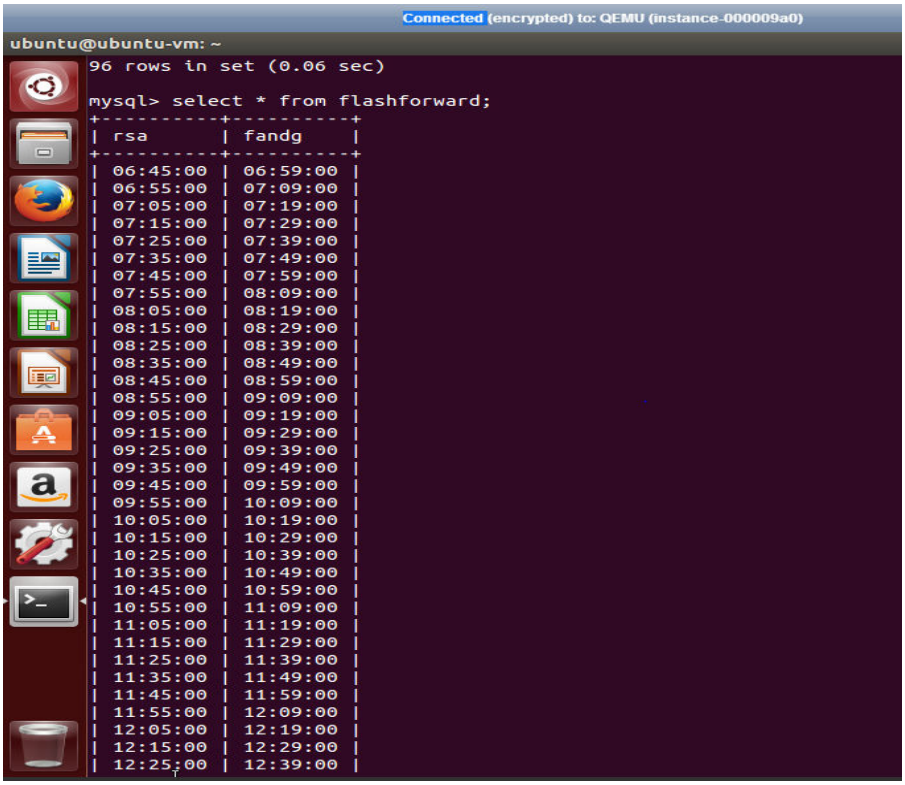


Flashback:



rsa	fandg
06:50:00	07:03:00
07:00:00	07:13:00
07:10:00	07:23:00
07:20:00	07:33:00
07:30:00	07:43:00
07:40:00	07:53:00
07:50:00	08:03:00
08:00:00	08:13:00
08:10:00	08:23:00
08:20:00	08:33:00
08:30:00	08:43:00
08:40:00	08:53:00
08:50:00	09:03:00
09:00:00	09:13:00
09:10:00	09:23:00
09:20:00	09:33:00
09:30:00	09:43:00
09:40:00	09:53:00
09:50:00	10:03:00
10:00:00	10:13:00
10:10:00	10:23:00
10:20:00	10:33:00
10:30:00	10:43:00
10:40:00	10:53:00
10:50:00	11:03:00
11:00:00	11:13:00
11:10:00	11:23:00
11:20:00	11:33:00
11:30:00	11:43:00
11:40:00	11:53:00
11:50:00	12:03:00
12:00:00	12:13:00
12:10:00	12:23:00
12:20:00	12:33:00
12:30:00	12:43:00
12:40:00	12:53:00

Flashforward:



96 rows in set (0.06 sec)

```
mysql> select * from flashforward;
```

rsa	fandg
06:45:00	06:59:00
06:55:00	07:09:00
07:05:00	07:19:00
07:15:00	07:29:00
07:25:00	07:39:00
07:35:00	07:49:00
07:45:00	07:59:00
07:55:00	08:09:00
08:05:00	08:19:00
08:15:00	08:29:00
08:25:00	08:39:00
08:35:00	08:49:00
08:45:00	08:59:00
08:55:00	09:09:00
09:05:00	09:19:00
09:15:00	09:29:00
09:25:00	09:39:00
09:35:00	09:49:00
09:45:00	09:59:00
09:55:00	10:09:00
10:05:00	10:19:00
10:15:00	10:29:00
10:25:00	10:39:00
10:35:00	10:49:00
10:45:00	10:59:00
10:55:00	11:09:00
11:05:00	11:19:00
11:15:00	11:29:00
11:25:00	11:39:00
11:35:00	11:49:00
11:45:00	11:59:00
11:55:00	12:09:00
12:05:00	12:19:00
12:15:00	12:29:00
12:25:00	12:39:00

Task 2: Developing GUI for the android application

For user to use the android application we need a GUI. This GUI will have two options, books and buses. If the user selects the buses option information related to the buses will be displayed. The users can not only view, but also update the information. They can drop a post if the bus is full, so that students waiting in the next stop can find an alternative. The second option in the main menu is books. The students can put the books for sale, rent or exchange. Interested buyers can find their location and contact them.

Summary:

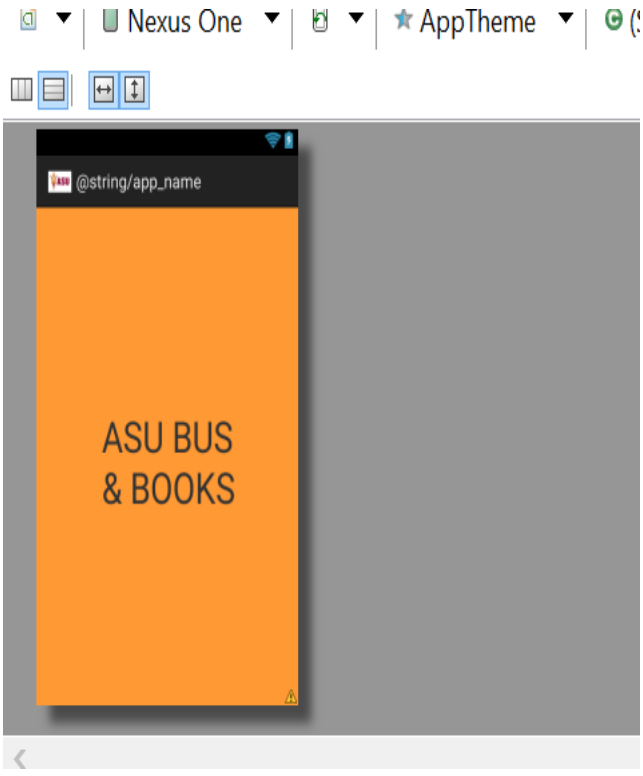
Android Screen Development: screens from book to be created in Visio and in android SDK.

Screen flow is given below. Android Screens layouts are developed with minimal graphics as of now with more emphasis on the functionality (considering smaller time frame and team).

Screen 1 – This is the start-up screen which stays for about 2 seconds before Screen 2 is displayed.

Prototype

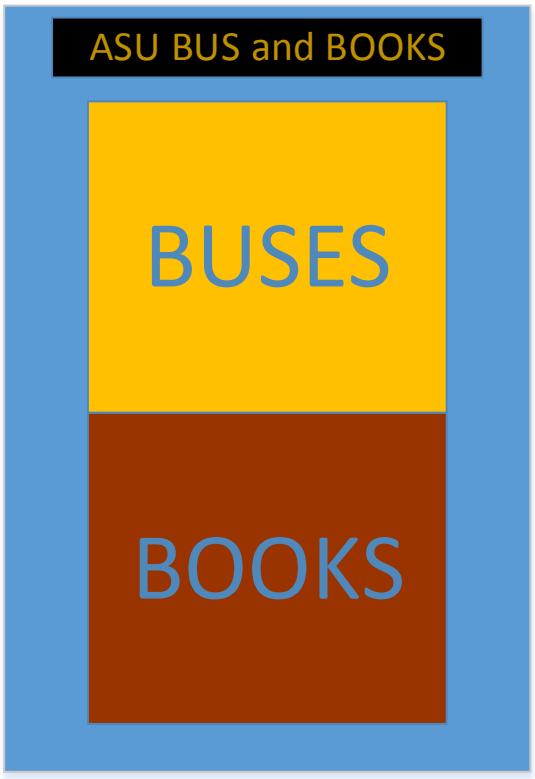
Android Screen – Screenone.xml



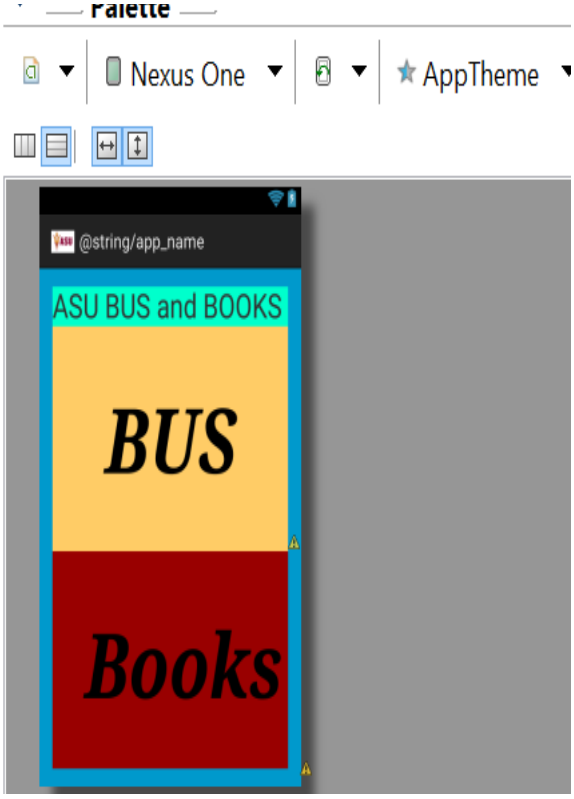
The startup screen is displayed for 1-2 seconds and then the user is given an option to choose between BUS/BOOKS as shown in screen 2.

Screen 2 – User can select one among the two buttons – 'BUSES', 'BOOKS'. Based on the user selection, control moves to Screen 3 or Screen 6.

Prototype



Android Screen-Screentwo.xml

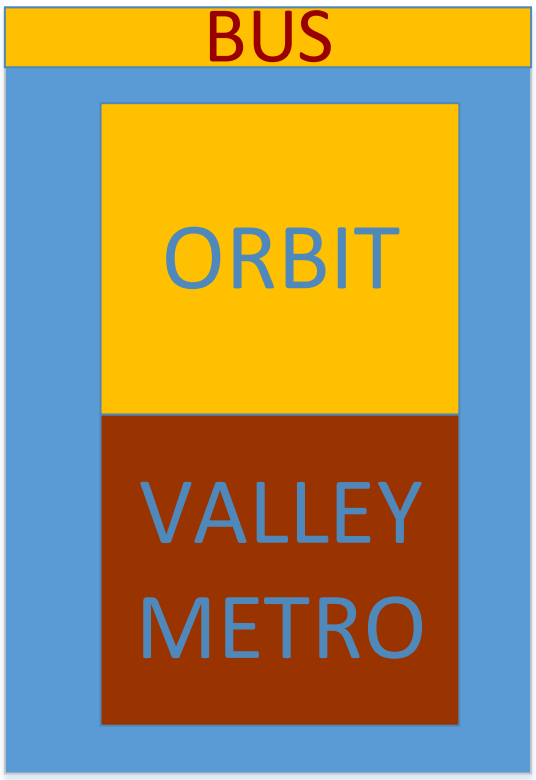


User can choose between BUS or BOOKS by clicking on the respective button. If the user clicks on button – BUS, then the screen 3 pops up where user is given an option to choose the route for which the timing and the current status will be displayed.

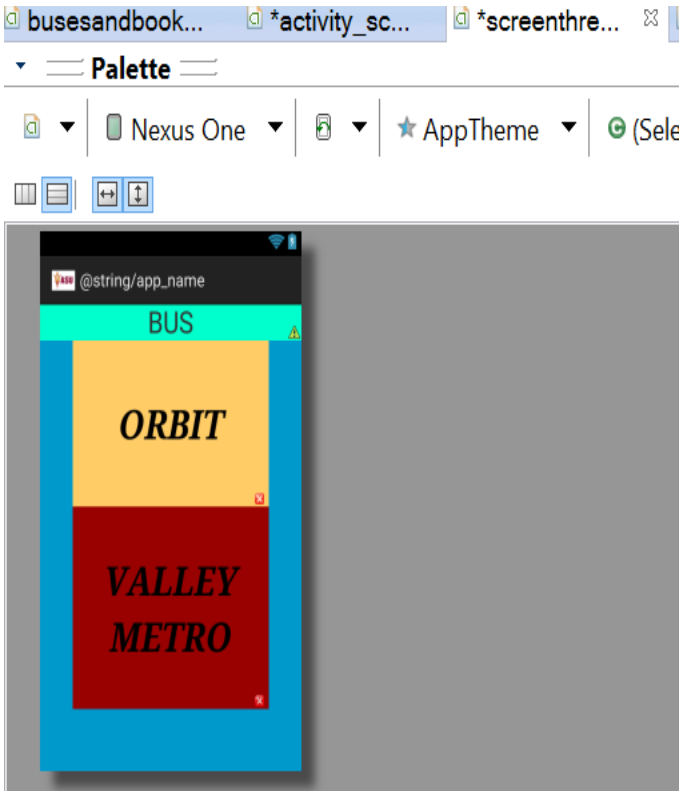
If the user click on button Books, screen 6 pops up where user can post his comments on the details about books or from where he can read the feeds or posts about peers on the sale of books.

Screen 3 – User can select between 2 buttons ORBIT or VALLEY METRO and the requested data of Buses will be displayed.

Prototype



Android screen-Screenthree.xml

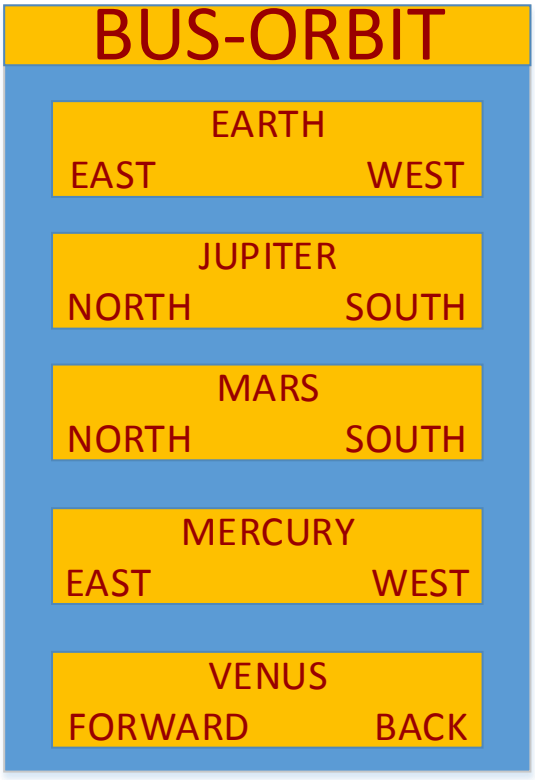


User can select between Orbit service or Valley Metro service. In Tempe, AZ, Orbit service offers free bus service to the people living in Tempe. Valley Metro offers paid bus service.

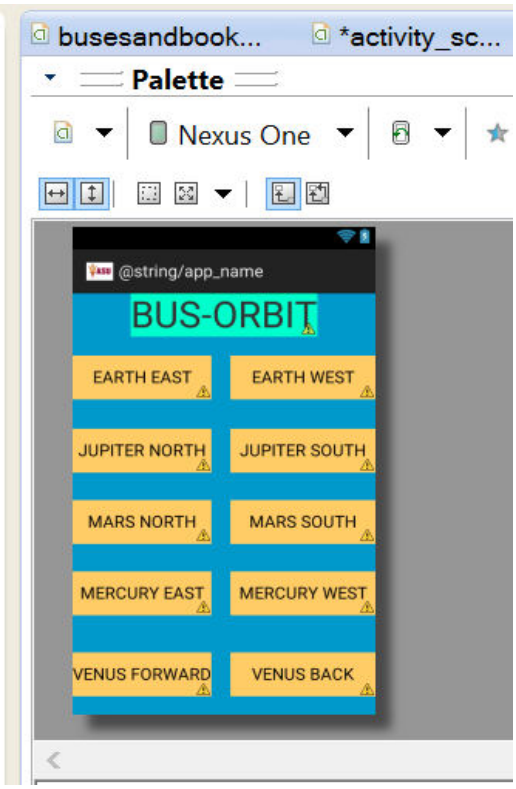
They operate on same and different routes and the details have been captured.

Screen 4 – User selected ORBIT and the list of Buses in ORBIT will be displayed as shown below. Each of the options given acts as buttons and the user can choose between the options given.

Prototype

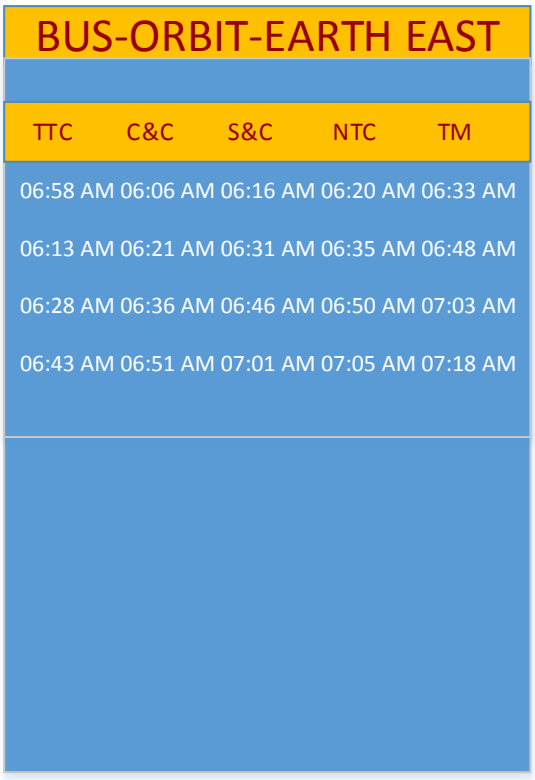


Android Screen-Screenfour.xml



Once the user selects the operator, he has an option to choose the bus route by which he will get details on bus stops, timings and the status of the bus. There are a total of 10 routes as mentioned above for Orbit bus service.

Screen 5 – User has selected EARTH EAST and hence the details of bus timings are displayed as table as shown below.



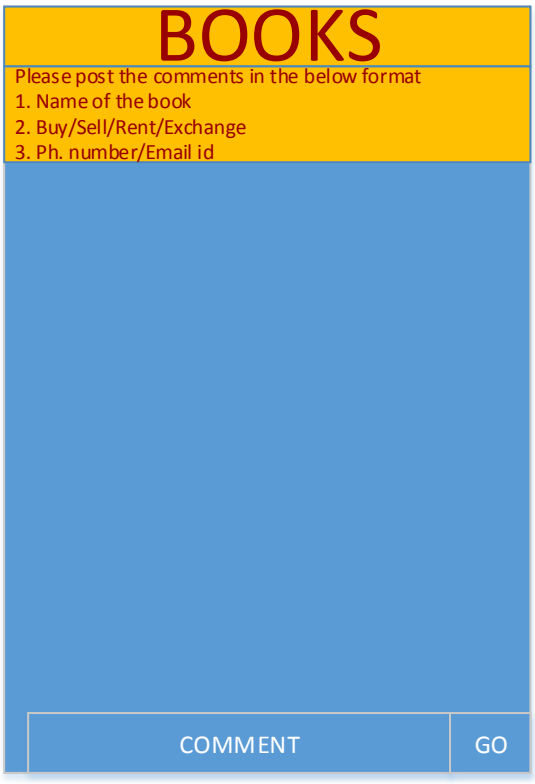
The details fetched from the tables from the cloud platform based on the user input. JASON class is used send the user input from android device to PHP layer on the cloud side. Once the relevant data has been fetched, it is returned back to the Android application.

The data received from the cloud side is displayed as table format shown in the above prototype. Android screen for this prototype will be placed on the back end coding is completed.

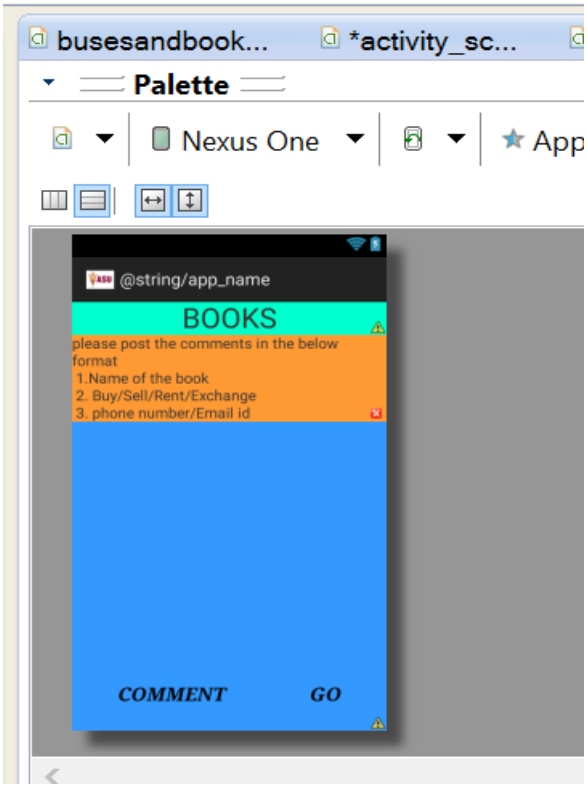
Bas name and route are shown on the top of the application. In this particular example, it is BUS-ORBIT-EARTH EAST. Different stops in this bus route along with the timings are displayed in the other portion of the application.

Screen 6 – If User selects BOOKS option in Screen 2, he/she will be directed to the screen shown below.

Prototype



Android Screen-Screenfive.xml



This acts as a live feed screen where user can see the current listings of books. He/She can post comments using the comments section given at the bottom of the screen. Rules of the comments that can be posted are given on the top of the screen. As part of extension to this project, spam word filtering can be embedded into application to ensure that no spam messages/comments are posted from this screen. 'GO' acts as button.

Tasks to be completed by Final submission:

Sno.	Task	Status	Date of Completion
3	Cloud Side	On Track	09-30-2014
4	Back end coding to facilitate two way communication	To be started	10-02-2014
5	Testing	To be started	10-04-2014

REFERENCES

[1] http://routes.valleymetro.org/timetables/6/route_list

[2] http://developer.android.com/sdk/index.html

[3] http://developer.android.com/training/cloudsync/index.html

[4] http://en.wikipedia.org/wiki/OSGi

[5] http://books.google.com/books?id=XVi0AAAAQBAJ&pg=PT41&lpg=PT41&dq=osgi+framework+config+for+cloud&source=bl&ots=2vDArA8AtT&sig=jvcCWUwUNDLBdHvHPetUs9oJz98&hl=en&sa=X&ei=RcYHVKT4F6jFigKk4ICgBg&ved=0CC8Q6AEwAg#v=onepage&q=osgi%20framework%20config%20for%20cloud&f=false

[6] https://www.secret.ly/

[7] http://en.wikipedia.org/wiki/Black-box_testing

[8] http://en.wikipedia.org/wiki/Regression_testing