**ANDROID SMART CITY TRAVELER**

**ABSTRACT:**

Android Smart City Traveler by the Name indicated smartly makes it way in analyzing user’s likes and dislikes and the time period the user is willing to explore a place and gives him with Amazing results in the form of 3 paths to utilize the time. This System is basically used to help a traveler new to the city or anyone who wants to explore a city in the given time period, the system makes use of the Foursquare Api to get all the locations and places with all their information to sort and place it before the user in 3 paths to make his choice. The Places are sorted and selected based on the top rakings by the foursquare. During the user Registration the user is asked some questions helping them to filter out in searching the places, the places are displayed on the maps giving a clear idea of the location and giving the paths from one place to another from the start location to the end location. The Time shouldn’t exceed 22 hours and the travel plan u chooses is saved only for a single day and exceeding will be dissolved. If the time specified by the user exceeds 8 hours i.e. between 9am to 7 pm the system also asks u whether u wants to visit a adventure or water park and will show u options based on the rankings and reviews about it. Since the Traveler may be new to the city not knowing any place, in the map view if the user clicks on the marker he can see the ratings and reviews which are recorded from the Foursquare itself. The System requires An Working Internet Connection all the time for the app to work.

**INTRODUCTION:**

At present, in general tourists and travellers waste a lot of time planning and deciding their trips to achieve maximum satisfaction. In this context, this application aims to identify the main computing needs to support the improvement of tourist point of promotion for the traveller, by the means of an easy to use mobile application proposal.

Normally, most travellers like to visit the famous sightseeing spots as well as local charms unique to that place. To achieve this, we propose a system that can automatically show a travel route and plan for the user. This application also leads to quicker decision making with respect to places to visit.

This system is basically used to help a traveller new to the city or anyone who wants to explore a city within a specific time period. The user is supposed to enter his/her interests and preferences while signing up. Once the account has been created, the user can choose the location manually or let the system detect his/her current location as the starting and ending point of the trip. Then, the start and end time of the trip must be specified by the user. Since all the trips of a user will be stored, he/she can also view the previous trips. Smart City Traveller as the name indicates, smartly makes its way in analyzing users’ interests and preferences and the time period the user is willing to explore a place and designs an itinerary and a route with the best tourist spots around the selected location such that he/she returns to the starting location by the specified end time. This makes use of shortest path algorithms for determining the route.

The system makes use of the Google Maps API to get all the places around the selected location with all their information. Then, these locations are sorted based on ratings, distance, and various other constraints to place it before the user.

**LITERATURE SURVEY**

The development of internet stimulates the emergence of various Online Tourism Agencies (OTA) to post their service information online [1]. One challenging problem of OTA is how to recommend appropriate travel routes for users with different requirements. The path planning of scenic tourism is a relatively new research field. Some recent research on the travel route planning has been carried out.

Ying Xu, Tao Hu and Ying Li [2] proposed that a new Improved PRR algorithm (IPRR) based on the PRR by considering different personalized requirements in order to recommend high-quality travelling routes for customers. The IPRR algorithm takes various factors into account, including the user’s personal preferences, user types, the real-time traffic condition of the tourism region (i.e. the real-time nodes and the number of people on the path), and the historical statistical data (i.e. historical tourists number at the spot).

Graph search algorithms have often been adapted for both indoor and outdoor path planning [3]–[7]. In these applications, an operational terrain of a mobile agent (e.g. a vehicle, a mobile robot, etc.) is represented using a graph that consists of a set of nodes and a set of edges. A node represents a special location on the terrain surface and an edge represents the connection between two nodes. An edge is associated with one or more costs. For examples, the edge cost is a distance measurement when finding the shortest path and a time measurement when finding the fastest path. Early works on graph search algorithms are based on uniform cost search mechanisms. Dijkstra’s algorithm [4] is a prime example for such uniform cost search algorithms. It starts a search process from a source node and iteratively selects a node for expansion until it selects a target node for expansion. Here, node selection is based on the cost between the source node and a given node without considering its remaining cost to the target node. Thus, in finding the shortest path, it expands all the nodes that are closer to the source node compared to the target node. Obviously, this results in excessive expansion of nodes that do not lie on the optimal path, thus, degrading the efficiency.

The same techniques discussed for shortest path planning might not always be used to find the fastest path in outdoor environments due to the inability of mobile agents to travel at their peak speed everywhere in irregular terrains. Mobility maps are an effective way for dealing with such irregularities. In the paper proposed by Wanmai Yuan, Nuwan Ganganath, Chi-Tsun Cheng, Guo Qing, and Francis C.M. Lau [8], a grid-based mobility maps was introduced for representing speed limitations in outdoor terrains. Further, a heuristic approach for finding the fastest path on such maps was introduced. The proposed heuristic is proven to be both admissible and consistent. Therefore, it can be used with A\*-like heuristic search algorithms for obtaining fastest paths efficiently.

**EXISTING SYSTEM**

The mobile client’s current location is one of the most important information for location related system. Mobile phones need to report their own locations to the remote server periodically, so that the information they want can be suitably queried. From the point of view of the service, the simplest method of locating is to let user tell his or her location, but this method requires extra effort because the user needs to define his or her location and input it to the system. The user can be located with different positioning systems. The advantage of this method is that the users do not need any extra equipment. Mainly through newspaper, magazines, radio and other simple ways those are available easily. But problem is that tourists are not able to get travel information timely when they are on the move. While today's mobile devices are becoming more intelligent, compared with PC, they still have the following limitations like small screen and tiny keyboard, limited CPU capacity, limited memory space, slow and fitful Internet connection. Many mobiles of recent decades have travel guide application. But the application on these mobiles works slow due to continues acquisition of the bandwidth. Therefore, the mobile end-user’s operation is very difficult, and the contents display on the screen of mobile device is limited.

**DRAWBACKS**

➢ It requires active internet connection else error may occur.

➢ Since for the data the system is dependent so if anything goes wrong with the foursquare,

the system is liable to give wrong data.

➢ The android mobile user will not be able to insert or view details if the server goes down.

Thus there is disadvantage of single point failure.

**PROPOSED SYSTEM**

The Places are sorted and selected based on the top rakings by the foursquare. During the

Make trip/create trip the user is asked some questions helping them to filter out in searching the places, the places are displayed on the maps giving a clear idea of the location and giving the paths from one place to another from the start location to the end location. The Time shouldn’t exceed 22 hours and the travel plan u chooses is saved only for a single day and exceeding will be dissolved. If the time specified by the user exceeds 8 hours i.e. between 9am to 7 pm the system also asks u whether u wants to visit a adventure or water park and will show u options based on the rankings and reviews about it. Since the Traveler may be new to the city not knowing any place, in the map view if the user clicks on the marker he can see the ratings and reviews which are recorded from the Foursquare itself. The System requires An Working Internet Connection all the time for the app to work. The frontend of the System makes use of Android Studio while SQL Server as the Backend.

**ADVANTAGES**

➢ The Data is very accurate and authentic as we take all the data from Foursquare.

➢ The Student has to Login to use keeping the data secure.

MODULES:

The application contains various modules from login to logout.

1. REGISTRATION
2. LOGIN
3. DEVICE GEO LOCATION CAPTURE
4. MAKE YOUR SCHEDULE
5. INTEGRATION OF FOURSQUARE APIS
6. DATABASE.

**REGISTRATION**: The application is allowed to the user to register in the application with basic details name, mobile, email, and password. This application uses Sqlite Database where the user details will be stored in the mobile database, which is primary source.

**LOGIN**: Once the user is registered user can get an access to login to the application. The entered mobile and password will be checked in the database, if the user is present then application allow him to do next process else it asks for check credentials or register again

**DEVICE GEO LOCATION CAPTURE:**

Nowadays many applications capture the location of the user to locate or track any one. The idea behind this the application show be added with google geo permission. On installing the app the user is asked to provide the location permission, then with help of location GPS which is inbuilt app, gives the current latitude and longitude of the device/user.

**MAKE YOUR SCHEDULE:**

The user can make the schedule the travel plan, initially the application captures the user latitude and longitude of the device and ask some question to select date and time. Once the date and time is captured again user has to select to which place he need to travel like shopping, temple, hills station, once user select these then application call FOURSQUARE API to retrieve the data based on the user intreset and gives the nearst laction and user can select and save.

INTEGRATION OF FOURSQUARE APIS:

What is Foursquare API?

Foursquare is a social location service that allows users to explore the world around them. Users can download the Foursquare application to their iPhone, Blackberry, or Android phone and sign up for free, then connect their Foursquare accounts to their other social media accounts.

After users download the free application and connect on Facebook or Twitter, they can connect with their friends who are also active on Foursquare. Whenever they or their friends check in to a place (that is, they're at a location and they tell others that they are there), the message is broadcast to their friends via Twitter or Facebook.

When a user checks in enough times, that user becomes the mayor of a location, which may or may not give the user access to special offers, depending on the business running a location. For example, a coffee shop might extend a free drink to anyone who becomes a mayor. Users can also earn badges as they explore and check in to locations, leave tips for other users (for instance, "The margaritas here are great!"), and may even have the opportunity to create locations that haven't appeared yet on the service.

Info about FourSuare API..!

The Foursquare API allows application developers to interact with the Foursquare platform. The API itself is a RESTful set of addresses to which you can send requests, so there's really nothing to download onto your server. That being said, in this article, you use a set of PHP libraries to help simplify requests and responses, but this is an entirely optional step.

You can currently request output in XML or JSON format, making requests to URLs that look like this: http://api.foursquare.com/v1/user.json. If you don't use an extension on your request,then XML is served back to you. A request to http://api.foursquare.com/v1/user results in an XML output.

There are both GET and POST methods you can use, which means you aren't just limited to reading from feeds, you can also do useful stuff like checking in and creating locations using the API. As for rate limits, your application is limited to 200 requests per hour per method, so you probably want to implement some kind of result caching to "play nice."

For the most part, you want to use basic or OAuth authentication to take full advantage of the various methods and services. The example in this article (/venues) doesn't require any authentication to get started, but it does have additional features available to those who do use authentication.

To make use of the FourSquareApi

1. Go to <https://developer.foursquare.com/>
2. Register for the account.
3. Please make down all the CLIENT ID and CLIENT SECRET\_ID provide buy foursquare Api
4. Take out all APIs which is necessary.
5. Implement web-services to integrate Mobile Android app.
6. Add all dependencies required for the web services.

**My schedule:** here all the list of data will be populating based on the user selection with respect to his latitude and longitude. And he can choose any location and start seeing in google map.

**Notification**: once the user selects places the data will be stored in user database and alarm with notification will be set. Once the time matches with device time and trip time user will get the notification saying that you can start moving to next place.

**Project Implementation:**

1. This task will contain more spotlight/focus and information to deal with, So be prepared.
2. Now let start with the first step is to make another undertaking by any name then as the documents get stacked and as we know that Gradle is built, we’ll have activity\_main.xml and MainActivity.java and here we have MainActivity which will be recognized as the main homepage screen with two-buttons.
3. In the second step, we have to make the User Interface of our App. The User Interface will be exceptionally major and simple to utilize.
4. After UI is complete, move forward toward the coding part.
5. In MainActivity.java class, Intent is utilizing to explore users’ decisions. An Intent is a class that is utilized to depict an activity to be performed. It is an essential way that assists engineers with beginning another movement inside an application. It can likewise be utilized as a device to impart between activities.
6. For all activities, we should declare the intent filters in the manifest file.
7. Now our Smart city traveler App utilizes Foursquare and google map to achieve show the guide, include markers to guide and access clients' scopes and longitudes. So as to use Google Maps administration, the application needs to enroll for a building key on Web headways.
8. Presently make java class according to your needs, for example, a hotel.java, transport.java,shopping.java, and so on and start coding.
9. Next, it's time for setting up (here setting means coding, declare, initialize) all methods, listeners, and buttons needed in  all the pages

Like for transport.java, shopping.java,  hotel.java, cafe.java, etc. , these means get done with all the coding for each and every page.

Advantages:

* The Data is very accurate and authentic as we take all the data from Foursquare.
* The Student has to Login to use keeping the data secure.
* The user can also find the paths to follow to reach the final destination in map which gives a better view to the users.
* Since the location can be viewed in map, the user can even zoom in and zoom out to get a better view.
* The system gives many travel plans for the user to select.
* The usage of this application greatly reduces the time required to search for a place.
* The application also leads to quicker decision making with respect to places to visit.

Disadvantages:

* It requires active internet connection else error may occur.
* Since for the data the system is dependent so if anything goes wrong with the foursquare, the system is liable to give wrong data.
* The android mobile user will not be able to insert or view details if the server goes down. Thus there is disadvantage of single point failure

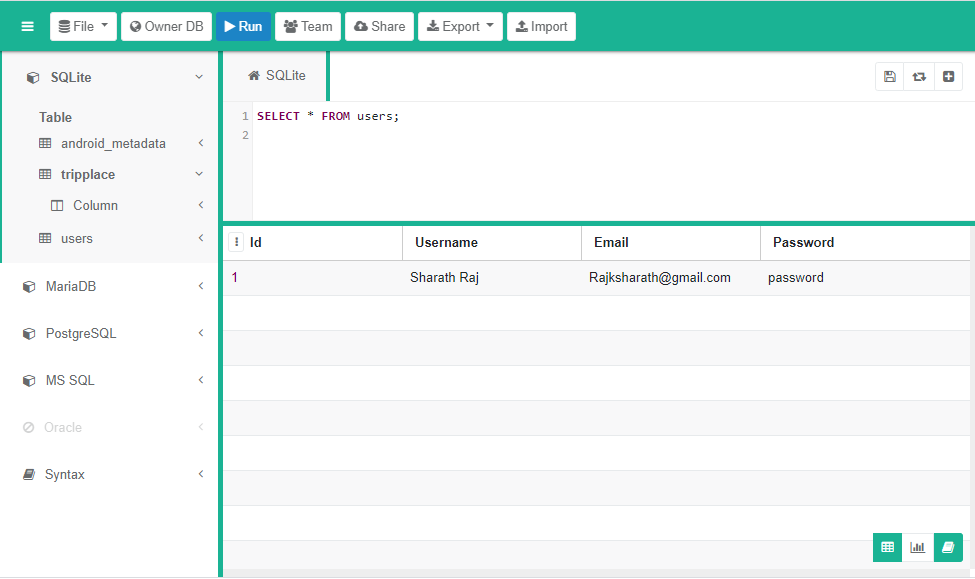
Data base Schemes.

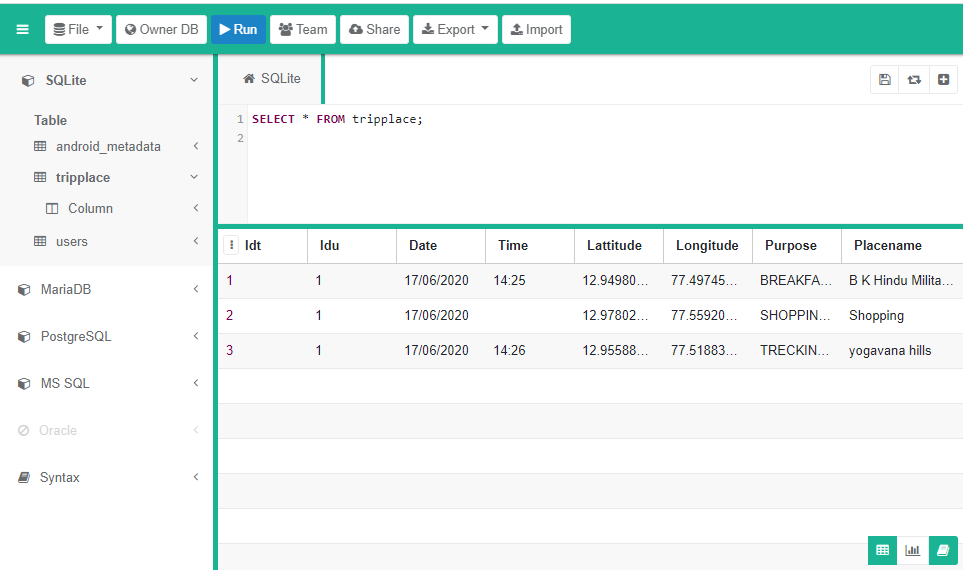
1. USER REGISTRATION.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute | Type | Length | Not-Null | key |
| User\_id | Integer | 6 | True | Primary key |
| Name | String | 20 | True |  |
| Mobile | Integer | 10 | True |  |
| Email | String | 20 | True |  |
| Password | String | 10 | True |  |

1. SECELECTED PLACES

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute | Type | Length | Not-null | key |
| Trip\_id | Integer | 6 | True | Primary key |
| Schedule\_id | Integer | 6 | True | Secondary key |
| User\_id | Integer | 6 | True | Secondary key |
| Place name | String | 20 | True |  |
| Place\_lat | Integer | 6 | True |  |
| Place\_lang | Integer | 6 | True |  |





DATA FLOW DIAGRAMS:

REGISTER

LOGIN

MAKE SCHEDULE

HOME PAGE

VIEW MY SCHEDULE

SELECT TIME DATE

NOTIFICATION WHEN TIME REACHES

FROM DATABASE

SELECT OPTIONS TO TRAVELL

DATA FROM API

CALL **FOURSQUARE** API

EXIT

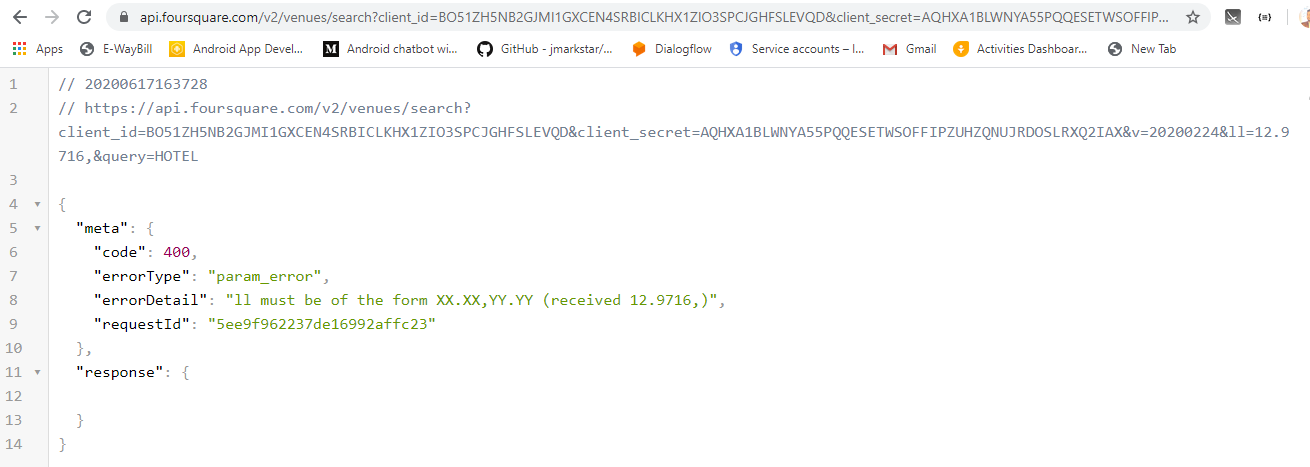
SELECT PLACE

RAW DATA FROM FOUR SQUARE API

SEARCH BASED ON HOTEL:



IF THERE IS AN ERROR:



ABOUT TECHNLOLY

**What is Android?**



Before learning all topics of android, it is required to know what is android.

**Android** is a software package and linux based operating system for mobile devices such as tablet computers and smartphones. It is developed by Google and later the OHA (Open Handset Alliance). Java language is mainly used to write the android code even though other languages can be used. The goal of android project is to create a successful real-world product that improves the mobile experience for end users. There are many code names of android such as Lollipop, Kitkat, Jelly Bean, Ice cream Sandwich, Froyo, Ecliar, Donut etc which is covered in next page.

Applications ("[apps](https://en.wikipedia.org/wiki/Mobile_app)"), which extend the functionality of devices, are written using the [Android software development](https://en.wikipedia.org/wiki/Android_software_development) kit (SDK)  and, often, the [Java](https://en.wikipedia.org/wiki/Java_(programming_language)) programming language.[[78]](https://en.wikipedia.org/wiki/Android_(operating_system)#cite_note-78) Java may be combined with [C](https://en.wikipedia.org/wiki/C_(programming_language))/[C++](https://en.wikipedia.org/wiki/C%2B%2B), together with a choice of non-default [runtimes](https://en.wikipedia.org/wiki/Runtime_library) that allow better C++ support. The [Go](https://en.wikipedia.org/wiki/Go_(programming_language)) programming language is also supported, although with a limited set of [application programming interfaces](https://en.wikipedia.org/wiki/Application_programming_interface) (API). In May 2017, Google announced support for Android app development in the [Kotlin programming language](https://en.wikipedia.org/wiki/Kotlin_(programming_language)" \o "Kotlin (programming language)).

The SDK includes a comprehensive set of development tools, including a [debugger](https://en.wikipedia.org/wiki/Debugger), [software libraries](https://en.wikipedia.org/wiki/Software_library), a handset [emulator](https://en.wikipedia.org/wiki/Emulator) based on [QEMU](https://en.wikipedia.org/wiki/QEMU), documentation, sample code, and tutorials. Initially, Google's supported [integrated development environment](https://en.wikipedia.org/wiki/Integrated_development_environment) (IDE) was [Eclipse](https://en.wikipedia.org/wiki/Eclipse_(software)) using the Android Development Tools (ADT) plugin; in December 2014, Google released [Android Studio](https://en.wikipedia.org/wiki/Android_Studio), based on [IntelliJ IDEA](https://en.wikipedia.org/wiki/IntelliJ_IDEA), as its primary IDE for Android application development. Other development tools are available, including a [native development kit](https://en.wikipedia.org/wiki/Native_development_kit) (NDK) for applications or extensions in C or C++, [Google App Inventor](https://en.wikipedia.org/wiki/Google_App_Inventor), a visual environment for novice programmers, and various [cross platform mobile web applications frameworks](https://en.wikipedia.org/wiki/Multiple_phone_web_based_application_framework). In January 2014, Google unveiled an framework based on [Apache Cordova](https://en.wikipedia.org/wiki/Apache_Cordova) for porting [Chrome](https://en.wikipedia.org/wiki/Google_Chrome) [HTML 5](https://en.wikipedia.org/wiki/HTML_5) [web applications](https://en.wikipedia.org/wiki/Web_app) to Android, wrapped in a native application shell.

Android has a growing selection of third-party applications, which can be acquired by users by downloading and installing the application's [APK](https://en.wikipedia.org/wiki/APK_(file_format)) (Android application package) file, or by downloading them using an [application store](https://en.wikipedia.org/wiki/Application_store) program that allows users to [install, update, and remove applications](https://en.wikipedia.org/wiki/Package_manager) from their devices. [Google Play Store](https://en.wikipedia.org/wiki/Google_Play_Store) is the primary application store installed on Android devices that comply with Google's compatibility requirements and license the Google Mobile Services software.[[86]](https://en.wikipedia.org/wiki/Android_(operating_system)#cite_note-ars-irongrip-86)[[87]](https://en.wikipedia.org/wiki/Android_(operating_system)#cite_note-geek-poweredby-87) Google Play Store allows users to browse, download and update applications published by Google and third-party developers; as of July 2013, there are more than one million applications available for Android in Play Store. As of July 2013, 50 billion applications have been installed.[[89]](https://en.wikipedia.org/wiki/Android_(operating_system)#cite_note-Google_Play_Hits_1_Million_Apps-89)[[90]](https://en.wikipedia.org/wiki/Android_(operating_system)#cite_note-android-compatibility-90) Some carriers offer direct carrier billing for Google Play application purchases, where the cost of the application is added to the user's monthly bill. As of May 2017, there are over one billion active users a month for Gmail, Android, Chrome, Google Play and Maps.

Due to the open nature of Android, a number of third-party application marketplaces also exist for Android, either to provide a substitute for devices that are not allowed to ship with Google Play Store, provide applications that cannot be offered on Google Play Store due to policy violations, or for other reasons. Examples of these third-party stores have included the [Amazon Appstore](https://en.wikipedia.org/wiki/Amazon_Appstore), [GetJar](https://en.wikipedia.org/wiki/GetJar" \o "GetJar), and SlideMe. [F-Droid](https://en.wikipedia.org/wiki/F-Droid), another alternative marketplace, seeks to only provide applications that are distributed under [free and open source](https://en.wikipedia.org/wiki/Free_and_open_source)[licenses](https://en.wikipedia.org/wiki/Free_software_license).

What is Open Handset Alliance (OHA)

It's a consortium of 84 companies such as google, samsung, AKM, synaptics, KDDI, Garmin, Teleca, Ebay, Intel etc. It was established on 5th November, 2007, led by Google. It is committed to advance open standards, provide services and deploy handsets using the Android Plateform.

History of Android

The history and versions of android are interesting to know. The code names of android ranges from A to O currently, such as **Aestro**, **Blender**, **Cupcake**, **Donut**, **Eclair**, **Froyo**, **Gingerbread**, **Honeycomb**, **Ice Cream Sandwitch**, **Jelly Bean**, **KitKat**, **Lollipop**.**Marshmellow,** and **Naugot,** Oreo. Let's understand the android history in a sequence.

1) Initially, **Andy Rubin** founded Android Incorporation in Palo Alto, California, United States in October, 2003.

2) In 17th August 2005, Google acquired android Incorporation. Since then, it is in the subsidiary of Google Incorporation.

3) The key employees of Android Incorporation are **Andy Rubin**, **Rich Miner**, **Chris White** and **Nick Sears**.

4) Originally intended for camera but shifted to smart phones later because of low market for camera only.

5) Android is the nick name of Andy Rubin given by coworkers because of his love to robots.

6) In 2007, Google announces the development of android OS.

7) In 2008, HTC launched the first android mobile.

Android Versions, Codename and API

Let's see the android versions, codenames and API Level provided by Google.

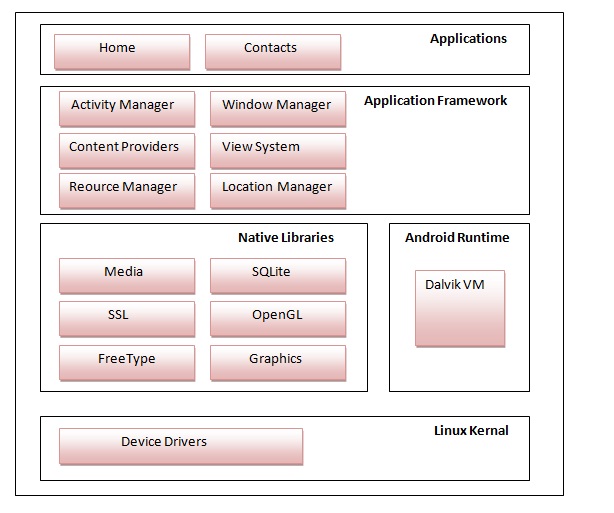
|  |  |  |
| --- | --- | --- |
| **Version** | **Code name** | **API Level** |
| 1.5 | Cupcake | 3 |
| 1.6 | Donut | 4 |
| 2.1 | Eclair | 7 |
| 2.2 | Froyo | 8 |
| 2.3 | Gingerbread | 9 and 10 |
| 3.1 and 3.3 | Honeycomb | 12 and 13 |
| 4.0 | Ice Cream Sandwitch | 15 |
| 4.1, 4.2 and 4.3 | Jelly Bean | 16, 17 and 18 |
| 4.4 | KitKat | 19 |
| 5.0 | Lollipop | 21 |

# Android Architecture

**android architecture** or **Android software stack** is categorized into five parts:

1. linux kernel
2. native libraries (middleware),
3. Android Runtime
4. Application Framework
5. Applications

Let's see the android architecture first.



Android Emulator

**Android Emulator** is used to run, debug and test the android application. If you don't have the real device, it can be the best way to run, debug and test the application.

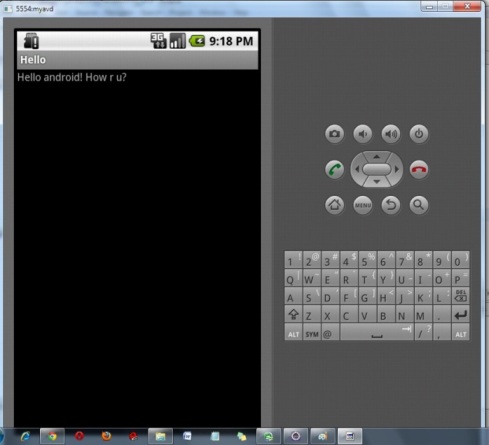
It uses an open source processor emulator technology called **QEMU**.

The emulator tool enables you to start the emulator from the command line. You need to write:

emulator -avd <AVD NAME>

In case of Eclipse IDE, you can create AVD by **Window menu > AVD Manager > New**.

In the given image, you can see the android emulator, it displays the output of the hello android example.



DESIGN OF SCREENS .XML LANGUAGE;

# Build a UI with Layout Editor

1. [Introduction to the editor](https://developer.android.com/studio/write/layout-editor.html#intro)
   1. [Change the preview appearance](https://developer.android.com/studio/write/layout-editor.html#change-appearance)
2. [Create a new layout](https://developer.android.com/studio/write/layout-editor.html#create-layout)
   1. [Create a layout variant](https://developer.android.com/studio/write/layout-editor.html#create-variant)
   2. [Convert a layout to ConstraintLayout](https://developer.android.com/studio/write/layout-editor.html#convert-constraint)
3. [Add views to your layout](https://developer.android.com/studio/write/layout-editor.html#add-views)
4. [Edit view attributes](https://developer.android.com/studio/write/layout-editor.html#edit-properties)
   1. [Download and apply fonts to text](https://developer.android.com/studio/write/layout-editor.html#download_and_apply_fonts_to_text)

In Android Studio's Layout Editor, you can quickly build layouts by dragging widgets into a visual design editor instead of writing the layout XML by hand. The editor can preview your layout in different Android devices and versions, and you can dynamically resize the layout to be sure it works well on different screen sizes. The Layout Editor is especially powerful when building a new layout with ConstraintLayout—a layout manager provided in a support library that's compatible with Android 2.3 (API level 9) and higher.

This page provides an overview of the Layout Editor's interface and features. To learn more about how to build a layout with ConstraintLayout, see [Build a Responsive UI with ConstraintLayout](https://developer.android.com/training/constraint-layout/index.html).

## Introduction to the editor

The Layout Editor appears when you open an XML layout file. Corresponding to the numbers in figure 1, the regions of the editor are as follows:

1. **Palette**: Provides a list of widgets and layouts that you can drag into your layout in the editor.
2. **Component** **Tree**: Shows the view hierarchy for your layout. Click an item here to see it selected in the editor.
3. **Toolbar**: Provides buttons to configure your layout appearance in the editor and to change some layout attributes.
4. **Design Editor**: Displays your layout in a combination of the Design and Blueprint views.
5. **Attributes**: Provides controls for the selected view's attributes.

**ABOUT API:**

In [computer programming](https://en.wikipedia.org/wiki/Computer_programming), an application programming interface (API) is a set of [subroutine](https://en.wikipedia.org/wiki/Subroutine) definitions, [protocols](https://en.wiktionary.org/wiki/Protocol), and tools for building [application software](https://en.wikipedia.org/wiki/Application_software). In general terms, it is a set of clearly defined methods of communication between various software components. A good API makes it easier to develop a [computer program](https://en.wikipedia.org/wiki/Computer_program) by providing all the building blocks, which are then put together by the [programmer](https://en.wikipedia.org/wiki/Programmer). An API may be for a web-based system, [operating system](https://en.wikipedia.org/wiki/Operating_system), [database system](https://en.wikipedia.org/wiki/Database_system), [computer hardware](https://en.wikipedia.org/wiki/Computer_hardware) or [software library](https://en.wikipedia.org/wiki/Library_(computing)). An API specification can take many forms, but often includes specifications for [routines](https://en.wikipedia.org/wiki/Subroutine), [data structures](https://en.wikipedia.org/wiki/Data_structure), [object classes](https://en.wikipedia.org/wiki/Class_(computer_programming)), [variables](https://en.wikipedia.org/wiki/Variable_(computer_science)) or [remote calls](https://en.wikipedia.org/wiki/Remote_procedure_call). [POSIX](https://en.wikipedia.org/wiki/POSIX), [Windows API](https://en.wikipedia.org/wiki/Windows_API) and [ASPI](https://en.wikipedia.org/wiki/Advanced_SCSI_programming_interface) are examples of different forms of APIs. Documentation for the API is usually provided to facilitate usage.

Android Sqlite database.(LOCAL STORAGE)

Android SQLite Tutorial

SQLite is an open-source relational database i.e. used to perform database operations on android devices such as storing, manipulating or retrieving persistent data from the database.

It is embedded in android bydefault. So, there is no need to perform any database setup or administration task.

Here, we are going to see the example of sqlite to store and fetch the data. Data is displayed in the logcat. For displaying data on the spinner or listview, move to the next page.

SQLiteOpenHelper class provides the functionality to use the SQLite database.

SQLiteOpenHelper class

The android.database.sqlite.SQLiteOpenHelper class is used for database creation and version management. For performing any database operation, you have to provide the implementation of onCreate() and onUpgrade() methods of SQLiteOpenHelper class.

Constructors of SQLiteOpenHelper class

There are two constructors of SQLiteOpenHelper class.

Constructor

SQLiteOpenHelper(Context context, String name, SQLiteDatabase.CursorFactory factory, int version) creates an object for creating, opening and managing the database.

SQLiteOpenHelper(Context context, String name, SQLiteDatabase.CursorFactory factory, int version, DatabaseErrorHandler errorHandler) creates an object for creating, opening and managing the database. It specifies the error handler.

Methods of SQLiteOpenHelper class

There are many methods in SQLiteOpenHelper class. Some of them are as follows:

Method

public abstract void onCreate(SQLiteDatabase db) called only once when database is created for the first time.

public abstract void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) called when database needs to be upgraded.

public synchronized void close () closes the database object.

public void onDowngrade(SQLiteDatabase db, int oldVersion, int newVersion) called when database needs to be downgraded.

SQLiteDatabase class

It contains methods to be performed on sqlite database such as create, update, delete, select etc.

Methods of SQLiteDatabase class

There are many methods in SQLiteDatabase class. Some of them are as follows:

Method

void execSQL(String sql) executes the sql query not select query.

long insert(String table, String nullColumnHack, ContentValues values) inserts a record on the database. The table specifies the table name, nullColumnHack doesn't allow completely null values. If second argument is null, android will store null values if values are empty. The third argument specifies the values to be stored.

int update(String table, ContentValues values, String whereClause, String[] whereArgs) updates a row.

Cursor query(String table, String[] columns, String selection, String[] selectionArgs, String groupBy, String having, String orderBy) returns a cursor over the resultset.

## Advantage Android

1. Multitasking – Yups, Android pshones can run many applications, it means you can browse, Facebook while listened to the song.
2. Ease of Notification – Any SMS, Email, or even the latest articles from an RSS Reader, there will always be a notification on the Home Screen Android phone, do not miss the LED indicator is blinking, so you will not miss a single SMS, Email or even Misscall .
3. Easy access to thousands of applications via the Google Android Android App Market – When you love to  install applications or games, through Google’s Android App Market, Agan can download applications for free. There are many thousands of applications and games that are ready for download on Android phones You.
4. Phone options are diverse – Talk Android phone, it will feel ‘different’ than the IOS, if the IOS is limited to the iPhone from Apple, then Android is available on mobile phones from various manufacturers, from [Sony Ericsson](http://mobilecon.info/category/cellphone/manufacturer/sony-ericsson-manufacturer), [Motorola](http://mobilecon.info/category/cellphone/manufacturer/motorola), [HTC](http://mobilecon.info/category/cellphone/manufacturer/htc-manufacturer) to [Samsung](http://mobilecon.info/category/cellphone/manufacturer/samsungs). And each handset manufacturer also presents an Android phone in the style of each, such as Motorola with its Motoblur, Sony Ericsson with its Timescape. So You can freely choose the Android phone in accordance with the ‘brand’ favorite.
5. Can install a modified ROM – not satisfied with the standard view of Android, do not worry there are many custom ROM that can be used in your mobile phones Android .
6. Widget – absolutely right, with the widgets on the homescreen, You can easily access a variety of settings quickly and easily.
7. Google Maniac – If you are a loyal user of Google services ranging from Gmail to Google Reader Android phone has integrated with Google services, so you can quickly check e-mail from Gmail.

### Disadvantage Android

1. Continuous Internet connection – Yups, most Android phones require a simultaneous Internet connection alias continuously active. that means must be prepared to subscribe Agan GPRS packet that suits your needs.
2. Advertising – Application in the Android phones can indeed be obtained easily and for free, but the consequences in each of these applications, will always be ads on display, either the top or bottom of the application.

**REQUIREMENT ANALYSIS**

**3.1 Specific Requirements**

**3.1.1 TECHNOLGIES USED:**

* ANDROID STUDIO as a tool used to build both the Applications, User end and Admin end
* XML language is used to build screen of android app(User Interface)
* Java language is used as Listener for accessing User details.
* APIs(application programming interface) is developed and integrated to android studio in order to communicate with server.
* All the database will be maintained S QLITE db

**3.1.2 Hardware Requirements**

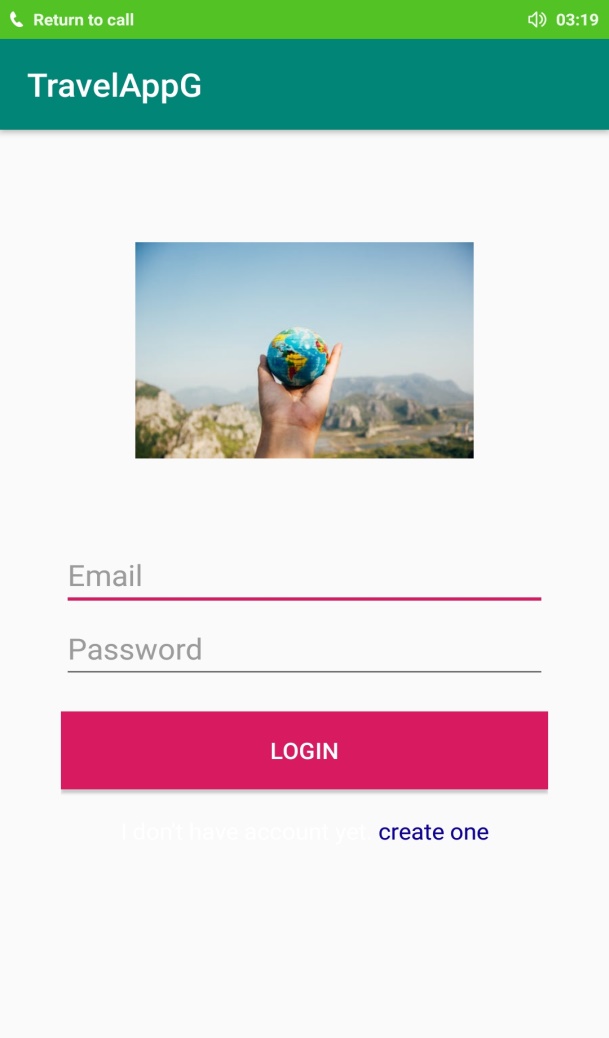
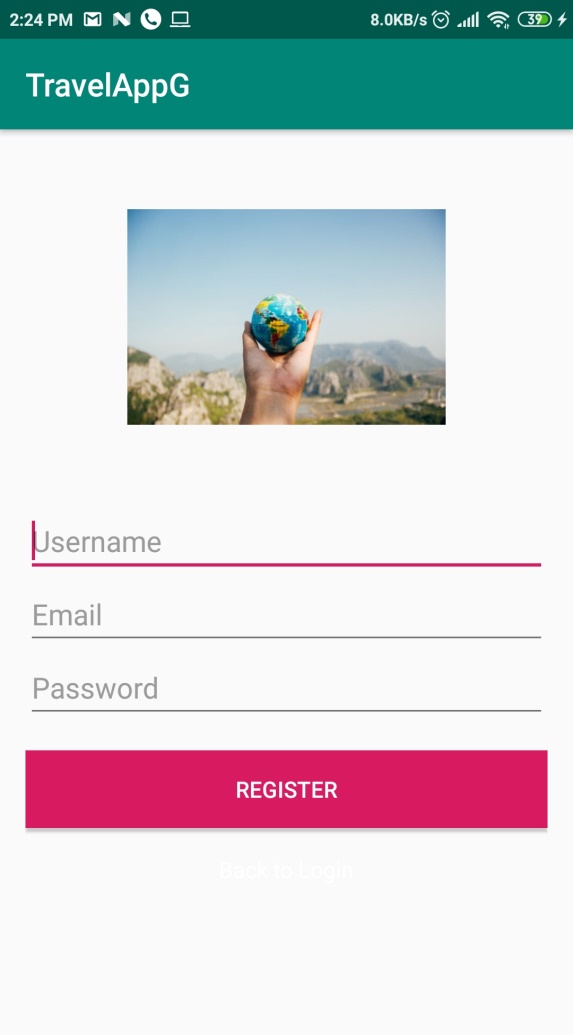
**Mobile Application**

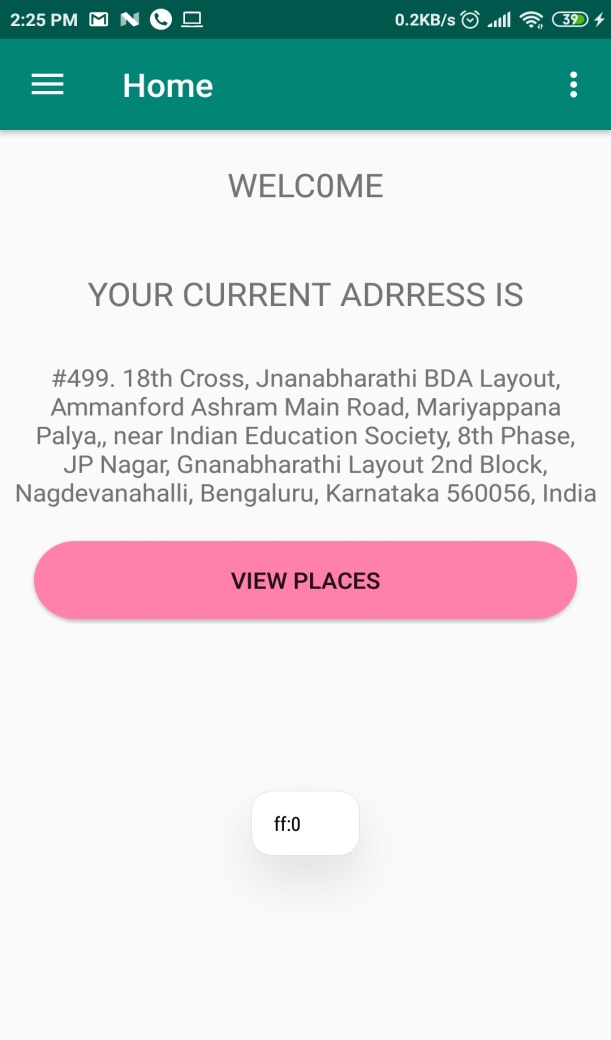
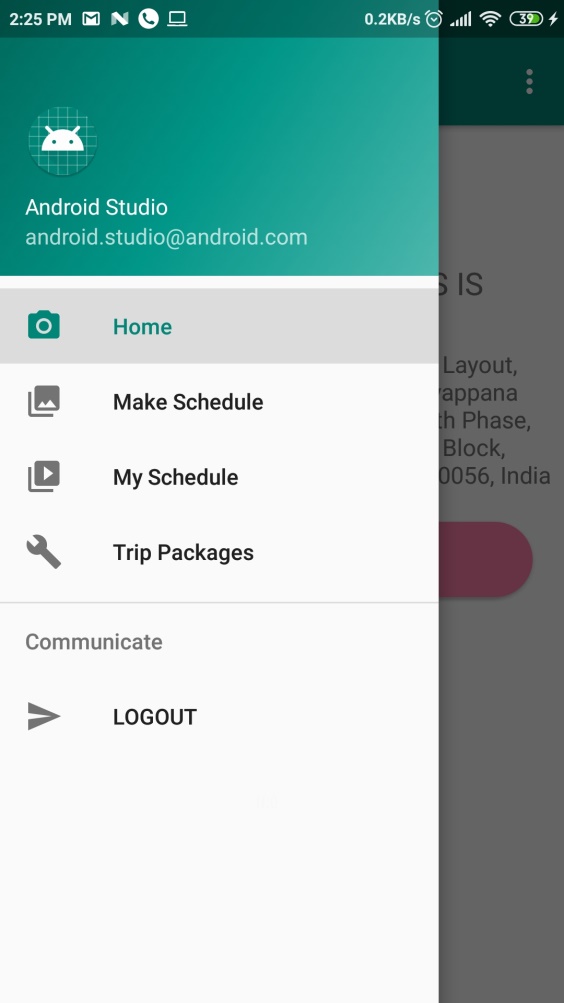
* + - Processor - 1GHz and above
    - Battery - 1200mAh and above
    - RAM - 512 MB and above
    - Memory - 2 GB and above

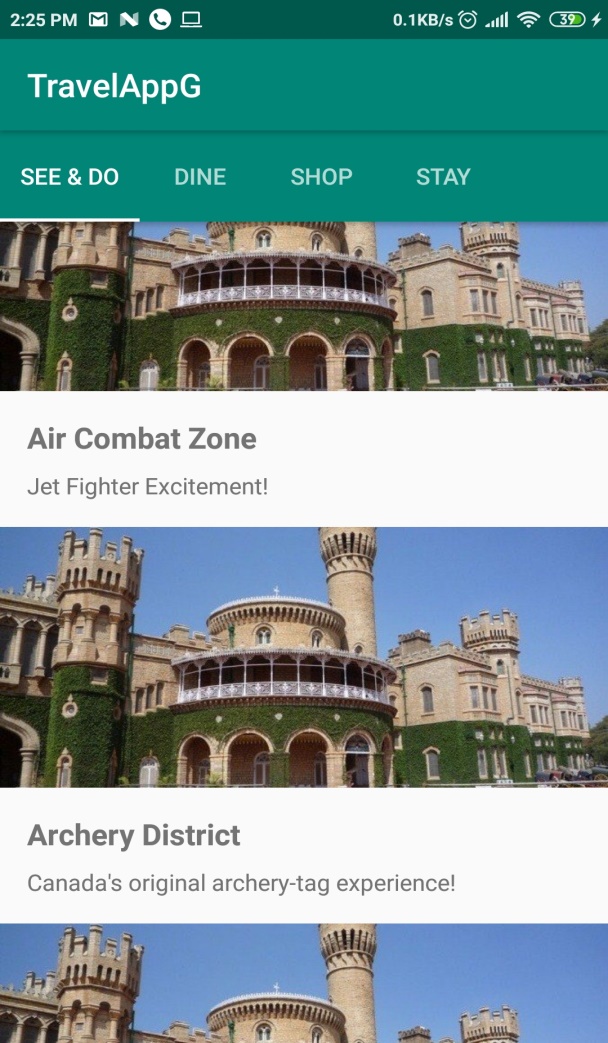
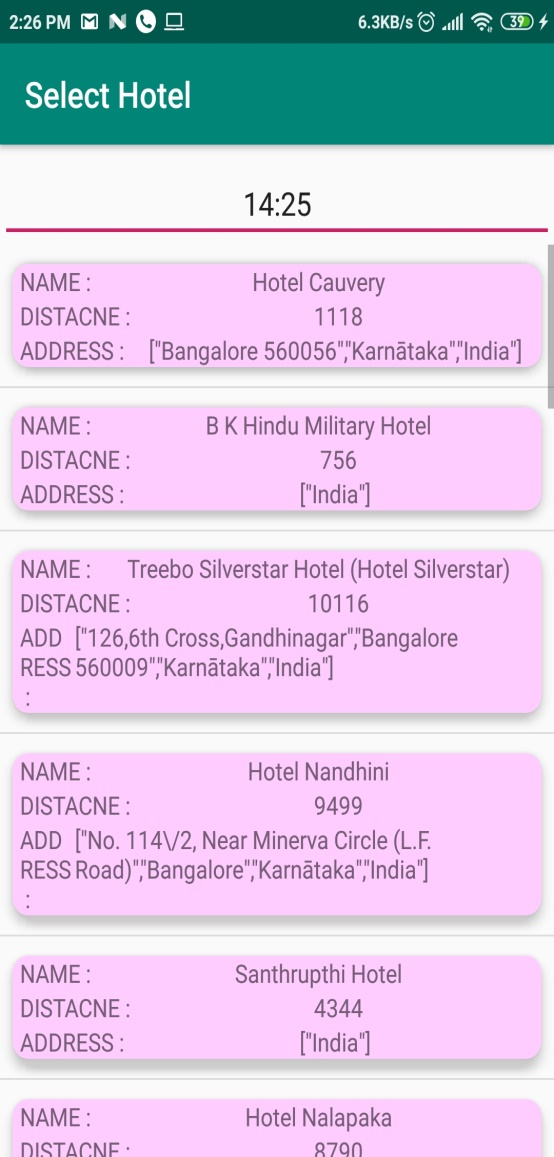
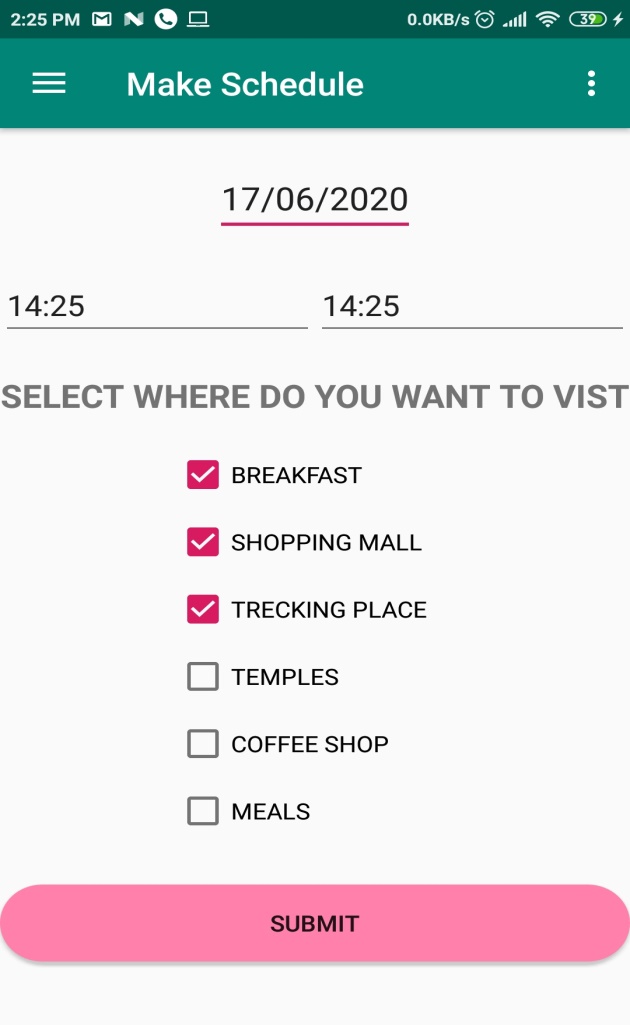
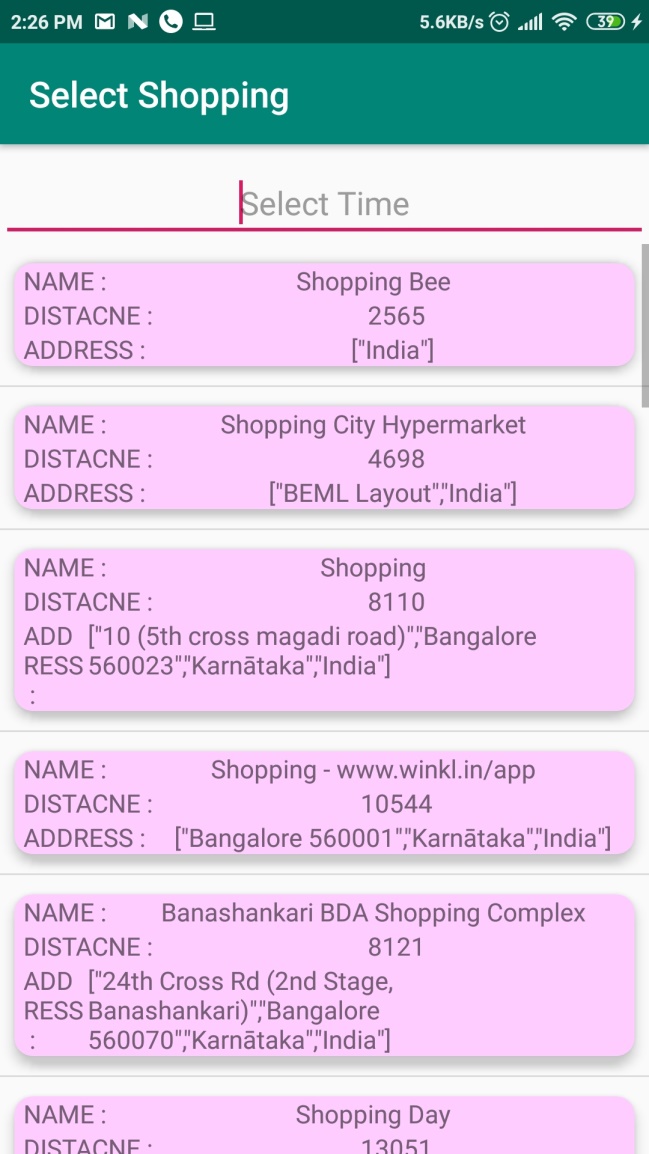
**3.1.3 Software Requirement Specification**

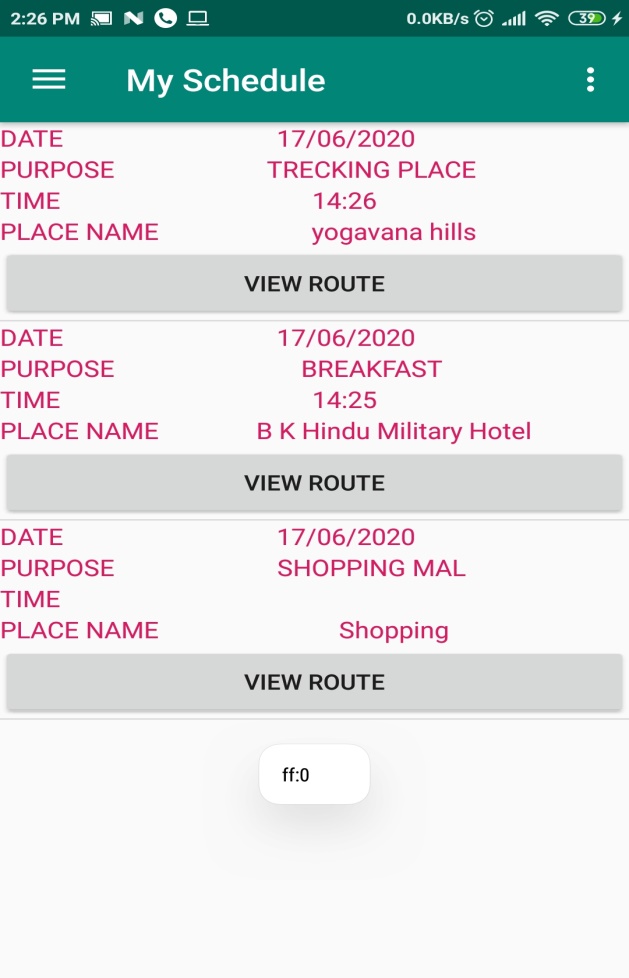
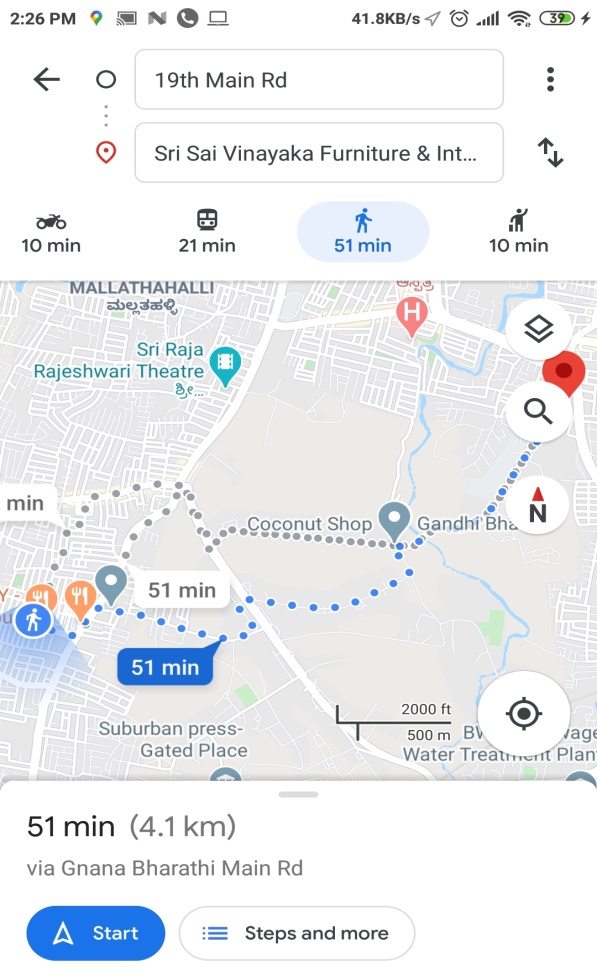
* Operating system : Windows 7 Ultimate.
* Coding Language : ANDROID
* Front-End : ANDROID
* Data Base : SQLITE
* Software : Android studio

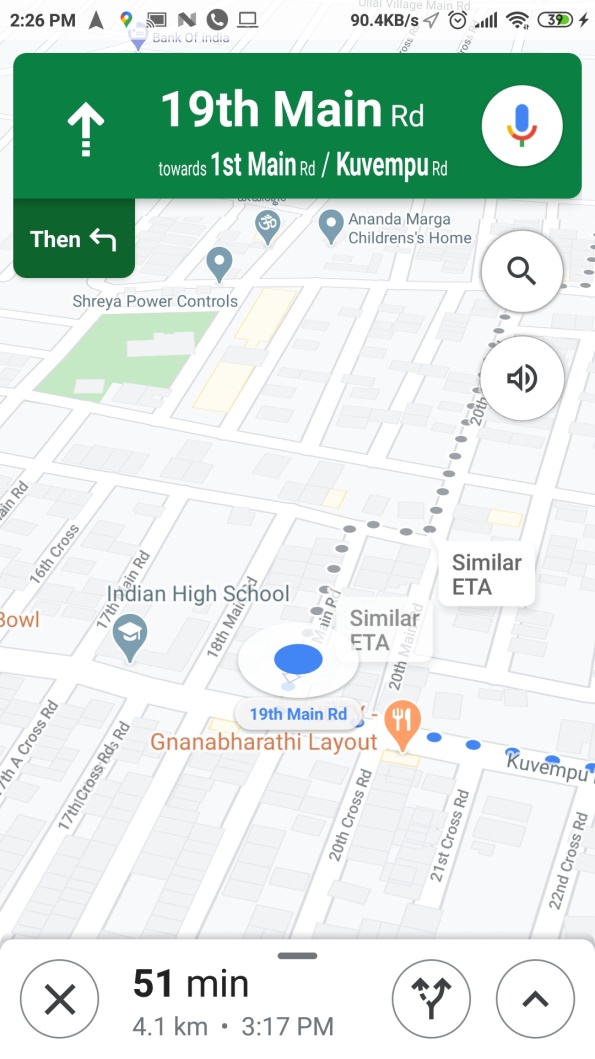
OUTPUT SCREENS:

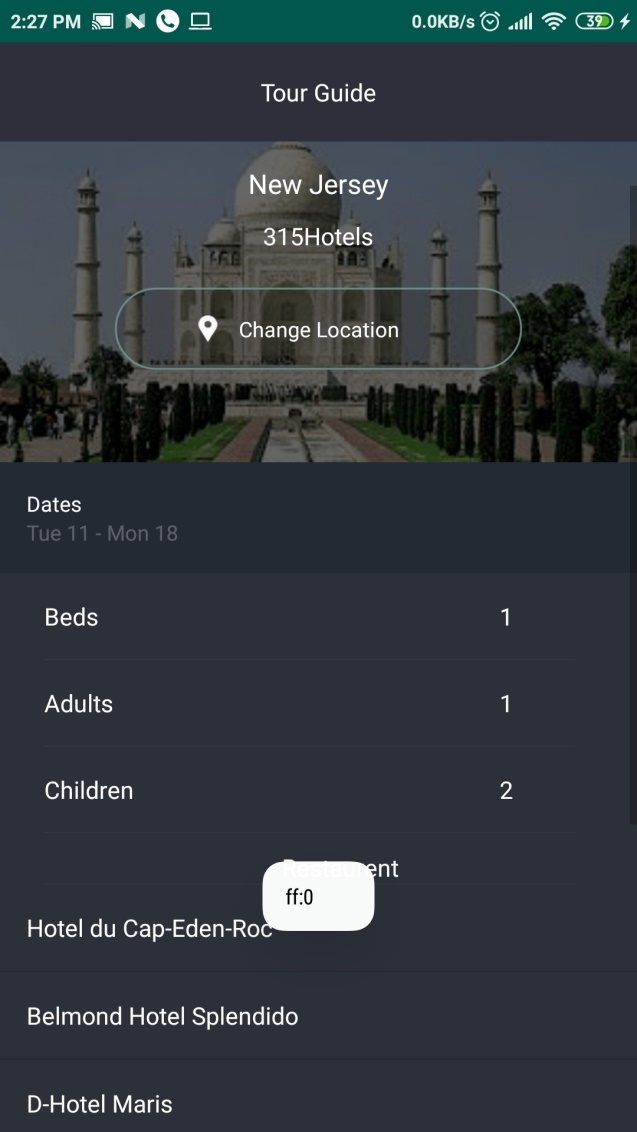
 



**7. CONCLUSION**

Since travelling is one of the important aspect today, it is very necessary that proper planning need to be done beforehand in terms of time management. Most people without using the latest technology waste a lot of time just planning trips. So, an application like Smart City Traveller really helps tourists to utilize their precious time to the fullest and also enjoy their trip at the same time.

**REFERENCES**

**1.** LV H.L., WANG J.L. & DENG F, A Recommendation Algorithm For Individualized Travelling Route. Network New Media

**2.** Ying Xu, Tao Hu, Ying Li “A Travel Route Recommendation Algorithm with Personal Preference” 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD) 2016.

**3.** N. Ganganath, C.-T. Cheng, and C. K. Tse, “Rapidly replanning A\*,” in International Conference on

CyberEnabled Distributed Computing and Knowledge Discovery (CyberC). IEEE, 2016, pp. 386–389.

**4.** E. W. Dijkstra, “A note on two problems in connexion with graphs,” Numerische mathematik, vol. 1, no. 1, pp. 269–271, 1959.

**5.** M. Likhachev, D. Ferguson, G. Gordon, A. Stentz, and S. Thrun, “Anytime search in dynamic graphs,” Artificial Intelligence, vol. 172, no. 14, pp. 1613 – 1643, 2008.

**6.** N. G. Marasinghe Arachchige, “Heuristic search algorithms with applications to path planning on uneven terrains,” Ph.D. dissertation, The Hong Kong Polytechnic University, 2016

**7.** N. Ganganath, C.-T. Cheng, and C. K. Tse, “A constraint-aware heuristic path planner for finding energy-efficient paths on uneven terrains,” IEEE Transactions on Industrial Informatics, vol. 11, no. 3, pp. 601–611,2015.

**8.** Wanmai Yuan, Nuwan Ganganath, Chi-Tsun Cheng, Guo Qing, and Francis C.M. Lau “A Consistent Heuristic for Efficient Path Planning on Mobility Maps” Department of Electronic and Information Engineering, the Hong Kong Polytechnic University (Projects RUWM and GYBKH) and NSF of China (Grant No. 61372095).

**9.** Kazuya Murata and Takayuki Fujimoto “Proposal of Multiple Travel Scheduling System based on Inverse Operation Method” 978-14799- 8679-8/15/$31.00 copyright 2015 IEEE ICIS 2015 June 28-July 1 2015, Las Vegas, USA.

**10.** Ivaldir de Farias Junior, Nelson Leitão Júnior and Marcelo M. Teixeira, “Urbis: A Touristic Virtual Guide”

**11.** https://developer.Google Maps.com/docs/resourc es/categories

**12.** https://www.researchgate.net/figure/2714580 21\_fig2\_Figure-2-Classification-of-DynamicRoute-Planning-Algorithms

**13.** Chin-Jung Huang, Ying-Hong Lin,The Approximate Shortest Distance Route Intelligent System for Travelling for Taiwan Innovative Computing Information and Control, 2006

**14.** K. Ogawa, Y. Sugimoto, K. Naito, T. Hishida, T. Mizuno, “Basic design of a sightseeing recommendation system using Characteristic Words”, IPSJ SIG technical reports 2014-MBL-71(14), 16, 2014-05-08.

**15.** W. Souffriau, P. Vansteenwegen, J. Vertommen, G. V. Berghe, and D. V. Oudheusden. a Personalized Tourist Trip Design Algorithm for Mobile Tourist Guides. Applied Artificial Intelligence, 22(10):964- 985, Oct. 2008.