iTimer
Count On Your Time

Mrs.Sanjeetha R.
Assistant Professor
Dept. of CSE
Ramaiah Institue of Technology
Bangalore, India
sanjeetha.r@msrit.edu

Ankit Goyal
Student
Dept. of CSE
Ramaiah Institue of technology Bangalore, India
ankitgoyal1012@gmail.com

Biswajit Kumar
Student
Dept. of CSE
Ramaiah Institue of technology Bangalore, India
bksinha4497@gmail.com

Abhinav Priyadarshi
Student
Dept. of CSE
Ramaiah Institute of Technology Bangalore, India
rush2rain@gmail.com

Abstract - In current lifestyle people tend to waste a lot of time in doing unnecessary activities like watching movies, playing video games, window shopping, etc. Many a times they are not aware of the cumulative time they spend wastefully. They can use their time constructively if they were informed about it through a mobile app. In our project we aim to develop an app which helps in monitoring the amount of time spent in such activities at various locations and inform the user about the same through visual display on Google maps. This app displays the summary of the locations visited during the day along with timestamps for better analyzing of data.

Keywords – *GPS; Internet; Location; Geocoding; Database; Time; Google Api; LBS; Maps;*

*****

I. INTRODUCTION

This application aims to provide an interface for the user by which the user can track his location with the duration of time spent at various locations through which the user can get to know cumulative amount of time wasted or used productively. The locations can be displayed both on map as well as in pie chart format with the selected date respectively. The update interval of coordinates is based on the distance between the consecutive locations and the time intervals, also there are notifications available for GPS service and network information.

II. LITERATURE SURVEY

Location-based Services

Location Based Services are applications depend on the location of the user ‘s device. There are two types of LBS: the closed system and the open system. The closed LBS system does not have the capability to transmit data. One example of the closed LBS system is the GPS device, which receives information from the satellite with the application of the GIS system in the device itself. The other type, open LBS system, has the capability to transmit data [1]. Compared to the closed LBS system, the open LBS system shows more advantages by operating with platforms. During the interactive processes, the latest information may be sent to users. The feasibility of value-added service in the closed system LBS is not high. In addition, for mobile network operators, location-based services represent an additional stream of revenue that can be generated from their investments in a fixed infrastructure. For the end user, these services can help reduce confusion, improve the consumption experience, and deliver high-quality service options. However, the open LBS system can be applied in various devices. Hence, the present study focuses on mobile devices (cellular phones), rather than discussing every kind of device (i.e., RFID, GPS navigation device, and so on.). The open LBS system, which focuses on mobile devices, is called MLBS [2].

SQLite

SQLite is an embedded SQL database engine. Unlike most other SQL databases, SQLite does not have a separate server process. The database file format is cross-platform - you can freely copy a database between 32-bit and 64-bit systems or between big-endian and little-endian architectures. These features make SQLite a popular choice as an Application File Format.

In our application SQLite database is being used for storing attributes like latitude, longitudes, time instances and location Id's which are later extracted and normalized for analysis purposes [3].

Geocoder

Geocoding is the process of transforming a description of a location—such as a pair of coordinates, an address, or a name of a place—to a location on the earth's surface. You can geocode by entering one location description at a time or by providing many of them at once in a table. The resulting

Locations are output as geographic features with attributes, which can be used for mapping or spatial analysis [4].

With geocoded addresses, you can spatially display the address locations and begin to recognize patterns within the information. This can be done by simply looking at the information or by using some of the analysis tools available with ArcGIS (Geographic information system for working with maps and geographic information). You can also display your address information based on certain parameters, allowing you to further analyze the information [5].

Google Maps API

The Google Maps JavaScript API has several types of overlays that you can add programmatically. Single locations on the map are displayed using **markers**. Markers may sometimes display custom icon images, in which case they are usually referred to as "icons." An **info window** is a special kind of overlay for displaying content (usually text or images) within a popup balloon at a given location on a map. Lines on the map are displayed using **polylines** representing an ordered sequence of locations. Areas of arbitrary shape on the map are displayed using **polygons**. Like polylines, polygons are an ordered sequence of locations. Unlike polylines, polygons define a region which they enclose. You can also define circles and rectangles on the map. Use a **symbol** to customize the icon on a marker or add images to a polyline. A symbol is a vector-based image defined by a path, using SVG path notation. The API also provides options to control how the symbol will be displayed. If you want to place an image on a map, you can use a **ground overlay**. You may also implement your own **custom overlays** by implementing the Overlay [6].

III. PROPOSED METHODOLOGY

Initially raw location data is obtained from the Google location services. Then the data is stored in SQLite database in the form of predefined location id's. Further stored data is normalized in the proximity of 30 meters which gives the location of the user. Then locations are indicated on Google maps with the help of Google markers. Later data is plotted on graphs to analyze activities of user over the whole day.

IV. SYSTEM OVERVIEW

There are five major modules which help in processing the entire application data. **Main Activity** where all the other activities are controlled from this activity. This activity shows the stored locations in My SQLite database with predefined ids and also latitudes and longitudes for the new location is shown in this activity. All the activity calls are done from this activity with the help of intents particular to specific activity. **Database Helper**, here the database is created and it contains all the attributes of My SQLite database like on Create instance will create the rows and columns required for storing the location characteristics like time spent at particular location and the address of that place.

GPS Service, here the location components like latitudes and longitudes are determined with the help of Geocoder. Date and Time of the location is also extracted in this activity and after this location details are passed to database helper activity to store in the SQLite database. Implements a background service which periodically refreshes our location in the remote database against our different locations.

Maps Activity initializes the Map View which is the primary view being used in our application. In this Activity Map View implements the Google APIs for maps. It implements toggling between Street View and Satellite View for maps. It also performs addition of compass, controller to animate to location and zoom in/out. It implements extraction of the address from the particular coordinates and can place a pinpoint at that particular position &removal of all pin points. Implement AsyncTasks for (execution on a separate thread so as not to overload the main thread) getting the JSON encoded location of user though his/her location id, to create the user location in the remote database if the app is being used for the first time &to update the same user data in the remote database.

Marker Information is an info window displays text or images in a popup window above the map. Info windows are always anchored to a marker. Their default behaviour is to display when the marker is tapped. It is added to set the title () and snippet () of the corresponding marker. Setting these properties will cause an info window to appear whenever that marker is clicked.

Algorithm – Location Retriever

Input: coordinates of a location

Output: classification of locations

```
    Start service=Initialize modules
    If(GPS=OFF)
```

```
        notify(user);
      Else if(internet=off)
                notify user
      Else
        check for proximity () {

                getcoordinates ();

                geocode(getaddress)
       }
        call database_helper ()
      {
                if (user_choice==showmap ())
      {
                        datepicker ();
                        display(location);
      }
      else if
       {
      user_choice=graph_display;
              datepicker ();

              GraphDisplay (PieChart);
      }
}
```

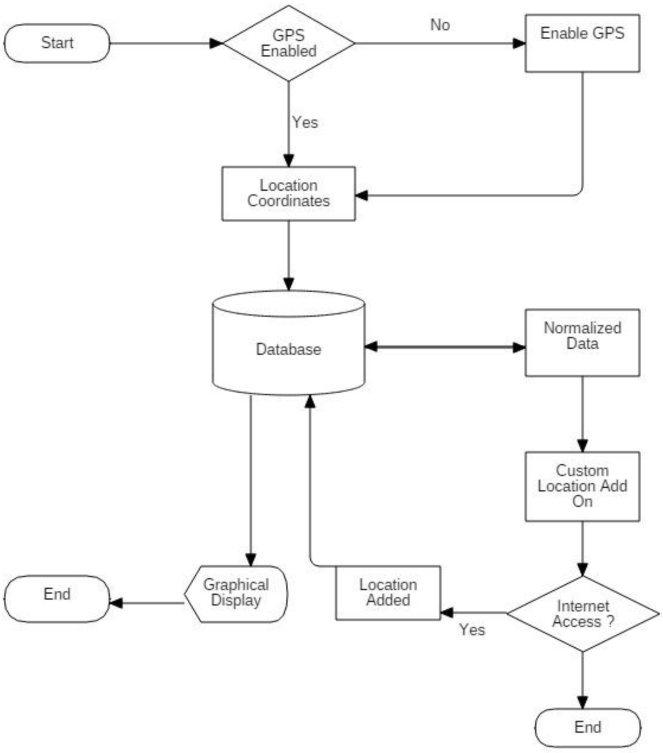


Figure 1: Data Flow Diagram

V. Conclusion

The idea of developing a location-based Android app facilitating constructive time management has been achieved. The app aims at acquainting the user time spent at various locations with graphical analysis. Future scope includes predicting continuous app usage based on past user involvement and notifying the same. Reminders can be displayed based on user's daily activity. App sending locations of children to their parents at regular intervals can also be introduced.

VI. References

[1]. X. C. Mei, "**The design and implementation of LBS system based on J2EE**", pp. 6-10, 2004

[2]. developer.google.com,"**LocationStrategies**",2018.[Online].Available:https://developer.android.com/guide/topics/location/strategies.html. [Accessed: 18- Mar- 2018].

[3]. developer.android.com,"**SQLite**",2017.[Online]. Availble: https://developer.android.com/training/data-storage/sqlite. [Accessed: 18- Mar- 2018].

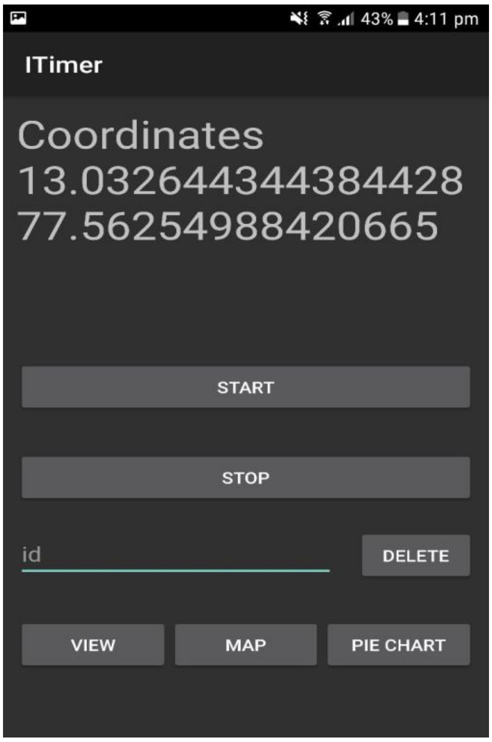
[4]. Bonner MR, Han D, Nie J, Rogerson P, Vena JE, and Freudenheim JL 2003, **"Positional Accuracy of Geocoded",***2012.*

[5]. R. Bakshi et al., **"Exploiting online sources to accurately geocode addresses"**, *ACM-Gis*,pp. 194203, 2004.

[6]. developer.android.com,"**Geocoder**",2018.[Online].Available: http://developer.android.com/reference/android/location/Geocoder.html.[Accessed:17- Mar-2018].

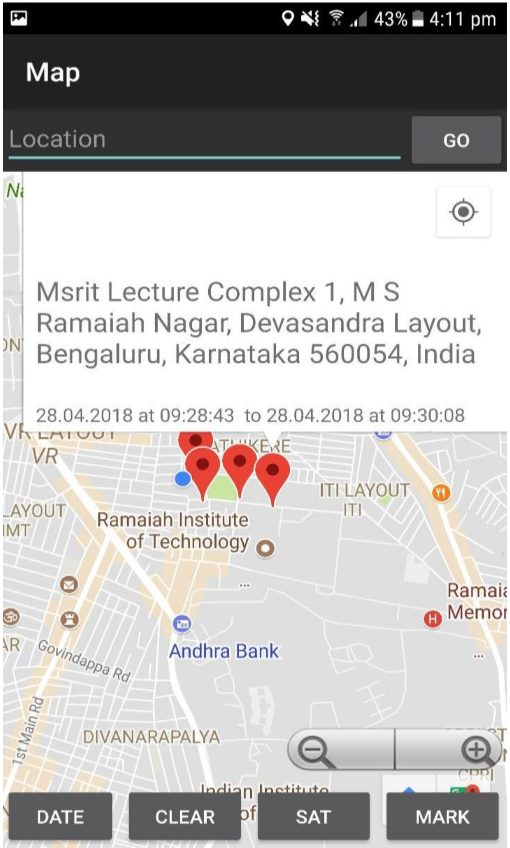
[7]. developer.google.com,"**MapsAPI**",2018.[Online].Available:https://developers.google.com/maps/ documentation/android-api. [Accessed: 18- Mar- 2018].

VII. Result Snapshots



Coordinates
13.032644344384428
77.56254988420665

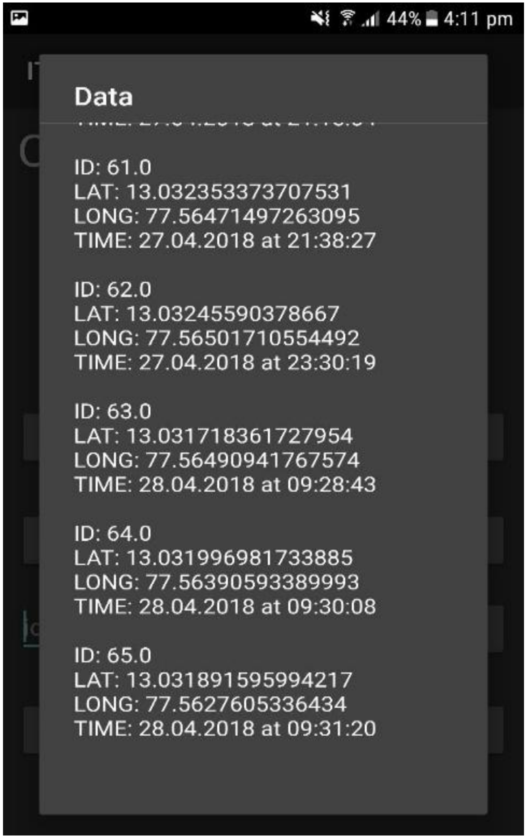
Figure 2: Home Page



Msrit Lecture Complex 1, M S Ramaiah Nagar, Devasandra Layout, Bengaluru, Karnataka 560054, India

28.04.2018 at 09:28:43 to 28.04.2018 at 09:30:08

Figure 3: Maps Display



ID: 61.0
LAT: 13.032353373707531
LONG: 77.56471497263095
TIME: 27.04.2018 at 21:38:27

ID: 62.0
LAT: 13.03245590378667
LONG: 77.56501710554492
TIME: 27.04.2018 at 23:30:19

ID: 63.0
LAT: 13.031718361727954
LONG: 77.56490941767574
TIME: 28.04.2018 at 09:28:43

ID: 64.0
LAT: 13.031996981733885
LONG: 77.56390593389993
TIME: 28.04.2018 at 09:30:08

ID: 65.0
LAT: 13.031891595994217
LONG: 77.5627605336434
TIME: 28.04.2018 at 09:31:20

Figure 5: Coordinate data

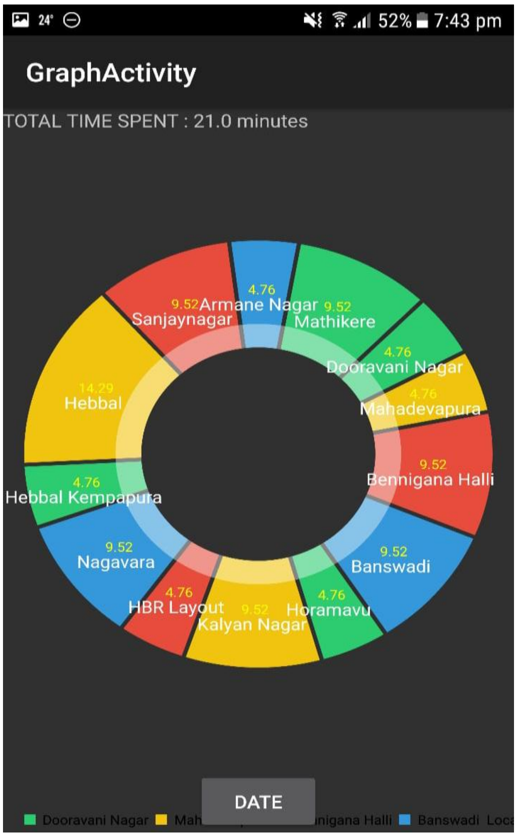


Figure 4: Pie Chart depicting time spent at various location