Batroid: Android Personal Assistant

CSE 535: Mobile Computing, Project Report

Abhishek Zambre
ASU ID 1210933864
MCS, CIDSE
Ira A. Fulton Schools of Engineering
Arizona State University
abhishek.zambre@asu.edu

Mayank Khullar
ASU ID 1210873882
MCS, CIDSE
Ira A. Fulton Schools of Engineering
Arizona State University
mkhullar@asu.edu

Chaitanya Phalak
ASU ID 1211203003
MCS, CIDSE
Ira A. Fulton Schools of Engineering
Arizona State University
cphalak@asu.edu

Murali Ravivarma
ASU ID 1211230706
MCS, CIDSE
Ira A. Fulton Schools of Engineering
Arizona State University
mravivar@asu.edu

***Abstract* – This document is a project report for Android-based Personal Assistant application called "Batroid". It contains detailed description of application architecture and the functionalities of the mentioned system.**

I. Introduction

Today's computing has rapidly grown from being confined to a single location. With mobile computing, people can work from the comfort of any location they wish to as long as the connection and the security concerns are properly factored. In the same light, the presence of high speed connections has also promoted the use of mobile computing.

Being an ever growing and emerging technology, mobile computing will continue to be a core service in Computing, and Information and Communications Technology. These continuing evolutions in this field caused emergence of many intelligent software which resides in the mobile phone itself and acts as a personal assistant for the user. Keeping this in mind, we developed our own Android-based Personal Assistant, called "Batroid", which has its own unique functionalities.

Batroid is a software agent that can perform tasks or services for an individual. These tasks or services are based on user input, location awareness, and the ability to access information from a variety of online sources (such as weather conditions, calendar schedules, etc.).

II. System Design

A. Product Perspective

Batroid has been designed to keep the ease of access for a novice user in mind. It is a based on material design with minimalistic Graphical User Interface (GUI) layout. Once application is configured, it requires as less as possible user intervention. We have also designed application in such a way that it will consume very less memory, by sharing data (such as GPS location) among its components.

B. Product Architecture

The application is designed using material design principles. It requires permission for User Location, Read SMS, Managing Calls, Access to Calendar Events, Access to Contacts and Storage Write Access. Location service gets triggered when application launches for the first time. It later broadcasts GPS coordinates to other components such as Google Maps fragment.

MainActivity component has Drawer Menu, with the relevant functionalities for each menu item.

Various types of notifications such as weather conditions, route information etc. will be provided by the application.

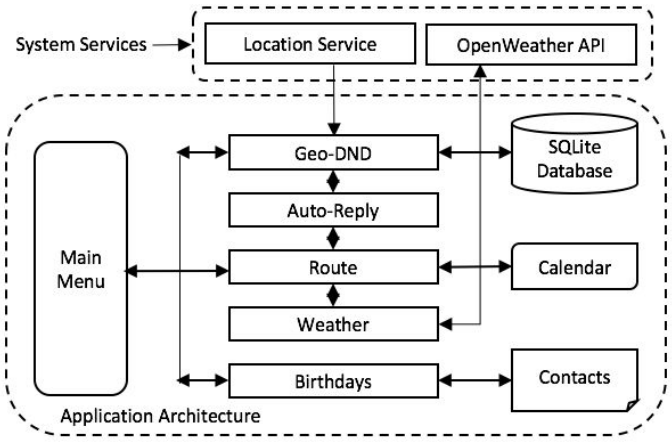

Figure 1. Application Architecture

III. Main Activity User Interface

When launching application for the first time, user will be greeted with Main Activity user interface. This interface has

drawer menu which helps user configure and interact with the application.

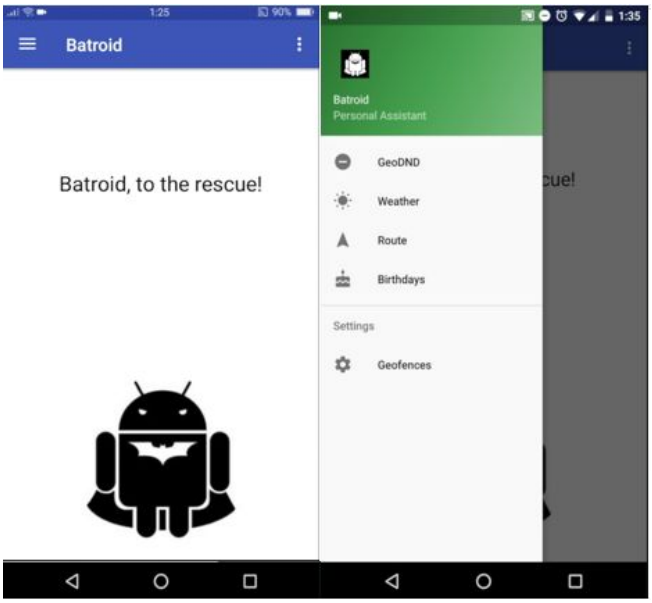

Figure 2. Main Activity User Interface

IV. Geo-DND (Geofence-based DND)

Many a time we don't want to get disturbed whenever we are in meeting or in a class. Every time we need to put phone into DND i.e. Do-Not-Disturb mode manually, and again manually have to turn-off the DND mode. What if there is a smart application that will perform these operations on our behalf?

Geo-DND is such a feature which activates DND mode, whenever user enters pre-defined location, and turn-off DND mode whenever user exits that location.

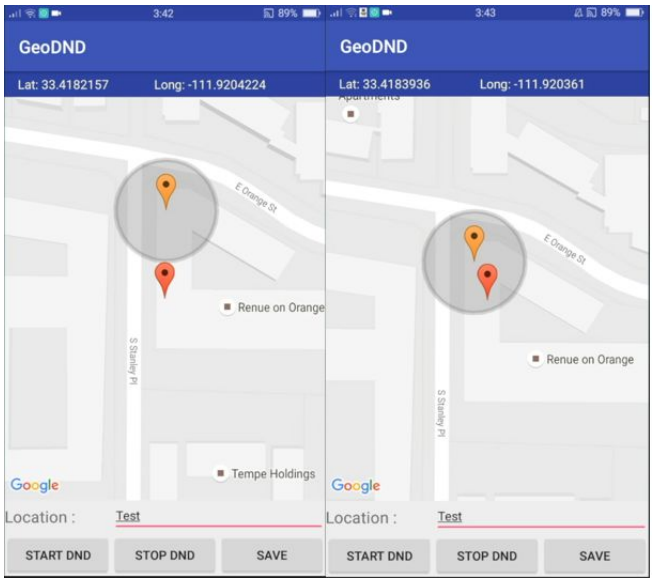

Figure 3. Geo-DND Activation

For example, at the start, user configures Library coordinates on map as a Geo-DND location. Now, whenever user enters pre-defined radius of that location, application will broadcast "Enter" event, and corresponding services will perform DND-related operations based on that event (such as putting phone into silent mode). Usually, all these operations will only be related to Do-Not-Disturb activation. Similarly, whenever user exits that location, application will trigger "Exit" event, and related services will perform activities to turn-off DND mode.

In order to implement this feature, we have used Google Maps API and Google Geofencing API. Below flow chart describes how Geo-DND works.

V. Auto-Reply

Users do get messages and phone calls all through the day. The sender will always expect a response from the user regardless of the situation at the user's end.

This feature is just the thing users are looking for. When a user receives a text in DND mode, automatically a text message will be sent from his phone saying he is busy at the moment and he will be responding to you as soon as possible. Similarly, when he receives a call in DND mode, call will be aborted and immediately a text will be sent automatically.

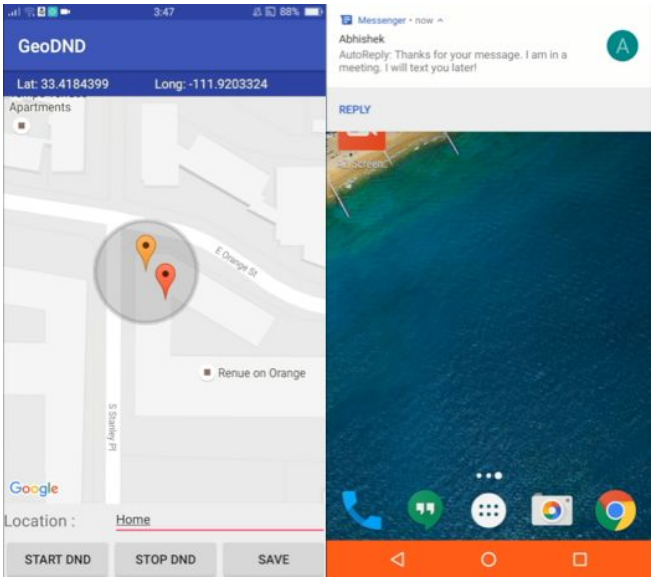

Figure 4. Auto-reply in Geo-DND mode

Misc. details.

VI. Geo-DND Database

The Geo-DND database stores the DND zones enabling the user to create a DND zone with just a click. The user is allowed to delete the DND zones.

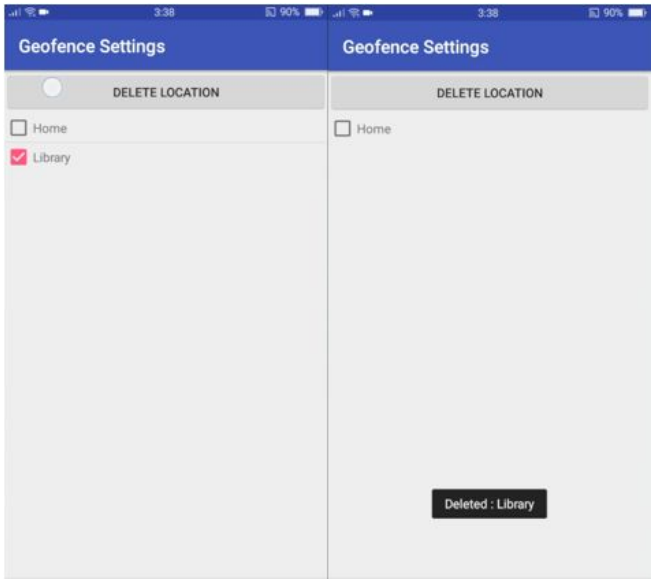

Figure 5. Geo-DND saved locations deletion

The database stores upto 100 DND zones per user.

VII. Weather

The weather API that is used for this project is openweathermap. The application pools the data every one minute to gather the GPS and weather data and if there is any event of rain or thunderstorm or snow, the applications gives the notification to the user for the event. The refresh button enable user to get the current weather data at any point of time irrespective of the one minute cycle of the application.

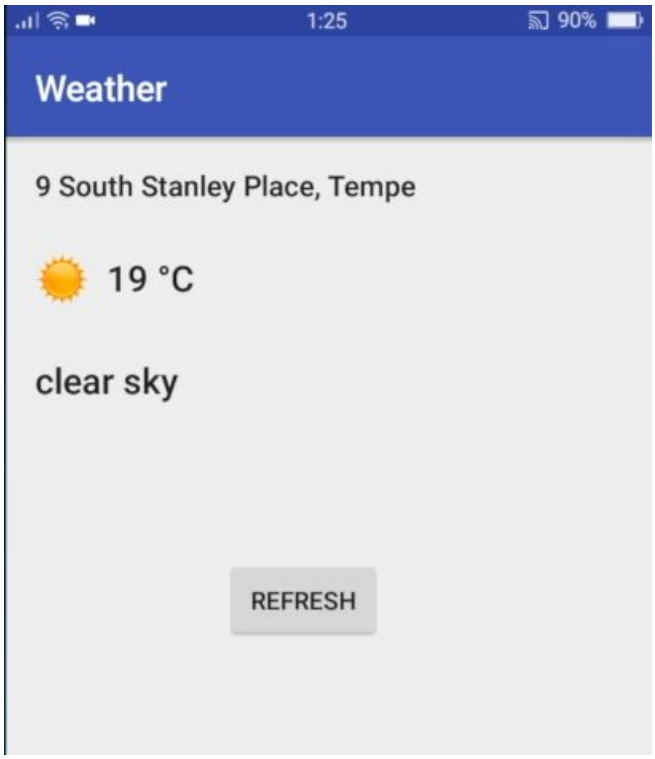

Figure 6. Weather Updates

VIII. Route Suggestions

We use our calendar on phones to store information about daily meetings, important events and commitments. Instead of looking up the information in calendar daily and calculating the route to an important meeting, we can automate this process by using the smart application that does it for the user.

Route suggestion is such a feature that acquires the user's current location and stored calendar the information to suggest a route to an important meeting and also showing the distance to the place where meeting is going to happen and the time required to travel to the place.

This feature is implemented considering two aspects i.e. getting calendar information and showing route suggestion. First, the calendar information for the present day is

acquired using the calendar provider library. The events which are after the current time are considered. The route is shown to the first event after the current time in the calendar. The location information of the event is also extracted. The route suggestion feature is implemented using Google Maps Android API. The API is queried for the route between the current location and the location where the event will happen. The API returns a JSON which is parsed to get the route information. Also the time required for travel between the two locations and the distance between the two locations is obtained. The route is plotted using polyline and the time and distance information is shown in the activity.

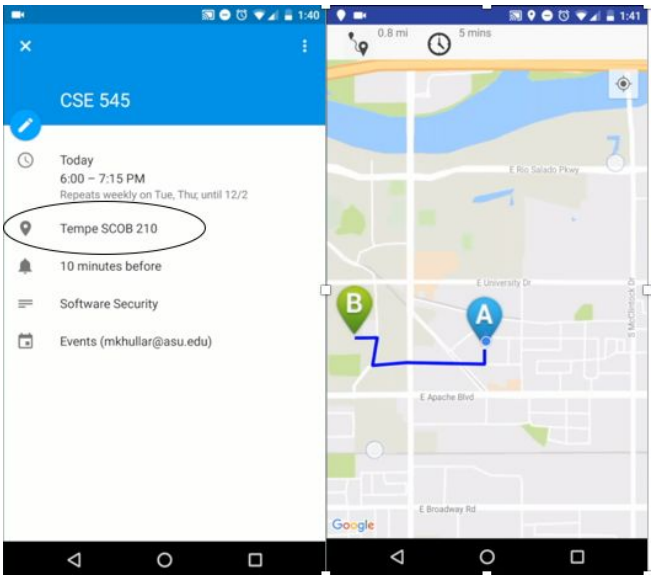

Figure 7. Route Suggestions

For example, the user has an event at SCOB 210 Tempe at 6 PM. Route is shown to the user from his current location to the location of the event. Also, distance between the two locations and time required to travel between the locations is shown on the activity.

VII. Birthday Texts

Time once gone is gone forever. This saying is very much applicable while wishing for birthdays. Users might know their friends, relatives and other near and dear ones' birth anniversary off the top of their head, but forgets to wish them on their D-day. Birthday Text feature in Batroid assists users to send wishes on their day without fail.

In this feature, we parse the phone contact list and retrieve the list of names along with their phone number whose birthday falls on the current day. We store these details in the buffer. When user clicks the birthday tab on the main UI (Figure 2. Main Activity User Interface), the list of names and numbers from the buffer are displayed on the UI. Batroid provides a one-click SEND button which enables the user to send wishes in one-go to all the buddies (Figure 8. Birthday Texts).

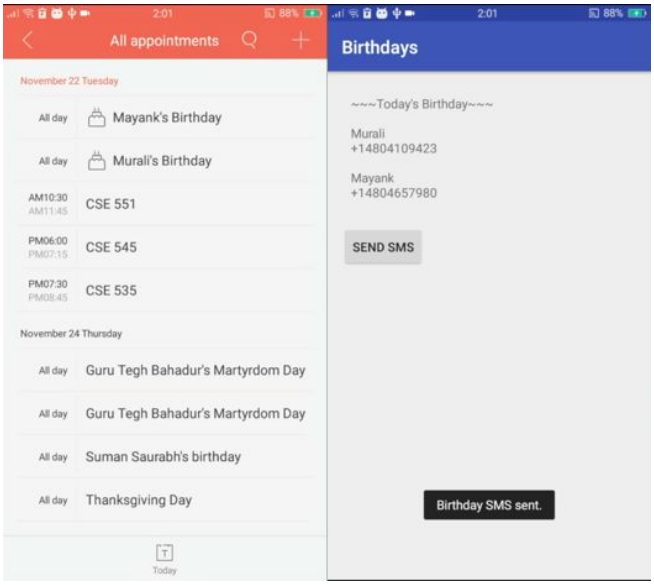

Figure 8. Birthday Texts

VIII. Task Completion

Task Number	Task	Task Assigned To
1	Route suggestion - Specification	Chaitanya Phalak
2	Calendar and Date/Time - Acquisition	Chaitanya Phalak
3	Route suggestion - Action	Chaitanya Phalak
4	Route Notification - Delivery And Reception	Chaitanya Phalak
5	Weather suggestion - Specification	Chaitanya Phalak
6	Location using GPS - Acquisition	Mayank Khullar

7	Notification Of weather - Action	Mayank Khullar
8	Auto Reply to text messages - Specification	Mayank Khullar
9	Location using GPS for Call rejection- Acquisition	Mayank Khullar
10	Auto Reply to text messages - Action	Mayank Khullar
11	Auto Reply to text messages - Delivery And Reception	Abhishek Zambre
12	Do Not Disturb mode - Specification	Abhishek Zambre
13	Location using GPS for Geo-DND - Acquisition	Abhishek Zambre
14	Do Not Disturb - Action	Abhishek Zambre
15	Do Not Disturb - Delivery And Reception	Abhishek Zambre
16	Do Not Disturb - Acquisition	Murali Ravivarma
17	Happy Birthday - Specification	Murali Ravivarma
18	Happy Birthday - Delivery and Reception	Murali Ravivarma
19	Happy Birthday - Acquisition	Murali Ravivarma
20	Happy Birthday - Action	Murali Ravivarma

IX. Conclusion

Batroid is an Android application making day to day life easy. Batroid comes with the features such as Geo-DND, weather updates, Route Suggestions for events, call rejection and Autoreply for DND zone. birthday text messages. These feature covers almost everything that a user needs in day to day life.

X.Acknowledgment

Implementation of route suggestion feature uses code from github libraries namely, Googlemapdirectionsimple[2] and Calendarprovider-Lib[4].

XI. References

[1] "Basic Google Maps API Android Tutorial + Google Maps Directions API". YouTube. N.p., 2016. Web. 5 Dec. 2016.
[2] "Hiepxuan2008/Googlemapdirectionsimple". GitHub. N.p., 2016. Web. 5 Dec. 2016.
[3] "Google Maps Android API | Google Developers". Google Developers. N.p., 2016. Web. 5 Dec. 2016.
[4] "Macisamuele/Calendarprovider-Lib". GitHub. N.p., 2016. Web. 5 Dec. 2016.