CSE 535 – Final Project Report – Group 22

Amit Kumar	Subhadarshi Samal	Sneha Manjunatha	Lakshmisagar Hiragappa Kusnoor

Arizona State University
akuma152@asu.edu, ssamal1@asu.edu, smanju10@asu.edu, lkusnoor@asu.edu
ASU ID:1211257070, 1210421105, 1211193968, 1211009498

ABSTRACT

"Mobile will ultimately be the way you provision most of your services. The way I like to put it is, the answer should always be mobile first." by Eric Schmidt clearly states that smartphones are everyone's' first "go to" in any scenario. They are at par with computers. They have may inbuilt features and sensors that help make life easy. With the integration of these devices into the mobile system comes more ideas and challenges to develop something new. Developing applications which help make a consumer's life easy and have a great user experience is one of the major goals of mobile computing. An assistant is one that helps you be on your schedule. The goal of our project is to build a context- aware application smart and adaptive application which helps a user reach on time without missing out on his daily habits.

Author Keywords

Context-aware, adaptive, application, machine learning, assist, location aware

INTRODUCTION

On Time is a smart and adaptive software application that helps by notifying the user his/her schedule by tracking his/her whereabouts and providing him/her with routes based on his/her preferences. Consider a scenario in which a user has a class scheduled at 3 PM in the afternoon, the application based on the data collected over a week predicts the time taken to reach the class and sets the alarm to ensure the user reaches class on time. On reaching the class the app automatically changes the mode of the phone to silent. The application also predicts the path based on the user preferences (like have coffee). This application uses the GPS sensor to locate the position and machine learning algorithms to predict the whereabouts of the user and his preferences.

CHALLENGES AND CONSIDERATIONS

Data Collection: The initial data collection is fed to the SVM for creating the pattern. So it is critically important that the initial data collection should be appropriate and error free.

Deciding Factors to Train SVM: Support Vector Machines are based on the concept of decision planes that define decision boundaries. When training an SVM, the practitioner needs to make a number of decisions: how to preprocess the data, setting parameters of the SVM etc. Uninformed choices may result in severely reduced performance

Assumption: Home and Away: The source and destination location is fed to the svm. An event is created based on source and destination location.

IMPLEMENTATION

Login Page: It implements the Google sign-in. We have used the Google OAuth API to verify user through their gmail account.

Event Creation: Start button is used to record the user's current location and stop button redirects the user to add in details of the event like class name, class time, preference, etc. For example if the user starts from his home, the location of his home is recorded as the source on clicking the start button and after reaching the class he clicks on the stop button, the location is recorded as the destination and other details are recorded using an event form. We have used Google Location Manager API, this service allows application to obtain periodic updates of geographic location. The data collected over a week is used to train the SVM classifier. The map also considers user's preference for coffee while coming to class over previous weeks. The coffee shops are showed in the user's route map to the destination.

Alarm Interface: We have used the Android Alarm Manager Service to set the time for the notification. Based on the previous data the alarm time is adjusted, like if the user was late the previous time, the alarm time will be increased, which will allow him to reach on time in future. If he reaches early the alarm time will not be changed.Next, when the alarm rings, a notification is displayed on the notification bar. On clicking the notification the user is redirected to the application from where the alarm can be stopped. The user can then click the 'show map' button.

Google Maps: This is used to display the shortest route to the class with/without the user preference. Markers are used to show the source (green), destination(red) and the location of the user preference (blue). Waypoints are used to show the path the user has to take.

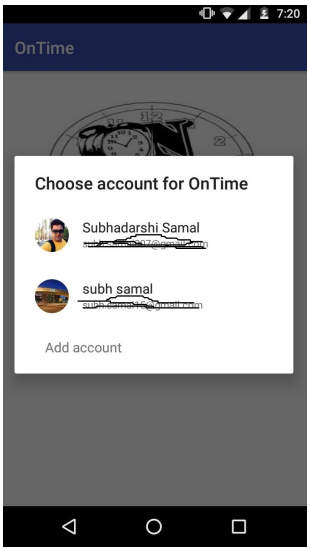
Mode Change: This feature is used to change the mode of the cell phone. The mode changes once the user reaches his destination. Once we reach inside the classroom, the phone goes into the silent mode and vice versa.

Context-Aware API: The following Google APIs are used to

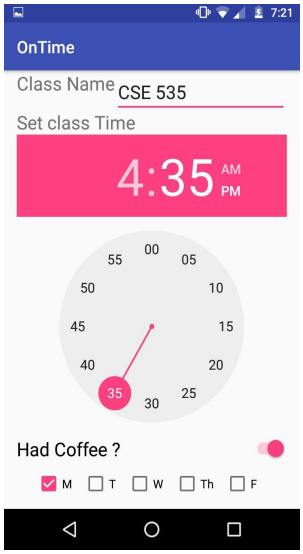
collect user's context information in our project.

1. **Location Service** - Google location service is used to fetch user's current location details in terms of latitude and longitude.
2. **Oauth API** - The oauth service which is used to authorize a user follows two prime steps. First, it generates the login url and then then it exchanges the login code with the user's application. An *accesstoken or refresh*token are specifically generated for your application. This means that you can't use these tokens to access a user account with another application or for different services.

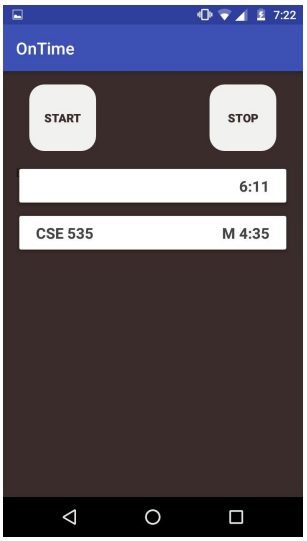
SCREEN SHOTS



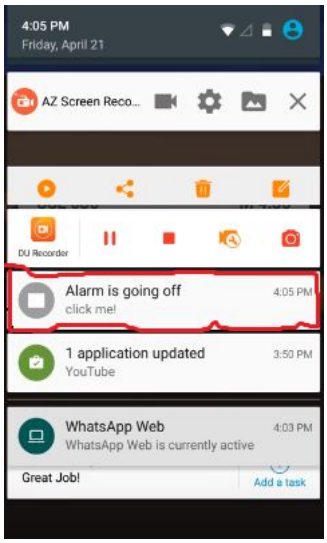
Signing in using oauth



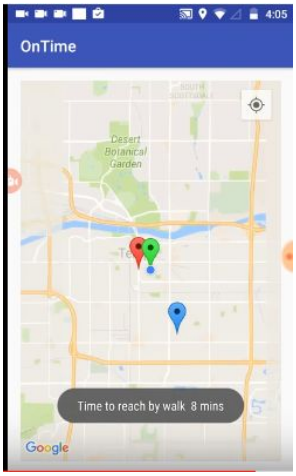
Event Creation



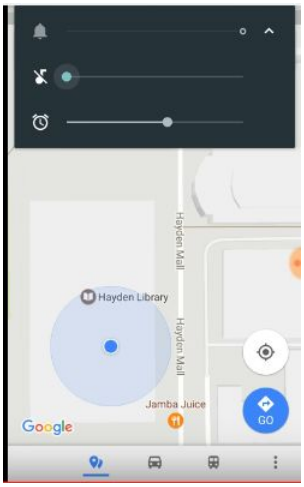
Created Event



Alarm Notification



map option



phone changes to silent mode

TASK COMPLETION

Sl no	Tasks	Assignee
1	Content Providers to store class timing	Amit
2	Activity to store class	Lakshmisagar
3	Login and Icon design	Sneha
4	Configuration changes handling	Subhadarshi
5	Activity of stored user preferences.	Lakshmisagar
6	Learning implementation to modify user preference of intermediate stops.	Amit
7	Material UI implementation	Sneha
8	Material design for alarm clock	Subhadarshi
9	Service to do learning	Amit
10	Service to change mobile on silent mode	Lakshmisagar
11	Functionality to change phone back to the normal mode/ ring mode once the class is over	Amit
12	UI implementation (via XML)	Lakshmisagar
13	UI design for alarm class	Sneha
14	Predicting shortest path based on current location	Sneha
15	SQLite Database Helper implementation	Subhadarshi
16	Learning implementation to modify the preference if user is reaching to class early or late.	Subhadarshi
17	UI handling for various models	Lakshmisagar
18	Adapter and Listview	Amit

	implementation	
19	Alarm class implementation to alert student regarding the class	Sneha
20	Maintain user profiles based on login information	Subhadarshi

Table 1. Task Completion

All tasks shown in Table 1 have been completed 100%.

DATABASE CREATION

We have used two databases for our application; one for to record the details of the event and another one is to analyse the user preference. The schema of the two databases are as follows:

1. *Event Data Collection*: source location, destination location, class name, class time.

These parameters were used to train the SVM to predict the path with/without the user preference to reach the class on time.

2. *Preference Analysis Table*: preference (whether the user wants coffee or no), location (if he is at home or away) and time (morning, noon and evening)

These parameters were used in the Naive Bayes prediction algorithm to decide the preference of the user.

CONCLUSION

In this project we have implemented a location and context aware application that helps the user maintain his schedule and the application tracks his whereabouts and preferences. By doing this the application notifies the user in advance to give the user ample time to reach his destination on time. It also changes the sound settings of the phone based on location and time. We have utilized the Google APIs such as Maps and OAuth to help authenticate the user and to process the user location.

FUTURE WORK

The application can be linked with google calendar to automatically update the database or create events. Other machine learning algorithms can also be used on the database to compare the accuracy of training and classification. We can also extend the project to include other factors like traffic, mode of transport, etc to predict the way prediction of the user from the source to destination.

ACKNOWLEDGMENTS

We would like to thank Dr. Ayan Banerjee for giving us the opportunity to develop this application as part of our coursework and for providing us with a well-rounded curriculum which explained various domains of a mobile system.

REFERENCES

1. Google API library, Retrieved April 1, 2017 from https://console.developers.google.com
2. Google Maps, Retrieved April 1, 2017 from https://developers.google.com/maps/
3. Android Developer, Retrieved April 1, 2017 from https://developer.android.com
4. StackOverflow. 2006. StackOverflow developers forum (2006). https://stackoverflow.com.