NAVIGATION OF UNIVERSITY ON ANDROID PLATFORM

Parth Khunteta , Vaibhav Pandey ,Subham kumar, Jayesh Lalwani

B.Divager(Assistant Professor)

[1] *Under Graduated Student, Computer Science Engineering, SRM University, Tamil Nadu, India*
[2] *Under Graduated Student, Computer Science Engineering, SRM University, Tamil Nadu, India*
[3] *Under Graduated Student, Computer Science Engineering, SRM University, Tamil Nadu, India*
[4] *Under Graduated Student, Computer Science Engineering, SRM University, Tamil Nadu, India*

ABSTRACT

Nowadays, location based technologies are widely used due to their profound benefits and affordances in real life applications. College campuses can be large, confusing, and intimidating for new students andvisitors. Finding the campus may be easy using a GPS unit or Google Maps directions, but this is not the case when you are actually on the campus. There is no service that provides directional assistance for the campus itself. This thesis proposes a user assistant application running on an Android platform that can direct users to different buildings and parking lots in the campusCollege campuses can be large, confusing, and intimidating for new students and visitors. Finding the campus may be easy using a GPS unit or Google Maps directions, but this changes when you are actually on the campus. In this the navigation system is divided into two parts the inside campus and the outside campus. The College fresher students can see the stores, medical shops, edible shops, shopping mall, places outside or inside the campus in or over 10 km of radius. There is no service that provides directional assistance within the campus itself. In this the contacts of higher officials has also provided for the betterment of the campus. This paper presents the architecture and design specifications for a campus assistant application on an Android platform.

Keywords: *campus assistant, android platform, campus map, Google map, campus navigation, navigation view, campus navigator*

1. INTRODUCTION AND RELATED WORKS

Recent technological advancements have gain popularity finding many applications in everyday activities. There are many devices and different applications that focus on directing the user to desired locations. Today's drivers are well equipped for travel thanks to the GPS units many have in their cars. GPS applications allow users to enter a destination and based on their current coordinates display the fastest way to the destination .
Additional features have evolved over time, such as displaying congested routes, which allow users to make smart driving decisions and improve driving safety. This saves time and stress when going to unfamiliar places or taking long trips. Google Maps Navigation is a mobile application developed by Google forth Android anlater integrated into the Google Maps mobile app. The application uses an Internet connection to a GPS navigation system to provide turn-by-turn voice-guided instructions on how to arrive at a given destination. The application requires connection to Internet data (e.g. 3G, 4G, WiFi, etc.) and normally uses a GPS satellite connection to determine its location. A user can enter a destination into the application, which will plot a path to it. The app displays the user's progress along the route and issues instructions for each turn.

Google Maps and GPS systems have become indispensable in recent years, with vast amounts of users relying on them for directions but their capabilities have not yet been fully applied to university campuses.

Directions within campuses are not available using the Google maps application. So from these we are providing the directions of the every block, parking, library, etc. This application is provides the navigation inside the campus because inside the campus goggle maps does not gives the direction .if anyone is having difficulty in finding the surroundings things in the campus can be able to use this android application.

One of the most popular Smartphone platforms is Android , which is a Linux-based operating sys-tem designed primarily for touch screen mobile devices. Google has released its code as open source, triggering a large community of developers to write applications that extend the functionality of the device using a cus-tomized version of Java. Due to its superior technological capabilities, we propose to develop the campus assistant application on the Android platform.

1.1 Motivation and Problem Statement

Google Maps and GPS have become very popular in recent years, with vast amounts of users relying on them for directions, but their capabilities have not yet been applied to university campuses. Directions within campuses are not available using the Google maps application. The campuses can be quite large and confusing. New students and staff, as well as visitors can have a very difficult time getting around. Even when they are asking for directions, they often time cannot find the destination because the directions involve knowing the surrounding buildings and landmarks of the campus. This can get quite stressful, especially considering students are often on a schedule and need to get to classes on time. Eliminating this stress and confusion would improve the overall atmosphere of the campus. Since smart phones are a ubiquitous technology nowadays, it makes sense to use them to resolve such issues. The problem that we address in this thesis is how to use the current advances in technology to provide a mechanism to facilitate navigation

1.2 Contribution

In this thesis, we design and develop a campus assistant application that runs on an Android platform and is intended to assist campus fresher's ,teachers and more other users that want to find directions to a certain building in the campus. The application provides directions to the destination building or to a parking lot, and this parking selection is based on the type of user (e.g. visitor, staff, and students). The computed path (e. g. the shortest-path) is displayed on the Smartphone to assist the user in reaching the destination Google has been involved with a few projects that have similarities to ours. The one that most resembles our project is the 2007 Google Street View Project. This project involves taking images using car, trike or bike mounted cameras and mapping unique areas such as parks, university campuses and malls . This allows users to interactively view the areas, seeing the location in 360°. While this technology is interesting, and it does allow very in debt mapping of the campuses, including pedestrian walkways as well as streets, it does not allow the user to enter their desired destination and have the shortest path to their destination be displayed on the map that is to be expected in walking time as shown ..

2. CAMPUS ASSISTANT APPLICATION ARCHITECTURE

In this section we present the architecture for the campus assistant application. We derive the following requirements for the system:

- The application is available on the Android plat-form.
- In this application the destination is set by default .you want to choose the place where you want to go adjust click on it .The navigation system will start and show the direction by the blue mark and expected time is shown. if you want to see the map of the block upon reaching there it is also available through the floor level wise.

- The application provides navigation capabilities based on the user type. A user enters the destination location and based on the current location a shortest driving or pedestrian path is displayed. If the destination is a building or Block, then based on the user type (e.g. student, staff, visitor) a path to a compatible path is provided. .
- This is particularly important in campuses, where there are different parking lots assigned to visitors, students, and staff.

- The application provides rerouting if the user de-parts from the projected path.

It presents the architecture of our campus assistant application. Since Google Maps does not provide any information on campus locations (e.g. buildings, parking lots), we have to build the map data structures that describe campus locations, roads, alleys, traffic signs, everything needed for directions and navigation. . These maps are stored on the server as XML files.

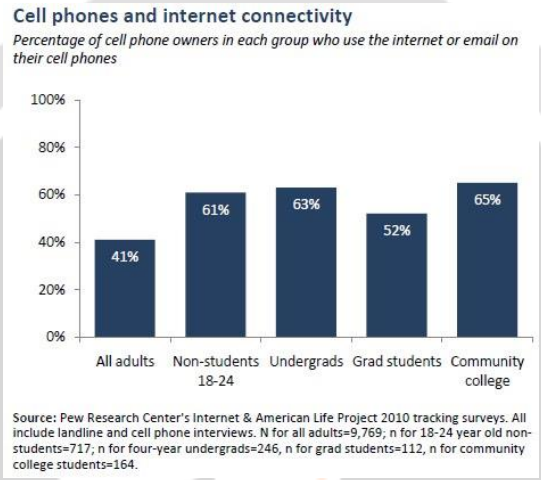

Chart -1: Survey

When the campus assistant app is deployed, the GoogleMap is used to keep the campus map up to date. Smartphone users automatically download the most recent version of a map and this simplifies maintenance. Map editing is useful when new locations (e.g. buildings) are being added to the campus or locations change their names.

User: load an existing campus map, edit map features (create/edit buildings, parking lots, road/alley segments, traffic signs), save a map to the Mapping Server as an XML file, display the map XML representation. Figure 2 shows a screenshot with a section of the FAU campus map loaded in the MapEditor, with icons for buildings ("B" markers), parking lots ("P" markers), walkable (green), and drivable (blue) segments. These map features will be discussed in section III.

The main component of our architecture is the Cam-pus Assistant smartphone app that runs on Android phones. The app user interface prompts the user to enter relevant information such as user type, campus map,

and destination location. The app provides both directions and walking/driving navigation capabilities while on campus. The app relies on a JSON/HTTP protocol to request an XML map file from the Mapping Server. This XML file is parsed and the map graph features are constructed. The smartphone GPS device is used to determine the user's current location or the user can indicate a source location. If the user wants driving directions, the app will direct the user to the compatible parking lot nearest to the intended destination.The app then computes the shortest path between source and destination, and displays it on the user's screen on top of google maps. Icons identify the user's current location. Smart phones have also made it easier for people who work on the move to stay in tune with constant email alerts and web browsing access. If this is not enough, there are endless applications to keep you entertained if you need it. From housewives needing to keep their kids occupied in doctor's offices, to businessmen waiting for another delayed flight, smart phone

applications have become irreplaceable. Many of these applications-including games, book libraries, video streaming, GPS navigation and many more- are free of charge. They are also simple to install and use.

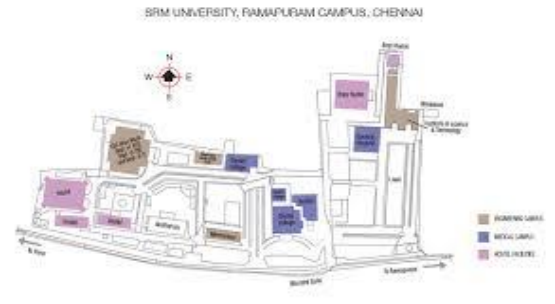

Fig -1: Campus Map

2.1 UML Diagram

When starting a project, an important step is to make a clear plan or outline to follow, such that each task is specified. This gives a clear jumping off point and keeps track of the project's progress. Creating such a model allows developers to lay out their thoughts and ideas about how to approach the project before the work begins, and prevents them from having to backtrack or start over due to confusion about what needs to be done. A model diagram ensures everyone working on the project knows how it will be done and allows for tasks to be divided clearly preventing overlapping of duties. The most accepted way of creating such a structured model is by using UML . UML is used not only for engineering application projects but it also can be applied to many different fields, such as business and architecture

The UML encompasses many different aspects. There are activity diagrams, use case diagrams, sequence diagrams, collaboration diagrams, and class diagrams . Activity diagrams represent the control flow in the system. Use case diagrams look at the external entities that interact with the system, those entities are actors and the interactions with the system are events. Sequence diagrams show timed sequence of object interaction. Collaboration diagrams show the links between objects. The diagram we are most interested in for our project is the class diagram.

A class diagram is basically a collection of objects in an object oriented application, which share common structure, behavior, relationships, and semantics . A class diagram shows these aspects visually. A class is represented by a rectangular icon which has a location for the name, the attributes, and the operations. The relationships, dependency, inheritance, composition, and association/aggregation are depicted by special symbols at the end of arrows connecting the classes.

There are quite a few different software programs available for designing UML diagrams. UMLet, Altova UModel, Visual Paradigm for UML, UML Designer. ArgoUML are just some of the software available for designing UML diagrams. Most of.these are free, some are open source, and they are often available as a plug in to an IDE such as Eclipse, or as a stand-alone program. For our UML diagram we decided to work with ArgoUML. It is a free open source, stand-alone, platform independent tool with a userfriendly interface and is very easy to install .

This class provides every subsequent decendent with an individual Id and is the basic, simplest unit of map data. This Id is provided by the map of the campus later on in our code. Inheriting directly from the Campus Map Object class are two different classes which create two branches in the class structure. These are the Vertex class and the Segment class. A vertex is basically an ending point on the map, such as a building, an intersection, a parking lot and so on, and has two boolean attributes that store information on whether that vertex is walkable and/or drivable.

A segment is connecting two vertices. The relationship between the Vertex class and

the Segment class is such that every vertex can have many segments, but every segment can have only two vertices (two endpoints). These classes in turn havetheir own subclasses. The Vertex class is a direct superclass

to the Segment End class, which in turn has a subclass itself, the Walk Segment End class. The Walk Segment End class sets the walkable variable to true, and also has a subclass, the Road Segment End class

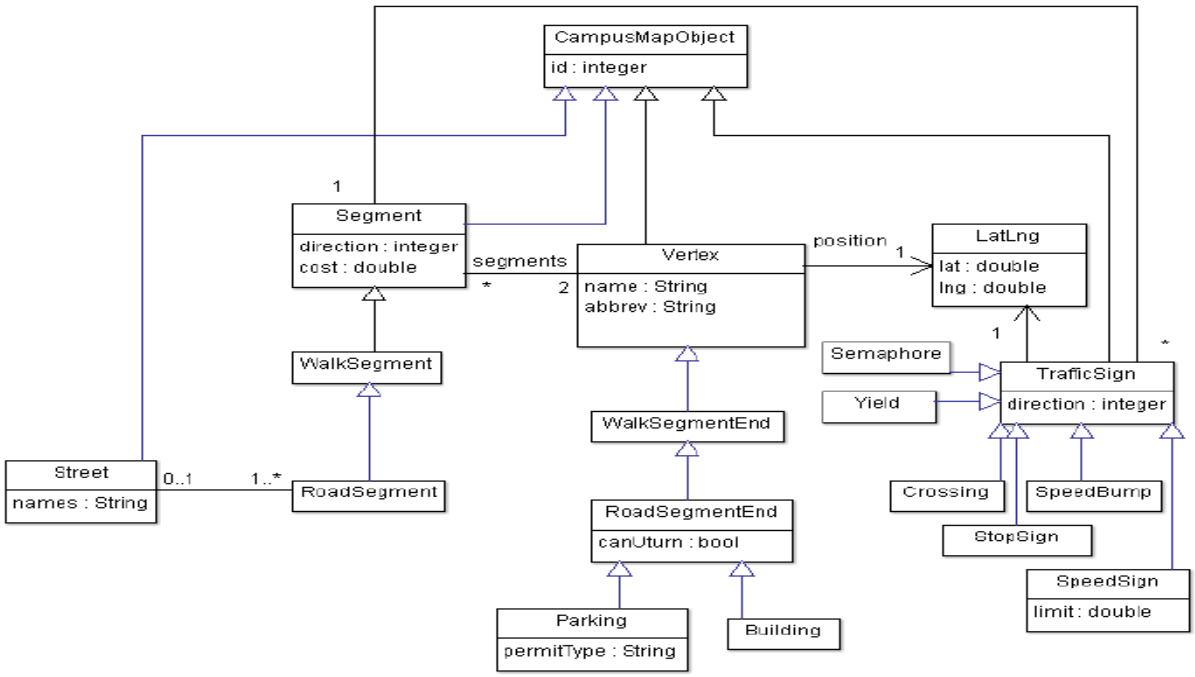

Fig -2 Uml Diagram

2.2 Implementation

Android apps are written in the Java programming language. The Android SDK tools compile your code along with any data and resource files into an APK, an Android package, which is an archive file with an .apk suffix. One APK file contains all the contents of an Android app and is the file that Android-powered devices use to install the app.

Each Android app lives in its own security sandbox, protected by the following Android security features:

- The Android operating system is a multi-user Linux system in which each app is a different user.
- By default, the system assigns each app a unique Linux user ID (the ID is used only by the system and is unknown to the app). The system sets permissions for all the files in an app so that only the user ID assigned to that app can access them.
- Each process has its own virtual machine (VM), so an app's code runs in isolation from other apps.

- By default, every app runs in its own Linux process. The Android system starts the process when any of the app's components need to be executed, and then shuts down the process when it's no longer needed or when the system must recover memory for other apps.

The Android system implements the principle of least privilege. That is, each app, by default, has access only to the components that it requires to do its work and no more. This creates a very secure environment in which an app cannot access parts of the system for which it is not given permission. However, there are ways for an app to share data with other apps and for an app to access system services:

- It's possible to arrange for two apps to share the same Linux user ID, in which case they are able to access each other's files. To conserve system resources, apps with the same user ID can also arrange to run in the same Linux process and share the same VM. The apps must also be signed with the same certificate.

- An app can request permission to access device data such as the user's contacts, SMS messages, the mountable storage (SD card), camera, and Bluetooth. The user has to explicitly grant these permissions. For more information, see Working with System Permissions.

The rest of this document introduces the following concepts:

- The core framework components that define your app.
- The manifest file in which you declare the components and the required device features for your app.
- Resources that are separate from the app code and that allow your app to gracefully optimize its behavior for a variety of device configurations.

App components are the essential building blocks of an Android app. Each component is an entry point through which the system or a user can enter your app. Some components depend on others.

There are four different types of app components:

- Activities.
- Services.
- Broadcast receivers.
- Content providers.

Each type serves a distinct purpose and has a distinct lifecycle that defines how the component is created and destroyed. The following sections describe the four types of app components.

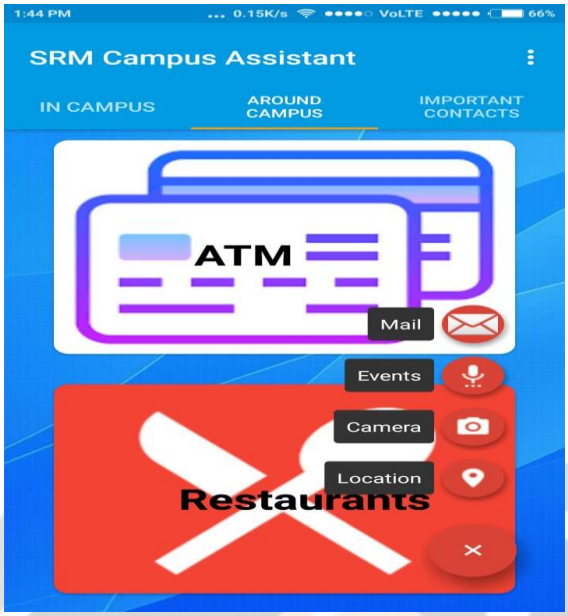

Fig-3: Floating view

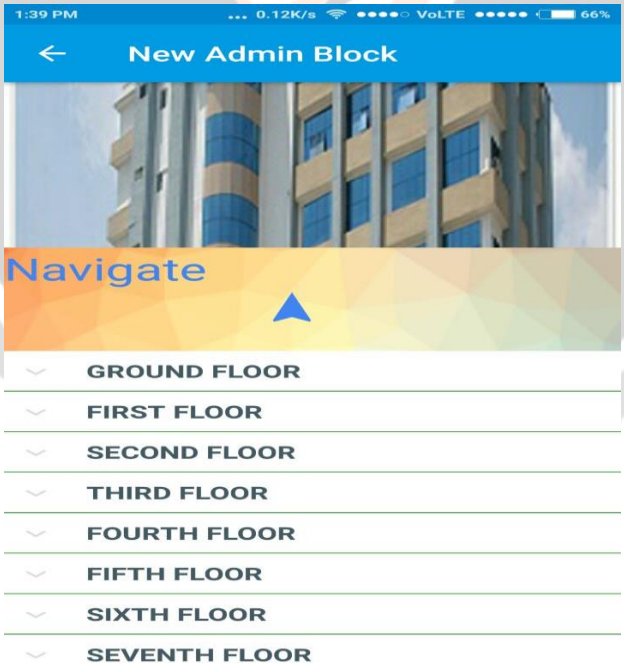

Fig-4 : Block View

3. CAMPUS ASSISTANT APPLICATION FOR THE ANDROID PLATFORM

The Campus Assistant app offers directions and walk-ing/driving navigation on one's Android smartphone. The steps performed by the app are as follows:

- Prompts the user for information: the user type (student, staff, or visitor), the campus (e.g. Boca Raton, Davie), and the destination location – usually a building or a parking lot.

- Downloads the campus' map XML file from the Mapping Server using HTTP.

- Parses the campus map XML file and builds the campus map graph data structure.

  Calculates the shortest path from the source to the destination (detailed below). The current GPS location is used as the source. The destination is computed based on the user type. For example, if a driving path to a building is selected, then driving directions to the closest parking lot where the user is permitted to park are provided.

- Directions and navigation prompts are displayed on the campus map on the Android smartphone. The shortest path is displayed, overlaid on the google map view. The current user location is displayed using a car/pedestrian icon. In addition, the user has the following options: select drivable or walkable path, select a new destination using the phone's touchscreen by tapping on the destination vertex marker, and zoom in/out with pinch gestures.

We implemented the app in Java using the Android SDK. For this project we used the Eclipse IDE and utilized the ADT plugin to edit files and manage the project. The Android API provides many useful packages that allow users to access the vast functionality of the Android device, such as the classes for the user interface and for accessing the compass and the GPS device. The Eclipse environment has excellent integration with the Android platform and allows users to debug applications running directly on the phone.

An important step in our application is parsing the XML map file to create the map graph data structures. A parser basically reads the XML document, identifies the tags, and extracts the data between the tags. This allows a computer program to access and use the data from the XML file. We used Document Object Model (DOM) to perform parsing. As result of parsing, we build data structures storing the campus map objects and their attributes, such as vertices (e.g. Buildings, Parking lots, etc.) and segments (e.g. road segments, walking segments).We compute the shortest-path between source and destination using Dijkstra's algorithm The original algorithm presented in computes shortest-paths from a source vertex to all other vertices in the graph. The weight a path is computed as the sum of the cost of all segments on the path. The algorithm stores the vertices to which the shortest-path has not been computed yet in a minimum-priority queue. At each step, one vertex with the minimum weight is removed, and its shortest-path calculation is completed.

Chart -2 : Social Media Survey

3.1 **System Testing**

The code review was done with the objective of identifying and correcting deviations from standards, identifying and fixing logical bugs and fall through and recording code walkthrough findings after successful completion. The programs were checked and the code structure was made readable. The variable names are meaningful. It follows certain naming conventions, which makes the program readable.

- Variable names are prefixed with their scope and data type
- Checking out for the correct scopes for various functions.
- .All possible explanations for the code were given as comments.
- Sufficient labels and comments are included in the code as the description of it.
- Code optimization was carried out.

The first level of testing different modules are tested against the specifications pro-duced during the design of the module. During this testing the number of the arguments is compared to input parameters, matching of parameter and arguments etc. Unit Test is conducted using a Test User usually.A series of stand-alone tests are conducted during Unit Testing. Each test examines an individual component that is new or has been modified. A unit test is also

called a module test because it tests the individual units of code that comprise the application.Each test validates a single module that, based on the technical design documents,was built to perform a certain task with the expectation that it will behave in a specific way or produce specific results. Unit tests focus on function-ality and reliability, and the entry and exit criteria can be the same for each module or specific to a particularmodule.

Unit testing is done in a test environment prior to system integration. If a defect is discovered during a unit test, the severity of the defect will dictate whether or not it will be fixed before the module is approved. Integration testing examines all the components and modules that are new, changed ,affected by a change or needed to form a complete system. It also differs from system testing in that when a defect is discovered, not all previously executed tests have to be rerun after the repair is made. Only those tests with a connection to the defect must be rerun, but retesting must start at the point of repair if it is before the point of failure. Integration testing is a systematic testing for constructing the program structure, while at the same time conducting test to uncover errors associated within the interface. Bottom-up integration is used for this phase. It begins construction and testing with atomic modules

Following Testing Are Done:-

- Unit Testing
- Integrated Testing
- User Acceptance Testing

4 . CONCLUSION

The SRM Campus Assistant can be used in the area of navigation system, contacting important personnels, reporting ragging complaint, browsing events and announcements in the college. The objective of this thesis is to develop a navigation and information system to help the students and more importantly freshers to get to the desired place in the area and get information about it in the app. The performance of the application depends on the features used and the classifier employed for recognition.

This research work attempts to propose a novel feature navigation method for navigating to local places and getting information about that place from the region of interest. Also the research work attempts to hybrid the traditional classifiers to recognize the object. The navigation and information system developed in this research is tested thoroughly. The application is implemented in Android Studio. In the first part of the work, a novel feature navigation method is used to navigate to the desired places. The proposed feature navigation method is compared with the traditional feature navigation methods. The proposed feature navigation method directs the control to the google maps.

To compare the efficiency of the proposed feature. The experimentation is done by varying the number of places. From the experimental results, the proposed feature performs well and provides higher navigation rate compared to other traditional methods. Also information about these places is included and site of the college is accessed to get the details of events , news and any announcement on the college website. The location for places is obtained from the address of that particular place on google maps.

5. ACKNOWLEDGEMENT

We respect and thank Mr B.Divager, for providing us an opportunity to do the project work at SRM University Ramapuram, Chennai and giving us all support and guidance which made us complete the project duly.

6. REFERENCES

[1]. Campus Assistant Application on an Android Platform ,Mihaela Cardei, Iana Zankina, Ionut Cardei, and Daniel Raviv,Department of Computer and Electrical Engeneering And Computer Science

[2]. Devlopment of GPS Based Navigator forKasetsart University,Kamphaeng Saen Campus, Siriwan chompukaew,Kwanchai Kungchaoren,,Wichian Premchaiswadi Gradute School of Information and technology, Department of Computer Engineering,Gradute School of Information and Technology

[3]. Android Devloper- https://developer.android.com/index.html

[4]. Stack overflow- https://stackoverflow.com/

[5] Udacity- https://in.udacity.com/android-developer