|  |  |  |
| --- | --- | --- |
| **CHAPTER 1**  **INTRODUCTION** |  |  |
| * 1. **INTRODUCTION**   The major society concern in recent period is security of women and children. Different severe crimes making this issue worse day by day. It is clearly observed that large share among these crimes is during travelling to and from the school. This fact draws a need of assisting this ‘travel’ in a smarter way.          In the era of smart phones with facilities like GPS, this would serve the best possible result if it is used for such critical issue like security of children. Purpose of developing this application is to make parents assure of the security and safety of their child by providing various notification, real-time view on map and various alerts of security.          Pre-notifying the school bus reaches near the house will certainly assure parents that child boards safely to the bus. Along with this presence of the child in the bus is also need to assured.           Current available applications do not provide integrity of all the major issues specified. This strongly motivated us to develop a smart phone application on popular android operating system to track school going children.  **1.2 PROBLEM DEFINITION**  Location based services for mobile telephony has reached new scale. System of location based services use GPS\GPRS services in the process. Modern traffic complexity and situations concern need to have advanced application to ensure security of school going children. It introduces and describes the internet-based Client-server methodology using Android Platform with inclusion of new technologies such as QR code and Google Cloud Messaging (GCM). QR code method ensures the authentication of particular student. Hence, we achieve the target to track children through their school bus.  **1.3 OVERVIEW**  Currently there are several applications such as ‘GPS Tracking Pro’, ‘Find my Friends’ which are used for tracking purpose but these applications have drawbacks such as excessive battery consumption. On the other hand, proposed application will provide application for both child and its respective parents. Server will continuously send child’s current position coordinates to its parent. QR code technique is used to confirm entry and exit of child with unique identification of child. |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**CHAPTER 2**

LITERATURE SURVEY

Literature survey discusses published information in particular area. It is summary of sources that we use while developing project. These useful documents help to maintain the knowledge up to date in current field. Comprehensive knowledge of the field is essential for proper implementation of project. It gives knowledge about the technologies implemented and guides us to make use of them.

For implementation of technologies like GCM, QR code we refer following papers:

1*.* Alexandros Doulamis, Nikos Pelekis, Yannis Theodoridis.” Easy Tracker: An Android application for capturing mobility behavior”, Dept. of Statistics & Insurance Sci. University of Piraeus, Piraeus, Greece.

It focuses on developing an android application in which real time coordinates of tracking object fetched by GPS services and they are visualize on maps. It provides facility to select between Google Map, Open Street maps or Microsoft Bing map. This encouraged us to implement the functionality to display the current position of child on Google map by fetching coordinates using GPS/GPRS services.

2. SR.Vijayshreivas and P.Antony Joe Christo,” Android –Geo Finder –A Real Time Tracking of Moveable Assets” , International Conference & Workshop on Recent Trends in Technology, (TCET) 2012 *Proceedings published in International Journal of Computer Applications® (IJCA).*

Concept of Reverse Geocoding is discussed in this paper. Public reverse geo-coding services require input of coordinate, captured from GPS, to look up a street address or neighboring places. This feature used to display the latitude and longitude position of child device that is to be tracked with the help of coordinates fetched using GPS/GPRS services and to display location on map for those particular values of coordinates.

3. Peter Kieseberg, Manuel Leithner, Martin Mulazzani, Lindsay Munroe, Sebastian Schrittwieser, Mayank Sinha, Edgar Weippl,” QR Code Security”, SBA Research

Favoritenstrasse 16 AT-1040 Vienna, Austria.

Paper discusses about the QR code technology, which contains information in encrypted format. It may website or personal information. Type of information varies from one application to another. It also discusses security and error correction capabilities of QR code. QR code concept proved helpful in maintaining the record regarding entry and exit of child at school bus. Each time child enters or exits from bus need to scan I-card, which is having QR-code. This confirms attendance of child to school bus.

4. Jin-Hwan Jeong,Hak-Young Kim,”Cloud Systems and Their Applications for Mobile Devices”, Cloud Computing Division Electronics and Telecommunication Research Institute, Daejeon,South Korea, *MOBILITY 2011: The first International Conference on Mobile Services, Resources and Users*

It focuses on collaboration of the cloud services with mobile application. Inspiring from this concept we also make use of GCM (Google Cloud Messaging) services for sending notifications to parent devices after specific time interval.

.

**BACKGROUND:**

For implementation of project, knowledge of different technologies, their efficiency and their advantages and importance in project need to be analyzed. Depth and strength of knowledge helps in deciding which technologies are helpful for the project. For this purpose, we considered the following aspects:

**CHOICE OF DOMAIN**:

**Android** is a Linux-based mobile operating system developed and distributed by Google, designed primarily for touchscreen smartphones and tablet computers. Android has a large community of developers writing applications ("apps") that extend the functionality of devices, written primarily in a customized version of *Java*. Google releases the Android code as open source, under the *Apache License 2.0*.

Android powers hundreds of millions of mobile devices in more than 190 countries around the world

It’s the largest installed base of any mobile platform

Android users download more than 1.5 billion app from Google Play

Open marketplace for distributing apps

**RELEVANCE OF PROJECT:**

A growing class of smartphone applications are the tasking applications that run continuously, process data from sensors to determine the user’s context (such as location) and activity, and optionally trigger certain actions when the right conditions occur.

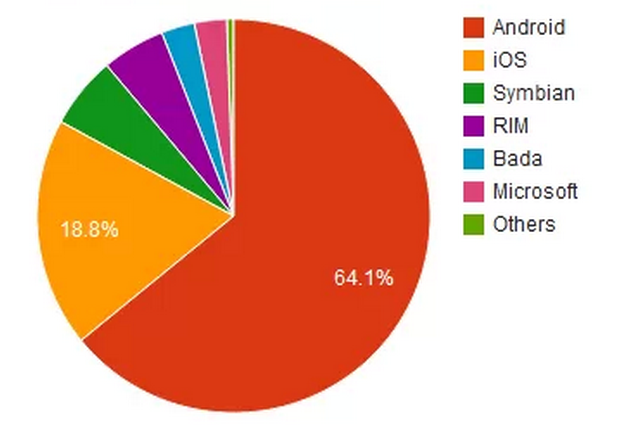


Fig 2.1: percentage distribution of OS

EXISTING SYSTEM:

Android applications can use device sensors to perform actions as required. Some of these applications are present in the device with OS itself, called as stock applications. Unfortunately we cannot modify these applications according to user requirements because complexity of code.

Following are the major drawbacks of current tasking applications:

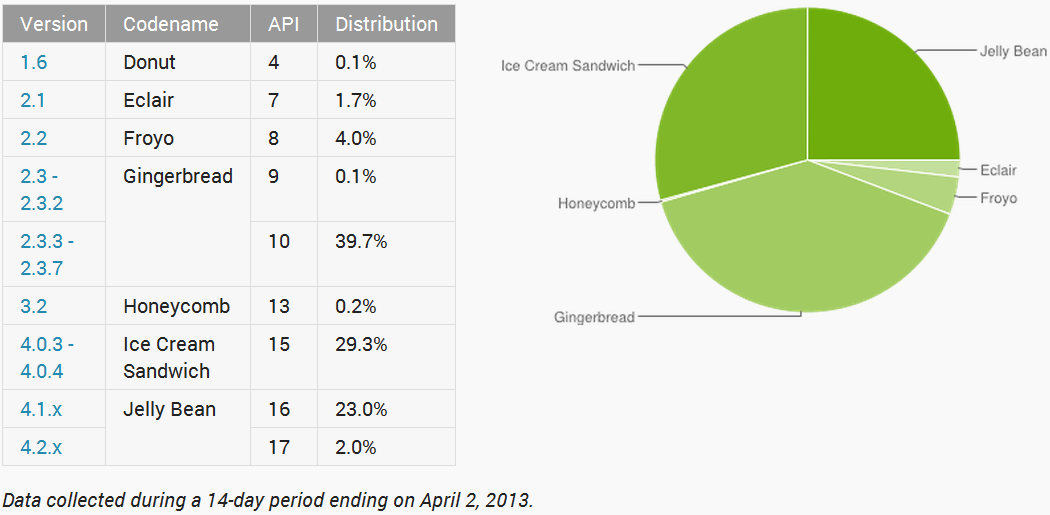
* Excessive battery usage, since continuous acquisition from sensors drains the battery.
* Most applications require Internet support for implicit updates.

PROPOSED SYSTEM:

It enables end users to easily express simple tasks on their phone. We would be bringing following major improvements:

* Simplification of tasking applications at User level (using GUI).
* Control over battery usage by efficient use of sensors.
* No spams or involuntary access to device network.
* Open source product that can be further improved

CHOICE OF PLATFORM:

******Fig 2.2: Current distribution of android versions (survey diagram)

From the survey diagram (Fig 2.2), we can say that around 39.7% of Android users use API level 10, almost 54.5% use above API level 10 and only 5.9% use API level below 10. Hence, going in accordance with the current market trend, we propose to set our platform at API Level 10 and above for the project. Therefore, all the applications that would use this framework should have minimum Android version of 2.3.1

QR CODE:

QR code stands for Quick Response Code which is used to store information in faster way. QR code is nothing but rectangular part that contains black and white part in form of matrix. QR code can store information such as name, URL, image etc.

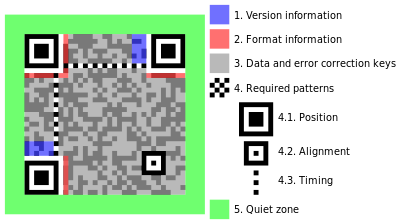


Fig.2.3 QR Code Structure

The major point comes in when we have to validate entry and exit of the person & identify him uniquely which is to be tracked in related vehicle. As per needs it is necessary to evaluate the person’s presence in vehicle, because there comes some situations when the person to be tracked might leave vehicle before its decided stop. Idea of QR code comes in picture to handle such situation. Every user to be tracked will be provided with unique ID which stored in form of QR code. This QR code can be printed on ID-card. QR code scanner will be provided in tracking application which will be mounted in every vehicle as child device to be tracked.

Whenever person enters in vehicle it can scan its QR code printed on ID card by this child device. It will trigger the application and will inform server that person with particular ID has entered in vehicle and server should manage track data for this person with that vehicle. Parent device or user will get track information of that particular device with help of its unique ID. At the exit point the person in vehicle or to be tracked can again scan the QR code to indicate end of the journey. This simple process helps to validate entry and exit of person in vehicle which is to be tracked.

There can be different layers to be considered such as bus failure. QR code method will handle such situation. Every Unique ID stored in QR code will help to track particular person. QR code will authenticate child’s presence on bus. Person can be tracked without condition of unique bus or bus route.

GOOGLE CLOUD MESSAGING:

GCM is as service that allows sending data from servers to the android devices. It provides service to send lightweight data such as notifications as well as messages containing up to 4kb of payload data.

GCM provides facility of sending data over the network by taking Registration ID as key. GCM stores the data if the target is not connected to the network and pushes the data as soon as it gets connected. GCM provides facility of Push Notifications.

Most important feature of GCM supports high flexibility. It does not provide any built-in interface to handle the message data but it provides control of handling to the android applications.

**CHAPTER 3**

**SOFTWARE REQUIREMENT SPECIFICATION**

**3.1 introduction**

**3.1.1 Project Scope**

The motivation behind this project is to assist and guide parents/guardians of school going children about vehicle position i.e. location of bus through which these children travels. System works with child phone (with Android O.S.) i.e. device to be tracked and parent phone (with Android O.S.) i.e. device used by user to track child phone. Both devices will be installed with developed android application.

Server plays vital and important role of information management and communication between the child and parent phone. Child phone will be mounted with vehicle, which is to be tracked. And parent device will be used by end-users which will track the related child. Child phone again uses the GPS or GPRS services to avail the co-ordinates. Usage of GPS or GPRS is been decided depending upon the accuracy they provide at run-time. Parent device will generate appropriate notifications as per position and condition of child device i.e. vehicle to be tracked. Co-ordinates will be sent by server to parent device using Google Cloud Messaging (GCM). Concept of QR code will be used to authenticate and validate entry and exit of child in vehicle. This QR code will also satisfy real time situations very well. New technology Co-ordinates fetched by child device, sent to the parent phone will display exact position of vehicle on Google maps.

**3.1.2 User Classes and Characteristics**

The system has different types of user classes one is parent user and another is child user. Both these user communicate with each other through server. Parent user communicates through an application interface. Basic operation will be co-ordinates of child device that is to be tracked will be fetched by GPS/GPRS services. Parent device will get notification after specific interval of time. Notification will be the current co-ordinates of child device. Parent can also see the actual position with the help of Google maps. Except these two users DBA plays vital role. DBA authenticate and keep track of user’s activities.

**3.1.3 Assumption and Dependencies**

* Users are assumed to have cellphones with Android OS to use this application.
* The availability of GPS is assumed. so that co-ordinates can be made available whenever required.
* It is mandatory to have internet access.
* Google map (APIs) for the mapping of co-ordinates.
* QRDroid Application.

**3.2 SYSTEM FEATURES**

**3.2.1 Registration**

Registration is performed by parents with help of android application. Parents provide details such as username, email ID, child name and child ID; then parents confirm registration. After registration database is updated. Once the parent is registered, for further usage direct log in can be performed. Parent also gets registered with GCM (Google Cloud Messaging) service.

**3.2.2 Scanning**

Each child will be provided with I-card having QR code. Each time child enters the bus, I-card is scanned which ensures the entry of child. Again scanning procedure is performed at the time of exit. Thus QR code helps in maintaining security at the time of entry as well as exit. It avoids unnecessary notifications to parents and ensures exact tracking of child.

**3.2.3** **Location Tracking**

Server will fetch co-ordinates from child device which uses GPS/GPRS services and further sends coordinates to parent devices availing GCM service. Parent devices will get notification from server after specific interval of time which also includes notification about the entry and exit of child in school bus.

**3.2.4** **Display Location**

The position of child can displayed on Google map with the help of latitude and longitude coordinates fetched by server.

**3.3 EXTERNAL INTERFACE REQUIREMENT**

**3.3.1 User Interface**

Android application is developed for both the ends child device and parent device. Parent will be provided with registration activity for the first time and database is updated accordingly. Parent can see the position of child device on Google map with provided utility. Child device will have an interface which will be used for scanning QR code.

**3.3.2 Software Interface**

As the application is developed in android, software assistance required is android environment. So smartphone with android OS is the basic interface required for this application. Google map APIs. QRDroid qr-code scanning utility.

**3.3.3 Hardware Interface**

Hardware part such as server should configure with both end devices. Communication between these hardware nodes should be done with help of internet connection.

**3.3.4 Communication Interface**

For communication purpose GPS/GPRS services are used. It helps to fetch coordinates of child devices using these services and these coordinates are send to parent devices using GCM services.

**3.4 SYSTEM FEATURES AND FUNCTIONAL REQUIREMENTS**

|  |  |  |
| --- | --- | --- |
| **Feature** | **Description** | **Functional requirements** |
| QR code scanner | It scans the QR code on the child’s Id card and authenticates its presence. | * Scanner application should be installed in phone and it should get updated automatically as new releases. |
| Google Cloud Messaging (GCM) | GCM is used to send notification messages from server to parent device application. | * It requires devices running Android 2.0 or higher that also have Google Play Store application installed. * Google account is required for pre-3.0 device. * Register ID, application ID are needed. Registration is mandatory. |
| Authentication | It provides secure access to users of system so that one user neither accidently nor intentionally accesses other user’s data and profile. | * Unique ID name for each user. * Password should be stored in encrypted format. |

Table 3.4: Functional Requirements

**3.5 NONFUNCTIONAL REQUIREMENTS**

**3.5.1 Performance Requirements**

* Entire project works depending on the coordinate fetching
* For best co-ordinate fetching GPRS and GPS services can be used.
* Service that provides coordinates with better accuracy is selected at run time
* It gives faster coordinate fetching facility
* Also the internet connectivity needs to be maintained throughout the project for faster and better response.
* Also the GCM technology is used which facilitates buffering of messages when application is offline. And delivers the message as soon as it gets connectivity.

**3.5.2 Safety Requirements**

* Use of public and private key constraints facilitates the safety
* Each child is having unique child\_id which has primary key constraint

**3.5.3 Security Requirements**

* Security is achieved with the help of QR code technique
* When QR code is scanned the information related to each child is given to server
* With the help of received information server sends notification to respective parents of child.
* Information of each child is secured.
* Notifications sent to parents after fixed interval of time, which ensures the security of child.

**3.5.4 Software Quality Attribute**

Software Quality Attributes (SQA) consists of a means of monitoring the software engineering processes and methods used to ensure quality.

SQA encompasses the entire software development process, which includes processes such as requirements definition, software design, coding, source code control, code reviews, change management, configuration management, testing, release management and product integration.

**3.6 OTHER REQUIREMENTS**

**3.6.1 Database Requirements**

Database Stores child profile and inter-relate it with Parent profile. Co-ordinates are stored in database, which keeps updating at run time.

Database consists of three tables:

1. **Child ID**

It has attributes: Device ID, Latitude, Longitude.

It helps in maintaining the information regarding ID of each child device and its current positional coordinates.

1. **Parent**

It has attributes: ID, gcm\_regID, name, Email, Child\_name, Child\_ID, created \_date

It has record of each child device and its respective parent. GCM services are used to send coordinates to parent devices. This service requires particular registration ID for its activation so that we can use this service. This unique registration ID is indicated by gcm\_regID field.

**3. Student**

It has attributes: child\_ID, Device\_ID, Child\_name

It maintains the record of no of children in particular bus. Gives information about child\_ID and their name for particular Device\_ID.

**3.6.2 Reuse Objectives for the Project**

The basic part of the module can be extended further to use for different applications such as tracking of corporate employees especially those who work in night shift, package delivery, public bus tracking, personal tracking etc.

**CHAPTER 4**

**SYSTEM DESIGN**

**4.1 SELECTION OF LIFE CYCLE MODEL**

Before actually starting with the description of the process model, it is worth answering the question, ‘Why Agile? Why XP?’In a simple words, agile philosophy encourages customers satisfaction and early incremental delivery of software engineering work; small highly motivated project teams; informal methods; minimal software engineering work products; and overall development simplicity. The agile development guidelines stress delivery and design, and active and continuous communication between developers and customers.

To answer ‘Why agile?’ consider the statement : It is difficult to predict in advance which software requirements will persist and which will change and in the world full of uncertainties, it is always impossible to plan in advance. One has to be adaptable and flexible enough to meet those uncertainties. It is here the answer to the question lies. Agile methodologies, at the bottom, accept that things always changes in this uncertain world and they made themselves lean and flexible yet not discouraging analysis and design completely but keeping under the check and instead focus on communication, simplicity, feedback, respect and courage.

In conclusion, we choose Agile as a software development life cycle model.

The Agile software development life cycle is a set of methodologies as follows:

**4.1.1 Scope:**

The Agile software development life cycle recognizes the complexity of system development. The development process is inclusive of more than the traditional perspectives in Information Technology. This means adopting multiple plans and seeing how the processes can attain the full cycle of development. The multiple scope of the development cycle then expands as the plan progresses to being developed and being operational.

**4.1.2 Planning:**

The planning stage of the Agile software development life cycle involves the following considerations: specifying the opportunity available at hand, strategizing, and feasibility. The goal of the planning stage is to determine development potentials.

**4.1.3 Initiation:**

The actual field work starts on this state. It involves funding, seeking for stakeholder support, determining the scope of the project, team building and environment set up, project estimation, and establishing an operational development model.

**4.1.4 Construction Iterations:**

This step in the Agile software development life cycle involves arriving at quality software output. It involves analytical and repetitive work to determine viability in preparation for production.

**4.1.5 Release Iterations:**

This stage in the Agile software development life cycle involves final output testing, defect-finding steps and improvement measures, documentation, training for both users and support teams and dispatch or endorsement to production.

**4.1.6 Production:**

This means manufacturing the systems and taking steps to effectively launch it in the market and ensuring that the system will be used. Opportunities for releasing enhanced system versions may be seen.

**4.1.7 Retirement:**

This is the part in the Agile software development life cycle where a part of the system or the entire system is going to be pulled from production. Several and complementary reasons for this happening may be system replacement, being outdated and redundancy.

The outline of the model is outlined in next figure (Figure 4.1).

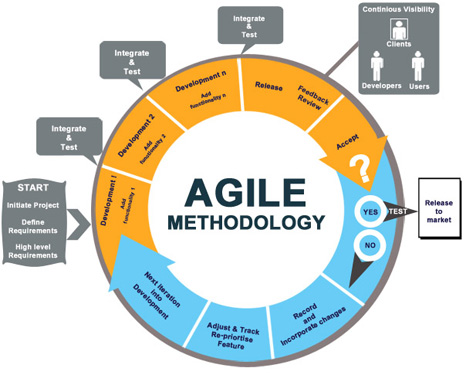
****

Fig 4.1: Agile Methodology (SDLC)

**­­­­­­­­**

**4.2 SYSTEM IMPLEMENTATION PLAN**

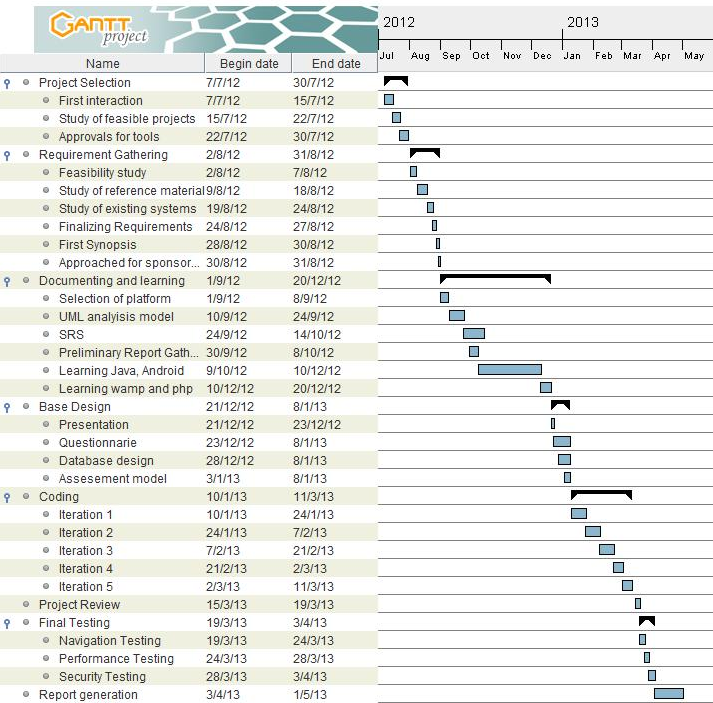
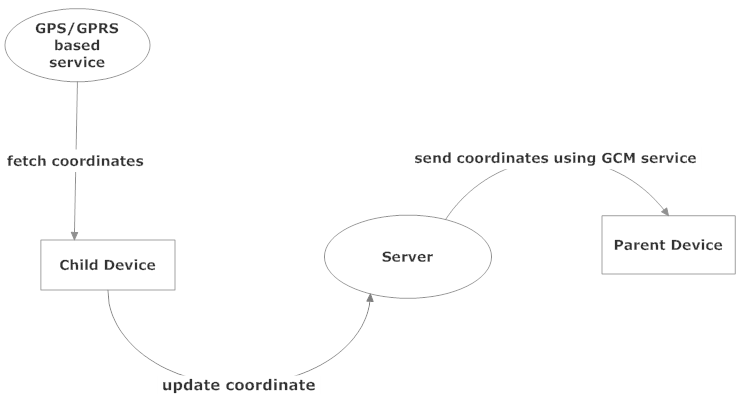
****

Fig 4.2: Timeline chart

**4.3 Data Flow Diagram**

Fig 4.3.1: Data flow-0 level

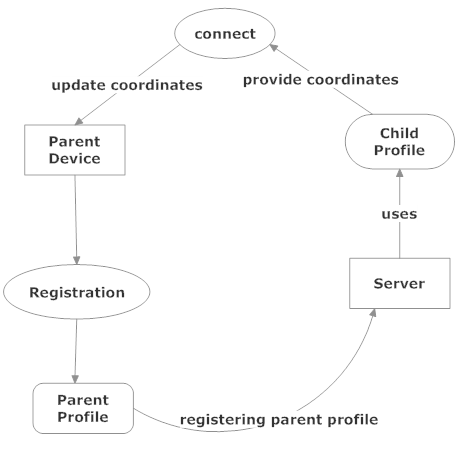
****

Fig 4.3.2: Data flow-1 level

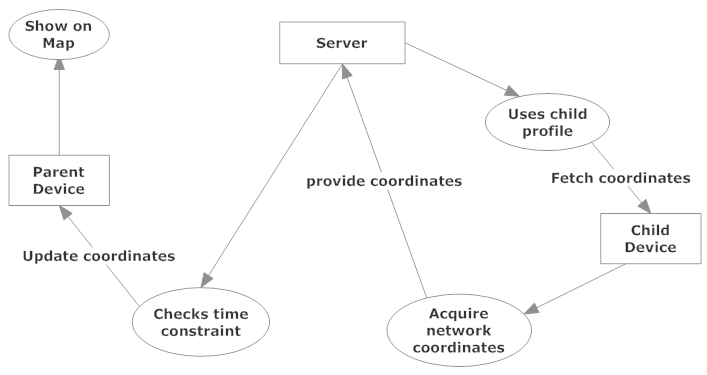
****

Fig 4.3.3: Data flow -2 levels

**4.4 Class Diagram**

****

Fig 4.4.1: Parent side class diagram

****

Fig 4.4.2: Child side class diagram

****

Fig 4.4.3: Server side class diagram

**4.5 USE CASE DIAGRAMS**

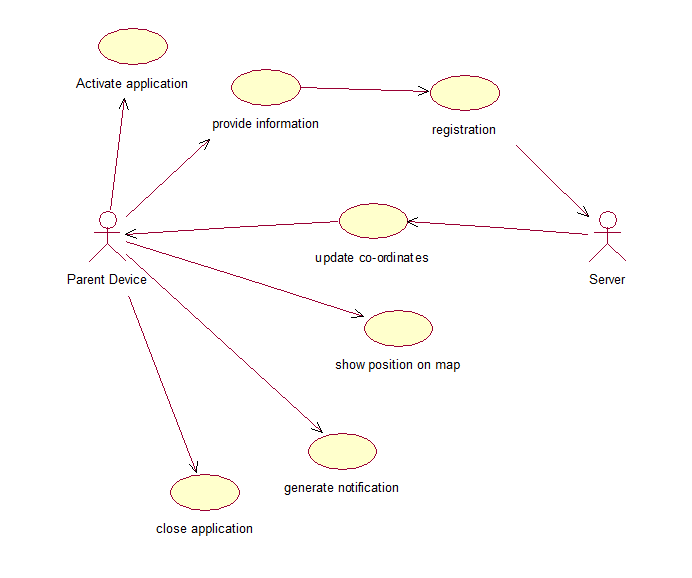
****

Fig 4.5.1: Use Case Diagram-1

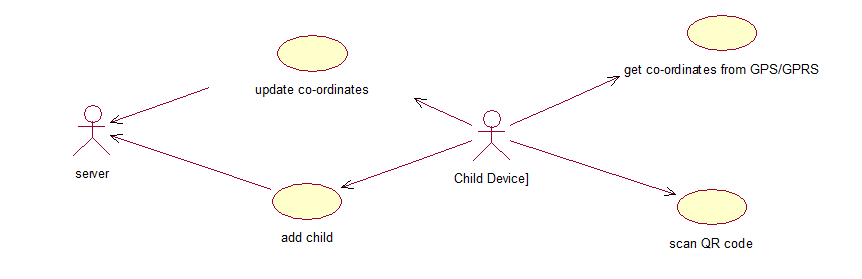


Fig 4.5.2: Use Case Diagram-2

**4.6 SEQUENCE DIAGRAM**

Fig 4.6.1: Sequence diagram for parent device

****

Fig: 4.6.2: Sequence diagram for child device

**4.7 ACTIVITY DIAGRAM**

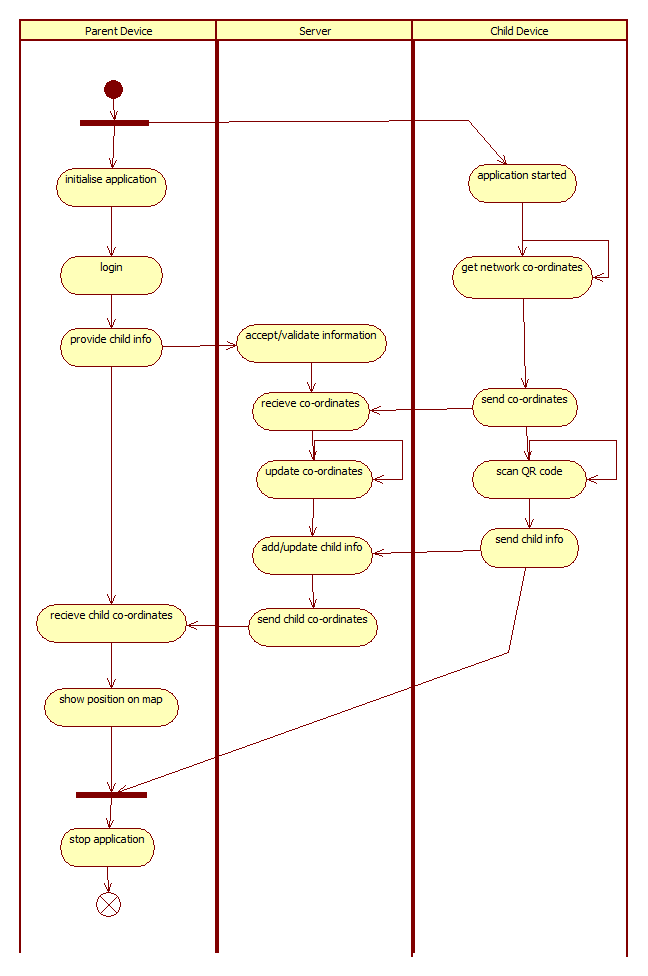


Fig 4.7: Activity diagram

**4.8 DIPLOYMENT DIAGRAM**

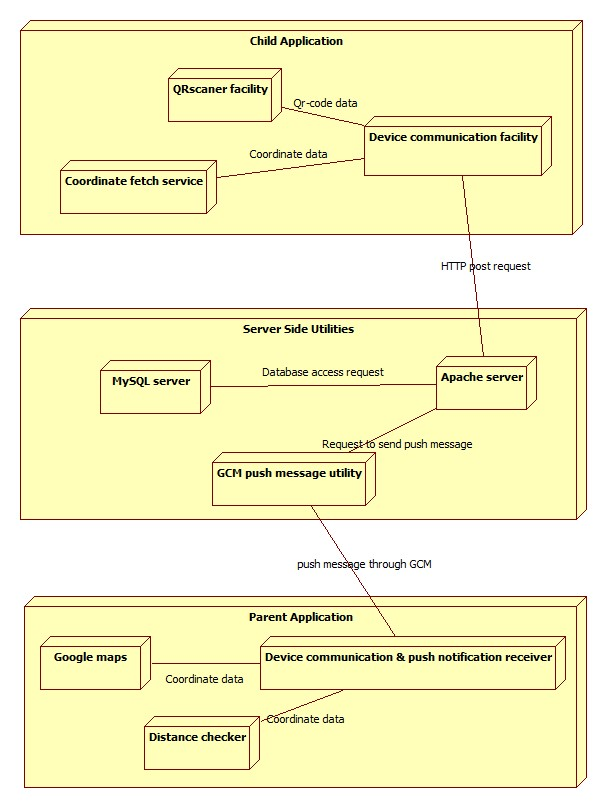
****

Fig 4.8: Deployment diagram

**CHAPTER 5**

**SYSTEM ARCHITECTURE**

**5.1 OVERVIEW:**

System architecture is the conceptual model that defines the structure, behavior, and more views of a system.

An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures of the system, which comprise system components, the externally visible properties of those components, the relationships between them, and provides a plan from which products can be procured, and systems developed, that will work together to implement the overall system. More recently, there have been efforts to formalize languages to describe system architecture, collectively these are called architecture description languages (ADLs).

We consider different aspects of this system to be developed. There are hardware components such as child device, parent device and server. Moreover, software components are there such as parent side application, child side application and server programs. These all hardware and software components communicate and exchange data in the whole execution process. These all components work together to serve user. As we look at different modules of the system, we can find module for communication, authentication, QR, co-ordinate fetch, Request acceptance and request serving modules etc.

**5.2 SYSTEM ARCHITECTURE**

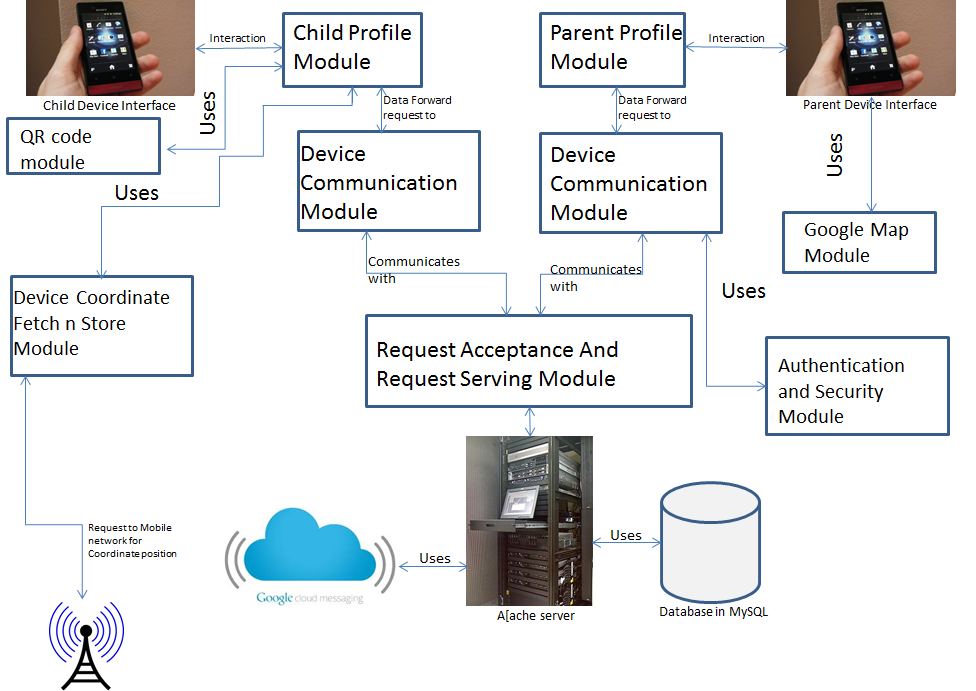


Fig 5.1: System Overview

**SYSTEM ARCHITECTURE DISCRIPTION:**

* **From child-device's prospective:**
  + 1. Device scans QR-code and sends it to server through device\_communication module
* for this purpose it uses
  + - * qr-code module
      * device coordinate fetch and store module
    1. Device\_communication module communicate with server using http post methods and it also inserts other required data like child-device id to data to be sent
* **From parent-device's prospective:**
  + 1. Parent-device first registers to avail push message facility using authentication and security module
    2. For communication with server it uses device\_communication module
    3. It uses other functionalities for receiving and displaying push messages
    4. It also uses Google map for displaying location on map and reverse Geo-coding facility for coordinate to address conversion
* **From server's prospective:**
  + - 1. Server accepts coordinates from child-device and stores them to database using MySQL manager
      2. It registers parent-device and also checks authenticity for parent-device using database stored
      3. It communicates through http responses
      4. Server uses GCM service for sending push notifications
      5. On receiving new coordinates from child-device it stores them to database and forwards them to all valid parent-device using gcm service

**5.3 ALGORITHM:**

**5.3.1 On Child device:** (devices are made available at every school bus)

1. School going children enter and leave the bus scanning their I-card printed with QR code.

2. The very moment any child enter or leaves the bus device scans its QR code fetches current coordinate of bus and sends it through http requests with additional information of bus device id.

* + This activity performed using functionalities of following classes:
* Qr\_read\_activity
* GPS\_tracking\_activity

1. Apart from children entering or leaving bus, child device also fetch coordinate location of bus after a interval of 3 min and send them along with bus device id.
2. This device also checks for the total no of children left in bus before closing application.
3. Default activities of display and data sending are handled by main class only.

**5.3.2 On parent device:** (these are the android devices used by parents)

1. Parents register it self to server for activation purpose
   * This activity performed using functionalities of following classes:
     + 1. RegisterActivity
       2. AlertDialogueManager
       3. ConnectionDetector
2. On successfully registration device, activate WakeLocker class for listening to any incoming push messages.
3. On receiving push messages from GCM service the arrived coordinates of child are stored, and are made available for looking on map and converted to real address.
   * + - * This activity performed using functionalities of following classes:

ShowOnMap

MainActivity

1. Parent device also performs check for knowing bus's localtiy with respect to his pre-decided home location.
   * This activity performed using functionalities of following classes:
2. MainActivity

**5.3.3 On server side:**

* Server mainly handle 4 sections:
  1. Receiving coordinate location from child-devices
  2. Sending coordinate to parent-devices
  3. Checking authentication and registration activity of parent-devices
  4. Database storage and retrieval process
* Section one: Receiving coordinate location from child-device
  1. This section involves lesser authentication and concentrate more on grabbing more and more data at faster rate.
  2. Involves http response methods.
  3. Further communicate with data storage section for storage of arrived data.
  4. This section do not involve bidirectional communication
* Section two: Sending coordinate to parent-devices
  1. This section involves frequent read access of stored data for the purpose of sending bus locations to parent-devices.
  2. On receiving the new coordinates from child-device the parent section undergoes a process of sending coordinates to all parent-devices having their children in that particular bus.
* Section three: Checking authentication and registration activity of parent-devices
  1. This section involves registration and activation of parent-devices for receiving push notifications.
  2. This section allows parent to register with their unique user-name and personal details as well as id of their children, this creates their profile in server storage.
  3. Registering the device allows server to send them push notification using GCM service.
* Section four: Database storage ans retrieval process
  1. This section involves the php embedded sql queries for storage and data retrieval purpose.
  2. MySQL database manager is used with efficient nesting of fired queries.
  3. None of the database elements are directly accessible to users the are accessed through securely designed php page.
  4. This section mainly stores the data from authentication and registration section and also from data receiving section.
  5. Data is mainly accessed by two section one is sending coordinate to parent-devices and other is by registration & authentication section. Both have different purpose and level of normalization for their concern tables are different.
  6. Tables used by registration & authentication section are designed with security prospective.
  7. Tables accessed by coordinate sending section are designed with a view of faster retrieval and update prospective.

* 1. **FLOWCHART:**

**5.4.1 Child Application:**

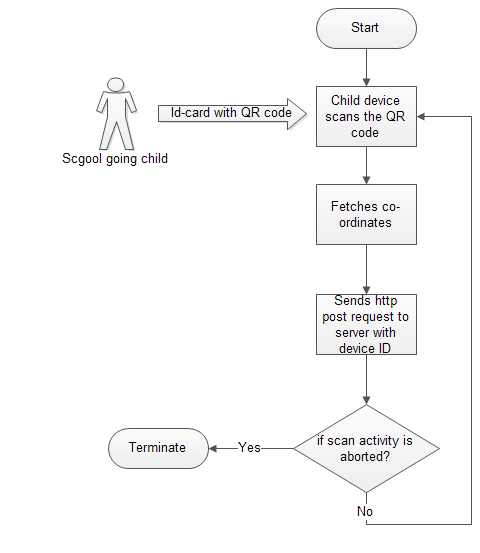
****

Fig 5.4.1: Child Application Flowchart

**5.4.2 Parent Application:**

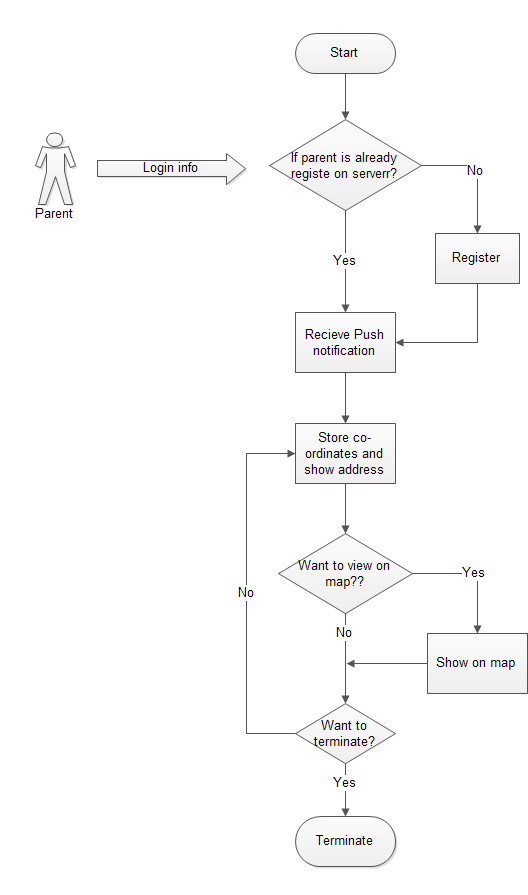
****

Fig 5.4.2: Parent Application Flowchart

**CHAPTER 6**

**TECHNICAL SPECIFICATIONS**

**6.1 TECHNOLOGY DETAILS USED IN PROJECT:**

As the project is mainly concerned for smart phone users, it has been developed on Android platform. Android provides an Open Source platform for application development with the help of various APIs available.

**Android Features having major concern with the application:**

 Android is an open source platform which allows programmers to make changes in the readily available codes as per their application specific requirements.

 Android phone has great popularity, which makes available the larger community of application users.

 Inclusion of location-aware strategies in an Android application is easily possible as we can utilize GPS and Android's Network Location Provider to acquire the user location.

* Security of child information is achieved by using QR code technology. Each child I-card is supposed to have QR code which, after scanning provides necessary details.
* New attempt is made to send notification message using GCM services.

 Google Cloud Messaging requires separate registration to access its services.

 GCM ensures message passing efficiently as buffer facility is provided when application is offline

 No need to check connectivity each time while sending the message. Whenever user device gets connection, messages are delivered to an application.

.

**6.2 HARDWARE REQUIREMENT:**

**On Client Side:**

1. Android Phone with version above 2.3.3

2. Internet connectivity

3. Phone must be GPS enable

4. Memory requirement 3MB

**On Server Side:**

- Intel Xeon E3- 1220v2 CPU (Quad Core)

- 3.1 Ghz 8 MB Cache,

- 1600 MHz, 1x4GB Memory,

- 1x500 GB SATA 7200 RPM, 3.5" Simple Swap

**6.3 SOFTWARE REQUIREMENT:**

**On client side:**

1. Android OS

2. QR droid application

3. Google Maps

**On server side:**

1. Apache 2.2

2. MySQL

**CHAPTER 7**

**SOFTWARE IMPLEMENTATION**

**7.1 INTRODUCTION**

Implementation involves all those activities that take place to convert from the old system to the new system. The new system may be totally new; replacing an existing manual or automated system, or it may be a major modification to an existing system. Proper implementation is essential to provide reliable system to meet the organizational requirements. Unsuccessful implementation may not guarantee improvement in the organization using the new system, as well as, improper installation will prevent any improvement.

The implementation phase involves the following tasks:-

* Careful planning.
* Investigation of system and constraints.
* Design of methods to achieve the functionality.
* Training of users involved in the operation.
* Evaluation of changeover.

**7.2 DATABASE:**

Our project has major concern with databases. Child id, Device id, phone number, Scanned QR code, Network co-ordinates etc. are the main attributes stored in the database.

Following are the important tables in our database:-

1. Track\_child\_id

| [**device\_id [Descending](http://localhost/phpmyadmin/sql.php?db=track&table=child_device&sql_query=SELECT+*+FROM+%60child_device%60+ORDER+BY+%60child_device%60.%60device_id%60++ASC&token=4a3190c035a55a6c44798a0f1ada5a95)**](http://localhost/phpmyadmin/sql.php?db=track&table=child_device&sql_query=SELECT+%2A+FROM+%60child_device%60+ORDER+BY+%60child_device%60.%60device_id%60++ASC&token=4a3190c035a55a6c44798a0f1ada5a95) | [**latitude**](http://localhost/phpmyadmin/sql.php?db=track&table=child_device&sql_query=SELECT+%2A+FROM+%60child_device%60+ORDER+BY+%60child_device%60.%60latitude%60+ASC&token=4a3190c035a55a6c44798a0f1ada5a95) | [**longitude**](http://localhost/phpmyadmin/sql.php?db=track&table=child_device&sql_query=SELECT+%2A+FROM+%60child_device%60+ORDER+BY+%60child_device%60.%60longitude%60+ASC&token=4a3190c035a55a6c44798a0f1ada5a95) |
| --- | --- | --- |
| 394883400202345 | 18.12 | 73.56 |
| 353561056489911 | 18.4803639 | 73.8215635 |
| 353043058871487 | 18.5082272 | 73.8261814 |

Table 7.2.1: Track Child-Id Table

2.Parent\_table

| [**id**](http://localhost/phpmyadmin/sql.php?db=track&table=parent&sql_query=SELECT+%2A+FROM+%60parent%60+ORDER+BY+%60parent%60.%60id%60+ASC&token=4a3190c035a55a6c44798a0f1ada5a95) | [**gcm\_regid**](http://localhost/phpmyadmin/sql.php?db=track&table=parent&sql_query=SELECT+%2A+FROM+%60parent%60+ORDER+BY+%60parent%60.%60gcm_regid%60+ASC&token=4a3190c035a55a6c44798a0f1ada5a95) | [**name**](http://localhost/phpmyadmin/sql.php?db=track&table=parent&sql_query=SELECT+%2A+FROM+%60parent%60+ORDER+BY+%60parent%60.%60name%60+ASC&token=4a3190c035a55a6c44798a0f1ada5a95) | [**email**](http://localhost/phpmyadmin/sql.php?db=track&table=parent&sql_query=SELECT+%2A+FROM+%60parent%60+ORDER+BY+%60parent%60.%60email%60+ASC&token=4a3190c035a55a6c44798a0f1ada5a95) | [**child\_name**](http://localhost/phpmyadmin/sql.php?db=track&table=parent&sql_query=SELECT+%2A+FROM+%60parent%60+ORDER+BY+%60parent%60.%60child_name%60+ASC&token=4a3190c035a55a6c44798a0f1ada5a95) | [**child\_id**](http://localhost/phpmyadmin/sql.php?db=track&table=parent&sql_query=SELECT+%2A+FROM+%60parent%60+ORDER+BY+%60parent%60.%60child_id%60+ASC&token=4a3190c035a55a6c44798a0f1ada5a95) | [**created\_at**](http://localhost/phpmyadmin/sql.php?db=track&table=parent&sql_query=SELECT+%2A+FROM+%60parent%60+ORDER+BY+%60parent%60.%60created_at%60+ASC&token=4a3190c035a55a6c44798a0f1ada5a95) |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | APA91bHKxMnQakK5o6lPurVZOnU0DaJNb\_MG3DnmVEoEPKfyvn... | Ramesh | [ramesh@gmail.com](mailto:ramesh@gmail.com) | Raj | 9999 | 2013-03-02 11:41:08 |
| 2 | APA91bFR5ju1S0nZpsQHQJ1GVHXUJ5kO53bG2N6A3h\_5ZeO71v... | Ganesh | [ganeshr@yahoo.com](mailto:ganeshr@yahoo.com) | Sham | 9998 | 2013-03-02 13:34:21 |
| 4 | APA91bHKxMnQakK5o6lPurVZOnU0DaJNb\_MG3DnmVEoEPKfyvn... | Suyash | [hulk@avengers.com](mailto:hulk@avengers.com) | Nisha | 9876 | 2013-03-06 08:36:07 |
| 5 | APA91bFMx29507rX8-Ai3YUzz3CvuDQILDsVxyO6ZrAOBFAKaZ... | Apurva | [suyashsjadhav@gmail.com](mailto:suyashsjadhav@gmail.com) | Chinu | 9454 | 2013-03-10 11:56:58 |
| 6 | APA91bGzPp8Z\_iMzVDl7q93nIhIrGhqZ0lnBrdykxBIqZe03sp... | Swapnil | [suya@gmail.com](mailto:suya@gmail.com) | Rohit | 6789 | 2013-03-10 14:07:21 |

Table 7.2.2: Parent Table

Bottom of Form

Top of Form

Bottom of Form

Bottom of Form

3.Student table

| [**child\_id**](http://localhost/phpmyadmin/sql.php?db=track&table=student&sql_query=SELECT+%2A+FROM+%60student%60+ORDER+BY+%60student%60.%60child_id%60+ASC&token=4a3190c035a55a6c44798a0f1ada5a95) | [**device\_id**](http://localhost/phpmyadmin/sql.php?db=track&table=student&sql_query=SELECT+%2A+FROM+%60student%60+ORDER+BY+%60student%60.%60device_id%60+ASC&token=4a3190c035a55a6c44798a0f1ada5a95) | [**child\_name**](http://localhost/phpmyadmin/sql.php?db=track&table=student&sql_query=SELECT+%2A+FROM+%60student%60+ORDER+BY+%60student%60.%60child_name%60+ASC&token=4a3190c035a55a6c44798a0f1ada5a95) |
| --- | --- | --- |
| 6789 | 353561056489911 | Rohit |
| 9454 | 353043058871487 | Chinu |
| 9876 | 353043058871487 | Nisha |
| 9998 | 353561056489911 | Sham |
| 9999 | 353561056489911 | Raj |

Table 7.2.3: Student Table

**7.3 Graphical User Interface:**

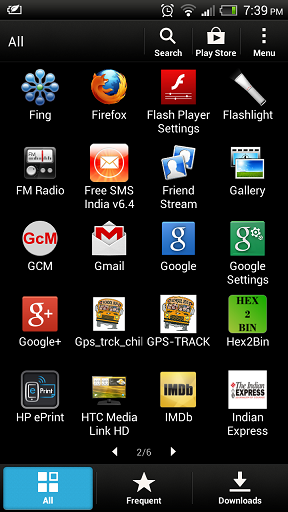


Fig 7.3.1: Application Launch Icons

This shows two different applications for two intended users of this application.

1.GPS\_trck\_child :- Child side application.

2.GPS-TRACK :- Parent side application.



Fig 7.3.2: Gps\_track\_child home screen

This is the interface of child side application.It displays welcome message with instructions. Button named ‘Start QRcode scan’ starts the scanning functionality of the child side of application.

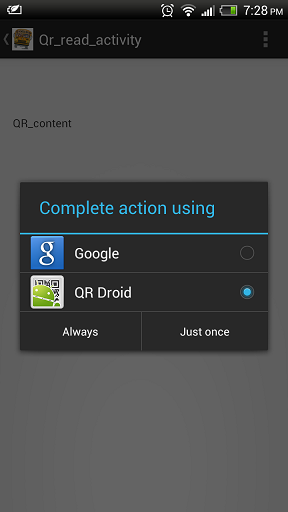


Fig 7.3.3: Gps\_track\_child- QR switch screen

QR switch screen giving option of choosing from google web or installed QR Droid application.Options like Always and Just once are embedded in the windowpane.

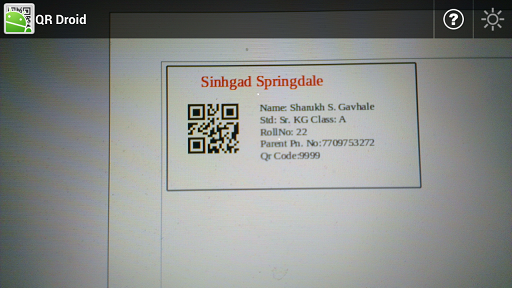


Fig 7.3.4:Gps\_track\_child - QR scan screen

It displays the correct way of putting QR-code in front of the camera of the device.

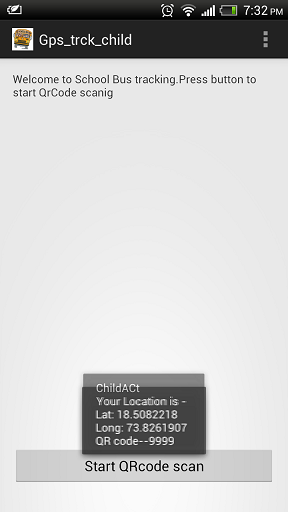


Fig 7.3.5: Gps\_track\_child with toast

After successful scan of QR code, toast is generated which, includes co-ordinates of scanning location along with the scanned QR code in decoded String format.

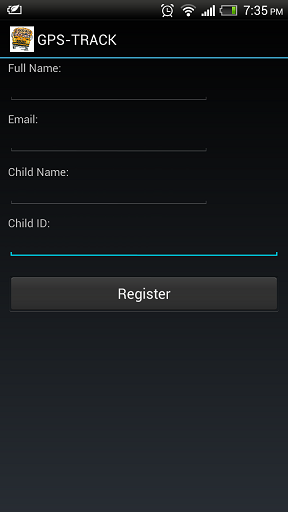


Fig 7.3.6: GPS-TRACK Registration screen

Parent side application provides registration interface for the first login.

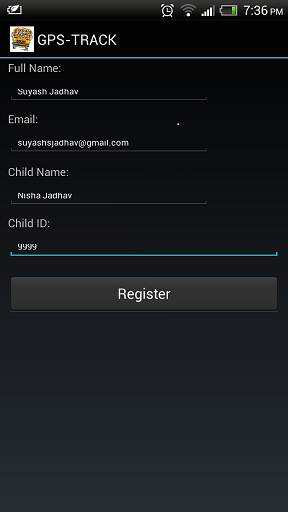


Fig 7.3.7: GPS-TRACK- Registration screen with required info sample

It shows default way to fill up in the registration form. ‘Full Name’, ’Child Name’ are filled in text format, ‘Email’ is filled in email format and ‘Child ID’ is filled with number format.

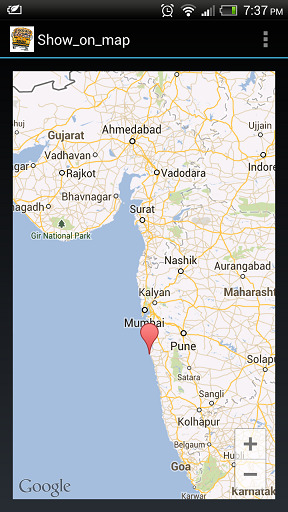


Fig 7.3.8: GPS-TRACK Showing child location on Google map

Co-ordinates get fetched during the run-time of the application and are displayed on the Google maps with pin pointing the exact location mapped by these co-ordinates.

**CHAPTER** 8

**PROJECT ESTIMATE, SCHEDULE & TEAM STRUCTURE**

**8.1 TASKS & MILESTONES**

|  |  |  |
| --- | --- | --- |
| Tasks | Start Date | Finish Date |
| Requirement Gathering | **10-09-2012** | **18-10-2012** |
| Analysis | **24-10-2012** | **02-12-2012** |
| Design | **15-12-2012** | **25-02-2013** |
| Documentation | **03-03-2013** | **15-03-2013** |

Table 8.1: Tasks and Milestones

**8.2 COST AND EFFORT ESTIMATION**

The Constructive Cost Model (COCOMO) is generally used estimation measures of cost, project duration, work force etc.

Like all estimation model, the COCOMO model requires sizing information. This information can be specified in the form of

* Object Point(OP)
* Function Point(FP)
* Lines of code(KLOC)

For our project, we use the sizing information in the form of Lines of source code.

Total Lines of source code in our project KLOC =6 K (approx.)

Cost of each person per month Cp= 2500/-(Per Person-month)

**Equations**

Equation for calculation of effort in person-month for the COCOMO model is:

E=a\*(KLOC) ^ b

Where,

a=3.0

b=1.12 for semidetached projects

E=Effort in person-months

D= a\*(E) ^ b

Where,

a=2.5

b=0.35 for a semidetached projects

D=Duration of project in months.

**Semi-Detached Projects:**

* A project of moderate size and complexity where teams with mixed experience levels must meet a mix of rigid and less than rigid requirements.
* Equation for calculation of Number of people required for completion of project, using the COCOMO model is
* N= E/D

Where,

N=Number of people required

E=Efforts in person-month

D=Duration of project in month

* Equation for calculation of cost of project, using the COCOMO model is:

C= D \*Cp

Where,

* C=Cost of project
* D=Duration of project in months
* Cp= Cost incurred per person-month
* Efforts:

E= 3.0(6) ^ 1.12

E=22.31person-months

Total of 22.31 person-months are required to complete the project

Successfully

* Duration of project:

D=2.5\*(E) ^ 0.35

D=5.5 months

* Number of people required for project:

N=22.31/5.5

N=4.05

N= 4 people

Therefore, four people are required to complete the project

* Cost of project:

C= 5.5\* 2500

C=13,750/-

Therefore cost of project is approx. Rs.13750/-

After calculating the time and cost by COCOMO model observation says that, obtained person-month result is nearer to the planned values. It is simpler way to calculate the cost as the only input require is Lines of codes. It not only calculates just efforts but also resources and cost. Considering the resources and time stamp of 5.5 months the estimated cost is affordable and hence the project is cost efficient.

According to surveys, the cost required for developing static android applications which involves simple functionalities is 3K to 5K USD. On the contrary, this project involves functionalities like gcm services, QR code, gps services and also involves server actions. In spite of that cost of project is low. Thus estimation results in best efficiency in efforts as well as cost.

**CHAPTER 9**

**SOFTWARE TESTING**

**9.1 INTRODUCTION**

Testing focuses primarily on the evaluation or assessment of product quality realized through a number of core practices.

1. Finding and documenting defects in software quality.
2. Advising about perceived software quality.
3. Proving the validity of the assumption made in design and requirement specifications through concrete demonstration.
4. Validating the software product functions as designed.
5. Validating that the requirements have been implemented appropriately

Test challenges the assumptions and risks. Testing of software is difficult. The different ways in which a program can behave are indefinable. A well-conceived methodology and use of state-of-art tolls can help improve the productivity and effectiveness of software testing.

1. The requirement discipline captures requirements for the software product and those requirements are one of the primary inputs for identifying what tests are to be performed.
2. The analysis and design discipline determines the appropriate design for the software product; this is another important input for identifying what tests are to be performed.
3. The implementation discipline produces builds of software product that are validated by the test discipline. Within iteration multiple builds will be tested.
4. The environment discipline develops and maintains supporting artifacts that are used during test, such as the guidelines and test environment.
5. The management discipline plans the product.

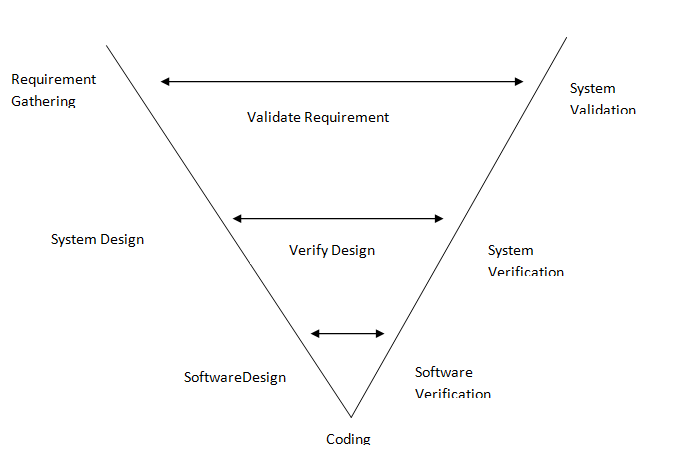


Fig: 9.1 V-Testing Models.

**9.2 UNIT TESTING WITH TEST CASES AND RESULTS**

During the unit testing phase, the system is tested while it is developed. Here all the options of the system are validated.

During the first phase of this testing the testing person tests the system by entering the valid data, or by performing the appropriate function which the system requests for. This phase of testing is done to verify whether the system performs all requested functions.

**9.2.1 White box testing**

White box testing sometimes called, as glass box testing is a test case design method that uses control structures of the procedural design to derive test cases. Using white box testing methods, the software engineer can derive test cases that:

1. Guarantee that all independent paths within module have been exercised at least once.
2. Exercise all logical decisions and their or false sides.
3. Execute all loops and within their operational bounds.
4. Exercise internal data structures to ensure their validity

**9.2.2 Black Box Testing**

It focuses on functional requirement of the software. The area of interest is the functional performance of the system i.e. how it converts the input to output. Black box testing attempts to find errors in the following categories:

* Incorrect error
* Interface error
* Initialization error
* Performance error
* Errors in data structures and external database errors

**9.2.3 Loop testing**

Loops are the cornerstone for the vast majority of all algorithms implemented in software. Loop testing is a white-box testing technique that focuses exclusively on the validity of loop constructs. Four different classes of loops can be defined: simple loop, nested loop, concatenated loop and unstructured loop.

**9.2.4 TEST CASES**

**Test case 1 –**

**Test Case ID:** 1 **Test Method:** Manual

**Version:** 1.0

**Test Case Name:** First Run

**Test Case Description:** Test case involving first run of the Application

**Pre-requisites:** Application is present on android mobile device.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step No.** | **Step Description** | **Input Data** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 11. | Check is it the first run of the application |  | Home Page | Home Page is displayed | Pass |
| 22. | User will enter valid details. | Correct Fields | User will get details displayed as the entered values and registration status. | It should be first run of the application. | Pass |

Table 9.2.1: First run test case

**Test Case Status [Pass/Fail]:** Pass

**Test case 2 –**

**Test Case ID: 2 Test Method: Manual**

**Version: 1.0**

**Test Case Name:** Create New Parent

**Test Case Description:** Create a new Parent

**Pre-requisites:** Parent should use application on his own device.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step No.** | **Step Description** | **Input Data** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1. | Enter personal information | Full name, email id, name of child, Id of child | As the User types the data, it should be displayed in the respective fields. | Entered data is displayed in the fields. | Pass |
| 2. | User will save the record by clicking on “Register” |  | Data will be store in database using foreign key constraints and parent will get register on GCM. | New Record created  Successfully. | Pass |

Table 9.2.2: Create New Parent

**Test Case Status [Pass/Fail]:** Pass

**Test case – 3**

**Test Case ID: 3 Test Method: Manual**

**Version: 1.0**

**Test Case Name:** Childs entry

**Test Case Description:** Child scans its QR code in particular device in bus

**Pre-requisites:** Child should have its QR-code I-card.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step No.** | **Step Description** | **Input Data** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1. | Make entry of child with respect to child-device. | QR code, and device ID | QR code is scanned. Store the child’s position and update the co-ordinates of device. | Profile updated successfully and updated value stored in database. | Pass |

Table 9.2.3: Childs entry

**Test Case Status [pass/Fail]:** Pass

**Test case 4 –**

**Test Case ID: 4 Test Method: Manual**

**Version: 1.0**

**Test Case Name: Send child position to parent.**

**Test Case Description:** Send co-ordinates of child to the respective parent.

**Pre-requisites:** Parent should be registered and child must be in the Bus.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step No.** | **Step Description** | **Input Data** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1. | Send co-ordinates to parent device and show it on map. | Co-ordinates send by server to the respective parent. | Co-ordinates received on parent device and position is displayed on map. | Same as Expected Result. | Pass |

Table 9.4: Send child position to parent

**Test Case Status [Pass/Fail]:** Pass

**Test case 5 –**

**Test Case ID: 5 Test Method: Manual**

**Version: 1.0**

**Test Case Name:** Co-ordinates updates.

**Test Case Description:** Child device will send co-ordinates to server.

**Pre-requisites:** Database maintains Device ID and its entry.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step No.** | **Step Description** | **Input Data** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1. | Search particular device id and update it. | Device Id provided by child device. | Co-ordinates of particular device-Id get updated. | Device co-ordinates get updated. | Pass |

Table 9.2.5: Co-ordinates updates

**Test Case Status [Pass/Fail]:** Pass

**Test case 6 –**

**Test Case ID: 6 Test Method: Auto**

**Version: 1.0**

**Test Case Name:** Check connectivity

**Test Case Description:** Check if devices are connected to network and GPS is available.

**Pre-requisites:** Application is activated.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step No.** | **Step Description** | **Input Data** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1. | Child device is connected to network or GPS. |  | Application works fine if connection is established else it shows connection error. | Application works fine without connection error. | Pass |
| 2. | Parent device is connected to network. |  | Application works fine if connection is established else it shows connection error. | Application works fine without connection error. | Pass |

Table 9.2.6: Check connectivity

**Test Case Status [Pass/Fail]:** Pass

**Test case 7 –**

**Test Case ID: 7 Test Method: Manual**

**Version: 1.0**

**Test Case Name:** QR scan

**Test Case Description:** Check if valid application is initialized to scan QR code.

**Pre-requisites:** Child application is working.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step No.** | **Step Description** | **Input Data** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1. | After we click ‘Scan QR code’ button, pop-up dialog box to select available QR code scan application. |  | Valid QR code scan application opened after selection. | QRDroid application opened after we select it. | Pass |
| 2. | Camera gets activated to scan QR code after selection of application. |  | Camera scan QR code and data is displayed with the help of toast. | QR code gets scanned and data is displayed. | Pass |

Table 9.2.7: QR scan

**Test Case Status [Pass/Fail]:** Pass

**CHAPTER 10**

**RESULT**

**10.1 RESULT PHASES :**

Result of the application is obtained in the four main phases. These phases comprise of both parent and child side applications.

**10.1.1 QR Code Scanning:**

Result produces at this phase includes the decoded QR code (i.e. in the string format) along with the location Co-ordinates (Longitude and Latitude) of the location where the actual scanning of QR code took place.

QR code is uniquely used in our application building which makes this application superior in terms of Security constraints as compared with other application. Popular applications in this domain use traditional Email-Password for the authentication.

**10.1.2 Display On Map:**

This is the result on the Parent side of application. After Server pushes message which includes Co-ordinates (Latitude, Longitude), using these co-ordinates, the exact location is displayed on the Google Map.

All the applications in this domain uses the same method of display with minor changes in aesthetics.

**10.1.3 Display Address:**

This is unique result generated on Parent side of application. After reception of server’s push message, using Reverse GeoCoding functionality, location (Area) of the location pointed by co-ordinates is fetched and displayed it in Toast.

This functionality makes application stands different from existing application as it eliminates the need of loading Google map on each run.

**10.1.4 Notifications Alert:**

This Result is the key, which distinguishes this application from existing ones. After reception of server’s push message, using co-ordinates and Mathematical Model, The notification is generated as the result if it satisfies the condition specified in mathematical model.

This result endorses security with the added advantage of lower battery consumption.

**10.2 COMPARATIVE ANALYSIS CHART :**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Comparison**  **Parameter** | **School Bus Tracking** | **Life360 Family Locator** | **Where’s**  **My droid** | **Cell Phone Tracker** |
| User Group | Scalable | Family | Single | 5(maximum) |
| Authentication Type | QR-Code Scanning | Email-Password | Text code | Email-Password |
| GPS/GPRS option | Yes | Yes | Only GPS | Yes |
| Cost | Free | Free | Free | Free(7 days only) |
| Battery Consumption | Moderate | High | Low | Moderate |
| Size | 2.47Mb | 7.03Mb | 650Kb | 5.07Mb |
| Background Run | Yes | No | Yes | Yes |
| Message Facility | Server Push Messages | User Interaction | Absent | User Interaction |
| Separate Child-Parent applications | Yes | No | No | No |

Table 10.2: Comparative Analysis

**CHAPTER 11**

**DEPLOYMENT AND MAINTAINANCE**

**11.1 INSTALLATION AND UN-INSTALLATION**

* Initially application need to be installed in both child device and parent device
* For this purpose, as it is android application .apk file need to be copied
* Child device having facility to scan the I-card must be installed in school bus
* Child device provides the facility of scanning at entry as well as exit.
* Parent device should install parent application which will track child device
* Internet connection is mandatory during execution.
* Un-installation of an application is same as that of other android application.

**11.2 USER MANUAL**

When application is deployed for the use in real time environment user can take help of following points for its usage:

* For child device application, basic requirement is child must have an I-card consists of unique QR code.
* Each child must scan his/her I-card with the help of device installed in a bus.
* Each child need to scan the I-card at the time of entry and exit this confirms the attendance of child in a school bus
* For parent application, initially parent need to register to the application
* During registration information like Name, Email-ID, Child\_name, Child\_id need to provide.
* After registration, whenever parents want to see the position of a school bus, they can request for co-ordinates by selecting the option ‘get co-ordinates’
* Another facility is parent can directly see the position on map by selecting the option ‘show on map’
* Parents get notified continuously after fix interval of time about the location of a school bus.

**CHAPTER 12**

**CONCLUSION**

Implementation of school bus tracking system resulted in improving overall safety and security of school going children. Usage of QR code technique helps in maintaining the record of entry and exit. Thus this helps in avoiding the unnecessary notifications to parent even when the child is on leave. QR code is provided on I-card of each child so explicitly no need to carry any special card. It is offered in simplified format. As the system is focusing on every individual there is no limitations for unique path and vehicle.

GCM technology helps in improving message passing efficiency as there is facility to store messages even when the application is offline. For accurate co-ordinate fetching we have used both the services GPS/GPRS. Accurate coordinates are fetched at run time.

School bus tracking system is thus efficient and useful application for parents to assure safety of their children. This application shows location of bus on map which add-ons to the utility of software. Parents are updated automatically after fixed time. Parent device needs to have internet connectivity and updates will be provided. It results into an efficient application, which can serve best for personal as well as business purpose.

.

**CHAPTER 13**

**REFERENCES**

**13.1 BOOKS**

* Agile software development, Alistair Cockburn, Addison Wesley Publication
* Distilled UML 2.0, Martin Fowler, Addison-Wesley Press
* Head First Java, O’REILLY, Kathy Sierra, Burt Bates
* Java – The complete Reference, Osborne, Herb Schildt
* Learning Android, O’REILLY, Marko Gargento

**13.2 WEBSITES**

* + http://PHP.net
  + http://Android.developer.com
  + http://W3school.com
  + http://agilemanifesto.org/principles.org

**13.3 RESEARCH PAPERS**

[1] K.V.Barai,R.M.Samant,2011 ,”GPS/GPRS based system for vehicle tracking and traffic management”, *ICWET’11* *proceeding of the International Conference & Workshop on Emerging Trends in Technology.*

[2] Tamil, E.M., D. B. Saleh, and M.Y.I. Idris,” A Mobile Vehicle Tracking System with GPS/GSM Technology”, *Proceedings of the 5th Student Conference on Research and Development*(SCORED), Permala Bangi ,Malaysia, may 2007.

[3] Khondker Shajadul Hasan, Tanzil Rahman, “ Cost-Effective GPS-GPRS Based Object Tracking System”, *Proceedings of the International Multi-Conference of Engineers and Computer Scientist 2009* Vol I IMECS 2009, March 18-20,2009, Hong Kong.

[4] Dr. Kamal Jain and Rahul Goel , GPS Based Low Cost Intelligent Vehicle Tracking System (IVTS)”, *2012 International Conference on Traffic and Transportation Engineering (ICTTE 2012)*IPCSIT vol. 26 (2012) © (2012) IACSIT Press, Singapore.

[5] Deepak Mishra, Apurv Vasal, Puneet Tandon, “A Novel and cost effective approach to public vehicle tracking system”, *International Journal of* *UbiComp* (IJU), Vol.3, No.1, January 2012.

**ANNEXURE-A**

[[1]](#footnote-1)

Towards a Secure Travel Assistance Device for GPS/GPRS Enabled Mobile Users to Aid Transit Riders with Special Needs

**Swapnil Mahajan, Suyash Jadhav, Sourabh Gavhale, Apurva Inamdar, Suvarna Baheti**

**{swapnilmahajan510,** **suyashsjadhav,** **sourabh.gavhale,** **inamdarapurva,suvarna.baheti}@gmail.com**

**First Author Affiliation, Second Author Affiliation, Third Author Affiliation**

*Abstract— As urban living environment is becoming more and more complex; the road condition is becoming worse because of heavy traffic, tremendous increase in accidents and most importantly the security issues that have been raised by recent incidences. To solve such problem with respect to school going children, a school bus tracking system has been developed. The system provides ANDROID Applications for the device in a bus as well as for parents of children travelling by the bus. A school bus tracking system determines the position of the bus with a GPS/GPRS enabled cell phone and displays the position on a Google map. This School Bus tracking system not only meets the security expectations of parents by providing features as Alert Messages and Conformance of Attendance of the Child on the bus, but also provides convenience to parents as it notifies when school bus comes near the house.*

*Index Terms*— *Android, GPS, GPRS, QR code, Google maps*.

# INTRODUCTION

The motivation behind this project is to assist and guide parents/guardians of school going children about vehicle position i.e. location of bus through which these children travels. System works with child phone (with Android O.S.) i.e. device to be tracked and parent phone (with Android O.S.) i.e. device used by user to track child phone. Both devices will be installed with developed android application.

Server plays vital and important role of information management and communication between the child and parent phone. Child phone will be mounted with vehicle which is to be tracked. And parent device will be used by end-users which will track the related child. Child phone again uses the GPS or GPRS services to avail the co-ordinates. Usage of GPS or GPRS is been decided depending upon the accuracy they provide at run-time. Parent device will generate appropriate notifications as per position and condition of child device i.e. vehicle to be tracked. Co-ordinates will be sent by server to parent device using Google Cloud Messaging (GCM). Concept of QR code will be used to validate entry and exit of child in vehicle. This QR code will also satisfy real time situations very well. New technology Co-ordinates fetched by child device, sent to the parent phone will display exact position of vehicle on Google maps.

.

# PROBLEM DEFINITION

Android tracking applications has reached new scale. Various applications has been researched and developed to track different objects. System uses GPS\GPRS services in the process. If we look at the modern traffic complexity and situation, need of advanced application comes up. This paper discusses new methodologies to tackle these situations. It introduces and describes the internet-based Client-server methodology using Java-Android Platform with inclusion of new technologies such as QR code and Google Cloud Messaging (GCM).

# MATH

Major concern of this application is to calculate the distance.Application is mainly intended to be energy efficient.Distance calculation formula includes very tidious calculations due to involvent of many trignometric functions. This application is very little concern about accurracy as it requires to generate a notification over a comparetively wide range of 0-1100 meters. It gives rise to the new formula/concept.

After bus is being located in the specified range about the predefined point, generation of notification is carried out on the Parent Type of Device.This formula plays a vital role in security as well as it leads to the energy efficiency of the application.

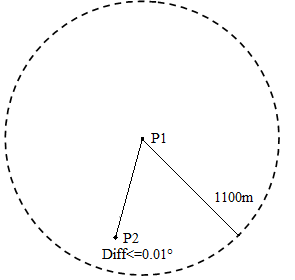


Fig.1.Proposed Architecture

1.Calculate distance between Latitude of points

ΔΘ=| Θ1 - Θ2 |

2.Calculate distance between Longitude of points

Δλ=| λ1 - λ2 |

3.Diff = ΔΘ + Δλ.

4.If (Diff< =0.01°) then P2 is expected to be within the range of 0m to 1100m from P1.

# Observations and Results

Table 1.1 Evaluation of Related Work

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Place1** | **(Latitude,Longitude)**  **(θ1, λ1)** | **Place2** | **Latitude, Longitude)**  **(θ2, λ2)** | **Δθ=**  **θ1-θ2** | **Δλ=**  **λ1- λ2** | **Δθ+Δλ** | **Distance (m)** |
| **Sinhgad College** | **(18.4684, 73.8351)** | **Jadhav Nagar** | **(18.4676,**  **73.8286)** | **0.0008** | **0.0065** | **0.0073** | **809.14** |
| **Bhansali**  **Campus** | **(18.4695,**  **73.8308)** | **0.0011** | **0.0043** | **0.0054** | **619.382** |
| **Jadhav Gadh** | **(18.457**  **,**  **73.8358)** | **0.0114** | **0.0007** | **0.0121** | **1211.14** |
| **Rajaram Bridge** | **(18.4879, 73.8299)** | **Japnese Garden** | **(18.4932,**  **73.8355)** | **0.0053** | **0.0056** | **0.0109** | **866.050** |
| **Ganesh Mala** | **(18.4963,**  **73.8355)** | **0.0084** | **0.0086** | **0.0170** | **1356.000** |
| **Vithal**  **wadi** | **(18.4838,**  **73.8276)** | **0.0041** | **0.0023** | **0.0064** | **550.534** |
| **Karishma Society.** | **(18.5043, 73.8214)** | **City Pride kothrud** | **(18.4994,**  **73.821)** | **0.0049** | **0.0004** | **0.0053** | **541.614** |
| **Nal Stop** | **(18.5091,**  **73.8319)** | **0.0048** | **0.0105** | **0.0153** | **1256.490** |
| **Karve Chowk** | **(18.5017,**  **73.815)** | **0.0026** | **0.0064** | **0.009** | **631.126** |
| **Deccan Chowk** | **(18.5142, 73.8491)** | **Fergusson College** | **(18.5267,**  **73.8418)** | **0.0125** | **0.0073** | **0.0198** | **1051.170** |
| **Nal Stop** | **(18.5091,**  **73.8319)** | **0.0051** | **0.0172** | **0.0223** | **1898.690** |
| **MacD**  **JM**  **Road** | **(18.518,**  **73.844)** | **0.0038** | **0.0051** | **0.0089** | **477.193** |

# Description

## *QR Code*

QR code stands for Quick Response Code which is used to store information in faster way. QR code is nothing but rectangular part that contains black and white part in form of matrix. QR code can store information such as name, URL, image etc.

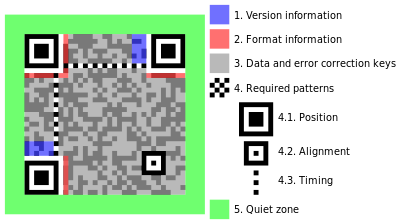


Fig.2 QR Code Structure

The major point comes in when we have to validate entry and exit of the person which is to be tracked in related vehicle. As per needs it is necessary to evaluate the person’s presence in vehicle, because there comes some situations when the person to be tracked might leave vehicle before its decided stop. Idea of QR code comes in picture to handle such situation. Every user to be tracked will be provided with unique ID which stored in form of QR code. This QR code can be printed on ID-card. QR code scanner will be provided in tracking application which will be mounted in every vehicle as child device to be tracked.

Whenever person enters in vehicle it can scan its QR code printed on ID card by this child device. It will trigger the application and will inform server that person with particular ID has entered in vehicle and server should manage track data for this person with that vehicle. Parent device or user will get track information of that particular device with help of its unique ID. At the exit point the person in vehicle or to be tracked can again scan the QR code to indicate end of the journey. This simple process helps to validate entry and exit of person in vehicle which is to be tracked.

There can be different layers to be considered such as bus failure. Such situation can also be considered with help of QR code method. Every Unique ID stored in QR code will help to track particular person. Tracking procedure will now be focused on person but not on only bus. Uniqueness of bus and route for particular person to be track won’t be needed with this method. It will solve many problems.

## *Google Cloud Messaging (GCM)*

GCM is as service that allows sending data from servers to the android devices. It provides service to send lightweight data such as notifications as well as messages containing up to 4kb of payload data.

GCM provides facility of sending data over the network by taking Registration ID as key. GCM stores the data if the target is not connected to the network and pushes the data as soon as it gets connected.GCM provides facility of Push Notifications.

Most important feature of GCM supports high flexibility. It does not provide any built-in interface to handle the message data but it provides control of handling to the android applications.

# System Architecture

An android application of School Bus tracking is mainly featured as to play the role of Child Device And/Or Parent Device with the Same Interface. Both devices are provided witg Device Communication Modules separately as per their Role of acting. Major requirement of security is fullfilled by Higher degree of Authentication which leads to enable live tracking by an autherized user after accepting the request of Parent By the Child phone.

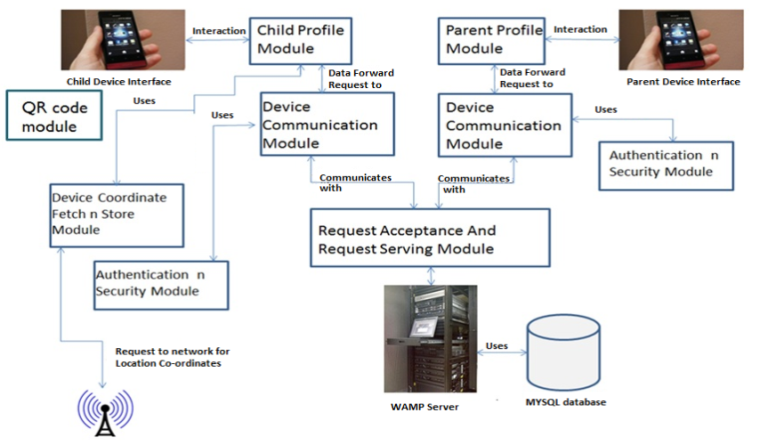


Fig.3 system architecture diagram

GPS And/Or GPRS facilities are used to fetch the network co-ordinates. Enabling both the services for fetching co-ordinates provides the greater flexibility to choose between the criterias required as per the need of the application. QR codes are used to scan the identity card of the student and it will be used as a key to look for the entry in the database . Request coming from both types of devices are categorized and processed accordingly.The system Architecture mainly focuses on combining simpler functions together so as to provide a strong answer to the question of security.

# Conclusion

Vehicle tracking system resulted in improving overall safety and security of school going children. As the system is focusing on every individual there is no limitations for unique path and vehicle. It results into an efficient application which can serve best for personal as well as business purpose.

**Acknowledgment**

We take this opportunity to thank for the invaluable help and support we have received directly and indirectly during the preparation on paper.

We would like to express our deep sense of gratitude to our External guide Prof.Vivek Gadgilfor his valuable assistance and guidance in different aspects of application development and constant support.We would also like to express our gratitude towardsProf. Pratikshit Mahalle (HOD of Computer)for providing all facilities and every help for the smooth work in the process.

**REFERENCES**

[1] K.V.Barai,R.M.Samant,2011 ,”GPS/GPRS based system for vehicle tracking and traffic management”, *ICWET’11* *proceeding of the International Conference & Workshop on Emerging Trends in Technology.*

[2] Tamil, E.M., D. B. Saleh, and M.Y.I. Idris,” A Mobile Vehicle Tracking System with GPS/GSM Technology”, *Proceedings of the 5th Student Conference on Research and Development*(SCORED), Permala Bangi ,Malaysia, may 2007.

[3] Khondker Shajadul Hasan, Tanzil Rahman, “ Cost-Effective GPS-GPRS Based Object Tracking System”, *Proceedings of the International Multi-Conference of Engineers and Computer Scientist 2009* Vol I IMECS 2009, March 18-20,2009, Hong Kong.

[4] Dr. Kamal Jain and Rahul Goel , GPS Based Low Cost Intelligent Vehicle Tracking System (IVTS)”, *2012 International Conference on Traffic and Transportation Engineering (ICTTE 2012)*IPCSIT vol. 26 (2012) © (2012) IACSIT Press, Singapore.

[5] Deepak Mishra, Apurv Vasal, Puneet Tandon, “A Novel and cost effective approach to public vehicle tracking system”, *International Journal of* *UbiComp* (IJU), Vol.3, No.1, January 2012.

**ANNEXURE-B**

**MATHEMATICAL MODEL**

Major concern of this application is to calculate the distance.Application is mainly intended to be energy efficient.Distance calculation formula includes very tidious calculations due to involvent of many trignometric functions. This application is very little concern about accurracy as it requires to generate a notification over a comparetively wide range of 0-1100 meters. It gives rise to the new formula/concept.

After bus is being located in the specified range about the predefined point, generation of notification is carried out on the Parent Type of Device.This formula plays a vital role in security as well as it leads to the energy efficiency of the application.

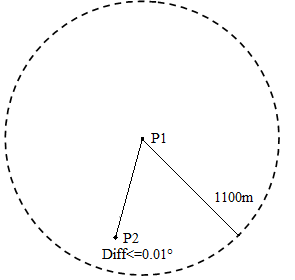


Fig.1.Proposed Architecture

1.Calculate distance between Latitude of points

ΔΘ=| Θ1 - Θ2 |

2.Calculate distance between Longitude of points

Δλ=| λ1 - λ2 |

3.Diff = ΔΘ + Δλ.

4.If (Diff< =0.01°) then P2 is expected to be within the range of 0m to 1100m from P1.

1. [↑](#footnote-ref-1)