

A Project Report

On

**“ADVANCE TRAFFIC NAVIGATION SYSTEM”**

Batch Details

|  |  |  |
| --- | --- | --- |
| Sl. No. | Roll Number | Student Name |
| 1 | SHAIK ANAS | 20201CIT0057 |
| 2 | Y. SAI KIRAN | 20201CIT0042 |
| 3 | G. KOTESWARARAO | 20201CIT0006 |

**School of Computer Science,**

**Presidency University, Bengaluru.**

Under the guidance of,

Dr. DIPALI KHUSHALRAO DAKHOLE

School of Computer Science,

Presidency University, Bengaluru

**CONTENTS**

|  |
| --- |
|  |
| **Abstract** |
| **1.INTRODUCTION** |
| 1.1 Motivation |
| 1.2 Problem Statement |
| 1.3 Objective of the Project |
| 1.4 Scope |
| 1.5 Project Introduction |
| **2.LTERATURE SURVEY** |
| 2.1 Related Work |
| **3. SYSTEM ANALYSIS** |
| 3.1 Existing System |
| 3.2 Disadvantages |
| 3.3 Proposed System |
| 3.4 Advantages |
| 3.5 work Flow of Proposed system |
| **4. REQUIREMENT ANALYSIS** |
| 4.1Function and non-functional requirements |
| 4.2 Hardware Requirements |
| 4.3 Software Requirements |
| 4.4 Architecture |
| **5. SYSTEM DESIGN** |
| 5.1 Introduction of Input design |
| 5.2 UML Diagram(class, use case, sequence, collaborative, deployment, activity, ER diagram and Component diagram) |
| 5.3 Data Flow Diagram |
| **6. IMPLEMENTATION** |
| 6.1 Modules |
| **7. PYTHON ENVIRONMENT** |
| **8. SYSTEM STUDY AND TESTING** |
| 8.1 Feasibility study |
| 8.2 Types of test & Test Cases |
| **9. CONCLUSION** |
| **10. FUTURE ENHANCEMENT** |
| **11. REFERENCES** |

Page **3** of **29**

**ABSTRACT:**

This Android application offers a user-friendly platform for community-driven road maintenance. Upon signing up and logging in, users access the main dashboard where they can promptly register complaints or requests related to road accidents. The unique selling point of the app is its ability to auto-detect the user's location when a complaint is lodged, adding an efficiency layer in addressing road concerns. Further amplifying its community-centric approach, the app allows users to relay their requests to nearby volunteers. These volunteers, upon receiving the concerns via SMS notifications, have the capability to update the complaint status. This ensures a dynamic feedback loop, fostering transparency and collaboration. Users also benefit from a history feature, enabling them to track past complaints and resolutions. In essence, this application serves as a bridge between individuals and volunteers, ensuring that road-related issues are addressed efficiently through collective effort.

# 1. INTRODUCTION

**1.1 Motivation:**

Addressing civic issues such as road maintenance and safety is crucial for community well-being. This app motivates users by empowering them to actively participate in resolving road-related concerns. By providing a user-friendly interface and leveraging location detection, it encourages prompt reporting of incidents. The integration of volunteers fosters a sense of shared responsibility and collaboration, motivating individuals to engage in collective problem-solving. Through this platform, users witness the impact of their contributions, fostering a deeper sense of community involvement and driving positive change in our urban landscapes.

**1.2 Problem Statement:**

**1.2 Problem Statement**

Road maintenance often suffers due to delayed responses to accidents and lack of efficient communication between affected individuals and authorities. Current systems lack a direct, community-driven approach for addressing road concerns promptly. This project aims to bridge this gap by creating an Android application that enables users to report accidents and road issues directly from their location. The lack of a streamlined system for users to report incidents promptly and for volunteers to respond swiftly highlights the necessity for an innovative solution like this

**1.3 Objective of the Project:**

To evaluate the efficacy and usability of the Advance Traffic Navigation System, assessing its capacity to facilitate community engagement in addressing road-related issues. This study aims to analyze the app's user interface, functionality in registering complaints, auto-location detection, volunteer engagement, and feedback mechanisms. The objective is to discern the app's effectiveness in fostering collaborative problem-solving, enhancing transparency, and providing a seamless platform for users and volunteers to collectively manage and resolve road concerns.

**1.4 Scope:**

The scope of the "Advance Traffic Navigation System" encompasses a community-oriented Android platform designed for streamlined road maintenance. Through user registration and login, the app facilitates prompt reporting of road accidents and issues. Its unique feature of auto-detecting user locations enhances efficiency in addressing complaints. The app fosters community collaboration by notifying nearby volunteers via SMS, enabling them to update complaint statuses. Users can track past complaints, promoting transparency and collective effort in resolving road-related concerns.

**1.5** **Project Introduction:**

Road safety is an essential component of urban development and public welfare. As cities expand and vehicular traffic intensifies, the occurrence of road accidents has unfortunately become a regular phenomenon. Such incidents not only disrupt the flow of traffic but also endanger lives and property. While there are mechanisms in place for reporting and addressing these mishaps, many of these systems are antiquated and lack the integration of modern technology. They often rely on manual input, leading to potential inaccuracies or delays in response. Recognizing this gap, the advent of mobile technology provides an unparalleled opportunity to innovate and streamline the reporting process. The Android application described herein harnesses this potential. By combining the convenience of smartphones with real-time location tracking and community engagement, the app offers a holistic solution. It empowers individuals to promptly report accidents and collaborate with volunteers, facilitating timely interventions and possibly saving lives, underpinning the collective commitment to safer roads.

1. **LITERATURE SURVEY**

**2.1 Related Work:**

**Ch. Ramya Keerthi, G. Shanmukh, Dr. R. Sivaram.** **Various Accident Detection Technologies and Recovery Systems with Victim Analysis "International Journal of Advanced Trends in Computer Science and Engineering (/JATCSE), Special Issue of ICCSIE 2013.** **24** **May, 2013**

In this technological revulsion world there is no time for anyone to know what happening round them they keep on moving without any care. As they give importance to their work rather than others. Due to reduce in moral values once cannot get proper help when they need.

**Manisha Ruikar National statistics of road traffic accidents in India" Journal of Orthopedics, Traumatology and Rehabilitation** **Jan-Apr 2013**

Road safety continues to be a major developmental issue, a public health concern and a leading cause of death and injury across the world. At least one out of 10 people killed on roads across the world is from India, according to the World Health Organization. The cost of road accidents is borne not only by the victims and their family, but by the economy as a whole in terms of untimely deaths, injuries, disabilities and loss of potential income. It is indeed a matter of great concern that despite the continuing efforts of the Government in this regard and our commitments for halving fatalities we have not been able to register significant progress on this front.

**S. M. Tang and H. J. Gao** **Traffic-incident detection-algorithm basedon nonparametric regression** **2005**

This paper proposes an improved nonparametric regression (INPR) algorithm for forecasting traffic flows and its application in automatic detection of traffic incidents. The INPRA is constructed based on the searching method of nearest neighbors for a traffic-state vector and its main advantage lies in forecasting through possible trends of traffic flows, instead of just current traffic states, as commonly used in previous forecasting algorithms.

1. **SYSTEM ANALYSIS**

* 1. **Existing System**

The existing system for reporting road accidents primarily relies on manual inputs, often involving direct calls to emergency services. It lacks real-time location tracking and instantaneous community engagement, leading to potential response delays and miscommunication. This traditional approach doesn't fully leverage the capabilities of modern mobile technology.

* 1. **Disadvantages**
* Relies on manual inputs for accident reports.
* Lacks real-time location tracking mechanisms.

Misses instantaneous community engagement features

**3.3 Proposed System**

The proposed Android app enhances road safety through community engagement. Users can report road accidents, with automatic location detection streamlining responses. Nearby volunteers, notified via SMS, can act and update the complaint status. The app promotes transparency with a history tracking feature, collectively addressing road concerns through community collaboration.

**3.4 Advantages**

* Enables users to report accidents with automatic location detection.
* Notifies volunteers via SMS for prompt, collaborative action.
* Features history tracking, promoting transparency and community oversight.

1. **REQUIREMENT ANALYSIS** 
   1. **Functional and non-functional requirements**

Requirement’s analysis is very critical process that enables the success of a system or software project to be assessed. Requirements are generally split into two types: Functional and nonfunctional requirements.

**Functional Requirements**: These are the requirements that the end user specifically demands as basic facilities that the system should offer. All these functionalities need to be necessarily incorporated into the system as a part of the contract. These are represented or stated in the form of input to be given to the system, the operation performed and the output expected. They are basically the requirements stated by the user which one can see directly in the final product, unlike the nonfunctional requirements.

Examples of functional requirements:

* + - 1. Authentication of user whenever he/she logs into the system
      2. System shutdown in case of a cyber-attack
      3. A verification email is sent to user whenever he/she register for the first time on some software system.

**Non-functional requirements**: These are basically the quality constraints that the system must satisfy according to the project contract. The priority or extent to which these factors are implemented varies from one project to other. They are also called non-behavioral requirements.

They basically deal with issues like:

* + - * + Portability
        + Security
        + Maintainability
        + Reliability
        + Scalability
        + Performance
        + Reusability
        + Flexibility

Examples of non-functional requirements:

* + - 1. Emails should be sent with a latency of no greater than 12 hours from such an activity.
      2. The processing of each request should be done within 10 seconds
      3. The site should load in 3 seconds whenever of simultaneous users are > 10000
  1. **Hardware Requirements:**

|  |  |  |
| --- | --- | --- |
| **.** Processor |  | - I3/Intel Processor |
| • Hard Disk |  | - 160GB |
| • RAM |  | - 8 GB |
|  |  |  |
|  |  |  |
|  |  |  |
| **3 Software Requirements**   * Operating System - Windows 10 * JDK - java * Plugin -Kotlin * SDK - Android * IDE -Android studio   Database` - sql | |

* 1. **Architecture**

**A diagram of a user activity

Description automatically generated**

1. **SYSTEM DESIGN** 
   1. **Introduction of Input Design:**

**INPUT DESIGN**

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

* What data should be given as input?
* How the data should be arranged or coded?
* The dialog to guide the operating personnel in providing input.

.

**Objectives for Input Design:**

1. Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.

2. It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.

3. When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow

**Output Design:**

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively
2. . Select methods for presenting information.
3. Create document, report, or other formats that contain information produced by the system.

The output form of an information system should accomplish one or more of the following objectives.

* Convey information about past activities, current status or projections of the
* Future.
* Signal important events, opportunities, problems, or warnings.
* Trigger an action.
* Confirm an action.
  1. **UML Diagrams:**

UML stands for Unified Modelling Language. UML is a standardized general-purpose modelling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object-oriented computer software. In its current form UML is comprised of two major components: A Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified modelling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modelling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modelling of large and complex systems.

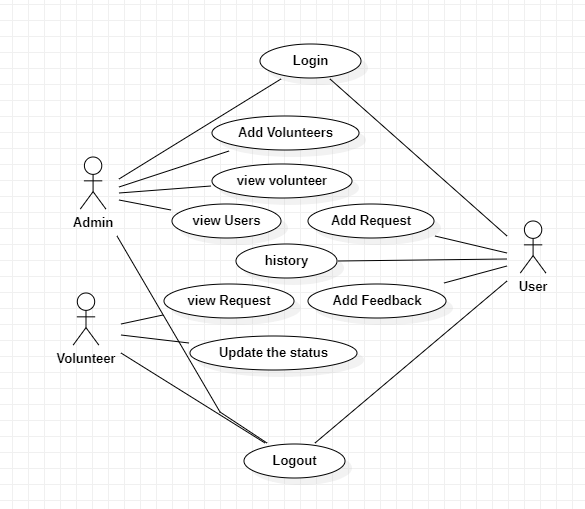
The UML is a very important part of developing objects-oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

**GOALS:**

The Primary goals in the design of the UML are as follows:

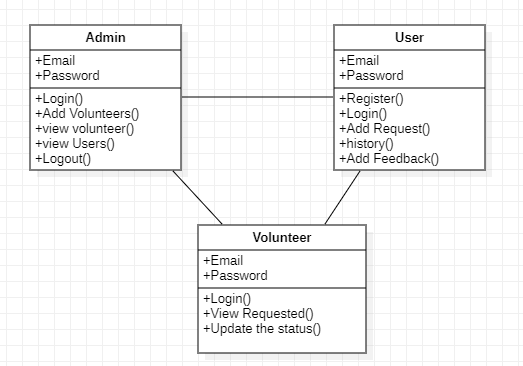
1. Provide users a ready-to-use, expressive visual modelling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modelling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.
   * 1. **Use Case Diagram:**

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.



* + 1. **Class Diagram:**

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.



* + 1. **Sequence Diagram:**

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

A diagram of a program

Description automatically generated

**5.2.4 Collaboration Diagram:**

In collaboration diagram the method call sequence is indicated by some numbering technique as shown below. The number indicates how the methods are called one after another. We have taken the same order management system to describe the collaboration diagram. The method calls are similar to that of a sequence diagram. But the difference is that the sequence diagram does not describe the object organization whereas the collaboration diagram shows the object organization.

A diagram of a server

Description automatically generated

**5.2.5 Deployment Diagram**

Deployment diagram represents the deployment view of a system. It is related to the component diagram. Because the components are deployed using the deployment diagrams. A deployment diagram consists of nodes. Nodes are nothing but physical hard ware’s used to deploy the application.

A diagram of a computer network

Description automatically generated

**5.2.6 Activity Diagram:**

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

A diagram of a program

Description automatically generated

**5.2.7 Component Diagram**:

Component diagrams are used to describe the physical artifacts of a system. This artifact includes files, executable, libraries etc. So the purpose of this diagram is different, Component diagrams are used during the implementation phase of an application. But it is prepared well in advance to visualize the implementation details. Initially the system is designed using different UML diagrams and then when the artifacts are ready component diagrams are used to get an idea of the implementation.

A diagram of a computer

Description automatically generated

**5.2.8 ER Diagram:**

An Entity–relationship model (ER model) describes the structure of a database with the help of a diagram, which is known as Entity Relationship Diagram (ER Diagram). An ER model is a design or blueprint of a database that can later be implemented as a database. The main components of E-R model are: entity set and relationship set.

An ER diagram shows the relationship among entity sets. An entity set is a group of similar entities and these entities can have attributes. In terms of DBMS, an entity is a table or attribute of a table in database, so by showing relationship among tables and their attributes, ER diagram shows the complete logical structure of a database. Let’s have a look at a simple ER diagram to understand this concept.

A diagram of a program

Description automatically generated

**5.3 DFD Diagram:**

A Data Flow Diagram (DFD) is a traditional way to visualize the information flows within a system. A neat and clear DFD can depict a good amount of the system requirements graphically. It can be manual, automated, or a combination of both. It shows how information enters and leaves the system, what changes the information and where information is stored. The purpose of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communications tool between a systems analyst and any person who plays a part in the system that acts as the starting point for redesigning a system.

A diagram of a database

Description automatically generated

A diagram of a program

Description automatically generated

**IMPLEMENTATION**

**6.1 Modules:**

**Admin:** The admin will add the volunteers and view Volunteer and users.

**Volunteer:** The Volunteer will login with in email and passwords. He can view Request and Update the status of request.

**User:** The user will register with his details, he can login with his email and password. He can add request for road accidents, view the history and add feedbacks

**7. ANDROID ENVIRONMENT**

**Software Installation**

**Software Installation of JDK kit**

This Java Development Kit (JDK) allows you to code and run Java programs. It's possible that you install multiple JDK versions on the same PC. But it’s recommended that you install only latest version.

## How to install Java for Windows

Following are the steps for JDK 8 free download for 32 bit or JDK 8 download 64 bit and installation

**Step 1)** Go to [link](https://www.oracle.com/java/technologies/javase-downloads.html) Click on JDK Download for Java

**Step 2)** Next,

**Step 3)** when you click on the Installation link the popup will be open. Click on I reviewed and accept the Oracle Technology Network License Agreement for Oracle Java SE and you will be redirected to the login page. If you don't have an oracle account you can easily sign up by adding basics details of yours.

**Step 4)** once the Java JDK 8 download is complete, run the exe for install JDK. Click Next

**Step 5)** Select the PATH to install Java in Windows and click next.

**Step 6)** Once you install Java in windows, click Close

The PATH variable gives the location of executable like javac, java etc. It is possible to run a program without specifying the PATH but you will need to give full path of executable like **C:\Program Files\Java\jdk-13.0.1\bin\javac A.java** instead of simple **javac A.java**

The CLASSPATH variable gives location of the Library Files.

Let's look into the steps to set the PATH and CLASSPATH

**Step 1)** Right Click on the My Computer and Select the properties

**Step 2)** Click on advanced system settings

**Step 3)** Click on Environment Variables

**Step 4)** Click on new Button of User variables

**Step 5)** Type PATH in the Variable name.

**Step 6)** Copy the path of bin folder which is installed in JDK folder.

**Step 7)** Paste Path of bin folder in Variable value and click on OK Button.

**Step 8)**You can follow a similar process to set CLASSPATH

**Step 9)** Click on OK button

**Step 10)** Go to command prompt and type java commands.

If you see a screen like below, Java is installed.

**2.Android Studio IDE and SDK Installation**

Installing Android software is probably the most challenging part of this project. It takes times - from 30 minutes to *n* hours to forever - depending on your luck, your programming knowledge, and your PC. You probably need a fairly decent PC (with 8GB RAM) and 10GB of free disk space to run the Android emulator!!! Running on "actual" Android phone/tablet requires much lesser resources.

##### **Step 0: Pre-Installation Check List**

**Step 1: Install "Android Studio IDE**

##### **Step 2: Installing Android SDK**

1. Launch Android Studio ⇒ It will run the "setup" wizard for the first launch.

2.(For Windows) Use "File Explorer" to check the SDK installed directory.

3. Also use "Android Studio" to check the SDK packages installed by selecting "Configure"

**STEPS FOR EXECUTING THE PROJECTS**

**Step 1:**

Open Android Studio

**Step2:**

Choose a virtual device or Physical device from the menu

**Step3:**

Click on the project Run

**Step4:**

View the application performance on virtual or Physical device.

**SOFTWARE DEVELOPMENT LIFE CYCLE**

**Principles of Agile model:**

**SOFTWARE ENVIRONMENT**

Android is a software stack for mobile devices that includes an operating system, middleware and key applications. Google Inc. purchased the initial developer of the software, Android Inc., in 2005.

Features: -

* **Application framework** enabling reuse and replacement of components
* **Dalvik virtual machine** optimized for mobile devices
* **Integrated browser** based on the opensource Web Kit engine
* **Optimized graphics** powered by a custom 2D graphics library; 3D graphics based on the OpenGL ES 1.0 specification (hardware acceleration optional)
* **SQLite** for structured data storage
* **Media support** for common audio, video, and still image formats (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF)
* **GSM Telephony** (hardware dependent)
* **Bluetooth, EDGE, 3G, and WIFI** (hardware dependent)
* **Camera, GPS, compass, and accelerometer** (hardware dependent)
* **Rich development environment** including a device emulator, tools for debugging, memory and performance profiling, and a plugin for the Eclipse IDE

### **Important Android components**

Step 1: preparing your development computer

Step 2: Downloading the SDK Starter Package

Step 3: Installing the ADT plugin for Eclipse

Step 4: adding platforms and other components

The Android Repository offers these types of components:

* **SDK Tools** — Contains tools for debugging and testing your application and other utility tools. These tools are installed with the Android SDK starter package and receive periodic updates. You can access these tools in the <sdk>/tools/ directory of your SDK. To learn more about them, see SDK Tools in the developer guide.
* **SDK Platform-tools** — Contains platform-dependent tools for developing and debugging your application. These tools support the latest features of the Android and platform are typically updated only when a new platform becomes available. You can access these tools in the <sdk>/platform-tools/ directory. To learn more about them, see Platform Tools in the developer guide.
* **Android platforms** — An SDK platform is available for every production Android platform deployable to Android-powered devices. Each SDK platform component includes a fully compliant Android library, system image, sample code, and emulator skins. To learn more about a specific platform, see the list of platforms that appears under the section "Downloadable SDK Components" on the left part of this page.
* **USB Driver for Windows** (Windows only) — Contains driver files that you can install on your windows computer, so that you can run and debug your applications on an actual device. You do not need the USB driver unless you plan to debug your application on an actual Android-powered device. If you develop on Mac OS X or Linux, you do not need a special driver to debug your application on an Android-powered device. See Using Hardware Devices for more information about developing on a real device.
* **Samples** — Contains the sample code and apps available for each Android development platform. If you are just getting started with Android development, make sure to download the samples to your SDK.
* **Documentation** — Contains a local copy of the latest multi-version documentation for the Android framework API.

## History

* Eclipse began as an IBM Canada project. It was developed by Object Technology International (OTI) as a Java-based replacement for the Smalltalk based Visual Age family of IDE products,[4] which itself had been developed by OTI.[1] In November 2001, a consortium was formed to further the development of Eclipse Eclipse as open source. In January 2004, the Eclipse Foundation was created.[5]

Eclipse Platform

The Eclipse Platform provides the core frameworks and services upon which all plug-in extensions are created. It also provides the runtime in which plug-ins are loaded, integrated, and executed. The primary purpose of the Platform is to enable other tool developers to easily build and deliver integrated tools.

Features include:

* Supports the construction of a variety of tools for application development
* Supports an unrestricted set of tool providers, including independent software vendors (ISVs)
* Supports tools to manipulate arbitrary content types (e.g., HTML, Java, C, JSP, EJB, XML, and GIF)
* Facilitates seamless integration of tools within and across different content types and tool providers
* Supports both GUI and non-GUI-based application development environments

Java Development Tools (JDT)

The JDT project provides the tool plug-ins that implement a Java IDE supporting the development of any Java application, including Eclipse plug-ins. It adds a Java project nature and Java perspective to the Eclipse Workbench as well as a number of views, editors, wizards, builders, and code merging and refactoring tools. The JDT project allows Eclipse to be a development environment for itself.

Features include:

* Java projects with source files arranged in package directories
* Editing with keyword and syntax colouring, outline showing declaration structure
* Code formatter
* Refactoring
* Search
* Compare
* Compile - JCK-compliant Java compiler
* Run Java programs in a separate target Java virtual machine
* Debug programs with JPDA-compliant Java virtual machine

Android Source Code

The following step is optional.

During Android development it is very useful to have the Android source code available as Android uses a lot of defaults.

Haris Peco maintains plugins with provides access to the Android Source code. Use the Eclipse update manager to install two of his plugins.

# 8. SYSTEM STUDY AND TESTING

**8.1 FEASIBILITY STUDY**

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

* Economical feasibility
* Technical feasibility
* Social feasibility

**Economical feasibility**

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

**Technical feasibility**

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

**Social feasibility**

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

## 8.2 SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product it is the process of exercising software with the intent of ensuring that the

Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.  **Types of tests**

**Unit testing**

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

**Integration testing**

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components. **Functional test**

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid input: identified classes of valid input must be accepted.

Invalid input: identified classes of invalid input must be rejected.

Functions: identified functions must be exercised.

Output: identified classes of application outputs must be exercised.

Systems/procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

**White Box Testing**

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

**Black Box Testing**

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

**Test objectives**

* All field entries must work properly.
* Pages must be activated from the identified link.
* The entry screen, messages and responses must not be delayed.

**Features to be tested**

* Verify that the entries are of the correct format• No duplicate entries should be allowedAll links should take the user to the correct page

**IV. Acceptance Testing**

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

# 9.CONCLUSION

The Android application revolutionizes community-driven road maintenance by seamlessly integrating user reporting with volunteer responses. Leveraging automatic location detection and real-time notifications, it ensures efficient and collaborative handling of road accidents. The inclusion of a history tracking tool further emphasizes its commitment to transparency, making it an indispensable tool for safer roads.

# 10. FUTURE ENHANCEMENT

Future enhancements for the "Advance Traffic Navigation System" could involve incorporating machine learning algorithms to predict potential road hazards based on historical data, thus proactively preventing accidents. Integrating real-time traffic updates and alternative route suggestions could offer users optimized navigation choices. Enhanced communication features, such as in-app messaging between users and volunteers, might streamline issue resolution. Additionally, implementing a gamification element to incentivize volunteer participation and adding language support for broader accessibility could further enrich the app's functionality and user engagement. These developments would fortify the application's effectiveness in fostering community-driven road maintenance.

# 11.REFERANCES

[I] Ch. Ramya Keerthi, G. Shanmukh, Dr. R. Sivaram, "Various Accident Detection Technologies and Recovery Systems with Victim Analysis "International Journal of Advanced Trends in Computer Science and Engineering (/JATCSE), Special Issue of ICCSIE 2013 - Held during 24 May, 2013 Bangalore, India

[2] Manisha Ruikar "National statistics of road traffic accidents in India" Journal of Orthopedics, Traumatology and Rehabilitation Vol-6 Issue-I, Jan-Apr 2013.

[3] S. M. Tang and H. J. Gao, "Traffic-incident detection-algorithm basedon nonparametric regression," IEEE Transactions on Intelligent Transportation Systems, vol. 6, 2005, pp. 38-42.

[4] Yong-Kul Ki ,Jin-Woo Kim and Doo-Kwon Baik," A Traffic Accident Detection Model using Metadata Registry ",Conference on Software Engineering Research, Management and Applications, 2006 ppno: 0-7695-2656-X.

[5] Jung Lee, "An accident detection system on highway through CCTV with calogero-moser system , Conference on Communications (APCC), 2012 18th Asia-Pacific ppno: 978-1- 4673-4726-6.

[6] L. Chuan-zhi, H. Ru-fu, Y.E. Hong-wu, "Method of Freeway Incident Detection Using wireless Positioning," in Proceedings of the IEEE International Conference on Automation and Logistics, 2008, pp. 2801 -2804.