A
Project Report
on
Navigation System Depicting Live Potholes

Submitted for the Course of BE in Computer Engineering by

Mr. Ritesh Pawar [B191204250]
Ms. Anushree Suryavanshi [B191204267]
Ms. Pranjal Patil [B191204248]
Mr. Shankar Khadar [B191204228]

Under the guidance of

Prof. Pramod C. Patil

Department of Computer Engineering

Guru Gobind Singh College of Engineering and Research Center

Nashik-422009

2022-23

GURU GOBIND SINGH COLLEGE OF ENGINEERING AND RESEARCH CENTER

Nashik-422009

2022-2023

Department of Computer Engineering

CERTIFICATE

This is to certify that the PROJECT REPORT entitled

Navigation System Depicting Live Potholes

is submitted as partial fulfilment of the

Project Examination BE in Computer Engineering

BY

Mr. Ritesh Pawar [B191204250]
Ms. Anushree Suryavanshi [B191204267]
Ms. Pranjal Patil [B191204248]
Mr. Shankar Khadar [B191204228]

____

Project Guide
Prof. P. C. Patil

____

Project Coordinator
Prof. P. K. Bachhav

____

Head of the Department
Prof. S. G. Shukla

____

Principal
Dr. N. G. Nikam

SAVITRIBAI PHULE PUNE UNIVERSITY

CERTIFICATE

This is to certify that,

Mr. Ritesh Pawar [B191204250]

Ms. Anushree Suryavanshi [B191204267]

Ms. Pranjal Patil [B191204248]

Mr. Shankar Khadar [B191204228]

of BE in Computer Engineering was examined in the Project Examination entitled

Navigation System Depicting Live Potholes

on 01/06/2023

At

Department of Computer Engineering

GURU GOBIND SINGH COLLEGE OF ENGINEERING AND RESEARCH CENTER

Nashik-422009

2022-2023

______________________ ______________________

Internal Examiner *External Examiner*

Acknowledgement

It is a great pleasure to acknowledge those who extended their support, and contributed time and psychic energy for the completion of this project work.

At the outset, I would like to thank my project guide **Mr. Pramod C. Patil**, for his valuable and skillful guidance, assessment and suggestions from time to time improved the quality of work in all respects. I would like to take this opportunity to express my deep sense of gratitude towards him, for his invaluable contribution in completion of this project.

I am also thankful to **Mr. Sandip. G. Shukla**, Head of Computer Engineering Department for his timely guidance, inspiration and administrative support without which my work would not have been completed.

I am also thankful to the all staff members of Computer Engineering Department and Librarian, Guru Gobind Singh College of Engineering and Research Center, Nashik.

Also I would like to thank my colleagues and friends who helped me directly and indirectly to complete this project.

Mr. Ritesh P. Pawar
Ms. Anushree K. Suryavanshi
Ms. Pranjal V. Patil
Mr. Shankar R. Khadar

Contents

List of Figures

List of Tables

Abstract

Potholes are becoming a growing cause of concern, resulting in numerous road accidents across the country. To address this problem, a smart pothole reporting system has been developed that can report and address potholes as soon as they occur. The system ensures transparency and accountability between citizens and the government while being user-friendly. The smart system employs an image recognition-based method that uses machine learning techniques, such as the Classification Algorithm using TensorFlow, to identify potholes. The system collects data and uses it to represent areas with a higher density of potholes on a map. If the density of potholes in a certain area is high, the road appears red on the map. Conversely, if the potholes have been patched up or there are no potholes, the road appears green. Color codes are used to warn and alert drivers about the observed road conditions. The smart pothole reporting system enables prompt reporting and resolution of potholes by the appropriate authorities. The system allows the government to prioritize areas that require attention and repair, leading to a more efficient allocation of resources. Additionally, the system helps to eliminate the threat posed by potholes and ensures safer roads for everyone. In conclusion, the smart pothole reporting system is an easy-to-use solution that addresses the growing concern about potholes and provides a safer road network for citizens. The system ensures transparency and accountability between the government and citizens, using advanced technology to identify and report potholes promptly.

.

Keywords:-*Potholes, Smart Reporting, Classification Algorithm, TensorFlow, Colour code, Alert*

Chapter 1

Introduction

In this, we are going elaborate introduction of the proposed system. This chapter includes an overview of the system, its motivation, and its objective of the system. This chapter also explains how the report is organized.

1.1 Overview

The Ministry of Road Transport and Highway (MoRTH) survey reveals that more than 5000 people die yearly due to bad road conditions and potholes. The increasing number of accidents caused by potholes is a massive issue, and no serious actions have been taken to address it. Maharashtra is currently the second most pothole-ridden state in India. Currently, road maintenance companies require several working hours to roughly estimate the damage on a road. It becomes difficult to spot potholes during the night due to reduced visibility. While there are systems available that allow users to report potholes by using their mobile phones, the current system has many limitations. It cannot guarantee the presence of potholes at the reported location, and it also lacks transparency in the repair process. Although users can upload pictures of potholes, there is no algorithm to verify the authenticity of the images, and no standard definition of what is considered a pothole. While Google Maps offers many advanced features, it does not provide real-time updates on road conditions. So, a smart pothole reporting system is designed to report problems to the authorities as soon as they arise and maintain transparency between the government and citizens. The system is user-friendly and efficient in resolving pothole-related threats throughout the nation. While potholes are being repaired, the system uses navigation to guide citizens about road conditions.

To address the problems related to potholes using an application where users can upload images of potholes along with their coordinates, which results in an indication of road conditions by using a color code on the Navigation system. The road with the highest intensity of potholes should appear as red on the map, while under-construction roads should be denoted by a yellow color, and roads with no flaws are shown in green. This system should improve road understanding, allowing drivers to analyze the road properly using the provided information.

1.2 Aim

The aim of the project is to address the growing concern of potholes causing road accidents. This is being achieved by developing a smart pothole reporting system that utilizes advanced technology for prompt detection and resolution. The system ensures transparency and accountability between citizens and the government by providing a user-friendly platform for reporting potholes and monitoring their resolution. It employs image recognition and machine learning techniques, such as the Classification Algorithm using TensorFlow, to accurately identify potholes. The system collects data and maps areas with a higher density of potholes on a color-coded map, alerting drivers to road conditions. Prompt reporting enables authorities to quickly address reported potholes, and the system assists the government in efficiently allocating resources for repairs. Ultimately, the goal is to eliminate the threat of potholes and create safer roads for everyone.

1.3 Objectives

1. To gather information about the pothole's location.

2. To Share the location and coordinates of potholes with corresponding authorities of that area.

3. To check the progress on fixing the potholes.

4. To implement the data in the navigation system used by cars which are denoted by color codes. (Red- Pothole ahead, Yellow- Under construction, Green- Safe)

5. To allocate contractors particular potholes to fix.

6. To provide facility to the customers to their feedback about the same.

1.4 Organization of Report

The rest of this report is organized in the following manner. In all chapters, related contents are described in detail.

- **Introduction (Chapter 1)**: In this chapter, the overview of existing systems and their problem is discussed. This chapter describes the aim, motivation, and objectives of the software system.
- **Literature Survey (Chapter 2)**: In this chapter, Related work done in the Previous papers have advantages and disadvantages. Related information is available in standard Books, Journals, Transactions, Internet Websites, etc. is discussed.
- **Software Requirement Specification (Chapter 3)**: In this chapter, the detailed description of requirements is specified.
- **System Design (Chapter 4)**: This chapter discusses the proposed system with the help of system architecture, system design, and UML diagrams
- **Technical Specifications (Chapter 5)**: This chapter, discusses the technical details used in the project
- **Project Estimation Schedule and Team Structure (Chapter 6)**: This chapter discusses project estimate, brief of COCOMO model, and related calculation and team structure
- **Software Implementation (Chapter 7)**: This chapter discusses important module and algorithm also business logic and archite
- **Software Testing (Chapter 8)**: This chapter gives a briefing about testing for various modules
- **Software Testing (Chapter 9)**: This chapter discusses about installation and uninstallation of project as well as maintenance
- **Conclusion and Future Scope (Chapter 10)**: This chapter summarizes and concludes the project report and give the future scope.
- **Plagiarism Report(Chapter 11)**: This chapter shows the plagiarism report.

Chapter 2

Literature Survey

This chapter provides a detailed literature survey of the proposed system. Based on Many existing systems have been studied which provide let user post on the wall upload graphical content and share them.

The design of a system providing a customizable Digital Signature Algorithm for digital signature verification, based on OCR technique text extraction of the done would be done. Therefore, in what follows, we survey the literature in both these fields.

Indian pothole detection based on CNN and anchor-based deep learning method.:

Mallikarjun Anandhalli., and Vishwanath Baligar, In Proc. in International Journal of Information Technology. A study from February 2022 demonstrates the application of deep learning techniques for pothole identification. For the purpose of spotting potholes on the road, a convolutional neural network is employed. With the use of this method, image flaws are found and extraneous images of other objects are avoided. The pothole in the image is detected using deep learning methods as well as the layers in the CNN algorithm. In Proc. in International Journal of Information Technology · February 2022.[1]

A Real-Time Pothole Detection Approach for Intelligent Transportation System:

Hsiu-WenWang, Chi-Hua Chen, Ding-Yuan Cheng, ChunHao Lin, and Chi-Chun. An intelligent transport system is a cutting-edge program that intends to offer cutting-edge services connected to various modes of transport and traffic management and to give users the information they need to use transport networks in a safer, more effective, and "smarter" way. The intelligent transportation system needs to be able to detect potholes. Road accidents caused by potholes are less common since it detects them in real time and prevents accidents on the roadways. 2015 in the Proceedings of HsiuWen Wang et al., Taiwan, China.[2]]

Convolutional neural networks-based pothole detection using thermal imaging

Aparna, Yukti Bhatia, Rachna Rai, Varun Gupta, Naveen Aggarwal, and Aparna Akula. Convolutional neural networks are a kind of artificial neural network that are commonly used to analyze visual data. Deep learning is relevant to this. Thermal imaging is a type of infrared imaging that uses a camera to gather and create images of objects using infrared radiation that is emitted from the objects throughout a process. It helps in determining the potholes' width and depth as well as in evaluating the specifics of their structural design. Computer and Information Sciences 34 (2022) 578-588, King Saud University Privacy Journal, February 2019. [3]

Real-Time Pothole Detection using Android Smartphones with Accelerometers

Artis Mednisy, Girts Strazdinsy, Reinholds Zviedris, Georgijs Kanonirs, and Leo Selavoy.An accelerometer is a piece of electromechanical equipment used to monitor acceleration forces. These forces may be static, like the gravitational pull that is always there, or dynamic, like those used by many mobile devices to sense motion or vibrations. Here, the user can utilize the camera and accelerometer on their smartphone to find potholes. In Proc. of Institute of Electronics and Computer Science 14 Dzerbenes Str., Riga, LV 1006, Latvia Faculty of Computing University of Latvia 19 Raina Blvd., Riga, LV 1586, Latvia. [4]

Road Pothole Detection using Deep Learning Classifiers :

Surekha Arjapure, D. R. Kalbande.A classifier is a particular kind of machine learning algorithm used to categorize data input. Machines can recognize and extract features from photographs thanks to deep learning. The photographs will be classified by classifiers, who will also assist in spotting any potholes. In Proc. International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878 (Online), Volume8 Issue-6, March 2020.[5]

Web-based framework for smart parking system :

Alharbi A, Halikias G, Yamin. This study shows how parking on specific streets and in important metropolitan areas greatly increases traffic congestion. The proposed web application, developed using the OCR algorithm, introduces the idea of an intelligent system to address this issue by enabling users to pre-reserve seats automatically before arriving at the venue and then modifying the process in response to the automatic detection of the composite plate. On pages 1495–1502 of JJT Information Technology.[6]

Chapter 3

Software Requirement Specification

3.1 Introduction

Potholes pose a significant threat, leading to numerous accidents every year, with heavy rains being a major contributing factor. To address this critical issue, we propose a system that empowers citizens to effectively deal with this situation. Our system allows citizens to register complaints through an application, either by logging in or signing up. Upon raising a complaint and uploading a photo of the pothole, the system verifies the reported pothole's authenticity. Once the verification is complete, our dedicated team of administrators assigns a contractor based on the pothole's location. The assigned contractor takes up the task and initiates the necessary repair work. Throughout the process, the contractor provides regular updates regarding the progress of the pothole repair. This approach ensures a streamlined feedback loop, where citizen inputs regarding potholes are effectively utilized to allocate contractors through the administrative panel.

The system maintains a comprehensive database that stores all relevant information, enabling continuous monitoring and updates regarding road conditions. To provide users with a visual representation of the identified potholes, we utilize a red flag symbol on the map interface. This symbol indicates the presence of reported potholes, helping users navigate and avoid areas with potential road hazards. By actively involving citizens in the reporting and resolution process, our system ensures timely and efficient handling of pothole complaints. The integration of contractors, administrators, and the database facilitates a collaborative approach to tackle road conditions and improve overall infrastructure. Together, we can create safer roads and enhance the driving experience for everyone.

3.1.1 Purpose

There are currently thousands of death cases in India each year due to potholes. Maharashtra is identified as the second most pothole-ridden state. The number of accidents caused by potholes is increasing on a daily basis, presenting a significant issue that has not been adequately addressed with serious actions. Existing systems suffer from certain drawbacks, including a lack of transparent communication, a slow process of repairing potholes, and manual information gathering that consumes a significant amount of time. Consequently, the purpose of the proposed system is to overcome all these problems by expediting the repairing process and gathering data directly from the citizens themselves. This data will be efficiently stored in the database, providing readily accessible information to help citizens better comprehend the current road conditions.

3.1.2 Intended audience and reading suggestion

- **Government Authorities:** Municipal and transportation departments responsible for road maintenance and infrastructure planning could benefit from understanding the potential of smart pothole systems and navigation systems to improve road conditions and enhance public safety.

- **Road Maintenance and Repair Agencies:** Organizations and contractors involved in road maintenance and repair would be interested in learning about innovative solutions that can optimize their operations, increase efficiency, and reduce accidents caused by potholes.

- **Technologists and Engineers:** Professionals in the field of technology and engineering, including software developers, data analysts, and IoT specialists, would find the report valuable for exploring the technical aspects of developing and implementing smart pothole systems and navigation tools.

- **Researchers and Academics:** : Scholars and researchers studying transportation infrastructure, smart city technologies, or road safety would benefit from the report as a resource for understanding the current advancements and potential future developments in the field.

- **Citizen:**The report could also be accessible to the general public, including drivers and commuters, who are interested in gaining knowledge about innovative solutions that could improve road conditions and enhance their overall travel experience.

3.1.3 Project Scope

1. Live data collection: The project scope involves developing a mechanism to collect real-time data about potholes. This can be achieved through various methods such as crowd-sourcing, sensor-equipped vehicles, or municipal data sources.

2. Data processing and mapping: The collected data needs to be processed, validated, and mapped onto the navigation system. This step involves developing algorithms and software components to handle the data and ensure its accuracy and reliability.

3. User interface integration: The project scope includes designing and implementing a user-friendly interface that overlays the live pothole data onto the navigation system. This can involve developing mobile applications or integrating with existing navigation apps.

4. Notifications and alerts: The project should provide timely notifications and alerts to drivers about the presence of potholes on their selected routes. This can be done through visual cues, audio alerts, or haptic feedback, depending on the user's preferences.

5. Feedback mechanism: To improve data accuracy and maintain up-to-date information, the project should incorporate a feedback mechanism for users to report new potholes, confirm existing ones, or provide other relevant information.

6. Collaboration with authorities: It's essential to establish collaboration with local authorities responsible for road maintenance to ensure the accuracy of the data, streamline repairs, and contribute to overall road safety.

The project has a wide scope. The project scope is to take less time in the verification of a document that needs to be verified or the scholarship and admission process. The user will be notified as quickly as possible after the document is verified. The document verification process has a demand in every field. The system will take a document and verify it with the help of an original document. Also, this system will help to recognize fake or forged documents which are used. the system can help to reduce the use of forged documents.

3.1.4 Design and Implementation Constrain

1. Until the pictures of potholes are uploaded by the citizens, the system does not come to know about the location of the potholes by itself.

2. The application contains only English Language. Hence, there is a linguistic barrier between the regional languages of the citizen.

3. For the live navigation functionality in the application, there is a requirement for internet connectivity.

   +

3.1.5 Assumption and Dependencies

1. Availability of User Input: The project assumes that users will actively contribute by uploading photos of potholes and providing relevant information about their location. The success and accuracy of the system rely on the active participation and engagement of users.

2. Internet Connectivity: The system depends on a stable and reliable internet connection for various functionalities such as uploading photos, retrieving and updating data, and displaying real-time information on the navigation map. The project assumes that users will have access to an internet connection during their interaction with the system.

3. Accuracy of Image Recognition: The project relies on the accuracy and effectiveness of the image recognition algorithm used to identify potholes in the uploaded photos. It assumes that the algorithm will accurately detect and classify potholes, minimizing false positives and false negatives.

4. Integration with Mapping Service: The system assumes seamless integration with a mapping service, such as Google Maps, to provide real-time navigation information and display the location of potholes. It depends on the availability and proper functioning of the mapping service's APIs and services.

5. User Adoption and Engagement: The success of the project depends on users adopting the system and actively using it to report and navigate around potholes. The assumption is that users will find value in the system and continue to engage with it, providing valuable data and feedback.

3.2 System Features

3.2.1 Photo Uploading:

- The system allows users to upload pictures of potholes through the application.

- Users can directly contact the relevant authorities by uploading the photos, eliminating the need for a cumbersome process.
- This feature enables instant action by higher authorities as the information is accessible to all.

3.2.2 Transparent Communication:

- After a user completes the photo uploading process, they can engage in direct communication with the high authorities.
- The high authorities are responsible for contacting the contractors and assigning them the task.
- All conversations between users, high authorities, and contractors are transparent, fostering transparency among citizens as well.

3.2.3 Navigation:

- The system includes navigation functionality along with a color code feature.
- The navigation system, using color codes, informs citizens about road conditions from the source to their destinations, helping them avoid potential problems.
- Road conditions are represented using red, yellow, and green colors. Red denotes bad road conditions, yellow represents under-construction areas, and green signifies safe roads to travel.

3.2.4 Feedback:

- Citizens can provide feedback once their complaint is resolved.
- As soon as the work is completed, citizens can update their feedback through the application.
- This feedback mechanism allows users to share their reviews and experiences, contributing to the overall improvement of the application.

3.3 External Interface Requirement

3.3.1 User Interface

The system comprises a website and an Android application, catering to mobile phones and desktops. Upon accessing the web application, users are presented with a registration and login page. Each type of user (citizen, admin, and contractor) has distinct account designs tailored to their specific information requirements.

1. Within the citizen's application interface, a comprehensive Dashboard is displayed, offering options such as Create Case, All Cases, and View Map sections. Through the Create Case section, citizens can raise complaints by capturing images of potholes using their device's camera or uploading pre-existing images. The All Cases section provides visibility into the status of all cases, including pending, resolved, and those currently in progress. The View Map option enables users to identify areas heavily affected by potholes.

2. Upon logging into the admin account on the website, administrators are presented with various options such as Home, Users, Assign, AssignDetails, About, and PotHole Support. Admins have access to view new complaints related to potholes, as well as an overview of active cases currently under investigation or being addressed. They can also review the list of identified potholes and the corresponding contractors assigned to address each issue. Furthermore, admins can access the feedback portal, providing valuable insights and user perspectives.

3. Contractors logging into their accounts can view the potholes assigned to them, along with their respective locations displayed on maps. Contractors have the ability to update the status of each assigned pothole, reflecting progress and completion. Additionally, contractors can access the feedback portal, enabling them to gather valuable feedback and insights for continuous improvement.

3.3.2 Hardware Interface

The navigation system depicting live potholes relies on the seamless interaction between various hardware components to facilitate efficient and accurate reporting of potholes. The following hardware interfaces are crucial for the system's operation:

1. Camera Interface: The system interfaces with cameras or imaging devices to capture images of potholes. It should support compatibility with different camera

models and ensure smooth integration to enable users to capture high-quality images.

2. GPS Interface: The hardware interface incorporates GPS technology to determine the precise location of potholes. It communicates with the GPS module or receiver to retrieve accurate geographical coordinates for each reported pothole.

3. Network Connectivity: The hardware interface facilitates seamless communication between the system and network infrastructure. It supports wireless protocols, such as Wi-Fi or cellular connectivity, to transmit data, including images and GPS coordinates, to the central server for processing and analysis.

4. Display Interface: The system requires a display interface to present information to users effectively. It should be compatible with various display technologies, including mobile screens or desktop monitors, ensuring clear visualization of the smart map, color-coded roads, and relevant alerts.

5. Server Infrastructure: The hardware interface encompasses the server infrastructure that hosts the smart pothole reporting system. It includes powerful servers with adequate storage capacity to store and process the collected data, run machine learning algorithms, and generate real-time insights for users.

3.3.3 Software Interface

The system incorporates several software interfaces to facilitate seamless communication and interaction between its components. The following software interfaces play crucial roles in the system's operation:

1. Camera Interface: The system utilizes the camera interface of mobile phones to enable citizens to capture and upload photos of potholes. The camera functionality is integrated into the system to facilitate the image capture process. Verification of potholes in the uploaded photos is performed using a Classification algorithm.

2. SQL Database: The system employs an SQL database to store relevant data related to reported potholes. The database is generated using a Code First approach, ensuring efficient management and storage of pothole information. This interface enables the system to store and retrieve data seamlessly.

3. .NET Platform: The system is built on the .NET platform, leveraging its robust features and libraries for development. The user interface (UI) design is implemented

using HTML, CSS, and Bootstrap, ensuring a visually appealing and user-friendly interface.

4. Google Maps API: The navigation functionality within the system relies on the Google Maps API. It integrates the Google Maps service, allowing users to access accurate and reliable navigation features. This interface facilitates real-time mapping, route planning, and visual representation of road conditions.

3.4 Non Functional Requirements

3.4.1 Performance Requirements:

The system has specific performance requirements to ensure efficient functionality and user satisfaction. The following performance requirements are essential for the system's operation: ant uploads a document, it should be verified within a short period of time.

1. Real-time Pothole Verification: The system must verify the uploaded pothole photographs promptly. Verification should occur as soon as the citizen uploads the photo to the platform, ensuring timely processing and response. This requirement ensures quick identification and classification of potholes.

2. Seamless Internet Connectivity: Active internet connectivity is crucial for the system's performance. Users must have an active internet connection to upload photos, update the color code on the maps, and access the navigation system. A stable and reliable internet connection is necessary to enable smooth data transfer and real-time interaction with the system.

   Color Code Update: When a pothole is accepted and verified, the color code representing the road condition should be updated on the maps immediately. This real-time update allows users to view the current road conditions accurately and make informed decisions while navigating.

3.4.2 Security Requirements :

The smart pothole reporting system incorporates essential security measures to protect user data and ensure secure access to user accounts. The following security requirements are crucial for maintaining the integrity and confidentiality of user information:

1. Authentication and User Data Protection: To access their accounts, users (citizens, contractors, and admins) must provide authentication credentials, such as usernames and passwords. The system must implement robust authentication mechanisms to verify the user's identity and protect against unauthorized access. User passwords should be securely stored using industry-standard encryption techniques to safeguard sensitive information.

2. Safeguarding User Profile Information: The login form of the system collects additional user information, including the city, state, and current location of the citizen. This data should be treated as confidential and protected from unauthorized access or misuse. Appropriate security measures, such as data encryption and access controls, should be implemented to ensure the confidentiality and integrity of user profile information.

3. Secure Data Transmission: When users interact with the system, particularly during the login process or while submitting sensitive information, secure data transmission protocols (e.g., HTTPS) should be utilized to encrypt the communication between the user's device and the system's servers. This prevents unauthorized interception or tampering of data during transmission.

4. User Access Control: The system should enforce proper access control mechanisms to ensure that users can only access the data and features relevant to their roles. Role-based access control (RBAC) can be implemented to grant appropriate permissions and restrict unauthorized access to sensitive functionalities or confidential information.

3.4.3 Software Quality Attribute:

1. Availability: The system ensures 24/7 availability for citizens to upload pothole photographs. Instant verification of the uploaded photo is performed, and notifications are promptly sent to the citizen, admin, and allocated contractor.

2. Correctness: The pothole verification algorithm utilized in the system is based on Binary Classification. This algorithm accurately determines the presence of potholes in uploaded photographs, ensuring the correctness of the verification process.

3. Adaptability: The system is designed to be compatible with various mobile and web browsers. It maintains consistent functionality regardless of the browser used, such as Google Chrome, Opera Mini, Microsoft Firefox, and others.

4. Interoperability: The system seamlessly exchanges information with other software components, specifically Google Maps. It ensures smooth data sharing and effective communication between systems, enhancing interoperability.

5. Flexibility: The system's interface remains consistent and user-friendly, regardless of the user's access method, whether through a mobile browser or a web browser. The system adapts to different devices and platforms, providing a flexible user experience.

6. Maintainability: In case of any delays or issues encountered in the specified processes, the admin promptly acknowledges and resolves them to maintain system performance. Regular maintenance activities are carried out to ensure optimal functionality.

7. Portability: The system exhibits portability, enabling it to run on different computers and operating systems. It can be deployed and accessed across various environments without compromising functionality.

8. Robustness: The system incorporates robust measures to handle invalid inputs. If a user uploads a photograph that does not contain pothole images, the photo is not considered or stored in the database. This ensures the system's robustness by minimizing the acceptance of invalid inputs.

3.5 Other Requirement

3.5.1 Database Requirements:

The system utilizes a dynamic database creation approach, where tables are automatically generated based on the entity models. Upon image upload and acceptance, tuples representing pothole information are automatically formed in the pre-existing database.

- **Dynamic Database Creation:**The system employs a code-first approach to dynamically create the database. This means that the database is generated based on the system's code and entity models.

- **Automatic Table Creation:**With the code-first approach, there is no need to manually create tables in the database. The tables are automatically created based on the defined entity models.

- **Tuple Formation:**When a user uploads an image, and it is accepted, a tuple representing the pothole information is automatically created in the existing database. This tuple is added to the relevant table in the database.

3.6 Analysis Model

3.6.1 Data Flow Diagram

A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles, and arrows, plus short text labels, to show data inputs, outputs, storage points, and the routes between each destination. Data flowcharts can range from simple, even hand-drawn process overviews, to in-depth, multi-level DFDs that dig progressively deeper into how the data is handled. They can be used to analyze an existing system or model a new one. A data flow diagram can dive into progressively more detail by using levels and layers, zeroing in on a particular piece. DFD levels are numbered 0, 1, or 2, and occasionally go to even Level 3 or beyond. The necessary level of detail depends on the scope of what you are trying to accomplish.

DFD Level 0

DFD Level 0 is also called a Context Diagram. It's a basic overview of the whole system or process being analyzed or modeled. It's designed to be an at- a-glance view, showing the system as a single high-level process, with its to external entities. It should be easily understood by a wide audience, including stakeholders, business analysts, data analysts and developers.

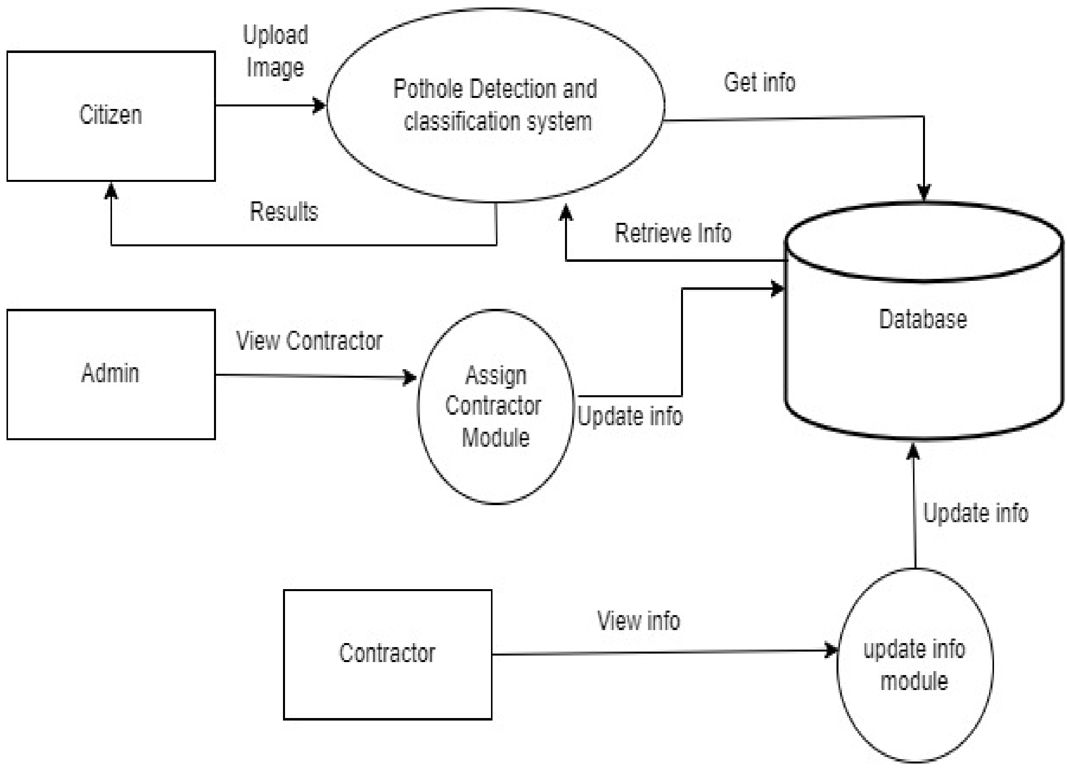


Figure 3.1: Dataflow Diagram Level 0

DFD Level 1

In level 1 DFD, the single process node from the context diagram is broken down into sub-processes. As these processes are added, the diagram will need additional data flows and data stores to link them together.

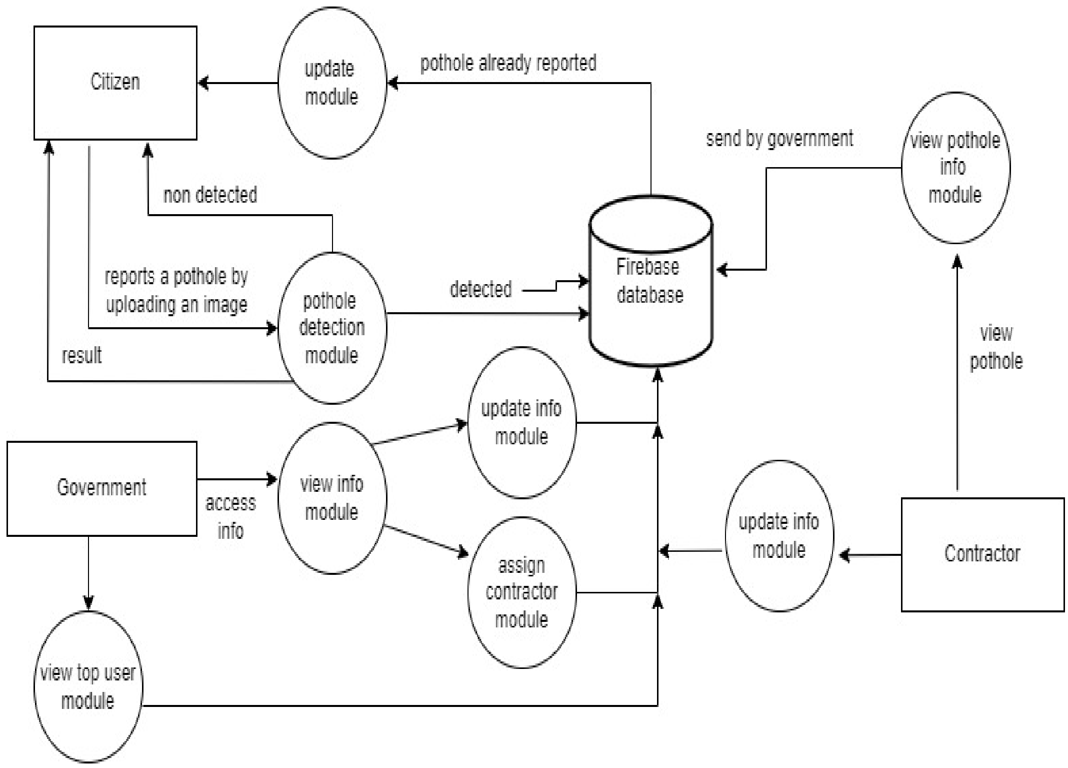


Figure 3.2: Dataflow Diagram Level 1

3.6.2 Class Diagram

The most widely use UML diagram is the class diagram. It is the building block of all object-oriented software systems. We use class diagrams to depict the static structure of a system by showing the system's classes, their methods and attributes. Class diagrams also help us identify relationships between different classes or objects.

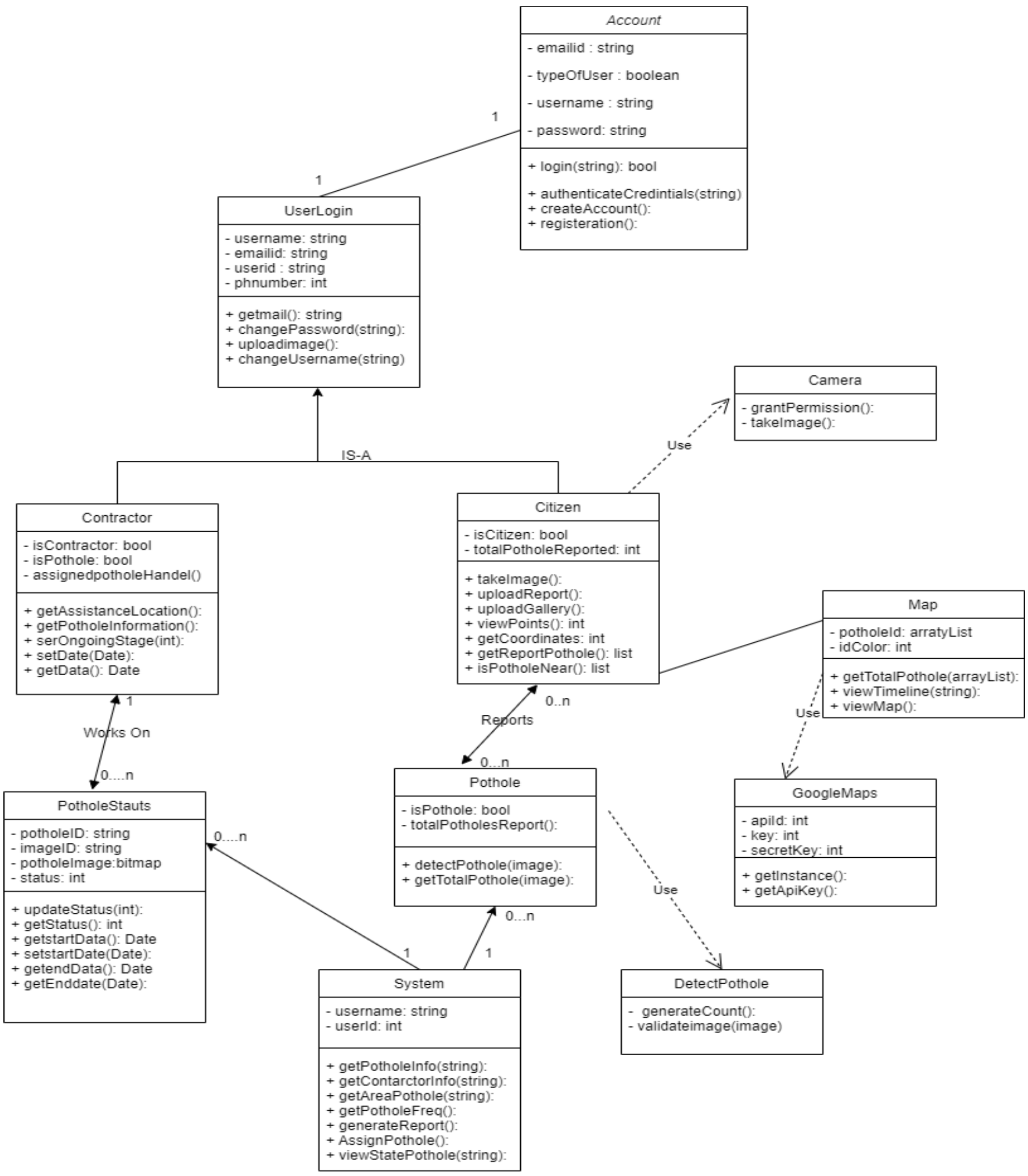


Figure 3.3: Class Diagram

3.6.3 State Machine Diagram

A state diagram is used to represent the condition of the system or part of the system at finite instances of time. It's a behavioral diagram and it represents the behavior using finite state transitions. State diagrams are also referred to as State machines and State-chart Diagrams. These terms are often used interchangeably. So simply, a state diagram is used to model the dynamic behavior of a class in response to time and changing external stimuli.

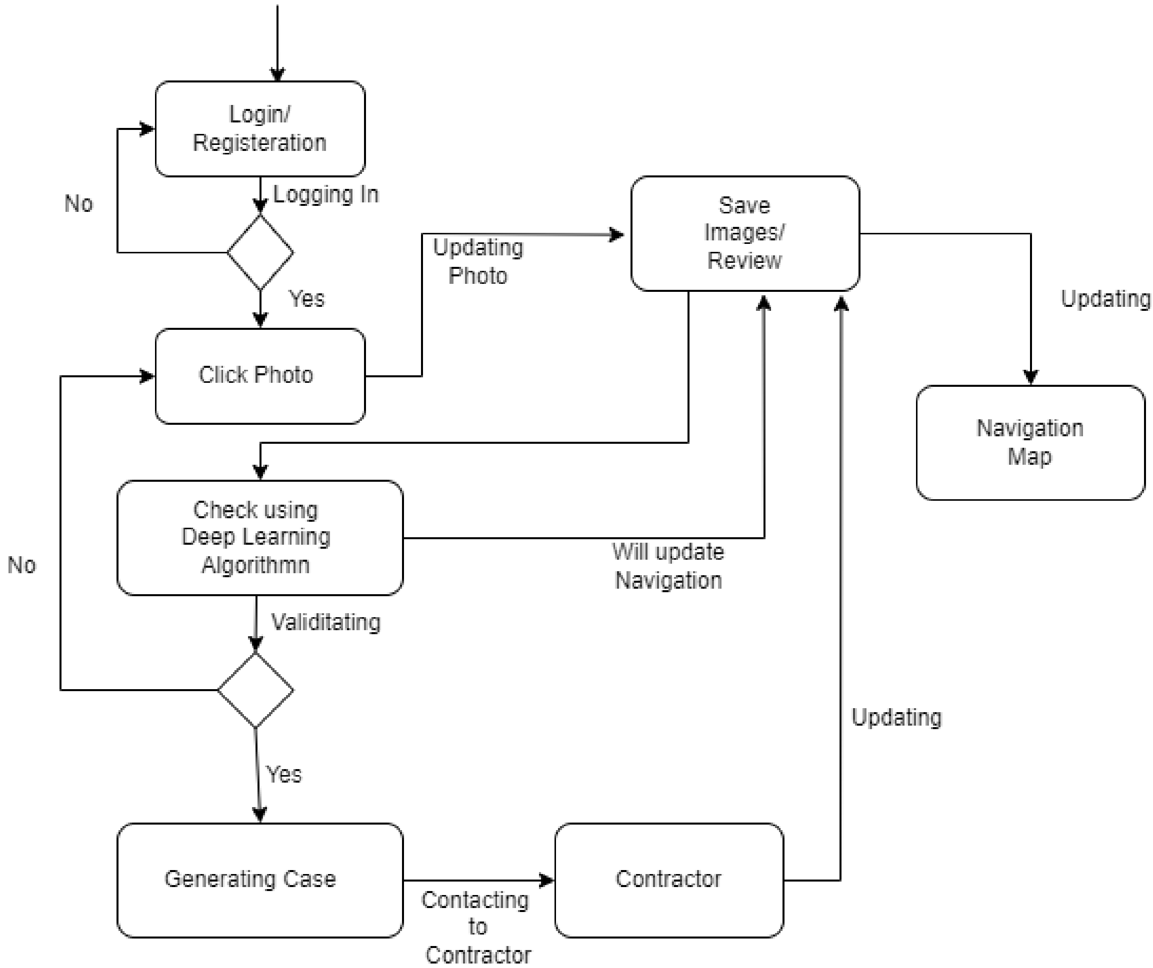


Figure 3.4: State Machine Diagram

3.7 System Implementation Plan

3.7.1 Implementation Plan

Task No.	Task to be Accomplished
T1	Topic Finalization
T2	Requirement specification
T3	Technology Familiarization
T4	System Set up
T5	Concept Review Study
T6	Study of technologies used in the project
T7	Design of user interface
T8	Designing of Constraints Rules
T9	Creation of Connectivity for Services
T10	Creation of database files and rules
T11	Designing the Architectural layout
T12	Creating module using database and rules
T13	Integration of T11 to T7 Tasks
T14	Testing
T15	Documentation Preparation
T16	Maintenance

Table 3.1: Implementation Plan

Chapter 4

System Design

4.1 System Architecture

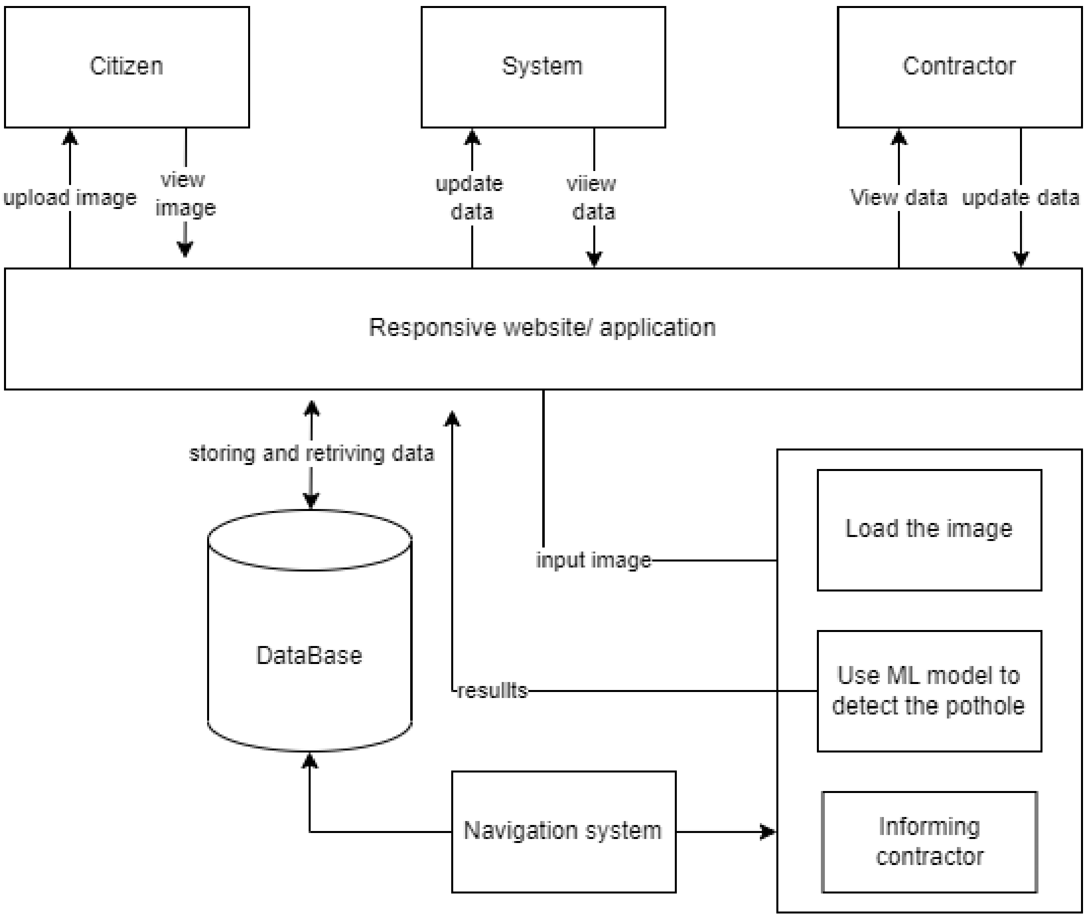


Figure 4.1: System Architecture

The System Architecture contains the following main components.

1. Citizen:

View Image: Citizens can use an Android application or website to access a user interface and view images of road potholes.
Upload Image: They can also capture and upload images of road potholes through the Android application.

2. Android Application:
Store and Retrieve Information: The Android application manages the storage and retrieval of information related to road potholes, including images and other relevant data.It communicates with the server and database for these operations.

3. Contractor:
View Image: Similar to citizens, contractors can view images of road potholes through the Android application or website's user interface. Upload Image: Contractors can also capture and upload images of road potholes using the Android application.

4. Website:
The website serves as a platform for administrators and contractors to maintain pothole data. Administrators can assign contractors to query and update the data. The website provides UI components for viewing and uploading images of road potholes.

5. Database:
All information related to road potholes, such as images and user details, is stored in the database. The Android application and server interact with the database for storing and retrieving information.

6. Server:
The server acts as a mediator between the Android application, website, and database. It receives and processes requests from the Android application and website, facilitating communication with the database. It also enables integration with other services like the map navigation system.

7. Map Navigation System (Integrated with Google Maps): - The map navigation system is integrated with Google Maps to provide navigation and mapping functionalities. It can display the locations of reported potholes on the map, assisting users in visualizing and identifying areas with road issues.

This revised overview captures the main elements of your project's architecture, focusing on the interactions between citizens, contractors, the Android application, website, database, server, and map navigation system.

4.1.1 Working :

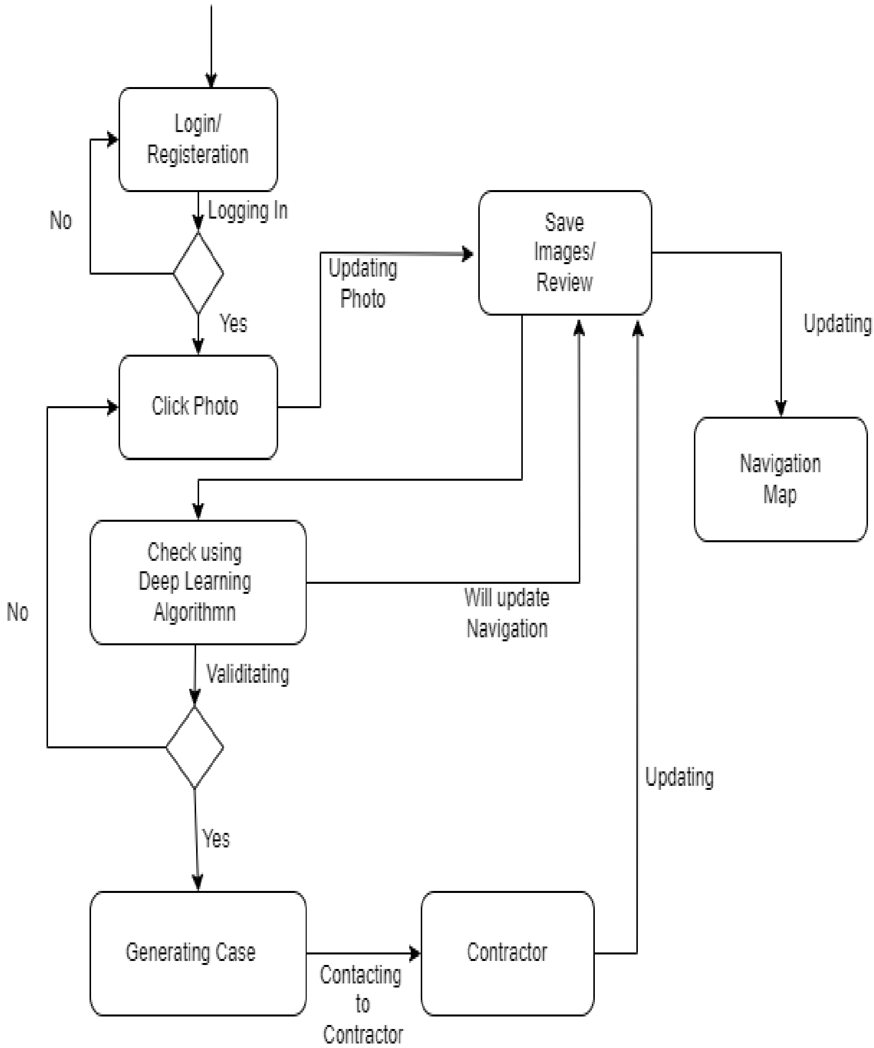


Figure 4.2: Working flow of the Model

Users will begin by registering on the platform, followed by logging in using the credentials sent to them via email. Upon successful login, they will have the option to capture an image of a pothole using their device. The captured image will be processed through a binary classification algorithm to determine if it indeed depicts a pothole. If the classification confirms the presence of a pothole, the image will be uploaded for further action. In the case of a non-pothole image, it will not be uploaded. Once the correct image is uploaded, contractors will be hired to address the pothole and initiate the necessary repairs.

The uploaded pothole will be visible on the google map option provided in the application. The pothole will appear red in color and the if it is not yet considered for the reparing, orange when under reparing and will be removed when the work is done.

4.2 UML Diagrams

4.2.1 Entity Relationship Diagram

The entity-relationship model is widely used for designing databases and can also be used to describe the data of a system and their structure. As for the class diagram, the entity-relationship diagram is easy to use and powerful enough to represent relational structures. It is mainly based on a graphical representation that facilitates its understanding.

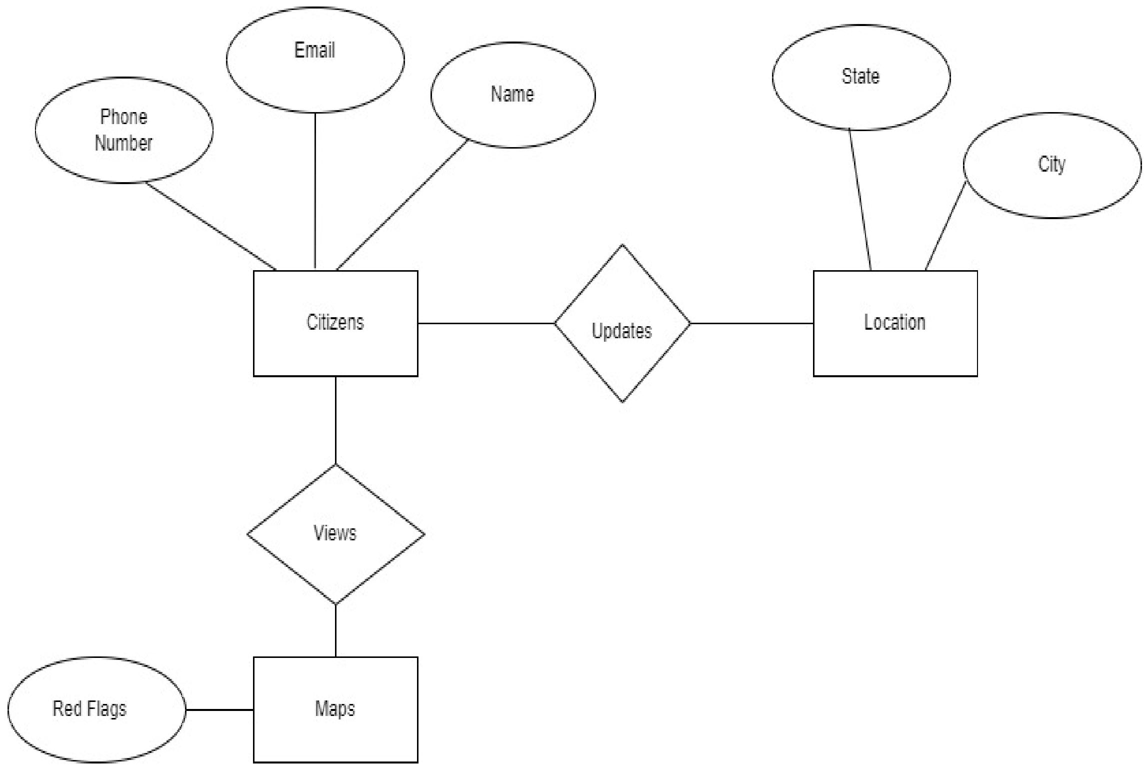


Figure 4.3: Application Interface 2

4.2.2 Activity Diagram

We use Activity Diagrams to illustrate the flow of control in a system. We can also use an activity diagram to refer to the steps involved in the execution of a use case. We model sequential and concurrent activities using activity diagrams. So, we basically depict workflows visually using an activity diagram. An activity diagram focuses on the condition of flow and the sequence in which it happens. We describe or depict what causes a particular event using an activity diagram

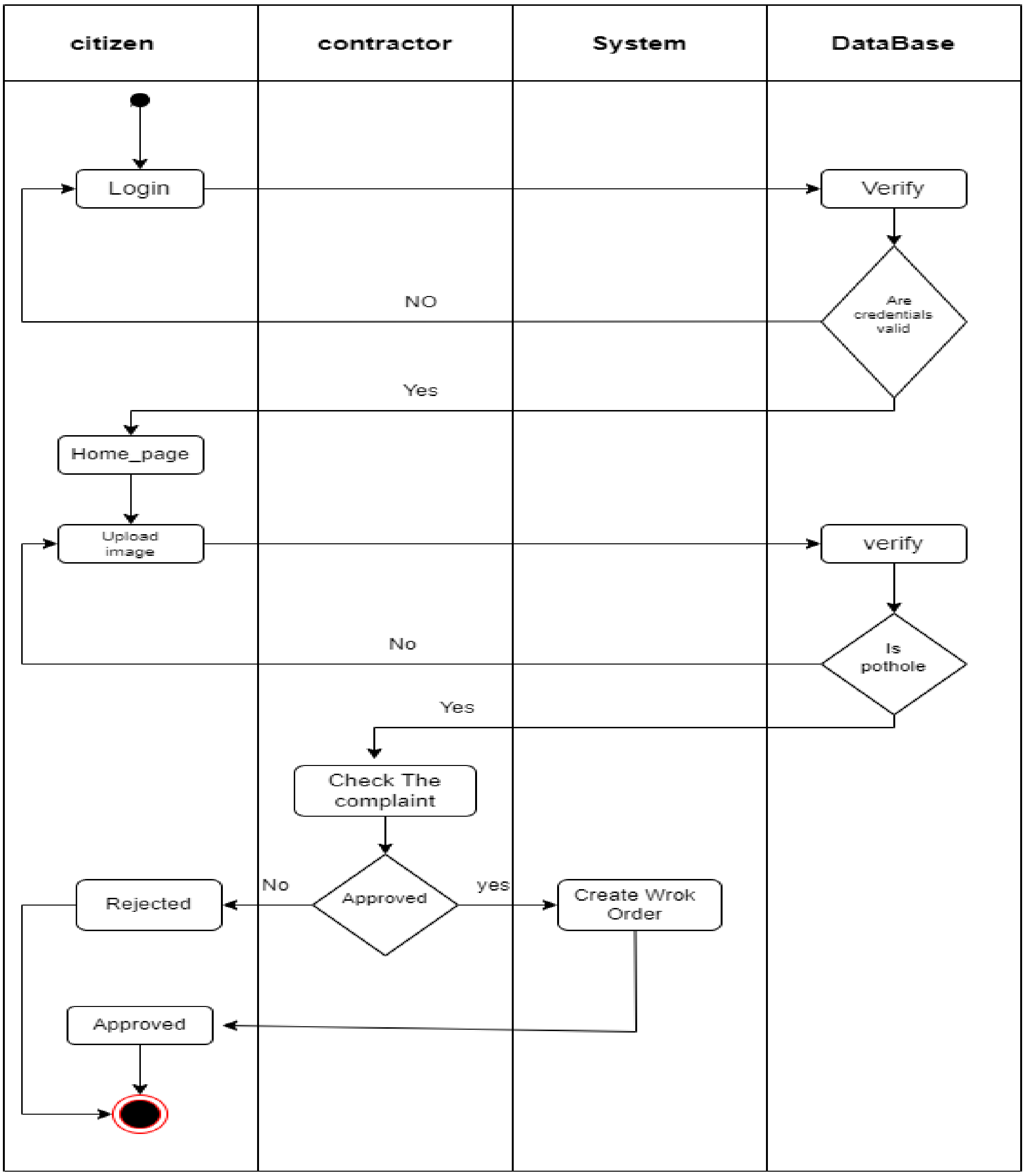


Figure 4.4: Activity Diagram

4.2.3 Use Case Diagram

Use Case Diagrams depict the functionality of a system or a part of a system. They are widely used to illustrate the functional requirements of the system and its interaction with external agents(actors). A use case is basically a diagram representing different scenarios where the system can be used. A use case diagram gives us a high-level view of what the system or a part of the system does without going into implementation details.

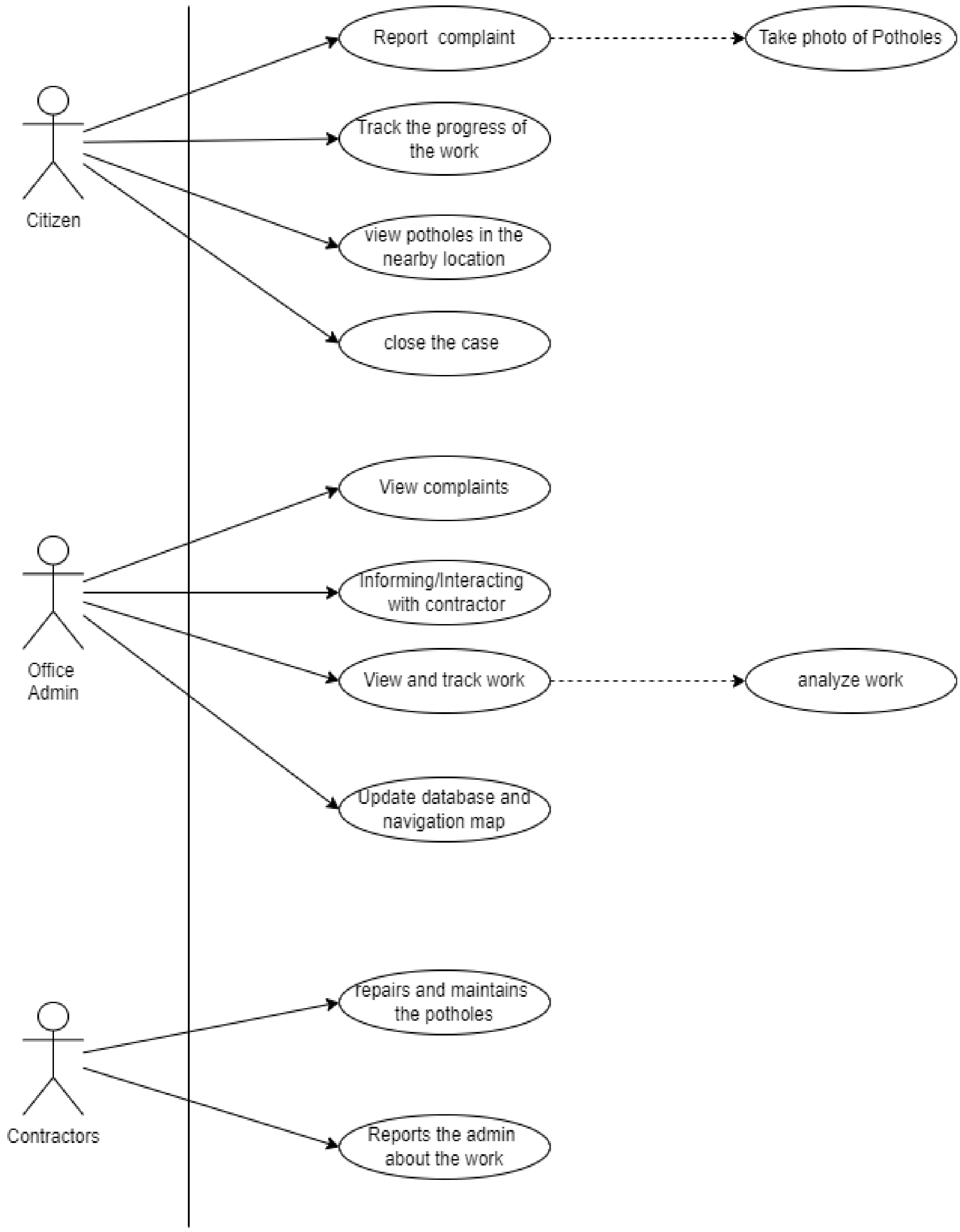


Figure 4.5: Use Case Diagram

4.2.4 Communication Diagram

A Communication Diagram(known as Collaboration Diagram in UML 1.x) is used to show sequenced messages exchanged between objects. A communication diagram focuses primarily on objects and their relationships. We can represent similar information using Sequence diagrams, however, communication diagrams represent objects and links in a free form.

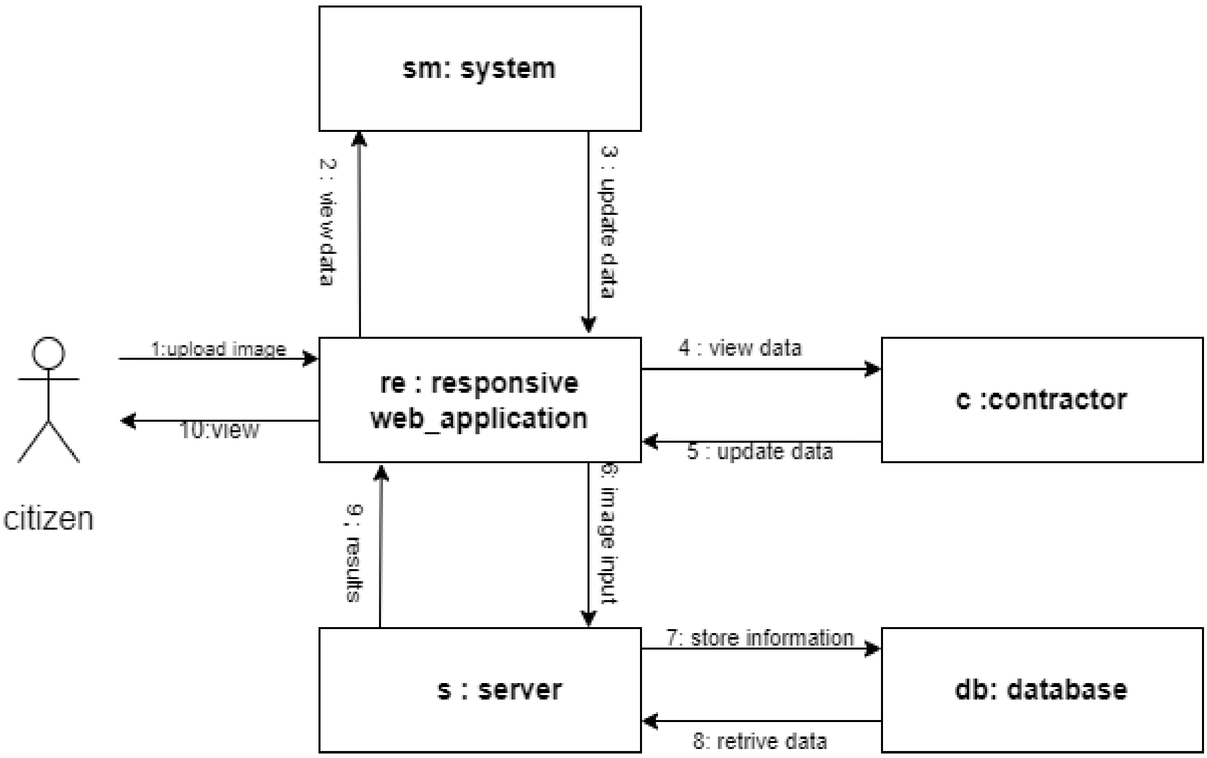


Figure 4.6: Communication Diagram

4.2.5 Sequence Diagram

A sequence diagram depicts the interaction between objects in a sequential order i.e. the order in which these interactions take place. We can also use the terms event diagrams or event scenarios to refer to a sequence diagram. Sequence diagrams describe how and in what order the objects in a system function.

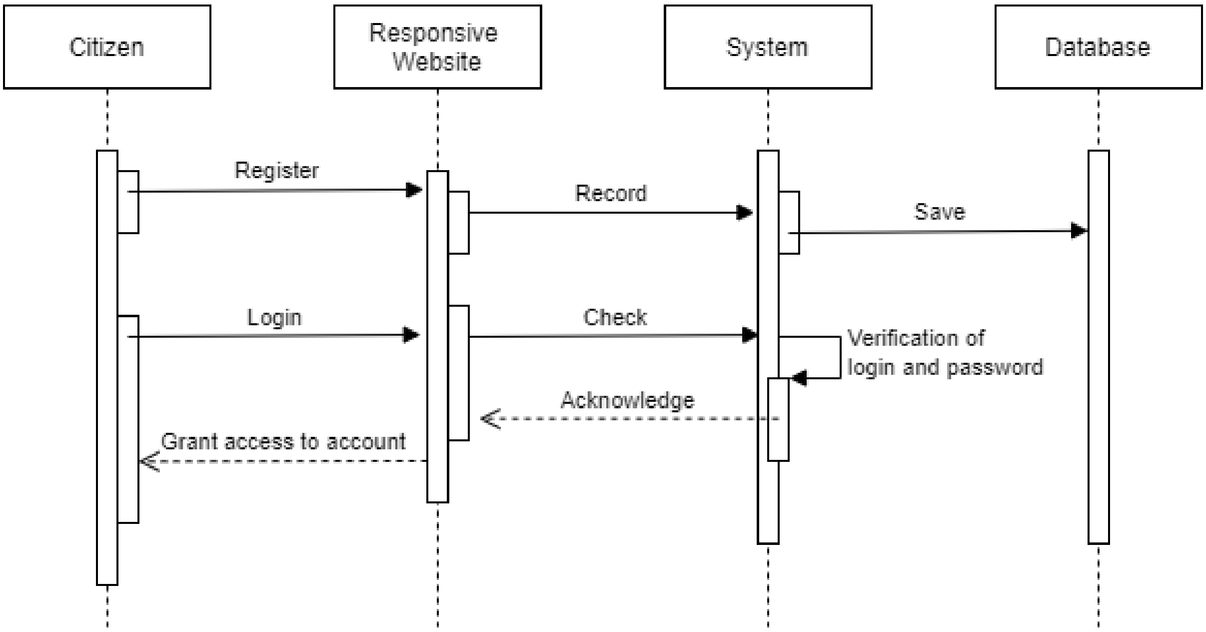


Figure 4.7: Sequence Diagram

4.2.6 Component Diagram

Component diagrams are used to represent how the physical components in a system have been organized. We use them for modeling implementation details. Component Diagrams depict the structural relationship between software system elements and help us understand if functional requirements have been covered by planned development. Component Diagrams become essential to use when we design and build complex systems. Interfaces are used by components of the system to communicate with each other.

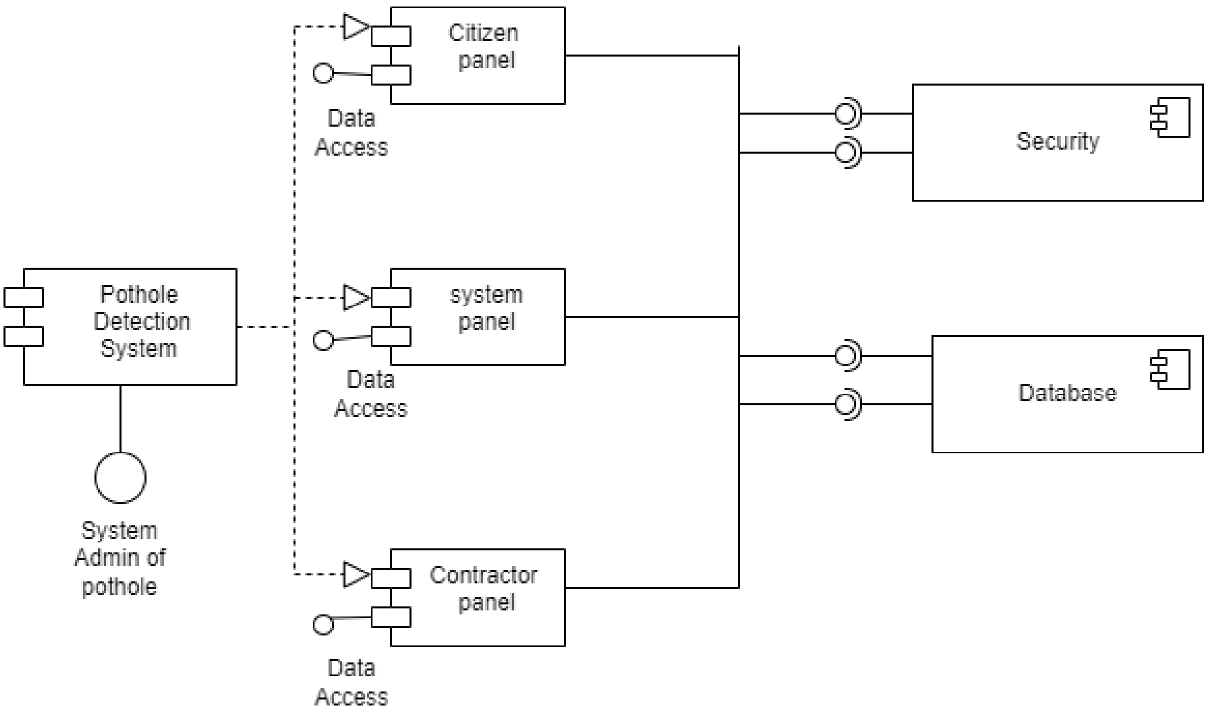


Figure 4.8: Component Diagram

4.2.7 Deployment Diagram

Deployment Diagrams represent system hardware and its software. It tells us what hardware components exist and what software components run on them. We illustrate system architecture as the distribution of software artifacts over distributed targets. An artifact is the information that is generated by system software. They are primarily used when software is being used, distributed, or deployed over multiple machines with different configurations.

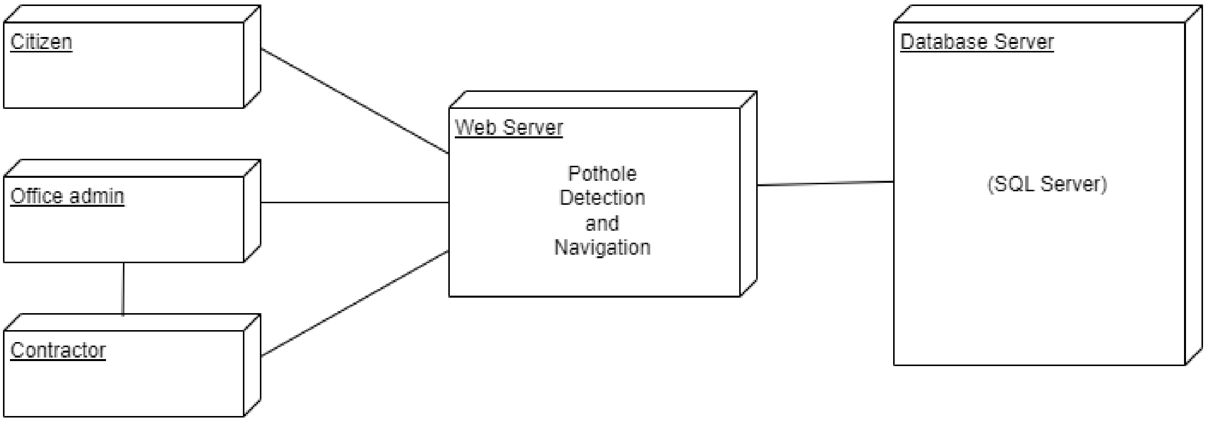


Figure 4.9: Deployment Diagram

Chapter 5

Technical Specifications

5.1 Technology details used in the project

5.1.1 Application

1. **Mobile Application:**

   - Android Studio Editor: When it comes to application development, Android Studio app serves as a fundamental tool in our process. Android Studio provides us with a comprehensive integrated development environment (IDE) specifically designed for creating Android applications. It serves as our primary platform for designing, coding, testing, and debugging our applications.
   - Java programming - The mobile application we have developed utilizes the power and versatility of the Java programming language. Java has been a prominent language choice for Android app development due to its robustness, portability, and extensive community support.

2. **Image Recognition:**

   - Tensorflow: To ensure accurate and efficient image recognition capabilities, we employed TensorFlow as a key component in training our model. TensorFlow has emerged as a leading open-source machine learning framework widely adopted for developing deep learning models. By utilizing TensorFlow's comprehensive suite of libraries and tools, we were able to construct and train a sophisticated image recognition model.
   - Binary Classification: Our model incorporates the logic of binary classification by leveraging two fundamental algorithms: Convolutional Neural Networks

(CNNs) and K-means clustering. This combination allows us to achieve accurate and efficient classification of input data. CNNs are a powerful deep learning architecture widely used for image classification tasks. These networks excel at extracting relevant features from input images by applying convolutional filters and pooling layers. By employing CNNs, our model can learn and identify complex patterns and structures within the images, enabling accurate classification. In addition to CNNs, we integrate the K-means clustering algorithm into our model. K-means clustering is an unsupervised learning algorithm that groups similar data points together based on their feature similarities. In our case, K-means is applied to extract distinct clusters of features from the image data. This process aids in enhancing the discriminative power of the model, enabling it to make more precise distinctions between different classes in the binary classification task.

5.1.2 Website

1. **Frontend:** To create an engaging and user-friendly front-end interface, we employed a combination of HTML, CSS, Bootstrap, and DevExtreme controls. HTML (Hypertext Markup Language) serves as the backbone of our web pages, allowing us to structure and organize the content. CSS (Cascading Style Sheets) enables us to customize the appearance and layout of the elements, ensuring a visually appealing and consistent design. Bootstrap, a popular front-end framework, played a pivotal role in our development process. By leveraging its pre-built components and responsive grid system, we were able to achieve a responsive and mobile-friendly design effortlessly. Bootstrap also provides a plethora of styling options and utility classes, allowing us to create a polished and professional user interface. To enhance the functionality and user experience, we integrated DevExtreme controls into our front-end. DevExtreme offers a rich set of UI controls and widgets, such as data grids, charts, and form elements, that are optimized for cross-platform web development. These controls enable us to present and interact with data in a dynamic and intuitive manner, enhancing the overall usability of our application.

2. **Backend:** For the backend of our application, we utilized a robust technology stack consisting of .NET MVC 4.5 and above, C .NET code, and an architecture based on the Model-View-Controller (MVC) pattern. The .NET MVC framework provided us with a structured and scalable approach to develop our backend logic. We leveraged C.NET, a powerful and versatile programming language, to write the code that powers the functionality of our application. The MVC architecture helped

us organize our code into separate modules, facilitating better code maintainability and separation of concerns. Additionally, we adopted the code-first approach, allowing us to define our data models using code and automatically generate the corresponding database schema. This approach streamlined the development process and ensured seamless integration between our application's data layer and backend logic.

3. **Database:** To support the data storage and management needs of our project, we opted for the robust and reliable Microsoft SQL Server 2019 and above as our database system. Microsoft SQL Server is a popular and widely used relational database management system (RDBMS) known for its performance, scalability, and comprehensive feature set.

Chapter 6

Project Estimation Schedule and Team Structure

6.1 Project Estimate

COCOMO Model A popular method for estimating software costs is called the Constructive Cost Model (COCOMO), which was created by Barry Boehm. It offers an organised method for determining the amount of work, time, and money needed to develop software projects. To determine the effort and cost estimations, COCOMO takes into account a variety of variables and project features. COCOMO is available in three versions: Basic, Intermediate, and Advanced. An overview of each version is given below:

1. **COCOMO Basic:** Based on the project size, expressed in lines of code (LOC), the COCOMO Basic model calculates the software development effort. It takes into account the formula:

   $$Effort = a * (KLOC)^b$$

   where,

   - The total development effort is measured in person-months as effort.
   - KLOC stands for thousand lines of code, which is a measure of the software's projected size.
   - The constants a and b depend on the experience of the development team and the type of project (such as organic, semi-detached, or embedded).

2. **COCOMO Intermediate:** The COCOMO Intermediate model builds on the Basic model by include extra project aspects such product qualities, hardware

limitations, human resource capacity, and development flexibility. It takes into account 15 various cost factors that affect the calculation of overall effort and cost. Four categories have been established for these cost factors: product, platform, staff, and project.

3. **COCOMO Advanced:** The COCOMO Advanced model improves the estimation by taking into account more elements including multi-site development, software reuse, and software stability. It considers elements including the degree of software reuse, the complexity of the dependability requirements for the software, and the effect of geographically dispersed development teams.

There are three modes of development.

Development mode	**Size**	**Innovation**	**Deadline**	**Dev. Environment**
Organic	Small	little	not light	Stable
Semi-detached	Medium	Medium	Medium	Medium
Embedded	Large	Greater	Tight	Complex hardware

Table 6.1: Modes of development

Here are the coefficients related to development mode for the intermediate model.

Development mode	**a**	**b**	**c**	**d**
Organic	11.32	1.05	2.5	0.38

Table 6.2: Coefficients related to development modes for intermediate model

6.1.1 Equations:

$$E = a * (KLOC)^b$$

where,
a = 11.32,b = 1.05, for an organic project.
E = Efforts in person month

6.1.2 Organic project:

Project of moderate size and complexity, where teams with teams with mixed experience levels must meet a mixed rigid and less than rigid requirements(project modway between embedded and oraganic types).

Number of People:

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 months

Cost of Project:

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

$$C = D * Cp$$

where,
C = Cost of project
D = Duration in months
Cp = Cost incurred per person-month

6.1.3 Calculation

Efforts:

$$E = a * (KLOC)^b$$

$$E = 11.32 * (10.940)^{1.05}$$

E = 35.88 person-months
Total of 35.88 person-months are required to complete the project successfully.

Duration of Project:

$$D = c * (E)^d$$

$$D = 2.5 * (E)^{0.38}$$

D = 10 months
The approximate duration of project is 10 months.

Number of people required for the project:

N=38.88/10
N=3.6
N=4 people Therefore 4 people are required to successfully complete the project on schedule.

Cost of Project:

C=10.00 *2500 = 25,000 /- Therefore, the cost of the project is 25,000/- (approx)

6.2 Project Schedule and Team Structure

All our project tasks are divided as shown in the table

Task No.	Task Title
T1	Topic Finalization
T2	Requirement specification
T3	Technology Familiarization
T4	System Set up
T5	Concept Review Study
T6	Study of technologies used in the project
T7	Design of user interface
T8	Designing of Constraints Rules
T9	Creation of Connectivity for Services
T10	Creation of database files and rules
T11	Designing the Architectural layout
T12	Creating module using database and rules
T13	Integration of T11 to T7 Tasks
T14	Testing
T15	Documentation Preparation
T16	Maintenance

Table 6.3: Lists Of Tasks

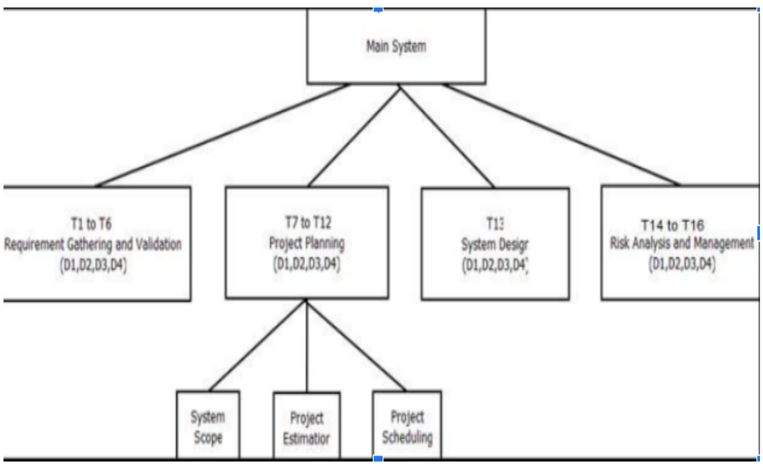


Figure 6.1: Project Work Breakdown Structure

Task No.	No. of Days	Developer
T1	7	D1,D2,D3,D4
T2	4	D1,D2
T3	4	D1,D2
T4	2	D1,D2
T5	4	D1,D2,D3,D4
T6	7	D1,D2
T7	5	D1,D2
T8	8	D1,D2
T9	5	D1,D2
T10	5	D1,D2,D3,D4
T11	7	D1,D2
T12	5	D1,D2,D3,D4
T13	10	D1,D2,D3
T14	7	D1,D2
T15	6	D1,D2,D3,D4
T16	4	D1,D2,D3,D4

Table 6.4: Task Organization

Each task is assigned to one or more team members as shown in fig:

Developer ID	Developer Name
D1	Miss. Anushree Suryavanshie
D2	Miss. Pranjal Patil
D3	Mr. Ritesh Pawar
D4	Mr. Shankar Khadar

Table 6.5: Lists of Developers

Chapter 7

Software Implementation

7.1 Introduction

The project utilizes a combination of technologies for both the mobile app and website components. The mobile app is developed using the Android Studio editor and Java programming language, providing a platform for creating a user-friendly and interactive mobile application. Additionally, image recognition capabilities are integrated into the app using TensorFlow, allowing for advanced image analysis and processing. Techniques such as binary classification, K-Means clustering, and convolutional neural networks (CNN) are employed to enhance the image recognition functionality.

On the website side, the front end is built using HTML, CSS, and Bootstrap, ensuring a visually appealing and responsive user interface. DevXtreme controls are leveraged to enhance the user experience and provide seamless interactions. The backend of the website is developed using .NET MVC 4.5 and above, utilizing C.Net code. The Model-View-Controller (MVC) architecture is employed for efficient code organization and separation of concerns. The code-first approach is used for database management, allowing for flexible and efficient development by defining the data model through code. Microsoft SQL Server 2019 and above serve as the database management system, providing a reliable and robust platform for storing and managing data. The database facilitates data storage, retrieval, and manipulation, ensuring seamless integration with the application and website components.

Overall, the project showcases a comprehensive technological stack, incorporating mobile app development, image recognition, web development, and database management. By utilizing these technologies, the project aims to deliver a user-friendly and efficient solution for its intended purpose.

7.2 Databases

To support the data storage and management needs of our project, we opted for the robust and reliable Microsoft SQL Server 2019 and above as our database system. Microsoft SQL Server is a popular and widely used relational database management system (RDBMS) known for its performance, scalability, and comprehensive feature set.

7.3 Important module and algorithms

1. **Registration:**
   The method by which a user registers in the system, submits the necessary data, and declares his acceptance of following by the applicable user agreement. This module helps the user sign up for the subsequent step on the portal.

2. **Login:**
   Users receive an email with their username and password after registering with the site. This must be used by the user to log into the application. The correctness of the login and password is also verified. If either the username or the password is incorrect, the system provides another opportunity to input the proper username and password.

3. **Capturing of the image:**
   Once users access their account on our application, they are presented with a "Create case" option on the dashboard. By clicking on this option, users can provide information about the location and pothole they have encountered. Furthermore, they have the option to capture a photograph of the pothole. If the photograph captured by the user contains a pothole, it is added to the database for further analysis and processing. On the other hand, if the photo does not contain a pothole, it is discarded as it does not contribute to the dataset. This process allows users to actively participate in documenting and reporting potholes through our application. By selectively adding photos with potholes to the database, we aim to enhance the accuracy and effectiveness of our system in detecting and addressing road damage.

4. **Classification:**
   When a user uploads a photograph using our application, it undergoes verification through a binary classification model designed to determine the presence or absence of a pothole. This classification model is trained using TensorFlow, a popular

machine-learning framework. To train the binary classification model, we employ a combination of K-Means clustering and Convolutional Neural Networks (CNN). K-Means clustering is used as a preprocessing technique to extract relevant features from the input images. It helps identify patterns and group similar image regions together, aiding in the subsequent classification process.

Following the K-Means clustering step, the extracted image features are fed into a CNN architecture. CNNs are powerful deep-learning models specifically designed to process visual data efficiently. They learn to recognize patterns and structures within images, enabling accurate classification. Through the training process, the binary classification model learns to distinguish between images that contain potholes and those that do not. By leveraging the combined capabilities of K-Means and CNN, the model gains the ability to effectively analyze and classify the uploaded photographs.

The utilization of TensorFlow provides a robust and flexible platform for training the binary classification model. TensorFlow offers extensive tools and libraries specifically designed for machine learning tasks, ensuring efficient training and optimization of the model.

By employing K-Means and CNN within the TensorFlow framework, we aim to enhance the accuracy and reliability of the binary classification model in detecting the presence of potholes in the uploaded photographs. This allows us to provide users with reliable feedback on whether a given image contains a pothole or not, contributing to the effectiveness of our application in addressing road damage.

5. **Upload in the database:**

   Once the binary classification process determines that a photo contains a pothole, it is added to the database specifically used for training the model. This database serves as a valuable resource for refining and improving the performance of the model over time.

   By incorporating these pothole-containing photos into the training dataset, we ensure that the model learns from real-world examples and becomes more proficient in accurately identifying and classifying potholes. The addition of new data helps the model capture a wider range of pothole variations, such as different shapes, sizes, and road conditions.

   Training the model with the expanded database allows it to adapt and refine its classification abilities based on the collective experiences and contributions of users. The model analyzes the added photographs, identifies patterns and features, and adjusts its internal parameters to improve its accuracy in detecting and categorizing potholes.

6. **Contractors:**

   After new pothole cases are reported on the website, the administrator is empowered to assign these cases to the appropriate contractors for further action. This ensures efficient distribution of workload and facilitates timely resolution of the reported potholes.

7. **Maps:**

   The integrated map navigation system in our application leverages the powerful features of Google Maps to provide comprehensive navigation and mapping functionality. Users can easily find routes from their selected source to a desired destination using this system.

   An important feature of the map navigation system is its ability to display the locations of reported potholes directly on the map interface. To visually represent these pothole locations, red flags are utilized as markers. These red flags serve as visual indicators, helping users easily identify and locate areas where potholes have been reported.

   By incorporating the red flag markers on the map, users gain valuable insights into the presence and distribution of potholes in their vicinity or along their chosen route. This information empowers users to make informed decisions and adjust their travel plans accordingly, potentially avoiding roads with known pothole issues.

   The integration of Google Maps as the underlying mapping technology provides a reliable and widely recognized platform for accurate navigation and mapping data. By seamlessly combining this with the visual representation of pothole locations through red flag markers, we enhance the user experience and assist in creating a safer and more efficient travel environment.

   Ultimately, the inclusion of red flags on the map interface helps users stay informed about the presence of reported potholes and aids them in planning their routes accordingly, contributing to smoother and more comfortable journeys.

7.4 Business logic and architecture

The business logic for the project revolves around providing a comprehensive solution to address the issue of potholes on roadways. The primary objective is to create a platform that enables users to navigate through roads efficiently while being aware of the presence of potholes in real time. The key components of the business logic include:

1. Live Pothole Detection: The system incorporates image recognition technology to detect and classify potholes in real time. By leveraging machine learning algorithms and computer vision techniques, the system can analyze user-uploaded images or images from other sources to identify and locate potholes accurately.

2. User Reporting and Collaboration: The project encourages active user participation in identifying and reporting potholes. Users can upload images of potholes they encounter, providing valuable data for the system. This collaborative approach allows for a broader coverage of pothole information and facilitates a collective effort in maintaining road quality.

3. Real-time Map Integration: The navigation system integrates with mapping technologies, such as Google Maps, to provide users with real-time information on pothole locations. By overlaying pothole markers on the map interface, users can visualize areas with reported potholes, enabling them to plan their routes more effectively and avoid roads with significant damage.

4. Data Collection and Analysis: The system collects and stores data related to potholes, including images, location information, and user reports. This data serves as a valuable resource for analysis, enabling the identification of patterns, trends, and areas prone to pothole formation. Insights derived from data analysis can be utilized for proactive road maintenance and infrastructure planning.

5. Contractor Management: The system incorporates features to manage contractors responsible for pothole repair and maintenance. Administrators can assign tasks to contractors based on reported potholes, ensuring timely and efficient resolution of road damage. Contractors can access the system to view assigned tasks, update their status, and communicate with administrators.

6. User Feedback and Verification: The system provides mechanisms for users to provide feedback on the accuracy of pothole detections. This feedback loop helps improve the performance of the image recognition algorithms and ensures the reliability of the system's results. Users can verify or dispute pothole detections, contributing to the continuous improvement of the system's accuracy.

7. Data Privacy and Security: The project places a strong emphasis on data privacy and security. User information and images are securely stored and handled to

protect user privacy. Access controls and encryption measures are implemented to safeguard sensitive data.

Overall, the business logic of the project aims to enhance road navigation by empowering users with real-time information about pothole locations. By leveraging user collaboration, data analysis, and map integration, the system enables efficient pothole management and contributes to improving road infrastructure and safety.

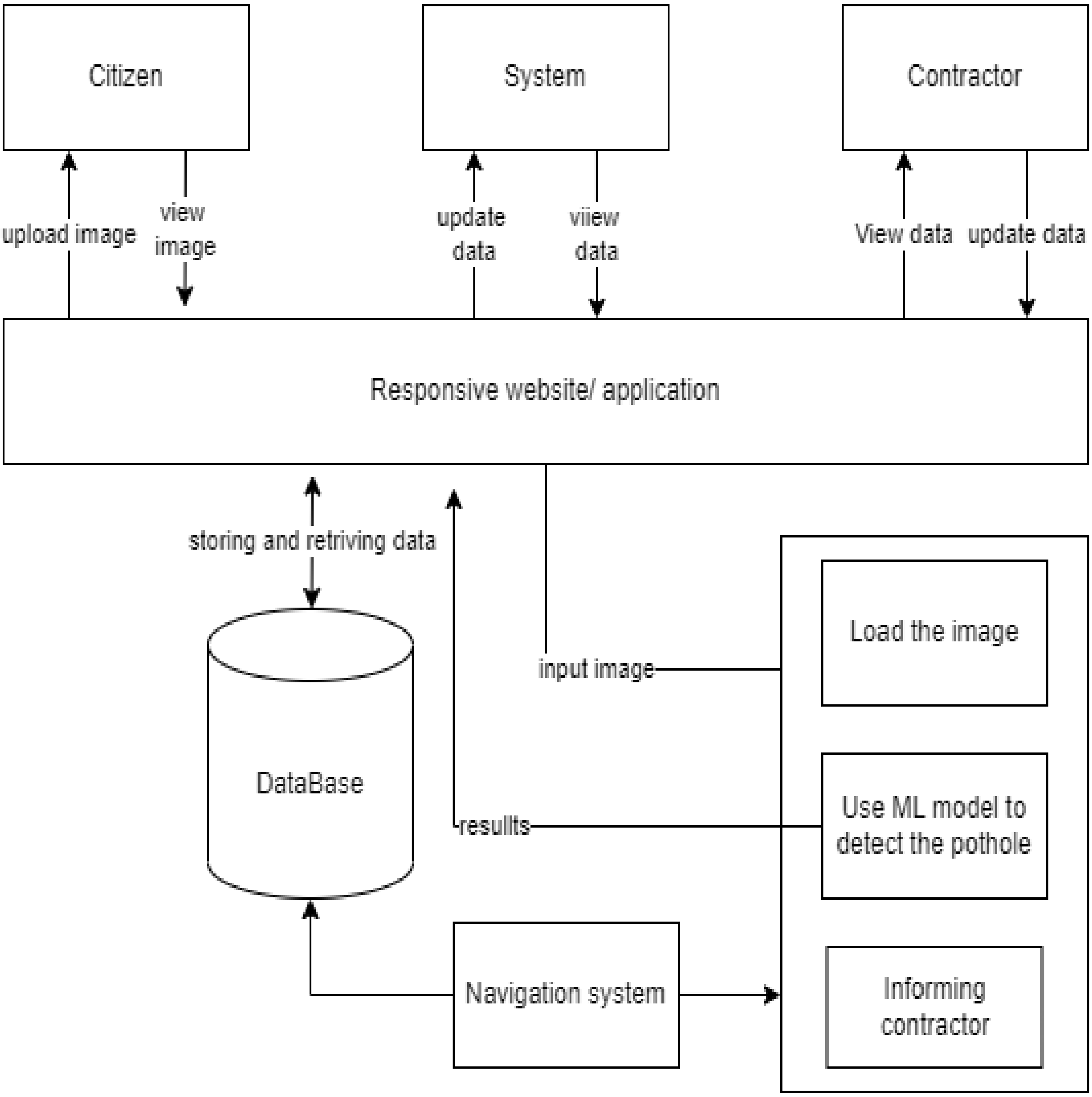


Figure 7.1: System Architecture

Chapter 8

Software Testing

8.1 Introduction

Software testing is a crucial process in the software development life cycle that involves evaluating a software application or system to ensure it meets specified requirements, functions as intended, and performs reliably. It is performed to identify defects, errors, or discrepancies between the expected and actual results of the software.The primary goal of software testing is to uncover and report issues that may impact the functionality, performance, usability, and security of the software. By identifying and addressing these issues early in the development process, testing helps improve the overall quality and reliability of the software.

8.1.1 Test cases for Registration and Login

No	Behaviour Description	Property
1	Unique Test case ID	TC001
2	Test Case Name	Successful Registration
3	Prerequisites	Constant Internet Connection
4	Test Case Description	Enter Data in the field
5	Input	First Name, Last Name, Email, Phone number, User Name, Password
6	Expected Result	User should get registered to the system
7	Actual Result	Registered Successfully!
8	Pass/Fail	Pass

Table 8.1: Test Case for successful registration

No	Behaviour Description	Property
1	Unique Test case ID	TC002
2	Test Case Name	Blank required fields
3	Prerequisites	Constant Internet Connection
4	Test Case Description	Do not enter any data in the field
5	Input	Blank Field
6	Expected Result	Error message ”Please fill out all required fields”
7	Actual Result	Required
8	Pass/Fail	Pass

Table 8.2: Test case for blank field

No	Behaviour Description	Property
1	Unique Test case ID	TC003
2	Test Case Name	Email field is not in the correct format
3	Prerequisites	Constant Internet Connection
4	Test Case Description	Enter invalid email
5	Input	Invalid Email
6	Expected Result	Error message ”Invalid Email ID”
7	Actual Result	Invalid Email Address
8	Pass/Fail	Pass

Table 8.3: Test case for invalid email address

No	Behaviour Description	Property
1	Unique Test case ID	TC004
2	Test Case Name	Input the phone no with more than 10 numbers
3	Prerequisites	Constant Internet Connection
4	Test Case Description	Enter invalid phone no.
5	Input	phone no with more than 10 numbers
6	Expected Result	Error message” Phone no exceeds the limit”
7	Actual Result	Phone no exceeds the limit
8	Pass/Fail	Pass

Table 8.4: Test case for incorrect phone number

No	Behaviour Description	Property
1	Unique Test case ID	TC005
2	Test Case Name	Password verification
3	Prerequisites	Constant Internet Connection
4	Test Case Description	Enter correct password
5	Input	Password
6	Expected Result	User can access the account
7	Actual Result	User is able to access the account
8	Pass/Fail	Pass

Table 8.5: Test cases for password

8.1.2 Test cases for Application Dashboard

No	Behaviour Description	Property
1	Unique Test case ID	TC006
2	Test Case Name	Checking the dashboard
3	Prerequisites	Constant Internet Connection
4	Test Case Description	Enter Valid username and password
5	Input	Valid username and password
6	Expected Result	All options on dashboard should be visible
7	Actual Result	All options on dashboard are visible
8	Pass/Fail	Pass

Table 8.6: Test cases for dashboard

No	Behaviour Description	Property
1	Unique Test case ID	TC007
2	Test Case Name	Accessing "Create Case" Option
3	Prerequisites	Constant Internet Connection
4	Test Case Description	Tap on "Create Case" Option
5	Input	Valid username and password
6	Expected Result	"Create Case" Option should be accessible
7	Actual Result	"Create Case" Option is accessible
8	Pass/Fail	Pass

Table 8.7: Test cases for dashboard

No	Behaviour Description	Property
1	Unique Test case ID	TC008
2	Test Case Name	Accessing ”All cases” Option
3	Prerequisites	Constant Internet Connection
4	Test Case Description	Tap on ”All Cases” Option
5	Input	Valid username and password
6	Expected Result	”All Cases” Option should be accessible
7	Actual Result	”All Cases” Option is accessible
8	Pass/Fail	Pass

Table 8.8: Test cases for dashboard

No	Behaviour Description	Property
1	Unique Test case ID	TC009
2	Test Case Name	Accessing ”View Map” Option
3	Prerequisites	Constant Internet Connection
4	Test Case Description	Tap on ”View Map” Option
5	Input	Valid username and password
6	Expected Result	”View Map” Option should be accessible
7	Actual Result	”View Map” Option is accessible
8	Pass/Fail	Pass

Table 8.9: Test cases for dashboard

No	Behaviour Description	Property
1	Unique Test case ID	TC010
2	Test Case Name	Create case using ”Camera” option
3	Prerequisites	Constant Internet Connection
4	Test Case Description	Click the photo of pothole using camera
5	Input	Photograph
6	Expected Result	Camera should be able to capture photograph
7	Actual Result	Camera is able to capture photograph
8	Pass/Fail	Pass

Table 8.10: Test cases for dashboard

No	Behaviour Description	Property
1	Unique Test case ID	TC011
2	Test Case Name	Create case using ”Gallery” option
3	Prerequisites	Constant Internet Connection
4	Test Case Description	Photos can be uploaded from Gallery
5	Input	Photograph
6	Expected Result	Photos from the gallery should be visible
7	Actual Result	Photos from the gallery are visible
8	Pass/Fail	Pass

Table 8.11: Test cases for dashboard

No	Behaviour Description	Property
1	Unique Test case ID	TC012
2	Test Case Name	Filling information to create new case
3	Prerequisites	Constant Internet Connection
4	Test Case Description	To enter all the required fields in form
5	Input	All the required fields in form (Location, Photo)
6	Expected Result	It should accepts all the valid inputs
7	Actual Result	It accepts all the valid inputs
8	Pass/Fail	Pass

Table 8.12: Test cases for creating new case

8.1.3 Test Cases For Implementation of binary classification

No	Behaviour Description	Property
1	Unique Test case ID	TC013
2	Test Case Name	Checking if Photograph contains pothole
3	Prerequisites	Constant Internet Connection
4	Test Case Description	Upload the photo through Camera or Gallery
5	Input	Photograph containing pothole
6	Expected Result	Message showing ”Contains pothole”
7	Actual Result	Message shows ”Contains pothole”
8	Pass/Fail	Pass

Table 8.13: Test cases for checking pothole in photograph

No	Behaviour Description	Property
1	Unique Test case ID	TC014
2	Test Case Name	Checking if Photograph contains pothole
3	Prerequisites	Constant Internet Connection
4	Test Case Description	Upload the photo through Camera or Gallery
5	Input	Photograph without pothole
6	Expected Result	Message should show that it does not contain pothole
7	Actual Result	Message shows that it does not contain pothole
8	Pass/Fail	Pass

Table 8.14: Test cases for checking pothole in photograph

8.1.4 Test Cases For Live Navigation System

No	Behaviour Description	Property
1	Unique Test case ID	TC015
2	Test Case Name	Accepting "source" and "destination"
3	Prerequisites	Constant Internet Connection
4	Test Case Description	Provide "source" and "destination"
5	Input	"source" and "destination"
6	Expected Result	It should accept "source" and "destination"
7	Actual Result	It accepts "source" and "destination"
8	Pass/Fail	Pass

Table 8.15: Test cases for Live Navigation

No	Behaviour Description	Property
1	Unique Test case ID	TC016
2	Test Case Name	Red Flags showing potholes on Maps
3	Prerequisites	Constant Internet Connection
4	Test Case Description	Provide "source" and "destination"
5	Input	irrelevant document
6	Expected Result	Red flags should be visible on maps
7	Actual Result	Red flags are visible on maps
8	Pass/Fail	Pass

Table 8.16: Test cases for Live Navigation

8.2 Snapshot of testcases

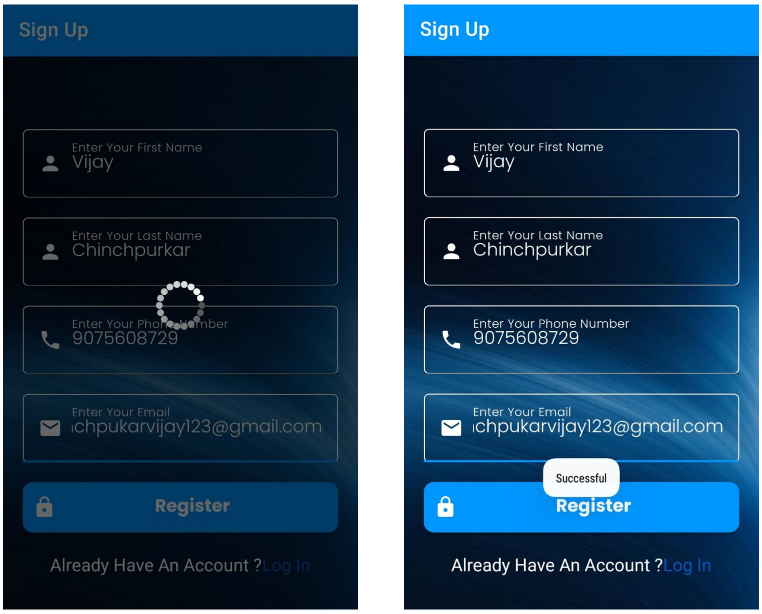


Figure 8.1: Successful Registeration

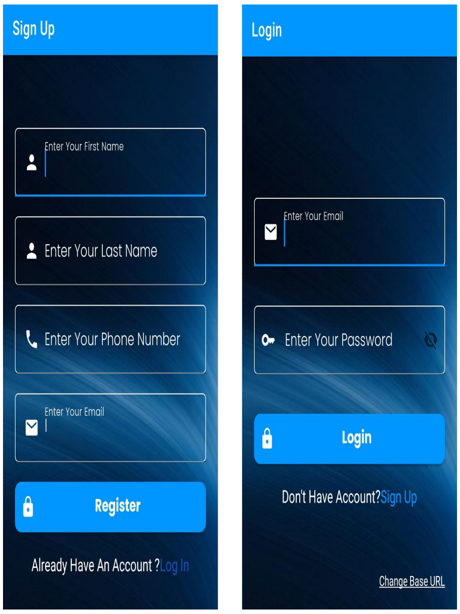


Figure 8.2: Required blank field

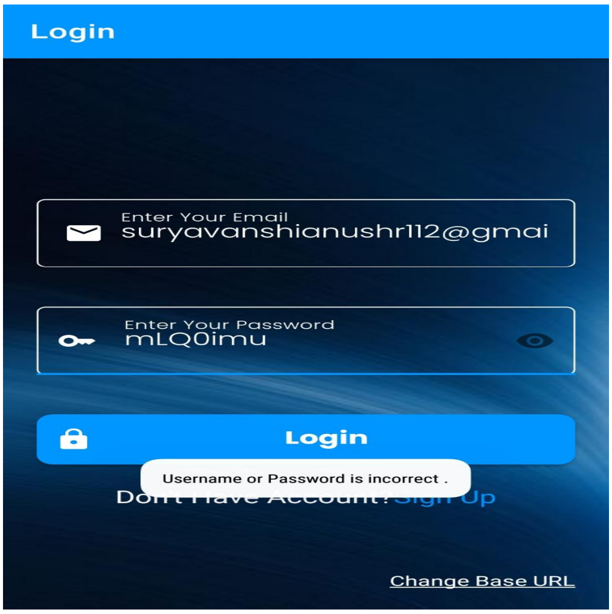


Figure 8.3: Email not correct

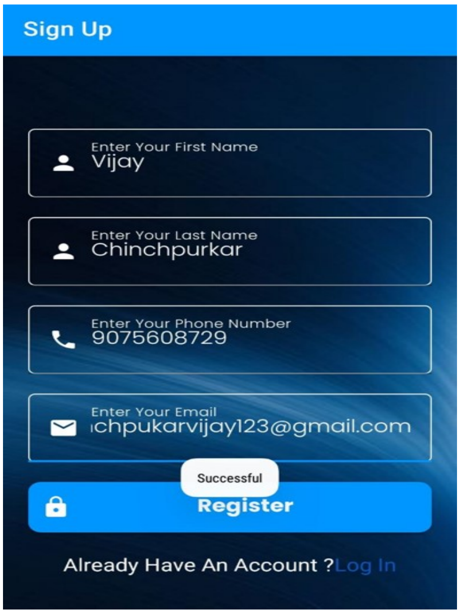


Figure 8.4: Input - phone number

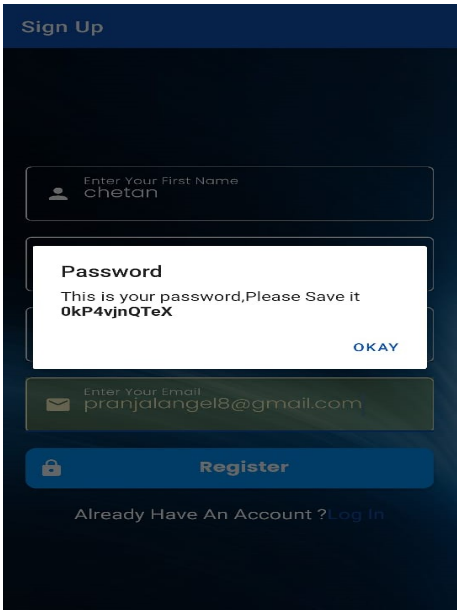


Figure 8.5: Checking validity of password

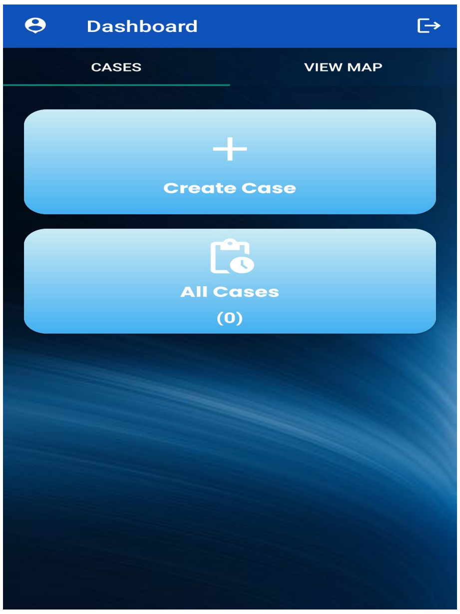


Figure 8.6: Accessing Dashboard

Figure 8.7: Accessing ”create case” option

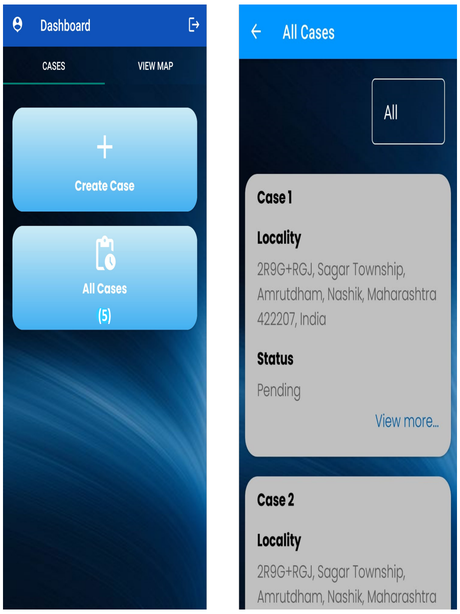


Figure 8.8: Accessing ”All cases” option

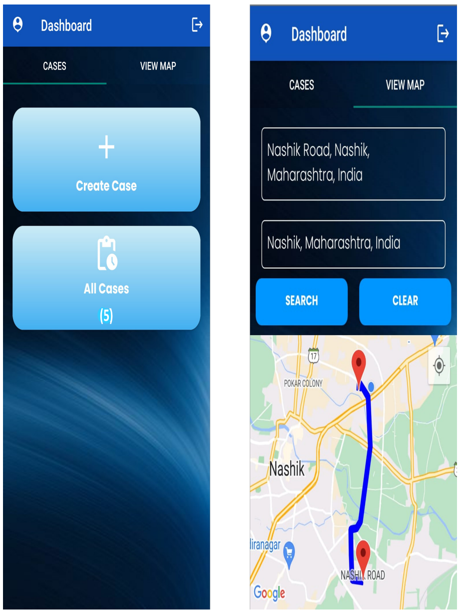


Figure 8.9: Accessing "View Map" option

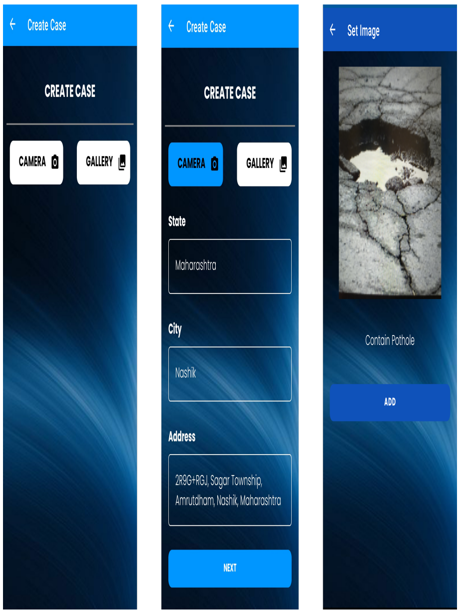


Figure 8.10: create case using "camera" option

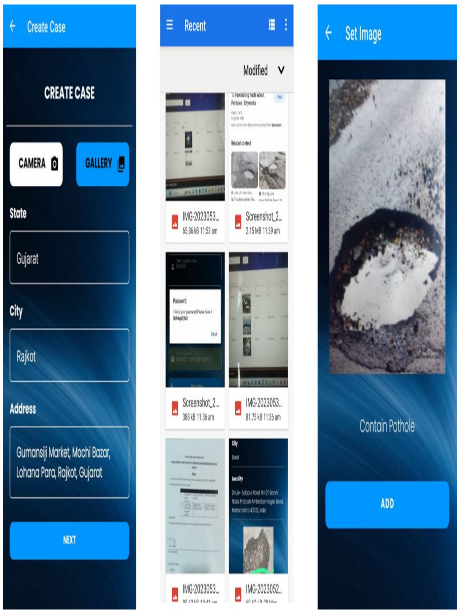


Figure 8.11: Create Case using gallery

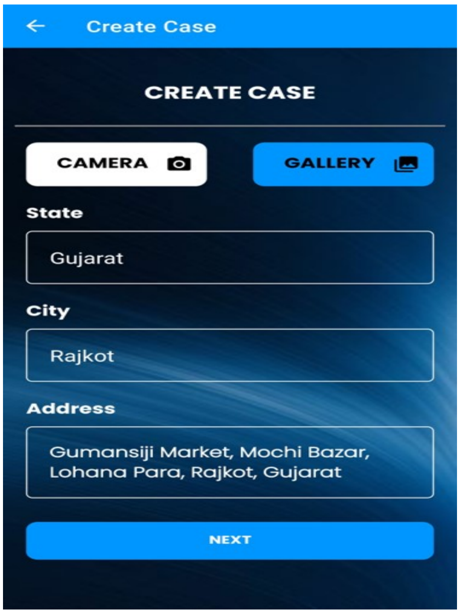


Figure 8.12: Filling Details

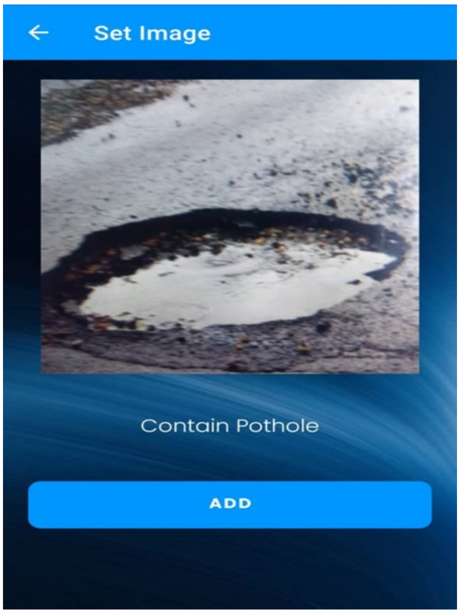


Figure 8.13: Detecting Pothole

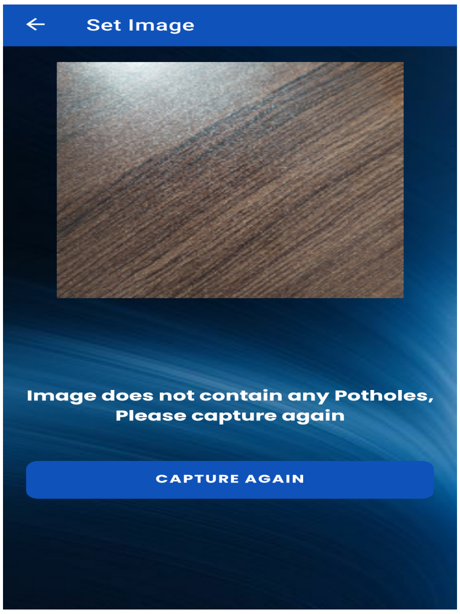


Figure 8.14: Checking for Pothole

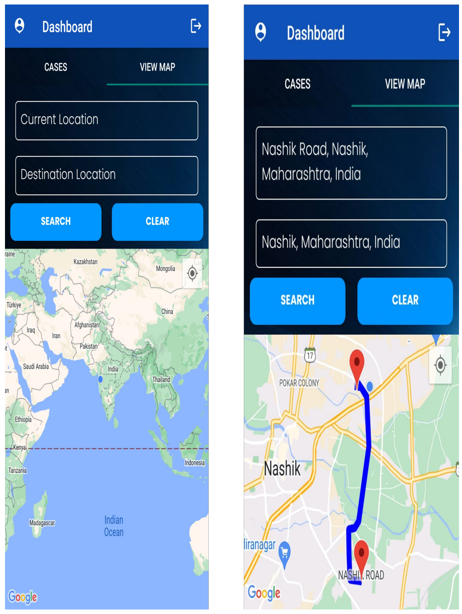


Figure 8.15: Excepting Source and Destination

Chapter 9

Result

9.1 Snapshots of the results

9.1.1 Application

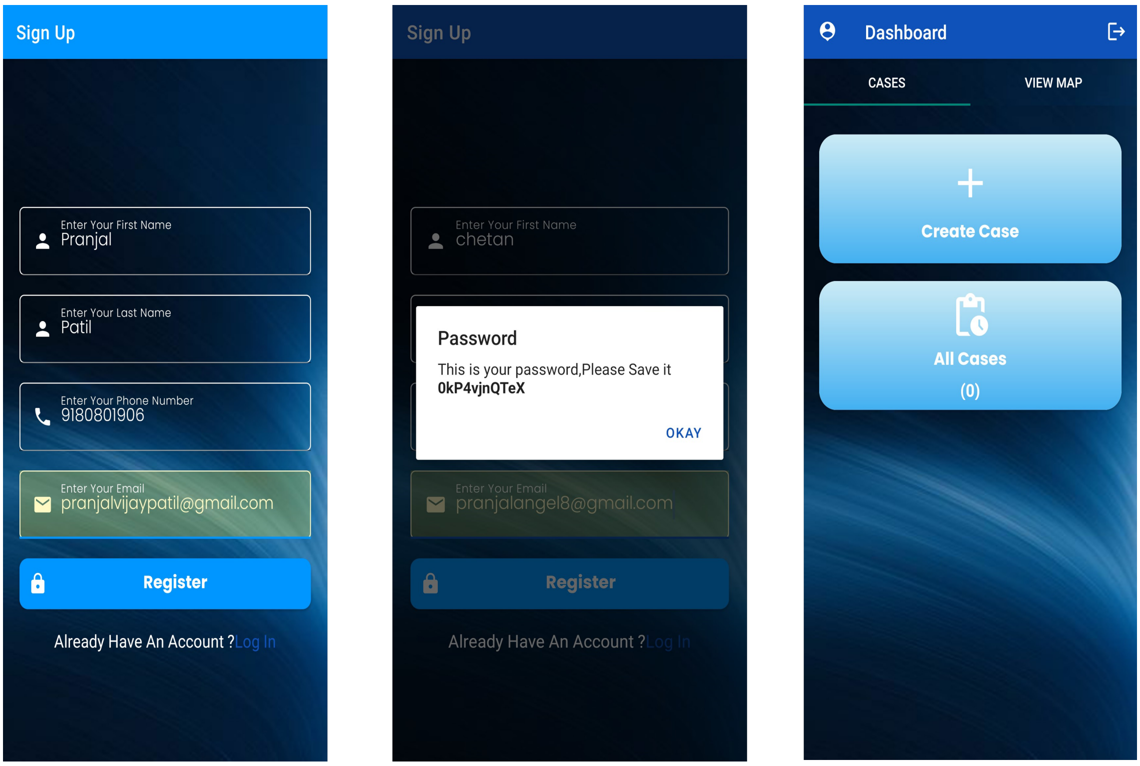


Figure 9.1: Application Interface 1

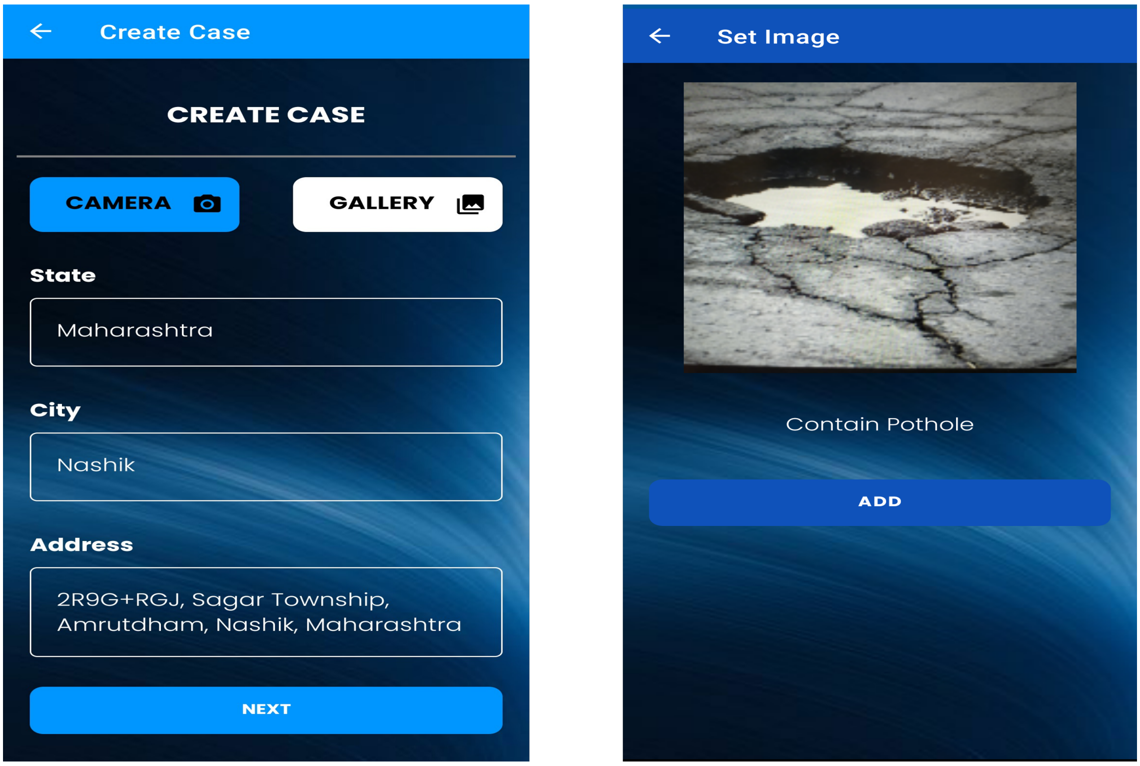


Figure 9.2: Application Interface 2

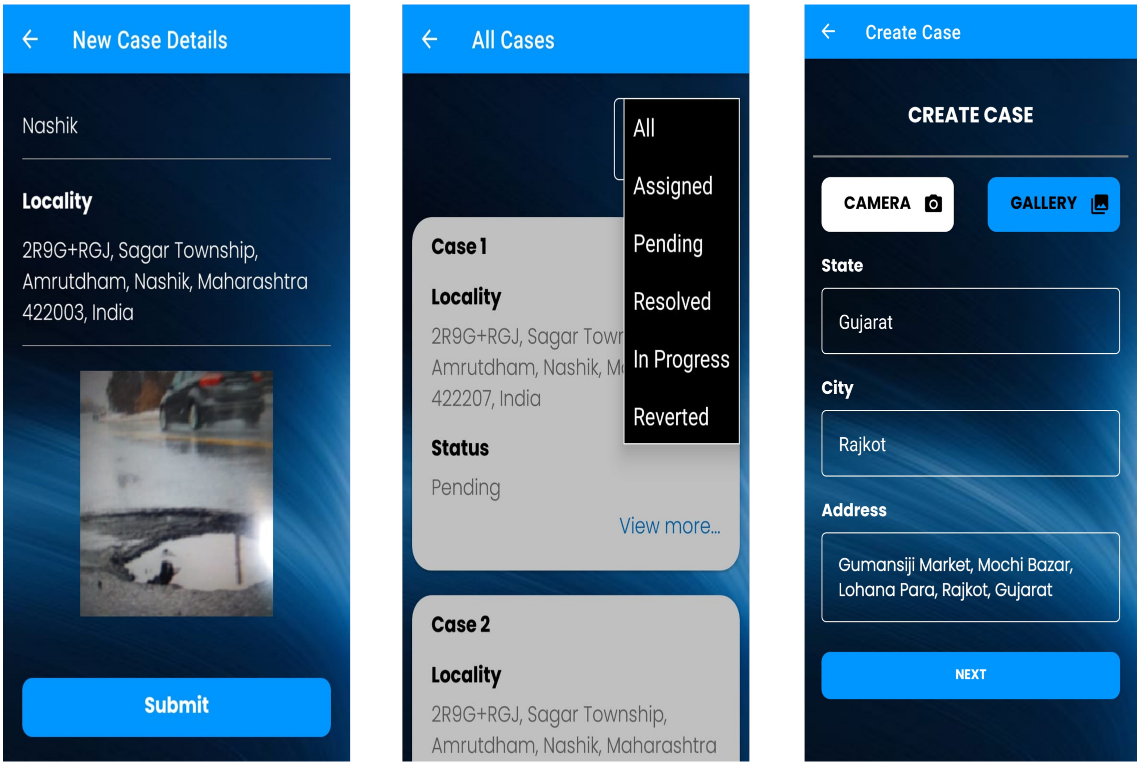


Figure 9.3: Application Interface 3

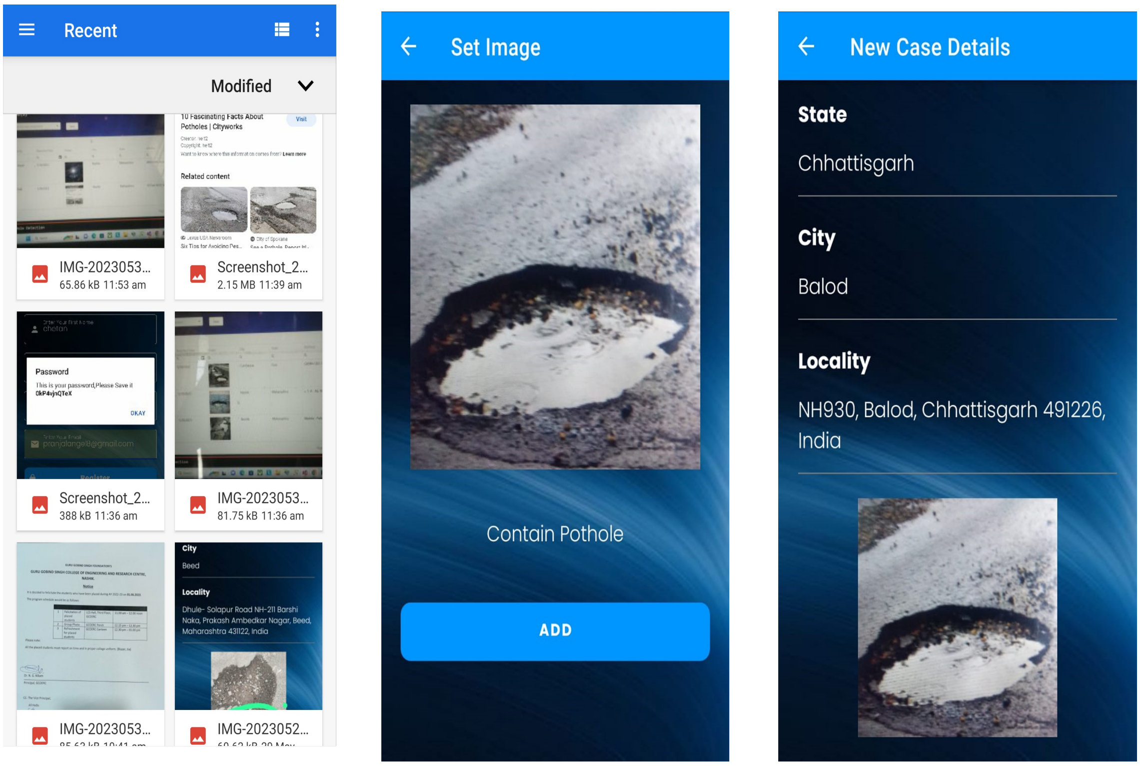


Figure 9.4: Application Interface 4

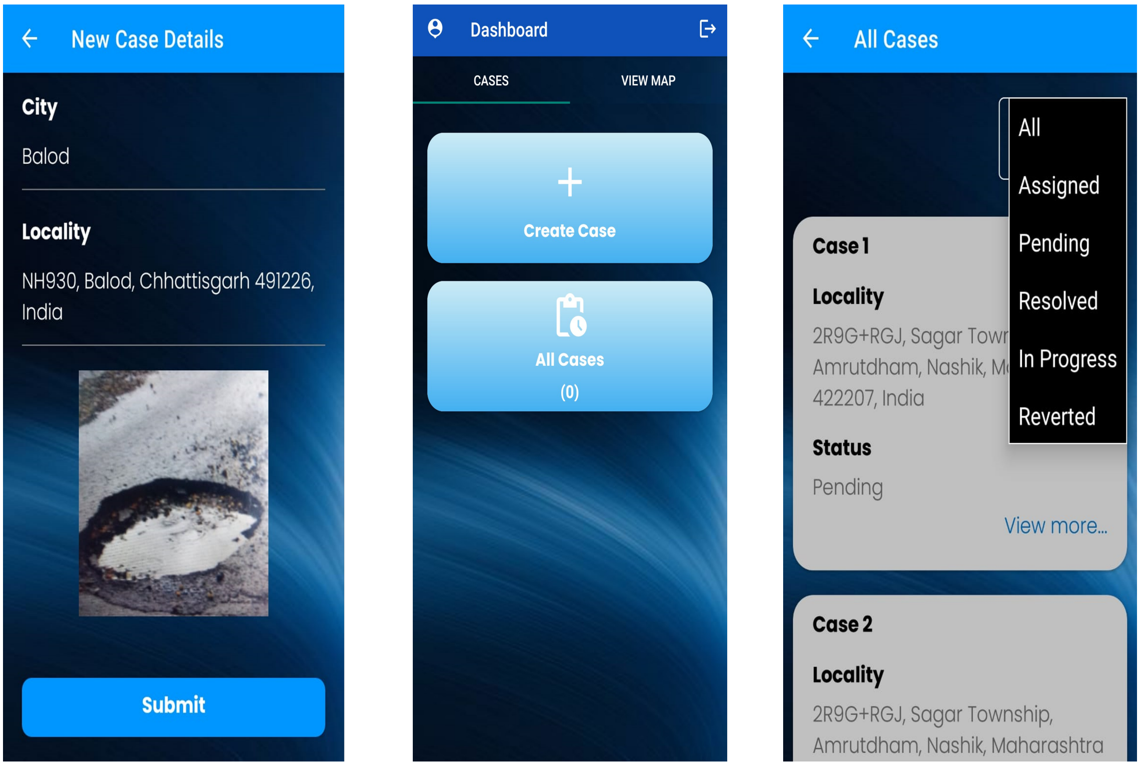


Figure 9.5: Application Interface 1

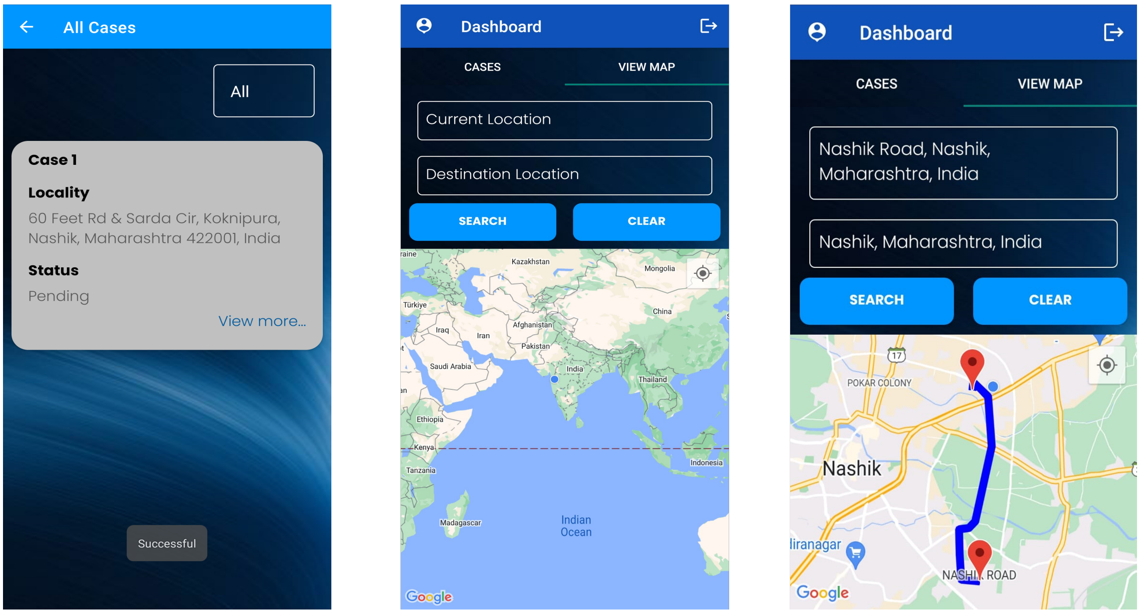


Figure 9.6: Application Interface 1

9.1.2 Website

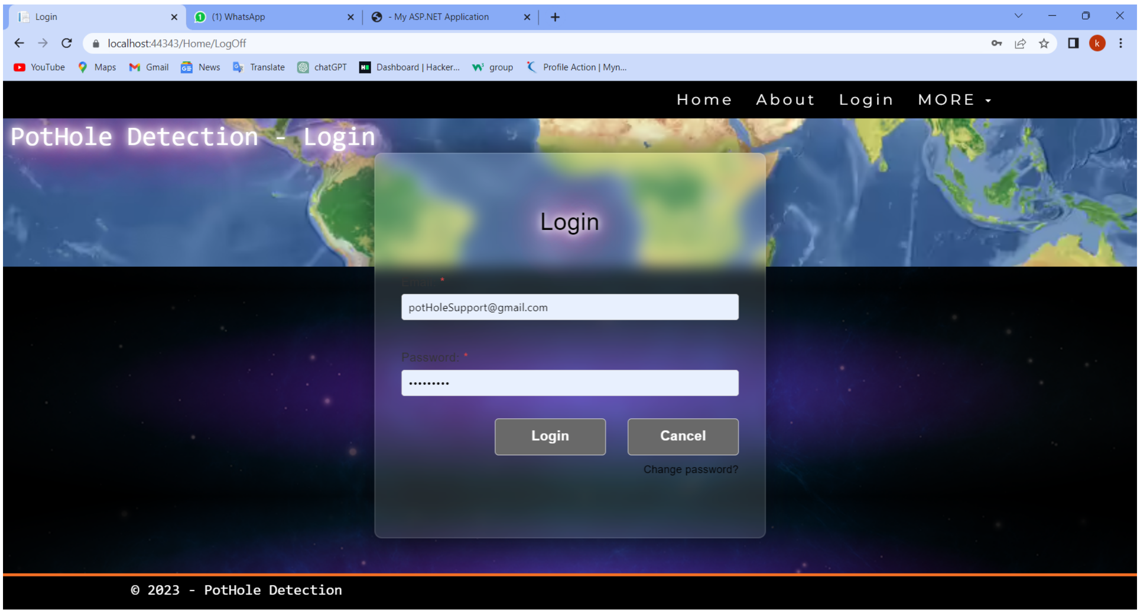


Figure 9.7: Website Interface 1

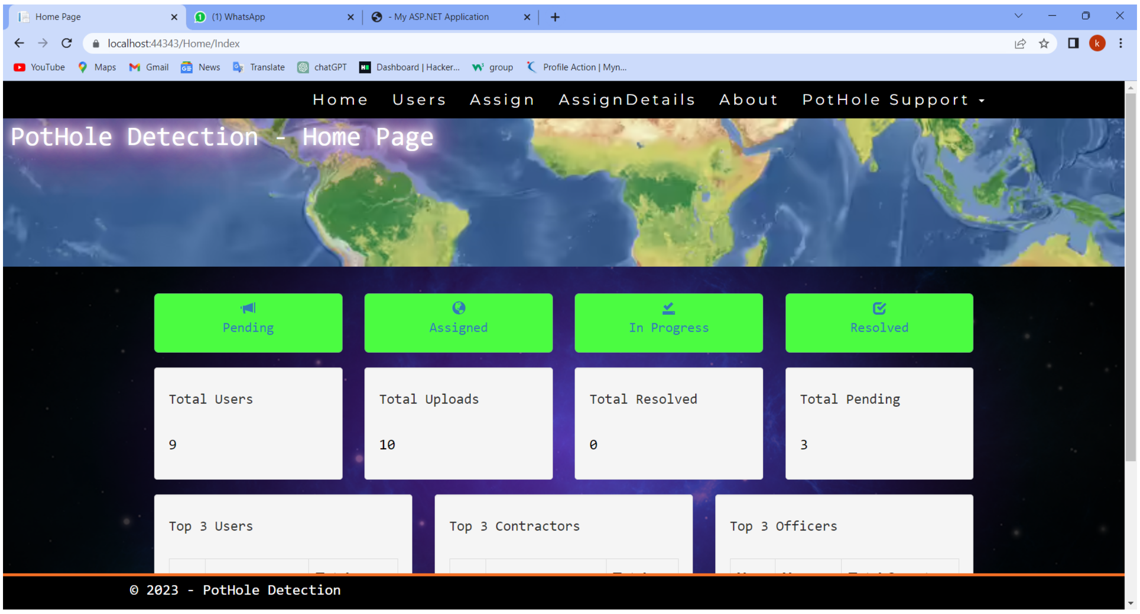


Figure 9.8: Website Interface 2

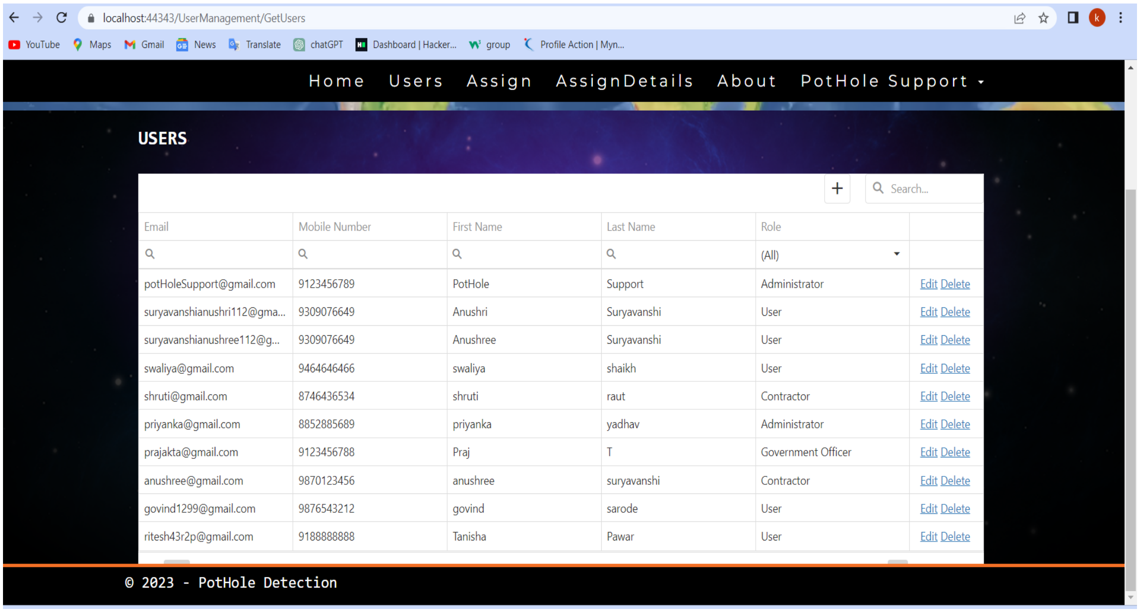


Figure 9.9: Website Interface 3

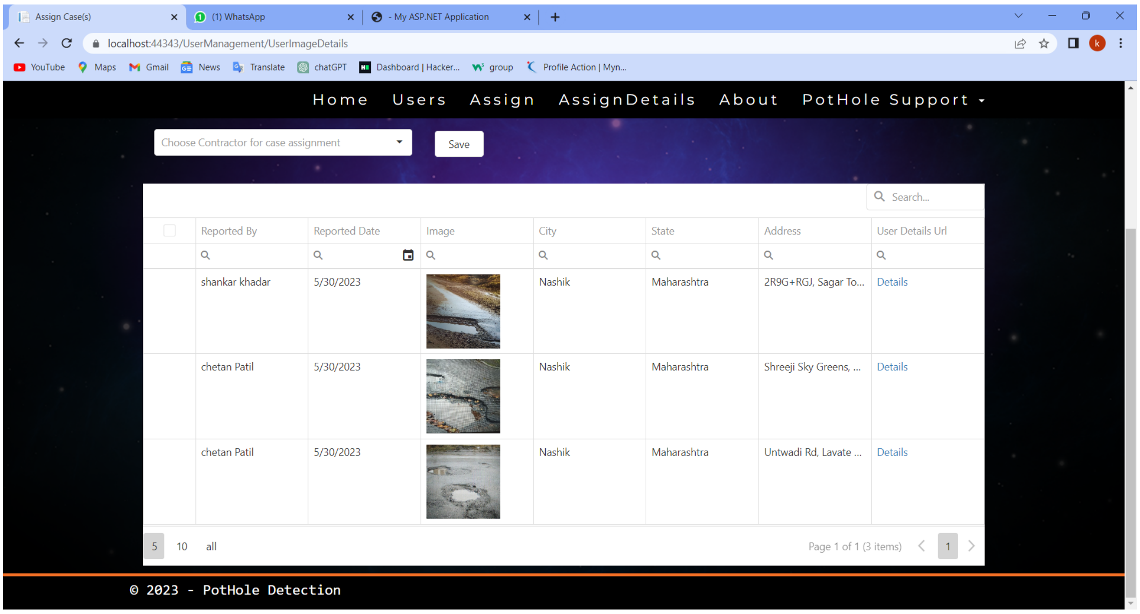


Figure 9.10: Website Interface 4

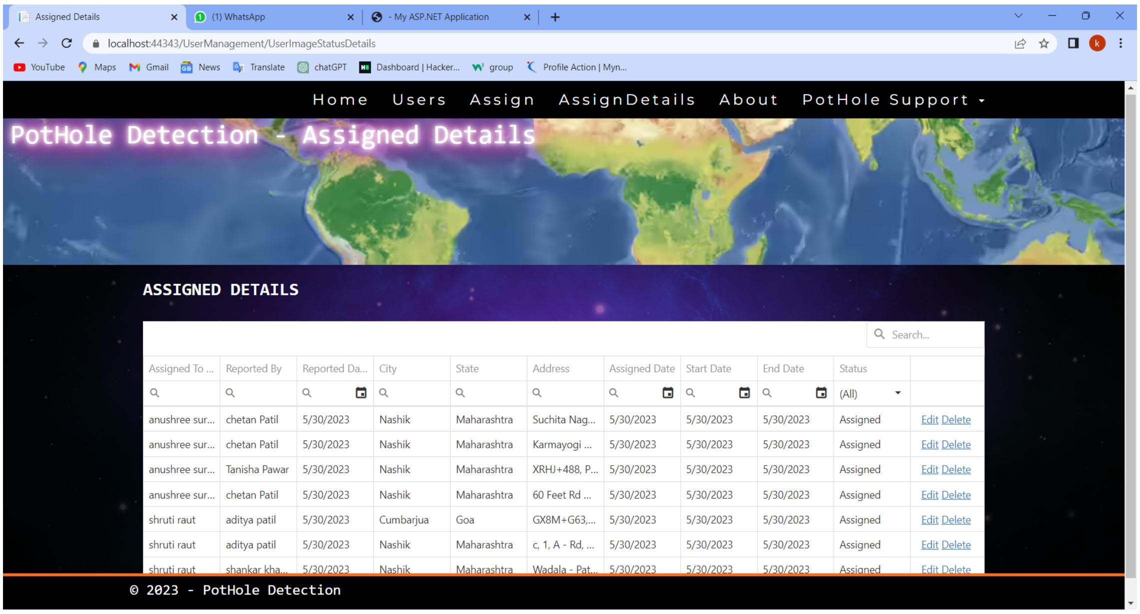


Figure 9.11: Website Interface 5

Figure 9.12: Website Interface 6

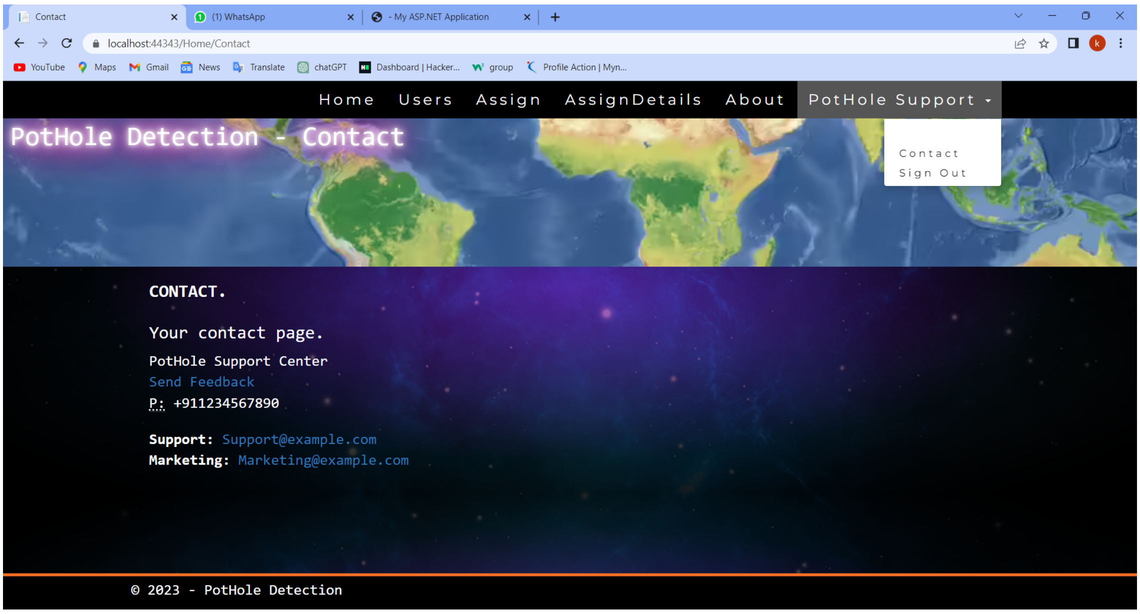


Figure 9.13: Website Interface 7

9.2 Result

A smart pothole navigation system has been developed, consisting of both a mobile application and a website. The mobile application is designed for use by the general public, allowing citizens to access and utilize the system. On the other hand, the website is specifically intended for administrative officers and contractors involved in the management and maintenance of the road infrastructure.
The primary component of the smart pothole navigation system is the user application. Upon registration, users gain access to a dashboard that offers several key features. The dashboard includes options such as "Create Case," "All Cases," and "View Map."

In the "Create Case" section, users can raise a complaint regarding a pothole by either capturing a photo using the device's camera or uploading an image from their gallery. This feature enables users to report potholes they encounter on the roads.

All registered cases can be viewed in the "All Cases" section, providing users with an overview of the reported potholes. This section allows users to track the progress of their complaints and stay informed about the status of reported potholes.

The "View Map" option provides users with the ability to input their source and destination locations. Upon doing so, the map displays the entire route, highlighting potholes as red flags. This feature assists users in identifying and planning routes that avoid roads with reported potholes, enhancing their navigation experience and potentially minimizing vehicle damage.

The website serves as a platform for administrators and contractors involved in the smart pothole navigation system. Both administrators and contractors have the ability to register and log in to the web portal. Once logged in, they are presented with a comprehensive dashboard containing various sections, including "Home," "Assign," "Assign Information," "About the Program," and "Pothole Support."

The "Home" section provides an overview and summary of the system, offering a centralized hub of information for administrators and contractors.

In the "Assign" section, administrators can view and manage different parameters such as user assignments and assignment information. This allows them to allocate specific potholes to contractors for repair and monitor the progress of each assigned task.

The "Assign Information" section provides detailed information about the assignments, specifying which potholes are assigned to which contractors and displaying the progress status for each individual pothole. This feature assists administrators and contractors in efficiently managing the repair process.

The "About the Program" section offers a brief overview and description of the smart pothole navigation system, providing relevant information to users and stakeholders.

The "Pothole Support" section includes options such as logging out and access to helpline information. This ensures that administrators and contractors can seek assistance or contact support if needed.

The website serves as a crucial tool for administrators and contractors to effectively manage and monitor the overall system, facilitating streamlined communication, assignment tracking, and efficient maintenance of the road infrastructure.

Chapter 10

Deployment and Maintenance

10.1 Deployment and Maintenance

10.1.1 Installation and un-installation

Installation:

- Step 1: Enable "Unknown Sources" in Android Settings By default, Android devices block the installation of APK files from sources other than the Google Play Store. To allow installations from other sources, follow these steps:

  1. Open the "Settings" app on your Android phone.
  2. Scroll down and select "Security" or "Privacy," depending on your device.
  3. Look for the option labeled "Unknown Sources" or "Install apps from unknown sources."
  4. Enable the toggle switch next to it. You may see a warning about potential risks; proceed only if you trust the APK source.

- Step 2: Download the PotholeDetection APK File Download the PotholeDetection APK file you wish to install onto your Android phone. Ensure that you download APKs from trusted sources to avoid potential security risks.

- Step 3: Locate the Pothole Detection APK File After downloading the APK file, you need to locate it on your Android phone. You can usually find downloaded files in the "Downloads" folder or the "Files" app. Use a file manager app if you're having trouble locating the APK file.

- Start the Installation Process To begin the installation, follow these steps:

1. Tap on the Pothole Detection APK file you downloaded. This action will typically display a prompt asking for confirmation to install the app.
2. Read the permissions requested by the app.
3. Tap the "Install" button to proceed with the installation.
4. Wait for the installation process to complete. It may take a few seconds or longer.
5. Once the installation is finished, you'll see an "App Installed" or similar message.

- Step 5: Launch the Installed PotholeDetection App

Uninstallation: To uninstall an APK (Android Package) file from an Android application, follow these step-by-step instructions:

- Step 1: Open Android Settings Open the "Settings" app on your Android phone.
- Step 2: Access App Settings In the Settings menu, scroll down and look for an option called "Apps," "Applications," or "App Manager."
- Step 3: Locate the App to Uninstall In the App Settings menu, you'll see a list of all installed applications on your Android device. Scroll through the list or use the search bar at the top to find the app you want to uninstall. Tap on the app to open its settings.
- Step 4: Open App Details and Uninstall Within the app's settings, you'll see information about the app, including its storage usage and permissions. Look for a button labeled "Uninstall" and tap on it.
- Step 5: Confirm Uninstallation A confirmation dialog will appear, asking if you want to uninstall the app. Read any additional information provided, such as data deletion warnings. If you're certain you want to uninstall the app, tap "OK" or "Uninstall" to confirm.
- Step 7: Uninstallation Complete Once the uninstallation process is finished, you'll see a message confirming that the app has been uninstalled. You can now close the Settings app and proceed with other tasks.

10.1.2 Maintenance

1. Data management: Regularly update and maintain the database of live pothole data. Implement mechanisms to remove outdated or irrelevant data and ensure the accuracy and integrity of the information.

2. Bug fixing and updates: Continuously monitor the system for any issues, bugs, or vulnerabilities. Release regular updates and bug fixes to address identified problems and improve system performance.

3. User feedback and support: Establish channels for users to provide feedback, report issues, or suggest improvements. Actively engage with users and address their concerns promptly to maintain user satisfaction.

4. Security and privacy: Implement robust security measures to protect the data collected and ensure user privacy. Adhere to data protection regulations and regularly update security protocols to address emerging threats.

5. Scalability and performance optimization: Monitor system performance and scalability as user numbers grow. Optimize the system to handle increasing data volumes and user requests efficiently.

6. Collaboration with authorities: Maintain a cooperative relationship with relevant municipal authorities responsible for road maintenance. Collaborate to ensure the accuracy of the data and leverage their expertise to prioritize repairs and enhance road safety.

7. Technology upgrades: Stay updated with advancements in navigation systems, data processing, and communication technologies. Continuously evaluate and incorporate new technologies that can enhance the system's functionality and user experience.

Conclusion and Future Scope

10.2 Conclusion

This system aims to tackle the challenges faced by citizens as a result of potholes by providing a platform for direct communication with the government. It offers several benefits to citizens in achieving three main objectives. Firstly, citizens can actively contribute by reporting the locations of newly formed potholes, enabling the government to swiftly identify and address them. Secondly, citizens can monitor the progress of pothole repairs, ensuring transparency and accountability in the government's actions. Lastly, utilizing a navigation system, citizens can proactively check for potholes along their route, facilitating smoother and safer journeys. By effectively addressing pothole-related issues, this system has the potential to reduce accidents and improve road safety. Implementing smart city initiatives not only benefits the residents by enhancing their quality of life but also has positive implications for the overall development and progress of the city.

10.3 Future Scope

The future scope for a navigation system depicting live potholes holds immense potential for leveraging technology to improve road safety, optimize maintenance efforts, and enhance the overall driving experience. Continued advancements in data collection, analysis, and communication technologies can lead to more effective and efficient management of pothole-related issues on road networks. Not only this but the system can further be in use for various purposes such as mentioned below:

- The navigation system can establish partnerships with road maintenance agencies, enabling seamless communication between the system and these authorities. This collaboration can facilitate timely pothole repairs based on the reported data, ensuring a quicker response and improved road conditions.

- The road can be surveyed using a vehicle with infrared sensors mounted on it, this will help in 3-dimensional mapping of the road and will help in detecting the depth and structure of the pothole.

- A potential improvement to the current application is the addition of a feature that facilitates direct communication between citizens and contractors without the need for an intermediary, such as an admin. This would enable more straightforward and streamlined communication between citizens and contractors. This feature could be incorporated into applications that already have a direct communication channel between citizens and contractors.

- In the future, the capabilities of this system could potentially be expanded to include the detection of accidents and various disasters, along with assessing their severity. This could lead to a more comprehensive and advanced system that provides critical information to both citizens and concerned authorities.

References

[1] Mallikarjun Anandhalli., and Vishwanath Baligar, "Indian pothole detection based on CNN and anchor-based deep learning method," In Proc. in International Journal of Information Technology · February 2022.

[2] Hsiu-WenWang, Chi-Hua Chen, Ding-Yuan Cheng, Chun-Hao Lin, and Chi-Chun Lo1., "A Real-Time Pothole Detection Approach for Intelligent Transportation System," In Proc. of Hsiu-WenWang et al, Taiwan, China, 2015.

[3] Aparna, Yukti Bhatia, Rachna Rai, Varun Gupta, Naveen Aggarwal, Aparna Akula, "Convolutional neural networks-based potholes detection using thermal imaging," In Privacy iJournal of King Saud University – Computer and Information Sciences 34 (2022) 578–588, February 2019.

[4] Artis Mednisy, Girts Strazdinsy, Reinholds Zviedrisy, Georgijs Kanonirs, Leo Selavoy, "Real Time Pothole Detection using Android Smartphones with Accelerometers," In Proc. of Institute of Electronics and Computer Science 14 Dzerbenes Str., Riga, LV 1006, Latvia Faculty of Computing University of Latvia 19 Raina Blvd., Riga, LV 1586, Latvia. CIRCA-2020) IEEE Xplore Part Number: CFP20N67-ART; ISBN: 978-1-7281-5374-2

[5] Surekha Arjapure, D. R. Kalbande., "Road Pothole Detection using Deep Learning Classifiers," In Proc. International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878 (Online), Volume-8 Issue-6, March 2020.

[6] Alharbi A, Halikias G, Yamin M, et al (2021) Web-based framework for smart parking system. Int J Inf Tecnol 13:1495–1502.

[7] Akagic, A., Buza, E., Omanovic, S., 2017, May. Pothole detection: an efficient visionbased method using RGB color space image segmentation. In: Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2017 40th International Convention on (pp. 1104-1109). IEEE.

[8] Moazzam, I., Kamal, K., Mathavan, S., Usman, S., Rahman, M., 2013. Metrology and visualization of potholes using the Microsoft Kinect sensor. In: Proceedings of the 16th International IEEE Annual Conference on Intelligent Transportation Systems (2013), 1284-1291.

[9] An, K.E., Lee, S.W., Ryu, S.K., Seo, D., 2018, January. Detecting a pothole using deep convolutional neural network models for an adaptive shock observing in a vehicle driving. In: Consumer Electronics (ICCE), 2018 IEEE International Conference on (pp. 1-2).

[10] Erikson, J., Girod, L., Hull, B., 2008. The pothole patrol: using a mobile sensor network for road surface monitoring. In: Proceedings of the 6th International Conference on Mobile Systems, Applications, and Services (2008), pp. 29-39.

[11] J. Lin and Y. Liu, "Potholes detection based on SVM in the pavement distress image, " in Distributed Computing and Applications to Business Engineering and Science (DCABES), 2010 Ninth International Symposium on, 2010, pp. 544-547.

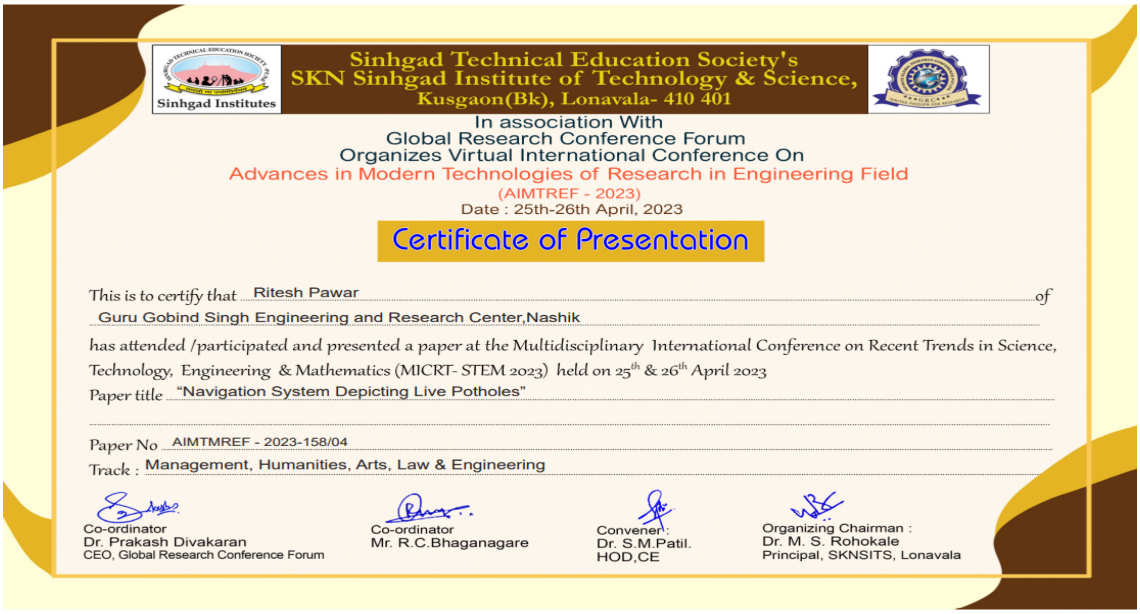
[12] Kwang Eun An , Sung Won Lee ,Seung-Ki Ryu , Dongmahn Seo, Detecting a pothole using deep convolutional neural network models for an adaptive shock observing in a vehicle driving, IEEE International Conference on Consumer Electronics (ICCE), 2018. 7. Hyunwoo Song, Kihoon Baek and Yungcheol Byun.

Abbreviation

Sr No.	Abbriviation	Full Form
1	KNN	K Nearest Neighbour
2	SQL	Structured Query Language
3	CNN	Convolutional Neural Networks

Chapter 11

Paper Publication and Certificate Details



Sinhgad Technical Education Society's
SKN Sinhgad Institute of Technology & Science,
Kusgaon(Bk), Lonavala- 410 401

Sinhgad Institutes

In association With
Global Research Conference Forum
Organizes Virtual International Conference On
Advances in Modern Technologies of Research in Engineering Field
(AIMTREF - 2023)
Date : 25th-26th April, 2023

Certificate of Presentation

This is to certify that Ritesh Pawar of
Guru Gobind Singh Engineering and Research Center,Nashik
has attended /participated and presented a paper at the Multidisciplinary International Conference on Recent Trends in Science, Technology, Engineering & Mathematics (MICRT- STEM 2023) held on 25th & 26th April 2023
Paper title "Navigation System Depicting Live Potholes"

Paper No AIMTMREF - 2023-158/04
Track : Management, Humanities, Arts, Law & Engineering

Co-ordinator
Dr. Prakash Divakaran
CEO, Global Research Conference Forum

Co-ordinator
Mr. R.C.Bhaganagare

Convener :
Dr. S.M.Patil.
HOD,CE

Organizing Chairman :
Dr. M. S. Rohokale
Principal, SKNSITS, Lonavala

Figure 11.1: Paper Publication Certificate

Figure 11.2: Paper Publication Certificate

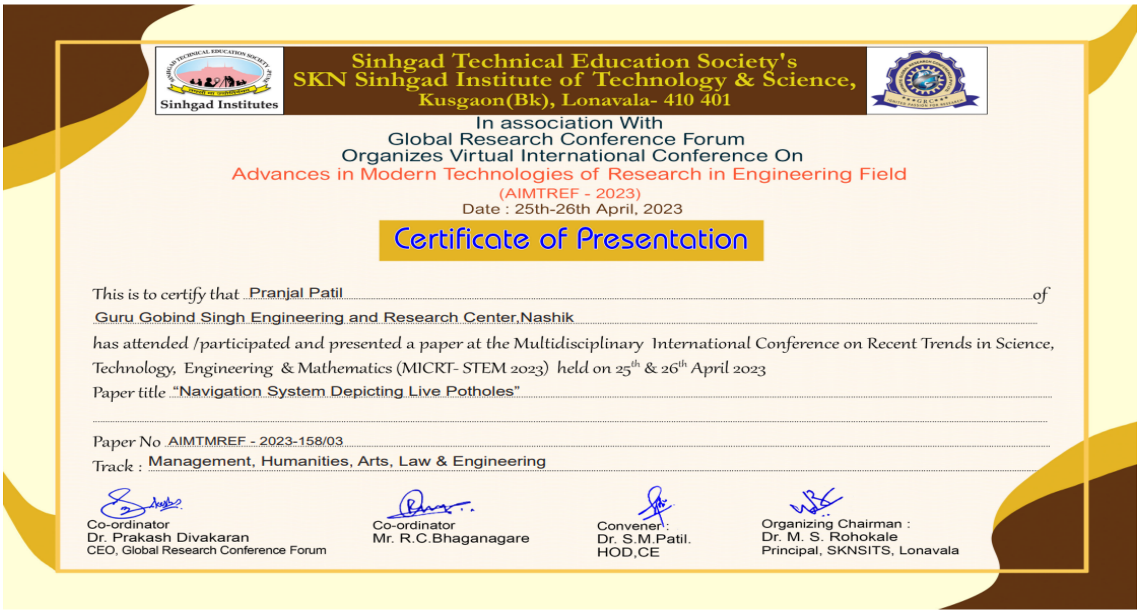


Figure 11.3: Paper Publication Certificate

Figure 11.4: Paper Publication Certificate

Figure 11.5: PICT Poject Competition

Figure 11.6: PICT Poject Competition

Figure 11.7: PICT Poject Competition

Figure 11.8: PICT Poject Competition

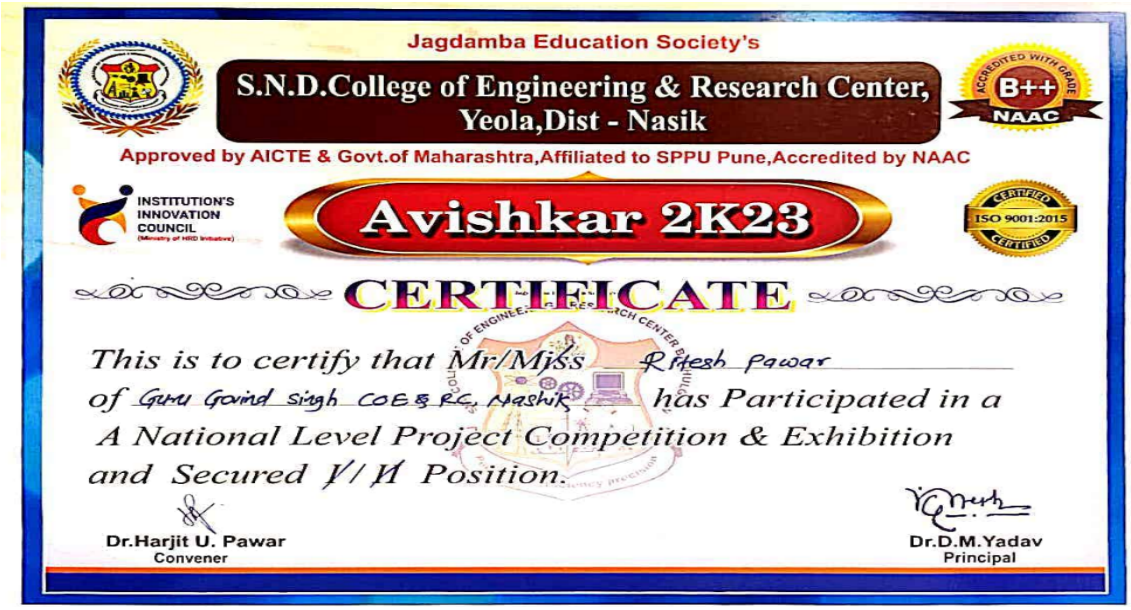


Figure 11.9: Avishkar Poject Competition

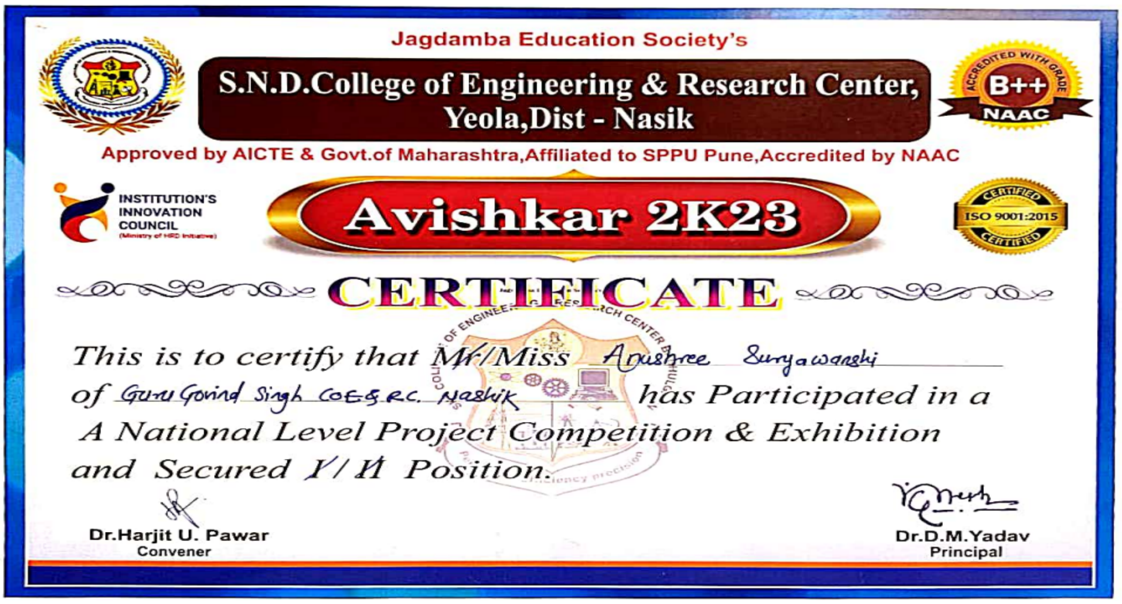


Figure 11.10: Avishkar Poject Competition

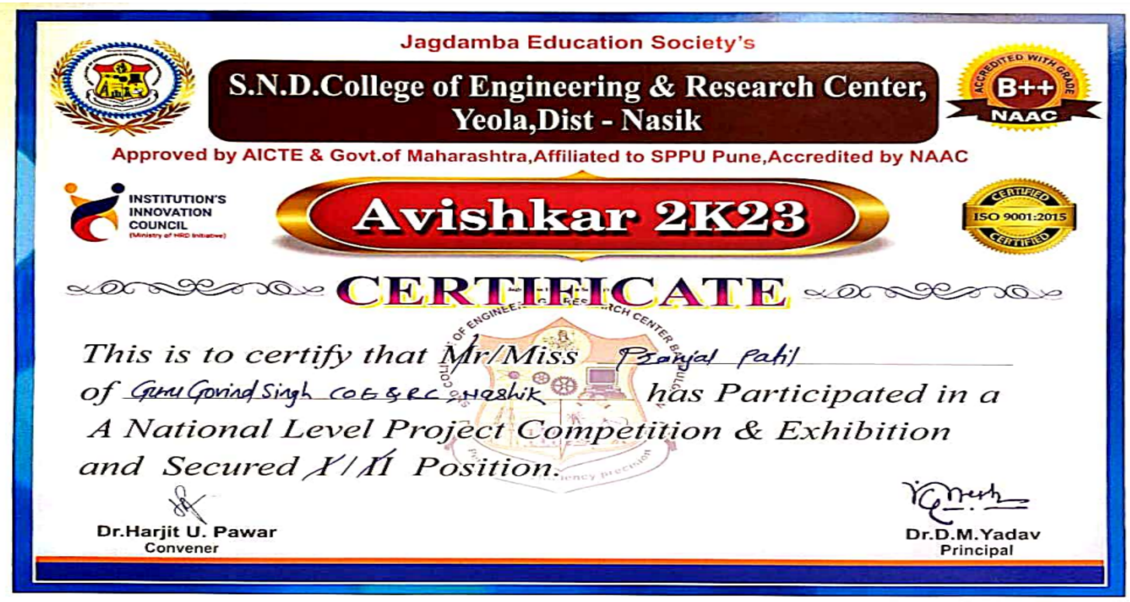


Figure 11.11: Avishkar Poject Competition

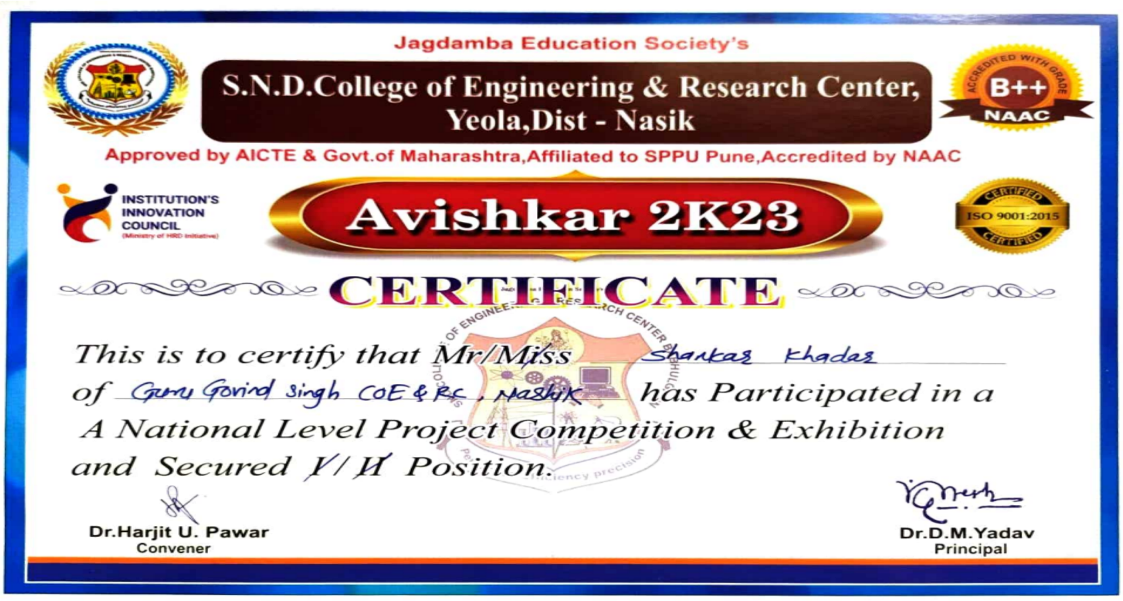


Figure 11.12: Avishkar Poject Competition

Figure 11.13: SNJB Poject Competition

Figure 11.14: SNJB Poject Competition

Figure 11.15: SNJB Poject Competition

Figure 11.16: SNJB Poject Competition

Figure 11.17: GGSF Poject Competition

Figure 11.18: GGSF Poject Competition

Figure 11.19: GGSF Poject Competition

Guru Gobind Singh Foundation's.

Guru Gobind Singh College of Engineering & Research Center, Nashik.

Computer Engineering Students' Association (COSA)

GGSF GURU GOBIND SINGH FOUNDATION

ESTD. : 2013

Certificate

This is to Certify that

Shankar Rajabhau Khadar

Has Participated in Project Poster Presentation Competition of titled "Road Potholes Detection and Notification Using Color Code" Organized by Computer Engineering Students' Association (COSA), Guru Gobind Singh College of Engineering & Research Center, Nashik. on 18th October 2022.

Mr. S. G. Shukla
Head of Department

JFPAY4-CE000033

Dr. N. G. Nikam
Principal

Made for free with Certify'em

Figure 11.20: GGSF Poject Competition

Paper Publication and Certificate Details

Published paper in Science Technology and Development Journal on "Automated Documeent Verification System" Paper ID:19, ISSN-0950-0707.