SOFTWARE REQUIREMENT SPECIFICATIONS

for

POTHOLE DETECTION IN FLOODED CONDITIONS.

Version 1.0 approved

Prepared By:
2106039_Mohammad Saud Khan
2106054_Sagnik Mukherjee
2106262_Sourav Guharay
2106303_Shashank Gupta

KIIT, Deemed To Be University
School of Computer Science & Engineering

21st November'2023

Table of Contents 1

Revision History

Name	Date	Reason for Changes	Version

1. Introduction

1.1 Purpose

When the road gets flooded during rains, it becomes very difficult to ascertain if there are any open drains, manholes or potholes underwater, resulting in many accidents. The students have individually thought about an idea to have a device in vehicles, which could detect such openings on the road so that accidents can be prevented.

1.2 Product Scope

The creation of a specialized method for locating and assessing submerged or partially hidden potholes on water-covered road surfaces is required for pothole identification in flooded situations. With this wide-ranging device, precise data can be collected even in inclement weather thanks to the use of flood-resistant sensors and cutting-edge detecting technology.

To distinguish between potholes and other surface abnormalities brought on by floods, reliable data processing algorithms must be included, along with real-time monitoring capabilities that set off warnings and messages for quick maintenance activities.

1.3 Uses

Pothole detection systems in flooded conditions can enhance road safety, prevent vehicle damage, improve road maintenance planning, enable cost-effective maintenance strategies, enhance flood preparedness, facilitate data collection and analysis, and raise public awareness.

Incorporating pothole detection systems in flooded conditions within the System Requirements Specification can help ensure the system's effectiveness and relevance in challenging weather conditions.

1.4 Features

The purpose of this project is to detect potholes in roads in flooded conditions including monitoring flood conditions and water levels, providing real-time data and using GPS to pinpoint the exact pothole locations, notifying relevant authorities, suggesting maintenance schedules based on pothole data, including comprehensive documentation for system operation and maintenance, ensuring how the system can handle varying flood levels and volumes of data and implementing security measures to protect data and prevent tampering.

1.5 Overview

To detect potholes during floods, this document contains the functional and non-functional requirements, design specifications, testing and validation protocols, deployment strategy, and maintenance methods.

2. Specific Requirements

2.1 Functional Requirements

The exact capabilities and functionalities that the system must have to work properly are defined by the functional requirements for a pothole detecting system under flooded situations. These specifications guarantee that the system can reliably identify potholes, send out notifications, and record pertinent information even in inclement weather. Below mentioned are the functional requirements of the above system:

2.1.1 Image Acquisition in Flooded Conditions

The acquisition of images in flooded situations refers to the process of obtaining visual data or photographs in areas impacted by water. Here's an explanation of each component:

- **Resilient Environment:** Flooded circumstances entail water covering the ground, causing tough conditions for picture capturing owing to low sight, water distortion, and safety issues.
- **Lighting Considerations:** Adequate lighting is necessary since water may scatter and absorb light, lowering vision. Specialized underwater or floodlights may be utilised to illuminate the area of interest.
- **Optical Quality:** High-quality lenses and optical components are needed to minimize distortion caused by water, while anti-reflective coatings assist in decreasing glare and reflections on the water's surface.
- **Stability:** Maintaining a steady camera posture is crucial for clear picture capture. Tripods or stabilizers help guarantee the camera remains stable, even in running water.
- **Waterproof Housings:** In certain circumstances, cameras are put within waterproof housings or enclosures to protect them from direct water contact while permitting picture capture underwater.
- **Aerial Imaging**: Drones fitted with specialized cameras may collect aerial photos of flooded areas, offering an overview of the degree of flooding.

2.1.2 Image Processing for Pothole Detection

To discover and identify potholes in photos or video recordings, image processing for pothole identification employs digital image analysis techniques. The stages involved in image processing for pothole identification are summarised as follows:

- **Image Acquisition:** Using cameras or sensors placed on moving vehicles or aerial drones, the procedure begins by taking pictures or video frames of the road surface.
- **Pre-processing:**
  Image Enhancement: Adjust the contrast, brightness, and sharpness of the image to improve its quality. Important elements and features are highlighted as a result.
  Noise Reduction: Apply filters or algorithms to the image to decrease noise or undesirable artefacts.
- **Segmentation:**
  Thresholding: To distinguish the road surface from the backdrop, convert the image to binary format by establishing a threshold value.
  Edge recognition: Edge detection can help you spot snags in the road by locating the edges or borders of items in a picture.
  Segmentation with Colour: Use colour information to separate road surfaces from other features using colour-based segmentation.
- **Visualisation:** Detected potholes might be overlaid on the source image or given visual cues to help with automated decision-making or visual examination.
- **Keeping and reporting data:** Keeping track of details regarding found potholes, such as their location, size, and severity and creating reports so that road authorities or maintenance workers can take appropriate action.

2.1.3 Pothole Detection Algorithm in Flooded Conditions

Due to the presence of water covering the road surface, which can distort visual cues and render conventional pothole detection methods less efficient, spotting potholes in flooded areas can be difficult. However, by integrating multiple sensor technologies and data processing strategies, it is feasible to construct a specialised pothole-detecting system for flooded settings.

- **Sensor Selection:** Choose flood-resistant sensors like LiDAR and radar for data collecting.
- **Data Collection:** Gather data using automobiles or drones negotiating flooded roadways.
- **Data Pre-processing**: Correct for water-induced distortions and standardise data.
- **Pothole Detection:** Analyse sensor data for road surface anomalies.
- **Data Fusion:** Combine data from various sensors for increased accuracy.
- **Machine Learning:** Train models to recognise potholes from other irregularities.

2.1.4 Alert Detection

For informing relevant parties about the presence of potholes, such as road maintenance authorities or vehicle drivers, alert detection in pothole-detecting systems is essential.

- **Alert Initiation:**
  Real-Time alarm: If a possible pothole is found in real-time, the system quickly sends out an alarm.
  Accumulated Data: In certain instances, notifications are sent based on the total number of potholes that have been spotted over a given period of time.
- **Alert Levels:**
  Visual Alerts: Drivers are alerted to the existence of potholes using visual indications, like as symbols or warnings on a dashboard or mobile app.
  Audio Alerts: To inform the motorist of the pothole, Audible Alerts can be turned on.
  Data logging: Notifications are recorded together with pertinent data about the pothole, such as its GPS location and degree of severity.
- **Transmission Methods:**
  Vehicle-to-Driver: Alerts are transmitted from the car to the driver via the dashboard or a mobile app in vehicle-based systems.
  Vehicle-to-Infrastructure: Alerts can also be sent to maintenance teams or road authorities for quick response.
  Cloud Integration: Pothole alarm data may be sent to a cloud-based platform for additional analysis and reporting.
- **Alert Severity Levels:** Pothole detecting systems may categorise discovered potholes by severity to ascertain the urgency of the alarm and the proper action.

2.1.5 Data Logging

- The system must record information about potholes that are found when there is excessive flooding.
- The GPS coordinates, timestamp, and degree of pothole severity must all be recorded in the logged data.
- Data must be securely preserved for subsequent analysis and upkeep.

2.2 Non - Functional Requirements

Non-functional requirements are crucial characteristics of a software system or product that specify how it should behave rather than what it should accomplish. They are also known as quality requirements or system attributes. These specifications are essential for ensuring that the programme satisfies expectations for performance, usability, security, and other crucial features. Following are a few typical groups of non-functional requirements:

2.2.1 Performance

- **Response Time:** Identifying and reporting potholes, the system should be able to respond in real-time or very close to it.
- **Throughput:** Describing the system's capacity for processing a large amount of visual data quickly, especially in places where potholes are common.

2.2.2 Reliability

- **Availability:** Ensuring ongoing pothole detection and reporting capabilities, maintaining a high degree of system availability.
- **Fault Tolerance:** Building the system with the ability to absorb possible errors with grace and minimally interrupt operation.

2.2.3 Usability

- **User Interface (UI):** Ensuring that administrators and maintenance teams can easily monitor and manage pothole data using an intuitive and user-friendly UI.
- **Accessibility:** In order to serve a diverse user base, the system must adhere to accessibility guidelines and be usable by people with impairments.

2.2.4 Security

- **Data Security:** Safeguarding sensitive information, pothole data should be encrypted while it is in transit and at rest.
- **User Authentication and Authorization:** Managing access to the functions and data of the system, putting safe user authentication and authorization procedures in place.
- **Privacy:** Ensuring that data protection laws are followed, particularly when working with location data or personally identifiable information (PII).

2.2.5 Scalability

- **Horizontal Scalability:** Planning for the possibility of expanding the system horizontally by including more drones or cars with sensors to cover more regions.
- **Vertical Scalability:** As the amount of data gathered increases, it should be ensured that the system can manage higher data processing demands.

2.2.6 Maintainability

- **Modifiability:** Making sure that the system is easily modified to consider modifications to sensor technologies, pothole pothole-detecting, or data storage needs.
- **Record-keeping:** Helping the administrators, developers, and maintenance teams manage and debug the system, and provide thorough documentation.

2.3 External Interface Requirements

External interface requirements would outline how a software system for pothole detection interfaces with different external entities and data sources that are pertinent to its operation.

2.3.1 User Interfaces

- **User Dashboard:** Describing the interface that emergency responders, transit departments and flood management authorities use to communicate with the pothole detecting system.
- **Mobile Application:** Describing the features to the field workers having access to a mobile application who need to report potholes or view system warnings while on the job site.

2.3.2 Hardware Interfaces

- **Sensor Integration:** Describing how the software integrates with physical sensors mounted on cars or in flood-prone locations to gather information about road damage and flood conditions. and the process through which the program communicates with these sensors to get pothole information.
- **GPS Integration:** Describing how the system communicates with GPS devices to acquire precise position data if GPS data is utilised to pinpoint pothole sites.

2.3.3 Software Interfaces

- **Forecasting Weather:** Providing details on how the programme communicates with outside weather forecasting services to obtain up-to-date meteorological information and the protocols, and APIs that are utilised to get meteorological and flood data.
- **Mapping Services:** Describing the interaction with mapping APIs (e.g., Google Maps) if mapping services are used to visualise pothole locations on maps.

2.3.4 Communication Interfaces

- **Alert Notifications:** Specifying how the system notifies emergency responders or pertinent authorities of alerts.
- **Data Exchange:** Describing the data shared between the external systems and the pothole-detecting system.
- **Security Protocols:** Explaining the security procedures, such as authentication and encryption, to safeguard data while it is being transmitted if sensitive data is being shared.

2.4 System Requirements

2.4.1 Database Requirements

- **Logical Database Requirements:** A logical database can stretch over multiple physical hard disks and information files. The data storage unit is still a single database for information retrieval purposes. To have a logical database, all given hard disks and information files must be accessible from a single source.

- **Physical Database Requirements:** A physical database is technically a smaller unit of storage referred to as a company, field, record or table, de-pending on how much information the physical storage device contains. A field is the smallest unit of storage housing only a single file.

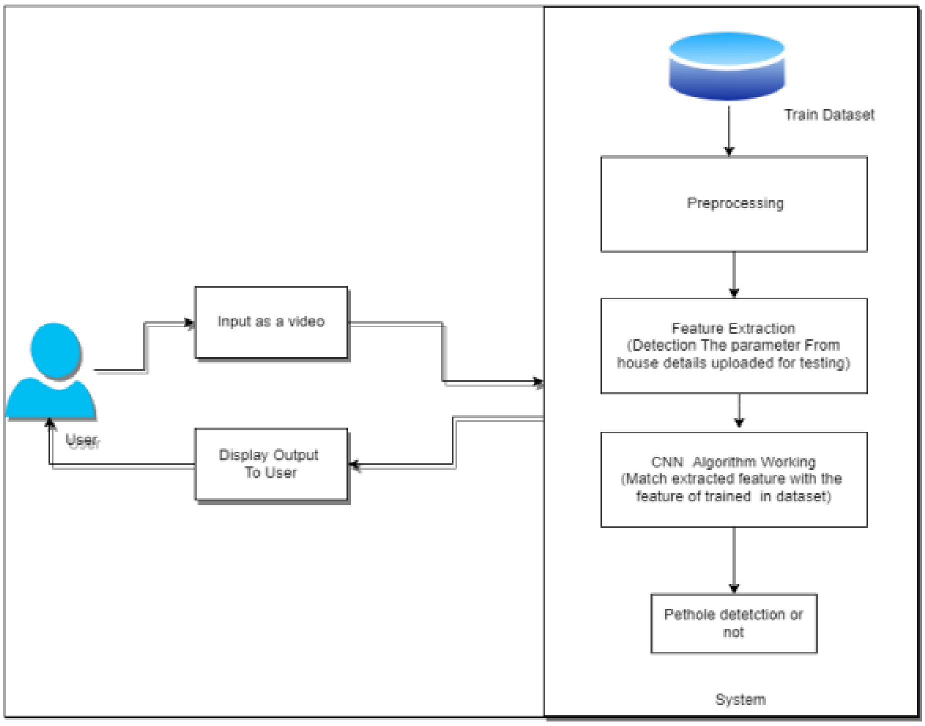
2.4.2 Database Requirements

- **Operating System:** Windows 7-10.
- **Coding Language:** Python Version 3
- **IDE:** Spyder
- **Windows Platform:** Anaconda Navigator Hardware Requirements
- **System:** Pentium IV 2.4 GHz.
- **Hard Disk:** 40 GB.
- **Monitor:** 15 VGA Color.
- **Mouse:** Logitech

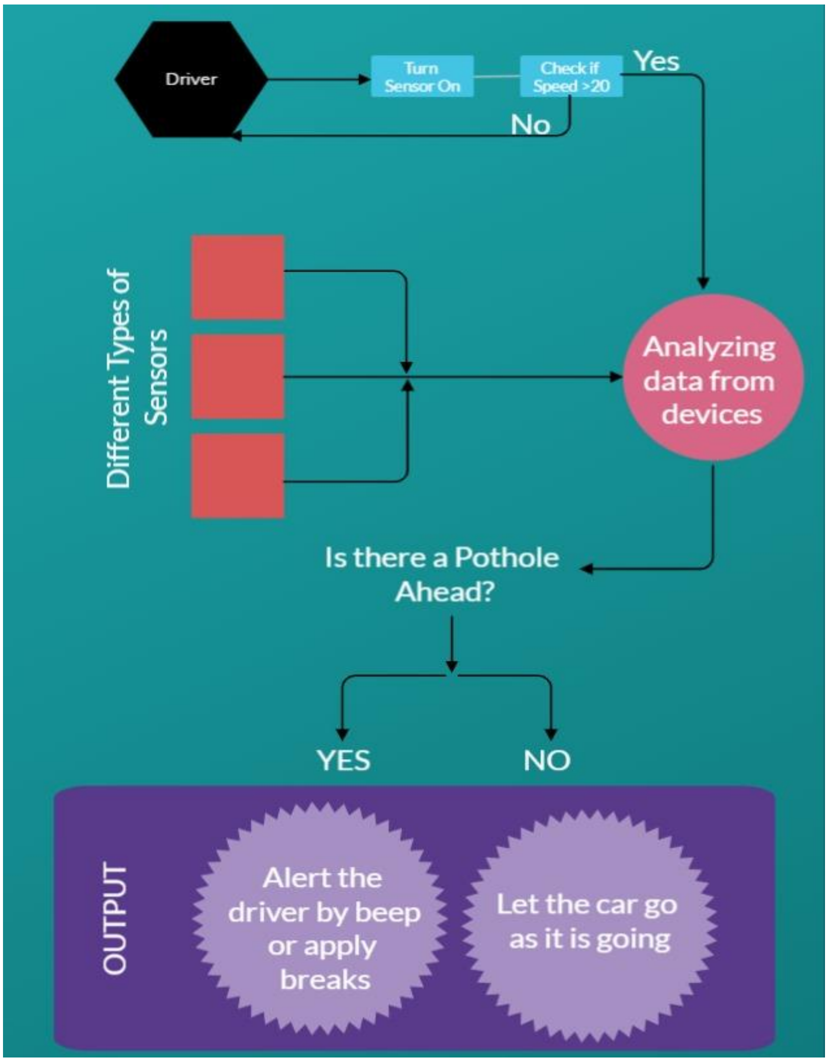
3. System Architecture

3.1 Overview

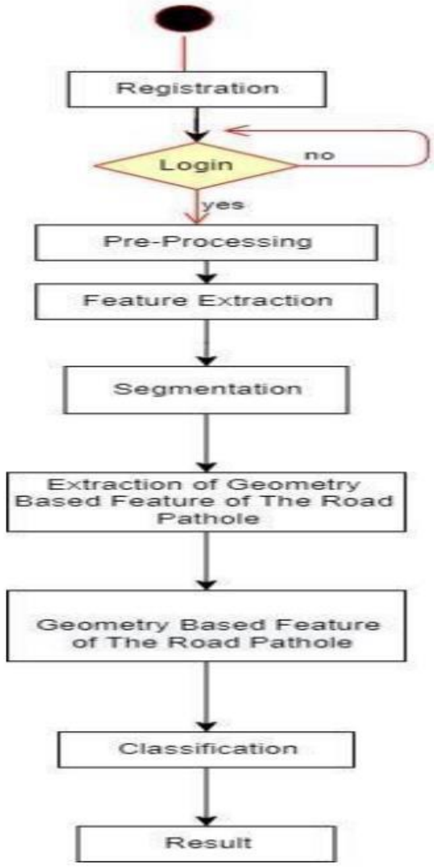
Two significant and well-known classification algorithms are tested in this work. The dataset is subjected to SVM and density-based clustering techniques. The datasets are evaluated to determine their accuracy. Every algorithm has been applied to the training dataset, and its accuracy performance is assessed in conjunction with the testing dataset's predictions. A data flow is produced during the whole analysis process.



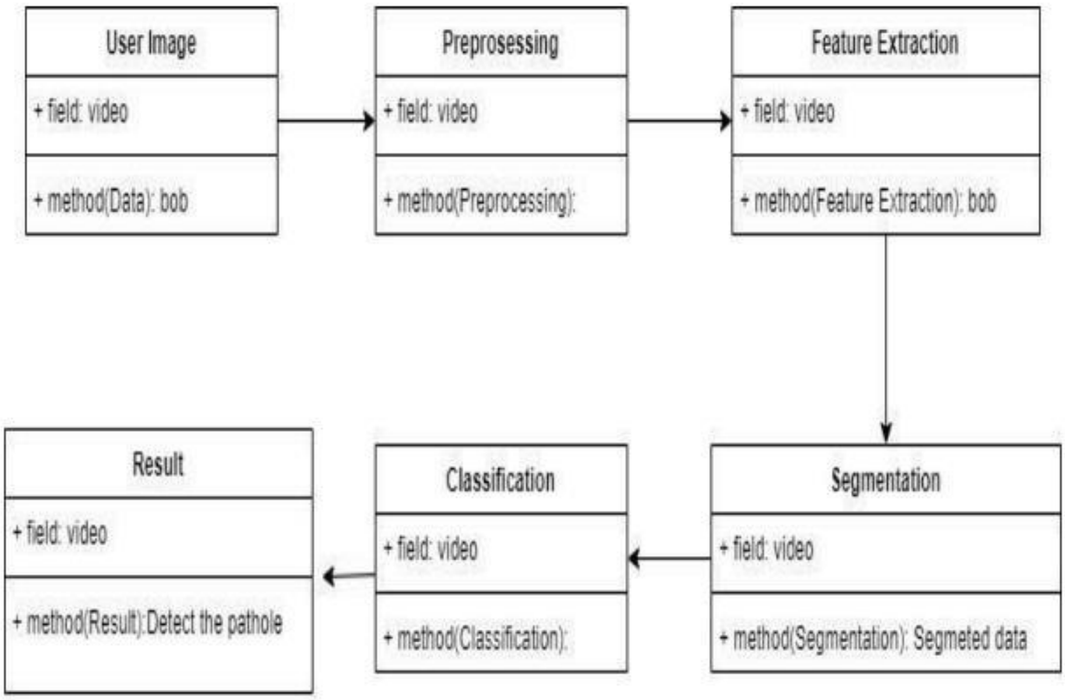
3.2 High-Level Diagram



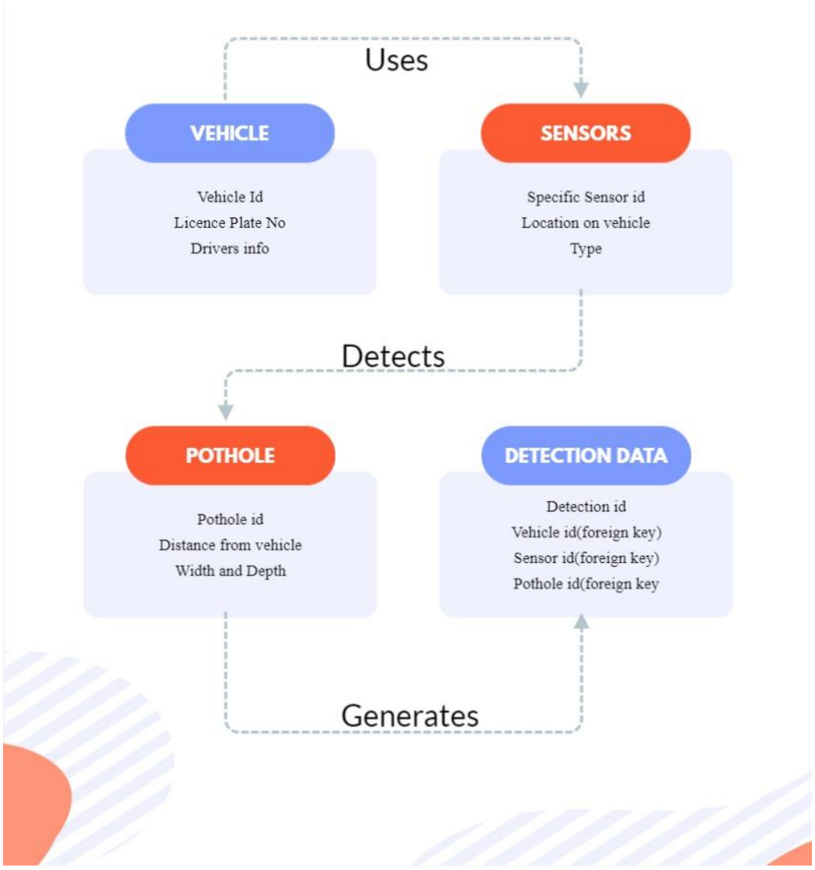
3.3 Activity Diagram



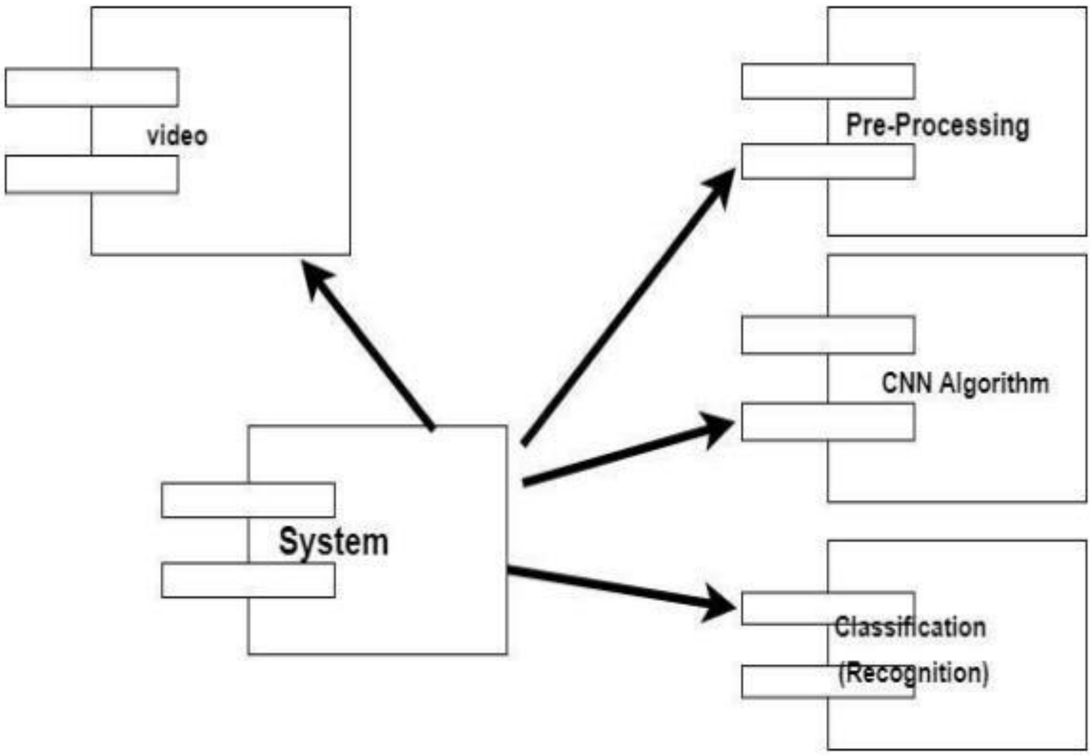
3.4 Class Diagram



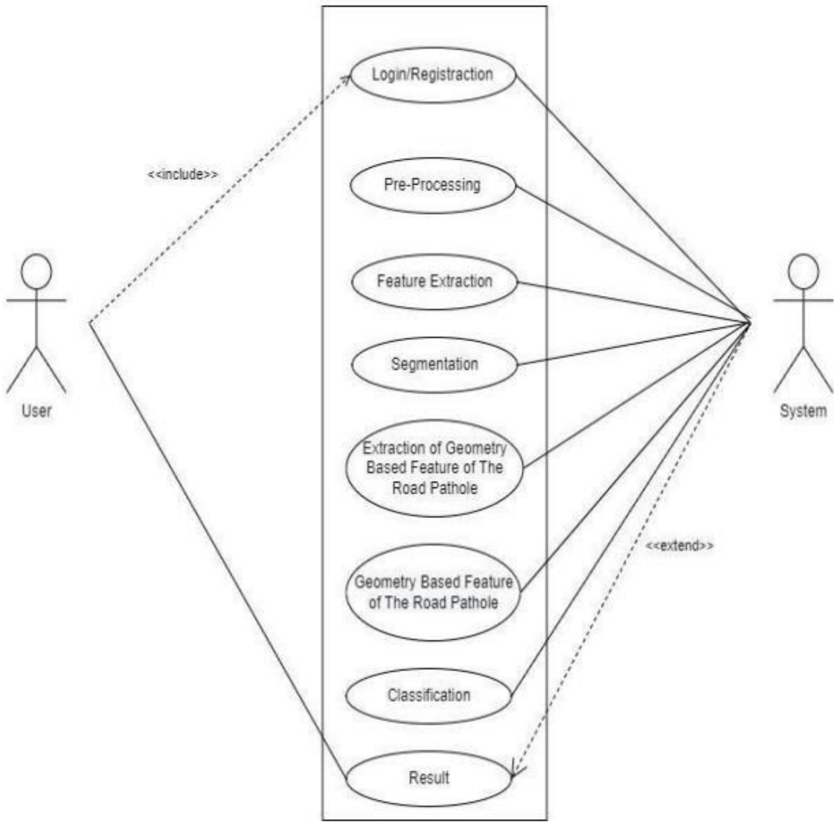
3.5 Entity-Relationship Diagram



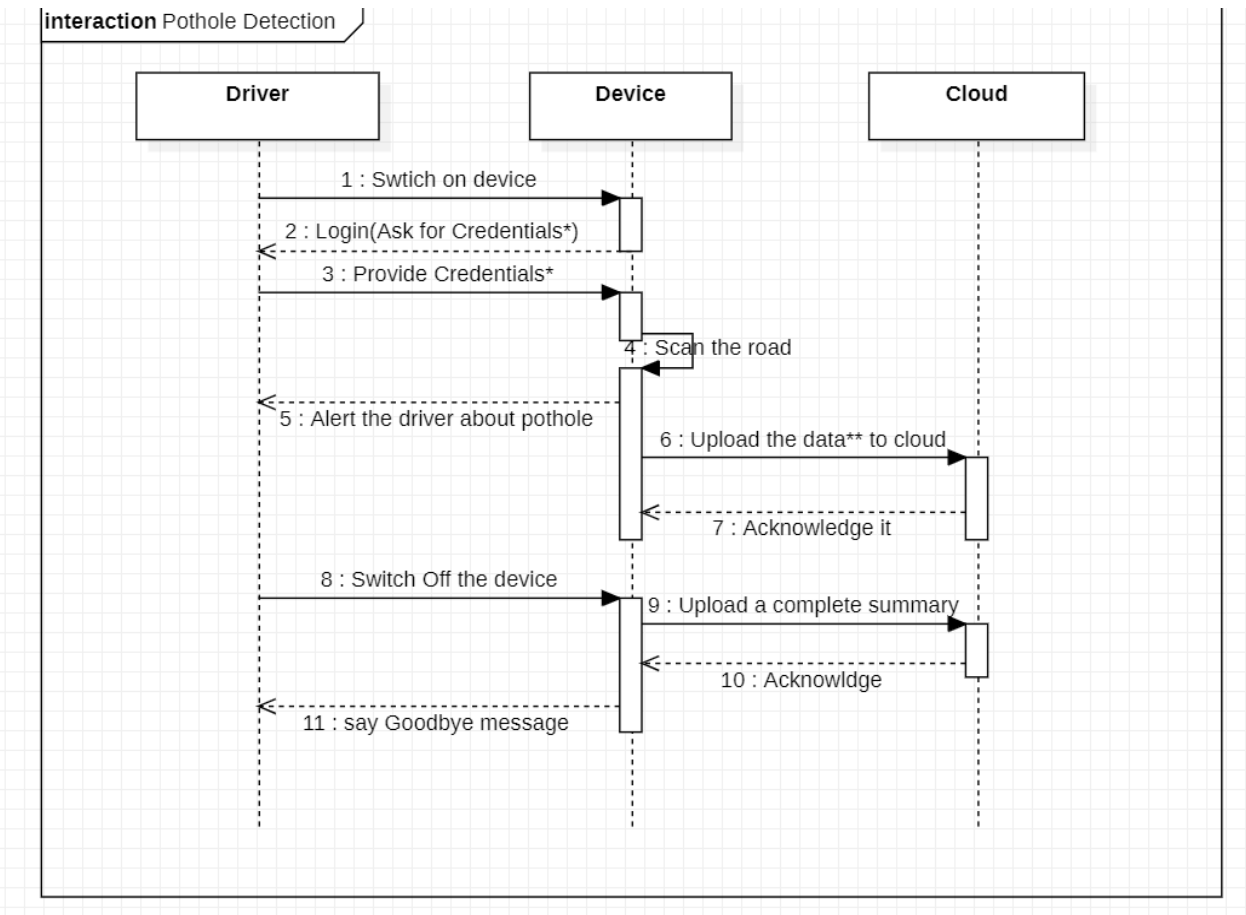
3.6 Component Diagram



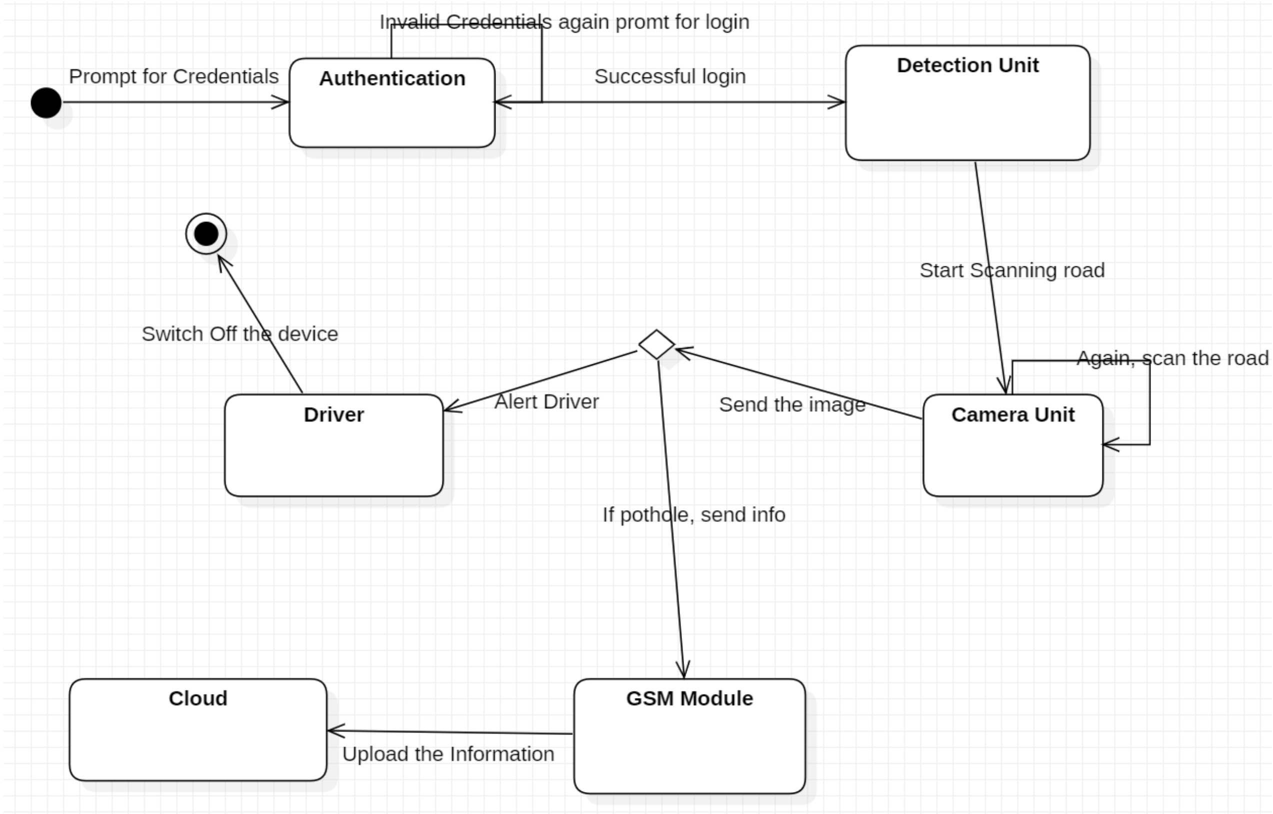
3.7 Use-Case Diagram



3.8 Sequence Diagram

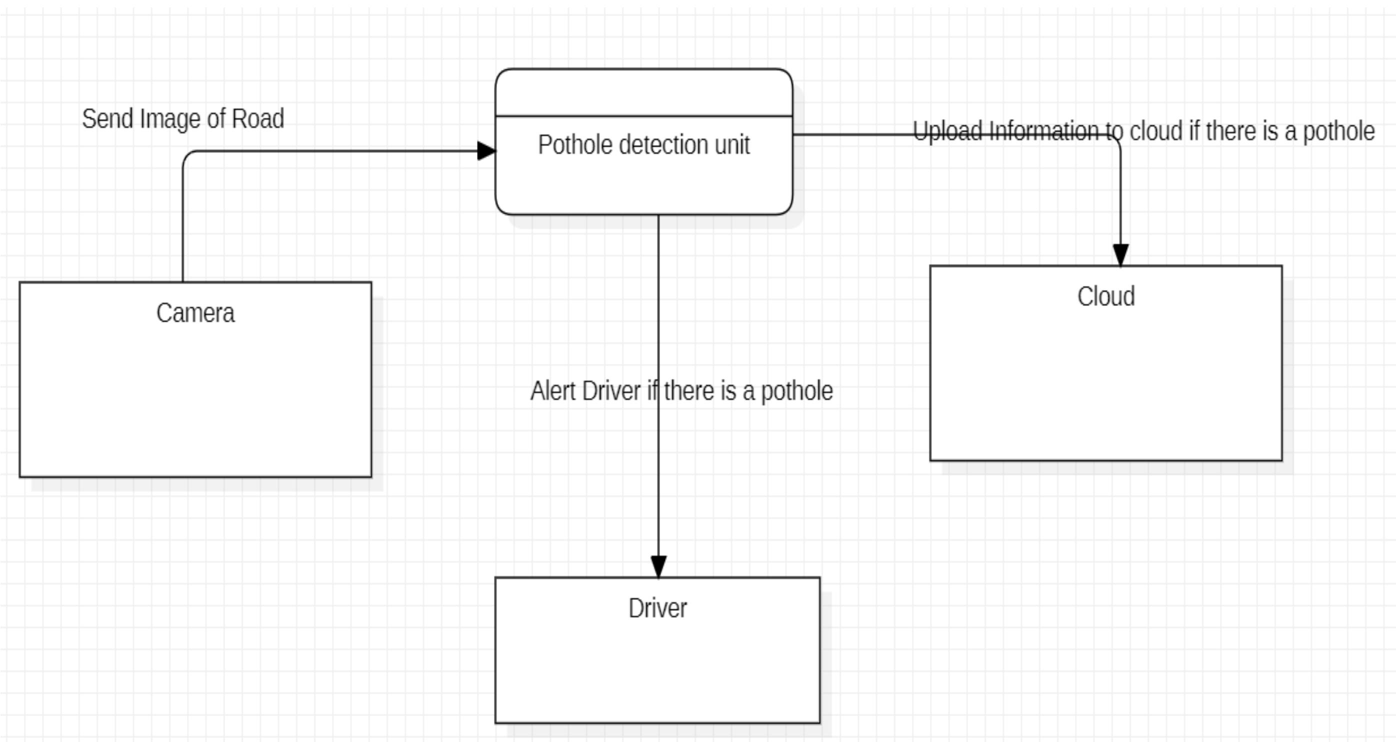


3.9 State Diagram

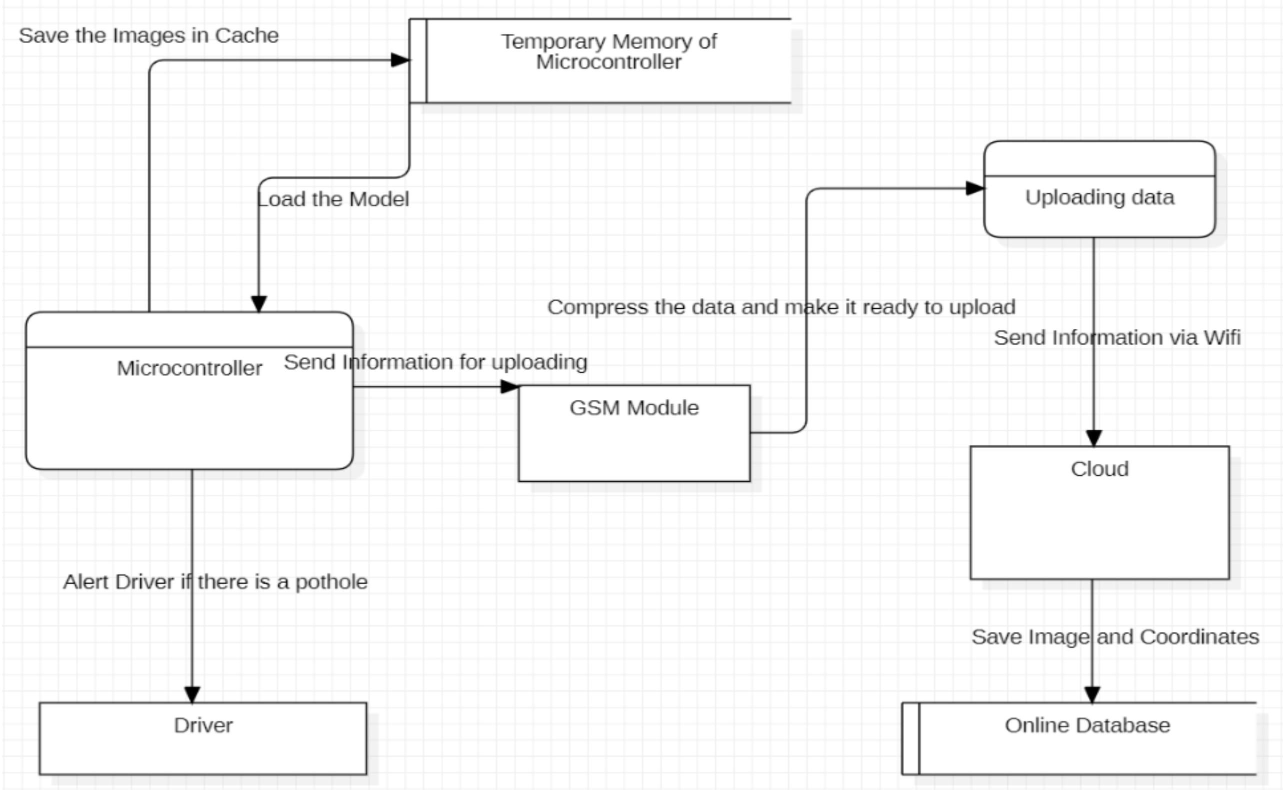


3.10 Data Flow Diagram (DFD)

Level – 0 DFD



Level – 1 DFD



3.11 Unified Modeling Language (UML) Diagram