VERSION 1.0

20TH JUNE, 2020

Pothole Detection

PRESENTED BY: YAGNAHAUN JONNADULA   ANURAG PEDDI

ABHISHEK THAKUR

IDRBT

INTRODUCTION

PROBLEM STATEMENT

The goal of our project is to design a Pothole detection System which assists the driver in avoiding potholes on the roads, by giving the driver prior warnings. For example, it can be like a buzzer or series of LED, if the driver is approaching a pothole driver may be warned regarding the pothole on the road. In this paper, we propose a robust and straightforward design of a portable and affordable device that can alert the driver about the detected pothole. The hardware system installed in a moving vehicle can automatically detect and report potholes via image-processing of Raspberry-Pi microcontroller. The detailed images of the pothole and its location are stored and viewed through the GOOGLE API.

Index Terms—Pothole Detection, Hardware system, Raspberry-Pi microcontroller, GOOGLE API.

LIKELY IMPACT

In this fast-moving world, people want to reach their destinations as soon as possible. Some apps suggest the routes which can make us reach our goal early, and other apps show all possible ways with traffic congestion for whatever reason it may be, such as google maps and many more.

But there are fewer apps that tell the condition of the road, whether it is good to travel or not, whether it suggests the route to the driver based on the state of the way. So this model mainly focuses on the travel safety of the passengers and updates the passengers with the best route to travel.

BUDGET MANAGEMENT

BUDGET TABLE

Module Name	*Quantity*	*Price*
Raspberry Pi 3	*1*	*3090/-*
Raspberry Pi 5MP Camera Module	*1*	*425/-*
Mini Night Vision Camera 1080P	*1*	*1719/-*
Jumper Wires	*As per requirement*	*400/- -- 500/-*
Remaining Required Accessories	*As per requirement*	*4000/- -- 5000/-*
Total Costing		*10000/- -- 11000/-*

TEAM STRUCTURE

TEAM GOALS

- ***The main goal of the team is to develop a model that makes the travel safer by suggesting the best and safer route***

OUR TEAM

Team Member	*Member Name*	*Roll Number*
Member 1	*Anurag Peddi*	*17MCME13*
Member 2	*Yagnahaun Jonnadula*	*17MCME22*

DEVELOPMENT PROCESS

The embedded device processes the incoming stream of video and detects any pothole in the flow, if any, it immediately intimates the user with a signal.

It also saves the information of pothole detected and coordinates in a file for cross-validation.

SYSTEM ANALYSIS

STAKEHOLDERS

The below are all the stakeholders taking part in the total scenario:

- ***User(driver)***
- ***DataAnalyst who takes the data provided by the devices and perform some data analysisf.***
- ***Software Developer who debugs any issues after the installation.***
- ***Company which produces these systems.***
- ***IT Staff who installs and maintain the system.***

COMPETITIVE ANALYSIS

People in this time need instant solutions for problems; one of the issues is that they need to reach their destinations early, and many applications can do this work. Still, what our app does differently makes us stand out from others, it not only suggests which road reaches your destination faster but also indicates which route is most safe to travel.

The work which can be done by our app is the most critical work, which makes it to find its way into the minds of the users.

REQUIREMENTS

The embedded device processes the incoming stream of video and detects any pothole in the flow, if any, it immediately intimates the user with a signal. It also saves the information of pothole identified and coordinates in a file for cross-validation.

FUNCTIONAL REQUIREMENTS:

The system can also provide it with two features like:

- ***Detect pothole and Alert driver***
- ***Alert the driver about the pothole***

The type of software which we are going to use is python scripts; drivers are our expected users, are going to use this in embedded systems.

FUNCTIONAL USER REQUIREMENTS:

The user needs not to provide any of their confidential details for the module to work.

FUNCTIONAL SYSTEM REQUIREMENTS:

For the module to work, the car should be running and should also limit their speed to under 40kmph.

NON-FUNCTIONAL REQUIREMENTS:

As we are using a pre-trained model for detecting the potholes, the time constraints will be on alert the driver about the pothole and suggesting to him the best and safe possible route.

PRODUCT DETAILED REQUIREMENTS

The system restarts once for every 6 hours of work, which will flush all the cache, additional to this we need to run the monthly updates to keep the software up to date. As the lifespan of the pi camera is less compared to raspberry pi, we need to change it once every five years.

ORGANIZATIONAL REQUIREMENTS

One can create an account/log in using his driving license.

DETAILED VIEW

HARDWARE SUB-SYSTEMS

The following are the hardware sub-sytems in the model:

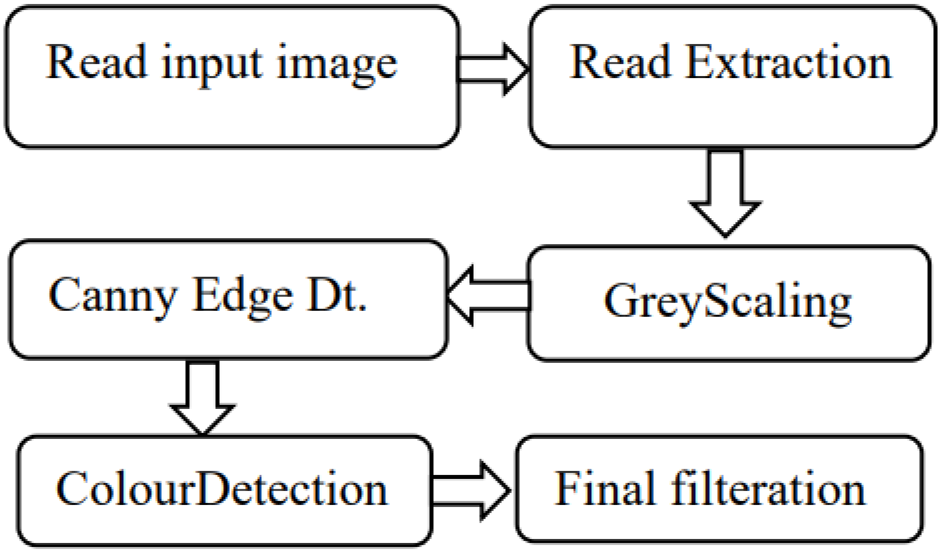
- ***Raspberry Pi 3***
- ***Raspberry Pi 5MP Camera Module***
- ***Mini Night Vision Camera 1080P***
- ***Jumper Wires***

SOFTWARE SUB-SYSTEMS

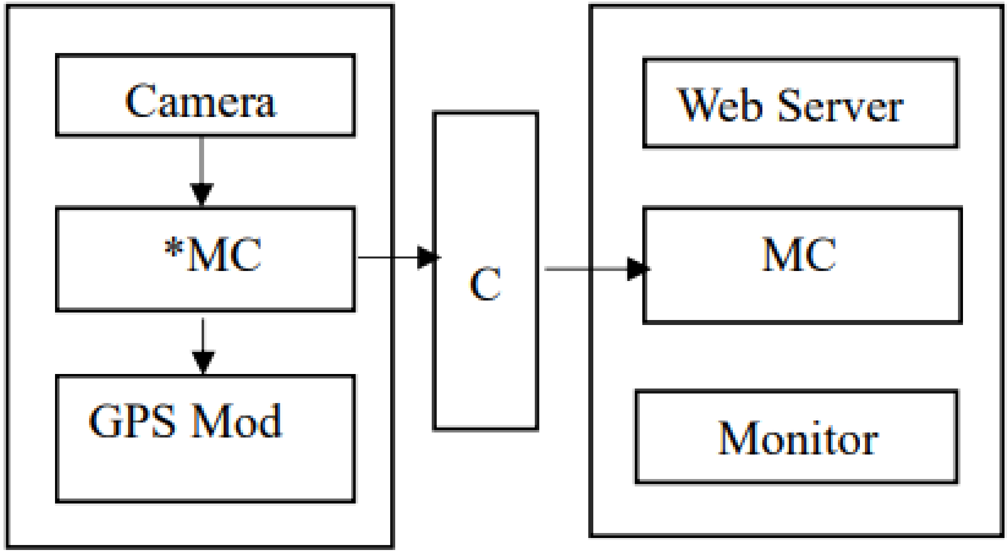
The following are the software sub-systems in the model:

- ***AWS CloudBuid***
- ***AWS Config***

THE IMAGE PROCESSING SCHEME



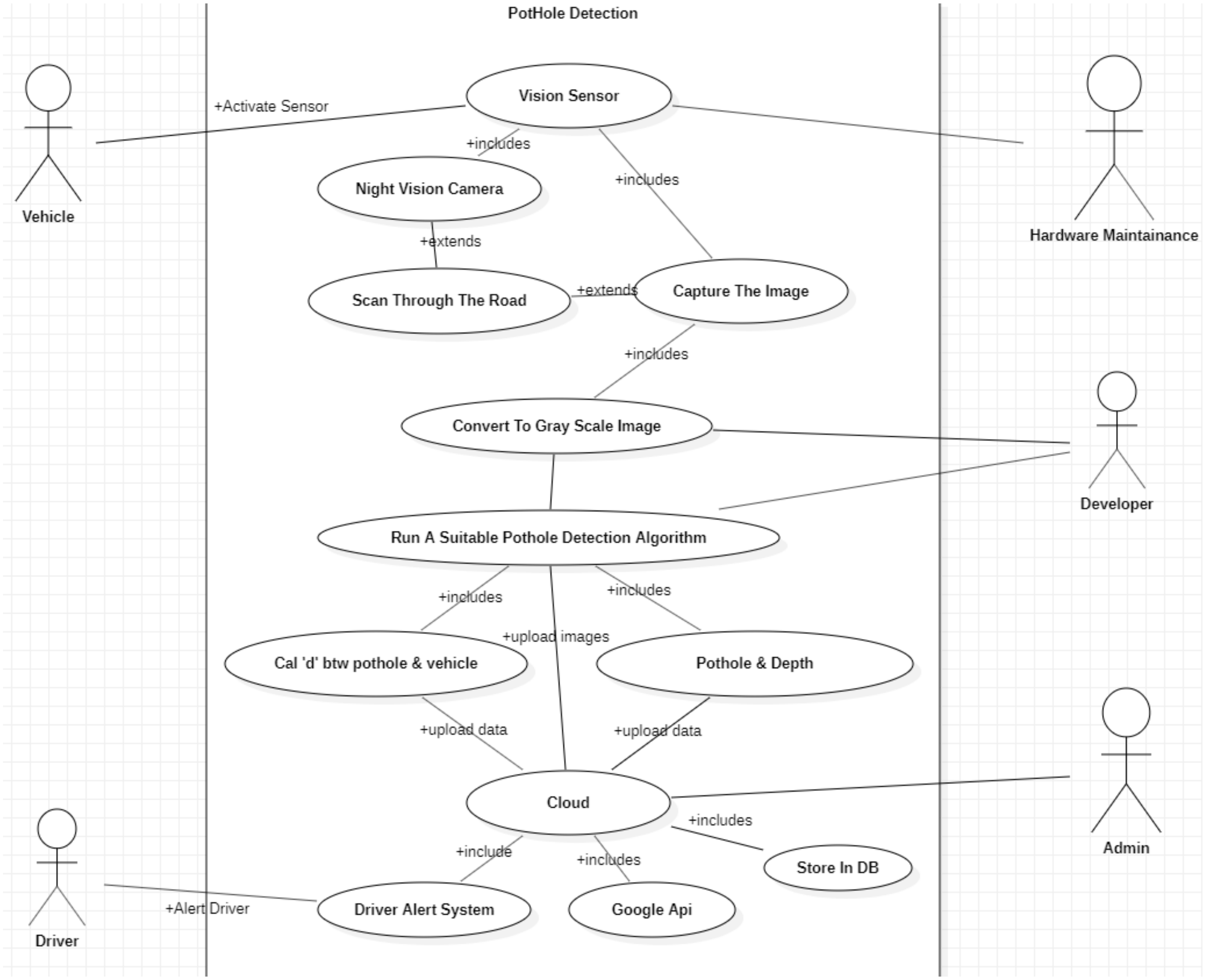
BRIEF OVERVIEW OF THE MODEL



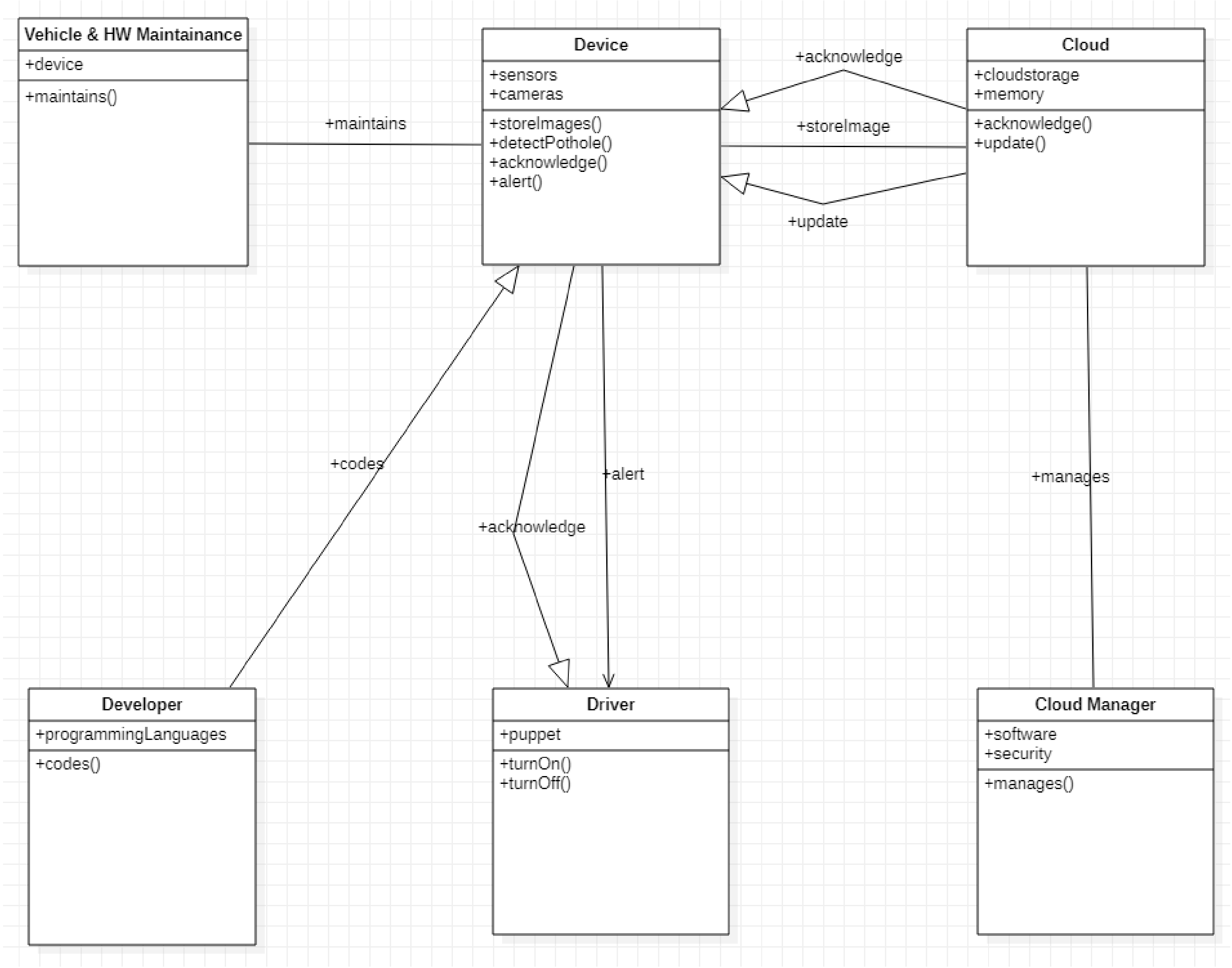
MC and C here indicate microcomputer and cloud respectively

SYSTEM UML DIAGRAMS

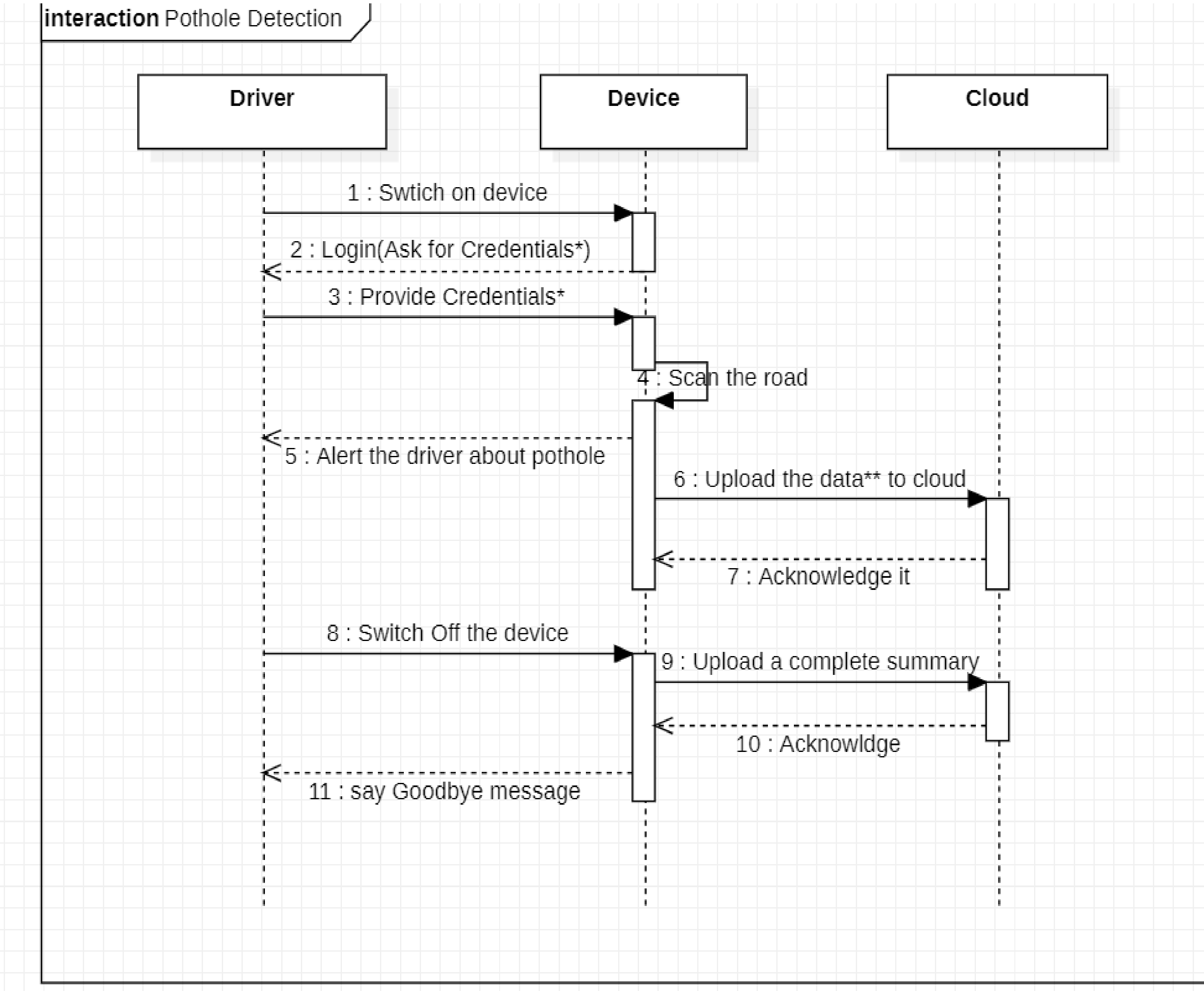
USE CASE DIAGRAM



CLASS DIAGRAM



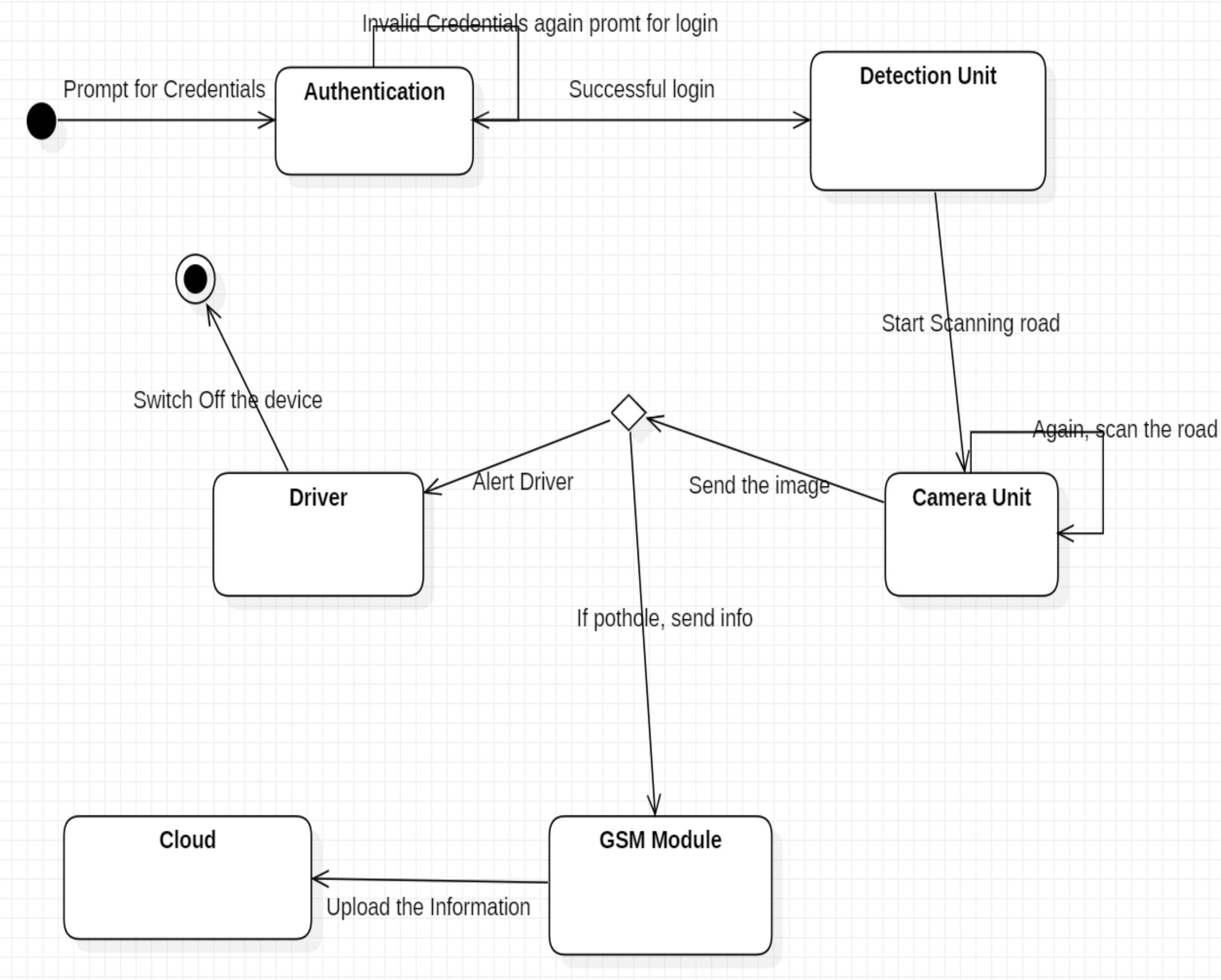
SEQUENCE DIAGRAM


interaction Pothole Detection
Driver
Device
Cloud
1 : Swtich on device
2 : Login(Ask for Credentials*)
3 : Provide Credentials*
4 : Scan the road
5 : Alert the driver about pothole
6 : Upload the data** to cloud
7 : Acknowledge it
8 : Switch Off the device
9 : Upload a complete summary
10 : Acknowldge
11 : say Goodbye message

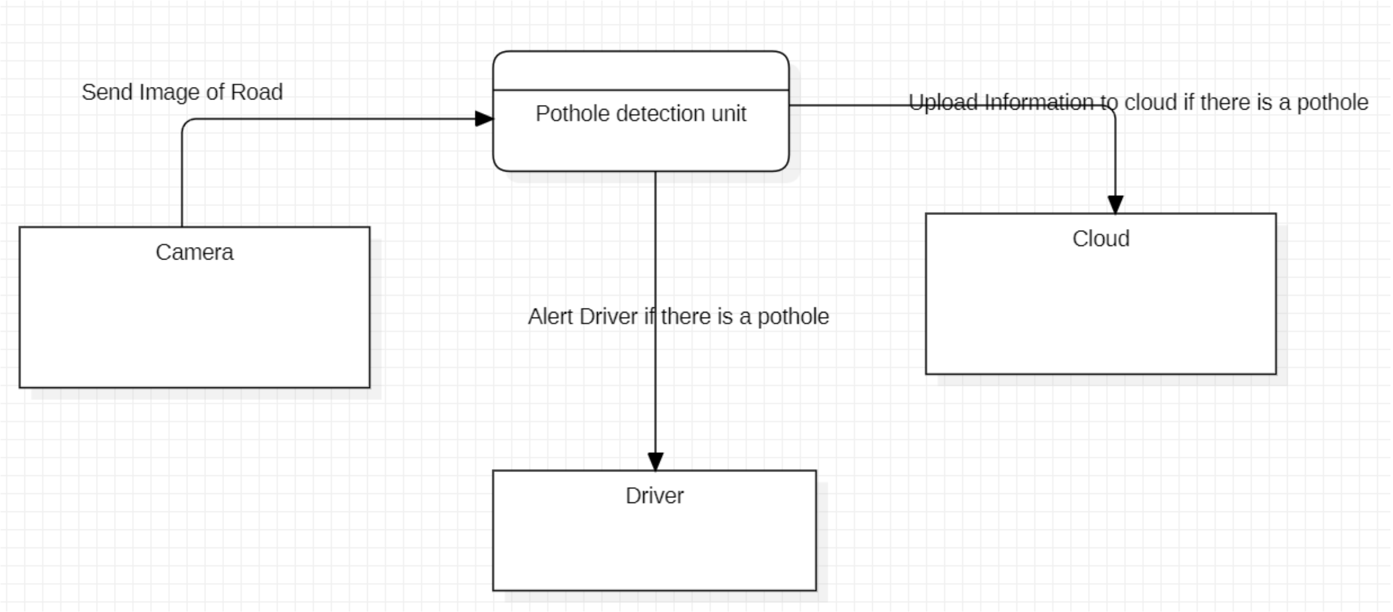
* Credentials such as Driving License and Password

** Data such as Type of Pothole and Google Map coordinates

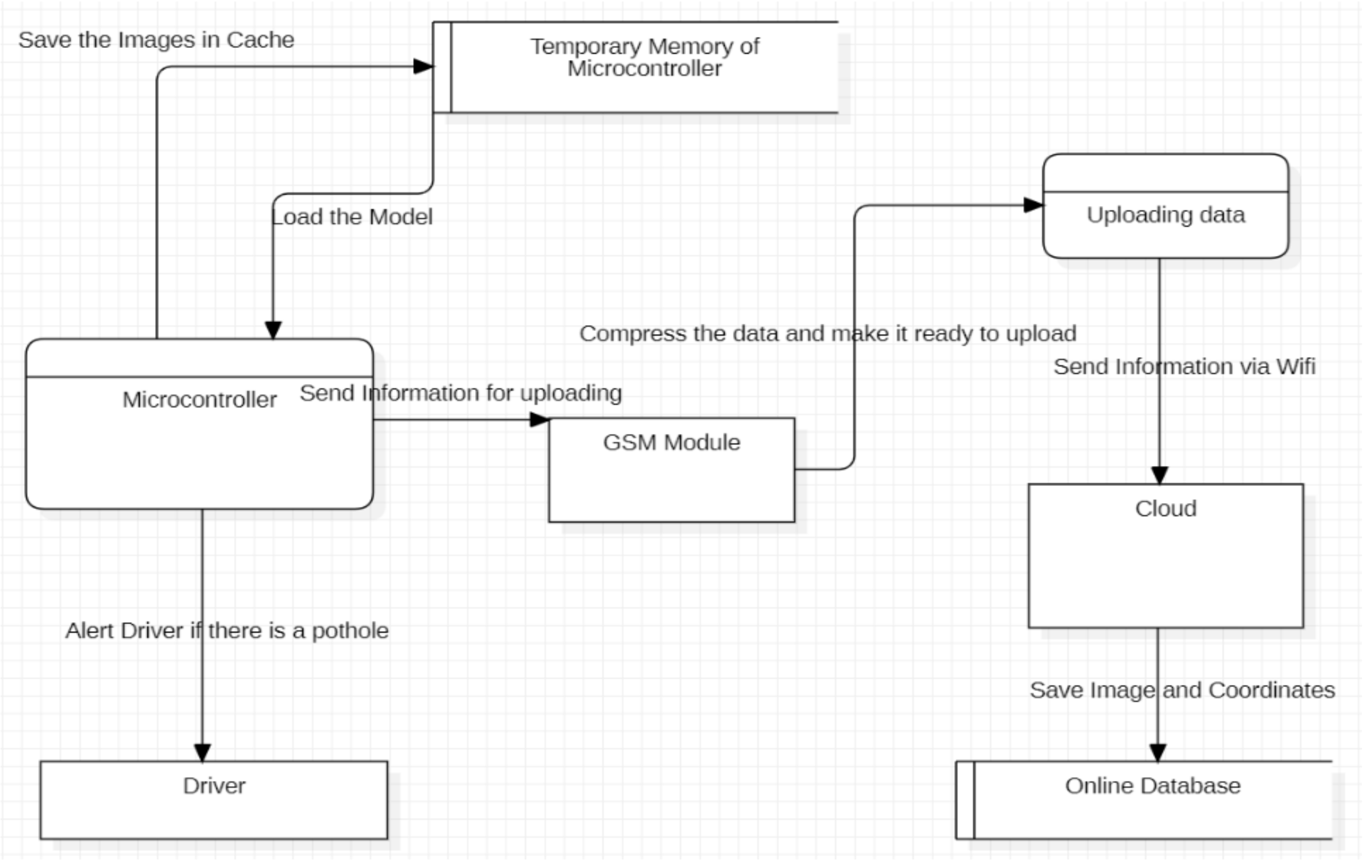
STATE DIAGRAM



LEVEL-0 DATA FLOW DIAGRAM


Send Image of Road
Pothole detection unit
Upload Information to cloud if there is a pothole
Camera
Cloud
Alert Driver if there is a pothole
Driver

LEVEL-1 DATA FLOW DIAGRAM


Save the Images in Cache
Temporary Memory of Microcontroller
Uploading data
Load the Model
Compress the data and make it ready to upload
Send Information via Wifi
Microcontroller
Send Information for uploading
GSM Module
Cloud
Alert Driver if there is a pothole
Save Image and Coordinates
Driver
Online Database

TEST PLAN

AUTOMATION

THE AUTOMATION OF THE MODEL GOES ON THIS PROCESS:

The sensor nodes on the car and the GPS receiver can detect or identify the potholes on the first note, or the manual recordings can also be used to update the location of the pothole to the data file which runs through a pothole detection algorithm which then can give the accurate positions of the detected potholes.

TESTING

UNIT TESTING:

Software test automation tools enable you to simplify testing and reduce time to release by automating functional tests for your applications.

Unit testing tools which we are going to use are:

- ***Pytest***
- ***Travis CI***

FIGURE 1 PYTEST

FIGURE 2 TRAVIS CI

VALIDATION TESTING:

For validation testing, we use the images which we have collected from sites to check whether all the scripts are bug-free. This validation testing is also done out during the demonstration to our stakeholders(excluding developers).

DEFECT TESTING:

For defect testing, we record some real-world videos of road and present it, which not only includes images of the road but also pictures of some patterns which are similar to potholes, to measure how accurate it is detecting potholes when both potholes and potholes like patterns are present.

MANAGEMENT FRAMEWORKS

CHANGE OR CONTROL MANAGEMENT

Replacement of raspberry pi should be made once every five years and additional updates every month. AWS Config is a service that let's assess, audit, and evaluate the configurations of AWS resources which we are using.

BUILD

The version control system which we are using is git. AWS CodeBuild is a fully managed continuous integration service that compiles source code, runs tests, and produces software packages that are ready to deploy.

FIGURE 3. GIT

PACKAGING

To make the python scripts standalone, we are using a package called pyinstaller. AWS packaging automates the process of packaging and publishing software to managed Windows and Linux instances across the cloud landscape, as well as to on-premises servers, through a single simplified interface.

PROJECT PLAN

TIMELINE FOR DEVELOPMENT OF THIS MODEL

Milestone Number	*Date On Which Milestone Is Achieved*	*Description*
Milestone 1	*March 12, 2020*	*First phase Idea completion*
Milestone 2	*April 19, 2020*	*Completion of planning of designing the model (Designing of the working model)*
Milestone 3	*Expecting Date* *September 15, 2020*	*Purchase of all the units required for the model*
Milestone 4	*Expecting Date* *December 10,2020*	*Assembling the units to a well-designed model*
Milestone 5	*Expecting Date* *January 1, 2021*	*Testing the model with all the data sets*
Milestone 6	*Expecting Date* *February – March , 2021*	*The final model that is ready to deliver*

RISKS AND ISSUES MANAGEMENT

RISKS

HARDWARE:

One of the dangers which the device faces is external accidents/shocks, which are unpredictables example:

- *Someone hitting it*
- *Snow covering the lens*
- *The camera being covered by some object(paper).*
- *Rainwater going into the circuits and frying them.*

SOFTWARE:

The other kind of risks which the device faces are privacy and security of data. Time to time updates about the potholes to the cloud and intimate the driver about it correctly . The data should be transmitted only in a secure environment.

RISK MITIGATION

HARDWARE

The above hardware risks can be avoided if the device have a dedicated slot where it is safe enough and should detect the potholes as well.

SOFTWARE

The privacy and the security of the device should be maintained by a good security analyst so that it would not be lost. And the software developer should schedule time to time checks about the cloud management and the software.

CONCLUSIONS

CHALLENGES

The challenges in front are, the budget as we cant take that to a high budget model so that it can be used by many and should get a high pixel camera with a low budget and the model should not make false interpretations of the pothole and make the user feel sure about this model.

POSSIBLE EXTENSIONS

- ***To make it budget-friendly.***
- ***Reduce the latency between the uploads.***
- ***Detect potholes even when the car is at a very high speed.***

There could be even more possibilities of extending the model to a state at which it not only detects the potholes but also detecting the fixed potholes and updating the earlier data and detecting speed breakers where the drivers need to slow down. Another extension that we can make is to estimate the depth of the pothole and to categorize them to a different different level of danger and alert the driver about that danger, whether he can travel on that road or not. And can be notified to higher officials for the fixing of these potholes.

CONSTRAINTS

There should be enough light to detect the pothole, and car should be moving with an average speed of 40-60km/hr to get a good quality of the video.

CHANGE CONTROL BOARD(CCB)

Identify who will serve on the CCB, which determines whether issues are within the current project scope and whether they should be addressed.

OUR WORK CAN BE FOUND AT

https://github.com/AnuragAnalog/Pothole-Detection