SPOT THE HOLE

15CSE387 OPEN LAB

PROJECT REPORT

Submitted by

BL.EN.U4CSE17001	ABHINAY KRISHNA CHITARALA
BL.EN.U4CSE17003	ABHISHEK SAI KANDIKATTU
BL.EN.U4CSE17004	ADITYA D UDAPUDI
BL.EN.U4CSE17010	ARAVIND V NAIR
BL.EN.U4CSE17036	GOLLA KEERTHANA
BL.EN.U4CSE17043	HARSHITA REDDY EDUGUDI

AMRITA SCHOOL OF ENGINEERING, BANGALORE

AMRITA VISHWA VIDYAPEETHAM

BANGALORE 560 035

JULY 2020

ABSTRACT

Potholes are the biggest problem faced by everyone today. Driving down these roads is a nightmare. Repaving the roads would be a big investment in many ways, not only would it improve the towns look, but it would also make it safer for the community members to drive on the roads. Now the problem lies in the fact that concerned departments are not able to coordinate to resolve the issue and citizens also find it hard to file a complaint.

Fundamentally there is a disconnection between civil agencies, people, and elected representatives. Our aim is to build a web-based dynamic reporting system which will facilitate the flow of information among all concerned stakeholders. It also provides better governance by more involvement of the public.

To automate the process and make tracking of potholes much easier we have planned on developing a Progressive Web App. It can be used by everyone irrespective of their choice of computing device, be it mobile, tablet or laptops. It will be lightweight and easy to install as well. Machine Learning and Computer Vision will be used to check if the pictures uploaded are valid pictures of potholes. Natural Language Processing will be used to analyze the description entered by the user and get a negativity score. Cloud Computing will be used to host the entire application and implement Continuous Testing / Continuous Integration and Continuous Deployment in the form of Development Operations. There will also be a heat map which would be displaying the location and severity of the potholes for easier tracking. Thus, all in all, we plan to build an end to end system.

Contents

ABSTRACT **2**

CHAPTER 1: INTRODUCTION **4**
1.1 Objective 4
1.2 Motivation 4
1.3 Problem Statement 4
1.4 Proposed Solution 4

CHAPTER 2: USE CASE DIAGRAM AND DESCRIPTION 5

CHAPTER 3: IMPLEMENTATION **7**
3.1 Overview of the Implementation Phase 7
Low-Level Diagrams: 7
High-Level Diagrams: 8
3.1 Screen Snippets of the Application 10

CHAPTER 4: RESULTS **18**

CHAPTER 5: CONCLUSION AND FUTURE SCOPE **19**
5.1 Conclusion 19
5.2 Future Scope 19

Chapter 6: REFERENCES **20**
News articles: 20
Google Cloud Platform: 20
Natural Language Processing: 20

Appendix 1: 21

CHAPTER 1: INTRODUCTION

1.1 Objective

To automate the tracking and reporting of potholes in addition to giving stakeholders a common dashboard.

1.2 Motivation

We currently live in a world where fixing potholes is done in an inefficient manner. To fix the potholes, the government must go through the process of manual screening of all the roads through appointed personnel. There is an app 'Spothole. Pothole Tracking App' which was built with an aim to track potholes by the public uploading images through their phones. But it does not validate the image uploaded. Also, it lacked features and functionality. The whole process is manually handled by an admin and a group of personnel who must constantly monitor every step is even more tedious. Hence, we decided to come up with a solution to solve these problems with an end to end system.

1.3 Problem Statement

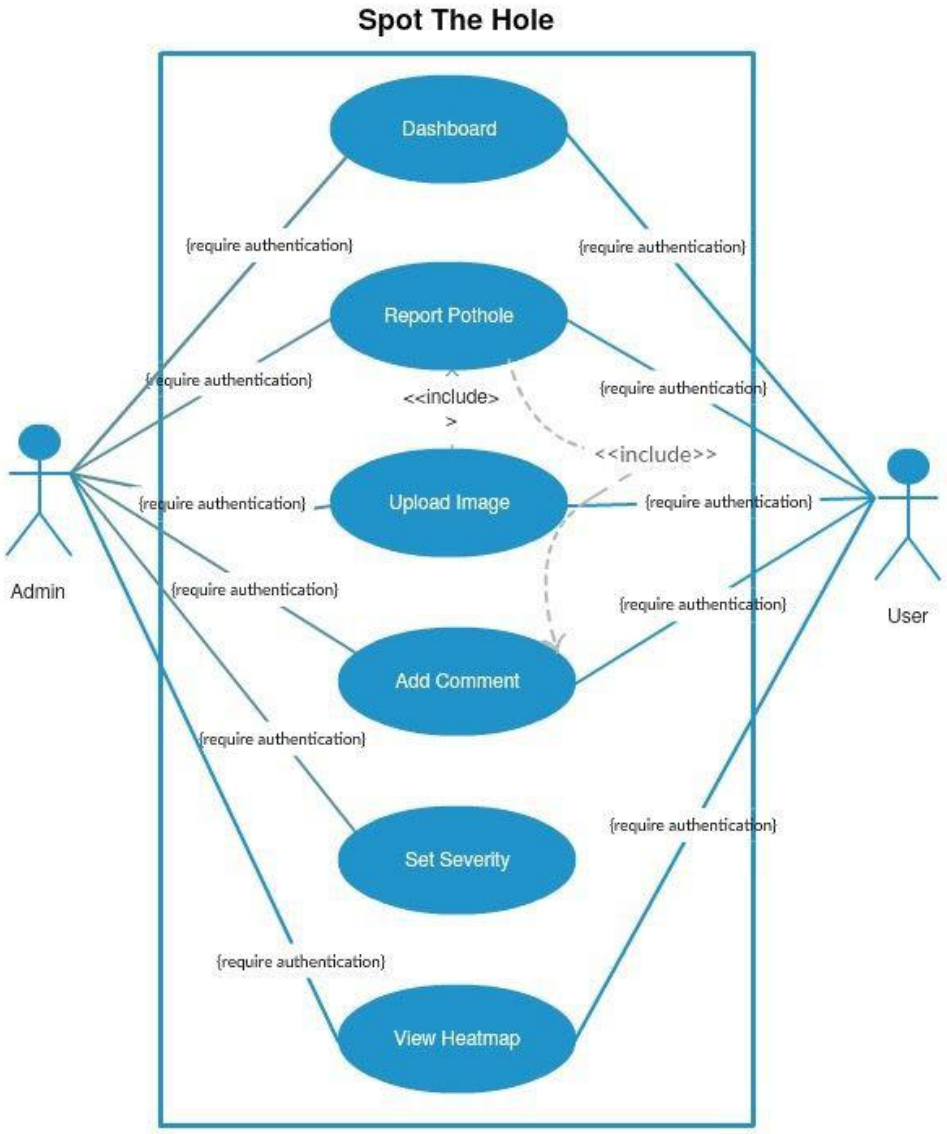
To build a web-based dynamic reporting system which will facilitate the flow of information among all concerned stakeholders. This app helps to solve the challenges faced by all ends by completely automating the processes of analyzing, classifying, and tracking of the reports received from the user.

1.4 Proposed Solution

We decided to build a Progressive Web Application (PWA) which encompasses of Machine Learning (ML), Natural Language Processing (NLP) and Computer Vision (CV) which will be used to analyze all the data generated, to address the issues submitted by the users before the admin prioritizing them based on the severity. Big Data Analytics (BDA) can be used to generate informative reports in the long run. Cloud Computing (CC) will be used to host the entire application

and will also implement Continuous Testing (CI), Continuous Development (CD) and Continuous Integration (CI) in the form of Development Operations (DevOps).

CHAPTER 2: USE CASE DIAGRAM AND DESCRIPTION



Description:

- It starts with the user/admin authentication using the Google Open ID platform.
- After that, the user/admin is greeted with the dashboard where their previously reported pothole cases are shown.
- The user/admin can report the case in the report pothole page.
- The first step to report a pothole is to upload the picture of it.
- The image uploaded will get validated by the ML-based model which we have built and trained using Google's Auto ML platform.
- The user/admin then has to add the description of the pothole.
- The inputted description then gets analyzed for negativity and given a rating accordingly by using VADER analysis.
- The admin can set the severity score of the pothole reported by the users.
- Both the user/admin can view the heatmap based on the severity score set by the admin.

CHAPTER 3: IMPLEMENTATION

3.1 Overview of the Implementation Phase

We have used AutoML provided by Google Cloud Platform to train our model. We collected pictures of potholes and non-potholes to train the model. This is for the validation of the picture taken by the user before he/she uploads it in our system. The location from which the pothole picture has been taken will also be stored to represent the heatmap in the later stages.

Along with the image, he/she can also tag a comment indicating how problematic the pothole is. After the user fills the comments it is analyzed, and a sentiment score is given to depicting the severity. It was accomplished using Valence Aware Dictionary and Sentiment Reasoner (VADER). Based on the sentiment score and the number of potholes reported from a location, a heat map will be plotted.

Low-Level Diagrams:

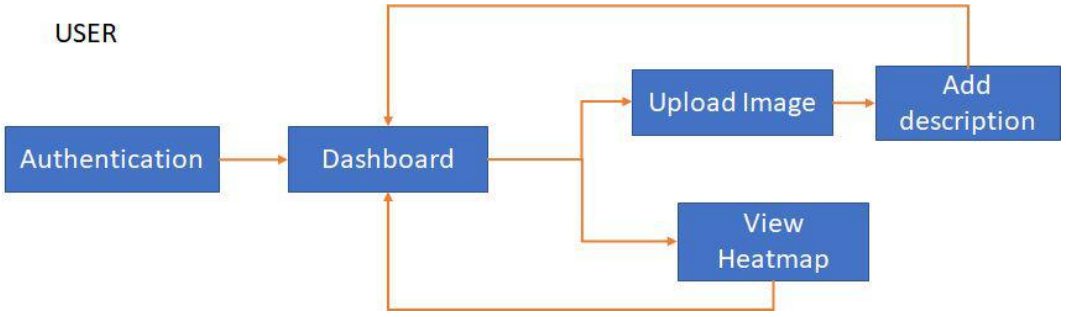


Fig 1(a): Low-Level Diagram for the User side implementation

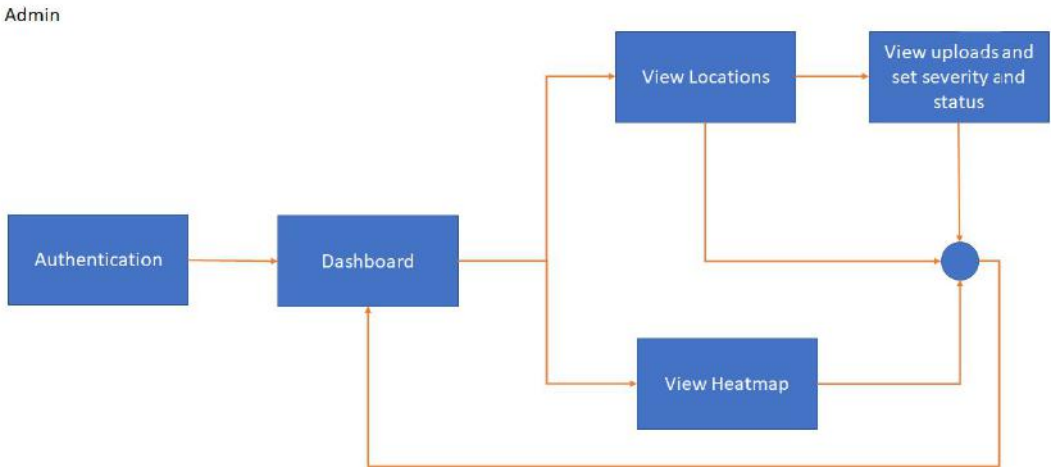


Fig 1(b): Low-Level Diagram for the Admin side implementation

High-Level Diagrams:

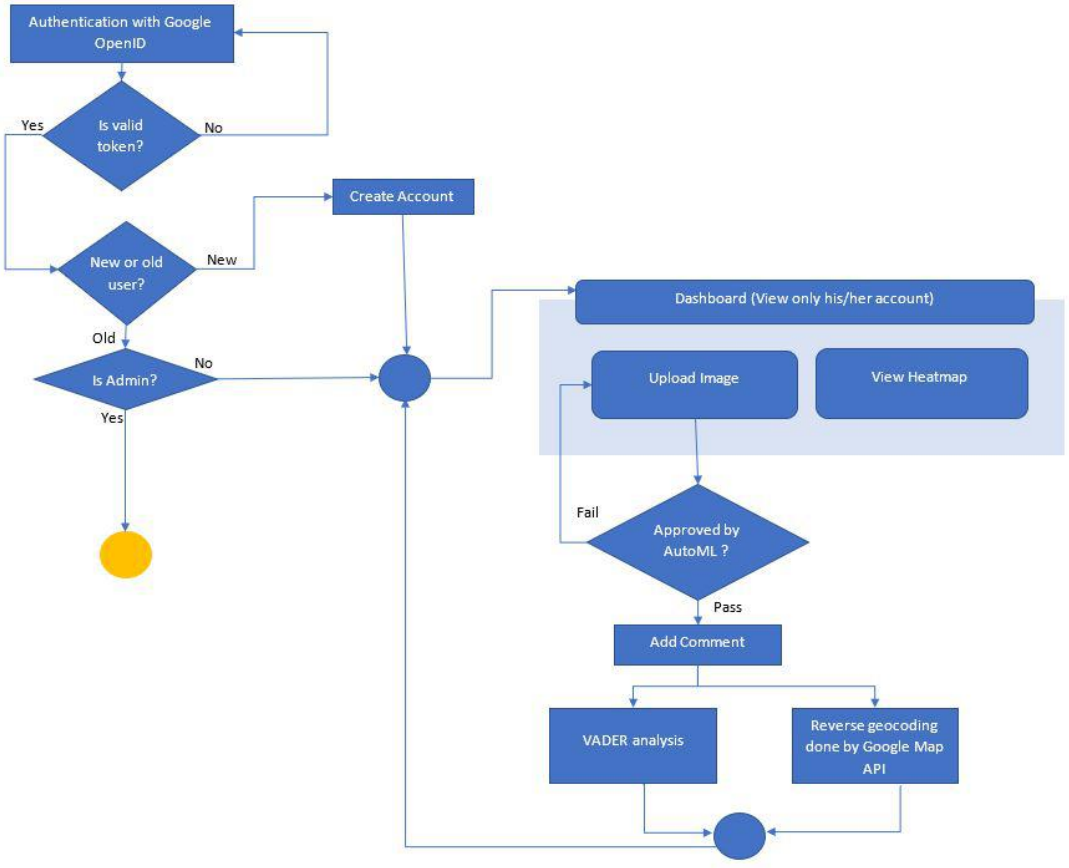


Fig 2(a): High-Level Diagram for the User side implementation

ADMIN

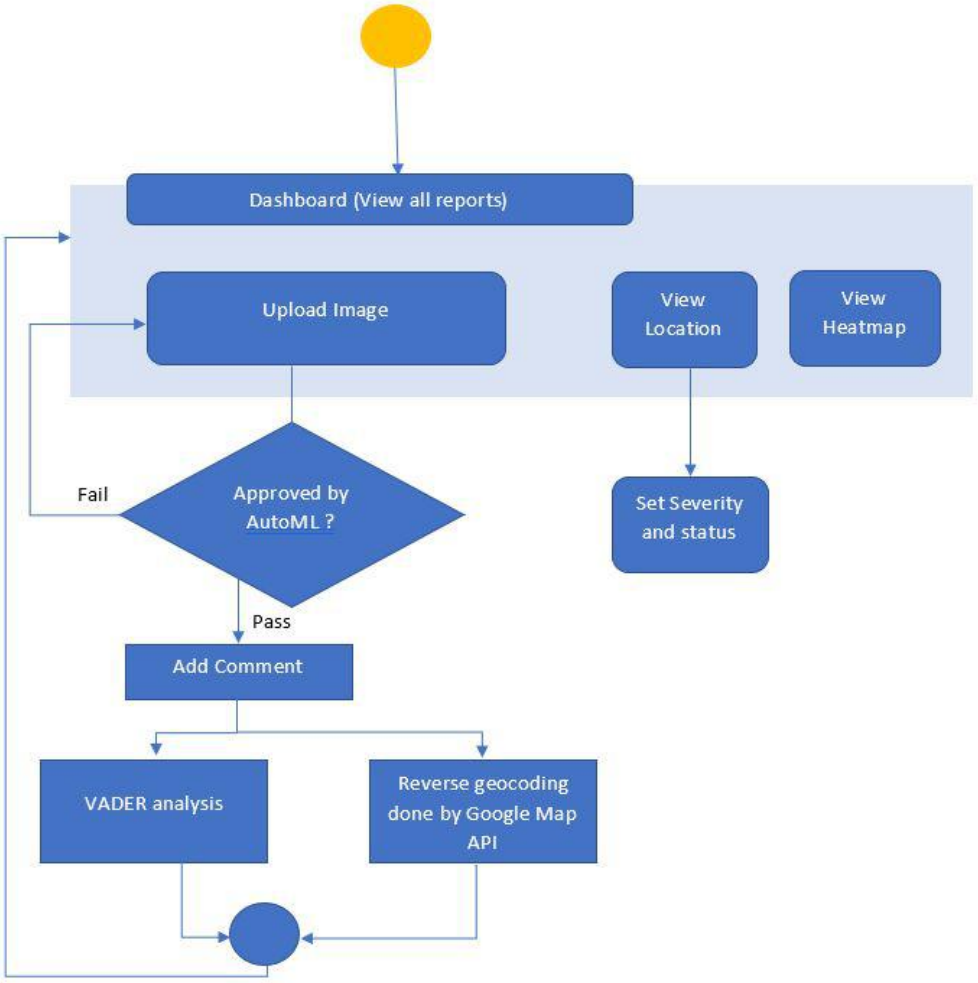


Fig 2(b): High-Level Diagram for the Admin side implementation

3.1 Screen Snippets of the Application

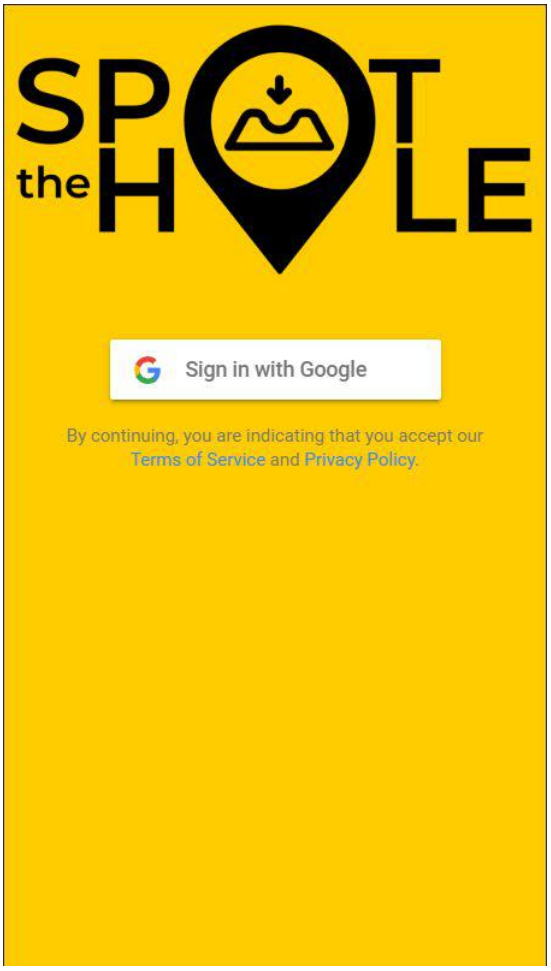


Fig 3: Landing page of the Spot the Hole Application which showcases the App Logo and signs in methods. Signing in with an existing Google account is preferred. It also contains links to our 'Terms of Service' and 'Privacy Policy' pages.

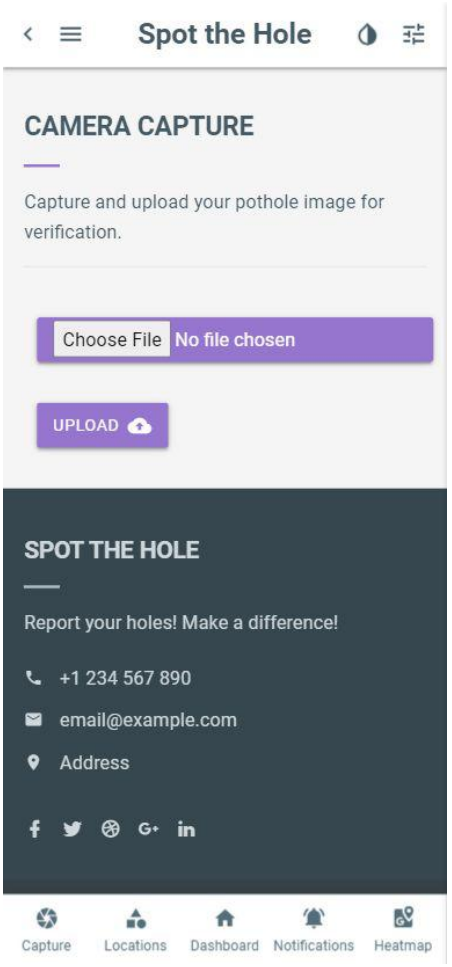


Fig 4: The Capture tab is where the user can upload an image of the pothole he/she wishes to report which will be verified by the ML model in the backend for authenticity and then the user will be taken to a new page to add in the description for the same.

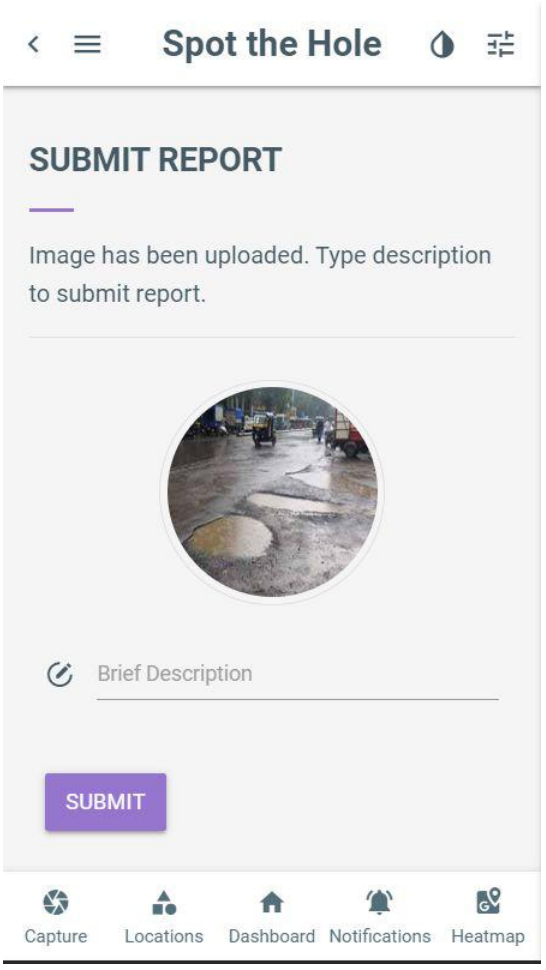


Fig 5: Here the user can describe the details regarding the pothole that has been captured on the previous page. The description given is later analysed using the Natural Language Toolkit to analyse the human emotion conveyed and negativity score will be marked accordingly.

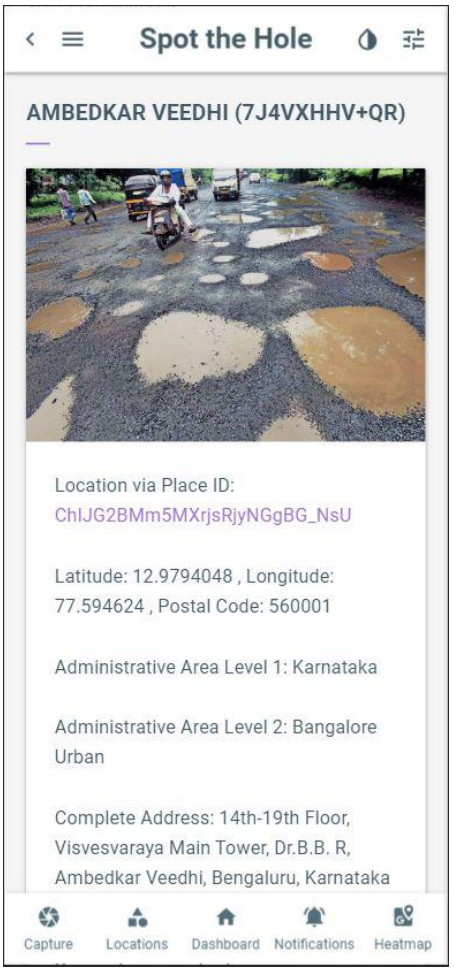


Fig 6: The User Dashboard page is the page we land on once a user successfully logs in. The user gets the following tabs to navigate on his/her dashboard: Report tab- User can click a picture of a new pothole and add a description to it; View Reports tab- To get a list of all the potholes reported by him/her; Notifications tab- Get notified about the status of the potholes reported; Heatmap-To view a heatmap of all the potholes reported by different users on Google Maps.

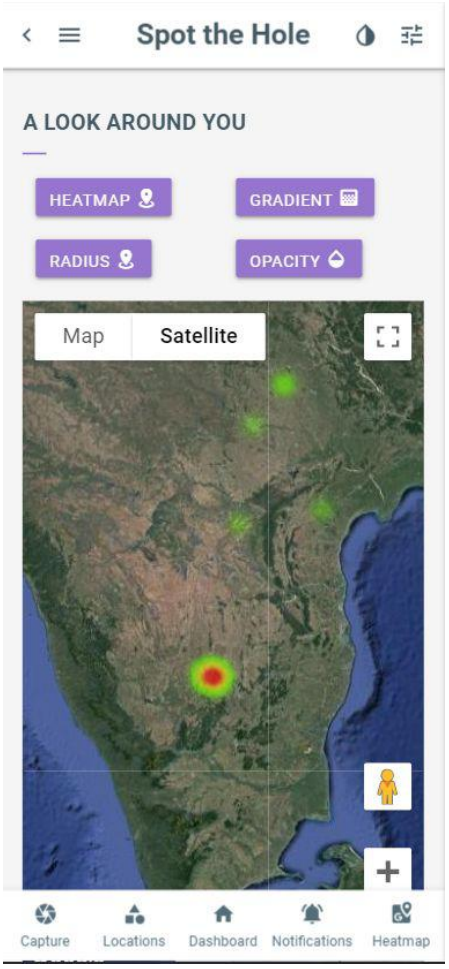


Fig 7: The heatmap tab wherein the user can view the areas in which potholes have been reported. The data on the map is represented by varying colours based on the density of the reports received for the respective areas. The colour gradient, radius and the opacity of the hotspots can be customised according to the user's needs.

Fig 8: The admin dashboard wherein the administrator gets a list of the potholes reported by all the users listed according to the location. He/she can view the individual reports by selecting the location id and then take the necessary steps. The admin can also view the heatmap.

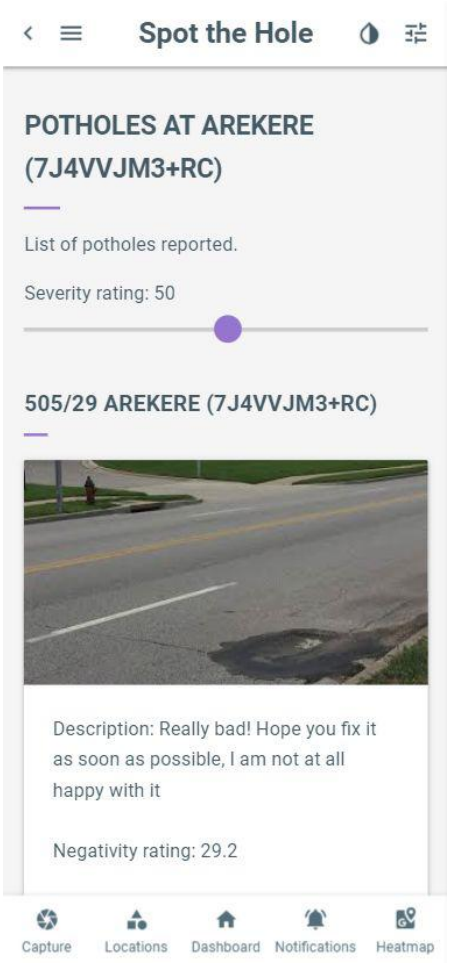


Fig 9: This is the page admin lands on once he/she selects a particular location in the admin dashboard. Here, he/she can view the details of the pothole reported. The image, description and severity are seen. The admin can set the severity as well.

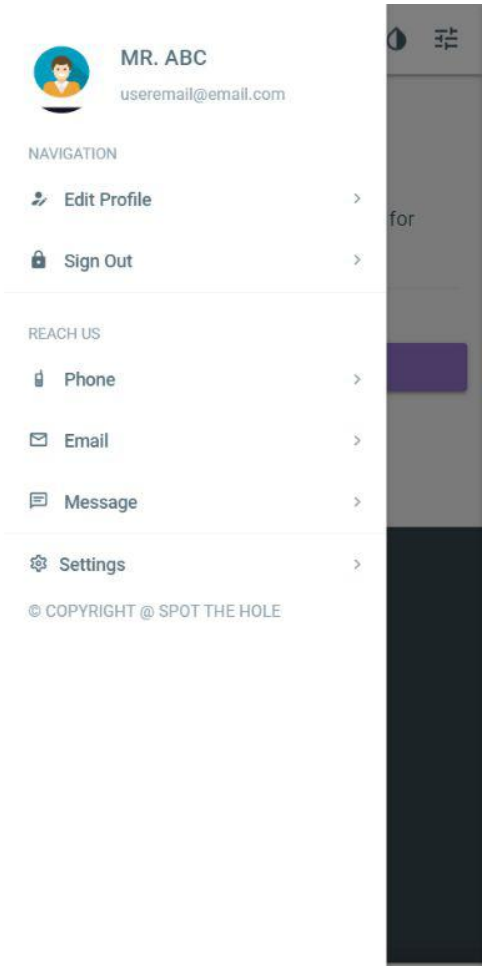


Fig 10: The ☰ button on the top left corner will lead the user into the navigation bar, wherein he/she gets the Edit Profile and Sign Out options.

CHAPTER 4: RESULTS

The application has been developed successfully with the intended features. The AutoML model built on the Google Cloud Platform evaluates the picture captured by the user and evaluates the authenticity of whether it is actually a pothole picture with 99% accuracy. The app has undergone several iterations of testing by us and it is safe to say that it is secure and bugfree. The domain on which it is hosted is secure (HTTPS) and so are the cookies used. The application is reactive and the page elements adapt themselves if and when the window size changes (or the mobile screen sizes vary).

The user hierarchy is well defined, the user and admin roles segregated has resulted in easy tracking of the potholes identified by users. The admin can evaluate and custom set severities.

The app is integrated with various APIs such as Google Maps API, Compute Engine API, Cloud Functions API as well as other integrated npm packages. There is no conflict observed while these above-mentioned modules simultaneously run and therefore the application runs without a glitch.

Any user can successfully track all the potholes easily by simply looking at the heat map.

CHAPTER 5: CONCLUSION AND FUTURE SCOPE

5.1 Conclusion

We have developed a PWA which helps users to report potholes for the government to take necessary actions. Once the user uploads an image, it is then validated by the machine learning model we have built and trained after which the user can enter a description of the pothole. The description is then analyzed for negativity and given a rating accordingly. The government bodies can then view the uploaded images and take necessary actions while also updating status and setting severity. Thus, we have come up with an end to end solution for a real-life problem.

5.2 Future Scope

This is a fully functional app that helps to solve the challenges faced by completely automating the processes of sorting, classifying, and prioritizing of the input received from the user. We hope the citizens and the government can make the best use of this app. With this app, citizens can report potholes in an easier manner and the government can take the necessary actions. The app provides a platform for keeping track of complaints.

Chapter 6: REFERENCES

News articles:

- https://m.timesofindia.com/city/goa/potholes-unavoidable-as-state-receives-heavy-rainfall-sudin/amp_articleshow/65663142.cms
- http://englishnews.thegoan.net/M/#!/detail.html?id=54271&secid=1
- https://www.goa365.tv/issues/B/those-responsible-for-potholes-be-ready-to-face-fury-potholes-free-goa/08146.html
- https://www.heraldgoa.in/Edit/Opinions/Goa%E2%80%99s-pothole-syndrome/151730

Google Cloud Platform:

- https://cloud.google.com/vision/automl/doc
- https://developers.google.com/maps/documentation/geocoding/start
- https://developers.google.com/maps/documentation/javascript/tutorial
- https://firebase.google.com/
- https://cloud.google.com/compute#section-5

Natural Language Processing:

- https://www.nltk.org/_modules/nltk/sentiment/vader.html

Appendix 1:

- APIs Used:

Name	↓ Requests	Errors (%)	Latency, median (ms)	Latency, 95% (ms)
Compute Engine API	431,706	0	235	490
Identity Toolkit API	1,052	1	137	307
Cloud Functions API	243	0	214	693
Cloud AutoML API	85	15	632	14,680
Geocoding API	63	31	50	243
Maps JavaScript API	23	4	31	65
Cloud Runtime Configuration API	7	0	98	432
Firebase Management API	4	0	262	996

Fig 11:Above image is the list of all the APIs used in our project

- Dataset Sample

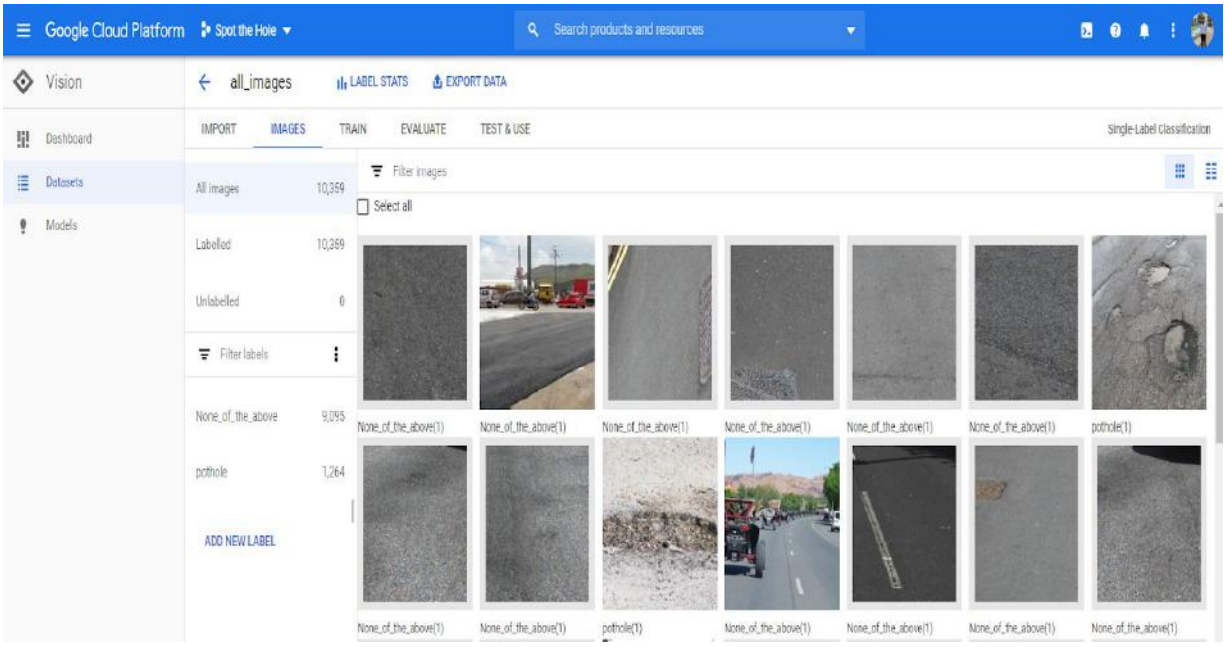


Fig 12: Above shown images are a few sample images taken to train our model. A total of 10,359 images were a part of the training dataset.

- Machine Learning model

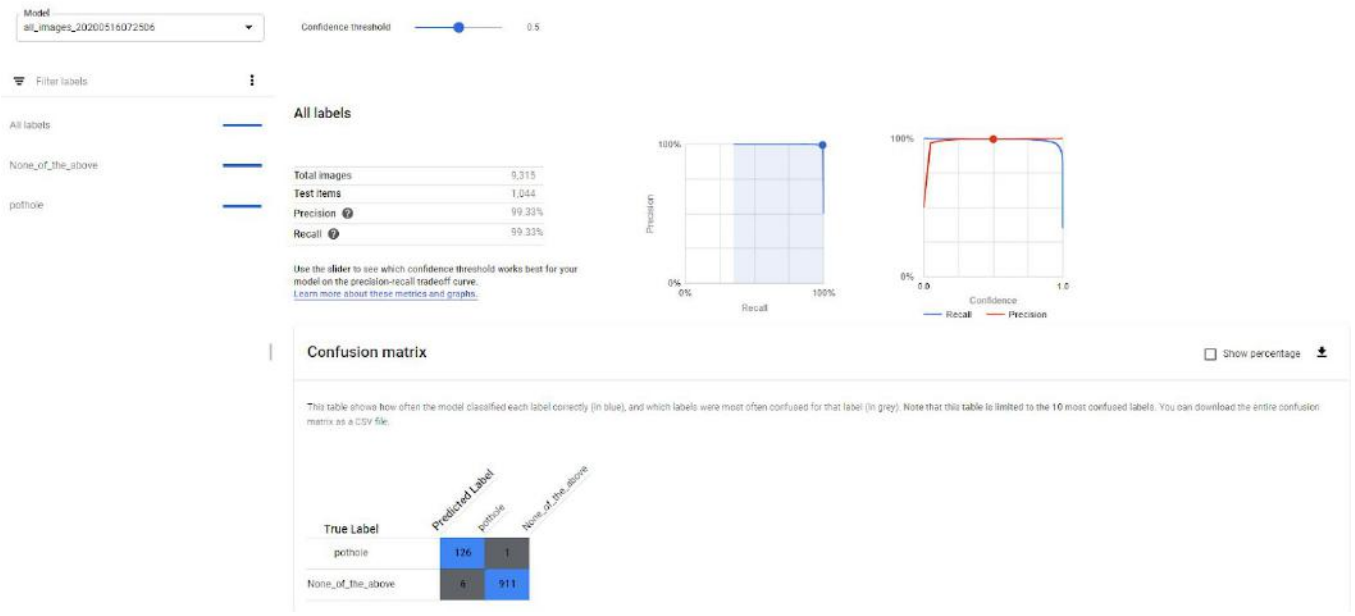


Fig 13: The ML model that is built on the Google Cloud Platform using Google's AutoML. This is the model that verifies the authenticity of the pothole image captured by the user.

- Firestore Database

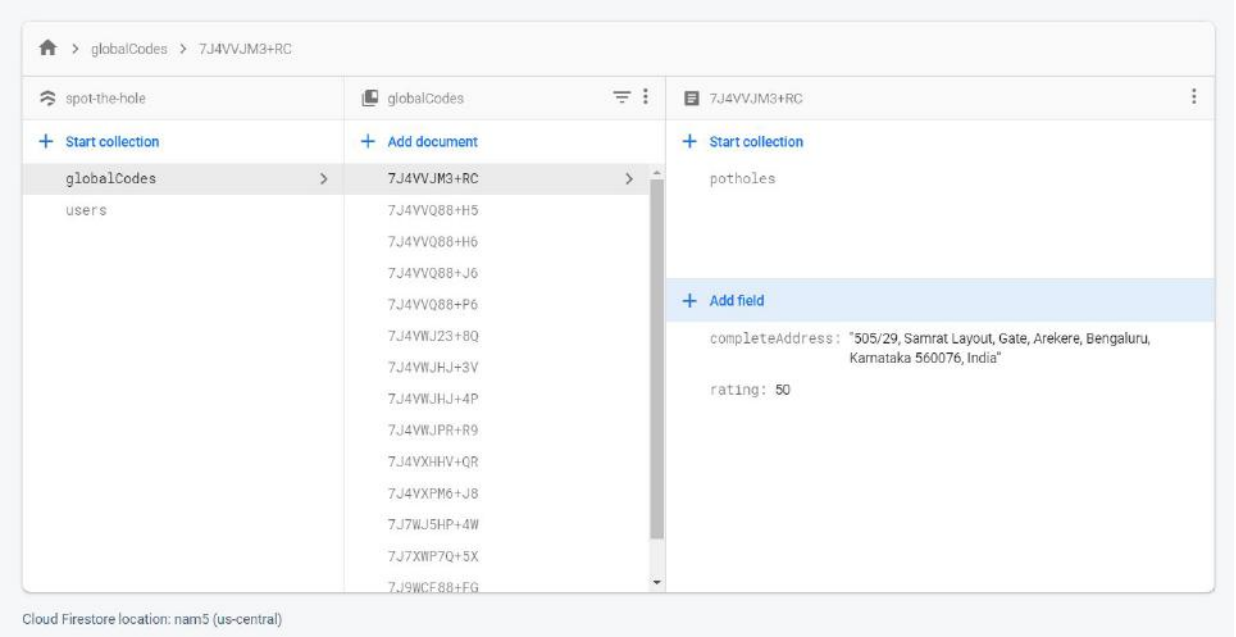


Fig 14: The database of the pothole data reported by the users as stored in Google Firestore in the form of collections. It shows all reported potholes, including their location and severity rating. Using the global codes one can search the location directly on the internet by simply typing that code.

- User Authentication for the application as viewed in the Firebase Console.

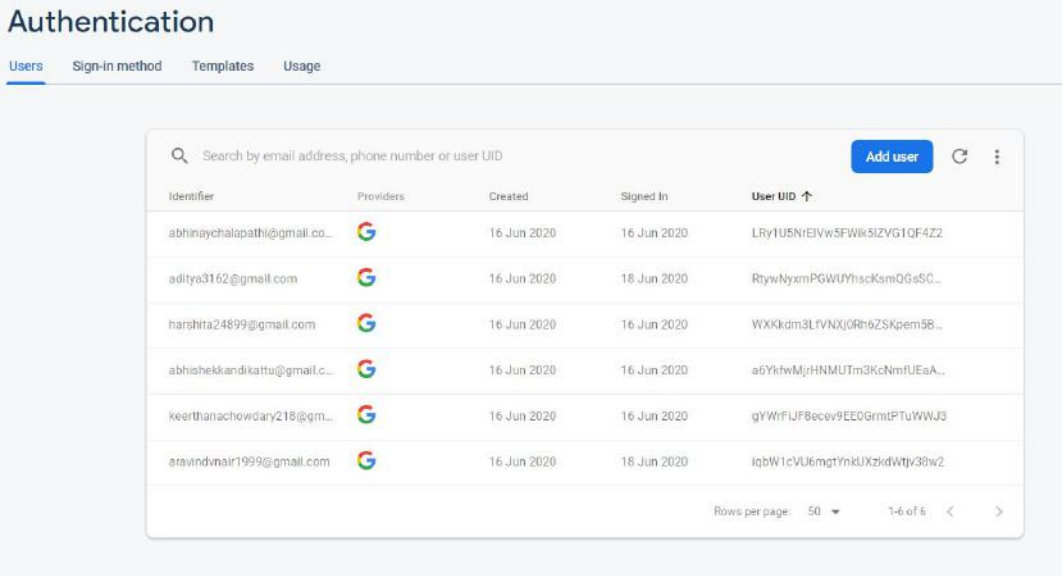


Fig 15: Above image, taken from firebase, shows all the authenticated users of our application.