Cairo University
Faculty Of Computers and Artificial Intelligence
Department of Computer Sciences

My Way Project

Supervised by

Dr. Basheer Abdelfattah

TA. Hassan Mourad

Implemented by

20196009	*Mahmoud Mustafa El Sayed*
20196068	*Mohammed Khaled Sabrah*
20196052	*Mariam Mohamed Abdelraouf*
20196065	*Ahmed Ashraf Ahmed*
20196019	*Hamdi Hossam Hamdi*

Graduation Project
Academic Year 2022-2023
Final Documentation

Abstract

There are many problems that face transportation movement such as Driving Behavior and Attitude, Traffic Accidents, Road and Environment Incidents.

Our main problem that we chose is Road and Environment Incidents. We chose this problem as people ignore it even though it is one of the most important transportation issues.

Road and Environment Incidents include ground crashes, pipeline explosions, falling rocks, tree crashes, rains and floods, fire and explosions.

Maintaining and fixing these problems as early as possible reduces the possibility of accidents.

We came up with this idea to help the government to fix these issues as early as possible, and users to be up to date with road issues to avoid being in danger.

We will develop a mobile application to make users report incidents easily and fast using their mobile camera which images captured will be processed by our AI model. On the other hand, we have a web application to monitor reports and help admins to manage them.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

Abbreviation	Meaning
AI	Artificial Intelligence
API	Application Programming Interface
CSRF	Cross-Site Request Forgery
ERD	Entity Relationship Diagram
GUI	Graphical User Interface
Gov	Government
iOS	iPhone Operating System
ML	Machine Learning
MVC	Model-View-Controller
ORM	Object-Relational Mapping
SPA	Single Page Application
SQL	Structured Query Language
UI	User Interface

Chapter 1: Introduction

Transportation is one of the primary causes of accidents and environmental incidents, especially those which involve road and highway infrastructure. Road crashes are one of the most common causes of accidents, with approximately 2.5 million people injured or killed each year worldwide. These incidents can be ground crashes, falling rocks, tree crashes, and more.

Government cannot be knowledgeable about everything happening in the country and our surroundings, but residents can. With our application, residents can help our government be aware of environmental incidents that they cannot deal with by reporting them. This will also help authorities to have better planning to prevent such incidents from happening. The app will connect the users with the authorities and help resolve the issues as early as possible, thereby improving the overall safety of our citizens.

The reporting process is made to make users access the camera in the quickest way possible. They can easily report the incidents by just raising the camera. Upon raising the camera, the residents will be directed to the incident report form which will contain basic details of the incident such as description, and the location will be taken automatically. After submitting the report, our AI (Artificial Intelligence) model will detect the environmental incident according to the captured photo and will pass the information to the local authorities for further action.

Our dataset is classified based on 7 classes which are Normal/no incident (class 0), fallen trees severity 1 (class 1), fallen trees severity 2 (class 2), fire and explosions severity 1 (class 3), floodings severity 1 (class 4), potholes severity 1 (class 5), and potholes severity 2 (class 6).

1.1 Motivation

1. Our main first motivation is to increase road safety: Early reporting of incidents will help responders to act quickly which will not only help solving the issue in less time but also it will prevent further accidents/incidents.
2. Increasing road efficiency: Road incidents will naturally lead to more traffic congestion which will decrease the road efficiency so by detecting and reporting those incidents quickly we will improve the efficiency of the road and the efficiency of the emergency response services.
3. Community engagement: collaboration in communities is very important and our goal is to encourage individuals and communities to be more engaged in maintaining road safety and it also enhances the trust between the community and the government.
4. Improve service response time: In many cases the government isn't notified of the incidents on the road which leads to decreased safety and delay in response time. That's why we aim to create a system that can quickly detect and report incidents on the road in real-time to minimize the damage and improve response time.

1.2 Problem definition

Road incidents, such as potholes, fires, traffic disruptions. A significant problem that can have serious consequences for public safety, infrastructure, and property. These incidents can lead to injuries, accidental deaths, and property damage, as well as significant traffic congestion and disruption.

1.3 Project Objective (suggested solution)

Our objective is to build a real-time detection and reporting system that can quickly notify community members and emergency responders of road issues, such as incidents and dangers.

1.4 Gantt chart and Time Plan:

Time plan:

Task	Task Title	Description	Task status (completed/expected in time)
1	Determine project idea	This task is the responsibility of every team member	Completed on January 28, 2023
2	Find similar ideas	This task was very important for us to gather information and to get an idea of different features	Completed on February 2, 2023
3	Writing the abstract	We have to understand, write more information about the project and write the motivation to solve the problem	Completed on February 2, 2023
4	Background and the problem definition	Determine the main techniques and the scope of main application	Completed on February 3, 2023
5	System analysis	Trying to gather information about the functional, non-functional requirements and the stakeholders	Completed on February 5, 2023
6	Database analysis	Choose the perfect database (SQL/NoSQL) and trying to determine the hierarchy of the tables	Completed on February 6, 2023
7	System design	Starting designing the system diagrams	Completed on February 10, 2023
8	UML diagrams revision	We have to revise the diagrams	Completed on February 11, 2023
9	Frontend Design	Designing our application frontend with Figma	Completed on February 16, 2023
10	Mid-year presentation	Presenting our project features and a demo of it	Finished February 18, 2023

11	Find a dataset	find a suitable dataset for our ML model with a very good accuracy.	Completed on May 2 ,2023.
12	Study web development	We have to study the front and the backend to be able to do the web application	Completed on May 17,2023.
13	Study flutter	We have to study flutter framework to be able to do the mobile application	Completed on May 20,2023.
14	Web app implementation	Implement our website with React framework.	Completed on June 27,2023.
15	RESTful APIs implementation	Implement our RESTful APIs with Laravel and integrate it with our frontend	Completed on May 18,2023.
16	Study AI	At first, we have to know how to detect the objects	Completed on May 14,203.
17	Mobile app implementation	Implement our mobile app with Flutter and Dart	Completed on June 29,2023.
18	Testing 1	Testing our application using use case testing	Completed on May 5 ,2013.
19	AI feature	Implementing the photo capturing feature and integrate our AI with it	Completed on June 14,2023.
20	Testing 2	Testing our application using equivalence partitioning and boundary value analysis	Completed on June 30,2023.
21	Deployment	Deploy our project on cloud	Completed on July 3,2023.
22	Final presentation	Present our project with its all fully functionable features	Expected to be July 13, 2023.

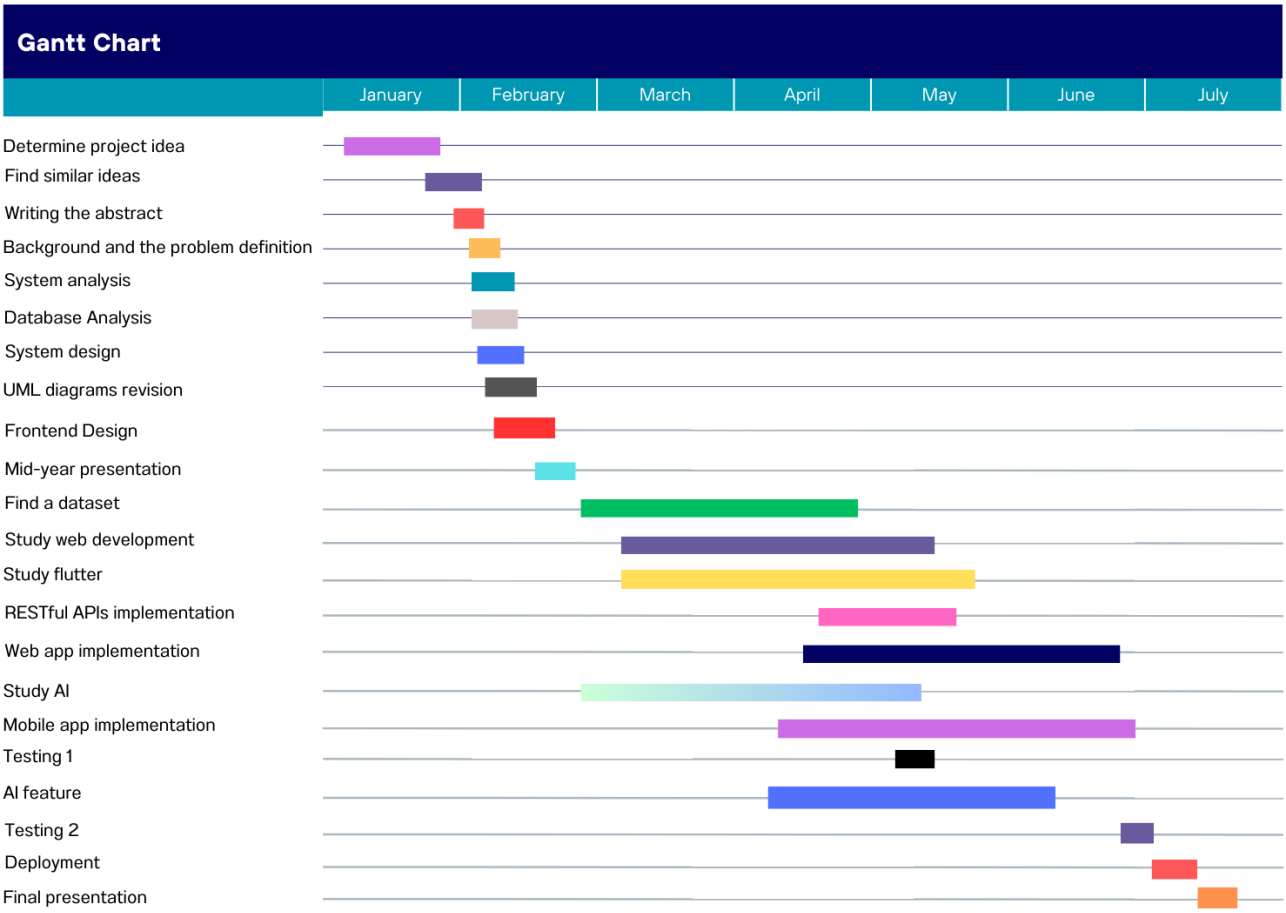
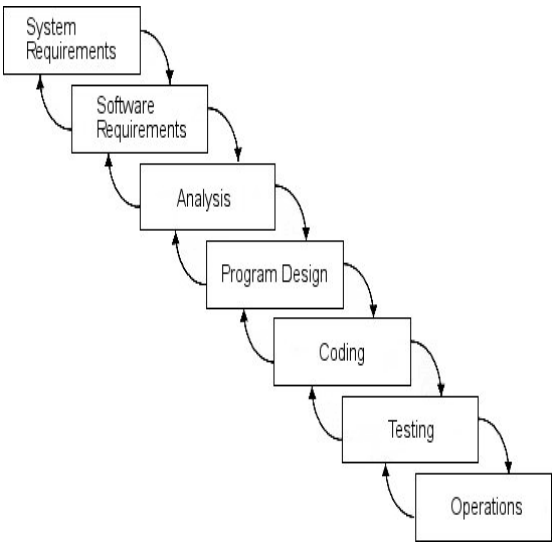


Figure 1 Gantt Chart

1.5 Project development methodology:



We are using the waterfall methodology in our project, which is a linear and sequential approach to software development. This methodology requires that each phase of the development process be completed before moving on to the next phase.

Figure 2 Project development methodology

1.6 Technologies:

- Database: MySQL
- Frontend design: Figma.
- Web development Frontend: HTML, CSS, JavaScript with React framework.
- Backend development: Laravel framework.
- Mobile development: Flutter and Dart.
- AI & ML (Machine learning): Python, Tensor flow, Google Collab, Flask, Keras API(Application Programming Interface), NumPy and Scikit-learn libraries and InceptionV3 model.
- Hardware: The user's mobile phone.

1.7 Report Organization (Summary of the rest of the report)

Next Chapter is chapter 2 (Related work) this chapter will provide an overview of the existing related work on the topic of the project. It will cover the existing projects that are similar to ours and how our project will be different from them.

Then we will have Chapter 3 (System Analysis) this chapter will provide a detailed analysis of the system, including the project specifications, stakeholder requirements, and use-case diagrams. It will also cover the functional and non-functional requirements of the system

Next will be Chapter 4 (System design) This chapter will provide a detailed overview of the system design, starting with the system component diagram and moving on to the system class diagram, system sequence diagrams, project ERD(Entity Relationship Diagram), and GUI(Graphical user Interface) design using the Figma framework. The system component diagram will provide an overview of the system's major components and how they interact with each other. The system class diagram will provide a detailed description of the system's classes and their relationships. The sequence diagrams will provide a detailed description of the system's behavior in various situations. The project ERD, will provide a detailed representation of the system's database schema. Finally, the chapter will present the GUI design of the system using the Figma framework. This design will provide a visual representation of what the system will look like and how users will interact with it.

Next and final chapter is chapter 5 (Implementation and testing) it will contain the actual implementation of our project and some tests using samples of the applied test cases.

Lastly, we will provide the references we used in our graduation project.

Chapter 2: Related work

There are no local applications that focus on this problem, but we found similar application in other countries like:

Applications:

Balady:
A Saudi Arabia application that offers many features, one of these features is to send reports about any incidents.
تطبيق معالجة الحفر:
A Saudi Arabia application that aims to send reports about ground crashes.

Differences:

What sets MyWay apart from other related apps is our AI-powered road incident detection feature that enables fast and reliable detection of incidents on the road. With our highly accurate AI model, we can help save valuable time for government authorities and emergency responders. Our AI model also helps filter out fake incident reports, ensuring that only genuine incidents are reported. Moreover, our app provides a feature that allows users to check for nearby incidents around them, anywhere, anytime. This helps users stay informed and avoid potential traffic delays or hazards. Additionally, we offer a rewards program where users earn points for every non-fake incident report that is detected by our AI. These points can be redeemed by users for various benefits. Our unique AI-powered incident detection feature, combined with our user-friendly interface and rewards program, make MyWay a better choice for users who value accuracy, reliability, and convenience.

Chapter 3: System analysis

3.1.1 Functional Requirements

Mobile Application:

1-User can register using full name, phone number, national id and password.
2-The mobile Application must validate that the national id consists of 14 numbers and no user has registered with this id before.
3-User can login using email and password.
4-User can take a photo through the mobile application.
5-User can send a description, severity and type of the incident with the photo (optional).
6-Incident location is detected using the user's device current location.
7-Each report has a severity from 1 (highest) to 4 (lowest).
8-User has the ability to change the photo he took before submitting the report (if the photo is blurry or not clear).
9-User can view all the reports he sent and track the status of each report.
10-The mobile application provides see nearby incidents from the user location.
11-The mobile application provides help and Q&A section to help users to use the features.
12-The mobile application provides hotlines for emergency services.

Web Application:

1-The website lists all the requested reports and their status like pending, in progress and completed.
2-There are 3 administrative roles owner, manager and admin.
3-The owner can add or delete a manager.
4-The manager can add an admin by using full name, phone number, national id and password.
5-The manager can delete or restrict the admins.
6-The admin can login using email and password.
7-The admin can view the reports for a specific district sorted by the severity.
8-The admin can list all the reports for a specific user.
9-both admin and manager can ban or restrict users.
10-The government user can login email password.
11-The government user can view approved reports and mark it as done.
12-For every five successful reports the user gets a 10-points reward.

3.1.2 Non-functional Requirements

1 - **Availability:** the system must be available all day every day.
2 - **Low latency:** the hosts should be able to receive reports from mobile app users within
3 - seconds at most.
3 - **Usability:** must be easy to use.
4 - **Security:** system must have high security level.
5 - **Performance:** any page must load within at most 2 seconds.
6 - **Compatibility:** the application must be compatible with any device.
7 - **Scalability:** the system must be accessible during the high workload.
8 - **Flexibility:** the system must be flexible to add more functionalities or features.

3.2 Use-case Diagram

This is the use case diagram for our system, and we used it to specify the expected behavior (what) and not the exact method of making it happen (how).

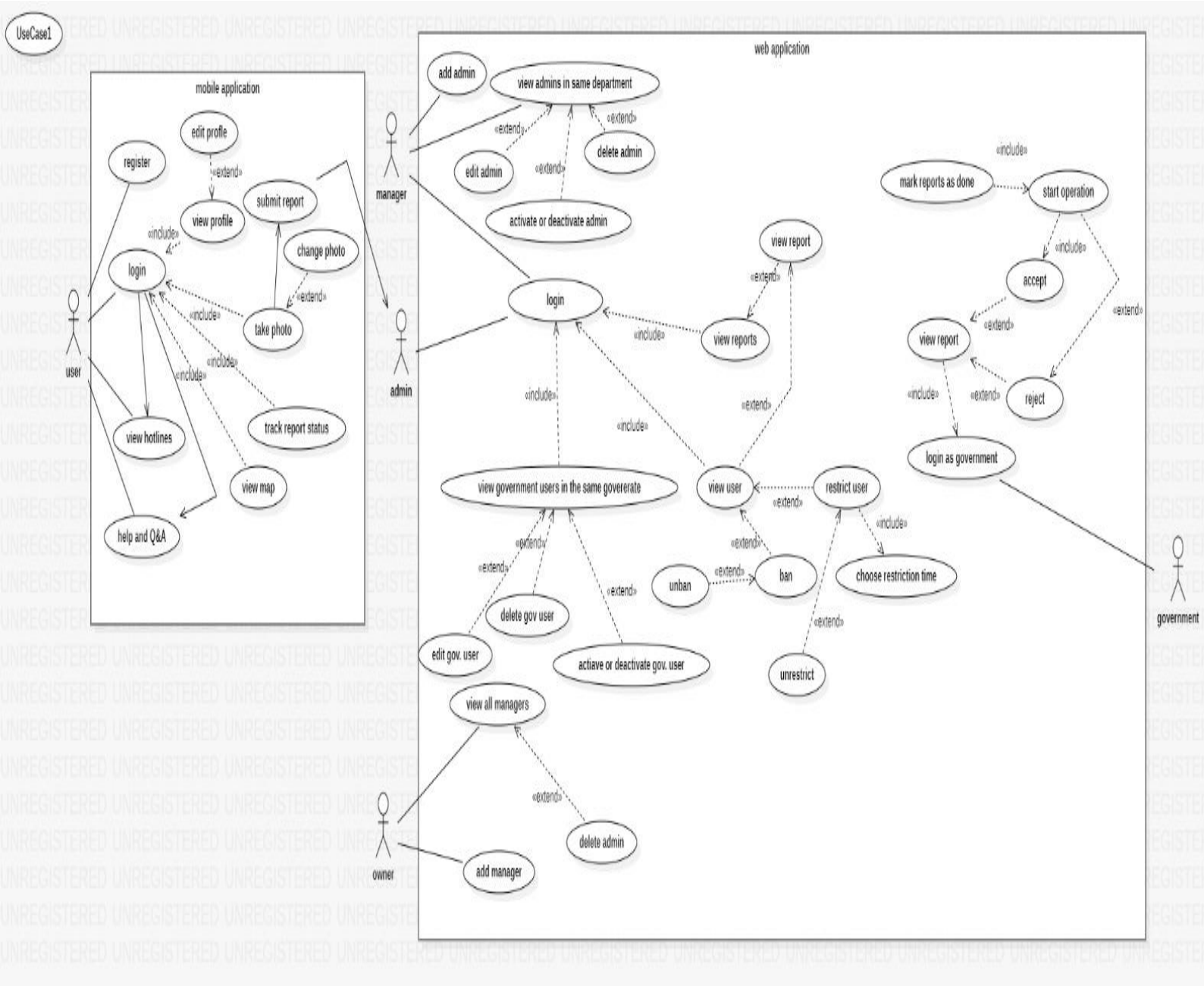


Figure 3 Use-case Diagram

Chapter 4: System design

4.1 System Component Diagram

This is the component diagram for our system and the component diagram used in modelling the physical aspects of object-oriented systems that are used for visualizing, specifying, and documenting component-based systems and also for constructing executable systems through forward and reverse engineering. Component diagrams are essentially class diagrams that focus on a system's components and are often used to model the static implementation view of a system.

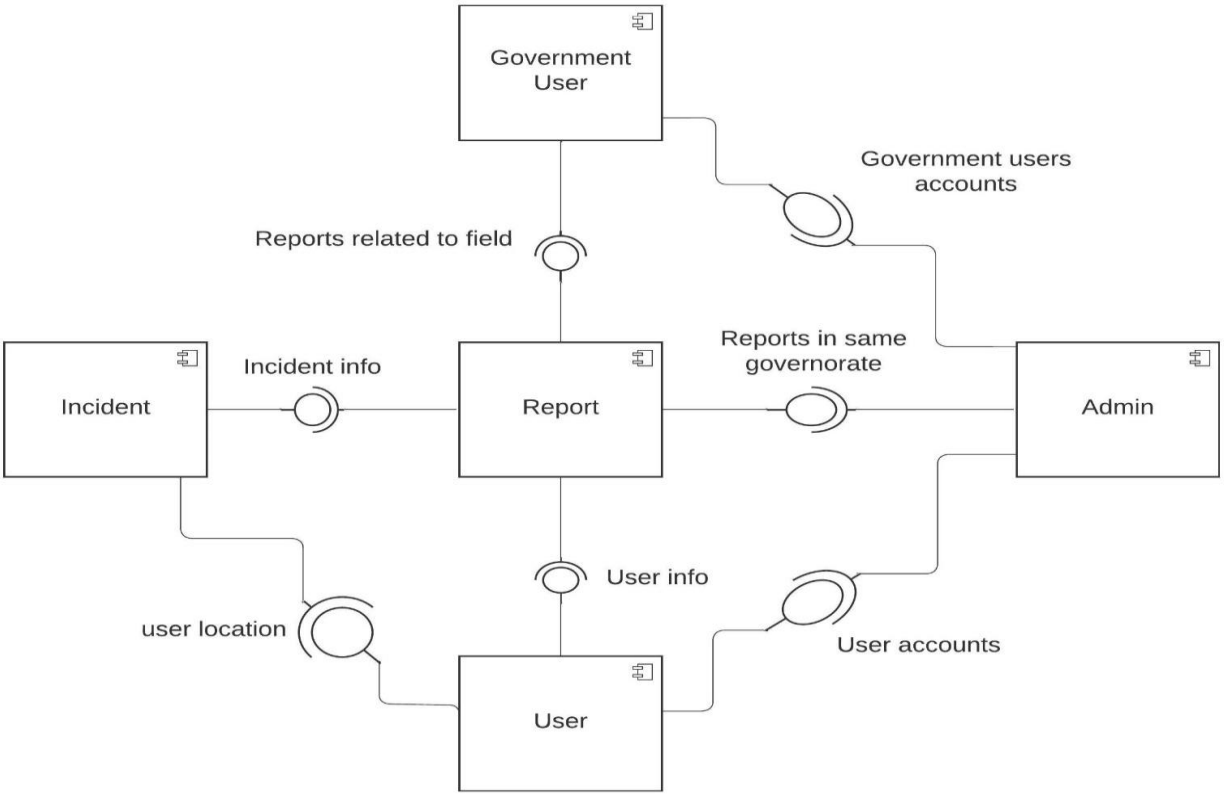


Figure 4 System Component diagram

4.2 Class Diagram

This is the class diagram for our system, and we used it to show the static structure of classifiers in a system. The diagram provides a basic notation for other structure diagrams prescribed by UML, and Business Analysts can use class diagrams to model systems from a business perspective.

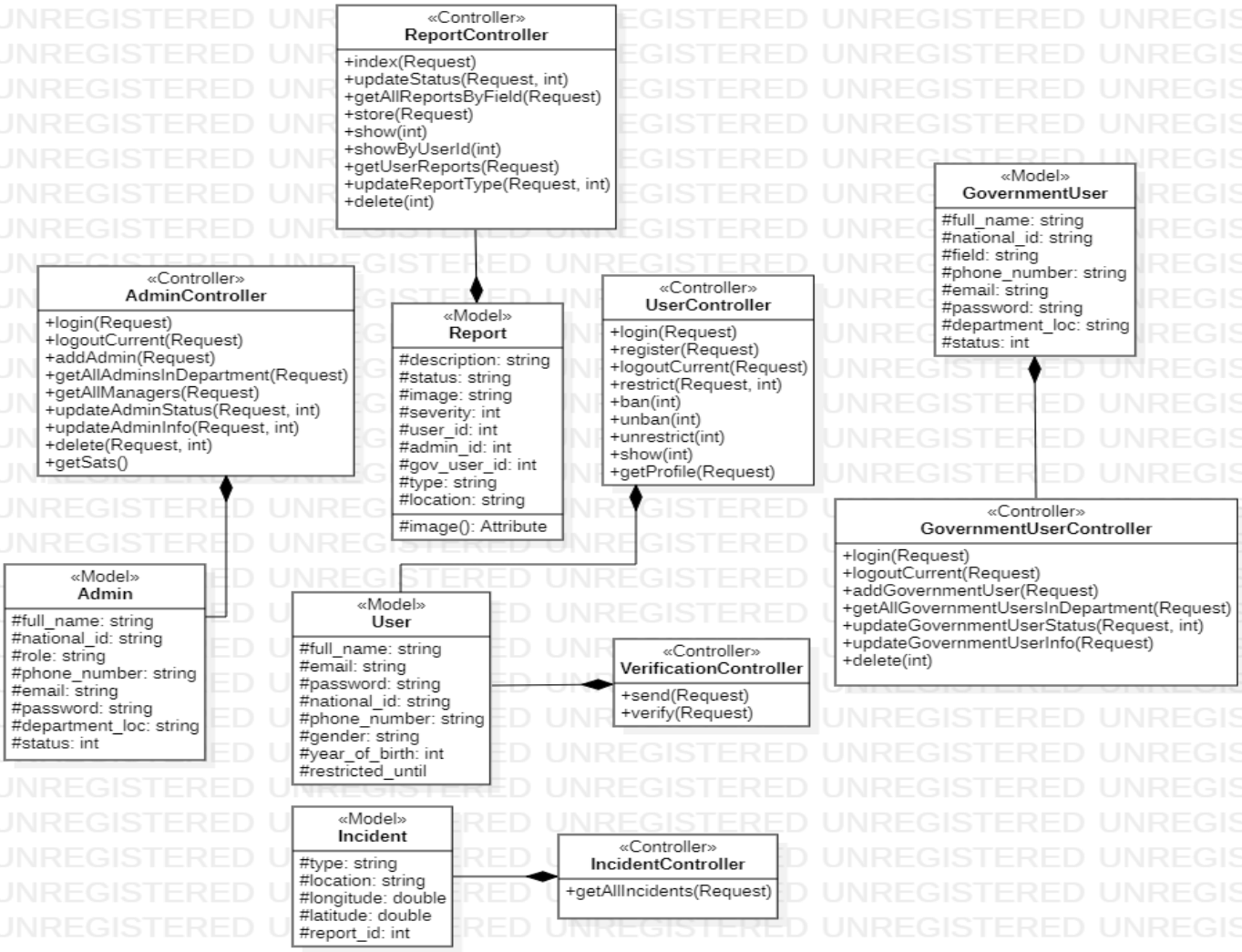


Figure 5 System class diagram

4.3. Sequence Diagram

This is the sequence diagrams for our system, and the sequence diagrams are interaction diagrams that detail how operations are carried out. They capture the interaction between objects in the context of a collaboration.

4.3.1. Send report

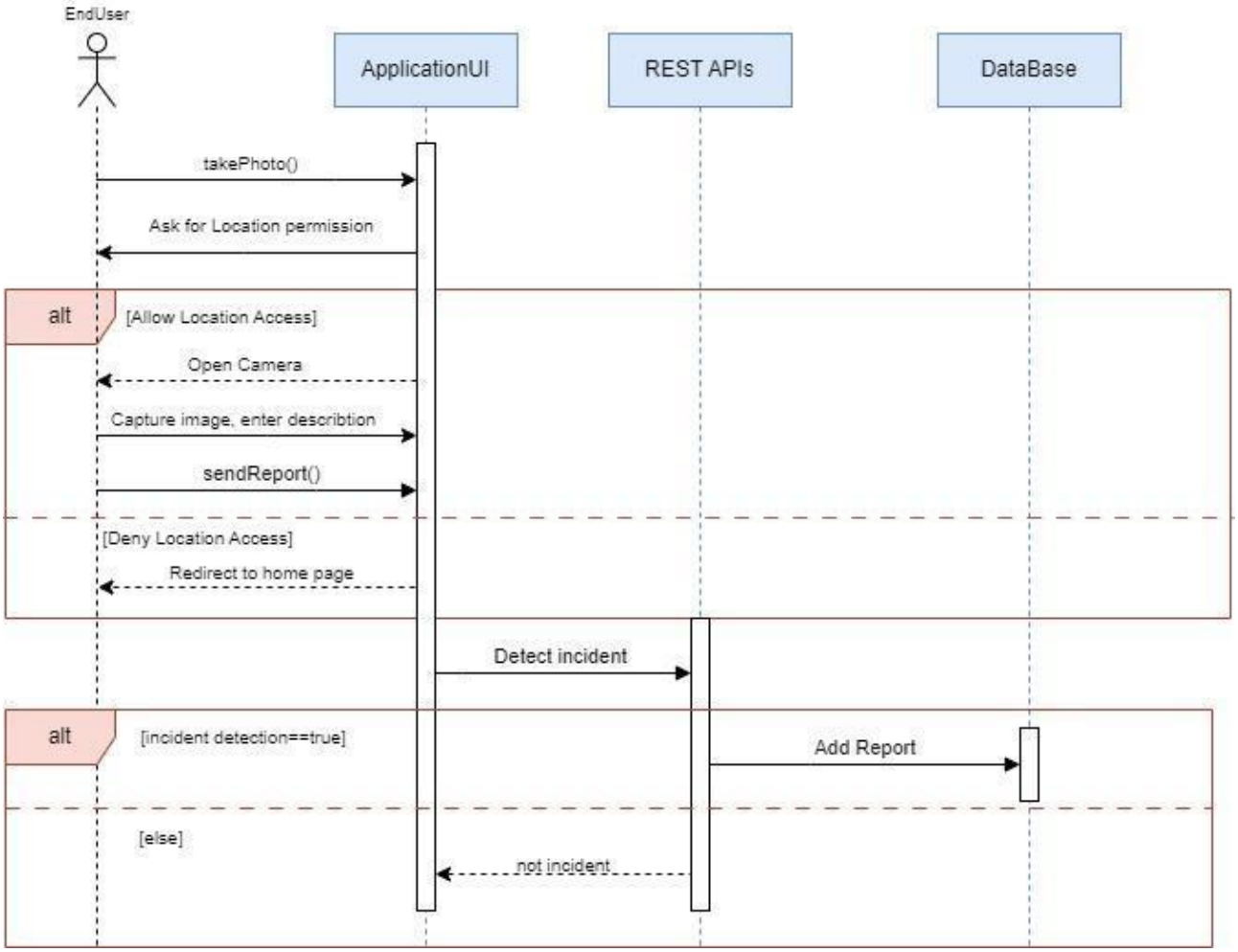


Figure 6 Sequence diagram for send report

4.3.2. Register

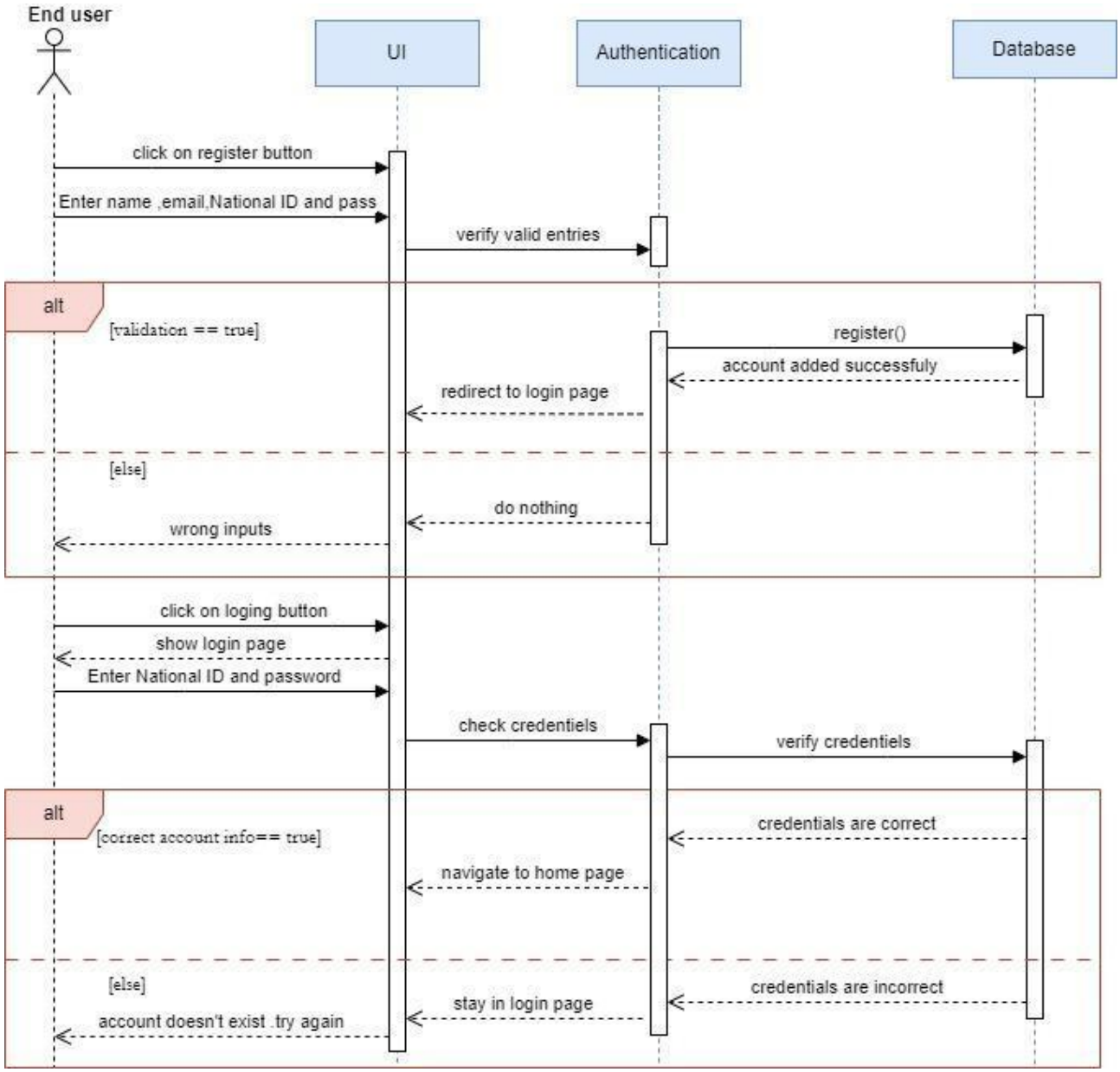


Figure 7 Sequence diagram for register

4.3.3. Restrict user

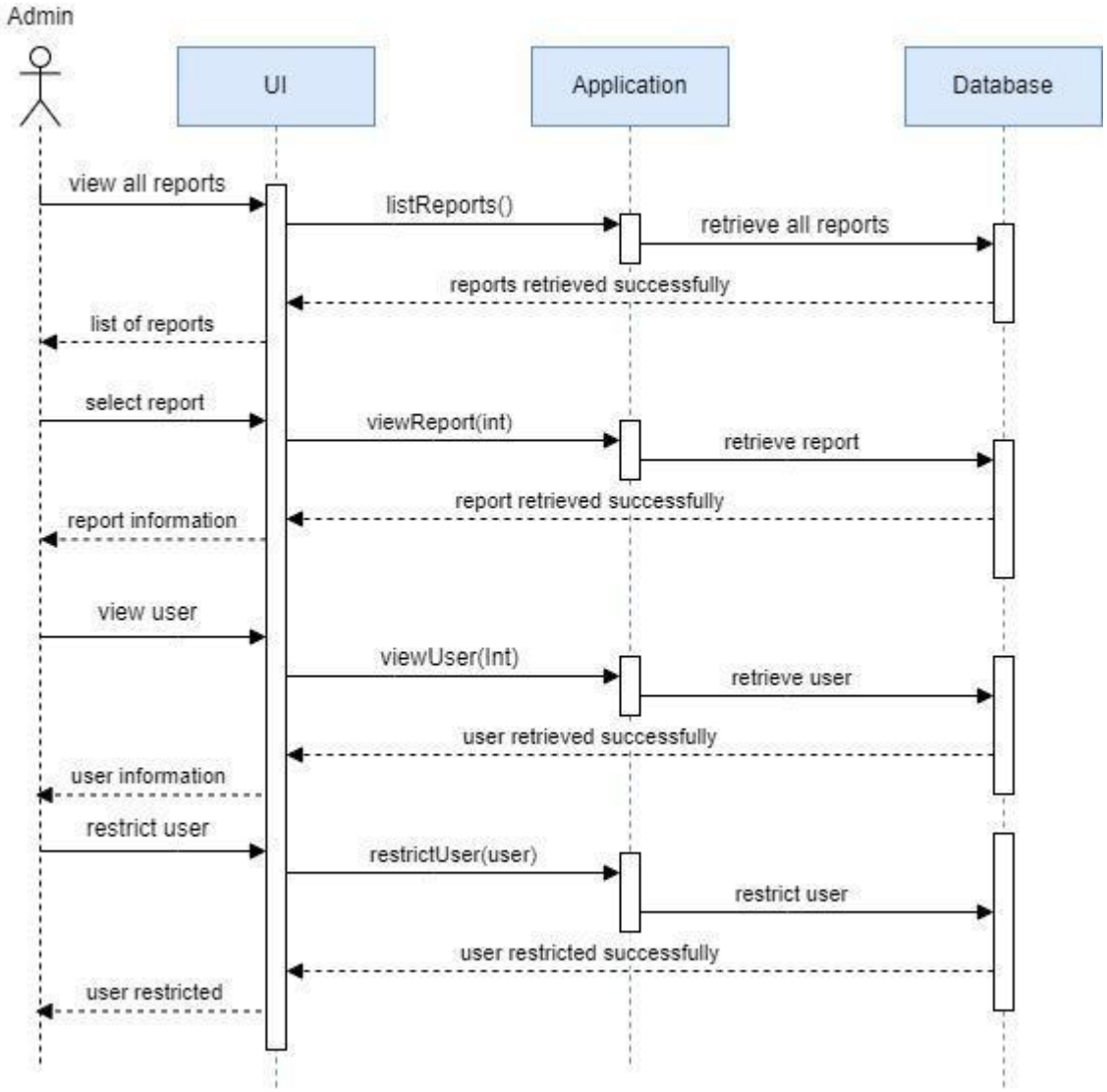


Figure 8 Sequence diagram for restrict user

4.4 Entity Relationship Diagram (ERD)

This is the ERD for our system, and the sequence diagrams are used to display the major entities within the system scope and the inter-relationships among these entities.

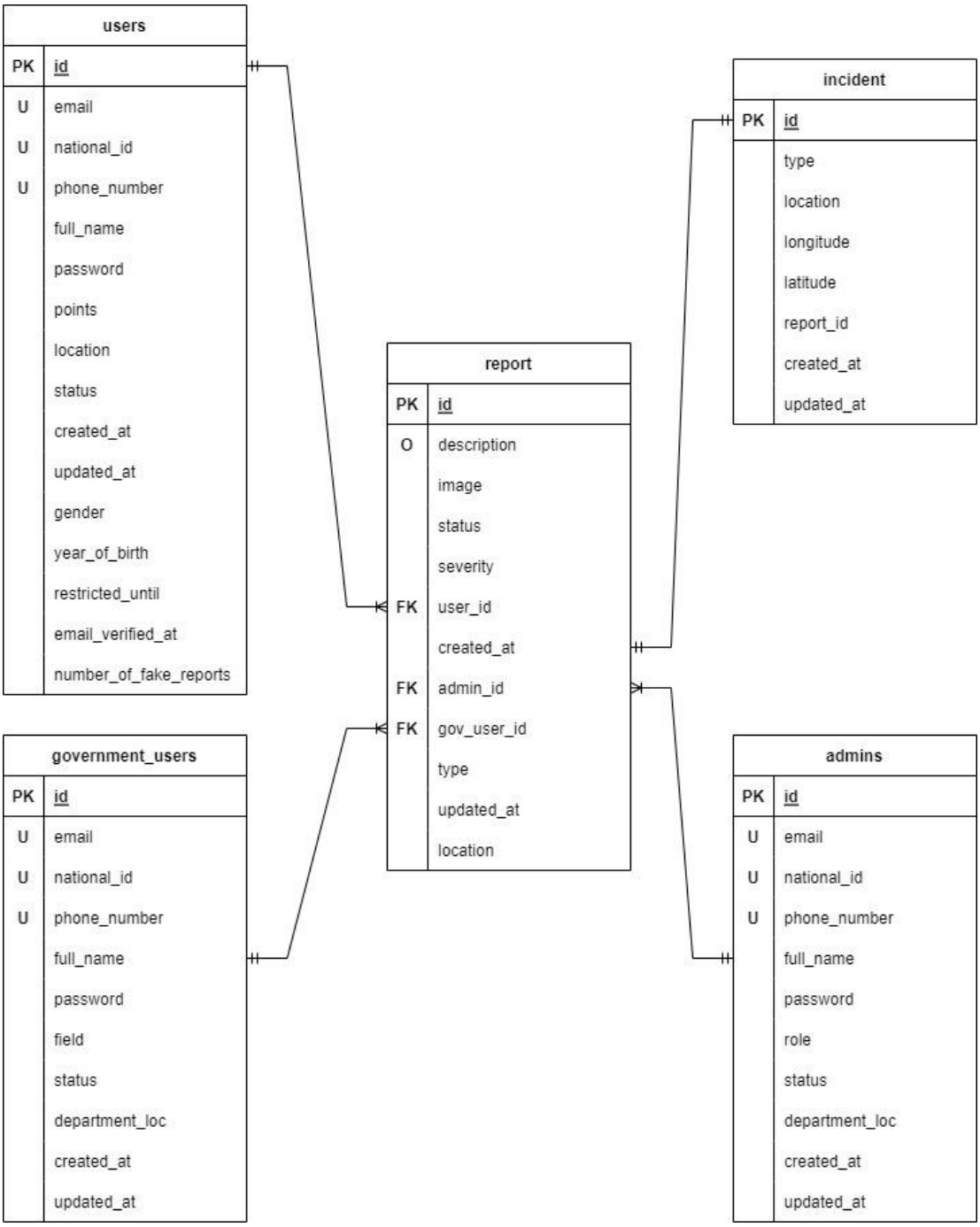


Figure 9 Project ERD

4.5 System GUI design

4.5.1 Mobile application GUI:

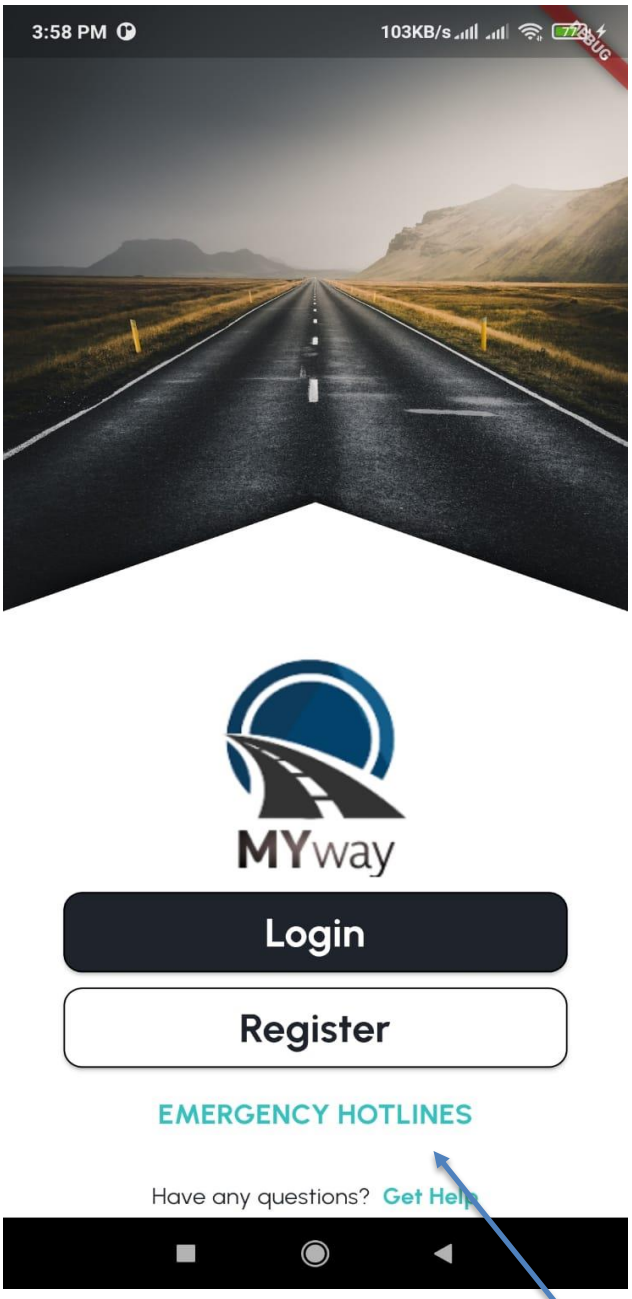


Figure 10 MYway page

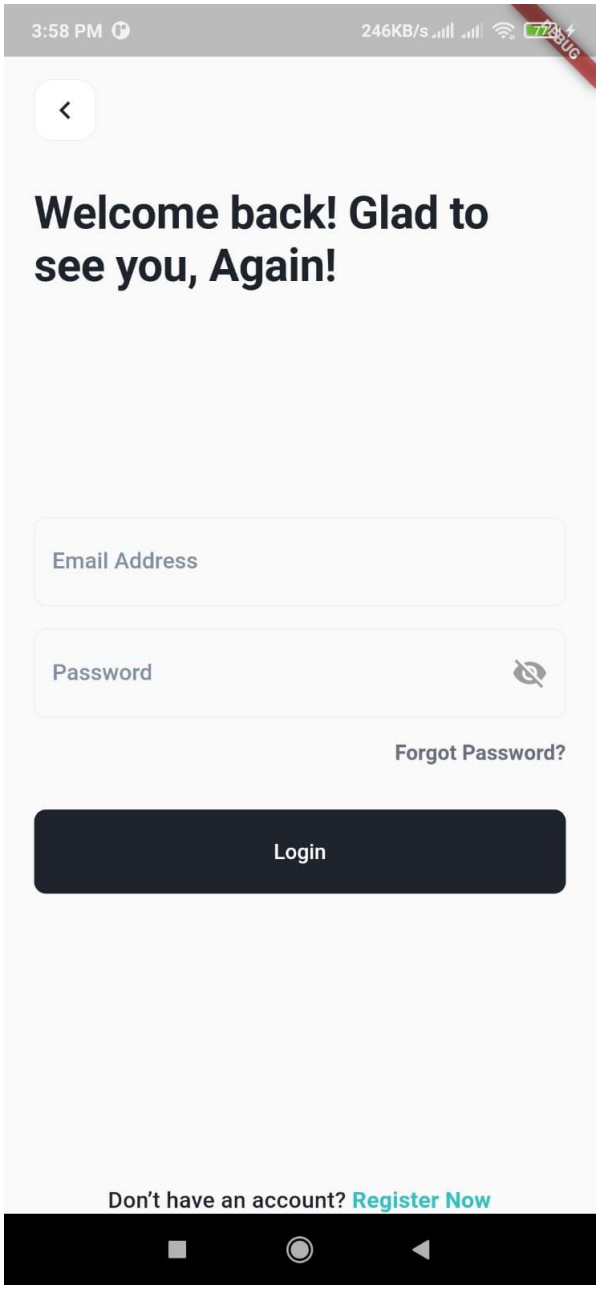


Figure 11 Login page

Show emergency hotlines

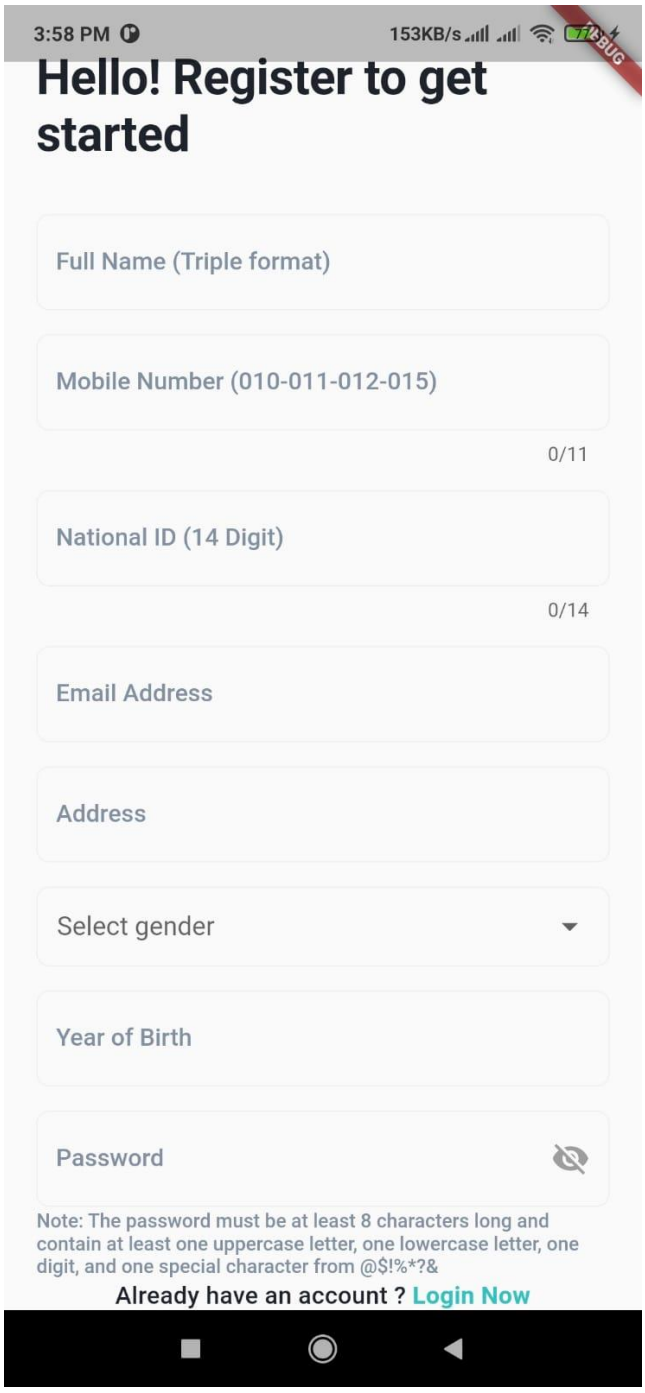


Figure 13 Register page

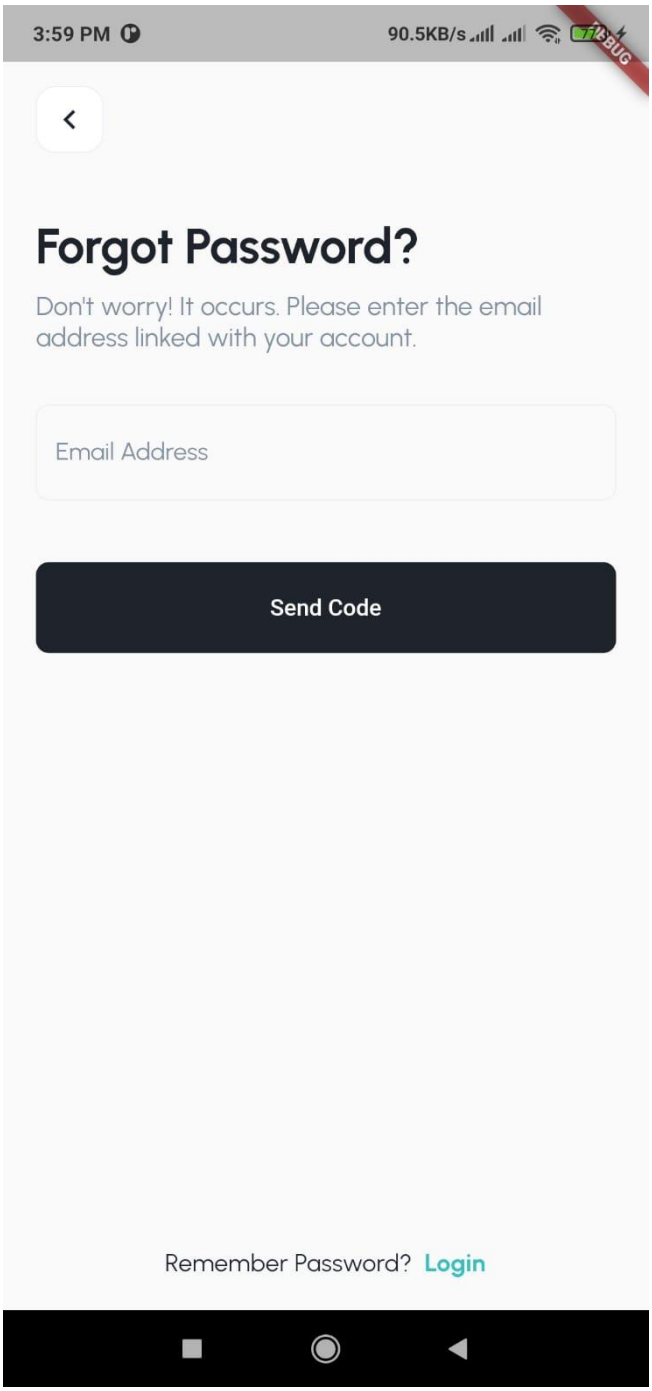


Figure 12 Forgot password page

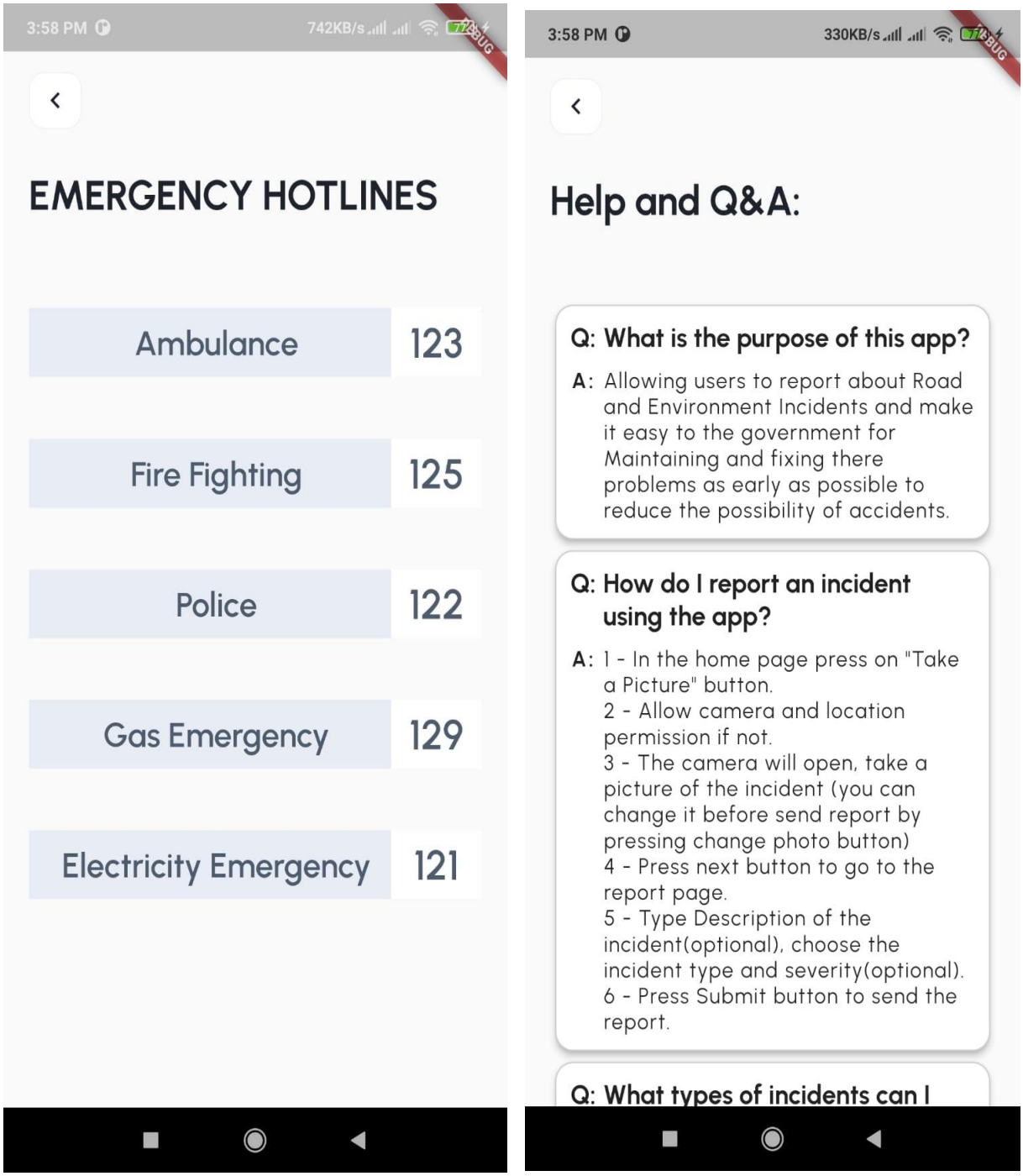


Figure 14 Hotlines page and Get help page

Take a picture to begin the reporting process

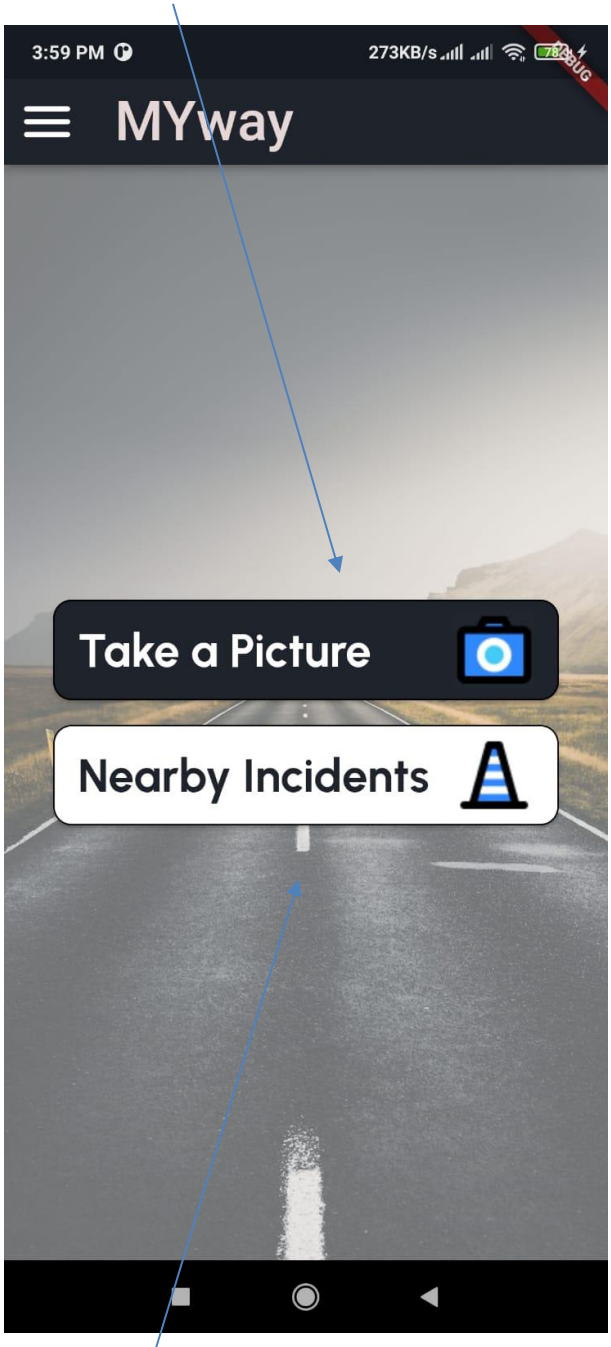


Figure 16 Home page

Open a map shows nearby reported incidents

User points for rewards

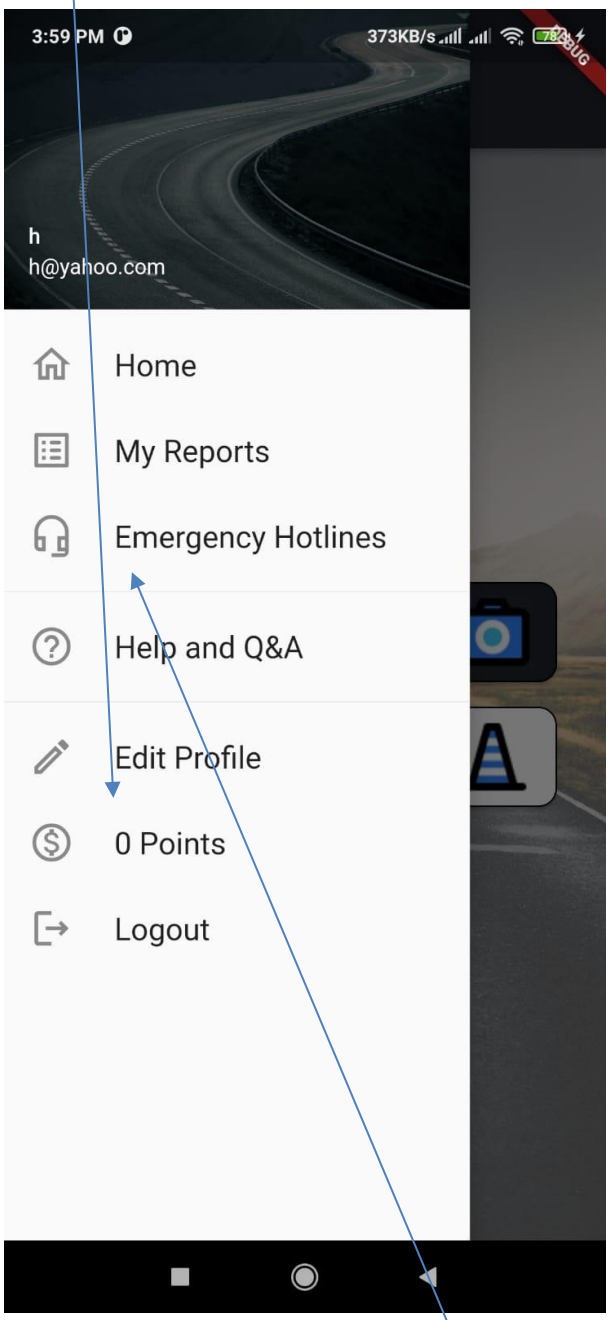


Figure 15 Navigation bar

Emergency hotlines

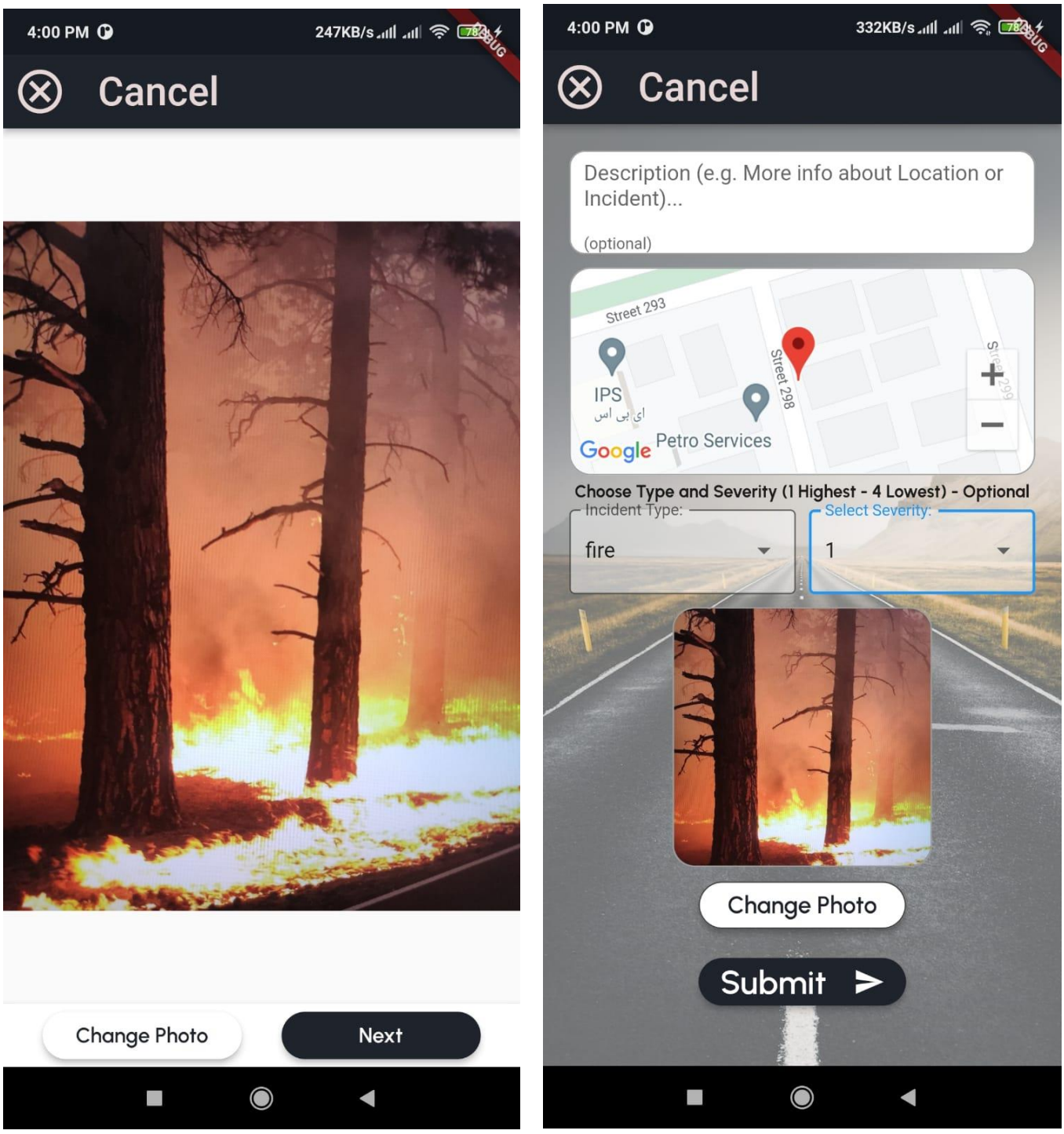


Figure 17 Take photo and Send report pages

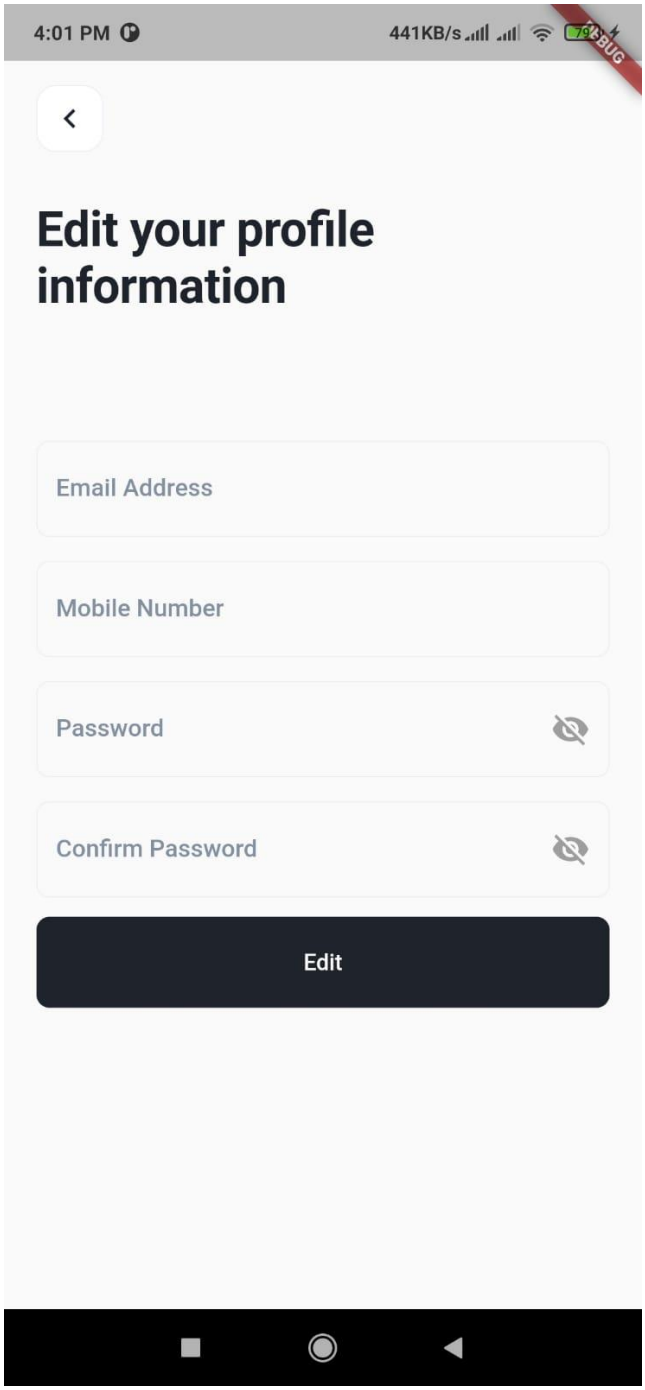


Figure 18 Edit profile page

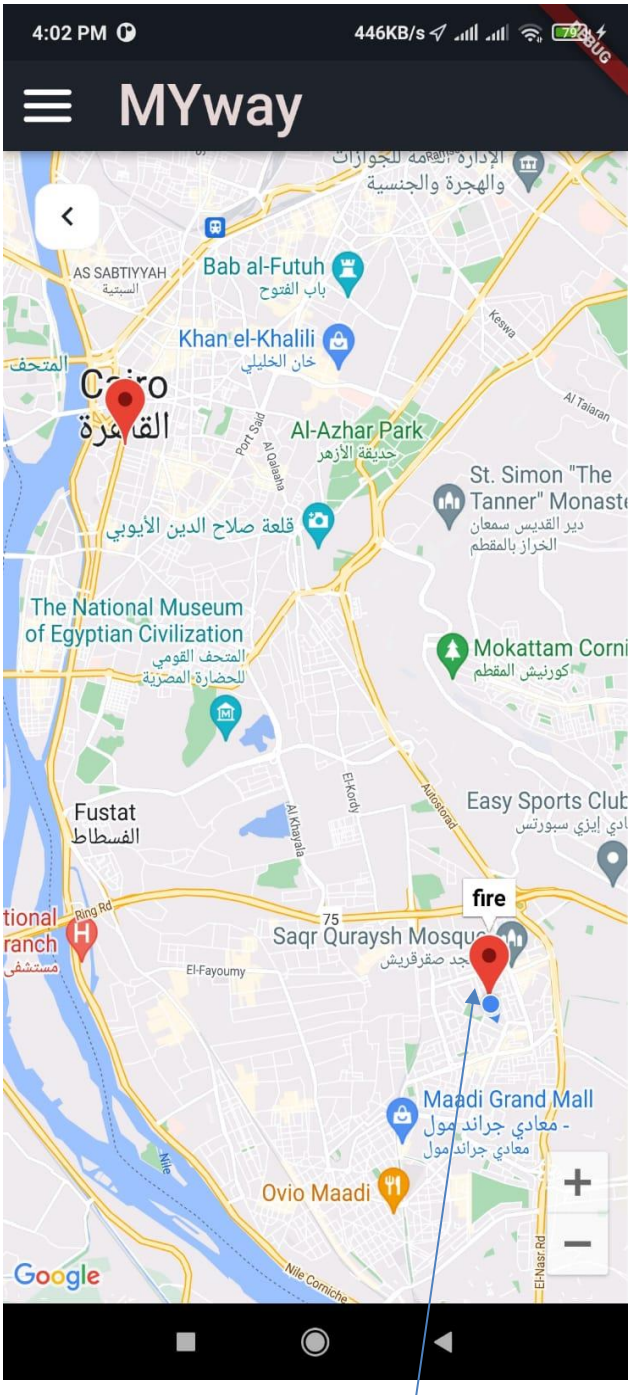


Figure 19 Nearby incident page

Open a map shows nearby reported incidents

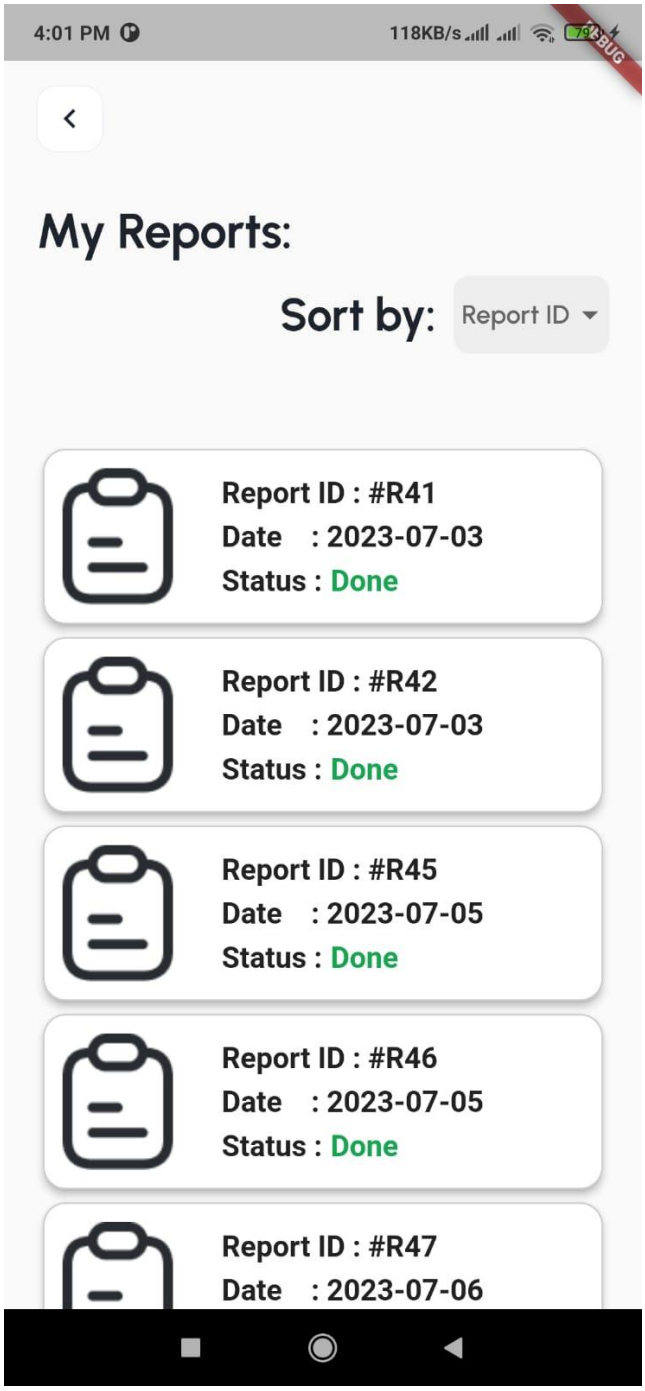


Figure 21 My reports page

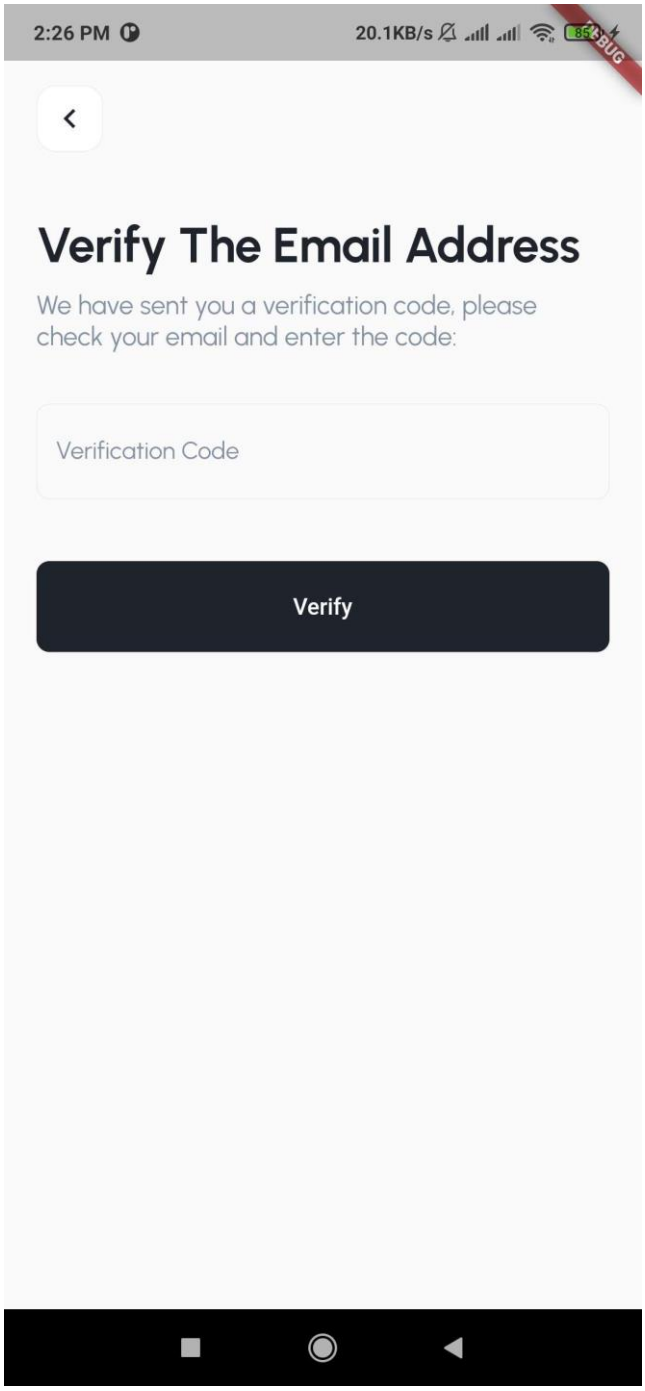


Figure 20 Verify email page

4.5.2 Website GUI

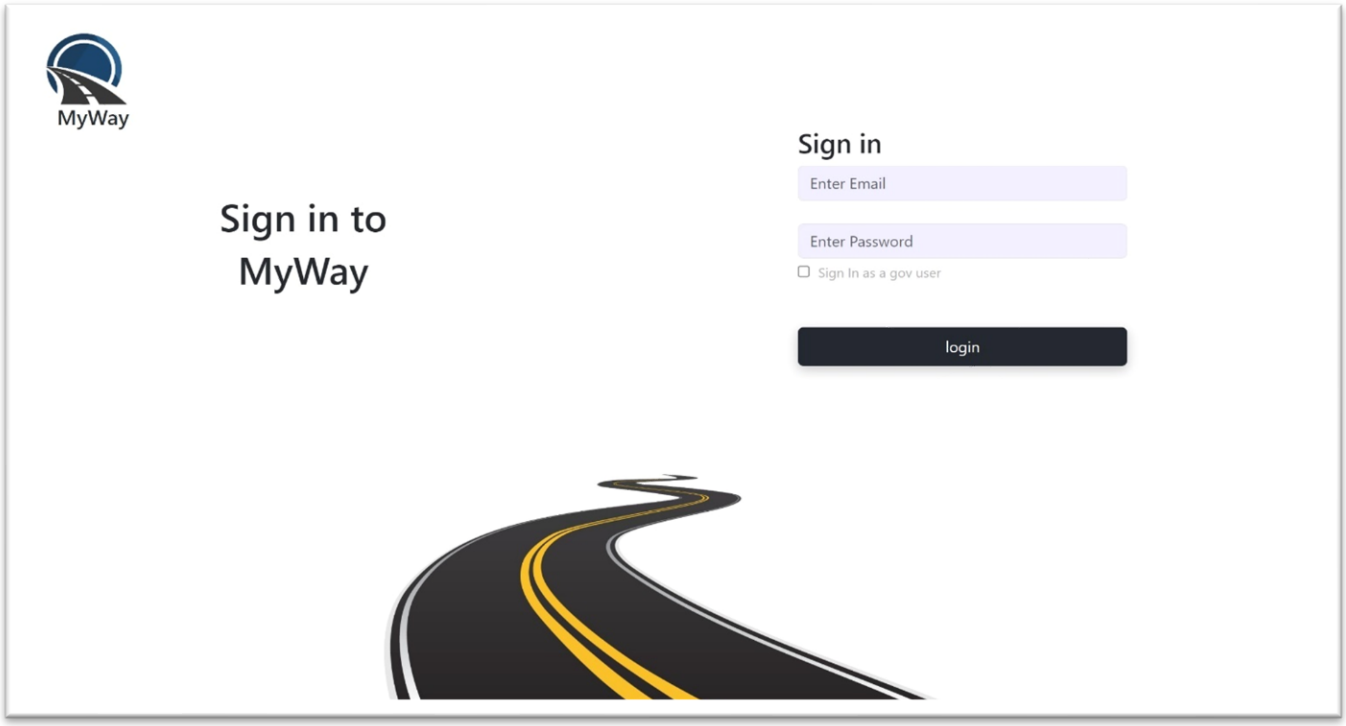


Figure 22 Login page

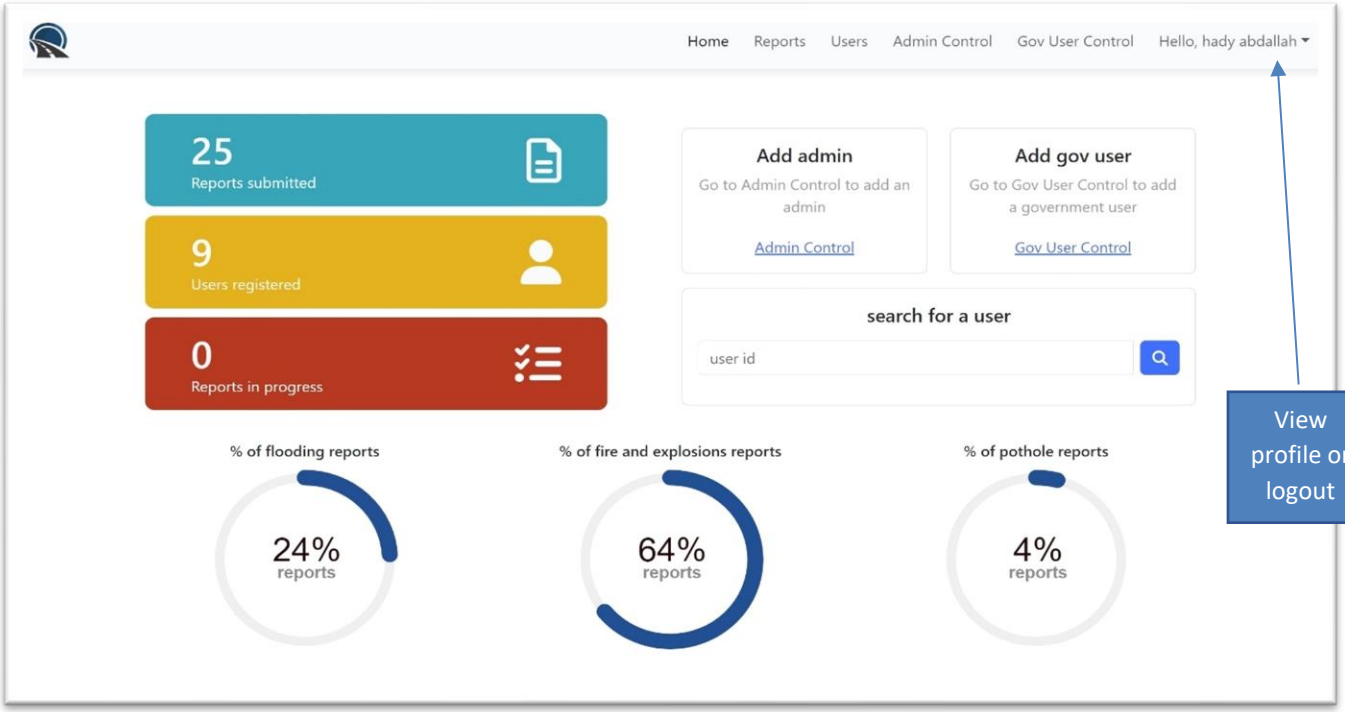


Figure 23 Home page

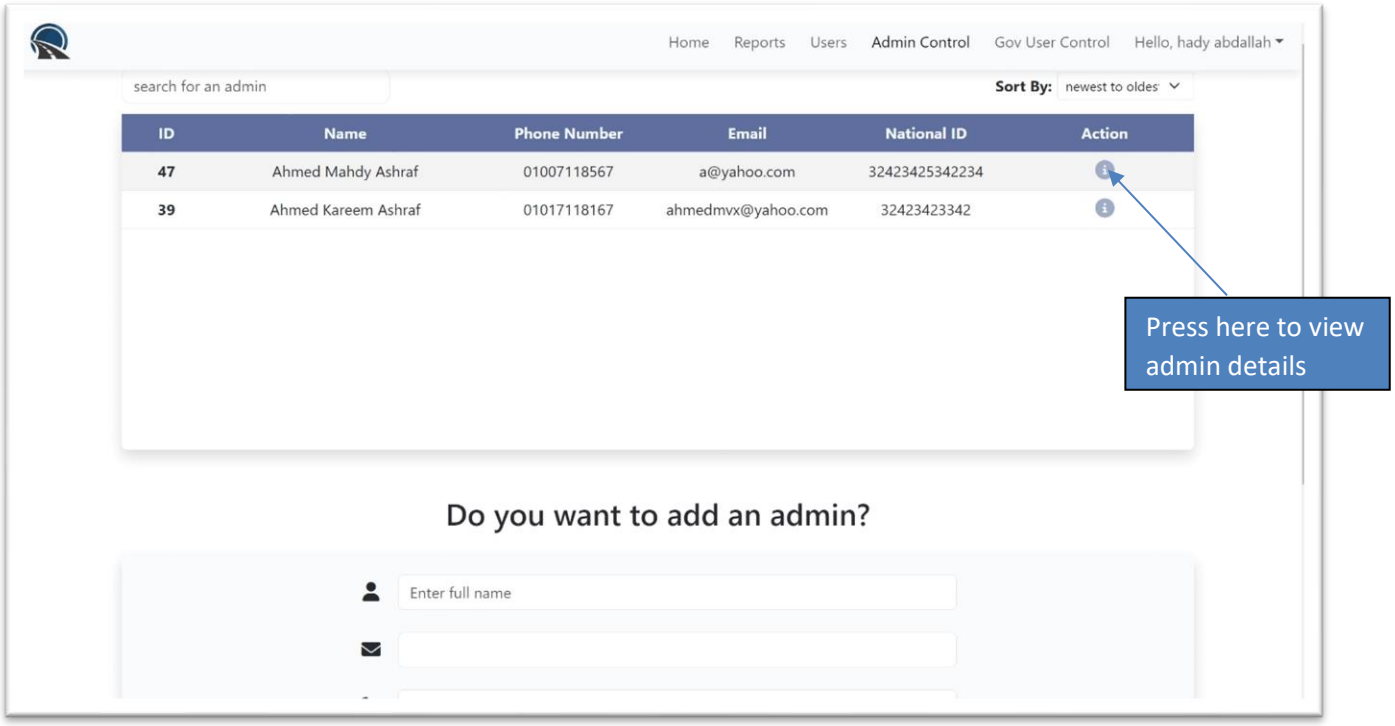


Figure 24 Admin Control page

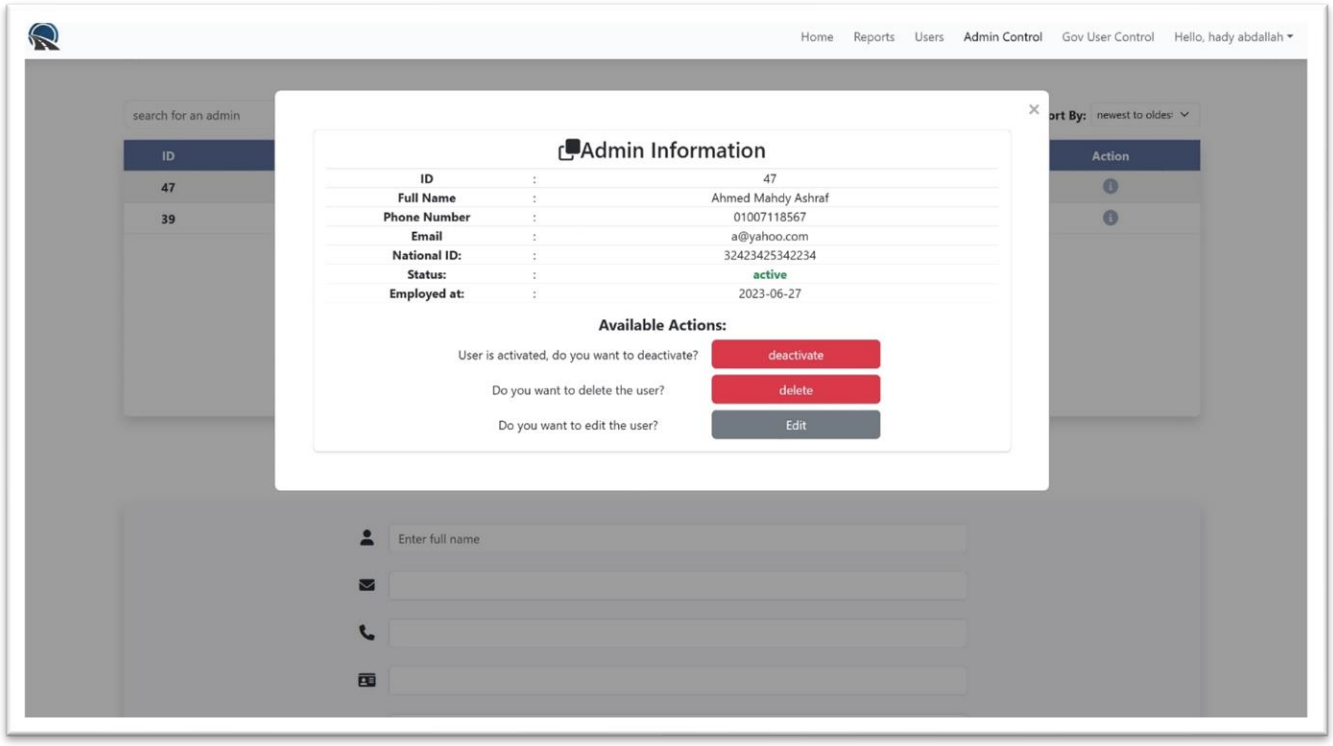


Figure 25 Admin information page

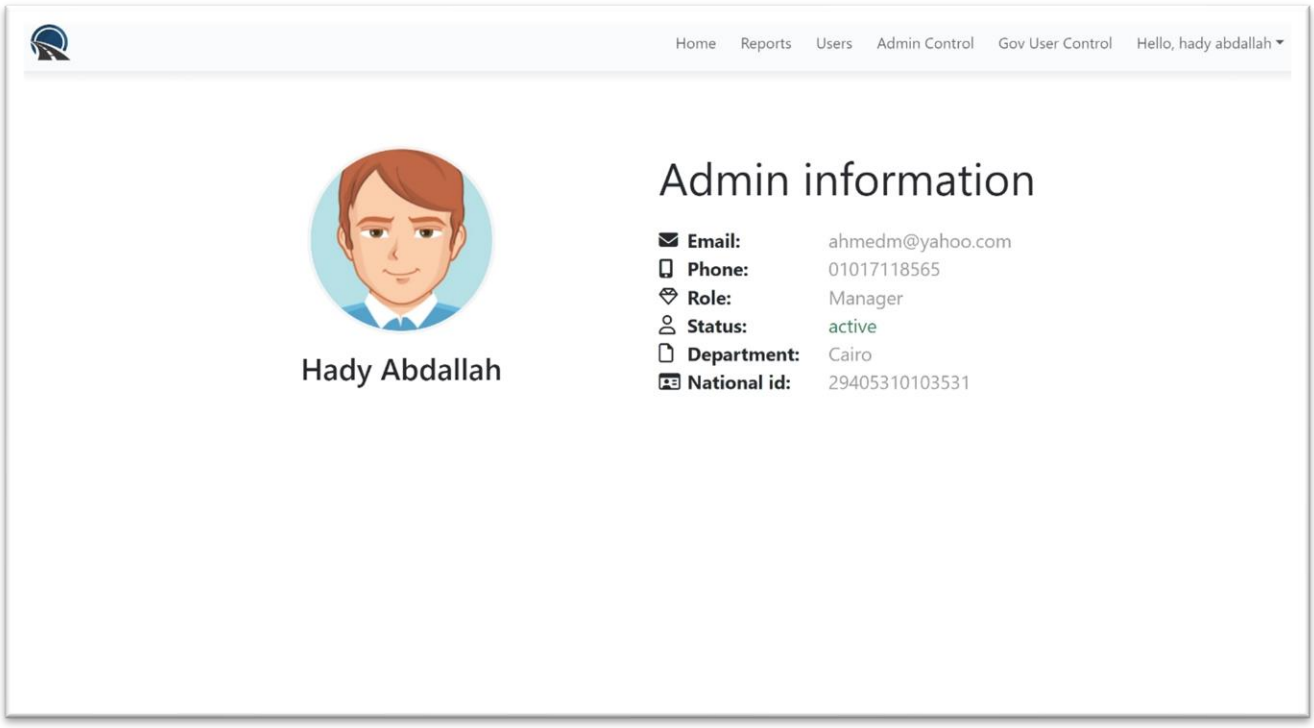


Figure 27 Admin profile page

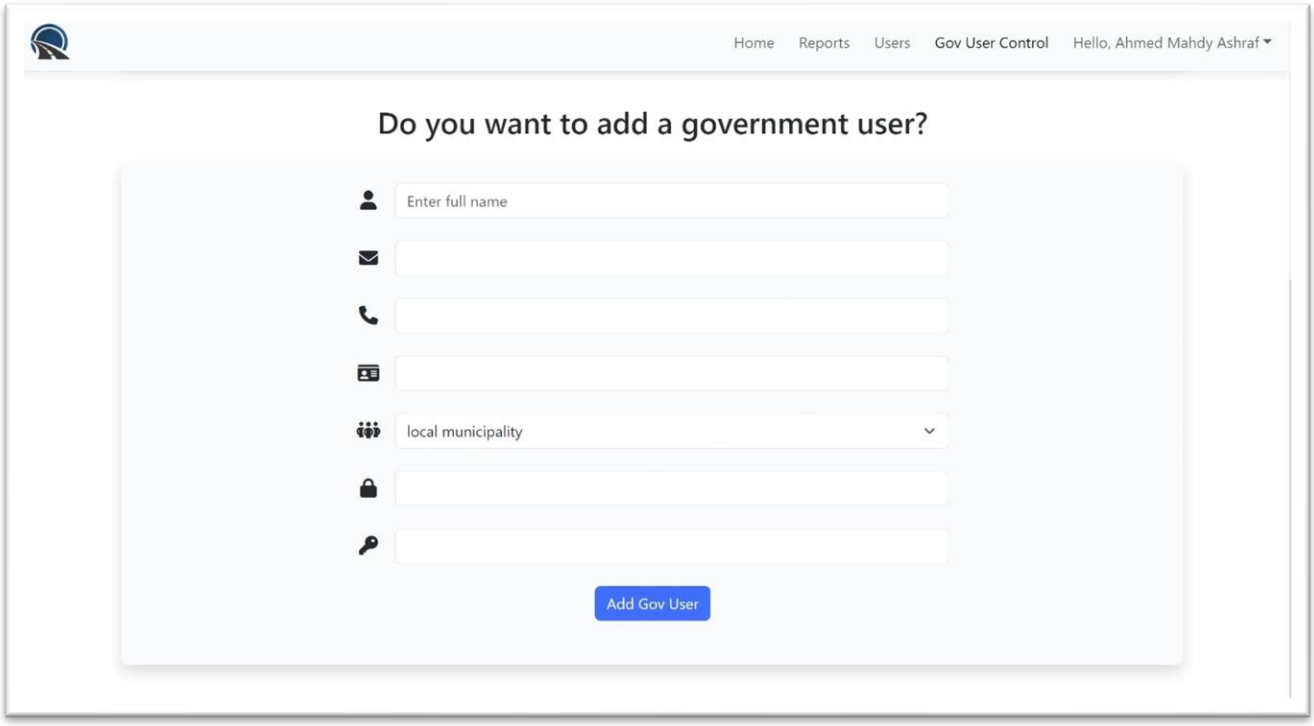


Figure 26 Gov user control page

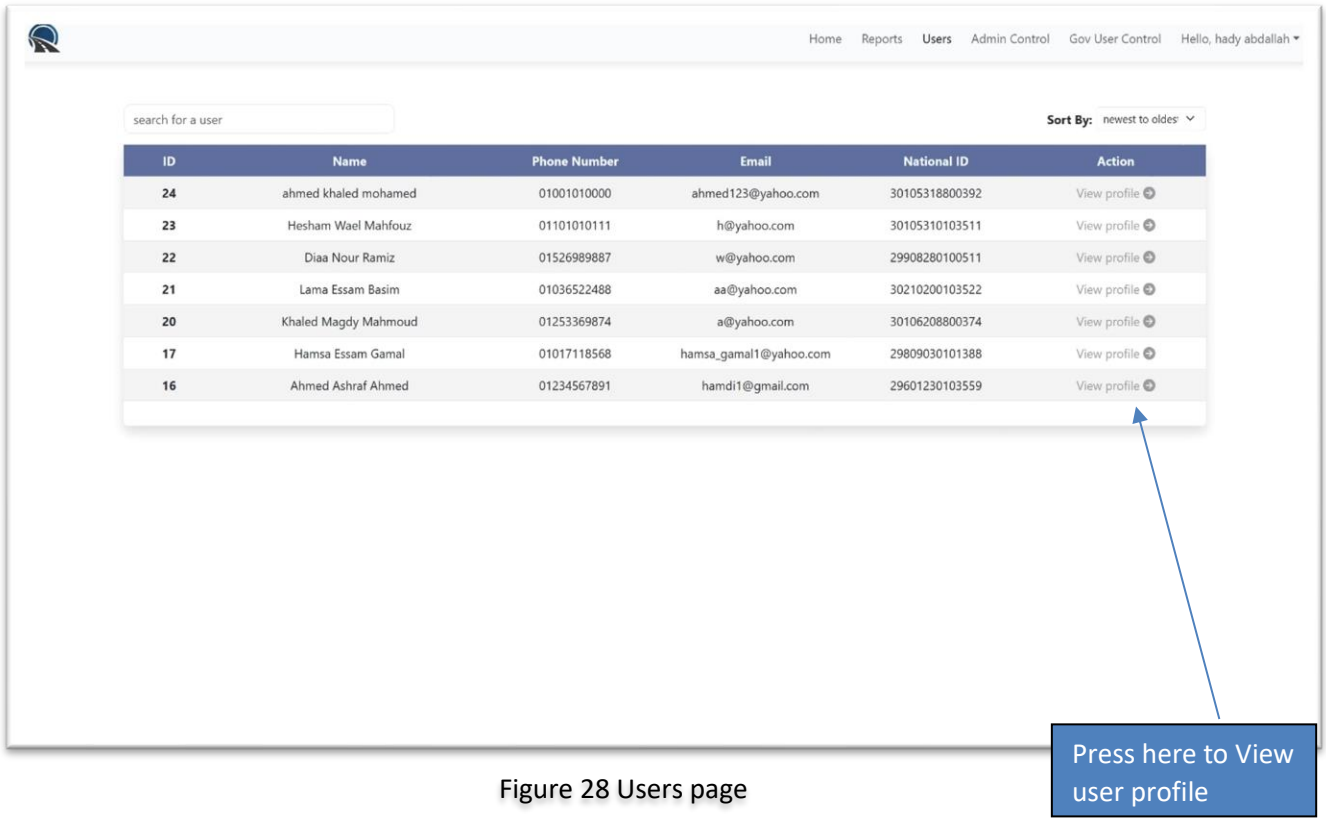


Figure 28 Users page

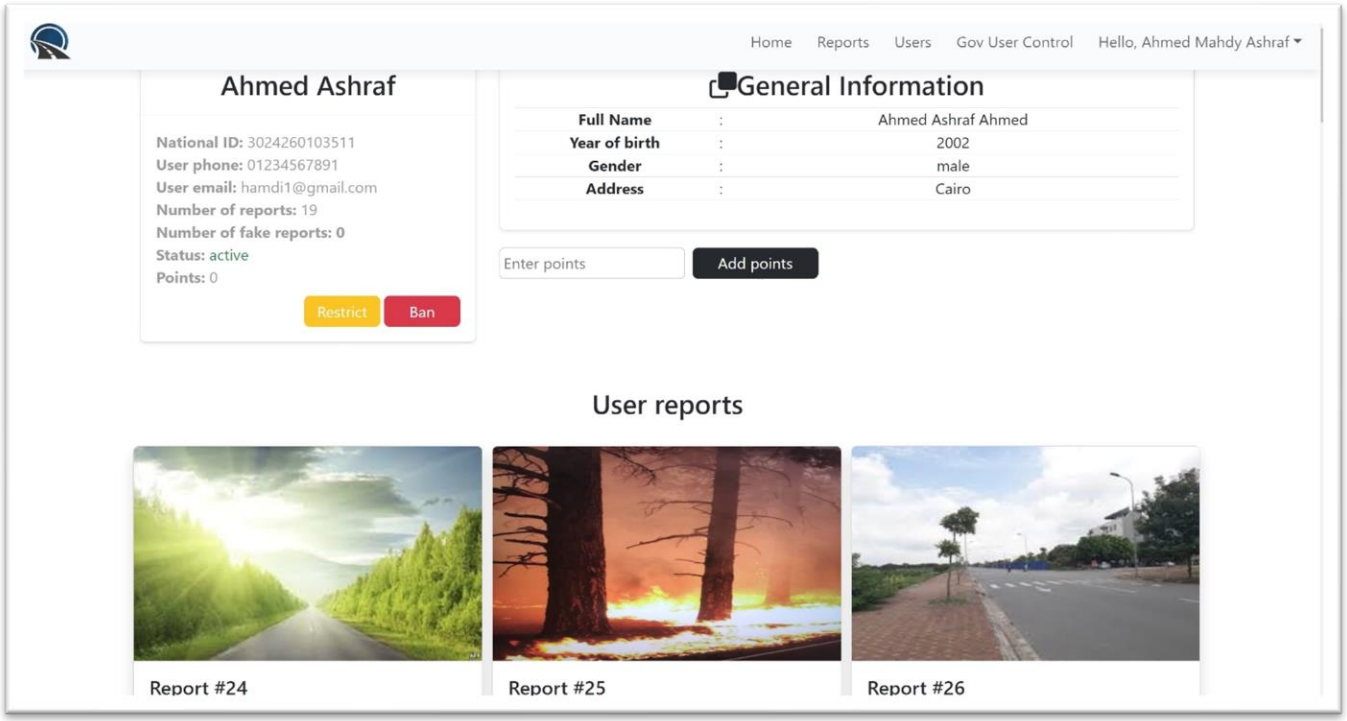


Figure 29 View user profile page

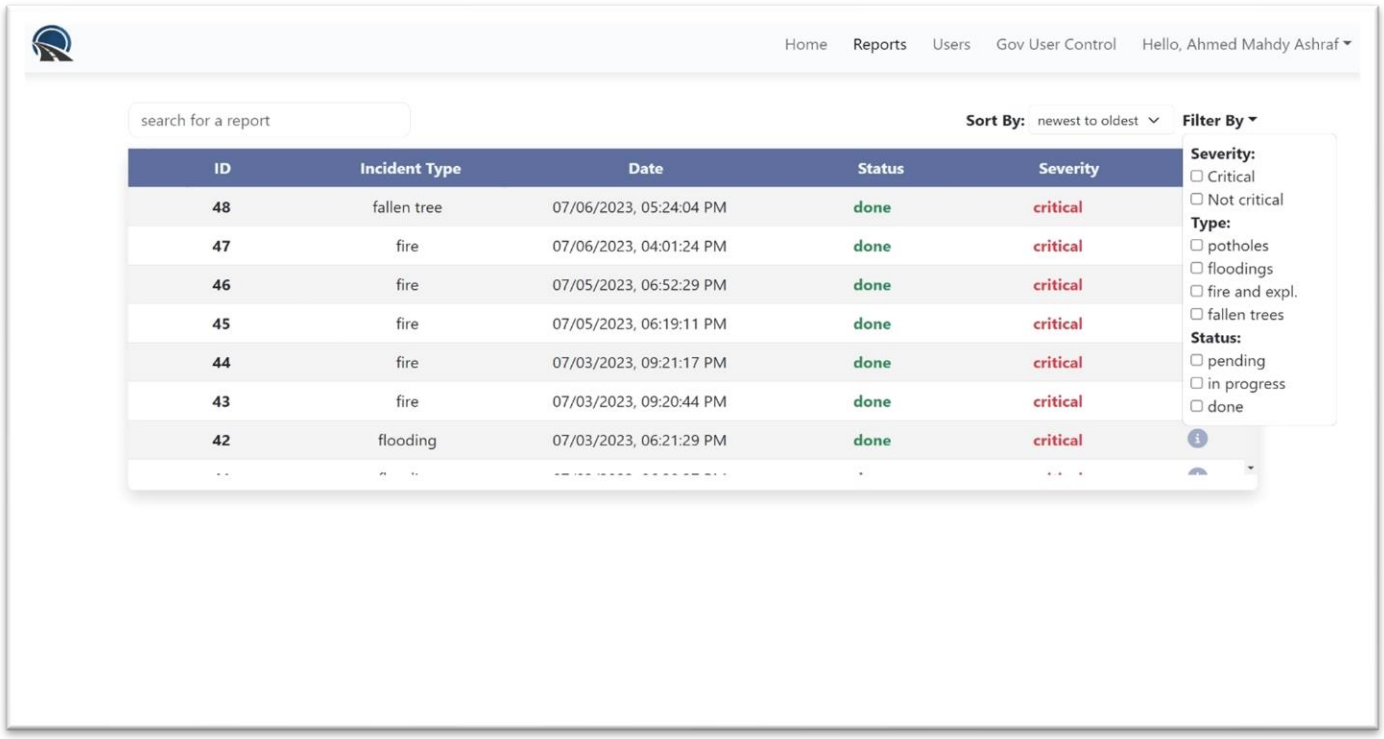


Figure 30 Reports page

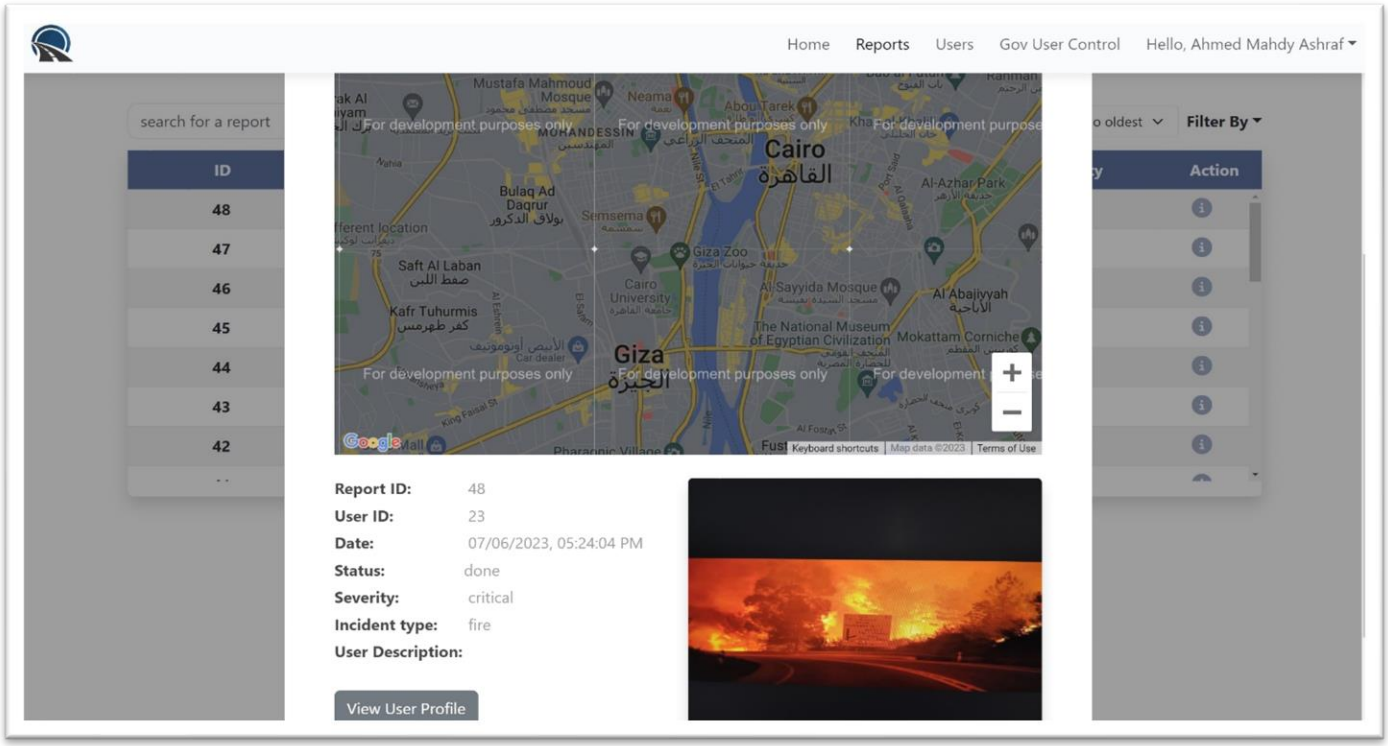


Figure 31 Incident information page

Chapter 5: Implementation and Testing

5.1 Implementation

For AI we used a combination of Python programming language and various machine learning and deep learning technologies to build an image classification model. The NumPy and Scikit-learn libraries are used for numerical computing, data splitting, and label encoding. The TensorFlow library with the Keras API is used for building and training a deep neural network on top of the pre-trained InceptionV3 model for image classification. The ImageDataGenerator class is used for data augmentation, which helps to increase the size and diversity of the dataset. The EarlyStopping and ModelCheckpoint callbacks are used for optimizing the training performance. Finally, the trained model is saved to and accessed from Google Drive. Overall, this code demonstrates the power of combining multiple cutting-edge technologies to build a robust and accurate image classification model. The dataset used in this project consists of 9,000 photos, which were collected through a combination of manual efforts and sourcing from Kaggle. The process of acquiring the dataset involved carefully curating a diverse range of images relevant to the specific image classification task at hand. The availability of such a large and diverse dataset ensured that our model could learn from a broad range of visual patterns and make accurate predictions on new, unseen images. The combination of carefully curated data and state-of-the-art technologies played a vital role in achieving a robust and accurate image classification solution.

The InceptionV3 model is a powerful pre-trained convolutional neural network (CNN) architecture that has been widely used for image classification tasks. It was introduced by Google researchers and has gained popularity due to its remarkable performance. One of its key advantages is that it comes with pre-trained weights, which were learned on a large dataset (ImageNet) containing millions of images. This pre-training allows InceptionV3 to capture general features and patterns from a wide range of images, making it highly effective for various image classification applications. By leveraging the knowledge gained from pre-training, the InceptionV3 model can be fine-tuned or used as a feature extractor on specific datasets, enabling efficient and accurate classification of new images.

Model Architecture:

The InceptionV3 model with pre-trained weights from ImageNet is used as the base model.

The pre-trained layers are frozen to prevent them from being updated during training.

Additional layers are added on top of the base model, including a global average pooling layer, batch normalization, and dense layers with dropout for regularization.

The number of output neurons in the last dense layer corresponds to the number of classes in the dataset, and SoftMax activation is applied for multi-class classification.

Model Evaluation:

Train accuracy: 0.9579283595085144 Test accuracy: 0.9056079983711243

Precision: 0.910 Recall: 0.906

F1 score: 0.906

Dataset:

Class 'Fallen Tress - Severity 1' has 1161 photos.

Class 'Fire and explosions - Severity 1' has 1444 photos.

Class 'Potholes - Severity 1' has 1015 photos.

Class 'Potholes - Severity 2' has 722 photos.

Class 'Normal - Severity none' has 3123 photos.

Class 'Flooding - Severity 1' has 1118 photos.

Class 'Fallen Trees - Severity 2' has 420 photos.

Total number of photos: 9003.

Some samples of Input Images for Prediction:

Figure 32 Input images and their prediction

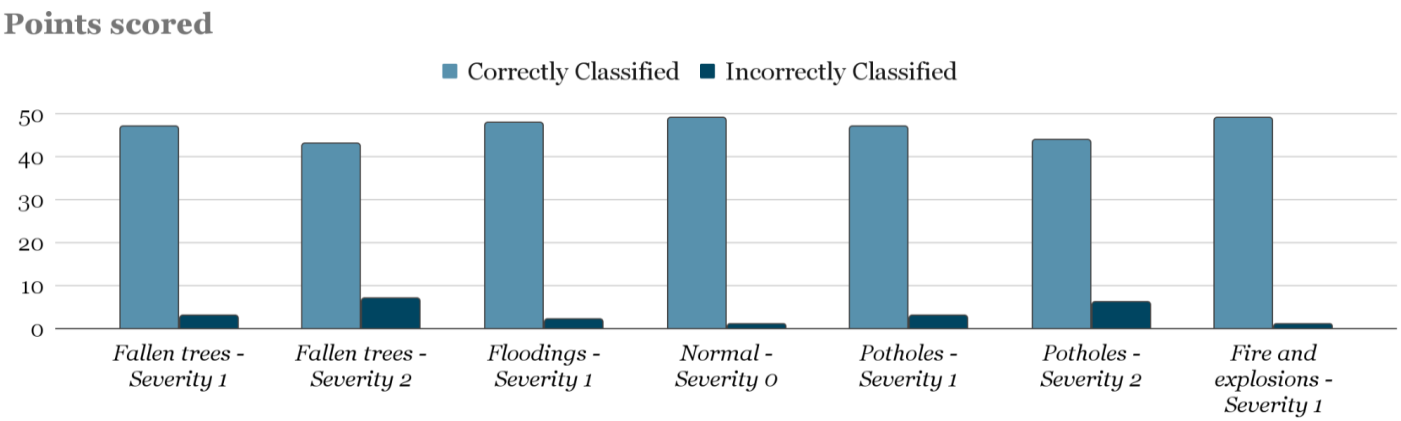


Figure 33 statistics for the model

Out of a total of 350 photos used, the majority of them (313 photos) were correctly detected by the model. However, there were 37 photos that were misclassified or detected incorrectly.

As shown by the pie charts and column chart above we note that the two classes with the best classification accuracy were "Normal - Severity 0" and "Fire and explosions - Severity 1", while the two classes with the lowest accuracy were "Fallen trees - severity 2" and "Potholes - Severity 2". Although the accuracy for these classes was not bad, we strive to increase the accuracy even further. To achieve this, we will implement a self-adaptive system that utilizes non-fake incident reports that were beneficial for the community and the road and adds them to our dataset as images. This will help our model improve its accuracy and ensure that it continues to adapt to changing conditions and trends over time. Our ultimate objective is to create a road incident detection and management system that provides the highest level of accuracy and reliability.

For the implementation of our mobile application, we utilized the Flutter framework, which allowed us to create a high-quality, visually appealing, and performant application for both the Android and iOS (iPhone Operating System) platforms using a single codebase. This significantly reduced development time and cost compared to writing separate codebases for each platform. The programming language used for Flutter is Dart, a modern, object-oriented language that is easy to learn and write. Dart offers various packages that simplify the implementation of the application, such as permission handler, http, image picker, and more. These packages enable developers to add critical functionality to their apps quickly, without having to write complex code from scratch.

For the implementation of our website application, we utilized Laravel for the backend due to its many useful features. Laravel follows the MVC (Model-View-Controller) architectural pattern, which separates the application logic into three components, facilitating easier maintenance and scalability. Laravel's Eloquent ORM (Object-Relational Mapping) provides a simple way to interact with databases, allowing for efficient querying, insertion, updating, and deleting of records. Additionally, Laravel prioritizes security, offering features such as CSRF (Cross-Site Request Forgery) protection, encryption, and password hashing.

For the frontend, we chose React due to its SPA (Single Page Application) feature, which results in a lightweight and faster website. React follows a component-based architecture, enabling the reuse and composition of UI (User Interface) components, making it easier to build and maintain the project.

5.2 Testing

Mobile app test cases

User sign up positive scenario					
Test scenario	Sign up			Test Case ID	T-A1
Test case	User sign up for new account positive scenario				
Pre-condition	None				
Test case description	The user tries to create a new account.				
Test steps	Test data	Expected output	Actual output	Test browser	Test result
1-sign up screen 2-write personal information	personal information	Navigate to login page	Navigated to login page	Mobile android emulator	pass

User sign up negative scenario					
Test scenario	Sign up			Test Case ID	T-A2
Test case	User sign up for new account negative scenario				
Pre-condition	None				
Test case description	The user tries to create a new account with invalid data such as national ID of length 13 not 14.				
Test steps	Test data	Expected output	Actual output	Test browser	Test result
1-sign up screen 2-write personal information	personal information	Error message	Error message	Mobile android emulator	pass

User sign in positive scenario					
Test scenario	Sign in			Test Case ID	T-A3
Test case	User sign in positive scenario				
Pre-condition	None				
Test case description	The user tries to sign in to his account.				
Test steps	Test data	Expected output	Actual output	Test browser	Test result
1-sign in screen 2-write Email and password	Email and password	Navigate to home page	Navigated to the home page	Mobile android emulator	pass

User sign in negative scenario					
Test scenario	Sign in			Test Case ID	T-A4
Test case	User sign in negative scenario				
Pre-condition	None				
Test case description	The user tries to sign in to his account with invalid data.				
Test steps	Test data	Expected output	Actual output	Test browser	Test result
1-sign in screen 2-write Email and password	Email and password	Error message	Error message	Mobile android emulator	pass

User send report positive scenario					
Test scenario	Send report			Test Case ID	T-A5
Test case	User Send report positive scenario				
Pre-condition	The user must be logged in.				
Test case description	The user will make a new report of an incident he sees on the road and will provide the necessary information.				
Test steps	Test data	Expected output	Actual output	Test browser	Test result
1-Home screen 2-take a picture and write the necessary information	Photo and necessary informatio n	Navigate to home page	Navigated to the home page	Mobile android emulator	pass

User send report negative scenario					
Test scenario	Send report			Test Case ID	T-A6
Test case	User Send report negative scenario				
Pre-condition	The user must be logged in.				
Test case description	The user will make a new report of normal things he sees on the road.				
Test steps	Test data	Expected output	Actual output	Test browser	Test result
1-Home screen 2-take a picture	Photo	The error message says that the photo does not have any incidents.	An error message appeared saying that the photo does not have any incidents.	Mobile android emulator	pass

User edit profile positive scenario					
Test scenario	Edit profile			Test Case ID	T-A7
Test case	User edit his profile positive scenario				
Pre-condition	The user must be logged in.				
Test case description	The user tries to change his password using a valid password.				
Test steps	Test data	Expected output	Actual output	Test browser	Test result
1-Home screen 2-Navigate to navbar 3-Edit profile	New Password	An output message that the information has been changed successfully.	An output message that the information has been changed successfully.	Mobile android emulator	pass

User edit profile negative scenario					
Test scenario	Edit profile			Test Case ID	T-A8
Test case	User edit his profile negative scenario				
Pre-condition	The user must be logged in.				
Test case description	The user tries to change his password using invalid password.				
Test steps	Test data	Expected output	Actual output	Test browser	Test result
1-Home screen 2- Navigate to navbar 3-Edit profile	New Password	An error message that the password is invalid	An error message that the password is invalid	Mobile android emulator	pass

User view nearby incidents positive scenario						
Test scenario	view nearby incidents				Test Case ID	T-A9
Test case	User view nearby incidents					
Pre-condition	The user must be logged in and he must open the map to see.					
Test case description	The user views the incidents near him to know which road to take and which to avoid.					
Test steps	Test data	Expected output	Actual output	Test browser	Test result	
1-Home screen 2-Nearby incidents	-	A map with nearby incidents visible to the user.	A map with nearby incidents visible to the user.	Mobile android emulator	Pass	

User view reports positive scenario					
Test scenario	view reports			Test Case ID	T-A10
Test case	User views the reports he sent.				
Pre-condition	The user must be logged in and click on My reports in the navbar.				
Test case description	The user views the reports that he sent and can keep track of their status.				
Test steps	Test data	Expected output	Actual output	Test browser	Test result
1-Home screen 2-navbar 3-My reports	-	A list of the reports that the user made and their status.	A list of the reports that the user made and their status.	Mobile android emulator	Pass

Web application test cases

User sign in positive scenario					
Test scenario	Sign in			Test Case ID	T-B1
Test case	User sign in positive scenario				
Pre-condition	None				
Test case description	The user tries to sign in to his account with the Email and password provided by the manager.				
Test steps	Test data	Expected output	Actual output	Test browser	Test result
1-sign in screen 2-write Email and password	Email and password	Navigate to home page	Navigated to the home page	Chrome browser	pass

User sign in negative scenario					
Test scenario	Sign in			Test Case ID	T-B2
Test case	User sign in negative scenario				
Pre-condition	None				
Test case description	The user tries to sign in to his account with invalid data.				
Test steps	Test data	Expected output	Actual output	Test browser	Test result
1-sign in screen 2-write Email and password	Email and password	Error message	Error message	Chrome browser	pass

Web application test cases for the manager

Add an admin positive scenario					
Test scenario	Add admin			Test Case ID	T-C1
Test case	Add an admin positive scenario				
Pre-condition	The Manager must be logged in successfully and in the Admin control page.				
Test case description	The admin will try to add a new admin from the admin control page.				
Test steps	Test data	Expected output	Actual output	Test browser	Test result
1-admin control page 2-write personal information	Personal information	Redirect to the same page but the admin will be added	Redirected to the same page but the admin will be added	Chrome browser on a laptop	pass

Add a Gov user positive scenario					
Test scenario	Add Gov user			Test Case ID	T-C2
Test case	Add a Gov user positive scenario				
Pre-condition	The Manager must be logged in successfully and in the Gov user control page.				
Test case description	The admin will try to add a new Gov user from the Gov User control page.				
Test steps	Test data	Expected output	Actual output	Test browser	Test result
1-Gov User control page 2-write personal information	Personal information	Redirect to the same page but the Gov user will be added	Redirected to the same page but the Gov user will be added	Chrome browser on a laptop	pass

Delete an admin positive scenario					
Test scenario	Delete admin			Test Case ID	T-C3
Test case	Delete an admin positive scenario				
Pre-condition	The Manager must be logged in successfully and in the Admin control page.				
Test case description	The admin will try to delete an admin from the admin information page.				
Test steps	Test data	Expected output	Actual output	Test browser	Test result
1-admin control page 2-press on the action button 3-press the Delete button	-	Navigate to the Admin control page	Navigated to the Admin control page	Chrome browser on a laptop	pass

Deactivate an admin positive scenario					
Test scenario	Deactivate admin			Test Case ID	T-C4
Test case	Deactivate an admin positive scenario				
Pre-condition	The Manager must be logged in successfully and in the Admin control page.				
Test case description	The admin will try to deactivate an admin from the admin information page.				
Test steps	Test data	Expected output	Actual output	Test browser	Test result
1-admin control page 2-press on the action button 3-press the Deactivate button	-	Redirect to the same page but the Admin will be deactivated	Redirect to the same page and the Admin is deactivated	Chrome browser on a laptop	pass

Search for a user by ID positive scenario					
Test scenario	Search for a user			Test Case ID	T-C5
Test case	Search for a user positive scenario				
Pre-condition	The Manager must be logged in successfully and in the Users page.				
Test case description	The manager will try to search for a specific user by his ID.				
Test steps	Test data	Expected output	Actual output	Test browser	Test result
1-Users page 2-write user ID	User id	The specific user row will appear	The specific user row appeared	Chrome browser on a laptop	pass

Search for a user by ID negative scenario					
Test scenario	Search for a user			Test Case ID	T-C6
Test case	Search for a user negative scenario				
Pre-condition	The Manager must be logged in successfully and in the Users page.				
Test case description	The manager will try to search for a user with the wrong ID.				
Test steps	Test data	Expected output	Actual output	Test browser	Test result
1-Users page 2-write wrong user ID	User id	There will be no rows displayed.	There are no rows displayed	Chrome browser on a laptop	pass

Web application test cases for the admin

View profile positive scenario					
Test scenario	View profile			Test Case ID	T-D1
Test case	View profile positive scenario				
Pre-condition	The Admin must be logged in successfully.				
Test case description	The admin will try to see his information from his profile.				
Test steps	Test data	Expected output	Actual output	Test browser	Test result
1-Home page 2-press on his name on the navbar 3-press on view profile	-	Display the admin information	The admin information is displayed	Chrome browser on a laptop	pass

Search for a report positive scenario					
Test scenario	Search for a report			Test Case ID	T-D2
Test case	Search for a report positive scenario				
Pre-condition	The Admin must be logged in successfully and in the Reports page.				
Test case description	The admin will try to see a specific report by the report id.				
Test steps	Test data	Expected output	Actual output	Test browser	Test result
1-Reports page 2-write the report ID	Report ID	Display only the report row	Only the report row is displayed	Chrome browser on a laptop	pass

Delete a Gov user positive scenario					
Test scenario	Delete Gov user			Test Case ID	T-D3
Test case	Delete a Gov user positive scenario				
Pre-condition	The admin must be logged in successfully and in the Gov user control page.				
Test case description	The admin will try to delete a Gov user from the Gov user information page.				
Test steps	Test data	Expected output	Actual output	Test browser	Test result
1- Gov user control page 2-press on the action button 3-press the Delete button	-	Navigate to the Gov user control page	Navigated to the Gov user control page	Chrome browser on a laptop	pass

Deactivate a Gov user positive scenario					
Test scenario	Deactivate Gov user			Test Case ID	T-D4
Test case	Deactivate a Gov user positive scenario				
Pre-condition	The admin must be logged in successfully and in the Gov user control page.				
Test case description	The admin will try to deactivate a Gov user from the Gov user information page.				
Test steps	Test data	Expected output	Actual output	Test browser	Test result
1- Gov user control page 2-press on the action button 3-press the Deactivate button	-	Redirect to the same page but the Gov user will be deactivated	Redirect to the same page and the Gov user is deactivated	Chrome browser on a laptop	pass

Web application test cases for the gov user

Search for a report positive scenario					
Test scenario	Search for a report			Test Case ID	T-E1
Test case	Search for a report positive scenario				
Pre-condition	The Manager must be logged in successfully and in the Reports page.				
Test case description	The manager will try to search for a specific report by his ID.				
Test steps	Test data	Expected output	Actual output	Test browser	Test result
1-Reports page 2-write report ID	Report id	The specific report row will appear	The specific report row appeared	Chrome browser on a laptop	pass

Mark report as done positive scenario					
Test scenario	Mark report as done			Test Case ID	T-E2
Test case	Mark report as done positive scenario				
Pre-condition	The Gov user must be logged in successfully and in the reports page.				
Test case description	The Gov user will try to mark a specific report that was assigned to him as done.				
Test steps	Test data	Expected output	Actual output	Test browser	Test result
1- Reports page 2-press on the action button 3-press on Mark as done button	-	Redirect to the same page but the report will mark as Done.	Redirect to the same page and the report is marked as Done.	Chrome browser on a laptop	pass

Filter the reports by status positive scenario					
Test scenario	Filter by			Test Case ID	T-E3
Test case	Filter the reports by status positive scenario				
Pre-condition	The Gov user must be logged in successfully and in the reports page.				
Test case description	The admin will try to filter the reports by status Done.				
Test steps	Test data	Expected output	Actual output	Test browser	Test result
1- Reports page 2-press on the filter by and choose done	-	Redirect to the same page, but the report that is done will only display.	Redirected to the same page, and the report with the done statue is displayed.	Chrome browser on a laptop	pass

logout positive scenario					
Test scenario	Logout			Test Case ID	T-E4
Test case	logout positive scenario				
Pre-condition	The Gov user must be logged in successfully.				
Test case description	The admin will try to logout from his account.				
Test steps	Test data	Expected output	Actual output	Test browser	Test result
1- Press on the name in the navbar. 2-Press on logout.	-	Navigate to the login page.	Navigated to the login page.	Chrome browser on a laptop	Pass

Conclusion

In conclusion, our AI-powered road incident detection app is a unique and effective solution for road safety management. With fast and reliable incident detection, a multi-platform approach, and a rewards program that motivates users to report non-fake incidents, our app provides a user-friendly and efficient experience for users and the community. Moreover, our self-adaptive system addresses the shortage of images in some classes of our dataset, ensuring that our model continues to improve its accuracy over time. By using reported images of incidents that were solved and beneficial to the community, we can increase the number of images in our dataset, which will improve the accuracy of our model even further. As we continue to develop and refine our app, we believe that it will become an essential tool for government authorities, emergency responders, and the general public, contributing to a safer and more efficient transportation system for all.

References

1- **Flutter documentation**
https://docs.flutter.dev/

2- **Laravel documentation**
https://laravel.com/docs/10.x

3- **TensorFlow documentation**
https://www.tensorflow.org/guide

4- **Complete Machine Learning & Data Science Bootcamp 2023**

https://www.udemy.com/course/complete-machine-learning-and-data-science-zero-to-mastery/

5- **A selection of links to datasets**
https://www.kaggle.com/datasets/tunhunhminh/demodata

https://public.roboflow.com/object-detection/pothole/1