جامعة الملك سعود
King Saud University

كلية علوم الحاسب والمعلومات
قسم تقنية المعلومات

A Smart Dash Camera system to report Traffic Violations

IT 497: Graduation Project Report
Product Release-2

Prepared by

Student Name	University ID
Dhai AlMurshed	441200912
Manar Bin Muatieb	441201106
Sarah AlOtaibi	441200815
Ghadeer AlMaymoon	441200441

Supervised by
Dr. Luluh Aldhubayi

Second Semester 1444
2022/2023

Table of Contents

Table of Figures

Table of Tables

A Smart Dash Camera system to report Traffic Violations

Sarah Alotaibi[1], Ghadeer Almaymoon[2], Manar bin Mutaieb[3] and Dhai Almurshed[4]

[1]Information Technology Department, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia; 441200815@student.ksu.edu.sa
[2]Information Technology Department, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia; 441200441@student.ksu.edu.sa
[3]Information Technology Department, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia; 441201106@ student.ksu.edu.sa
[4]Information Technology Department, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia; 441200912 @student.ksu.edu.sa

Abstract (English):
Traffic accidents kill 4,621 people and injure 25,561 in Saudi Arabia [1]. Traffic violations reflect the large number of traffic accidents; in 2019, the number of traffic violations surpassed 50,000,000 [2]. Saudi Arabia has a system to detect violations such as speeding and crossing red lights, but not drifting, driving in the opposite direction, or overtaking, as these occur on roads outside the range of these systems. Taking advantage of visual data such as live stream videos of the dashcam and combining it with computer vision technology, which is a field of artificial intelligence (AI) that enables systems to derive meaningful information from digital videos, can reveal a technology that has not yet been initiated in Saudi Arabia. The proposed methodology consists of developing a mobile application linked to the driver's dashcam to obtain the live stream and detect violations, such as drifting, driving in the opposite direction, and overtaking. The traffic violation system consists of a pipeline of two different modules that analyze the video frames: vehicle detection and traffic violation detection modules. The system was implemented using the data collected by "RASD" team and achieved 98% accuracy. This result shows that the used methodology is applicable in real life.

Abstract (Arabic):
في المملكة العربية السعودية، تتسبب حوادث المرور في ٤,٦٢١ حالة وفاة و٢٥,٥٦١ إصابة [1]. ينعكس هذا العدد من الحوادث المرورية على السبب الرئيسي لها وهي المخالفات المرورية، حيث تجاوز عدد المخالفات المرورية ٥٠,٠٠٠,٠٠٠ في عام ٢٠١٩[2]. على الرغم من وجود أنظمة فعّالة في المملكة العربية السعودية لرصد المخالفات المرورية مثل قطع الإشارة المرورية وتجاوز حد السرعة المسموح به، هناك مخالفات تقع خارج نطاق هذه الأنظمة مما يعرقل عملية رصدها مثل التفحيط، أو التجاوز المخالف، أو عكس الطريق. يمكن الاستفادة من البيانات المرئية مثل البث المباشر لمقاطع الفيديو الخاصة بالداش كام ودمجها مع تقنية الرؤيه الحاسوبية، والتي تندرج تحت الذكاء الاصطناعي حيث تمكّن أجهزة الكمبيوتر والأنظمة من استخلاص المعلومات من مقاطع الفيديو الرقمية. بدمج هذه التقنيات باستطاعتنا خلق تقنية جديدة في المملكة العربية السعودية باالمنهجية المقترحة والتي تقوم على تطوير تطبيق. جوال مرتبط مع كاميرا الداش كام للسائقين من اجل الحصول على البث مباشر للطريق ورصد المخالفات التفحيط، والتجاوز المخالف، وعكس الطريق. هذا النظام قائم على جزئين متتابعين لتحليل مقاطع الفيديو؛ الأول وهو رصد المركبات والثاني هو رصد المخالفات المرورية. تم تنفيذ النظام باستخدام البيانات التي جمعها فريق رصد وقد حقق دقة تصل إلى٩٨٪.
تظهر هذه النتيجة أن المنهجية المستخدمة قابلة للتطبيق في الحياة الواقعية.

Keywords: Traffic violations, dashcam, RTSP link, Neural Network, Readymade report, Drifting, Driving in the opposite direction, Overtaking, YOLO, Computer Vision.

1 Introduction

One of the main pillars of the 2030 vision of the Kingdom of Saudi Arabia is to transform society into a vibrant one to increase the quality of life. A safe and healthy environment will impact all community members, including Saudi citizens and residents. According to the 2030 vision strategy, enhancing traffic safety is one of the main objectives that has priority among the others as it directly impacts the quality of life and public health.

Saudi Arabia has put a lot of effort into the transportation field to decrease the number of accidents due to violations of traffic rules. According to the latest statistical release by the General Authority for Statistics, “Traffic Accident by region”, the number of accidents almost reached 300,000 [54], meaning that there is still a problem despite all of these efforts. The General Department of Traffic (GDT) is incapable of mentoring all the roads in Saudi Arabia, especially at rush hours to ensure safe transportation for people, which demonstrates the need for citizen collaboration. There is a real-life situation where a driver encountered a drifting violation caused by another driver. This put the lives of the people on that road at risk and violated one of the traffic rules. The driver called 911 to file an official violation report against the drifter, even though the action of using a mobile phone while driving is a violation in itself, but it was the only option at the moment. To complicate the situation further, the aggrieved driver was unable to describe the situation to authorities, who asked whether there was a video recording of the violation; unfortunately, there was none. An official report was filed, but with missing information. This situation could have been avoided if there had been a well-known automated system that detected this type of violation. Furthermore, the process of reporting violations can be troublesome, as drivers need to film the violating car with their phone, which is considered another violation. Also, there is no standard way to report a violation, as some drivers are using Twitter, 'كلنا أمن' or calling 911, which confuses the driver due to having various means to file the report, and this may lead them to drop the thought of reporting the violation completely. Therefore, the driver's demand for a system that can recognize, record, and automatically generate a violation report to be sent to the proper authorities for traffic violations such as drifting, driving in the opposite direction, and overtaking is increasing.

Saudi Arabia is one of the developed countries that has addressed the problem highlighted in the preceding section and has already implemented some solutions to control it using surveillance systems and CCTV cameras. With the number of traffic violations going up every year, we noticed that the deployed solutions encounter some barriers, such as the inability to detect attempted violations on highways with no officer points or when the violations occur outside the range of the surveillance system.

Providing a platform that is linked to the driver's dashcam where computer vision libraries can be utilized to process the records received from the dashcam in order to detect the violation and overcome these barriers. As Figure 1 shows, the process of detecting the violation from the dashcam stream will be as follows:

1. Detect the nearby object:

   We will use the recording to recognize nearby objects, whether they are vehicles or not.

2. Determine the occurrence of the violation:

   If the object has been successfully identified as a vehicle, it must be classified as either violating or not violating traffic rules (violation or not violation).

3. Send the violation video to the application:

   After determining the occurrence of a violation, the application will receive the video that contains the violation and display it in the application for the driver.

4. The driver confirms the violation in the video:

   The driver then has the choice to confirm/reject the video. Once the video is confirmed, the application will generate a readymade report that contains the driver’s information, the violation video, time, date, violation type, and any additional information added by the driver. Furthermore, the driver will have the ability to share an email with a prefilled email address, subject, and body with the report as a PDF attachment so that it can be sent to the traffic authorities to take the proper action. In addition, the driver rejection on violation video will be used to enhance the machine learning model.

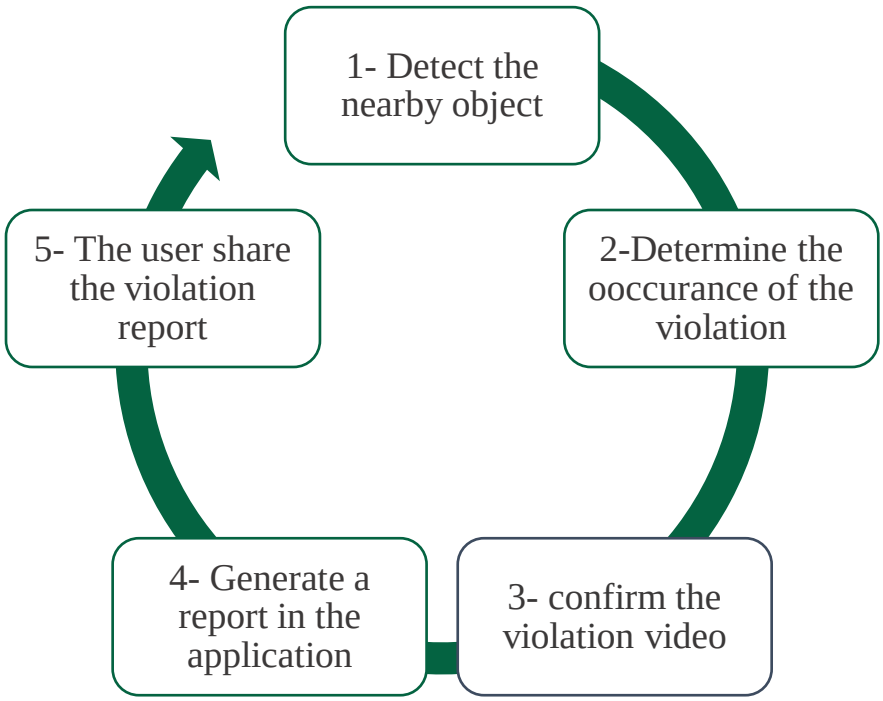


Figure 1 "RASD" Process Cycle

By providing such a platform, collaboration between authorities and drivers can be achieved. The drivers will have clear evidence of the violations automatically, as they will no longer need to use their phones to film them. This will preserve both the driver's time and rights and ease the process of reporting violations since the platform offers a readymade report to be sent to the authorities. If there are multiple platforms for reporting traffic violations, it can lead to inconsistencies in the data that is reported. This is because different platforms may have different reporting procedures or may collect different types of data, which can make it difficult to compare and analyze the information. It will be a more suitable and convenient method for both drivers and authorities.

The authorities will benefit from this platform by having a detection system embedded in most of the drivers' vehicles, which will help them cover different areas that they cannot reach all the time. As more areas are covered, the number of violations will decrease.

"RASD" will benefit drivers with regular dashcams in their vehicles by providing an application that is linked to the driver's dashcam to:

- Register for the driver's profile.
- Browse pending violation videos and confirmed violation reports.
- View the detected violation such as: drifting, driving in the opposite direction, and overtaking after it was processed by "RASD" violation detection system.
- Send the detected violation videos to "RASD" application.
- Confirm the video and generate a report by attaching the driver's information, time and date of the violation, violation types, the violation video URL, and any additional information provided by the driver.
- Send the report to the authorities.

To achieve "RASD" objectives, the listed activities should be associated with it:

- Create a database to save driver information and videos of misclassified violations.
- Build the application using the Flutter platform.
- Drivers can view/delete/edit the confirmed reports and pending violations videos.
- Drivers can confirm pending violation videos.
- Drivers can only link/edit one dashcam to "RASD".
- Drivers can view the stream coming from the dashcam in "RASD".
- Detects only violations in Saudi Arabia.
- Support English and Arabic language only.
- Use the services of firebase's cloud storage for each driver to store the recording coming from their dashcam and upload the violation video after model processing.
- Use OpenCV to obtain video frames and process them, and then train the model using MobileNetv2 (which implements CNN architecture) in a Google Colab environment to detect traffic violations such as drifting, driving in the opposite direction, and overtaking.

- Test the model by processing only the frames that include vehicles, which is done by feeding them to the YOLOv5 small object detection model.
- Deploy the model to Heroku servers.

For drivers with a dashcam in their vehicle who need to report traffic violation such as: drifting, driving in the opposite direction, and overtaking traffic violations in Saudi Arabia, “RASD” is a mobile application that detects and reports traffic violations in real time.

Unlike regular traffic violation detection systems, “RASD” will detect traffic violations using the stream coming from the dashcam and generate reports of them.

“RASD” implementation started by conducting interviews and distributing questionnaires to the drivers in Saudi Arabia, followed by analyzing the results to understand and determine the users' needs. According to the user's needs, an application was developed that supports two languages: Arabic and English. After collecting traffic violation and normal traffic videos and having human annotators annotate them, we did some additional preprocessing on the collected videos by clipping only the violation and normal scenes from the video and labeling them as violation and not violation, respectively. Then we use OpenCV to obtain video frames and process them to train the model using MobileNetv2 (which implements CNN architecture) to detect traffic violations such as drifting, driving in the opposite direction, and overtaking. Furthermore, to deploy the constructed model, Heroku servers were used. After integrating the software with the dashcam and completing the construction of the traffic violation detection model, we performed user acceptance testing with 21 drivers who satisfied the criteria defined for “RASD” system users. The objective was to assess the performance and efficiency of the overall system. The drivers who participated in the testing demonstrated a high level of enthusiasm for “RASD” and acknowledged its importance.

This project makes significant contributions to the fields of technology and research. Firstly, it offers a public dataset that contains more than 348 labeled videos of positive road violations and 348 videos of normal scenes in Saudi Arabia. The dataset was manually labeled by human annotators with a high level of agreement and is available to the public for further research and technological advancements.

Secondly, the project demonstrates that the operation and maintenance costs of driver's dashcams are lower compared to standard CCTVs. However, it relies on drivers using dashcams in their vehicles to sustain the process. As a result, the development of an intelligent

system to monitor drivers who violate traffic rules could potentially reduce traffic accidents and have a significant economic impact. “RASD” is making a significant impact on reducing road violations at the local level, resulting in several benefits. It can reduce traffic congestion, which can directly enhance people's quality of life, and lower road injuries and fatalities. Additionally, it can decrease the cost of public property damage caused by road accidents.

Moreover, the “RASD” project has a global impact by utilizing the latest advancements in artificial intelligence and computer vision to increase road safety. The integration of such technologies can significantly enhance the accuracy and speed of road violation detection, leading to a reduction in road accidents worldwide. “RASD” is a novel solution, as there is no similar system that combines both a computer vision system and a dashcam in order to automate the process of detecting traffic violation such as: drifting, driving in the opposite direction, and overtaking, and generate a ready-made report of the violation.

In this report, we will first introduce the background section that describes the required domain knowledge and the literature review that contains the competitive product analysis. Following that, the system design and development section includes our methodology, system requirements, which contain system users, requirement elicitation and analysis, and user interactions, as well as a road map and our user stories in the product backlog for both functional and non-functional requirements. Then, in the system design section, we will show the architectural diagram, class diagram, and component level design Following that, we will demonstrate our data design, which includes data models, data collection and preparation, interface design and implementation, and system evaluation, which includes experimental results, user acceptance testing, participant demographics, and questionnaire results. Also, we will measure the quality attribute using NFR testing and then discuss the results. Then came the conclusion and future work, furthermore, we have the acknowledgements for those who contribute to “RASD” success. Last, we will include our references and the appendices for the interviews, questionnaires, annotation guidelines, user acceptance testing and UX guidelines.

2 Background

In this section, we will describe in detail the necessary domain knowledge and theoretical background required for understanding the project.

2.1 The traffic law in Saudi Arabia

Our domain is traffic violations in Saudi Arabia. The GDT is responsible for setting procedures and regulations for traffic in order to minimize violations.

Traffic violations occur when drivers violate laws that regulate vehicle operation on streets and highways[3]. Even though the GDT puts effort into publishing procedures and regulations for traffic, they are still being dismissed by reckless drivers as they commit some of the most dangerous traffic violations, such as: driving in the opposite direction, overtaking, and drifting, where driving in the opposite direction, is the act of driving a motor vehicle against the direction of traffic[5]. Overtaking or passing is the act of one vehicle going past another slower moving vehicle, traveling in the same direction, on a road[6]. Drifting is a driving maneuver where the driver of a car intentionally steers too much, causing the car's rear tires, or sometimes all the tires, to lose their grip on the road[7].

In this project, we aim to detect and reliably classify road violations caused by drivers from front dashcam. The initial detection outcome is to classify whether the recorded scene contains a violated rule. To do so, we aim to design the classification model to capture the surrounding vehicles and then classify any driving violation that is violate one of these common three driving rules:

1- Drifting: the act of when a driver purposefully steers too much, causing the back tires, or sometimes all tires, to lose grip on the road[50] (Figure 2).

2- Driving in the opposite direction: the act of driving a vehicle against the direction of traffic[51] (Figure 3).

3- Overtaking: the act of passing a slower moving vehicle from any side without maintaining a safe distance. (Figure 4).

Figure 2 Drifting

Figure 3 Driving in the opposite direction

Figure 4 Overtaking

2.2 Vehicle dash camera (dashcam)

Figure 5 An example of a front dashcam

Recently, drivers have begun embedding dashcams in their vehicles to have solid evidence of any violation that may occur while they are driving outside the range of the authority's surveillance systems. Dashcam cameras refer to an onboard camera that continuously records the view through a vehicle's front windscreen and sometimes rear or other windows[8].

2.2.1 Types of Dashcam

1- Front Dashcam: These dashcams typically mount on the front windscreen or dashboard, facing forward, and record everything in view.

2- Rear Dashcam: A camera that mounts at the back of the vehicle to provide coverage of all events happening behind the vehicle.

3- Side-Dashcam: A camera mounted on the vehicle's left and right sides.

2.2.2 The current Dashcam capabilities

Dashcams have a lot of capabilities, some of which are:

- GPS: Save the location of the vehicle.
- Speed Camera Alerts: notify drivers of speed camera locations.
- G-sensor: Sense the surroundings of the car.
- Protection for when the vehicle is parked: The dashcams only record video when they sense movement when the vehicle is off.
- Software: dashcams come with software to view the recorded videos and a map of the area where the videos were recorded.

2.3 Computer vision models

Mainly, dashcams are for recording purposes, but they can be enhanced with AI technologies to be more functional and smarter. By using AI and computer vision models with dashcams to detect objects and identify traffic violations in real time, this is done by making use of machine learning models and libraries such as: OpenCV, MobileNetv2, and YOLOv5.

A machine learning model is a file that has been trained to recognize certain types of patterns[9].

OpenCV (Open-Source Computer Vision Library) is an open-source computer vision and machine learning software library that uses deep neural networks (DNN), which are a powerful category of machine learning algorithms implemented by stacking layers of neural networks along the depth and width of smaller architectures[10]. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in commercial products[11]. OpenCV is a highly optimized library with a focus on real-time applications[12].

MobileNet-v2 is a convolutional neural network that is 53 layers deep. You can load a pretrained version of the network trained on more than a million images from the ImageNet database. The pretrained network can classify images into 1000 object categories, such as keyboard, mouse, pencil, and many animals[13].

YOLOv5 is a model in the You Only Look Once (YOLO) family of computer vision models. YOLOv5 is commonly used for detecting objects. YOLOv5 comes in four main versions: small (s), medium (m), large (l), and extra-large (x), each offering progressively higher accuracy rates. Each variant also takes a different amount of time to train[14].

2.4 RTSP link

To simulate a vehicle's dashcam, we will use an IP camera to obtain a stream of videos and use it as the model's input by using the RTSP (Real Time Streaming Protocol) URL that requires the username and password specified by the driver and the IP address of the camera as an input.

The Real-Time Streaming Protocol (RTSP) is a tried-and-true video technology. It's used to control audio/video transmission between two endpoints and facilitate the transportation of low-latency streaming content across the internet [15].

3 Literature Review

The detection of traffic violations from a variance camera view is not a well-known problem. Hence, in this section, we will discuss the similar approaches that have been used in the past that can be adapted to represent this situation.

- **Similarities Across the papers:**

Real-time violence detection: In[16], the author proposed a comparison between four different models, which are: MobileNet, which includes 30 layers with convolutional layers with stride 2, AlexNet, which consists of 5 convolutional layers and 2 fully connected layers with ReLu activation function, VGG-16, which consists of 13 convolutional and 13 max polling layers respectively, and GoogleNet, which has 22 layers. To compare between them, the author used the Hockey Fight dataset and evaluated the performance using a confusion matrix. After evaluating all four models, the author agrees that the best model for real-time violence detection is using a MobileNet model.

Detection from a moving view: In [17], the authors proposed an unsupervised deep learning approach for traffic accident detection in dashboard-mounted camera videos. Their major novelty is to detect anomalies by predicting the future locations of objects such as cars, bikes, pedestrians, etc. in the field of view of a dashboard-mounted camera on a moving ego-vehicle. Their model relies on learning from the massive collections of dashboard-mounted video of normal driving without any manual labeling. Then they compare the predicted object locations to the actual locations observed in the next few video frames. They hypothesize that anomalous roadway events can be detected by looking for major deviations between the predicted and actual locations because unexpected roadway events, such as cars striking other objects, result in sudden and unexpected changes in an object's speed or position. Finally, to evaluate their method on realistic traffic scenarios, they introduce a new dataset, AnAn Accident Detection (A3D), of on-road abnormal event videos compiled as 1500 video clips from a YouTube channel of dashboard cameras from different cars. includes a total of 128,175 frames (ranging from 23 to 308 frames) at 10 frames per second and is clustered into 18 types of traffic accidents. Each video contains an abnormal traffic event at different temporal locations. Also, they used another publicly available dataset. Experiments showed that the model significantly outperformed published baselines, which had a value of 60.1 in AUC.

Video-based traffic violation detection system: In [19], the authors proposed a Video-based Traffic Violation Detection System. They relied on an algorithm called the background-updating algorithm, which uses wavelet transform on a dynamic background. The system consists of three main functions: first, video loading, which allows the system to open/close the video. Second, detection of violating vehicles, which is obtained by image preprocessing, violation detection, and background updating, Lastly, there is the violation evidence storage function, which allows the user to save it or upload it to the server. The system was built using C++ and OpenCV 2.3.1.

Computer vision-based: The authors of [20], propose a novel self-driving car support system that detects vehicle accidents in front of the autonomous car and reports them to medical authorities and traffic police. The system uses two models: the Mask R-CNN framework used to classify vehicle types as car, bus, and truck, and a centroid tracking algorithm to track the detected vehicle. When the model catches an overlap between the vehicles, it calculates speed, acceleration, trajectory, and a change in angle anomaly to determine whether it is an accident or not. The proposed system can recognize vehicular accidents accurately, with a 79.05 percent accident detection rate and 34.44% false alarms. The novelty of the author's framework is magnified due to the inclusion of a number of parameters for assessing the chances of a mishap that have been recorded under different ambient conditions, including sunlight, night, and snow. The author used two datasets: MS COCO Dataset and Custom Dataset for Accidents on Roads.

Research	[16]	[17]	[19]	[20]	RASD
Sensors	Surveillance camera	Autonomous Vehicle camera	Dashcam	CCTV camera	External Dashcam camera
Detecting violation/accident model	MobileNet	unsupervised deep learning	Mask R-CNN	OpenCV	MobileNetv2
Detected violation type	Irregular behavior	Accidents	Accidents	Running red lights, speeding, and vehicle retrogress in real time	Traffic violations such as: drifting, driving in the opposite direction, overtaking,
Type of the report	None	None	Automatic	None	Automatic
Crowdsourcing	None	No	Yes	No	Yes
Data source	Hockey fight dataset	HEV-I A3D	MS COCO YouTube	-	Twitter scraping

Table 1 Comparison Between "RASD " and Other Research

Implementing a mobile application that detects traffic violations and reports them to the authorities goes through multiple phases: collecting data, choosing a model to determine the violation type, and then reporting it to the authorities to take the appropriate action. All these procedures can be implemented in different ways.

The majority of researchers collected their data from CCTV cameras, like in [16] and [20]. while “RASD” gets its data from Twitter. In addition, a variety of algorithms and models have been employed to detect traffic violations; researchers [16] and “RASD” employed the MobileNet model according to its high performance among all the models. Furthermore, many did not implement the idea of automatically reporting a detected violation or accident. However, [19] and “RASD” put it into action. Also, the researchers have agreed on the challenges of applying detection models to videos from a variance camera view. Lastly, none of them take a step into the violation detection field as “RASD” does, such as drifting, driving in the opposite direction, or overtaking.

- **Across Applications:**

- **كلنا أمن :**

‘كلنا أمن’[21] is a mobile application that lets the citizens and residents in Saudi Arabia play the role of a police officer. As they can send a report of any violation by attaching a video, photo, or an audio note.

Figure 6 "كلنا أمن" Logo

- **70mai:**

70mai[22] is a mobile application that is used to connect with 70mai dashcam by WI-FI, after the connection completed it will view the real-time video from the dashcam. Also, it provides a historical record and the ability to download them to the mobile phone.

Figure 7 70mai Logo

- **Dubai police:**

Dubai police[23] is a mobile application that provides the user access to all smart services. It has 5 categories of service: Report services, permit services, traffic services, certification services, and community services. The traffic services include an unknown accident report, reporting vehicle obstruction, traffic violations copy, smart impound, fines inquiry and payment, reissuing accident report, change vehicle color, traffic fines installment, and simple accident report. Unknown accident report enables users to report any vehicle damage caused by an unknown party to allow them to repair their vehicles. Simple accident report enables users to report a traffic accident in order to get a report from the authorities that allows them to review the insurance company to repair their vehicles. Both reports require filling the report manually and specifying the damages and attaching photos of the accident. Also, in both reports, there is a service fee range from 400 to 520, in the case of a simple accident report the individual who caused the accident has to pay the fees. Traffic Violations Copy service enables the users to get a copy of documents that is related to fines such as a copy of the fine or a photo of the radar. The user has to fill up the application request form and pay the service fee in order to get the documents.

Figure 8 Dubai police Logo

- **BlackVue:**

DR750X-2CH Plus[24] is a dashcam that records videos of the front and back of the vehicle at a wide 139-view angle. It is connected to an application called BlackVue App that allows the users to manage their videos and change the settings of the dashcam. The application support GPS logging, which allows to save the location of each video recorded in the dashcam. The dashcam has a built-in impact and motion detection which detect impacts and changes of speed while driving and detects movements in parking mode. The application classifies the videos and tags them as event or parking mode motion or normal so that the users can easily distinguish between normal and detected videos.

Figure 9 BlackVue Logo

Features	كلنا أمن	70 mai	Dubai police- شرطة دبي	Dash Cam Black Vue DR750X-2CH Plus	RASD
Saving videos	No	Yes	No	Yes	Yes
Traffic violation detected	None	None	None	No	Traffic violations such as: tailgating, driving in the opposite direction, overtaking, and drifting
Sending reports to authorities	Yes	No	Yes	No	Yes
Support Arabic language	Yes	No	Yes	No	Yes
Type of the Report	Manually	None	Manually	None	Automated

Table 2 Comparison Between "RASD" and Other Applications

To conduct, **there are no applications that support automated detection for traffic violations.** Thus, “RASD” will provide two models for the drivers the first model is for detecting objects specifically to identify vehicles, the second model will detect and classify whether there was a violation or not. Moreover, there are not many applications that provide reporting traffic violations in Saudi Arabia where a lot of violators are left without punishments. Therefore, “RASD” will provide for drivers in Saudi Arabia automatic reporting application for driving in the opposite direction, overtaking, and drifting traffic violations that support two languages (Arabic and English) in order to serve both citizens and residents.

4 System Design and Development

4.1 Methodology

In “RASD”, we used agile approach to develop the application. The agile approach is defined as an iterative approach to project management and software development that helps deliver the product in valuable increments. The implementation of the agile approach was by walking through four sprints, regularly reassessing and adjusting our plans to stay on track and respond to changing requirements, and finally producing two releases.

The scrum framework consists of three roles: product owner, developer, and scrum master. “RASD” had one product owner who set the vision for “RASD” and its priorities. Also, the product owner was the mediator between the stakeholders and developers. The developers have the necessary skills to satisfy the product vision, adapt to changes, decompose the user stories, provide shippable features, and bring the vision to reality. Lastly, we have the scrum master, who coaches and guides the team to ensure that it adheres to the values and principles of agile methodology, facilitates meetings, and provides feedback.

Five events occurred during the “RASD” life cycle, beginning with the sprint that defined the sprint's duration. Then, during sprint planning, which happens at the beginning of each sprint, we decide which user stories will be worked on in this sprint based on their priority in the product backlog and how they will be worked on and completed.

During the sprint, there is a daily scrum meeting to check the progress and how close we are to the sprint goal, inspect the problems that occurred during the sprint, and replan to fix these problems. Moreover, the sprint review was conducted between “RASD” developers and the product owner to gain feedback, inspect problems, and adapt to changes. Before starting the next sprint, a sprint retrospective meeting is conducted to evaluate the process, the team's performance, and take note of the issues that arose during the sprint so they can be avoided in the next sprint.

The Scrum artifacts of "RASD" consist of three components:

- Product backlog: it shows a prioritized list of features as user stories with acceptance criteria so that we could work on them during sprints.
- Sprint backlog: a chosen list of the user stories from "RASD" product backlog to work on them during the sprint.
- Product increment: This is what "RASD" delivers at the end of each sprint, where we integrate the previous increments with the current sprint increment.

Jira[1] and GitHub[2] were used as tools to achieve "RASD" objectives. Jira is a tool for managing the agile software development process. Jira made it easier for us to manage our project by writing the system requirements, monitoring their development, and recording the progress in the form of meeting notes. On the other hand, we used GitHub as another helping tool where it is a code hosting platform for version control and collaboration [25]. By using GitHub, we were able to simply manage the cooperation of "RASD" application and machine learning model codes, commit new changes among "RASD" team, and view these changes on the source code without any conflicts or missing parts.

[1] Jira: https://2022-1st-gp9.atlassian.net/jira/software/projects/GP/boards/1/backlog
[2] GitHub: https://github.com/ghadeerAbdulmajeed/2022-GP1-Group9

4.2 System Requirements

4.2.1 System Users

We are clarifying in this sub-section who is going to use “RASD” by defining each possible user within the system and their educational level, experience, and technical expertise. We have one user that interacts with “RASD” application: Drivers.

The focus of “RASD” is on drivers in Saudi Arabia who currently possess active driving licenses and have installed dashcams with a front view and **1080P Full HD** resolution, specifically of the type 70mai M300 or Ezviz, within their vehicles.

The targeted drivers should read Arabic or English without any specific educational background.

Interacting with “RASD” does not necessitate a high level of technical knowledge. Lastly, it is required for drivers to have a mobile device that is compatible with the Android operating system, and it must have a stable internet connection in order to download the application and to access and perform the application’s functions.

4.2.2 Requirements Elicitation and Analysis

This section briefly describes the requirement elicitation methods we used to formulate our requirements. First, we conducted interviews via WhatsApp chats, Zoom, and in person with drivers in Saudi Arabia to collect general facts and useful and necessary information about traffic violations and the traffic violations reporting process. Second, we distributed an online questionnaire via Google Forms to reach out to a large number of the target audience. Then, we analyzed these interviews and questionnaires -as shown in Appendix- as follows:

Interviews Findings:

After interviewing four people with different driving experiences in different cities. Appendix A: Interviews showed that all of the interviewees did not have a dashcam camera in their vehicle and did not think to have one because of the variety of dashcam types.

Then we wanted to gain knowledge about the main factors that lead to unsafe transportation in Saudi Arabia to understand how our interviewees define “unsafe transportation”. We observed that half of them agreed that driver’s behavior is the key factor in safe transportation. And others mentioned traffic jams, warming signs, lights on the road, and trucks as the main factors that affect the safety of transportation.

From the interviewee’s perspective, overtaking and tailgating are the most common traffic violations that frustrate them while driving. There are also other violations that irritate Saudi drivers, such as breaking the rules by using bright lights. Furthermore, all of the interviewees, agreed that CCTV cameras are not enough to detect traffic violations as they need to cover all the roads and be distributed more.

Moreover, most interviewees agree that they respond to traffic violations committed against them by discarding them. One, however, said that she would only report if the violation led to an accident. Another one said that he would memorize the plate number and call the authorities.

We conclude from their answers that there is a problem in the reporting process that makes interviewees avoid it, and that will have a negative impact on road safety and allow many violators to go without punishment. Additionally, some interviewees said that they had never filed a violation report before, so they were unfamiliar with the process. This fits with their claim that they usually ignore the violations committed against them. Other interviewees said

that it is difficult to obtain evidence of the violation as well as remember the exact details while reporting.

Eventually, the majority of the interviewees agreed that having a smart dashcam linked to a mobile application will be beneficial in various ways, including saving their time, simplifying the process of reporting and reducing traffic violations, therefore leading to safe transportation.

Questionnaire Findings:

From the responses to the distributed questionnaire in Appendix B: Questionnaire, we gathered 140 responses, the majority of these responses came from a sample of people who are between 18 and 25 years old, with a percentage of 70.7% Figure 86. Also, we noticed that the responses came from both females and males in roughly the same proportion Figure 85.

After understanding the sample, we analyzed their response to a question that was concerned with traffic violations. First of all, 32.1% of the sample selected driving in the opposite direction violation and 29.3% of the sample selected overtaking as the most common traffic violations that they encountered Figure 88.

Furthermore, we can observe that 55% of the sample believes that CCTV cameras and other roadside surveillance systems are insufficient to detect all of the traffic violations Figure 87 which shows the necessity of having additional techniques to detect traffic violations.

In addition, when it comes to dropping the thought of reporting the violation, 66.4% of the sample thinks that they are wasting their time if they report a violation due to the complexity of the currently offered process, while most of the remaining sample drop the thought of reporting a violation for reasons divided between having multiple platforms and the lack of the violated vehicle's information Figure 90.

On top of that, 40.7% of the sample ignored the violated vehicle when they encountered a traffic violation, while 31.4% attempted to memorize the vehicle's information, and 34.3% attempted to record the violator's vehicle using their phones Figure 89. These findings show that trying to report a traffic violation may cause the driver to become distracted from the road and commit another violation.

Finally, the results show that 55% of the sample does not own a dashcam Figure 92, which demonstrates the low popularity of dashcams in Saudi Arabia. However, since 85% of the sample showed interest in having a smart dashcam that can detect traffic violations and allow

them to report them simply and quickly Figure 91, we come to the conclusion that this will motivate them to own a dashcam in the future.

4.2.3 User Interactions

This use case diagram in Figure 10 shows the possible sequence of interactions between "RASD" and its actors, -the users of the application: drivers- related to their goals. The drivers could be any driver in Saudi Arabia with a dashcam in their vehicle. They can register once, so the next time they access the system, they only need to log in. They also have the ability to reset their password, update their information, such as their first name, last name, email, and phone number, link or edit their dashcam information, and log out of the system. However, their main functionalities are to view, delete, and confirm the violation video via viewing the pending violation videos, and delete, edit, and send the confirmed report to the authorities.

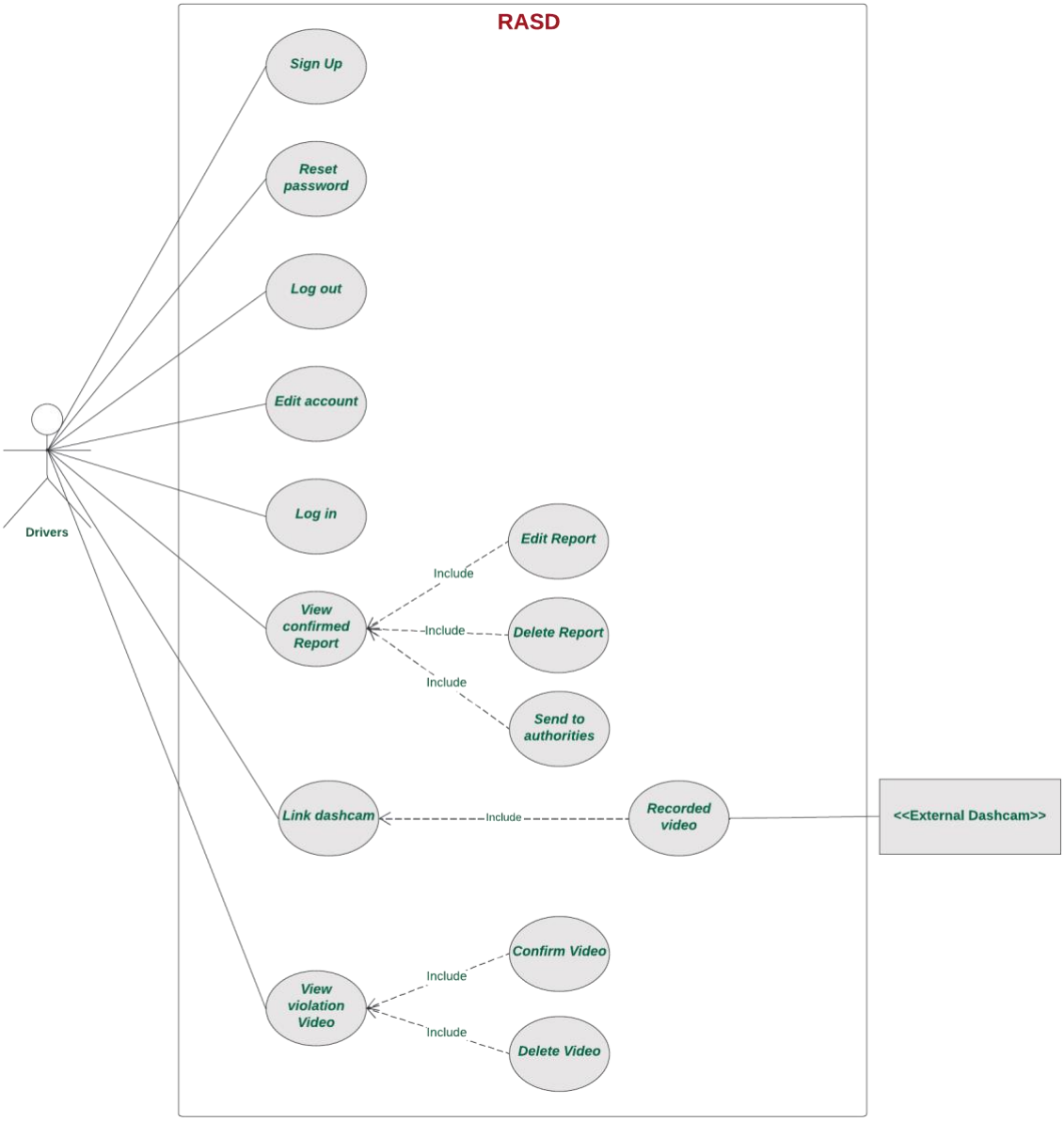


Figure 10 Use Case Diagram

4.2.4 Roadmap and Product Backlog

In this section we are going to represent "RASD" roadmap, and the product backlog to represents the user stories.

- **Roadmap:**

"RASD" roadmap is composed of five sprints, starting with sprint zero, where we have done some research to determine the technology we are going to use and list our features as user stories in the product backlog. Followed by the sprint one, where we have collected, cleaned, and annotated the data. Also, we have designed and implemented the user interfaces, along with the log in, register, and log out functions, and released the first version of "RASD". Furthermore, in sprint two, we trained a model to detect traffic violations such as: drifting, driving in the opposite direction, and overtaking. Then, we reflected on the changes after hearing the committee's comments. Furthermore, in sprint three, we connected the dashcam to the application and the model to Firebase storage in order to obtain the stream videos. Also, we implemented the view, delete, and confirm report functions. Lastly, we tested "RASD" functionalities and released the second version of "RASD".

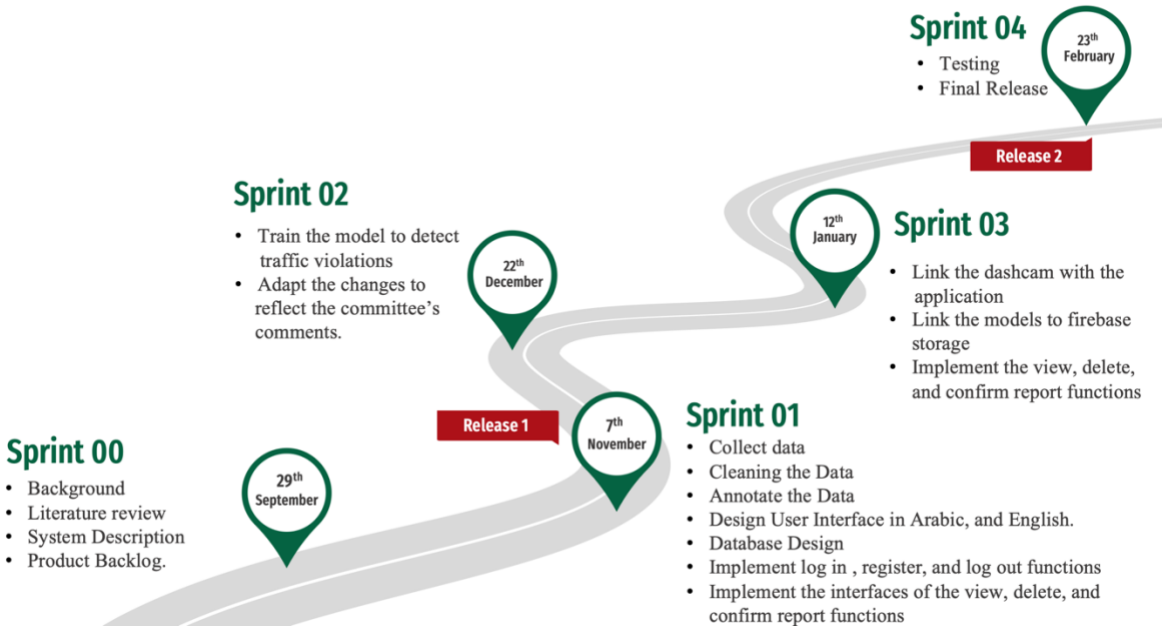


Figure 11 Product Roadmap

- **Product Backlog:**

ID	PBIs	Size	Type	Status	Acceptance Criteria
1	**As** a new driver, **I want to** register to the application **so that** my information will be saved in the application's database.	5	Feature	Done	-**As** a new driver, **If I** enter my first Name, last name, unique email, phone number, and password and agreed to the application's privacy policy to sign up and click on the " Sign Up " button, **then** my information should be saved to the application's database. **If** I enter any of the following information incorrectly: email, phone number and password to sign up or did not agree on the application's privacy policy, **then** signing up fails and a detailed error message will be displayed.
2	**As a** driver, **I want to** log in to the application **so that** it grants me access to preform application functions, which are to view, delete, and confirm the pending violation videos and view, edit, share, and delete the readymade reports, link or edit my dashcam information and modify my personal information.	3	Feature	Done	-**As** a registered driver, **If** I enter my unique email and password to log in and click on the "log in" button, **then** my information should be retrieved from the application's database. **If** I enter either my email or password incorrectly to log in, **then** the log in fails and an error message will be displayed. **If** I log in successfully, **then** I will be redirected to the driver's home page where it will display: A list containing 4 options: - Link or edit the dashcam information. - View and delete pending violation videos. - View, edit, share and delete confirmed reports.

					- “Need Help” button to be redirected to “RASD” Twitter direct message and a bottom app bar that allows drivers to navigate between the home page, the pending violation videos, confirmed reports and the settings of the application.

ID	PBIs	Size	Type	Status	Acceptance Criteria
3	**As a** driver, **I want** to sign out of the application **so that** I can ensure no one else can access the application functions, which are to view, delete, and confirm the pending violation videos and view, edit, share, and delete the readymade reports, link or edit my dashcam information and modify my personal information.	5	Feature	Done	-**As** a registered driver, **If** I was in the settings page and click on the "Sign out" button, **then** a system confirmation dialog will be displayed confirming that I want to sign out. **If** I click on the "Sign out" button, **then** the system should prevent me from performing any application functions. **If** I click on the "Sign out" button, **then** the system should redirect me to the application's log in page.
4	**As** a driver who speaks Arabic, **I want** the application to support Arabic language.	3	Feature	Done	-**As** a new/registered driver, **If** I register to the application, **then** I should be able to choose Arabic language as my preference language. **If** I log into the application, **then** I should be able to change my language preferences. **If** I was in the registration page, **then** I should be able to change the language preferences. **If** I was in the log in page, **then** I should be able to change the language preferences.
5	**As** a driver who speaks English, **I want** the application to support English language.	3	Feature	Done	-**As** a new/registered driver, **If** I register to the application, **then** I should be able to choose English language as my preference language.

					If I log into the application, **then** I should be able to change my language preferences. **If** I was in the registration page, **then** I should be able to change the language preferences. **If** I was in the log in page, **then** I should be able to change the language preferences.

ID	PBIs	Size	Type	Status	Acceptance Criteria
6	**As a** driver I want to see a link button in the home page **so that** I can link **my** dashcam information to the model through the application.	3	Feature	Done	-**As** a new/registered driver, **If** I was on the application's homepage, **then** I should be able to see linking dashcam button. **If** I click on the "link" button, **then** I should be redirected to the linking dashcam page. **If** I was in the linking dashcam page and entered the dashcam IP or the dashcam with username and password option, **then** my dashcam information should be saved to the application's database. **If** I click on "link" button, **then** a confirmation dialog should appear showing me the stream video coming from the dashcam to confirm the linking. **If** I confirmed that the dashcam stream is obtained and it was actually obtained, **then** a successful dialog should appear indicating that my dashcam is linked successfully. **If** I confirmed that the dashcam stream is obtained and it was not really obtained, **then** an error message should appear indicating that the stream is not obtained, and I should rewrite my dashcam information or check my internet connection.

					If I was in the linking dashcam page and entered the dashcam IP with incorrect format or entered the dashcam username and password incorrectly, **then** the linking fails, and a detailed error message will be displayed.

ID	PBIs	Size	Type	Status	Acceptance Criteria
7	**As a** driver, **I want to** edit my linked dashcam information **so that** the application can receive the video stream from the new dashcam.	3	Feature	Done	-**As** a registered driver, **If** I was on the application's homepage, **then** I should be able to see editing dashcam button. **If** I click on the "edit" button, **then** I should be redirected to the editing dashcam page. **If** I was in the editing dashcam page and entered my new dashcam IP or the dashcam with username and password option, **then** my dashcam information should be updated in the application's database. **If** I click on "Edit" button, **then** a confirmation dialog should appear showing me the stream video coming from the dashcam to confirm the linking. **If** I confirmed that the dashcam stream is obtained and it was actually obtained, **then** a successful dialog should appear indicating that my dashcam is linked successfully. **If** I confirmed that the dashcam stream is obtained and it was not really obtained, **then** an error message should appear indicating that the stream is not obtained, and I should rewrite my dashcam information and check my internet connection.

					If I was in the editing dashcam page and entered the dashcam IP with incorrect format or entered the dashcam username and password incorrectly, **then** the linking fails, and a detailed error message will be displayed.
8	**As a** driver, **I want to** be able to reset my password **so that** I can still be able to access my account if I accidently forget the password.	5	Feature	Done	- **As** a registered driver, **If I** was in the log in page, and clicked on the "forgot password?" link, **then** it should redirect me to the reset password page. **If** I was in the reset password page and entered my email and clicked on the "reset password" button, **then** a reset password link will be sent to my email.

ID	PBIs	Size	Type	Status	Acceptance Criteria
9	**As a** driver, **I want to** be able to view confirmed violation reports **so that** I can view, edit, share and delete the confirmed report.	8	Feature	Done	-**As** a registered driver, **If** I was on the confirmed report page, **then** I should see a list of confirmed reports if any. **If** I click on a confirmed report from the list, **then** I will be redirected to the preview report page and a "delete", "edit" and "share" buttons will be shown.
10	**As a** driver**, I want to** be able to edit the confirmed report **so that** I can modify the report's information.	8	Feature	Done	-**As** a registered driver, **If** I was in the preview report page, and clicked on the "edit" button, **then** I will be redirected to the edit report page. **If** I was in the edit report page, **then** I should have the report's current information displayed. **If** I edit the report information, click on "edit" button and click on the confirm button, **then** all the report's information should be updated in the application's database. **If** I edit the report information and clicked on "edit" button, **then** I will be redirected to the confirmed reports page.
11	**As a** driver, **I want to** be able to view pending violation videos **so that** I could verify if the model classified the violation video correctly or not. Also, specify the	5	Feature	Done	-**As** a registered driver, **If** I was on the pending violation page, **then** I should see a list of pending violation videos if any. **If** I click on a pending violation video, **then** I will be redirected to the violation

	violations type in the video if its indeed a violation, and add additional information about the violation if any.				video page and I will be able to view the video, along with a radio button asking if the video is correctly classified as violation “yes” or not “no”. **If** I click on the “yes” radio button, **then** a checkbox list of violation types and a textbox for additional information will appear. Along with a “confirm” and “delete” buttons. **If** I click on the “no” radio button, **then** a “delete” button will appear.

ID	PBIs	Size	Type	Status	Acceptance Criteria
12	**As a** driver, **I want to** confirm a pending violation video **so that** a confirmed violation report will be generated.	3	Feature	Done	**As** a registered driver, **If** I was in the violation video page and click on the "yes" radio button and click on the confirm button, **then** the status of the video will be changed to "confirmed" in the application's database. **If** I confirmed a pending violation video, **then** a confirmed report will be generated, and I will be redirected to the confirmed report page where all the previously confirmed reports are listed.
13	**As a driver I want to** be able to send the confirmed report to the authorities **so that** they can take the appropriate action.	1	Feature	Done	-**As** a registered driver, **If** I was in the preview report page and click on the "Send to the authorities via email" share button and my phone number was verified, **then** a share dialog will be seen. **If** I click on any email application, **then** I will be redirected to that email application and a pre-filed email will be shown. **If** I send the email, **then** I will be redirected back to the preview report page. **If** I was in the preview report page and click on the "Send to the authorities via email" share button and I have not verified my phone number yet, **then** a dialog will be shown informing that I

					must verify my phone number to be able to share the confirmed report with the authorities.

ID	PBIs	Size	Type	Status	Acceptance Criteria
14	**As a** driver, **I want to** delete the previously confirmed reports **so that** I can get rid of unwanted report.	3	Feature	Done	-**As** a registered driver, **If** I was in the preview report page, and click on the "delete" button, **then** a confirmation message will be displayed to ensure if I want to delete the report or not. **If** I confirmed the deletion, **then** the report should be removed from the confirmed report's page, and I should receive a success message confirming that the report has been deleted. **If** the report was deleted from the confirmed reports page, **then** it should also be removed from the application's database.
15	**As a** driver, **I want to** delete pending violation video **so that** I can get rid of misclassified violations.	5	Feature	Done	-**As** a registered driver, **If** I was in the violation video page and click on the "yes" or "no" radio buttons and click on the delete button, **then** a confirmation message will be displayed to ensure if I want to delete the detected violation video or not. **If** I confirmed the deletion, **then** the detected violation video should be removed from the list of pending violation videos, and I should receive a success message confirming that the detected violation video has been deleted. **If** the detected violation video was deleted from the pending violation videos list, **then** it should also be removed from the application's database.

ID	PBIs	Size	Type	Status	Acceptance Criteria
16	**As a** driver, **I want to** use my dashcam that is linked to the application **so that** the application can drive benefit from incoming video stream to detect drifting, driving in the opposite direction and overtaking violations.	8	Feature	Done	-**As** a registered driver, **If I** use my linked dashcam, **then** it will start send stream videos to the application to be able to detect drifting, driving in the opposite direction and overtaking.
17	When the object is Infront of the vehicle the object detection model must determine whether it is a vehicle or not.	8	Feature	Done	**If** the object detection model detects a vehicle object in a video frame, **then** the frame should be passed to the violation detection model for further processing.
18	When the object detection model detects a vehicle, the traffic violation classification model should classify if it is a violation or not.	8	Feature	Done	**If** the model detects a violation, **then** the detection accuracy should be at least 75% since similar violation detection model[27] have approximate result.

Table 3 Product Backlog for Functional Requirements

The majority of the user stories have been altered to reflect the actual implementation and since no major changes done to these user story, we did not consider them as deleted ones. Only one of the user stories was deleted which was "As a driver, I want to view the violation video so that I can verify the legitimacy of the violation" the user story and the corresponding acceptance criteria considered redundant to user story #7 from Table 3 which was one of the altered user stories.

- **Non-Functional Requirements:**

ID	PBIs (User Stories)	Size	Type	Status	Acceptance Criteria
1	**As a** driver, **I want** my password to be concealed during register/login **so that** I can assure no one else knows it.	2	Feature	Done	-**As** a new/registered driver, **If** I start typing the password while logging in, **then** the password must be shown as dots to ensure security. **If** I enter my password in the specific text box while logging in or registering, **then** the password must not be written as plain text.
2	**As a** driver, **I want** my sessions to be inactivated and destroyed after I log out **so that** no one can exploit my previous active sessions to enter into the application.	5	Feature	Done	-**As** a registered driver, **If** I successfully logged out from the application, **then** my session should be inactivated and destroyed.
3	**As** a driver, **I want to** be able to see the violation records in range between 2 to 3 minutes while having a stable internet connection **so that** I do not waste time waiting for the record to be uploaded.	3	Feature	Done	-**As** a registered driver, **If** I have the application open and the violation record is clicked, **then** the video should be displayed within 2 to 3 minutes while having a stable internet connection.
4	**As a** driver, **I want** the application to be available 80% of the time while the car is moving to start detecting drifting, driving in the opposite direction and overtaking traffic violations **so that** I can check the violation through the application.	5	Feature	Done	- **As** a registered driver, **If** I was driving and the car was moving, **then** the application should be running 80% of the time to detect any drifting, driving in the opposite direction and overtaking violations on my way.

ID	PBIs (User Stories)	Size	Type	Status	Acceptance Criteria
5	**As a** driver, **I want to** be able to use all the application functions for the first time within 10 minutes **so that** I do not get overwhelmed and drop the thought of using the application to report the violations.	2	Feature	Done	- **As** a new driver, **If I** have the application open, **then** it should take me 10 minutes at most to register and navigate around the application and understand its functions.

Table 4 Product Backlog for Non-Functional Requirements

4.3 System Design

4.3.1 Architectural Diagram

In this sub-section, we will describe our system's organizational style by showing the major subsystems and data repositories and their inter-connections.

Figure 12 represents our system architecture on a high level. First, it starts with driver registration. The input of the system will be obtained from the recording stream coming from the driver's dashcam, these recordings will be saved at the driver's local storage for a short period of time to reduce the probability of video loss that might happen if the videos were directly uploaded to firebase cloud storage. Then it will be fed to the machine learning model where it will detect the occurrence of the vehicle and then passes the frame that contains a vehicle to the violation detection model in order to detects the traffic violation and classify the recorded video if it is a violation or not, after classifying the violation videos and getting the violation confirmation from the driver it will generate a report that contains the violation record along with the driver's information.

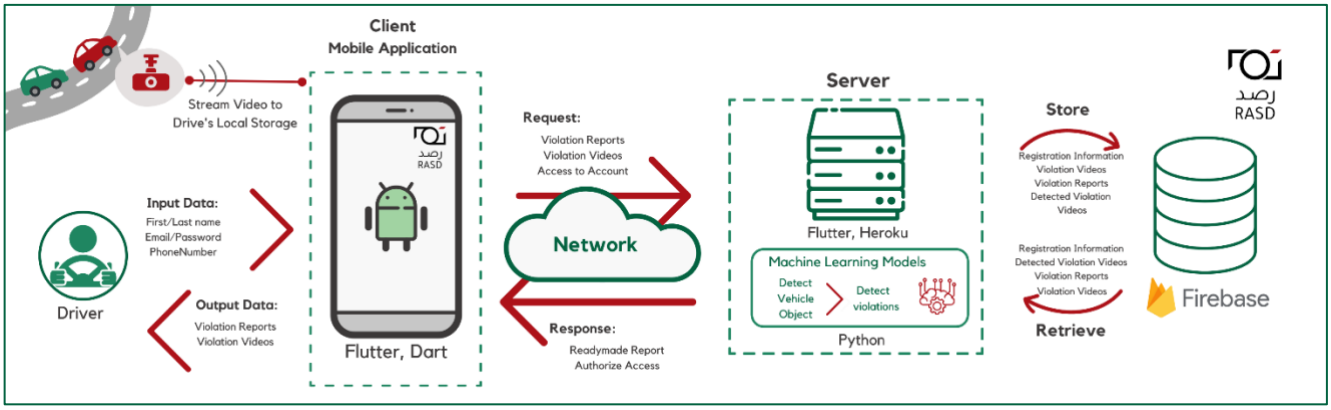


Figure 12 System Architecture

4.3.2 Class Diagram

In this sub section, the organization of "RASD" application is illustrated as a class diagram in Figure 13 .

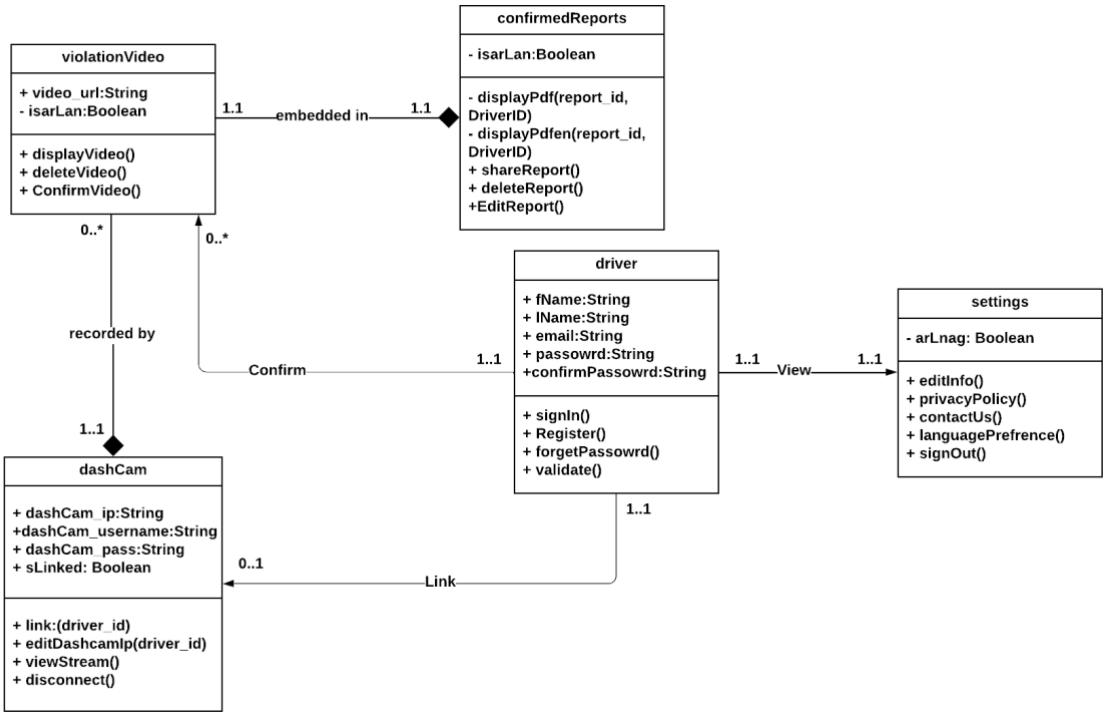


Figure 13 UML Class Diagram

4.3.3 Component Level Design

In this section, we will show detailed modeling of different components of "RASD" system, which are: link dashcam, live stream and detect violation in a single video process, sign up, and confirm a violation video component, with the help of pseudocode and flowchart where we managed to demonstrate the algorithms, we are going to follow the pseudocode and flowchart to design each component.

-Link dashcam:

Sprint number	PBI	User story tasks with effort estimate in hours					Sum of effort
4	**As a** driver I want to see a link button in the home page **so that** I can link **my** dashcam information to the model through the application.	Define link button and link dashcam page style	Develop the interface code for the input fields and button using dart language	Develop the validation code in dart language	Develop the translation code for the Arabic/English interface	Create document in the database for dashcam information	42.5 hours
		15 hours	20 hours	2 hours	1.5 hours	4 hours	

Table 5 User Story for Link Dashcam

Pseudocode
BEGIN **DISPLAY** link page **DISPLAY** 'IP dashcam' **AND** 'IP dashcam with username and password' **IF driver** clicks on 'IP dashcam' option **THEN** **DISPLAY** 'IP address' input field **AND** info tool tip **AND** 'Link' button **READ** 'IP address' input field **WHILE** driver entered invalid input **OR** empty fields **DISPLAY** error message **READ** 'IP address' input field **ENDWHILE** **IF driver** entered a valid 'IP address' **AND** 'Link' button clicked **THEN** **DISPLAY** 'live stream' confirmation dialog **IF** live stream appears **AND** driver click 'Yes' button **AND** 'battery optimization' enabled **THEN** **DISPLAY** 'Disable battery optimization' message **IF** 'Allow' button clicked **THEN** **DISPLAY** 'live stream' in the home page **AND** enable "RASD" application in the background **ENDIF** **IF** 'Deny' button clicked **THEN** **DISPLAY** 'live stream' in the home page **AND** disable "RASD" application in the background **ENDIF** **ENDIF** **IF** live stream does not appear **AND** driver clicks 'yes' **THEN** **DISPLAY** error message **ENDIF** **IF** live stream does not appear **AND** driver clicks 'No' **OR** live stream appears **AND** driver clicks 'yes' **THEN** **DISPLAY** link page **ENDIF**

ENDIF

IF driver clicks on 'IP with username and password' option **THEN**

DISPLAY 'IP address', 'Username', 'Password' input fields **AND** info tool tip **AND** 'Link' button

READ 'IP address', 'Username', 'Password' input fields

WHILE driver entered invalid input **OR** empty fields

DISPLAY error message

READ 'IP address', 'Username', 'Password' input fields

ENDWHILE

IF driver entered a valid 'IP address', 'Username', 'Password' **AND** 'Link' button clicked **THEN**

DISPLAY 'live stream' confirmation dialog

IF live stream appears **AND** driver clicks 'Yes' button **AND** 'battery optimization' enabled **THEN**

DISPLAY 'Disable buttery optimization' message

IF 'Allow' button clicked **THEN**

DISPLAY 'live stream' in the home page **AND** enable "RASD"application in the background

ENDIF

IF 'Deny' button clicked **THEN**

DISPLAY 'live stream' in the home page **AND** disable "RASD"application in the background

ENDIF

ENDIF

IF live stream does not appear **AND** driver clicks 'Yes' **THEN**

DISPLAY error message

ENDIF

ENDIF

Table 6 Pseudocode for Link Dashcam

- **upload live stream to storage:**

Sprint number	PBI	User story tasks with effort estimate in hours					Sum of effort
4	**As a** driver, **I want to** use my dashcam that is linked to the application **so that** the application can drive benefit from incoming video stream to detect driving in the opposite direction, overtaking, and drifting violations.	Define the live stream process	Develop the code that obtain the stream	Develop the code that upload the stream to storage	Define the name of the videos	Create a folder in storage for each driver	75 hours
		10 hours	40 hours	20 hours	2 hours	3 hours	

Table 7 User Story for Dashcam Stream

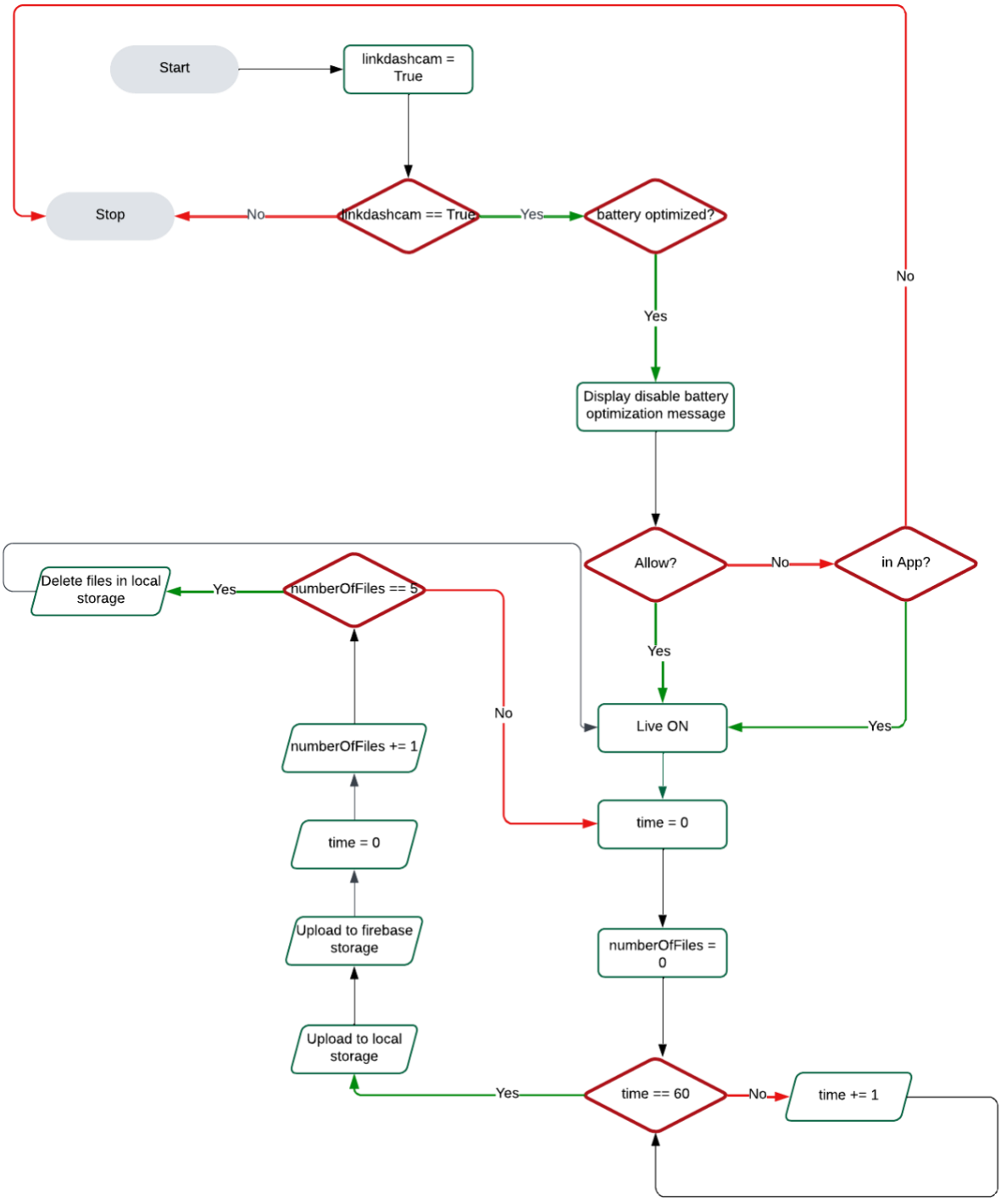


Figure 14 Flowchart for Dashcam Stream

- **single video process:**

Sprint number	PBI	User story tasks with effort estimate in hours					Sum of effort
4	When the object detection model detects a vehicle, the traffic violation classification model should classify if it is a violation or not.	Define the type of model to use	Look for similar approaches	Train and test the model	Try different models and choose the best fit	Upload the model to server	368 hours
		10 hours	50 hours	168 hours	100 hours	40 hours	

Table 8 User Story for violation Detection

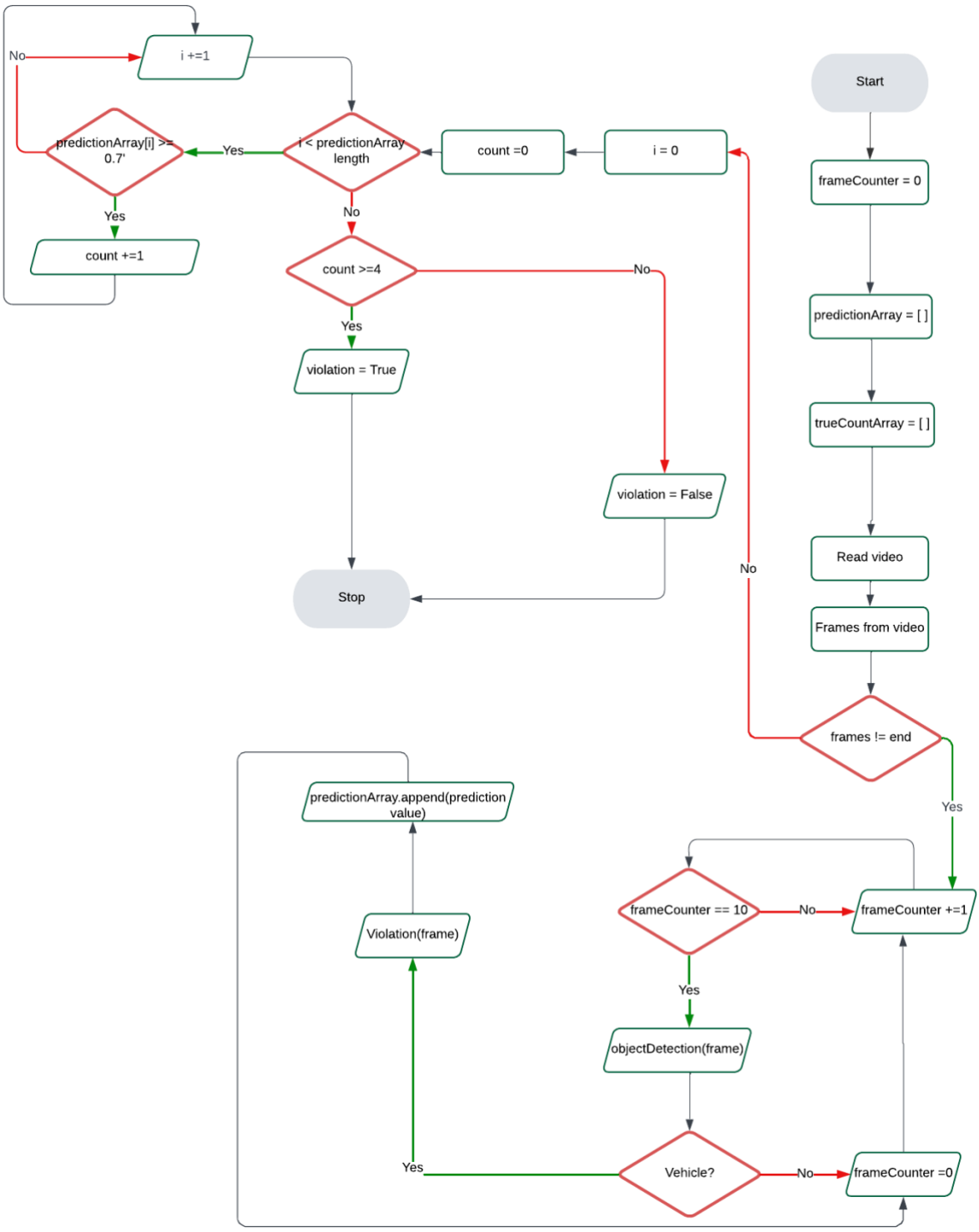


Figure 15 Flowchart for Violation Detection

- **Sign up:**

Sprint number	PBI	User story tasks with effort estimate in hours					Sum of effort
4	**As** a new driver, **I want to** register for the application **so that** my information will be saved in the application's database	Define Register button and register page style	Develop the interface code for the input fields and button using dart language	Develop the validation and verification method code in dart language	Develop the translation code for the Arabic/English interface	Create document in the database for registering information	72 hours
		12 hours	12 hours	24 hours	7 hours	5 hours	

Table 9 User Story for Sign Up

Pseudocode
BEGIN **IF driver** clicks on sign up link on log in page **THEN** **DISPLAY** first name, last name, email, phone number, password, confirm password input fields, privacy policy checkbox **AND** sign-up button **READ** first name, last name, email, phone number, password, confirm password, privacy policy checkbox **WHILE** driver entered invalid input **OR** empty fields **DISPLAY** error message **AND** disable sign up button **READ** first name, last name, email, phone number, password, confirm password **ENDWHILE** **WHILE driver** clicks on sign up button **READ** first name, last name, email, phone number, password, confirm password, confirm privacy policy checkbox **IF** email already exist **THEN** **DISPLAY** 'the email address is already in use by another account' **ENDIF** **IF** empty fields **THEN** **DISPLAY '**Enter your Email' **ENDIF** **IF** privacy policy checkbox not checked **DISPLAY** 'You have to agree on "RASD" privacy policy' **ENDIF** **IF** non empty fields **AND** valid email **AND** confirm privacy policy **THEN** **WRITE** first name, last name, email, phone number, password **AND** **DISPLAY** verification email page **IF** email verified **THEN** **DISPLAY** home page **ENDIF** **ENDIF**

ENDWHILE **ENDIF**

Table 10 Pseudocode for Sign Up

- **Confirm violation video:**

Sprint number	PBI	User story tasks with effort estimate in hours				Sum of effort
1	**As a** driver, **I want to** confirm pending violation video **so that** a confirmed violation report of the violation will be generated.	Define confirm button, delete button and violation video page style	Develop the interface code for violation video page confirm button, and delete button using dart language	Develop the confirmation and success method code in dart language	Create collection in the database for misclassified videos and document for generating report and videos	45 hours
		15 hours	24 hours	4 hours	2 hours	

Table 11 User Story for Confirm violation Video

Pseudocode

BEGIN

DISPLAY pending violation videos page

DISPLAY pending violation videos list

IF driver click on violation video from the list **THEN**

DISPLAY violation video page

DISPLAY video

DISPLAY radio button to confirm violation

IF driver clicks on not violation **THEN**

DISPLAY Delete button only

IF driver clicks on Delete **THEN**

DISPLAY confirmation message

IF driver clicks on confirm **THEN**

DISPLAY success message

ENDIF

ENDIF

ENDIF

IF driver clicks on it is violation **THEN**

DISPLAY violation video types checkboxes **AND** additional information text input field

IF driver click on confirm **THEN**

DISPLAY confirmation message

IF driver clicks on confirm **THEN**

DISPLAY success message **AND** generate report

DISPLAY confirmed reports page

ENDIF

ENDIF

IF driver click on delete **THEN**

DISPLAY confirmation message

```
        IF driver clicks on confirm THEN
            DISPLAY success message
        ENDIF
    ENDIF
ENDIF
```

Table 12 Pseudocode for Confirm Violation Video

4.4 Data Design

4.4.1 Data Models

This subsection describes the data structure of “RASD” application using an ER diagram, an embedded document, and a tree structure.

- **ER Diagram:**

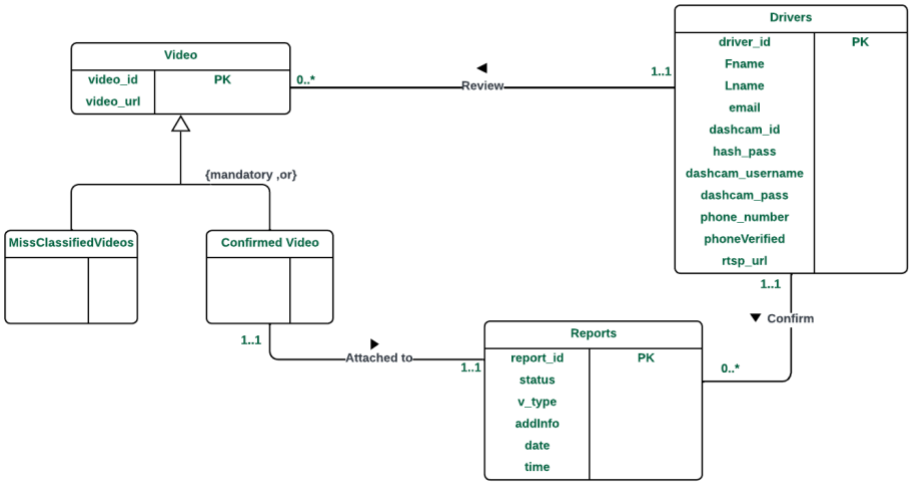


Figure 16 ER Diagram

- **Document based data model (the collections Tree-like structure):**

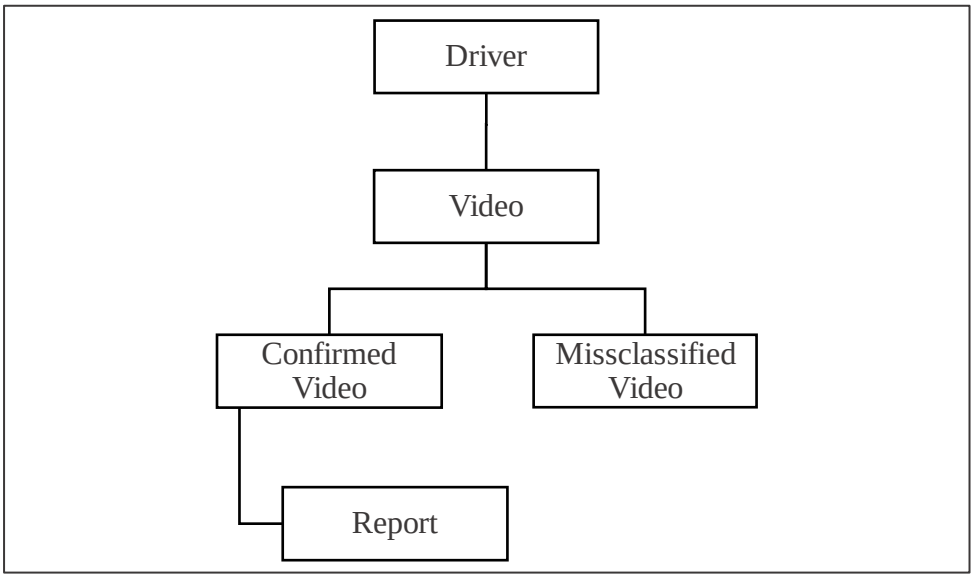


Figure 17 Tree Structure

- **Embedded document:**

Database: RASD

Collection: Driver Document

```
{   Driver_id: “”
    Fname:””
    Lname:””
    email:””
    dashcam_id:””
    hashe_pass:””
    dashcam_username:””
   dashcam_pass:””
   phone_number:””
  phoneVerfied:””
  rtsp_url:””
    Video: [{
                Missclassified {
                Video_url:””
                Video_id:””
  },
                Confirmed {
                Video_url:””
                Video_id:””
                    Reports{
                    V_type:””
                    Status:””
                    Report_id:””
                    addInfo:””
                    date:””
                    time:””
        }
}
       }]
}
```

Figure 18 Embedded Document

4.4.2 Data Collection and Preparation

This section will cover the data collection method we followed as well as the preprocessing steps we performed.

To begin, we collected our data from several Twitter accounts: @Dashcamksa1 @eMoroor and hashtags: #داش_كام #إرهاب الشوارع #صيد الداش كام #داش كام_السعوديه. Since Twitter is the main source of our desired data, we have collected the data by applying Twitter scraping. Our main objective was to scrape tweets with video content and discard any other tweets from the mentioned accounts/hashtags that did not include a video in them. Then, we started scrapping each account and each hashtag individually. The result from each scraping procedure was a folder that included all the video files in that account as an MP4 and one CSV file with 13 columns. Twelve of them were unrelated tweets, and only one column was taken, the Media URL column. Since the scrapping method scraped every tweet in the account/hashtag, the resulting folders needed to be cleaned, and the cleaning process went through two main steps:

1. Clean the folder by deleting any video that is not related to traffic or vehicles in general.

2. Cleaning the folder of any traffic video that is not within our scope, which covers tailgating violations only.

While we were cleaning the data with the aim of having over 1,000 tailgating violation videos, we noticed that the tailgating violation videos were few compared to the other violations in the collected data. One of the folders included 400 violation videos, and when we cleaned that folder, we only got 30 tailgating violation videos, which makes us reconsider the type of violation since the lack of tailgating violation videos could be a major problem. This is because the videos are meant to be used in building a machine-learning model.

Due to the mentioned problem, we have come up with a solution where we are going to build a machine learning model that will classify the video as to whether it contains a traffic violation

or not instead of classifying the violation type and deciding whether it was a tailgating violation or not.

Due to a lack of tailgating violation videos and the scope expansion, we went through another cleaning process in which we repeated Step 2 of cleaning and focused on all traffic violations. We noticed that the most frequent types of traffic violations are as follows:

1- Driving in the opposite direction

2- Overtaking

3- Drifting

Furthermore, after repeating data cleaning processes, we started counting how many violation videos we got, and it turns out that we have a total of 1,176 traffic violation videos.

To ensure that every video in the dataset is correctly classified, we distributed six Microsoft Forms. Each form is sent to three different people and contains approximately 200 traffic videos from the collected data that will be annotated as (violation, not violation, or not related).

To complete the annotation process, the annotators were selected based on the defined criteria and guidelines in Appendix C: Annotation Criteria and Guidelines.

Since we have three annotators and three options in each form, there was a possibility to have a tie in the selected options where each annotator selects a different option:

(annoutater1: violation) (annoutater2: non-violation) (annoutater3: not-related).

In such a case, a fourth annotator is used as a tiebreaker.

What we concluded after the completion of the data annotation process is that:

Out of 1,176 videos, and before breaking the tie cases, 1,131 videos were annotated as violations, 12 videos were annotated as not violations, and 14 videos were annotated as not related. Moreover, there were 19 tie cases in the whole dataset, and after breaking these ties by a fourth annotator, the number of violation videos increased to 1,143. In addition, the number of non-violation videos increased to 17 videos. Finally, the number of unrelated videos reached 16.

After conducting research, we determined that Inter-rater Reliability (IRR) is the most suitable metric for evaluating the reliability of our annotation process[53]. IRR measures the level of agreement among raters or judges, with a score of 1 (or 100%) indicating unanimous agreement and 0 (or 0%) indicating complete disagreement. The methods for measuring IRR vary depending on the type of data and the number of raters involved. For the "RASD" dataset annotation process, we had multiple raters, and we utilized the Percent Agreement for Multiple Raters method to calculate IRR. Our evaluation yielded an inter-rater reliability score of 76%. The annotated data were collected and annotated for the purpose of training "RASD" machine-learning model. Unfortunately, as the numbers show, we have unbalanced data, so we had to go through the data collection process all over again in order to collect normal road scenes (non-violation videos). To overcome this, we had to collect normal road videos ourselves via installing dashcams in two of "RASD" team members' cars. Then, have them drive around different roads in Riyadh and record the necessary normal videos as they carefully select the roads that include neighborhoods, main roads, branch roads, roads at rush hour, roads during the day and night, and so on, in order to add diversity to the normal scene that will be fed to the model. After recording almost 288 videos along with the previously annotated videos as not-violation videos, which were 17 videos, we obtained 305 not-violation videos in total.

The primary goal of data collection and pre-processing is to train "RASD" machine-learning model that, theoretically, should be able to detect drifting, overtaking, driving in the opposite direction, and tailgating traffic violations after being trained over the collected data (violation videos).

In reality, it appears that we need to go through a third preprocessing phase, where the 1,176 annotated violation videos were of varying lengths and not all of them were of great quality, and some had stickers placed on them by the person who uploaded the video.

Moreover, some of the videos included two scenes where the driver had two dashcams in their vehicle (front windscreen and rear window), which cannot be used as part of the training dataset since they are going to disturb the training phase and are considered noisy data.

We start the preprocessing phase all over again, where we tried to divide the annotated videos over us - "RASD" team members- and crop the videos to include only the violation scene. Because later, the videos will be converted into frames in order to be fed to the model. Consequently, a single video will have both normal and violation frames, of which we require

only the latter. We have also omitted any violation videos that may affect the performance of the machine learning model either by being noisy or of poor quality, as shown in Figure 19.

Figure 19 Poor Quality Video's Examples

While preprocessing the data, we attempted to label each violation type in order to understand how many videos of each violation we have for training purposes. After reconsidering the model's performance, we realized that focusing on one view of the dashcam (the vehicle's front windscreen) and developing a model specific for that view was preferable to working on two views of the dashcam (the vehicle's front windscreen and rear window).

Hence, we omit the tailgating violation because it is captured by the rear dashcam, and any view captured by the rear dashcam contradicts the driving in the opposite direction violation, which is captured by the front dashcam. Because the current model's capabilities are not implemented to distinguish between two distinct views, as any view from the rear dashcam will be classified as a violation, thinking it may be a driving in the opposite direction violation, as shown in Figure 20.

This leads us to reduce the violation types and focus only on three, which are:

1- Driving in the opposite direction

2- Overtaking

3- Drifting

Figure 20 Rear Camera View Classification Example

4.5 Interface Design

In this section, we will show the application structure using a navigation diagram and the UX guidelines that are incorporated while designing the interface.

- **Navigation diagram:**

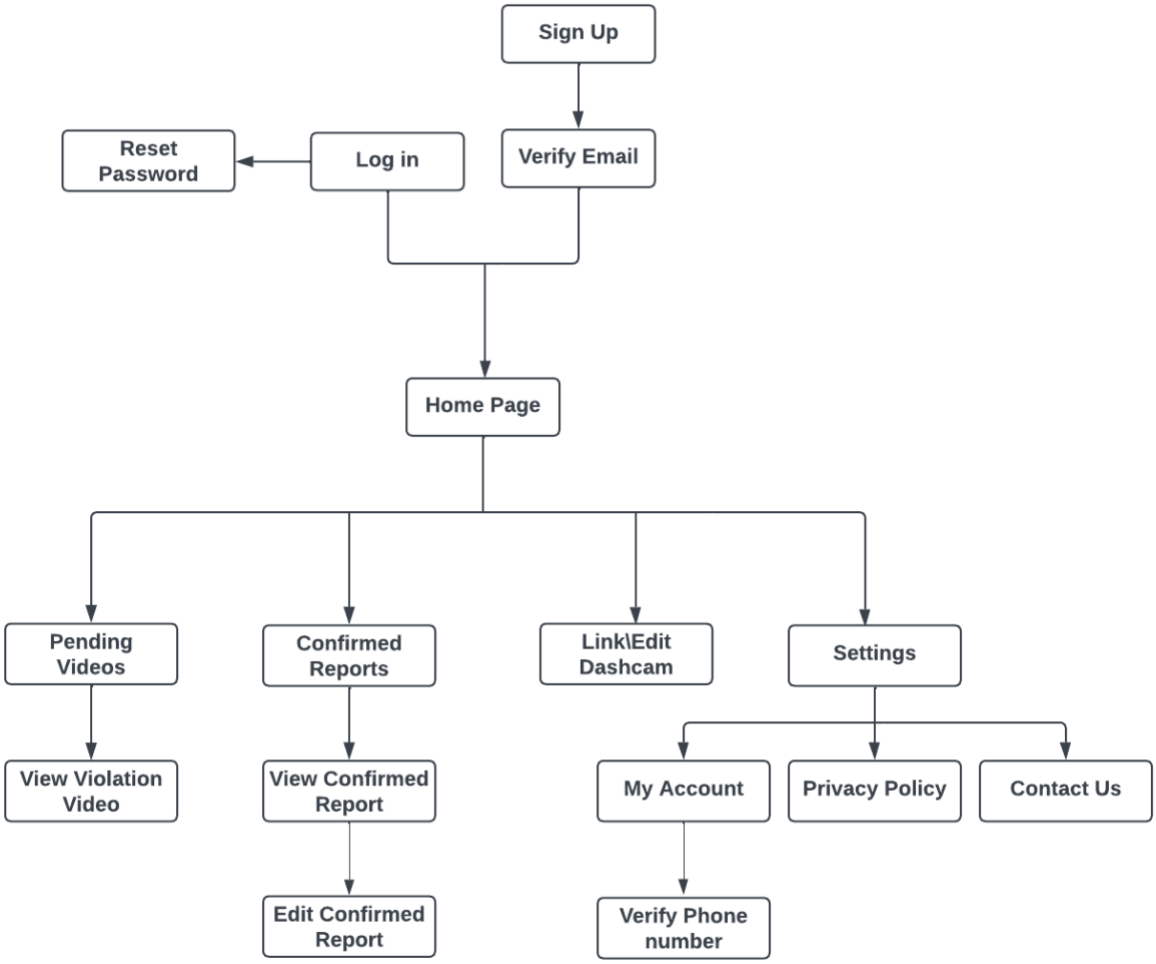


Figure 21 Navigation Diagram

- **UX guidelines applied on the application:**

#Rule	Principle		Figure NO	How is this principle applied?
1	Learnability principle	predictability	-Figure 111	The driver can predict that by pressing the "Save" button on the "My Account" page, their information will be updated and saved in the application's database.
2		Synthesizability (immediate honesty)	-Figure 112	When the driver deletes a violation video, they will immediately get a confirmation message telling them if the action was successful or not.
3		Familiarity	-Figure 113	The "delete" icon while previewing a confirmed report is a metaphor for real-life objects.

#Rule	Principle		Figure NO	How is this principle applied?
4		Generalizability	-Figure 114 -Figure 115 -Figure 116	The pages for "sign up", "log in" and "settings" are similar to those found in most applications.
5	Flexibility	Dialog initiative (system preemptive)	-Figure 117	The driver must answer the confirmation message that appears when they delete or confirm a violation video.
6	Robustness	Observability (Reachability)	-Figure 118	The bottom app bar allows the driver to navigate through pages.
7		Recoverability	-Figure 119	If drivers accidentally click on the "sign up" link, there is a link below that takes them back immediately to the "log in" page.

Table 13 UX Guidelines

4.6 Implementation

In this section, we will describe the key implementation steps and procedures. We will also explain the challenges we faced during the implementation of “RASD”.

First of all, we started constructing our database following the NoSQL mechanism using Firebase, which is Google's cloud computing and development platform. It is a set of hosting services that are provided to any type of application, such as Android, iOS, etc.

The databases offered by Firebase depend on NoSQL relations [30]. Firebase provides many features that can be applied to projects, and from these, we used cloud Firebase authentication and cloud storage services.

“Cloud Firestore” is a NoSQL document database that allows us to easily store, sync, and query data for our application [31], and the authentication service provides multiple ways of authenticating the users of our application [32]. “Cloud Storage” for Firebase allows uploading and sharing user-generated content, such as images and video [33].

We start creating our database with the “Drivers” and “MisclassifiedVideos” collections. Also, we have a sub-collection inside the Drivers collection called Reports. The Reports collection includes the Video collection as a sub-collection. Although we have two main collections, only the driver's collection will be directly used in our application; the “MisclassifiedVideos” collection, on the other hand, will be used in our model to improve it. Because this collection will store violation videos that the driver claims do not exist, meaning our model classified it incorrectly as a violation.

secondly, we start implementing the interfaces of the application using the Flutter framework and the Dart programming language. We started with implementing the interfaces for pages that include the main functions of the application, which are the login and the sign-up, and connecting them to our database. Since these two functions depend on authentication, we used the authentication service provided by Firebase. The authentication service provides different kinds of methods for registering the users with the application, but we only used email/password authentication as a sign-in method, email verification, and reset password services in our application. We have also used the phone number verification service, which allows the drivers to verify their phone numbers after registration. From there, we branch out to the implementation of the other pages, which are:

- Home page
- Link/Edit dashcam page
- Pending Video page
- Show Violation Video page
- Confirmed Reports page
- Preview Report page
- Edit confirmed report page
- Settings page
- Privacy policy page
- My Account page
- Verify Phone number page

Since our main functionalities depend on the live stream coming from the dashcam, we had to figure out how to link it to the application and get access to the stream. To begin, we searched for ways to obtain the stream from cameras with IP addresses and discovered that we could use the RTSP link, which allows us to transmit the stream from the dashcam to the driver via the network [15].

After figuring out how to retrieve the stream from the dashcam, we start to look for a Flutter library, which enables us to manipulate and leverage the stream that comes from the RTSP link.

Using the "flutter_vlc_player" package, we can obtain a video from the network by taking the RTSP link from the driver as an input. As a result, we were able to show the stream in the confirmation dialog after drivers linked the dashcam successfully, see Figure 22. Also, we were able to show the stream on the "home page" to indicate that the stream is still running, see Figure 23. In addition, one of the main advantages of the "flutter_vlc_player" package is that it records the stream and saves it to the driver's local storage. This allows us to upload the files from the driver's local storage to the Firebase cloud storage.

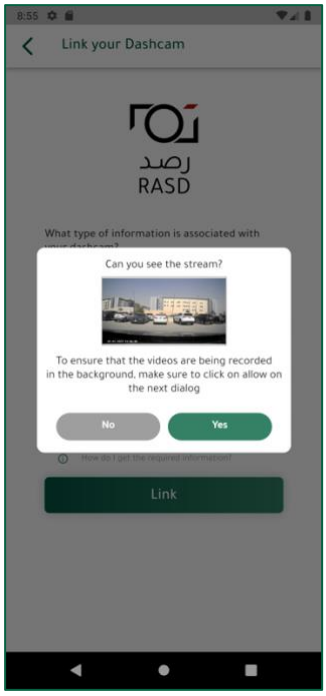


Figure 22 Stream Confirm Dialog

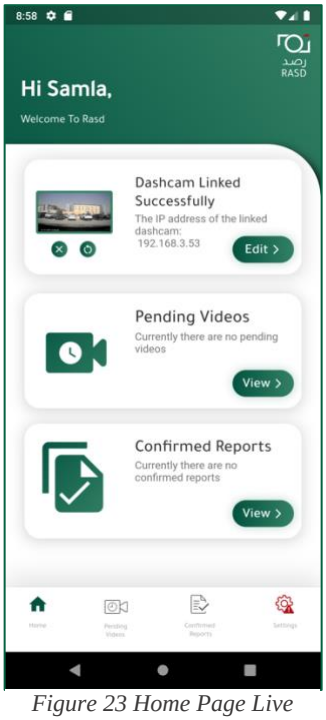


Figure 23 Home Page Live Stream

Another main functionality that our application depends on is having a machine learning model that classifies the obtained videos from the driver's dashcam as a violation or not and shows it for drivers, in order to achieve that we have done some research and found some papers to guide us on which type of machine learning models is most suitable to solve our problem, we come across a lot of machine learning models but only one of them was very helpful. “Real-Time Violence Detection” was discussed in depth in the Literature Review, and it was manipulated to suit our needs. The authors of the project research [52] provided a GitHub repository that included the source code of the model and the dataset that they used to train the model. Furthermore, after reading the paper, and understanding the concept, methodology, what are the phases and the steps that were taken to produce the final model and finally going through the source code by reading it and understanding the implementation process. Then, after all of these steps, we tried to run the model using the provided data in the repository. This allows us to understand how the model's results are obtained and what modifications we need to apply in order to use it for our needs.

After modifying the code, we trained the model with our dataset and start monitoring and analyzing the results and apply the necessary modification such as trying different parameters and techniques like:

- Batch size
- Number of epochs
- Polling type (max and average)
- Augmentation
- Train and test split
- Regularization (early stopping and data augmentation)

To make sure that the violation detection model only processes the frames that include vehicles we add an object detection model called ‘YOLOv5s’ as a pre-step before the frame passed through the violation detection model.

The last step of implementing the model is to determine the formula to obtain the final decision of the video either a ‘violation’ or ‘Not violation’. After multiple trials and techniques, we found out that the best possible formula that can be generalized to make the final decision about

the processed video is to use “the Count with the High threshold rate” found in Table 27 differences to obtain the variance of the prediction frames.

After implementing our machine learning model, we start looking for a way to deploy it. we found out Heroku. Heroku is a cloud service platform based on a managed container (called dynos within the Heroku paradigm) system. It has integrated data services and a powerful ecosystem for deploying and running modern applications [34].

All the previously mentioned implementation phases represent “RASD” project.

During the process of implementing “RASD”, we faced many challenges, but we had a strong desire to overcome these challenges.

The significant obstacle we stumbled upon, is having a problem with a newly proposed solution, even though some of the researchers mentioned a similar solution but they were more like black boxes where they did not share enough information about the phases, steps, and procedures taken to reach the results that were mentioned in their papers. Also, the important details about the proposed solution were described at a very high level, where we could not get something useful out of it such as the used algorithm so we can reflect it on our problem [18]. On the other hand, some papers represent their solutions at a very low level where only experts can deal with such details and even experts might need a guide to follow up with the mentioned methodologies and mathematical algorithms to be applicable. those papers were not understandable nor readable at all [35][35]. Not only the provided papers were on a low level, but the provided GitHub repositories also did not run their source code and some of them even have intended errors such as[36][37][38][39][40][41]. Where we found lines of errors when we were going through their codes to understand them, which makes it impossible to execute them. This led us to a dead end and made us explore more solutions to a semi-similar problem so we can modify it to suit the problem we are trying to solve, we found one project research [52] that intended to detect violence made by humans. They claim that they got 96% accuracy on the test set and we found that some of the data processing steps were unsuited to our problem, like flipping, rotating, and resizing the frames, where in our problem the direction of the car view is important and should not be tampered with such as Figure 24.

Although there were some problems with their model, by consuming a lot of effort and time we were able to solve them and modify them to suit our problem and obtain a model that detects traffic violations.

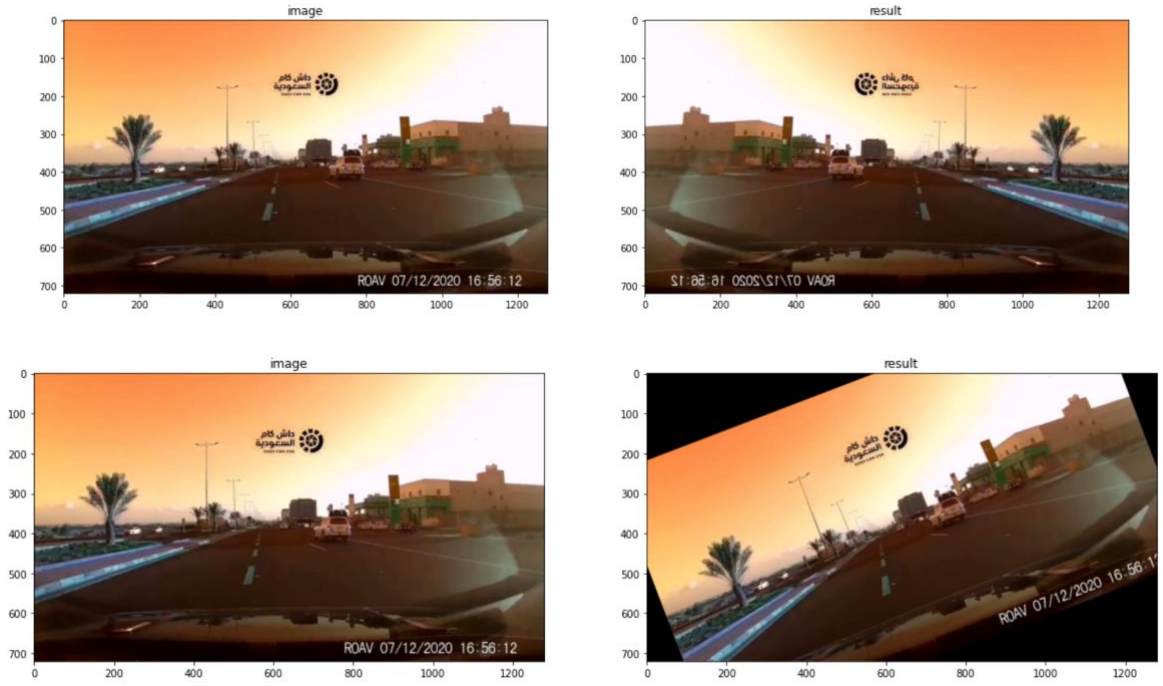


Figure 24 Data Augmentation Examples

The challenges did not stop here, we went through other challenges with flutter libraries, in normal cases we only had to install the library to get the benefit of what it is providing but when we want to verify the phone number input. One of the cases is customizing the phone number error message rather than the default error message which only comes in English format. Finally, we adjusted the library's error messages to only appears in the following cases:

- The driver leaves the phone number input field empty while registering or editing the phone number as it is required.

- Entering an invalid phone number either on the registration page or in the driver edit information page.

The other case were when we tried to obtain the stream, the first library we came across was working but it did not provide the feature of recording and saving the obtained videos, Then we had to search for another library that provide these feature which is "flutter_vlc_player", the problem we faced with this library was controlling the length of the video since "flutter_vlc_player" library did not have a method to automatically stop the recording after a specific duration which means that the driver needs to manually start and stop the recording. To overcome this challenge, we utilized a timer class to write a method that automatically starts, stops and uploads the recording to the driver's internal storage without any human interaction.

In order to make sure that the stream is running and recording all the time, we must make the app run in the background, so we needed to disable the battery optimization option. However, that requires us to access the driver's mobile battery settings. This can only be done if we have permission for disabling the battery optimization option. And this is done by displaying the battery optimization message. To display this message, we used the "optimize_battery" library that facilitated the interaction between the driver's battery settings and the "RASD" application.

One of the frustrating challenges was in the "Show Violation Video Page", where we intended to retrieve the URL from the database and display it as media content in the application. This process of displaying the video in the application using Flutter was hard since the resources were limited and Flutter updates its packages regularly. Because of that, some suggested solutions do not work anymore. We spent a lot of time on that function, but we were able to

get through this problem. There is also another function that is considered very challenging and very hard to achieve, which is generating a ready-made report. In order to generate the report, we used a specific library called Printer.

Unfortunately, this library did not have any additional resources on how to customize the PDF other than the official library documentation [42]. Although the implementation of generating the report worked well, the part of sending and customizing the email template that will be sent to the authorities was extremely difficult. We tried many different solutions, such as using the attributes given by the class “PdfPreview” [43][43], but it did not work. Then we tried to modify the library code itself, but the changes we made did not reflect on our code, so we tried to make a copy of the library file and add it manually to our code, but the whole project did not run at all. Then, after digging deep into the documentation provided by Flutter to find the code that customizes the shared email, we found a class under “PdfPreview” called “PdfshareonAction” [44] that can be used to customize the shared email; however, there was no clear instruction on how we use these two classes together. After digging deeper, we found that in order to use these two classes together, we have to use an attribute called "actions" that can hold multi-functions and classes. Finally, after implementing these two classes together, the customization of the shared email worked well and as we wanted it to. As well as the language preferences in the application, it was an important feature to implement and an applicable one, but the real challenge was when we started translating every single word in the application, which required a lot of time. Furthermore, since our application offers for the registered drivers to reset their password, we tried to force the same validation that we have done when the driver ringsters for the first time which was: at least 8 characters, containing capital letters, small letters, numbers, and symbols, but unfortunately Firebase Authentication does not offer such a feature while it offers a weak validation which is 6 characters only [45]. This is a frustrating challenge because we tried to use different ways of validation on the password the driver enters when resetting.

We contacted firebase support to grab their attention to such important feature to offer in the future as shown in Figure 25.

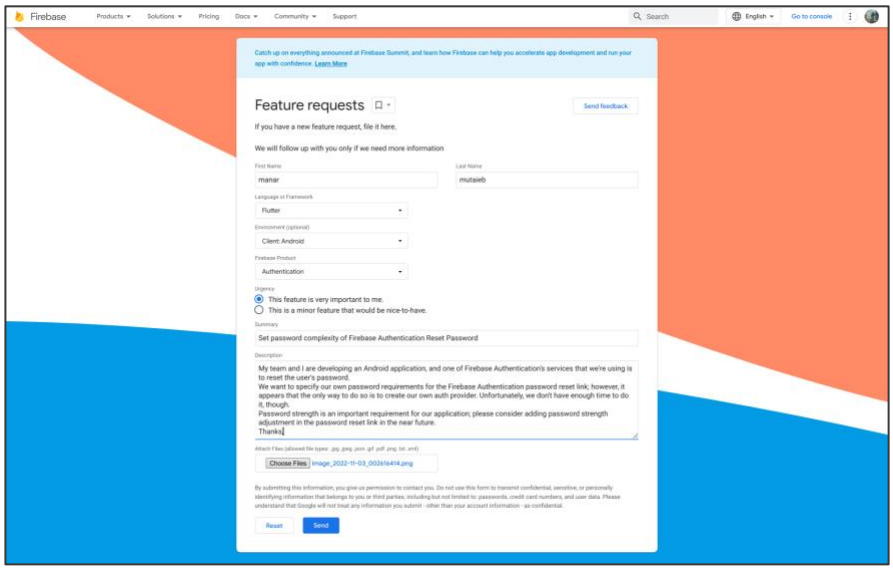


Figure 25 Firebase Feature Request

We did use the flutter framework and we will complete the development on it, but we believe that flutter documentation for dart language is insufficient.

Finally, there were some problems that are beyond our control. Since our code demand high performance machines in order to perform better, however, our current devices are not computationally powerful, and this led to a slower performance thus delaying our schedule as some processes took a lot of time of execution and other processes were delayed because they relied on it. Also, some of the used services such as: Heroku, Firebase Cloud Storage and Google Colab they both provide limited memory size, limited uploading space, limited number of free requests, we exceed all of these limits since it was not enough for our project. Thus, we had to upgrade our subscription plans in order to avoid being blocked by these services.

No doubt, the journey of implementing the application was full of challenges as we are implementing a novel solution. we have encountered challenges since the moment we started the implementation till the end. But if we learned one thing from this project, it is that every obstacle is solvable, and by exploring, learning, and working together with an ambitious team, each challenge is going to be solved after many trials.

4.6.1 GitHub Repository

You can access our GitHub repository through this link

Function	Process the frames to detect vehicle
Description	This function allows the frames to be processed by the vehicle detection model before being passed to the violation detection model.
Function Flow	The video is first divided into frames. Then, as we have 30 frames per second, we want to take the 10th frame from each iteration, which is in total 3 frames per second, in order to reduce the overload for our machine learning model and keep its performance high. After the vehicle is detected, it is passed to the violation detection model to detect the violation, if any.
Code	

```
while True: # loop till end of the video file
  # read the next frame from the file
  (grabbed, frame) = vs.read() # to save a frame
  frameCounter += 1 # increment for each frame
  # if the frame was not grabbed, then we have reached the end of the stream
  if not grabbed:
        break
  # print("frameCounter",frameCounter)
  if frameCounter == 10: # as per second we have 30 frames and we want to take the 10th frame in each ieteration
    output = frame.copy()
    # Process image.
    detections = pre_process(frame, net)
    output,detected_vehicles_arr = post_process(frame.copy(), detections) # to get the labeling for the object detection
    # Put efficiency information. The function getPerfProfile returns the overall time for inference(t)
    #  and the timings for each of the layers(in layersTimes)
    t, _ = net.getPerfProfile()
    label = 'Inference time: %.2f ms' % (t * 1000.0 / cv2.getTickFrequency())

    if (classes[2] or classes[3] or classes[5]or classes[7]) in detected_vehicles_arr:
    # if the frame include a vehicle then process the frame with our model
    else:
      frameCounter = 0 # it is important to go to the next iteration
      continue
```

Table 14 Detect Vehicle Frames Code Segments

Function	Calculate the true frames that have more than 0.70 prediction value
Description	This function computes the violation video's final decision based on the number of the true frames that have more than 0.70 prediction value.
Function Flow	Firstly, the model predicts the frames and inserts the prediction in the "predCountArray" array. Then, it will count the number of "True" frames which have more than 0.70 prediction values. Lastly, if the number of "True" frames that have more than 0.70 prediction value is more than or equal to four then the function will return True -which means the video is classified as "Violation"-, and otherwise it will return False -which means the video is classified as "Not Violation"-.
Code	`# this function to calculate how many high prediction values we have` `def countTruePred(Q):` `  countTrue = 0` `  for i in range(0,len(Q)):` `  if(Q[i][0]>=0.7):` `  countTrue +=1` `  if(countTrue >= 4): # if we have 4 high value predictions in a video then it is a violation` `  return True` `  else:` `  return False` `  return False`

Table 15 Calculate Decision Code Segments

Function	Setting the timer that will run the “recordingLoop” method periodically, every one minute.
Description	This function allows the “recoringLoop” method to be called every minute after obtaining the life stream. the “recoringLoop” method gather all the recording and saving processes, including the start, stop, and upload of the recording.
Function Flow	At first, a Timer object will call the “recordingLoop” method every 60 seconds. This method will check if there is a live stream; if there is no live stream, the method will not take any action. If there is, the “startRecording” method will be called. First, it will get the path of the driver's mobile internal storage and store it in the “_directoryPath” variable, which is the path where the recording will be saved. Then, it will verify that the “recordPath” variable, which is a VLC library variable that contains the path of the saved recording, is not empty before calling the “uploadFile” method. However, since no video has been recorded at the first iteration of the loop, it will be empty. Then, the “startRecording” function, which is provided by the VLC library, will be called, which starts the recording of the video and takes the “_directoryPath” variable as a parameter. After that, we will have another timer object that is set to countdown for 55 seconds, and when it reaches the end, it will call the stop recording method, which is also provided by the VLC library, and stop the recording and save it. then there will be 5 seconds left before the “recordingLoop” method begins again to release all variables. And once “recordingLoop” begins again, the “recordPath” variable, which is in the “startRecording” method, will not be empty and will invoke the “uploadFile” method, which will take the recording path and save it in firebase storage inside the user's specific directory.

Code

```
//repete recording every minute
timer = Timer.periodic(
  Duration(seconds: 60),
  (_) => recordingLoop(),
);

//This method collect all the recording and saving process. (start->stop->upload recording)
void recordingLoop() {
  if (isLive!) {
    print('rec loop');
    //Record if the video is playing
    vidLength = 55; //set the video length to be 55 seconds
    startRec(); //call start recording method
    const oneSec = Duration(seconds: 1); //decrement vidLength each second
    countDown = Timer.periodic(
      oneSec,
      (Timer timer) {
        if (vidLength == 0) {
          //if the recording reached the end stop recording
          stopRec(); //call stop recording method
          setState(() {
            timer.cancel(); //stop timer
          });
        } else {
          setState(() {
            vidLength--; //decrement by one second
          });
        }
      },
    );
  }
}

//This methods is to start recording the video
void startRec() async {
  //the path of driver's mobile internal storage, which we will store the recordings in
  var _directoryPath = await _requestExternalStorageDirectory();
  if (_videoPlayerController.value.recordPath != '') {
    //Tha [_videoPlayerController.value.recordPath] is where videos will be saved in,
    // If it's empty then the video is not saved yet
    uploadFile(_videoPlayerController.value
        .recordPath); //call [uploadFile] method to upload the video to storage
  }
  _videoPlayerController
      .startRecording(_directoryPath.toString()); //start recording the video
}

//This methods is to stop the recording of the video
void stopRec() async {
  _videoPlayerController.stopRecording(); //stop recording the video
}


//This methods is to upload the recording of the video to the firebase storage
Future uploadFile(String _path) async {
  var pathSplit = _path.substring(_path.lastIndexOf('/') + 1);
  //Split the the received path, and store it in an array.
  // a typical path would look like 'folder/folder/folder/filename'
  //after splitting it would be like ['folder', 'folder', 'folder', 'filename']
  final path = widget.uid +
      '/$pathSplit'; //this is the path the video will be stored in at Firebase Storage,
                     //it consist of the user id and the video file name
  final file = File(_path); //Creates a [File] object.
  final ref = FirebaseStorage.instance.ref().child(
      path); //Returns a reference to a relative path from this reference.
  ref.putFile(file); //Upload a [File] at FirebaseStorage.
}
```

Table 16 Stream Timer Code Segments

Function	Disable battery optimization to run the application in the background.
Description	This function allows the drivers to receive the stream and record it all the time by running the app in the background and disabling the battery optimization option.
Function Flow	Firstly, the driver is required to access the link/edit dashcam page by clicking on the Link/Edit button from the home page. Then, on the link/edit page, a form will be displayed that allows the driver to choose the dashcam type. After selecting a dashcam type and entering valid inputs, a confirmation dialog will appear with a box showing the stream to indicate if the linking was successful or not. After linking successfully and the driver clicking the "Ok" button, the system first will check if the battery is optimized, then it will show the battery optimization dialog with two options: "Allow" and "Deny". By clicking on "Allow", the driver will be able to receive the stream and record all the time, and by clicking on "Deny", the driver will be able to receive the stream and record only if they were in the application. After choosing either of the options, the driver will be redirected to the home page. If the battery is already unoptimized, no dialog will show, and the driver will be directed to the home page.
Code	``` OptimizeBattery.isIgnoringBatteryOptimizations().then((onValue) { // if onValue true then it is not optimized setState(() { if (onValue) { // if it is not optimized then the app is runining in background } else { // if in else then the app is not runing in the background // optimized // it will show a dialog asking the driver to allow the app to run in the background OptimizeBattery.stopOptimizingBatteryUsage(); } }); }); ```

Table 17 Battery Optimization Code Segments

Function	Changing the language preferences from the settings of the application.
Description	This function allows the drivers to change the preferred language that have been selected while registering or logging in. The driver will also have the ability to change it from the settings page whenever they needed in the easiest way. In order to have the whole interfaces displayed in the chosen language.
Function Flow	Firstly, the driver is required to access the settings page from the bottom app bar which will be displayed in every main interface. Then, In the Settings page all the applicable functions will be displayed clearly to the driver along with the function of changing the preferred language. The driver can see the current application language and by clicking on it. The system will response by showing a message asking the driver to choose their new preference, when the driver click on the preferred language a splash page will occurs with the application's logo for a few minutes and the translations for all the text in the interfaces will be retrieved from the "LocalString" page where every single word in the application is translated, then the driver will be redirected to the application home page where all the interfaces changed to the selected language.
Code	`class _SettingsScreenState extends State<SettingsScreen> {` `  final List locale = [ // a list to hold the two languages` `  {'name': 'English', 'locale': const Locale('en', 'US')},` `  {'name': 'العربية', 'locale': const Locale('ar', 'AE')},` `  ];` `  bool arLnag = "Plang".tr == 'Preferred Language' ? false : true;` `  updateLanguage(Locale local) { // this method to update the app to the selected language` `  Get.back();` `  Get.updateLocale(local);` `  }` `Map<String, Map<String, String>> get keys => {` `  'en_US': {` `  //global` `  'signout': 'Sign Out',` `  'Sure': 'Are you sure?',` `  //Homepgae` `  'greeting': 'Welcome To Rasd',` `  'LD': 'Link Dashcam\n',` `  'LDA': 'Link your dashcam to our application',` `  'L': 'Link',` `'ar_AE': {` `  //global` `  'signout': 'تسجيل الخروج',` `  'Sure': "هل انت متأكد ؟ ",` `  //Homepgae` `  'greeting': 'مرحبًا بك في رصد',` `  'LD': 'ربط الداش كام\n',` `  'LDA': 'قم بربط وتوصيل كاميرتك الداش كام بتطبيقنا',` `  'L': 'اربط',`

Table 18 Change Language Preferences Code Segments

Function	Share the readymade report for the confirmed violation video with the authorities via email.
Description	This function will allow the drivers to share the generated violation report by the application with the authorities via any email application with a predefined receivers email, subject of the email, attachments and body of the email.
Function Flow	First of all, to generate the report the driver must access the pending violation videos from the home page or from the bottom app bar and confirm the legitimacy of the violation and they could choose the type of the violation or leave it unspecified. After that, the system will redirect the driver to the confirmed violation reports. From there, the driver can share the report immediately or they can access the confirmed report later. From the confirmed violation reports the driver can select the report that they decided to share, and they will be redirected to the preview report page where all the needed information is displayed there. Finally, when the driver clicks on the share button a share dialog will open with all email applications in the driver device shown, and then when the driver clicks on one of these email applications an email will be prepared where the body of the email and the subject is prewritten, and the receiver of that email is prespecified. The generated report will be attached automatically to that email the driver needs nothing else than click on “send” to file the report.
Code	`child: PdfShareAction( // to customize the share option and the email` `  icon: arLnag` `  ? new Image.asset(` `  "assets/images/shareIconAr.png",` `  ) // Image.asset` `  : new Image.asset(` `  "assets/images/shareIconEn.png",` `  ), // Image.asset` `  body: 'body'.tr,` `  emails: ['group9.gp1444@gmail.com'],` `  filename: "filename".tr + ".pdf",` `  subject: 'subject'.tr,` `  onShared: () {` `  UpdateStatus();` `  },` `), // PdfShareAction`

Table 19 Pre-filled Email Code Segments

5 System Evaluation

In this section, we will first walk through the model experiments where we applied the machine learning process over "RASD" dataset and then test the obtained model by passing part of "RASD" dataset that was designated for testing purposes. From there, we have come up with multiple decision-making methodologies to compare their results and choose one of them according to its performance. Secondly, we will check if the system fulfills the business requirements and if it can be used by our end users (drivers in Saudi Arabia) in the user acceptance testing subsection. Our testers are composed of twenty-one drivers. The participants are described in more detail in the "Demographics of Participants" subsection. At the end of each test, we gave the participants a questionnaire composed of 14 questions to provide their feedback on the application and the testing process.

5.1 Experimental Results

In this subsection, we will present the experimental results for the traffic violation machine learning model.

5.1.1 Experimental Set up

The experimental set up that is used to build the traffic violation machine learning model is as shown in Table 20

Environment	Google Colab
Programming language	Python
Libraries	TensorFlow, keras, NumPy, cv2, imageio, imagaug, sklearn
Dataset	"RASD" dataset for traffic violation detection

Table 20 Experimental Set up

5.1.2 Video dataset

"RASD" contains 348 videos of traffic violations and 348 videos of non-violations. To build our model, we used 64 traffic violation videos and 65 non-violation videos, which is a total of 13038 frames. The frame sizes were unified at 128 by 128.

The dataset includes traffic videos (violation and non-violation) at different times and driving states, such as:

- Day
- Night
- Turns
- Traffic jam
- Parking
- Main roads
- Branch roads
- Neighborhoods

And the traffic violation videos include certain types of violations, which are:

- Drifting
- Overtaking
- Driving in the opposite direction.

As shown in Figure 26.

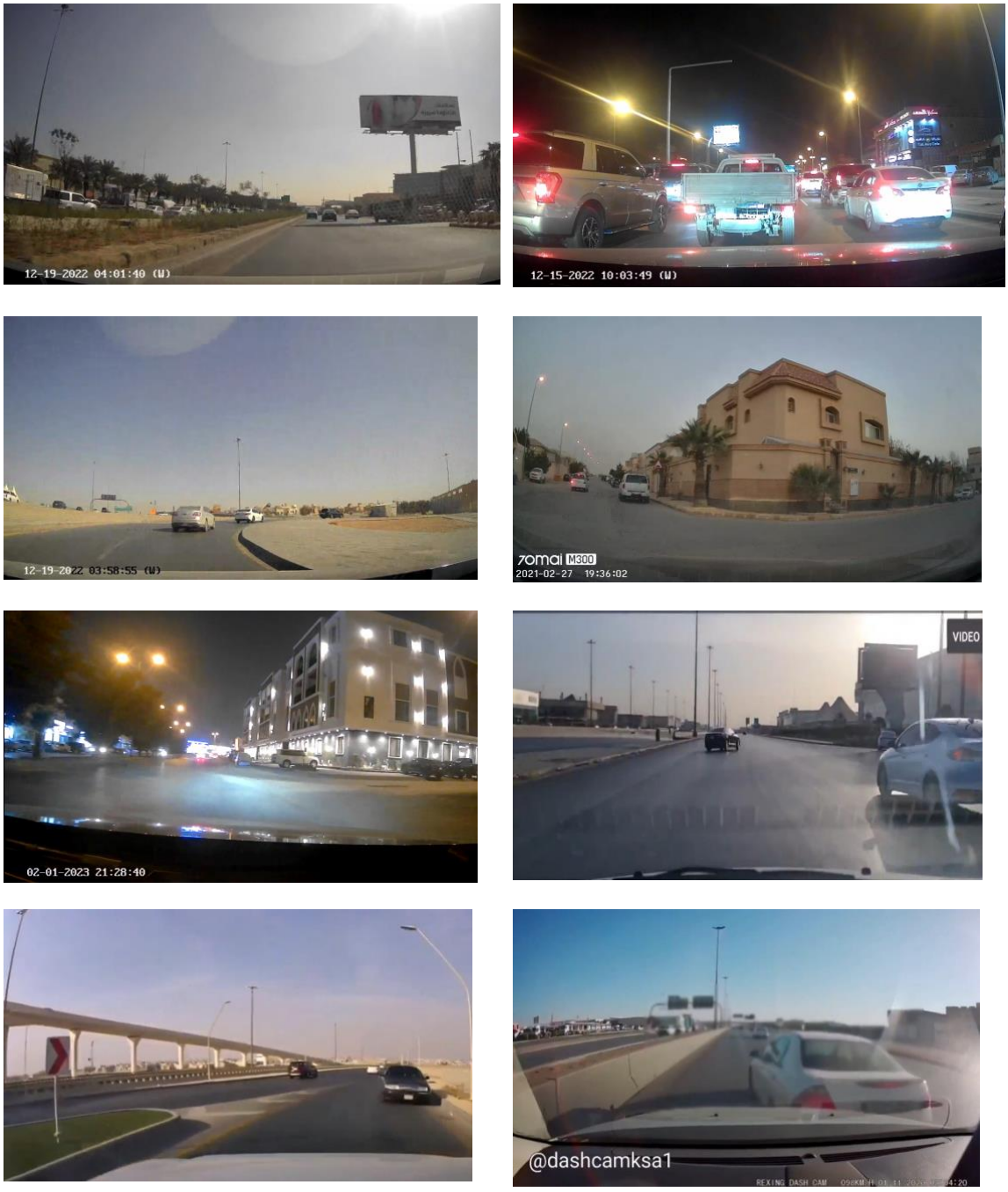


Figure 26 "RASD" Dataset Video Examples

5.1.3 Pre-processing step

First, we extracted the frames from each video; we only took the frames that were multiples of three to avoid duplicating frames. Then, we have applied one of the preprocessing techniques over the frames, which is data augmentation, in the hope of achieving better performance. The augmentation techniques used are brightness; converting image colors from BGR to RGB as OpenCV reads images in BGR ordering[47], while other libraries and models are more familiar with the RGB ordering, like the MobileNetV2 model[48]. The frames were then resized to 128*128 as shown in Figure 27. After that, we normalized the values of the frames to be in the range 0 to 1 by dividing the frames array by 255, as pixel values range from 0 to 256.

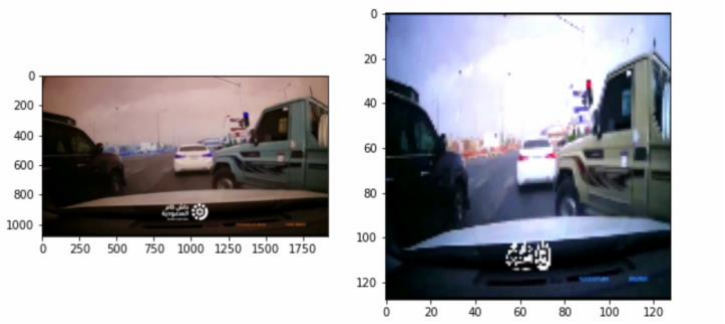


Figure 27 Frame After Pre-processing

5.1.4 Performance of CNN-RNN

The Keras library provides multiple code examples for different use cases; one of them was for classifying videos by utilizing transfer learning and a recurrent model on the UCF101 dataset[49]. The UCF101 dataset used in this example consists of videos of different actions to build an action recognition system. The idea behind the CNN-RNN architecture is to use a hybrid architecture with convolutional layers for processing frames with spatial information and recurrent layers for processing the sequence of those frames, which has temporal information. This model architecture was considered since "RASD" addresses a related issue, violation detection in dashcam videos, which falls under the category of action recognition in videos. The model was tuned with a set of hyperparameters, which are image size, batch size, number of epochs, the maximum sequence length, and the number of features. Except for the maximum sequence length parameter, which sets the number of frames from which temporal information would be extracted, the model hyperparameter was suitable for our data. Since our data consists of videos with short

durations (ranging from 0.5 milliseconds to 10 seconds), each video contains a tiny number of frames; hence, the number of frames from which we wish to extract temporal information is also minimal. We therefore set it to 30 frames rather than 20 to get the most learning from the training data. After training the model with a maximum sequence length of 30, it achieved an accuracy of 80%. We discovered that when it comes to short-duration videos (from 1 to 5 seconds), it can distinguish between violation videos (Figure 28 and Figure 29), and normal videos (Figure 30). Except for the overtaking violation (Figure 31), it seems that the model always misclassifies it as not-violation. But when it comes to long-duration videos (from 10 seconds and on), the model was not able to correctly predict the violation videos from the normal ones. As when the model was tested on the same violation video but with different durations, it predicted that the short duration video (1 seconds) was 62.02% violation while the long duration video (20 seconds) was predicted as 58.11% not-violation, see Figure 29 and Figure 33. Same with Figure 28 and Figure 32, as it predicted that the short duration video (0.36 milliseconds) was a 61.37% violation, while the long duration video (11 seconds) was predicted as a 51.29% violation. Although both trails in Figure 28 and Figure 32 were predicted as violations, the difference in percentages raised our concerns. So, we thought that this model was not suitable for our problem since we wanted to feed our prediction model with one-minute-long videos.

Figure 28 Predicted Violation Video A.1

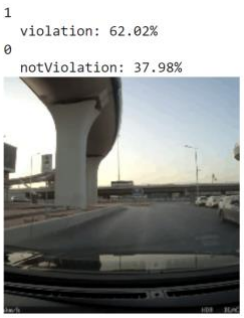


Figure 29 Predicted Violation Video B.1

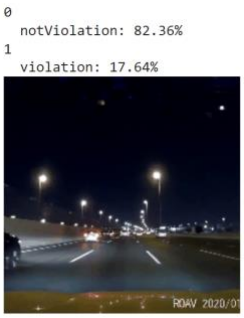


Figure 30 Predicted Normal Video A

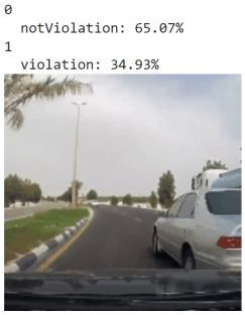


Figure 31 Predicted Violation Video C

Figure 32 Predicted Violation Video A.2

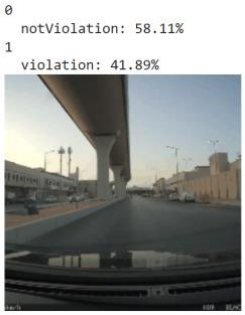


Figure 33 Predicted Violation Video B.2

5.1.5 Performance on MobileNetv2

For abnormal activity recognition, a MobileNet deep learning model is used. The inspiration for using MobileNet is its lightweight architecture. MobileNet uses depth-wise separable convolutions on each color channel. After that, a pointwise convolution is applied that is further combined with depth-wise convolutions. This combination reduces computation and model size significantly. Even though the learning algorithms under the "deep learning" branch are highly advanced in solving complex problems, their results are heavily dependent on the nature of the information preparation, its quality, and the nature of the issues[16]. After training the model on the "RASD" dataset, it is ready to be tested.

The MobileNetV2 model is a pre-trained convolutional neural network (CNN) with 53 layers, and according to our previous knowledge, CNN is a neural network architecture that is used for image classification problems. The model could be applicable to our problem, traffic violation detection, as the images in our case are the frames extracted from "RASD" traffic video dataset and could be classified as violation and non-violation frames. Also, based on [16] comparison, MobileNet is the most suitable model for detecting violence in surveillance videos, which is similar to our problem.

As with other machine learning models, MobileNetV2 is a model that can be tuned using different hyperparameters; therefore, while we were training the MobileNetV2 model, we froze the weights of all the model's layers except for the output layer and tuned the model's hyperparameters with different values, then compared their results. Firstly, we have tried different batch sizes (4 and 32). We noticed that there was no difference in the final results, but the training time was faster when we used a batch of size 32. Secondly, we tried different values for the number of epochs, which are 20, 30, 35, and 50. We observed that the testing accuracy increased as the number of epochs increased, as the accuracy values were 79%, 85%, 89%, and 98%, respectively.

After training the MobileNetV2 model with 70% of the data, we have obtained an accuracy of 98% for frame detection. The model was able to predict 3837 frames out of 3912 frames correctly. By looking at Figure 34 from the precision values, we can observe that the model's predictions were 98% correct for both violations and not-violations. We also examine the recall values to determine the model's correct predictions over the violation and non-violation frames among the true frame labels. The recall for the not-violation

frames is 100% and for the violation frames, 89%, and this means that the model can perform well in identifying the normal frames but not as well as the violation one.

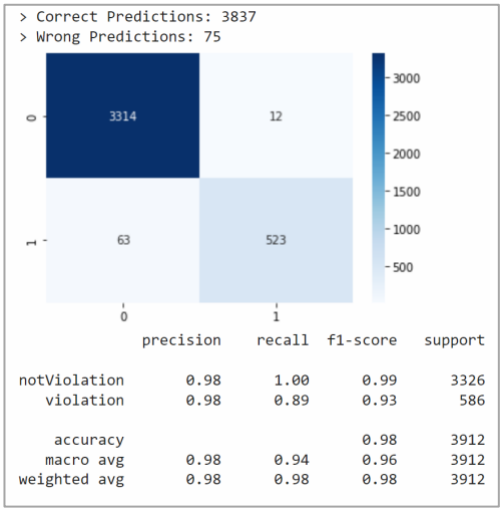


Figure 34 The Confusion Matrix

5.1.6 Comparison between the two models

Below, is a comparison of the CNN-RNN and the MobileNetV2 results after being trained on "RASD" dataset. We can see that the MobileNetV2 outperformed the CNN-RNN model in terms of precision, recall, f1-score, accuracy, and loss values.

Model Name	Number of epochs	Class	Precision	Recall	F1-Score	Accuracy	Loss
CNN-RNN	50	Violation	0.86	0.67	0.75	0.80	0.63
		Not violation	0.77	0.91	0.83	0.80	0.63
MobileNetV2	50	Violation	0.98	0.89	0.99	0.98	0.076
		Not violation	0.98	1.00	0.93	0.98	0.076

Table 21 Comparing The Results of The CNN-RNN and MobileNetV2 After Training with "RASD" Dataset

5.1.7 Combine both models

Due to the limitations of both models, we decided to merge them to overcome their individual shortcomings. To address the CNN-RNN model's constraint of only handling short videos (1-5 seconds) rather than a full 1-minute clip, we will feed it with only the two frames detected by the mobilenetv2 model. Additionally, the CNN-RNN model, rather than the equation stated in Final Results, will be utilized to determine the outcome of the entire video.

After combining the two models and testing them in the Google Colab environment, we observed a slight enhancement in the decision for the entire video compared to using the equation. However, we encountered issues when trying to deploy the model on the Heroku server since it requires more computational resources than Heroku provides.

5.1.8 Final Results

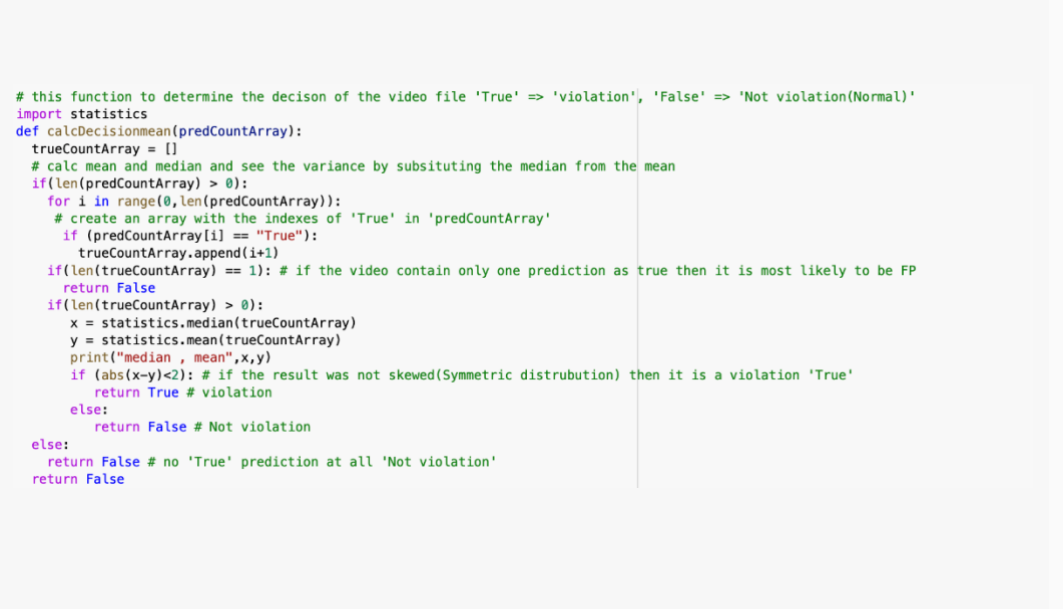
In this subsection, we will go over the various decision-making methodologies that were used to determine the video's final decision and classify it as a violation (true) or non-violation (false).

Decision making methodology	# Not Violation	# Violation Videos	Correctly Classified videos	Accuracy	Overall Accuracy
Difference between mean-median	10	10	Violation: 3/10	30%	60%
			Not violation: 9/10	90%	
Count sequence	10	10	Violation: 10/10	100%	80%
			Not violation: 6/10	60%	
Average prediction	10	10	Violation: 1/10	10%	50%
			Not violation: 9/10	90%	
Average true count	10	10	Violation: 5/10	50%	60%
			Not violation: 7/10	70%	
	10	10	Violation: 9/10	90%	95%

True count with high threshold rate			Not violation: 10/10	100%	
The Five Seconds Decision	10	10	Violation: 9/10	90%	80%
			Not violation: 7/10	70%	

Table 22 Decision making methodologies

In the tables below, we will discuss each decision-making methodology in great detail.

Function	Calculate the variance in the prediction to predict the final decision.
Description	This function computes the violation video's final decision based on the differences between the mean and median in that violation prediction.
Function Flow	Firstly, the model predicts the frames and inserts the prediction in the “predCountArray” array. Then, the model predicts the frames and inserts the prediction in the “predCountArray” array. Then, we loop through the array in order to determine the indices of each "true" frame and create an array of these true indices called the “trueCountArray” array. After that, we call the “calcDecision” method in order to calculate the mean, median, and difference between them, then return a boolean variable to indicate whether the video is a violation or not, “True” or “False”.
Code	

```
# this function to determine the decison of the video file 'True' => 'violation', 'False' => 'Not violation(Normal)'
import statistics
def calcDecisionmean(predCountArray):
  trueCountArray = []
  # calc mean and median and see the variance by subsituting the median from the mean
  if(len(predCountArray) > 0):
    for i in range(0,len(predCountArray)):
     # create an array with the indexes of 'True' in 'predCountArray'
      if (predCountArray[i] == "True"):
        trueCountArray.append(i+1)
    if(len(trueCountArray) == 1): # if the video contain only one prediction as true then it is most likely to be FP
      return False
    if(len(trueCountArray) > 0):
       x = statistics.median(trueCountArray)
       y = statistics.mean(trueCountArray)
       print("median , mean",x,y)
       if (abs(x-y)<2): # if the result was not skewed(Symmetric distrubution) then it is a violation 'True'
          return True # violation
       else:
          return False # Not violation
  else:
    return False # no 'True' prediction at all 'Not violation'
  return False
```

Table 23 Difference Between Mean and Median Methodology

Function	Calculate the sequence of the truly predicted frames.
Description	This function computes the violation video's final decision based on the sequence of the true predicted array.
Function Flow	Firstly, the model predicts the frames and inserts the prediction in the "predCountArray" array. Then, we will check if there were frames that had prediction values saved in the "predCountArray" by making sure its length is greater than 0. Secondly, we will loop through the whole array and check if the current prediction and the subsequent prediction are both true. If so, we will add 1 to the sequence count, which means that we have a sequence of true predictions; if not, we will save the sequence count value in a temporary value for the sequence length and check if the maximum sequence length that we achieved till the current iteration is less than the temporary value of the sequence length; if so, the maximum sequence length will hold the value of the temporary sequence length, and we will reset the value of the sequence count to zero for the next iteration. After that, the "sequence" variable will hold the maximum value between the sequence count and the maximum sequence length. Finally, we will send the value of the sequence to the other method, where we check if the sequence was larger than or equal to 3 and return True or False accordingly.
Code	

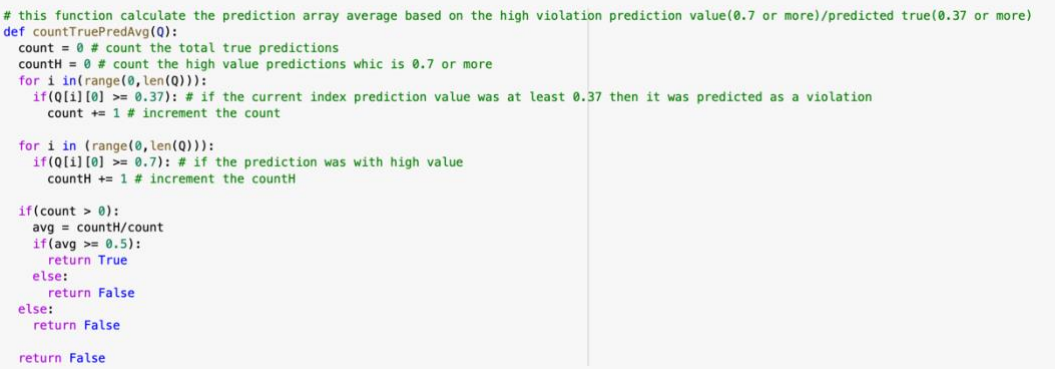
```
def calcDecisionSequence2(predCountArray):
        maxSeqLength = 0
        seqCount = 0
        if(len(predCountArray) > 0):
          for i in range(0,len(predCountArray)-1):
            #if i and i+1 are all true 3 times then the decesion is true
            if (predCountArray[i] == "True" and predCountArray[i+1] == "True"):
              seqCount +=1
              print('seqCountIn if', seqCount)
            else:
              print('-'*9)
              print('final seq', seqCount)
              tempSeqLength = seqCount
              print(f'before max: {maxSeqLength}, temp {tempSeqLength}')
              if (maxSeqLength < tempSeqLength):
                maxSeqLength = tempSeqLength
                print(f'after max: {maxSeqLength}, temp {tempSeqLength}')
              seqCount =0
          sequence = max(seqCount,maxSeqLength)
          print('final sequence', sequence)
          decision = calcDecisionSequence(sequence)
          print('seqCount', seqCount)
          print("pred",predCountArray)
          print("decision",decision)
        return decision

# if the true sequence was at least 3 then it is a violation
def calcDecisionSequence(sequence):
  if(sequence >= 3):
    return True
  else:
     return False
```

Table 24 Count Sequence Methodology

Function	Calculate the average of the truly predicted frames.
Description	This function computes the violation video's final decision based on the average true count in the total frames.
Function Flow	Firstly, the model predicts the frames and inserts the prediction in the "predCountArray" array. Then, we will check if the "predCountArray" is not empty; if so, we will walk through all of the array and check if we have a true prediction; if so, we will increment the "countTrue" by one and repeat until the end of the "predCountArray" Furthermore, we will compute the average of the "countTrue" by dividing its value over the length of the "predCountArray" which holds all the video frame predictions. Finally, we will check if the average value is greater than 0.5, which means that the prediction contains at least 50% true counts, in which case the video is a violation, otherwise it is not.
Code	`# this function to determine the decison of the video file 'True' => 'violation', 'False' => 'Not violation(Normal)' based on the average prediction` `#  averag = true count/total prediction array length` `def calcDecisionAvg(predCountArray):` `  countTrue = 0 # true count prediction in the array` `  if(len(predCountArray) !=0):` `  for i in (range(0,len(predCountArray))):` `  if(predCountArray[i] == "True"):` `  countTrue +=1` `  avg = countTrue/len(predCountArray)` `  print(avg)` `  if(avg >= 0.5): # if the prediction contain at least 50% True count over the predictions then it is a violation` `  return True` `  else:` `  return False` `  return False`

Table 25 Average prediction Methodology

Function	Calculate the average of the truly predicted frames.
Description	This function computes the violation video's final decision based on the average true count in the total frames.
Function Flow	Firstly, the model predicts the frames and inserts the prediction in the "predCountArray" array. Then, the function will start counting the number of "True" frames which have more than 0.37 prediction values. Then, it will count the number of "True" frames which have more than 0.70 prediction value. After that, if the video has one or more "True" frames, the average will be calculated based on the number of "True" frames that have more than 0.70 prediction value over the total number of "True" frames. If the average is more or equal to 0.50 then the function will return True -which means the video is classified as "Violation". If the average is less than 0.50 the function will return False -which means the video is classified as "Not Violation". If the video does not have any "True" frame, the function will return False -which means the video is classified as "Not Violation".
Code	

```
# this function calculate the prediction array average based on the high violation prediction value(0.7 or more)/predicted true(0.37 or more)
def countTruePredAvg(Q):
  count = 0 # count the total true predictions
  countH = 0 # count the high value predictions whic is 0.7 or more
  for i in(range(0,len(Q))):
    if(Q[i][0] >= 0.37): # if the current index prediction value was at least 0.37 then it was predicted as a violation
      count += 1 # increment the count

  for i in (range(0,len(Q))):
    if(Q[i][0] >= 0.7): # if the prediction was with high value
      countH += 1 # increment the countH

  if(count > 0):
    avg = countH/count
    if(avg >= 0.5):
      return True
    else:
      return False
  else:
    return False

  return False
```

Table 26 Average true count Methodology

Function	Calculate the true frames that have more than 0.70 prediction value
Description	This function computes the violation video's final decision based on the number of the true frames that have more than 0.70 prediction value.
Function Flow	Firstly, the model predicts the frames and inserts the prediction in the "predCountArray" array. Then, it will count the number of "True" frames which have more than 0.70 prediction values. Lastly, if the number of "True" frames that have more than 0.70 prediction value is more than or equal to four then the function will return True -which means the video is classified as "Violation"-, and otherwise it will return False -which means the video is classified as "Not Violation"-.
Code	`# this function to calculate how many high prediction values we have` `def countTruePred(Q):` `  countTrue = 0` `  for i in range(0,len(Q)):` `  if(Q[i][0]>=0.7):` `  countTrue +=1` `  if(countTrue >= 4): # if we have 4 high value predictions in a video then it is a violation` `  return True` `  else:` `  return False` `  return False`

Table 27 True count with high threshold rate Methodology

Function	Calculate and estimate the overall prediction.
Description	This function computes the final decision for the violation video every 5 seconds.
Function Flow	Firstly, the model predicts the frames and inserts the prediction in the “predCountArray” array. Then, estimate the violation occurrence for each second (three frames). If there were at least two frames that were classified as violations, then it is initially true; otherwise, it is initially false. Then, we will append the initial decision to the “decisionArray” After that, in the “decisionArray” we estimated that 5 indices would approximately represent 5 seconds, so we give a final decision for each 5 seconds as if there were more than true counts in these 5 seconds, then it is estimated that it has a violation in 2 seconds, or else the 5 seconds do not contain a violation. If one of the 5 seconds produces a true value, then the video is a violation, or else it is not a violation.

Code

```
# This function to determine the decison of the video file 'True' => 'violation', 'False' => 'Not violation(Normal)'
# Based on the predictions first its take each 3 frames together(1s) and define the final prediction(if more than 2 trues then set the decision array to true, else set to false)
# Second its devide the decision array to 5 chunks and if in the five chunks there is 2 or more trues then it is a violation, else look for the next 5 chunk
def calcDecision(predCountArray):
    countTrue = 0 # this counts number of trues in the 3 frames
    iter = 0 # this to take the decision after 3 frames passed
    decsionArray = [] # this is an array that have the decion for each 3 frames
    for i in (range(0,len(predCountArray))): # to loop through the predictions
      countTrue = 0
      iter +=1
      if(i == (len(predCountArray)-1)): # if it was the last index then we need to check for previous missed prediction
        if(iter == 3): # we wase already in the third prediction
         print("nothing continue")
        if(iter ==1): # we are in the first interation
          if(predCountArray[i]== "True"):
            decsionArray.append("True")
            break # take the decision and break from the loop
          else:
            decsionArray.append("False")
            break
        if(iter ==2): # we are in the third, and we missed 2 iterations
          for j in (range(1,iter+1)):
            if(predCountArray[i]== "True"):# the current index
              countTrue +=1
            if(predCountArray[i-1]== "True"): # the previously missed index
               countTrue +=1
            if(countTrue >= 2): # take decision for the frames
              decsionArray.append("True")
              break
            else:
              decsionArray.append("False")
              break
      if(iter == 3):# we qre in the third prediction
        iter = 0
        if(predCountArray[i] == "True"): # the current index
            countTrue +=1
        if(predCountArray[i-1] == "True"):
            countTrue +=1
        if(predCountArray[i-2] == "True"):
            countTrue +=1
        if(countTrue >= 2):
            decsionArray.append("True")
        else:
          decsionArray.append("False")

    finalCount = 0 # count of trues in a chunck
    fiveChunks = 1
    lengthOftheArray = len(decsionArray)%5 # to get the reminder
    if(lengthOftheArray == 0): # if the reminder was 0
      for l in (range(0,len(decsionArray))): # to loop through each 5 decisions
        if(decsionArray[l] == "True"):
          finalCount +=1
        if(fiveChunks == 5):
          if(finalCount>=2):
            finalCount = 0
            return True
            break
          else:
            fiveChunks = 0
            finalCount = 0
            continue
        fiveChunks +=1
    else: # if there was a reminder
        fiveChunks = 1
        for i in (range(0,(len(decsionArray)-lengthOftheArray))): # first we loop till the last index that mutiblies of 5
          if(decsionArray[i] == "True"):
           finalCount +=1
          if(fiveChunks == 5):
           if(finalCount>=2):
            finalCount = 0
            return True
            break
           else:
            fiveChunks = 0
            finalCount = 0
          fiveChunksk +=1
        fiveChunks =  1
        for j in (range((len(decsionArray)-lengthOftheArray),len(decsionArray))): # to loop through the reminder ndices
          if(decsionArray[j] == "True"):
            finalCount +=1
          if(fiveChunks == lengthOftheArray-1):
           if(finalCount>=2):
            finalCount = 0
            return True
            break
           else:
            fiveChunks = 0
            finalCount = 0
          fiveChunks +=1
    return False
```

Table 28 The Five Seconds Decision Methodology

5.1.9 Contribution

Finally, we have presented a deep learning approach for detecting traffic violations which are: drifting, driving in the opposite direction and overtaking and we try to cover all the conditions like: traffic jam, day and night without the need for human intervention.

we also introduced a new dataset scraped from twitter that contains over a thousand of different traffic violations video scenes which can be used for further improvement and enhancement of the machine learning model to detect traffic violations using a dashcam.

Lastly, as a “Novel solution” we believe that the obtained results are acceptable for the moment, but it is definitely needs to be enhanced. The traffic violation detection model still suffers from the dashcam and its quality, it is needs at least full HD dashcam to be able to detect the traffic violations on the road especially at night where the quality of the dashcam vision decreases.

5.2 User Acceptance Testing

In this subsection, we will present the user acceptance testing tables for each driver, which demonstrate their interaction with “RASD” application. Also, we will present the demographics of the drivers to whom we applied the test, and we will analyze their testing experiences through the questionnaire that we distributed among them.

Version	English		
Driver's name	Sarah Bin Mutaieb		
Driving Duration	30 minutes		
Task	# Of errors	Time needed min:sec.msec	Completion status
Register to the application.	0	01:11.46	Pass
Link dashcam to the application	0	03:30.02	Pass
View pending videos	0	00:30.66	Pass
Confirm pending video	0	00:07.23	Pass
View another pending video	0	00:46.69	Pass
Delete pending video	0	00:10.73	Pass
View confirmed report	0	00:17.55	Pass
Edit confirmed report	-	-	-
Share confirmed report	0	00:19.15	Pass
Delete confirmed report	-	-	-
Verify driver's phone number	0	00:27.86	Pass
Update driver's information	0	00:26.18	Pass
Log out of the application.	0	00:07.53	Pass
Reset Password	0	00:42.50	Pass
Log in to the application.	0	00:13.88	Pass
Total time (must be within 10 minutes)	10:51.44		
Testing Notes	After 30 minutes of driving, the model detected 14 violations, of which 6 were misclassified and 8 were accurate. Three of the correct violations were		

	specified by the tester as “driving in the opposite direction” and “overtaking” violations, while the rest were specified as others by the tester.
Correctly classified as violation	
 *Figure 35 Testing Result from Driver 1*	

Table 29 User Acceptance Testing for driver 1

Version	English		
Driver's name	May AlAmri		
Driving duration	30 minutes		
Task	# Of errors	Time needed min:sec.msec	Completion status
Register to the application.	0	01:44.07	Pass
Link dashcam to the application	0	01:34.56	Pass
View pending videos	0	00:11:37	Pass
Confirm pending video	-	-	-
View another pending video	0	00:08.08	Pass
Delete pending video	0	00:08.29	Pass
View confirmed report	-	-	-
Edit confirmed report	-	-	-
Share confirmed report	-	-	-
Delete confirmed report	-	-	-
Verify driver's phone number	0	01:08.98	Pass
Update driver's information	0	00:20.45	Pass
Log out of the application.	0	00:04.05	Pass
Reset Password	0	00:51.66	Pass
Log in to the application.	0	00:45.97	Pass
Total time (must be within 10 minutes)	06:57.48		
Testing Notes	The testing duration was 30 minutes and out of the recorded videos while the driver -May- was driving, only 2 was detected as violations by the violation		

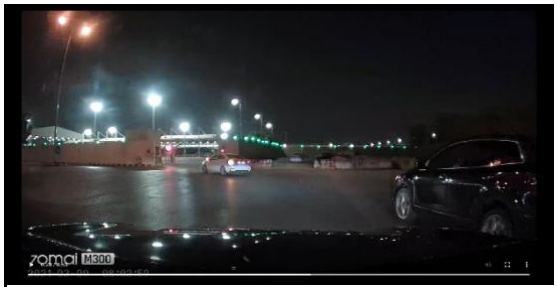
	detection model and after viewing the 2 videos the driver decided that it is not violations and deleted them accordingly.
Incorrectly classified as violation	
 *Figure 36 Testing Result from Driver 2*	 *Figure 37 Second Testing Result from Driver 2*

Table 30 User Acceptance Testing for Driver 2

Version	English		
Driver's name	Abdulmajeed AlMaymoon		
Testing Duration	40 minutes		
Task	# Of errors	Time needed min:sec.msec	Completion status
Register to the application.	0	01:22.70	Pass
Link dashcam to the application	0	01:23.34	Pass
View pending videos	0	00:19.81	Pass
Confirm pending video	-	-	-
View another pending video	0	00:08.08	Pass
Delete pending video	0	00:08.76	Pass
View confirmed report	-	-	-
Edit confirmed report	-	-	-
Share confirmed report	-	-	-
Delete confirmed report	-	-	-
Verify driver's phone number	0	00:35.09	Pass
Update driver's information	0	01:09.30	Pass
Log out of the application.	0	00:09.05	Pass
Reset Password	0	00:22.98	Pass
Log in to the application.	0	00:21.82	Pass
Total time (must be within 10 minutes)	05:00.93		
Testing Notes	The driving session lasts for 40 minutes. The violation detection model detected 1 video as a violation while it was not (False Positive). The driver -		

	Abdulmajeed- realized that no violation occurred in his way which means that the rest of the videos was correctly classified.
Incorrectly classified as violation	
 *Figure 38 Testing Result from Driver 3*	 *Figure 39 Second Testing Result from Driver 3*

Table 31 User Acceptance Testing for Driver 3

Version	English		
Driver's name	Deemah AlHamlan		
Testing Duration	11 minutes		
Task	# Of errors	Time needed min:sec.msec	Completion status
Register to the application.	0	01:01.87	Pass
Link dashcam to the application	0	01:02.90	Pass
View pending videos	0	00:08.50	Pass
Confirm pending video	-	-	-
View another pending video	-	-	-
Delete pending video	0	00:04.33	Pass
View confirmed report	-	-	-
Edit confirmed report	-	-	-
Share confirmed report	-	-	-
Delete confirmed report	-	-	-
Verify driver's phone number	0	00:16.10	Pass
Update driver's information	0	00:13.18	Pass
Log out of the application.	0	00:30.00	Pass
Reset Password	0	00:21.65	Pass
Log in to the application.	0	00:17.70	Pass
Total time (must be within 10 minutes)	04:56.23		
Testing Notes	Deemah was driving for 11 minutes; the driving session was at evening and it was around the neighborhood. 4 of the recorded videos were considered as		

	a violation which were deleted by the driver -Deemah- and considered as false positive, these videos included parking cars against the view of the dashcam which may confuse the machine learning model and leads to detect them as a violation.
Incorrectly classified as violation	
 *Figure 40 Testing Result from Driver 4*	 *Figure 41 Second Testing Result from Driver 4*

Table 32 User Acceptance Testing for Driver 4

Version	English		
Driver's name	Jood AlHamlan		
Testing Duration	14 minutes		
Task	# Of errors	Time needed min:sec.msec	Completion status
Register to the application.	0	01:21.27	Pass
Link dashcam to the application	0	00:43.60	Pass
View pending videos	0	00:51.51	Pass
Confirm pending video	0	00:05.06	Pass
View another pending video	0	00:15.05	Pass
Delete pending video	0	00:20.91	Pass
View confirmed report	0	00:10.71	Pass
Edit confirmed report	0	00:10.46	Pass
Share confirmed report	0	00:19.45	Pass
Delete confirmed report	0	00:11.59	Pass
Verify driver's phone number	0	00:23.47	Pass
Update driver's information	0	00:09.36	Pass
Log out of the application.	0	00:15.60	Pass
Reset Password	0	00:25.12	Pass
Log in to the application.	0	00:12.29	Pass
Total time (must be within 10 minutes)	05:55.45		

Testing Notes	The driving session for Jood lasts for 14 minutes and 7 of the recorded videos was considered violations since Jood drove through a cross-road at rush hours.
Correctly classified as violation	
 *Figure 42 Testing Result from Driver 5*	 *Figure 43 Second Testing Result from Driver 5*

Table 33 User Acceptance Testing for Driver 5

Version	English		
Driver's name	Latifah Almurshed		
Driving duration	30 minutes		
Task	# Of errors	Time needed min:sec.msec	Completion status
Register to the application.	0	01:23.54	Pass
Link dashcam to the application	0	00:20.43	Pass
View pending videos	0	00:10.45	Pass
Confirm pending video	-	-	-
View another pending video	-	-	-
Delete pending video	0	00:05.30	Pass
View confirmed report	-	-	-
Edit confirmed report	-	-	-
Share confirmed report	-	-	-
Delete confirmed report	-	-	-
Verify driver's phone number	0	00:55.94	Pass
Update driver's information	0	01:40.30	Pass
Log out of the application.	0	00:06.42	Pass
Reset Password	0	00:06.08	Pass
Log in to the application.	0	00:30.10	Pass
Total time (must be within 10 minutes)	05:19.56		

Testing Notes	We start by registering to the application and then link the camera using the 'IP Camera Lite' application to simulate the process as the driver did not have a dashcam then we drive around 30 minutes morning across the main roads and neighborhoods, we did not face any traffic violations such as : drifting, driving in the opposite direction and overtaking, but the model did classify 1 video as a violation, and the driver delete it as misclassified video.
Incorrectly classified as violation	
 *Figure 44 Testing Result from Driver 6*	 *Figure 45 Second Testing Result from Driver 6*

Table 34 User Acceptance Testing for Driver 6

Version	English		
Driver's name	Modi Aljedaie		
Driving duration	25 minutes		
Task	# Of errors	Time needed min:sec.msec	Completion status
Register to the application.	0	1:17.26	Pass
Link dashcam to the application	0	00:30.43	Pass
View pending videos	-	-	-
Confirm pending video	-	-	-
View another pending video	-	-	-
Delete pending video	-	-	-
View confirmed report	-	-	-
Edit confirmed report	-	-	-
Share confirmed report	-	-	-
Delete confirmed report	-	-	-
Verify driver's phone number	0	00:34.32	Pass
Update driver's information	0	00:42.83	Pass
Log out of the application.	0	00:09.09	Pass
Reset Password	0	00:38.23	Pass
Log in to the application.	0	00:16.68	Pass
Total time (must be within 10 minutes)	04:08.84		

Testing Notes	We start by registering to the application and then link the camera using the 'IP Camera Lite' application to simulate the process as the driver did not have a dashcam then we drive around 25 minutes morning across the main roads and neighborhoods, we did not face any traffic violations such as : drifting, driving in the opposite direction and overtaking, also the model did not classify any video as a violation, thus there was not any violation videos in the application.

Table 35 User Acceptance Testing for Driver 7

Version	English		
Driver's name	Wail Almurshed		
Driving duration	20 minutes		
Task	# Of errors	Time needed min:sec.msec	Completion status
Register to the application.	0	00:58.76	Pass
Link dashcam to the application	0	00:27.57	Pass
View pending videos	0	00:45.48	Pass
Confirm pending video	0	00:17.38	Pass
View another pending video	0	00:13.61	Pass
Delete pending video	0	00:10.50	Pass
View confirmed report	0	00:14.82	Pass
Edit confirmed report	0	00:10.23	Pass
Share confirmed report	0	00:09.69	Pass
Delete confirmed report	0	00:07.50	Pass
Verify driver's phone number	0	00:23.25	Pass
Update driver's information	0	00:58.20	Pass
Log out of the application.	0	00:05.30	Pass
Reset Password	0	00:50.44	Pass
Log in to the application.	0	00:53.86	Pass
Total time (must be within 10 minutes)	06:36.59		

Testing Notes	We start by registering to the application and then link the camera using the 'IP Camera Lite' application to simulate the process as the driver did not have a dashcam then we drive around 20 minutes morning across the main roads and neighborhoods, after finishing the driving session the driver have 3 videos classifies as violation and only 1 video was confirmed by the driver and we notice that the quality of the camera is important as it is play a main rule in the classification and the other misclassified video was almost like overtaking as shown Figure 46.

Incorrectly classified as violation	Correctly classified as violation
	
Figure 46 Testing Result from Driver 8	*Figure 47 Second Testing Result from Driver 8*

Table 36 User Acceptance Testing for Driver 8

Version	English		
Driver's name	Sarah Alodan		
Driving duration	30 minutes		
Task	# Of errors	Time needed min:sec.msec	Completion status
Register to the application.	0	01:00.04	Pass
Link dashcam to the application	0	00:27.27	Pass
View pending videos	0	00:08.70	Pass
Confirm pending video	0	00:08.47	Pass
View another pending video	0	00:21.10	Pass
Delete pending video	0	00:03.00	Pass
View confirmed report	0	00:12.19	Pass
Edit confirmed report	0	00:08.52	Pass
Share confirmed report	0	00:11.87	Pass
Delete confirmed report	0	00:06.54	Pass
Verify driver's phone number	0	00:15.79	Pass
Update driver's information	0	00:09.64	Pass
Log out of the application.	0	00:03.90	Pass
Reset Password	0	00:28.10	Pass
Log in to the application.	0	00:30.40	Pass
Total time (must be within 10 minutes)	04:15.53		

Testing Notes	We start by registering to the application and then link the camera using the 'IP Camera Lite' application to simulate the process as the driver did not have a dashcam then we drive around 30 minutes morning across the main roads and neighborhoods, after finishing the driving session the driver has 2 videos classifies as violation and two videos was confirmed by the driver the other was deleted as misclassified videos.

Table 37 User Acceptance Testing for Driver 9

Version	English		
Driver's name	Nouf Alosaimi		
Driving Duration	20 minutes		
Task	# Of errors	Time needed min:sec.msec	Completion status
Register to the application.	0	00:57:45	Pass
Link dashcam to the application	0	01:30:02	Pass
View pending videos	0	00:09:02	Pass
Confirm pending video	-	00:00:00	-
View another pending video	0	00:05:10	Pass
Delete pending video	0	00:08:81	Pass
View confirmed report	-	00:00:00	-
Edit confirmed report	-	00:00:00	-
Share confirmed report	-	00:00:00	-
Delete confirmed report	-	00:00:00	-
Verify driver's phone number	0	00:10:16	Pass
Update driver's information	0	00:20:18	Pass
Log out of the application.	0	00:05:01	Pass
Reset Password	0	00:26:38	Pass
Log in to the application.	0	00:18:36	Pass
Total time (must be within 10 minutes)	04:10.49		

Testing Notes	The total trip duration was 20 minutes. During the trip there were no violation. And out of recorded videos the model classifies 5 of them as violation. The quality of the dashcam effected the model performance.

Table 38 User Acceptance Testing for Driver 10

Version	English		
Driver's name	Abdulkarim Alosaimi		
Driving duration	43 minutes		
Task	# Of errors	Time needed min:sec.msec	Completion status
Register to the application.	0	00:54:10	Pass
Link dashcam to the application	0	01:10:02	Pass 1
View pending videos	0	00:06:02	Pass
Confirm pending video	0	00:08:10	Pass
View another pending video	0	00:05:10	Pass
Delete pending video	0	00:06:10	Pass
View confirmed report	0	00:06:40	Pass
Edit confirmed report	0	00:18:52	Pass
Share confirmed report	0	00:17:55	Pass
Delete confirmed report	0	00:06:43	Pass
Verify driver's phone number	0	00:14:20	Pass
Update driver's information	0	00:19:90	Pass
Log out of the application.	0	00:04:20	Pass
Reset Password	0	00:18:70	Pass
Log in to the application.	0	00:20:88	Pass
Total time (must be within 10 minutes)	04:36.22		

Testing Notes	The total trip duration was 43 minutes. During the trip there were one violation and the model classified it correctly and the user confirmed it. And out of the recorded videos the model classifies 10 of them as violation.
Correctly classified as violation	

Figure 48 Testing Result from Driver 11

Table 39 User Acceptance Testing for Driver 11

Version	English		
Driver's name	Abdullah Alosaimi		
Driving duration	30 minutes		
Task	# Of errors	Time needed min:sec.msec	Completion status
Register to the application.	0	00:57:14	Pass
Link dashcam to the application	0	01:30:05	Pass
View pending videos	-	00:00:00	-
Confirm pending video	-	00:00:00	-
View another pending video	-	00:00:00	-
Delete pending video	-	00:00:00	-
View confirmed report	-	00:00:00	-
Edit confirmed report	-	00:00:00	-
Share confirmed report	-	00:00:00	-
Delete confirmed report	-	00:00:00	-
Verify driver's phone number	0	00:17:09	Pass
Update driver's information	0	00:30:10	Pass
Log out of the application.	0	0:05:07	Pass
Reset Password	0	00:40:90	Pass
Log in to the application.	0	00:33:06	Pass
Total time (must be within 10 minutes)	04:33.41		

Testing Notes	The total trip duration was 30 minutes. During the trip there were 1 violation and the model classified it as normal. And the remaining videos was normal, and the model classified them correctly as normal.

Table 40 User Acceptance Testing for Driver 12

Version	English		
Driver's name	Maha Alhamlan		
Driving Duration	13 minutes		
Task	# Of errors	Time needed min:sec.msec	Completion status
Register to the application.	0	01:24.75	Pass
Link dashcam to the application	0	01:05.62	Pass
View pending videos	0	00:20.19	Pass
Confirm pending video	-	-	-
View another pending video	-	-	-
Delete pending video	0	00:08.48	Pass
View confirmed report	-	-	-
Edit confirmed report	-	-	-
Share confirmed report	-	-	-
Delete confirmed report	-	-	-
Verify driver's phone number	0	00:32.56	Pass
Update driver's information	0	00:14.44	Pass
Log out of the application.	0	00:04.29	Pass
Reset Password	0	00:25.60	Pass
Log in to the application.	0	00:23.71	Pass
Total time (must be within 10 minutes)	04:39.64		
Testing Notes	After 13 minutes of driving, the model detected three violations, which was misclassified.		

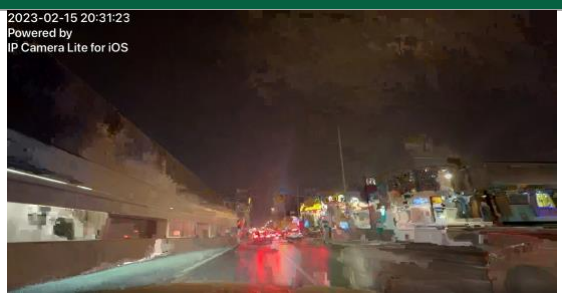
Incorrectly classified as violation
 *Figure 49 Testing Result from Driver 13*

Table 41 User Acceptance Testing for Driver 13

Version	Arabic		
Driver's name	Fahad Bin Mutaieb		
Driving Duration	11 minutes		
Task	# Of errors	Time needed min:sec.msec	Completion status
Register to the application.	0	02:48.22	Pass
Link dashcam to the application	0	02:06.71	Pass
View pending videos	0	00:08.65	Pass
Confirm pending video	0	00:06.56	Pass
View another pending video	0	00:46.21	Pass
Delete pending video	-	-	-
View confirmed report	0	00:17.62	Pass
Edit confirmed report	0	00:20.43	Pass
Share confirmed report	0	00:22.95	Pass
Delete confirmed report	0	00:08.46	Pass
Verify driver's phone number	0	00:47.66	Pass
Update driver's information	0	00:19.26	Pass
Log out of the application.	0	00:06.63	Pass
Reset Password	0	00:43.24	Pass
Log in to the application.	0	00:25.93	Pass
Total time (must be within 10 minutes)	09:28.53		
Testing Notes	After 11 minutes of driving, the model detected two violations, all of which are accurate, but one violation was misclassified as not a violation while it		

	was. Also, the tester had difficulty reading the text in the application due to its small size.
Correctly classified as violation	

Figure 50 Testing Result from Driver 14

Table 42 User Acceptance Testing for Driver 14

Version	Arabic		
Driver's name	Maram Bin Mutaieb		
Driving Duration	11 minutes		
Task	# Of errors	Time needed min:sec.msec	Completion status
Register to the application.	0	02:17.72	Pass
Link dashcam to the application	0	01:30.13	Pass
View pending videos	0	00:33.61	Pass
Confirm pending video	0	00:05.20	Pass
View another pending video	0	00:34.05	Pass
Delete pending video	0	00:05.90	Pass
View confirmed report	0	00:16.45	Pass
Edit confirmed report	0	00:26.15	Pass
Share confirmed report	0	00:12.11	Pass
Delete confirmed report	0	00:09.51	Pass
Verify driver's phone number	0	00:45.50	Pass
Update driver's information	0	00:22.29	Pass
Log out of the application.	0	00:04.40	Pass
Reset Password	0	00:34.33	Pass
Log in to the application.	0	00:24.85	Pass
Total time (must be within 10 minutes)	08:22.19		
Testing Notes	After 11 minutes of driving, the model detected eight violations, seven of which were misclassified, but one was accurate. Also, the tester was confused		

	about the phone number verification process and thought that she needed to re-enter her phone number first in order to verify it.
Incorrectly classified as violation	**Correctly classified as violation**
 *Figure 51 Testing Result from Driver 15*	 *Figure 52 Second Testing Result from Driver 15*

Table 43 User Acceptance Testing for Driver 15

Version	Arabic		
Driver's name	Abrar AlAbikah		
Testing Duration	35 minutes		
Task	# Of errors	Time needed min:sec.msec	Completion status
Register to the application.	0	01:45.82	Pass
Link dashcam to the application	0	02:04.78	Pass
View pending videos	0	00:11.24	Pass
Confirm pending video	0	00:06.69	Pass
View another pending video	0	00:38.01	Pass
Delete pending video	0	00:08.94	Pass
View confirmed report	0	00:11.83	Pass
Edit confirmed report	0	00:21.78	Pass
Share confirmed report	0	00:39.70	Pass
Delete confirmed report	0	00:16.39	Pass
Verify driver's phone number	0	00:34.65	Pass
Update driver's information	0	00:25.65	Pass
Log out of the application.	0	00:06.63	Pass
Reset Password	0	00:33.90	Pass
Log in to the application.	0	00:40.80	Pass
Total time (must be within 10 minutes)	08:46.81		
Testing Notes	This driving session lasts for 35 minutes. The violation detection model predicted that two of the recorded videos were a real violation and show it		

	for the driver for further confirmation while the remaining recorded videos was correctly classified as non-violation videos. The two violation videos were examined by the driver -Abrar- and she confirmed one of the videos as a real violation while the other one she thought that because the video was in a neighborhood the model predicted the stopping cars Infront of its view as violations.
Incorrectly classified as violation	**Correctly Classified as violation**
 *Figure 53 Testing Result from Driver 16*	 *Figure 54 Second Testing Result from Driver 16*

Table 44 User Acceptance Testing for Driver 16

Version	Arabic		
Driver's name	Nasser AlJraythi		
Testing Duration	25 minutes		
Task	# Of errors	Time needed min:sec.msec	Completion status
Register to the application.	0	02:09.51	Pass
Link dashcam to the application	0	01:08.55	Pass
View pending videos	-	-	-
Confirm pending video	-	-	-
View another pending video	-	-	-
Delete pending video	-	-	-
View confirmed report	-	-	-
Edit confirmed report	-	-	-
Share confirmed report	-	-	-
Delete confirmed report	-	-	-
Verify driver's phone number	0	00:36.41	Pass
Update driver's information	0	00:28.39	Pass
Log out of the application.	0	00:04.15	Pass
Reset Password	0	00:59.78	Pass
Log in to the application.	0	00:10.69	Pass
Total time (must be within 10 minutes)	05:29.48		

Testing Notes	We drove with Nasser around 25 minutes and all of recorded videos were normal driving actions (not violations) and correctly classified by the model as the driver -Nasser- was driving in the morning in a clear road.

Table 45 User Acceptance Testing for Driver 17

Version	Arabic		
Driver's name	Joud Aljedaie		
Driving duration	20 minutes		
Task	# Of errors	Time needed min:sec.msec	Completion status
Register to the application.	0	00:59.03	Pass
Link dashcam to the application	0	00:40.57	Pass
View pending videos	0	00:07.10	Pass
Confirm pending video	-	-	-
View another pending video	0	00:10.20	Pass
Delete pending video	0	00:05.20	Pass
View confirmed report	-	-	-
Edit confirmed report	-	-	-
Share confirmed report	-	-	-
Delete confirmed report	-	-	-
Verify driver's phone number	0	00:14.84	Pass
Update driver's information	0	00:40.09	Pass
Log out of the application.	0	00:05.30	Pass
Reset Password	0	00:57.62	Pass
Log in to the application.	0	00:08.60	Pass
Total time (must be within 10 minutes)	04:08.55		

Testing Notes	We start by registering to the application and then link the camera using the 'IP Camera Lite' application to simulate the process as the driver did not have a dashcam then we drive around 20 minutes morning across the main roads and neighborhoods, we did not face any traffic violations such as : drifting, driving in the opposite direction and overtaking, also the model did not classify any video as a violation, thus there was not any violation videos in the application.

Table 46 User Acceptance Testing for Driver 18

Version	Arabic		
Driver's name	Amal Almurshed		
Driving duration	20 minutes		
Task	# Of errors	Time needed min:sec.msec	Completion status
Register to the application.	0	03:15.23	Pass
Link dashcam to the application	0	00:17.83	Pass
View pending videos	0	00:14.07	Pass
Confirm pending video	-	-	-
View another pending video	0	00:14.45	Pass
Delete pending video	0	00:04.18	Pass
View confirmed report	-	-	-
Edit confirmed report	-	-	-
Share confirmed report	-	-	-
Delete confirmed report	-	-	-
Verify driver's phone number	0	01:02.12	Pass
Update driver's information	0	00:34.21	Pass
Log out of the application.	0	00:06.62	Pass
Reset Password	0	00:46.13	Pass
Log in to the application.	0	00:42.67	Pass
Total time (must be within 10 minutes)	07:17.39		

	We start by registering to the application and then link the camera using the 'IP Camera Lite' application to simulate the process as the driver did not have a dashcam then we drive around 20 minutes afternoon across the main roads and neighborhoods, we did not face any traffic violations such as : drifting, driving in the opposite direction and overtaking, but the model classifies 2 videos as a violation and the driver delete them as misclassified videos.

Table 47 User Acceptance Testing for Driver 19

Version	Arabic		
Driver's name	Reyoof Altaweel		
Driving duration	20 minutes		
Task	# Of errors	Time needed min:sec.msec	Completion status
Register to the application.	0	02:20.40	Pass
Link dashcam to the application	0	00:40.83	Pass
View pending videos	0	00:30.07	Pass
Confirm pending video	-	-	-
View another pending video	0	00:20.88	Pass
Delete pending video	0	00:06.14	Pass
View confirmed report	-	-	-
Edit confirmed report	-	-	-
Share confirmed report	-	-	-
Delete confirmed report	-	-	-
Verify driver's phone number	0	00:59.12	Pass
Update driver's information	0	00:40.49	Pass
Log out of the application.	0	00:11.10	Pass
Reset Password	0	00:43.52	Pass
Log in to the application.	0	00:57.07	Pass
Total time (must be within 10 minutes)	07:29.62		

Testing Notes	We start by registering to the application and then link the camera using the 'IP Camera Lite' application to simulate the process as the driver did not have a dashcam then we drive around 20 minutes morning across the main roads and neighborhoods, we did not face any traffic violations such as : drifting, driving in the opposite direction and overtaking, but the model classifies 2 videos as a violation and the driver delete them as misclassified videos.

Table 48 User Acceptance Testing for Driver 20

Version	Arabic		
Driver's name	Salma Bin Mutaieb		
Driving duration	46 minutes		
Task	# Of errors	Time needed min:sec.msec	Completion status
Register to the application.	0	1:46.17	Pass
Link dashcam to the application	0	02:27.10	Pass
View pending videos	0	00:14.43	Pass
Confirm pending video	0	00:26.31	Pass
View another pending video	0	00:08.18	Pass
Delete pending video	0	00:03.81	Pass
View confirmed report	0	00:26.48	Pass
Edit confirmed report	0	00:13.71	Pass
Share confirmed report	0	00:20.43	Pass
Delete confirmed report	0	00:07.90	Pass
Verify driver's phone number	0	00:40.10	Pass
Update driver's information	0	00:46.60	Pass
Log out of the application.	0	00:07.73	Pass
Reset Password	0	00:40.45	Pass
Log in to the application.	0	00:20.26	Pass
Total time (must be within 10 minutes)	08:49.66		

Testing Notes	After 46 minutes of driving, the model detected eight violations, of which seven were misclassified and one was accurate. Also, the tester needed more guidance regarding the dashcam linking process. When the tester was asked to register for the application, she made a mistake and entered her information on the login page rather than the sign-up page; additionally, the tester thought that the dashcam information button was difficult to find.

Incorrectly classified as violation	Correctly classified as violation
 *Figure 55 Testing Result from Driver 21*	 *Figure 56 Second Testing Result from Driver 21*

Table 49 User Acceptance Testing for Driver 21

5.2.1 Demographics of Participants

We will provide in this sub section a statistical data relating to the audience who is going to use our application “RASD”.

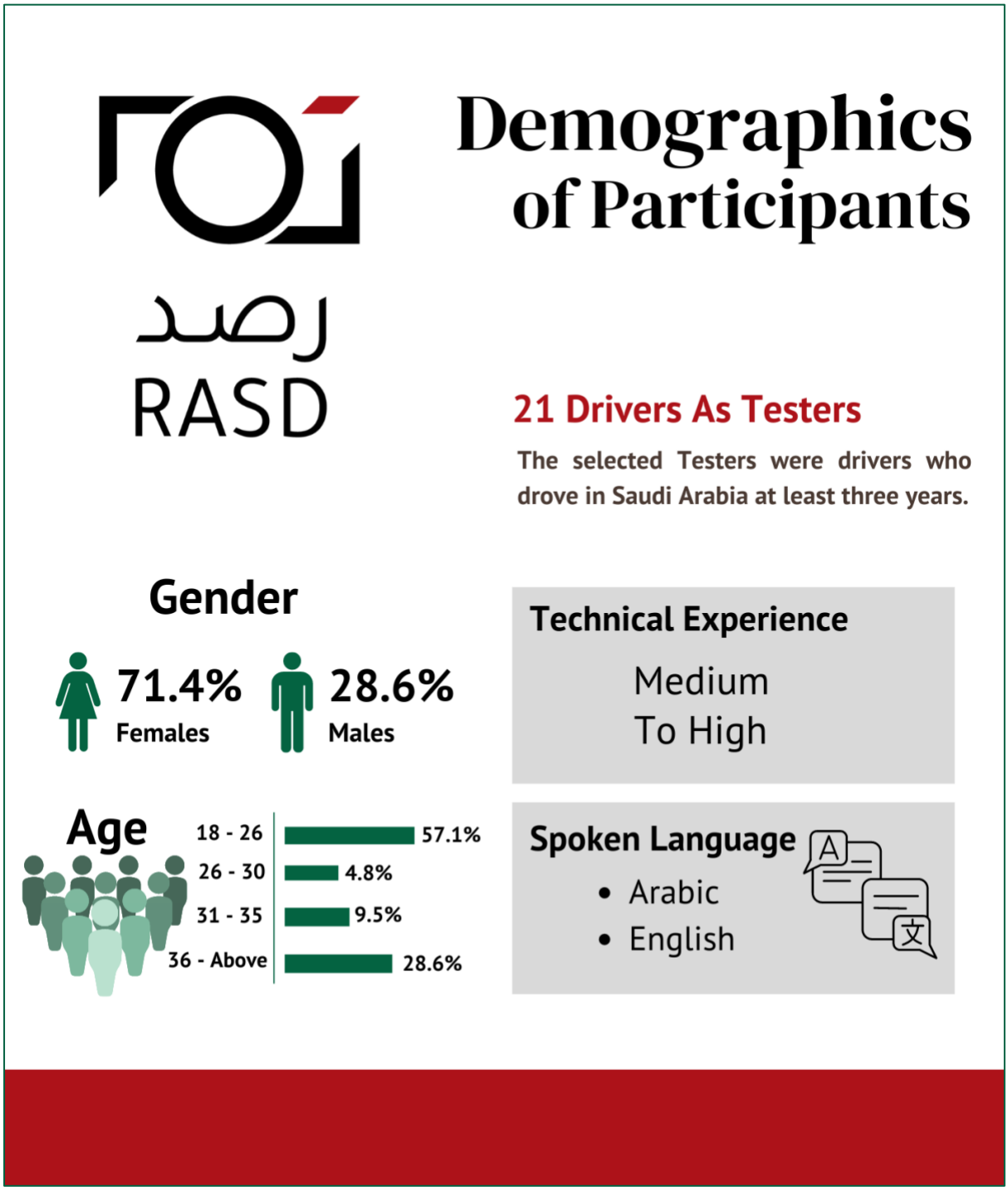


Figure 57 "RASD" Demographics of Participants

5.2.2 Questionnaire Results

We have tested our application “RASD” on twenty-one drivers, and then in order to collect data about their experience using “RASD” we prepared a questionnaire composed of 14 questions attached in Appendix D: User Acceptance Testing Feedback, the results come as follows:

Figure 58 shows that 71.4% of our drivers were female and the rest, 28.6%, were male. Those drivers were from different ranges of ages, and from Figure 59Figure 85, we saw that the majority of our drivers were between 18 and 26 years old, with a percentage of 57.1%. Moreover, more than half of the drivers speak both Arabic and English, with a percentage of 76.2%, and 23.8% of them speak Arabic only, as shown in Figure 60, which makes “RASD” suitable for them since it provides an Arabic interface as well as an English one. Figure 61 illustrates that only a few of the drivers, with a percentage of 4.8%, strongly agree that they need someone to show them how to use the application, while a large percentage of them (52.4%) do not agree with that statement strongly. We tried to understand our users' perspective on our application and its ease of use for them as a naïve user. From Figure 62, we observed that 85.7% of them think that “RASD” application is easy to use, disagreeing with the statement that says, “It will take me a few more trials to master the application.” And only a few of them, with a percentage of 14.3%, think that they need guidance. Even those who need guidance agreed with the rest of the drivers that the next time they use “RASD” application, they will know exactly what to do, and it will be easy for them to use it as shown in Figure 63. Given that “RASD” application relies on linking the dashcam to it and that it is one of the main functions, we tried to measure how easy it was for the drivers to link the dashcam by asking them about the provided information about the procedure. 57.1% of them strongly agreed, and the rest agreed only as shown in Figure 64. This leads us to the conclusion that the drivers understood the information provided about the dashcam. Furthermore, we provide a small screen on the home page for the user to keep track of the stream and ensure it is running, and their dashcams are connected to “RASD” application, so we asked if the mentioned feature is useful to them or not. From Figure 65, we can notice that all of the drivers were pleased by that feature since it made it easy for them to check the status of the live stream. Moreover, none of the drivers faced any difficulties on the page where the pending violation videos were

displayed. According to Figure 66, 71.4% of them strongly agree that it was clear and simple to use. Also, according to the drivers, deleting the pending violation videos was easy; 76.2% strongly agreed with that, as shown in Figure 67. Similarly, confirming the pending violation video was easy according to the drivers, who strongly agreed on it by a percentage of 71.4%, as shown in Figure 68. Furthermore, since "RASD" generates a report for each confirmed violation, we wanted to be sure that the report maintained all the needed information, so 76.2% of the drivers strongly agreed on it, as shown in Figure 69, if we exclude those who did not encounter any violations from having a report about it. Also, the majority of the drivers think it is easy to edit the violation report, with a percentage of 66.7%, as shown in Figure 70. Finally, we wanted to make sure that the drivers had solid knowledge about how the report will be shared with authorities, so we measured their understanding of the "share" button and whether it was an indicator for sharing the report with authorities via any email provider. It seems like they have that knowledge about the "share" button, as all of them vary from strongly agreeing to agreeing only with a percentage of 61.9% and 14.3% as illustrated in Figure 71, respectively.

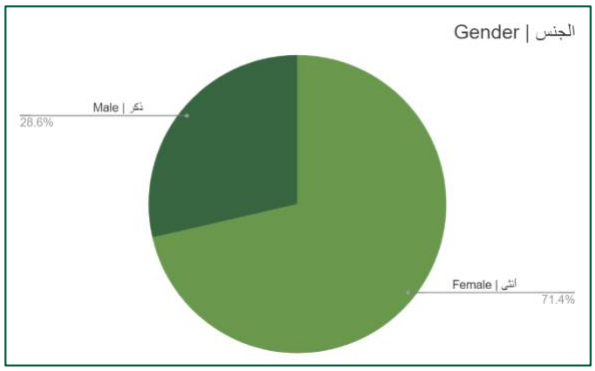


Figure 58 First Question

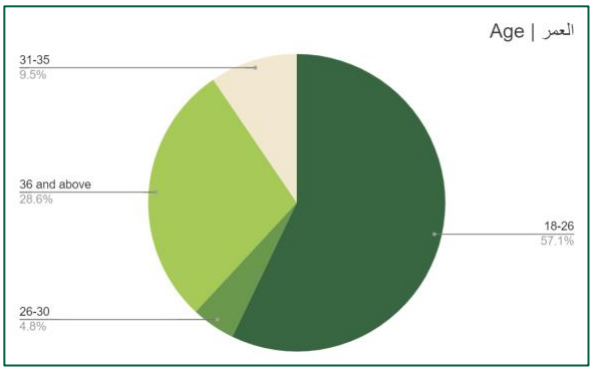


Figure 59 Second Question

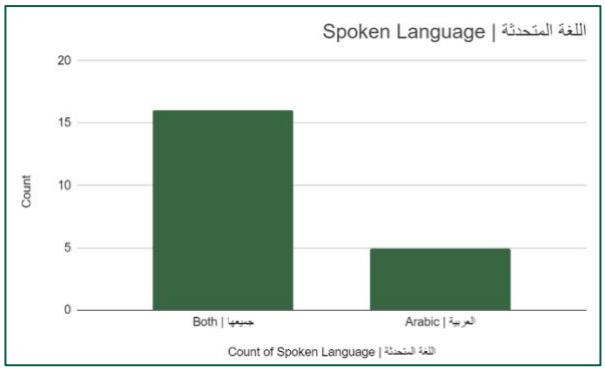


Figure 60 Third Question

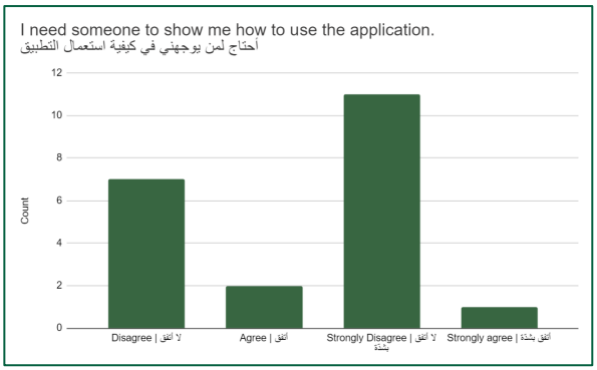


Figure 61 Fourth Question

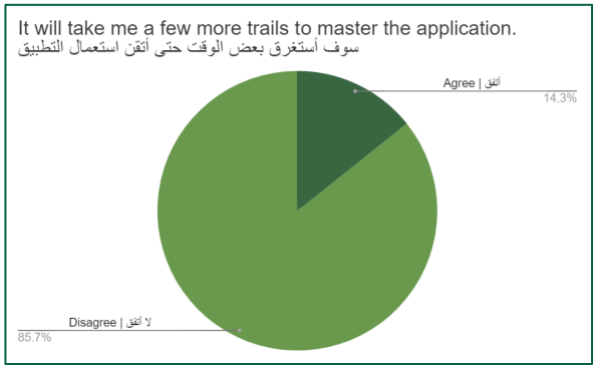


Figure 62 Fifth Question

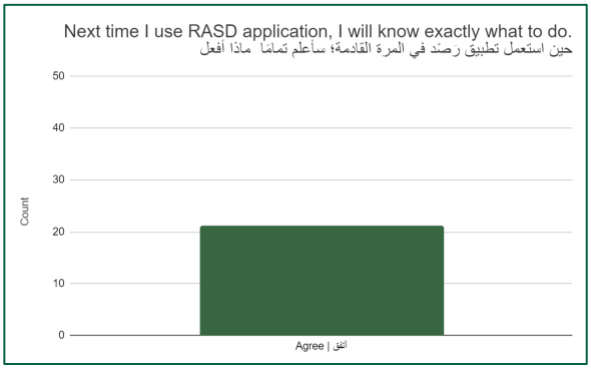


Figure 63 Sixth Question

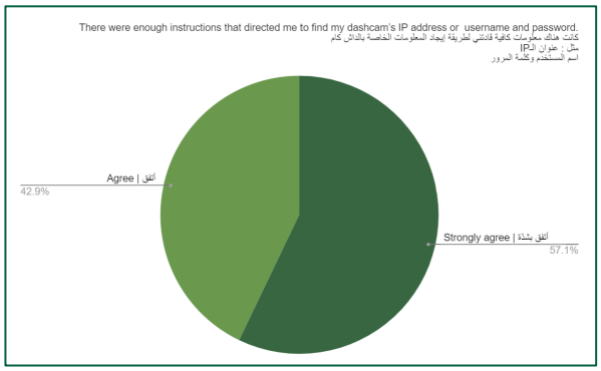


Figure 64 Seventh Question

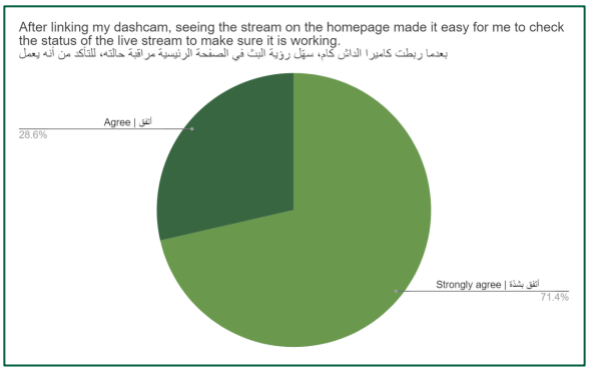


Figure 65 Eighth Question

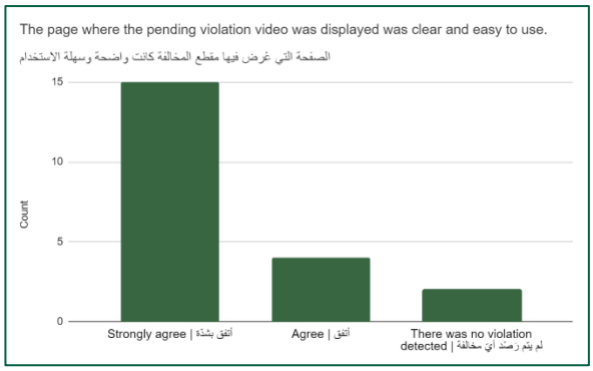


Figure 66 Ninth Question

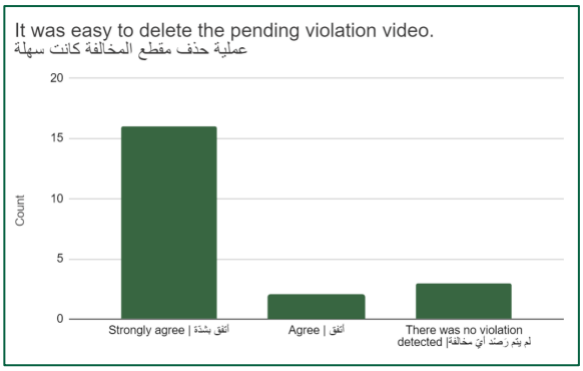


Figure 67 Tenth Question

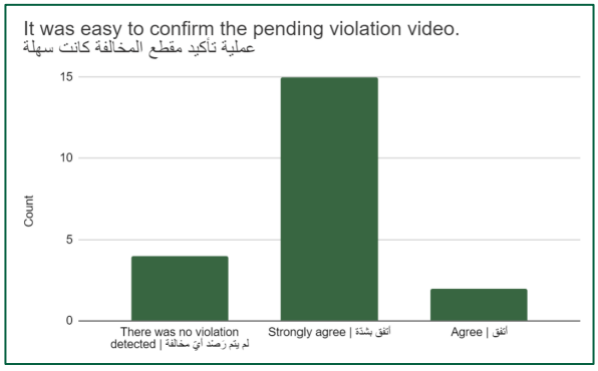


Figure 68 Eleventh Question

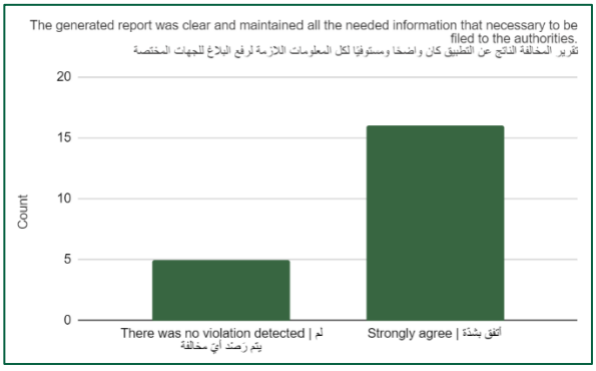


Figure 69 Twelfth Question

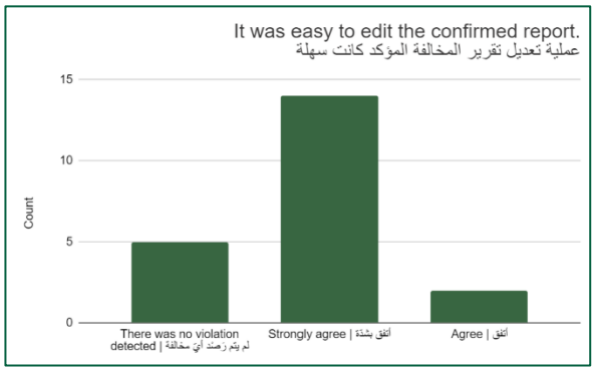


Figure 70 Thirteenth Question

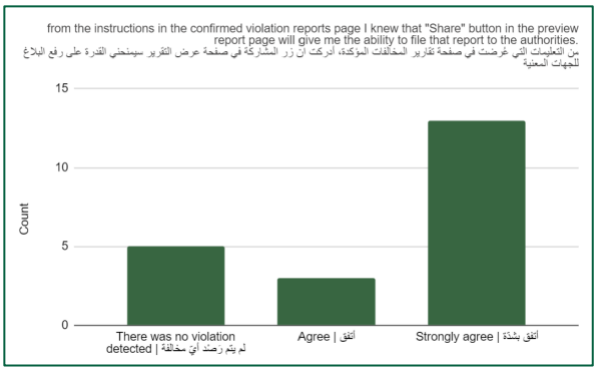


Figure 71 Fourteenth Question

5.3 Quality Attributes (NFR testing)

In this subsection we explain how we tested non-functional requirements and their results as shown in Table 50.

User story	Quality Attribute	Measure	Results
As a driver, I want my password to be concealed during register/login so that I can assure no one else knows it.	Security: How well are the system and its data protected against attacks?[46]	The password must be invisible while the drive is registering or logging in.	All the entered passwords are secured and replaced by dots when the driver enter them automatically Figure 72. 21 drivers were subjected and all of them experienced the same level of security as all of the password input fields were represented as dots.
As a driver, I want my sessions to be inactivated and destroyed after I log out so that no one can exploit my previous active sessions to enter into the application.	Security: How well are the system and its data protected against attacks? [46]	The driver session must be inactivated and destroyed which means no one can exploit the session after logging out from "RASD" application.	All the 21 driver's session expired after logging out from the application. The maximum response time was in 00:30.00, the average response time was in 00:07.38 and the

			minimum response time was in 00:03.90.
As a driver, I want to be able to see the violation records in range between 2 to 3 minutes while having a stable internet connection so that I do not waste time waiting for the record to be uploaded.	Performance: How fast does the system return results?[46]	The driver must see the video in less than or equal 3 minutes.	All the 21 drivers viewed the violation videos in range between 00:06.02 to 00:51.51 The maximum response time was in 00:51.51, the average response time was in 00:19.00 and the minimum response time was 00:06.02.
As a driver, I want to be able to use all the application functions for the first time within 10 minutes so that I do not get overwhelmed and drop the thought of using the application to report violations.	Learnability: How fast is it for users to complete the main actions once they see the interface? [46]	Compute the time it takes the driver to use all the application functions. The computed time must be 10 minutes or less.	All the 21 drivers completed all actions in 10:51.44 minutes or less. The maximum completion time was in 10:51.44, the average completion time was in 05:07.00 and the minimum completion time was in 04:08.84.

Table 50 NFR testing

Based on Table 50, we observed that “RASD” drivers have cleared the non-functional requirements testing with success, as indicated by the results column. For each non-functional requirement, we computed the minimum, maximum, and average time across all 21 drivers, and the resulting durations were deemed satisfactory. As for the first non-functional requirement regarding “Security” we could not measure the mentioned durations because it's instantly reflected in the interface Figure 72, as it should be.

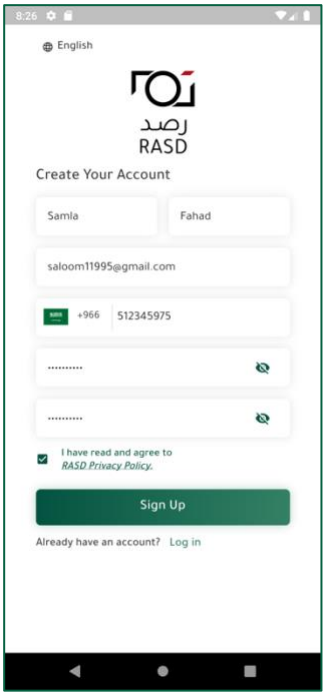


Figure 72 Sign up password

5.4 Discussion

In this subsection, we will provide an interpretation of the results presented in the previous sections. In general, the results obtained from the system evaluation phase were considered good. First of all, by looking at the user acceptance tables provided in User Acceptance Testing we can observe that we fulfilled all of our non-functional requirements and its acceptance criteria. Also, all the functions have been passed by the drivers without any errors and in an acceptable time. Moreover, we distributed a questionnaire for the drivers to provide feedback about their experience using “RASD” and in Questionnaire Results we discussed the obtained results. From the received responses, all drivers have given positive feedback about the application interface and the functions that we have tested them on.

Also, some drivers provided us with useful feedbacks on how we could improve our application, the feedback we received regarding the homepage pertained to drivers' preference for replacing the label “pending videos” with “violation videos” accompanied by a note to ensure that users are aware of the need for confirmation. Drivers expressed the belief that an inexperienced user might not readily grasp the meaning of “pending” and as their primary objective was to view violation videos, this should be the prominent feature highlighted in the interface. We considerably adjusted the interface and implemented their feedbacks as shown in Figure 74. In addition, they felt that it would be beneficial to have a means of communication readily available on the homepage, in addition to the existing option within the settings. They believed that an issue might arise, or they may have questions about the application, and a straightforward way to access support would be convenient. To address this, we implemented a “Need Help?” button, which, upon being clicked, will direct the user to “RASD” account's Direct Messages on Twitter Figure 74.

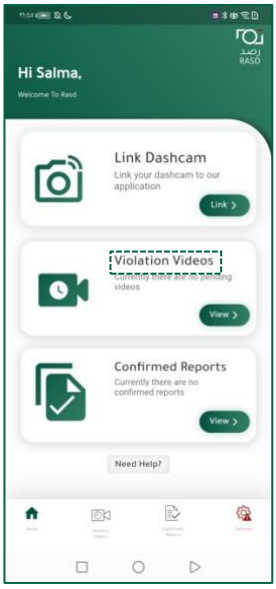


Figure 73 Changing Pending videos label

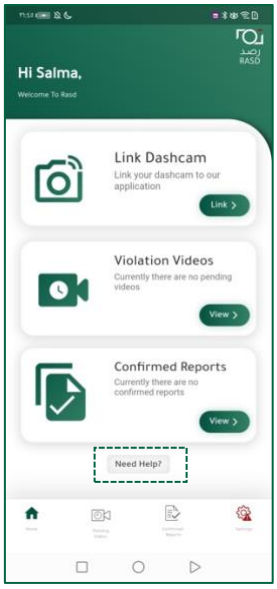


Figure 74 Add help button

Likewise, In the “link Dashcam” page, many drivers initially found it confusing to retrieve the dashcam information, as they had trouble locating the tooltip button, which was positioned underneath the input field and outlined in green. To address this issue, we relocated the tooltip to the top of the input field and filled in the icon with green, making it more noticeable and easier to locate Figure 76.

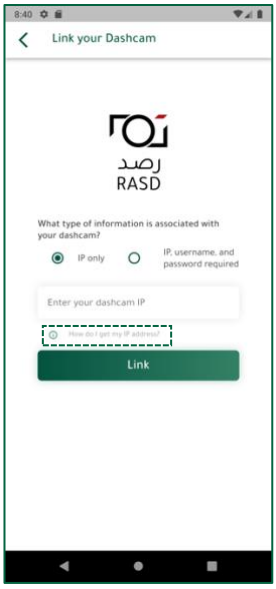


Figure 75 Tool Tip before

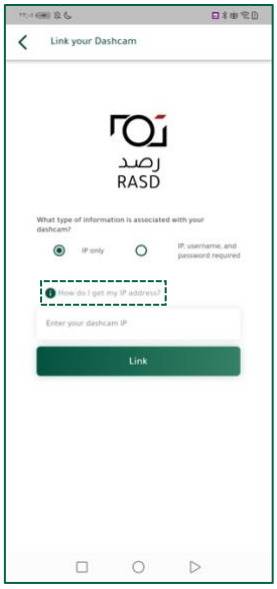


Figure 76 Tool Tip After Modification

Furthermore, in “Violation video” page, the majority of the drivers provided feedback on the video's speed, with some requesting to increase the speed and others preferring to slow it down for a more detailed observation of any violations. As a solution, we presented them with a range of speed options to adjust the video's duration according to their preferences Figure 78. We also received another feedback from two drivers concerning the “Show Violation Video” page, where in they expressed a desire to zoom in on the video to obtain a clearer view of the license plate or any other information. As a result, we incorporated this feedback into the application by adding an “expand” button that allows users to closely examine the violation video and retrieve important details from it Figure 78.

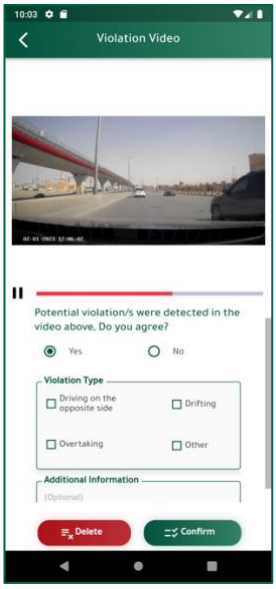


Figure 77 Violation video page before

Figure 78 Violation video page after modification

Additionally, we received feedbacks regarding the lists that display the pending violation videos and the confirmed reports. Since we display the pending violation videos and confirmed reports in numbered lists, as the number of items in each list grows over time, users require a way to differentiate between them. To address this, we concluded that the most effective approach would be to display the date of each violation alongside its corresponding pending violation video or confirmed violation report in the list Figure 80.

Figure 79 Reports list before

Figure 80 Reports list after modification

Finally, Regarding the “My Account” page, the majority of drivers expressed a preference for prefilled fields for their information rather than hint text. We took their feedback into consideration and incorporated it into the interface, as illustrated in the Figure 82.

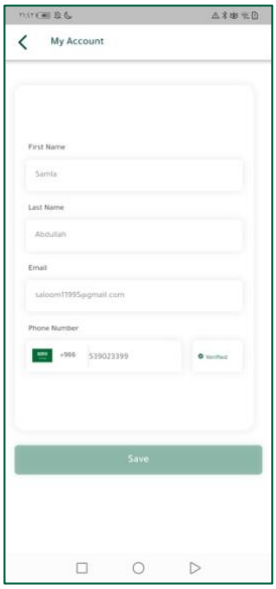


Figure 81 "My account" page before

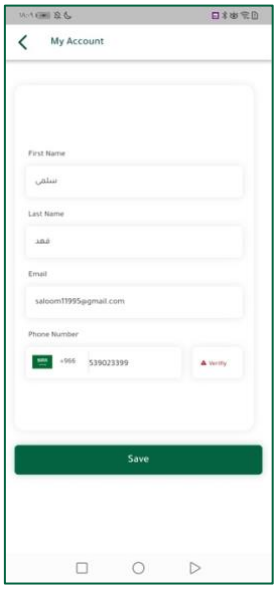


Figure 82 "My account" page After Modification

In the “Phone verification” page, some drivers provided feedback regarding the “Resend OTP” button, suggesting that it should be disabled with any visible timer or indicator for when they can click it again. In response to this, we incorporated their feedback into the application by displaying a countdown timer alongside the resend button Figure 84.

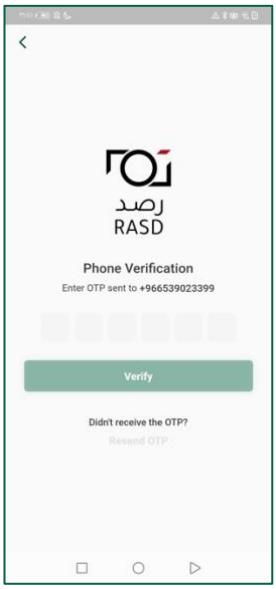


Figure 83 "Phone verification" page before

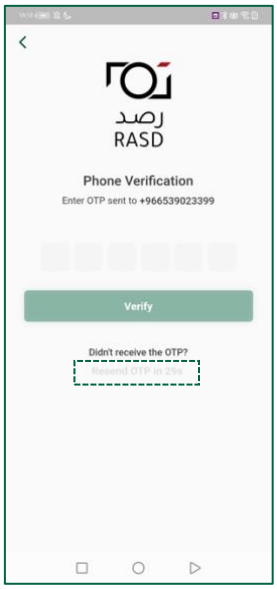


Figure 84 "Phone verification" page after modification

In conclusion, our users' satisfaction is our top priority, which is why we carefully incorporated all of their feedback and comments into the interface. This has resulted in significant improvements in the application's usability, which, in turn, ensures that the application meets the needs and expectations of its users.

6 Conclusions and Future Work

Reaching the conclusion means that our story is done. “RASD” started as a small idea that needed to grow, and in this document, we have represented our journey from transforming it from an idea to a product. In the introduction chapter, we clarified the idea, presented the problem that “RASD” is aiming to solve, produced a possible solution for it, and also defined the objectives. Following the introduction, we moved on to the Background chapter, where we described in detail the necessary domain knowledge and theoretical background required for the readers to understand “RASD”. In order to produce the application that aims to narrow the gap in detecting traffic violations and help both the community and authorities, we, as a team, reviewed the recent research papers and applications in the field and discussed how they relate to “RASD”. From there, we started the system design and development chapter, where we described the methodology, we were following to produce “RASD” and the features of “RASD” were transformed into a structure that is used to facilitate the implementation of “RASD” application as well as supporting the understanding of “RASD” components and how they depend on each other. Also, we included the road map for “RASD” to inform the readers of our working timeline. After analyzing our system, we started implementing “RASD” using the Flutter framework. Implementing “RASD” depends on several phases. At first, we started implementing the frontend with all the product features, then we implemented the backend and configured the desired machine learning model to apply it to “RASD” application. The backend development was deeply explained in the experimental results chapter, where we defined all the approaches and compared them to come up with the best possible solution. Then we started testing “RASD” as a whole product to ensure that it is functioning as desired. This is how we bring this project closure, and at this point, we believe that “RASD” makes a significant contribution to our country.

6.1 Local impact

Nowadays, drivers use Twitter to post videos from their dashcams to show dangerous road violations and accidents that happen daily. The application “RASD” aims to automate this process of using crowdsourcing to efficiently collaborate in monitoring road safety and detect road violations in a faster way. Thus, the local impact of “RASD” is potentially increasing the quality of life and enhancing public health. Reducing road violations will lead to several benefits, such as it reduces traffic congestion that will directly impact the life quality for people and reducing road injuries and deaths

6.2 Global impact

We believe that “RASD” has an impact on the worldwide economy by utilizing the advancement of artificial intelligence and computer vision to increase road safety. The automatic detection in the smart dashcam will decrease the number of drivers that violate the road rules, and therefore, the number of road accidents will decrease accordingly. The potential costs of road traffic accidents to the worldwide economy are enormous. In fact, the consequences of traffic accidents can range from injuries and deaths to life disruption and delays to public transport. The long-term of accident and road congestion consequences can lead to economic burdens and decrease the quality of life. Therefore, we see that “RASD” has a significant global impact by indirectly reducing the road injuries, property damages, deaths, road congestion and delay.

6.3 Problems and challenges encountered during the software development

The final chapter of "RASD" had some rough patches; we encountered many difficulties from the start, as we changed the scope after we started collecting the related data for our old scope and noticed the lack of tailgating violation videos. Due to that, we had to expand our scope and repeat the data collection process by covering traffic violations in general such as: drifting, driving in the opposite direction and overtaking. Also, as beginners in the Flutter framework, we struggled at the beginning of functions implementation and data connection, for instance: obtaining stream from the dashcam; upload the stream to the storage; verifying phone number; generating the report as a Pdf file and customizing it; edit the report; displaying the video; validating input text fields; and implementing "RASD" in both Arabic and English.

As novices in the field of computer vision with no prior knowledge or experience. We faced challenges in identifying the most suitable model for our needs and learning how to implement and customize it. We had to invest a significant amount of time and efforts in researching and experimenting with different models and parameters to find the best fit for our application. This involved learning about various techniques, including deep learning and computer vision, and learning how to use different tools and frameworks like TensorFlow, keras and OpenCV.

Also, one of the major challenges we faced is that the machine learning models often require a significant amount of memory and processing power to run, which can make them slow or even crash on our machines.

In the end, creating a "novel solution" as software that seamlessly integrates various components, including Python, Firebase, storage, and dashcam, within a tight timeframe and with no prior knowledge was a daunting challenge. The project required us to quickly learn and adjust to new technologies, tools, and languages, fix issues that arose during "RASD" development process. Despite the stress and difficulties, we encountered, we persevered and ultimately succeeded in delivering a functional and innovative solution that we are proud of.

6.4 Limitations of the system.

For now, “RASD” detects traffic violations only without defining the type of the violation, and the machine learning model offered by “RASD” does not extract any additional information related to the violated vehicle, such as the plate number, vehicle brand, or color. Also, there are drivers in Saudi Arabia who are not familiar with the supported languages offered by “RASD”. Moreover, to achieve the most accurate prediction possible with the violation detection model provided by “RASD”, it is crucial to ensure an optimal environment. This entails having a high-quality dashcam installed at the ideal position on the windscreen of the car to ensure an unobstructed view. Also, currently “RASD” support only 70mai M300 as a dashcam and Ezviz.

6.5 The main contribution of the project

This project has many contributions to the fields of technology and research. First, it provides a public dataset for road violations in Saudi Arabia that contains over 340 labelled videos of positive violations scenes and 340 as a normal scene. This dataset is labelled manually by human annotators with a reliable agreement of 76%. The dataset is public and can be used for further research studies and technology-based enhancements.

The second contribution is that when we compare the operation and maintenance costs of the standard CCTVs to the driver’s dashcam, it will show that these costs are less for the dashcam but sustain the process as long as there are drivers using dashcams in their vehicles.

6.6 Future work.

In future work, there are some functionalities that we might implement in the future, for instance: provide the drivers with a notification functionality that can be enabled or disabled based on their preference, which will notify them of the occurrence of a pending violation video, develop an AI model that automatically writes a description of the violation to be included in the violation report, support more dash cam types and support rear window screen, extract the plate number of the violated vehicle, and enhance the machine learning model by adding a tracking object for the vehicles and collaborate with the authorities.

7 Acknowledgements

We express our gratitude to Allah for helping us overcome any obstacles and challenges that we encountered during the project's completion. We extend our appreciation to our supervisor Dr.Luluh Aldhubayi for her invaluable assistance, unwavering encouragement, and invaluable insights have been critical to the success of “RASD”. Additionally, we would like to thank Modi Aljedaie for sharing her dashcam videos with “RASD”. Furthermore, we gratefully acknowledge the annotators for consuming their time and effort to accurately label “RASD” dataset. We would like to thank Note Almurshed for sharing her exceptional talent and expertise by contributing in designing the visual identity of “RASD”.

In conclusion, we acknowledge the indispensable role that our families and friends played in the successful completion of this project. Their endless support, encouragement, prayers and belief in our abilities have been a constant source of motivation and inspiration, and we are forever grateful for them.

8 References

[1] “Road Traffic Injuries and Deaths” stats.gov.sa https://www.moh.gov.sa/en/Ministry/Statistics/Pages/Traffic-accidents.aspx [accessed Sep. 02, 2022].

[2] “Saudi Arabia registers 40% rise to 54.4 mln traffic violations in 2019” argam.com https://www.argaam.com/en/article/articledetail/id/1428084 [accessed Jan. 31,2023].

[3] “What is a traffic violation?” findlaw.com https://www.findlaw.com/hirealawyer/choosing-the-right-lawyer/traffic-violations.html [accessed Sep. 18, 2022].

[4] “Tailgating” wikipedia.org https://en.wikipedia.org/wiki/Tailgating [accessed Sep. 18, 2022].

[5] “Wrong-way driving” wikipedia.org https://en.wikipedia.org/wiki/Wrong-way_driving#:~:text=Wrong-way%20driving%20(WWD),against%20the%20direction%20of%20traffic [accessed Oct. 31, 2022].

[6] “Overtaking” wikipedia.org https://en.wikipedia.org/wiki/Overtaking [accessed Oct. 31, 2022].

[7] “The Science Behind Drifting” burtbrothers https://burtbrothers.com/blog/the-science-behind-drifting/ [accessed Oct. 31, 2022].

[8] “Dashcam” wikipedia.org https://en.m.wikipedia.org/wiki/Dashcam [accessed Sep. 02, 2022].

[9] “About” opencv.org https://opencv.org/about/ [accessed Sep. 18, 2022].

[10] “Deep Neural Network” Science Direct https://www.sciencedirect.com/topics/engineering/deep-neural-network#:~:text=Deep%20neural%20networks%20are%20a%20powerful%20category%20of,wide%20range%20of%20applications%20in%20the%20contemporary%20years.[accessed Nov. 6, 2022].

[11] “Home” opencv.org https://opencv.org [accessed Sep. 18, 2022].

[12] "What is a machine learning model?" microsoft.com https://docs.microsoft.com/en-us/windows/ai/windows-ml/what-is-a-machine-learning-model [accessed Sep. 18, 2022].

[13] "mobilenetv2" mathworks.com https://www.mathworks.com/help/deeplearning/ref/mobilenetv2.html [accessed Jan. 31,2023].

[14] "What is YOLOv5" robflow.com https://blog.roboflow.com/yolov5-improvements-and-evaluation/ [accessed Jan. 31,2023].

[15] "RTSP: The Real-Time Streaming Protocol Explained (Update)" wowza.com https://www.wowza.com/blog/rtsp-the-real-time-streaming-protocol-explained#rtsp [accessed Jan. 31,2023]

[16] Irfanullah, & Hussain, Tariq & Iqbal, Arshad & Yang, Bailin & Hussain, Altaf. (2022). *Real time violence detection in surveillance videos using Convolutional Neural Networks*. Multimedia Tools and Applications. 81. 10.1007/s11042-022-13169-4.

[17] Y. Yao, M. Xu, Y. Wang, D. J. Crandall and E. M. Atkins, "Unsupervised Traffic Accident Detection in First-Person Videos," 2019 *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Macau, China, 2019, pp. 273-280, doi: 10.1109/IROS40897.2019.8967556.

[18] P. Jin, L. Mou, G. -S. Xia and X. X. Zhu, "Anomaly Detection in Aerial Videos With Transformers," in *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1-13, 2022, Art no. 5628213, doi: 10.1109/TGRS.2022.3198130.

[19] Xiaoling Wang, Li-Min Meng, Biaobiao Zhang, Junjie Lu and K. . -L. Du, "A video-based traffic violation detection system," *Proceedings 2013 International Conference on Mechatronic Sciences*, Electric Engineering and Computer (MEC), 2013, pp. 1191-1194, doi: 10.1109/MEC.2013.6885246.

[20] D. Chand, S. Gupta and I. Kavati, "Computer Vision based Accident Detection for Autonomous Vehicles," 2020 *IEEE 17th India Council International Conference* (INDICON), 2020, pp. 1-6, doi: 10.1109/INDICON49873.2020.9342226.

[21] "كلنا امن" App Store, 24 Mar. 2021, https://apps.apple.com/sa/app/%D9%83%D9%84%D9%86%D8%A7-%D8%A3%D9%85%D9%86/id1085635249.

[22] "70mai (Worldwide)." App Store, 17 Aug. 2022, https://apps.apple.com/sa/app/70mai-worldwide/id1345384242.

[23] "Dubai Police - شرطة دبي." App Store, 22 Aug. 2022, https://apps.apple.com/sa/app/dubai-police-%D8%B4%D8%B1%D8%B7%D8%A9-%D8%AF%D8%A8%D9%8A/id384374316.

[24] "BlackVue." App Store, 9 Aug. 2022, https://apps.apple.com/sa/app/blackvue/id1049209637.

[25] "Hello World" github.com https://docs.github.com/en/get-started/quickstart/hello-world [accessed Feb. 03,2023]

[26] X. Liu, G. Tang and W. Zou, "Improvement of Detection Accuracy of Aircraft in Remote Sensing Images Based on YOLOV5 Model," 2021 *IEEE International Geoscience and Remote Sensing Symposium IGARSS*, Brussels, Belgium, 2021, pp. 4775-4778, doi: 10.1109/IGARSS47720.2021.9554234.

[27] K. Klubsuwan, W. Koodtalang and S. Mungsing, "Traffic Violation Detection Using Multiple Trajectories Evaluation of Vehicles," 2013 4th International Conference on Intelligent Systems, Modelling and Simulation, 2013, pp. 220-224, doi: 10.1109/ISMS.2013.143.

[28] "Multi-Class Metrics Made Simple, PartIII: the Kappa Score(aka Cohen's Kappa Coefficient)" towardsdatascience.com https://towardsdatascience.com/multi-class-metrics-made-simple-the-kappa-score-aka-cohens-kappa-coefficient-bdea137af09c [accessed Feb. 05,2023]

[29] "Fleiss' kappa in SPSS Statistics" https://statistics.laerd.com/spss-tutorials/fleiss-kappa-in-spss-statistics.php [accessed Feb. 05,2023]

[30] "Firebase" Wikipedia.org https://en.wikipedia.org/wiki/Firebase [accessed Nov. 1, 2022].

[31] "Cloud Firestore" Firebase https://firebase.google.com/products/firestore [accessed Nov. 2, 2022]

[32] "Google Firebase — Introduction and using Authentication service" Factory.hr https://factoryhr.medium.com/google-firebase-introduction-and-using-authentication-service-7321fcd1b5aa [accessed Nov. 1, 2022].

[33] "Get started with Cloud Storage on Web" firebase.google.com https://firebase.google.com/docs/storage/web/start [accessed Feb. 06,2023]

[34] "What is Heroku and what is it Used For" mentormate.com https://mentormate.com/blog/what-is-heroku-used-for-cloud-development/ [accessed Feb. 06,2023]

[35] M. Mazhar Rathore, Anand Paul, Seungmin Rho, Murad Khan, S. Vimal, Syed Attique Shah,Smart traffic control: Identifying driving-violations using fog devices with vehicular cameras in smart cities,Sustainable Cities and Society,Volume 71,2021,102986,ISSN 2210-6707.

[36] "Detection-of-Traffic-Anomaly" github.com https://github.com/kpandey008/Abnormal-Event-Detection [accessed Dec. 10,2022]

[37] "tad-IROS2019" github.com https://github.com/kpandey008/Abnormal-Event-Detection [accessed Dec. 10,2022]

[38] "HumanBehaviorBKU" github.com https://github.com/kpandey008/Abnormal-Event-Detection [accessed Dec. 10,2022]

[39] "Abnormal-behavior-Detection" github.com https://github.com/kpandey008/Abnormal-Event-Detection [accessed Dec. 10,2022]

[40] "highway-lane-violation-detector" github.com https://github.com/kpandey008/Abnormal-Event-Detection [accessed Dec. 15,2022]

[41] "Abnormal-Event-Detection" github.com https://github.com/kpandey008/Abnormal-Event-Detection [accessed Dec. 19,2022]

[42] "printing - Dart API docs" pub.dev https://pub.dev/documentation/printing/latest/ [accessed Oct. 30, 2022].

[43] "PdfPreview class - printing library - Dart API" pub.dev https://pub.dev/documentation/printing/latest/printing/PdfPreview-class.html [accessed Oct. 30, 2022].

[44] "PdfShareAction class - printing library - Dart API" pub.dev https://pub.dev/documentation/printing/latest/printing/PdfShareAction-class.html [accessed Oct. 30, 2022]

[45] "Is there a way to set a password strength for Firebase?" stackoverflow https://stackoverflow.com/questions/49183858/is-there-a-way-to-set-a-password-strength-for-firebase [accessed Oct. 26, 2022]

[46] "Non-functional Requirements: Examples, Types, How to Approach" altexsoft.com https://www.altexsoft.com/blog/non-functional-requirements/ [accessed Feb. 08, 2023]

[47] "imshow doesn't need convert from BGR to RGB" stackoverflow https://stackoverflow.com/questions/50963283/imshow-doesnt-need-convert-from-bgr-to-rgb [accessed Feb. 10, 2023]

[48] "How to load grayscale image dataset to Mobile net model" stackoverflow https://stackoverflow.com/questions/57357049/how-to-load-grayscale-image-dataset-to-mobile-net-model [accessed 10, 2023]

[49] "Video Classification with a CNN-RNN Architecture" keras.io https://keras.io/examples/vision/video_classification/ [accessed 11, 2023]

[50] "The Science Behind Drifting" burtbrothers https://burtbrothers.com/blog/the-science-behind-drifting/ [accessed Oct. 31, 2022].

[51] "Wrong-way driving" wikipedia.org https://en.wikipedia.org/wiki/Wrong-way_driving#:~:text=Wrong-way%20driving%20(WWD),against%20the%20direction%20of%20traffic [accessed Oct. 31, 2022].

[52] "Violence-Alert-System" github.com https://github.com/chandran-jr/Violence-Alert-System [accessed Dec.27,2022]

[53] "Inter-rater Reliability IRR: Definition, Calculation" statisticshowto.com https://www.statisticshowto.com/inter-rater-reliability/ [accessed Feb.20, 2023]

[54] "Traffic Accident by region" stats.gov.sa https://www.stats.gov.sa/en/6370 [accessed Sep. 02, 2022].

9 Appendix

9.1 Appendix A: Interviews

Interview Questions:

Questions 1:

What is your name, age, and where do you live?

Questions 2:

Lately, the use of dashcams has been increased as a result of drivers' encountering numerous events. Have you been using it, and why?

Questions 3:

What are the main factors that affect the safety of transportation?

Questions 4:

In your opinion, what are the most common traffic violations that frustrate drivers while driving?

Questions 5:

Can you evaluate how well CCTV cameras and surveillance systems work in detecting traffic violations?

Questions 6:

How would you respond if you were driving on a highway and someone committed a traffic violation against you?

Questions 7:

Can you describe the challenges you have when filing a traffic violation report?

Questions 8:

What difference would it make to have a smart dashcam linked to a mobile application that detects the traffic violation and makes a readymade report that includes the information of the violated vehicle?

Interviews' Transcriptions:

Interview 1 outline		
Interviewee: Note Abdullah	Interviewer: Dhai AlMurshed	
Location/Medium: Riyadh In person	Appointment Date: 20-09-2022	
	Start Time: 17:03	End Time: 17:10
Objectives: Collect general facts, useful and necessary information about traffic violations and traffic violation reporting process.	Reminders: None.	
Agenda: Introduction Background about the application Overview of the interview Topics to be covered Permission to record Summary of Major points Closing	Approximate Time: 30 seconds 1 minutes 30 seconds 30 seconds 30 seconds 2 minutes 1 minutes	
General Observation: The Interviewee was very excited for such a product that can detect traffic violations and produce a readymade report.		

Unresolved Issues, Topics Not Covered: None.

Interview 1 outline	
Questions:	Notes:
Questions 1: What is your name, age, and where do you live?	Answer: My name is Note, 32 years old, and I live in Riyadh
	Observations: None.
Questions 2: Lately, the use of dashcams has been increased as a result of drivers' encountering numerous events. Have you been using it, and why?	Answer: No
	Observations: She seems like did not think to have one.
Questions 3: What are the main factors that affect the safety of transportation?	Answer: A binding traffic rules, reckless driving, unexperienced drivers, and awareness of traffic rules
	Observations: She took a while to summarize the factors that affect the safety of transportation
Questions 4: In your opinion, what are the most common traffic violations that frustrate drivers while driving?	Answer: Cutting traffic like when someone overtaking from the right-side, and tailgating
	Observations: She gives a quick response.

Interview 1 outline	
Questions 5: Can you evaluate how well CCTV cameras and surveillance systems work in detecting traffic violations?	Answer: It detects traffic violations only if there is a CCTV, which means it depends on the area coverage by the CCTV, when a violation happened in an area with no coverage then it won't detect it , also the quality of the CCTV angle in identifying the picture that has been captured.
	Observations: None.
Questions 6: How would you respond if you were driving on a highway and someone committed a traffic violation against you?	Answer: Ignore, if it is did not result an accident usually I ignore
	Observations: She seems like she is very calm while driving as she did not take any action against the violated vehicle when occurs
Questions 7: Can you describe the challenges you have when filing a traffic violation report?	Answer: I have never filled a traffic violation report only an accident report
	Observations: She is not familiar of reporting any traffic violations

Interview 1 outline	
Questions 8: What difference would it make to have a smart dashcam linked to a mobile application that detects the traffic violation and makes a readymade report that includes the information of the violated vehicle?	**Answer:** It makes it very easy to record the violation and since the recording of the violation made easy it will tempt me more to report the violation
	Observations: She was a strong supporter of this idea.

Table 51 Interview 1

<table>
<tr><th colspan="3">Interview 2 outline</th></tr>
<tr><td>Interviewee:
Sara Fahad</td><td colspan="2">Interviewer:
Manar Mutaieb</td></tr>
<tr><td rowspan="2">Location/Medium:
Via WhatsApp</td><td colspan="2">Appointment Date:
13-09-2022</td></tr>
<tr><td>Start Time:
18:43</td><td>End Time:
18:53</td></tr>
<tr><td>Objectives:
Collect general facts, useful and necessary information about traffic violations and traffic violation reporting process.</td><td colspan="2">Reminders:
None.</td></tr>
<tr><td>Agenda:
Introduction
Background about the application
Overview of the interview
Topics to be covered
Permission to record
Summary of Major points
Closing</td><td colspan="2">Approximate Time:
10 seconds
1 minutes
20 seconds
20 seconds
10 seconds
7 minutes
30 seconds</td></tr>
<tr><td colspan="3">General Observation:
The interviewee has three years of driving experience.</td></tr>
<tr><td colspan="3">Unresolved Issues, Topics Not Covered:
None.</td></tr>
</table>

Interview 2 outline	
Questions:	**Notes:**
Questions 1: What is your name, age, and where do you live?	**Answer:** Sara Fahad. 25 years. I live in Riyadh.
	Observations: None.
Questions 2: Lately, the use of dashcams has been increased as a result of drivers' encountering numerous events. Have you been using it, and why?	**Answer:** I'm not using it because I'm not that interested in buying stuff for my car (not a priority). Also, there are multiple types of dashcams, and it is constantly updated.
	Observations: None.
Questions 3: What are the main factors that affect the safety of transportation?	**Answer:** 1- Traffic jam. 2- Driver’s behavior. 3- The presence of the trucks. 4- Road excavations.
	Observations: She stresses that the driver's behavior is the most factor that affects the safety of transportation.

Interview 2 outline	
Questions 4: In your opinion, what are the most common traffic violations that frustrate drivers while driving?	**Answer:** 1- Speed. 2- Not keeping a safe distance. 3- Not using the turn signals.
	Observations: She means by "turn signals" the signals that indicate that the car is going to turn right or left.
Questions 5: Can you evaluate how well CCTV cameras and surveillance systems work in detecting traffic violations?	**Answer:** Good to a certain amount, it is not detecting the safe distance, turn signals, although they said it will. It is not detecting other violation as it should.
	Observations: None.
Questions 6: How would you respond if you were driving on a highway, and someone committed a traffic violation against you?	Answer: I will call NAJM.
	Observations: She misunderstood the question at first, and after further explanation, she said that she usually ignores it.

Interview 2 outline	
Questions 7: Can you describe the challenges you have when filing a traffic violation report?	**Answer:** 1- Writing the exact scenario. 2- Not having a picture or a video of the violation. 3- Remembering the exact location of the violation if I am reporting from home.
Questions 8: What difference would it make to have a smart dashcam linked to a mobile application that detects the traffic violation and makes a readymade report that includes the information of the violated vehicle?	**Observations:** She files reports through 'كلنا امن'.
	Answer: It will be very helpful. It will save time, and effort causing less traffic. Also, the report will be more efficient.
	Observations: She seems enthusiastic about the readymade report.

Table 52 Interview 2

Interview 3 outline		
Interviewee: Fahad Abdullah	**Interviewer:** Manar Mutaieb	
Location/Medium: Riyadh In person	**Appointment Date:** 14-09-2022	
	Start Time: 18:38	**End Time:** 18:52
Objectives: Collect general facts, useful and necessary information about traffic violations and traffic violation reporting process.	**Reminders:** None.	
Agenda: Introduction Background about the application Overview of the interview Topics to be covered Permission to record Summary of Major points Closing	**Approximate Time:** 10 seconds 2 minutes 20 seconds 20 seconds 10 seconds 10 minutes 30 seconds	
General Observation: The interviewee has thirty-eight years of driving experience.		
Unresolved Issues, Topics Not Covered: None.		

Interview 3 outline	
Questions:	**Notes:**
Questions 1: What is your name, age, and where do you live?	**Answer:** Fahad Abdullah I am 54 years old and I live in Riyadh.
	Observations: None.
Questions 2: Lately, the use of dashcams has been increased as a result of drivers' encountering numerous events. Have you been using it, and why?	**Answer:** No, I don't know how reliable the dashcam evidence is.
	Observations: He thinks that the authorities will not accept the dashcam evidence.
Questions 3: What are the main factors that affect the safety of transportation?	**Answer:** - The lights on the street. - How smooth the roads are. - Having roads with lanes. - The presence of traffic officers. - Setting speed limits. - putting warning signs for speed bumps, turns, exits, and construction.
	Observations: None.

Interview 3 outline	
Questions 4: In your opinion, what are the most common traffic violations that frustrate drivers while driving?	**Answer:** - irregular overtaking. - When someone flashes you to forcibly clear the way for them. Also, someone once did not leave enough space behind me, so I had to move over to let him pass, and when he was in front of me, he suddenly slammed on his brakes, which made me crash into him.
	Observations: When he told the story, he appeared to be very angry and upset.
Questions 5: Can you evaluate how well CCTV cameras and surveillance systems work in detecting traffic violations?	**Answer:** They are excellent, but they need to be installed at every traffic light.
	Observations: He truly believes in the effectiveness of CCTV cameras and surveillance systems in detecting all violations.

Interview 3 outline	
Questions 6: How would you respond if you were driving on a highway, and someone committed a traffic violation against you?	**Answer:** memorize their vehicle plate number and call 999.
	Observations: None.
Questions 7: Can you describe the challenges you have when filing a traffic violation report?	**Answer:** When you suddenly encounter a road violation, you might not be in the right mental state to understand what's going on, and you might forget a lot of things.
	Observations: None.
Questions 8: What difference would it make to have a smart dashcam linked to a mobile application that detects the traffic violation and makes a readymade report that includes the information of the violated vehicle?	**Answer:** I think that a lot of people do not use dashcams, and I don't think it's completely authorized in our country. So, the authorities may not think that these reports are valid.
	Observations: He thinks that this application will detect any road violations that happen -not only the ones that directly involve him-, thus taking over the role of road officers.

Table 53 Interview 3

<table>
<tr><th colspan="3">Interview 4 outline</th></tr>
<tr><td>Interviewee:
Omar AlShehri</td><td colspan="2">Interviewer:
Ghadeer AlMaymoon</td></tr>
<tr><td rowspan="2">Location/Medium:
Via Zoom Video Communications</td><td colspan="2">Appointment Date:
20-09-2022</td></tr>
<tr><td>Start Time:
21:00</td><td>End Time:
21:30</td></tr>
<tr><td>Objectives:
Collect general facts, useful and necessary information about traffic violations and traffic violation reporting process.</td><td colspan="2">Reminders:
None.</td></tr>
<tr><td>Agenda:
Introduction
Background about the application
Overview of the interview
Topics to be covered
Permission to record
Interview Questions
Summary of Major points
Closing</td><td colspan="2">Approximate Time:
30 seconds
5 minutes
2 minutes
3 minutes
30 seconds
10 minutes
7 minutes
2 minutes</td></tr>
<tr><td colspan="3">General Observation:
The Interviewee has been driving for almost 7 years.</td></tr>
<tr><td colspan="3">Unresolved Issues, Topics Not Covered:
None.</td></tr>
</table>

Interview 4 outline	
Questions:	**Notes:**
Questions 1: What is your name, age, and where do you live?	**Answer:** Omar AlShehri. 24 years. I live in AlKhoubar.
	Observations: None.
Questions 2: Lately, the use of dashcams has been increased as a result of drivers' encountering numerous events. Have you been using it, and why?	**Answer:** No, I'm not using dashcams because I don't think that I need it and so far I haven't been in a situation where I thought I should've used it.
	Observations: None.
Questions 3: What are the main factors that affect the safety of transportation?	**Answer:** 1- At rush hours especially the traffic police cannot always cover each and every violation. 2- reckless drivers can cause serious harm to themselves and to other drivers in the road.
	Observations: The interviewee took a time to think of the main factors, he thinks there is more factors that affect safety transportation but these two reasons are what he could remember during the interview as he said.

Interview 4 outline	
Questions 4: In your opinion, what are the most common traffic violations that frustrate drivers while driving?	**Answer:** 1- tailgating. 2- overtalking. 3-crossing the left-side to go to the right-side or vice versa immediately with a high speed.
	Observations: None.
Questions 5: Can you evaluate how well CCTV cameras and surveillance systems work in detecting traffic violations?	**Answer:** It is not that reliable as some places in the road not covered and out of these cameras range but with the available technologies they are functioning well.
	Observations: The interviewee wants to have some examples on the mentioned systems and cameras then he answered fluently.
Questions 6: How would you respond if you were driving on a highway, and someone committed a traffic violation against you?	**Answer:** I will try to calm down and leave him alone.
	Observations: The interviewee had a moment to remember some related situations so he can remember his respond and then he answered.

Interview 4 outline	
Questions 7: Can you describe the challenges you have when filing a traffic violation report?	**Answer:** I did not even try to report a violation, actually I don't know the process of reporting a violation.
	Observations: The interviewee asked about the different ways of reporting these days which concludes that he have never report a violation.
Questions 8: What difference would it make to have a smart dashcam linked to a mobile application that detects the traffic violation and makes a readymade report that includes the information of the violated vehicle?	**Answer:** People's driving will be more careful, and they will obey to the previously violated rules. Even though I've never tried to report a violation, but I think the process of reporting violations will be much easier and highways will be safer.
	Observations: The interviewee was interested in the whole idea especially the violation detection model, he wanted to be informed whenever the application released.

Table 54 Interview 4

9.2 Appendix B: Questionnaire

Questionnaires questions:

Questionnaires	
Q1- How old are you? ○ 18-25 ○ 26-30 ○ 31-35 ○ 36 and above	Q2- Gender? ○ Female ○ Male

Questionnaires	
Q3- Select among the following options, what is/are the most traffic violation/s you have encountered?  ☐ Tailgating  ☐ Overtaking vehicles in curves and uphill(s)  ☐ Driving on the opposite direction  ☐ Non-stopping completely at the stop signal ☐ Other:	Q4- CCTVs cameras and surveillance system on the roads such as: Saher, are enough to detect each and every traffic violation? ◯ Yes ◯ No ◯ Maybe
Q5- Select among the following options, what make you drop the thought of reporting the traffic violation completely: ☐ Having multiple platforms ☐ Lack of violated vehicle's information ☐ Wasting time while reporting due to the process's complexity ☐ Other:	Q6- Select among the following options, when someone commits a traffic violation against you, you respond by: ☐ Recording the violated vehicle with your phone ☐ Trying to memorize the vehicle's information ☐ Write down the vehicle's information ☐ Ignore the violated vehicle ☐ Other:

Questionnaires	
Q7- Do you have a dashcam in your vehicle? ○ Yes ○ No ○ Planning to have one	Q8- Are you interested in having a smart dashcam linked to a mobile application that detects the traffic violation and makes a readymade report that includes the information of that violation? ○ Yes ○ No

Table 55 Questionnaires Questions

Questionnaires Results:

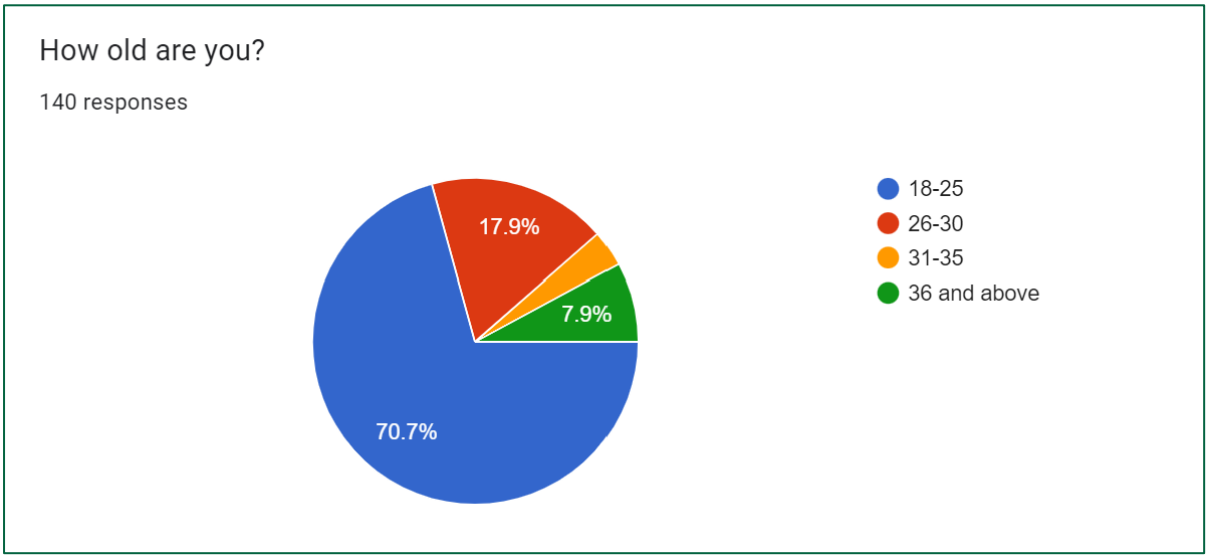


Figure 86 First Question Chart

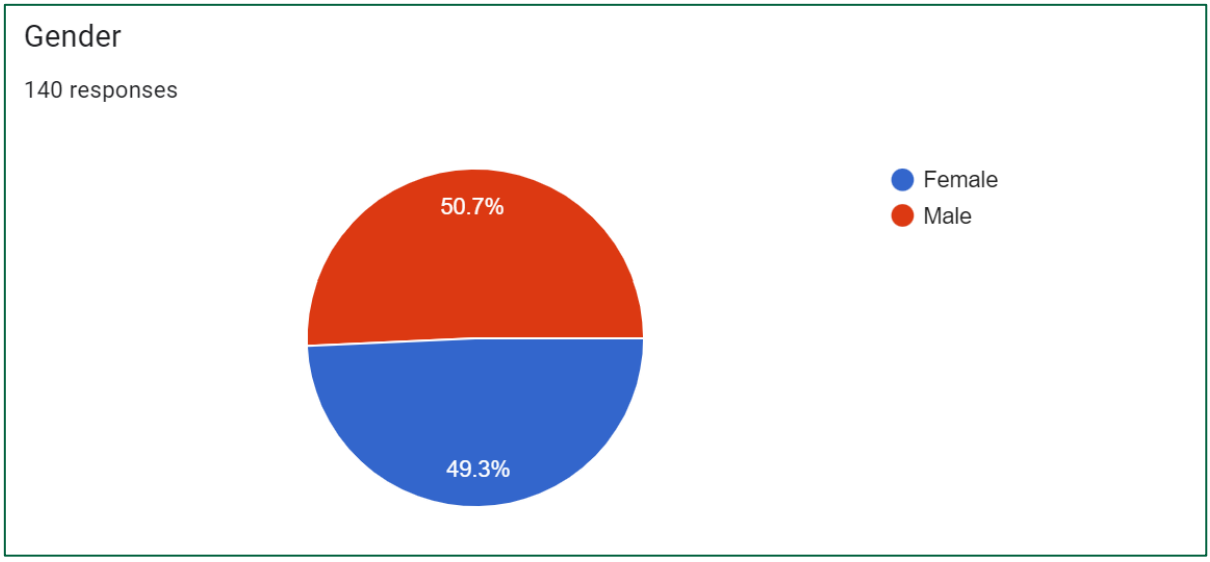


Figure 85 Second Question Chart

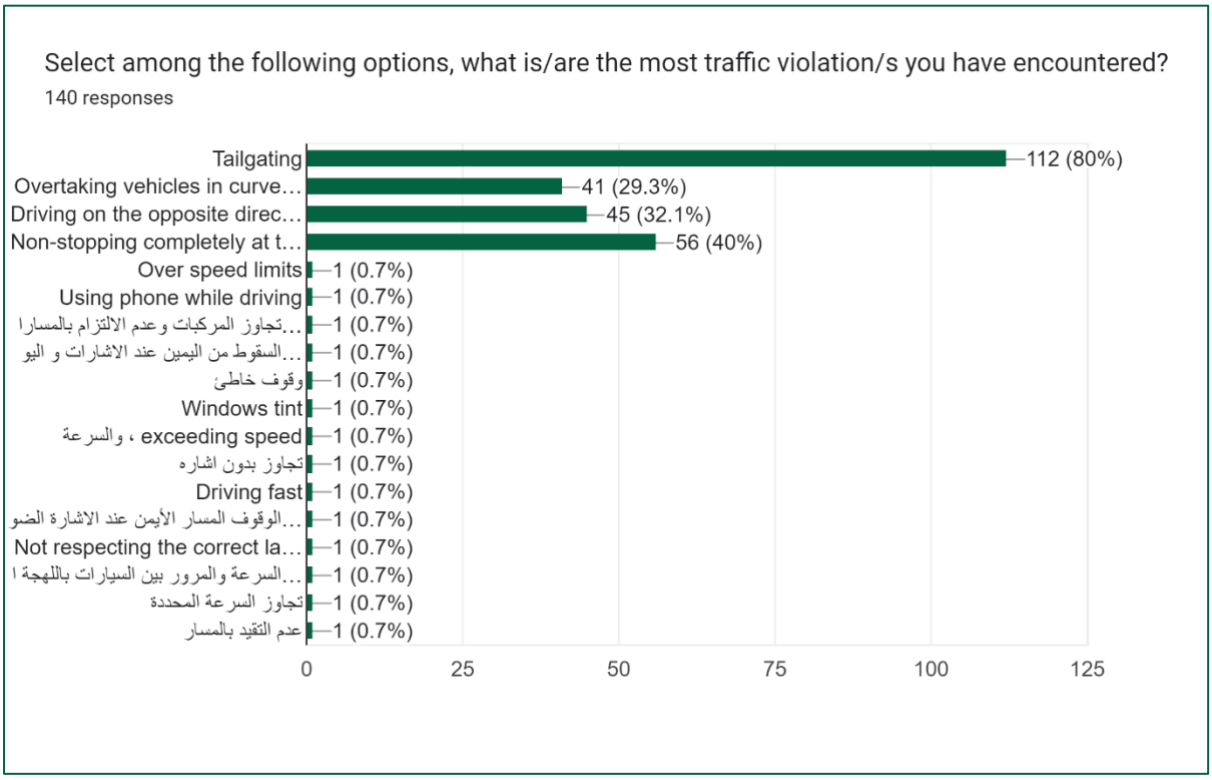


Figure 88 Third Question Chart

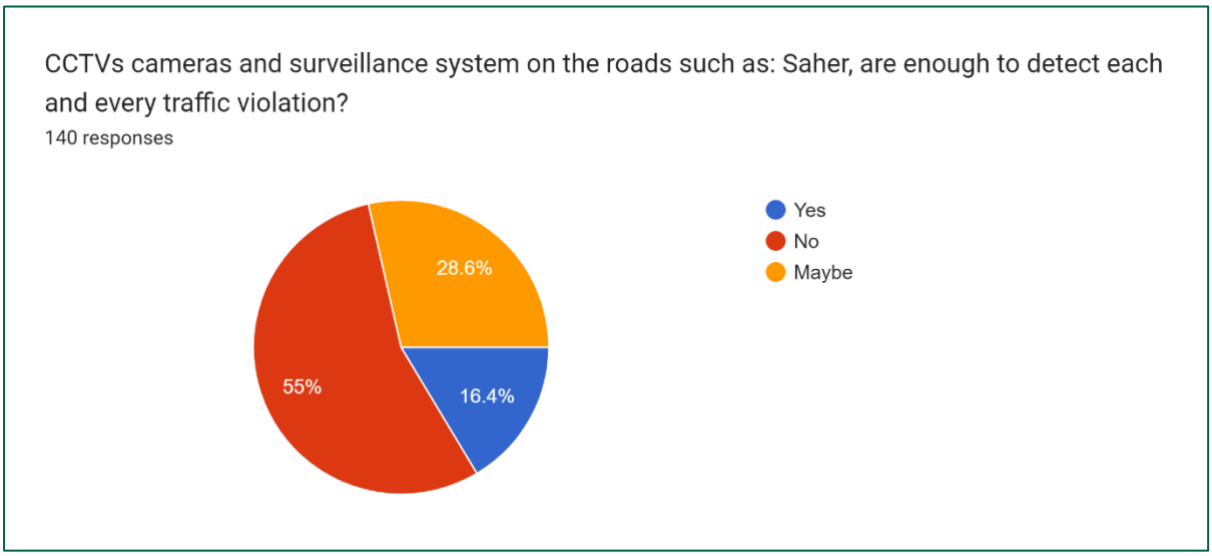


Figure 87 Fourth Question Chart

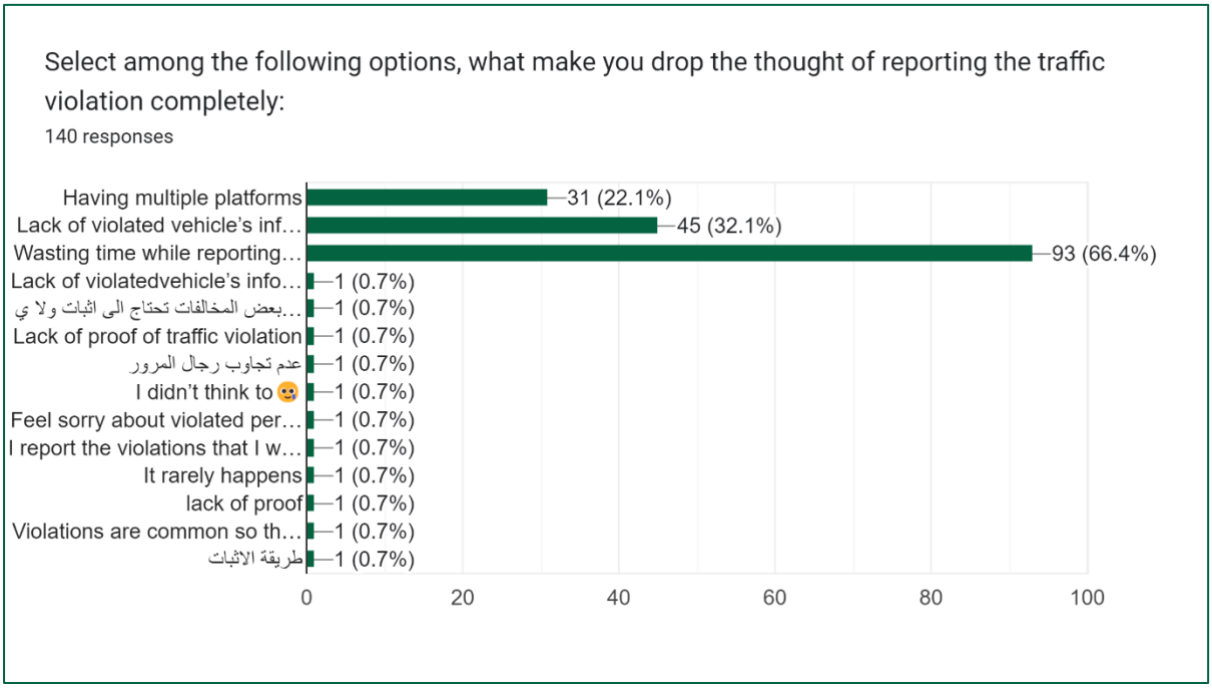


Figure 90 Fifth Question Chart

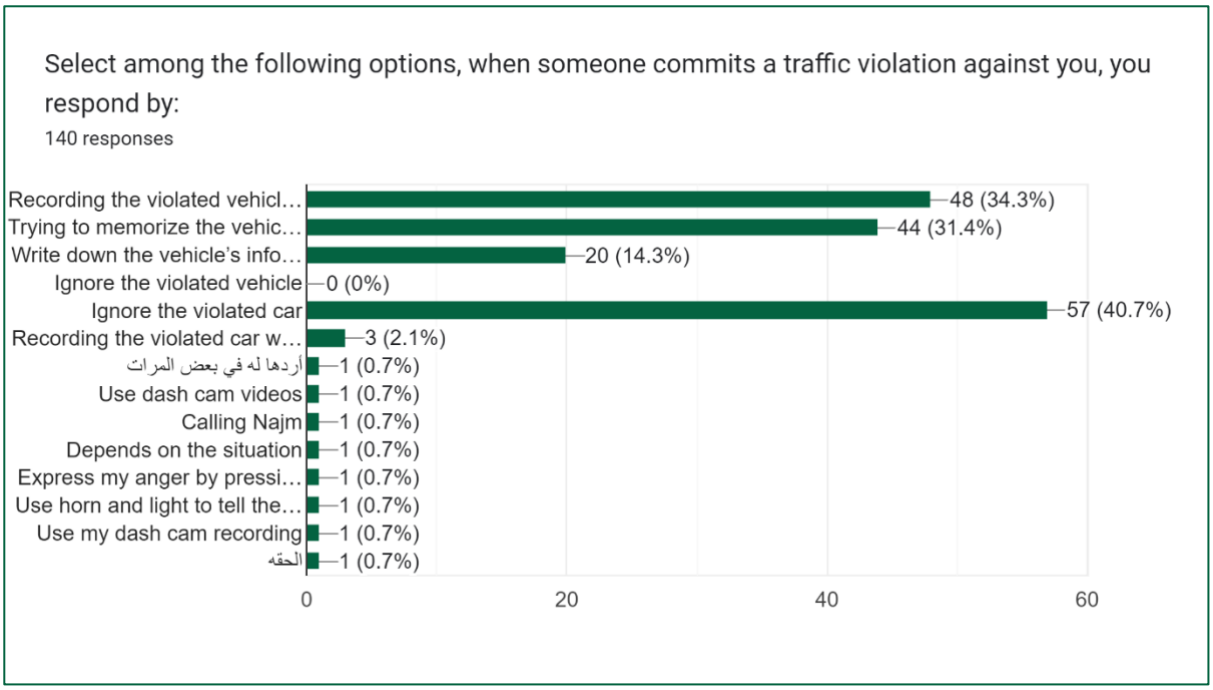


Figure 89 Sixth Question Chart

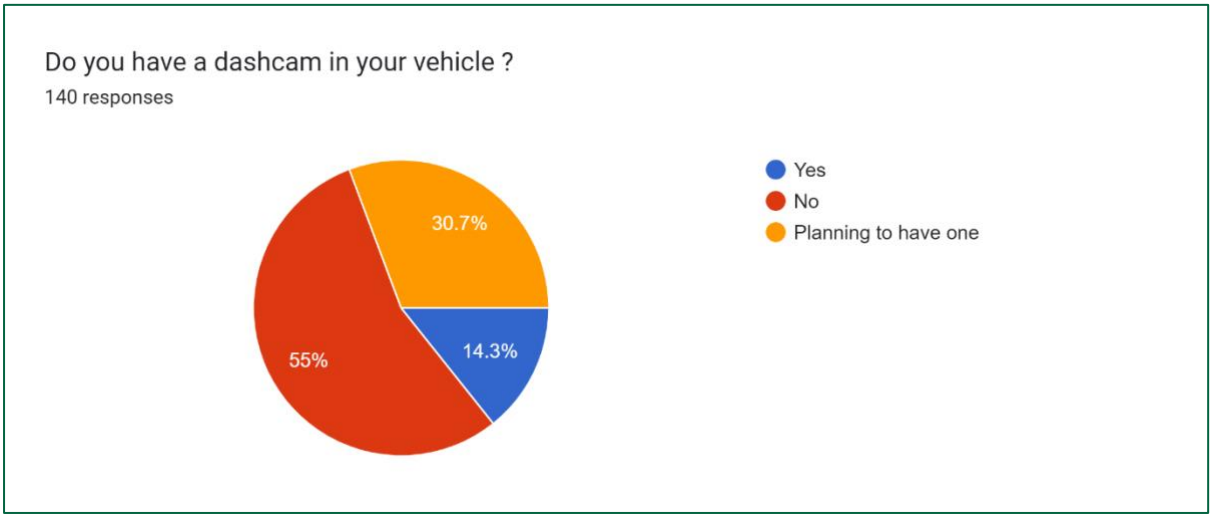


Figure 92 Seventh Question Chart

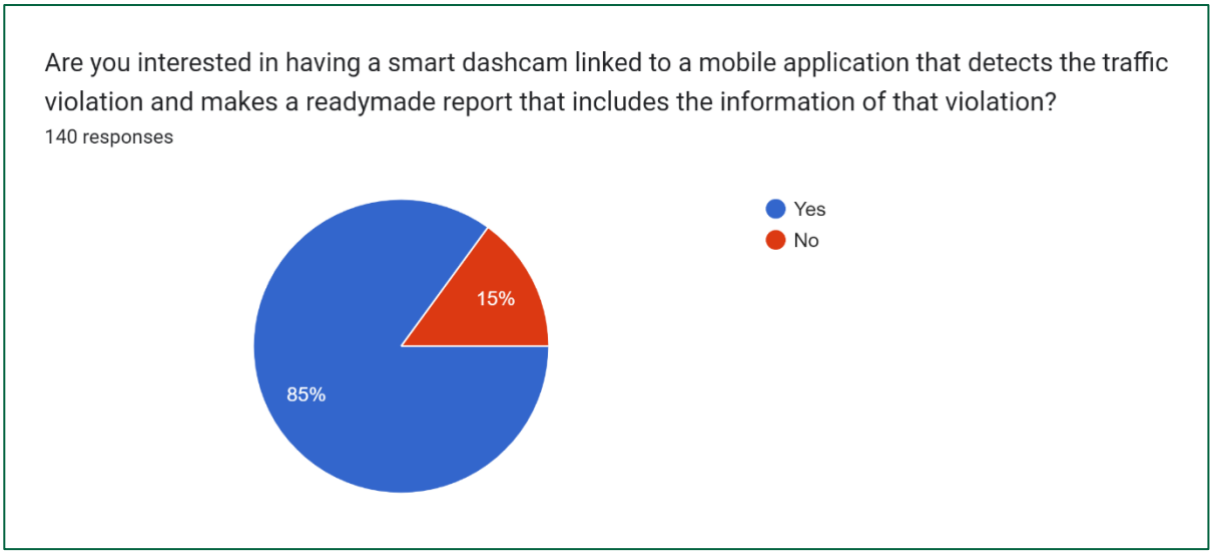
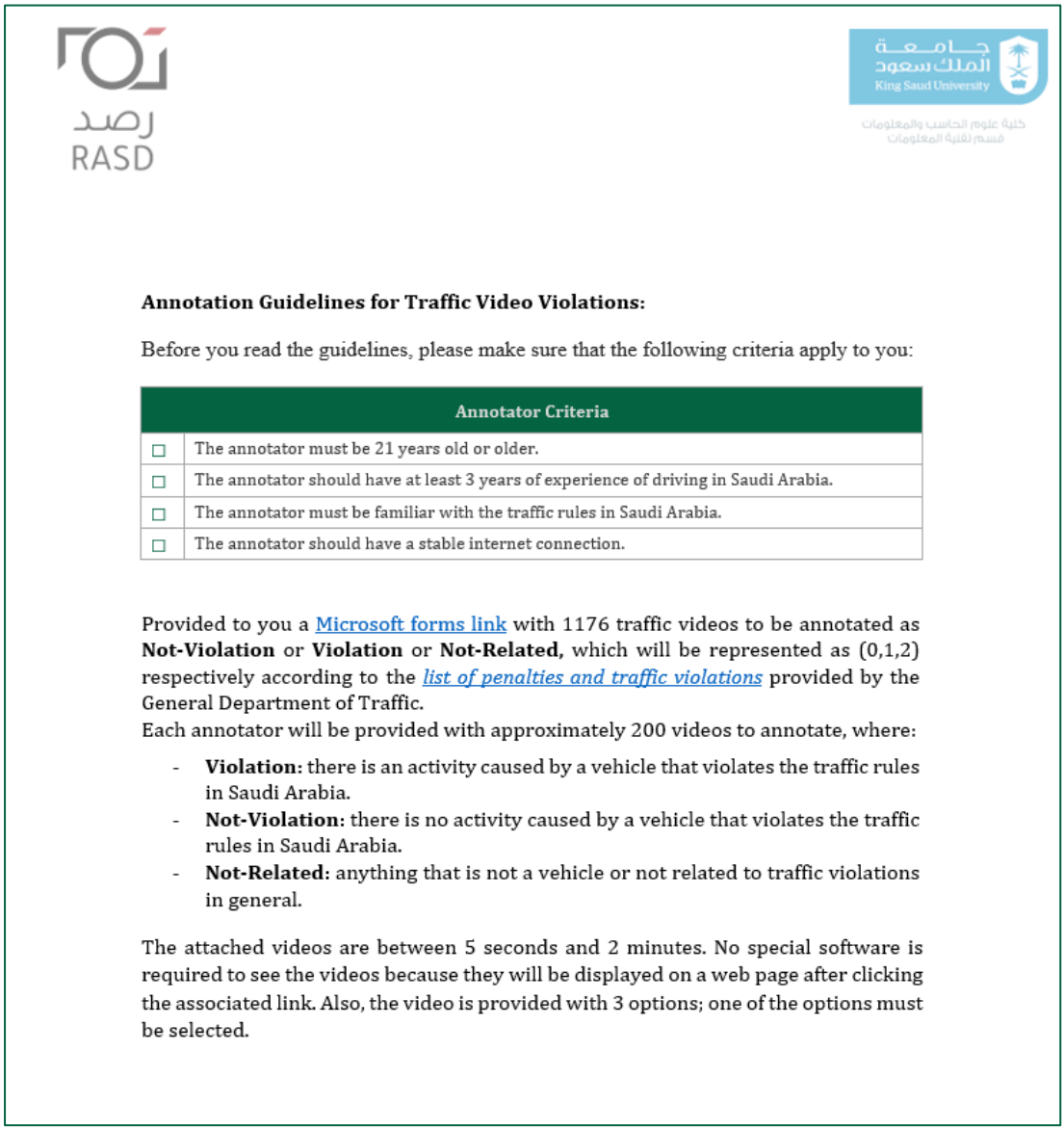


Figure 91 Eighth Question Chart

9.3 Appendix C: Annotation Criteria and Guidelines



Annotation Guidelines for Traffic Video Violations:

Before you read the guidelines, please make sure that the following criteria apply to you:

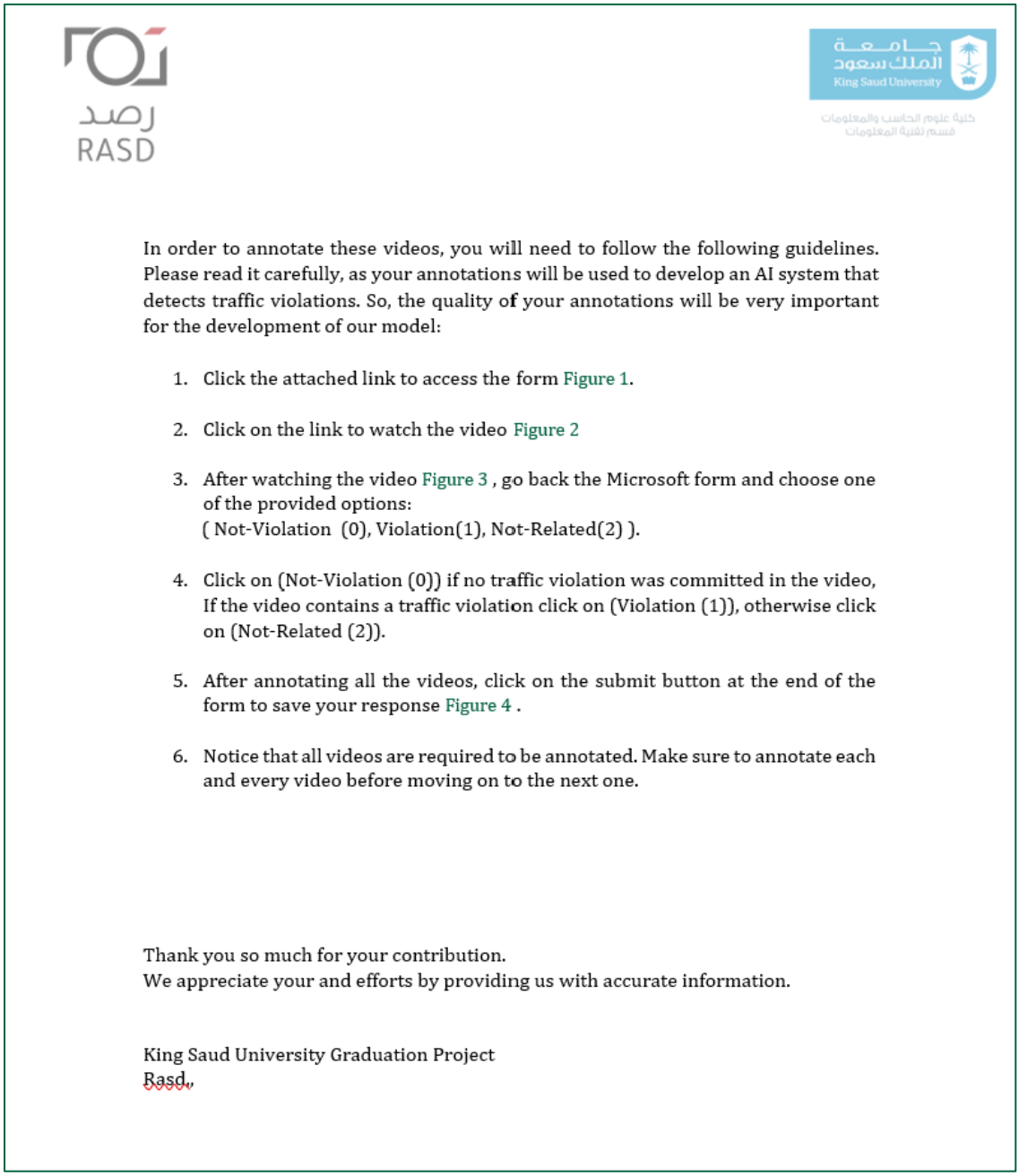
Annotator Criteria	
☐	The annotator must be 21 years old or older.
☐	The annotator should have at least 3 years of experience of driving in Saudi Arabia.
☐	The annotator must be familiar with the traffic rules in Saudi Arabia.
☐	The annotator should have a stable internet connection.

Provided to you a Microsoft forms link with 1176 traffic videos to be annotated as **Not-Violation** or **Violation** or **Not-Related**, which will be represented as (0,1,2) respectively according to the *list of penalties and traffic violations* provided by the General Department of Traffic.
Each annotator will be provided with approximately 200 videos to annotate, where:

- **Violation:** there is an activity caused by a vehicle that violates the traffic rules in Saudi Arabia.
- **Not-Violation:** there is no activity caused by a vehicle that violates the traffic rules in Saudi Arabia.
- **Not-Related:** anything that is not a vehicle or not related to traffic violations in general.

The attached videos are between 5 seconds and 2 minutes. No special software is required to see the videos because they will be displayed on a web page after clicking the associated link. Also, the video is provided with 3 options; one of the options must be selected.

Figure 93 Annotator Criteria



رصد
RASD

جامعة الملك سعود
King Saud University
كلية علوم الحاسب والمعلومات
قسم تقنية المعلومات

In order to annotate these videos, you will need to follow the following guidelines. Please read it carefully, as your annotations will be used to develop an AI system that detects traffic violations. So, the quality of your annotations will be very important for the development of our model:

1. Click the attached link to access the form Figure 1.
2. Click on the link to watch the video Figure 2
3. After watching the video Figure 3 , go back the Microsoft form and choose one of the provided options:
   (Not-Violation (0), Violation(1), Not-Related(2)).
4. Click on (Not-Violation (0)) if no traffic violation was committed in the video, If the video contains a traffic violation click on (Violation (1)), otherwise click on (Not-Related (2)).
5. After annotating all the videos, click on the submit button at the end of the form to save your response Figure 4 .
6. Notice that all videos are required to be annotated. Make sure to annotate each and every video before moving on to the next one.

Thank you so much for your contribution.
We appreciate your and efforts by providing us with accurate information.

King Saud University Graduation Project
Rasd,,

Figure 94 Annotation Guidelines

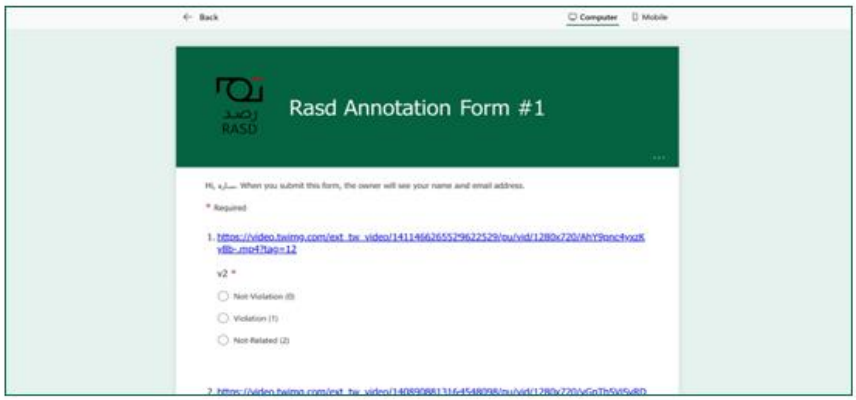


Figure 1

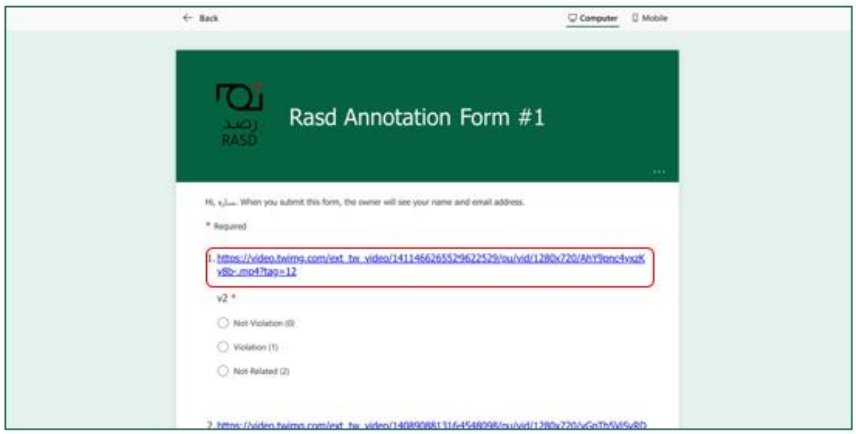


Figure 2

Figure 95 Annotation Guidelines

Figure 3

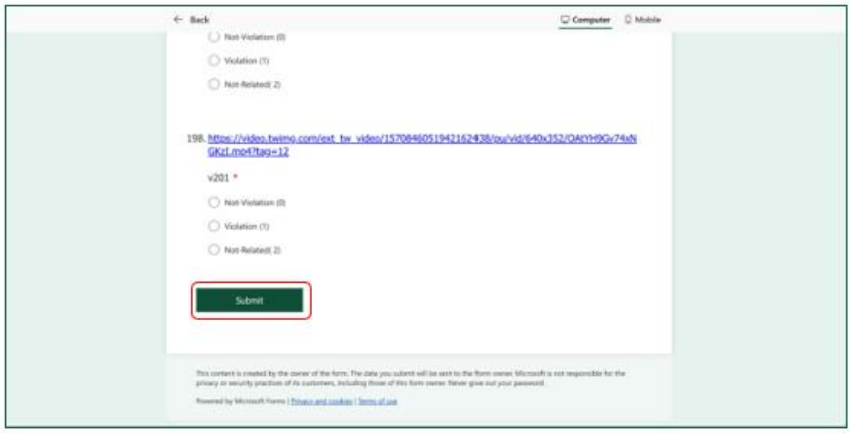


Figure 4

Figure 96 Annotation Guidelines

9.4 Appendix D: User Acceptance Testing Feedback

Figure 97 1st Question of User Acceptance Testing

Figure 98 2nd Question of User Acceptance Testing

Figure 99 3rd Question of User Acceptance Testing

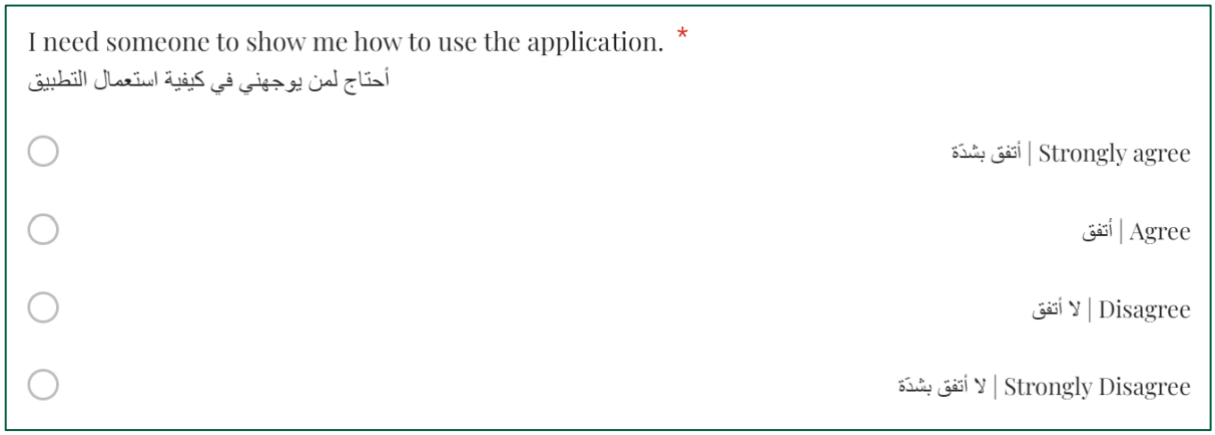


Figure 100 4th Question of User Acceptance Testing

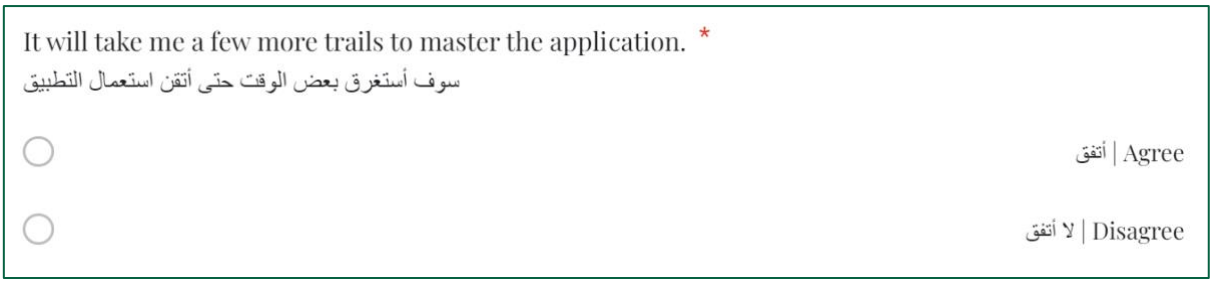


Figure 102 5th Question of User Acceptance Testing

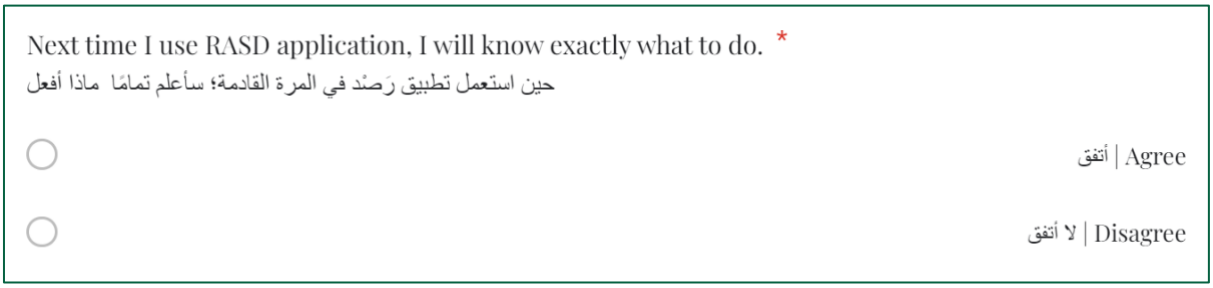


Figure 101 6th Question of User Acceptance Testing

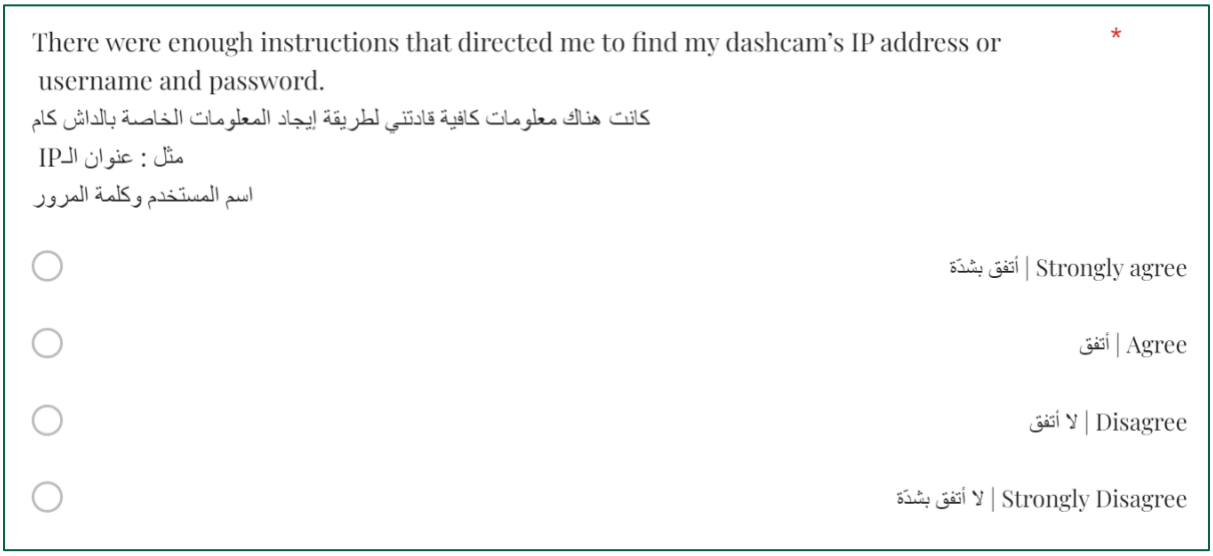


Figure 104 7th Question of User Acceptance Testing

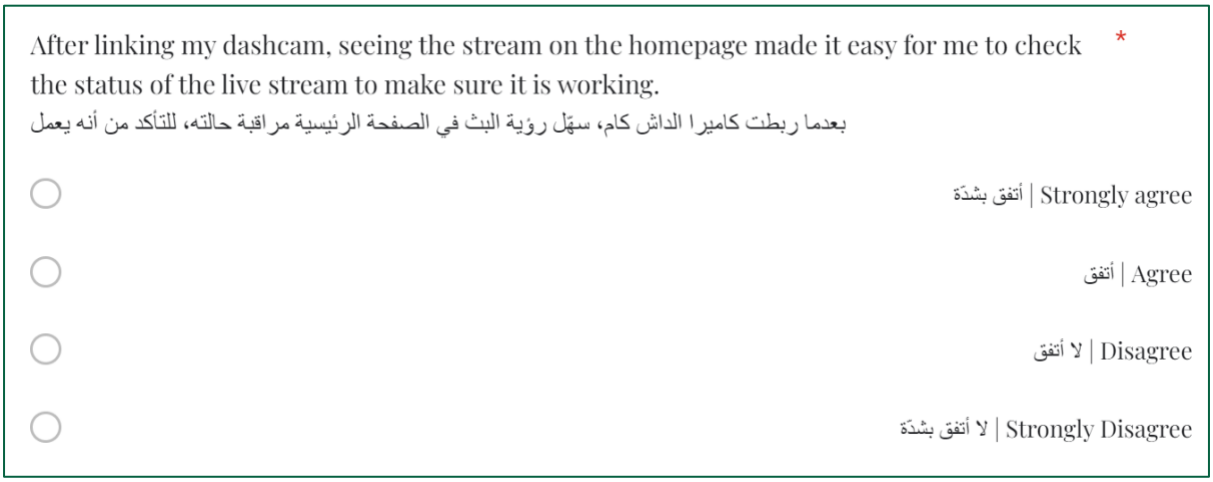


Figure 103 8th Question of User Acceptance Testing

Figure 106 9th Question of User Acceptance Testing

Figure 105 10th Question of User Acceptance Testing

Figure 108 11th Question of User Acceptance Testing

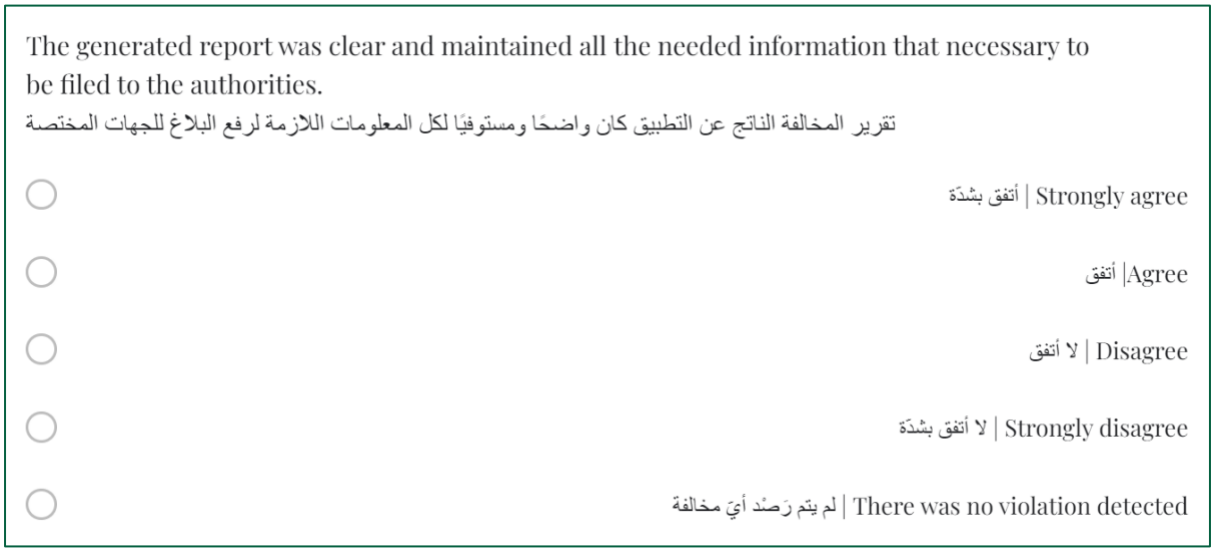


Figure 107 12th Question of User Acceptance Testing

Figure 110 13th Question of User Acceptance Testing

Figure 109 14th Question of User Acceptance Testing

9.5 Appendix E: UX Guidelines Figures Reference

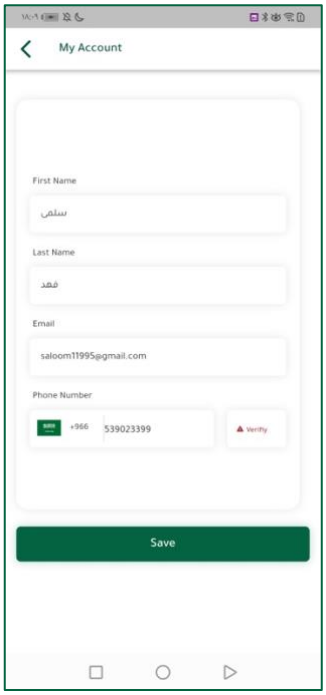


Figure 111 Save Button in 'My Account' page

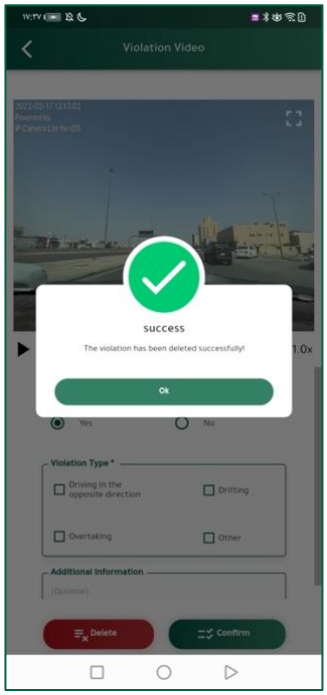


Figure 112 Delete Violation Video Confirmation Message

Figure 113 Delete Icon In report page

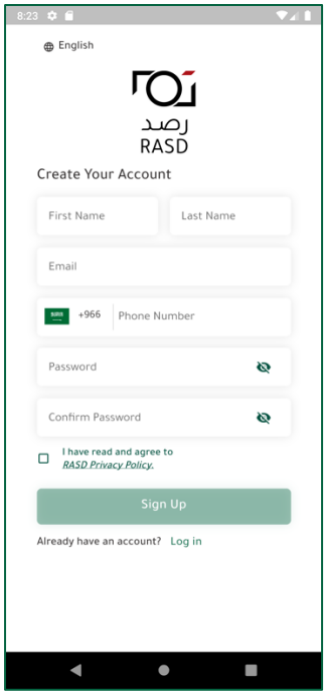


Figure 114 Sign Up Page

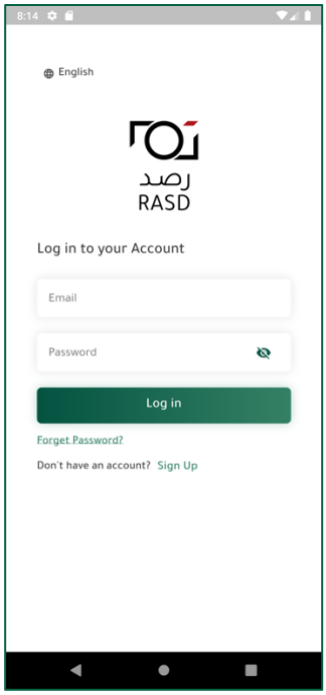


Figure 115 Log In Page

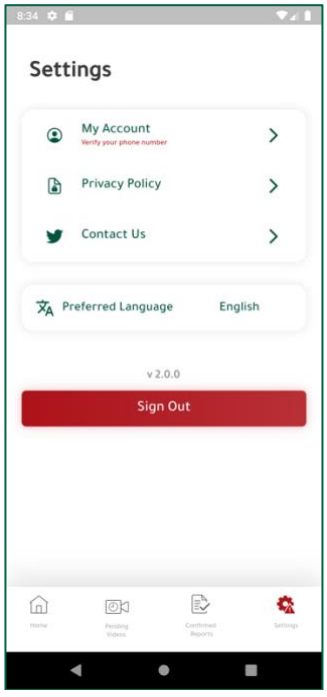


Figure 116 Settings Page

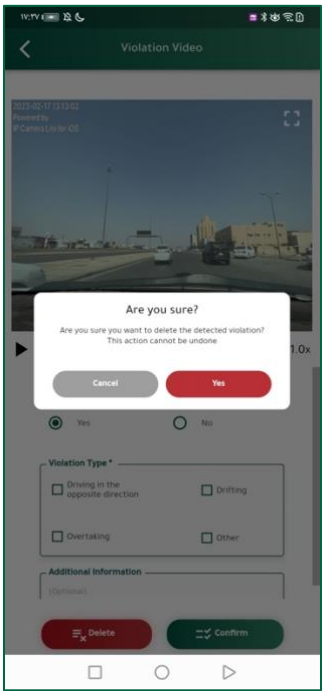


Figure 117 Delete Video

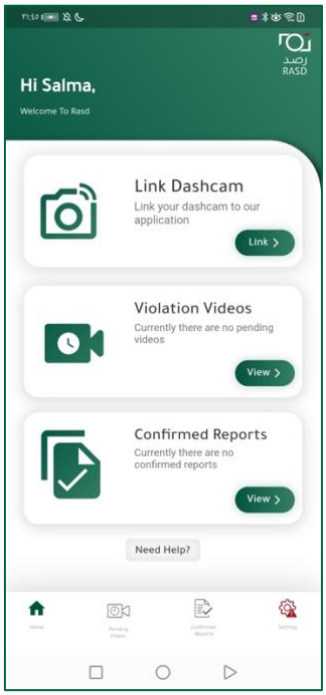


Figure 118 Bottom App Bar

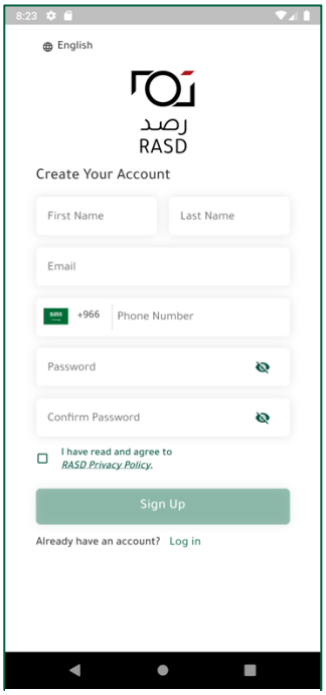


Figure 119 Log In Link