



# DSMS

Driver Safety Monitoring System

Summar Semester

COIT - 415

7 Aguest 2024

**Prepared By:**

Aseel Bahakim

**Supervised By:**

Entisar Alkayal

## TABLE OF CONTENTS

TABLE OF CONTENTS.....	II
LIST OF TABLES.....	IV
LIST OF FIGURES .....	V
LIST OF ABBREVIATIONS.....	VI
CHAPTER 1 .....	7
INTRODUCTION .....	7
1.1    Project Background .....	7
1.2    Project Objectives.....	8
1.3    Problem significant and scope.....	8
1.4    Project Description .....	8
1.5    Conclusion.....	8
CHAPTER 2 .....	10
SYSTEM ANALYSIS AND DESIGN.....	10
2.1    Analysis of System Requirements.....	10
2.1.1    Functional Requirements .....	10
2.1.2    Non-Functional requirements.....	11
2.2    Class Diagram .....	11
2.2.1    Classes, attributes, and operations .....	11
2.2.1    Relations with its names.....	12
2.2.2    Class diagram.....	13
2.3    Use Case Diagram .....	14
2.3.1    Actors Description .....	14
2.3.2    Use Cases Description.....	14
2.3.3    Use case diagram.....	22
2.4    Sequence Diagram.....	24
2.5    Activity Diagram.....	28
2.6    Conclusion.....	31
REFERENCES .....	32
APENDIX.....	33

## LIST OF TABLES

Table 2.1: DSMS Functional Requirements .....	10
Table 2.2: Non-Functional Requirements.....	11
Table 2.3: Classes, attributes, and operations of the Class Diagram .....	11
Table 2.4: Class Diagram Relations with their names.....	12
Table 2.5: DSMS Actor Description.....	14
Table 2.6: Register Use Case Description .....	14
Table 2.7: Login Use Case Description .....	15
Table 2.8: Capture Live Video Use Case Description.....	16
Table 2.9: View Report Use Case Description .....	17
Table 2.10: Manage Emergency Contact Use Case Description .....	17
Table 2.11: Manage Account Use Case Description .....	18
Table 2.12: Manage Monitoring Use Case Description.....	19
Table 2.13: Detect Critical Indicators Use Case Description .....	20
Table 2.14: Trigger Alert Use Case Description .....	20
Table 2.15: Send SMS Alert Use Case Description .....	21
Figure 2.4: Login Sequence Diagram .....	24

## LIST OF FIGURES

Figure 2.1: DSMS Class Diagram .....	13
Figure 2.2: DSMS Use Case Diagram.....	23
Figure 2.3: Register Sequence Diagram .....	24
Figure 2.4: Login Sequence Diagram .....	24
Figure 2.5: Manage Account Sequence Diagram .....	25
Figure 2.6: View Report Sequence Diagram .....	26
Figure 2.7: Manage Emergency Contact Sequence Diagram .....	26
Figure 2.8: Driver Safety Detection Sequence Diagram .....	27
Figure 2.9: Manage Monitoring Session Sequence Diagram .....	28
Figure 2.10: Monitoring Session and Camera Connection Activity Diagram.....	29
Figure 2.11: Detection Activity Diagram .....	30

## **LIST OF ABBREVIATIONS**

- SMS - Short Message Service  
NHTSA - National Highway Traffic Safety Administration  
UML. - Unified Modelling Language

# CHAPTER 1

## INTRODUCTION

### 1.1 Project Background

A lot of people rely on cars for mobility as it makes transportation a quicker and safer task. When you are driving, you don't have to worry about the weather, and you can load anything in just one car. It can also give you the freedom to stop anywhere at anytime. Additionally, better than that, when it comes to the brain and the cognitive function, driving makes your brain process a huge amount of information simultaneously, which can develop and grow neuroplasticity [1].

However, driving can be a huge problem if not managed correctly. The problem is driving in fatigue can cause drowsiness, a big contributor to road accidents. One study found that as you lose concentration while driving, you are responsible for a 25% of accidents [2]. Another study found that being distracted by the mobile phone while driving can increase the probability of getting a car accident by about 4 times compared to when you are not using the phone [3]. National Safety Council data, about 1.6 million people have lost their lives in crashes related to driver distraction. As a matter of fact, according to National Highway Traffic Safety Administration (NHTSA), at any given moment, about 660,00 drivers are attempting to use their phones while driving [4].

Another final and important factor, as of a study conducted by Mahler and colleagues found that people with epilepsy have a 71% higher chance of having accidental injuries [5]. The study also found that people with chronic diseases such as diabetes, strokes, and blood pressure have a higher risk of getting into accidents compared to people who do not suffer from this type of a disease [5].

As a result of these studies, statistics, and personal experience, having a system to be able to detect critical indicators such as drowsiness, distraction, and other health indicators such as seizures and remind and alert to take a break and making sure they stay safe is a one way to address this problem.

The subsequent sections will delve deeper into various aspects of this chapter. Section 1.2 outlines the objectives of the project, 1.3 presents the project's significant

and scope, 1.4 is about the project's description, and finally section 1.5 will conclude the chapter.

## **1.2 Project Objectives**

DSMS main objectives are to analyze, design, and develop a mobile application that alert the driver and send Short Message Service (SMS) alert to emergency contacts. A webcam that sends frames to the a deep learning and computer vision model that can receive these frames and promptly monitor and detect signs of drowsiness, such as yawning, and eye blinks, signs of distraction such cas using cell phone, talking to others and eating, and signs of fainting.

## **1.3 Problem significant and scope**

The significant of the project as outlined in section 1.1, lies in alerting drivers when they show signs of drowsiness or distraction or other critical indicators which can have several benefits:

1. enhancing drivers and pedestrians' safety
2. reducing yearly accidents rate
3. reducing medical costs and repairing costs and others which can have economic savings.
4. peace of mind for the driver's family who work long hours or night shifts

## **1.4 Project Description**

DSMS is a system that uses technologies like computer vision and a high-resolution camera to monitor critical indicators like eye closure, phone usage, seizures, yawning, and head movements. If it detects something dangerous, it alerts the driver. If the driver doesn't respond, it sends an SMS with their live location to an emergency contact. When the monitoring stops, the system generates a report on the driver's condition, which can be shared with doctors to help manage their health better. So, the idea aims to make driving safer for those who need help by providing real-time monitoring, alerts, and valuable health insights using advanced technologies.

## **1.5 Conclusion**

In conclusion, this chapter has conducted a comprehensive introduction to the project, and Throughout this chapter relevant scholarly references have been cited to

enhance the credibility of the examination. The next chapter will discuss and outline the design and analysis of the system using unified modeling language (UML) diagrams.

## CHAPTER 2

### SYSTEM ANALYSIS AND DESIGN

#### 2.1 Analysis of System Requirements

The software requirements describe the features and functionalities of the system. It is classified into functional requirements which are system's core functionalities, and non-functional requirements which are factors that extend beyond features and place constraints on the overall system.

##### 2.1.1 Functional Requirements

This section outlines the functional requirements of DSMS system in details.

**Table 2.1:** DSMS Functional Requirements

System Requirements	
Ref.ID	Description
<b>FR1</b>	The system shall allow the driver to register a new account.
<b>FR2</b>	The system shall allow the driver to log in to an existing account.
<b>FR3</b>	<p>The system shall allow the driver to manage account by</p> <ul style="list-style-type: none"> <li>• <b>FR3.1:</b> allowing the driver to change the password.</li> <li>• <b>FR3.2:</b> allowing the driver to log out.</li> <li>• <b>FR3.3:</b> allowing the driver to delete the account.</li> </ul>
<b>FR4</b>	<p>The system shall allow the driver to manage emergency contacts by</p> <ul style="list-style-type: none"> <li>• <b>FR4.1:</b> allowing the driver to add an emergency contact.</li> <li>• <b>FR4.2:</b> allowing the driver to delete an emergency contact.</li> </ul>
<b>FR5</b>	The system shall allow the driver to view reports.
<b>FR6</b>	<p>The system shall allow the driver to manage monitoring session by</p> <ul style="list-style-type: none"> <li>• <b>FR6.1:</b> allowing the driver to start the monitoring.</li> <li>• <b>FR6.2:</b> allowing the driver to stop the monitoring.</li> <li>• <b>FR6.3:</b> allowing the driver to connect to the camera via Wi-Fi.</li> </ul>
<b>FR7</b>	The system shall trigger an alert if any critical conditions are detected.
<b>FR8</b>	The system shall send an SMS message with the driver's live location to the emergency contact if the driver does not respond to the alert.

<b>FR9</b>	The system shall detect critical indicators including: <ul style="list-style-type: none"> <li>• <b>FR10.1:</b> The system shall detect drowsiness indicators.</li> <li>• <b>FR10.2:</b> The system shall detect phone usage.</li> <li>• <b>FR10.3:</b> The system shall detect indicators of fainting.</li> </ul>
<b>FR10</b>	The system shall capture a live video of the driver.

### 2.1.2 Non-Functional requirements

This section outlines the non-functional requirements of DSMS system in details.

**Table 2.2:** Non-Functional Requirements

Ref.ID	Description
<b>NFR1</b>	<b>Availability:</b> The system must be available to be used by users at any time.
<b>NFR2</b>	<b>Usability:</b> The system should be easy and clear to use for users.
<b>NFR3</b>	<b>Performance:</b> The system will be able to return results faster and must be able to handle the required number of users without any degradation in performance.
<b>NFR4</b>	<b>Security:</b> The system must be secure from unauthorized access since it holds sensitive information like reports.
<b>NFR5</b>	<b>Speed:</b> Since the app will be designed to send an emergency text to the Emergency contacts, it should react quickly without any delay.

## 2.2 Class Diagram

This section outlines the class diagram of the system and two tables related to the diagram, table 2.3 and table 2.4.

### 2.2.1 Classes, attributes, and operations

**Table 2.3:** Classes, attributes, and operations of the Class Diagram

Class	Attributes	Operations
<b>Person</b>	personId, name, phoneNumber	getPersonId(), getName(), getPhoneNumber(), setName(String name), setPhoneNumber(String phoneNumber)
<b>EmergencyContact</b>	contactId, relationship	receiveAlert(), addEmergencyContact(Person contact, String relationship), deleteEmergencyContact(int contactId)

<b>Account</b>	accountId, password, email	login(), register(), getContent(), changePassword(String newPassword), logout(), deleteAccount(), getAccountId(), getPassword(), getEmail(), setPassword(String password), setEmail(String email)
<b>MonitoringSystem</b>	monitoringId	startMonitoring(), stopMonitoring(), sendAlert(), getMonitoringId()
<b>Camera</b>	cameraId, frames	connectViaWiFi(), captureLiveVideo(), getCameraId()
<b>Frame</b>	frameId	getFrameId(), getFrame()
<b>Model</b>	modelId	detectDrowsiness(Frame frame), detectPhoneUsage(Frame frame), detectFainting(Frame frame), triggerAlert()
<b>Report</b>	reportId, dataGenerated, content	getReportId(), getDataGenerated(), getContent(), setContent(String content)

### 2.2.1 Relations with its names

**Table 2.4:** Class Diagram Relations with their names

Classes	Relationship	name	Multiplicity
<b>Person, Account</b>	Person class is a superclass to Account class, it passes its attributes to it	Generalization	-
<b>MonitoringSystem, Account</b>	MonitoringSystem creates Account	Generalization	-
<b>MonitoringSystem, Camera</b>	MonitoringSystem connects to Camera	Association	1 – 1
<b>MonitoringSystem, Model</b>	MonitoringSystem uses Model	Association	1 – 1
<b>Model, Frame</b>	Model aggregates multiple Frame instances	Aggregation	1 - 1..*
<b>MonitoringSystem, Report</b>	MonitoringSystem generates multiple Report instances	Association	1 - 1..*
<b>Account, Report</b>	Account views Report	Association	1 - 1..*
<b>EmergencyContact, Account</b>	Account creates EmergencyContact	Association	1 - 1..*
<b>MonitoringSystem, EmergencyContact</b>	MonitoringSystem message EmergencyContact	Association	1 – 1..*
<b>Camera, Frame</b>	Camera send frames	Composition	1 - 1..*

## 2.2.2 Class diagram

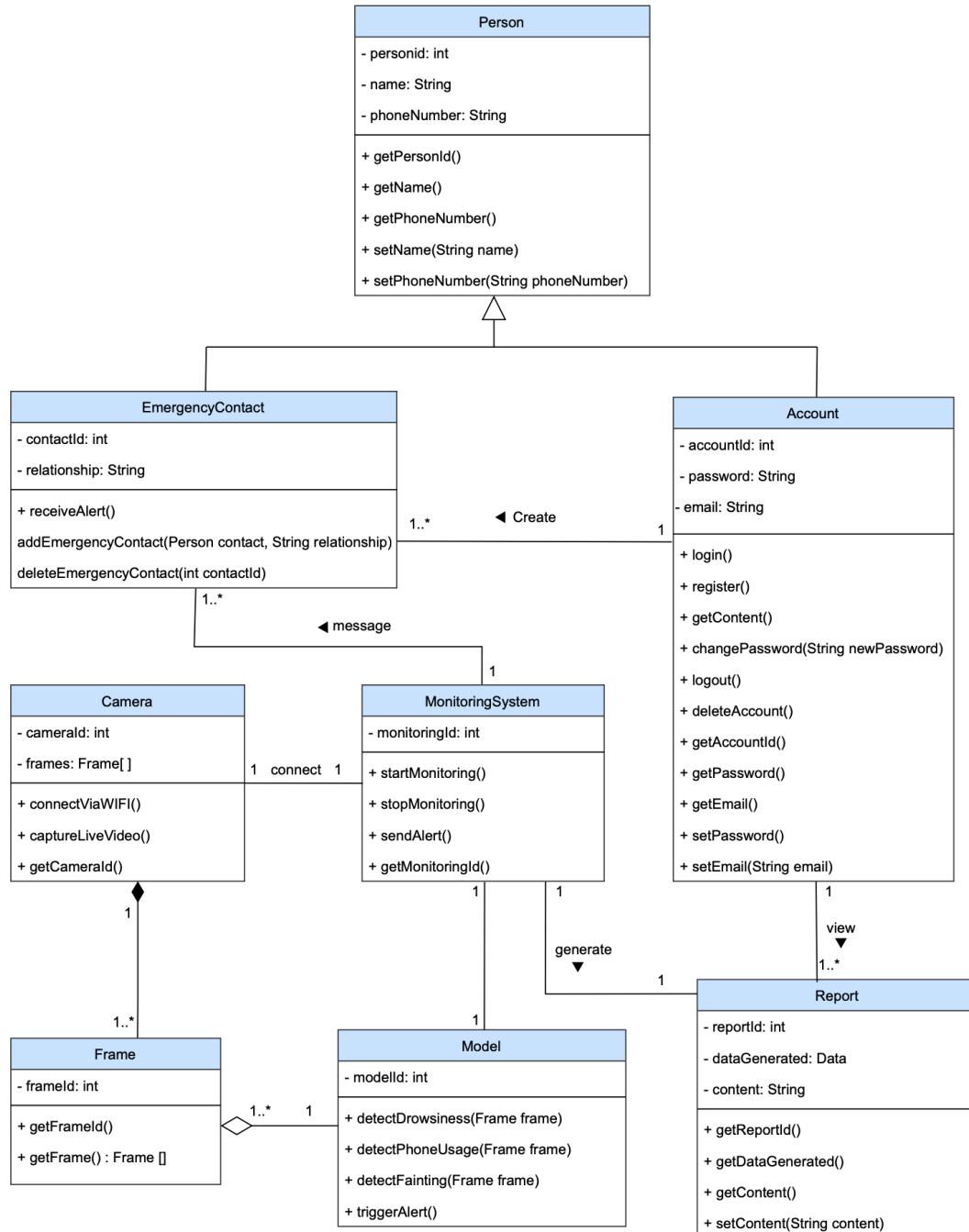


Figure 2.1: DSMS Class Diagram

## 2.3 Use Case Diagram

### 2.3.1 Actors Description

An actor in the use case diagram is any entity interacting with the system. An actor can be a person, a system, or an organization. Table 2.5 details each actor as depicted in the use case diagram.

**Table 2.5:** DSMS Actor Description

Actor	Description
<b>Driver</b>	The primary user of the system who operates the vehicle.
<b>Emergency Contact</b>	A person or more chosen by the driver to receive SMS alerts with live location in case of emergencies.
<b>Camera</b>	A device that captures live video of the driver and connects to the mobile application via Wi-Fi and sends frames to the model.

### 2.3.2 Use Cases Description

This section provides a detailed description of each use case, specifying how actors can interact with the system. The interactions and functionalities are clearly illustrated in tables from Table 2.6 to Table 2.18

**Table 2.6:** Register Use Case Description

<b>Use Case Name</b>	Register	
<b>Scenario</b>	A new driver registers into the system.	
<b>Actor</b>	Driver	
<b>Triggering Event</b>	The Driver clicks on the "Register" button.	
<b>Brief Description</b>	This use case describes the process of a new driver creating a new account by entering email, username and password to access the system's functionalities.	
<b>Related Use Case</b>	Login.	
<b>Pre-condition</b>	The Driver must have installed the mobile application on their device.	
<b>Post-condition</b>	The driver's account is successfully created with their credentials securely stored in the database.	
<b>Flow Of Activities</b>	<b>Actor</b>	<b>System</b>

	<ol style="list-style-type: none"> <li>1. Launches the mobile application on their device.</li> <li>2. Click on the registration button.</li> <li>4. Enters required information into the designated fields on the registration page.</li> <li>5. Click on the 'Register' button.</li> </ol>	<ol style="list-style-type: none"> <li>3. Displays a registration page for the driver to input their information.</li> <li>6. Validate the provided information</li> <li>7. Creates a new user account upon successful validation.</li> <li>8. Add driver's information to the database.</li> <li>9. Redirect the driver to the login page after account creation.</li> </ol>
<b>Exception Condition</b>	The driver unintentionally provides incomplete or inaccurate information during the registration process.	
<b>Exception Handling</b>	The system highlights specific fields needing attention.	

**Table 2.7:** Login Use Case Description

<b>Use Case Name</b>	Login	
<b>Scenario</b>	The driver logs into the system.	
<b>Actor</b>	driver	
<b>Triggering Event</b>	The driver clicks on the "Login" button.	
<b>Brief Description</b>	This use case describes the process where an existing driver logs into their account in the system to access system functionalities.	
<b>Related Use Case</b>	-	
<b>Pre-condition</b>	The driver must have an existing account in the system.	
<b>Post-condition</b>	The driver is granted access to their account and is directed to the landing page.	
<b>Flow Of Activities</b>	<b>Actor</b>	<b>System</b>

	<ol style="list-style-type: none"> <li>1. Launches the mobile application on their device.</li> <li>3. Enters Required Information into the Designated Fields on the Login Page.</li> <li>4. Clicks on the Login Button</li> </ol>	<ol style="list-style-type: none"> <li>2. Displays a login page for the user to input their credentials.</li> <li>5. Validates the credentials against the system's driver database.</li> <li>7. Grants access if the credentials are correct.</li> <li>8. Redirect the driver to the landing page.</li> </ol>
<b>Exception Condition</b>	The driver enters incorrect credentials during the login process.	
<b>Exception Handling</b>	The system detects that the entered credentials do not match any record in the database and displays an error message.	

**Table 2.8:** Capture Live Video Use Case Description

<b>Use Case Name</b>	Capture Live Video					
<b>Scenario</b>	The camera captures a live video feed of the driver.					
<b>Actor</b>	Camera					
<b>Triggering Event</b>	The driver starts the monitoring session.					
<b>Brief Description</b>	This use case describes the process where the camera captures a live video of the driver to monitor their condition in real-time.					
<b>Related Use Case</b>	Detect Critical Indicators					
<b>Pre-condition</b>	The camera must be connected to the system via Wi-Fi.					
<b>Post-condition</b>	The system receives a continuous live video feed from the camera.					
<b>Flow Of Activities</b>	<table border="1"> <thead> <tr> <th>Actor</th> <th>System</th> </tr> </thead> <tbody> <tr> <td> <ol style="list-style-type: none"> <li>1. Connects to the system via Wi-Fi.</li> <li>3. Starts capturing live video.</li> </ol> </td> <td> <ol style="list-style-type: none"> <li>2. Verifies the Wi-Fi connection.</li> <li>4. Establishes a live video feed from the camera.</li> <li>5. Displays the live video feed.</li> <li>6. Continuously monitors and processes the frames.</li> </ol> </td> </tr> </tbody> </table>	Actor	System	<ol style="list-style-type: none"> <li>1. Connects to the system via Wi-Fi.</li> <li>3. Starts capturing live video.</li> </ol>	<ol style="list-style-type: none"> <li>2. Verifies the Wi-Fi connection.</li> <li>4. Establishes a live video feed from the camera.</li> <li>5. Displays the live video feed.</li> <li>6. Continuously monitors and processes the frames.</li> </ol>	
Actor	System					
<ol style="list-style-type: none"> <li>1. Connects to the system via Wi-Fi.</li> <li>3. Starts capturing live video.</li> </ol>	<ol style="list-style-type: none"> <li>2. Verifies the Wi-Fi connection.</li> <li>4. Establishes a live video feed from the camera.</li> <li>5. Displays the live video feed.</li> <li>6. Continuously monitors and processes the frames.</li> </ol>					
<b>Exception Condition</b>	The camera is not connected when the monitoring session starts.					

<b>Exception Handling</b>	The system prompts the driver to connect the camera before starting the monitoring session.
---------------------------	---

**Table 2.9:** View Report Use Case Description

<b>Use Case Name</b>	View Report	
<b>Scenario</b>	The driver views the report.	
<b>Actor</b>	Driver	
<b>Triggering Event</b>	The driver chooses to view one of the reports.	
<b>Brief Description</b>	This use case describes the process where the driver access and view one of the reports.	
<b>Related Use Case</b>	Login	
<b>Pre-condition</b>	The driver must be logged into their account.	
<b>Post-condition</b>	The driver views the report.	
<b>Flow Of Activities</b>	<b>Actor</b>	<b>System</b>
	1. Opens the mobile application 3. Navigates to the reports section. 4. Clicks on one of the reports	2. Display the main page 5. Retrieves and displays the report.
<b>Exception Condition</b>	The driver is not logged in	
<b>Exception Handling</b>	The system prompts the driver to log in	

**Table 2.10:** Manage Emergency Contact Use Case Description

<b>Use Case Name</b>	Manage Emergency Contact	
<b>Scenario</b>	The driver manages their emergency contact information.	
<b>Actor</b>	Driver	
<b>Triggering Event</b>	The driver chooses to add, or delete an emergency contact.	
<b>Brief Description</b>	This use case describes the process where the drive manages (add or delete) their emergency contact information.	
<b>Related Use Case</b>	Login	
<b>Pre-condition</b>	The driver must be logged into their account.	
<b>Post-condition</b>	The emergency contact information is updated in the system.	

	<b>Actor</b>	<b>System</b>
<b>Flow Of Activities</b>	1. Opens the mobile application 3. Navigates to the emergency contacts section. 5. Selects to add, or delete an emergency contact. 6. Save changes	2. Display the main page 4. Displays the emergency contacts page. 7. Saves the changes to the database.
<b>Exception Condition</b>	The driver is not logged in	
<b>Exception Handling</b>	The system prompts the driver to log in	

**Table 2.11:** Manage Account Use Case Description

<b>Use Case Name</b>	Manage Account.	
<b>Scenario</b>	The driver can manage their account using the various functions provided by the application.	
<b>Actor</b>	driver	
<b>Triggering Event</b>	The driver clicks on the "≡" icon.	
<b>Brief Description</b>	This use case describes the process of the driver viewing their account management options which are changing their password, deleting their account, or logging out.	
<b>Related Use Case</b>	Login.	
<b>Pre-condition</b>	The driver must have an account and logged into the application.	
<b>Post-condition</b>	The driver has access to all the manage account's options.	
	<b>Actor</b>	<b>System</b>
<b>Flow Of Activities</b>	1. The driver clicks on the "≡" icon to open the side drawer menu	2. System displays the account management screen with options: Change password, delete account, Logout.
<b>Exception Condition</b>	The driver is not logged in	
<b>Exception Handling</b>	The system prompts the driver to log in	

**Table 2.12:** Manage Monitoring Use Case Description

<b>Use Case Name</b>	Manage Monitoring	
<b>Scenario</b>	The driver manages the monitoring process, including starting and stopping the monitoring session or connect camera via Wi-Fi.	
<b>Actor</b>	Driver	
<b>Triggering Event</b>	The driver chooses to start, stop the monitoring session or connect camera via Wi-Fi.	
<b>Brief Description</b>	This use case describes the process where the driver start or stop the monitoring of their driving conditions connect camera via Wi-Fi.	
<b>Related Use Case</b>	Login	
<b>Pre-condition</b>	The driver must be logged into their account and the camera must be connected.	
<b>Post-condition</b>	The monitoring session is started or stopped as per the driver's command.	
<b>Flow Of Activities</b>	<b>Actor</b>	<b>System</b>
	1. Opens the mobile application 3. Navigates to the monitoring section. 4. Connect camera via Wi-Fi 6. Selects the "Start Monitoring" or "Stop Monitoring" option.	2. Display the main page 5. Displays the monitoring page with options to start if not started or stop monitoring if already started. 7. If "Start Monitoring" is selected, system checks the camera connection and other necessary components. 8. If "Stop Monitoring" is selected, the system stops capturing frames and ends the monitoring session, then processes and saves the collected data for the report.

<b>Exception Condition</b>	The camera is not connected
<b>Exception Handling</b>	The system prompts the driver to ensure the camera is connected

**Table 2.13:** Detect Critical Indicators Use Case Description

<b>Use Case Name</b>	Detect Critical Indicators	
<b>Scenario</b>	The system detects various indicators such as drowsiness, phone usage, and signs of fainting.	
<b>Actor</b>	Cloud Service	
<b>Triggering Event</b>	The monitoring system is actively monitoring the driver.	
<b>Brief Description</b>	This use case describes the process where the system continuously monitors the driver's condition to detect critical indicators.	
<b>Related Use Case</b>	Send SMS Alert	
<b>Pre-condition</b>	The monitoring session must be active, and the camera must be capturing live video, and sending frames.	
<b>Post-condition</b>	The system detects any indicators and logs the event or triggers an alert if necessary.	
<b>Flow Of Activities</b>	<b>Actor</b> 2. The model receives frames from the live video feed captured by the camera. 3. Processes each frame to detect any critical indicators. 4. If an indicator is detected, the model flags the frame and notifies the system.	<b>System</b> 1. The system continuously sends frames to the model for processing. 5. Logs the event and triggers an alert if necessary
<b>Exception Condition</b>	The camera resolution is insufficient for accurate detection.	
<b>Exception Handling</b>	Requires high-resolution camera for precise and accurate detection	

**Table 2.14:** Trigger Alert Use Case Description

<b>Use Case Name</b>	Trigger Alert
<b>Scenario</b>	The system triggers an alert when a critical indicator is detected.

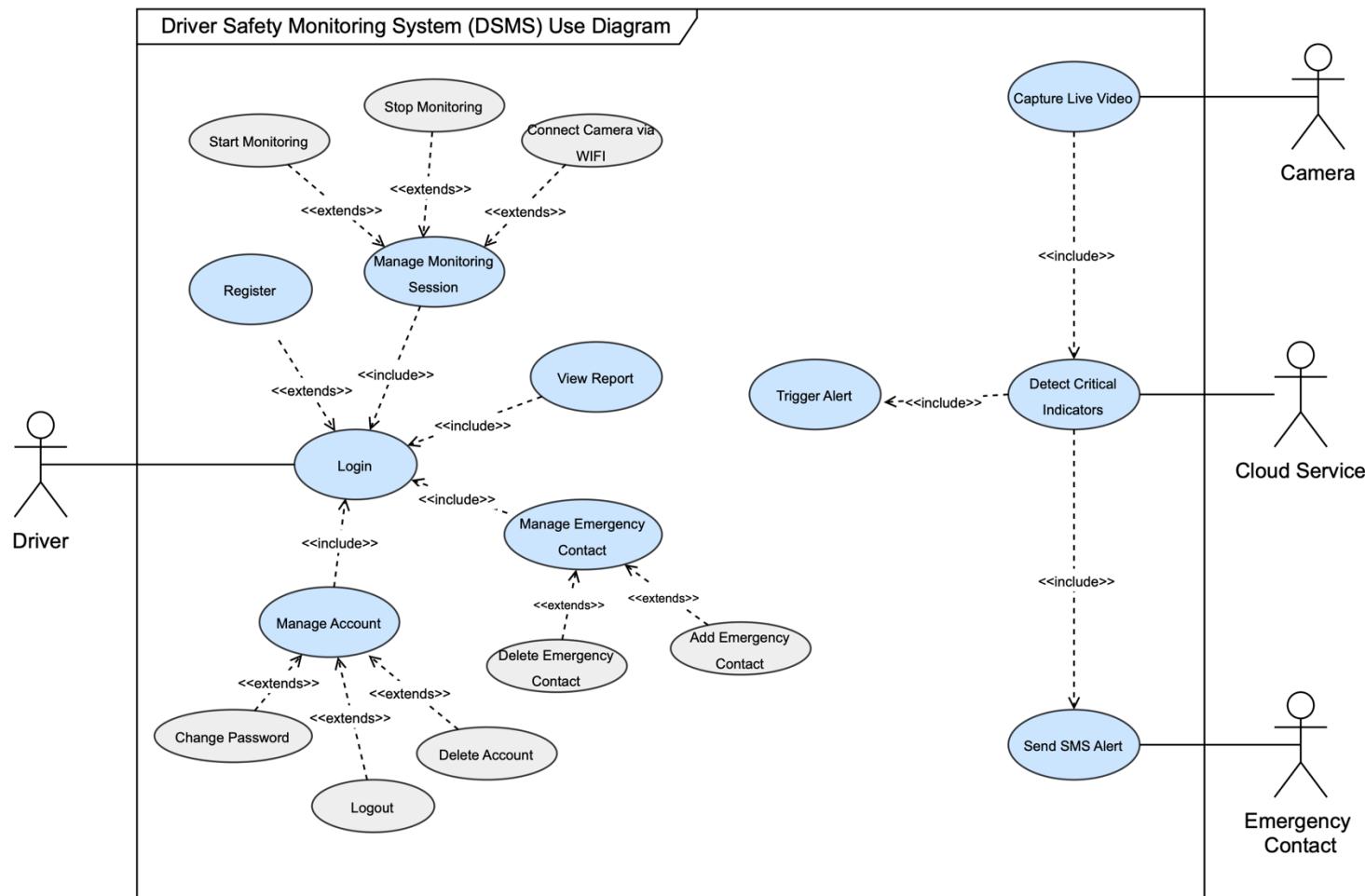
<b>Actor</b>	Cloud Service	
<b>Triggering Event</b>	The system detects a critical indicator such as drowsiness, phone usage, or signs of fainting.	
<b>Brief Description</b>	This use case describes the process where the system triggers an alert based on the detection of critical indicators.	
<b>Related Use Case</b>	-	
<b>Pre-condition</b>	The system must be actively monitoring the driver and must detect a critical indicator.	
<b>Post-condition</b>	An alert is triggered	
<b>Flow Of Activities</b>	<b>Actor</b>	<b>System</b>
	4. Displays the alert to the driver.	1. Continuously monitors and processes data from the model. 2. Detects a critical indicator 3. Logs the detected condition.
<b>Exception Condition</b>	The system fails to trigger the alert.	
<b>Exception Handling</b>	The system logs the failure and attempts to retrigger the alert	

**Table 2.15:** Send SMS Alert Use Case Description

<b>Use Case Name</b>	Send SMS Alert
<b>Scenario</b>	The system sends an SMS alert to the emergency contact when a critical condition is detected and the driver does not respond within a specified period.
<b>Actor</b>	Emergency Contact
<b>Triggering Event</b>	The system triggers an alert based on the detection of a critical condition and the driver fails to respond within the specified time.
<b>Brief Description</b>	This use case describes the process where the system sends an SMS alert to the emergency contact with the driver's location and details of the critical indicator detected, if the driver does not respond within a specified time.
<b>Related Use Case</b>	-

<b>Pre-condition</b>	The system must have detected a critical indicator, the alert must have been triggered, and the driver is not responding within the specified time.	
<b>Post-condition</b>	An SMS alert is sent to the emergency contact.	
<b>Flow Of Activities</b>	<p><b>Actor</b></p> <ol style="list-style-type: none"> <li>1. Triggers an alert and waits for the driver's response.</li> <li>2. If the driver does not respond within the specified time, gather necessary information such as the driver's live location</li> <li>5. Sends the SMS alert to the emergency contact.</li> <li>6. Displays notification to the driver confirming the alert has been sent.</li> </ol>	<p><b>System</b></p> <ol style="list-style-type: none"> <li>3. Retrieves emergency contact information from the database.</li> <li>4. Generates the SMS alert content.</li> <li>7. Logs the SMS alert event in the database for the report.</li> </ol>
<b>Exception Condition</b>	The system fails to send the SMS alert.	
<b>Exception Handling</b>	The system retries sending the SMS alert and logs the failure	

### 2.3.3 Use case diagram



**Figure 2.2:** DSMS Use Case Diagram

## 2.4 Sequence Diagram

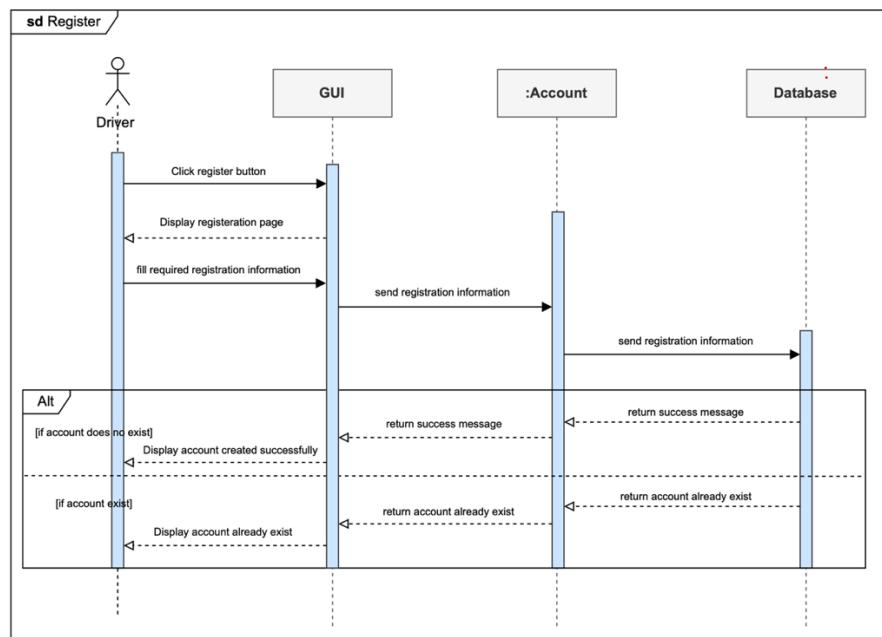


Figure 2.3: Register Sequence Diagram

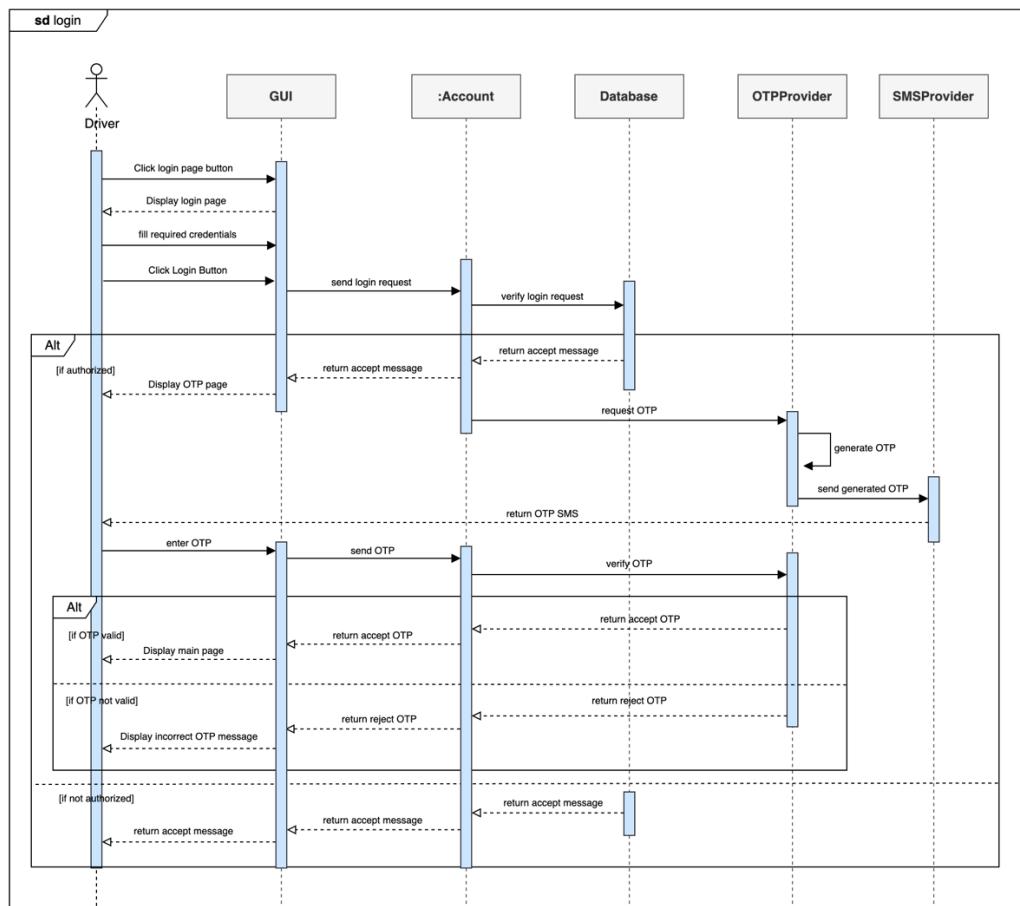
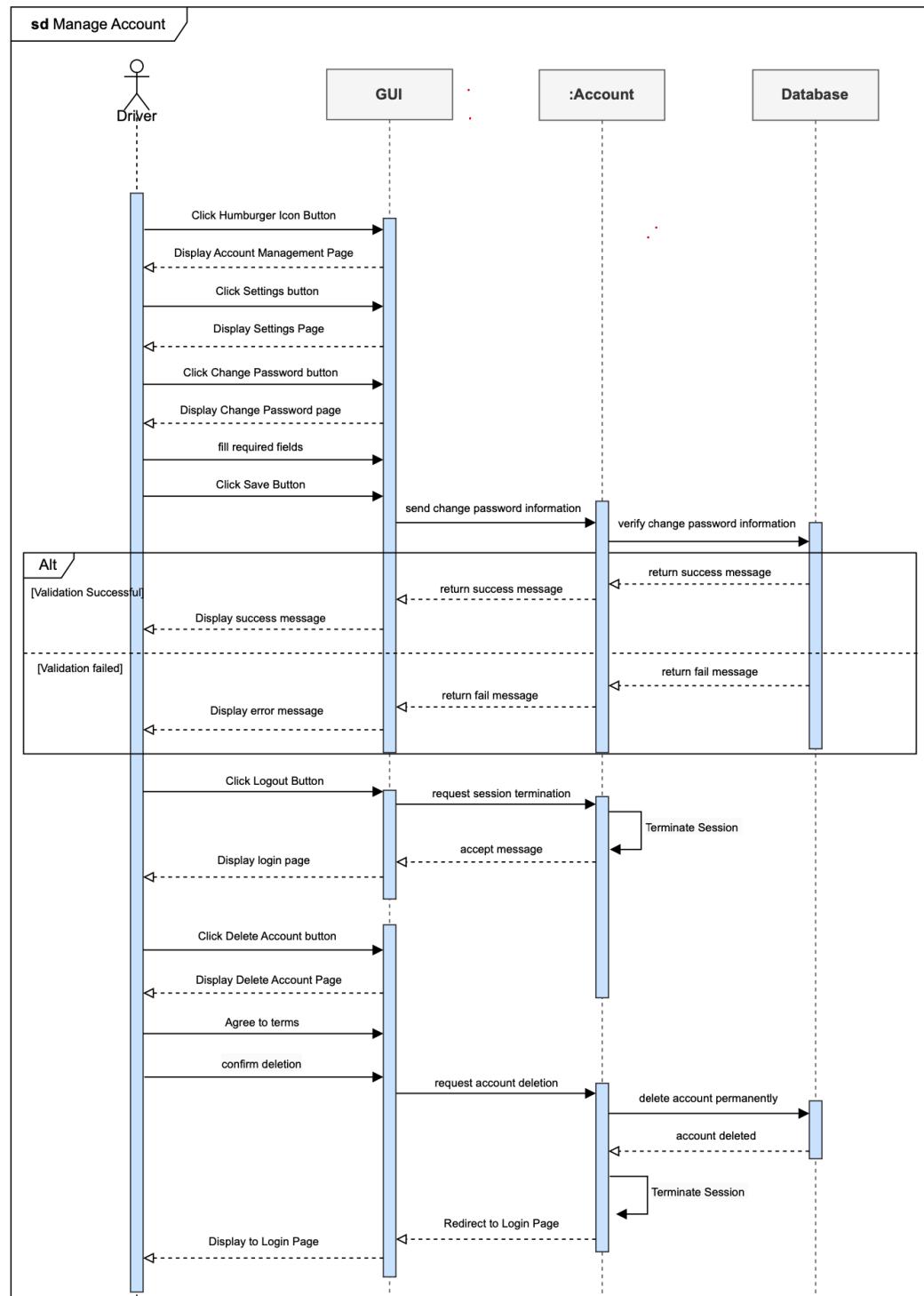
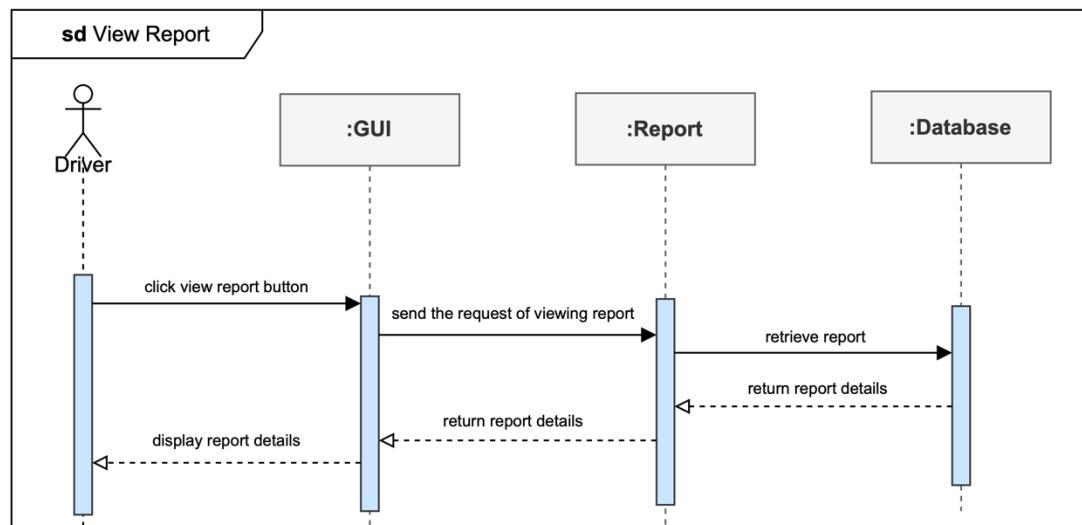


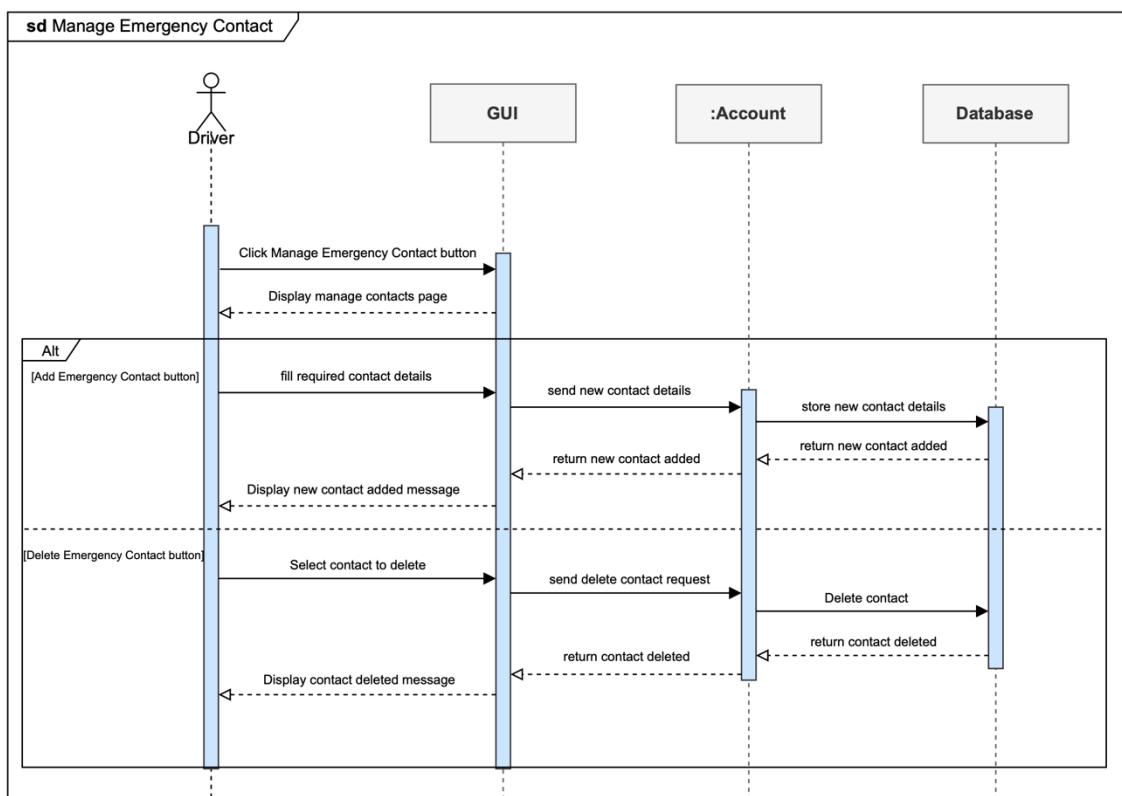
Figure 1.4: Login Sequence Diagram



**Figure 2.5:** Manage Account Sequence Diagram



**Figure 2.6:** View Report Sequence Diagram



**Figure 2.7:** Manage Emergency Contact Sequence Diagram

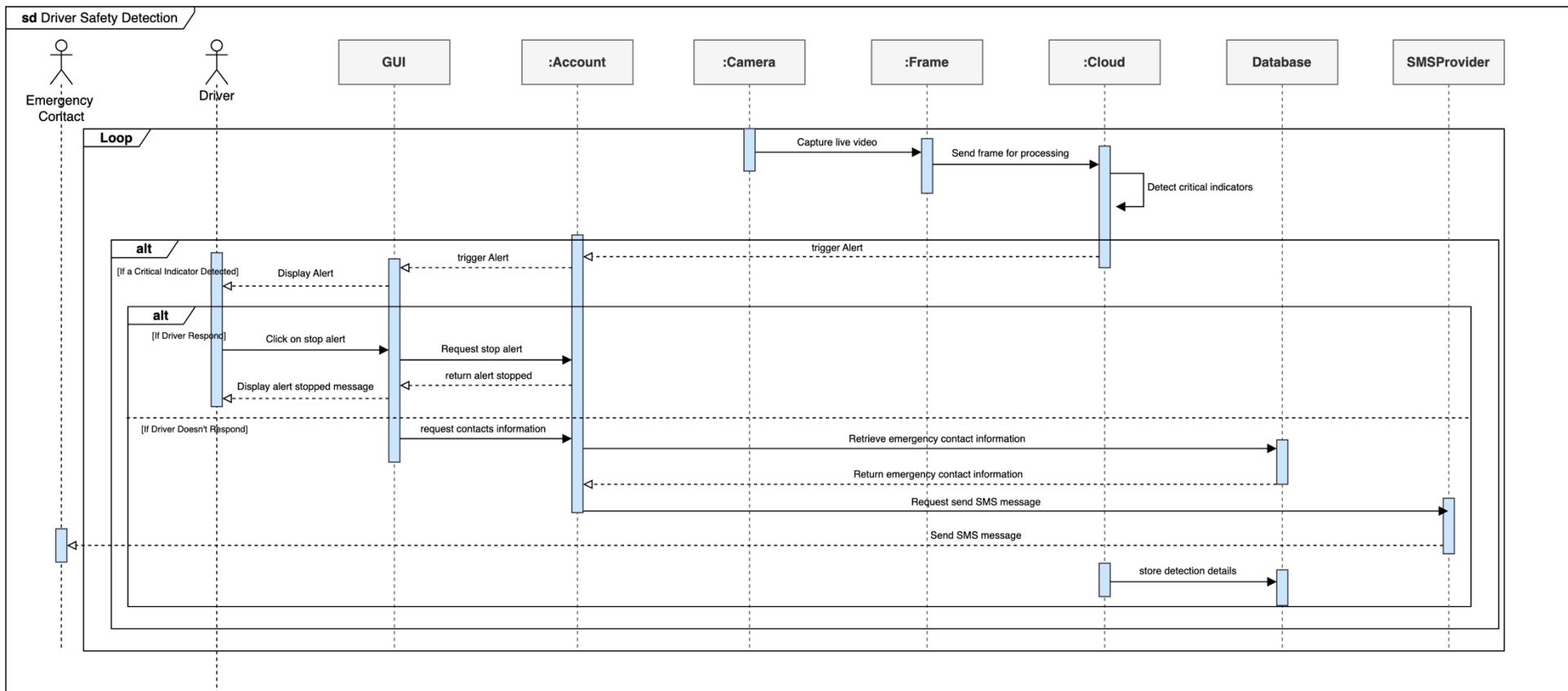
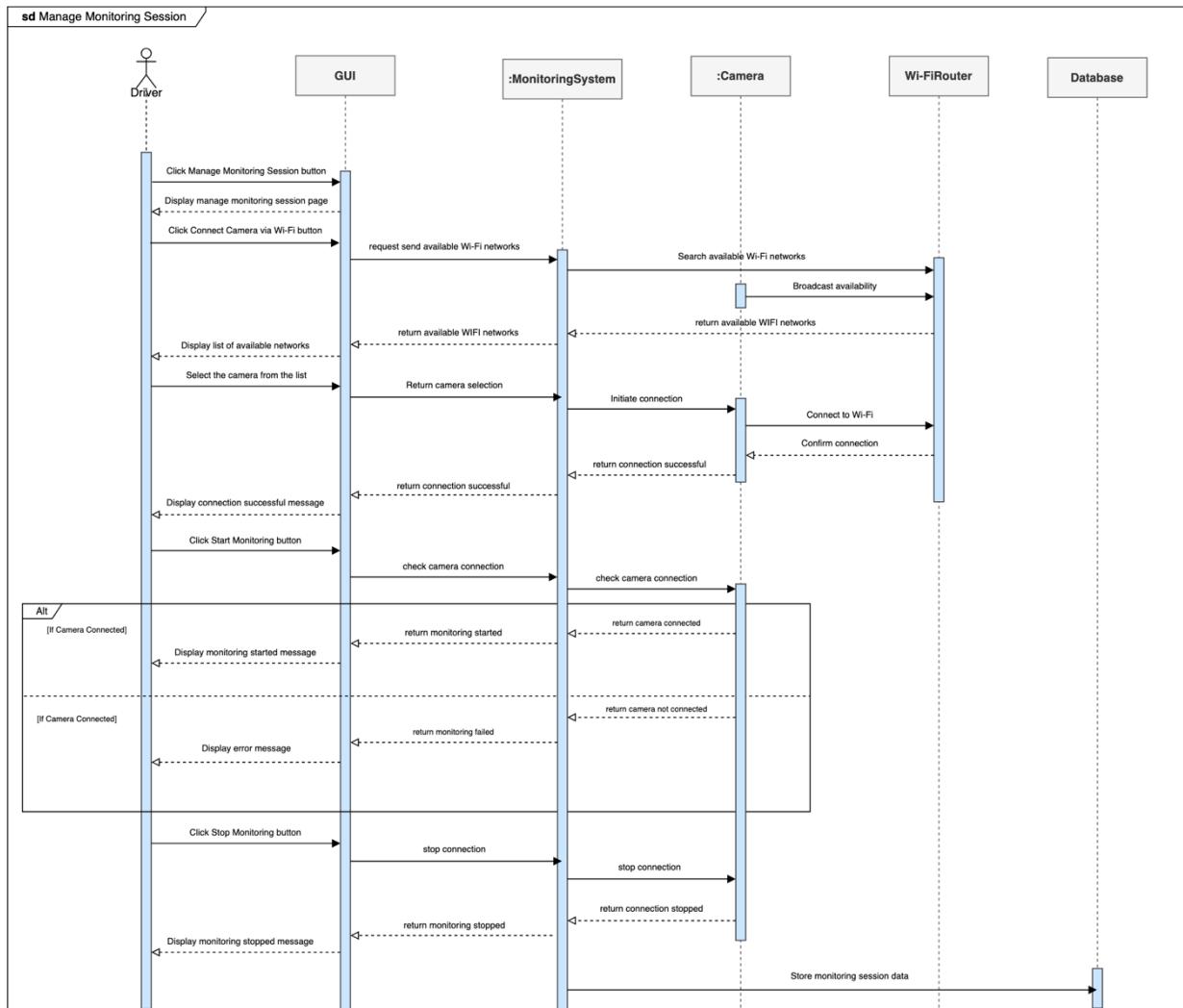


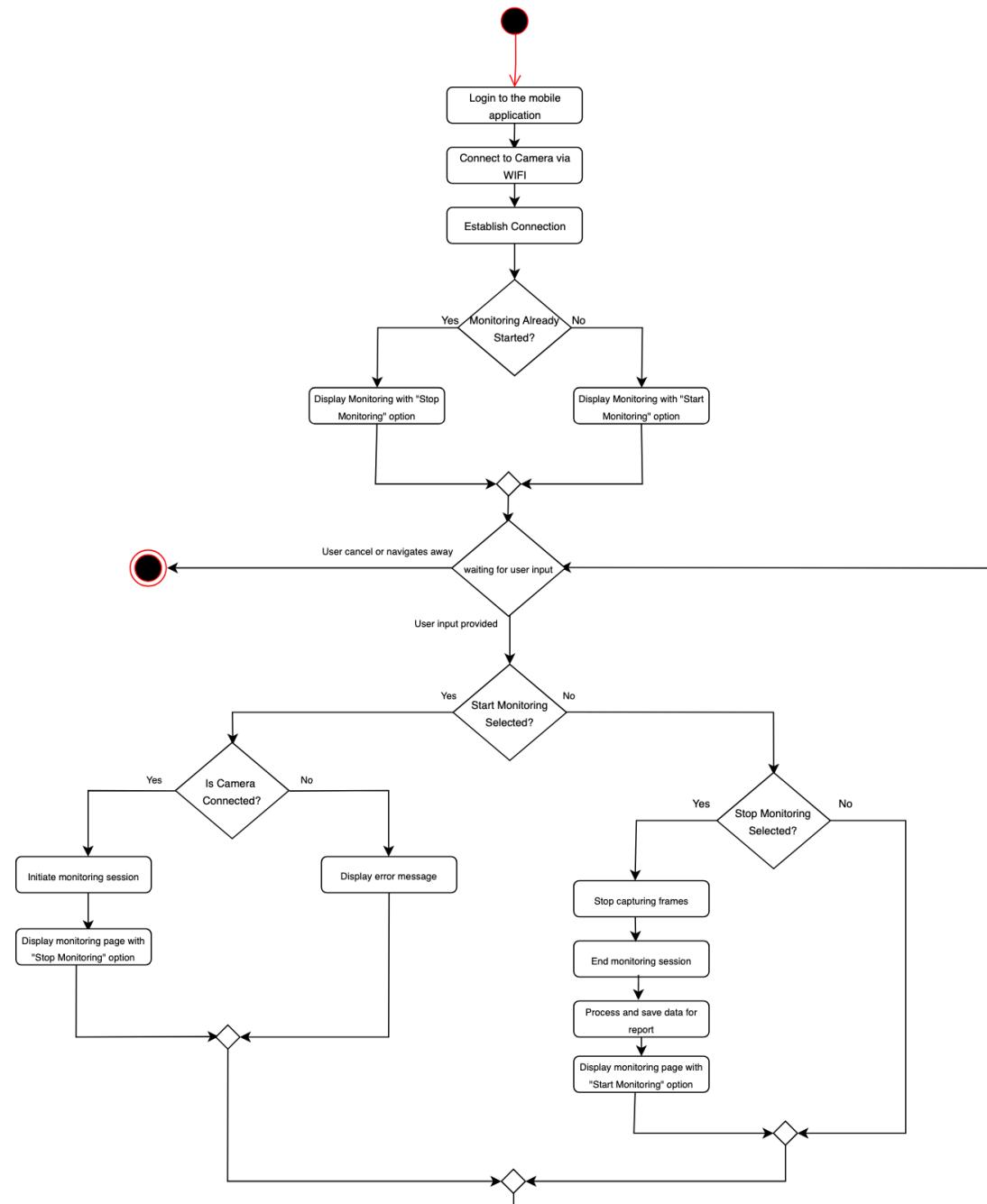
Figure 2.8: Driver Safety Detection Sequence Diagram



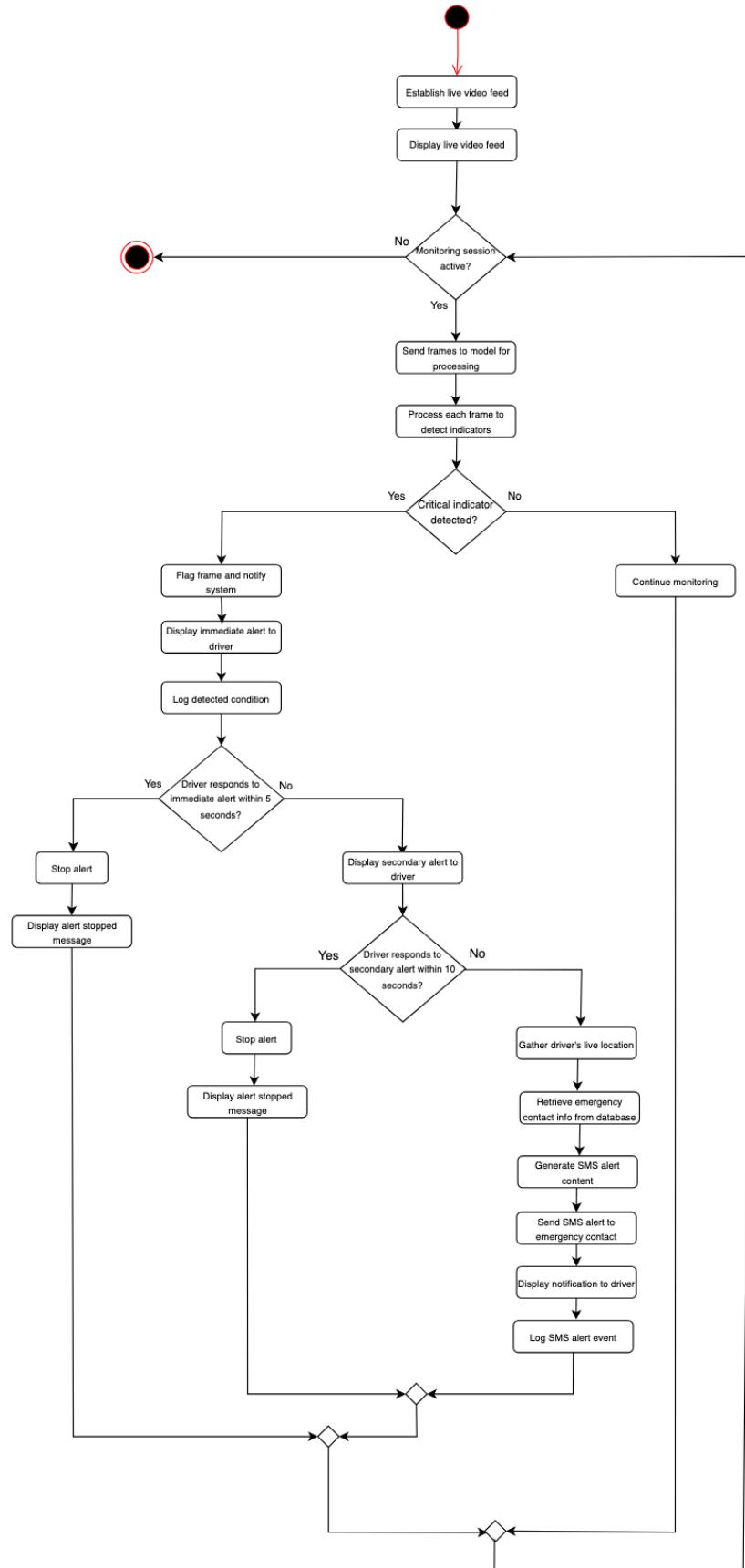
**Figure 2.9:** Manage Monitoring Session Sequence Diagram

## 2.5 Activity Diagram

This section presents a graphical representation of the activity diagram, an advanced flow chart. These flows of the diagrams are illustrated in from Figure 2.10 to Figure 2.11.



**Figure 2.10:** Monitoring Session and Camera Connection Activity Diagram



**Figure 2.11:** Detection Activity Diagram

## 2.6 Conclusion

This chapter thoroughly covered the analysis and design of the DSMS system. It started with the system's functional and non-functional requirements, and the class diagram, use case diagram, sequence diagram, and activity diagrams were presented to illustrate user-system interactions and the system's processes.

## References

- [1] Admin, “Why You Should Learn to Drive – The Benefits of Driving – The Yale Wave,” May 28, 2022. <https://campuspress.yale.edu/wave/why-you-should-learn-to-drive-the-benefits-of-driving/>.
- [2] E. Comission, “Driver Distraction,” 2015. [https://road-safety.transport.ec.europa.eu/system/files/2021-07/ersosynthesis2015-driverdistraction25\\_en.pdf](https://road-safety.transport.ec.europa.eu/system/files/2021-07/ersosynthesis2015-driverdistraction25_en.pdf).
- [3] W. H. O. WHO, “Road traffic injuries,” *www.who.int*, Jun. 2022, [Online]. Available: <https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries>.
- [4] “Distracted Driving | NHTSA,” *NHTSA*, <https://www.nhtsa.gov/risky-driving/distracted-driving>.
- [5] “Epilepsy and the Risks of Accidents,” *Epilepsy Foundation*, <https://www.epilepsy.com/stories/epilepsy-and-risks-accidents>.

## Apendix

You can view the presentation slides in pdf or word docx in the following google drive link: [Click here](#)