TED UNIVERSITY

CMPE 491

Project Analysis Report

Project Name: Drive Safe-Off

Team Members

Gökçe BEKAR

Rabia Esra ŞENDUR

Rumeysa OMAY

Oğuzhan Uğur SARISAKALOĞLU

Supervisor

Venera ADANOVA

Jury Members

Aslı GENÇTAV

Orkunt SABUNCU

Contents

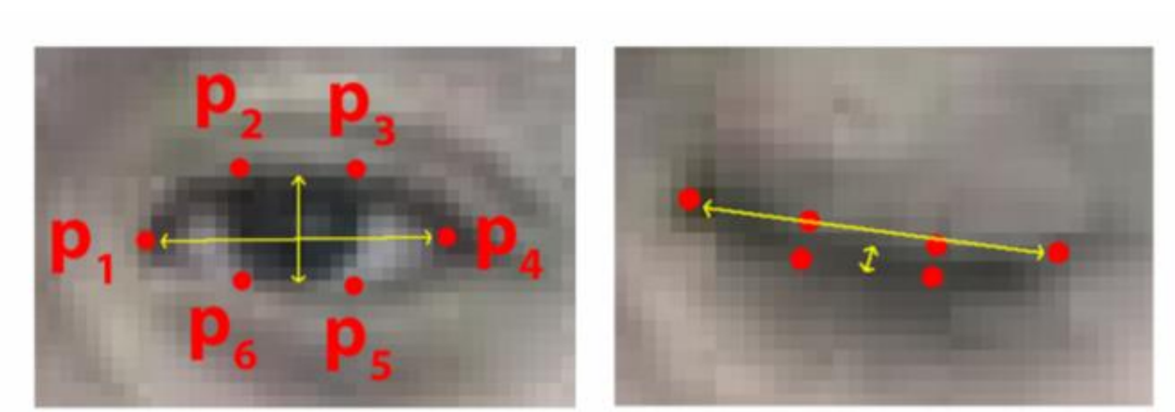
Introduction

One of the most important causes of traffic accidents, especially in the long car driving, is the fatigue of the driver. Fatigue affects the driver's consciousness, reducing her attention and perception while driving. Both are essential competencies for a safe ride. Studies show that drivers who reduce their level of consciousness due to fatigue have more accidents than other more conscious drivers. Therefore, observing the driver's fatigue allows us to predict his accident tendency.

We aim to observe and monitor the fatigue of the driver and inform her about her fatigue. With our information activity, we will try to prevent the driver from causing a traffic accident due to fatigue. In our system, we will be able to collect information about the driver's driving experience with our observations on the driver's drowsiness.

The current system(if any)

First of all, there is a program written using python's OpenCV library for computer vision that can detect the face, eye, and mouth. This program, which works using a computer camera, can detect a human face instantly. Also, it can detect the face, eye, and mouth from videos which include data sets instead of a computer camera. The current system gives much faster results when using a computer camera instead of video. Secondly, the program conducts due diligence by looking at the eye aspect ratio to measure the driver's drowsiness.



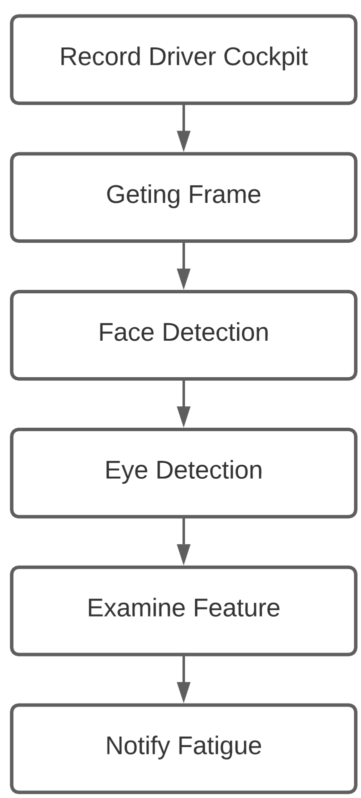
$$\text{EAR} = \frac{\|p_2 - p_6\| + \|p_3 - p_5\|}{2\|p_1 - p_4\|}$$

For every video frame, the eye landmarks are detected. The eye aspect ratio (EAR) between the height and width of the eye is computed. As the eye aspect ratio approaches zero, it is detected that the driver's drowsiness increases and a warning is sent to the driver immediately.

Proposed System

Overview

The flow of the system is initiated by the user starting with a driving session after logged in to the system. During the session, the scene that contents the driver's facial landmark will be perceived by the system frame by frame.



Functional Requirements

Lighting conditions of the environment

Frequent or tough changes in the brightness of a frame received by a camera during the driving periods have been an important challenge for many computer vision algorithms.

Eye Region

The system detects and examines the frame pieces from the driver's scene. Therefore the region must be continuously observable without obstructions.

Absolute positioning of receptor device

The camera must be located within a defined range from the driver and within a defined viewing angle.

Adaptability to Various Operator's Physical Properties

The ideal system works with a wides range of driver's demographic features, physical features, and temporary features.

Other Functional Requirements

The system should be able to process images while driving.

The system will warn when the drowsiness exceeds a certain value.

The system must be able to calculate the yawning time.

The system should be able to calculate the amount of blinking and how long it was off.

The system must be able to calculate the time for head movement the driver does other than looking at the road.

Nonfunctional Requirements

Performance / Response Time

A device that is unable to detect drowsiness and warn the driver inefficiently can have severe consequences. The response time of the device should be fast and the alert should be effective enough to make the driver sober up.

Reliability

The system should identify drowsiness correctly so that it can serve its function as a driver safety promotion device. The system defect rate shall be less than 1 failure per 1000 hours of operation.

Availability

The system should always be ready for use (7/24).

Maintainability

The system shall not be shut down for maintenance more than once in a 24-hour period.

Economical

There are existing solutions to this problem available today, but for broad implementation, the successful ones are typically too costly.

Usability

The use of the device should be as simple as possible. People who are not trained to use the system will be able to use the product.

Security

The driver's personal information shall not be accessed or reached by anyone.

Accessibility

The system shall be accessible to a wide range of driver as well as the prevalence of driver's license.

Efficiency

The system restart cycle must execute completely in less than 60 seconds. The system should deliver a notification in 1 second or less.

Pseudo Requirements

The Affordable Hardware

The software will run on Raspberry pi.

Sealed Software

The software must not disclose users' information.

In-car Equipment

The target platform is to be used in the vehicle.

Shareable Session Context

The driver can share the driver's experience in the session with others. As the internet connectivity on cars and data streaming socially is the state of the art in intelligent systems, users can be benefited from this experience. On the other hand, that is the proper way to make others feel safe.

Potential to be a black box

In-plane crashes, the reasons for the accidents can be found by looking at the black boxes. In this way, our product can be implemented as a black box on the car.

System Models

The user is the backbone actor of the system that acts according to scene record data provided by the user via camera. The supervisor is an administrator actor that interacts with the system directly with absolute authority.

The analysis object model consists of the entity, boundary, and control objects. Frame, fatigue rate, model, and time are considered as entity objects. Boundary objects display screen and button. The change in fatigue is a control object of the system.

Scenarios

Scenario name: fatigueDetected
Participating actor instances: Bill:User
Flow of the events:

1) Bill driving the car in the middle of the night.
2) He had sleeplessness after work and still under the influence of sleeplessness.
3) The symptoms on his face began to be detected by the system
4) The system confirms that his drowsiness at the critic level.
5) Bill is alerted by the system.

Scenario name: firstSetup
Participating actor instances: Joe:User
Flow of event:

1) Joe get the system install
2) Joe positioning the equipment correctly in the car
3) He turns it on the correct calibration
4) System ready to start

Scenario name: giveWarning
Participating actor instances: Mark:User, Others Mark’s Farther:contactMember
Flow of events:

1) Early in the day, Mark gets into his car sleepily and starts driving, and starts the software after clicks the button.
2) Mark starts to stretch more than once while driving and after a while, the stretch time increased.
3) The software sounds like many warnings and Mark did not hear.
4) Mark's eye closure time has increased and the system has given a warning again. Mark continues its drowsy and the system warns again.
5) As he continues to drive sleepily, the system sends a warning to the contact member Mark enters when the system was set up.
6) His father called Mark several times by phone and tried to reach him to stop his vehicle.
7) Mark, whose phone is not reached, the location information sent with the warning is shared with the nearby police station and the vehicle is stopped.

Scenario name: setup for a user
Participating actor instances: Mark
Flow of events:

1) Mark's information is saved in the system.
2) Contact member information is saved in the system for emergency access.

Use Case Model

Use case name: NotifyFatigue
Participating actors: User
Entry condition: The system activates the "Notify Fatigue" function
Flow of events:

1) The subsystem gets the frame came from video receiver at the time.
2) Face of the driver is detected on the frame.
3) Eyes are detected in the face boundaries.
4) The subsystem classify the eye states is processing and it communicating with calculate drowsiness subsystem.
5) The drowsiness is calculated and if it is not ordinary state, the system starts notify.

Exit condition: The user is notified.

Use case name: CollectHistory
Participating actors: Initiated by Supervisor, Communicates with the system
Entry condition: Supervisor accessing the system.
Flow of events:

1) The supervisor accessed the system.
2) The request is created.
3) The system replies the request.

Exit condition: The stored data of the system is exported to the Supervisor

Use case name: StartDrivingSession
Initiating actor: User
Entry condition: The user logged into the system
Flow of events:

1. The user starts the session from the interface.

Exit condition: The user ends the session.

Use case name: ExamineEyeState
Initiating actor: SubSystem
Entry condition: The eye region is extracted
Flow of events:

2. Subsystem received the detected face.
3. Subsystem classifies the state that is blinked or not.
4. Subsystem exports its assessment to CalculateDrowsiness.

Exit condition: Export the survey at the state to NotifyManager.

Use case name: EndDrivingSession
Initiating actor: User
Entry condition: The end is requested by the user via GUI
Flow of events:

1. The user finished the drive.
2. The user interacts with the button that ends the session on a GUI.

3. System summarizes the session and stores the investigating data.
4. The system represents the session report on GUI.
5. The user turns off the system via GUI or button. Otherwise, the system is closed by an internal decision.

Exit condition: System successfully ended by the user.

Use Case Model Diagrams

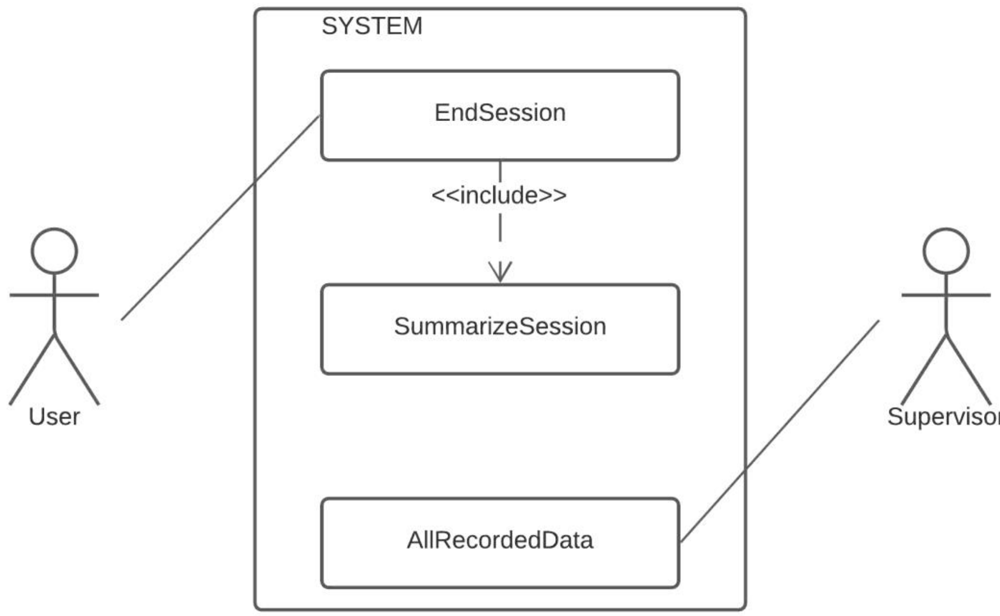


Figure 1CollectHistory use case model

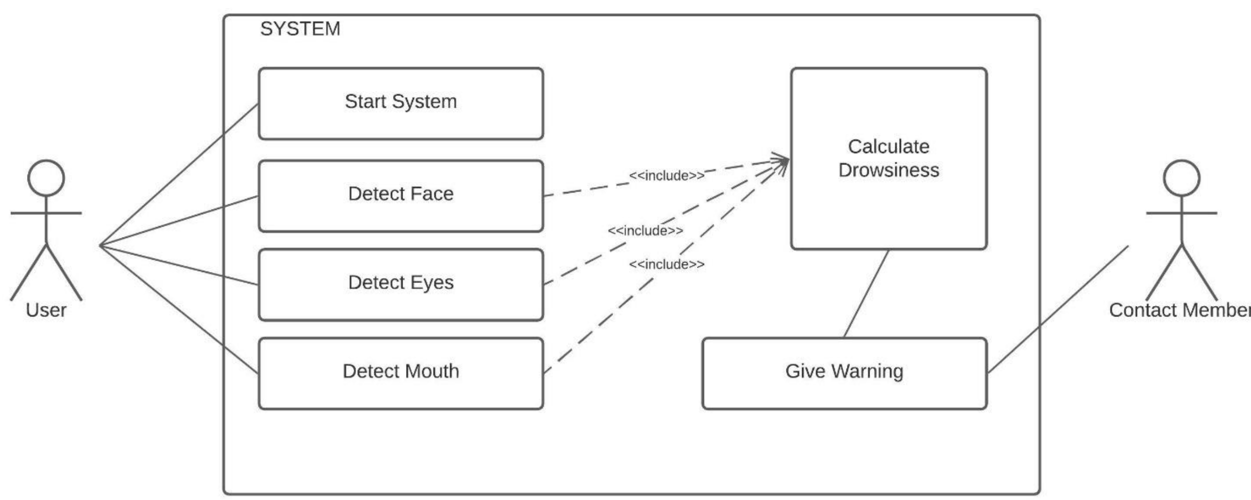
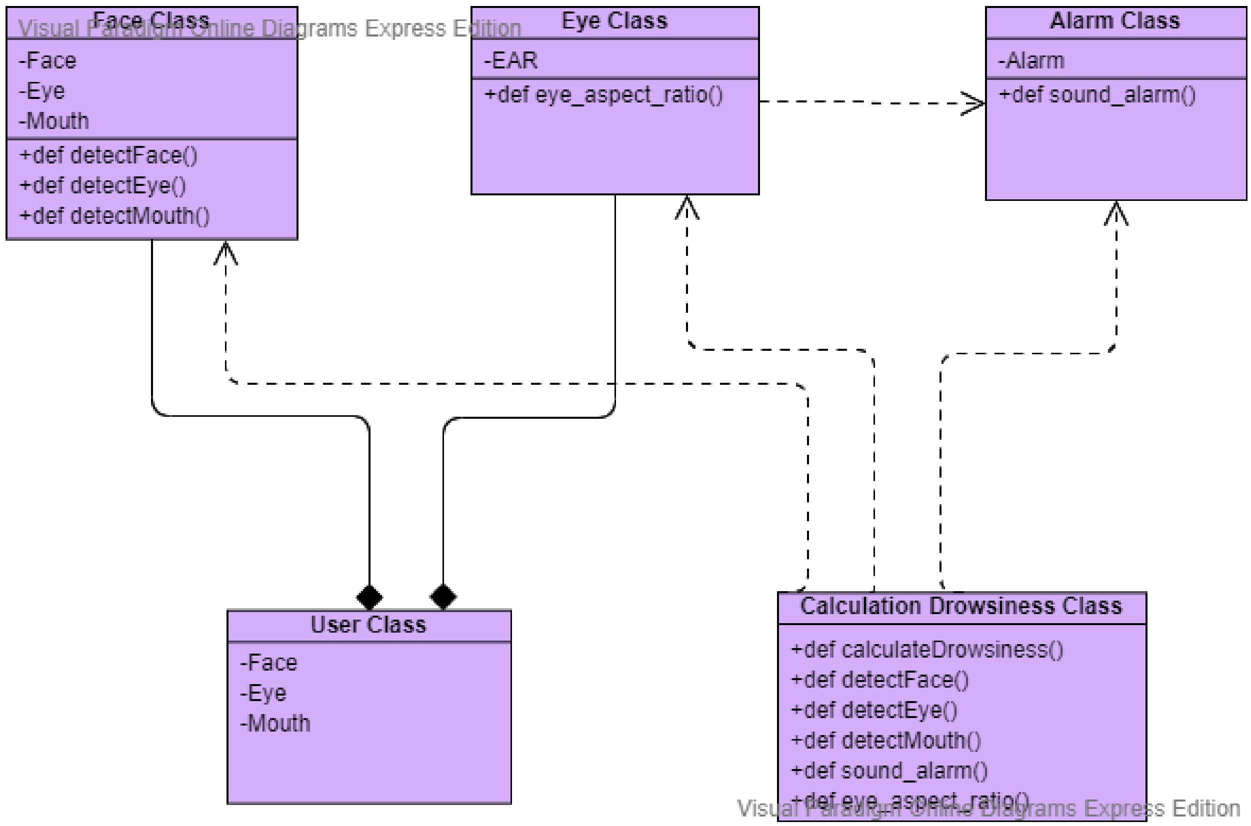


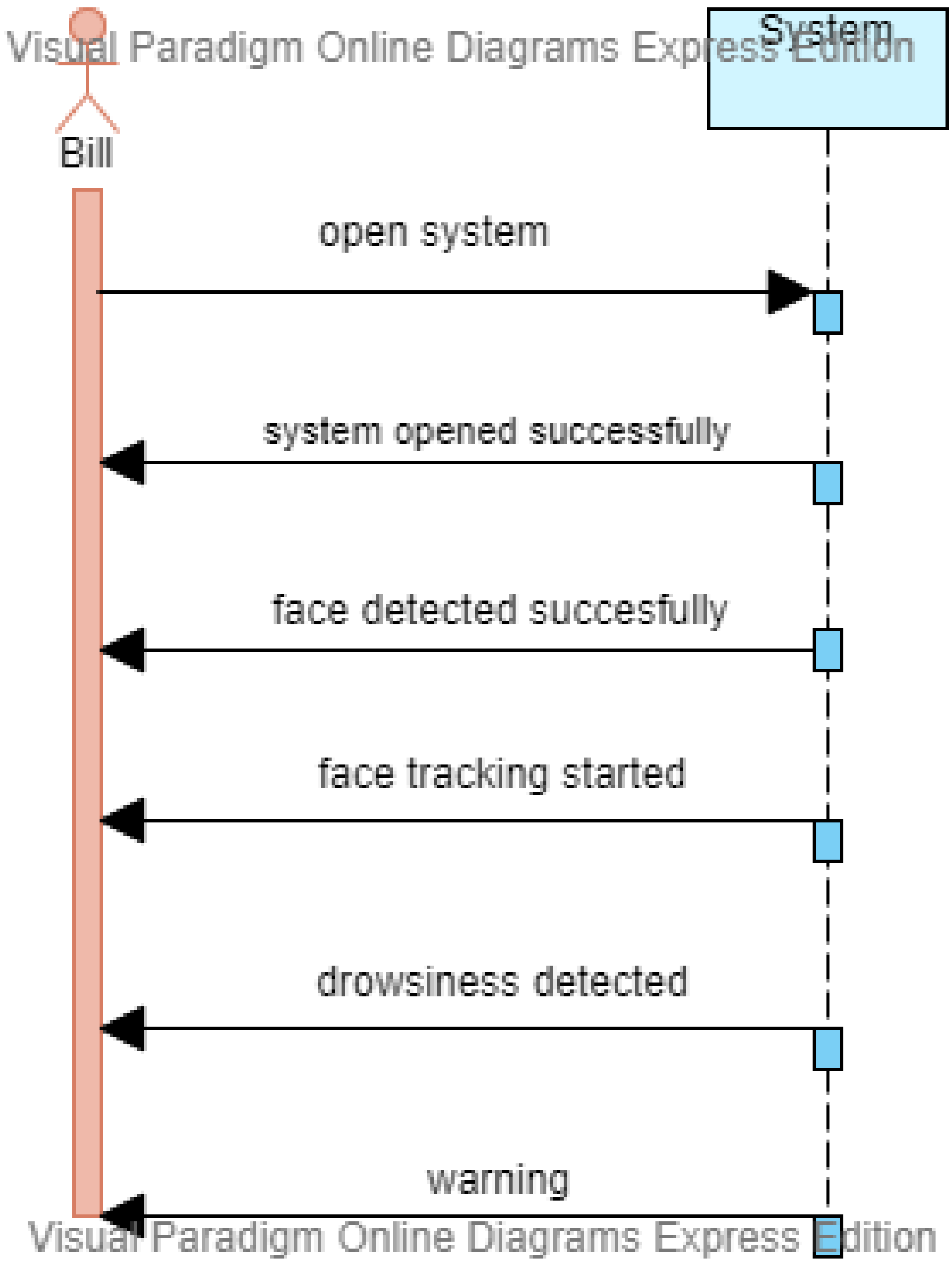
Figure 2GiveWarning use case model

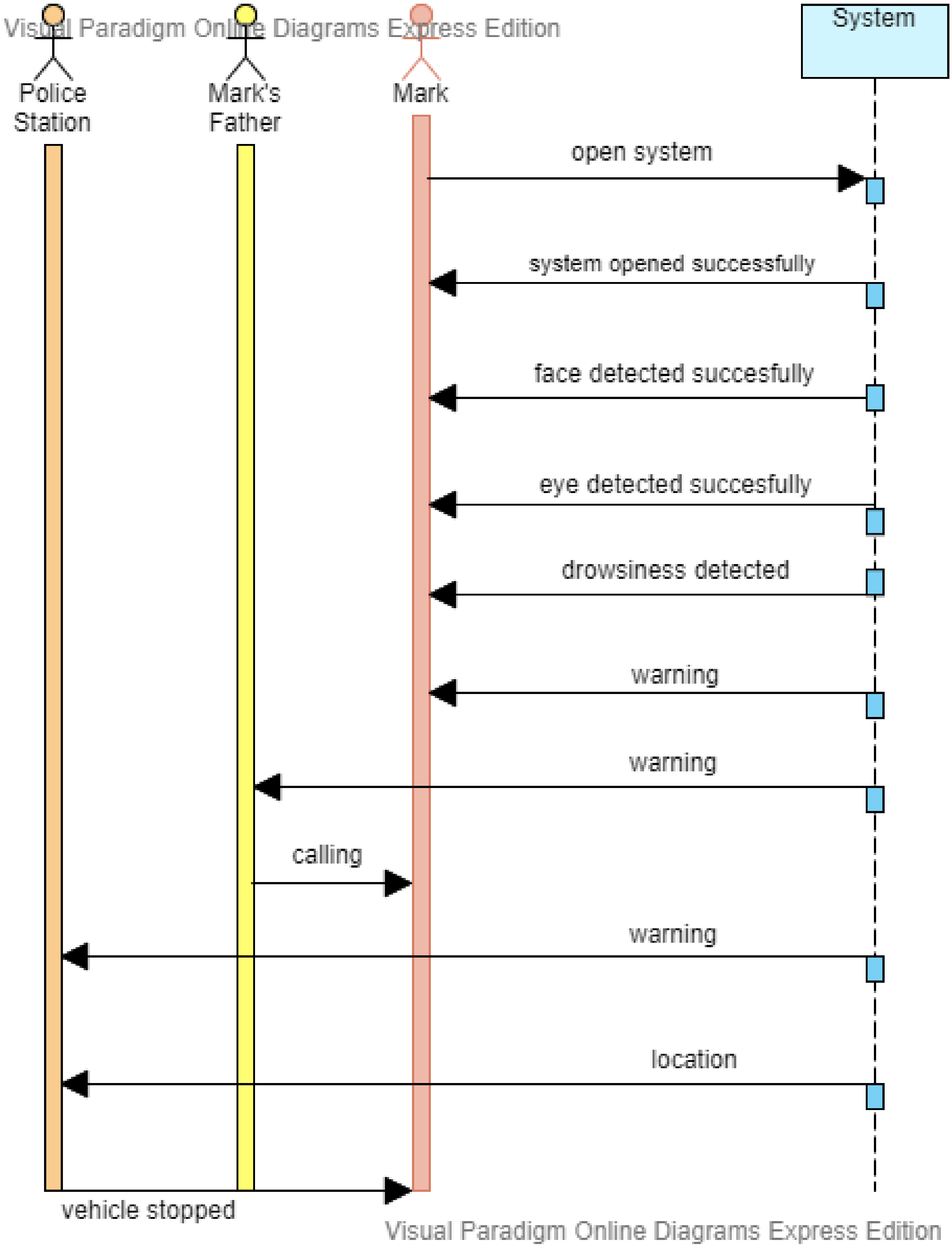
Object and Class Model



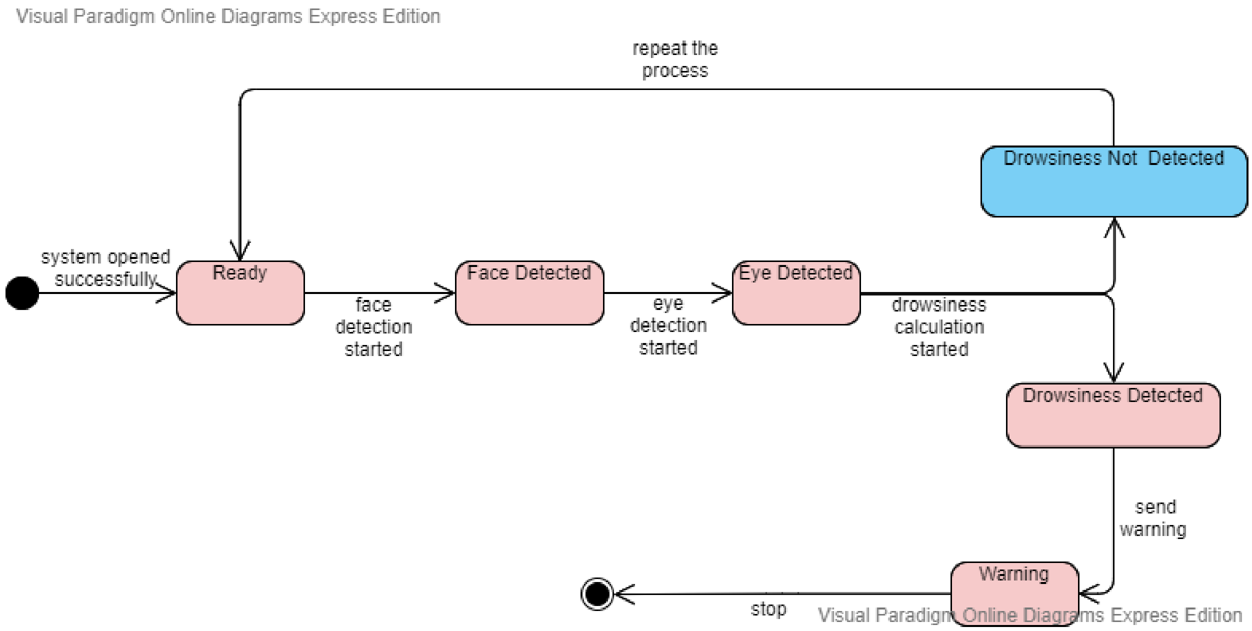
Dynamic models

Sequence Diagrams



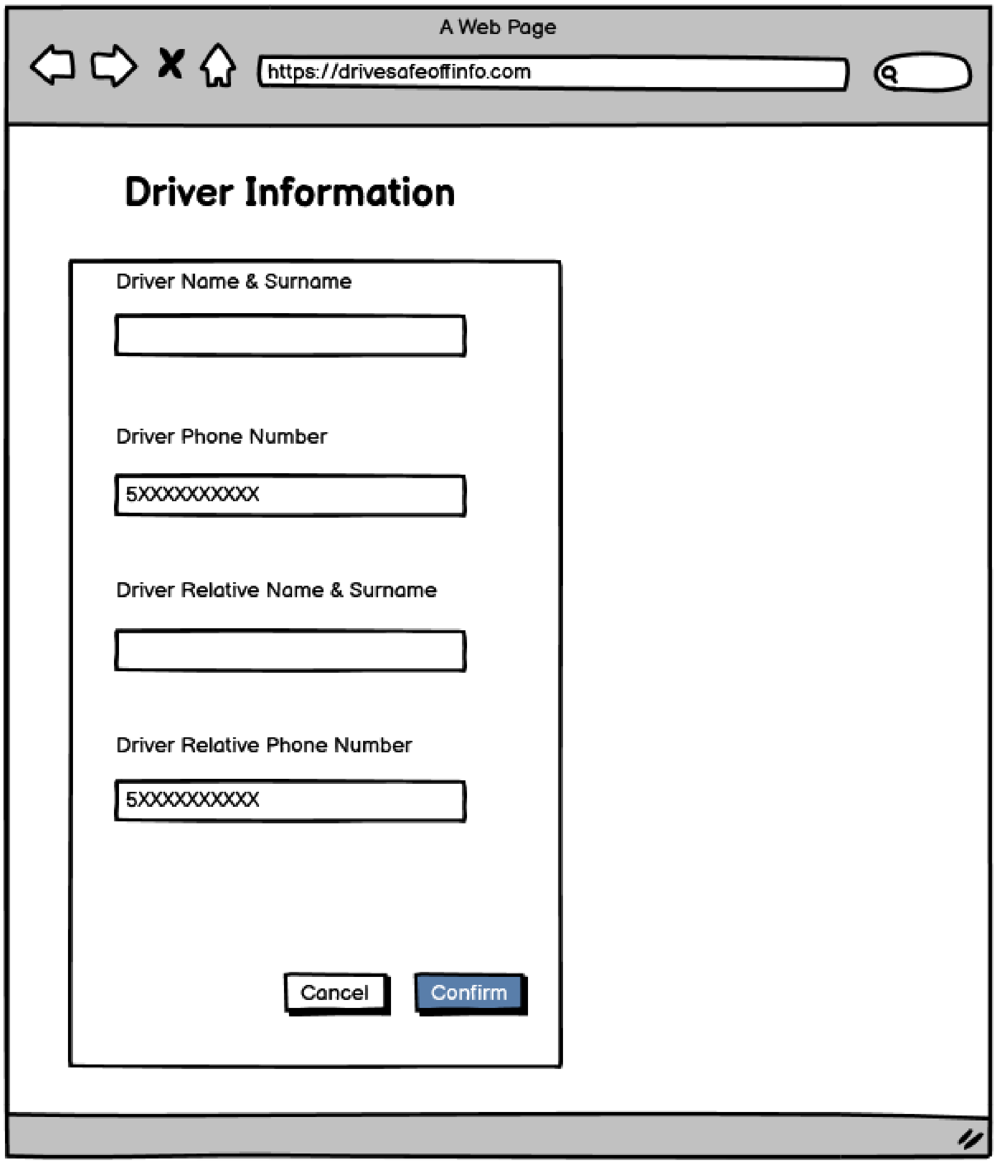

Visual Paradigm Online Diagrams Express Edition
System
Police Station
Mark's Father
Mark
open system
system opened successfully
face detected succesfully
eye detected succesfully
drowsiness detected
warning
warning
calling
warning
location
vehicle stopped
Visual Paradigm Online Diagrams Express Edition

State Machine Diagram



User interface -navigational paths and screen mock-ups

There will be no application that the user will use to run the program. Therefore, there will not be a comprehensive interface for the user to use. So, there will be an interface to send the necessary information to a relative of the user. Since the driver fills in the necessary parts in the interface, a message can be sent to her/his relatives when necessary. When drowsiness is detected in the driver, the driver's location will be sent with a warning message to the driver's relative. In this way, the risk of a possible accident will be minimized by ensuring that the driver stops her/his car. Consequently, there will be places and a button on the interface where the user can enter her/his information and run the system.



Glossary

Session: Time interval between driving start and end.

Supervisor: The person is the administrative authority of the environment of the system.

User: The person driving the car on the scene in a session

References

1. Bruegge, B., & Dutoit, A. H. (2003). Object-oriented software engineering: Using UML, patterns and Java. Upper Saddle River, NJ: Prentice Hall.
2. Soukupova, T., & Cech, J. (2016). Real-Time Eye Blink Detection using Facial Landmarks.