

**TEST PLAN FOR**

**DISTRACTED DRIVER DETECTION**

1. [INTRODUCTION 2](#_TOC_250006)
   1. [SCOPE 2](#_TOC_250005)
      1. [In Scope 2](#_TOC_250004)
      2. Out of Scope 3
   2. QUALITY OBJECTIVE 3
   3. ROLES AND RESPONSIBILITIES 3
2. [TEST METHODOLOGY 4](#_TOC_250003)
   1. [OVERVIEW 4](#_TOC_250002)
   2. [TEST LEVELS 4](#_TOC_250001)
   3. SUSPENSION CRITERIA AND RESUMPTION REQUIREMENTS ERROR! BOOKMARK NOT DEFINED.
   4. TEST COMPLETENESS 5
3. [TEST DELIVERABLES 5](#_TOC_250000)
4. RESOURCE & ENVIRONMENT NEEDS 6
   1. TESTING TOOLS 6
   2. TEST ENVIRONMENT 6
5. TERMS/ACRONYMS 6

# Introduction

Drowsiness detection system is a system which will detect fatigue of the human and it will give warning. Due to drowsiness, office employees are not able to meet the deadlines of the projects allotted to them. Due to the drowsiness in offices the cost of company increases due to the delay in their projects. In this project the system will continuously monitor the eye of the human using a camera. By applying a perfect algorithm, we can detect the symptoms of drowsiness in peoples, and we will give a warning to avoid the person to get into sleep.In this application warning will only be deactivated manually. For this a deactivation dialog box will appear which will have some basic operation for employee or we can add a device which will give an alert signal and that signal will stop only when the employee manually deactivates the signal. Moreover, if the person is drowsy, he/she may give the wrong answer in the dialog box. We can judge this by plotting a graph in time domain. If all the three input variables show a possibility of fatigue at one moment, then a warning is given in form of text or sound. This will directly give an indication of drowsiness.some of objectives are -

. Drowsiness detection system is an application that many organizations can use to analyze the employees. It may be used to analyze if their employees are working or suffering from any fatigue.

. They can calculate the various data in a better way to get the detailed data about the employees working. It will tell us about the employees working time and rest time by calculations.

**SCOPE**

* 1. In Scope

Data Collection and Preprocessing:

. Gather a diverse and extensive dataset of images and videos capturing various instances of distracted driving.

. Preprocess the data to enhance its quality and standardize the format for model training.

CNN Model Development:

. Develop a Convolutional Neural Network (CNN) architecture capable of analyzing images and videos in real-time.

. Train the CNN to recognize signs of distracted driving, including activities like texting, talking on the phone, eating, and more.

Machine Learning Algorithm:

. Implement machine learning algorithms to fine-tune the CNN model's ability to identify different forms of distraction.

**.** Optimize the model's accuracy and efficiency through iterative training and testing.

Real-Time Image Analysis:

. Create a system that can process and analyze images and videos in real-time, eitherfromin-vehicle cameras or external sources.

. Ensure the system can identify and classify distracted driving behaviors promptly and accurately.

* 1. Out Of Scope

Autonomous Driving:

* + This project does not involve the development of autonomous driving technology, self-driving vehicles, or full automation of vehicles. Its primary focus is on distracted driver detection within the context of human-operated vehicles.

Hardware Development:

* + The project does not encompass the design or development of specialized hardware, such as vehicle-mounted cameras, sensors, or onboard computer systems. It relies on existing hardware for data input.

Regulatory Changes:

* + While the project may address legal and ethical considerations, it does not aim to influence or initiate regulatory changes in traffic laws or standards. Compliance with existing regulations is assumed.

Distracted Pedestrian Detection:

* + The project's primary focus is on identifying distracted drivers. Detection of distracted pedestrians or other road users is not within its scope.
  1. **Quality objective**

Here make a mention of the overall objevtie that you plan to achive withou your testing

Some objectives of your testing project could be

* Ensure the system exhibits a high level of reliability, minimizing false positives and false negatives in distraction detection.
* Develop a system that can analyze and respond to potential distracted driving events in real-time, with a minimal latency of no more than 0.5 seconds.
* Consider accessibility standards to make the system inclusive and usable for a wide range of users, including individuals with disabilities.
  1. **Roles And Responsibilities**

Detail description of the Roles and responsibilities of different team members like

* **QA Analyst** : Ashu Verma

The Quality Assurance (QA) Analyst conducted testing on software, websites, and other technical products to identify and resolve bugs, defects, and other potential issue.

* **Test Manager :** Ms. Akansha Tyagi

Managed all test processes, including test plans, resources, costs, timescales, test deliverables and traceability.

* **Configuration Manager :** Mrs. Neha Shukla
* **Developers :** Daksh Kumar, Ashu Verma , Chandan Kr. Gupta Developed the model and trained it.
* **Installation Team :** Daksh Kumar, Ashu Verma , Chandan Kr. Gupta , Akansha Tyagi , Neha Shukla Responsible for smooth execution of the program

**TEST METHODOLOGY**

**2.1 Overview**

The decision to adopt a Waterfall methodology for a project is typically based on specific project requirements, constraints, and organizational factors. Here are some common reasons for choosing the Waterfall methodology:

* Well-Defined Requirements: When the project has clearly defined and stable requirements that are unlikely to change significantly throughout the project's lifecycle. Waterfall is suitable when you can gather and document all the requirements up front.
* Low Uncertainty: If there is a high level of confidence in the project scope and objectives, and the technology and processes to be used are well-understood, Waterfall can be a good choice. It is less adaptable to uncertainty and change.
* Regulatory Compliance: In cases where the project needs to adhere to strict regulatory or compliance standards, Waterfall provides a structured and documented approach that can help meet these requirements.
* Large-Scale and Complex Projects: Waterfall can be beneficial for large-scale, complex projects where a comprehensive and detailed project plan is essential for successful execution.

**2.2 Test Levels**

Testing a Web Application Firewall (WAF) typically involves multiple test levels to ensure comprehensive coverage of its security features and effectiveness. These test levels can be organized as follows:

#### Unit Testing:

* + **Rule Validation:** Verify that individual security rules within the WAF are correctly configured and accurately detect or block specific types of attacks.
  + **Logging and Alerting:** Test that the WAF generates appropriate logs and alerts for specific rule violations.

#### Integration Testing:

* + **Rule Interaction:** Assess how different security rules interact when multiple rules are applied to the same request or response. Ensure they do not conflict or produce unintended outcomes.
  + **Communication with Other Security Components:** Test the WAF's ability to integrate with other security components in your infrastructure, such as intrusion detection systems (IDS) or load balancers.

#### System Testing:

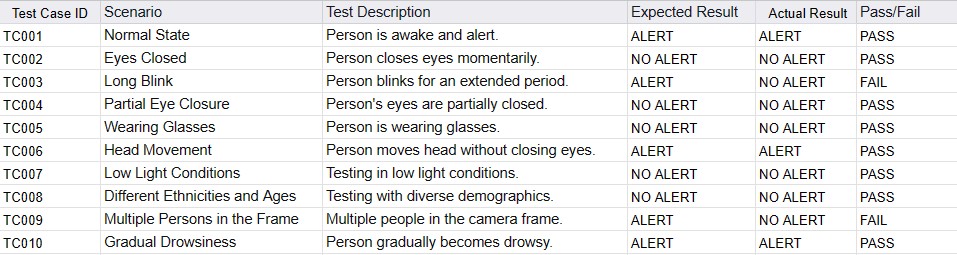
* + **Rule Coverage:** Validate that the WAF provides comprehensive coverage for known vulnerabilities and attacks, including SQL injection, cross-site scripting (XSS), cross-site request forgery (CSRF), and other common web application threats.
  + **Custom Rule Testing**: Ensure that any custom rules configured to protect application-specific vulnerabilities are working as intended.

**2.3 Test Completeness**

Few criteria to check Test Completeness are:

* 100% test coverage
* All open bugs are fixed or will be fixed in next release

**TEST DELIVERABLES**



**BOUNDARY VALUES**

Minimum value= 0.0

Maximum value= 0.5

Nominal value=(min+max)/2=0.25

Minimum+0.1=0.1

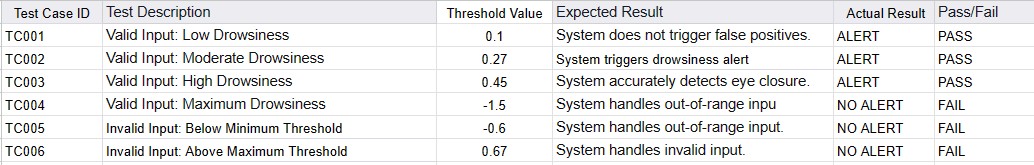
Maximum-0.1=0.4

A screenshot of a computer

Description automatically generated

**EQUIVALENCE TABLE**





**RESOURCE AND ENVIRONMENT NEEDS**

**Testing Tools**

No Testing Tool is required. Manual Testing is done

**Test Environment**

It mentions the minimum **hardware** requirements that will be used to test the Application. The following **software’s** are required in addition to client-specific software.

* Windows Operating System
* VS Code
* Camera and Microphone

**TERMS/ ACRONYM**

Make a mention of any terms or acronyms used in the project

|  |  |
| --- | --- |
| **TERM/ACRONYM** | **DEFINITION** |
| cv2 | Open CV |
| py | Python |
| imu | Imutils library |

**Made By -:**

Daksh Kumar (2000290120060)

Ashu Verma(2000290120045)

Chandan Kr. Gupta(2000290120058)

**Test Manager -:**

Prof. Akansha Tyagi

**Project Coordinator -:**

Prof. Anshula Gupta

**Submitted To -:**

Prof. Neha Shukla