

# Low Level Design

Food Recommendation System

Written By	Bhavya M Shah
Document Version	0.1

Last Revised Date	16 – Dec-2022
-------------------	---------------

## Document Control

### Change Record:

Version	Date	Author	Comments
0.1	16 – Dec-2022	Bhavya Shah	Introduction & Architecture defined
0.1	16 – Dec-2022	Bhavya Shah	Architecture & Architecture Description appended and updated
0.1	16 – Dec-2022	Bhavya Shah	Unit Test Cases defined and appended

### Reviews:

Version	Date	Reviewer	Comments
0.1	16 – Dec-2022	Bhavya Shah	Document Content , Version Control and Unit Test Cases to be added

### Approval Status:

Version	Review Date	Reviewed By	Approved By	Comment

## Contents

1. Introduction	1
1.1. What is Low-Level design document?	1
1.2. Scope	1
2. Architecture	2
3. Architecture Description	3
3.1. Data Description	3
3.2. ML Project	3
3.3. Data Cleaning	3
3.4. EDA (Exploratory Data Analysis)	3
3.5. Export Data from Database	3
3.6. Data Pre-processing	3
3.7. Data pre-processing	3
3.10. Creation of train test and validation sets	4
3.11. Model Creation	4
3.12. Hyperparameter tuning	4
3.13. Model deployment	4
4. Unit Test Cases	5

## 1. Introduction

### 1.1. What is Low-Level design document?

The goal of LLD or a low-level design document (LLDD) is to give the internal logical design of the actual program code for Food Recommendation System. LLD describes the class diagrams with the methods and relations between classes and program specs. It describes the modules so that the programmer can directly code the program from the document.

### 1.2. Scope

Low-level design (LLD) is a component-level design process that follows a step-by-step refinement process. This process can be used for designing data structures, required software architecture, source code and ultimately, performance algorithms. Overall, the data organization may be defined during requirement analysis and then refined during data design work

## 2. Architecture

### Training the YOLOv5 Object Detector on a Custom Dataset



### 3. Architecture Description

#### 3.1. Data Collection

Load,summary statistics,information about dataset. Drowsiness Detection dataset is the biggest publicly available dataset.

#### 3.2. CV Project

In order to generate an accurate prediction of driver and drowsiness we need some more data.

#### 3.3. Data Pre-processing

We have a pre-trained model in YOLO v5 just need to train our images using the pre-trained model with an accurate number of epochs eg 999 and more.

#### 3.4. Creation of train test and validation sets

Divide the data into train and test split

#### 3.5. Model Creation

Divide the data into train and test split

## Test Case

Test-Case Description	Pre-Requisites	Executed Result
Verify whether the Application URL is accessible to the user	1. Application URL should be defined	Application URL should be accessible to the user
Verify whether the Application loads completely for the user when the URL is accessed	1. Application URL is accessible	The Application should load completely for the user when the URL is accessed
Verify whether the user is able to see the camera for the detection	Application is accessible	User should be able to see the camera and for the detection of drowsiness.
Verify whether the model is able to detect the face of the Driver and the drowsiness	Driver and Drowsiness	Yes the model through the camera is able to access and identify the Driver and the face of the driver and Drowsiness.
Verify whether KPIs modify as per the user inputs for the user's health	Application is accessible	KPIs should modify as per the user inputs for the user's
Verify whether the KPIs indicate details of the suggested sales	Application is accessible	The KPIs should indicate details of the suggested sales





