

# Low Level Design

## Healthcare Analytics on Heart Disease Data

Written By	Bhushan Raut Sana Tadv
Document Version	1.0
Last Revised Date	10/10/2021

## Document Control

### Change Record:

Version	Date	Author
1.0	10/10/2021	Bhushan Raut Sana Tadv

### Review:

Version	Date	Author

### Approval status:

Version	Date	Author

## Content:

1. Introduction	
1.1. What is Low Level Design Document?	4
1.2. Scope	4
2. Architecture	5
3. Architecture Description	5
3.1. Data Description	5
3.2. Import Data	5
3.3. Data Cleaning	5
3.4. Exploratory Data Analysis	6
3.5. Data Pre-processing	6
3.6. Model Building	6
3.7. Model Dump	6
3.8. Data From User	6
3.9. Data Validation	6
3.10. Model Call for Specific Input	6
3.11. User Interface	6
3.12 Deployment	8
3.13 Dashboard	9
4. Technology Stack	9

## Introduction:

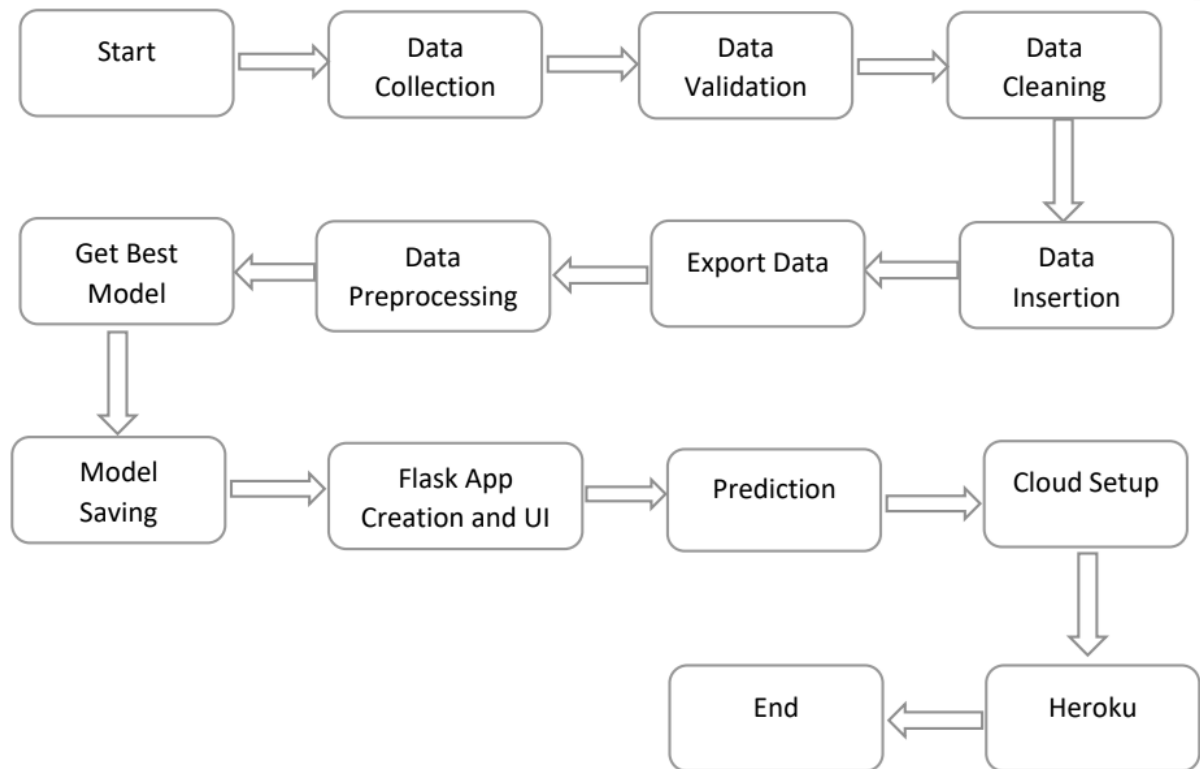
### 1.1 What is Low-Level design document?

The goal of the LDD or Low-level design document (LLDD) is to give the internal logic design of the actual program code for the House Price Prediction dashboard. LDD describes the class diagrams with the methods and relations between classes and programs 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. The 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



## 3 Architecture Description

### 3.1 Data Description

The proposed system will add some more parameters significant to heart attack with their weight, age and the priority levels are by consulting expertise doctors and the medical experts. The heart attack prediction system designed to help the identify different risk levels of heart attack like normal, low or high and also giving the prescription details with related to the predicted result

### 3.2 Import Data

Data Import - The data in a stored database is imported as a CSV file to be used for Data Pre-processing and Model Training.

### 3.3 Data Cleaning

There are no null values in the data and all the variables are categorical, some of the observations meaningless and they are converted into meaningful observation. Example: “?” is converted into letter “m” (Missing)

### 3.4 Exploratory Data Analysis

EDA is done in such way that every independent variable in the dataset is visualized by multiple bar plot with respect to dependent variable having classes of Heart diseases factors

### 3.5 Data Pre-processing

Data pre-processing steps are converting categorical variables into numerical variables using label encoding method and train and test split of the data etc.

### 3.6 Model Building

After Data pre-processing split the data train and test (Simple Random Sampling) and implemented different Classification Machine Learning Algorithm. Random Forest model gives the better accuracy.

### 3.7 Model Dump

After comparing all accuracies and finding the best model for the dataset I have created a model and dumped the model in a pickle file format with the help of pickle module.

### 3.8 Data from User

Here the user will have to enter all the features values in correct order and have to submit it to the model with the help of UI interface. The data will be fed to the model which will predict whether the feature set represents the required traits.

### 3.9 Data Validation

Here Data Validation will be done, which has given by the user.

### 3.10 Model Call for specific input

Based on the user Input, the data will be processed in the backend with the variable format, then it will be converted into a NumPy array which will be fed to ML model. The loading of the pickle file will be done and then the model will predict whether the inputs were meeting the required traits by sending the result to our html page.

### 3.11 User Interface

In Frontend creation, we have made a user interactive page where users can enter their input values to application in their frontend page. a web page which is having attractive

overview with CSS. This HTML user input data is transferred in variable format to the backend and same is in a decoupled format.

## Input page:

### Heart Disease Predictor

Accuracy Level 86%

• Age	<input type="text" value="Your age.."/>
• Sex	<input type="text" value="---select option---"/>
• Chest Pain Type	<input type="text" value="---select option---"/>
• Resting Blood Pressure	<input type="text" value="A number in range [94-200] mmHg"/>
• Serum Cholesterol	<input type="text" value="A number in range [120-504] mg/dl"/>
• Fasting Blood Sugar	<input type="text" value="---select option---"/>
• Resting ECG Results	<input type="text" value="---select option---"/>
• Max Heart Rate	<input type="text" value="A number in range [71-202] bpm"/>
• Exercise-induced Angina	<input type="text" value="---select option---"/>
• ST depression	<input type="text" value="ST depression, typically in [0-8.2]"/>
• slope of the peak exercise ST segment	<input type="text" value="---select option---"/>
• Number of Major vessels	<input type="text" value="Typically in [0-4]"/>
• Thalassemia	<input type="text" value="---select option---"/>

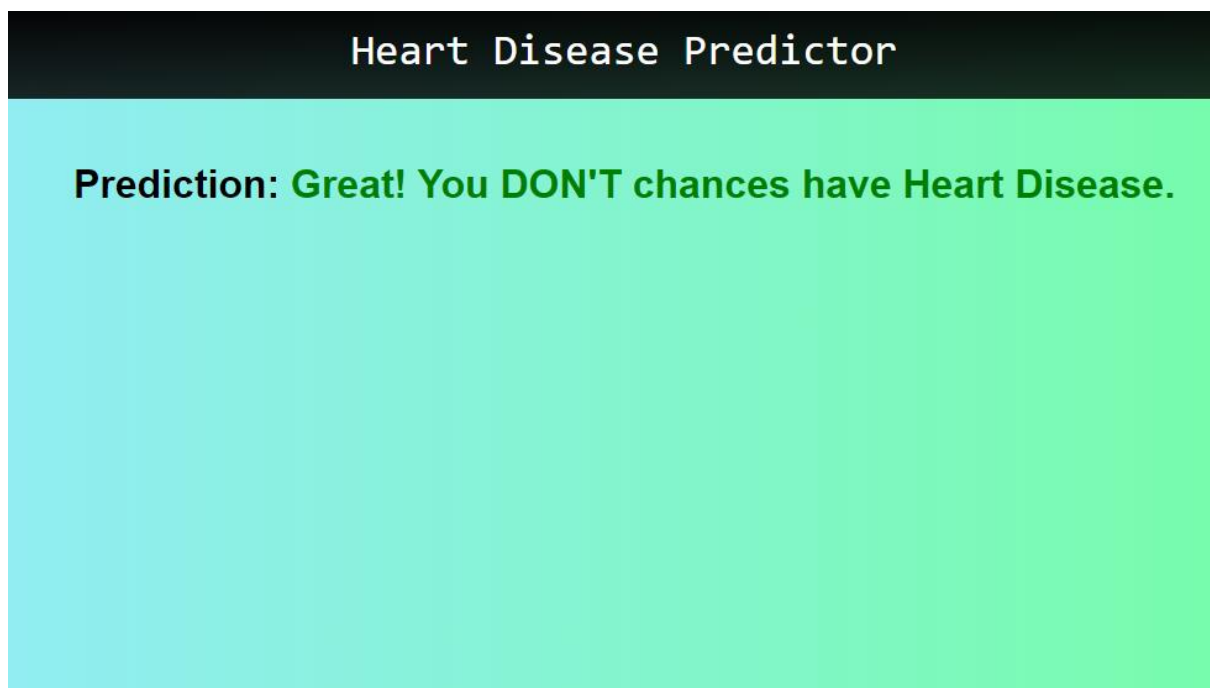
Predict

## Output page (If having heart disease):

### Heart Disease Predictor

**Prediction: Oops! You have Chances of Heart Disease.**

**Output page (If not having heart disease):**



### 3.12 Deployment

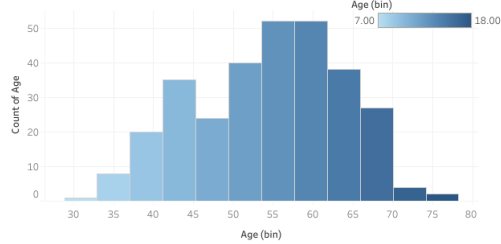
The deployment of the model with the help of Heroku cloud platform



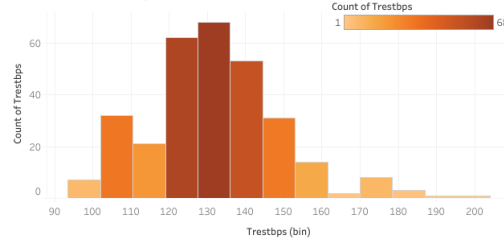
## 3.12 Dashboard

Healthcare Analysis on Heart Disease

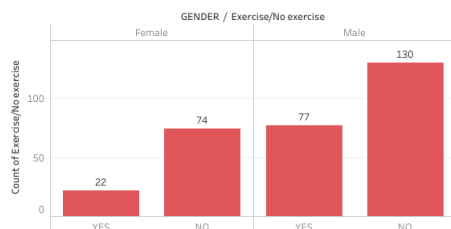
AGE DISTRIBUTION



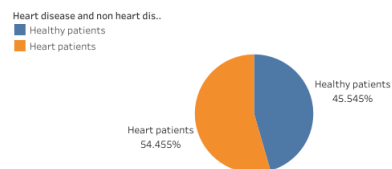
HEART RATE RANGE



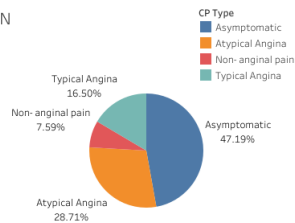
EXERCISE ANALYSIS



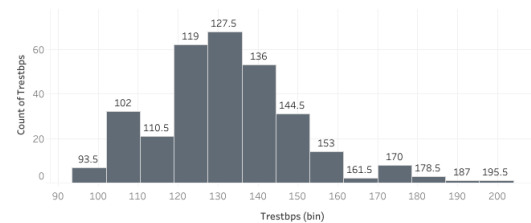
Heart Patients VS non heart disease patients



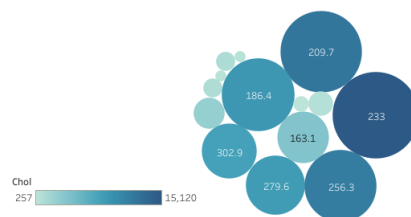
CHEST PAIN



BLOOD PRESSURE RATE



CHOLESTROL LEVEL



## 4. Technology Stack

Front End	HTML/CSS
Back End	Flask, Pandas, NumPy, Sci-kit learn, etc
Deployment	Heroku