Low Level Design (LLD)

Thyroid detection

|  |  |
| --- | --- |
| Written By | Rohan Negi |
| Document Version | Initial LLD -V1.0 |
| Last Revised Date | 12-03-2022 |

**Document Control**

**Change Record:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Comments** |
| **0.1** | **12-03-2022** | **Rohan** | **First version of Document finished.** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Reviews:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Reviewer** | **Comments** |
| **0.1** | **13-01-2022** | **Rohan** | **Document content updated.** |

**Approval Status :**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version** | **Review Date** | **Reviewed By** | **Approved By** | **Comments** |
|  |  |  |  |  |

# Contents

1.Introduction.....................................................................................................................4

1.1 What is Low-Level design document........................................................................4

1.2 Scope..........................................................................................................................4

2. Architecture...................................................................................................................5

3. Architecture Description...............................................................................................6

3.1 Data Description..........................................................................................................6

3.2 Data Pre-processing ....................................................................................................6

3.3 Model Building ...........................................................................................................6

3.4 Deployment.................................................................................................................6

4. Unit Test Cases ............................................................................................................7.

1. Introduction

1.1. What is a 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 store sales prediction. 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

2. Architecture Description

2.1. Data Description

Given is the variable name, variable type, the measurement unit, and a brief description. The concrete compressive strength is the regression problem. The order of this listing corresponds to the order of numerals along the rows of the database.

|  |  |  |
| --- | --- | --- |
| Name | Data Type | Measurement |
| Age | Float | Age of the person |
| Sex | Boolean | Sex of the person |
| on\_thyroxine | Boolean | Person is on thyroxine or not |
| on\_antithyroid\_medication | Boolean | Person is on anti thyroid medication or not |
| L131 treatment | Boolean | The person had done l131 treatment or not |
| Query hypothyroid | Boolean | Has query on hypothroid or not |
| Goitre | Boolean | Goitre present or not |
| Psych | Boolean | Has psychological disfunction or not |
| Tsh measured | Boolean | Is tsh measured or not |
| Tsh | float | Value or tsh |
| T3 measured | Boolean | Is t3 measured or not |
| T3 | Float | Value or t3 |

**2.2 Data Gathering**

Data source: <https://archive.ics.uci.edu/ml/datasets/thyroid+disease>

Train and Test data are stored in .csv format.

**2.3 Data Preprocessing**

In data preprocessing all the processes required before sending the data for model building are performed. Filling null values ,removing out liers.

**2.4 Feature Engineering**

After preprocessing it was found that some of the attributes are not important to the item sales for the particular outlet. So those attributes are removed. Even one hot encoding is also performed to convert the categorical features into numerical features.

**2.5.Parameter Tuning**

Parameters are tuned using Randomized searchCV. Four algorithms are used in this problem, Logistic RegressionRandom Forest, and XGBoost .The parameters of all these 4 algorithms are tuned and passed into the model.

**2.6 Model Building**

After doing all kinds of preprocessing operations mentioned above and performing scaling and hyperparameter tuning, the data set is passed into all four models, Logistic Regression, Random Forest, and XGBoost . Xgboost performs best

**2.7 Deployment**

The cloud environment was set up and the project was deployed from GitHub into the google cloud platform.

App link- https://thyroid-detection-in.herokuapp.com/

App demo link - https://youtu.be/ujZX50IXOxo

3. Unit Test Cases

|  |  |  |
| --- | --- | --- |
| Test Case Description | Pre-Requisite | Expected 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  2. Application is  deployed | The Application should load  completely for the user when the  URL is accessed |
| Verify whether user is able to see input  fields | 1. Application is  accessible | User should be able to see input  fields |
| Verify whether user gets Submit  button to submit the inputs | 1. Application is  accessible | User should get Submit button to  submit the inputs |
| Verify whether the predicted  results are in accordance to the  selections user made | 1. Application is  accessible | The predicted results should  be in accordance to the selections  user made |