Low Level Design (LLD)

Mushroom Classification

|  |  |
| --- | --- |
| Written By | Varun Ingle, Abhishek Doifode |
| Document Version | Initial LLD -V1.0 |
| Last Revised Date |  |

**Document Control**

**Change Record:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Comments** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Reviews:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Reviewer** | **Comments** |
|  |  |  |  |

**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

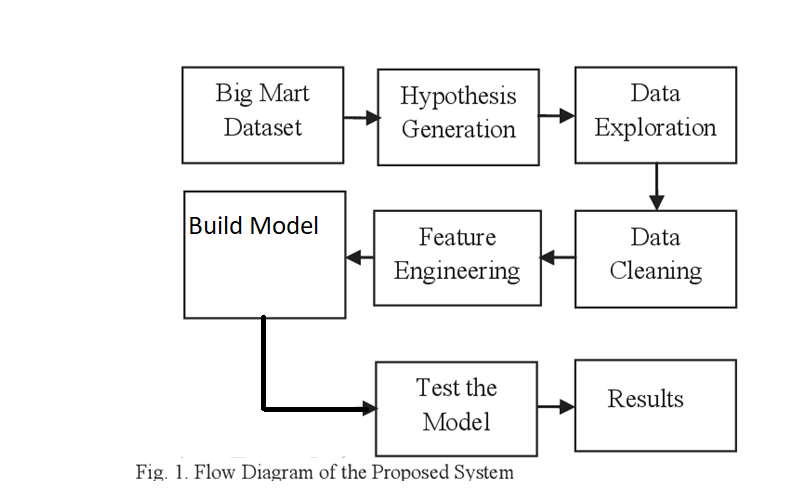
1.1. What is Low-Level design document?

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



3. Architecture Description

3.1. Data Description

This dataset includes descriptions of hypothetical samples corresponding to 23 species of gilled mushrooms in the Agaricus and Lepiota Family Mushroom drawn from The Audubon Society Field Guide to North American Mushrooms (1981). Each species is identified as definitely edible, definitely poisonous, or of unknown edibility and not recommended. This latter class was combined with the poisonous one. The Guide clearly states that there is no simple rule for determining the edibility of a mushroom; no rule like "leaflets three, let it be'' for Poisonous Oak and Ivy.

3.2. Data Pre-processing

Data Pre-processing steps we could use are Null value handling, stop words removal, punctuation removal, , Imbalanced data set handling, Handling columns with standard deviation zero or below a threshold, etc.

3.3. Model Building

We use various Ml algorithms for training the model. We selected the best algorithm by comparing scores of them.

3.4. Deployment

We will be deploying the model to Heroku.

4. 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 recommended  results are in accordance to the  selections user made | 1. Application is  accessible | The recommended results should  be in accordance to the selections  user made |