# High Level Design (HLD)

**MUSHROOM CLASSIFICATION**

***PARAS DAHIYA***

**ABSTRACT**

Mushrooms can be found extensively in a variety of natural environments and visual identification of mushroom species is well established. Some mushrooms are known because of their nutritional and therapeutical properties. Some species are known all over the world because of their toxicity that causes fatal accidents every year mainly due to misidentification. Some of the edible mushrooms are Ganoderma spp, Cantharellus spp, Agaricus spp, Pleurotus spp, Russula spp, Auricularia spp and Termitomyces spp; but the ornamentals are the beautifully ringed Microporous spp. Amanita spp, Lepiota cristata, Lepiota brunneoincarnata and Inocybe asterospora, C o p r i n u s s p p are among the most important species responsible for mushroom poisoning. Morphological and chemical analyses for mushrooms are occasionally required in forensic science practice. In this work, we will try to predict which mushroom is poisonous & which is edible.

**CONTENTS**

Title Page i

Abstract ii

Content iii

|  |  |  |
| --- | --- | --- |
|  | Introduction | 1 |
|  | Why this HlD Documentation? | 1 |
| 1 | Description | 1 |
| 1.1 | Problem Perspective | 1 |
| 1.2 | Problem Statement | 1 |
| 1.3 | Proposed Solution | 1 |
| 1.4 | Technical Requirements | 1 |
| 1.5 | Tools Used | 2 |
| 1.6 | Data Requirements | 2 |
| 1.7 | Constraints | 2 |
| 1.8 | Assumptions | 2 |
| 2 | Design Flow | 3 |
| 2.1 | Modelling Process | 3 |
| 2.2 | Deployment Process | 3 |
| 2.3 | Logging | 3 |
| 2.4 | Error Handling | 3 |
| 3 | Performance Evaluation | 4 |
| 3.1 | Reusablity | 4 |
| 3.2 | Application Compatiblity | 4 |
| 3.3 | Resourse Utilization | 4 |
| 3.4 | Deployment |  |
| Conclusion |  |  |
|  |  |  |

# INTRODUCTION

**Why this High-Level Design Document?**

The main purpose of this HLD documentation is to feature the required details

of the project and supply the outline of the machine learning model and also the

written code. This additionally provides a careful description on how the

complete project has been designed end-to-end.

# 1 Description

## Problem Perspective

Mushroom classifier is a machine learning model which helps us to predict wheter the mushroom is edible or poisionous.

## Problem Statement

The Audubon Society Field Guide to North American Mushrooms contains descriptions

of hypothetical samples corresponding to 23 species of gilled mushrooms in the Agaricus and Lepiota Family Mushroom (1981). Each species is labelled as either definitely edible, definitely poisonous, or maybe edible but not recommended. This last category was merged with the toxic category. The Guide asserts unequivocally that there is no simple rule for judging a mushroom's edibility, such as "leaflets three, leave it be" for Poisonous Oak and Ivy.

The main goal is to predict which mushroom is poisonous & which is edible.

## Proposed Solution

The solution proposed to take dataset from the client and process all the provided data to meet the requirements of the machine learning model and finally displaying the output csv file.

# 1.4 Technical Requirements

There are no hardware requirements required for using this application, the user must have an interactive device which has access to the internet and must have the basic understanding of providing the input. And for the backend part the server must run all the software that is required for the processing the provided data and to display the results.

## 1.5 Tools Used

* + - Python 3.6 is used as the programming language and frame works like numpy, pandas, sklearn and other modules for building the model.
    - PyCharm is used as IDE.
    - For visualizations seaborn and parts of matplotlib are being used.
    - For data collection SQl database is being used.
    - Front end development is done using HTML/CSS/JAVASCRIPT.
    - Flask is used for both data and backend deployment.
    - GitHub is used for version control.
    - Heroku is used for deployment.

**1.6 Data Requirement**

The info demand totally supported the matter statement. and also, the information set is accessible on the Kaggle within the type of standout sheet(.xlsx). Because the main theme of the project is to induce the expertise of real time issues, we have a tendency to ar once more mercantilism {the information into the prophetess database and commerce it into csv format.

**1.7 Constraints**

The Mushroom classifier prediction answer should be attainable and also the user should not be needed to understand any of the operations.

**1.8 Assumptions**

The most objective of the project is to implement the utilization cases for the

new dataset that the user provides through the programme. Machine learning

model is employed to process the on top of a computer file. it's additionally

assumed that each one aspect of this project has the flexibility to figure along

within the approach the designer is expecting.

1. **Design Flow**

**2.1 & 2.2 Model Flow & Deployment Process**



# 2.3 Logging

# Each step is being logged within the system that runs internally, that shows the

# date time and therefore the process that has been performed, work is completed

# in several layers as information, DEBUG, ERROR, WARNINGS. This provides

# the US the perception of the logged info.

# 2.4 Error Handling

# Once a slip has occurred, the reason is logged in its several log files, in order that

# the developer will rectify the error.

**3 Performance analysis**

**3.1 Reusability**

Elements of the code written are accustomed to different applications and

therefore the rest is changed and reused.

**3.2 Application Compatibility**

The various parts for this project are exploitation python as an associate interface

between them. every element can have its own tasks to perform, and it's the work

of the python to make sure the transfer of data.

**3.3 Resource Utilization**

Once any task is performed, it'll doubtless use all the process power offered till

that performance is finished.

**3.4 Deployment**

The model is being deployed on Heroku.

**Conclusion**

The flight fare prediction will predict the worth of the trained knowledge set

within the rule. therefore, the user will recognize the approximate value for his

or her journey