High Level Design (HLD)

Concrete Compressive Strength Prediction

Revision Number : 1.0
Last date of revision : 03/07/2023

Document Version Control

Date Issued	Version	Description	Author
04/07/2023	1	Initial HLD – V1.0	Shailendra Sahu

Contents

1. Introduction

1.1 Why this High-Level Design Document?

The purpose of this High-Level Design (HLD) Document is to add the necessary detail to the current project description to represent a suitable model for coding. This document is also intended to help detect contradictions prior to coding, and can be used as a reference manual for how the modules interact at a high level.

The HLD will:

- Present all of the design aspects and define them in detail
- Describe the user interface being implemented
- Describe the hardware and software interfaces
- Describe the performance requirements
- Include design features and the architecture of the project
- List and describe the non-functional attributes like:
    - Security
    - Reliability
    - Maintainability
    - Portability
    - Reusability
    - Application compatibility
    - Resource utilization
    - Serviceability

1.2 Scope

The HLD documentation presents the structure of the system, such as the database architecture, application architecture (layers), application flow (Navigation), and technology architecture. The HLD uses non-technical to mildly-technical terms which should be understandable to the administrators of the system.

1.3 Definitions

Term	Description
CCSP	Concrete Compressive Strength Prediction
IDE	Integrated Development Environment
DATABASE	Collection of all the information monitored by this system
GCP	Google Cloud Platform

2. General Description

2.1 Product Perspective

Concrete Compressive Strength Prediction model is a machine learning regression model which will help us to predict the compressive strength of concrete.

2.2 Problem statement

To create a system to predict the compressive strength of concrete.

2.3 Proposed Solution

The solution proposed here is to build a machine learning regression model, in which, model will take the data of contents mixed in the processes of making the concrete and predict the strength of the concrete based on the data provided.

2.4 Data Requirements

We will require the dataset having complete details about the quantity of the contents mixed in the construction of concrete like cement, blast furnace slag, fly ash, water, superplasticizer, coarse aggregate, fine aggregate, age(days after construction) and the compressive strength found in various test runs.

2.5 Tools used

Python programming language and frameworks such as NumPy, Pandas, Scikit-learn, are used to build the whole model.

- VSCode is used as IDE.
- For visualization of the plots, Matplotlib, Seaborn and Plotly are used.
- GCP is used for deployment of the model.
- MySQL/MongoDB is used to retrieve, insert, delete, and update the database.
- Front end development is done using HTML/CSS
- Python Flask is used for backend development.
- GitHub is used as version control system.

2.5.1 Hardware Requirements

- PC (system with minimum i3 processor and 6GB ram).

2.6 Constraints

The CCSP system must be user friendly, as automated as possible and users should not be required to know any of the workings.

2.7 Assumptions

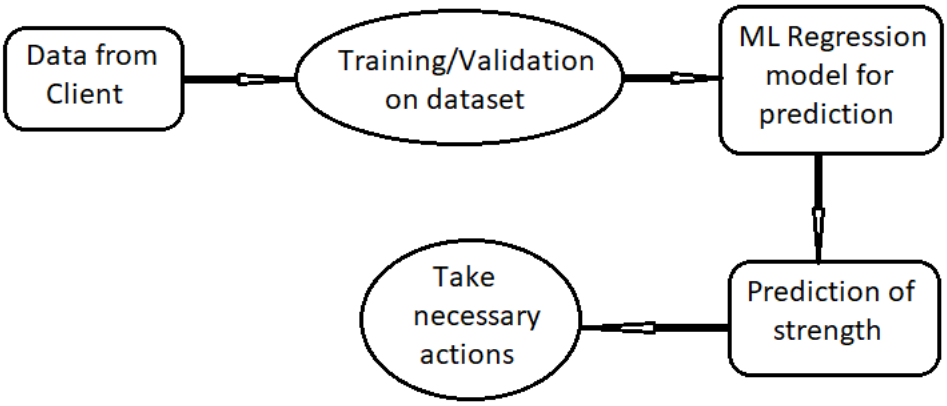
The main objective of the project is to implement the use cases as previously mentioned (2.2 Problem Statement) for new dataset that comes through client. Machine learnig model is used for predicting the above-mentioned use cases based on the input data. It is also assumed that all aspects of this project have the ability to work together in the way the designer is expecting.

3. Design Details

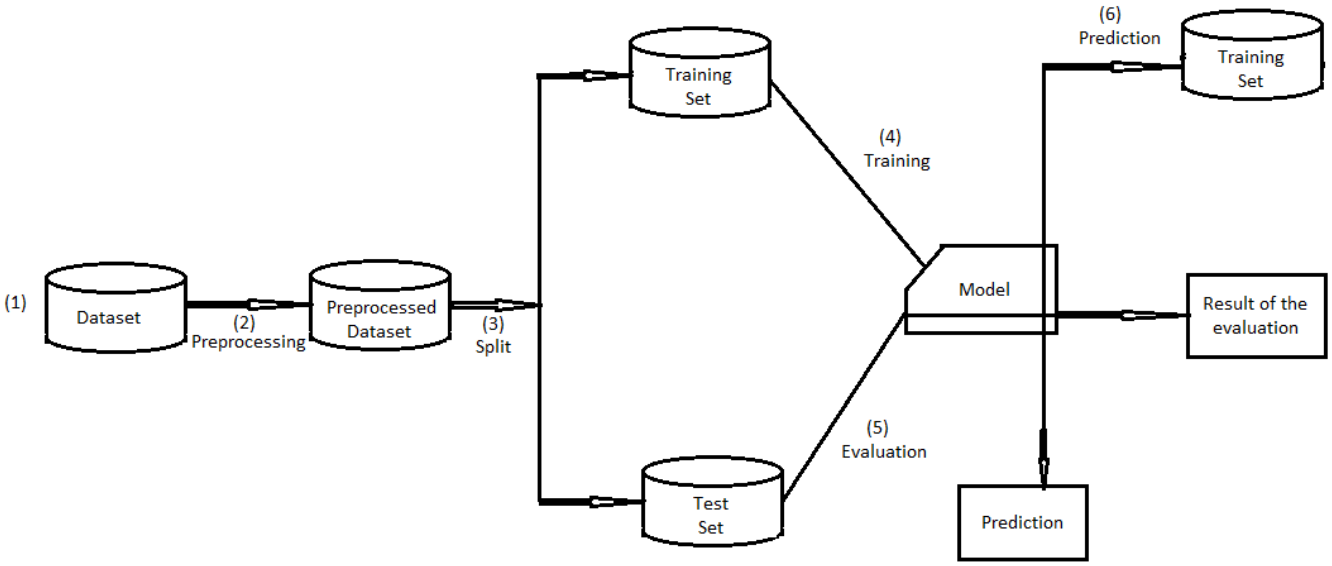
3.1 Process Flow

For predicting the strength, we will use a machine learning model.
Below is the process flow diagram.

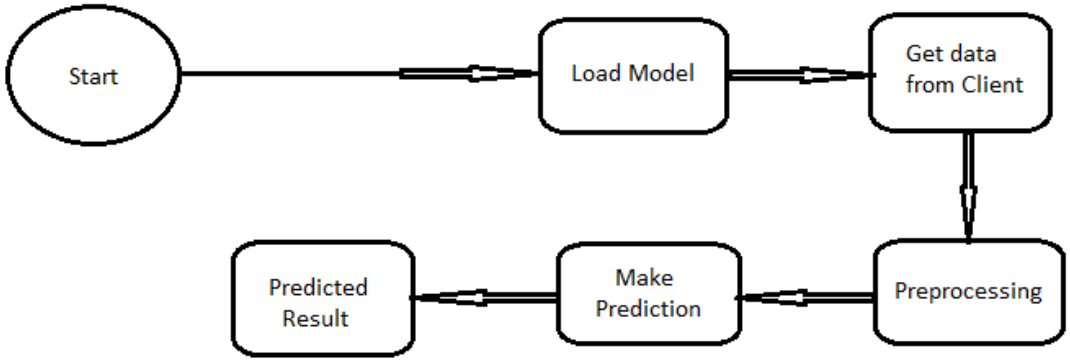
Proposed methodology



3.1.1 Model Training and Evaluation



3.1.2 Deployment Process



3.2 Event log

The system should log every event so that the user will know what process is running internally.

Initial Step-By-Step Description:

1. The System identifies at what step logging required
2. The System should be able to log each and every system flow.
3. Developer can choose logging method. He can also choose database logging/ File logging as well.
4. System should not hang even after using so many loggings. Logging is done so that we can easily debug issues. Therefore logging is mandatory to do.

3.3 Error Handling

Whenever any errors is encountered, an explanation will be displayed as to what went wrong? An error can be defined as anything that falls outside the normal and intended usage.

3.4 Performance

The CCSP model is used for compressive strength prediction of concrete. It will predict the strength of concrete after receiving prediction data and it should be as accurate as possible. Also, model retraining is very important to improve the performance.

3.5 Reusability

The code written and the components used should have the ability to be reused with no problems

3.6 Application Compatibility

The different components for this project will be using Python as an interface between them. Each component will have its own task to perform, and it is the job of the Python to ensure proper transfer of information.

3.7 Resource Utilization

When any task is performed, it will likely use all the processing power available until that function is finished.

3.8 Deployment

4. Conclusion

The CCSP (Concrete Compressive Strength Prediction) model will help us to predict the compressive strength of concrete. Model will take the data of contents mixed in the processes of making the concrete and predict the strength of the concrete based on the data provided. This will make the concrete making process time efficient, cost optimization and more economical use of raw materials.