Campus Placement Prediction System

Revision Number: 1.0

Last date of revision: 16/10/2023

# Document Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| Date Issued | Version | Description | Author |
| 16/10/2023 | 1 | Initial HLD — V1.0 | Kunal Dagar |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Contents

[Document Version Control. 2](#_TOC_250021)

[Abstract. 4](#_TOC_250020)

1. [Introduction 5](#_TOC_250019)
   1. [Why this High-Level Design Document?. 5](#_TOC_250018)
   2. [Scope. 5](#_TOC_250017)
   3. [Definitions 5](#_TOC_250016)
2. [General Description. 6](#_TOC_250015)
   1. [Product Perspective 6](#_TOC_250014)
   2. [Problem statement 6](#_TOC_250013)
   3. [PROPOSED SOLUTION 6](#_TOC_250012)
   4. [FURTHER IMPROVEMENTS 6](#_TOC_250011)
   5. [Data Requirements 7](#_TOC_250009)
   6. [Tools used. 8](#_TOC_250008)
   7. [Constraints 9](#_TOC_250007)
   8. [Assumptions. 9](#_TOC_250006)
3. [Design Details 10](#_TOC_250005)
   1. [Process Flow. 10](#_TOC_250004)
      1. [Model Training and Evaluation 11](#_TOC_250003)
      2. [Deployment Process 12](#_TOC_250002)
   2. [Event log 12](#_TOC_250001)
   3. [Error Handling 12](#_TOC_250000)
4. Performance. 13
   1. Reusability. 13
   2. Application Compatibility 13
   3. Resource Utilization 13
   4. Deployment. 13
5. Conclusion 14

## Abstract

In today's competitive job market, the ability to accurately predict a student's employability is of paramount importance. Our Campus Placement Prediction System is designed to precisely assess and forecast a student's job prospects based on specific, data-driven parameters. Leveraging advanced machine learning techniques and a wealth of relevant data, the system evaluates factors such as academic performance, skills, and past placement records to make informed predictions regarding a student's job placement potential. By reducing uncertainty and subjectivity in the placement process, this system provides valuable insights for both students and recruiters, aiding students in making well-informed career decisions and assisting recruiters in identifying the most suitable candidates for their organizations. This research underscores the pivotal role that technology can play in enhancing the efficiency and effectiveness of campus placement processes, ultimately contributing to more successful transitions from education to employment.

1. **Introduction**

###### 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: o Security
      * Reliability
      * Maintainability
      * Portability
      * Reusability
      * Application compatibility
      * Resource utilization
      * Serviceability

##### 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. **Definitions**

Term CPPS

*Dataset*

*IDE Render*

*Description*

Campus Placement Prediction System

Collection of all the information monitored by this system

Integrated Development Environment

Cloud Service Provider

### General Description

#### Product Perspective

#### 

#### The CPP or Campus Placement Prediction System is a machine learning based web application that provides an information about the placement status of an individual. Based on the certain parameters it will check the chances of an individual to get placed in his college campus placements.

* 1. Problem statement

To create a web application to the predict the placement status of a student based on a certain parameters

* 1. PROPOSED SOLUTION

The proposed solution is a comprehensive web application designed to streamline the assessment and prediction of an individual's placement status. By incorporating a wide array of educational and demographic data, this platform offers a holistic view of a candidate's qualifications, enhancing the accuracy of placement predictions. With inputs including SSC marks and board, HSC marks, board, and specialization, degree marks and category, entrance test scores, as well as specialization and MBA marks, the system's algorithm processes this data to provide a conclusive placement status. This innovative solution leverages advanced data analytics and machine learning techniques to assess a candidate's potential fit for various job opportunities. The web application not only benefits students by offering clear insights into their career prospects but also assists educational institutions and recruiters in making well-informed decisions. In an era driven by data-driven insights, this proposed system is a pivotal step towards optimizing the campus placement process, enhancing the efficiency of matching candidates with job opportunities.

* 1. FURTHER IMPROVEMENTS

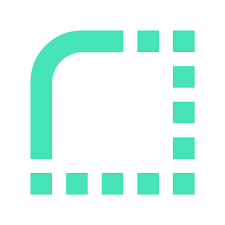
CPPS can be enhanced by integrating additional components, including personality assessments and skill-based analyses. These additions will augment the platform's capabilities, providing students with more holistic insights into their career paths and improving its overall predictive accuracy. Furthermore, introducing job fit checking will enable users to explore opportunities that align with their individual attributes and aspirations, fostering more successful placements and career choices.

* 1. **Data Requirements**

Data requirement completely depend on our problem statement.

* + - Dataset Size: The dataset comprises 274 records, representing individual profiles.
    - Diversity: Despite its modest size, the dataset is diverse, allowing for meaningful analysis and prediction.
    - Attributes: Important attributes include SSC marks, SSC board, HSC marks, HSC board, HSC specialization, degree marks, category, entrance test scores, specialization for the degree program, and MBA marks
  1. Tools used

Python programming language and frameworks such as NumPy, Pandas, Scikit-learn, Matplotlib, Seaborn, Flask, Gunicorn, Render, Html, Css, and Bootstrap were used for this poject.



* + - VS Code iis used as IDE.
    - For visualization of the plots, Matplotlib and Seaborn were used.
    - Anaconda was used for managing virtual environments.
    - Render is used for deployment of the model.
    - Front end development is done using HTML/CSS and Bootstrap
    - Python Flask is used for backend development.
    - Python version 3.8.0 is used.
    - GitHub is used as version control system

#### Constraints

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

#### Assumptions

The main objective of the project is to implement the use cases as previously mentioned (2.2 Problem Statement) for new dataset that is taken as the input by the web application and is provided to Machine Learning model that uses Support Vector Classifier as classification algorithm for prediction results.

## Design Details

##### Process Flow

For identifying the different types of anomalies, we will use a deep learning base model. Below is the process flow diagram is as shown below.

Proposed methodology

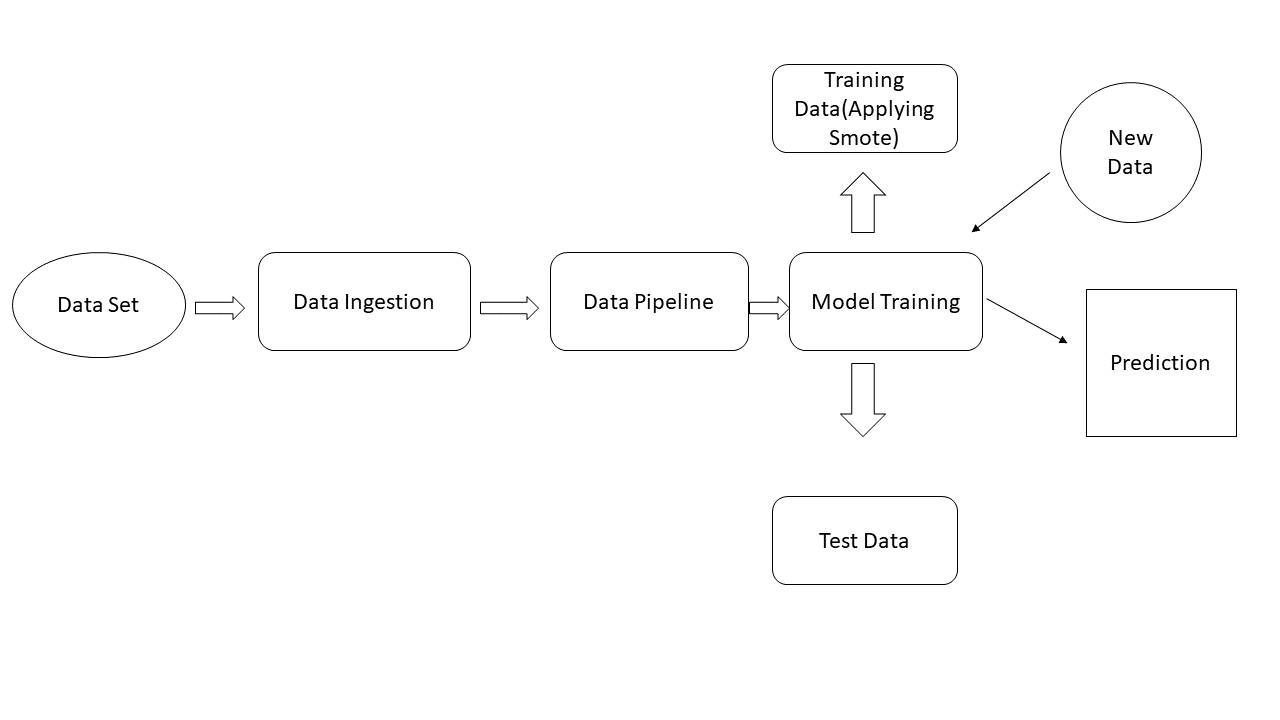
APPLY ML MODEL

DATA VALIDATION

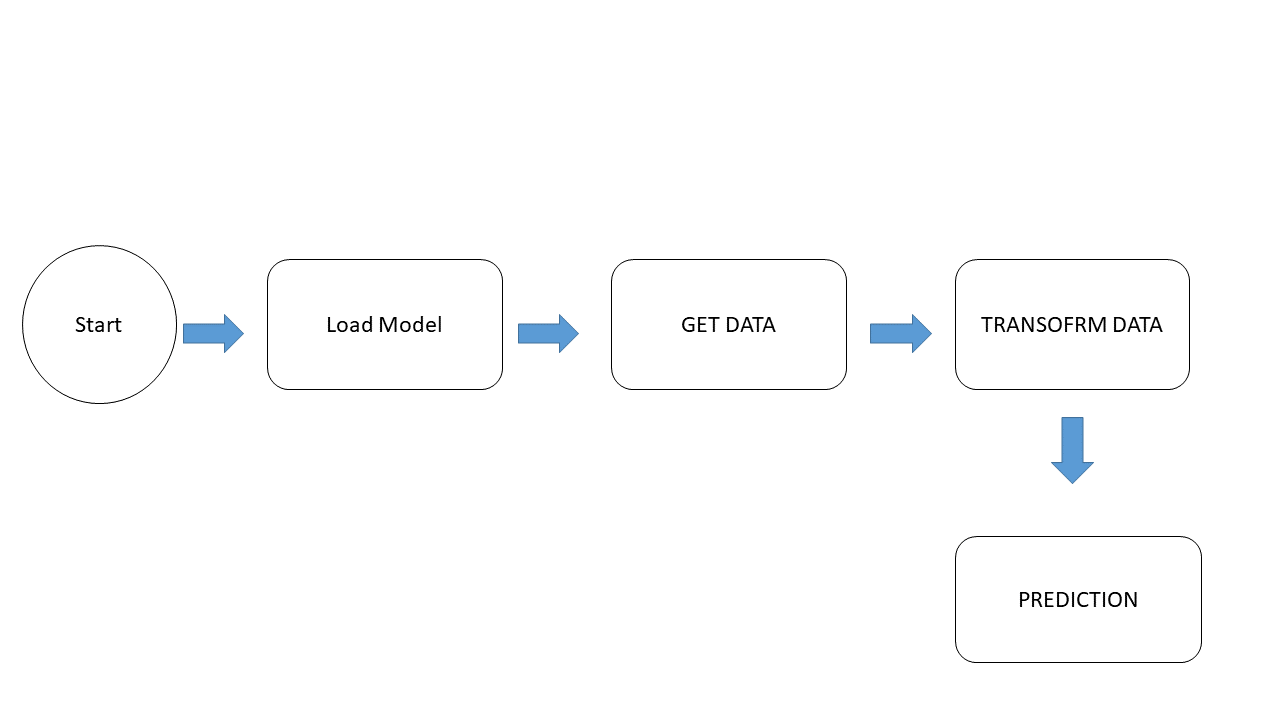
NEW DATA FROM WEB APP

PREDICTING RESULT

##### Model Training and Evaluation



##### Deployment Process

\

* 1. 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. You can choose database logging/ File logging as well.
4. System should not hang even after using so many loggings. Logging just because we can easily debug issues so logging is mandatory to do.
   1. **Error Handling**

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

## Performance

The CPPS can be used by both students and organization to check whether a student can fulfill their academic requirements and a student can also check which his growth so he can also improve his skill set..

#### Reusability

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

problems.

#### 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.

#### Resource Utilization

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

#### Deployment



## Conclusion

The CPPS or the Campus Placement prediction System is a Machine Learning Based prediction system that uses Support Vector Classifier to predict that whether a student is a fit for a job or not based on his academic performance.

## References

1. <https://ieeexplore.ieee.org/abstract/document/9758214>
2. https://www.ripublication.com/ijaer19/ijaerv14n9\_19.pdf