# Table of Contents

|  |  |
| --- | --- |
| **1** | **INTRODUCTION** |
| **2** | **SYSTEM ANALYSIS** |
|  | 2.1 Scope of the Project |
|  | 2.2 Aim of Project |
|  | 2.3 Hardware and Software Requirement |
| **3** | **SYSTEM DESIGN** |
|  | 3.1 System Perspective |
|  | 3.1.1 Architecture Diagram 3.1.2 Data flow Diagram |
| **4** | **IMPLEMENTATION.** |
|  | 4.1 Discussion of code segment |
|  | 4.2 Datasets Collection. |
|  | 4.3 Modules Explanation |
|  | 4.4 Software Testing |
| **5 DISCUSSION OF THE RESULTS.** | |
|  | 5.1 Screen Shots |
| **6 CONCLUSION AND FUTURE ENHANCEMENT** | |
|  | **REFERENCES** |

**ABSTRACT**

As the name specifies “STUDENT PERFORMANCE ANALYSIS SYSTEM” is the software developed for analyzing the student performances in the academic. The student performance analysis system provides an easy way for the teachers and students in searching the details of the academic attendance report and marks/percentage details the with graph. Performance analysis of outcome based on learning is a system which strive for excellence at different levels and diverse dimensions in the field of student’s interests.

Student performance modelling is one of the challenging and popular research topics in educational data mining, we could find only a few specific surveys on student performance analysis and prediction. These specific surveys are limited in nature and primarily focus on studies that try to identify possible predictor or model student’s performance. However, the previous works do not address the temporal aspect of prediction. Moreover, we could not find any such specific survey which focuses only on classroom-based education.

* It helps in maintaining students’ records.
* Graphical user interface is user friendly.
* Easy for student to view his academic records.
* Machine learning regression algorithm is used to predict marks for upcoming test.
* Easy to handle.
* It provides information about Test Performances, detailed analysis of each subject, anytime.
* It also acts as a platform where the complete academic information about the student.

**Chapter 1**

**INTRODUCTION**

Student Performance Analysis System is an emerging field and is very crucial to schools and

universities in helping their students and professors. Most of the pre-existing methods are based

only on past academic performance of students. This system aims to develop models which can

predict the student’s performance and grades while keeping in mind other equally essential

personality factors like interests, attributes and opinions (IAO variables) which affect their

lifestyle. It uses various machine learning and deep learning techniques to predict the

performance of students, and basic exploratory data analysis to derive various correlations of

student’s performance

Student Performance Analysis System is an emerging field and is very crucial to schools and

universities in helping their students and professors. Most of the pre-existing methods are based

only on past academic performance of students. This system aims to develop models which can

predict the student’s performance and grades while keeping in mind other equally essential

personality factors like interests, attributes and opinions (IAO variables) which affect their

lifestyle. It uses various machine learning and deep learning techniques to predict the

performance of students, and basic exploratory data analysis to derive various correlations of

student’s performance

Student Performance Analysis System is an emerging field and is very crucial to schools and

universities in helping their students and professors. Most of the pre-existing methods are based

only on past academic performance of students. This system aims to develop models which can

predict the student’s performance and grades while keeping in mind other equally essential

personality factors like interests, attributes and opinions (IAO variables) which affect their

lifestyle. It uses various machine learning and deep learning techniques to predict the

performance of students, and basic exploratory data analysis to derive various correlations of

student’s performance

Student Performance Analysis System is an emerging field and very crucial to schools and universities in helping their students and professors. Most of the pre-existing methods are based only on past academic performance of students. This system aims to develop models which can predict the student’s performance and grades while keeping in mind other equally essential personality factors like interests, attributes which affects their lifestyle. It uses various machine learning and deep learning techniques to predict the performance of the students.

The project aims to predict a student’s performance by analyzing behavioral patterns and existing grades. It helps us to identify the success factors and success blockers. The project contains an algorithm which analyses attributes like: existing grades, absences, number of hours studied, etc. The Algorithm then predicts an expected performance based on the attributes. The project is based on the Machine Learning and the algorithm used is Linear Regression. Linear Regression Algorithm helps us to identify correlation between variables or attributes.

* **PYTHON:**

Python is a general-purpose, interpreted, high-level programming language popularly used for website development, data analytics and automation.

Python is a general-purpose language which means it is versatile and can be used to program many different types of functions. Because it is an interpreted language, it precludes the need for [compiling](https://www.techopedia.com/definition/540/compile) code before execution and because it is a high-level programming language, Python is able to abstract details from code. In fact, Python focuses so much attention on abstraction that its code can be understood by most novice programmers.

Python code tends to be short and when compared to compiled languages like C and C++, it executes programs slower. Its user-friendliness makes it a popular language for citizen developers working with machine learning algorithms in low-code no-code (LCNC) software applications.

Python has a simple [syntax](https://www.techopedia.com/definition/3959/syntax) and is known for having a large community that actively contributes to a growing selection of software modules and libraries. Python’s initial development was spearheaded by Guido van Rossum in the late 1980s. Today, Python is managed by the [Python Software Foundation](https://www.python.org/psf/).

**Modules used:**

* **Pandas:**

Pandas is an open-source library that is made mainly for working with relational or labeled data both easily and intuitively. It provides various data structures and operations for manipulating numerical data and time series. This library is built on top of the NumPy library. Pandas is fast and it has high performance & productivity for users.

* **NumPy:**

NumPy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays. It is the fundamental package for scientific computing with Python. It is open-source software.

* **Matplotlib:**

Matplotlib is an amazing visualization library in Python for 2D plots of arrays. Matplotlib is a multi-platform data visualization library built on NumPy arrays and designed to work with the broader SciPy stack. It was introduced by John Hunter in the year 2002. One of the greatest benefits of visualization is that it allows us visual access to huge amounts of data in easily digestible visuals. Matplotlib consists of several plots like line, bar, scatter, histogram etc.

* **Streamlit:**

The trend of Data Science and Analytics is increasing day by day. From the data science pipeline, one of the most important steps is model deployment. We have a lot of options in python for deploying our model. Some popular frameworks are Flask and Django. But the issue with using these frameworks is that we should have some knowledge of HTML, CSS, and JavaScript. Keeping these prerequisites in mind, Adrien Treuille, Thiago Teixeira, and Amanda Kelly created “Streamlit”. Now using streamlit you can deploy any machine learning model and any python project with ease and without worrying about the frontend. Streamlit is very user-friendly.

* **SkLearn:**

Scikit-learn is mainly coded in Python and heavily utilizes the NumPy library for highly efficient array and linear algebra computations. Some fundamental algorithms are also built in Cython to enhance the efficiency of this library. Support vector machines, logistic regression, and linear SVMs are performed using wrappers coded in Cython for LIBSVM and LIBLINEAR, respectively. Expanding these routines with Python might not be viable in such circumstances. Scikit-learn works nicely with numerous other Python packages, including SciPy, Pandas data frames, NumPy for array vectorization, Matplotlib, seaborn and plotly for plotting graphs, and many more.

* **Datasets:**
  + **CSV files:**

A CSV is a comma-separated values file, which allows data to be saved in a tabular format. CSVs look like a garden-variety spreadsheet but with a .csv extension. CSV files can be used with most any spreadsheet program, such as Microsoft Excel or Google Spreadsheets. They differ from other spreadsheet file types because you can only have a single sheet in a file, they cannot save cell, column, or row. Also, you cannot not save formulas in this format.

* + **Excel Files:**

Excel is a spreadsheet program from Microsoft and a component of its Office product group for business applications. Microsoft Excel enables users to format, organize and calculate data in a spreadsheet. By organizing data using software like Excel, data analysts and other users can make information easier to view as data is added or changed. Excel contains a large number of boxes called cells that are ordered in rows and columns. Data is placed in these cells. Excel is a part of the Microsoft Office and Office 365 suites and is compatible with other applications in the Office suite. The spreadsheet software is available for Windows, macOS, Android and iOS platforms.

**Chapter 2**

**SYSTEM ANALYSIS**

**2.1 Scope of the Project**

The different areas where we can use this application are:

* Any Educational Institute can make use of it, for the departmental level and at basic subject level for providing a brief overview of the performance related to particular course.
* This system is specifically developed for the faculty and the students.

**2.2 Aim of Project**

The aim of the project is to providing the online interface for students, faculty etc. Student information analysis system aims to improve the efficiency of college information management, and the main function is to predict the upcoming exam’s results with high accuracy and understand where the student is lacking and also providing insights on various aspects of student’s academic performance. Through this the teacher can keep track of overall class performance and also individual student performance.

**2.3 Hardware and Software Requirement**

**2.3.1 Hardware Configuration**

* Intel i3 or Ryzen 3 Processor
* 4 GB RAM
* 30 GB HDD
* 1024 \* 768 Resolution Color Monitor.

**2.3.2 Software Configuration**

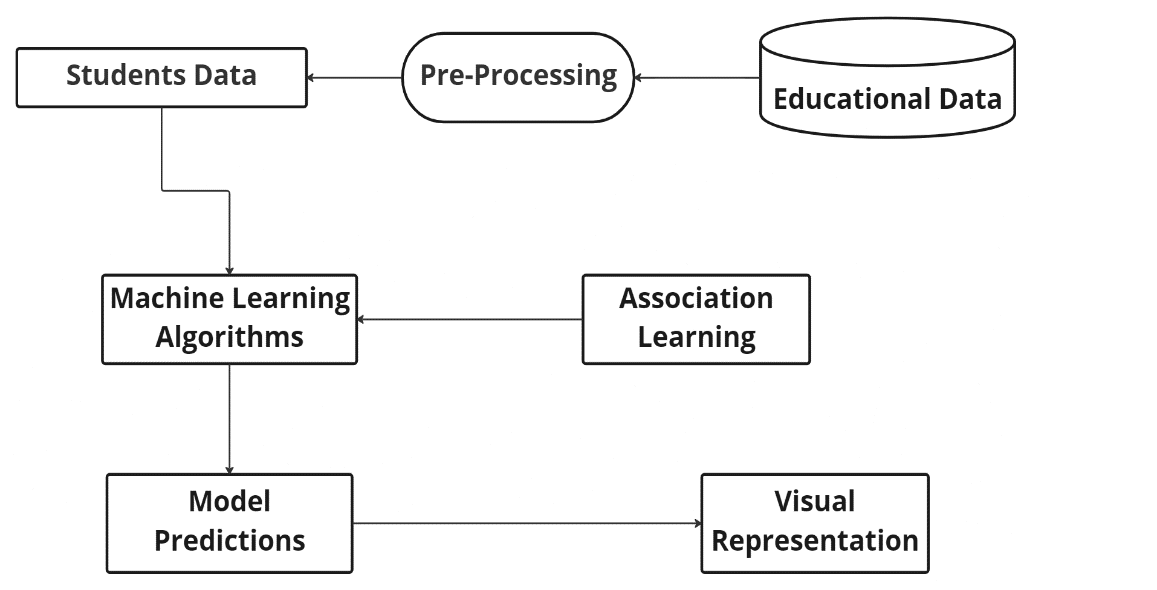
* OS: Windows 7 or above, Linux, Mac
* Datasets (excel or csv files)
* Required python modules
* Web Browser
* Streamlit (Local host)

**Chapter 3**

**SYSTEM DESIGN**

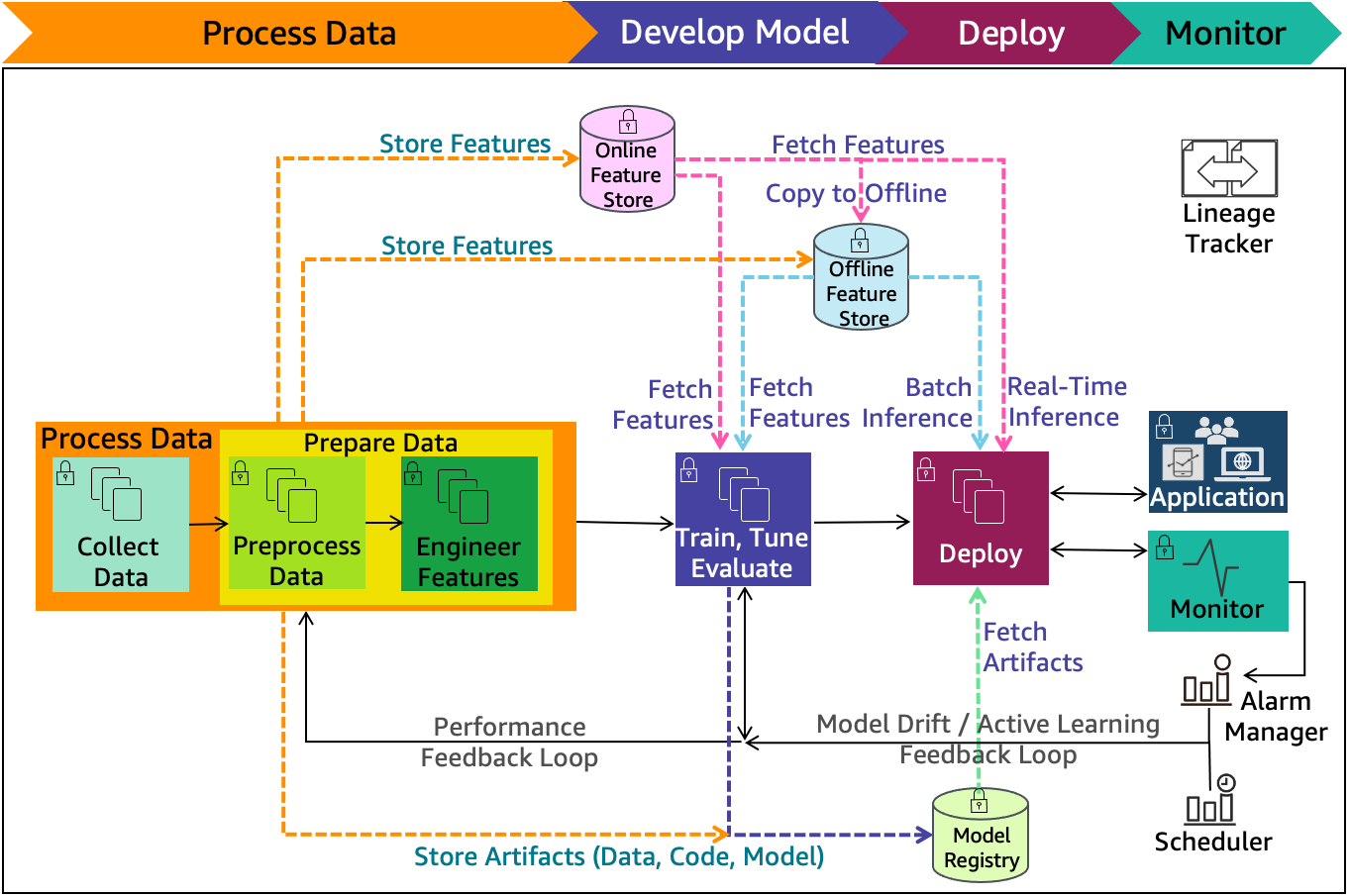
**3.1 System Perspective:**

**3.1.1 Architecture Diagram**

****

Machine Learning (ML) is concerned with the construction and study of system that can learn from data. For example, ML can be used in e-mail message to learn how to distinguish between spam and inbox messages. Classification is the process of finding a model (or function) that describes and distinguishes data classes or concepts. The model is derived based on the analysis of a set of training data (i.e., data objects for which the class labels are known). The model is used to predict the class label of objects for which the class label is unknown.

**3.1.2 Data flow Diagram**

****

A data-flow diagram (DFD) is a graphical representation of the “flow” of data through information system. DFDs can also be used for the visualization of data processing (structured design). On a DFD, data items flow from an external data source or an internal data store to an internal data store or an external data sink, via an internal process. A DFD provides no information about the timing or ordering of processes, or about whether the flow processes will operate in sequence or in parallel. It is therefore quite different from a flowchart, which shows of control through an algorithm, allowing a reader to determine what operations will be performed, in what order, and under what circumstances, but not what kinds of data will be input to and output from the system, nor where the data will come from and go to, nor where the data will be stored (all of which are shown on a DFD).

**Chapter 4**

**IMPLEMENATION**

**4.1 Discussion of Code segment**

* **Result Prediction Part**

# Load the data into a Pandas dataframe

    training\_data = pd.read\_csv("datasets/csv/training\_data.csv", ';')

    # Training the Machine learning model

    training\_data = training\_data[['G1', 'G2', 'G3', 'absences', 'failures', 'studytime']]

    predict = 'G3'

    X = np.array(training\_data.drop(['G3'], axis = 1))

    y = np.array(training\_data[predict])

# Split the data into training and test sets

    X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size = 0.2)

# Train the model using different Machine Learning algorithms

    match choose:

        case 'LinearRegression':

            model = LinearRegression()

            model.fit (X\_train, y\_train)

        case 'RandomForestRegressor':

            model = RandomForestRegressor(n\_estimators =100, random\_state = 0)

            model.fit(X, y)

        case 'ElasticNet':

            model = ElasticNet(alpha = 1.0, l1\_ratio = 0.5)

            model.fit(X\_train, y\_train)

        case 'ExtraTreesRegressor':

            model = ExtraTreesRegressor(n\_estimators = 100)

            model.fit(X\_train, y\_train)

        case 'SVR':

            model = SVR(kernel = 'rbf', degree = 3, C = 1.0, gamma = 'auto')

            model.fit(X\_train, y\_train)

        case 'GradientBoostingRegressor':

            model = GradientBoostingRegressor(n\_estimators = 50)

            model.fit(X\_train, y\_train)

# Evaluate the model

    st.subheader("Accuracy of the model:")

    train\_accuracy = int(model.score(X\_train, y\_train) \* 100)

    test\_accuracy = int(model.score(X\_train, y\_train) \* 100)

# Display the accuracy

    st.write(f"Train accuracy: {test\_accuracy}%")

    st.write(f"Test accuracy: {test\_accuracy}%")

# Displaying the predicted results

    sub1 = data.columns[0]

    sub2 = data.columns[1]

    st.subheader("Student Grades")

    array = data.to\_numpy()

    new\_data = np.array(array)

    predictions = model.predict(new\_data)

    st.write("Select the number of students to display")

    length = st.slider('Pick', 0, len(predictions))

    predictions = pd.DataFrame(predictions)

    test = pd.read\_csv("datasets/csv/testing\_data.csv")

    test = test[['regno', 'name', 'age', 'gender', sub1, sub2]]

    result = pd.concat([test, predictions], axis = 1, join = 'inner')

    result.rename(columns = {0:'Prediction'}, inplace = True)

    st.table(result[:length])

* **Data Analysis part**

st.write("## Insights on your data...")

with st.container():

    st.subheader("Age of the students:")

    col1, col2 = st.columns(2)

    with col1:

        fig = plt.figure(figsize = (10,4))

        sns.countplot(x = 'age', data = df)

        st.pyplot(fig)

with col1:

        fig = plt.figure(figsize = (10,4))

        sns.kdeplot(df['age'])

        st.pyplot(fig)

    with col1:

        st.subheader("Number of Failures")

        fig = plt.figure(figsize = (10,4))

        sns.countplot(x = 'failures', data = df)

        st.pyplot(fig)

col5, col6 = st.columns(2)

    with col5:

    st.subheader("Do Urban students perform better than Rural students?")

            fig = plt.figure(figsize = (10,4))

            sns.kdeplot(df.loc[df['address'] == 'U', 'CG1'], label = 'Urban', shade = True)

            sns.kdeplot(df.loc[df['address'] == 'R', 'CG1'], label = 'Rural', shade = True)

            plt.title('Do urban students score higher than rural students?')

            plt.xlabel('Grade')

            plt.ylabel('Density')

            st.pyplot(fig)

**4.2 Datasets Collection**

Data is an essential component of any AI model and, basically, the sole reason for the spike in popularity of machine learning that we witness today. Due to the availability of data, scalable ML algorithms became viable as actual products that can bring value to a business, rather than being a by-product of its main processes.

Collecting and organizing the data plays a prominent role in the Machine Learning Projects.

The datasets collected should be in the excel or csv file types.

The data should be cleaned before accessing it. That is the data in the files should not contain the NULL values. NULL values interrupt the processing of the data in the calculation part. The data should be in the correct format as shown with the example file. All the columns should be present with respect to each student.

Then after collecting the data, it is then imported into the project with the help of Pandas library. Pandas’ library in python helps in reading the given data from csv or excel file types in to Pandas DataFrames, this DataFrame is later used for the analysis part.

* **Splitting Your Data: Training, Testing, and Validation Datasets in Machine Learning**

Usually, a dataset is used not only for training purposes. A single training dataset that has already been processed is usually split into several parts, which is needed to check how well the training of the model went. For this purpose, a testing dataset is usually separated from the data. Next, a validation dataset, while not strictly crucial, is quite helpful to avoid training your algorithm on the same type of data and making biased predictions.

**4.3 Modules Explanation**

1. **Strategy: matching the problem with the solution**

In the first phase of an ML project realization, company representatives mostly outline strategic goals. They assume a solution to a problem, define a scope of work, and plan the development. For example, your eCommerce store sales are lower than expected. The lack of customer behavior analysis may be one of the reasons you are lagging behind your competitors.

1. **Dataset preparation and preprocessing**

Data is the foundation for any machine learning project. The second stage of project implementation is complex and involves data collection, selection, preprocessing, and transformation.

* **Data collection**

It’s time for a data analyst to pick up the baton and lead the way to machine learning implementation. The job of a data analyst is to find ways and sources of collecting relevant and comprehensive data, interpreting it, and analyzing results with the help of statistical techniques. The type of data depends on what you want to predict.

* **Data visualization**

A large amount of information represented in graphic form is easier to understand and analyze. Some companies specify that a data analyst must know how to create slides, diagrams, charts, and templates.

* **Data selection**

After having collected all information, a data analyst chooses a subgroup of data to solve the defined problem. For instance, if you save your customers’ geographical location, you don’t need to add their cell phones and bank card numbers to a dataset. But purchase history would be necessary. The selected data includes attributes that need to be considered when building a predictive model.

* **Data preprocessing**

The purpose of preprocessing is to convert raw data into a form that fits machine learning. Structured and clean data allows a data scientist to get more precise results from an applied machine learning model. The technique includes data formatting, cleaning, and sampling.

1. **Dataset splitting**

A dataset used for machine learning should be partitioned into three subsets — training, test, and validation sets.

**Training set.** A data scientist uses a training set to train a model and define its optimal parameters — parameters it has to learn from data.

**Test set.** A test set is needed for an evaluation of the trained model and its capability for generalization. The latter means a model’s ability to identify patterns in new unseen data after having been trained over a training data. It’s crucial to use different subsets for training and testing to avoid model overfitting, which is the incapacity for generalization we mentioned above.

**Validation set.** The purpose of a validation set is to tweak a model’s hyperparameters — higher-level structural settings that can’t be directly learned from data. These settings can express, for instance, how complex a model is and how fast it finds patterns in data. The proportion of a training and a test set is usually 80 to 20 percent respectively. A training set is then split again, and its 20 percent will be used to form a validation set. At the same time, machine learning practitioner Jason Brownlee suggests using 66 percent of data for training and 33 percent for testing. A size of each subset depends on the total dataset size.

1. **Model training**

After a data scientist has preprocessed the collected data and split it into three subsets, he or she can proceed with a model training. This process entails “feeding” the algorithm with training data. An algorithm will process data and output a model that is able to find a target value (attribute) in new data — an answer you want to get with predictive analysis. The purpose of model training is to develop a model.

1. **Model deployment**

The model deployment stage covers putting a model into production use. Once a data scientist has chosen a reliable model and specified its performance requirements, he or she delegates its deployment to a data engineer or database administrator. The distribution of roles depends on your organization’s structure and the amount of data you store.

**4.4 Software Testing**

**White Box Testing:**

**White box testing** techniques analyze the internal structures the used data structures, internal design, code structure, and the working of the software rather than just the functionality as in black box testing. It is also called glass box testing or clear box testing or structural testing. White Box Testing is also known as transparent testing, open box testing.

Working process of white box testing**:**

* Input**:** Requirements, Functional specifications, design documents, source code.
* Processing**:** Performing risk analysis for guiding through the entire process.
* Proper test planning**:** Designing test cases so as to cover the entire code. Execute rinse-repeat until error-free software is reached. Also, the results are communicated.
* Output**:** Preparing final report of the entire testing process.

**Black Box Testing:**

Black Box Testing is a software testing method in which the functionalities of software applications are tested without having knowledge of internal code structure, implementation details and internal paths. Black Box Testing mainly focuses on input and output of software applications and it is entirely based on software requirements and specifications. It is also known as Behavioral Testing.

Types of Black Box Testing:

There are many types of Black Box Testing but the following are the prominent ones –

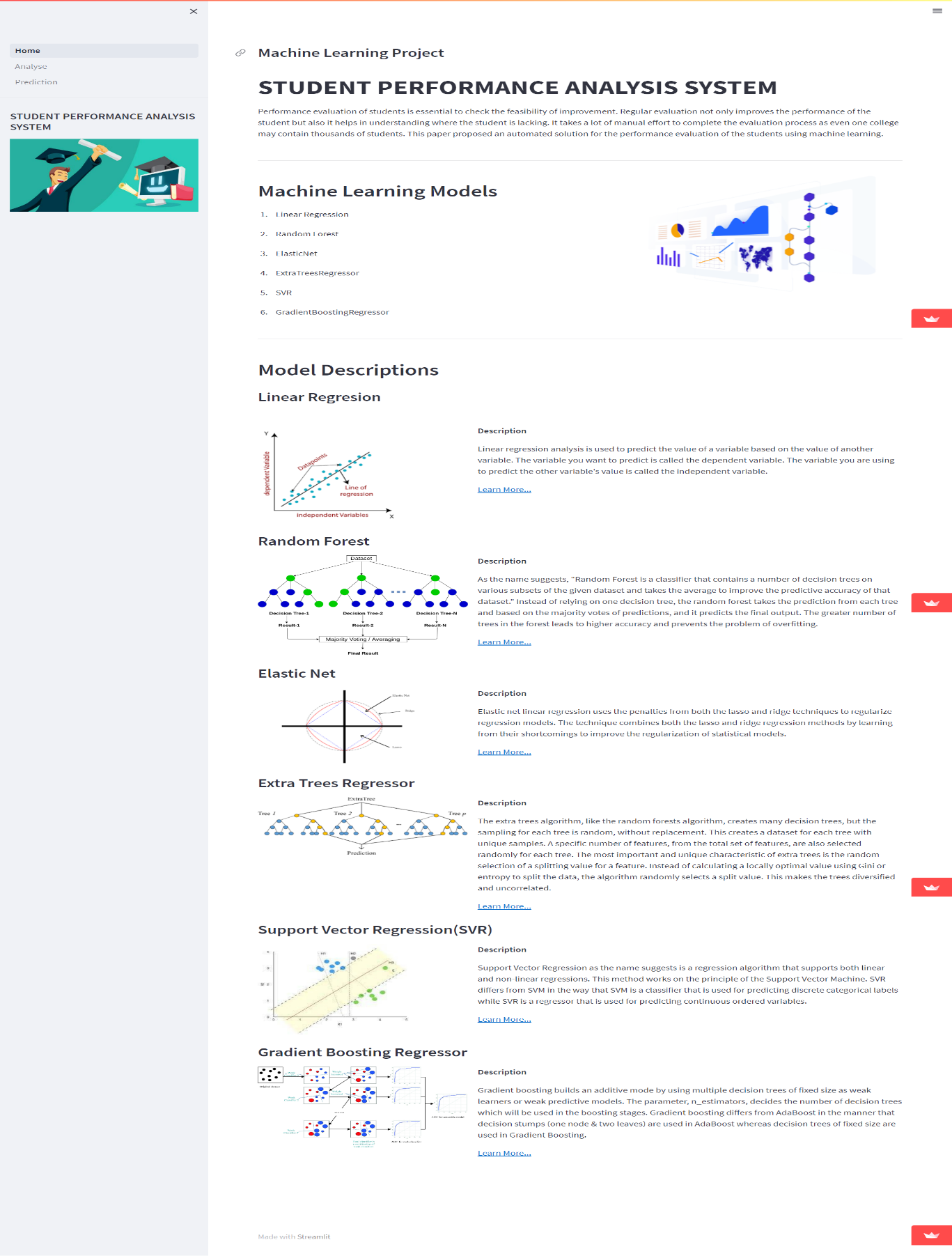
* **Functional testing**: This black box testing type is related to the functional requirements of a system; it is done by software testers.
* **Non-functional testing:** This type of black box testing is not related to testing of specific functionality, but non-functional requirements such as performance, scalability, usability.
* **Regression testing:** [Regression Testing](https://www.guru99.com/regression-testing.html) is done after code fixes, upgrades or any other system maintenance to check the new code has not affected the existing code.

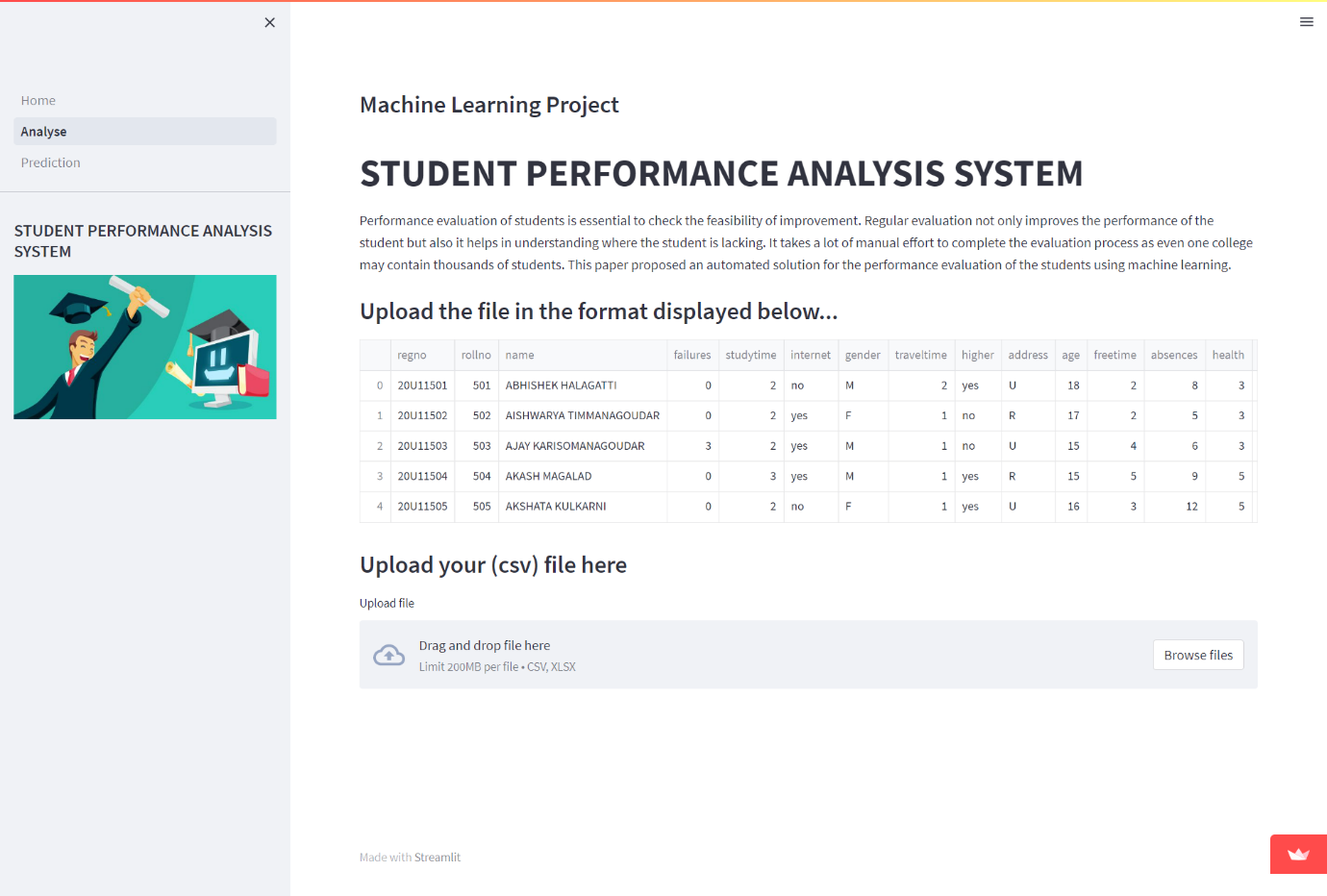
**Chapter 5**

**DISCUSSION OF THE RESULTS**

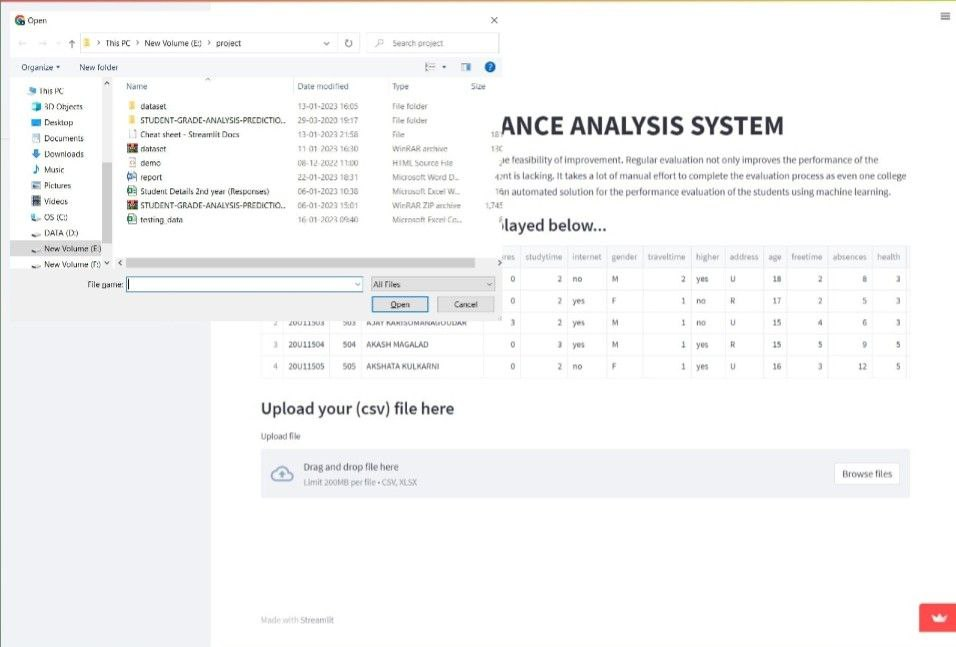
**5.1 Screen Shots**

**Home Page:**

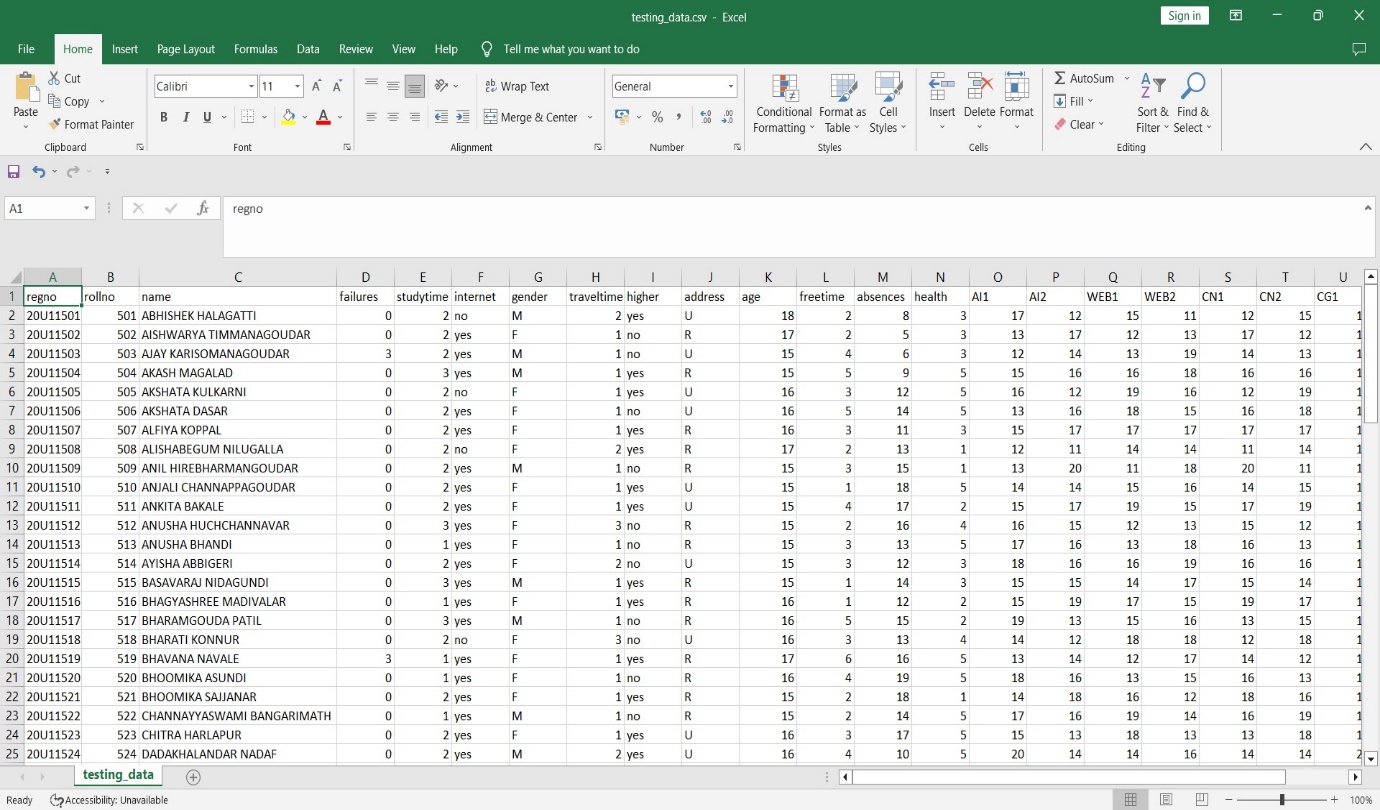
**Analysis page:**

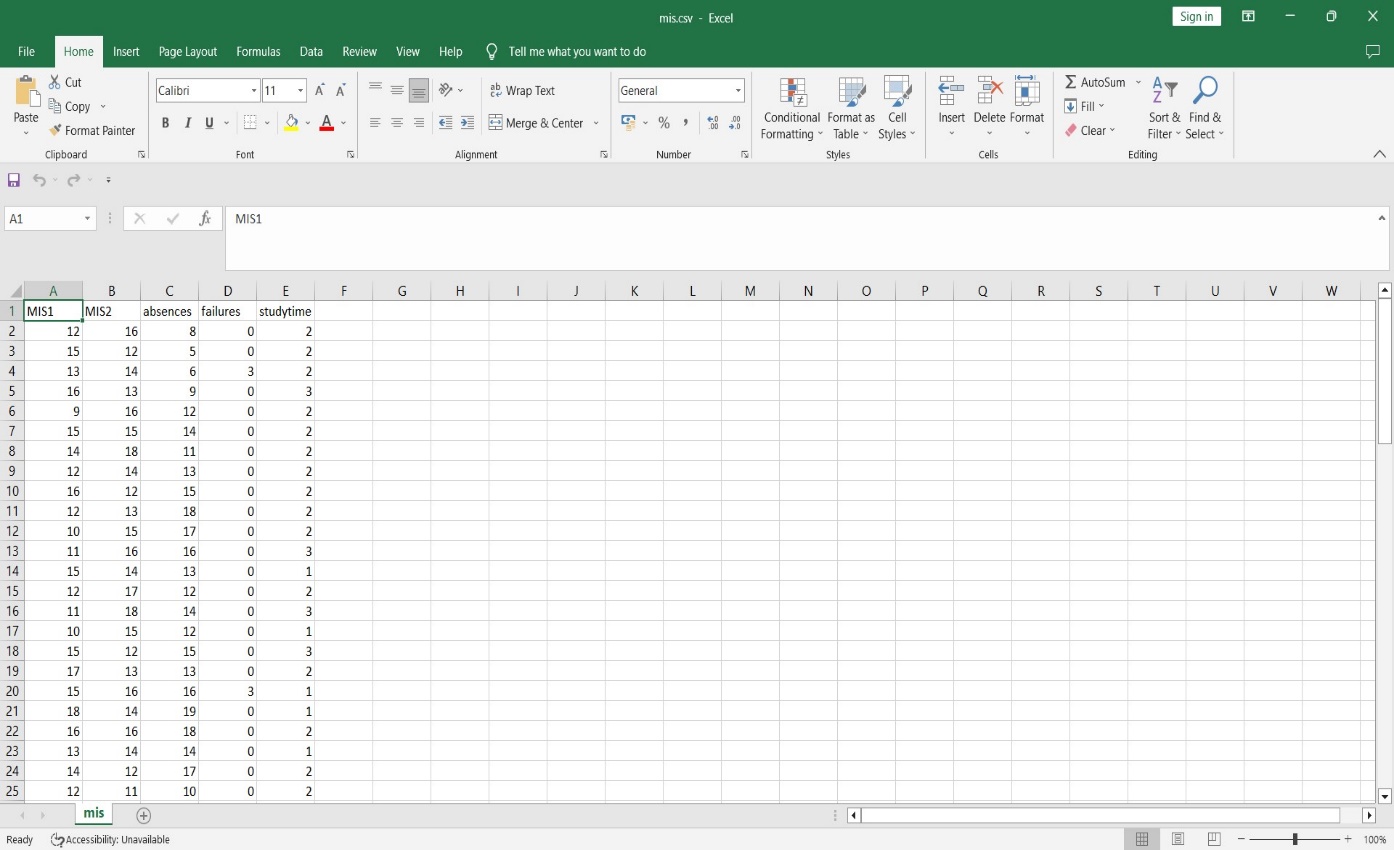


**Uploading the csv file page:**

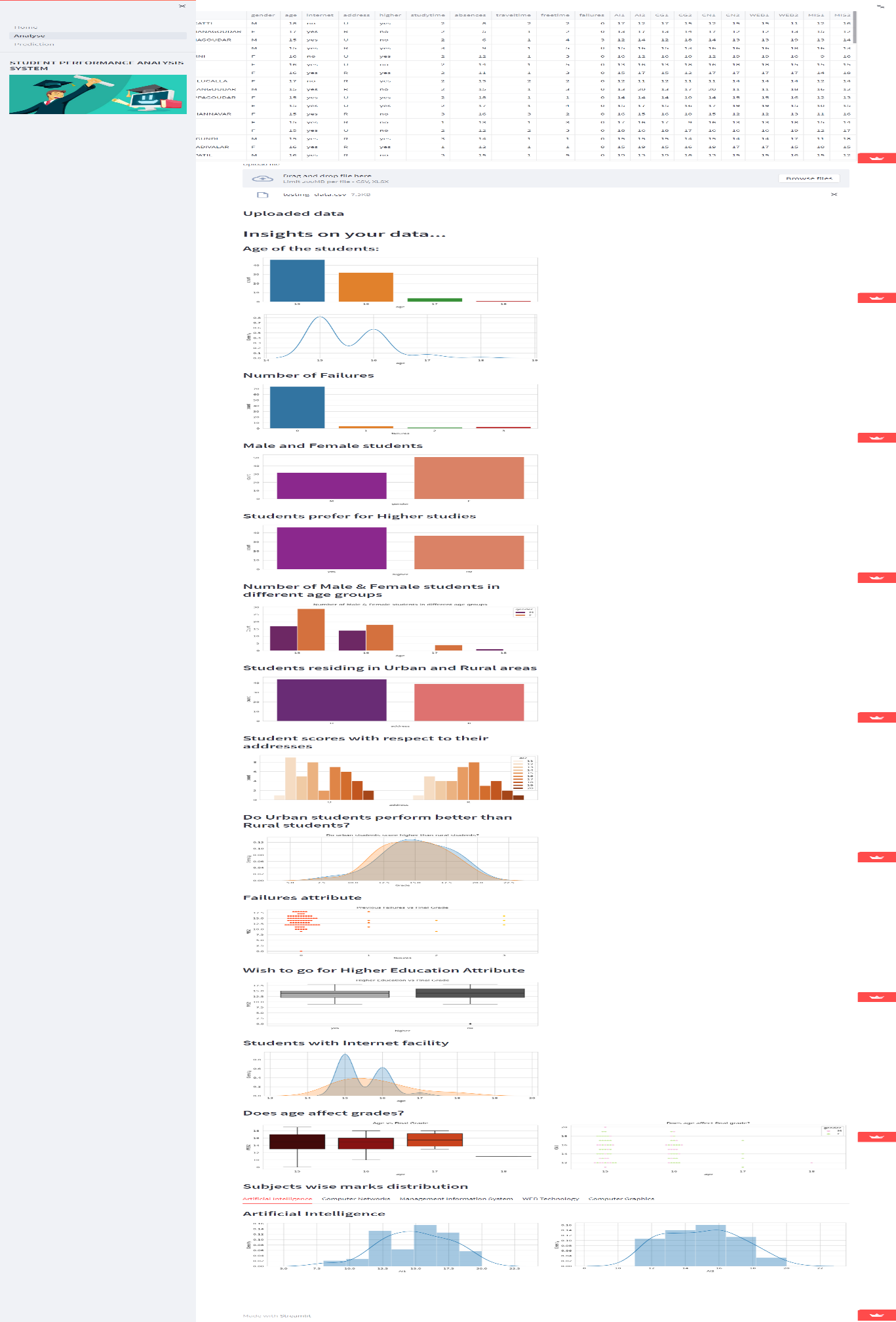


**Dataset:**

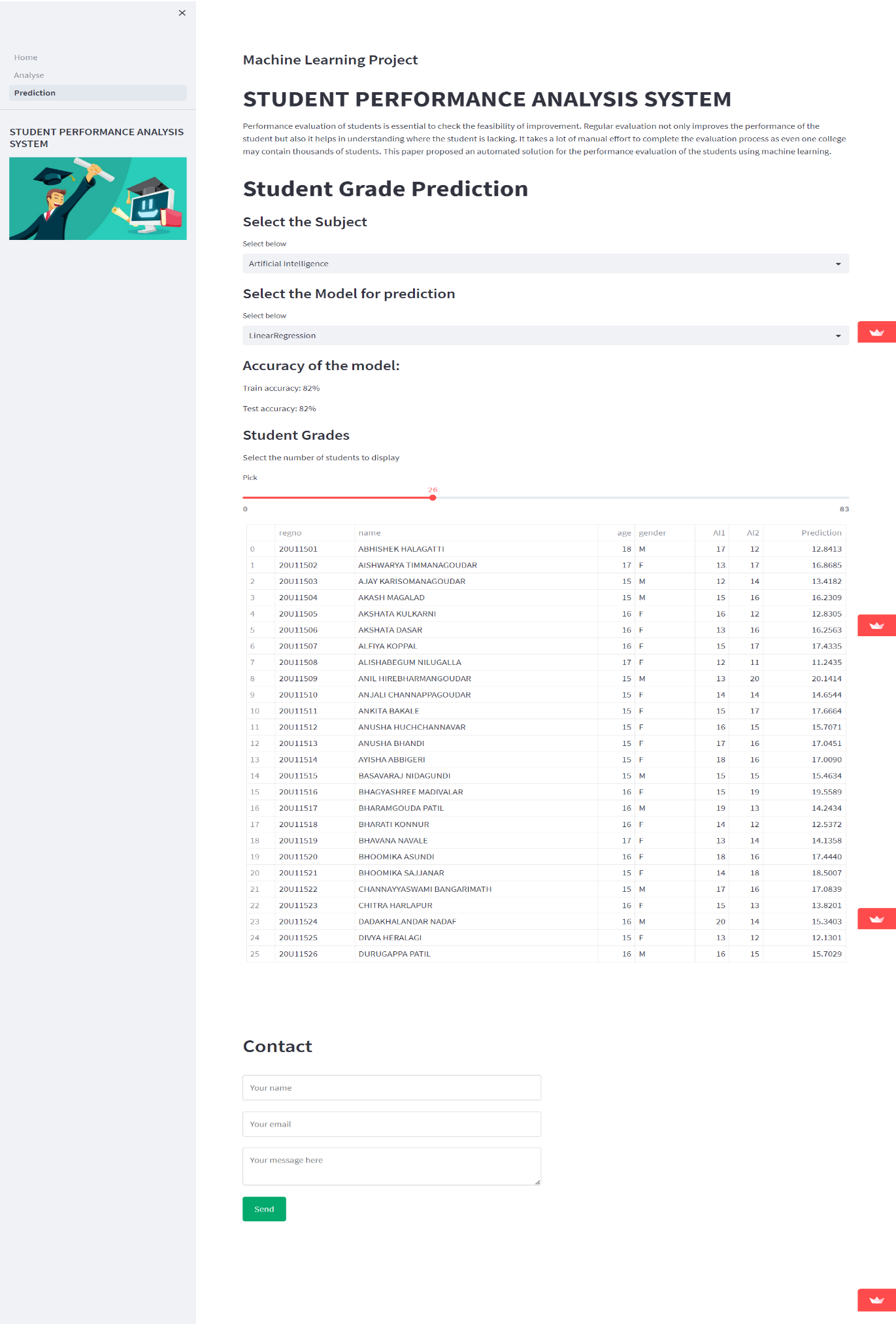




**Data Analysis Page:**



**Prediction Page:**



**Chapter 6**

**CONCLUSION**

In conclusion, the project concentrates on the development of a system for student performance analysis. A data mining technique, classification algorithm is applied in this project to ensure the prediction of the student’s performance in course “System Analysis and Design” is possible. The main contribution of the SPAS is that it assists the lecturers in conducting student performance analysis. The system assists lecturers in identifying the student’s that are predicted to fail in the course “System Analysis and Design”. Other than that, SPAS assists lecturers to retrieve information of their student’s performance throughout the semester.

In conclusion, the project concentrates on the

development of a system for student performance

analysis. A data mining technique, classification

algorithm is applied in this project to ensure the prediction

of the student performance in course “TMC1013 System

Analysis and Design” is possible. The main contribution

of the SPAS is that it assists the lecturers in conducting

student performance analysis. The system assists lecturers

in identifying the students’ that are predicted to fail in the

course “TMC1013 System Analysis and Design”. Other

than that, SPAS assist lecturers’ to retrieve information of

their students’ performance throughout the semesters.

In conclusion, the project concentrates on the

development of a system for student performance

analysis. A data mining technique, classification

algorithm is applied in this project to ensure the prediction

of the student performance in course “TMC1013 System

Analysis and Design” is possible. The main contribution

of the SPAS is that it assists the lecturers in conducting

student performance analysis. The system assists lecturers

in identifying the students’ that are predicted to fail in the

course “TMC1013 System Analysis and Design”. Other

than that, SPAS assist lecturers’ to retrieve information of

their students’ performance throughout the semesters.

In conclusion, the project concentrates on the

development of a system for student performance

analysis. A data mining technique, classification

algorithm is applied in this project to ensure the prediction

of the student performance in course “TMC1013 System

Analysis and Design” is possible. The main contribution

of the SPAS is that it assists the lecturers in conducting

student performance analysis. The system assists lecturers

in identifying the students’ that are predicted to fail in the

course “TMC1013 System Analysis and Design”. Other

than that, SPAS assist lecturers’ to retrieve information of

their students’ performance throughout the semesters.

**REFERENCES**

1. <https://docs.python.org/3/>
2. <https://pandas.pydata.org/docs/>
3. <https://numpy.org/doc/>
4. <https://scikit-learn.org/stable/>
5. <https://docs.streamlit.io/>
6. <https://seaborn.pydata.org/>
7. <https://matplotlib.org/stable/index.html>
8. Stack Overflow