

ACKNOWLEDGEMENT

It is known that a project is a cumulative effort of an entire team. But behind the achievements and work done by this team is the encouragement and support of the faculty. This report would be an injustice if gratitude is not expressed to those who have been assisting in making this project a reality.

We would like to express our heartfelt gratitude to our guide **Ms Eden Jesleen Sequeira**, Assistant Professor of the Department of Computer Science and Engineering, for her valuable suggestions for improvement, consistent guidance and kindhearted support to carry out the project work.

We would like to thank our project coordinator **Ms Lavina Jean D'Silva**, Assistant Professor of the Department of Computer Science and Engineering for her valuable inputs and suggestions that helped improve the project.

We also extend our sincere thanks to **Dr Sridevi Saralaya**, HOD of the Department of Computer Science and Engineering, for her support and encouragement.

We express our immense pleasure and deep sense of gratitude to the Principal, Dr Rio D'Souza for providing the right ambience for carrying out this project and for the unfailing encouragement. We would also like to convey our sincere regards to our Director, Rev. Fr. Wilfred Prakash D'Souza, Assistant Director Rev. Fr. Rohith D'costa, for their constant support and encouragement.

We also extend our thanks to all teaching, non-teaching and administrative staff of Computer Science and Engineering Department who have been very helpful and cooperative towards the successful completion of the project work.

TABLE OF CONTENTS

SL NO	TITLE	PAGE NO
I	Introduction	1
1.1	Problem Definition	1
1.2	Scope and Importance	2
II	Software Requirement Specification	3
2.1	Introduction	3
2.1.1	Purpose	3
2.1.2	Objective	3
2.1.3	Definitions	3
2.1.4	Overview	4
2.2	Overall Description	4
2.2.1	Product Perspective	4
2.2.2	User Characteristics	4
2.2.3	Specific Constraints	5
2.2.4	General Constraints	5
2.3	Specific Requirements	5
2.3.1	External Interface Requirements	5
2.3.1.1	User Interfaces	5
2.3.1.2	Hardware Interfaces	6
2.3.1.3	Software Interfaces	6
2.3.2	Functional Requirements	6
2.3.3	Performance Requirements	6
2.3.4	Design Constraints	7
III	Design	8
3.1	Abstract Design	8
3.1.1	Architectural Diagram	8
3.1.2	Use Case Diagram	9
3.2	Functional Design	10
3.2.1	Sequence Diagram	10

	3.2.2	Complete Data Flow Diagram	11
3.3		Control Flow Design	12
	3.3.1	Complete System Flow Diagram	12
3.4		Access Layer Design	14
	3.4.1	Login page	14
	3.4.2	Signup page	14
	3.4.3	Dashboard page	15
	3.4.4	Page to analyze chart	15
	3.4.5	Statistical analysis	16
IV		Implementation	17
	4.1	Hardware Used	17
	4.2	Software Used	17
	4.2.1	Operating System	17
		4.2.1.1 Windows OS	17
	4.2.2	Languages	18
		4.2.2.1 HTML	18
		4.2.2.2 CSS	18
		4.2.2.3 Python	18
	4.2.3	Flask	18
	4.3	Pseudo Code	19
	4.3.1	New User	19
		4.3.1.1 Sign Up	19
	4.3.2	Registered User	19
		4.3.2.1 Login	19
	4.3.3	Prediction	20
		4.3.3.1 Prediction for good choice or take Alternative measures	20
	4.3.4	Performance	20
		4.3.4.1 Prediction to display top performers among vendors	20

V	Testing	21
5.1	Testing levels	21
5.1.1	Unit testing	21
5.1.2	Integration testing	22
5.1.3	System testing	22
5.1.4	Acceptance testing	22
5.2	Test cases	22
5.3	Limitations	24
VI	Conclusions and Future Work	25
6.1	Future Work	25
VII	References	26
VIII	Appendix	27

LIST OF FIGURES

SL NO	FIGURE	PAGE NO
3.1.1	Architectural Diagram	8
3.1.2	Use Case Diagram of Application user	9
3.2.1	Sequence Diagram	10
3.2.2	Complete Dataflow diagram	12
3.3.1	Complete System Flow diagram	13
3.4.1	Login page	14
3.4.2	Signup page	14
3.4.3	Dashboard page	15
3.4.4	Prediction page	15
3.4.5	Statistical analysis page	16

LIST OF TABLES

SL NO	TABLE	PAGE NO
5.1	Test cases for Signup	22
5.2	Test cases for Login	23
5.3	Test cases for checking performance of Vendor	23
5.4	Test cases for Prediction	24

CHAPTER 1

INTRODUCTION

There are some items in their movement that are highly predictable and routine, while some items are highly unpredictable but still equally important to keep in stock. Consider of conducting testing of significance before developing any implementations of artificial intelligence.

Machine learning and statistics are elements of data science. The term learning in Machine Learning means that the programs depend on some data, used as a training set, to fine-tune some model or algorithm parameters. This includes many techniques such as regression, naive Bayes or supervised clustering. The Stock Management, Profit and Loss Prediction application uses company's datasets and train the model using SVM. Further, the algorithm is able to predict the Vendors based on clustered model data when the test inputs are given. People who deal with the stocks only deal with and manage products.

Stock Management, Profit and Loss Prediction will help these people and it will be easier for people to evaluate and settle on the trade to make the right decision and help them resolve the loss they might have faced for a very long time without having the right vendors.

To overcome the problems listed we propose a desktop application to determine the vendors in their trade quickly and accurately. Stock Management, Profit and Loss prediction uses machine learning algorithms to predict the vendors and it has a precision of around 85 percent for supporting vector machine algorithm and around 90 percent for SKfuzzy algorithm

.

1.1 Problem Definition

Companies rely on their vendors to provide them with goods or services that allow them to run their businesses. Types of vendors include equipment manufacturers, suppliers, such as office supply stores, service providers, such as cleaning services, or companies that supply parts and materials. A company could work with dozens or even hundreds of different vendors.

Effectively managing your list of vendors is important for many reasons, such as reducing spend and ensuring that all processes go smoothly between the order and the payment.

This application is going to sort down the vendors based on necessary parameters and helps these companies to overcome the loss and increase the profit.

1.2 Scope and Importance

- Stock Management, Profit and loss prediction is a machine learning based application which is used to predict the vendor in vendor selection process.
- The application basically counts the performance of the vendor based on his product price, delivering time and quality of the product.
- The model will be loaded with training datasets in-order to predict. The application finds its depth of usefulness when there are a large number of suppliers that the user is dealing with.
- A large input on the supplier performance will also yield the accurate performance of the suppliers. Any missing input would result in an unfair evaluation for the users.

CHAPTER 2

SOFTWARE REQUIREMENT SPECIFICATION

2.1 Introduction

The Software Requirements Specification (SRS) is a document, which will describe the external behavior of the software in-depth. Before heading to the design phase, it is of vital importance that the system developer has a proper understanding of the system so that he can list out the exact user and system requirements that is needed by the system. This document is supposed to entail how the system will behave and respond. A SRS document deals with some of the major requirements of the project and conclude with the design constraints that the system must comply with. In this project we make use of Machine learning and data science.

2.1.1 Purpose

The Applications purpose is to provide a detailed explanation of selecting a Vendor based on certain features. Stock Management, Profit and Loss prediction helps the users in choosing a vendor whose products will eventually lead to the growth of the organization. Selecting a vendor whose products are reliable will help in gaining more profits and also to compare between the vendors.

2.1.2 Objective

A vendor will be able to save their purchase and sales information in the database to manipulate. Our project aims at adding extra useful information that is to suggest the vendors that other same applications are unable to provide. Along with information of the stocks we also providing the behavior of the vendors so that they can be evaluated for the best.

2.1.3 Definitions

Machine Learning: It is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed

Data Science: Data science is a multi-disciplinary field that uses scientific methods, processes, algorithms and systems to extract knowledge and insights from structured and unstructured data.

Training Model: The process of training an ML model involves providing an ML algorithm (that is, the learning algorithm) with training data to learn from. The term ML model refers to the model artifact that is created by the training process.

2.1.4 Overview

The next chapter of this document gives an overview of the functionality of the entire application. It will enlist all the general requirements of the system and is used to create a general basis of the technical requirements that are needed in the subsequent chapters. The third chapter is written primarily for the developers and describes in technical terms the details of the functionality of the application and implementation. Both sections of the document describe the software application and the general and technical implementation of it.

2.2 Overall Description

2.2.1 Product perspective

The model is trained with the large number of datasets. The data sets will contain some certain features .The feature may be regarding the on-time delivery, supply damages, customer feedbacks, price of the product and location of the vendor. The SVM(support vector algorithm) will form a cluster of these datasets which helps for vendor prediction.

2.2.2 User Characteristics

The Users for Stock Management, Profit and Loss prediction mainly includes Entrepreneurs. Entrepreneur may own a shop or Small scale industry unit or large scale industry unit etc. The users need not be in large numbers for the usage of this application. Users do not need to have any prior knowledge for the use of this application except the usage of the application. The user should be able to understand the supplier behavior in

real time. In order to get an effective result of supplier evaluation the user must input the performance parameters on time.

2.2.3 Specific Constraints

Since the application performance is based on the inputs provided to it, Some of the specific constraints are:

- The intimacy between the user and the suppliers is required for the user to know the behavior of the supplier.
- The application needs the database connection since the data to be stored in it.
- The user needs to know the definition and differences of Cost, Quality & Safety, Delivery, Service, Social Responsibility, Convenience/Simplicity, Risk, and Agility.

2.2.4 General Constraints

- Users must accurately input the performance parameters and other inputs.
- On time recording of the behavior of the supplier while in case of on time delivery.
- To be known to differentiate the damaged and undamaged products and its intensity.

2.3. Specific Requirements

This section will provide an in-depth explanation of all the inputs and proposed outputs of the application or system in development.

2.3.1 External Interface Requirements

2.3.1.1 User Interfaces

Client Side

- The user interface is a desktop/web application and can be accessed in smartphones as well.
- User logs in using a login id and a password.
- On successful validation the user can then manipulate and control the application.

- The internet access might or might not be required to use the application, since it depends on user requirement.

Server Side

The application can be hosted on the web or can be locally used in the system with connection to the database. The user can save, update and retrieve the values and also store the results in the database.

2.3.1.2 Hardware Interfaces

- Desktop/Smartphones
- Internet access
- local Storage

2.3.1.3 Software Interfaces

- Database Server
- Windows/Android Operating System

2.3.2 Functional Requirements

Functional requirements are all of the technical details, processing and other specific functionality that defines what a given system must accomplish. Use cases will capture all the functional requirements that are described by behavioral requirements that the system uses.

- The application must be able to register and verify new users for the application.
- The application should detect inappropriate inputs fed into it.
- The user information must be securely stored in order to avoid tampering of suppliers crucial data.
- The application should dynamically update the result for each and every input fed into the application by the users.
- The application must also display the result in graphical format to make the user understand the concept better.

2.3.3 Performance Requirements

- The performance of the overall system should be faster (prediction of vendors need to be quick) and accurate.
- The system should be able to handle large amounts of training data.

- The response time of the login process, displaying the results must fast that the user should feel comfortable with.

2.3.4 Design Constraints

- Username and password field has to be entered in their respective formats.
- The application shall prompt the when entered wrong username and password.
- There should be ease of access for the user to use the application i.e to understand each and every element implemented in the application.
- The design must prompt, redirect, alert, suggest with appropriate messages for the action taken by the user on the UI.

CHAPTER 3

DESIGN

3.1 Abstract Design

Abstract Design is a conceptualization of what someone wishes to create. Abstract design uses a visual language of form, color and line to create a composition which may exist with a degree of independence from visual references in the world. Abstraction indicates a departure from reality in description of art. This departure from accurate description can only be slight or it may be partial, or it can be complete.

Design is the planning that lays the basis for the making of every object or system. There are different design philosophies, approaches and methods which strike a balance between a number of different components. Depending on the situation, it can lay emphasis on one or another component.

3.1.1 Architectural Diagram

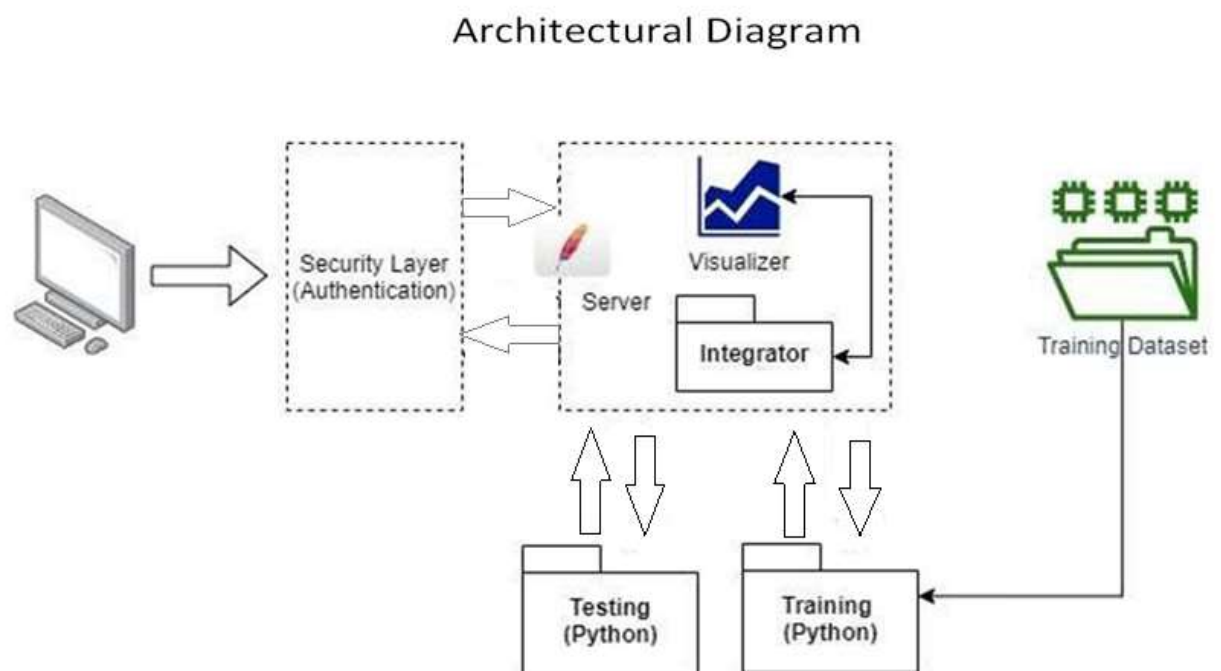


Figure 3.1.1 Architectural design for Stock Management, Profit and Loss Prediction from Vendor performance

The architecture of the project is designed in the figure 3.1.1. Here the application allows a secured way of authentication using apache PHP server. Where the server acts as an intermediate between the front end and the algorithm implementation. The datasets stored in the server are portioned into train and test data sets. Which are the input parameters to the algorithm. The output of the algorithmic computation is then stored back into the server, where the visualization is done based on it in the front end.

In the server the design is based on the relational database tables. Where each database tables are related to one another resulting in providing the input parameters to the algorithm.

3.1.2 Use Case Diagram

A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case is represented by an ellipse containing the name of the use case. An actor is represented by a stickman with the name of the actor below.

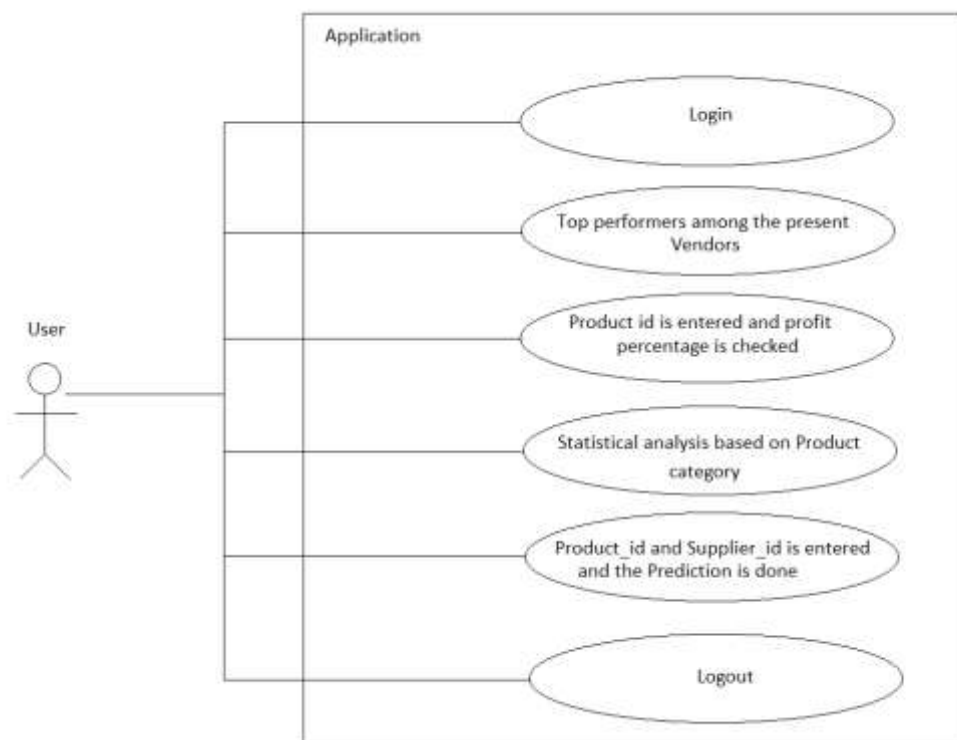


Figure 3.1.2 Use case diagram for Application user

Detailed description of Use Case diagram

Use case: Application

Summary: The End user predicts an appropriate Vendor for the product.

Actor: User

Description: The user need to login to the application using necessary credentials. There will be a dashboard page which represents the top performance of the vendors. If user needs to check the profit percentage of the vendor, then he/she need to enter the product id. There will be a prediction page where user need to enter the test inputs for prediction. The final result of the prediction says where it's a good choice or to take an alternative vendor. Finally there will a statistical analysis page showing the profit on the stocks based on category. Users can exit the application using logout option.

3.2 Functional Design

Functional design simplifies the design of computer software with each modular part of the functional design possessing a responsibility which it performs with minimum of side effects on other parts. Functional design is used to convey that the product's functionality is taken into account in important ways as it is imagined and built.

3.2.1 Sequence Diagram

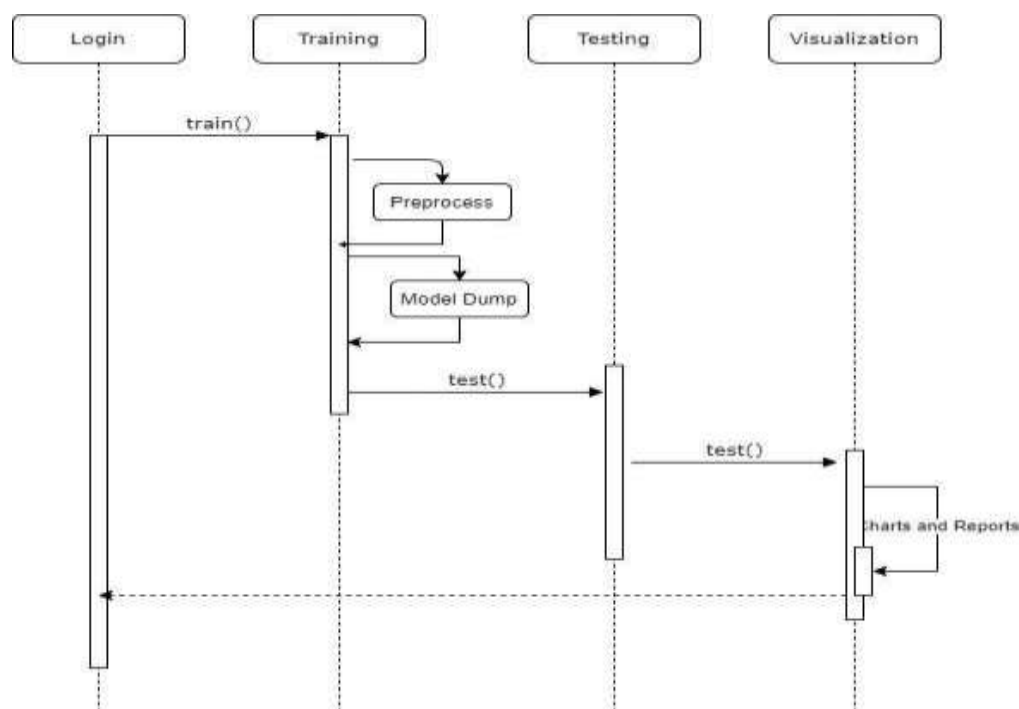


Figure 3.2.1 Sequence Diagram for Stock management and profit loss prediction System

A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. We can also use the terms event diagram or event scenarios to refer to a sequence diagram. It describes how and what order the objects in a system function. In the figure 3.2.1 after a successful login ,training process will begin with the training model and when the training is done we need to input some test data. The testing process is followed by visualization where the output will be shown either in Charts or Bar graphs.

3.2.2 Complete Data Flow Diagram

Graphically representing the functions, or processes, which capture, manipulate, store, and distribute data between a system and its environment and between components of a system. The visual representation makes it a good communication tool between User and System designer. Structure of DFD allows starting from a broad overview and expands it to a hierarchy of detailed diagrams.

Figure 3.2.2 depicts the data flow design pertaining to the admin in which, the admin logs into the account and can manage the user database. After logged in admin will get the information from the varies database such as customer, supplier and stock details. Based on the above details evaluated and the result is shown in terms of chart or report.

In data flow diagram the project is visualised using the database tables and the interfaces. Where the data manipulation is done from the front end that is from the user interfaces. The database is depicted by the cylindrical shapes and the interfaces are in the ellipse shaped structures. Each arrow represents shows the data manipulation either in one direction or in the both directions.

The arrow indicating the both sides shows the control. The direction in both side depicts the data manipulation such as create, alter, display, update, delete. And that shown in one direction has only control over the retrieval of the data.

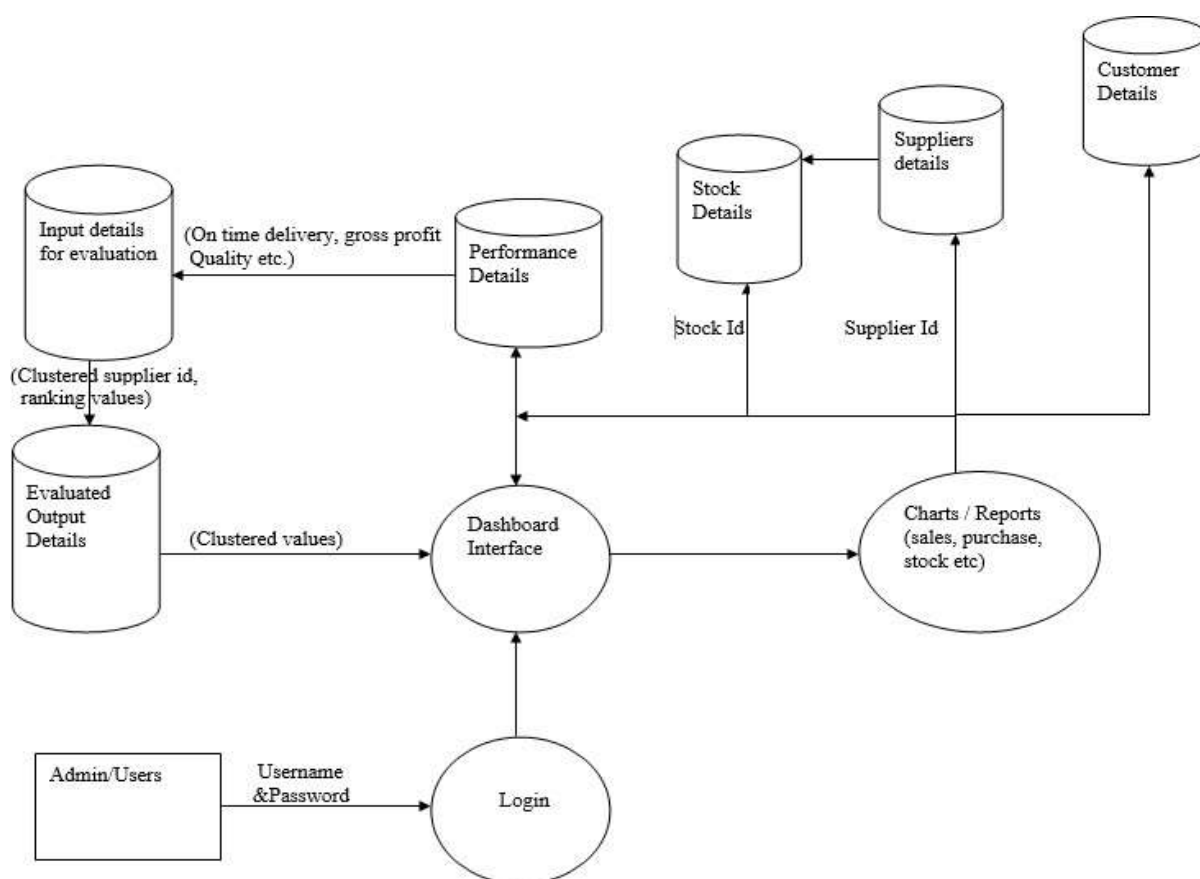


Fig 3.2.2 Data Flow diagram for Stock Management, Profit and Loss prediction

3.3 Control Flow Design

A control flow diagram helps us understand the detail of a process. It shows us where control starts and ends and where it may branch off in another direction, given certain situations. Let's say you are working on software to start a machine. What happens if the engine is flooded, or a spark plug is broken? Control then changes the flow to other parts of the software. We can represent these branches with a diagram. The flow diagram is helpful because it can be understood by both stakeholders and systems professionals.

3.3.1 Complete System Flow Diagram

The complete system flow diagram shown in Figure 3.3.1 demonstrates the flow of the process involved in the functioning of the stock management and profit loss predictor for a user. The flow starts with the registration by the user using unique username and password to create an account. These details are entered into login page. If the login

details are validated, the user is allowed to enter the application, if not, they are required to login with the right credentials.

Once the user is authenticated he/she can create a supplier information, customer information, stock information or invoice information and can create performance information. On getting necessary information the performance information and evaluation can be created. Then the display of ranking and trends takes place.

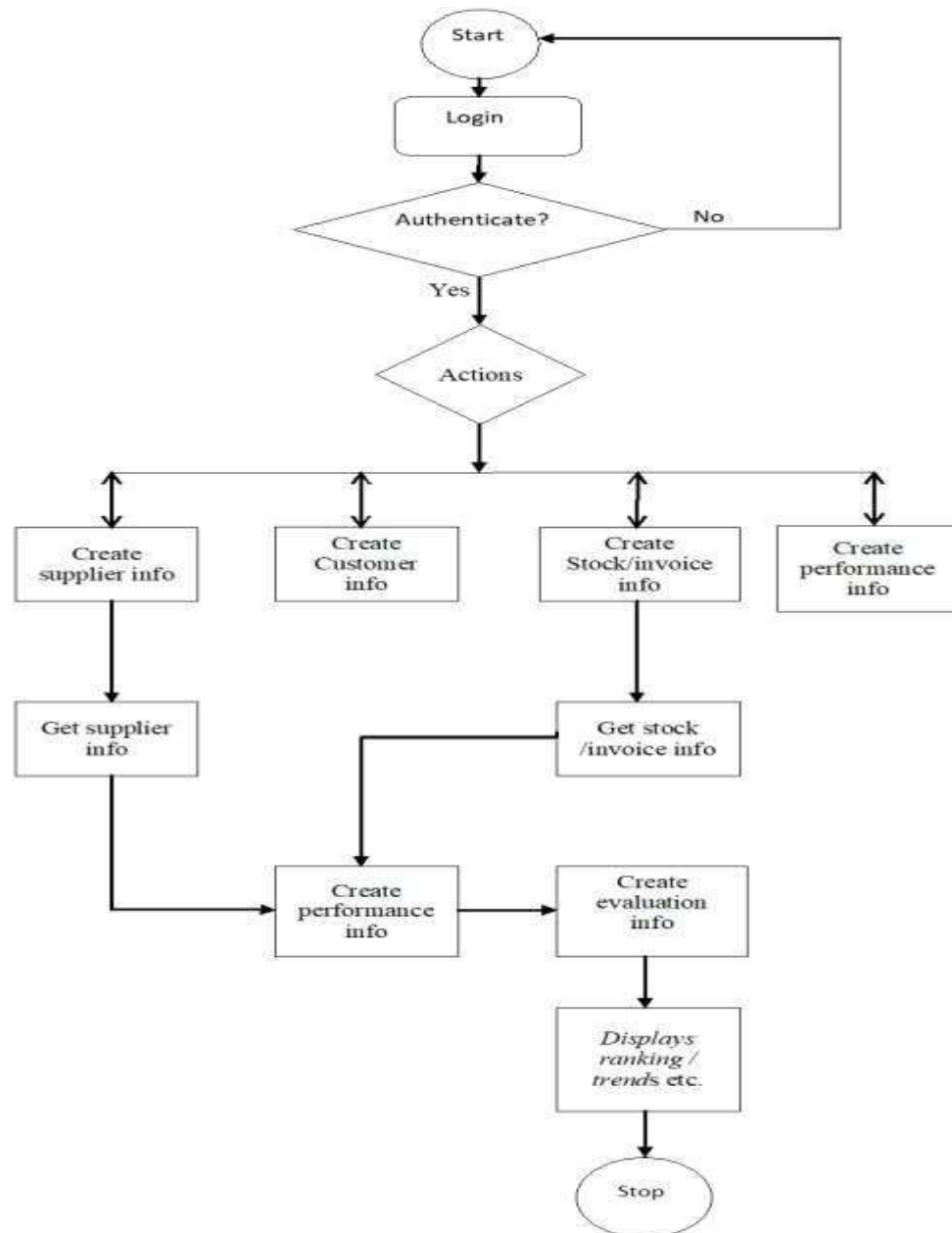


Fig 3.3.1 Complete system flow diagram for Stock Management, Profit and Loss prediction

3.4 Access Layer Design

User interface flow design

The user interface consists of several pages such as Login page, Dashboard page with various parameters, page to display Analysis chart, page to add information or transaction details, Page to add information about particular vendor, page that shows the trend of a particular product and page to display the sales, purchase and stocks.



The image shows a wireframe for a 'Login Page'. It features a central rectangular box containing two input fields: 'Name' and 'Password'. Below the 'Password' field is a 'Login' button. The entire form is centered within a larger rectangular frame.

Figure 3.4.1 Login page



The image shows a wireframe for a 'SignUp' page. It has a title 'SignUp' at the top. Below the title are four input fields with labels: 'Enter the Name', 'Enter the Email', 'Enter the Password', and 'Re-Enter the Password'. At the bottom of the form are two buttons: 'Submit' and 'Login'.

Figure 3.4.2 Signup page

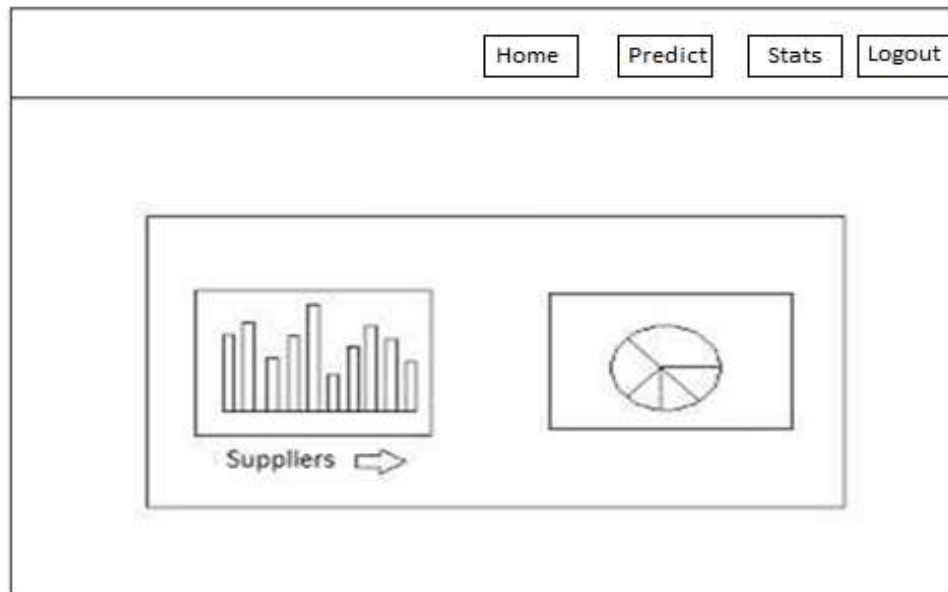
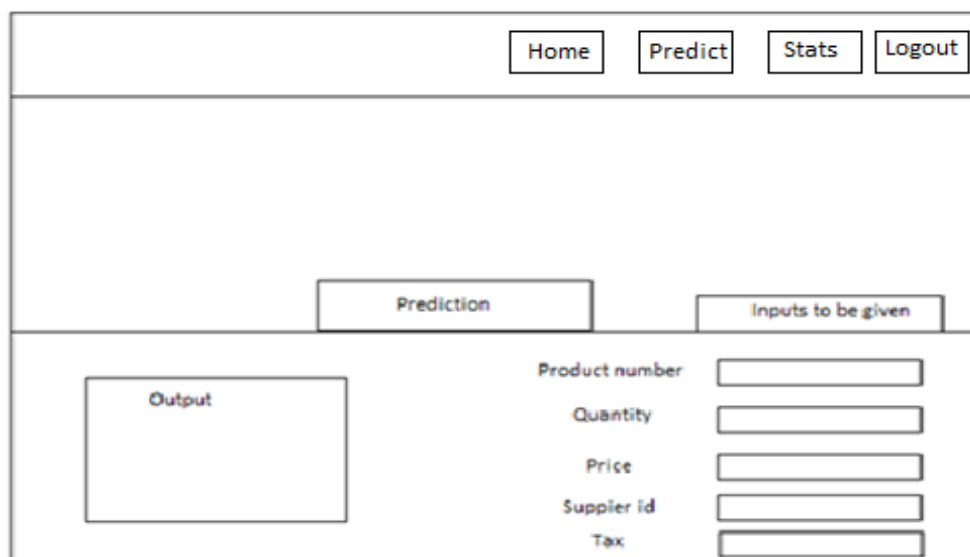


Figure 3.4.3 Dashboard page



The 'Page for Prediction' layout includes the same top navigation bar with 'Home', 'Predict', 'Stats', and 'Logout' buttons. Below the navigation bar, there are two tabs: 'Prediction' and 'Inputs to be given'. The 'Prediction' tab is currently selected. Under the 'Prediction' tab, there is a large rectangular box labeled 'Output'. To the right of the 'Output' box, there are five input fields, each preceded by a label: 'Product number', 'Quantity', 'Price', 'Supplier id', and 'Tax'.

Figure 3.4.4 Page for Prediction

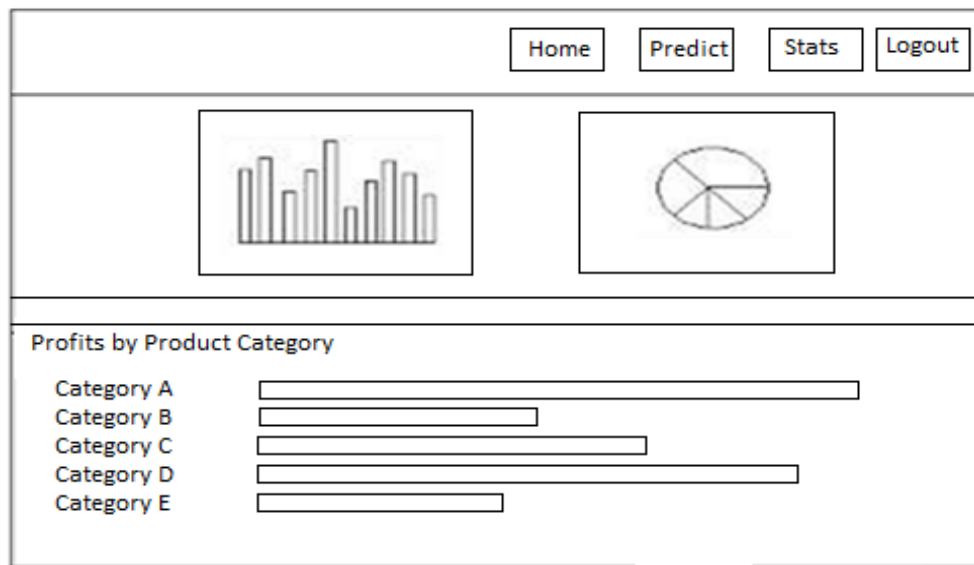


Figure 3.4.5 Page displaying Statistical Analysis on Products

CHAPTER 4

IMPLEMENTATION

Implementation is the stage of the project where the proposed theoretical design is turned into a working system. At this stage the main workload and the major impact on the existing system shifts to the user department. If the implementation is not carefully planned and controlled, it can easily lead to a manic state.

4.1 Hardware Used

Laptop/Personal Computer

- Processor : Intel Core i3-3110M
- RAM : 4GB or above

4.2 Software Used

- Operating System : Windows 7 or above
- Web Interface (Frontend) : HTML & CSS
- Programming Language : Python
- Cross platform web server : Flask

4.2.1 Operating System

An Operating System (OS) is software that manages computer hardware and software resources and provides common services for computer programs. The Operating systems used for the development of this project are as follows:

4.2.1.1 Windows OS

Windows is an operating system designed by Microsoft. The operating system is what allows you to use a computer. Windows comes preloaded on most new personal computers (PCs), which helps to make it the most popular operating system in the world.

4.2.2 Languages

4.2.2.1 HTML

Hypertext Markup Language (HTML) is the standard markup language for creating web page and web applications. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets [2]. In this project HTML is used for displaying the headings, include buttons, inputs, references, images and so on.

4.2.2.2 CSS

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML. CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts [3]. In this project, CSS is used for styling the contents such as font color, size, style, height and width of HTML components.

4.2.2.3 Python

Python is an interpreter, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding; make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

4.2.3 Flask

Flask is a web framework. This means flask provides you with tools, libraries and technologies that allow you to build a web application. This web application can be some web pages, a blog, a wiki or go as big as a web-based calendar application or a commercial website.

4.3 Pseudo Code

4.3.1 New User

4.3.1.1 Signup()

Begin

If (Name, Email, Password) **Then**

If (Length of password > 4) **Then**

 Signup()

 Display “Registration Successful”

Else

 Display “Authentication failed. The given password is invalid”. [Password should be at least 5 characters.]”

EndIf

Else

 Display “Enter all the details”

EndIf

End

4.3.2 Registered User

4.3.2.1 Login()

Begin

If (Name, Password) **Then**

If (Name, Password is valid) **Then**

 Login()

 Display “Login Successful”

Else

 Display “Authentication Failed”

EndIf

Else

 Display “Enter all the required details”

EndIf

End

4.3.3 Prediction

4.3.3.1 Prediction for good to go or take alternative measures

Begin

If (Product_id ,Quantity ,Price ,Supplier_id ,Tax) Then

Predict() [prediction based on profit from that vendor] **Then**

If (The products yields profit with that vendor) Then

 Display “good choice”

Else

 Display “not good choice and take up alternative measures”

EndIf

Else

 Display “Enter all the required fields”

EndIf

End

4.3.4 Performance

4.3.4.1 Prediction to display the Top performers among vendors

Begin

If (Product_id) Then

Predict() [prediction based on profit percentage] **Then**

If (valid product_id) Then

 Display “vendor yielding highest profit and alternative vendors”

Else

 Display “the product is not present within the stock”

EndIf

Else

 Display “Please enter the product_id”

EndIf

End

CHAPTER 5

TESTING

Software testing is a process, to analyze the functionality of a software application to find whether software that has been developed has met all the required specifications and to provide stakeholders with the information about the quality of the product. Software testing also helps to identify the defects present in the application and ensures that the application is defect-free. It is executing a software component or system component to identify if there are any errors, bugs, or, missing requirements incompatible to the actual requirement. It is the process of validating and verifying an application or a software product. It examines if the software product works as per the requirement, if it has achieved all the characteristics and if it has met all the technical requirements.

Software testing plays a very important role in the development of the application and software product. It is cost-effective that is if the bugs are detected in the early stages it costs much less to fix it. It also provides security, improves product quality, and ultimately provides customer satisfaction. Software testing uses some strategies to select the test that is feasible for available time and resources. The possible tests for even a simple software are practically infinite, so the primary aim of testing is to detect software bugs so that defects can be recognized and corrected. Scope of software testing includes analysing the code as well as executing the code in different environments and conditions. It also examines the different aspects of the code so that it has met all the requirements.

5.1 Testing Levels

There are mainly four levels of testing. They are Unit testing, Integration testing, System testing and Acceptance testing.

5.1.1 Unit Testing

Unit testing is a level of software testing, which is used to check whether the individual modules of the code are working properly. It tests each and every unit of the application separately, thus commonly it is known as individual function or procedure. Unit testing is performed by White Box Testing method, where the test is based on the analysis of the internal structure of the component or system. Unit testing is normally done by software developers themselves to ensure the code has met its requirements as intended.

5.1.2 Integration Testing

The second level of software testing is integration testing where individual units are combined and tested in groups. This testing is done to expose the faults in the interaction between the combined units. This process is carried out by running dummy programs like test drivers and test stubs. This is carried out by the software developers or independent testers.

5.1.3 System Testing

The third level of software testing is System testing where complete and integrated software is tested. This testing is carried out to evaluate system compliance with specified requirements. This is normally performed using Black Box Testing and it is done by independent software testers. It is a behavioral testing where the internal structure of the item that is being tested is not known by the software tester.

5.1.4 Acceptance Testing

The final level of software testing is Acceptance testing. In this the system is tested for acceptability before system is available for actual use. The main aim of this testing is to evaluate system compliance with business requirements and check if it is acceptable for delivery. This is done using Black Box Testing and it does not follow any strict procedure.

5.2 Test Cases

5.2.1 Signup()

Table 5.1: Test cases for Signup

Sl.No	Test Cases	Expected output	Observed output	Result
1.	When Name, Email, Password entered is correct	Home page displaying statistical analysis	The statistical analysis page will be displayed	Pass
2.	When name is not entered and all the other fields are entered	Display "Enter Name more than 4 characters"	"Enter Name more than 4 characters" is displayed	Pass

3.	When email is not entered and all other fields are entered	Display “Enter email address”	“Please enter the email address is displayed”	Pass
4.	When Password is not entered and all the other fields are entered	Display “Enter password greater than 4 characters”	“Enter password greater than 4 characters” is displayed	Pass
5.	When re-entered does not match	Display “The re-entered password does not match”	“The re-entered password does not match	Pass

5.2.2 Login()

Table 5.2: Test cases for Login

SI.No	Test Cases	Expected outcomes	Observed output	Result
1.	When name and password entered are correct	Display Home page	“Home page” is displayed	Pass
2.	When name entered is incorrect	Display “Invalid name”	“Invalid name” is displayed	Pass
3.	When password entered is incorrect	Display “Invalid password	“Invalid password” is displayed	Pass

5.2.3 Performance on Product()

Table 5.3: Test cases for checking performance of Vendor

SI.No	Test Cases	Expected outcomes	Observed output	Result
1	Select the product_id	Display the top performers among the vendors for product	Displays “The top performance among the vendors for product”	Pass
2	Enter the name of Vendor	Display the names of Vendors along with overall profit percentage	Displays “The names of vendors along with the overall profit percentage”	Pass

5.2.4 Prediction()

Table 5.4: Test cases for Prediction

Sl.No	Test Cases	Expected output	Observed output	Result
1.	Enter the product_id , quantity, Estimated profit price, supplier_id , Estimated tax	It's a good choice.	Displays "It's a good choice."	Pass
2.	Enter the product_id , quantity, Estimated profit price, supplier_id , Estimated tax	Its not a good choice and go with the following alternative options shown below.	Displays "It's not a good choice and go with the following alternative options shown below."	Pass

5.3 Limitations

- The proposed project needs large amount of training data's of suppliers as well as items.
- If the performance parameters of the supplier are not that accurate then the prediction of the result will be not that accurate.
- If the training data's are not accurate then the cluster formation will be in improper format and the the prediction goes wrong.

CHAPTER 6

CONCLUSIONS AND FUTURE WORK

The problem of supplier selection essentially deals with the selection of the right supplier and their quota allocation which also requires that a variety of vendor attributes such as price and quality needs to be considered. This application will help the user to predict an accurate vendor for a particular product and thus help in prevention of loss and helps the purchase department of small scale or a large scale unit to increase their profit. Thus the usage of this application also helps in analyzing the performance of Vendors and give alternative solution if the test inputs of the user are not accurate.

6.1 Future Work

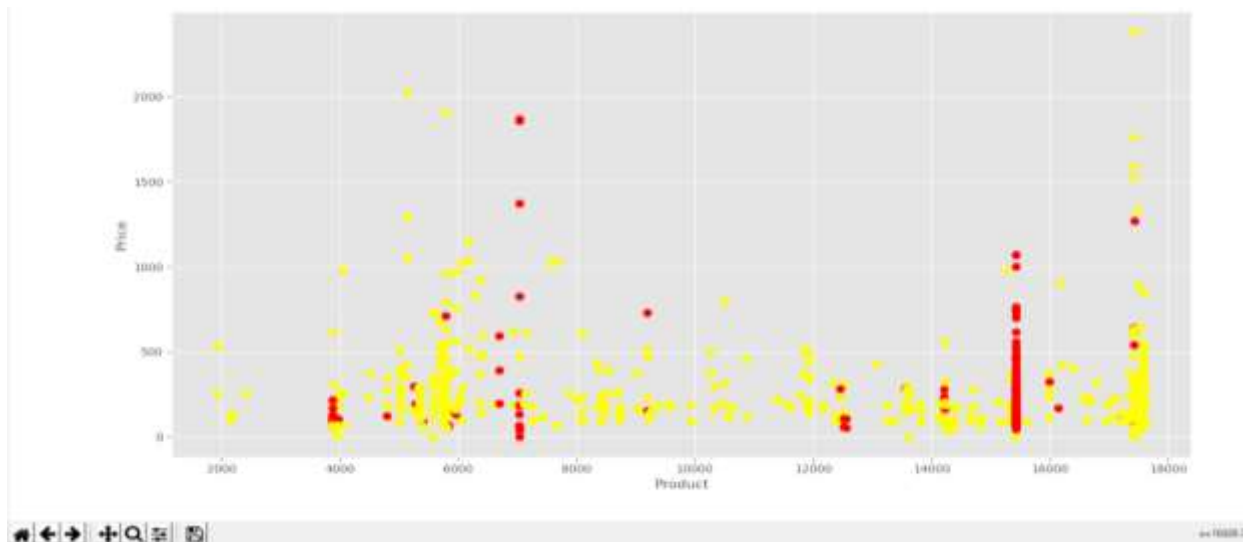
The future work for this application would be developing a training model which will help in analyzing the trends of the products with the help of datasets from sales department. Based on that prediction the purchase department of a company or the small scale unit will take necessary steps.

REFERENCES

- [1]. F. Hedderich, R. Giesecke, and D. Ohmsen. "Identifying and evaluating Chinese suppliers": China sourcing practices of German manufacturing companies. Practix, 9:1–8, (2006).
- [2]. L.M. Ellram. "Total cost modelling in purchasing". (1994). Centre for Advanced Purchasing Studies
- [3]. Arthur L. Corbin. Corbin on Contracts. Matthew Bender & Company, Inc., (2007)
- [4]. Parmar , "Stock Market Prediction Using Machine Learning," 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC), Jalandhar, India, 2018
- [5]. S. M. Idrees, M. A. Alam and P. Agarwal, "A Prediction Approach for Stock Market Volatility Based on Time Series Data", 2019
- [6]. N. Yang, X. Jin, T. Su and J. Kong, "Multisource Data Analysis for Stock Prediction," 2018 10th International Conference on Modelling, Identification and Control (ICMIC), Guiyang, 2018

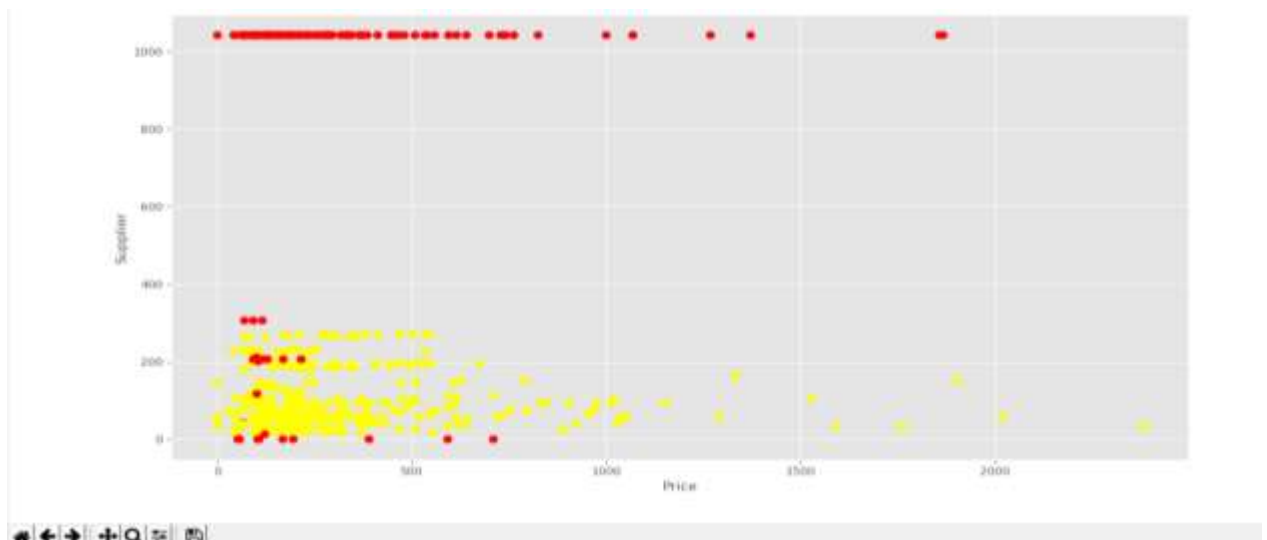
APPENDIX

SCREENSHOTS:



Graph showing Price vs Product

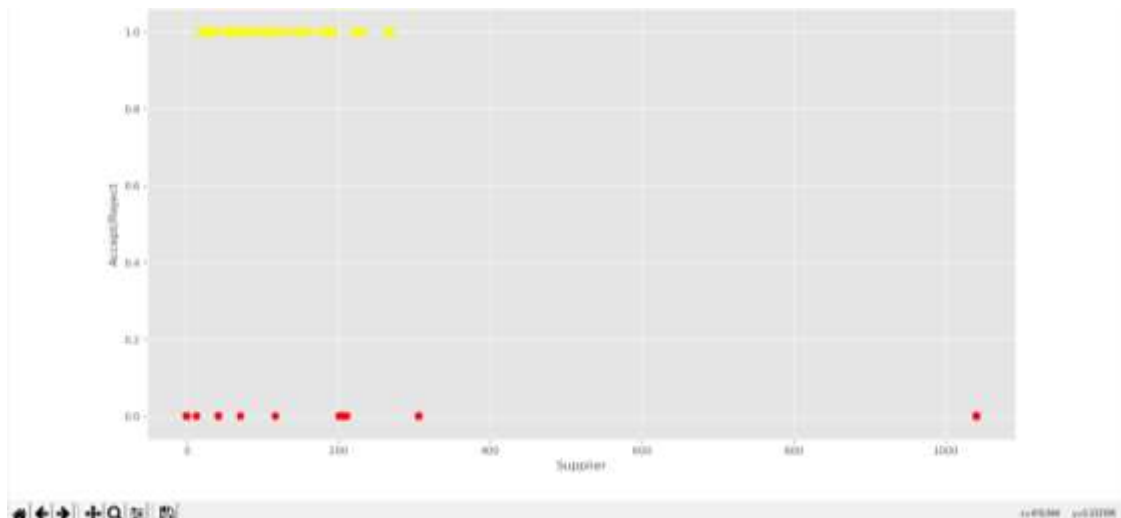
The above graph is visible only after we train our model. Once the model is created (pickle model) the clustering of datasets will be done. This Graph shows the number of products (present in x-axis) related to the price (present in y-axis) based on the datasets that is present within an Excel sheet. The yellow dots denote the positive result and the red dot denotes negative results. For example, product with id 4000 is feasible to buy at the price 1000.



Graph showing Supplier vs Price

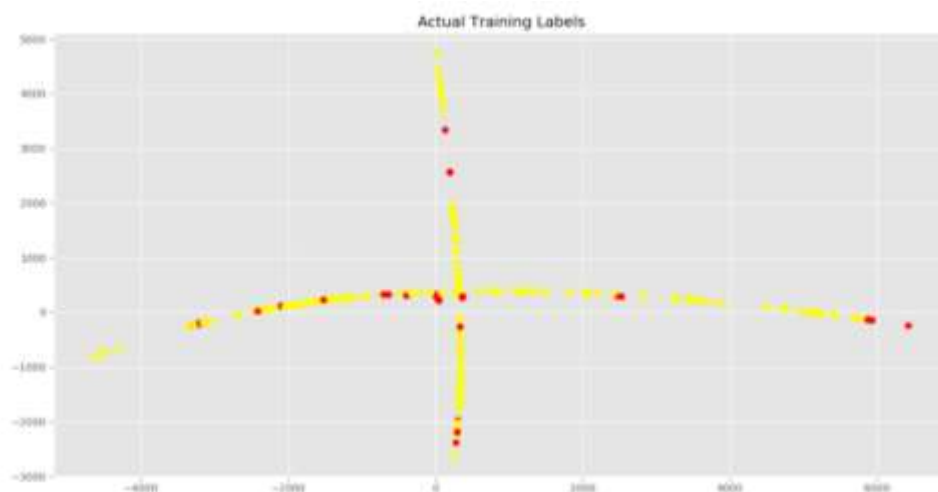
In the above graph, the x-axis shows the price of the product and the y-axis shows the supplier of the product. Here it says that those vendors with the supplier-id more than 1000 are

not feasible to go for. Because the decision of those suppliers are not so good and they yield no profit to the user. Every supplier is having a decision field for some particular product in the datasets of stock.csv. As we improve the datasets we can see a proper scattering of dots.



Graph showing Accept/Reject for the supplier

The above graph shows the relation between supplier and Accept /Reject. Supplier is for x-axis and accept/reject for y-axis. Generally as we improve the dataset we can have a better result over this graph. In this it says those suppliers with different suppliers id stands on accept or reject state. We can notice those supplier with id around 200 stand on reject state for different products. As the datasets improve we can see lot of red and yellow dots within this graph.



Graph showing Training labels

The above graph shows the representation of cluster(SVC). Here we can see that 2 clusters formed. Though some data's are overlapping at the intersection, it has clustered. So 90% of the data looks good from cluster point of view.



Login page

In the above page the user login by giving necessary credentials. User need to enter the valid name and the password. If he/she is a new user , then need to signup by clicking onto the signup button. Once the user log's in , the dashboard page will be displayed.



SignUp Page

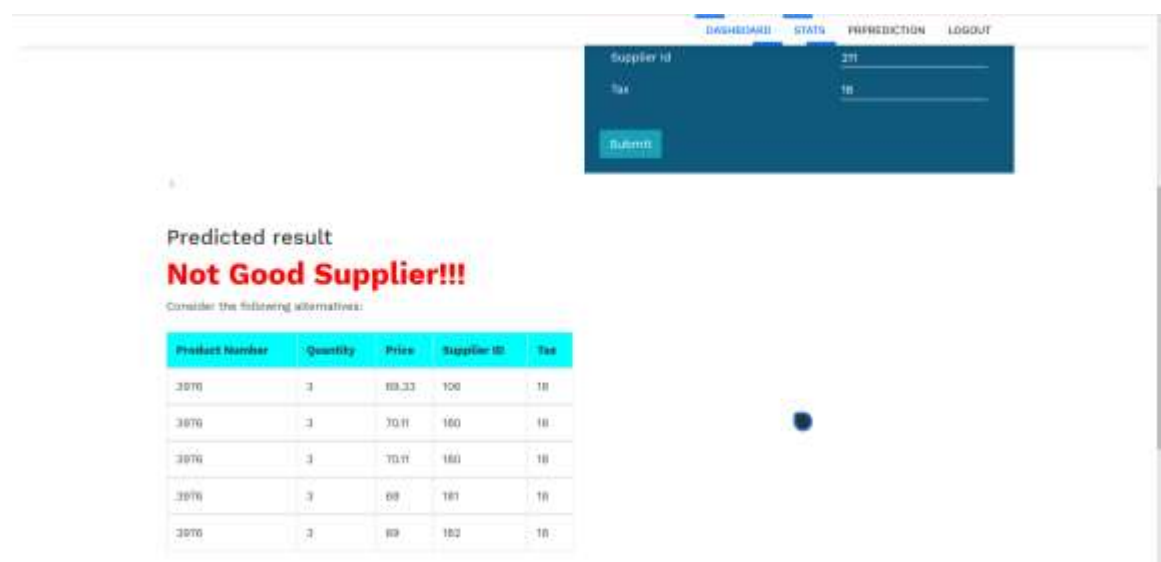
Once the user clicks onto the signup button in the login page, he/she need to enter the name, email, password and re-enter the password. Once all the fields are entered with the valid credentials , the application can be accessed for comparison or for the prediction of Vendors.



The screenshot shows a web application interface with a navigation bar at the top containing 'DASHBOARD', 'STATS', 'PREDICTION', and 'LOGOUT'. The 'PREDICTION' tab is active. Below the navigation bar, there is a 'HOME' button. On the left, there is a circular diagram with various icons representing different aspects of a supply chain or business. On the right, there is a 'Provide your inputs' form with the following fields: Product Number (3076), Quantity (3), Price (68), Supplier Id (100), and Tax (10). A 'Submit' button is located below the form. Below the form, the 'Predicted result' section displays 'Good to Go!!!' in green text, followed by the message 'Your supplier 100 for product 3076 for the Price of 68 looks good!'.

Prediction page for good choice

In the Prediction page the application user is going to enter the test inputs .The test inputs include the product id (product he is willing to purchase from vendor), quantity,price,supplier id and the tax. When user clicks on submit the predicted output is shown.In the above page we can see the product is good to go with that perticular vendor.



The screenshot shows a web application interface with a navigation bar at the top containing 'DASHBOARD', 'STATS', 'PREDICTION', and 'LOGOUT'. The 'PREDICTION' tab is active. Below the navigation bar, there is a 'HOME' button. On the left, there is a circular diagram with various icons representing different aspects of a supply chain or business. On the right, there is a 'Provide your inputs' form with the following fields: Supplier id (20), Price (100), and Tax (10). A 'Submit' button is located below the form. Below the form, the 'Predicted result' section displays 'Not Good Supplier!!!' in red text, followed by the message 'Consider the following alternatives:'. Below this message, there is a table with the following data:

Product Number	Quantity	Price	Supplier ID	Tax
3070	3	103.33	100	10
3070	3	70.11	100	10
3070	3	70.11	100	10
3070	3	69	101	10
3070	3	69	102	10

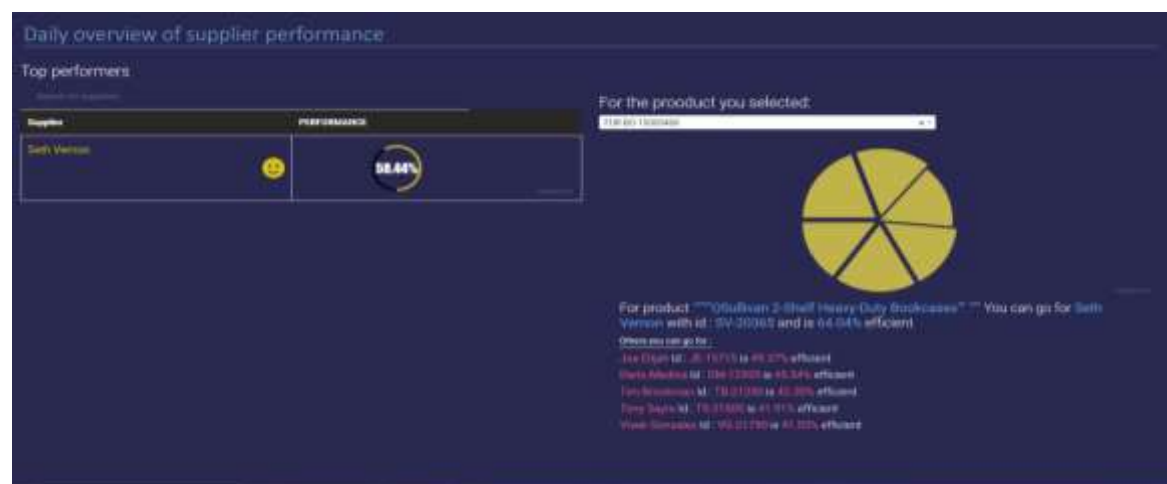
Prediction page showing alternative vendors

In this page the alternative solution is shown. Generally , if the vendor is not feasible to that particular product then the prediction output will show some alternative vendors.



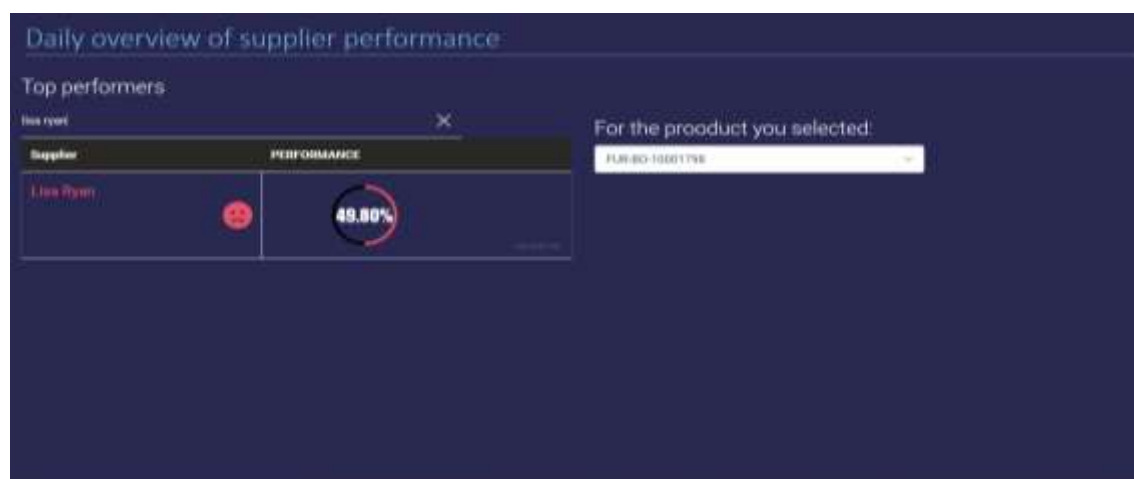
Dashboard page displaying top performers

In the above page of dashboard we can see the top performers i.e those vendors whose products are yielding large amount of profit. In the above show page we can notice that the table showing the list of vendors and there is a textbox present to insert the product id and check for the vendors for particular product.



Page showing the performance of vendors based on product id

In the above page ,we can see the user types some product id in the textbox that is present. It lists the n-number of vendors for that particular product. The top performer for that particular product will be highlighted and shown.



Page showing the Overall profit percentage for the entered name

The user can check the overall profit percentage of the vendor. If the vendor name is entered within the text box, the profit percentage is shown along with an emoji.



Page showing the statistical analysis of the stock

In this page we can see the statistical analysis on the stocks that are present. The profit is shown based on the product categories. There are three line graphs which show the statistics of the number of items sold, total revenue of the stock and total profit over the stock respectively. Each of the graphs shows items sold, revenue of the stock, profit of the stock over the time period which is shown in the x-axis and y-axis. The lines in the graphs which are present above the x-axis are positive in range and those which are below the x-axis are negative in range. The bar graph that is shown in the bottom of the page is to show or to visualize profit by each product category where x-axis represents the profit percentage and y-axis represents the categories of the product. The x-axis shows the highest profit of the product based on categories.