CHRIST
(DEEMED TO BE UNIVERSITY)
BENGALURU · INDIA

ASA SALES FORECASTING

Sachin Davis Mundassery

(1947256)

Under the guidance of

Dr SARAVANAN K N

Specialization project report submitted in partial fulfillment of the requirements of V semester Master of computer Applications, CHRIST (Deemed to be University)

November -2021

CHRIST
(DEEMED TO BE UNIVERSITY)
BENGALURU · INDIA

CERTIFICATE

This is to certify that the report titled ***ASA Sales forecasting*** *is a bona fide record of work done by* ***Sachin Davis Mundassery (1947256)*** *of CHRIST (Deemed to be University), Bangalore, in partial fulfillment of the requirements of V Semester MCA (Computer Applications) during the year 2021.*

Head of the Department **Project Guide**

Valued-by:

Name :
1 . Register Number :
Examination Centre : CHRIST (Deemed to be University)

2 . Date of Exam :

ACKNOWLEDGEMENTS

First of all, we thank the lord almighty for his immense grace and blessings showered on me at every stage of this work. We are greatly indebted to our Head of the Department of Computer Science, Dr. Joy Paulose, CHRIST (Deemed to be University) for providing the opportunity to take up this project as part of my curriculum and we wholeheartedly thank Dr.Tulasi B, Coordinator, Department of computer science for being constant support and for helping us to do our best.

We are deeply indebted to our project guide Dr. Saravanan KN, for his assistance and valuable suggestions as a guide. He made this project a reality.

We express our sincere thanks to Dr. Nismon Rio R, and Dr.Ramesh Chandra Poonia lecturers of the Department of Computer Science, CHRIST (Deemed to be University), for their valuable suggestions during the course of this project. Their critical suggestions helped us to improve the project work.

Acknowledging the efforts of everyone, their chivalrous help in the course of the project preparation and their willingness to corroborate with the work, their magnanimity through lucid technical details lead to the successful completion of my project.

We would like to express our sincere thanks to all our friends, colleagues, parents and all those who have directly or indirectly assisted during this work.

ABSTRACT

The project titled 'ASA Sales Forecasting,' is a solution to an organization's growth. Startups and MNCs' are increasing day by day. Therefore, accurate sales forecasting contributes to a vital impact on business. A layman might wish to see organizational growth as per the current trend. A store owner might want to see the predicted sales if he or she increases the promotion or number of working days. A company might wish to see its current sales trend. Then this project would help them out. Many of such existing sales prediction websites and software are not free, but this one is. The application would be ready to receive three types of people, each having different purposes.An ordinary user can view the trends of a particular store using pictorial representations and sales values. This would help them to decide whether they should make an investment or not.The next group of people would be store owners. They can predict the sales, say for the next week or month, based on specific attributes like promotions and competition distances. This would help the store owners to decide which attribute of their store must be optimized to enhance the sales. The final group would be the company owners, who have multiple stores and wish to host their sales trends on our website. The project is planned to be implemented using Python Django framework as back-end and React.js as front-end. The project uses 'dbsqlite3', the inbuilt database of Django. The major objective of this project is to obtain reliable sales trend prediction, with the support of machine learning techniques to achieve the best possible sales. Technically, the project would exhibit company sales trends using various visualization plots to a non-registered user. For registered users, the project would predict the sales based on specific attributes provided by them. The application could also receive sales data from companies, and once received, its sales trends would be visually represented on the website. A provision for feedback and clarification would be provided to contact the team.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

The project titled 'ASA Sales Forecasting,' is a solution to an organization's growth. Startups and MNCs' are increasing day by day. Therefore, accurate sales forecasting contributes to a vital impact on business. A layman might wish to see organizational growth as per the current trend. A store owner might want to see the predicted sales if he or she increases the promotion or number of working days. A company might wish to see its current sales trend. Then this project would help them out. Many of such existing sales prediction websites and software are not free, but this one is.

1.1 Purpose

The purpose of this project is to help people get a better understanding of the sales trends of various companies. This would help them in taking decisions on investing and buying stocks. The product identities 3 types of audiences namely company owner, store owner and layman. Here companies can host their sales trends, store owners can predict sales and laymen can watch and get insights of the sales of different companies.

1.2 Scope

The project builds an elegant website where users are rendered with various services. It can predict accurate sales considering the seasonality and variation in different attributes of the sales given by the user. It showcases various pictorial representations of the sales trends and enables companies with hosting their sales trends.

1.3 Definitions, Acronyms, and Abbreviations.

- Sales Forecasting - process of estimating future revenue by predicting the amount of product or services a sales unit will sell in the next week, month, quarter, or year. Using machine learning instead of an excel model makes the forecasting process much more traceable. The prediction algorithm can be used with a machine learning environment or a virtual machine and write predictions directly to a database. These predictions could then be distributed to business users through interactive dashboards. This way, there is a single source of truth for the entire company. A centrally run and managed algorithm is more secure, more adaptable and more efficient. Machine learning algorithms would learn the rules that would have to be manually designed in rule-based forecasting. This is done through a process called supervised learning. Supervised learning is the task of learning the relationship between outputs (sales) and inputs (past sales, economic indicator, holiday calendar etc.) Machine Learning algorithms find these relationships by minimizing prediction error – i.e. finding the relationships and parameters that maximize prediction accuracy.

- ReactJS - Free and open-source front-end JavaScript library for building user interfaces or UI components. React is a library for building compassable user interfaces. It encourages the creation of reusable UI components, which present data that changes over time. Lots of people use React as the V in MVC. React abstracts away the DOM from you, offering a simpler programming model and better performance. React can also render on the server using Node, and it can power native apps using React Native. React implements one-way reactive data flow, which reduces the boilerplate and is easier to reason about than traditional data binding.

  JSX − JSX is JavaScript syntax extension. It isn't necessary to use JSX in React development, but it is recommended. Components − React is all about components. You need to think of everything as a component. This will help you maintain the code when working on larger scale projects. Unidirectional data flow and Flux − React implements one-way data flow which makes it easy to reason about your app. Flux is a pattern that helps keep your data unidirectional. License − React is licensed under Facebook Inc. Documentation is licensed under CC BY 4.0.
- Django - Python-based free and open-source web framework that follows the model–template–views (MTV) architectural pattern. It is also called batteries included framework because Django provides built-in features for everything including Django Admin Interface, default database – SQLlite3, etc. When you're building a website, you always need a similar set of components: a way to handle user authentication (signing up, signing in, signing out), a management panel for your website, forms, a way to upload files, etc. Django gives you ready-made components to use and that too for rapid development.
- SQLite - relational database management system (RDBMS) contained in a C library. SQLite is a software library that implements a self-contained, serverless, zero-configuration, transactional SQL database engine. SQLite is one of the fastest-growing database engines around, but that's growth in terms of popularity, not anything to do with its size. The source

code for SQLite is in the public domain. SQLite is an in-process library that implements a self-contained, serverless, zero-configuration, transactional SQL database engine. It is a database, which is zero-configured, which means like other databases you do not need to configure it in your system. SQLite engine is not a standalone process like other databases, you can link it statically or dynamically as per your requirement with your application. SQLite accesses its storage files directly.

1.4 Overview

This is a working document and, as such, is subject to change. In its initial form, it is incomplete by definition, and will require continuing refinement. Requirements may be modified and additional requirements may be added as development progresses and the system description becomes more refined. This information will serve as a framework for the current definition and future evolution of the Sales forecasting website.

2. THE OVERALL DESCRIPTION

The product satisfies the interests of various stakeholders ensuring that it maintains usability and reliability. The services undertaken include visualization, prediction, etc. The product lends a quick responsive neat user interface with accurate functionalities. Granted, humans do not have set patterns of buying and consuming. However, with the backing of datasets spanning years into the past, it is highly possible to identify patterns in sales and consumption. This can be done on an individual basis or focus on a target group or demographic. Machine learning features here because of its ability to mine through years of data to spot patterns and repetitive behavior, which can then be leveraged to forecast sales and demand.

Probability is the foundation of most machine learning algorithms– even in situations of sales forecasting, probability algorithms are the key to envisioning certainty well into the future. The base of probability is that the likelihood of an event occurring can either be 0 or 1, which is to say impossible or certain.

The algorithms take into account all external and internal influential factors that go into the sales process and the likelihood of the said process being a success. This type of ‘what if’ analysis allows sales leaders to understand the impact of these factors on sales numbers and evaluate how to use these insights as levers to have a greater positive impact on sales.

2.1 Product Perspective

The project builds a machine learning model with the historical sales data. This model predicts sales based on certain attributes. Users can vary the input attributes and predict accurate sales. These sales trends are visualized using pictorial representations.

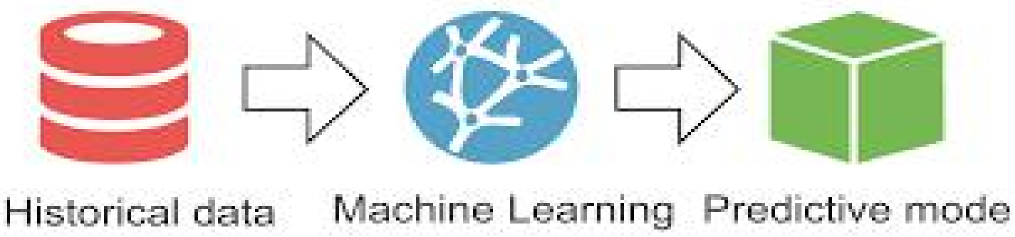


Fig.2.1.1 Product perspective with respect of ML

The machine learning model is plugged into the backend built using Django Framework and rendered into the frontend built using React.

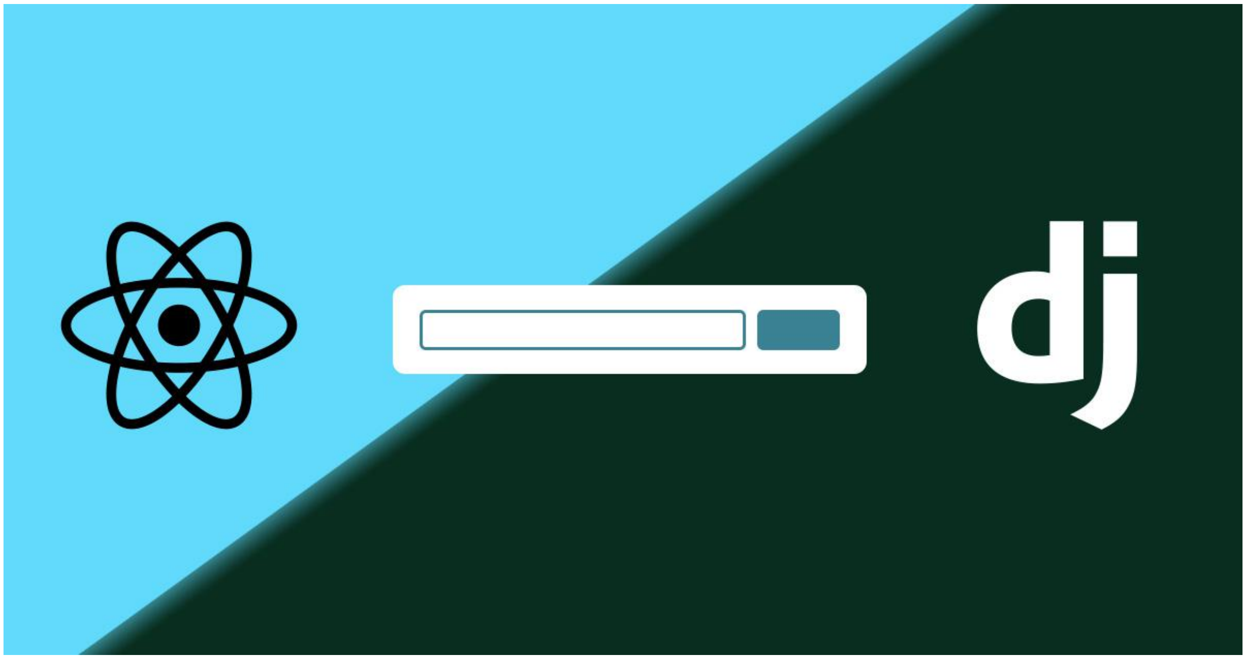


Fig.2.1.2 Product perspective with respect of Web application

Unlike existing websites, this one is completely free and user friendly. The website renders services like enabling users to vary different input attributes and predict sales. Users can view sales trends with search filters. Companies can host their sales trends in the website which helps users to get insight about the company.

2.2 System Interfaces

- Sales visualization page where the user can see and analyze the trend of the particular store at the present and the future stage.
- Login page for the authentication of a class of users other than the ordinary user, who want to evaluate and provide a dataset to the system. It also provides the feature for ‘new user registration’ and ‘forget password’.
- Prediction of sales page where the authenticated user can know the sales for a given set of attribute values.
- The authenticated users also have the option to enter the .csv files of their company to create prediction models and evaluate the growth or trend of each store.
- Download report page where the user can download the results obtained from the system for which they are looking for.

2.3 Hardware Interfaces

- Processor: Intel Core i3 8th gen and up or AMD Ryzen 3 3100 and up.
- Processor speed: 1.5 to 3.40 GHz
- RAM: 4 GB and more

2.4 Software Interfaces

- Operating System : Any
- Database: SQLite.
- Front End: ReactJs, HTML, CSS, Bootstrap.
- Back End: Django.
- Code Editor: Any editor is fine, but preferably VS Code.
- Browser : Chrome or Firefox or Internet Explorer or Microsoft Edge

2.5 Communications Interfaces

There are no specific communication interface requirements. Existing OS and network infrastructure will be leveraged for communication.

2.6 Memory Constraints

Memory constraints will come into play when the size of the database grows to a considerable size.

2.7 Operations

The product shall have operations to protect the database from being corrupted or accidentally altered during a system failure

2.8 Product Functions

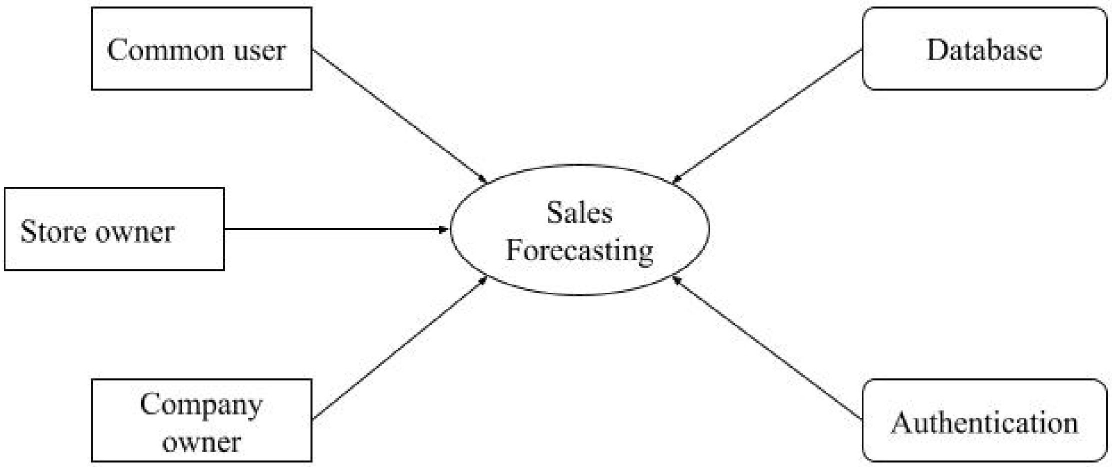


Fig.2.8.1 Product Functions

2.9 User Characteristics

2.9.1 Public

Ordinary users are the ones who can directly access the information provided by the system without the need of login. They can see the visualization of trends of a particular store which they chose and also the position of each store in the market.

2.9.2 Store Owner

Store owners are the next class of users who have to login to perform their tasks. Once the system authenticates the user, they are provided with an option to give input (value of attributes) and thus able to see the future sales of the store.

2.9.3 Company owners

Company owners are the most privileged user class of this system. They have to successfully login to the website in order to use the owner privileges. They have the option to upload new dataset to the system and thus to generate a prediction model for the given dataset.

2.10 Constraints

- Using this system is fairly simple and intuitive. A user familiar with basic browser navigation skills should be able to understand all functionality provided by the system.
- The system should work on most home desktop and laptop computers which support JavaScript and HTML5
- The system will be intended to run on Firefox 4 and above, Google Chrome 10 and above and Internet Explorer 8 and above.
- The system is limited by its operating server in terms of the maximum number of users it can support at a given time.

3. SPECIFIC REQUIREMENTS

An extend the use of the intelligent systems to areas where they were so far neglected due to their insistence on comprehensible models. A separation of the machine learning model selection from model explanation is another significant benefit for expert and intelligent systems. Explanations unconnected to a particular prediction model positively influence acceptance of new and complex models in the business environment through their easy assessment and switching.

A complexity of business dynamics often forces decision-makers to make decisions based on subjective mental models, reflecting their experience. However, research has shown that companies perform better when they apply data-driven decision-making. This creates an incentive to introduce intelligent, data-based decision models, which are comprehensive and support the interactive evaluation of decision options necessary for the business environment, in the terms of business perspective that is for store owners and business stakeholders.

3.1 External Interfaces

In the external interface, only we need to provide an input dataset of the particular store, in which you want to find out the forecasting/analysis/comparison.

3.2 Functions

- The system shall provide sales trends of a particular store/company as per requested by the user also it will gives the future prediction of the same store
- The software shall return the prediction of the sales with regard to particular attributes given by the store owner.
- The software also provides the option to accept new data files in the form of .csv, and the above mentioned functionalities will be performed on the newly created dataset.
- Real-world application on a difficult business problem - sales forecasting.
- Support for validation and updates of decision makers' mental models.
- Uniform and comprehensive explanations for a comparative prediction model.
- Interactive what-if analysis for the evaluation of decision options.

3.3 Performance Requirements

The performance of environmental control will be always proportional to the network speed available and the throughput will be at maximum if the network speed will be at megabits per second.

For experiencing better performance of the system we need to have a good internet facility and uninterrupted resource for accessing the application. The real-world business-to-business sales data set used is publicly available. Interactive what-if analysis for the evaluation of decision options. Uniform and comprehensive explanations for an arbitrary prediction model. For getting better performance we need to provide accurate, relevant dataset in the specified format.

3.4 Logical Database Requirements

Database will be used to generate reports. Initially data such as inputs for the prediction and excel formatted data is taken and stored into the database then reports will be generated accordingly along with the predictions and analysis.

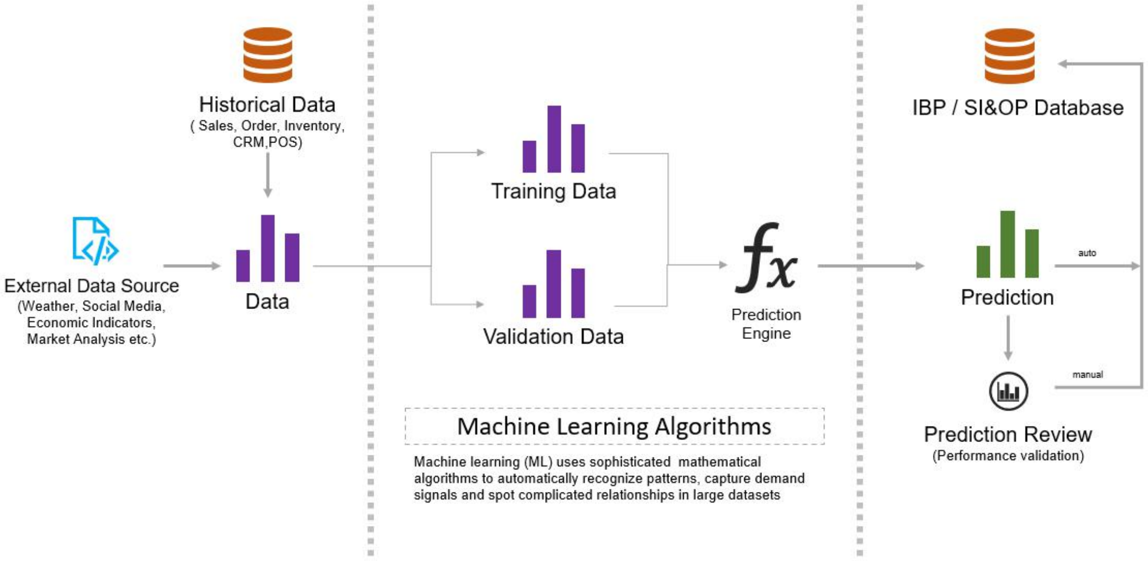

Historical Data
(Sales, Order, Inventory, CRM,POS)
External Data Source
(Weather, Social Media, Economic Indicators, Market Analysis etc.)
Data
Training Data
Validation Data
Prediction Engine
Machine Learning Algorithms
Machine learning (ML) uses sophisticated mathematical algorithms to automatically recognize patterns, capture demand signals and spot complicated relationships in large datasets
IBP / SI&OP Database
Prediction
auto
manual
Prediction Review
(Performance validation)

Fig.3.4.1 Logical database

3.5 Design Constraints

Web pages should be designed using ReactJs and HTML 5.0 transitional standards. Design constraints include the limitations which we are facing while designing the software architecture of ASA sales forecasting.

3.5.1 Standards Compliance

- System should have enough memory and processor.
- All the devices and system should be very reliable.
- All the tools should be compatible with the embedded system on the device.

3.6 Software System Attributes

3.6.1 Reliability

This application can be useful for all the types of stakeholders who just want to know the functionality or those who are into the domain, in both case this will work; we are also addressing all the types of stakeholders through this application, even if you have the business domain knowledge or not irrespective of the primary knowledge you can use our application; this provision is done by user friendly interface; by implementing different login authentication methods.

3.6.2 Availability

The system has a high rate of acceptance if there is a power source and a steady internet connection everything will work smoothly.

3.6.3 Security

Use of hash function can also increase security greatly. The application will also be secured with authentication.

3.6.4 Maintainability

Best performance and precision and so the initial establishment cost will be high. But the maintenance cost will be less once setup, afterwards maintenance cost is less.

3.6.5 Portability

The project can be easily implemented in any other operating system as long as the system requirements are met. The application can be accessed from mobile or tablet also; can easily be established and the operations can easily be implemented.

3.7 Organizing the Specific Requirements

3.7.1 System Mode

The systems provide three different kinds of users, therefore the systems behave differently depending upon the different users and it provides different interfaces for them, by considering all these points, interfaces and performances are dependent on mode of users.

3.7.2 User Class

The systems provide different sets of functions to different classes of users. There are mainly three classes of users. The first class of user is a common man who can view the present and future trend of a particular store/company. The second type of user is the store owners, who can input attributes to the system and thus predicting the sales. The last set of users are the company owners who can give the .csv file to the system and build the prediction model of the same.

3.7.3 Objects

The proposed system interacts with different users, as we mentioned in the previous, therefore the different users are consider here as different objects, each stakeholder can be categorized into the different object as the example of that, consider the scenario where the store owner as the user / stakeholder therefore the store owner will be the one particular type of object, this is how we objects are considered in this project.

3.7.4 Features

Eight major features of the sales forecasting method can be identified with forecasting methods (techniques) to identify key characteristics of a good sales forecasting method. Which includes time horizon, level of detail, stability, pattern of data, type of model, cost, accuracy and ease of application.

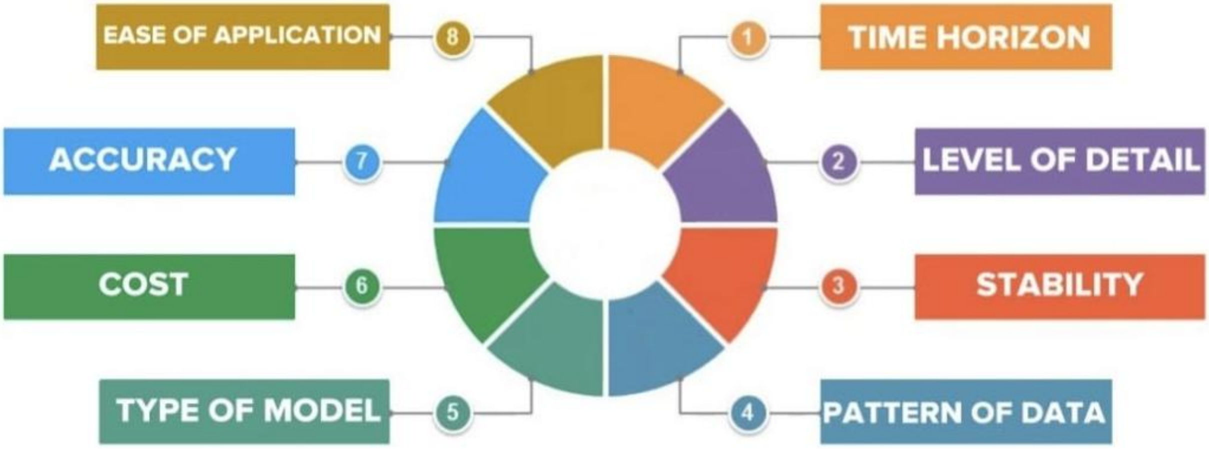


Fig 3.7.4.1 Features

Time Horizon- The length of time over which a decision is being made has a bearing on the appropriate technique to use. The probability of forecasting error generally decreases with an increase in the length of the time horizon.

- Level of Detail - The level of detail needed should match the focus of the decision-making unit in the forecast. For example, production planning must make its decision at the individual product level, whereas the corporate planning department is likely to be happy with aggregate demand forecasts by product categories.
- Stability - Forecasting in situations that are relatively stable over time requires less attention than those that are in constant flux. In stable situations, the existing pattern is assumed to continue in the future and past patterns can be easily extrapolated in future.

- Pattern of Data - Data required to use the underlying-relationships should be available on a timely basis. Each forecasting method is based on an underlying assumption about the data. As different forecasting methods vary in their ability to identify different patterns, it is useful to make the pattern in the data fit with the method that suits it the most.
- Type of Model - Other assumptions are also made in each forecasting technique that must fit the situation under consideration. The technique used should be easily comprehended by the management to give quick meaningful results.
- Cost - Several costs are associated with adopting a forecasting procedure. The variation in costs affects the selection of the forecasting method. There is a need for an economic consideration of balancing the benefits against the extra cost of providing the improved forecasting.
- Accuracy - It is measured by the degree of deviations between past forecasts and current actual performance or present forecasts and future performance. If the likely state comes close to the actual state, it means that the forecast is dependable.
- Ease of Application - Models must be chosen within the abilities of the users to understand them and within the time allowed for using them. This will enable management to properly interpret the results. The simplicity of handling the method matters in the selection of the method.

4. CHANGE MANAGEMENT PROCESS

The Change Management Process is the mechanism used to initiate, record, assess, approve and resolve project changes. Project changes are needed when it is deemed necessary to change the scope, time or cost of one or more previously approved project deliverables. People generally dislike change. It disrupts the familiar and the comfortable. Change forces you to rethink what were previously routine actions that required no new thought. It requires familiarizing yourself with new processes and procedures. In many cases, change requires the acquisition of new skills. It is, however, inevitable. Some people may choose to resist a particular transformation, but resistance to change isn’t usually successful. In most cases, change is unavoidable because the world is dynamic. Since most change occurs to improve a process, a product, or an outcome, it is critical to identify the focus and to clarify goals. Providing clear and open lines of communication throughout the process is a critical element in all change modalities. The methods advocate transparency and two-way communication structures that provide avenues to vent frustrations, applaud what is working, and seamlessly change what doesn't work. After having a clear communication with the customer through phone calls or by mail, planning should be done. As part of the planning process, resource identification and funding are crucial elements. These can include infrastructure, equipment, and software systems. Resistance is a very normal part of change management, but it can threaten the success of a project. Most resistance occurs due to a fear of the unknown. It also occurs because there is a fair amount of risk associated with change – the risk of impacting dependencies, return on investment risks, and risks associated with allocating budget to something new. Anticipating and preparing for resistance by arming leadership with tools to manage it will aid in a smooth change lifecycle. When managing a change through its lifecycle, it’s important to recognize the success of teams and individuals involved. This will help in the adoption of both change management processes as well as adoption of the change itself. Once the change is made, revise and then review it from the customer. And the final project is delivered to the customer as per his/her requirements.

5. DESIGN SPECIFICATION

5.1 ARCHITECTURAL DESIGN

The 3-tier schema is an extension of the 2-tier architecture. 3-tier architecture has the following layers ; Presentation layer (your PC, Tablet, Mobile, etc.),Application layer (server) and Database Server. This DBMS architecture contains an Application layer between the user and the DBMS, which is responsible for communicating the user's request to the DBMS system and send the response from the DBMS to the user.The application layer (business logic layer) also processes functional logic, constraint, and rules before passing data to the user or down to the DBMS.Three tier architecture is the most popular DBMS architecture. Fig. 5.1.2 Architecture Diagram depicts the architectural design of ASA sales forecasting which contains all the steps.

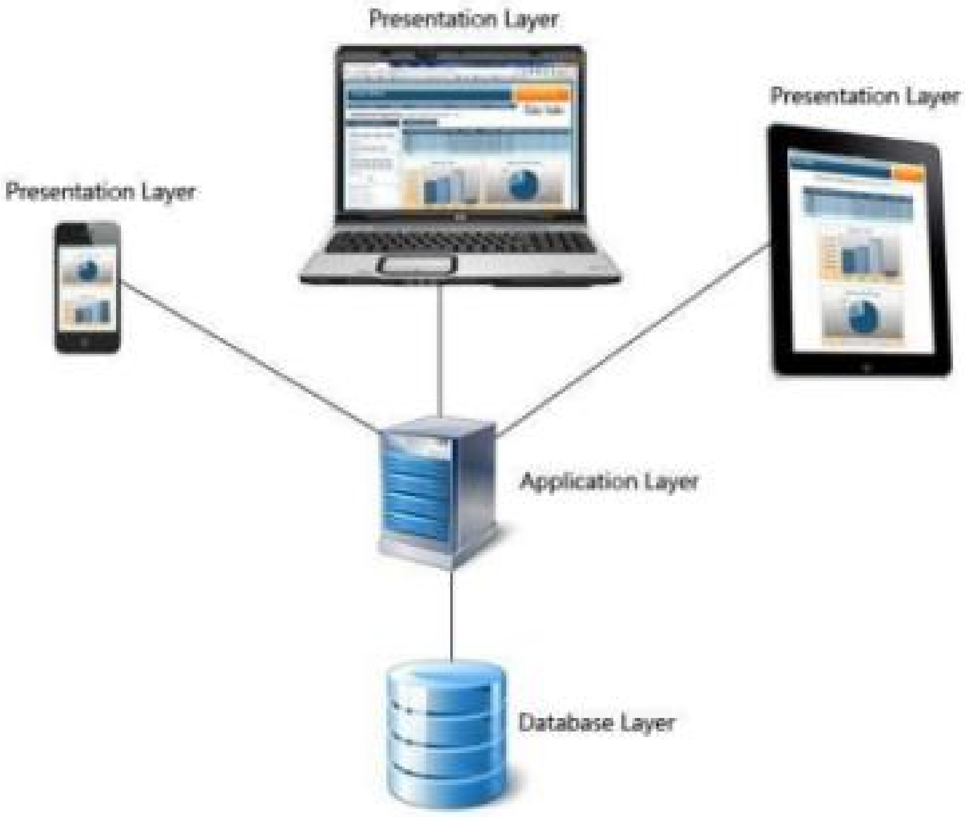


Fig. 5.1 3-tier Architecture Diagram

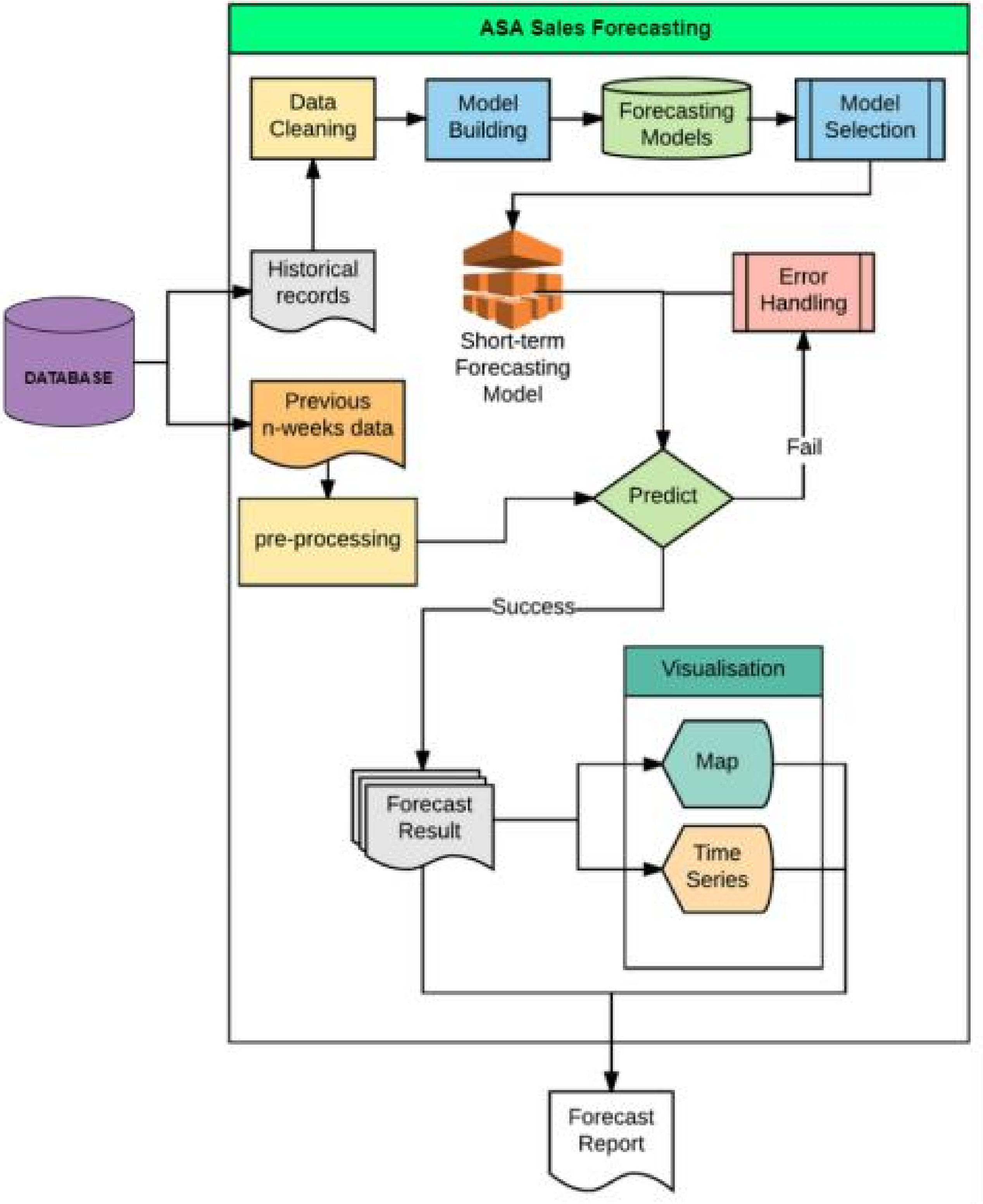


Fig. 5.1.2 Architecture Diagram

5.2 DATA FLOW DIAGRAM

The context diagram also called the Level 1 DFD (Fig 5.2 level 0 DFD) depicts the overall data flow in the system. The Figure (Fig 5.3 level 1 DFD) depicts the operations on the system and the more detailed details of the data flow has been included in the Figure (Fig 5.4 level 2 DFD & Fig 5.5 level 2). These figures constitute the data flow representation in the system.

5.2.1 CONTEXT DIAGRAM (LEVEL 0 DFD)

In the level 0 of DFD we have the three different users and they request the services from our system ASA sales forecasting and according to their requirements the system interacts and responds to the users.The application acts as a middle layer to facilitate the interaction between the users and the system as a whole.

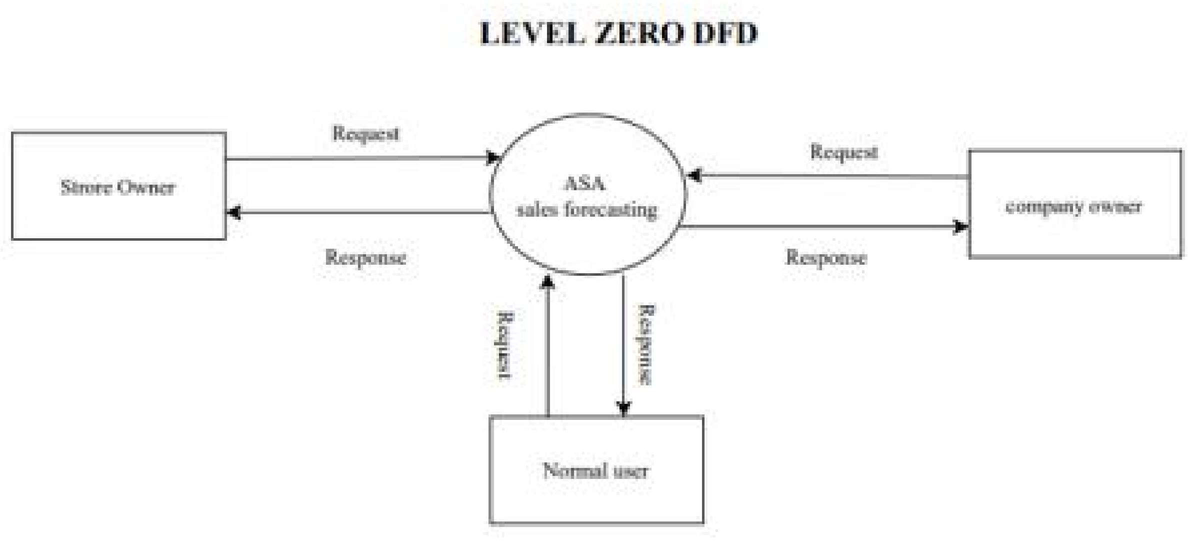


Fig. 5.2 level 0 DFD

5.2.2 LEVEL 1 DFD

In level 1 of the DFD we have the various operations that a store owner,normal user and company owner can perform. The CSV file details,predicting accuracy,and visualization are the operations that perform over this level.

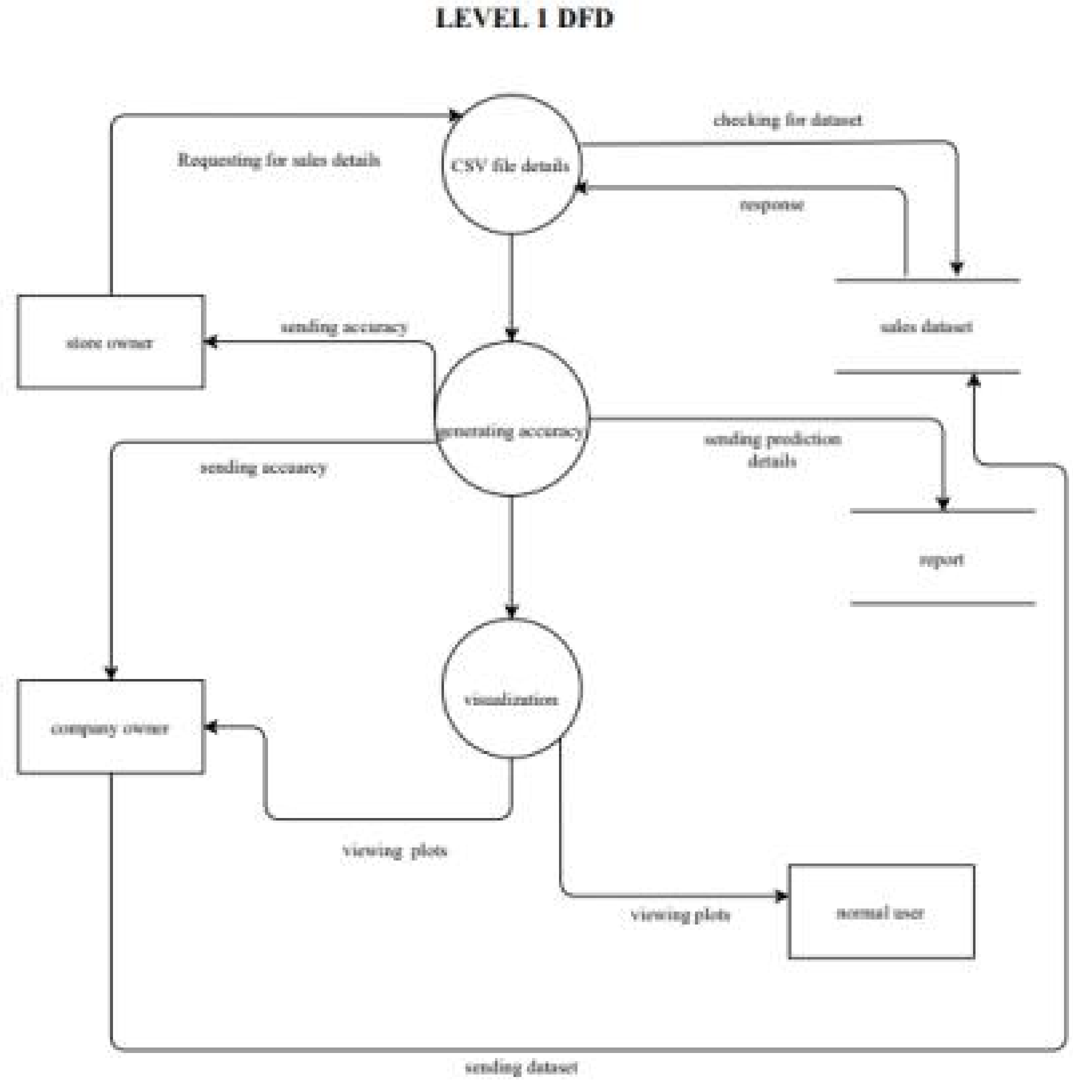


Fig. 5.3 level 1 DFD

5.2.3 LEVEL 2 DFD FOR STORE OWNER

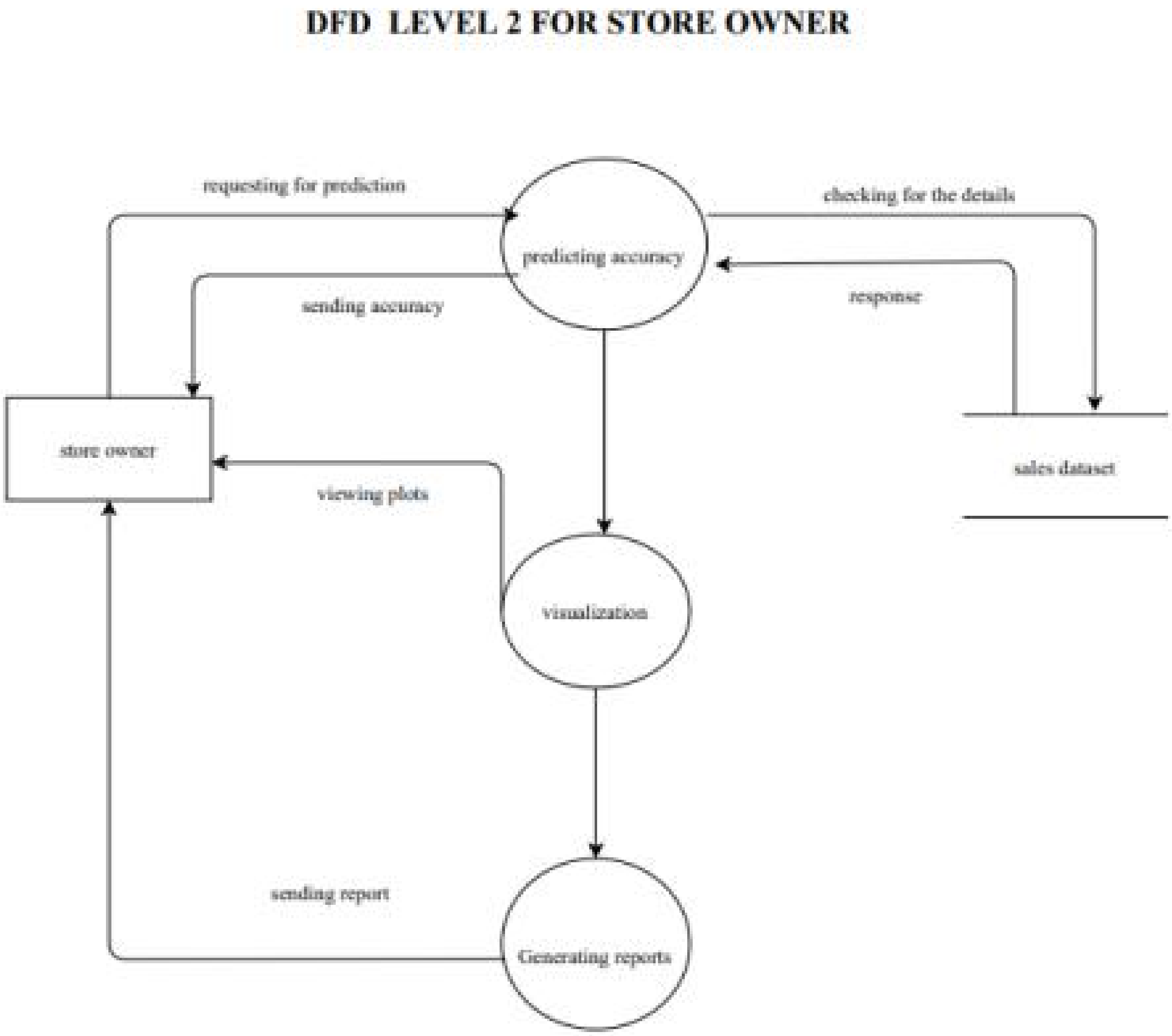


Fig 5.4 level 2 DFD

5.2.4 LEVEL 2 DFD FOR COMPANY OWNER

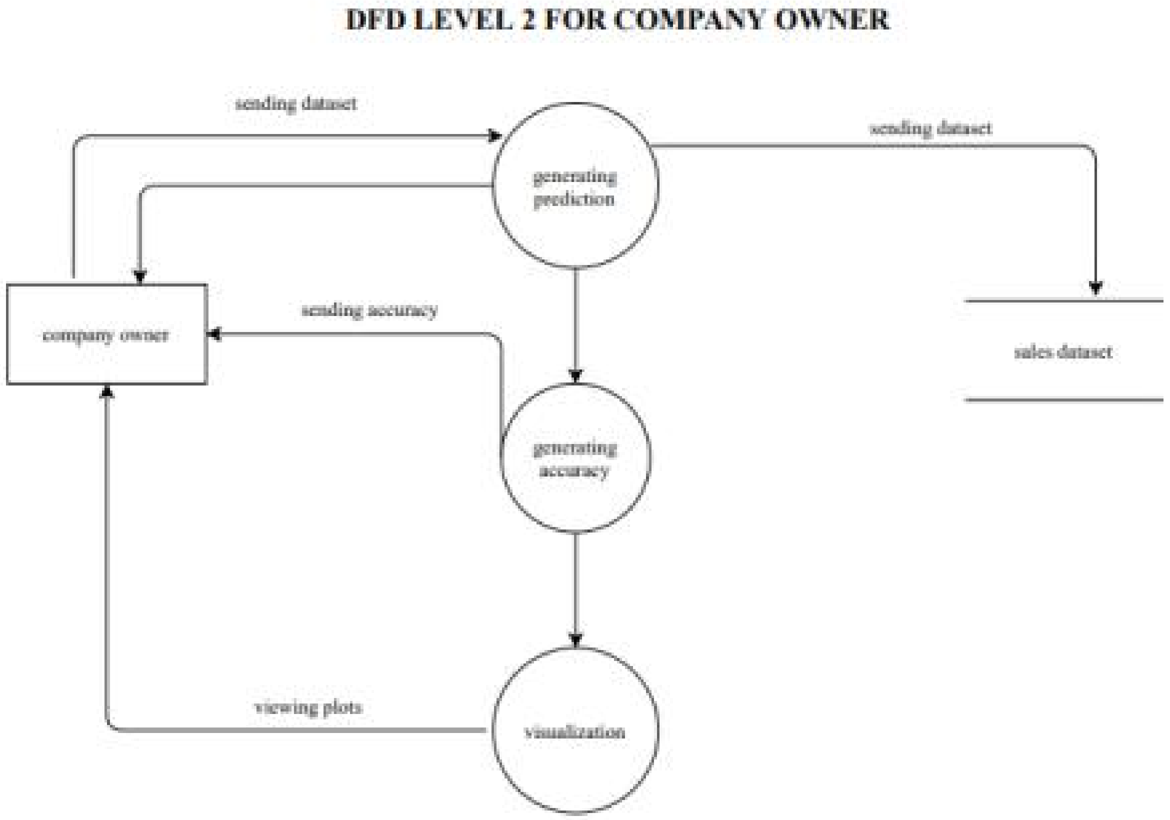


Fig 5.5 level 2 DFD

5.3 ENTITY RELATIONSHIP DIAGRAM

The entity relationship diagram describes interrelated things in the project. A basic ER model is composed of entity types and specifies relationships that can exist between instances of those entity types. An entity relationship diagram shows the relationships of entity sets stored in a database. An entity in this context is an object, a component of data. An entity set is a collection of similar entities. These entities can have attributes that define its properties.

The entity relationship diagram (Fig. 5.6 ER Diagram), represents the inter relativity of the different project components and how it is related together. the different modules of the project. the entity relationship diagram helps in the design and development of the database.

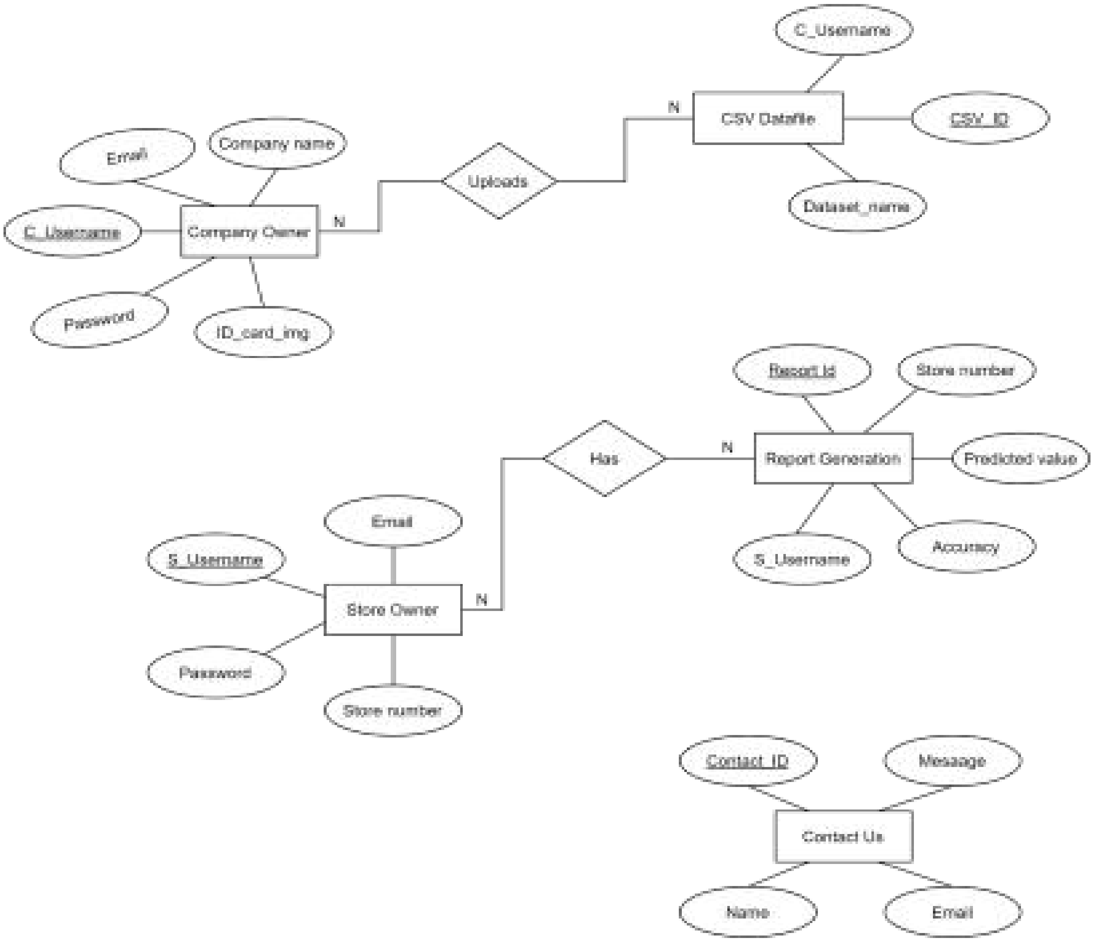


Fig. 5.6 ER Diagram

5.4 DATABASE DESIGN DIAGRAM

Database design illustrates a detailed data model of a database also known as the database schema. It shows the various tables that are in the system and the relationships between those tables. In our project a total five tables are presented the relationship between them is mentioned using primary key (PK) and foreign key (FK) in the database design.

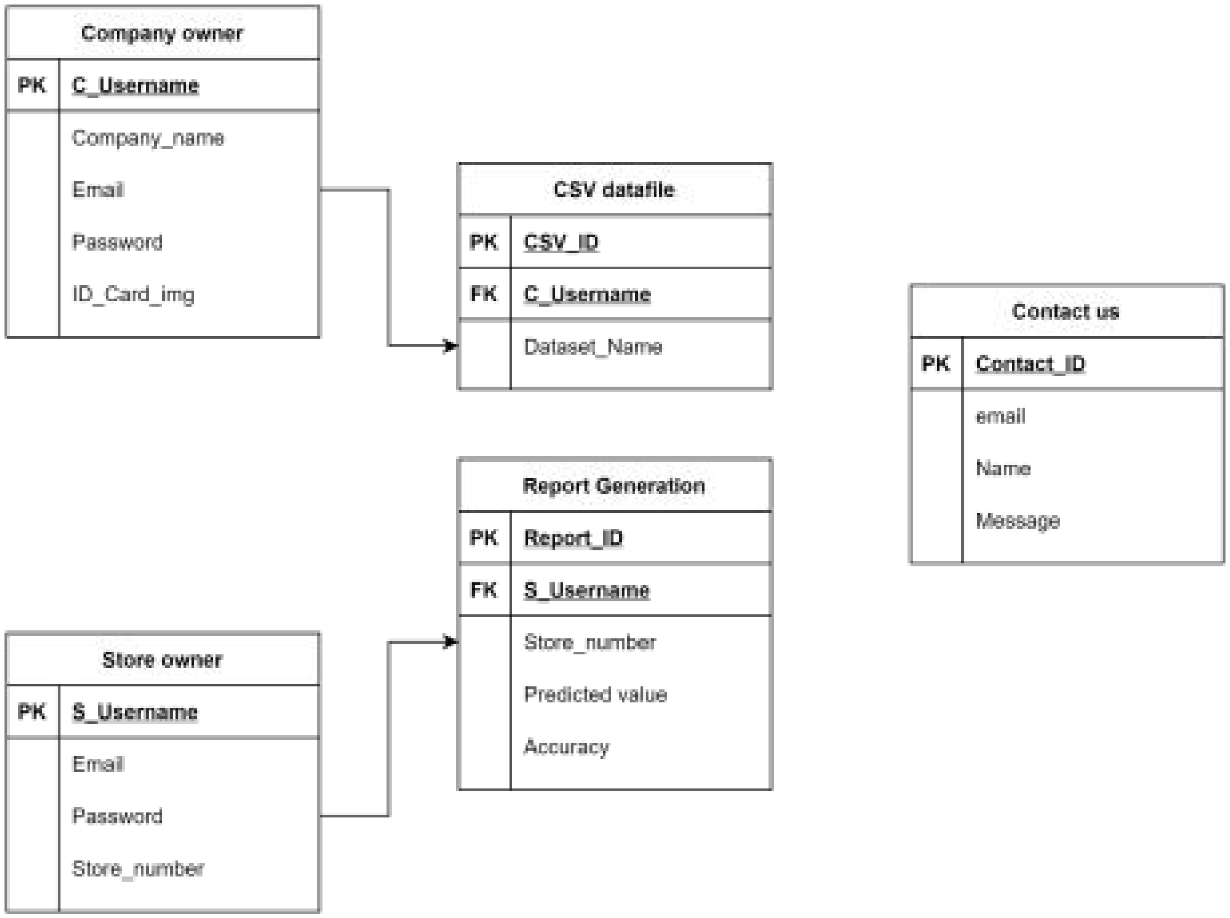
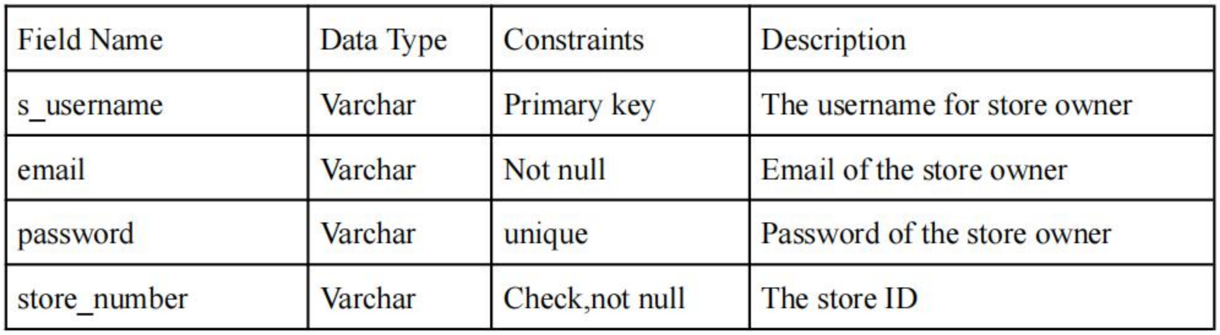


Fig.5.6 Database design Diagram

5.5 TABLES

In our project a total five tables are presented ,those are listed below.

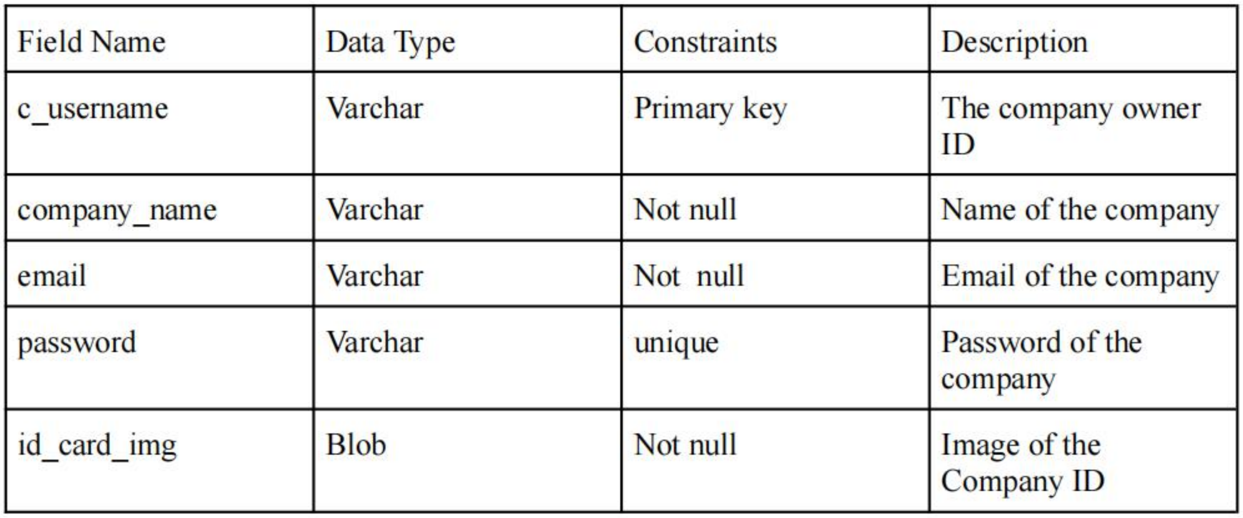
5.5.1 STORE OWNER



Field Name	Data Type	Constraints	Description
s_username	Varchar	Primary key	The username for store owner
email	Varchar	Not null	Email of the store owner
password	Varchar	unique	Password of the store owner
store_number	Varchar	Check,not null	The store ID

Table 5.5.1 store owner

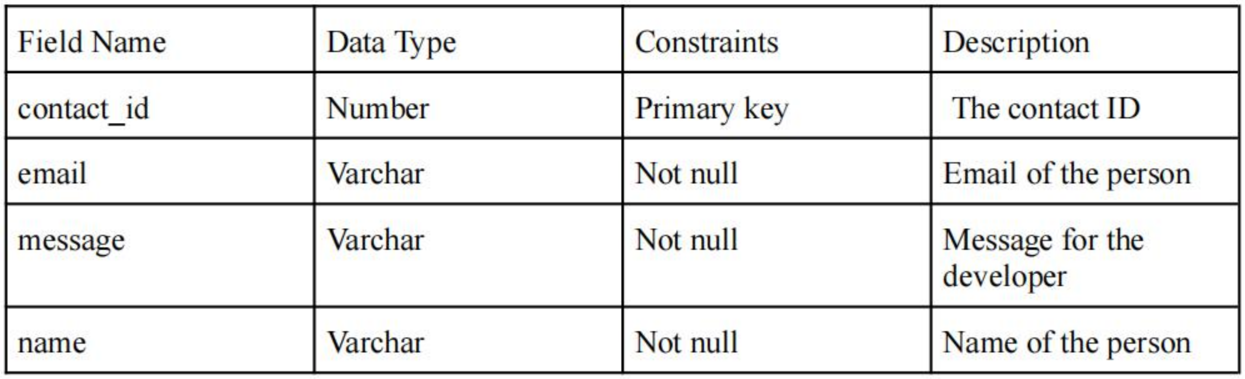
5.5.2COMPANY OWNER



Field Name	Data Type	Constraints	Description
c_username	Varchar	Primary key	The company owner ID
company_name	Varchar	Not null	Name of the company
email	Varchar	Not null	Email of the company
password	Varchar	unique	Password of the company
id_card_img	Blob	Not null	Image of the Company ID

Table 5.5.2 company owner

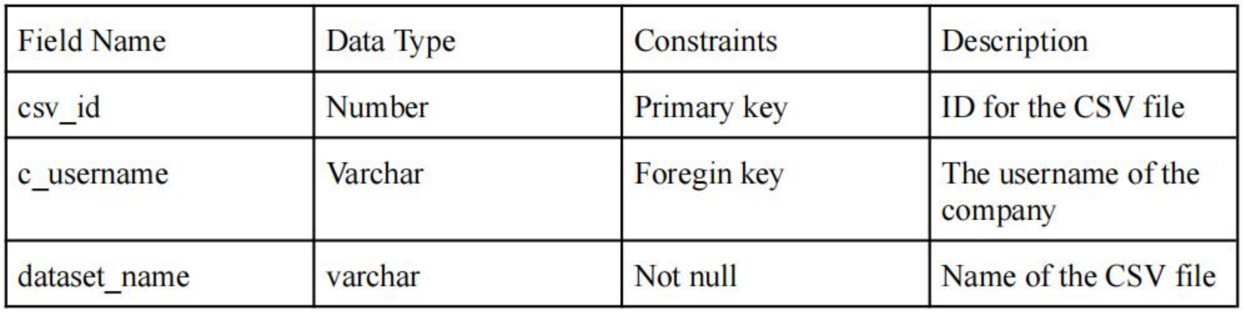
5.5.3 CONTACT US



Field Name	Data Type	Constraints	Description
contact_id	Number	Primary key	The contact ID
email	Varchar	Not null	Email of the person
message	Varchar	Not null	Message for the developer
name	Varchar	Not null	Name of the person

Table 5.5.3 contact us

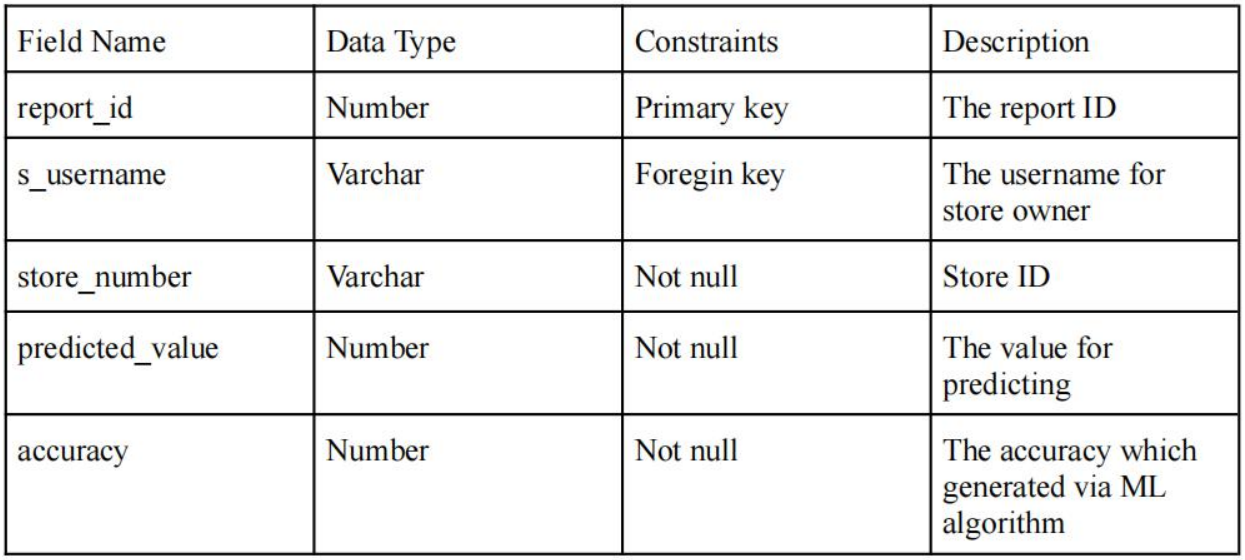
5.5.4 CSV DATAFILE



Field Name	Data Type	Constraints	Description
csv_id	Number	Primary key	ID for the CSV file
c_username	Varchar	Foregin key	The username of the company
dataset_name	varchar	Not null	Name of the CSV file

Table 5.5.4 csv datafile

5.5.5 REPORT GENERATION



Field Name	Data Type	Constraints	Description
report_id	Number	Primary key	The report ID
s_username	Varchar	Foregin key	The username for store owner
store_number	Varchar	Not null	Store ID
predicted_value	Number	Not null	The value for predicting
accuracy	Number	Not null	The accuracy which generated via ML algorithm

Table 5.5.5 report generation

6.IMPLEMENTATION DETAILS

6.1 Code

CompanyRegister.jsx

```
import React, { useState } from "react";

import { ListGroupItem } from "react-bootstrap";

import CompanyLogin from "./CompanyLogin";

// import loginImg from "../../login.svg";

const CompanyRegister = (props) => {

        const [registerDetails, setRegisterDetails] = useState({

                username: "",

                email: "",

                password: "",

                companyname: "",

                companyidcard: "",

        });

        // const [registered, setRegistered] = useState(false)

        const handleChange = (e) => {

                e.preventDefault();

                const input = e.target.name;

                var value;

                if (e.target.name === "companyidcard") {
```

```
            value = e.target.files[0];
        } else {
            value = e.target.value;
        }

        setRegisterDetails({ ...registerDetails, [input]: value });
    };

    const handleValidation = () => {
        var usernameRegex = /^[a-zA-Z\-0-9_]+$/;
        if (!registerDetails.username.match(usernameRegex)) {
            alert("Username should not contain numbers, or special characters");
            return false;
        }

        var emailRegex =
            /[a-zA-Z0-9_]+[\.]?([a-zA-Z0-9]+)?[\@][a-z]{3,9}[\.][a-z]{2,5}/;
        if (!registerDetails.email.match(emailRegex)) {
            alert("email cannot contain special characters other thand @ and _");
            return false;
        }
        var passwordRegex =
            /^(?=.*[0-9])(?=.*[!@#$%^&*])[a-zA-Z0-9!@#$%^&*]{7,15}$/;
        if (!registerDetails.password.match(passwordRegex)) {
            alert(
                "password must be of 7 to 15 characters which contain at least one numeric digit and a special character"
            );
            return false;
        }

        if (registerDetails.companyidcard.length == 0) {
            alert("you need to upload the image of your company id card");
            return false;
        }
        return true;
    };

    const handleSubmit = async (e) => {
        e.preventDefault();

        if (handleValidation()) {
```

```
        const formData = new FormData();


        formData.append("type", "Register");

        formData.append("username", registerDetails.username);

        formData.append("companyname", registerDetails.companyname);

        formData.append("email", registerDetails.email);

        formData.append("password", registerDetails.password);

        formData.append("companyidcard", registerDetails.companyidcard);


        const response = await fetch("http://127.0.0.1:8000/company", {

            method: "POST",

            body: formData,

        });

        const result = await response.json();

        if(result["response"]==true)

            {

                alert("Submitted");

            }

    } else {

        console.log("false");

    }
};


return (
    <>
        <div className="base-container" ref={props.containerRef}>
            <h1>
                <strong>Company</strong>
            </h1>
            <div className="header">Register</div>
            <div className="content">
                <div  className="image">{/*  <img  src={loginImg}  /> */}</div>
                <div className="form">
                    <div className="form-group">
                        <input
                            type="text"
                            name="username"
                            placeholder="username"
                            onChange={handleChange}
                        />
                    </div>
                    <div className="form-group">
                        <input
                            type="text"
                            name="companyname"
                            placeholder="company name"
                            onChange={handleChange}
                        />
                    </div>
                    <div className="form-group">
                        <input
                            type="text"
                            name="email"
                            placeholder="email"
                            onChange={handleChange}
                        />
                    </div>
                    <div className="form-group">
                        <input
                            type="password"
                            name="password"
                            placeholder="password"
                            onChange={handleChange}
```

```
                            />
                        </div>
                        <div className="form-group">
                            <h6>Upload your company ID card.</h6>
                            <input
                                type="file"
                                name="companyidcard"
                                placeholder="Upload your company id"
                                accept="image/*"
                                onChange={handleChange}
                            />
                        </div>
                    </div>
                </div>
                <div className="footer">
                    <button
                        type="button"
                        className="loginRegisterbtn"
                        onClick={handleSubmit}
                    >
                        Register
                    </button>
                </div>
            </div>
        </>
    );
};


export default CompanyRegister;
```

companyLogin.jsx

```
import React, { useState } from "react";
// import Company from "./Company";


const CompanyLogin = (props) => {
    const [loginDetails, setLoginDetails] = useState({
        username: "",
        password: "",
    });
    const [loggedinUser, setLoggedinUser] = useState({
        userLoggedIn: "",
    });
    const [loginSuccessful, setLoginSuccessful] = useState(false);
    const handleChange = (e) => {
        const input = e.target.name;
        const value = e.target.value;

        setLoginDetails({ ...loginDetails, [input]: value });
    };

    const handleSubmit = async (e) => {
        e.preventDefault();
        const formData = new FormData();

        formData.append("type", "Login");
        formData.append("username", loginDetails.username);
        formData.append("password", loginDetails.password);



        const response = await fetch("http://127.0.0.1:8000/company", {
            method: "POST",
            body: formData,
        });
        const result = await response.json();
```

```
        console.log("%%%%%%%%%%5", result);


        if (result["authenticationStatus"] == true) {

            props.loginStatusDetails.changeLoginStatus(true);

            const userLoggedIn = result["userSignedIn"];

            setLoggedinUser({

                ...loggedinUser,

                [loggedinUser.userLoggedIn]: result["userSignedIn"],

            });

            console.log("companylogin.jsx", userLoggedIn);

            // console.log("------", loggedinCompany.companyLoggedIn);

            props.userNameLoggedin.changeUserLoggedin(userLoggedIn);

        } else {

            alert("Wrong credentials !");

        }

    };


    return (

        <>

            <div className="base-container" ref={props.containerRef}>

                <div className="header">Login</div>
```

```
<div className="content">

    <div  className="image">{/*  <img  src={loginImg}  /> */}</div>

    <div className="form">

        <div className="form-group">

            <input

                type="text"

                name="username"

                placeholder="username"

                onChange={handleChange}

            />

        </div>

        <div className="form-group">

            <input

                type="password"

                name="password"

                placeholder="password"

                onChange={handleChange}

            />

        </div>

    </div>

</div>
```

```
                                        <div className="footer">
                                                  <button
                                                            type="button"
                                                            className="loginRegisterbtn"
                                                            onClick={handleSubmit}
                                                  >
                                                            Login
                                                  </button>
                                        </div>
                              </div>
                    </>
          );
};


export default CompanyLogin;
```

Team.js

```
import { TeamMembers } from "./data";
import Card from "react-bootstrap/Card";
import Image from "react-bootstrap/Image";
const Team = () => {
          return (
        <>
            <h3 style={{ textAlign: "center" }}>Team</h3>
            <main id="team">
                {TeamMembers.map((team) => {
                    const { id, name, image } = team;

                    return (
                        <Card key={id} className="team-card">
                            <Image src={image} roundedCircle
className="team-img" />
                            <Card.Body>
                                <Card.Title className="text-
center">{name}</Card.Title>
                            </Card.Body>
                        </Card>
                    );
                })}
            </main>
        </>
    );
};
export default Team;
```

7.RESULTS AND DISCUSSION

7.1 USER INTERFACE DESIGN

In the user interface module, we included the screenshots of various interfaces in which different stakeholders can view and interact via that area. We provided the authorization to them, according to that stakeholders could interact with them. The front end part (that is fig 7.1, fig 7.2 & fig 7.3) is the standard interface in which all the stakeholders can view and other screenshots indicating the specific functionalities of our various stakeholders. Along with the screenshot, the specific figure names indicate its unique functionality. User interface screenshots are attached below.

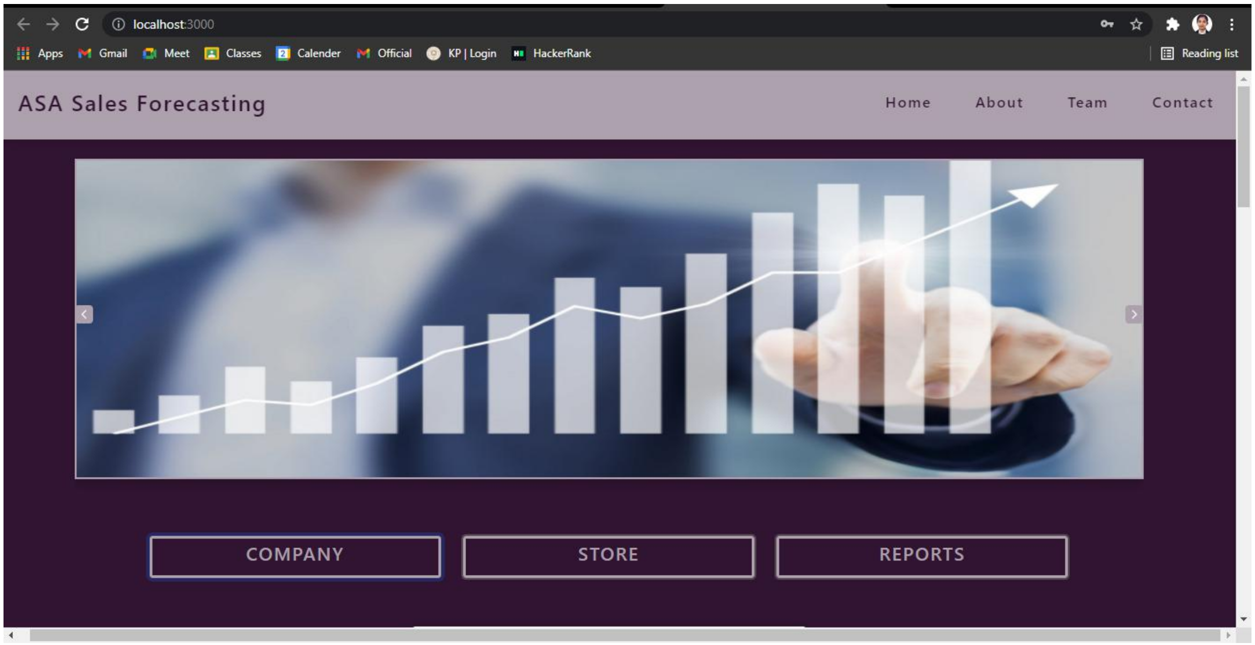


Fig. 7.1.1Frontend

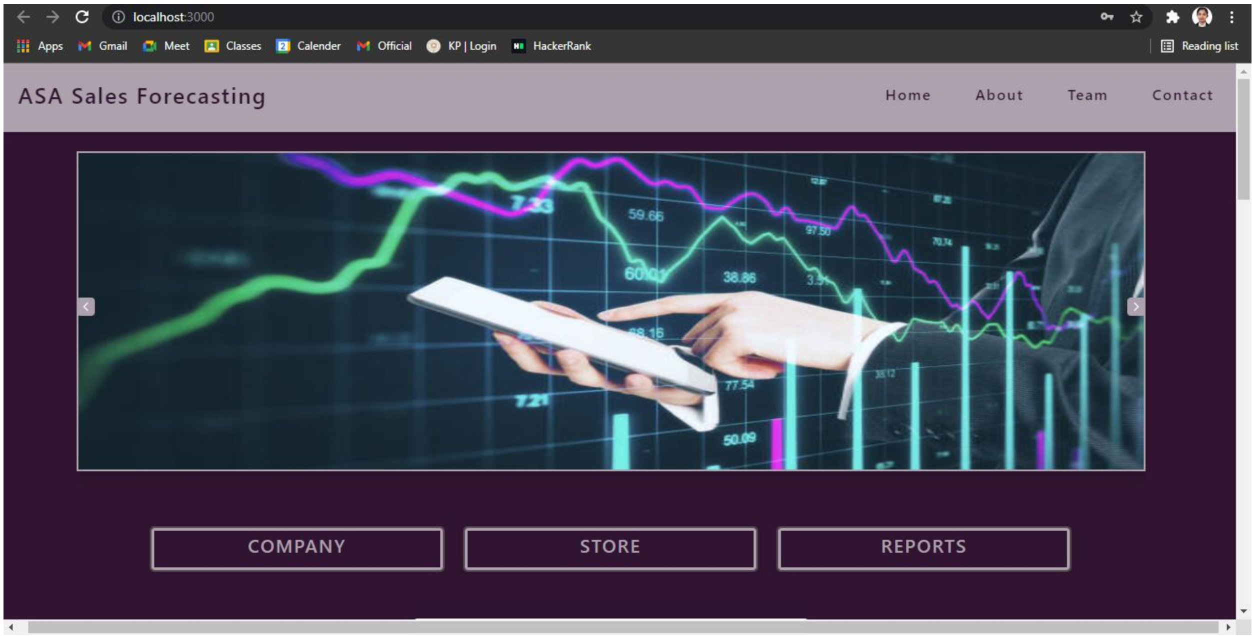


Fig. 7.1.2 Frontend

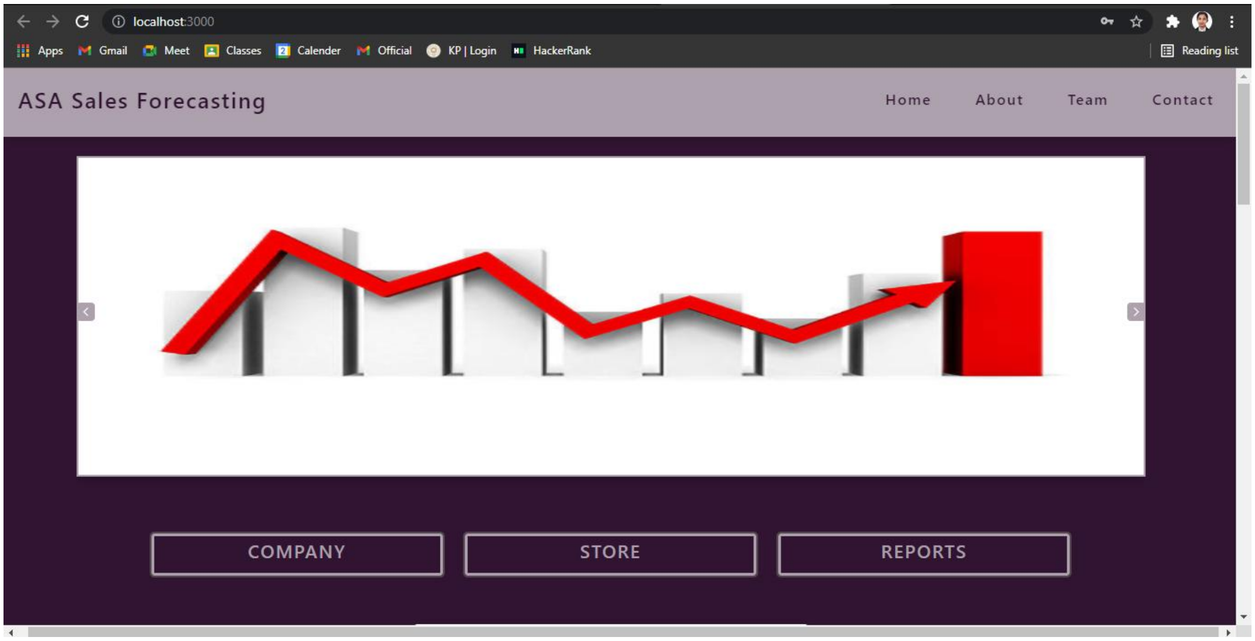
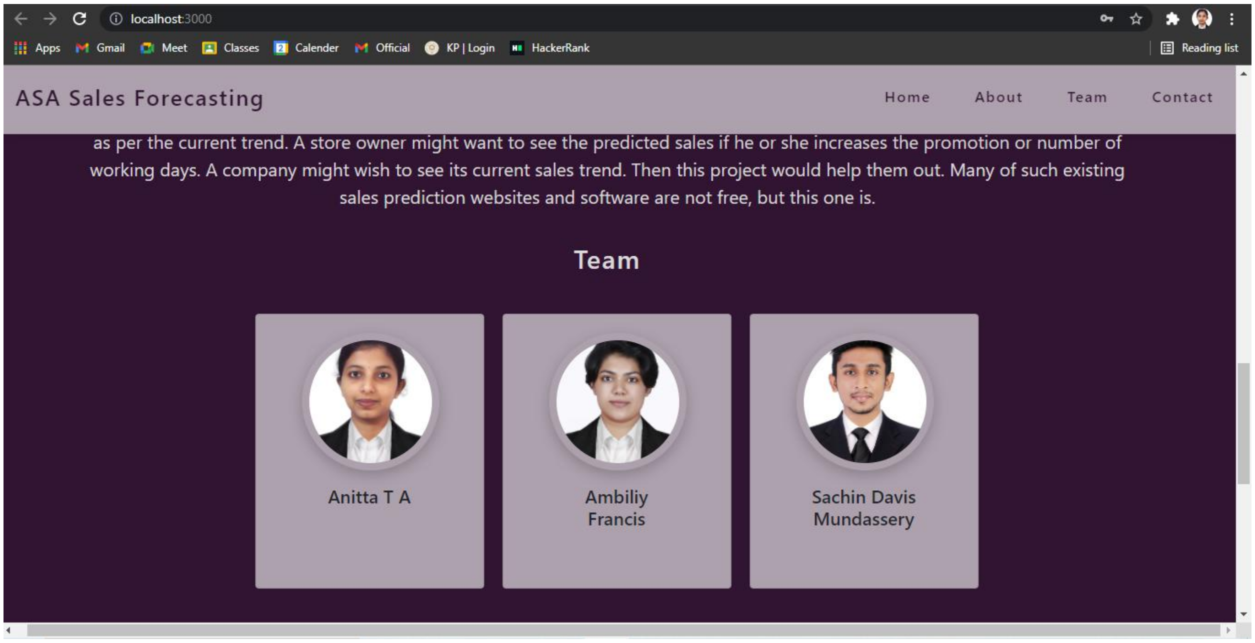


Fig. 7.1.3 Frontend



as per the current trend. A store owner might want to see the predicted sales if he or she increases the promotion or number of working days. A company might wish to see its current sales trend. Then this project would help them out. Many of such existing sales prediction websites and software are not free, but this one is.

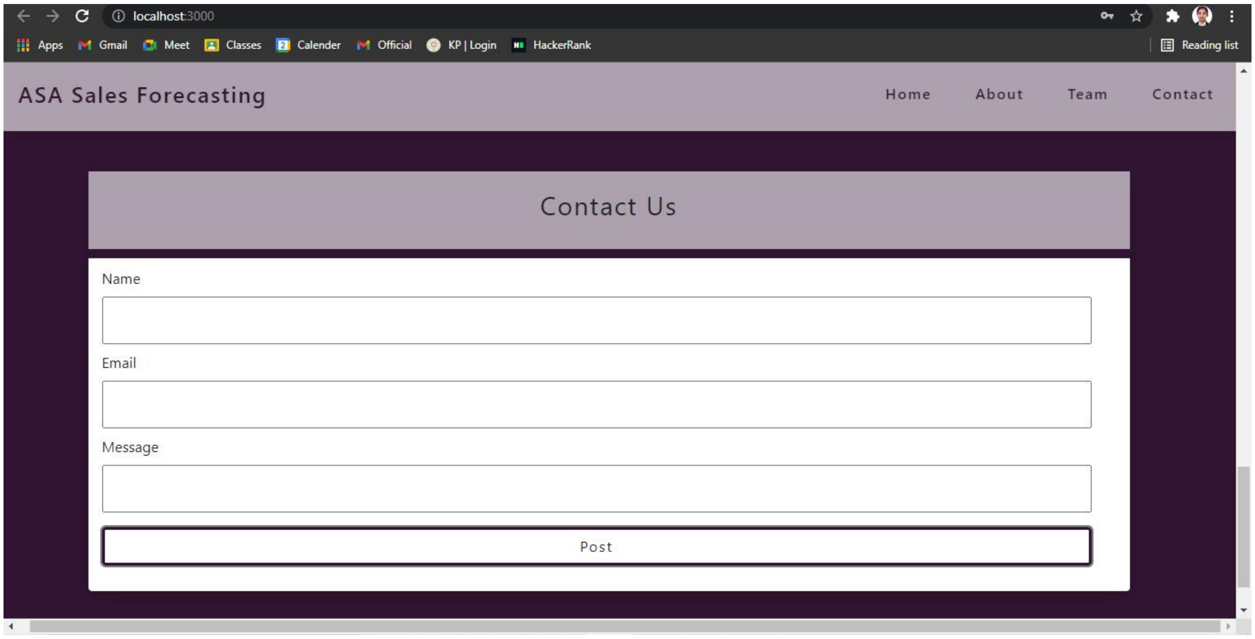
Team

Anitta T A

Ambiliy Francis

Sachin Davis Mundassery

Fig. 7.1.4 Frontend



Contact Us

Name

Email

Message

Post

Fig. 7.2 Contact us

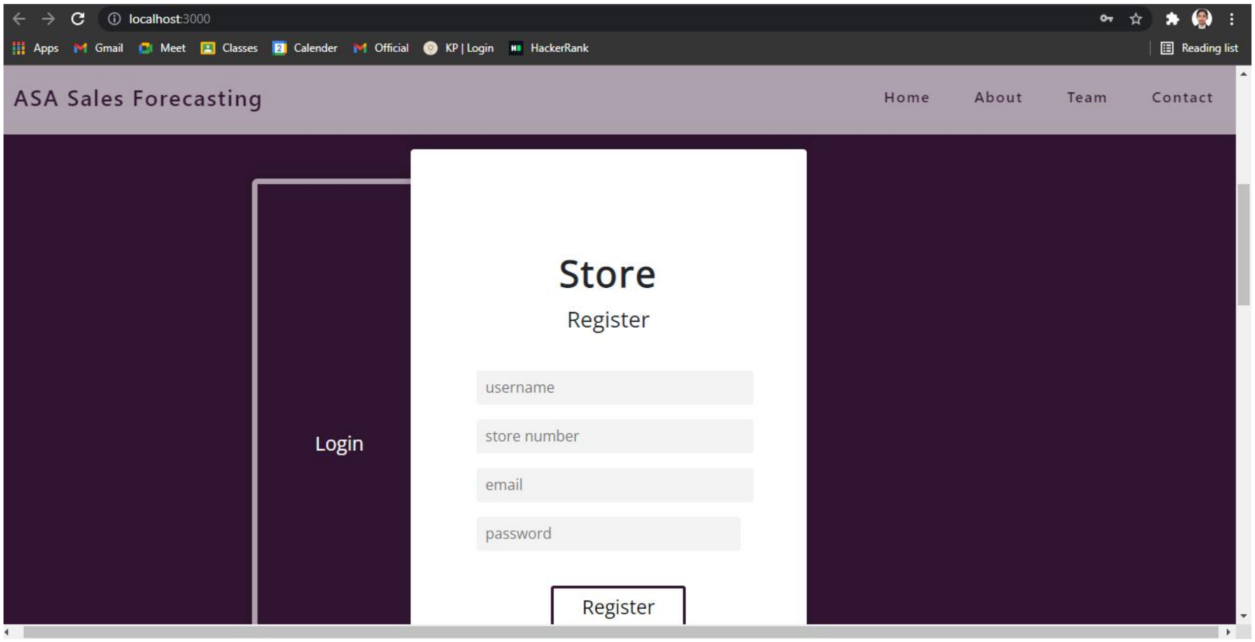


Fig. 7.3 Register for store owner

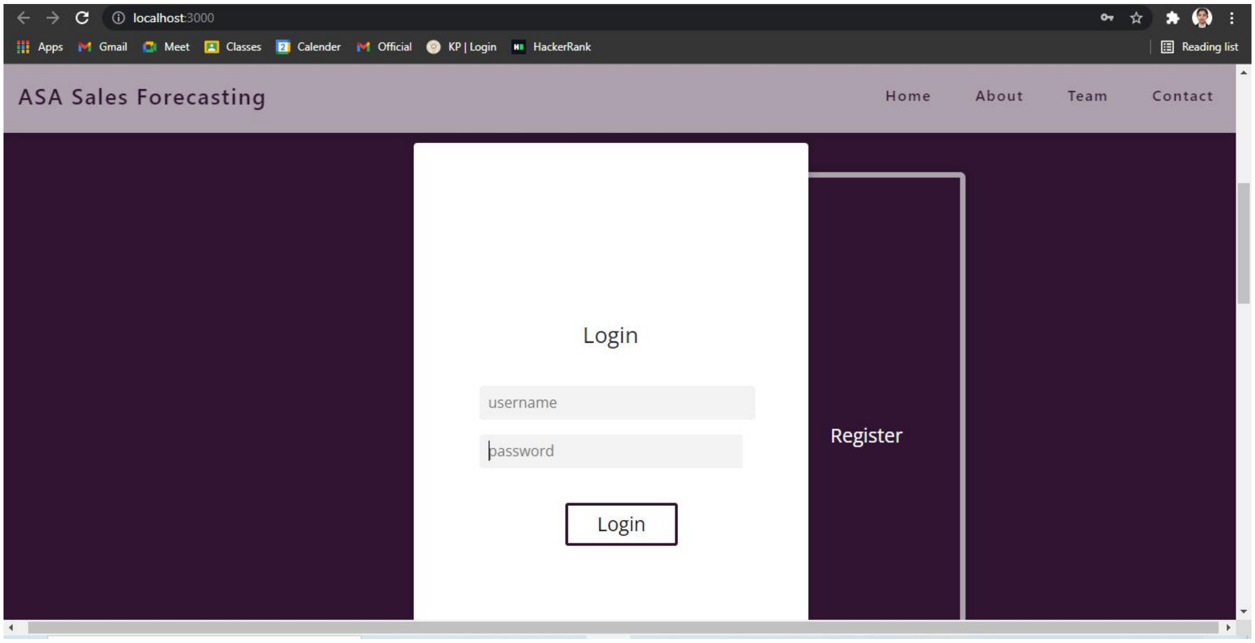


Fig. 7.4 Login for store owner

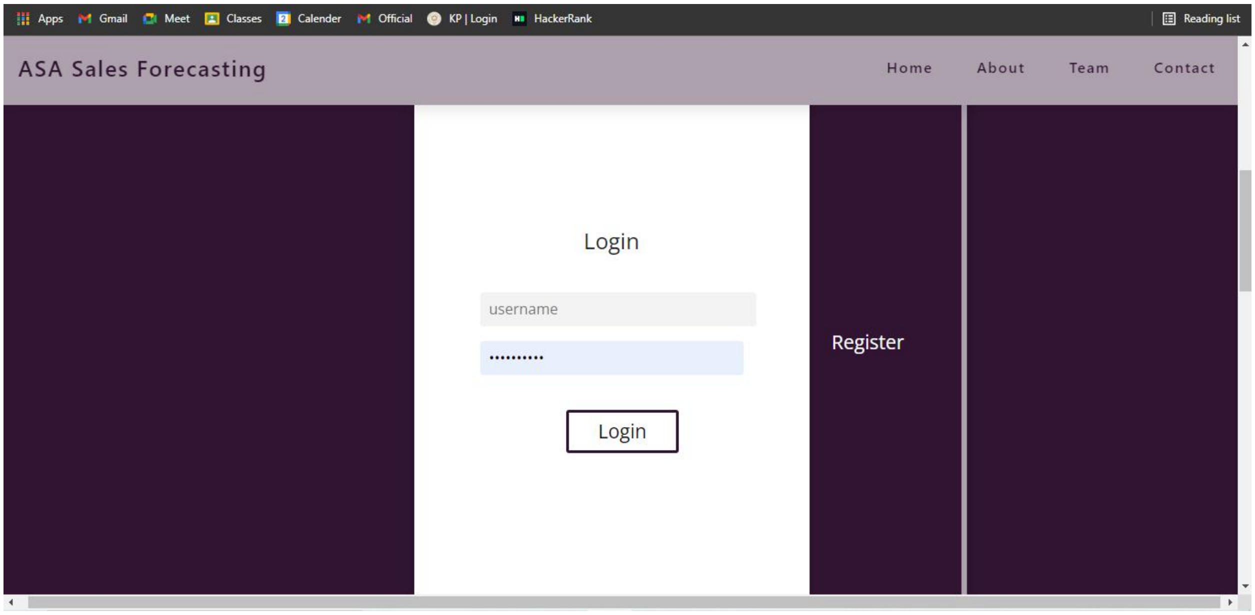


Fig.7.5 Login for company owner

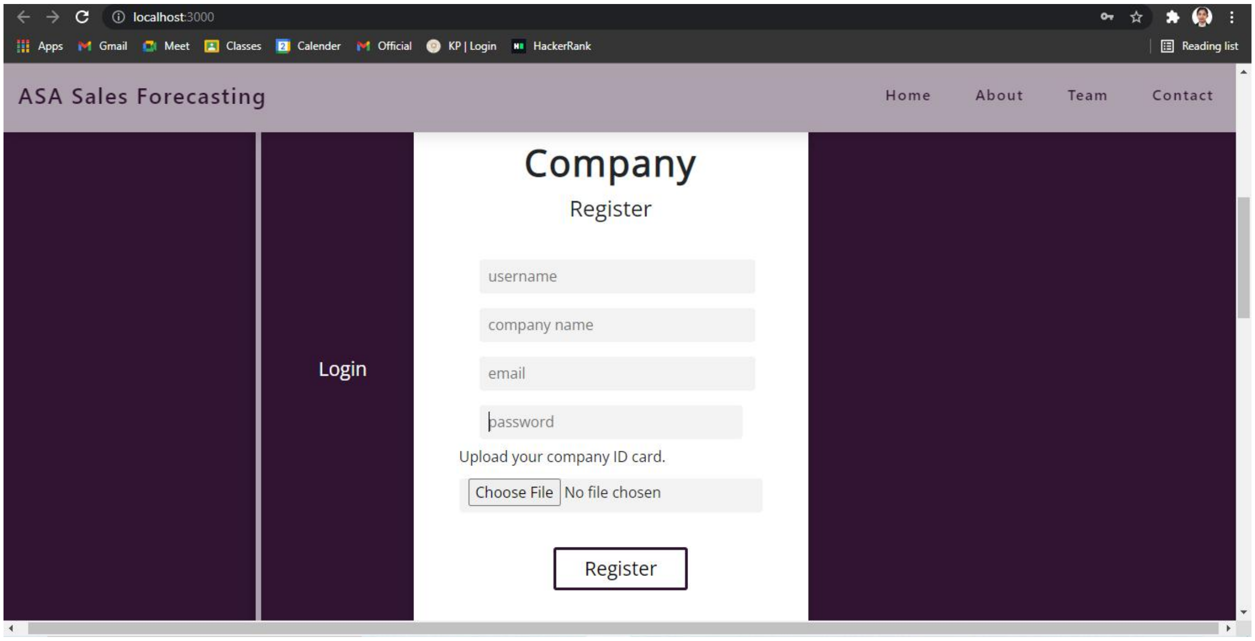


Fig. 7.6 Register for company owner

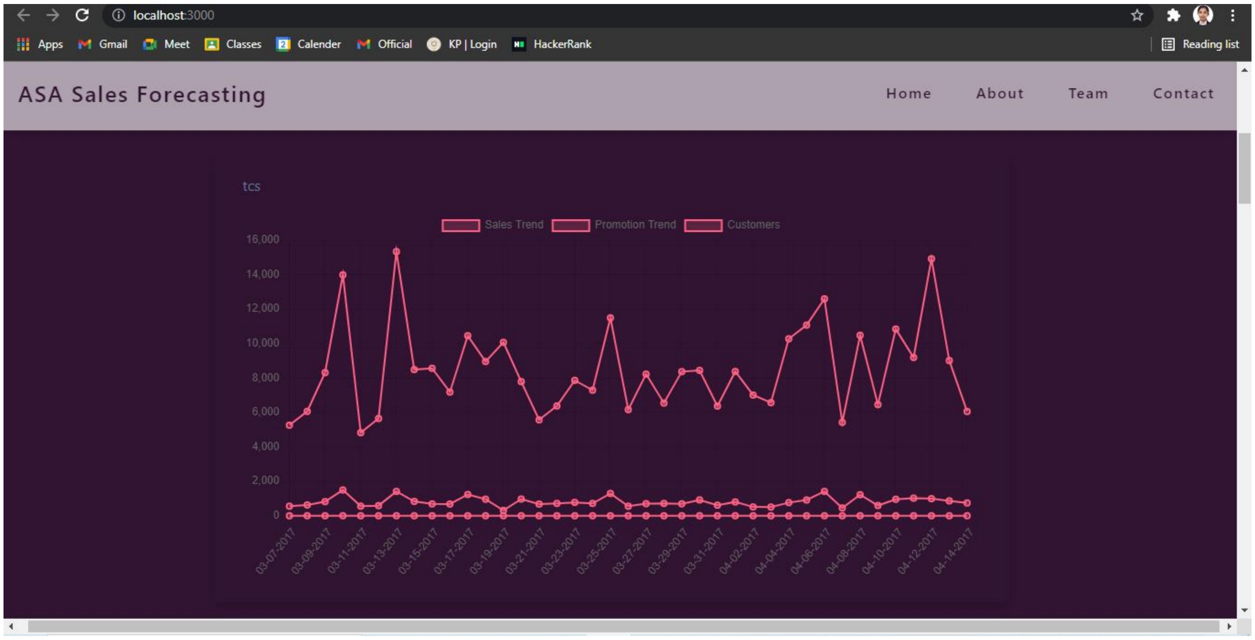


Fig. 7.7.1 Fig Reports

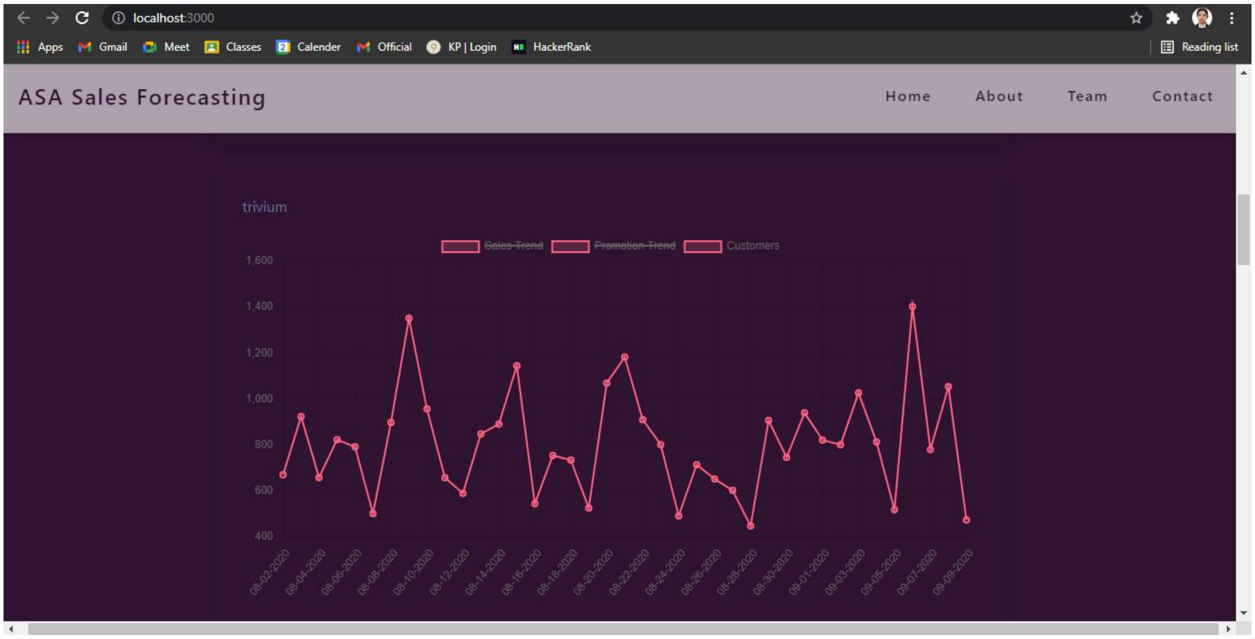
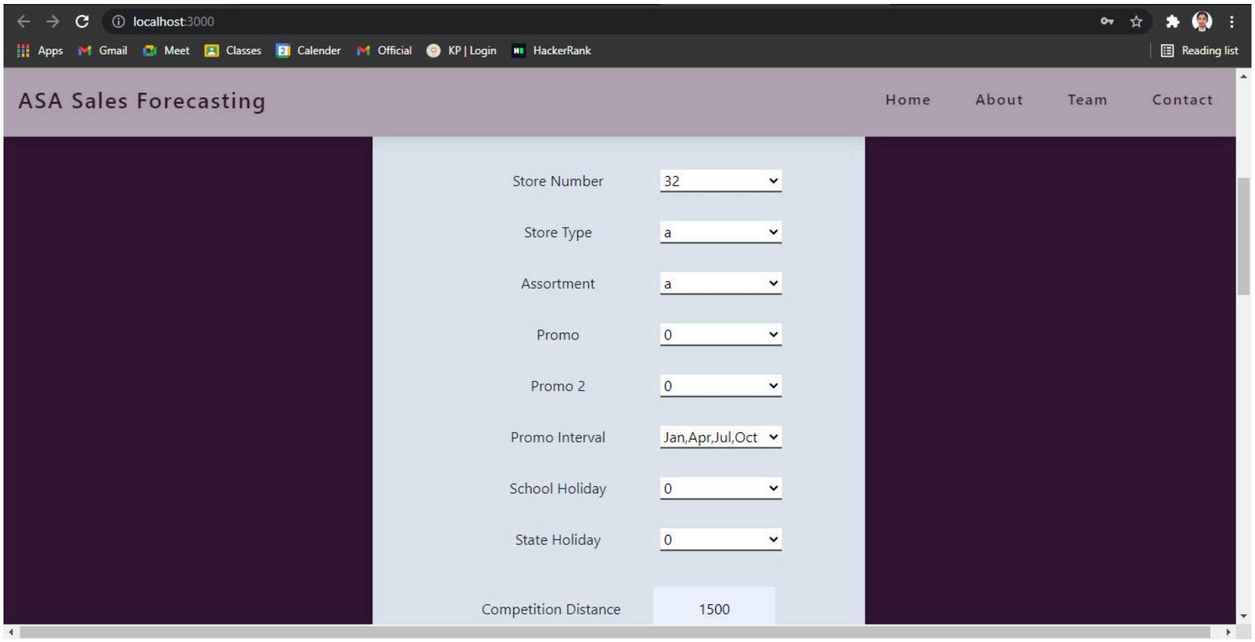


Fig. 7.7.2 Fig Reports



ASA Sales Forecasting

Store Number: 32

Store Type: a

Assortment: a

Promo: 0

Promo 2: 0

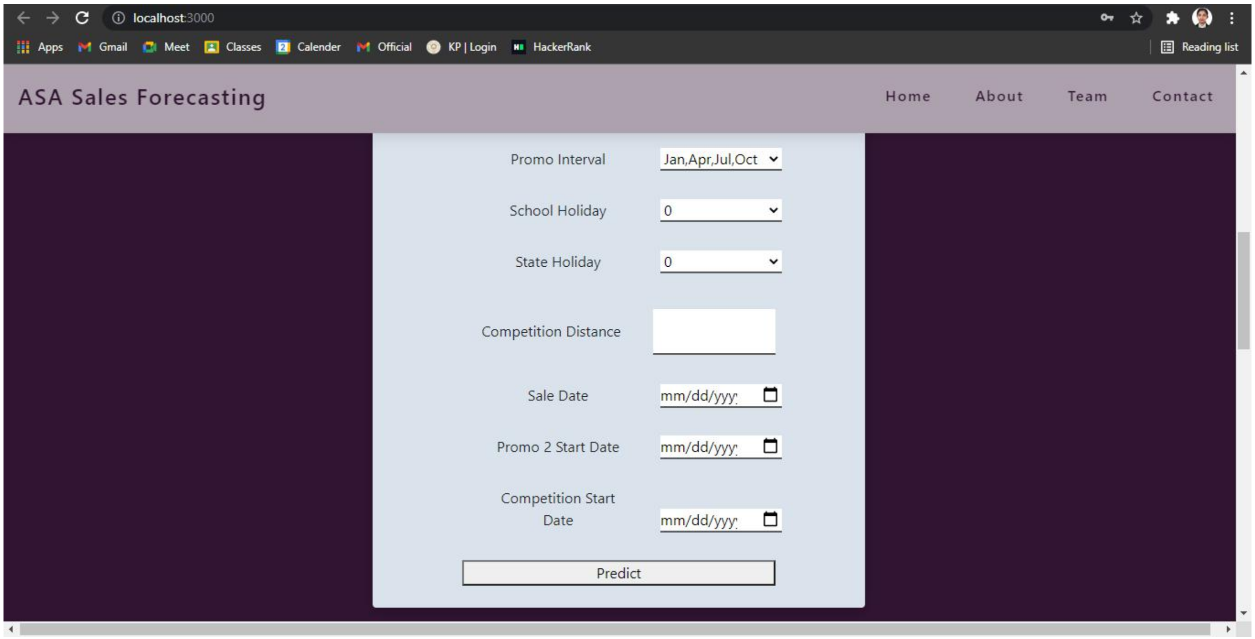
Promo Interval: Jan,Apr,Jul,Oct

School Holiday: 0

State Holiday: 0

Competition Distance: 1500

Fig. 7.8.1 Prediction



ASA Sales Forecasting

Promo Interval: Jan,Apr,Jul,Oct

School Holiday: 0

State Holiday: 0

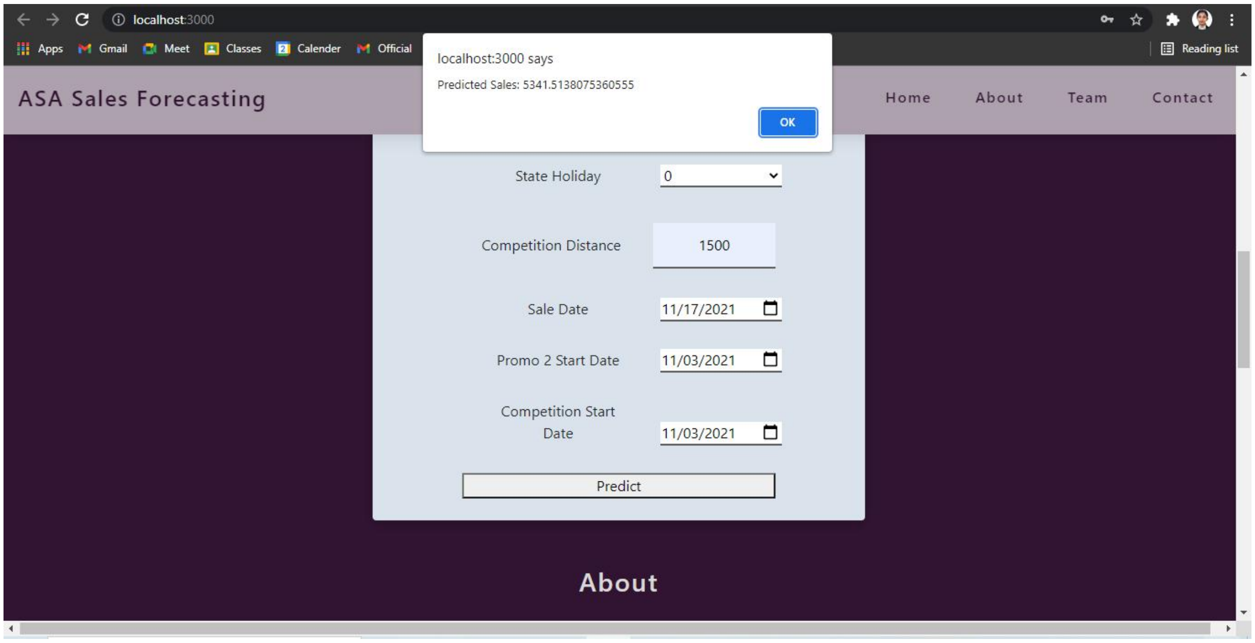
Competition Distance:

Sale Date: mm/dd/yyyy

Promo 2 Start Date: mm/dd/yyyy

Competition Start Date: mm/dd/yyyy

Fig. 7.8.2 Prediction



localhost:3000 says

Predicted Sales: 5341.5138075360555

OK

State Holiday: 0

Competition Distance: 1500

Sale Date: 11/17/2021

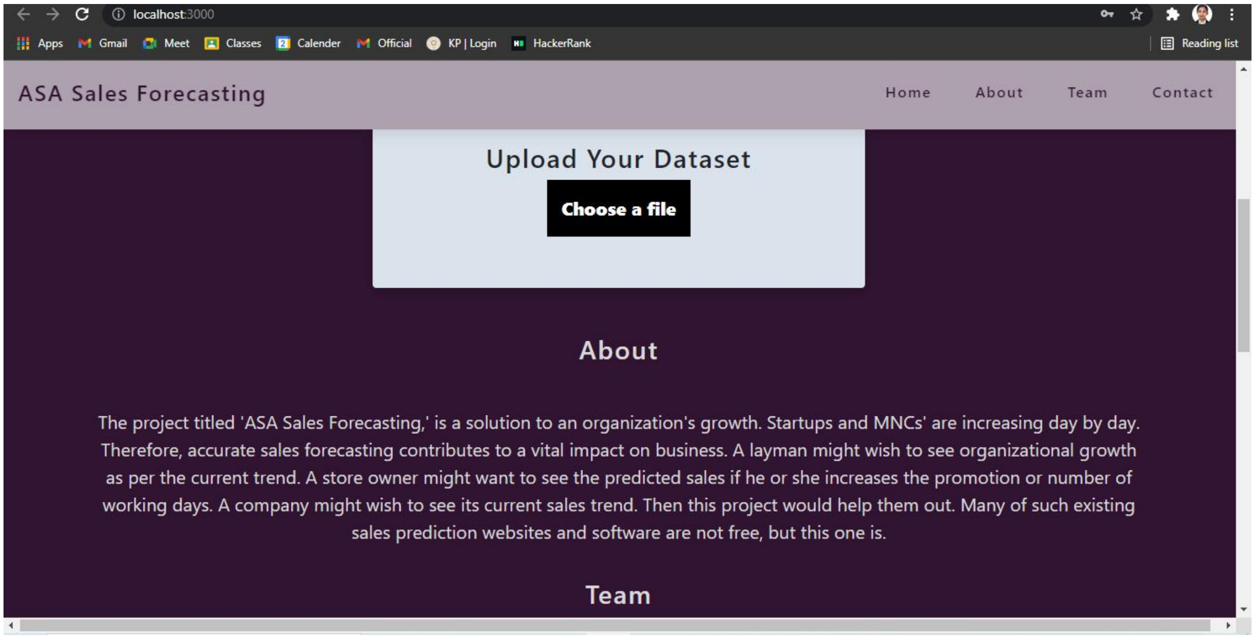
Promo 2 Start Date: 11/03/2021

Competition Start Date: 11/03/2021

Predict

About

Fig. 7.8.3 Prediction



Upload Your Dataset

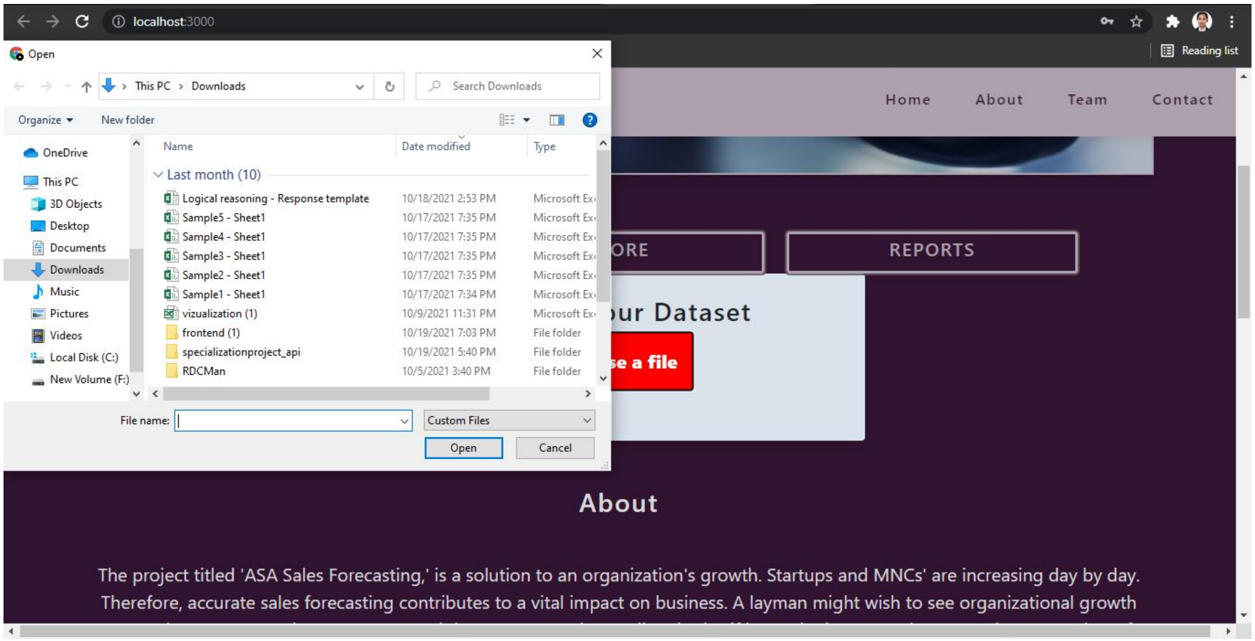
Choose a file

About

The project titled 'ASA Sales Forecasting,' is a solution to an organization's growth. Startups and MNCs' are increasing day by day. Therefore, accurate sales forecasting contributes to a vital impact on business. A layman might wish to see organizational growth as per the current trend. A store owner might want to see the predicted sales if he or she increases the promotion or number of working days. A company might wish to see its current sales trend. Then this project would help them out. Many of such existing sales prediction websites and software are not free, but this one is.

Team

Fig. 7.9 Interface for uploading CSV file



Last month (10)

Name	Date modified	Type
Logical reasoning - Response template	10/18/2021 2:53 PM	Microsoft Ex
Sample5 - Sheet1	10/17/2021 7:35 PM	Microsoft Ex
Sample4 - Sheet1	10/17/2021 7:35 PM	Microsoft Ex
Sample3 - Sheet1	10/17/2021 7:35 PM	Microsoft Ex
Sample2 - Sheet1	10/17/2021 7:35 PM	Microsoft Ex
Sample1 - Sheet1	10/17/2021 7:34 PM	Microsoft Ex
vizualization (1)	10/9/2021 11:31 PM	Microsoft Ex
frontend (1)	10/19/2021 7:03 PM	File folder
specializationproject_api	10/19/2021 5:40 PM	File folder
RDCMan	10/5/2021 3:40 PM	File folder

The project titled 'ASA Sales Forecasting,' is a solution to an organization's growth. Startups and MNCs' are increasing day by day. Therefore, accurate sales forecasting contributes to a vital impact on business. A layman might wish to see organizational growth

Fig. 7.10 Saving report

The developed web application could predict sales with adequate accuracy based on certain attributes provided by the user. It could also bring out interactive visualizations of sales trend belonging to various company's sales data which would bring out an insight to the viewers about the company's profitability.

8. CONCLUSION

The number of startups and multinational corporations (MNCs) is growing every day. As a result, accurate sales forecasting has a significant impact on the business. The project is intended to use machine learning techniques to obtain reliable sales trend predictions to obtain the best sales. This project is to obtain reliable sales trend prediction, with the support of machine learning techniques to achieve the best possible sales. Technically, the project would exhibit company sales trends using various visualization plots to a non-registered user. For registered users, the project would predict the sales based on specific attributes provided by them. The application could also receive sales data from companies, and once received, its sales trends would be visually represented on the website. A provision for feedback and clarification would be provided to contact the team. The project could bring out the visualizations of sales trends of various companies to non-registered users. It could also forecast sales for registered store owners based on specific attribute values they gave. The application was also able to accept sales data from companies, which would be visually shown on the website once received. As a whole, the project was able to bring out an easy user interface to obtain sales prediction and sales trend visualizations.

9. REFERENCES

1. Liebeskind, M. (2020, May 14). 5 Machine Learning Techniques for Sales Forecasting Medium.https://towardsdatascience.com/5-machine-learning-techniques-for-sales-forecasting-598e4984b109
2. Lütkepohl, H. (2006). Chapter 6 Forecasting with VARMA Models. Handbook of Economic Forecasting, 287–325. https://doi.org/10.1016/s1574-0706(05)01006-2
3. A. (2021, April 22). Sales Forecasting Methodology: A Beginner’s Guide. Anaplan. https://www.anaplan.com/blog/sales-forecasting-guide/
4. MacManus, R. (2021, April 5). Django’s Place in a Web Development World Ruled by React. The New Stack. https://thenewstack.io/djangos-place-in-a-web-development-world-ruled-by-react/