

Project report on
DEMANDEST - AI POWERED FOOD DEMAND FORECASTER

TEAM ID: PNT2022TMID32006

R.SAI SIVA SANJAY

B.E(CSE) - [731619104051]

M.GANESH

B.E(CSE) - [731619104013]

R.MATHIVANAN

B.E(CSE) - [731619104032]

K.GAYATHRI

B.E(CSE) - [731619104014]

CONTENTS

1. INTRODUCTION

Project Overview

Purpose

2. LITERATURE SURVEY

Existing problem

References

Problem Statement

Definition

3. IDEATION & PROPOSED SOLUTION

Empathy Map Canvas

Ideation & Brainstorming

Proposed Solution

Problem Solution fit

4. REQUIREMENT ANALYSIS

Functional requirement

Non-Functional requirements

5. PROJECT DESIGN

Data Flow Diagrams

Solution & Technical Architecture

User Stories

6. PROJECT PLANNING & SCHEDULING

Sprint Planning & Estimation

Sprint Delivery Schedule

Reports from JIRA

7. CODING & SOLUTIONING

Data Dictionary

Libraries Used

Data Pre-Processing

Feature Engineering

Data Transformation

Evaluation Metric

Initial Approach

Advanced Models

8. TESTING

Test Cases User Acceptance

Testing

9. RESULTS

Performance Metrics

10. ADVANTAGES & DISADVANTAGES

11. APPLICATIONS

12. CONCLUSION

13. FUTURE SCOPE

14. APPENDIX

Source Code

Output Screenshots

GitHub & Project Demo Link

1. INTRODUCTION

OVERVIEW

A food delivery service has to deal with a lot of perishable raw materials which makes it all, the most important factor for such a company is to accurately forecast daily and weekly demand. Too much inventory in the warehouse means more risk of wastage, and not enough could lead to out-of-stocks - and push customers to seek solutions from your competitors. The replenishment of the majority of raw materials is done on weekly basis and since the raw material is perishable, the procurement planning is of utmost importance, the task is to predict the demand for the next 10 weeks.

PURPOSE

The main aim of this project is to create an appropriate machine learning model to forecast then number of orders to gather raw materials for next ten weeks. To achieve this, we should know the information about of fulfillment center like area, city etc., and meal information like category of food, sub category of food, price of the food or discount in particular week. By using this data, we can use any classification algorithm to forecast the quantity for 10 weeks. For this a web application is built which is integrated with the model.

2. LITERATURE SURVEY

EXISTING PROBLEM

The replenishment of the majority of raw materials is done on weekly basis and since the raw material is perishable, the procurement planning is of utmost importance. Also the recruiting of staff members at the fulfillment center is an prospect wherein the prediction of orders would be beneficial. Although this is a process that can be done manually.

REFERENCE

- Adi, G. N. (2018, March 9). Thousands of GO-CAR Drivers on Strike in Surakarta. The Jakarta Post. <https://www.thejakartapost.com/news/2018/03/08/thousands->

-of-go-car- drivers-on-strike-in-surak arta.html

- Alkhatib, A., & Bernstein, M. (2019, May). Street-level algorithms: A theory at the gaps between policy and decisions. In Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (pp. 1-13).
- Brown, Tim. (2008). Design Thinking. Harvard Business Review. 86. 84-92, 141.
- Colley, A., & Häkkinä, J. (2018, November). Service Design Methods for Human Computer Interaction. In Proceedings of the 17th International Conference on Mobile and Ubiquitous Multimedia (pp. 563-566).

PROBLEM STATEMENT DEFINITION

Problem statement to understand your customer's point of view. The Customer Problem Statement template helps you focus on what matters to create experiences people will love.

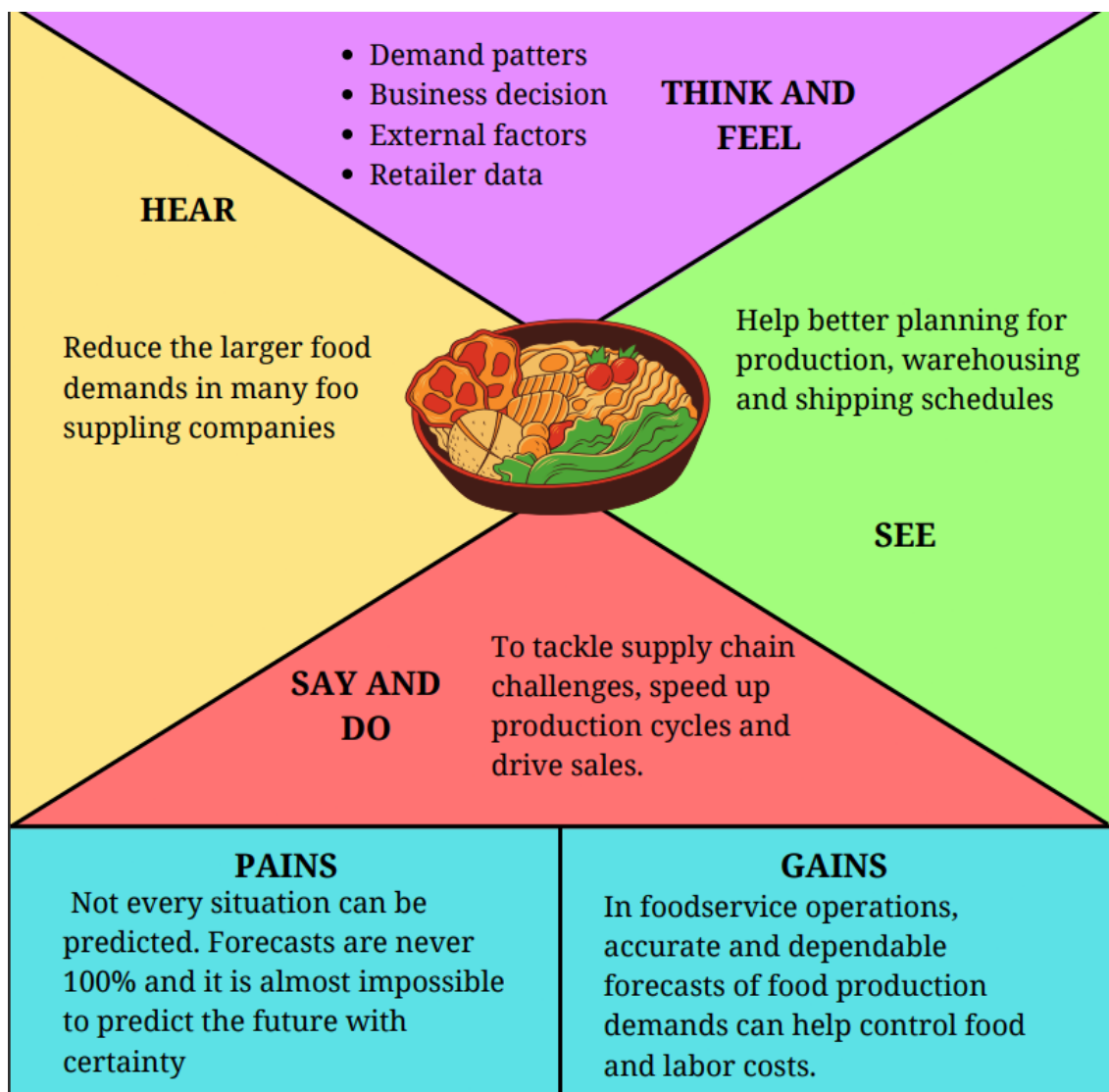
Problem Statement (PS)	I am (Customer)	I'm trying to	But	Because	Which makes me feel
PS-1	Food Delivery Company	(i) buy raw materials. (ii) Plan to predict the demand.	(i) It leads to risk of wastage or out-of-stock. (ii) It takes long time to check the stock.	(i) They don't know the amount of raw materials to be purchased (ii) The prediction is not correct	(i) waste of cost. (ii) Frustrated.
PS-2	Restaurant	(i) Expect sales in a particular period of time. (ii) Predict the amount of sales done in a week.	(i) It is not correct in seasonal days. (ii) not to predict the amount of materials need to buy.	(i) In seasonal days more customers will come. (ii) It leads to out of stock or wastage.	(i) less profitability. (ii) Reduce turnover. (iii) To be complicated.

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

An empathy map is a collaborative visualization used to articulate what we know about a particular type of user. It externalizes knowledge about users in order to 1) create a shared understanding of user needs, and 2) aid in decision making.

Traditional empathy maps are split into 4 quadrants (Says, Thinks, Does, and Feels), with the user or persona in the middle. Empathy maps provide a glance into who a user is as a whole and are not chronological or sequential.




3.2 Ideation & Brainstorming

Brainstorming is a method design teams use to generate ideas to solve clearly defined design problems. In controlled conditions and a free-thinking environment, teams approach a problem by such means as “How Might We” questions. They produce a vast array of ideas and draw links between them to find potential solutions .

Step-1: Team Gathering, Collaboration and Select the Problem Statement


Template



Brainstorm & idea prioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.


🕒 10 minutes to prepare
🕒 1 hour to collaborate
👤 2-8 people recommended





Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.


🕒 10 minutes

 **Team gathering**
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

 **Set the goal**
Think about the problem you'll be focusing on solving in the brainstorming session.

 **Learn how to use the facilitation tools**
Use the Facilitation Superpowers to run a happy and productive session.

[Open article](#) →




Define your problem statement

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

🕒 5 minutes


PROBLEM


Predicting how much the consumer/customer will purchase has always been a challenge





Key rules of brainstorming


To run a smooth and productive session!


 Stay in topic.

 Encourage wild ideas.

 Defer judgment.

 Listen to others.

 Go for volume.

 If possible, be visual.

Step-2: Brainstorm, Idea Listing and Grouping

2

Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

SAI SIVA SAKUNAR

Basic User story types	Subsequent priority	How to make the user
Feature story types	Feature story types	Feature story types
Feature story types	Feature story types	Feature story types
Feature story types	Feature story types	Feature story types

GANESH M

Basic User story types	Subsequent priority	How to make the user
Feature story types	Feature story types	Feature story types
Feature story types	Feature story types	Feature story types
Feature story types	Feature story types	Feature story types

MATHYANAN R

Basic User story types	Subsequent priority	How to make the user
Feature story types	Feature story types	Feature story types
Feature story types	Feature story types	Feature story types
Feature story types	Feature story types	Feature story types

GAYATHRI K

Basic User story types	Subsequent priority	How to make the user
Feature story types	Feature story types	Feature story types
Feature story types	Feature story types	Feature story types
Feature story types	Feature story types	Feature story types

TP

You can convert a sticky note into a card if the card layout is better than the sticky note layout.

3

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you can break it up into smaller sub-groups.

20 minutes

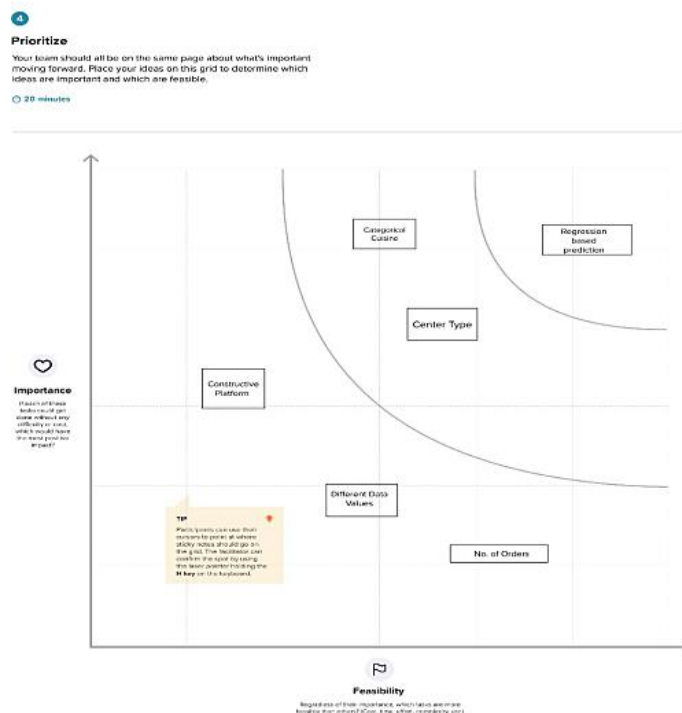
TP

And a reminder to keep the sticky notes to keep a close eye on the board, keeping track of the sticky notes as they are added to the board.

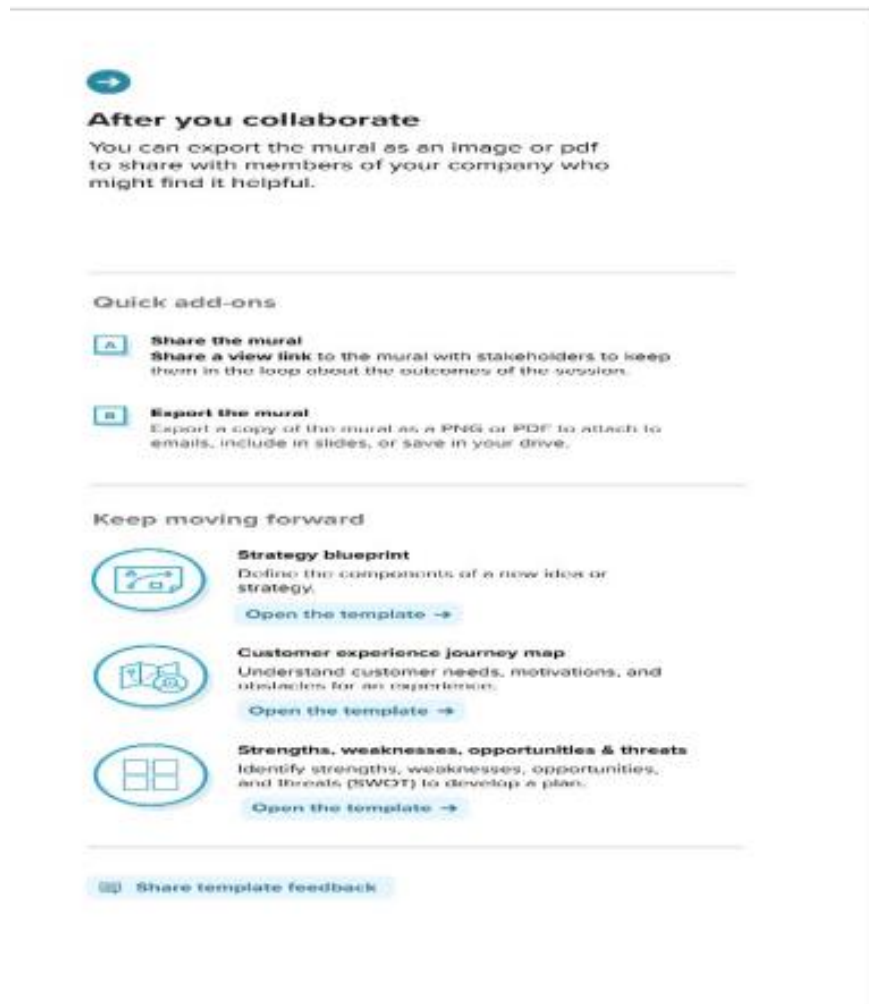
```

graph LR
    Platform[Platform] --- OrderID[Order ID Separation]
    Platform --- Orders[Orders with respect to city]
    Platform --- FoodType[Food Type]
    Datasets[Datasets] --- DynamicData[Dynamic data extraction in a test case]
    Datasets --- DataForFood[Data for different food types]
    Datasets --- Training[Training data evaluation]
    Analysis[Analysis & Recognition] --- Results[Get Region wise Results]
    Analysis --- Optimal[Optimal Decision making]
    Analysis --- Accuracy[Analysis for more accuracy]
    Analysis --- Split[Train and Test Split]
  
```

Step-3: Idea Prioritization



Step-4 :



3.3 Proposed Solution

Proposed Solution means the technical solution to be provided by the Implementation agency in response to the requirements and the objectives of the Project. Proposed Solution means the Proposed System with modifications that meet the Agency's requirements as set forth in this RFP. Proposed Solution means the combination of software, hardware, other products or equipment, and any and all services (including any installation, implementation, training, maintenance and support services) necessary to implement the solution described by Vendor in its Proposal.

S.No	TOPIC	CONTENT
1.	Problem Statement	A food delivery company needs an appropriate machine learning model to accurately forecast daily and weekly demand . number of orders to gather raw materials because there isa risk of wastage and not enough could lead to out of stocks and push customers to seek solutions.
2.	Solution	<p>Develop software to forecast the food demands for a week or 15 days.</p> <p>Use Machine learning model that help to make efficient balance between the order demand and planned courier service.</p> <p>Store the leftover food materials properly to avoid risk of wastage.</p> <p>Determine the demand and then place order.</p> <p>Use efficient tools for predicting the risk of wastage and over stocking problem.</p> <p>Track sales with POS system that helps to decide how much stock needs to buy.</p>
3.	Business Model	<p>The replenishment of raw materials is done only weekly and since the raw material is perishable, the procurement planning is of utmost importance.</p> <p>Therefore predicting the demand helps in reducing the wastage Of raw materials which would result in the reduced cost of operation. Increased customer satisfaction by timely fulfilling their expectations and requirements.</p>
4.	Novelty	AI powered demand forecasting Is the strongest method by which to grow revenue and increase profit The increased accuracy that AI demand forecasts provides results in precise future replenishment quantity predictions. This data reduces missed sales opportunities through stock outs, and also reduces waste, which both directly impact overall profits.
5.	Scalability of Solution	<p>Most demand forecasting solutions focus on small to medium sized applications that offer low scalability. However,modem food industry businesses are highly diversified in the products that they provide through large chains of network stores.</p> <p>This scale of business presents a significant challenge for traditional demand forecasting software, which is why startups are providing doud -based solutions geared towards large-scale operations.</p>

3.4 Problem Solution Fit

The Problem-Solution Fit simply means that you have found a problem with your customer and that the solution you have realized for it actually solves the customer's problem.

Define CS, fit into CC	1. CUSTOMER SEGMENT(S) CS <ul style="list-style-type: none"> Organizations that need to predict changes in consumer demand to provide higher customer satisfaction. Manufacturer of food products who uses demand forecasting for improved markdown/discount optimization. 	6. CUSTOMER CONSTRAINTS CC <ul style="list-style-type: none"> Seamless Internet connectivity. Power supply in case of usage of desktop system. Basic technical knowledge for performing efficient operations on the platform. 	5. AVAILABLE SOLUTIONS AS <ul style="list-style-type: none"> Prediction using LSTM :Short-term memory neural networks (LSTMs) are suitable for demand forecasting but requires longer time and more memory. Prediction using Ensemble Learning: An ensemble can make better predictions and achieve better performance than any single contributing model but expensive.
	2. JOBS-TO-BE-DONE / PROBLEMS J&P <ul style="list-style-type: none"> Provide daily and weekly demand that needs to be precise and accurate. Eliminate wastage which would otherwise increase the operating cost. Having products always available which improves customer loyalty and brand perception To avoid cash-in-stock situation for retail companies, where products remain unsold for a longer period than expected. 	9. PROBLEM ROOT CAUSE RC <ul style="list-style-type: none"> Lack of Adequate, Accurate and Timely Demand Data. Traditional demand forecasting falls short of this vital aspect as it stretches over a longer time period which can dilute the essence of real-time data tracking and near-term visibility Food retailers very often get misled by the slightest abnormal shift in demand, which if not explained can be treated as a signal for change in planning and forecasting models. 	7. BEHAVIOUR BE <ul style="list-style-type: none"> Company might conduct client-intent surveys to ask what the customer is planning on buy in the future. Conjoint analysis is used to obtain consumer input about the most favorable attributes of their products. These surveys ask consumers how they would use and respond to certain product attributes. Traditional forecast model used to estimate future value – usually from historical records of business performance metrics.
Identify strong TR & EM	3. TRIGGERS TR <ul style="list-style-type: none"> Social media advertisements. The cost-free, self intuitive simple nature of platform that favors interaction. Other food retailers gaining maximum profit using the AI platform. 	10. YOUR SOLUTION <ul style="list-style-type: none"> To use simple linear regression model which is a quantitative way to predict future values with past values. Post applying feature engineering and data transformation. Feature engineering is the process of using domain knowledge of the data to create features that improves the performance of the machine learning models. 	8. CHANNELS of BEHAVIOUR CH
	4. EMOTIONS: BEFORE / AFTER EM <ul style="list-style-type: none"> Before : Doubt, ambiguous, stressed, disoriented. After accurate prediction: Happiness,determined,explicit,calmness. 		8.1 ONLINE <ul style="list-style-type: none"> Cost free, simplistic prediction can be made online. User Interactive website that can be accessed by anyone anytime 8.2 OFFLINE

4. REQUIREMENT ANALYSIS

Functional requirement

Functional requirements may involve calculations, technical details, data manipulation and processing, and other specific functionality that define what a system is supposed to accomplish. Behavioral requirements describe all the cases where the system uses the functional requirements, these are captured in use cases.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	The web ordering system	Create an account. Manage their account. Login to the system. Navigate the restaurant's menu. Select an item from the menu. Review their current order.
FR-2	Menu Management system	Add a new/update/delete vendor to/from the menu. Add a new/update/delete food category to/from the menu. Add a new/update/delete food item to/from the menu. Update price for a given food item.
FR-3	Order Retrieval system	Retrieve new orders from the database. Display the orders in an easily readable, graphical way. Mark an order as having been processed and remove it from the list of active orders.

Non-Functional requirements

In systems engineering and requirements engineering, a non-functional requirement (NFR) is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviours.

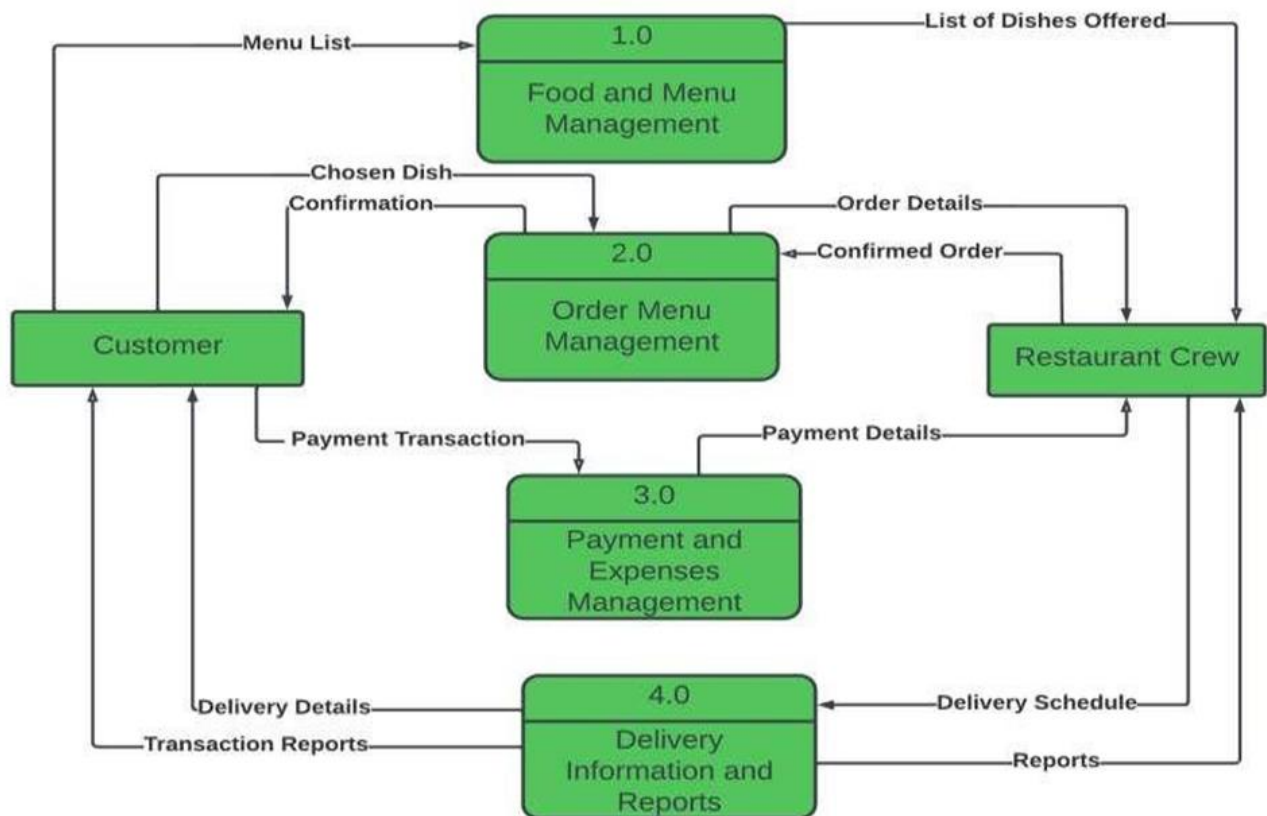
FR No.	Non-Functional Requirement	Description
NFR-1	Usability	The system should provide an interactive user-friendly interface that is easily understandable for all users.
NFR-2	Security	Only authorized users must be able to access the system and view and modify the data.
NFR-3	Maintainability	The software should be easily maintainable and adding new features and making changes to the software must be as simple as possible.
NFR-4	Dependability	The system should provide consistent performance with easy tracking of records and updating of records.

NFR-5	Availability	The system should be available at least during the restaurant operating hours and must be recovered within an hour or less if it fails. The system should respond to the requests within two seconds or less.
NFR-6	Configurability	Configurability is just as important as integrability.

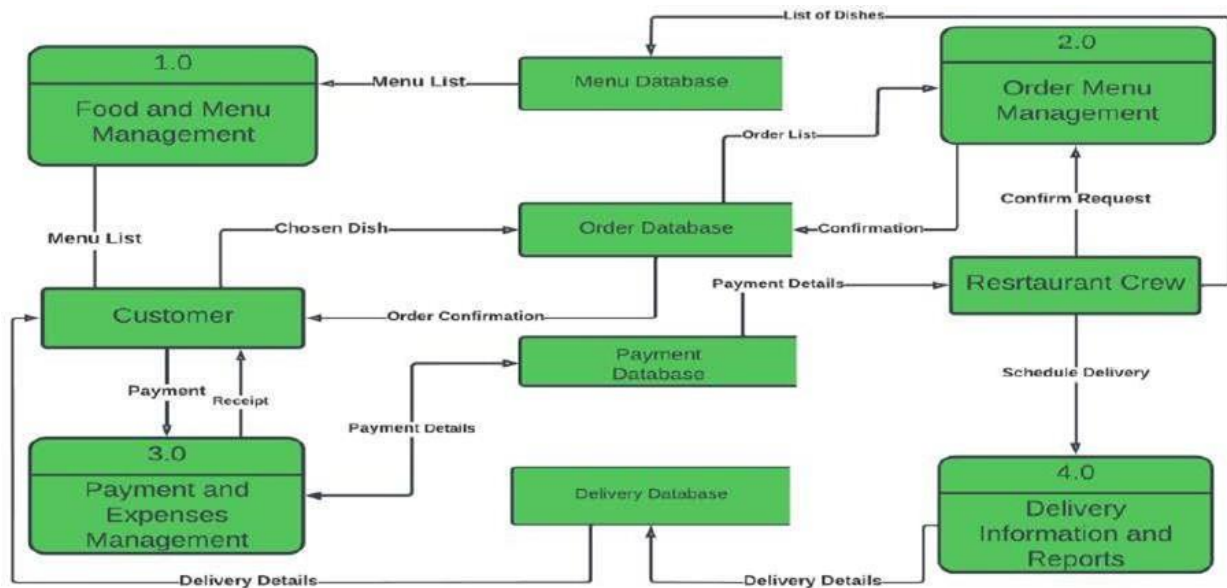
5. PROJECT DESIGN

Data Flow Diagrams

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



DATA FLOW DIAGRAM –LEVEL 1

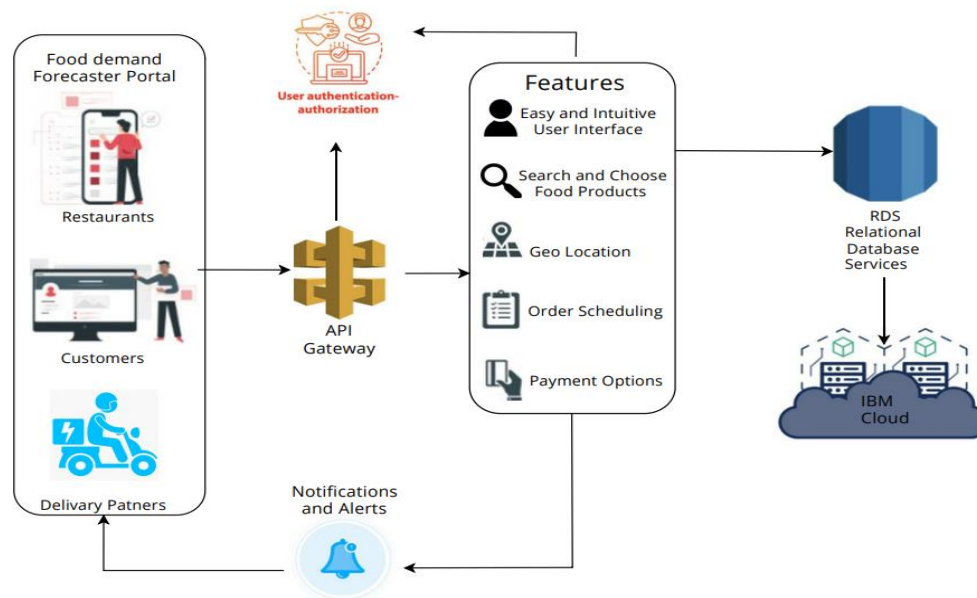


DATA FLOW DIAGRAM -LEVEL 2

Solution & Technical Architecture

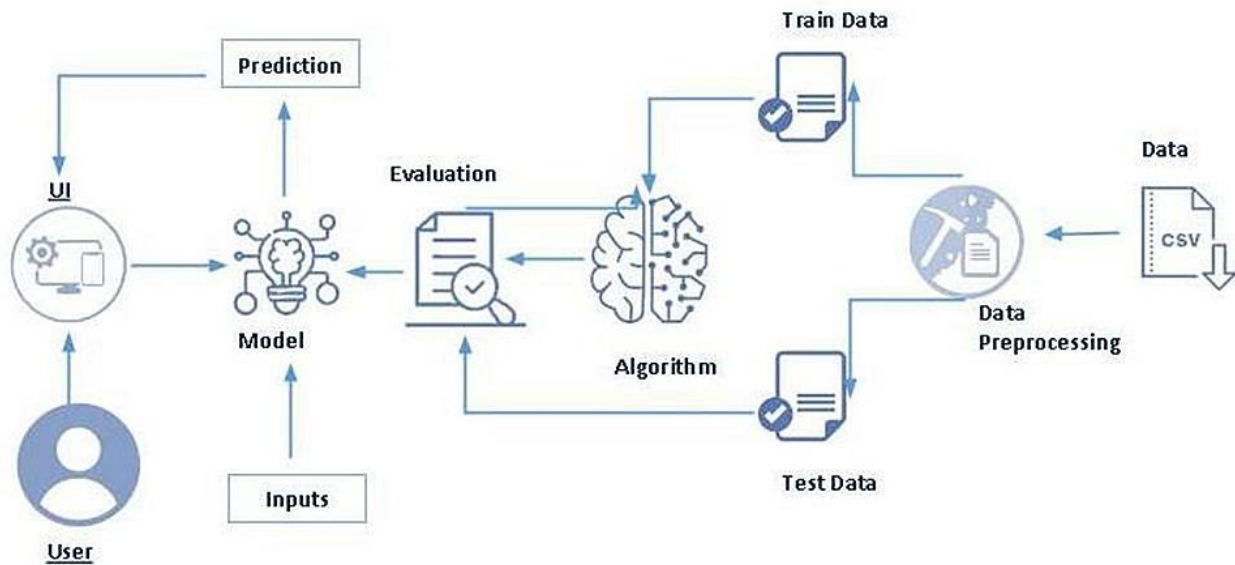
Solution Architecture:

A solution architecture (SA) is an architectural description of a specific solution. SAs combine guidance from different enterprise architecture viewpoints (business, information and technical), as well as from the enterprise solution architecture (ESA).



Technical Architecture:

Technical Architecture (TA) is a form of IT architecture that is used to design computer systems. It involves the development of a technical blueprint with regard to the arrangement, interaction, and interdependence of all elements so that systemrelevant requirements are met.



User Stories

A user story is an informal, general explanation of a software feature written from the perspective of the end user or customer. The purpose of a user story is to articulate how a piece of work will deliver a particular value back to the customer.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
Business staff	Confirmation	USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
Tourist People	Accessibility	USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
College Students	Customer access through mail.	USN-4	As a user, I can register for the application through Gmail	I can confirm the order and get the OTP through email	Medium	Sprint-1
Customer (websites)	Login	USN-5	As a user, I can log into the application by entering email & password	I can log into the application to check either precise location or approximate location.	Medium	Sprint-2
	Dashboard	USN-6	Choosing the menu, Restaurant and payment process. after receiving the food rating process.	Hazard analysis and critical control point system.	Low	Sprint-1
Customer (Web user)	Customers Orders	USN-7	Delivery partner simply tracks the order and lets the customer know when it will arrive.	Tracking through GPS makes sure whether the given time slot is achieved.	Medium	Sprint-2
Customer Care Executive	Customer order delivery	USN-8	Doorstep delivery. easy process to get the order.	Rating on delivery partner and food quality.	Medium	Sprint-2
Administrator	Hotel management, website holders	USN-9	Choosing the restaurant. Multiple choice for restaurant profile.	Advertising through websites.	Low	Sprint-1

6. PROJECT PLANNING & SCHEDULING

Sprint Planning & Estimation

In Scrum Projects, Estimation is done by the entire team during Sprint Planning Meeting. The objective of the Estimation would be to consider the User Stories for the Sprint by Priority and by the Ability of the team to deliver during the Time Box of the Sprint.

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Pre - Requisites	USN-1	A prerequisite is a required prior condition. If something is required in advance of something else, like if you have to take a beginning Spanish class before signing up for Spanish II, then it's a prerequisite.	10	Low	Sai Siva Sanjay R Ganesh M Mathivanan R Gayathri K
Sprint-1	Dataset collection	USN-2	A tool in Agile software development used to capture a description of a software feature from a user's perspective.	10	Medium	Mathivanan R Gayathri K
Sprint-2	Data Pre-Processing. Improving the libraries	USN-3	In this post I am going to walk through the implementation of Data Preprocessing methods using Python.	5	High	Sai Siva Sanjay R Ganesh M
Sprint-2	Reading the dataset. Exploratory data analysis	USN-4	Exploratory Data Analysis refers to the critical process of performing initial investigations on data so as to discover patterns, to spot anomalies, to test hypothesis and to check assumptions with the help of summary statistics and graphical representations.	5	High	Sai Siva Sanjay R Ganesh M

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-2	Checking for null values. Reading and merging.csv files.	USN-5	A null indicates that a variable doesn't point to any object and holds no value. Step 1: Create & Export Multiple Data Frames. First, we'll use the following code to create and export three data frames to CSV files: #create three data frames df1 <- data. ... Step 2: Import & Merge Multiple CSV Files.	2	Medium	Mathivanan R Gayathri K Sai Siva Sanjay R Ganesh M
Sprint-2	Dropping columns. Label encoding	USN-6	First, you define the table name from which you wish to remove or delete the column. Label Encoding refers to converting the labels into a numeric form so as to convert them into the machine-readable form.	6	Medium	Mathivanan R Gayathri K
Sprint-2	Splitting the dataset into dependent and independent variable. Split the dataset into train set and test set	USN-7	The simplest way to split the modelling dataset into training and testing sets is to assign 2/3 data points to the former and the remaining one-third to the latter.	2	Low	Sai Siva Sanjay R Ganesh M
Sprint-3	Model Building	USN-8	What the person using the product wants to Be able to do. A traditional requirements focuses on functionality.	10	High	Sai Siva Sanjay R Ganesh M
Sprint-3	Train and test model algorithms Model evaluation	USN-9	The train-test split procedure is used to estimate the performance of machine learning algorithms when they are used to make predictions on data .	5	Low	Mathivanan R Gayathri K
Sprint-3	Save the model. Predicting the output using the model.	USN-10	predict passes the input vector through the model and returns the output tensor for each datapoint.	5	Medium	Mathivanan R Gayathri K
Sprint-4	Application building. Create an HTML file	USN-11	An app builder is an online software tool that allows everyone to create and publish apps for mobile devices without code development.	10	High	Mathivanan R Gayathri K Sai Siva Sanjay R Ganesh M
Sprint-4	Build python code. Run the app	USN-12	A tool provided by the Python Packaging Authority (PyPA) for building Python packages.	10	High	Sai Siva Sanjay R Ganesh M

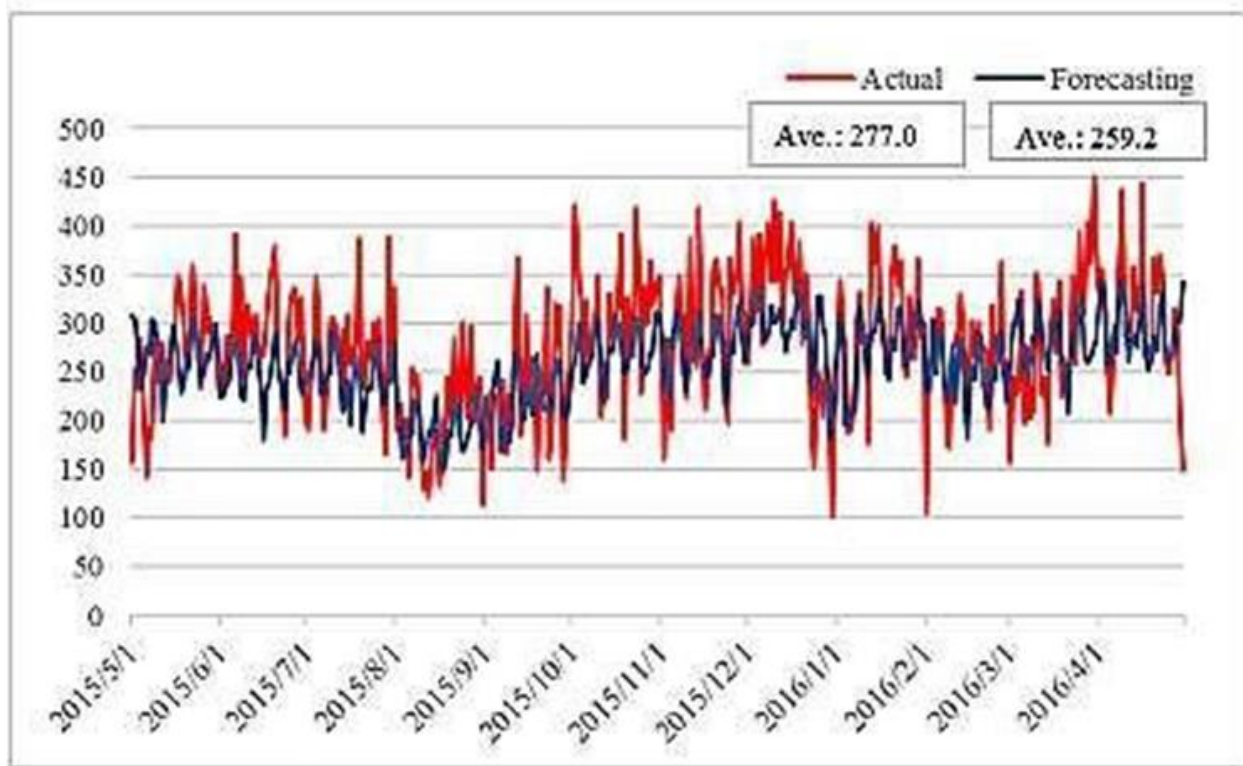
Sprint Delivery Schedule

A sprint schedule is a document that outlines sprint planning from end to end. It's one of the first steps in the agile sprint planning process—and something that requires adequate research, planning, and communication.

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	30 Oct 2022	04 Nov 2022	20	04 Nov 2022
Sprint-3	20	6 Days	05 Nov 2022	10 Nov 2022	20	10 Nov 2022
Sprint-4	20	6 Days	11 Nov 2022	16 Nov 2022	20	16 Nov 2022

Reports From JIRA

Jira helps teams plan, assign, track, report, and manage work and brings teams together for everything from agile software development and customer support to startups and enterprises. Software teams build better with Jira Software, the #1 tool for agile teams.



7. CODING & SOLUTIONING

Data Dictionary

Our base data consists of four csv files containing information about test data, train data and other required information.

- train.csv: Contains information like id, week, center id, meal id, checkout price, base price, emailer for promotion, homepage featured, number of orders. This file is used for training.

Variable	Definition
id	Unique ID
week	Week No
center_id	Unique ID for fulfillment center
meal_id	Unique ID for Meal
checkout_price	Final price including discount, taxes & delivery charges
base_price	Base price of the meal
emailer_for_promotion	Emailer sent for promotion of meal
homepage_featured	Meal featured at homepage
num_orders	(Target) Orders Count

- test.csv: Contains information like id, week, centerid, meal id, checkout price, base price, emailer for promotion, homepage featured. This file is used for testing.
- fulfillment_center_info.csv: Contains information of each fulfillment center.

Variable	Definition
center_id	Unique ID for fulfillment center
city_code	Unique code for city
region_code	Unique code for region
center_type	Anonymized center type
op_area	Area of operation (in km ²)

- meal_info.csv: Contains information of each meal being served.

Variable	Definition
meal_id	Unique ID for the meal
category	Type of meal (beverages/snacks/soups....)
cuisine	Meal cuisine (Indian/Italian/...)

Libraries Used

pandas, numpy, scikit learn, matplotlib, seaborn, xgboost, lightgbm, catboost.

Data Pre-Processing

- There are no Missing/Null Values in any of the three datasets.
- Before proceeding with the prediction process, all the three data sheets need to be merged into a single dataset. Before performing the merging operation, primary feature for combining the datasets needs to be validated.
- The number of Center IDs in train dataset is matching with the number of Center IDs in the Centers Dataset i.e 77 unique records. Hence, there won't be any missing values while merging the datasets together.
- The number of Meal IDs in train dataset is matching with the number of Meal IDs in the Meals Dataset i.e 51 unique records. Hence, there won't be any missing values while merging the datasets together.
- As checked earlier, there were no Null/Missing values even after merging the datasets.

Feature Engineering

Feature engineering is the process of using domain knowledge of the data to create features that improves the performance of the machine learning models.

With the given data, We have derived the below features to improve our model performance.

- Discount Amount : This defines the difference between the “base_Price” and “checkout_price”.
- Discount Percent : This defines the % discount offer to customer.
- Discount Y/N : This defines whether Discount is provided or not - 1 if there is Discount and

0 if there is no Discount.

- Compare Week Price : This defines the increase / decrease in price of a Meal for a particular center compared to the previous week.
- Compare Week Price Y/N : Price increased or decreased - 1 if the Price increased and 0 if the price decreased compared to the previous week.
- Quarter : Based on the given number of weeks, derived a new feature named as Quarter which defines the Quarter of the year.
- Year : Based on the given number of weeks, derived a new feature named as Year which defines the Year.

Data Transformation

- Logarithm transformation (or log transform) is one of the most commonly used mathematical transformations in feature engineering. It helps to handle skewed data and after transformation, the distribution becomes more approximate to normal.
- In our data, the target variable 'num_orders' is not normally distributed. Using this without applying any transformation techniques will downgrade the performance of our model.
- Therefore, we have applied Logarithm transformation on our Target feature 'num_orders' post which the data seems to be more approximate to normal distribution.
- After Log transformation, We have observed 0% of Outlier data being present within the Target Variable – num_orders using 3 IQR Method.

Evaluation Metric

The evaluation metric for this competition is $100 \times \text{RMSLE}$ where RMSLE is Root of Mean Squared Logarithmic Error across all entries in the test set.

Initial Approach

- Simple Linear Regression model without any feature engineering and data transformation which gave a RMSE : 194.402
- Without feature engineering and data transformation, the model did not perform well and could't give a good score.
- Post applying feature engineering and data transformation (log and log1p transformation), Linear Regression model gave a RMSLE score of 0.634.

Advanced Models

- With improvised feature engineering, built advanced models using Ensemble techniques and other Regressor algorithms.
- Decision Tree Regressors performed well on the model which gave much reduced RMSLE.
- With proper hyper-parameter tuning, Decision Tree Regressor performed well on the model and gave the least RMSLE of 0.5237

8. TESTING

Test Cases

A test case includes information such as test steps, expected results and data while a test scenario only includes the functionality to be tested.

Test case ID	Feature Type	Component	Test Scenario
LoginPage_TC_OO1	Functional (Registration)	Home Page	As a user, I can register for the application by entering my email, password, and confirming my password.
LoginPage_TC_OO2	Functional (Conformation)	Home Page	As a User, I will receive confirmation email once I have registered for the application.
LoginPage_TC_OO3	Functional (Accessibility)	Home page	As a user, I can register for the application through Facebook
LoginPage_TC_OO4	Functional (Customer access through mail)	Login page	As a user, I can register for the application through Gmail.
LoginPage_TC_OO4	Functional (Login)	Login page	As a user, I can log out into the application by entering email & password.
LoginPage_TC_OO5	Functional (Dashboard)	Home page	Choosing the menu, Restaurant and payment process. after receiving the food rating process.
LoginPage_TC_OO6	Functional (Customers order)	Home page	Delivery partner simply tracks the order and lets the customer know when it will arrive.
LoginPage_TC_OO7	Functional (Customer order delivery)	Home page	Doorstep delivery. Easy process to get the order.
LoginPage_TC_OO8	Functional (Hotel Management)	Home page	Choosing the restaurant. Multiple choice for restaurant profile.

Pre-Requisite	Steps To Execute
Network Accessing device	<ol style="list-style-type: none"> 1. Check all the textboxes, radiobuttons, buttons. etc. 2. Check the required fields by not filling any data. 3. Check user should Register by filling all the required fields.
Network Accessing device	<ol style="list-style-type: none"> 1. Check results on entering valid user ID & Password. 2. Check results on entering invalid User ID & Password. 3. Check response when a user ID is empty & login button is pressed, and many more.
Network Accessing device	<ol style="list-style-type: none"> 1. If the labels are correctly written and placed or not. 2. If the audio/video content is properly audible/visible or not. 3. If the color contrast ratio is maintained or not. 4. If the control actions for video are working fine or not.
Network Accessing device	<ol style="list-style-type: none"> 1. Enter URL (http://127.0.0.1:5000/) and click go 2. Click on My Account dropdown button 3. Enter Invalid username/email in Email textbox 4. Enter valid password in password textbox 5. Click on login button
Network Accessing device	<ol style="list-style-type: none"> 1. Login with valid credentials. 2. Check the show password feature. 3. Check the Remember Me checkbox. 4. Check the email. 5. Click on login button
Network Accessing device	<ol style="list-style-type: none"> 1. Test Case ID. 2. Test Description. 3. Assumptions and Pre-Conditions. 4. Test Data.
Network Accessing device	<ol style="list-style-type: none"> 1. Keep things simple and transparent. 2. Make test cases reusable. 3. Peer review is important. 4. Keep test case IDs unique.
Network Accessing device	<ol style="list-style-type: none"> 1. Making sure that functionalities are easy to find 2. Navigation should be easy and user-friendly 3. Buttons of the application should be visible. 4. Verification that font should be of appropriate size so that anyone can read them.

Network Accessing device	<ol style="list-style-type: none"> 1. Making sure that functionalities are easy to find 2. Navigation should be easy and user-friendly 3. Buttons of the application should be visible. 4. Verification that font should be of appropriate size so that anyone can read them.
--------------------------	---

Test Data	Expected Result	Actual Result	Status
http://127.0.0.1:5000	Login/Signup popup should display	Working as expected	Pass
http://127.0.0.1:5000	Application should show below UI elements: a. email textbox b. password text box b. Login button with orange colour c. New customer? Create account link	Working as expected	Pass
Username: Jdk@gmail.com password: FDF123	User should navigate to user account homepage	Working as expected	Pass
Username: Jdk@gmail.com password: FDF123	Application should show 'Incorrect email or password' validation message.	Working as expected	Pass
Username: Jdk@gmail.com password: FDF123678686786876876	Application should show 'Incorrect email or password' validation message.	Working as expected	Pass
Username: Jdk@gmail.com password: FDF123678686786876876	Application should show 'Incorrect email or password' validation message.	Working as expected	Pass
Username: jdk@gmail.com password: FDF123	Everything that a customer expects from a product, service or organisation.	Working as expected	Pass
username: Jdk@gmail.com password: FDF123	It should be made clear how many days a delivery might take to process.	Working as expected	Pass
Username: Jdk@gmail.com password: FDF123	It will be commercially accountable for budgeting and financial management and Will need to plan, organise and direct hotel services.	Working as expected	Pass

Defect Analysis:

Resolution	Severity1	Severity2	Severity3	Severity4	Subtotal
By Design	10	4	2	3	20
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	11	2	4	20	37
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won'tFix	0	0	0	1	1
Totals	24	9	11	26	71


Test Case Analysis:

Section	TotalCases	Not Tested	Fail	Pass
PrintEngine	7	0	0	7
ClientApplication	51	0	0	51
Security	2	0	0	2
OutsourceShipping	3	0	0	3
ExceptionReporting	9	0	0	9
FinalReportOutput	4	0	0	4
VersionControl	2	0	0	2

9. RESULTS

Performance Metrics

Performance testing is the practice of evaluating how a system performs in terms of responsiveness and stability under a particular workload. Performance tests are typically executed to examine speed, robustness, reliability, and application size.

S.No.	Parameter	Values	Screenshot
1.	Metrics	Regression Model: MAE 89.10334778841495, MSE - 43129.82977026746, RMSLE -207.67722496765856, R2 score -0.6946496854280233,	Evaluating the model  <pre> In [33]: from sklearn.metrics import mean_squared_error In [34]: RMSE=np.sqrt(mean_squared_error(y_test,pred)) RMSE Out[34]: 209.71961740201198 In [39]: from sklearn import metrics from sklearn.metrics import mean_absolute_error In [40]: MSE=print(metrics.mean_squared_error(y_test,pred)) MSE 43982.31792324628 In [41]: R2S=print(metrics.r2_score(y_test,pred)) R2S 0.6886142448276894 In [42]: MAE=print(mean_absolute_error(y_test,pred)) 89.10334778841495 </pre>

10. ADVANTAGES & DISADVANTAGES

Advantages:

1. Food wastage will be minimized.
2. Simple and easy to use framework.

Disadvantages:

1. The output obtained may not be precised, due to the use of limited datasets.

11. APPLICATIONS

This project focuses on one food delivery client, which delivers food in many different cities through distribution networks and fulfillment centers.

12. CONCLUSION

The main moto behind this project is to reduce food wastage.The availability of the food items makes the society better. Our purposed model would definitely come handy to a company for predicting then number of food orders and help them to serve their customers better.

13. FUTURE SCOPE

1. Working on the frontend to make the framework more dynamic.
2. In the future, we also plan to improve forecasting accuracy and research on the efficiency of store management.

14. APPENDIX

SOURCE CODE:

home.html

```
<!DOCTYPE html>
<html lang="en">
<head>
<style>
  *{
    margin: 0;
    padding: 0;
    box-sizing: border-box;
    font-family: sans-serif;
  }
  body{
    background: url(./strw\ 2.jpg);
    background-repeat: no-repeat;
    background-size: cover;
  }
  .menubar{
    /* background-color: rgb(171, 171, 254);
    height: 5rem; */
    display: flex;
    align-items: center;
    justify-content: space-between;
    padding: 1% 3%;
  }
  .menubar ul{
    display: flex;
    list-style: none;
  }
```

```

.menubar ul li{
    padding: 5px 25px;
}
.menubar ul li a{
    text-decoration: none;
    font-family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS', sans-serif;
    font-weight: 500;
    font-size: 18px;
    color: white;
    padding: 2px 10px;
    /* background-color: white; */
    border-radius: 5px;
}
#spn{
    color: blue;
}
h1{
    color: white;
    letter-spacing: 2px;
}
.menubar ul li a:hover{
    color: blue;
    background-color: white;
}
#FD-content{
    border: 2px solid violet;
    height: 15rem;
    width: 28rem;
    float: right;
    margin-right: 5rem;
    transform: translateY(8rem);
    border-radius: 0px 20px 0px 20px;
}
h5{
    color: white;
    font-size: 1.9rem;
    padding-top: 1rem;
    padding-left: 2rem;
    line-height: 3rem;
}
#FD-content p{
    color: white;
    font-family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS', sans-serif;
    font-size: 1.5rem;
}

```

```

padding-top: 2rem;
padding-left: 2rem;
line-height: 2rem;
}

</style>
<title>Diagram page</title>
</head>
<body>
  <!--  -->
  <div class="menubar">
    <h1 id="headLine">Food <span id="spn">Demand</span> Forecaster</h1>
    <ul>
      <li><a href="./home.html" id="f1">Home</a> </li>
      <li><a href="./predict.html">Predict</a> </li>
      <li><a href="./login.html">Login</a> </li>
    </ul>
  </div>
  <div id="FD-content">
    <h5>Welcome To FOOD DEMAND FORECASTER!</h5>
    <p> We are happy to help you.
      <br> Lets get in and predict....
    </p>
  </div>

</body>
</html>

```

login.html

```

<!DOCTYPE html>

<html lang="en">

<head>

  <style>

    *{

      margin: 0;

      padding: 0;

      box-sizing: border-box;

```

```
    font-family: Verdana, Geneva, Tahoma, sans-serif;
}
form{
    border: 2px solid violet;
    width: 25%;
    transform: translateX(150%);
    text-align: center;
    margin-top: 7rem;
    height: 20rem;
    border-radius: 1.5rem;
    box-shadow: -1px 0px 9px 6px rgba(0,0,0,0.75);
    /* background-color: white; */
}
```

```
h1{
    padding-top: 1.5rem;
}
```

```
p{
    padding: 10px 0px;
    color:black;
}
```

```
input{
    border: none;
    border-bottom: 1px solid black;
    outline: none;
    font-size: 10px;
}
```

```
#emailid{
  transform: translateX(-4.8rem);
}
#passid{
  transform: translateX(-3.8rem);
}
form p a{
  text-decoration: none;
  font-size: 13px;
}

button{
  padding: 5px 30px;
  border-radius: 1.5rem;
  color: aqua;
  border: none;
  background-color: rgb(171, 106, 232);
  cursor: pointer;
}
#btn{
  padding-bottom: 1rem;
  transform: translateX(0.5rem);
}
i{

  padding-right: 8px;

}
p{
```

```
padding: 0.5rem 10px;
border-radius: 0.2rem;
color: white;
}
```

```
input{
padding: 0.5rem 10px;
border-radius: 0.2rem;

}
```

```
::placeholder{
text-align: center;
}
```

```
#emldiv,#pasdiv{
padding: 7px 0px;
}
```

```
#headLine
{
```

```
padding-top: 0.5rem;
```

```
color: beige;
font-family: Verdana, Geneva, Tahoma, sans-serif;
transform: translateX(1rem);
}
```

```
#fp a{
padding-left: 1rem;
color: white;
```



```
}
*{
  margin: 0;
  padding: 0;
  box-sizing: border-box;
  font-family: sans-serif;
}
body{
  background: url(../salad-gc76461915_640.jpg);
  background-repeat: no-repeat;
  background-size: cover;
  /* background-size: 80rem 40rem; */
}
.menubar{
  /* background-color: rgb(171, 171, 254);
  height: 5rem; */
  display: flex;
  align-items: center;
  justify-content: space-between;
  padding: 1% 3%;
}
.menubar ul{
  display: flex;
  list-style: none;
}
.menubar ul li{
  padding: 5px 25px;
}
```

```
.menubar ul li a{  
    text-decoration: none;  
    font-family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS', sans-serif;  
    font-weight: 500;  
    font-size: 18px;  
    color: white;  
    padding: 2px 10px;  
    /* background-color: white; */  
    border-radius: 5px;
```

```
}
```

```
#spn{  
    color: blue;
```

```
}
```

```
h2{  
    color:white;  
    margin-top: 2rem;
```

```
}
```

```
h1{  
    color: white;  
    letter-spacing: 2px;
```

```
}
```

```
.menubar ul li a:hover{  
    color: blue;  
    background-color: white;
```

```
}
```

```
#btn a{  
    text-decoration: none;
```

```

    color: white;
}
i{
    color: white;
}

</style>

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.2.0/css/all.min.css" />

    <title>Login page</title>
</head>
<body>
    <div class="menubar">
        <h1 id="headLine">Food <span id="spn">Demand</span> Forecaster</h1>
        <ul>
            <li><a href="./home.html" id="f1">Home</a> </li>
            <li><a href="./predict.html">Predict</a> </li>
            <li><a href="./login.html">Login</a> </li>
        </ul>
    </div>
    <form action="#">
        <h2>LOGIN</h2>
        <div id="emldiv"> <p id="emailid"> Email</p>
            <i class="fa-solid fa-user"></i><input type="text" id="logiuser" placeholder="Enter Your Email"></div>
            <div id="pasdiv"> <p id="passid"> Password</p>
                <i class="fa-solid fa-lock"></i><input
type="password" id="logiuser" placeholder="Enter Your Password"> </div>
                <p id="fp"><a href="#">Forgot Password?</a></p>
                <div id="btn"> <button type="submit"><a href="./diagram.html">
Login</a></button></div>

```

```
</form>
```

```
</body>
```

```
</html>
```

predict.html

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
  <style>
```

```
    *{
```

```
      margin: 0;
```

```
      padding: 0;
```

```
      box-sizing: border-box;
```

```
      font-family: sans-serif;
```

```
    }
```

```
    body{
```

```
      background: url(/noodles-g7f0cbaa1c_640.jpg);
```

```
      background-repeat: no-repeat;
```

```
      background-size: cover;
```

```
    }
```

```
    .menubar{
```

```
      /* background-color: rgb(171, 171, 254);
```

```
      height: 5rem; */
```

```
      display: flex;
```

```
      align-items: center;
```

```
    justify-content: space-between;
    padding: 1% 3%;
}
.menubar ul{
    display: flex;
    list-style: none;
}
.menubar ul li{
    padding: 5px 25px;
}
.menubar ul li a{
    text-decoration: none;
    font-family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS', sans-serif;
    font-weight: 500;
    font-size: 18px;
    color: white;
    padding: 2px 10px;
    /* background-color: white; */
    border-radius: 5px;
}
#spn{
    color: blue;
}
h1{
    color: white;
    letter-spacing: 2px;
}
.menubar ul li a:hover{
```

```
color: blue;
background-color: white;
}

*{
    margin: 0;
    padding: 0;
    box-sizing: border-box;
}

select{
    transform: translateX(7rem);
    padding-right: 7.5rem;
    padding-bottom: 0.2rem;
    padding-top: 0.2rem;
    padding-left: 0.2rem;
}

input{
    transform: translateX(7rem);
    padding-right: 3rem;
    padding-bottom: 0.2rem;
    padding-top: 0.2rem;
    outline: none;
}

.wholeForm{
    height: 39rem;
    width: 30rem;
    border-radius: 3px;
    border: 2px solid rgb(171, 106, 232);
    transform: translateX(31rem);
}
```

```
#lb{
padding-left: 7rem;
color: white;
}
h1{
text-align: center;
font-family: Verdana, Geneva, Tahoma, sans-serif;
}
```

```
#sub{
/* background-color: brown; */
background-color: rgb(241, 0, 0);
border-radius: 1rem;
transform: translateX(11rem);
cursor: pointer;
```

```
height: 1.5rem;
width: 6rem;
font-family: Verdana, Geneva, Tahoma, sans-serif;
}
```

```
#sub a{
text-decoration: none;
color: white;
padding-left: 1.6rem;
```

```
}
```

```

h3{
    color: white;
    font-size: 25px;
    text-align: center;
    padding-top: 8px;

}

form{
    transform: translateY(-2rem);
}

</style>

<title>Predict</title>
</head>

<body>
    <div class="menubar">
        <h1 id="headLine">Food <span id="spn">Demand</span> Forecaster</h1>
        <ul>
            <li><a href="./home.html" id="f1">Home</a> </li>
            <li><a href="./predict.html">Predict</a> </li>
            <li><a href="./login.html">Login</a> </li>
        </ul>
    </div> <br> <br>
    <div class="wholeForm">
        <h3>FORECASTING MODEL</h3> <br> <br> <br>
        <form action="">
            <label for="areaName" id="lb">Area Name :</label> <br>
            <input type="text"> <br> <br>

```


<label for="cityName" id="lb">City Name :</label>

<input type="text">

<label for="area" id="lb">City Code :</label>

<input type="text">

<label for="meals" id="lb">Category : </label>

<select id="meals">

<option value="veg">-----</option>

<option value="veg" id="bev">Beverages</option>

<option value="non-veg" id="ric">Rice Bowl</option>

<option value="veg" id="str">Starters</option>

<option value="non-veg" id="pta">Pasta</option>

<option value="veg" id="bir">Biryani</option>

<option value="non-veg" id="sea">Sea Food</option>

<option value="veg" id="piz">Pizza</option>

<option value="non-veg" id="sal">Salads</option>

<option value="non-veg" id="des">Dessert</option>

<option value="veg" id="cak">Cake</option>

</select>

<label for="Ingredients" id="lb">Cuisine : </label>

<select id="Ingredients">

<option value="veg">-----</option>

<option value="veg" >Thai</option>

<option value="veg">Indian</option>

<option value="non-veg">Continental</option>

</select>

<label for="Week" id="lb"> Week :</label>


```
<input type="text" placeholder="Enter the number of weeks" id="resweek"> <br><br>
<label for="check" id="lb">Checkout price</label> <br>
<input type="text"> <br><br>
<label for="check" id="lb">Base price</label> <br>
<input type="text"> <br><br>

<p id="sub"> <a href="/predict result.html">Predict</a></p>
<br><br>
```

```
</form>
</div>
</body>

</html>
```

predict result.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <style>
    *{
      margin: 0;
      padding: 0;
      box-sizing: border-box;
      font-family: sans-serif;
    }
  body{
```

```
background: url(./snack-g4676300ca_1280.jpg);
background-repeat: no-repeat;
background-size: cover;
}
.menubar{
  /* background-color: rgb(171, 171, 254);
  height: 5rem; */
  display: flex;
  align-items: center;
  justify-content: space-between;
  padding: 1% 3%;
}
.menubar ul{
  display: flex;
  list-style: none;
}
.menubar ul li{
  padding: 5px 25px;
}
.menubar ul li a{
  text-decoration: none;
  font-family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS', sans-serif;
  font-weight: 500;
  font-size: 18px;
  color: white;
  padding: 2px 10px;
  /* background-color: white; */
  border-radius: 5px;
```

```
}  
#spn{  
    color: blue;  
}  
h1{  
    color: white;  
    letter-spacing: 2px;  
}  
.menubar ul li a:hover{  
    color: blue;  
    background-color: white;  
}  
.output p{  
    color: white;  
    text-align: center;  
    font-family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS', sans-serif;  
    font-size: 18px;  
    letter-spacing: 2px;  
}  
h4{  
    color: white;  
    text-align: center;  
    font-size: 23px;  
    letter-spacing: 3px;  
}  
#resultno{  
    color: yellow;  
}  
.count{
```

```
color:yellow;
font-size: 20px;
font-weight: 300;
}
```

```
#biryani{
    font-size: 23px;
}
```

```
.result-hd a{
    text-decoration: none;
    color: white;
}
```

```
</style>
```

```
<title>Result</title>
```

```
</head>
```

```
<body>
```

```
<div class="menubar">
```

```
<h1 id="headLine">Food <span id="spn">Demand</span> Forecaster</h1>
```

```
<ul>
```

```
<li> <a href="./home.html" id="f1">Home</a> </li>
```

```
<li> <a href="./predict.html">Predict</a> </li>
```

```
<li> <a href="./login.html">Login</a> </li>
```

```
</ul>
```

```
</div> <br><br> <br>
```

```
<div class="result-hd">
```

```
<h4>For the Next <span id="resultno"> 10 </span>Weeks </h4></div> <br><br>
```

```
<div class="output" id="opt">
```

```
    <p> The Predicted Output : <span class="count" data-val="3578">0000 </span>
<span id="pizz">Pizza</span></p>
</div>
```

```
<script>
    var value = document.querySelectorAll(".count");

value.forEach((valueDisplay) => {
    let start = 0;
    let end = parseInt(valueDisplay.getAttribute("data-val"));
    console.log(end);
    let ans = setInterval(() => {
        start += 1;
        valueDisplay.textContent = start;
        if (start == end) {
            clearInterval(ans);
        }
    },1);
});
```

```
</script>
```

```
</body>
```

```
</html>
```

fooddemandapp.py

```
# import the necessary packages
```

```
import pandas as pd
```

```
import numpy as np
```

```
import pickle
```

```
import os
```

```

from flask import Flask,request, render_template
app=Flask(__name__,template_folder="templates")
@app.route('/', methods=['GET'])
def index():
    return render_template('home.html')
@app.route('/home', methods=['GET'])
def about():
    return render_template('home.html')
@app.route('/pred',methods=['GET'])
def page():
    return render_template('upload.html')
@app.route('/predict', methods=['GET', 'POST'])
def predict():
    print("[INFO] loading model...")
    model = pickle.load(open('fdemand.pkl', 'rb'))
    input_features = [float(x) for x in request.form.values()]
    features_value = [np.array(input_features)]
    print(features_value)

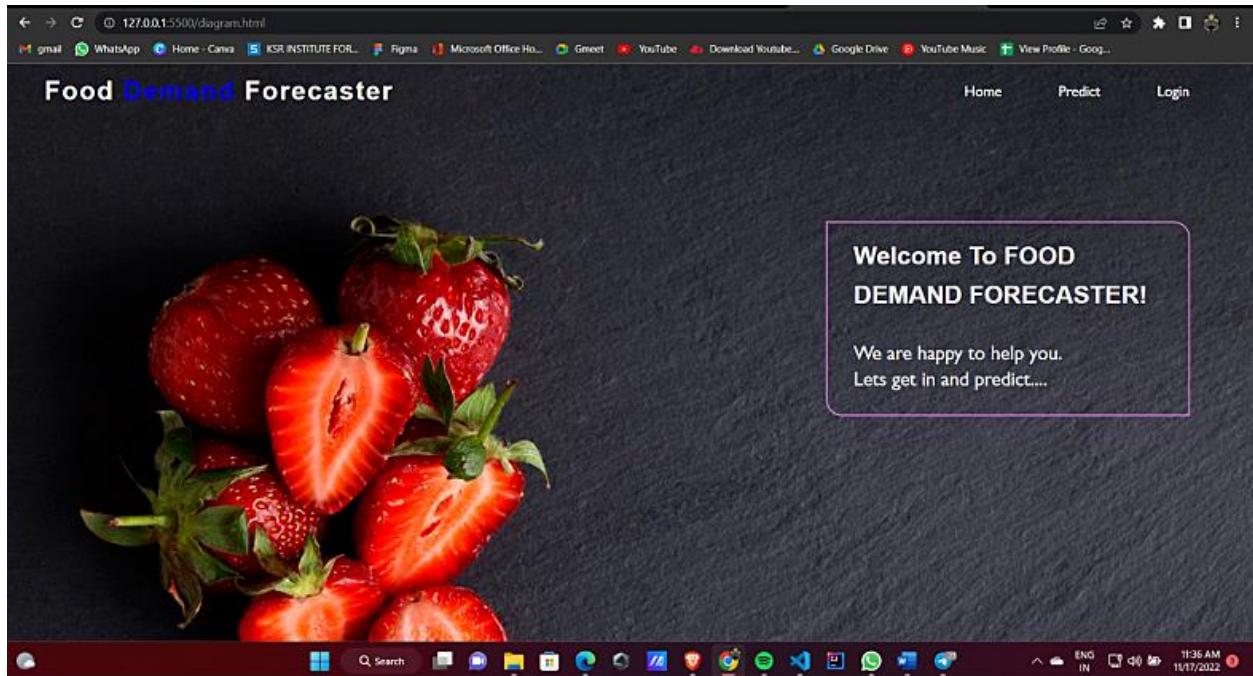
    features_name = ['homepage_featured', 'emailer_for_promotion', 'op_area', 'cuisine',
        'city_code', 'region_code', 'category']
    prediction = model.predict(features_value)
    output=prediction[0]
    print(output)
    return render_template('upload.html', prediction_text=output)

if __name__ == '__main__':
    app.run(debug=False)

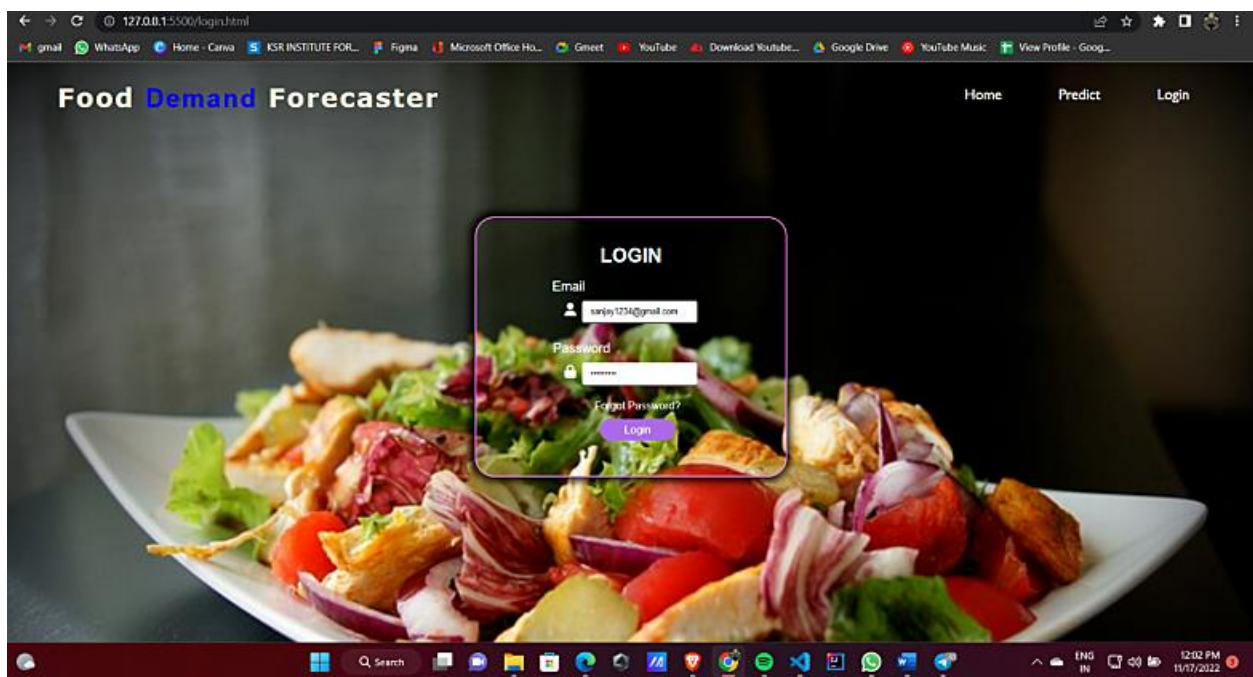
```

SCREENSHOTS

HOME PAGE



LOGIN PAGE



PREDICT PAGE

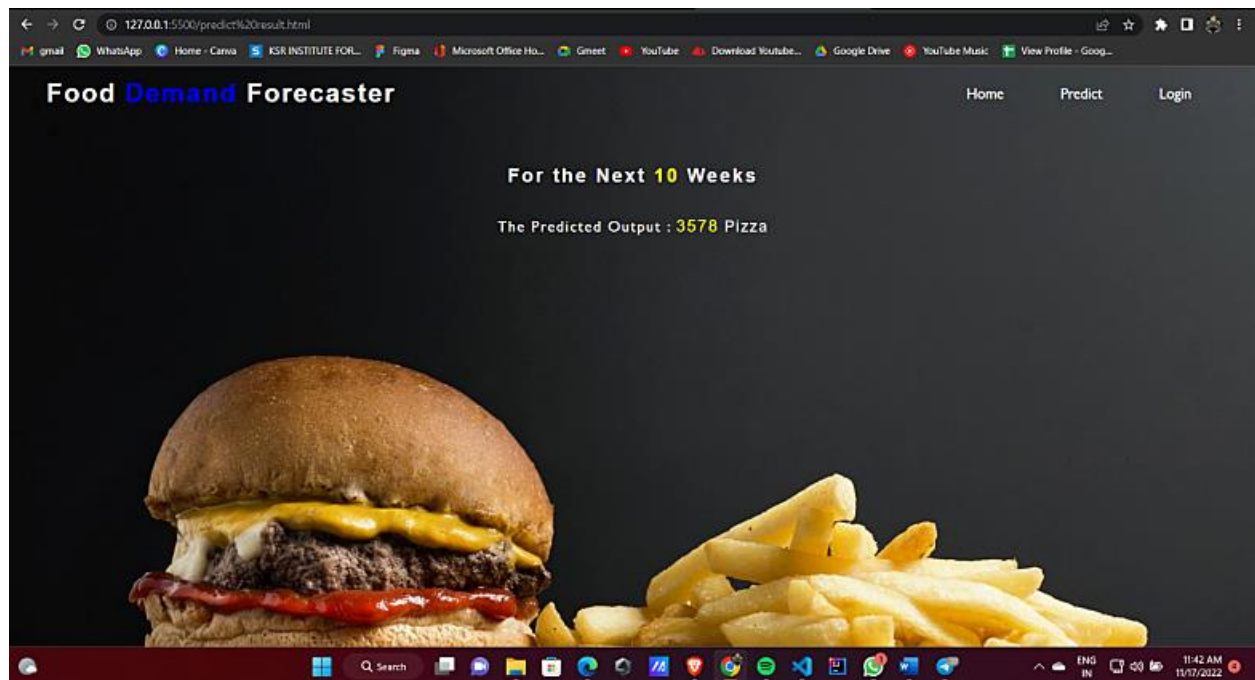


The screenshot shows a web browser window with the URL `127.0.0.1:5500/predict.html`. The page title is "Food Demand Forecaster". The background image is a bowl of food containing spaghetti, tomatoes, and peppers. A white form titled "FORECASTING MODEL" is centered on the page. The form contains the following fields and values:

Field	Value
Area Name	Chithamthur
City Name	Tiruchengode
City Code	209
Category	Pizza
Cuisine	Indian
Week	10
Checkout price	299
Base price	250

A red "Predict" button is located at the bottom right of the form. The browser's taskbar at the bottom shows the Windows logo, search bar, and various application icons. The system clock in the bottom right corner displays "11:42 AM 10/17/2022".

PREDICT RESULT PAGE



GITHUB LINK

<https://github.com/IBM-EPBL/IBM-Project-10288-1659159051>

PROJECT DEMONSTRATION VIDEO LINK

<https://youtu.be/wG9YfDhaZm0>

PROJECT PRESENTATION VIDEO LINK

<https://www.youtube.com/embed/aFCYgt82aQM>