Project report on

DemandEst - AI Powered Food Demand Forecaster

Prepared by

Gokul L - B.TECH(IT) - [727619BIT077]

Tharun Balaji AP-B.TECH(IT)-[727619BIT061]

ChethanKumar C-B.TECH(IT)-[727619BIT057]

Logesh Kumar M-B.TECH(IT)-[727619BIT055]

BalaMurugan P- B.TECH(IT) - [727619BIT063]

CONTENTS

1. INTRODUCTION

1.1 OVERVIEW

A food delivery service to be 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. The application of AI in the food industry has been growing for years due to various reasons such as food sorting, classification and prediction of the parameters, quality control, and food safety. Expert system, fuzzy logic, ANN, adaptive neuro-fuzzy inference system (ANFIS), and machine learning are among the popular techniques that have been utilized in the food industries.

1.2 PURPOSE

The use of forecasting methods is nowadays regarded as a business ally since it supports both the operational and the strategic decision making processes. This paper is based on a research project aiming the development of demand forecasting models for a company (designated here by PR) that operates in the food business, more specifically in the delicatessen segment. In particular, we focused on demand forecasting models that can serve as a tool to support production planning and inventory management at the company. The analysis of the company ' s operations led to the development of a new demand forecasting tool based on a combination of forecasts, which is now being used

2. LITERATURE SURVEY

2.1 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. Anyways the recruiting of staff members at the fulfillment center is an prospect wherein theprediction of orders would be beneficial. Although this is a process that can be done manual process.

2.2 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., & Häkkilä, 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).

- Clarke, S. (2006). Transformation Lessons from Coca-Cola Enterprises Inc.: Managing the Introduction of a Structured Forecast Process. Foresight: The International Journal of Applied Forecasting, (4), 21-25.

2.3 PROBLEM STATEMENT DEFINITION

I am	Customer spends time to decide the food to order,website to order
I am trying	Customer expects the easier way to order
But	Inconsistent food delivered Problem with payments
Because	Difference in pricing Failed payment issue
Feel	Good-Easy to order and get thr food easily Bad-issue on qulity,delayed delivery

Example:

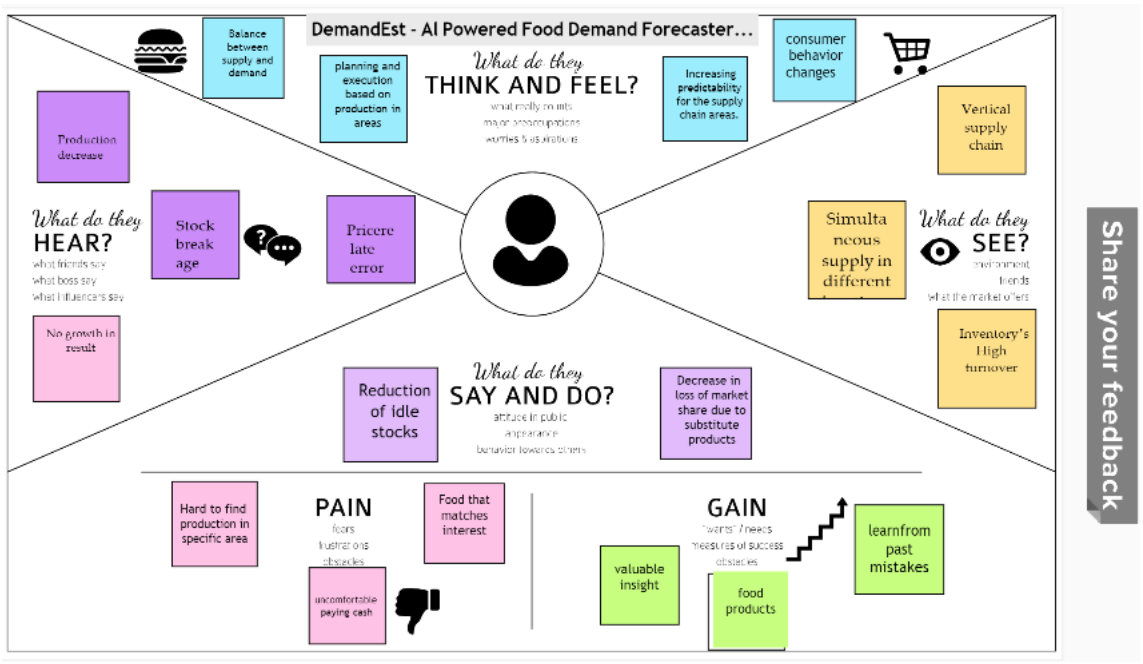
Problem Statement (PS)	**I am (Customer)**	**I'm trying to**	**But**	**Because**	**Which makes me feel**
PS-1	Tourist	Tracking the delivery status	Unable to guess the current location.	network error.	Literally Hunger.
PS-2	Food app custom er	Expecting to get the food quickly	Due to transp ortati on latenc y	On vacations.	Make me too cancel the food.

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.

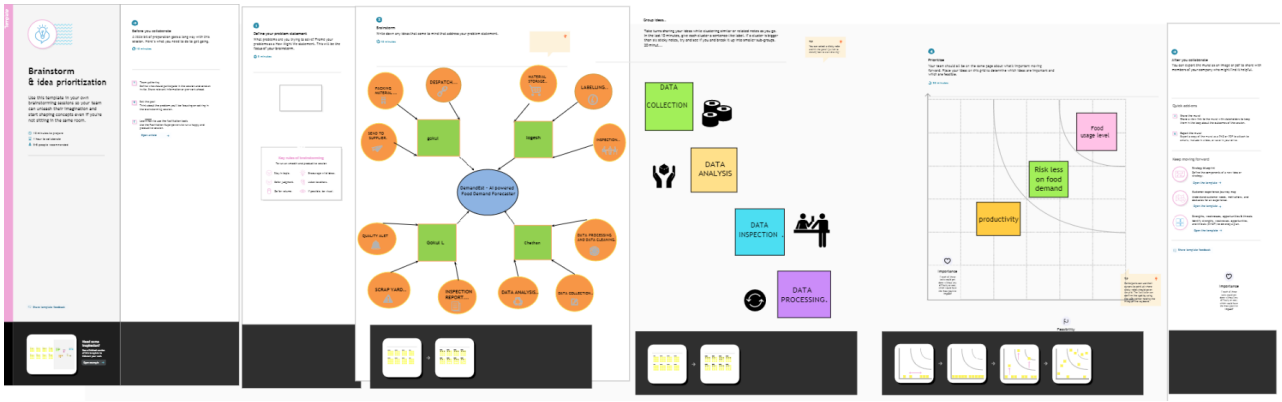
Traditional empathy maps are split into 4 quadrants (Says, Thinks, Does, and Feels), with the user or persona in the middle. Empathy maps provide a glance into who a user is as a whole and are not chronological or sequential.



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

Everyone in a design team should have a *clear* definition of the target problem. They typically gather for a brainstorming session in a room with a large board/wall for pictures/Post-Its. A good mix of participants will expand the experience pool and therefore broaden the idea space.



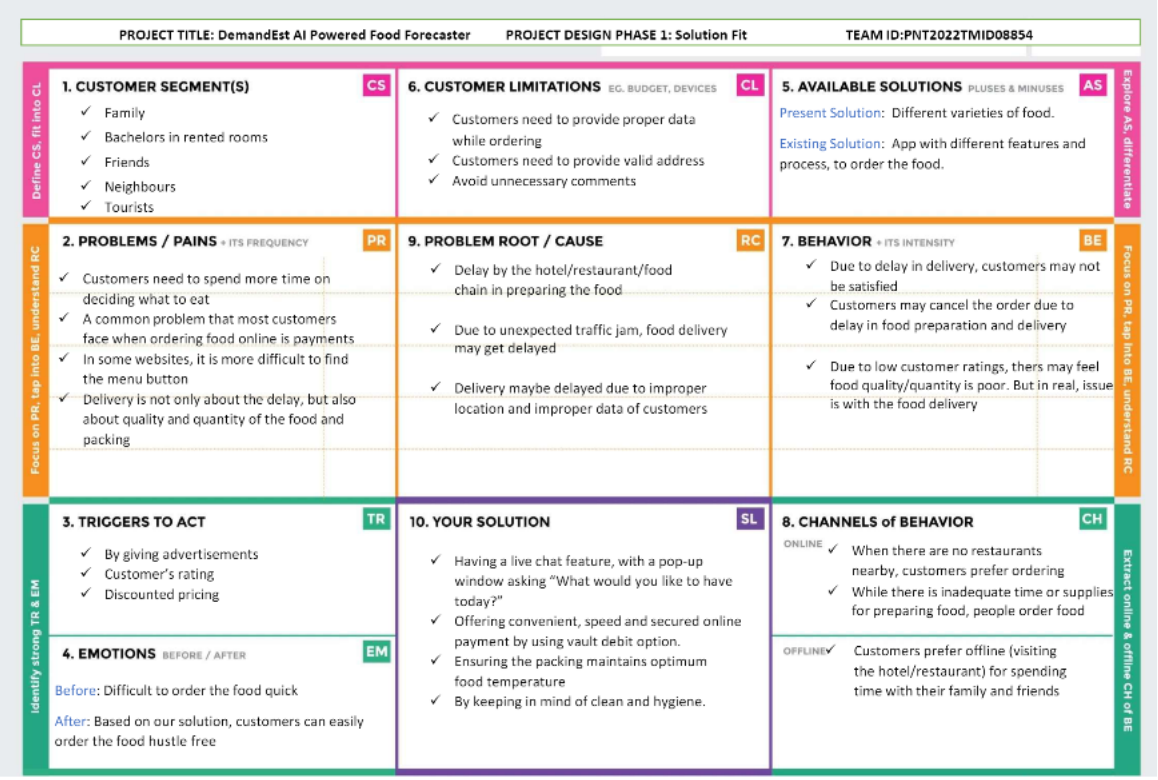
3.2 Proposed Solution

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.	Parameter	Description
1.	Problem Statement (Problem to be solved)	➢ Estimate profit & loss ➢ Increase the order processing speed ➢ Provide a great customer experience ➢ Reduce the manual redundant work
2.	Idea / Solution description	➢ Friendly UI ➢ Cart management ➢ Compatibility with other tools ➢ Order management
3.	Novelty / Uniqueness	➢ Sales tracking ➢ Automated analysis ➢ Increased productivity ➢ Healthy customer relationship
4.	Social Impact / Customer Satisfaction	➢ Hygiene ➢ Reliability ➢ Fast service ➢ Low downtime
5.	Business Model (Revenue Model)	➢ Order management ➢ Time management ➢ Cost management ➢ Menu management ➢ Customer management
6.	Scalability of the Solution	➢ Based on quality ➢ Based on quantity ➢ Based on maintenance

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.



4. REQUIREMENT ANALYSIS

4.1 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	User Registration	Registration through Website app Registration through Gmail Registration through Mobile app
FR-2	User Confirmation	Confirmation via Email Confirmation via phone number
FR-3	User profile	Complete your profile by using login registration process.
FR-4	User search	Search your favourite item to put delivery in this product
FR-5	User preference	Search for food delivery based on their location preference.
FR-6	Result	Finally you can reach your delivery items.

4.2 Non-Functional requirements

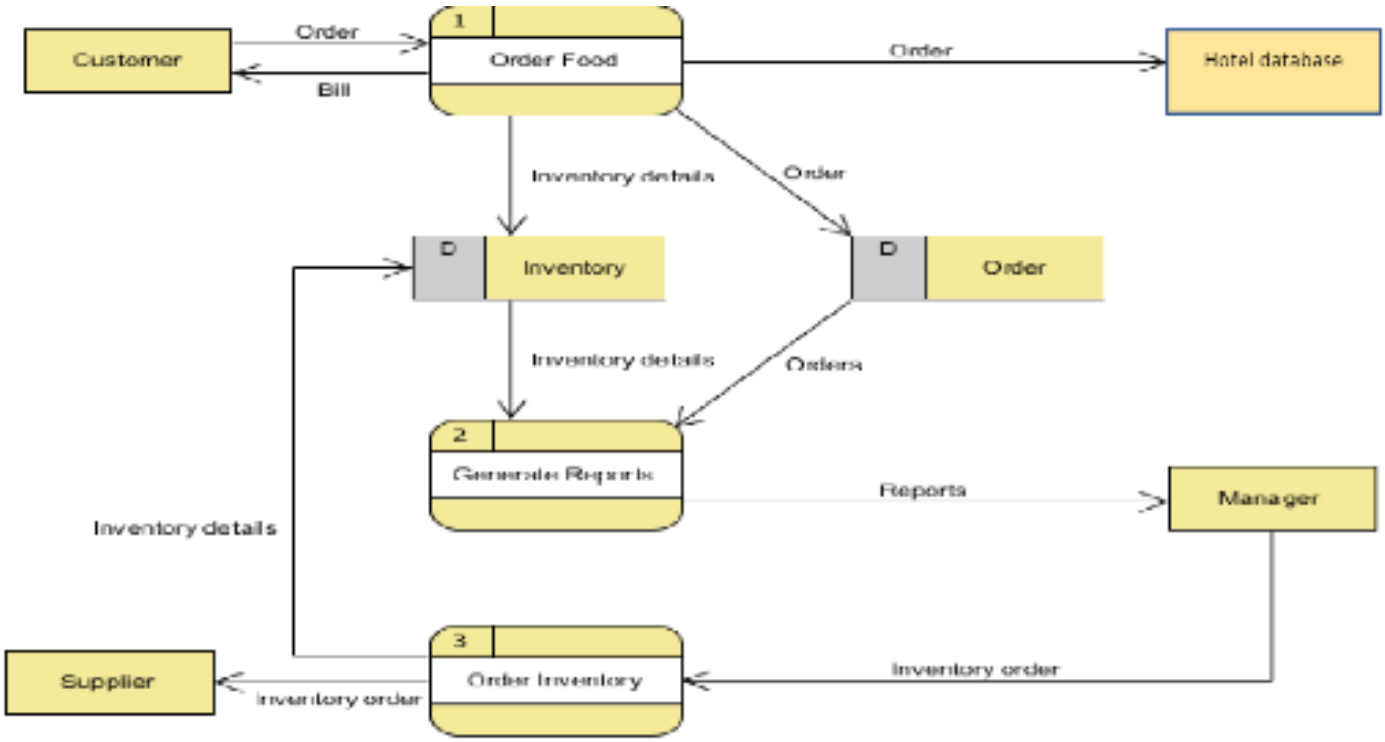
In systems engineering and requirements engineering, a non-functional requirement (NFR) is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviours.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Filters the AI based on the user profile.
NFR-2	**Security**	User details are secured from unauthorized parties
NFR-3	**Reliability**	The user can find the ordered items based on their preferred items.
NFR-4	**Performance**	The website will provide the list of orders within 60 seconds.
NFR-5	**Availability**	Users can access the website for anytime.
NFR-6	**Scalability**	The order of the solution will be helpful for using ratings and feedback.

5. PROJECT DESIGN

5.1 Data Flow Diagrams

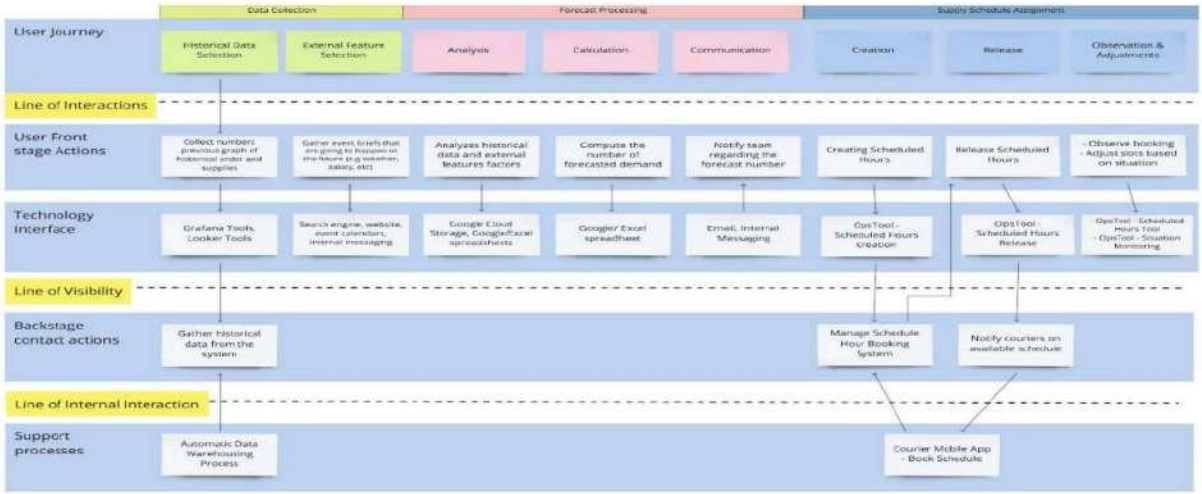
A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



5.2 Solution & Technical Architecture

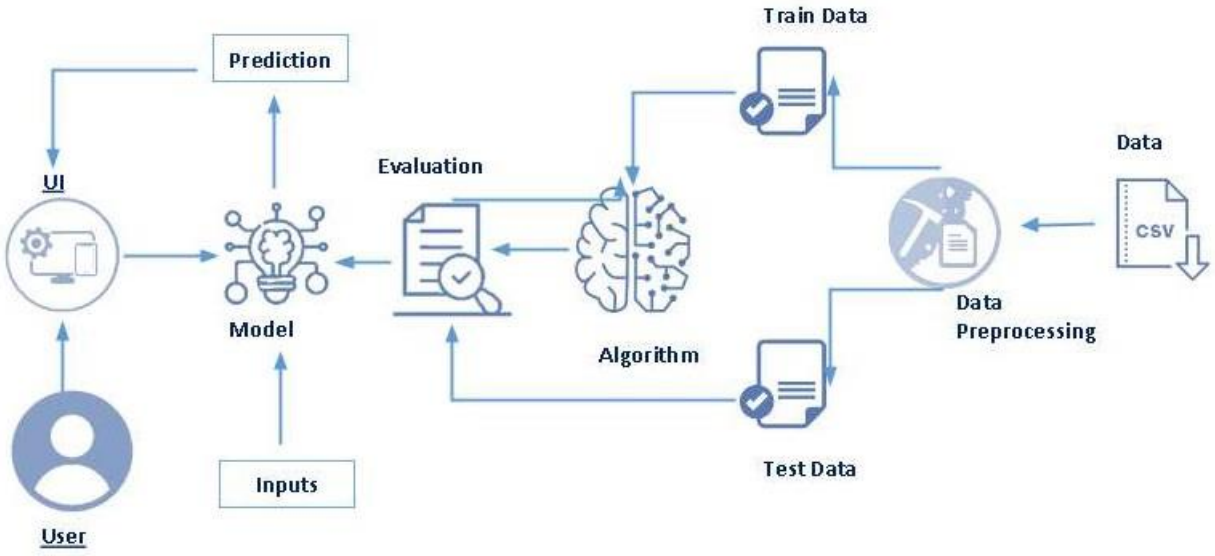
Solution Architecture:

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 system-relevant requirements are met.



5.3 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
		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
		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
		USN-4	As a user, I can register for the application through Gmail	I can receive confirmation Gmail & click confirm	Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password	I can receive confirmation email & password	High	Sprint-1
	Dashboard	USN-6	AS a user I can check the facilities of the Dashboard	I can receive and help to more support in the Dashboard	High	Sprint-2
Customer (Web user)	Web user	USN-7	As a user I can use this website to learn more about this quantity method	I can receive confirmation & click to check the quantity	Low	Sprint-2
Customer Care Executive	Customer support	USN-8	As a user I can need to want an customer support care	I can receive your queries accept the confirmation	Medium	Sprint-1
Administrator	Admin	USN-9	AS a user I can see your varieties of the products	I can receive your product to check confirmation	Medium	Sprint-2
		USN-10	As a user I can see your products of the Ingredients	I can receive your Ingredients to check confirmation	High	Sprint-3
		USN-11	AS a user I can compare your product with another administrator website	I can receive you to compare another product confirmation	Medium	Sprint-2
Customer(order)	Order	USN-12	As a user I can order to delivery a product of items	I can receive your order to verify the delivery of the product	High	Sprint-1

6. PROJECT PLANNING & SCHEDULING

6.1 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 Requirements (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 abeginning Spanish class before signing up for Spanish, then it's aprerequisite.	10	Low	Gokul Logesh Balamurugan
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	Chethan Gokul Logesh Ba;lamurugan
Sprint-2	Data Pre-Processing. Importing the libraries	USN-3	In this post I am going to walk through the implementation of Data Pre-processing methods using Python.	5	High	Chethan Tharun Logesh Balamurugan
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.	5	High	Tharun Chethan Balamurugan Logesh
Sprint-2	Checking for null values. Reading andmerging.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 thefollowing code to create and export three data frames to CSV files: #create three data frames df1 <- data ... Step 2:	2	Medium	Tharun Balamurugan Logesh
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	Tharun Balamurugan Logesh
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	Tharun Balamurugan Logesh
Sprint-3	Model Building	USN-8	What the person using the product wants to Be able to do. A traditional requirement focuses on functionality.	10	High	Gokul Chethan Tharun
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	Gokul Chethan Tharun
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	Gokul Chethan Tharun
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	Gokul Chethan Logesh
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	Gokul Chethan Logesh

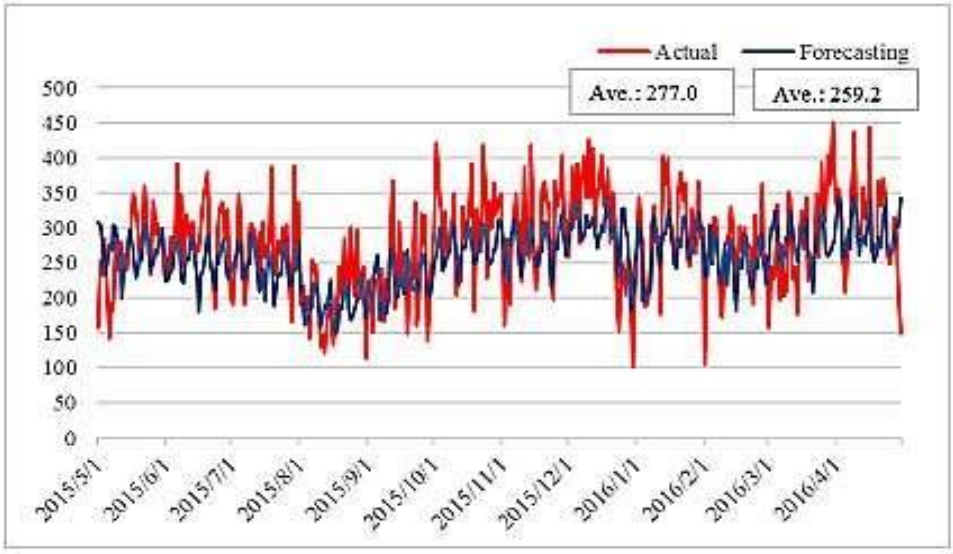
6.2 Sprint Delivery Schedule

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	7	6 Days	24 Oct 2022	29 Oct 2022	7	29 Oct 2022
Sprint-2	4	9 Days	30 Oct 2022	07 Nov 2022	4	05 Nov 2022
Sprint-3	6	7 Days	08 Nov 2022	14 Nov 2022	6	12 Nov 2022
Sprint-4	2	7 Days	15 Nov 2022	21 Nov 2022	2	19 Nov 2022

6.3 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 start-ups and enterprises. Software teams build better with Jira Software, the #1 tool for agile teams.



7. CODING & SOLUTIONING

7.1 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, center id, meal id, checkout price,

base price, emailer for promotion, homepage featured. This file is used for testing.

- fulfilment_center_info.csv: Contains information of each fulfilment 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/...)

7.2 Libraries Used

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

7.3 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.

7.4 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.

7.5 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.

7.6 Evaluation Metric

The evaluation metric for this competition is 100*RMSLE where RMSLE is Root of Mean Squared Logarithmic Error across all entries in the test set.

7.7 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'nt 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.

7.8 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 lease RMSLE of 0.5237

8. TESTING

8.1 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	1.Check all the text boxes, radio buttons, 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	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	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
Network Accessing device	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 text box 4.Enter valid password in password text box 5.Click on login button
Network Accessing device	1.Log in 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	1.Test Case ID. 2.Test Description. 3.Assumptions and Pre-Conditions. 4.Test Data.
Network Accessing device	1. Keep things simple and transparent. 2. Make test cases reusable. 3. Peer review is important. 4. Keep test cases IDs unique.
Network Accessing device	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	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 text box b.password text box c.Login button with orange colour d.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 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 password: FDF123678686786876876	Application should show 'Incorrect email or password ' validation message.	Working as expected	Pass
Username: jdk@gmail password: FDF123	Everything that a customer expects from a product, service or organisation.	Working as expected	Pass
Username: jdk@gmail password: FDF123	It should be made clear how many days a delivery might take to process.	Working as expected	Pass
Username: jdk@gmail password: FDF123	It will be commercially accountable for budgeting and financial management and will need to plan, organise amd direct all hotel services.	Working as expected	Pass

8.2 User Acceptance Testing

User Acceptance Testing (UAT), which is performed on most UIT projects, sometimes called beta testing or end-user testing, is a phase of software development in which the software is tested in the "real world" by the intended audience or business representative.

Defect Analysis:

Resolution	Severity1	Severity2	Severity3	Severity4	Subtotal
By Design	9	3	3	2	17
Duplicate	1	1	3	2	7
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	23	9	12	27	71

Test Case Analysis:

Section	TotalCases	Not Tested	Fail	Pass
PrintEngine	10	0	0	10
ClientApplication	46	0	0	46
Security	4	0	0	4
OutsourceShipping	2	0	0	2
ExceptionReporting	8	0	0	8
FinalReportOutput	6	0	0	6
VersionControl	2	0	0	2

9. RESULTS

9.1 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** In [33]: `from sklearn.metrics import mean_squared_error` In [34]: `RMLSE=np.sqrt(mean_squared_error(y_test,pred))` `RMLSE` 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

2.	Tune the Model	**Hyperparameter Tuning -** RMSLE- 52.85812511759974 avg R-squared- 0.123 MSE: -64230.918	

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>
   <meta charset="UTF-8">
   <meta http-equiv="X-UA-Compatible" content="IE=edge">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>DemandEst - AI powered Food Demand Forecaster</title>
   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
     rel="stylesheet"
        integrity="sha384-
     EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC"
     crossorigin="anonymous">
   <style>
      .bg-for-all{
         background-color: #08AEEA;
         background-image: linear-gradient(0deg, #08AEEA 0%, #2AF598 100%);
      }

      .bg-for-nav{
         background-color: #21D4FD;
         background-image: linear-gradient(19deg, #21D4FD 0%, #B721FF 100%);
      }

   </style>
</head>
<body>
   <div>
      <div class="card" style="border:none;">
         <div class="card-header h4 text-light p-3 bg-for-nav">
            DemandEst - AI powered Food Demand Forecaster
         </div>
         <div class="container p-4">
            <div class="row">
              <div class="col-md-6">
                <img src="https://images.pexels.com/photos/1640772/pexels-photo-
     1640772.jpeg?auto=compress&cs=tinysrgb&w=600"
                   class="img-fluid p-4 rounded-start" alt="...">
              </div>
               <div class="col-md-6">
```

```
            <div class="card-body container">
              <h2 class="card-title">know your food supply for 10 weeks?</h2>
              <p class="card-text col">
                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 dem and.
                Too much inventory in the warehouse means more risk of wastage, and not
enough could lea d to
                out-of-stocks - and push customers to seek solutions from your competitors.
                The replenishment of 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.
              </p>
              <br>
              <a href="#pred_form" class="btn-lg bg-for-nav text-light rounded-pill text-
decoration-none">start predicting</a>
            </div>
          </div>
        </div>

        <!-- background-color: #FFDEE9;
background-image: linear-gradient(0deg, #FFDEE9 0%, #B5FFFC 100%);
 -->
 <!--  style="background-repeat: repeat, repeat;background-image:linear-gradient(19deg, #21D4FD
0%, #B721FF 100%), url('https://i.gifer.com/Y3ie.gif');background-blend-mode:
multiply;opacity: 0.5" -->
        <div class="card border-warning mb-3">
          <div class="card-header h4"> Results </div>
          <div style="padding:70px 0;text-align:center;">
          {% if prediction_text %}
          <div class="card-body text-center">
            <p class="h2">Predicted number of food orders: {{ prediction_text }}</p>
          </div>
          {% endif %}

          {% if not prediction_text %}
          <button class="btn" type="button">
            <span class="spinner-grow spinner-grow-sm" role="status" aria-
hidden="true"></span>
            <span class="spinner-grow spinner-grow-sm" role="status" aria-
hidden="true"></span>
```

```
                waiting for prediction...
              </button>
              {% endif %}
              </div>
            </div>
          </div>
        </div>
        <!--  -->

        <!-- svg -->
        <svg style="background-color:#28F19E;border-style:none;" width="100%" height="70"
        viewbox="0 0 100 100" preserveAspectRatio="none">
          <path d="M0,0  L110,0C35,150 35,0 0,100z" fill="#ffffff" />
        </svg>
        <!-- svg -->

<!-- upload page -->
<div class="container-fluid">

<div class="row card p-4 text-white bg-for-all" id="pred_form" style="min-
        height:568px;background-color: #26EEA0;border:none">
        <div class="col h2 text-center  p-4">
          Get your number of food orders?
        </div>

        <form class="col row g-4 needs-validation" action="{{ url_for('predict') }}" method="POST">
          <div class="col-md-4">
            <label for="validationCustom01" class="form-label fs-5">homepage_featured</label>
            <select class="form-select" id="homepage_featured" name="homepage_featured"
        required>
              <option value="">----</option>
              <option value="0">No</option>
              <option value="1">Yes</option>
            </select>
          </div>
          <div class="col-md-4">
            <label for="validationCustom01" class="form-label fs-5">emailer_for_promotion</label>
            <select class="form-select" id="emailer_for_promotion" name="emailer_for_promotion"
        required>
              <option value="">----</option>
              <option value="0">No</option>
              <option value="1">Yes</option>
            </select>
          </div>
```

```
    <div class="col-md-4">
      <label for="formGroupExampleInput" class="form-label fs-5">Enter your op_area</label>
      <input type="text" class="form-control" id="formGroupExampleInput" name="op_area"
        placeholder="like op_area=27" required>
    </div>

    <div class="col-md-4">
      <label for="formGroupExampleInput" class="form-label fs-5">Enter your region
code</label>
      <input type="text" class="form-control" id="formGroupExampleInput"
name="region_code"
        placeholder="Enter region_code" required>
    </div>

    <div class="col-md-4">
      <label for="formGroupExampleInput" class="form-label fs-5">Enter your city
code</label>
      <input class="form-control" type="text" name="city_code" placeholder="Enter city_code"
required>
    </div>

    <div class="col-md-4">
      <label for="validationCustom01" class="form-label fs-5">select the food cuisine</label>
      <select class="form-select" id="cuisine" name="cuisine" required>
        <option value="">----</option>
        <option value="0">Continental</option>
        <option value="1">Indian</option>
        <option value="2">Italian</option>
        <option value="3">Thai</option>
      </select>
    </div>

    <div class="col-md-4">
      <label for="validationCustom01" class="form-label fs-5">select the food
category</label>
      <select class="form-select" id="category" name="category" required>
        <option value="">----</option>
        <option value="0">Beverages</option>
        <option value="1">Biryani</option>
        <option value="2">Desert</option>
        <option value="3">Extras</option>
        <option value="4">Fish</option>
        <option value="5">Other Snacks</option>
        <option value="6">Pasta</option>
```

```
                <option value="7">Pizza</option>
                <option value="8">Rice Bowl</option>
                <option value="9">Salad</option>
                <option value="10">Sandwich</option>
                <option value="11">Seafood</option>
                <option value="12">Soup</option>
                <option value="13">Starters</option>
              </select>
            </div>

            <div></div>
            <div class="col-md-4 d-grid gap-2 col-6 mx-auto">
              <input class="btn btn-dark btn-lg rounded-pill" type="submit" value="predict" data-bs-
        toggle="modal"
                data-bs-target="#exampleModal">
            </div>
          </form>
        </div>

        </div>
      </div>


      <!-- upload page-->

      <!-- about page -->
      <!-- <div class="container px-4 py-5" id="featured-3">
        <h2 class="pb-2 border-bottom">About us</h2>
        <div class="row g-4 py-5 row-cols-1 row-cols-lg-3">
          <div class="feature col">
            <div class="feature-icon bg-primary bg-gradient">

            </div>
            <h2>Featured title</h2>
            <p>Paragraph of text beneath the heading to explain the heading. We'll add onto it with
        another sentence and
                probably just keep going until we run out of words.</p>
            <a href="#" class="icon-link">
              Call to action

            </a>
          </div>

          <div class="feature col">
```

```
        <div class="feature-icon bg-primary bg-gradient">

        </div>
        <h2>Featured title</h2>
        <p>Paragraph of text beneath the heading to explain the heading. We'll add onto it with another sentence and
          probably just keep going until we run out of words.</p>
        <a href="#" class="icon-link">
          Call to action

        </a>
      </div>

      <div class="feature col">
        <div class="feature-icon bg-primary bg-gradient">

        </div>
        <h2>Featured title</h2>
        <p>Paragraph of text beneath the heading to explain the heading. We'll add onto it with another sentence and
          probably just keep going until we run out of words.</p>
        <a href="#" class="icon-link">
          Call to action

        </a>
      </div>
    </div>
</div> -->
<!-- about page -->

    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"
      integrity="sha384-MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"
      crossorigin="anonymous"></script>
</body>
</html>
```

app.py

```
import numpy as np
import pickle
import os
from flask import Flask, request, render_template
import requests
import json

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('/predict', methods=['GET', 'POST'])
def predict():
    print('[INFO] Loading model…')
    #print(request.form.values())
    input_features = [float(x) for x in request.form.values()]
    print(input_features)

    API_KEY = "RSUKnz_dvhPrn3OXEeNdh0hZHTYSaexP0OEFqJgSFU9a"
    token_response = requests.post('https://iam.cloud.ibm.com/identity/token', data={"apikey":
                                                        API_KEY, "grant_type":
    'urn:ibm:params:oauth:grant-type:apikey'})
    mltoken = token_response.json()["access_token"]

    header = {'Content-Type': 'application/json',
          'Authorization': 'Bearer ' + mltoken}

    # NOTE: manually define and pass the array(s) of values to be scored in the next line
    payload_scoring = {"input_data": [{"fields": ['homepage_featured', 'emailer_for_promotion',
    'op_area', 'cuisine',
                                      'city_code', 'region_code', 'category'], "values": [
     input_features]}]}

    response_scoring = requests.post('https://jp-
    tok.ml.cloud.ibm.com/ml/v4/deployments/8c4cb961-7490-4977-8763-
    65929bc9bfb7/predictions?version=2022-11-17', json=payload_scoring,
```

```
                                headers={'Authorization': 'Bearer ' + mltoken})
    print("Scoring response")
    #res_scr=response_scoring.json()
    pred_res = response_scoring.json()['predictions'][0]['values'][0][0]
    prediction=round(pred_res)

    return render_template('home.html', prediction_text = prediction)

if __name__ == "__main__":
    app.run(debug=True)
```

OUTPUT SCREENSHOTS:

DemandEst - AI powered Food Demand Forecaster

know your food supply for 10 weeks?

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 dem and. Too much inventory in the warehouse means more risk of wastage, and not enough could lea d to out-of-stocks - and push customers to seek solutions from your competitors. The replenishment of 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.

start predicting

Results

waiting for prediction...

Get your number of food orders?

homepage_featured

----

emailer_for_promotion

Enter your op_area

like op_area=27

Enter your region code

Enter region_code

Enter your city code

Enter city_code

select the food cuisine

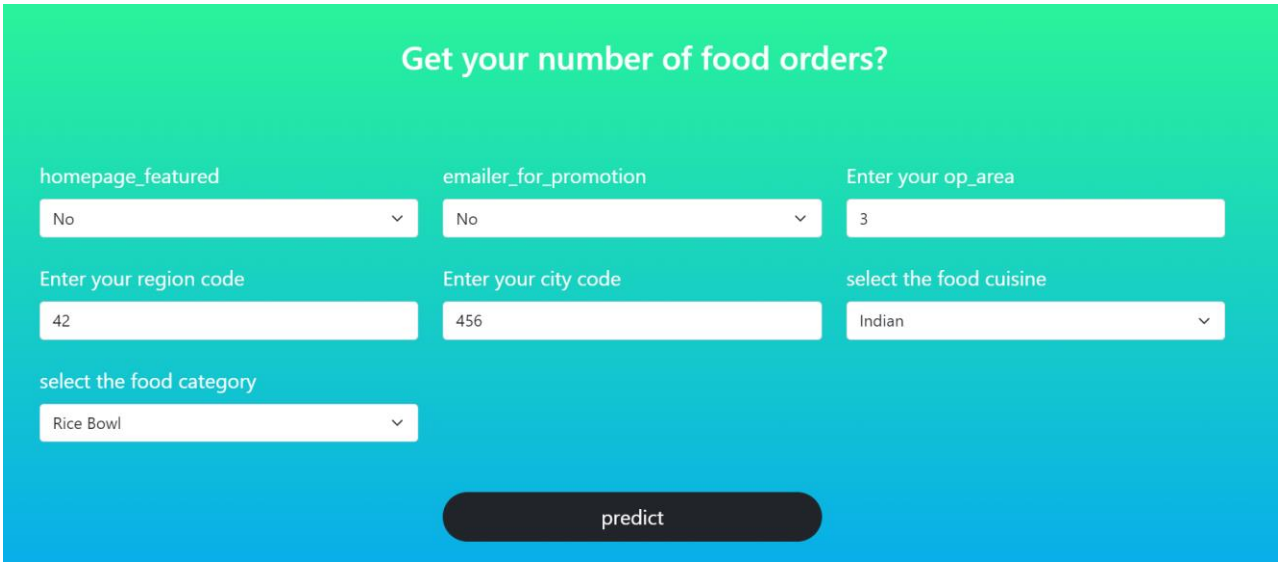
----

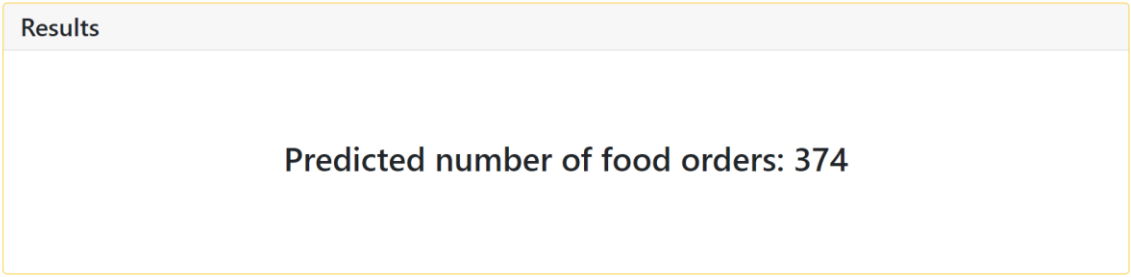
select the food category

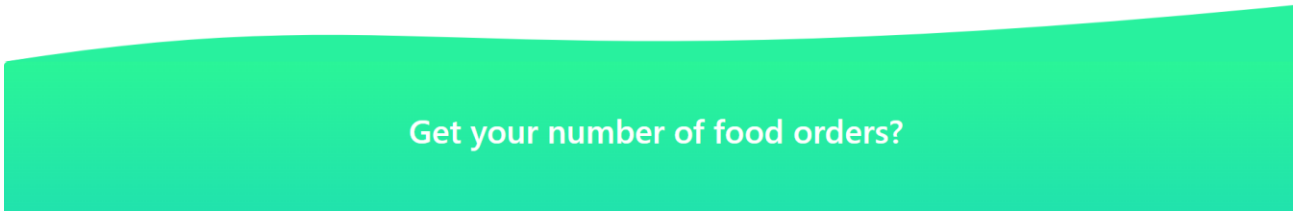
----

predict

Demo:







GITHUB LINK:

https://github.com/IBM-EPBL/IBM-Project-2589-1658475994

PROJECT DEMO LINK:

https://youtu.be/uVlSKh80vBs