IBM Hack Challenge 2020

TEAM NAME: DEXTER

TEAM MEMBERS:

1. Neelam Somai (Group Leader)
2. Gayatri Patil
3. Yash Mate
4. Gaurav Tirodkar

TITLE : Optimized Warehouse Management of Perishable Goods for a Food Delivery Company

PROBLEM DESCRIPTION

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 majority of raw materials is done on weekly basis and since the raw material is perishable, the procurement planning is of utmost importance.

OBJECTIVES

- Forecasting demand for every product in every warehouse
- Robust Database Architecture
- Gradient Boosting Machine Learning
- Interactive Visualization
- An integrated virtual assistant
- Sentiment Analysis
- Tone Analyzer to predict the sentiment of users review.

LITERATURE SURVEY

1. Snappi Warehouse:
    a. Track goods shipped in and out
    b. Create supply records
2. LoMag Warehouse Management:
    a. Import Data from excel
    b. Create & restore backup copy
3. Inventory Now:
    a. Get a snapshot of your inventory

TECH STACK

Programming Languages & Frameworks:

1. Python 3
2. IBM Watson Studio
3. IBM Cloud for Deployment
4. IBM Node-Red
5. IBM DB2
6. IBM Tone Analysis

ML Libraries for Prediction:

- Tensorflow
- Keras
- Sklearn
- Numpy
- Pandas
- Scipy
- xgboost

ML PREDICTION USING CATBOOST

```
0:      learn: 1.1849327        test: 1.1849340 best: 1.1849340 (0)     total: 674ms    remaining: 22m 28s
100:    learn: 0.5388659        test: 0.5401141 best: 0.5401141 (100)   total: 1m 13s   remaining: 22m 53s
200:    learn: 0.4974519        test: 0.4995066 best: 0.4995066 (200)   total: 2m 24s   remaining: 21m 37s
300:    learn: 0.4870540        test: 0.4893915 best: 0.4893915 (300)   total: 3m 35s   remaining: 20m 15s
400:    learn: 0.4803929        test: 0.4829653 best: 0.4829653 (400)   total: 4m 44s   remaining: 18m 53s
500:    learn: 0.4755796        test: 0.4784782 best: 0.4784782 (500)   total: 5m 52s   remaining: 17m 35s
600:    learn: 0.4717799        test: 0.4749675 best: 0.4749675 (600)   total: 7m 6s    remaining: 16m 31s
700:    learn: 0.4685677        test: 0.4720000 best: 0.4720000 (700)   total: 8m 15s   remaining: 15m 18s
800:    learn: 0.4656376        test: 0.4693082 best: 0.4693082 (800)   total: 9m 26s   remaining: 14m 8s
900:    learn: 0.4632414        test: 0.4671955 best: 0.4671955 (900)   total: 10m 36s  remaining: 12m 56s
1000:   learn: 0.4612206        test: 0.4654540 best: 0.4654540 (1000)  total: 11m 45s  remaining: 11m 44s
1100:   learn: 0.4594095        test: 0.4638916 best: 0.4638916 (1100)  total: 12m 54s  remaining: 10m 32s
1200:   learn: 0.4576874        test: 0.4624379 best: 0.4624379 (1200)  total: 14m 3s   remaining: 9m 20s
1300:   learn: 0.4560388        test: 0.4610471 best: 0.4610471 (1300)  total: 15m 12s  remaining: 8m 10s
1400:   learn: 0.4545852        test: 0.4598578 best: 0.4598578 (1400)  total: 16m 21s  remaining: 6m 59s
1500:   learn: 0.4532131        test: 0.4587557 best: 0.4587557 (1500)  total: 17m 32s  remaining: 5m 50s
1600:   learn: 0.4518347        test: 0.4576279 best: 0.4576279 (1600)  total: 18m 41s  remaining: 4m 39s
1700:   learn: 0.4505804        test: 0.4566560 best: 0.4566560 (1700)  total: 19m 51s  remaining: 3m 29s
1800:   learn: 0.4494160        test: 0.4557873 best: 0.4557873 (1800)  total: 21m      remaining: 2m 19s
1900:   learn: 0.4481502        test: 0.4548170 best: 0.4548170 (1900)  total: 22m 10s  remaining: 1m 9s
1999:   learn: 0.4470469        test: 0.4540289 best: 0.4540289 (1999)  total: 23m 19s  remaining: 0us

bestTest = 0.4540288931
bestIteration = 1999

err:  0.45402889813574215
```

ML Prediction: Predicted Output

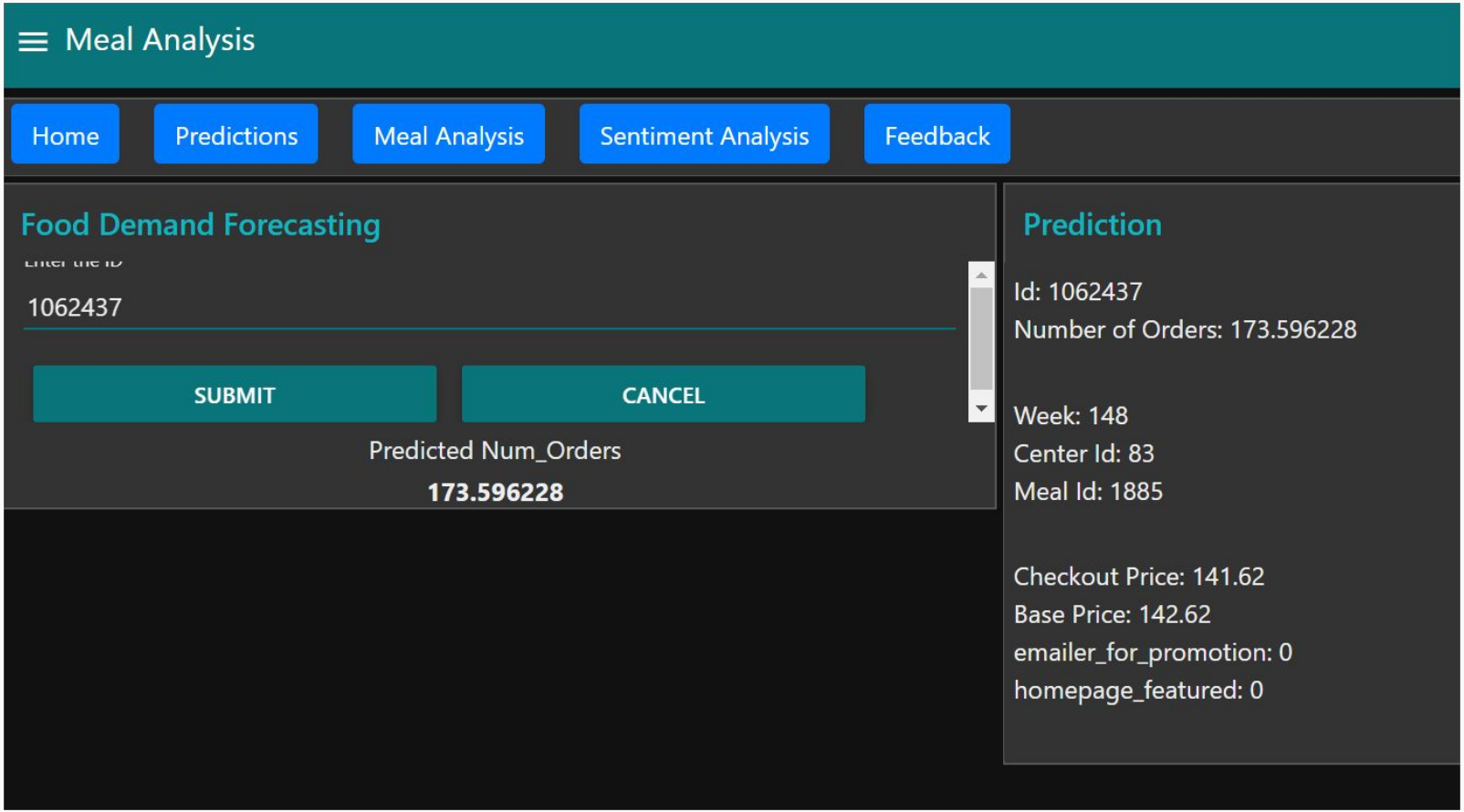
```
[ ] pred = model.predict(test_data[features])
    pred = (np.exp(pred) - 1)
    submission = pd.DataFrame({'id':test['id'], 'week':test['week'], 'center_id':test['center_id'], 'meal_id':test['meal_id'], 'checkout_price':test['che
    submission = submission[['id','week', 'center_id', 'meal_id', 'checkout_price', 'base_price', 'emailer_for_promotion', 'homepage_featured', 'num_orde
    submission.head()
```

	id	week	center_id	meal_id	checkout_price	base_price	emailer_for_promotion	homepage_featured	num_orders
0	1028232	146	55	1885	158.11	159.11	0	0	1043.790425
1	1262649	147	55	1885	159.11	159.11	0	0	1064.439018
2	1453211	149	55	1885	157.14	158.14	0	0	1040.641858
3	1262599	150	55	1885	159.14	157.14	0	0	896.750876
4	1495848	151	55	1885	160.11	159.11	0	0	1052.958971

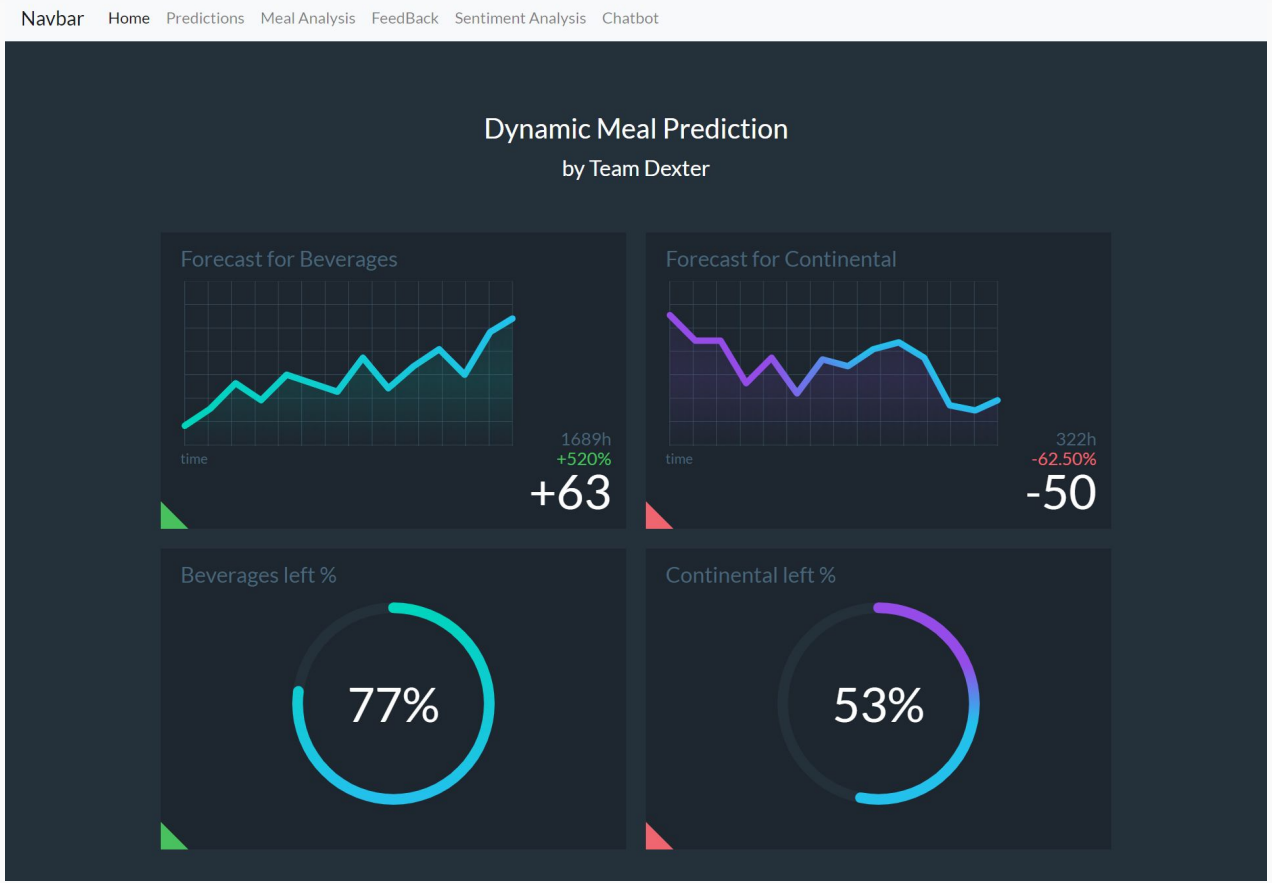
```
[ ] submission.to_csv('catboost_1.csv', index=False)
```

HOME PAGE - Dynamic Chart

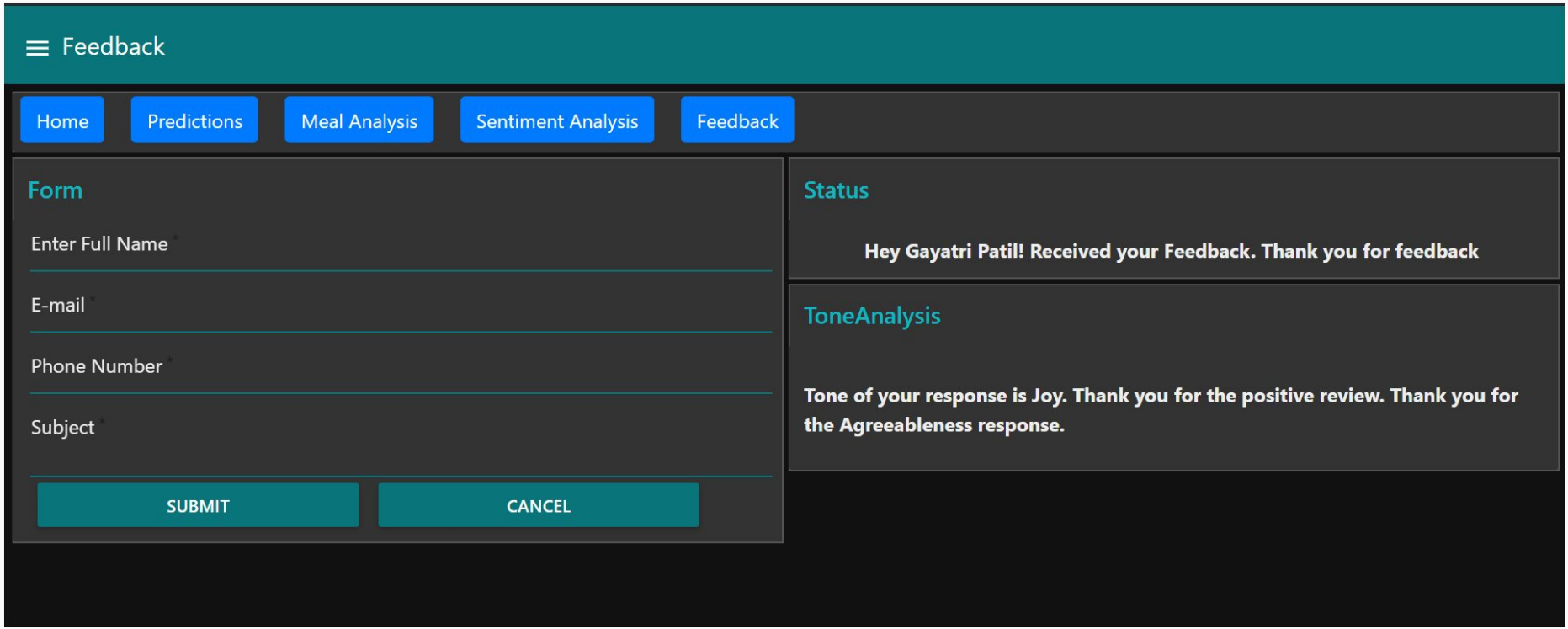
UI - Prediction Visualization



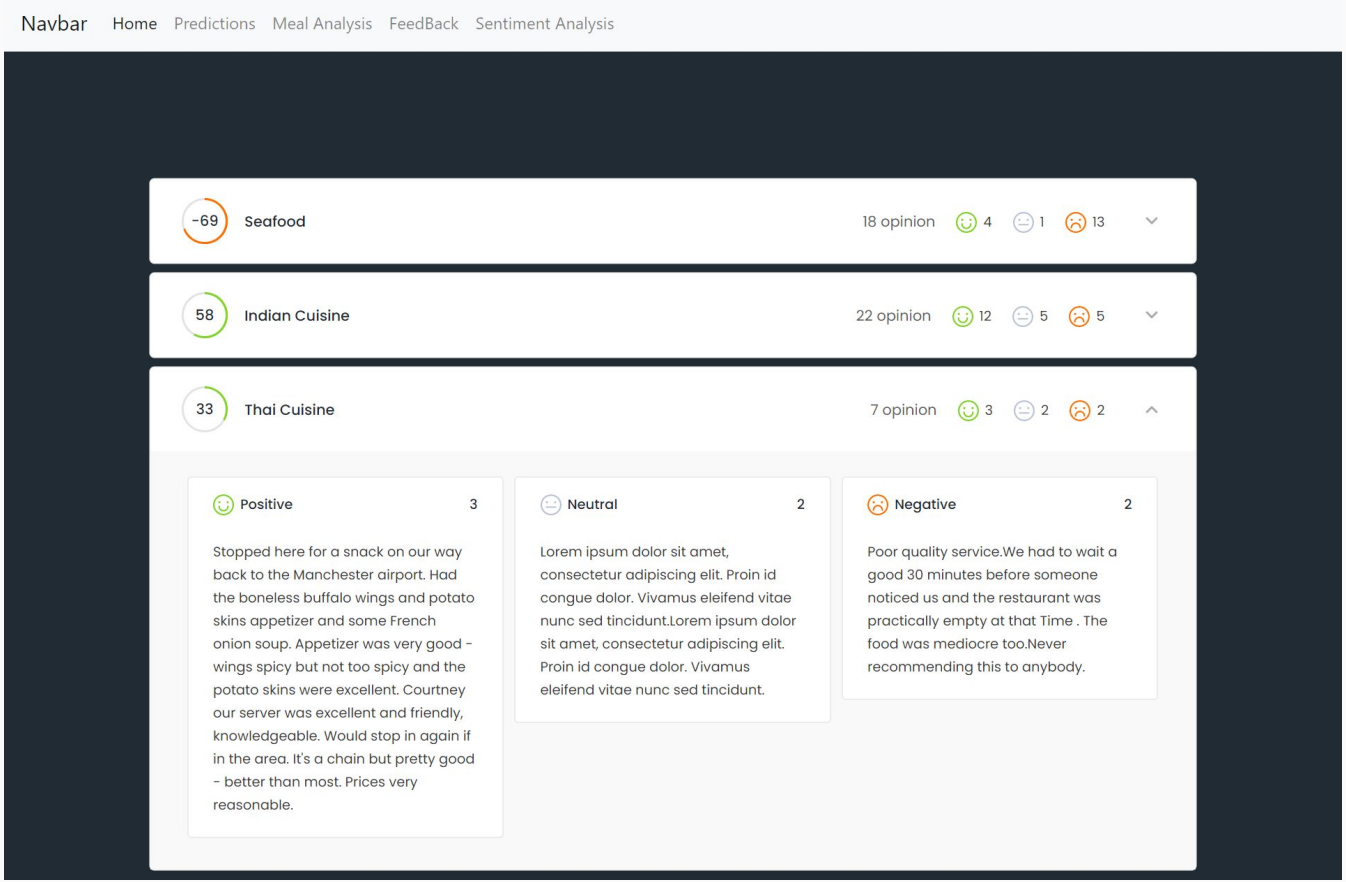
UI - Meal Analysis



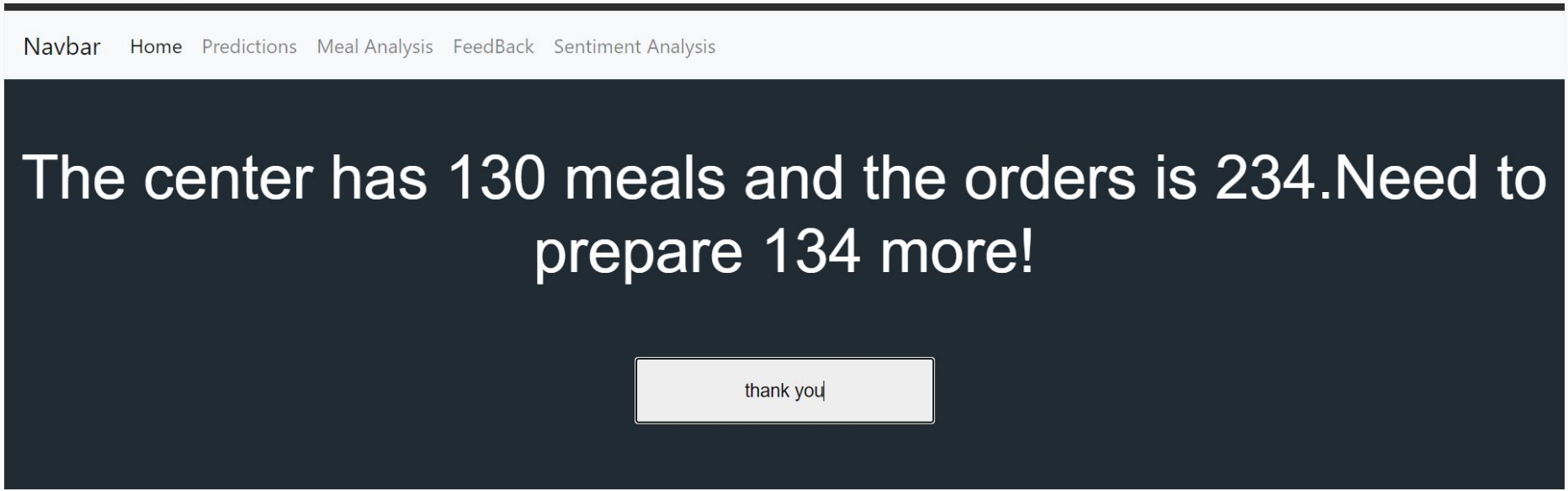
UI- Feedback Form and Tone Analysis



UI - Sentiment Analysis



UI - ChatBot


Navbar
Home
Predictions
Meal Analysis
FeedBack
Sentiment Analysis
The center has 130 meals and the orders is 234.Need to prepare 134 more!
thank you

BUSINESS IMPACT

BEFORE

1. Items used to perish in the warehouse only.
2. Shortage of food items due to high demand.
3. Due to low accuracy of existing systems, inadequate or insufficient orders were placed.
4. Monitoring of resources was manhandled leading to low efficiency.
5. No help of statistics to analyze patterns.

AFTER

1. Accurate prediction with ML assistance on requirements.
2. Data visualisation graphs for warehouse owners
3. analyze and make smart business decisions.
4. Virtual Assistance
5. Tone Analysis of Feedback

HURDLES & EXPERIENCE

HURDLES

1. Could not find IBM ML support for Catboost Model
2. Lack of online resources for using csv node in Node-Red
3. Integrating ui builder node with Node-Red dashboard

EXPERIENCE

1. Hands on experience with cloud services
2. Excellent courses offered
3. Project management and implementation systematically

CONTRIBUTION

1. Neelam Somai:
    a. Research and UI design
    b. Documentation
2. Gayatri Patil:
    a. UI and Integration
    b. Deployment
3. Gaurav Tirodkar:
    a. ML Prediction
    b. Dashboard
4. Yash Mate:
    a. ML Model
    b. Functionalities

THANK YOU !