Demand forecasting for energy consumption

DASS Project : Team 12 Report

SPRING 2020, IIIT Hyderabad

Client

Indriyn Data Analytics
CIE, IIIT Hyderabad
POC - Mr. Malliswar Tekumudi

Development team

Akshat Goyal
Kanish Anand
Nikunj Nawal
Sridhar M

Index

1. Introduction and background
2. User profile and Usage model
3. Technologies used
4. Use Case
5. Use Case diagram
6. Design overview
7. UML state diagram (for single use case)
8. Approach towards project
9. Challenges faced
10. Learning
11. Final product

Introduction & Background

Given hourly energy consumption data of a client's commercial complex, we need to make a web-app where users can input a time range and get predicted energy consumption value and predicted electricity cost for that duration of time which can be used to manage the monthly budget of the company. Our goal with product is to:

1. **Predict Energy Consumption** - The project will help in predicting the value of the energy consumption of a complex using a machine learning model trained with initial given data and features which affects the value of the energy prediction like holiday, weather.The values of temperature was obtained from API. The energy prediction is done on a daily as well as hourly basis and can be obtained by giving proper inputs in the Webapp.

2. **Graphical Analysis** Of The Predicted Data - Different graphs are plotted using the predictions made by the model and these graphs can be downloaded by the user.User will help them to analyse different things like the energy consumption and budget planning of the complex and help them to detect if there is technical fault in the electric connections providing the electricity to the complex etc.

3. **Functionalities varying as per user type** - There are three different type of users of the project The Owner, The Finance Team and The Maintenance Team of the company and different functionalities required by all the types of users after discussion to the client and tried to fulfill everyone's demand like adding budget prediction and it's graphical analysis for the financial team etc.

4. **View Weather** - The users will be able to see the weather forecast whose data will be fetched from an API used. The API used is finalized by the client, a functional UI frame should be available. The values of the weather forecast will be fetched from this API and displayed on the Dashboard available on home screen.

5. **Notification and Feedback Form** - If there's any kind of anomaly in the data a notification will be sent to the maintenance team so they can handle and update the

data and the user can give their feedback using the feedback form regarding the Webapp and any type of user can fill the feedback form.

The objective of this Request for Proposal is to locate a source that will provide the best overall value to Indriyn Data Analytics . While price is a significant factor, other criteria will form the basis of our award decision, as more fully described in the Evaluation Factors section of this Request for Proposal below.

User Profile and Usage model

The different users (wrt a particular complex) who would be using the software include :

1. Owner - The Owner will have full access to the product, will be familiar with using the software and can monitor the energy consumption rate, generate reports and act accordingly.
2. Finance team - The Finance team will use future energy consumption to plan expenditure and predict the cost of electricity.
3. Maintenance team - The Maintenance team will ensure the maintenance of the software with the changing system environment and will provide updates.

Like any other web application, once our system is deployed, users will need only a system,like a laptop or mobile, capable of using the internet.

Technologies used

Following are categorized technologies used for the system:

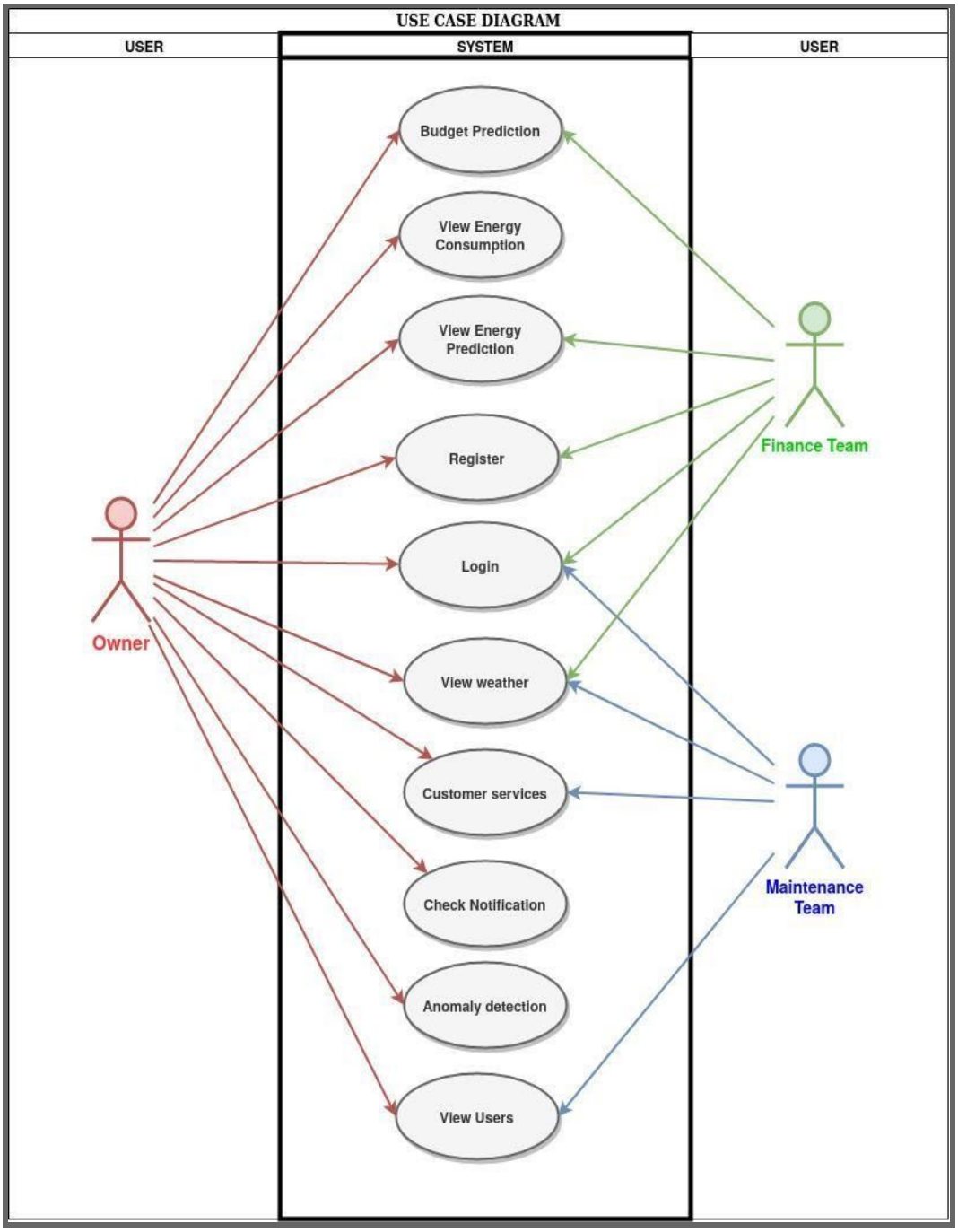
1. Development environment
    a. Vim and VS code - edit program files
    b. Jupyter notebook - for ML
    c. Facebook's Prophet model
    d. MongoDB Compass
2. Programming tools
    a. HTML/CSS
    b. Javascript
    c. React JS
    d. Node JS
    e. Express JS
    f. Python3
    g. Mongo DB
3. Collaboration tools
    a. Gitlab
    b. Whatsapp
    c. Outlook mail
    d. Microsoft Teams
4. Creating project documents
    a. Google docs
    b. Google sheets
    c. Google slides
    d. Google drive

Use Case

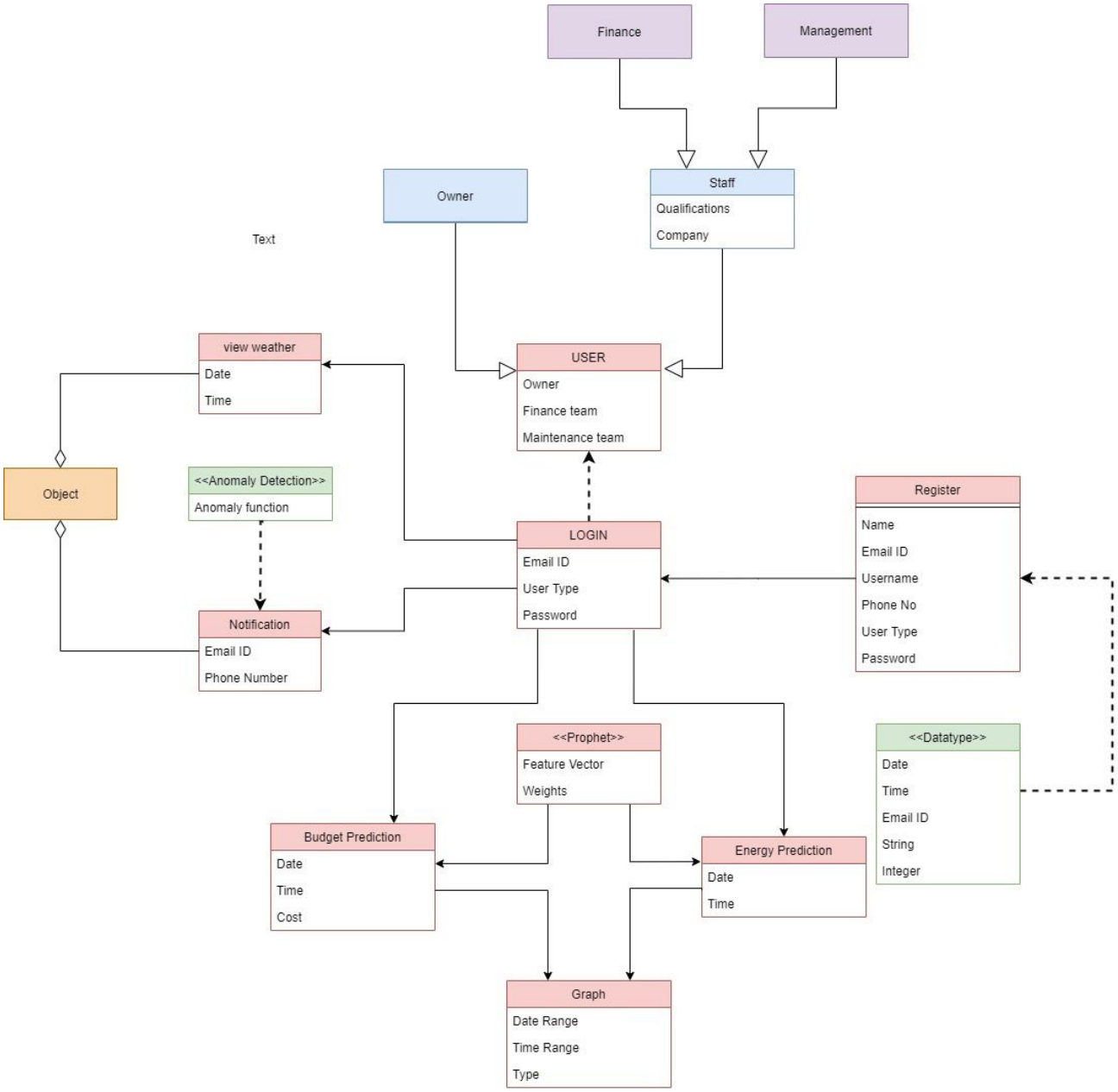
No.	Use Case Name	Description
1.	View predicted values for energy consumption on Jupyter notebook	Users can get the prediction of hourly and daily energy consumption by using the Jupyter Notebook. Functional Web-app not included
2.	View energy consumption graph of data provided on Jupyter notebook	Users can get a graphical analysis of energy consumption.The output should correspond to the given data
3.	Login and Register for different roles for the system	Different type of users can firstly register and then log in to access pages customized to their needs
4.	View predicted value for energy consumption	Users can get the prediction of hourly and daily energy consumption in the functional Web-app.
5.	Get help regarding the web-app	User will be able to contact the maintainer of the Web-app for any kind of help using automated mail
6.	Give user feedback via the web-app	User will be able to give feedback using the feedback form and mail to the help person
7.	View graphical analysis of energy consumption	Users can get a graphical analysis of energy consumption using Web-app.
	Download the generated energy consumption prediction graphs	Users will be able to download the generated graph of energy consumption predictions.
9.	Download the given energy consumption data in form of graphs	Users will be able to download the generated graph for energy consumption data.
10.	Download the generated budget prediction graphs	Users will be able to download the generated graph of budget predictions.
11.	View budget Prediction	Users will be able to get a predicted value of budget after giving desired inputs.
12.	View notification for an anomaly	In case of some forged anomalous data a notification will be given to the user
13.	View actual data of energy consumption	User will be able to check the actual data of energy consumptions

14.	View current energy consumption	User will be able to check live energy consumption data which will be updating with time
15.	Check the total consumption of energy	User will be able to view the total energy consumption
16.	Update profile of the logged-in user	User can update their profiles after log in
17.	Update the email for receiving the anomalous data updates	User can update the email to receive the emails about the anomalous data and other notifications.
18.	View weather forecast	User will be able to view weather related forecast(e.g. temperature)
19.	View anomaly cases as a list	A list for all the anomalies will be available to the user
20.	Select API for weather forecast	Selecting API will give value corresponding to their site

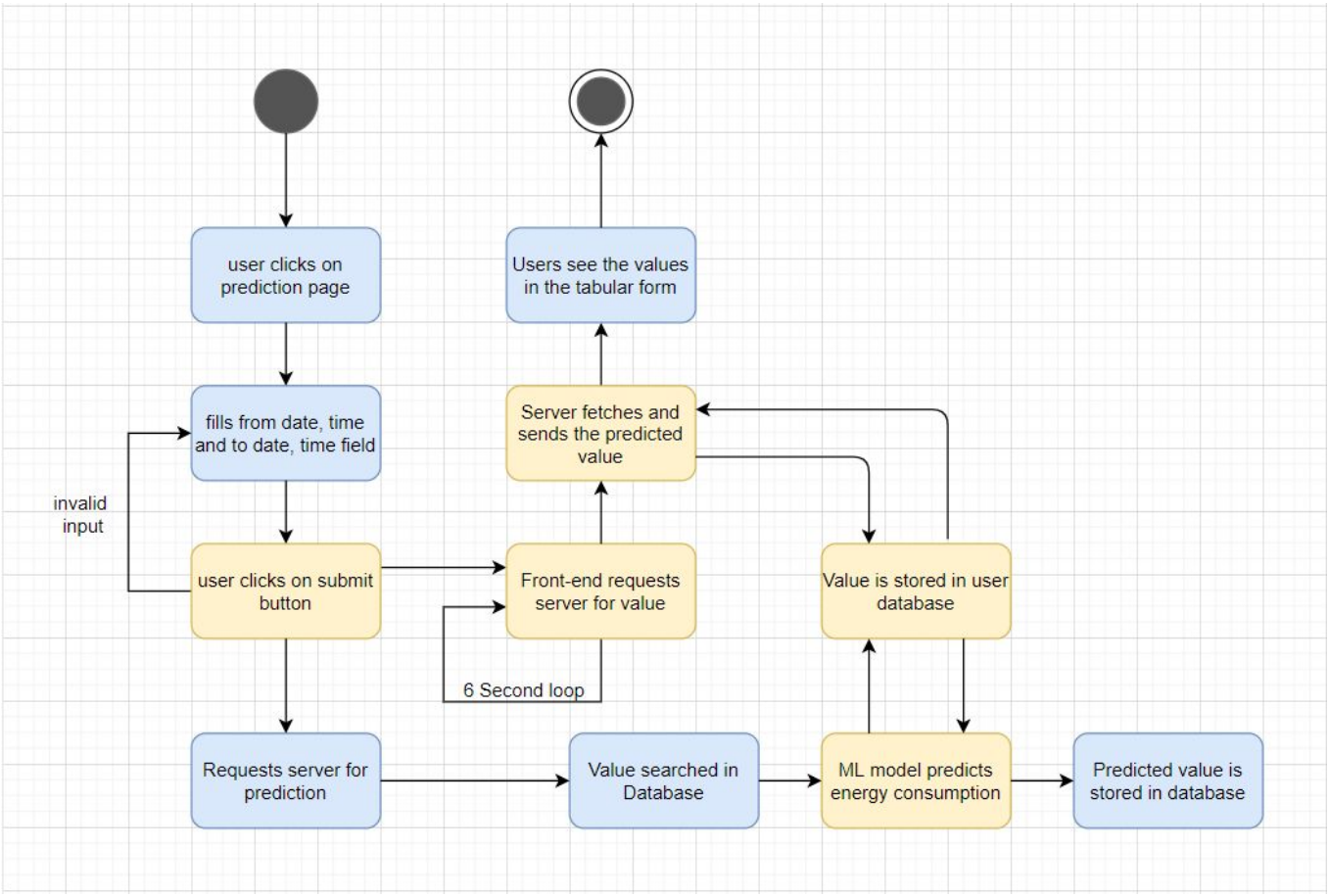
Use Case diagram (overview)


USE CASE DIAGRAM
USER
SYSTEM
USER
Budget Prediction
View Energy Consumption
View Energy Prediction
Register
Login
View weather
Customer services
Check Notification
Anomaly detection
View Users
Owner
Finance Team
Maintenance Team

Design overview



State Diagram



Approach

We decided to take one of the agile process frameworks called scrum methodology.
In this methodology processes were inspired by empirical *inspect and adapt* feedback loops to cope with complexity and risk. We emphasized decision making from real-world results rather than speculation. Also, the time was divided into short work cadences, known as sprints, typically one week or two weeks long. The product was kept in a potentially shippable (properly integrated and tested) state at all times. At the end of each sprint, stakeholders and team members met to see a demonstrated product increment and plan its next steps.

In this method we had a simple set of roles, responsibilities, and meetings that never changed. By removing unnecessary unpredictability, we were better able to cope with the necessary unpredictability of continuous discovery and learning.

Challenges

Initially the project was going smooth but as the time went on we found drawbacks in our planning strategies. This project being our first team project came with varieties of challenges along with the project itself. We were new to project planning, team work distribution, adapting the SDLC, etc.
Regarding the project:

- Our design and SRS changed a lot corresponding to client's demands
- Data was not available for free and paid ones were very expensive. So for that we have to struggle a lot to get the historical temperature data for training of the model.
- Since the client travelled a lot we had fewer chances for face to face meetings
- Our project was to predict future energy consumption models through a machine learning model. So prediction through a web app was taking a lot of time. In order to make it more efficient we need to cache the data for the fast results.
- Also we were not able to get exact temperature values due to some constraints as the API to get historical temperature data was costly.

Learning

The project gave us real-world experience in software development with a client. We had experienced a wide range of situations that a team would face in the software development process including changing requirements, difficulties in meeting deadlines, clashes in deadlines, changing the project plan as and when required to accommodate deadlines and a lot more.

1. Exploring MERN stack alot. Before this project we were almost unaware about the integration process of frontend with backend , we only knew about making a simple frontend page. But along with this project we got to learn much about integration with using MERN stack.
2. We got to explore various libraries in react eg : fusion-charts, mail-provider etc.
3. We learned alot about Machine Learning as our project was based upon mainly developing a machine learning algorithm for given conditions. For this purpose we need to study Machine Learning a lot so as to clear the basic concepts of Machine Learning so that we could choose the best algorithm and select best features for the given problem.
4. Apart from all these technical stuff we got exposure about how to perform such projects in the real world because till now we were only doing small assignments and this project was first of such a large project for us all. So we got to learn about how to work with client like :
    a. Managing meetings with clients and TA's.
    b. Regularly updating them with our current progress.
    c. Completing daily and weekly goals.
    d. How to negotiate with a client.
5. This was our first large project with a team so it gave us all a great experience about how to collaborate on a project in a team :
    a. Distributing work among us all.
    b. Conducting Regular Team meetings.
    c. Working remotely with a team in COVID19 conditions.
    d. Mutually helping each other.
    e. Also this was our first project requiring such proper documentation so we got to learn about why these documents are really very useful in long term projects.

6. We were bound to only do the things mentioned in the SRS according to the requirements of the client . So due to this we were not able to apply more features which we feel could enhance the project much more.

Final Product

The link for the final product demo video is here.
So our final product is a webapp made using MERN stack which has various functionalities :

- Register & Login with various user types.
- A Dashboard showing statistics of energy consumption and expenditure values of the complex. The Dashboard showed the values of previous month's energy consumption and total expenditure. Also it showed the graphs and pie charts expressing the month wise consumption and expenditure data of last year.
- To get predicted energy consumption values in the form of a table through that table we could also download the data in the form of .csv file which could be used later.
- To get proper graphical analysis of predicted energy consumption data for a certain time range selected. Graphs could also be downloaded in the form of an image or the data of the graph could be saved in the form of a csv file.
- To see a 5-day weather forecast of the place of office.
- To get notifications daily about the full statistics of energy consumption for the previous day.
- Feature to mail maintainer's regarding some problems.
- For the owner there was an option to see all the users of the webapp.
- Feature to edit your profile.