PROJECT REPORT

### **Team id:- NM2023TMID11742**

### TOPIC :- A Reliable Energy Consumption Analysis System for Energy-Efficient Appliances.

**Table of Contents**

1. Introduction
2. Overview of the Complete Project
3. Tools and Technologies used
4. Problem Statement & Architecture
5. Data Collection & Prepration
6. Perform the Analysis of your Project
7. Machine Learning Models in Project
8. Prediction & Future Enhancements
9. Outro & Conclusion

Completed By :-

Team Lead :- RANA PRATAP RAO

Team member 1 :- PRATIK KUMAR JHA

Team member 2 :- VISHAL KUMAR

Team member 3 :- VIVEK KUMAR

Introduction

The Energy Consumption Analysis System is a project aimed for analyzing and predicting energy consumption for energy-efficient appliances in households. This system utilizes Machine Learning(ML) algorithms to analyze the historical energy consumption data and make accurate predictions for future usage. The goal of the project is to assist house owners or utility companies to manage their energy usage, reducing waste, and lowering costs.

This project is useful for the house owners or utility companies to manage their electricity consumptions. Manage the record of the electric consumptions and finding the way of using less energy by maintaining data of the particular house or utility companies which they can use to increase energy-efficiency at their places.

Overview of the Complete Project

**Home Page:-** The home page provides an overview of the project and displays the project name, "Energy Consumption."

**About Page:-** The about page provides detailed information about the project, including its purpose, methodology, and potential impact on energy usage and sustainability. It also includes a visually appealing image and accompanying content.

**Contact Page:-** The contact page allows users to get in touch with the project team. It includes contact details such as location, email address, and contact number. Users can fill in their name, email, and message in the provided input fields and submit the form to send a message to the project team.

**Prediction Page:-** The prediction page enables users to make energy consumption predictions by providing input values for various parameters such as global reactive power, global intensity, sub metering 1, metering 2, and metering 3. Users can then click the "Predict" button to submit the form and receive the predicted result.

**Prediction Result Page:-** The prediction result page displays the predicted result obtained from the prediction page. Users can view the result in a visually appealing format.

TOOLS AND TECHNOLOGIES USED

**HTML:-** Used for creating the structure and layout of the web pages.

**CSS:-** Used for styling the web pages and enhancing the visual appearance.

**Flask:-** Used as the backend framework to handle the routing and logic of the web application.

**Python:-** Used for implementing the backend functionality, data processing, and prediction algorithms.

**Data Visualization:-** The system incorporates data visualization techniques to present energy consumption trends and patterns in a visually appealing and intuitive manner. Graphs, charts, and interactive visual elements provide users with a clear understanding of their energy usage.

**PROBLEM STATEMENT &ARCHITECTURE**

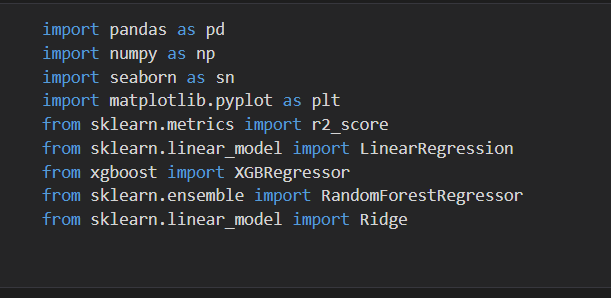
Problem Statement :- I am a house owner who is trying to monitor and optimize the energy usage of my energy-efficient appliances. But, I am facing challenges in accurately tracking individual appliance energy consumption, which makes it difficult for me to identify energy-saving opportunities and make informed decisions to reduce my energy consumption. This lack of visibility and analysis capabilities leaves me feeling frustated and unable to effectively manage and optimize my energy usage, leading to potential wastage and higher energy bills.

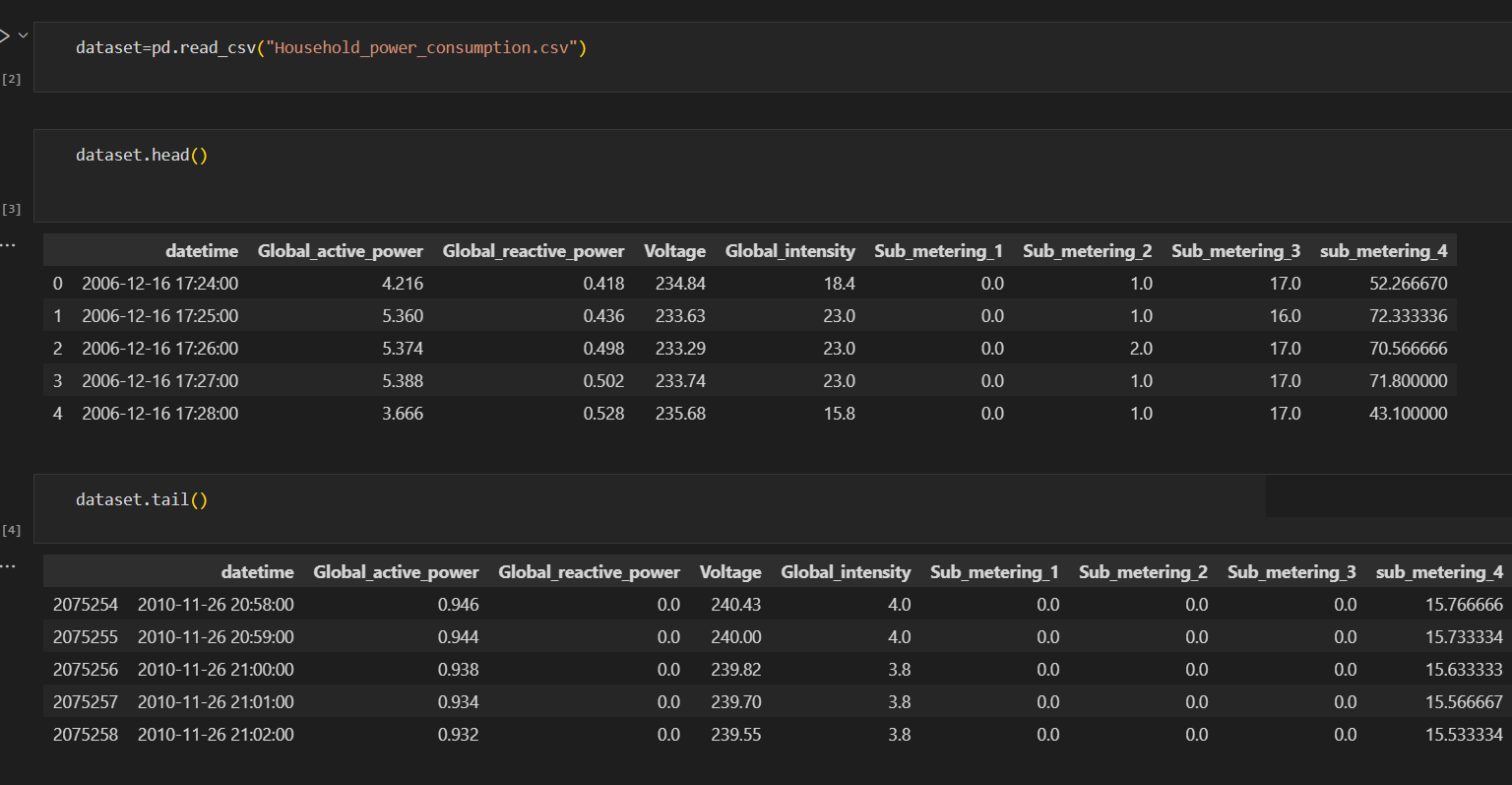
Architecture :- The Energy Consumption Analysis System follows a client-server architecture. The client-side consists of HTML, CSS, and JavaScript, while the server-side is implemented using the Flask framework and Python. The server handles the routing, data processing, and prediction functionalities, while the client-side renders the web pages and interacts with the server through HTTP requests.

Data Collection & Preparation

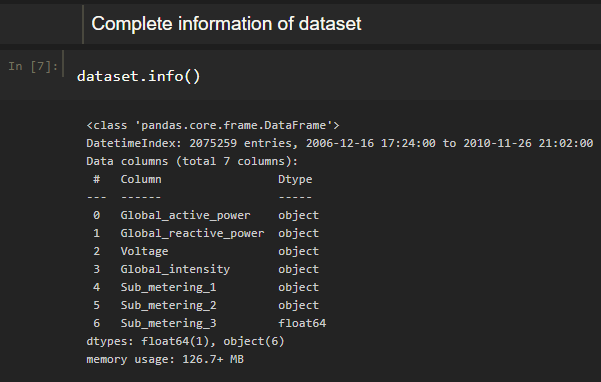
Dataset link :- [*https://www.kaggle.com/datasets/uciml/electric-power-consumption-data-set*](https://www.kaggle.com/datasets/uciml/electric-power-consumption-data-set)

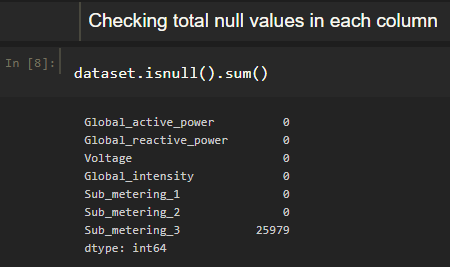
Data Collection :- Data collection is a crucial step in the Energy Consumption Analysis System. The system collects energy consumption data from households through smart meters or other monitoring devices. Additional data sources, such as weather data or occupancy information, may also be integrated into the system. This diverse dataset provides comprehensive information for accurate analysis and prediction.

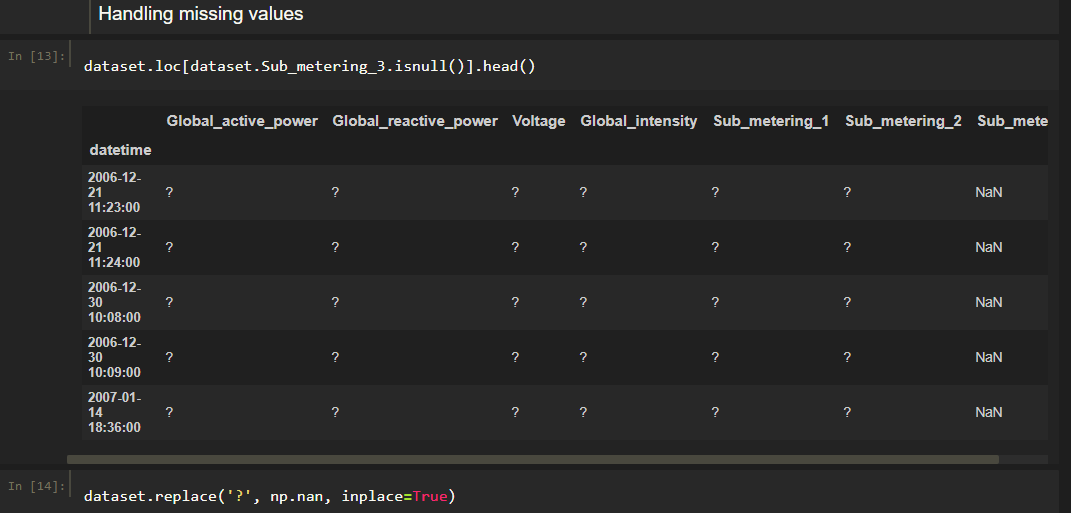


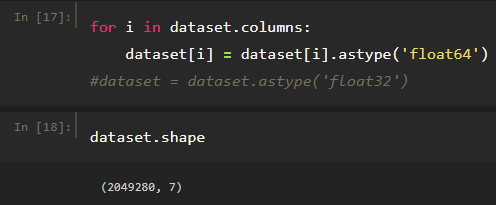


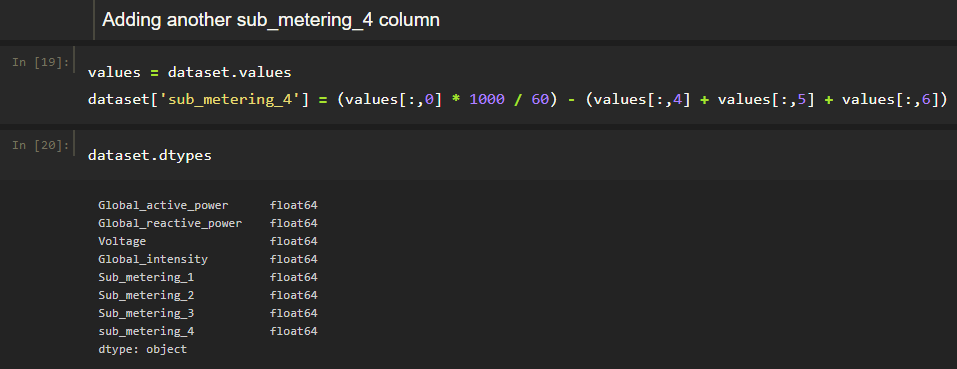
Data Processing :- Once the data is collected, it undergoes pre-processing and cleaning. This involves handling missing values, normalizing the data, and transforming it into a suitable format for machine learning algorithms. Data processing techniques ensure the quality and reliability of the input data.

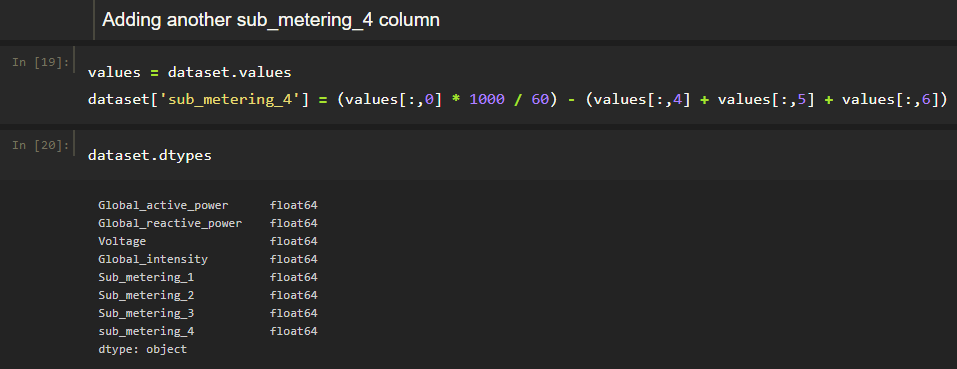






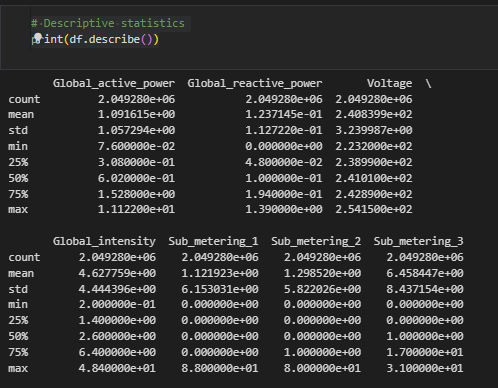






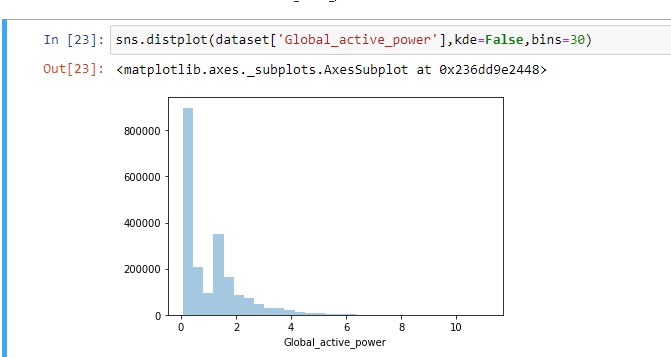
Perform the Analysis of your Project

Descriptive Statistical :-

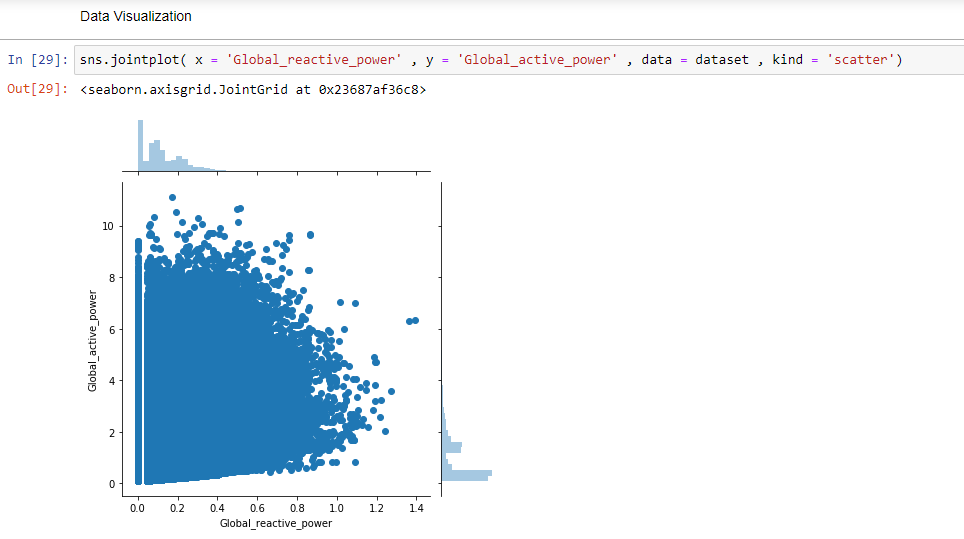


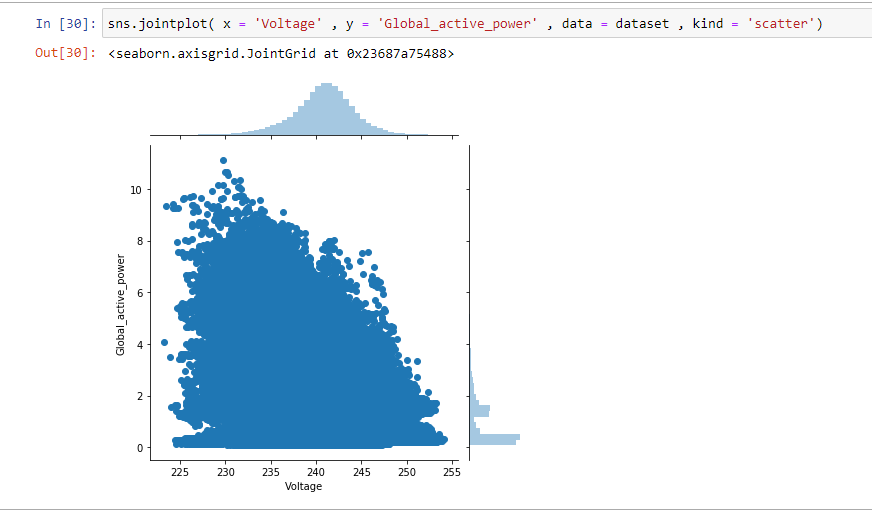
Visual Analysis :-

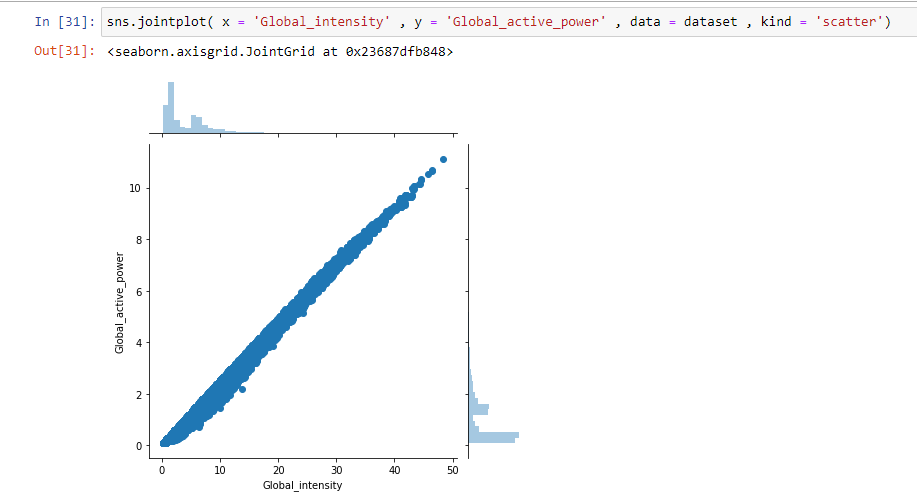
Univariate

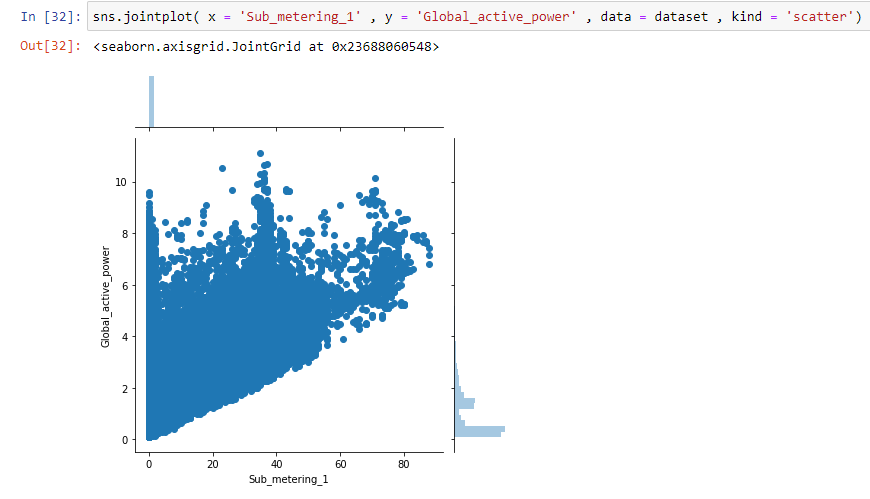


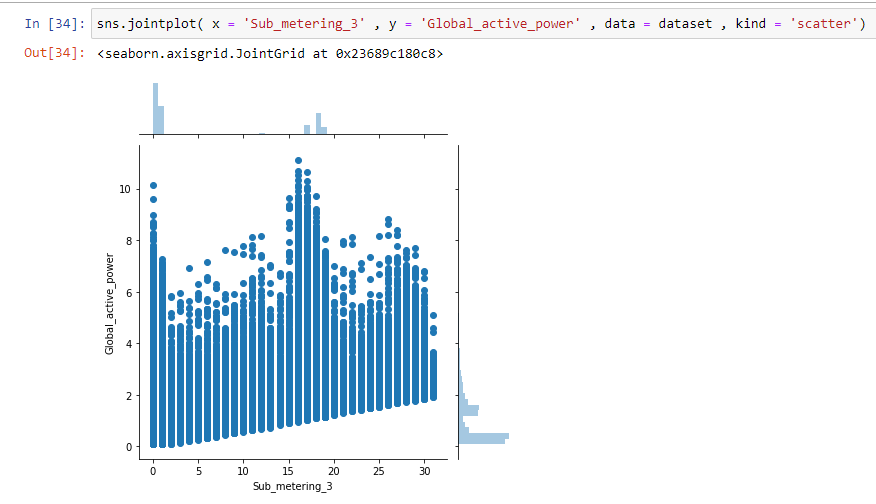
Bivariate

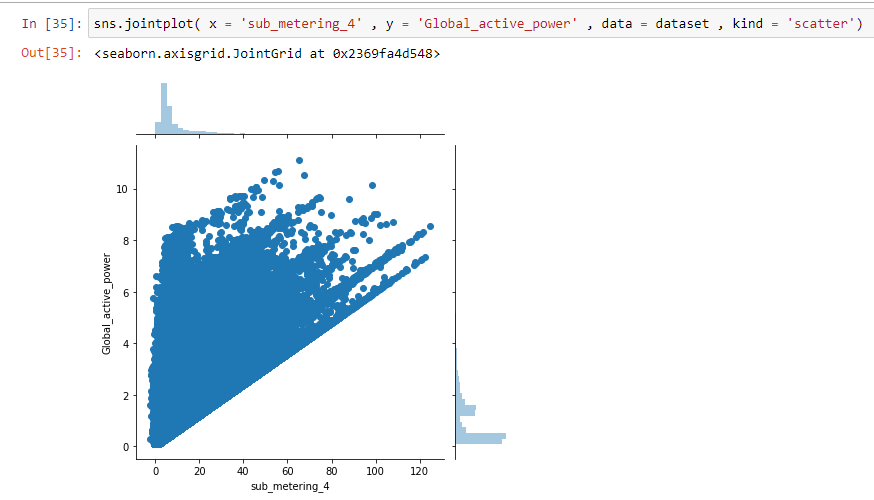




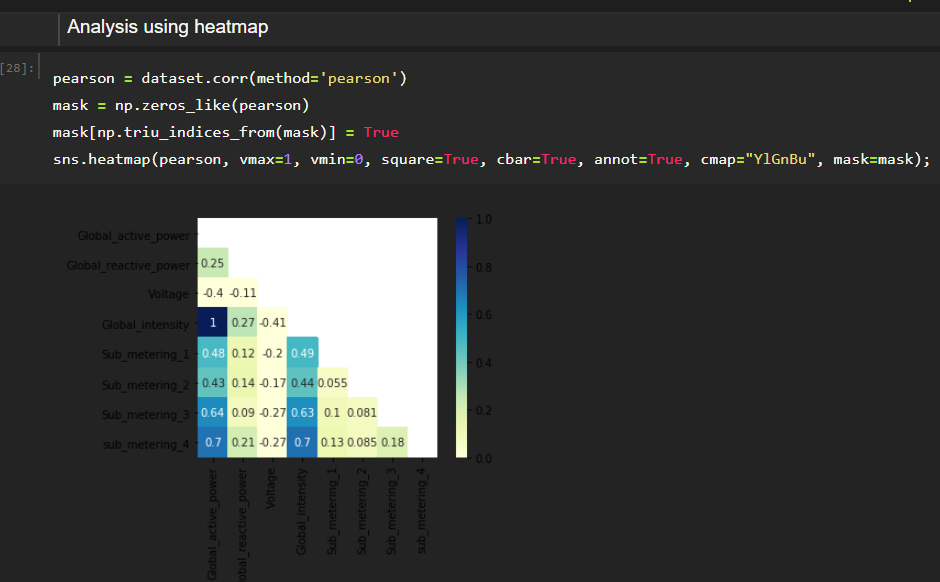




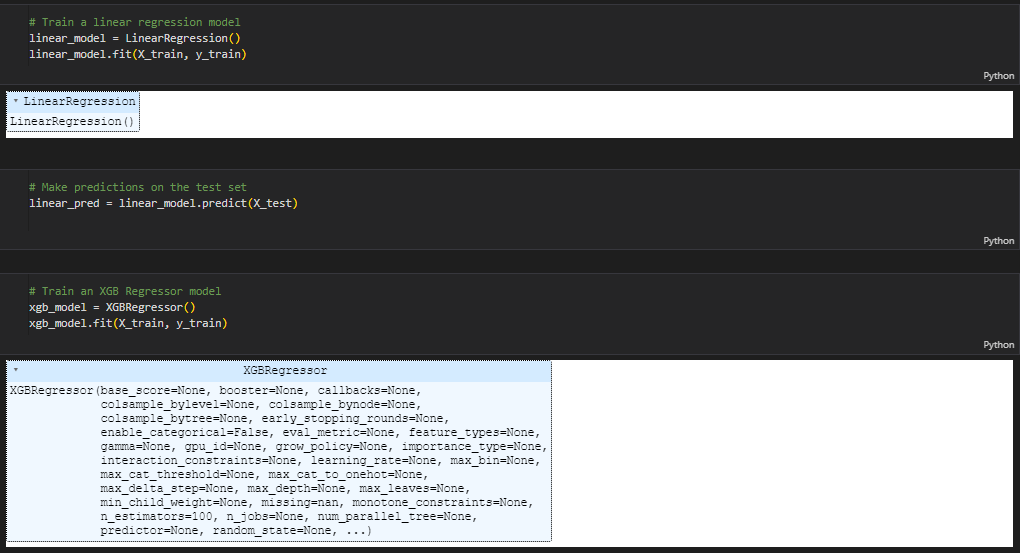


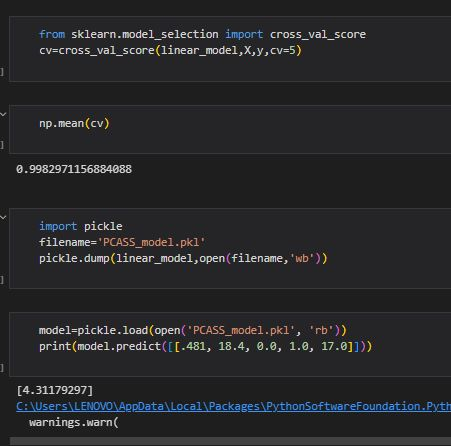


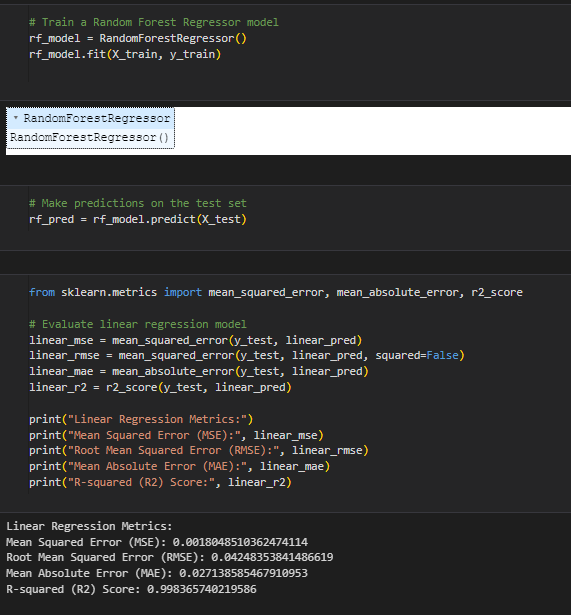
Multivariate

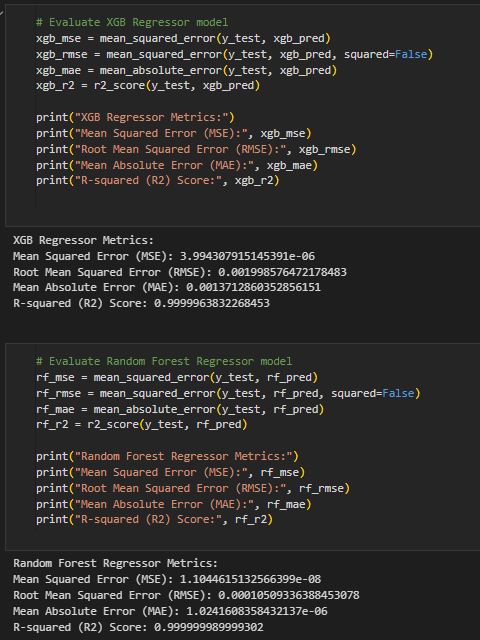


Machine Learning Models in Project









Prediction & Future Enhancements

Prediction:-

The prediction phase involves taking user input for the parameters such as global reactive power, global intensity, sub metering 1, metering 2, and metering 3. These input values are then passed through the trained machine learning models to generate predictions for future energy consumption. The predicted results are displayed to the user, providing valuable insights into their energy usage patterns and potential areas for improvement.

Future Enhancements :-

Integration with smart home systems: Enhance the system by integrating it with smart home systems to automatically collect real-time energy consumption data and control appliances based on energy-saving recommendations.

Energy efficiency recommendations: Provide personalized energy efficiency recommendations to users based on their consumption patterns, helping them optimize their energy usage and reduce waste.

Integration with utility companies: Collaborate with utility companies to incorporate real-time energy pricing and incentivize energy-efficient behaviour among users.

Mobile application: Develop a mobile application version of the Energy Consumption Analysis System, allowing users to access and monitor their energy consumption data on the go.

Advanced analytics: Expand the system's analytics capabilities by incorporating advanced techniques such as anomaly detection, clustering, or time-series forecasting to provide more in-depth insights into energy consumption patterns.

Social engagement: Implement social sharing features to encourage users to share their energy-saving achievements and engage in friendly competitions, fostering a sense of community and awareness around energy conservation.

### Integrate With Web Framework

In this section, we will be building a web application that is integrated to the model we built. A UI is provided for the uses where he has to enter the values for predictions. The enter values are given to the saved model and prediction is showcased on the UI.

This section has the following tasks

* Building HTML Pages
* Building server-side script
* Run the web application

**Build HTML Code:-**

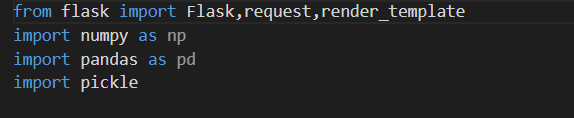
In this HTML page, we will create three pages namely

* index.html
* inspect.html
* output.html

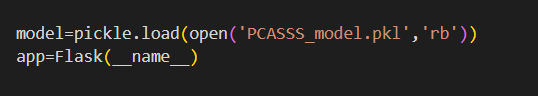
and save them in the templates folder

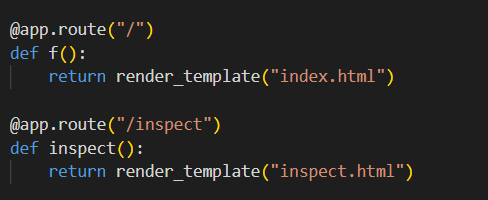
**Build Python Code:-**

Import the libraries



Load the saved model. Importing the flask module in the project is mandatory. An object of Flask class is our WSGI application. Flask constructor takes the name of the current module (\_\_name\_\_) as argument.

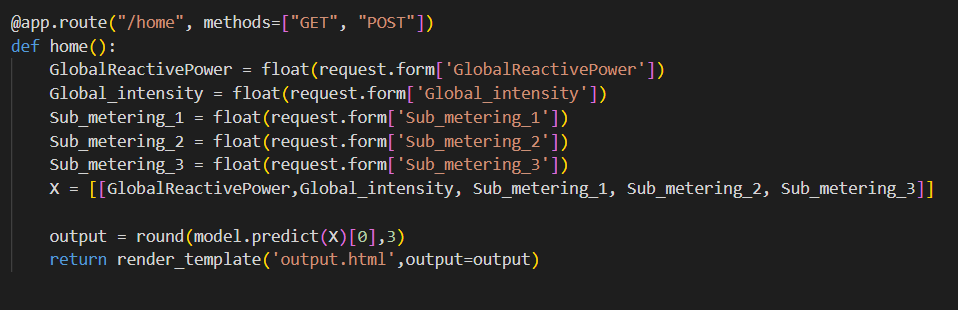




Here we will be using a declared constructor to route to the HTML page which we have created earlier.

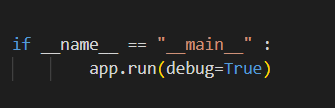
In the above example, ‘/’ URL is bound with the home.html function. Hence, when the home page of the web server is opened in the browser, the html page will be rendered. Whenever you enter the values from the html page the values can be retrieved using POST Method.

Retrieves the value from UI: -



Here we are routing our app to home() function. This function retrieves all the values from the HTML page using Post request. That is stored in an array. This array is passed to the model.predict() function. This function returns the prediction. And this prediction value will be rendered to the text that we have mentioned in the output.html page earlier.

Main Function:-



**Run the Web Application**

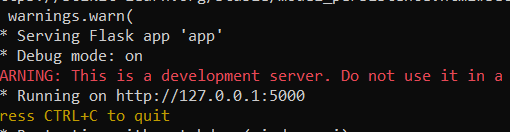
Open anaconda prompt from the start menu

Navigate to the folder where your python script is.

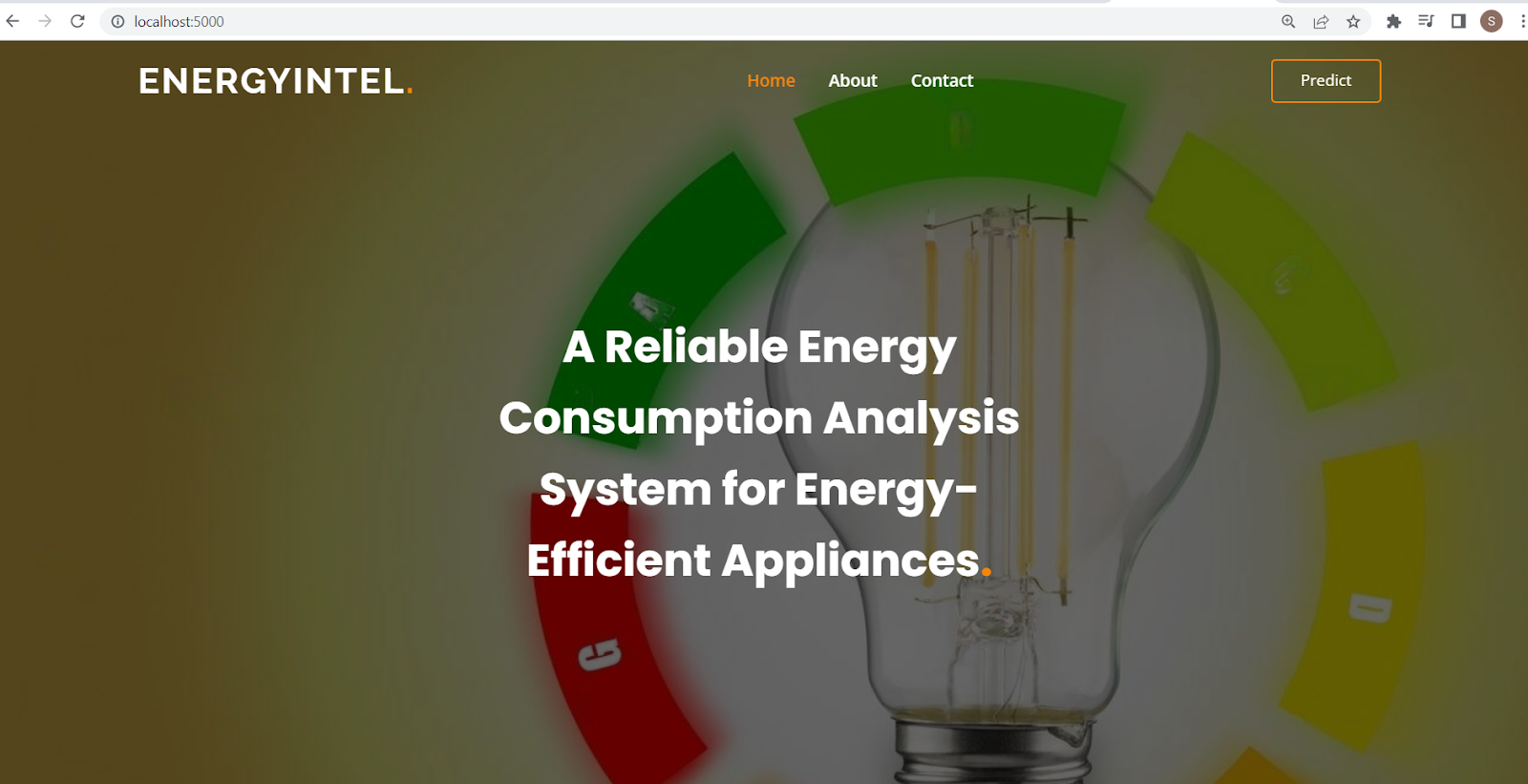
Now type “python app.py” command

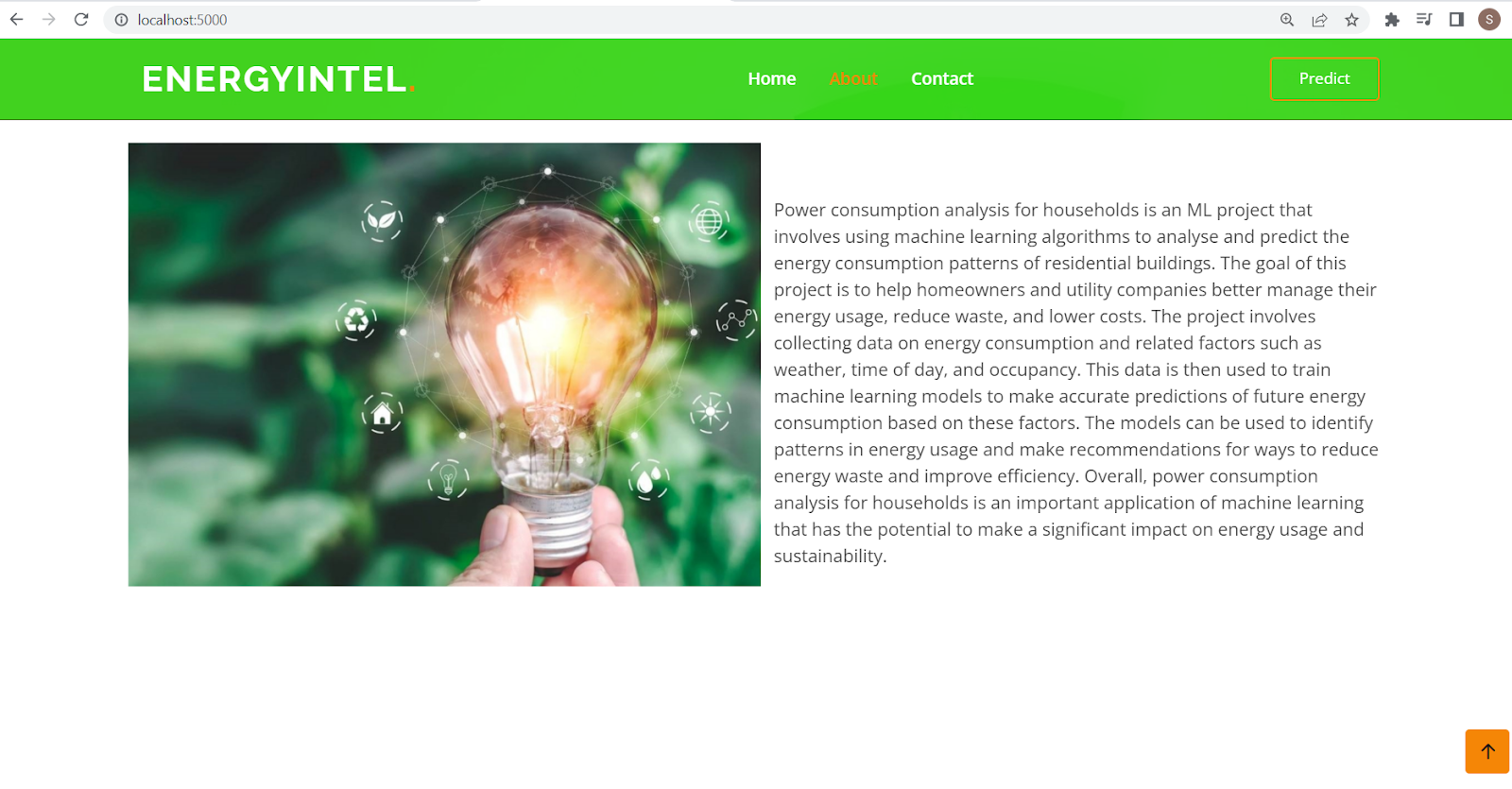
Navigate to the localhost where you can view your web page.

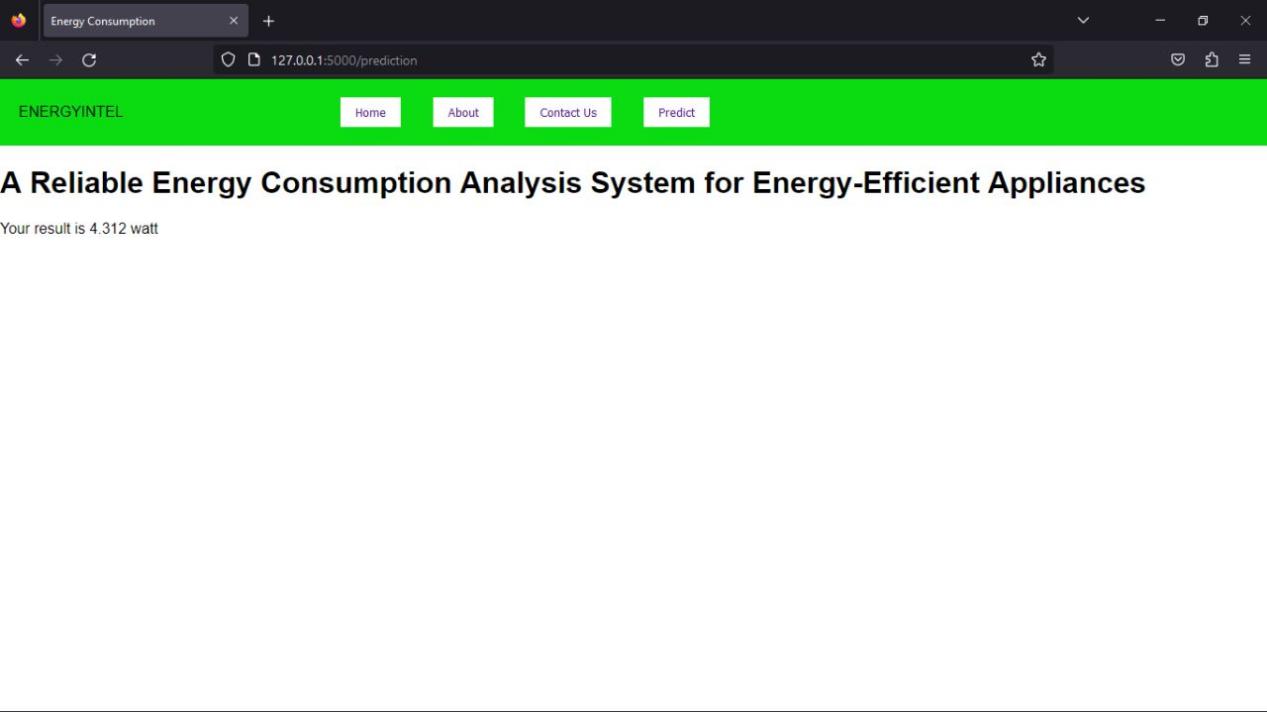
Click on the predict button from the top left corner, enter the inputs, click on the submit button, and see the result/prediction on the web.

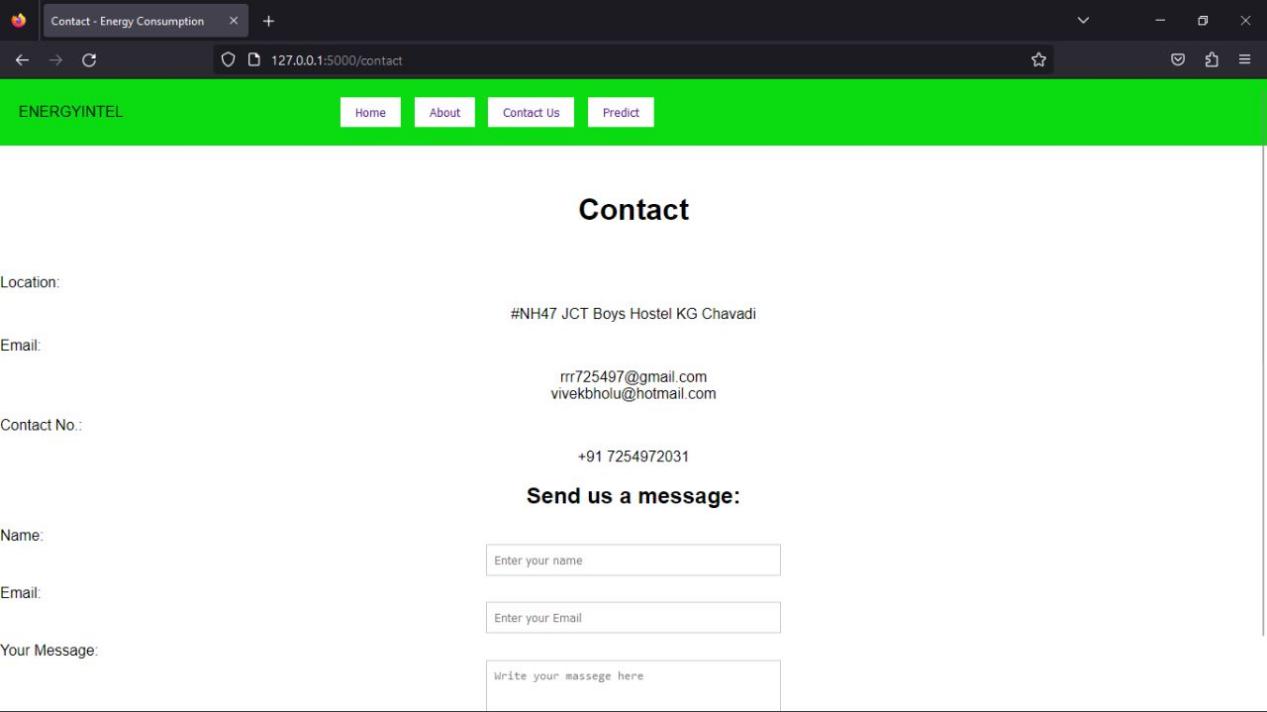


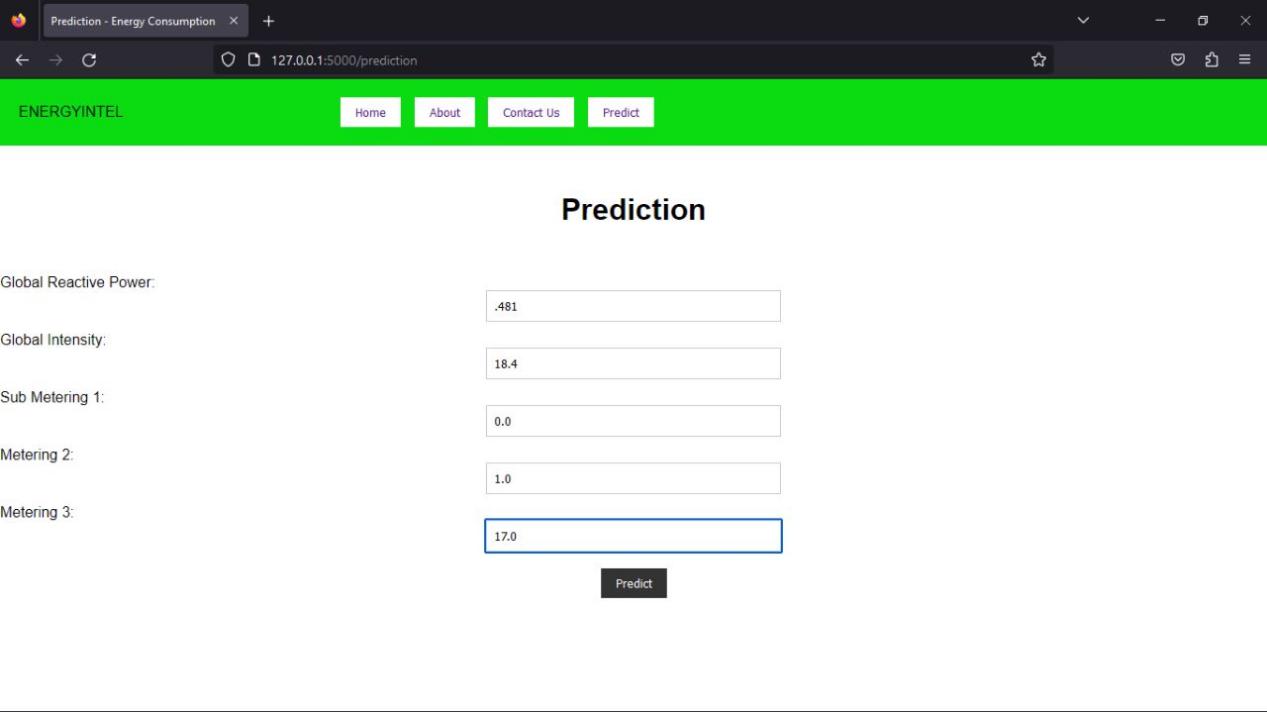
Now, Go the web browser and write the localhost url (http://127.0.0.1:5000) to get the below result

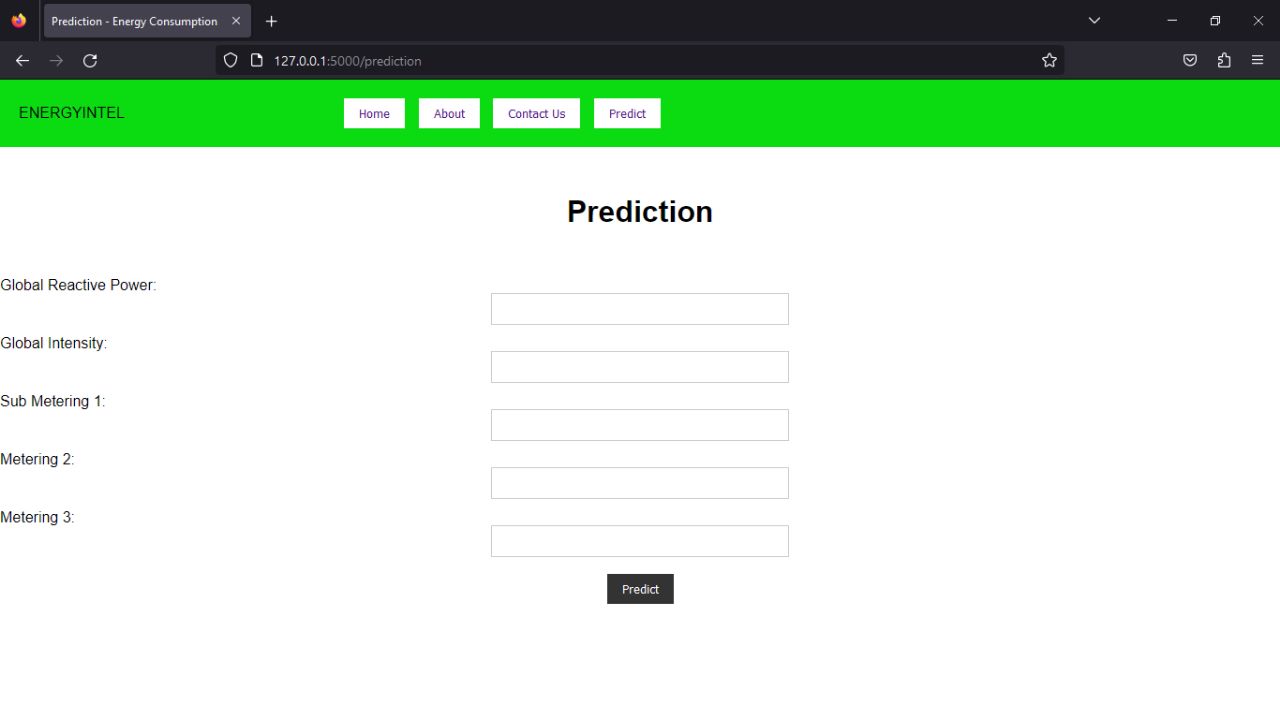












Outro & Conclusion

Hence, the above shared project report shows the whole & sole of this project which is named as a

“**A Reliable Energy Consumption Analysis System for Energy-Efficient Appliances”.**

Refrences

https://aws.amazon.com/blogs/industries/voice-applications-in-clinical-research-powered-by-ai-o

n-aws-part-1-architecture-and-design-considerations/