A Project Report

on

Hybrid Energy inspired DC Micro-grid System using Machine Learning

Submitted to the
Savitribai Phule Pune University, Pune
In partial fulfilment for the award of the Degree of
Bachelor of Engineering in Computer Engineering

By

Sakshi Jadhav	B190294222
Samruddhi Khairnar	B190294231
Kunika Narnaware	B190294239
Pranjal Shewale	B190294262

Under the guidance of

Dr. B. S. Tarle

Department of Computer Engineering

(Accredited by NBA)

MARATHA VIDYA PRASARAK SAMAJ'S
KARMAVEER ADV. BABURAO GANPATRAO THAKARE COLLEGE OF ENGINEERING
Udoji Maratha Boarding Campus, Gangapur Road, Nashik, Maharashtra 422013

(2023-24)

SAVITRIBAI PHULE PUNE UNIVERSITY
2023-24

CERTIFICATE

This is to certify that the project report entitled

"Hybrid Energy inspired DC Micro-grid System using Machine Learning"

Submitted by

Sakshi Subhash Jadhav	B190294222
Samruddhi Mahesh Khairnar	B190294231
Kunika Ratnakar Narnaware	B190294239
Pranjal Vijay Shevale	B190294262

are bonafide students of this institute and the work has been carried out by him under the supervision of **Dr. B. S. Tarle** and it is approved for the fulfillment of the requirement of Savitribai Phule Pune University, for the award of degree of **Bachelor of Engineering**, Computer Engineering.

Dr. B. S. Tarle Guide / HOD	External	Dr. S. R. Devane Principal

Place:- Nashik
Date:-

Acknowledgement

We would like to express our sincere gratitude to our Project guide and Head of Department, Dr B.S.Tarle, for guiding us on choosing projects for effective development of the relevant technical skills in us and also for guiding us with our project work and helping us compile the project report for the fulfillment of the same.

We are grateful to our project coordinator Mrs. R. P. Chandwadkar for guiding us through the project process as well as the entire staff of our department for being cooperative and helping us whenever we were in need.

We are also grateful to the Principal, Dr. S. R. Devane, for giving us the opportunity to work on such projects as part of our curriculum, to enhance our technical and soft skills, in a structured manner.

And lastly, we would like to express our gratitude towards the fellow students of our class for creating a competitive atmosphere whilst exploring a multitude of domains in the field of Computer Science, which we got to learn from the project presentations.

B190294222 - Sakshi Subhash Jadhav
B190294231 - Samruddhi Mahesh Khairnar
B190294239 - Kunika Ratnakar Narnaware
B190294262 - Pranjal Vijay Shevale

Abstract

Traditional power plants make use of manual demand and generation matching processes, causing time delays and wastage of electric power. Hence, our project aims at the creation of a smart management system for hybrid power-plants where the power generation can be controlled as per the estimated power demand. Renewable sources like solar can power the grid while the load demand is low and as per the increase in the estimated power demand, the conventional sources' output power can be used to make up for the excess demand. We make use of LSTM (Long Short Term Memory) based machine learning models to estimate the load demand of the users, based on their historical power usage readings, to ensure that there is minimum wastage of the electrical power as the generation aspect is handled. For our hardware prototype, we make use of a simulated motor load and by measuring the voltage drop and line current over a period, we obtain the load readings on which the LSTM model is trained, to learn the power usage pattern of the load. The LSTM model's predictions are then used to check whether our renewable source - solar - is capable of providing that much power. If the demand exceeds the renewable generation, we provide alerts (meant to control the output power of the conventional source). Hence, our project simulates a Hybrid DC micro-grid.

Keywords - Energy Demand Forecasting, Hybrid Energy Controller, Energy Management System, Internet of Things, Machine Learning.

Contents

List of Figures

List of Tables

Chapter 1

Hybrid Energy inspired DC Micro-grid System using Machine Learning

1.1 Introduction

A smart grid is an electrical grid which includes various operations and energy measures including: advanced metering infrastructure, smart distribution boards, load control switches, energy efficient resources.

Artificial Intelligence can be integrated with smart grids, to facilitate energy demand forecasting and output prediction, to ensure minimal wastage of electricity. Also, renewable energy sources, having an intermittent nature can be integrated in such a system.

Hence, our project is aimed at building an energy management system software, to use machine learning to control and regulate renewable and non-renewable energy modules in a DC micro-grid system, to minimize outages and maximize the output of the system by efficient use of energy demand forecasting.

1.2 Problem Statement

We must gradually shift our focus to integrating non-polluting renewable energy sources with the conventional grid system to harvest renewable energy economically, while ensuring minimal wastage of electricity generated conventionally.

In traditional thermal power-plants, the demand and generation of electricity is matched using a manual process - the grid operators convey the power demand of the users (using their meter readings) to the power-plant operators, who in turn control the power output of the plants. This process causes time delays and wastage of electrical energy. Hence, a smart hybrid energy management system is necessary to override the communication latencies for minimum wastage of electricity while controlling the integration of various energy sources.

1.3 Motivation

Sustainability of any country leads to its growth over a long term. Thus, we must shift gradually towards the use of renewable energy as making a drastic shift might be costly. This implies that Hybrid Power Plants will play a vital role in the power industry in coming years. This is what motivated us to build a hybrid DC micro-grid, in fulfillment of the final year project.

Also, most smart grid projects focus on controlling the management of the energy that is already generated whereas our idea aims to automate the generation aspect in real time. If our system is deployed in thermal power-plants, we can minimize the wastage of electricity, while establishing profitable relationships with the electricity manufacturing companies by reducing their manual labor and time delays.

Thus, we intend to build an energy management system, for controlling the power output of the sources according to the energy demand of the users.

1.4 Current Statistics of Indian Power sector

Installed capacity of thermal power in India is 237.26 GW which accounts for 60 % of the total electricity generation in India. The rest of which is derived from non-renewable energy sources.

64 % of the consumers are located in rural areas, where there is an abundance of renewable energy from biomass, wind etc. Hence these resources can be integrated with the main grid to reduce the average cost per unit of electricity, which currently is ₹7/unit (average cost as of April '23). [11]

1.5 Objectives of Work

The Objectives of the project work are as follows:

1. Predict the power output generated by renewable energy generators, using illumination and wind-speed sensors.
2. Predict load demand of the users in the grid using their previous power usage readings, according to their observed behaviour, by measuring the voltage drop and line current at their end.
3. Switch and regulate the energy sources based on the load / consumption of the users as per the estimates. For uncertain demands, make use of a battery backup module.
4. Work on securing a wireless setup of the system from cyber attacks as a part of future scope.

1.6 Project Scope

1. Assembling the entire wired setup of the DC micro-grid and executing the switching and regulation of the sources as well as the backup energy sources (battery) according to the load demand.
2. Training multiple Machine Learning and Deep Learning models to estimate the power generation and forecast the load demand.
3. Building our own dataset consisting of sensor readings, to measure the illumination, wind speed and other parameters relevant to our energy sources.
4. To try obtaining as good an accuracy at the predictions as mentioned in our base paper, by the author, work and investigate more on improving the accuracy using newer techniques like Ensemble Learning.
5. Building a wireless micro-grid and securing it against data manipulation attacks, as future scope.

Chapter 2

Literature Survey

2.1 Overview

The base paper [1] suggests the use of a two-stage machine learning based energy dispatch management system for hybrid power plants (HPPs) , designed to control renewable energy sources, reserve and backup energy sources. The first stage aims to forecast the power output of renewable energy sources, as well as the load demand. The second stage aims to coordinate the energy output of the reserve and backup sources to achieve the required objective.

The second paper [2] discusses about distributed energy resources (DERs) being largely interconnected to electrical power grids, causing significant impact to the operation of traditional centralized generation power plants and the dispatch control centers. Further, the authors introduced the use of a simplified LSTM algorithm to forecast one day-ahead solar power generation, elaborating on data processing, model fitting, cross validation, metrics evaluation, hyper-parameters tuning and the results showing as average RMSE of 0.512 which offers quite a promising solution for short-term solar power forecasting applications. The authors made use of raw data collected from Photo Voltaic (PV) systems to train a model using a long short term memory approach to forecast one day-ahead solar power generation.

The third paper [3] deals with a residential hybrid thermal grid-connected home energy system, including a fuel-cell and a battery as Energy Storage System (ESS). A day-ahead scheduling algorithm for managing different resources is developed to generate an efficient look-up table that determines an optimal operation schedule for the distributed energy resources at each time interval, so that the operation cost of a smart house is minimized. We plan to use a similar approach for our project and will try to include additional functionalities of controlling conventional power generation.

[4] Suggests an energy management system for HPPs powered by PV and diesel batteries. [5] Suggests the use of hourly scheduling of energy sources.

2.2 Limitations

1. The gaps which we identified in the base paper [1] were that all of the model training and testing was done using simulated datasets and not on any actual grid / micro-grid system.
2. The second paper [2] made use of raw data collected from PV systems to train a model using a long short term memory approach to forecast one day-ahead solar power generation but did not explore other renewable energy sources.

2.3 Current Market Survey

Presently, in India, power grids are manually controlled. However, there are numerous pilot projects, some of which have been completed and some undergoing development, under the National Smart Grid Mission, established by the Government of India, since 2015. Below are the Smart Grid pilot projects sanctioned by the Ministry of Power, completed in India.

Sr No.	Location	Functionalities Adopted
1.	PED, Puducherry	Advanced Metering Infrastructure
2.	IIT Kanpur	Renewable Integration
3.	UHBVN, Haryana	Outage Management
4.	UGVCL, Gujarat	Outage Management
5	HPSEB, Himachal Pradesh	Power Quality Measurement
6	AVVNL, Ajmer	Advanced Metering Infrastructure
7	APDCL, Assam	Distributed Generation
8	WBSEDCL, West Bengal	Advanced Metering Infrastructure
9	SGKC, Manesar	Micro-Grid / Distributed Generation, EV with Charging Infra, Home Energy Management, Cyber Security and Training Infrastructure
10	TSSPDCL, Telangana	Advanced Metering Infrastructure
11	TSECL, Tripura	Advanced Metering Infrastructure
12	CESC, Mysore	Micro-Grid

Table 2.1: Smart Grid Pilot Projects in India

GLOBAL

Smart Grid market size was valued at USD 31.5 billion in 2022 and is expected to grow up to USD 108.238 billion by 2032, at an annual growth rate of 14.70%. Countries having established smart grids are : the U.S.A., Canada, Germany, France, the UK, Italy, Spain, China, Japan, India, Australia, South Korea, and Brazil.

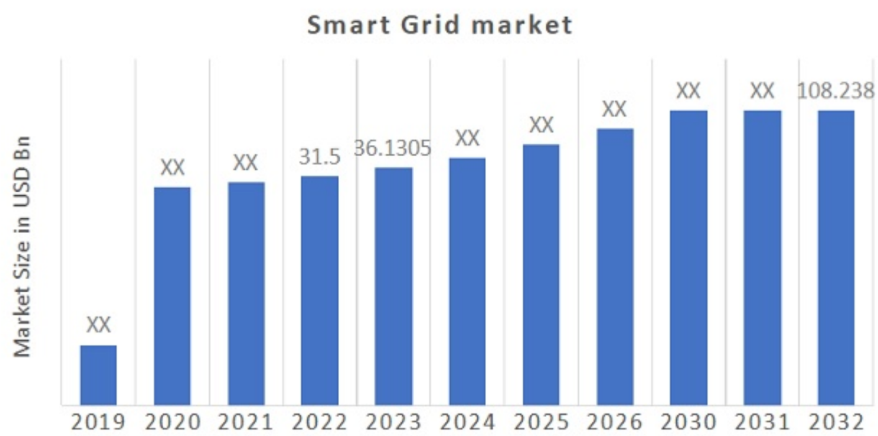


Figure 2.1: Smart Grid Global market size [6]

TECHNOLOGICAL INSIGHTS

Smart grids can either be wired or wireless (transferring information over a distance without electrical conducting wires). A reliable, secure, and low-latency bi-directional communication infrastructure is required between the intelligent electronic devices and the control center to implement wireless communication in a substation. Wireless technology ensures cost savings and secure data transmission compared to wired technology. This technology offers utility support by offering high bandwidth, covering a large area, and optimizing complex logistics and production processes. [7]

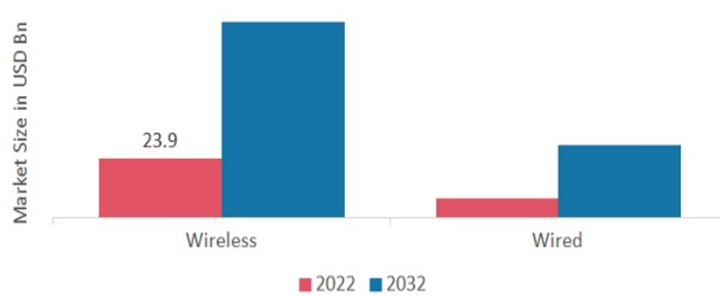


Figure 2.2: Wired vs Wireless Grid market size [6]

Companies leading the smart grid market are - IBM Corporation (US), Schneider Electric (France), Open Systems International (US), Oracle Corporation (US), Wipro Ltd (India).

Chapter 3

Design Details

3.1 PHASE 1: Requirement Analysis

3.1.1 Software Requirements

Operating System - Linux
The proposed system is designed to work on Linux operating systems as all distributions of Linux come with Python installed out of the box. Also, as mentioned in our future scope, working on securing a wireless model requires the use of security and forensics tools, which are free to install and use, on the Linux platform.

3.1.2 Minimum Hardware Requirement

- **Microprocessor** - Intel Core i5 / i7 (= 6th gen)
- **RAM** - 4 GB or more (on AWS Virtual Machine)

3.1.3 Functional Requirements

System Features

We aim to offer our users, an energy management system which will :
1. Predict the power output generated by renewable energy generators.
2. Predict load demand of the users in the grid using historical power usage readings.
3. Switch and regulate the energy sources based on the load / consumption of the users in real time.

Database Management System (DBMS)

We will be aggregating the sensor readings - illumination, wind speed, using a Raspberry Pi Pico and pushing them onto the cloud storage, in the form of JSON. On cloud, this data will be stored as CSV files, hence our databases will actually be datasets along with trained models as pickle files.

3.1.4 External Interface Requirements

User Interfaces

1. An intuitive and easy to interpret dashboard showing the integration of energy sources.

Software Interfaces

The proposed system will be a set of Python scripts, which will estimate the load power consumed by the users as well as the output power of the powerplants and generate a control sequence to control the switching and regulation of all the sources. The system will be deployable on local as well as remote machines. The admin of the system (electricity board users) will be able to modify the scripts to change the parameters of the machine learning as well as switching phases.

Hardware Interfaces

The proposed system will be executed in the form of a prototype, simulating an actual DC micro-grid.

Hardware Components

- Raspberry Pi Pico W
- Solar Panel, Charge Controller and Lead Acid Battery
- DC Adaptor, Motor, Relay
- For future modules - RF Transceivers, Wireless Network Adaptor, Steam generator and steam burner flame controller.

Communication Interfaces

For training of the machine learning models, to control the system, we will require computers having 8 GB RAM and 32 GB of storage, running Linux operating systems to ensure security and responsiveness. These requirements will be harnessed through the use of remote virtual machines, on the AWS EC2. Our trained model will be deployed on a virtual machine hosted on AWS EC2. For our prototype, a single 8 GB RAM and 32 GB storage space VM will be enough.

Network and Communication Protocols

- **IEEE 802.11 (WiFi)** :
  Wireless Fidelity, is widely used to enable wireless internet access in various devices, including smartphones, laptops, and IoT devices. It operates using the IEEE 802.11 family of protocols, which define the standards for wireless communication. We will make use of this protocol to transfer data to and from the Raspberry Pi module and the AWS virtual instances.

- **SSH (Secure Shell)** :
  It is a network protocol that allows secure remote access to and control of devices and servers over an encrypted connection, using strong encryption and authentication methods.

3.1.5 Non-Functional Requirements

Software Quality Attributes

1. **Reliability** : As the models will be trained to fit the data just correctly, the system will give reliable estimates.
2. **Portability** : Accessible from anywhere, as the dashboard will be hosted on cloud.
3. **Usability** : User friendly dashboard.

Operational and Maintenance Considerations

For maintenance of the trained model and prevention of corruption of the files, we will schedule cronjobs for repeatedly training the model at regular intervals.

Scalability and Performance

The proposed system is designed, considering a DC microgrid; hence, for the machine learning part, we will make use of only virtual machines for the training purpose and serving of the dashboard module. As the system scales, we can integrate multiple remote virtual machines for the training and serving purpose. Hence, our system is scalable, considering the fact that we are using Apache Spark for distributed machine learning and a lightweight, responsive dashboard library - Dash.

Security Considerations

As part of security, we hope to secure the physical setup of our project from external threats like fire and water. Also, as detailed in the scope of this document, we will work on securing our system against man in the middle attacks.

Security and Access Control Constraints

We will ensure that the virtual instance can only be accessed by authorized persons, through SSH, by storing the private keys required to login, in hidden files, which will be password protected.

3.2 PHASE 2: ANALYSIS PHASE

3.2.1 Technologies Used

Programming Languages and Technologies

- **Python 3** - It is a high-level, general-purpose programming language.
- **Apache Kafka** - It is a distributed event store and stream-processing platform.
- **Apache Spark** - It is an open-source analytics engine for large-scale data processing.

Development Tools and Environments

- **JupyterLab** - It is a web-based interactive computing platform.
- **AWS EC2** - Virtual machines on the Elastic Cloud Compute platform will enable us to deploy trained models and host the dashboard software.
- **Colab** - To speed up the training process using GPUs offered by Google's web notebooks.

Software Frameworks and Libraries

- **Pandas** - Open source data analysis and manipulation tool, using Python.
- **Scikit-Learn** - Library for implementing machine learning in Python.
- **Tensorflow** - Library for implementing deep learning in Python.
- **Confluent-Kafka** - Library for using Kafka's publish-subscribe model using Python.
- **PySpark** - Library for distributed machine learning for big data, using Spark, in Python.
- **Dash** - Library for building dashboards using Python.

3.3 PHASE 3: DESIGNING PHASE

3.3.1 System Architecture Diagram

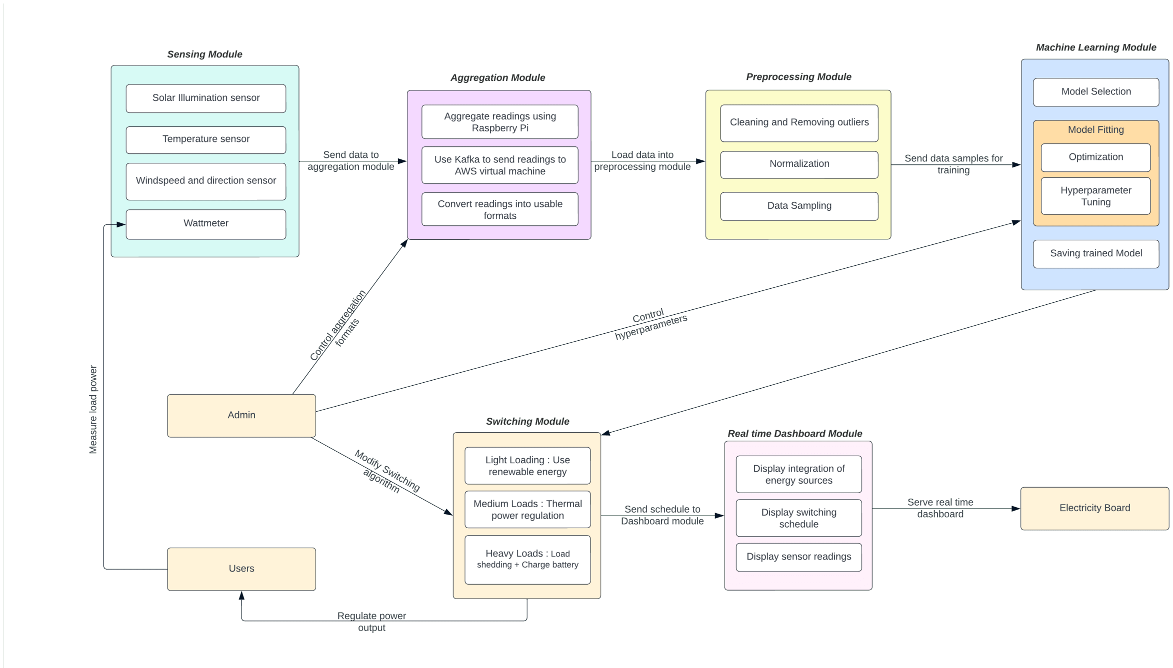


Figure 3.1: System Modules Overview

Sensing Module

The sensing module is responsible for obtaining readings for generating solar, wind, and users' power demand datasets over time to study the renewable energy generation patterns and users' power usage patterns and provide predictions.

1. An illumination sensor is used to obtain readings of sunlight falling on the operating area of the system, along with the assistance of a temperature sensor.
2. To estimate the wind power generation, wind speed and direction sensors are employed.
3. A combination of voltmeters and ammeters embedded in the users' end will be used to obtain load demand readings to assist the energy demand forecasting.

Aggregation Module

This module is responsible for aggregating the sensed readings using a Raspberry Pi, formatting the readings, and sending them to a cloud setup - a virtual machine running on AWS's EC2 using Apache Kafka's publish-subscribe model.

Preprocessing Module

This module has the responsibility of cleaning the aggregated data - removing anomalous points/outliers, normalization of the dataset, and sampling the dataset to pass a subset of data to the machine learning module.

Machine Learning Module

This module of the proposed system will be responsible for fitting models to the aggregated and preprocessed datasets and consists of selecting machine learning algorithms, training them on the data, tuning their hyperparameters, validating the model's accuracy using MAE and MSE metrics, and saving the trained model. Algorithms - SVM (Support Vector Machine) Regressor and deep learning algorithms - LSTM (Long Short Term Memory) and RNN (Recurrent Neural Network) will be made use of.

Switching Module

This module will switch/regulate the energy sources based on various operating cases.

Dashboard Module

This module is responsible for displaying the aggregation of renewable power and conventional power generation by the system as a hybrid energy controller and their actual production versus the estimates provided by the trained models. This module is intended to assist electricity manufacturing companies and power plant operators.

3.3.2 Data Flow Diagrams

Data Flow Diagram 0

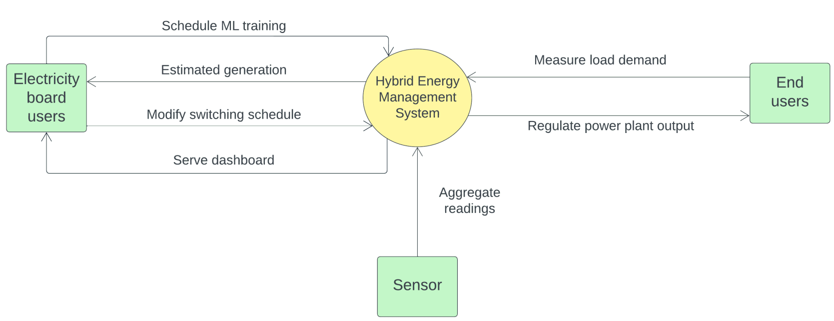


Figure 3.2: DFD Level 0

Data Flow Diagram 1

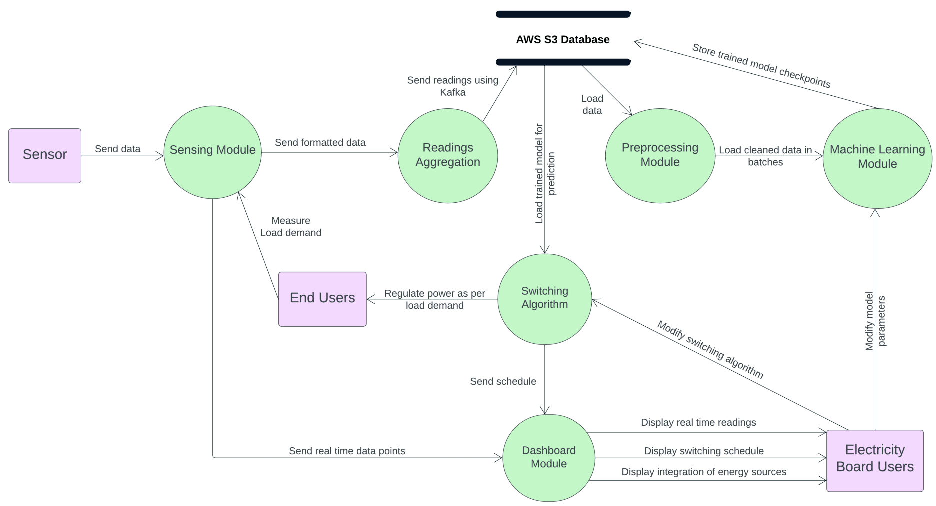


Figure 3.3: DFD Level 1

3.3.3 E-R Diagram

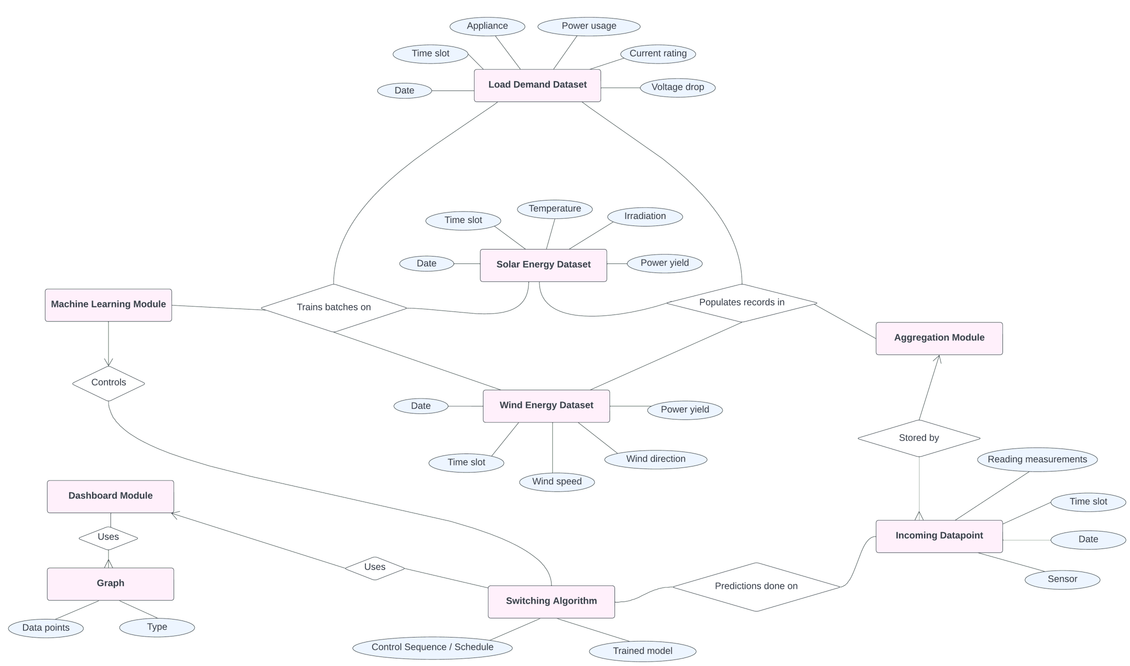


Figure 3.4: ER Diagram

3.3.4 Use Case Diagram

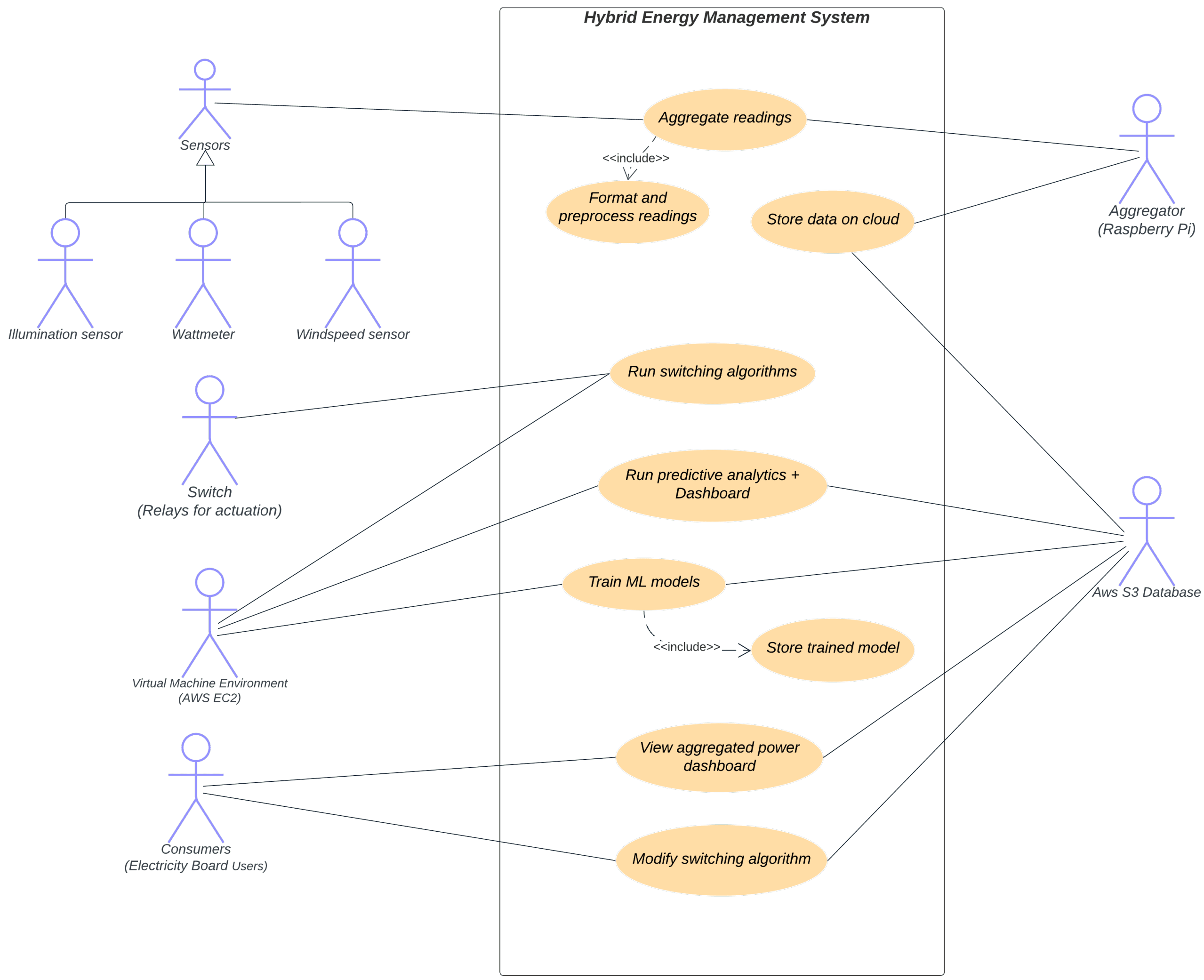


Figure 3.5: Use Case Diagram

The use cases of the system include -

- Sensors will send their reading to the aggregator which will push the readings on the cloud setup.
- The remote setup running on a virtual machine will run ML, predictive analytics, switching as well as dashboard scripts, which will be consumed by the users.
- The switching algorithm will generate a control sequence and control the flow and generation of electricity using relay modules.

3.3.5 Class Diagram

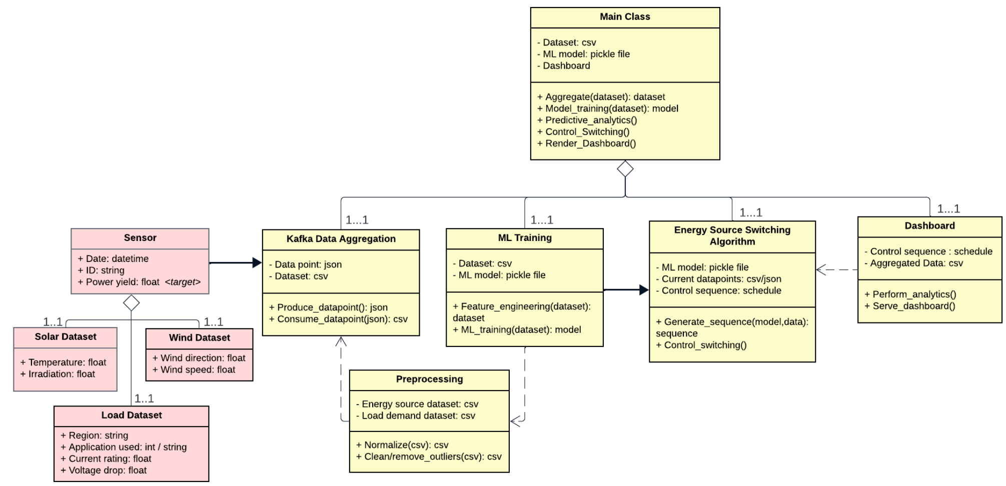


Figure 3.6: Class Diagram

3.3.6 Sequence Diagram

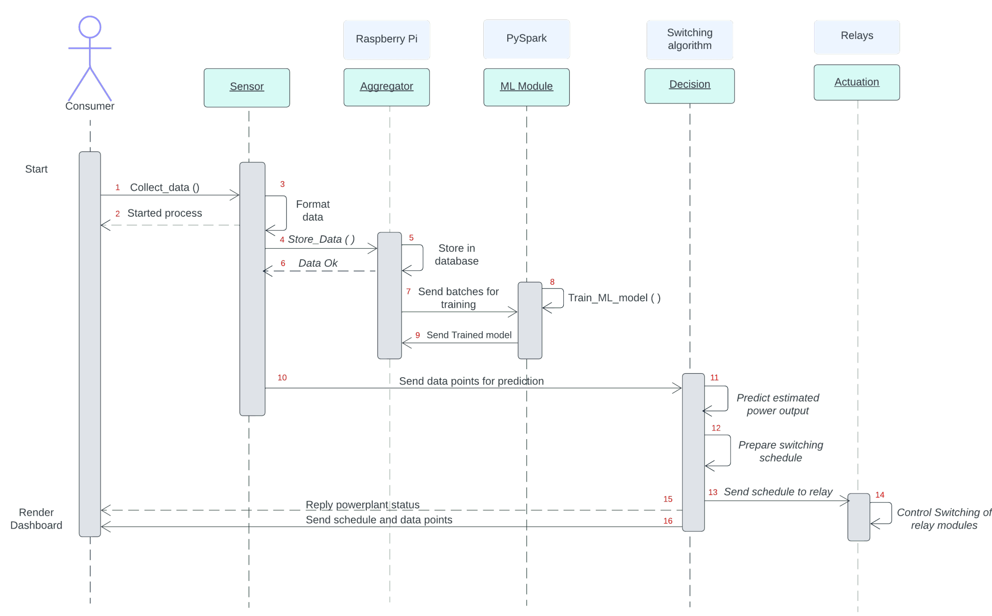


Figure 3.7: Sequence Diagram

3.3.7 State Diagram

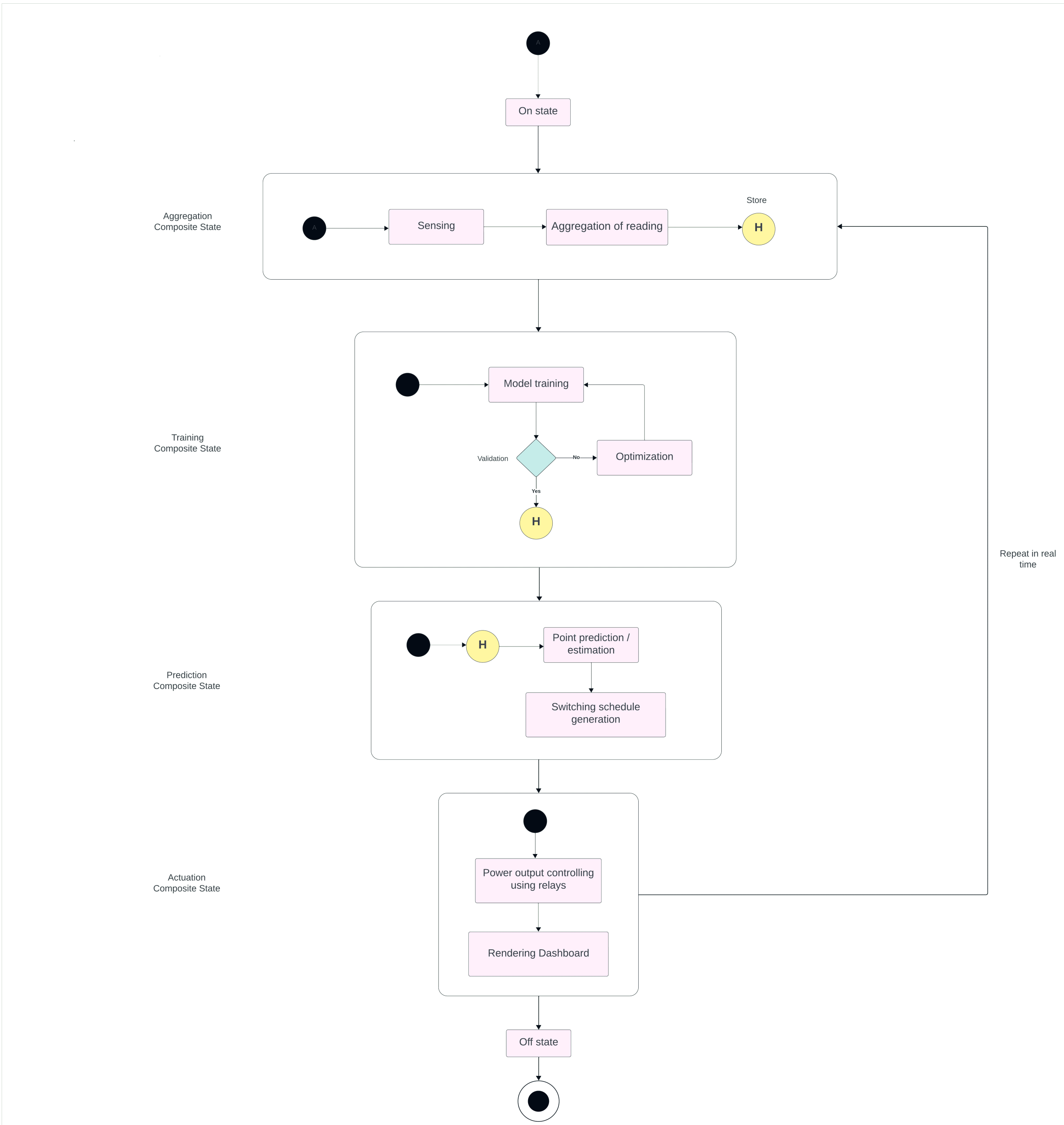


Figure 3.8: State Chart Diagram

3.3.8 Activity Diagram

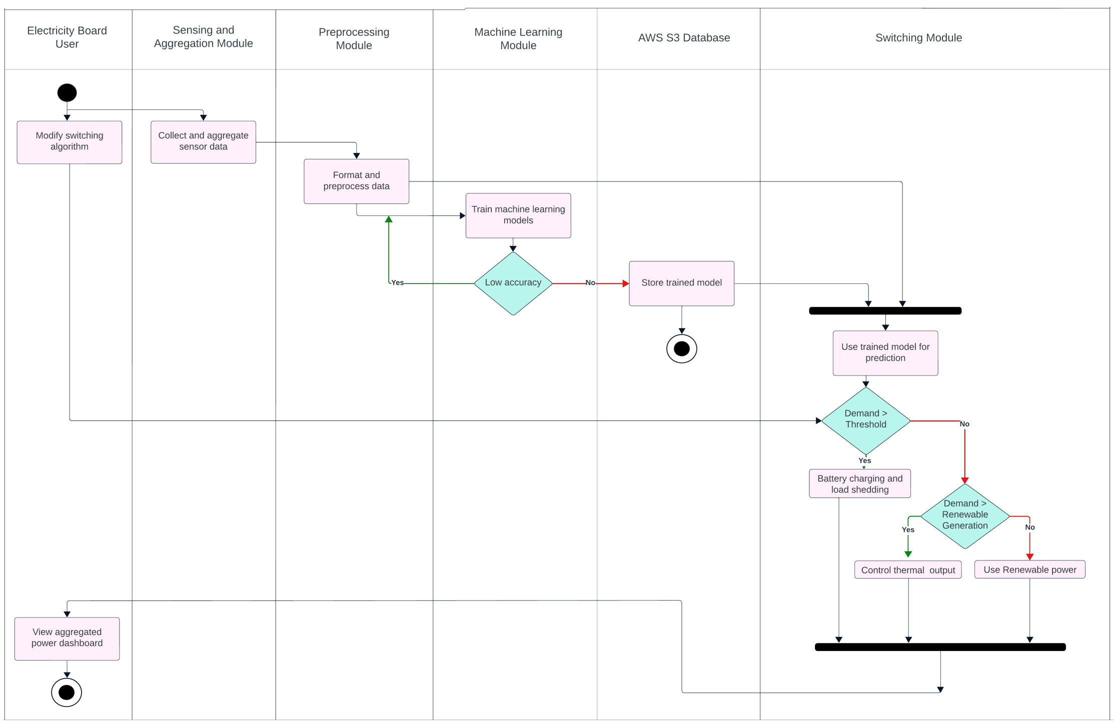


Figure 3.9: Activity Diagram

3.3.9 Package Diagram

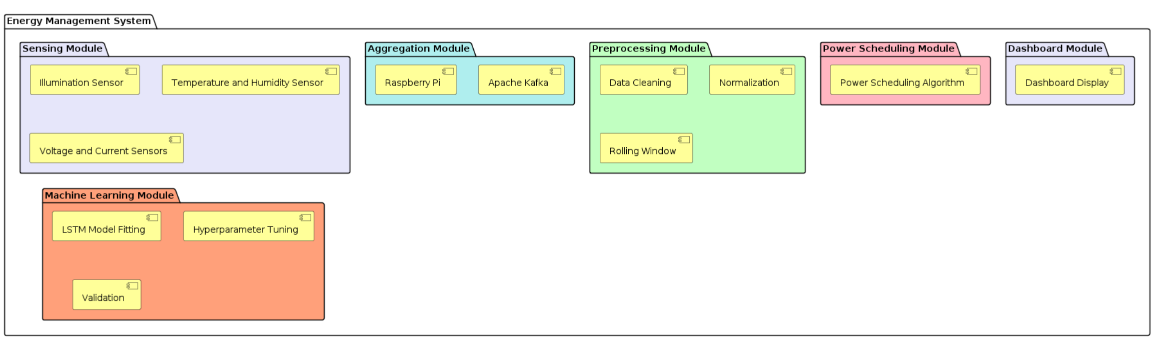


Figure 3.10: Package Diagram

The package diagram shows how our Energy Management System project is divided into different parts. Here's a breakdown:

Energy Management System Package:

This is the main part of our project. It shows all the different pieces of our system and how they fit together.

Sensing Module:

This part deals with collecting data from sensors, i.e. those that measure light, temperature, and electricity.

Aggregation Module:

Here, we bring all the sensor data together in one place using a Raspberry Pi and Apache Kafka.

Preprocessing Module:

This section cleans up and prepares the data for analysis by removing errors and making it easier to understand.

Machine Learning Module:

In this part, we use fancy math (machine learning) to analyze the data and make predictions about energy usage.

Power Scheduling Module:

This part decides how to distribute power efficiently based on the predictions made by the machine learning module.

Dashboard Module:

This is where we show all the information and predictions in an easy-to-understand way.

3.3.10 Deployment Diagram

The deployment diagram shows where all of the modules / parts are located physically. Here's what it tells us:

Nodes:

These are the places where we put our system's parts, like computers or devices.

Artifacts:

These are the individual pieces of software or data that make up our system, like the sensing module or the machine learning module.

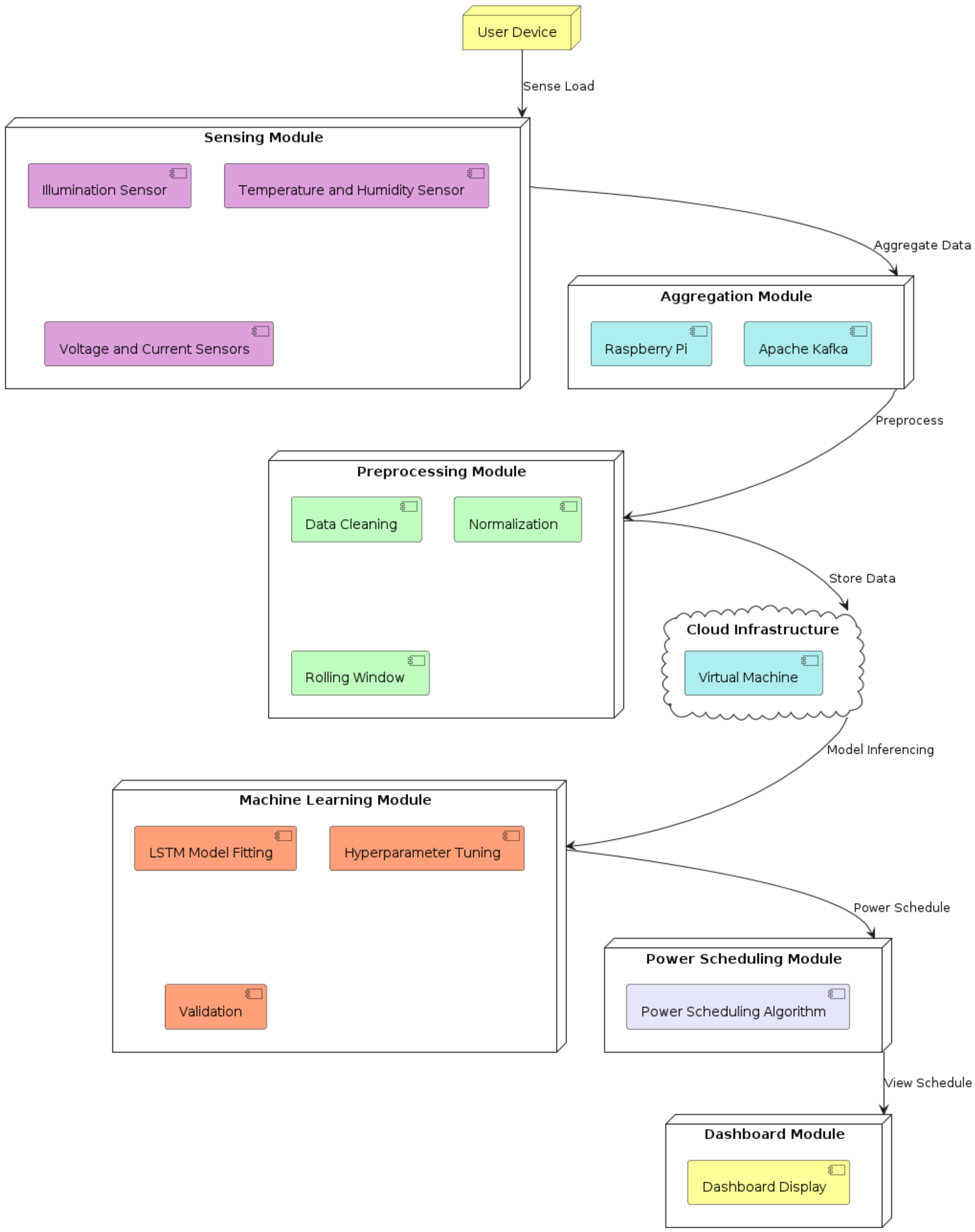


Figure 3.11: Deployment Diagram

Relationships:

This shows how the different parts of our system communicate and work together, i.e. how data moves between the sensing module and the machine learning module.

3.4 PHASE 4: PLANNING PHASE

3.4.1 Operating Environment

Our system will work in the physical setup of our DC microgrid, connected with the virtual machine instance running the machine learning scripts, via the Internet / WiFi hotspot.

3.4.2 Design Implementation

Hardware module

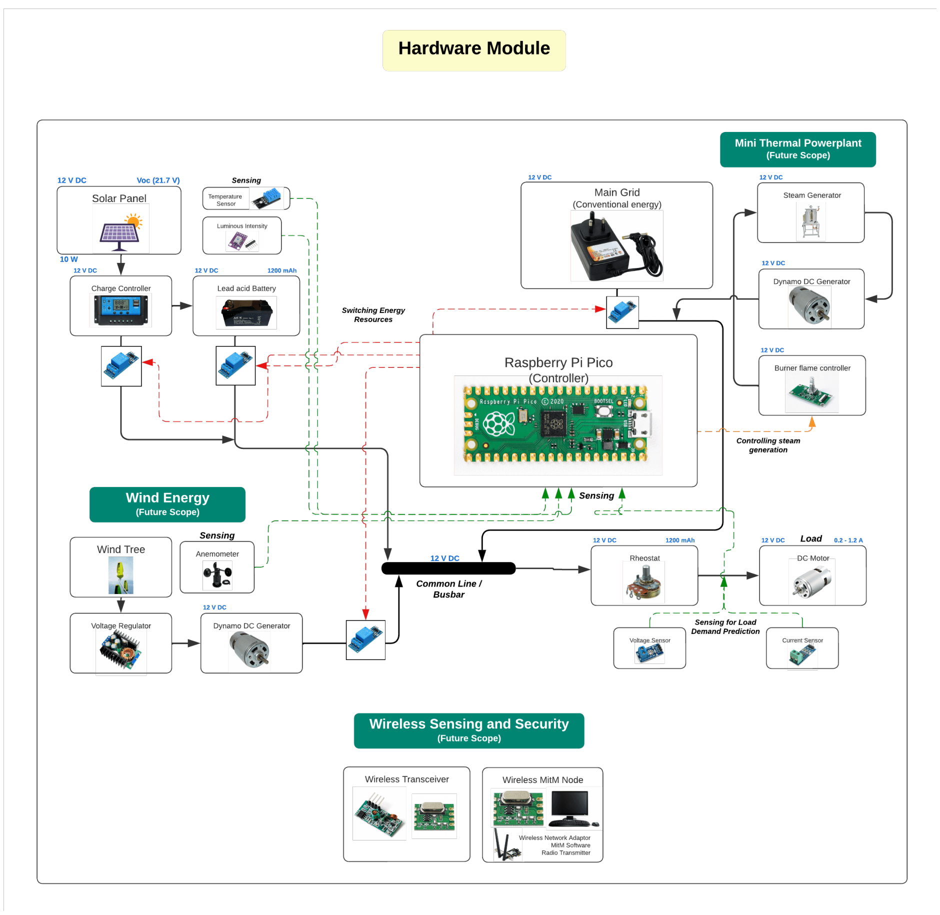


Figure 3.12: Hardware Module

1. **Mini Solar Energy Power Plant** : Comprising a 12V, 10W Solar Panel, a Charge Controller and a 1200 mAh Lead Acid Battery, this module will be responsible for

harvesting the solar energy and powering the microgrid. Also, we will deploy Luminous Intensity and Temperature Sensors to obtain readings in order to predict the power generation output.

2. **Main Grid** : As the conventional energy source in our project, we are going to make use of a stable 12 V DC connection using an Adaptor which will be connected to the Main Power Grid.

FUTURE SUB-MODULES :

3. **Mini Wind Energy Power Plant** : Comprising a Wind Tree Module, a 12 V Dynamo Generator and a Voltage Regulator, this module will be responsible for harvesting the wind energy. Also, we aim to deploy an Anemometer to obtain wind speed readings in order to predict the power generation output of the entire module.
4. **Mini Thermal Power Plant** : This module will consist of a Steam Generator, controlling a 12 V Dynamo, powered by a burner. We aim to deploy a Burner Flame Controlling Unit to control the power output of the thermal plant in real time.
5. **Wireless Sensing and Cyber Security Module** : This module will make use of Wireless Sensors (using Radio Frequency) instead of the above listed wired ones. To secure this wireless smart grid, we will try to execute data manipulation attacks on the wireless communication going on between the transmitter and the receiver and suggest possible security measures in opposition to the attacks.

Remote Machine Learning Module

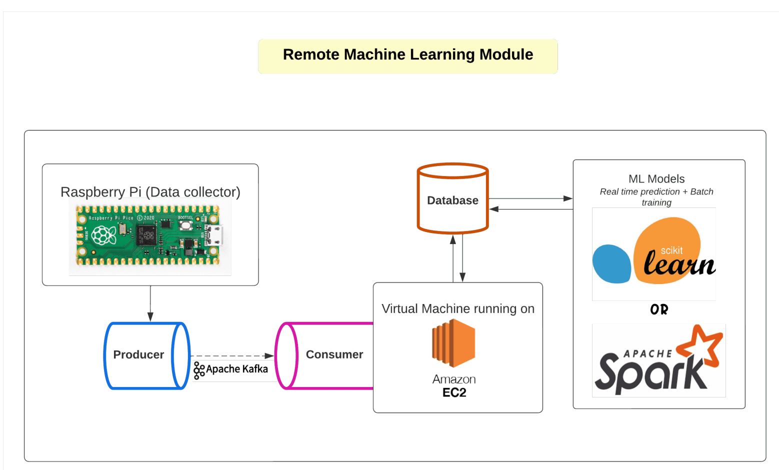


Figure 3.13: Remote Machine Learning Module

1. **Data Acquisition** : This module consists of a Raspberry Pi Pico which will obtain the sensor readings and control the actuation / management and switching of the mini power plant modules as the power load.
2. **Data Concentration** : The Raspberry Pi will stream the readings using Apache Kafka to a remote consumer program residing on an Amazon EC2 virtual machine, where all readings will be concentrated and machine learning programs will be executed.
3. **Machine Learning** : This module will consist of the batch training of the gathered data as well as real time prediction of the power generation estimates, load estimates to control the switching of the power plants in real time.

Dashboard Module

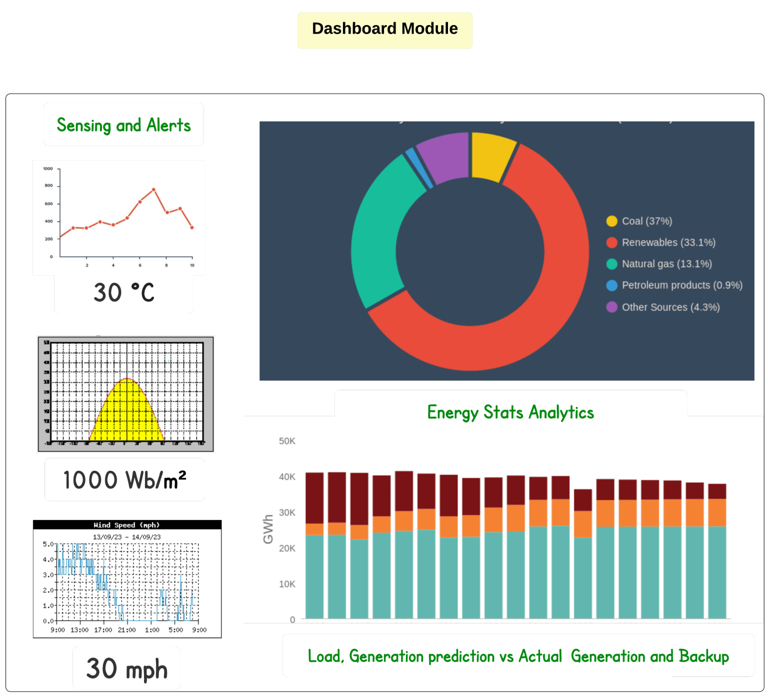


Figure 3.14: Dashboard Module

This module will consist of a real time dashboard to show the sensor readings over time and to give the users a visual representation of how the predicted load and generation estimates match with the actual power consumption and generation and how the switching of the energy sources is handled, to account for the minimum power wastage.

3.4.3 Assumption and Dependencies

- **External Hardware Dependencies**
  The proposed system will make use of AWS hosted virtual machines. Hence, the owner of the hardware running the code will be a third party - AWS. Hence, the security of our data, beyond authentication mechanisms, is in the hands of AWS. To tackle any discrepancies arising due to this, we can make use of a local machine instead.
- **Software Dependencies**
  The proposed system will run on locally installed softwares - Kafka, Spark hence there will not be the use of any third party softwares unless we opt for remote versions of these softwares, to ease the process.
- **Data Source Dependencies**
  The proposed system will be making use of datasets procured from the IEEE Dataport. Hence, our system will rely on this data, aggregated by a third party.

3.4.4 Testing and Quality Assurance

1. Regression Testing will be made use of, to ensure that new code changes or don't affect the existing functionalities.
2. To assess how the system performs under various conditions, like - high traffic asking to serve the dashboard, high rate of incoming sensor readings etc, we'll make use of Load and Performance Testing methods.
3. Testing of the various cases of operation of the micro-grid system - light and heavy loading and how our system reacts to the cases.
4. Testing the performance of the system using different ML algorithms.

3.4.5 User Documentation

User Manual

To guide our users through the process of installing our system locally, instead of a virtual machine, we will prepare a detailed documentation, listing all the steps regarding the setup of the system as well as annotated source code so that the users can customize our system according to their requirements.

Chapter 4

Project Plan

4.1 Project Estimates

4.1.1 Reconciled Estimates

Time and Schedule Estimates

Regarding the schedule, the entire project can be implemented in a time frame of 2 months, starting - December 2023. Beyond which, we can work on fulfillment of the items we previously listed as part of our future scope.

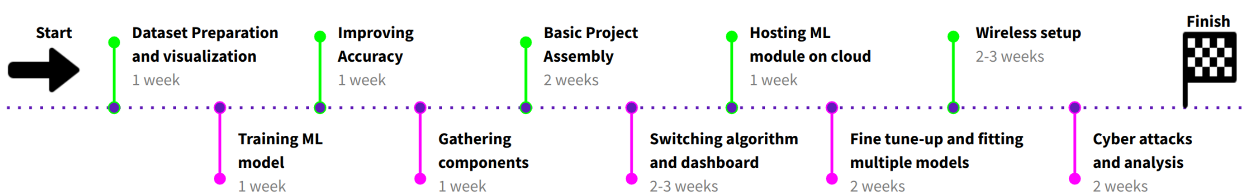


Figure 4.1: Project Schedule

Budget and Cost Estimates

We have limited the budget of our project to ₹10,000 to 15,000 considering the hardware cost of miniature prototype of the DC micro-grid along with the costs required to run the AWS virtual machines.

Regarding the software development cost of the project using the COCOMO model,

- $Effort\ Estimation\ (E) = a * (KLOC^b) = 3 * 0.8^1.12) \approx 2.144\ person\ months$
- $Development\ Time\ Estimation\ (D) = c * (E^d) = 2.5 * (2.144^{0.35}) \approx 2.55\ months$
- $Cost\ Estimation\ (C) = D * X = 2.55 * 40000\ (median\ salary) \approx$ ₹$1,02,000\ Lakh$

4.1.2 Project Resources

Team Members

- The project requires a development team comprising of experience from the domains of IoT, machine learning and software development.

Hardware

- Sensors (current, voltage, environment conditions)
- Solar panels, battery storage
- Microcontrollers (Raspberry Pi)

Software

- Machine Learning libraries (TensorFlow/ Scikit-learn)

4.2 Risk Management

4.2.1 Risk Identification

The proposed system relies of third party hardware and libraries. Keeping this in mind, we have identified security risks that can most likely occur in the development stage.

4.2.2 Risk Mitigation, Monitoring and Management Matrix

Risk ID	Risk Name	Type	Impact (1-10)	Backup Plan
1	Hardware module / component failure.	Technical Risk	8	Keeping backup components at hand.
2	Remote Consumer (ML program) not reachable.	Operational Risk	5	Keep a local computer as a backup for remote machine learning.
3	Execution of the project in the real world would be challenging.	Business Risk	3	Communicate with stakeholders in the electricity department board, for real-world feasibility analysis.

Table 4.1: Risk Assessment Table

4.2.3 Overview of Risk Analysis

We have identified three risks in the development and execution phases of our project and rated their impact towards the completion of the project on a scale of 1 to 10, 1 being the lowest impact and 10 being the largest impact. Also, we have listed the strategies which we can employ to mitigate these identified risks.

4.3 Project Schedule

4.3.1 Project Task Set

The project tasks are divided into several stages:

- **Dataset Preparation and Visualization:** This involves gathering initial data, cleaning it, and creating visual representations to understand data patterns. This task is scheduled for 1 week.
- **Training ML Model:** Using the prepared dataset, this task involves training machine learning models to make predictions. This task is scheduled for 1 week.
- **Improving Accuracy:** This involves refining the model to improve its accuracy and performance. This task is scheduled for 1 week.
- **Gathering Components:** This task involves collecting all the necessary hardware components required for the project. This task is scheduled for 1 week.
- **Basic Project Assembly:** This involves assembling the basic components of the project. This task is scheduled for 2 weeks.
- **Switching Algorithm and Dashboard:** This involves developing the switching algorithm and setting up the dashboard for monitoring. This task is scheduled for 2-3 weeks.
- **Hosting ML Module on Cloud:** This task involves deploying the trained machine learning model to the cloud for real-time predictions. This task is scheduled for 1 week.
- **Fine Tune-Up and Fitting of Multiple Models:** This involves fine-tuning the deployed models and fitting multiple models for better performance. This task is scheduled for 2 weeks.

4.3.2 Task Network

The task network diagram shows the flow and dependencies of tasks. It starts with dataset preparation, followed by model training, accuracy improvement, component gathering, project assembly, switching algorithm and dashboard development, hosting on the cloud, and fine-tuning multiple models.

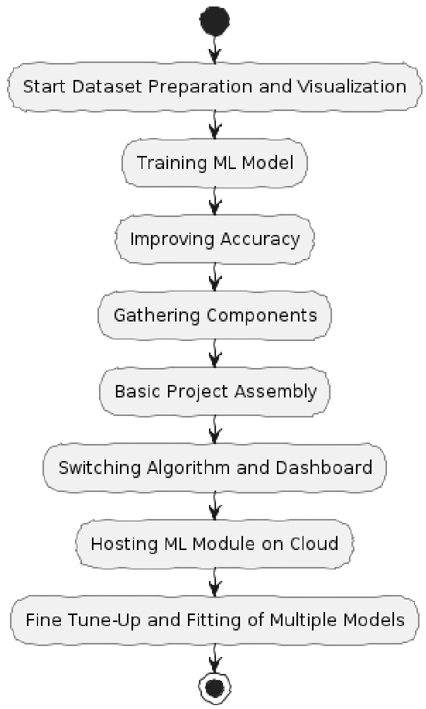


Figure 4.2: Task Network Diagram

4.3.3 Timeline Chart

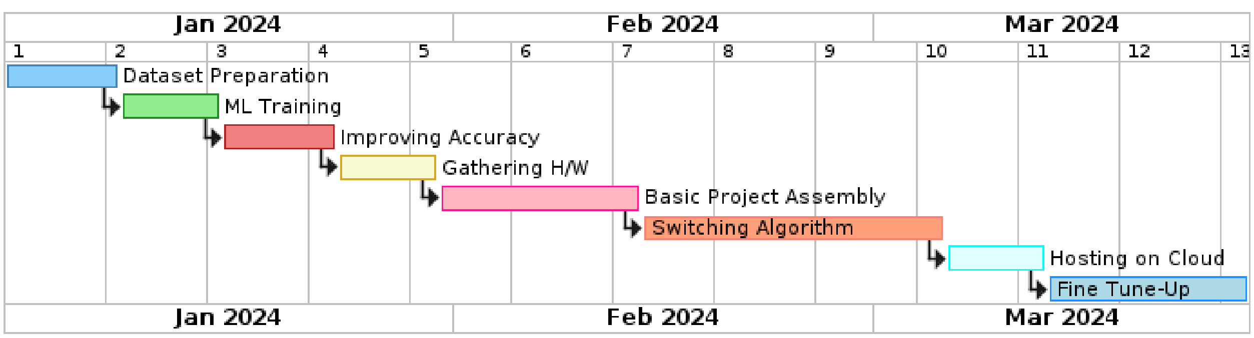


Figure 4.3: Timeline Chart

The timeline chart provides a visual representation of the project schedule, indicating the duration and overlap of each task. Each task is represented with its duration, starting from dataset preparation to fine-tuning models.

4.4 Team Organization

4.4.1 Management Reporting and Communication

Effective management reporting and communication are critical for the success of the project. Regular discussions will be held to discuss progress, address challenges, and ensure alignment with project goals. Communication tools such as WhatsApp and email will be used to facilitate collaboration among team members.

4.4.2 Team Structure

The project team is structured such that member is assigned specific responsibilities based on their skills and experience.

- **Samruddhi Khairnar:** Responsible for machine learning, focusing on LSTM (Long Short-Term Memory) models for predictive analytics and energy management.
- **Kunika Narnaware:** Manages Kafka data flow, ensuring efficient and reliable data streaming within the microgrid system.
- **Sakshi Jadhav:** Responsible for hardware assembly, including the integration of solar panels, battery storage, and sensors.
- **Pranjal Shewale:** Develops the dashboard for monitoring and visualizing the microgrid's performance and energy metrics.

Chapter 5

Project Implementation

5.1 Energy Management System Modules

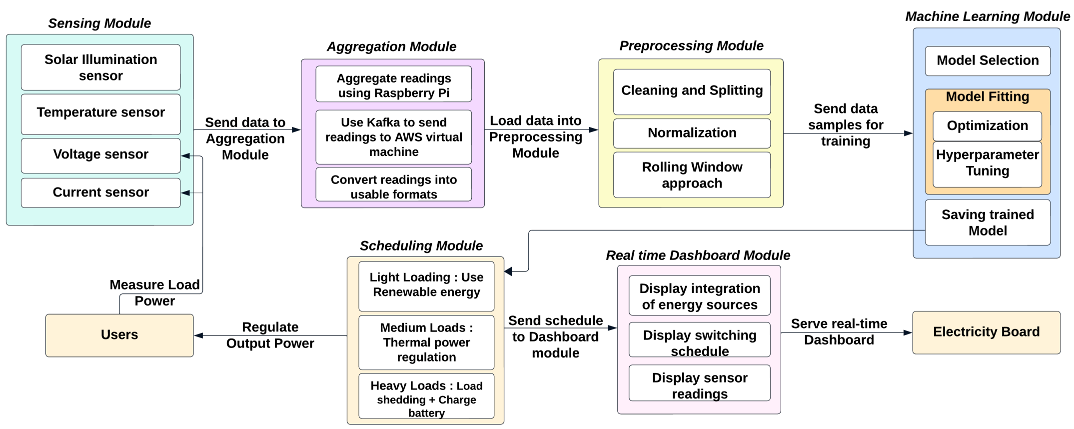


Figure 5.1: System Architecture

5.1.1 System Overview

The proposed system can be deployed in Hybrid power plants, to automate the power generation in real-time, as per the estimated load demand.

1. **Sensing Module:**
   (a) An illumination sensor: for sunlight readings.
   (b) A temperature and humidity sensor: weather data.
   (c) A combination of voltage and current sensors, to obtain load demand readings.
2. **Aggregation Module:** Aggregation of sensor readings using a Raspberry Pi, sending them to a virtual machine running on the AWS Cloud using Apache Kafka.

Algorithm 1: Naive Power Scheduling Algorithm

```
Input: Sensors, Window size=60, Burst time=30 minutes
Output: None
while True do
    Illumination, Voltage, Temperature, Humidity ← Get readings(Sensors);
    Current ← Solar model.predict();
    Estimated solar power ← Voltage * Current (mA) * 0.001;
    Size ← length(Historical Load readings);
    Windowed readings ← Historical Load readings[Size - window size : Size];
    Estimated demand ← Load model.predict(Windowed readings);
    Battery power ← Sense battery charging();
    Conventional power ← Estimated demand - Estimated solar power - Battery power;
    User List ← Get all users in the grid();
    Demand List ← Sense demand when switching on(User List);
    Sort(Demand List, Arrival time, Power demand);
    foreach demand in Demand List do
        if demand.max ≤ Solar power then
            Solar power -= demand.max;
            demand.assign(Solar line);
        end
        else if demand.max ≤ Conventional power then
            Conventional power -= demand.max;
            demand.assign(Conventional line);
        end
        else if demand.max ≤ Battery power then
            Battery power -= demand.max;
            demand.assign(Battery line);
        end
        else
            pass;
            // Implement Load shedding OR Connect to battery backup line
        end
    end
    Clear demand list();
    Turn off assigned demands after(burst time);
    Actual demand ← Sense line current() * Sense voltage drop();
    View Actual vs Predicted Load on Dashboard();
end
```

3. **Preprocessing Module:** Cleaning and normalization of the aggregated data; applying a rolling window.
4. **Machine Learning Module:** Fitting an LSTM model to the dataset, tuning its hyperparameters, validating its accuracy using the MSE metric, saving the model.
5. **Power Scheduling Module:** Switches the power sources' line connections at the users' end so each user gets connected to either the solar or conventional source.
6. **Dashboard Module:** Displays actual vs estmated power generation.

5.1.2 Power Scheduling Algorithm

Power scheduling is performed by regulating the switches on the user line (busbar) as discussed in Algorithm 1. Each user is assigned to one of the power sources - solar, battery backup, or conventional power source, depending on the power available with each source. The following are three operating cases, considered in the scheduling algorithm.

1. **When Demand $<$ Renewable Generation**: The system allots renewable power to all users and extra power is used for charging backups.

2. **Renewable Generation $<$ Power Demand $<$ Power Plant Capacity**: Conventional Power = Estimated demand − Renewable Generation (output is controlled as per the load demand estimates). Uncertain loads can be handled using power backups/batteries.

3. **Power Plant Capacity $<$ Power Demand**: Load shedding is carried out in the intended user's power line.

5.2 Tools and Technologies Used

5.2.1 Tools

- **Illumination Sensor**: For capturing sunlight readings.
- **Temperature and Humidity Sensor**: For capturing weather data.
- **Voltage and Current Sensors**: For capturing load demand readings.
- **Relays**: For managing power flow between different sources.
- **Raspberry Pi**: For aggregating sensor readings.

5.2.2 Technologies

- **AWS EC2**: For running virtual machines and processing data.
- **Apache Kafka**: For data streaming and real-time data pipelines.
- **Python**: For scripting and implementing machine learning models.
- **TensorFlow**: For training deep learning models.
- **Keras**: Framework for quick model development on top of Tensorflow.

5.3 Model Architecture

5.3.1 LSTM Overview

Long Short Term Memory is a type of Recurrent Neural Network (RNN), used for learning sequences and time series data. It resolves the vanishing gradient issue of traditional RNNs. LSTM networks have cells that can store information over long time periods, allowing them to learn and remember patterns in time series data. These cells have an internal cell state and three gates (forget, input, and output gates).

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

1. **Forget Gate** (f_t): Enables an LSTM cell to selectively forget a certain amount of previously stored information

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

2. **Input Gate** (i_t): Enables an LSTM cell to selectively add new information at each time step.

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$

3. **Candidate Cell State** ($\tilde{C}_t$): It is the proposed update to the cell state at a given time step, calculated using the current input and the previous hidden state (output).

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t$$

4. **Cell State** (C_t): Represents the long-term memory of the LSTM cell, allowing it to retain information over long sequences.

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$

5. **Output Gate** (o_t): Determines what should be the output of the LSTM cell at any given time step.

$$h_t = o_t \cdot \tanh(C_t)$$

6. **Hidden State** (h_t): Captures the information learnt by an LSTM cell at a specific time step, as its current output.

- $\tilde{C}_t, C_t, h_t$ are the candidate, cell, and hidden states
- x_t is the incoming input at time step t
- W_f, W_i, W_C, W_o are weight matrices of the respective gates
- b_f, b_i, b_C, b_o are bias vectors used by the gates
- $\tanh$, σ are the hyperbolic tan, sigmoid activation functions
- $\tanh(z) = \dfrac{e^z - e^{-z}}{e^z + e^{-z}}$ and $\sigma(z) = \dfrac{1}{1 + e^{-z}}$

5.3.2 Data Preprocessing

The load demand data for compiling the model architecture was procured from the Dataset-of-HRP-38-test-system by Z. Zhou [12]. The load demand data was present in the form of hourly reading columns, which were flattened into a 1-D array of 8760 readings, followed by using a rolling window approach to generate a supervised learning dataset for LSTM.

$$[1,2,3,4,5] \Rightarrow [6]$$
$$[2,3,4,5,6] \Rightarrow [7]$$
$$[3,4,5,6,7] \Rightarrow [8]$$

Eg: Time series data = [1,2,3,4,5,6,7] and window size = 5. Five records from the data become a record of input data to be passed and 6^{th} record becomes the output of the model. The last record, 8, wasn't present in the dataset. Likewise, trained LSTM models can predict next data points using historical readings. The dataset is then reshaped into a 3D matrix of shape (inputs, time-steps, features) as shown below.

$$[[[1]\,,[2]\,,[3]\,,[4]\,,[5]]]$$

5.3.3 Data Normalization

Normalization is a technique used to scale up/down data to the range of [0,1]. Normalization brings all features on a comparable scale and makes it easier for models to train on the data and generalize better.

$$X_{\text{norm}} = \frac{X - X_{\text{min}}}{X_{\text{max}} - X_{\text{min}}}$$

5.3.4 Splitting of the Dataset

The windowed and normalized data was divided into three disjoint sets - 80%, 10%, 10%, sequentially - **Training set**: used to train the model, **Validation set**: used to analyze the model's performance on unseen data while training, **Testing set**: used to evaluate the model's performance. The model is fine-tuned by the optimizer - **Adam** in our case, at every iteration, such that the validation error is minimized.

5.3.5 Hyperparameter Tuning and Model Evaluation

The following hyperparameters were tweaked and the models were evaluated on the testing set using the MSE metric.

1. No. of LSTM units (using a single LSTM layer).

2. Dropout: percentage of random connections to be dropped in each layer (to avoid overfitting).

Units	20	30	40	50	60	70	80	84
MSE	7.72	3.43	3.21	2.91	2.06	1.91	1.99	1.92

Dropout	0.1	0.2	0.3	0.4
MSE	1.91	4.19	3.43	4.33

3. Batch size: No. of data points after which the gradients are backpropagated.

Batch Size	16	32	48	60
MSE	1.81	1.861	2.15	1.97

4. Window size: The size of the rolling window.

Window Size	24	48	60	84
MSE	1.81	2.48	1.63	2.63

5. Layers: No. of hidden layers in the network.

Layers	1	2	3
MSE	1.63	2.09	2.21

Table 5.1: Finalizing the best performing hyperparameters

Parameter	Units	Dropout	Batch Size	Window Size	Layers
Best Value	70	0.1	16	60	1

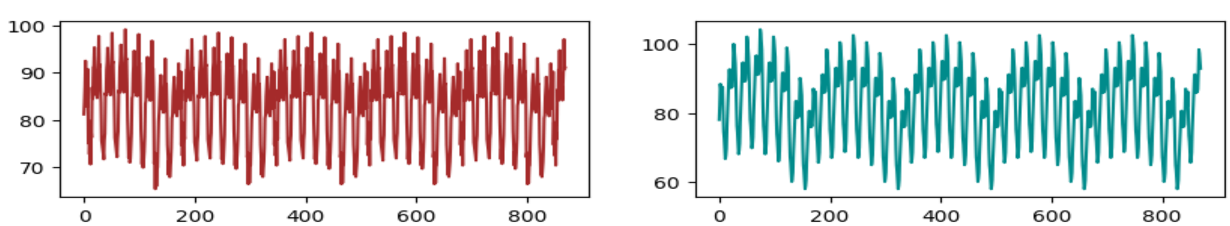


Figure 5.2: Predicted (Left) and Actual (Right) Load Demand in GW vs Time

On selecting the best value for each of the mentioned hyperparameters, the above table was obtained. After fitting 1000 epochs of a model with these parameters on the training dataset, the MSE (Mean Squared Error) obtained on the testing dataset is 0.0053.

$$\text{Mean Squared Error} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

- n: Number of data records
- y_i: Actual value of the record i
- $\hat{y}_i$: Predicted value of the record i

5.4 Load Demand Forecasting

5.4.1 Data Acquisition

The finalized hyperparameters were further used to fit an LSTM model to the data gathered from an actual fan load of 12V and maximum current rating of 1A, with a simulated behavior of changing speeds, controlled by a PWM (Pulse Width Modulation) speed regulator. The voltage dropped across the fan's ends and the line current was measured using sensors, at regular intervals, to obtain a total of 1475 readings.

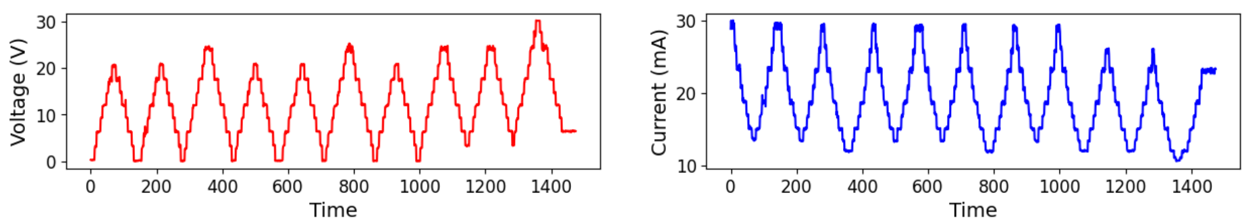


Figure 5.3: Voltage and Current sensor readings

The sensor readings were calibrated to accurately represent the true voltage and current values. The voltage and current values were then substituted in the below formula to obtain the load (power) demand readings.

$$\text{Power (W)} = \text{Voltage (V)} * \text{Current (mA)} * 0.001$$

5.4.2 Data Preprocessing

The load power readings were flattened into a 1-D series and a rolling window of size 60 was applied on to the data to transform it into a supervised learning matrix of sequences, using the approach discussed earlier. Further, the dataset was split into 80%, 10%, 10%, as the training, validation and testing sets followed by normalization.

5.4.3 Model Fitting and Evaluation

The best-performing hyperparameters listed in Table I were used to fit an LSTM model to the dataset, using a learning rate of 0.0001. The MSE of the trained model on the testing data was 0.0013. Hence, the model was successful in providing reliable estimates for the load demand pattern.

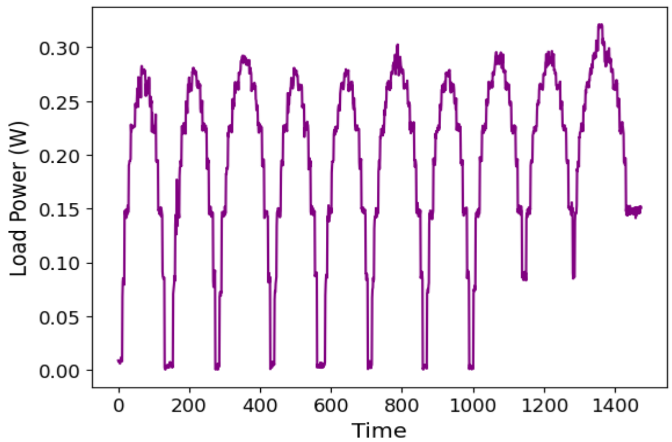


Figure 5.4: Load Demand readings

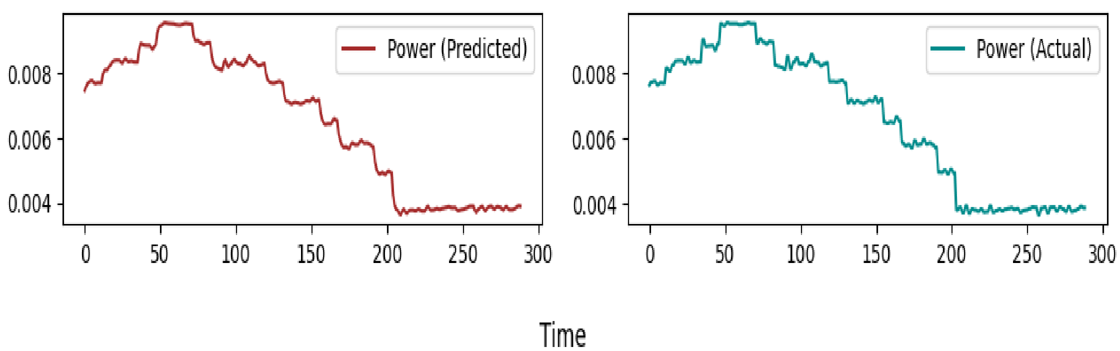


Figure 5.5: Load Power (Predicted vs Actual) (W)

5.5 Analysis of Solar Data

5.5.1 Data Acquisition

The modeling of load behavior was followed by the analysis of solar power generation data. A 12V, 10W solar panel was connected to a dummy motor load (12 V, 1.5A maximum current rating). The voltage generated across the ends of the panel, the line current in the circuit, illumination, temperature, and humidity at the site were measured using sensors, at regular intervals. A total of 3892 readings were collected. The aim of the data collection was to use current (which depends on the load) as the target feature, which can be predicted using the remaining features. Hence, given the - panel voltage, illumination, temperature, humidity - at any instant of the day,

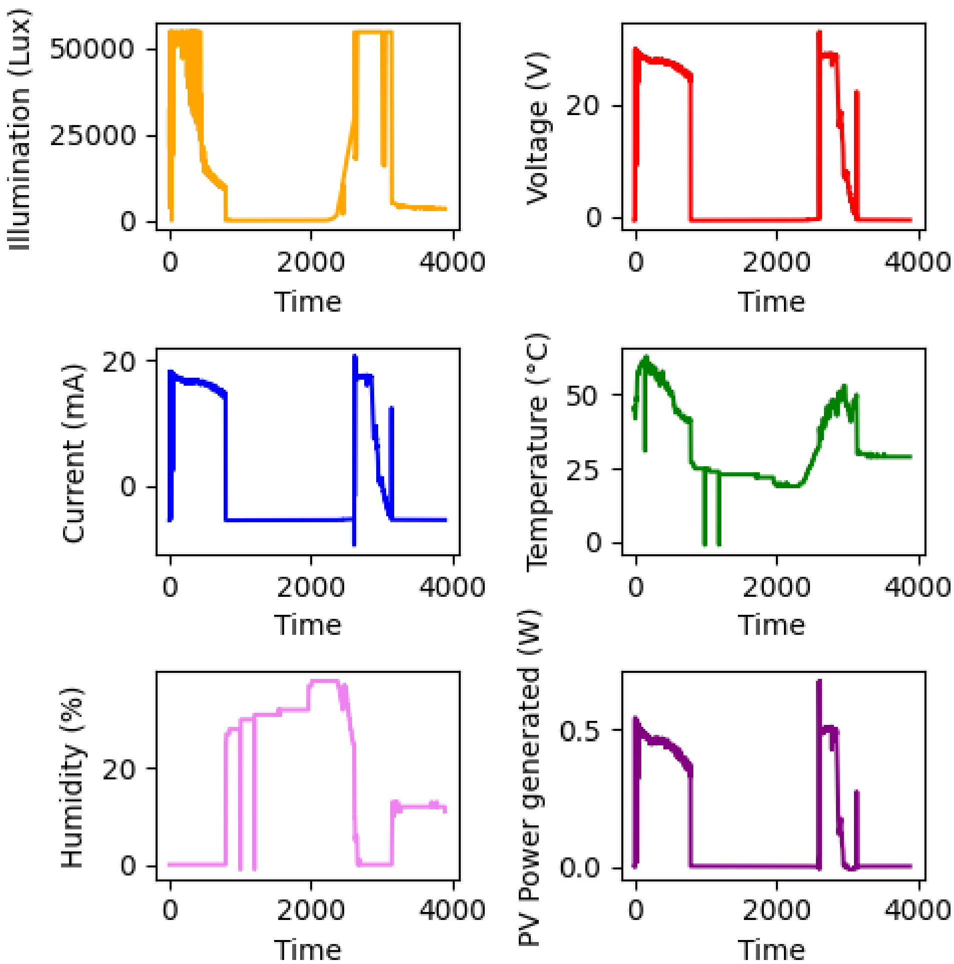


Figure 5.6: Solar Power Generation readings

the line current can be predicted (even when no load is connected to the panel) and multiplied with the voltage to obtain the estimated solar power generation.

5.5.2 Data Preprocessing

The dataset was cleaned and split into 80%, 10%, and 10%, as the training, validation, and testing sets and normalized.

5.5.3 Model Training and Evaluation

A 2-layered MLP (Multi-Layer Perceptron) was trained on the dataset with the line current as the target feature. The layers contained 10 dense units each and ReLU (Rectified Linear Unit) activation function was applied at each layer. A learning rate of 0.0001 with the Adam optimizer were used to fit 100 epochs on the dataset.

$$\mathrm{ReLU}(z) = \max(0,z)$$

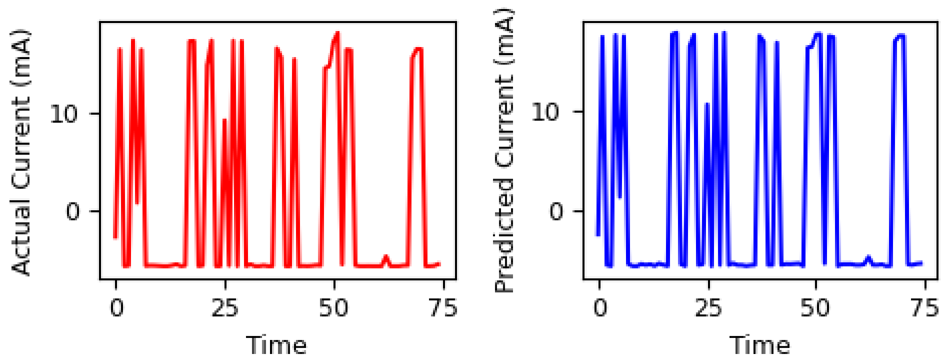


Figure 5.7: Solar Line Current prediction

The trained model had an MSE of 0.31 on the testing set.

Chapter 6

Software Testing

6.1 Types of Testing

- **Unit Testing**: Testing individual components or functions to ensure they work correctly in isolation.
- **Integration Testing**: Verifying that different modules or services used by the application work well together.
- **System Testing**: Testing the complete and integrated software to evaluate the system's compliance with the specified requirements.
- **Performance Testing**: Assessing the system's performance under different conditions, including load testing and stress testing.
- **Validation Testing**: Ensuring that the final product meets the requirements and performs its intended functions correctly.
- **Regression Testing**: Ensuring that new code changes do not wrongly affect the existing functionality of the product.

6.2 Test Cases

- **Test Case 1: Sensor Data Aggregation**
  - **Objective**: Verify that the Raspberry Pi correctly aggregates sensor data.
  - **Inputs**: Simulated data from illumination, temperature, humidity, voltage, and current sensors.
  - **Expected Output**: Correctly formatted and timestamped data packets sent to the cloud.
  - **Result**: Pass

- **Test Case 2: Data Streaming with Apache Kafka**
  - **Objective**: Ensure real-time data streaming and pipeline functionality.
  - **Inputs**: Continuous data stream from sensors.
  - **Expected Output**: Real-time data availability at the consumer end.
  - **Result**: Pass
- **Test Case 3: LSTM Model Training**
  - **Objective**: Validate the LSTM model's ability to learn and predict load demand.
  - **Inputs**: Training dataset with load demand readings.
  - **Expected Output**: Model achieves an MSE below a predefined threshold on the test dataset.
  - **Result**: MSE = 0.0053 (Pass)
- **Test Case 4: Real-Time Load Demand Prediction**
  - **Objective**: Verify real-time prediction accuracy of the trained LSTM model.
  - **Inputs**: Live sensor data from a simulated load.
  - **Expected Output**: Predicted values closely match actual load demand.
  - **Result**: MSE = 0.0013 (Pass)
- **Test Case 5: Solar Power Generation Prediction**
  - **Objective**: Validate the MLP model's ability to predict solar power generation.
  - **Inputs**: Data from solar panel sensors (voltage, illumination, temperature, humidity).
  - **Expected Output**: Accurate prediction of line current.
  - **Result**: MSE = 0.31 (Pass)
- **Test Case 6: Power Scheduling Algorithm**
  - **Objective**: Ensure the naive power scheduling algorithm allocates power correctly based on demand and availability.
  - **Inputs**: Predicted load demand and solar power generation data.
  - **Expected Output**: Correct power allocation from solar and conventional sources.
  - **Result**: Pass
- **Test Case 7: System Performance under Load**
  - **Objective**: Test the system's performance under maximum expected load.
  - **Inputs**: Simulated peak load conditions.
  - **Expected Output**: Stable system performance without crashes or significant delays.
  - **Result**: Pass

Chapter 7

Result

7.1 Outcomes

The LSTM model trained on load demand data from the Dataset-of-HRP-38-test-system by Zhenyu Zhou [12] had a mean squared error of 0.0053 on the testing set, which comprised 10% of the readings from the entire dataset. The hyperparameters finalized from this training were used to train another model on data collected from a simulated fan load. The trained model resulted in an MSE of 0.0013 on the testing set.

7.2 Screenshots

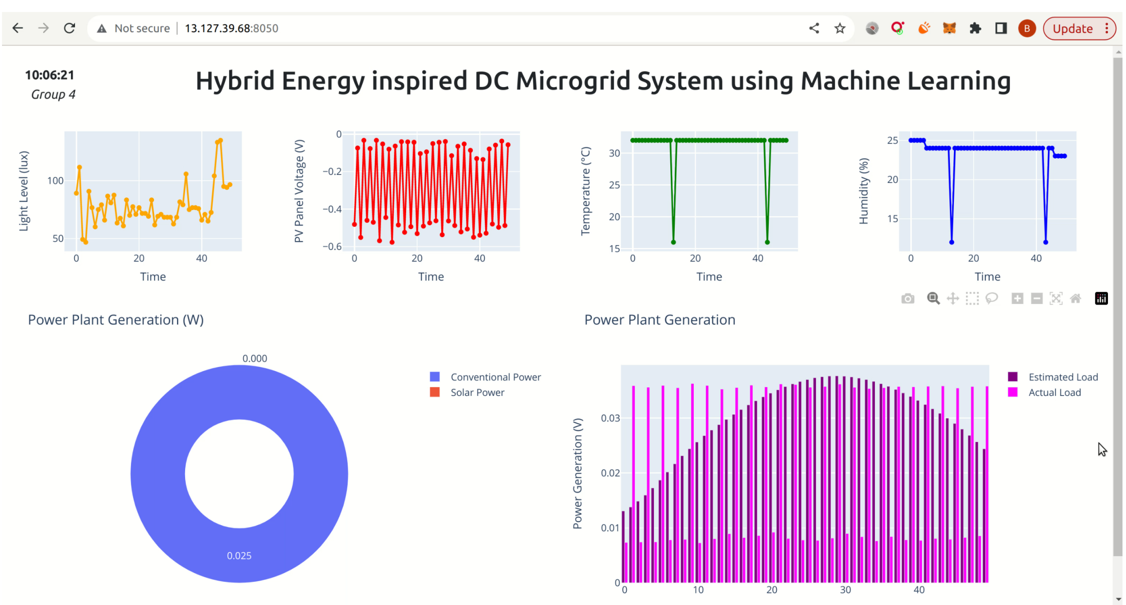


Figure 7.1: Dashboard Module

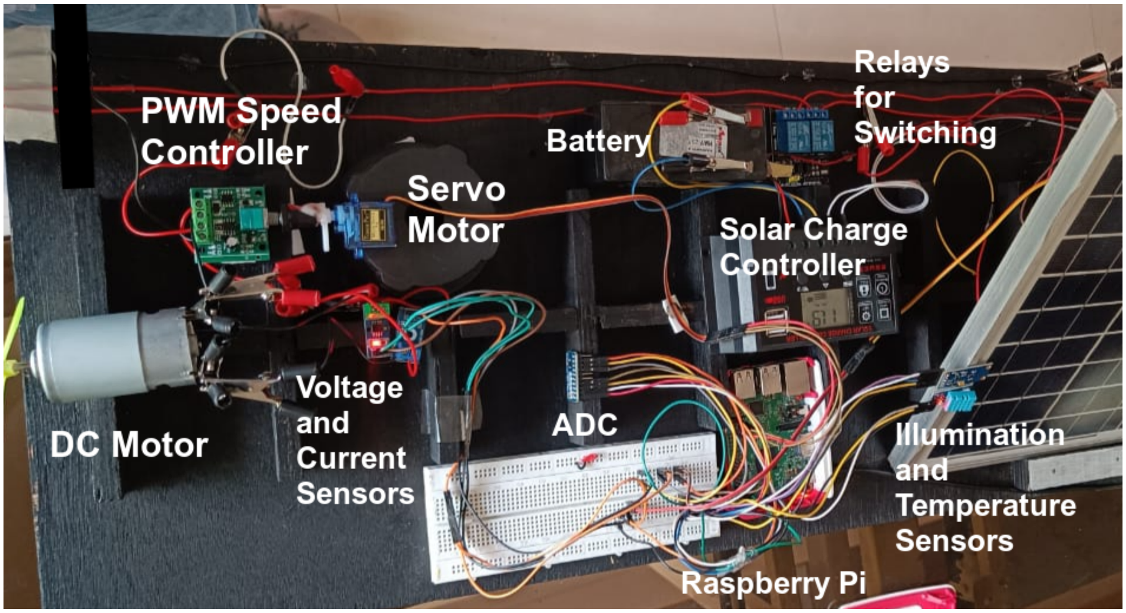


Figure 7.2: Sensors and Actuators

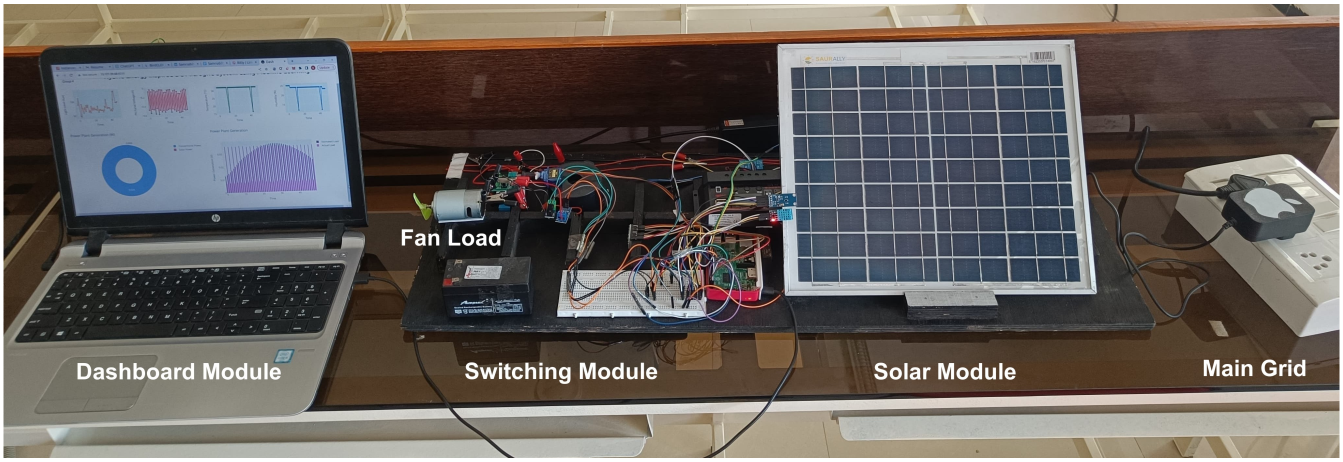


Figure 7.3: Hybrid DC Micro-grid System

Chapter 8

Conclusion

8.1 Conclusion

A hybrid energy management system for DC micro-grids is proposed, having a full fledged hybrid energy management system. Our aim was to predict the power output of various energy resources, to design a system, to predict as well as sense real time load demand and to have a fully working real time dashboard which will show the current readings as well as the process by which the energy sources will be switched and regulated. Hence, the domain of energy demand forecasting was explored, and an LSTM model was trained on the collected load demand dataset to estimate the power demand of the users in the future. Another neural network was trained to estimate the solar power generated at any time instant using data collected from an actual solar panel. A prototype for a Hybrid DC Microgrid was implemented, and power scheduling was done using a naive approach that makes use of the estimated load demand and solar power to regulate the conventional power. The users are allotted power from either of the sources, solar or conventional, depending on the power availability of each.

8.2 Future Work

We are currently limiting our scope to building a wired microgrid setup. If feasible, we will work on building a wireless microgrid and study more on securing this wireless smart grid. We will try to execute data manipulation attacks on the wireless communication going on between the transmitter and the receiver and suggest possible security measures in opposition to the attacks. Also, we will try to integrate a steam generator and show the controlling of the thermal output using burner flame controlling knobs.

8.3 Advantages

- Aggregating various distributed energy sources together.
- Reducing the average cost of generation of electricity per unit.
- Ease the work of electricity board users, by automating the communication process.

8.4 Applications

- Control distributed hybrid energy systems for integration of generated electricity.
- Predicting time series behavioral data of users to obtain their estimated power requirement, which can be applied to sectors other than power and energy too.
- Optimize the operation of smart homes and buildings by integrating demand prediction with automated energy management systems.
- Enable the development of smart cities with intelligent energy distribution grid networks that can adapt in real-time to changing energy needs.
- Support agricultural applications by managing the power supply for irrigation systems, crop monitoring devices, and other farming tools and technologies.

Appendix A

Computational Complexity

A.1 LSTM Model

The computational complexity of an LSTM model is determined by the operations performed at each time step. Each gate in the LSTM (forget, input, candidate, and output) involves matrix multiplications and element-wise operations. The complexity per gate can be expressed as:

$$O(n(n+m))$$

where:

- n = the total number of LSTM units.
- m = the total number of input features.

Since there are four gates, the total complexity per time step is still $O(n(n+m))$. For T time steps, the total complexity is:

$$O(T \cdot n(n+m))$$

A.2 Naive Scheduling Algorithm

The naive scheduling algorithm consists of two main steps:

1. Forecasting Demand: This step uses the LSTM model.
2. Allocating Power: This step iterates over the demand points and allocates power from renewable and conventional sources.

The complexity for forecasting demand using the LSTM model is approximately:

$$O(1)$$

For allocating power, if d is the number of demand points and r is the number of power sources, the complexity is:

$$O(d \cdot r)$$

In our case, r is small, so this is simplified to:

$$O(d)$$

The overall complexity of the naive scheduling algorithm is dominated by the no. of demand points / users in the grid, thus it is:

$$O(1 + d) \approxeq O(d)$$

A.2.1 Complexity Class

Both algorithms fall into the polynomial time complexity class (P), indicating they are efficient and scalable for practical use.

Appendix B

Published /Presented Paper , Certificates of Participation/Prize winning

K. K. WAGH EDUCATION SOCIETY'S

K. K. WAGH INSTITUTE OF ENGINEERING EDUCATION AND REASEARCH, NASHIK

DEPARTMENT OF COMPUTER, AI&DS AND CSD ENGINEERING

EQUINOX 2024

CERTIFICATE

This certificate is awarded to that Mr./Miss. Samrudhi Khairnar of Computer Engg. department of MVP's KBT college of Engg. college for securing 1st/ 2nd / 3rd/prize / participation / worked as Committee Member Project Competition in held in **EQUINOX 2024** at K. K. Wagh Institute of Engineering Education & Research, Nashik on 4th & 5th April, 2024.

Faculty Co-ordinator
Debuggers' / CSI Students' Chapter/ Phoenix / DeSoc Club

Prof.Dr.S.S.Sane
Head Of Department
Dept. of Comp. Engg. CSD & AI&DS

Dr. K.N.Nandurkar
Principal
KKWIEER, Nashik


K. K. WAGH EDUCATION SOCIETY'S

K. K. WAGH INSTITUTE OF ENGINEERING EDUCATION AND REASEARCH, NASHIK

DEPARTMENT OF COMPUTER, AI&DS AND CSD ENGINEERING

EQUINOX 2024

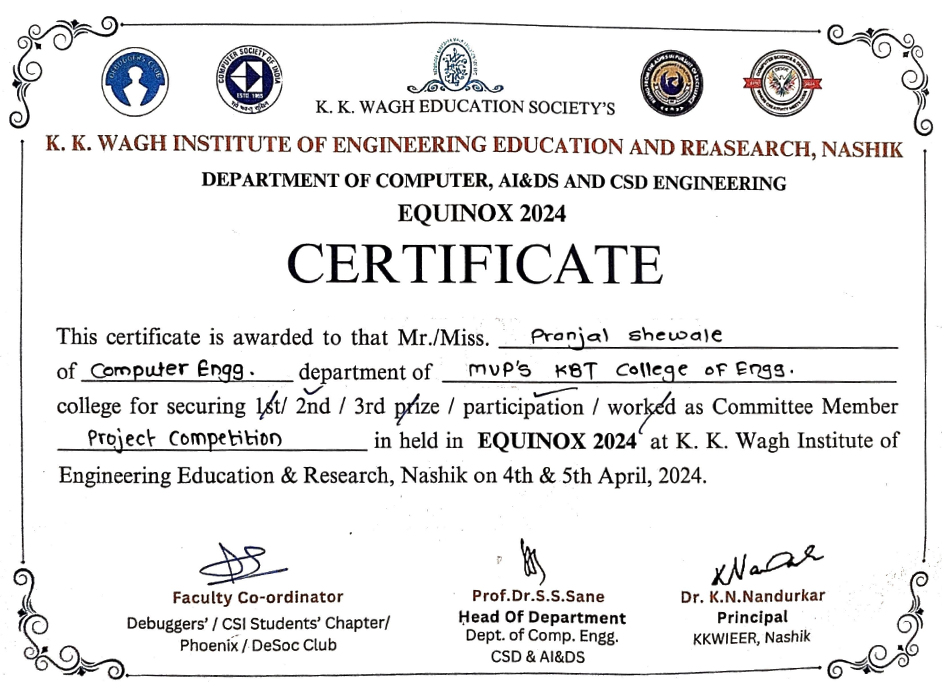
CERTIFICATE

This certificate is awarded to that Mr./Miss. Sakshi Jadhav of Computer Engg. department of MVP's KBT College of Engg. college for securing 1st/ 2nd / 3rd prize / participation / worked as Committee Member Project Competition in held in **EQUINOX 2024** at K. K. Wagh Institute of Engineering Education & Research, Nashik on 4th & 5th April, 2024.

Faculty Co-ordinator
Debuggers' / CSI Students' Chapter/ Phoenix / DeSoc Club

Prof.Dr.S.S.Sane
Head Of Department
Dept. of Comp. Engg.
CSD & AI&DS

Dr. K.N.Nandurkar
Principal
KKWIEER, Nashik


K. K. WAGH EDUCATION SOCIETY'S

K. K. WAGH INSTITUTE OF ENGINEERING EDUCATION AND REASEARCH, NASHIK

DEPARTMENT OF COMPUTER, AI&DS AND CSD ENGINEERING

EQUINOX 2024

CERTIFICATE

This certificate is awarded to that Mr./Miss. Pranjal Shewale of Computer Engg. department of MVP's KBT College of Engg. college for securing 1st/ 2nd / 3rd prize / participation / worked as Committee Member Project Competition in held in **EQUINOX 2024** at K. K. Wagh Institute of Engineering Education & Research, Nashik on 4th & 5th April, 2024.

Faculty Co-ordinator
Debuggers' / CSI Students' Chapter/ Phoenix / DeSoc Club

Prof.Dr.S.S.Sane
Head Of Department
Dept. of Comp. Engg.
CSD & AI&DS

Dr. K.N.Nandurkar
Principal
KKWIEER, Nashik

K. K. WAGH EDUCATION SOCIETY'S

K. K. WAGH INSTITUTE OF ENGINEERING EDUCATION AND REASEARCH, NASHIK

DEPARTMENT OF COMPUTER, AI&DS AND CSD ENGINEERING

EQUINOX 2024

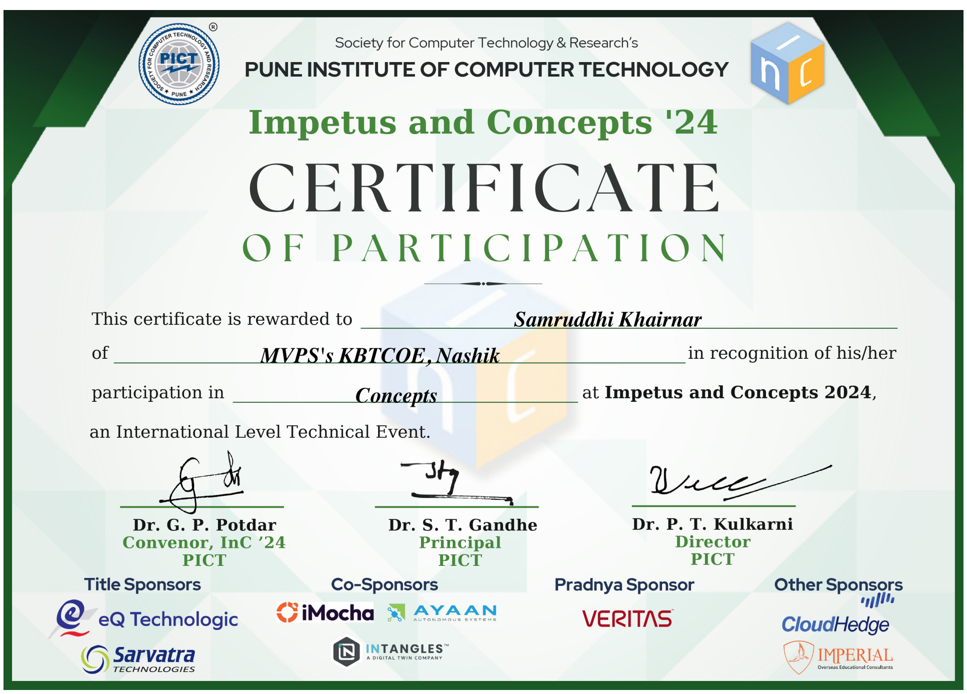
CERTIFICATE

This certificate is awarded to that Mr./Miss. Kunika Namaware of Computer Engg. department of MVP's KBT College of Engg. college for securing 1st / 2nd / 3rd prize / participation / worked as Committee Member Project Competition in held in **EQUINOX 2024** at K. K. Wagh Institute of Engineering Education & Research, Nashik on 4th & 5th April, 2024.

Faculty Co-ordinator
Debuggers' / CSI Students' Chapter/ Phoenix / DeSoc Club

Prof.Dr.S.S.Sane
Head Of Department
Dept. of Comp. Engg.
CSD & AI&DS

Dr. K.N.Nandurkar
Principal
KKWIEER, Nashik



Society for Computer Technology & Research's

PUNE INSTITUTE OF COMPUTER TECHNOLOGY

Impetus and Concepts '24

CERTIFICATE

OF PARTICIPATION

This certificate is rewarded to *Samruddhi Khairnar* of *MVPS's KBTCOE, Nashik* in recognition of his/her participation in *Concepts* at **Impetus and Concepts 2024**, an International Level Technical Event.

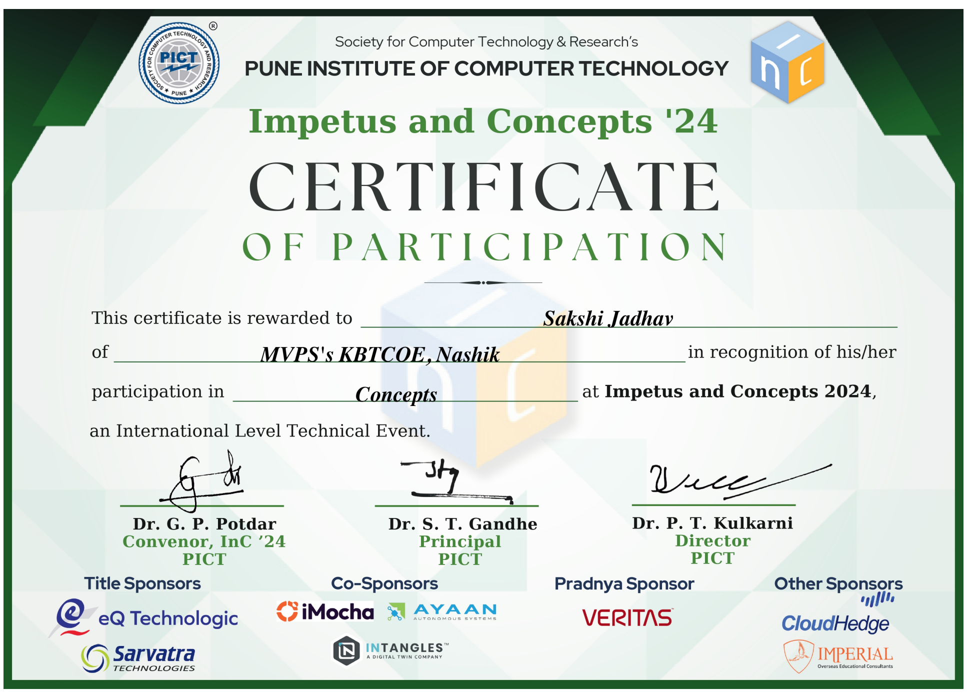
Dr. G. P. Potdar
Convenor, InC '24
PICT

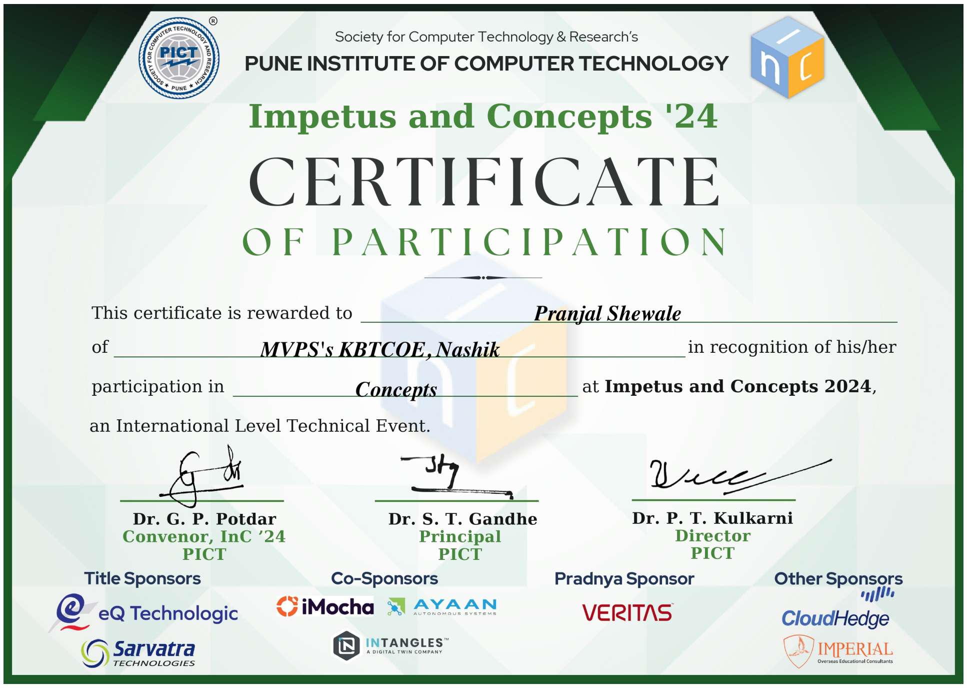
Dr. S. T. Gandhe
Principal
PICT

Dr. P. T. Kulkarni
Director
PICT

Title Sponsors: eQ Technologic, Sarvatra Technologies
Co-Sponsors: iMocha, AYAAN, INTANGLES
Pradnya Sponsor: VERITAS
Other Sponsors: CloudHedge, IMPERIAL


Society for Computer Technology & Research's
PUNE INSTITUTE OF COMPUTER TECHNOLOGY
Impetus and Concepts '24
CERTIFICATE
OF PARTICIPATION
This certificate is rewarded to Kunika Narnaware
of MVPS's KBTCOE, Nashik in recognition of his/her
participation in Concepts at Impetus and Concepts 2024,
an International Level Technical Event.
Dr. G. P. Potdar
Convenor, InC '24
PICT
Dr. S. T. Gandhe
Principal
PICT
Dr. P. T. Kulkarni
Director
PICT
Title Sponsors
eQ Technologic
Sarvatra
Co-Sponsors
iMocha
AYAAN
INTANGLES
Pradnya Sponsor
VERITAS
Other Sponsors
CloudHedge
IMPERIAL


Society for Computer Technology & Research's
PUNE INSTITUTE OF COMPUTER TECHNOLOGY
Impetus and Concepts '24
CERTIFICATE
OF PARTICIPATION
This certificate is rewarded to Sakshi Jadhav
of MVPS's KBTCOE, Nashik in recognition of his/her
participation in Concepts at Impetus and Concepts 2024,
an International Level Technical Event.
Dr. G. P. Potdar
Convenor, InC '24
PICT
Dr. S. T. Gandhe
Principal
PICT
Dr. P. T. Kulkarni
Director
PICT
Title Sponsors
eQ Technologic
Sarvatra
Co-Sponsors
iMocha
AYAAN
INTANGLES
Pradnya Sponsor
VERITAS
Other Sponsors
CloudHedge
IMPERIAL


Society for Computer Technology & Research's
PUNE INSTITUTE OF COMPUTER TECHNOLOGY

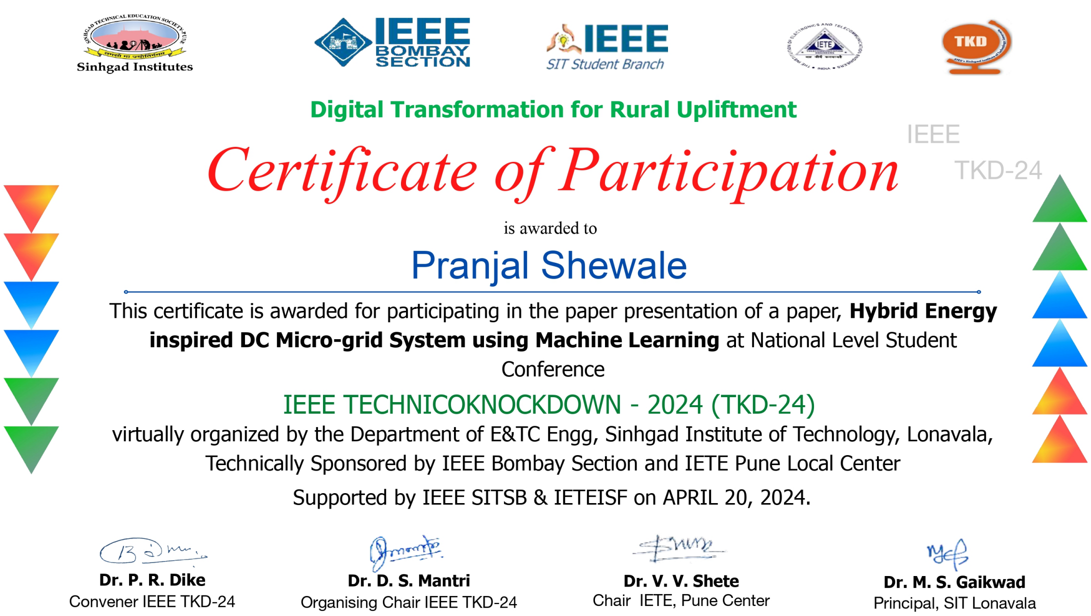
Impetus and Concepts '24

CERTIFICATE
OF PARTICIPATION

This certificate is rewarded to *Pranjal Shewale*
of *MVPS's KBTCOE, Nashik* in recognition of his/her
participation in *Concepts* at **Impetus and Concepts 2024**,
an International Level Technical Event.

Dr. G. P. Potdar Convenor, InC '24 PICT	Dr. S. T. Gandhe Principal PICT	Dr. P. T. Kulkarni Director PICT

Title Sponsors: eQ Technologic, Sarvatra Technologies
Co-Sponsors: iMocha, AYAAN, INTANGLES
Pradnya Sponsor: VERITAS
Other Sponsors: CloudHedge, IMPERIAL


Sinhgad Institutes
IEEE Bombay Section
IEEE SIT Student Branch
TKD

Digital Transformation for Rural Upliftment

IEEE
TKD-24

Certificate of Participation

is awarded to

Pranjal Shewale

This certificate is awarded for participating in the paper presentation of a paper, **Hybrid Energy inspired DC Micro-grid System using Machine Learning** at National Level Student Conference

IEEE TECHNICOKNOCKDOWN - 2024 (TKD-24)

virtually organized by the Department of E&TC Engg, Sinhgad Institute of Technology, Lonavala, Technically Sponsored by IEEE Bombay Section and IETE Pune Local Center

Supported by IEEE SITSB & IETEISF on APRIL 20, 2024.

Dr. P. R. Dike Convener IEEE TKD-24	**Dr. D. S. Mantri** Organising Chair IEEE TKD-24	**Dr. V. V. Shete** Chair IETE, Pune Center	**Dr. M. S. Gaikwad** Principal, SIT Lonavala

Sinhgad Institutes

IEEE BOMBAY SECTION

IEEE SIT Student Branch

TKD

Digital Transformation for Rural Upliftment

IEEE TKD-24

Certificate of Participation

is awarded to

Sakshi Jadhav

This certificate is awarded for participating in the paper presentation of a paper, **Hybrid Energy inspired DC Micro-grid System using Machine Learning** at National Level Student Conference

IEEE TECHNICOKNOCKDOWN - 2024 (TKD-24)

virtually organized by the Department of E&TC Engg, Sinhgad Institute of Technology, Lonavala, Technically Sponsored by IEEE Bombay Section and IETE Pune Local Center

Supported by IEEE SITSB & IETEISF on APRIL 20, 2024.

Dr. P. R. Dike	**Dr. D. S. Mantri**	**Dr. V. V. Shete**	**Dr. M. S. Gaikwad**
Convener IEEE TKD-24	Organising Chair IEEE TKD-24	Chair IETE, Pune Center	Principal, SIT Lonavala

Sinhgad Institutes

IEEE BOMBAY SECTION

IEEE SIT Student Branch

TKD

Digital Transformation for Rural Upliftment

IEEE TKD-24

Certificate of Participation

is awarded to

Samruddhi Khairnar

This certificate is awarded for participating in the paper presentation of a paper, **Hybrid Energy inspired DC Micro-grid System using Machine Learning** at National Level Student Conference

IEEE TECHNICOKNOCKDOWN - 2024 (TKD-24)

virtually organized by the Department of E&TC Engg, Sinhgad Institute of Technology, Lonavala, Technically Sponsored by IEEE Bombay Section and IETE Pune Local Center

Supported by IEEE SITSB & IETEISF on APRIL 20, 2024.

Dr. P. R. Dike	**Dr. D. S. Mantri**	**Dr. V. V. Shete**	**Dr. M. S. Gaikwad**
Convener IEEE TKD-24	Organising Chair IEEE TKD-24	Chair IETE, Pune Center	Principal, SIT Lonavala

Sinhgad Institutes

IEEE BOMBAY SECTION

IEEE SIT Student Branch

Digital Transformation for Rural Upliftment

IEEE TKD-24

Certificate of Participation

is awarded to

Kunika Narnaware

This certificate is awarded for participating in the paper presentation of a paper, **Hybrid Energy inspired DC Micro-grid System using Machine Learning** at National Level Student Conference

IEEE TECHNICOKNOCKDOWN - 2024 (TKD-24)

virtually organized by the Department of E&TC Engg, Sinhgad Institute of Technology, Lonavala, Technically Sponsored by IEEE Bombay Section and IETE Pune Local Center

Supported by IEEE SITSB & IETEISF on APRIL 20, 2024.

Dr. P. R. Dike	**Dr. D. S. Mantri**	**Dr. V. V. Shete**	**Dr. M. S. Gaikwad**
Convener IEEE TKD-24	Organising Chair IEEE TKD-24	Chair IETE, Pune Center	Principal, SIT Lonavala

Maratha Vidya Prasarak Samaj's
Karamveer Adv Baburao Ganpatrao Thakare
College of Engineering, Nashik

Permanently Affiliated to Savitribai Phule Pune University,
Pune and Approved by AICTE, New Delhi

KBTCOE

NAAC A++

CERTIFICATE

KBTCOE TECH FEST 2K24

25TH ANNIVERSARY CELEBRATION

This certificate is proudly presented to Mr/Miss Sakshi Jadhav of College MVP's KBT college of Engineering who has Participated / Co-ordinated / Secured Runner up in the event InnoTech Expo of KBTCOE TECH-FEST 2K24

Prof. S. M. Waysal	Dr. V. M. Birari	Dr. S. R. Devane
Tech Fest 2K24 Co-ordinator	Vice Principal	Principal

OUR SPONSORS

POLAAD, MAAC, CONVERSA, CADD CENTRE NASHIK, ANAND GROUP BUILDERS & DEVELOPERS, YOGESHWAR, KRANTI, AKSHAY, SHIVENDU, MOTWANE, NEBULA

Maratha Vidya Prasarak Samaj's
Karamveer Adv Baburao Ganpatrao Thakare
College of Engineering, Nashik

Permanently Affiliated to Savitribai Phule Pune University, Pune and Approved by AICTE, New Delhi

KBTCOE

CERTIFICATE

KBTCOE TECH FEST 2K24

This certificate is proudly presented to Mr/Miss Kunika Narnaware
of College MVP's KBT College of Engineering who has Participated / Co-ordinated / Secured Runner up in the event INNOVTECH EXPO
of KBTCOE TECH-FEST 2K24

Prof. S. M. Waysal
Tech Fest 2K24 Co-ordinator

Dr. V. M. Birari
Vice Principal

Dr. S. R. Devane
Principal

OUR SPONSORS

Maratha Vidya Prasarak Samaj's
Karamveer Adv Baburao Ganpatrao Thakare
College of Engineering, Nashik

Permanently Affiliated to Savitribai Phule Pune University, Pune and Approved by AICTE, New Delhi

KBTCOE

CERTIFICATE

KBTCOE TECH FEST 2K24

This certificate is proudly presented to Mr/Miss Pranjal Shewale
of College MVP's KBT college of Engineering who has Participated / Co-ordinated / Secured Runner up in the event INNOVTECH EXPO
of KBTCOE TECH-FEST 2K24

Prof. S. M. Waysal
Tech Fest 2K24 Co-ordinator

Dr. V. M. Birari
Vice Principal

Dr. S. R. Devane
Principal

OUR SPONSORS

Maratha Vidya Prasarak Samaj's
Karamveer Adv Baburao Ganpatrao Thakare
College of Engineering, Nashik

KBTCOE

Permanently Affiliated to Savitribai Phule Pune University,
Pune and Approved by AICTE, New Delhi

CERTIFICATE

KBTCOE TECH FEST 2K24

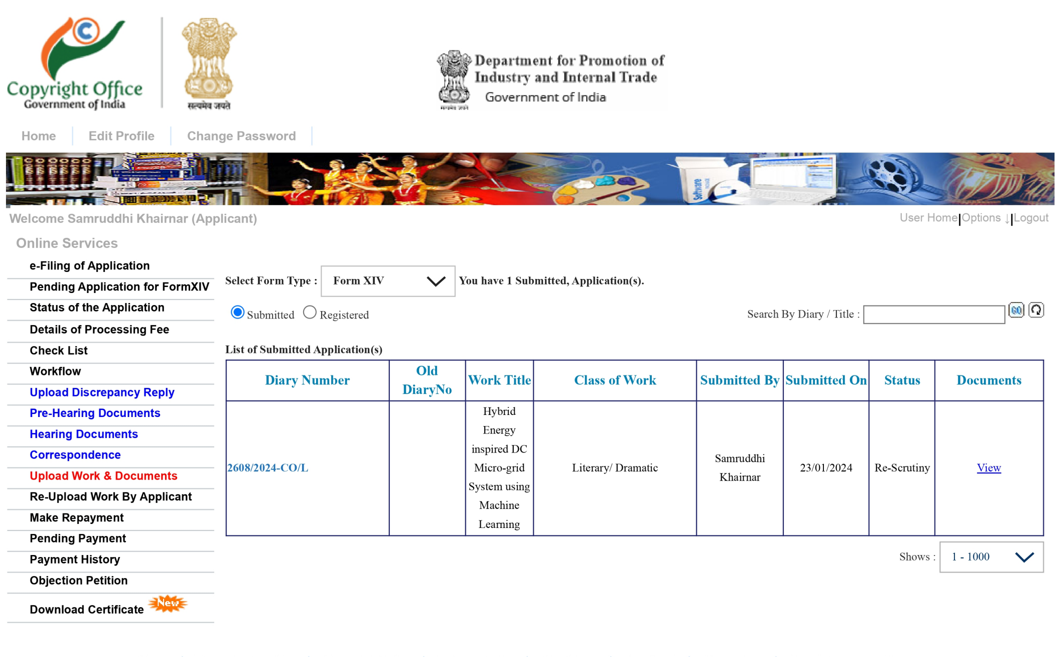
This certificate is proudly presented to Mr/Miss Samruddhi Khairnar
of College MVP's KBT College of Engineering who has Participated / Co-ordinated / Secured Runner Up in the event INNOTECH EXPO of KBTCOE TECH-FEST 2K24

Prof. S. M. Waysal
Tech Fest 2K24 Co-ordinator

Dr. V. M. Birari
Vice Principal

Dr. S. R. Devane
Principal

OUR SPONSORS



Copyright Office
Government of India

Department for Promotion of Industry and Internal Trade
Government of India

Home | Edit Profile | Change Password

Welcome Samruddhi Khairnar (Applicant)

User Home | Options | Logout

Online Services

- e-Filing of Application
- Pending Application for FormXIV
- Status of the Application
- Details of Processing Fee
- Check List
- Workflow
- Upload Discrepancy Reply
- Pre-Hearing Documents
- Hearing Documents
- Correspondence
- Upload Work & Documents
- Re-Upload Work By Applicant
- Make Repayment
- Pending Payment
- Payment History
- Objection Petition
- Download Certificate New

Select Form Type : Form XIV You have 1 Submitted, Application(s).

Submitted Registered Search By Diary / Title :

List of Submitted Application(s)

Diary Number	Old DiaryNo	Work Title	Class of Work	Submitted By	Submitted On	Status	Documents
2608/2024-CO/L		Hybrid Energy inspired DC Micro-grid System using Machine Learning	Literary/ Dramatic	Samruddhi Khairnar	23/01/2024	Re-Scrutiny	View

Shows : 1 - 1000

Home | Terms & Condition | Hyperlink Policy | Privacy Policy | Disclaimer | Feedback | Contact Us | Web Information Manager

BE_34_ Samruddhi Khairnar <kbtug20170@kbtcoe.org>

Congratulations! Project Selected in Top 50 Projects at NES Innovation Awards 2024

Sagar Kabra <sagark@gttfoundation.org> Wed, Mar 6, 2024 at 6:40 PM
Cc: NES Innovation Award <awards@nesconnect.org>
Bcc: kbtug20170@kbtcoe.org

Dear Participants and Guides,

Warm greetings from Team NES Innovation Awards!

Thank you for your enthusiastic participation in the 11th edition of NES Innovation Awards 2024. This year has been no exception to the outstanding quality of entries and ideas, with over 800 projects registered. Your contribution has truly made the evaluation process challenging for our jury.

We are thrilled to announce that your project has successfully secured a spot among the Top 50 Entries from the pool of 800+ registrations.
Congratulations on this remarkable achievement!
It's time to share this exciting news with your friends, family, faculties, and the head of your institute.

Please read the following details carefully just in case you don't miss anything -
• Please have a look at the **attached PDF document containing the list of Top 50 Entries** to locate your project.
• Please note **Project Unique ID** following your project details and team member/guide names are correct. If not, then please update it via Google Form.
• Once identified, kindly proceed to click on the link below and **complete the mandatory Google Form to confirm your further participation**
• **Click here to access the Google Form:**👉 https://forms.gle/oL6YwfVD1vMzjE2UA
• The **deadline for filling out the form is Monday, 11th March 2024**.

Stay tuned for upcoming announcements regarding the next series of events.

Quick Update on Upcoming Event:
• Business Plan Workshop is scheduled tentatively in the 3rd/4th week of March'24. Dates will be confirmed upon receipt of responses through Google Forms.
• Mode of attending the session (online/offline) is yet to be finalized.
• Detailed calendar of upcoming events will be shared soon.
• Ensure you obtain the necessary permissions from your college and mark your availability.

* All the details will be communicated on Tuesday, 12th March 2024, based on the responses received through the Google Form.

Communication Channels:
• To facilitate communication, a temporary WhatsApp group will be established, where only crucial updates will be shared.
• Kindly consider email and WhatsApp as the official communication channels.
• WhatsApp Group Link will be sent on your personal mobile numbers. All participants and guides are requested to join the group using the same link.

We deeply appreciate your patience, dedication, and enthusiasm throughout this process.
Congratulations once again for moving one step closer to the prestigious NES Innovation Awards 2024. A significant opportunity awaits you!

For any queries, please feel free to reach out via email at sagark@gttfoundation.org or through WhatsApp/Call at +91 9420443489.

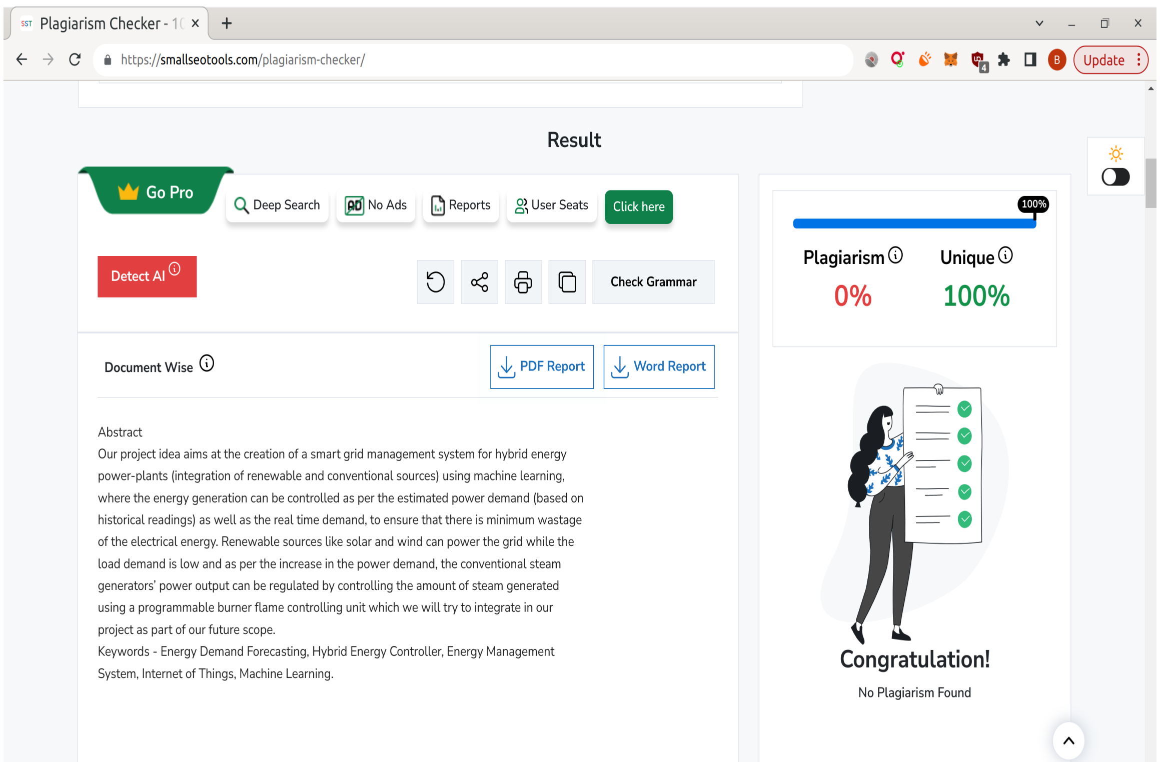
Thank you for your participation and commitment.

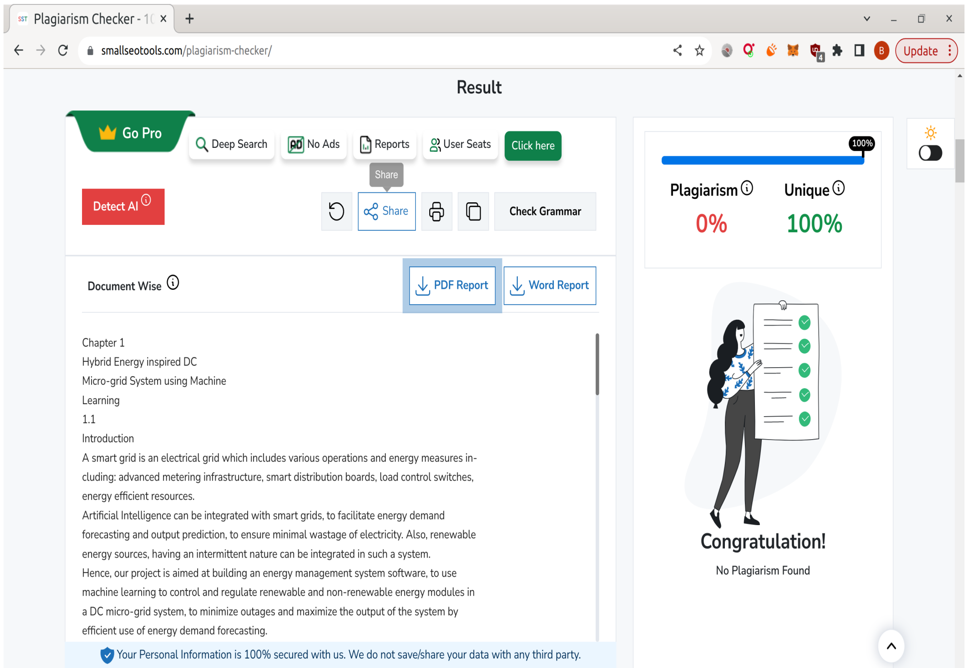
Best regards,

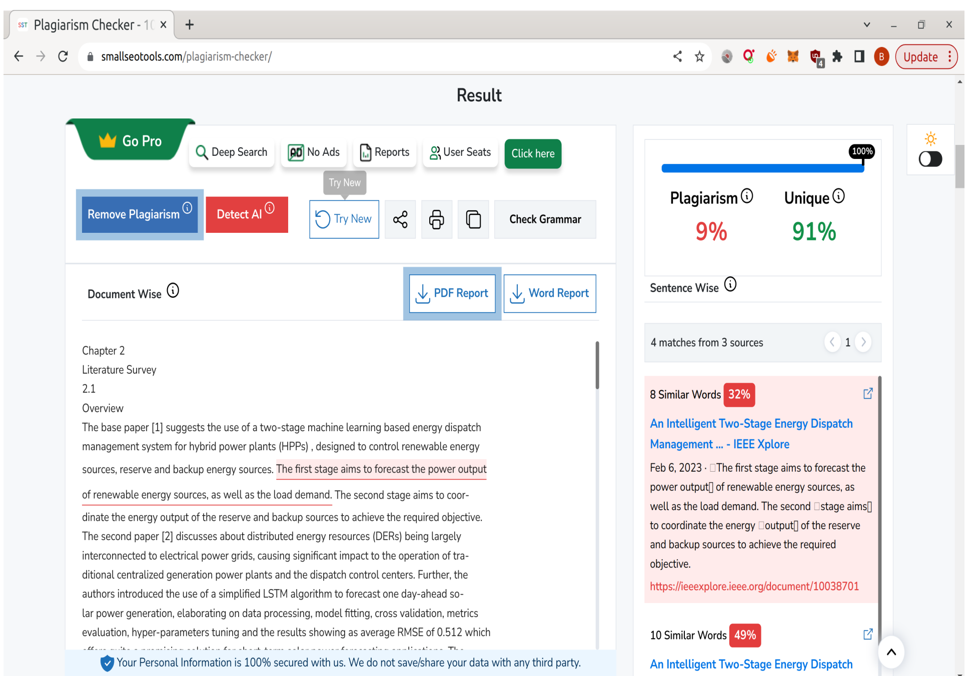
--
Saagar Kabra
Director - Operations | GTT Foundation (NES)

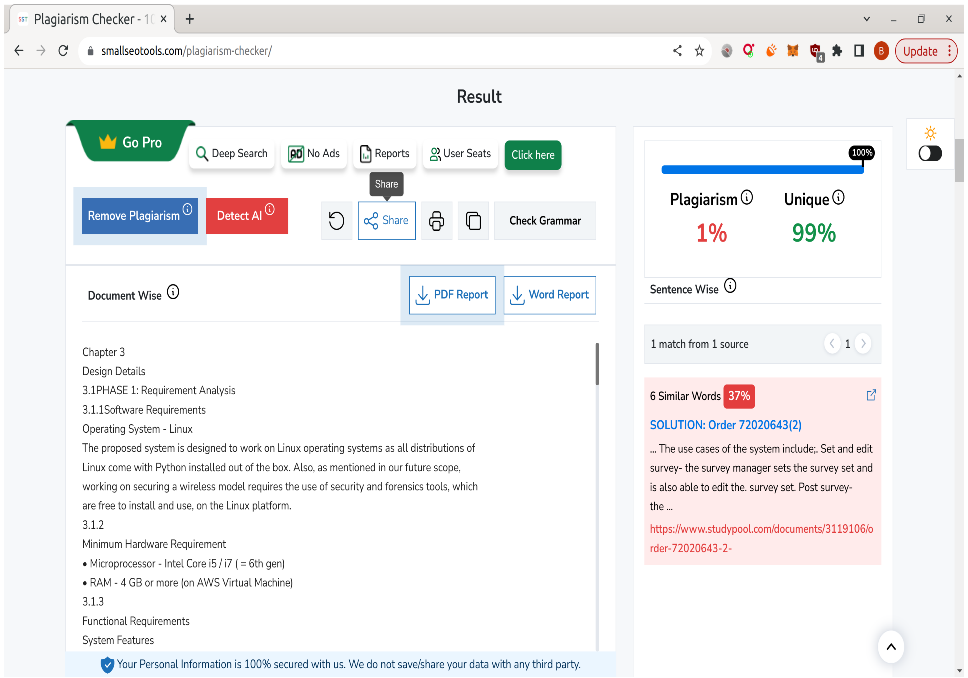
Appendix C

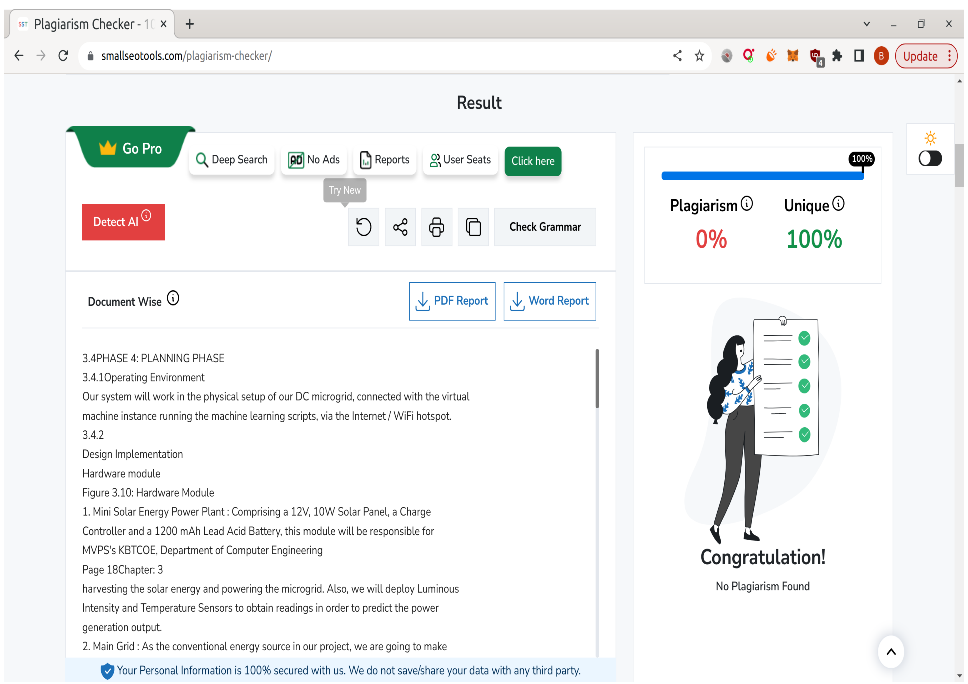
Plagiarism Check Report

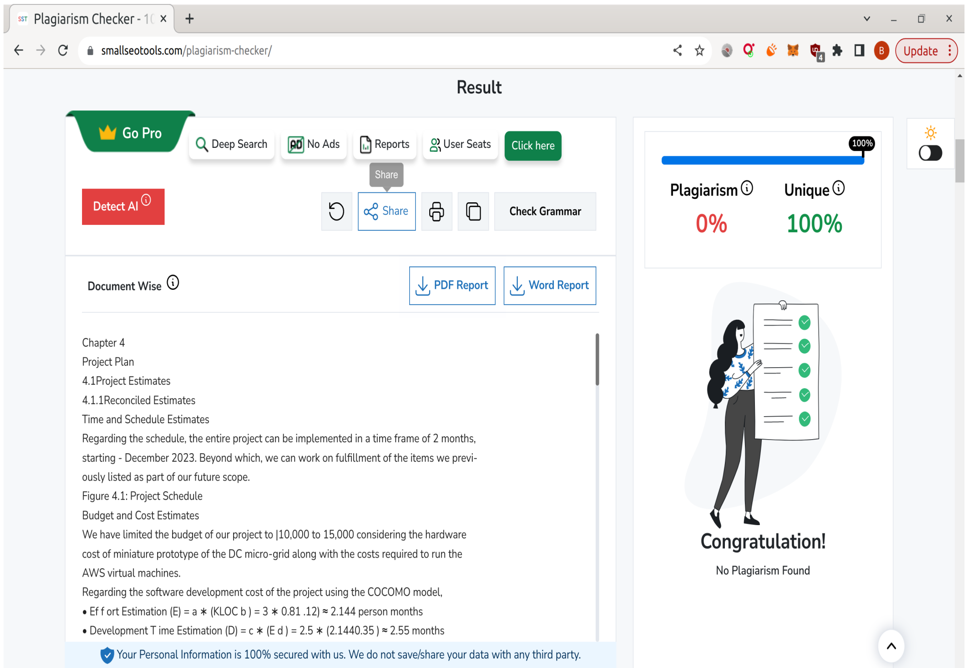


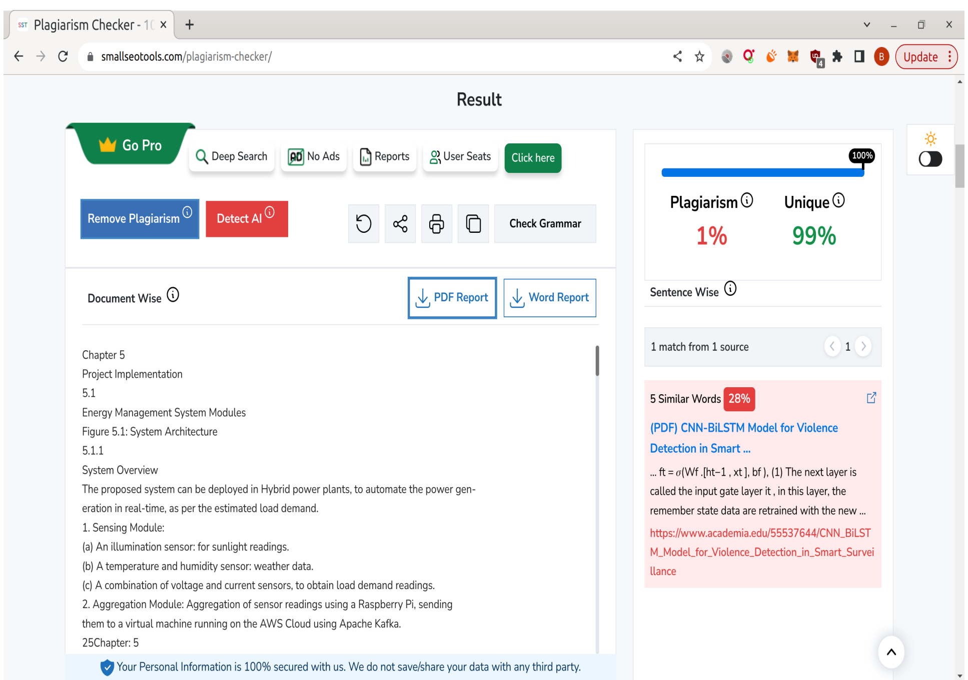

Plagiarism Checker - 1(
smallseotools.com/plagiarism-checker/
Update
Result
Go Pro
Deep Search
No Ads
Reports
User Seats
Click here
Share
Detect AI
Share
Check Grammar
100%
Plagiarism
Unique
0%
100%
Document Wise
PDF Report
Word Report
Chapter 1
Hybrid Energy inspired DC
Micro-grid System using Machine
Learning
1.1
Introduction
A smart grid is an electrical grid which includes various operations and energy measures in-
cluding: advanced metering infrastructure, smart distribution boards, load control switches,
energy efficient resources.
Artificial Intelligence can be integrated with smart grids, to facilitate energy demand
forecasting and output prediction, to ensure minimal wastage of electricity. Also, renewable
energy sources, having an intermittent nature can be integrated in such a system.
Hence, our project is aimed at building an energy management system software, to use
machine learning to control and regulate renewable and non-renewable energy modules in
a DC micro-grid system, to minimize outages and maximize the output of the system by
efficient use of energy demand forecasting.
Congratulation!
No Plagiarism Found
Your Personal Information is 100% secured with us. We do not save/share your data with any third party.

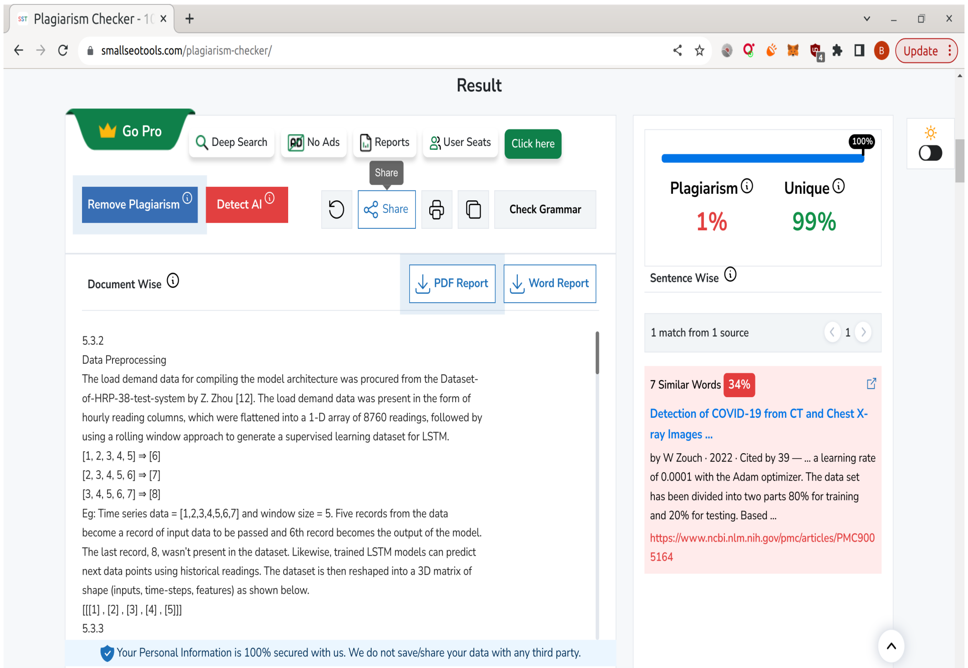

Plagiarism Checker - 1(
smallseotools.com/plagiarism-checker/
Update
Result
Go Pro
Deep Search
No Ads
Reports
User Seats
Click here
Try New
Remove Plagiarism
Detect AI
Try New
Check Grammar
100%
Plagiarism
Unique
9%
91%
Sentence Wise
Document Wise
PDF Report
Word Report
4 matches from 3 sources
1
Chapter 2
Literature Survey
2.1
Overview
The base paper [1] suggests the use of a two-stage machine learning based energy dispatch
management system for hybrid power plants (HPPs) , designed to control renewable energy
sources, reserve and backup energy sources. The first stage aims to forecast the power output
of renewable energy sources, as well as the load demand. The second stage aims to coor-
dinate the energy output of the reserve and backup sources to achieve the required objective.
The second paper [2] discusses about distributed energy resources (DERs) being largely
interconnected to electrical power grids, causing significant impact to the operation of tra-
ditional centralized generation power plants and the dispatch control centers. Further, the
authors introduced the use of a simplified LSTM algorithm to forecast one day-ahead so-
lar power generation, elaborating on data processing, model fitting, cross validation, metrics
evaluation, hyper-parameters tuning and the results showing as average RMSE of 0.512 which
8 Similar Words 32%
An Intelligent Two-Stage Energy Dispatch
Management ... - IEEE Xplore
Feb 6, 2023 · The first stage aims to forecast the
power output of renewable energy sources, as
well as the load demand. The second stage aims
to coordinate the energy output of the reserve
and backup sources to achieve the required
objective.
https://ieeexplore.ieee.org/document/10038701
10 Similar Words 49%
An Intelligent Two-Stage Energy Dispatch
Your Personal Information is 100% secured with us. We do not save/share your data with any third party.

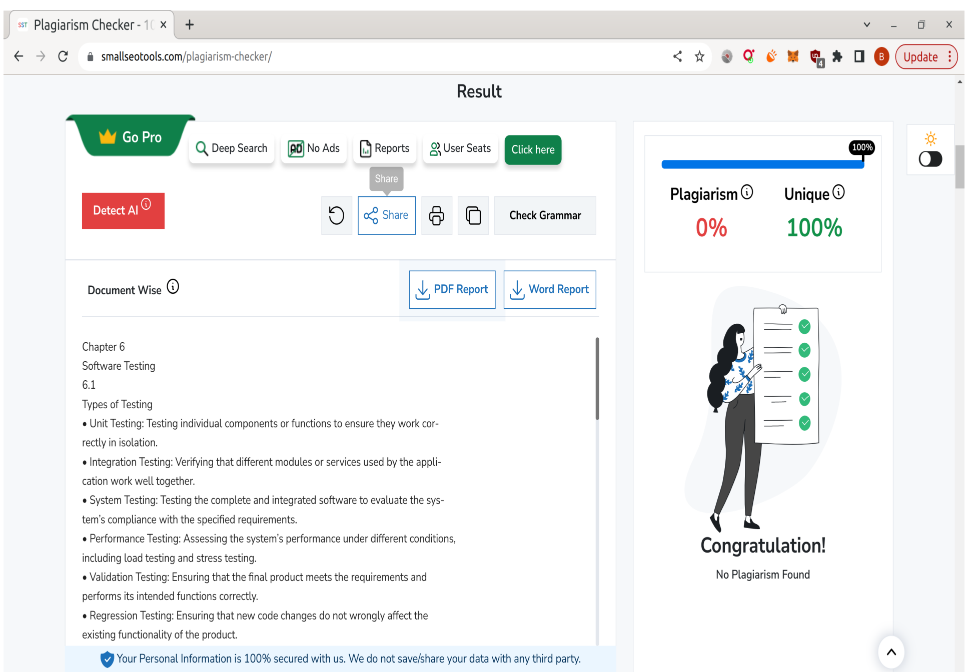

Plagiarism Checker - 1(
smallseotools.com/plagiarism-checker/
Result
Go Pro
Deep Search
No Ads
Reports
User Seats
Click here
Remove Plagiarism
Detect AI
Share
Check Grammar
Document Wise
PDF Report
Word Report
Chapter 3
Design Details
3.1PHASE 1: Requirement Analysis
3.1.1Software Requirements
Operating System - Linux
The proposed system is designed to work on Linux operating systems as all distributions of
Linux come with Python installed out of the box. Also, as mentioned in our future scope,
working on securing a wireless model requires the use of security and forensics tools, which
are free to install and use, on the Linux platform.
3.1.2
Minimum Hardware Requirement
• Microprocessor - Intel Core i5 / i7 (= 6th gen)
• RAM - 4 GB or more (on AWS Virtual Machine)
3.1.3
Functional Requirements
System Features
Your Personal Information is 100% secured with us. We do not save/share your data with any third party.
100%
Plagiarism
Unique
1%
99%
Sentence Wise
1 match from 1 source
6 Similar Words 37%
SOLUTION: Order 72020643(2)
... The use cases of the system include:. Set and edit survey- the survey manager sets the survey set and is also able to edit the. survey set. Post survey- the ...
https://www.studypool.com/documents/3119106/order-72020643-2-

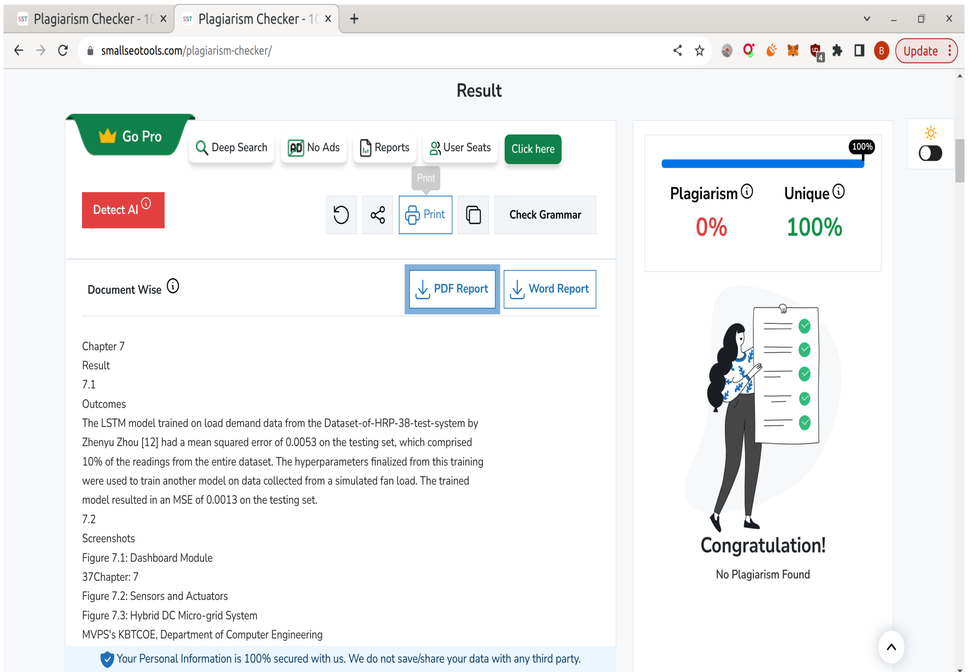

Plagiarism Checker - 1(
smallseotools.com/plagiarism-checker/
Result
Go Pro
Deep Search
No Ads
Reports
User Seats
Click here
Try New
Detect AI
Check Grammar
Document Wise
PDF Report
Word Report
3.4PHASE 4: PLANNING PHASE
3.4.1Operating Environment
Our system will work in the physical setup of our DC microgrid, connected with the virtual
machine instance running the machine learning scripts, via the Internet / WiFi hotspot.
3.4.2
Design Implementation
Hardware module
Figure 3.10: Hardware Module
1. Mini Solar Energy Power Plant : Comprising a 12V, 10W Solar Panel, a Charge
Controller and a 1200 mAh Lead Acid Battery, this module will be responsible for
MVPS's KBTCOE, Department of Computer Engineering
Page 18Chapter: 3
harvesting the solar energy and powering the microgrid. Also, we will deploy Luminous
Intensity and Temperature Sensors to obtain readings in order to predict the power
generation output.
2. Main Grid : As the conventional energy source in our project, we are going to make
Your Personal Information is 100% secured with us. We do not save/share your data with any third party.
100%
Plagiarism
Unique
0%
100%
Congratulation!
No Plagiarism Found

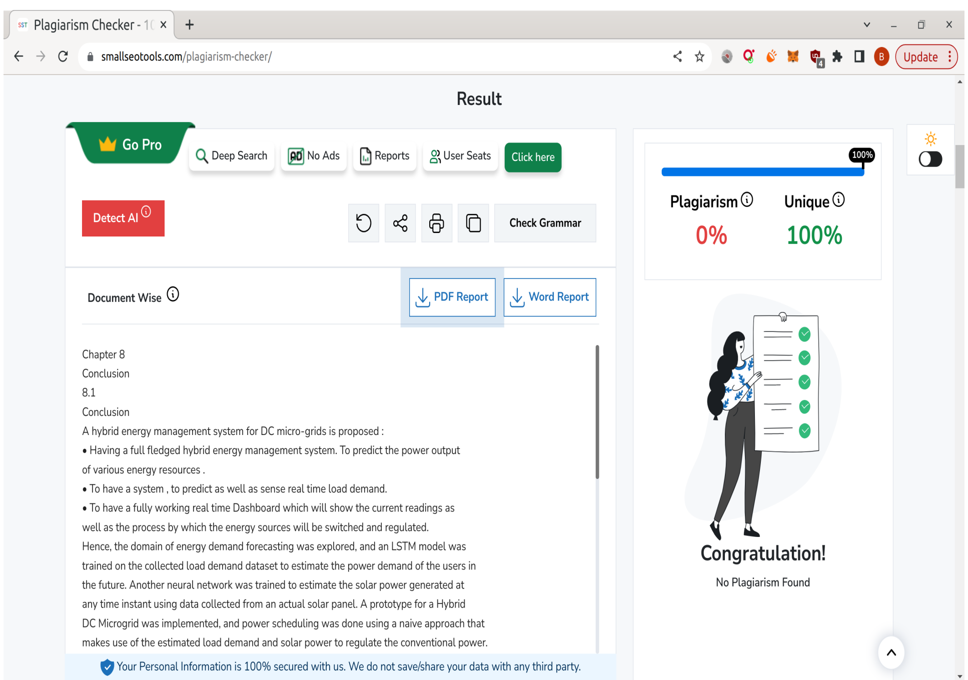

Plagiarism Checker - 1
smallseotools.com/plagiarism-checker/
Update
Result
Go Pro
Deep Search
No Ads
Reports
User Seats
Click here
Share
Detect AI
Share
Check Grammar
100%
Plagiarism
Unique
0%
100%
Document Wise
PDF Report
Word Report
Chapter 4
Project Plan
4.1Project Estimates
4.1.1Reconciled Estimates
Time and Schedule Estimates
Regarding the schedule, the entire project can be implemented in a time frame of 2 months,
starting - December 2023. Beyond which, we can work on fulfillment of the items we previ-
ously listed as part of our future scope.
Figure 4.1: Project Schedule
Budget and Cost Estimates
We have limited the budget of our project to |10,000 to 15,000 considering the hardware
cost of miniature prototype of the DC micro-grid along with the costs required to run the
AWS virtual machines.
Regarding the software development cost of the project using the COCOMO model,
• Ef f ort Estimation (E) = a ∗ (KLOC b) = 3 ∗ 0.81 .12) ≈ 2.144 person months
• Development T ime Estimation (D) = c ∗ (E d) = 2.5 ∗ (2.1440.35) ≈ 2.55 months
Your Personal Information is 100% secured with us. We do not save/share your data with any third party.
Congratulation!
No Plagiarism Found

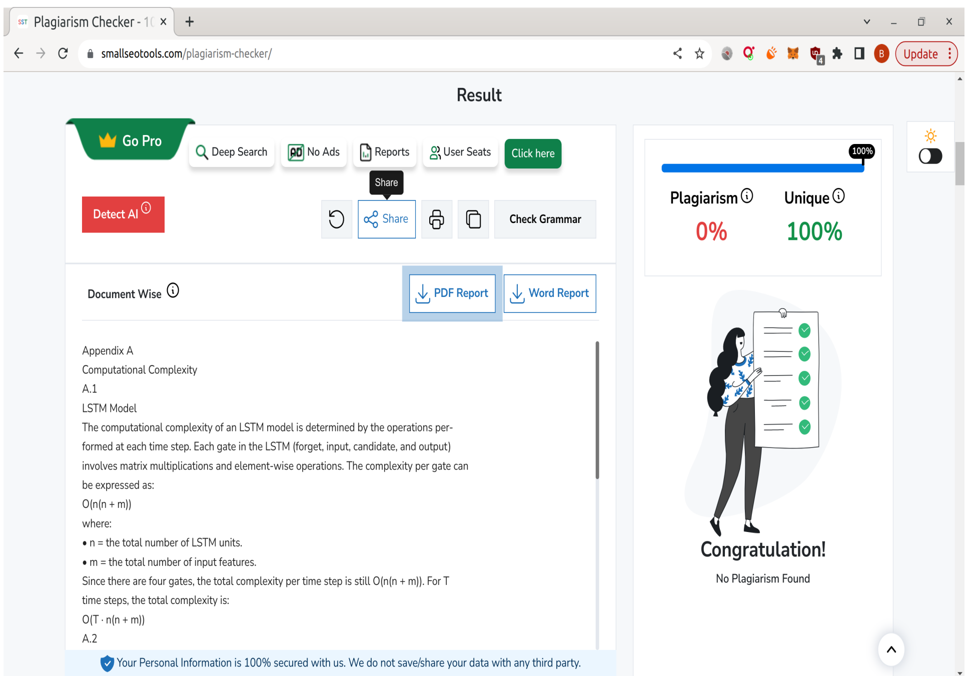

Plagiarism Checker - 1
smallseotools.com/plagiarism-checker/
Update
Result
Go Pro
Deep Search
No Ads
Reports
User Seats
Click here
Remove Plagiarism
Detect AI
Check Grammar
100%
Plagiarism
Unique
1%
99%
Sentence Wise
Document Wise
PDF Report
Word Report
Chapter 5
Project Implementation
5.1
Energy Management System Modules
Figure 5.1: System Architecture
5.1.1
System Overview
The proposed system can be deployed in Hybrid power plants, to automate the power gen-
eration in real-time, as per the estimated load demand.
1. Sensing Module:
(a) An illumination sensor: for sunlight readings.
(b) A temperature and humidity sensor: weather data.
(c) A combination of voltage and current sensors, to obtain load demand readings.
2. Aggregation Module: Aggregation of sensor readings using a Raspberry Pi, sending
them to a virtual machine running on the AWS Cloud using Apache Kafka.
25Chapter: 5
Your Personal Information is 100% secured with us. We do not save/share your data with any third party.
1 match from 1 source
1
5 Similar Words
28%
(PDF) CNN-BiLSTM Model for Violence
Detection in Smart ...
... ft = σ(Wf .[ht−1 , xt], bf), (1) The next layer is
called the input gate layer it , in this layer, the
remember state data are retrained with the new ...
https://www.academia.edu/55537644/CNN_BiLST
M_Model_for_Violence_Detection_in_Smart_Survei
llance


Plagiarism Checker - 1
smallseotools.com/plagiarism-checker/
Update
Result
Go Pro
Deep Search
No Ads
Reports
User Seats
Click here
Share
Remove Plagiarism
Detect AI
Share
Check Grammar
Document Wise
PDF Report
Word Report
5.3.2
Data Preprocessing
The load demand data for compiling the model architecture was procured from the Dataset-
of-HRP-38-test-system by Z. Zhou [12]. The load demand data was present in the form of
hourly reading columns, which were flattened into a 1-D array of 8760 readings, followed by
using a rolling window approach to generate a supervised learning dataset for LSTM.
[1, 2, 3, 4, 5] ⇒ [6]
[2, 3, 4, 5, 6] ⇒ [7]
[3, 4, 5, 6, 7] ⇒ [8]
Eg: Time series data = [1,2,3,4,5,6,7] and window size = 5. Five records from the data
become a record of input data to be passed and 6th record becomes the output of the model.
The last record, 8, wasn't present in the dataset. Likewise, trained LSTM models can predict
next data points using historical readings. The dataset is then reshaped into a 3D matrix of
shape (inputs, time-steps, features) as shown below.
[[[1] , [2] , [3] , [4] , [5]]]
5.3.3
Your Personal Information is 100% secured with us. We do not save/share your data with any third party.
100%
Plagiarism
Unique
1%
99%
Sentence Wise
1 match from 1 source
1
7 Similar Words
34%
Detection of COVID-19 from CT and Chest X-ray Images ...
by W Zouch · 2022 · Cited by 39 — ... a learning rate of 0.0001 with the Adam optimizer. The data set has been divided into two parts 80% for training and 20% for testing. Based ...
https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9005164


Plagiarism Checker - 1
smallseotools.com/plagiarism-checker/
Update
Result
Go Pro
Deep Search
No Ads
Reports
User Seats
Click here
Share
Detect AI
Share
Check Grammar
Document Wise
PDF Report
Word Report
Chapter 6
Software Testing
6.1
Types of Testing
• Unit Testing: Testing individual components or functions to ensure they work cor-
rectly in isolation.
• Integration Testing: Verifying that different modules or services used by the appli-
cation work well together.
• System Testing: Testing the complete and integrated software to evaluate the sys-
tem's compliance with the specified requirements.
• Performance Testing: Assessing the system's performance under different conditions,
including load testing and stress testing.
• Validation Testing: Ensuring that the final product meets the requirements and
performs its intended functions correctly.
• Regression Testing: Ensuring that new code changes do not wrongly affect the
existing functionality of the product.
Your Personal Information is 100% secured with us. We do not save/share your data with any third party.
100%
Plagiarism
Unique
0%
100%
Congratulation!
No Plagiarism Found


Plagiarism Checker - 1
Plagiarism Checker - 1
smallseotools.com/plagiarism-checker/
Update
Result
Go Pro
Deep Search
No Ads
Reports
User Seats
Click here
Detect AI
Print
Check Grammar
Document Wise
PDF Report
Word Report
Chapter 7
Result
7.1
Outcomes
The LSTM model trained on load demand data from the Dataset-of-HRP-38-test-system by
Zhenyu Zhou [12] had a mean squared error of 0.0053 on the testing set, which comprised
10% of the readings from the entire dataset. The hyperparameters finalized from this training
were used to train another model on data collected from a simulated fan load. The trained
model resulted in an MSE of 0.0013 on the testing set.
7.2
Screenshots
Figure 7.1: Dashboard Module
37Chapter: 7
Figure 7.2: Sensors and Actuators
Figure 7.3: Hybrid DC Micro-grid System
MVPS's KBTCOE, Department of Computer Engineering
Your Personal Information is 100% secured with us. We do not save/share your data with any third party.
100%
Plagiarism
Unique
0%
100%
Congratulation!
No Plagiarism Found


Plagiarism Checker - 1
smallseotools.com/plagiarism-checker/
Update
Result
Go Pro
Deep Search
No Ads
Reports
User Seats
Click here
Detect AI
Check Grammar
Document Wise
PDF Report
Word Report
Chapter 8
Conclusion
8.1
Conclusion
A hybrid energy management system for DC micro-grids is proposed :
• Having a full fledged hybrid energy management system. To predict the power output
of various energy resources .
• To have a system , to predict as well as sense real time load demand.
• To have a fully working real time Dashboard which will show the current readings as
well as the process by which the energy sources will be switched and regulated.
Hence, the domain of energy demand forecasting was explored, and an LSTM model was
trained on the collected load demand dataset to estimate the power demand of the users in
the future. Another neural network was trained to estimate the solar power generated at
any time instant using data collected from an actual solar panel. A prototype for a Hybrid
DC Microgrid was implemented, and power scheduling was done using a naive approach that
makes use of the estimated load demand and solar power to regulate the conventional power.
Your Personal Information is 100% secured with us. We do not save/share your data with any third party.
100%
Plagiarism
Unique
0%
100%
Congratulation!
No Plagiarism Found


Plagiarism Checker - 1(
smallseotools.com/plagiarism-checker/
Update
Result
Go Pro
Deep Search
No Ads
Reports
User Seats
Click here
Share
Detect AI
Share
Check Grammar
100%
Plagiarism
Unique
0%
100%
Document Wise
PDF Report
Word Report
Appendix A
Computational Complexity
A.1
LSTM Model
The computational complexity of an LSTM model is determined by the operations per-
formed at each time step. Each gate in the LSTM (forget, input, candidate, and output)
involves matrix multiplications and element-wise operations. The complexity per gate can
be expressed as:
O(n(n + m))
where:
• n = the total number of LSTM units.
• m = the total number of input features.
Since there are four gates, the total complexity per time step is still O(n(n + m)). For T
time steps, the total complexity is:
O(T · n(n + m))
A.2
Your Personal Information is 100% secured with us. We do not save/share your data with any third party.
Congratulation!
No Plagiarism Found

Bibliography

[1] Mostafa M. Shibl, Loay S. Ismail, Ahmed M. Massou (2023) An Intelligent Two-Stage Energy Dispatch Management System for Hybrid Power Plants: Impact of Machine Learning Deployment, https://ieeexplore.ieee.org/document/10038701

[2] Chun-Hung Liu, Jyh-Cherng Gu, Ming-Ta Yang (2021) A Simplified LSTM Neural Networks for One Day-ahead Solar Power Forecasting, https://ieeexplore.ieee.org/document/9333638

[3] Baraa Mohandes, Maisam Wahbah, Mohamed Shawky El Moursi, Tarek H.M. El-Fouly (2021) Renewable Energy Management System: Optimum Design and Hourly Dispatch, https://ieeexplore.ieee.org/document/9351770

[4] Bing Zhu, Henerica Tazvinga, Xiaohua Xia (2014) switched model predictive control for energy dispatching of a photovoltaic diesel battery hybrid power system https://ieeexplore.ieee.org/document/6933909

[5] Team National Smart Grid Mission (2023) Smart Grid Pilot Projects under IPDS (erstwhile RAPDRP Part-C), https://www.nsgm.gov.in/en/sg-pilot

[6] Ankit Gupta (2023) Smart Grid Market Overview, https://www.marketresearchfuture.com/reports/smart-grid-market-1110

[7] Team Wikipedia (2023) Smart Grid, https:/en.wikipedia.org/wiki/Smart_grid

[8] Team SAP (2023) The smart grid: How AI is powering today's energy technologies — SAP Insights, https://www.sap.com/insights/smart-grid-ai-in-energy-technologies.html

[9] Kamil Talar (2023) Understanding what is Peak Shaving: Techniques and Benefits, https://howtostoreelectricity.com/what-is-peak-shaving

[10] Team IBEF (2023) Power Sector - India, www.ibef.org/industry/power-sector-india.

[11] Stephen Turner, Suleyman Uludag (2015), https://www.researchgate.net/publication/276288597_Towards_Smart_Cities_Interaction_and_Synergy_of_the_Smart_Grid_and_Intelligent_Transportation_Systems

[12] Zhenyu Zhuo, "DATASET-OF-HRP-38-TEST-SYSTEM", IEEE DataPort, DOI: 10.21227/ggy4-7497, May 17, 2022.