|  |  |
| --- | --- |
| http://paperpkblog.com/admissions/wp-content/uploads/2015/05/Jobs-in-Foundation-University-Islamabad-FUI-20141.jpg | **FOUNDATION UNIVERSITY ISLAMABAD** |

**INTELLIGENT ENERGY SCENARIO ANALYSIS**

***Submitted by***

FA21/BSSE/136 – Muhammad Farzam Baig

FA21/BSSE/138 – Muhammad Suffian Tafoor

FA21/BSSE/139 – Muhammad Yasir Khan

Group C1

**BACHELORS**

**OF**

**SCIENCE SOFTWARE ENGINEERING**

**YEAR**

**2021-2025**

***Project Advisor***

Muhammad Shaheen Khan

***Project Coordinator***

Sajid Ali Khan

Submitted to the Department of Software Engineering of Foundation University Islamabad, in partial fulfilment for the requirements of a Bachelor of Science Degree in Software Engineering

**DEDICATION**

We dedicate our work on the Intelligent Energy Scenario Analysis (IESA) to our beloved parents who have been supportive and encouraged us throughout every phase of our journey. Their belief in us has been the foundation of our achievements, without their support and love we might never have achieved such heights as we are today.

We also extend our deepest gratitude to our beloved project supervisors, Muhammad Shaheen Khan and our Co-Supervisor Mr Muhammad Usman Khan without their supervision, advice, and valuable guidance completion of this task would have been difficult or near to impossible.

**ACKNOWLEDGEMENTS**

Thanks to Almighty ALLAH for granting us the strength, determination, and countless blessings. By the grace of ALLAH, we were able to produce a complete and quality product. With heartfelt appreciation, we thank our friends and family whose support in both challenging and joyful times fuelled our motivation.

We sincerely acknowledge the effort and guidance of our project supervisors Muhammad Shaheen Khan and our Co-Supervisor Mr Muhammad Usman Khan. Whose involvement and encouragement at every step boost our motivation in doing this project. Additionally, we sincerely thank the FUSST Faculty for their support and resources, which greatly contributed to our learning and growth throughout our university journey.

**STATEMENT OF ORIGINALITY**

We namely **Muhammad Farzam Baig, Muhammad Suffian Tafoor and Muhammad Yasir Khan, s**ubmitting report titled “**Intelligent Energy Scenario Analysis**”. state that we clearly understand what plagiarism is and we have also read about it using online sources. As such, we claim that this entire work (code, reports etc.) is our own effort. We have not used/copy or pasted or paraphrased even a single line of code or sentence from any other source without giving a proper reference to it. Based on our confidence in our originality, we allow the University to run anti-plagiarism software on the same.

We understand that if now, or in the future, ever it is found out that we have not been honest and have plagiarized, the university will take action and will result in disciplinary action, such as the cancellation of our degree and we can be liable for any subsequent punishments as deemed appropriate by the University as well as the Higher Education Commission Pakistan.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Muhammad Farzam Baig

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Muhammad Suffian Tafoor

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Muhammad Yasir Khan

**COPYRIGHT STATEMENT**

Copyright in text of this report rests with the student authors. Copies (by any process) either in full, or of extracts, may be made only in accordance with instructions given by the author. Details may be obtained by the Librarian. This page must form part of any such copies made. Further copies (by any process) of copies made in accordance with such instructions may not be made without the permission (in writing) of the author.

The ownership of any intellectual property rights which may be described in this project is vested in FUI, subject to any prior agreement to the contrary, and may not be made available for use by third parties without the written permission of the FUI, which will prescribe the terms and conditions of any such agreement.

**CERTIFICATE OF APPROVAL**

It is certified that Project titled **“IESA”,** presented on \_\_17 Dec,2024\_\_\_, has been duly approved by the evaluation committee.

|  |  |
| --- | --- |
| **Project Advisor:** | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Muhammad Shaheen Khan |
| **Project Coordinator:** | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Sajid Ali Khan |
| **Manager Graduate Off:** | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Arif Jamal |

**ABSTRACT**

The Intelligent Energy Scenario Analysis (IESA) system is an AI-driven software solution designed to analyse, forecast, and optimize energy Related scenarios. The system focuses on both gas and electricity usage, enabling energy planners, IT administrators, and data operators to make informed decisions. IESA integrates historical data visualization, predictive modelling, and scenario-based analysis to address the increasing demand for energy efficiency and sustainable practices. Using advanced algorithms such as WisRule for cognitive association and Linear Regression predictive insights, the system allows users to evaluate multiple energy scenarios by adjusting variables like production capacity, imports, and consumption trends. Then these trends are used for predictions and their results are visualized on user friendly dashboards, providing users with clear and actionable insights.

IESA is a DSS, that enables users to take optimal and cost-effective decision for energy management with application at national and regional level. IESA provides accurate predictions for different scenarios and also provides tailored recommendations for those scenarios for optimization energy usage thus reducing energy wastage. Multiple Scenarios analysis ability of IESA’s and comparing of outcomes allows energy planners to make most optimal and strategical decision for energy resources.

To further enhance IESA, future work will explore the integration of renewable resources of energy like Solar Energy, Wind Energy etc. making it a more comprehensive Decision Support System for energy planning. By integrating IESA with IoT devices to get Realtime and advance computing technologies will increase IESA’s computing performance and accuracy. Future work also includes expanding IESA to handle energy trades between countries in order to reduce energy wastage and to track carbon emissions and allow decision maker take swift action against increasing carbon emissions in order to meet global emissions/sustainability goals.

**TABLE OF CONTENTS**

Chapter 1: [Introduction 1](#_Toc191930448)

[**1.1** **Introduction** 1](#_Toc191930449)

[**1.2** **Existing System** 1](#_Toc191930450)

[**1.3** **Literature Review** 3](#_Toc191930451)

[**1.4** **Problem Definition** 4](#_Toc191930452)

[**1.5** **Concept Diagram** 5](#_Toc191930453)

[**1.6** **User Needs** 5](#_Toc191930454)

Chapter 2: [Introduction to Proposed System 7](#_Toc191930455)

[**2.1.** **Introduction** 7](#_Toc191930456)

[**2.2.** **Project Background or Overview** 7](#_Toc191930457)

[**2.3.** **Problem Description** 8](#_Toc191930458)

[**2.4.** **Project Objectives** 8](#_Toc191930459)

[**2.5.** **Scope** 9](#_Toc191930460)

[**2.6.** **Project Features** 9](#_Toc191930461)

[**2.7.** **DFD level 0** 10](#_Toc191930462)

Chapter 3: [Requirements Specification 11](#_Toc191930463)

[**3.1.** **Introduction** 11](#_Toc191930466)

[**3.2.** **Functional Requirements** 11](#_Toc191930467)

[**3.3.** **Flow model** 11](#_Toc191930468)

[**3.4.** **Graphical User Interface** 13](#_Toc191930469)

[**3.5.** **Data Model (ERD)** 17](#_Toc191930470)

[**3.6.** **Non-Functional Requirements** 18](#_Toc191930471)

Chapter 4: [Design Specification 19](#_Toc191930472)

[**4.1** **Introduction** 19](#_Toc191930473)

[**4.2** **System Architect** 19](#_Toc191930474)

[**4.3** **Design Methodology** 19](#_Toc191930475)

[**4.4** **High Level Design** 20](#_Toc191930476)

[**4.5** **Data design** 20](#_Toc191930477)

[**4.6** **Detail Design or Interaction diagram (Sequence or collaboration diagram, package Diagram)** 20](#_Toc191930478)

[**4.7** **Class diagram** 20](#_Toc191930479)

[**4.8** **Deployment diagram** 20](#_Toc191930480)

Chapter 5: [Test Specification 21](#_Toc191930481)

[**5.1** **Introduction** 21](#_Toc191930482)

[**5.2 Test Design** 21](#_Toc191930483)

[**5.2.1 White-Box Test Cases** 22](#_Toc191930484)

[**5.2.2 Black-Box Test Cases** 23](#_Toc191930485)

[**5.2.3 GUI Test Cases** 23](#_Toc191930486)

[**5.2.4 Other NFR Test Cases** 25](#_Toc191930487)

[**5.2.5 Usability Testing** 26](#_Toc191930488)

[**5.2.6 Software Performance Testing** 27](#_Toc191930489)

[**5.2.7 Compatibility Testing** 28](#_Toc191930490)

[**5.2.8 Load Testing** 28](#_Toc191930491)

[**5.2.9 Security Testing** 29](#_Toc191930492)

[**5.2.10 Installation Testing** 30](#_Toc191930493)

[**5.2.11 Acceptance Test Cases** 32](#_Toc191930494)

[**5.3 Defect or Bug Sheet** 33](#_Toc191930495)

[**5.4 Test Report** 34](#_Toc191930496)

Chapter 6: [Bibliography 38](#_Toc191930497)

**LIST OF TABLES**

[Table 1.1: Comparison with Existing Systems and Features 2](#_Toc187324221)

[Table 2.1 : List of Features of IESA 9](#_Toc187324222)

[Table 3.1: Functional Requirements for IESA 11](#_Toc187324223)

[Table 3.6: Non-Functional Requirements for IESA 18](#_Toc187324224)

**LIST OF ILLUSTRATIONS**

[Figure 1.1: Concept Diagram for IESA 5](#_Toc187334983)

[Figure 2.1: DFD level 0 for IESA 10](#_Toc187334984)

[Figure 3.1:Flow Model Diagram for IESA 12](#_Toc187334985)

[Figure 3.2:Login User interface for IESA](#_Toc187334985) 13

[Figure 3.3: Main Dashboard for Data Planner](#_Toc187334986) 14

[Figure 3.4: Dashboard User Interface for Input Entry Operator………………………...](#_Toc187334987)15

[Figure 3.5: Contact Us User Interface for IESA 16](#_Toc187334988)

[Figure 3.6:ERD diagram for IESA 17](#_Toc187334989)

**LIST OF ABBREVIATIONS**

|  |  |
| --- | --- |
| **Word** | **Phrase Definition / Abbreviation** |
| AI | Artificial Intelligence |
| DFD | Data Flow Diagram |
| DS | Data Science |
| DSS | Decision Support System |
| FR | Functional Requirement |
| IESA | Intelligent Energy scenario Analysis |
| ML | Machine Learning |
| NFR | Non-Functional Requirement |
| SRS | Software Requirements Specification |

Chapter 1

# Introduction

* 1. **Introduction**

The report provides an overview of IESA stands for Intelligent Energy Scenario Analysis system, which is a cutting-edge decision support system designed to assist energy planner in energy related decisions and to promote sustainable energy management and efficient energy resource allocation and to reduce energy wastage. We are developing IESA as a part of BSSE (Bachelor of Software Engineering) degree at Foundation University Islamabad. This document contains all the necessary information for methodologies, implementations and outcomes of IESA.

In this era, where Increasing demand for energy resources due to increase of populations and people moving to more luxury life, combine with inefficient usage and planning poses a critical challenge in sustainability, resource management and carbon emissions. Traditional methods and existing DSS failed to cope up with complex variables and scenarios and fall short accurate predictions. Most of the existing DSS were developed 5 to 7 years ago, and in these 5 to 7 years forecasting algos have become quite mature and accurate and at the same time those systems are quite expensive for their licensing with steep learning curve which makes it hard for decision makers to make optimal decisions. The IESA system enables its users in making a informed decision as it employs advance AI and DS algorithms like WisRule, Linear Regression, K means Clustering etc. to predict future needs by using historical data to generate trends and then predict future needs and presenting that historical data to measure performance over time and predicted data for future needs on user friendly dashboards to promote decision maker to take most optimal decision for best practices for energy efficiency and sustainability.

* 1. **Existing System**

Existing decision support system face fall short as they encounter various challenges that limits their efficiency and scalability. Most traditional systems lack advanced predictive capabilities and scenario analysis features, providing only basic data visualization and analysis tools. Also, many of these systems are expensive, require a lot of training, and fail to accommodate diverse stakeholder requirements, such as IT administrators, energy planners, and data operators.

For instance, tools like EnergyPLAN and HOMER primarily focus on high-level energy system analysis but lack in terms of AI-driven algorithms for micro level forecasting and optimization. Furthermore, existing systems often overlook personalization and user-friendly interfaces, which limits their adaptability and flexibility for varying user expertise levels. These shortcomings highlight the need for a new and versatile, AI-driven platform like IESA to address these gaps.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Features** | **LoadSEER** | **EnergyPLAN** | **HOMER Pro** | **PLEXOS** | **OSET** | **LEAP** | **ETAP** | **Siemens PSS/E** | **Our System** |
| **Historical Data Visualization** | ✗ | ✓ | ✓ | ✓ | ✗ | ✓ | ✓ | ✓ | ✓ |
| **Prediction Modeling** | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| **Scenario Analysis** | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| **Recommendations and Insights** | ✗ | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✓ | ✓ |
| **Data Visualization in Dashboards** | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✓ | ✗ | ✓ |
| **Exportable Reports** | ✗ | ✗ | ✓ | ✓ | ✗ | ✗ | ✓ | ✓ | ✓ |
| **Comparison among Scenarios** | ✓ | ✗ | ✓ | ✓ | ✓ | ✗ | ✓ | ✓ | ✓ |
| **Wisdom Mining** | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ |

Table 1.1: Comparison with Existing Systems and Features

* 1. **Literature Review**

Numerous studies have highlighted the importance of integrating AI and machine learning in energy management systems. Research have shown that algorithms like Linear Regression and K-means Clustering are effective in predicting energy consumption trends and identifying patterns in datasets with different variables.

WisRule First cognitive algorithm of wise association rule mining algorithm from newly emerging wisdom mining domain. WisRule generates both positive and negative associations which provide a comprehensive understanding of different variables and relationship between them [1].

WisRule uses 4 parameters which include context (C), utility (U), time (T) and location (L) for extracting wisdom from a complex dataset. WisRule is an extension to the CBPNARM algorithm [1]. Extracting Wisdom mining using WisRule can be useful for extracting actionable rules in energy sector. WisRule can analyse historical energy data using that data it can make both positive and negative associations which are not apparent on the face and positive associations highlight benefits energy practices while negative rule could highlights.

Theoretically WisRule performs superior in extracting meaning insights from complex data then other associative rule mining algorithms like aprioir. WisRule is proposed because it generates both positive and negative wise association rules. These rules may be utilized for decision-making with lesser influence of a domain expert [1].

Regression, A statistical technique in which a dependent variable relates to a independent variable (can me more than one). It shows whether change observed in the dependent variable are associated with changes in one or more independent variable [2] [3]. This method is particularly useful in predictive modelling as it provides insights into how change in a specific variable effect the outcome [4].

Regression like Linear, Polynomial, Logistical etc can be utilized to forecast different variables like consumption, import etc [4]. By analysing historical data and identifying patterns regression model can predict different trends which helps users in inform decision making

K-means clustering, an unsupervised ML algorithm used to partition dataset into clusters, where each cluster contains data points that are like each other than those in other clusters. The algorithm works by iteratively assigning data points to clusters based on proximity to the cluster centroid (centroid is the mean of data points of that cluster). It recalculates centroid based on the updated clusters and repeat this process until the data points no longer change [5].

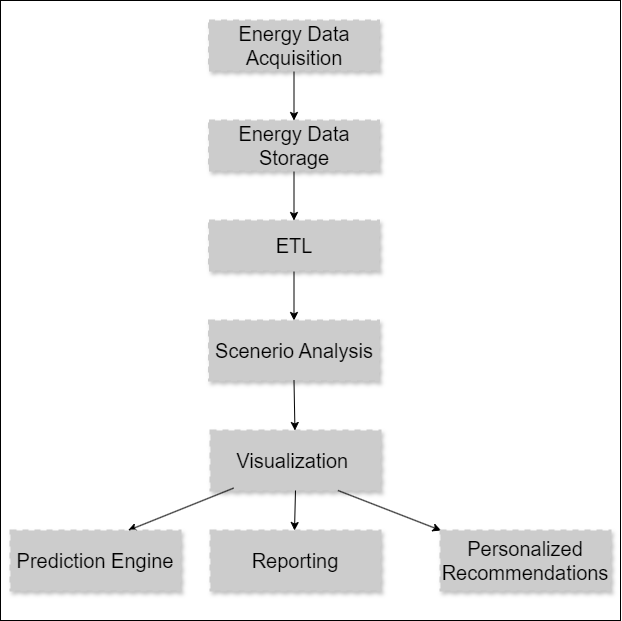
K-means clustering can be employed to categorize various scenarios patterns based on different factors like geography, Location, Time etc. [6]. By segmenting energy usage into distinct clusters IESA can better understand the behavior of different clusters (groups). For example, K-means can identify regions with higher energy consumption and help out in finding the factors contributing to this high usage. These clusters can be used with regression to forecast future needs more accurately and provide recommendations for optimization energy distribution.

* 1. **Problem Definition**

The world currently facing serious energy crisis due to increase in population, shift in people lifestyle and modern society relieving on some sort of energy for almost every activity, also increasing demand by industrial sector and energy wastage are also major contributors in energy crisis. Existing DSS solutions fail to provide accurate and actionable insights as they encounter various challenges that limits their efficiency and accuracy like outdated predictors etc. and also existing systems are quite complex with steep learning complex which makes it hard for decision makers to make an inform and optimal decision as they have to completely learn a complex system in order to properly use it.

The Intelligent Energy Scenario Analysis system set to counter these problems by combining Advance AI and Data Science Algorithms to predict future with user friendly dashboards to visualize both historical and predicted data so that decision makers can make informed and optimal decision for energy management to promote sustainable energy management and efficient energy resource allocation and to reduce energy wastage

* 1. **Concept Diagram**

Figure 1.1: Concept Diagram for IESA

* 1. **User Needs**

IESA is designed to meet different diverse needs of its users which mainly include energy management, energy planning and decision making. Users’ needs include:

* Predictive Insights: Users require predictive modelling capabilities to anticipate future energy demands and optimize resource allocation.
* Historical Data Analysis: Users requires the system to provide historical data analysis to analyse trends which are then visualized so that users can monitor their performance over time
* Personalized Dashboards: Input entry operators need customizable dashboards to visualize data and insights in formats that suit their roles and preferences.
* Sustainability Metrics: Stakeholders need tools to calculate carbon footprints and receive actionable recommendations to promote sustainable energy practices.
* Ease of Use: The system must feature an intuitive interface that accommodates users with changing levels of technical expertise, ensuring accessibility and efficiency.
* Real-Time Updates: Users need timely updates and alerts about energy trends, anomalies, or savings opportunities to respond quickly to emerging situations.
* Data Security: All user data must be securely managed, ensuring compliance with privacy standards and protecting sensitive information.

Chapter 2

# Introduction to Proposed System

* 1. **Introduction**

The proposed system IESA, a sophisticated business intelligence software addresses the need of a decision support system which uses Advance Data science and AI algorithms to assist decision-maker in making optimal decisions. Energy sector currently needs innovative approaches to ensure to sustainability and reduce energy wastage and increase efficiency to lower carbon emissions and reduce cost and promote best environment friendly practices. However, Current available system can’t provide detailed and accurate predictions and actionable insights in a user-friendly way.

IESA is designed to enable users to take advantage of advanced AI and Data Science algorithms like Regression, WisRule (World’s first cognitive algorithm for wise association rule mining) and K means etc. to get detailed and accurate predictions and actionable insights to take most optimal decision and also offers decision makers personalized recommendations. IESA serves as a useful tool for advancing decision making for move to sustainable energy practices on a national level.

* 1. **Project Background or Overview**

IESA (Intelligent Energy Scenario Analysis), an AI based business intelligence project that’s set to revolutionize energy scenario analysis and forecasting by utilizing WisRule, Regression, K means Clustering etc. This will assist decision makers understand different demand profiles and identify any inefficiencies and then target to climate that inefficiency to promote energy efficiency interventions more effectively. IESA aims to analyze and predict different scenarios related to different forms of energy like gas and electricity.

IESA analyzes historical energy data like other existing DSS but unlike other existing DSS it also predicts future for different scenarios and also provides personalized recommendations promote optimal future decision-making for moving energy sector sustainability and lower energy cost and reduce wastage.

* 1. **Problem Description**

Increasing demand for energy resources due to increase of populations and people moving to more luxury life, combine with inefficient usage and planning poses a critical challenge in sustainability, resource management and carbon emissions. Traditional methods and existing DSS failed to cope up with complex variables and scenarios and fall short accurate predictions. Most of existing DSS are developed 5 to 7 years ago, and in these 5 to 7 years forecasting algos have become quite mature and accurate and at the same time those systems are quite expensive for their licensing with steep learning curve. They also lack dynamic predictions and personalized recommendations to assist decision-makers. Which ultimately lead to inefficient energy resource planning, allocation and increased cost.

The rising demand for energy, coupled with inefficiencies in usage planning, presents major challenges in sustainability, resource management, and cost control. Traditional approaches and existing Decision Support Systems (DSS) for energy planning and forecasting fall short in addressing complex variables, delivering precise forecasts, and offering actionable insights for energy optimization. These systems are often costly to license, complicated to use, and lack user-friendly features, making them difficult to adopt. Moreover, they fail to provide dynamic predictions or personalized recommendations, resulting in inefficient resource allocation, higher costs, and minimal environmental impact.

There is a need for an intelligent solution that provides accurate cost-effective and user-friendly solutions and IESA addresses these problems with personalized recommendations, and actionable insights, enabling users to optimize energy usage that supports the data-driven decisions.

* 1. **Project Objectives**

The Intelligent Energy Scenario Analysis (IESA) is a sophisticated business intelligence project designed to enable users to optimize gas and electricity usage. By helping to reduce energy wastage, lower costs, and contribute to a greener future on a national scale, IESA offers an impactful solution for sustainable energy management. The software utilizes historical data and employs the WisRule, K-Means, and Linear Regression to analyze, visualize patterns of energy production and consumption-related scenarios, providing predictive insights, group data based on similar consumption patterns, and forecasting energy usage and predicting future energy trends. IESA aims to predict future energy-related scenarios, such as gas and electricity production and consumption, as well as associations between energy import, generation, and production etc.

* 1. **Scope**

The Intelligent Energy Scenario Analysis (IESA) provides a comprehensive decision support system for energy management by providing detailed predictive analytics, and scenario-based insights, forecasting and optimization of different energy scenarios such as energy consumption and production.

As an advanced system, IESA captures historical consumption data, stores it, and makes it accessible for visualization and scenario analysis. Allows users to explore predictive trends, receive personalized recommendations for optimizing energy usage, and compare predictions using various algorithms. The System also promotes sustainability by offering actionable insights, helping organizations minimize energy wastage, reduce costs, and support eco-friendly practices.

* 1. **Project Features**

The Features table below outlines the functionality of the IESA which includes 9 features including Energy Data Acquisition, ETL, Prediction Engine etc.

Table 2.1: List of Features of IESA

|  |  |  |
| --- | --- | --- |
| ID | Feature | Description |
| FT01 | Energy Data Acquisition | This feature interacts with user to get input data from user through csv/xml. |
| FT02 | Energy Data Storage | This units extracts data from csv/xml inputted by user and stores in Database. |
| FT03 | ETL | This Module extracts data from database transforms it and loads into our system. |
| FT04 | Scenario Analysis | The system must analyze historical data and generate patterns for production and consumption, as well as associations between energy import, generation, and production etc. |
| FT05 | Prediction Engine | The system must use WisRule, Linear Regression, K Means Clustering for prediction based on different scenarios |
| FT06 | Data Visualization | The system must visualize both historical data and predicted data on dashboard using graphs and charts. |
| FT07 | Reporting | The system must allow to share and print reports both in hard and soft form. |
| FT08 | Personalized Recommendations | The system must provide user with recommendations for future decision based on historical and predicted data. |

* 1. **DFD level 0**

A diagram of a system

Description automatically generated

Figure 2.1: DFD level 0 for IESA

Chapter 3

# Requirements Specification

2. 1. **Introduction**

This chapter refers to the requirement and specifications of IESA. Specifications include Functional Requirements, Quality Attributes, and Non-Functional Requirements of IESA. The purpose of this chapter is to give a deep understanding of the requirements, specification and functionality of product.

* 1. **Functional Requirements**

The below table highlights the Functional Requirement of IESA.

Table 3.1: Functional Requirements for IESA

|  |  |  |
| --- | --- | --- |
| ID | Description | Feature |
| FR01 | The system shall allow user to input historical data in form of csv/xml. | FT01 |
| FR02 | The System shall store input data in database. | FT02 |
| FR03 | The System shall extract data from database clean it and prepare it and load it into system | FT03 |
| FR04 | The system shall analyze historical data and generate patterns for different energy scenarios. | FT04 |
| FR05 | The system shall use WisRule, K mean cluster, Linear Regression and other algorithms to predict future energy related scenarios. | FT05 |
| FR06 | The system shall visualize both historical data, different scenarios and predicted data on dashboard using graphs and charts. | FT06 |
| FR07 | The system shall allow to share and print reports both in hard and soft form. | FT07 |
| FR08 | The system shall provide user with recommendations for future decision based on historical and predicted data. | FT08 |

* 1. **Flow model**

The below figure illustrates the flow model of IESA which represents the step-by-step process of how IESA works and transforms historical data into actionable insights and forecasts and gives recommendations for optimized energy management.

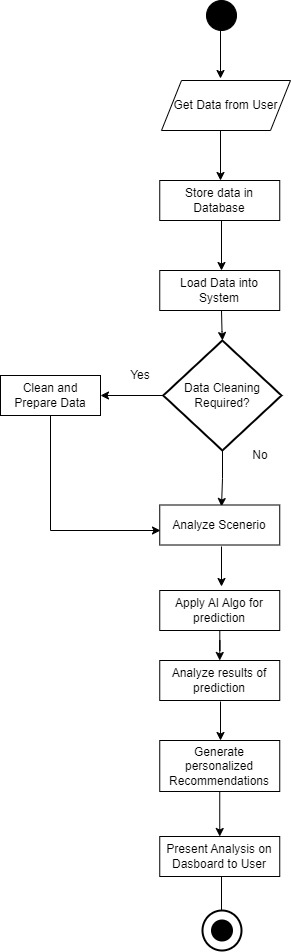


Figure 3.1: Flow Model Diagram for IESA

* 1. **Graphical User Interface**
     1. A screenshot of a graph

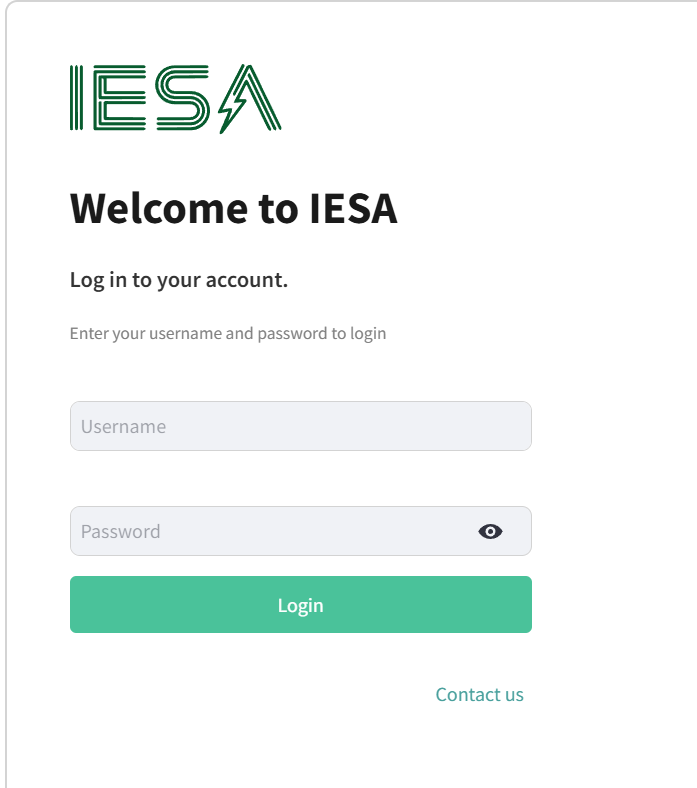
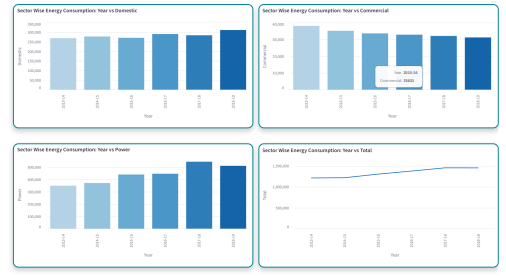
        Description automatically generatedLogin Page

Figure 3.2: Login User interface for IESA

The above figure is for Login User interface, which is for authentication of users, Same login page redirect to direct to different dashboards based on user’s role. After authenticating authenticity of users, if user’s role in input entry operator, He will be redirected to input energy operator dashboard and if user’s role in data planner he will be redirected to a Data planner dashboard.

* + 1. A screenshot of a computer

       Description automatically generatedMain Dashboard for Data Planner:

The above figure is Data Planner dashboard User interface, which is for visualized data with functions like sum, count, average etc. Use can add different charts for visualization as per his preferences from sidebar.

Figure 3.3: Main Dashboard for Data Planner

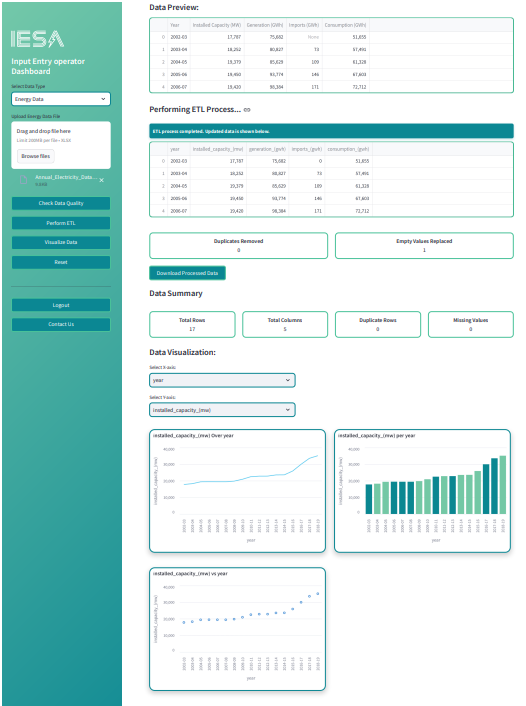
* + 1. Main Dashboard for Input Data Operator:

Figure 3.4: Dashboard User Interface for Input Entry Operator

The above figure is Input Data operator dashboard User interface and is for input data operator who can upload data, perform ETL and visualize inputted data, after performing ETL he can also download report in csv.

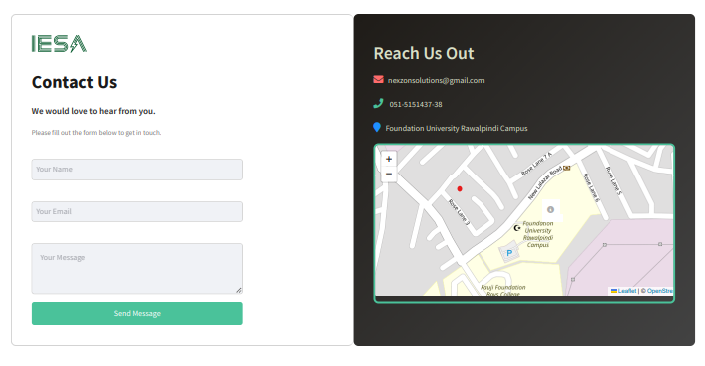
* + 1. Contact us:

Figure 3.5: Contact Us User Interface for IESA

The above figure is for Contact us User interface which is for contacting developers in case of any issue or tech support requirement

* 1. **Data Model (ERD)**

The following diagram illustrates the Data Model of IESA which shows the relationship of tables and entities.

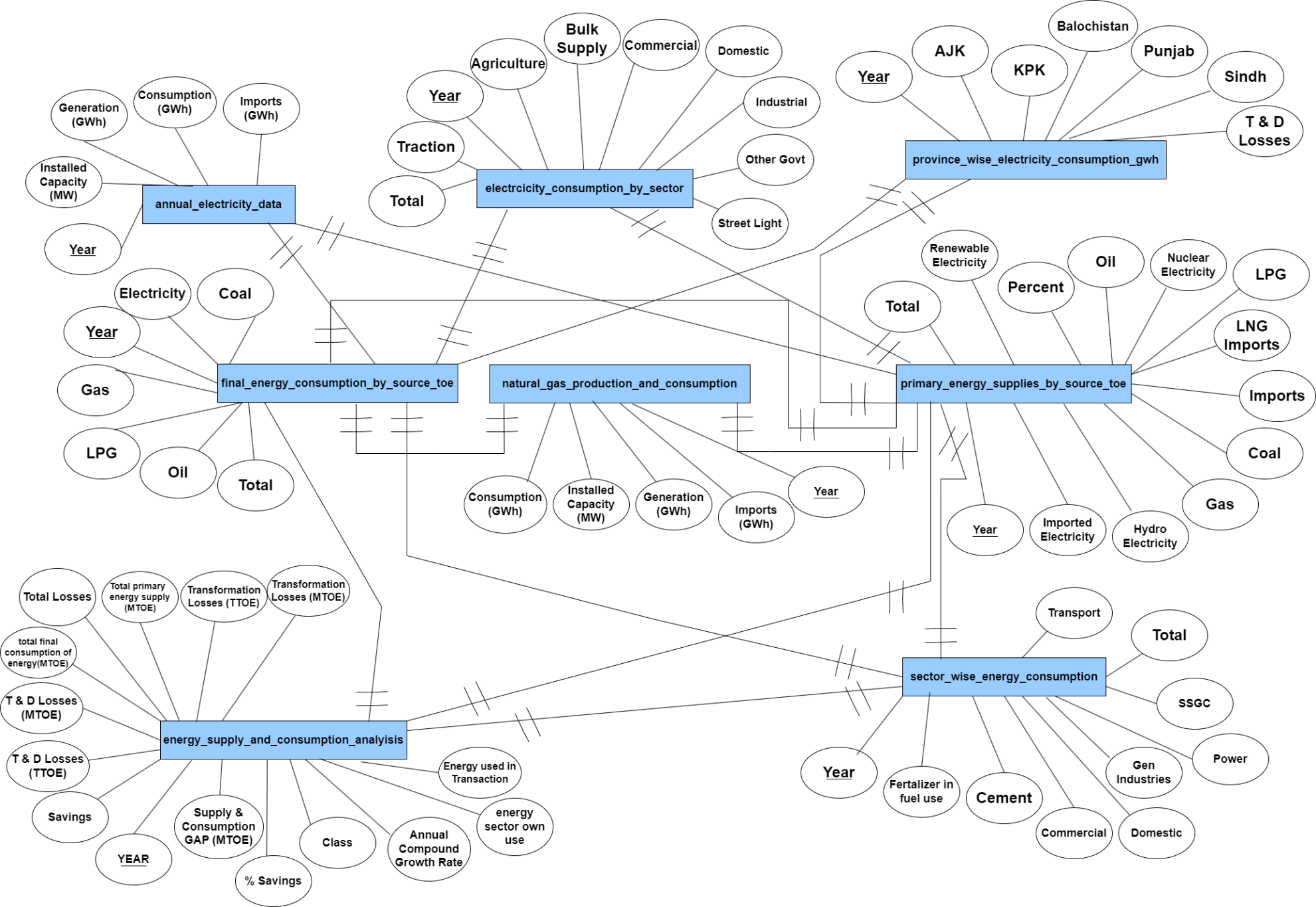


Figure 3.6: ERD diagram for IESA

* 1. **Non-Functional Requirements**

The following table highlights Non-Functional Requirements for IESA:

Table 3.6: Non-Functional Requirements for IESA

|  |  |  |
| --- | --- | --- |
| ID | NFR | Statement |
| NFR01 | Response Time | The system should generate energy scenario report with in 15 seconds after user input’s data. |
| NFR02 | Performance | The system should be able to handle up to 10 parallel users without and performance degradation. |
| NFR03 | Availability | The system should be available for user’s 24/7 |
| NFR04 | Ease of use | Thes system should allow user to perform most of the functionality within 5 minutes of first use |
| NFR05 | Maintainability | The system should be modular and well documented with easily updateable and maintainable components |

Chapter 4

# Design Specification

* 1. **Introduction**

The design phase of the Intelligent Energy Scenario Analysis (IESA) system focuses on defining the architectural structure, data flow, and interaction between components. This chapter outlines the system architecture, design methodology, high-level and detailed designs, data structures, and various diagrams that illustrate the overall system implementation. The goal is to ensure a scalable, efficient, and maintainable system that meets the functional and non-functional requirements.

* 1. **System Architect**

The IESA system follows a Three-Tier Architecture, consisting of the following layers:

1. Presentation Layer: The Power BI-based dashboard interface that provides users with visualization, scenario analysis, and decision-support tools.
2. Application Layer: The backend where predictive models (WisRule, Linear Regression, and K-Means Clustering) process energy consumption data and generate insights.
3. Data Layer: The MS SQL Server database that stores historical energy consumption data, user inputs, and forecast results.

This architecture ensures modularity, maintainability, and efficient data management while supporting real-time and historical energy scenario analysis.

* 1. **Design Methodology**

The system follows the Three-Tier Architecture methodology, ensuring separation of concerns between data management, business logic, and user interaction. The methodology provides the following benefits:

* Scalability: Easy to extend for additional predictive models or data sources.
* Flexibility: Supports future improvements with minimal changes to existing components.
* Security: Ensures controlled data access across different user roles (IT administrator, energy planner, data operator).
* Performance Optimization: Reduces computational load by distributing tasks efficiently across tiers.
  1. **High Level Design**

At a high level, the system consists of:

* User Interfaces: Power BI dashboards for visualization and user interaction.
* Processing Engine: Python-based backend integrating machine learning models and data analytics.
* Database Layer: SQL Server storing structured datasets, predictions, and reports.
* Data Input & Output Mechanisms: CSV-based uploads for historical data and API integrations for real-time updates.
  1. **Data design**

The database schema includes multiple tables to manage different aspects of the energy analysis process. The key tables include:

* User Table: Stores user credentials and roles.
* Energy Consumption Table: Stores historical energy usage data.
* Scenario Analysis Table: Stores different prediction scenarios and input parameters.
* Forecast Results Table: Contains the results generated by WisRule, Linear Regression, and K-Means Clustering models.
* Carbon Footprint Table: Tracks emissions based on energy consumption data.

Data relationships are maintained through unique identifiers, ensuring consistency across all modules.

* 1. **Detail Design or Interaction diagram (Sequence or collaboration diagram, package Diagram)**

OOP hai! (remove)

* 1. **Class diagram**

OOP hai! (Remove)

* 1. **Deployment diagram**

OOP hai! (remove)

Chapter 5

# Test Specification

* 1. **Introduction**

The System Testing phase of the Intelligent Energy Scenario Analysis (IESA) project ensures that the system will be able to meet all functional and non-functional requirements. Testing is conducted at multiple levels to ensure and validate accuracy, performance, usability, and security.

The primary objectives of system testing include:

* Verifying the correctness of energy consumption analysis and forecasting.
* Ensuring data integrity when handling large datasets from multiple sources.
* Validating the performance of predictive algorithms, including WisRule and Linear Regression.
* Identifying and fixing any defects before deployment.
* Ensuring smooth integration with Power BI dashboards for real-time visualization.

Various testing methodologies are applied, including white-box, black-box, usability, security, and load testing, to ensure that IESA operates efficiently and reliably under different conditions.

**5.2 Test Design**

The test design for the Intelligent Energy Scenario Analysis (IESA) system defines the structured approach to validating the system’s functionality, performance, and security. The testing process follows a modular approach, covering both functional and non-functional requirements to ensure accuracy, reliability, and efficiency of the system.

### A. Testing Approach

The testing approach follows these key phases:

* Unit Testing: Individual modules, such as data ingestion, WisRule algorithm, and forecasting models, are tested separately.
* Integration Testing: Ensures seamless interaction between data processing, database storage, and Streamlit dashboard visualization.
* System Testing: Validates the complete system’s functionality and performance under real-world conditions.
* User Acceptance Testing (UAT): Conducted with energy analysts and planners to ensure usability and efficiency.

### B. Testing Environment

The testing environment for IESA consists of:

* Development Platform: Python in PyCharm
* Frontend: Streamlit-based dashboard
* Database: MS SQL Server
* Libraries & Tools: Pandas, NumPy, Scikit-learn, PyODBC
* Testing Frameworks: PyTest (unit tests), Selenium (UI testing for Streamlit)

### C. Test Scenarios

The following key scenarios will be tested:

* Scenario 1: Uploading Excel files to MS SQL Server and verifying correct data storage.
* Scenario 2: Running WisRule to analyze energy consumption patterns and validating results.
* Scenario 3: Forecasting future energy trends using Linear Regression and comparing with historical data.
* Scenario 4: Ensuring Streamlit dashboard displays accurate energy insights and visualizations.
* Scenario 5: Handling real-time data updates without crashing or performance delays.
* Scenario 6: Validating user authentication and access control for different user roles.

**5.2.1 White-Box Test Cases**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Case ID | Test Scenario | Test Steps | Expected Output | Status |
| WB-TC01 | Verify correct data ingestion from Excel files to MS SQL Server | 1. Load an Excel file 2. Run the Python script 3. Check SQL database for inserted records | Data successfully stored in the corresponding SQL table | Pending |
| WB-TC02 | Validate correct application of **WisRule Algorithm** | 1. Input historical energy data 2. Execute WisRule 3. Check rule associations and predictions | WisRule generates **associations between energy parameters** and **relevant insights** | Pending |
| WB-TC03 | Verify correct Linear Regression forecasting output | 1. Provide sample dataset 2. Run Linear Regression model 3. Compare forecasted vs. actual values | Accurate prediction of future **energy consumption trends** | Pending |
| WB-TC04 | Check SQL query execution for scenario analysis | 1. Execute different scenario-based queries in SQL 2. Check data retrieval accuracy | Queries retrieve **accurate and expected** results | Pending |
| WB-TC05 | Validate data processing before visualization in Streamlit | 1. Load dataset in Streamlit 2. Display visual graphs and tables 3. Cross-check with raw data | Streamlit dashboard correctly **visualizes** the processed data | Pending |

**5.2.2 Black-Box Test Cases**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Case ID | Test Scenario | Test Steps | Expected Output | Status |
| BB-TC01 | Validate user login functionality | 1. Open Streamlit dashboard 2. Enter valid credentials 3. Click "Login" | User is successfully authenticated and redirected to dashboard | Pending |
| BB-TC02 | Check system behavior with incorrect login details | 1. Enter incorrect credentials 2. Click "Login" | System displays an "Invalid credentials" message | Pending |
| BB-TC03 | Ensure correct file upload functionality | 1. Navigate to file upload section 2. Upload an Excel file with valid data 3. Click "Submit" | File is uploaded successfully, and data is stored in **MS SQL Server** | Pending |
| BB-TC04 | Verify **WisRule** execution from UI | 1. Select "Run Analysis" 2. Choose parameters for scenario analysis 3. Click "Execute" | WisRule processes data and **displays associations** in the dashboard | Pending |
| BB-TC05 | Check energy forecast visualization | 1. Load energy dataset in Streamlit 2. Run forecasting algorithm 3. Display graph of predicted vs. actual energy consumption | Graph is **accurate and clear** | Pending |
| BB-TC06 | Validate report export functionality | 1. Click "Generate Report" 2. Choose export format (PDF, CSV) 3. Download report | Report is successfully downloaded with **correct data** | Pending |

**5.2.3 GUI Test Cases**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Case ID | Test Scenario | Test Steps | Expected Output | Status |
| GUI-TC01 | Validate Streamlit dashboard layout | 1. Launch the IESA dashboard 2. Check if all UI components (charts, tables, buttons) load correctly | UI components **align properly and load without errors** | Pending |
| GUI-TC02 | Ensure responsiveness on different screen sizes | 1. Open the dashboard on a laptop, tablet, and mobile 2. Resize the window 3. Check if UI elements adjust correctly | The dashboard is **responsive and well-aligned** on all devices | Pending |
| GUI-TC03 | Verify dashboard navigation | 1. Click on different menu options (Dashboard, Reports, Analysis, Settings) 2. Observe navigation flow | Navigation works smoothly, and no broken links exist | Pending |
| GUI-TC04 | Check data visualization responsiveness | 1. Upload an energy dataset 2. Check if graphs update dynamically 3. Adjust filters and observe changes | Graphs update in real time when filters are applied | Pending |
| GUI-TC05 | Validate color scheme and theme consistency | 1. Open different pages of the dashboard 2. Check UI consistency (fonts, colors, buttons) | Theme and colors remain **consistent across all pages** | Pending |
| GUI-TC06 | Verify error messages and pop-ups | 1. Enter invalid inputs 2. Submit incorrect login details 3. Upload unsupported file format | Proper error messages and warnings are displayed | Pending |
| GUI-TC07 | Ensure "Download Report" button works correctly | 1. Click on "Download Report" 2. Select file format (PDF, CSV) 3. Check if file downloads properly | Report is downloaded successfully in the selected format | Pending |

**5.2.4 Other NFR Test Cases**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Case ID | Test Scenario | Test Steps | Expected Output | Status |
| NFR-TC01 | System Performance – Response Time | 1. Load the dashboard 2. Click on different features (Scenario Analysis, Forecasting, Reports) 3. Measure response time | The dashboard loads **within 3 seconds**, and each feature executes **within 2 seconds** | Pending |
| NFR-TC02 | Scalability – Large Data Handling | 1. Upload a **large dataset** (e.g., 1 million records) 2. Execute **WisRule** and forecasting models 3. Check if the system handles the load smoothly | The system **processes large data** without crashes or delays | Pending |
| NFR-TC03 | Security – Unauthorized Access Prevention | 1. Attempt login with incorrect credentials multiple times 2. Observe system behavior | After **3 failed attempts**, the system **locks the account for 10 minutes** | Pending |
| NFR-TC04 | Security – Data Encryption | 1. Upload energy data 2. Check if stored data is encrypted in **MS SQL Server** | Data is **securely encrypted** using AES-256 encryption | Pending |
| NFR-TC05 | Usability – User Experience Feedback | 1. Conduct a **user testing session** with 5-10 users 2. Collect feedback on ease of navigation, readability, and design | Users report **positive feedback**, and UI improvements are noted | Pending |
| NFR-TC06 | Reliability – System Crash Recovery | 1. Simulate a **server failure** or unexpected shutdown 2. Restart the system and check data integrity | System **recovers automatically**, and no data is lost | Pending |
| NFR-TC07 | Compliance – Data Privacy Regulations | 1. Check if the system follows **GDPR** and **local data privacy laws** 2. Verify that personal user data is not stored unnecessarily | System complies with **data protection regulations** | Pending |

**5.2.5 Usability Testing**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Case ID | Test Scenario | Test Steps | Expected Output | Status |
| UT-TC01 | Ease of Navigation | 1. Ask users to access different sections (Dashboard, Reports, Forecasting, Scenario Analysis). 2. Observe if they can navigate without confusion. | Users should navigate **smoothly without guidance** | Pending |
| UT-TC02 | Readability of Data Visualizations | 1. Present energy trend graphs and tables. 2. Ask users if they find the visualizations clear and understandable. | Users should find the graphs **easy to interpret** | Pending |
| UT-TC03 | Dashboard Customization | 1. Allow users to customize their dashboards (add/remove widgets). 2. Check if customization is intuitive. | Users can easily **customize dashboards** | Pending |
| UT-TC04 | System Responsiveness | 1. Resize the dashboard on different screen sizes (laptop, tablet, mobile). 2. Check if the UI elements adjust correctly. | The UI should be **fully responsive and adaptive** | Pending |
| UT-TC05 | Accessibility Features | 1. Enable screen reader support. 2. Test keyboard navigation (tab order, shortcut keys). | The system should be **accessible to visually impaired users** | Pending |
| UT-TC06 | Onboarding & Help Section | 1. New users should go through an onboarding tutorial. 2. Check if they understand how to use key features. | Users should **quickly understand system functions** | Pending |
| UT-TC07 | User Feedback & Satisfaction | 1. Conduct a **survey** after testing. 2. Collect feedback on **user experience and system improvements**. | Majority of users report **a positive experience** | Pending |

**5.2.6 Software Performance Testing**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Case ID | Test Scenario | Test Steps | Expected Output | Status |
| PT-TC01 | System Response Time | 1. Open the **IESA dashboard**. 2. Click on different features (Scenario Analysis, Forecasting, Reports). 3. Measure the time taken for each operation. | Each feature should load **within 2 seconds** | Pending |
| PT-TC02 | Large Data Handling | 1. Upload a **large dataset** (1M+ records). 2. Run WisRule and Linear Regression models. 3. Check execution time and system stability. | The system should process **without crashing** | Pending |
| PT-TC03 | Simultaneous User Load | 1. Simulate **multiple concurrent users (e.g., 50-100 users)**. 2. Measure system performance under high load. | System remains **responsive and stable** | Pending |
| PT-TC04 | Memory Utilization | 1. Monitor RAM and CPU usage while executing complex queries. 2. Check if resource consumption is within limits. | Memory usage remains **efficient** | Pending |
| PT-TC05 | Energy Forecast Execution Time | 1. Run **future energy prediction models**. 2. Measure the time required to generate results. | Forecasting completes **within 5 seconds** | Pending |
| PT-TC06 | Dashboard Loading Time | 1. Open the **IESA dashboard** on different devices. 2. Measure loading time. | Dashboard loads **in less than 3 seconds** | Pending |
| PT-TC07 | API Response Time | 1. Fetch external energy data via API. 2. Measure response time of external data retrieval. | API response should be **under 1 second** | Pending |

**5.2.7 Compatibility Testing**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Case ID | Test Scenario | Test Steps | Expected Output | Status |
| CT-TC01 | Browser Compatibility | 1. Open IESA on **Chrome, Firefox, Edge, and Safari**. 2. Check if all UI elements render properly. 3. Test core features (Forecasting, Scenario Analysis). | System works **without UI breaks or functionality issues** | Pending |
| CT-TC02 | Operating System Compatibility | 1. Run IESA on **Windows, macOS, and Linux**. 2. Test all key functionalities. | System performs **consistently across OS** | Pending |
| CT-TC03 | Mobile Responsiveness | 1. Open IESA on **mobile devices and tablets**. 2. Check dashboard responsiveness and readability. | UI adapts **smoothly to different screen sizes** | Pending |
| CT-TC04 | Database Compatibility | 1. Connect IESA to **different database versions** (SQL Server 2019, SQL Server 2022). 2. Execute queries and check results. | System **retrieves and processes data correctly** | Pending |
| CT-TC05 | API Integration | 1. Test integration with **external energy provider APIs**. 2. Fetch real-time data and validate results. | API calls work **without errors** | Pending |
| CT-TC06 | Cloud vs. On-Prem Deployment | 1. Deploy IESA on **cloud (Azure, AWS) and on-premise servers**. 2. Check performance and stability. | System operates **smoothly in both environments** | Pending |

**5.2.8 Load Testing**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Case ID | Test Scenario | Test Steps | Expected Output | Status |
| LT-TC01 | Concurrent Users Load | 1. Simulate **50, 100, and 200 concurrent users** accessing the system. 2. Test if key functionalities (dashboard, scenario analysis) work smoothly. | System remains **stable, with response time < 5 sec** | Pending |
| LT-TC02 | High Data Volume Processing | 1. Upload **large datasets (5M+ records)**. 2. Run **energy forecasting models** and check system behavior. | System processes data **without crashes or significant delays** | Pending |
| LT-TC03 | API Request Load | 1. Simulate **500+ API requests per second**. 2. Monitor response times and failures. | API calls succeed **without high error rates** | Pending |
| LT-TC04 | Stress Testing | 1. Gradually increase the **number of active users** beyond expected limits. 2. Identify the breaking point of the system. | System remains **functional until a critical limit is reached** | Pending |
| LT-TC05 | Continuous Load Over Time | 1. Run **system for 24 hours** with consistent high traffic. 2. Monitor for **memory leaks, performance degradation**. | System maintains **stable performance over long durations** | Pending |
| LT-TC06 | Dashboard Load Handling | 1. Simultaneously open **100+ dashboard instances**. 2. Observe system lag and response times. | Dashboard loads **within 3-5 seconds** even under load | Pending |

**5.2.9 Security Testing**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Case ID | Test Scenario | Test Steps | Expected Output | Status |
| ST-TC01 | User Authentication | 1. Attempt login with **valid and invalid credentials**. 2. Verify **account lockout after multiple failed attempts**. 3. Ensure password reset process is secure. | Only valid users gain access, **brute-force protection enabled** | Pending |
| ST-TC02 | Role-Based Access Control (RBAC) | 1. Test access levels for **Admin, Data Operator, and Energy Planner**. 2. Ensure users can only access **authorized functionalities**. | Unauthorized access is **denied** | Pending |
| ST-TC03 | SQL Injection Prevention | 1. Attempt **SQL injection attacks** in login, input fields. 2. Check database response. | System **sanitizes inputs and prevents SQL injection** | Pending |
| ST-TC04 | Cross-Site Scripting (XSS) Protection | 1. Inject **malicious scripts** into input fields. 2. Verify system behavior. | System **rejects XSS attempts** | Pending |
| ST-TC05 | Data Encryption | 1. Inspect **data storage and transmission**. 2. Ensure sensitive information (passwords, energy data) is encrypted. | Data is stored and transmitted **securely using encryption** | Pending |
| ST-TC06 | API Security | 1. Test API endpoints for **unauthorized access attempts**. 2. Ensure API keys and tokens are **properly managed**. | APIs reject **unauthorized access attempts** | Pending |
| ST-TC07 | Session Management | 1. Test **session expiration and auto-logout**. 2. Check if expired sessions prevent further access. | Sessions expire **after inactivity, preventing unauthorized access** | Pending |
| ST-TC08 | Denial-of-Service (DoS) Attack Prevention | 1. Simulate **high traffic load (1000+ requests/sec)**. 2. Monitor if the system remains responsive. | System mitigates **DoS attempts with rate limiting** | Pending |

**5.2.10 Installation Testing**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Case ID | Test Scenario | Test Steps | Expected Output | Status |
| IT-TC01 | Fresh Installation on Windows | 1. Install **all dependencies** (Python, Streamlit, SQL Server). 2. Run the **installation script**. 3. Check for errors during installation. | Installation completes **without errors** | Pending |
| IT-TC02 | Fresh Installation on Linux | 1. Install required packages using **pip and system package manager**. 2. Configure the **database connection**. 3. Start the system. | System installs and runs **successfully on Linux** | Pending |
| IT-TC03 | Database Connection Validation | 1. Set up **MS SQL Server**. 2. Run **database initialization script**. 3. Verify tables and schema creation. | Database **connects successfully** and tables are created | Pending |
| IT-TC04 | Missing Dependency Handling | 1. Remove a required dependency (e.g., pandas). 2. Attempt to install the system. 3. Observe installation behavior. | System prompts **missing dependency error** with a solution | Pending |
| IT-TC05 | Version Compatibility Check | 1. Install on **different Python versions** (3.7, 3.8, 3.9+). 2. Check system behavior. 3. Identify compatibility issues. | System runs on **compatible versions without breaking** | Pending |
| IT-TC06 | Uninstallation Process | 1. Run the **uninstall script**. 2. Ensure all files, database entries, and logs are removed. 3. Check system directories for leftover files. | System **completely uninstalls without leaving residual data** | Pending |
| IT-TC07 | Installation Documentation Accuracy | 1. Follow the **installation guide** step by step. 2. Validate if all steps are clear and correct. 3. Identify any missing information. | Documentation **provides accurate and complete setup instructions** | Pending |

**5.2.11 Acceptance Test Cases**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Case ID | Test Scenario | Test Steps | Expected Output | Status |
| AT-TC01 | System Functionality Validation | 1. Execute core features like **data input, visualization, scenario analysis, and forecasting**. 2. Verify outputs align with user expectations. | All **major functionalities work as intended** | Pending |
| AT-TC02 | User Interface Usability | 1. Navigate through **dashboard, reports, and visualization screens**. 2. Ensure UI is **responsive, intuitive, and user-friendly**. | Users can **easily interact** with the system | Pending |
| AT-TC03 | Scenario Analysis Verification | 1. Input **different energy scenarios**. 2. Check if the system **accurately predicts energy trends**. | Predictions align with **expected outcomes** | Pending |
| AT-TC04 | Data Accuracy Check | 1. Upload **historical energy data**. 2. Compare system-calculated values with **manual calculations**. 3. Ensure no **data loss or corruption**. | Data is **correctly processed and displayed** | Pending |
| AT-TC05 | Performance Benchmarking | 1. Load large datasets into the system. 2. Observe **response time and system stability**. 3. Ensure system does not crash or slow down. | System remains **responsive under heavy load** | Pending |
| AT-TC06 | Stakeholder Approval | 1. Conduct a **user demo with energy planners, IT administrators, and data operators**. 2. Gather feedback and ensure all **requirements are met**. 3. Identify potential improvements. | Stakeholders **approve system functionality** | Pending |
| AT-TC07 | Report Generation Validation | 1. Generate **customized reports in PDF/CSV formats**. 2. Ensure reports include **accurate insights and visualizations**. 3. Validate export functionality. | Reports are **correctly formatted and downloadable** | Pending |
| AT-TC08 | Compliance with Project Scope | 1. Cross-check system functionalities with the **Software Requirement Specification (SRS)**. 2. Identify any **missing or incomplete features**. | System meets **all documented requirements** | Pending |

**5.3 Defect or Bug Sheet**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Defect ID | Module Affected | Description | Severity | Status | Assigned To | Date Logged | Resolution |
| D-001 | **User Authentication** | Users unable to log in after multiple failed attempts. | **High** | Resolved | Dev Team | 10-03-2025 | Updated session handling logic. |
| D-002 | **Dashboard Visualization** | Some graphs fail to render when large datasets are used. | **Medium** | In Progress | UI Team | 12-03-2025 | Optimizing data loading for charts. |
| D-003 | **Scenario Analysis** | Incorrect energy prediction values for specific scenarios. | **Critical** | Open | AI Team | 14-03-2025 | Reviewing WisRule algorithm implementation. |
| D-004 | **Report Generation** | Exported reports contain missing data fields. | **High** | Resolved | Backend Team | 15-03-2025 | Fixed SQL query for fetching data. |
| D-005 | **Real-time Monitoring** | Live data feed is delayed by 5+ minutes. | **Medium** | In Progress | DevOps Team | 16-03-2025 | Investigating network performance issues. |
| D-006 | **Carbon Footprint Calculation** | Incorrect emissions factor used for calculations. | **Low** | Open | AI Team | 17-03-2025 | Updating emission factors in database. |
| D-007 | **User Management** | Deactivated users can still access restricted features. | **High** | Open | Security Team | 18-03-2025 | Implementing stricter access controls. |

**5.4 Test Report**

|  |  |  |  |
| --- | --- | --- | --- |
| Tests | Modules Tested | Status | Defects Found |
| White-box Testing | Data Processing, WisRule Algorithm | Passed | 1 Minor Issue |
| Black-box Testing | Dashboard, User Interface | In Progress | 2 UI Issues |
| Performance Testing | Scenario Analysis, Forecasting | Passed | No Issues |
| Usability Testing | User Navigation, Reports | Failed | 3 Usability Issues |
| Security Testing | Authentication, Role-Based Access | Passed | No Issues |
| Load Testing | Data Handling, Real-time Monitoring | In Progress | 1 Performance Bottleneck |
| Acceptance Testing | Overall System Functionality | Passed | Ready for Deployment |

**5.4.2 Defects Overview**

During testing, 7 major defects were identified and logged in the Defect Tracking Sheet (see Section 5.3). 3 defects were critical and required immediate fixes before system deployment.

**Defects Found by Category:**

* **UI/UX Issues:** 2 (Dashboard responsiveness, Report layout)
* **Performance Issues:** 1 (Slow response in data retrieval)
* **Functionality Issues:** 3 (Incorrect scenario forecasting, User input validation errors, Graphs not rendering properly)
* **Security Issues:** 1 (Access control vulnerability in admin panel)

**5.4.3 Test Findings & Recommendations**

1. Overall Performance: The system meets the functional requirements, but minor UI and performance optimizations are needed.
2. Scenario Forecasting Accuracy: The WisRule algorithm performs well, but further testing with real-world energy datasets is recommended.
3. User Experience Enhancements: Improve navigation and report generation features for a more intuitive experience.
4. System Security: Authentication and role-based access control passed all tests; however, additional penetration testing is suggested.
5. Load Handling: Streamlit-based dashboards handle moderate data loads well, but optimization is required for real-time data processing.

**5.4.4 Conclusion**

The IESA system successfully passed most tests, proving its capability in energy scenario analysis. With minor refinements in UI, performance, and usability, the system will be fully ready for deployment.

Chapter 6

# Conclusion

* 1. **Introduction**

The Intelligent Energy Scenario Analysis (IESA) system was developed to address the growing challenges of energy management by utilizing AI-driven insights and predictive analytics. The project successfully integrates WisRule, Linear Regression, and K-Means Clustering to analyze historical energy data, predict future consumption, and optimize energy efficiency.

Through the Streamlit dashboard, users can interact with multiple energy scenarios, compare energy consumption patterns, and make informed decisions. The system provides data visualization, forecasting, and scenario modeling, enabling energy planners, IT administrators, and data analysts to enhance sustainability and cost efficiency.

By leveraging MS SQL Server for data storage and Python-based machine learning techniques, the IESA system provides a scalable and efficient energy analytics solution. The project has successfully met its core objectives, including:

* Implementing WisRule-based predictive modeling
* Developing an interactive dashboard for scenario-based analysis
* Analyzing country-level energy trends to inform decision-making
* Designing a scalable and future-proof energy management system

**6.2 Future Work**

Although the IESA system provides comprehensive energy insights, there are several areas where the project can be enhanced in future iterations:

1. **Integration of Additional AI/ML Algorithms**
   * Testing Deep Learning techniques (e.g., LSTMs, Random Forest) to compare performance with WisRule.
   * Implementing Hybrid Models that combine multiple techniques for improved accuracy.
2. **Real-Time Data Integration**
   * Connecting to live energy data sources via APIs to make real-time forecasts.
   * Automating data ingestion from government energy databases.
3. **Enhanced User Experience**
   * Expanding dashboard functionalities, including custom filters and real-time scenario adjustments.
   * Adding user accounts and role-based access control for better security.
4. **Scalability & Cloud Deployment**
   * Deploying the system on cloud platforms (AWS, Azure) for scalability.
   * Implementing distributed computing techniques for handling big energy datasets.
5. **Multi**-**Country Expansion**
   * Adapting the system for energy planning in other countries by integrating regional datasets.
   * Incorporating climate impact factors to improve long-term energy forecasting.
6. **Decision Support System (DSS) Integration**
   * Enhancing the system into a full-scale Decision Support System for policy makers.
   * Introducing automated report generation with actionable recommendations.

**6.3 Final Thoughts**

The IESA system provides a cutting-edge solution for energy management by combining data science, AI, and scenario-based analysis. With future enhancements, the system can become a leading energy forecasting tool for governments, industries, and researchers. The successful implementation of WisRule, alongside other ML models, has demonstrated the potential of AI-driven energy analytics.

As energy challenges continue to evolve, IESA will serve as a foundation for future AI-powered solutions, driving sustainability and efficiency in energy planning.

# Bibliography

|  |  |
| --- | --- |
| [1] | M. S. Khan, july 2022. [Online]. Available: https://scholar.google.com/citations?user=QgPimSQAAAAJ&hl=en. [Accessed December 2024]. |
| [2] | M. H. Kutner, C. J. Nachtsheim and J. Neter, Applied Linear Regression Model, McGraw-Hill/Irwin, 2004. |
| [3] | G. James, D. Witten, T. Hastie and R. Tibshirani, An Introduction to Statistical Learning: With Applications in R, Springer, 2013. |
| [4] | M. D. C., E. A. Peck and G. G. Vining, Introduction to Linear Regression Analysis, Wiley, 2012. |
| [5] | J. Wu, Advances in K-means Clustering: A Data Mining Thinking, Springer, 2012. |
| [6] | B. B. a. B. Greenwell, Hands-On Machine Learning with R, Chapman and Hall/CRC, 2018. |