

Software Requirements Specification

for

POWER SAVER HQ

Final Version

Prepared by Project X-celerate

Syracuse University CSE687

April 25, 2023

Table of Contents

1	Introduction	4
1.1	Purpose	4
1.2	Scope	4
1.3	Definitions	4
1.4	References	5
1.5	Overview	5
2	Overall Description	6
2.1	Product Perspective	6
2.2	Product Functions	7
2.3	Use Case Descriptions	8
3	Specific Requirements	12
3.1	Overview	12
3.2	Interface Requirements	12
3.3	System Features	13
3.4	Performance Requirements	20
3.5	Design Constraints	20
3.6	Software System Attributes	21
3.7	Non-functional Requirements	21

Table of Figures

Figure 1	System Block Diagram	6
Figure 2	Generic System Use Case	7
Figure 3	Registration Characteristic System Use Case	9
Figure 4	Billing Characteristic System Use Case	10
Figure 5	Settings Characteristic System Use Case	11
Figure 6	System architecture	13
Figure 7	Login Page Activity Diagram	14
Figure 8	Home Page Activity Diagram	16
Figure 9	Billing Page Activity Diagram	17
Figure 10	Notification Page Activity Diagram	18

Figure 11 Energy Insights Page Activity Diagram	19
Figure 12 Overall Use Case	21
Figure 13 Class Diagram	22
Figure 14 Sequence Diagram	23

Table of Tables

Table 1 Acronyms and Definitions	4
----------------------------------	---

Name	Date	Reason for Change	Version
Team Xccelerate	3/23	Initial revision	1
	3/25	UML Diagrams added	1
	3/26	Requirements added	1
	3/27	Version 1 final draft complete	1
	4/13	UML diagram changes	2
	4/17	Class diagram,sequence diagram	2
	4/24	Version 2 final draft complete	2

1 Introduction

1.1 Purpose

The Power Saver HQ is a platform designed to help customers reduce their energy consumption and save on their electricity bills. The Power Saver platform will offer customers a range of features and functionalities to help them manage their energy consumption, including personalized tips and recommendations, billing and payment history, customer support, and more.

1.2 Scope

The centralized idea for this app can be summarized in following points:

1. Login to the website and see a personalized dashboard.
2. User friendly interface for easy navigation.
3. Provide Energy consumed and its price.
4. Secure Payment Gateway.
5. User Energy Consumption Insights.
6. Comparing Energy Consumption with their neighbors.
7. Previous Energy consumption details for comparison.
8. Ways to save energy are recommended.
9. Daily heater temperature suggestions based on data obtained from weather information via notifications.
10. Notify users of power outages.

1.3 Definitions

Table 1 Acronyms and Definitions

BDD	Block Definition Diagram.
IA	Information Assurance.
IP	Internet Protocol.

SRS	Software Requirements Specification.
STIGS	Security Technical Implementation Guides
TCP	Transmit Control Protocol
UDP	User Datagram Protocol.
UML	Unified Modeling Language.
V&V	Verification and Validation.

1.4 References

1. IEEE Std 830-1998 - IEEE Recommended Practice for Software Requirements Specifications - Revision of IEEE Std 830-1993

1.5 Overview

This document follows the recommended format specified in IEEE Std 830-1998 IEEE Recommended Practice for Software Specifications. For Section 3, the specific template A.5 for organizing information by feature is followed.

2 Overall Description

2.1 Product Perspective

The Power Saver HQ system is intended to be an application to facilitate homeowners (referred as customers/users hereafter in this document) to create an account in the system and perform various operations ranging from viewing their electricity bill, seeing in-depth analysis of energy consumption, and trends among neighbors. This document specifies capabilities for a user to register into the system, add their home address and access the system and utilize all the features provided by the system to ease payments, manage bills and view insights of their energy consumption to promote better energy management. The document describes various capabilities of the system in detail. Figure 1 System Block Diagram shows the system overview, using a Unified Modeling Language (UML) Block Definition Diagram (BDD).

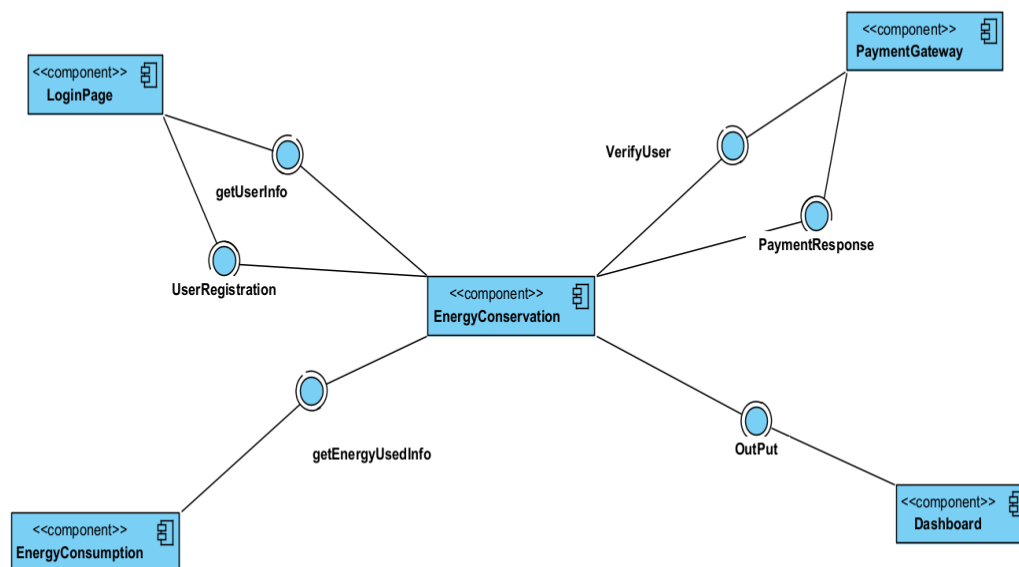


Figure 1- System Block Diagram

2.2 Product Functions

The following use case diagram depicts the users of the system, and the intended way in which they will interact with the system.

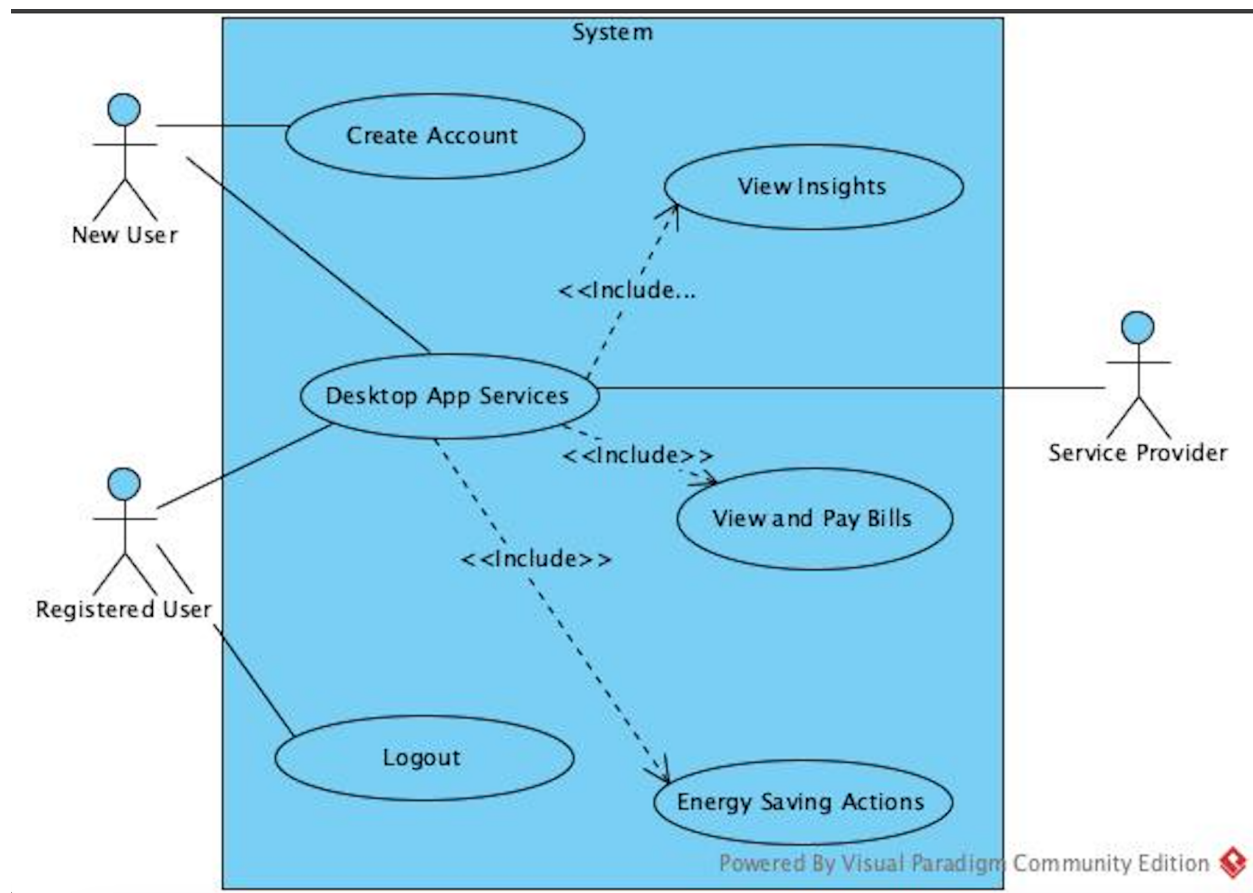


Figure 2- Generic System Use Case

2.3 Use Case Descriptions

- User creates a new profile/sign in
- User adds addresses
- User adds appliances usage info
- User edits notification preferences
- User views bills history
- User makes bill payment
- User views energy usage insights
- User views appliances usage insights

2.3.1 Overall Use case description

General Characteristics	
Intent	Allow the Customer to log in/create an account and provide insights on energy conservation.
Scope	Power Saving
Primary Actor	Customer
Secondary Actors	Service Provider
Preconditions	User owns/lives in a house in the area to which the system provides this service
Assumptions	Each house has an energy connection from the provider.
Trigger	Successful completion of Use Case Pay With Credit Card Reader or Use Case Pay With Mobile App
Success Post Condition	The customer is provided with useful insights on saving power and hence saving money.
Failed Post Condition	The customer is not getting any useful insights to save on power and money.

2.3.2: Registration Use Case

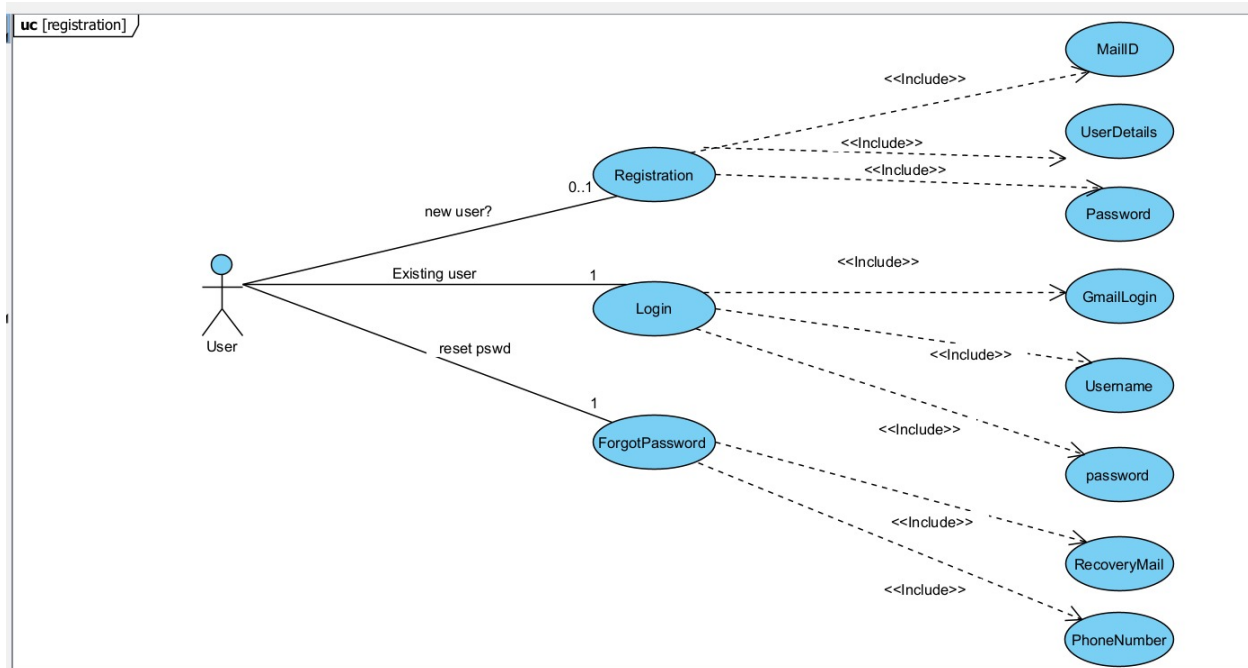


Figure 3-Registration Characteristic System Use Case

Step	Action
Start	This scenario begins when the user has initiated logging in to the portal, irrespective of new or old customer.
1	The Customer is instructed by the User Interface to select the options from Login, Register or Forget Password.
2	The Customer is instructed by the User Interface to Input User name and Password if Login is selected or gmail login option is given.
3	The Customer is instructed by the User Interface to provide all the basic details such as First Name, Last Name, Email ID, Mobile Number, House Address, etc. if Register is selected.
4	The Customer is instructed by the User Interface to provide an authenticated method such as Recovery mail , phone number if the Forgot Password is selected.
5	Finally, the scenario ends when the user successfully logs in to the system.

2.3.3: Billing Use Case

uc [BillingPortal]

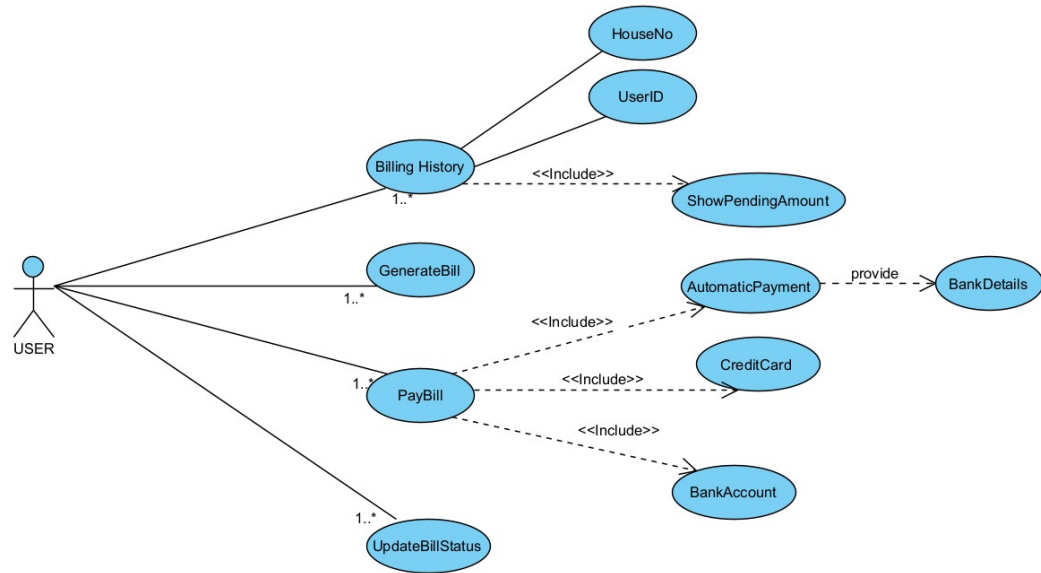


Figure 4- Billing Characteristic System Use Case

Step	Action
Start	This scenario begins when the user has initiated the billing and payment option from the Dashboard.
1	The customer is provided with the History of previous bills if Bill History option is selected.
2	The customer is provided with the Current bill if the Generate Bill option is selected.
3	In this scenario, the customer will be provided with multiple payment options if it chooses the Pay Bill option.
4	Once the payment is done, the status will automatically be updated on the Dashboard.

5	Finally, the scenario ends when the user successfully pays the bill or gets to know the status of it.
---	---

2.3.4: Setting Use case

uc [settings] /

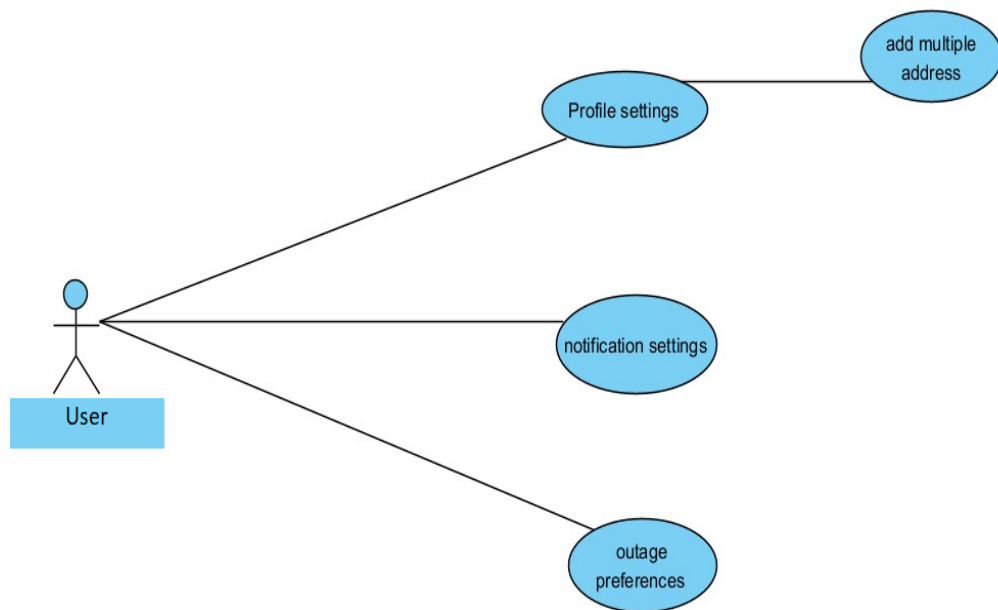


Figure 5- Settings Characteristic System Use Case

Step	Action
Start	This scenario begins when the user has initiated the settings option from the Dashboard.
1	The customer is taken to the Personal details settings option if the Profile Settings option is chosen. They get the option to add multiple addresses on one account.

2	The customer is taken to Notification settings options if the Notification Settings option is chosen where one can get many options to set alerts and their timings and frequency, all personalized.
3	The customer is given the option of how he wants to get notified in case of power outage.
4	Finally, the scenario ends when the user successfully explores all the options of settings.

3 Specific Requirements

3.1 Overview

The purpose of this system is to provide a centralized platform for customers to access information and manage their energy and utility services for all of their home addresses. The system provides services such as bill payment and management, energy usage tracking, outage reporting, and customer support. Additionally, the system provides deep insights and information on energy-saving initiatives to promote sustainability and environmental responsibility. The goal of the system is to enhance customer experience by providing convenient, efficient, and accessible services that meet the evolving needs of its users.

3.2 Interface Requirements

3.2.1: User Interfaces

- 3.2.1.1 The interface **shall** allow homeowners to register with the system, which in turn will allow them to login into their account. The user **will** have to add their home address while registering.
- 3.2.1.2 The interface **shall** have an option to Forget Password, if the user forgets the old password.
- 3.2.1.3 After logging in, an elaborative display **will** show all the information and take the user to the homepage, which provides the user with multiple options.
- 3.2.1.4 The user has the option to view their latest generated bill, previous bills and usage history.
- 3.2.1.5 The user shall be able to access their energy consumption statistics in detail.
- 3.2.1.6 The user **shall** be able to view low energy consumed neighbors energy usage through energy insights and ways to save energy **will** be generated accordingly.
- 3.2.1.7 The user will be able to save up both their energy used and cost of the bills generated by identifying the ideal settings for the appliances.

3.2.1.8 Residents also have an option of receiving alerts and notifications from the system, based on their preference.

3.2.2: Software Interfaces:

3.2.2.1: The software **shall** support Transmit Control Protocol (TCP) for communicating between the REST APIs and the User Interface.

3.2.2.2: The software **shall** interface with an SQL database for the purpose of storing all the required information related to the users, profiles, account preferences their energy consumption etc.

3.3 System Features

System Architecture

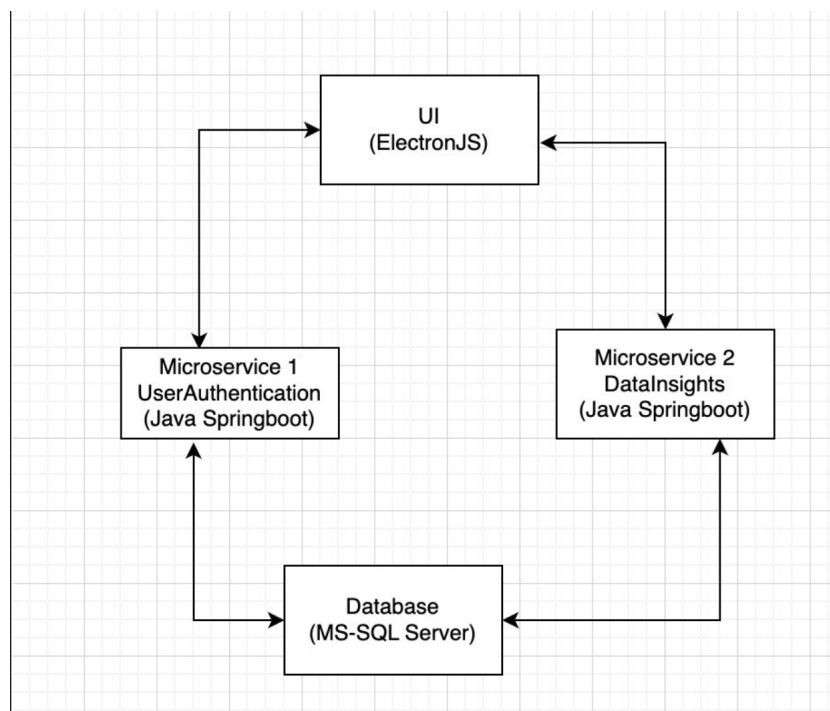


Figure 6- System architecture

Our desktop application will be made using a combination of Vue.JS & Electron.JS to make it compatible with both websites and a native desktop application. We have two microservices in this project, the UserAuthentication Microservice and the DataInsights Microservice which will both be utilizing JAVA Spring Boot. Our Database will be Microsoft SQL Server.

3.3.1 The system software supports the Use Cases described in Figure 2 Overall System Use Cases.

3.3.1.1 UserAuthentication Microservice

3.3.1.1.1 Introduction/Purpose of Feature

This is our primary microservice which will contain the user to login, register and navigate to the dashboard of our system.

3.1.1.1.2 Stimulus/Response Sequence

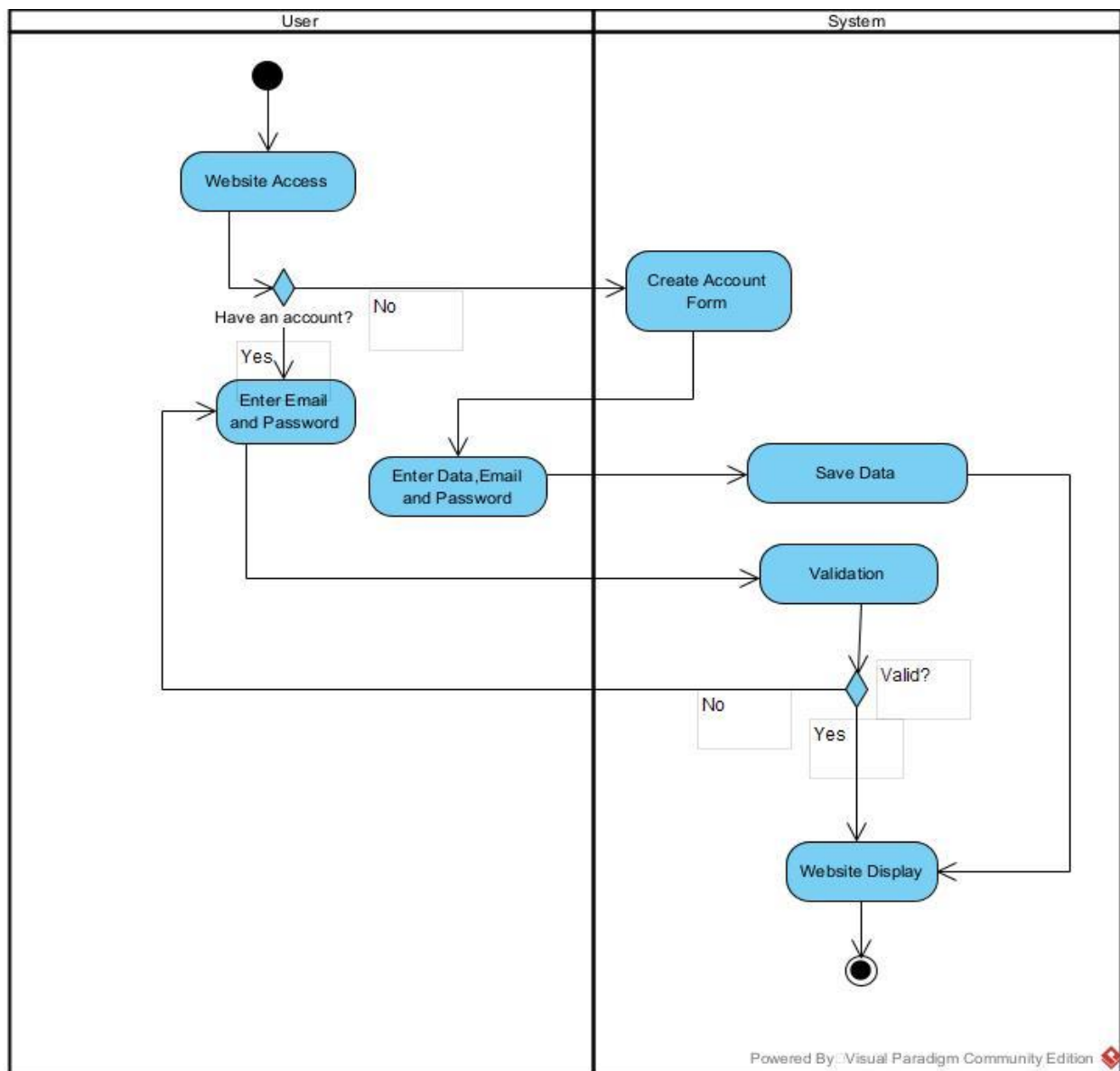


Figure 7-Login Microservice

3.1.1.1.3 Associated Functional Requirements

3.1.1.1.3.1 The application shall provide the user with the ability to select the create account form for the new user.

3.1.1.1.3.2 The application shall provide the new user with the ability to provide their email address and password for the login account creation.

3.1.1.1.3.3 The application shall provide the existing user with the ability to enter their email address and password to login into their account.

3.1.1.1.3.4 The application shall provide the user with the ability to validate the existing user login details and display the website dashboard if no error occurs.

3.3.1.2 DataInsights Microservice

3.1.1.2.1 Introduction/Purpose of Feature

Once the user logs in, they will have the option to view their energy bills, change their user preferences, and navigate to the payment gateway to pay their bills. For all these services, we will be using JAVA's Spring Boot framework to expose a REST API port and through that we will create all the required classes and methods using object oriented methodologies to get the desired result of the APIs. We will connect this microservice to the SQL Database via Java Libraries and maintain a connection for transfer of data through the pipeline.

3.1.1.2.2 Stimulus/Response Sequence

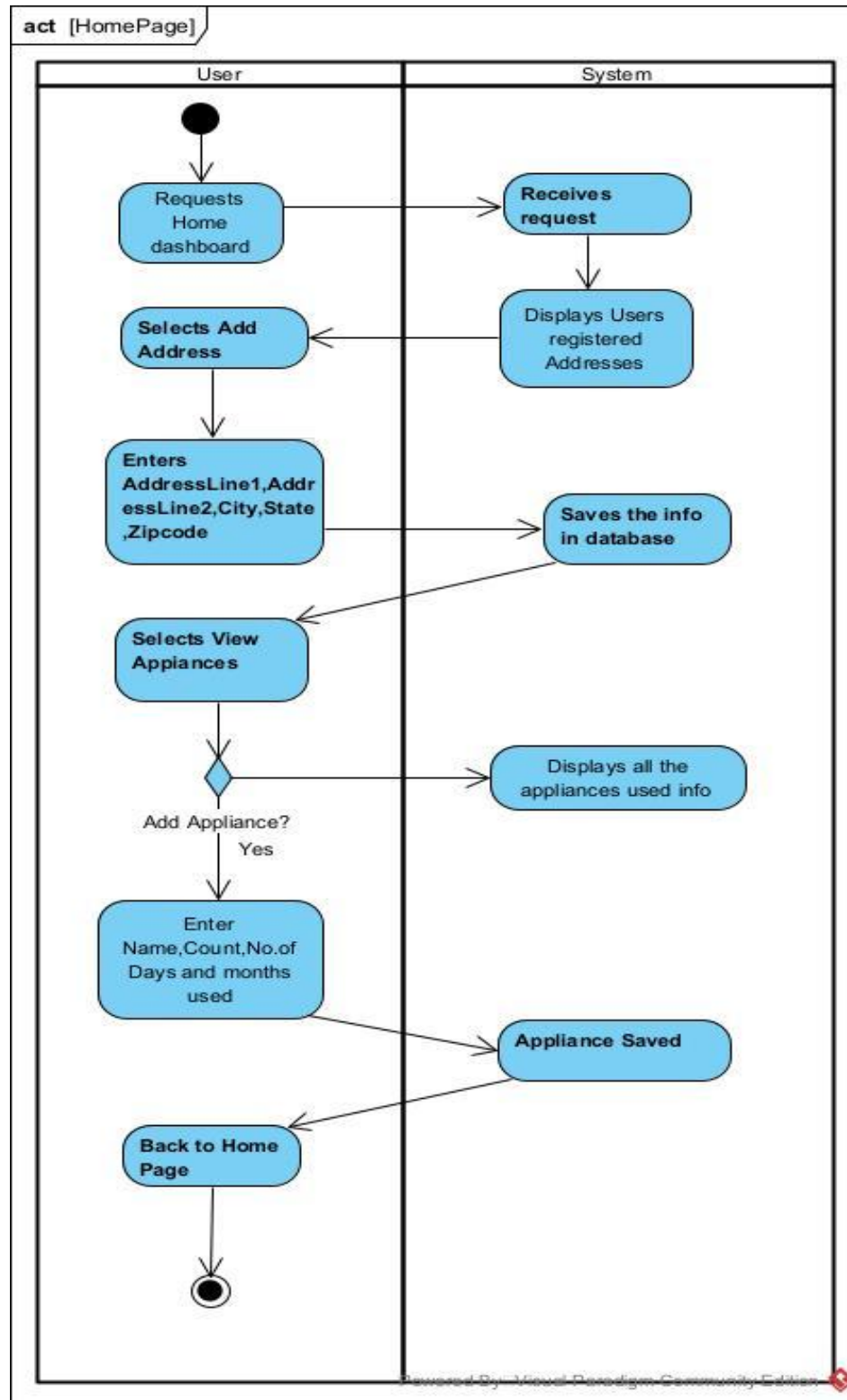


Figure 8-HomePage Activity Diagram

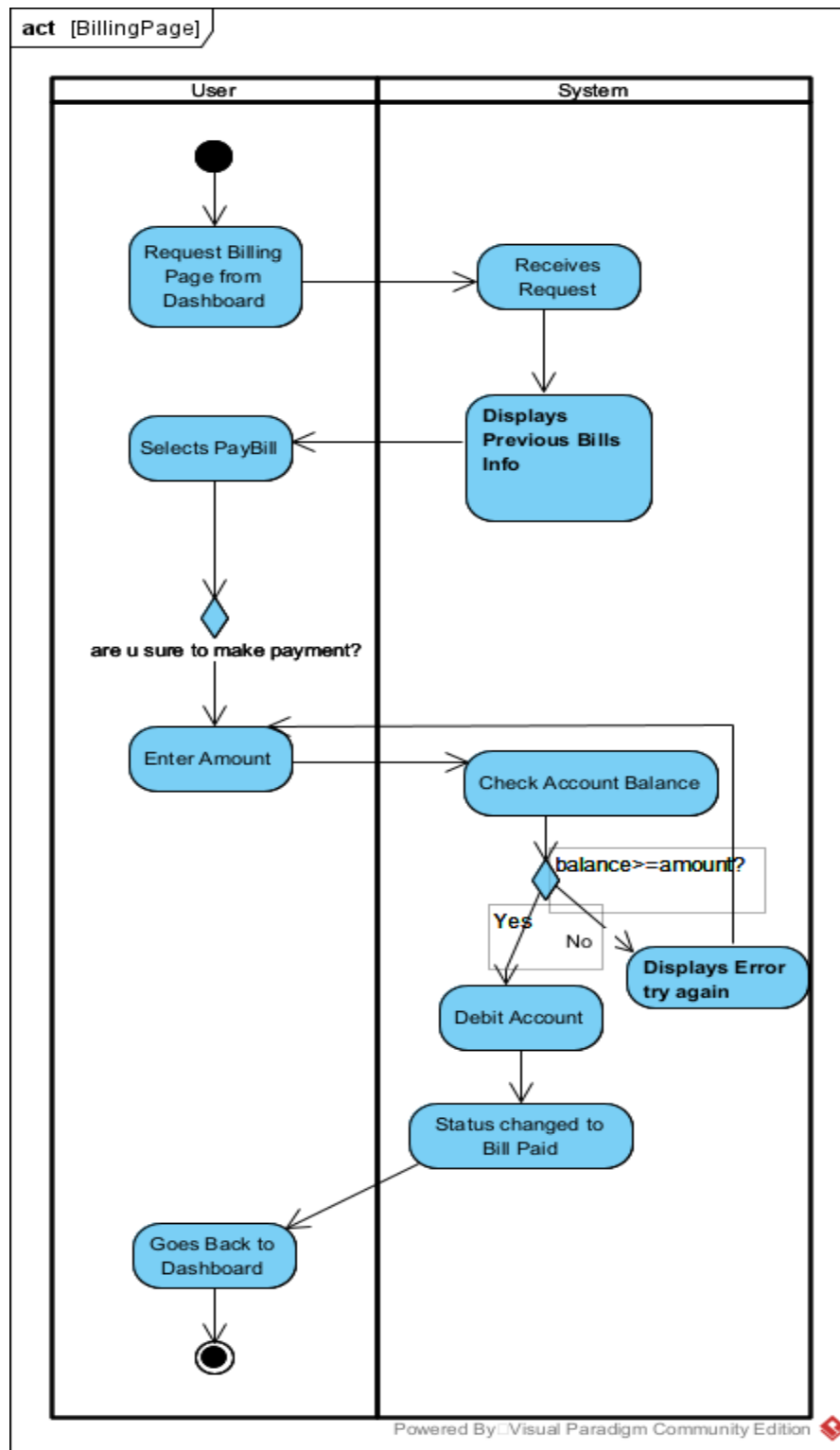


Fig 9- Billing Page Activity Diagram

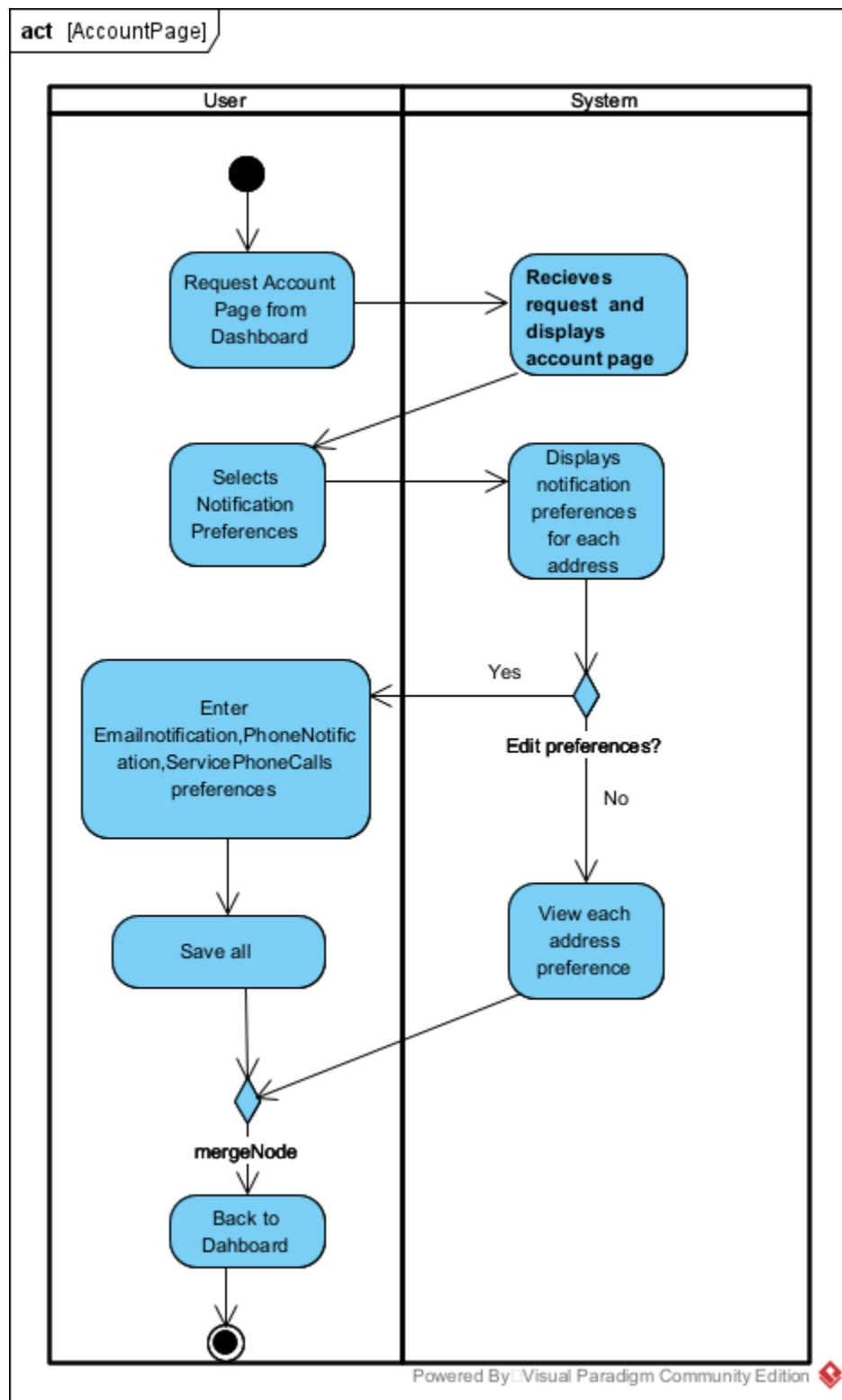


Fig 10- Notification Page Activity Diagram

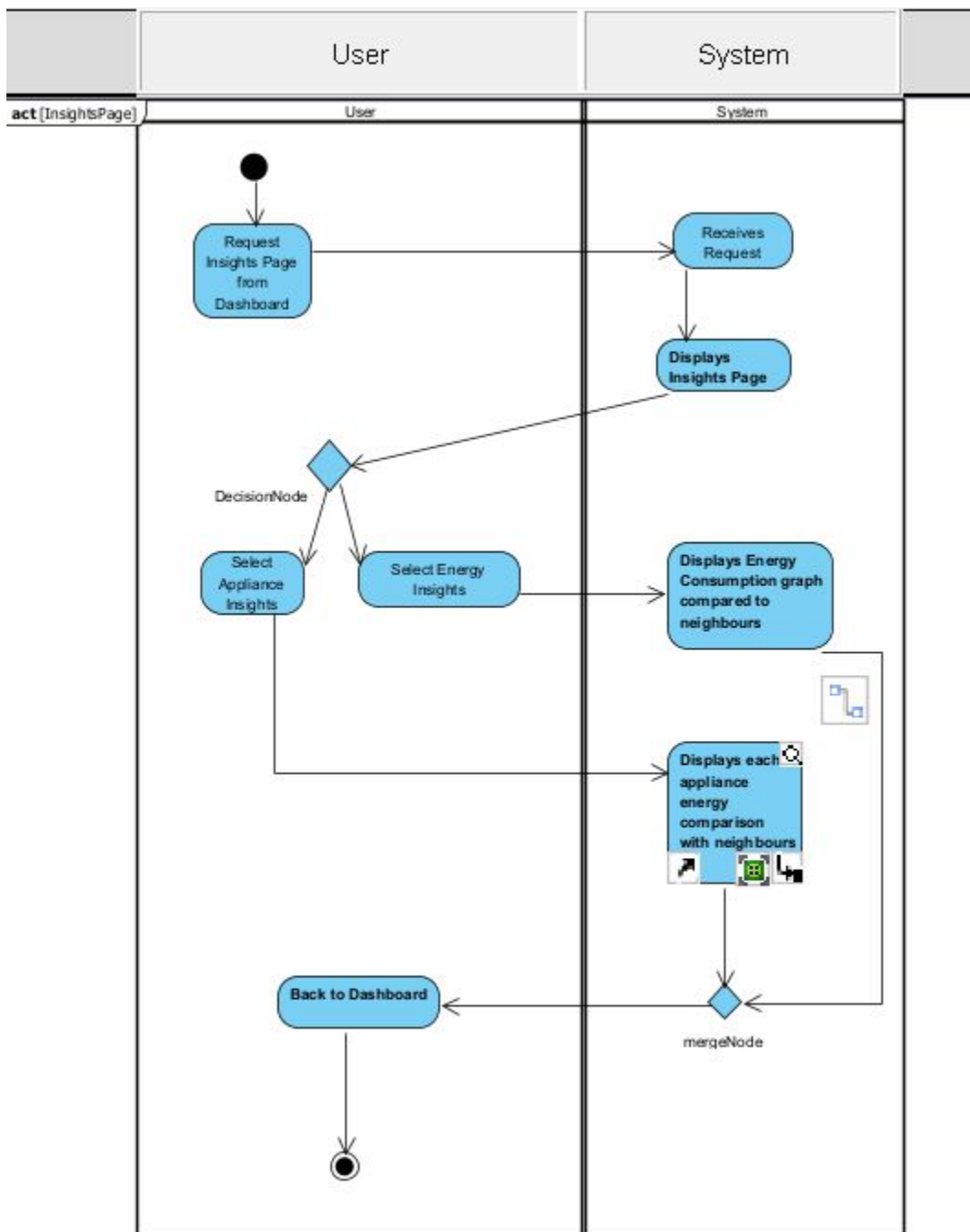


Fig 11- Insights Page Activity Diagram

3.1.1.2.3 . Associated Functional Requirements

The application shall provide the user with the ability to add new address details and edit previous address details.

The application shall provide the user with the ability to add new appliance details for each address selected.

The application shall provide the user with the billing details of present and previous months.

The application shall provide the user the ability to pay bills online

The application shall provide the user the ability to set notification preferences for email, phone messaging and phone calls.

The application shall provide the user the ability to compare their energy usage with that of their neighbors.

The application shall provide the user with the ability to compare each appliance usage comparison with their neighbors.

3.4 Performance Requirements

The application **shall** be responsive and load quickly, even on slow internet connections.

The application **shall** be able to handle a large number of users and accommodate future growth.

3.5 Design Constraints

Software processing **shall** allow a spare capacity of 50% for memory, CPU utilization and long term storage (e.g. disk storage).

The system **shall** implement security measures, such as SSL encryption, to protect user data and prevent unauthorized access

The software **shall** be developed and maintained in accordance with industry standard **ISO/IEC 5055:2021**.

The system **shall** be designed for easy maintenance and updates, with a clear separation between content and functionality.

3.6 Software System Attributes

The software **shall** use data encryption across all interfaces.

The software **shall** allow only authorized users to see their respective house's data to maintain privacy.

3.7 Non-functional Requirements

The desktop application **shall** be secure and protect users' personal information and payment details.

The desktop application **shall** use HTTPS encryption for all pages and use secure cookies to prevent session hijacking.

The desktop application **shall** also use multi-factor authentication to ensure that user accounts are protected.

The desktop application **shall** also implement regular security checks and updates to ensure that it remains secure.

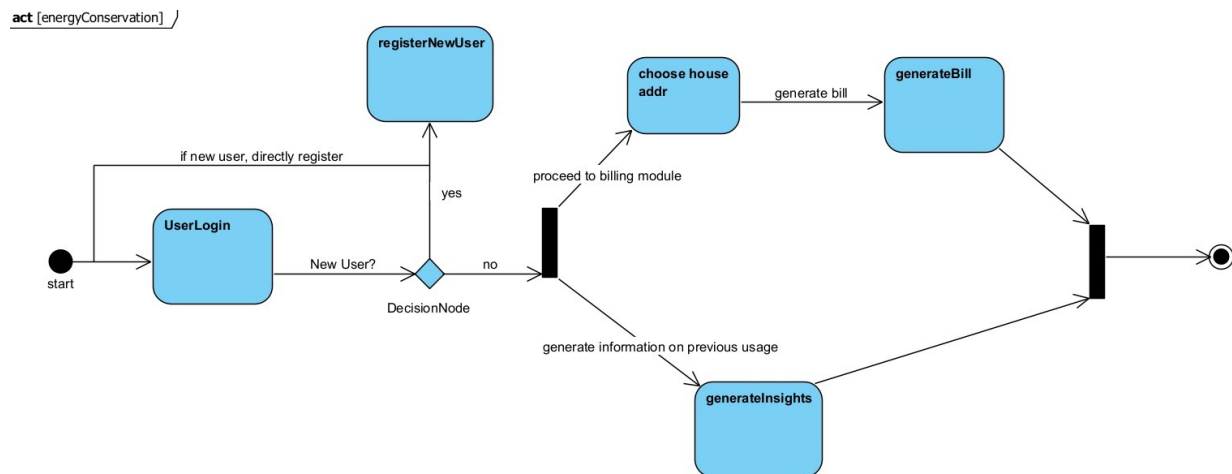


Figure 12- Overall Activity Diagram

Class Diagram

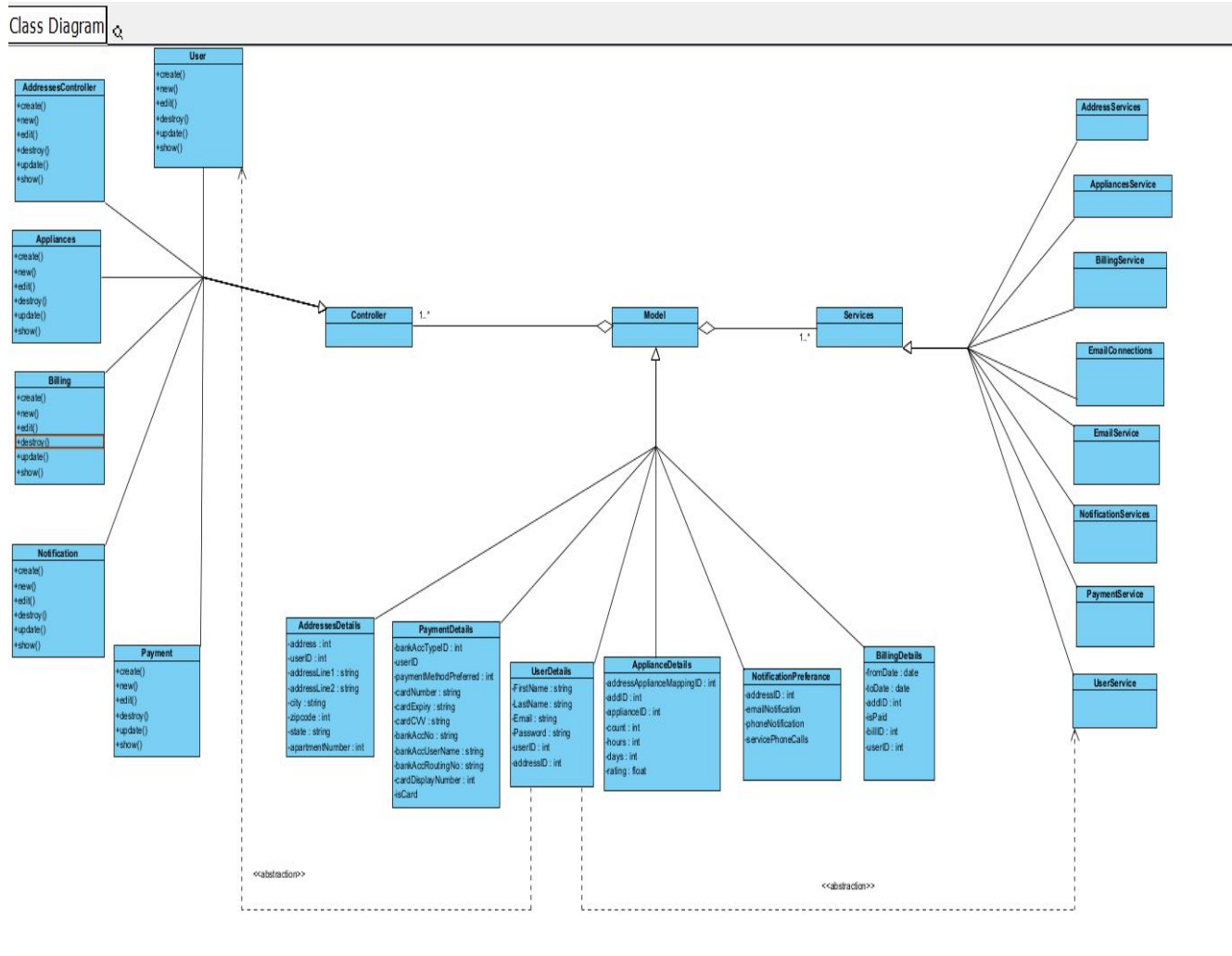


Fig 13- Class Diagram

Sequence Diagram

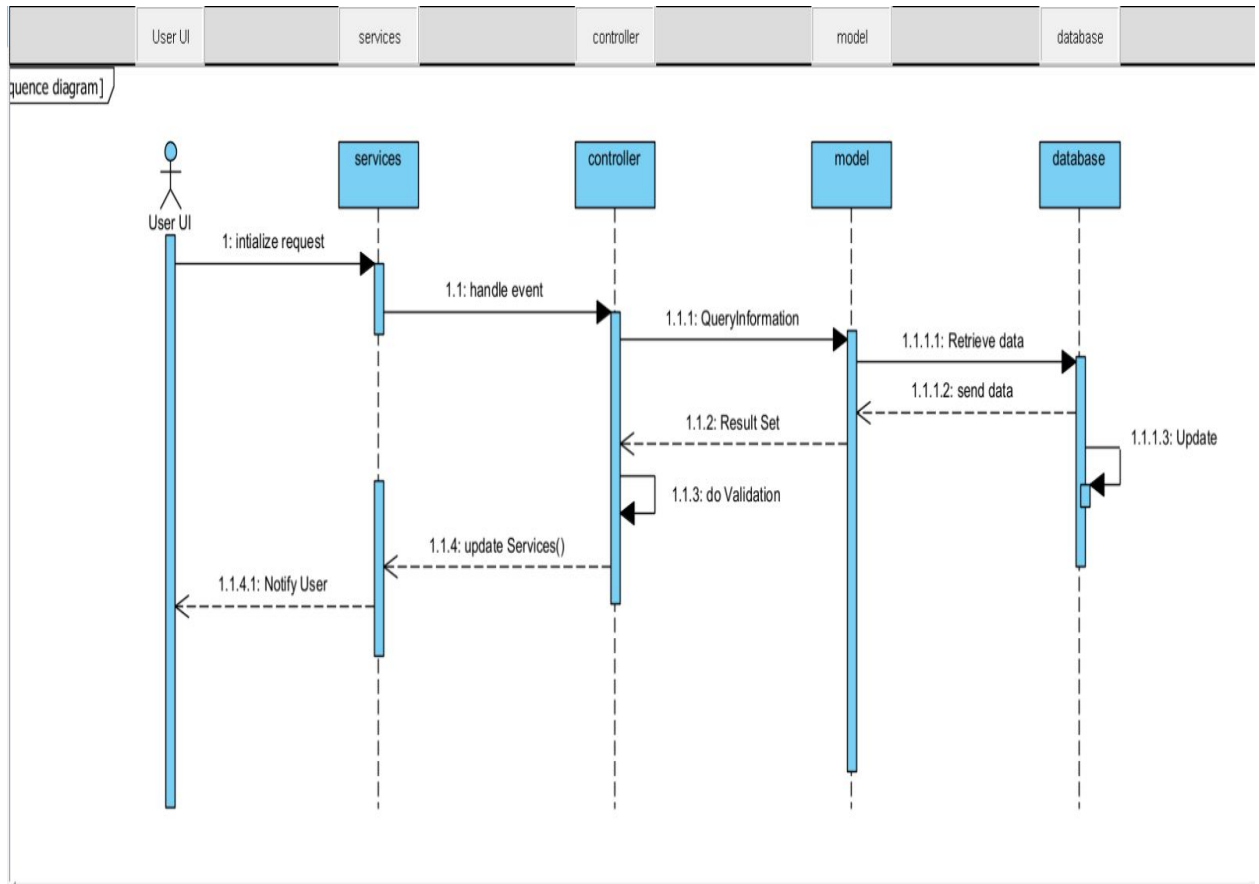


Fig 14 -Sequence Diagram