BILLING SYSTEM

Report on Project of Management Information System

Billing System

Report on Project of MIS
BUS 602

Submitted to:

Dr. Md. Mahbubul Alam Joarder
Professor

Amit Seal Ami
Lecturer

Submitted by:

Md. Jubair Ibna Mostafa (BSSE0614)

Rubaiyat Jahan Mumu (BSSE0620)

Ariful Hoque Maruf (BSSE0621)

Syeda Tasnim Fabiha (BSSE0622)

Fazle Mohammed Tawsif (BSSE0628)

Institute of Information Technology
University of Dhaka

Submission Date:

17.11.2016

Letter of Transmittal

17.11. 2016
Amit Seal Ami
Lecturer
Institute of Information Technology
University of Dhaka

Subject: Submission of term report on Project of MIS; "Billing System".

Dear Sir,

With due respect, we are pleased to submit the final report on "Billing System" that you had asked. In this report, we had to give our best effort albeit there might be some shortcomings. We would be highly obliged if you consider those from excusable point.

Yours sincerely

Md. Jubair Ibna Mostafa (BSSE0614)

Rubaiyat Jahan Mumu (BSSE0620)

Ariful Hoque Maruf (BSSE0621)

Syeda Tasnim Fabiha (BSSE0622)

Fazle Mohammed Tawsif (BSSE0628)

3rd Year, 6th Semester, 6th Batch
Institute of Information Technology
University of Dhaka
Session: 2013-14

Document Authentication

The following Project Report has been accepted and approved by following:

Name	Title	Date	Signature
Md. Jubair Ibna Mostafa (BSSE0614)	Developer	17.11.16	
Rubaiyat Jahan Mumu (BSSE0620)	Analyzer	17.11.16	
Ariful Hoque Maruf (BSSE0621)	Developer	17.11.16	
Syeda Tasnim Fabiha (BSSE0622)	Analyzer	17.11.16	
Fazle Mohammed Tawsif (BSSE0628)	Developer	17.11.16	

Document Approval

Signature

Acknowledgement

By the Grace of ALMIGHTY ALLAH we completed the report on our MIS project **BILLING SYSTEM**.

We would like to thank whole-heartedly our supervisor Amit Seal Ami, Lecturer, Institute of Information Technology, University of Dhaka, for giving us guideline about how can we prepare this report. He helped us a lot by sharing his valuable knowledge with us.

Abstract

Billing System is a web application for maintaining the billing process of Institute of Information Technology. It is an information system. Management of information system is the purpose of the course, Management of Information System (MIS). Implementing an automated billing system will increase the productivity of employees while decreasing the maintenance cost, time and make life easier. So, the necessity of an automated billing system is mandatory for IIT.

Table of Contents

Index of Figures

Chapter 1: Project Plan of "Billing System"

This chapter covers the project proposal and feasibility of the proposal along with background study, product and business perspective, the scopes and some preliminary idea of our web application.

1.1 Background of the Project:

The implementation and deployment of the project Billing System is for the purpose of management of the billing procedure of Institute of information Technology. Billing system is an information system. Management of information system is the purpose of the course, Management of Information System (MIS). Implementing an automated billing system will increase the productivity of employees while decreasing the maintenance cost, time and make life easier. So, the necessity of an automated billing system is mandatory for IIT.

1.2 About the project:

The project billing system is for the assistance of billing process of IIT. The user will have their individual profile and the tracks of the bills will be recorded with this system. So, the system will help to make the billing system transparent to all the user.

1.3 Scope of Our Project:

For the implementation of the billing system we defined our following scopes:

1. There is a well-organized budget management system.
2. There is a well-organized expenditure management system.
3. We will not automate the bill payment process

1.4 Project Scheduling:

Start Date	End Date	Project State and Objectives
1.7.2016	09.8.2016	Project Proposal, meeting with supervisor about our idea
10.9.2016	30.10.2016	Planning, thinking about project story, features and Learning Technology
1.11.2016	16.11.2016	Start constructing SRS document & project report, choose tools, environment and learning technology
4.11.2016	17.11. 2016	Complete SRS document & project report, Start design and implementation
20.10.2016	17.11.2016	Start full phase implementation
----------	17.11.16	Project Submission

1.5 Conclusion

In this chapter, the planning of building the project Billing System is described in brief. The background of the project and scopes are also defined here.

Chapter 2: Introduction of SRS

This chapter is a part of our software requirement specification for the project "Billing System". In this chapter we focus on the intended audience for this project.

2.1 Purpose

This document briefly describes the software requirement analysis of Billing System. It contains functional, non-functional and support requirements and establishes a requirements baseline for the development of the system. The requirements contained in the SRS are independent, uniquely numbered, and organized by topic. The SRS serves as official means of communicating user requirements to the developer and provides a common reference point for both the developer team and stakeholder community. The SRS will evolve over time as users and developers work together to validate, clarify and expand its contents.

2.2 Intended Audience

This SRS is intended for several audiences including the customers as well as the project managers, designers, developers, and testers.

- The customer will use this SRS to verify that the developer team has created a product that is acceptable to the customer.
- The project managers of the developer team will use this SRS to plan milestones and a delivery date, and ensure that the developing team is on track during development of the system.
- The designers will use this SRS as a basis for creating the system's design.
- The designers will continually refer back to this SRS to ensure that the system they are designing will fulfill the customer's needs.

- The developers will use this SRS as a basis for developing the system's functionality. The developers will link the requirements defined in this SRS to the software they create to ensure that they have created software that will fulfill all of the customer's documented requirements.
- The testers will use this SRS to derive test plans and test cases for each documented requirement. When portions of the software are complete, the testers will run their tests on that software to ensure that the software fulfills the requirements documented in this SRS. The testers will again run their tests on the entire system when it is complete and ensure that all requirements documented in this SRS have been fulfilled.

2.3 Conclusion

This analysis of the audience helped us to focus on the users who will be using our system. This overall document will help each and every person related to this project to have a better idea about the project.

Chapter 3: Inception

In this chapter, the Inception part of the software requirement analysis of our project will be discussed briefly.

3.1 Introduction

Inception is the beginning phase of requirements engineering. It defines how does a software project get started and what is the scope and nature of the problem to be solved. The goal of the inception phase is to identify concurrence needs and conflict requirements among the stakeholders of a software project. At project inception, we establish a basic understanding of the problem, the people who want a solution, the nature of the solution that is desired, and the effectiveness of preliminary communication and collaborations between the other stakeholders and the software team.

3.2 Establishing the ground work

To establish the groundwork, we have worked with the following factors related to the inception phases:

- List of stakeholders
- Recognizing multiple viewpoints
- Working towards collaboration
- Requirements questionnaire

3.2.1 List of stakeholders

Stakeholder refers to any person or group who will be affected by the system directly or indirectly. Stakeholders include end-users who interact with the system and everyone else in an organization that may be affected by its installation. At inception, a list of people who will contribute input as requirements are elicited (Chapter 3) is created. The initial list will grow as

stakeholders are contacted because every stakeholder will be asked: “Whom else do you think I should talk to?”

To identify the stakeholders, we consulted with the teachers and student who are cyclists or runner asked those following questions:

- What will be the main purpose of using the web application?
- Who will be using the project outcomes?
- Where we will find the resources we need to get the project done?
- Whose work will my project affect? (During the project and also once the project is completed).

Concluding thoughts on Stakeholders thoughts we identified that the common stakeholders of this system are all the faculties and the worker related to the existing manual billing system.

Faculties of IIT: All the faculties will use the system as their part of billing procedure.

Director of IIT: Director will use the system to approve bill and he/she will also be able to make his own bill.

Coordinator of IIT: Coordinator will use the bill use the system for approving other’s bill as well as his/her own bill.

Other people working in billing process in IIT: Other accountants of IIT will use the system as a part of billing procedure.

Developers: Developers will build the system and work on further development of the project. They will be responsible for any type of interruption and they will solve it.

3.2.2 Recognizing multiple viewpoints

Though all types of stakeholders we identified will use the application on same purposes, each of them has slightly different view of the system. So we have to identify the multiple views of requirements. Assumptions are given below:

Viewpoint of Faculties:

- User friendly efficient system
- System must generate the bill in pdf format
- The generated value from the system should be error free
- System must show the approval of the bill
- Strongly authenticated system
- Bill should be made dynamically
- Easily operable system

Viewpoint of Coordinator:

- User friendly system
- System must generate the bill in pdf format
- Billing calculation must be specific and exact
- Dynamic bill making process should be included
- Bill should be configurable by the user
- System must be strongly authenticated
- System should give the configuration option to change role of other user
- System must generate the bill in pdf format
- Easily operable system

Viewpoint of Director:

- User friendly system
- System must be strongly authenticated
- Dynamic bill making process should be included
- Billing calculation must be specific and exact
- System must generate the bill in pdf format
- It will be easy to operate the system

Viewpoint of Other Workers:

- User friendly system
- System must be strongly authenticated
- It will be easy to operate the system

Viewpoint of Developers:

- All the requirements will be well defined
- No major requirement change will come in the development phase
- All the stakeholders will be available for further information and consult

3.2.3 Working towards collaboration

Every stakeholder has their own requirements from their own point of view. We followed following steps to merge all the requirements. We-

- Identified the common and conflicting requirements
- Categorized the requirements
- Take priority points for each requirements from stakeholders
- Make final decision about the requirements

Common requirements: We found some requirements are all the same from different users. These are:

- User friendly and efficient system
- Easy to operate
- Authentication
- Making the bill in pdf format
- Dynamic billing process
- Error free and exact calculation

Conflicting requirements: We found some requirements conflicting each other. We had to trade-off between the requirements.

- Availability of all requirements within the budget
- No ambiguous requirement
- Easy access
- Strong authentication and high security
- No harmful effects on existing technology

Final requirements: We finalized following requirements for the system by categorizing and prioritizing the requirements.

- Error free system (Maximum 5% error may be considerable)
- Allow the users login and logout
- User friendly and efficient system
- Give the specific user specific role
- Automated entry of billing information
- Dynamic billing process
- Generate bill in pdf
- Central database contains all billing information

3.2.4 Requirement Questionnaire

We set our first set of context-free questions focuses on the customer and other stakeholders, overall project goals and benefits. The questions are mentioned above. These questions helped us to identify all stakeholders, measurable benefit of the successful implementation and possible alternatives to custom software development. Next set of question helped us to gain a better understanding of problem and allows the customer to voice his or her perception about the solution. The final set of question focused on the effectiveness of the communication activity itself.

3.3 Conclusion

Inception phase helped us to establish basic understanding about the project Billing System, identify the people who will be benefited using this system, define the nature of the project and establish a preliminary communication with our stakeholders.

In our project, we have established a basic understanding of the problem, the nature of the solution that is desired and the effectiveness of preliminary communication and collaboration between the stake-holders and the software team. More studies and communication will help both side (developer and client) to understand the future prospect of the project. Our team believes that the full functioning document will help us to define that future prospect.

Chapter 4: Elicitation

After discussing on inception part, we need to keep focus on the elicitation part. So this chapter specifies the elicitation part.

4.1 Introduction

Requirements elicitation is a part of requirement engineering that is the practice of gathering requirements from the users, customers, and other stakeholders. We have faced many problems like understanding the problems, problems of making questions for the stakeholders, problems of less communication with the stakeholders for time limitation, problems of volatility. Though it is not too easy to gather requirements within a very short time, we have surpassed these problems in an organized and systematic manner.

4.2 Eliciting Requirements

We have seen Question and Answer (Q&A) approach in the previous chapter, where the inception phase of requirement engineering has been described. The main task of this phase is to combine the elements of problem solving, elaboration, negotiation and specification. The collaborative working approach of the stakeholders is required to elicit the requirements. We have finished the following tasks for eliciting requirements-

- Collaborative Requirements Gathering
- Quality Function Deployment
- Usage Scenarios
- Elicitation work products

4.2.1 Collaborative Requirements Gathering

Actually, we met with many stakeholders in the inception phase such as cyclists, runners, our classmates and teacher as the common people and also the developers. These meetings created an indecisive state for us to elicit the

requirements. To solve this problem, we have met with the stakeholders (who are acting a vital rule in the whole process) again to elicit the requirements. A slightly different scenario from these approaches has been found.

Following activities have been completed to accomplish this task.

- The meetings were conducted with the director, coordinator, accountants and other faculties of IIT; they were questioned about their requirements and expectations from the web application we are developing.
- They were asked about the problems they are facing without using any relevant application regarding sports activity.
- At last we selected our final requirement list from the meetings.

4.2.2 Quality Function Deployment

Quality Function Deployment (QFD) is a technique that translates the needs of the customer into technical requirements for software. Ultimately the goal of QFD is to translate subjective quality criteria into objective ones that can be quantified and measured and which can then be used to design and manufacture the product. It is a methodology that concentrates on maximizing customer satisfaction from the software engineering process. So we have followed this methodology to identify the requirements for the project. The requirements, which are given below, are identified successfully by the QFD.

4.2.2.1 Normal Requirements

Normal requirements are generally the objectives and goals that are stated for a product or system during meetings with the customer. The presence of these requirements fulfills customers' satisfaction. These are the normal requirements for our project.

- Allow user to do registration
- Allow valid user to sign in and sign up
- Check user validity

- Features to have an idea about the user's activity
- Feature that will allow user role
- Feature that will allow dynamic billing process
- Allow users to generate pdf of the bill
- Efficient and user friendly
- The user interface of the system would be easy
- Allow user to view approval of the admin panel
- Allow user to give approval/rejection of a bill
- Allow user to search other using their user name
- Security issue
- Error free activity

4.2.2.2 Expected Requirements

These requirements are intrinsic to the product or system and may be so elementary that the customer does not explicitly state them. Their absence will be a cause for significant dissatisfaction. Below the expected requirements for our project are briefly described.

- The application shall be easily maintainable.
- The application will be stable.
- The application will be open for future extension and modification.
- All the records and bills will be saved to database for future references.
- The application shall be user friendly for all type of users.
- The user interface shall make use of input such as drop downs, check boxes and radio buttons as much as possible to avoid invalid and incorrect input.

4.2.2.3 Exciting Requirements

These requirements are for features that go beyond the customer's expectations and prove to be very satisfying when present. Following are some exciting requirements of our project.

- Users can save graphs as images.
- Different session will be provided for different users.
- Connect user account with Facebook or other social media

4.2.3 Usage Scenario

The billing system has four kinds of users: an admin, a director, coordinators and normal users. The admin has the supreme authority over the whole system. He can promote any user to a coordinator. Also anytime he can demote a coordinator to a normal user.

Every user needs to be authenticated to enter into the system. Each user must provide following information during registration.

1. User's full Name
2. User's Email Address
3. User's designation
5. Password

There are some categories on which bills are made. If a user wants to make a bill, he enters the name of the bill. Then he selects the category/categories of the corresponding bill and enters the amount for each category. If the category is not listed, the user can request for adding new category. The admin adds new categories based on requests from the users.

Then the user creates a document of the bill and sends this document to the corresponding coordinator of the bill. The coordinator makes a validation check on the bill and then approves or rejects the bill. If the bill is approved or rejected, the user is notified about it. If a bill is rejected, the user can edit the billing document and send it back to the coordinator again for approval.

If the bill is approved, the coordinator sends it to the director for final approval. When a bill is sent for final approval, the user is notified about it again. The director approves it and the system notifies the user about the approval. After final approval, user can generate a pdf of the bill and print it out.

Billing fields:

- Admission
- Examination
- Extra-curriculum activity
  - ❖ Games
  - ❖ Cultural programs
  - ❖ Picnic
- Script Checking
- Overtime
- Food
- Training Programs
- Regular course
- Short course
- Maintenance
  - ❖ Buying new things
  - ❖ Repair

4.2.4 Elicitation Work Product

At first we have to know that the output of the Elicitation task may vary because of the dependency on size of the system or product to be built. Here, the elicitation work product includes:

- Making a statement of our requirements for the project
- Making a bounded statement of scope for our system.
- Making a list of customers, users, and other stakeholders who participated in the requirements elicitation.
- Making a list of requirements that are organized by function and domain constraints that apply to each.
- A set of usage scenarios that provide insight into the use of the system.
- Description of the system's technical environment.

4.3 Conclusion

Elicitation process gives us a clear view of the requirements of the stakeholders and develops our understanding of the whole project. It also enables us to deliver a product that will satisfy all the stakeholders. This phase also helps us to identify the requirements, negotiate different approaches and specify a preliminary set of solution requirements in an atmosphere that is conducive to the accomplishment of the goal.

Chapter 5: Scenario Based Modeling

This chapter describes the scenario based model for the project Billing System.

5.1 Introduction

Although the success of a computer-based system or product is measured in many ways, user satisfaction resides at the top of the list. If we understand how end users (and other actors) want to interact with a system, our software team will be better able to properly characterize requirements and build meaningful analysis and design models. Hence, requirements modeling with begins with the creation of scenarios in the form of use cases, activity diagrams, and swim lane diagrams.

5.2 Definition of Use case

A use case captures a contract that describes the system behavior under various conditions as the system responds to a request from one of its stakeholders. In essence, a use case tells a stylized story about how an end user interacts with the system under a specific set of circumstances. A use case diagram simply describes a story using corresponding actors, who perform important role in the story and makes the story understandable for the users.

The first step in writing a use case is to define that set of “actors” that will be involved in the story. Actors are the different people that use the system or product within the context of the function and behavior that is to be

described. Actors represent the roles that people play as the system operators. Every user has one or more goals when using system.

Primary Actor: Primary actors interact directly to achieve required system function and derive the intended benefit from the system. They work directly and frequently with the software.

Secondary Actor: Secondary actors support the system so that primary actors can do their work. They either produce or consume information.

5.3 Use Case Diagrams

Use case diagram is the non-technical view of overall system. The system is described from the user's point of view. As this is the first model, it serves as input for creation of other modeling elements.

5.3.1 System Description of Level-0 Use Case Diagram

After analyzing the user story, we found six actors who will directly use the system as a system operator. Primary actors are those who will play action and get a reply from the system whereas secondary actors only produce or consume information.

We identified that our all our actors of the system fall into the same category. They will all be our users, who will be using our system to fulfill their billing procedures.

5.3.1.1 Level-0 Use Case Diagram

In this level of use case diagram describes the overall system and the actors interacting with the system. Here in our project we have only one category user dealing with the system.

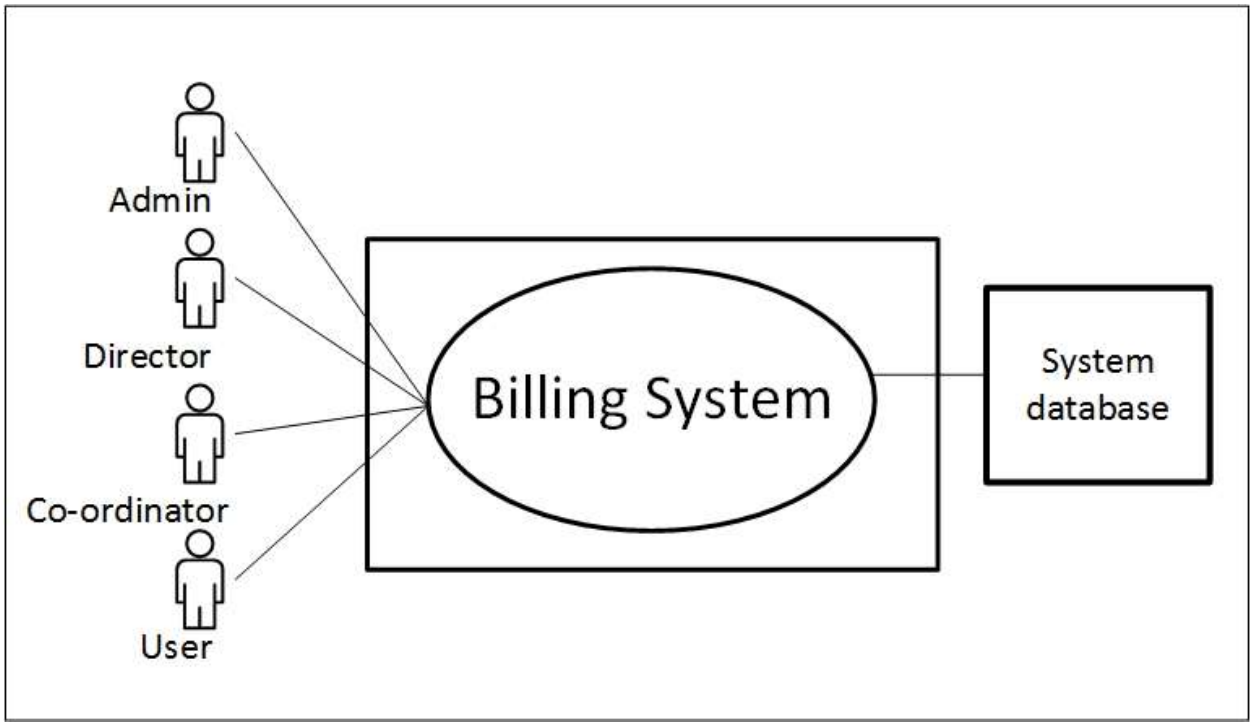


Figure 1: Level-0 Use Case Diagram

5.3.2 System Description from Level-1 Use Case Diagram

The actors our system have to play different actions and system will reply according to these actions –

Action 1: Enters signup.

Reply 1: Please fill up the required information.

Action 2: Enters the information

Reply 2: Registration successful.

Action 3: Enters username and password.

Reply 3: Sign in successful.

Action 4: Enters **Create Bill** button

Reply 4: Create bill interface will appear

Action 5: Enters **Bill description**s.

Reply 5: Prompts for further Bill information.

Action 6: Enter Add-User button to create coordinators.

Reply 6: Coordinator added.

Action 7: Enter approved button for Bill.

Reply 7: Bills are sent for next verification

Action 8: Enters to view the available features.

Reply 8: Features appeared in the page.

Exception: No Exceptions

5.3.2.1 Level-1 Use Case Diagram

Level-1 use case where total system is divided into its subsystems which elaborately described in section 5.3.2.

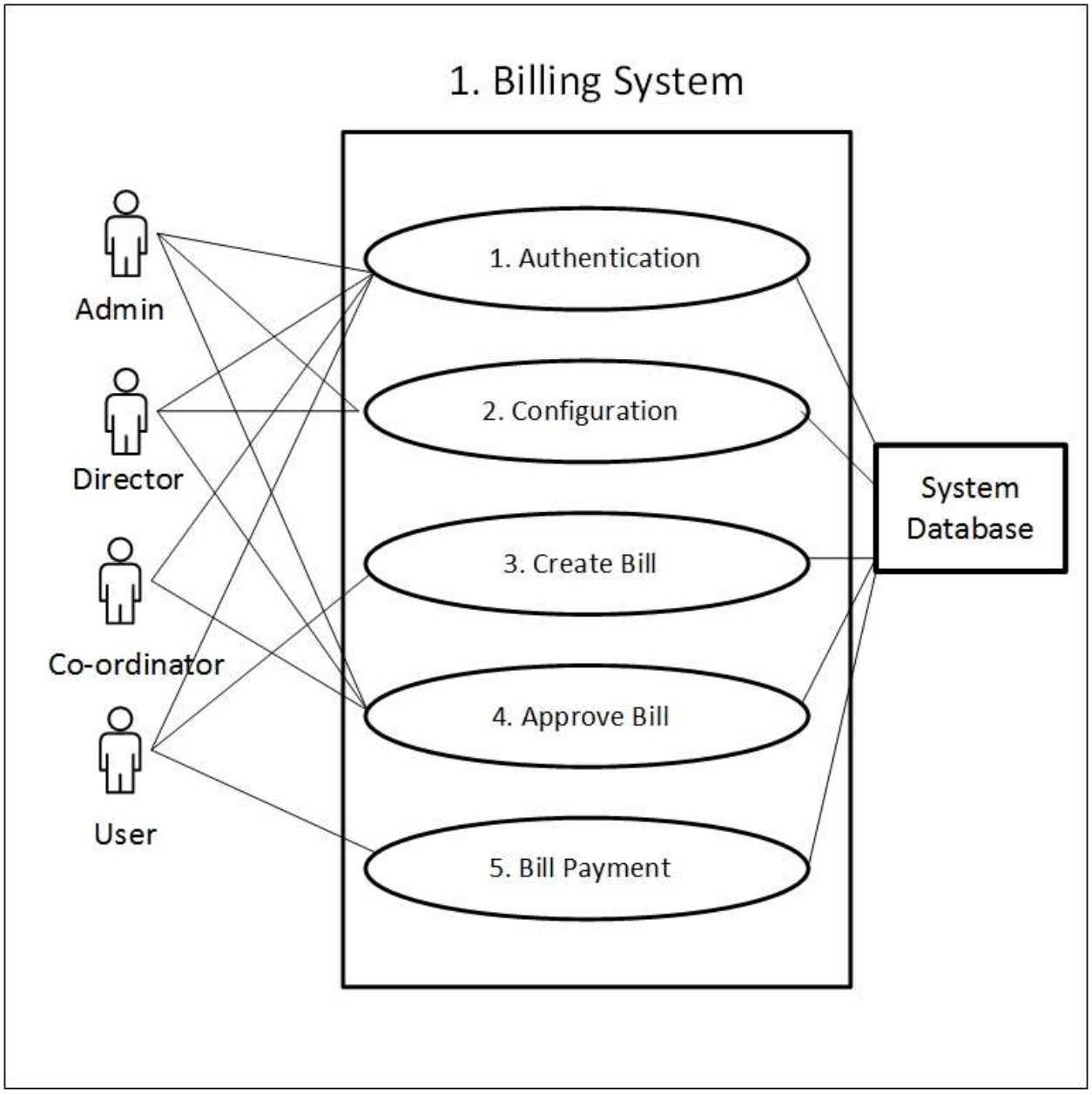


Figure 2: Level-1 Use Case Diagram

5.3.3 Level-2.1 Use Case Diagram

Subsystems of subsystem 1.1 of level-1 use case diagram. System description described in the section 5.3.2.

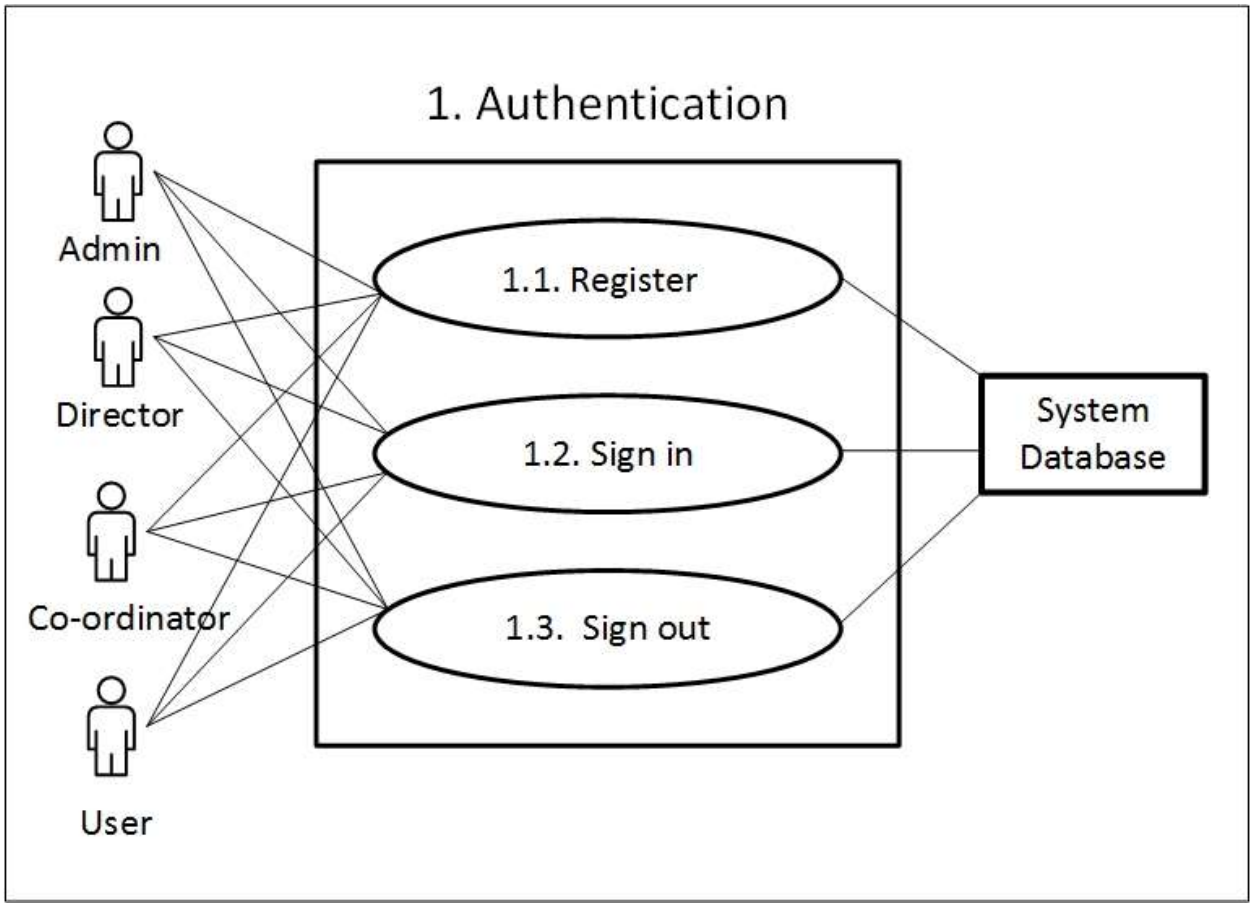


Figure 3: Level-2.1 Use Case Diagram

5.3.4 System Description from Level-2.2 Use Case Diagram

Here the sub-subsystems of level-1 subsystem 2.1 are described according to users' point of view.

Once a user clicked on **Edit Info** button following actions he/she need to conduct and the system will response on user's corresponding action.

Action 1:

- Enters First Name
- Enters email.
- Enter password
- Clicks on **Save Changes.**

Reply 1: Changes successfully saved.

Action 2: Enters **Change Password.**

Reply 2: Prompts for enter previous password.

Action 3: Enters previous password.

Reply 3: Prompts for new password.

Action 4:

- Enters new password.
- Re-enters new password.
- Enters **Save Changes.**

Reply 4: Password successfully changed.

Action 5: Enters sign out.

Reply 3: Logged out user.

Exception: No Exceptions

5.3.4.1 Level-2.2 Use Case Diagram

The subsystem 1.2 is divided into three sub-subsystems which are shown in following level-2.2 Use Case Diagram.

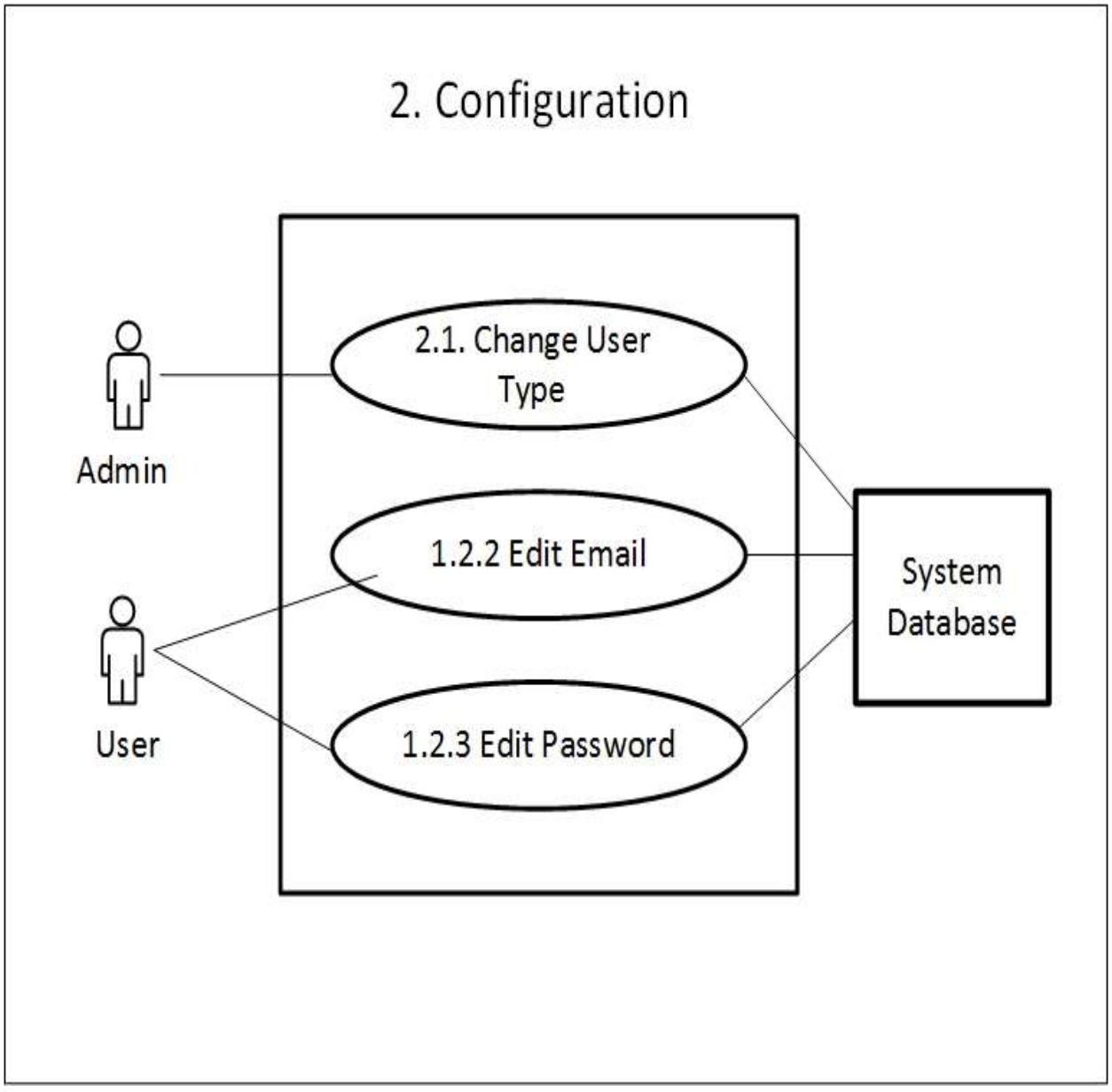


Figure 4: Level-2.2 Use Case Diagram

5.3.5 System Description from Level-2.3 Use Case Diagram

Here the sub-subsystems of level-1 subsystem 1.3 are described.

Once a user clicked on **Configuration** button the user will be able to configure the information provided by the user. Following are actions he/she need to conduct for further configuration process and the system will response on user's corresponding action.

Action 1: Enters **Configuration.**

Reply 1: Prompt for Information configuration.

Action 1: Enter change user type.

Applicable when: Actor is admin.

Reply 2: User type changed

Action 3: Enter edit email

Reply 3: Email is edited.

Action 4: Enter edit password

Reply 4: Password is edited

5.3.5.1 Level-2.3 Use Case Diagram

The subsystem 1.3 Bill Creation is divided into four sub-subsystems which are shown in following level-1.3 Use Case Diagram.

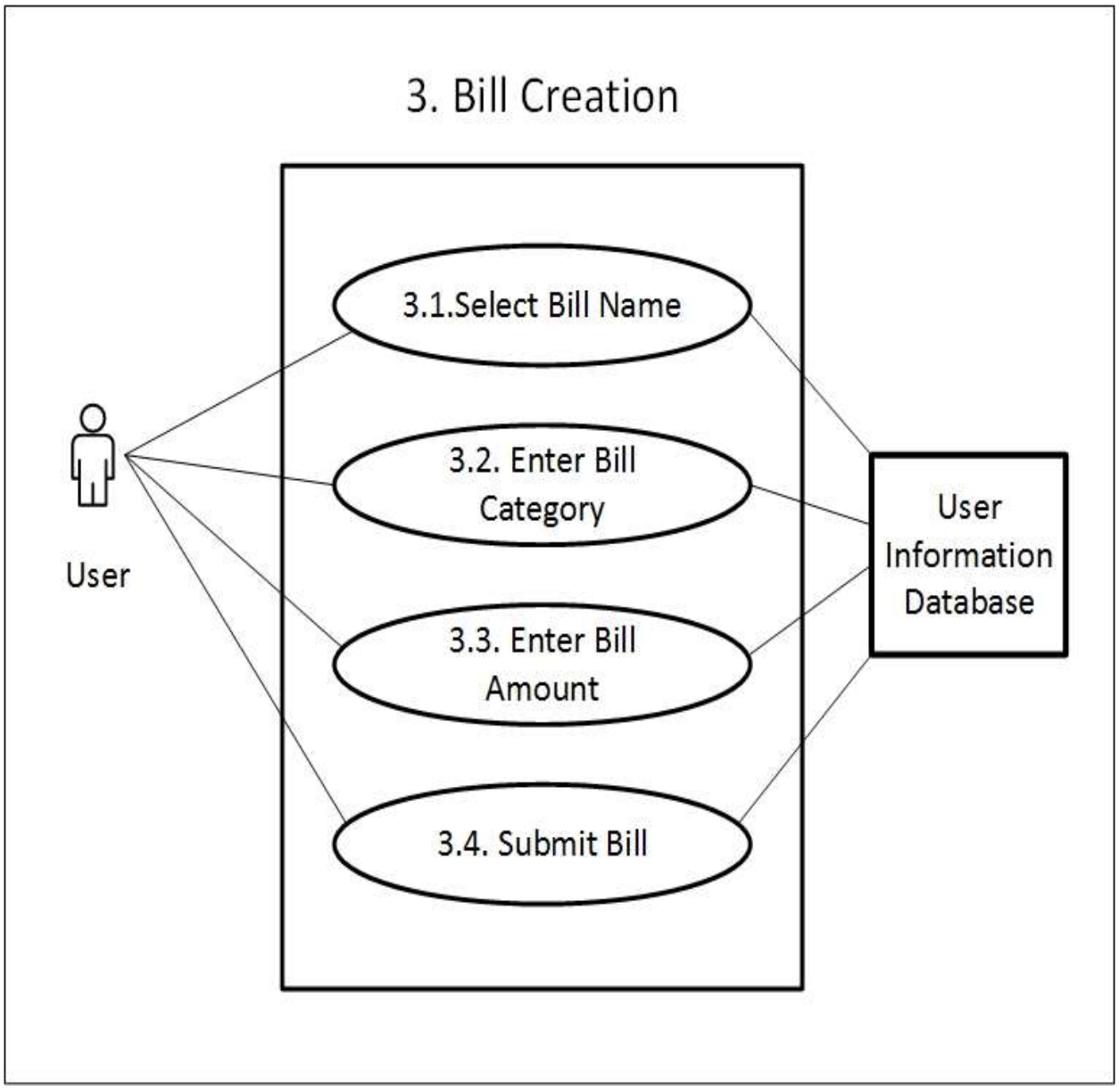


Figure 5: Level-1.3 Use Case Diagram

5.3.6 System Description from Level-2.4 Use Case Diagram

Once a user clicked on **bill create** button following are actions are needed to conduct for further post submission process and the system will response on user's corresponding action.

Action 1: Enters **Bill create** Button

Reply 1: Take to the create bill page

Action 2: Enters Bill name, Category and Bill amount and clicks on **submit.**

Reply 2: Bill name, category and amount is saved.

Action 3: Request for approval from Admins

Reply 3: Bill is accepted/ rejected

5.3.6.1 Level-2.4 Use Case Diagram

The subsystem 1.4 Posting is divided into four sub-subsystems which are shown in following level-1.4 Use Case Diagram.

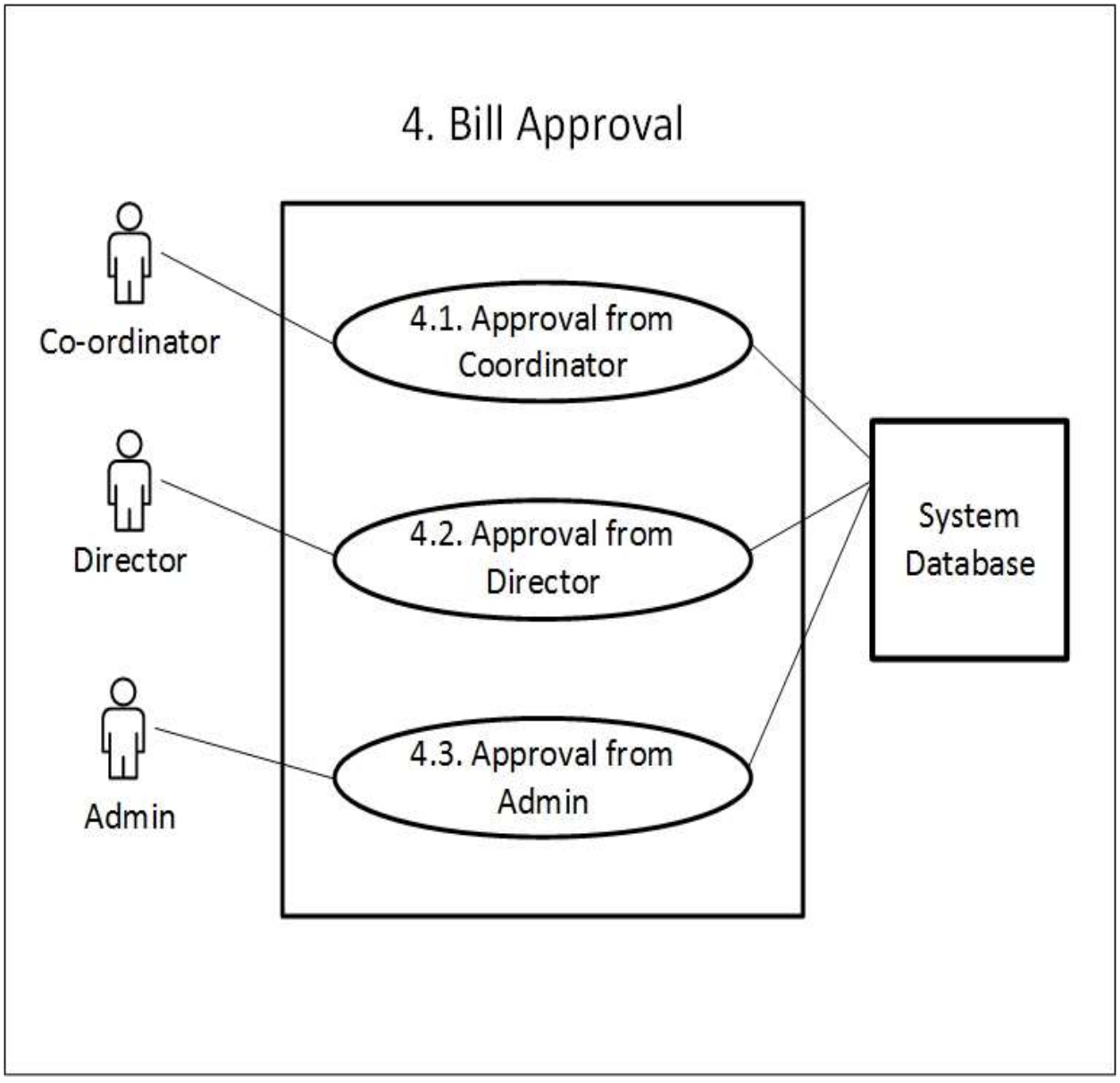


Figure 6: Level-2.4 Use Case Diagram

5.3.7 System Description from Level-3 Use Case Diagram

The Data Analysis subsystem is divided into three major sub systems. Following are the actions and replies regarding this system.

Action 1: Request approval from admins

Reply 1: Permission accepted/ rejected

Action 2: If “Action 1” is accepted, request approval from coordinators

Reply 2: Permission accepted/ rejected

Action 3: If “Action 2” is accepted, request approval from director

Reply 3: Permission accepted or rejected.

Exception: No Exceptions

5.3.7.1 Level-3 Use Case Diagram

The subsystem Data Analysis is divided into two sub-subsystems. The system description of this subsystem are described in previous section.

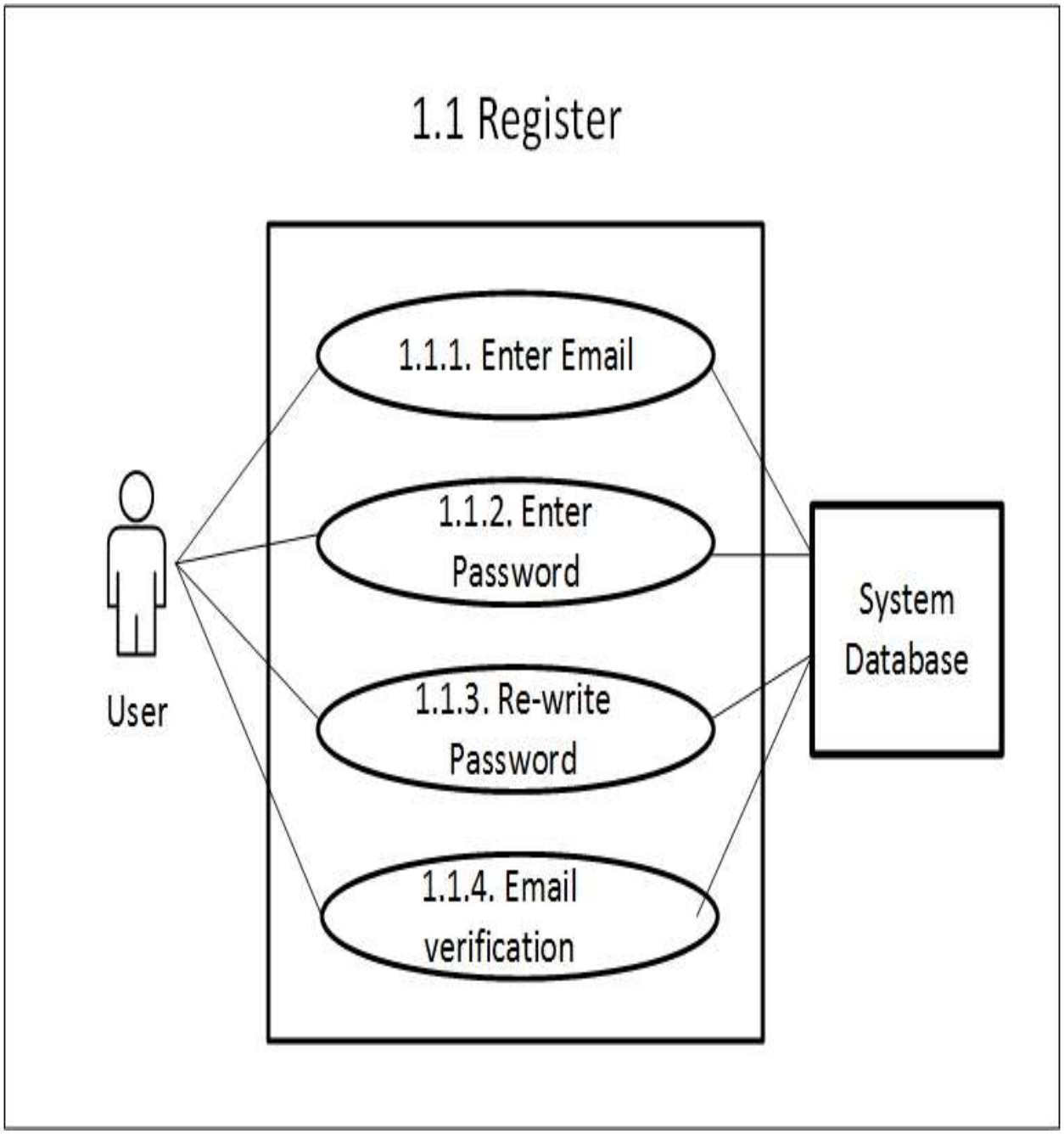


Figure 7: Level-3 Use Case Diagram

5.4 Activity & Swim Lane Diagrams

Activity diagram shows the technical view of the system for every use case from which we can understand how the system actually works and how the actors interact with the system. Here is the activity diagram for our Sign Up use case.

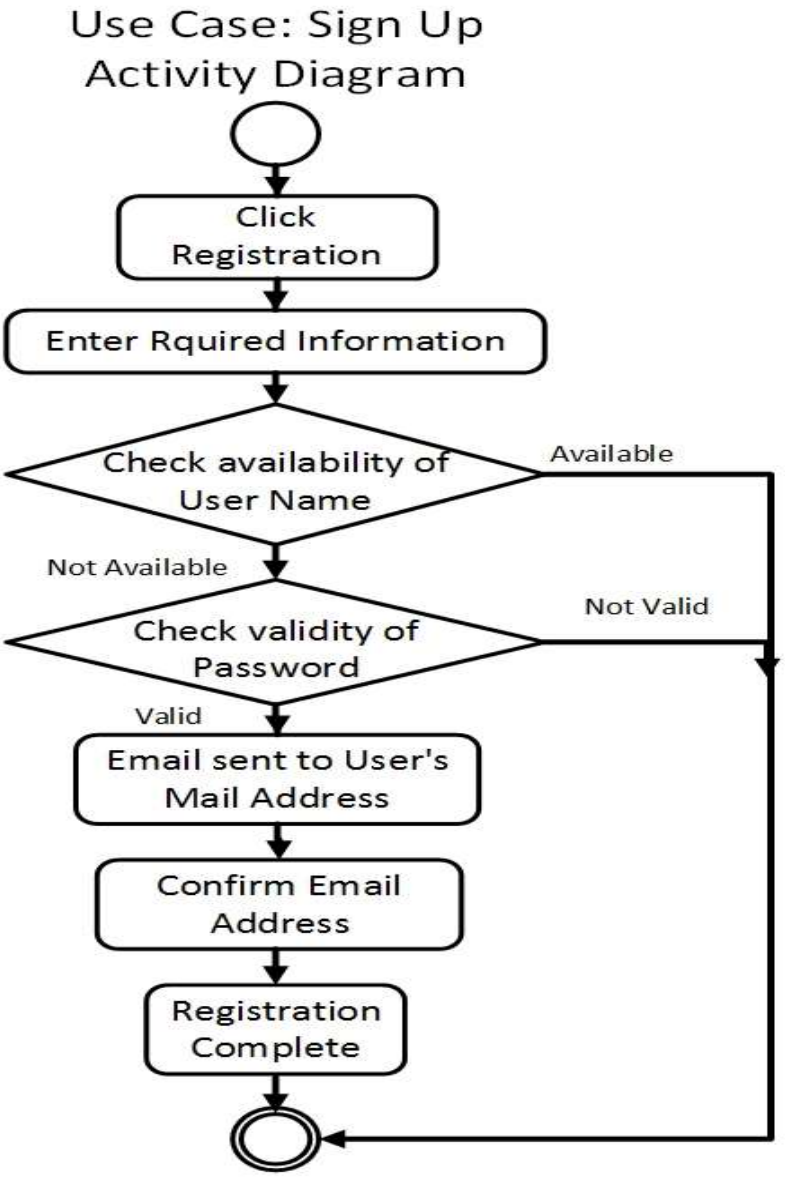


Figure 8: Activity for Sign Up

Swim lane diagram of a specific activity diagram shows the responsibilities of each actor dividing them into lanes. From this diagram we can improve our understanding about how the system works and which actors play what role.

Following is the swim lane of use case Sign Up.

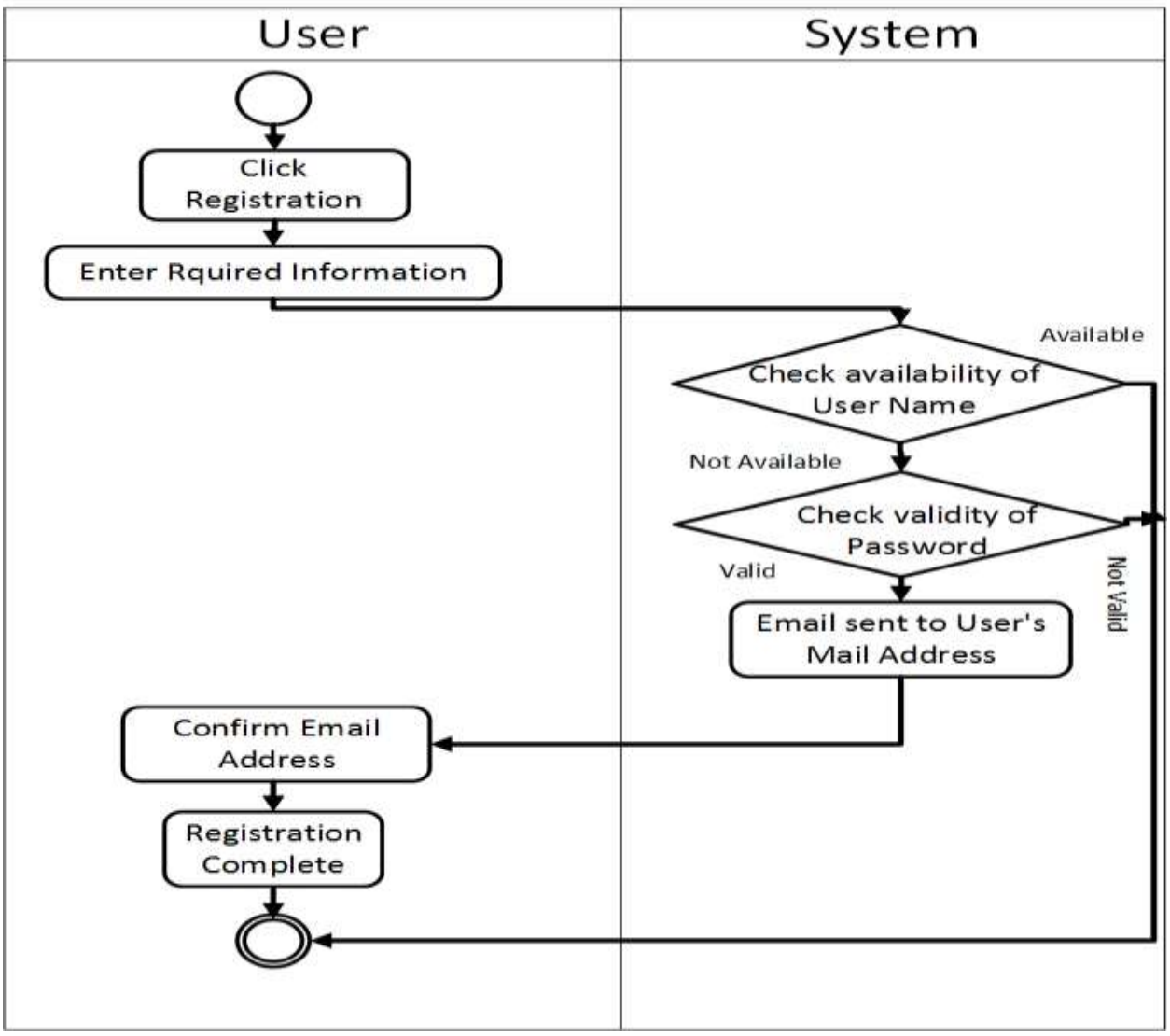


Figure 9: Swim lane for Sign Up

Following is the activity diagram of use case Sign In.

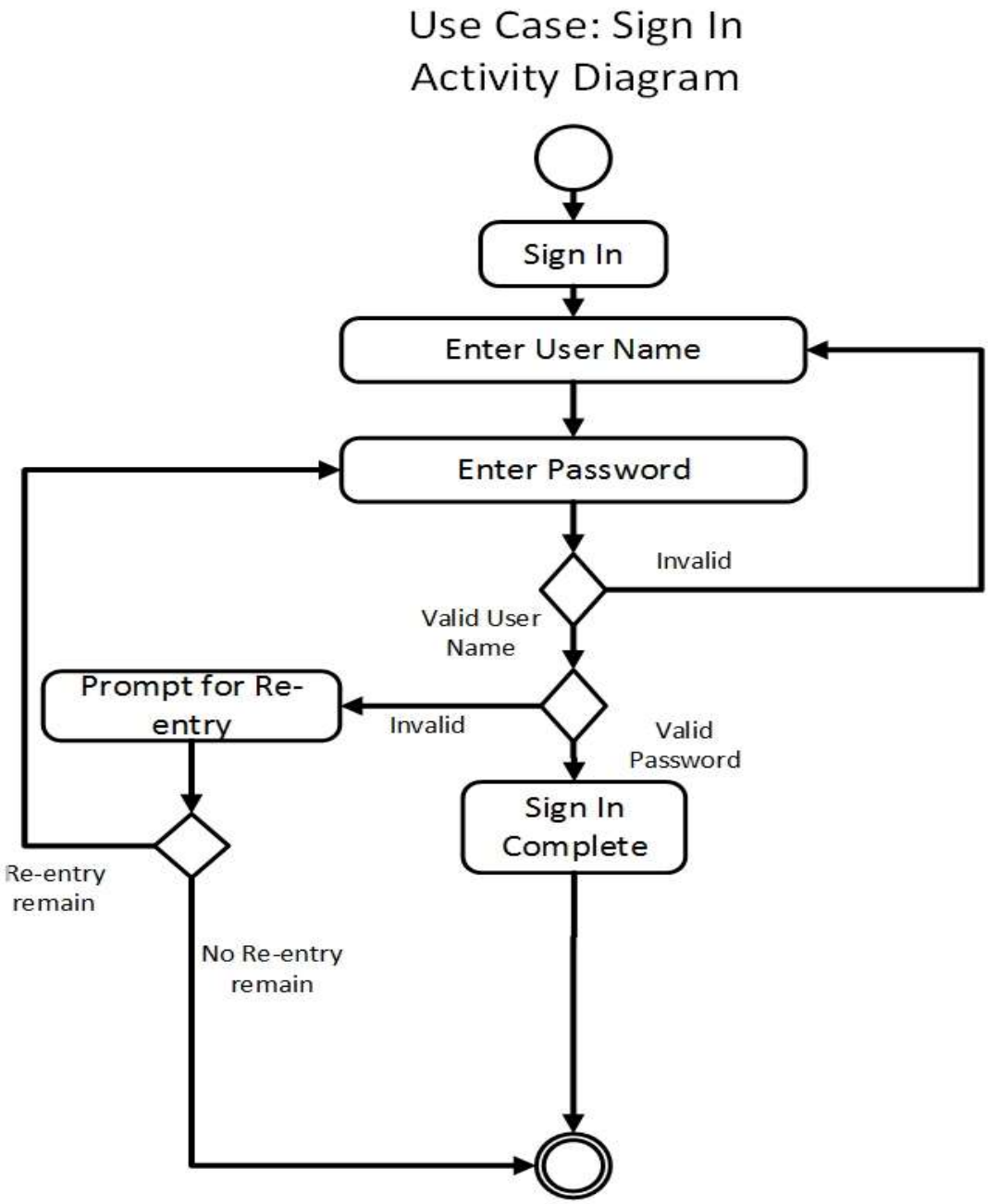


Figure 10: Activity of Sign In

Following is the swim lane diagram of use case Sign In.

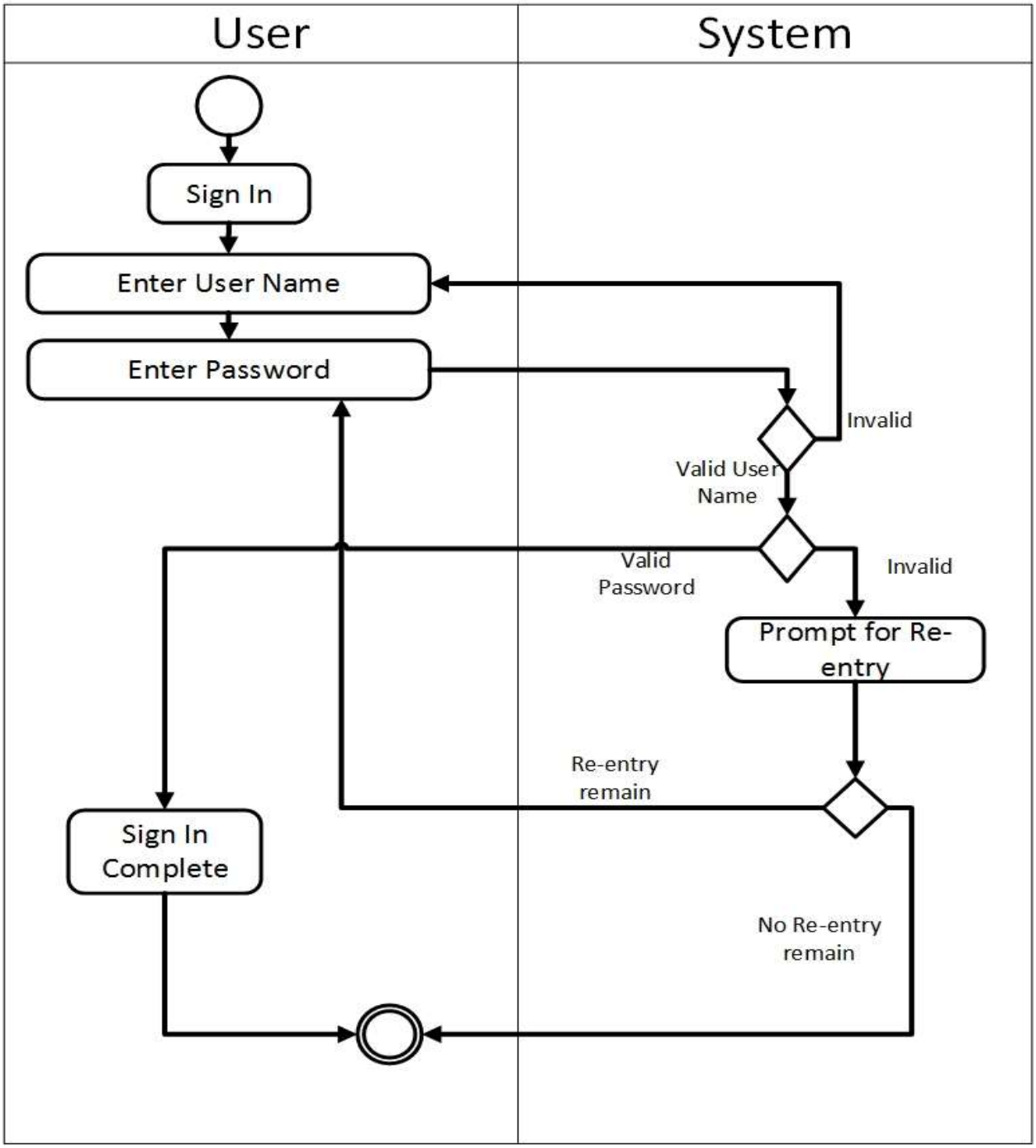


Figure 11: Swim lane for Sign In

Following is the activity diagram of use case Sign out.

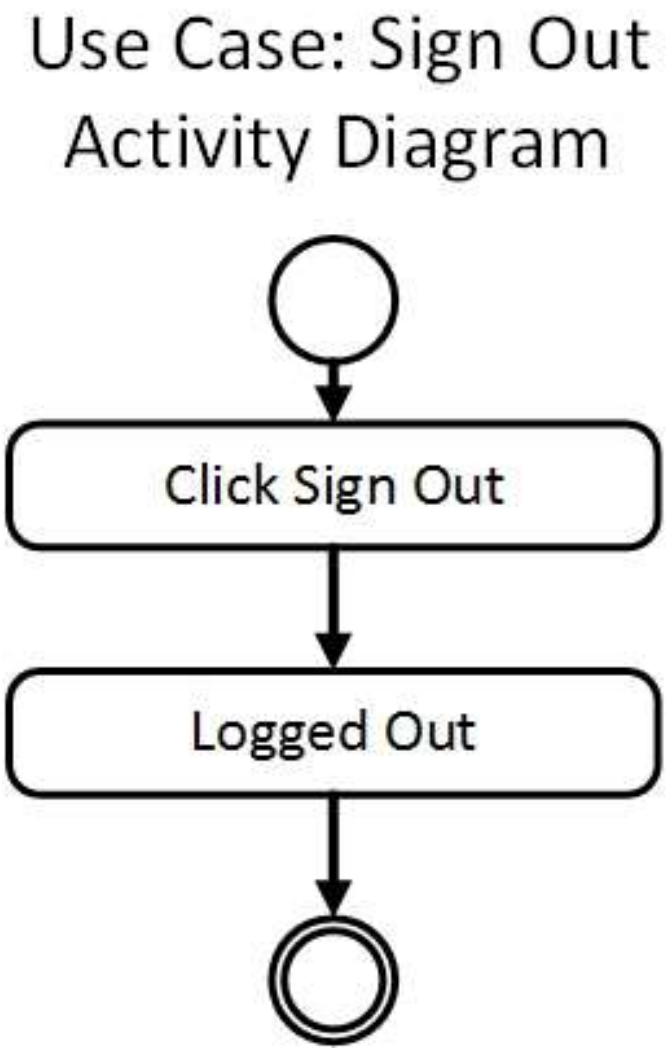


Figure 12: Activity for Sign Out

Following is the swim lane diagram of use case Sign out

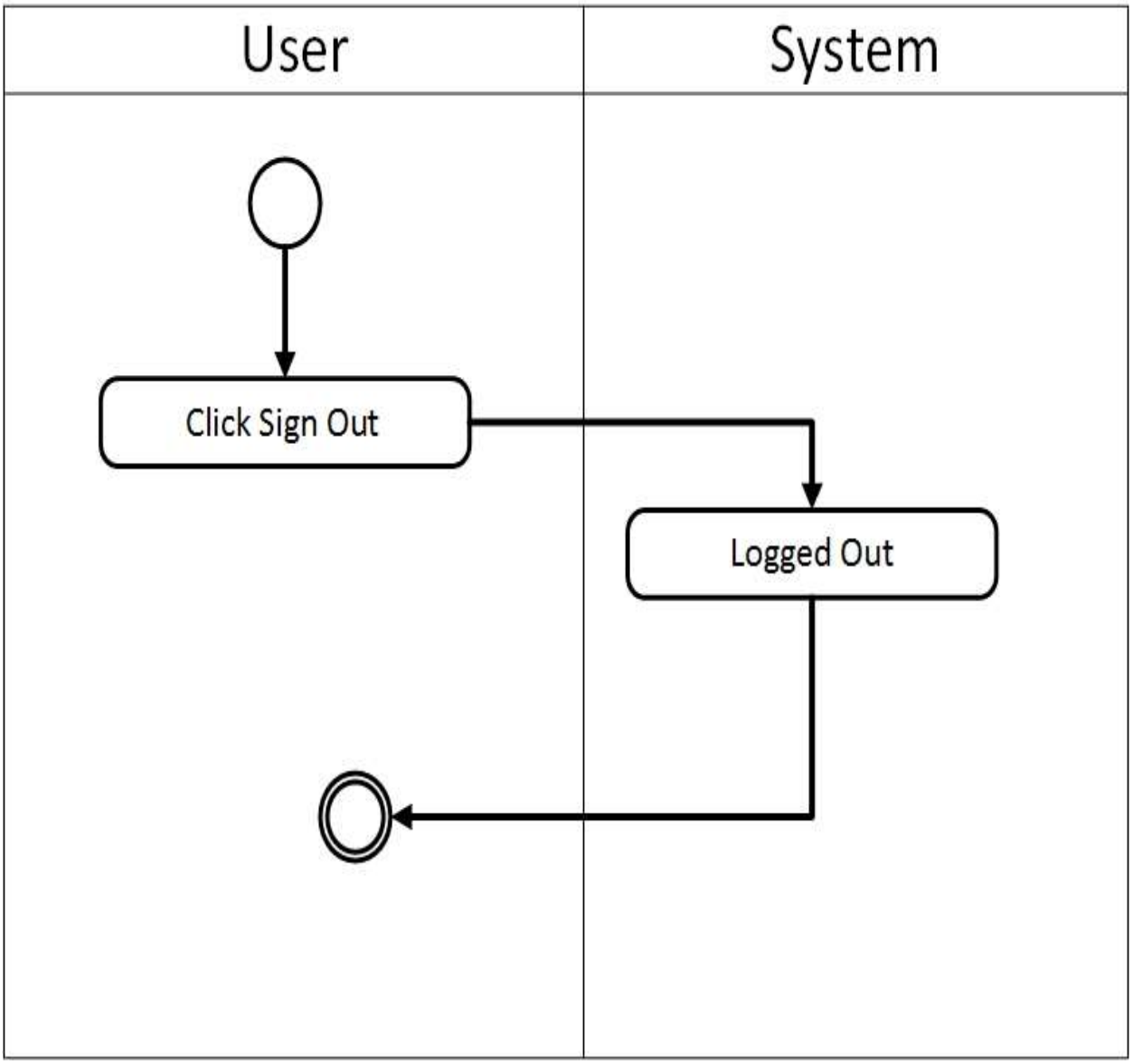


Figure 13: Swim lane for Sign Out

Following is the activity diagram of use case Bill Creation.

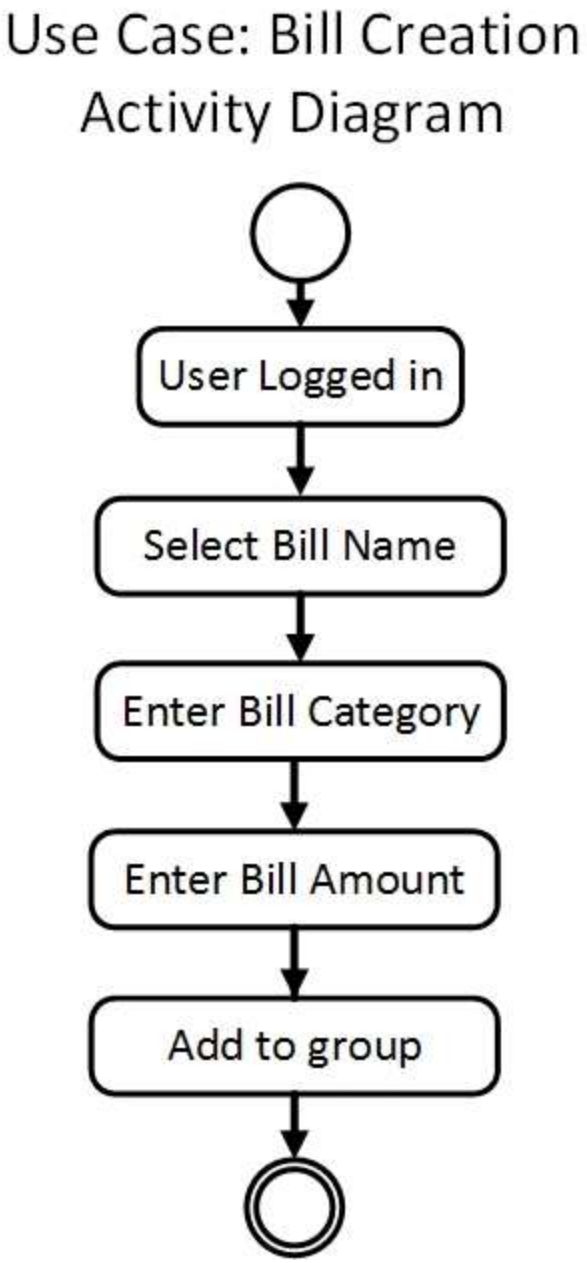


Figure 14: Activity for Bill Creation

Following is the swim lane of use case Edit Info.

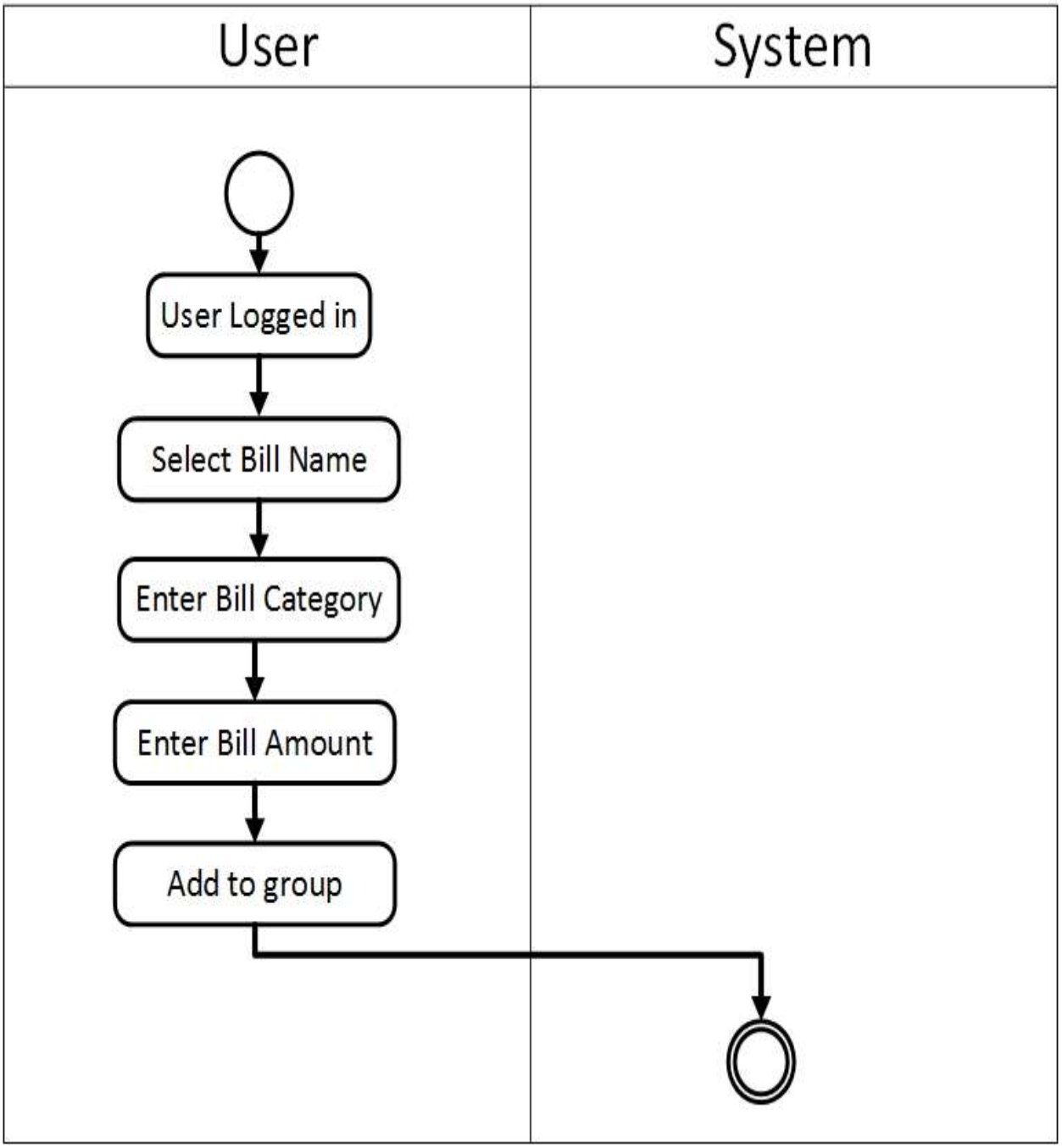


Figure 15: Swim lane for Bill Creation

Following is the activity diagram of use case Bill Approval.

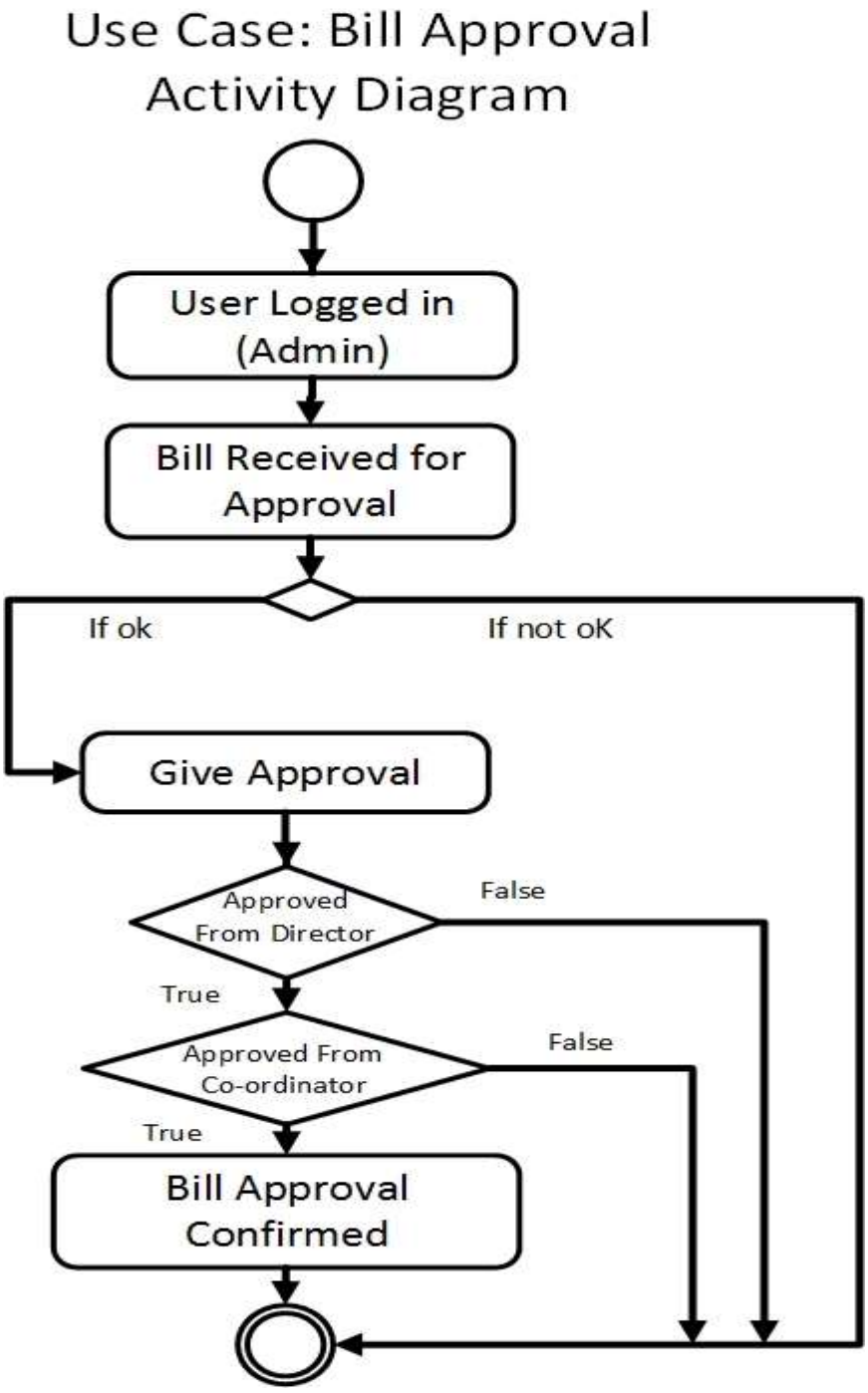


Figure 16: Activity diagram for Bill Approval

Following is the swim lane of use case Bill Approval.

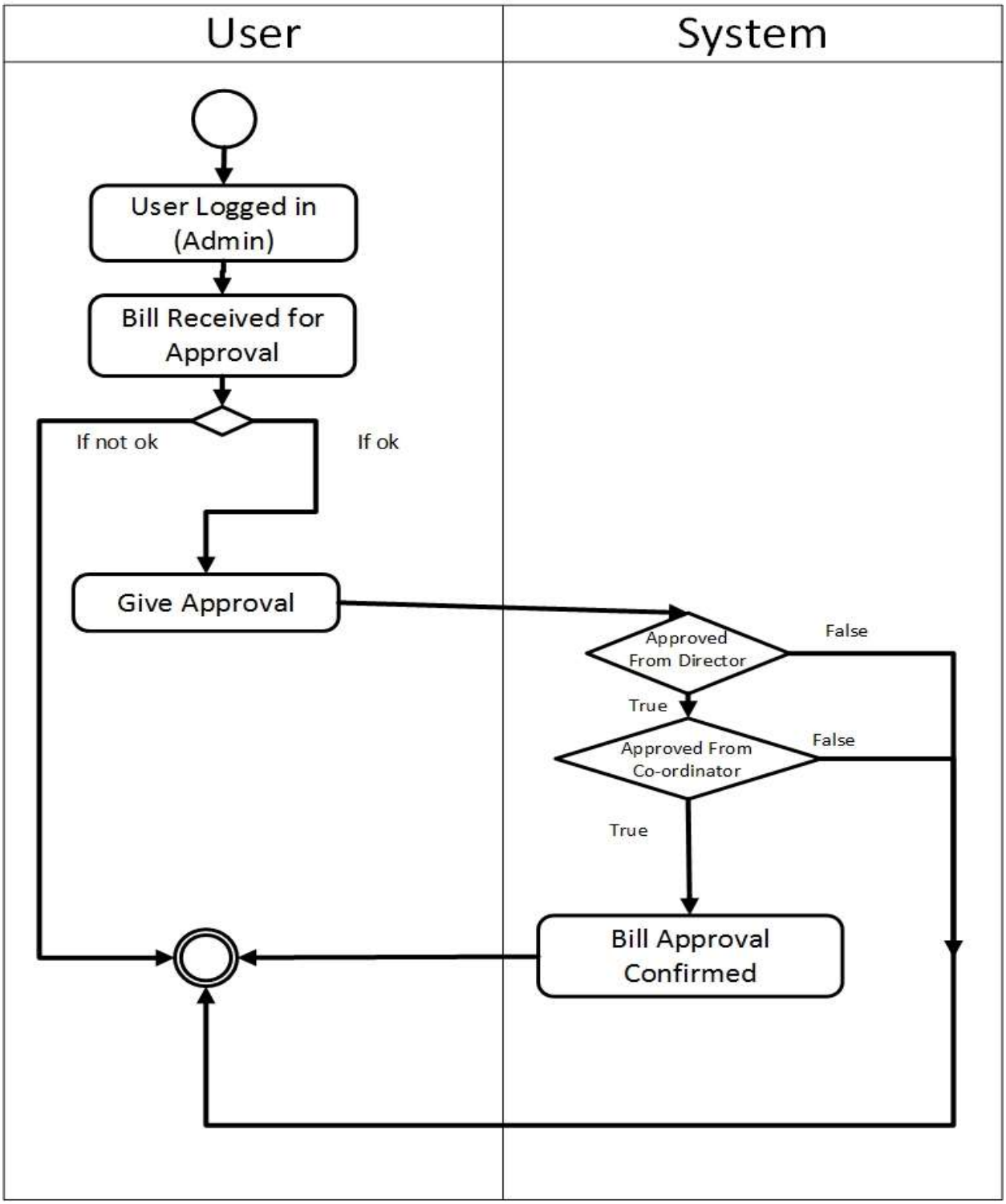


Figure 17: Swim lane for Bill Approval

Following is the activity diagram of use case Configuration.

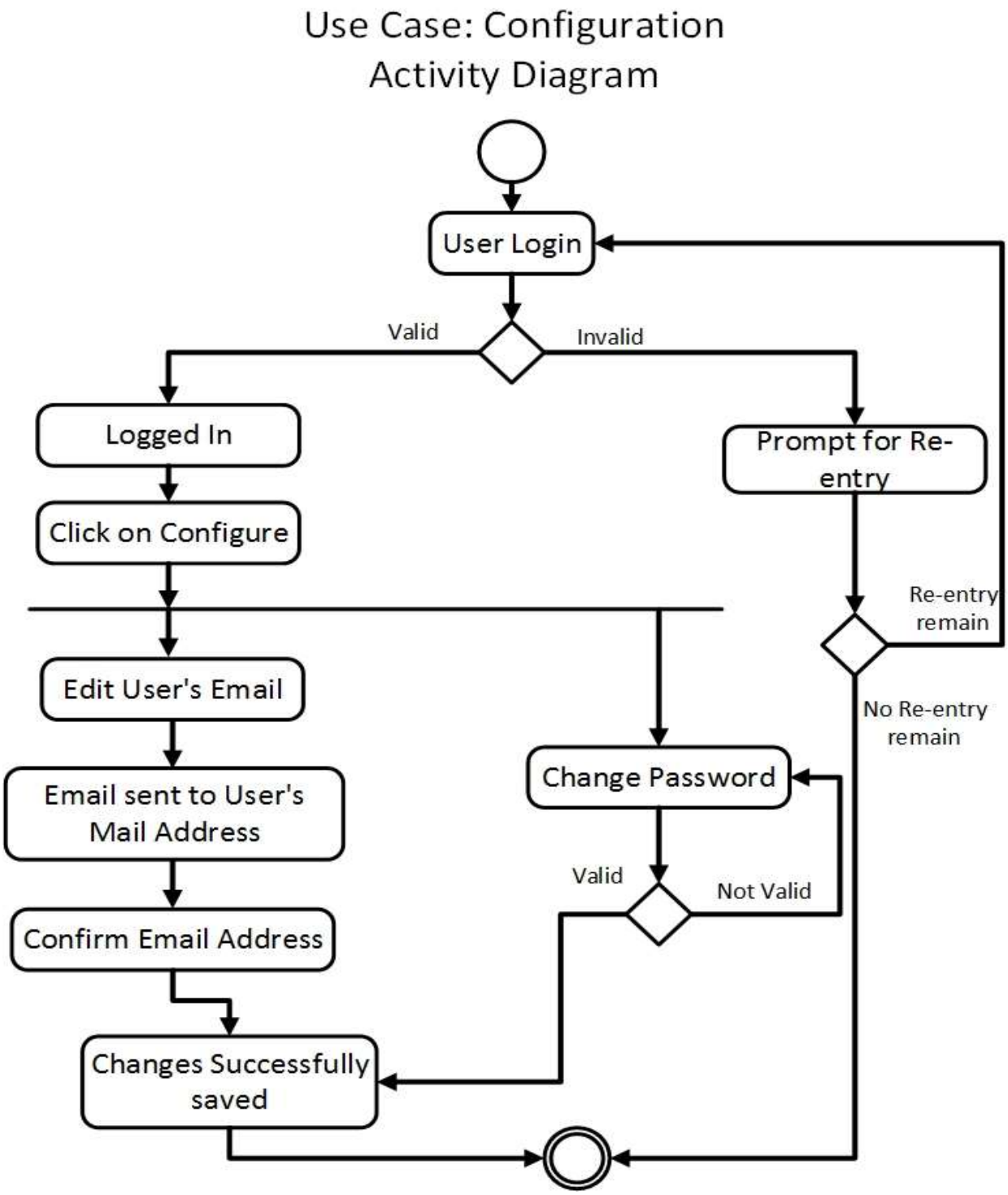


Figure 18: Activity diagram for Configuration

Following is the swim lane diagram of use case Configuration.

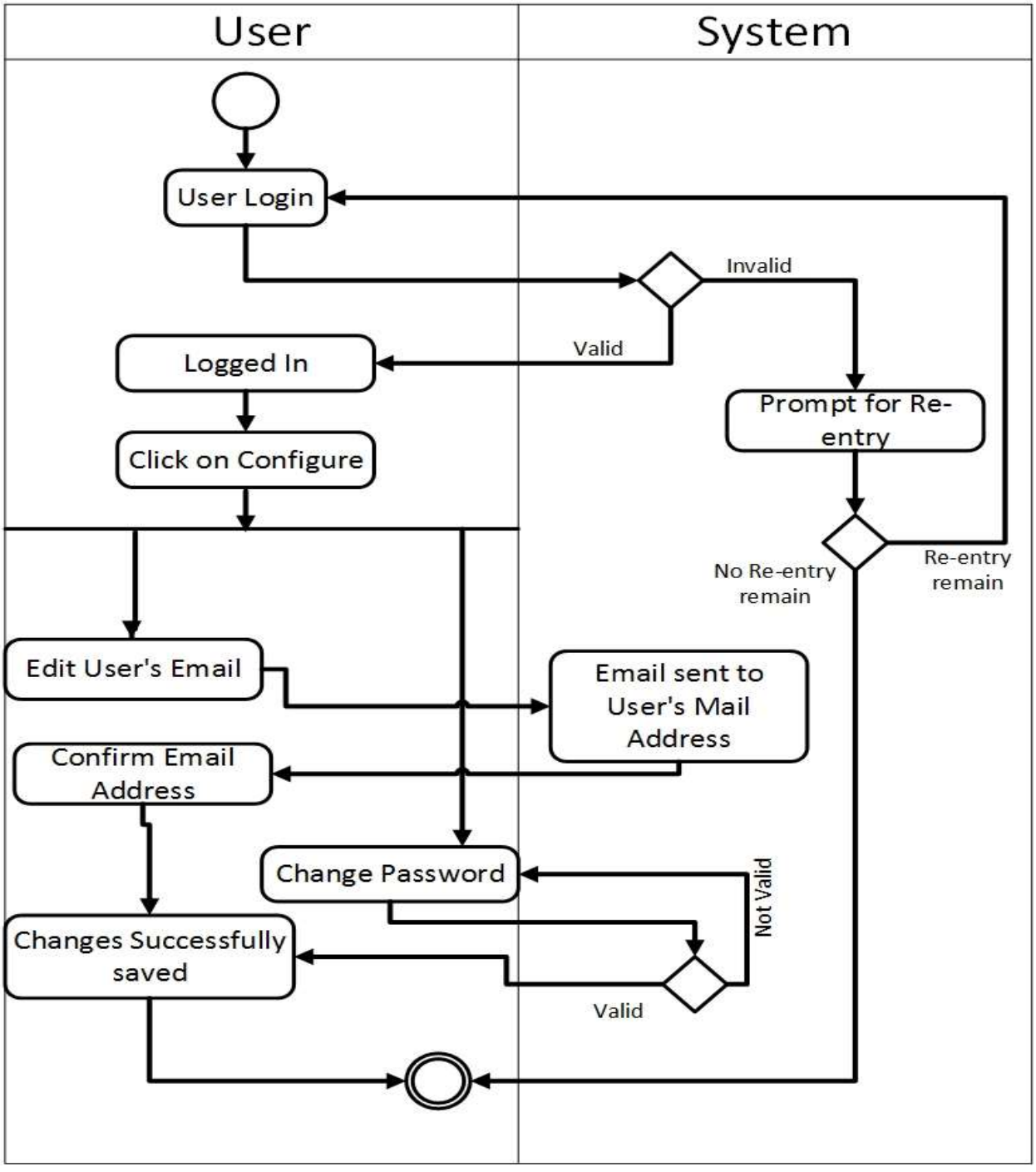


Figure 19: Swim lane diagram for Configuration

Following is the activity diagram of use case Configuration (change user type)

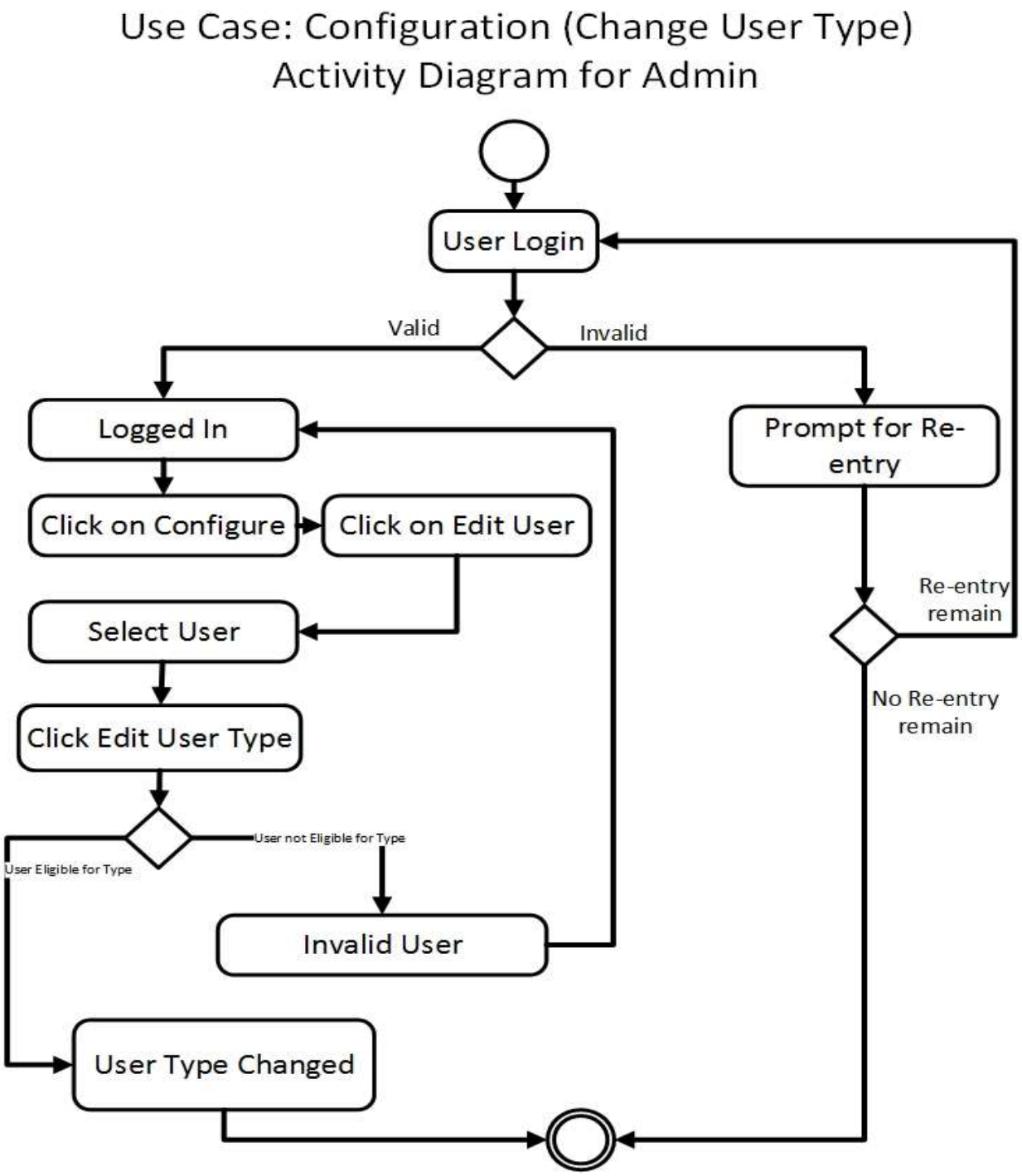


Figure 20: Activity for Configuration (for admin)

Following is the swim lane diagram of use case Configuration (change user type).

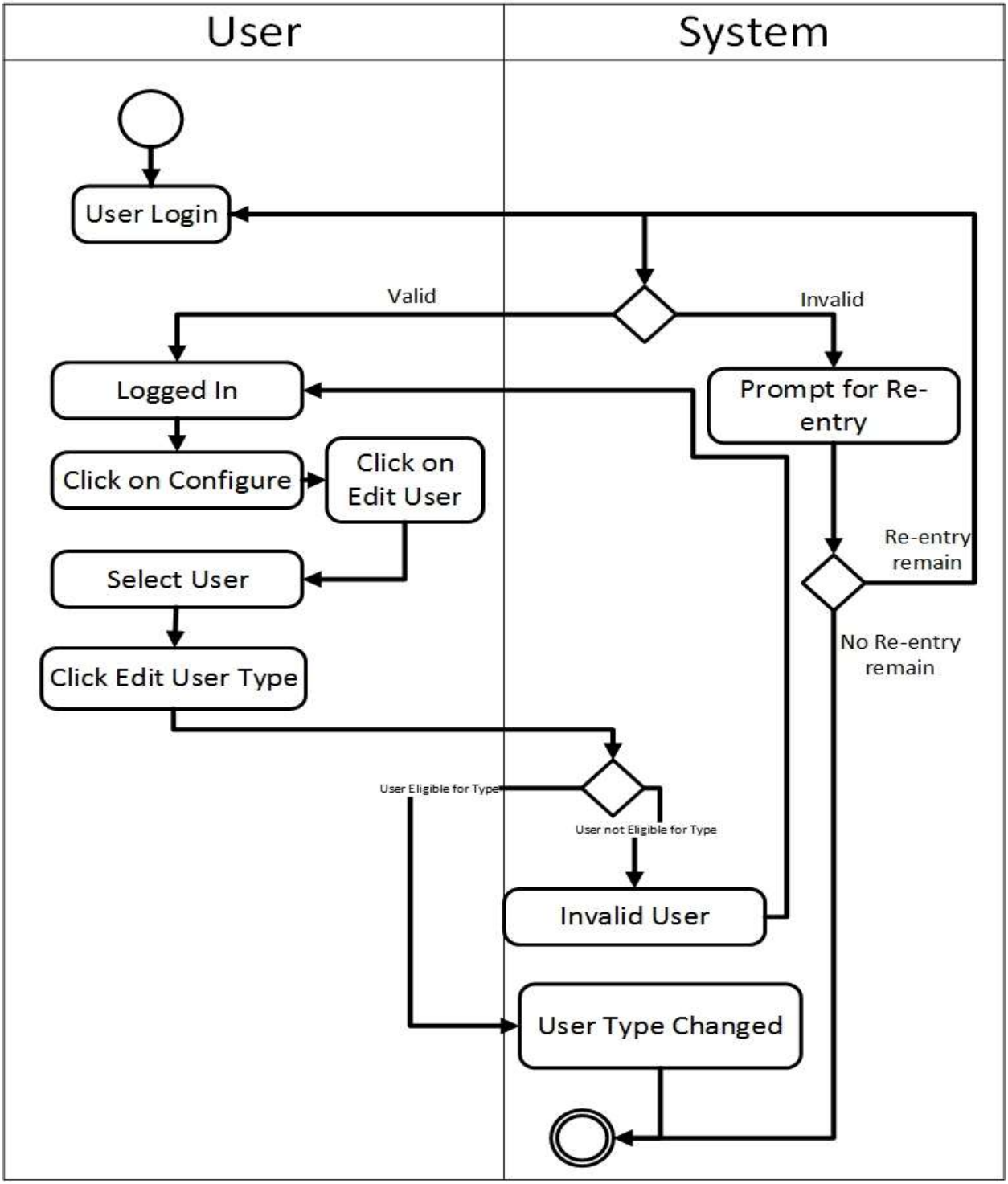


Figure 21: Swim lane diagram for Profile Post

5.5 Conclusion

This type of scenario based modeling helps to improve the idea about our project from user perspective. Each diagram will help the user to understand the system more closely. They can define from these diagrams whether the product will meet their need or fulfill their usability.

Chapter 6: Data Model

In this chapter we will discuss about the data models of our system.

6.1 Data Modeling Concept

If software requirements include the necessity to create, extend or interact with a data base or complex data structures need be constructed and manipulated, then the software team choose to create data model as part of overall requirements modeling. The entity-relationship diagram (ERD) defines all data objects that are processed within the system, the relationships between the data objects and the information that how the data objects are entered, stored, transformed and produced within the system.

6.2 Data Objects

A data object is representation of composite information that must be understood by software. Here, composite information means that has a number of different properties or attributes. A data object can be an external entity, a thing, an occurrence, a role, an organizational unit, a place or a structure.

6.2.1 Grammatical parsing & analysis

We identified all the nouns whether they are in problem space or in solution space from our usage scenario and categorized them according to their attributes.

Nouns	Problem space/solution space	Attributes
1. Billing System	P	
2. Admin	S	22, 27, 28, 29, 30
3. Director	S	22, 27, 28, 29, 30
4. Coordinator	S	22, 27, 28, 29, 30
5. Supreme Authority	P	
6. Category	S	23, 24, 26
7. Document of Bill	S	22, 23, 24, 25, 26
8. Validation	P	
9. User	S	22, 27, 28, 29, 30
10. Approval	P	
11. Admission	P	
12. Examination	P	
13. Extra Curriculum Activities	P	
14. Picnic	P	
15. Script Checking	P	
16. Overtime	P	
17. Food	P	
18. Training Program	P	
19. Regular Course	P	
20. Short Course	P	
21. Maintenance	P	
22. User Id	S	
23. Bill Id	S	
24. Bill Name	S	
25. Bill Amount	S	
26. Category Id	S	
27. Password	S	
28. Email Id	S	
29. User Type	S	
30. User Name	S	

6.2.2 Identify Data Objects

Nouns having attributes are selected as data object. Those who doesn't have any attributes have covered under the data objects.

Data Object: Admin

Attributes:

- User Id
- Email
- Password
- User Name
- User Type

Data Object: Director

Attributes:

- User Id
- Email
- Password
- User Name
- User Type

Data Object: Coordinator

Attributes:

- User Id
- Email
- Password
- User Name
- User Type

Data Object: User

Attributes:

- User Id
- Email
- Password
- User Name

Data Object: Document of Bill (Bill)

Attributes:

- Bill Id
- Bill Name
- Bill Amount
- Category Id
- User Id

Data Object: Category

Attributes:

- Bill Id
- Bill Name
- Category Id

Here, we found that Director, Coordinator and Admin have same attributes. So we can consider a new data object named “Admin Panel” which will hold the attributes of these data objects.

Data Object: Admin Panel

Attributes:

- User Id
- Email
- Password
- User Name
- User Type

6.2.3 Data Object Relation

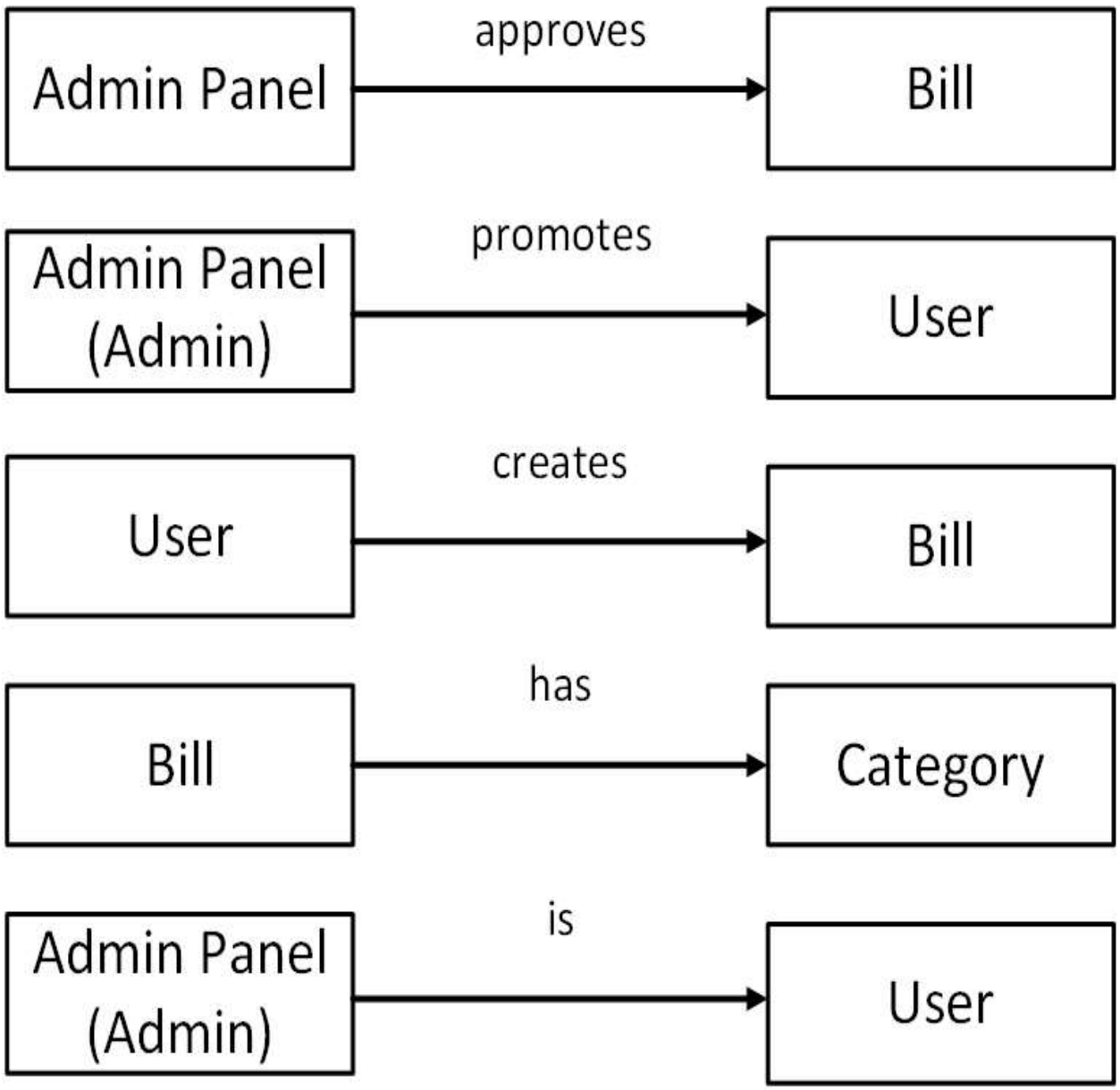


Figure 22: Data Object Relation Diagram

6.2.4 E-R Diagram

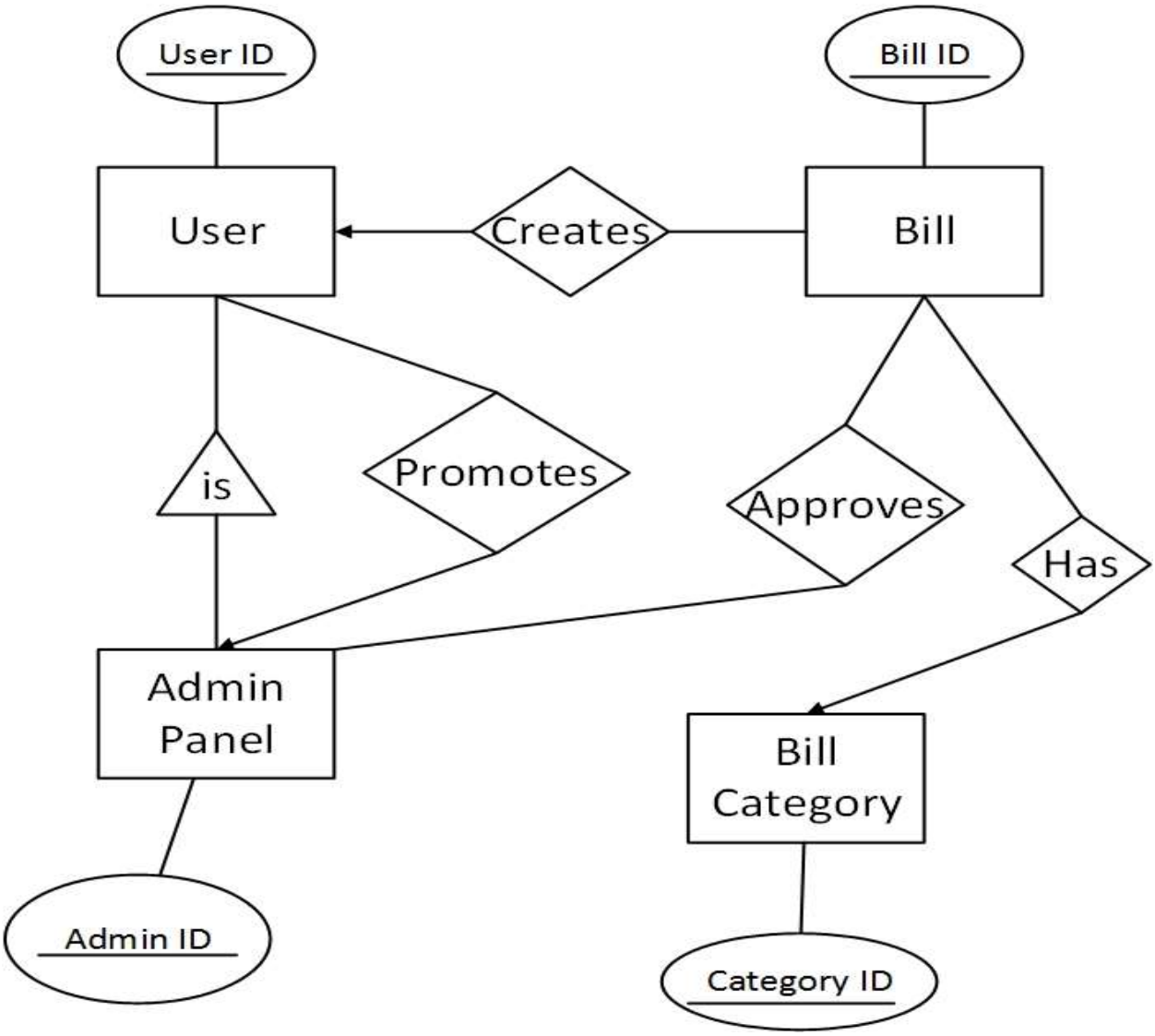


Figure 23: E-R Diagram

6.3.5 Schema Form (Tables)

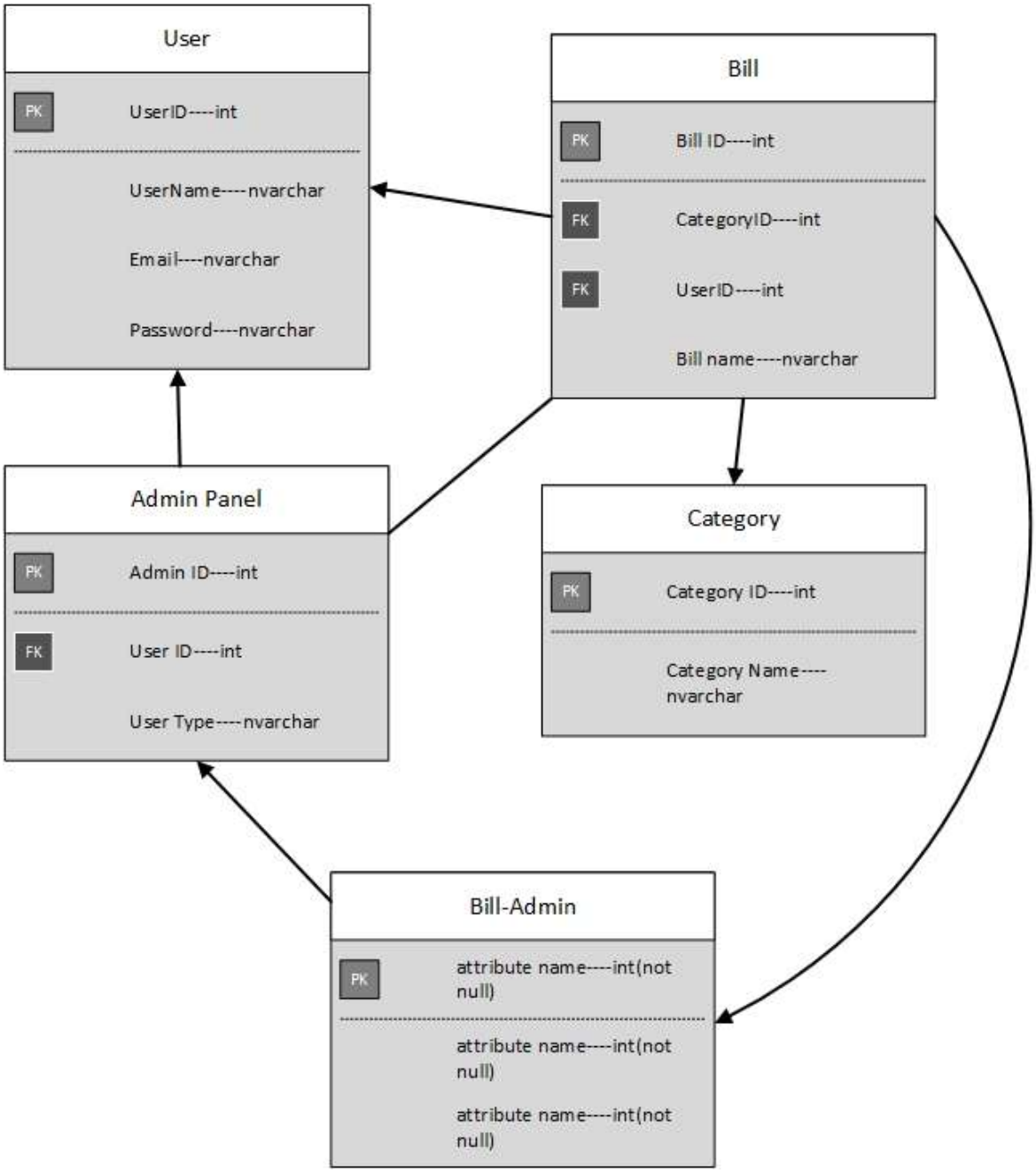


Figure 24: Data Schema

6.3 Conclusion

Data modeling serves the purpose of the technical view of the system. It is helpful for the developers to design the database. This modeling gives us the idea how data will be manipulated, how to store an information of our system in the database or how to retrieve information from the database.

Chapter 7: Class Based Model

This Chapter is intended to describe class based modeling of Billing System Website.

7.1 Class Based Modeling Concept

Class-based modeling represents the objects that the system will manipulate, the operations that will applied to the objects, relationships between the objects and the collaborations that occur between the classes that are defined.

7.2 Identifying Analysis Class

To identify our analysis class, we firstly grammatically parsed all the nouns and then categorized them according to general classification and selection criteria.

Following are the steps we used to analysis the classes for our system.

Step-1: Grammatical parsing (noun identifying) and categorizing using general classification:

External Entities	User
Things	Password, User Name, Email, Bill Name, Category, Bill Amount, User Type, Bill
Occurrence or events	-
Roles	Admin, Director, Coordinator
Organizational units	-
Places	-
Structure	-

After categorizing noun using general classification, we follow the next step to identify preliminary classes. In this step, we find the noun which fulfill at least three selection criteria or is an essential requirement.

Step-2: Selection Criteria:

1. Retained information
2. Needed services
3. Multiple attributes
4. Common attributes
5. Common operations
6. Essential requirements

Potential Class	Characteristic Number That Applies	Remarks
User	2,3,6	Accepted
Admin	2,3,6	Accepted
Director	2,3,6	Accepted
Coordinator	2,3,6	Accepted
Category	1,3	Accepted
Password	1	Rejected
User Name	1	Rejected
Bill Name	1	Rejected
Bill Amount	1	Rejected
Email	1,2	Rejected
User Type	1	Rejected
Bill	1,3,6	Accepted

Nouns that hold essential requirement we took them essential for our system and include them in our analysis classes.

Preliminary Classes:

1. User
2. Director
3. Coordinator
4. Admin
5. Bill

Attributes Selection:

So, the retrieved attributes are listed below with the preliminary classes.

Class Name	Attributes
User	User Name User Id Password Email
Director	User Name Password User Id User Type
Admin	User Name Password User Id User Type
Coordinator	User Name Password User Id User Type
Bill	Bill Name Bill Id Bill Amount

Step-3: Method Identification:

To identify methods, we will find out the verbs of the user story and will identify which one is in scope of the problem and which one is out of scope.

Verb Detection:

No	Verb	Remark
1.	Promote	Yes
2.	Demote	Yes
3.	Sign Up	Yes
4.	Sign In	Yes
5.	Sign Out	Yes
6.	Provide	Out of Scope
7.	Select	Out of Scope
8.	Approve	Yes
9.	Enter	Out of Scope
10.	Pdf Generate	Yes
11.	Notify	Yes
12.	Bill Create	Yes
13.	Reject	Yes
14.	Add	Yes
15.	Reject	Yes
16.	Send	Out of Scope
17.	Check	Out of Scope
18.	Edit	Yes

So, the methods of the classes are listed below.

Class Name	**Methods**
User	Sign Out () Sign In () Sign Up () Pdf Generate () Edit ()
Admin	Sign Out () Sign In () Sign Up () Promote () Demote () Approve () Reject ()
Director	Sign Out () Sign In () Sign Up () Approve () Reject ()
Coordinator	Sign Out () Sign In ()

	Sign Up () Approve () Reject ()
Bill	Edit Bill () Delete Bill () Generate Bill ()

Class Card:

User	
Attributes	**Methods**
User ID User Name Password Email	Sign Out () Sign In () Sign Up () Pdf Generate () Edit ()
Responsibilities	**Collaborative Class**
Creates bill	Bill
Generates pdf	Admin, Director, Coordinator

Admin	
Attributes	**Methods**
User ID User Name Password Email User Type	Sign Out () Sign In () Sign Up () Promote () Demote () Approve ()
Responsibilities	**Collaborative Class**
Promotes coordinator	Coordinator, User
Demotes coordinator	Coordinator
Approves bill	Bill
Rejects bill	Bill

Director	
Attributes	**Methods**
User ID User Name Password Email User Type	Sign Out () Sign In () Sign Up () Approve () Reject ()

Responsibilities	**Collaborative Class**
Approves bill	Bill
Rejects bill	Bill

Coordinator	
Attributes	**Methods**
User ID User Name Password Email User Type	Sign Out () Sign In () Sign Up () Approve () Reject ()
Responsibilities	**Collaborative Class**
Approves bill	Bill
Rejects bill	Bill

Bill	
Attributes	Methods
Bill Id Bill Name Category	Edit Bill () Delete Bill () Generate Bill ()

Responsibilities	Collaborative Class
Generates bill	User
Deletes bill	User
Edit bill	User

After analyzing each classes, we found that Admin, Coordinator, Director have similar attributes and methods. So we can consider a super class named "Admin Panel" of them which will hold common attributes and common methods. Other classes will be sub class of this super class.

Common Attributes: User ID, User Name, Password, Email, User Type

Common Methods: SignOut(), SignIn(), SignUp(), Approve(), Reject()

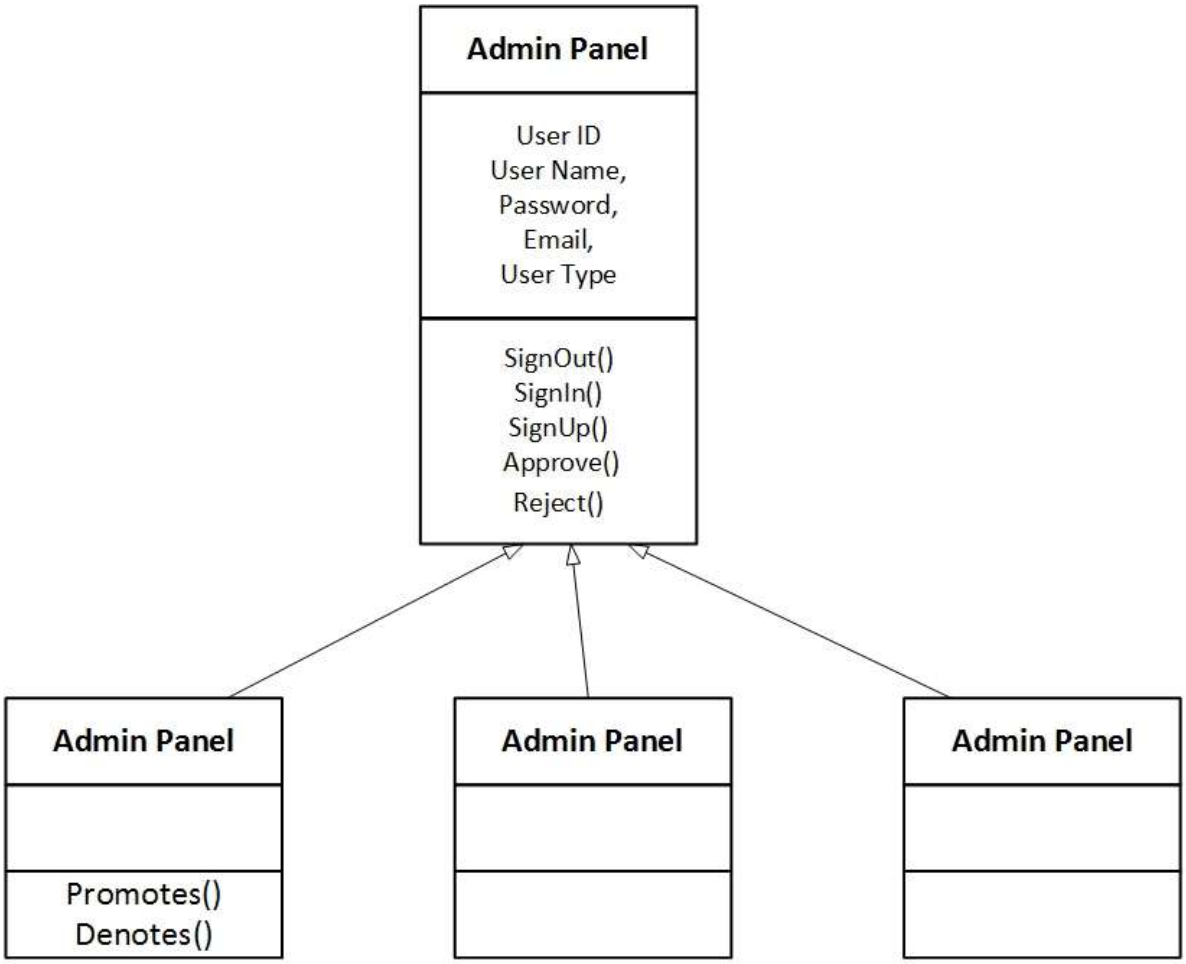


Figure 25: "Admin Panel" Super Class

7.2.1 Class Responsibility Collaboration (CRC)

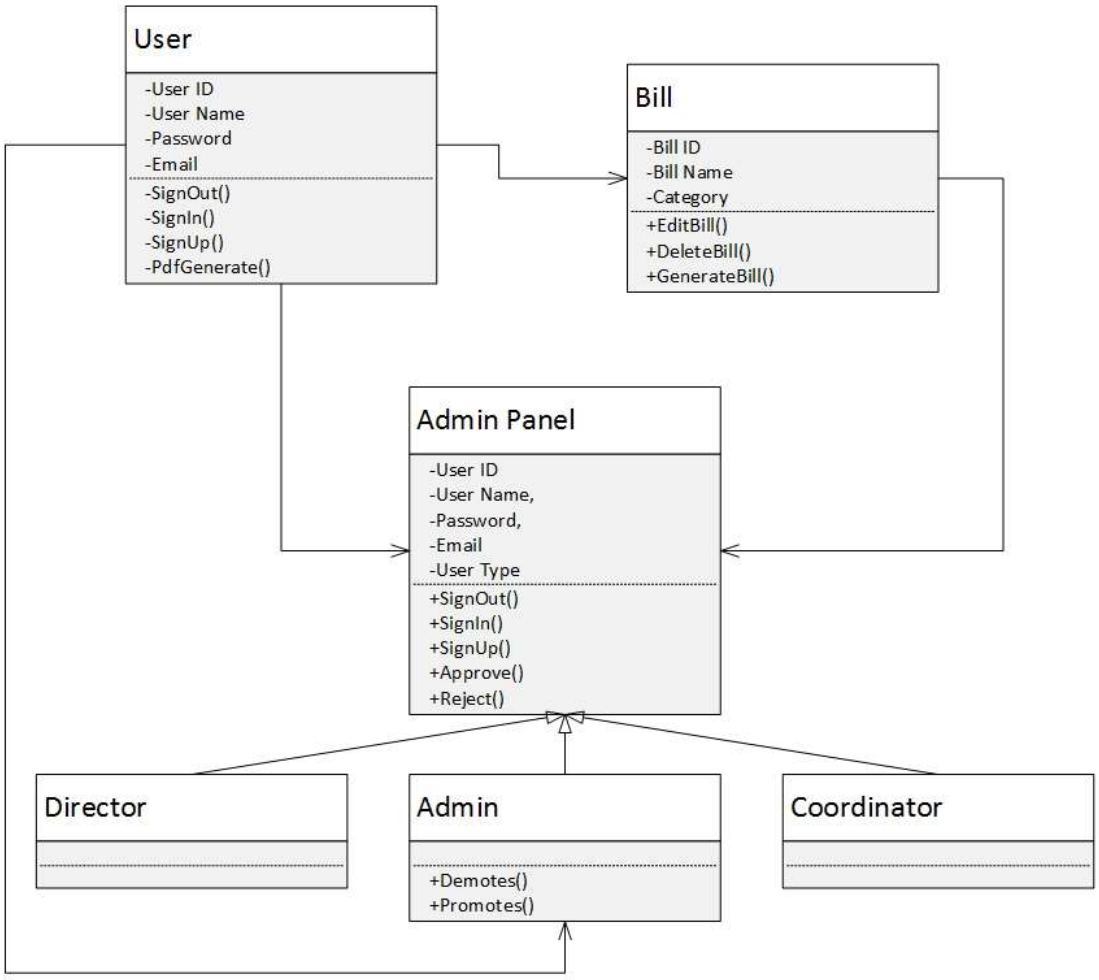


Figure 26: CRC Diagram

7.3 Conclusion

The elements of a class-based model include classes and objects, attributes, operations, class responsibility-collaborator (CRC) models, collaboration diagrams, and packages. This mainly represents the relationship between objects. Class based modeling normally helps a system to be developed in a way for future reusability.

Chapter 8: Flow Oriented Model

This chapter focuses on the flow oriented modeling.

8.1 Introduction

Although data flow-oriented modeling is perceived as an outdated technique by some software engineers, it continues to be one of the most widely used requirements analysis notations in use today. It provides additional insight into system requirements and data flow. The data flow diagram enables you to develop models of the information domain and functional domain. As the DFD is refined into greater levels of detail, you perform an implicit functional decomposition of the system.

8.2 Data Flow Diagram

The DFD takes an input-process-output view of a system. That is, data objects flow into the software, are transformed by processing elements, and resultant data objects flow out of the software. In the figures, data objects are represented by labeled arrows and transformations are represented by circles.

Level-0 DFD: The level 0 data flow diagram should depict the software/system as a single bubble. Here, Level-0 DFD is describing the overall system's input and output data.

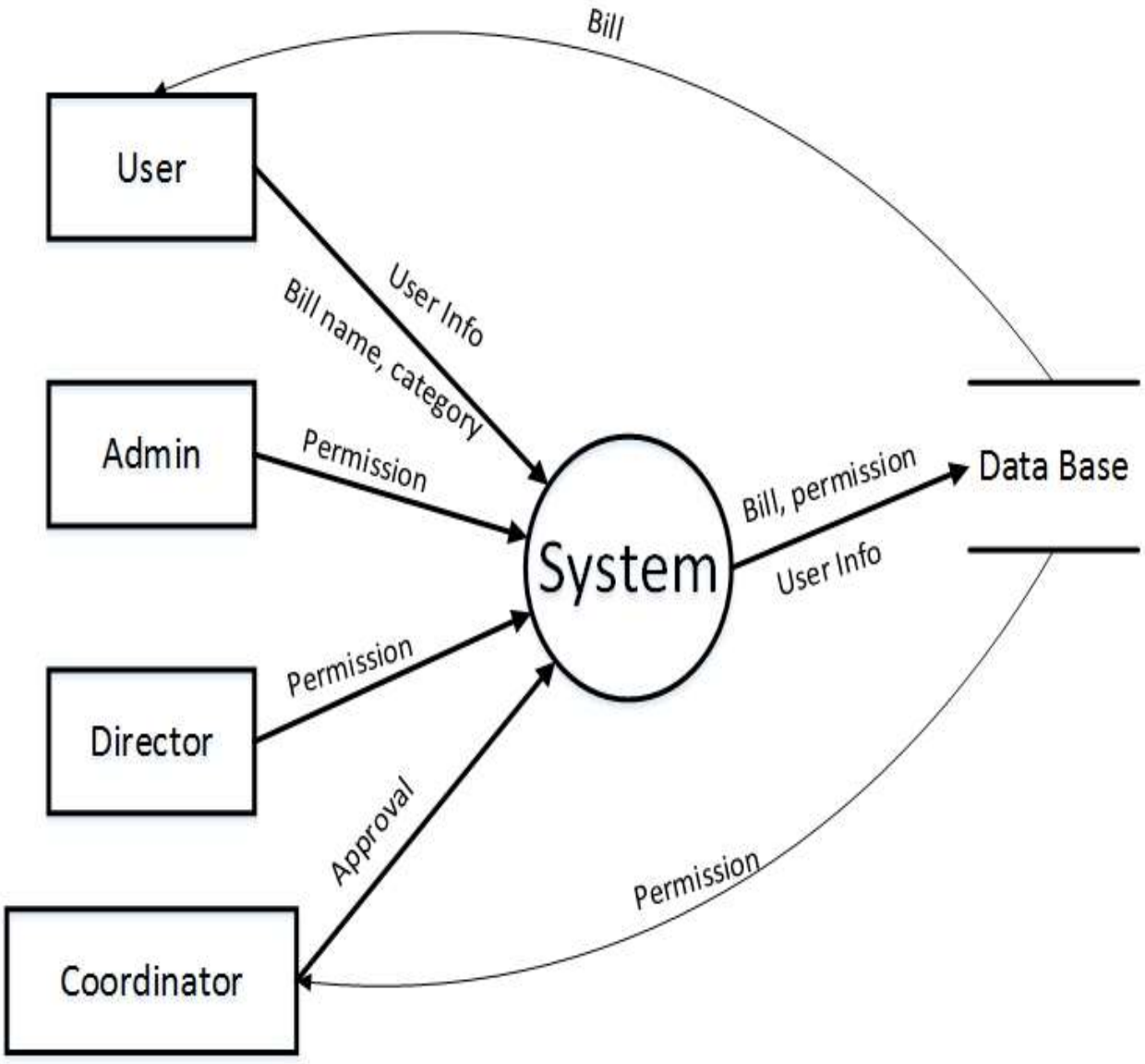


Figure 27: Level-0 DFD

Level-1 DFD: At DFD level 1 can be further refined into lower levels. This level-1 DFD derived from level-0 DFD.

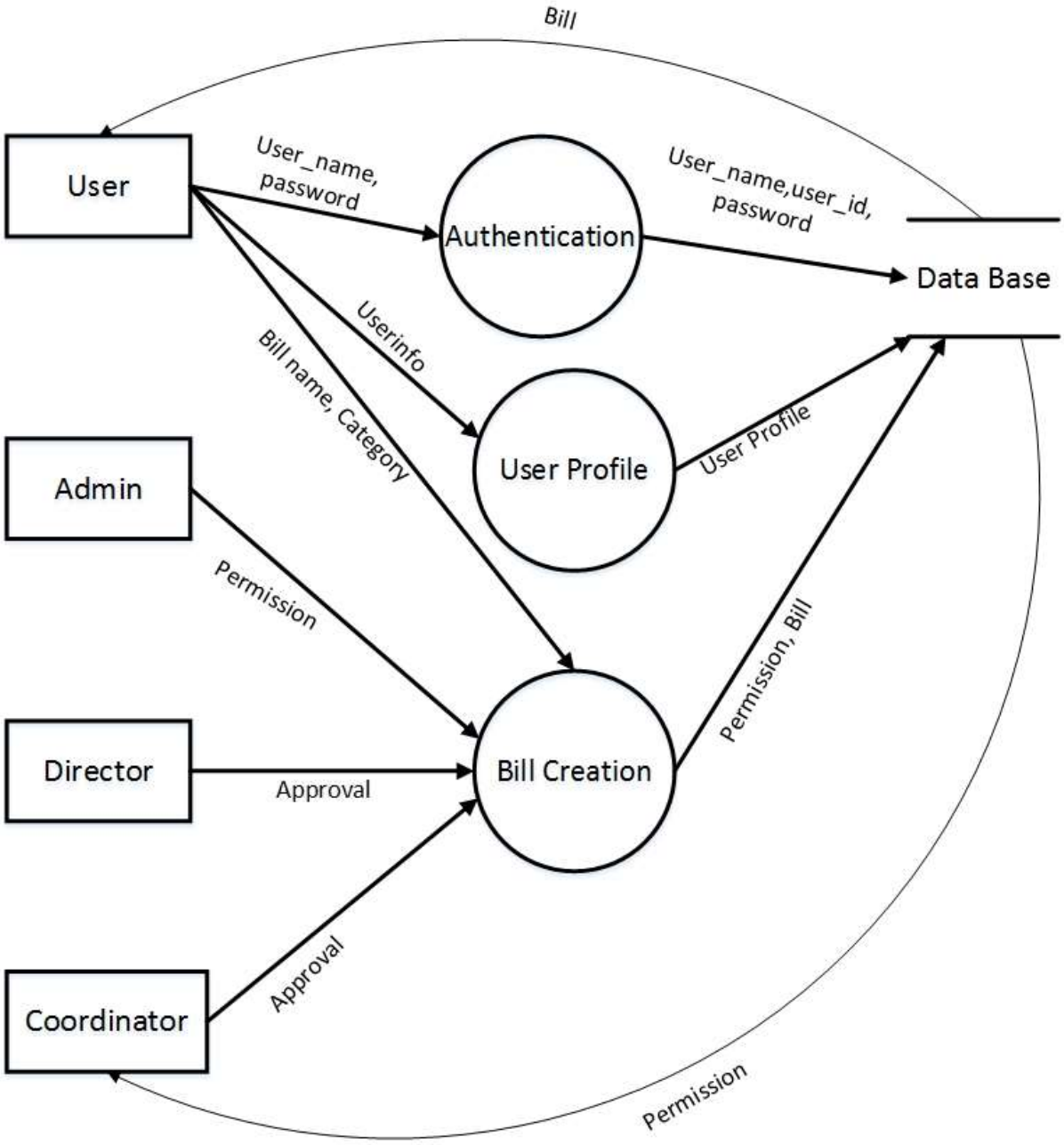


Figure 28: Level-1 DFD

Level-2 DFD: Every process of data flow can be described into many sub process. And thus we got level-2 DFD from the process Authentication in level-1 DFD.

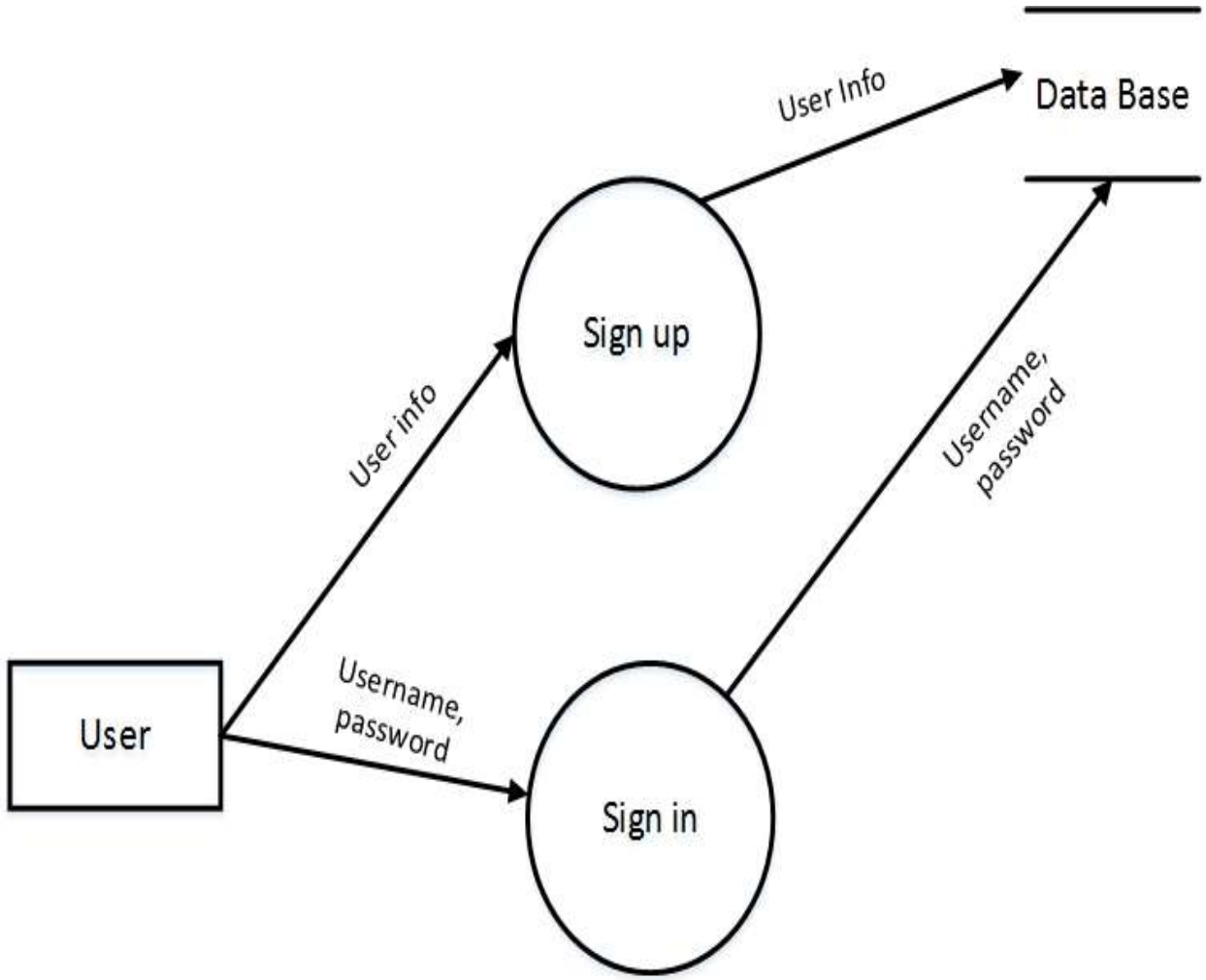


Figure 29: Level-2 DFD

8.3 Conclusion

The refinement of DFDs continues until each bubble performs a simple function. That is, until the process represented by the bubble performs a function that would be easily implemented as a program component.

Chapter 9: Behavioral Model

The behavioral model indicates how software will respond to external events. Two different behavioral representations are discussed in this chapter that follow. The first indicates how an individual class changes state based on external events and the second shows the behavior of the software as a function of time.

9.1 State Transition Diagram

State diagram represents active states for each class the events (triggers). For this we identified all the events, their initiators and collaborators.

Event	Initiator	Collaborations
1. Generating pdf	User	User, Admin Panel
2. Promotes user	Admin Panel	User
3. Demotes user	Admin Panel	User
4. Approves bill	Admin Panel	Bill
5. Rejects bill	Admin Panel	Bill
6. Generating bill	Bill	User
7. Editing bill	Bill	User
8. Deleting bill	Bill	User

STD for User: State transition diagram for User shows us actions and states for every action occurred in this class.

User:

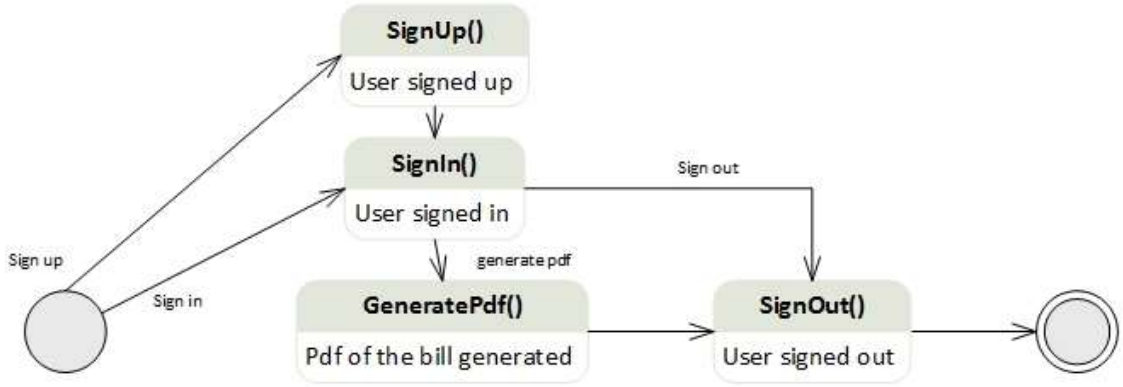


Figure 30: STD for User

STD for Bill: State transition diagram for Bill class shows us actions and states for every action occurred in this class.

Bill:

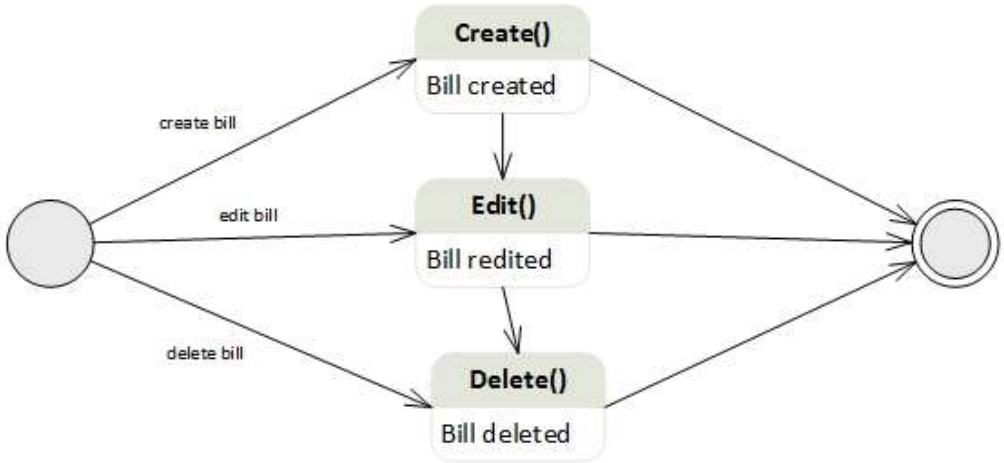


Figure 31: STD for Bill Class

STD for Admin Panel: State transition diagram for Admin-Panel class shows us actions and states for every action occurred in this class.

Admin Panel:

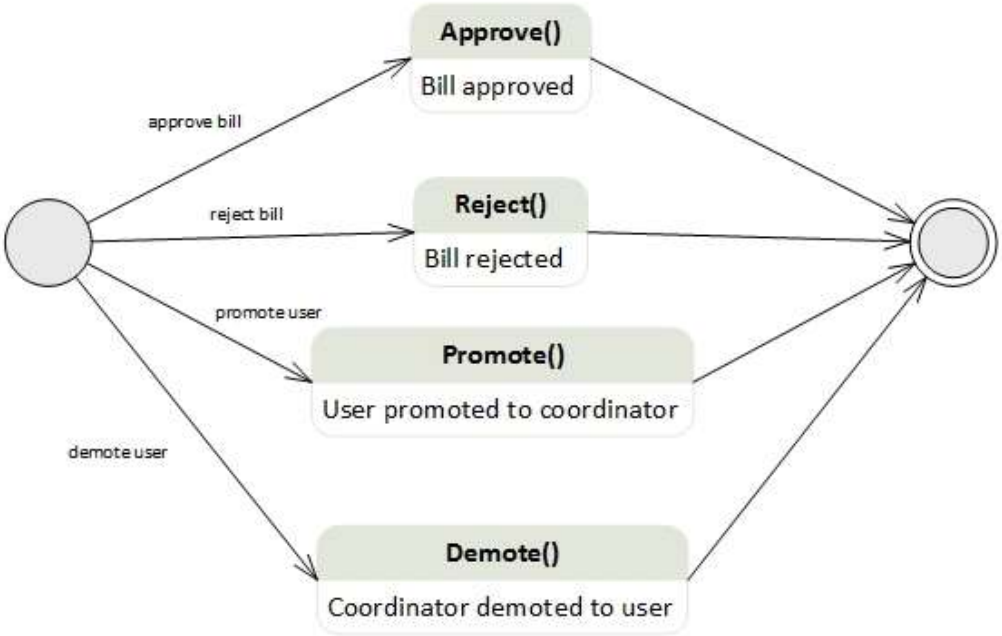


Figure 32: STD for Admin-Panel class

9.2 Sequence Diagram

The second type of behavioral representation, called a *sequence diagram* in UML, indicates how events cause transitions from object to object. Once events have been identified by examining a use case.

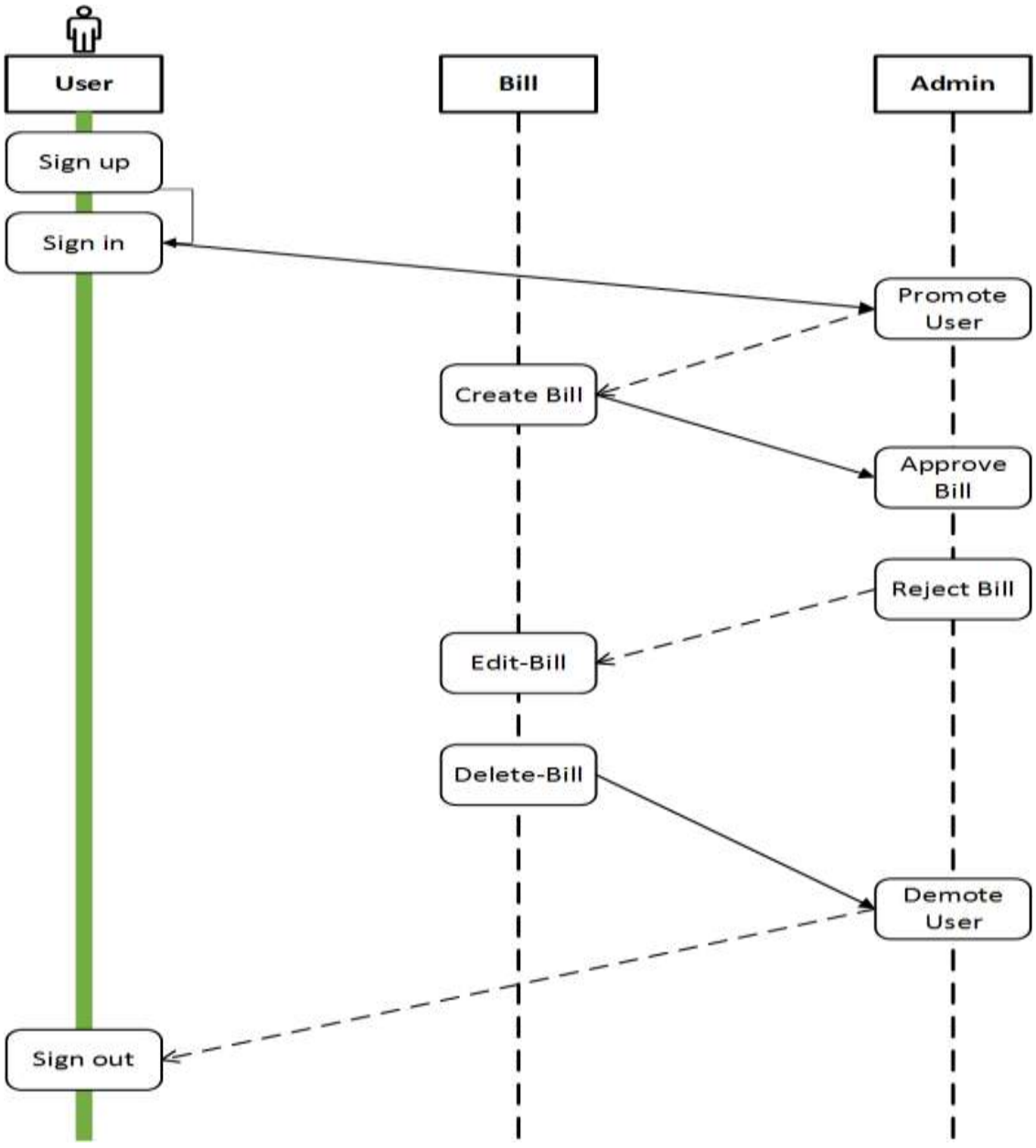


Figure 33: Sequence Diagram

9.3 Conclusion

Although the active state model provides useful insight into the "life history" of an object, it is possible to specify additional information to provide more depth in understanding the behavior of an object. A sequence diagram—a representation of how events cause flow from one object to another as a function of time. In essence, the sequence diagram is a shorthand version of the use case. It represents key classes and the events that cause behavior to flow from class to class.

Chapter 10: User Interface

This chapter described the user interface of our project “Billing System”.

10.1 Introduction

Our Billing System will definitely a user friendly application which sole intention is to give the user a complete feel of satisfaction about his/her activity. Though the development of the project is not finished yet, still we planned for the user interface for the further development of the project.

10.2 Basic Interface

Following is the very simple interface of Billing System which we assumed for the project development.

10.2.1 Registration

The Registration view is the following:

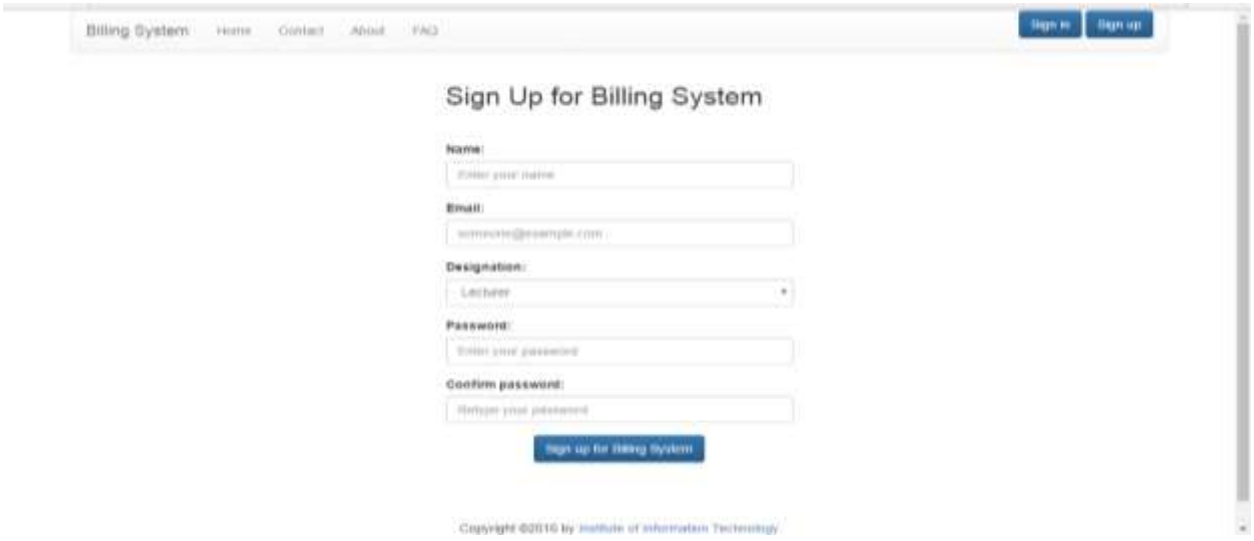


Figure 34: Registration Page

10.2.2 Sign in

The sign in view is the following:

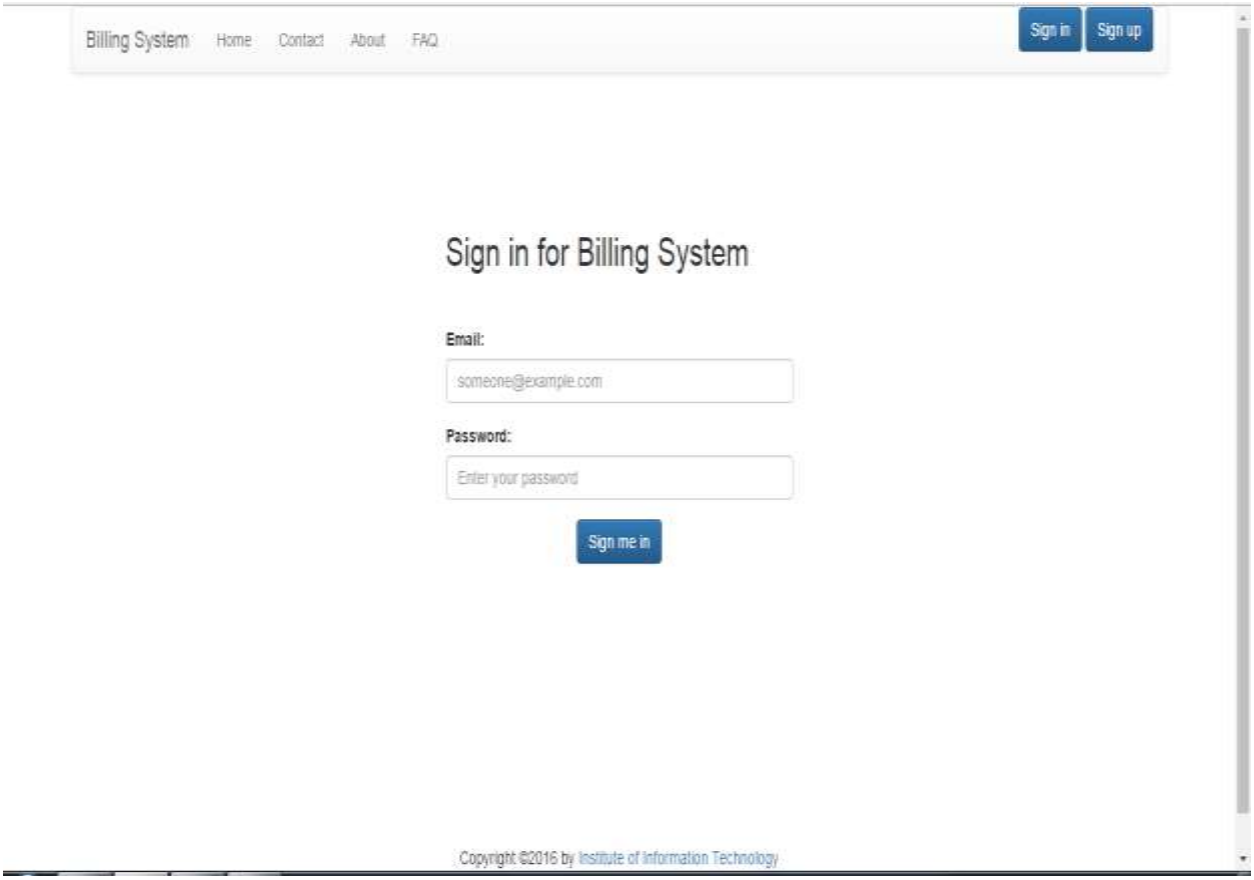


Figure 35: Sign In

10.2.4 Home (Normal Users)

The Home screen for normal users' is the following:

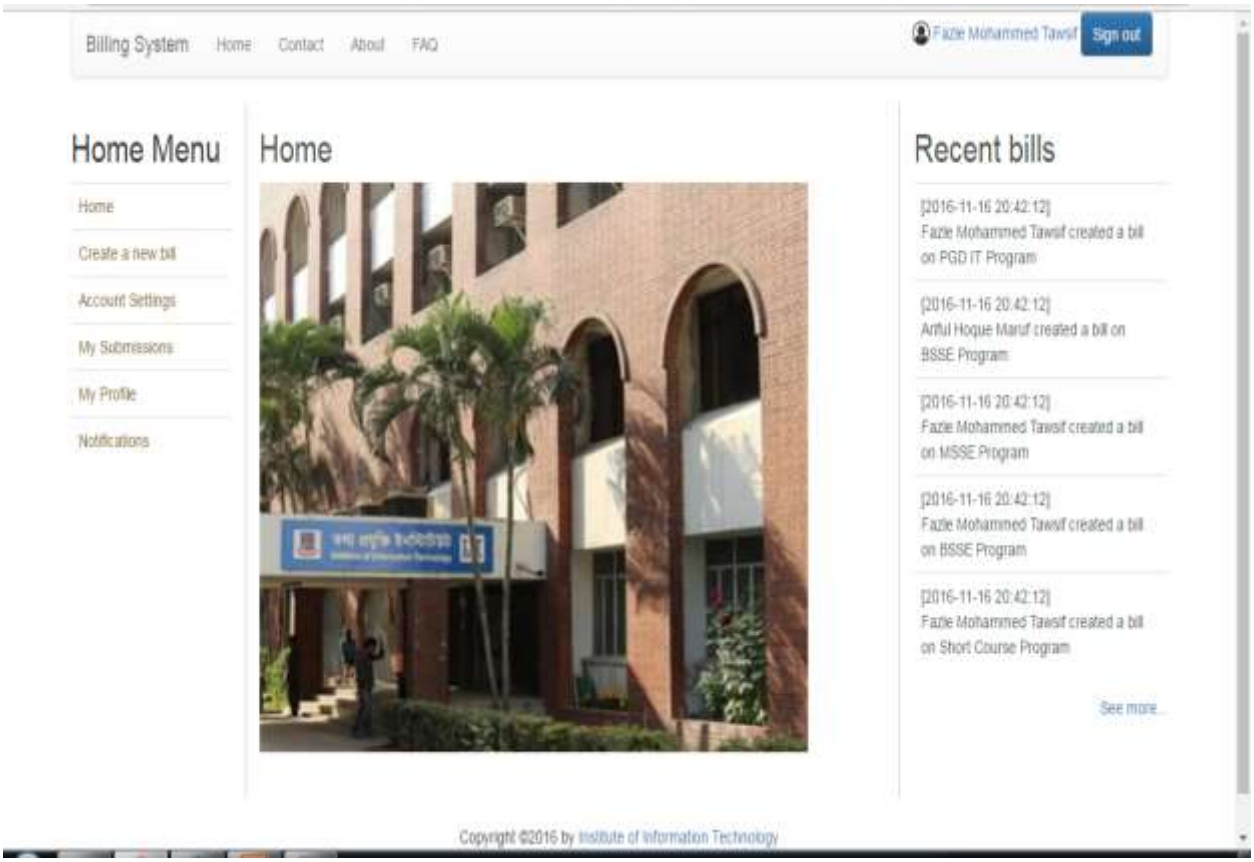


Figure 36: Home (for user) page

10.2.5 Home (Admin)

The Home for Admin view is the following:

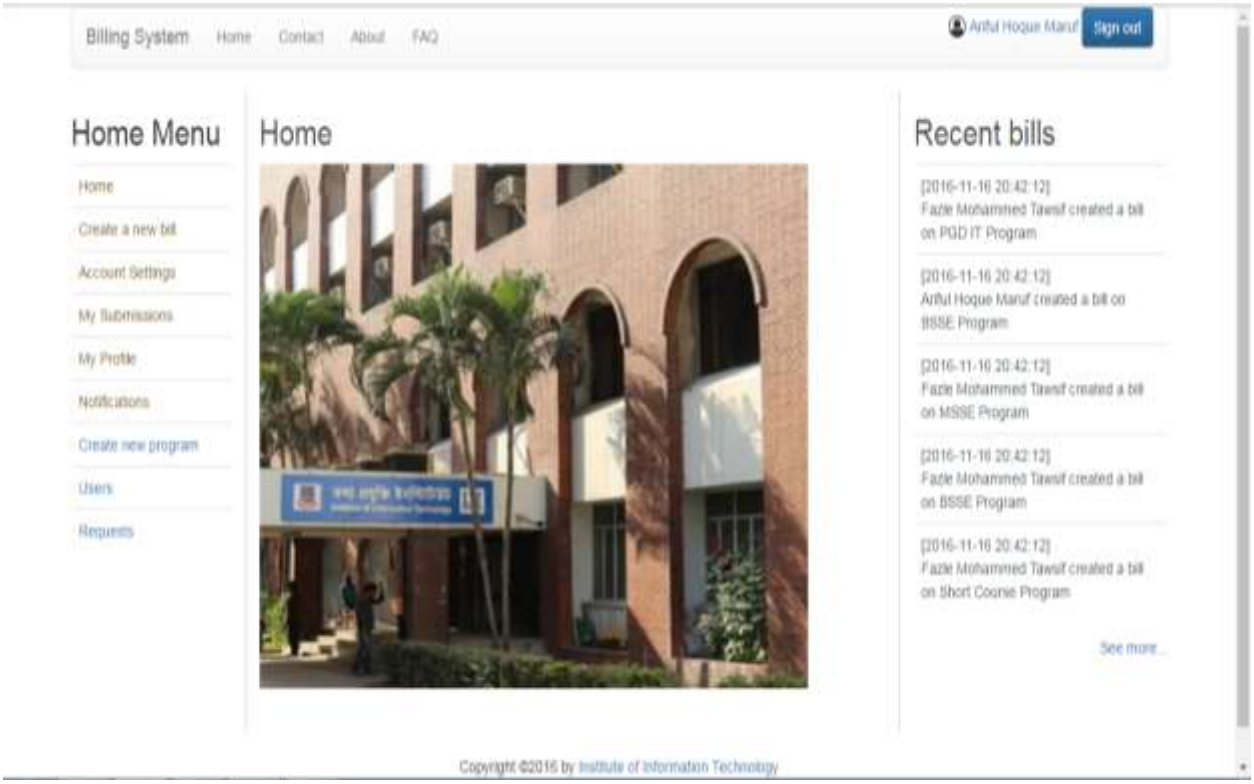


Figure 37: Home (for admin) page

10.2.6 Create New Bill

The Create New Bill view is the following:

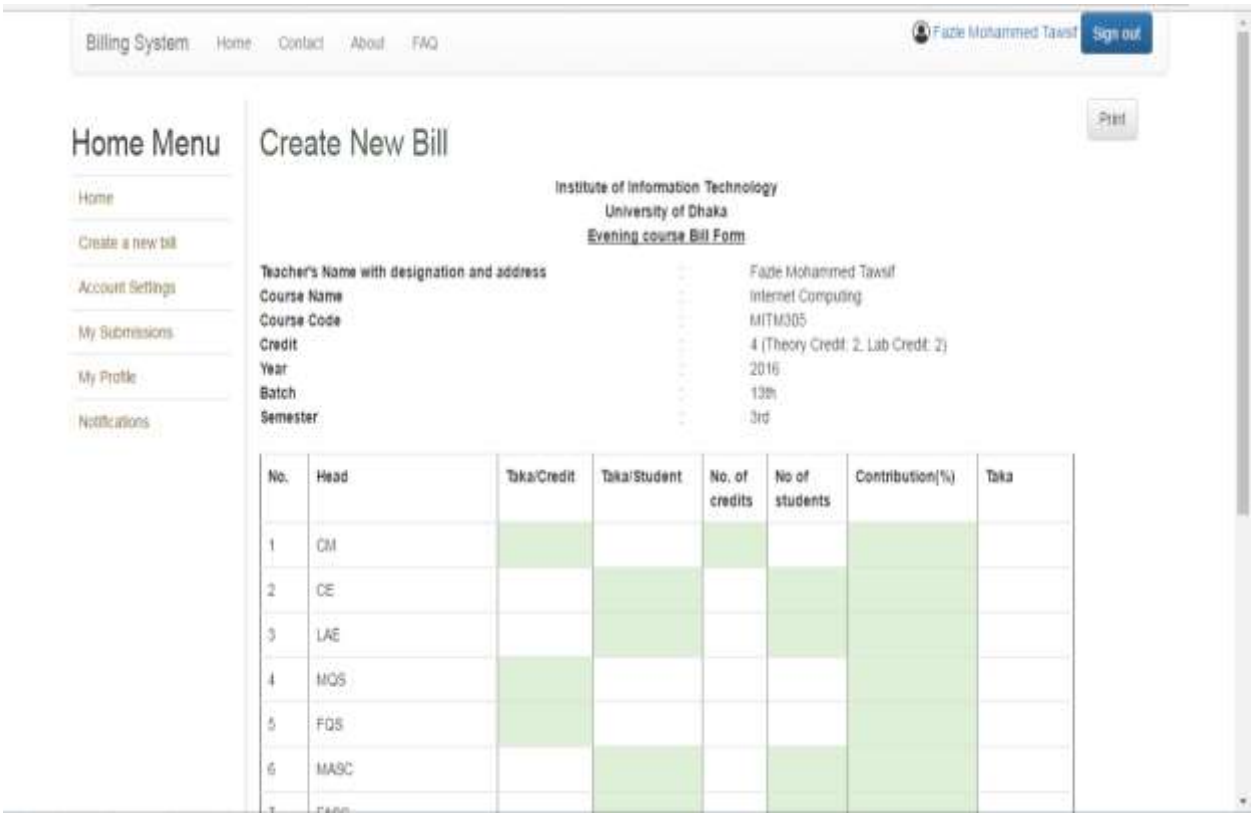


Figure 38: Create new bill page

10.2.7 Account Settings

The Account Settings view is the following:

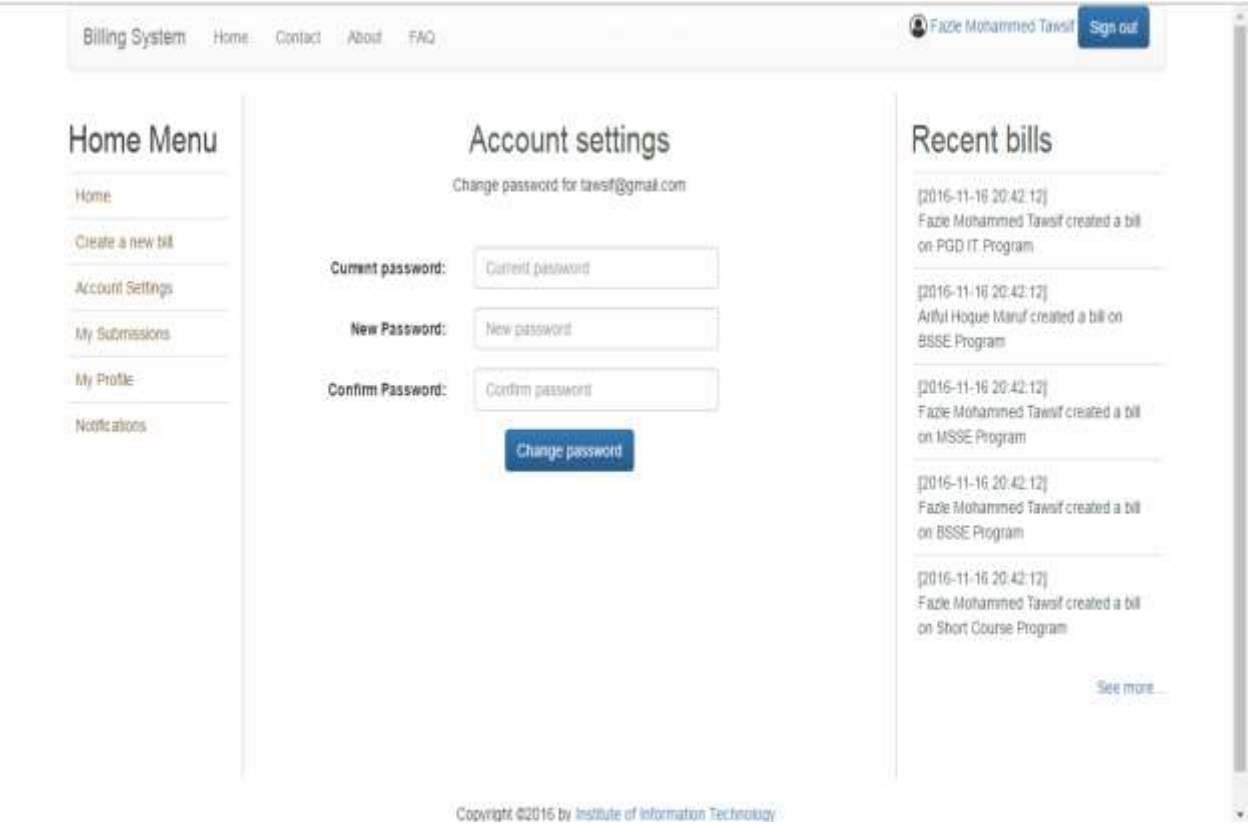


Figure 39: Account Settings page

10.2.8 My Submissions

The My Submissions view is the following:

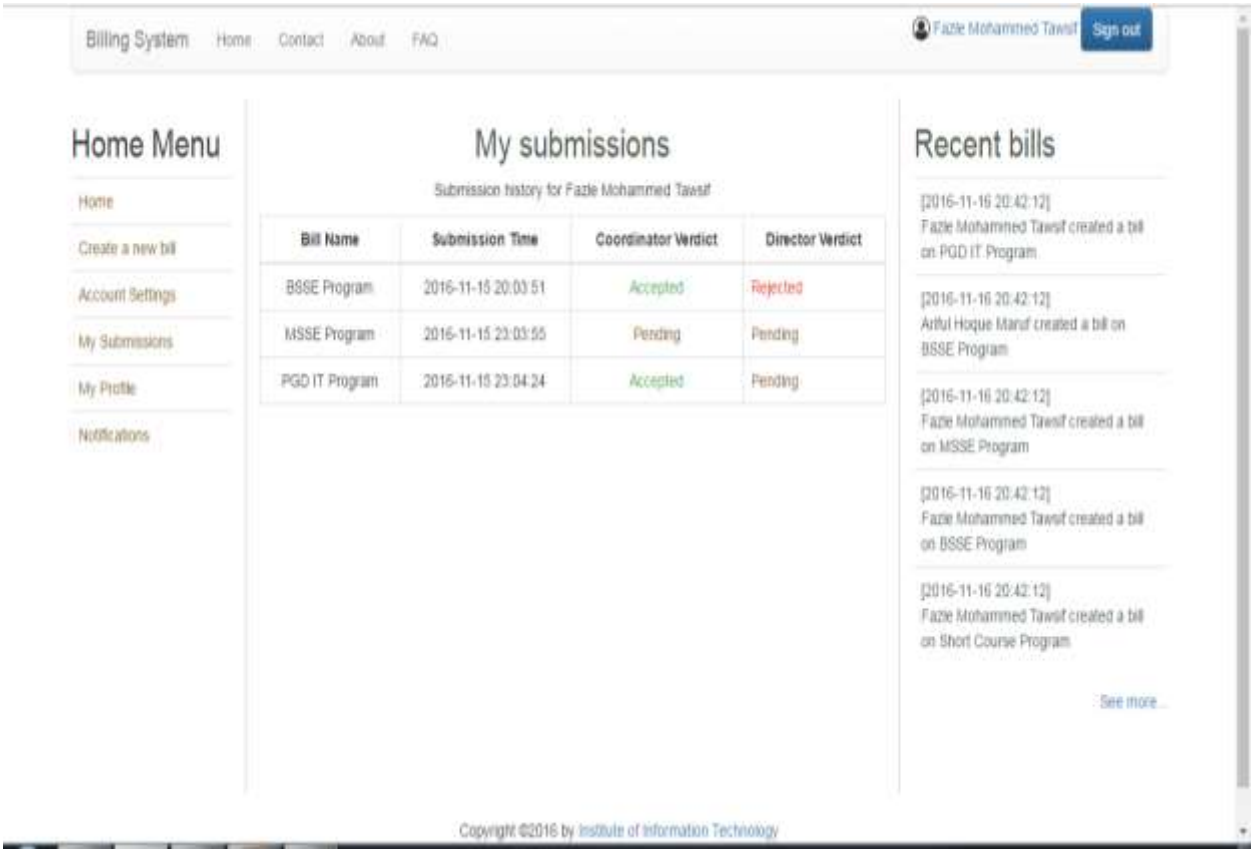


Figure 40: My submissions page

10.2.9 My Profile

The Contact information view is the following:

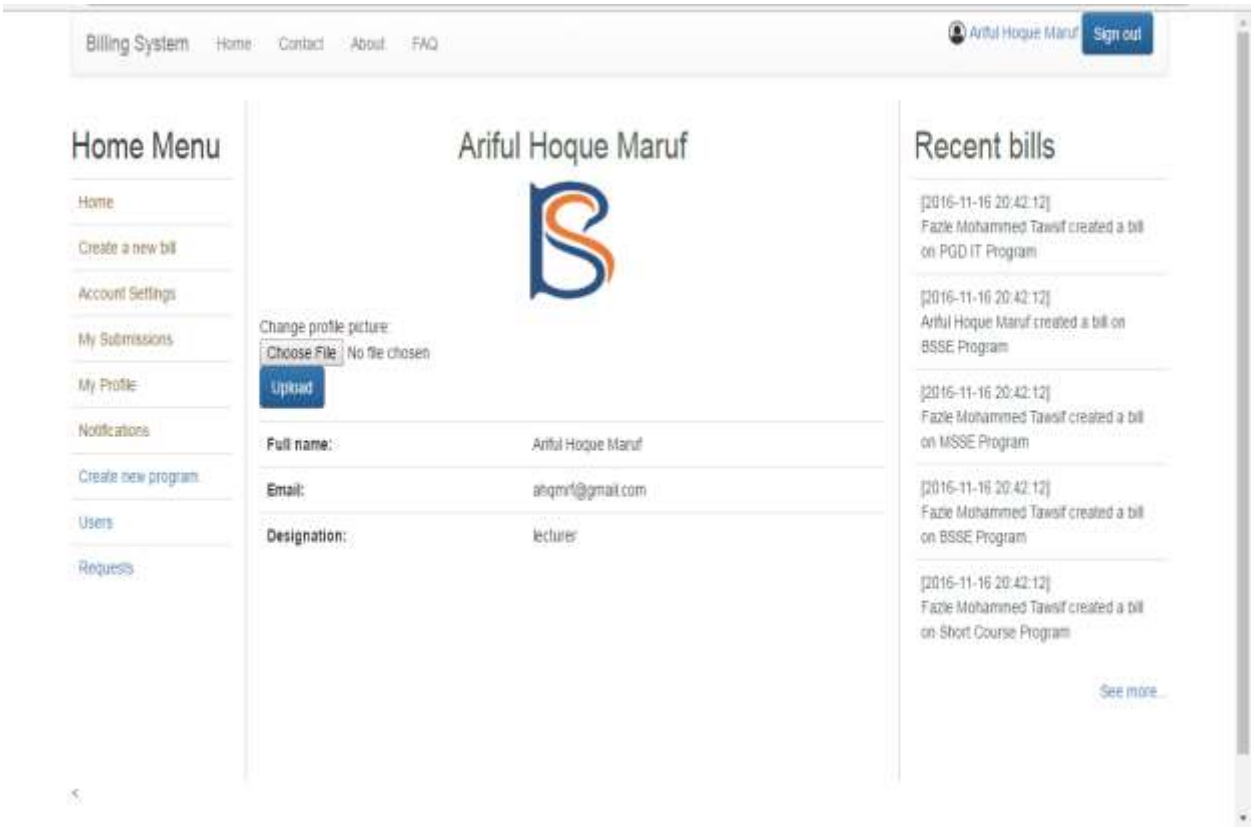


Figure 41: My Profile page

10.2.10 Notifications

The Notifications view is the following:

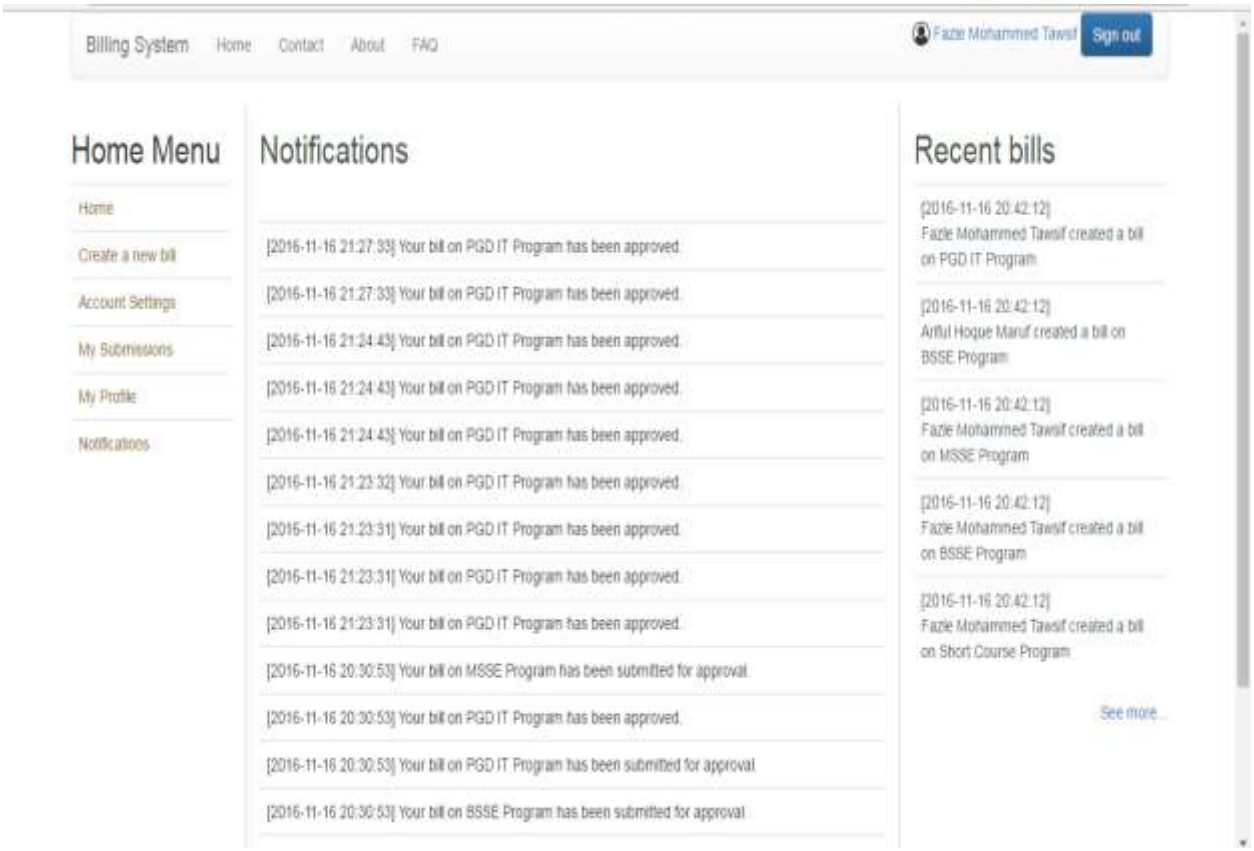


Figure 42: Notifications page

10.2.11 Create New Program

The Create New Program view is the following:

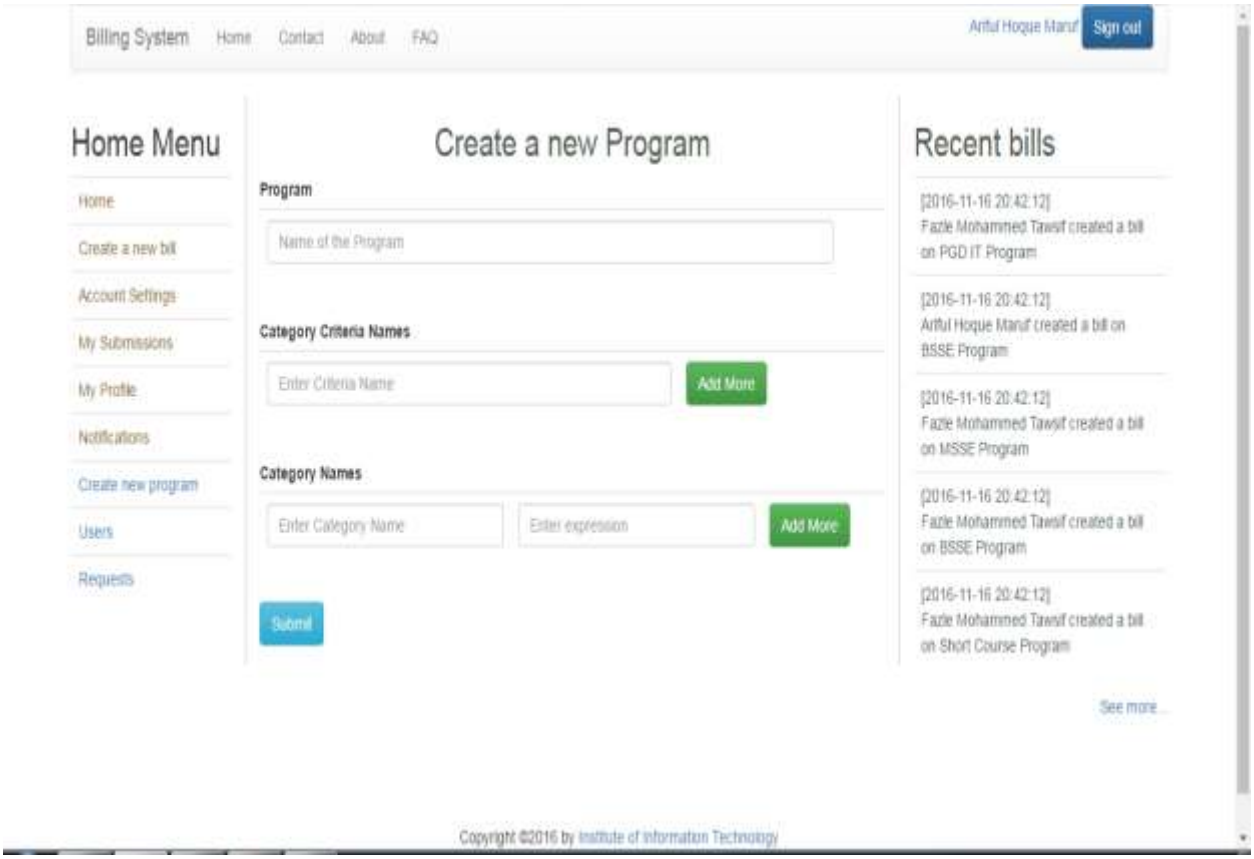


Figure 43: Create New Program page

10.2.12 FAQ

The FAQ view is the following:

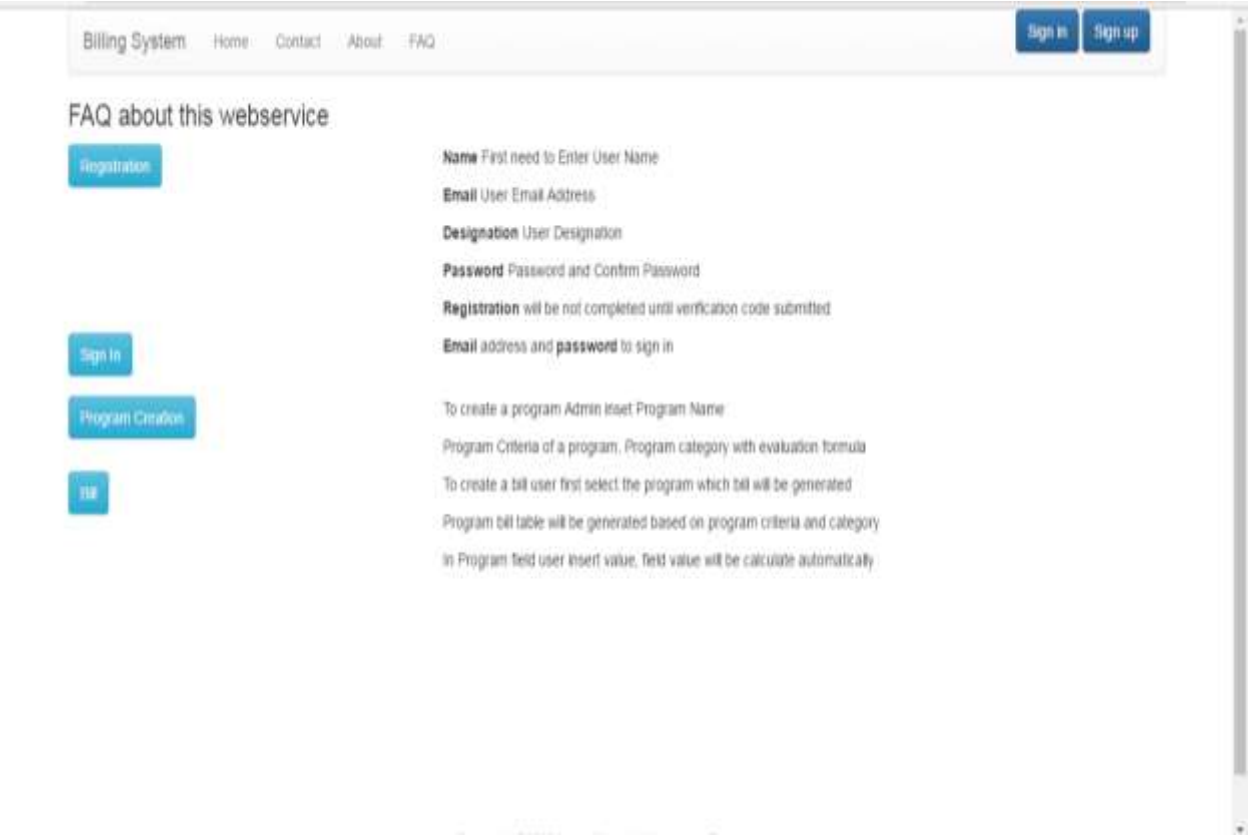


Figure 44: FAQ page

10.2.13 About

The About view is the following:

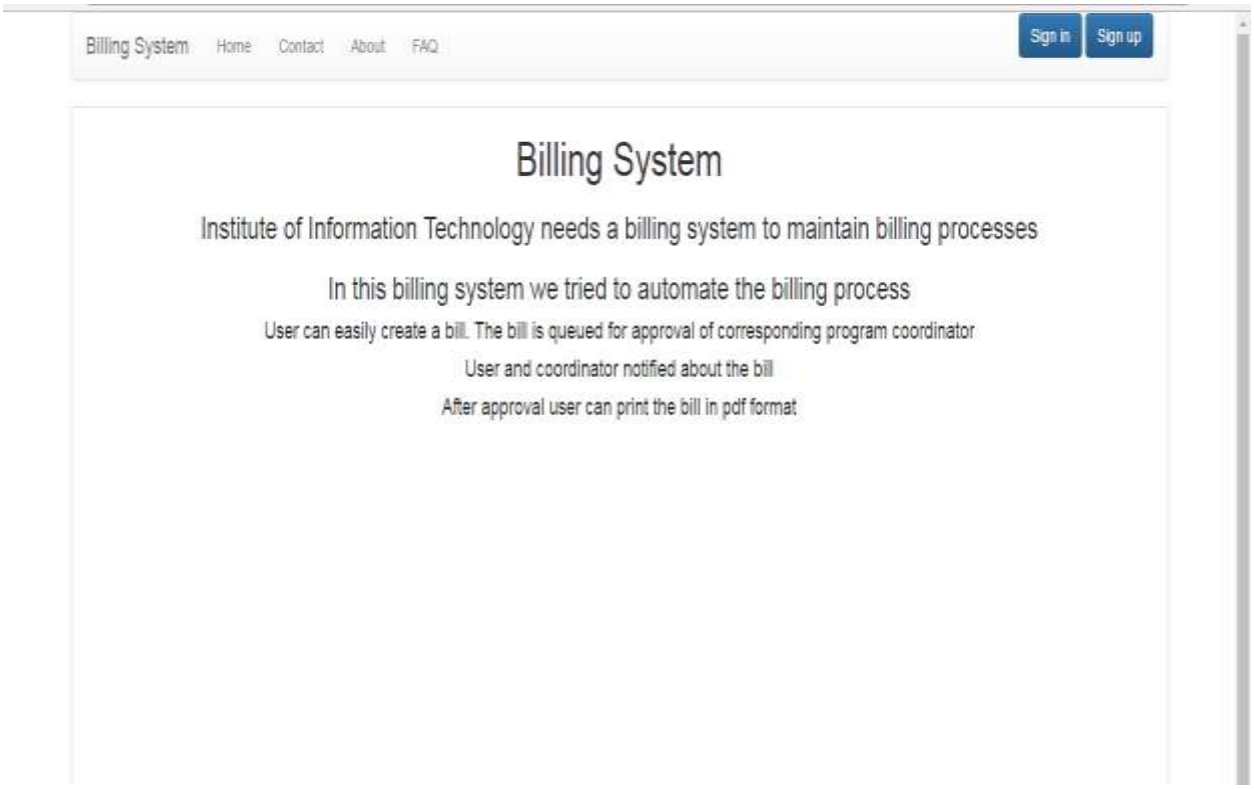


Figure 45: About page

10.2.14 Contact

The Contact information view is the following:

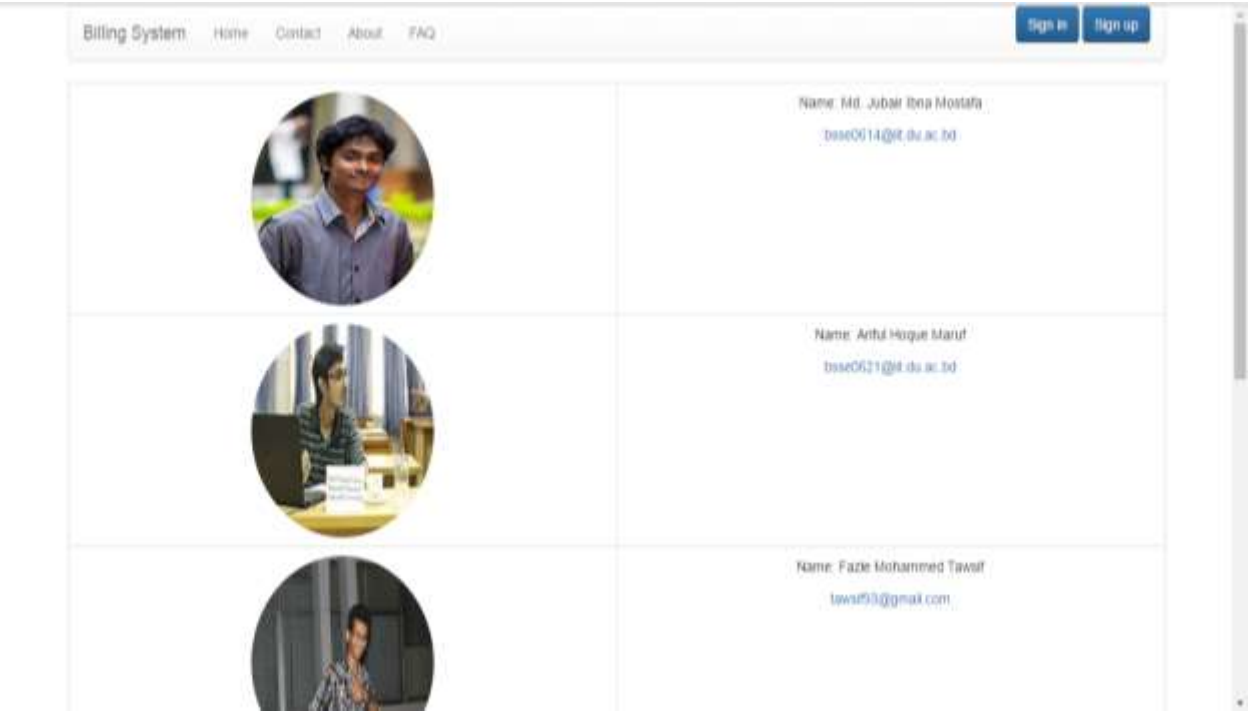


Figure 46: Contact page

These are the major user interface for the Billing System application.

10.3 Conclusion

The basic interface is only a preliminary planned interface. It may be changed after the development of the total project "Billing System". Hopefully this preliminary interface will help the users to get a better understand about the project.

Chapter 11: User Manual

We carefully created this user manual for the Billing system application of Institute of Information Technology.

11.1 General Information

The overall description of the system and the purpose of the development is mentioned here:

11.1.1 System Overview

The billing system is an application that provides with following major functionalities:

- Creates user profile
- Creates a bill
- Approves the bill
- Generates Final bill
- Proper authentication
- Saves the bill

The application saves data collected to database. The operational status is under construction. Billing System operates on web interface on mobile or computer.

11.1.2 Architecture of the System

The user's manual consists of five sections: General Information, System Summary, Getting Started, Using the System, and Reporting. General Information section explains in general terms the system and the purpose for which it is intended. System Summary section provides a general overview of the system. The summary outlines the uses of the system's hardware and software requirements, system's configuration, and user access levels system's behavior in case of any contingencies. Getting Started section explains how to get Billing System and install it on the device. The section

presents briefly system menu. Using The System section provides a detailed description of system functions. Reporting section describes in what way information collected by the application are presented and how to access the information.

11.2 System Summary

System Summary section provides a general overview of the system. The summary outlines the uses of the system's hardware and software requirements, system's configuration, user access levels and system's behavior in case of any contingencies.

11.2.1 System Configuration

Billing System operates on any devices with web interface. It is compatible with Google Chrome (4.0), Mozilla Firefox (3.5), Internet Explorer (6.0), Opera (10.1), Safari (3.1). The application requires connection to Internet in order to save data to database. Data saved in database can be seen using any major Internet browser. The Billing System doesn't need any further installation and configurations.

11.2.2 User Access Level

Only the faculty member of IIT and the people relevant with the billing system of IIT have the access in this web application. In addition, they must have to be registered and currently employed under Dhaka University. They can save data in the database.

11.2.3 Contingencies

In case there is no Internet connection available data cannot be saved in internal memory of the operating device and the system database.

11.3 Getting Started

Getting Started section explains how to get Billing System and use it on the device. The section presents briefly system menu.

11.3.1 Installation and Logging In

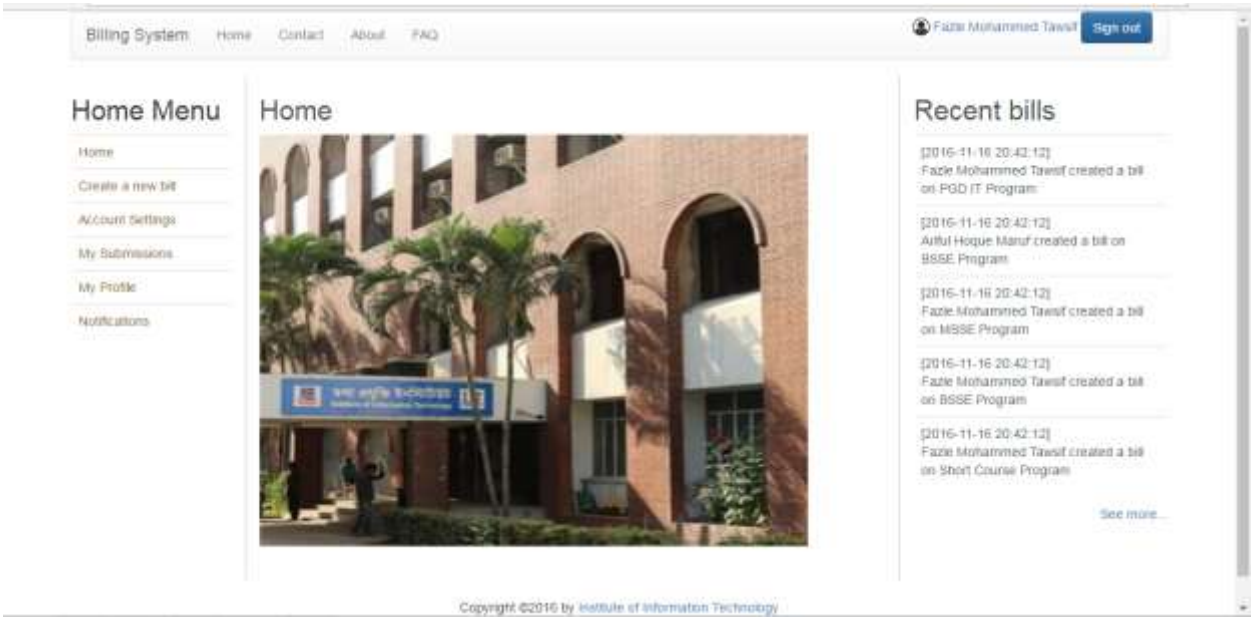
As it is a web application, it doesn't need any installation. User just have to hit the corresponding address. User have to register of the site providing name, email and password. For login user has to use email and password of the registration.

11.3.2 System Menu

Billing System has five main tabs. These are Billing System, Home, Contact, FAQ. Home tab also contains some other page links like create new bill, account settings, my submissions, my profile, notifications. The Billing System tab displays the home screen if the user is logged in. Otherwise it displays the index screen containing sign up/ sign in page.

11.3.2.1 Billing System

The Billing System tab display has two different views for registered and non-



registered users. For registered users, the view is following:

Figure 47: Home page

The Home page contains the basic options of the website. Its displays the sign out button, username and the recent activities of the user.

For Admin, the home page has three more options. The view is following:

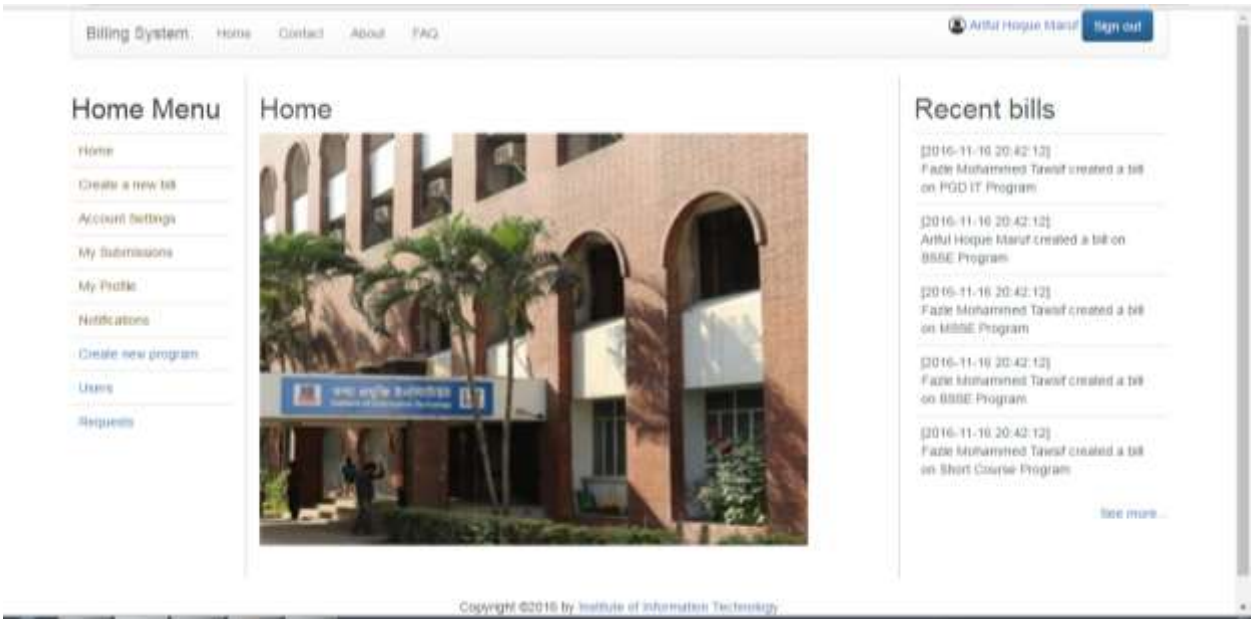


Figure 48: Home for admin page

For non-registered users, the view is following:

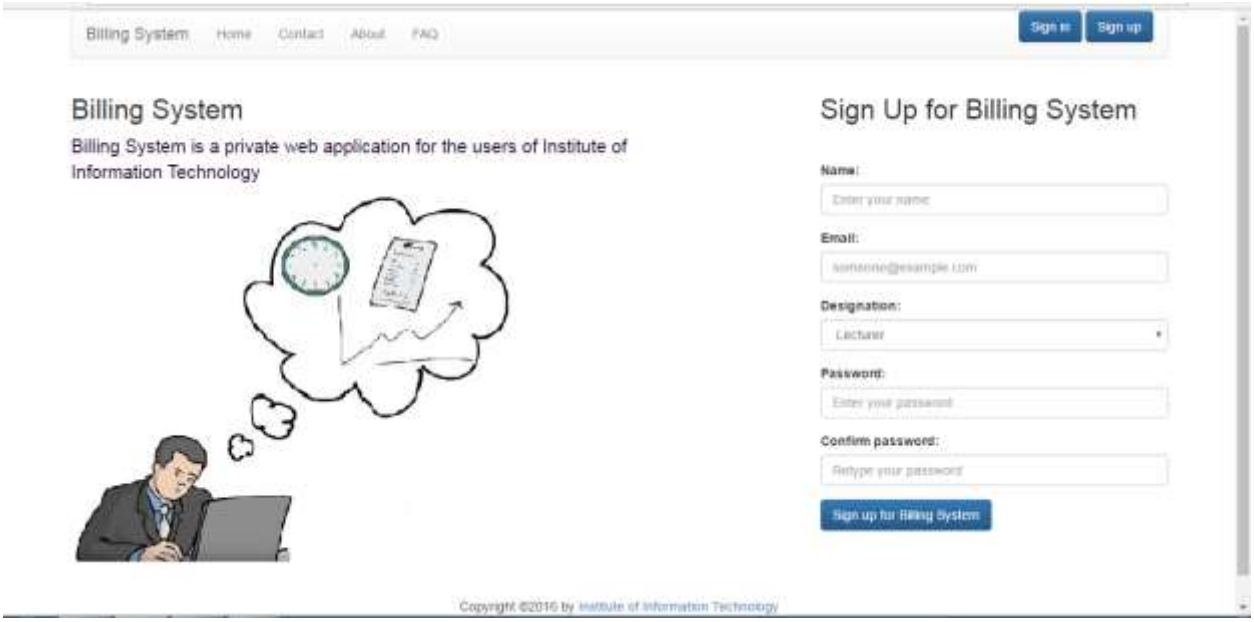


Figure 49: Sign up page

This page has five input fields for taking user information and registration. These are user name, email, designation of the user, password and confirm password fields. There are some buttons like sign up, sign in and sign up for Billing System.

After completing the sign up process once, the user can sign in by using the email and password. The sign in page is the following:

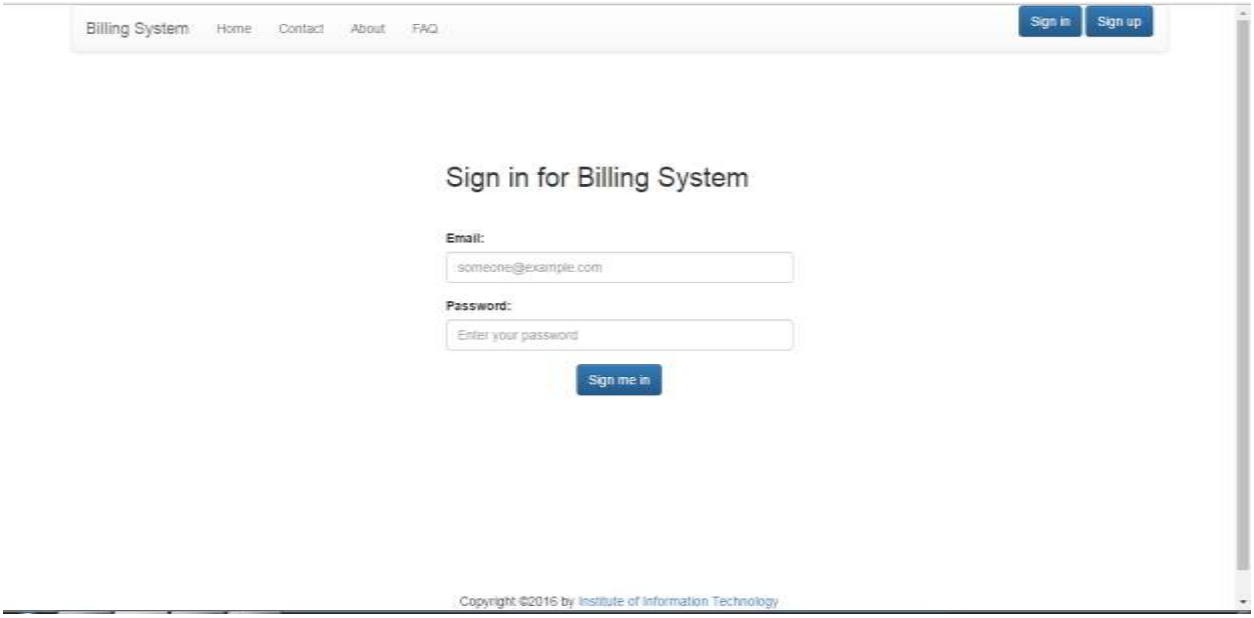


Figure 50: Sign in page

11.3.2.2 Contact

The contact page shows the contact information of the developers of the site.

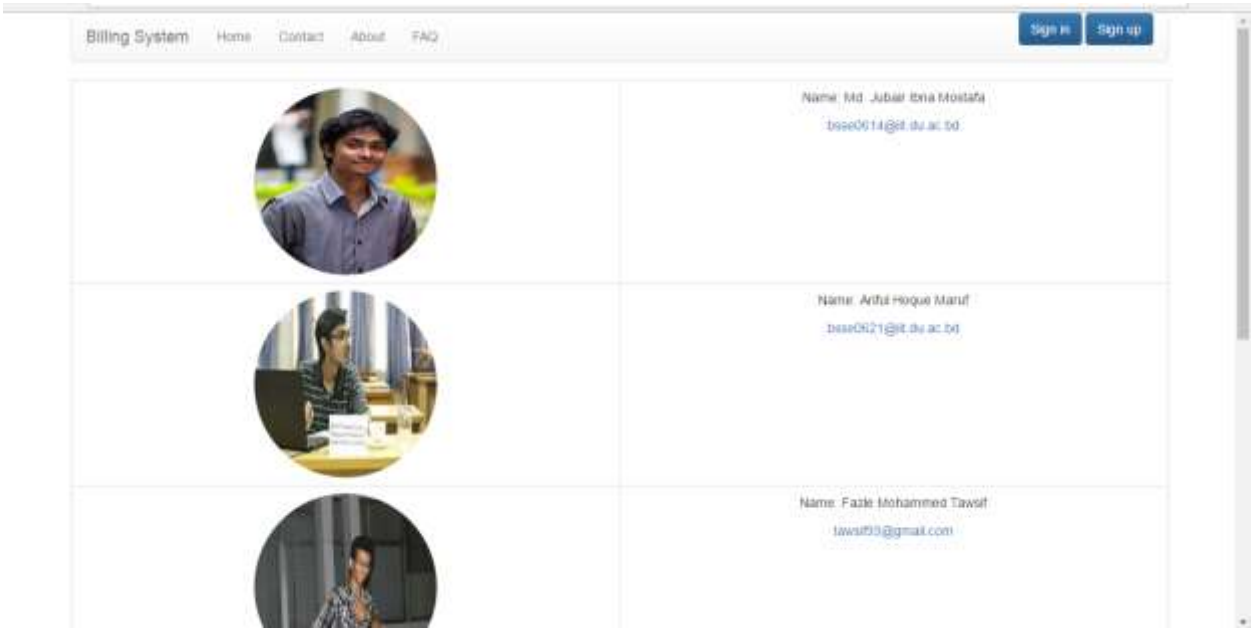


Figure 51: Contact page

11.3.2.3 About

This page contains a short description about the application.

Figure 52: About page

11.3.2.4 FAQ

This is for the help of the user. If the user has any question about the application, the user can go the link and write the question and submit it.

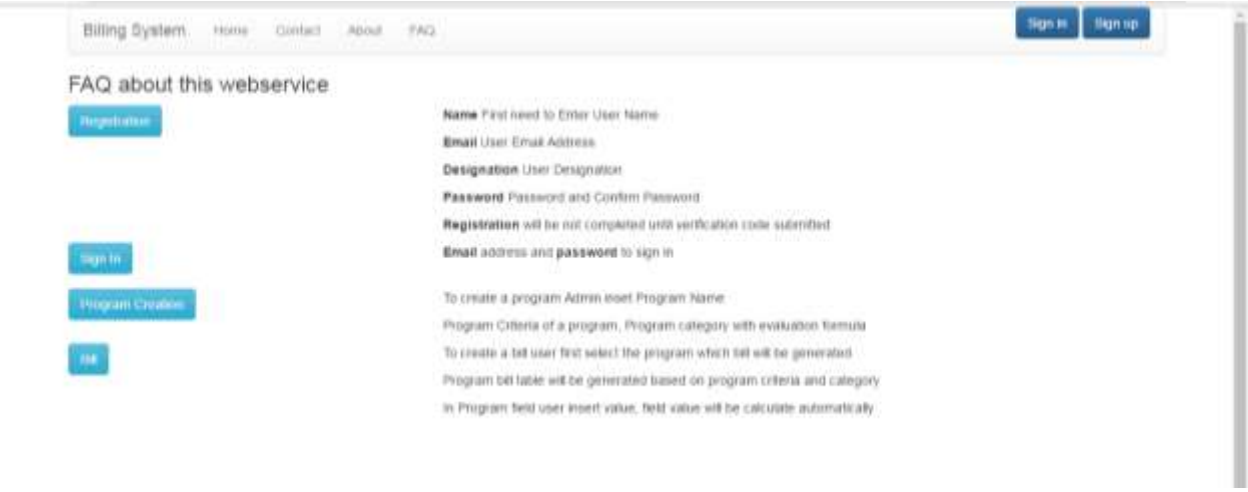


Figure 53: FAQ page

11.3.2.5 Create a New Bill

This page is for creating new bill. The registered users can visit this page and create new bill filling up required input fields.

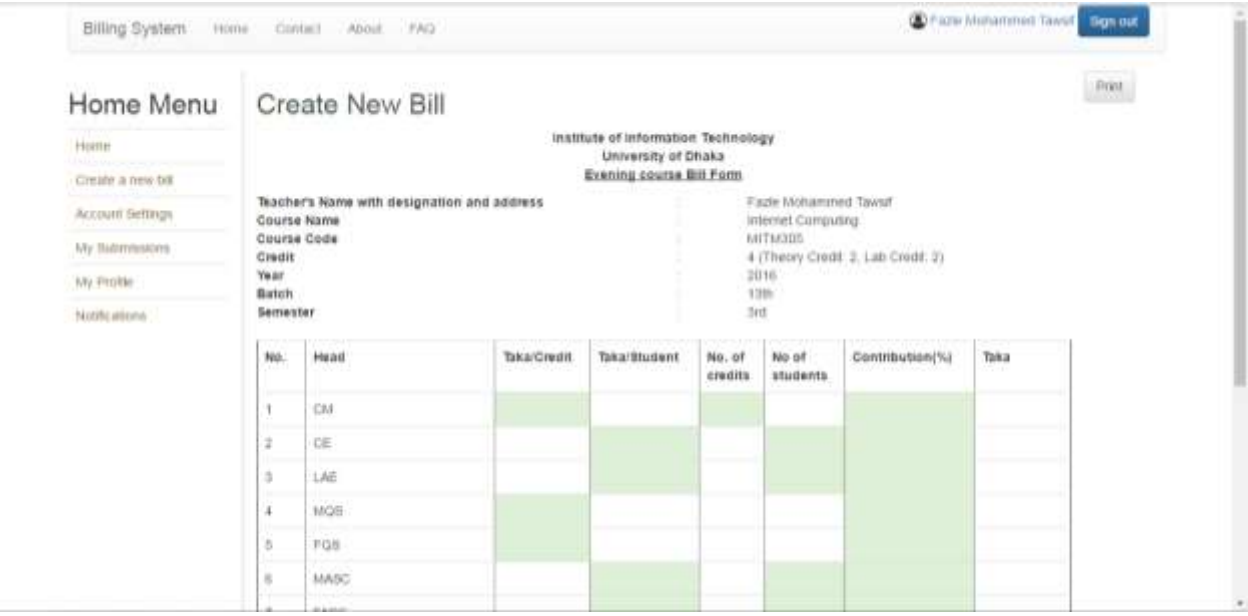


Figure 54: Create a new bill page

In this page, there is a print button at the corner. This is for printing the bill after completing. There are several input fields in this page and for different categories, different required input fields are needed. Only valid input fields are active for respected category. This page also contains the bill makes information. After completing the input, an automatic bill will be created.

11.3.2.6 Account settings

The account setting page is for changing the password of the current user. To change the previous password, the user has to provide the previous password and then enter the new password and reenter the new password again. And then click on the change password button.

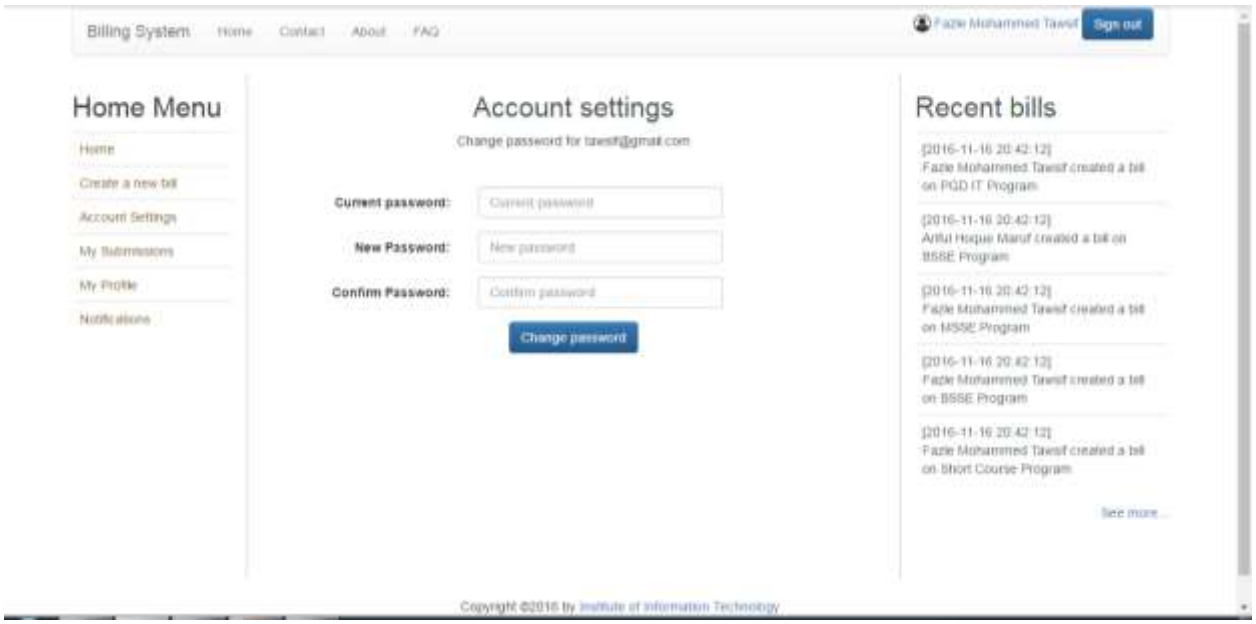


Figure 55: Account Setting page

11.3.2.7 My Submissions

This page contains the list of the submitted bill of the logged in user. It also contains the submission details like submission date and the approval status of the submissions.

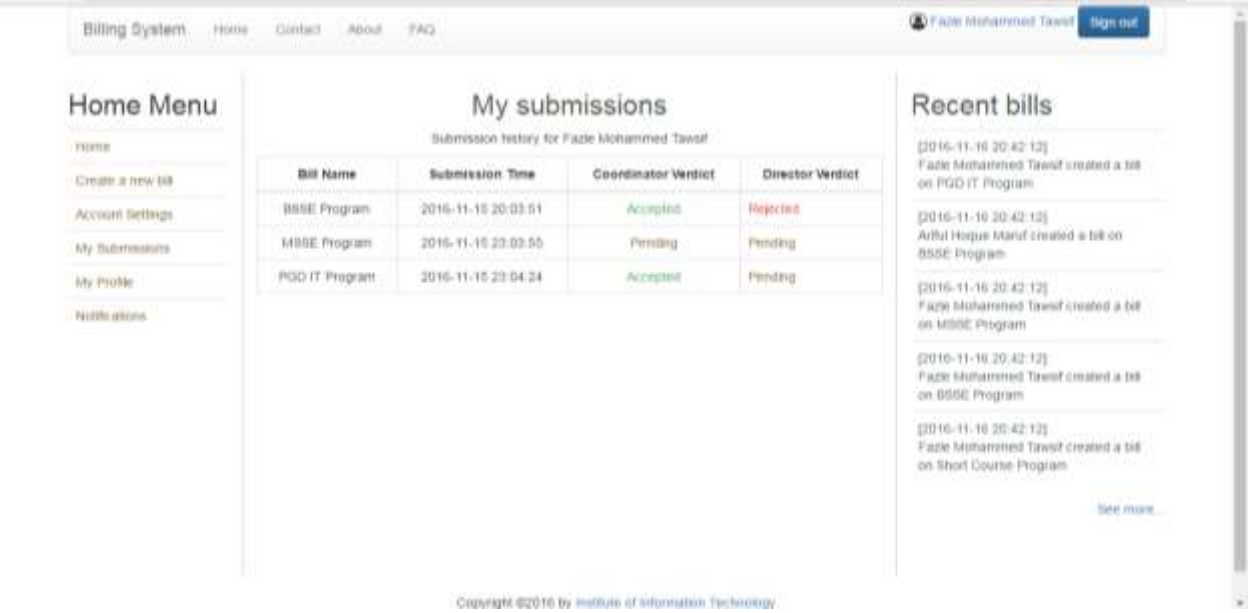


Figure 56: My submission page

11.3.2.8 My Profile

My profile page shows the current users' information. There is an upload button incase the user wants to upload any picture in his/her profile. It also displays the users' name, email and designation and recent bills.

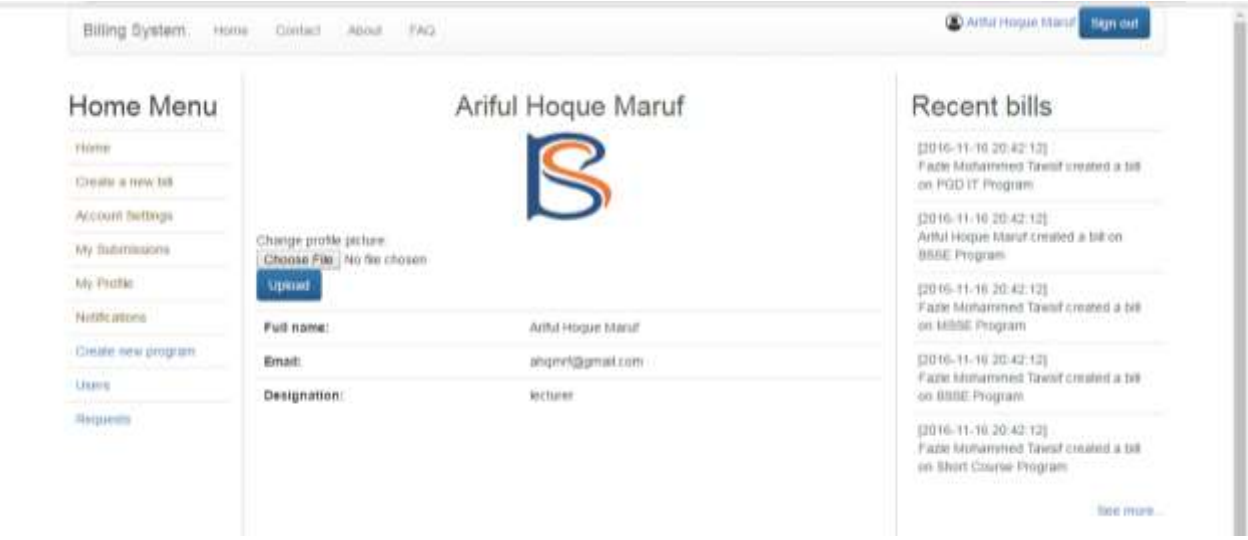


Figure 57: My profile page

11.3.2.9 Notifications

The notification page shows the approval and rejection notifications for the bill requests. Approved requests are displayed in blue color and rejected requests are displayed in red.

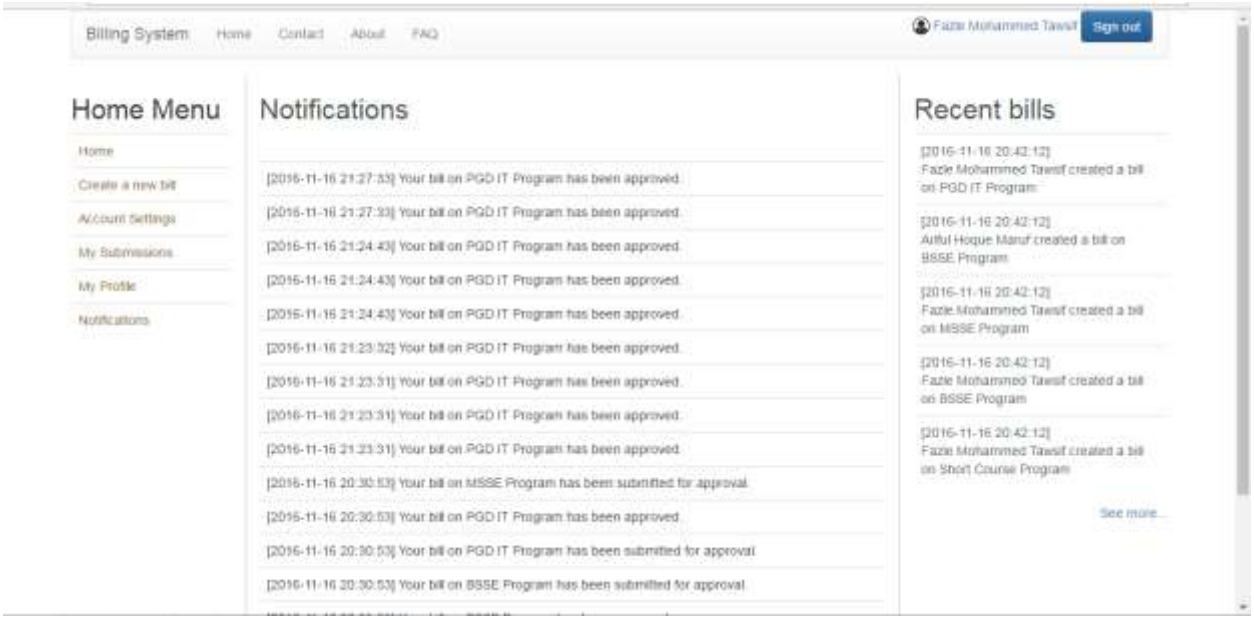


Figure 58: Notification page

11.3.2.10 Create a New Program

This page is only available for admin. Only admin can create new program for billing system. Normal users can request for billing programs but cannot create a program without admin help. This page contains some input fields. These are: Program name, category criteria name, category name. The admin can add more category criteria by clicking on the Add More Button. It will create new input field for new category criteria. The admin also can create new category by Add more button under category field. After finishing the input, the admin has to click on the submit button for creating new program and a new program will be added on the bill creation page.

In the category field, there is a field for category expression. This field defines the relation between the category criteria. The admin has to define the relation using mathematical expressions like (1*2). The 1 and 2 is not actual value. These are the number of the category criteria column in the create new Bill page.

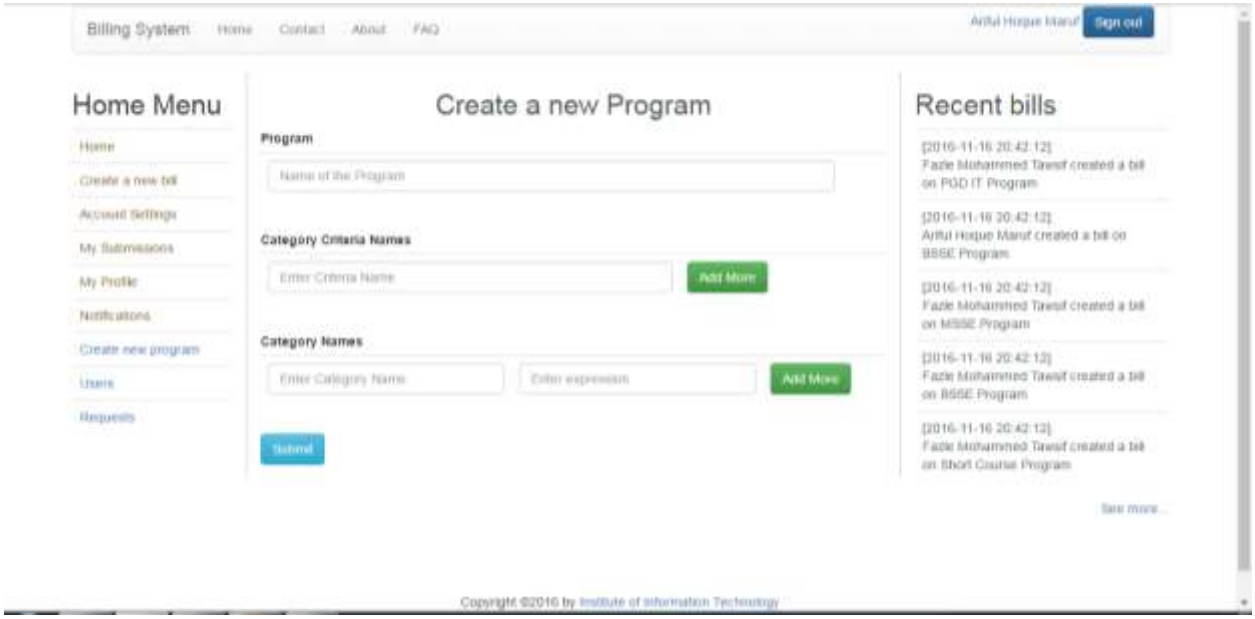


Figure 59: Create New Program page

11.3.2.11 Users

This page shows the list of the users and the search option for the admin as he/she can change any user into coordinator.

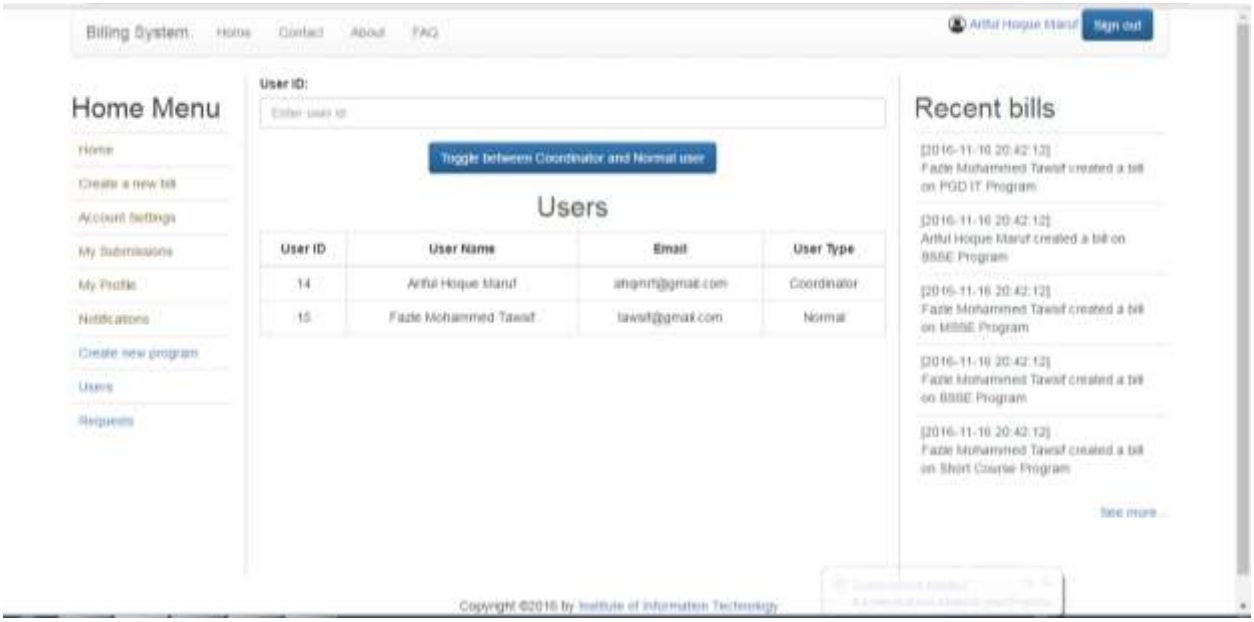


Figure 60: Users page

11.3.2.12 Requests

This page displays the list of the requests for admin and a search option by request id. Admin can view the requests, accept and reject them. There is two buttons for these purposes.

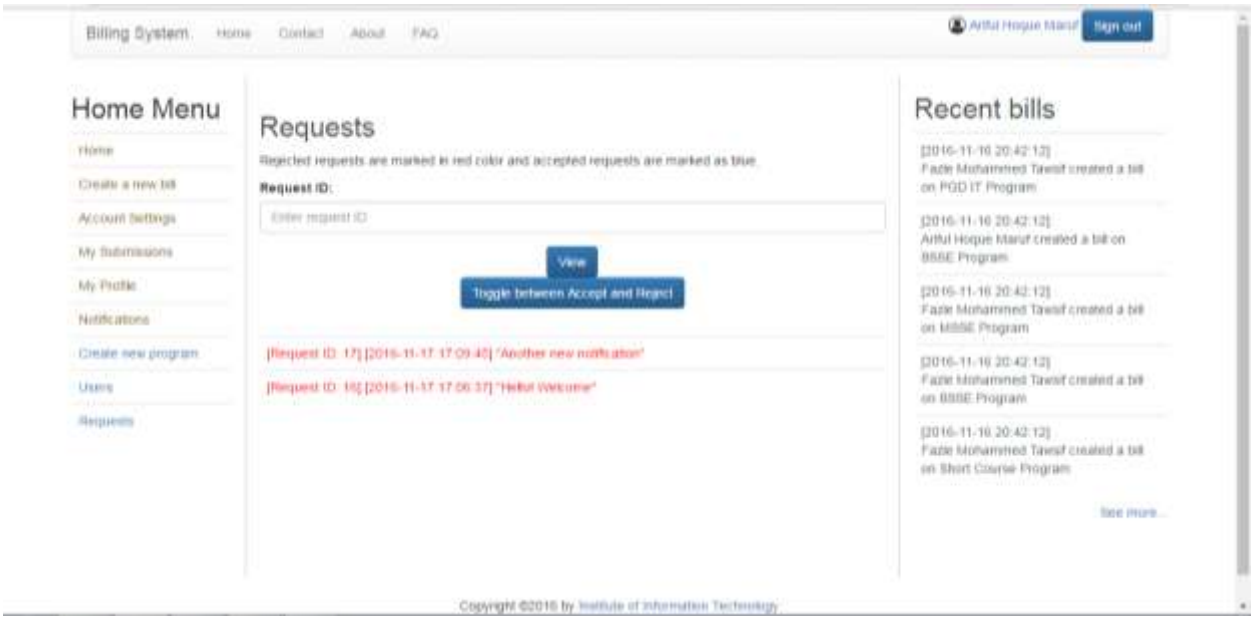


Figure 61: Requests page

11.4 Conclusion

The user manual is for the assistance of the user. This manual is very helpful for the user if anyone wants to use is manually.

Chapter 12: Conclusion

A software project means a lot of experience. In this section we summarize the experience gained by project team during analysis and learning phase of "Billing System".

12.1 Obstacles

There were several obstacles we faced when started the project. These are:

- Working with web application and server site development are completely a new experience for us. Normally we work with different OO languages, DBMS, mark up languages etc.
- We learned new technologies for getting the work done, still we are learning for the better manageability of our project.
- We are trying to adopt these things by video tutorials, text tutorials, and internet and learning materials given by the tools themselves. It's a matter of time, patience and hard work.

In essence the thing is that, a Billing System project with the social networking facilities is not a project of 2 or 3 months for two people!

12.2 Achievements

- Now we know much more about the server site application, how it works, the properties, objects and others.
- We know how a data base model is constructed, and how to manage all the data with the application.
- The main thing is that as a software engineer, skill and expertise to create a SRS document and an overall software product report is now better than before.

- Co-Operation between two group members.
- Develop communication skills
- Growing creative thinking and imagination capability.

12.3 Future Plan

- Introduce new features to the application.
- Launch the application for IIT.

12.4 Last Few Words

We are really very glad to submit the final report on Billing System. From this, the readers will get a clear and easy view of Billing System. This document can be used effectively to maintain software development cycle. It will be very easy to conduct the whole project using this document. Hopefully, this document can also help our junior BSSE batch students. We tried our best to remove all dependencies and make effective and fully designed document. We believe that reader will find it in order.

There were times that we almost lost hope but we recovered and will recover through constant concentration and hard work.

If any kind of suggestion, improvements, more efficient development idea please feel free to communicate with us.

Appendix

References:

- **Books**
  - Pressman, Roger S. Software Engineering: A Practitioner's Approach (7th Ed.)

- **URLs**
  - www.stackoverflow.com
  - www.stackexchange.com
  - www.w3schools.com