

SPL-2 SRS Report, 2020

Bus Automation System

Course: Software Project Lab II

Course No: SE 505

Submitted by

Md. Bashirul Islam, BSSE Roll No: 1022

BSSE Session: 2017-2018

Supervised by

Professor Dr. Zerina Begum

Designation: Professor

INDEX

1. INTRODUCTION

1.1 PURPOSE

1.2 INTENDED AUDIENCE

1.3 CONCLUSION

2. INCEPTION

2.1 INTRODUCTION

2.2 EXISTING STORY

2.3 INCEPTION PROCEDURE

2.3.1 IDENTIFY THE CLIENT OF OUR PROJECT

2.3.2 ICEBREAKING

2.3.3 IDENTIFYING THE STAKEHOLDERS

2.4 CONCLUSION

3. ELICITATION

3.1 INTRODUCTION

3.2 ELICITING REQUIREMENTS

3.2.1 COLLABORATIVE REQUIREMENTS

3.2.2 QUALITY FUNCTION DEPLOYMENT

3.2.3 USAGE SCENARIO

4. SCENARIO BASED MODELING

4.1 INTRODUCTION

4.2 DEFINITION OF USE CASE

4.3 USE CASE DIAGRAMS

4.4 ACTIVITY DIAGRAMS

5. DATA-BASED MODELLING

5.1 DataModellingConcept

5.1.1 Data Objects

5.1.1.1 Data Object Identification

5.1.1.2 Final Data Objects

5.2 DATAOBJECT RELATIONSHIPS

5.3 ENTITY RELATIONSHIP DIAGRAM

5.4 SCHEMA DIAGRAM

6. CLASS-BASED MODELING

6.1 CLASS BASED MODELING CONCEPT

6.2 Noun List for CMS

6.3 Verb List of CMS

6.4 GENERAL CLASSIFICATION

Table: 1 General Classification

6.5 SELECTION CRITERIA

Table: 2 Selection Criteria

6.6 ATTRIBUTE AND METHOD IDENTIFICATION

6.7 Analyzing Classes

6.8 CRC Card

6.9 Class Card

7. FLOW ORIENTED MODEL

8. BEHAVIORAL MODELING

Chapter One

INTRODUCTION

This chapter is a part of our software requirement specification for the project “Bus Automation System”. In this chapter we will focus on the intended audience for this project.

1.1 PURPOSE

This document briefly describes the Software Requirement Analysis of our mobile application Bus Automation System. It contains the functional, non-functional and the supporting requirements and establishes a requirement’s baseline for the development of the system. The requirements contained in the SRS are independent, uniquely numbered and organized by topics. The SRS serves as an official means of communicating user requirements to the developer and provides a common reference point for both the developer team and the stakeholder community. The SRS will evolve over time as users and developers work together to validate, clarify and expand its contents.

1.2 INTENDED AUDIENCE

Our intended audiences are:

1. Students
2. Teachers
3. Officers of DU

This SRS will be used to verify whether the developer team has created a product that is acceptable to the desired users. The developers will use this SRS to plan milestones and a delivery date, and ensure that they are 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 a 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.

1.3 CONCLUSION

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

Chapter Two

INCEPTION OF Bus Automation System

In this chapter, the Inception part of the SRS will be discussed briefly.

2.1 INTRODUCTION

This is an android application which provides students the Dhaka University Bus related things such as current location, schedule, committee members list, information etc. Students can locate varsity buses during any up trips (Directed to varsity) and see any kind of information about varsity bus as well as any updates.

2.2 EXISTING STORY

Goal Developing a DU BUS android Application for DU students to know schedule updates and real time location of any varsity bus.

Features:

- Providing bus information control to admin
- Bus scheduling list and committee list
- Instant notification from admin to user
- Providing user real time location of any varsity bus
- Search option for finding bus on a specific route
- Providing communication between customer to admin

Out of scope:

- Admin to admin communication is out of scope
- Without internet connection location can't be detected

2.3.1 ICEBREAKING

Icebreaking refers to diminishing the communication barrier between you and the other person. It is a crucial part since it decides the acceptance of our proposal. We started this phase by talking with them in context free languages. Their behavior, response to our question or willing to accept the new course management system solely depends on this phase.

2.3.3 IDENTIFYING THE STAKEHOLDERS

IDENTIFYING THE STAKEHOLDERS of Bus Automation System refers to any person or group who will be affected directly or indirectly by the system. Stakeholders include end-users who interact with the system and everyone else in an organization who may be affected by its installation.

The Bus Automation System has a limited number of stakeholders. Identification of the stakeholders was done from the information provided by the teachers and the students. It turned out that we don't have any stakeholders other than the clients.

2.4 CONCLUSION

The primary goal of this project is to model and design software for those people who are related to a course. For these reasons, the software will be designed in such a way that it will bring comfort to the client who will use it. The software will be simple and user friendly. Otherwise, it will not be appreciated by the clients. The software will be designed in such a way that it takes very little time to manage. To make this software project successful, collaboration with stakeholders is the main priority. To sum up, what they want, how the software will work and how it can be more efficient than the earlier times is the main concern in this phase.

Chapter 3

ELICITATION OF Bus Automation System

After discussing the Inception phase, we need to focus on the Elicitation phase. So this chapter specifies the Elicitation phase.

3.1 INTRODUCTION

Requirements Elicitation is a part of requirements engineering that is the practice of gathering requirements from the stakeholders. We have faced many difficulties, like understanding the problems, making questions for the stakeholders, limited communication with stakeholders due to shortage of time and volatility of the stakeholders. Though it is not easy to gather requirements within a very short time, we have surpassed these problems in an organized and systematic manner.

3.2 ELICITING REQUIREMENTS

We have seen the 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 of Work Products

3.2.1 COLLABORATIVE REQUIREMENTS GATHERING

We have met with many students in the Inception phase. Besides, we have met with 1 teacher. These meetings created an indecisive state for us to elicit the requirements. To solve this problem, we have met with some stakeholders (who are playing a vital role in the whole process) more than once to elicit the requirements.

3.2.2 QUALITY FUNCTION DEPLOYMENT

3.2.3 USAGE SCENARIO

Registration:

The students, teachers and the officers of Dhaka University are the users of this android app. To use this app, the user will have to register first. For registration, the user will have to provide user name, email

address, phone number and password. The password should contain at least 8 digits. The user can also add his profile picture.

The system will create a user ID for each user. All information will be stored in the database.

Sign in:

No user can sign in before registration. A user can sign in with his password and email address.

Welcome Screen:

After signing in, the user will see a welcome screen with a search icon. Here a user can write a bus route name, the system will show him the bus schedule.

Main Menu:

In the main menu, there will be a number of sections. The sections are user profile, bus schedule, bus location, notification, contributor, about and settings.

Profile:

A profile will be created for each user. It will contain user name, email address, phone number and the profile photo of the user. A user can edit his photo and phone number in the profile section.

Bus Schedule:

In this section, it will contain the schedule of each bus route. The schedule will be stored in the database. The user can search any bus route schedule from this section. When the user searches for a bus route, the system will give him a schedule for both up-trips and down-trips. The bus schedule will be dynamic. The summer-schedule will be shown during January- September and the winter-schedule will be shown during October-December.

Bus Location:

From this section, a user will get the current location of the bus in the map. It will give information about the bus on the route.

Notification:

In this section, the recent information about a bus route will be given. That information will also be sent to the email address of the user by the system.

Contributor:

This section will bear information about the contributors of this android app.

About

The about section will bear information about the whole system.

Chapter 4

SCENARIO BASED MODELING

This chapter describes the Scenario Based Model for the “Bus Automation System”

4.1 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 begins with the creation of scenarios in the form of Use Cases, activity diagrams and swim lane diagrams.

4.2 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 roles 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 the system. Actors either consume/ produce/ modify information.

Primary Actor

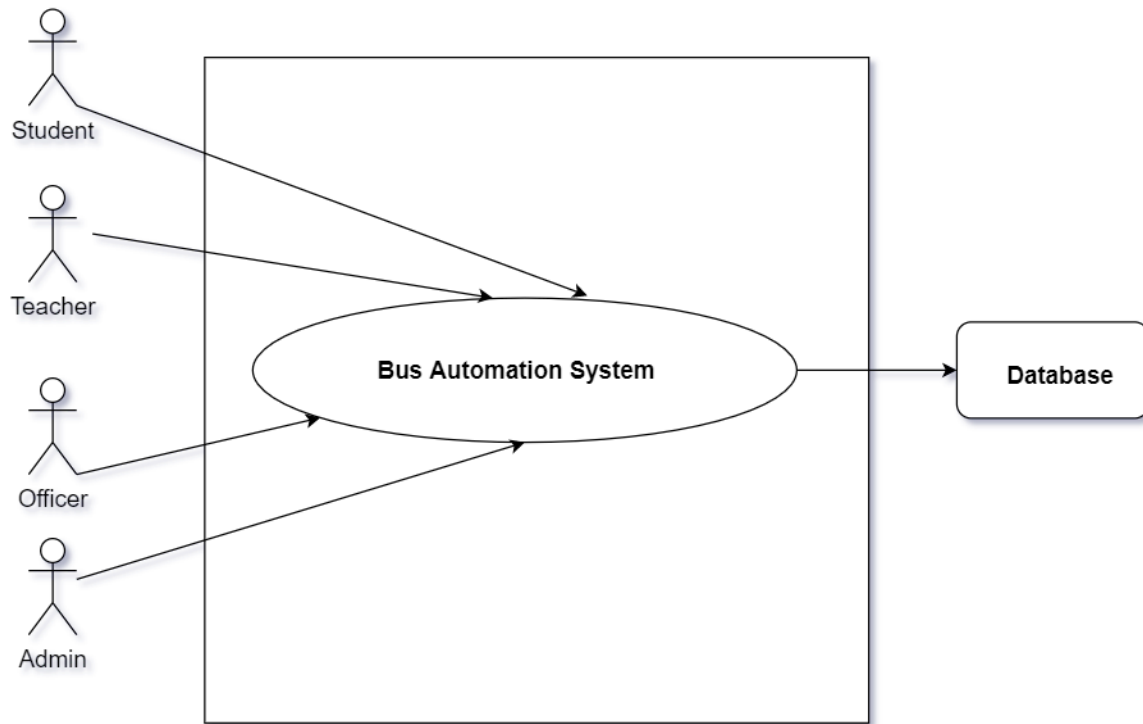
Primary actors interact directly to achieve required system function and derive the intended benefit from the system. They work directly and frequently with the software. The actors who do more than one job (consume/produce/ manipulate information) are our primary actors.

Secondary Actor

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

Use Case Diagram Level: 0

Name: Bus Automation System



Description: The use case “Bus Automation System” at level 0 shows the top-most view of the project and all the actors, namely:

Primary Actor:

1. Students
2. Teachers
3. Officers
4. Admin

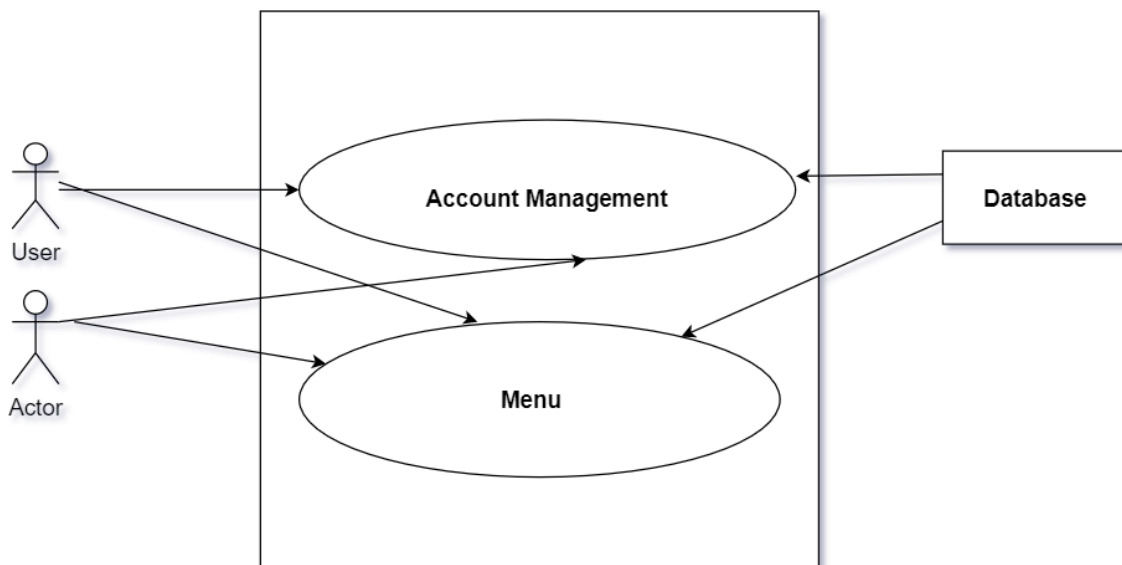
Secondary Actor:

5. Database

Use Case Diagram Level: 1

Name: Sub-system of Bus Automation System

ID No: 02



Description: Bus Automation System is divided into 2 main subsystems:

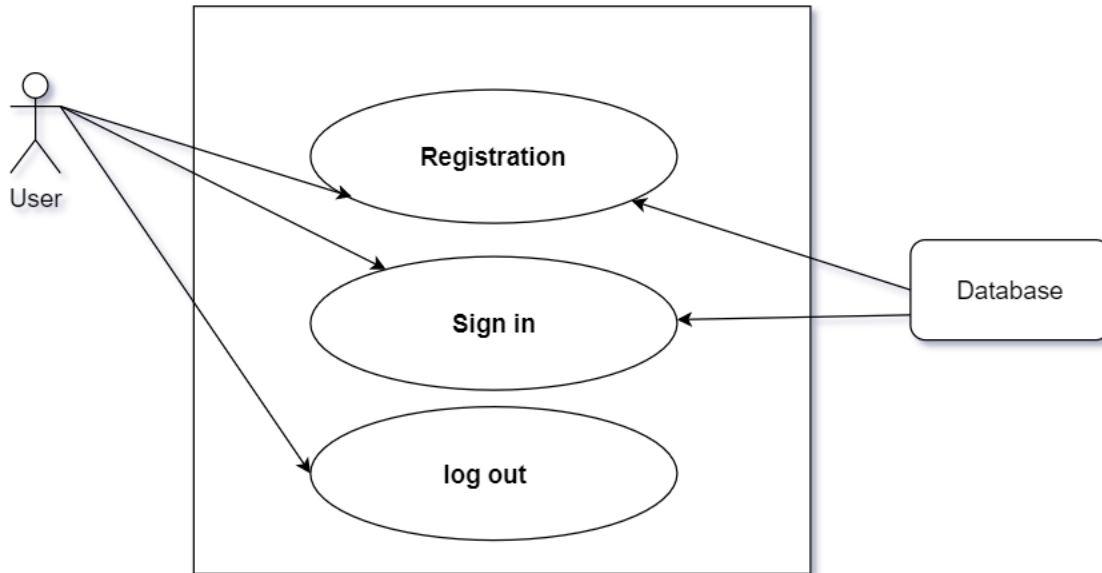
1. 1 **Account Management:** this section is used for registration and sign in procedure. This section is further divided in level 1.1.

1.2. **Menu:** this section shows various features of this android app.

Use Case Diagram Level: 1.1

Name: Account Management

ID No: 03



Description: Account Management System is divided into 3 main subsystems:

1. Registration: This section is used for registration.

User action: User tries to register with name, email, phone number and profile picture.

Reply: System verifies if the email number is in the database. If not, the user register successfully.

2. Sign in: User tries to sign in with email and password.

User action: User tries to sign in with email and password

Reply: System verifies if the email number and password matches in the database.

3. Log out:

User action: User tries to log out of the system.

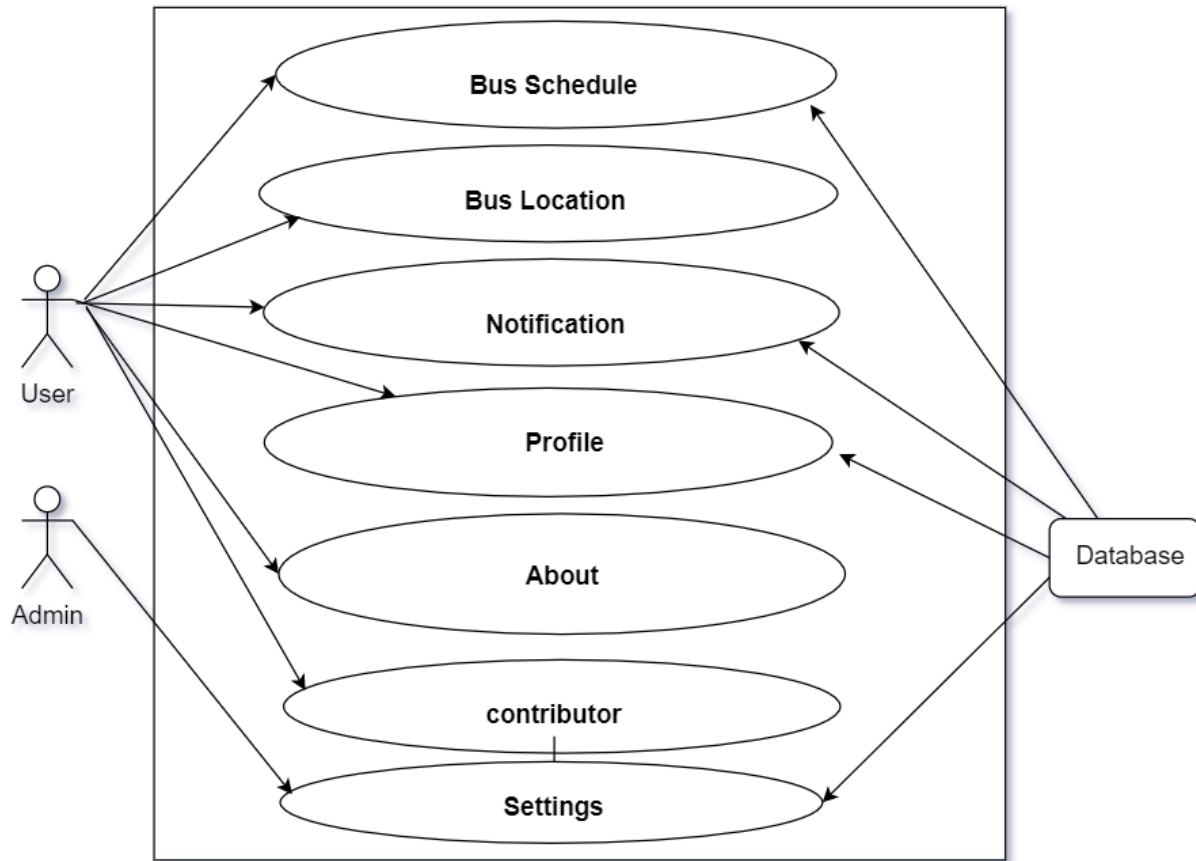
Reply: User will be logged out.

THE NEXT DIAGRAM IS SHOWN IN THE NEXT PAGE

Use Case Diagram Level: 1.2

Name: Menu

ID No: 04



Description: Menu System is divided into 7 main subsystems:

1. Bus Schedule
2. Bus Location
3. Notification
4. Profile
5. About
6. Contributor
7. Settings

1.2.1 Bus Schedule

Action: User tries to see bus schedule

Reply: System shows the bus schedule from the database

1.2.2 Bus Location

Action: User tries to see bus location

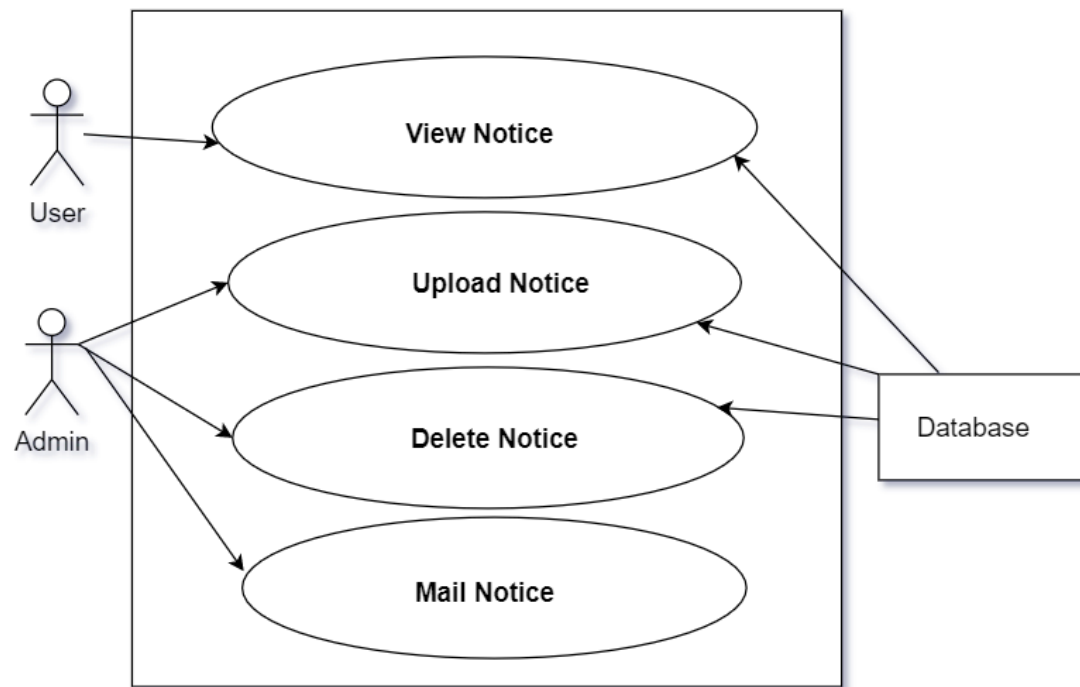
Reply: System shows the nearest bus location

THE NEXT DIAGRAM IS SHOWN IN THE NEXT PAGE

Use Case Diagram Level: 1.2.3

Name: Notification

ID No: 05



Description: Notification sub-system is divided into 4 main subsystems:

1. View Notice :

Action: User tries to see notice

Reply: System shows the bus notice from the database

2. Upload Notice :

Action: Admin tries to upload notice

Reply: System uploads the post in the database

3. SMS Notice :

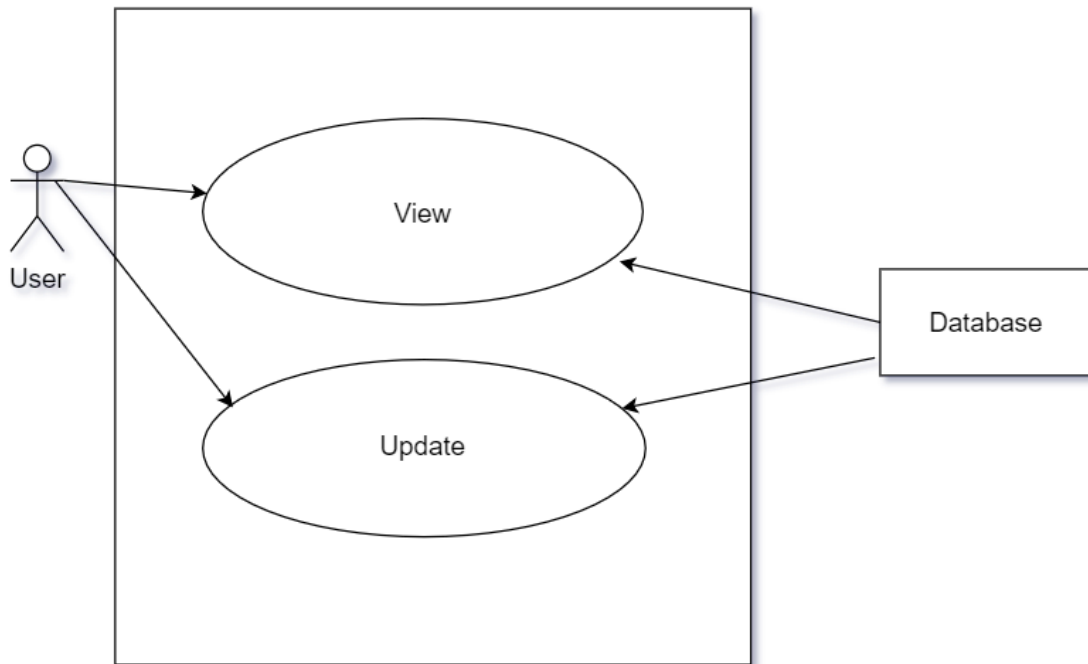
Action: Admin tries to mail the notice to the user

Reply: System mails the notification

Use Case Diagram Level: 1.2.4

Name: Profile

ID No: 06



Description: Profile sub-system is divided into 2 main subsystems:

1. View :

Action: User tries to view the profile

Reply: System shows the info from the database

2. Update :

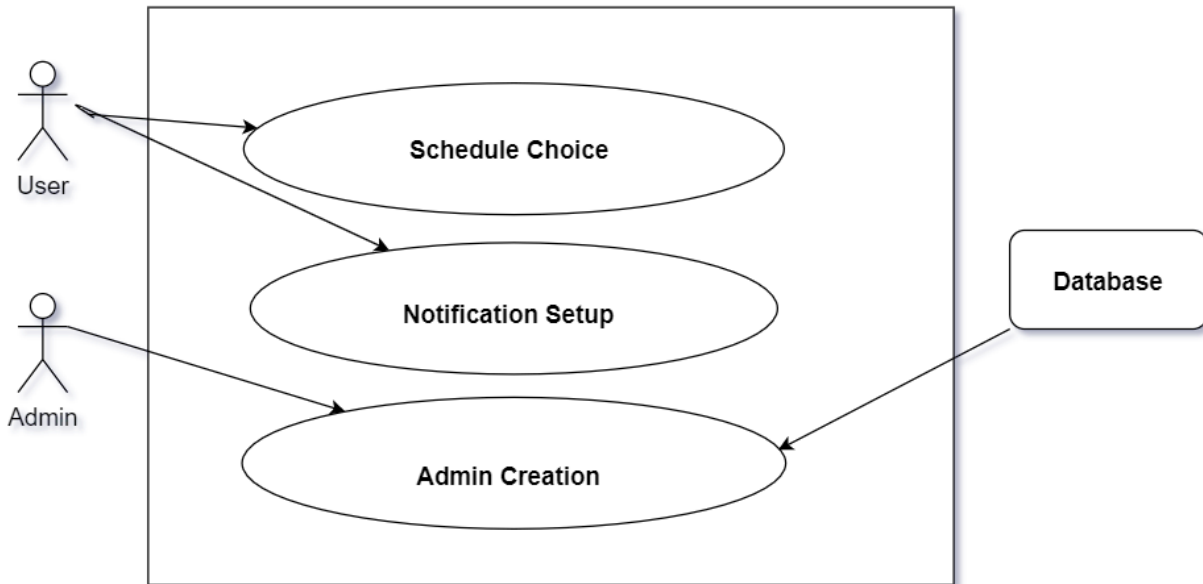
Action: User tries to **update** the profile pic and the phone number

Reply: System updates the database

Use Case Diagram Level: 1.2.7

Name: Settings

ID No: 07



Description: Settings sub-system is divided into 3 main subsystems:

1. Schedule Choice :

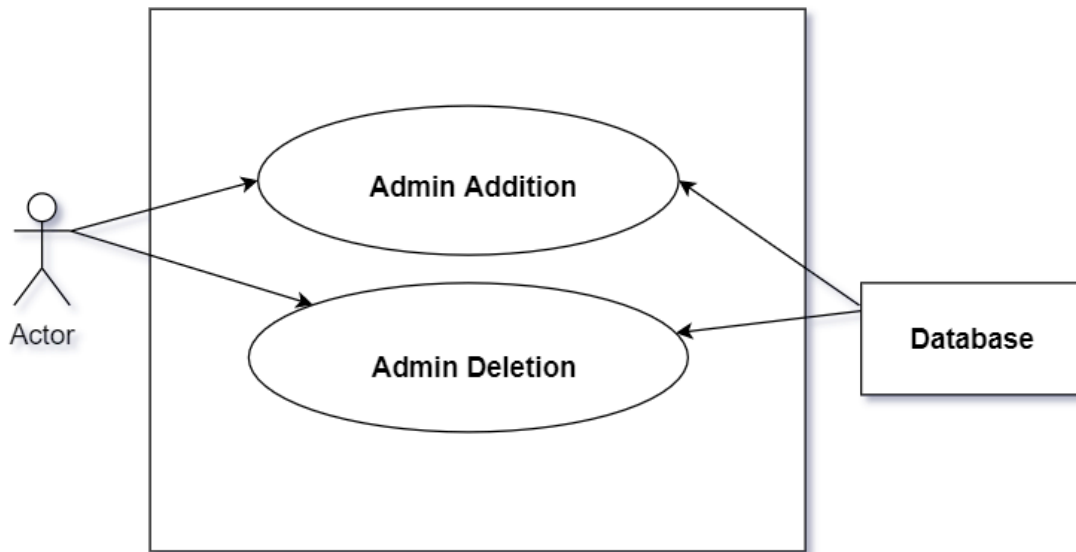
Action: User tries to choose summer / winter schedule

Reply: System updates the schedule

Use Case Diagram Level: 1.2.7.1

Name: Admin Creation

ID No: 08



Description: Admin Creation sub-system is divided into 2 main subsystems:

1. Admin Addition :

Action: Admin tries to create another admin with user email and password

Reply: System creates new admin after verification

Chapter Five

DATA-BASED MODELLING OF Bus Automation System

5.1 Data Modelling Concept

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

5.1.1 Data Objects

A data object is a representation of composite information that must be understood by the software. Here, composite information means information 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.

5.1.1.1 Data Object Identification

We identified all the nouns whether they are in problem space or in solution space from our story:

Problem (p)/ Solution(s) Space

No	Noun	p/s	Attribute
01	Registration	s	2,3,4
02	Students	s	6-10
03	Teachers	s	6-10
04	officers	s	6-10
05	Dhaka University	p	
06	user	s	6-10
07	User name	s	
08	Email address	s	
09	Phone number	s	
10	password	p	
11	digits	s	
12	Profile picture	s	2,3,4
13	Sign in	s	6-10
14	user	s	6-10
15	Welcome screen	p	2,3,4
16	icon	s	
17	bus	s	
18	Bus route	s	
19	system	s	
20	Bus schedule	s	
21	menu	s	
22	sections	s	

23	Bus location	s	
24	notification	s	
25	contributor	s	
26	about	s	
27	settings	s	
28	profile	s	6-10
29	Profile photo	p	
30	database	s	
31	suggestion	s	
32	Up-trip	s	
33	Down-trip	s	
34	summer	p	6-10
35	winter	s	6-10
36	location	s	
37	information	s	
38	notification	s	2,3,4
39	contributor	p	
40	Android app	s	
41	about	s	
42	settings	s	

5.1.1.2 Final Data Objects

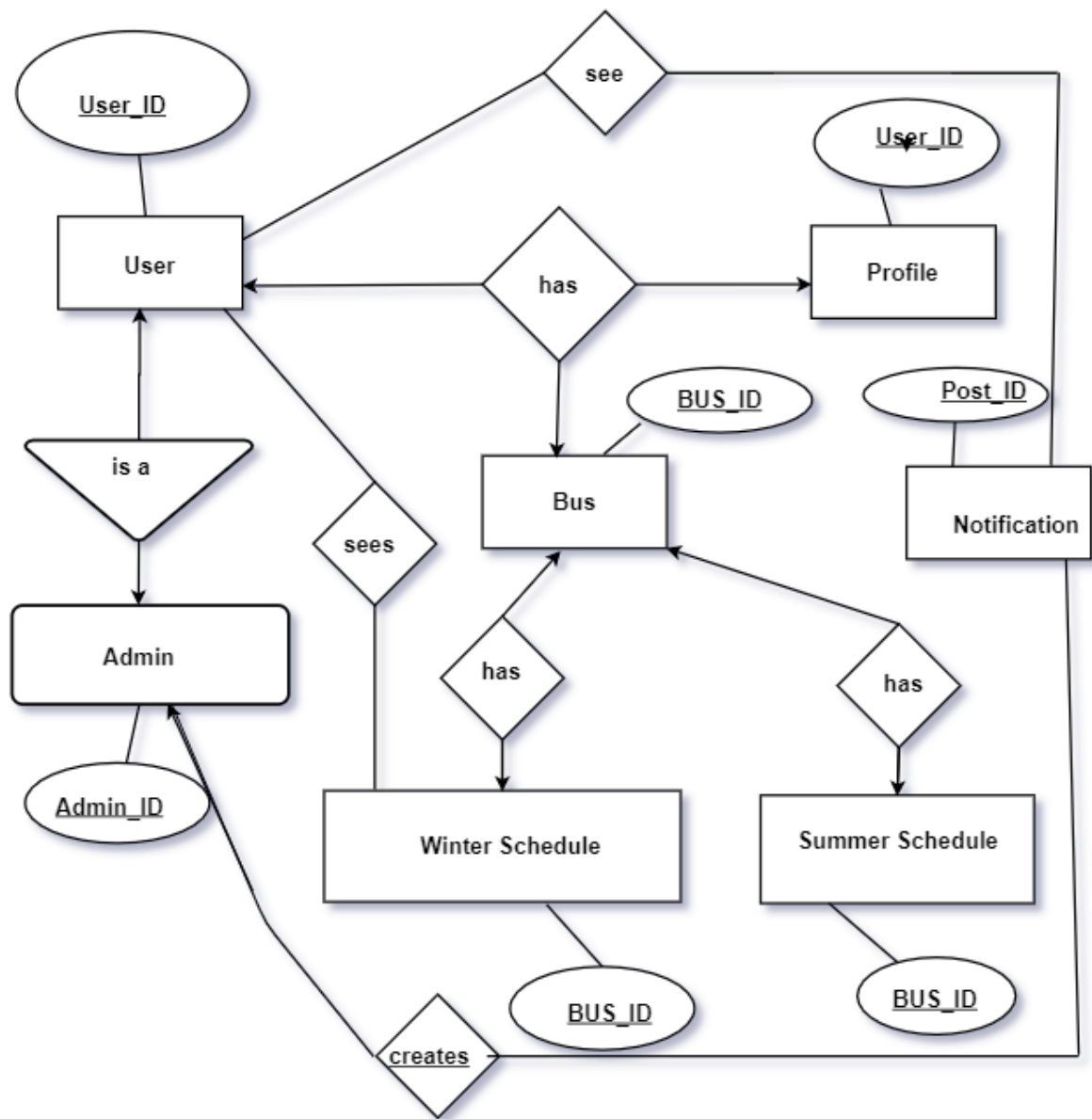
In the following table we finalize the data objects with their attributes Most of the attributes of the data objects are selected from the usage scenario and some of the attributes are selected to complete the system which are not in the usage scenario but important for the data objects.

1. User
2. Admin
3. Profile
4. Notification
5. Bus
6. Summer Schedule
7. Winter Schedule

5.2 DATA OBJECT RELATIONSHIPS

- One user has one profile (one to one relationship)
- Admin is a user
- Admin can create multiple notification (one to many)
- A user can search for different buses. A bus can have different users (Many to Many)
- A bus can have multiple notification (one to many)
- A bus has a summer schedule (one to one)
- A bus has a winter schedule (one to one)

5.3 ENTITY RELATIONSHIP DIAGRAM



5.4 SCHEMA DIAGRAM

Data Object	Attribute	Type	Size
Bus	name	String	

	<u>BUS ID(PK)</u>	Number	
	<u>User ID</u>	Number	
	<u>Post ID</u>		

Data Object	Attribute	Type	Size
user	name	sting	40
	Phone number	int	
	image	jpg	
	email	email	
	<u>User ID (PK)</u>	NUMBER	
	<u>Bus ID</u>	NUMBER	

Data Object	Attribute	Type	Size
Admin	name	sting	40
	Phone number	int	
	image	jpg	
	email	email	
	<u>Admin ID</u>	NUMBER	
	<u>User id</u>	NUMBER	
	<u>Post id</u>		

Data Object	Attribute	Type	Size
Notification	<u>Post id (PK)</u>	NUMBER	
	<u>Admin ID</u>	NUMBER	
	<u>Bus ID</u>	NUMBER	

Data Object	Attribute	Type	Size
Summer Schedule	<u>Bus ID</u>	NUMBER	
	<u>User ID</u>	NUMBER	

Data Object	Attribute	Type	Size
Winter Schedule	<u>Bus ID</u>	NUMBER	
	<u>User ID</u>	NUMBER	

Data Object	Attribute	Type	Size
Profile	name	string	40
	Phone number	int	
	image	jpg	
	email	email	
	<u>User ID (PK)</u>		

Chapter Six

Class Based Modelling for Bus Automation System

6.1 CLASS BASED MODELING CONCEPT

This chapter is intended to describe class based modelling of “Bus Automation System “.

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

6.2 List of Nouns for CMS

Noun List	
Registration	
Students	
Teachers	
officers	
Dhaka University	
user	
User name	
Email address	
Phone number	
password	
digits	
Profile picture	
Sign in	
user	
Welcome screen	
icon	
bus	
Bus route	
system	
Bus schedule	

menu	
sections	
Bus location	
notification	
contributor	
about	
settings	
profile	
Profile photo	
database	
suggestion	
Up-trip	
Down-trip	
summer	
winter	
location	
information	
notification	
contributor	
Android app	
about	
settings	

6.3 List of Verbs

Verb List	
Register	
provide	
contain	
create	
see	
Sign in	
write	
show	
edit	

see	
store	
give	
search	
show	
get	
sent	
bear	
Turn off	
change	

6.4 GENERAL CLASSIFICATION

Candidate classes were then characterized in seven general categories. The seven general characteristics are as follows:

1. **External entities** (e.g. other systems, devices, people) that produce or consume information to be used by a computer-based system.
2. **Things** (e.g., reports, displays, letters, signals) that are part of the information domain for the problem.
3. **Occurrences or events** (e.g., a property transfer or the completion of a series of robot movements) that occur within the context of system operation.
4. **Roles** (e.g., manager, engineer, salesperson) played by people who interact with the system.
5. **Organizational units** (e.g. division, group, teams) that are relevant to an application.
6. **Places** (e.g., manufacturing floor or loading dock) that establish the context of the problem and the overall function of the system.
7. **Structures** (e.g., sensors, four-wheeled vehicles, or computers) that define a class of objects or related classes of objects.

Following are the specifications of the nouns according to the general Classifications:

Table 1: General Classification:

Noun List	General Classification
Registration	2, 3, 7
Students	4, 5, 7
Teachers	4, 5, 7
officers	4, 5, 7
Dhaka University	2
welcome	4,5,7
Homepage	1,2,3,4,7
user	1,2,3,4,7
User name	2,4,7
Email address	2,4,7
Phone number	1,2,4,7
password	1,2,3,4,7
digits	1,2,3,4,7
Profile picture	2,4,7
Sign in	2,4,7
user	1,2,4,7
Welcome screen	1,2,3,4,6
icon	1,2,3
bus	2,4,7
Bus route	2,4,7
system	1,2,4,7
Bus schedule	1,2,3,4,7
menu	1,2,3,4,7
sections	2,4,7
Bus location	2,4,7
notification	1,2,4,7
contributor	1,2,3,4,7
about	1,2,3,4,7
settings	2,4,7
profile	2,4,7
Profile photo	1,2,4,7

database	1,2,3,4,7
suggestion	1,2,3,4,7
Up-trip	2,4,7
Down-trip	2,4,7
summer	1,2,4,7
winter	1,2,3,4,7
location	1,2,3,4,7
information	2,4,7
notification	2,4,7
contributor	1,2,4,7
Android app	1,2,3,4,7
about	1,2,3,4,7
settings	2,4,7

6.5 SELECTION CRITERIA

The candidate classes are then selected as classes by six Selection Criteria. A candidate class generally becomes a class when it fulfills around three characteristics.

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

Table 2: Selection Criteria

Noun List	SELECTION CRITERIA
Registration	1, 2, 3, 6
Students	1,2,3,4,5,6
Teachers	1,2,3,4,
officers	1,2,3,4,5,6

Dhaka University	
user	1,2,3,4,5,6
User name	
Email address	
Phone number	
password	
digits	
Profile picture	
Sign in	1,2,,4,5,6
user	1,2,3,4,,6
Welcome screen	
icon	
bus	
Bus route	
system	
Bus schedule	1,2,3,4,5
menu	
sections	
Bus location	2-6
notification	1-6
contributor	2,3,4
about	2,3,4
settings	
profile	1-6
Profile photo	1
database	
suggestion	
Up-trip	2
Down-trip	2
summer	
winter	
location	
information	2-6
notification	2,4,6

contributor	2,4,6
Android app	
about	
settings	1-6

The nouns which are selected as probable class are:

1. Students
2. Teachers
3. Officers
4. Homepage
5. Registration
6. Bus Search
7. Bus Location
8. Schedule
9. Profile
10. Database
11. Notification
12. Photo Upload
13. Contributor
14. About

6.6 ATTRIBUTE AND METHOD IDENTIFICATION

After identifying the classes, we have specified their attributes and methods.

Class Name	Attribute	Method
Students	<ul style="list-style-type: none"> ● User name ● Password ● Registration number ● Phone number 	

Teachers	<ul style="list-style-type: none"> • User name • Password • Registration number • Phone number 	
Officers	<ul style="list-style-type: none"> • User name • Password • Registration number • Phone number 	
Registration	<ul style="list-style-type: none"> • User name • Password • Registration number • Phone number • User type • Profile pic 	Get image from gallery() verify email () verify user type()
Sign in	<ul style="list-style-type: none"> • User Email • Password 	VERIFY USER()
Menu		Show menu()
Bus Search	<ul style="list-style-type: none"> • Bus name 	Show bus schedule()
Bus Location	<ul style="list-style-type: none"> • Bus name • Bus number 	Show nearest bus ()
Schedule	<ul style="list-style-type: none"> • Bus number 	Show up-trips () Show down-trip() Show bus number()
Profile	<ul style="list-style-type: none"> • User name • Password • Registration number • Phone number • User type • Profile pic 	Show profile picture() Show user info () Edit phone number() Edit profile picture()
Database		
Notification	<ul style="list-style-type: none"> • Notice id 	View() Update notice () Delete notice () Send mail()
Contributor	<ul style="list-style-type: none"> • Contributor profile pic 	Show contributor()

Analysis:

Here students, teachers and officers have the same attributes and methods. So, they can have a common name. So, they can be eliminated by the class “user”.

6.7 CRC Card

Class Card After identifying our final classes we have generated the following class cards.

Class Name	Responsibility	Collaborator
Menu	Shows different features	User Admin Database Contributor About Notification Profile settings
Settings	Sets schedule Turns off notification Creates admin Deletes admin	User Admin Bus schedule
Schedule	Show up-trips Showdown-trip Show bus number	User Bus search admin
Profile	Shows profile picture Shows user info Edits phone number Edits profile picture	Database user
Database	Stores user information Verify user Fetch user information	Registration Profile Sign in
Notification	Creates notice Updates notice Deletes notice Sends mail	User database
Photo Upload	Uploads image Picks image	User Admin Contributor Registration Profile Database

6.8 Final CRC Card

Class Card After identifying our final classes we have generated the following class cards:

Sign In	
Attribute	Method
<ul style="list-style-type: none">• User Email• Password	Verify user()
Responsibility	Collaborator

User	
Attribute	Method
<ul style="list-style-type: none">• User name• Password• Registration number• Phone number• User type	
Responsibility	Collaborator
	Registration

Registration	
Attribute	Method
<ul style="list-style-type: none">• User name• Password• Registration number• Phone number• User type• Profile pic	Get image from gallery() verify email () verify user type()
Responsibility	Collaborator
	User admin

Homepage	
Attribute	Method

	Show map () Show current location () Search bus ()
Responsibility	Collaborator

Bus Search	
Attribute	Method
<ul style="list-style-type: none"> Bus name 	Show bus schedule()
Responsibility	Collaborator
Shows bus schedule	Bus schedule

Bus Schedule	
Attribute	Method
Bus number	Show up-trips () Showdown-trip() Show bus number()
Responsibility	Collaborator
Show up-trips Showdown-trip Show bus number	User Bus search admin
Profile	
Attribute	Method
<ul style="list-style-type: none"> User name Password Registration number Phone number User type Profile pic 	Show profile picture() Show user info () Edit phone number() Edit profile picture()
Responsibility	Collaborator
Shows profile picture Shows user info Edits phone number Edits profile picture	Database user

Bus Location	
Attribute	Method
Bus name	Show nearest bus ()

Bus number	
Responsibility	Collaborator
	User admin

Database	
Attribute	Method
Responsibility	Collaborator
Stores user information Verify user Fetch user information	Registration Profile Sign in

About	
Attribute	Method
	Show app info()
Responsibility	Collaborator
Shows information about application	user

Contributor	
Attribute	Method
Contributor profile pic	Show contributor()
Responsibility	Collaborator
Show info about contributor	user

Notification	
Attribute	Method
Notice id	View() Update notice () Delete notice () Send mail()
Responsibility	Collaborator
Creates notice Updates notice Deletes notice Sends mail	User database

Image Upload

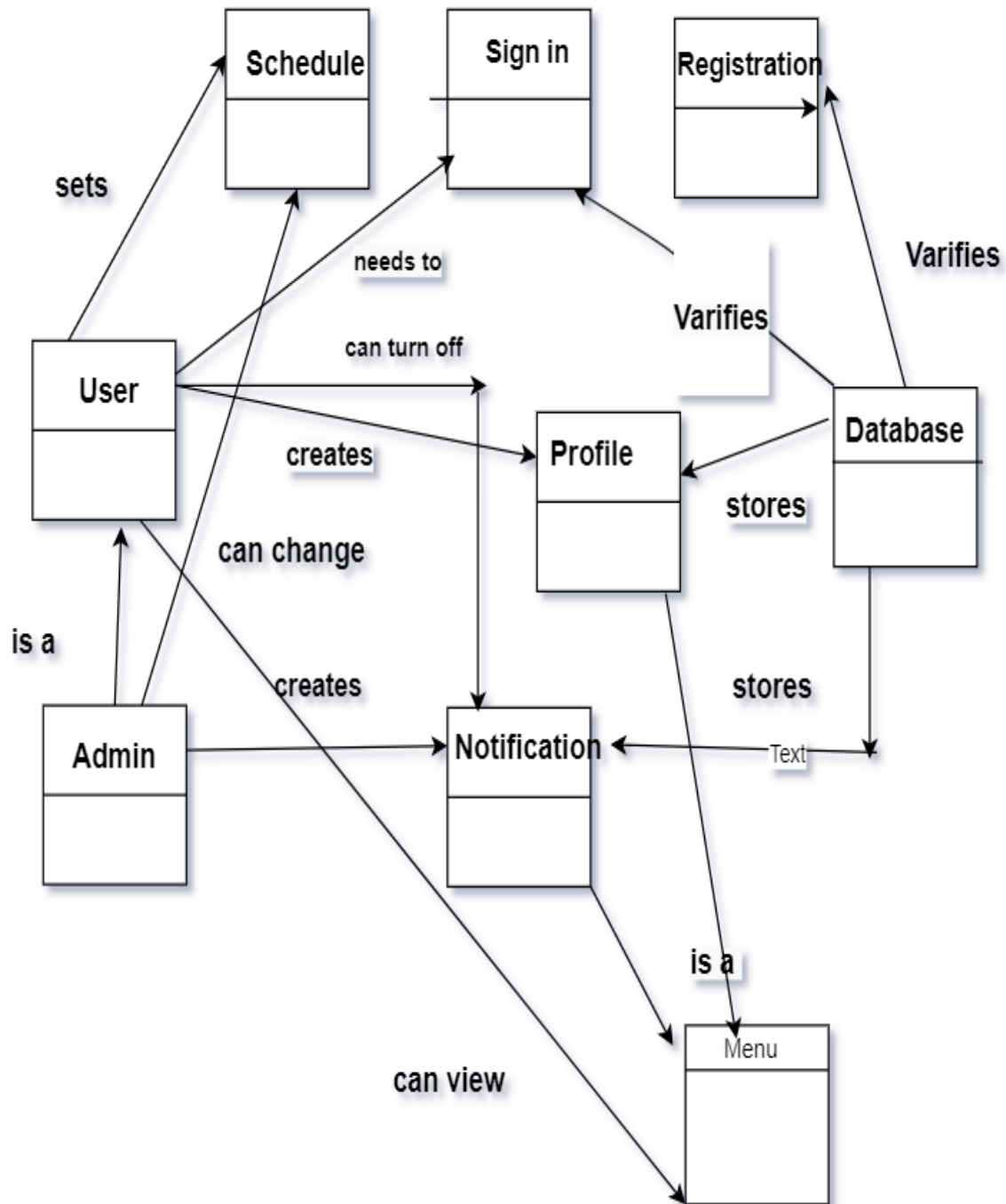
Attribute	Method
image	Upload image() Pick image()
Responsibility	Collaborator
Uploads image Picks image	User Admin Contributor Registration Profile Database

Menu	
Attribute	Method
	Show menu()
Responsibility	Collaborator
Shows different features	User Admin Database Contributor About Notification Profile settings

Settings	
Attribute	Method
	Set schedule() Turn off notification() Create admin() Delete admin() Delete user()

Responsibility	Collaborator
Sets schedule Turns off notification Creates admin Deletes admin	User Admin Bus schedule

6.8 Class Diagram



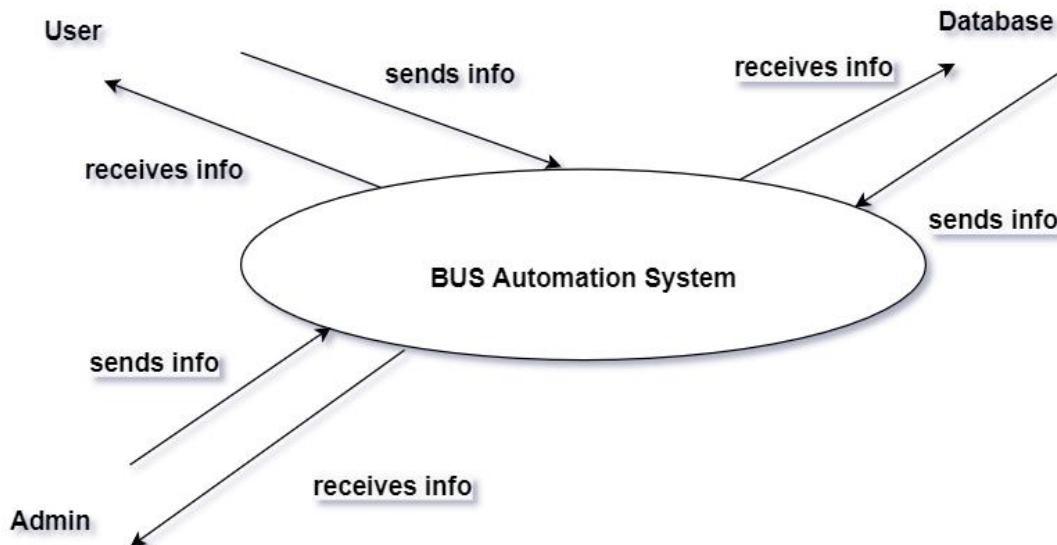
Chapter Seven

Flow Oriented Model of Bus Automation System

Data Flow Diagrams:

ID -01

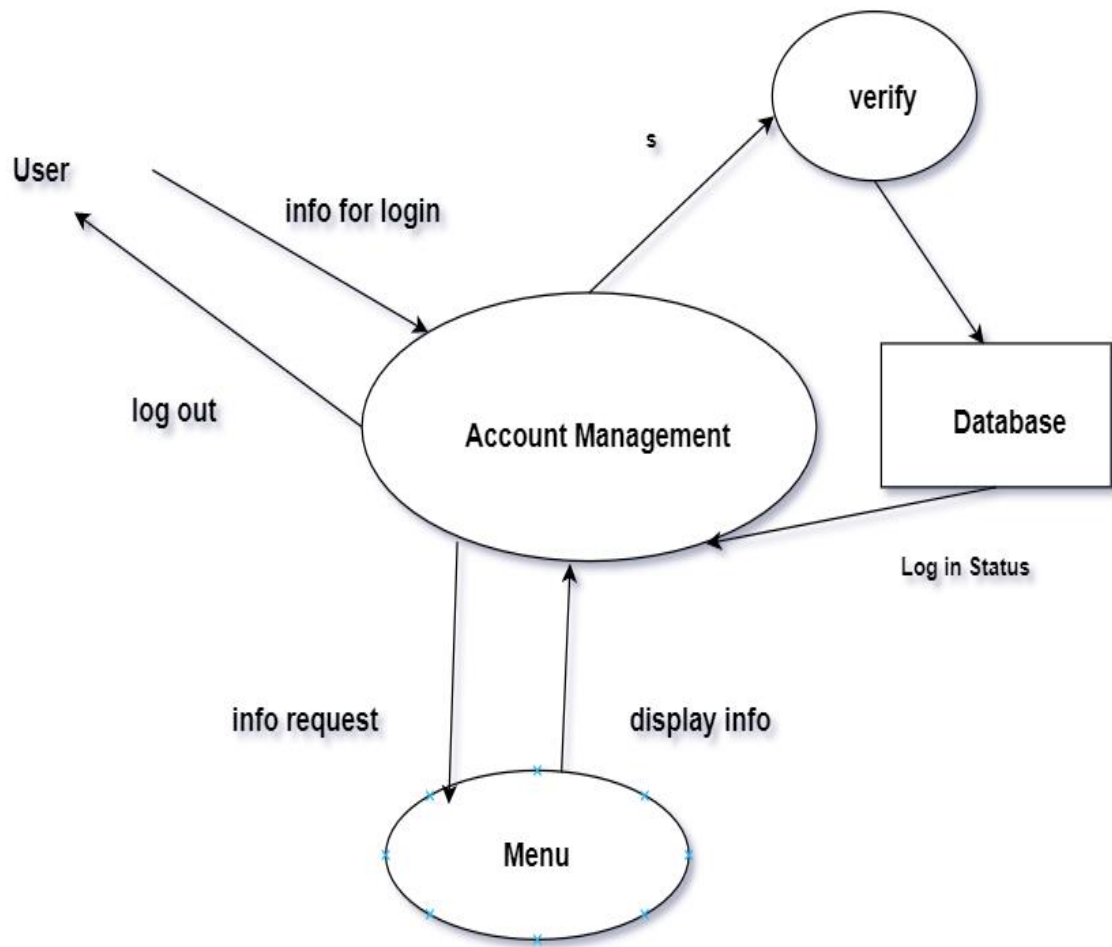
Name: Birds Eye View



Description: This is the level 0 DFD of the system. This will be further divided into smaller sub-systems in DFD

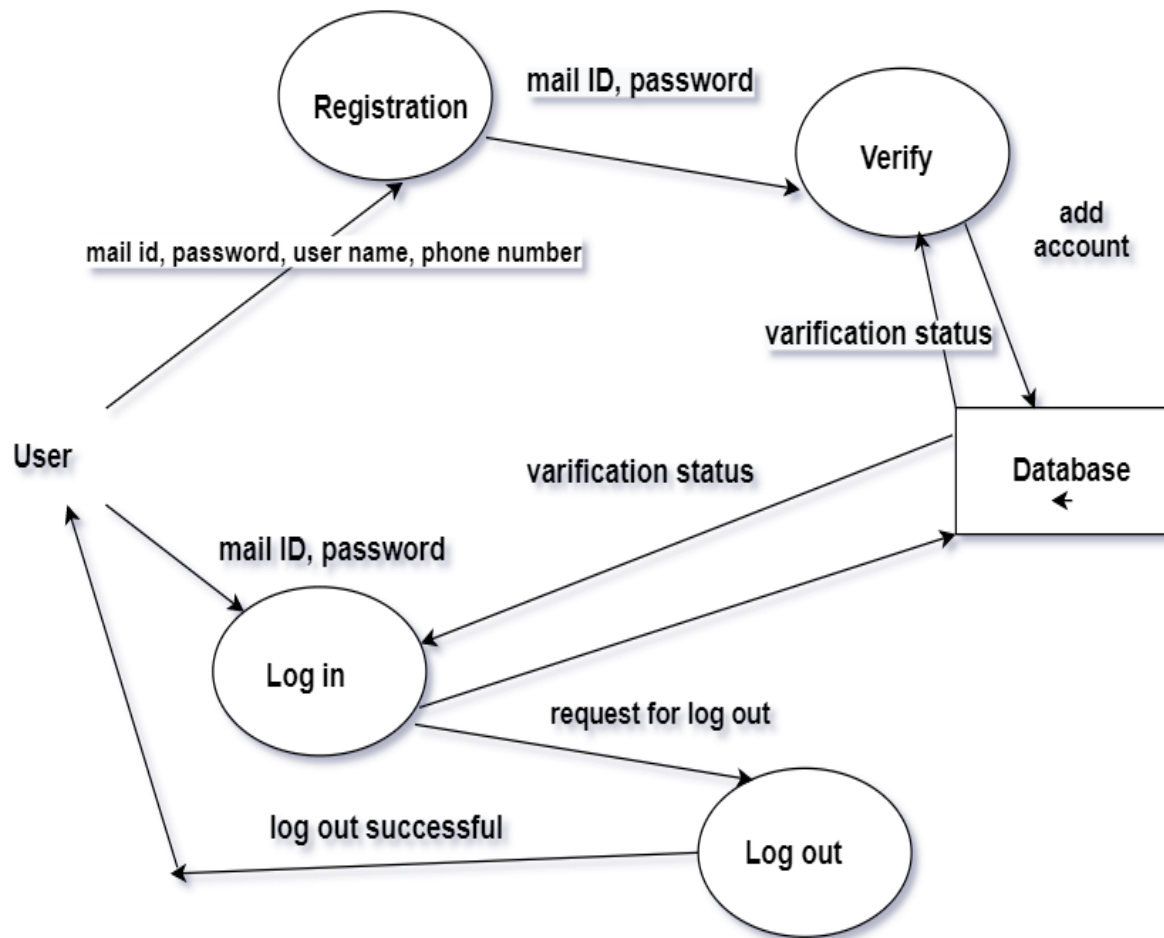
ID- 02

Name: Bus Automation System



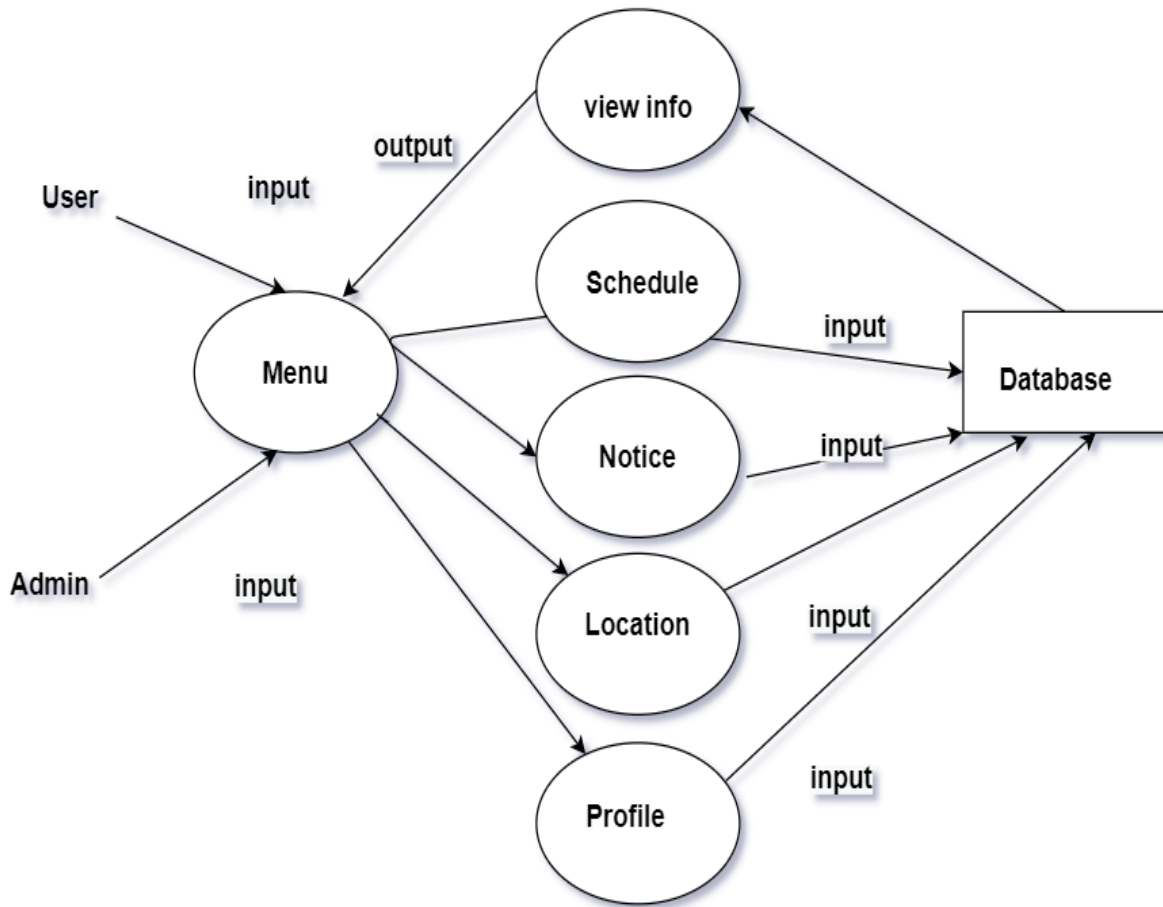
ID- 03

Name: Account Management



ID- 04

Name: Menu



Chapter Eight

Behavioral Modelling OF Bus Automation System

Identifying Events

Serial No.	Event	State Name	Initiator	collaborator
1	Create an account	Create account	Use Manager	Database, Authentication, University, Notification Staff
2	Get notification	Notify	Notification	User Manager staff
3	Can log in	Log in	User Manager staff	Database
4	Can log out	Log out	User Manager staff	
5	Can recover account	Recover Account	User Manager Staff	Database Notification
6	Can edit items to the menu	Edit item	manager	Stock management Notification database
7	Can give order for reserving a table	Reserve table	User	Order processing Payment Database notification
8	can Notify	Confirm Order	staff	Database notification
9	Show event calendar	Event calendar	manager	Event management

10	Give event info	Give event details	user	manager
11	Give event confirmation	Confirm event	manager	Notification Database
12	Give feedback	Feedbacks	user	Manager database

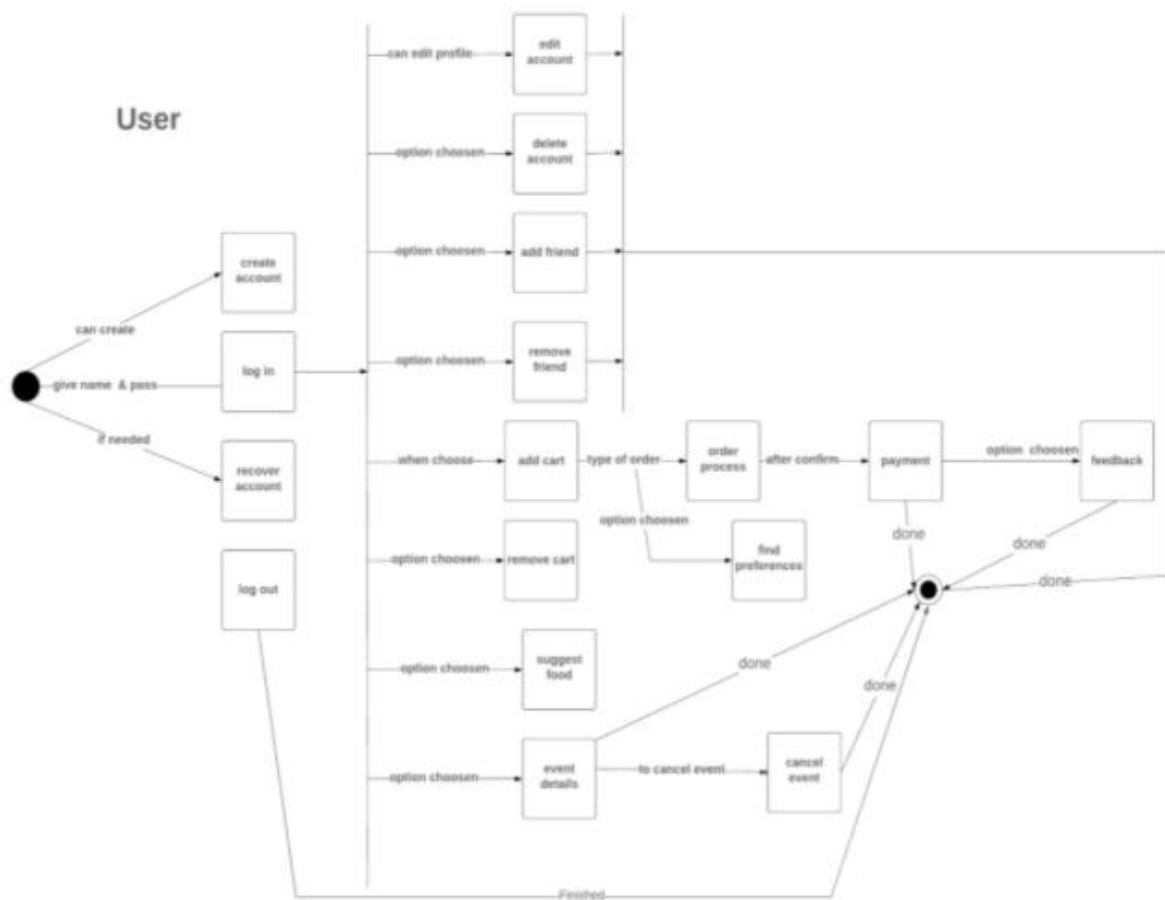


Figure 86: State Transition Diagram-User

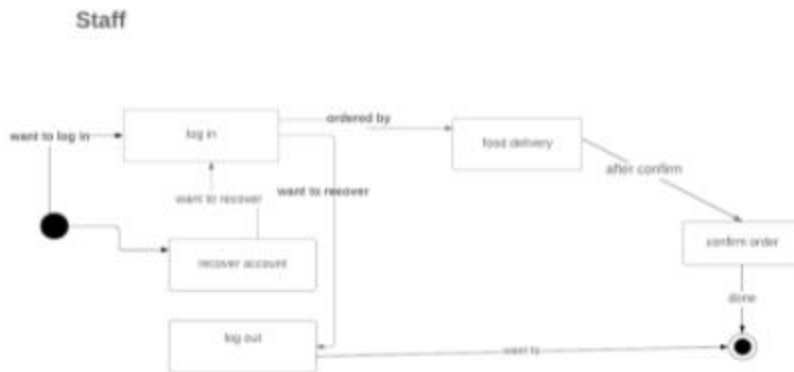


Figure 88: State Transition Diagram-Staff



Figure 89 : State Transition Diagram-Notification

This is the complete SRS for BUS AUTOMATION SYSTEM DU CHAKKA