Dhaka University Calendar Management Application

Submitted by

Pritom Kumar Das	BSSE0919
Miran Al Mehrab	BSSE0931

Supervised By

Dr. Kazi Muheymin-Us-Sakib
Professor
Institute of Information Technology,
University of Dhaka

Date: 20/03/2019

Letter of Transmittal

Dr. Kazi Muheymin-Us-Sakib
Professor
Institute of Information Technology,
University of Dhaka

Subject: Submission of term report on “Dhaka University Calendar Management Application”.

Sir,
With due respect, we would like to submit you the report on Software Requirement Specification on the above topic you assigned us. The report reflects our effort to gather requirements and analyzing them. We have included every steps what we have done through the whole time for requirement specification of the mentioned topic.

Therefore, we earnestly hope that you will excuse our error and obliged thereby.

Yours Sincerely,

Pritom Kumar Das BSSE0919
Miran Al Mehrab BSSE0931

3rd year, 5th semester, 9th batch

Institute of Information Technology

University of Dhaka

Session: 2016-17

Acknowledgement

We are highly indebted for getting such a tremendous opportunity to prepare the report on Dhaka University Calendar Management Application .We would like to thank whole-heartedly our supervisor, Dr. Kazi Muheymin-Us-Sakib , Professor , Institute of Information Technology, University of Dhaka, for giving us guidelines about how we can prepare this report. We are also thankful of those people who gave requirements to do this job.

Abstract

The calendar committee of Dhaka University publishes a calendar for the very first year students of the university to be familiar with the University campus and its glorious history every year . We are fortunate to take the task of creating an android application that represents Dhaka University Calendar . So, this is the SRS (Software Requirements Specification) of Dhaka University Calendar Management Application from the gathered requirements. The report represents the objective and also describes the need of this trendy strategy.

Contents

Chapter 1: Introduction

This chapter is a part of "Dhaka University Calendar Management Application" intended to specify the purpose of this document and the intended audience of it.

1.1 Purpose

This document is the Software Requirements Specification (SRS) for "Dhaka University Calendar Management Application". It contains detailed functional, non-functional, and support requirements and establishes a requirements baseline for development of the system. The requirements contained in the SRS are independent, uniquely numbered, and organized by topic. The SRS serves as the 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.

1.2 Intended Audience

This SRS is intended for several audiences, including the customer 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.

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 2: Inception

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

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

- Identifying Stakeholders
- Recognizing multiple viewpoints
- Working towards collaboration
- Asking the First Questions

2.1.1 Identifying 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. The initial list will grow as stakeholders are contacted because every stakeholder will be asked: "whom else do you think I should talk to?"
The following stakeholders were identified for the "Dhaka University Calendar Management Application".

- **Dhaka University Administrative Body:** As the highest ranking Administrative body of Dhaka University they take all the important decisions concerning Dhaka University . Only with their support and approval this application can serve its purpose to its fullest potential.
- **Dhaka University Calendar Committee** : As the committee responsible for delivering and maintaining Dhaka University Calendar each year they are considered the Admin of this application . They operate to gather the necessary information needed for the application .
- **Student:** A student is a first year student of Dhaka University . They are the main users of this application. A student provides his/her information to create his/her profile .
- **Teacher :** As they are the faculty members of various departments/institutions of Dhaka University they can use the app to find relative information . They are also part of the Dhaka University Administrative Body and Dhaka University Calendar Committee.
- **Software Developer:** A software developer is concerned with facets of the software development process, including the research, design, programming, maintenance and testing of computer software. He/She will be responsible for the outcomes of the software.

2.1.2 Recognizing Multiple Viewpoints

Different stakeholders demand different features from the software. To satisfy the stakeholders, most of these features should be included in the software.

Dhaka University Administrative Body's viewpoint

- Error free system
- Getting all the useful information about Dhaka University
- Checking the security of the system
- Check the validity of provided information
- Keeping record of users
- Update any information that might be wrong
- Maintain public image of Dhaka University
- Strong authentication system

Dhaka University Calendar Committee's viewpoint

- User friendly
- Error free system
- Keeping a track of new information provided by departments/institutes
- Getting accurate information and check validity of the information
- Easy maintainability
- Add new structures that's not currently in the provided information
- Respond to any complaints and provide solutions
- Strong authentication system
- Response to security issues
- High scalability
- Keeping a backup of the data

Student's viewpoint

- User friendly
- Quick search result for information
- Error free system
- Easy to use
- Fast response
- Easy to connect with friends
- Check history on conversation

Teacher's viewpoint

- User friendly
- Quick search result for information
- Error free system
- Check validity of the information
- Check history

Developer's viewpoint

- Easy to built
- Error free effective software
- Readable and effective code review
- No conflicting requirement
- Easy to add expand or change software
- Create secure software

2.1.3 Working towards collaboration

While working with different stakeholders, some conflicting and common viewpoints can be noticed. For this reason, final requirements can be gotten by collaborating the viewpoints. We followed following steps to merge these requirements:

- Identify the common and conflicting requirements
- Categorize the requirements
- Take priority points for each requirements from stakeholders and on the basis of this voting prioritize the requirements
- Make final decision about the requirements

Common Requirements:

- Web based interfaces
- The application can be accessed from any computer that has internet access.
- Allow any user to search for Information
- Attractive and easy to use User Interface
- Secure application
- Easy to maintain
- Maintain a database for all users and information in the system

Conflicting Requirements:

- Strong authentication problem when check out
- Whether login required or not to use the system
- Design of the User Interface
- Weather or not add messenger in application
- Separate application for admins

Final Requirements:

- Error free easy accessible system
- Android application
- Separate web based interface for admins
- Accessible via the Internet.
- Allow valid users to login and logout.
- Restrict access to functionality of the system based upon user roles
- Allow administrators of the system to change provided information and configure parameters of the system
- Allow any user to search for information in the application without having to log in to the system
- Allow valid users that log in to use exciting features of application.
- Allow Administrators to delete , modify, add or update any information regarding Dhaka University
- This application can be used by any android device with connection to internet.
- A dedicated server have to run all the time to help users get information.
- Maintain proper security for all the data.
- Maintain a database of all users and information.
- Allows user option to create profile or not.
- Allows user option to change language.
- Allows user to contact with admin through email if needed

Restrict access to functionality of the system based upon user roles. For example, only Administrators of the system will be provided functionality to change static information of Dhaka university .

2.1.4 Asking the First Questions

We set our first set of context-free questions focuses on the students 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 user to voice his or her perception about the solution. The final set of question focused on the effectiveness of the communication activity itself and the acceptability of the application itself.

2.2 Conclusion

Inception phase helped us to establish basic understanding about "Dhaka University Calendar Management Application" and its importance in Dhaka University, identify the people who will be ben**e**fited if Dhaka University Calendar Management System becomes automated, define the nature of the Calendar management software and establish a preliminary communication with our stakeholders. 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 3: Elicitation

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 students, admin and other stakeholders. Many difficulties were faced, like understanding the problems, making questions for the stakeholders, limited communication with the stakeholders due to a short amount of time and volatility. Though it is not easy to gather requirements within a very short time, these problems have been surpassed in an organized and systematic manner.

3.2 Eliciting Requirements

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. The following tasks were done for eliciting requirements:

1. Collaborative Requirements Gathering
2. Quality Function Deployment
3. Usage Scenarios
4. Elicitation work products

3.3 Collaborative Requirements Gathering

Many different approaches to collaborative requirements gathering have been proposed. Each makes use of a slightly different scenario. We completed following steps to do it.

- The meetings were conducted with the Dhaka University Calendar Committee members and they were questioned about their requirements and expectations from the Dhaka University Calendar Management Application.
- The Calendar Committee was asked to see there satisfaction with the current system.
- At last we selected our final requirements from the meetings

3.4 Quality Function Deployment

Quality Function Deployment (QFD) is a technique that translates the needs of the users into technical requirements for software. It concentrates on maximizing user satisfaction from the Software engineering process. With respect to our project the following requirements are identified by a QFD.

3.4.1 Normal Requirements

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

1. Error free easy accessible system
2. Android application
3. Effective System
4. Separate web based interface for admins
5. Accessible via the Internet.
6. Allow valid users to login and logout.
7. Restrict access to functionality of the system based upon user roles
8. Allow administrators of the system to change provided information and configure parameters of the system
9. Allow any user to search for information in the application without having to log in to the system
10. Allow valid users that log in to use exciting features of application.
11. Allow Administrators to delete , modify, add or update any information regarding Dhaka University
12. This application can be used by any android device with connection to internet.
13. A dedicated server have to run all the time to help users get information.
14. Maintain proper security for all the data.
15. Maintain a database of all users and information.
16. Allows user option to create profile or not.
17. Allows user option to change language.
18. Allows user to contact with admin through email if needed
19. Fast loading application.
20. Provide users with Dhaka University Map.

3.4.2 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.

1. Error free software
2. Strong authentication system
3. User friendly
4. Effective system
5. No ambiguous feature
6. Data backup
7. Sending notification to users if necessary.

3.4.3 Exciting requirements

These requirements are for features that go beyond the customer's expectations and prove to be very satisfying when present

1. The user interface should provide appropriate error messages for invalid input or show message if search results are not found.
2. The user interface should follow standard web practices such that the web interface is consistent with typical internet applications.
3. Offer log in with mobile phone
4. Users will also be given a list of exciting places within Dhaka University .
5. There will be a personalized calendar for each student which will contain academic schedule.
6. Messenger for first year students.

Usage scenario

This is an android application that is entitled to the name “Dhaka University Calendar”. People can avail this application to be familiar with Dhaka University by using the basic features such as – infrastructure info, transport info, map info, general calendar, university map as well as exciting feature like specific calendar and group chat.

Account Creation

Sign Up:

The user can download the application from Google play store or Dhaka University main website page (admission page). Any person with Google account are eligible for sign up and to complete their profile the users will have to provide their user name and type, such as- student, teacher or other. Depending on their type they will be required to enter their department/institute name and academic year.

Sign In:
After sign up is done, users can sign in to their account with Firebase Auth 2.0 Sign In.

Update account:
Logged users can update their profile by entering new name, department/institute name or academic year.

Delete account: Logged users can delete their account as their wish any time and their data from firebase database will be deleted accordingly.

User Based Privilege

As exciting features are specific for different users, so these features are shown according to the user type:

Other: This type of users can use the basic features, such as - infrastructure info, transport info, map info, general calendar and university map.

Teachers: Teachers can use the basic features as well as specified calendar, group chat based on their given department.

Student: Students can use the basic features can use all the basic features as well as specified calendar, group chat based on their given academic year.

Management Parts of Project

Information management:

It includes university brief history, notable achievement, information about Administrative body, department / institute description, halls, Club details, description of popular places in university campus. Description of the information management parts are given below:

Dhaka University Administrative body: This includes the name, rank and picture of the Dhaka University Administrative body.
Administrative committee: This includes the name and rank of the administrative committee members.
Editorial committee: This includes the name and rank of the editorial committee members.
List of Faculty Deans: This includes a list of faculty names and its Deans.
Department /Institute/hall/offices: This includes department /institute name, establishment year , description of the department/institutes/hall/offices and locations.

Transport Management:

Users can see the schedule of buses on different route and their stoppage on the way.

Map management:

Users can see their current position. They can also search and get route for their destination. They can also see exciting places near their position within the university area.

Communication Management:

A group chat option is available for the students based on their academic session and department. Students can also leave and rejoin this chat group as their wish.

Calendar Management:

There will be two kinds of calendar for the users. A general calendar that entails all holiday and events marked by Dhaka University calendar committee is available for general or other users. And there will be a specific calendar for each department / institute that will mark down all the academic activities as well as university holidays.

Administrative Responsibilities

There will be multiple role based admin account for this project. They will have a different web interface for their convenience. They can upload, update or delete information to the server and database will be updated accordingly. They have the responsibility to maintain the database and server that's providing service for the application. They will have ensure the data integrity and security.

Design Overview of Application

Homepage:

There will be icons for all the functions such as map, calendar, transport, chat group, infrastructure icons in the homepage. Users will find a search option to search information. In the settings option users can change the language to Bengali or English.

Other Pages:

Users can go to other pages from home page. The design of the page will be created on the content of the page like – the combination of text, image or other buttons.

Search

Users can search for any information in the application. The search string will be used to generate database query to find information from the database. If result of the query is null then it will show specific page regarding no result found. If query result is found, then link to the page will be shown to the users.

Chapter 4: Scenario Based Modeling

4.1 Introduction

In this model 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.

4.2 Use Case Scenario

Table 1: Use Case Scenario

Level 0	Level 1	Level 2	Level 3
Dhaka University Calendar	Account management	Sign up	
		Log out	
		Log in	
		Update	
		Remove	
	Information management	Infrastructure	University
			Information modification
			Department
			Hall
			Club
			Office
		Administrative body	Governing body
			Administrative committee
			Editorial committee
			Faculty deans
			Governing body modification
		Transport	
		Update information	
		Search location	

	Map management	Nearby places	
		Location suggestion	
		Location modification	
	Calendar management	General calendar	
		Specific calendar	
		Calendar modification	
	Communication management		

4.3 Use Case Descriptions

Use case: Sign up

Primary actor: user
Goal in context: to create an account
Preconditions:
1.System has been designed to have sign up option
2.System has interface to sign up

Triggers: User have to sign up
Scenario:
1. Get authenticated using OAuth 2.0
2. Enter user type, user name, department, academic session

Exception:
1. Already authenticated

Priority: Essential, must be implemented
When available: First increment

Use case: Sign in

Primary actor: user
Goal in context: to enter into system
Preconditions:
1.System has been designed to have sign up option
2.System has interface to sign up
3.User must be authenticated using oauth 2.0

Scenario:
1.Click on the right authenticated Google account

Exception:
1.User is blocked

Priority: Essential, must be implemented
When available: First increment

Use case: update account
Primary actor: user
Goal in context: change user information
Preconditions:
1.System has been designed to have update option
2.System has interface to update
3.User must be logged in

Scenario:

1.Click on the right interface option
2.Enter required data in the fields
3.confirm to change

Triggers: User has to change his department or academic year.
Exception: 1.information is like the old one.
Priority: Essential, must be implemented
When available: First increment

Use case: Log out

Primary actor :user
Goal in context: get out of the system
Preconditions:

1.System has been designed to have log out option
2.System has interface to og out
3.User must be logged in

Scenario:

1. Click on the log out option on the interface.

Triggers:

User needs to stop current system service.

Exception:

1.User isn't logged in.

Priority: Essential, must be implemented
When available: First increment

Use case: Transport info

Primary actor: user
Goal in context: to see available transport service in the university
Preconditions:

1.System has been designed to have transport info option
2.System has interface to transport info option
3.User must be logged in

Scenario:

1.Click on the transport info option

Triggers: User want to use desired bus service.
Exception:

1.User isn't logged in.
2. Data not available

Priority: Optional, should be implemented
When available: First increment

Use case: update database

Primary actor: admin
Goal in context: To manipulate existing data.
Preconditions:

1.System has been designed to have update database
2.System has interface to update database option
3.Admin must be logged in.

Scenario:

1. Admin select data to be updated
2. Admin add new data
3. Admin delete existing data

Triggers:

Data needs to be changed to various reason.

Exception:

1.Network failure.
2.System failure.
3.Database inconsistency
4.partial update

Priority: Essential, must be implemented
When available: First increment

Use case: Map

Primary actor: user
Goal in context: to get familiar with the university locations
Preconditions:

1.System has been designed to have map option
2.System has interface to map option
3.User must be logged in

Scenario:

1.User turns on location option in device
2.search location and get route to destination

Triggers: To need to go for official or personal purpose on unfamiliar place.
Exception:

1.Location not available.
2.Google map service not available.

Priority: Optional, should be implemented
When available: First increment

Use case: Information management

Primary actor: user
Goal in context: To get familiar with the university
Preconditions:

1.System has been designed to have information
2.System has interface to option for information
3.User must be logged in

Scenario:

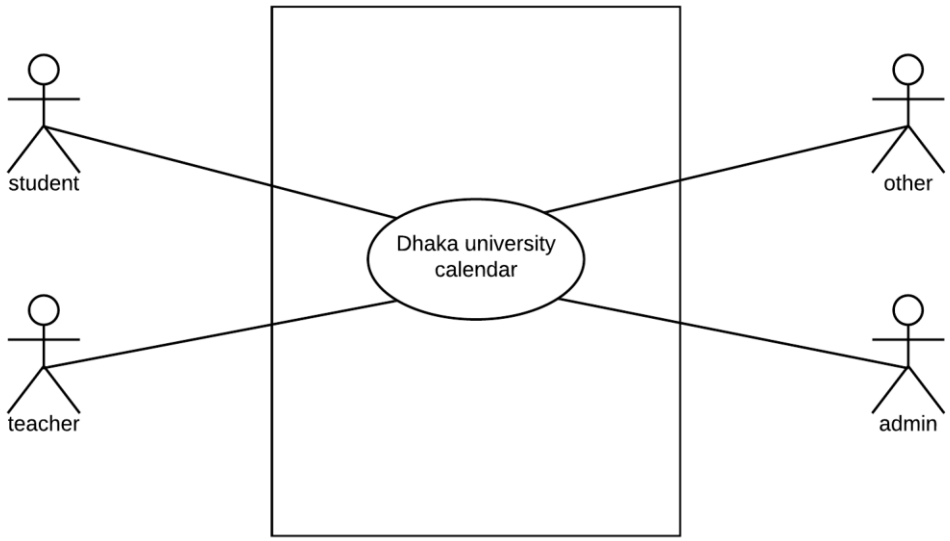
1.Users select different information option from the interface

Triggers: Students,teachers or other people to get familiar with the university
Exception:

1. System doesn't provide information.
2. Network error to fetch data from server

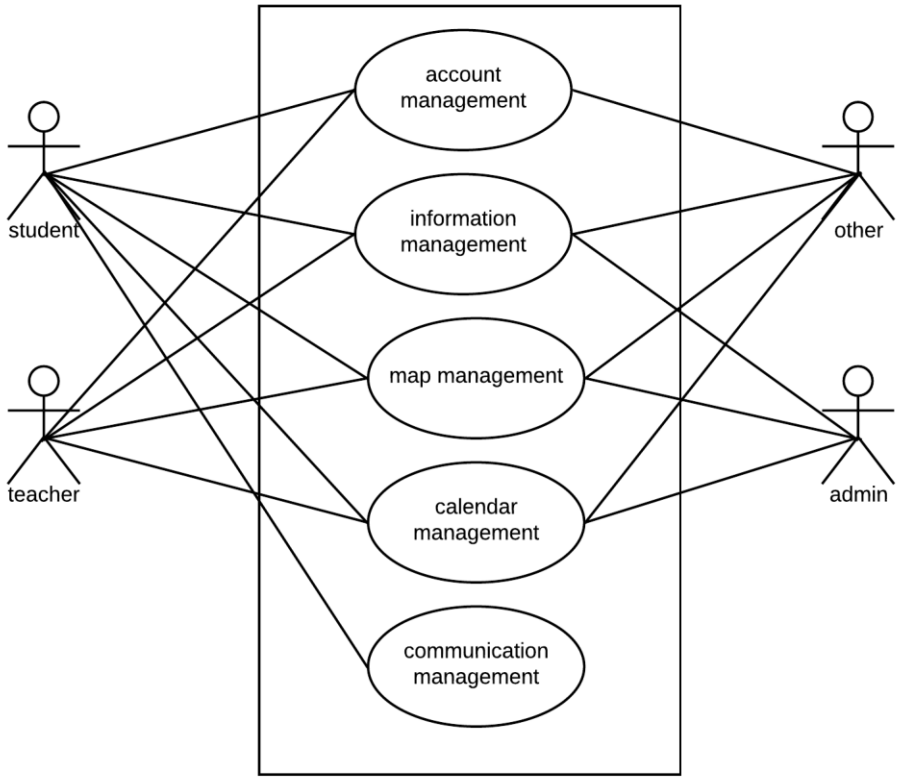
Priority: Essential, must be implemented
When available: First increment

4.4 Use Case Diagram



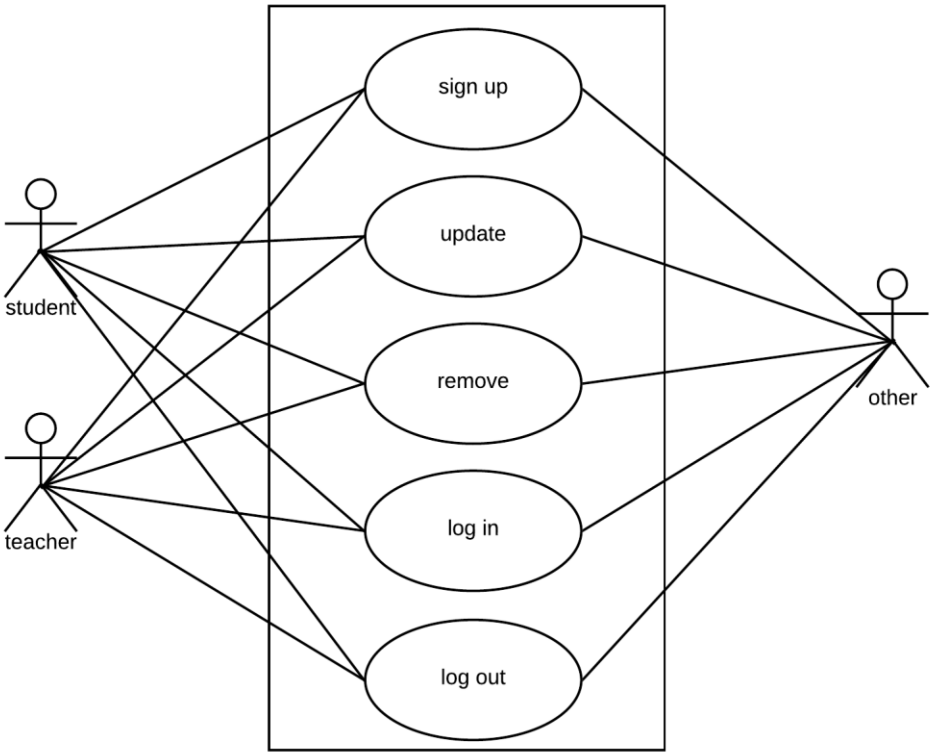
Level : 0

Figure : Level 0 for Dhaka University Calendar Management System



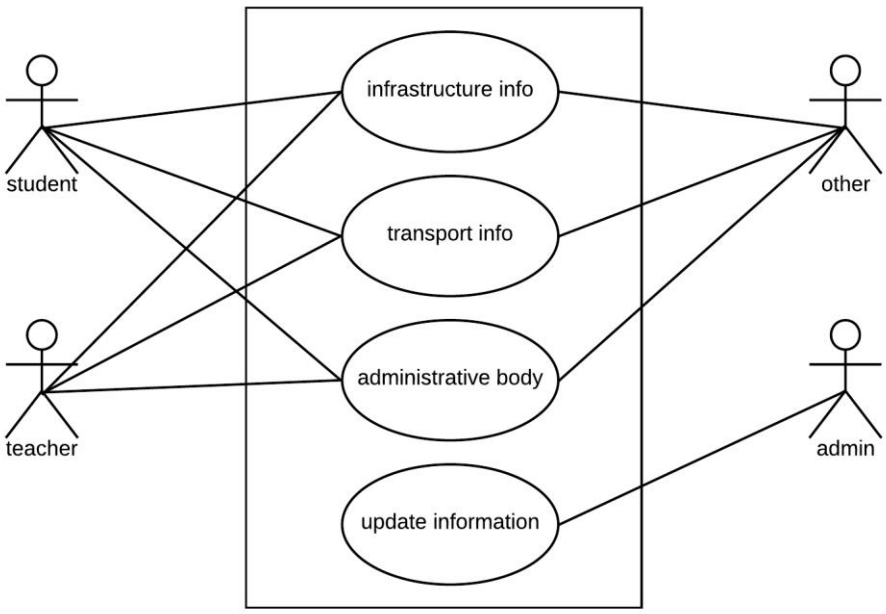
level: 1

Figure : Level 1 for Dhaka University Calendar Management System



level: 2.1

Figure : Level 2.1 for Dhaka University Calendar Management System



level: 2.2

Figure : Level 2.2 for Dhaka University Calendar Management System

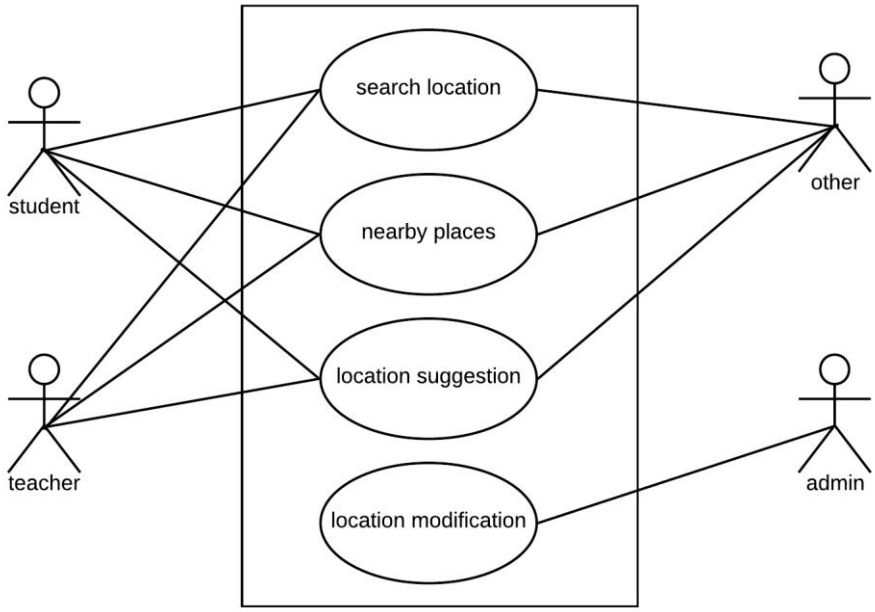


Figure : Level 2.3 for Dhaka University Calendar Management System

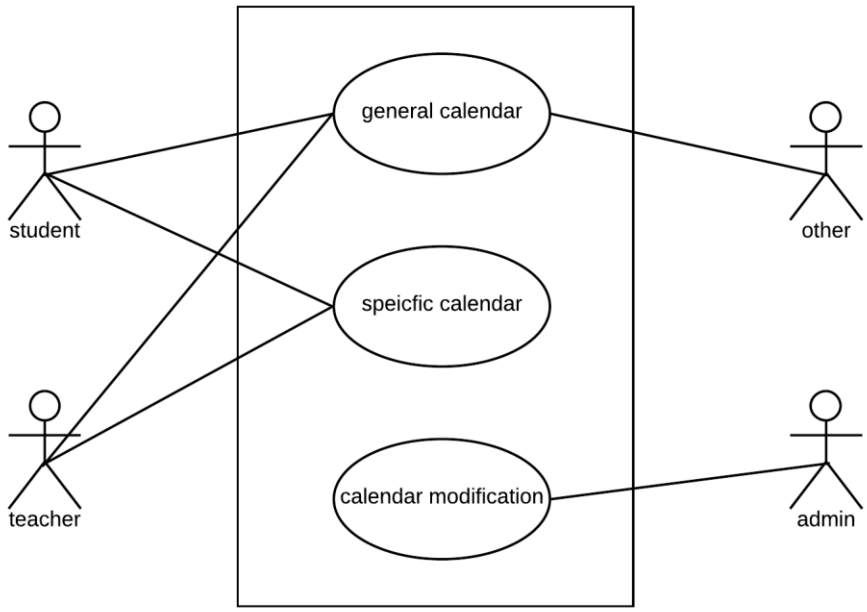
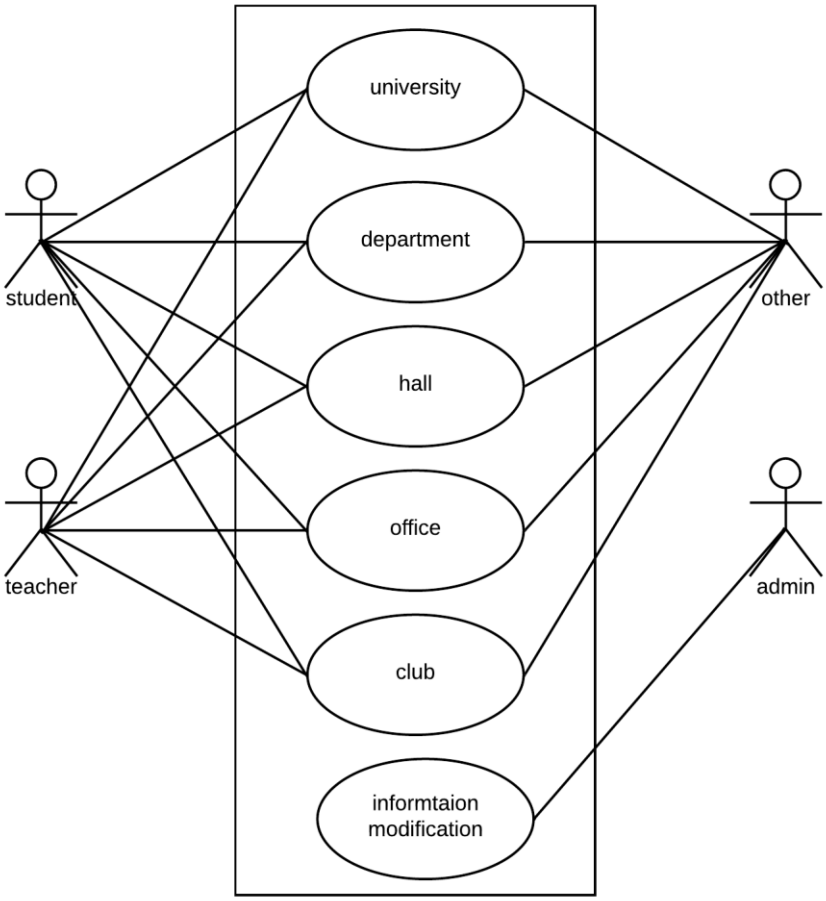


Figure : Level 2.4 for Dhaka University Calendar Management System



level: 3.2.1

Figure : Level 3.2.1 for Dhaka University Calendar Management System

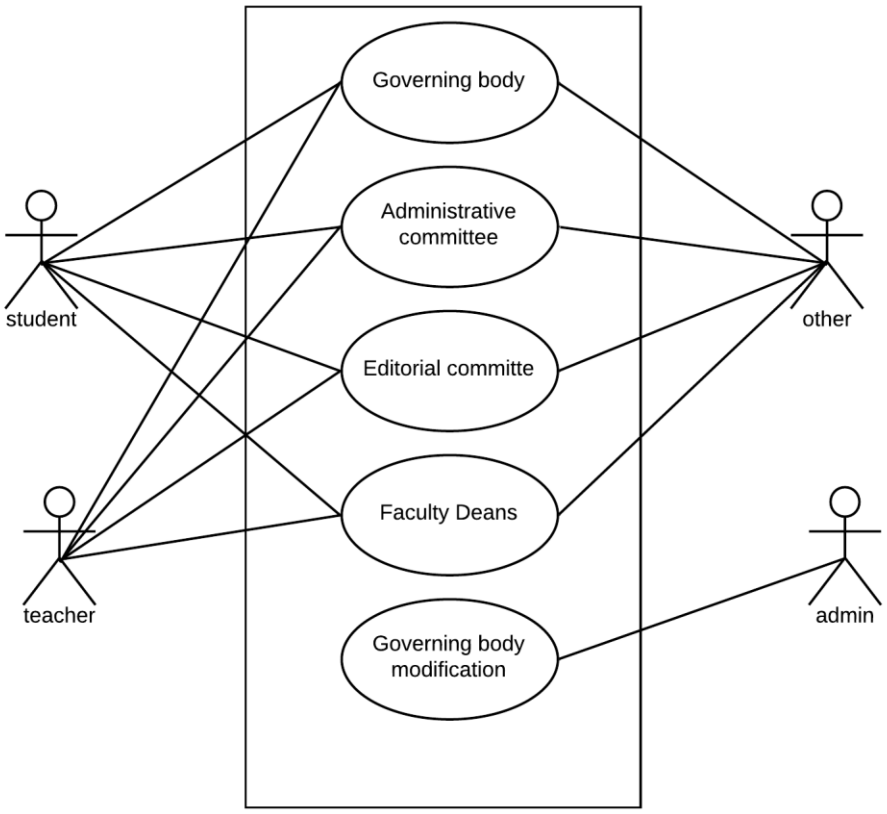


Figure : Level 3.2.3 for Dhaka University Calendar Management System

4.5 Activity Diagram and Swimlane Diagram of generated Use Cases:

Activity Diagram:

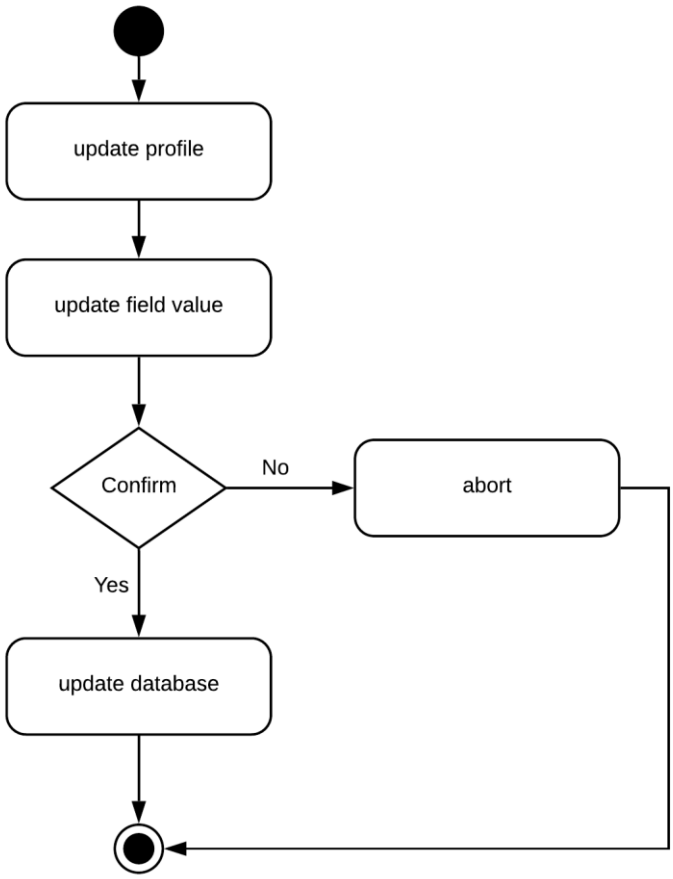


Figure: Account update

Figure : Activity Account Update

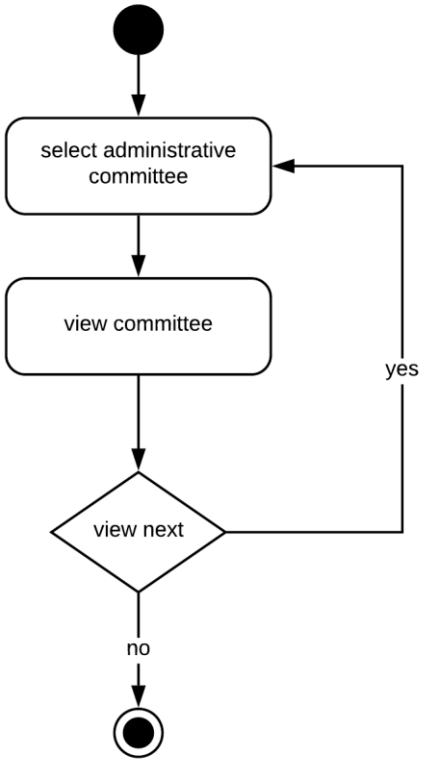


Figure: Administrative body

Figure : Activity Administrative Body

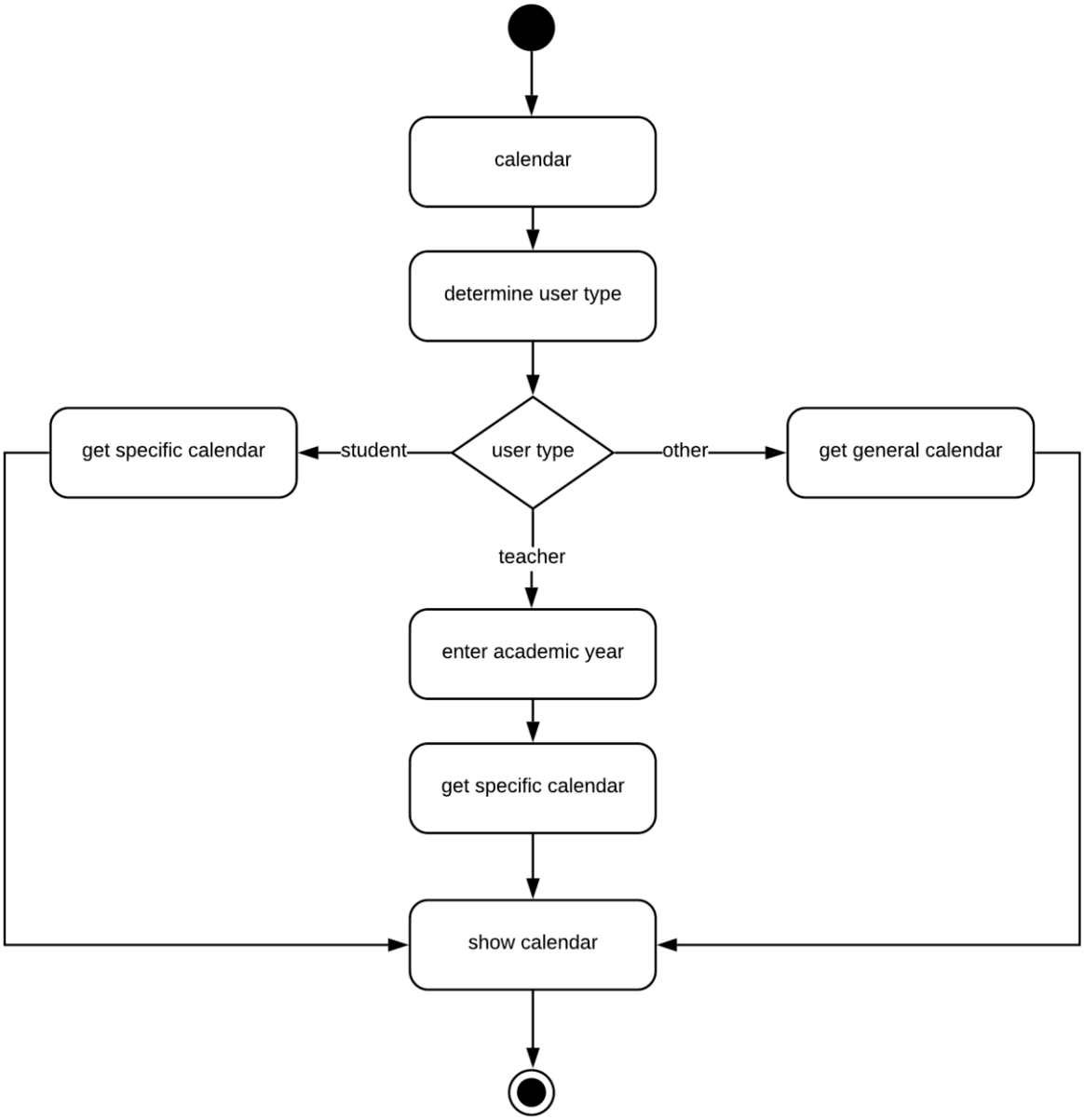


Figure: calendar view

Figure : Activity Calendar View

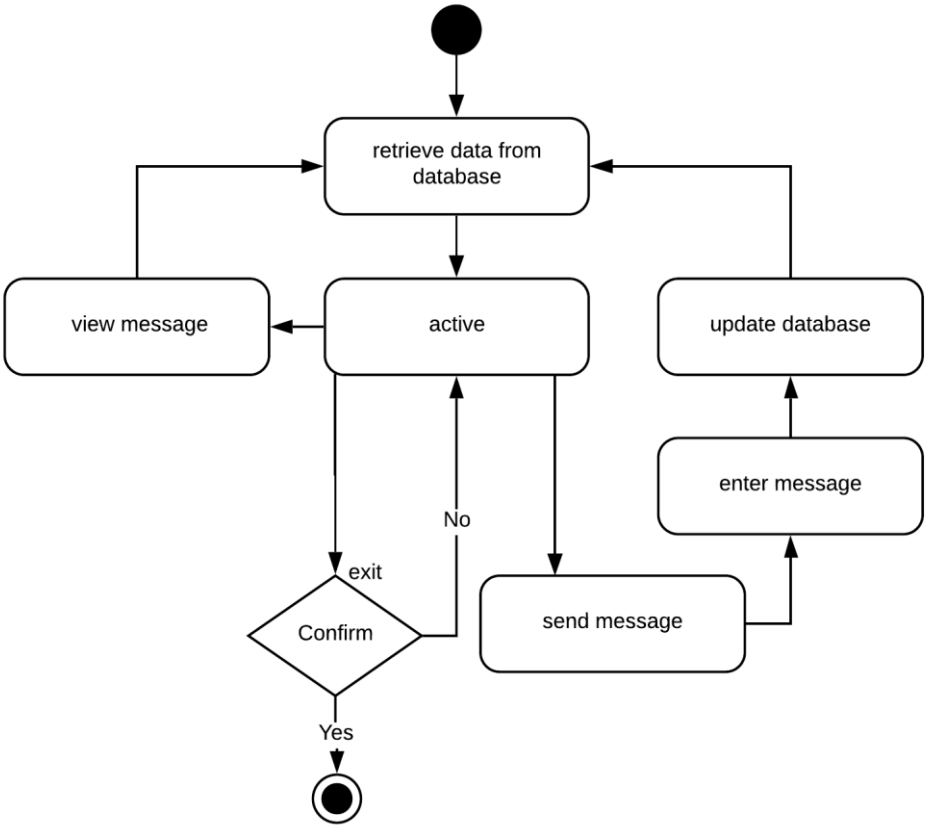


Figure: Communication

Figure : Activity Communication

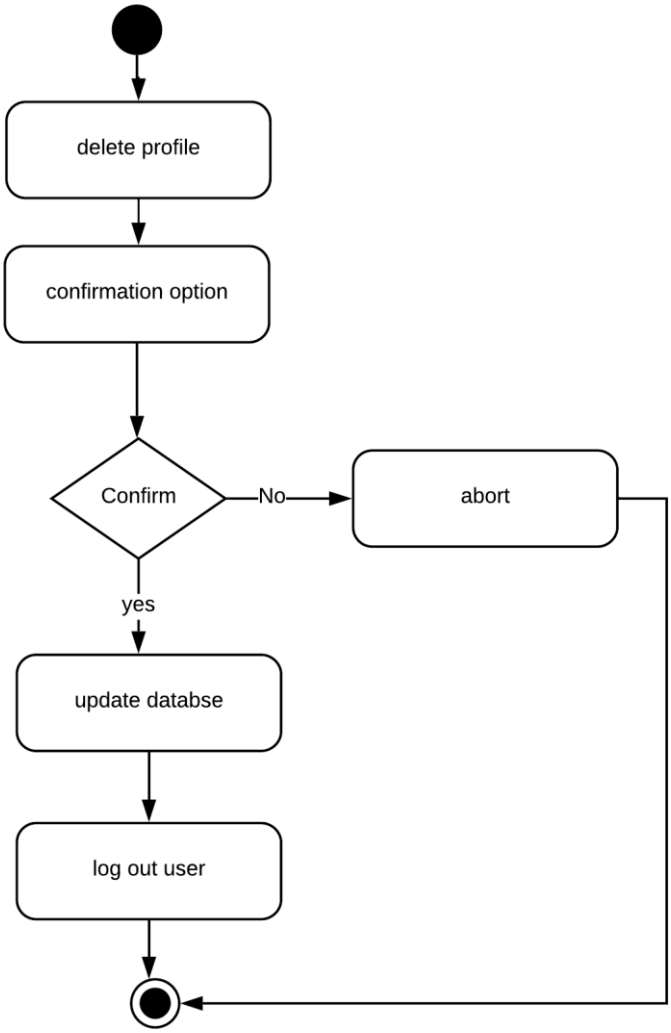


Figure: delete account

Figure : Activity Delete Account

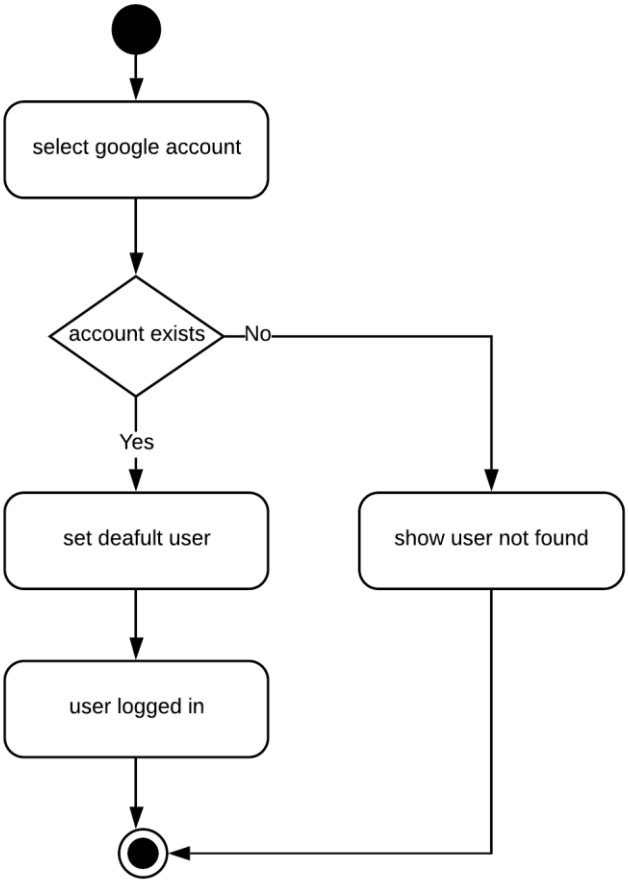


Figure: Log in

Figure : Activity Sign In

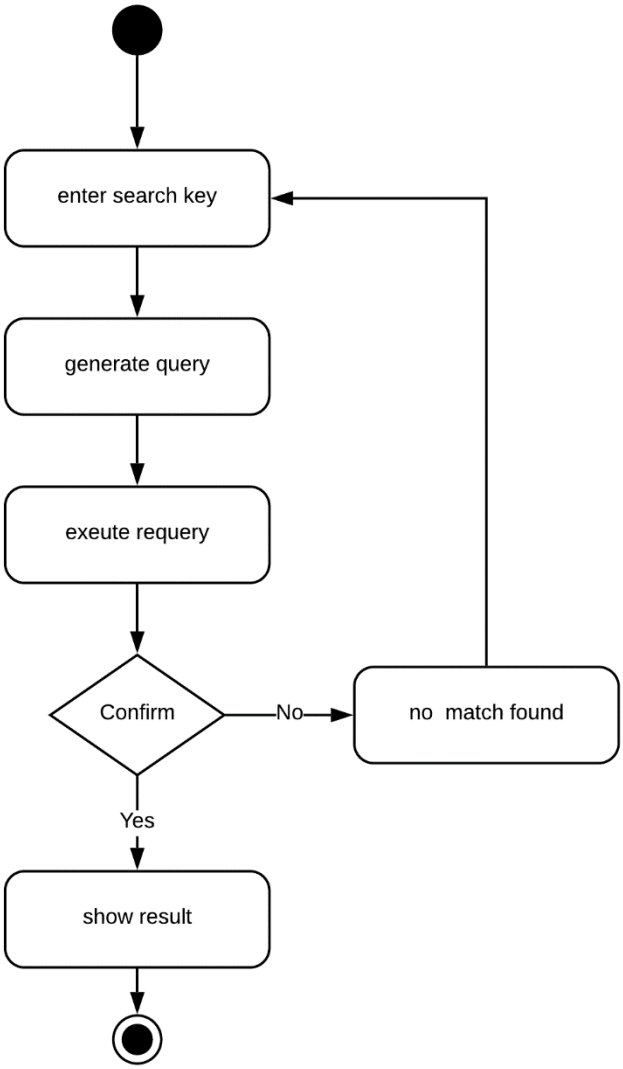


Figure: search

Figure : Activity Search

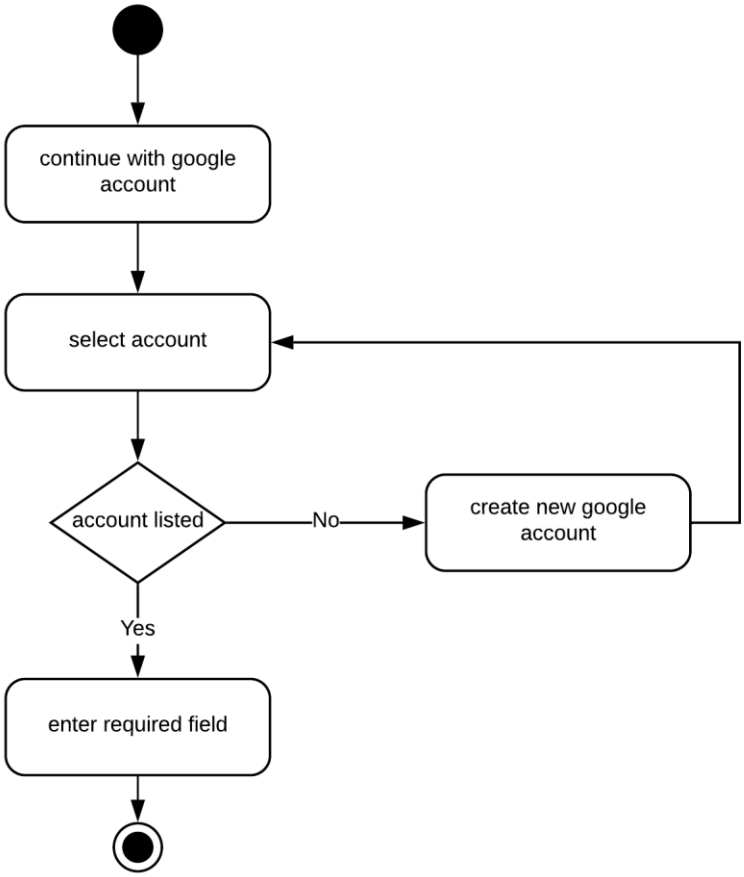


Figure: Sign up

Figure : Activity Sign UP

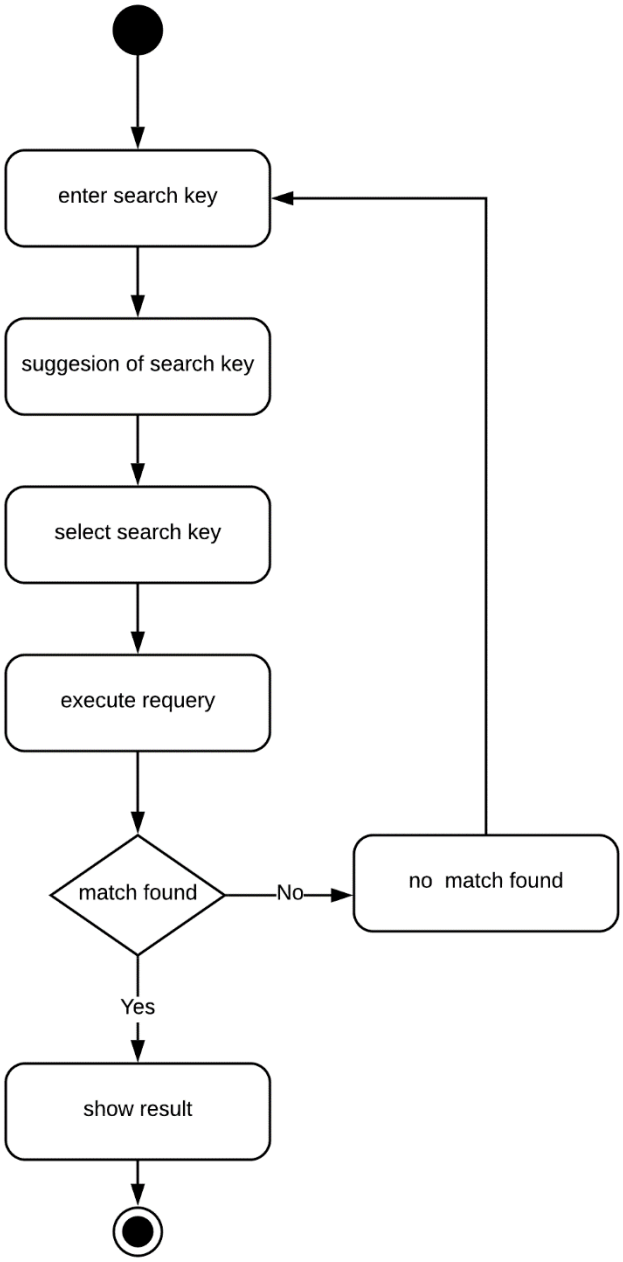


Figure: suggestion

Figure : Activity Suggestion

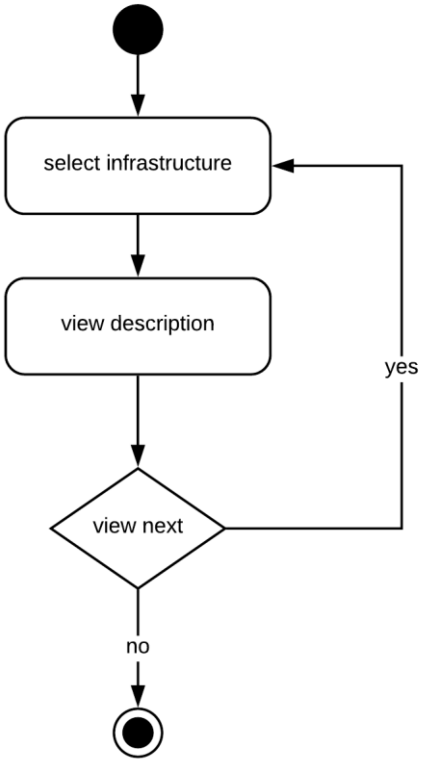


Figure : Activity Infrastructure

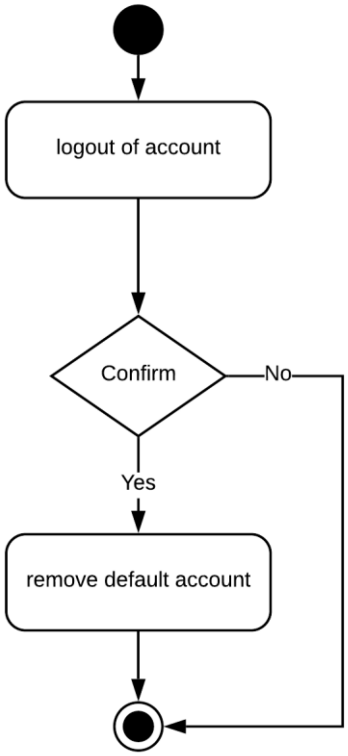


Figure: Log out

Figure : Activity Log out

Swimlane Diagram

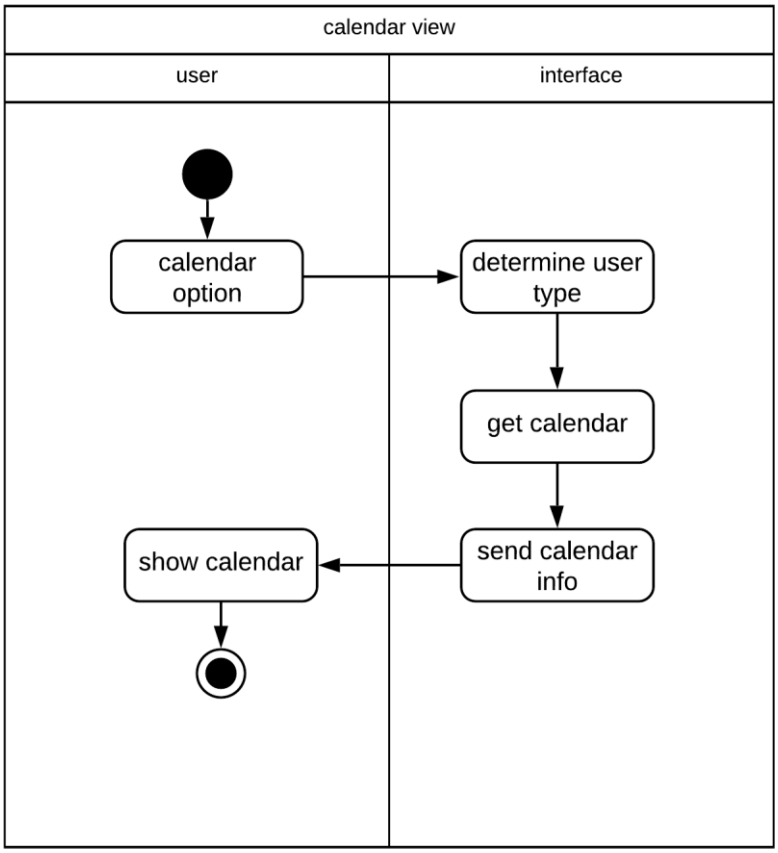


figure: Calendar view

Figure : Swimlane Calendar View

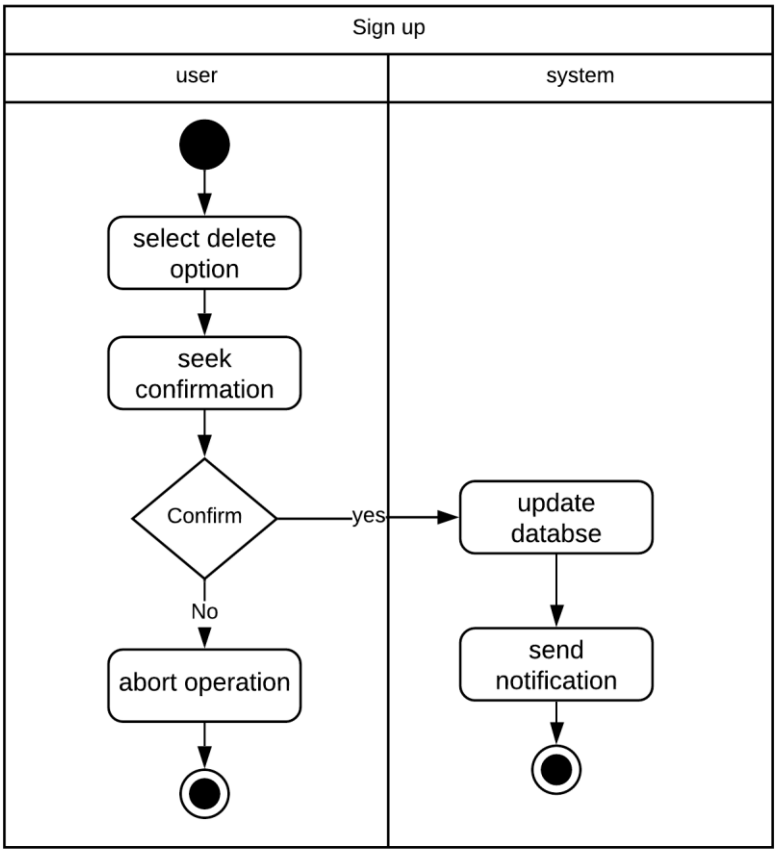


figure: delete profile

Figure : Swimlane Delete Profile

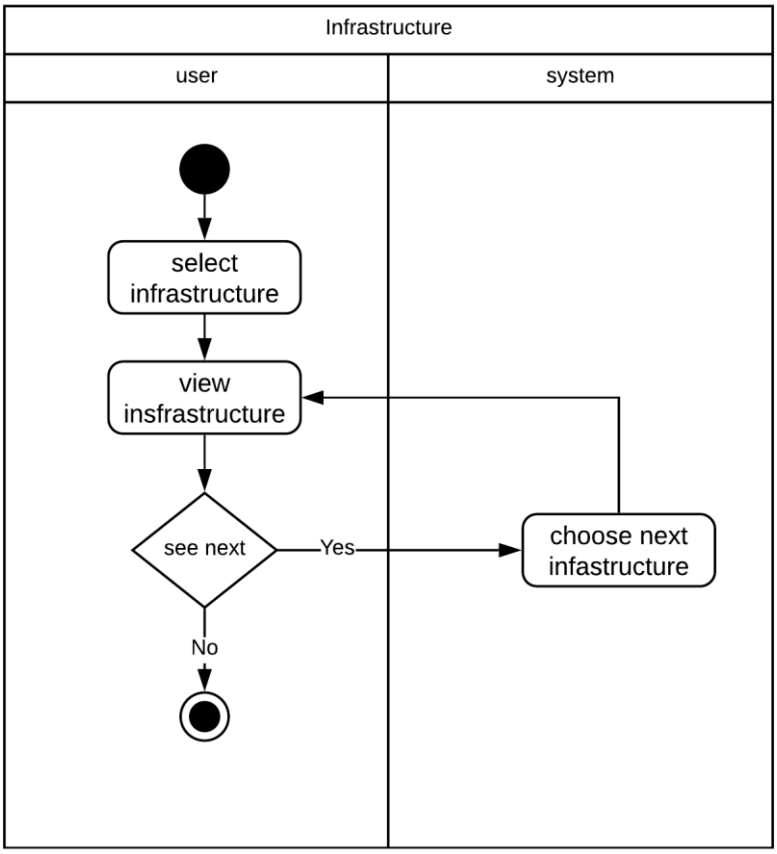


Figure: Infrastructure

Figure : Swimlane Infrastructure

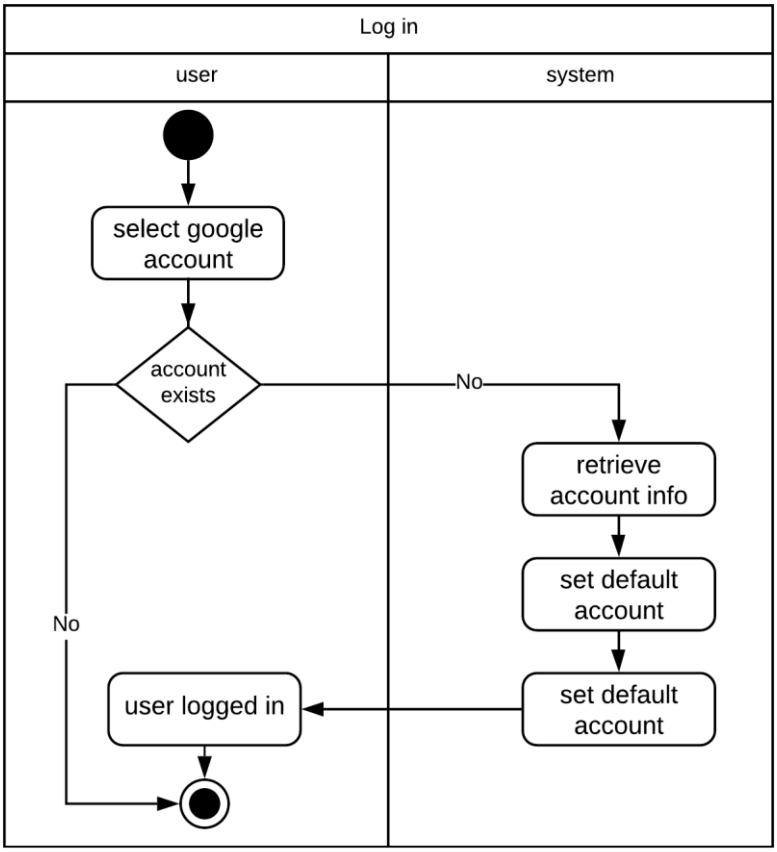


Figure: Log in

Figure : Swimlane Sign in

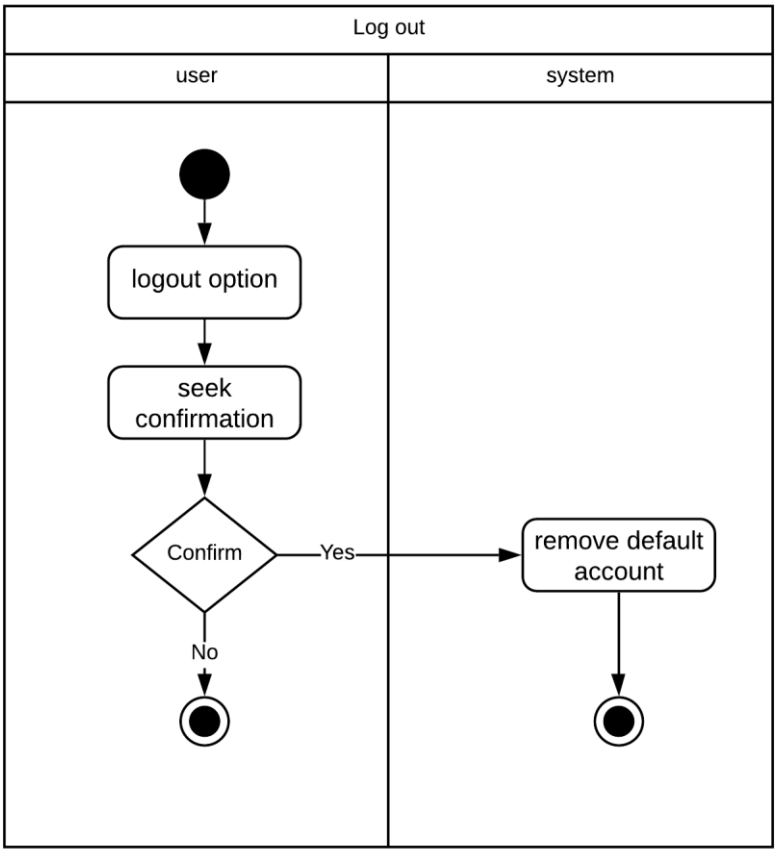
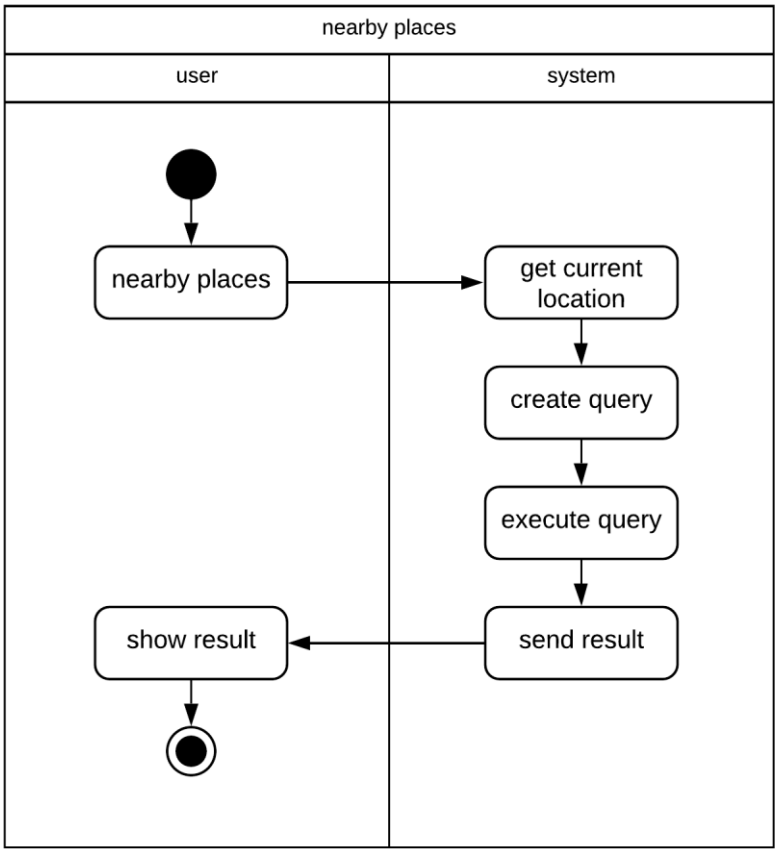


Figure: Logout

Figure : Swimlane Log Out



level: nearby places

Figure : Swimlane Exciting Places

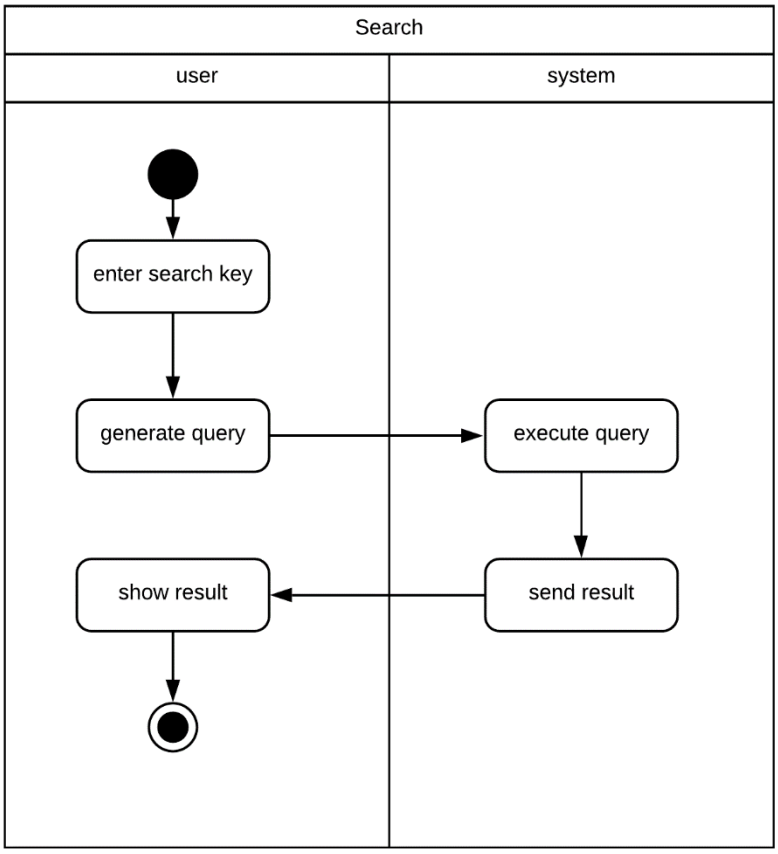


Figure: swimlane for search

Figure : Swimlane Search

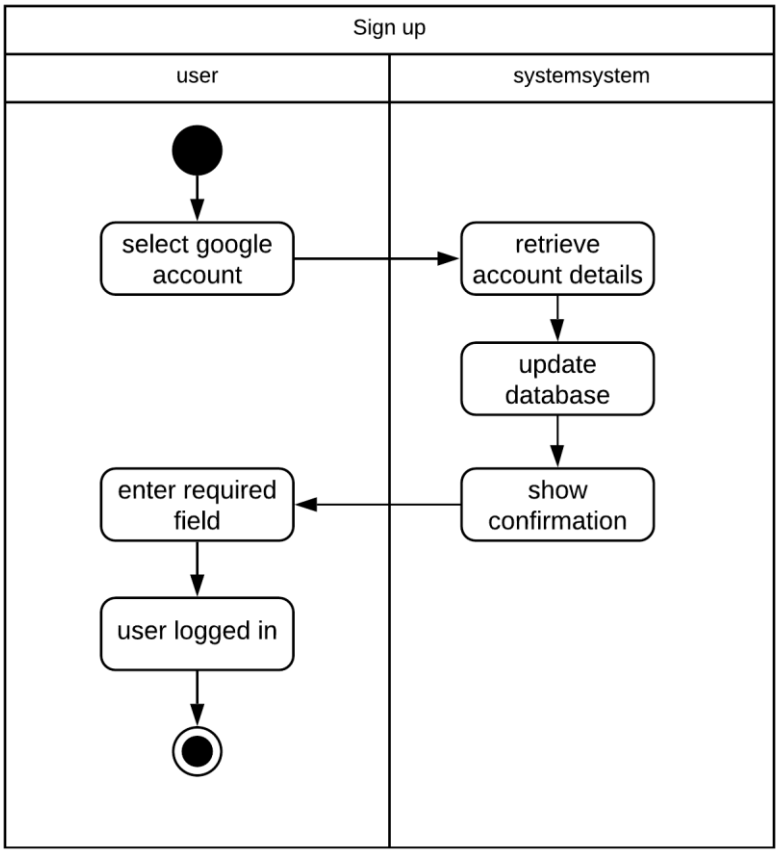


Figure: sign up

Figure : Swimlane Sign Up

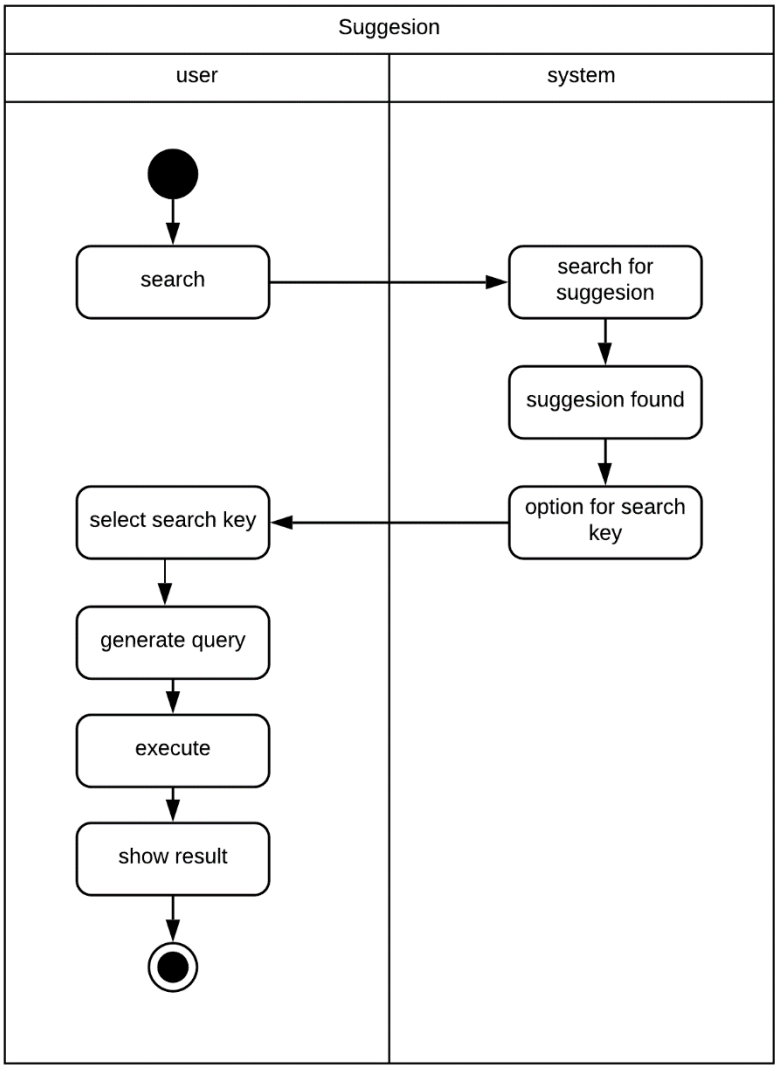


Figure: Swimlane for suggesion

Figure : Swimlane Suggestion

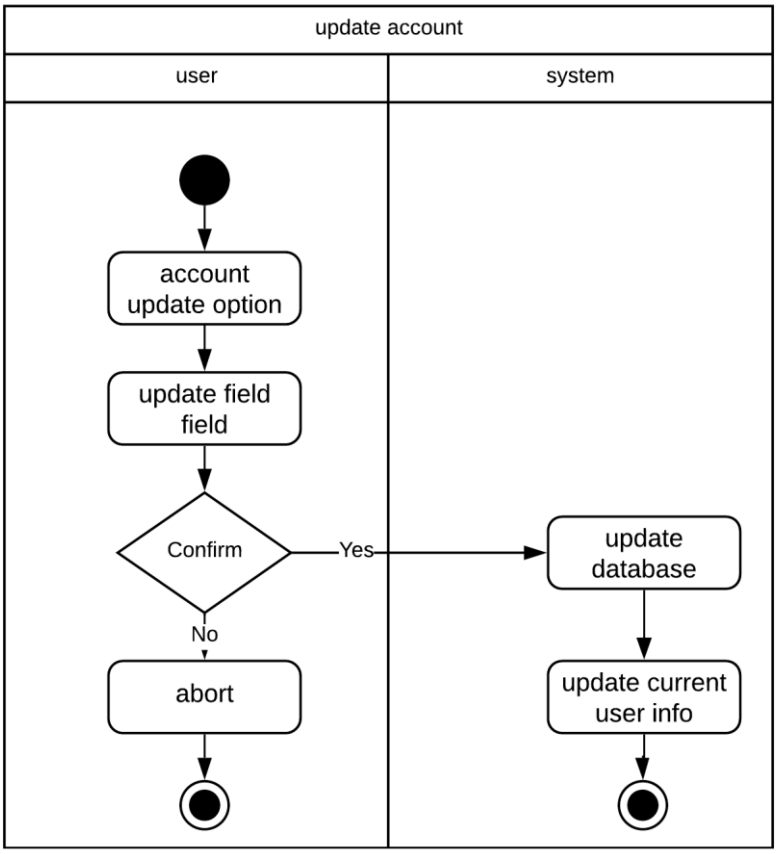


figure: update account

Figure : Swimlane Update Account

Chapter 5: Data Model

5.1 Data modeling 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) defines 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.

5.2 Data objects

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

Noun Identification

- All the nouns in the scenario were identified.

Table 2: Noun Identification for data modelling

SL NO	NOUN	P/S	Attributes
1	Account	S	6
2	User	S	6,11
3	Application	P	71,72,74,75,76
4	Google play store	S	-
5	Dhaka University	P	13, 37-43
6	Google account	P	17,18
7	Profile	S	8,9,10,11,105
8	User Name	S	-
9	Academic Year	S	-
10	Email	S	6

11	Type	S	-
12	delete	S	14
13	Department	S	87,96,100,101,102,104,105
14	specified	S	-
15	Sign up	S	-
16	Sign in	S	-
17	Authentication	S	14 , 18
18	Firebase Auth 2	P	-
19	Information	P	-
20	Log out	P	-
21	dropdown	P	-
22	Menu	S	-
23	Admin	P	8,10,35,115
24	features	P	-
25	year	P	26
26	Keyword	S	-
27	exciting	P	-
28	calendar	S	60,61
29	group chat	P	30,54,64
30	messenger	P	83
31	activities	P	-
32	update	P	-
33	Confirmation	S	-
34	separate	S	-
35	website	P	-
36	wish	S	-
37	DU brief history	P	-
38	achievement	S	-
39	Administrative Body	S	88,89,90
40	Halls	S	87,100,101,102
41	Club	S	87,100,101,102
42	society	S	-
43	places	P	-
44	view	S	-
45	search	S	26,80
46	Transport	S	47,48,51,52
47	bus	P	113,48,51,52,114
48	stoppage	P	-
49	Map	P	45,51,52,103

50	current position	S	-
51	destination	P	-
52	route	P	-
53	offices	P	-
54	Communication	P	29,30,64
55	first	P	-
56	person	P	-
57	Database	S	-
58	private chat	S	-
59	option	S	-
60	holiday	P	-
61	events	P	-
62	Notification	S	-
63	Contact1	S	-
64	Chat	S	-
65	academic	S	-
66	web interface	S	-
67	server	S	-
68	data transparency	S	-
69	service	S	-
70	security	S	-
71	Homepage	S	-
72	icons	S	-
73	top	S	-
74	settings	S	-
75	language	S	-
76	layout	S	-
77	tree based	S	-
78	design	S	77,76
79	structure	S	-
80	database query	S	-
81	two	S	-
82	restriction	P	-
83	Message	P	-
84	link	P	-
85	right	P	-
86	Status	S	-
87	Student	S	8,9,10,11,105
88	picture	P	-
89	name	P	-
90	rank	P	-

91	Administrative committee	S	11,89,90
92	Editorial committee	S	11,89,90
93	Faculty Deans	S	11,94,95
94	faculty name	P	-
95	Dean	P	-
96	Department ID	P	29,30,64
97	Institute ID	P	-
98	Hall ID	P	-
99	Office ID	S	-
100	establishment year	S	-
101	Description	S	-
102	Location	P	-
103	Area	P	-
104	Teacher	S	8,10,11,105
105	Department Name	P	-
106	Institute	S	87,97,100,101,102,104,107
107	Institute Name	P	-
108	Specific Calendar	S	60,61,109,110
109	Date	P	-
110	Academic Schedule	P	-
111	Other user	S	8,10,11
112	Infrastructure	S	100,101,102
113	Bus ID	P	-
114	Bus name	P	-
115	password	P	-

Potential Data Objects:

- Account : 6
- User : 6,7,11
- Application : 71,72,74,75,76
- Dhaka University : 13, 37-43
- Google account : 17,18
- Profile : 8,9,10,11,105
- Email : 6
- Department : 87,96,100,101,102,104,105
- Authentication : 18
- Admin : 8,10,32,35,115
- General Calendar : 60,61,109
- group chat : 30,54,64
- Administrative Body : 88,89,90
- Hall : 87,97,100,101,102
- Club : 87,98,100,101,102
- Search : 26,80
- Transport : 47,48,51,52
- Map : 51,52,103
- Office : 99,100,101,102
- Communication : 29,30,64
- Design : 77,76
- Student : 8,9,10,11,12,13
- Administrative committee : 11,89,90
- Editorial committee : 11,89,90
- Faculty Dean : 11,94,95
- Teacher : 8,9,10,11,105
- Department : 87,96,100,101,102,104,107
- Institute : 87,97,100,101,102,104,107
- Specific Calendar : 60,61,109,110
- Other User : 8,10,11
- Infrastructure : 100,101,102
- Bus : 48,51,52,113

Analysis for finalizing Data objects

- Account , user , application , profile , email are all part of other user object . So , they are not considered.
- Authentication , Search , communication are attributes of other data object. So they are not considerable .
- Department , Office , Hall , Club are all part of Infrastructure . So infrastructure is the parent class.
- Administrative body , Administrative committee , Editorial committee , Faculty Deans are all part of Administrative committee list .
- All other data objects can be used as data objects as they have enough importance in the system.

Final Data objects

Table 3: Final Data Objects

1	User : User ID, Email , Type.
2	Student: User Name, Academic Year, Email, Department Name, Type
3	Teacher: User Name, Email, Department Name , Type
4	Other User : User Name, Email, Type
5	Infrastructure : establishment year, Description, Location
6.	Department : Department ID, establishment year, Description, Location , Student id , Teacher id , Department name
7.	Institute : Institute ID, establishment year, Description, Location , Student id , Teacher id , Institute name
8.	Office : Office ID , establishment year, Description, Location
9.	Hall : Hall ID , Student Id ,establishment year, Description, Location
10.	Club : Club ID , Student Id , establishment year, Description, Location
11.	Administrative Body : Type, Name
12 .	Administrative committee : Type, Name , Rank

13 .	Map : location ID , destination , Area
14 .	Transport : Transport ID , Type
15 .	Bus : Bus ID , Bus Name , destination , Stoppage
16.	General Calendar : Holiday, Events , Date
17 .	Specific Calendar : Holiday, Events , Date , Academic Schedule
18 .	Group Chat : Messenger , Communication ,Chat
19.	Admin : Admin ID, User name , Email , password , website
20	Editorial committee : Type, Name , Rank
21	Faculty Dean List : Type, Name , Faculty name

Entity Relationship Diagram

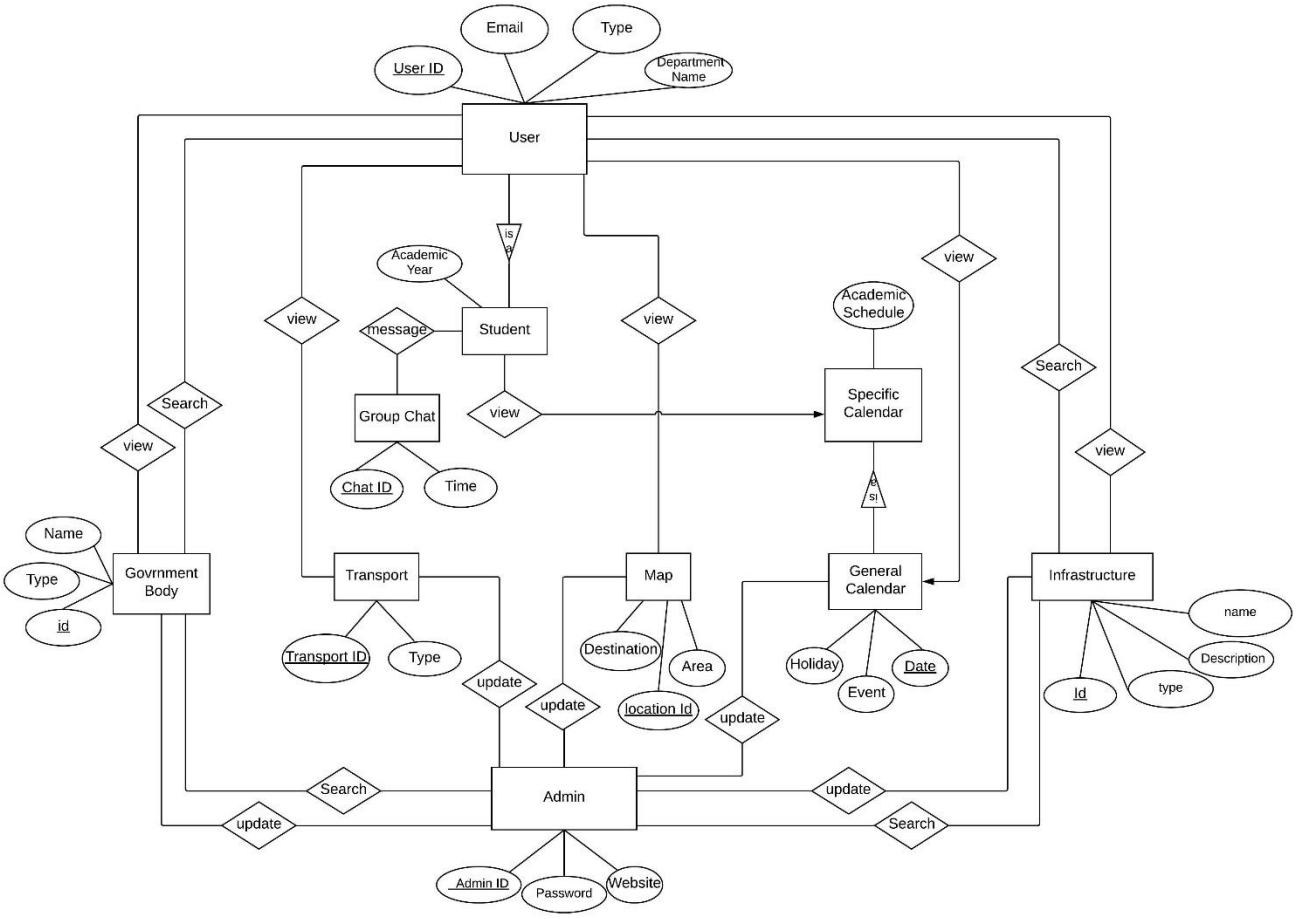


Figure : Entity Relationship Diagram for Dhaka University Calendar Management Application

Relational Schema

Table : Schema for User

User		
Attribute	**Type**	**Size**
User_id	Number	30
Email	Varchar2	40
Type	Varchar2	20

Table : Schema for Student

Student		
Attribute	**Type**	**Size**
Student_id	Number	30
Academic Year	Varchar2	20
Email	Varchar2	40
User Name	Varchar2	20
Department Name	Varchar2	20
Type	Varchar2	10

Table : Schema for Teacher

Teacher		
Attribute	**Type**	**Size**
Teacher_id	Number	30
Email	Varchar2	40
Department Name	Varchar2	30
Type	Varchar2	20

Table : Schema for Other User

Other User		
Attribute	**Type**	**Size**
User_id	Number	30
Email	Varchaer2	30
Type	Varchar2	20

Table : Schema for Infrastructure

Infrastructure		
Attribute	**Type**	**Size**
Infrastructure ID	Number	30
Established Year	Number	30
Description	Varchar2	3000
Location	Number , Number	32,32

Table : Schema for Department

Department		
Attribute	**Type**	**Size**
Department ID	Number	30
Student ID	Number	30
Teacher ID	Number	30
Established Year	Number	30
Description	Varchar2	3000
Department Name	Varchar2	30
Location	Number , Number	32,32

Table : Schema for Department

Department		
Attribute	**Type**	**Size**
Department ID	Number	30
Student ID	Number	30
Teacher ID	Number	30
Established Year	Number	30
Description	Varchar2	3000
Department Name	Varchar2	30
Location	Number , Number	32,32

Table : Schema for Hall

Hall		
Attribute	**Type**	**Size**
Hall ID	Number	30
Student ID	Number	30
Established Year	Number	30
Description	Varchar2	3000
Location	Number , Number	32,32

Table : Schema for Club

Club		
Attribute	**Type**	**Size**
Club ID	Number	30
Student ID	Number	30
Established Year	Number	30
Description	Varchar2	3000
Location	Number , Number	32,32

Table : Schema for Office

Office		
Attribute	**Type**	**Size**
Office ID	Number	30
Established Year	Number	30
Description	Varchar2	3000
Location	Number , Number	32,32

Table : Schema for Person View

Person View		
Attribute	**Type**	**Size**
User ID	Number	30
User Name	Varchar2	30
Type	Varchar2	30

Table : Schema for Administrative Committee

Administrative Committee		
Attribute	**Type**	**Size**
User ID	Number	30
User Name	Varchar2	30
Rank	Varchar2	30
Type	Varchar2	30

Table : Schema for Editorial Committee

Editorial Committee		
Attribute	**Type**	**Size**
User ID	Number	30
User Name	Varchar2	30
Rank	Varchar2	30
Type	Varchar2	30

Table : Schema for Faculty Deans

Faculty Dean		
Attribute	**Type**	**Size**
User ID	Number	30
User Name	Varchar2	30
Faculty Name	Varchar2	30
Type	Varchar2	30

Table : Schema for Map

Map		
Attribute	**Type**	**Size**
Location ID	Number , Number	30 , 30
Destination	Float , Float	40,40
Area	Varchar2	50

Table : Schema for Transport

Transport		
Attribute	**Type**	**Size**
Transport ID	Number	30
Type	Varchar2	30

Table : Schema for Bus

Bus		
Attribute	**Type**	**Size**
Bus ID	Number	30
Bus name	Varchar2	30
Destination	Varchar2	50
Stoppage	Varchar2	50

Table : Schema for General Calendar

General Calendar		
Attribute	**Type**	**Size**
Date	Varchar2	30
Holiday	Varchar2	100
Event	Varchar2	100

Table : Schema for Special Calendar

Special Calendar		
Attribute	**Type**	**Size**
Date	Varchar2	30
Holiday	Varchar2	100
Academic Schedule	Varhcar2	100
Event	Varchar2	100

Table : Schema for search

Search		
Attribute	**Type**	**Size**
Source Id	Varchar2	30
Destination Id	Varchar2	30
Search key	Varchar2	30

Chapter 6: Class Based Modeling

6.1 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.

6.2 General Classification

To identify the potential classes, nouns were selected from the solution space of the story. These were then characterized in seven general classifications. The seven general characteristics are as follows:

1. External entities
2. Things
3. Events
4. Roles
5. Organizational units
6. Places
7. Structures

Following are the specifications of the nouns according to the general classifications

Table : General Classification of Noun

SL NO	NOUN	P/S	General Classification
1	Account	S	-
2	User	S	4,5
3	Application	P	-
4	Google play store	S	-
5	Dhaka University	P	5,7
6	Google account	P	-
7	Profile	S	-
8	User Name	S	-
9	Academic Year	S	-
10	Email	S	-

11	Type	S	-
12	delete	S	3
13	Department	S	5,7
14	specified	S	-
15	Sign up	S	3
16	Sign in	S	3
17	Authentication	S	-
18	Firebase Auth 2	P	1
19	Information	P	-
20	Log out	P	3
21	dropdown	P	-
22	Menu	S	-
23	Admin	P	4,5
24	features	P	-
25	year	P	-
26	Keyword	S	-
27	exciting	P	-
28	calendar	S	3,5
29	group chat	P	5
30	messenger	P	-
31	activities	P	2
32	update	P	3
33	Confirmation	S	-
34	separate	S	-
35	website	P	2
36	wish	S	-
37	DU brief history	P	-
38	achievement	S	-
39	Administrative Body	S	4,5
40	Halls	S	5,7
41	Club	S	5,7
42	society	S	-
43	places	P	-
44	view	S	-
45	search	S	3
46	Transport	S	2,7
47	bus	P	2,7
48	stoppage	P	-
49	Map	P	3

50	current position	S	-
51	destination	P	-
52	route	P	-
53	offices	P	7
54	Communication	P	-
55	first	P	-
56	person	P	5
57	Database	S	5
58	private chat	S	-
59	option	S	-
60	holiday	P	3
61	Events	P	3
62	Notification	S	3
63	Group	S	-
64	Chat	S	3
65	academic	S	7
66	web interface	S	5
67	server	S	2
68	data transparency	S	-
69	service	S	-
70	security	S	-
71	Homepage	S	-
72	icons	S	-
73	top	S	-
74	settings	S	-
75	language	S	-
76	layout	S	-
77	tree based	S	-
78	design	S	-
79	structure	S	-
80	database query	S	3,7
81	two	S	-
82	restriction	P	-
83	Message	P	3
84	link	P	-
85	System	P	4,5
86	Status	S	-
87	Student	S	4,5,7
88	picture	P	-
89	name	P	-
90	rank	P	-

91	Administrative committee	S	4,7
92	Editorial committee	S	4,7
93	Faculty Deans	S	4,7
94	faculty name	P	-
95	Dean	P	-
96	Department ID	P	-
97	Institute ID	P	-
98	Hall ID	P	-
99	Office ID	S	-
100	establishment year	S	-
101	Description	S	-
102	Location	P	-
103	Area	P	-
104	Teacher	S	4,5,7
105	Department Name	P	-
106	Institute	S	5,7
107	Institute Name	P	-
108	Specific Calendar	S	4,5,
109	Date	P	-
110	Academic Schedule	P	5
111	Other user	S	4
112	Infrastructure	S	7
113	Bus ID	P	-
114	Bus name	P	-
115	password	P	-

6.3 Selection Criteria

The potential classes were then selected as classes by six Selection Criteria. A potential class becomes a class when it fulfills all six characteristics.

1. Retained Information
2. Needed Services
3. Multiple Attributes
4. Common attributes
5. Common operations
6. Essential requirements

Table : Selection Criteria of Potential Classes

SL NO	NOUN	P/S	Selection Criteria
1	User	S	1,2,3,4,5
2	Dhaka University	P	-
3	Delete	S	-
4	Department	S	4
5	Dhaka University	P	-
6	Sign up	S	-
7	Sign in	S	-
8	Infrastructure	S	1,3,4,5
9	Log out	P	-
10	Admin	P	1,2,3,4,5,6
11	General Calendar	S	1,2,4
12	Group Chat	P	2,4,5
13	activities	P	-
14	update	P	-
15	website	P	-
16	Administrative Body	S	1,2,4
17	Halls	S	4
18	Club	S	4
19	search	S	2
20	Transport	S	2,3,4,6
21	Map	P	1,2,4,5
22	bus	P	-

23	Offices	P	4
24	person	P	-
25	Database	S	1,2,6
26	Holiday	P	-
27	Events	P	-
28	Notification	S	-
29	Chat	S	-
30	academic	S	-
31	web interface	S	-
32	server	S	-
33	database query	S	-
34	Message	P	2
35	Student	S	1,2,3,4,5
36	Editorial committee	S	2
37	Faculty Deans	S	-
38	Teacher	S	1,2,3,4
39	Institute	S	4
40	Specific Calendar	S	1,2,4
41	Academic Schedule	P	-
42	Other user	S	1,2,3,4
43	System	S	2,3,4,6

6.4 Potential Classes

From above table, we have taken all the noun who passed three or more accepted criteria. So these are the candidate classes who are selected primarily:

- User
- Student
- Teacher
- Other User
- Admin
- Infrastructure
- Administrative Body
- Transport
- Map
- Database
- Group Chat
- General Calendar
- Specific Calendar
- System

6.5 Associate Noun and Verb Identification

The nouns and the verbs associated with the potential classes are identified to find out the attributes and methods of each class.

Table 18: Associate Noun and Verbs

No	Potential Class	Noun	Verb
1	User	User ID, Email , Type	Search , Sign In , Sign Out , view information , access , update account , delete Account , modify account ,access general calendar , view map , view transport
2	Student	User Name, Academic Year, Email, Department Name, Type	Search , Sign In , Sign Out , view information, access , update account , delete Account , modify account , complete profile , Send message to group chat , access general calendar , view map , view transport ,access specific calendar
3	Teacher	User Name, Email, Department Name , Type	Search , Sign In , Sign Out , view information , access , update account , delete Account , modify account ,access general calendar , view map , view transport
4	Other User	User Name, Email, Type	Search , Sign In , Sign Out , view information , access , update account , delete Account , modify account ,access general calendar , view map , view transport
5	Admin	Admin ID, User name , Email , password , website	Update information , delete information , modify information , Update map , delete map information , modify map information, Update transport , delete transport information , Update or delete calendar information , maintain server and database , maintain security
6	Infrastructure	establishment year, Description, Location	Store information ,knowledge , view location , view description
7	Administrative Body	Type, Name	View information about Administrative body

8	Transport	Transport ID , Type, destination, Stoppage	view the schedule of buses , View stoppage and route of the transport
9	Map	location ID , destination , Area	View current position , search , get route , view exciting places in map
10	Database	Information , store, Update , retrieve	Update , retrieve , store information
11	Group Chat	Messenger , Communication, Chat	Message other user, leave group , join group ,chat
12	General Calendar	Holiday, Events , Date	View all holiday ,events of Dhaka University
13	Specific Calendar	Holiday, Events , Date , Academic Schedule	View all holiday ,events of Dhaka University, View all academic activities
14	System	Send , retrieve , generate	Generate database Query , send formatted information to user , Communicate with database

6.6 Analysis of Potential Classes

- Teacher class , other user class and user class all have the same functionalities . So we will only consider user class .
- Student class has some similarities with user class , So , student extends user class.
- Specific calendar class has some similarities with general calendar class , So , Specific calendar extends general calendar class.

6.7 Final Classes

From above analysis, our final classes are:

1. User
    a. Student
2. Admin
3. Infrastructure
4. Administrative Body
5. Transport
6. Map
7. Group Chat
8. General Calendar
    a. Specific Calendar
9. Database
10. System

6.8 Class Cards

Table 24: Class Card of User

User	
Attributes	**Methods**
User ID User name Email Type	• viewMap() • viewTransport() • viewGeneralCalendar() • viewInformation() • viewaccount() • updateAccount() • search() • signIn() • signOut() • deleteAccount() • getEmail() • getUserID() • getType() • setUserID() • setEmail() • setType()
Responsibilities	**Collaborator**
Search Administrative Body	Administrative Body
Search Infrastructure	Infrastructure
Get Location	Map
Access Calendar	Calendar

Table 24: Class Card of Student

Student	
Attributes	**Methods**
User ID User Name Email Type Department Name Academic Year	• viewMap() • viewTransport() • viewGeneralCalendar() • viewInformation() • viewaccount() • updateAccount() • search() • signIn() • signOut() • deleteAccount() • getEmail() • getUserID() • getType() • setUserID() • setEmail() • setType() • getDepartmentName() • getAcademicYear() • joinGroupChat() • leaveGroupChat()
Responsibilities	**Collaborator**
Search Administrative Body	Administrative Body
Search Infrastructure	Infrastructure
Get Location	Map
Access Calendar	Calendar
Access Specific Calendar	Specific Calendar
Join Group Chat	Group Chat

Table 24: Class Card of Admin

Admin	
Attributes	**Methods**
Admin ID User name Email Password Website	• updateMap() • updateTransport() • updateGeneralCalendar() • updateInformation() • viewaccount() • search() • getWebsite() • getEmail() • getAdminID() • getPasword() • setWebsite() • setEmail() • setType() • updateInformation()
Responsibilities	**Collaborator**
Search Administrative Body	Administrative Body
Search Infrastructure	Infrastructure
Search Calendar	Calendar
Update Administrative Body	Administrative Body
Update Infrastructure	Infrastructure
Update Map	Map
Update transport	transport

Table 24: Class Card of General Calendar

General Calendar	
Attributes	**Methods**
Date Holiday Event	• getDate() • getHoliday() • getEvent() • setDate() • setHoliday() • setEvent()
Responsibilities	**Collaborator**
Store University Holiday	Admin
Store University Event	Admin

Table 24: Class Card of Specific Calendar

Specific Calendar	
Attributes	**Methods**
Date Holiday Event Academic Schedule	• getDate() • getHoliday() • getEvent() • setDate() • setHoliday() • setEvent() • getAcademicYear() • getAcademicSchedule() • setAcademicSchedule()
Responsibilities	**Collaborator**
Store University Holiday	Admin
Store University Event	Admin
Get Academic Year	Student

Table 24: Class Card of Infrastructure

Infrastructure	
Attributes	**Methods**
Establishment Year Description Location	• getEstablishmentYear() • getDescription() • getLocation() • setEstablishmentYear() • setDescription() • setLocation () • viewLocation()
Responsibilities	**Collaborator**
Collaborate with Administrative Body	Administrative Body

Table 24: Class Card of Administrative Body

Administrative Body	
Attributes	**Methods**
Name Type	• getName() • getUserID() • getType() • setType() • setName()
Responsibilities	**Collaborator**
Make Decision	Infrastructure
Get User ID	User

Table 24: Class Card of Transport

Transport	
Attributes	**Methods**
TransportationID Type Destination Stoppage	• getTransportID() • getType() • getDestination() • setTransportID() • setDestination() • setStoppage() • getLocation()
Responsibilities	**Collaborator**
Get Location	Map
List Stoppage	Administrative Body
Use Transport	User
View Current Position	Map

Table 24: Class Card of Map

Map	
Attributes	**Methods**
Location ID Destination Area	• getLocationID() • getArea() • getDestination() • setLocationID() • setDestination() • setArea() • viewLocation()
Responsibilities	**Collaborator**
View Exciting Places	Infrastructure

Table 24: Class Card of Group Chat

Group Chat	
Attributes	**Methods**
Messenger Communication Chat ID Message	• getMessage() • setMessage() • getChatID() • setChatID() • username() • time()
Responsibilities	**Collaborator**
Get User Name	Student
Get Time	System
Send Message	Student
Get Academic Year	Student

Table 24: Class Card of Database

Database	
Attributes	**Methods**
store update retrieve information	• store() • update() • getInformation() • SetInforamtion() • retrieve() • viewInformation()
Responsibilities	**Collaborator**
Store Information	Admin
update Information	Admin
Retrieve information	user

Table 24: Class Card of System

System	
Attributes	**Methods**
GenerateQuery Send Retrieve	• generateSearchQuery() • send() • retrieve() • showInfo() • FormatInfo()
Responsibilities	**Collaborator**
Generate search key to database query	database
Send information to database	admin
Format information to show to user	user

6.9 UML Diagram

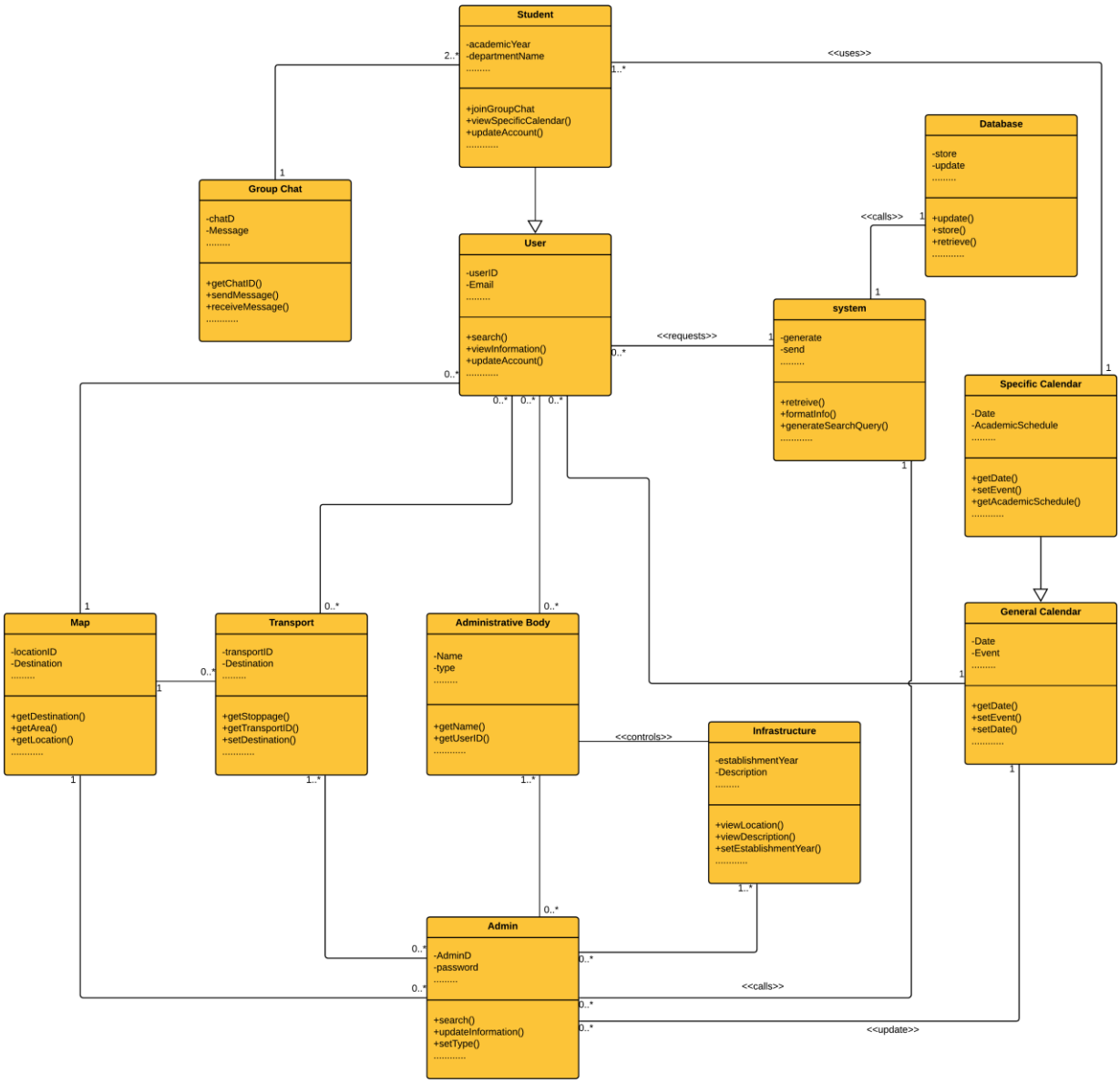


Figure 51: UML Diagram of Dhaka University Calendar Management Application

Chapter 7: Flow-Oriented Model

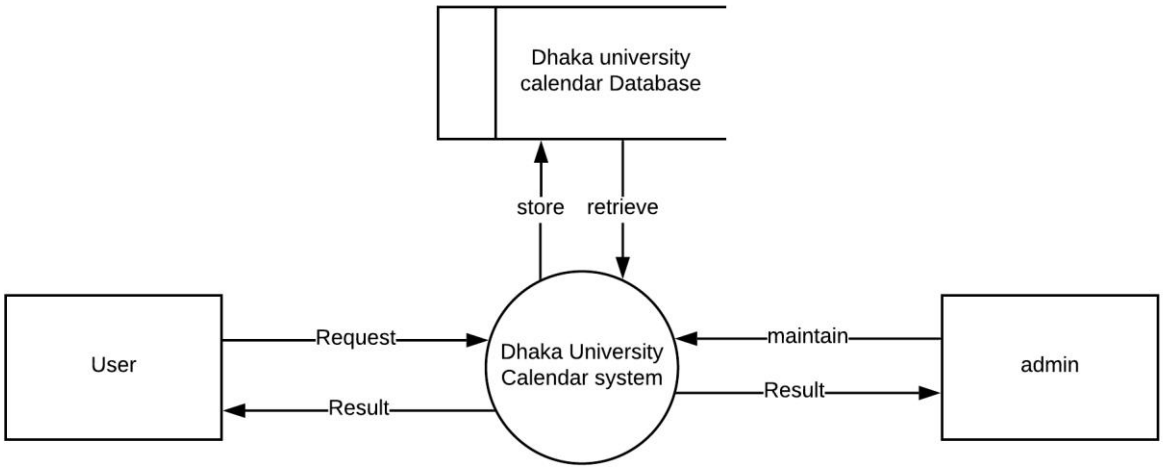
This chapter focuses on the flow oriented modeling.

7.1 Introduction

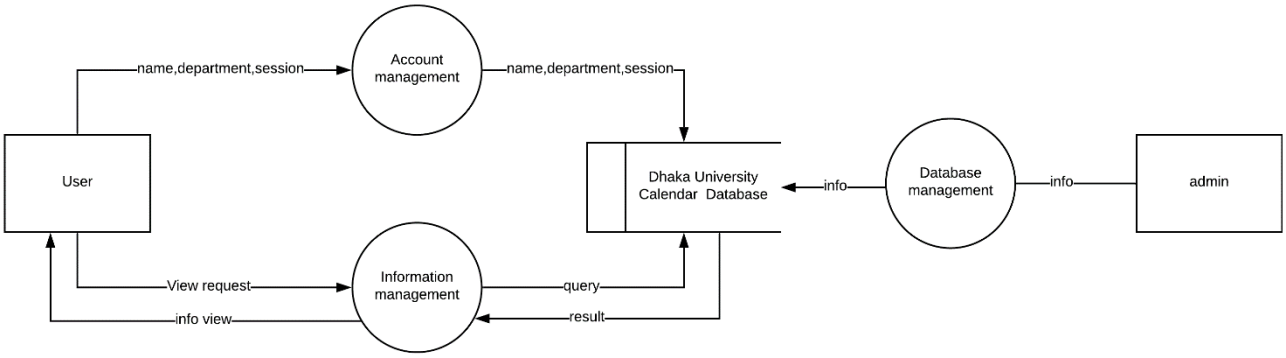
Although data flow-oriented modeling is perceived as an outdated technique by some software engineers, it continues to be one of the most widely used requirements analysis notations in use today. It provides additional insight into system requirements and flow.

7.2 Data Flow Diagram (DFD)

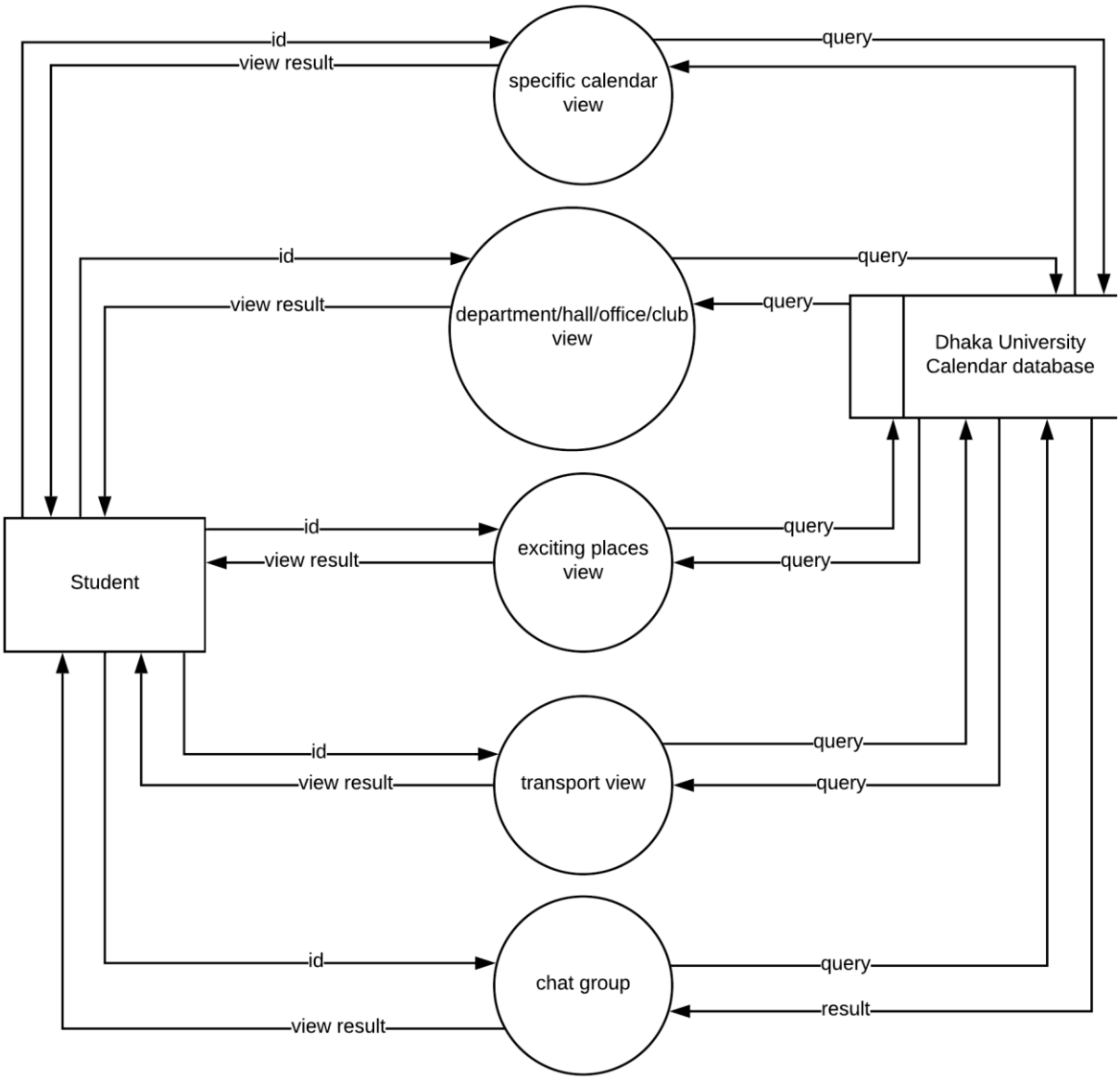
The DFD takes an input-process-output view of a system. In the figures, data objects are represented by labeled arrows and transformations are represented by circles.



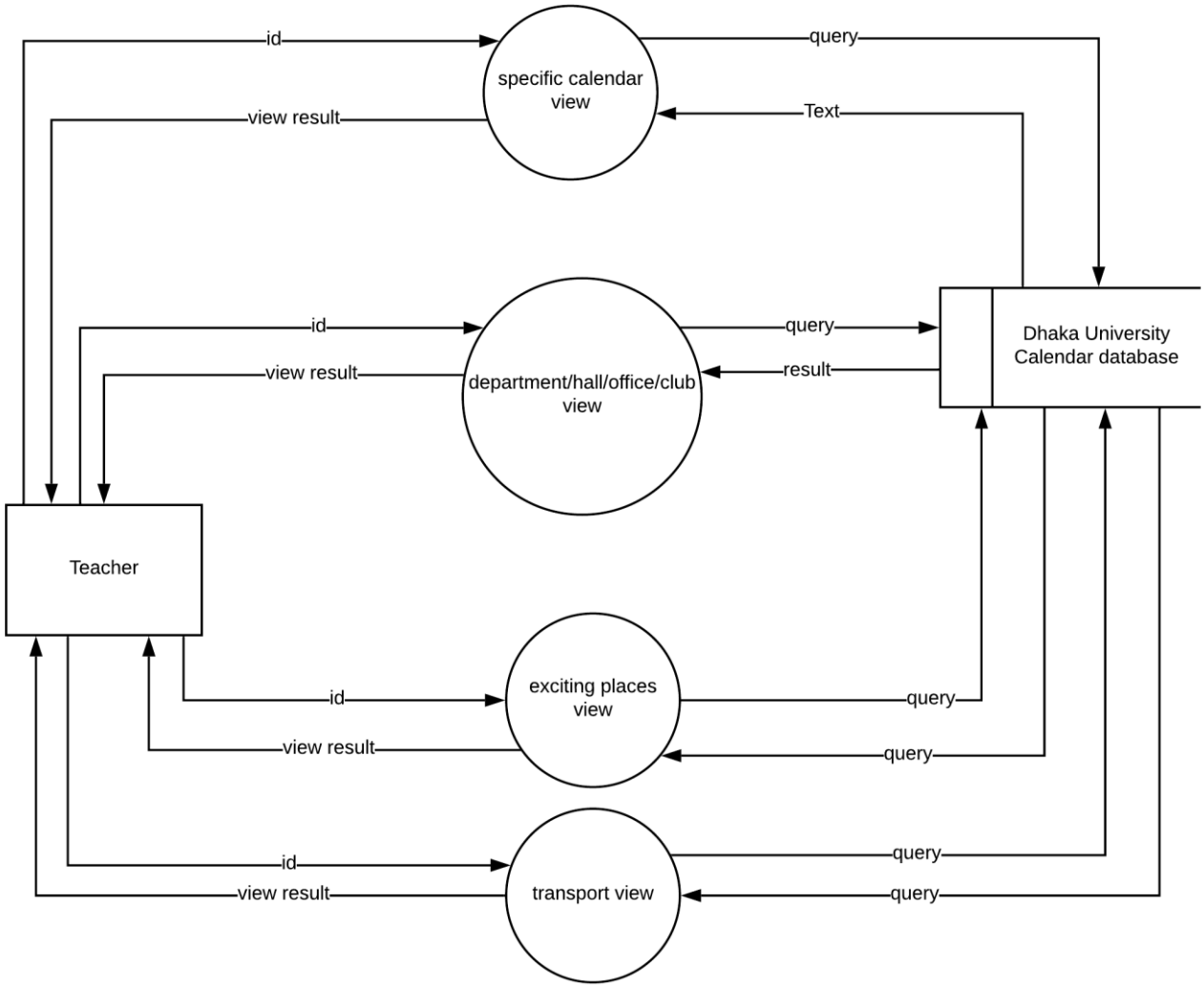
Level: 0



level: 1


id
view result
specific calendar view
query
id
view result
department/hall/office/club view
query
query
Dhaka University Calendar database
Student
id
view result
exciting places view
query
query
id
view result
transport view
query
query
id
view result
chat group
query
result

level: 3.1.1


id
specific calendar view
query
view result
Text
id
department/hall/office/club view
query
result
Dhaka University Calendar database
view result
Teacher
id
exciting places view
query
view result
query
id
transport view
query
view result
query

level: 3.1.2

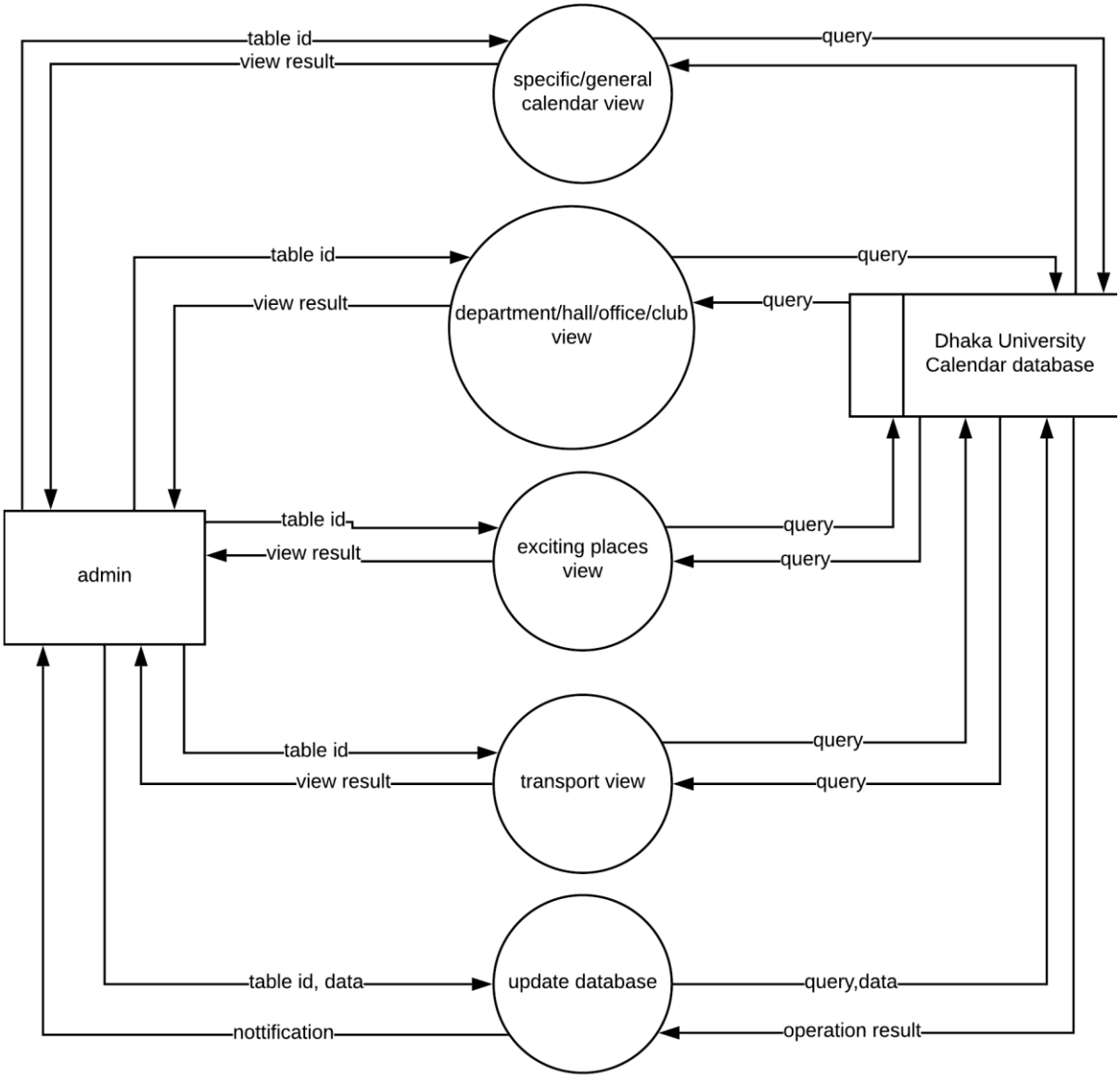
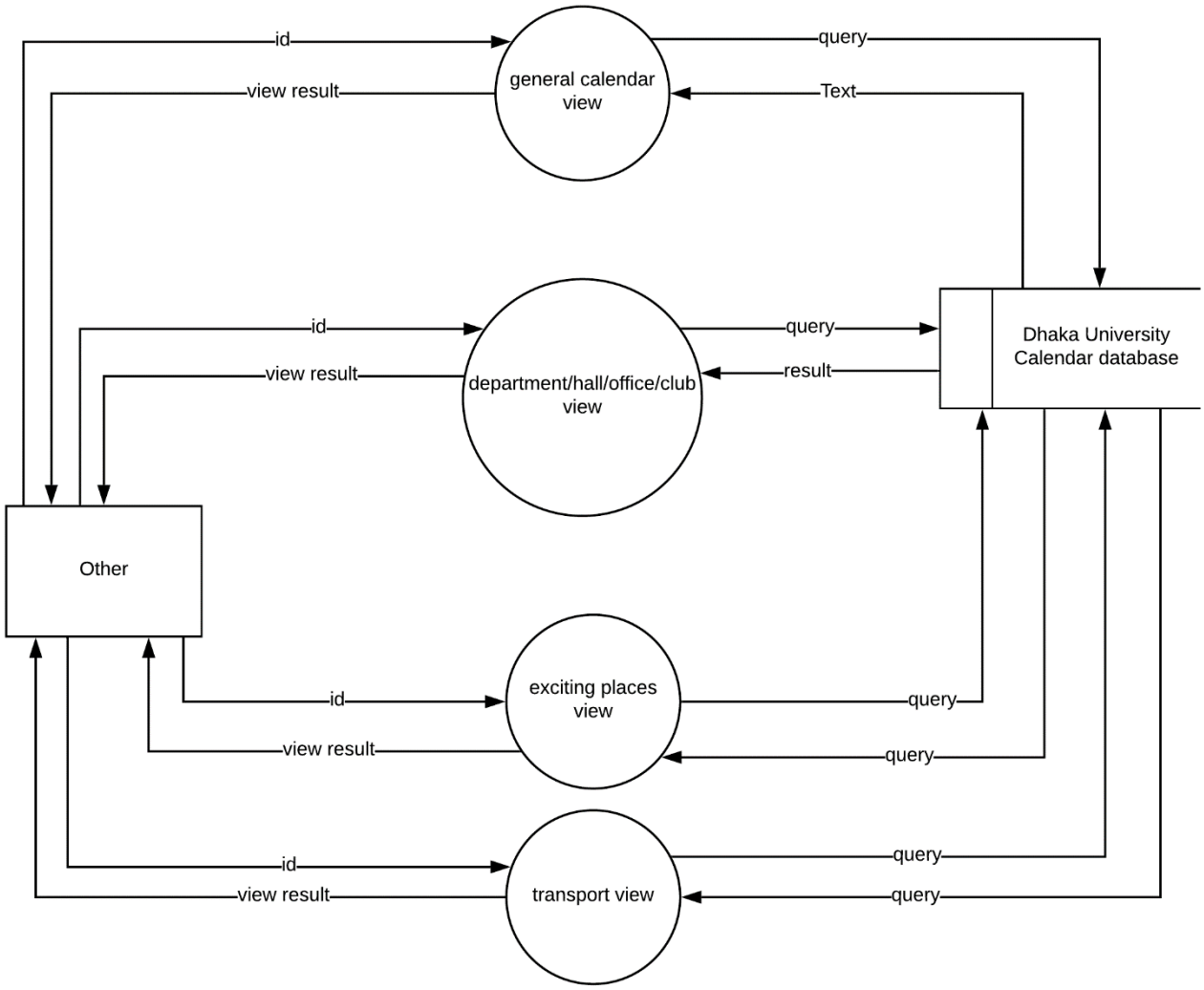
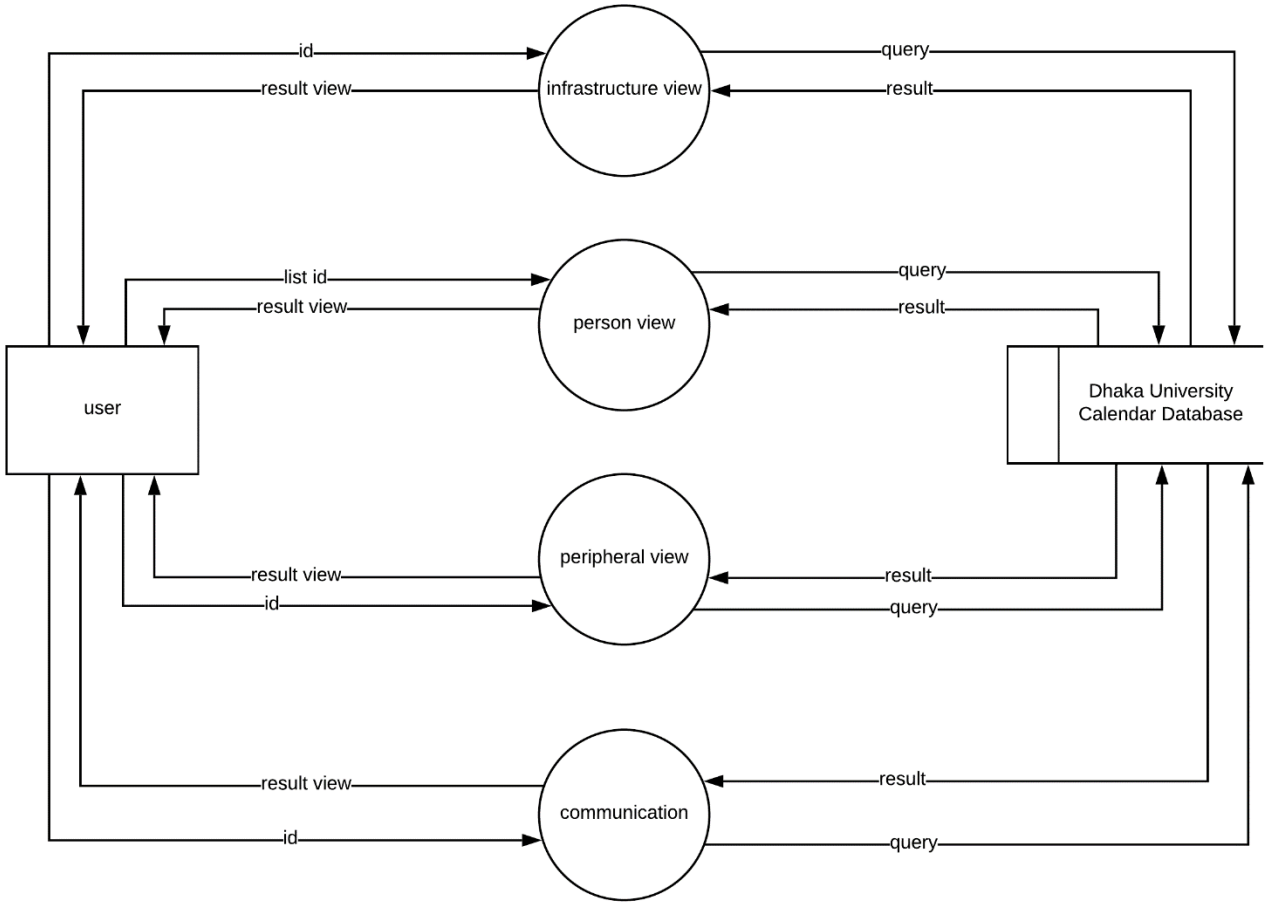


table id
view result
specific/general calendar view
query
table id
view result
department/hall/office/club view
query
query
Dhaka University Calendar database
admin
table id
view result
exciting places view
query
query
table id
view result
transport view
query
query
table id, data
nottification
update database
query,data
operation result

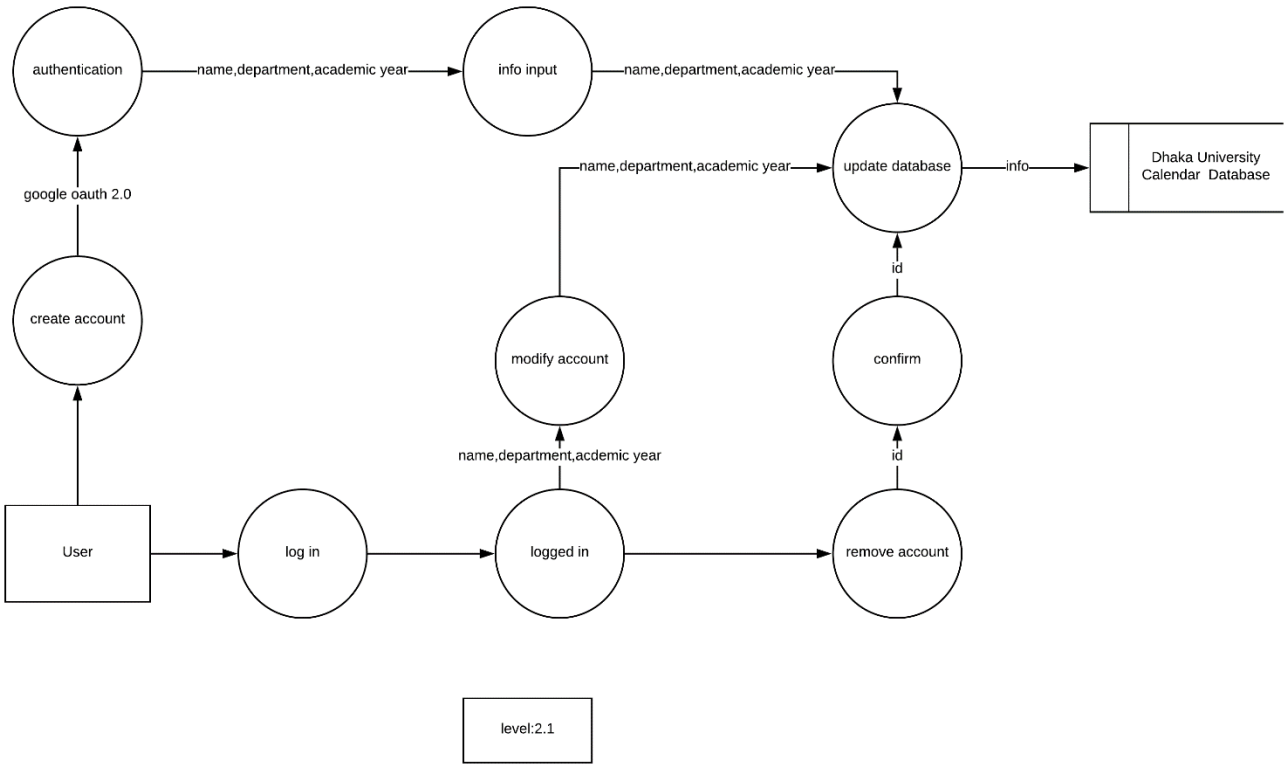
level: 3.1.4


id
query
general calendar view
view result
Text
id
query
Dhaka University Calendar database
department/hall/office/club view
view result
result
Other
id
exciting places view
query
view result
query
id
query
transport view
view result
query

level: 3.1.3


id
result view
infrastructure view
query
result
list id
result view
person view
query
result
user
Dhaka University
Calendar Database
result view
id
peripheral view
result
query
result view
id
communication
result
query

level: 2.2

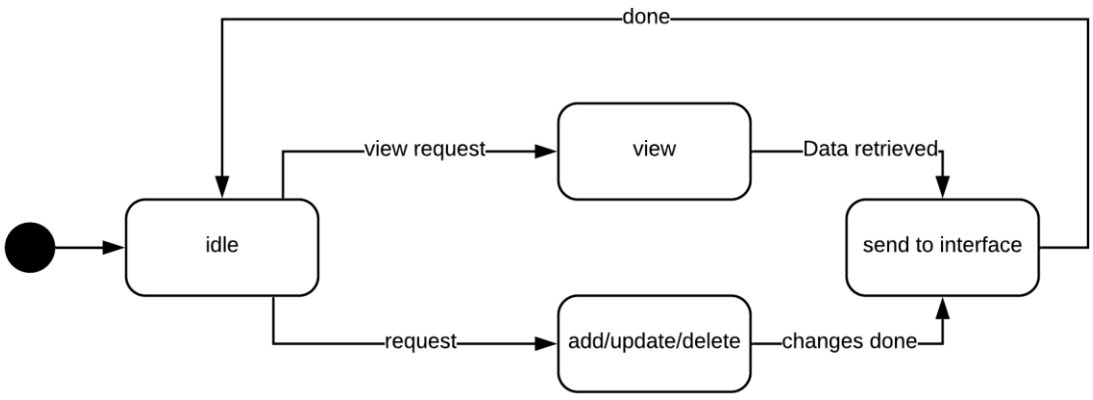

authentication
name,department,academic year
info input
name,department,academic year
update database
info
Dhaka University
Calendar Database
google oauth 2.0
name,department,academic year
id
create account
modify account
confirm
name,department,acdemic year
id
User
log in
logged in
remove account
level:2.1

Chapter 8: Behavioral Model

The behavioral model indicates how software will respond to external events.

8.1 State Diagram

State diagram represents active states for each class the events (triggers).



class : Calendar

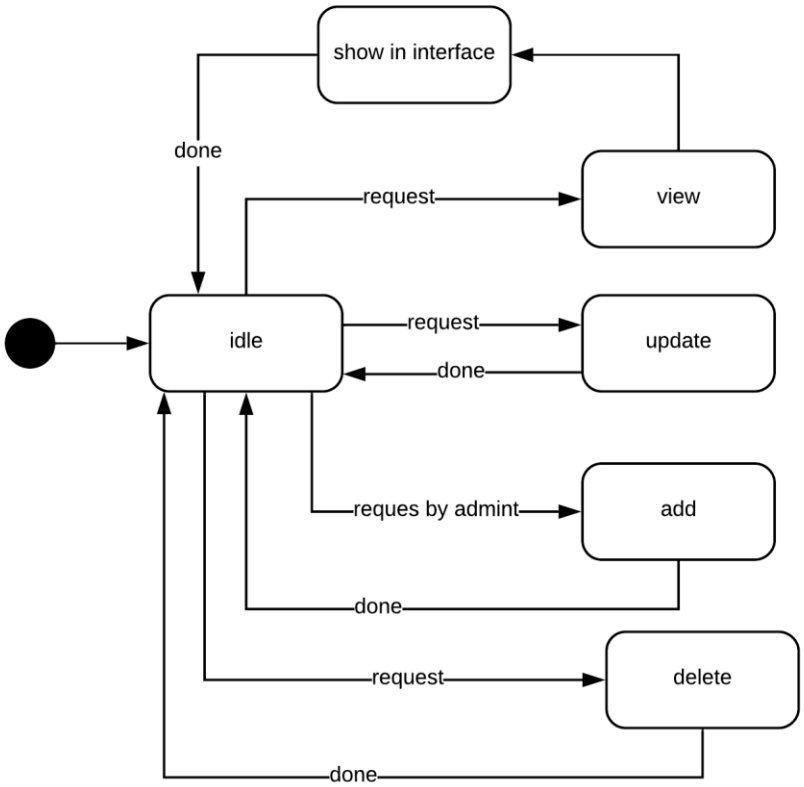


Figure: Administrative body

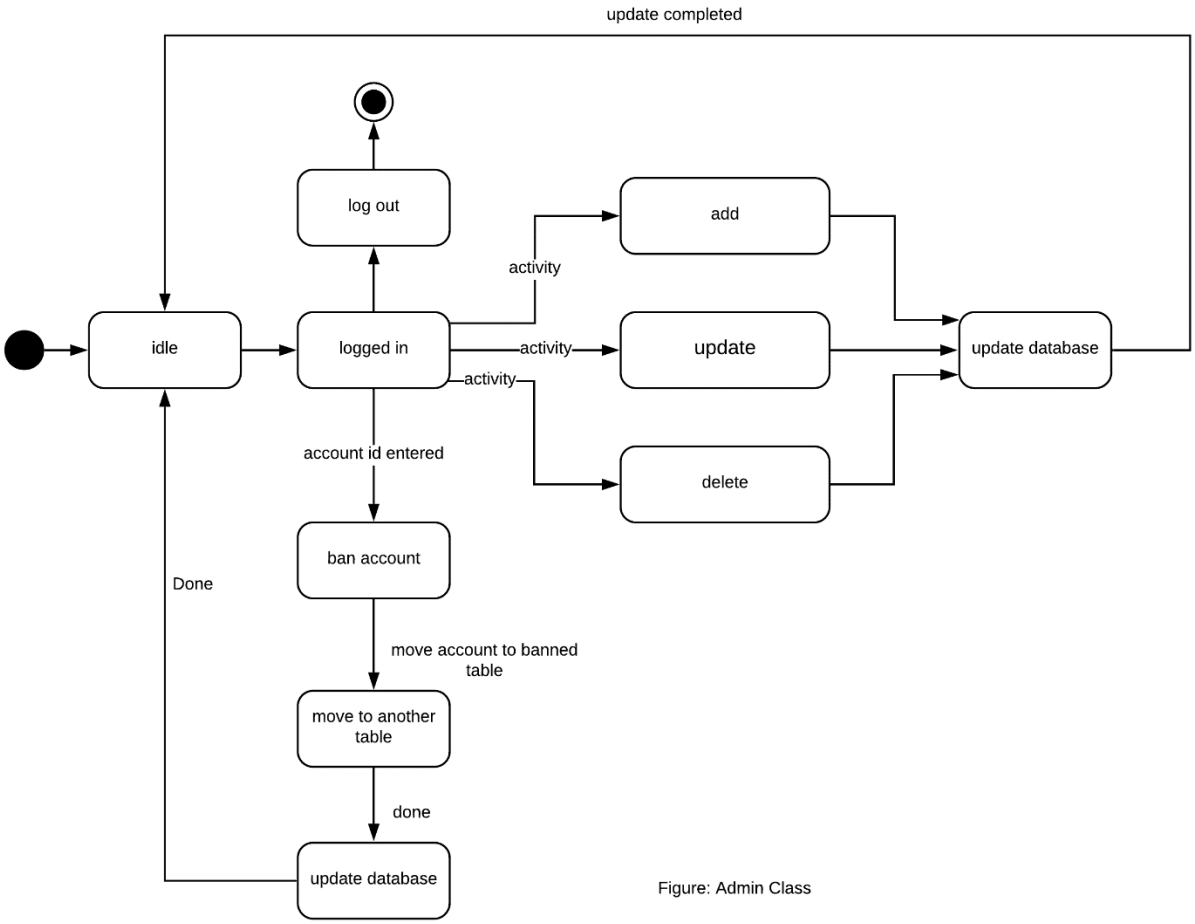


Figure: Admin Class

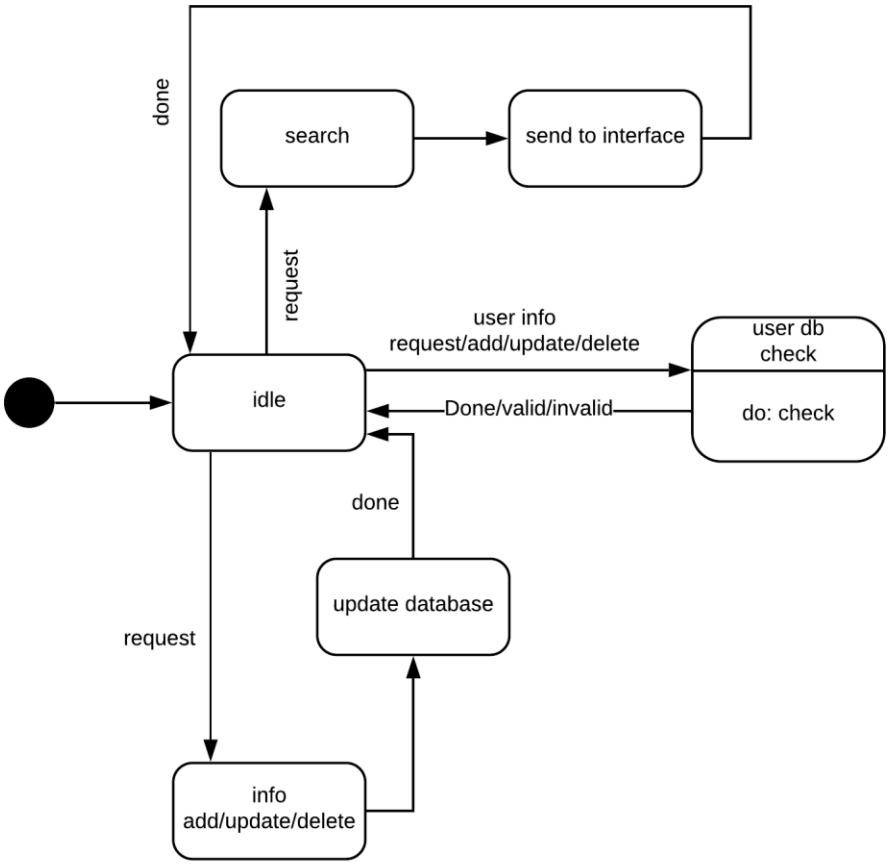
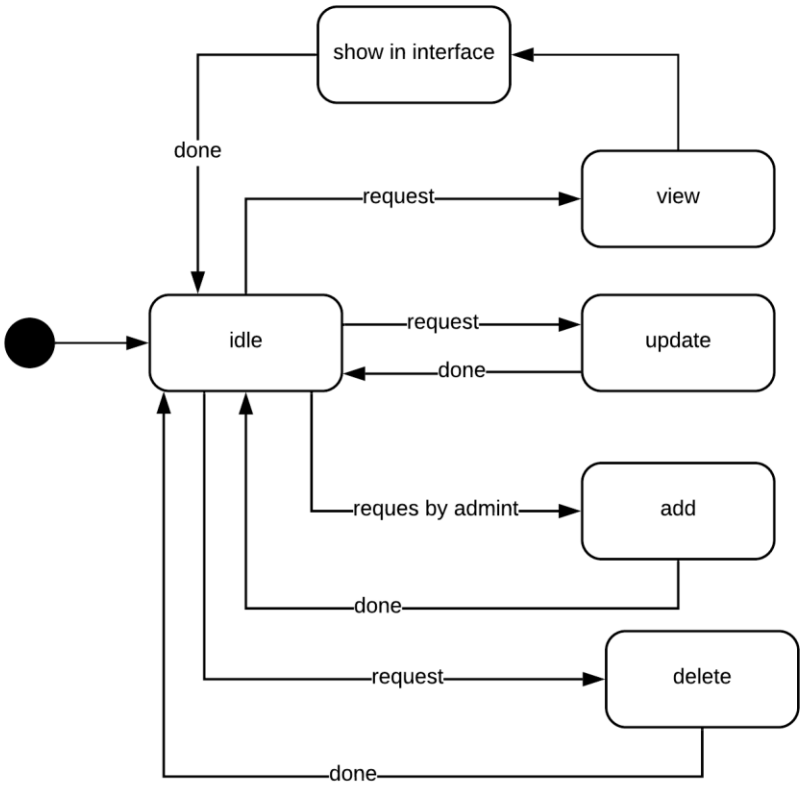


Figure: Database Class



level: infrastructure

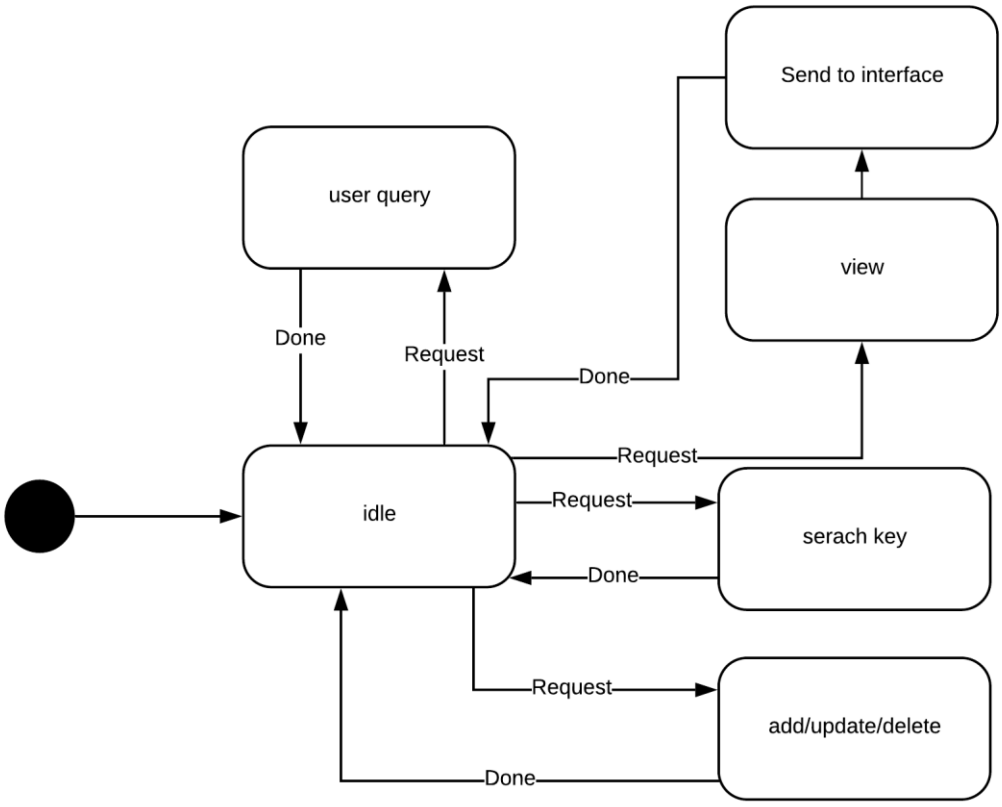
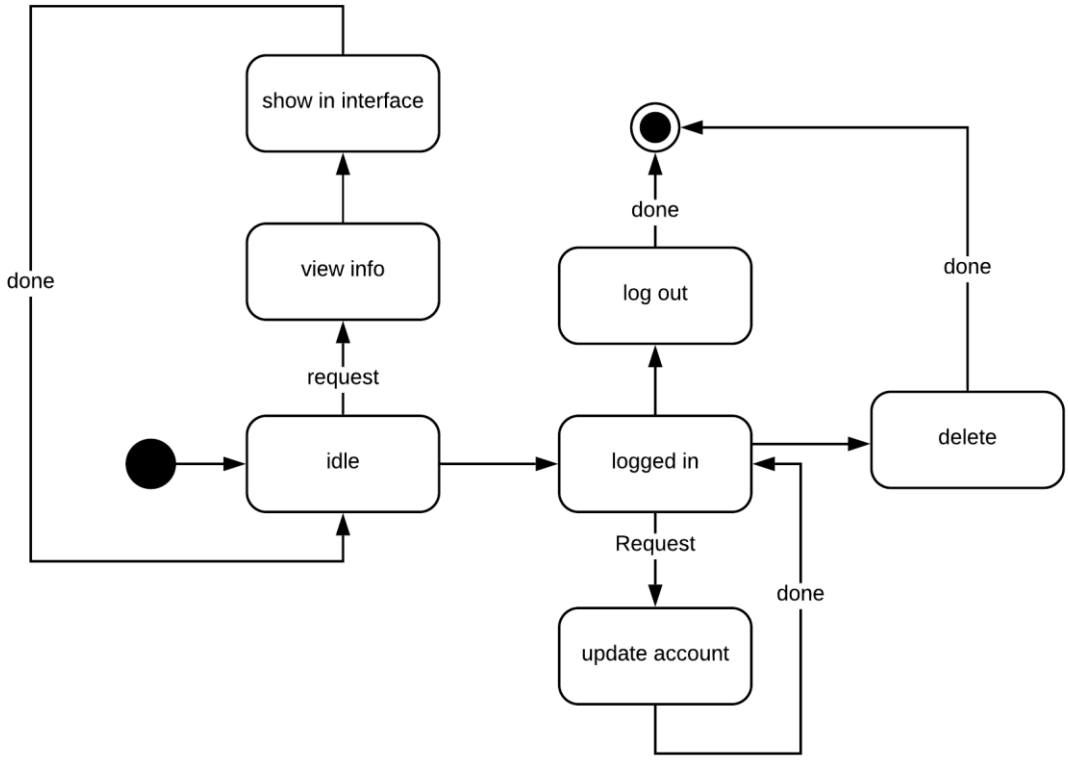


Figure: System Class



level: user class

8.2 Sequence Diagram

Sequence diagram indicates how events cause transitions from object to object.

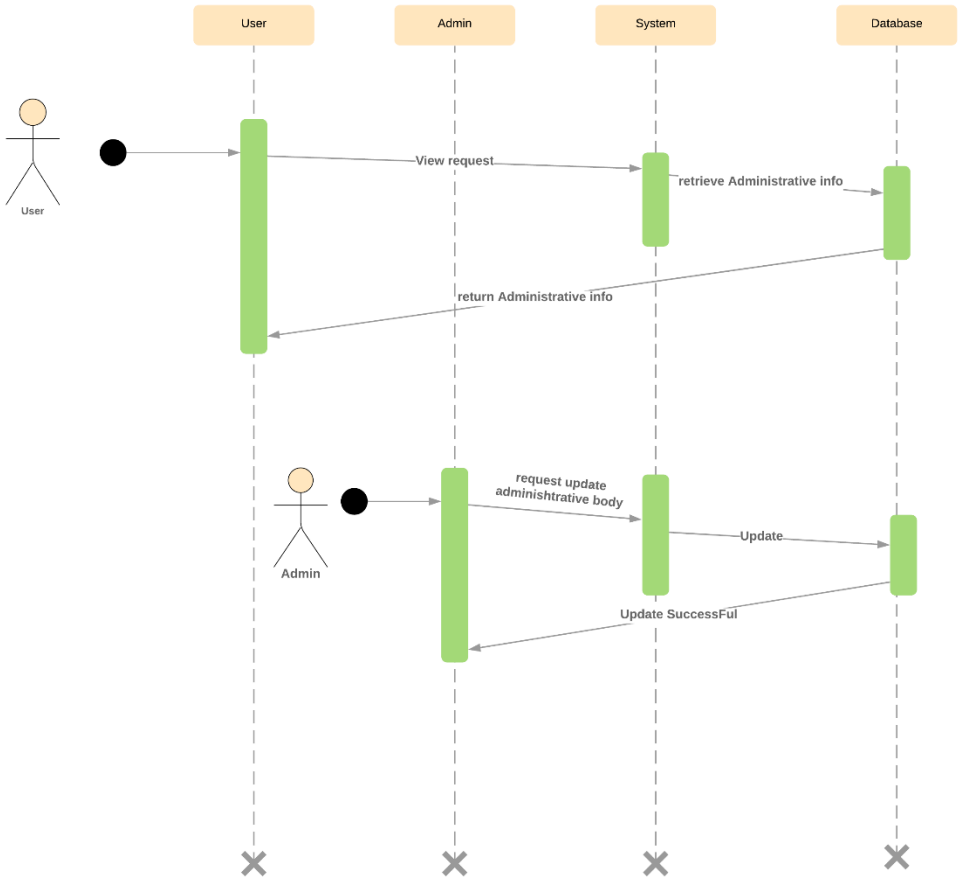


Figure : Sequence Diagram for Sign up

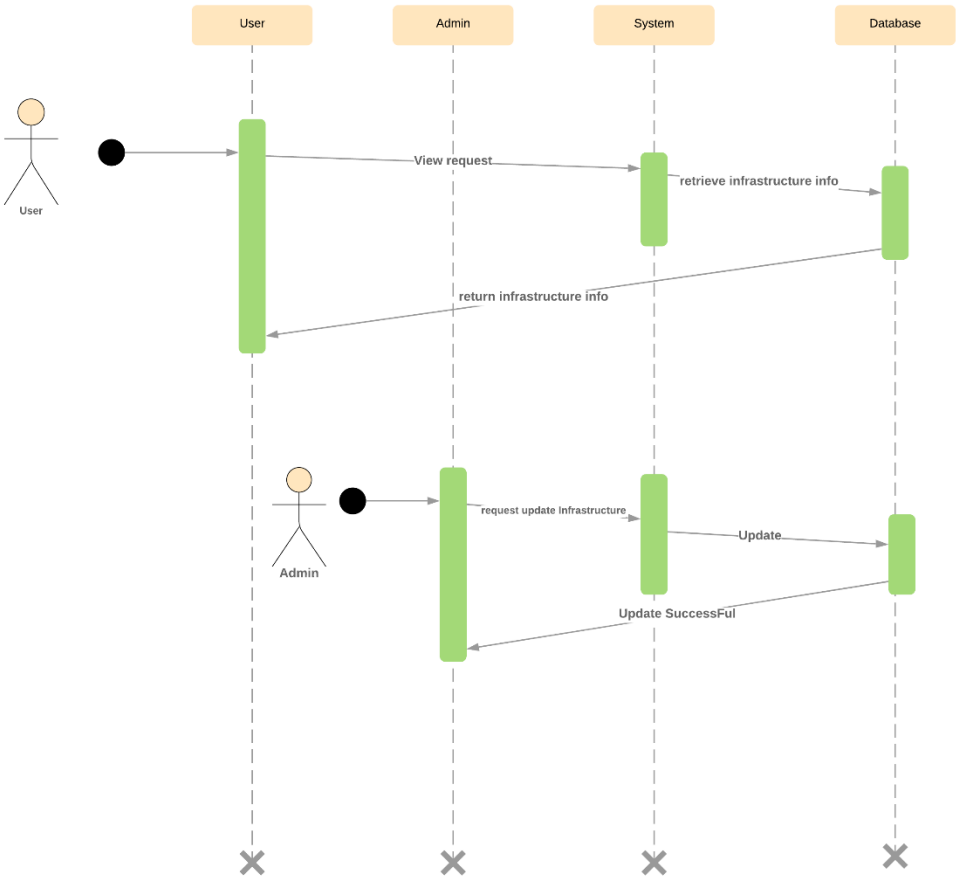


Figure : Sequence Diagram for Infrastructure

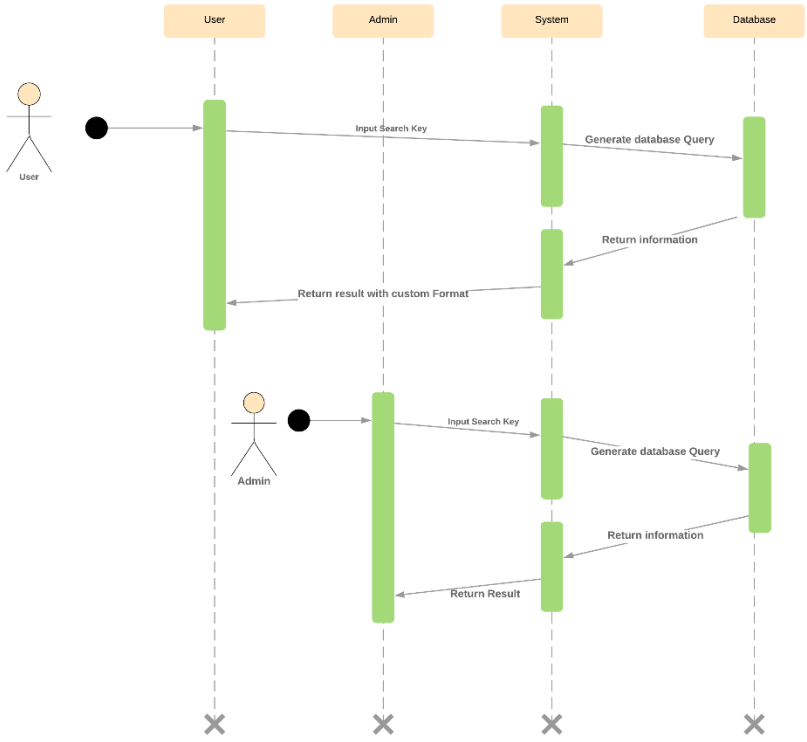


Figure : Sequence Diagram for Search

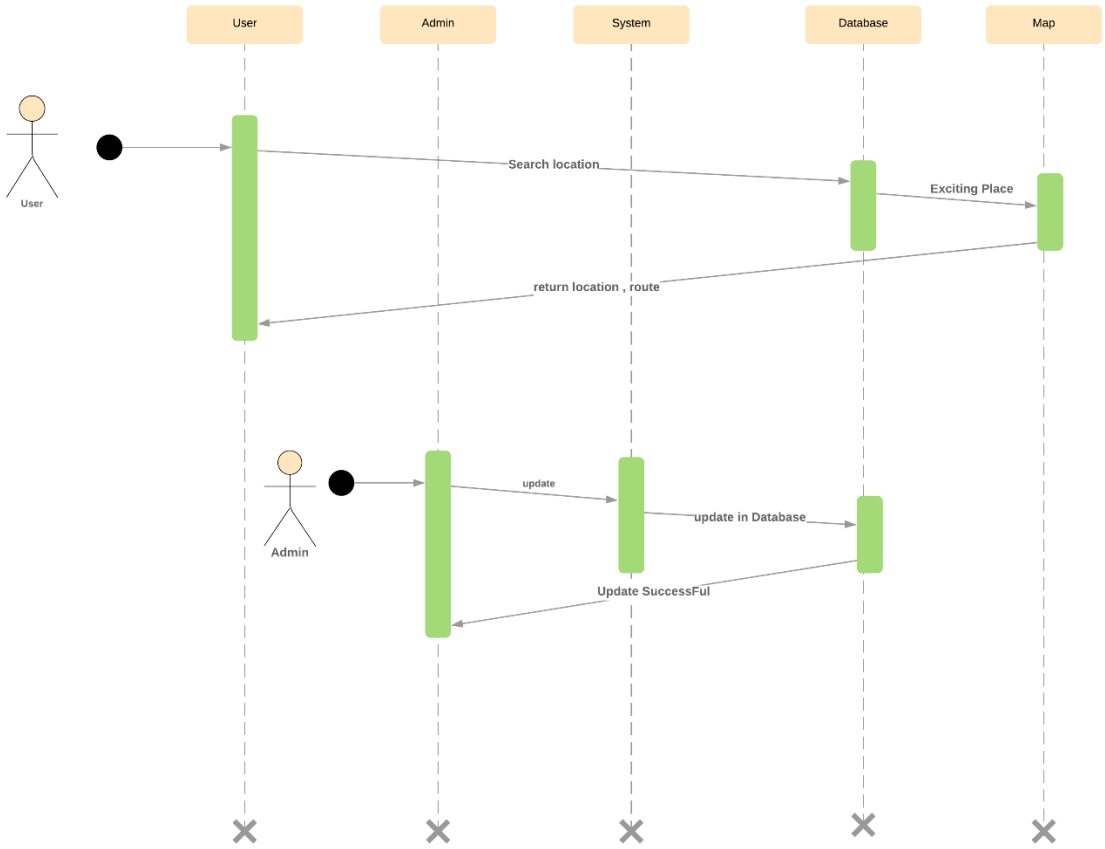


Figure : Sequence Diagram for Map

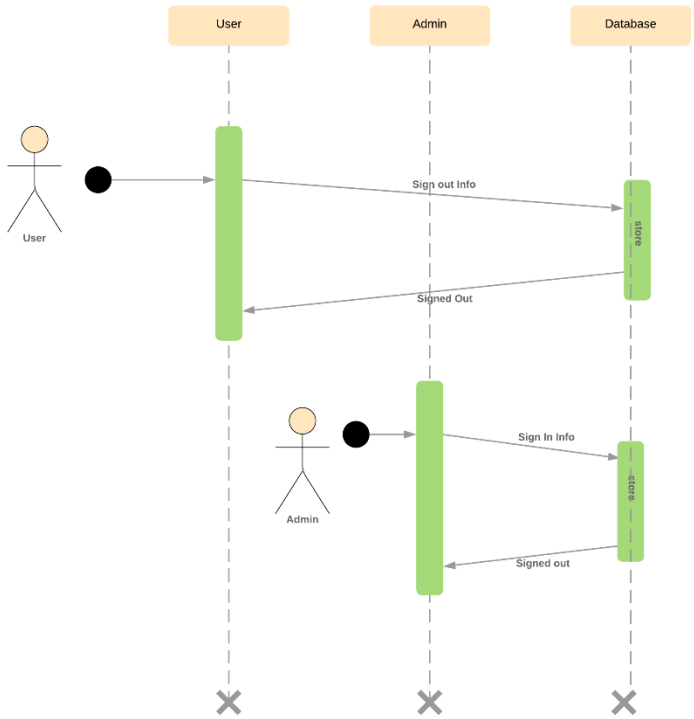


Figure : Sequence Diagram for Sign out

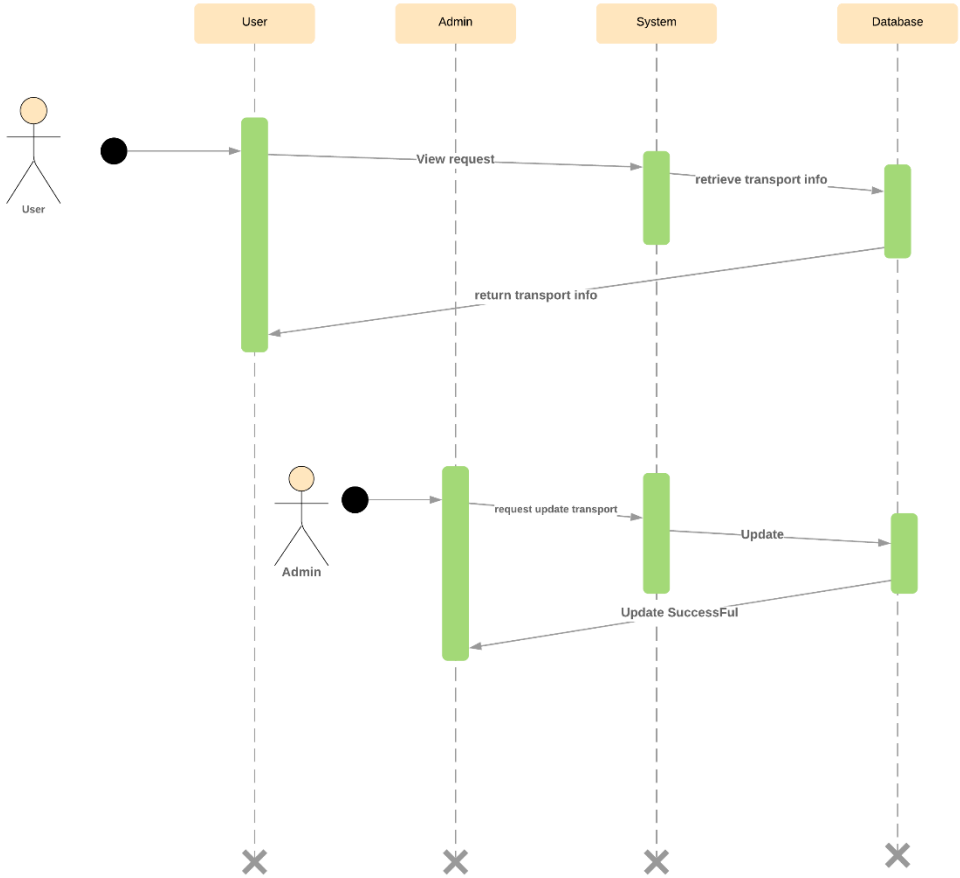


Figure : Sequence Diagram for Transport

Chapter 9: Conclusion

We are pleased to submit the SRS Dhaka University Calendar Management Application From this, the readers will get a clear and easy view of the overall system. This SRS document can be used effectively to maintain the software development cycle. It will be very easy to conduct the whole project using this SRS.

We tried our best to remove all dependencies and make an effective and fully designed SRS. We believe that the reader will find it in order.

References

1. Book- Pressman, Roger S. Software Engineering: A Practitioner's Approach (7th Edition)

2. ER diagram- https://www.tutorialcup.com/dbms/er-data-model.htm [Last accessed: 18/03/2019 4.30 PM]

3. CRC diagram-http://csis.pace.edu/~marchese/cs615sp/L4New/l4senewf.htm [Last accessed: 17/03/2019 3.50 PM]