DOCUMENT

SOFTWARE REQUIREMENT SPECIFICATION

RESERVAROOM

ROOM BORROWING APPLICATION

for:

…………………………….…

Prepared By:

M Hendri Febriansyah (5114100036)

Tosca Yoel Connery (5114100061)

Jurusan Teknik Informatika - Institut Teknologi Sepuluh Nopember

Kampus ITS Keputih Sukolilo Surabaya

	Jurusan **Teknik Informatika** **ITS**	**Document Number**		**Page**
		SRS-001		**1/#52**
		Revision	**-**	DD MM YYYY

LIST OF CHANGES

Revision	Description
A	
B	
C	
D	
E	
F	
G	

INDEX DATE	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperiksa oleh								
Approved by								

List Changes Page

Page	Revision	Page	Revision

Contents

1 Preliminary
1.1 Purpose of Writing The Document
1.2 Scope of The Problem
1.3 Definitions and Terms
1.4 Rules of Naming and Numbering
1.5 Reference
1.6 Document Overview
2 General Description of This Software
2.1 General Description of System
2.2 Product Function
2.3 User Characteristics
2.4 Limitation
2.5 Operation Interface
3 General Requirement Description
3.1 External Interface Requirement
3.1.1 Input/Output Interface
3.1.2 Hardware Interface
3.1.3 Software Interface
3.1.4 Communication Interface
3.2 Functional Description
3.2.1 Use Case Diagram
3.2.2 Function 1: Create Account
3.2.2.1 Scenario: Create Account
3.2.2.2 Activity Diagram: Create Account
3.2.2.3 Sequence Diagram: Create Account
3.2.2.4 Object Collaboration Diagram: Create Account
3.2.3 Function 2: Edit Account
3.2.3.1 Scenario: Edit Account
3.2.3.2 Activity Diagram: Edit Account
3.2.3.3 Sequence Diagram: Edit Account
3.2.3.4 Object Collaboration Diagram: Edit Account
3.2.4 Function 3: View Schedule
3.2.4.1 Scenario: View Schedule
3.2.4.2 Activity Diagram: View Schedule
3.2.4.3 Sequence Diagram: View Schedule
3.2.4.4 Object Collaboration Diagram: View Schedule
3.2.5 Function 4: Apply Proposal Music Studio
3.2.5.1 Scenario: Apply Proposal Music Studio
3.2.5.2 Activity Diagram: Apply Proposal Music Studio
3.2.5.3 Sequence Diagram: Apply Proposal Music Studio
3.2.5.4 Object Collaboration Diagram: Apply Proposal Music Studio
3.2.6 Function 5: Apply Proposal Classroom

3.2.6.1 Scenario: Apply Proposal Classroom
3.2.6.2 Activity Diagram: Apply Proposal Classroom
3.2.6.3 Sequence Diagram: Apply Proposal Classroom
3.2.6.4 Object Collaboration Diagram: Apply Proposal Laboratory
3.2.7 Function 6: Apply Proposal Laboratory
3.2.7.1 Scenario: Apply Proposal Laboratory
3.2.7.2 Activity Diagram: Apply Proposal Laboratory
3.2.7.3 Sequence Diagram: Apply Proposal Laboratory
3.2.7.4 Object Collaboration Diagram: Apply Proposal Laboratory
3.2.8 Function 7: Apply Proposal Hall
3.2.8.1 Scenario: Apply Proposal Hall
3.2.8.2 Activity Diagram: Apply Proposal Hall
3.2.8.3 Sequence Diagram: Apply Proposal Hall
3.2.8.4 Object Collaboration Diagram: Apply Proposal Hall
3.2.9 Function 8: Head of Major Give Permission
3.2.9.1 Scenario: Head of Major Give Permission
3.2.9.2 Activity Diagram: Head of Major Give Permission
3.2.9.3 Sequence Diagram: Head of Major Give Permission
3.2.9.4 Object Collaboration Diagram: Head of Major Give Permission
3.2.10 Function 9: Administrator Give Permission
3.2.10.1 Scenario: Administrator Give Permission
3.2.10.2 Activity Diagram: Administrator Give Permission
3.2.10.3 Sequence Diagram: Administrator Give Permission
3.2.10.4 Object Collaboration Diagram: Administrator Give Permission
3.2.11 Function 10: Get Permission
3.2.11.1 Scenario: Get Permission
3.2.11.2 Activity Diagram: Get Permission
3.2.11.3 Sequence Diagram: Get Permission
3.2.11.4 Object Collaboration Diagram: Get Permission
3.2.12 Function 11: Finish Borrowing
3.2.12.1 Scenario: Finish Borrowing
3.2.12.2 Activity Diagram: Finish Borrowing
3.2.12.3 Sequence Diagram: Finish Borrowing
3.2.12.4 Object Collaboration Diagram: Finish Borrowing

3.3 Classes Description
3.3.1 Class Diagram
3.3.2 Description of The Problem Domain
3.3.3 Description of The Class Controller
3.3.4 Description of The Class Entity (Persistant)
3.3.5 Description of The Class Boundary
3.4 Data Flow Diagram
3.5 Non Functional Need
3.6 Design Restriction
3.7 Summary of Need
3.7.1 Ringkasan Kebutuhan Fungsional
3.7.2 Ringkasan Kebutuhan Non Fungsional

Daftar Tabel

Tabel 1 Aturan Penamaan dan Penomoran
Tabel 2 Karakteristik Pengguna
Tabel 3 Deskripsi Kelas Domain Persoalan
Tabel 4 Deskripsi Kelas Pengendali
Tabel 5 Deskripsi Kelas *Entity*
Tabel 6 Deskripsi Kelas *Boundary*
Tabel 7 Deskripsi Kebutuhan Non Fungsional
Tabel 8 Ringkasan Kebutuhan Fungsional
Tabel 9 Ringkasan Kebutuhan Non Fungsional

List Picture

Picture 1. Use Case Diagram. 12
Picture 2. Activity Diagram “Create Account” 13
Picture 3. Activity Diagram “Edit Account” 16
Picture 4. Activity Diagram “View Schedule” 17
Picture 5. Activity Diagram “Apply Proposal Music Studio” 19
Picture 6. Activity Diagram “Apply Proposal Classroom” 21
Picture 7. Activity Diagram “Apply Proposal Laboratory” 24
Picture 8. Activity Diagram “Apply Proposal Hall” 25
Picture 9. Activity Diagram “Head of Major Give Permission” 27
Picture 10. Activity Diagram “Administrator Give Permission” 29
Picture 11. Activity Diagram “Get Permission” 30
Picture 12. Activity Diagram “Finish Borrowing” 32
Picture 13. Class Diagram 33

1 Preliminary

1.1 Purpose of Writing The Document

This document contains the Software Requirements Specification (SRS) for Room Booking System. The purpose of writing this document is to provide an explanation of the results of the software to be built either in the form of a general overview as well as detailed and thorough explanations.

This document will be used as reference material in the process of development and evaluation materials during the process of software development and the finishing. With this Software Requirement Specification document, we expect this development will be better and have a good direction without making any ambiguity, especially for all developers of this room booking information system.

1.2 Scope of The Problem

The software that will we build is Room Booking Information System on Informatics Department, a web based desktop application that we can used to booking a classroom, hall, music studio, and laboratory on Informatics Department. This software can do all this things below:

1) Record all schedule of room usage by user
2) A tools to book a room easily
3) Media too view all schedule of the room reservation status

With this room booking software we hope that all user can find out the status of all room and make a booking easily for each room.

1.3 Definitions and Terms

Below is a list of important definitions and terms which will be used in this Software Requirement Specification document:

- SRS : *Software Requirements Specification*
  A document from analysis which is contain all requirement of this software
- IEEE : *Institute of Electrical and Electronics Engineering*
  International standard for developing and designing software
- ANSII : *American National Standart Institute*

1.4 Rules of Naming and Numbering

This Software Requirement Software document is using various and different rules to naming and numbering for serveral certain part. The rules of naming and numbering that we will used in this document is listed on Table 1.

Table 1 Rules of Naming and Numbering

Hal/Bagian	Aturan Penomoran/Penamaan
Functional Requirement	SRS-FXX : Refering to the XXth functional requirement
Non-Functional Requirement	SRS-NFXX : Refering to the XXth non-functional requirement
Summary of Functional Requirement	SRS-Fxxx where xxx is three digit number started from 000
Summary of Non-Functional Requirement	SKPL-NFxxx where xxx is three digit number started from 000

1.5 *Reference*

Documents below is used as reference while making this Software Requirement Specification:

1) The *Software Requirement Specification (SRS) document – IEEE* 1999 by Karl E. Wiegers.
2) The User Guidelines and Software Requirement Specification, Informatics Departmen, Institute Teknologi Sepuluh Nopember.

1.6 *Document Overview*

This document outlines consist of three chapter with this following explanation:

- Chapter 1 Preliminary, an introduction of this Software Requirement Specification that consist of purpose of this document, the scope of the problem, definitions and terms that used in this document, also general description of this document which is the summary of this Software Requirement Specification document.
- Chapter 2 General Description of This Software, this chapter define the perspective of software products, assumptions and dependencies used in this Room Booking information system development.
- Chapter 3 Detailed Requirements Description, describe all special requirements for this room booking system which includes an external interface requirements, functionality requirements, performance requirements, design constraints, attributes of software, and other requirements of this information system.

2 General Description of This Software

2.1 *General Description of System*

This room booking information system gather all information about room that can be borrowed in Informatics Department. The information is like room capacity, schedule of reservation, reservation cost, and inventory of all room. There are four type of room that can be borrowed. Classroom, laboratory, hall, and music studio. User is divided into two kind, the first is informatics students, and the second is non-informatics student. Borrowing the room is free for informatics students. But for non-informatics student will be charged. All type of room have its own cost. There is also the administrator and head of major which have the right to give permission to borrow the room.

The information system that will we build has a couple main part based on the type of user :

1) From Head of Major side, this actor can view all the schedule and give a decision whether the reservation of the room is permitted or not based on type event that will be held on the reserved room.
2) From Administrator side, this actor have a same role, but the administrator will make sure that there is nothing will disturb this borrowing, may be something that not recorded like room repair.

This system has an access right protection for all type of user to prevent the user from doing things that not allowed into the system.

2.2 Product Function

This RESERVAROOM software has some main function:

1. (UC-001) User can create account
2. (UC-002) User can edit account
3. (UC-003) User can view schedule
4. (UC-004) User can apply proposal for music studio
5. (UC-005) User can apply proposal for classroom
6. (UC-006) User can apply proposal for laboratory
7. (UC-007) User can apply proposal for hall
8. (UC-008) Head of Major can give permission
9. (UC-009) Administrator can give permission
10. (UC-010) User can get permission

2.3 User Characteristics

The user characteristics of this information system is contained in this following table:

No	User Category	Task	Access Right in Application	Ability to be Owned
1.	Head of Major	The highest right to give permission to borrow the	Can give a decission whether the borrowing is permitted or not	1. Can operate a computer 2. Can use the internet

		room	based on the purpose of the borrowing	
2.	Administrator	Manage information system and oversee the granting of borrowing room	Can give a decision whether the borrowing is peritted or not based on the room condition	1. Can operate a computer 2. Can use the internet 3. Can use the database
3.	Informatics User	Borrowing a room	Can propose to borrow a room	1. Can operate a computer 2. Can use the internet
4.	Non-informatics User	Borrowing a room and pay the cost for the room	Can propose to borrow a room	1. Can operate a computer 2. Can use the internet

2.4 Limitation

This room booking information system development have limitation like:

1. Room booking information system created with Laravel, HTML, PHP, SQL-Server
2. Interface only with a very basic style.
3. A limited memory capacity, this memory is for saving user profile picture, data user, room reservation status, and proposal.
4. Software support like DBMS SQL-Server, Sublime Text,

2.5 Operation Interface

This room booking information system will worked under the specification :

Operating System Platform	: Microsoft Windows, Linux, and Mac OS
Operating System version	: Windows Server 2003/XP SP2/Vista/7/8 , Ubuntu 16.04, and Mac OS
DBMS	: SQL-Server
Framework	: Laravel, HTML, PHP, SQL-Server

3 General Requirement Description

3.1 External Interface Requirement

3.1.1 Input / Output Interface

RESERVAROOM can used graphic interface (GUI). User also can interact with system using keyboard and mouse on Windows, Linux and Mac operating system.

3.1.2 Hardware interface

RESERVAROOM system running in computer server that has RESERVAROOM system installed on it.

3.1.3 Software Interface

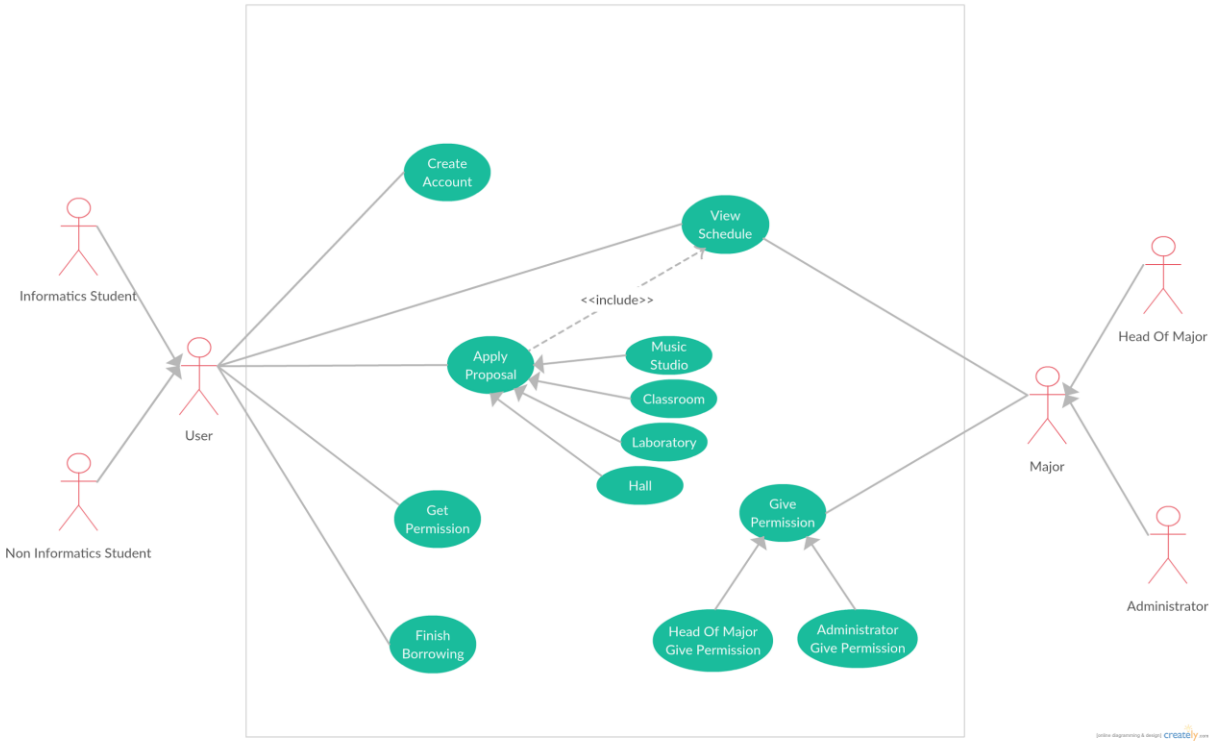
RESERVAROOM is program which is build using HTML, PHP, and SQL-Server. Working on Windows OS, Linux OS, and Mac OS.

3.1.4 Communication Interface

RESERVAROOM is connected to the internet.

3.2 Functional Description

3.2.1 Use Case Diagram



Picture 1. Use Case Diagram

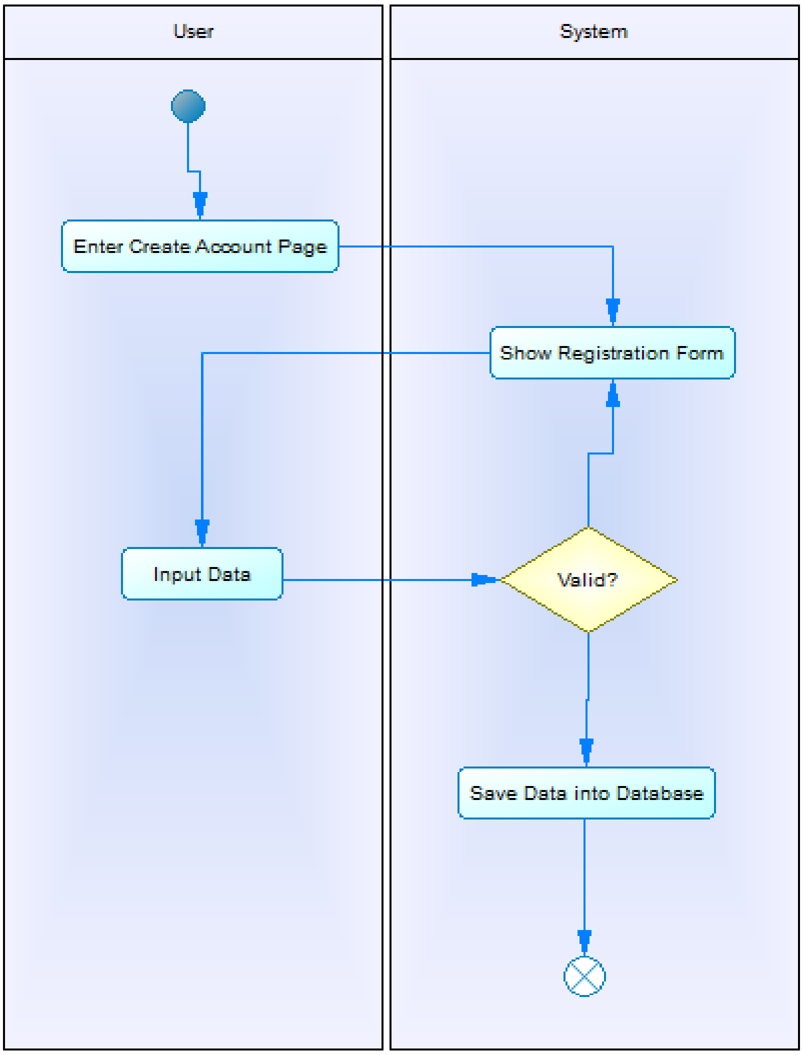
3.2.2 Function 1: Create Account

3.2.2.1 Scenario: Create Account

Use Case Code	UC 001
Use Case Name	Create Account
Actor	User
Description	This use case is used when a user want to sign up on this information system so the user can borrow the room
Relation	
Precondition	User doesn’t have an account to borrow the room

Postcondition	User have a registered account on the database of this information system
Normal Flow	
Actor	System
1. User get into the create account page	
	2. System showing the register form that must be filled by the user
3. User input their name, NRP, email, phone number, address	
	4. System checking the validity of user input A1. The inputs is not valid 5. System saving user input to the database 6. Finish
Alternative Flow	
A1. The input is not valid	
Actor	System
	A1.1 System display a notification that the input is not valid
A1.2 Use receive a notification that the input is not valid	
	A1.3 Back to number 2

3.2.2.2 Activity Diagram: Create Account



Picture 2. Activity Diagram “Create Account”

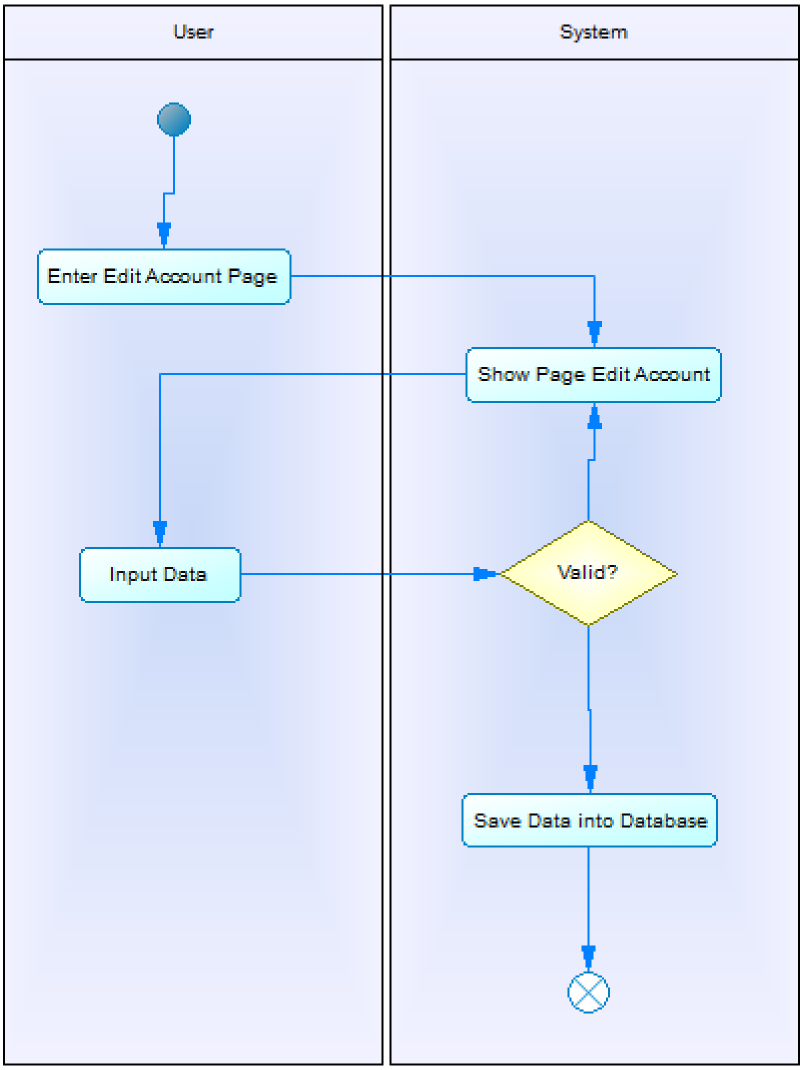
3.2.3 Function 2: Edit Account

3.2.3.1 Scenario: Edit Account

Use Case Code	UC 002
Use Case Name	Edit Account
Actor	User
Description	This use case describe how to edit an account
Relation	
Precondition	User already have an account on the database
Postcondition	User account updated with a new data
Normal Flow	
Actor	System
1. User enter to the edit account page	

3. User change the value of their previous input	2. System display a page that showing user’s account detail 4. System checking the validity of user’s input A1. The input is not valid 5. System update user’s data on the database 6. Finish
Alternative Flow	
A1. The input is not valid	
Actor	System
A1.2 User receive a notification that the input is not valid, user have to press ‘OK’ button on the pop up to close the notification	A1.1 System showing a notification to that user input is not valid pop up A1.3 Back to number 2

3.2.3.2 Activity Diagram: Edit Account



Picture 3. Activity Diagram "Edit Account"

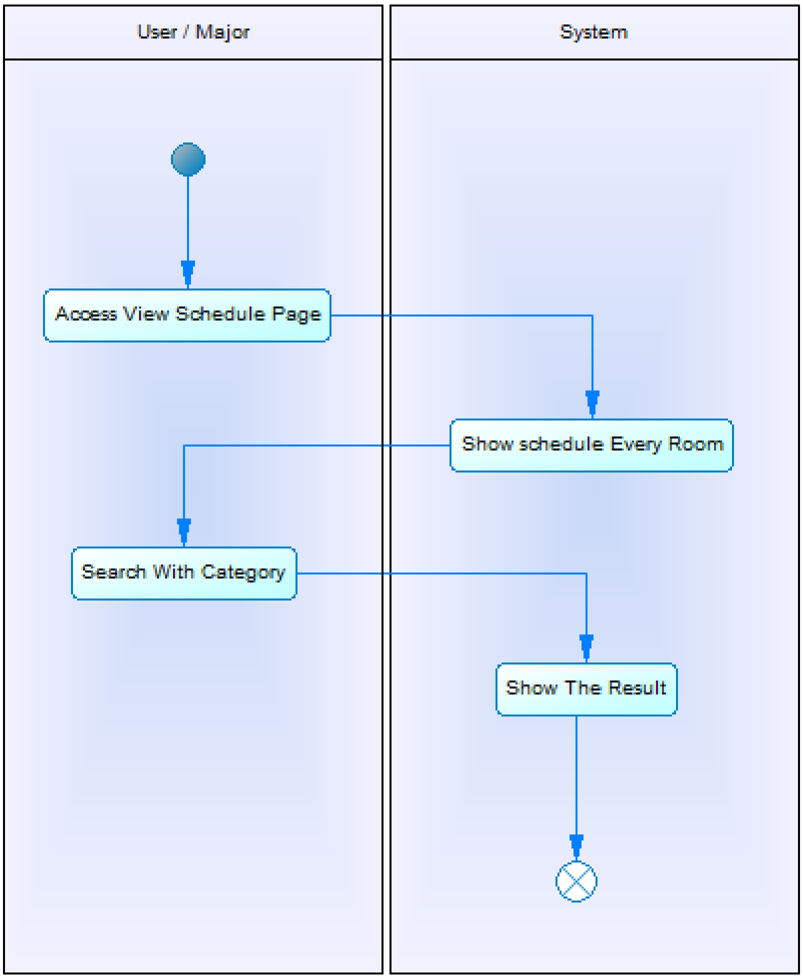
3.2.4 Function 3: View Schedule

3.2.4.1 Scenario: View Schedule

Use Case Code	UC 003
Use Case Name	View Schedule
Actor	User or Major
Description	This use case describe how a user or major can see all the schedule that show when a room is available to be booked and when a room is reserved by someone
Relation	

Precondition	User or major have already login before
Postcondition	User and major can see all the room schedule which is divided based on already booked room, waiting for confirmation, and still available
Normal Flow	
Actor	System
1. User / major accessing the view schedule page 3. User / major can search and filter the schedule base on time, room type, and availability	2. System show the schedule of every room 4. System show all the room that match with search category 5. Finish

3.2.4.2 Activity Diagram: View Schedule



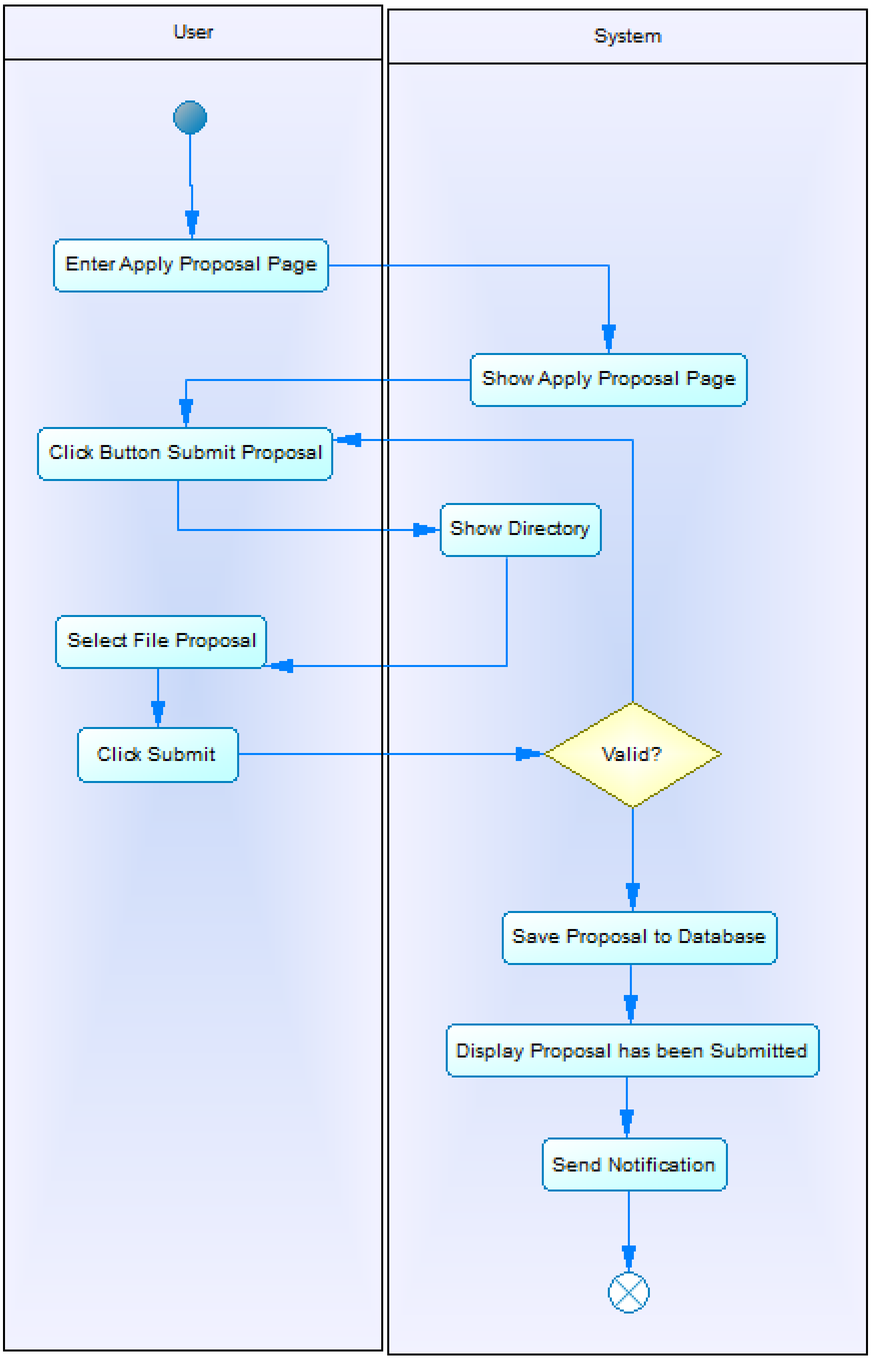
Picture 4. Activity Diagram "View Schedule"

3.2.5 Function 4: Apply Proposal Music Studio

3.2.5.1 Scenario: Apply Proposal Music Studio

Use Case Code	UC 004
Use Case Name	Apply Proposal Music Studio
Actor	User
Description	This use case describe how a user can apply proposal and asking for permission to the head of major and administrator
Relation	Include
Precondition	User already see the schedule and the user has not yet borrowed the room
Postcondition	Proposal for borrowing room has been sent to the head major
Normal Flow	
Actor	System
1. User enter to the 'apply proposal' page	
	2. System show the detail of selected room
3. User clicking 'submit proposal' button	
	4. System provide an 'upload proposal' pop up to search proposal on local directory
5. User select the proposal from local directory to upload 6. User clicking 'submit' button	
	7. Checking the proposal extention and size of the document A1. File don't match with the criteria 8. Saving the proposal to the database 9. Showing to the user that proposal has been submitted 10. Give notification to the head of major and the administrator 11. Finish
Alternative Flow	
A1. File do not match with the criteria	
	A1.1 System give a notification to the user through a pop up
A1.2 User press the 'ok' button to close the notification	
	A1.3 Back to number 3

3.2.5.2 Activity Diagram: Apply Proposal Music Studio


User
System
Enter Apply Proposal Page
Show Apply Proposal Page
Click Button Submit Proposal
Show Directory
Select File Proposal
Click Submit
Valid?
Save Proposal to Database
Display Proposal has been Submitted
Send Notification

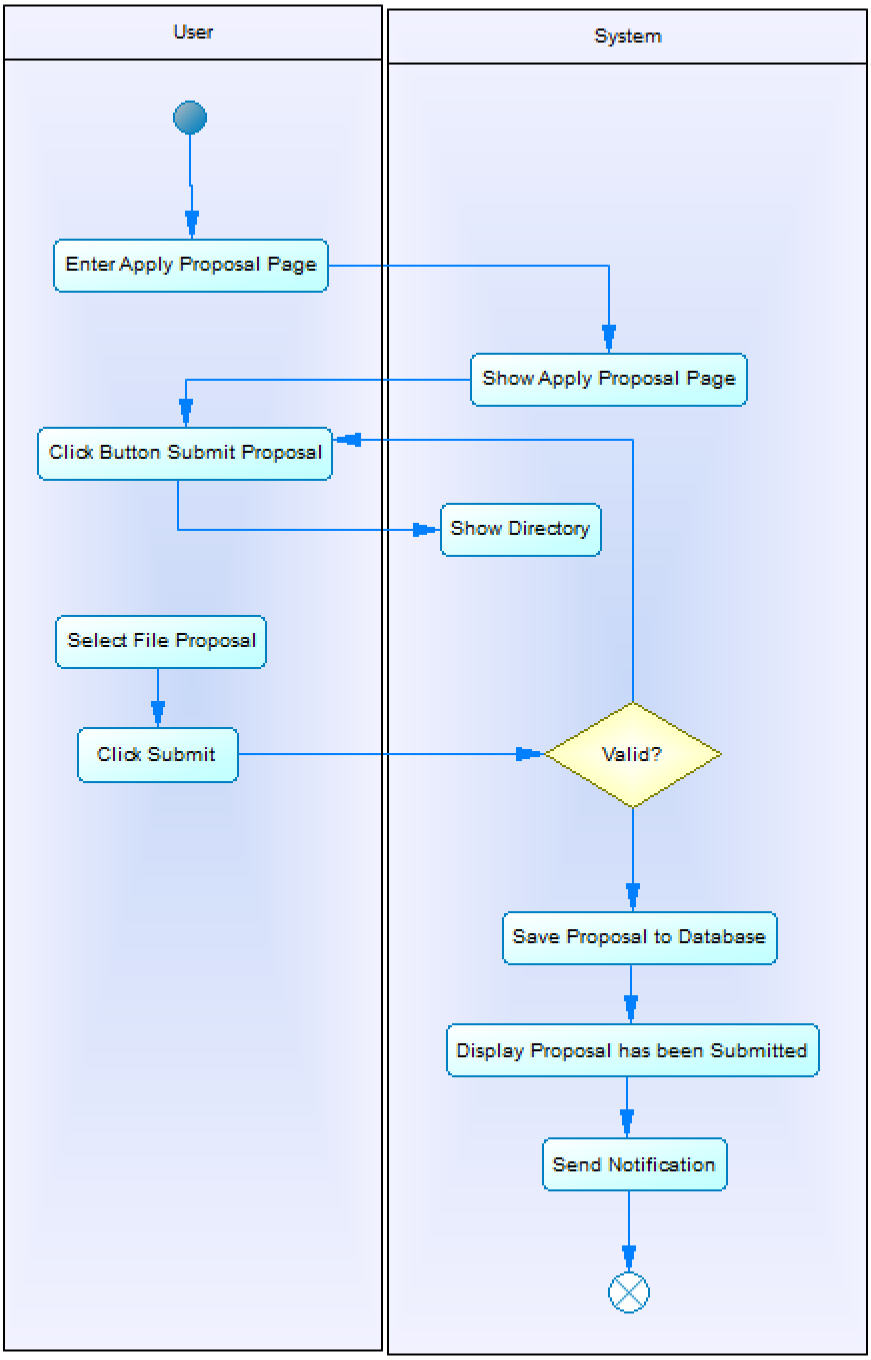
Picture 5. Activity Diagram "Apply Proposal Music Studio"

3.2.6 Function 5: Apply Proposal Classroom

3.2.6.1 Scenario: Apply Proposal Classroom

Use Case Code	UC 005
Use Case Name	Apply Proposal Classroom
Actor	User
Description	This use case describe how a user can apply proposal and asking for permission to the head of major and administrator
Relation	Include
Precondition	User already see the schedule and the user has not yet borrowed the room
Postcondition	Proposal for borrowing room has been sent to the head major
Normal Flow	
Actor	System
1. User enter to the 'apply proposal' page 3. User clicking 'submit proposal' button 5. User select the proposal from local directory to upload 6. User clicking 'submit' button	2. System show the detail of selected room 4. System provide an 'upload proposal' pop up to search proposal on local directory 7. Checking the proposal extention and size of the document A1. File don't match with the criteria 8. Saving the proposal to the database 9. Showing to the user that proposal has been submitted 10. Give notification to the head of major and the administrator 11. Finish
Alternative Flow	
A1. File do not match with the criteria	
A1.2 User press the 'ok' button to close the notification	A1.1 System give a notification to the user through a pop up A1.3 Back to number 3

3.2.6.2 Activity Diagram: Apply Proposal Classroom


User
System
Enter Apply Proposal Page
Show Apply Proposal Page
Click Button Submit Proposal
Show Directory
Select File Proposal
Click Submit
Valid?
Save Proposal to Database
Display Proposal has been Submitted
Send Notification

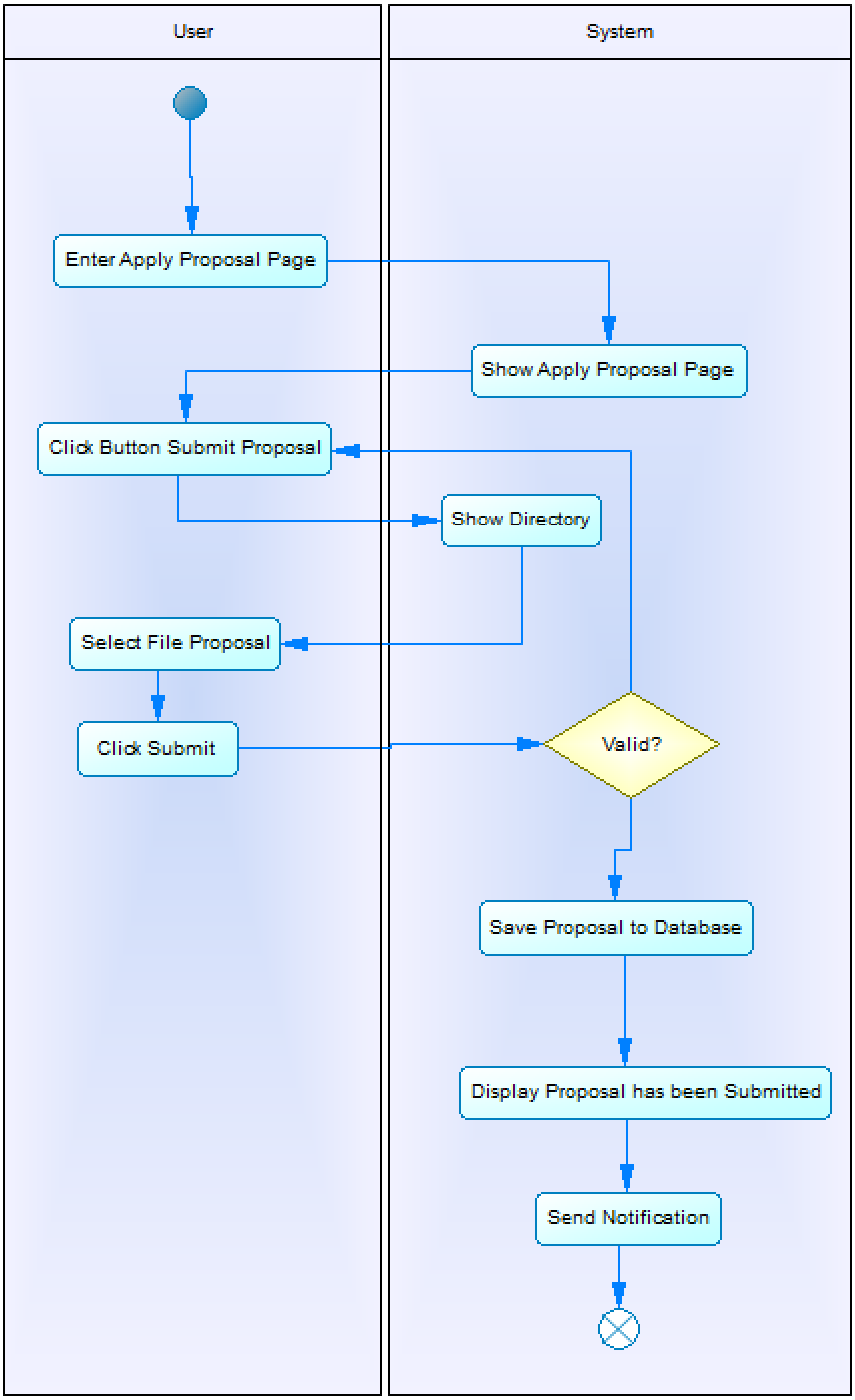
Picture 6. Activity Diagram "Apply Proposal Classroom"

3.2.7 Function 6: Apply Proposal Laboratory

3.2.7.1 Scenario: Apply Proposal Laboratory

Use Case Code	UC 006
Use Case Name	Apply Proposal Laboratory
Actor	User
Description	This use case describe how a user can apply proposal and asking for permission to the head of major and administrator
Relation	Include
Precondition	User already see the schedule and the user has not yet borrowed the room
Postcondition	Proposal for borrowing room has been sent to the head major
Normal Flow	
Actor	System
1. User enter to the 'apply proposal' page	
	2. System show the detail of selected room
3. User clicking 'submit proposal' button	
	4. System provide an 'upload proposal' pop up to search proposal on local directory
5. User select the proposal from local directory to upload	
	7. Checking the proposal extention and size of the document A1. File don't match with the criteria 8. Saving the proposal to the database 9. Showing to the user that proposal has been submitted 10. Give notification to the head of major and the administrator 11. Finish
Alternative Flow	
A1. File do not match with the criteria	
	A1.1 System give a notification to the user through a pop up
A1.2 User press the 'ok' button to close the notification	
	A1.3 Back to number 3

3.2.7.2 Activity Diagram: Apply Proposal Laboratory


User
System
Enter Apply Proposal Page
Show Apply Proposal Page
Click Button Submit Proposal
Show Directory
Select File Proposal
Click Submit
Valid?
Save Proposal to Database
Display Proposal has been Submitted
Send Notification

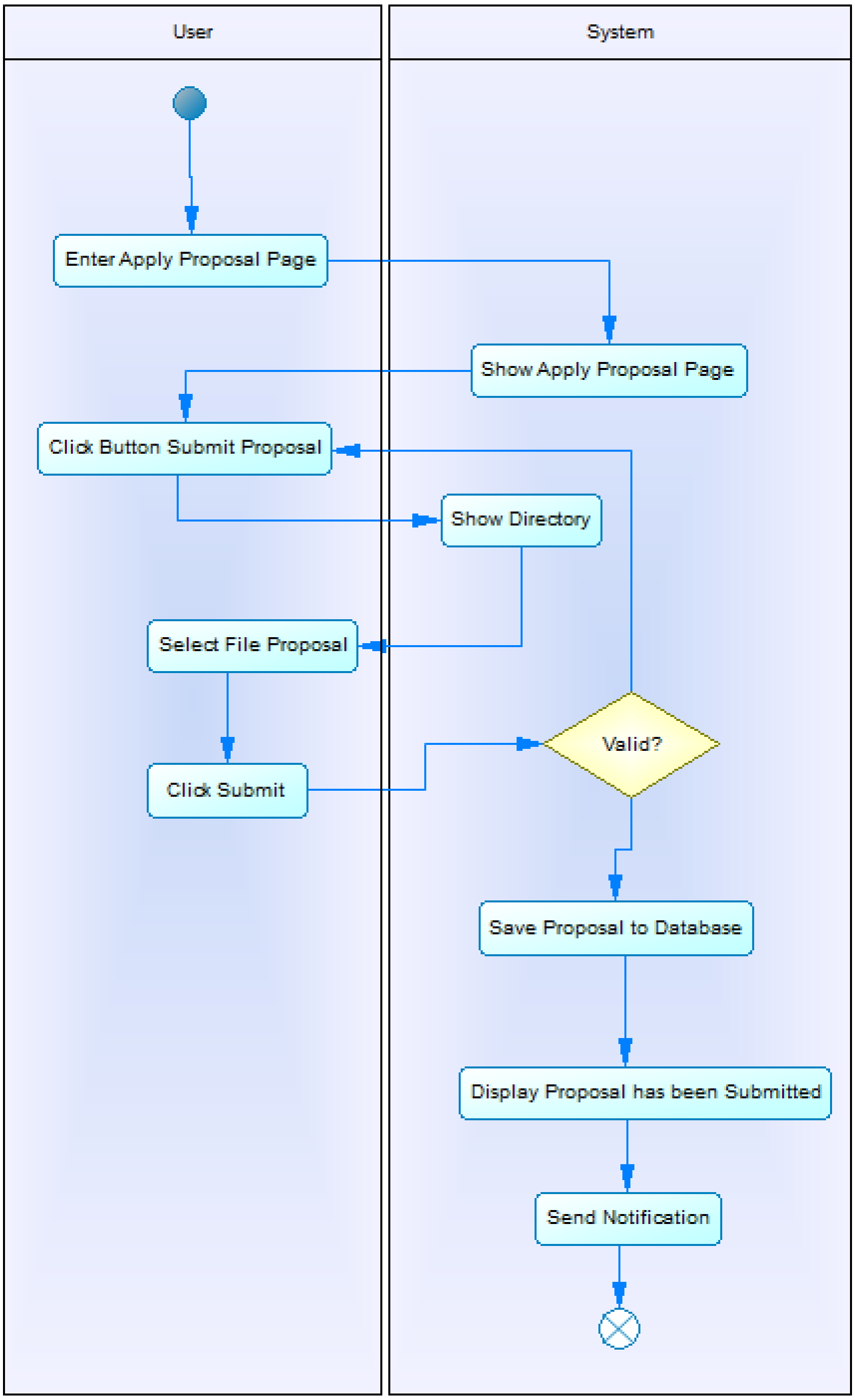
Picture 8. Activity Diagram “Apply Proposal Laboratory”

3.2.8 Function 7: Apply Proposal Hall

3.2.8.1 Scenario: Apply Proposal Hall

Use Case Code	UC 007
Use Case Name	Apply Proposal Hall
Actor	User
Description	This use case describe how a user can apply proposal and asking for permission to the head of major and administrator
Relation	Include
Precondition	User already see the schedule and the user has not yet borrowed the room
Postcondition	Proposal for borrowing room has been sent to the head major
Normal Flow	
Actor	System
1. User enter to the ‘apply proposal’ page	2. System show the detail of selected room
3. User clicking ‘submit proposal’ button	4. System provide an ‘upload proposal’ pop up to search proposal on local directory
5. User select the proposal from local directory to upload 6. User clicking ‘submit’ button	
	7. Checking the proposal extention and size of the document A1. File don’t match with the criteria 8. Saving the proposal to the database 9. Showing to the user that proposal has been submitted 10. Give notification to the head of major and the administrator 11. Finish
Alternative Flow	
A1. File do not match with the criteria	
	A1.1 System give a notification to the user through a pop up
A1.2 User press the ‘ok’ button to close the notification	
	A1.3 Back to number 3

3.2.8.2 Activity Diagram: Apply Proposal Hall



User
System
Enter Apply Proposal Page
Show Apply Proposal Page
Click Button Submit Proposal
Show Directory
Select File Proposal
Click Submit
Valid?
Save Proposal to Database
Display Proposal has been Submitted
Send Notification

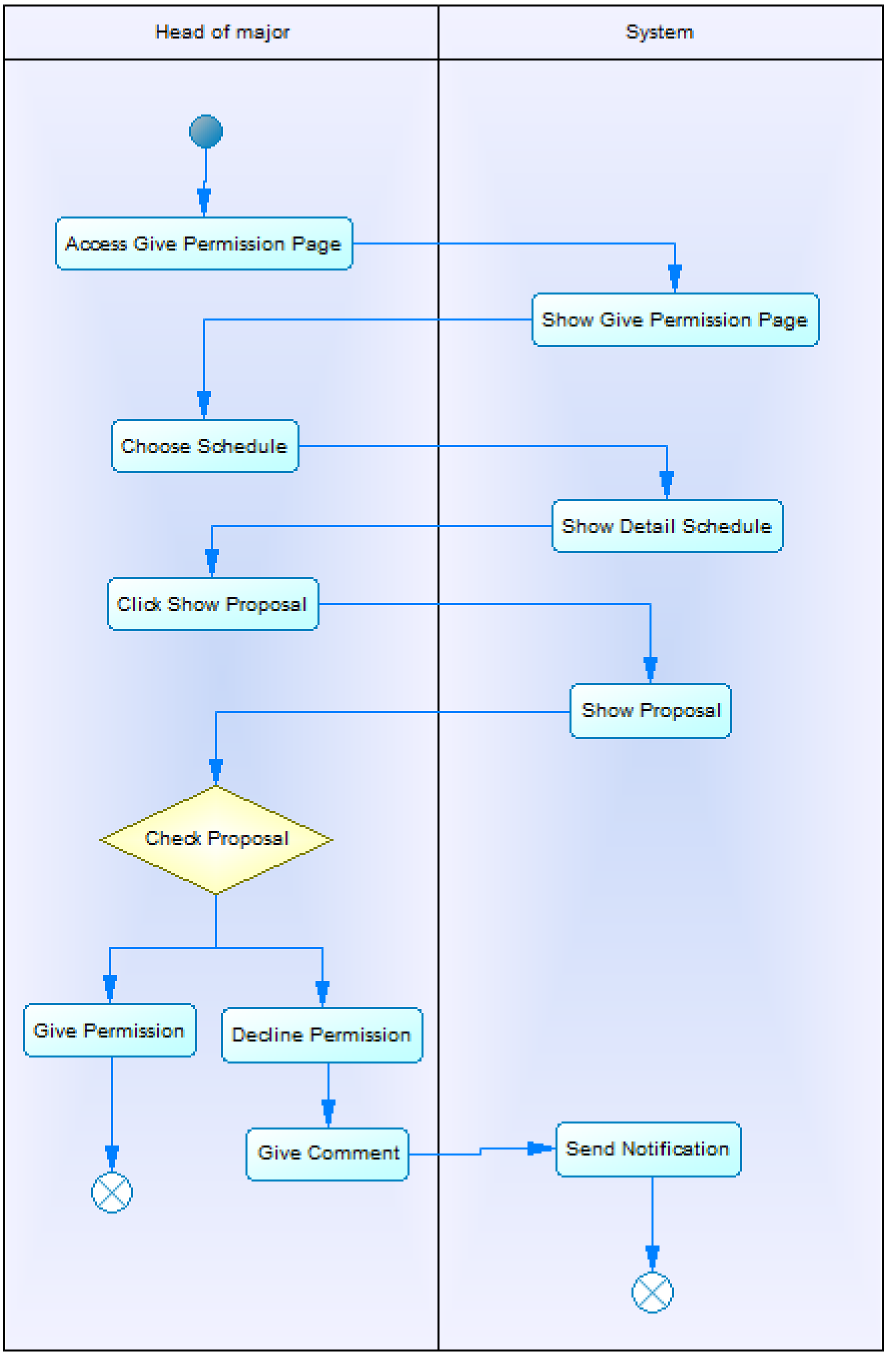
Picture 9. Activity Diagram “Apply Proposal Hall”

3.2.9 Function 8: Head of Major Give Permission

3.2.9.1 Scenario: Head of Major Give Permission

Use Case Code	UC 008
Use Case Name	Head of Major Give Permission
Actor	Head of Major
Description	This use case show how a permission from head of major is given to the user
Relation	-
Precondition	The head of major has been logged in and receive a notification that there is a reservation for a room waiting for confirmation
Postcondition	The head of major give his/her decision whether permitted or not
Normal Flow	
Actor	System
1. Head of major access the ‘give permission’ page	
	2. System showing all schedule that has been reserved by the user
3. Head of major choose a schedule to confirm	
	4. System showing the detail of selected schedule
5. Head of major clicking the ‘show proposal’ button	
	6. System directly show the proposal on the page
7. Head of major checking the proposal 8. Head of major give permission to the user A1. Permission declined	9. Finish
Alternative Flow	
A1. Permission declined	
Actor	System
A1.1 Head of major giving a reason or comentar to the user about why the proposal is declined	
	A1.2 System send a notification to the user that the proposal is declined A1.3 Finish

3.2.9.2 Activity Diagram: Head of Major Give Permission



Picture 10. Activity Diagram “Head of Major Give Permission”

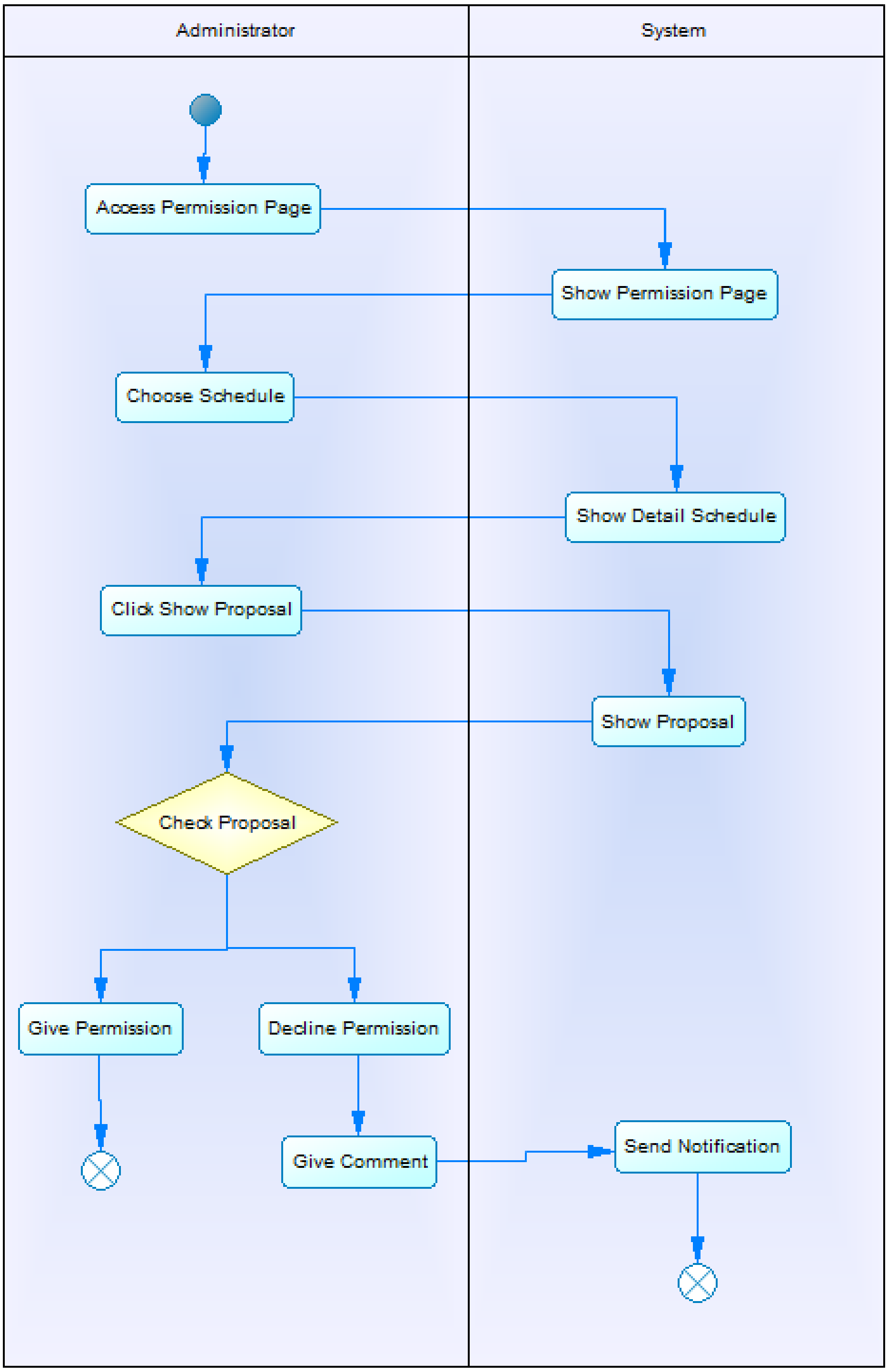
3.2.10 Function 10: Administrator Give Permission

3.2.10.1 Activity Diagram: Administrator Give Permission

Use Case Code	UC 009

Use Case Name	Administrator Give Permission
Actor	Administrator
Description	This use case describe how administrator give a permission to the user to reserve a room
Relation	-
Precondition	Head of major has given the permission
Postcondotion	Administrator give a decision to the user whether the permission is granted or declined
Normal Flow	
Actor	System
1. Administrator access the ‘give permission’ page	
	2. System showing all schedule that has been confirmed by the head of major
3. Administrator choose a schedule that has been confirmed by head of major to confirm	4. System showing the detail of selected schedule
5. Administrator clicking the ‘show proposal’ button	6. System directly show the proposal on the page
7. Administrator checking the proposal 8. Administrator give permission to the user A1. Permission declined	8. Finish
Alternative Flow	
A1. Permission declined	
Actor	System
A1.1 Administrator giving a reason or comentar to the user about why the proposal is declined	
	A1.2 System send a notification to the user that the proposal is declined A1.3 Finish

3.2.10.2 Activity Diagram: Administrator Give Permission



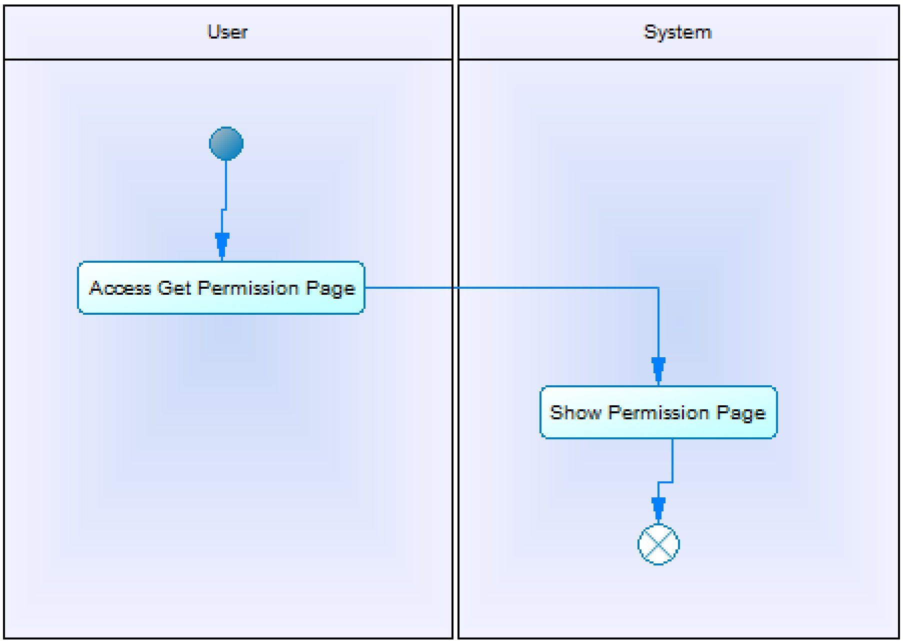
Picture 11. Activity Diagram “Administrator Give Permission”

3.2.11 Function 10: Get Permission

3.2.11.1 Scenario: Get Permission

Use Case Code	UC 010
Use Case Name	Get Permission
Actor	User
Description	This use case describe how user get a permission to reserve a room
Relation	-
Precondition	Head of major and administrator has given the permission
Postcondition	User know that permission has granted
Normal Flow	
Actor	System
1. User access the 'get permission' page	2. System showing the reservation status 3. Finish
Alternative Flow	
-	

3.2.11.2 Activity Diagram: Get Permission



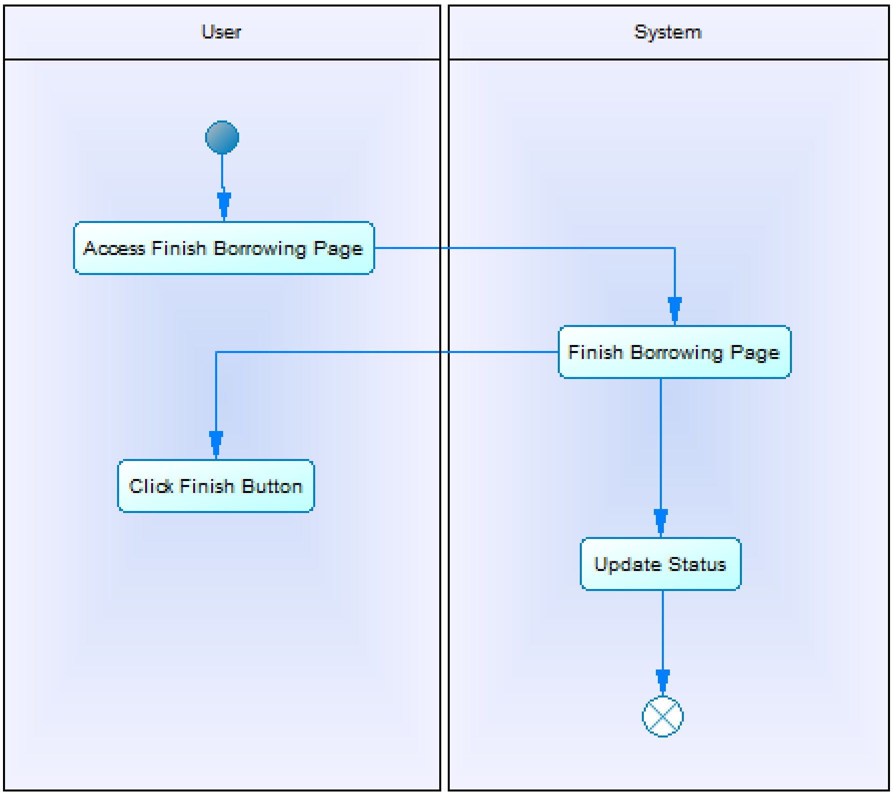
Picture 12. Activity Diagram "Get Permission"

3.2.12 Function 12: Finish Borrowing

3.2.12.1 Scenario: Finish Borrowing

Use Case Code	UC 011
Use Case Name	Finish Borrowing
Actor	User
Description	This use case describe how a user declare to finish the borrowing
Relation	-
Precondition	User has borrow the room
Postcondition	User do not have access to the room
Normal Flow	
Actor	System
1. User access the ‘finish borrowing’ page 3. User declare that the borrowing is over by clicking the ‘finish’ button	2. System showing ‘finish borrowing’ page 5. System update the status of reservation to ‘finish’ state 4. Finish
Alternative Flow	
-	

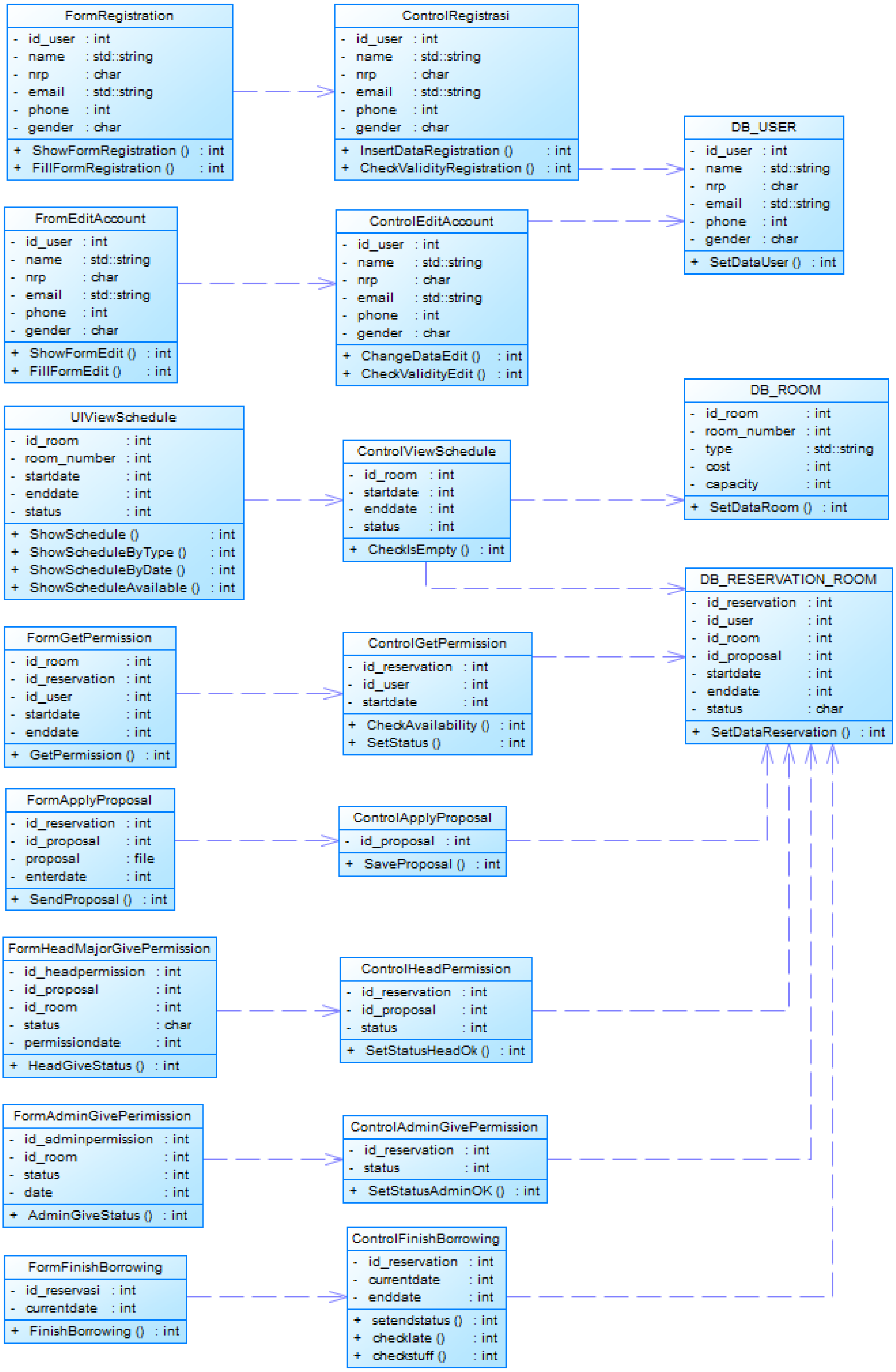
3.2.12.2 Activity Diagram: Finish Borrowing



Picture 13. Activity Diagram “Finish Borrowing”

3.3 Classes Description

3.3.1 Class Diagram



Picture 14. Class Diagram