Welcome To My Presentation

Presentation On
Development of Thesis Repository System

Prepared By

Md. Maruf Hosen

ID# 16203091

Department : BCSE

Supervised By

Rashedul Islam

Assistant Professor

Department of Computer Science and Engineering

Topics To Be Covered..

❑Project Introduction

❑Objectives

❑Software Process Model

❑Requirement Engineering

❑Use Case Diagram

❑Activity Diagram

❑Data Flow Diagram

❑Entity Relationship Diagram

❑Functions Of Proposed System

❑Functional Point Count

❑Effort Based Estimation

❑Testing

❑Conclusion

PROJECT INTRODUCTION

- ❑Student can store their thesis paper with proper security.
- ❑Prevent copy of the other’s work.
- ❑Provide a social media for the student.
- ❑Real time chatting.
- ❑Uploading post, images, videos and reacting system.
- ❑Easy to communicate with supervisor.
- ❑Management of all reports securely.

OBJECTIVES

- Store thesis paper
- Repository security
- Prevent the copy of other’s works
- Provide communication
- Provide a social media
- Paper downloading request system
- Upload post
- Image upload (profile, cover, others)

Software Process Model

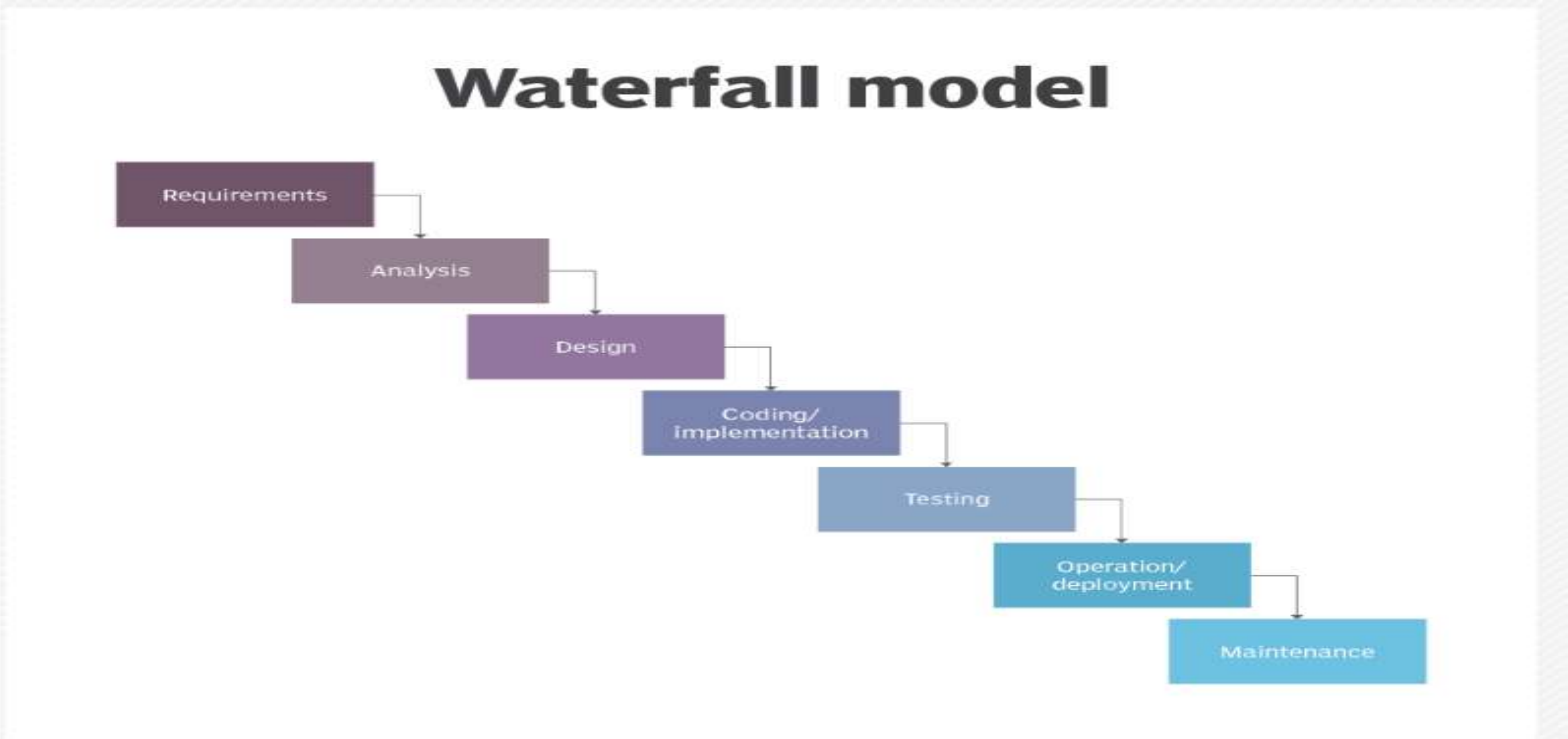


Figure: Waterfall Process Model

Benefits of This Model:

- ❑Waterfall model works well of smaller projects where requirements are very well understood.
- ❑Its process activities are clearly separated and organized.
- ❑All the requirements for this project are already known.
- ❑This model is simple and easy to understand and use.
- ❑Technology is understood.

Feasibility Study

- ❑Technical Feasibility
- ❑Economic Feasibility
- ❑Operational Feasibility

Requirement Engineering

❑User Requirements

❑System Requirements

❑Functional Requirements

❑Non-functional Requirements

User & System Requirements

- ❑Admin can add thesis category.
- ❑Admin can see all repository.
- ❑Admin can handle the downloading process.
- ❑Admin can remove user account (in case of offences).
- ❑Admin can update & delete category.
- ❑Admin can manage security.
- ❑Admin can generate repository reports.
- ❑Admin can block user.

- ❑ User can create account.
- ❑ User can manage their repository.
- ❑ User can post status.
- ❑ User can upload images and videos.
- ❑ User can see all user and can send friend request.
- ❑ User can engaged in real time chatting.
- ❑ User can update their profile (about, account, cover image).
- ❑ User can see others post, images, videos.
- ❑ User can give like comment to other post.
- ❑ User can remove their account.

Functional Requirements

- ❑ Admin can maintain whole system.
- ❑ Admin can add, update and delete Categories.
- ❑ Admin can see all repositories.
- ❑ Admin can block user account.
- ❑ Admin can remove user account.
- ❑ Admin can see all user and print user information.
- ❑ Admin can print the user list.
- ❑ Admin can disable whole system.
- ❑ Admin can print the individual repository information with its owner information.
- ❑ User can print their own repositories.
- ❑ User can downloads others works by taking permission from the admin.
- ❑ User can delete their account.

Non-functional Requirements

❑The System has a strong security factor. Without the valid user nobody can access this system.

❑Admin can log in by using username and password.

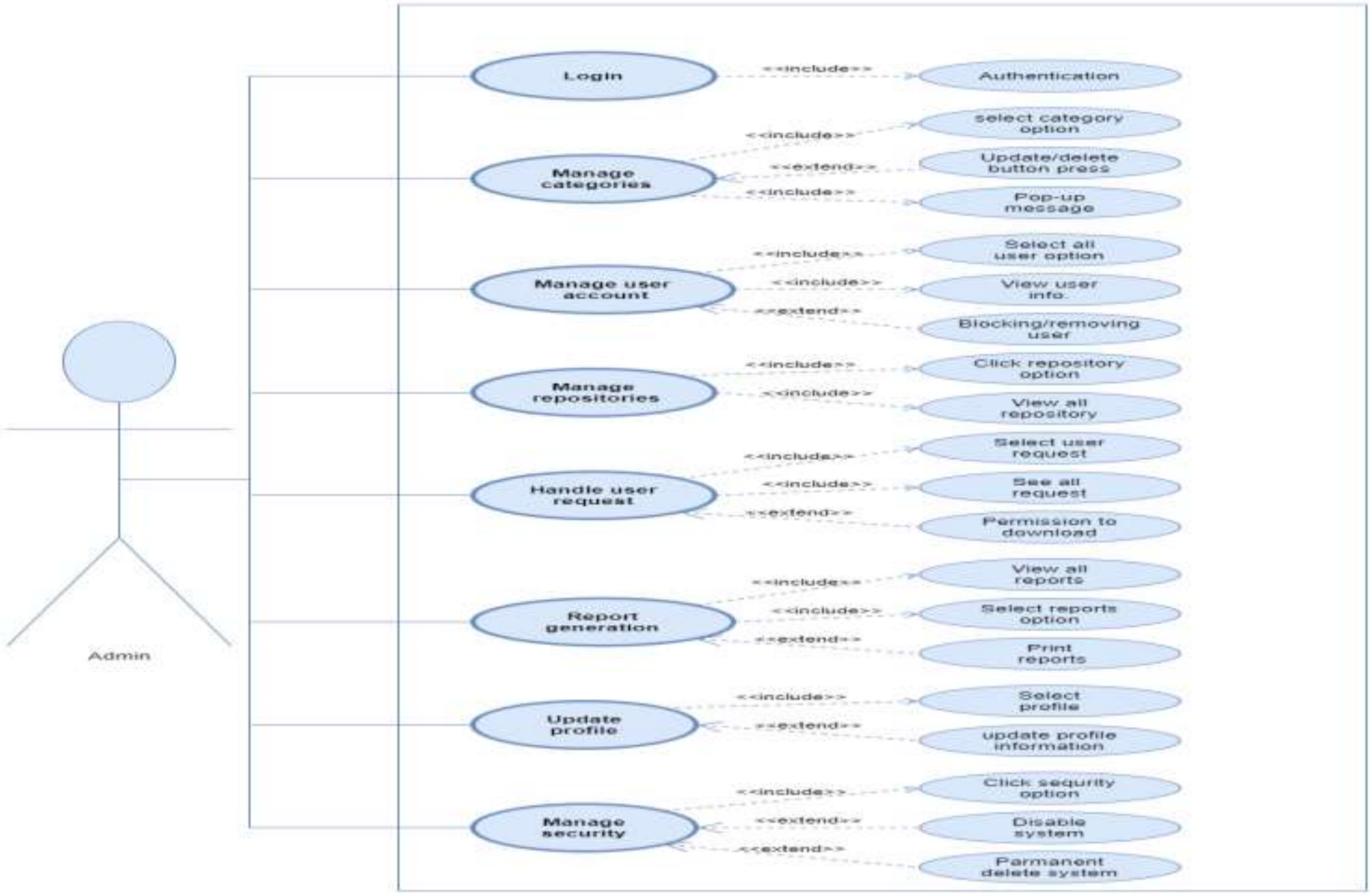
❑Admin can change his/her password.

❑Student can log in by using username and password.

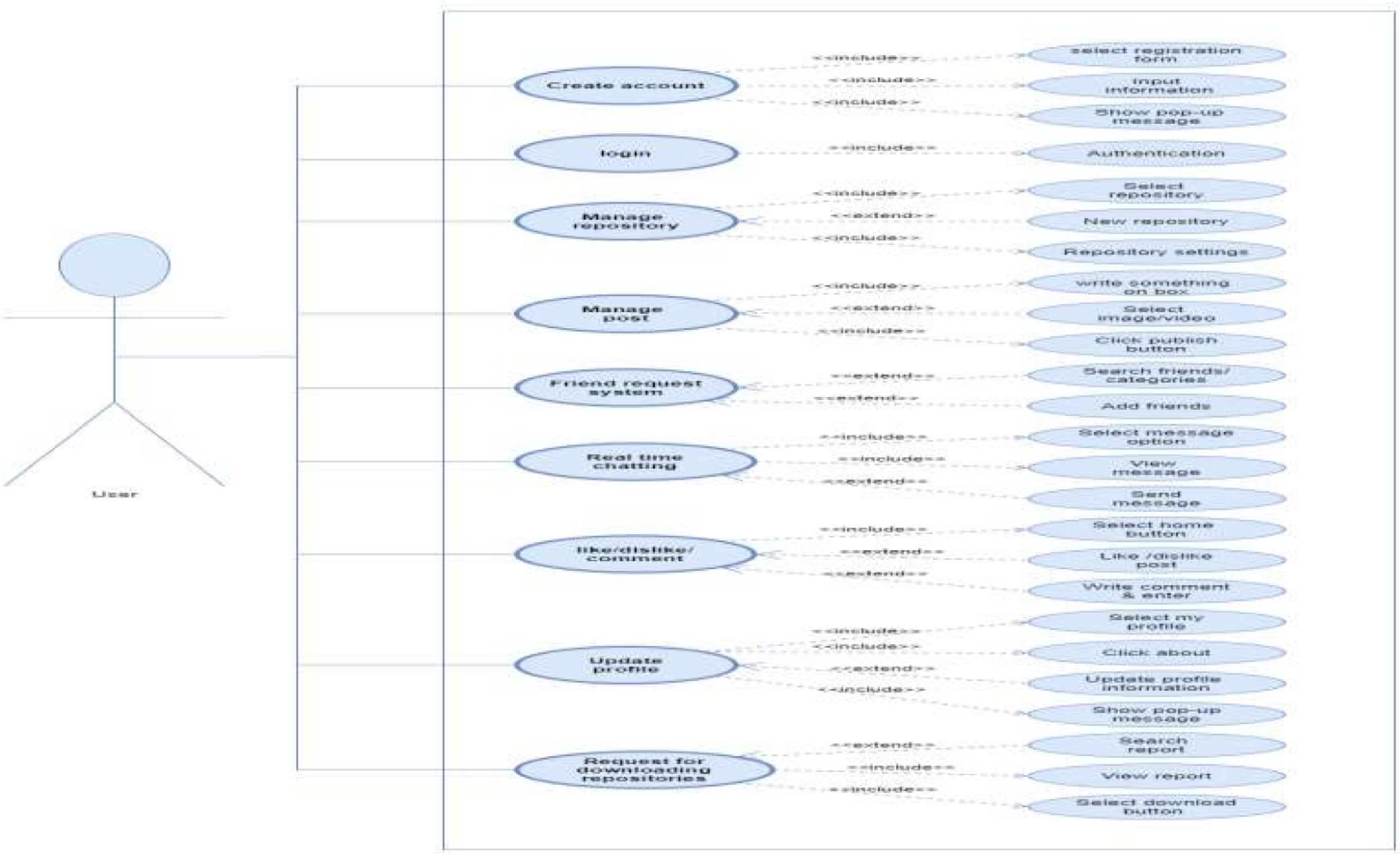
❑Only admin can maintain the whole system.

❑This system supports only Windows7/8/10.

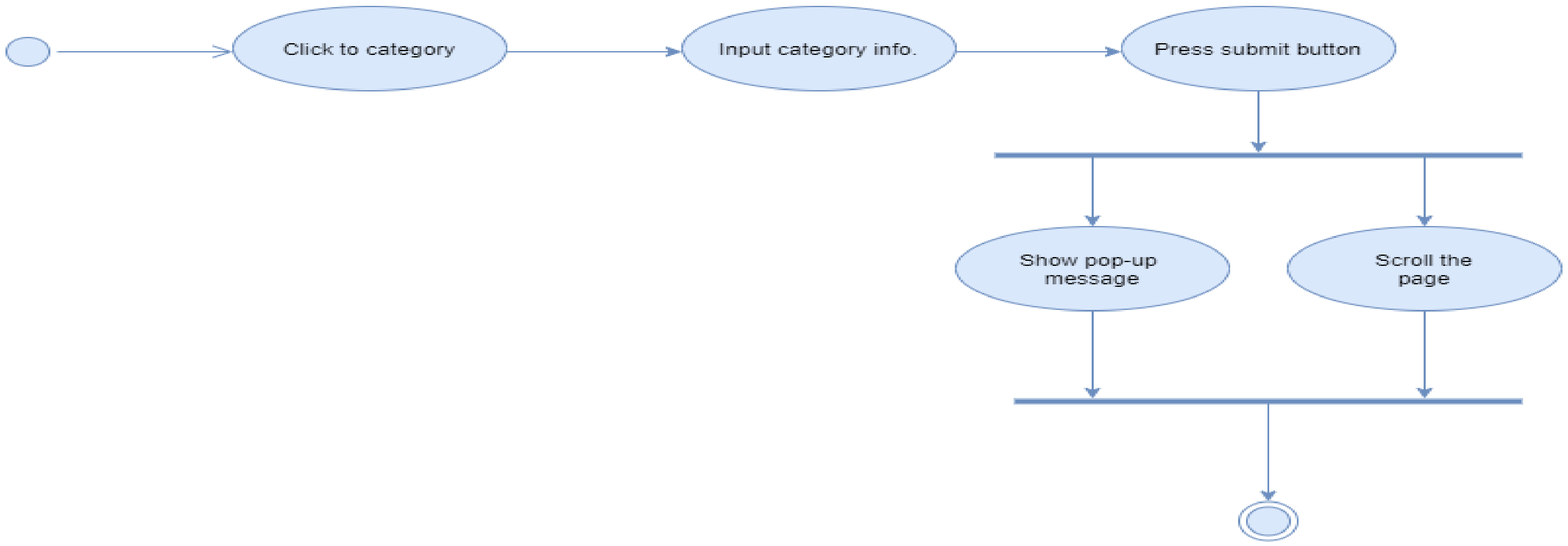
Use Case Diagram of Admin



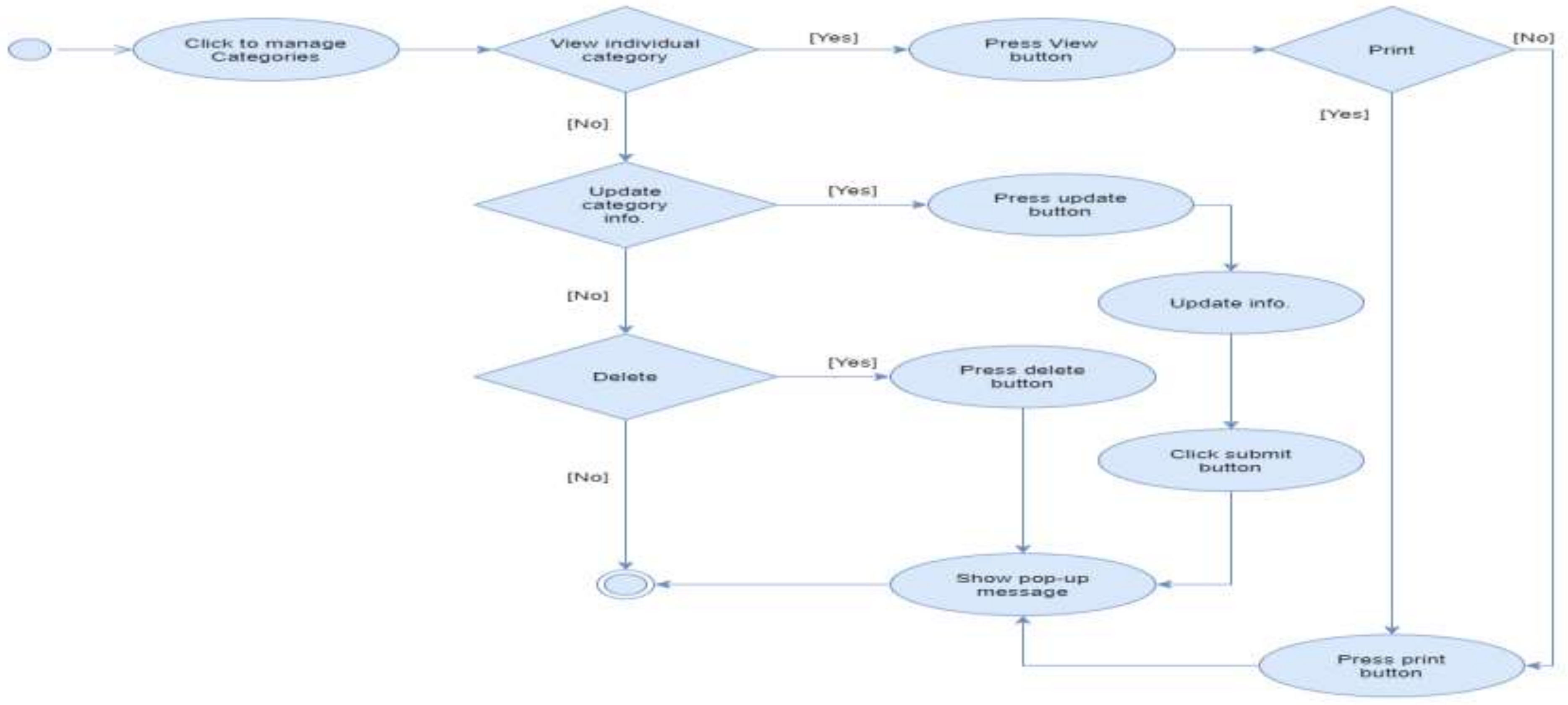
Use Case Diagram of User



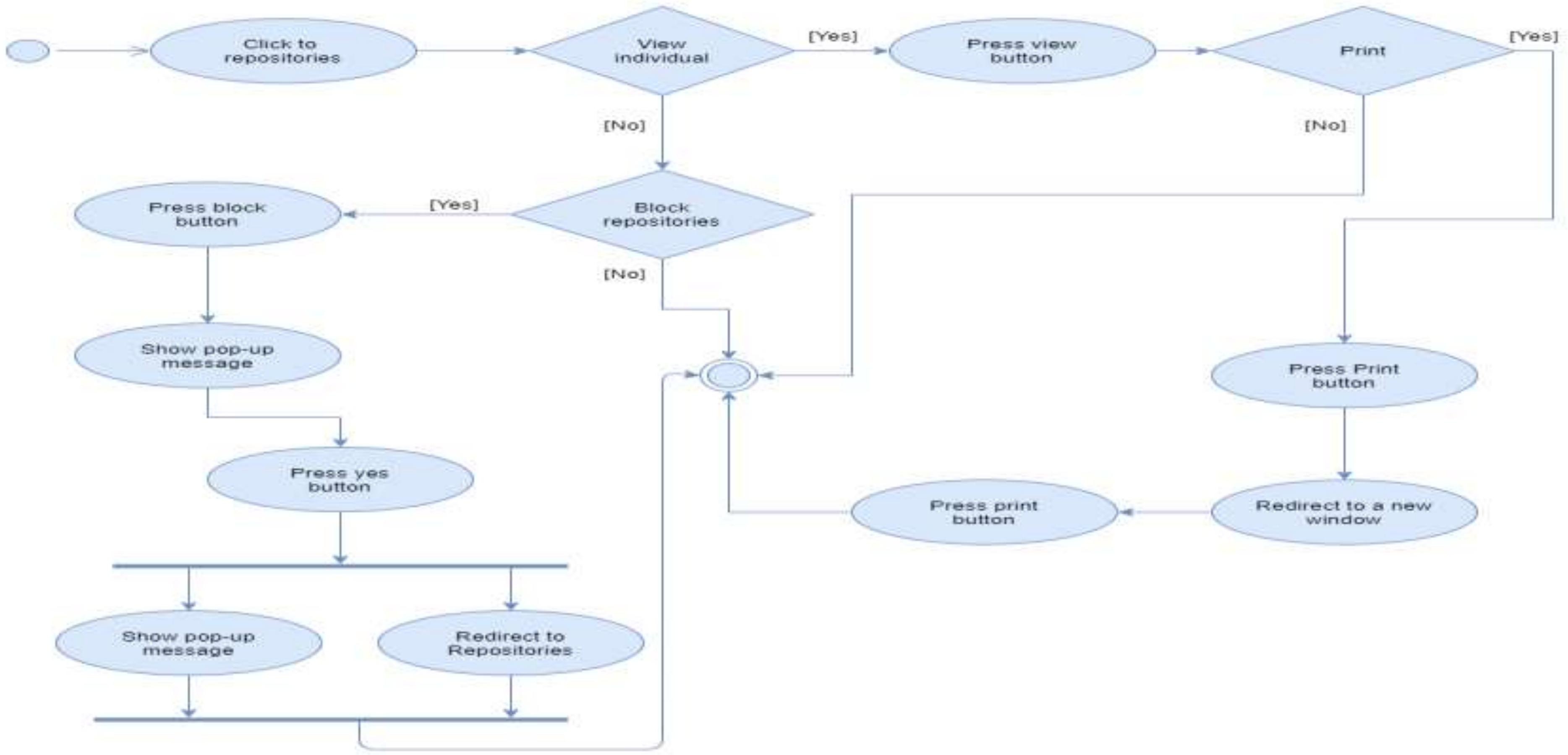
Activity Diagram of Adding Category Information



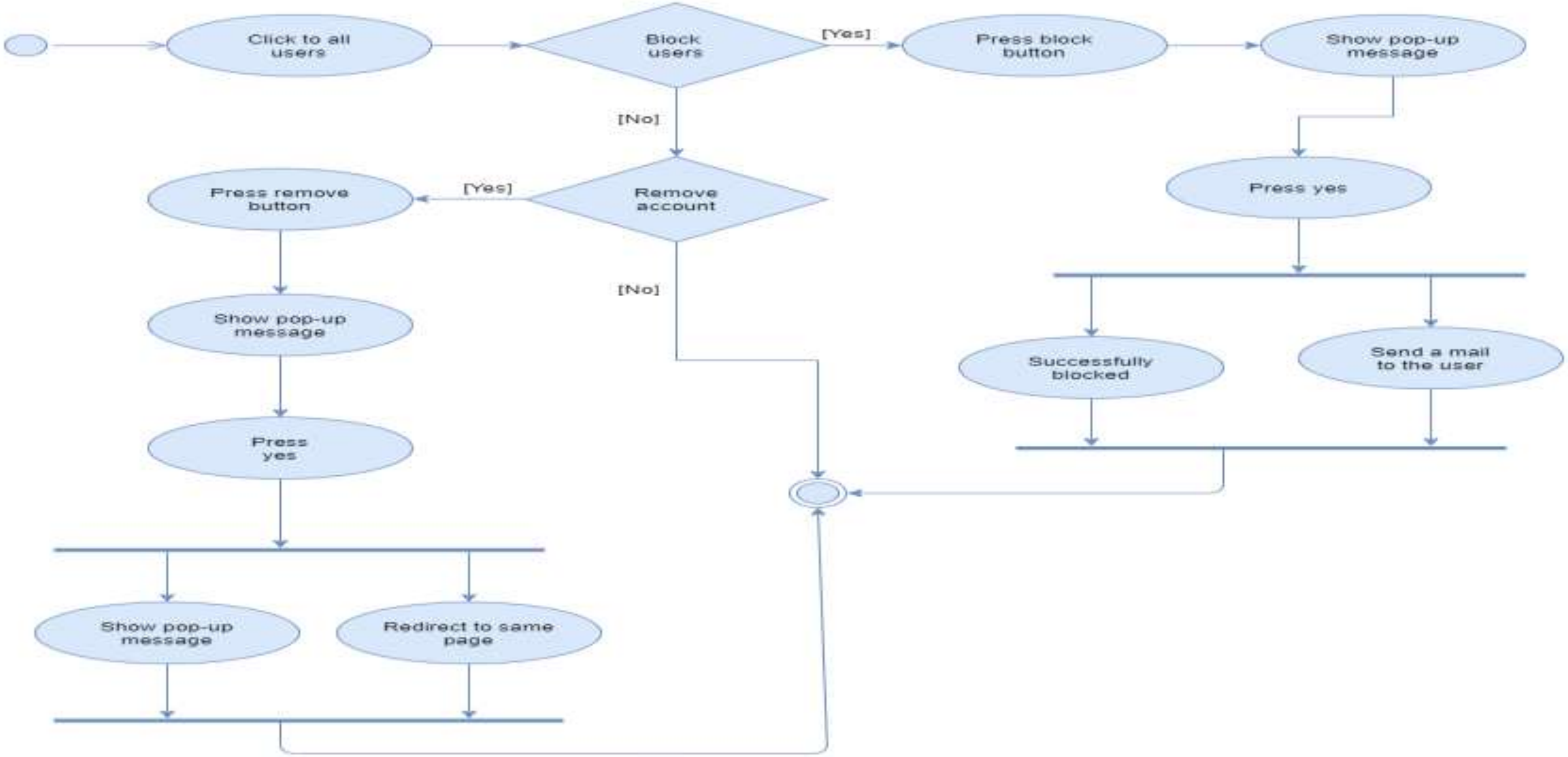
Activity Diagram of Category Management



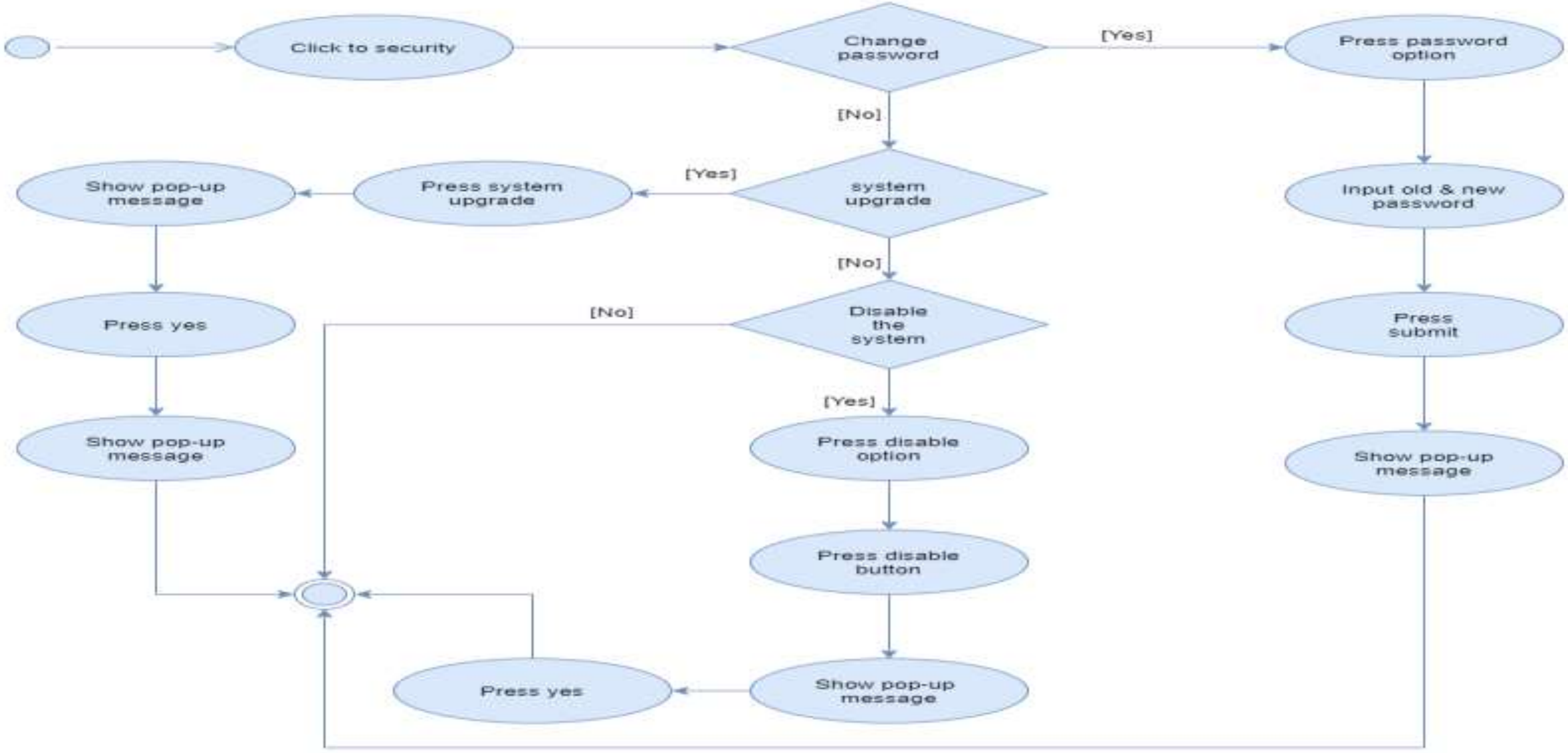
Activity Diagram of Management of Repository



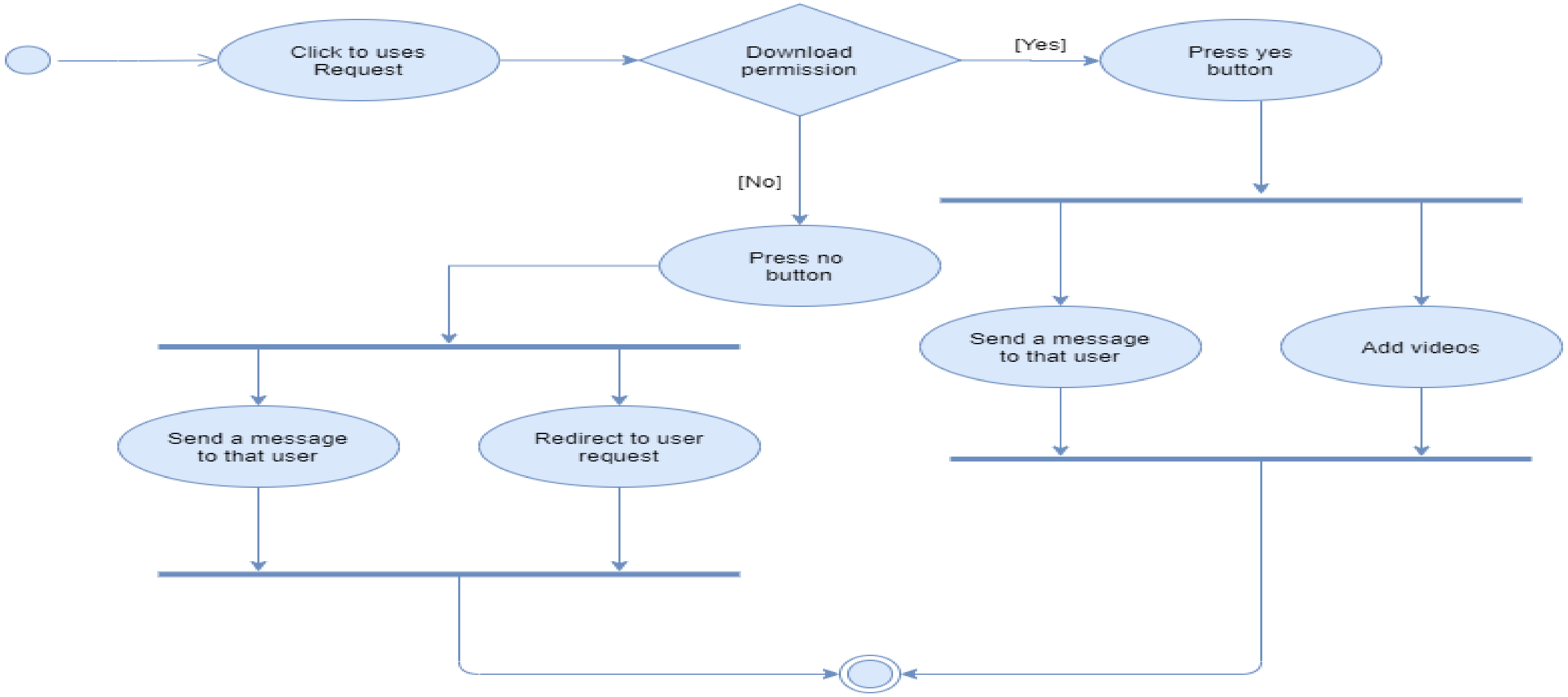
Activity Diagram for Manage User Account



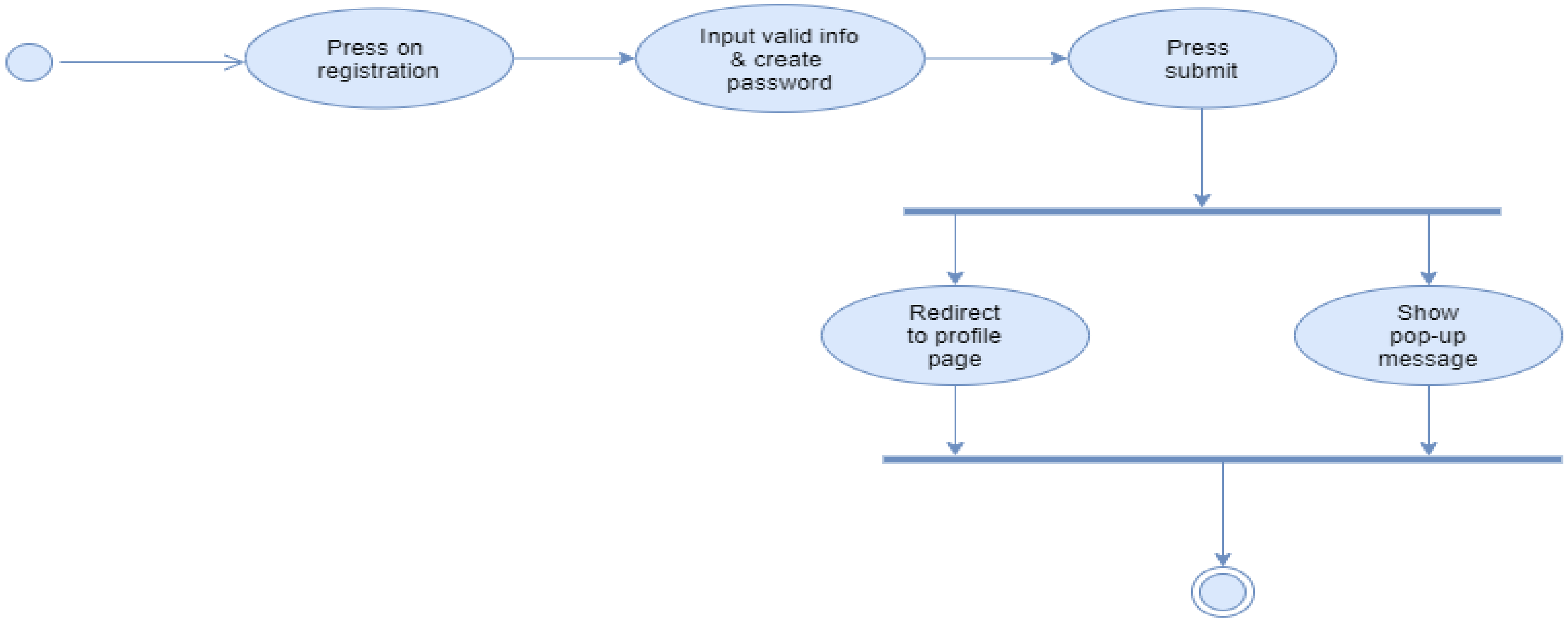
Activity Diagram of Security Management



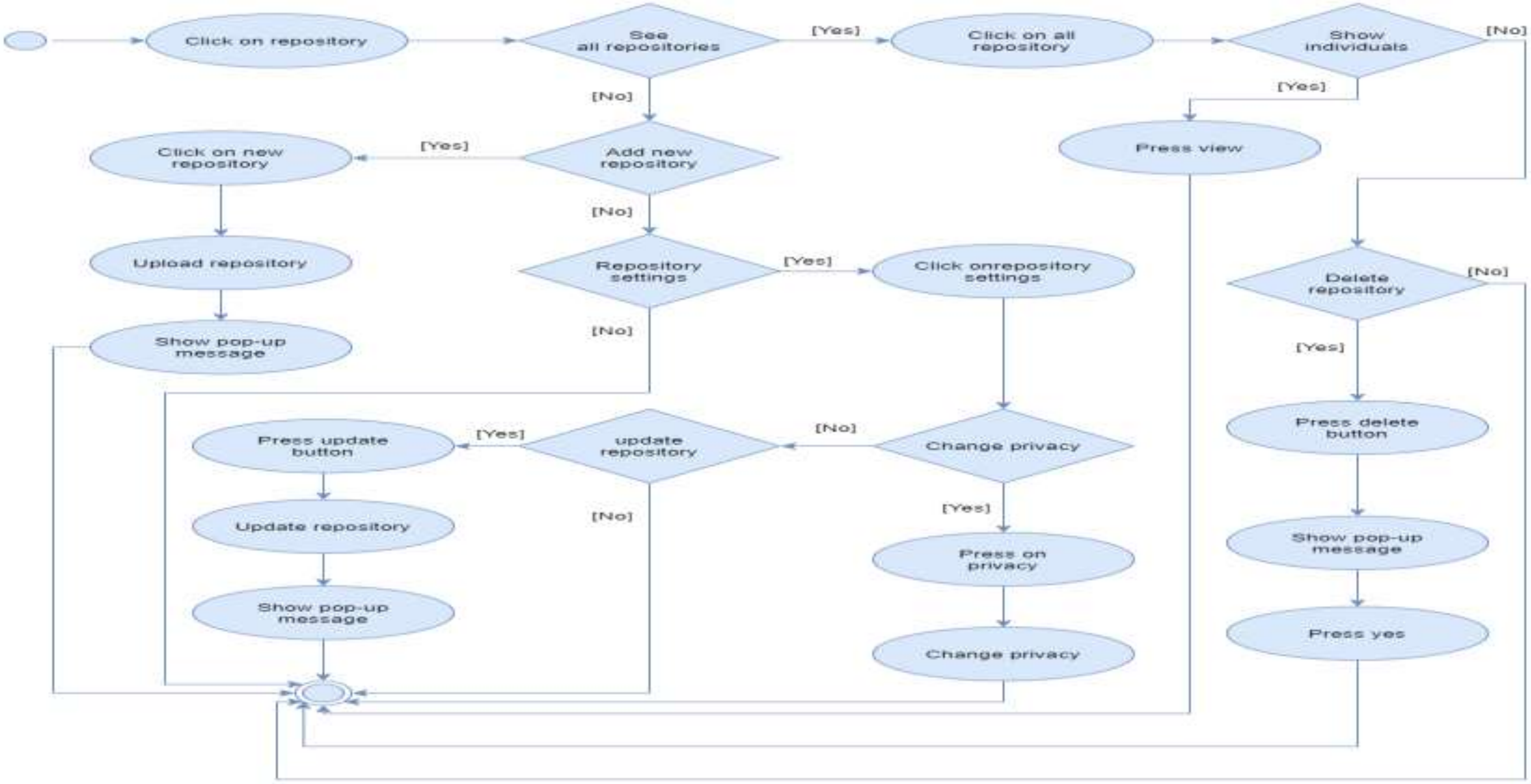
Activity Diagram of Handling Download Process



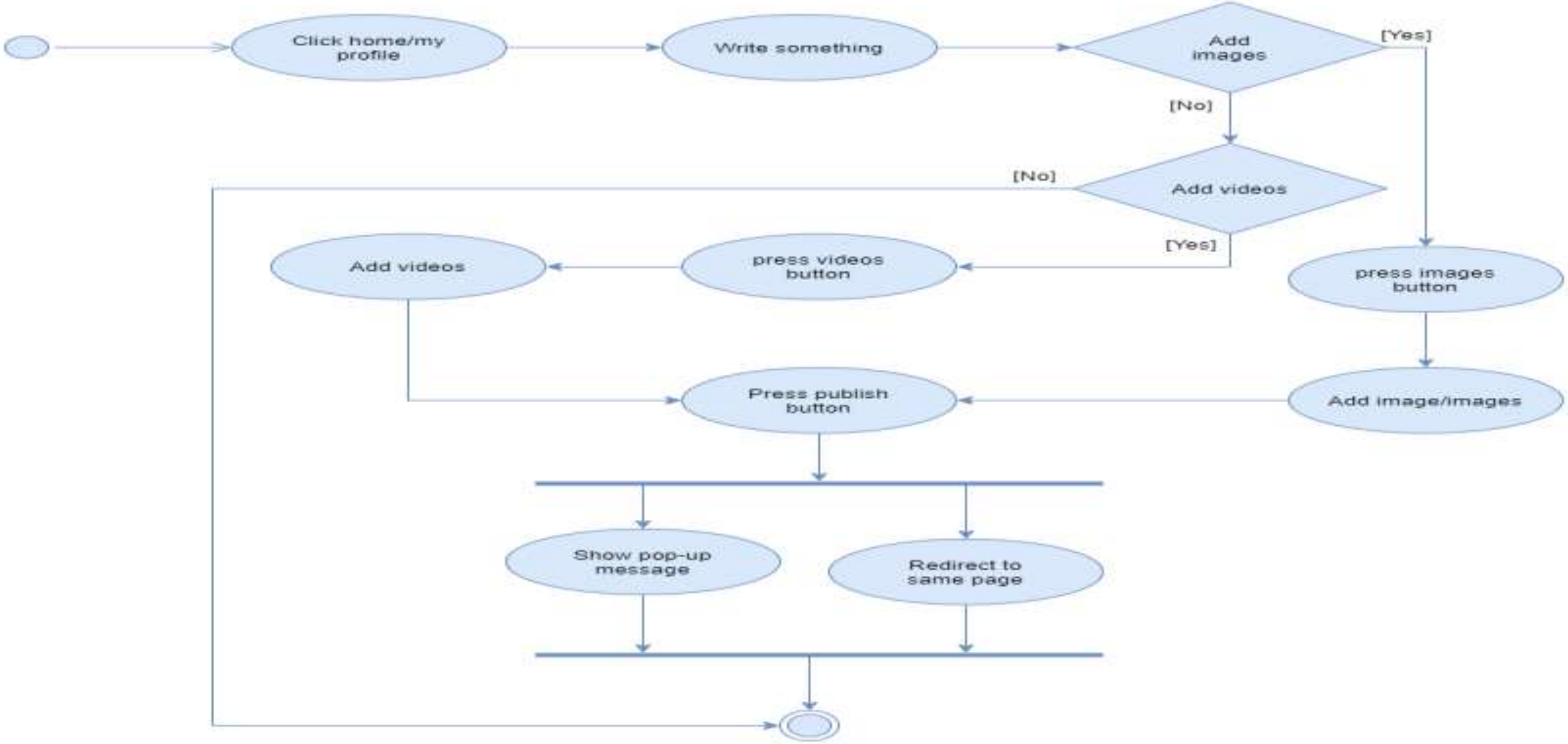
Activity Diagram for User Registration



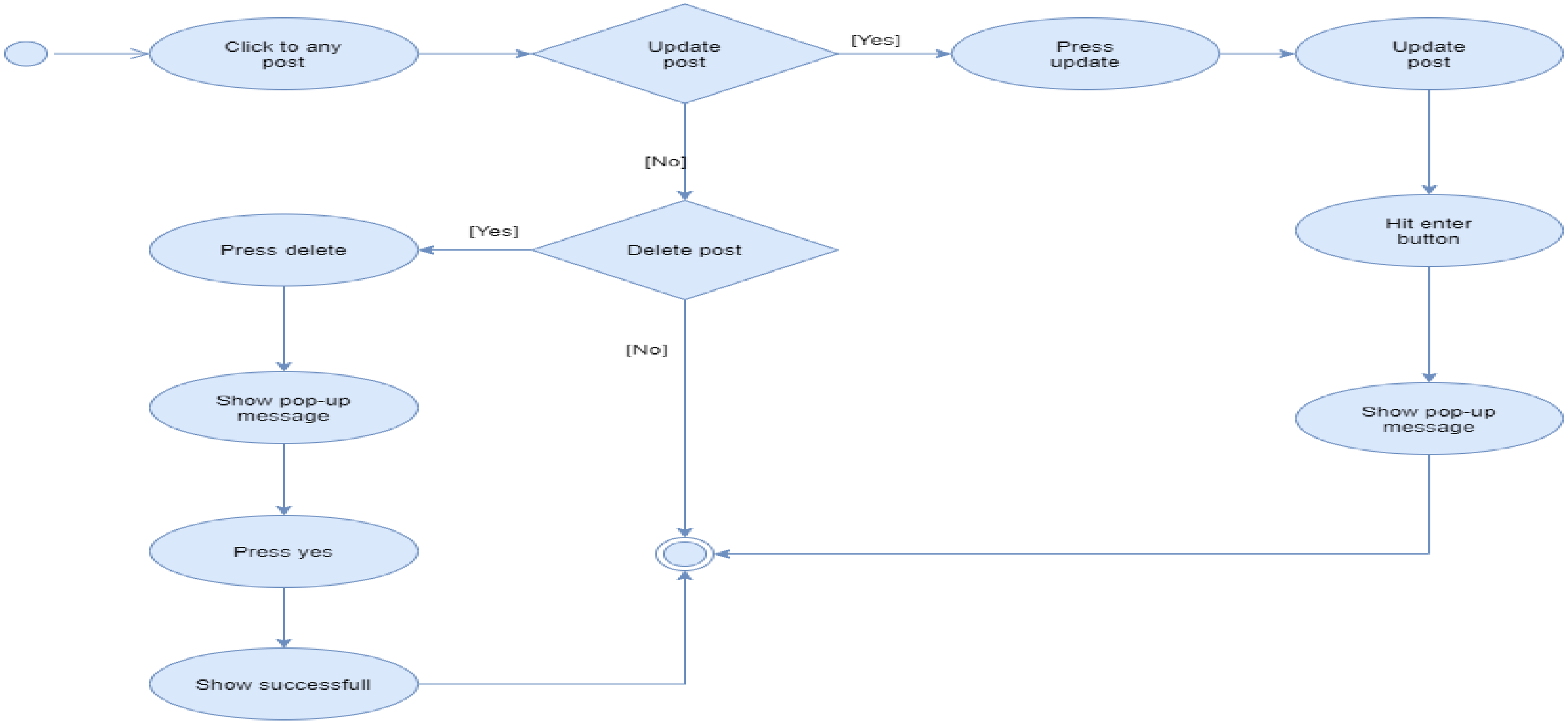
Activity Diagram of Repository Management



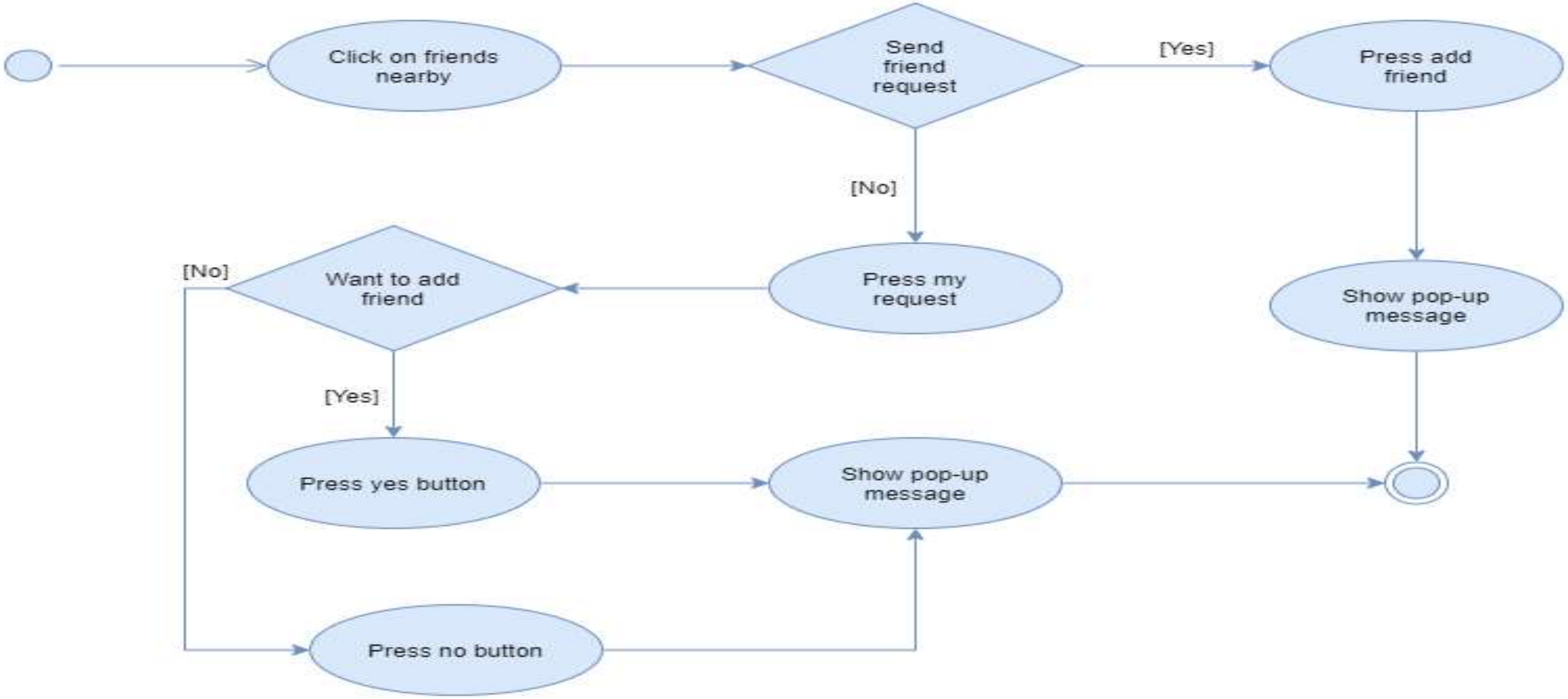
Activity Diagram of Uploading Post



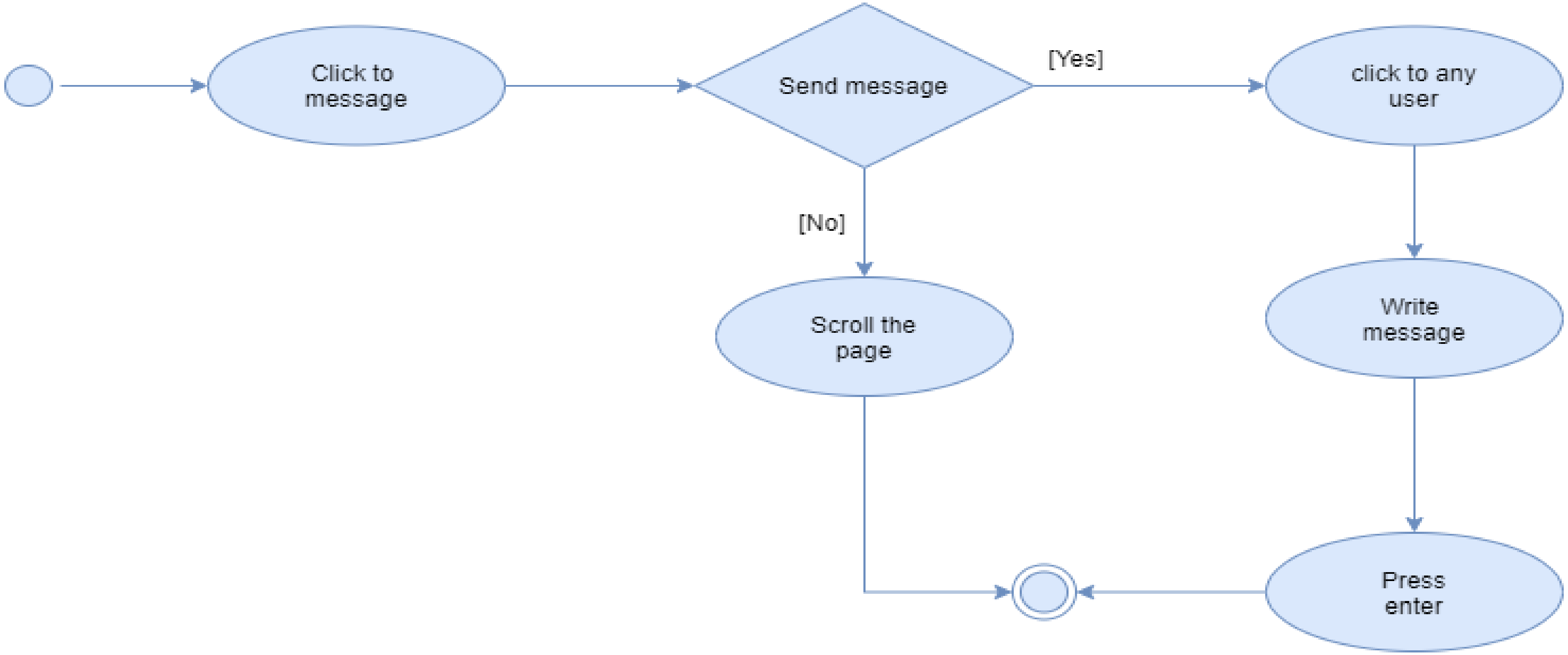
Activity Diagram of Post Management



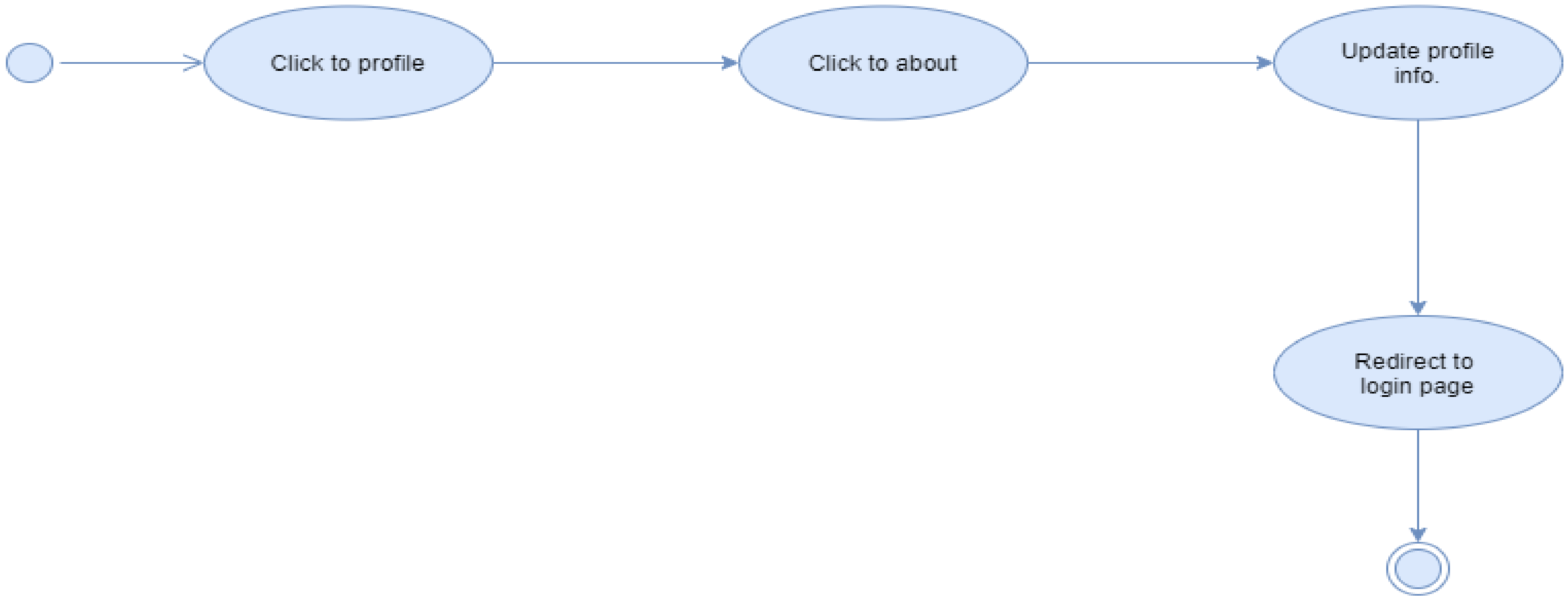
Activity Diagram of Friend Requesting System



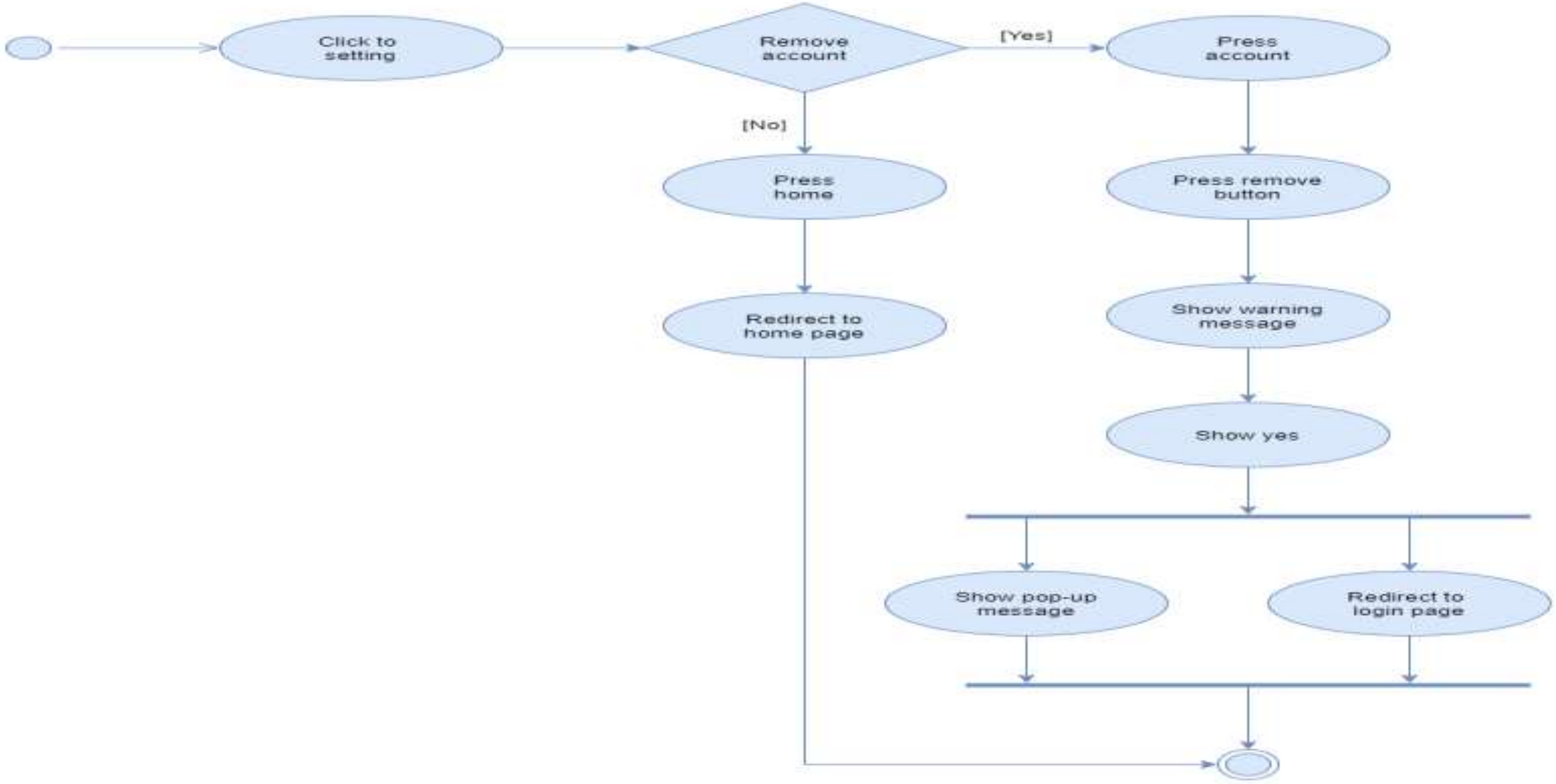
Activity Diagram of Real Time Chatting



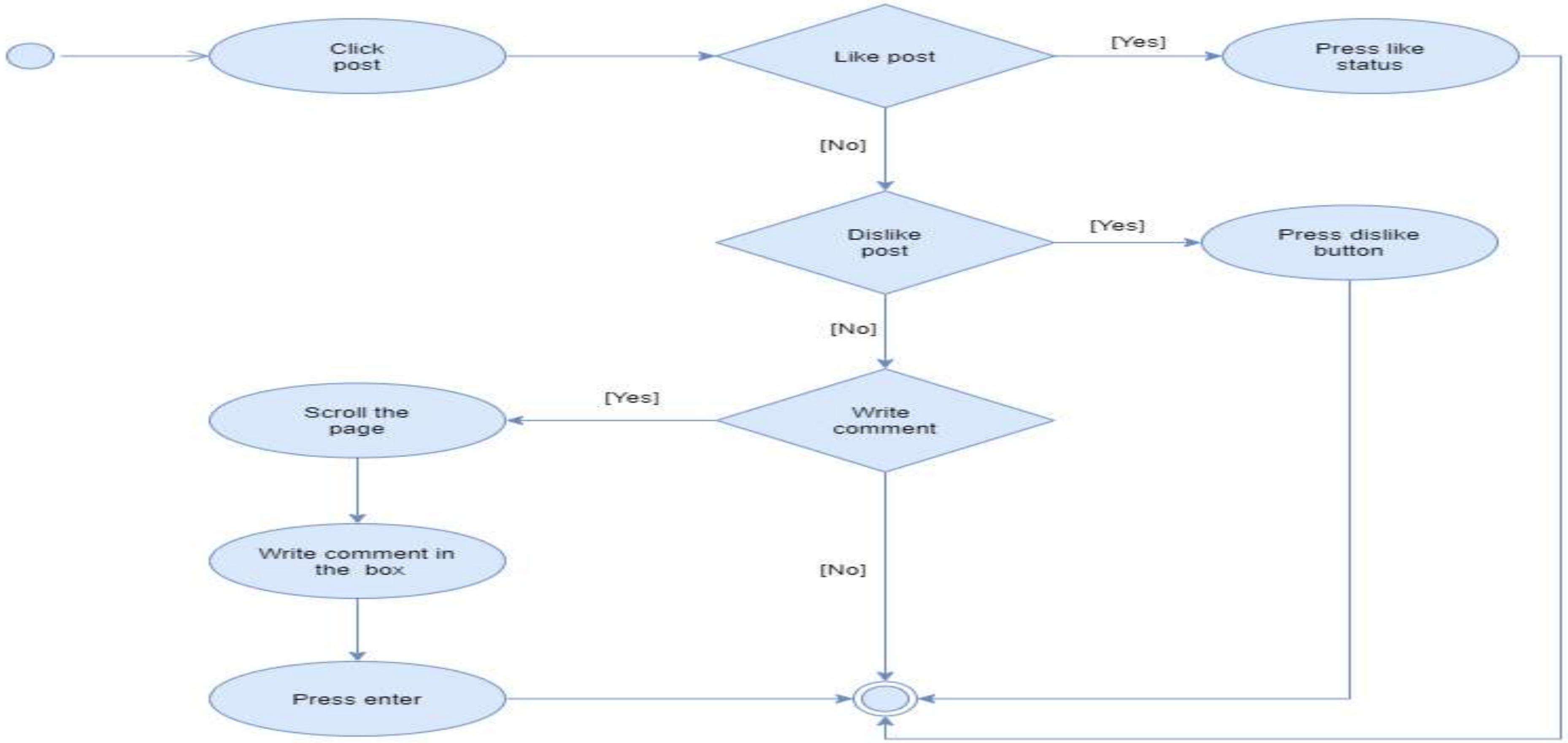
Activity Diagram for Updating Profile



Activity Diagram for Managing Account

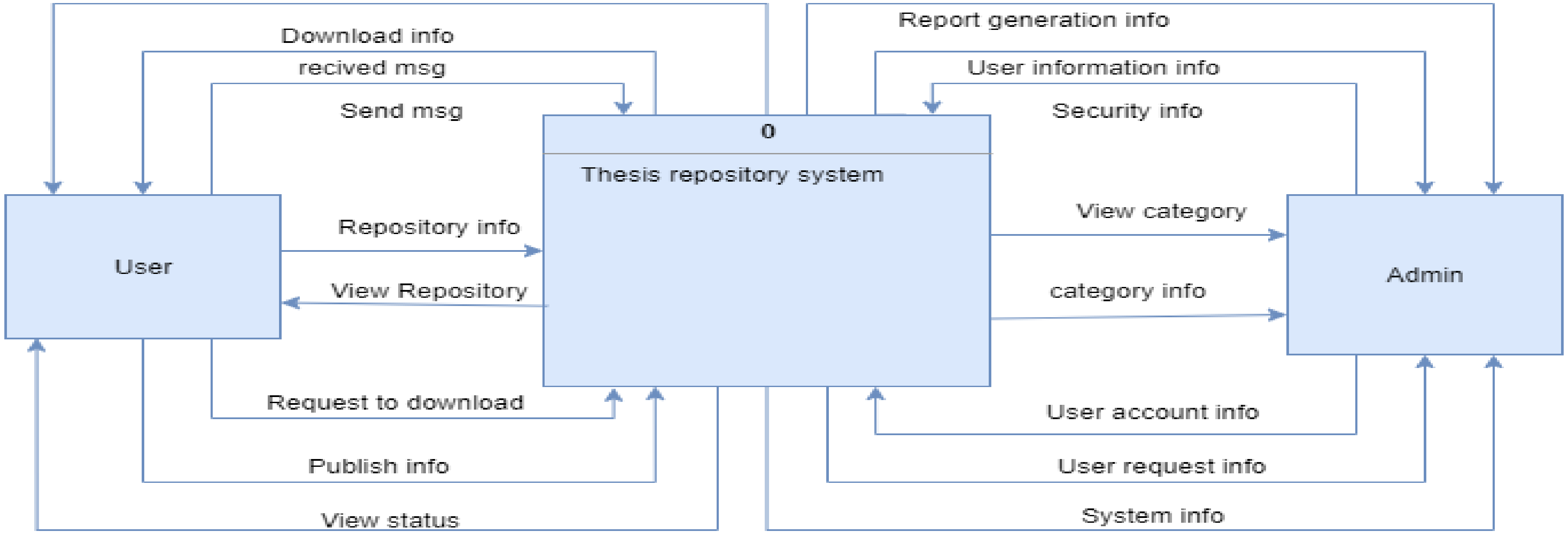


Activity Diagram for Reacting Post

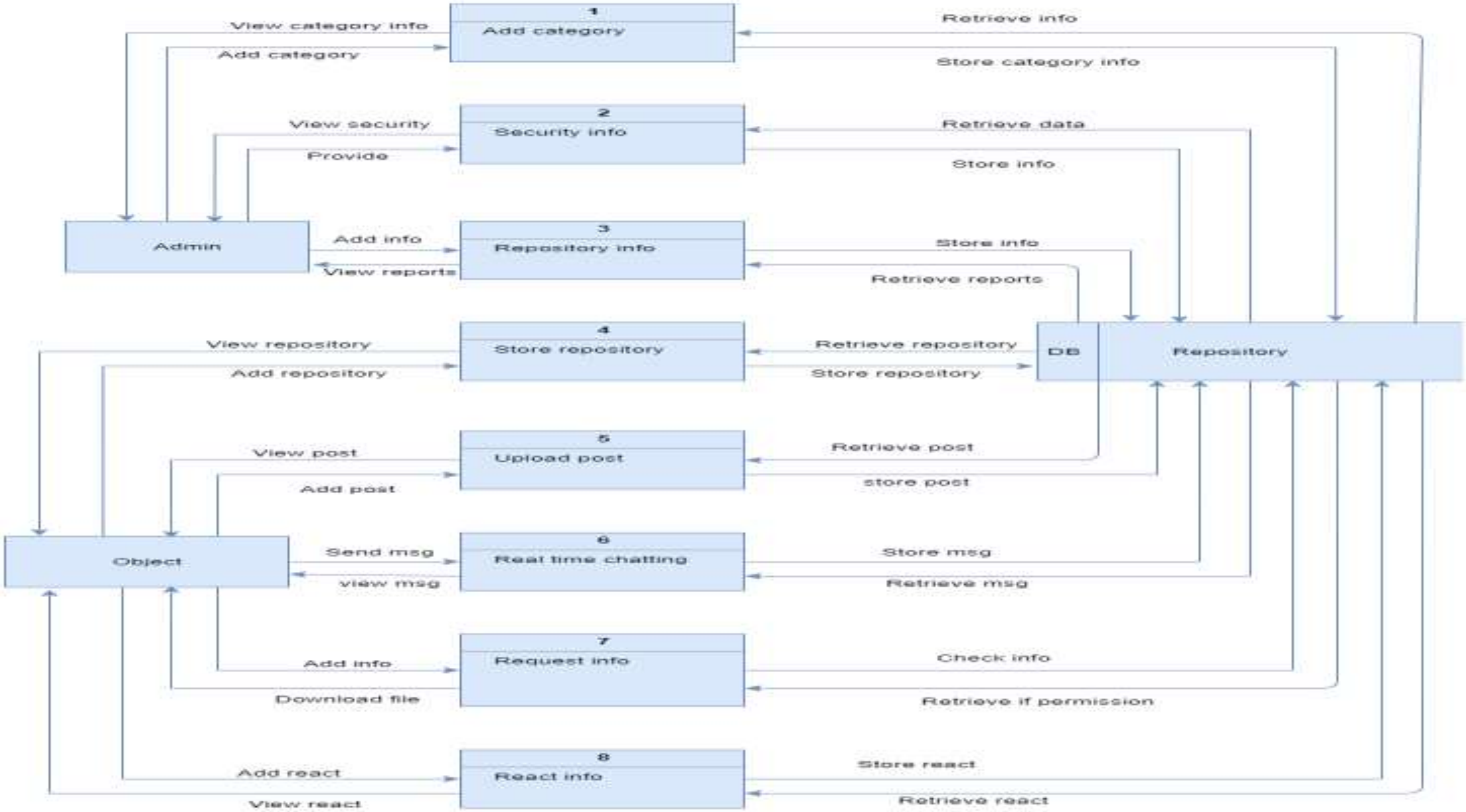


Data Flow Diagram

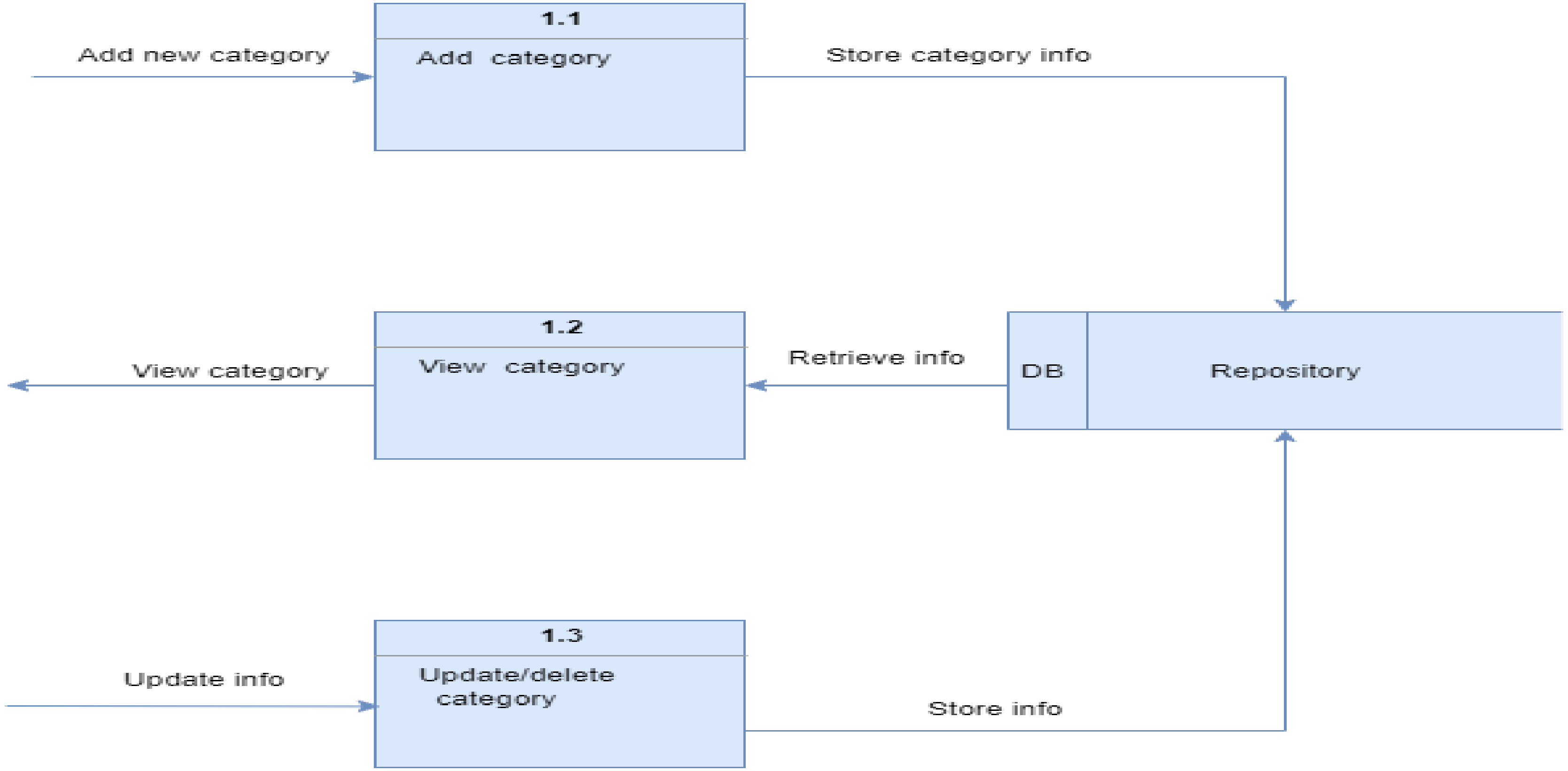
- Context Level Diagram



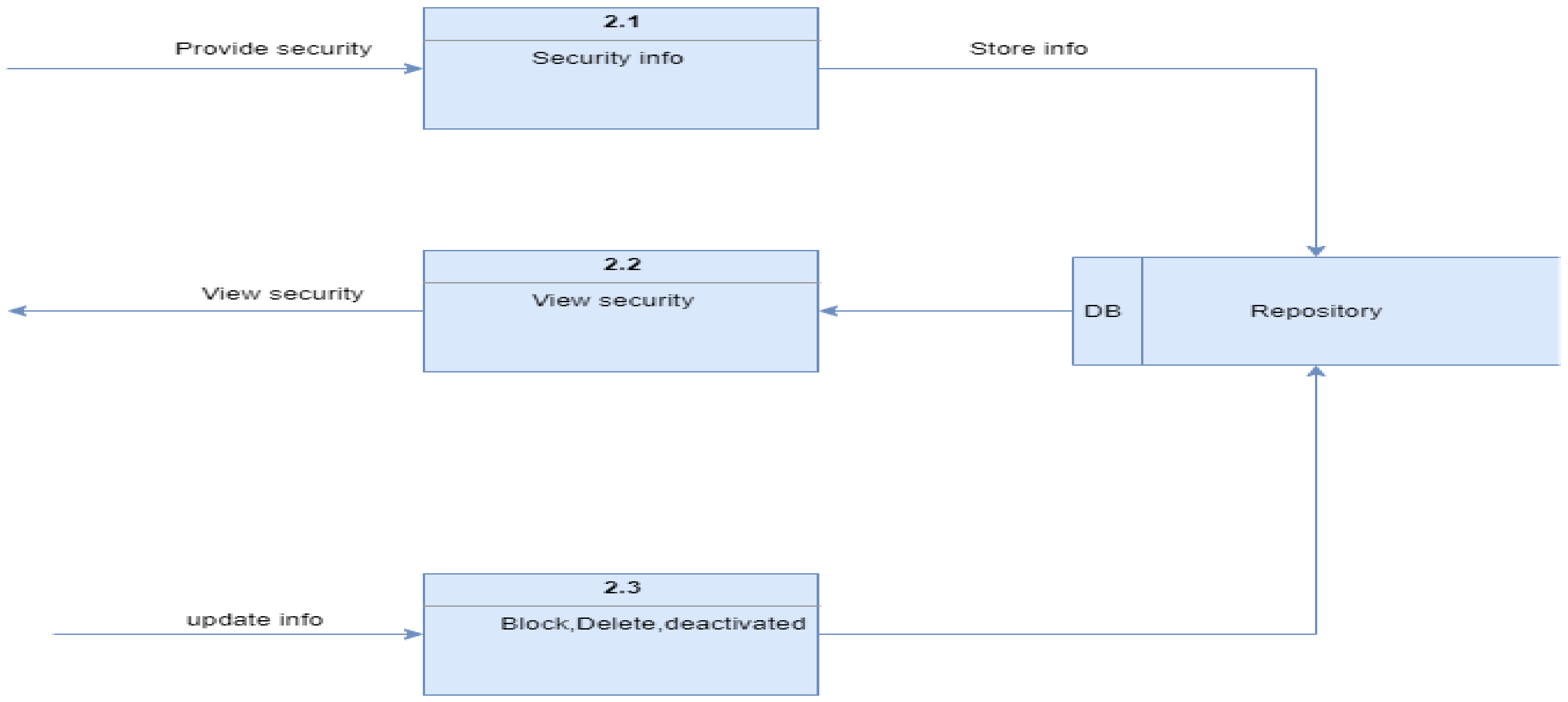
Level 1 DFD


1
Add category
View category info
Add category
Retrieve info
Store category info
2
Security info
View security
Provide
Retrieve data
Store info
Admin
3
Repository info
Add info
View reports
Store info
Retrieve reports
4
Store repository
View repository
Add repository
Retrieve repository
Store repository
DB
Repository
5
Upload post
View post
Add post
Retrieve post
store post
Object
6
Real time chatting
Send msg
view msg
Store msg
Retrieve msg
7
Request info
Add info
Download file
Check info
Retrieve if permission
8
React info
Add react
View react
Store react
Retrieve react

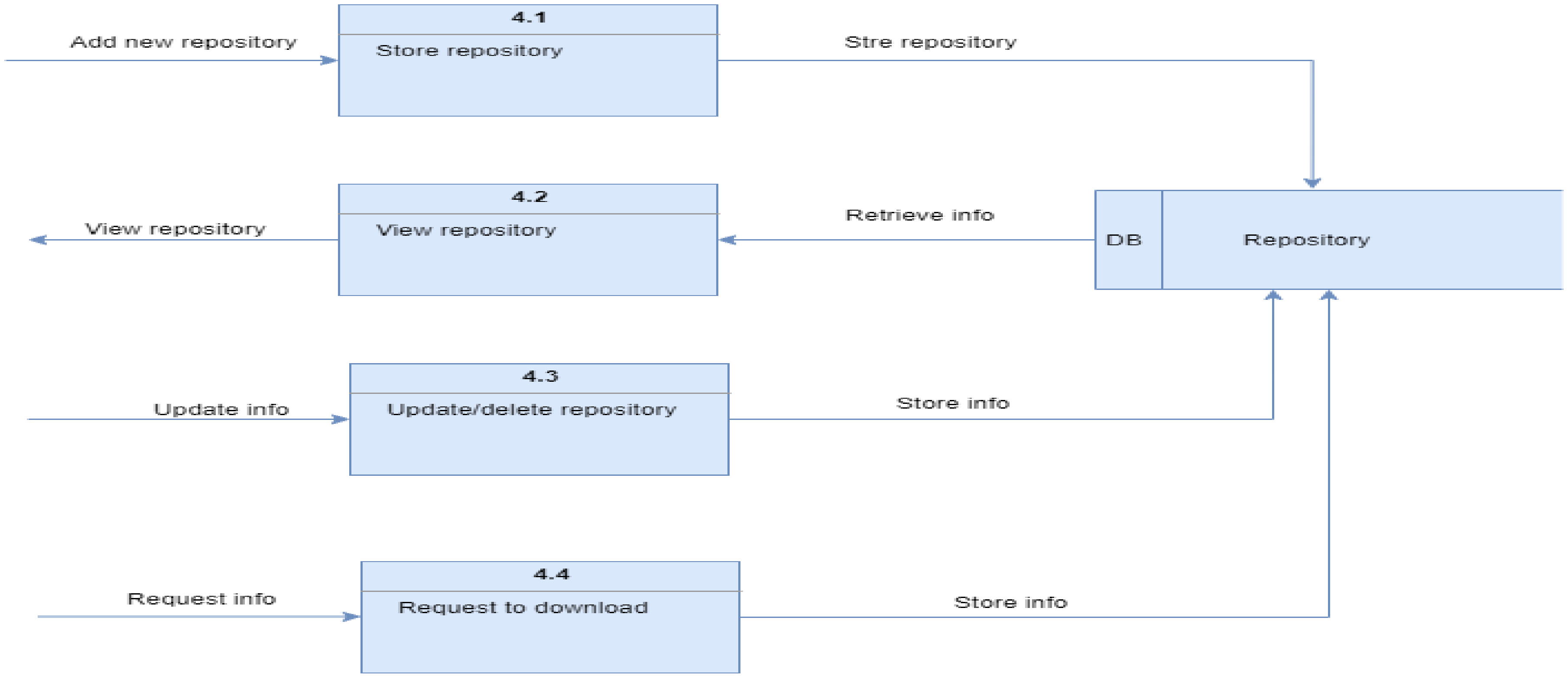
DFD of Level 1(process 1):



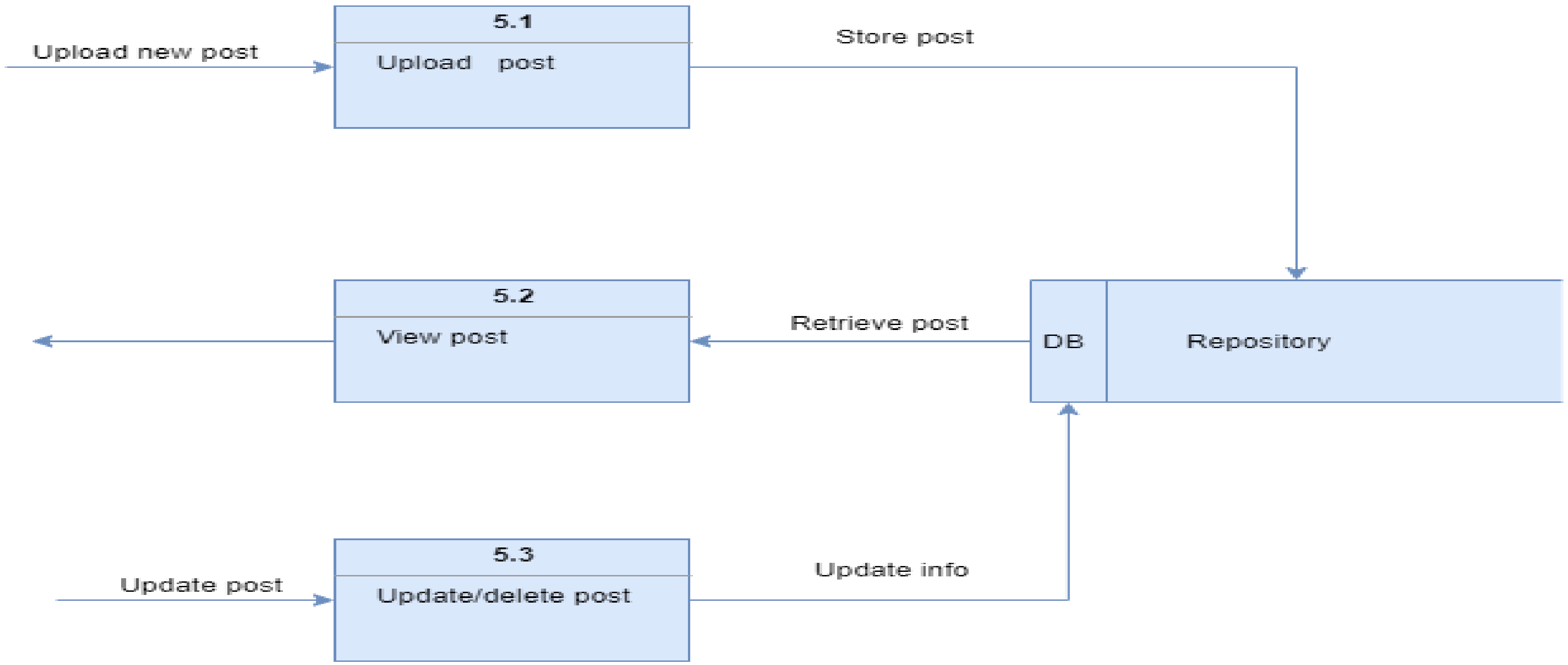
DFD of Level 1(process 2):



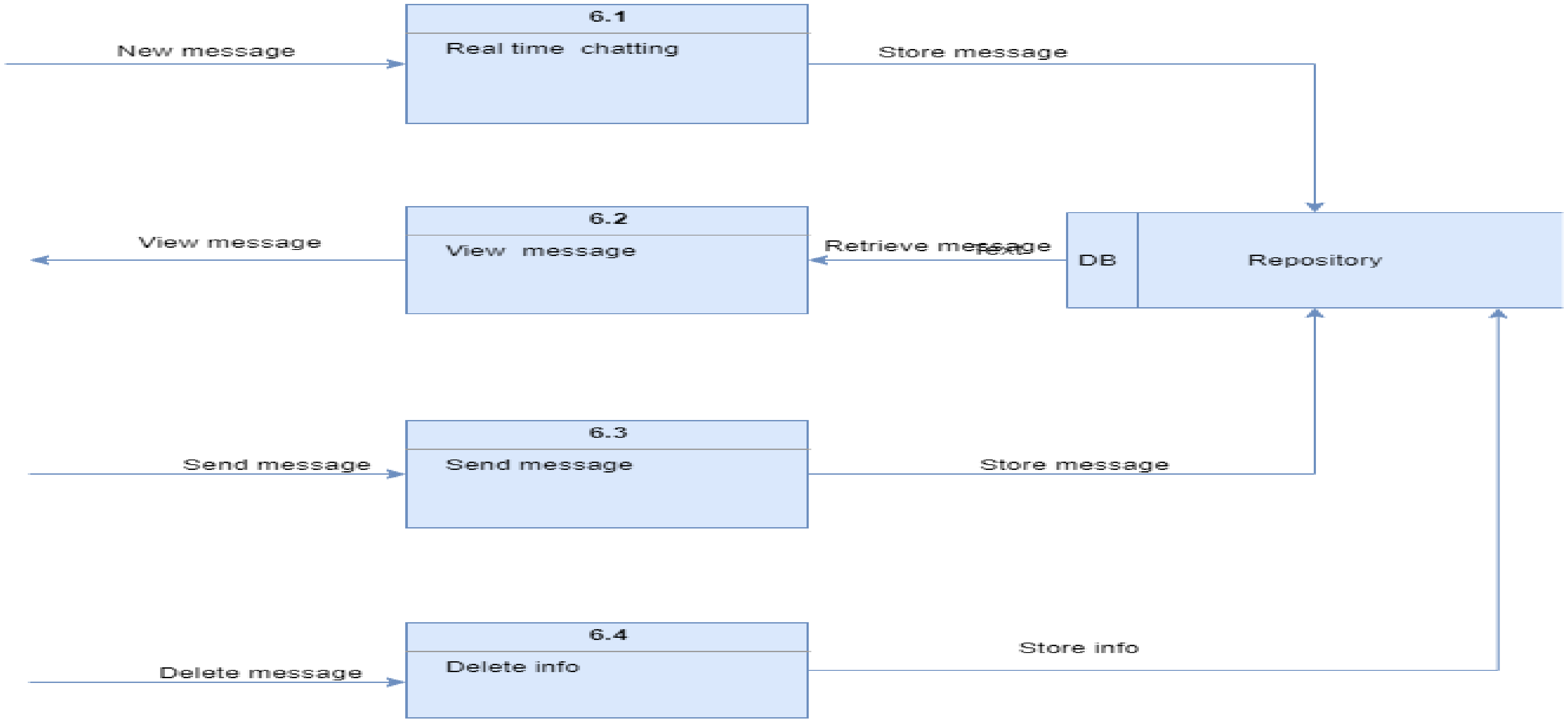
DFD of Level 1(process 4):



DFD of Level 1(process 5):



DFD of Level 1(process 6):



New message

6.1

Real time chatting

Store message

View message

6.2

View message

Retrieve message

Text

DB

Repository

Send message

6.3

Send message

Store message

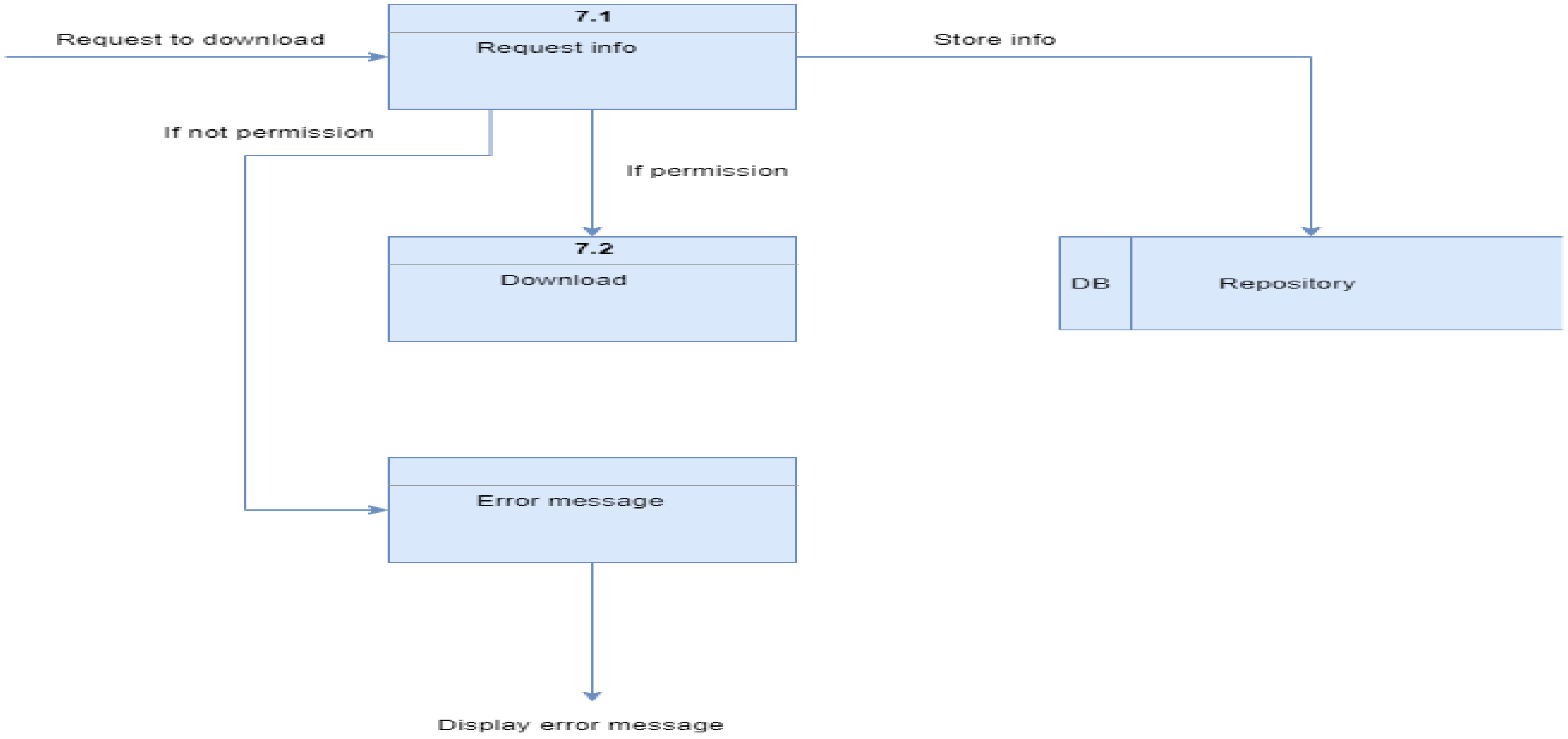
Delete message

6.4

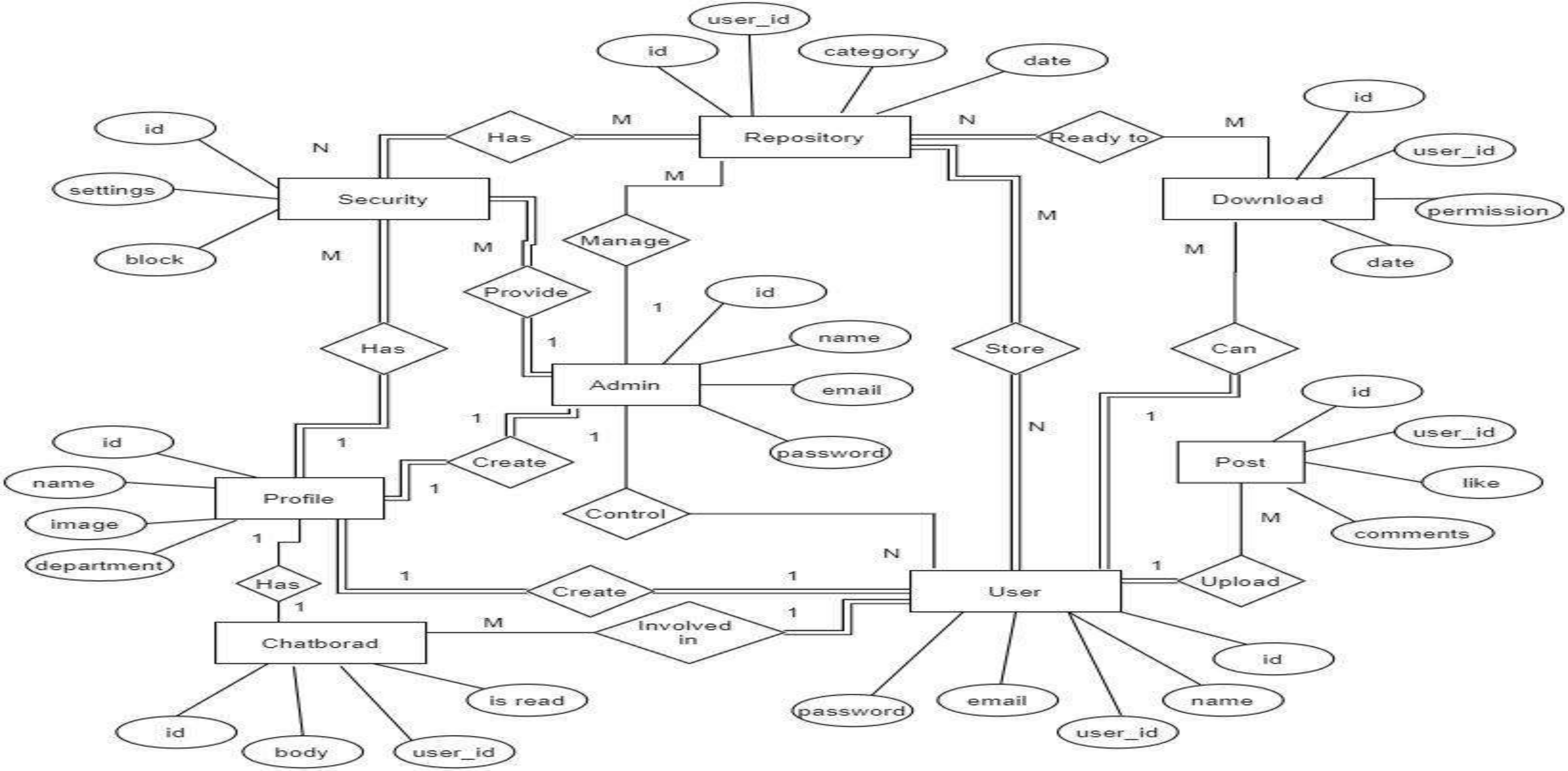
Delete info

Store info

DFD of Level 1(process 7):



Entity Relationship Diagram



Functions of Proposed System

Login into the System	F1
User Registration	F2
Add Categories	F3
Update Categories	F4
Delete Categories	F5
Manage Repositories	F6
Manage Users	F7
Handle Downloading	F8
Remove account	F9
Store Repository	F10
Manage Repository	F11
Upload Post	F12
Real Time Chatting	F13
Provide Security	F14
Friend Requesting	F15
Reacting to Post	F16
Profile set up	F17

Identifying Complexity for Transaction Function

#	Transition Function	FTRs	DETs	Complexity	UFP
1	Add Categories (EI)	1	2	Low	3
2	Update Categories (EI)	1	2	Low	3
3	Delete Categories (EI)	1	2	Low	3
4	Manage Repositories (EO)	1	8	Low	4
5	View Users (EO)	1	7	Low	4
6	Handle Downloading (EQ)	1	9	Low	3
7	Remove Account (EI)	1	5	Low	3
8	Delete User(EI)	1	8	Low	3
9	Block User (EQ)	1	8	Low	3
10	Add Security (EI)	1	6	Low	3
11	Disable System (EQ)	1	4	Low	3
12	Generate User Reports (EO)	1	13	Low	4
13	Generate Report of Repository (EO)	1	7	Low	4
14	View User Post (EO)	4	8	High	7
15	Remove User Post (EI)	2	9	Average	4
16	Generate Report of User Post (EO)	2	10	Average	5
17	Store Repository (EI)	1	9	Low	3
18	Update Repository (EI)	1	10	Low	3
19	Delete Repository (EI)	2	10	Average	4
20	Upload Post (EI)	1	3	Low	3
21	Real Time Chatting(EQ)	1	2	Low	3
22	Friend Requesting(EQ)	1	2	Low	3

Identifying Complexity for Data Function

#	Data Function	RETs	DETs	Complexity	UFP
1	categories (ILF)	1	6	Low	7
2	comments (ILF)	1	4	Low	7
3	friends (ILF)	1	4	Low	7
4	friend_post (ILF)	1	3	Low	7
5	likes (ILF)	1	5	Low	7
6	messages (ILF)	1	5	Low	7
7	posts (ILF)	1	6	Low	7
8	repository(ILF)	1	9	Low	7
9	admin(ILF)	1	5	Low	7
10	users(ILF)	1	9	Low	7
11	security(ILF)	1	5	Low	7
12	password_resets(ILF)	1	3	Low	7
Total					84

Performance and Environmental Impact

#	GSC (General System Characteristics)	DI
1	Data Communication	4
2	Distributed Data Processing	2
3	Performance	3
4	Heavily Used Configuration	1
5	Transaction Rate	1
6	Online Data Entry	3
7	End-user Efficiency	4
8	Online Update	2
9	Complex Processing	0
10	Reusability	2
11	Installation Ease	2
12	Operational Ease	3
13	Multiple Sites	1
14	Facilitate Change	3
Total Degree of Influence (TDI) (Range 0 to 70 -> influence size by +- 35%)		31

Function Point Estimation

Value adjustment factor (VAF) = (0.65+ (0.01* TDI))
= (0.65+ (0.01* 31))
= 0.96

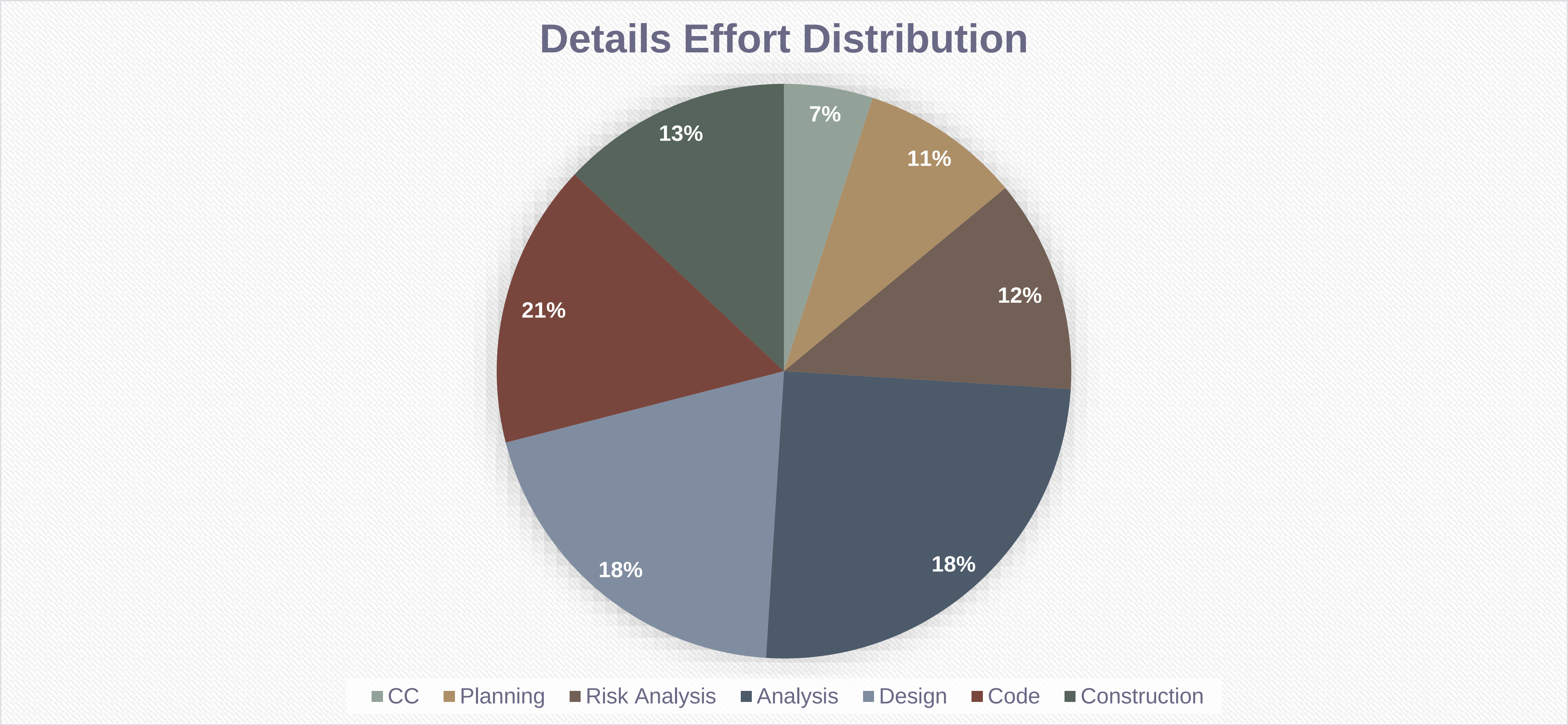
UFP = UFP (Data function) + UFP (Transaction function)
=84 + 84 = 168

AFP= UFP * VAF = 168* 1 = 168 Approx.

Total time calculation frame = 168 * 15.5 [Productivity of PHP is 15.5]
= 2604 person hours / 9 hours
= 289.33 person days /22 days
= 13.15 person months/4 persons
= 3.28 months for four person

Approximately 3.5 months required for four persons to finish the project.

Effort Distribution



Cost Estimation

❑ Personnel cost

❑ Software cost

❑ Hardware cost

❑ Other cost

Cost Estimation (Continued...)

- Personal Cost

Designation	No.of Person	Working Hours	Person Working Hours Total Salary	Person Working Hours Total Salary First Payment at 60% of Salary	Remaining 20% Payment of Salary	Remaining 20% Payment of Salary Distributed Each Month	Total Salary
System Analyst	1	190	30000	18000	6000	6000	30000
Designer	1	265	32000	19200	6400	6400	32000
Coder	1	310	35000	21000	7000	7000	35000
Total							TK. 97000

Cost Estimation (Continued...)

- Software Cost

Sl.	Software	Number	Depreciation Calculation	Depreciation Expense	Total
1	Windows 10	1	12000*33.34%	39999	((12000-3999)/48) *4=TK. 666
2	Microsoft Office	1	8000*33.34%	2666	((8000-2666)/48) *4= TK. 444
3	Xampp	1	Free	Free	-
4	Notepad++	1	Free	Free	-
5	Sublime Text	1	Free	Free	-
6	Laravel installer	1	Free	Free	-
					Total = TK. 1110

Cost Estimation (Continued….)

- Hardware Cost

Sl.	Hardware	Number	Depreciation Calculation	Depreciation Expense	Total
1	Laptop	1	40000*33.34%	13336	((40000-13336)/48) *4= 2222 Tk
2	Router	1	2000*33.34%	666	((2000-666)/48) *4=111Tk
3	Printer	1	3200*33.34%	1066	((3200-1066)/48) *4=177Tk
					Total= TK. 2510

Cost Estimation (Continued….)

- Other Cost

Particular	Cost (for 4 Month)
Office rent	TK. 34000
Electric Bills	TK. 5000
Others	TK. 5000
Total	TK. 44000

Risk Management

❑Project Risks

❑Technical Risks

❑Business Risks

Project Risk

Project Risk	
Name	Changes the requirements
Probability	Low (25%)
Impact	Marginal (2)
Description	Customer may change their requirements
Mitigation & Monitoring	Requirements are redefined by the company due to time or business needs. Meeting will be held with the company regularly. This ensures that the product we are producing solves problem.
Management	Emergency meeting between both parties to identify new project requirements and goals.
Status	Not occur

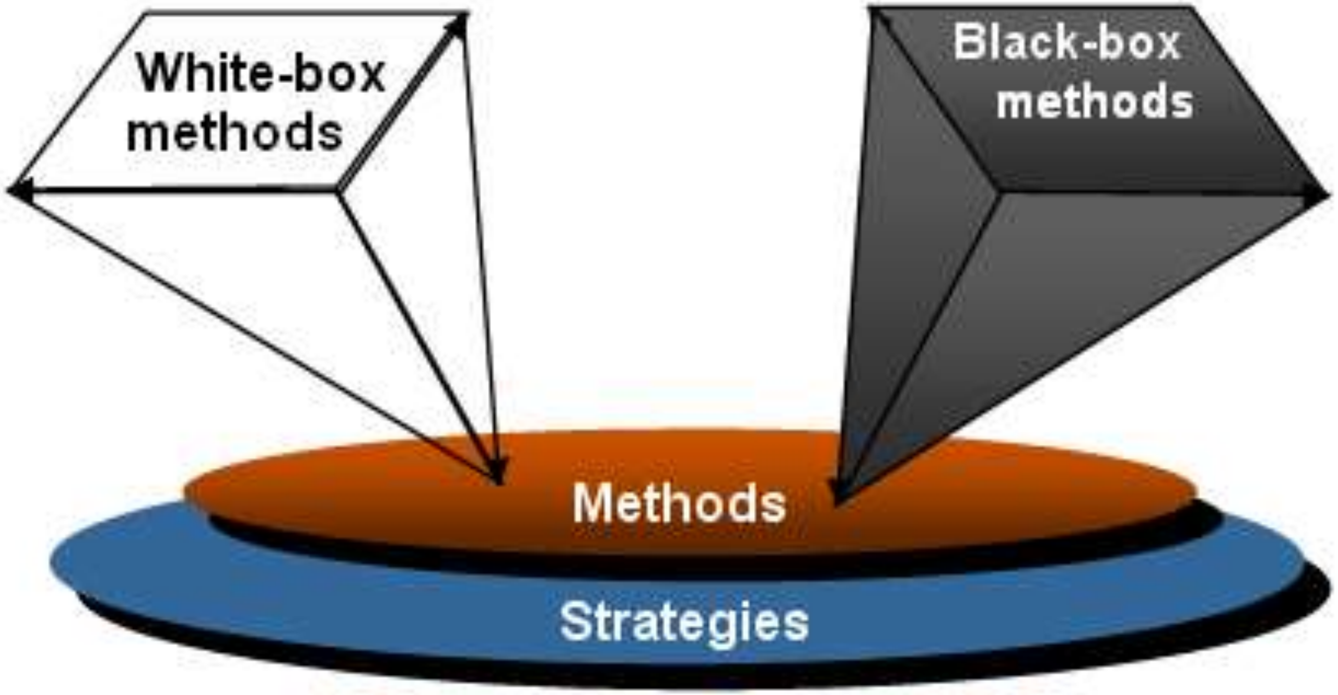
Technical Risk

Technical Risk	
Name	Computer Crash
Probability	Moderate (25-40%)
Impact	Catastrophic (1)
Description	Computer may crash due to several reasons.
Mitigation & Monitoring	We should take proper follow up of computers. We also take regular data backup every day, we can use IPS to stop unexpected shutdown.
Management	If our computer has been crashed then we will restore backup.
Status	We have not encountered such issue yet

Business Risk

Business Risk	
Name	Late delivery of the project
Probability	Very Low (05%)
Impact	Catastrophic (1)
Description	The project may take more time to complete what was estimated.
Mitigation & Monitoring	Steps have been taken to ensure a timely delivery by determining the scope of project.
Management	The only course of action available would be to request an extension to the deadline from customer.
Status	My project is completed in time.

System Testing Methodology



Testing Scenario

Testing scenario No: 1	
Scenario	User Login testing scenario of my system
Input’s	E-mail, password of User for Login
Desired Output’s	When enter E-mail, user type, password then get access level define.
Actual Output’s	For login my system works correctly
Verdict	Getting result from Desired Output’s and Actual Output’s decided this system is successful for login.

Testing Scenario

Testing scenario No: 2	
Scenario	Admin can view and monitor user details.
Input’s	Request to view user basic information
Desired Output’s	Show the basic information to the admin
Actual Output’s	For showing all users basic information my system works correctly.
Verdict	The process has worked correctly and successfully.

Testing Scenario

Testing scenario No: 3	
Scenario	Users can store Repository
Input’s	Input the repository information along with the file
Desired Output’s	Show the list of lists of repository
Actual Output’s	For showing all available repositories my system works correctly.
Verdict	The process has worked correctly and successfully.

Software Demonstration

Future Plan

❑Add a minimum and maximum storage system for storing repositories

❑Add only me and public privacy system for repository

❑Add audio video chatting

❑Add react to the message

Conclusion

Thesis Repository system is designed for the student who are doing thesis. This system helps the student as well as teachers to store their work and contact each other when need arises. In our university there has no repository system for storing the work of the thesis student. Authority keeps those thesis in manual system in the library. I have developed this system using Laravel as backend and for the frontend I have used html5, Css3, bootstrap, js, jQuery, node js, Ajax, etc. I have used Mysql database. I have used Vue Js and laravel pusher for real time chatting module.

Thank You !!!