AGENT BASED NETWORK SURVEILLANCE SYSTEM

A PROJECT REPORT

Submitted By

Gunjan Bhatt (120770107127)

Chandrashekar TR (120770107110)

In fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

In

Computer Engineering

SILVER OAK COLLEGE OF ENGINEERING AND TECHNOLOGY

OPP. BHAGWAT VIDYAPITH, NEAR GOTA CROSS ROAD,

AHMEDABAD- 382481.

SILVER OAK COLLEGE OF ENGINEERING AND TECHNOLOGY

COMPUTER ENGINEERING

2015-16

CERTIFICATE

Date: / / 2016

This is to certify that the dissertation entitled "AGENT BASED NETWORK SURVELLIENCE SYSTEM" has been carried out by "MR.BHATT GUNJAN M. (120770107127), MR.CHANDRASHEKAR T.R. (120770107110)" Under my guidance in fulfillment of the degree of Bachelor of Engineering in COMPUTER ENGINEERING (8th Semester) of Gujarat Technological University, Ahmedabad during the academic year 2015-16

External Guide:

Signature:

Name: Mr. Parth Trivedi

Designation: Project Scientist

Organization: Bhaskaracharya Institute For Space Application And Geo-Informatics

Internal Guide:

Signature:

Name: Ms. Pooja Jardosh

Designation: Assistant professor

Organization: SOCET, Ahmedabad

Head Of Department:

Signature:

Name: A/.Prof. Satvik Khara

Designation: H.O.D. [CE]

Organization: SOCET, Ahmedabad

Bhaskaracharya Institute for Space Applications and Geo-informatics

Department of Science & Technology

Government of Gujarat

Phone: 079 - 23213081 Fax: 079 - 23213091

E-mail: info@bisag.gujarat.gov.in, website: www.bisag.gujarat.gov.in

ISO 9001:2008
ISO 27001:2013
CMMI LEVEL-5

CERTIFICATE

This is to certify that the project report compiled by ***Mr. Bhatt Gunjan M. and Mr. Chandrashekar Tr.*** *students of 8th Semester* ***BE-CE from Silver Oak College OF Engineering and Technology, Ahmedabad, Gujarat Technological Univercity, Ahmedabad*** *have completed their final semester project satisfactorily. To the best of our knowledge this is an original and bonafide work done by them. They have worked on web-based application for* ***"Agent Based Network Surveillance System"****, starting from July 3rd, 2015 to April 30th, 2016.*

During their tenure at this Institute, they were found to be sincere and meticulous in their work. We appreciate their enthusiasm & dedication towards the work assigned to them.

We wish them every success.

Parth Trivedi

Project Scientist,

BISAG, Gandhinagar

T. P. Singh

Director,

BISAG, Gandhinagar

Candidate's Declaration

We hereby declare that project report titled "Agent Based Network Surveillance System" submitted towards the completion of project in 8th semester of bachelor of Computer Engineering in Silver Oak College Of Engineering & Technology, Ahmedabad is an authenticate record of our work carried out.

Candidate's signature:

First candidates name: Gunjan Bhatt

Branch: CE

Enrollment Number: 120770107127

Candidate's signature:

Second candidates name: Chandrashekar TR

Branch: CE

Enrollment Number: 120770107110

Submitted to:

Silver Oak College of Engineering & Technology, Ahmedabad

Affiliated to:

Gujarat Technological University

ACKNOWLEDGEMENT

We would like to express our sincere thanks to our Head of Department & internal guide. **Prof. Satvik Khara** and **Ms. Pooja Jardosh,** who gave us an opportunity to undertake such a great challenging and innovative work. We are grateful to them for their guidance, encouragement, understanding and insightful support in the development process.

We would also like to thank DIRECTOR OF BISAG, for providing us opportunity to undergo training during last year. Our sincere thanks to **Mr. Saurabh Bhabhor**, Project Manager of the organization and our external guide **Mr. Parth Trivedi** who gave us valuable guidance during the project.

My special thanks to my family members, for their unique support in successful completion of Project. Finally I would like to thank each and every person who has directly or indirectly helped me in this project.

With sincere regards,

Gunjan Bhatt

Chandrashekar TR

ABSTRACT

In a world which is moving forward at a great pace, Information sharing and communication with others is very vital to sustain the development of the human kind. Networking provided us a great platform for this and has since become an irreplaceable tool of communication whose need is only going to increase as we move forward. Increase in networking also means increase of burden on a network admin.

So to cope with this growing burden a network admin will need all the tools which can make his work a little simpler. Agent Based Network Surveillance System is one such tool. We are trying to provide a network admin a single application which will monitor, analyze and control network traffic exactly as he wants.

LIST OF FIGURES

LIST OF TABLES

TABLE OF CONTENTS

CHAPTER

1

INTRODUCTION

CHAPTER 1:

INTRODUCTION

1.1 PROJECT SUMMARY :-

Agent Based Network Surveillance System is a centralized application to monitor and manage networked systems using Java-based packet capturing and manipulating over LAN & WAN both. The increasing use of communication networks has raised the demand for advanced network management. A network management system handles problems related to the reliability, efficiency, security and accountability of networked systems.

This application is concerned with monitoring, analysis, security, file sharing and control of network performance to ensure effortless network operations. Accurate and effective monitoring is vital for network management, and is the main focus of our work. The program captures all the packets moving inside and outside around the network to extract useful information from it and represent it all. This information obtained from the packets will be helpful for controlling the network with more precision. Also one can analyze and manage bandwidth utilization, keep track of all activities occurring in the network and be sure that the network performance is maintained at optimal levels. The security of the resources and data available on our network is of paramount importance. File and message transferring will also be provided through our software. This application will contain a server side and a client side program. The Server program is installed on the Raspberry Pi and is operated by the network admin. It provides all the above mentioned functionalities to the network admin. The client program is installed on all the PC in the network. The clients are differentiated on the basis of Privilege level set by the admin during account creation. The client side can also be used to monitor all the activities done by that particular client himself. This application hence tries to cover a lot of ground when it comes to easing the work of a network administrator and so acts as a complete solution to for most of his/her problems.

1.2 PURPOSE :-

The main purpose of this project is to provide a wide array of functionalities at as low a cost as possible. To make this possible we chose raspberry pi a low cost mini-PC to act as our Server agent which controls the whole network.

The project also provides a network administrator all the tools needed to manage the network efficiently in one software. This makes collecting and managing the data much more easy and effective. Using different software's for gaining information of the same system seems not only tedious but also unreasonable both work and cost wise.

The network monitoring which is the main focus of application is also the most important of all the purposes. The reason why it so important is because it is required for maintaining the network's health and ensuring it is available always and for boosting its performance. Through the information obtained from network monitoring we can make better plans for future growth of the company. Meticulous monitoring of network provides us crucial information needed to warrant network upgrades and expansion in order to support new corporate goals. The big business can afford to buy costly programs out there to tackle this problem but the small or new ones usually ignore this problem which makes them vulnerable to hackers. Hence our job is to provide these business the security required at a much cheaper cost than others.

1.3 SCOPE :-

Since the use of networks especially www is becoming increasingly popular the load on network administrators will obviously increase. This make the scope of this program really wide and the need of increasing its functionality is imminent.

- It is possible to expand the program to cover intrusion detection and hold up our fort against external threat and thereby protecting important data from leaking. This can be done by using an open source software called SNORT which is written in C

programming language. SNORT is used for making our own digital signatures and stop any specific application from communicating with its server. It also detects and stops attacks.

- To overcome network congestion, network congestion control can be implemented.
- With gaining popularity of IPv6 it is feasible to migrate from IPv4 to IPv6.
- This program can also be expanded to incorporate automated vulnerability discovery.
- For increasing the security a ticketing authority can be implemented.
- Network File Discovery functionality can also be implemented
- A solution for blocking proxy services in windows so that client cannot access blocked websites can be found.
- A dedicated OS for network monitoring based on Linux kernel can be created.

1.4 TECHNOLOGY & LITERATURE REVIEW :-

1.4.1 TECHNOLOGY AND TOOLS

Technology Used: Java SE

Front End: JavaFx, Css

Back End: MySql, JNetPcap, Medusa, Iptables, JavaMail, ControlFx, Libpcap.

Development Tool: Netbeans with JDK 1.8.65

Development Language: Java

Simulation Tool: JRE (Java Runtime Environment)

Simulation OS Server: Any Debian Based Linux

Simulation OS Client: Windows/Linux/Mac

Versions support: Any version of JVM higher than 1.8.65 with javafx Libraries Present and Libpcap is supported.

1.4.2 LITERATURE REVIEW

- **Java :-**

  Java is the most popular object-oriented programming language today. It is intended to create applications that follow "Write Once, Run Anywhere Principle". The language was originally named *Oak* after an oak tree that was outside Gosling's (an Initial developer) office. It was then renamed Java from java coffee. Java was released as free and open-source software (FOOS), on November 13, 2006under GNU General Public License. Since then Java has grown exponentially with help of thousands of APIs built by many fellow developers to help other developers create a program better and more conveniently.

  The main purpose of development of java was to attain portability of software, which means an application written in java must be able to run similarly regardless of any hardware or operating system disparity. To attain this Java language code is directly compiled to an intermediate representation known as Java bytecode rather than architecture-specific machine code. Java bytecode are similar to machine code, but are designed to run on a virtual machine (VM) written specially for the host hardware. End user usually use a Java Runtime Environment (JRE) for running a java application or a web browser for java applets.

Fig 1.1 Duke, the java mascot

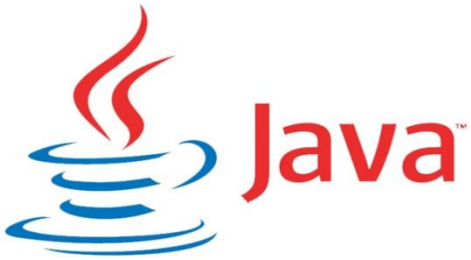


Fig 1.2 Logo of Java

- **JavaFx : -**

  JavaFX is a software development platform for developing cutting edge desktop applications and rich internet applications (RIAs) that can run on a large number of devices. JavaFX was proposed to replace Swing as the standard library for developing GUI in Java SE, but swing will also be included with java for the foreseeable future. JavaFX has support for PC and web browsers on all main operating System like Windows, Linux, and Mac OS.

  Before JavaFX version 2.0, programmers used a declarative language called JavaFX Script to create JavaFX applications. As JavaFX Script was compiled to Java bytecode, the developers also used Java code instead of JavaFX Script. JavaFX applications can run on all the computer that support Java SE, on any browser that support Java EE, or on any mobile phone that supports Java ME.

  JavaFX 2.0 and later is implemented as a native Java library, and programs using JavaFX are written in native Java. Since JavaFX 2.0, the Solaris operating system or mobile phones are not supported. Recently is Oracle planning to integrate JavaFX with Java SE Embedded version 8 and above. Also Java FX for ARM processors like cortex is in the developer preview phase.

  On desktops, JavaFX supports Windows XP, Windows Vista, Windows 7, Windows 8, Windows 8.1, Windows 10, Mac OS X and most of the Linux operating systems. Since JavaFX 1.2, Oracle has released beta versions for Open Solaris and for a few mobile Operating Systems.

- Oracle added a lot of new features in JavaFX 8, including

  - Support for 3D graphics.
  - Support for different sensors.

- Support for rich text and Printing.
- Standard dialog and alerts templates by adding ControlsFX which can be used instead of JOptionPane since JavaFX 8u40.

Fig 1.3 Logo of JavaFX

- **NetBeans :-**

NetBeans IDE is an open source integrated development environment for developing application on Windows, Mac, Linux and Solaris operating systems. NetBeans, originally known as Xelfi, began as a project by students in Czechoslovakia back in 1996. The goal was to create a Delphi-like Java IDE in Java. It became the first Java IDE written in Java, on its first pre-release in 1997. Xelfi was taken as a fun project to work on, as Java IDE space was inconceivable at that time. The project attracted enough attention for the students to decide to create a startup around it. Roman Stanek, an entrepreneur discovered Xelfi and created a plan to develop a network-enabled JavaBeans components. Jaroslav Tulach, designer of the IDE's basic architecture, suggested the name of NetBeans. Around 1999, Sun Microsystems started searching for a better Java development tools, and became interested in NetBeans. Soon NetBeans became the flagship tool set of the maker of Java itself. When Oracle acquired Sun in 2010, NetBeans became a part of Oracle and since then NetBeans IDE became the official IDE for the Java Platform.

Fig 1.4 Logo of Netbeans

- **<u>jNetPcap</u> :-**

  jNetPcap is an open-source java programming interface (API) developed by Sly Technologies. It is used as a wrapper based on both LibPcap and WinPcap native libraries. It follows native LibPcap programming model hence is very easy to understand. It uses a mixture of native and java implementation for optimum packet decoding performance. Also due to low level and efficient JNI implementation it is really lag free and works almost real-time. And as it is based on both LibPcap (UNIX system packet capturer) and WinPcap (Windows system packet capturer), It is one of the best wrapper class for Pcap applications written in Java as it will give the program both Windows and Linux support.

Fig 1.5 Logo of jNetPcap

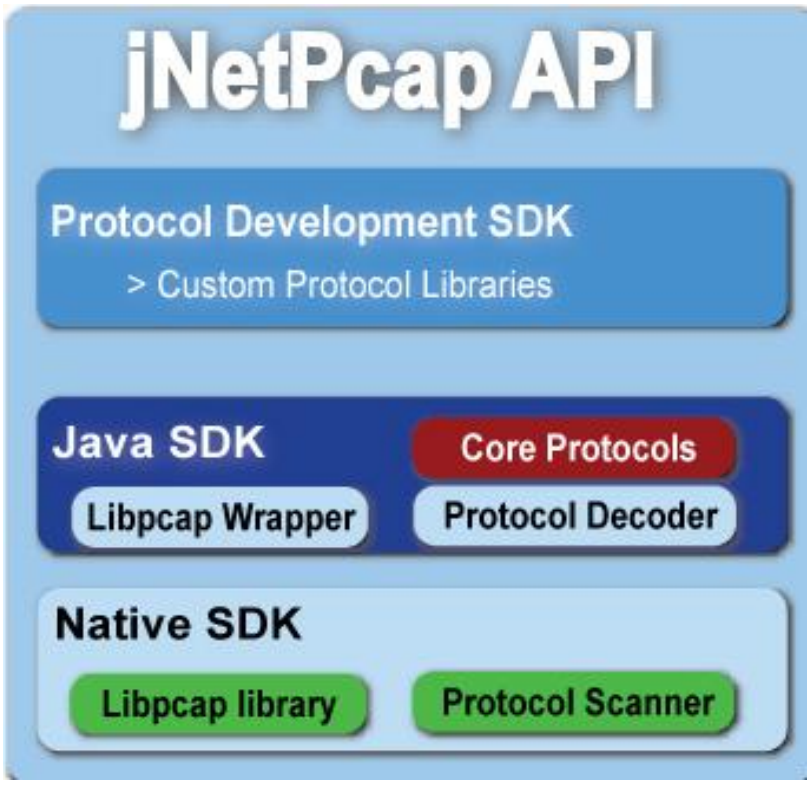


Fig 1.6 Architecture of jNetPcap

- **MySql:-**

MY SQL is an application used to create database for the Microsoft windows family of server operating system it provides an environment used to generate databases that can be accessed from work stations the web or other media such as personal digital assistant(PDA). MY SQL is probably the most accessible and the most documented enterprise datable environment right now using sql one can create and maintain data manipulating objects such as tables views sequence etc.

Fig 1.7 Logo of MySql

- **Raspberry Pi :-**

  A Raspberry Pi is a credit-card sized computer originally designed for education, inspired by the 1981 BBC Micro. Creator Eben Upton's goal was to create a low-cost device that would improve programming skills and hardware understanding at the pre-university level. But thanks to its small size and accessible price, it was quickly adopted by tinkerers, makers, and electronics enthusiasts for projects that require more than a basic microcontroller (such as Arduino devices).

  The Raspberry Pi is slower than a modern laptop or desktop but is still a complete Linux computer and can provide all the expected abilities that implies, at a low-power consumption level.

  In February 2016, the Raspberry Pi Foundation announced that they had sold eight million devices, making it the best selling UK personal computer, ahead of the Amstrad PCW.

➢ The raspberry pi used is pi B 2 model with following features:

- A 900MHz quad-core ARM Cortex-A7 CPU
- 1GB RAM
- 4 USB ports
- 40 GPIO pins
- Full HDMI port
- Ethernet port
- Combined 3.5mm audio jack and composite video
- Camera interface (CSI)
- Display interface (DSI)
- Micro SD card slot
- VideoCore IV 3D graphics core

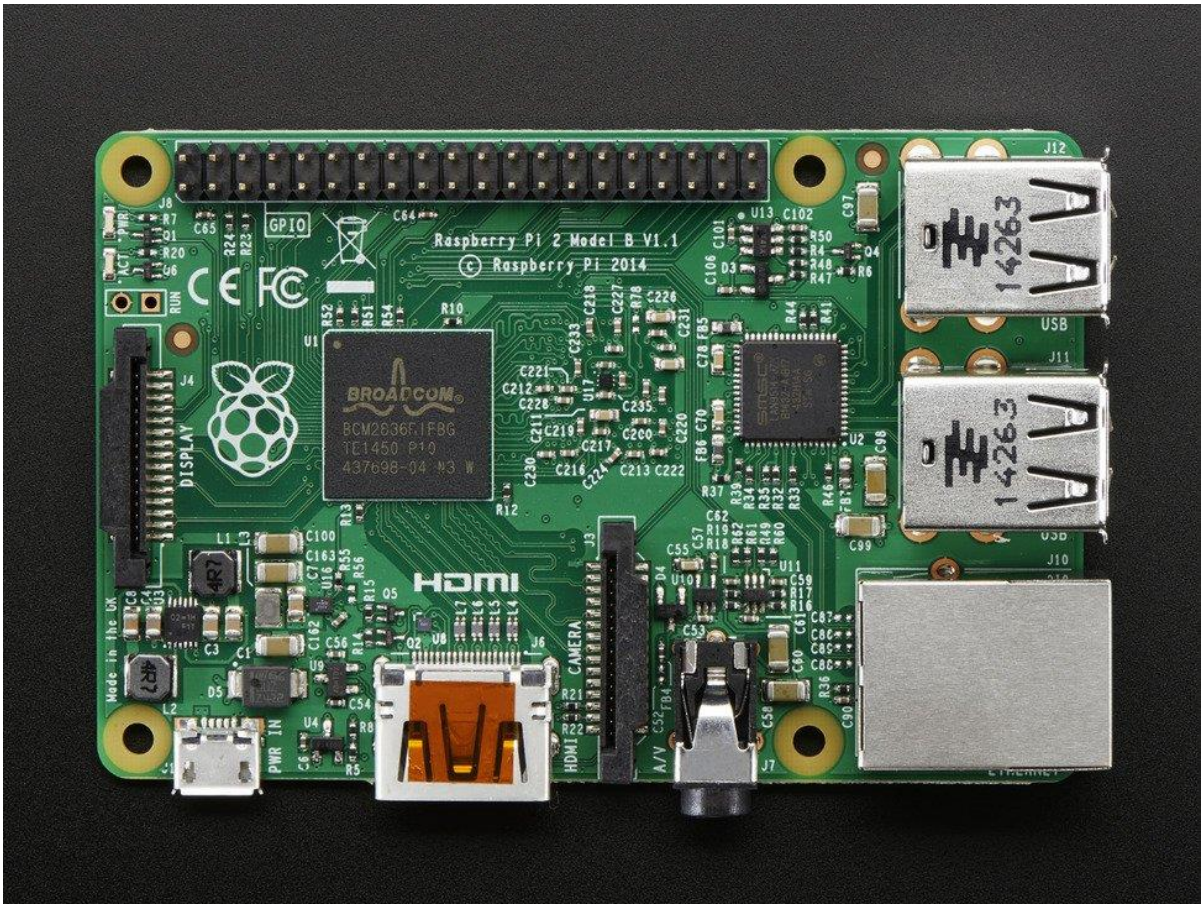


Fig 1.8 Raspberry pi 2

CHAPTER

2

SOFTWARE PROJECT MANAGEMENT AND PLAN

CHAPTER 2:

SOFTWARE PROJECT MANAGEMENT AND PLAN

2.1 PROJECT PLANNING & SCHEDULING:-

2.1.1 PROJECT DEVELOPMENT APPROACH:

For the development of this project we have used Evolutionary Prototype Model as we had little experience in this field and had to understand the requirements better even after spending adequate time on Requirement gathering and analysis part. Also our project was quite large and a lot of modules were added in our scope. We lacked the necessary skill set required to complete all those modules in the scope using a traditional model like waterfall or incremental which didn't offer us the freedom to backtrack. Evolutionary Prototype model offered us both flexibility and also modular approach for building our software and hence was the most suitable one for us.

- **Evolutionary Prototype Model** :-

Evolutionary Prototype which is also called as Breadboard Prototype is fundamentally different from a Throwaway Prototype as we just build skeleton software which will be casted aside in a throwaway prototype model while in Evolutionary prototype the prototype produced will form the heart of the system and the whole software will be built around it. In this type of Prototype model we continuously refine and rebuild our software till we reach the desired quality and function set. Due to its modular approach we can focus on building the requirements which are well understood before hand and ignore those which are not fully understood or for which we don't possess the required technical sets yet. We can implement these ignored requirements or modules in future after we fully grasp all the required concepts. This allows the developers the freedom to add features which were not even identified during requirement gathering and analysis phase without making much difference to the already finished work. This makes evolutionary prototyping a very dynamic and up to date model and we don't stand the risk of creating an obsolete system

which can be true in the case of traditional models in a fast changing IT landscape. Also as evolutionary prototypes are functional prototypes they can be used as a temporary solution till the whole system is created which will not be the case in throwaway prototype which can't be used in place of the whole system. Also in these types of models the errors and bug are found pretty easily in the early stages of development which will prevent the downward flow of them. The main disadvantage of this model is that it can escalate the complexity of the software as the scope may grow way more than what we had planned at the start of the project. This can badly affect the cost-effectiveness of the system and also too much extra insignificant functionality may make the software laggy and inefficient.

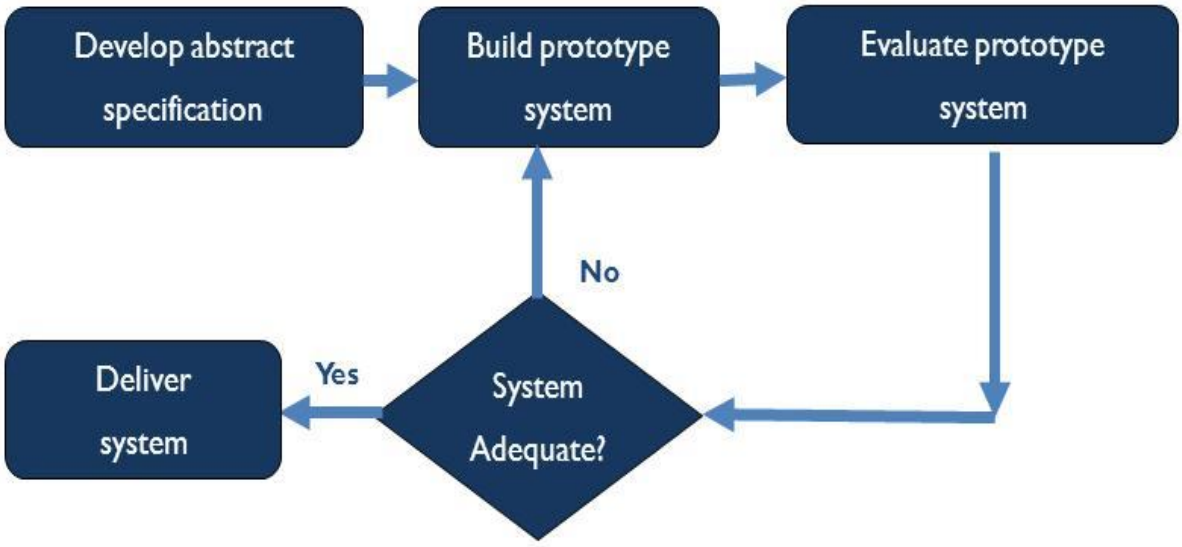


Fig 2.1 Evolutionary Prototype Model

In the prototype model, we start by gathering very basic requirements of our software and thereby develop abstract specifications of the software. After that we start the development phase by building a prototype system using the specifications gathered. We then use the prototype system to understand the system better and also find out the missing requirements if any. If we find the system is adequate we deliver the system to the end user but if we don't we again build a new prototype with upon the old prototype, i.e., we add more modules in the next build and then again check if the system is adequate. We keep repeating this cycle till we find the system adequate and we can deliver the system.

2.1.2 PROJECT PLAN

- This is the part of project management which deals with creating schedules using Gantt charts and progress report within the project environment.
- At the Beginning, the project scope is spelled out and the convenient techniques for completing the project are determined. After this step, the time period for the different tasks important to complete the project are listed and are broken down into a smaller structures of work.
- Project planning is used for organizing various areas of a project, Along with project plans, workloads and the management of groups and individuals.

➔ **Milestones & Deliverables**:-

Milestones are the points which we have to go through for completing our project successfully on specified timing with having any major hiccups. The outputs which comes after completion of any milestone are called deliverables.

Table 2-1 Milestones & Deliverables

Milestones	Deliverables
Study many different network monitor systems & its Problems. Started collecting SPMP the Requirements.	Software Project Management Plan (SPMP)
Study of existing System and its flow; Understand the situation and its solution.	System Requirement Study
Gathering the requirements of the project using different fact-finding techniques.	
Understanding the project in depth and doing the requirement analysis by studying process of different Network Monitors modules and database creation.	System Analysis
On the basis of detailed study and specification of the requirements Data dictionary was initiated.	
Creating UML design based on analysis.	

Started Application Flow Diagram.	Detailed Design Document
The functional Design of the application.	
Started making a quick design or 'prototype' of modules to understand and refine requirements accordingly.	First Prototype
Completed the prototype design.	
Studying the prototype for better understanding of system and checking if any more module is required.	Requirements for next prototype if any
Checking if the system is adequate or not.	
System is found complete.	Final Prototype
Completed Application Testing and deployed on client device.	Deployed Application

➔ **Roles and Responsibilities**:-

Table 2-2 Roles and Responsibilities

Role	Responsibility	Team/Member
Project Guide	Defining scope	Mr. Parth Trivedi and Ms. Pooja Jardosh
	Providing required resources	
	Tracking and monitoring progress of project.	
	Analysis and Effort Estimation.	
Team Leader	Designing & Documentation	Gunjan Bhatt T.R. Chandrashekar
	Execution and implementation of project as per defined schedule.	
Team Member	Software development as per the design and Documentation and defined scope	T.R. Chandrashekar Gunjan Bhatt

QA	Testing and Quality Check as per Defined Requirement.	T.R. Chandrashekar Gunjan Bhatt

2.1.3 SCHEDULING:-

Project scheduling is all about the various techniques that can be engaged to handle the activities that need to be done during the development of a project.

Scheduling is undertaken before the initialization of project and involves:

- Identifying the various tasks that need to be undertaken.
- Calculating how long each task will take.
- Providing all necessary resources including man force needed.
- Scheduling which task to be carried out when.

Once the project is on the way, control needs to be applied to make sure that the plan continues to show the best forecast of what is going to happen in the future:

- Depends on what occurs during the development;
- Usually causes recreation of the plan.

Table 2-3 Scheduling

Task Name	Start Date	Finish Date
Feasibility Study	26/06/2015	10/7/2015
Requirement Analysis	13/72015	8/9/2015
System Design	10/9/2015	9/11/2015
Prototyping	16/11/2015	1/12/2015
Implementation	1/1/2016	15/4/2016
Testing	18/4/2016	22/4/2016
Deployment	25/4/2016	29/4/2016

- **Gantt Charts** :-

Gantt chart contain a bar or line graph containing information about the scheduling of each tasks of a project. It will have the following features:

- activities of the project are described on the left hand side;
- time scale of the project will be either at the top or at the bottom;
- estimated duration of each task are represented using a horizontal bar or line;
- relationships between different activities are displayed;
- the horizontal bars are usually shaded to discuss the actual time used for an activity (an alternative is to show actual and estimated by two different lines);
- At the review point, a vertical cursor is kept to makes it easy to determine activities which are running on schedule and which are not.

Gantt chart therefore gives a lot of important information graphically, so that it is easy to understand and demnostrate how well the project schedule is working. It basically gives all the information we found out and estimated during the project scheduling.

ID	Task Name	Start	Finish	Duration
1	Feasibility Study	6/26/2015	7/10/2015	11d
2	Requirement Analysis	7/13/2015	9/8/2015	42d
3	System Design	9/10/2015	11/9/2015	43d
4	Prototyping	11/16/2015	12/1/2015	12d
5	Implementation	1/1/2016	4/15/2016	76d
6	Testing	4/18/2016	4/22/2016	5d
7	deployment	4/25/2016	4/29/2016	5d

Fig 2.2 Gantt Chart

2.2 RISK MANAGEMENT:-

A risk is a potential problem – it might occur or it might not, this is uncertain. We can't be sure that which event will occur at which time or not, but if it does has an unfavorable effect on a project. The project should be handled in such a way that the risks don't affect the project in such a big way. So risk management commence where a simple project management completes.

Risks can be categorized as follows:

Table 2-4 Risk Management

Components / Category	Performance	Support	Cost	Schedule
Catastrophic	Failure to meet the requirement would result in mission failure.	Non responsive or unsupportable software.	Financial shortage and lack of resources.	Failure results in schedule delays. Unachievable line of code.
Critical	Database server crashes.	A backup system with the server installed to be ready for operation.	Cheaper system that does not meet sufficient requirement.	System maintenance after every fixed time period.
Marginal	Overload on server system due to large increase in no. of parallel connections.	Either the bit-rate of the transmission of the video has to be reduced or to decrease the no. of parallel connections.	Sufficient financial support	Reduce the no. of multicasting and also provide some constraints

2.2.1 RISK IDENTIFICATION:-

Risk identification is the process of understanding risks that could potentially avoid the program, business, or investment from gaining its objectives. It contains documenting and delivering the concern. To identify the necessary risks which might affect a software project, it is important to categorize risks into different classes.

Table 2-5 Risk Identification

Risk Type	Description
Technology	The underlying technology on which the system is built is superseded by new technology.
Performance	The server may not handle large no. of parallel requests made by users.
Viruses	Viruses may corrupt software files.
System Failure	In case of system crash, development software and files may get lost.
Hacking	The hackers may spam or flood server with large no. of requests or data.
	Hackers can perform brute force attack on login page to gain access to the system
	Hackers can also try to perform denial-of-service attack to take down the network

2.2.2 RISK ANALYSIS:-

Risk analysis and management are various steps that help a software team to determine and manage difficulties. Many problems bugs a software project. Risk is conceivable problem;

it might occur, it might not. But whatever be the outcome, it's really important to identify it, measure its probability of it happening, determine how it would impact, and create a backup plan should the problem actually occur.

Table 2-6 Risk Analysis

Risk		Description	Probability	Impact
Technological Changes		The underlying technology on which the system is built is superseded by new technology.	Low	Medium
Database server crash		Big amount of data can cause database server crash.	Medium	High
Viruses		Viruses may corrupt the software file.	Low	Medium
System Failure		In case of system crash, development software and files may get lost.	Low	High
Hacking	Flooding	The hackers may flood the system with unnecessary requests	Low	Low
	Spamming	The hackers may spam the system	High	High
	Brute Forcing	Hackers can brute force the login system.	Low	Medium

2.2.3 RISK PLANNING:-

Risk planning the identification and scheduling of actions needed to reduce the level of risk within a project.

Table 2-7 Risk Planning

Risk		Remedies/Plan
Database crash		Create a backup database
Viruses		Install good antivirus software on the system.
System Failure		Backup should be kept on another system or on cloud storage services like Google drive, Dropbox etc.
Hacking	Flooding	Configure the firewall setting
	Spamming	
	Brute Forcing	Limit the no. of password tries

2.3 ESTIMATION:-

2.3.1 EFFORT AND COST ESTIMATION:

The process of forecasting the most realistic amount of effort necessary to create and maintain software using an incomplete set of inputs is Effort Estimation. The most common methods used for effort estimations are:

2.3.2 FUNCTION POINT:

Function Point Analysis (FPA) is the measure of clear business relevance. FPA was introduced by Allan Albrecht of IBM during 1979, the FPA technique appraises the functions present within software in terms that are relevant to the software users. The measure is directly related to the business requirements that the software is created to

address. Therefore it can be easily applied across various development environments and all through the life of a project, from the beginning of requirements analysis till the deployment of the software. Also business measures, like the productiveness of the development process or the cost per unit to maintain the software, can be derived easily. The function point measure is usually derived after a no. of stages. Using an established set of criteria, every business functions are numeric indices in accordance to their types and complexities. These indices are calculated to give a basic measure of size which is then filtered by adding a number of other factors related to the software. The final result is a single digit called the Function Point index. It measures the software product's size and complexity.

So the equation becomes, FP = UFP * TCF.

➔ **Function Point equations :-**

- **Unadjusted Function Point (UFP) =** (Number of inputs) × 4 + (Number of outputs) × 5 + (Number of inquiries) × 4 + (Number of files) × 10 + (Number of interfaces) × 10.
- **Technical Complexity Factor (TCF) =** 0.65 + 0.01 × Degree of Influence
- **Function Point (FP)** = Unadjusted Function Point × Technical Complexity Factor
  - No. of inputs: 140
  - No. of outputs: 65
  - No. of Inquiries: 58
  - No. of files: 514
  - No. of interfaces: 311
- **UFP** = 140 × 4 + 65 × 5 + 58 × 4 + 514 × 10 + 311 × 10 = 9,367
- **TCF** = 0.65 + 0.01 × DI = 0.65 + 0.01 × 25 = 0.9
- **FP** = UFP × TCF = 9,367 × 0.9 = 8431

2.3.3 COCOMO MODEL:

The most commonly used software cost algorithmic is Constructive Cost Model (COCOMO). The COCOMO model has a very simple form:

MAN-MONTHS = K1 × (Thousands of Delivered Source Instructions) ^{K2}

Where K1 and K2 are two variables lying on the application and development environment.

Calculation of the basic COCOMO model can be done more accurately by applying other factors concerning the necessary functions of the software to be developed, the eligibility and experience of the project group. Some of these factors are:

1. Intricacy of the software.
2. Required accuracy.
3. Size of data base.
4. Required competence.
5. Analyst and developer ability.
6. Knowledge of team in the project's domain.
7. Tools and software development practices used.

COCOMO model can be applied to three different classes of projects:

- Organic projects – This is used when a small team of people with good domain experience work with a requirements that are not rigid.
- Semi-detached projects – This is used when a team of medium size with some mixed experience work with formally decided rigid requirements.
- Embedded projects – This is used when the set of constraints are rigid and no changes are allowed. It is also combination of organic and semi-detached projects.

Mode Used for our project: Semi-detached

We used semi-detached type project because we have very less experience in our selected domain of network programming and related systems.

Hence, the values are as follows:

$a_b = 3.0$, $b_b = 1.12$, $c_b = 2.5$, $d_b = 0.35$.

➔ **COCOMO equations:**

- Effort Estimation (E): $E=a_b \times (KLOC)\ ^\wedge\ b_b$
- Duration Estimation (D): $D =c_b \times (E)\ ^\wedge\ d_b$
- Person Estimation (P): P =E / D
- Cost Estimation (C): C = D × Labour Charge

Table 2-8 Lines of Codes of Server

Modules	Line Of Code (LOC)
Mail Notification	53
Packet Monitoring	625
Access Control	120
Bandwidth Monitor	80
Database Controller	1160
Communication	40
Unlock Code generator	40
Server Controller	763
Fxml Document Controller	1780
Fxml GUI	832
Login Controller	50
Login GUI	21
User Detail GUI	33
User Detail Controller	70
GUI Css	43
Total Line Of Code	5713

Table 2-8 Lines of Codes of Client

Modules	Line Of Code (LOC)
Login/ Register	306
Login GUI	28
Login Controller	239
Register Controller	134
Register GUI	31
Dashboard GUI	123
Dashboard Controller	400
Client Controller	82
File Transfer and Send Message	189
Bandwidth	63
Total Line Of Code	1595

∴ Total KLOC: 7.308

➔ **Effort Applied (E):**

$E = a_b \times (KLOC)\ ^\wedge\ b_b$

$E = 3.0 \times 7.308^\wedge\ 1.12$

E = 27.8 ~ 28 person-months

➔ **Development Time (D):**

$D = c_b \times (E)\ ^\wedge\ d_b$

D = 2.5 × (27.8) ^ 0.35

D = 8.08 ~ 8 months

➔ **People Required (P):**

P=E/D

P = 27.8/ 8.08

P = 3.44 ~ 3 people

➔ **Project Cost (C):**

C = Development time (D) × Labour charge

C =8.08 × 9,000

∴ Cost = 72,720 Rs.

2.3.4 COST ANALYSIS:

- Estimates are used to calculate the cost of building software. In this we predict how much amount of resource we will be using and how much developers will be taking part in creating the system and then use it to calculate the price which we have to charge the customer with for making our project profitable.

- A cost-benefit analysis is required to determine economic feasibility. The main goal of the cost-benefit analysis is to find out whether it is economically sound to invest in the project. If the return on the investments is profitable, then the project is considered economically sound. Else the idea can be scraped without spending much on some worthless venture.

CHAPTER

3

SYSTEM REQUIREMENT STUDY

CHAPTER 3

SYSTEM REQUIREMENT STUDY

3.1. USER CHARACTERISTICS:-

This section describes the type of user which deals with the applications. Basically, this application has types of users as given below:

3.1.1 NETWORK ADMINISTRATOR :-

- Setup up gateways and design network.
- Monitor all the traffic on the network and do deep packet inspection when required.
- Monitor and manage bandwidth usage.
- Remote desktop access and monitoring.
- Identify and Troubleshooting network defects.
- Create list of accessible websites (Access control list).
- Update and delete entries in database.
- Crete new user id and password for new users.
- Grant and revoke privileges as and when required.
- Transfer important files to different users.
- Create the list of required network updates in future.

3.1.2 END USER :-

- Login/logout to abnss
- Retrieve forgot password
- Check own session usage
- Transfer file and messages to other users.
- Remote desktop sharing
- Request for specific resources

3.2. HARDWARE AND SOFTWARE CHARACTERISTICS :-

Table 3-1 Hardware Characteristics

Requirement Specification	For Simulation Platform	For Admin System	For End User
Processor	Intel Core i3 or higher	1.0 GHz	1.0 GHz
Disk space	1 GB	3 GB	2 MB
RAM	4 GB	1 GB	512 MB

Table 3-2 Software Characteristics

Requirement Specification	For Simulation Platform	For Admin System	For End user
Operating System	Any OS with JVM 1.8.65 support	Debian Based OS with JVM 1.8.65 support	Any OS With JVM 1.8.65 support
Development Kits	JDK 1.8.65	-	-
IDE/ Workbench	Netbeans/ MySql Workbench/ Scene Builder	-	-
Database Server	-	Mysql	-

3.3. CONSTRAINTS :-

3.3.1 REGULARITY POLICIES :-

- Commonly used icon/shortcuts should be used for representation for ease of usage.
- User interface should be snappy and lag free and should not use a lot of resource.
- User interface should also be easy to understand with all features easily accessible.

3.3.2 HARDWARE LIMITATIONS :-

- High speed internet is required for this program to work at optimal level.
- Admin System must be high powered state-of-art machine with a really good NIC.

3.3.3 INTERFACE TO OTHER APPLICATION :-

- The program requires an interface to libpcap for fetching information about all the packets passing through NIC.

3.3.4 PERFORMANCE REQUIREMENTS :-

- The application must be responsive and bug-free, so that it doesn't crash.
- The output must be displayed at almost realtime.

3.4. ASSUMPTIONS AND DEPENDANCIES :-

- It is assumed that the user is familiar with all the Environment in which he is running the program.
- User is assumed to be familiar with all the rules and regulations of the organization.
- Network admin is familiar with all the functionalities provided in the application and can derive required data from it.

CHAPTER

4

SYSTEM ANALYSIS

Chapter 4

SYSTEM ANALYSIS

4.1 STUDY OF CURRENT SYSTEM:

- There are many network monitor systems today which gives a host of functionalities but not any one offer what can be considered as a complete solution.
- Most functionalities found in these softwares are complex and hence special care has to be taken in UI to make the system understandable.

4.2 PROBLEMS IN EXISTING SYSTEMS:

- The current systems found in the market are either too costly or don't have even the basic required functionalities.
- The open source softwares like wireshark have a complex and non-user friendly User interface or are very basic and don't give more functionalities.
- The paid software like PRTG, even though provide a lot of functionalities, are too costly to be bought by small startups.
- The need is to find the right balance between both functionalities and cost.

4.3 REQUIREMENT OF NEW SYSTEM:

4.3.1 FUNCTIONAL REQUIREMENTS:

- The user interface must be simple and easy to understand.
- The program should be snappy and lag free and provide real-time outputs.

4.3.2 NON-FUNCTIONAL REQUIREMENTS:

Security:-

- The program must start only after proper authentication of the user.
- The message and file transferred must be encrypted properly.

Reliability:-

- The database must be updated and the system should not crash

Scalability:-

- New functionalities can be added to the software without having to change the whole system.

4.4 FEASIBILITY STUDY:

➔ **Technical Feasibility**:

Technical feasibility of a project determines whether a project can be developed using the technology on hand. The system is technically feasible as the front-end and the back-end required for it is available and already installed.

➔ **Operational Feasibility**:

In the system operational feasibility, checks are made whether the user who is going to use the system is able to work with the system. If the user does not understand the functionality of the system or is not able to work on the system further development is of waste. Modular approach can be adapted in this type of system where in as and when one module is done

it can be integrated with the existent prototype for user to access and become familiar with it.

➔ **Economical Feasibility:**

The software must be economically feasible i.e., its cost must be very low so that even small businesses can take advantage of our system to make their network more secure and robust.

4.5 REQUIREMENT VALIDATION:

In requirement validation we perform validation of all our requirements using a checklist. This step is performed to confirm that we have correctly gathered all the requirements and it will suffice all the needs of the consumer. It shows that the requirement defines the system just as the user wants it to be. If validation is not performed the errors in requirements will be carried forward to the design and implementation phases too. Requirements are verified to find out if they are complete, persistent and is according to what users want for their proposed system.

4.6 FUNCTIONS OF SYSTEM:

4.6.1 USE CASES:

Use case diagrams are generally used for describing actions performed by some user within a specified system boundary. In this a set of use cases i.e. actions that can be performed are drawn inside a rectangular box which describes a system boundary. The persons who will use the system are called actors and are drawn outside the boundary.

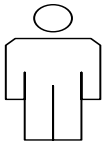
Followings are the components of use case diagram:

- **Use Case:** A Use Case shows some possible action of the system which can be accessed by the users. It is drawn inside system boundary.

- **Actors:** The people who will be using our system are shown using actor symbols. These symbols are drawn outside the system boundary.

- **Associations:** Association between use cases and actors are displayed in usecase diagrams by using solid lines. Association is used whenever a person interacts with the system functions.

Table 4-1 Usecase Symbols & Meanings

Symbol	Meaning
	System Boundary
	Actor
	Extend
	Association
	Include
	Generalization

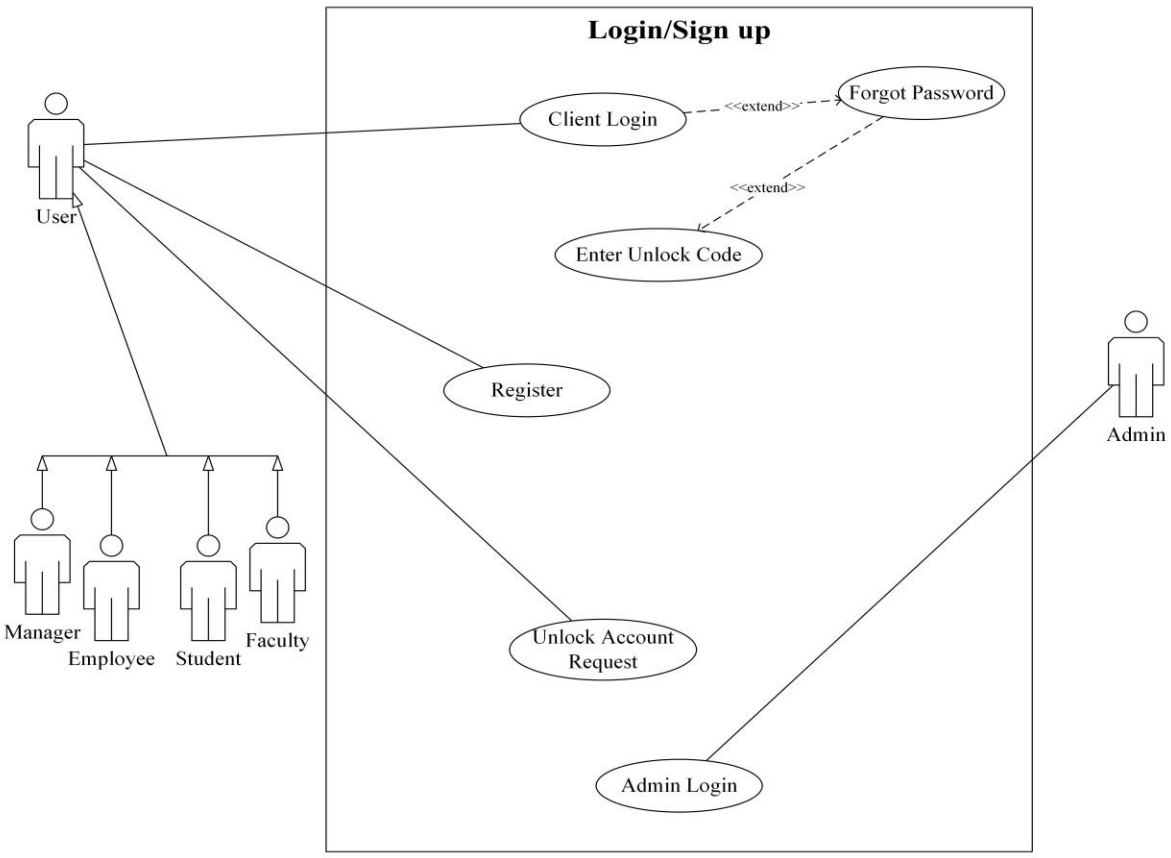


Fig 4.1 Usecase login

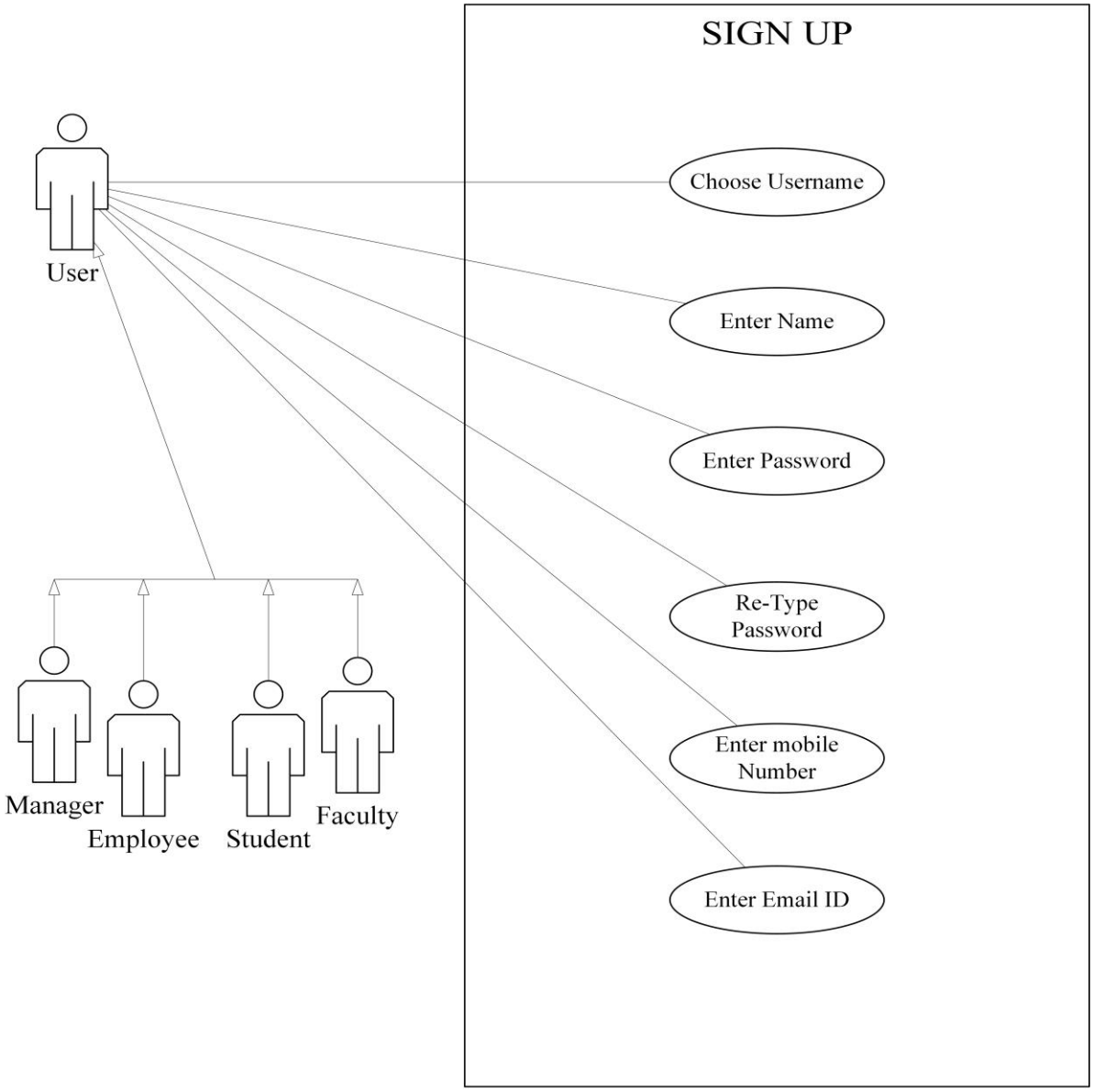


Fig 4.2 Usecase signup

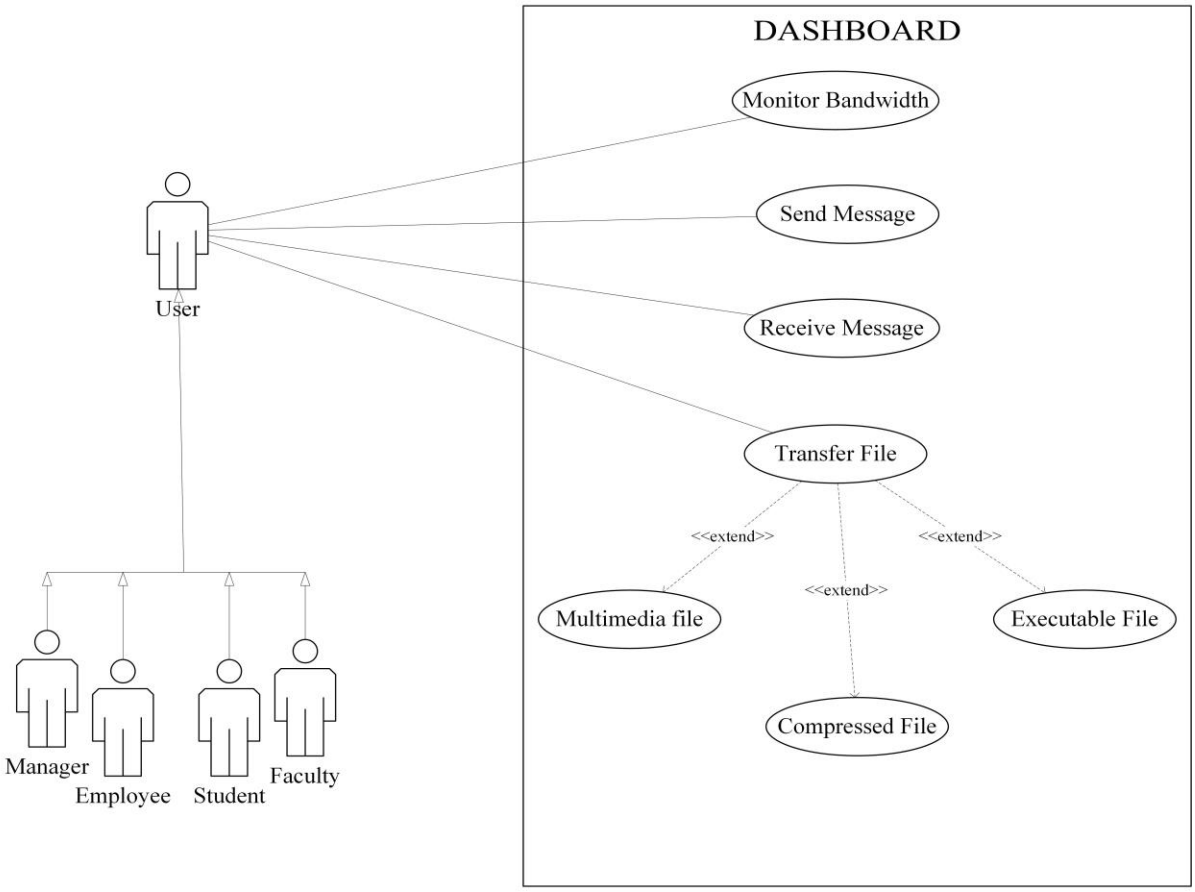


Fig 4.3 Usecase user dashboard-1

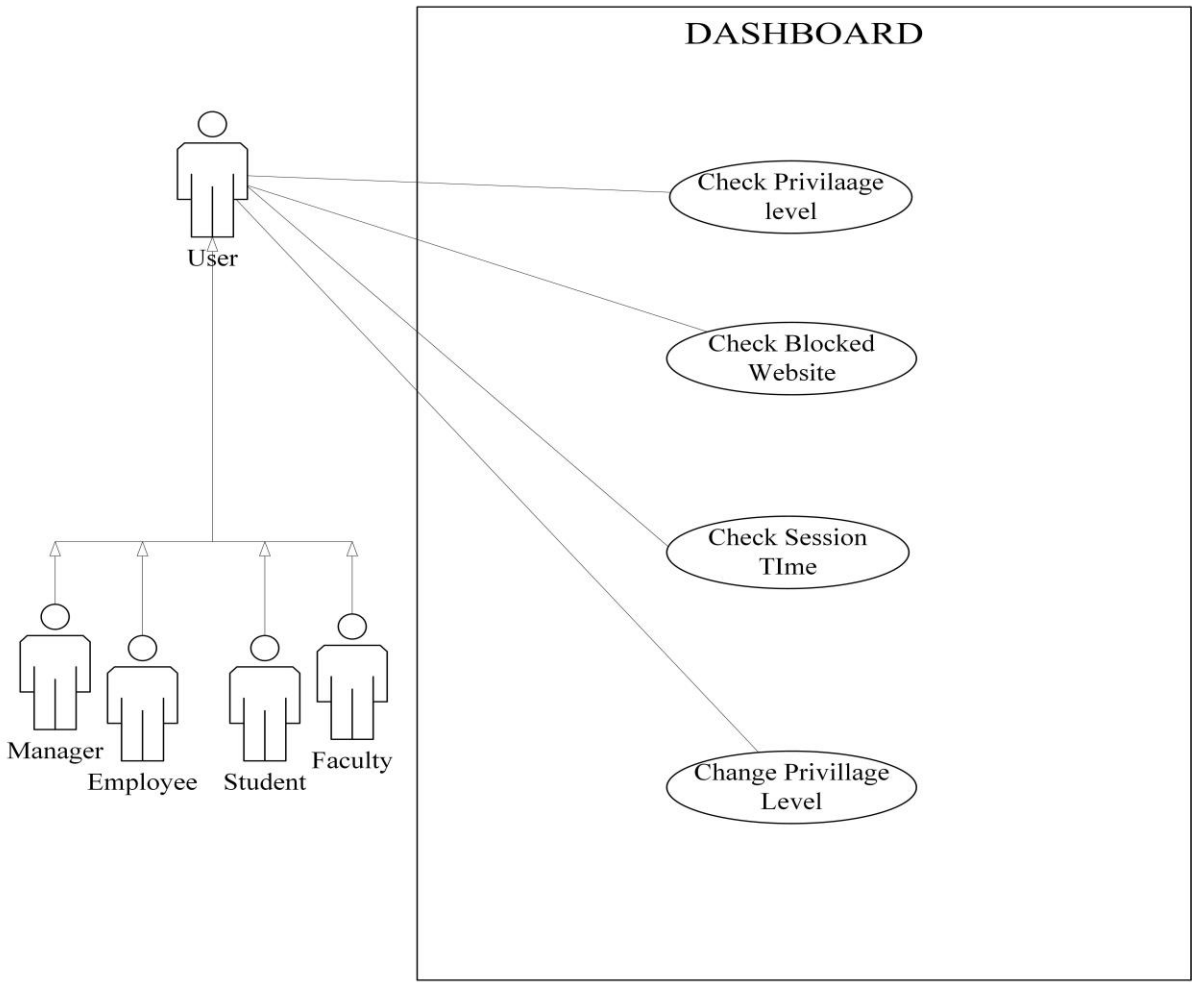


Fig 4.4 user dashboard 2

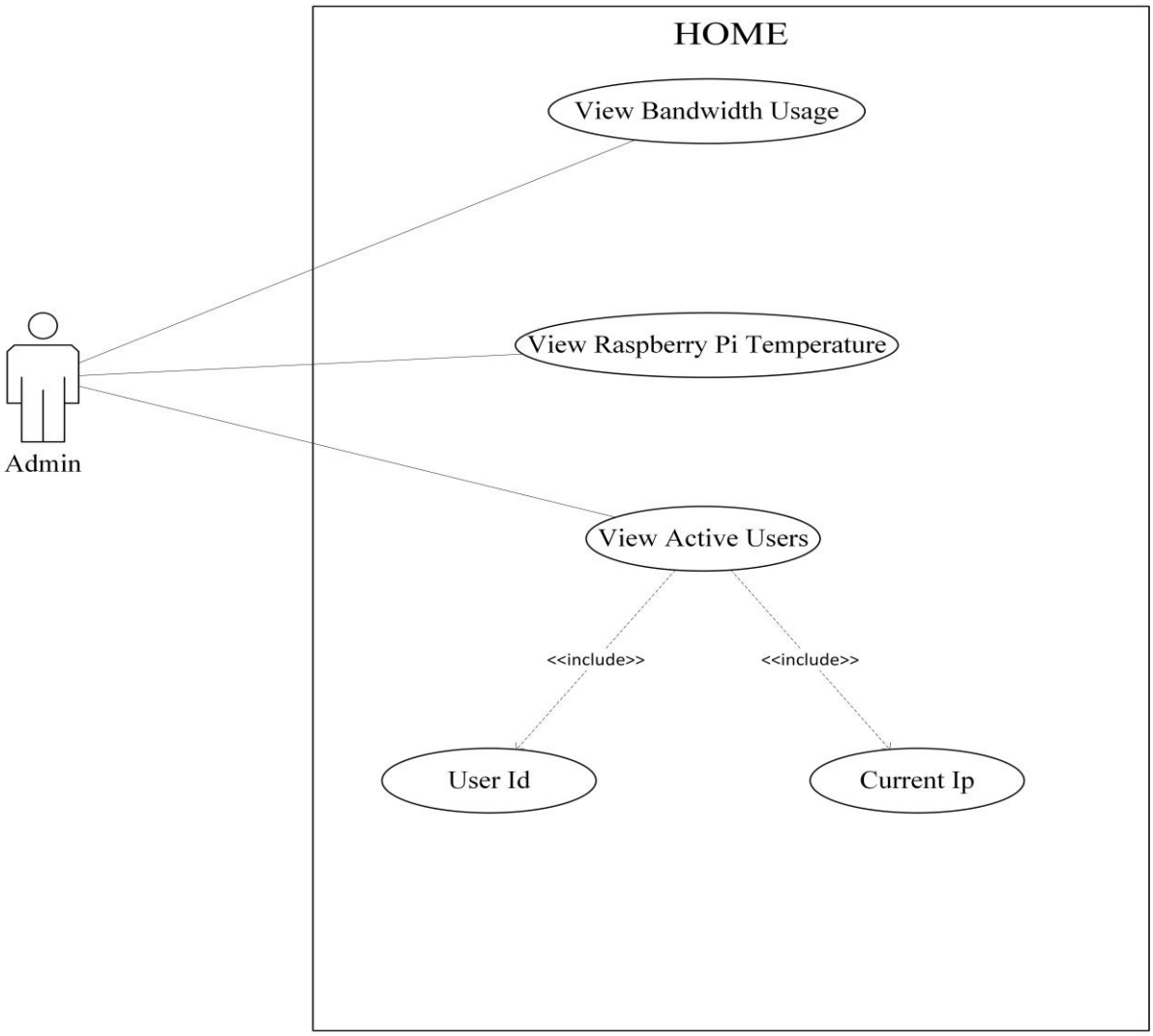


Fig 4.5 usecase admin home 1

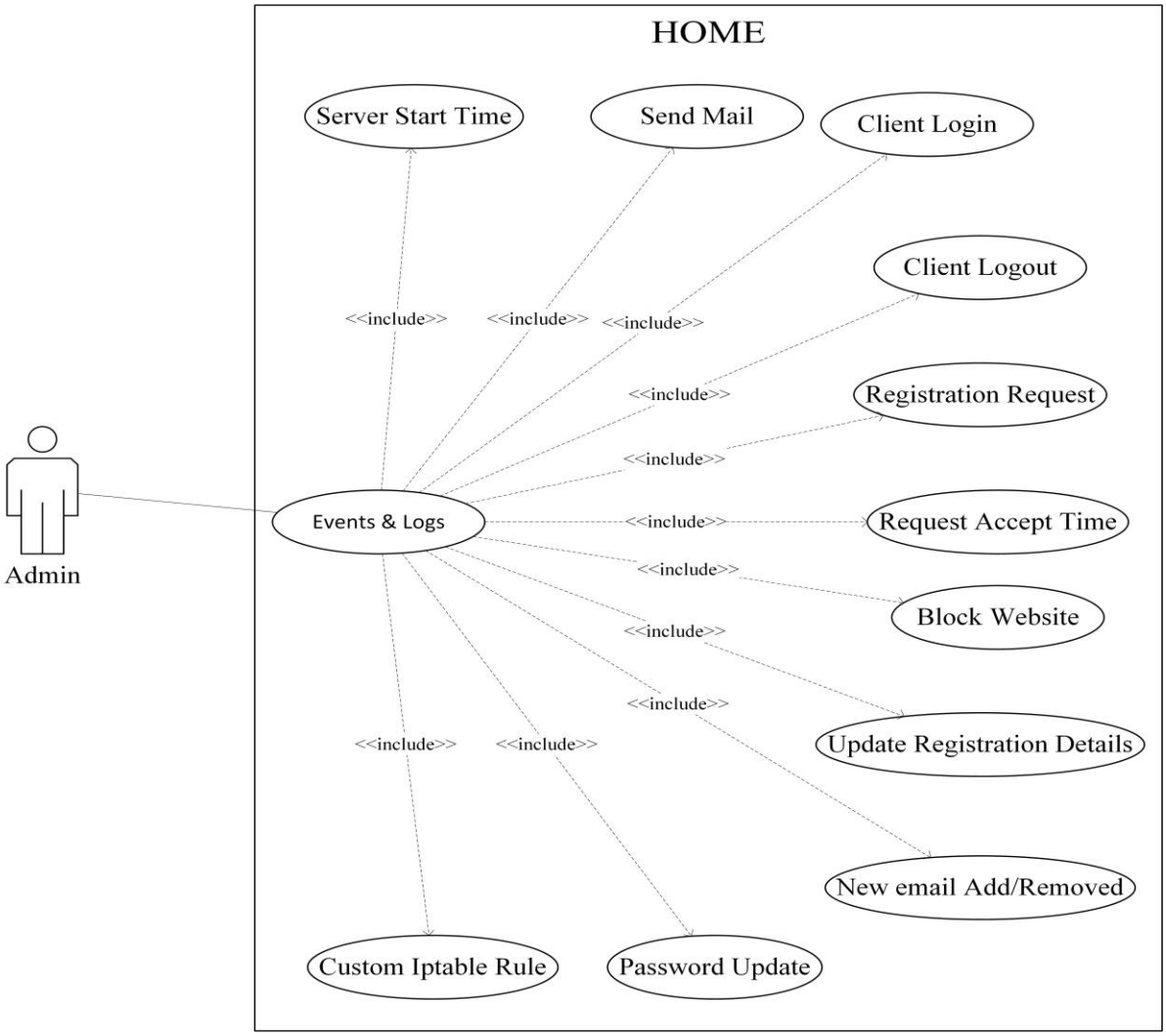


Fig 4.6 usecase admin home 2

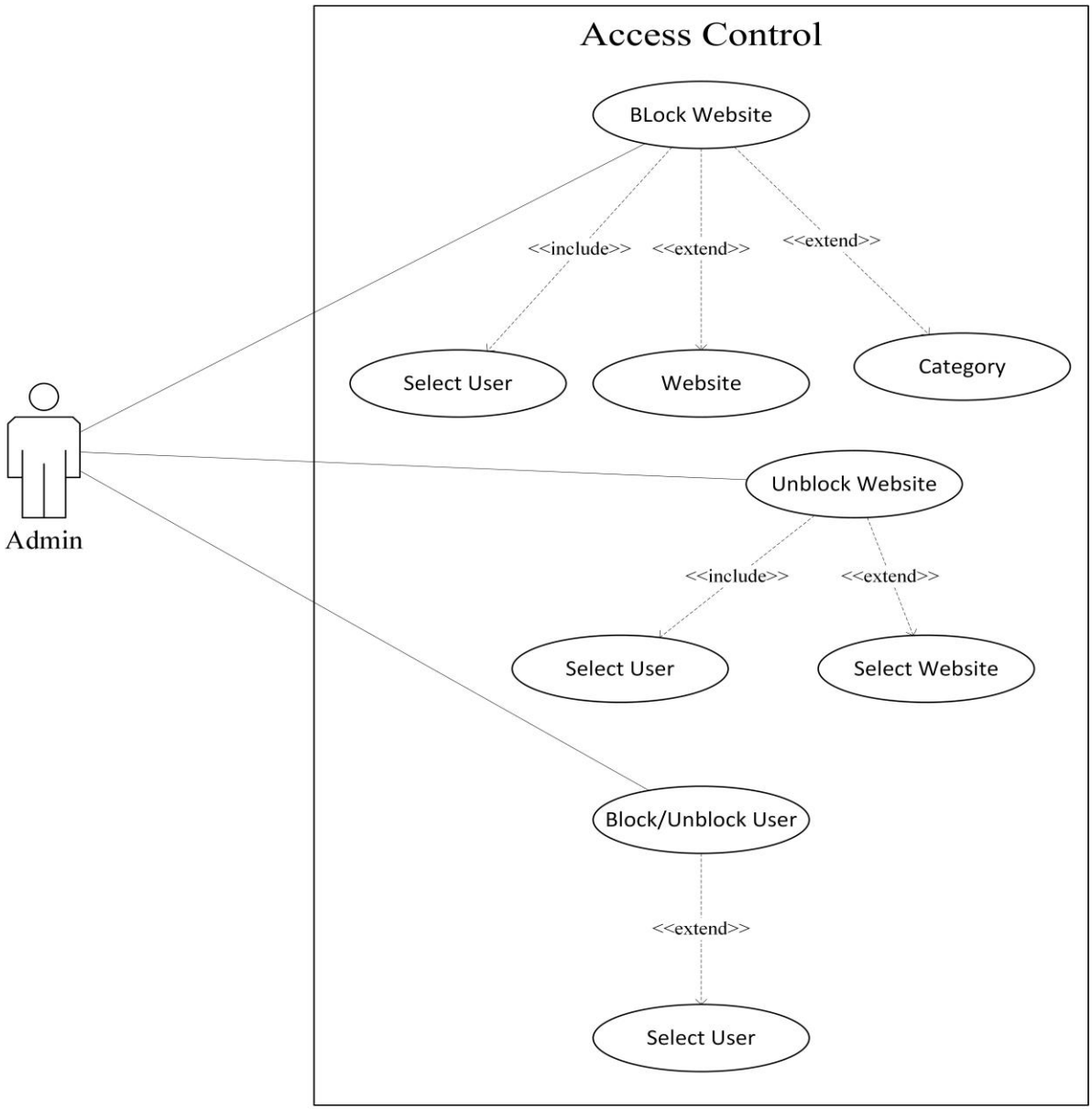


Fig 4.7 usecase admin access control

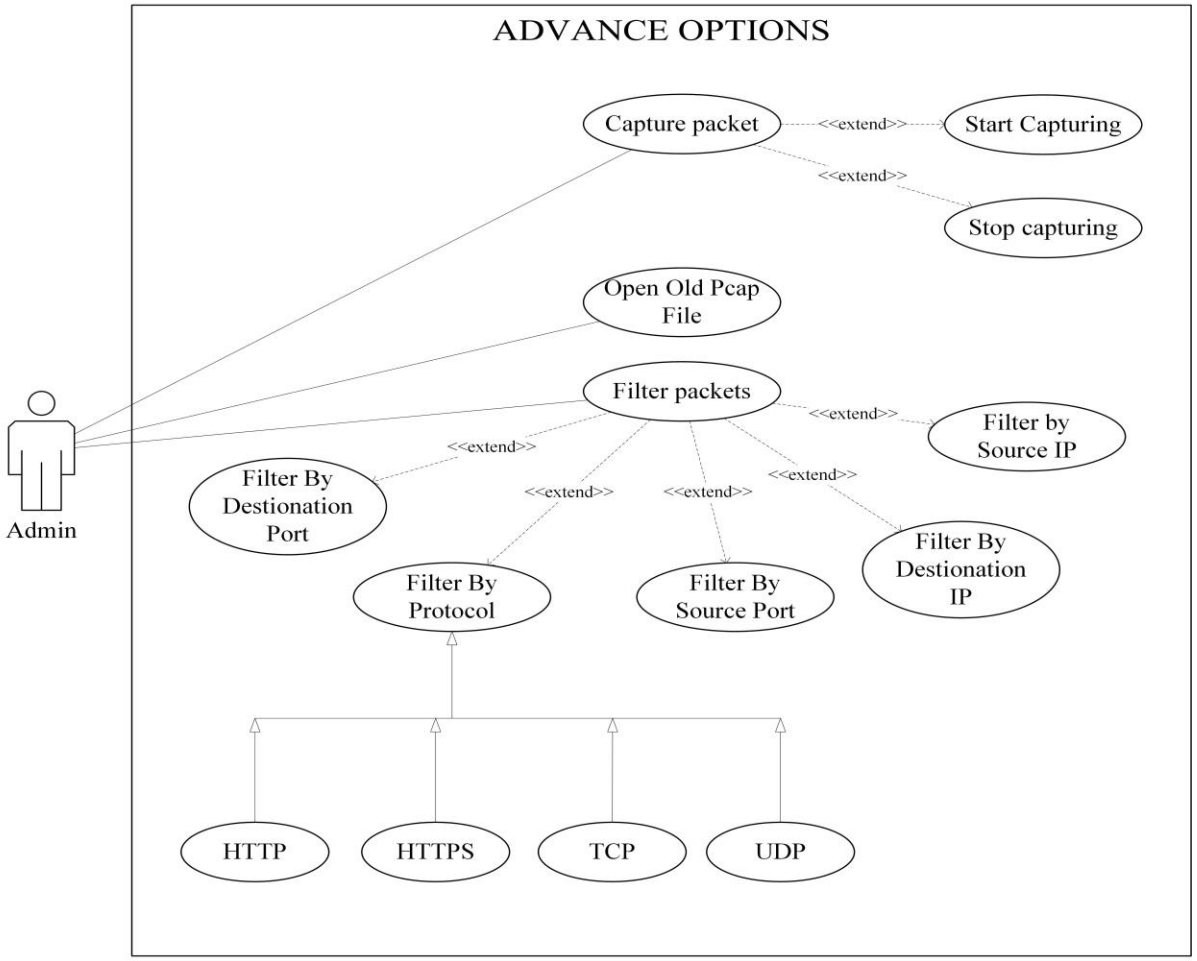


Fig 4.8 usecase admin advance options 1

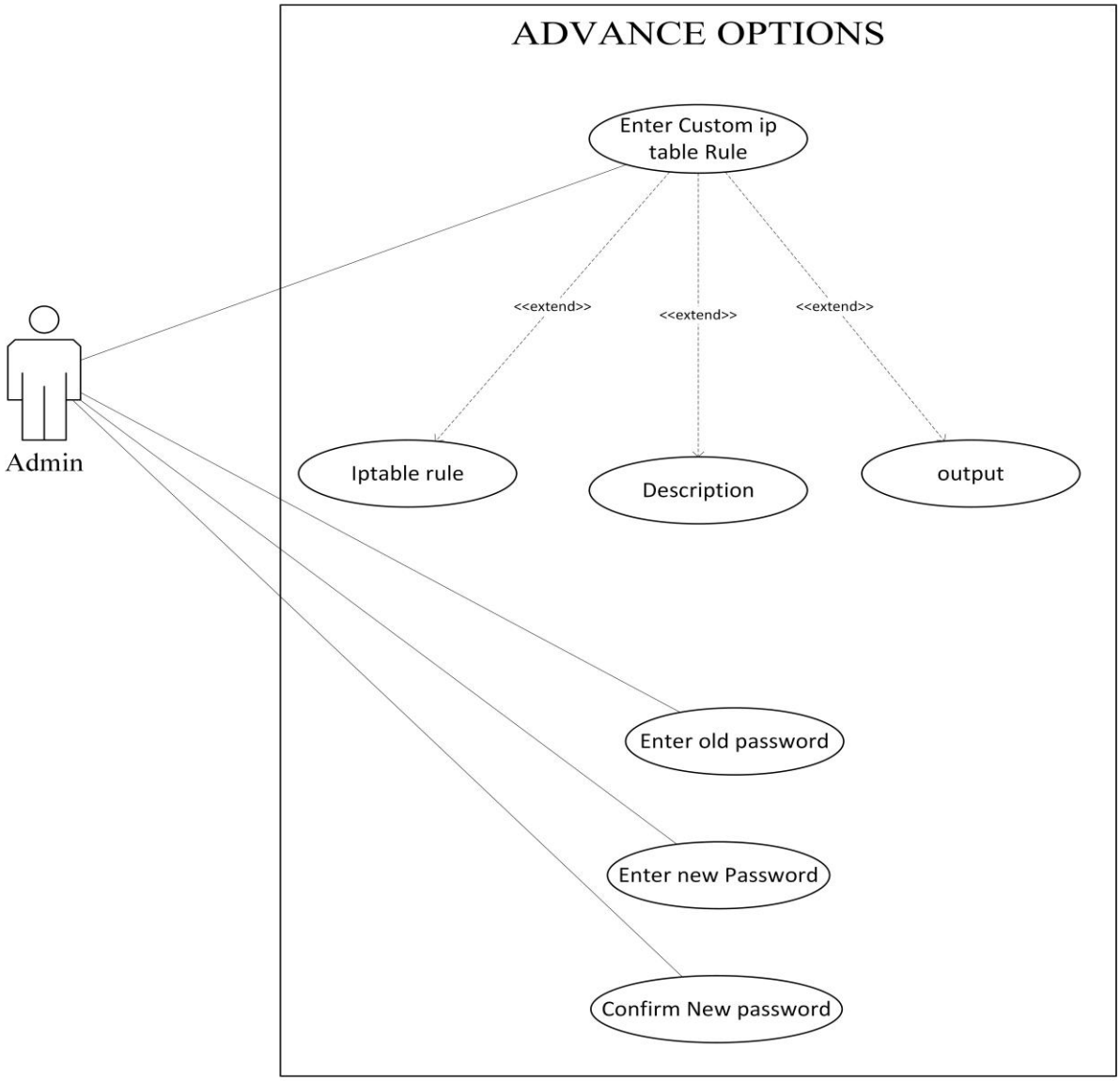


Fig 4.9 usecase admin advance options 2

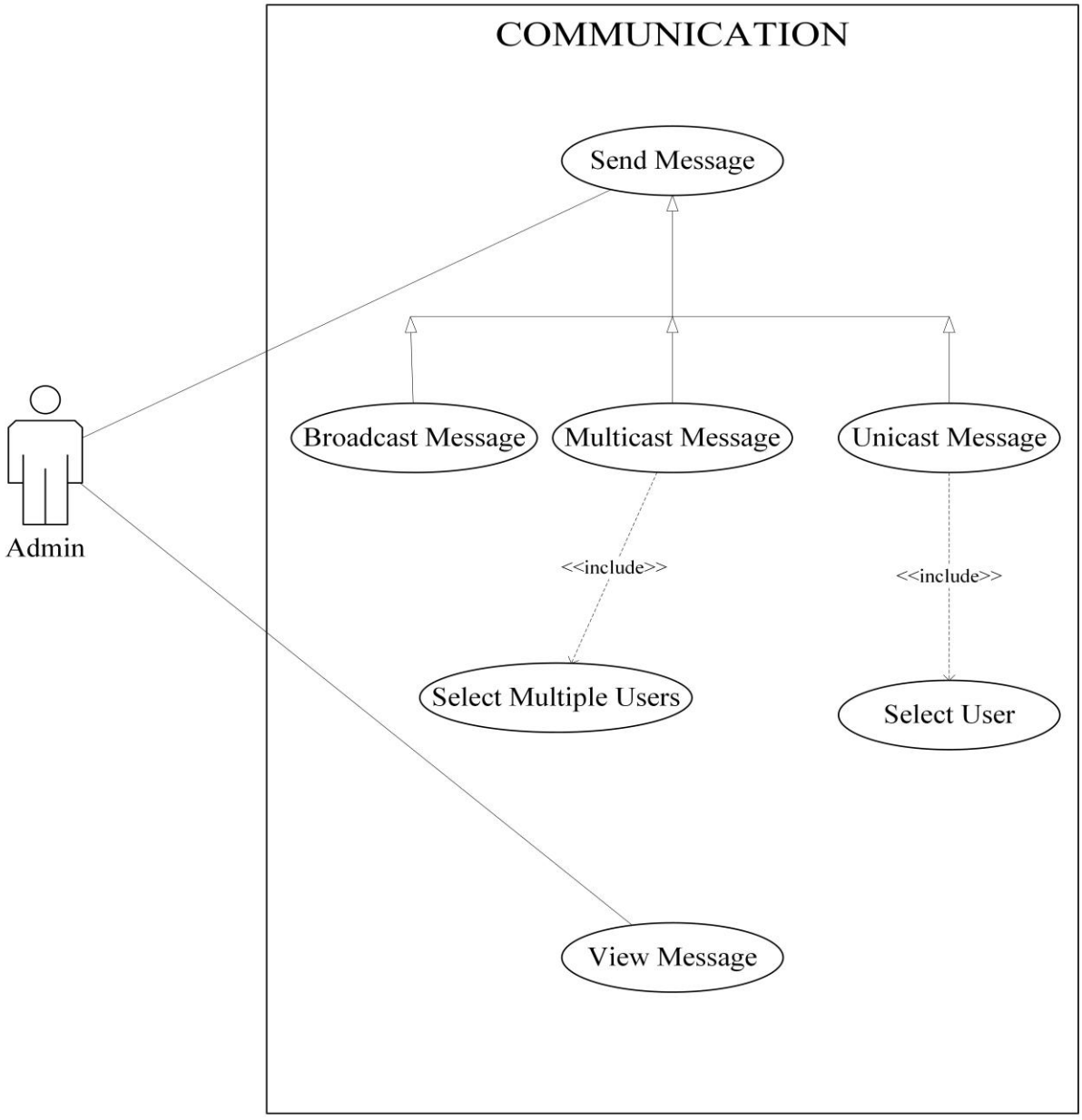


Fig 4.10 usecase admin communication

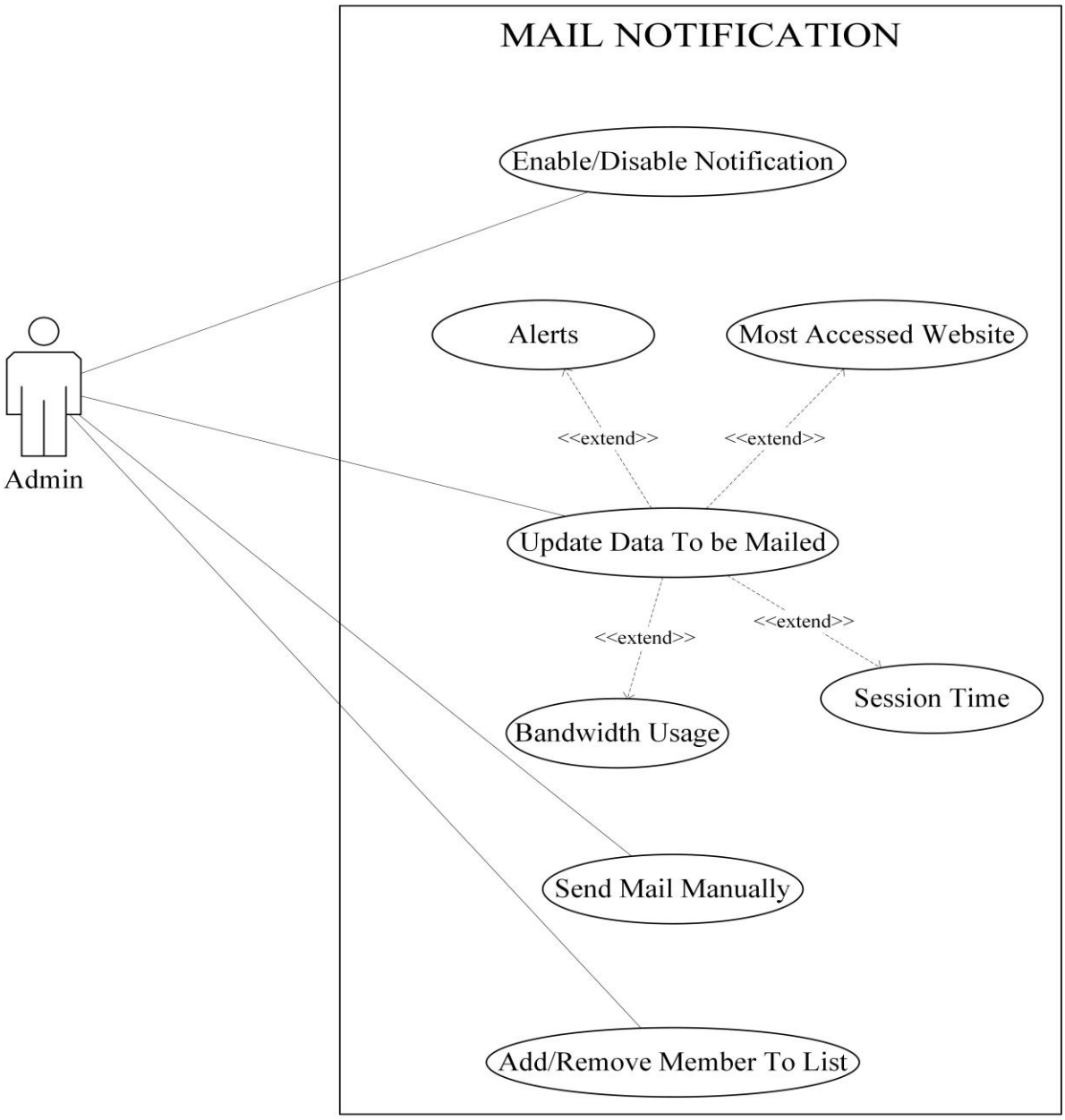


Fig 4.11 usecase admin mail notification

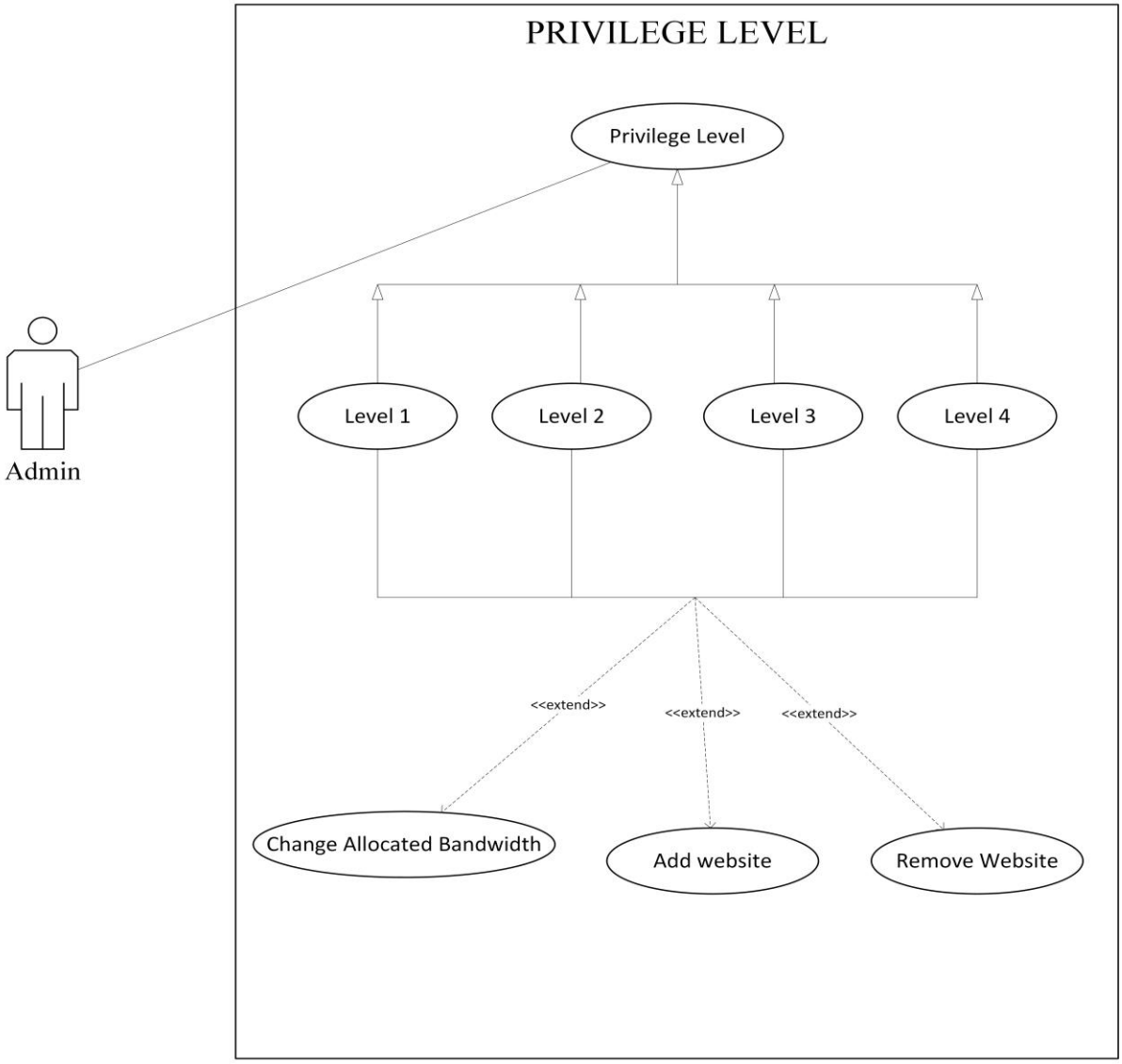


Fig 4.12 usecase admin privilege level

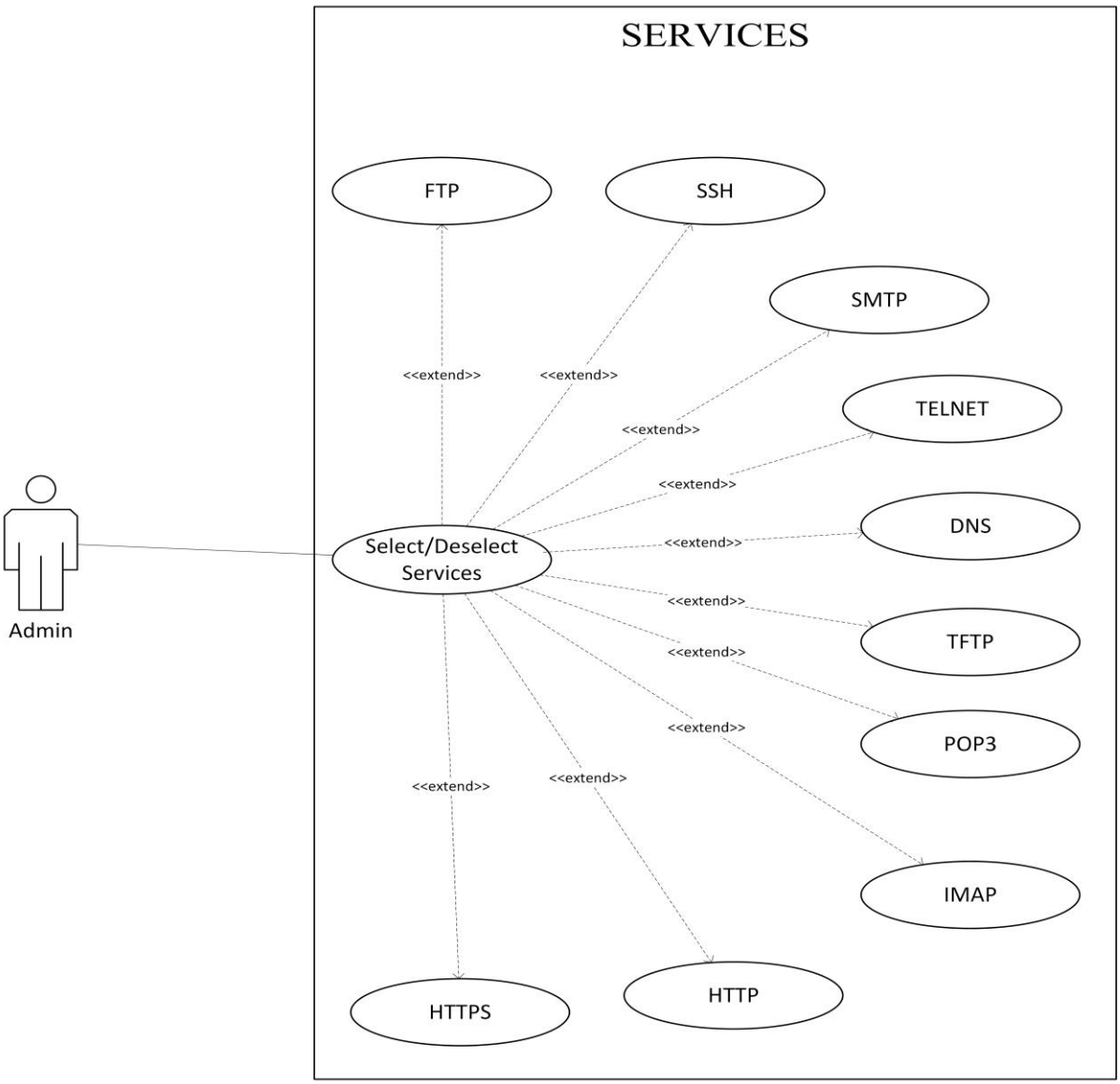


Fig 4.13 usecase admin services

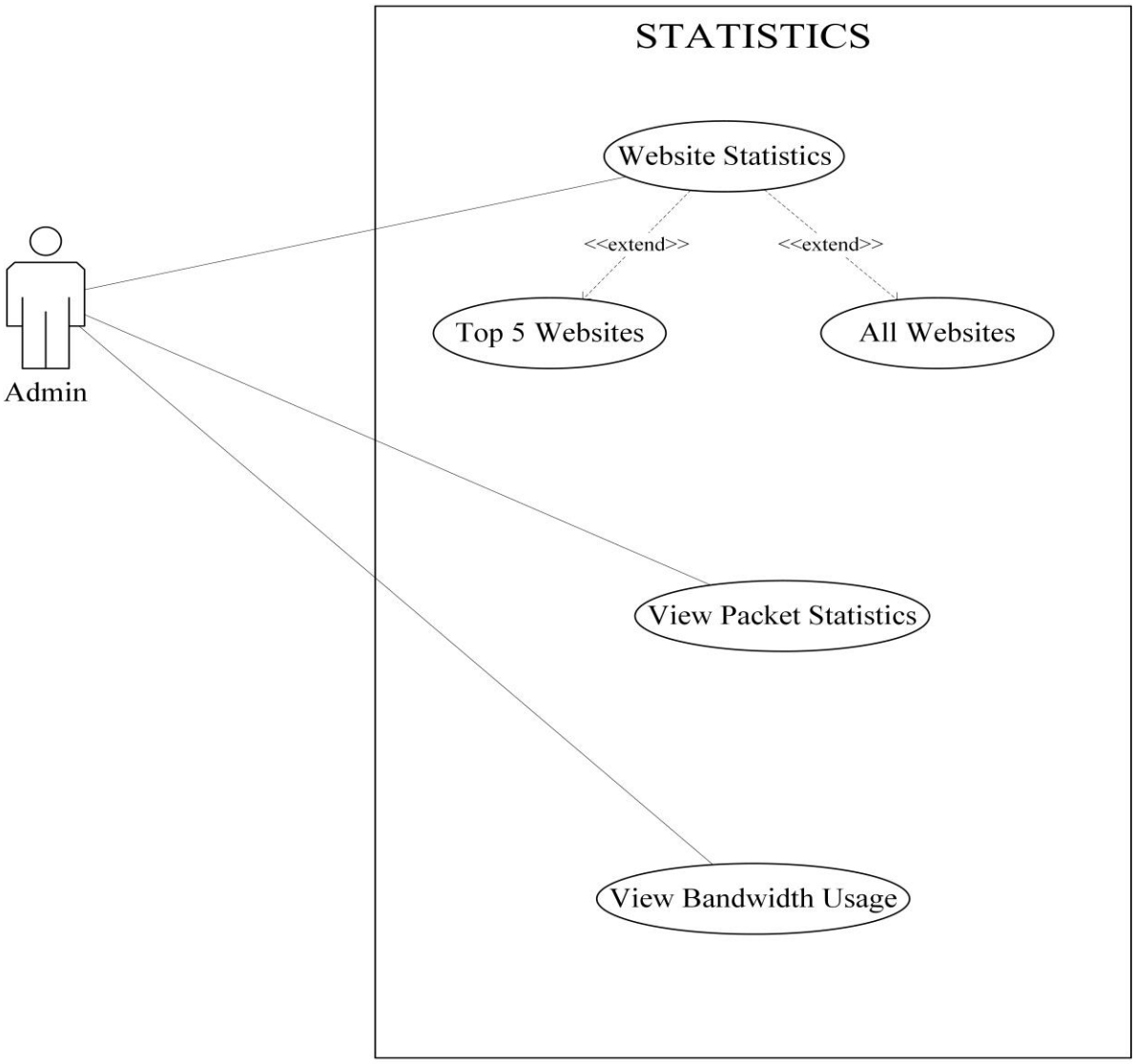


Fig 4.14 usecase admin statistics

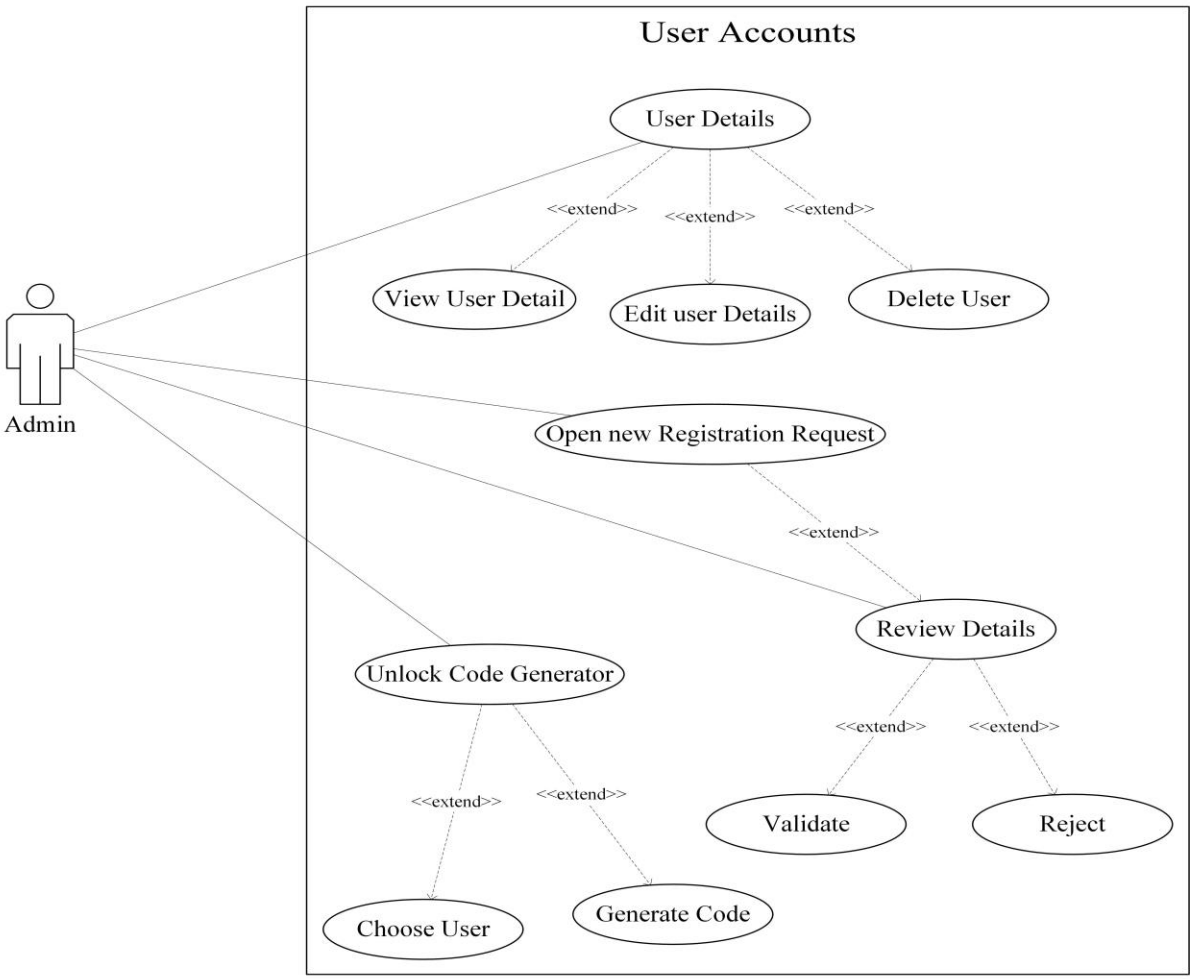


Fig 4.15 usecase admin user accounts

4.7 DATA MODELING

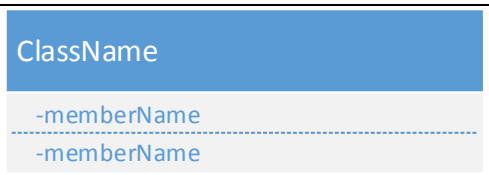
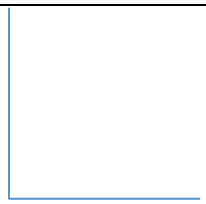
4.7.1 CLASS DIAGRAM:

It is used for describing structure and behavior in the use cases. It provides a conceptual model of the system in terms of entities and their relationships.

Following Modifiers are used to indicate visibility of attributes and operations.

- '+'is used to denote *Public* visibility (everyone)
- '#'is used to denote *Protected* visibility (friends and derived)
- '-' is used todenote *Private* visibility (no one)

Table 4-2 Class Symbols & Meanings

Symbol	Meaning
	Class
	Association
	Composition
-memberName	member

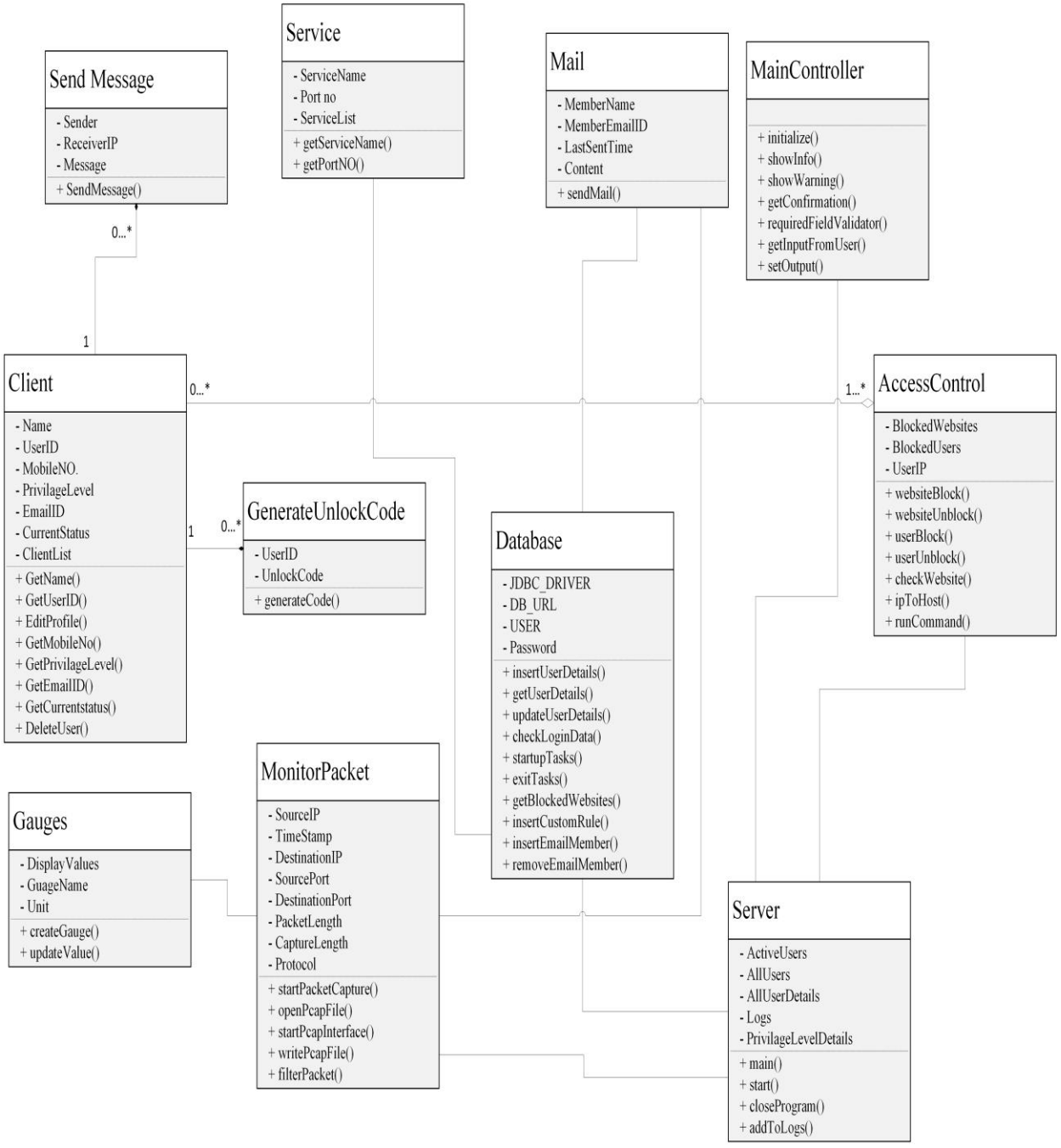
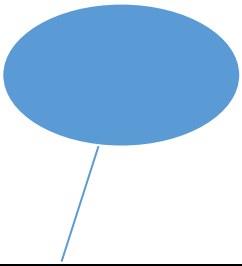
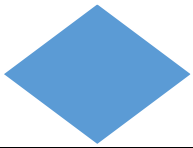
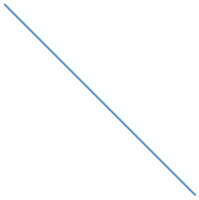


Fig 4.16 Class Diagram

4.7.2 ENTITY RELATIONSHIP DIAGRAM:-

Entity–relationship model is a data model giving information about business domain and its process requirements, in a way through which we can easily execute it into a database.

Table 4-3 Entity Relationship Symbols & Meaning

Symbol	Meaning
	Entity
	Attribute
	Relationship
	Relationship Connector

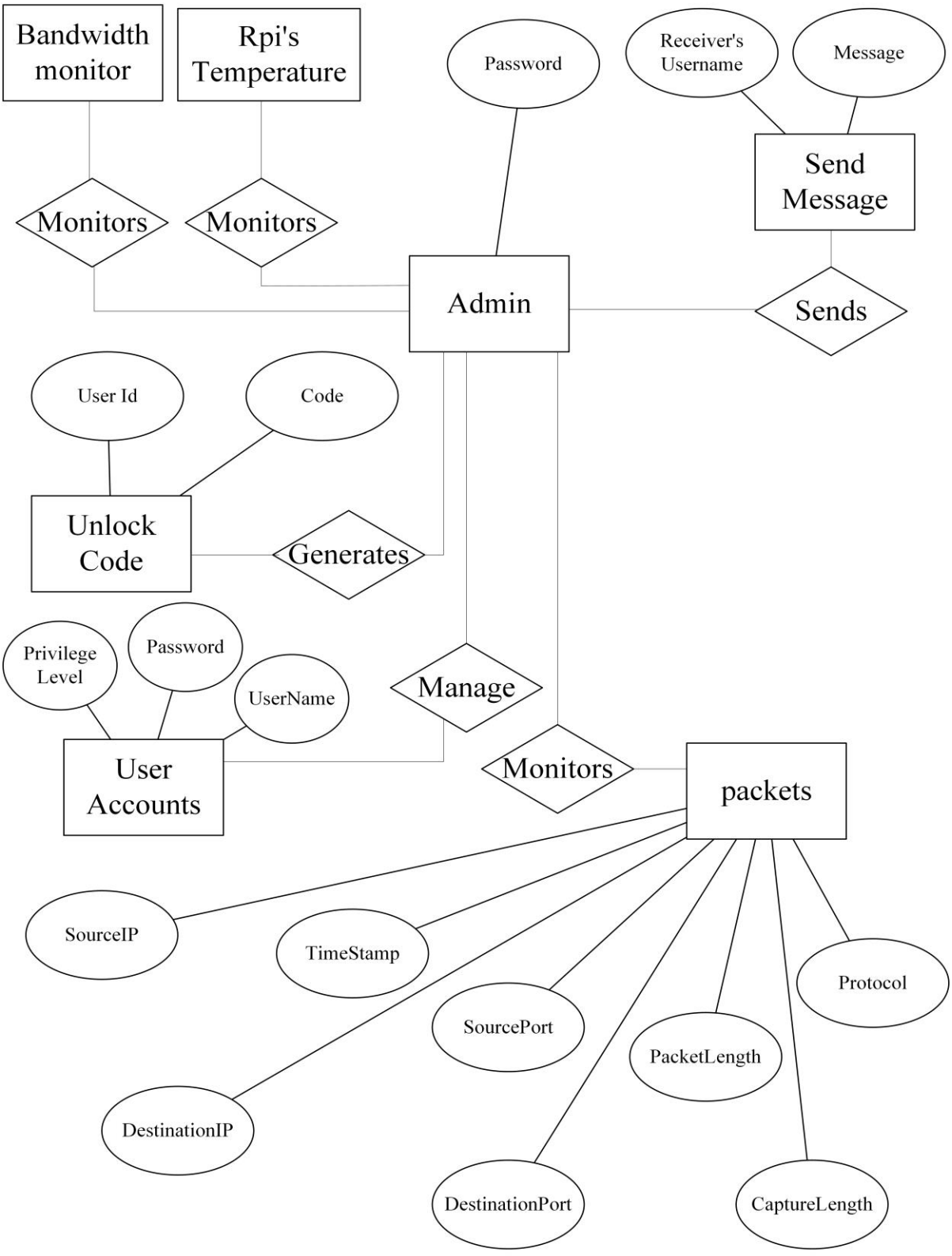


Fig 4.17 ER Diagram Admin 1

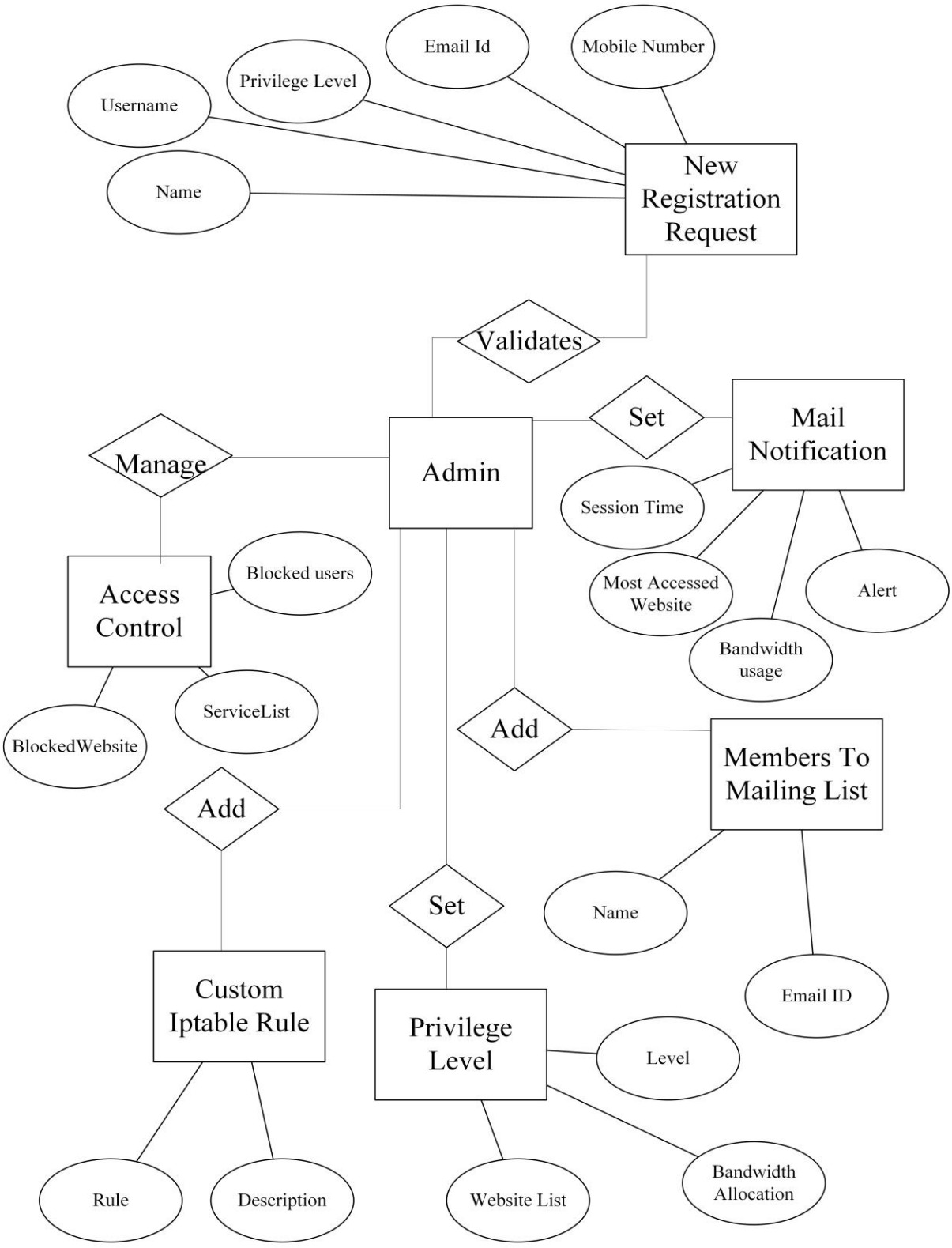


Fig 4.18 ER Diagram Admin 2

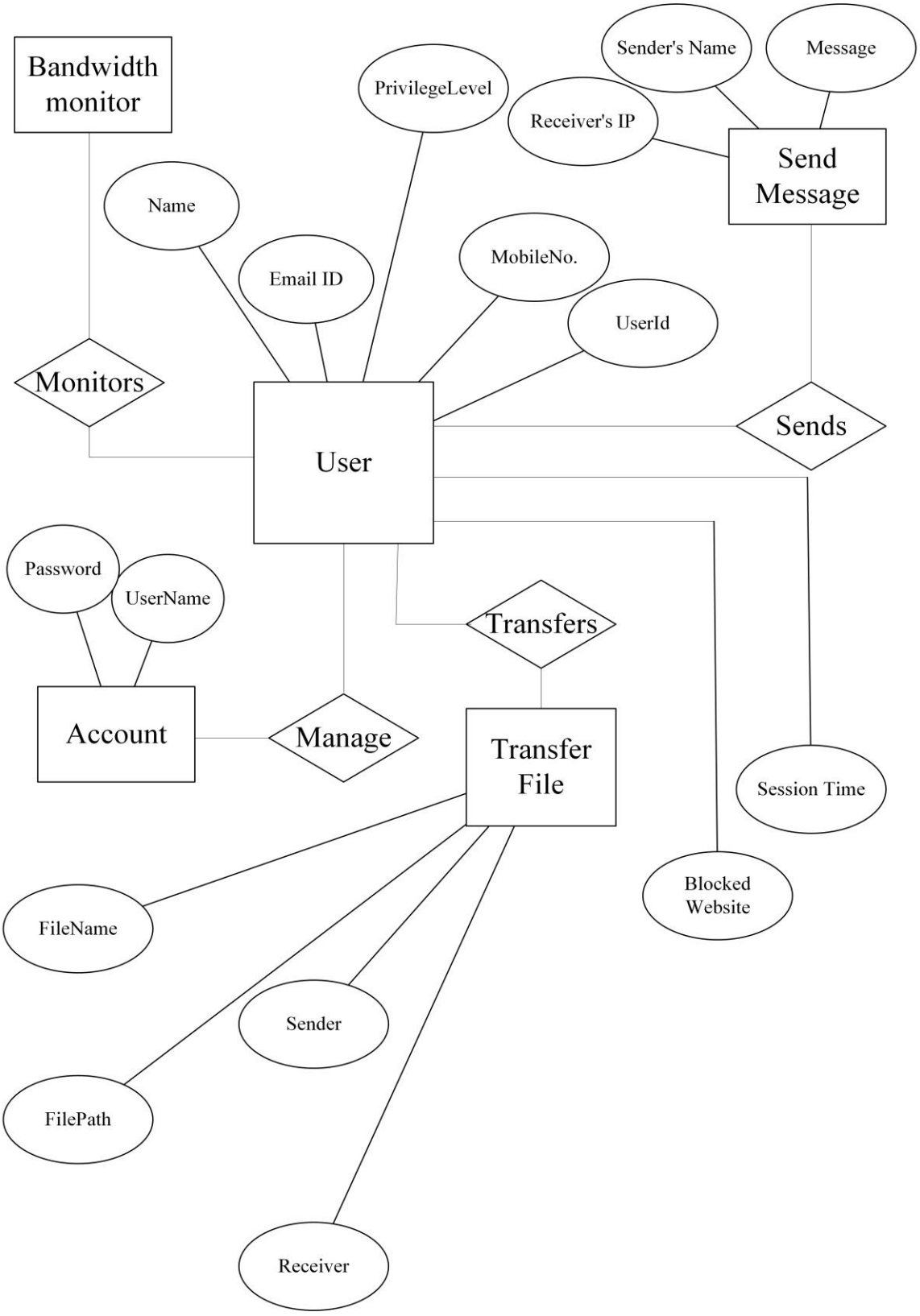


Fig 4.19 ER Diagram User

4.7.3 ACTIVITY DIAGRAM:

- Activity diagrams are used to represent the workflow of the activities of our system. It show how transition from one state to another occur and what all step we go through to reach the end result.

- In UML, activity diagrams are made to show all functional, non-functional and logical processes of the system. Activity diagram therefore displays the flow control of the entire system

Table 4-4 Activity Diagram Symbols

Symbol	Meaning
	Action
	Relationship
	Fork
	Join
	Initial State
	Final State

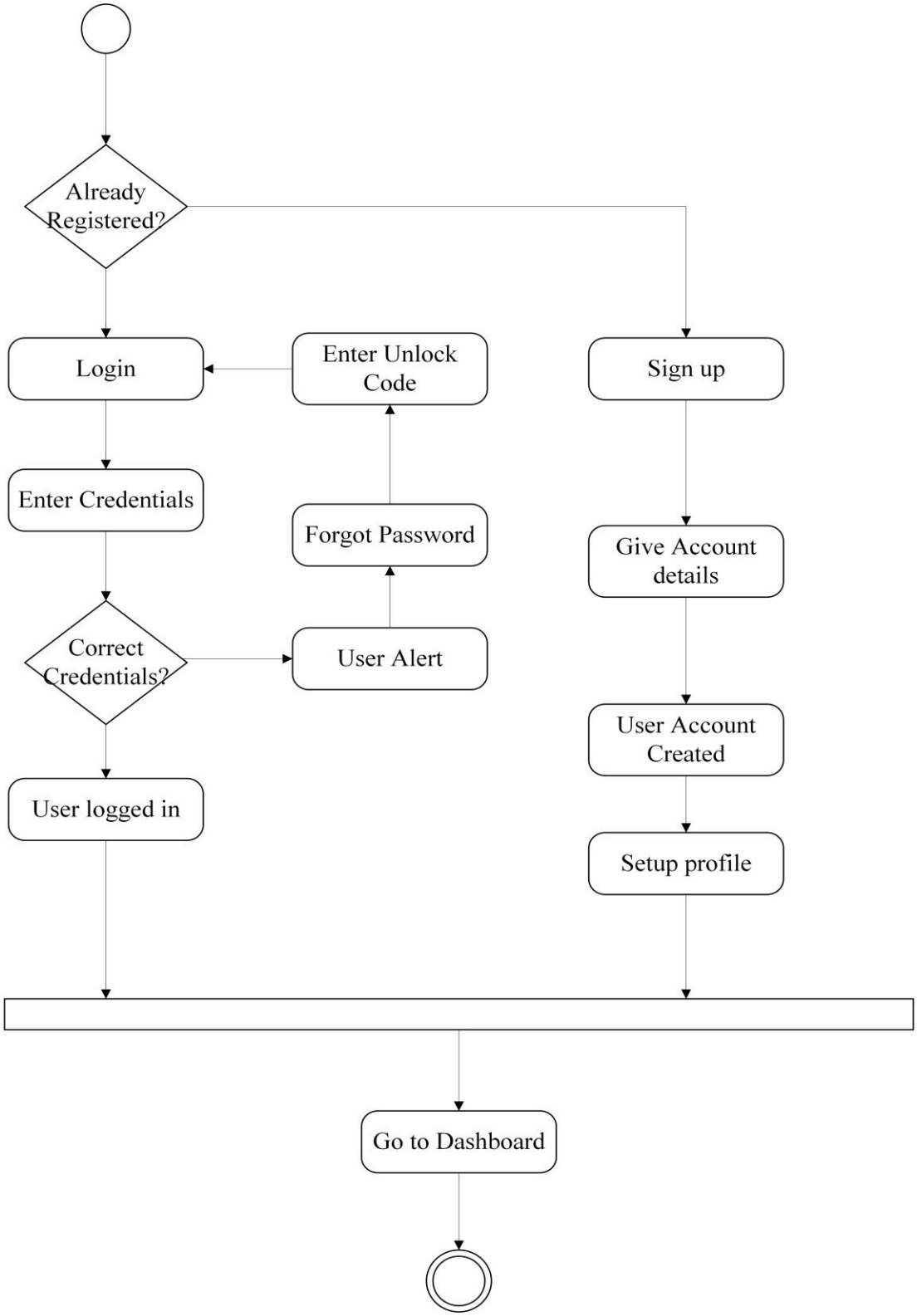


Fig 4.20 Activity user Login

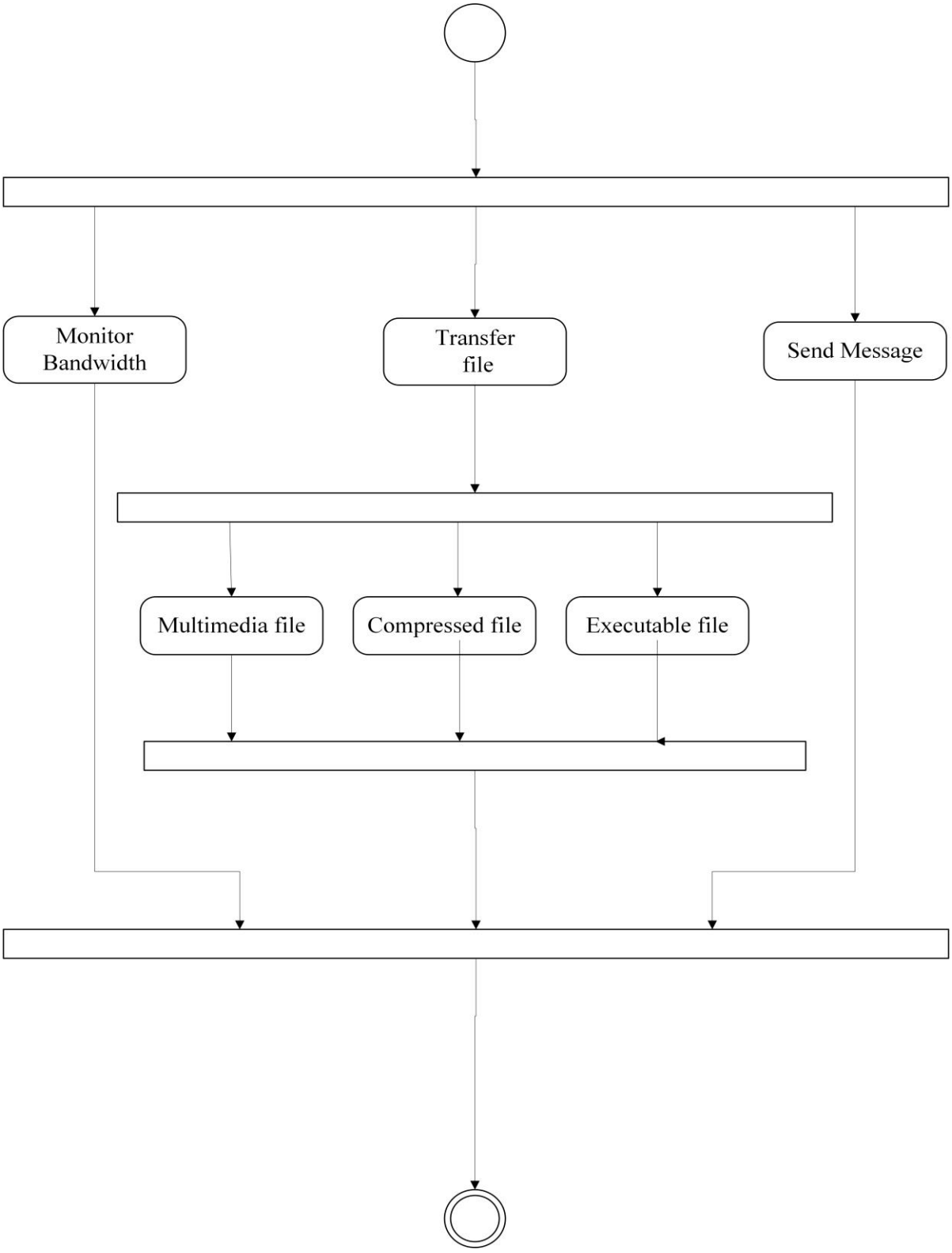


Fig 4.21 Activity user dashboard 1

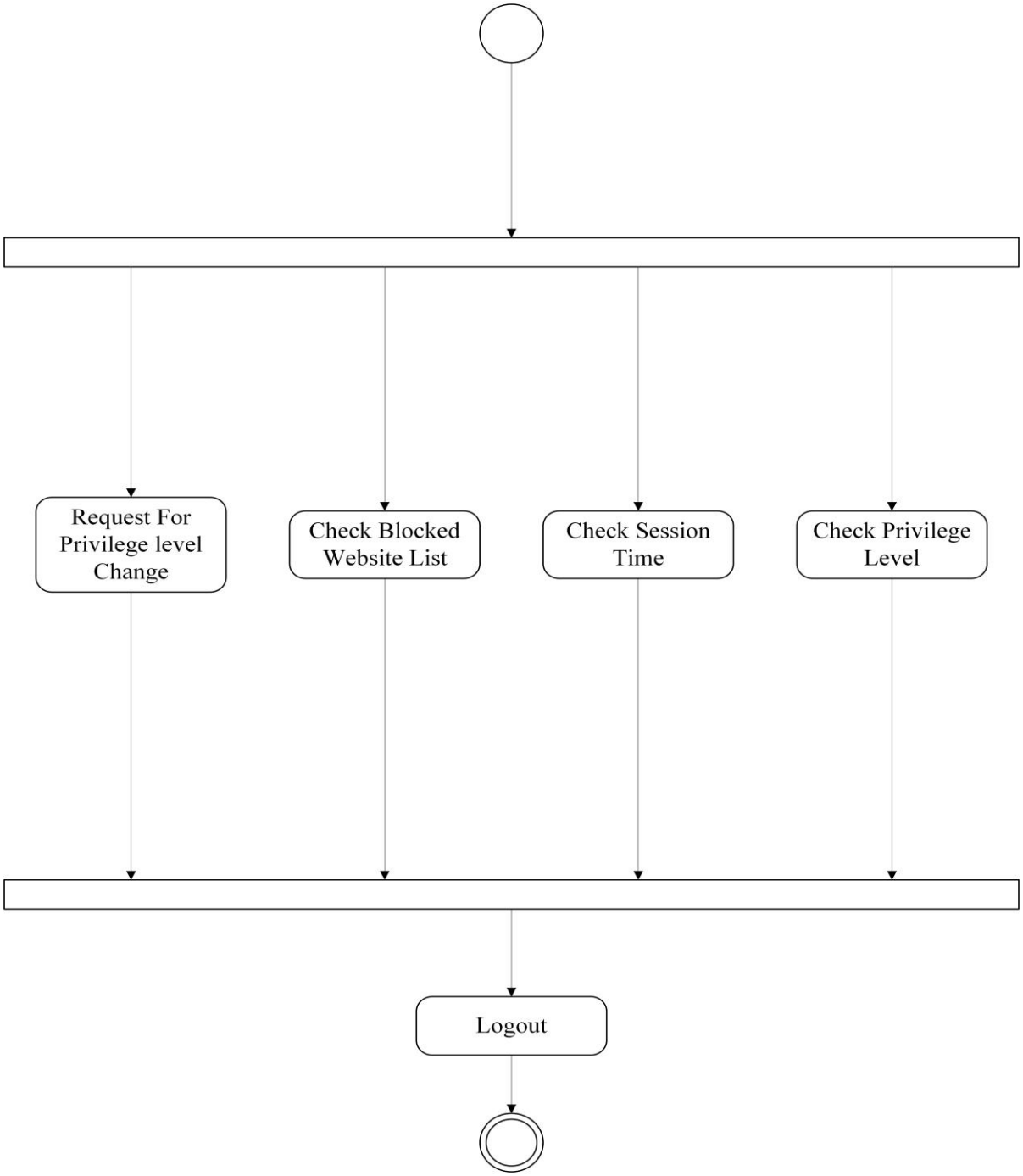


Fig 4.22 Activity user dashboard 2

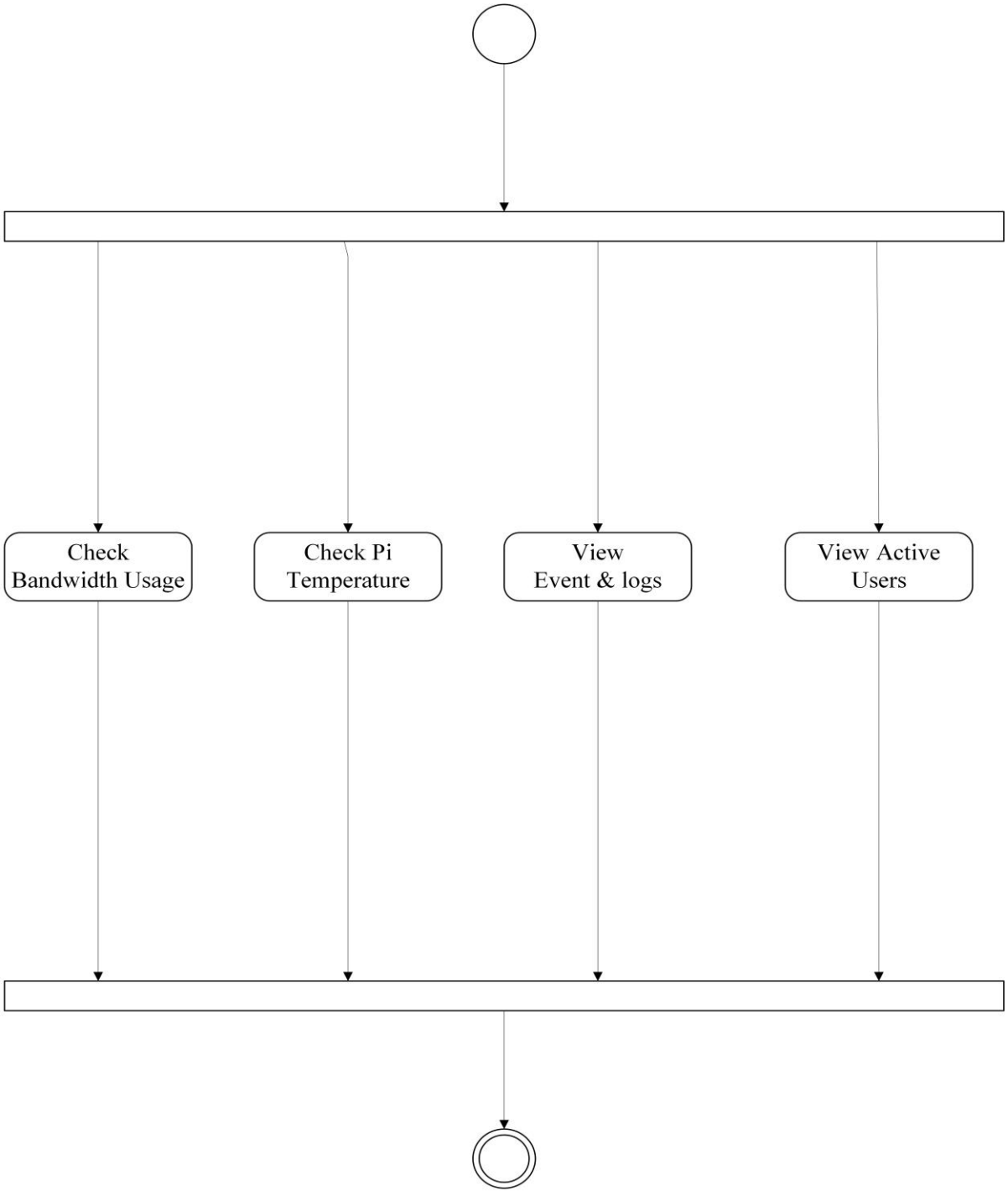


Fig 4.23 Activity Admin Dashboard

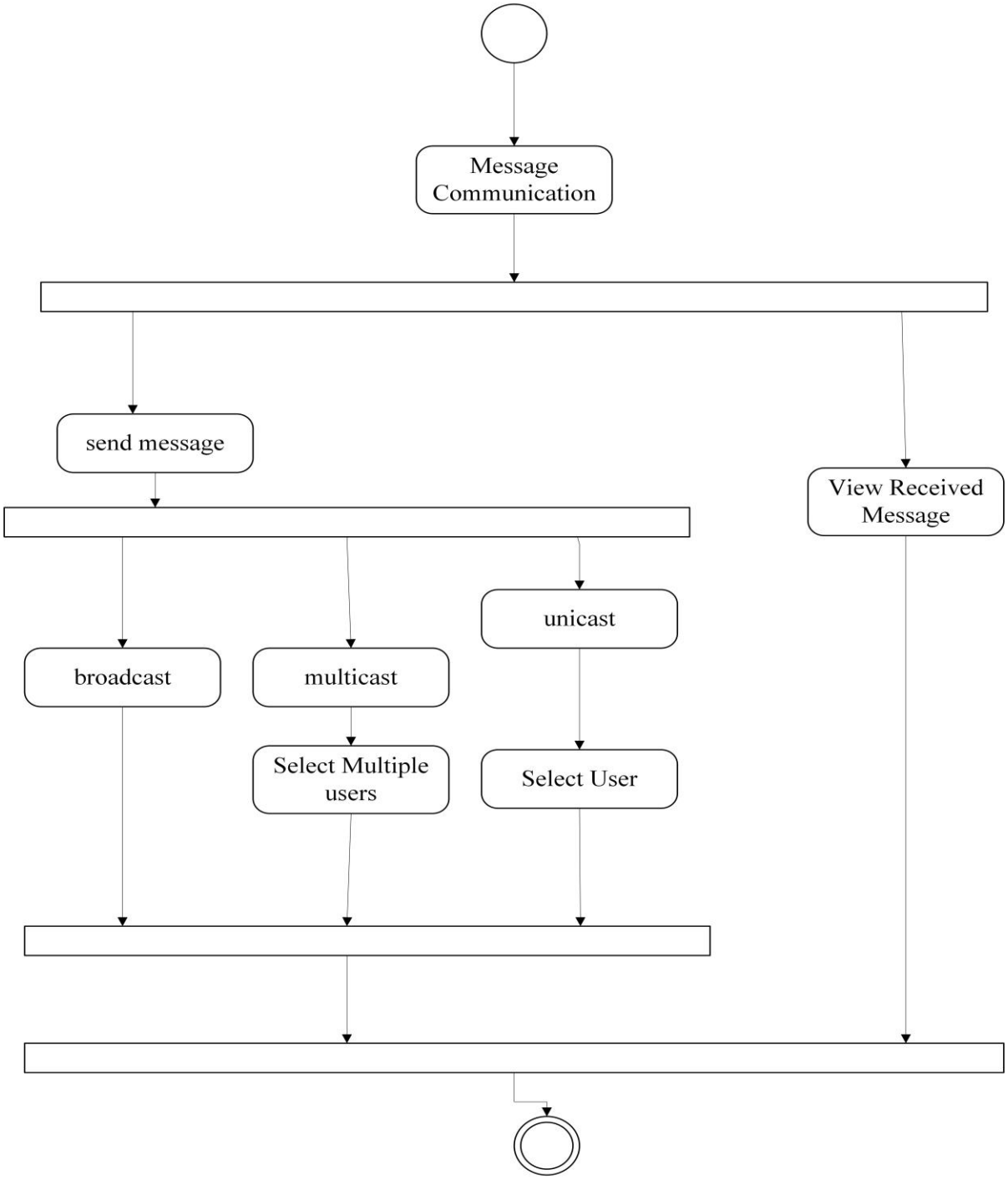


Fig 4.24 Activity Admin Communication

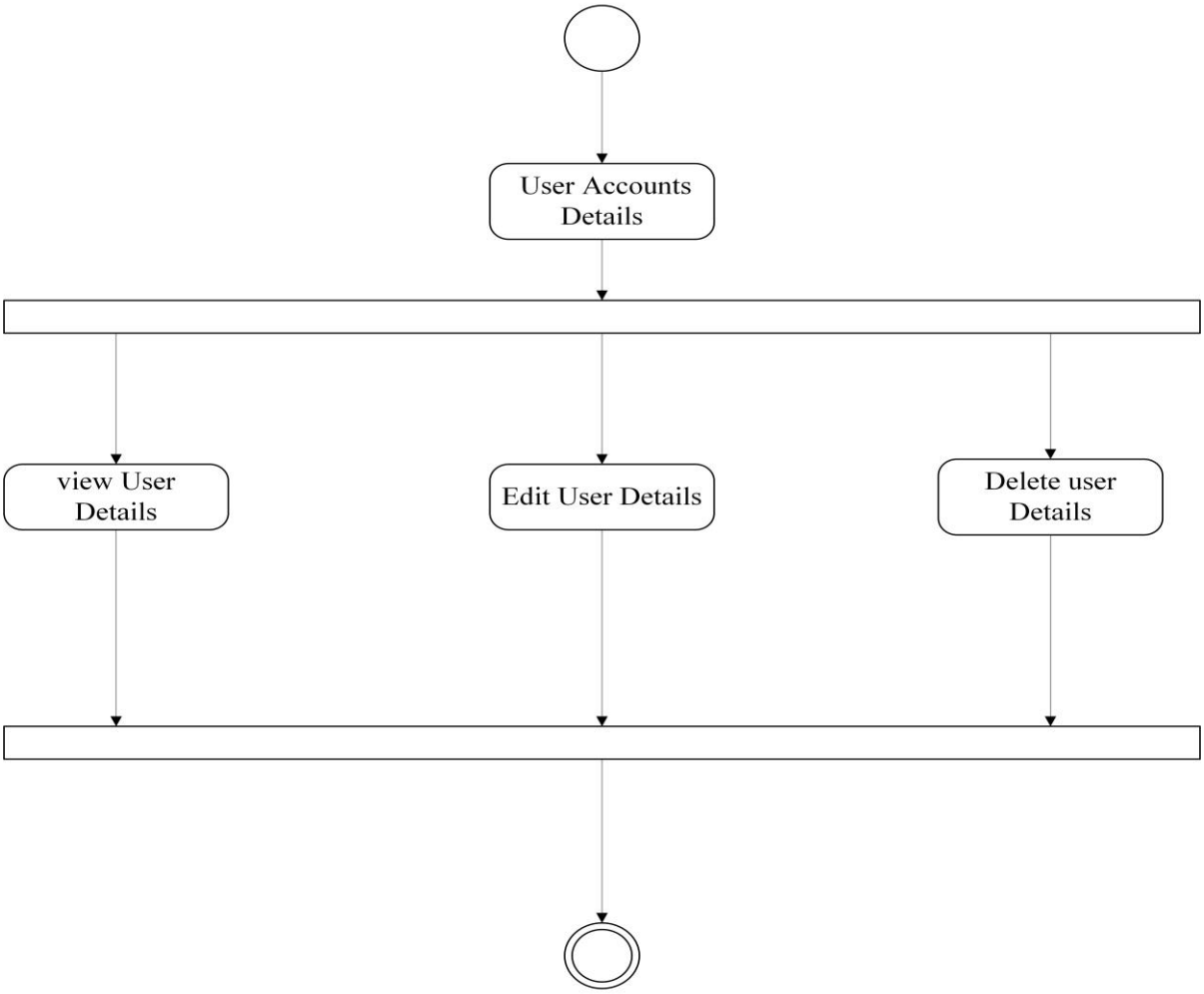


Fig 4.25 Activity Admin User Accounts Details

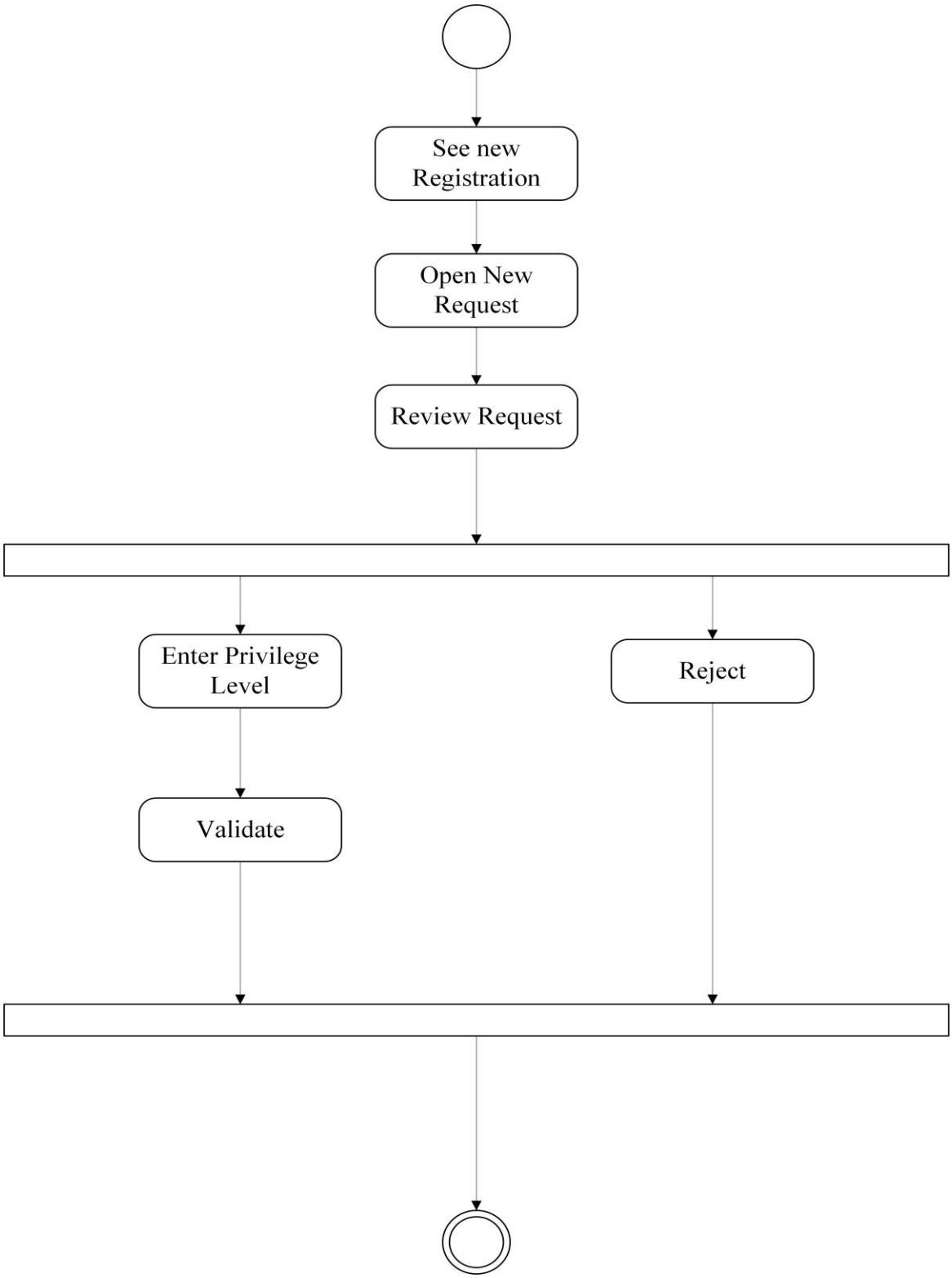


Fig 4.26 Activity Admin New Registration

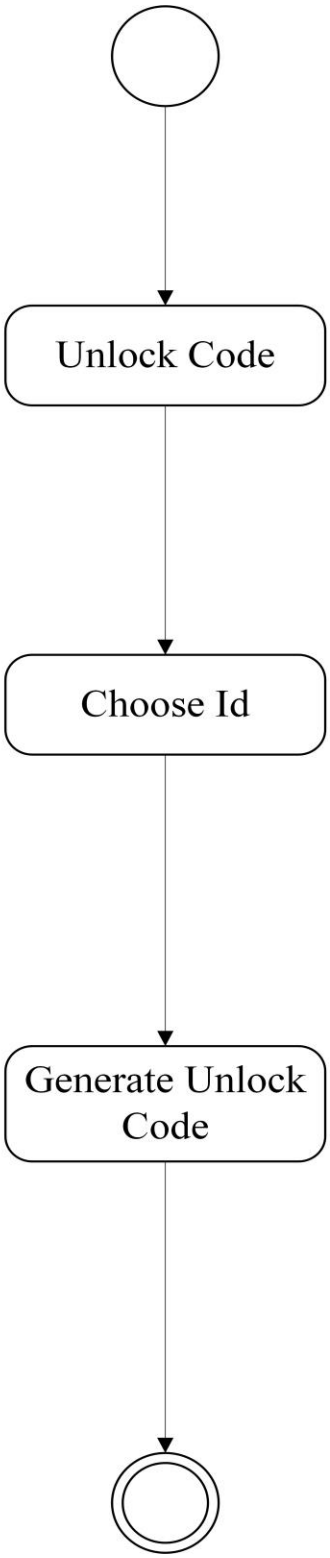


Fig 4.27 Activity Admin Unlock Code generator

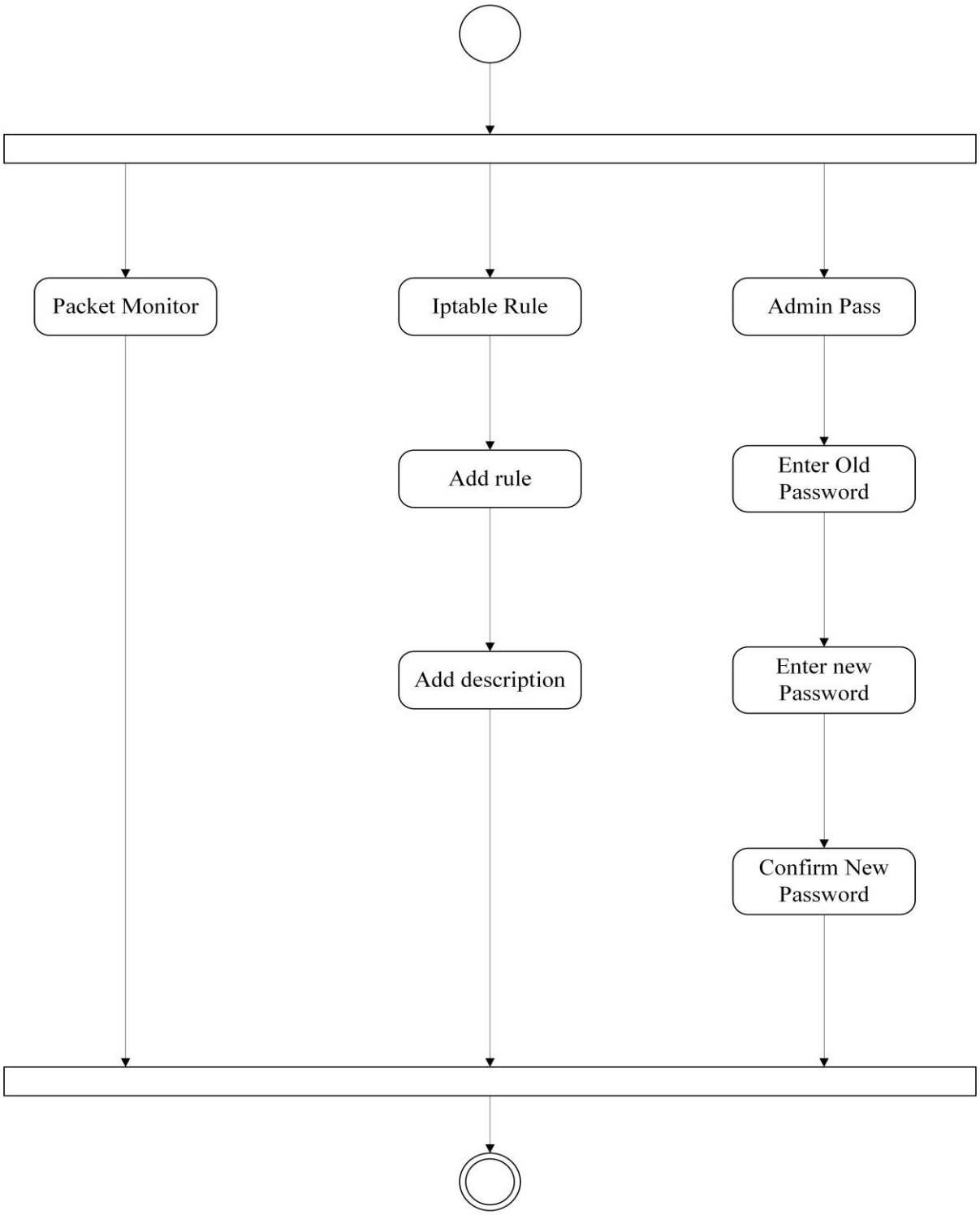


Fig 4.28 Activity Admin Advance Options 1

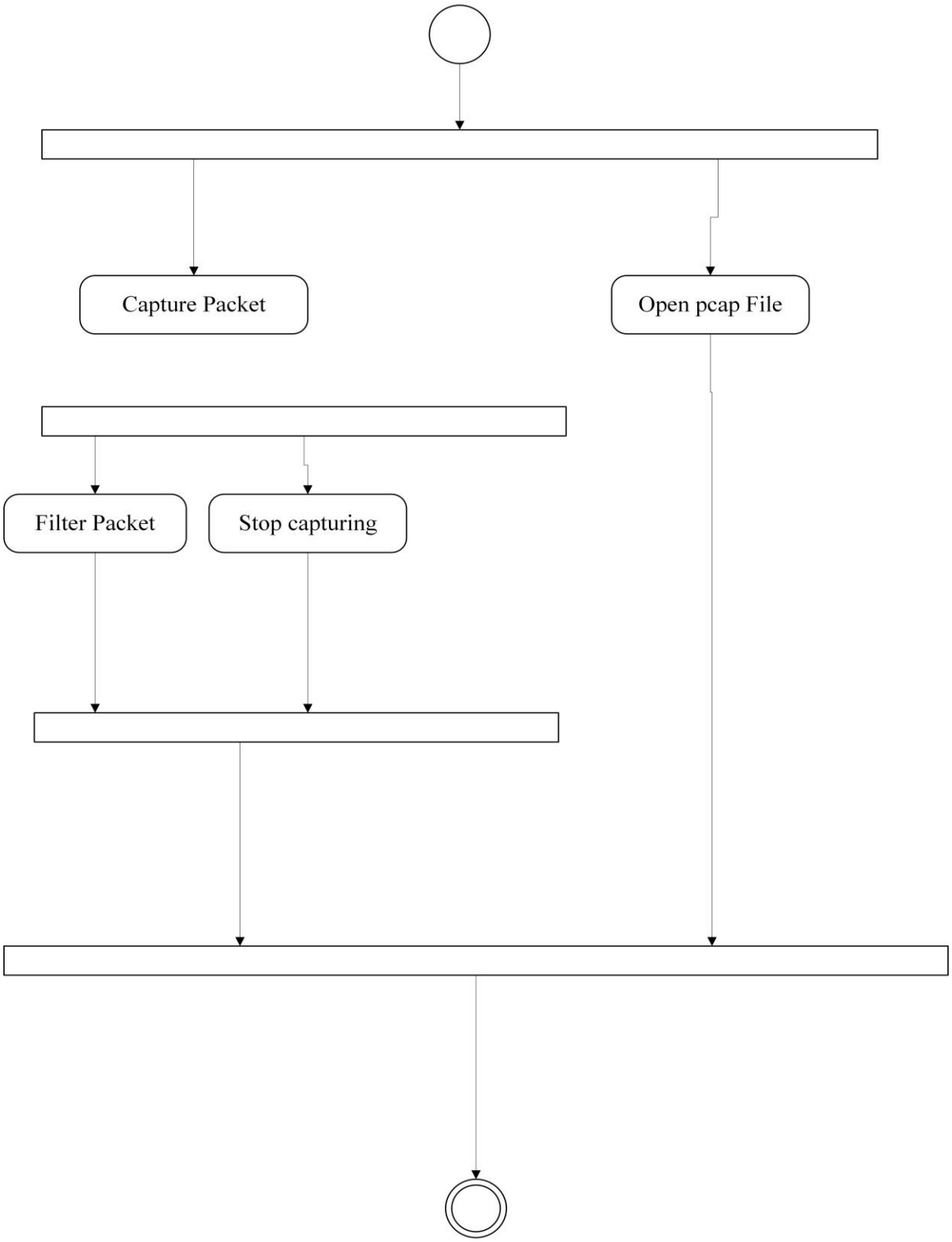


Fig 4.29 Activity Admin Advance Options 2

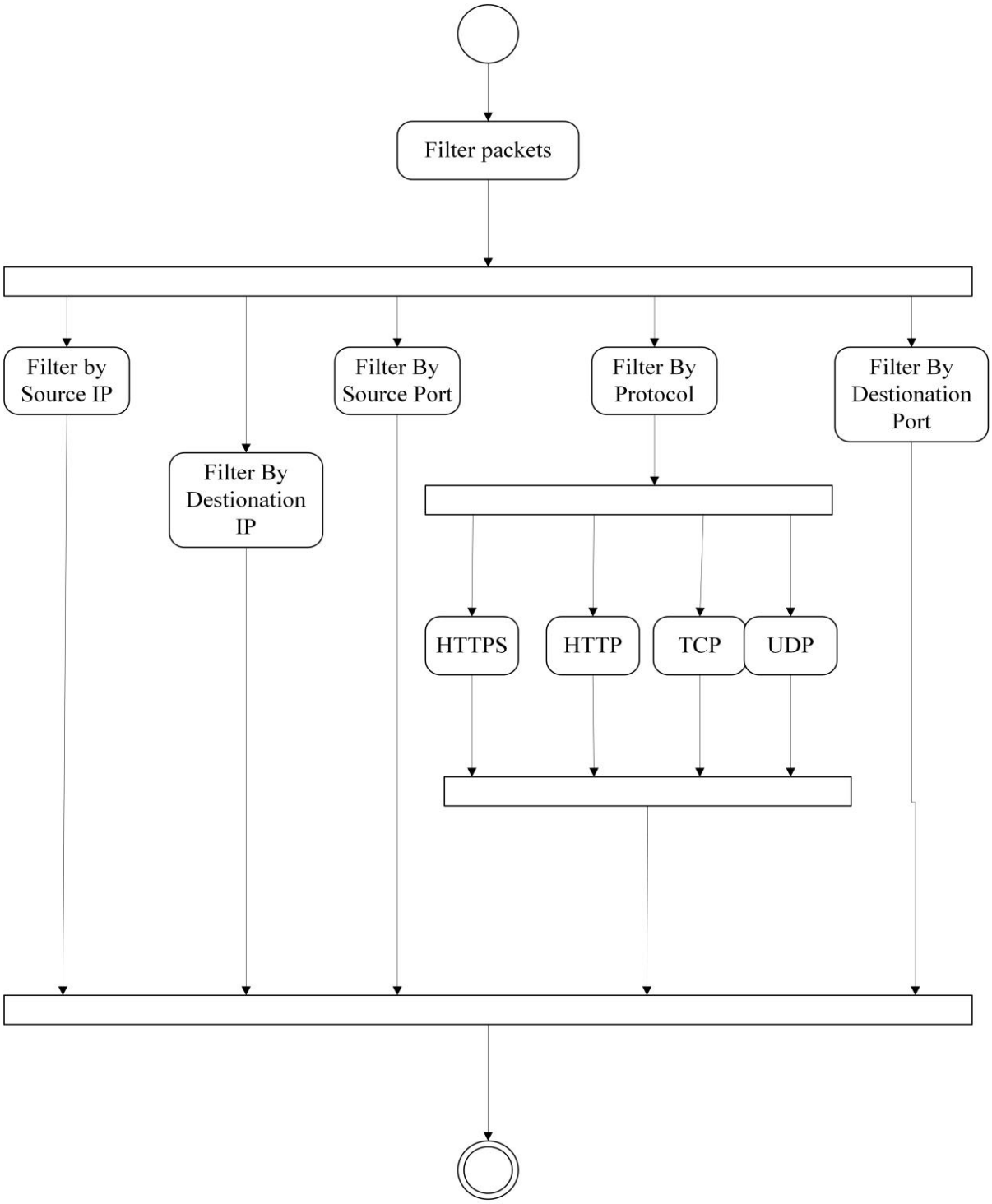


Fig 4.30 Activity Admin Advance Options 3

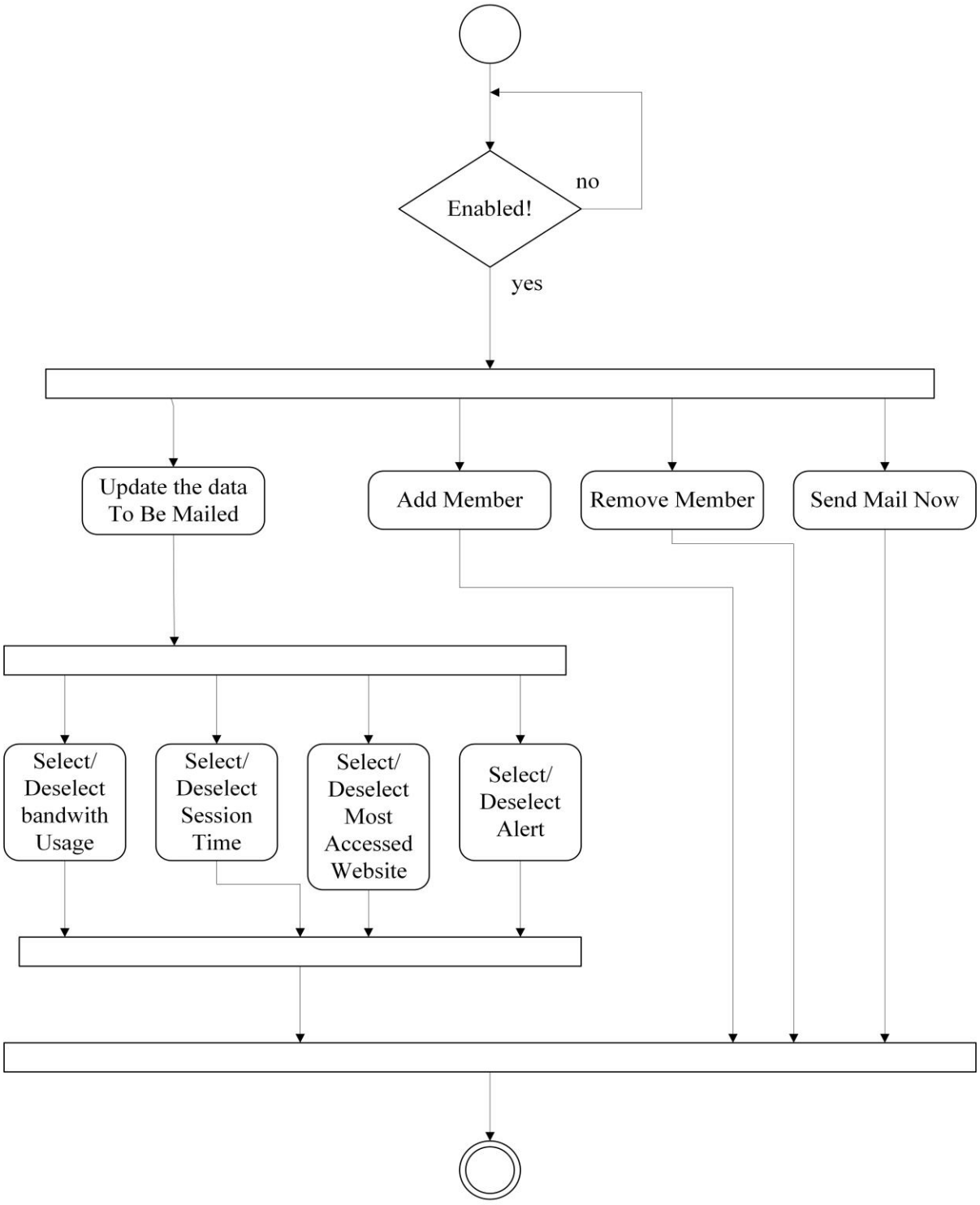


Fig 4.31 Activity Admin Mail Notification

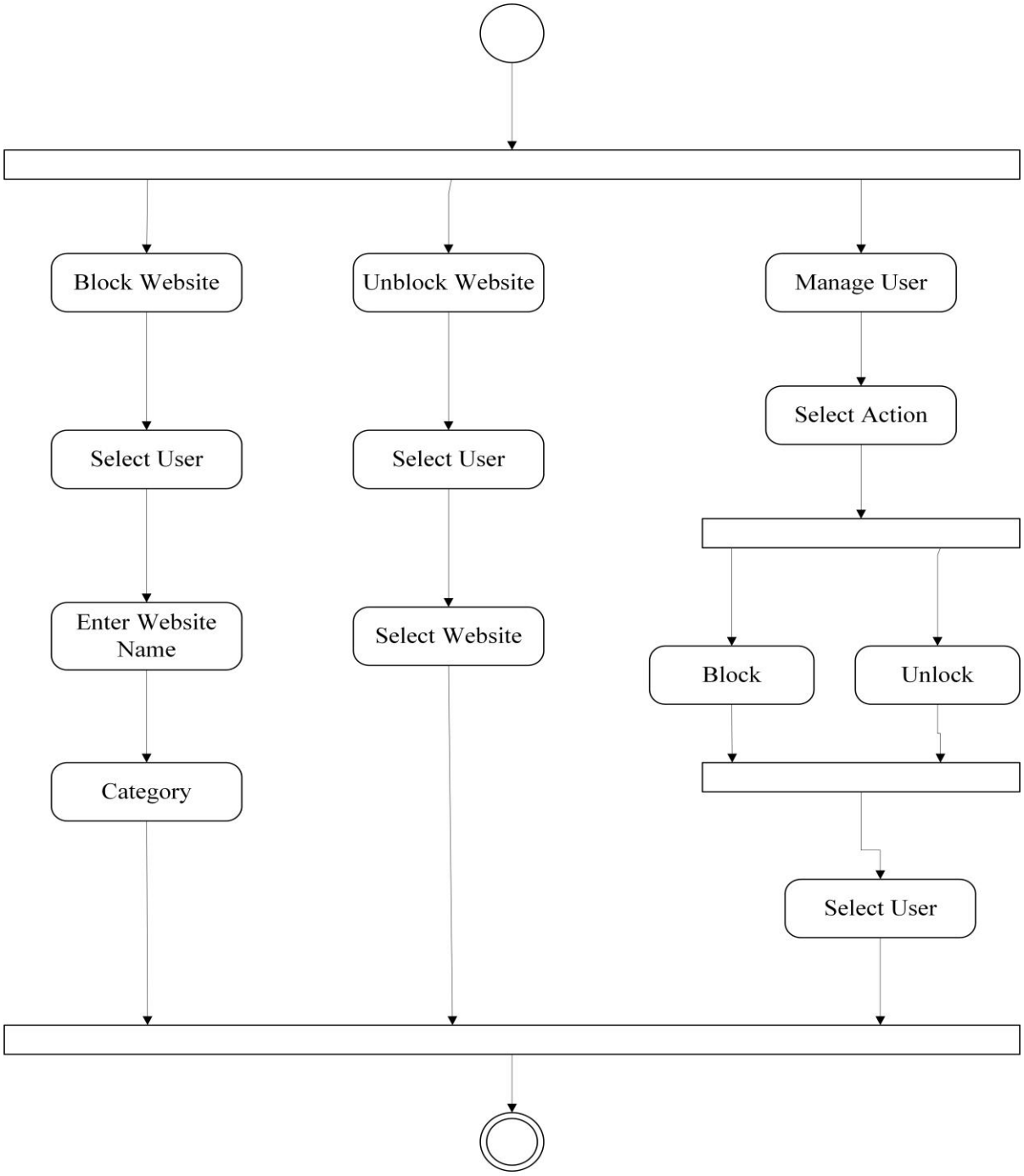


Fig 4.32 Activity Admin blocked Websites

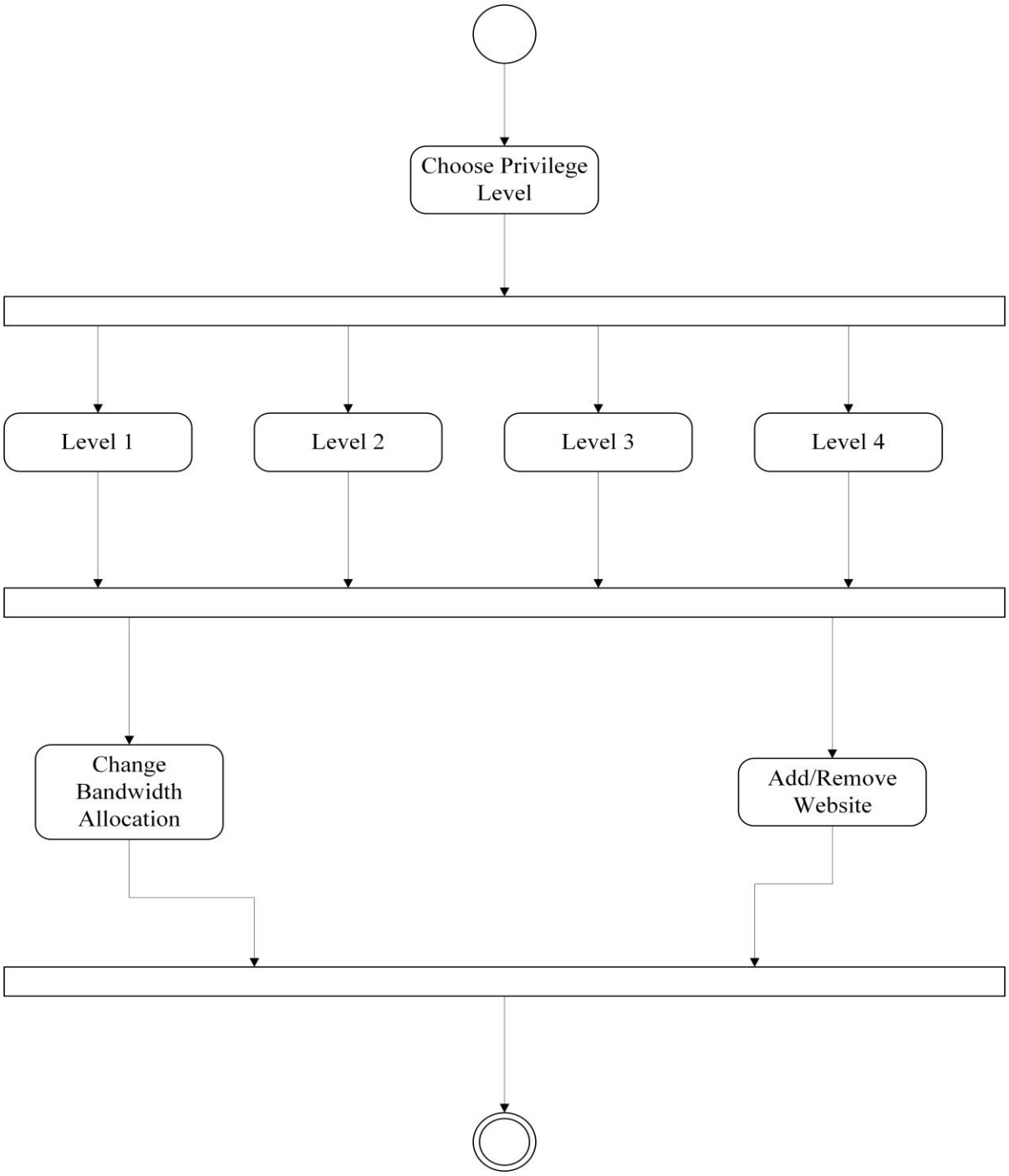


Fig 4.33 Activity Admin Privilege level

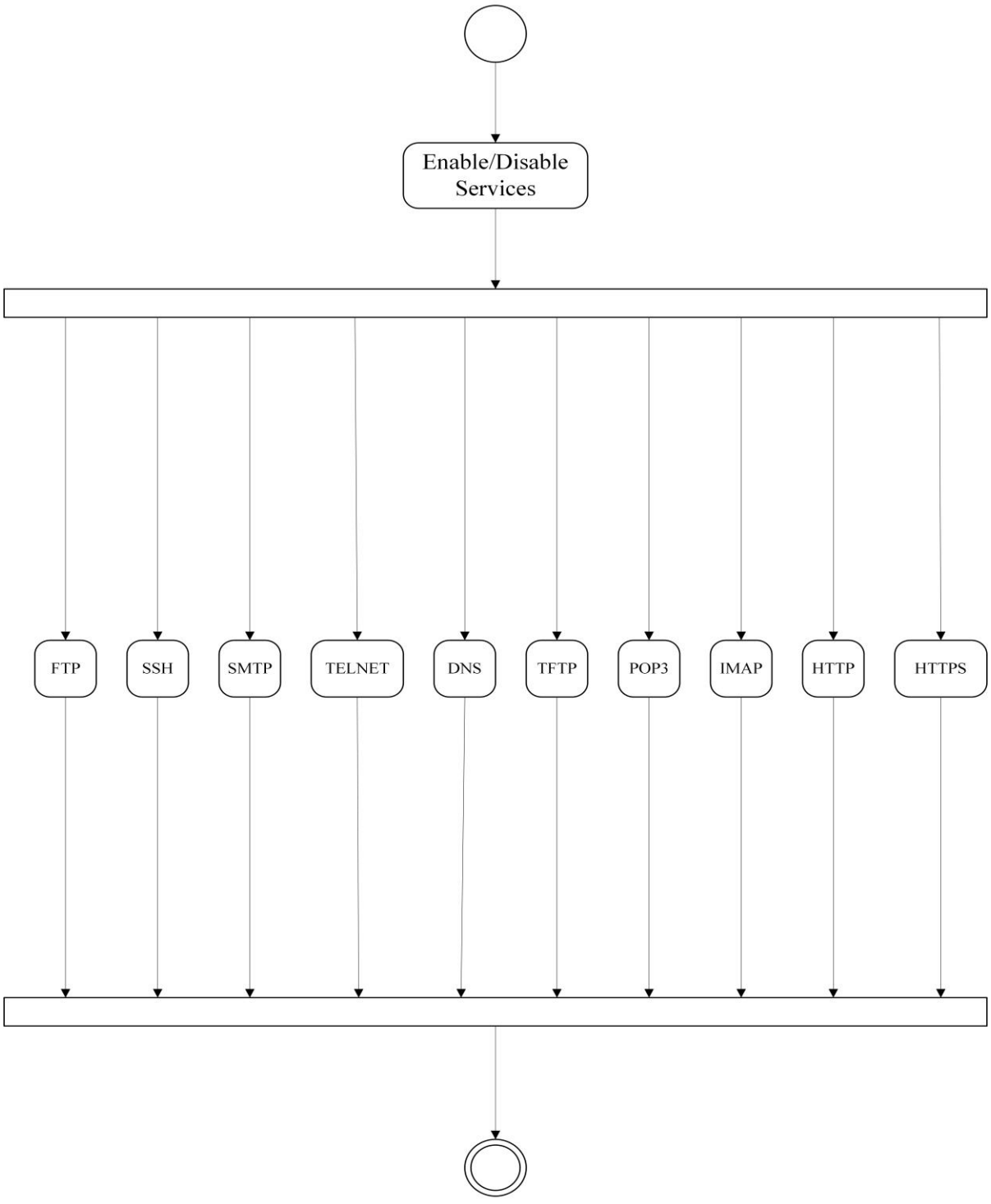


Fig 4.34 Activity Admin Services

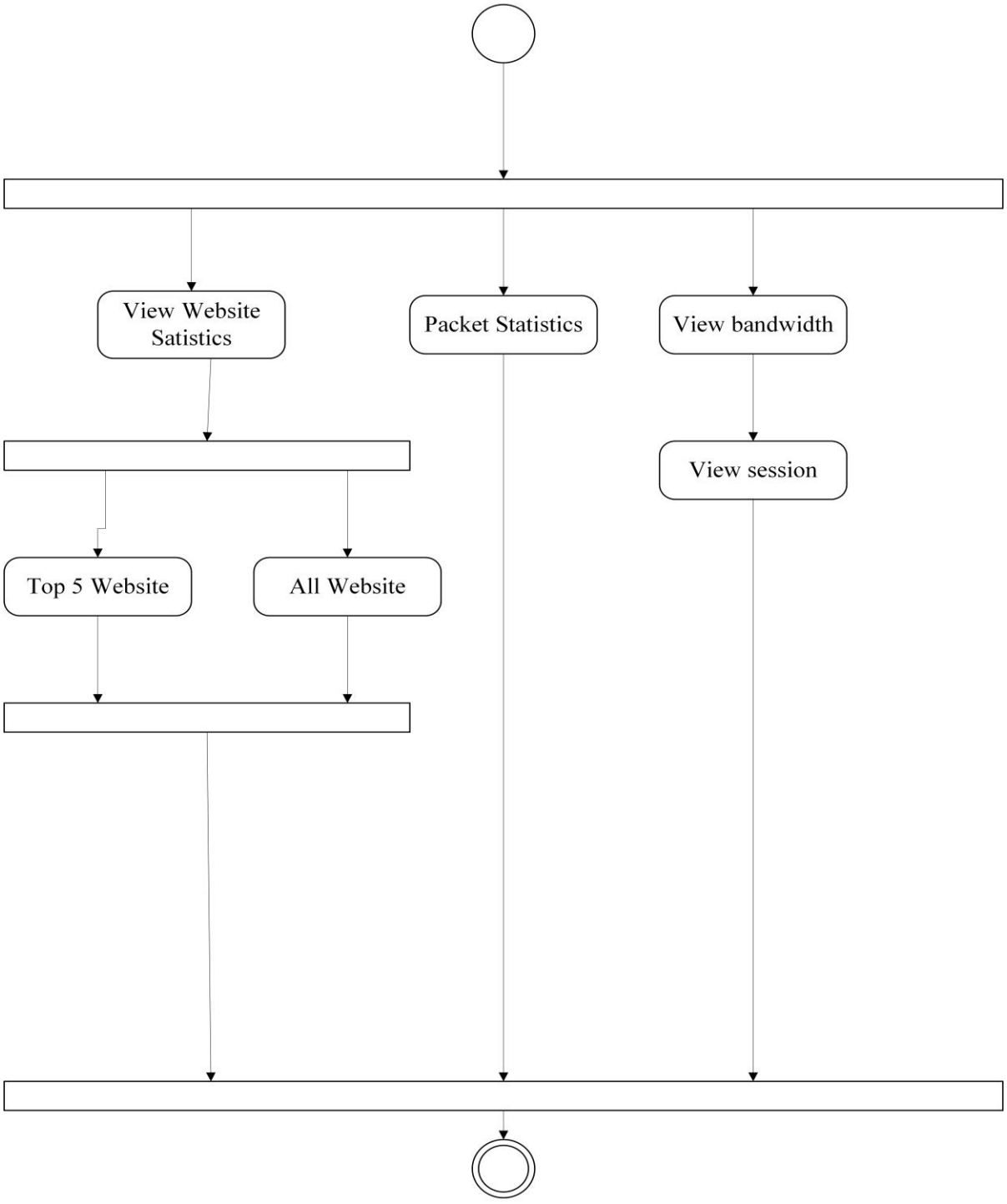


Fig 4.35 Activity Admin Statistics

4.7.4 SEQUENCE DIAGRAM :-

A sequence diagram is a graphical representation of how different object and processes interact with each other. It shows how messages are passed between different objects in the system. It depicts object interaction in a timeline sequence. It shows how different message interactions between the objects in the scenario together constitute to the whole module which in turn joined together with other modules forms the whole system.

Table 4-5 Sequence Diagram Symbols

Symbol	Meaning
	Activation
	Message
	Return Message
	Object Lifeline
	Actor Lifeline
	Self Message

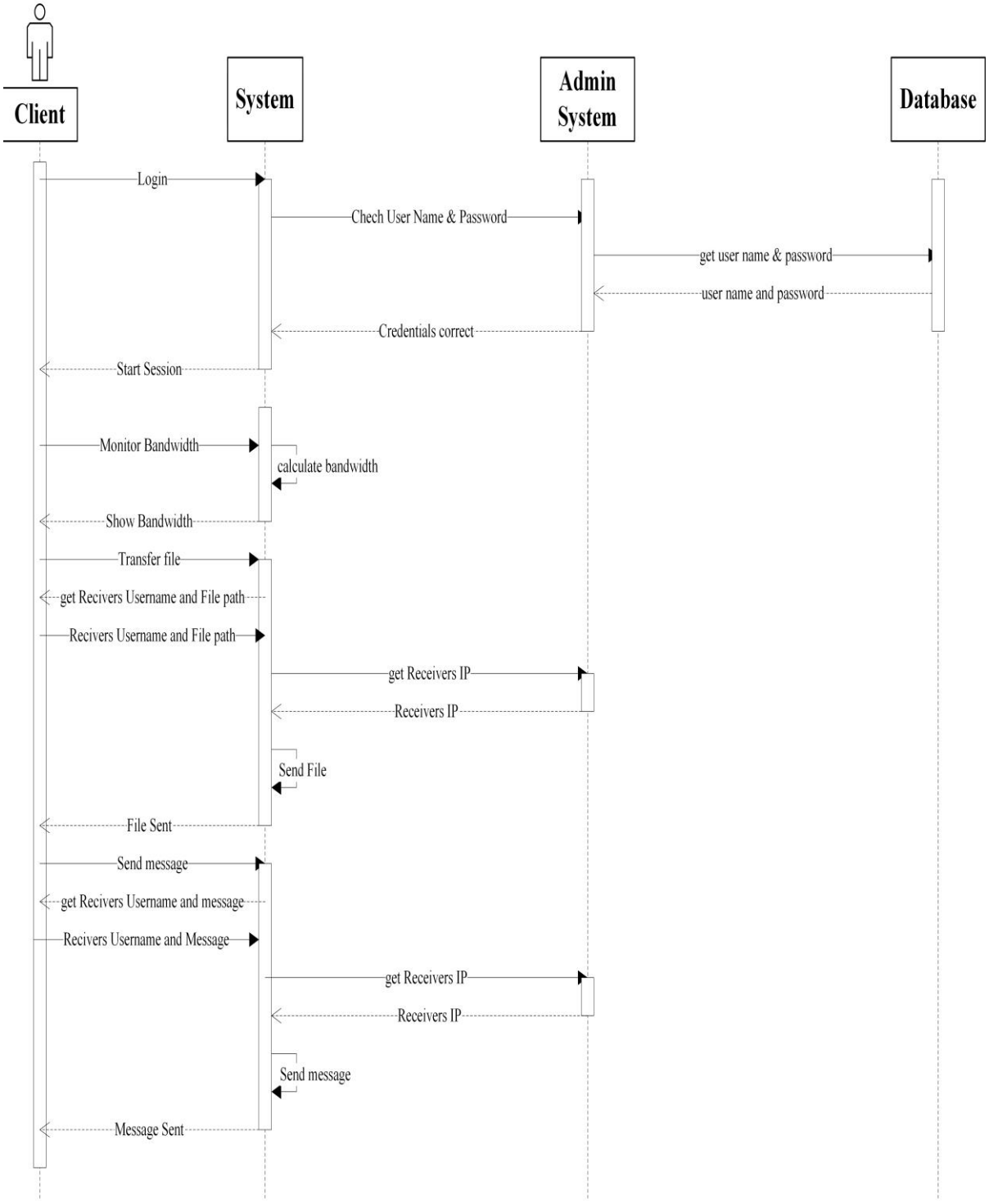


Fig 4.36 Sequence Diagram User 1

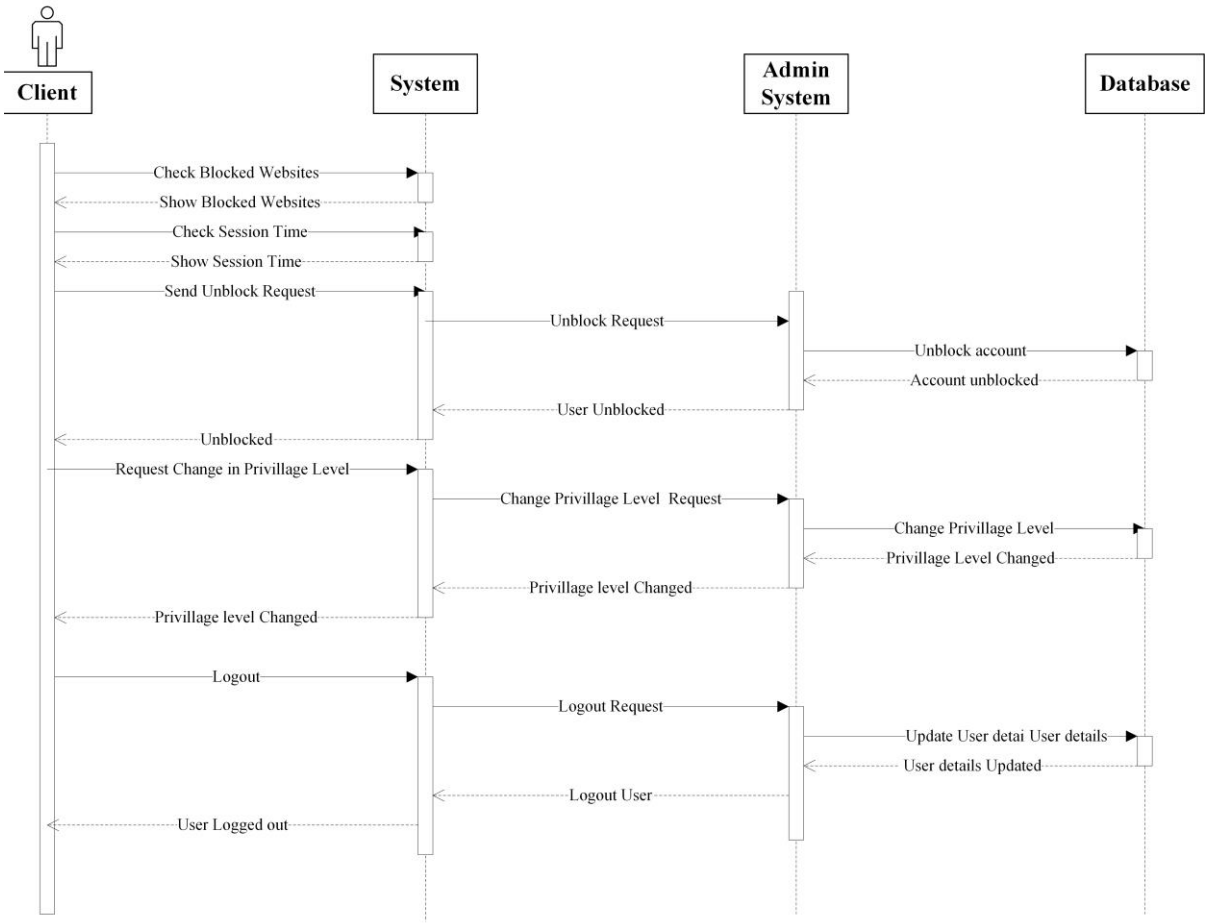


Fig 4.37 Sequence Diagram User 2

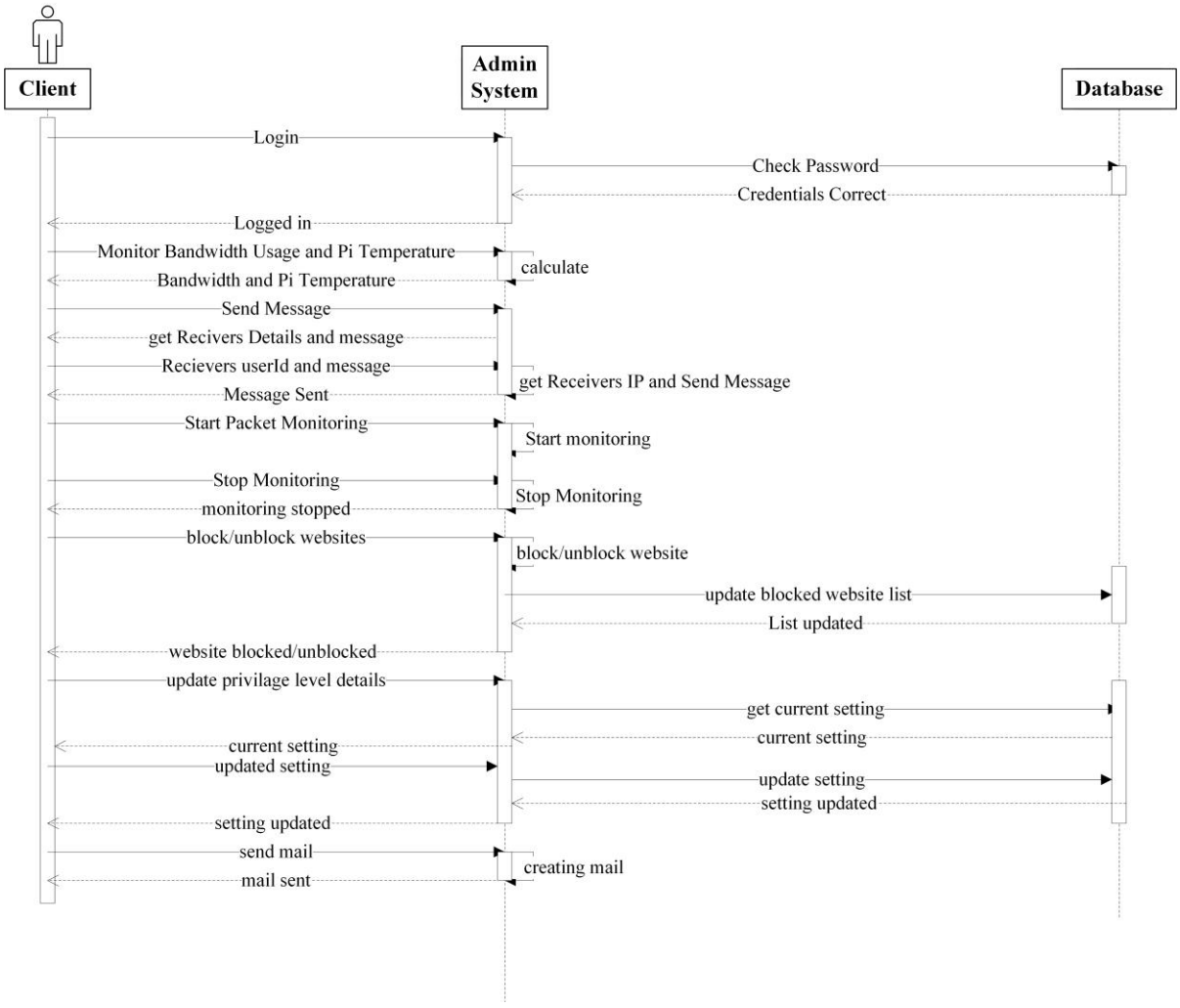


Fig 4.38 Sequence Diagram User 3

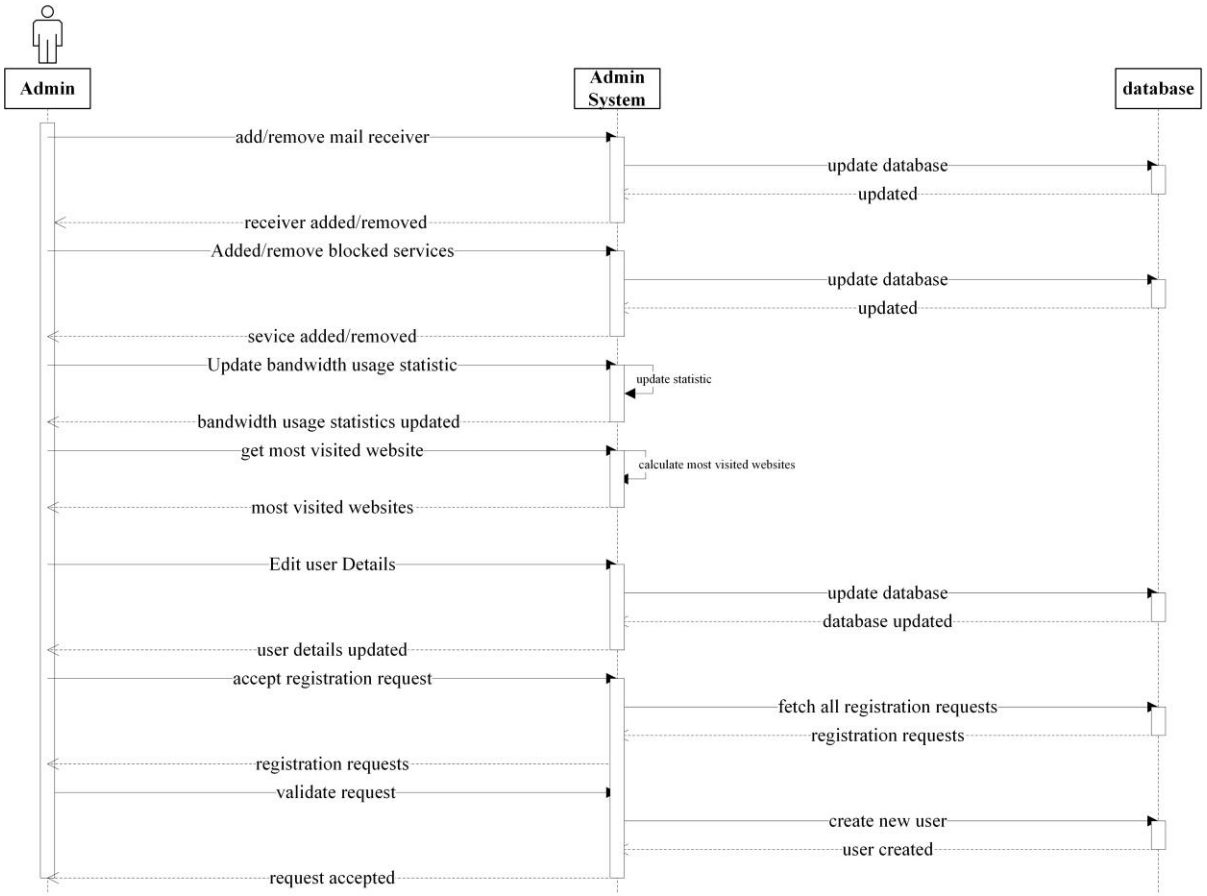


Fig 4.39 Sequence Diagram Admin 1

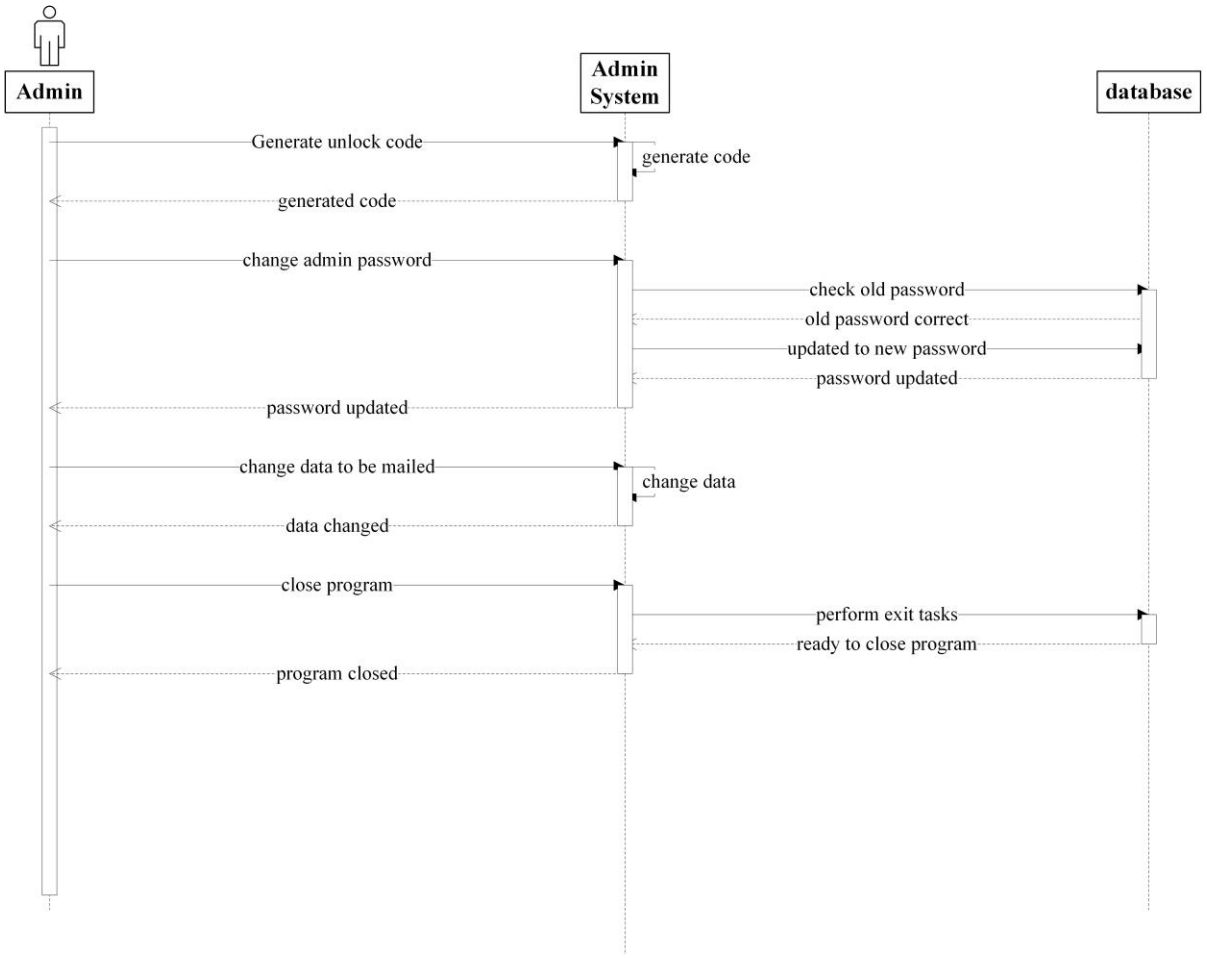


Fig 4.40 Sequence Diagram Admin 2

4.7.5 DATA DICTIONARY:

Data Dictionary is tabular representation of all the data which will be created and stored during the operation of the system. In this all the attributes which might have to be declared during developing the application are predicted and what all constraints we will be adding to those when we store them in the database are determined.

- Importance of Data Dictionary are:
- It helps to manage details of a system however small or large.
- It helps to create a common purpose for all elements of a system.
- It helps to predict how data will flow inside the system. It makes it easier to develop the database during development of the software. As most of the attributes and its constraints are already known and most of the relational connectivity problems will be dealt with before developing the database.

Table 4-6 access control

TABLE NAME: ACCESSCONTROL			
TABLE DESCRIPTION: THIS TABLE DESCRIBES INFORMATION ABOUT BLOCKED WEBSITES AND USERS .			
PRIMARY KEY: NONE		FOREIGN KEYS:UID	
TABLE STRUCTURE			
FIELD NAME	DATA TYPE	SIZE	CONSTRAINTS
UID	VARCHAR	25	PRIMARY KEY,
WEBNAME	VARCHAR	100	NONE
CATEGEORY	VARCHAR	20	NONE
STATUS	VARCHAR	8	NONE

Table 4-7 bandwidth usage

TABLE NAME : BANDUSED			
TABLE DESCRIPTION: THIS TABLE DESCRIBES BANDWIDTH USAGE AND SESSION TIME INFORMATION.			
PRIMARY KEY: NONE		FOREIGN KEYS: UID	
TABLE STRUCTURE			
FIELD NAME	DATA TYPE	SIZE	CONSTRAINTS
UID	VARCHAR	25	PRIMARY KEY
BANDUSAGE	DOUBLE	NONE	NOT NULL
TOTALSESSION	VARCHAR	9	NOT NULL

Table 4-8 block service

TABLE NAME : BLKSERVICE			
TABLE DESCRIPTION: THIS TABLE DESCRIBES BLOCKED SERVICES INFORMATION			
PRIMARY KEY: PORTNO		FOREIGN KEYS: -NONE	
TABLE STRUCTURE			
FIELD NAME	DATA TYPE	SIZE	CONSTRAINTS
PORTNO	INTEGER	5	PRIMARY KEY
SERNAME	VARCHAR	50	NOT NULL

Table 4-9 iptables

TABLE NAME :IPTABLES			
TABLE DESCRIPTION: THIS TABLE STORES CUSTOM IPTABLES RULES			
PRIMARY KEY: - NONE		FOREIGN KEYS: - NONE	
TABLE STRUCTURE			
FIELD NAME	DATA TYPE	SIZE	CONSTRAINTS
RULE	VARCHAR	100	NOT NULL
DESCRIPTION	VARCHAR	100	NOT NULL

Table 4-10 Register

TABLE NAME :- REGISTER			
TABLE DESCRIPTION: THIS TABLE DESCRIBES REGISTRATIOON INFORMATION.			
PRIMARY KEY: - UID		FOREIGN KEYS: - NONE	
TABLE STRUCTURE			
FIELD NAME	DATA TYPE	SIZE	CONSTRAINTS
NAME	VARCHAR	25	PRIMARY KEY
UID	VARCHAR	25	NOT NULL
PASSWD	VARCHAR	15	NOT NULL
EMAILID	VARCHAR	30	NOT NULL
MOBNO	BIG INTEGER	12	NOT NULL
VERIFIED	CHAR	3	NOT NULL

Table 4.11 Notification

TABLE NAME : NOTIFICATION			
TABLE DESCRIPTION: THIS TABLE DESCRIBES NOTIFICATION INFORMATION.			
PRIMARY KEY: -NAME		FOREIGN KEYS: -NONE	
TABLE STRUCTURE			
FIELD NAME	DATA TYPE	SIZE	CONSTRAINTS
DATA_MAIL	VARCHAR	50	NOT NULL
NAME	VARCHAR	25	PRIMARY KEY
EMAIL	VARCHAR	30	NOT NULL

Table 4.12 privilege level

TABLE NAME:-PLEVEL			
TABLE DESCRIPTION:-THIS TABLE DESCRIBES PRIVILEGE LEVEL INFORMATION			
PRIMARY KEY: - NONE		FOREIGN KEYS: - NONE	
TABLE STRUCTURE			
FIELD NAME	DATA TYPE	SIZE	CONSTRAINTS
LEVEL	INTEGER	1	NOT NULL
BANDALLOC	DOUBLE	NONE	NONE
WEBLIST	VARCHAR	100	NONE

Table 4.13 user login

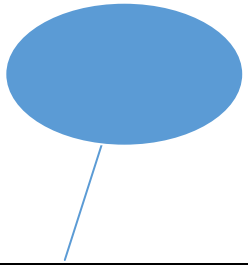
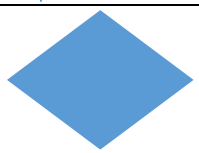
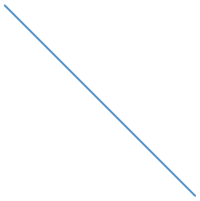
TABLE NAME:-ULOGIN			
TABLE DESCRIPTION:-THIS TABLE DESCRIBES LOGIN INFORMATION			
PRIMARY KEY: - UID		FOREIGN KEYS: - NONE	
TABLE STRUCTURE			
FIELD NAME	DATA TYPE	SIZE	CONSTRAINTS
UID	VARCHAR	25	PRIMARY KEY
PASS	VARCHAR	15	NOT NULL
PRIVILLAGE	IINTEGER	2	NOT NULL

4.8 FUNCTIONAL & BEHAVIOURAL MODELING:

4.8.1 DATA FLOW DIAGRAM:

A Data flow diagram (DFD) represents the flow of data inside the system between different objects and entities. A DFD is usually used for graphically representing data processing and also to create an overview of the system. This overview will then be used to create a more elaborated view in the future. A DFD elaborately describes the different input and output data of all the functions of a system, i.e., where this data comes from and what it turns to be after processing and where that processed data will be stored.

Table 4.14 DFD Symbols & Meaning

Symbol	Meaning
	Entity
	Attribute
	Relationship
	Relationship Connector

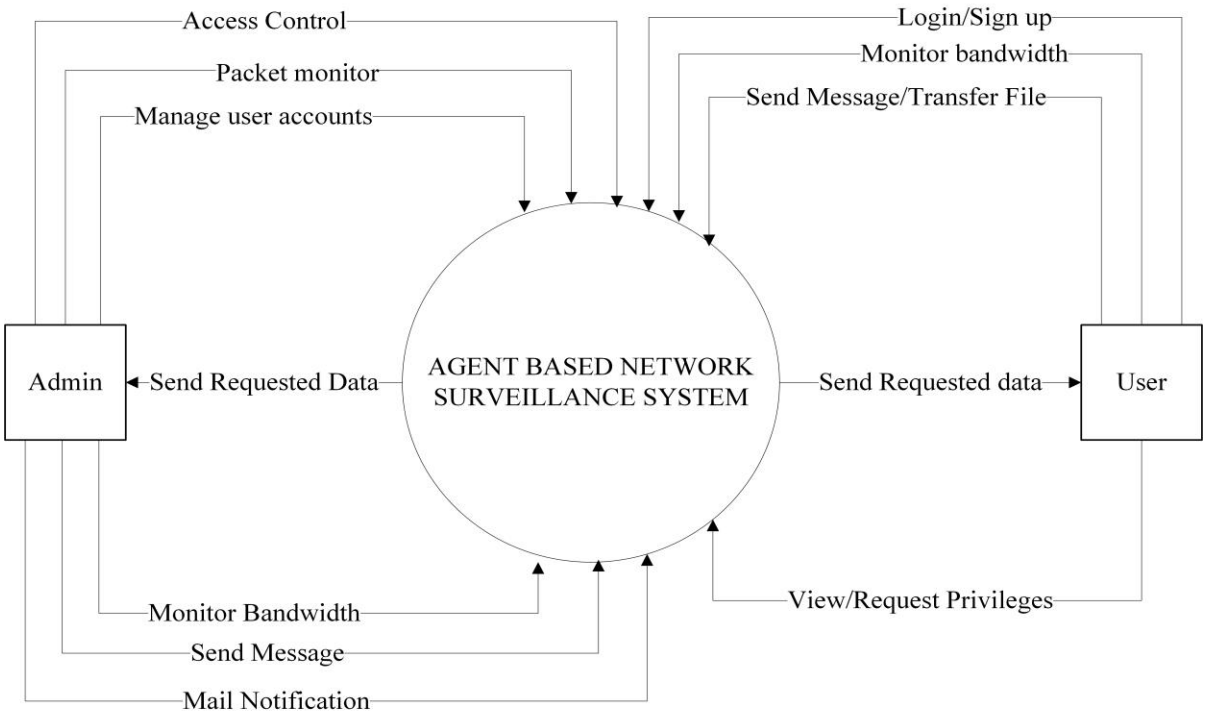


Fig 4.41 DFD Level-0

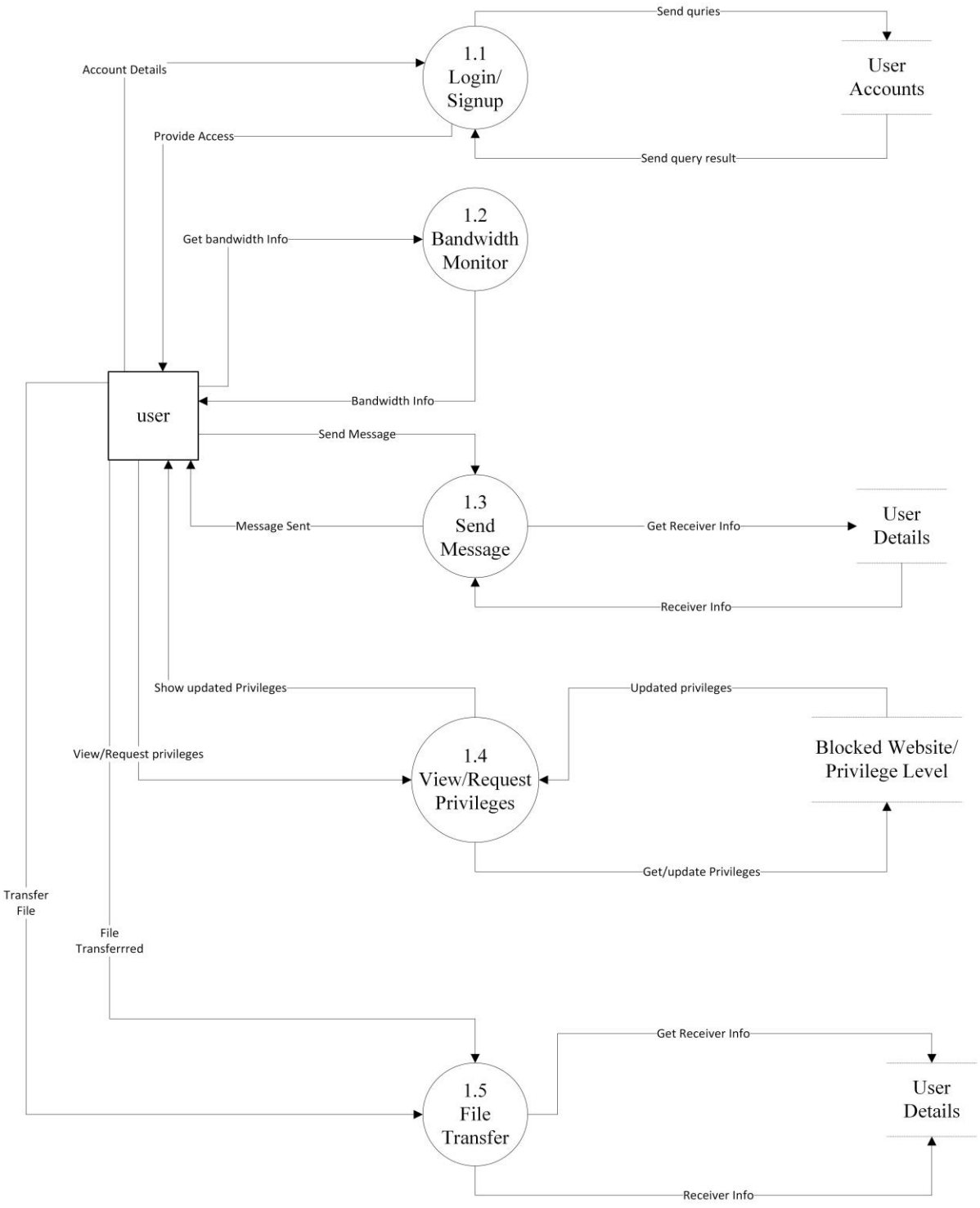


Fig 4.42 DFD Level-1 User

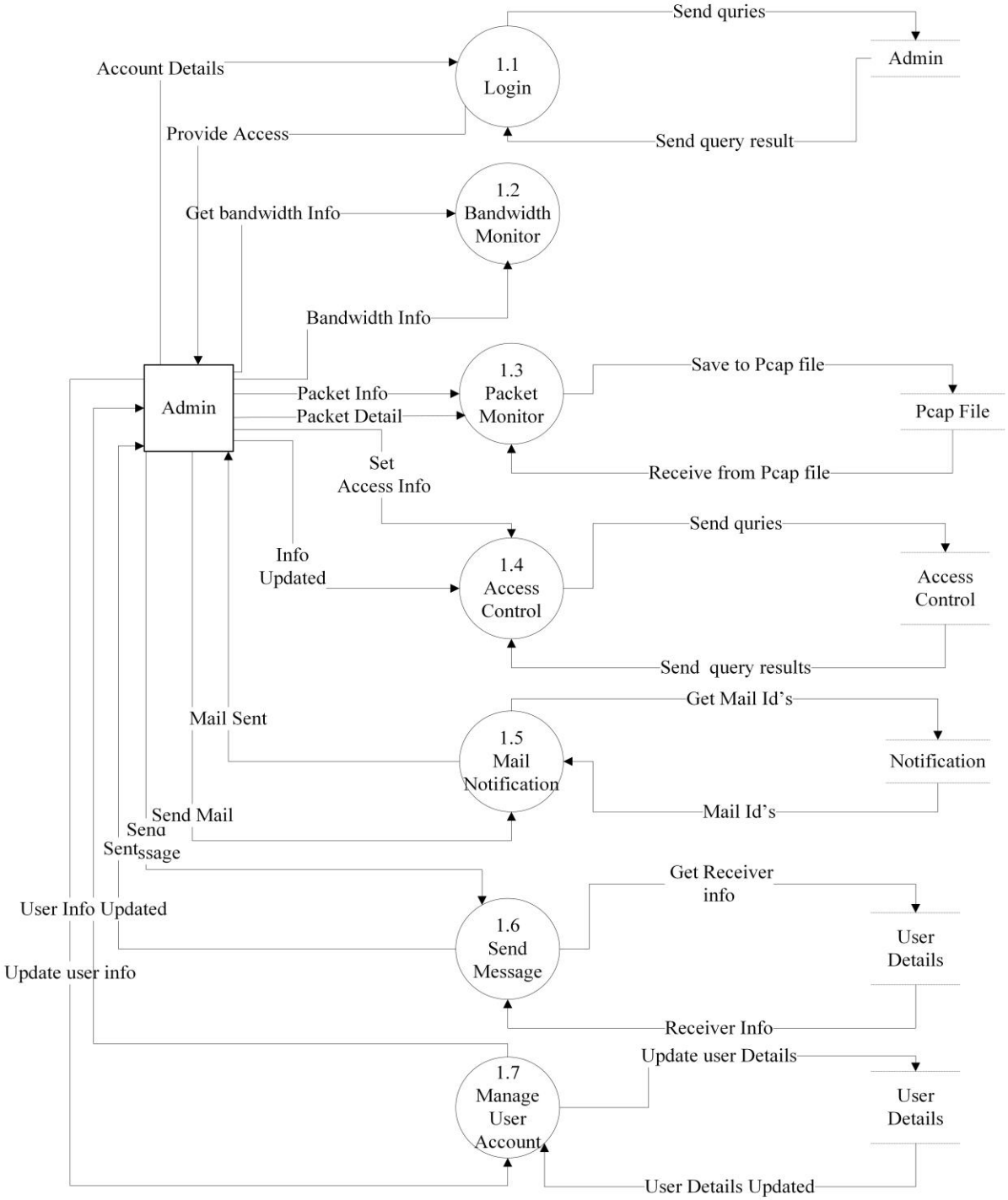


Fig 4.43 DFD Level-1 Admin

CHAPTER

5

SYSTEM DESIGN

CHAPTER 5

SYSTEM DESIGN

5.1 DATABASE/ DATA STRUCTURE DESIGN

5.1.1 MAPPING OBJECT/CLASSES TO TABLES

abnss.ulogin

Info | Columns | Indexes | Triggers | Foreign keys | Partitions | Grants

Column	Type	Default Value	Nullable	Character Set	Collation	Privileges
uid	varchar(25)		NO	latin1	latin1_swedish_ci	select,insert
pass	varchar(15)		NO	latin1	latin1_swedish_ci	select,insert
privillage	int(2)		NO			select,insert

Figure 5.1 ulogin

abnss.register

Info | Columns | Indexes | Triggers | Foreign keys | Partitions | Grants

Column	Type	Default Value	Nullable	Character Set	Collation	Privileges
name	varchar(25)		NO	latin1	latin1_swedish_ci	select,insert
uid	varchar(25)		NO	latin1	latin1_swedish_ci	select,insert
passwd	varchar(15)		NO	latin1	latin1_swedish_ci	select,insert
emailid	varchar(30)		NO	latin1	latin1_swedish_ci	select,insert
mobno	bigint(12)		NO			select,insert
verified	char(3)		NO	latin1	latin1_swedish_ci	select,insert

Figure 5.2 register

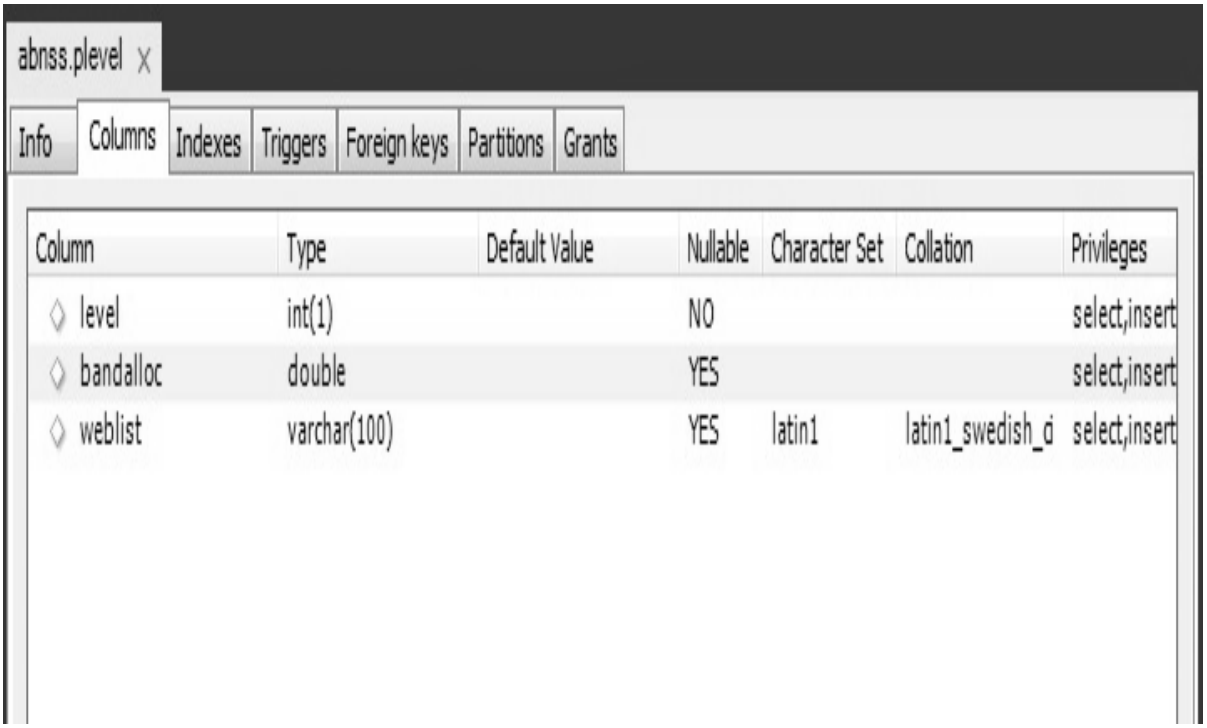


Figure 5.3 privilege level

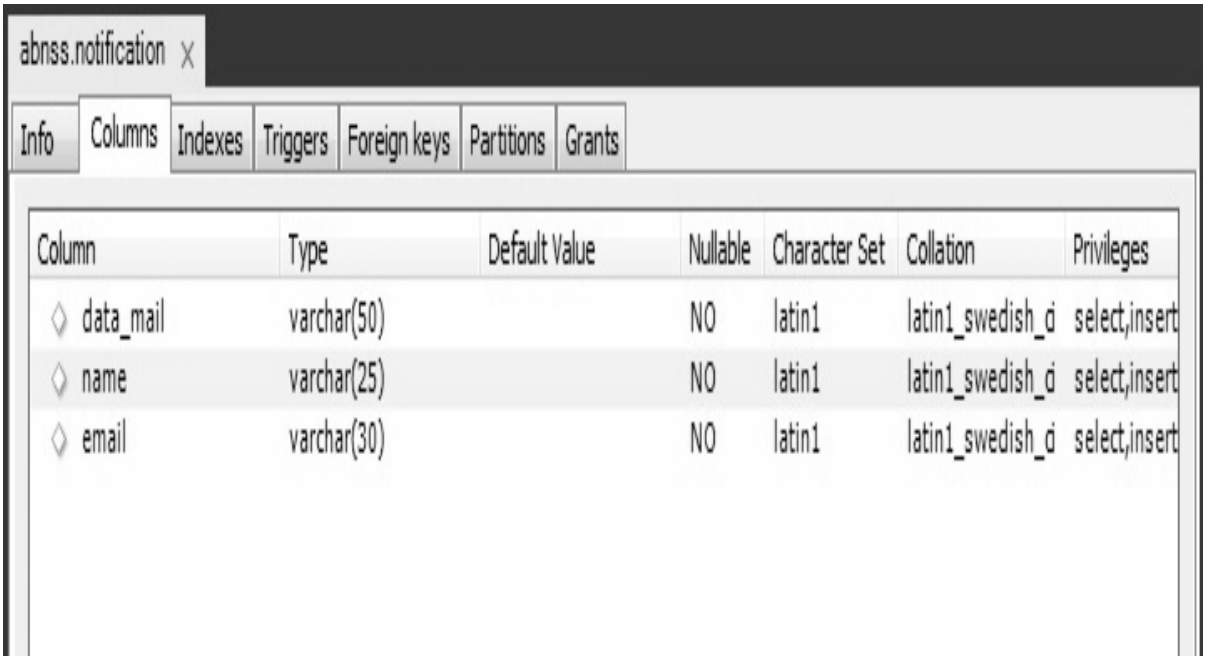


Figure 5.4 notification

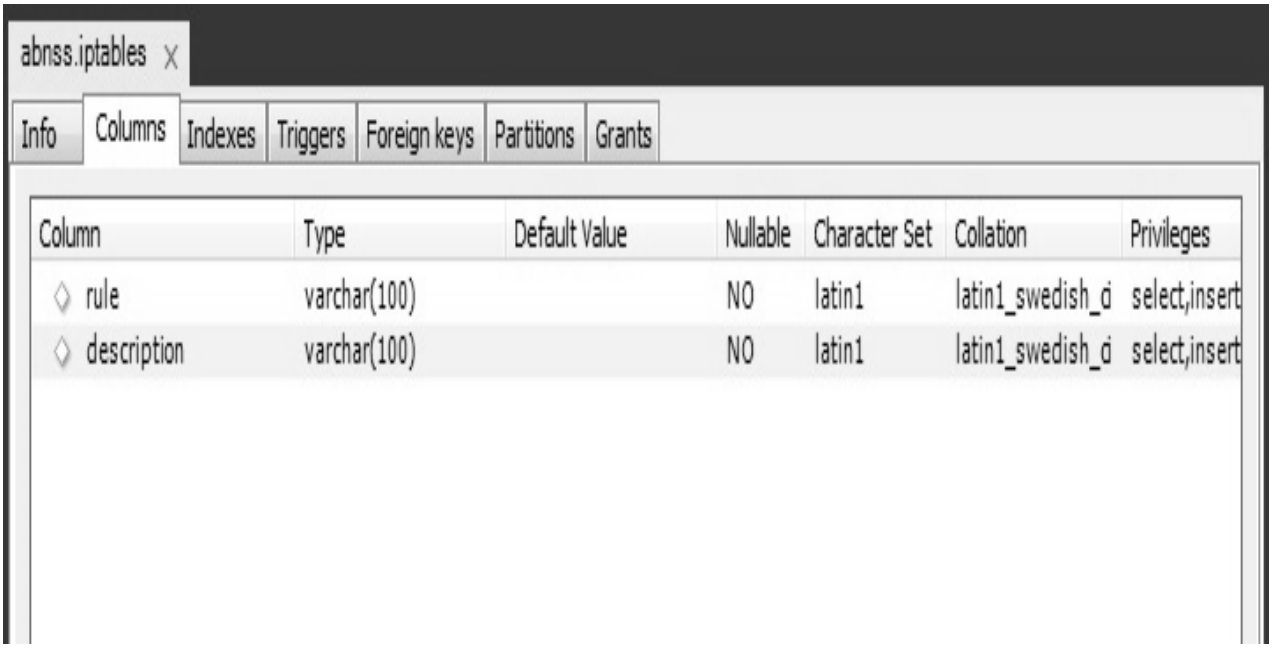


Figure 5.5 iptables

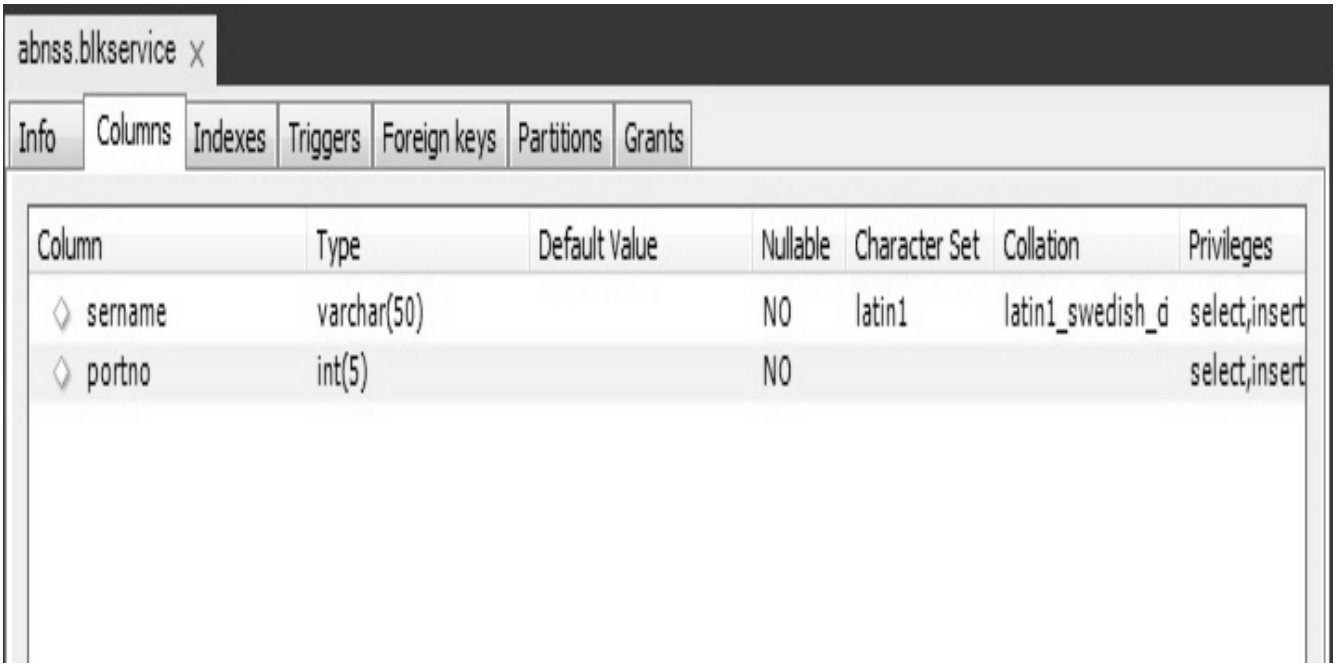


Figure 5.6 block service

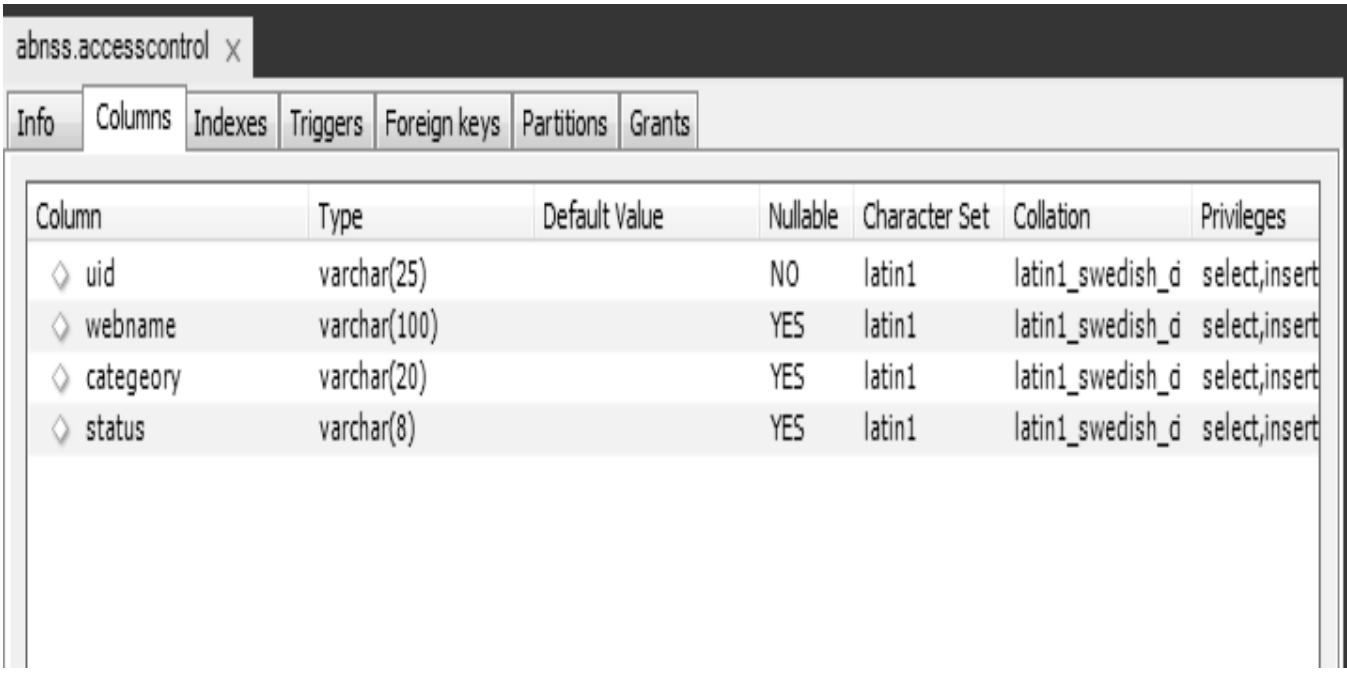


Figure 5.7 access control

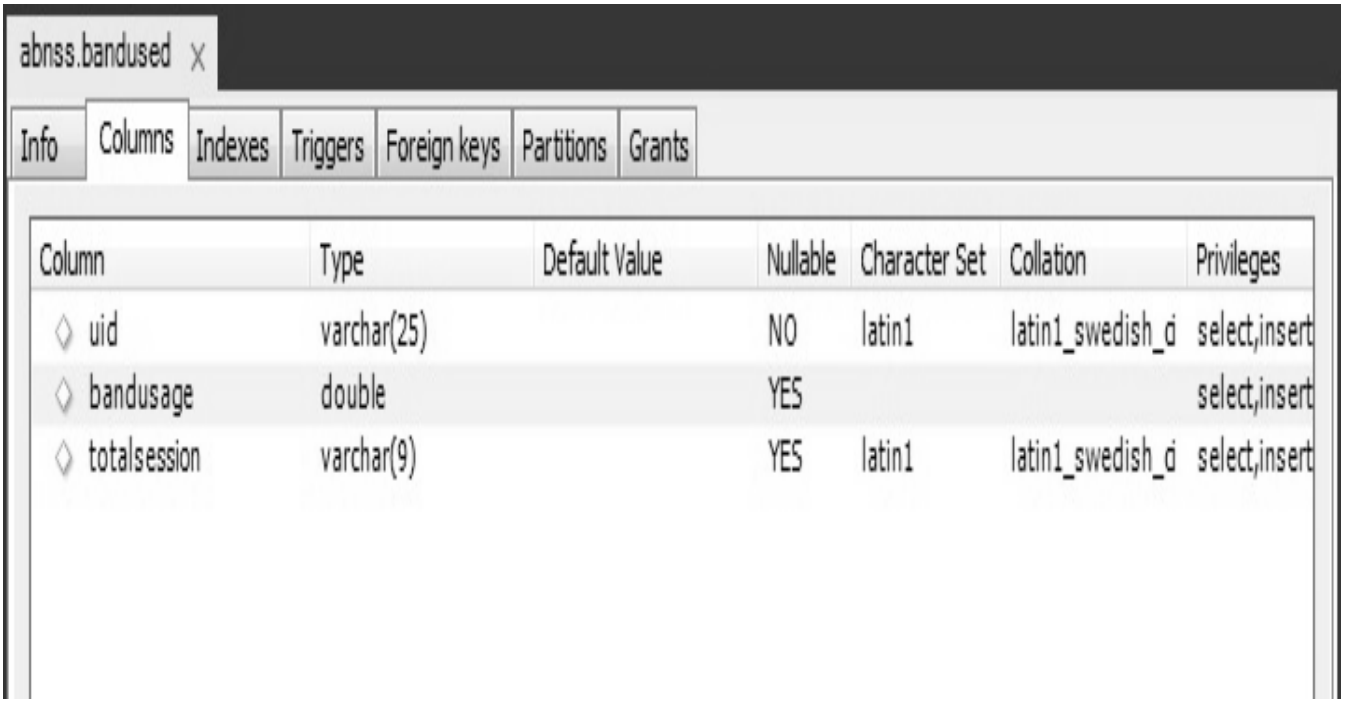


Figure 5.8 bandwith usage

5.1.2 LOGICAL REPRESENTATION OF DATA

- **Ulogin: -**This table includes information of user id, password and privilege level that is used for login.

- **Register: -** This table includes user account information including his full name, email, password, mobile number, user id and whether the account is verified or not.

- **Blkservice: --** This table includes information about all the blocked services and port numbers

- **Plevel: -** This table includes information about blocked website name, allowed bandwidth usage for the particular privilege level.

- **Accesscontrol: -** This table includes information about blocked website name, blocked users list

- **Bandused: -** This table includes information about the total session time and bandwidth used by particular user.

- **Notification: -** This table includes information about the email address and name of the members to whom the periodic mail is sent.

- **Iptables: -** This table includes information about custom iptables rule and description.

5.2 SYSTEM PROCEDURAL DESIGN

5.2.1 DESIGN PSEUDO CODE OR ALGORITHM FOR METHOD OR OPERATION

User Side:

Step 1: Tap icon to open the application.

Step 2: Login by providing valid credentials.

Step 3: After login, Dashboard is displayed.

Step 4: Application Starts Monitoring the Bandwidth Usage Of User and get the privilege level from Server.

Step 5: Dashboard contains navigation to main operations.

Step 6: LogOut & Close the application.

Admin Side:

Step 1: Power up the Raspberry Pi

Step 2: Login by providing valid Password.

Step 3: After login Home is displayed.

Step 4: Application Starts Monitoring the Bandwidth Usage and Raspberry Pi's Temperature.

Step 5: Start Writing all packets to the Pcap File.

Step 6: Admin can able to perform operation like insert, update and delete Blocked website to the database.

Step 7: Close the application.

5.3 INPUT/OUTPUT & INTERFACE DESIGN

5.3.1 SAMPLES OF FORMS, REPORTS & INTERFACE

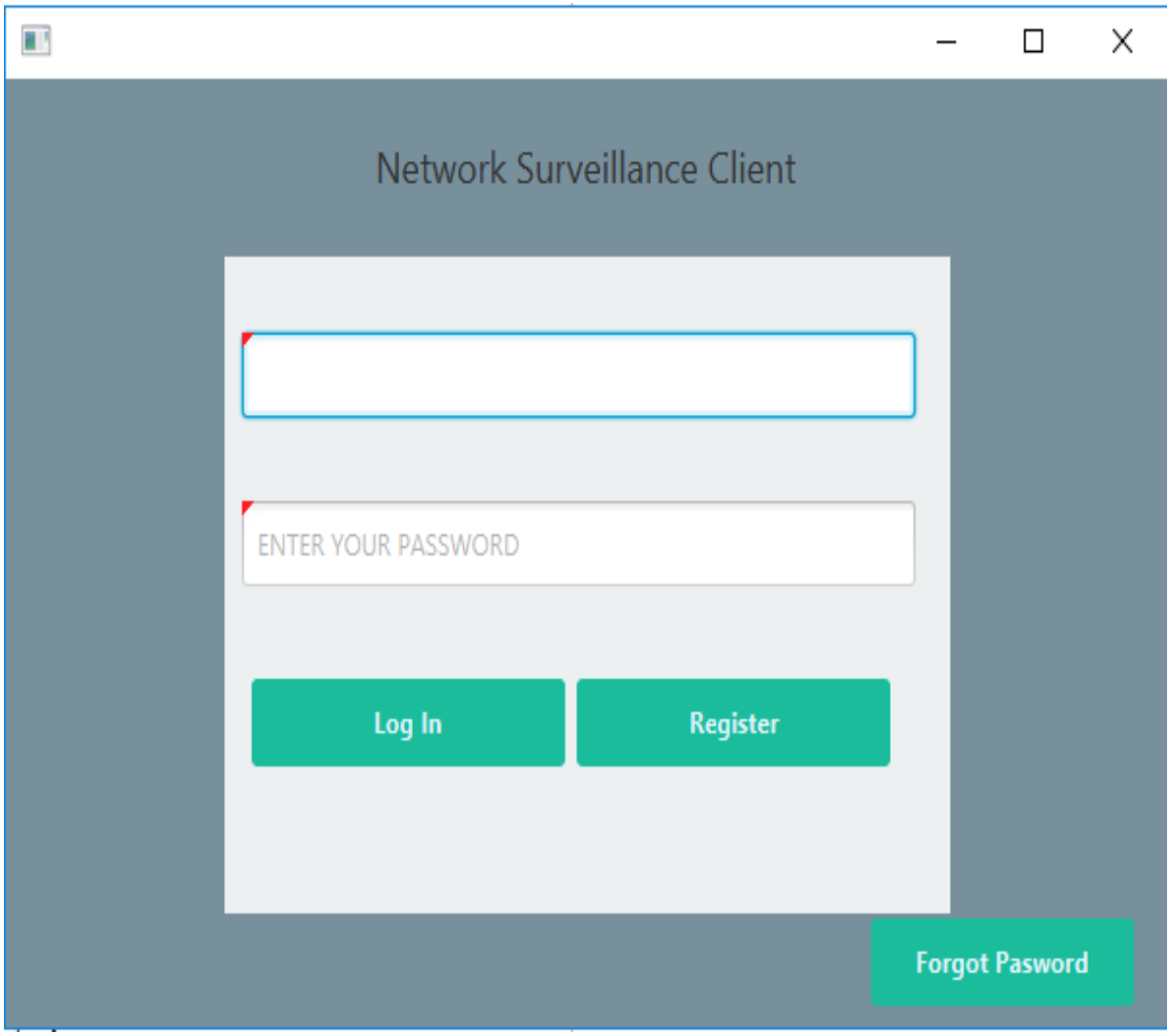


Figure 5.9 Sample User Login

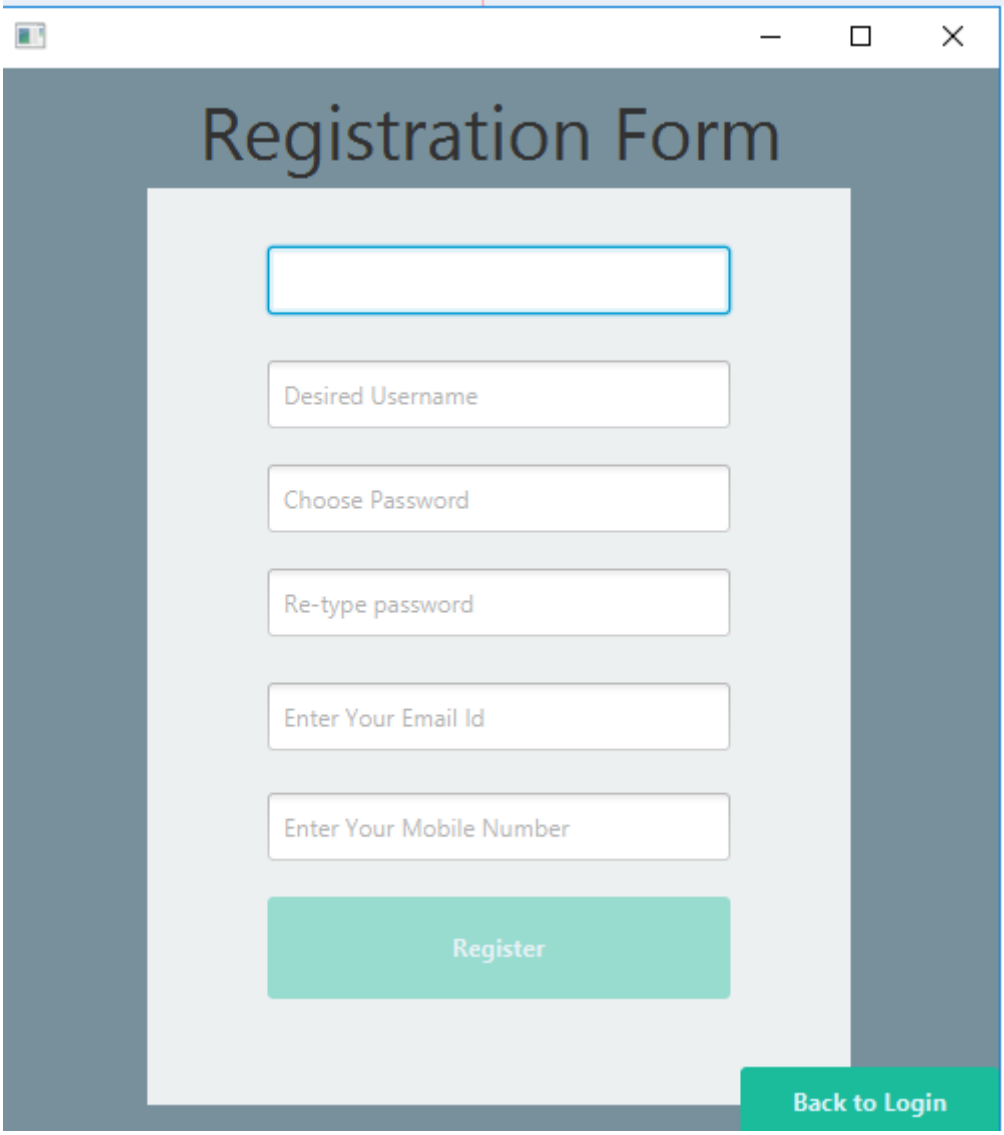


Figure 5.10 Sample User Register

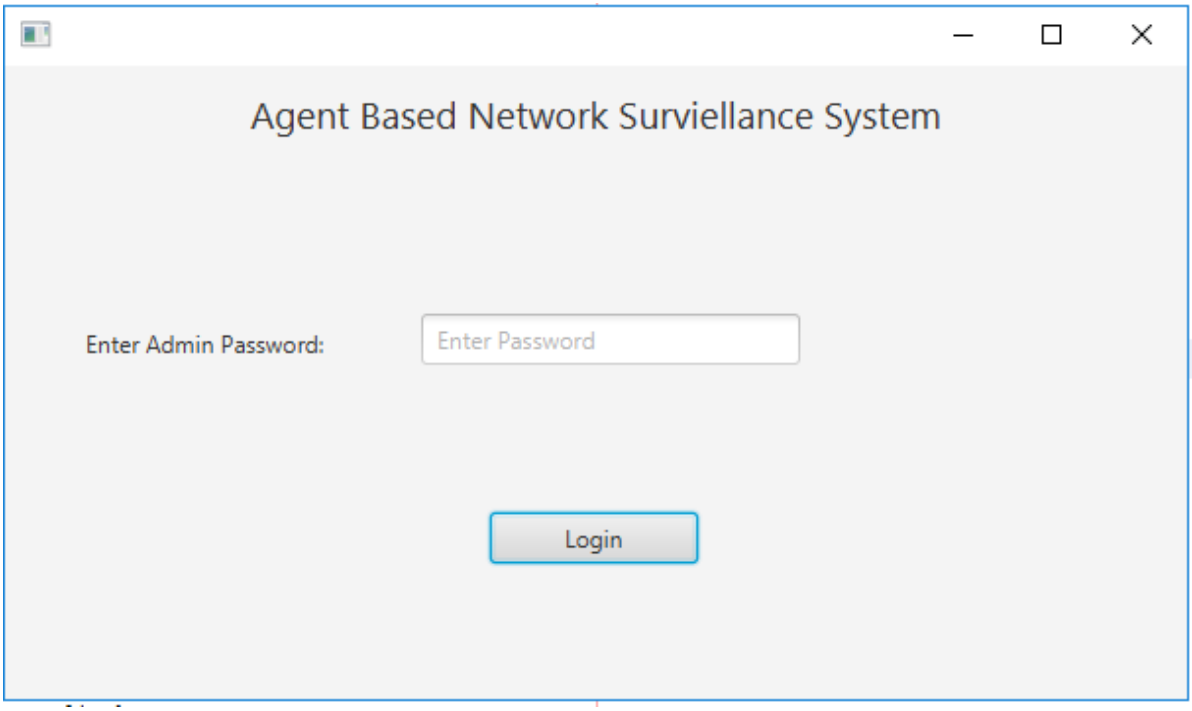


Figure 5.11 Sample Admin Login

CHAPTER

6

IMPLEMENTATION PLANNING AND DETAILS

CHAPTER 6

IMPLEMENTATION PLANNING AND DETAILS

6.1 IMPLEMENTATION ENVIRONMENT

- Agent Based Network Surveillance system is java based application which require a Debian based OS for implementation.so appropriate IDE is required for developing the application and a Debian based system for testing, running debugging it.

➢ **Hardware:**

- **User Side:**
  - Any Computer having Support for Below Configuration: -
    - OS: Any OS having Support for JVM or JRE 1.8.65
    - Processor: 1GHZ and Above
    - RAM: 256 MB and above

- **Admin Side:**
  - Any Computer or Raspberry pi with Below Configuration: -
    - OS: Any Based Debian OS With Java 1.8.65
    - Processor: 1GHZ and Above
    - RAM: 512 MB and above
    - Graphical Processor: 60 Frame Per Second Support

- **Developer Side:**
  - Any Computer with Below Configuration: -
    - OS: Any OS With Java 1.8.65
    - Processor: 2 GHZ and Above
    - RAM: 4 GB and above

- Graphical Processor: 60 Frame Per Second Support
- IDE: Netbeans and Scene Builder

➢ Network Setup Used

In order for the program work properly it is mandatory for the network to be setup properly. The Raspberry Pi on which the server program is implemented has to be set as the gateway for all the PCs in the network so that all the packets pass through it. To enable that Feature in Raspberry pi it is mandatory to Follow a bunch of procedure and that are: -

- IPV4 Forwarding Must be enabled.
- In all the Client pc the gateway must be set to the ip address of Raspberry Pi.
- Network Address Translation is must be enabled.

- **Physical Setup of the network**

The below image shows the physical setup of the network. This is how all the components are connected with each other.

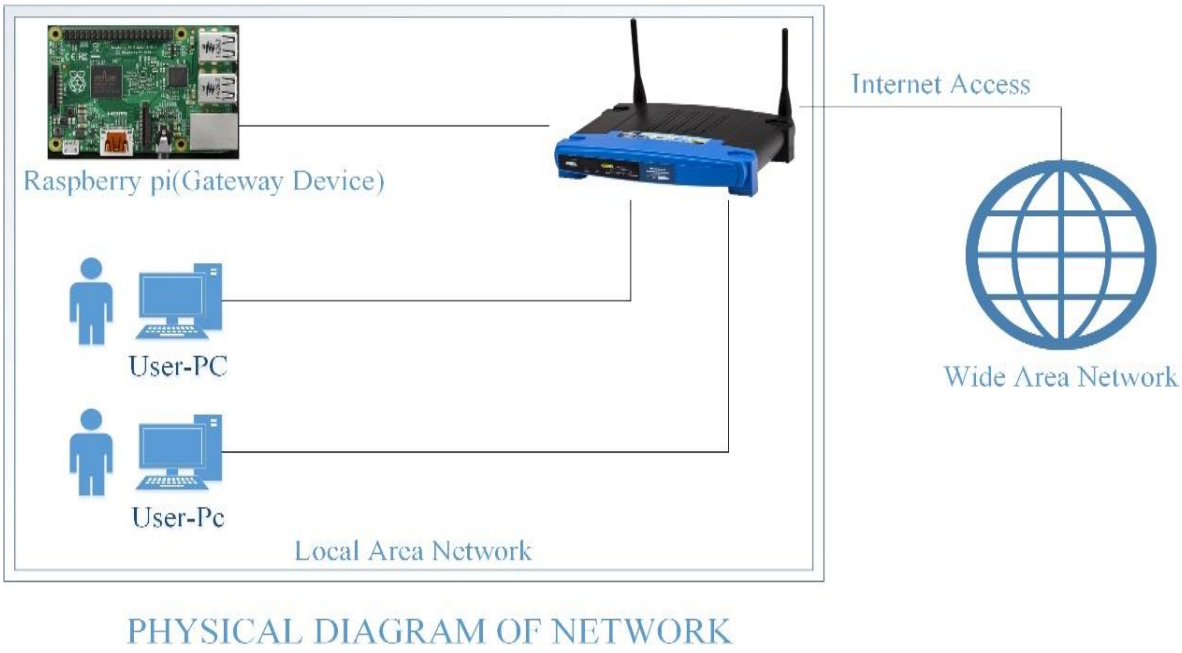


Fig 6.1 physical setup of network

In, fig 3.1 the user pc and the raspberry pi is connected through the router and the router is connected to the internet. Here all the clients get their internet through the raspberry pi.

The role of router is only to switching the packets from internetwork to wide Area network.

- **Logical Setup of the network**

The below image shows how the packet travels logically in the network.

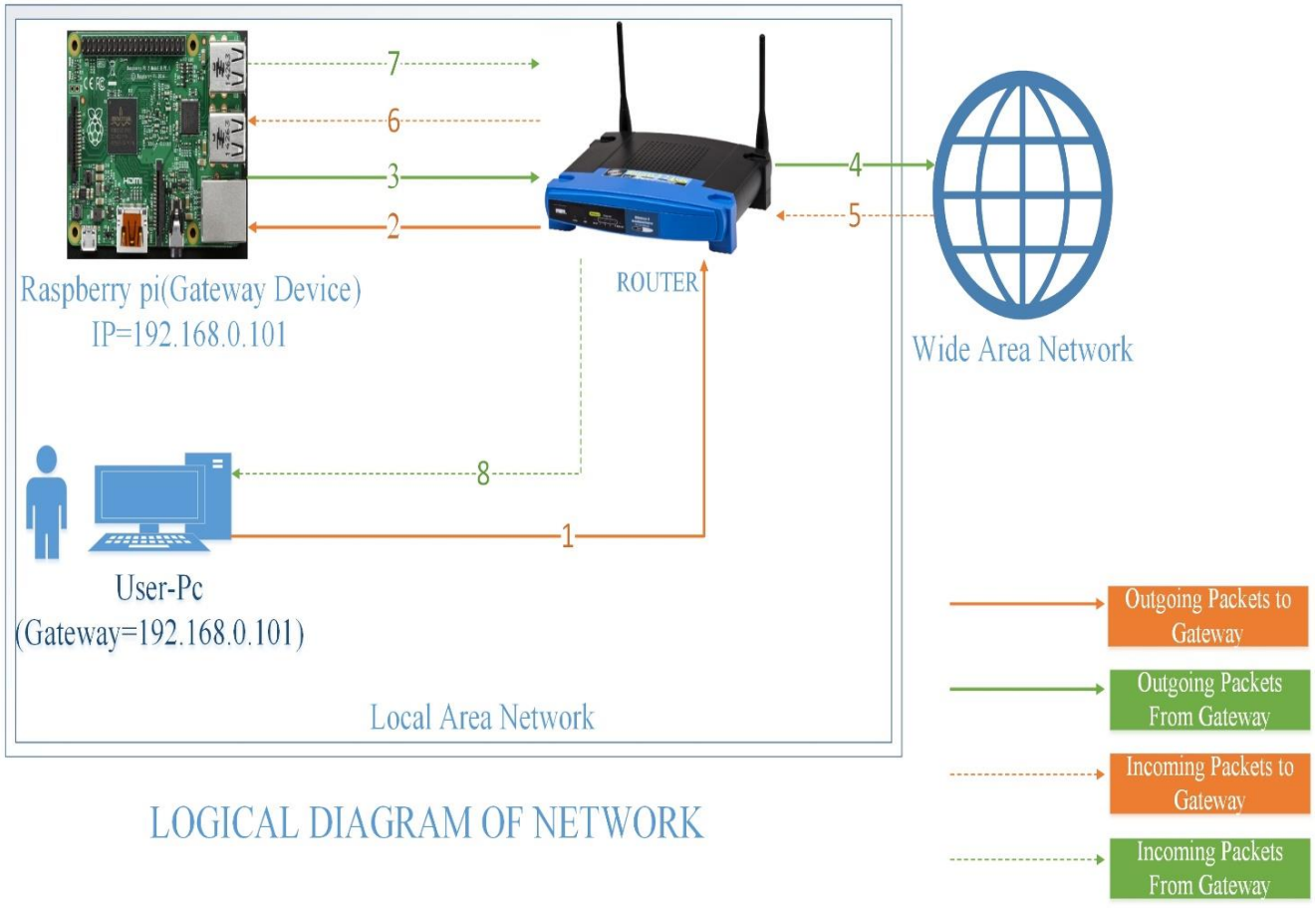


Fig 6.2 Logical setup of network

In, Fig 3.2 the arrow 1 shows the packet leaving from a PC to the router. The router then sends the packet to raspberry pi in 2. The Pi then check if the packet has to be forwarded or blocked and if it has to be forwarded it send it back to the router in 3. The arrow 4 indicates the packet moving from the router to the Internet service or website. In arrow 5 the incoming traffic is shown using dotted line and it arrives at the router. The router then forwards the packets to the Pi at 6 and the pi route the packets to the Pc though the router at 7 and 8 arrows.

• 6.2 MODULES SPECIFICATION

The modules involved in the application are:

- User Login / Register
- Access Control
- Packet Monitoring
- Mail Notification
- File Transfer and Messaging
- Manage User Accounts
- Bandwidth Usage and System (Pi) Temperature Monitoring

- **Module Wise Description:**

  - **User Login / Register:** Login module is used to check whether the user is an authorized person or not. Also the privileges allocated to that particular user can be determined from this. We also get current IP of the user while login only for file transfer or messaging. For this the user should give the correct username and password. User must have to register an account and the admin has to validate his account before using functionalities within the application.

  - **Access Control:** Through this module we control the network access provided to a particular user. We can block any Website or Service for a particular user or privilege level or for the whole network through this module. Also we can specify an upper limit on the amount of data usage for a user of a particular privilege level. We can also block a particular user if we want so that he/she can't use any internet services or website. We can also revert or change all the above mentioned functionalities as per our wish. There is also an option to implement any iptables rule though our project for Advanced users.

  - **Packet Monitoring:** Though this module an admin can monitor each and every packet moving around the system. This is an advanced option provided to a admin if he wants to use. The admin also can apply filter for seeing packets from a particular source or destination IP or port or for a particular protocol. Though this module a raw pcap file is also stored which

can be used for deep packet inspection in other dedicated packet monitoring software's like Wireshark.

- **Mail Notification:** Through this module an admin can select to get mail updates of all the ongoing activities periodically. He can add multiple email Id of different members to send them all the periodic updates. He can set the time interval between mails and also select what all the contents that has to be mailed.

- **File Transfer and Messaging:** Through this module the users can do file transferring and messaging inside the network among themselves. The Client can do file transfer or unicast messaging while the Server can do multicast and broadcast messages too. As the Server Side of the application will be running on a raspberry pi dedicated only for this software keeping File transfer module in the server side seemed redundant and hence it will not be available on the server side of the project.

- **Manage User Account:** The admin can change all the details of a client except his username and password through this module he can also upgrade the users privilege level or block or unblock him using this module.

- **Bandwidth Usage and System (Pi) Temperature Monitoring:** Through this module live updates of Bandwidth Usage and Pi's Temperature is displayed on the home tab of the server side.

6.3 SECURITY FEATURES

- Oracle Java platform provides many features for increasing the security of Java all applications. This includes the use of the Java Virtual Machine (JVM), a virtual machine that sandboxes the java code from the rest of the operating system, and a whole host of security APIs that Java programmers can use. Even with this many security features, criticism has been thrown on the programming language, and Oracle, because of an increase in malicious programs that brought out security vulnerabilities in the JVM, which are yet to be properly addressed by Oracle.

6.4 CODING STANDARDS

- **Java Language Rules**

- **Conventions:**

We follow standard Java coding conventions.

- **Package and Import Statements:**

The first non-comment line of most Java source files is a package statement

After that, import statements can follow. For example:

Package

java.awt;import

java.awt.peer.CanvasP

eer;

- **Order Import Statements:**

- The ordering of import statements is:

7. Java and Javafx

2. Imports from third parties (com, mysql, net, org)

To exactly match the IDE settings, the imports should be:

- Alphabetical within each grouping, with capital letters before lower case letters

(e.g. Z before a).

- There should be a blank line between each major grouping (com, junit, net, org, java, javax).

- **Fully Qualify Imports:**

When you want to use class Bar from package foo, there are two possible ways to import it:

import foo. *;

import foo.Bar;

An explicit exception is made for java standard libraries (java.util.*, java.io. *, etc.) and unit test code (junit.framework.*).

- **Number per Line:**

One declaration per line is recommended since it encourages commenting. In other words,

```
int level; // indentation
level int size; // size of
table
```

- **Naming Conventions:**

Naming conventions make programs more understandable by making them easier to read. They can also give information about the function of the identifier-for example, whether it's a constant, package, or class-which can be helpful in understanding the code.

- Use terminology applicable to the domain

  If the users of the system refer to their clients as Customer, then use the term

  Customer for the class, not client.
- Use mixed case to make names readable.
- Use abbreviations sparingly, but if you do so then use then intelligently and document it

- **Don't Ignore Exceptions:**
  - Sometimes it is tempting to write code that completely ignores an exception like this:

```
void setServerPort(String value)
{
   try
      {
         serverPort = Integer.parseInt(value);
       } catch (NumberFormatException e) {   }
 }
```

- One must never do this. Even though these exceptions rarely ever happen, ignoring exceptions like these can come back to haunt you some day and then if you have not handled the exception correctly you find it too hard to find out what set off this problem in you code. So you must handle each and every Exception in the code in the proper way. The specific way for handling varies from case to case.

6.5 SAMPLE CODING

Start Page | Accesscontrol.java | Pcaper.java | Client.java | Server.java

Source | History

```
69
70
71
72
73      public static void userunblock(String usrunblock ) throws IOException{
74
75       String[] command = { "sudo","iptables","-D" ,"FORWARD","-s",usrunblock,"-j","DROP" };
76       command_run(command);
77
78
79      }
80
81      public static void checkweb(String ipaddr) throws UnknownHostException
82      {
83      InetAddress ip=InetAddress.getByName(ipaddr);
84      String hostaddr=ip.getHostAddress();
85      //System.out.println(ipaddr+" : "+hostaddr);
86      }
87
88         public static String iptohost(byte[] addr) throws UnknownHostException
89      {
90      InetAddress ip=InetAddress.getByAddress(addr);
91      String hostaddr=ip.getCanonicalHostName();
92      //System.out.println(ipaddr+" : "+hostaddr);
93      return(hostaddr);
94      }
95
96      public static String command_run(String[] args) throws IOException{
97           Process p = Runtime.getRuntime().exec(args);
98           BufferedReader br = new BufferedReader(new InputStreamReader(p.getErrorStream()));
99           //System.out.println(br.readLine());
100          String line="",l;
101          while((l=br.readLine())!=null)
102              line+=l+"\n";
103          return(line);
104     }
105     public static String command_run(String args) throws IOException{
106
107          Process p = Runtime.getRuntime().exec(args);
108          BufferedReader br = new BufferedReader(new InputStreamReader(p.getErrorStream()));
109          String line="",l;
110          while((l=br.readLine())!=null)
111              line+=l+"\n";
112          return(line);
```

Start Page | Accesscontrol.java | Pcaper.java | Client.java | Server.java

Source | History

```
90              //readpacketfromfile();
91              Thread cap=new Thread(new Runnable() {public void run(){
92              capturepackets(pcap);}});
93              cap.start();
94          }
95          static Pcap startpcap()
96            {
97              List<PcapIf> alldevs = new ArrayList<>(); // Will be filled with NICs
98              final StringBuilder errbuf = new StringBuilder(); // For any error msgs
99              int r = Pcap.findAllDevs(alldevs, errbuf);
100             if (r == Pcap.NOT_OK || alldevs.isEmpty()) {
101                 JOptionPane.showMessageDialog(null, "Can't read list of devices");
102                 return(null);
103             }
104
105             PcapIf device = alldevs.get(1); // We know we have atleast 1 device
106             final String s=device.getName();
107            // JOptionPane.showMessageDialog(null, s );
108             final int snaplen = 64 * 1024;           // Capture all packets, no trucation
109             final int flags = Pcap.MODE_NON_PROMISCUOUS; // capture all packets
110             final int timeout = 100;           // 1 seconds in millis
111             final Pcap pcap = Pcap.openLive(s, snaplen, flags, timeout, errbuf);
112             if (pcap == null) {
113                 System.err.printf("Error while opening device for capture: " + errbuf.toString());
114                 return(null);
115             }
116           Thread dump = new Thread(new Runnable() {public void run(){
117            writepackettofile(s, snaplen, flags, timeout, errbuf);
118            }});
119           dump.start();
120            return(pcap);
121     }
122
123         static void  capturepackets(Pcap pcap)
124           {
125                 final Ethernet eth=new Ethernet();
126                 final Ip4 ip=new Ip4();
127                 final Ip6 ip6=new Ip6();
128                 final Arp arp=new Arp();
129                 JBufferHandler<String> handler = new JBufferHandler<String>() {
130                 private final PcapPacket packet = new PcapPacket(JMemory.POINTER);
131                 public void nextPacket(PcapHeader header, JBuffer buffer, String user) {
132                 nobytes+= buffer.size();
133                     if(livemonitoring)
```

...ava | Pcaper.java | Client.java | Server.java | FXMLDocumentController.java | FXMLDocument.fxml

```
1  <?xml version="1.0" encoding="UTF-8"?>
```

```
1  /*
2   * To change this license header, choose License Headers in Project Properties.
3   * To change this template file, choose Tools | Templates
4   * and open the template in the editor.
5   */
6  package server8;
7
8  import java.util.Random;
9
10 public class passGen{
11     //Verison 1.0
12     private static final String dCase = "abcdefghijklmnopqrstuvwxyz";
13     private static final String uCase = "ABCDEFGHIJKLOMNOPQRSTUVWXYZ";
14     private static final String sChar = "!@#$%^&*<>";
15     private static final String intChar = "0123456789";
16     private static Random r = new Random();
17     private static String pass = "";
18
19     public static String gens () {
20         System.out.println ("Generating pass...");
21         while (pass.length () != 5){
22             int rPick = r.nextInt(4);
23             if (rPick == 0){
24                 int spot = r.nextInt(dCase.length());
25                 pass += dCase.charAt(spot);
26             } else if (rPick == 1) {
27                 int spot = r.nextInt (uCase.length());
28                 pass += uCase.charAt(spot);
29             } else if (rPick == 2) {
30                 int spot = r.nextInt (7);
31                 pass += sChar.charAt(spot);
32             } else if (rPick == 3){
33                 int spot = r.nextInt (10);
34                 pass += intChar.charAt (spot);
35             }
36         }
37         System.out.println ("Generated Pass: " + pass);
38         return(pass);
39     }
40 }
```

...age | Accesscontrol.java | Pcaper.java | Client.java | Server.java | FXMLDocumentContr

```
1  <?xml version="1.0" encoding="UTF-8"?>
4
5  .sectionTitle {
6      -fx-text-fill    : #01579B;
7      -fx-font-weight : bolder;
8      -fx-font-size    : 14px;
9      }
10
11 .warningText {
12     -fx-fill : #0277BD;
13     }
14
15 .inputSection {
16     -fx-background-color : #E1F5FE;
17     }
18
19 .button {
20     -fx-background-color : #1ABC9C;
21     -fx-text-fill        : #E1F5FE;
22     -fx-font-weight      : 900;
23     }
24
25 .button:hover {
26     -fx-background-color : #1DE9B6;
27     }
28         .theback{
29             -fx-background-color : #ECF0F1;
30         }
31         .width{
32             -fx-border-width : 25px;
33 border-width: 15px;
34
35         }
36  .tab *.tab-label {
37     -fx-rotate: 90;
38 }
39
40 .tab {
41     -fx-padding: 5em 1em 5em 1em;
42     -fx-focus-color: transparent;
43 }
```

Start Page | Accesscontrol.java | Pcaper.java | Client.java | Server.java | FXMLDocumentCo

```
1   <?xml version="1.0" encoding="UTF-8"?>
79  static boolean chcklogindata(String user, String pass){
80      Connection conn = null;
81      Statement stmt = null;
82      boolean loginb=false;
83     try{
84
85       Class.forName("com.mysql.jdbc.Driver");
86       conn = DriverManager.getConnection(DB_URL,USER,PASS);
87       stmt = conn.createStatement();
88       String sql;
89       sql = "SELECT * FROM ulogin";
90       ResultSet rs = stmt.executeQuery(sql);
91
92       while(rs.next()){
93          //Retrieve by column name
94          String first = rs.getString("uid");
95          String last = rs.getString("pass");
96          if(user.equals(first)&&pass.equals(last))
97          {
98              loginb=true;
99              prvl= rs.getInt("privillage");
100             break;
101
102         }
103
104      }
105      //STEP 6: Clean-up environment
106      rs.close();
107      stmt.close();
108      conn.close();
109
110     }catch(SQLException se){
111      //Handle errors for JDBC
112      se.printStackTrace();
113    }catch(Exception e){
114      //Handle errors for Class.forName
115      e.printStackTrace();
116    }finally{
117      //finally block used to close resources
118      try{
119         if(stmt!=null)
120            stmt.close();
121      }catch(SQLException se2){
122      }
```

Start Page | Accesscontrol.java | Pcaper.java | Client.java | Server.java | FXMLDocumentController.java

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
package server8;
import java.io.BufferedReader;
import java.io.DataOutputStream;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.io.OutputStream;
import java.net.ServerSocket;
import java.net.Socket;

public class chat {

static ServerSocket msgsoc;
public static void sendmsg(String[] args) throws Exception
  {

     Socket sock = new Socket(args[0], 5216);
     OutputStream ostream = sock.getOutputStream();
     DataOutputStream dos = new DataOutputStream(ostream);
     dos.writeBytes("Admin : "+args[1]);
     dos.close();
     ostream.close();
     sock.close();
    }

public static String recievemsg() throws Exception
   {
     msgsoc=new ServerSocket(5216);
     Socket sock = msgsoc.accept();
     InputStream istream = sock.getInputStream();
     BufferedReader br=new BufferedReader(new InputStreamReader(istream));
     String message2="";
     while(br.ready())
     message2+= br.readLine()+"\n";
     br.close();
     istream.close();
     sock.close();
     msgsoc.close();
     return(message2+"\n");
  }
}
```

Start Page | Accesscontrol.java | Pcaper.java | Client.java | Server.java | FXMLDocumentController.java | FXMLDocument.fxml | Gmail.java | Login

```
1   <?xml version="1.0" encoding="UTF-8"?>
13  import javafx.scene.paint.Color;
14  import javafx.scene.shape.Rectangle;
15
16  public class bandwidth  {
17
18  public static Gauge download,temp;
19  public static Thread moveNeedle=new Thread();
20      public static void createbwm(GridPane bwpane) {
21          GaugeBuilder builder = GaugeBuilder.create().skinType(Gauge.SkinType.BULLET_CHART);
22          download  = builder.decimals(1).maxValue(500).unit("KB/Sec").title("Bandwidth").build();
23          VBox downBox = getTopicBox("Bandwidth Usage", Color.rgb(77,208,225), download);
24          bwpane.setPadding(new Insets(20));
25          bwpane.setBackground(new Background(new BackgroundFill(Color.rgb(244,244,244), CornerRadii.EMPTY, Insets.EMPTY)));
26          bwpane.add(downBox, 0, 0);
27
28          GaugeBuilder builder2 = GaugeBuilder.create().skinType(Gauge.SkinType.BULLET_CHART);
29          temp  = builder2.decimals(1).maxValue(100).unit("°C").title("Pi Temperature").build();
30          VBox downBox2 = getTopicBox("Temperature", Color.rgb(250,20,25), temp);
31
32
33          bwpane.add(downBox2, 0, 1);
            moveNeedle = new Thread(new Runnable() {public void run(){
                        double down=0;
36            while(true){
37              try {
38                down=Pcaper.nobytes/1024;
39
40                download.setValue(down);
41                temp.setValue(40);
42                Pcaper.nobytes=0;
                   moveNeedle.sleep(1000);//as speed is per second and 1000 makes 1 sec
44                 } catch (InterruptedException ex) {
45                     System.out.println("Interrupt exception in moveneedle");
46                 }
47              }
48  }});
49
50
51          moveNeedle.start();
52      }
53
54
```

Start Page | Accesscontrol.java | Pcaper.java | Client.java | Server.java | FXMLDocumentController.java | FXMLDocument.fxml

```
1  <?xml version="1.0" encoding="UTF-8"?>
7  import java.util.logging.Logger;
8  import javafx.fxml.FXML;
9  import javafx.fxml.FXMLLoader;
10 import javafx.fxml.Initializable;
11 import javafx.scene.Parent;
12 import javafx.scene.Scene;
13 import javafx.scene.control.PasswordField;
14 import javafx.scene.input.MouseEvent;
15 import javafx.stage.Stage;
16
17 public class LoginController implements Initializable {
18     @FXML
19     private PasswordField adminps;
20
21     Stage stage;
22     /**
23      * Initializes the controller class.
24      */
25     @Override
26     public void initialize(URL url, ResourceBundle rb) {
27
28     }
29
30     @FXML
31     private void adminlogin(MouseEvent event) {
32         //if(databse.chcklogindata("Admin", adminps.getText()))
33         if(true){
34             try {
35             stage=(Stage)adminps.getScene().getWindow();
36             Parent root1;
37             stage.close();
38             root1 = FXMLLoader.load(getClass().getResource("FXMLDocument.fxml"));
39             Scene scene1 = new Scene(root1);
40             stage.setScene(scene1);
41         } catch (IOException ex) {
42             Logger.getLogger(LoginController.class.getName()).log(Level.SEVERE, null, ex);
43         }
44             stage.show();
45             Server.addtolog("Server Side Program Started");
46          }
```

Start Page | Accesscontrol.java | Pcaper.java | Client.java | Server.java | FXMLDocumentController.java | FXMLDocument.fxml

```
1  <?xml version="1.0" encoding="UTF-8"?>
10 import javax.mail.*;
11 import javax.mail.internet.*;
12
13 import javax.mail.PasswordAuthentication;
14
15
16 public class Gmail {
17
18 static ArrayList<String> to=new ArrayList<>();
19 public static void mail(String args) {
20
21   //Get the session object
22   Properties props = new Properties();
23   props.put("mail.smtp.host", "smtp.gmail.com");
24   props.put("mail.smtp.socketFactory.port", "465");
25   props.put("mail.smtp.socketFactory.class",
26             "javax.net.ssl.SSLSocketFactory");
27   props.put("mail.smtp.auth", "true");
28   props.put("mail.smtp.port", "465");
29
30   Session session = Session.getDefaultInstance(props,
31    new javax.mail.Authenticator() {
32    protected PasswordAuthentication getPasswordAuthentication() {
33    return new PasswordAuthentication("rpiproject0@gmail.com","rpiproject0@gtu.com");//change accordingl
34    }
35   });
     for(int i=0;i<to.size();i++){
37   //compose message
38   try {
39    MimeMessage message = new MimeMessage(session);
40    message.setFrom(new InternetAddress("rpiproject0@gmail.com"));//change accordingly
41    message.addRecipient(Message.RecipientType.TO,new InternetAddress(to.get(i)));
42    message.setSubject("Updates From Your Network");
43    message.setText("\n\nThe Recent updates from your network are : \n\n"+args);
44
45    //send message
46    Transport.send(message);
47    System.out.println("message sent successfully");
48
49   } catch (MessagingException e) {throw new RuntimeException(e);}
50    }
51  }
52 }
53
```

Start Page | Accesscontrol.java | Pcaper.java | Client.java | Server.java | FXMLDocumentController.java

```
String reason=split[4];
568         }
569     }
570
571     private void pausemonitoring(MouseEvent event) {
572         if(Pcaper.livemonitoring)
573         {
574         pause=true;
575         Pcaper.livemonitoring=false;
576         }
577     }
578
579     @FXML
580     private void showtotalcapturedata(MouseEvent event) {
581          Alert alert = new Alert(Alert.AlertType.INFORMATION);
582          alert.setTitle("Information Dialog");
583          alert.setHeaderText("Opening Captured Data");
584          alert.setContentText("It might take some time to load fully depending on the capture size.....");
585          alert.showAndWait();
586          Pcaper.sholddata=true;
587          Pcaper.data.clear();
588          read=new Thread(new Runnable() {public void run(){
589          Pcaper.readpacketfromfile(10);}});
590          read.start();
591          Thread th = new Thread(pctsk);
592          th.setDaemon(true);
593          th.start();
594     }
595
596     @FXML
597     private void close(ActionEvent event) {
598          Alert alert = new Alert(Alert.AlertType.CONFIRMATION);
599          alert.setTitle("Exit");
600          alert.setContentText("Do you really want to CLOSE this application now??");
601          ButtonType buttonTypeOne = new ButtonType("Yes");
602          ButtonType buttonTypeTwo = new ButtonType("No");
603          alert.getButtonTypes().setAll(buttonTypeOne, buttonTypeTwo);
604
605           Optional<ButtonType> result = alert.showAndWait();
606           if (result.get() == buttonTypeOne){
607
608               Server.closeprogram();
609            }
610
```

Start Page | Accesscontrol.java | Pcaper.java | Client.java | Server.java

Source | History

```
582  }
583
584  public static void register(String[] split,DataOutputStream dout,Socket sock){
585          String fname=split[1];
586          String uid=split[2];
587          String pass=split[3];
588          String email=split[4];
589          String mobno=split[5];
590         // System.out.println(user+"::"+pass);
591      try {
592          boolean log = databse.wrtregdata(fname, uid, pass, email, mobno);
593          if (log) {
594              dout.writeUTF("reg");
595              addtolog("New User " + uid + " has requested for registration");
596          } else {
597              dout.writeUTF("false");
598          }
599
600      } catch (IOException ex) {
601          Logger.getLogger(Server.class.getName()).log(Level.SEVERE, null, ex);
602      }
603  }
604  }
605
606  public static void shwupalrt(String usid,String usname,String emil,String crrst,String mbno,Integer prvlev){
607          try {
608              ui=usid;
609              un=usname;
610              ue=emil;
611              um=mbno;
612              up=prvlev+"";
613              Stage st=new Stage();
614              Parent rot = FXMLLoader.load(Server.class.getResource("userdetail.fxml"));
615              Scene scene = new Scene(rot);
616              st.setScene(scene);
617              st.show();
618          } catch (IOException ex) {
619              Logger.getLogger(Server.class.getName()).log(Level.SEVERE, null, ex);
620          }
621  }
622
623  public static class servoblist{
```

CHAPTER

7

TESTING

CHAPTER 7

TESTING

7.1 TESTING PLAN

Testing is the process of examining all the functionalities of any application to check whether it is working as expected and to make sure that at developer level, unit testing is done. Unit testing is the testing of the smallest unit of a program (i.e. a module or a function). Unit testing is necessary to all the software companies to provide a quality software to their customers.

Unit testing can be done in two ways

Manual testing	Automated testing
Executing the test cases manually without using any tools for it is known as manual testing. • It is time consuming and tedious: Since all test cases manually executed it is very slow and tedious. • More investment in human resources: As all test cases have to be checked manually by testers, more testers are used in manual testing. • Less reliable: Many tests are not performed while manual testing making it less reliable each time because of human errors. • Non-programmable: No program is done to write sophisticated test cases which bring out hidden information.	Using tools for executing the test cases by means of automation tool is known as automation testing. • Automation tools runs test cases much faster than testers recruited for testing purposes. • Lesser investment in human resources: As all test cases are checked by some automation tool, lesser number of tester are used in automation testing. • More reliable: Automation tools perform all tests precisely each time they are used making them more reliable. • Programmable: Testers program sophisticated test cases to fetch hidden information.

- JUnit is the most popular unit testing framework for the Java Programming Language. It belongs to a family of unit testing frameworks known as xUnit.

- JUnit is based on the idea known as "first testing then coding". This kind of approach is usually "code a little, test a little, code a little.....". This increases the productivity of programmers and the stability of the code and hence reduces stress on programmers and the time usually spent on debugging is also significantly reduced.

Features:

- JUnit is an open source framework used for writing & running tests.
- JUnit provides functionality to determine the test methods.
- JUnit provides functionality for testing anticipated results.
- By using JUnit test we can write faster code with improved quality
- JUnit is very simple and very fast. It takes very less time to run its tests.
- JUnit tests can be ordered into test suites enclosing many different test cases.
- JUnit displays test progress in a progress bar that is green in color if test is running correctly and the bar turns red color when a test fails.

7.2 TESTING STRATEGY

Different levels of testing are used in the test process; each level of testing aims to test different aspects of the system.

- Automated testing of Java applications is especially important because of the huge variety of available devices. As it is not possible to test a java application on all possible device configurations, it is common practice to run java test on typical device configurations.
- Java testing is based on JUnit. Testing for Java can be classified into two categories:
  - Tests which require only the JVM
  - Tests which require a full fledge Java based system

- **Testing Preconditions:**
  - It is important in Java testing to have a method called testingPrecondition() which tests all the pre-conditions required for all other tests. In case method

fails, you can immediately come to know that the assumptions for the other tests have been breached.

7.3 TESTING METHODS

- **White Box Testing**
  - White box testing is also known as open box testing or glass testing. For performing a white box testing on any application, the tester needs to possess knowledge of the internal working of the code.

- **Black Box Testing**
  - The method of testing without having any prior knowledge of the internal workings of the program is Black Box testing.
  - The tester is unaware of the system architecture and cannot access to the source code of the application.
  - Usually, while performing a black box test, a tester will interact with the applications UI by giving various inputs and observing output received without actually knowing where and how the inputs are worked upon.

- **Unit Test**
  - A Unit test tests only the functionality of a certain component.
  - Let's, for example, assume a button in a Java application is used to call another function. A unit test would find out if the corresponding objective was completed, not if the second function was called. A functional test would also check if the activity was correctly started.
  - Java uses JUnit 3. This version doesn't use annotations and uses introspection to detect the tests.

- **Integration Test**
  - Integration tests are designed to test the way individual components work jointly.

    Modules that have been unit tested independently are now combined together to test the integration.

- Usually Java Activities require some integration to the system infrastructure to be able to run. They need the Activity lifecycle provided by the Activity Manager, and access to resources, file system, and databases.

- **Interface Testing**

In the system, standards test for GUIs have been performed, which are as follows.

- Testing the screen control for its position and size.
- The position and related labels for all controls were checked.
- Name of the form in system is given appropriately.
- All menu functions and sub functions were verified for correctness.

7.4 TEST CASES

1. Test case for Client Log In:

Project: -Agent Based Network Surveillance System.

Objective: - To check whether user name & Password valid or invalid.

Prepared by: Group- 44589

Page: - Log In

Test Data:

Table 7-1 Test case for Client Sign In

Sr.no.	Steps	Expected Data	Status
1	Enter a Valid Username and Password and then Click Login Button	Should login to the system and Show Dashboard	Pass
2	Enter only a Username	Login Button should remain Disabled	Invalid
3	Enter only a Password	Login Button should remain Disabled	Invalid
4	Enter a blank username and password	Login Button should remain Disabled	Invalid
5	Enter a Wrong Username and Password and Click Login Button	Should Display a message : Wrong Username or Password!	Invalid

2. Test case for Client Register:

Project: -Agent Based Network Surveillance System.

Objective: - To check whether the data entered by the user while registering is valid.

Prepared by: Group- 44589

Page: - Register

Test Data:

Table 7-2 Test case for Client Register

Sr.no.	Steps	Expected Data	Status
1	Enter Full Name, a User Name, a Password and Confirm the Password, an Email Id and a Valid Mobile number	Should Display the message: Your data has been sent to the Admin. You can login once the Admin accepts your request	Pass
2	Enter only a name	Register Button should remain Disabled	Invalid
2	Enter only a Username	Register Button should remain Disabled	Invalid
3	Enter only a Password	Register Button should remain Disabled	Invalid
4	Enter only a Email ID	Register Button should remain Disabled	Invalid
5	Enter only a Mobile No.	Register Button should remain Disabled	Invalid
6	Enter Different Passwords while confirm password	Register Button should remain Disabled	Invalid
7	Leave any one or two fields Empty while entering all other field	Register Button should remain Disabled	Invalid
8	Enter a char or invalid mobile no	Register Button should remain Disabled	Invalid

3. Test case for Client Request to Change Privilege Level:

Project: -Agent Based Network Surveillance System.

Objective: - To check whether the data entered by client while requesting change in privilege level

Prepared by: Group- 44589

Page: - Request to Change Privilege Level

Test Data:

Table 7-3 Test case for Client Request for Privilege Level Change

Sr.no.	Steps	Expected Data	Status
1	Enter a valid Privilege Level and Reason and Click Request Button	The Request must be sent to admin side for consideration	Pass
2	Enter a char or any Value other than 1 or 2 or 3 or 4 in the New Privilege level Field	Request Button should remain Disabled	Invalid
3	Leave the New Privilege Level Field Blank	Request Button should remain Disabled	Invalid
4	Leave the Reason Field Blank	Request Button should remain Disabled	Invalid

4. Test case for Admin Update User Details:

Project: -Agent Based Network Surveillance System.

Objective: - To check whether the data entered by the admin while updating user details is valid.

Prepared by: Group- 44589

Page: - Update User Details

Test Data:

Table 7-4 Test case for Admin Update User Details

Sr.no.	Steps	Expected Data	Status

1	Enter a Name, Email ID, valid Mobile no and Privilege level and Click Update Button	Should Display the message: User data Updated	Pass
2	Leave the Name Field Blank	Update Button should remain Disabled	Invalid
3	Leave the Email ID Field Blank	Update Button should remain Disabled	Invalid
4	Leave the Mobile No Field Blank	Update Button should remain Disabled	Invalid
5	Leave the Privilege level Field Blank	Update Button should remain Disabled	Invalid
6	Enter a char or invalid mobile no	Update Button should remain Disabled	Invalid
7	Enter a char or a Value more than 4 or less than 1 in the Privilege level	Update Button should remain Disabled	Invalid

5. Test case for Admin Accept User Registration:

Project: -Agent Based Network Surveillance System.

Objective: - To check whether the admin enters a valid privilege level while accepting user Registration

Prepared by: Group- 44589

Page: - Accept User Registration

Test Data:

Table 7-5 Test case for Admin Accept User Registration

Sr.no.	Steps	Expected Data	Status
1	Enter a valid Privilege Level and Click Validate	Should Display the message: User Registration Accepted	Pass
2	Enter a char or any Value other than 1 or 2 or 3 or 4	Should Display the message: Enter valid privilege level	Invalid

6. Test case for Admin Adding email member for mail notification:

Project: -Agent Based Network Surveillance System.

Objective: - To check whether the data entered by the admin while adding a new email member to mail notification is valid.

Prepared by: Group- 44589

Page: - Mail Notification

Test Data:

Table 7-6 Test case for Admin Adding email member for mail notification

Sr.no.	Steps	Expected Data	Status
1	Enter a Name, Email ID Click Update Button	New member is added to mail notification and is shown in the notification table	Pass
2	Leave the Name Field Blank	Add Button should remain Disabled	Invalid
2	Leave the Email ID Field Blank	Add Button should remain Disabled	Invalid

7. Test case for Changing Admin Password:

Project: -Agent Based Network Surveillance System.

Objective: - To check whether the data entered by the admin while changing Admin password.

Prepared by: Group- 44589

Page: - Mail Notification

Test Data:

Table 7-7 Test case for Changing Admin Password

Sr.no.	Steps	Expected Data	Status
1	Enter a the correct old password and a new password and confirm that password and click on Change Button	Should Display the message: Admin Password Changed	Pass
2	Enter a wrong old password and a new password and confirm that password and click on Change Button	Should Display the message: Wrong Password... Try Again	Invalid
3	Leave the Old Password Field Blank	Change Button should remain Disabled	Invalid
4	Leave the New Password Field Blank	Change Button should remain Disabled	Invalid
5	Leave the Re-type Password Field Blank	Change Button should remain Disabled	Invalid
6	Enter Different Passwords while confirm password	Change Button should remain Disabled	Invalid

8. Test Results while running on raspberry pi 2:

The following results were obtained when the memory was split 30-70 % between GPU and RAM

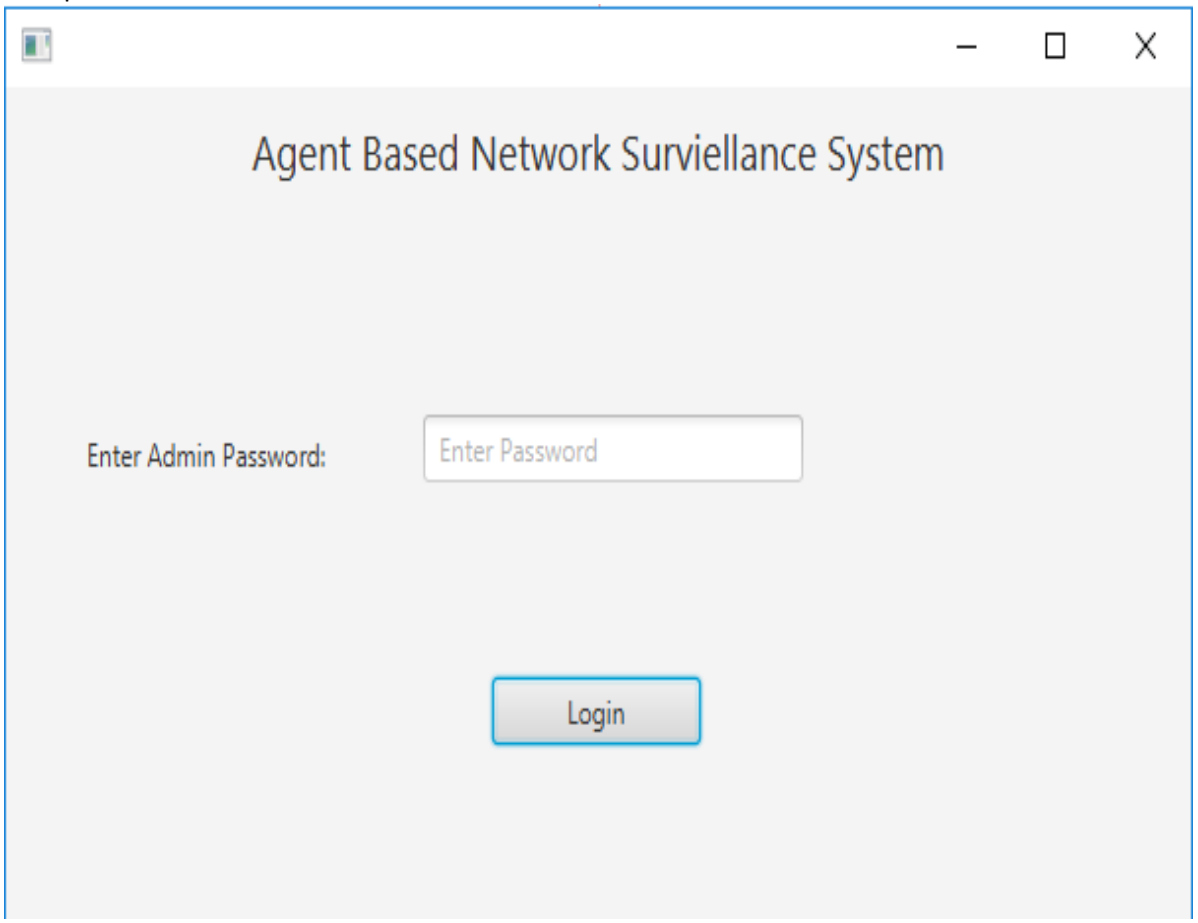
Table 7-8 Raspberry Pi Statistics

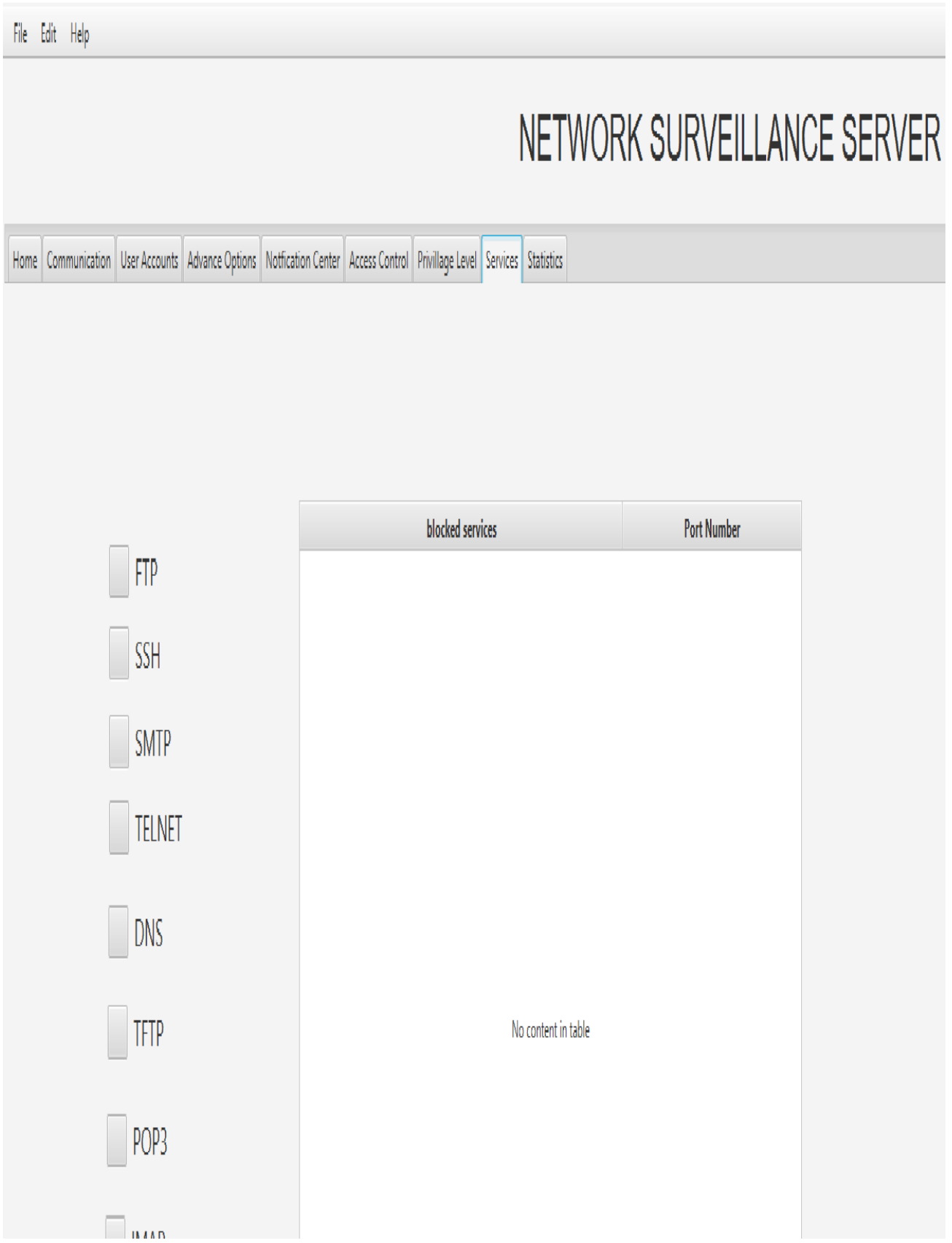
State of PI	RAM used(MB)	GPU memory used(MB)	Network utilization	Threads	CPU usage (%)
Before program starts	150-200	<50	Depends on Internet Speed	20-30	10
When program is idle	250-270	=<150	Same as Above	50-60	70-75
When program is in Usage	300-350	150-200	54 Mbps	65-80	76-99

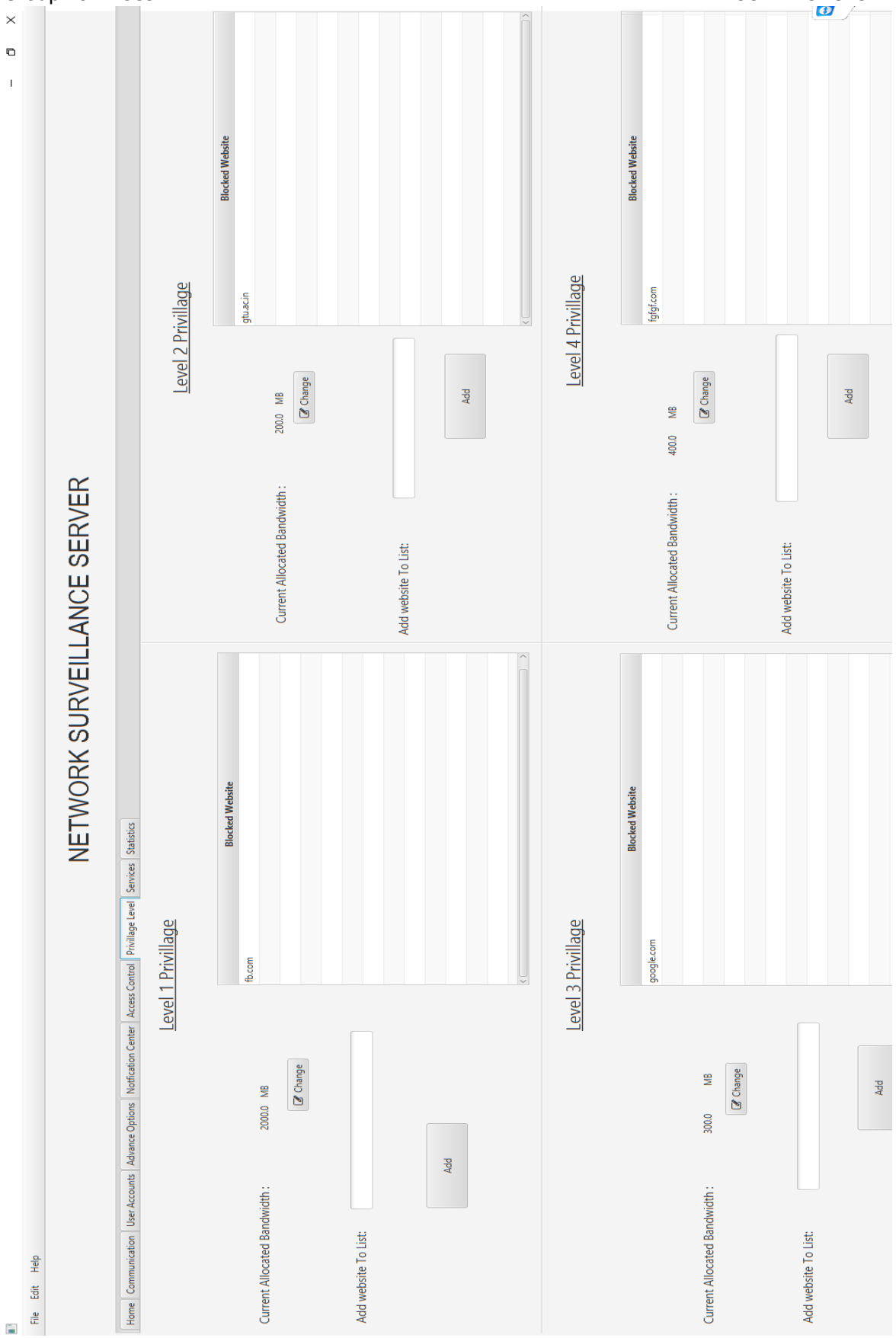
CHAPTER

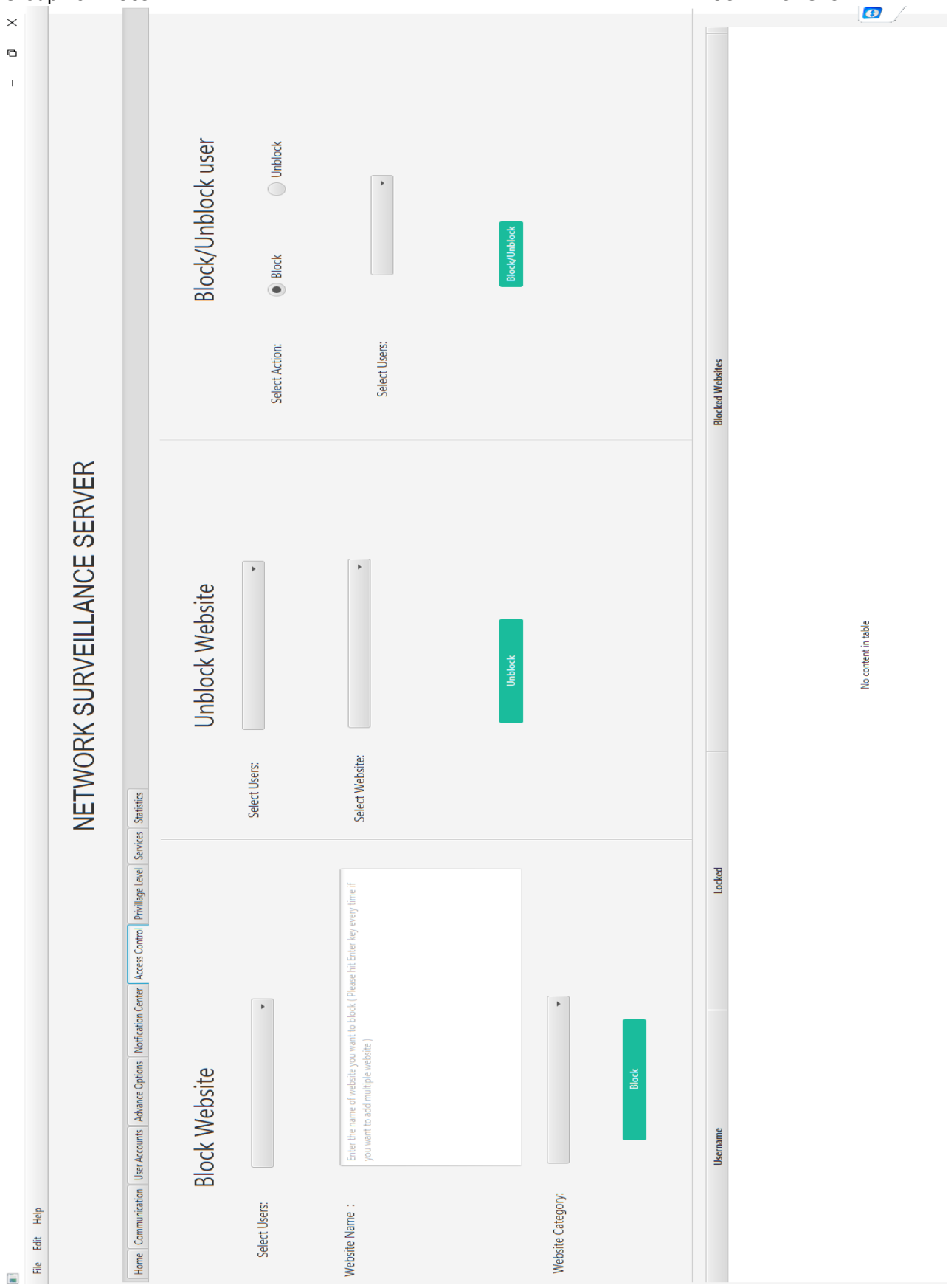
8

SCREENSHOTS


Agent Based Network Surviellance System
Enter Admin Password:
Enter Password
Login

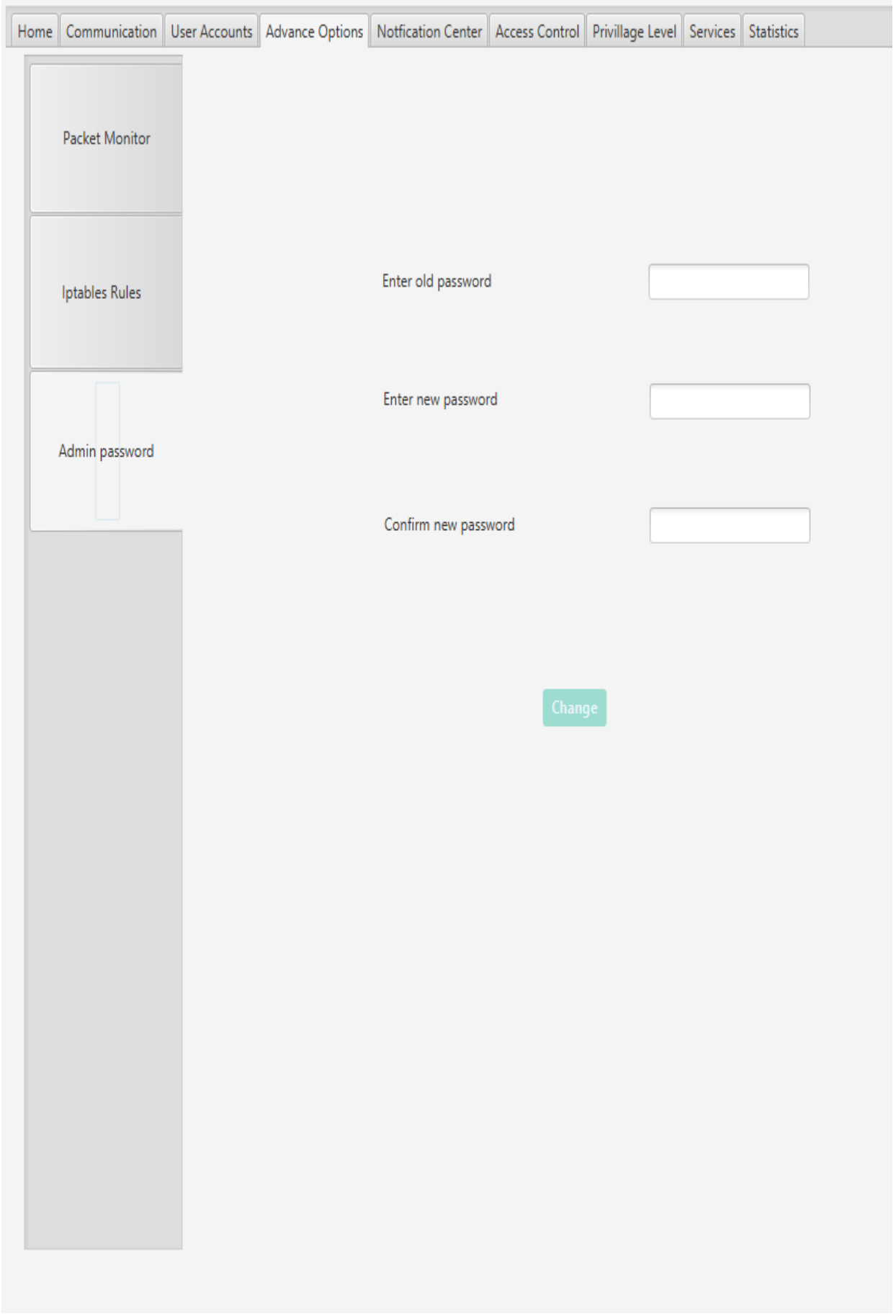

File Edit Help
NETWORK SURVEILLANCE SERVER
Home
Communication
User Accounts
Advance Options
Notification Center
Access Control
Privillage Level
Services
Statistics
FTP
SSH
SMTP
TELNET
DNS
TFTP
POP3
blocked services
Port Number
No content in table

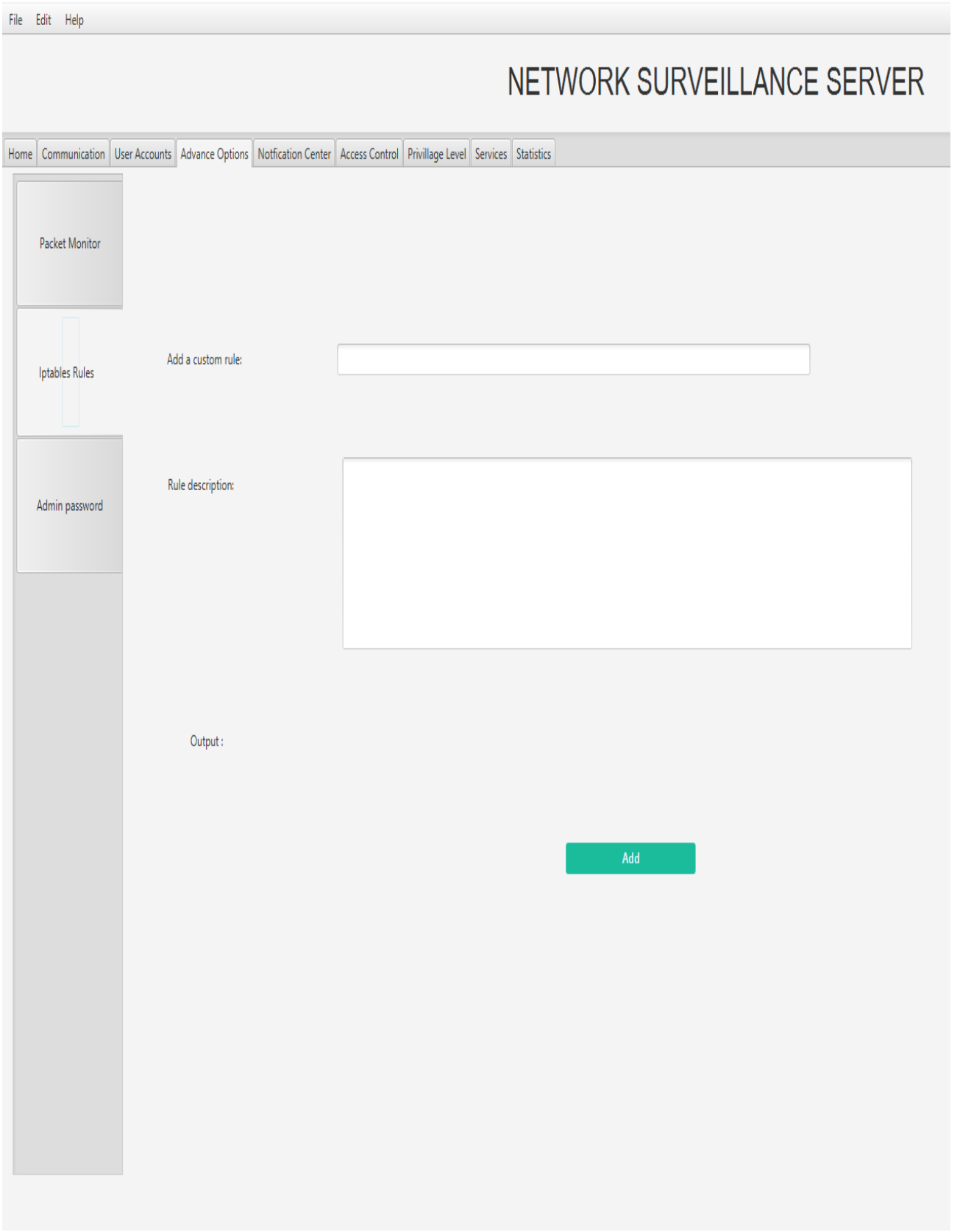

NETWORK SURVEILLANCE SERVER
File Edit Help
Home Communication User Accounts Advance Options Notification Center Access Control Privillage Level Services Statistics
Level 1 Privillage
Blocked Website
fb.com
Current Allocated Bandwidth : 2000.0 MB
Change
Add website To List:
Add
Level 2 Privillage
Blocked Website
gtu.ac.in
Current Allocated Bandwidth : 200.0 MB
Change
Add website To List:
Add
Level 3 Privillage
Blocked Website
google.com
Current Allocated Bandwidth : 300.0 MB
Change
Add website To List:
Add
Level 4 Privillage
Blocked Website
fgfgf.com
Current Allocated Bandwidth : 400.0 MB
Change
Add website To List:
Add

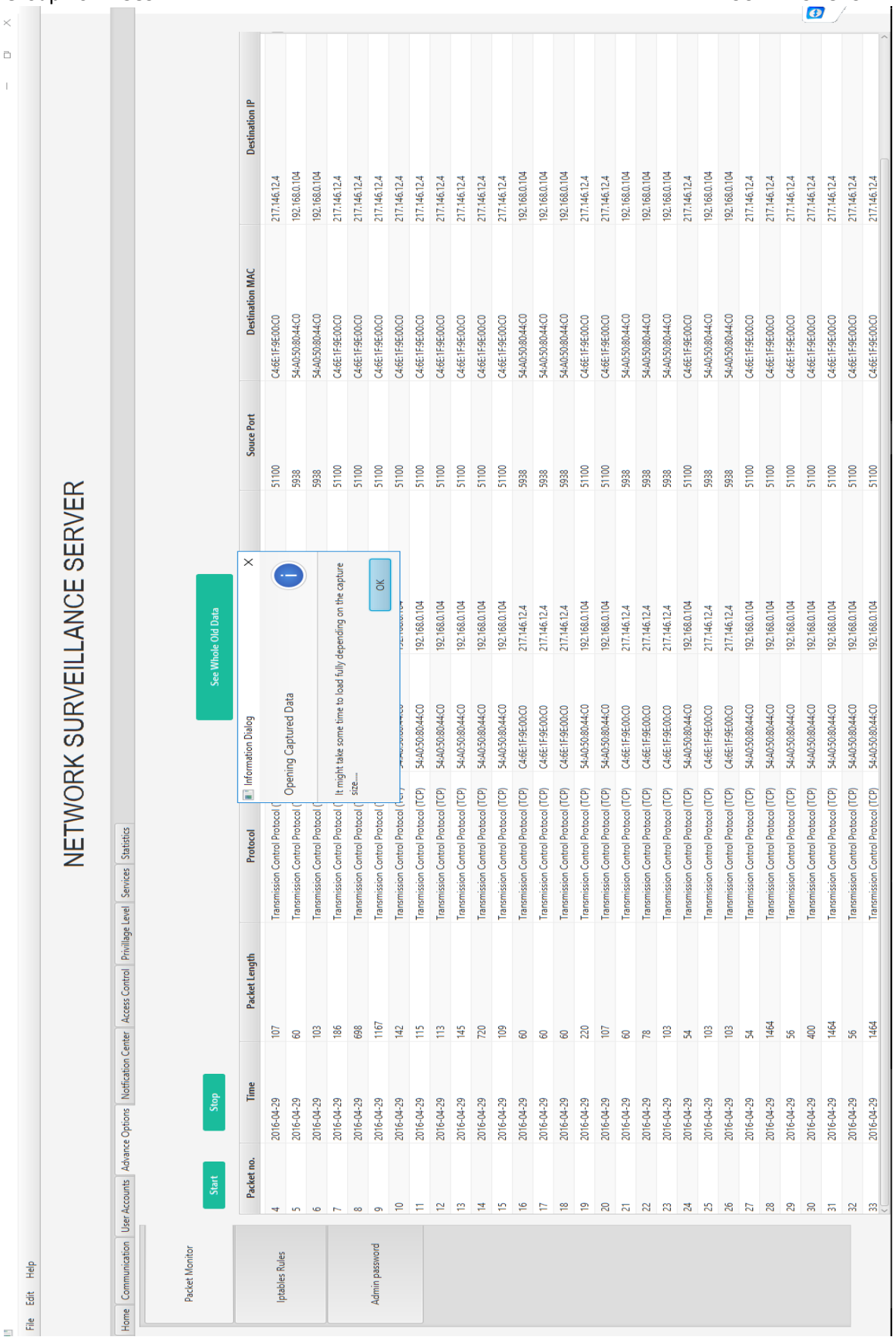

NETWORK SURVEILLANCE SERVER
File Edit Help
Home Communication User Accounts Advance Options Notification Center Access Control Privillage Level Services Statistics
Block Website
Select Users:
Website Name :
Enter the name of website you want to block (Please hit Enter key every time if you want to add multiple website)
Website Category:
Block
Unblock Website
Select Users:
Select Website:
Unblock
Block/Unblock user
Select Action:
Block
Unblock
Select Users:
Block/Unblock
Username
Locked
Blocked Websites
No content in table

NETWORK SURVEILLANCE SERVER
File Edit Help
Home Communication User Accounts Advance Options Notification Center Access Control Privillage Level Services Statistics
Mail Notifications
Disabled
Update the data that has to be mailed
Bandwidth Usage
Session Time
Most Accessed Website
Alerts
Change Mail Time Period
Current Status
Bandwidth usage
Session Time
Most accessed websites
Alerts
Send mail now
Add a Member to Mailing List
Name
Email ID
ADD
Remove a Member to Mailing List
Name
Remove
Alerts
No content in table
Name
Email ID

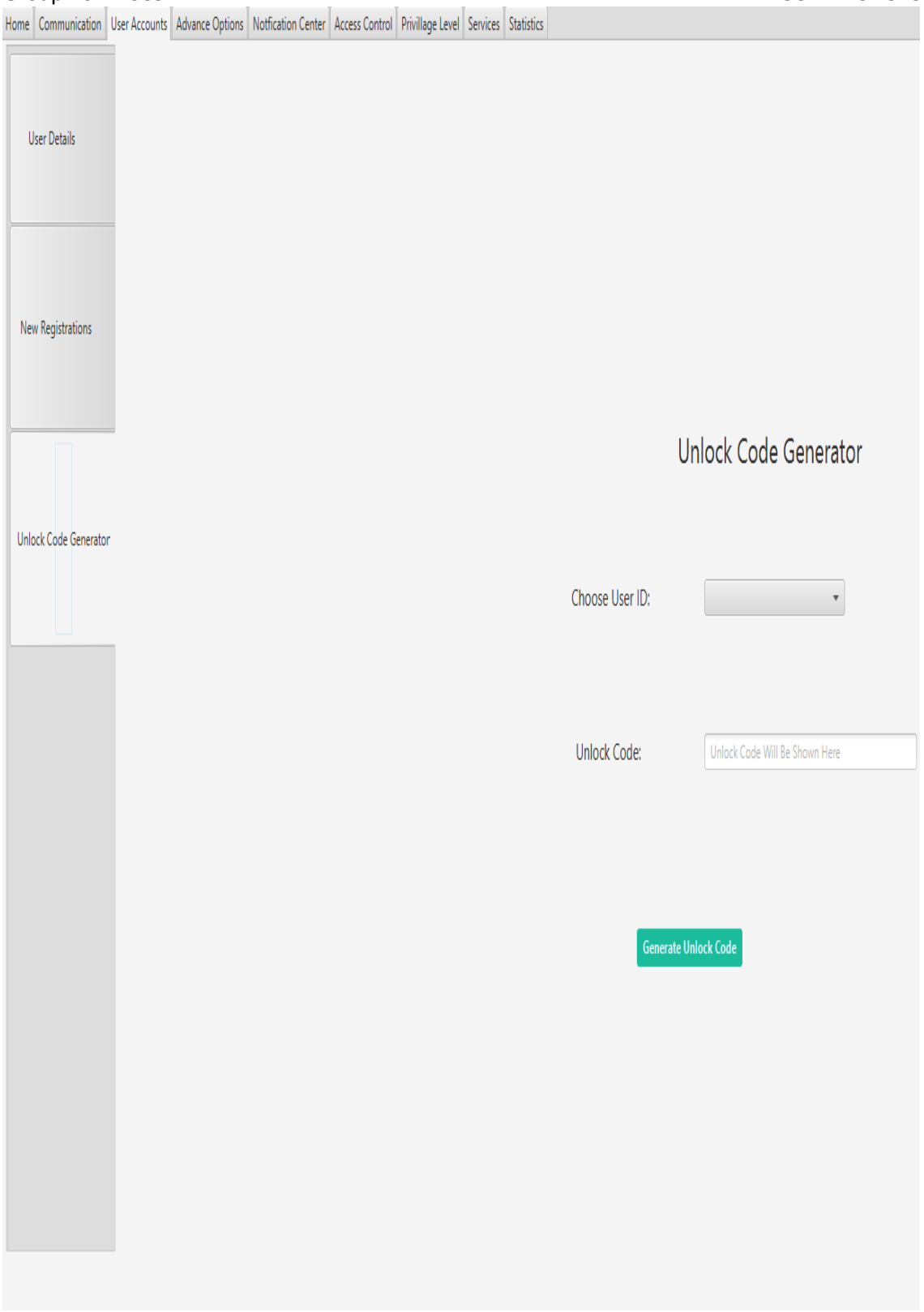
NETWORK SURVEILLANCE SERVER
File Edit Help
Home Communication User Accounts Advance Options Notification Center Access Control Privillage Level Services Statistics
Mail Notifications
Enabled
Update the data that has to be mailed
Bandwidth Usage
Session Time
Most Accessed Website
Alerts
Change Mail Time Period
Current Status
Bandwidth usage
Session Time
Most accessed websites
Alerts
Send mail now
Add a Member to Mailing List
Name
Email ID
ADD
Remove a Member to Mailing List
Name
Remove
Name
Email ID
Alerts
No content in table

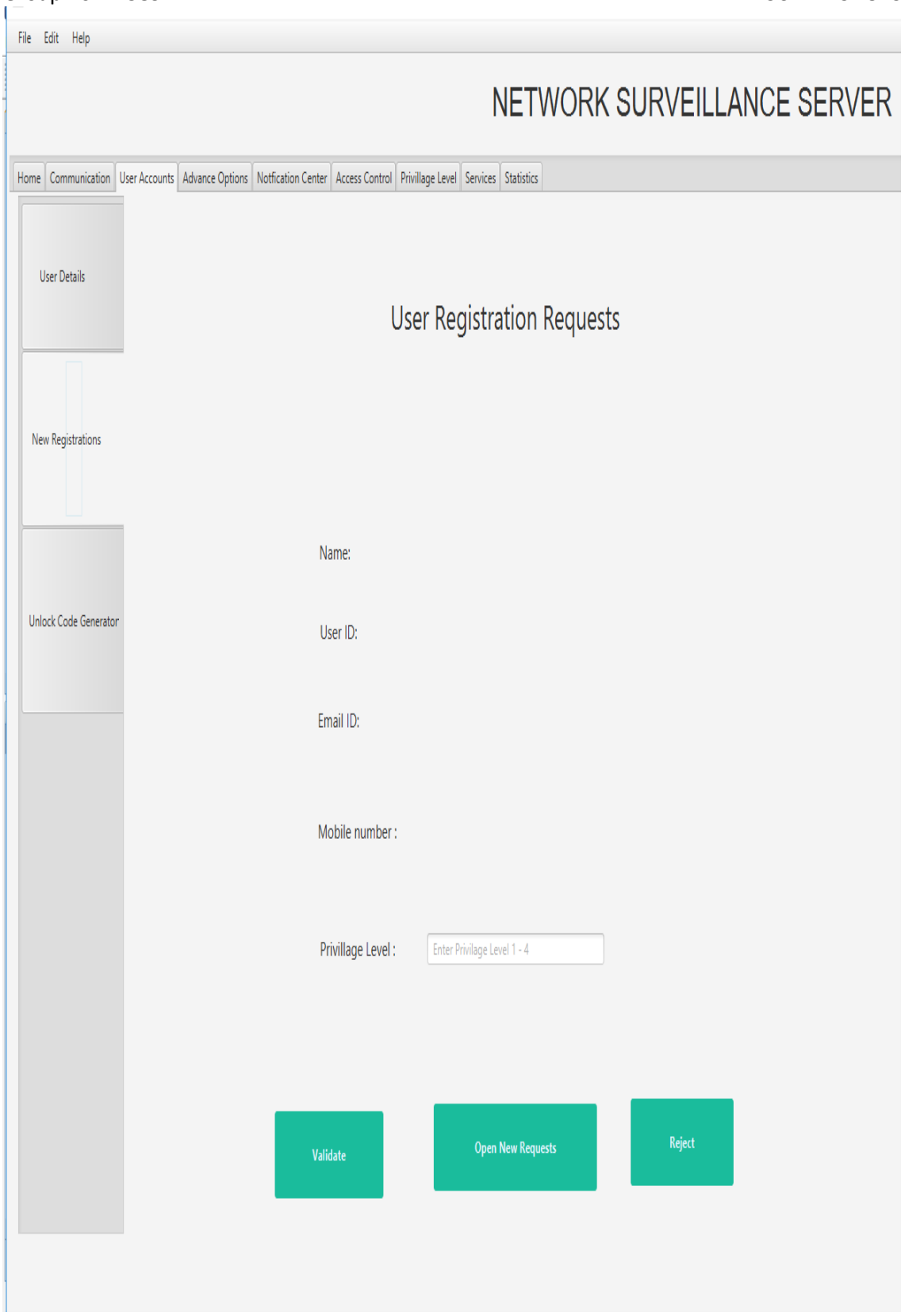

Home
Communication
User Accounts
Advance Options
Notfication Center
Access Control
Privillage Level
Services
Statistics
Packet Monitor
Iptables Rules
Admin password
Enter old password
Enter new password
Confirm new password
Change

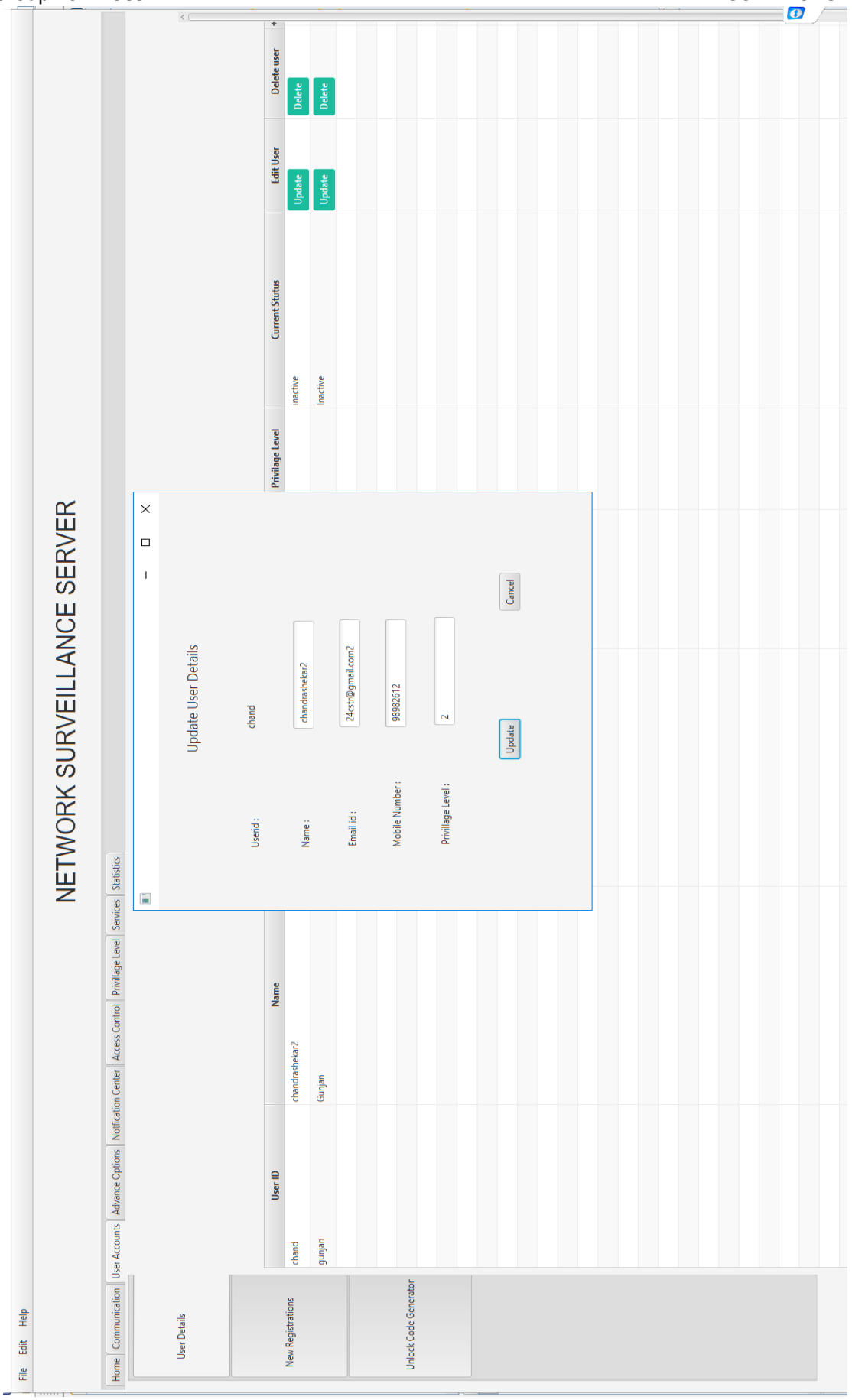

File Edit Help
NETWORK SURVEILLANCE SERVER
Home
Communication
User Accounts
Advance Options
Notfication Center
Access Control
Privillage Level
Services
Statistics
Packet Monitor
Iptables Rules
Admin password
Add a custom rule:
Rule description:
Output :
Add

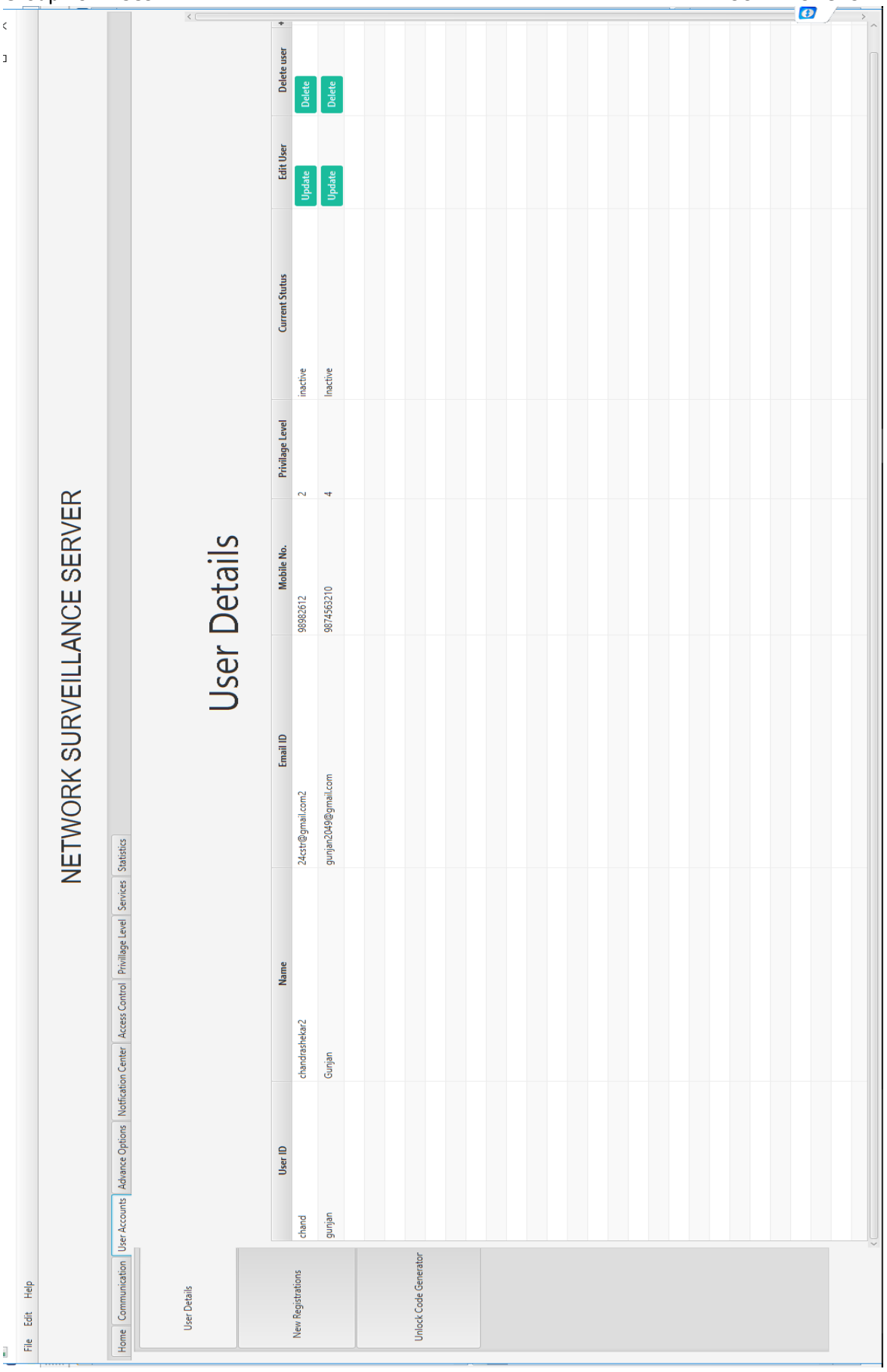

NETWORK SURVEILLANCE SERVER
File Edit Help
Home Communication User Accounts Advance Options Notification Center Access Control Privillage Level Services Statistics
Packet Monitor
Iptables Rules
Admin password
Start
Stop
See Whole Old Data
Information Dialog
Opening Captured Data
It might take some time to load fully depending on the capture size.....
OK
Packet no. Time Packet Length Protocol Souce Port Destination MAC Destination IP
4 2016-04-29 107 Transmission Control Protocol (51100 C4:6E:1F:9E:00:C0 217.146.12.4
5 2016-04-29 60 Transmission Control Protocol (5938 54:A0:50:80:44:C0 192.168.0.104
6 2016-04-29 103 Transmission Control Protocol (5938 54:A0:50:80:44:C0 192.168.0.104
7 2016-04-29 186 Transmission Control Protocol (51100 C4:6E:1F:9E:00:C0 217.146.12.4
8 2016-04-29 698 Transmission Control Protocol (51100 C4:6E:1F:9E:00:C0 217.146.12.4
9 2016-04-29 1167 Transmission Control Protocol (51100 C4:6E:1F:9E:00:C0 217.146.12.4
10 2016-04-29 142 Transmission Control Protocol (TCP) 51100 C4:6E:1F:9E:00:C0 217.146.12.4
11 2016-04-29 115 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
12 2016-04-29 113 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
13 2016-04-29 145 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
14 2016-04-29 720 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
15 2016-04-29 109 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
16 2016-04-29 60 Transmission Control Protocol (TCP) C4:6E:1F:9E:00:C0 217.146.12.4 5938 54:A0:50:80:44:C0 192.168.0.104
17 2016-04-29 60 Transmission Control Protocol (TCP) C4:6E:1F:9E:00:C0 217.146.12.4 5938 54:A0:50:80:44:C0 192.168.0.104
18 2016-04-29 60 Transmission Control Protocol (TCP) C4:6E:1F:9E:00:C0 217.146.12.4 5938 54:A0:50:80:44:C0 192.168.0.104
19 2016-04-29 220 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
20 2016-04-29 107 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
21 2016-04-29 60 Transmission Control Protocol (TCP) C4:6E:1F:9E:00:C0 217.146.12.4 5938 54:A0:50:80:44:C0 192.168.0.104
22 2016-04-29 78 Transmission Control Protocol (TCP) C4:6E:1F:9E:00:C0 217.146.12.4 5938 54:A0:50:80:44:C0 192.168.0.104
23 2016-04-29 103 Transmission Control Protocol (TCP) C4:6E:1F:9E:00:C0 217.146.12.4 5938 54:A0:50:80:44:C0 192.168.0.104
24 2016-04-29 54 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
25 2016-04-29 103 Transmission Control Protocol (TCP) C4:6E:1F:9E:00:C0 217.146.12.4 5938 54:A0:50:80:44:C0 192.168.0.104
26 2016-04-29 103 Transmission Control Protocol (TCP) C4:6E:1F:9E:00:C0 217.146.12.4 5938 54:A0:50:80:44:C0 192.168.0.104
27 2016-04-29 54 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
28 2016-04-29 1464 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
29 2016-04-29 56 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
30 2016-04-29 400 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
31 2016-04-29 1464 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
32 2016-04-29 56 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
33 2016-04-29 1464 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4

NETWORK SURVEILLANCE SERVER
File Edit Help
Home Communication User Accounts Advance Options Notification Center Access Control Privillage Level Services Statistics
Packet Monitor
Iptables Rules
Admin password
Start
Stop
See Whole Old Data
Packet no. Time Packet Length Protocol Source MAC Source IP Souce Port Destination MAC Destination IP
33 2016-04-29 1464 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
34 2016-04-29 56 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
35 2016-04-29 1464 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
36 2016-04-29 56 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
37 2016-04-29 393 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
38 2016-04-29 1464 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
39 2016-04-29 56 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
40 2016-04-29 1464 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
41 2016-04-29 56 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
42 2016-04-29 60 Transmission Control Protocol (TCP) C4:6E:1F:9E:00:C0 217.146.12.4 5938 54:A0:50:80:44:C0 192.168.0.104
43 2016-04-29 1464 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
44 2016-04-29 1464 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
45 2016-04-29 60 Transmission Control Protocol (TCP) C4:6E:1F:9E:00:C0 217.146.12.4 5938 54:A0:50:80:44:C0 192.168.0.104
46 2016-04-29 1464 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
47 2016-04-29 60 Transmission Control Protocol (TCP) C4:6E:1F:9E:00:C0 217.146.12.4 5938 54:A0:50:80:44:C0 192.168.0.104
48 2016-04-29 1464 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
49 2016-04-29 850 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
50 2016-04-29 60 Transmission Control Protocol (TCP) C4:6E:1F:9E:00:C0 217.146.12.4 5938 54:A0:50:80:44:C0 192.168.0.104
51 2016-04-29 1464 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
52 2016-04-29 60 Transmission Control Protocol (TCP) C4:6E:1F:9E:00:C0 217.146.12.4 5938 54:A0:50:80:44:C0 192.168.0.104
53 2016-04-29 60 Transmission Control Protocol (TCP) C4:6E:1F:9E:00:C0 217.146.12.4 5938 54:A0:50:80:44:C0 192.168.0.104
54 2016-04-29 1464 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
55 2016-04-29 60 Transmission Control Protocol (TCP) C4:6E:1F:9E:00:C0 217.146.12.4 5938 54:A0:50:80:44:C0 192.168.0.104
56 2016-04-29 1464 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
57 2016-04-29 66 Transmission Control Protocol (TCP) C4:6E:1F:9E:00:C0 217.146.12.4 5938 54:A0:50:80:44:C0 192.168.0.104
58 2016-04-29 1464 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
59 2016-04-29 66 Transmission Control Protocol (TCP) C4:6E:1F:9E:00:C0 217.146.12.4 5938 54:A0:50:80:44:C0 192.168.0.104
60 2016-04-29 1464 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4
61 2016-04-29 66 Transmission Control Protocol (TCP) C4:6E:1F:9E:00:C0 217.146.12.4 5938 54:A0:50:80:44:C0 192.168.0.104
62 2016-04-29 1464 Transmission Control Protocol (TCP) 54:A0:50:80:44:C0 192.168.0.104 51100 C4:6E:1F:9E:00:C0 217.146.12.4

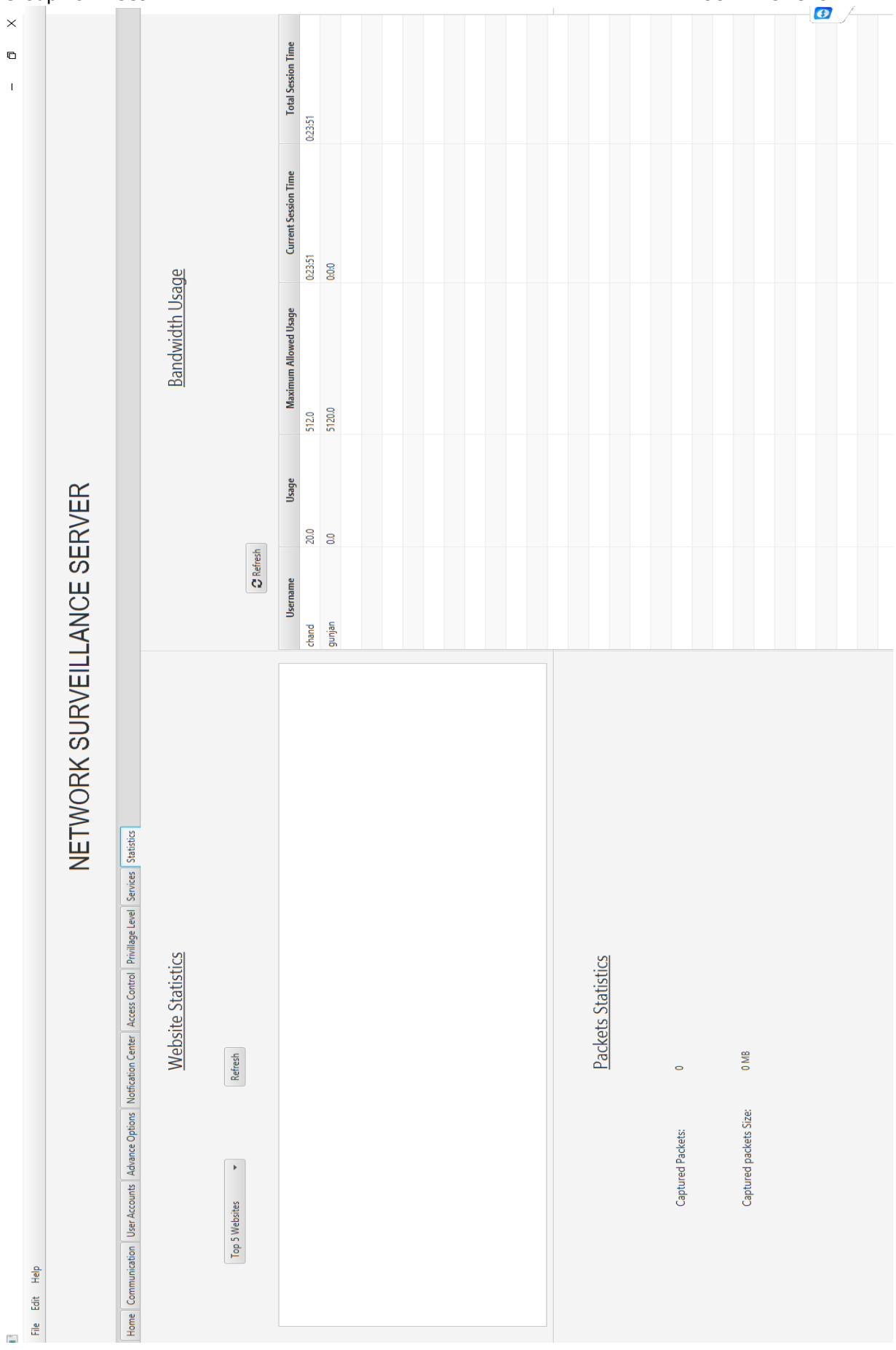

Home
Communication
User Accounts
Advance Options
Notfication Center
Access Control
Privillage Level
Services
Statistics
User Details
New Registrations
Unlock Code Generator
Unlock Code Generator
Choose User ID:
Unlock Code:
Unlock Code Will Be Shown Here
Generate Unlock Code


File Edit Help
NETWORK SURVEILLANCE SERVER
Home
Communication
User Accounts
Advance Options
Notfication Center
Access Control
Privillage Level
Services
Statistics
User Details
New Registrations
Unlock Code Generator
User Registration Requests
Name:
User ID:
Email ID:
Mobile number :
Privillage Level :
Enter Privilage Level 1 - 4
Validate
Open New Requests
Reject


NETWORK SURVEILLANCE SERVER
File Edit Help
Home Communication User Accounts Advance Options Notification Center Access Control Privillage Level Services Statistics
User Details
New Registrations
Unlock Code Generator
User ID
Name
Privilage Level
Current Stutus
Edit User
Delete user
chand
chandrashekar2
inactive
Update
Delete
gunjan
Gunjan
Inactive
Update
Delete
Update User Details
Userid :
chand
Name :
chandrashekar2
Email id :
24cstr@gmail.com2
Mobile Number :
98982612
Privillage Level :
2
Update
Cancel


NETWORK SURVEILLANCE SERVER
File Edit Help
Home Communication User Accounts Advance Options Notification Center Access Control Privillage Level Services Statistics
User Details
New Registrations
Unlock Code Generator
User Details
User ID Name Email ID Mobile No. Privilage Level Current Stutus Edit User Delete user
chand chandrashekar2 24cstr@gmail.com2 98982612 2 inactive Update Delete
gunjan Gunjan gunjan2049@gmail.com 9874563210 4 Inactive Update Delete


NETWORK SURVEILLANCE SERVER
File Edit Help
Home Communication User Accounts Advance Options Notification Center Access Control Privillage Level Services Statistics
Incoming Message:
Message:
Broadcast Message
Send
Message:
Multicast Message
To:
Send
Message:
Unicast message
To:
Send
Message:


NETWORK SURVEILLANCE SERVER
File Edit Help
Home Communication User Accounts Advance Options Notification Center Access Control Privillage Level Services Statistics
Website Statistics
Top 5 Websites
Refresh
Bandwidth Usage
Refresh
Username Usage Maximum Allowed Usage Current Session Time Total Session Time
chand 20.0 512.0 0:23:51 0:23:51
gunjan 0.0 5120.0 0:0:0
Packets Statistics
Captured Packets: 0
Captured packets Size: 0 MB

CHAPTER

9

LIMITATIONS AND FUTURE ENHANCEMENT

CHAPTER 9

LIMITATION AND FUTURE ENHANCEMENT

9.1 LIMITATIONS

Though we tried our best in developing this application limitation are bound to occur. The limitations of our application are:

- Since we are using raspberry pi as server with a 54 Mbps lan port there will be some network congestion while more than one file transfer is taking place there by limiting the whole network speed.
- The application has a lot of GUI components so it takes some time to load on raspberry pi.
- At high usage the program will be less responsive to user action and can appear like it has stopped working for a while.

9.2 FUTURE ENHANCEMENT

The future enhancement for our project is vast. This application can be extended for providing a very large array of functionalities for a network admin. Some of the proposed enhancements are: -

- Snort can be added in background with this application to provide Intrusion Detection System.
- An extra Lan port can be added to raspberry pi using a usb to lan device for controlling network congestion.
- To overcome network congestion, network congestion control can be implemented.
- With gaining popularity of IPv6 it is feasible to migrate from IPv4 to IPv6.
- This program can also be expanded to incorporate automated vulnerability discovery.
- For increasing the security, a ticketing authority can be implemented.
- Network File Discovery functionality can also be implemented.
- A solution for blocking proxy services in windows so that client cannot access blocked websites can be found.
- A dedicated OS for network monitoring based on Linux kernel can be created.

CHAPTER

10

CONCLUSION AND DISCUSSION

CHAPTER 10

CONCLUSION AND DISCUSSION

10.1 CONCLUSION:

- We have successfully deployed our application on a raspberry pi 2. The application is working as expected and is pretty stable in small size networks.
- The Access control, packet monitoring, bandwidth monitoring and all other modules are giving expected outputs.
- The application will run on any Linux based OS with installation of libpcap and on any hardware which can support java with javafx 8.

10.2 Bibliography:

The websites used for reference purposes were mostly:

[1] http://qcktech.blogspot.in/2012/08/raspberry-pi-as-router.html
[2] https://code.google.com/archive/p/iotsys/wikis/groupCommunication.wiki
[3] http://www.rpiblog.com/2014/03/installing-oracle-jdk-8-on-raspberry-pi.html
[4] http://docs.gluonhq.com/javafxports/#_setting_up
[5] https://www.raspberrypi.org/forums/viewtopic.php?f=63&t=35826
[6] http://stackoverflow.com/questions/17830333/start-raspberry-pi-without-login
[7] http://elinux.org/R-Pi_Troubleshooting#
[8] http://www.tutorialspoint.com/junit/junit_overview.htm
[9] https://forum.pfsense.org/index.php
[10] http://www.developer.com/java/ent/article.php/1356891/A-PatternFramework-for-ClientServer-Programming-in-Java.htm
[11] http://stackoverflow.com/questions/2627706/how-to-icmps-and-traceroutes-in-java

APPENDIX

A

COMPLETION CERTIFICATE

APPENDIX

B

PERIODIC PROGRESS REPORT

APPENDIX

C

PATENT DRAFTING EXECERCISE

APPENDIX

D

PLAGARISM REPORT