



A Project Report on
TAPAL

Submitted by

Mansi Deep Dandiwala (1280010)
Kshitij Manish Shah (1280047)
Ankit Purvesh Shah (1280051)

Guided By

Mrs. Swapna A. Naik

Term

December'14 to April'15

Department of Computer Engineering

SVKM's

Shri Bhagubhai Mafatlal Polytechnic,
Irla, N. R. G. Marg, Vile Parle (W), Mumbai -400005



A Project Report on

TAPAL

Submitted by

Mansi Deep Dandiwala (1280010)

Kshitij Manish Shah (1280047)

Ankit Purvesh Shah (1280051)

Guided By

Mrs. Swapna A. Naik

Term

December'14 to April'15

Department of Computer Engineering

SVKM's

Shri Bhaghubhai Mafatlal Polytechnic,
Irla, N. R. G. Marg, Vile Parle (W), Mumbai -400056

Tapact

Throwback to the olden time letters, the modern way...

I Preface

The curriculum of Computer Science Engineering at Shri Bhagubhai Mafatlal Polytechnic extends to us, students an opportunity to change our ideas into realities. All the knowledge and information gathered is expected to be put to use, and a fruitful outcome is to be generated. This Final Year Project, has given us the chance to generate a system that shall be put to use within an organization raising their means of communication.

Through this project we have set up an email system over a Linux machine, we regard as our server and the rest user computers as its clients (with respect to implementation in an organization). Our system is well suited for Google Chrome Browser, also the Safari Browser. Various components have been used in conjunction with one another to achieve the same. We have been cognizant to many new terms, concepts and implementations throughout the project tenure. Well known Mail Transfer Agent, Mail Delivery Agent and Mail User Agent are put together with security provided by SASL Authentication. Perceptive of our system, TAPAL, a combination of the olden Indian mail times with today's modern day email functionalities. Also a Manager Application for the administrator has been designed, allowing him to work effortlessly with multi-instance management, user management and service handling.

This document shall guide you through our efforts, study, and implementation along with aids to set up a similar server. This experience shall prove to be helpful in our further walks of life. We regard it as a great start to our practical outlook and approach.

II Acknowledgement

Not every time can a task be done by oneself, the success is always a share for all those who've helped knowingly or unknowingly.

We would like to express our gratitude to each and every individual who lead us to the right path and backed us with their support, love, care and advice. This project wouldn't have been possible without their motivation and expert knowledge. Not only did they help us technically or on an academic base but also groomed us to become better individuals. They have imbibed in us the feelings and the roots of team work, negotiation, communication, patience and teachings. We thank you'll.

A good teacher can inspire hope, ignite the imagination and instil a love of learning.

We are grateful to our guide, Mrs Swapna Naik for guiding us throughout this project tenure and taking immense efforts in making this brain child possible. She has been a constant support and has directed us to the right decisions. We are grateful to our Head of Department, Mr Janardan Kulkarni, for giving us an opportunity to prove our calibre and also for taking loads of efforts in helping us know and work better with Linux. We are thankful for his guidance throughout. Further, we would like to thank our technician, Mr Sachin Talekar, for being very patient and providing us with all the necessary technical support. Last but not the least, we would love to thank Mr Manish Solanki for helping us with concepts of Java.

The roots of education are bitter, but the fruit is sweet!

We would like to extend our honest gratitude towards our dean, Mr Y. I. Shah, for giving us his support and guidance in abundance.

A positive criticism never did any harm, it just helped us grow better. We thank them for their support and appreciation.

Talent wins games, but teamwork and intelligence win championships.

Table of Contents

I Preface	IV
II Acknowledgement	V
Table of Contents	VI
Table of Figures	VIII
1 Introduction	1
1.1 Types of Server	1
1.1.1 File Server	1
1.1.2 Proxy Server	1
1.1.3 Web Server	2
1.1.4 Home Server	2
1.1.5 Mail Server	2
1.2 Approaches for implementing Mail Server.....	2
1.3 Applications	3
1.4 Our Selection	3
1.5 Scope of the Project	3
2 Literature Survey	4
2.1 Jordi Guitart Fernandez, “Performance Improvement of Multithreaded Java Application Execution on Multi-Processor Systems”, IEEE 2005	4
2.2 Jae-Young Kim and James Won-Ki Hong, “Design and Implementation of a Web-based Internet/Intranet Mail Server Management System”, IEEE	4
3 Proposed System	5
3.1 Overview	5
3.2 Content.....	5
3.2.1 Postfix MTA.....	5
3.2.2 Dovecot MDA.....	7
3.2.3 Squirrelmail Proxy Server with web-based e-mail Client	8
3.2.4 Miscellaneous	8
3.3 Requirements	10

3.3.1	Software Requirements.....	10
3.3.2	Hardware Requirements:	10
3.4	System Flow.....	11
3.4.1	Diagram.....	11
3.4.2	Explanation	11
3.5	System Architecture:.....	13
3.5.1	Diagram.....	13
3.5.2	Explanation	13
3.6	Flowchart	15
3.7	Use Case Diagram.....	16
4	Implementation	17
4.1	Timeline	17
4.2	Setup	18
4.2.1	Installing and configuring Postfix.....	18
4.2.2	Installing and configuring dovecot	32
4.2.3	Installing and configuring Squirrelmail.....	36
4.2.4	Postfix Multi-Instance	40
4.3	Coding.....	42
4.3.1	Squirrelmail Modifications	42
4.3.2	TAPAL Instance Manager.....	66
4.3.3	Multithreaded Java Mail.....	138
4.4	Screenshot	142
4.4.1	Squirrelmail.....	142
4.4.2	TAPAL Multi-Instance Manager	148
4.4.3	Multithreaded Java Mail Output	151
5	Future Scope	152
6	Bibliography	153
7	About Us	154
8	Contact us.....	155

Table of Figures

Figure 3.4-1: System Flow Diagram	11
Figure 3.5-1: System Architecture Diagram.....	13
Figure 3.6-1: System Flowchart	15
Figure 3.7-1: Use Case Diagram	16
Table 1: Timeline.....	17
Figure 4.2-1: Open /etc/hostname.....	18
Figure 4.2-2: Add Hostname	18
Figure 4.2-3: Opening /etc/hosts.....	18
Figure 4.2-4: Assigning IP to Hostname	18
Figure 4.2-5: Updating Repository	19
Figure 4.2-6: Installing Postfix	20
Figure 4.2-7: Select Server Type	20
Figure 4.2-8: Assigning FQDN	20
Figure 4.2-9: Postfix installed.....	21
Figure 4.2-10: Reconfiguring Postfix	21
Figure 4.2-11: Select Server Type	22
Figure 4.2-12: Listing of domains	22
Figure 4.2-13: Local network addresses	22
Figure 4.2-14: Mailbox Size limit	22
Figure 4.2-15: Local address extension character	23
Figure 4.2-16: Internet Protocol Selection.....	23
Figure 4.2-17: Open /etc/postfix/main.cf.....	23
Figure 4.2-18: Add Statements	24
Figure 4.2-19: Open Secure Socket Layer.....	24

Figure 4.2-20: Enter pass phrase	24
Figure 4.2-21: OpenSSL RSA	24
Figure 4.2-22: Writing RSA Key.....	25
Figure 4.2-23: Moving Key	25
Figure 4.2-24: Security Setup	25
Figure 4.2-25: Fill Details.....	25
Figure 4.2-26: Challenge Password	25
Figure 4.2-27: Secure Socket Layer	26
Figure 4.2-28: Setting private key	26
Figure 4.2-29: Copy Certificate	26
Figure 4.2-30: Copy Key	26
Figure 4.2-31: Configure Certificate Path	26
Figure 4.2-32: Certificate Configuration	26
Figure 4.2-33: Open /etc/postfix/master.cf/.....	27
Figure 4.2-34: Modifications in master.cf	27
Figure 4.2-35: Install SASL.....	27
Figure 4.2-36: Create Self-signed SSL Certificate	28
Figure 4.2-37: Assigning Hostname to Dovecot	28
Figure 4.2-38: Open /etc/dovecot/conf.d/10-master.conf	28
Figure 4.2-39: Additions in Dovecot configuration file	29
Figure 4.2-40: Open /etc/dovecot/conf.d/10-auth.conf.....	29
Figure 4.2-41: Assigning Authentication	30
Figure 4.2-42: Restart Postfix	30
Figure 4.2-43: Restart Dovecot.....	30
Figure 4.2-44: Checking SMTP Connection	31
Figure 4.2-45: Install IMAP and POP3	32

Figure 4.2-46: Service Installed.....	32
Figure 4.2-47: Open mail.conf.....	32
Figure 4.2-48: Mail Directory Location	33
Figure 4.2-49: Open pop3.conf.....	33
Figure 4.2-50: Modify POP3 UIDL Format	34
Figure 4.2-51: Open ssl.conf	34
Figure 4.2-52: Setting SSL	35
Figure 4.2-53: Restart Dovecot.....	35
Figure 4.2-54: Checking Port 110	35
Figure 4.2-55: Acknowledgement to Connection.....	36
Figure 4.2-56: Create User	36
Figure 4.2-57: Assign Password	36
Figure 4.2-58: Install Squirrelmail	36
Figure 4.2-59: Squirrelmail Installed.....	37
Figure 4.2-60: Configure Apache with Squirrelmail.....	37
Figure 4.2-61: Configuration	37
Figure 4.2-62: Restart Apache Service	38
Figure 4.2-63: Configure Squirrelmail	38
Figure 4.2-64: Configuration Options	38
Figure 4.2-65: Modify Configuration Values	39
Figure 4.4-1: Login Page	142
Figure 4.4-2: Home Page	142
Figure 4.4-3: Compose	143
Figure 4.4-4: Inbox	143
Figure 4.4-5: Address Book.....	144
Figure 4.4-6: Folder	144

Figure 4.4-7: Options.....	145
Figure 4.4-8: Search.....	145
Figure 4.4-9: Help.....	146
Figure 4.4-10: About Us	146
Figure 4.4-11: Mail Read.....	147
Figure 4.4-12: Sign Out	147
Figure 4.4-13: Main Frame.....	148
Figure 4.4-14: Add User.....	148
Figure 4.4-15: Delete User	149
Figure 4.4-16: Manage Instance	149
Figure 4.4-17: Creating Instance	150
Figure 4.4-18: Output on Terminal.....	150
Figure 4.4-19: Terminal Output.....	151
Figure 4.4-20: Mails received using Multithreading	151

1 Introduction

Email is familiar. It's comfortable. It's easy to use.

Email has proved to be the most important milestone in communication, and in turn has taken the world by a swirl. It is an enhanced and multi-capable communication medium whole heartedly accepted over the world. It is now the most basic but at the same time the most essential and on point method of communicating with the masses or officials. Email seeped into each and every field or domain or business area, due to which a great hike in its need and a rise in its capabilities is observed.

Every organization needs a communication system that allows easy transfer and sharing of information and data now and then. Also security is of high concern. A system that allows internetwork communication i.e. within the organization; and also provides an acceptable level of security is highly needed. Keeping the needs of the administrator in mind, a UI for management is a plus point.

Further this document will guide you through the steps followed for setting up or preparing this system over a Linux based computer with images to illustrate the procedure, working of our system and its component to understand the flow, followed by steps to use our system and its provisions. The codes for implementing a GUI based Manager have been mentioned too.

1.1 Types of Server

1.1.1 File Server

A file server is a computer attached to a network that has the primary purpose of providing a location for shared disk access, i.e. shared storage of computer files (such as documents, sound files, photographs, movies, images, databases, etc.) that can be accessed by workstations that are attached to the same computer network. It is designed primarily to enable the storage and retrieval of data while the computation is carried out by the workstations.

They are commonly found in schools and offices, where users use a LAN to connect their client computers. File servers are frequently accessed by File Transfer Protocol (FTP) and Hyper Text Transfer Protocol (HTTP). These differ from web servers.

1.1.2 Proxy Server

A proxy server is a server (a computer system or an application) that acts as an intermediary for requests from clients seeking resources from other servers. A client connects to the proxy server, requesting some service, such as a file, connection, web page, or other

resource available from a different server and the proxy server evaluates the request as a way to simplify and control its complexity.

Proxies were invented to add structure and encapsulation to distributed systems. Today, most proxies are web proxies, facilitating access to content on the World Wide Web and providing anonymity. They can be classified as forward or reverse proxies.

1.1.3 Web Server

A web server is an information technology that processes requests via HTTP, the basic network protocol used to distribute information on the World Wide Web. The term can refer either to the entire computer system, an appliance, or specifically to the software that accepts and supervises the HTTP requests.

1.1.4 Home Server

A home server is a server located in a private residence providing services to other devices inside or outside the household through a home network or the Internet. Such services may include file and printer serving, media centre serving, web serving (on the network or Internet), web caching, account authentication and backup services. Because of the relatively low number of computers on a typical home network, a home server commonly does not require significant computing power and can be implemented with a re-purposed, older computer, or a plug computer.

1.1.5 Mail Server

An email server is a high capacity computing device within your network that works as your virtual post office. A mail server usually consists of a storage area where email is stored for local users, a set of user definable rules to which determine how the mail server should react to the destination of a specific message, a database of user accounts that the mail server recognizes and will deal with locally, and communications modules which are the components that actually handle the transfer of messages to and from other mail servers and email clients.

Generally the person(s) responsible for the maintenance of the e-mail server (editing users, monitoring system activity) are referred to as the postmaster. Most mail servers are designed to operate without any manual intervention during normal operation.

1.2 Approaches for implementing Mail Server

1. Configuration of a full featured mail server using Postfix, Cyrus IMAP, ClamAV, Amavisnew, and Spam Assassin.
2. Configuration of a full features mail server Using James Server which is based on Java Mail API.
3. Configuring with IIS Server on Windows. SMTP is provided by many email service providers like Google via Gmail and by Microsoft via Microsoft's Hotmail/Outlook.

4. Configuring Postfix on Linux. SMTP is provided by many email service providers like Google via Gmail.
5. Setup mail server on Ubuntu using Postfix (for sending), Dovecot (for receiving) and Squirrel mail (for webmail access).

1.3 Applications

1. Intra organizational communication via an email service based on a LAN network.
2. Administrator operated mail service.
3. Authenticated administrator privileged to add and remove users.

1.4 Our Selection

Out of all these approaches, our system is based on the last mentioned approach, i.e. Configuration of a mail server using Postfix, Dovecot and Squirrelmail as MTA, MDA and MUA respectively. This approach has been emphasized upon further in the document.

1.5 Scope of the Project

This project spans over various servers, which were configured by us on a Linux System. These servers proved to be the base for our email system. With the help of Mail Transfer Agent that routes and delivers electronic mail; Internet Message Access Protocol (IMAP) and Post Office Protocol (POP) for email retrieval and storage; and an intermediary proxy server that further provides a web based email client, we have successfully setup an email server that is capable of sending and receiving text and attachment mails within the organization and/or to whomsoever the LAN is available. Further a GUI-based Management control panel has been implemented so as to allow the administrator to administer the instances running on the server, toggle the services of the server, and create and delete users. A multithreaded mail sending program has been implemented in order obtain maximum performance.

This system is the answer to various organization's privacy. It allows secure communication within its span, because of which all valid users, as stated by the administrator, shall be given the permission and are granted with the ease of communication amongst themselves. Easy transfer of text and data files of up to 2 MB is provided. The user creation is a task left upon the administrator, hence providing only a single trusted body the authentication to create and remove users. This system shall prove to be a simple yet an effective implementation.

2 Literature Survey

2.1 Jordi Guitart Fernandez, “Performance Improvement of Multithreaded Java Application Execution on Multi-Processor Systems”, IEEE 2005

In this paper, emphasis has been made over understanding the behaviour of multithreaded applications written mainly in Java. Helping to realize how an application behaves when executed on environments that include server and virtual machines. This thesis contributes with scheduling mechanisms and policies oriented towards the efficient execution of multithreaded Java applications on multiprocessor systems considering the interaction and co-ordination between scheduling mechanisms and policies at the different levels involved in the execution.

2.2 Jae-Young Kim and James Won-Ki Hong, “Design and Implementation of a Web-based Internet/Intranet Mail Server Management System”, IEEE

In this paper, a system of an Intranet/Internet Mail Server management system has been implemented using three architectural parts: SNMP Agent, SNMP Manager, and Java-based management application. The design and implementation of the system were looked upon. SNMP MIB module for monitoring mail server process mentioned and have design SNMP manager and agent system which can handle SNMP MIB information included. Web technology such as CGI and Java that enables human users to learn and maintain management applications easily and cost-effectively is utilized hence achieving highly efficiency. Though they have developed a management system for mail service only, our design and implementation architecture is general enough to be applied to manage any Internet/Intranet application service system.

3 Proposed System

3.1 Overview

Our project, ‘TAPAL’ provides a system that blends the modern day email facilities with an Indian touch for a homely environment. Our main aim is to provide organizations with their own TAPAL system for communicating amongst the employees, workers and whosoever concerned. It allows access to users validated only and only by the server. We can say that user creation is looked after only by the administrator operating the server. The server ought to be a Linux based platform. All user files are stored upon the server in the form of mail directories.

A lot of effort and thought has been linked with this project, not only did it open new windows for learning but aided us in understanding the need for project making and relationship building, i.e. soft skills.

Our thought engraved with this project is,

Throwback to the olden time letters, the modern way...

3.2 Content

3.2.1 Postfix MTA

Postfix is a free and open-source mail transfer agent (MTA) that routes and delivers electronic mail. It was released under the IBM Public License 1.0 which is a free software licence. As an SMTP client, Postfix implements a high performance parallelized mail delivery engine. Postfix is often combined with mailing list software.

Operating Systems

Postfix runs on AIX, BSD, HP-UX, GNU/Linux, OS X, and Solaris and on every Unix-like operating system that ships with a C compiler and delivers a standard POSIX development environment. It is default MTA for the OS X and Ubuntu operating systems.

Architecture

Postfix consist of a combination of server programs that run in the background, and client programs that are invoked by user programs or by system administrators. The Postfix core consists of several dozen server programs that run in the background each handling one specific aspect of email deliver. Examples are the SMTP Server, the scheduler, the address rewriter, and the local delivery server. For damage-control purposes, most server programs run with fixed reduced privileges, and terminate voluntarily after processing a limited number of

requests. To conserve system resources, most server programs terminate when they become idle.

Client programs run outside the Postfix core. They interact with Postfix server programs through mail delivery instructions in the user's `~/.forward` file, and through small "gate" programs to submit mail or to request queue status information.

Other programs provide administrative support to start or stop Postfix, query status information, manipulate the queue, or to examine or update its configuration files.

Implementation

The Postfix implementation uses safe subsets of the C language and of the POSIX system API. These subsets are buried under an abstraction layer that contains about 50% of all Postfix source code, and that provides the foundation on which all Postfix programs are built. For example, the "string" primitive makes Postfix code resistant to buffer overflow attacks, and the "safe open" primitive makes Postfix code resistant to race condition attacks on systems that implement the POSIX file system API.

This abstraction layer does not affect the attack resistance of non-Postfix code, such as code in system libraries or in third-party libraries.

Robustness

Conceptually, Postfix manages pipelines of processes that pass the responsibility for message delivery and error notification from one process to the next. All messages and notification "state" information is persisted in the file system. The processes in a pipeline operate mostly without centralized control, this relative autonomy simplifies error recovery. When a process fails before completing its part of a file or protocol transaction, its predecessor in the pipeline backs off and retries the request later, and its successor in the pipeline discards unfinished work. Many Postfix daemons can simply "die" when they run into a problem; they are automatically restarted when the next service request arrives. This approach makes Postfix highly resilient, as long as the operating system or hardware don't fail catastrophically.

Multi-Instance

Postfix provides a multi-instance functionality. With multiple Postfix instances, each instance has its own directories for configuration, queue and data files, but it shares all Postfix program and documentation files with other instances. The purpose of multi-instance support in Postfix is not to create multiple Postfix instances, but rather to give tasks a choice. Multiple instances give users the freedom to tune each Postfix instance to a single task that it does well and to combine instances into complete systems. Introducing multi-instance functionality increases server performance and thus reduces the per-instance configuration footprint of a secondary Postfix instance to just a `main.cf` and `master.cf` file.

3.2.2 Dovecot MDA

Dovecot is an open source IMAP and POP3 email server for Linux/ UNIX- like systems, written with security primarily in mind. It is a choice suitable for both small and large implementations. The mail boxes are transparently indexed which are self-optimizing. These indexes can be modified by multiple computers at the same time. It aims to be lightweight, fast and easy to setup open source mail server.

Implementation and Constituents

Each mail account served by Dovecot, has a local user account defined on the server. The server authenticates the user against the local user database. SSL is used to encrypt the authentication passwords. The dovecot package consists a script to generate the server SSL certificate. A .sh file is to be executed in order to achieve the same. Dovecot is a service that can be started, stopped or restarted.

POP

The Post Office Protocol is an application-layer internet standard protocol used by local email clients to retrieve email from a server over a TCP/IP connection. The current standard is POP version 3. It supports simple download-and-delete requirements for access to remote mailboxes.

Although an option does exist for the clients to leave mail on server after download, email clients using POP generally connect, retrieve all messages, store them on the user's PC as new messages, delete them from the server and then disconnect. It listens on well-known port 110.

IMAP

Internet Message Access Protocol is a protocol for email retrieval and storage developed by Mark Crispin in 1986 at Stanford University as an alternative to POP. It specifically allows multiple clients simultaneously connected to the same mailbox, and through the flags stored on the server, different clients accessing the same mailbox at the same or different times can detect state changes made by other clients. STARTTLS can be used to provide secure communication in between the MUA communicating with the MSA or MTA implementing SMTP protocol.

Incoming email messages are sent to an email server that stores messages in the recipient's email box. The user retrieves the messages with an email client that uses one of a number of email retrieval protocols. Most clients and servers support SMTP for sending email and POP and IMAP for retrieving email, allowing interoperability with other servers and clients.

3.2.3 Squirrelmail Proxy Server with web-based e-mail Client

Squirrelmail is a project that provides both a web-based email client and a proxy server for the IMAP protocol. The webmail portion of the project was started by Nathan and Luke Ehresman in 1999 and is written in PHP. The web server needs access to the IMAP server hosting the email and to an SMTP server to be able to send mails. The Proxy server product was created by Dave McMurtrie in 2002 while at the University of Pittsburgh; adopted by Squirrelmail in 2010. It is written in C and is primarily made to provide stateful connections for stateless webmail client software to an IMAP server, thus avoiding the new IMAP logins for every client action and in some cases significantly improving webmail performances.

Squirrelmail makes use of a plugin architecture to accommodate additional features around the core application, and over 200 plugins are available on the Squirrelmail website.

3.2.4 Miscellaneous

PHP

PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. It stands for Hypertext Pre-processor, which is a recursive backronym. PHP code can be simply mixed with HTML code, or it can be used in combinations with various template engines and web frameworks. It is usually processed by a PHP interpreter, implemented as a web server's native module or a Common Gateway Interface (CGI) executable.

After interpretation and execution, the web server sends resulting outputs to its client, usually in a form of a part of a generated web page.

Implementations of PHP are enlisted below:

- Hip Hop Virtual Machine or HHVM
- Parrot, a Virtual Machine to run dynamic languages
- Phalange, compiles PHP into Common Intermediate Language Byte code

HTML

Hypertext Markup Language, commonly referred to as HTML, is the standard markup language used to create web pages. It is written in the form of HTML elements consisting of tags enclosed in angel brackets.

Web browsers can read HTML files and compose them into visible or audible web pages. Browsers do not display the HTML tags and scripts, but use them to interpret the content of the page. HTML describes the structure of a website semantically along with cues for presentation, making it a markup language, rather than a programming language.

HTML elements form the building blocks of all websites. HTML allows images and objects to be embedded and can be used to create interactive forms. It provides a means to create structured documents by denoting structural semantic for text such as headings, paragraphs, links, lists, quotes and other items. It can be embedded scripts written in language such as JavaScript which affect the behaviour of HTML web pages.

Web browsers can also refer to Cascading Style Sheets (CSS) to define the look and layout of text and other material. The World Wide Web Consortium (W3C), maintainer of both the HTML and the CSS standards, encourages the use of CSS over explicit presentational HTML.

CSS

Cascading Style Sheets (CSS) is a style sheet language used for describing the look and formatting of a document written in a markup language. While most often used to change the style of web pages and user interfaces written in HTML and XHTML, the language can be applied to any kind of XML document, including plain XML, SVG and XUL. Along with HTML and JavaScript, CSS is a cornerstone technology used by most websites to create visually engaging webpages, user interfaces for web applications, and user interfaces for many mobile applications.

CSS is designed primarily to enable the separation of document content from document presentation, including elements such as the layout, colours, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple HTML pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content

Linux

Linux is a Unix-like and mostly POSIX-compliant computer operating system assembled under the model of free and open-source software development and distribution. The defining component of Linux is the Linux kernel, an operating system kernel first released on 5 October 1991 by Linus Torvalds. Linux was originally developed as a free operating system for Intel x86-based personal computers. It is the leading operating system on servers and other big iron systems such as mainframe computers and supercomputers.

Ubuntu is a Debian-based Linux operating system, with Unity as its default desktop environment. It is based on free software. A default installation of Ubuntu contains a wide range of software that includes LibreOffice, Firefox, Thunderbird, Transmission, and several lightweight games such as Sudoku and chess. Many additional software packages, including titles no longer in the default installation such as Evolution, GIMP, Pidgin, and Synaptic, are accessible from the built in Ubuntu Software Centre as well as any other APT based package management tool. Execution of Microsoft Office and other Microsoft Windows applications

can be facilitated via the Wine compatibility package or through the use of a virtual machine such as VirtualBox or VMware Workstation.

Java

Java provides a system for developing application software and deploying it in a cross platform computing environment. Also it is a general purpose computer programming language that is concurrent, class-based, object-oriented and specifically designed to have as few implementation dependencies as possible. It is intended to let application developers “write one, run anywhere” (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation. Swing is a graphical user interface library allowing to create a different look from those of Windows, etc.

3.3 Requirements

3.3.1 Software Requirements

Server Side

1. Ubuntu Desktop OS 14.04 LTS
2. Internet Connection
3. Browser (Suits best for Chrome)
4. Administrative/ Root privileges.
5. JDK and JRE

Client Side

1. Browser (Suits best for Chrome)
2. Valid connectivity to LAN via Computer/Smart phones.

3.3.2 Hardware Requirements:

1. Computer System with further given specifications:
2. CPU : 64 bit Possessor, Frequency 1GHz
3. RAM : 512MB
4. Memory Space : 1.75 GB
5. Organization wide LAN Network

3.4 System Flow

3.4.1 Diagram

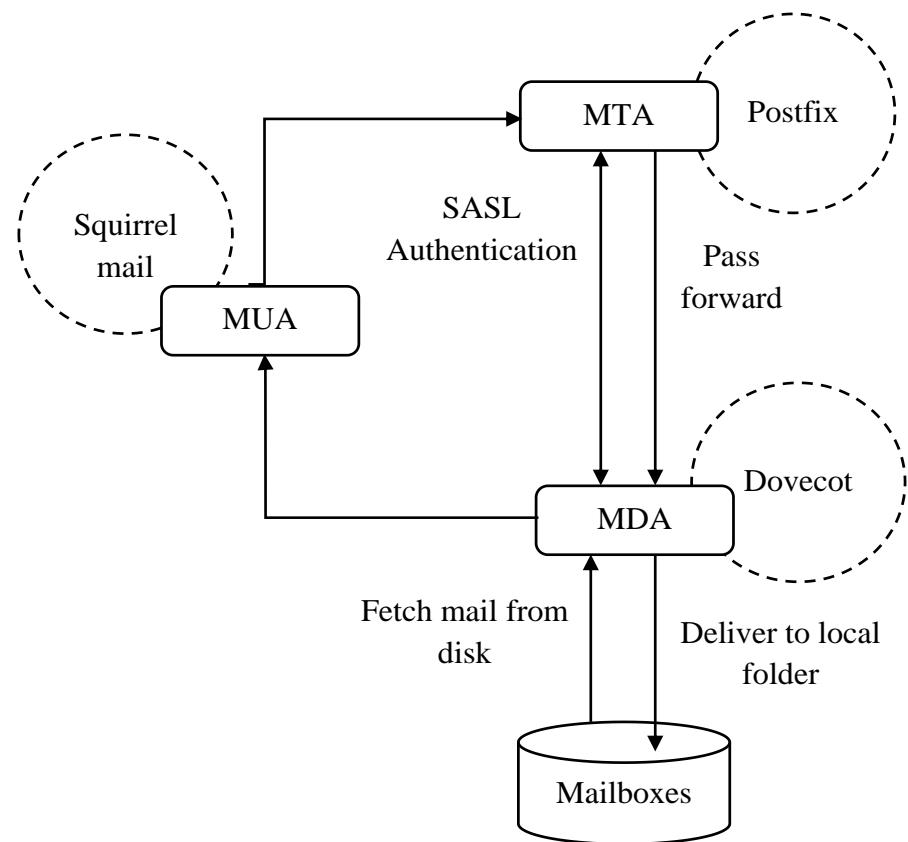


Figure 3.4-1: System Flow Diagram

3.4.2 Explanation

The System flow has been explained in further detail with reference to Figure 3.4-1.

Mail Transfer Agent

A message transfer agent or mail transfer agent (MTA) or mail relay is software that transfers electronic mail messages from one computer another using client-server application architecture. An MTA implements both the client (sending) and server (receiving) portions of the Simple Mail Transfer Protocol (SMTP). Postfix is a free and open-source mail transfer agent (MTA). Postfix routes and delivers electronic mail.

Mail Delivery Agent

A mail delivery agent (MDA) is a computer software component that is responsible for the delivery of e-mail messages to a local recipient's mailbox. Also called an LDA, or local delivery agent. Within the Internet mail architecture, local message delivery is achieved through a process of handling messages from the MTA, and storing mail into the recipient's environment (typically a mailbox). Dovecot is a mail server suite that includes a mail delivery agent. Dovecot is an open source IMAP and POP3 server for LINUX/Unix –like systems, written primarily with security in mind. Dovecot primarily aims to be a lightweight, fast and easy to set up open source mail server.

Mail User Agent

An email client, email reader or more formally mail user agent (MUA), is a computer program used to access and manage a user's email. The term can refer to any system capable of accessing the user's email mailbox. A web application that provides message management, composition, and reception functions widely referred to as web mail. A web mail has the ability to send and receive mail away from the user's normal base using a web browser.

Squirrelmail is an Open Source project that provides a web based email application written in the PHP is present. Squirrelmail webmail outputs valid HTML 4.0 for its presentation, making it compatible with a majority of current web browsers. Squirrelmail webmail uses plugin architecture to accommodate additional features around the core application, and over 200 plugins are available on the Squirrelmail website.

Miscellaneous

The three elements cited above are the basic elements for a complete web e-mail server. However, we have to care also about security. Authentication mechanism is a client/server protocol. It's about how the client and server talk to each other in order to perform the authentication. Most people use only PLAIN authentication, which basically means that the user and password are sent without any kind of encryption to server. SSL/TLS can then be used to provide the encryption to make PLAIN authentication secure. The Simple Authentication and Security Layer (SASL) is a framework for providing authentication and data security services in many connection-oriented protocols. Our implementation of SASL will use the Transport Layer Security (TLS) to encrypt the segments of network connections.

3.5 System Architecture:

3.5.1 Diagram

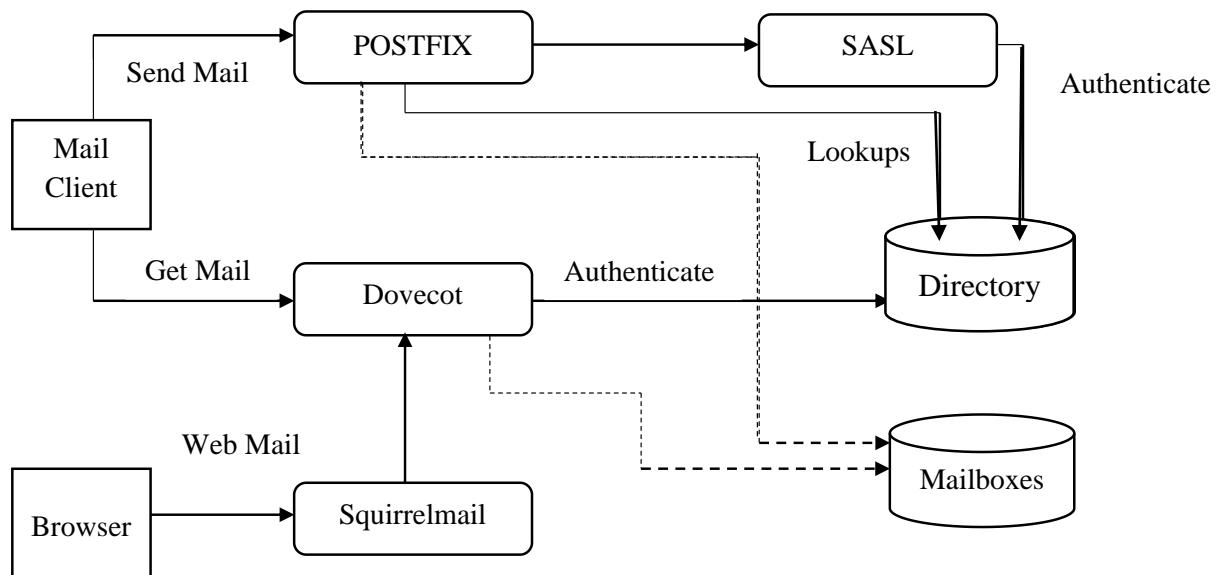


Figure 3.5-1: System Architecture Diagram

3.5.2 Explanation

The System architecture along with its components has been explained in further detail with reference to Figure 3.5-1.

Message Composition

Email clients usually contain user interfaces to display and edit text. Some applications permit the use of a program-external editor. Here we shall implement a web based email client. That provides a text editor or composer within the website under the Compose tab. This space enables the user to draft their emails with various properties that may be altered according to need. Also attachments of up to 2 MB of type document, multimedia or image can be made. Headers are included which consist of the destination fields, To, CC, and Bcc, and the originator fields, From, which is the message's author(s). Sender in case there are more authors, and Reply-To in case responses should be addresses to a different mailbox. To better assist the user with destination fields, our client maintains one or more address books and / or are able to connect to an LDAP directory server.

Client setting require the user's real name and email address for each user's identity, and possibly a list of LDAP servers.

Submitting messages to a server

When a user wishes to create and send an email, the email client will handle the task. The email client is usually set up automatically to connect to the user's mail server, which is typically an MTA. The email client using the SMTP protocol creates an authentication extension, which the mail server uses to authenticate the sender.

Client settings require the name or IP address of the preferred outgoing mail server, the port number (25 for MTA), and the user name and password for the authentication, if any. There is a non-standard port 465 for SSL encrypted SMTP sessions, that many clients and servers support for backward compatibility.

Message Delivery

For recipients hosted locally, the final delivery of email to a recipient mailbox is the task of a Message Delivery Agent (MDA). For this purpose the MTA transfers the message to the message handling service component of the MDA. Upon final delivery, the Return-Path field is added to the envelope to record the return path.

3.6 Flowchart

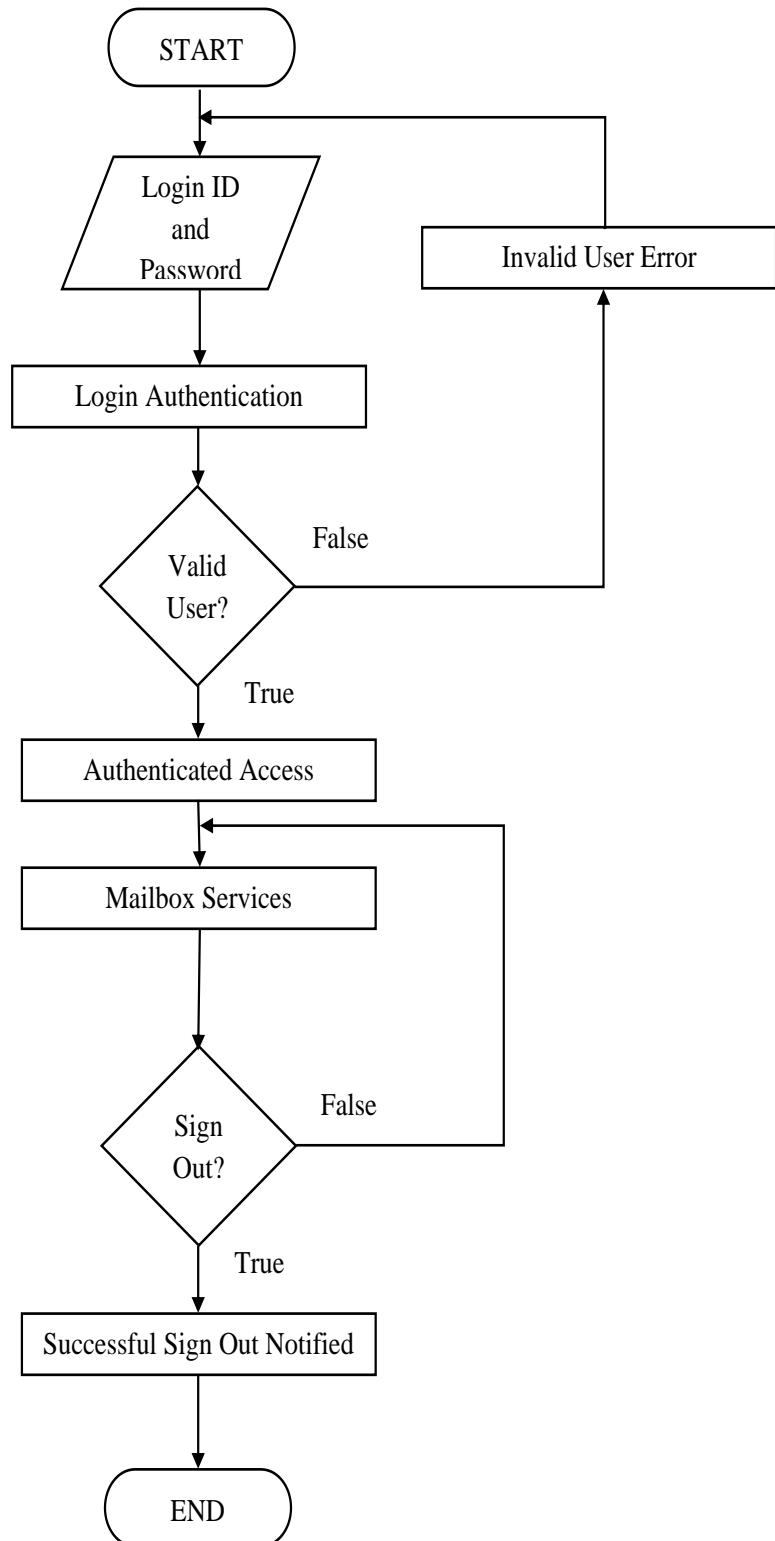


Figure 3.6-1: System Flowchart

3.7 Use Case Diagram

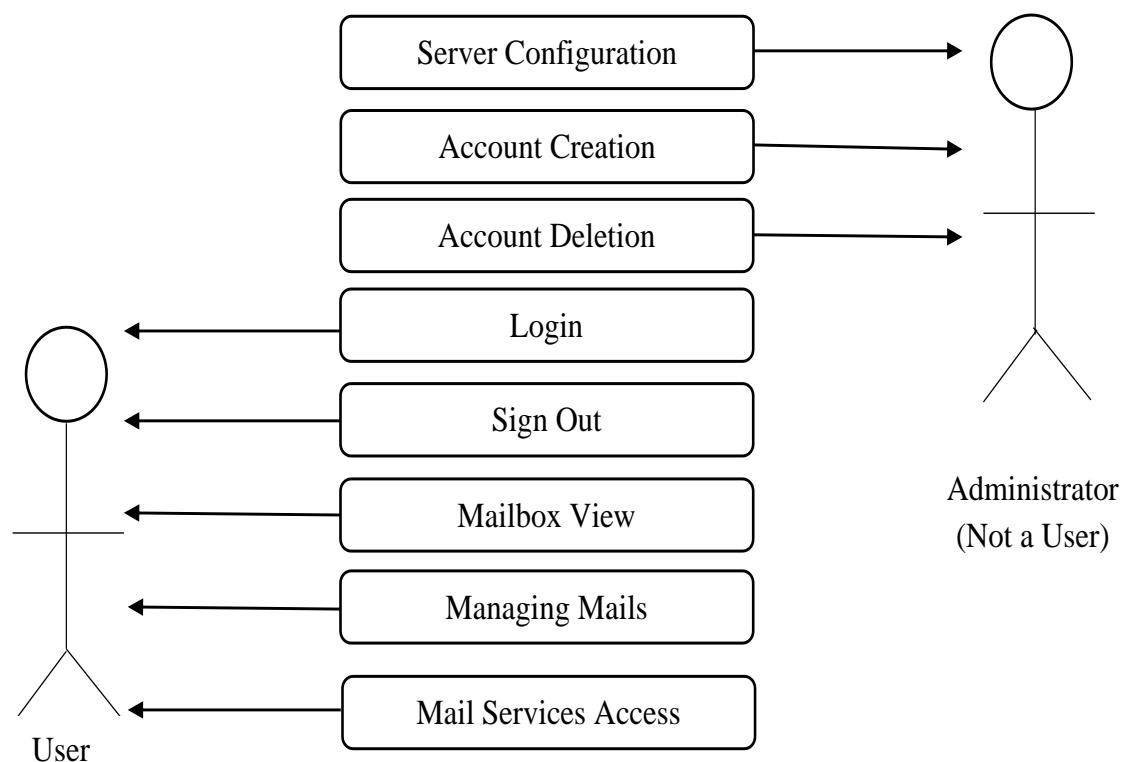


Figure 3.7-1: Use Case Diagram

4 Implementation

4.1 Timeline

Week	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Requirement Analysis																
Information Gathering																
Module Design																
Server Setup																
Coding																
Solving Dependencies																
Testing																
Implementation																

Table 1: Timeline

4.2 Setup

4.2.1 Installing and configuring Postfix

Step 1: Assign static IP and hostname and add a host entry for the host name.

Assign hostname in **/etc/hostname**

```
root@mail:~# nano /etc/hostname
```

Figure 4.2-1: Open /etc/hostname

```
GNU nano 2.2.6                                         File: /etc/hostname

###Tapal Hostname###
mail.tapal.com
```

Figure 4.2-2: Add Hostname

Add a host entry in **/etc/hosts**

```
root@mail:~# nano /etc/hosts
```

Figure 4.2-3: Opening /etc/hosts

```
GNU nano 2.2.6                                         File: /etc/hosts

127.0.0.1      localhost
127.0.1.1      kshitij-PC

###Tapal Server IP Address###
192.168.1.157  mail.tapal.com

# The following lines are desirable for IPv6 capable hosts
::1      ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

Figure 4.2-4: Assigning IP to Hostname

Step 2: Update the repositories.

```
root@mail:~# sudo apt-get update
sudo: unable to resolve host ###Tapal Hostname###
mail.tapal.com
Ign cdrom://Ubuntu 14.04 LTS _Trusty Tahr_ - Release amd64 (20140417) trusty InRelease
Ign cdrom://Ubuntu 14.04 LTS _Trusty Tahr_ - Release amd64 (20140417) trusty/main Translation-en_IN
Ign cdrom://Ubuntu 14.04 LTS _Trusty Tahr_ - Release amd64 (20140417) trusty/main Translation-en
Ign cdrom://Ubuntu 14.04 LTS _Trusty Tahr_ - Release amd64 (20140417) trusty/restricted Translation-en_IN
Ign cdrom://Ubuntu 14.04 LTS _Trusty Tahr_ - Release amd64 (20140417) trusty/restricted Translation-en
Ign http://dl.google.com stable InRelease
Get:1 http://dl.google.com stable Release.gpg [198 B]
Get:2 http://dl.google.com stable Release [1,347 B]
Err http://dl.google.com stable Release

Ign http://extras.ubuntu.com trusty InRelease
Ign http://security.ubuntu.com trusty-security InRelease
Ign http://in.archive.ubuntu.com trusty InRelease
Ign http://ppa.launchpad.net trusty InRelease
Get:3 http://security.ubuntu.com trusty-security Release.gpg [933 B]
Ign http://in.archive.ubuntu.com trusty-updates InRelease
Get:4 http://extras.ubuntu.com trusty Release.gpg [72 B]
Hit http://ppa.launchpad.net trusty Release.gpg
Get:5 http://security.ubuntu.com trusty-security Release [62.0 kB]
Ign http://in.archive.ubuntu.com trusty-backports InRelease
Hit http://extras.ubuntu.com trusty Release
Hit http://in.archive.ubuntu.com trusty Release.gpg
Hit http://ppa.launchpad.net trusty Release
Hit http://extras.ubuntu.com trusty/main Sources
Get:6 http://in.archive.ubuntu.com trusty-updates Release.gpg [933 B]
Hit http://ppa.launchpad.net trusty/main amd64 Packages
Hit http://in.archive.ubuntu.com trusty-backports Release.gpg
Hit http://extras.ubuntu.com trusty/main amd64 Packages
Hit http://ppa.launchpad.net trusty/main i386 Packages
Hit http://in.archive.ubuntu.com trusty Release
Get:7 http://security.ubuntu.com trusty-security/main Sources [72.5 kB]
Hit http://extras.ubuntu.com trusty/main i386 Packages
Get:8 http://in.archive.ubuntu.com trusty-updates Release [62.0 kB]
Get:9 http://security.ubuntu.com trusty-security/restricted Sources [2,061 B]
Get:10 http://security.ubuntu.com trusty-security/universe Sources [17.9 kB]
Hit http://in.archive.ubuntu.com trusty-backports Release
Get:11 http://security.ubuntu.com trusty-security/multiverse Sources [1,905 B]
Hit http://in.archive.ubuntu.com trusty/main Sources
```

Figure 4.2-5: Updating Repository

Step 3: Install postfix and dependencies. Press enter for all prompted questions during installation. We will do that in the next step.

```
root@mail:~# sudo apt-get install postfix
```

Figure 4.2-6: Installing Postfix

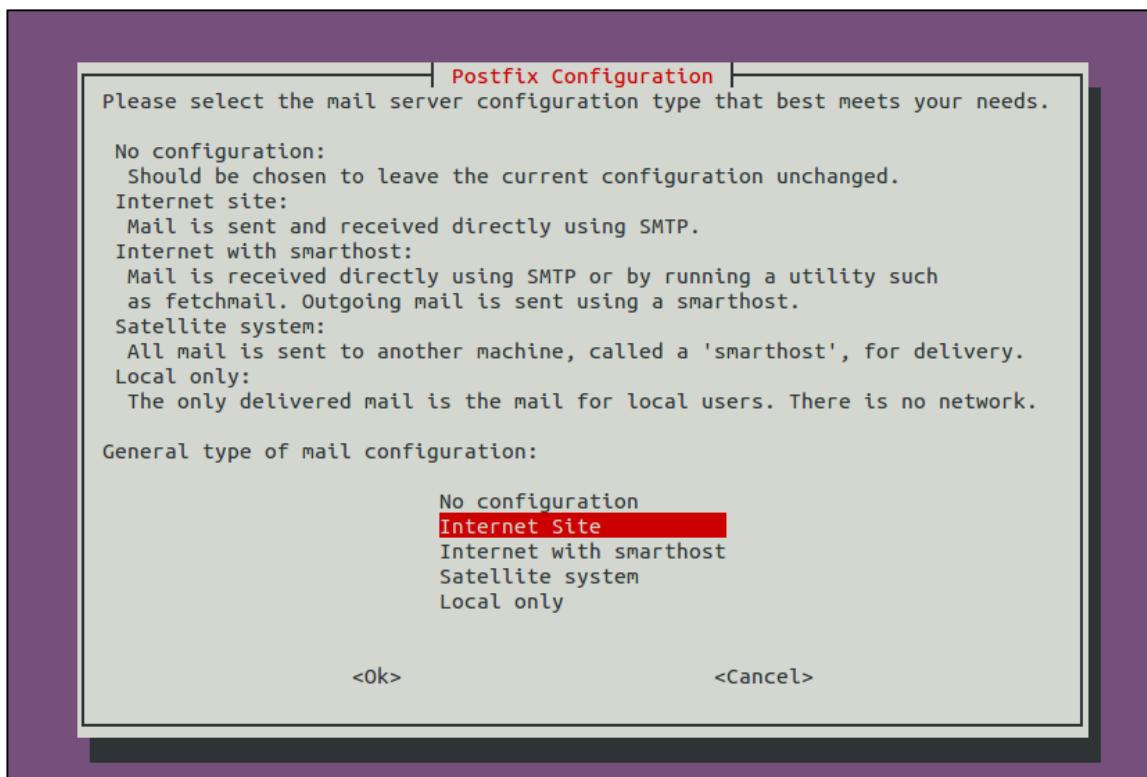


Figure 4.2-7: Select Server Type

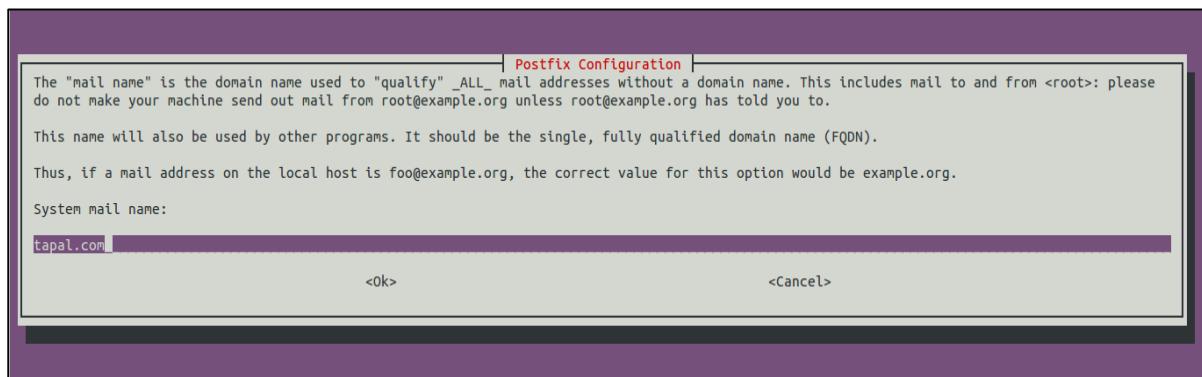


Figure 4.2-8: Assigning FQDN

```
root@mail:~# sudo apt-get install postfix
sudo: unable to resolve host ###Tapal Hostname###
mail.tapal.com
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
  procmail postfix-mysql postfix-pgsql postfix-ldap postfix-pcre sasl2-bin
  dovecot-common postfix-cdb postfix-doc
The following NEW packages will be installed:
  postfix
0 upgraded, 1 newly installed, 0 to remove and 19 not upgraded.
Need to get 0 B/1,084 kB of archives.
After this operation, 3,609 kB of additional disk space will be used.
Preconfiguring packages ...
```

Figure 4.2-9: Postfix installed

Step 4: After installation issue the below command to configure postfix.

```
root@mail:~# sudo dpkg-reconfigure postfix
```

Figure 4.2-10: Reconfiguring Postfix

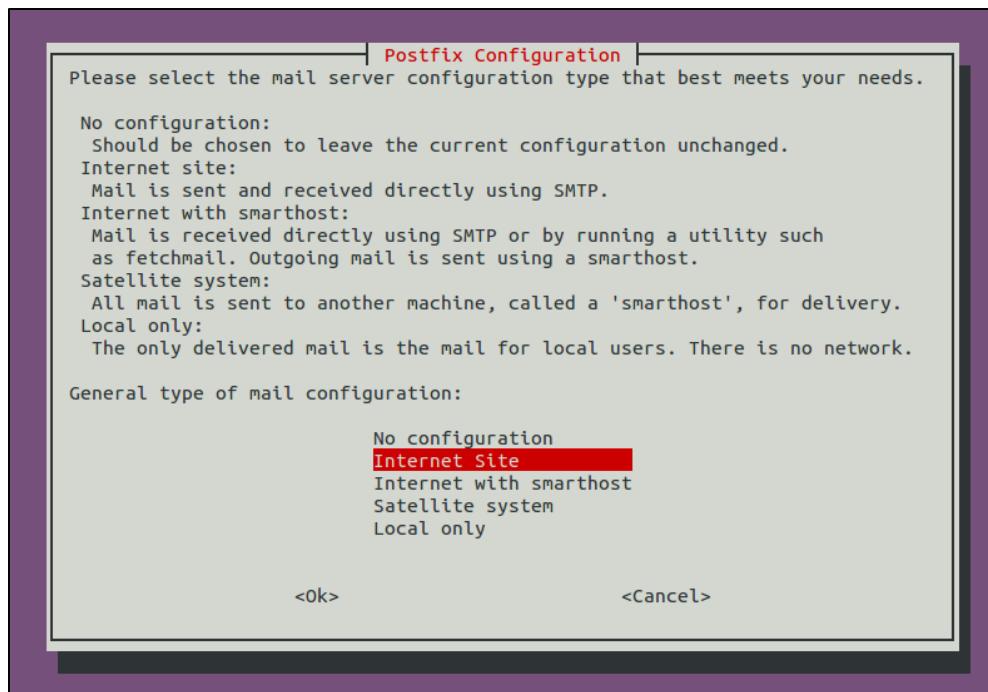
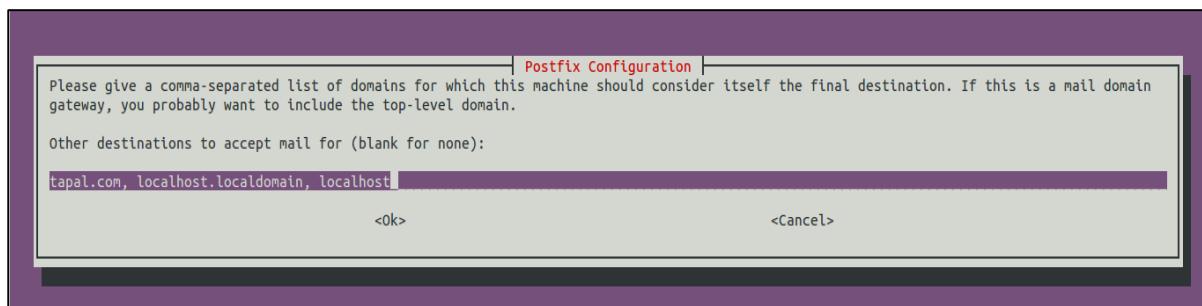
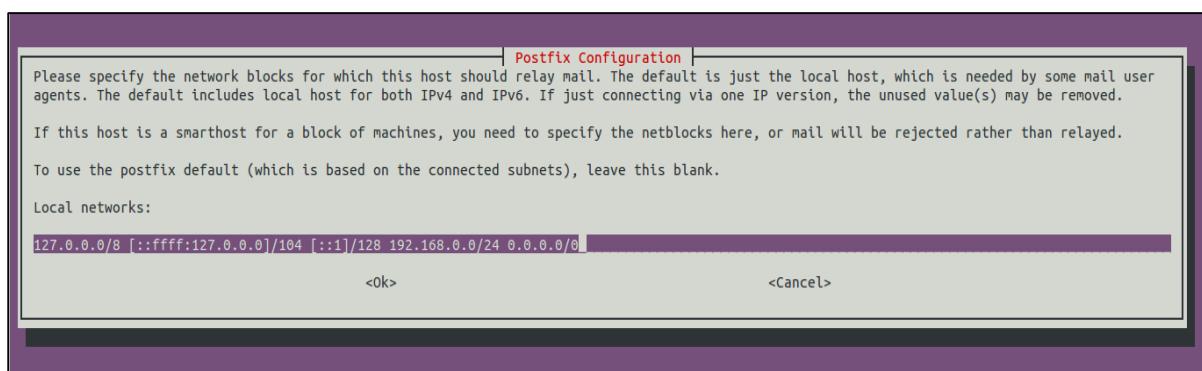
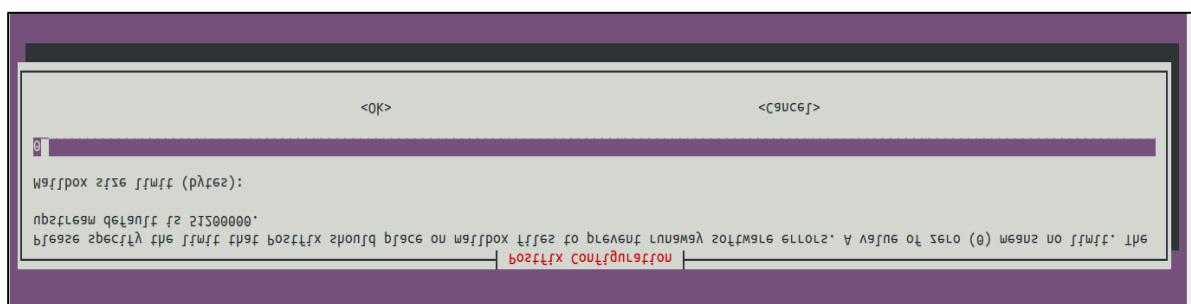


Figure 4.2-11: Select Server Type**Figure 4.2-12: Listing of domains****Figure 4.2-13: Local network addresses****Figure 4.2-14: Mailbox Size limit**

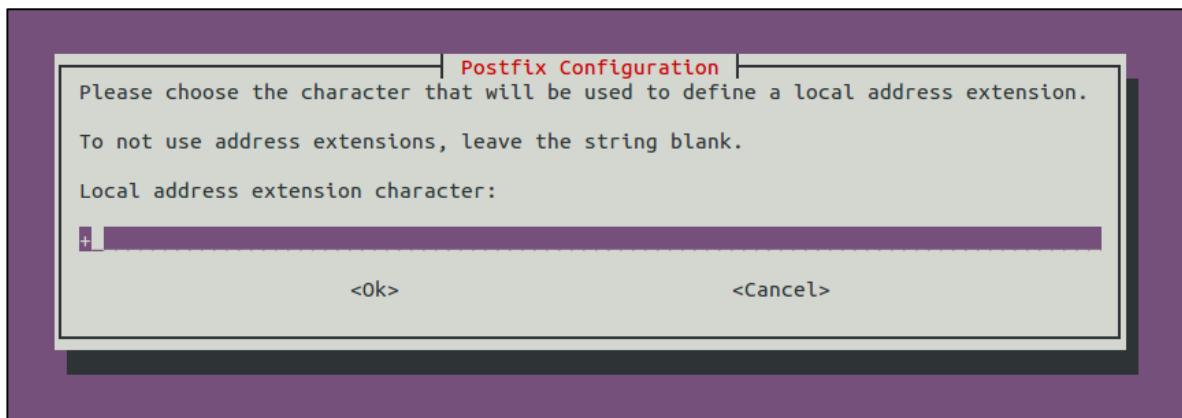


Figure 4.2-15: Local address extension character

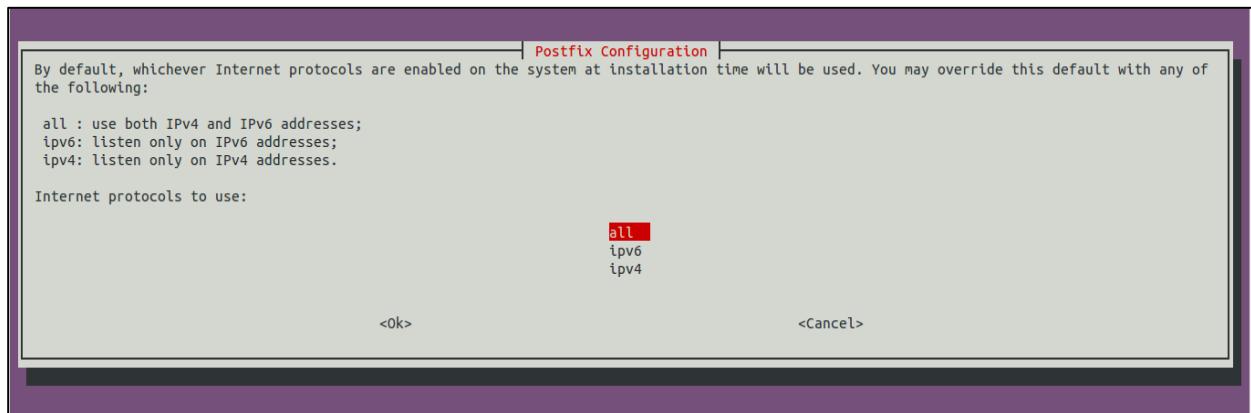


Figure 4.2-16: Internet Protocol Selection

Step 5: Now configure Postfix for SMTP-AUTH using Dovecot SASL by adding the below lines to postfix config file /etc/postfix/main.cf.

```
root@mail:~# nano /etc/postfix/main.cf
```

Figure 4.2-17: Open /etc/postfix/main.cf

```
GNU nano 2.2.6                                         File: /etc/postfix/main.cf

mailbox_size_limit = 0
recipient_delimiter =
inet_interfaces = all
inet_protocols = all
mynetworks = 127.0.0.0/8 [::ffff:127.0.0.0]/104 [::1]/128 192.168.0.0/24 0.0.0.0/0

###Tapal Addition Start###

home_mailbox = Maildir/
smtpd_sasl_type = dovecot
smtpd_sasl_path = private/auth
smtpd_sasl_local_domain =
smtpd_sasl_security_options = noanonymous
broken_sasl_auth_clients = yes
smtpd_sasl_auth_enable = yes
smtpd_recipient_restrictions = permit_sasl_authenticated,permit_mynetworks,reject_unauth_destination
smtpd_tls_security_level = may
smtpd_tls_security_level = may
smtpd_tls_note_starttls_offer = yes
smtpd_tls_loglevel = 1
smtpd_tls_received_header = yes

###Tapal Addition End###
```

Figure 4.2-18: Add Statements

Step 6: Now generate a digital certificate for tls. Issue the commands one by one and provide details as per your domain.

```
root@mail:~# openssl genrsa -des3 -out server.key 2048
```

Figure 4.2-19: Open Secure Socket Layer

```
root@mail:~# openssl genrsa -des3 -out server.key 2048
Generating RSA private key, 2048 bit long modulus
.....+ ++
.....+ ++
e is 65537 (0x10001)
Enter pass phrase for server.key:
Verifying - Enter pass phrase for server.key:
```

Figure 4.2-20: Enter pass phrase

```
root@mail:~# openssl rsa -in server.key -out server.key.insecure
```

Figure 4.2-21: OpenSSL RSA

```
root@mail:~# openssl rsa -in server.key -out server.key.insecure
Enter pass phrase for server.key:
writing RSA key
root@mail:~#
```

Figure 4.2-22: Writing RSA Key

```
root@mail:~# mv server.key server.key.secure
```

Figure 4.2-23: Moving Key

```
root@mail:~# mv server.key.insecure server.key
```

Figure 4.2-24: Security Setup

```
root@mail:~# openssl req -new -key server.key -out server.csr
```

Figure 4.2-25: Fill Details

```
root@mail:~# openssl req -new -key server.key -out server.csr
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:IN
State or Province Name (full name) [Some-State]:Maharashtra
Locality Name (eg, city) []:Mumbai
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Tapal
Organizational Unit Name (eg, section) []:Mail.Tapal
Common Name (e.g. server FQDN or YOUR name) []:Tapal
Email Address []:2014tapal@gmail.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:aaaa
An optional company name []:aaaa
root@mail:~#
```

Figure 4.2-26: Challenge Password

```
root@mail:~# openssl x509 -req -days 365 -in server.csr -signkey server.key -out server.crt
```

Figure 4.2-27: Secure Socket Layer

```
root@mail:~# openssl x509 -req -days 365 -in server.csr -signkey server.key -out server.crt
Signature ok
subject=/C=IN/ST=Maharashtra/L=Mumbai/O=Tapal/OU=Mail.Tapal/CN=Tapal/emailAddress=2014tapal@gmail.com
Getting Private key
root@mail:~#
```

Figure 4.2-28: Setting private key

```
root@mail:~# sudo cp server.crt /etc/ssl/certs
```

Figure 4.2-29: Copy Certificate

```
root@mail:~# sudo cp server.key /etc/ssl/private
```

Figure 4.2-30: Copy Key

Step 7: Now configure certificate path

```
root@mail:~# sudo postconf -e 'smtpd_tls_key_file = /etc/ssl/private/server.key'
```

Figure 4.2-31: Configure Certificate Path

```
root@mail:~# sudo postconf -e 'smtpd_tls_cert_file = /etc/ssl/certs/server.crt'
```

Figure 4.2-32: Certificate Configuration

Step 8: Open /etc/postfix/master.cf file and uncomment below lines to enable smtps (465) and submission (587)

```
root@mail:~# nano /etc/postfix/master.cf
```

Figure 4.2-33: Open /etc/postfix/master.cf/

```
GNU nano 2.2.6                                         File: /etc/postfix/master.cf

###Tapal Modifications Start###

submission inet n - - - - smtpd
  -o syslog_name=postfix/submission
  -o smtpd_tls_security_level=encrypt
  -o smtpd_sasl_auth_enable=yes
#  -o smtpd_reject_unlisted_recipient=no
#  -o smtpd_client_restrictions=$mua_client_restrictions
#  -o smtpd_helo_restrictions=$mua_helo_restrictions
#  -o smtpd_sender_restrictions=$mua_sender_restrictions
#  -o smtpd_recipient_restrictions=
  -o smtpd_relay_restrictions=permit_sasl_authenticated,reject
  -o milter_macro_daemon_name=ORIGINATING
smtps    inet  n   -   n   -   -      smtpd
  -o syslog_name=postfix/smtps
  -o smtpd_tls_wrappermode=yes
  -o smtpd_sasl_auth_enable=yes
#  -o smtpd_reject_unlisted_recipient=no
#  -o smtpd_client_restrictions=$mua_client_restrictions
#  -o smtpd_helo_restrictions=$mua_helo_restrictions
#  -o smtpd_sender_restrictions=$mua_sender_restrictions
#  -o smtpd_recipient_restrictions=
  -o smtpd_relay_restrictions=permit_sasl_authenticated,reject
  -o milter_macro_daemon_name=ORIGINATING

###Tapal Modification End###■

#628      inet  n   -   -   -   -      qmqpd
pickup   unix  n   -   -   60    1      pickup
cleanup   unix  n   -   -   -     0      cleanup
qmgr      unix  n   -   n   300    1      qmgr
#qmgr     unix  n   -   n   300    1      oqmgr
tlsmgr    unix  -   -   -   1000?  1      tlsmgr
rewrite   unix  -   -   -   -      trivial-rewrite
■G Get Help          ■O WriteOut        ■R Read File       ■Y Prev Page
■X Exit            ■J Justify         ■W Where Is        ■V Next Page
```

Figure 4.2-34: Modifications in master.cf

Step 9: Now install Dovecot SASL by typing the below command.

```
root@mail:~# sudo apt-get install dovecot-common
```

Figure 4.2-35: Install SASL

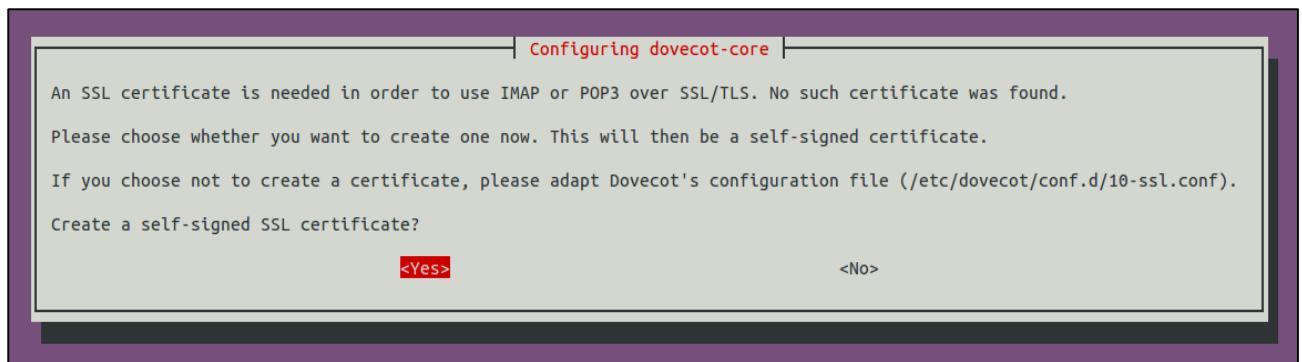


Figure 4.2-36: Create Self-signed SSL Certificate

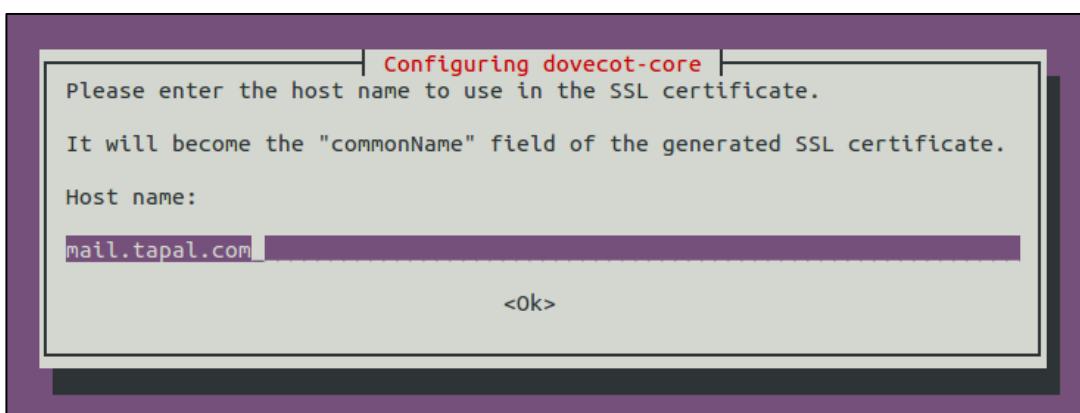


Figure 4.2-37: Assigning Hostname to Dovecot

Step 10: Make changes to the files as follows.

Open **/etc/dovecot/conf.d/10-master.conf** file and find **# Postfix smtp-auth** line (line no: 95) and add the below lines.

```
root@mail:~# nano /etc/dovecot/conf.d/10-master.conf
```

Figure 4.2-38: Open /etc/dovecot/conf.d/10-master.conf

```

GNU nano 2.2.6                                         File: /etc/dovecot/conf.d/10-master.conf

#unix_listener /var/spool/postfix/private/auth {
#  mode = 0666
#}

###Tapal Addition Start###

# Postfix smtp-auth
unix_listener /var/spool/postfix/private/auth {
mode = 0660
user = postfix
group = postfix
}

###Tapal Addition End###■

# Auth process is run as this user.
#user = $default_internal_user
}

service auth-worker {
  # Auth worker process is run as root by default, so that it can access
  # /etc/shadow. If this isn't necessary, the user should be changed to
  # $default_internal_user.
  #user = root
}

service dict {
  # If dict proxy is used, mail processes should have access to its socket.
  # For example: mode=0660, group=vmail and global mail_access_groups=vmail
  unix_listener dict {
    
```

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page
^X Exit ^J Justify ^W Where Is ^V Next Page

Figure 4.2-39: Additions in Dovecot configuration file

Open **/etc/dovecot/conf.d/10-auth.conf** file and find (line no: 100). Replace ‘Plain’ with ‘plain login’.

```
root@mail:~# nano /etc/dovecot/conf.d/10-auth.conf■
```

Figure 4.2-40: Open /etc/dovecot/conf.d/10-auth.conf

```

GNU nano 2.2.6                               File: /etc/dovecot/conf.d/10-auth.conf

# Space separated list of wanted authentication mechanisms:
# plain login digest-md5 cram-md5 ntlm rpa apop anonymous gssapi otp skey
# gss-spnego
# NOTE: See also disable_plaintext_auth setting.

###Tapal Modification Start###
auth_mechanisms = plain login
###Tapal Modification End###■

## 
## Password and user databases
##

#
# Password database is used to verify user's password (and nothing more).
# You can have multiple passdbs and userdbs. This is useful if you want to
# allow both system users (/etc/passwd) and virtual users to login without
# duplicating the system users into virtual database.
#
# <doc/wiki/PasswordDatabase.txt>
#
# User database specifies where mails are located and what user/group IDs
# own them. For single-UID configuration use "static" userdb.
#
# <doc/wiki/UserDatabase.txt>

#!include auth-deny.conf.ext
#!include auth-master.conf.ext

^G Get Help          ^O WriteOut        ^R Read File      ^Y Prev Page
^X Exit              ^J Justify         ^W Where Is       ^V Next Page

```

Figure 4.2-41: Assigning Authentication**Step 11:** Restart postfix and dovecot services

```

root@mail:~# sudo service postfix restart
* Stopping Postfix Mail Transport Agent postfix
* Starting Postfix Mail Transport Agent postfix
root@mail:~# ■

```

Figure 4.2-42: Restart Postfix

```

root@mail:~# sudo service dovecot restart
dovecot stop/waiting
dovecot start/running, process 6216
root@mail:~# ■

```

Figure 4.2-43: Restart Dovecot

Step 12: Now test SMTP-AUTH and smtp/pop3 port access.

Type the below command and should get below response.

```
root@mail:~# telnet mail.tapal.com smtp
Trying 192.168.0.6...
Connected to mail.tapal.com.
Escape character is '^].
220 mail.tapal.com ESMTP Postfix (Ubuntu)
ehlo mail.tapal.com
250-mail.tapal.com
250-PIPELINING
250-SIZE 10240000
250-VRFY
250-ETRN
250-STARTTLS
250-AUTH PLAIN LOGIN
250-AUTH=PLAIN LOGIN
250-ENHANCEDSTATUSCODES
250-8BITMIME
250 DSN
■
```

Figure 4.2-44: Checking SMTP Connection

4.2.2 Installing and configuring dovecot

Step 13: Install dovecot using the below command

```
root@mail:~# sudo apt-get install dovecot-imapd dovecot-pop3d
```

Figure 4.2-45: Install IMAP and POP3

```
root@mail:~# sudo apt-get install dovecot-imapd dovecot-pop3d
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  dovecot-imapd dovecot-pop3d
0 upgraded, 2 newly installed, 0 to remove and 19 not upgraded.
Need to get 150 kB of archives.
After this operation, 1,592 kB of additional disk space will be used.
Get:1 http://in.archive.ubuntu.com/ubuntu/ trusty-updates/main dovecot-imapd amd64 1:2.2.9-1ubuntu2.1 [122 kB]
Get:2 http://in.archive.ubuntu.com/ubuntu/ trusty-updates/main dovecot-pop3d amd64 1:2.2.9-1ubuntu2.1 [28.5 kB]
Fetched 150 kB in 1s (91.1 kB/s)
Selecting previously unselected package dovecot-imapd.
(Reading database ... 194539 files and directories currently installed.)
Preparing to unpack .../dovecot-imapd_1%3a2.2.9-1ubuntu2.1_amd64.deb ...
Unpacking dovecot-imapd (1:2.2.9-1ubuntu2.1) ...
Selecting previously unselected package dovecot-pop3d.
Preparing to unpack .../dovecot-pop3d_1%3a2.2.9-1ubuntu2.1_amd64.deb ...
Unpacking dovecot-pop3d (1:2.2.9-1ubuntu2.1) ...
Processing triggers for dovecot-core (1:2.2.9-1ubuntu2.1) ...
Setting up dovecot-imapd (1:2.2.9-1ubuntu2.1) ...

Creating config file /etc/dovecot/conf.d/20-imap.conf with new version
Setting up dovecot-pop3d (1:2.2.9-1ubuntu2.1) ...

Creating config file /etc/dovecot/conf.d/20-pop3.conf with new version
Processing triggers for dovecot-core (1:2.2.9-1ubuntu2.1) ...
root@mail:~#
```

Figure 4.2-46: Service Installed

Step 14: Now configure mailbox. Open **/etc/dovecot/conf.d/10-mail.conf** file and find (Line no: 30)

```
root@mail:~# nano /etc/dovecot/conf.d/10-mail.conf
```

Figure 4.2-47: Open mail.conf

```

GNU nano 2.2.6                                         File: /etc/dovecot/conf.d/10-mail.conf

# %h - home directory
#
# See doc/wiki/Variables.txt for full list. Some examples:
#
# mail_location = maildir:~/Maildir
# mail_location = mbox:~/mail:INBOX=/var/mail/%u
# mail_location = mbox:/var/mail/%d/%1n/%n:INDEX=/var/indexes/%d/%1n/%n
#
# <doc/wiki/MailLocation.txt>
#


####Tapal Modification Start####
mail_location = maildir:~/Maildir
####Tapal Modification End####

# If you need to set multiple mailbox locations or want to change default
# namespace settings, you can do it by defining namespace sections.
#
# You can have private, shared and public namespaces. Private namespaces
# are for user's personal mails. Shared namespaces are for accessing other
# users' mailboxes that have been shared. Public namespaces are for shared
# mailboxes that are managed by sysadmin. If you create any shared or public
# namespaces you'll typically want to enable ACL plugin also, otherwise all
# users can access all the shared mailboxes, assuming they have permissions
# on filesystem level to do so.
namespace inbox {
    # Namespace type: private, shared or public
    #type = private

    # Hierarchy separator to use. You should use the same separator for all
    # namespaces or some clients get confused. '/' is usually a good one.
    # The default however depends on the underlying mail storage format.
    #separator =

```

^{^G} Get Help ^{^O} WriteOut ^{^R} Read File [Cancelled]

^{^X} Exit ^{^J} Justify ^{^W} Where Is ^{^Y} Prev Page

^{^V} Next Page

Figure 4.2-48: Mail Directory Location

Step 15: Now change pop3_uidl_format. Open /etc/dovecot/conf.d/20-pop3.conf file and find and

Uncomment the below line (Line no: 50)

```
root@mail:~# nano /etc/dovecot/conf.d/20-pop3.conf
```

Figure 4.2-49: Open pop3.conf

```

GNU nano 2.2.6                                         File: /etc/dovecot/conf.d/20-pop3.conf

# If you want UIDL compatibility with other POP3 servers, use:
# UW's ipop3d      : %08Xv%08Xu
# Courier          : %f or %v.%u (both might be used simultaneously)
# Cyrus (<= 2.1.3)  : %u
# Cyrus (>= 2.1.4)  : %v.%u
# Dovecot v0.99.x   : %v.%u
# tpop3d           : %Mf
#
# Note that Outlook 2003 seems to have problems with %v.%u format which was
# Dovecot's default, so if you're building a new server it would be a good
# idea to change this. %08Xu%08Xv should be pretty fail-safe.
#
####Tapal Modification Start####
pop3_uidl_format = %08Xu%08Xv
####Tapal Modification End####

# Permanently save UIDs sent to POP3 clients, so pop3_uidl_format changes
# won't change those UIDs. Currently this works only with Maildir.
#pop3_save_uidl = no

# What to do about duplicate UIDs if they exist?
#   allow: Show duplicates to clients.
#   rename: Append a temporary -2, -3, etc. counter after the UID.
#pop3_uidl_duplicates = allow

# This option changes POP3 behavior so that it's not possible to actually
# delete mails via POP3, only hide them from future POP3 sessions. The mails
# will still be counted towards user's quota until actually deleted via IMAP.
# Use e.g. "$POP3Deleted" as the value (it will be visible as IMAP keyword).
# Make sure you can legally archive mails before enabling this setting.
#pop3_deleted_flag =

```

^G Get Help **^O** WriteOut **^R** Read File **^Y** Prev Page
^X Exit **AJ** Justify **^W** Where Is **^V** Next Page

Figure 4.2-50: Modify POP3 UIDL Format

Step 16: Now enable SSL. Open **/etc/dovecot/conf.d/10-ssl.conf** file and find and uncomment the below line (Line no: 6)

```
root@mail:~# nano /etc/dovecot/conf.d/10-ssl.conf
```

Figure 4.2-51: Open ssl.conf

```
GNU nano 2.2.6                                         File: /etc/dovecot/conf.d/10-ssl.conf

## SSL settings

# SSL/TLS support: yes, no, required. <doc/wiki/SSL.txt>

###Tapal Modification Start###

ssl = yes

###Tapal Modification End###■

# PEM encoded X.509 SSL/TLS certificate and private key. They're opened before
# dropping root privileges, so keep the key file unreadable by anyone but
# root. Included doc/mkcert.sh can be used to easily generate self-signed
# certificate, just make sure to update the domains in dovecot-openssl.cnf
ssl_cert = </etc/dovecot/dovecot.pem
ssl_key = </etc/dovecot/private/dovecot.pem

# If key file is password protected, give the password here. Alternatively
# give it when starting dovecot with -p parameter. Since this file is often
# world-readable, you may want to place this setting instead to a different
# root owned 0600 file by using ssl_key_password = <path>.
#ssl_key_password =

# PEM encoded trusted certificate authority. Set this only if you intend to use
# ssl_verify_client_cert=yes. The file should contain the CA certificate(s)
# followed by the matching CRL(s). (e.g. ssl_ca = </etc/ssl/certs/ca.pem>
#ssl_ca =

# Require that CRL check succeeds for client certificates.
#ssl_require_crl = yes

# Directory and/or file for trusted SSL CA certificates. These are used only

^G Get Help          ^O WriteOut        ^R Read File      ^Y Prev Page
^X Exit              ^J Justify         ^W Where Is       ^V Next Page
```

Figure 4.2-52: Setting SSL

Step 17: Restart dovecot service.

```
root@mail:~# sudo service dovecot restart
dovecot stop/waiting
dovecot start/running, process 7504
root@mail:~# ■
```

Figure 4.2-53: Restart Dovecot

Step 18: Now test pop3 and imap port access using the telnet command.

```
root@mail:~# telnet mail.tapal.com 110■
```

Figure 4.2-54: Checking Port 110

```
root@mail:~# telnet mail.tapal.com 110
Trying 192.168.0.6...
Connected to mail.tapal.com.
Escape character is '^].
+OK Dovecot (Ubuntu) ready.
```

Figure 4.2-55: Acknowledgement to Connection

Step 19: Create some users

```
root@mail:~# sudo useradd -m tapaluser1 -s /sbin/nologin
```

Figure 4.2-56: Create User

```
root@mail:~# sudo passwd tapaluser1
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
root@mail:~#
```

Figure 4.2-57: Assign Password

4.2.3 Installing and configuring Squirrelmail

Step 20: Install Squirrelmail using the below command. This will install apache and PHP packages.

```
root@mail:~# sudo apt-get install squirrelmail
```

Figure 4.2-58: Install Squirrelmail

```

Creating config file /etc/php5/mods-available/opcode.ini with new version
php5_invoke: Enable module opcode for apache2 SAPI
php5_invoke: Enable module opcode for cli SAPI
Setting up php5-cli (5.5.9+dfsg-1ubuntu4.7) ...
update-alternatives: using /usr/bin/php5 to provide /usr/bin/php (php) in auto mode

Creating config file /etc/php5/cli/php.ini with new version
php5_invoke opcode: already enabled for cli SAPI
php5_invoke json: already enabled for cli SAPI
php5_invoke pdo: already enabled for cli SAPI
Setting up php5-readline (5.5.9+dfsg-1ubuntu4.7) ...

Creating config file /etc/php5/mods-available/readline.ini with new version
php5_invoke: Enable module readline for apache2 SAPI
php5_invoke: Enable module readline for cli SAPI
Processing triggers for ureadahead (0.100.0-16) ...
Processing triggers for ufw (0.34~rc-0ubuntu2) ...
Setting up libapache2-mod-php5 (5.5.9+dfsg-1ubuntu4.7) ...

Creating config file /etc/php5/apache2/php.ini with new version
php5_invoke readline: already enabled for apache2 SAPI
php5_invoke opcode: already enabled for apache2 SAPI
php5_invoke json: already enabled for apache2 SAPI
php5_invoke pdo: already enabled for apache2 SAPI
Module mpm_event disabled.
Enabling module mpm_prefork.
apache2_switch_mpm Switch to prefork
 * Restarting web server apache2
apache2_invoke: Enable module php5
 * Restarting web server apache2
Setting up squirrelmail (2:1.4.23~svn20120406-2) ...
Installing default squirrelmail config.
Run /usr/sbin/squirrelmail-configure as root to configure/upgrade config.
Setting up squirrelmail-locales (1.4.18-20090526-1) ...
Setting up squirrelmail-viewashtml (3.8-3) ...
Removing plugin view_as_html
Data saved in config.php
Activating plugin view_as_html
Data saved in config.php
Processing triggers for libc-bin (2.19-0ubuntu6.6) ...
root@mail:~# 

```

Figure 4.2-59: Squirrelmail Installed

Step 21: Now configure apache to enable Squirrelmail.

```

root@mail:~# sudo cp /etc/squirrelmail/apache.conf /etc/apache2/sites-available/squirrelmail.conf
root@mail:~# 

```

Figure 4.2-60: Configure Apache with Squirrelmail

```

root@mail:~# sudo a2ensite squirrelmail
Enabling site squirrelmail.
To activate the new configuration, you need to run:
    service apache2 reload
root@mail:~# 

```

Figure 4.2-61: Configuration

Step 22: Restart Apache service

```
root@mail:~# sudo service apache2 restart
 * Restarting web server apache2
root@mail:~# █
```

Figure 4.2-62: Restart Apache Service

Step 23: Configure Squirrelmail

```
root@mail:~# sudo squirrelmail-configure█
```

Figure 4.2-63: Configure Squirrelmail

```
SquirrelMail Configuration : Read: config.php (1.4.0)
-----
Main Menu --
1. Organization Preferences
2. Server Settings
3. Folder Defaults
4. General Options
5. Themes
6. Address Books
7. Message of the Day (MOTD)
8. Plugins
9. Database
10. Languages

D. Set pre-defined settings for specific IMAP servers

C Turn color on
S Save data
Q Quit

Command >> █
```

Figure 4.2-64: Configuration Options

```
SquirrelMail Configuration : Read: config.php (1.4.0)
-----
Organization Preferences
1. Organization Name      : Tapal
2. Organization Logo     : ../images/logo.png
3. Org. Logo Width/Height : (308/111)
4. Organization Title    :
5. Signout Page          :
6. Top Frame              : _top
7. Provider link         : ../src/about.php
8. Provider name         :

R  Return to Main Menu
C  Turn color on
S  Save data
Q  Quit

Command >> █
```

Figure 4.2-65: Modify Configuration Values

Step 24: Now open **http://serverIP/squirrelmail** in your browser and login using username (bobby). You can send and receive mail using Squirrelmail.

4.2.4 Postfix Multi-Instance

Listing managed instances

The list of managed instances consists of the default instance and the additional instances whose configuration directories are listed (in start order) under the multi_instance_directories parameter of the default main.cf configuration file.

You can list selected instances, groups of instances or all instances by specifying only the instance matching options with the "-l" option. The "-a" option is assumed if no other instance selection options are specified (this behaviour changes with the "-e" option). As a special case, even if it has an explicit name, the default instance can always be selected via "-i -".

```
# postmulti -l -a
# postmulti -l -g a_group
# postmulti -l -i an_instance
```

The output is one line per instance (in "postfix start" order):

name	group	enabled	config_directory
- -	yes	/etc/postfix	
mta-out	mta	yes	/etc/postfix/mta-out
mta-in	mta	yes	/etc/postfix-mta-in
msa-out	msa	yes	/etc/postfix-msa-out
msa-in	msa	yes	/etc/postfix-msa-in
test -	no		/etc/postfix-test

The first line showing the column headings is not part of the output. When either the instance name or the instance group is not set, it is shown as a "-".

When selecting an existing instance via the "-i" option, you can always use the full pathname of its configuration directory instead of the instance (short) name. This is the only way to select a non-default nameless instance. The default instance can be selected via "-i -", whether it has a name or not.

Enabling Multi instance on Postfix

To enable multi instance on postfix execute following command:

```
# postmulti -e init
```

Creating a new Postfix instance

To create a new instance just use the -e create option:

```
# postmulti -I postfix-myinst -e create
```

Destroying a Postfix instance

If you no longer need an instance, you can destroy it via:

```
# postmulti -i postfix-myinst -p stop  
# postmulti -i postfix-myinst -e disable  
# postmulti -i postfix-myinst -e destroy
```

Enabling/disabling managed instances

You can enable or disable a managed instance. As documented in `postfix-wrapper(5)`, disabled instances are skipped with actions that start, stop or control running Postfix instances.

```
# postmulti -i postfix-myinst -e enable  
# postmulti -i postfix-myinst -e disable
```

4.3 Coding

4.3.1 Squirrelmail Modifications

File Name: about.php

```

echo "<html><body>";

echo html_tag( 'table style="width:80%;"',
    html_tag( 'tr style="width:80%;",
        html_tag( 'td style="width:80%;'',<div style="width:500px;text-align: center;"><b>' .
        . _("About") . '</b></div>', 'center', $color[0] )

    ),
    'center', ", 'width="95%" cellpadding="1" cellspacing="2" border="0"" );

echo "<div style='padding:0px 50px 0px 50px;'>
<br/>&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp&nbsp

```

Our project, 'TAPAL' provides such a system that blends the modern day email facilities with an Indian touch for a homely environment. Our main aim was to provide organizations with their own TAPAL system for communicating amongst the employees, workers and whosoever concerned. It allows access to users validated only and only by the server. We can say that user creation is looked after only by the administrator operating the server. The server ought to be a Linux based platform. All user files are stored upon the server in the form of mail directories.

A lot of effort and thought has been linked with this project, not only did it open new windows for learning but aided us in understanding the need for project making and relationship building, i.e. soft skills.

Our thought engraved with this project is,

<center> <i>

Throwback to the olden time letters, the modern way... </i> </center>

TAPAL, means 'Letter' in Gujarati (India).We give you the best of both times! A comforting look to sooth your Indian souls blended with full functionality of a modern day email service.

We are the students of S.B.M. Polytechnic, who have put in efforts to create the modern email communication, thoughtfully named 'TAPAL'.

1280010 Mansi Deep Dandiwala

Handpicked colors, shades & added essence of India's culture, just for you!

1280047 Kshitij Manish Shah

Using our skills in conjunction with our thoughts to provide you with a realistic modern day email experience.

1280051 Ankit Purvesh Shah

Studying various mailing systems, in greater depth to give you the best! We strive to serve you better.

<center><pre>Contact Us

Email: 2014tapal@gmail.com

Web: mail.tapal.com/squirrelmail</pre></center>

</div>"

?> </td>

<tr>

</table>

</body>

</html>

Reflected Changes:

- Addition of contents to the about page.
- Styling to that content to fit the Tapal theme.

File name: adressbook.php

```

echo '<a name="AddAddress" style="height:100px;"></a>' . "\n" .
addForm($form_url, 'post', 'f_add', " ", " ", TRUE).

html_tag( 'table',
    html_tag( 'tr',
        html_tag( 'td', "\n" . '<strong>' . sprintf(_("Add to %s"), $abook->localbackendname) .
        '</strong>' . "\n",
        'center', $color[0]
    )
)
,
'center', " ", 'width="95%"' ) ."\n";

```

Reflected changes:

- Height of 100px has been assigned to addressbook field.

File Name: compose.php

```

echo ' <tr>' . "\n" .

html_tag( 'td', " ", 'right', $color[4], 'style="font-size:18px;"width="10%"' ) .
_("To:") . '&nbsp&nbsp</td>' . "\n" .

html_tag( 'td', " ", 'left', $color[4], 'width="90%"' ) .

substr(addInput('send_to', $send_to, 60), 0, -3). $onfocus . 'style="border:1px groove
black; border-radius:5px; width:480px; padding-left:5px; height:25px; font-
size:15px; background-color:rgba(0,0,0,0.2);"/><br />' . "\n" .

' </td>' . "\n" .

' </tr>' . "\n" .

' <tr>' . "\n" .

html_tag( 'td', " ", 'right', $color[4], 'style="font-size:18px;"' ) .

```

```

_("Cc:") . '&nbsp&nbsp</td>' . "\n" .

html_tag( 'td', ", 'left', $color[4] ) .

substr(addInput('send_to_cc', $send_to_cc, 60), 0, -3). $onfocus . 'style="border:1px
groove black;border-radius:5px;width:480px;padding-left:5px;height:25px;font-
size:15px;background-color:rgba(0,0,0,0.2);" /><br />' . "\n" .

'      </td>' . "\n" .

'    </tr>' . "\n" .

'  <tr>' . "\n" .

html_tag( 'td', ", 'right', $color[4], 'style="font-size:18px;"" ) .

_("Bcc:") . '&nbsp&nbsp</td>' . "\n" .

html_tag( 'td', ", 'left', $color[4] ) .

substr(addInput('send_to_bcc', $send_to_bcc, 60), 0, -3). $onfocus . ' style="border:1px
groove black;border-radius:5px;width:480px;padding-left:5px;height:25px;font-
size:15px;background-color:rgba(0,0,0,0.2);"/><br />' . "\n" .

'      </td>' . "\n" .

'    </tr>' . "\n" .

'  <tr>' . "\n" .

html_tag( 'td', ", 'right', $color[4], 'style="font-size:18px;"" ) .

_("Subject:") . '&nbsp&nbsp</td>' . "\n" .

html_tag( 'td', ", 'left', $color[4] ) . "\n";

echo '      .substr(addInput('subject', $subject, 600), 0, -3). $onfocus .style="border:1px
groove black;border-radius:5px;width:480px;padding-left:5px;height:25px;font-
size:15px;background-color:rgba(0,0,0,0.2);" />    </td>' . "\n" .

'    </tr>' . "\n\n";

```

/* why this distinction? */

```

if ($compose_new_win == '1') {

echo '  <tr>' . "\n" .

```

```

'      <td bgcolor="" . $color[0] . "" colspan="2" align="center">' . "\n" .
'
'          <textarea name="body" style="border:1px groove black; border-
radius:5px; padding-left:5px; font-size:15px; background-color:rgba(0,0,0,0.2); id="body"
rows="" . (int)$editor_height .
"
" cols="" . (int)$editor_size . "" wrap="virtual" . $onfocus . '>';
}

else {

echo ' <tr>' . "\n" .

'      <td bgcolor="" . $color[4] . "" colspan="2">' . "\n" .

'          &nbsp;&nbsp;<textarea style="padding:10px 10px; border:1px groove
black; border-radius:5px; font-size:15px; background-color:rgba(0,0,0,0.2); name="body"
id="body" rows="" . (int)$editor_height .
"
" cols="" . (int)$editor_size . "" wrap="virtual" . $onfocus . '>';

}

```

Reflected Changes:

- Editing compose page input fields.
- Adding borders, setting height, width etc. to form fields.

File Name: folders.php

```

addInput('folder_name', "", 25).

"<br />\n". _("as a subfolder of"). '<br />'.

"<tt><select name=\"subfolder\" style=\"height:30px; background-
color:rgba(0,0,0,0.2); font-weight:bold; border-radius:5px;; border:solid 2px #000; margin-
right:10px; margin-left:10px;\">\n";

-----
echo "<input type=\"submit\" value=\"\" . _("Create") . \" style=\"margin-top:10px; border-
radius:7px; border:2px solid ; height:30px; background-color:rgba(0,0,0,0.2); font-
weight:bold;\" />\n";
-----
```

```

if (count($skip_folders) < count($boxes)) {

    echo addForm('folders_rename_getname.php')

        . "<tt><select name=\"old\" style=\"height:30px;background-color:rgba(0,0,0,0.2);font-
weight:bold;border-radius:5px;;border:solid 2px #000;margin-right:10px;margin-
left:10px;\">\n"
        . '      <option value="">[ ' . _("Select a folder") . " ]</option>\n";
-----


echo "</select></tt>\n".

'<input type="submit" value="".
_("Rename").

"\\" style=\"margin-top:10px;border-radius:7px;border:2px solid
;height:30px;background-color:rgba(0,0,0,0.2);font-weight:bold;\"/>\n".
"</form></td></tr>\n";
-----


if (count($skip_folders) < count($boxes)) {

    echo addForm('folders_delete.php')

        . "<tt><select name=\"mailbox\" style=\"height:30px;background-
color:rgba(0,0,0,0.2);font-weight:bold;border-radius:5px;;border:solid 2px #000;margin-
right:10px;margin-left:10px;\">\n"
        . '      <option value="">[ ' . _("Select a folder") . " ]</option>\n";

        // send NULL for the flag - ALL folders are eligible for delete (except what we've got in
        // skiplist)

        // use long format to make sure folder names make sense when parents may be missing.

        echo sqimap_mailbox_option_list($imapConnection, 0, $skip_folders, $boxes, NULL,
true);

    echo "</select></tt>\n"

        . '<input type="submit" value=""'

```

```

. _("Delete")

. "\" style="margin-top:10px;border-radius:7px;border:2px solid
;height:30px;background-color:rgba(0,0,0,0.2);font-weight:bold;\">\n"

. "</form></td></tr>\n";
-----\n\n
echo addForm('folders_subscribe.php?method=unsub', 'post', "", "", "", TRUE)

. "<tt><select name=\"mailbox[]\" style=\"background-color:rgba(0,0,0,0.2);font-
weight:bold;border-radius:5px;;border:solid 2px #000;margin-right:10px;margin-
left:10px;\"multiple=\"multiple\" size=\"8\">\n";
-----\n\n

```

Reflected changes:

- Styling to the submit button.
- Styling the text fields and other input fields on the folder option page.
- Adding background colours to various table rows and table fields.

File Name: help.php

```

echo html_tag( 'table',
               html_tag( 'tr',
                         html_tag( 'td','<div style="text-align: center;"><b>' ._("Help") . '</b></div>', 'center',
$color[0] )
               ),
               'center', ", 'width="95%" cellpadding="1" cellspacing="2" border="0"" );
-----\n\n

```

// Write the TOC header

```

echo html_tag( 'table', ", 'center', ", 'cellpadding="0" cellspacing="0" border="0"" ).

html_tag( 'tr' ).

html_tag( 'td' ).

'<div style="text-align: center;"><b>' ._("Table of Contents") . '</b></div><br />';
-----\n\n

```

```

echo html_tag( 'ol' );

-----
// Write the chapter header.

echo '<div style="text-align: center;"><small>';

-----
echo '<br /><div style="text-align: center;"><a href="#pagetop">' . _("Top") . '</a></div>';

-----
```

Reflected Changes:

- Customizing help page
- Styling the links, content and the background.

File Name: left_main.php

```

/* Create the link for this folder. */

if ($status !== false) {

    echo "<br>";

    $line .= '<a
href="right_main.php?PG_SHOWALL=0&sort=0&startMessage=1&mailbox
='.

        $mailboxURL." target="right" style="font-size:17.5px;font-
weight:bold;width:100px;text-decoration:none;color:#000;">';

}

-----
if (($numMessages > 0) or ($box_array['parent'] == 1)) {

    $urlMailbox = urlencode($real_box);

    $line .= "\n<small>\n" .

        '&nbsp;&nbsp;(<a href="empty_trash.php?smtoken=' .
    sm_generate_security_token() . '" style="text-decoration:none">'._("Purge").'</a>)' .
```

```

'</small>';

}

-----
/* Create the link for this collapse link. */

$link = '<a target="left" style="text-decoration:none" ' .
'href="left_main.php?';

-----

echo "\n<body background=..../images/bg_left_1.jpg' style='margin:0px;padding:0px;
background-size:100% 100%;background-repeat:no-repeat;' text=\"$color[6]\"
link=\"$color[6]\" vlink=\"$color[6]\" alink=\"$color[6]\">\n";

-----

echo "\n\n" . html_tag( 'table', "", 'center', "", 'style="" border="0" cellspacing="0"
cellpadding="0" width="99%" ) .

html_tag( 'tr' ) .

html_tag( 'td style="width:100%;text-align:center;:"", "", "") .

html_tag( 'table', "", "", 'style="text-align:center;padding:10px;background-
color:rgba(0,0,0,0);style=""border="0" cellspacing="0" cellpadding="0"" ) .

/*html_tag( 'tr' ) .

html_tag('img src=..../images/logo.png" style="height:70px; width:140px;"') .*/
html_tag('tr') .

html_tag( 'td style="" ', "", 'center' ) .

'<font size="4"><b>'. _("") . "</b><br /></font>\n\n";

-----

echo '<small><span style="white-space: nowrap;">' .
. str_replace(' ', '&nbsp;', _("Last Refresh")) .
. ":</span><br /><span style=\"white-space: nowrap;\">$clk</span></small><br />';

}

```

```
/* Next, display the refresh button. */

echo '<small style="white-space: nowrap;">(<a href="../src/left_main.php" target="left">'.
      _("Refresh") . '</a>)</small></td></tr></table><br /><br /><br /><br /><br /><br />';

$line = "<center><span style="white-space: nowrap; font-size: 20px;
margin:auto; width:200px;"> </center><tt></tt>";
```

Reflected Changes:

- Modification and styling of left_main page i.e. left frame of home page
- Adding of background, setting margins and paddings, adding various styles to the folder options.
- Formatting of Refresh style and name value.

File Name: login.php

```
$bg="background-image: url('../images/loginbg.jpg');

background-repeat: no-repeat; background-size:100%;";

displayHtmlHeader( "$org_name - " . _("Login"), $header, FALSE );

echo "<body text=\"$color[8]\" style=\"$bg\" background='../images/loginbg.jpg'
link=\"$color[7]\" vlink=\"$color[7]\" alink=\"$color[7]\""
onLoad=\"squirrelmail_loginpage_onload();\">".

"\n" . addForm('redirect.php', 'post', 'login_form');

echo html_tag( 'table style="background-color:transparent; margin-left:350px; margin-top:350px;"',
html_tag( 'tr',
html_tag( 'td',
```

```

'<center>'.

( isset($org_logo) && $org_logo

? '<img style="position:absolute; top:225px; left:43%;" src="" . $org_logo . "" alt="" .

sprintf(_("%s Logo"), $org_name) ."" . $width_and_height .

' /><br />' . "\n"

: ").

( (isset($hide_sm_attributions) && $hide_sm_attributions) ? :

"\n".

" . _() . "\n" .

html_tag('strong style="font-size:20px;" ',

"WELCOME. Please Login."


).

html_tag( 'table style="background-color:transparent;" ',

html_tag( 'tr' ,

html_tag( 'td style="background-color:transparent;"" , "\n" .

html_tag( 'table style="background-color:transparent; font-size:20px;" ' ,

html_tag( 'tr' ,

html_tag( 'td' ,

"-("") ,


'right', " , 'width="30%'" ) .

html_tag( 'td' ,

addInput($username_form_name, $loginname_value, 0, 0,
'onfocus="alreadyFocused=true;" style="background-color: #e5e5e5;

border: none;

border-radius: 3px;

```

```
-moz-border-radius: 3px;  
-webkit-border-radius: 3px;  
  
color: #5a5656;  
  
font-size: 20px;  
  
height: 50px;  
  
outline: none;  
  
padding: 0px 10px;  
  
width: 280px;  
  
-webkit-appearance:none;  
  
box-shadow: 0 0 3px 3px #663300;"'),  
  
    'left', ", 'width='70%'" )  
  
) . "\n" .  
  
    html_tag( 'tr',  
  
        html_tag( 'td',  
  
            _(""),  
  
            'right', ", 'width='30%'" ).  
  
        html_tag( 'td',  
  
            addPwField($password_form_name, null, 'onfocus="alreadyFocused=true;"  
style="background-color: #e5e5e5;  
  
border: none;  
  
border-radius: 3px;  
  
-moz-border-radius: 3px;  
  
-webkit-border-radius: 3px;  
  
color: #5a5656;  
  
margin-top:5px;
```

```
font-size: 20px;  
height: 50px;  
outline: none;  
padding: 0px 10px;  
width: 280px;  
-webkit-appearance:none;  
box-shadow: 0 0 3px 3px #663300;"").  
addHidden('js_autodetect_results', SMPREF_JS_OFF).  
$mailtofield .  
addHidden('just_logged_in', '1'),  
'left', ", 'width="70%"" )  
) ,  
'center', $color[4], 'border="0" width="100%"') ,  
'left',$color[4] )  
) .  
html_tag( 'tr',  
html_tag( 'td',  
'<center>'. addSubmit(_("Login"),", 'style="position:absolute; top:410;  
left:610;background-color: #996633;box-shadow: 0 0 3px 3px #663300; border: none; border-  
radius: 180px; color: #f4f4f4; cursor: pointer; height: 80px; text-transform: uppercase; width:  
80px;") .'</center>',  
'left' )  
) ,  
", $color[4], " ) . '</center>',  
'center' )  
) ,
```

```
", $color[4],");  
  
do_hook('login_form');  
  
echo '</form>' . "\n";
```

Reflected Changes:

- Background is added.
- Form fields are styled and positioned according to our requirements.

File Name: read_body.php

```
$s = '<table width="100%" cellpadding="0" cellspacing="2" border="0"';  
  
' cellspacing="0" border="0" align="center" style="">."\n";  
  
$s .= ' align="center" bgcolor="'.$color[0].'" style="margin-top:-5px;padding-  
bottom:5px;background-color:rgba(0,0,0,0.1);border-bottom:solid 2px #000;">;  
  
-----  
  
$s = '<table width="100%" cellpadding="3" cellspacing="0" align="center"'.  
  
' border="0" bgcolor="'.$color[9].'" style="background-color:rgba(0,0,0,0.2);margin-  
top:-18px;"><tr> .  
  
html_tag( 'td', 'left', 'width="33%"' ) . '<small>';  
  
-----  
  
echo html_tag( 'table' , 'left', 'cellpadding="1" cellspacing="5" border="0"  
style="padding-top:10px; border: solid 2px #000; border-radius:5px; background-  
color:rgba(0,0,0,0.2); width:90%; margin-left:5%; margin-top:-7px;" );
```

Reflected Changes:

- Styling of mail display area, and other input fields.
- Adding background colour to improve the look and give it the TAPAL touch.

File Name: redirect.php

```
echo "<body background='..../images/bg.jpg' style='background-size:100%; background-repeat:no-repeat;'>";
```

Reflected Changes:

- Background is added.

File Name: right_main.php

```
echo'<body background="..../images/bg.jpg" style="background-size:100% 100%; background-repeat:no-repeat; margin:0px; padding:0px;">';
```

Reflected Changes:

- Background is added.

File Name: search.php

```
echo html_tag( 'div', '<b>' . _("Current Search") . '</b>', 'left' ) . "\n"

. '<form action="search.php" name="s">'

. addHidden('smtoken', sm_generate_security_token())

. html_tag( 'table', "", "", "width='95%' cellpadding='0' cellspacing='0' border='0'" )

. html_tag( 'tr' )

. html_tag( 'td', "", 'left' )

. '<select name="mailbox" style="height:30px; background-color:rgba(0,0,0,0.2); font-weight:bold; border-radius:5px; border:solid 2px #000; margin-right:10px; margin-left:10px;">'

. '<option value="All Folders"'; 

if ($mailbox == 'All Folders') {

    echo ' selected="selected"'; 

}

echo '>[ ' . _("All Folders") . " ]</option>\n";
```

```

$show_selected = array(strtolower($mailbox));

echo sqimap_mailbox_option_list($imapConnection, $show_selected, 0, $boxes);

echo '      </select>'.

"      </td>\n";

// FIXME: explain all str_replace calls.

$what_disp = str_replace(',', ' ', $what);

$what_disp = str_replace('\\\\\\', '\\', $what_disp);

$what_disp = str_replace("\\\"", "", $what_disp);

$what_disp = str_replace("'", '"', $what_disp);

echo html_tag( 'td', '<input type="text" size="35" name="what" style="border:1px groove black; border-radius:5px; width:480px; padding-left:5px; height:25px; font-size:15px; background-color:rgba(0,0,0,0.2);" value="" . $what_disp . "" /> . "\n", 'center' )

.html_tag( 'td', ", 'right' )

. "<select name=\"where\" style=\"height:30px; background-color:rgba(0,0,0,0.2); font-weight:bold; border-radius:5px; border:solid 2px #000; margin-right:10px; margin-left:10px;\">";

s_opt( 'BODY', $where, _("Body") );

s_opt( 'TEXT', $where, _("Everywhere") );

s_opt( 'SUBJECT', $where, _("Subject") );

s_opt( 'FROM', $where, _("From") );

s_opt( 'CC', $where, _("Cc") );

s_opt( 'TO', $where, _("To") );

echo "      </select>\n" .

"      </td>\n".

html_tag( 'td', '<input type="submit" style="margin-top:px; border-radius:7px; border:2px solid ; height:30px; background-color:rgba(0,0,0,0.2); font-weight:bold;" name="submit" value="" . _("Search") . "" /> . "\n", 'center', ", 'colspan=3'" ) .

```

```

    "  </tr>\\n".
    "  </table>\\n".
"</form>\\n";

```

Reflected Changes:

- Search page is modified according to the TAPAL theme.
- Buttons, input fields are styled.

File Name: signout.php

```

<body text=<?php echo $color[8]; ?>" background=..//images/bg.jpg" style="background-
size:100% 100%;" bgcolor=<?php echo $color[4]; ?>

link=<?php echo $color[7]; ?>" vlink=<?php echo $color[7]; ?>
alink=<?php echo $color[7]; ?>">

```

Reflected Changes:

- Background is modified.

File Name: webmail.php

```

if ($location_of_bar == 'right') {

    $output .= "<frame src=\"$right_frame_url\" name=\"right\" style=\"background-
color:transparent;margin:0px;padding:0px;border:0px;\" frameborder=\"0\">\\n" .

        "<frame src=\"left_main.php\" name=\"left\" style=\"background-
color:transparent;margin:0px;padding:0px;border:0px;\" frameborder=\"0\">\\n";

    }
else {

    $output .= "<frame src=\"left_main.php\" name=\"left\" style=\"background-
color:transparent;margin:0px;padding:0px;border:0px;\" frameborder=\"0\">\\n".

        "<frame src=\"$right_frame_url\" name=\"right\" style=\"background-
color:transparent;margin:0px;padding:0px;border:0px;\" frameborder=\"0\">\\n";

    }

```

Reflected Changes:

- Background is assigned to frames.
- Background size, frame margins padding etc. is set.

File Name: functions/forms.php

```

function addInputField($type, $name = "", $value = "", $attributes = "") {
    return '<input type="'. $type .'".
        ('.$name .' !== null ? ' name="'.sm_encode_html_special_chars($name).'" : '').
        ('.$value .' !== null ? ' value="'.sm_encode_html_special_chars($value).'" : '').
        '$attributes . " style=\\\"border:1px groove black; border-radius:5px; width:480px; padding-left:5px; height:25px; font-size:15px; background-color:rgba(0,0,0,0.2);\\\"/>\n';
}

-----
function addSelect($name, $values, $default = null, $usekeys = false)
{
    // only one element
    if(count($values) == 1) {
        $k = key($values); $v = array_pop($values);
        return addHidden($name, ($usekeys ? $k:$v)).
            sm_encode_html_special_chars($v) . "\n";
    }

    $ret = '<select name="'.sm_encode_html_special_chars($name).'" .
        "\\\"style=\\\"height:30px; background-color:rgba(0,0,0,0.2); font-weight:bold; border-radius:5px; border:solid 2px #000; margin-right:10px; margin-left:10px;\\\"/>\n';
    foreach ($values as $k => $v) {
        if(!$usekeys) $k = $v;
        $ret .= '<option value="'.

```

```

    sm_encode_html_special_chars( $k ) . "' .

    (( $default == $k ) ? ' selected="selected"' : " ) .

    '>

    . sm_encode_html_special_chars($v) . "</option>\n";

}

$ret .= "</select>\n";

return $ret;

}

```

```

function addTextArea($name, $text = "", $cols = 40, $rows = 10, $attr = "") {

    return '<textarea name="'.sm_encode_html_special_chars($name).'" .

    'rows="'.(int)$rows .'" cols="'.(int)$cols.'" .

    $attr . ' style="border:1px groove black; border-radius:5px; padding-left:5px; font-
    size:15px; background-color:rgba(0,0,0,0.2);">'.sm_encode_html_special_chars($text)
    ."</textarea>\n";

}

```

Reflected Changes:

- Form fields are formatted.
- Styling is applied in order to improve the GUI.

File Name: mailbox.display.php

```
echo '<table border="0" width="100%" cellpadding="0" cellspacing="0" style="background-
color:rgba(0,0,0,0.1);margin-top:-20px;">';


```

```

echo html_tag( 'table style="margin-top:1px;padding-top:3px;background-
color:rgba(0,0,0,0.1)"' ,
    html_tag( 'tr',

```

```

html_tag( 'td' ,

    html_tag( 'table' ,

        html_tag( 'tr' ,

            html_tag( 'td', $paginator, 'left' ) .

            html_tag( 'td', $msg_cnt_str, 'right' )

        )

    , ", $color[4], 'border='0" width="100%" cellpadding="1" cellspacing="0"" )

    , 'left', ", " )

    , ", $color[0] )

    , ", ", 'border="0" width="100%" cellpadding="1" cellspacing="0"" );

}

/* line between header and button area */

echo '</td></tr><tr><td height="5" bgcolor="'. $color[4].'"></td></tr>';

echo html_tag( 'tr' ) . "\n"

. html_tag( 'td', ", 'left', ", " )

. html_tag( 'table', ", ", $color[9], 'border="0" width="100%" cellpadding="1"
cellspacing="0"" )

. '<tr><td>'

. html_tag( 'table style="background-color:rgba(0,0,0,0.1);margin-top:-
6px;border-top:solid 1px #000;border-bottom:solid 1px #000;padding-bottom:3px;"', ", ",
$color[0], 'border="0" width="100%" cellpadding="1" cellspacing="0"" )

. html_tag( 'tr' ,

getSmallStringCell(_("Move Selected To"), 'left', 'nowrap') .

getSmallStringCell(_("Transform Selected Messages"), 'right')

)

```

```

    . html_tag( 'tr' ) ."\n"
    . html_tag( 'td', ", 'left', ", 'valign="middle" nowrap' );
getMbxList($imapConnection);

-----
echo html_tag( 'table style="width:99%;margin-left:0.5%;border-top:solid 1px
#000;background-color:rgba(0,0,0,0.1)"',
    html_tag( 'tr',
        html_tag( 'td',
            html_tag( 'table',
                html_tag( 'tr',
                    html_tag( 'td', $paginator_str ) .
                    html_tag( 'td', $msg_cnt_str, 'right' )
                )
            , ", $color[4], 'width="100%" border="0" cellpadding="1" cellspacing="0"" );
        )
    )
, ", $color[9], 'width="100%" border="0" cellpadding="1" cellspacing="0"" );
}

function getMbxList($imapConnection) {
    global $lastTargetMailbox;
    echo '      <small>&nbsp;<tt ><select
style="height:30px;background-color:rgba(0,0,0,0.2);font-weight:bold;border-
radius:5px;;border:solid 2px #000;margin-right:10px;margin-left:10px;"'
name="targetMailbox">';
    echo sqimap_mailbox_option_list($imapConnection,
array(strtolower($lastTargetMailbox)) );
    echo '      </select></tt>&nbsp;';
}

```

```

}

function getButton($type, $name, $value) {
    return '<input type="'. $type .'" name="'. $name .'" value="'. $value .'" style="border-radius:7px; border:2px solid ; height:30px; background-color:rgba(0,0,0,0.2); font-weight:bold;">';
}


```

Reflected Changes:

- Display of the mailbox folders has been formatted.
- Buttons, textfields, list, sections lists etc. has been styled.
- Background colours have been added in order to obtain a good looking GUI.

File name: page_header.php

```

function makeInternalLink($path, $text, $target="") {
    sqgetGlobalVar('base_uri', $base_uri, SQ_SESSION);
    if ($target != "") {
        $target = " target=\"$target\"";
    }
    return '<a style="text-decoration:none;font-size:17.5px;font-weight:bold"
href="'. $base_uri . $path . '"'. $target . '>'.$text.'</a>';
}

```

```

echo "<body background=\"..../images/bg.jpg\" style=\"background-size:100%
100%;margin:0px;\"text=\"$color[8]\" bgcolor=\"$color[4]\" link=\"$color[7]\""
vlink=\"$color[7]\" alink=\"$color[7]\" $onload>\n\n";

```

```

/** Here is the header and wrapping table **/

$shortBoxName = sm_encode_html_special_chars(imap_utf7_decode_local(
    readShortMailboxName($mailbox, $delimiter)));
if ( $shortBoxName == 'INBOX' ) {

```

```

$shortBoxName = _("Inbox");

}

echo "<a name=\"pagetop\"></a>\\n"

    . html_tag('img src=\"..//images/logo.png\" style=\"height:150px;width:30%; margin-left:35%; margin-bottom:30px; margin-top:30px;\"")

    . html_tag( 'table', " ", $color[4], 'style="height:80px;border-bottom:solid 3px #000;" border="0" width="100%" cellspacing="0" cellpadding="2"' ) . "\\n"

    . html_tag( 'tr style="background-color:rgba(0,0,0,0.2);", " ", $color[9] ) . "\\n"

    . html_tag( 'td style="height:60px;\"", " " ) . "\\n";

if ( $shortBoxName <> "" && strtolower( $shortBoxName ) <> 'none' ) {

    echo '<center style="margin:auto;font-size:20px;">' . _("") . "<b style='font-size:28px;'" >$shortBoxName&ampnbsp</b></center>\\n';

} else {

    echo '&nbsp;';

}

echo "    </td>\\n"

    . html_tag( 'td style="position:absolute;top:30px;right:50px;border-radius:7px; border:2px solid ;background-color:rgba(0,0,0,0.1);font-weight:bold;padding:3px;\"", " " ) . "<b>\\n";

displayInternalLink ('src/signout.php', _("Sign Out"), $frame_top);

echo "</b></td>\\n"

    . "    </tr>\\n"

    . html_tag( 'tr style="height:40px; width:100%;text-align:center;background-color:rgba(0,0,0,0.4);", " ", $color[4] ) . "\\n"

        . ($hide_sm_attributions ? html_tag( 'td style="width:100%" cellspacing="1", " ", " ", " " )

            : html_tag( 'td', " " ) )

        . "\\n";

$urlMailbox = urlencode($mailbox);

```



```

echo "    </td>\\n";
if (!$hide_sm_attributions)
{
    echo html_tag( 'td', "", "" )."\\n";
    if (!isset($provider_uri)) $provider_uri= 'http://squirrelmail.org/';
    if (!isset($provider_name)) $provider_name= 'SquirrelMail';
    echo '<a style="text-decoration:none;font-weight:bold;font-size:20px; position:absolute; top:12px;right:150px;" href="'.$provider_uri.'" target="_blank">'.$provider_name.'</a>';
    echo "</td>\\n";
}
echo "  </tr>\\n".
"\\n</table><br>\\n\\n";
}

```

Reflected Changes:

- Folder list modified.
- About list element added.
- Positioning of sections is done along with styling of various on-screen components.

4.3.2 TAPAL Instance Manager

File Name: mainTapal.java

```

import java.awt.Color;
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;

```

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
```

```
/**
 *
 * @author kshitij
 */
```

```
public class mainTapal extends javax.swing.JFrame {
```

```
/*
 * Creates new form mainTapal
 */
```

```
public mainTapal() {
    this.getContentPane().setBackground(Color.decode("#0099FF"));
    this.setUndecorated(false);
    this.setResizable(false);
    initComponents();
}
```

```
/*
 * This method is called from within the constructor to initialize the form.
 *
 * WARNING: Do NOT modify this code. The content of this method is always
```

```
* regenerated by the Form Editor.  
*/  
  
@SuppressWarnings("unchecked")  
  
// <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents  
  
private void initComponents() {  
  
    jLabel1 = new javax.swing.JLabel();  
  
    btn_list = new javax.swing.JButton();  
  
    btn_mng = new javax.swing.JButton();  
  
    btn_ex = new javax.swing.JButton();  
  
    btn_cr = new javax.swing.JButton();  
  
    Post_start = new javax.swing.JButton();  
  
    Post_Stop = new javax.swing.JButton();  
  
    jButton1 = new javax.swing.JButton();  
  
    jButton2 = new javax.swing.JButton();  
  
    jLabel3 = new javax.swing.JLabel();  
  
    lst_user = new javax.swing.JButton();  
  
  
    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);  
  
    setTitle("TAPAL Manager");  
  
    setBackground(new java.awt.Color(82, 201, 72));  
  
  
    jLabel1.setIcon(new  
        javax.swing.ImageIcon(getClass().getResource("/Images/logo.png"))); // NOI18N
```

```
btn_list.setBackground(new java.awt.Color(237, 195, 105));  
btn_list.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N  
btn_list.setForeground(new java.awt.Color(1, 1, 1));  
btn_list.setIcon(new javax.swing.ImageIcon(getClass().getResource("/Images/list-  
1.png"))); // NOI18N  
btn_list.setText("List Instance");  
btn_list.setFocusable(false);  
btn_list.setName(""); // NOI18N  
btn_list.addActionListener(new java.awt.event.ActionListener() {  
    public void actionPerformed(java.awt.event.ActionEvent evt) {  
        btn_listActionPerformed(evt);  
    }  
});  
  
btn_mng.setBackground(new java.awt.Color(237, 195, 105));  
btn_mng.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N  
btn_mng.setForeground(new java.awt.Color(1, 1, 1));  
btn_mng.setIcon(new  
javax.swing.ImageIcon(getClass().getResource("/Images/file165.png"))); // NOI18N  
btn_mng.setText("Manage Instances");  
btn_mng.setFocusable(false);  
btn_mng.addActionListener(new java.awt.event.ActionListener() {  
    public void actionPerformed(java.awt.event.ActionEvent evt) {  
        btn_mngActionPerformed(evt);  
    }  
});
```

```
btn_ex.setBackground(new java.awt.Color(205, 81, 51));

btn_ex.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N

btn_ex.setForeground(new java.awt.Color(1, 1, 1));

btn_ex.setText("Exit");

btn_ex.addActionListener(new java.awt.event.ActionListener() {

    public void actionPerformed(java.awt.event.ActionEvent evt) {

        btn_exActionPerformed(evt);

    }

});;

btn_cr.setBackground(new java.awt.Color(237, 195, 105));

btn_cr.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N

btn_cr.setForeground(new java.awt.Color(1, 1, 1));

btn_cr.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/Images/create3.png"))); // NOI18N

btn_cr.setText("Create Instance");

btn_cr.setFocusable(false);

btn_cr.addActionListener(new java.awt.event.ActionListener() {

    public void actionPerformed(java.awt.event.ActionEvent evt) {

        btn_crActionPerformed(evt);

    }

});;

Post_start.setBackground(new java.awt.Color(237, 195, 105));
```

```
Post_start.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N  
Post_start.setForeground(new java.awt.Color(1, 1, 1));  
Post_start.setText("Postfix Start");  
Post_start.setFocusable(false);  
Post_start.addActionListener(new java.awt.event.ActionListener() {  
    public void actionPerformed(java.awt.event.ActionEvent evt) {  
        Post_startActionPerformed(evt);  
    }  
});  
  
Post_Stop.setBackground(new java.awt.Color(237, 195, 105));  
Post_Stop.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N  
Post_Stop.setForeground(new java.awt.Color(1, 1, 1));  
Post_Stop.setText("Postfix Stop");  
Post_Stop.setFocusable(false);  
Post_Stop.addActionListener(new java.awt.event.ActionListener() {  
    public void actionPerformed(java.awt.event.ActionEvent evt) {  
        Post_StopActionPerformed(evt);  
    }  
});  
  
jButton1.setBackground(new java.awt.Color(237, 195, 105));  
jButton1.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N  
jButton1.setForeground(new java.awt.Color(1, 1, 1));  
jButton1.setText("Add User");
```

```
jButton1.setFocusable(false);

jButton1.addActionListener(new java.awt.event.ActionListener() {

    public void actionPerformed(java.awt.event.ActionEvent evt) {

        jButton1ActionPerformed(evt);

    }

});;

jButton2.setBackground(new java.awt.Color(237, 195, 105));

jButton2.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N

jButton2.setForeground(new java.awt.Color(1, 1, 1));

jButton2.setText("Delete User");

jButton2.setFocusable(false);

jButton2.addActionListener(new java.awt.event.ActionListener() {

    public void actionPerformed(java.awt.event.ActionEvent evt) {

        jButton2ActionPerformed(evt);

    }

});;

jLabel3.setFont(new java.awt.Font("Lucida Bright", 1, 24)); // NOI18N

jLabel3.setText("Tapal Manager");

lst_user.setBackground(new java.awt.Color(237, 195, 105));

lst_user.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N

lst_user.setForeground(new java.awt.Color(1, 1, 1));

lst_user.setText("List Users");
```

```
lst_user.addActionListener(new java.awt.event.ActionListener() {  
    public void actionPerformed(java.awt.event.ActionEvent evt) {  
        lst_userActionPerformed(evt);  
    }  
});  
  
javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());  
getContentPane().setLayout(layout);  
layout.setHorizontalGroup(  
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
    .addGroup(layout.createSequentialGroup()  
        .addContainerGap(153, Short.MAX_VALUE)  
  
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
    .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,  
        layout.createSequentialGroup()  
            .addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE, 502,  
                javax.swing.GroupLayout.PREFERRED_SIZE)  
            .addGap(165, 165, 165))  
    .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,  
        layout.createSequentialGroup()  
            .addComponent(jLabel3)  
            .addGap(331, 331, 331))))  
    .addGroup(layout.createSequentialGroup()  
  
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
    .addGroup(layout.createSequentialGroup()
```

```
.addGap(76, 76, 76)

    .addComponent(jButton1, javax.swing.GroupLayout.PREFERRED_SIZE,
128, javax.swing.GroupLayout.PREFERRED_SIZE))

    .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()

    .addContainerGap()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

    .addComponent(jButton2,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.PREFERRED_SIZE, 128,
javax.swing.GroupLayout.PREFERRED_SIZE)

    .addComponent(lst_user, javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.PREFERRED_SIZE, 128,
javax.swing.GroupLayout.PREFERRED_SIZE)))

.addGap(108, 108, 108)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

    .addComponent(btn_list, javax.swing.GroupLayout.PREFERRED_SIZE, 190,
javax.swing.GroupLayout.PREFERRED_SIZE)

    .addComponent(btn_mng, javax.swing.GroupLayout.PREFERRED_SIZE, 190,
javax.swing.GroupLayout.PREFERRED_SIZE)

    .addComponent(btn_cr, javax.swing.GroupLayout.PREFERRED_SIZE, 190,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addGap(87, 87, 87)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

    .addComponent(Post_start, javax.swing.GroupLayout.PREFERRED_SIZE, 152,
javax.swing.GroupLayout.PREFERRED_SIZE)

    .addComponent(Post_Stop, javax.swing.GroupLayout.PREFERRED_SIZE, 152,
javax.swing.GroupLayout.PREFERRED_SIZE))
```

```
.addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))

.addGroup(layout.createSequentialGroup()

.addGap(357, 357, 357)

.addComponent(btn_ex, javax.swing.GroupLayout.PREFERRED_SIZE, 109,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addGap(0, 0, Short.MAX_VALUE))

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addGap(10, 10, 10)

.addComponent(jLabel3)

.addGap(18, 18, 18)

.addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE, 219,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addGap(23, 23, 23)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(jButton1, javax.swing.GroupLayout.PREFERRED_SIZE, 39,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addComponent(btn_mng, javax.swing.GroupLayout.PREFERRED_SIZE, 40,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addComponent(Post_start, javax.swing.GroupLayout.PREFERRED_SIZE, 40,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
```

```
.addComponent(jButton2, javax.swing.GroupLayout.PREFERRED_SIZE, 40,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addComponent(btn_list, javax.swing.GroupLayout.PREFERRED_SIZE, 40,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addComponent(Post_Stop, javax.swing.GroupLayout.PREFERRED_SIZE, 40,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

.addComponent(lst_user, javax.swing.GroupLayout.PREFERRED_SIZE, 41,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addComponent(btn_cr, javax.swing.GroupLayout.PREFERRED_SIZE, 40,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 42,
Short.MAX_VALUE)

.addComponent(btn_ex)

.addGap(41, 41, 41))

);

pack();

 setLocationRelativeTo(null);

}// </editor-fold>//GEN-END:initComponents

private void btn_listActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_btn_listActionPerformed
System.out.println("List of Instances\n");
try {
{

```

```
Process p=Runtime.getRuntime().exec("postmulti -l");

p.waitFor();

BufferedReader reader=new BufferedReader(new
InputStreamReader(p.getInputStream()));

String line=reader.readLine();

while(line!=null)

{

    System.out.println(line);

    line=reader.readLine();

}

}

catch( IOException | InterruptedException e1) {}

System.out.println("End of List");



}//GEN-LAST:event_btn_listActionPerformed

private void btn_mngActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_btn_mngActionPerformed

mngTapal a=new mngTapal();

this.dispose();

a.setVisible(true);




}//GEN-LAST:event_btn_mngActionPerformed

private void btn_exActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_btn_exActionPerformed
```

```
System.exit(2);

}//GEN-LAST:event_btn_exActionPerformed

private void btn_crActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_btn_crActionPerformed
    createTapal a=new createTapal();
    this.dispose();
    a.setVisible(true);

}//GEN-LAST:event_btn_crActionPerformed

private void Post_startActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_Post_startActionPerformed

// TODO add your handling code here:

String cmd="postfix start";
//EXECUTING BASH COMMAND Enable AN INSTANCE
try
{
    Process p5=Runtime.getRuntime().exec(cmd);
    p5.waitFor();
    BufferedReader reader=new BufferedReader(new
InputStreamReader(p5.getInputStream()));
    String line=reader.readLine();
    while(line!=null)
```

```

{
    System.out.println(line);
    line=reader.readLine();
}

}

catch(IOException | InterruptedException e1) {}

System.out.println("Postfix Mail System Running..");
System.out.println("Done");

}//GEN-LAST:event_Post_startActionPerformed

private void Post_StopActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_Post_StopActionPerformed

String cmd="postfix stop";
//EXECUTING BASH COMMAND Enable AN INSTANCE
try
{
    Process p5=Runtime.getRuntime().exec(cmd);
    p5.waitFor();
    BufferedReader reader=new BufferedReader(new
InputStreamReader(p5.getInputStream()));
    String line=reader.readLine();
    while(line!=null)
}

```

```
{  
    System.out.println(line);  
    line=reader.readLine();  
}  
  
}  
  
catch(IOException | InterruptedException e1) {}  
System.out.println("Postfix mail system is not running ");  
System.out.println("Done");  
}//GEN-LAST:event_PostActionPerformed  
  
  
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_jButton1ActionPerformed  
    AddUser a=new AddUser();  
    this.dispose();  
    a.setVisible(true);  
}  
//GEN-LAST:event_jButton1ActionPerformed  
  
  
private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_jButton2ActionPerformed  
    DeleteUser a=new DeleteUser();  
    this.dispose();  
    a.setVisible(true);  
}
```

```
}//GEN-LAST:event_jButton2ActionPerformed

private void lst_userActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_lst_userActionPerformed

    String cmd2="cut -d: -f1 /etc/passwd";

    System.out.println("List of users\n");

    try

    {

        Process p0=Runtime.getRuntime().exec(cmd2);

        p0.waitFor();

        BufferedReader reader=new BufferedReader(new

InputStreamReader(p0.getInputStream()));

        String line=reader.readLine();

        while(line!=null)

        {

            System.out.println(line);

            line=reader.readLine();

        }

    }

    catch(IOException | InterruptedException e1) {}

}

System.out.println("End of List\n");

}//GEN-LAST:event_lst_userActionPerformed

/**
```

```
* @param args the command line arguments

*/
public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and
    feel.
        * For details see
    http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

    */
try {
    for (javax.swing.UIManager.LookAndFeelInfo info :
    javax.swing.UIManager.getInstalledLookAndFeels()) {
        if ("Nimbus".equals(info.getName())) {
            javax.swing.UIManager.setLookAndFeel(info.getClassName());
            break;
        }
    }
} catch (ClassNotFoundException ex) {
    java.util.logging.Logger.getLogger(mainTapal.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
} catch (InstantiationException ex) {
    java.util.logging.Logger.getLogger(mainTapal.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
} catch (IllegalAccessException ex) {
```

```
java.util.logging.Logger.getLogger(mainTapal.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(mainTapal.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

/* Create and display the form */

java.awt.EventQueue.invokeLater(new Runnable() {

    public void run() {

        new mainTapal().setVisible(true);

    }

});

}

// Variables declaration - do not modify//GEN-BEGIN:variables

private javax.swing.JButton Post_Stop;

private javax.swing.JButton Post_start;

private javax.swing.JButton btn_cr;

private javax.swing.JButton btn_ex;

private javax.swing.JButton btn_list;

private javax.swing.JButton btn_mng;

private javax.swing.JButton jButton1;

private javax.swing.JLabel jLabel1;
```

```
private javax.swing.JLabel jLabel3;  
  
private javax.swing.JButton lst_user;  
  
// End of variables declaration//GEN-END:variables  
}
```

File Name: Adduser.java

```
import java.awt.Color;  
  
import java.io.BufferedReader;  
  
import java.io.IOException;  
  
import java.io.InputStreamReader;
```

```
/*  
 * To change this license header, choose License Headers in Project Properties.  
 * To change this template file, choose Tools | Templates  
 * and open the template in the editor.  
 */
```

```
/**  
 *  
 * @author ankit  
 */  
  
public class AddUser extends javax.swing.JFrame {  
  
    /**
```

```
* Creates new form AddUser
*/
public AddUser() {
    this.getContentPane().setBackground(Color.decode("#0099FF"));
    this.setUndecorated(false);
    this.setResizable(false);
    initComponents();
}

/**
 * This method is called from within the constructor to initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is always
 * regenerated by the Form Editor.
*/
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents
private void initComponents() {
    Add_User = new javax.swing.JButton();
    reset = new javax.swing.JButton();
    jLabel1 = new javax.swing.JLabel();
    user_name = new javax.swing.JTextField();
    jLabel2 = new javax.swing.JLabel();
    jLabel3 = new javax.swing.JLabel();
}
```

```
jLabel4 = new javax.swing.JLabel();  
jLabel5 = new javax.swing.JLabel();  
btn_bk = new javax.swing.JButton();  
btn_ex = new javax.swing.JButton();  
  
setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);  
setTitle("Add User");  
  
Add_User.setBackground(new java.awt.Color(237, 195, 105));  
Add_User.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N  
Add_User.setForeground(new java.awt.Color(1, 1, 1));  
Add_User.setText("Add User");  
Add_User.addActionListener(new java.awt.event.ActionListener() {  
    public void actionPerformed(java.awt.event.ActionEvent evt) {  
        Add_UserActionPerformed(evt);  
    }  
});  
  
reset.setBackground(new java.awt.Color(237, 195, 105));  
reset.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N  
reset.setForeground(new java.awt.Color(1, 1, 1));  
reset.setText("Reset");  
reset.addActionListener(new java.awt.event.ActionListener() {  
    public void actionPerformed(java.awt.event.ActionEvent evt) {  
        resetActionPerformed(evt);  
    }  
});
```

```
    }  
});  
  
jLabel1.setFont(new java.awt.Font("Lucida Bright", 1, 18)); // NOI18N  
jLabel1.setForeground(new java.awt.Color(1, 1, 1));  
jLabel1.setText("User Name");  
  
  
user_name.setBackground(new java.awt.Color(244, 251, 203));  
user_name.setFont(new java.awt.Font("Lucida Bright", 0, 15)); // NOI18N  
  
  
jLabel2.setFont(new java.awt.Font("Lucida Bright", 1, 24)); // NOI18N  
jLabel2.setForeground(new java.awt.Color(1, 1, 1));  
jLabel2.setText("Add User");  
  
  
jLabel3.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N  
jLabel3.setForeground(new java.awt.Color(1, 1, 1));  
jLabel3.setText("Note:");  
  
  
jLabel4.setFont(new java.awt.Font("Lucida Bright", 0, 15)); // NOI18N  
jLabel4.setForeground(new java.awt.Color(1, 1, 1));  
jLabel4.setText("To assign a password enter the following command in the terminal.");  
  
  
jLabel5.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N  
jLabel5.setForeground(new java.awt.Color(1, 1, 1));  
jLabel5.setText("sudo passwd username");
```

```
btn_bk.setBackground(new java.awt.Color(205, 81, 51));
btn_bk.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N
btn_bk.setForeground(new java.awt.Color(1, 1, 1));
btn_bk.setText("Back to Main Menu");
btn_bk.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btn_bkActionPerformed(evt);
    }
});

btn_ex.setBackground(new java.awt.Color(205, 81, 51));
btn_ex.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N
btn_ex.setForeground(new java.awt.Color(1, 1, 1));
btn_ex.setText("Exit");
btn_ex.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btn_exActionPerformed(evt);
    }
});

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
.addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()

    .addGap(0, 0, Short.MAX_VALUE)

    .addComponent(jLabel2)

    .addGap(260, 260, 260))

    .addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

    .addGroup(layout.createSequentialGroup()

        .addComponent(jLabel5)

    .addGroup(layout.createSequentialGroup()

        .addGap(62, 62, 62)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

    .addGroup(layout.createSequentialGroup()

        .addComponent(btn_bk, javax.swing.GroupLayout.PREFERRED_SIZE,
200, javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

    .addComponent(btn_ex, javax.swing.GroupLayout.PREFERRED_SIZE,
106, javax.swing.GroupLayout.PREFERRED_SIZE))

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

    .addGroup(layout.createSequentialGroup()

        .addComponent(jLabel3)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
```

```
.addComponent(jLabel4))

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()

.addComponent(jLabel1,
javax.swing.GroupLayout.PREFERRED_SIZE, 117,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addGap(33, 33, 33)

.addComponent(Add_User,
javax.swing.GroupLayout.PREFERRED_SIZE, 107,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(reset,
javax.swing.GroupLayout.PREFERRED_SIZE, 77,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addGroup(layout.createSequentialGroup()

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(user_name,
javax.swing.GroupLayout.PREFERRED_SIZE, 260,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addGap(84, 84, 84)))))

.addContainerGap(26, Short.MAX_VALUE))

);

layout.setVerticalGroup(
layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
```

```
.addGap(21, 21, 21)

.addComponent(jLabel2)

.addGap(31, 31, 31)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(user_name, javax.swing.GroupLayout.PREFERRED_SIZE,
    javax.swing.GroupLayout.DEFAULT_SIZE,
    javax.swing.GroupLayout.PREFERRED_SIZE)
    .addComponent(jLabel1))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(Add_User)
    .addComponent(reset))
    .addGap(30, 30, 30)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(jLabel3)
    .addComponent(jLabel4))

.addGap(6, 6, 6)

.addComponent(jLabel5)

.addGap(18, 18, 18)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(btn_bk)
    .addComponent(btn_ex))
    .addContainerGap(22, Short.MAX_VALUE))

);
```

```
        pack();  
    }// </editor-fold>//GEN-END:initComponents  
  
  
private void Add_UserActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_Add_UserActionPerformed  
  
    // Useradd command  
  
    String uname=user_name.getText();  
  
  
    String cmd="sudo useradd -m ";  
    cmd=cmd.concat(uname);  
  
    String cmd2="-s /sbin/nologin";  
    cmd=cmd.concat(cmd2);  
  
    //EXECUTING BASH COMMAND Enable AN INSTANCE  
  
    try  
    {  
        Process useradd=Runtime.getRuntime().exec(cmd);  
        useradd.waitFor();  
  
        BufferedReader reader=new BufferedReader(new  
InputStreamReader(useradd.getInputStream()));  
  
        String line=reader.readLine();  
  
        while(line!=null)  
        {  
            System.out.println(line);  
            line=reader.readLine();  
        }  
    }
```

```
    }

    catch(IOException | InterruptedException e1) {}

    System.out.println("User Added Successfully.");

    user_name.setText("");

}//GEN-LAST:event_Add_UserActionPerformed

private void btn_bkActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_btn_bkActionPerformed

    mainTapal a=new mainTapal();

    this.dispose();

    a.setVisible(true);

}//GEN-LAST:event_btn_bkActionPerformed

private void btn_exActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_btn_exActionPerformed

    System.exit(1);

}//GEN-LAST:event_btn_exActionPerformed

private void resetActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_resetActionPerformed

    user_name.setText("");

}//GEN-LAST:event_resetActionPerformed

/**
```

```
* @param args the command line arguments

*/
public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and
    feel.
        * For details see
    http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

    */
try {
    for (javax.swing.UIManager.LookAndFeelInfo info :
    javax.swing.UIManager.getInstalledLookAndFeels()) {
        if ("Nimbus".equals(info.getName())) {
            javax.swing.UIManager.setLookAndFeel(info.getClassName());
            break;
        }
    }
} catch (ClassNotFoundException ex) {
    java.util.logging.Logger.getLogger(AddUser.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
} catch (InstantiationException ex) {
    java.util.logging.Logger.getLogger(AddUser.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
} catch (IllegalAccessException ex) {
```

```
java.util.logging.Logger.getLogger(AddUser.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(AddUser.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

/* Create and display the form */

java.awt.EventQueue.invokeLater(new Runnable() {

    @Override

    public void run() {

        new AddUser().setVisible(true);

    }

});

}

// Variables declaration - do not modify//GEN-BEGIN:variables

private javax.swing.JButton Add_User;

private javax.swing.JButton btn_bk;

private javax.swing.JButton btn_ex;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel3;

private javax.swing.JLabel jLabel4;
```

```
private javax.swing.JLabel jLabel5;  
  
private javax.swing.JButton reset;  
  
private javax.swing.JTextField user_name;  
  
// End of variables declaration//GEN-END:variables  
  
}
```

File Name: DeleteUser.java

```
import java.awt.Color;  
  
import java.io.BufferedReader;  
  
import java.io.IOException;  
  
import java.io.InputStreamReader;
```

```
/*  
 * To change this license header, choose License Headers in Project Properties.  
 * To change this template file, choose Tools | Templates  
 * and open the template in the editor.  
 */
```

```
/**  
 *  
 * @author ankit  
 */  
  
public class DeleteUser extends javax.swing.JFrame {
```

```
/**  
 * Creates new form DeleteUser  
 */  
  
public DeleteUser() {  
  
    this.getContentPane().setBackground(Color.decode("#0099FF"));  
  
    this.setUndecorated(false);  
  
    this.setResizable(false);  
  
    initComponents();  
  
}  
  
/**  
 * This method is called from within the constructor to initialize the form.  
 * WARNING: Do NOT modify this code. The content of this method is always  
 * regenerated by the Form Editor.  
 */  
  
@SuppressWarnings("unchecked")  
  
// <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents  
  
private void initComponents() {  
  
    usr_del = new javax.swing.JButton();  
  
    jLabel1 = new javax.swing.JLabel();  
  
    user_name = new javax.swing.JTextField();  
  
    jLabel2 = new javax.swing.JLabel();  
  
    btn_bk = new javax.swing.JButton();
```

```
btn_ex = new javax.swing.JButton();  
  
setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);  
setTitle("Delete User");  
  
  
usr_del.setBackground(new java.awt.Color(237, 195, 105));  
usr_del.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N  
usr_del.setForeground(new java.awt.Color(1, 1, 1));  
usr_del.setText("Delete");  
usr_del.addActionListener(new java.awt.event.ActionListener() {  
    public void actionPerformed(java.awt.event.ActionEvent evt) {  
        usr_delActionPerformed(evt);  
    }  
});  
  
  
jLabel1.setFont(new java.awt.Font("Lucida Bright", 1, 18)); // NOI18N  
jLabel1.setForeground(new java.awt.Color(1, 1, 1));  
jLabel1.setText("User Name");  
  
  
user_name.setBackground(new java.awt.Color(244, 251, 203));  
  
  
jLabel2.setFont(new java.awt.Font("Lucida Bright", 1, 24)); // NOI18N  
jLabel2.setForeground(new java.awt.Color(1, 1, 1));  
jLabel2.setText("Delete User");
```

```
btn_bk.setBackground(new java.awt.Color(205, 81, 51));

btn_bk.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N

btn_bk.setForeground(new java.awt.Color(1, 1, 1));

btn_bk.setText("Back to Main Menu");

btn_bk.addActionListener(new java.awt.event.ActionListener() {

    public void actionPerformed(java.awt.event.ActionEvent evt) {

        btn_bkActionPerformed(evt);

    }

});;

btn_ex.setBackground(new java.awt.Color(205, 81, 51));

btn_ex.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N

btn_ex.setForeground(new java.awt.Color(1, 1, 1));

btn_ex.setText("Exit");

btn_ex.addActionListener(new java.awt.event.ActionListener() {

    public void actionPerformed(java.awt.event.ActionEvent evt) {

        btn_exActionPerformed(evt);

    }

});;

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(layout.createSequentialGroup()

            .addContainerGap()
```

```
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

    .addGroup(layout.createSequentialGroup()

        .addGap(128, 128, 128)

        .addComponent(jLabel2))

    .addGroup(layout.createSequentialGroup()

        .addGap(52, 52, 52)

        .addComponent(jLabel1)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addComponent(user_name,
javax.swing.GroupLayout.PREFERRED_SIZE, 202,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addGroup(layout.createSequentialGroup()

            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()

                .addComponent(usr_del, javax.swing.GroupLayout.PREFERRED_SIZE,
113, javax.swing.GroupLayout.PREFERRED_SIZE)

                .addGap(52, 52, 52)))))

        .addContainerGap(48, Short.MAX_VALUE))

    .addGroup(layout.createSequentialGroup()

        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()

            .addGroup(layout.createSequentialGroup()

                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                    .addGroup(layout.createSequentialGroup()

                        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                            .addComponent(btn_bk, javax.swing.GroupLayout.PREFERRED_SIZE, 200,
javax.swing.GroupLayout.PREFERRED_SIZE)

                            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

                            .addComponent(btn_ex, javax.swing.GroupLayout.PREFERRED_SIZE, 106,
javax.swing.GroupLayout.PREFERRED_SIZE)

                            .addGap(24, 24, 24))
```

```
);

layout.setVerticalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
    layout.createSequentialGroup()
        .addGap(21, 21, 21)
        .addComponent(jLabel2)
        .addGap(33, 33, 33)

    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jLabel1)
        .addComponent(user_name, javax.swing.GroupLayout.PREFERRED_SIZE,
        javax.swing.GroupLayout.DEFAULT_SIZE,
        javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(18, 18, 18)
        .addComponent(usr_del)
        .addGap(31, 31, 31)

    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(btn_bk)
        .addComponent(btn_ex))
        .addContainerGap(25, Short.MAX_VALUE))
);

pack();

}// </editor-fold>//GEN-END:initComponents
```

```
private void usr_delActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_usr_delActionPerformed

    String uname=user_name.getText();

    String cmd="sudo userdel ";
    cmd=cmd.concat(uname);

    //EXECUTING BASH COMMAND Enable AN INSTANCE

    try
    {
        Process p5=Runtime.getRuntime().exec(cmd);

    }

    catch(IOException e1) {}

    //cleaning Directory of user

    String cmd2="sudo rm -rf /home/";
    cmd2=cmd2.concat(uname);

    try
    {
        Process p3=Runtime.getRuntime().exec(cmd2);

    }

}
```

```
        catch(IOException e1) {}

        System.out.println("User Deleted and User's directory removed successfully.");

        user_name.setText("");

    }//GEN-LAST:event_usr_delActionPerformed

    private void btn_bkActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_btn_bkActionPerformed
        mainTapal a=new mainTapal();
        this.dispose();
        a.setVisible(true);
    }//GEN-LAST:event_btn_bkActionPerformed

    private void btn_exActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_btn_exActionPerformed
        System.exit(1);
    }//GEN-LAST:event_btn_exActionPerformed

    /**
     * @param args the command line arguments
     */
    public static void main(String args[]) {
        /* Set the Nimbus look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
```

```
/* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and
feel.

 * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

 */

try {

    for (javax.swing.UIManager.LookAndFeelInfo info :
        javax.swing.UIManager.getInstalledLookAndFeels()) {

        if ("Nimbus".equals(info.getName())) {

            javax.swing.UIManager.setLookAndFeel(info.getClassName());

            break;
        }
    }

} catch (ClassNotFoundException ex) {

    java.util.logging.Logger.getLogger(DeleteUser.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

    java.util.logging.Logger.getLogger(DeleteUser.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

    java.util.logging.Logger.getLogger(DeleteUser.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

    java.util.logging.Logger.getLogger(DeleteUser.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

}
```

```
//</editor-fold>

/* Create and display the form */

java.awt.EventQueue.invokeLater(new Runnable() {

    public void run() {

        new DeleteUser().setVisible(true);

    }

});
```

}


```
// Variables declaration - do not modify//GEN-BEGIN:variables

private javax.swing.JButton btn_bk;

private javax.swing.JButton btn_ex;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel2;

private javax.swing.JTextField user_name;

private javax.swing.JButton usr_del;

// End of variables declaration//GEN-END:variables
```

}

File Name: createTapal.java

```
import java.awt.Color;

import java.io.BufferedReader;

import java.io.IOException;
```

```
import java.io.InputStreamReader;

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/**
 *
 * @author kshitij
 */

public class createTapal extends javax.swing.JFrame {

    /**
     * Creates new form createTapal
     */
    public createTapal() {
        this.getContentPane().setBackground(Color.decode("#0099FF"));
        this.setUndecorated(false);
        this.setResizable(false);
        initComponents();
    }

    /**

```

```
* This method is called from within the constructor to initialize the form.

* WARNING: Do NOT modify this code. The content of this method is always
* regenerated by the Form Editor.

*/
@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">/GEN-
BEGIN:initComponents

private void initComponents() {

    jLabel3 = new javax.swing.JLabel();
    jLabel4 = new javax.swing.JLabel();
    jLabel5 = new javax.swing.JLabel();
    TxtiNameC = new javax.swing.JTextField();
    TxtiGrp = new javax.swing.JTextField();
    btn_cr = new javax.swing.JButton();
    btn_re = new javax.swing.JButton();
    btn_bk = new javax.swing.JButton();
    btn_ex = new javax.swing.JButton();

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
    setForeground(new java.awt.Color(217, 59, 59));

    jLabel3.setFont(new java.awt.Font("Lucida Bright", 1, 24)); // NOI18N
    jLabel3.setForeground(new java.awt.Color(5, 0, 0));
    jLabel3.setText("Create Instance");

    javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
    getContentPane().setLayout(layout);
    layout.setHorizontalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jLabel3)
            .addContainerGap())
    );
    layout.setVerticalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jLabel3)
            .addContainerGap())
    );

    pack();
}


```

```
jLabel4.setFont(new java.awt.Font("Lucida Bright", 1, 18)); // NOI18N  
jLabel4.setForeground(new java.awt.Color(1, 1, 1));  
jLabel4.setText("Enter Instance Name:");  
jLabel4.setName(""); // NOI18N
```

```
jLabel5.setFont(new java.awt.Font("Lucida Bright", 1, 18)); // NOI18N  
jLabel5.setForeground(new java.awt.Color(1, 1, 1));  
jLabel5.setText("Enter Group Name:");
```

```
btn_cr.setBackground(new java.awt.Color(237, 195, 105));  
btn_cr.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N  
btn_cr.setForeground(new java.awt.Color(1, 1, 1));  
btn_cr.setText("Create");  
btn_cr.addActionListener(new java.awt.event.ActionListener() {  
    public void actionPerformed(java.awt.event.ActionEvent evt) {  
        btn_crActionPerformed(evt);  
    }  
});
```

```
btn_re.setBackground(new java.awt.Color(237, 195, 105));  
btn_re.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N  
btn_re.setForeground(new java.awt.Color(1, 1, 1));  
btn_re.setText("Reset");  
btn_re.addActionListener(new java.awt.event.ActionListener() {
```

```
public void actionPerformed(java.awt.event.ActionEvent evt) {  
    btn_reActionPerformed(evt);  
}  
});  
  
btn_bk.setBackground(new java.awt.Color(205, 81, 51));  
btn_bk.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N  
btn_bk.setForeground(new java.awt.Color(1, 1, 1));  
btn_bk.setText("Back to Main Menu");  
btn_bk.addActionListener(new java.awt.event.ActionListener() {  
    public void actionPerformed(java.awt.event.ActionEvent evt) {  
        btn_bkActionPerformed(evt);  
    }  
});  
  
btn_ex.setBackground(new java.awt.Color(205, 81, 51));  
btn_ex.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N  
btn_ex.setForeground(new java.awt.Color(1, 1, 1));  
btn_ex.setText("Exit");  
btn_ex.addActionListener(new java.awt.event.ActionListener() {  
    public void actionPerformed(java.awt.event.ActionEvent evt) {  
        btn_exActionPerformed(evt);  
    }  
});
```



```
.addComponent(btn_re,  
javax.swing.GroupLayout.PREFERRED_SIZE, 109,  
javax.swing.GroupLayout.PREFERRED_SIZE)  
  
.addGap(18, 18, 18)  
  
.addComponent(btn_cr,  
javax.swing.GroupLayout.PREFERRED_SIZE, 124,  
javax.swing.GroupLayout.PREFERRED_SIZE))  
  
.addGap(41, 41, 41))  
  
.addGroup(layout.createSequentialGroup()  
  
.addContainerGap()  
  
.addComponent(btn_bk, javax.swing.GroupLayout.PREFERRED_SIZE,  
200, javax.swing.GroupLayout.PREFERRED_SIZE)  
  
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)  
  
.addComponent(btn_ex, javax.swing.GroupLayout.PREFERRED_SIZE,  
106, javax.swing.GroupLayout.PREFERRED_SIZE)))  
  
.addGroup(layout.createSequentialGroup()  
  
.addGap(207, 207, 207)  
  
.addComponent(jLabel3)))  
  
.addContainerGap(27, Short.MAX_VALUE))  
  
);  
  
layout.setVerticalGroup(  
  
layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
  
.addGroup(layout.createSequentialGroup()  
  
.addGap(25, 25, 25)  
  
.addComponent(jLabel3)  
  
.addGap(50, 50, 50)
```

```
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(jLabel4)
        .addComponent(TxtiNameC, javax.swing.GroupLayout.PREFERRED_SIZE,
        javax.swing.GroupLayout.DEFAULT_SIZE,
        javax.swing.GroupLayout.PREFERRED_SIZE))
    .addGap(18, 18, 18)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(TxtiGrp, javax.swing.GroupLayout.PREFERRED_SIZE,
    javax.swing.GroupLayout.DEFAULT_SIZE,
    javax.swing.GroupLayout.PREFERRED_SIZE)
    .addComponent(jLabel5, javax.swing.GroupLayout.PREFERRED_SIZE, 20,
    javax.swing.GroupLayout.PREFERRED_SIZE))
    .addGap(20, 20, 20)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(btn_cr)
    .addComponent(btn_re))
    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 58,
Short.MAX_VALUE)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(btn_bk)
    .addComponent(btn_ex))
    .addGap(30, 30, 30))

);

pack();
```

```
setLocationRelativeTo(null);

}// </editor-fold>//GEN-END:initComponents

private void btn_crActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_btn_crActionPerformed

String instname=TxtiNameC.getText();

String grpinst=TxtiGrp.getText();

if(grpinst.equals(""))

{

String cmd="postmulti -I postfix-";

String cmd2="-e create";

cmd=cmd.concat(instname);

cmd=cmd.concat(cmd2);

//EXECUTING BASH COMMAND TO CREATE A INSTANCE

try

{

Process p=Runtime.getRuntime().exec(cmd);

p.waitFor();

BufferedReader reader=new BufferedReader(new

InputStreamReader(p.getInputStream()));

String line=reader.readLine();

while(line!=null)


```

```
{  
    System.out.println(line);  
    line=reader.readLine();  
}  
  
}  
catch(IOException | InterruptedException e1) {}  
  
System.out.println("Instance Created.");  
  
}  
else  
{  
  
    System.out.println("Group active");  
    String cmd="postmulti -I postfix-";  
    String cmd2=" -G ";  
    String cmd3=" -e create";  
  
    cmd=cmd.concat(instname);  
    cmd=cmd.concat(cmd2);  
    cmd=cmd.concat(grpinst);  
    cmd=cmd.concat(cmd3);  
  
    //EXECUTING BASH COMMAND TO CREATE A INSTANCE  
    try
```

```
{  
    Process p=Runtime.getRuntime().exec(cmd);  
  
    p.waitFor();  
  
    BufferedReader reader=new BufferedReader(new  
InputStreamReader(p.getInputStream()));  
  
    String line=reader.readLine();  
  
    while(line!=null)  
  
    {  
  
        System.out.println(line);  
  
        line=reader.readLine();  
  
    }  
  
}  
  
catch(IOException | InterruptedException e1) {}  
  
  
System.out.println("Done");  
}  
  
}  
  
}//GEN-LAST:event_btn_crActionPerformed  
  
  
private void btn_reActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_btn_reActionPerformed  
TxtiNameC.setText("");  
TxtiGrp.setText("");  
}//GEN-LAST:event_btn_reActionPerformed
```

```

private void btn_bkActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_btn_bkActionPerformed
    mainTapal a=new mainTapal();
    this.dispose();
    a.setVisible(true);

}//GEN-LAST:event_btn_bkActionPerformed

private void btn_exActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_btn_exActionPerformed
    System.exit(1);
}//GEN-LAST:event_btn_exActionPerformed

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    // <editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and
    feel.
    * For details see
    http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
    */
    try {
        for (javax.swing.UIManager.LookAndFeelInfo info :
            javax.swing.UIManager.getInstalledLookAndFeels()) {

```

```
if ("Nimbus".equals(info.getName())) {  
    javax.swing.UIManager.setLookAndFeel(info.getClassName());  
    break;  
}  
}  
  
} catch (ClassNotFoundException ex) {  
  
    java.util.logging.Logger.getLogger(createTapal.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);  
  
} catch (InstantiationException ex) {  
  
    java.util.logging.Logger.getLogger(createTapal.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);  
  
} catch (IllegalAccessException ex) {  
  
    java.util.logging.Logger.getLogger(createTapal.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);  
  
} catch (javax.swing.UnsupportedLookAndFeelException ex) {  
  
    java.util.logging.Logger.getLogger(createTapal.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);  
  
}  
  
//</editor-fold>  
  
  
/* Create and display the form */  
  
java.awt.EventQueue.invokeLater(new Runnable() {  
  
    public void run() {  
  
        new createTapal().setVisible(true);  
    }  
});
```

```
    }

});  
}  
  
// Variables declaration - do not modify//GEN-BEGIN:variables  
  
private javax.swing.JTextField TxtiGrp;  
  
private javax.swing.JTextField TxtiNameC;  
  
private javax.swing.JButton btn_bk;  
  
private javax.swing.JButton btn_cr;  
  
private javax.swing.JButton btn_ex;  
  
private javax.swing.JButton btn_re;  
  
private javax.swing.JLabel jLabel3;  
  
private javax.swing.JLabel jLabel4;  
  
private javax.swing.JLabel jLabel5;  
  
// End of variables declaration//GEN-END:variables  
  
}
```

File Name: mngTapal.java

```
import java.awt.Color;  
  
import java.io.BufferedReader;  
  
import java.io.IOException;  
  
import java.io.InputStreamReader;  
  
  
/*
```

* To change this license header, choose License Headers in Project Properties.

* To change this template file, choose Tools | Templates

* and open the template in the editor.

*/

/**

*

* @author kshitij

*/

public class mngTapal extends javax.swing.JFrame {

/**

* Creates new form mngTapal

*/

public mngTapal() {

 this.getContentPane().setBackground(Color.decode("#0099FF"));

 this.setUndecorated(false);

 this.setResizable(false);

 initComponents();

}

/**

* This method is called from within the constructor to initialize the form.

* WARNING: Do NOT modify this code. The content of this method is always

* regenerated by the Form Editor.

```
*/  
  
@SuppressWarnings("unchecked")  
  
// <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents  
  
private void initComponents() {  
  
    jLabel3 = new javax.swing.JLabel();  
  
    jLabel2 = new javax.swing.JLabel();  
  
    TxtiName = new javax.swing.JTextField();  
  
    btn_en = new javax.swing.JButton();  
  
    btn strt = new javax.swing.JButton();  
  
    btn_stp = new javax.swing.JButton();  
  
    btn_dis = new javax.swing.JButton();  
  
    btn_des = new javax.swing.JButton();  
  
    btn_ex = new javax.swing.JButton();  
  
    btn_bk = new javax.swing.JButton();  
  
    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);  
  
    jLabel3.setFont(new java.awt.Font("Lucida Bright", 1, 24)); // NOI18N  
    jLabel3.setForeground(new java.awt.Color(5, 0, 0));  
    jLabel3.setText("Manage Instances");  
  
    jLabel2.setFont(new java.awt.Font("Lucida Bright", 1, 18)); // NOI18N  
    jLabel2.setForeground(new java.awt.Color(1, 1, 1));
```

```
jLabel2.setText("Enter Instance Name:");
jLabel2.setName(""); // NOI18N

TxtiName.setBackground(new java.awt.Color(254, 253, 251));
TxtiName.setFont(new java.awt.Font("Lucida Bright", 0, 14)); // NOI18N
TxtiName.setBorder(javax.swing.BorderFactory.createLineBorder(new
java.awt.Color(255, 149, 0)));

btn_en.setBackground(new java.awt.Color(237, 195, 105));
btn_en.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N
btn_en.setForeground(new java.awt.Color(1, 1, 1));
btn_en.setText("Enable");
btn_en.setFocusable(false);
btn_en.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btn_enActionPerformed(evt);
    }
});

btn_strt.setBackground(new java.awt.Color(237, 195, 105));
btn_strt.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N
btn_strt.setForeground(new java.awt.Color(1, 1, 1));
btn_strt.setText("Start");
btn_strt.setFocusable(false);
btn_strt.addActionListener(new java.awt.event.ActionListener() {
```

```
public void actionPerformed(java.awt.event.ActionEvent evt) {  
  
    btn strtActionPerformed(evt);  
  
}  
  
});  
  
  
btn_stp.setBackground(new java.awt.Color(237, 195, 105));  
btn_stp.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N  
btn_stp.setForeground(new java.awt.Color(1, 1, 1));  
btn_stp.setText("Stop");  
btn_stp.setFocusable(false);  
btn_stp.addActionListener(new java.awt.event.ActionListener() {  
  
    public void actionPerformed(java.awt.event.ActionEvent evt) {  
  
        btn_stpActionPerformed(evt);  
  
    }  
  
});  
  
  
btn_dis.setBackground(new java.awt.Color(237, 195, 105));  
btn_dis.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N  
btn_dis.setForeground(new java.awt.Color(1, 1, 1));  
btn_dis.setText("Disable");  
btn_dis.setFocusable(false);  
btn_dis.addActionListener(new java.awt.event.ActionListener() {  
  
    public void actionPerformed(java.awt.event.ActionEvent evt) {  
  
        btn_disActionPerformed(evt);  
  
    }  
  
}
```

```
});

btn_des.setBackground(new java.awt.Color(237, 195, 105));
btn_des.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N
btn_des.setForeground(new java.awt.Color(1, 1, 1));
btn_des.setText("Destroy");
btn_des.setFocusable(false);
btn_des.addActionListener(new java.awt.event.ActionListener() {

    public void actionPerformed(java.awt.event.ActionEvent evt) {

        btn_desActionPerformed(evt);
    }
});

btn_ex.setBackground(new java.awt.Color(205, 81, 51));
btn_ex.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N
btn_ex.setForeground(new java.awt.Color(1, 1, 1));
btn_ex.setText("Exit");
btn_ex.addActionListener(new java.awt.event.ActionListener() {

    public void actionPerformed(java.awt.event.ActionEvent evt) {

        btn_exActionPerformed(evt);
    }
});

btn_bk.setBackground(new java.awt.Color(205, 81, 51));
btn_bk.setFont(new java.awt.Font("Lucida Bright", 1, 15)); // NOI18N
```

```
btn_bk.setForeground(new java.awt.Color(1, 1, 1));
btn_bk.setText("Back to Main Menu");
btn_bk.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btn_bkActionPerformed(evt);
    }
});

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(layout.createSequentialGroup()
        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addGap(210, 210, 210)
                .addComponent(jLabel3))
            .addGroup(layout.createSequentialGroup()
                .addGap(83, 83, 83)
                .addComponent(jLabel2)))
        .addGap(18, 18, 18)
        .addComponent(TxtiName, javax.swing.GroupLayout.PREFERRED_SIZE,
        240, javax.swing.GroupLayout.PREFERRED_SIZE))
    .addGroup(layout.createSequentialGroup()
        .addGap(58, 58, 58)
```

```
.addComponent(btn_en, javax.swing.GroupLayout.PREFERRED_SIZE, 80,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addComponent(btn strt, javax.swing.GroupLayout.PREFERRED_SIZE, 80,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addComponent(btn stp, javax.swing.GroupLayout.PREFERRED_SIZE, 80,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addGap(16, 16, 16)

.addComponent(btn dis, javax.swing.GroupLayout.PREFERRED_SIZE, 100,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addGap(18, 18, 18)

.addComponent(btn des, javax.swing.GroupLayout.PREFERRED_SIZE, 100,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addGroup(layout.createSequentialGroup()

.addGap(250, 250, 250)

.addComponent(btn bk, javax.swing.GroupLayout.PREFERRED_SIZE, 200,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addGap(18, 18, 18)

.addComponent(btn ex, javax.swing.GroupLayout.PREFERRED_SIZE, 90,
javax.swing.GroupLayout.PREFERRED_SIZE)))

.addContainerGap(68, Short.MAX_VALUE))

);

layout.setVerticalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addGap(30, 30, 30)
```

```
.addComponent(jLabel3)

.addComponent(jLabel3)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(TxtiName, javax.swing.GroupLayout.PREFERRED_SIZE, 30,
    javax.swing.GroupLayout.PREFERRED_SIZE)
    .addComponent(jLabel2))

.addComponent(jLabel2)

.addGap(34, 34, 34)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(btn_dis)
    .addComponent(btn_des)
    .addComponent(btn_stp)
    .addComponent(btn_strt)
    .addComponent(btn_en))

.addComponent(btn_en)

.addGap(38, 38, 38)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(btn_bk, javax.swing.GroupLayout.PREFERRED_SIZE, 30,
    javax.swing.GroupLayout.PREFERRED_SIZE)
    .addComponent(btn_ex))

.addComponent(btn_ex)

.addContainerGap(32, Short.MAX_VALUE))

);

pack();

setLocationRelativeTo(null);

}// </editor-fold>//GEN-END:initComponents
```

```
private void btn_enActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_btn_enActionPerformed

    String instname=TxtiName.getText();

    String cmd="postmulti -i postfix-";
    String cmd2="-e enable";

    cmd=cmd.concat(instname);
    cmd=cmd.concat(cmd2);

    //EXECUTING BASH COMMAND Enable AN INSTANCE

    try
    {
        Process p5=Runtime.getRuntime().exec(cmd);
        p5.waitFor();

        BufferedReader reader=new BufferedReader(new
InputStreamReader(p5.getInputStream()));

        String line=reader.readLine();
        while(line!=null)
        {
            System.out.println(line);
            line=reader.readLine();
        }
    }
}
```

```
}

catch(IOException | InterruptedException e1) {}

System.out.println("Instance Enabled.");

}//GEN-LAST:event_btn_enActionPerformed

private void btn_strtActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_btn_strtActionPerformed

String instname=TxtiName.getText();

String cmd="postmulti -i postfix-";

String cmd2="-p start";

cmd=cmd.concat(instname);

cmd=cmd.concat(cmd2);

//EXECUTING BASH COMMAND TO START AN INSTANCE

try

{

Process p5=Runtime.getRuntime().exec(cmd);

p5.waitFor();

BufferedReader reader=new BufferedReader(new

InputStreamReader(p5.getInputStream()));

String line=reader.readLine();
```

```
while(line!=null)
{
    System.out.println(line);
    line=reader.readLine();
}

}

catch(IOException | InterruptedException e1) {}

System.out.println("Instance Started.");

}//GEN-LAST:event_btn strtActionPerformed

private void btn_stpActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_btn_stpActionPerformed
// TODO add your handling code here:
String instname=TxtiName.getText();

String cmd="postmulti -i postfix-";
String cmd2="-p stop";

cmd=cmd.concat(instname);
cmd=cmd.concat(cmd2);
```

```
//EXECUTING BASH COMMAND TO STOP AN INSTANCE

try
{
    Process p5=Runtime.getRuntime().exec(cmd);
    p5.waitFor();

    BufferedReader reader=new BufferedReader(new
InputStreamReader(p5.getInputStream()));

    String line=reader.readLine();

    while(line!=null)

    {
        System.out.println(line);

        line=reader.readLine();
    }

}

catch(IOException | InterruptedException e1) {}

System.out.println("Instance Stopped.");

}//GEN-LAST:event_btn_stpActionPerformed

private void btn_disActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_btn_disActionPerformed
// TODO add your handling code here:

String instname=TxtiName.getText();
```

```
String cmd="postmulti -i postfix-";
String cmd2="-e disable";

cmd=cmd.concat(instname);
cmd=cmd.concat(cmd2);

//EXECUTING BASH COMMAND TO Disabe AN INSTANCE

try
{
    Process p5=Runtime.getRuntime().exec(cmd);
    p5.waitFor();

    BufferedReader reader=new BufferedReader(new
InputStreamReader(p5.getInputStream()));

    String line=reader.readLine();
    while(line!=null)
    {
        System.out.println(line);
        line=reader.readLine();
    }
}

catch(IOException | InterruptedException e1) {}

System.out.println("Instance Disabled.");
```

```
}//GEN-LAST:event_btn_disActionPerformed
```

```
private void btn_desActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_btn_desActionPerformed
```

```
// TODO add your handling code here:
```

```
String instname=TxtiName.getText();
```

```
// STOP
```

```
String cmdstop="postmulti -i postfix-";
```

```
String cmdstop2="-p stop";
```

```
cmdstop=cmdstop.concat(instname);
```

```
cmdstop=cmdstop.concat(cmdstop2);
```

```
//EXECUTING BASH COMMAND TO STOP AN INSTANCE
```

```
try
```

```
{
```

```
Process p5=Runtime.getRuntime().exec(cmdstop);
```

```
p5.waitFor();
```

```
BufferedReader reader=new BufferedReader(new  
InputStreamReader(p5.getInputStream()));
```

```
String line=reader.readLine();
```

```
while(line!=null)
{
    System.out.println(line);
    line=reader.readLine();
}

}
catch(IOException | InterruptedException e1) {}

System.out.println("Instance Destroyed.");

//DISABLE

String cmd="postmulti -i postfix-";
String cmd2="-e disable";

cmd=cmd.concat(instname);
cmd=cmd.concat(cmd2);

//EXECUTING BASH COMMAND TO Disabe AN INSTANCE

try
{
    Process p5=Runtime.getRuntime().exec(cmd);
    p5.waitFor();
}
```

```
BufferedReader reader=new BufferedReader(new  
InputStreamReader(p5.getInputStream()));  
  
String line=reader.readLine();  
  
while(line!=null)  
{  
  
    System.out.println(line);  
  
    line=reader.readLine();  
  
}  
  
}  
  
catch(IOException | InterruptedException e1) {}  
  
  
System.out.println("");  
  
  
//Destroy  
  
String cmddestroy="postmulti -i postfix-";  
  
String cmddestroy2="-e destroy";  
  
  
cmddestroy=cmddestroy.concat(instname);  
  
cmddestroy=cmddestroy.concat(cmddestroy2);  
  
  
try  
{  
  
    Process p5=Runtime.getRuntime().exec(cmddestroy);  
  
    p5.waitFor();
```

```
BufferedReader reader=new BufferedReader(new  
InputStreamReader(p5.getInputStream()));  
  
String line=reader.readLine();  
  
while(line!=null)  
{  
  
    System.out.println(line);  
  
    line=reader.readLine();  
  
}  
  
}  
  
catch(IOException | InterruptedException e1) {}  
  
System.out.println("");  
  
} //GEN-LAST:event_btn_desActionPerformed  
  
private void btn_exActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_btn_exActionPerformed  
  
    System.exit(2);  
  
}//GEN-LAST:event_btn_exActionPerformed  
  
  
private void btn_bkActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event_btn_bkActionPerformed  
  
    mainTapal a=new mainTapal();  
  
    this.dispose();  
  
    a.setVisible(true);  
  
}//GEN-LAST:event_btn_bkActionPerformed  
  
  
/**  
 * @param args the command line arguments
```

```
*/  
  
public static void main(String args[]) {  
  
    /* Set the Nimbus look and feel */  
  
    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">  
  
    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and  
    feel.  
  
     * For details see  
     http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html  
  
    */  
  
    try {  
  
        for (javax.swing.UIManager.LookAndFeelInfo info :  
            javax.swing.UIManager.getInstalledLookAndFeels()) {  
  
            if ("Nimbus".equals(info.getName())) {  
  
                javax.swing.UIManager.setLookAndFeel(info.getClassName());  
  
                break;  
  
            }  
  
        }  
  
        } catch (ClassNotFoundException ex) {  
  
            java.util.logging.Logger.getLogger(mngTapal.class.getName()).log(java.util.logging.Level.S  
EVERE, null, ex);  
  
        } catch (InstantiationException ex) {  
  
            java.util.logging.Logger.getLogger(mngTapal.class.getName()).log(java.util.logging.Level.S  
EVERE, null, ex);  
  
        } catch (IllegalAccessException ex) {  
  
            java.util.logging.Logger.getLogger(mngTapal.class.getName()).log(java.util.logging.Level.S  
EVERE, null, ex);  
  
    }
```

```
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {  
  
    java.util.logging.Logger.getLogger(mngTapal.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);  
  
}  
  
//</editor-fold>  
  
/* Create and display the form */  
  
java.awt.EventQueue.invokeLater(new Runnable() {  
  
    public void run() {  
  
        new mngTapal().setVisible(true);  
  
    }  
  
});  
  
}  
  
// Variables declaration - do not modify//GEN-BEGIN:variables  
  
private javax.swing.JTextField TxtiName;  
  
private javax.swing.JButton btn_bk;  
  
private javax.swing.JButton btn_des;  
  
private javax.swing.JButton btn_dis;  
  
private javax.swing.JButton btn_en;  
  
private javax.swing.JButton btn_ex;  
  
private javax.swing.JButton btn_stp;  
  
private javax.swing.JButton btn_strt;  
  
private javax.swing.JLabel jLabel2;  
  
private javax.swing.JLabel jLabel3;  
  
// End of variables declaration//GEN-END:variables  
}
```

File Name: try.sh

```
#!/bin/bash

echo "Welcome to the TAPAL Manager."

cd /home/kshitij/NetBeansProjects/TAPAL/src

gksu javac mainTapal.java

gksu java mainTapal
```

File Name: TAPAL.desktop

[Desktop Entry]

Name=TAPAL

Comment=Manager

Exec=/home/kshitij/NetBeansProjects/TAPAL/try.sh

Icon=/home/kshitij/Desktop/Project Docs/Logo/logo.png

Terminal=true

Type=Application

4.3.3 Multithreaded Java Mail

```
import java.io.UnsupportedEncodingException;
import java.util.Properties;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.mail.Message;
import javax.mail.MessagingException;
import javax.mail.Session;
```

```
import javax.mail.Transport;  
  
import javax.mail.internet.AddressException;  
  
import javax.mail.internet.InternetAddress;  
  
import javax.mail.internet.MimeMessage;  
  
public class Multithreadingmail extends Thread {  
  
    @Override  
  
    public void run() {  
  
        System.out.println("Thread Running");  
  
        String to="ankit1@ankit.com";  
  
        String from="ankit2@ankit.com";  
  
        System.out.println("running thread name is:"+Thread.currentThread().getName());  
  
        System.out.println("running thread "+Thread.currentThread().getName()+" priority  
is:"+Thread.currentThread().getPriority());  
  
        Properties props = new Properties();  
  
        Session session = Session.getDefaultInstance(props, null);  
  
        String msgBody = "Sending email using JavaMail API...";  
  
        try {  
  
            Message msg = new MimeMessage(session);  
  
            try {  
  
                msg.setFrom(new InternetAddress(from, "NoReply"));  
  
            } catch (UnsupportedEncodingException ex) {  
  
                Logger.getLogger(Multithreadingmail.class.getName()).log(Level.SEVERE, null,  
ex);  
  
            }  
  
            try {  
  
                msg.addRecipient(Message.RecipientType.TO,  
                
```

```
        new InternetAddress(to, "Mr. Recipient"));

    } catch (UnsupportedEncodingException ex) {

        Logger.getLogger(Multithreadingmail.class.getName()).log(Level.SEVERE, null,
ex);

    }

msg.setSubject("Welcome To Java Mail API");

msg.setText(msgBody);

System.out.println("Ready to send email");

Transport.send(msg);

System.out.println("Email sent successfully...");

} catch (AddressException e) {

throw new RuntimeException(e);

} catch (MessagingException e) {

throw new RuntimeException(e);

}

}

}

public static void main(String[] args) throws Exception{

Multithreadingmail t1= new Multithreadingmail();

Multithreadingmail t2= new Multithreadingmail();

Multithreadingmail t3= new Multithreadingmail();

Multithreadingmail t4= new Multithreadingmail();

t1.setName("1");

t2.setName("2");

t3.setName("3");

t4.setName("4");
```

```
t2.start();  
t1.start();  
t3.setPriority(Thread.MIN_PRIORITY);  
t4.setPriority(Thread.MAX_PRIORITY);  
t3.start();  
t4.start();  
}  
}
```

4.4 Screenshot

4.4.1 Squirrelmail

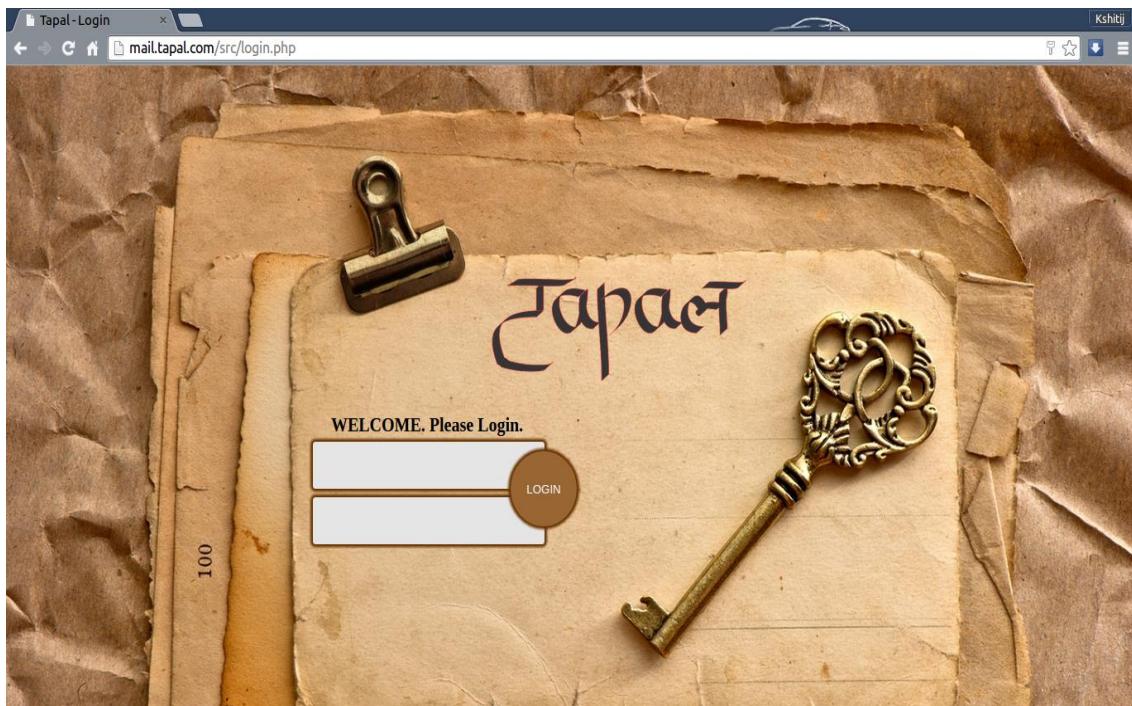


Figure 4.4-1: Login Page

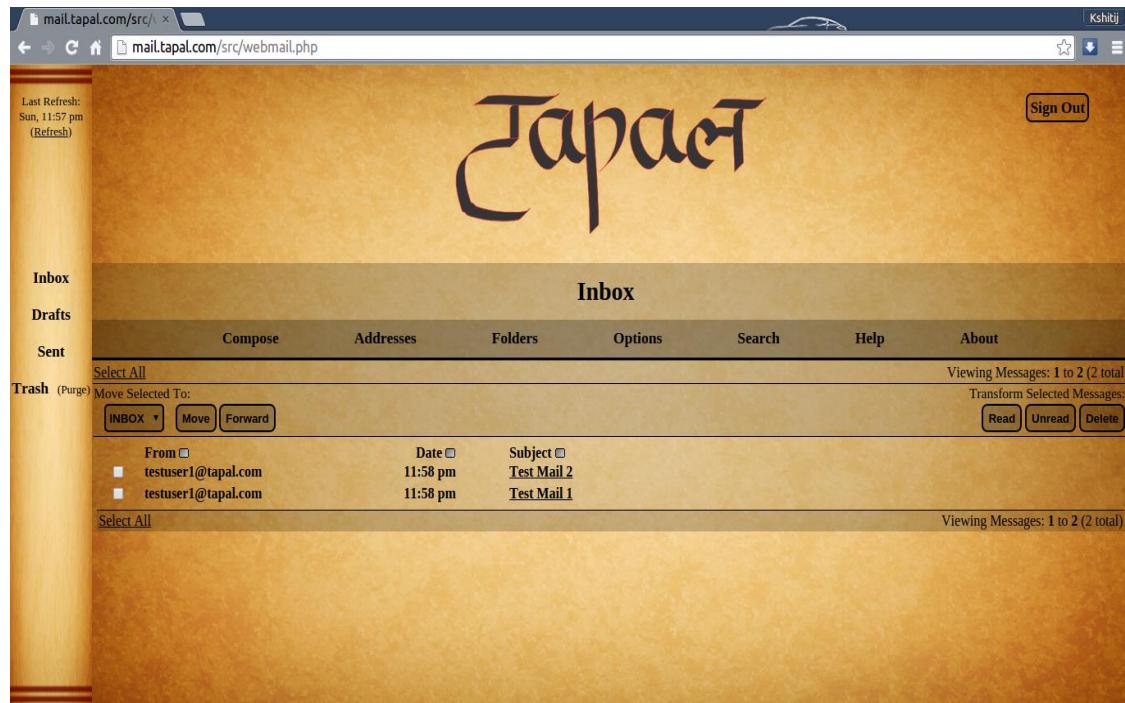


Figure 4.4-2: Home Page

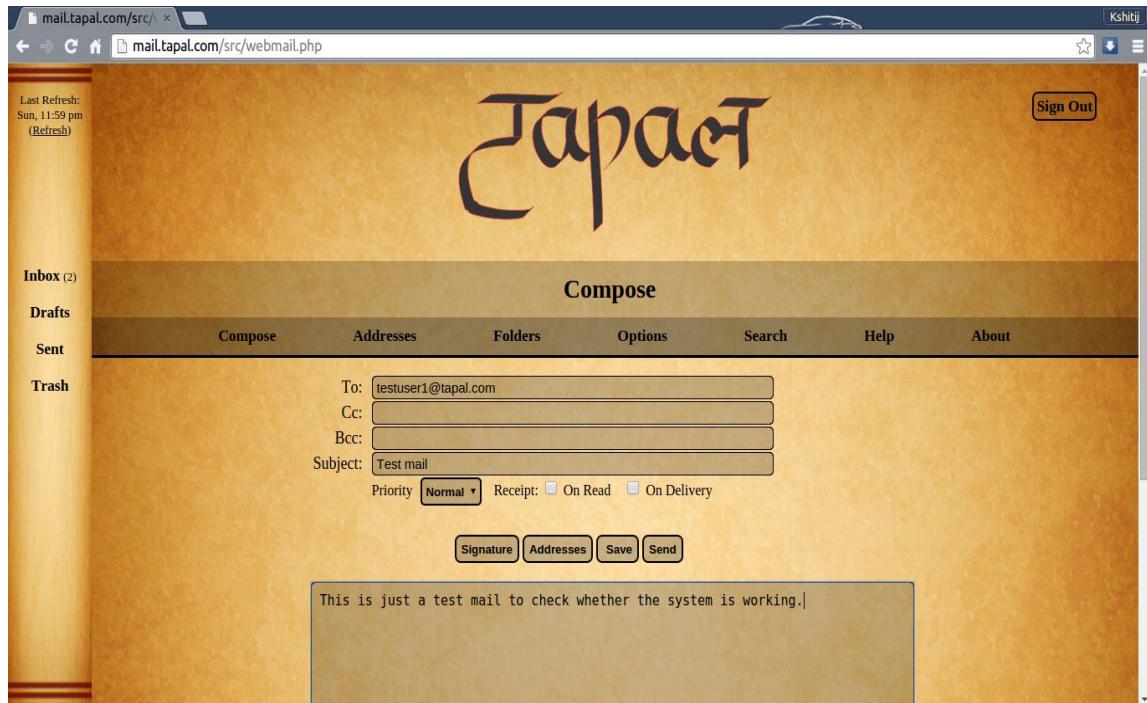


Figure 4.4-3: Compose

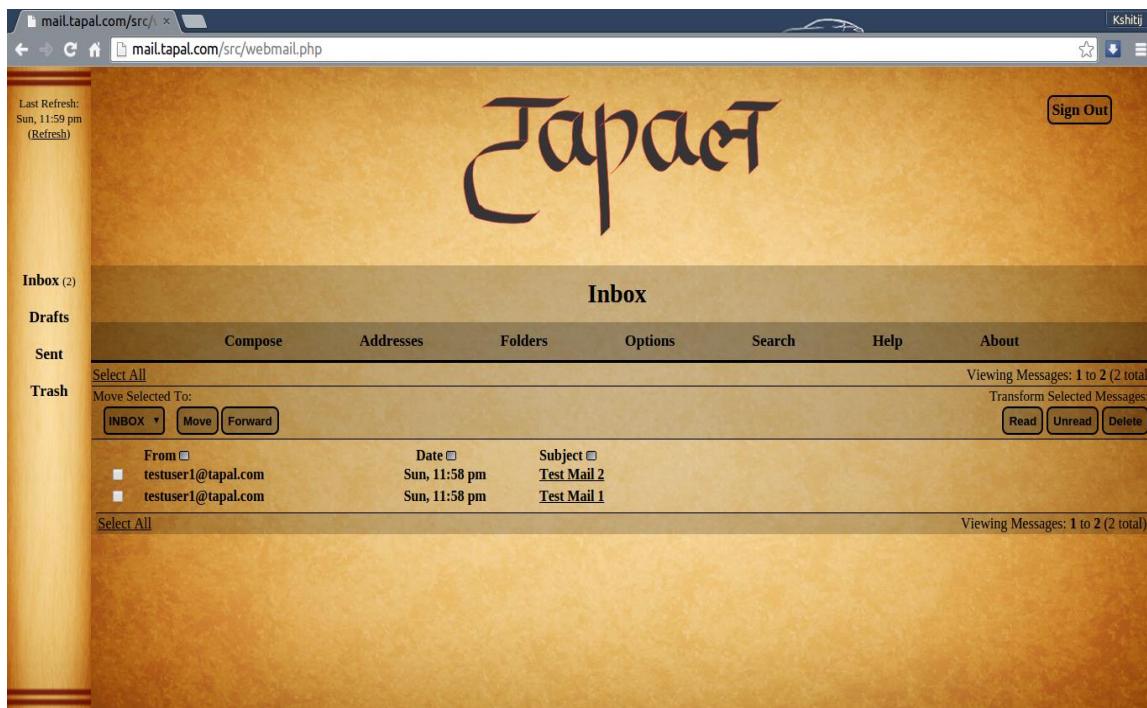
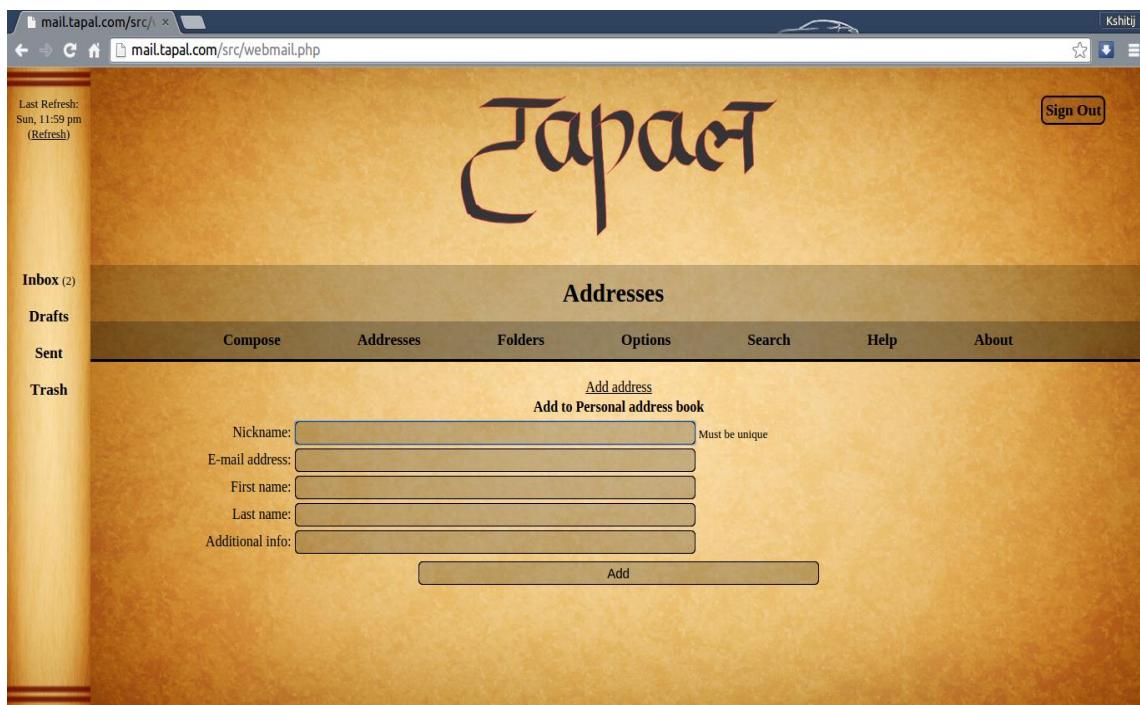
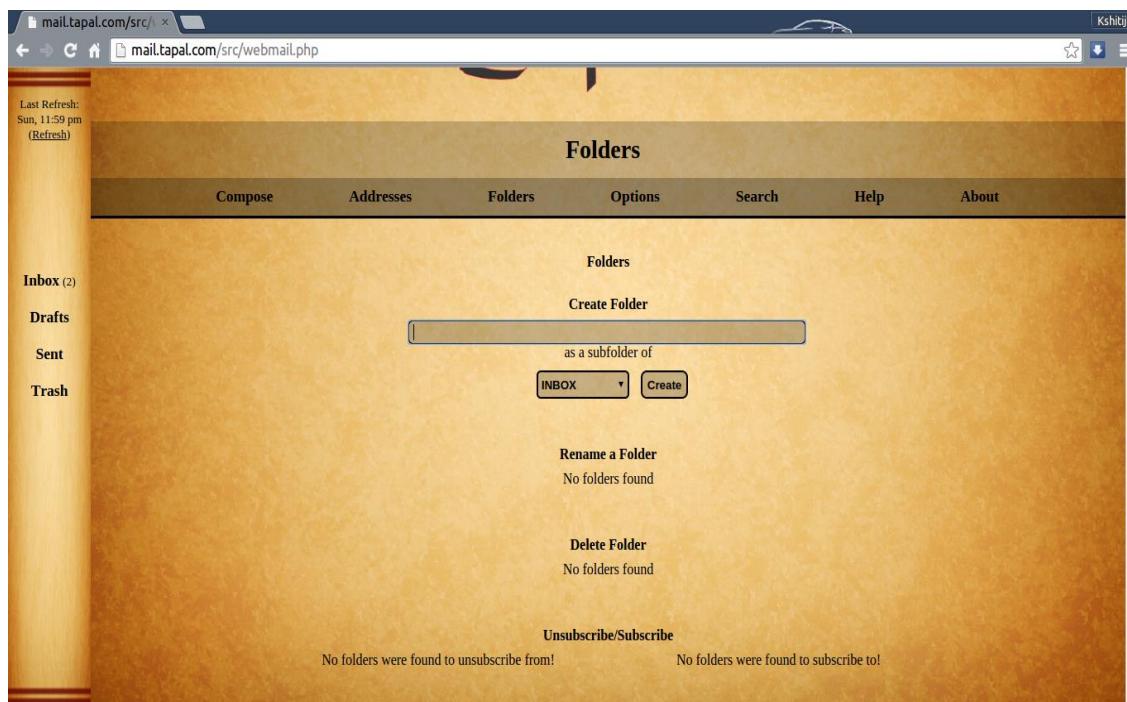


Figure 4.4-4: Inbox

**Figure 4.4-5: Address Book****Figure 4.4-6: Folder**

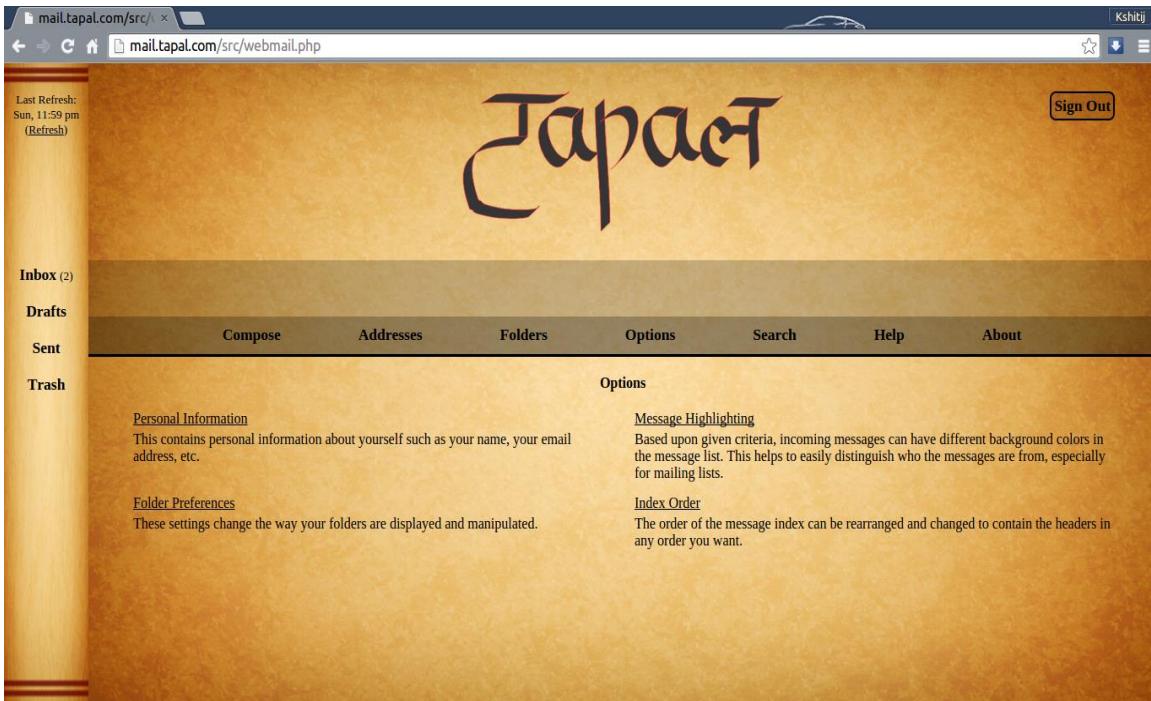


Figure 4.4-7: Options

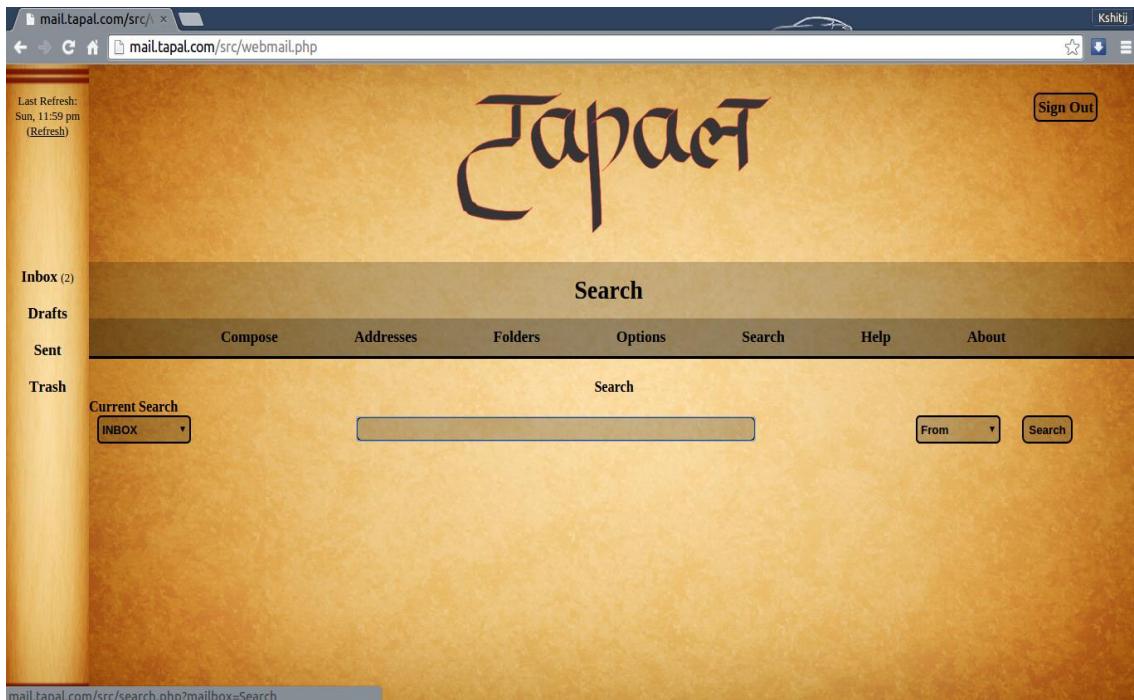
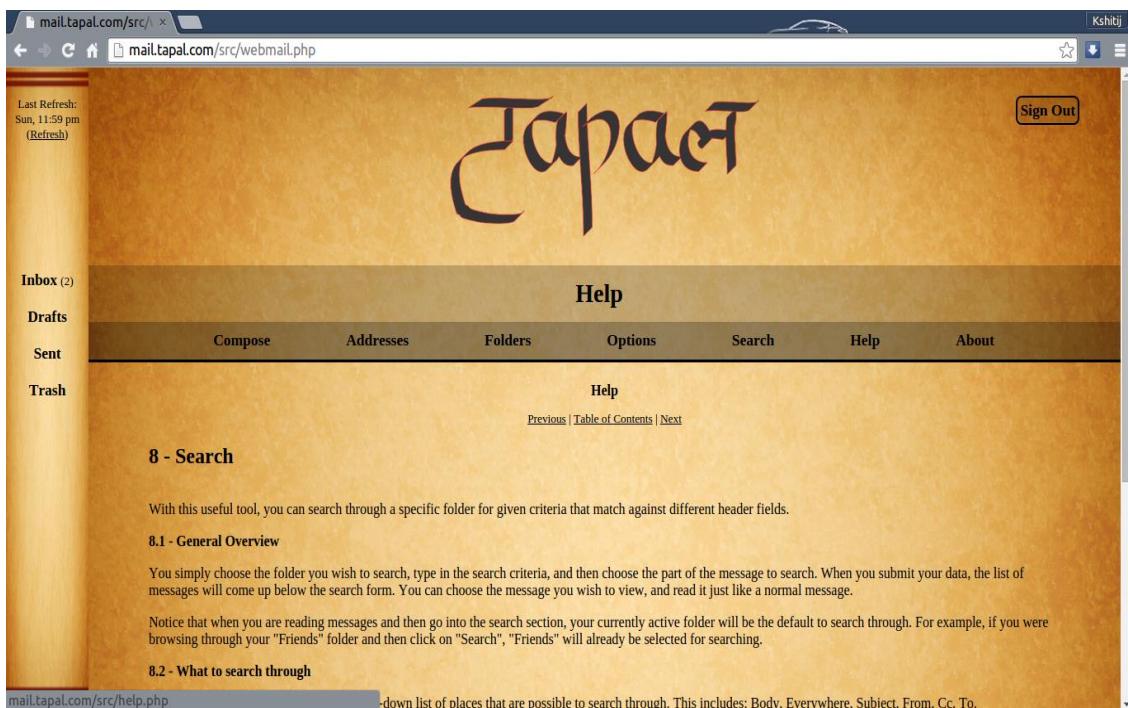
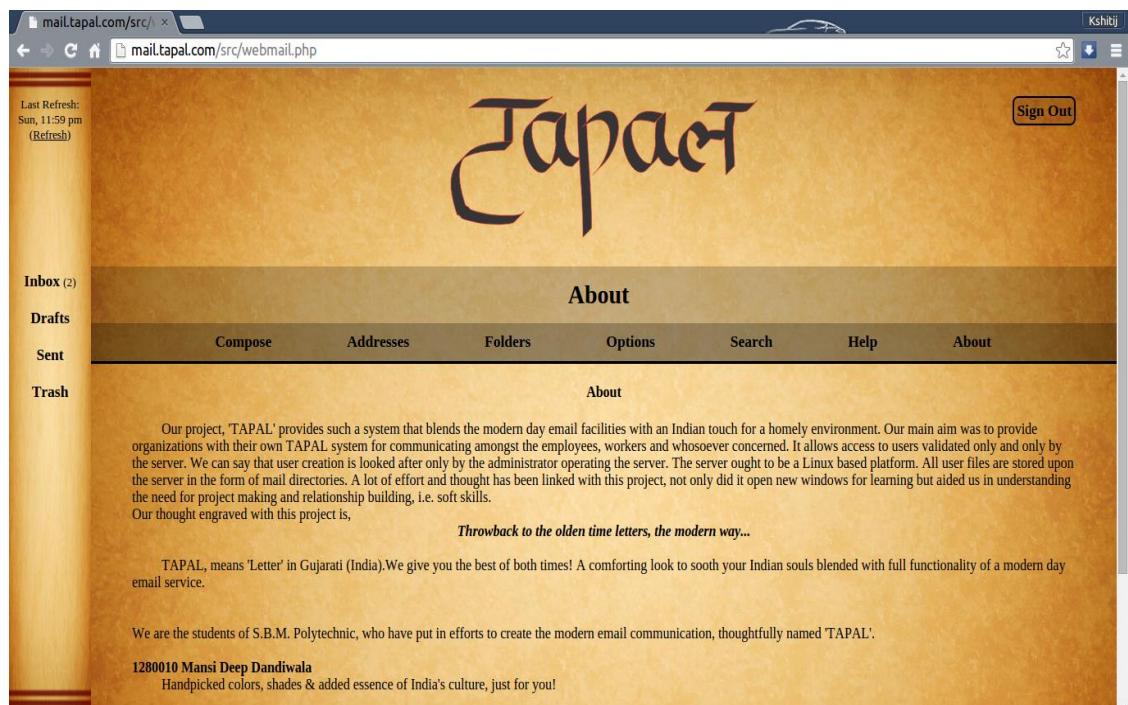


Figure 4.4-8: Search

**Figure 4.4-9: Help****Figure 4.4-10: About Us**

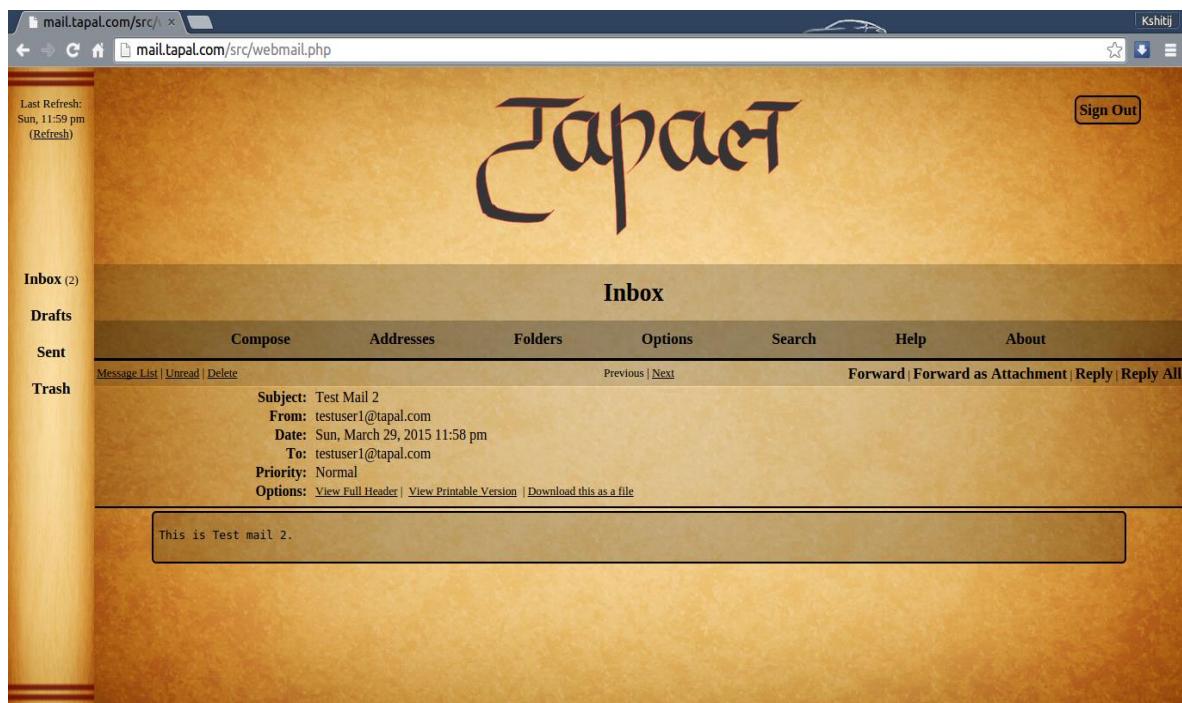


Figure 4.4-11: Mail Read

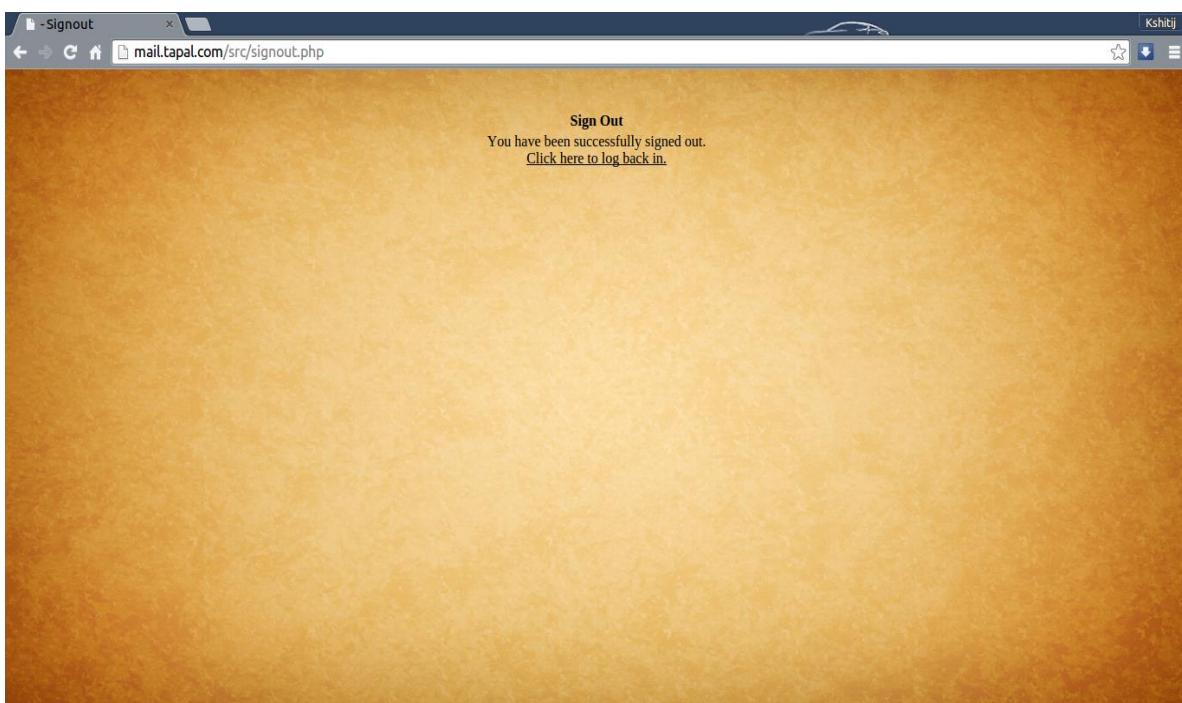


Figure 4.4-12: Sign Out

4.4.2 TAPAL Multi-Instance Manager



Figure 4.4-13: Main Frame



Figure 4.4-14: Add User



Figure 4.4-15: Delete User

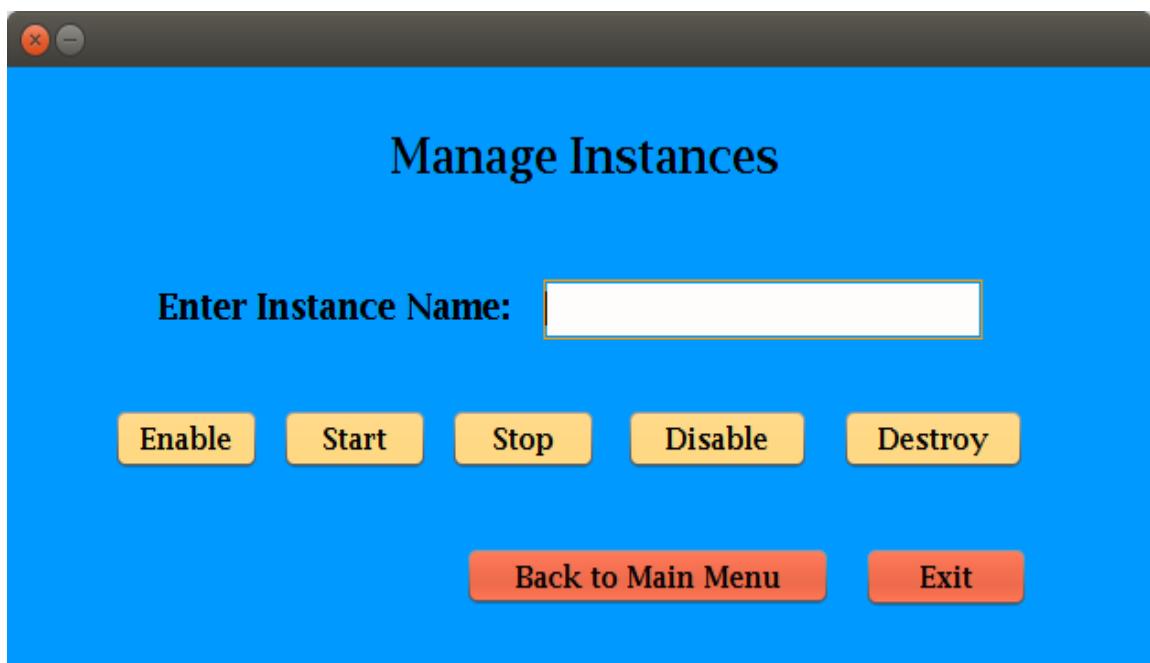


Figure 4.4-16: Manage Instance

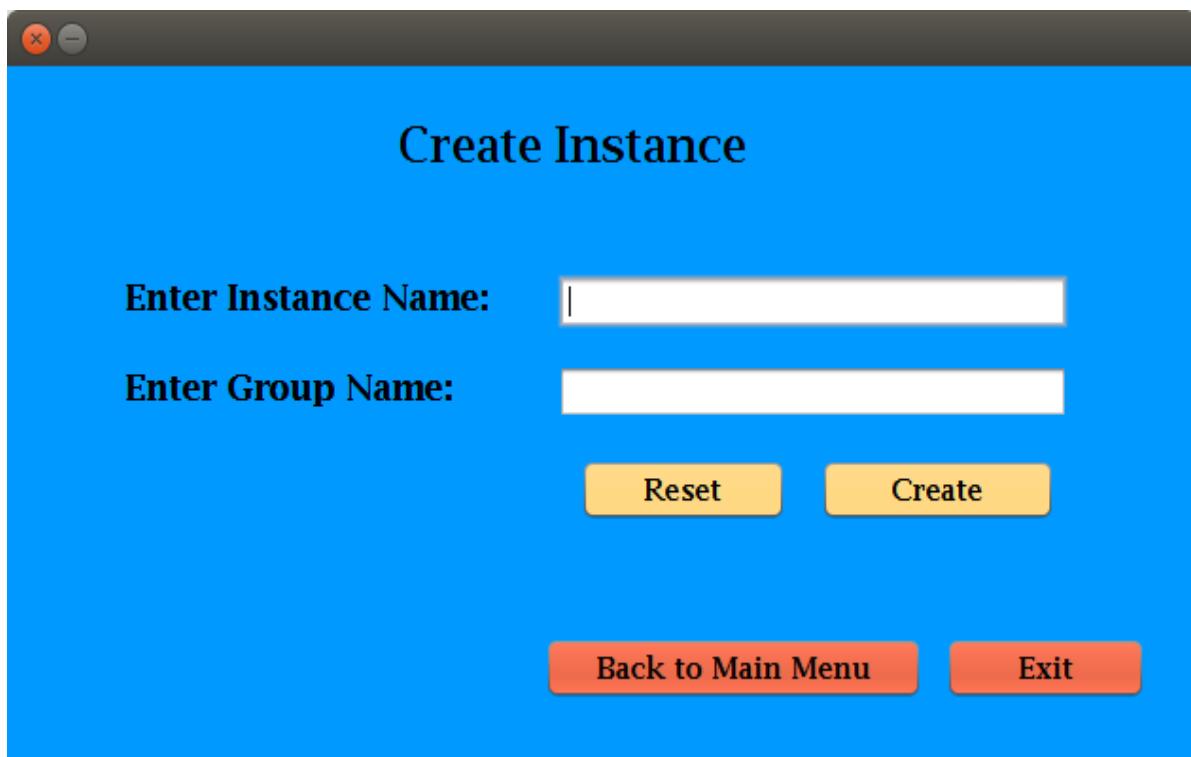


Figure 4.4-17: Creating Instance



Figure 4.4-18: Output on Terminal

4.4.3 Multithreaded Java Mail Output

```
ankit@mail:~/NetBeansProjects/Multithreading/Multithreadingmail/src$ java Multithreadingmail
Thread Running
Thread Running
Thread Running
Thread Running
running thread name is:4
running thread name is:3
running thread name is:2
running thread 3 priority is:1
running thread 4 priority is:10
running thread name is:1
running thread 2 priority is:5
running thread 1 priority is:5
Ready to send email
Ready to send email
Ready to send email
Ready to send email
Email sent successfully...
Email sent successfully...
Email sent successfully...
Email sent successfully...
ankit@mail:~/NetBeansProjects/Multithreading/Multithreadingmail/src$
```

Figure 4.4-19: Terminal Output

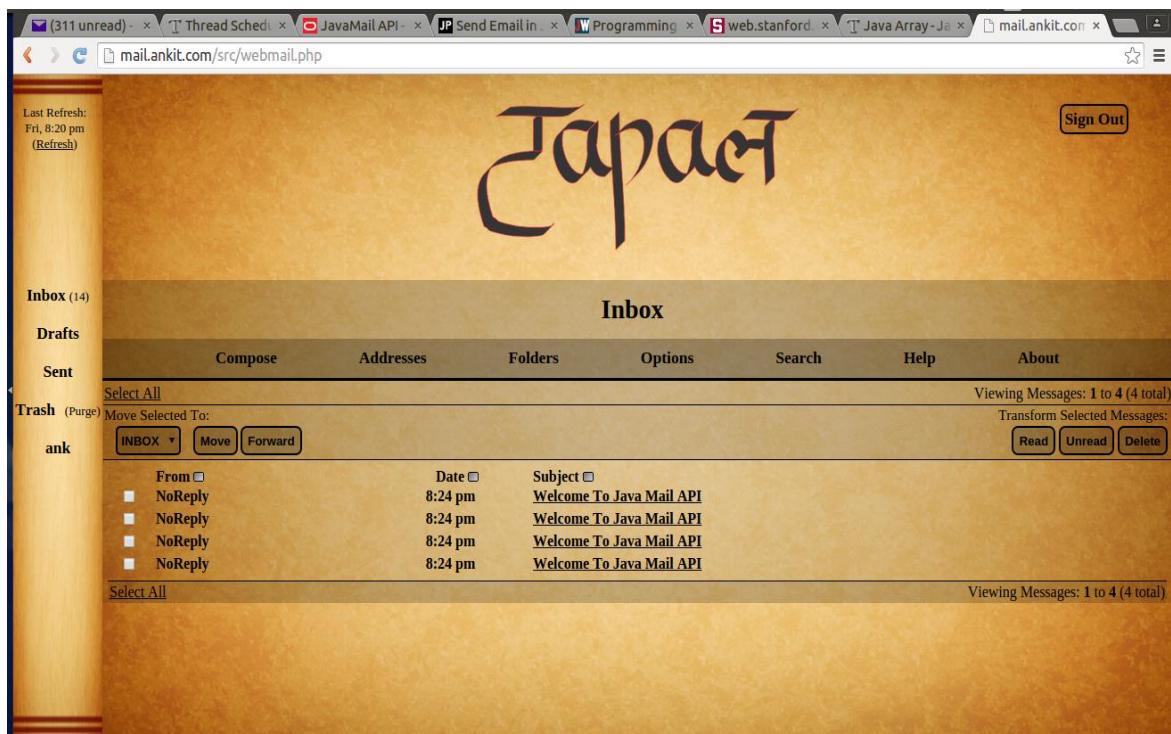


Figure 4.4-20: Mails received using Multithreading

5 Future Scope

The future belongs to those who prepare for it today.

Our thoughts don't halt here. There are various areas where much more precision and a better approach could be implemented. Further we plan to incorporate changes and advancement in this system of ours. They are as mentioned below:

- Modifying our servers in order to implementing full fledge multi-threading.
- Coding our own server to achieve maximum of our thoughts and ideas.
- Extending the communication to the outside world but at the same time not losing the security for in organizational communication.
- Developing our own email client for desktop access and management of accounts.

6 Bibliography

- [1] Jordi Guitart Fernandez, "Performance Improvement of Multithreaded Java Application Execution on Multi-Processor Systems", IEEE 2005
- [2] Jae-Young Kim and James Won-Ki Hong, "Design and Implementation of a Web-based Internet/Intranet Mail Server Management System", IEEE
- [3] http://www.postfix.org/MULTI_INSTANCE_README.html
- [4] <https://nixtree.com/blog/tuning-and-increasing-speed-in-postfix/>
- [5] <http://www.hsc.fr/ressources/cours/postfix/doc/rate.html>
- [6] <http://www.krizna.com/ubuntu/setup-mail-server-ubuntu-14-04/>
- [7] <http://www.opensourceforu.com/2014/12/setting-mail-server-can-fun/>
- [8] <http://www.postfix.org/postmulti.1.html>
- [9] <http://www.postfix.org/postconf.5.html>
- [10] <http://wiki.zimbra.com/wiki/Ajcody-MTA-Postfix-Topics>
- [11] <http://www.linux-magazine.com/Online/News/Multithreaded-Mail-Carrier-Postfix>
- [12] <http://unix.stackexchange.com/questions/187409/multithreading-in-mail-server-postfix>
- [13] <http://linux.die.net/man/1/postmulti>
- [14] <https://help.ubuntu.com/community/MailServer>
- [15] http://www.postfix.org/CONNECTION_CACHE_README.html

7 About Us

We are the students of Computer Engineering at Shri Bhagubhai Mafatlal Polytechnic belonging to group 6, have taken immense efforts in setting up an email server over the Linux platform within an organization. We strive to provide an email server secure enough to handle in organizational communication. Our efforts have been directed in achieving various milestones throughout this tenure. Our thoughts, hopes and wishes are forwarded along.

Our milestones:

- Working with Linux
- Reading Server Files
- Modifying Web Client
- Modifying Server Files

We have thoughtfully named our system as TAPAL, because we aimed at providing our users with an email system with an Indian essence. The name TAPAL, give a glimpse of our roots, India and to the olden times when mails were regarded as TAPAL.

Not only are we really happy to present this to you all but also are really glad on being able to accept this project. Through this document we wished to provide a glimpse of our effort, work schedule and our study regarding the system. It has been an informative adventure. We have tried our best to combine the best of our knowledge and abilities.

We hope TAPAL is useful to each and every organization.

8 Contact us

Organization e-mail address:

2014tapal@gmail.com

Get to know us up and close:

Mansi Deep Dandiwala (1280010)

mansidandi@yahoo.co.in

Kshitij Manish Shah (1280047)

kshitijshah95@gmail.com

Ankit Purvesh Shah (1280051)

ank6735@gmail.com