

National Institute of Technology,Raipur

Department of Computer Science & Engineering



Email Client

A Term Project on Network Programming

GitHub Project Link: https://github.com/mohit159/Email_Client

Submitted By:

Roll no: 14115013

Roll no: 14115025

Roll no: 14115058

Roll no: 14115061

Roll no: 14115087

Name: Anubhav Jain

Name: Dalai Was

Name: Mohit Laheri

Name: Naval Kishor Dewangan

Name: Somendra Patel

Abstract

The project Entitle “Mail Client” deals with identifying the clients to send and receive mail with the same login. This utility will allow multiple clients to login under the different login name and enabling them to send and receive mails. Each user willing to avail the services offered by the mail server application should exist as a user before he can send or receive mails. This is made possible by prompting each user to enter his user-id and password before he can send or view his mails.

The ISPs mail server handles the emails sent from an ISP (Internet Service Provider); the ISP mail server is similar to an electronic post office. The sent email first comes to the mail server; it is then processed and forwarded to the collector’s destination where another mail server is situated.

The collector mail server receives all the incoming emails and sorts them electronically in the collectors’ mailboxes. Then, the collector or the recipient of the email has to use their email program to pick up the email. This is done by directly connecting to the mail server through the program, and then the received items can be seen in the mailbox.

The problem with the existing Email Client System is that mailing is done through browser by using services like Hotmail, Yahoo, Google, etc. These systems use HTTP port 80 to access the emails, and the overall procedure here is not safe to send confidential messages. This existing system can be easily hacked by hackers, some data may be modified or even lost.

With the proposed Email Client Software, email client program is run on Java coding. It can be used to send/receive emails very securely. Additionally, the cost of mail client in the proposed system is less compared to the mail services.

Software/ Hardware Requirements

1. Hardware Requirements:

No of Systems : 2 – Client & Server (Minimum)
Processor : Intel P4 (Or equivalent)
Hard Disk : 40 GB
RAM : 512 MB (minimum)
Network :Standard Wired/Wireless network with either Ad-hoc or
I infrastructure mode

2. Software Requirements:

Operating System: Windows, Linux.
Software: Java SDK 2, Borland Net-Beans 6.0.

Future Scope of the Project: This object is having a broad future scope as it can be extended to provide sending messages on line. This system can be implemented for online sharing without the intervention of the authority. If it is done so user can read messages from anywhere in the world. He can read anywhere from is login account, smtp server. In other words the future scope is to provide the better service

Table of Contents

Introduction.....	4
Detailed description of the solution to the problem.....	5
POP3.....	5
IMAP	5
SMTP	5
HTTP	6
Modules and its Description.....	6
E-MAIL SYSTEM	7
Mailer.....	7
Mail Server	7
Mailboxes	7
WORKING OF EMAIL:	7
Snapshots	9
Conclusion	11
References.....	12

Introduction

Motivation: There are many organizations which use manual methods for sharing information between organization using manual methods. Using this simple mailing system we can computerize this process.

The development of the new system contains the various activities, which try to automate the entire process keeping in view of the database integration approach. User friendliness is provided in the application with various controls. The system makes the overall project management much easier and flexible. There is no risk of data mismanagement at any level while the project development is under process.

Problem: To make own email client server using java which allow user to use multiple email account. Our client should allow to compose mails in offline mode which should automatically be sent whenever there is Internet connectivity. This utility will allow multiple clients to login under the different login name and still have personalized mail information, enabling them to send and receive mails. Client can also able to store recently synced mails locally, for access in offline mode.

Project Outline: This proposed project is an Email Client application developed using Java and some Java APIs. Here, standard sockets and networking have been used in addition to Java Mail API. The protocols used in the project are SMTP, POP3.

E-mail is one of the most common and reliable methods of communication for both personal and business purposes. It also plays an important role in each and every Web site. This role will be in the type of automated e-mails from the server after posting information from a form. In this project there are different types of modules like (Member registration Module, Sending and Receiving mails, Admin Module, Login/Logout Module) are available.

Detailed description of the solution to the problem

POP3: Post Office Protocol version 3 (POP3) is a standard mail protocol used to receive emails from a remote server to a local email client. POP3 allows you to download email messages on your local computer and read them even when you are offline. Note, that when you use POP3 to connect to your email account, messages are downloaded locally and removed from the email server. This means that if you access your account from multiple locations, that may not be the best option for you. On the other hand, if you use POP3, your messages are stored on your local computer, which reduces the space your email account uses on your web server.

By default, the POP3 protocol works on two ports:

- **Port 110** - this is the default POP3 non-encrypted port
- **Port 995** - this is the port you need to use if you want to connect using POP3 securely

IMAP: The Internet Message Access Protocol (IMAP) is a mail protocol used for accessing email on a remote web server from a local client. IMAP and POP3 are the two most commonly used Internet mail protocols for retrieving emails. Both protocols are supported by all modern email clients and web servers.

While the POP3 protocol assumes that your email is being accessed only from one application, IMAP allows simultaneous access by multiple clients. This is why IMAP is more suitable for you if you're going to access your email from different locations or if your messages are managed by multiple users.

By default, the IMAP protocol works on two ports:

- **Port 143** - this is the default IMAP non-encrypted port
- **Port 993** - this is the port you need to use if you want to connect using IMAP securely

SMTP: Simple Mail Transfer Protocol (SMTP) is the standard protocol for sending emails across the Internet.

SMTP Authentication Methods: There are three ways email providers authenticate an SMTP user.

1. **IP address**– Not configurable in client
2. **Authentication via login** -Configurable in client
3. **Sender email address**–Configurable in client

By default, the SMTP protocol works on three ports:

- **Port 25** - this is the default SMTP non-encrypted port
- **Port 2525** - this port is opened on all SiteGround servers in case port 25 is filtered (by your ISP for example) and you want to send non-encrypted emails with SMTP
- **Port 465** - this is the port used if you want to send messages using SMTP securely.

HTTP: HTTP (Hypertext Transfer Protocol) is the set of rules for transferring files (text, graphic images, sound, video, and other multimedia files) on the World Wide Web. As soon as a Web user opens their Web browser, the user is indirectly making use of HTTP. HTTP is an application protocol that runs on top of the TCP/IP suite of protocols (the foundation protocols for the Internet).

HTTP concepts include (as the Hypertext part of the name implies) the idea that files can contain references to other files whose selection will elicit additional transfer requests. Any Web server machine contains, in addition to the Web page files it can serve, an HTTP daemon, a program that is designed to wait for HTTP requests and handle them when they arrive. Your Web browser is an HTTP client, sending requests to server machines. When the browser user enters file requests by either "opening" a Web file (typing in a Uniform Resource Locator or URL) or clicking on a hypertext link, the browser builds an HTTP request and sends it to the Internet Protocol address (IP address) indicated by the URL. The HTTP daemon in the destination server machine receives the request and sends back the requested file or files associated with the request. (A Web page often consists of more than one file.) The latest version of HTTP is HTTP 1.1.

Modules and its Description

Number of Modules

The system after careful analysis has been identified to be presented with the following modules: The modules involved are:

Member login Module:

- Member can sign in by providing their account details (Username and password).

Sending and Receiving mails:

- By making use of this module, Members can send/receive/view mails. Many features have been provided to members so that they can 1) manage (view/ edit/ delete) their mails, 2) forward mails, 3) send attachments, 4) send group mail, manage mails in folders etc.

Email Inbox Module: This module handles all the functions related to mailing like forwarding, view attachment, save attachment.

E-MAIL SYSTEM

E-mail system comprises of the following three components:

- Mailer
- Mail Server
- Mailbox

Mailer

It is also called **mail program, mail application** or **mail client**. It allows us to manage, read and compose e-mail.

Mail Server

The function of mail server is to receive, store and deliver the email. It is must for mail servers to be sunning all the time because if it crashes or is down, email can be lost.

Mailboxes

Mailbox is generally a folder that contains emails and information about them.

WORKING OF EMAIL:

Email working follows the client server approach. In this client is the mailer i.e. the mail application or mail program and server is a device that manages emails.

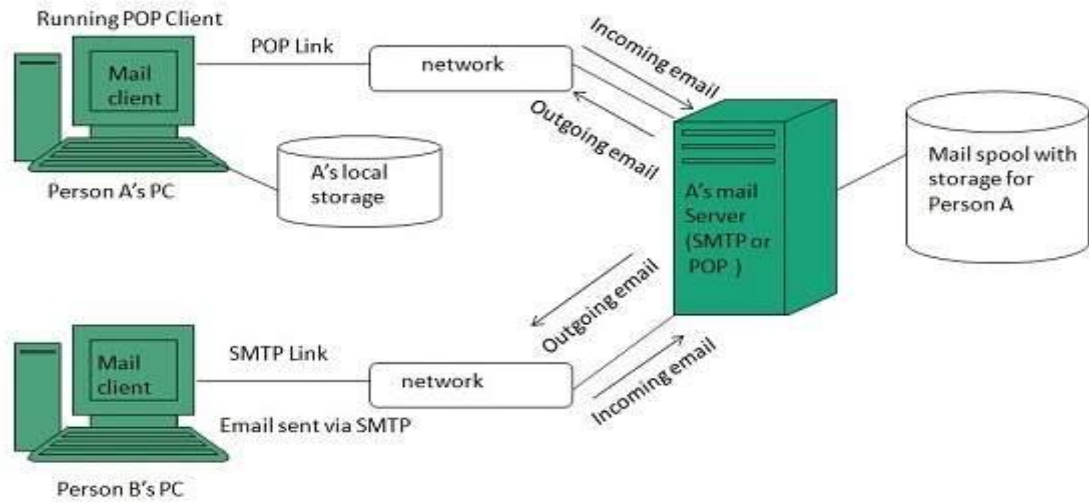
Following example will take you through the basic steps involved in sending and receiving emails and will give you a better understanding of working of email system:

- Suppose person A wants to send an email message to person B.
- Person A composes the messages using a mailer program i.e. mail client and then select Send option.
- The message is routed to **Simple Mail Transfer Protocol** to person B's mail server.
- The mail server stores the email message on disk in an area designated for person B.

The disk space area on mail server is called mail spool.

- Now, suppose person B is running a POP client and knows how to communicate with B's mail server.
- It will periodically poll the POP server to check if any new email has arrived for B. As in this case, person B has sent an email for person B, so email is forwarded over the network to B's PC. This is message is now stored on person B's PC.

The following diagram gives pictorial representation of the steps discussed above:



Snapshots

```
C:\Users\Somendra\Lab>java EmailClient
Welcome to Email Client.....

Choose Login Type:
1.Gmail Account
2.Outlook Account
3.Yahoo Account
4.Exit
1
Enter username: somen2995@gmail.com
Enter password: [REDACTED]
Logged in successfully

Choose Option:
1.Compose Mail
2.Inbox
3.Logout
1
To: mohit159.159@gmail.com
Subject: project
Message: Hello mohit, when we will do term project?
Message Sent
```

```
1
To: mohit159.159@gmail.com
Subject: project
Message: Hello mohit, when we will do term project?
Message Sent

Choose Option:
1.Compose Mail
2.Inbox
3.Logout
3
Logged out successfully

Choose Login Type:
1.Gmail Account
2.Outlook Account
3.Yahoo Account
4.Exit
1
Enter username: somen2995@gmail.com
Enter password: [REDACTED]
Logged in successfully

Choose Option:
1.Compose Mail
```

Logged in successfully

Choose Option:

1.Compose Mail

2.Inbox

3.Logout

2

DEBUG POP3: server doesn't support TOP, disabling it
messages.length---250

Email Number 0

Subject:

From: "somen2995@gmail.com" <somen2995@gmail.com>

Email Number 1

Subject: Photo

From: "somen2995@gmail.com" <somen2995@gmail.com>

Email Number 2

Subject:

From: "somen2995@gmail.com" <somen2995@gmail.com>

Email Number 3

Subject: Somendra Send us your birthday & get an advance gift from us

From: Bag It Today <todayshop@intoday.in>

Email Number 19

Subject: Declared Winner

From: Bag It Today <salesupdate@bagittoday.com>

Email Number 20

Subject: Somendra, You are appointed as Prime Minister

From: Prateebha <news-letter@ibibopromotion.com>

Email Number 21

Subject: Somendra, get 61% Off on San Diego Polo Club Watch, get 24% Off on Nike
, get Upto 62% Off on Ray-Ban, YSL, DKNY, Moschino, Police, Ted Baker & Tom Ford
, get upto 75% Off on Reebok apparels, get over 30% Off on Ecko Unltd and get 3
3% Off on Weishidun

From: Bag It Today <salesupdate@bagittoday.com>

Email Number 22

Subject: Dhamaka Deal of the Day: Somendra, get Lotto Shoes worth Rs 3999 for j
ust Rs 999

From: Bag It Today <salesupdate@bagittoday.com>

Email Number 23

Subject: Somendra 79% off on special summer dhamaka kit

From: Bag It Today <salesupdate@bagittoday.com>

Conclusion

This Email Client Software can be implemented to fulfill all the client requirements. The system interface is very user friendly, and the overall system has been successfully tested. It has a broad future scope as new features can be incorporated in the present proposed system. The system can be used for online sharing of data without the involvement of authority by which users can read message from any part of the world.

References

JavatPoint:

<https://www.javatpoint.com/example-of-sending-email-using-java-mail-api>

Tutorials Point

https://www.tutorialspoint.com/javamail_api/javamail_api_smtp_servers.htm

GeeksforGeeks

<http://www.geeksforgeeks.org/socket-programming-in-java/>