

# EMS: The Talking Mail Service

Utkarsh Goel, Kanika Shah, Shailendra Singh, Mohammed Abdul Qadeer  
Department of Computer Engineering  
Zakir Husain College of Engineering and Technology  
Aligarh Muslim University, Aligarh 202002, India  
{utkarshgoel, kanikashah, shailendrasingh, maqadeer}@zhcet.ac.in

**Abstract**—The application brings the facility which every person need in his each phase of life. It has facility of sending electronic mails with Google, Yahoo, MSN, AOL user accounts. This client is very similar to the other email clients but it can be used by anyone without the need of configuring differently for different users. It automatically sets its configuration depending upon the account type of the user that is currently logged in. Further this application consists of a feature that fetches user e-mails from the server. The fetching either POP3 or IMAP depends on the type of email server that is used and also on the user account settings. Only one at a time is fetched from the email server so as to avoid flooding of e-mails. It can write down your emails itself and can even read out your incoming messages.

**Keywords**—Send and receive emails, POP3, IMAP, SMTP.

## I. INTRODUCTION

- The first E-Mail client that speaks out E-Mails.
- A built in Text to Speech and Speech to Text converter.
- This application deals with services such as sending and receiving emails for Gmail, Yahoo mail, AOL and MSN only.
- The service with Gmail supports both IMAP and POP3 for incoming messages.
- The service with email accounts other than Gmail support only POP3 for incoming messages.
- Further, the application can send emails using respective outgoing mail servers.
- It also displays Inbox and Unseen emails separately.
- Moreover, it fetches only one email
- This service does not need any browser to be opened on Desktop/Laptop.

## II. APPLICATION DEVELOPMENT

### A. System Requirements

The Desktop application can run on any Windows operating system and need .NET framework installed.

### B. Application Architecture

The application architecture mainly deals with authenticating user with his respective E-Mail account and using it as a source for sending and receiving E-Mails. After user authentication is complete, the application transfers all the resources such as E-Mails and other documents related to it to the connected user.

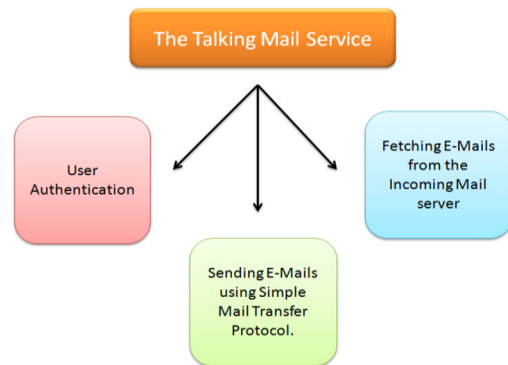


Figure 1: Contents.

The above figure presents an overview of the services/features contained in the application.

## III. DISCUSSION

Hotmail, Yahoo! Mail, Gmail and other providers are basically email services designed to provide email mailbox access directly from the web. However, going online and logging on to their sites is not always the most convenient way for reading and sending emails.

Advantages of using this application over other well known E-mail Clients:

- The user just need to speak his text and the application will enable the microphone and will convert it to desired text.
- It also reads out the user's incoming mails and even prompts the user by an alarm if the user receives a new E-Mail.

- **Incredibly Small:** The download size of this application is just a fraction of other E-mail clients such as Apple mail, Microsoft Outlook.
- **Breezing Fast:** The application launches instantly without any delay. The user is not forced to view an annoying splash window displaying company logo, author names, etc.
- The application highly respects the security and privacy of users and will never connect to the Internet without users' permission.

Based on the tests that measure the system performance, the application has some strong and weak features:

#### User Authentication:

- User authentication is a means of identifying the user and verifying that the user is allowed to access some restricted service.
- A user must be identified as a system administrator in order to access documents stored in the server to whom he wants to connect.

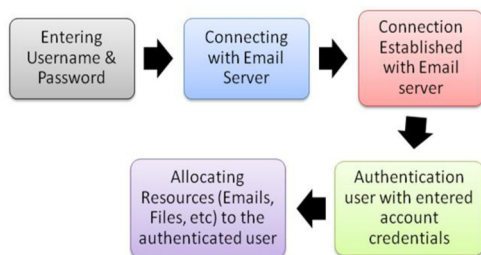


Figure 2: Connection with Email Server.

The above figure presents an overview of connecting with the email server to read and send emails from the logged in user account.

- When a user logs in to a network computer account, the server verifies that he is authorized to use its resources, and, additionally, that he is the one who owns that particular set of resources (files, e-mail, and so on), by giving the correct user id and password.

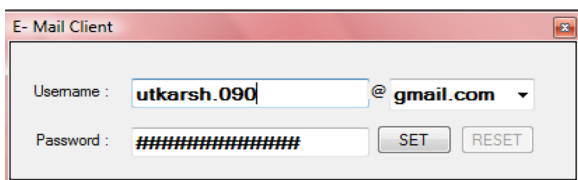


Figure 3: User Authentication.

The above figure displays the interface that is used by the user to sign in and to use his services to send and receive emails.

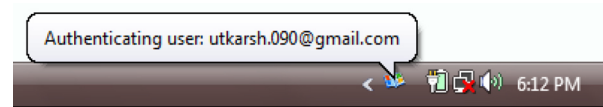


Figure 4: Notification of Authentication.

The above figure displays notification on the taskbar about the details of current user that wants to sign in to E-Mail service.

This notification fades automatically after two seconds and maintains this text until another operation is Performed with this service.

In case the user wants to view his status he just needs to bring his cursor over the icon and his status will get displayed.

#### Compose E-Mail:

- This is the server used only to send emails (to transport them from your email client program to the receiver).
- Most outgoing mail servers are using the SMTP protocol (Simple Mail Transfer Protocol) for sending emails.

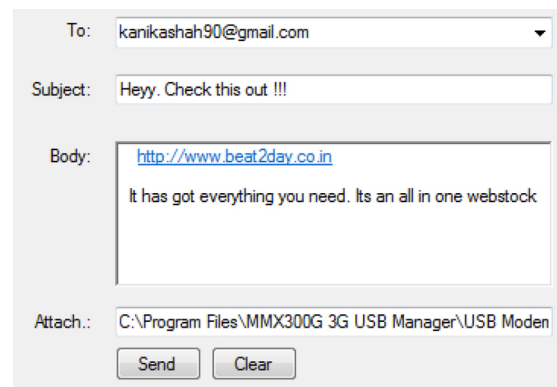


Figure 5: Compose E-Mail.

The figure above displays the interface used to compose email. Moreover, the application tells the user about each state of email sending procedure. It displays the success message when the email is successfully delivered to the targeted email address.

All the emails that will be sent will be formatted as HTML emails.

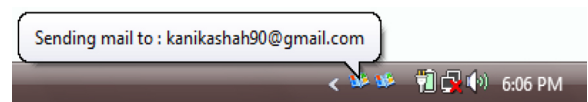


Figure 6: Notification of Sending SMS.

The above figure displays a notification in the taskbar that the mail is in the process of sending to the targeted email address.



Figure 7: Contents.

The above snapshot displays the notification when an E-Mail is sent successfully and you can send now send more mails.

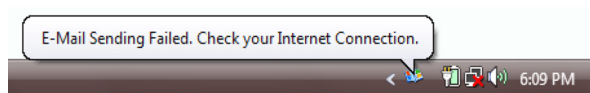


Figure 8: Notification of Email Sending Failure.

The above figure displays a notification in the taskbar that the mail cannot be delivered to the targeted email address because the user is not connected with internet.

#### Incoming E-Mails:

- This phase of application serves only with the Inbox and the unread emails.
- Easy and fast switching between Unread and Inbox categories of E-Mails.
- Also, only one email at a time is displayed to the user so as to cover all the emails in a small area of the application frontend.

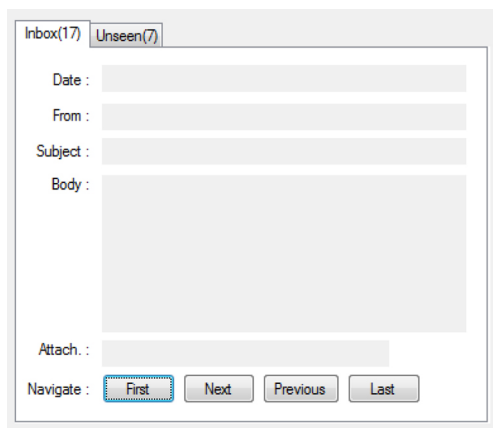


Figure 9: Incoming Mails.

The figure above displays the interface that is used to fetch incoming emails. It also let user know about the number of mails that are unread and are present in his inbox.

- Navigation of emails is made easy by adding four features which is not found in any of the well known e mail client till date.
- It brings the facility to view First, Last, Very next and the previous email.

- The same feature exists when the user wants to view his unread emails.
- For downloading any attachments, the user is provided with a link named "Download", which blinks when the fetched email contains any attachment with it.

#### Fetching E-Mails:

- For fetching E-Mails, user is provided with four options (First, Last, Previous, Next) which means he can directly jump to the specified E-Mail without downloading other E-Mails.
- Further, as all the E-mails are not displayed, it saves time and also requires less internet speed and bandwidth.

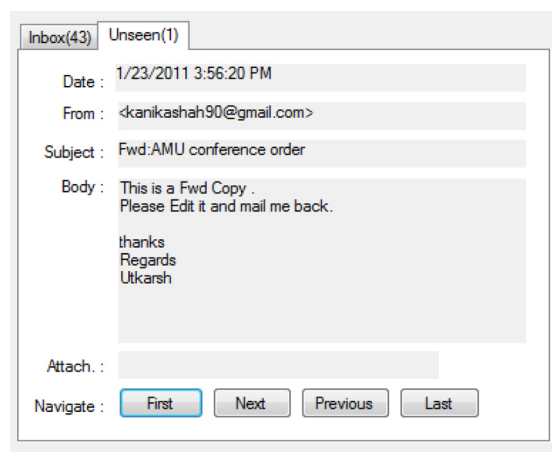


Figure 10: Notification of Email Sending Failure.

The figure above displays the interface that is used to fetch unread incoming emails.

#### Speech to Text:

- This feature enables user to avoid their keyboards just communicate wirelessly without touching the Desktop/Laptop.
- This helps user to compose and read E-Mails both.
- Further, if a user does not want to use this service, he can disable the speech recognition and continue using his keyboards.

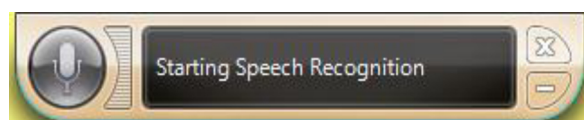


Figure 11: Speech Recognition.

The figure above displays the interface that is used to convert speech into text and text into speech for reading and writing emails.

Arrival of new Email:

- When a new email arrives the user is alarmed with a message that a new E-Mail has arrived and he must check it.
- The alarm also tells about the sender of the new E-Mail and the Subject of E-Mail.
- If the user ignores this alarm or in other words the user was not able to hear the alarm due to noisy environment, the alarm rings after every two minutes and automatically increases its volume after a regular period of interval.
- This feature is useful when user leaves his desk for a period of time and wants the application to handle the notification of incoming emails.
- For this he must activate the alarm facility so that application knows that the user is not on his desk.

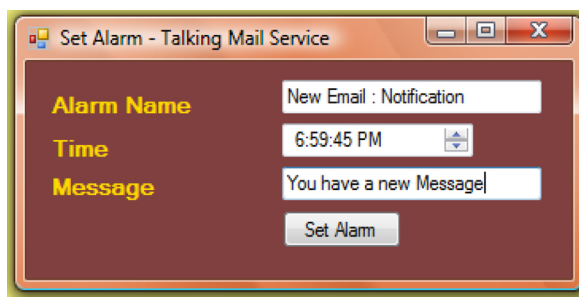


Figure 12: Set Alarm.

The figure above displays the interface that is used to set alarm for the new incoming E-Mails.

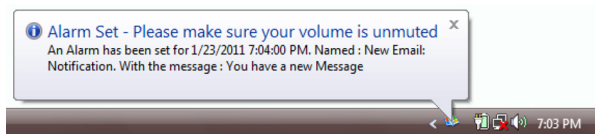


Figure 13: Set Alarm.

The above figure displays a notification in the taskbar that the user has set an alarm with his specified message and at specified time. If the user forgets that an alarm has been set long ago, then in that case the alarm window will become on the top of the screen and will not disappear until the user stops the alarm.

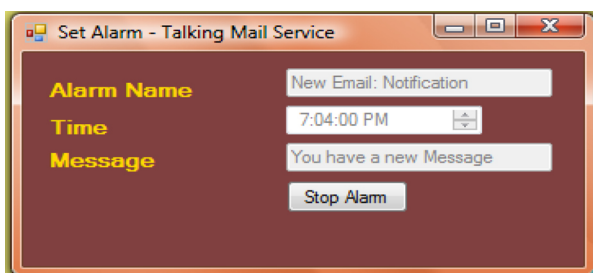


Figure 14: Ringing Alarm.

The figure above displays that a new E-Mail has arrived and the user should stop the alarm and check his new mail if he wants.

#### IV. IMPLEMENTATION

The Desktop application is created with Microsoft Visual Studio using VB.NET as programming language and needs .NET framework for its execution.

The term "email server" refers to the two servers required for sending and receiving email, i.e. the SMTP and POP servers.

Incoming Mail Server (POP3, IMAP, HTTP)

- The incoming mail server is the server associated with you email address account.
- There cannot be more than one incoming mail server for an email account.
- In order to access incoming messages, one needs an email client: a program that can retrieve email from an email account, allowing a user to read, forward, delete, and reply to email messages.
- Depending on the mail server, one can use a dedicated email client or a web browser. The mail is held in storage on the incoming mail server until it gets downloaded.
- Once it has been downloaded from the mail server it cannot be downloaded again. In order to download Email, one must have the correct settings configured in Email client program

Outgoing Mail Server (SMTP)

- Depending on the network settings, the outgoing mail server can belong to your ISP or to the server where the user setups his email account.
- As an alternative, he can use a subscription based SMTP server, which will allow him to send emails from any email account he already own.
- Due to anti-spam reasons, most of outgoing mail servers will not let him send emails if he is not logged on their network.
- An open-relay server will allow the user to use it for sending emails, no matter if he belongs to its network group or not, thus it is a heaven for spammers.

Email Ports:

For networks, a port means an endpoint to a logical connection. The port number identifies what type of port it is.

Here are the default email ports:

- POP3 - port 110
- IMAP - port 143
- SMTP - port 25
- HTTP - port 80
- Secure SMTP (SSMTP) - port 465
- Secure IMAP (IMAP4-SSL) - port 585
- IMAP4 over SSL (IMAPS) - port 993
- Secure POP3 (SSL-POP) - port 995

## V. ALGORITHM

### Sending Email

- Create an object of type MailMessage(), which represents an E-Mail message that can be sent.
- Set the "From" address of this E-Mail address.
- Set the "Subject" line for this E-Mail message.
- Set the message Body.
- Set a value indicating that the message body in HTML.
- Get the attachment collection used to store data attached to this E-Mail message.
- Create an object of type SmtpClient(), which allows application to send E-Mails by using Simple Mail Transfer Protocol.
- Set the name or IP address of the host used for SMTP.
- Specify whether the SmtpClient uses the secure socket layer (SSL) to encrypt the connection.
- Create an object of type NetworkCredential(), which provides credentials for password-based authentication schemes.
- Set the username associated with the credential.
- Set the password for the username associated with the credential.
- Set the credentials used to authenticate the sender.
- Set the Port used for SMTP transactions.
- Send the specified message to an SMTP server for delivery.

### Fetching E-Mails through IMAP/POP3

- Create an object of type IMAP/POP, which allows listing, moving, copying, downloading and deleting E-Mails from IMAP/POP server.
- Connect to the IMAP/POP server by providing server address and port number.
- Select the specified folder from the mail box.
- Create an object of type IMail, which provides simple access to email messages.
- Close the connection and dispose the object.

### Downloading Attachments

- Create an object of type MimeData, which represents MIME entity that contains raw byte data.
- Save the attachments at the specified location.

### Speech to Text and Text to Speech

- Create an object of type SpVoice(), which acts as an interface between the user's voice and the microphone and helps in converting the speech into appropriate text.
- For text to speech conversion, define the speaker rate and the message to be spoken.
- For speech to text, create an object of type SpSharedRecoContext() and use it for streaming user's voice into the microphone and convert it to appropriate text.

## VI. CONCLUSION AND FUTURE WORK

### Conclusion:

- Since bandwidth being the bottleneck and every single browser increases bandwidth requirement hence this application intentionally reduces the later.
- Efficient use of E-Mail service.
- Easy and fast access of E-Mails.
- The concept of new E-Mail notification makes this application different in use with other E-Mail clients.

### Future extensions:

- Displaying Sent E-mails as well.
- Modifying the application so that it can run on platforms other than Microsoft Windows.

## REFERENCES

- [1] <http://www.emailaddressmanager.com/tips/email.html>, accessed on November 11, 2010.
- [2] <http://www.emailaddressmanager.com/outlook/email-autoresponder.html>, accessed on November 18, 2010.
- [3] <http://www.emailaddressmanager.com/outlook/email-autoresponder.html>, accessed on December 01, 2010.
- [4] <http://www.lesnikowski.com/mail/faq.aspx>, accessed on December 13, 2010/