**Meeting on 04/05/23 at 11:30 and ended at 15:00**

1. The application shall act as an email client, where a user is allowed to send or receive emails.

2. The application shall make use of IMAP and SMTP protocols to achieve requirement 1 above.

3. The application shall allow a user to login using their email and password.

4. The application shall allow a user to indicate what emails they need to be read to them.

5. The user shall have the ability to select the preferred voice from a list of 2-3.

6. The application shall allow the user to select what if all unread emails in the Inbox should be read

7. The application shall allow a user to select specific times when emails should be read. E.g. Read all unread emails to me by 5pm.

8. The application shall be able to work with multiple email addresses. E.g. - Work email and personal email.

9. The application shall be able to work with any email provider that supports IMAP and SMTP.

**LIST OF EVERYTHING WE NEED TO DO:**

* **non-coding:**
* Write script for presentation
* Record presentation (Monday 8th May)
* Write documentation
* Adding comments and cleaning up code
* **coding:**
* TTS sounds weird for apostrophe ‘s so we need to append -s to the end of words and remove apostrophe
* ~~Website html stuff~~
* Support for changing name of wav file (so that there are multiple files with different voices)

**THINGS THAT NEED TO BE IN THE PRESENTATION:**

* Get the *app password*
  + > Google Account > Manage Accounts
    - Enable 2-step authentication if not yet: > Security > 2-step authentication
    - Search “app password” in the search bar.
    - Select “mail” for *app* and your current device for *device*.
    - Click “generate”.
* Login
  + First Time login
    - Open the localhost:8083/audioEmail/
    - Input account(example): [crowds723@gmail.com](mailto:crowds723@gmail.com)
    - Password(example): nbdnhxvaapxhkfql
    - Choose(example): work
  + Log out and log in with another account
    - Account
      * Input account: [mzhang122@sheffield.ac.uk](mailto:mzhang122@sheffield.ac.uk)
      * Password: xosuzcxjecwggcmg
      * Choose: personal
* Send Mail (two accounts in two browsers)
  + One account click write mail, send message
  + Another account click Sync OR logout and login again.
  + Delete some emails
* Settings
  + Choose Voice
    - Read an email
  + Read
    - Show him we can convert the mail in advance(Or we can convert it in advance).
      * Set 18:10( assume now is 17:10 )
    - And read one short mail directly
    - Then read one long mail
    - Scheduled Task
      * Set 17:11(assume now is 17:10)
      * Explain Read All unread is the same with the scheduled tasks, but need to wait longer
* **Note:**
  + **Don’t choose articles which have strange words!!!!**

## Features

* Regarding what we have done, we completed all requirements and we made some of the client-suggested changes from the Week 8 meeting, for example:
  + Every time the user logs in, the account syncs with the server and retrieves any new mail
  + ‘Sync’ is now displayed as a separate button, as in, it’s not listed with the links to the other pages, and thus the purpose of this function is clearer
* However, there are a few client-suggested changes that we did not complete. These were…
  + Hashing the password, and instead of hashing, the user needs to generate an APP Password several times.
    - As password hashing is irreversible, we found it difficult to incorporate into our app. This is because MyAudioEmailr needs the account to repeatedly connect to the server to retrieve emails, meaning the password needs to be repeatedly validated against an unhashed version of the password in the database
    - However, since the MyAudioEmailr login uses an APP Password obtained from Google, not the actual account password, we believe this already meets the user's security needs
  + The client suggested that we make switching between accounts easier (e.g. expand window/box on the side), as well as using a separate page to write mail - we were not able to complete these suggestions, but they are not part of our core functionality. It is more important for users to be able to switch accounts and write emails, which are both possible.
  + We also did not complete line by line processing, which the client suggested would speed-up the app. We were not able to complete this partly because we spent more time fixing major bugs.
    - We used a different method to speed-up processing, which achieved a similar result. Rather than processing line-by-line, after the user sets the schedule, the system will start converting mail to speech one hour in advance. We also used sockets to send paths to the wav files containing the synthesised speech. Transfer via sockets to the client is immediate, and thus this sped up processing.

In summary, we met all of the requirements, and of the additional changes, we were able to complete most of them or find another solution.