

---

# **Software Requirements Specification**

**for**

## **Email Client, Intro to SE**

**Version 1.0 approved**

**Prepared by Nick H., Jonathan S., Terryuntae G., and Brett H.**

**Group 9, Section 2**

**September 4, 2020**

# **1. Introduction**

## **1.1 Purpose**

The purpose of this project is to build an email client web application to enable a user to perform various tasks that are specified in this document. This project is developed to fulfill lab requirements for Mississippi State University's Intro to Software Engineering for the Fall semester, 2020. This document includes the requirements specification for the entirety of the product for version 1.0.

## **1.2 Document Conventions**

The following are conventions used in this document:

- Every requirement is given its own priority.
- Blue, underlined text indicates a hyperlink to a webpage.
- Some acronyms are used. Acronyms are given in all capital letters. Refer to appendix A for associated meanings.

## **1.3 Intended Audience and Reading**

This document is intended for the group 9 development team and the TA for Intro to Software Engineering. The document is organized as follows: Section 2 contains the overall description of the product which includes the major functions of the product, its operating environment, and constraints on the design and implementation. Section 3 provides detailed descriptions of the system's features. Section 4 lists the criteria by which the operation of the system can be assessed. Lastly, section 5 lists the system requirements not listed elsewhere in the document. It is recommended that the reader examine the document in the order it is presented.

## **1.4 Product Scope**

The product being developed is an email client with a GUI using a web application framework. Version 1.0 of the developed product will allow a user to login and access their emails as well as perform typical emailing tasks such as composing and editing emails, searching and forwarding emails, and sending attachments.

## **1.5 References**

It is the project team's intention to develop the product using the [Python](#) language for backend development and the web application framework [Flask](#) for frontend development. Python is a popular, general-purpose programming language. Flask is a [micro web application framework](#) written in the Python language.

## **2. Overall Description**

### **2.1 Product Perspective**

The email client will be a new self-contained product. An email client with a graphical user interface using a web application framework using the Python language. The project will follow an agile paradigm, being developed in four sprints.

### **2.2 Product Functions**

The email client will perform the following functionalities:

- A user should be able to login into his/her account to access his emails.
- A user should be able to compose, edit and send an email to other users.
- A user should be able to search and forward emails to other users.
- A user should be able to send attachments (image, files etc.) along with an email

### **2.3 User Classes and Characteristics**

The client only supports one type of privilege that being the customer. The customer will have access to all the functionalities listed above (see **2.2 Product Functions**).

### **2.4 Operating Environment**

Operating environment for the client is as listed below:

- Laptop/Desktop
- Smartphone
- Tablet

### **2.5 Design and Implementation Constraints**

A few design and implementation constraints have been identified:

- The client will be coded in the Python language (Flask).
- The client will be limited to platforms that can access a web application.

## **3. System Features**

### **3.1 Login access to emails**

#### 3.1.1 Description and Priority

This feature will allow the user to login and gain access to the email client. The priority of this feature is high given that it is required for general use of the client.

#### 3.1.2 Stimulus/Response Sequences

- Request user login
- Display user emails

#### 3.1.3 Functional Requirements

REQ-1: The user must have an account.

REQ-2: The user must enter the correct login or client will display an error requesting the appropriate login.

### **3.2 Compose, Edit, and Send Emails**

#### 3.2.1 Description and Priority

This feature will allow the user to compose, edit and send emails to other users. The priority of this feature is high based on it being a core function of the email client.

#### 3.2.2 Stimulus/Response Sequences

- Toggle composition of emails
- edit email (formatting,signature,links, attachments)
- address email to recipient
- send to recipient

#### 3.2.3 Functional Requirements

REQ-1: The user must address the email to a valid recipient or the client will return a “recipient not found” error.

REQ-2: Attachments must be within the constraints of valid types.

### **3.3 Search and Forward Emails**

#### 3.3.1 Description and Priority

This feature will allow the user to search and forward emails to other users. The priority of this feature is medium based on it being an additional function that makes the client more user friendly.

### 3.3.2 Stimulus/Response Sequences

- Input valid search criteria to be found
- Display content that includes requested data
- Send stored emails to new recipients

### 3.3.3 Functional Requirements

REQ-1: Searches that yield no results will return none found.

REQ-2: Emails that are forwarded to invalid recipients will return an error.

## 3.4 Send Attachments

### 3.4.1 Description and Priority

This feature will allow the user to send attachments to other users. The priority of this feature is medium based on it being an additional function for working with other file types.

### 3.4.2 Stimulus/Response Sequences

- Request upload of an attachment.
- Display successful uploads.

### 3.4.3 Functional Requirements

REQ-1: Attachment must be a valid type.

REQ-2: Attachment must be within the size limit.

## 4. Other Nonfunctional Requirements

### 4.1 Performance Requirements

For performance requirements, the system will need quick response times while dealing with sending and receiving emails, uploading attachments to an email, and logging in.

### 4.2 Safety Requirements

Email accounts can typically hold significant personal information, along with being tied to important online accounts for the user. As such, security measures need to be in place to remove the possibility of user information being used or stolen.

### 4.3 Security Requirements

The web application will require the use of passwords as a means of identity authentication. These passwords will not be stored on the database; instead, the database will store hashes of the passwords as a means of security.

## 4.4 Software Quality Attributes

As developers, the preferred attributes of this web application will be a combination of testability and correctness. These attributes are best for the application due to the nature of the project.

## 5. Other Requirements

This project will require the use of a DB as a means of storing user data and storing email content. The DB is expected to be stored locally on a user's system.

## Appendix A: Glossary

The following table lists the acronyms used in this document and their associated meanings.

DB	Database
GUI	Graphical User Interface
TA	Teaching Assistant

Table 1: Acronyms and associated meanings.

## Appendix B: Analysis Models

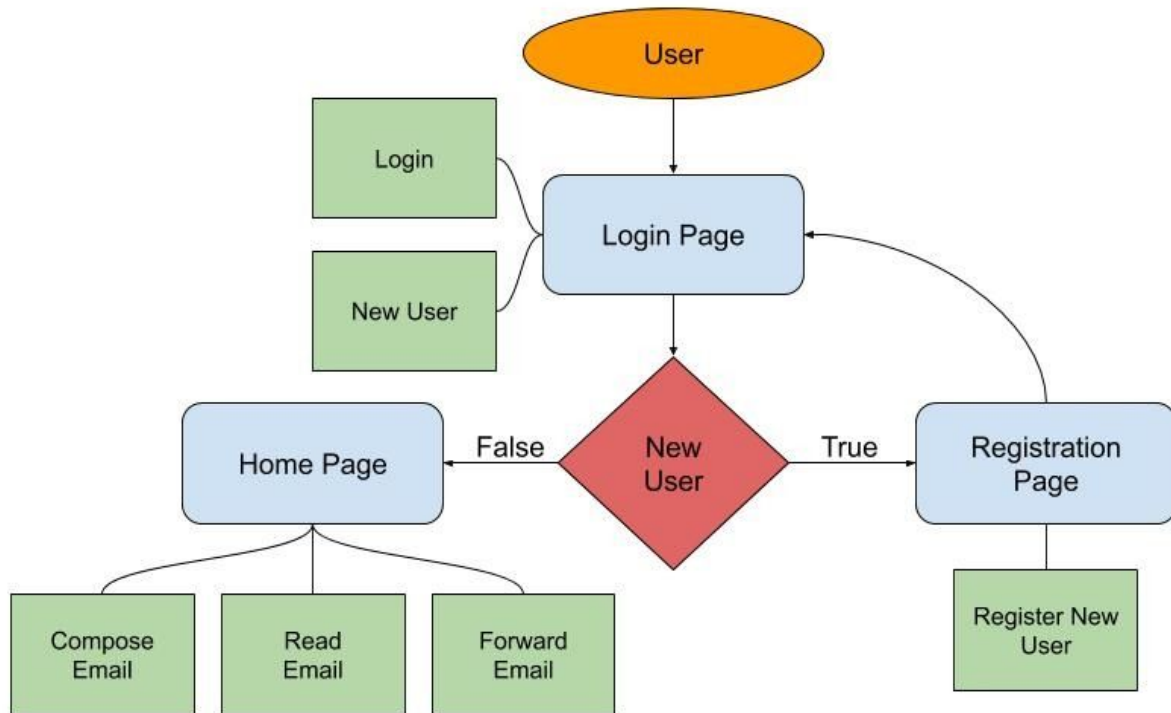


Figure 1: A simple diagram of the email client. Blue nodes indicate pages in the web application, the red node is a decision node, the orange node indicates the user, and green nodes indicate actions that can be performed on the webpage.

## Appendix C: To Be Determined List

1. How should the final software be delivered?
  - a. A [Docker container](#) is a possibility.
2. Should the email client support every email server?
  - a. Some servers have different security permissions, like [Gmail](#).
3. How much data should be stored locally?
4. How often should the email client update to receive new emails? Should this be user determined?