

Software Requirements Specification

for

G7 Email

Version 1.0 approved

Prepared by Courtney Stovall, Jonathan Harjono, Breshia Donald

Group 7

9/6/21

1. Introduction

1.1 Purpose

The purpose of G7 Email is to send, forward, reply, and read emails sent from other email addresses.

1.2 Document Conventions

S3	Simple Storage System
API	Application Programmable Interface
AWS	Amazon Web Services

1.3 Intended Audience and Reading

This document is intended for developers, project managers, testers, and documentation writers. This SRS contains the scope, systems, and other necessary requirements. This SRS is organized by sections starting with the overall description and moving to the specific system details.

1.4 Product Scope

The scope behind G7 Email is Angular, Typescript, and Google's Gmail API. Angular is used for created a single page website that the user can interact with, Typescript is the code that interacts with Google's Gmail API, and Google's Gmail API send and serves the user emails

1.5 References

Angular:

Angular is used to create single page web applications, using child and parent components.

<https://angular.io/docs>

Google's Gmail API:

Used to receive and send emails.

<https://developers.google.com/gmail/api/guides/sending>

2. Overall Description

2.1 Product Perspective

The email system has these following functionality:

- Receiving an email:
 - Opens the received email in the client
 - Enables the user to download the attached files
- Sending an email:
 - Forwarding an email
 - Attaching a file to the email
- Logging into the software:
 - Signing into the user account
 - Signing out of the user account
 - User authentication

2.2 Product Functions

- Write Email
- Send Email
- Forward Email
- Reply to Email
- Read Email
- Sign into Email Account
- Sign-Out of Email Account

2.3 User Classes and Characteristics

The user should be able to view his email, create and send emails to other email addresses. This system will be consistent for all of the users accessing this software. The user should be able to do the following functions:

- Receive an email
- Send an email
 - Forwarding
 - Replying
 - Attaching a file

2.4 Operating Environment

The software will operate on the AWS server inside S3 bucket. A software component that this software will be coexisting with is Google's Gmail API.

Operating environment for the airline management system is as listed below.

- Google Gmail API
- Operating system: AWS
- Platform: Angular/Typescript

2.5 Design and Implementation Constraints

1. Host the system using AWS server
2. Leverage Google's Gmail API for the basic functionality
3. Use AWS S3 Bucket to open files.

3. System Features

3.1 Send Email

3.1.1 Description and Priority

High priority feature that allows the user to send an email to another email. The cost and risk are both around 3.

3.1.2 Stimulus/Response Sequences

User clicks the button to compose a new email. A window pops up. The user enters the email of the receiver, a subject, and a desired message. Then, the user clicks send to send the message to the receiver. The system takes the email and sends it to the desired email.

3.1.3 Functional Requirements

TBD

REQ-1: TBD

REQ-2: TBD

3.2 Forward Email

3.2.1 Description and Priority

Medium priority feature that allows the user to resend the email thread.

3.2.2 Stimulus/Response Sequences

User clicks the forward email button. A window pops up containing the email thread. The user enters the email of the receiver. Then, the user clicks send to send the email. The system sends the email thread to the desired email.

3.2.3 Functional Requirements

TBD

REQ-1: TBD

REQ-2: TBD

3.3 Read Email

3.2.1 Description and Priority

High priority feature that allows the user to read emails received.

3.2.2 Stimulus/Response Sequences

User clicks the email they want to read. A window pops up containing the desired email message.

3.2.3 Functional Requirements

REQ-1: The software receives incoming email

REQ-2: The API decodes the email

REQ-3: The software takes the decoded email and displays it to the user through HTML.

3.4 Sign into Email Account

3.2.1 Description and Priority

High priority feature that allows the user to sign into their account to view and send emails.

3.2.2 Stimulus/Response Sequences

User enters their email and password into the specified fields, and the system logs them in.

3.2.3 Functional Requirements

The Gmail API allowed the user to sign into email account

REQ-1: TBD

REQ-2: TBD

3.5 Attach a File to Email

3.2.1 Description and Priority

High priority feature that allows the user to attach a file to an email to be sent.

3.2.2 Stimulus/Response Sequences

User clicks the create an email button. A new email window pops up. The user clicks the attach file button. The system pulls up the file manager, and the user selects the file to be attached. The system adds the selected file to the email.

3.2.3 Functional Requirements

TBD

REQ-1: TBD

REQ-2: TBD

3.6 Sign Out of Email Account

3.2.1 Description and Priority

Medium priority feature that allows the user to sign out of their email account.

3.2.2 Stimulus/Response Sequences

User clicks the sign-out button. The system logs them out of the account.

3.2.3 Functional Requirements

TBD

REQ-1: TBD

REQ-2: TBD

4. Other Nonfunctional Requirements

4.1 Performance Requirements

Since G7 Email plugs directly into Google's Gmail API, it's performance is solely based on Google's servers. Also, G7 Email is designed to take the load off the user front-end and pass that load to Google's Gmail API

4.2 Safety Requirements

One concern is Cross site scripting. Which occurs when the user inputs malicious html or javascript code into an input field and the program executes it to make the program do things that it's not supposed to do.

4.3 Security Requirements

Regarding security or privacy issues, we are leveraging Google's Gmail API security protocols. To authenticate a user, we are also leveraging the Gmail API's user authentication functions.

4.4 Software Quality Attributes