

Balkar Singh Rana
COMP 348
June 20th 2018

A. Reflective Component

How did I begin to lay out this assignment, considering the requirements?

To begin with, I began searching the COMP 348 group on the landing page for previous students that have worked on this assignment. This was to see how they have done this assignment and what approach they took as well as if there is something I need to know prior to beginning this assignment. I also started to google for similar programs to see what other APIs exist for such requirements to get a general idea of what and where to begin.

What are some general research questions I should use as my overall search terms to find out information on the topic?

I believe for this part, some of the research questions I can think of from brain storming a little are as following:

1. What is Java Mail API?
2. How to make a console based Java Mail application?
3. How does Java Mail API work?
4. What exactly is SMTP/IMAP/POP3
5. How does SMTP/IMAP/POP3 work?

Where should I look for information?

<https://javaee.github.io/javamail/FAQ>

This is Oracle's original github page with the frequently asked questions for the Java Mail API. It also includes many answers to my above research questions such as what is IMPA/SMTP/POP3, etc. It also has the installation and configuration guide on there as well.

<http://www.oracle.com/technetwork/java/index-138643.html>

This is the page to download the jar file for the latest JavaMail API. This page also has a guide for service providers in a pdf file.

<https://stackoverflow.com/questions/12496437/sending-mail-using-java-mail-api>

Also found this to see how a basic java mail code looks like to get an understanding of what my design will look like.

<https://stackoverflow.com/questions/16117365/sending-mail-attachment-using-java>

This stackoverflow example shows how to send attachments along with ur email.

How can I use the research information to begin to design and develop my programs for part C?

First of all, I figured that I need to use files for this application because a txt file will contain the credentials for the email, password, etc. I will read the file and read the credentials to store into private variables in the application. I will also need to read up on the oracles github page to understand the API and how to use its methods that perform the core actions to ensure the email is sent. For the attachments I found a little stackoverflow example that uses 1.4.7 version of the API that allows you to use attachments with your email and this is a great way for me to learn and incorporate it into my design.

What is the purpose for each program in part C?

I believe the first program is to get use to learn the basics of the JavaMail API and learn how to send basic emails using the built in api methods and its objects.

The second part seems like is to teach us how to add attachments along the email which is quite essential for such an application because most emails now include some kind of attachments.

The third application seems to be to teach us how to receive emails and this seems a bit difficult from my research but not impossible.

When coding, what problems did I encounter?

When I was coding I didn't encounter many problems because I found great tutorials that I followed and applied my own logic to meet the assignment requirements. The only problem I did encounter was I couldn't figure out a way to parse the text file data into the fields required to send emails, however, after a good amount of research I figured out a way which was to use Maps to read the file and store it in accordingly.

B. Research Component

First of all, I googled "javamail api installation" and the first link that popped up was the github page for the javamail faq. Which had frequently asked questions regarding the API. So I went ahead and downloaded the latest jar file version 1.6.0 from the link given on the page. Then I decided to use IntelliJIdea for my IDE/compiler and it asked to add the jar file to the classpath which I didn't know the way around in IntelliJIdea so I googled and it was basically Command + ; (for mac users). This whole process took about 10 minutes or so to find the right material and installing it and setting up my project.

I researched the API's core classes required to complete this project. The following is what I found on www.tutorialsonpoint.com :

1. **Session Class** – This is basically a primary class of the API as this acts as the connection object for the API that handles configuration and user authentication.

2. **Message Class** – This class handles the content of the email that will be sent to the recipient.
3. **Address Class** – This class handles the actual email address where the message is to be sent to for example test@mail.com.
4. **Authenticator Class** – This class acts as the object that handles the authentication for the network connection.
5. **Transport Class** – This class is used as the transport method of the message. As it uses the SMTP protocol to send the email to the recipient.
6. **Store Class** – This is an abstract class that models the message store and its access protocol.
7. **Folder Class** – This is another abstract class that represents a folder for the mail messages.

I further investigated what exactly is SMTP which basically stands for Simple Mail Transfer Protocol as this is what is used in the Transport Class. It is basically used to transfer the RFC822-style content between different mail hosts. It is also used the most since its origination back in 1982. In addition, I also knew that I had to parse the information such as username, password, CC, and BCC from a text file into my application for the first two parts. So I began researching as to what different ways I can approach this problem. I ended up finding Mapping was the best solution for this that keeps my code neat and efficient as well and using some regex to determine the difference between the information fields.

Furthermore, I had to figure out a way to begin this assignment so I started planning it and looking at various tutorials on JavaMail API based applications on the internet. I ended up finding a few good tutorials for this project.

TUTORIAL #1

First one I found was: <https://www.javatpoint.com/example-of-sending-email-using-java-mail-api-through-gmail-server> which shows an example of sending an email through a gmail server with SSL. This tutorial was good because it jumped to the basics and didn't talk about other unnecessary topics that were not relevant to this assignment. Moreover, this tutorial also included the second part of the assignment which was to send an email with attachments: <https://www.javatpoint.com/example-of-sending-attachment-with-email-using-java-mail-api>

This tutorial was good because it discussed the steps of completing the application and it had a breakdown of it at the top of the tutorial. It also had good commenting for the code for the programmers to understand what and why does that piece of code exist which made it useful to work off this example for my own assignment with modifications. The other good thing about this tutorial was it also explained how to run the application using command line as well at the end.

Some of the downsides to this tutorial was that it could have been more detailed in terms of just commenting the code, rather explain what the code is doing exactly and

why it is there. As I personally thought the comments were very vague and not much detailed.

TUTORIAL #2

The other tutorial I found that I thought was very well written is:

https://www.tutorialspoint.com/javamail_api/ which is by tutorialsonpoint and as usual they make great tutorials for almost every programming language and every major API/Framework out there. Now this tutorial I did use for help with clarification of certain classes that were used in the project given by the API. As usual tutorialsonpoint did a great job with writing a well detailed tutorial that goes through each class being used in the API and its essentials. It also had UML diagrams to further explain the logic behind the JavaMail API. I personally couldn't find a flaw in this tutorial as I thought it was bang on and flawless.

TUTORIAL #3

The other tutorial I found was also quite good and is written by Oracle itself the ones that created the JavaMail API so you can't go wrong with that. It can be found at

<http://www.oracle.com/webfolder/technetwork/tutorials/obe/java/javamail/javamail.html>

Anyhow, this tutorial is great because it is very well detailed and shows you step by step using images and visuals for Eclipse. The only downside to this is it only teaches you how to send a simple text email and is a bit hard to navigate through.