

# Email Application For Server

COS301 Omega Notifications Team

March 2020

## 1 Introduction

This Application is designed to send emails to clients from the server regarding changes, updates, notifications or just general information about the mouth piece mobile app.

## 2 External Libraries

In order to create this Application some External Libraries was needed to be downloaded and included into the project.

These Libraries include:

1. javax.mail.jar  
The javax.mail to control all of the operations related to sending the email.
2. org.json.jar  
The org.json library is used to create, access and manipulate the JSON object that is passed as a paramater.

## 3 JSONObject

a JSON object will be used to send the information of the email to the application.

The JSON object must have the following fields

1. UserEmail  
contains a valid email adress of the recipient
2. UserName  
The user name of the recipient
3. MsgType  
Determines which template to use
4. Subject  
Used to set the subject of the email.

5. Message  
contains a string for update messages.

These can however be expanded as deemed necessary.

## 4 Classes

The application consists of the following classes that will be explained individually later on.

1. Email.java
2. SMTPAuthenticator.java
3. Message.java

These classes all interact with each other to do the simple task of sending an email.

## 5 Email.java

This class is in charge of composing an email with the internet properties, create the message to be send and sending the email.

### 5.1 Variables

1. String SMTP-HOST-NAME: holds the name of the email service host we will use.
2. String SMTP-AUTH-USER: holds the company email address that will be used to send the email
3. String SMTP-AUTH-PASSWORD: holds the password associated with the email address used

### 5.2 Methods

1. getSMTPHostName(): string  
returns the SMTP-HOST-NAME variable
2. getSMTPAuthUser() : String  
returns the company email address
3. getSMTPAuthPassword() : String  
returns the email address password
4. email(json: JSONObject) : void

(a) parameters:

json : a json object that will hold the information of the recipient and the details of the email such as Email Type, user name ,etc.

The email function will set the properties to be used when sending the email, call a method from Message class to create a message and send the email

## 6 SMTPAuthenticator.java

This class is used to create an Authenticator object to be used when setting the internet properties in the Email.email() function. It uses the SMTP-AUTH-USER and SMTP-AUTH-Password variables when creating the authenticator.

### 6.1 Methods

getPasswordAuthentication() : PasswordAuthentication  
returns the authentication object that was created.

## 7 Message.java

This Class is used to create the MimeMessage that will be used in the Email.email(:JSONObject) function. This MimeMessage holds all the information of the email to be send such as the: Reciever , Sender , subject , email content and more.

### 7.1 Methods

1. createMessage(:JSONObject) : MimeMessage creates the MimeMessage and sets the To,From,Subject and content (generated by the template methods) then returns the MimeMessage object.
2. getMessageContent(:JSONObject): String  
Determines which template to use depending on the json's "MsgType" field, calls a function to create the string and returns the generated message

### 7.2 Message Template Generator Methods

These Methods can be expanded to allow more templates to be generated by adding more Generator methods and including the MsgType that will be in the json into the switch statement in the getMessageContent(:JSONObject) function.

The following functions already exists.

1. getValidateMessage(:JSONObject) : String

2. getPasswordChangeMessage(:JSONObject) : String
3. getUpdateMessage(:JSONObject) : String

## 8 References

### 8.1 libraries

The external libraries that was used can be found and downloaded at the following websites.

1. javax.mail  
<https://javaee.github.io/javamail/>
2. org.json  
<https://jar-download.com/artifacts/org.json>

### 8.2 websites

The following websites was used to develop this email application.

<https://stackoverflow.com/questions/7760637/javamail-and-adding-link-to-text>  
<https://www.codejava.net/java-ee/javamail/send-e-mail-in-html-format-using-javamail-api>  
<https://www.rgagnon.com/javadetails/java-0538.html>  
<http://edelstein.pebbles.cs.cmu.edu/jadeite/main.php?api=javamailstate=classpackage=com.sun.mail.smtp>  
<http://edelstein.pebbles.cs.cmu.edu/jadeite/main.php?api=javamailstate=classpackage=com.sun.mail.smtp>  
[https://www.tutorialspoint.com/java/java\\_sending\\_email.htm](https://www.tutorialspoint.com/java/java_sending_email.htm)  
<https://www.javatpoint.com/example-of-sending-email-using-java-mail-api>  
<https://www.programcreek.com/java-api-examples/?api=javax.mail.SendFailedException>  
<https://stackoverflow.com/questions/36819393/i-am-trying-to-develop-java-mail-client-i-am-getting-error->  
[https://www.tutorialspoint.com/json/json\\_java\\_example.htm](https://www.tutorialspoint.com/json/json_java_example.htm)