

---

## TP : Envoi des mails avec Spring Boot

---

## SOMMAIRE

I- Objectifs : .....	3
II- Outils utilisés : .....	3
III- Développement de l'application.....	3
1. Génération du squelette de l'application moyennant Spring Initializr .....	3
2. pom.xml.....	4
3. application.properties .....	5
4. La classe @SpringBootApplication .....	5
IV- Les tests .....	6

### I- Objectifs :

- ✓ Envoyer un mail moyennant Spring BOOT

### II- Outils utilisés :

Dans ce TP, nous allons utiliser les outils suivants :

- ✓ Eclipse Neon avec le plugin Maven 3.x ;
- ✓ JDK 1.8 ;
- ✓ Connection à Internet pour permettre à Maven de télécharger les dépendances nécessaires (Spring Boot 2.2.0, ...) ;

### III- Développement de l'application

#### 1. Génération du squelette de l'application moyennant Spring Initializr

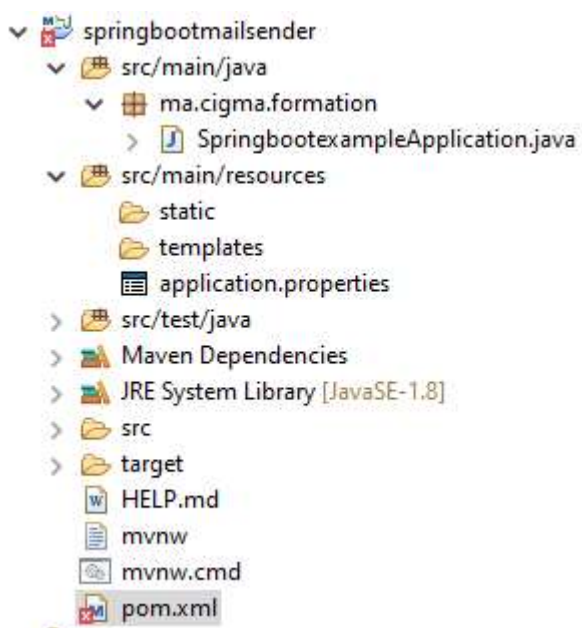
- Utiliser la plateforme SaaS <https://start.spring.io/> afin de générer le squelette de votre projet maven.  
Ajouter les dépendances suivantes :

```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-web</artifactId>
</dependency>

<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-mail</artifactId>
</dependency>
```

Décompresser ensuite votre zip et importer le projet maven dans Eclipse.

L'arborescence du projet est la suivante :



## 2. pom.xml

Le contenu du fichier pom.xml est le suivant :

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
  https://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <parent>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-parent</artifactId>
    <version>2.2.6.RELEASE</version>
    <relativePath /> <!-- lookup parent from repository -->
  </parent>
  <groupId>ma.cigma.formation</groupId>
  <artifactId>springbootmailsender</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <name>springbootexample</name>
  <description>Un exemple d'une application Spring BOOT</description>

  <properties>
    <java.version>1.8</java.version>
  </properties>

  <dependencies>
    <dependency>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-starter-web</artifactId>
    </dependency>

    <dependency>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-starter-mail</artifactId>
    </dependency>

    <dependency>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-starter-test</artifactId>
      <scope>test</scope>
      <exclusions>
        <exclusion>
          <groupId>org.junit.vintage</groupId>
          <artifactId>junit-vintage-engine</artifactId>
        </exclusion>
      </exclusions>
    </dependency>
  </dependencies>

  <build>
    <plugins>
      <plugin>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-maven-plugin</artifactId>
      </plugin>
    </plugins>
  </build>
```

```
</project>
```

### 3. application.properties

Configurer votre fichier application.properties comme suit :

```
spring.mail.host=smtp.gmail.com
spring.mail.port=587
spring.mail.username=mettez ici votre adresse mail
spring.mail.password=mettez ici votre mot de passe

# Other properties
spring.mail.properties.mail.smtp.auth=true
spring.mail.properties.mail.smtp.connectiontimeout=5000
spring.mail.properties.mail.smtp.timeout=5000
spring.mail.properties.mail.smtp.writetimeout=5000

# TLS , port 587
spring.mail.properties.mail.smtp.starttls.enable=true

# SSL, port 465
#spring.mail.properties.mail.smtp.socketFactory.port = 465
#spring.mail.properties.mail.smtp.socketFactory.class =
javax.net.ssl.SSLSocketFactory
```

### 4. La classe @SpringBootApplication

```
package ma.cigma.formation;

import java.io.File;
import java.io.IOException;

import javax.mail.MessagingException;
import javax.mail.internet.MimeMessage;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.boot.CommandLineRunner;
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.mail.SimpleMailMessage;
import org.springframework.mail.javamail.JavaMailSender;
import org.springframework.mail.javamail.MimeMessageHelper;

@SpringBootApplication
public class SpringbootexampleApplication implements CommandLineRunner {

    @Autowired
    private JavaMailSender javaMailSender;

    public static void main(String[] args) {
        SpringApplication.run(SpringbootexampleApplication.class, args);
    }

    @Override
    public void run(String... args) {
        System.out.println("Sending Email...");
        try {
            sendEmail();
            sendEmailWithAttachment();
        } catch (Exception e) {
```

```

        e.printStackTrace();
    }
    System.out.println("Done");
}

void sendEmail() {
    SimpleMailMessage msg = new SimpleMailMessage();
    msg.setTo("abbouhassane@gmail.com");
    msg.setSubject("Testing from Spring Boot");
    msg.setText("Hello World \n Spring Boot Email");
    javaMailSender.send(msg);
}

void sendEmailWithAttachment() throws MessagingException, IOException {
    MimeMessage msg = javaMailSender.createMimeMessage();
    // true = multipart message
    MimeMessageHelper helper = new MimeMessageHelper(msg, true);
    helper.setTo("abbouhassane@gmail.com");
    helper.setSubject("Testing from Spring Boot");
    // default = text/plain
    //helper.setText("Check attachment for image!");
    // true = text/html
    helper.setText("<h1>Check attachment for image!</h1>", true);
    helper.addAttachment("image1.jpg", new File("C:/images/image1.jpg"));
    javaMailSender.send(msg);
}
}

```

#### IV- Les tests

Lancer l'application et vérifier que deux mails ont été envoyés à votre adresse destinataire.

<input type="checkbox"/> ☆ 📧 me	Testing from Spring Boot - Check attachment for image!	📧 18:46
<input type="checkbox"/> ☆ 📧 me	Testing from Spring Boot - Hello World Spring Boot Email	18:45

**NB :** Vérifier que votre antivirus est désactivé avant d'effectuer le test.