



PSP - U5 Actividades

[Descargar estos apuntes](#)

- Para cada actividad, crea un proyecto nuevo con el nombre de la clase principal.
- Crea una/varias nuevas clases dentro del paquete `psp.u5.actividades` para cada ejercicio.
- Las clases deben tener el prefijo `psp.u5.actividades.U5AXX_XXXXXXX` y darles un nombre identificativo de la funcionalidad que tienen en el diseño (por ejemplo `psp.actividades.U5A20_FTPClient`).

Es importante que las salidas, tanto de los programas cliente, como del servidor, indiquen claramente de donde viene la información que están mostrando.

1. Actividad U5A01_TestMail

Este proyecto lo tienes que crear como Java with Maven > Java Application

1. Crea una clase de utilidad de correo.

Esta clase va a tener como propiedades todos los objetos necesarios para la composición de un email, y va a tener como métodos las distintas formas de enviar un correo.

Se puede ampliar con todos los métodos que necesites y/o creas convenientes.

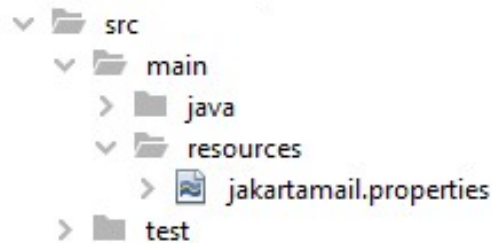
Os pongo un ejemplo de la especificación mínima debería tener la clase, añadiendo los constructores y métodos que necesites para tu implementación.

TestMail
-Session mailSession -Message mailMessage -Properties smtpConfig
+setTo(String[] toAddress) +setCc(String[] toAddress) +setBcc(String[] toAddress) +setSubject(String subject) +setMailText(String bodyText) +setMailHTML(String bodyHtml) +setMailHTMLFile(String htmlFile) +addAttachment(String fileName) +sendSimpleMail() +sendMultipartMail()

Las propiedades con la configuración del servidor SMTP, deberán leerse de un archivo `jakartamail.properties`, con el contenido de las variables que se muestra a continuación.

```
# jakartamail.properties
# Data to send emails from a GMAIL account mail.from=cuenta@iesdoctorbalmis.com
mail.fromtext = "Actividades PSP" mail.username=cuenta@iesdoctorbalmis.com
mail.password=XXXXX
mail.smtp.host=smtp.gmail.com
mail.smtp.port=587
mail.smtp.auth=true
mail.smtp.starttls.enable=true
#mail.debug=true
```

El archivo lo vamos a guardar en la carpeta **src/main/resources** del proyecto.



El acceso al archivo lo haremos de la siguiente forma:

```
InputStream input = this.getClass().getResourceAsStream("/" + propertiesFile)
```

En los apuntes tienes varios recursos para ver cómo trabajar con este tipo de archivos, que no deja de ser una subclase de HashMap.

Si no quieres poner las credenciales de tu correo electrónico en el archivo de properties, haz que se pida la cuenta de correo y la contraseña por consola y pásalas en el constructor del objeto TestMail.

Las propiedades mail.from y mail.fromtext nos permiten modificar lo que se muestra en la cabecera del correo enviado, creando el objeto InetAddress para el método **Message.setFrom** con dos parámetros.

2. Crea una clase con un main que instancie un objeto de tipo TestMail y envíe mensajes de diferentes tipos.

Ofrece un menú, en el main, que permita probar las diferentes opciones de la clase TestMail.

Para las rutas de los archivos, se pueden indicar tanto rutas relativas al proyecto como rutas absolutas del sistema.