

Unidad 5. Servicios en red: Anexo SMTP2GO

[Descargar estos apuntes](#)

Anexo I - Configuraciones SMTP2GO

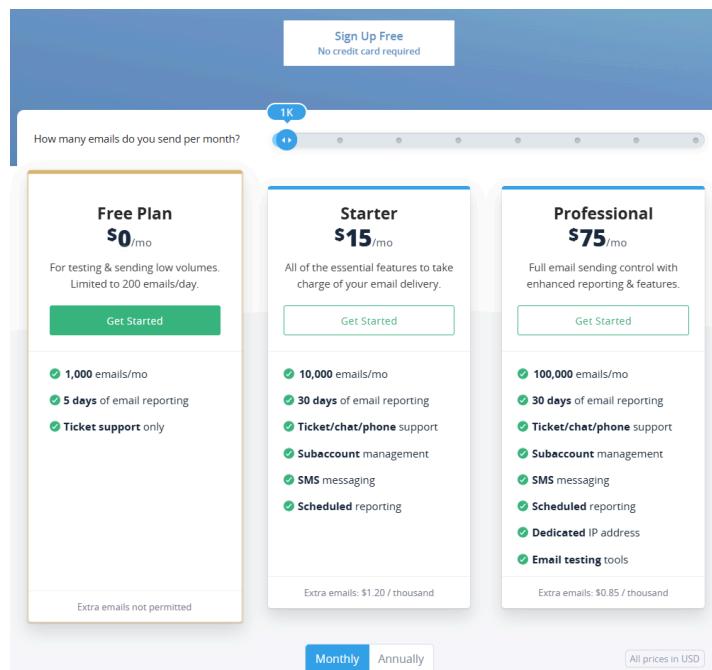
Normalmente, los proveedores de servicios de correo electrónico como Gmail, Yahoo, Outlook, etc., tienen sus propios servidores SMTP para enviar correos electrónicos. Desde hace unos años, por cuestiones de seguridad, estos proveedores han restringido el acceso a sus servidores SMTP desde aplicaciones de terceros requiriendo configuraciones adicionales como la verificación en dos pasos o la activación de aplicaciones menos seguras. En el caso de Gmail, estas configuraciones se explican en el Anexo II de esta unidad.

En caso de trabajar para una empresa, lo normal es que la empresa disponga de su propio servidor SMTP para enviar correos electrónicos corporativos. Esto nos permite tener un mayor control sobre los correos electrónicos enviados y recibidos, así como implementar políticas de seguridad y filtrado de spam personalizadas. Debemos ponernos en contacto con el administrador del sistema o el equipo de TI de la empresa para obtener la configuración correcta del servidor SMTP, incluyendo la dirección del servidor, el puerto, el método de autenticación y las credenciales necesarias.

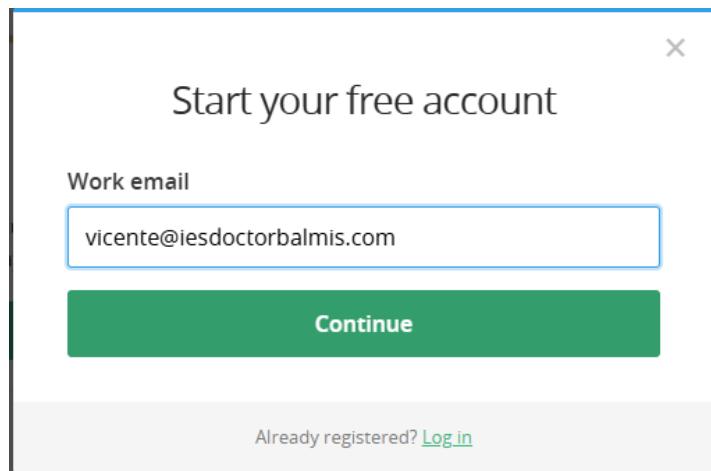
Sin embargo, si queremos usar un servidor SMTP externo para realizar pruebas de envío de correos electrónicos desde nuestras aplicaciones Java podemos utilizar servicios como SMTP2GO.

Registro de nueva cuenta

Lo primero que debemos hacer es registrarnos en [SMTP2GO](#). Podemos optar por una cuenta gratuita que nos permite enviar hasta 1.000 correos electrónicos al mes, lo cual es suficiente para pruebas y desarrollo.



1. Para crear una nueva cuenta, simplemente haga clic en el botón "Get Started Free" y complete el formulario de registro con su dirección de correo electrónico y una contraseña segura.

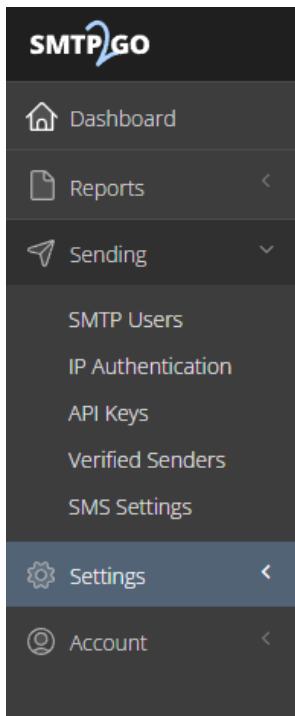


Cuenta corporativa

Si está utilizando SMTP2GO para fines comerciales, es recomendable optar por una cuenta corporativa que ofrece características adicionales como soporte prioritario, mayor capacidad de envío y opciones avanzadas de seguridad.

Os recomiendo que utilicéis la **cuenta @iesdoctorbalmis.es** para registrarse en SMTP2GO. Parte de las configuraciones que se explican en este anexo están hechas para estas cuentas.

2. Despues de registrarse, recibirá un correo electrónico de verificación. Haga clic en el enlace proporcionado en el correo para activar su cuenta SMTP2GO.
3. Una vez activada la cuenta, inicie sesión en el panel de control de SMTP2GO. En el panel de control, navegue a la sección **Sending** y luego a **SMTP Users** para obtener la configuración del servidor SMTP que utilizaremos en nuestro código Java.



4. Desde aquí crearemos un nuevo usuario SMTP haciendo clic en el botón **Add SMTP User**. Proporcione un nombre de usuario y una contraseña segura para este usuario SMTP. Anote estas credenciales, ya que las necesitaremos para configurar nuestro código Java. Así obtenemos una configuración similar a esta:

- **SMTP Server:** mail-eu.smtp2go.com
- **Port:** 2525 (también puede usar 587 o 465 para SSL)
- **Username:** tu_usuario_smtp (el que creaste en el paso anterior)
- **Password:** tu_contraseña_smtp (la que creaste en el paso anterior)

SMTP Users	
SMTP users are permitted to send emails over SMTP with a username and password.	
Username	test@balmis.com
Description	jakarta Mail Testing
Connecting via SMTP	SMTP Server: mail-eu.smtp2go.com SMTP Port: 2525 Alternative ports: 8025, 587 or 25. TLS is available on the same ports. SSL is available on ports: 465, 5465 and 460.
Default Rate Limit Per SMTP User	If a user has no rate limit set, they'll use this default limit. It's currently set to Unlimited. Change Rate Limit Default

Creación de usuario SMTP

No es necesario que el usuario SMTP sea el mismo que el de la cuenta principal. Podemos crear múltiples usuarios SMTP para diferentes aplicaciones o propósitos.

Configuración de puerto SMTP

Debemos pensar que estamos trabajando en diferentes entornos de desarrollo, en función de si estamos en casa o en el instituto.

Por cuestiones de seguridad, la red de aulas de Conselleria no permite el uso indiscriminado de los puertos de las aplicaciones. Muchos de estos puertos están bloqueados por políticas de seguridad de red.

Aunque el puerto que se nos indica por defecto, el 2525 es una buena opción para la mayoría de los casos, ya que es compatible con la mayoría de los proveedores de servicios de Internet, en nuestro caso, solicitamos que si este puerto no funciona, probéis con el puerto 587, que pedimos que esté abierto desde enero hasta junio para poder realizar pruebas de envío de correo..

5. Por último, **por cuestiones de seguridad** debemos restringir el envío de correos a la dirección de mail desde la que vamos a enviar los correos electrónicos, haciendo que esté autorizada en SMTP2GO. Navegue a la sección **Verified Senders** y luego a *Single sender emails* para agregar la/las direcciones de correo electrónico permitidas.

The screenshot shows the 'Verified Senders' interface. At the top, there are two tabs: 'Sender domains' and 'Single sender emails', with 'Single sender emails' being the active one. Below the tabs is a search bar labeled 'Search for a verified sender...'. A tip message at the top says: 'Deliverability performance tip: We highly recommend setting up a [sender domain](#) instead, if you are able to edit the DNS of your domain name.' The main list area shows one entry: 'Email Address' followed by 'vicente@iesdoctorbalmis.com'. At the bottom, it indicates '4 of 5 verified senders remaining on the free plan. [Upgrade for unlimited](#)'.

6. Ahora estamos listos para configurar nuestro código Java para enviar correos electrónicos utilizando SMTP2GO. A continuación, se muestra un ejemplo básico de cómo hacerlo con las librerías de Jakarta Mail.

```

import jakarta.mail.*;
import java.util.Properties;
import jakarta.mail.internet.InternetAddress;
import jakarta.mail.internet.MimeMessage;
public class EmailSender {
    public static void main(String[] args) {
        // Configuración del servidor SMTP
        String host = "mail-eu.smtp2go.com";
        String port = "2525"; // O 587 si 2525 no funciona
        String username = "tu_usuario_smtp"; // Tu usuario SMTP
        String password = "tu_contraseña_smtp"; // Tu contraseña SMTP

        // Propiedades de la conexión
        Properties props = new Properties();
        props.put("mail.smtp.auth", "true");
        props.put("mail.smtp.starttls.enable", "true");
        props.put("mail.smtp.host", host);
        props.put("mail.smtp.port", port);

        // Crear la sesión
        Session session = Session.getInstance(props, new Authenticator() {
            protected PasswordAuthentication getPasswordAuthentication() {
                return new PasswordAuthentication(username, password);
            }
        });

        try {
            // Crear el mensaje
            Message message = new MimeMessage(session);
            message.setFrom(new InternetAddress("tu_correo@example.com"));
            message.setRecipients(Message.RecipientType.TO, InternetAddress.parse("destinatario@example.com"));
            message.setSubject("Prueba de envío de correo con SMTP2GO");
            message.setText("Este es un correo de prueba enviado desde Java utilizando SMTP2GO.");
            // Enviar el mensaje
            Transport.send(message);
            System.out.println("Correo enviado exitosamente!");
        } catch (MessagingException e) {
            e.printStackTrace();
        }
    }
}

```

Consideraciones a tener en cuenta

Recuerda:

- Asegúrate de reemplazar `tu_usuario_smtp` y `tu_contraseña_smtp` con las credenciales que creaste en SMTP2GO.
- Cambia la dirección de correo electrónico en `setFrom` por una de las que hayas configurado como cuentas permitidas para el envío de emails a través de SMTP2GO.
- Si estás utilizando una cuenta gratuita, ten en cuenta las limitaciones de envío de correos electrónicos impuestas por SMTP2GO.

Con esta configuración, deberíamos poder enviar correos electrónicos desde nuestra aplicación Java utilizando el servicio SMTP2GO. Asegúrate de gestionar adecuadamente las excepciones y errores en tu código para una implementación robusta.