

# **Desafío Tecnológico en Ciberseguridad, 2025**

## **Sistema anti-phishing y anti-spamming**



Autores:

**Enrique Gómez Pacheco**  
**Pedro Gabriel Fernández Cañete**

16 de Mayo de 2025

**UNIVERSIDAD DE GRANADA**  
E.T.S de Ingenierías Informática y de Telecomunicaciones

## 1. Descripción General

Esta aplicación es una herramienta desarrollada en Java (Maven), dedicada a la detección automática de correos electrónicos en español de tipo *phishing* y *spam*. Su objetivo principal es mejorar la seguridad en la gestión del correo electrónico mediante un análisis automatizado de los mensajes recibidos, clasificándolos según los criterios previamente mencionados.

## 2. Diseño del Sistema

El sistema ha sido diseñado para leer los correos almacenados en formato .eml en una carpeta local del equipo del cliente. Mediante una interfaz gráfica se le facilita al cliente navegar por su explorador de archivos para seleccionar la carpeta donde tenga almacenados sus correos.

Para ello, está estructurado de forma modular bajo el paradigma de programación orientada a objetos. Las clases de nuestro proyecto son:

- **Email**: clase que modela la estructura de un correo electrónico (asunto, cuerpo, etc.).
- **EmailFilter**: es la clase que contiene el main del proyecto, el cual se ejecuta para iniciar la aplicación. En esta clase también se encuentra el método que guarda la ruta de la carpeta elegida e inicia la descarga de los correos presentes en la carpeta del cliente.
- **PhishingDetector** y **SpamDetector**: clases que contienen las listas negras/palabras clave consideradas como sospechosas de correos de spam o phishing. Estas se utilizan posteriormente, junto a más frases clave descargadas de unas bases de datos públicas, para realizar la clasificación de los correos.
- **CSVEmailLoader**: clase encargada de cargar las bases de datos de términos maliciosos desde archivos CSV.
- **EmailSave**: clase que realiza la conversión de los correos en formato .eml a un objeto Email que guarda los campos más relevantes de los mismos. Posteriormente se encarga de clasificar los correos como legítimos, de spam o de phishing según incluyan o no las frases clave consideradas.
- **Interfaz**: clase que proporciona una interfaz gráfica básica para la interacción con el usuario. Para ello hemos usado la herramienta “JavaFX”.

Las bases de datos públicas utilizadas son las siguientes:

- [https://huggingface.co/datasets/softcapps/spam\\_ham\\_spanish](https://huggingface.co/datasets/softcapps/spam_ham_spanish)
- <https://www.kaggle.com/datasets/luistestalter2025/spam-or-ham-in-spanish>

Por último, para asegurar la portabilidad de la aplicación y su fácil uso, también hemos creado un ejecutable completamente portable de la aplicación Java basada en JavaFX.

Para ello, utilizamos la herramienta “jlink” para construir una “runtime” personalizada que incluye únicamente los módulos necesarios de Java junto con los módulos de JavaFX. Esta “runtime” embebida permite que la aplicación funcione sin necesidad de que el usuario tenga instalado Java o JavaFX.

Posteriormente, se empleó “jpackage” para empaquetar el archivo JAR principal junto con esta “runtime” personalizada en una carpeta ejecutable que contiene el .exe y todos los archivos necesarios para la ejecución.

Realizado por: Enrique Gómez Pacheco y Pedro Gabriel Fernández Cañete

Este método garantiza que la aplicación sea autónoma, portable y compatible con cualquier sistema Windows sin instalaciones previas.

De esta forma facilitamos y simplificamos el uso de la aplicación al usuario final.

### 3. Funcionamiento

El sistema funciona de la siguiente manera:

1. **Carga de Datos:** Se cargan las listas de términos relacionados con *phishing* y *spam* desde archivos CSV.
2. **Petición de ruta:** Se pregunta al cliente la ruta de su equipo donde tiene almacenados sus correos en formato .eml.
3. **Análisis del Correo:** Cada correo es analizado usando estos términos y reglas predefinidas.
4. **Clasificación:** El sistema evalúa si el correo es legítimo, spam o phishing.
5. **Interfaz de Usuario:** Mediante la clase `Interfaz`, el usuario puede visualizar y procesar los correos cargados y separados en tres pestañas: buzón, spam y phishing.
6. **Refresco:** Si el usuario lo desea puede modificar los correos de su carpeta y mediante el botón “Refrescar correos” actualizar la interfaz gráfica para visualizar los nuevos correos.

### 4. Uso de la aplicación

Como hemos comentado, hemos realizado el archivo ejecutable “EmailFilter.exe” para iniciar la aplicación de manera sencilla para el cliente.

Sin embargo, este archivo solo es ejecutable en Windows, por lo que un usuario de Linux, MAC o Android, entre otros sistemas operativos, tendrá que tener instalado Java y ejecutar el proyecto que hemos creado para poder disfrutar de la aplicación.

De esta forma, si el usuario final tiene en su equipo un sistema operativo Windows, simplemente tendrá que hacer doble click en el icono de la aplicación para que se le abra el explorador de archivos donde debe elegir su carpeta local de correos. Tras seleccionarla, ya le aparecerá la interfaz gráfica con los correos clasificados.

En ella podrá navegar por los submenús “Buzón”, “Phishing” y “Spam” y leer los correos en texto plano.

Además, el cliente también dispone de la opción de refrescar la página mediante un botón “Refrescar correos” para ver si tiene correos nuevos.

Para facilitar la prueba de la aplicación, hemos proporcionado una carpeta llamada “emails” con diferentes correos legítimos, de spam y de phishing en formato .eml.