***WT&A***

**Reporte “Package Envios”**

**Envios (Interfaz):**

Esta interfaz define un método llamado "enviar" que toma un objeto Empleado como argumento y devuelve un String.

**CanalComunicacion (Clase base):**

Esta clase proporciona los atributos y métodos comunes a todos los canales de comunicación, es decir, Sms y WhatsApp.

**Propósito:** Representa un canal de comunicación genérico.

* **Atributos:**
* *account\_Sid*: Cadena que almacena el ID de cuenta del canal de comunicación.
* *auth\_Token*: Cadena que almacena el token de autenticación del canal de comunicación.
* *numero\_Telefonico*: Cadena que almacena el número de teléfono del canal de comunicación.
* **Métodos:**
* Constructor: inicializa los atributos de la clase.
* Getters y setters: para acceder y modificar los atributos.

**Sms (Clase hija de CanalComunicacion):**

Esta clase extiende la clase CanalComunicacion y proporciona la implementación para enviar un mensaje de texto (SMS) utilizando la biblioteca Twilio.

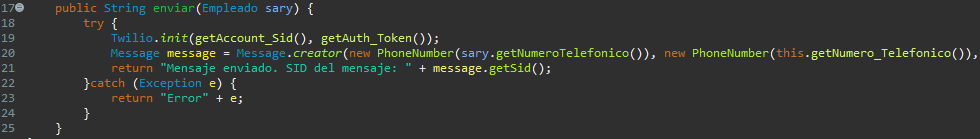
**Propósito:** Representa un canal de comunicación de SMS.

**Herencia:** Sms hereda de la clase CanalComunicacion.

* **Imports:**
* *com.twilio.Twilio*: Proporciona la funcionalidad para interactuar con la API de Twilio.
* *com.twilio.rest.api.v2010.account*.Message: Permite enviar mensajes SMS utilizando Twilio.
* *com.twilio.type.PhoneNumber*: Representa un número de teléfono utilizado por Twilio.
* **Métodos:**

Constructor: inicializa los atributos heredados de la clase base.

enviar: Método que envía un mensaje SMS utilizando la biblioteca Twilio. Recibe un objeto de tipo Empleado y devuelve una cadena indicando el estado del envío.



Se define el método enviar, el cual recibe un objeto de tipo Empleado y devuelve una cadena.

Se inicializa la biblioteca Twilio utilizando el accountSid y el authToken proporcionados.

Se obtiene el número de teléfono del empleado al que se enviará el mensaje SMS.

Se crea el mensaje que se enviará por SMS, el cual incluye el nombre y el sueldo del empleado.

Se utiliza la biblioteca Twilio para crear y enviar el mensaje de SMS, utilizando el número de teléfono del destinatario (toPhoneNumber), el número de teléfono del remitente (fromPhoneNumber) y el mensaje a enviar (mensaje).

Se devuelve una cadena indicando que el mensaje de SMS se ha enviado correctamente.

**WhatsApp (Clase hija de CanalComunicacion):**

Esta clase extiende la clase CanalComunicacion y proporciona la implementación para enviar un mensaje a través de WhatsApp utilizando la biblioteca Twilio.

**Propósito:** Representa un canal de comunicación de WhatsApp.

**Herencia:** WhatsApp hereda de la clase CanalComunicacion.

* **Imports:**
* *com.twilio.Twilio*: Proporciona la funcionalidad para interactuar con la API de Twilio.
* *com.twilio.rest.api.v2010.account.Message*: Permite enviar mensajes utilizando Twilio.
* *com.twilio.type.PhoneNumber*: Representa un número de teléfono utilizado por Twilio.
* **Métodos:**
* Constructor: inicializa los atributos heredados de la clase base.
* enviar: Método que envía un mensaje de WhatsApp utilizando la biblioteca Twilio. Recibe un objeto de tipo Empleado y devuelve una cadena indicando el estado del envío.

****

Se define el método enviar, el cual recibe un objeto de tipo Empleado y devuelve una cadena.

Se inicializa la biblioteca Twilio utilizando el accountSid y el authToken proporcionados.

Se obtiene el número de teléfono del empleado al que se enviará el mensaje de WhatsApp.

Se crea el mensaje que se enviará por WhatsApp, el cual incluye el nombre y el sueldo del empleado.

Se utiliza la biblioteca Twilio para crear y enviar el mensaje de WhatsApp, utilizando el número de teléfono del destinatario (toPhoneNumber), el número de teléfono del remitente (fromPhoneNumber) y el mensaje a enviar (mensaje).

Se devuelve una cadena indicando que el mensaje de WhatsApp se ha enviado correctamente.

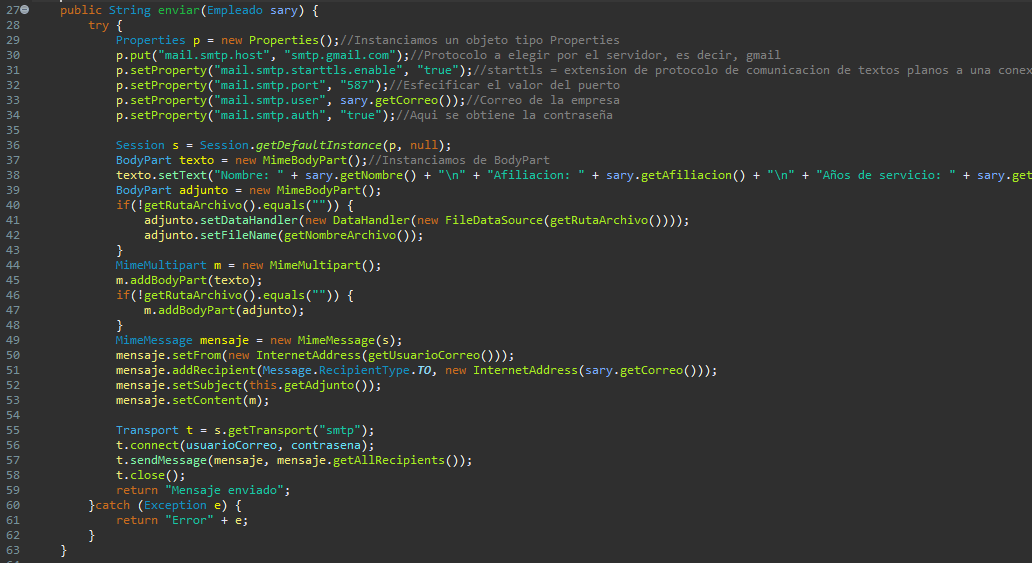
**Correo (Clase que implementa Envios):**

Esta clase implementa la interfaz Envios y proporciona la implementación para enviar un correo electrónico utilizando la biblioteca JavaMail.

**Propósito:** Representa un canal de comunicación de correo electrónico.

**Implementación:** Correo implementa la interfaz Envios.

* **Imports:**
* *java.util.Properties*: Proporciona una colección de propiedades del sistema.
* *javax.mail.\**: Proporciona las clases necesarias para enviar correos electrónicos utilizando JavaMail.
* **Atributos:**
* usuarioCorreo: dirección de correo electrónico del remitente.
* nombreArchivo: nombre del archivo adjunto.
* contrasena: contraseña de la cuenta de correo electrónico.
* rutaArchivo: ruta del archivo adjunto.
* adjunto: asunto del correo electrónico.
* **Métodos:**
* Constructor: inicializa los atributos de la clase.
* Getter y setter: para acceder y modificar los atributos.
* enviar: Método de la interfaz Envios que envía un correo electrónico con la información del empleado utilizando la biblioteca JavaMail. Recibe un objeto de tipo Empleado y devuelve una cadena indicando el estado del envío.

****

Se define el método enviar, el cual recibe un objeto de tipo Empleado y devuelve una cadena.

Se configuran las propiedades del servidor SMTP para el envío de correo electrónico.

Se crea una instancia de Session utilizando las propiedades configuradas y un objeto Authenticator para autenticarse en el servidor SMTP.

Se crea y configura el mensaje de correo electrónico utilizando las clases MimeMessage, MimeBodyPart y MimeMultipart. Se establece el remitente, el destinatario, el asunto y el cuerpo del mensaje, y se adjunta un archivo si se proporciona.

Se envía el mensaje utilizando Transport.send().

Se devuelve una cadena indicando que el correo se ha enviado correctamente, o se maneja una excepción y se devuelve una cadena indicando un error en caso de que se produzca una excepción durante el envío del correo.

**Mensajeria:**

Esta clase contiene el método main y proporciona un menú interactivo para enviar información de un empleado a través de diferentes canales.

Utiliza objetos de las clases Correo, WhatsApp y Sms para enviar los mensajes.