UD5.TRABAJO FINAL DE UNIDAD

SEGURIDAD EN LA PROGRAMACIÓN Y LAS COMUNICACIONES

TABLA DE CONTENIDO

NOTAS Y CONSEJOS

1) Antes de hacer nada, **lea atentamente el contenido**.
2) Es una **prueba individual**, no copies a tus compañeros. Si se observa que se ha copiado se resolverá calificando como no entregado a las partes implicadas.
3) Si se observa que el código es un **copia y pega de ChatGPT** o cualquier otro asistente de IA el trabajo será calificado con un 0/10.
4) **Revisar el programa es tu responsabilidad**, si no funciona correctamente cuando lo pruebas, no funcionará cuando tu profesor lo revise.
5) Nombre del proyecto: *NombreApellido1Apellido2_Tra_RA5_2024*
6) Los proyectos que no compilen o no se ejecuten y no incluyan el fichero .jar no se corregirán.
7) Se penalizará el código que esté mal estructurado y no esté refactorizado.
8) Es **obligatorio** poner comentarios en el código significativos que expliquen de manera clara el código programado.

ENTREGAS E INDICACIONES

Fecha de entrega límite:

6 de marzo de 2024

Qué hay que entregar:

- El Proyecto de **maven** en formato zip.
- Fichero **.jar** de la aplicación de manera que sea funcional.

TAREA

CONCEPTOS TRATADOS

El alumnado deberá integrar en el presente trabajo final de unidad los siguientes conceptos:

- Validación de entradas de usuario.
- Funciones resumen.
- Criptografía simétrica.
- Criptografía asimétrica.
- Firma electrónica.
- Creación de registros.

FUNCIONALIDAD

Se deberá desarrollar una aplicación que permita:

- Calcular la **función resumen** (hash) de un **fichero**.
- Calcular la función resumen de una cadena de **texto**.
- Creación de una **clave secreta** utilizando el algoritmo AES.
    - Se generará la clave secreta AES a partir de una contraseña introducida por el usuario convertida en un array de bytes. Para hacerlo, mirar la clase **SecretKeySpec**.
    - La contraseña introducida por el usuario será analizada por el programa para comprobar que es segura, y en caso contrario informarle de que no es segura y qué condiciones reúne una contraseña segura.
- **Cifrado y descifrado de ficheros** utilizando el algoritmo de criptografía simétrica **AES**.
- Creación de **pares de claves** utilizando el algoritmo **RSA**.
- **Gestión** de pares de claves RSA.
    - Salvar una clave pública o privada en un fichero.
    - Cargar una clave pública o privada de un fichero.
- **Cifrado y descifrado** de **ficheros** utilizando el algoritmo de criptografía asimétrica **RSA**.
- **Firma de ficheros y verificación**.
- Generar un **log** con los errores (excepciones) del programa y otro con la actividad del usuario en el tiempo.

INTERFAZ DE USUARIO

Interfaz de usuario de la aplicación:

- Cada estudiante deberá realizar un tipo diferente de interfaz, aquella que le haya indicado el profesor.
- Los tipos de interfaz serán:
    - Interfaz gráfica de usuario utilizando JavaFX.
    - Interfaz ASCII de usuario (terminal interactiva).
    - Ejecución no interactiva mediante comando y opciones.

DOCUMENTACIÓN

La aplicación se deberá documentar en el código:

- El estudiante deberá explicar detalladamente en el código el programa.
- Se comentará en la línea anterior a cada **método** qué función tiene éste dentro del programa de manera detallada.
- Al comienzo de cada **clase** ese explicará cuál es su funcionalidad.
- Se indicará para cada **atributo de clase** cuál es su función.

Además, deberá **incorporar ayuda de usuario**.

ESTRUCTURA DEL PROGRAMA

La aplicación deberá ser desarrollada siguiendo el concepto de una clase una función, por lo que para las funcionalidades pedidas **se deberán implementar** al menos las siguientes **clases**:

- HashTool.
- PasswordValidator.
- SecretKeyManagerAES.
- AESEncription.
- KeyPairManagerRSA.
- RSAEncryption.
- DigitalSigningTool.

Además, en el caso de la **aplicación con interfaz de usuario por terminal,** tanto interactiva como no, se deberán crear las siguientes clases:

- UserInterface.
    - Implementación de los mensajes que se muestran al usuario en función de sus entradas.
- Banner.
    - Se implementan dos métodos: uno para el mensaje de bienvenida y otro para el de despedida.

Además, la **clase principal** se llamará **App**, y se utilizará únicamente para ejecutar la interfaz de usuario e imprimir el banner.

EVALUACIÓN

Con esta práctica se evalúan los CE A, B, E, F y H del RA5.

- La aplicación implementa correctamente la función hash de un texto. (0,5 puntos)
- La aplicación implementa correctamente la función hash de un fichero. (0,5 puntos)
- Se ha analizado si la contraseña introducida por el usuario para generar la clave AES es segura. (0,5 puntos)
- Se genera la clave secreta AES a partir de la contraseña. (0,5 puntos)
- Se cifra y descifra correctamente el fichero con la clave secreta AES. (1,5 puntos)
- Se genera el par de claves RSA. (0,5 puntos)
- Se salvan y cargan las claves RSA en ficheros. (1 punto)
- Se utilizan el par de claves RSA para cifrar y descifrar un fichero. (1,5 puntos)
- Se utiliza el par de claves para firmar electrónicamente. (1 punto)
- Se genera el fichero .jar y funciona correctamente. (0,5 puntos)
- La aplicación es sólida, no presenta errores y controla correctamente las excepciones. (1 punto)
- La aplicación genera los registros de errores y acciones del usuario. (0,5 puntos)
- La clases están correctamente documentadas, con comentarios explicativos para la clase, atributos y cada uno de los métodos. (0,5 punto)