PASSWD.
GESTOR DE CONTRASEÑAS SEGURO.

ALEJANDRO ALBERTO JIMÉNEZ BRUNDIN

TRABAJO DE FINAL DE GRADO

TUTOR/A:---

FEMPA 2025

“La llave del éxito es habituarse en tu vida a hacer las cosas qué tienes miedo a hacer”

Vincent van gogh

Agradecimientos

Quiero dar las gracias a todos los compañeros de clase con los que he tenido relación, en especial a mi "Pareja de baile" por aguantarme durante los trabajos en parejas.

A mis profesores por su apoyo y guía a lo largo de estos momentos finales y por lo que he aprendido de ellos.

A mis amigos por el apoyo que han sido siempre y sobre todo, en los momentos más difíciles de mi vida.

Y por último y más importante a mis padres, ya que sin ellos esto no habría sido posible, en especial a mi madre, que en paz descanse, por lo que ha significado para mi durante todos estos años.

A todos vosotros muchas gracias.

Índice General

Resumen

Hoy en día con el incremento de la tecnología el número de cuentas que posee una persona media ha aumentado significativamente y es difícil utilizar métodos tradicionales para recordar tantas contraseñas sin recurrir al uso de estas en varios lugares, creando así situaciones que comprometen la seguridad de dichas cuentas.

El objetivo principal de este TFG es el desarrollo de una aplicación multiplataforma que permita gestionar contraseñas, es decir, que permita almacenar contraseñas, generar contraseñas seguras y hacer un seguimiento con el fin de poder realizar un cambio periódico de estas evitando la repetición de contraseñas usadas recientemente. Como objetivo secundario, se hará hincapié en la creación de una interfaz amigable que incentive su uso por parte de usuario no tan avanzados.

Palabras clave: contraseñas, java, AES, SHA

Abstract

Nowadays with the increase of technology the number of accounts that the average person has has increased significantly and it is difficult to use traditional methods to remember so many passwords without resorting to the use of these in several places, thus creating situations that compromise the security of these accounts.

The main objective of this TFG is the development of a multiplatform application to manage passwords, i.e., to store passwords, generate secure passwords and track them in order to be able to periodically change them, avoiding the repetition of recently used passwords. As a secondary objective, emphasis will be placed on the creation of a user-friendly interface that encourages its use by less advanced users.

Keywords: password, java, AES, SHA

Parte I

Memoria del proyecto

Capitulo 1

Introducción

1.1 Introducción

En la actualidad, es común que las personas utilicen una única contraseña o, en el mejor de los casos, un conjunto muy reducido de ellas. Una variante de esta práctica es la creación de contraseñas a partir de una frase base combinada con un elemento relacionado con el servicio en cuestión. Aunque este método puede generar contraseñas ligeramente más seguras, sigue presentando un riesgo significativo, ya que, si una de ellas se ve comprometida, las demás también podrían quedar expuestas. Además, no es una práctica ampliamente adoptada.

Por otro lado, almacenar las contraseñas en papel no está exento de inconvenientes, ya que es fácil perder o extraviar estos soportes físicos. Incluso, existen casos en los que fotografías compartidas en la red muestran notas adhesivas con contraseñas visibles, aumentando el riesgo de filtraciones.

Para mitigar los problemas de seguridad asociados con estos métodos tradicionales, se plantea el uso de tecnología como una solución más efectiva y fiable.

1.2 Motivación

Este proyecto surge como respuesta a la creciente necesidad de gestionar de manera eficiente el gran número de servicios que una persona utiliza en la actualidad, cada uno de los cuales requiere un usuario y una contraseña. Para ello, resulta conveniente centralizar esta información en un solo lugar, proporcionando una herramienta que permita almacenar de forma segura una cantidad prácticamente ilimitada de contraseñas.

Gracias a esta solución, se elimina la necesidad de recordarlas manualmente, lo que facilita el uso de credenciales únicas para cada servicio. Esto no solo mejora significativamente la seguridad, sino que también ofrece una alternativa más fiable y duradera en comparación con métodos tradicionales como el almacenamiento en papel.

1.3 Objetivos

El objetivo principal de este trabajo es la creación de una aplicación multiplataforma que permita centralizar la gestión de contraseñas reduciendo las contraseñas que el usuario final necesita a una sola (que es con la que iniciaría sesión en nuestra aplicación, con un especial hincapié en la seguridad sin dejar completamente de lado la usabilidad.

Para lograr ese objetivo principal es necesario cumplir los siguientes objetivos:

1. Identificación del usuario mediante una cuenta de correo o Google.
2. Almacenamiento de contraseñas en local bajo el algoritmo de codificación AES* y la clave maestra.
3. Permitir crear contraseñas seguras mediante un algoritmo personalizable.
4. Recordar al usuario de manera periódica el cambio de las contraseñas almacenadas.
5. Facilitar la inclusión de perfiles de usuario base para guardar contraseñas de aplicaciones no soportadas por la aplicación.

Además, como objetivos secundarios para la mejora de la usabilidad se contemplarán los siguientes objetivos secundarios:

1. Diseño e implementación de una interfaz gráfica que permita un uso fluido de la aplicación.
2. Un sistema de exportación de las contraseñas almacenadas, para su posterior importación desde un navegador.
3. Un sistema de exportación de las contraseñas almacenadas, para su posterior importación desde otro equipo que tenga instalada la aplicación.
4. Un sistema de importación de las contraseñas almacenadas en un navegador, para su incorporación a los perfiles de contraseñas almacenados en la aplicación.
5. Redacción de unos sencillos manuales de usuario y de instalación.
6. Creación de un instalador
7. Cambio automático de las contraseñas

1.4 Entorno tecnológico

1.4.1 Aplicaciones similares

Este proyecto cubre necesidades de primera necesidad en nuestro actual modelo de vida rodeados de tecnología y servicios online por lo que existen multitud de aplicaciones que cubren estas mismas necesidades.

Podemos encontrar principalmente dos corrientes para solventar estas necesidades, una es la eliminación de las contraseñas mediante diversos procesos de verificación, habitualmente vinculados al teléfono móvil, y la otra, la cual es la aproximación que tomaremos con la aplicación que vamos a desarrollar, es la centralización de todas las contraseñas en una aplicación o servicio online, dejando en la medida de lo posible que sea la aplicación de manera automática la encargada de introducir dichas contraseñas cuando sea preciso.

Ya que para nosotros no es relevante la primera corriente, a continuación, estudiaremos cuatro aplicaciones que utilizan la segunda aproximación:

► **Bitwarden**

▷ **Algoritmo:** Utiliza el algoritmo de cifrado AES de 256 bits para proteger la información.

▷ **Usuario:** La autenticación se realiza mediante un usuario vinculado a un correo electrónico, utilizando una contraseña maestra como credencial principal. Además, admite autenticación en dos pasos (2FA) para mayor seguridad.

▷ **Almacenamiento:** Las contraseñas y credenciales se almacenan en la nube con la opción de autoalojamiento para mayor control sobre los datos.

▷ **Generador de contraseñas:** Ofrece un generador de contraseñas altamente personalizable, permitiendo ajustar longitud, tipos de caracteres y exclusión de caracteres ambiguos.

▷ **Instalación:** Disponible como una extensión para navegadores, una aplicación de escritorio, una aplicación móvil y una interfaz web, brindando accesibilidad multiplataforma.

▷ **Expiración de contraseñas:** No cuenta con una función automática de expiración de contraseñas, aunque permite la revisión de contraseñas antiguas y su rotación manual.

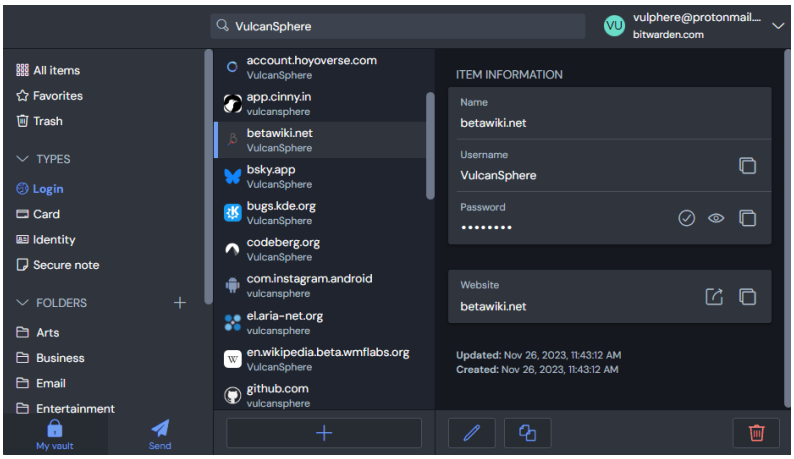

Figura 1: Gestor de contraseñas en Bitwarden.

► **Lastpass**

▷ **Algoritmo**: Utiliza como algoritmo de codificación AES de 256 bits

▷ **Usuario**: Verifica la identidad mediante un usuario vinculado a un correo electrónico utilizando la contraseña maestra como comprobante.

▷ **Almacenamiento**: El almacenamiento de las contraseñas y usuarios se realiza en la nube.

▷ **Generador de contraseñas**: Proporciona un generador de contraseñas personalizable, destacando la función de generar contraseñas "pronunciables"

▷ **Instalación**: Para su funcionamiento es necesaria la instalación de una extensión de navegador para su uso en PCs y de una aplicación en el caso de móviles.

▷ **Expiración de contraseñas**: No se aprecia una funcionalidad que haga referencia a este dato.

Figura 2: Generador de contraseñas en Lastpass.

► **Keepass**:

▷ **Algoritmo**: Utiliza como algoritmo de codificación AES de 256 bits o Chacha 20* de 256 bits, a elección del usuario.

▷ **Usuario**: Tiene una cuenta monousuario de carácter local sin ningún distintivo, utiliza la contraseña maestra cómo verificación.

▷ **Almacenamiento**: El almacenamiento de las contraseñas y usuarios se realiza de manera local en un archivo.

▷ **Generador de contraseñas:** Proporciona un generador de contraseñas personalizable, con una entrada que permite introducir una selección de símbolos elegibles para formar parte de la contraseña y con la opción de utilizar algoritmos proporcionados por los usuarios.

▷ **Instalación**: La aplicación posee un instalador convencional que realiza una instalación en el PC, y además dispone de una versión portable.

▷ **Expiración de contraseñas**: Permite establecer una fecha de expiración y lo indica en el resumen global, no notifica al usuario de manera directa.

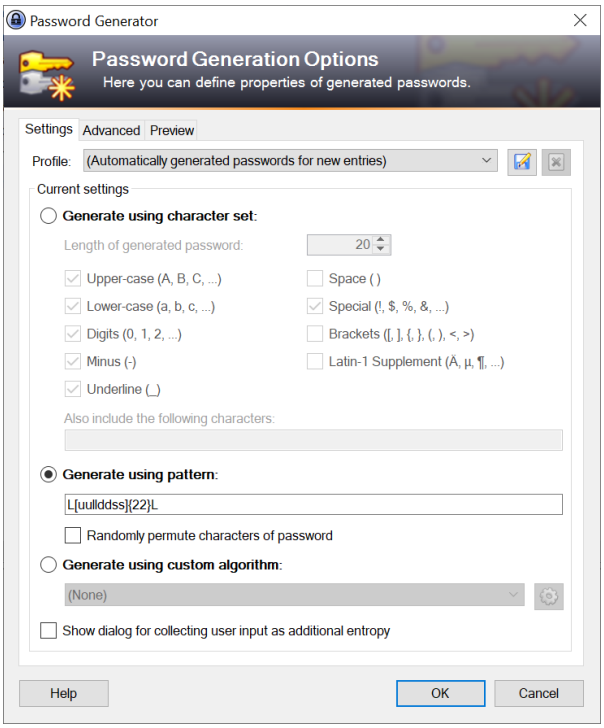

Figura 3: Generador de contraseñas en Keepass.

1.4.2 Lenguaje

Para el desarrollo de la aplicación se ha decidido utilizar el lenguaje de programación Flutter*. Esta decisión parte de una premisa importante, que es la gran portabilidad de las aplicaciones programadas en este lenguaje, la cual permite que la aplicación desarrollada pueda ser utilizada en una amplia cantidad de equipos.

1.4.3 Algoritmos de codificación

Para el desarrollo de la aplicación son importantes tres algoritmos que mencionaremos a continuación.

El primero de ellos es AES (Advanced Encryption Standard), concretamente en su versión de 256 bits. Este algoritmo nos permite codificar bloques de hasta 256 bits y será el algoritmo encargado de codificar las contraseñas que guardara nuestra aplicación, no se ha encontrado una manera eficaz de romper este algoritmo y un ataque de fuerza bruta* es algo computacionalmente hablando inviable en un tiempo razonable.

1.5 Organización del documento

En este anexo se explica la estructura de este documento. A continuación, podemos ver las diferentes partes y capítulos que componen este documento:

► Parte I: Memoria del proyecto

En esta parte se encuentran los apartados relacionados con el acometimiento del proyecto desde su concepción hasta el seguimiento de este.

▷ Capítulo 1: Introducción

Este capítulo sirve de presentación del proyecto mostrando la motivación y objetivos de este proyecto, así como el entorno tecnológico que lo engloba. Es el capítulo en el que nos encontramos.

▷ Capítulo 2: Planificación, estimaciones y presupuesto

En este capítulo se muestra la preparación previa al proyecto con estimaciones, la planificación y el presupuesto. También se explica la metodología utiliza para el desarrollo de la aplicación.

▷ Capítulo 3: Seguimiento

En este capítulo se enfrenta la planificación con la realidad, así como explicar las medidas correctivas que se han tomado debido a la diferencia entre ambas.

► **Parte II: Documentación técnica**

En esta parte se detallan el análisis (identificación y especificación de requisitos), diseño (arquitecturas lógica y física y diagrama de clases), implementación (detalles relevantes de la implementación realizada) y pruebas (pruebas de caja blanca y de caja negra) de cada una de las respectivas iteraciones realizadas.

▷ Capítulo 4: Primera iteración

Este capítulo está dedicado al análisis, el diseño, detalles de implementación y pruebas de la primera iteración.

▷ Capítulo 5: Segunda iteración

Este capítulo está dedicado al análisis, el diseño, detalles de implementación y pruebas de la segunda iteración.

▷ Capítulo 6: Tercera iteración

Este capítulo está dedicado al análisis, el diseño, detalles de implementación y pruebas de la tercera iteración.

► **Parte III: Manuales**

Esta parte contiene el capítulo 7 dedicado a los manuales.

▷ Capítulo 7: Manuales

Este capítulo está dedicado al manual de instalación que explica cómo poner a funcionar la aplicación, y el manual de usuario que explica la funcionalidad que tiene la aplicación y cómo hacer uso de ella.

► **Parte IV: Conclusiones**

Esta parte está el último capítulo, el capítulo 8 conclusiones.

▷ Capítulo 8: Conclusiones

En esta parte se exponen las conclusiones finales tras la realización de este proyecto.

► **Parte V: Webgrafía**

En esta parte se muestran las fuentes externas utilizadas para la realización de este proyecto, indicando el motivo y fecha de su visita.

Capítulo 2

Planificación, estimaciones y presupuesto

2.1 Metodología

Para la realización de este TFG se seguirá una metodología incremental donde a partir de pequeños prototipos se vaya añadiendo funcionalidad y la mejora de la interfaz.

Se planearán con antelación las funcionales a desarrollar en cada iteración, así como el número de estas. Al acabar una iteración se procederá a realizar una reunión de validación con el tutor, así como una comparación con la línea base del proyecto tras la cual se tomarán medidas correctivas en la planificación de ser necesario.