



Instituto Tecnológico de Costa Rica
Escuela de Ingeniería en Computación

Lenguajes de Programación

Proyecto 2: Programación Funcional

Alumnos:

Andrés Bonilla Solano | 2023101220
Emily Sánchez Orozco | 2021067314

I Semestre 2025

Índice

Enlace de GitHub.....	3
Pasos de instalación del programa.....	3
Requisitos previos.....	3
Manual de usuario.....	4
Arquitectura lógica utilizada.....	10
Flujo de uso.....	11
Librerías adicionales.....	12

Enlace de GitHub

https://github.com/AndyBS1/Proyecto2_Lenguajes

Pasos de instalación del programa

Requisitos previos

1. Tener instalado Cabal (Gestor de paquetes de Haskell, también se puede usar stack, pero para este proyecto se utilizó cabal)
2. Tener instalado GHC (compilador Haskell, versión 9.8.2 o superior)

Pasos de la instalación:

1. Clonar el repositorio:

```
git clone https://github.com/AndyBS1/Proyecto2_Lenguajes.git
```

```
cd proyecto-password-manager
```

2. Preferiblemente limpiar las construcciones de cabal anteriores para evitar conflictos:

En la terminal: cabal clean

3. Compilar el proyecto:

En la terminal: cabal build

4. Ejecutar el proyecto:

En la terminal: cabal run

Manual de usuario

Para ejecutar el programa lo primero que hay que hacer es introducir en la terminal el siguiente comando: `cabal run`

Inicio de sesión y Registro:

Luego de ejecutar el comando anterior, el usuario tendrá tres opciones dentro de un menú principal.

```
=== MENU PRINCIPAL ===  
1. Iniciar sesion  
2. Registrar nuevo usuario  
3. Salir  
Seleccione una opción: █
```

En caso de ser la primera vez que se utiliza el programa se puede seleccionar la opción de “Registrar nuevo usuario” (2). El sistema le solicitará un nombre de usuario único y un PIN para poder tener un acceso seguro.

```
Bienvenido al Administrador de Contraseñas.  
Por favor, introduce un nombre de usuario:  
Maria  
Por favor, define un PIN para proteger tu acceso:  
  
Confirma tu PIN:  
  
Usuario 'Maria' registrado correctamente.
```

Después de registrarse, se puede seleccionar la opción de “Iniciar sesión” (1).

```
Por favor, ingresa tu nombre de usuario:  
Maria  
Por favor, ingresa tu PIN para acceder:  
█
```

Si el usuario y PIN ingresado son correctos, le aparecerá lo siguiente:

```
Inicio de sesión exitoso!  
Bienvenido, Maria  
Cuenta creada para el usuario: Maria
```

El mensaje de “Cuenta creada para el usuario: ****” aparecerá solamente si es la primera vez que se utiliza el sistema de gestión de contraseñas.

Gestión de Contraseñas:

Luego del inicio de sesión exitoso, el usuario tendrá un nuevo menú para gestionar sus credenciales.

```
-----  
=== Gestion de Contraseñas ===  
1. Consultar todas  
2. Consultar por servicio  
3. Consultar por cuenta  
4. Agregar  
5. Modificar  
6. Eliminar  
7. Salir  
Seleccione una opción (1-7): 
```

Lo primero que puede hacer el usuario al ingresar es agregar nuevas credenciales (opción 4):

Para cada una de las credenciales se le solicitará el nombre o título del servicio, el nombre de usuario dentro del servicio y la contraseña. Además, cuando digite la contraseña, no aparecerá en pantalla por motivos de seguridad por lo que al terminar de ingresar la contraseña solamente debe presionar la tecla de “enter”.

```
-----  
Ingrese un servicio:  
Netflix  
Ingrese el usuario:  
maria3012  
Ingrese su contraseña:  
  
Credencial agregada exitosamente.  
-----
```

Luego, el usuario puede hacer uso de todas las funciones del menú por lo tanto a continuación se explicará cada una de las funciones que ofrece:

1. Consultar todas: Muestra la información de todas las credenciales asociadas al usuario en formato de tabla.

```
-----  
Credenciales de: Maria  
-----
```

Título	Usuario	Contraseña
Instagram	mari*****	*****
Spotify	Mari*****	*****
Netflix	mari*****	*****

```
-----
```

2. Consultar por servicio: en esta opción el usuario debe ingresar el nombre del servicio asociado a la credencial. Además, tendrá la opción de copiar el usuario o la contraseña al portapapeles.

```
-----  
Ingrese un servicio:
```

```
Netflix  
-----
```

```
Servicio: Netflix
```

```
Usuario: maria3012
```

```
Contraseña: password_ejemplo1  
-----
```

```
-----  
=== Menú de opciones para copiar al portapapeles ===
```

```
1. Usuario
```

```
2. Contraseña
```

```
3. Salir  
█
```

3. Consultar por cuenta: esta opción es una alternativa en caso de que el usuario prefiera realizar la consulta digitando el nombre de la cuenta.

```
-----  
Ingrese el nombre de la cuenta:
```

```
mariaaaaSH_1  
-----
```

```
Servicio: Instagram
```

```
Usuario: mariaaaaSH_1
```

```
Contraseña: password_ejemplo3  
-----
```

4. Agregar: Esta fue la función que se explicó al principio de la sección. Se le solicitará el nombre o título del servicio, el nombre de usuario dentro del servicio y la contraseña.

```
-----  
Ingrese un servicio:  
Instagram  
Ingrese el usuario:  
mariaaaaSH_1  
Ingrese su contraseña:  
  
Credencial agregada exitosamente.  
-----
```

5. Modificar: Al seleccionar esta opción, se le solicitará al usuario ingresar un servicio y luego se le solicitará el dato que desea editar (usuario/contraseña).

```
-----  
Ingrese un servicio:  
Netflix  
Ingrese el dato a modificar (usuario/contrasena):  
usuario  
Ingrese el nuevo nombre de usuario:  
marisolano1  
Credencial modificada exitosamente.  
-----
```

Además, luego se puede consultar para verificar el cambio.


```
-----  
Ingrese un servicio:  
Netflix  
-----  
Servicio: Netflix  
Usuario: marisolano1  
Contraseña: password_ejemplo1  
-----
```

6: Eliminar: esta opción es para eliminar una credencial asociada a un servicio.

```
-----  
Ingrese un servicio:  
Instagram  
Credencial eliminada exitosamente.  
-----
```

Asimismo, puede consultar todas las credenciales para verificar que se eliminó o también puede consultar por servicio.

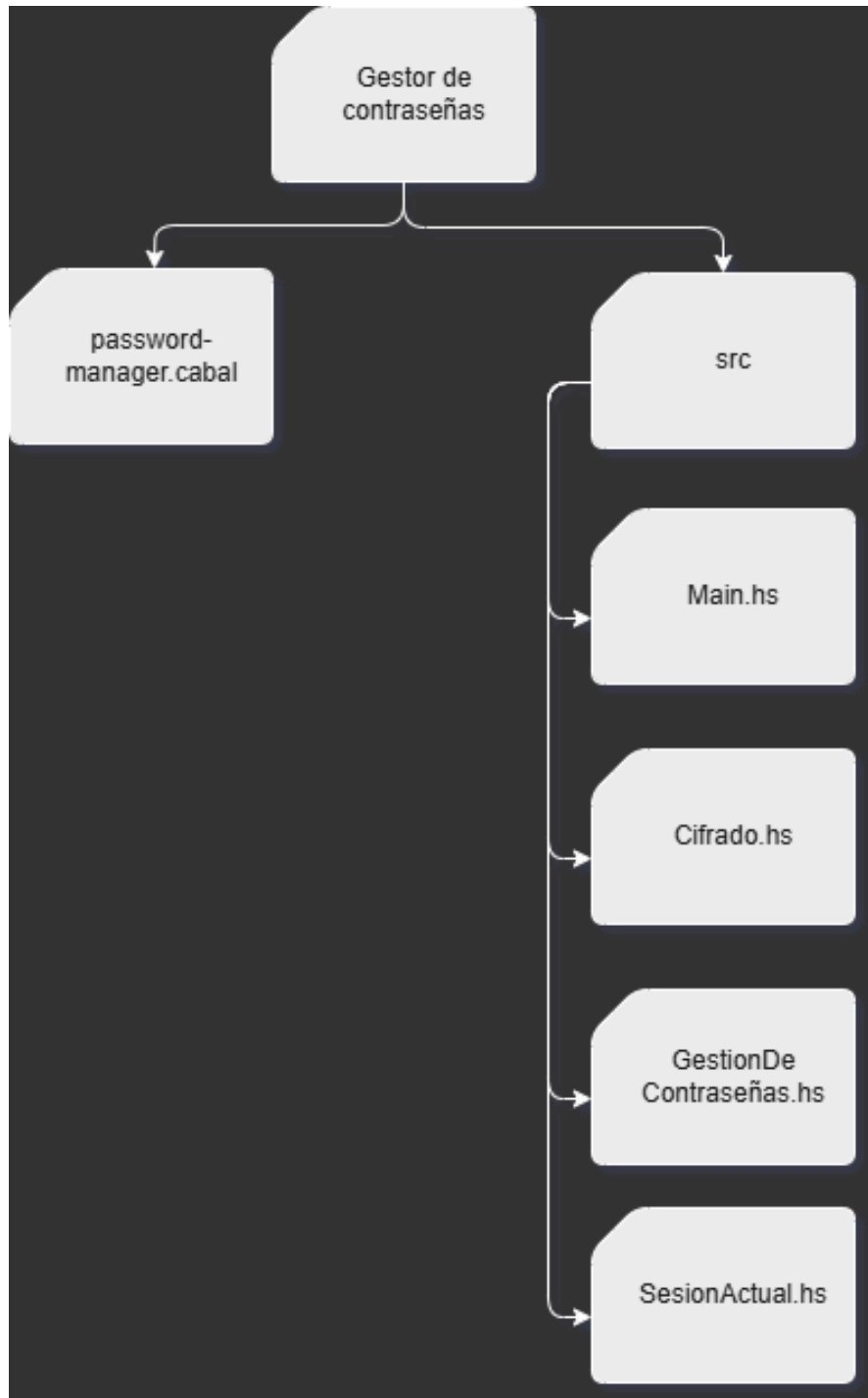
```
-----  
Credenciales de: Maria  
-----  
Título          | Usuario          | Contraseña  
-----  
Spotify         | Mari*****      | *****  
Netflix         | mari*****      | *****  
-----
```

7: Salir:

```
Seleccione una opción (1-7): 7  
Saliendo del programa...
```

Arquitectura lógica utilizada

Estructura del proyecto:



El sistema está organizado en módulos para facilitar su mantenimiento y escalabilidad:

Módulo	Descripción
Main.hs	Punto de entrada, muestra el menú y controla el flujo de la aplicación.
GestionDeContrasenas.hs	Permite agregar, consultar, modificar y eliminar contraseñas.
SesionActual.hs	Maneja el estado de la sesión del usuario.
Cifrado.hs	Funciones de cifrado y descifrado simétrico.

Importante el archivo password-manager.cabal define la configuración del proyecto, incluyendo nombre, versión, dependencias, el archivo principal, módulos fuente y otras propiedades que Cabal necesita, si alguna de esos componentes falta en este archivo el sistema no va a compilar.

Flujo de uso

1. Inicio

- Se muestra el menú principal.

2. Registro

- El usuario define su nombre y PIN.
- Se guarda el hash del PIN junto al nombre en un archivo json.

3. Inicio de sesión

- Se solicita el usuario y su PIN.
- Si es válido, se inicia sesión.

4. Gestión de contraseñas

- Se permite al usuario manejar sus credenciales por servicio.
- Toda la información se guarda en un archivo json.

Librerías adicionales

Aeson: es una biblioteca de análisis y codificación de JSON que ofrece un alto rendimiento. Para el proyecto se utilizó para almacenar los usuarios y también las credenciales.

Cryptonite: Es un repositorio que incluye muchas maneras de implementar criptografía. En el caso del presente proyecto, se utilizó AES que es una forma simétrica de cifrado.

Bytestring: Sirve para la manipulación eficiente de cadenas binarias.

Memory: Soporte de manejo de datos sensibles y estructuras cifrables.

Process: Ejecutar comandos externos desde Haskell.

Directory: Verificación y manejo de archivos/directorios (doesFileExist).

Echo: Captura segura de entrada oculta en consola (input de PIN).

Aeson: Codificación y decodificación de JSON.

Text: Manejo eficiente de cadenas Unicode como text.

Random: Generación de números aleatorios.

Hashable: Generación de hashes simples para valores como el PIN.