

## Instituto Tecnológico de Costa Rica

IC-4700 Lenguajes de Programación  
I-2025

## Proyecto 2: Administrador de Contraseñas en Haskell

### Paradigma Funcional

Realizado por:

- Daniel Zeas Brown

---

## Índice

1. Enlace de GitHub
2. Descripción del Proyecto
3. Requisitos
4. Instalación]
5. Manual de Usuario
6. Arquitectura Lógica
7. Funcionamiento

---

## Enlace de GitHub

[Repositorio del Proyecto](#)

---

# Descripción del Proyecto

Sistema de administración de contraseñas seguro escrito en Haskell, que permite:

- **Registro y acceso** mediante PIN.
- **Gestión cifrada** de credenciales (agregar, modificar, eliminar).
- **Persistencia** en archivos locales con encriptación AES-256.
- **Copia al portapapeles** de usuarios/contraseñas.

## Características clave:

- Cifrado de contraseñas con PIN maestro.
- Interfaz de línea de comandos intuitiva.
- Compatibilidad probada con Linux y macOS.

---

## Requisitos

Componente	Detalle
Sistema	Linux/macOS
Compilador	GHC (≥ 8.10)
Librerías	cryptonite , bytestring , process

Instalación de dependencias:

```
cabal install cryptonite bytestring
```

---

## Instalación

### 1. Clonar el repositorio:

```
git clone https://github.com/tu-usuario/proyecto2-haskell.git
cd proyecto2-haskell
```

### 2. Compilar:

```
ghc -o password-manager Main.hs -threaded
```

### 3. Ejecutar:

```
./password-manager
```

---

## Manual de Usuario

### Comandos Disponibles

Comando	Acción
1	Ver todas las contraseñas
2	Agregar nueva contraseña
3	Buscar contraseña por título
4	Modificar contraseña existente
5	Eliminar contraseña
6	Copiar al portapapeles
7	Cambiar PIN
8	Cambiar de usuario
9	Salir y guardar

### Ejemplo de Uso

#### 1. Registro:

```
> Seleccione una opción (1-3): 2
> Ingrese su nombre de usuario: alice
> Ingrese un PIN seguro (mínimo 4 dígitos): ****
✓ ¡Usuario creado con éxito!
```

#### 2. Agregar contraseña:

```
> Título/Descripción: Gmail
> Usuario/email: alice@gmail.com
```

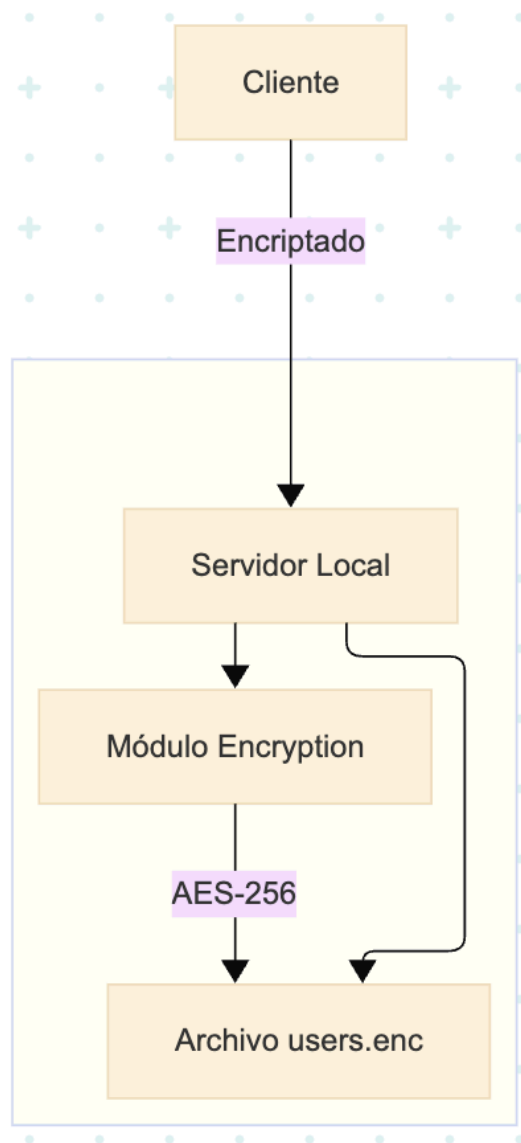
```
> Contraseña: *****  
✓ ¡Contraseña agregada!
```

### 3. Copiar al portapapeles:

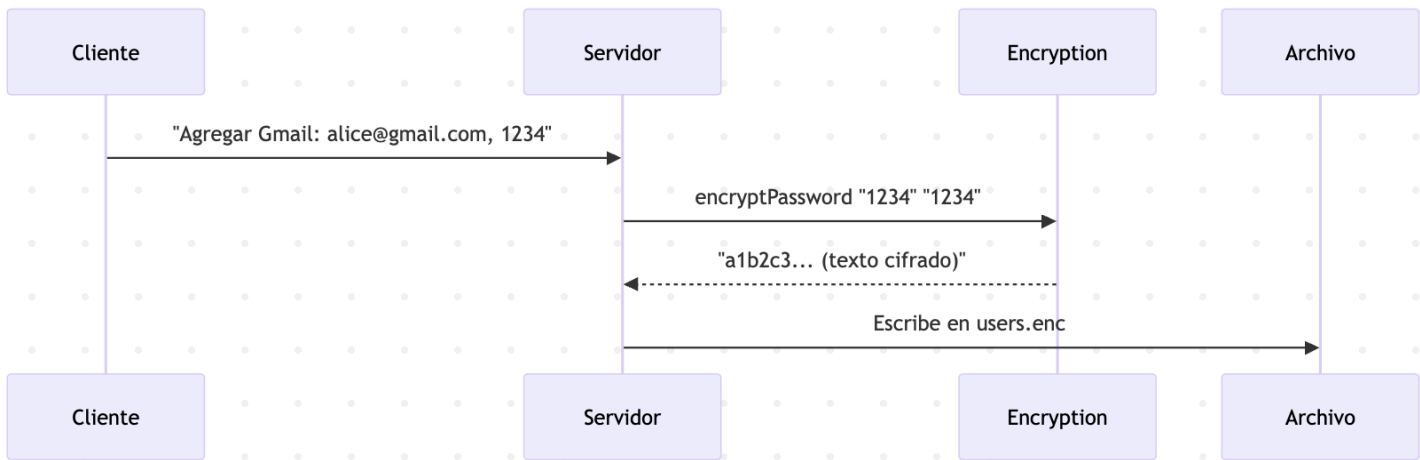
```
> Ingrese el número (1-3): 1  
> Seleccione: 1. Copiar usuario / 2. Copiar contraseña: 2  
✓ ¡Contraseña copiada al portapapeles!
```

---

## Arquitectura Lógica



## Ejemplo: Flujo de Datos



## Módulos Principales

Módulo	Función
Main.hs	Punto de entrada y flujo principal
UI.hs	Interfaz de usuario y menús
Storage.hs	Persistencia en archivos
Encryption.hs	Cifrado AES-256 y gestión de claves
Clipboard.hs	Integración con portapapeles del SO

## Funcionamiento

### 1. Registro y Acceso

- **PIN maestro:** Encriptado con AES-256 antes de almacenarse.
- **Validación:** 3 intentos fallidos bloquean el acceso temporalmente.

### 2. Gestión de Contraseñas

- **Cifrado:** Cada contraseña se encripta con el PIN del usuario.
- **Ocultamiento:** Contraseñas parcialmente ocultas ( us\*\*@gmail.com ).

### 3. Persistencia

- **Archivo `users.enc`** : Almacena datos en formato encriptado.
- **Portabilidad**: El archivo puede transferirse entre dispositivos.

## 4. Seguridad

- **Encriptación**: AES-256 en modo ECB (según clave derivada del PIN).
- **Protección**: No se almacenan datos en texto plano.