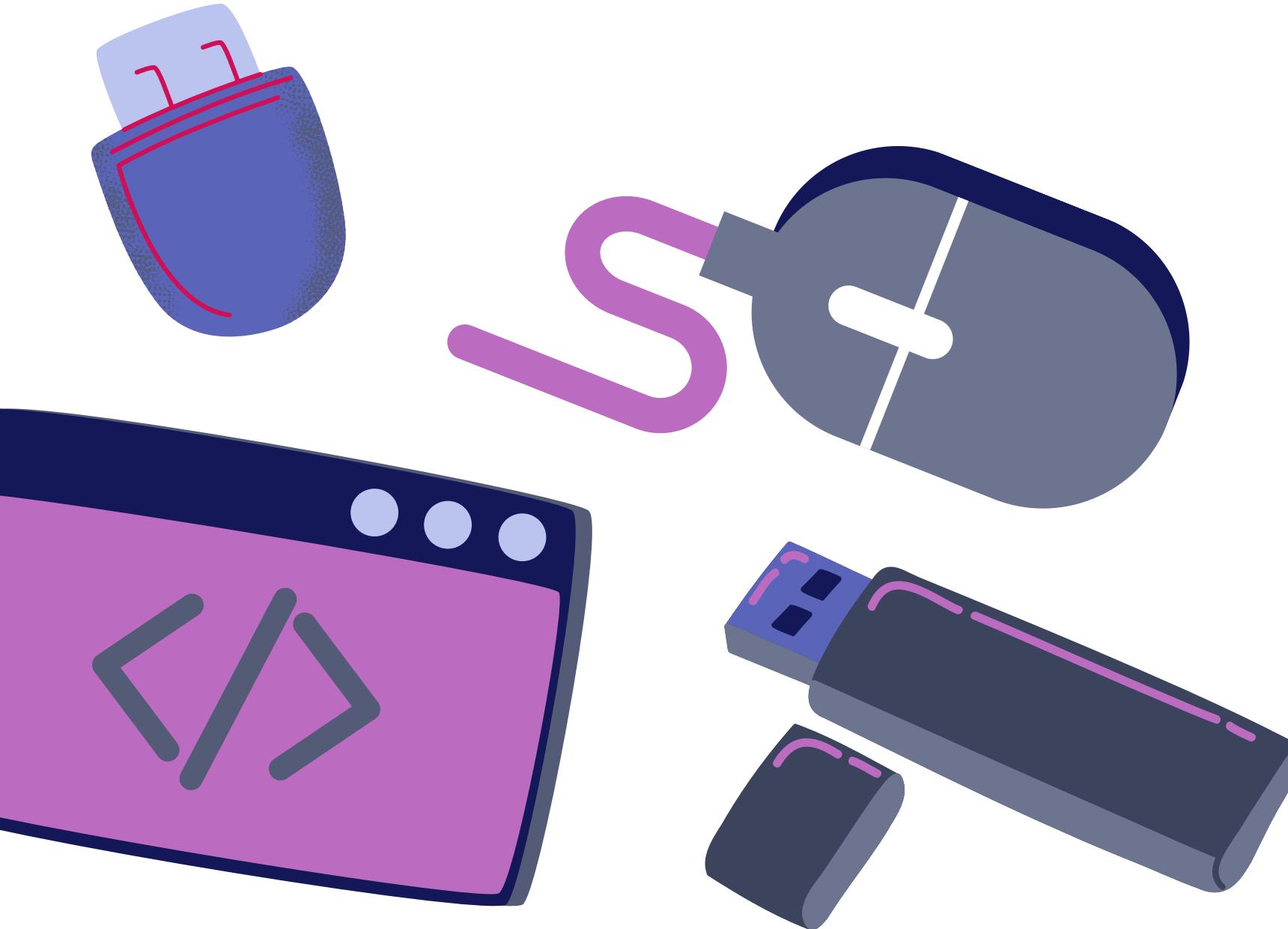


Proyecto Realizado por Martín Gómez

# PROYECTO

## *Lógica de programación*

# ÍNDICE



---

**01. Introducción**

---

**02. Propósitos**

---

**03. Selección**

---

**04. Inicio y desarrollo**

---

**05. Aplicación**

---

**06. Entrega**

---

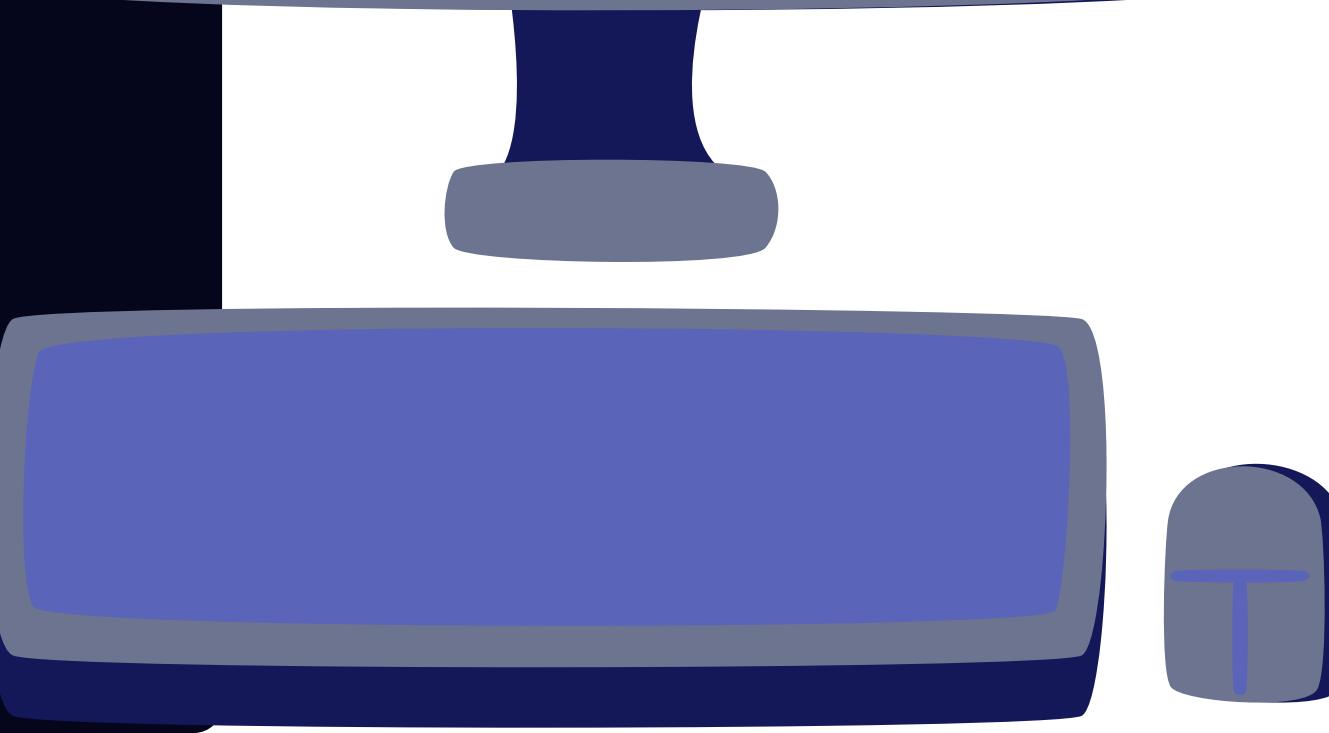
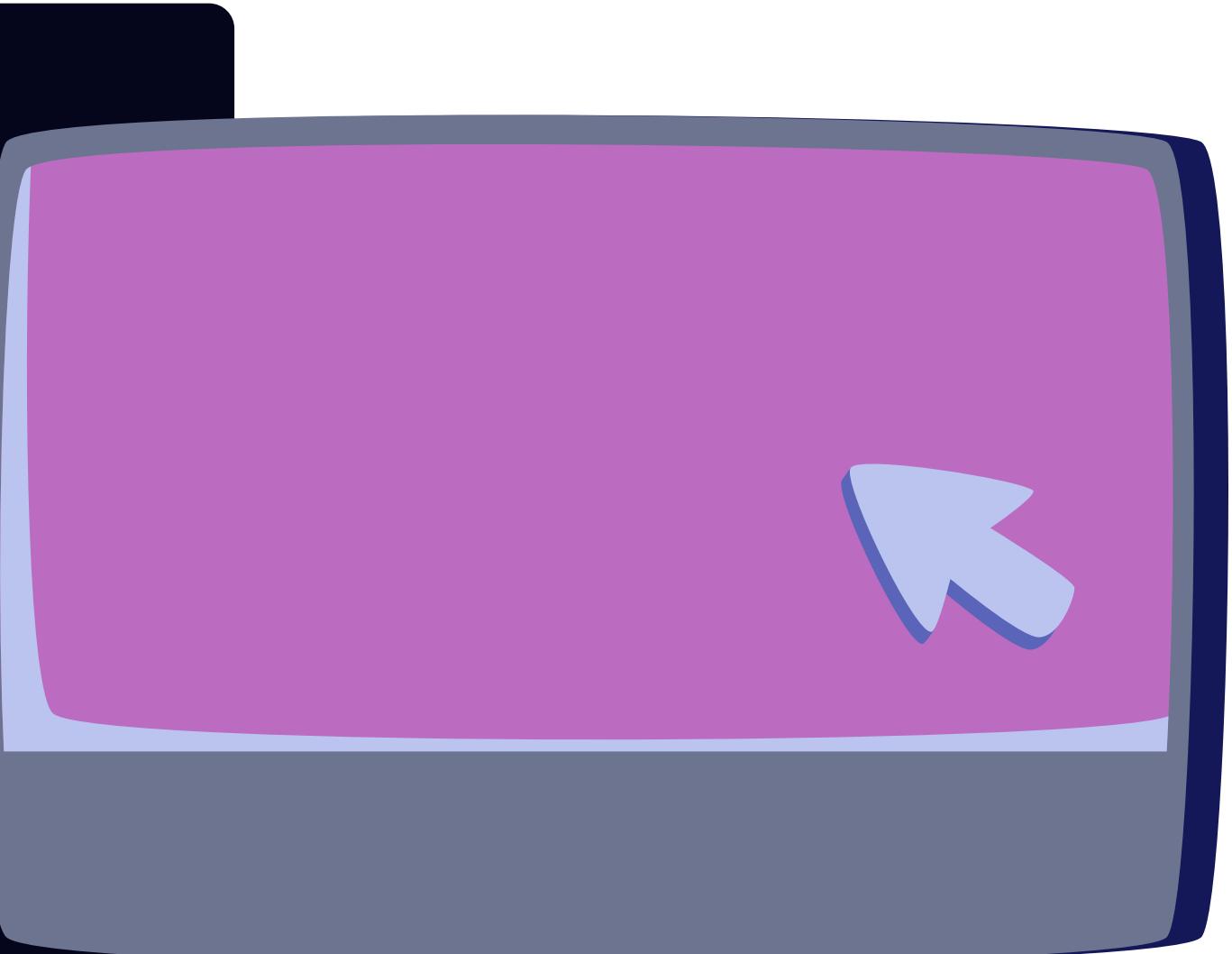
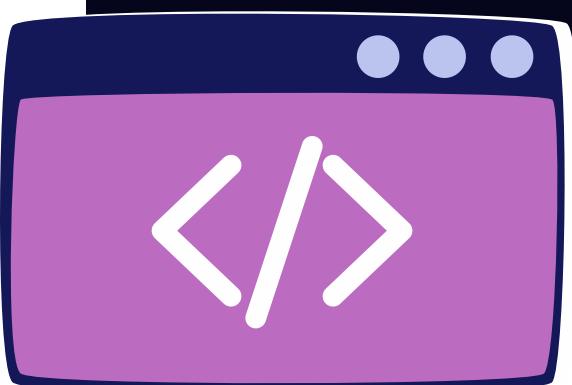
**07. Conclusiones**

---

# Introducción

En una sociedad en el que cada vez se recurre más a los servicios y plataformas electrónicas que requieren claves de acceso, las personas pasan a depender de múltiples claves de acceso, las claves o passwords de las que dan uso son débiles o repetidas, lo cual pone a los usuarios en peligro de sufrir el robo de su identidad, de que personas ajenas puedan acceder a la información personal de los usuarios, sin la debida autorización o la posibilidad que se pierde la información.

Este proyecto persigue el objetivo de informar a los usuarios sobre las mejores prácticas en cuanto a la forma en que pueden proteger sus cuentas cuando hacen uso de alguna herramienta mediante la creación de un software que sea capaz de instaurar o producir passwords difíciles de adivinar o piratear.



# PROPOSITOS

General: Visualizar el impacto de las nuevas tecnologías en la seguridad digital.

Específicos:

- Concientizar sobre la importancia de usar contraseñas seguras.
- Aplicar conocimientos de programación funcional en un software útil.
- Integrar aprendizajes de las cuatro unidades de la asignatura.



# Selección del Software

La selección del tema se basó en la relevancia social y tecnológica de la ciberseguridad en el siglo XXI.

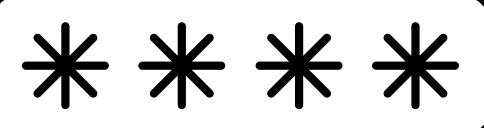
Cada día, millones de usuarios en el mundo crean cuentas, almacenan información personal y realizan transacciones en línea. Sin embargo, muchas personas aún utilizan contraseñas débiles o repetidas, facilitando los ataques de fuerza bruta o el robo de identidad.



Generador de  
contraseñas

De allí surge la idea de crear un software generador de contraseñas seguras, capaz de producir combinaciones aleatorias que cumplan con estándares modernos de protección.

Esta herramienta ejemplifica cómo la tecnología, correctamente aplicada, puede fortalecer la seguridad digital individual y colectiva.

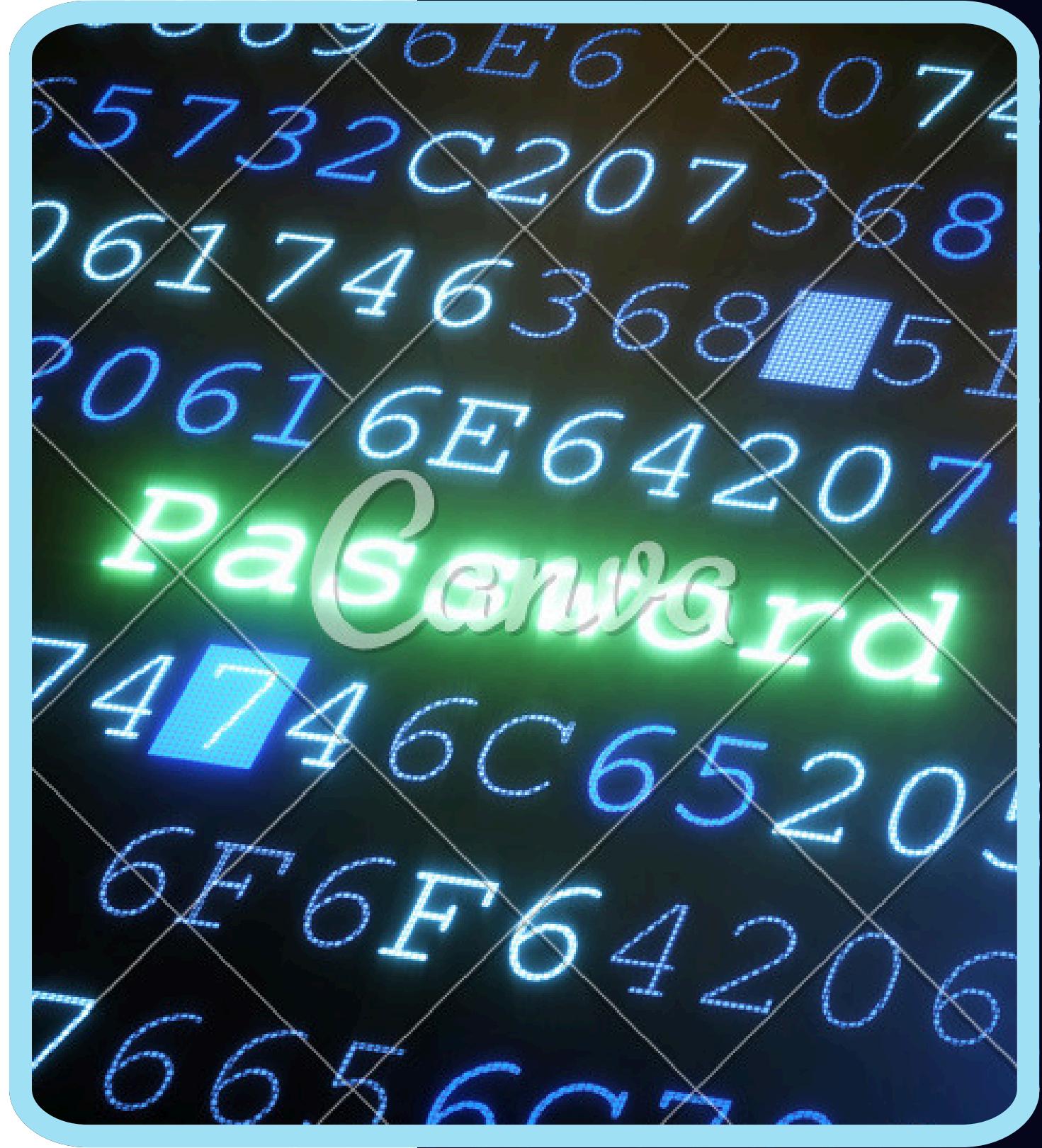


# INICIO Y DESARROLLO

- El desarrollo del proyecto comenzó con la planeación del diseño lógico del sistema, estableciendo sus principales funciones y restricciones.
- Se elaboraron diagramas de flujo y pseudocódigo para representar el proceso de generación de contraseñas.
- Posteriormente, se implementó el programa utilizando el lenguaje Python, por su versatilidad y facilidad para manejar funciones aleatorias mediante las librerías random y string.

1. **Diseño:** definición de los parámetros del generador (longitud, uso de caracteres, interfaz de entrada).
2. **Codificación:** implementación de funciones y estructuras de control.
3. **Pruebas:** ejecución repetida para asegurar la aleatoriedad y ausencia de errores.
4. **Optimización:** ajustes en la usabilidad y compatibilidad del programa.

El resultado fue un software ligero, eficaz y multiplataforma, ejecutable desde cualquier sistema operativo (Windows, Linux, macOS).



# Aplicación



El programa cumple una función práctica y directa en el ámbito de la protección de la información.

Permite al usuario seleccionar la longitud deseada de la contraseña, combinando letras mayúsculas, minúsculas, números y símbolos especiales.

Al ejecutarse, genera una clave única y segura en cuestión de segundos.

Esta aplicación puede utilizarse tanto en el ámbito personal (cuentas de redes sociales, correos electrónicos) como profesional (accesos administrativos, plataformas empresariales).

Además, representa una solución educativa y preventiva, ya que promueve la cultura de seguridad digital desde la formación académica.

# ENTREGA

```
    resp_iter = self.stub.GetHistoryStatuses(
        GetHistoryStatusesRequest())
    statuses = {}
    async for data in resp_iter:
        status = Status(
            status_id=data.id, name=data.name)
        statuses[status.name] = status
    return statuses
```

La entrega final del proyecto se realizó mediante un repositorio en GitHub, garantizando la accesibilidad y transparencia del código.

## REPOSITORIO

El repositorio incluye:

- El archivo principal del programa (`generador.py`).
- Diagramas de flujo y pseudocódigo documentados.
- Un archivo `README.md` con la descripción general, objetivos, fecha y datos del autor.
- La presentación Canva y un video demostrativo mostrando las funcionalidades del software.



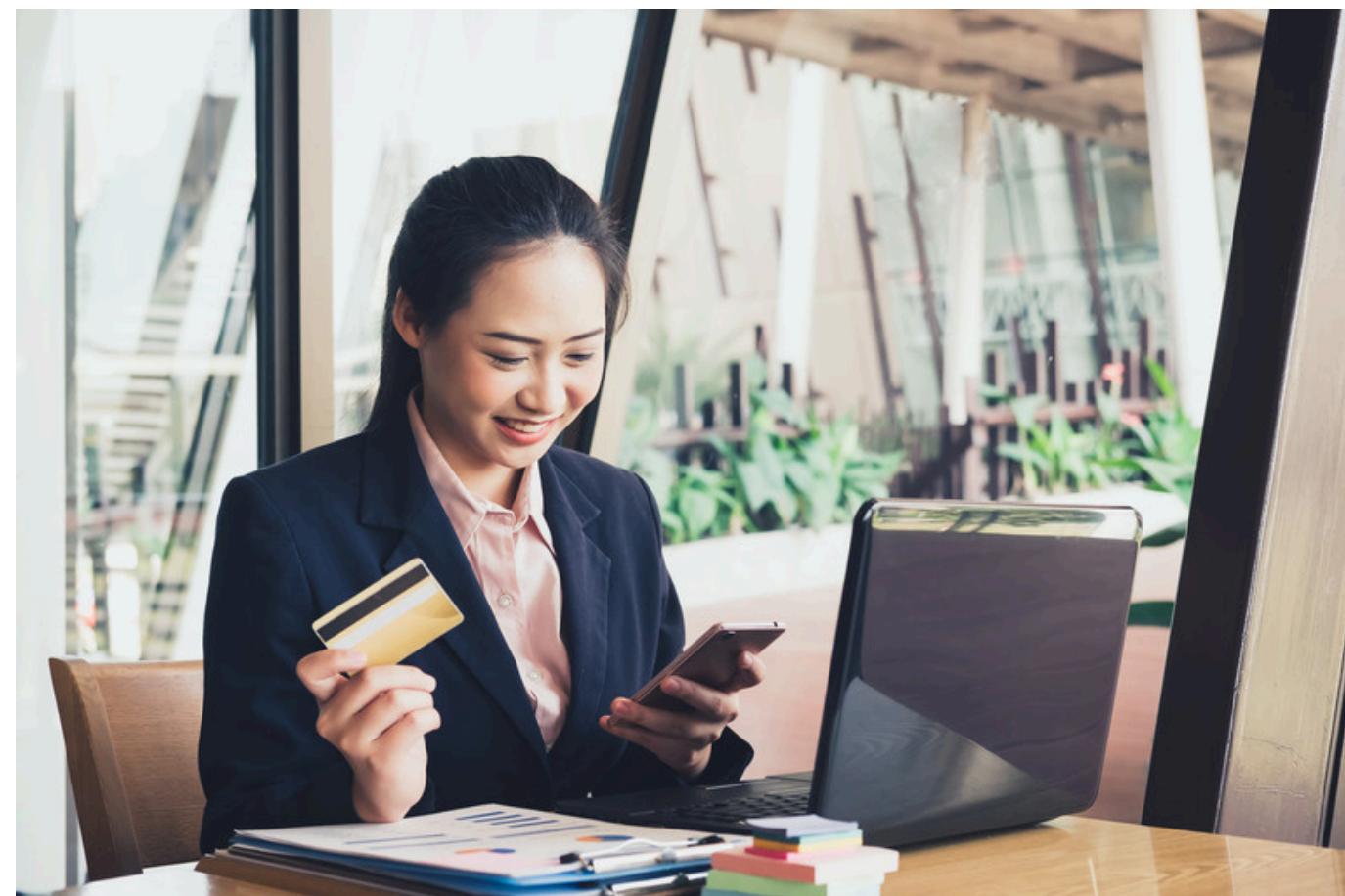
# Conclusiones

El proyecto permitió comprender que la seguridad digital comienza con pequeñas acciones individuales, como el uso de contraseñas seguras.

A través del desarrollo del software, se evidenció la importancia de la programación como herramienta para crear soluciones tecnológicas accesibles y efectivas.

- Se fortalecieron las habilidades de análisis lógico, diseño de algoritmos y documentación profesional.
- Se integraron los conocimientos de las cuatro unidades, conectando la teoría con la práctica.
- Se promovió una visión ética sobre el uso de la tecnología, orientada al bienestar y la seguridad de la sociedad.

En conclusión, el Generador de Contraseñas Seguras es una representación funcional del impacto positivo que las nuevas tecnologías pueden tener cuando se aplican con responsabilidad.



# MUCHAS GRACIAS