

GENERADOR SEGURO DE CONTRASEÑAS

"Evaluación en Contacto con el Docente"

Realizado por: Joseph Montero

Curso: Lógica de programación

Tutor: Mónica Salazar

Fecha: 24/02/2026

OBJETIVO

Desarrollar un programa en Python que permita generar contraseñas seguras de manera aleatoria, brindando al usuario la posibilidad de elegir la longitud y los tipos de caracteres, aplicando estructuras básicas de programación como condicionales, ciclos y validación de datos.

Cronograma Del Proyecto:

Semana	Unidad y Tema	Actividades	Trabajo Autónomo	Seguimiento y Evaluación	Producto
Semana 1	Unidad 1 – Tema 1	Análisis del problema y selección del proyecto.	Investigación sobre seguridad digital y definición de funcionalidades principales.	Validación del alcance del proyecto.	Documento de análisis inicial.
Semana 2	Unidad 1 – Tema 2	Diseño funcional del sistema.	Elaboración de diagramas de flujo y arquitectura del programa.	Retroalimentación y aprobación.	Diagramas funcionales aprobados.
Semana 3	Unidad 2 – Tema 3	Configuración del entorno de desarrollo.	Instalación de Python y creación del repositorio en GitHub.	Revisión de configuración.	Repositorio creado.
Semana 4	Unidad 2 – Tema 4	Desarrollo inicial del software.	Implementación de entrada de datos y generación básica.	Evaluación de avance.	Primera versión funcional.
Semana 5	Unidad 3 – Tema 5	Implementación de funcionalidades.	Validaciones y selección de caracteres.	Evaluación formativa.	Versión mejorada.
Semana 6	Unidad 3 – Tema 6	Optimización del código.	Corrección de errores y pruebas.	Retroalimentación técnica.	Versión estable.
Semana 7	Unidad 4 – Tema 7	Pruebas finales y documentación.	Elaboración del README y organización final.	Validación final.	Proyecto listo.
Semana 8	Unidad 4 – Tema 8	Entrega del proyecto integrador terminado.	Publicación en GitHub y presentación.	Evaluación final.	Software entregado.

DIAGRAMA DE ARQUITECTURA

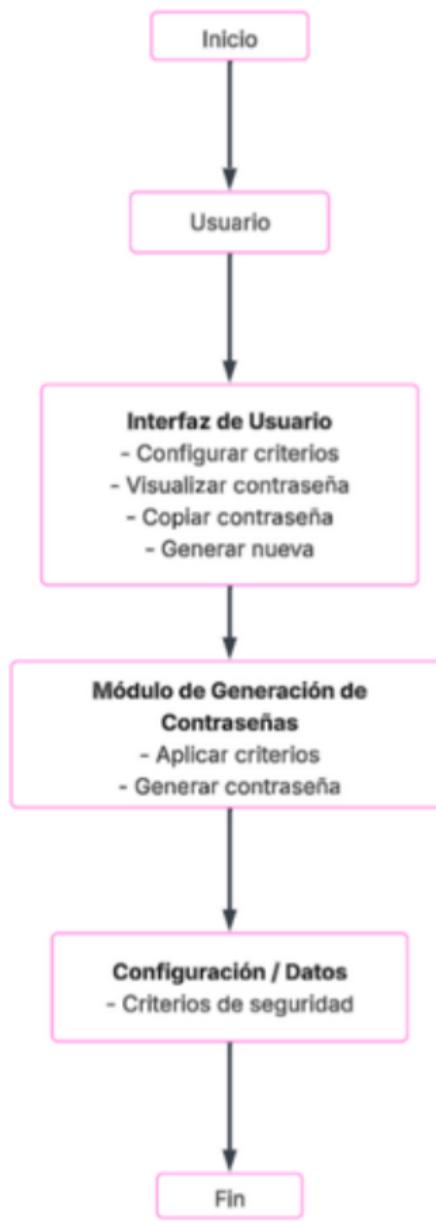




DIAGRAMA DE CASOS DE USO

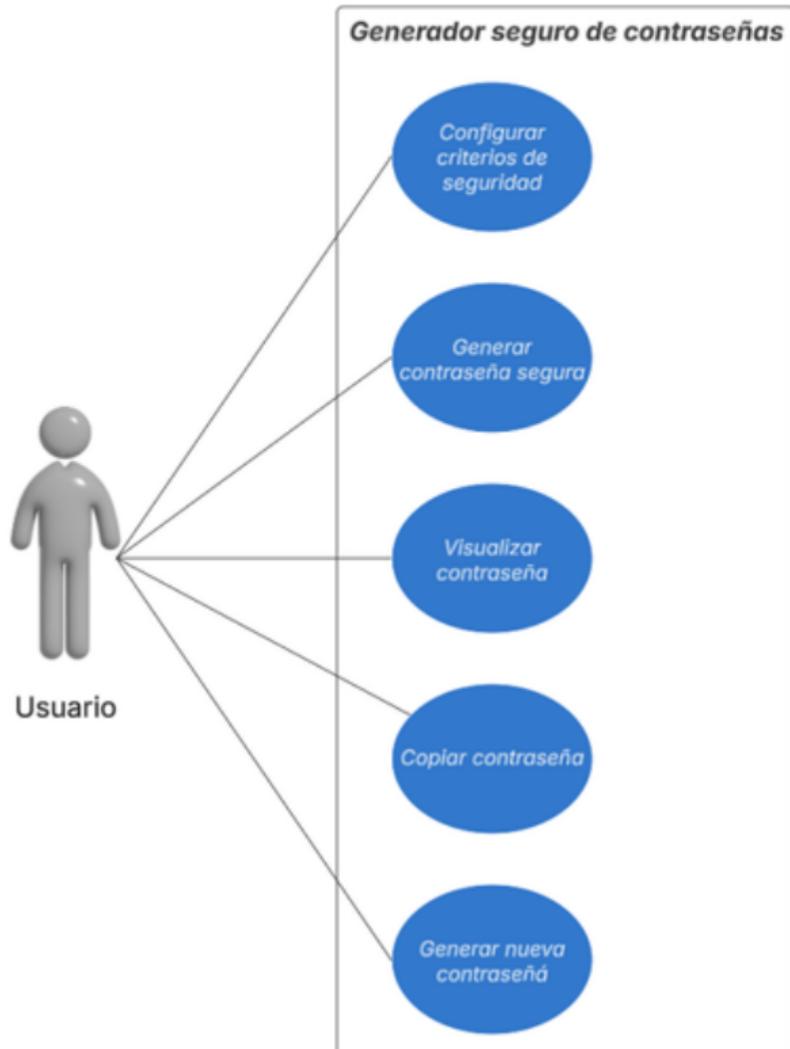
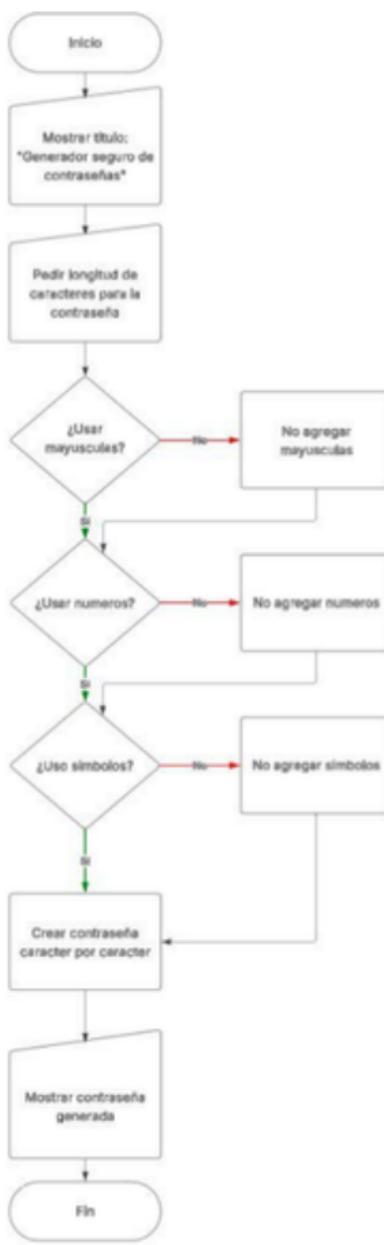
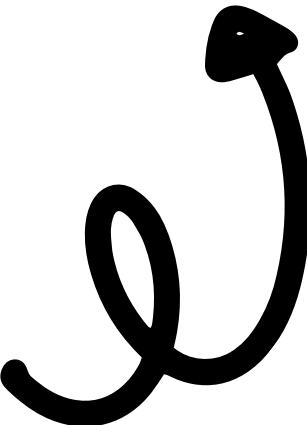
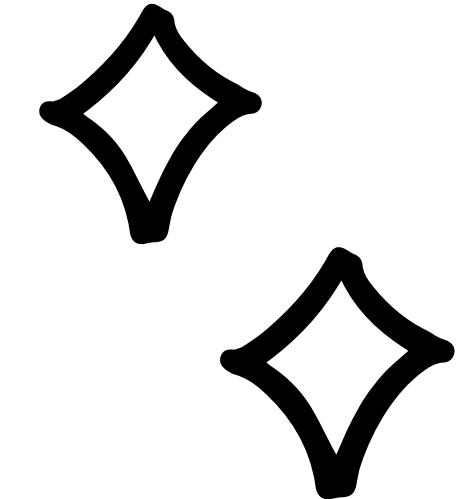
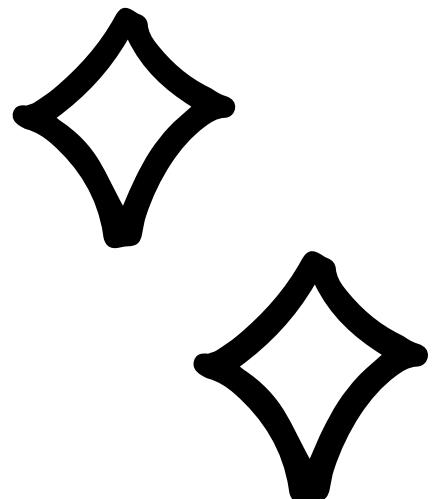


DIAGRAMA DE FLUJO



FUNCIONALIDADES



SOLICITUD DE LONGITUD

El sistema solicita al usuario la cantidad de caracteres que tendrá la contraseña.

```
while True:
    entrada = input("Ingrese la longitud de la contraseña: ")

    if entrada.isdigit(): # Verifica que sea número
        longitud = int(entrada)
        if longitud > 0:
            break
        else:
            print("Ingrese un número mayor que 0.")
    else:
        print("Ingrese un número válido.")
```

DEFINICIÓN DE TIPOS DE CARACTERES

El programa define diferentes conjuntos de caracteres:

```
# Definimos los caracteres básicos
minusculas = "abcdefghijklmnopqrstuvwxyz"
mayusculas = "ABCDEFGHIJKLMNPQRSTUVWXYZ"
numeros = "0123456789"
simbolos = "!@#$%&*"

caracteres = ""
caracteres += minusculas # Siempre agregamos minúsculas
```

SELECCIÓN DE PREFERENCIAS DEL USUARIO

El sistema pregunta al usuario si desea incluir:

```
# Preguntamos opciones
usar_mayus = input("¿Desea incluir mayúsculas? (si/no): ").lower()
if usar_mayus == "si":
    caracteres += mayusculas

usar_numeros = input("¿Desea incluir números? (si/no): ").lower()
if usar_numeros == "si":
    caracteres += numeros

usar_simbolos = input("¿Desea incluir símbolos? (si/no): ").lower()
if usar_simbolos == "si":
    caracteres += simbolos
```

GENERACIÓN ALEATORIA DE LA CONTRASEÑA

El programa utiliza la librería random para:

```
for i in range(longitud):
    letra = random.choice(caracteres)
    contraseña += letra
```

VISUALIZACIÓN

Finalmente, el sistema mostrará la contraseña generada

```
# Mostramos el resultado
print("La contraseña generada es:", contraseña)
```

MUCHAS
GRACIAS

