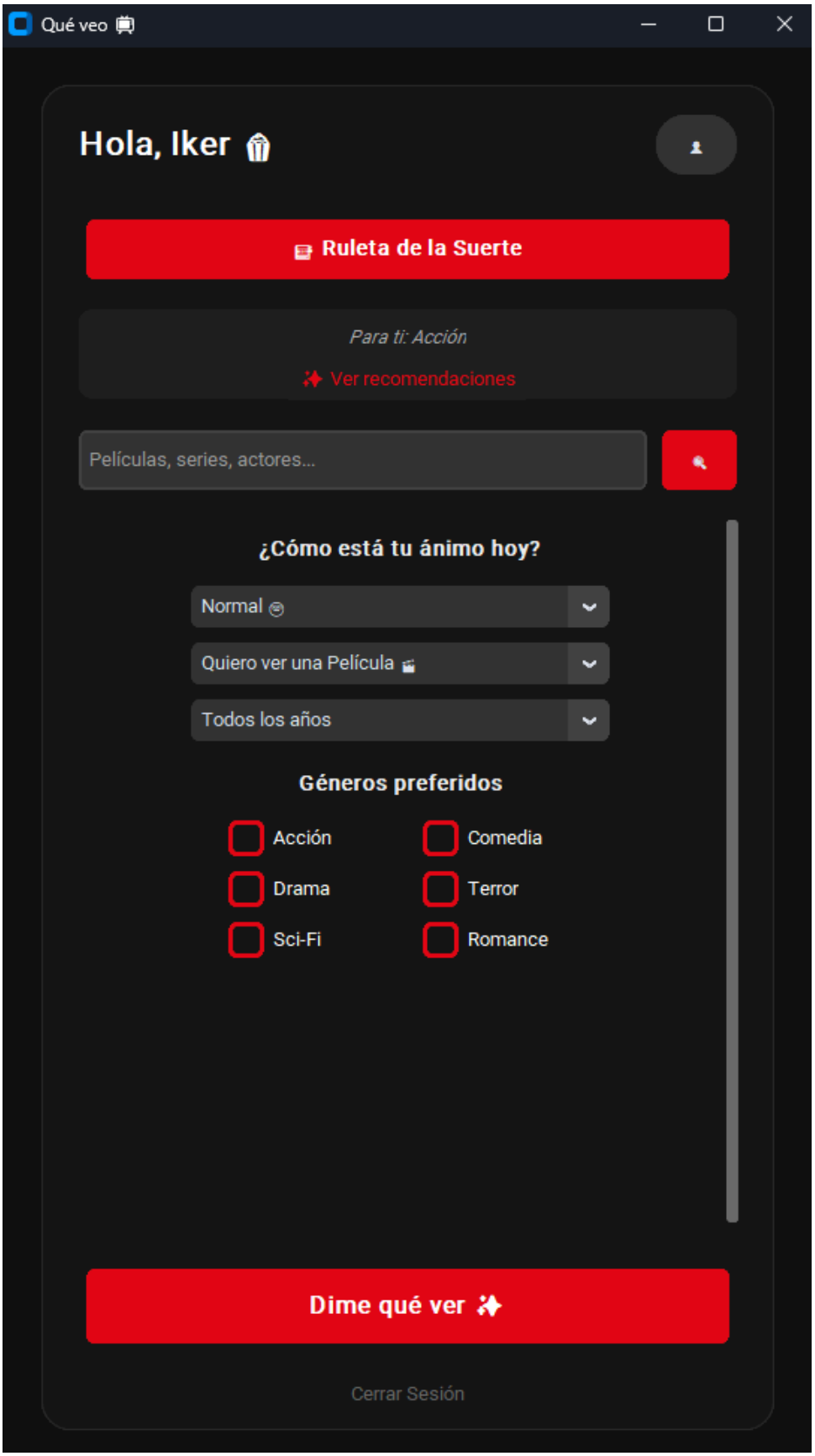
# MEMORIA DEL PROYECTO

***Qué Veo – Sistema Inteligente de Recomendación de Películas y Series***



# ÍNDICE:

# Introducción

Hoy en día hay tantas películas y series disponibles en las plataformas de streaming que muchas veces resulta difícil decidir qué ver. El proyecto **Qué Veo** surge para ayudar al usuario a elegir contenido de forma rápida y personalizada según sus gustos.

La aplicación permite registrarse, indicar preferencias y recibir recomendaciones de películas o series, mostrando información útil como sinopsis, plataformas disponibles y trailers. Para ello, se combinan datos obtenidos de APIs externas y técnicas de web scraping dentro.

# Objetivos del proyecto

El objetivo principal del proyecto es desarrollar una aplicación capaz de recomendar películas y series de forma personalizada, utilizando información procedente de diferentes fuentes externas y adaptándola a las preferencias del usuario.

De forma complementaria, se pretende implementar un sistema de autenticación seguro, aplicar buenas prácticas de programación, diseñar una arquitectura modular y escalable y ofrecer una interfaz gráfica clara e intuitiva. Asimismo, el proyecto busca sentar las bases para futuras ampliaciones relacionadas con técnicas de Inteligencia Artificial, como sistemas de recomendación más avanzados basados en aprendizaje automático.

# Tecnologías y herramientas empleadas

El desarrollo del sistema se ha realizado utilizando **Python** como lenguaje principal. Python permite una integración sencilla con APIs externas, así como la implementación de técnicas de scraping y el desarrollo de interfaces gráficas.

Para la interfaz de usuario se ha utilizado **CustomTkinter**, una librería que proporciona un diseño moderno y mejora la experiencia de uso frente a interfaces gráficas tradicionales, nos hubiera gustado usar **React** inspirandonos en nuestro compañero David, pero se nos hizo imposible por tema tiempo. La comunicación con servicios externos se ha gestionado mediante la librería **Requests**, encargada de realizar las solicitudes HTTP necesarias.

La información relacionada con películas y series se obtiene a través de la **OMDb API**, que proporciona datos estructurados como títulos, años de estreno, sinopsis, géneros y valoraciones. Estos datos se complementan mediante técnicas de **web scraping**, utilizando **BeautifulSoup** y **Selenium**, con el objetivo de identificar plataformas de streaming disponibles y enlaces a trailers.

En cuanto a la seguridad, se ha empleado la librería **bcrypt** para el cifrado de contraseñas, garantizando que las credenciales de los usuarios no se almacenan en texto plano. La persistencia de datos se gestiona mediante archivos **JSON**, utilizados para almacenar tanto usuarios como perfiles y preferencias.

## Arquitectura y diseño del sistema

El archivo principal, main.py, actúa como punto de entrada de la aplicación y gestiona la interfaz gráfica y el flujo general de interacción con el usuario. La lógica de autenticación y gestión de usuarios se encuentra encapsulada en el módulo auth_logic.py, donde se implementan el registro, la validación de contraseñas y el inicio de sesión.

La comunicación con la OMDb API se gestiona desde el módulo api_logic.py, encargado de realizar búsquedas y generar recomendaciones basadas en las preferencias del usuario. Por último, el módulo scraping_logic.py se encarga de extraer información adicional mediante técnicas de scraping, como la disponibilidad del contenido en plataformas de streaming y los enlaces a trailers.

## Funcionamiento de la aplicación

El uso de la aplicación comienza con el registro del usuario, quien debe introducir un correo electrónico y una contraseña que cumpla una serie de requisitos de seguridad, como una longitud mínima y la combinación de mayúsculas, números y caracteres especiales. Una vez validada, la contraseña se cifra y se almacena de forma segura.

Tras iniciar sesión, el usuario completa un breve cuestionario en el que indica sus preferencias cinematográficas. Esta información se utiliza como base para personalizar las recomendaciones.

El sistema consulta la OMDb API para obtener una primera selección de contenidos que se ajusten a los criterios indicados. Posteriormente, se realiza un proceso de enriquecimiento de datos mediante scraping, identificando en qué plataformas está disponible cada contenido y obteniendo enlaces a trailers oficiales.

Los resultados se presentan al usuario a través de una interfaz visual basada en tarjetas, mostrando información relevante como título, valoración, sinopsis y accesos directos tanto a la plataforma de streaming como al trailer correspondiente.

## Seguridad y gestión de datos

La seguridad ha sido un aspecto fundamental en el desarrollo del proyecto. Las contraseñas se almacenan cifradas utilizando el algoritmo bcrypt, lo que impide el acceso directo a información sensible incluso en caso de acceso no autorizado a los archivos de datos.

Además, se han implementado validaciones de entrada y manejo de excepciones para evitar errores durante las solicitudes a servicios externos. La separación entre los datos de autenticación y los perfiles de usuario contribuye a una mejor organización y protección de la información.

## Integración de APIs y web scraping

La combinación de una API pública con técnicas de web scraping ha permitido enriquecer significativamente la información mostrada al usuario. Mientras que la OMDb API proporciona datos estructurados y fiables, el scraping permite acceder a información adicional que no siempre está disponible mediante APIs oficiales.

## Problemas encontrados y soluciones adoptadas

Durante el desarrollo del proyecto se presentaron diversos desafíos, especialmente relacionados con la obtención de enlaces directos a plataformas de streaming y la variabilidad de las estructuras HTML utilizadas en el scraping. Estos problemas se abordaron mediante el uso de selectores flexibles, validaciones adicionales y mecanismos de control de errores.

## Conclusiones

El proyecto **Qué Veo** da como resultado una aplicación funcional que cumple con los objetivos planteados, ofreciendo recomendaciones personalizadas y una experiencia de usuario clara e intuitiva, nos gustaría continuar desarrollando y mejorando la aplicación.