



# HABITRACKER

## ELECCIÓN DE STACK PARA LA APLICACIÓN

Investigación: Analizador de hábitos (gamificado)  
Elaborado por: Developers  
Profesor: César Antonio Ríos Olivares  
Materia: Programación orientada a objetos  
Fecha: 17/10/2025

## Elección de Stack para la Aplicación “Analizador de Hábitos” Objetivos del Proyecto

El objetivo principal de la aplicación “Analizador de Hábitos” es permitir que los usuarios registren, visualicen y analicen sus hábitos diarios (como ejercicio, sueño, alimentación o estudio), con el fin de mejorar su productividad y bienestar personal. Las funcionalidades clave incluyen:

- Registro y seguimiento de hábitos mediante formularios interactivos.
- Visualización de estadísticas y gráficos de progreso.
- Almacenamiento local o en la nube de los datos del usuario.
- Interfaz amigable y multiplataforma (funcional tanto en PC como en dispositivos móviles).

Dado que se busca una aplicación descargable y multiplataforma, se optará por tecnologías que permitan generar interfaces modernas sin depender del navegador. Habilidades del Equipo

El equipo cuenta con conocimientos en:

- Python, como lenguaje principal de programación.
- GitHub, para control de versiones y trabajo colaborativo.
- Flet, como framework para crear aplicaciones gráficas modernas con Python sin necesidad de usar HTML, CSS o JavaScript.

Estas herramientas permiten desarrollar la aplicación de forma eficiente aprovechando las habilidades actuales del equipo, sin necesidad de aprender lenguajes adicionales. Requerimientos Técnicos

1.

Escalabilidad: El uso de Python junto con librerías como SQLite o Firebase (en una posible versión futura) permitirá escalar el proyecto fácilmente al manejar más usuarios y datos.

2.

Rendimiento:

Flet proporciona un rendimiento óptimo para aplicaciones ligeras de análisis y registro de datos. Además, Python permite procesar la información de manera rápida y eficiente.

3.

Seguridad:

Python cuenta con diversas librerías que facilitan la implementación de cifrado y protección de datos. GitHub permite mantener el código en repositorios privados, asegurando versiones controladas y seguras.

4.

Mercado laboral:

Python es uno de los lenguajes más demandados en el mercado tecnológico, tanto para desarrollo de software como para análisis de datos, inteligencia artificial y automatización.

5.

Comunidad y soporte:

Python tiene una comunidad muy activa, con abundante documentación, foros y cursos gratuitos.

Flet, aunque más reciente, está creciendo rápidamente y cuenta con buena documentación oficial y soporte en GitHub.

6.

Costo:

El stack propuesto es completamente gratuito y de código abierto, lo que lo hace ideal para un proyecto universitario sin presupuesto.

## Comparación con Otros Stacks

Stack	Lenguaje	Tipo de aplicación	Ventajas	Desventajas
PYTHON + FLET GITHUB	+ Python	Multiplataforma (desktop/móvil)	Fácil de gratuito, compatible habilidades equipo	usar, Flet aún en desarrollo con (limitaciones en del algunas funciones avanzadas)

## Cómo Empezar

1. Definir el alcance: Establecer las funciones básicas (registro de hábitos, estadísticas y guardado de datos).
2. Diseñar prototipos en Flet: Crear la interfaz principal y probar la navegación.
3. Configurar el repositorio en GitHub: Implementar control de versiones para colaboración y respaldo.
4. Desarrollar módulos en Python: Programar la lógica de registro, análisis y visualización.
5. Realizar pruebas: Validar rendimiento, usabilidad y compatibilidad en diferentes dispositivos.

## Conclusión

El stack Python + Flet + GitHub es la opción ideal para el desarrollo del “Analizador de Hábitos” porque:

- Se adapta a las habilidades del equipo.
- Permite crear una aplicación moderna, multiplataforma y sin costos adicionales.
- Ofrece buena escalabilidad, rendimiento y soporte comunitario.
- Favorece la práctica de programación orientada a objetos en Python, reforzando los conocimientos adquiridos en la materia.