

Grupo: K

Integrantes: Gabriela Rivera – Mario Ramírez – Cecilia Tomala – Fernando Benavides

Perfil de Proyecto de Software

1. Datos generales

- **Nombre del proyecto:** Hábitos & Rachas
- **Cliente / Usuario final:** Público general desde los 12 años (Mayor eficiencia en estudiantes).
- **Equipo de trabajo:** Product Owner, Desarrolladores, QA, UX/UI.
- **Fecha de inicio prevista:** 20/08/2025
- **Duración estimada:** 3 semanas

2. Visión del producto

Ayudar a las personas a construir constancia con una app web mínima que permita crear hábitos diarios, marcar el progreso con un clic y ver rachas y métricas simples, entregando motivación y claridad sin complejidad.

3. Objetivos

- **Objetivo general:** Crear, desarrollar y lanzar un producto web para el seguimiento de hábitos que esté listo para su uso práctico, que facilite establecer rutinas, registrar las actividades con un solo clic y mostrar rachas y estadísticas básicas de manera segura y confiable.
- **Objetivos específicos:**
 1. Implementar inicio de sesión con GitHub (OAuth) y rutas protegidas.
 2. Desarrollar CRUD de hábitos (nombre y color/emoji opcional) persistido en MongoDB.
 3. Registrar check-ins diarios con control de un registro por día y por hábito.
 4. Calcular y mostrar racha actual, mejor racha y porcentaje de cumplimiento por hábito.
 5. Entregar interfaz HTML/CSS/JS responsiva, con manejo de errores y validaciones.
 6. Publicar el producto con documentación de instalación/uso y consideraciones básicas de seguridad.

4. Alcance inicial

- Funcionalidades mínimas: registro/login/logout; CRUD de hábitos (nombre y color/emoji opcional); check-in diario (único registro por día y por hábito); estadísticas básicas (racha actual, mejor racha y % de cumplimiento); UI responsive con dashboard (lista + botón “✓”).
- Criterios de aceptación generales:
 1. Un usuario válido accede a vistas protegidas tras iniciar sesión.
 2. Al marcar “✓”, se crea un check-in único.

3. Las rachas se recalculan correctamente.
 4. Los datos persisten en MongoDB y se reflejan al recargar la página.
- Plataformas objetivo (web, móvil, escritorio): Web responsive y móvil en navegador.

5. Restricciones y supuestos

- Node.js + Express; MongoDB (Mongoose); Front HTML/CSS/JS; Auth OAuth 2.0 GitHub (Passport + express-session); seguridad CORS; IA: Gemini API para sugerir hábitos.
- App OAuth en GitHub; MongoDB Atlas; hosting (Render); Google AI Studio/Gemini API key (vars de entorno).
- Riesgo de coordinación, uso de free tiers (MongoDB Atlas, hosting, Gemini API); posibles límites de cuota.

6. Planificación inicial

Representa las **entregas por sprint** (plan alto nivel):

Sprint	Objetivo	Entregable funcional
Sprint 1	Configuración y arquitectura	Repo y base del proyecto (Node/Express + Mongo); login/logout con OAuth GitHub (rutas protegidas); UI inicial: Login y Dashboard vacío.
Sprint 2	Gestión de hábitos y Check-in	CRUD de hábitos (nombre/color); check-in diario con índice único (habitId, date); dashboard con botón “✓”.
Sprint 3	Estadísticas, IA y despliegue	Racha actual, mejor racha y % por hábito; endpoint “Sugerir hábitos” con Gemini y botón en UI; manejo de errores/validaciones; despliegue público + README.