

Generador de Rutinas de Aprendizaje Personalizadas

Desarrollar una aplicación web que permita a los usuarios organizar su rutina de aprendizaje semanal de forma visual e interactiva. El usuario podrá registrar sus intereses y objetivos, y luego asignar manualmente actividades a diferentes días y franjas horarias de la semana. La aplicación permitirá visualizar, modificar y guardar estas rutinas personalizadas.

Módulos principales

Frontend (Angular + Bootstrap o css):

- Registro / login (con JWT).
- Panel con:
 - Formulario para ingresar intereses, objetivos y disponibilidad.
 - Visualización de la rutina semanal (tipo calendario o lista por día).
 - Botón para **"Generar nueva rutina"** según preferencias.
 - Edición rápida de actividades.

Backend (Node/Express + MongoDB o MySQL):

- Autenticación JWT.
- Endpoints REST para:
 - Guardar/obtener perfil.
 - Guardar/consultar rutinas y actividades.
 - Generar nuevas rutinas a partir de reglas definidas en el interfaz de creación de rutinas

Objetivo del Proyecto

El objetivo de este proyecto es desarrollar una aplicación web completa que permita a los usuarios generar rutinas de aprendizaje personalizadas a partir de sus intereses y disponibilidad. El usuario podrá visualizar su rutina semanal y modificarla, y el sistema podrá generar nuevas rutinas automáticamente siguiendo una lógica personalizada.

Este proyecto tiene como fin poner en práctica lo aprendido durante el curso en tecnologías de frontend (Angular, HTML, CSS, Bootstrap, JavaScript) y backend (Node.js, Express, API REST), así como el uso de bases de datos (MongoDB o MySQL) a tu elección.

Tecnologías que deben usarse

- **Frontend:** Angular, Bootstrap
 - **Backend:** Node.js + Express (API REST)
 - **Base de datos:** MongoDB, MySQL debes tomar tú la decisión de qué motor de base de datos se adapta mejor a tu planteamiento.
 - **Autenticación tanto en frontend como en Backend**
-

Funcionalidades principales (obligatorias)

Estas funcionalidades forman el núcleo del proyecto. Se espera que estén todas completadas y funcionando correctamente:

1. **Registro e inicio de sesión de usuario.**
2. **Formulario de configuración de perfil:**
 - Intereses (ej. "Aprender JavaScript")
 - Objetivos (ej. "Estudiar 5 horas por semana")
3. **Creación manual de rutina semanal:**
 - El usuario podrá añadir actividades a días específicos y franjas horarias.
 - Ejemplo: Lunes 18:00–19:00 → "Practicar ejercicios de JavaScript".
 - Permitir guardar varias rutinas diferentes.
4. **Visualización de la rutina semanal** en formato lista o tipo calendario.

5. **Edición y eliminación de actividades y rutinas.**
 6. **Guardar y recuperar las rutinas desde la base de datos.**
 7. **Frontend desarrollado en Angular** con diseño responsive y uso de Bootstrap (opcional)
-

Funcionalidades opcionales (para alumnos que quieran más retos o que trabajen en grupo)

Estas características son opcionales y sirven para demostrar un nivel más avanzado o un esfuerzo extra:

- Crear una **librería de actividades frecuentes/favoritas** para que el usuario las pueda reutilizar.
 - Integración de **notificaciones o recordatorios por correo o en la propia aplicación**
 - Agrupar actividades por categorías (ej. "Técnico", "Lectura", "Idiomas").
 - Integrar un contador de tiempo de uso de la aplicación
 - Posibilidad de compartir rutinas con el resto de usuarios y poder agregarlas.
-

Estructura mínima esperada del proyecto

1. Frontend (Angular)

- login/register.component.ts/html
- home parte publica donde se explica un poco para que sirve la aplicación y como pagina de aterrizaje o landing page
- dashboard.component.ts/html – vista principal tras iniciar sesión.
- profile.component.ts/html – formulario de intereses y objetivos.
- routine.component.ts/html – vista de la rutina generada.
- routine.service.ts – comunicación con la API.

2. Backend (Node/Express)

- Rutas para usuarios: /api/users/register, /api/users/login
 - Rutas para perfil: /api/profile
 - Rutas para rutinas: /api/routine, /api/routine/generate
 - Modelo de datos con Mongoose o MySQL (usuarios, rutinas, favoritos(opcional))
-

Entrega y presentación del proyecto

Además de subir el código del proyecto (en una carpeta organizada o en un repositorio), deberás entregar:

1. Video de presentación del proyecto (máx. 10 minutos)

El video debe incluir:

- Breve presentación tuya (nombre, descripción del proyecto).
- Explicación de cómo se estructura el proyecto (qué tecnologías usaste y por qué).
- Demostración práctica:
 - Registro e inicio de sesión.
 - Creación de perfil.
 - Generación de rutina.
 - Modificación de tareas.
 - Cualquier funcionalidad opcional implementada.

Formato recomendado: grabación de pantalla con tu voz explicando (puedes usar Loom, OBS Studio, Screencastify o cualquier herramienta similar). El video en formato MP4

2. README del proyecto en GITHUB

Debe incluir:

- Breve descripción del proyecto.

- Instrucciones para ejecutar el backend.
 - Instrucciones para ejecutar el frontend.
 - Tecnologías utilizadas.
 - Capturas de pantalla si lo deseas.
-

Tiempo estimado

Tienes **tres semanas** para completar el proyecto. Se recomienda distribuir el tiempo así: Tendrás a un profesor en ese tiempo para responder a tus dudas y preguntas sobre el proyecto.

Semana	Objetivos
1	Estructura general, backend básico, conexión con BBDD, login/registro.
2	Desarrollo de formularios, generación de rutina, visualización.
3	Detalles, mejoras opcionales, grabación del video y preparación de entrega.