

# Planificador UCN (NestJS + React + Mongo)

---

## Tecnologias

- Backend: NestJS + Mongoose + Axios + Swagger opcional + Helmet + rate limit
- Frontend: React + Vite + Tailwind CSS
- Base de datos: MongoDB
- Integraciones: APIs UCN (Login, Malla, Avance) con stubs y respaldos CSV/JSON

## Requisitos

- Node.js 20+
- MongoDB 7+ (o Docker)

## Configuracion

### Backend

1. Copia `backend/.env.example` a `backend/.env` y ajusta:
  - `MONGO_URI` cadena de conexion
  - `ALLOWED_ORIGINS` (por defecto `http://localhost:5173,http://localhost:8080`)
  - `USE_STUBS=true` para trabajar offline
  - `ADMIN_API_KEY` clave para endpoints admin
  - `USE_BACKUP_FALLBACK=true` para servir respaldos si UCN falla
  - `SWAGGER_ENABLED=true` para ver `/api/docs` en dev

### Frontend

1. Copia `frontend/.env.example` a `frontend/.env` (por defecto `VITE_API_BASE=http://localhost:3000/api`)

## Ejecutar en desarrollo

- Local
  - Backend: `cd backend && npm install && npm run start:dev`
  - Frontend: `cd frontend && npm install && npm run dev` → abre `http://localhost:5173`
  - Desde la raiz: `npm install && npm run all:dev` (levanta ambos)
- Docker
  - `docker compose up --build`
  - Frontend: `http://localhost:8080` | Backend: `http://localhost:3000`

## Uso (UI)

1. Login (demo)

- Ingresa RUT y password (ver "Credenciales demo"). Al autenticar se listan carreras/catalogos del estudiante (elige una).
- "Olvide mi contraseña": valida RUT + email y muestra mensaje de exito (solo demostrativo, no envia correos).
- Acceso administrador esta en la pagina de login (no en la interfaz del estudiante).

## 2. Plan (/plan)

- Define tope de creditos y genera la seleccion principal (prioriza reprobados, luego atrasados segun nivel objetivo y luego prioritarios, siempre respetando prerrequisitos y tope).
- Debajo de la seleccion principal veras:
  - "Malla (selector de prioritarios)": marca cursos y agregalos a la lista de prioritarios.
  - "Opciones sugeridas": genera alternativas sin oferta. Se muestran opciones que incluyen tus cursos prioritarios y mantienen reprobados cuando aplica. Cada curso se muestra como:
    - COD · Asignatura (creditos) [prioridad, reprobado] (sin [PENDIENTE]).
- Guarda la proyeccion (o como favorita).

## 3. Mis proyecciones (/proyecciones)

- Lista, expande "Ver malla" para ver cursos y creditos, marca favorita y borra. Puedes renombrar cada proyeccion.

## 4. Demanda (/demanda)

- Ver demanda agregada por codigo o por NRC (solo favoritas).

## 5. Admin y oferta

- En /admin ingresa X-ADMIN-KEY para acceder a carga de oferta CSV en /oferta (solo admin). En esta demo la generacion "con oferta" esta deshabilitada.

# Credenciales demo

- Estudiante 1: rut 333333333, email juan@example.com, password 1234 (ICCI 201610)
- Estudiante 2: rut 222222222, email maria@example.com, password abcd (ITI 202410)
- Estudiante 3: rut 111111111, email ximena@example.com, password qwerty (ICCI 201610)

# API destacado

- Health: GET /api/health
- Auth (demo): POST /api/auth/login, POST /api/auth/forgot
- UCN (proxy): GET /api/ucn/malla/:cod/:catalogo, GET /api/ucn/avance?rut&codcarrera
- Proyecciones: generar / generar-opciones / guardar / favorita / renombrar / borrar / demanda
- Admin: oferta CSV y respaldos UCN (requiere X-ADMIN-KEY)

# Problemas comunes

- Si Vite usa otro puerto, abre la URL "Local: ..." que imprime y agrega ese origen a ALLOWED\_ORIGINS si es necesario.
- Si Mongo no levanta, usa Docker o revisa MONGO\_URI.

