

# Proyecto de Calculo y Visualizacion de Retencion Estudiantil - UCN

---

Este proyecto consiste en un sistema de backend diseñado para procesar registros historicos de la Universidad Catolica del Norte (UCN) con el fin de calcular indicadores de retencion estudiantil por carrera, año de cohorte y plan de estudio.

## Descripcion del Sistema

El sistema procesa registros de estados academicos para determinar la permanencia de los estudiantes. Se basa en una arquitectura desacoplada que permite el cambio de fuentes de datos (actualmente implementado con carga de archivos JSON hacia una base de datos relacional).

## Definicion de Retencion (Regla de Negocio)

De acuerdo a los requerimientos, la Retencion Año 1 (año+1) se calcula bajo los siguientes criterios:

1. **Primera matricula valida:** Un estudiante es contabilizado en la cohorte de un año "Y" si posee un registro con codigo de estado "M" (Matriculado) donde su año de admision coincide con el año del estado.
2. **Estudiante Retenido:** Un estudiante de la cohorte "Y" se considera retenido si existe un registro con codigo de estado "M" en el año "Y+1" para la misma carrera y catalogo.
3. **Indicadores:**
  - **Matriculados primera vez:** Total de estudiantes que cumplen el punto 1.
  - **Retenidos año siguiente:** Estudiantes del grupo anterior que cumplen el punto 2.
  - **Tasa de retencion:** Procentaje resultante de (Retenidos / Matriculados).

## Arquitectura

El backend esta construido siguiendo principios de separacion de capas para garantizar la mantenibilidad y escalabilidad:

- **Capa de Acceso a Datos (Repository):** Encargada de la persistencia. El sistema esta diseñado para trabajar con una base de datos relacional, permitiendo la migracion desde fuentes JSON.
- **Capa de Negocio (Service):** Contiene la logica de calculo de cohortes y validacion de estados de retencion.
- **Capa de API (Controller):** Expone los endpoints REST y gestiona la comunicacion con el cliente.

## Requisitos Previos

- Docker y Docker Compose
- Node.js (version LTS recomendada)
- npm

## Instrucciones de Ejecucion

1. **Iniciar Base de Datos:** Utilice Docker para levantar la instancia de la base de datos necesaria para el almacenamiento de los registros.

```
docker-compose up -d
```

2. **Instalar Dependencias:** Acceda al directorio del backend e instale los modulos necesarios.

```
cd backend  
npm install
```

3. **Configuracion de Datos (Seeding):** Asegurese de que el archivo de datos `output.json` se encuentre en el directorio `backend/database/seed/`.

4. **Ejecutar Aplicacion:** Inicie el servidor en modo desarrollo.

```
npm run start:dev
```

5. **Acceder a la Documentacion de la API:** Una vez en ejecucion, puede acceder a la documentacion interactiva en:

```
http://localhost:3000/api
```

6. **Cargar Datos:** En la documentacion de la API, utilice el endpoint para listar los archivos disponibles en `backend/database/seed/` y luego ejecute el endpoint correspondiente para cargar un archivo especifico. Tenga en cuenta que al cargar un archivo, los datos se agregan a la base de datos sin eliminar los registros anteriores.

- `GET /api/retencion/carreras`: Lista las carreras disponibles en el sistema.
- `GET /api/retencion/por-carrera?cod_programa={codigo}`: Entrega el detalle de retencion para una carrera especifica.

## Endpoints Principales

Una vez iniciada la aplicacion, puede acceder a la documentacion interactiva (Swagger) en:

`http://localhost:3000/api`

Los endpoints disponibles incluyen:

- `GET /api/student/filenames`: Obtiene la lista de archivos de datos cargados en el sistema.
- `POST /api/student/seed/:filename`: Obtiene la lista de archivos de datos cargados en el sistema.
- `GET /api/retencion/resumen`: Obtiene la retencion global por año para todas las carreras.

# Vistas y Uso

El sistema permite filtrar la informacion por:

- Codigo de programa (carrera).
- Catalogo (plan de estudios).
- Rangos de años.

La informacion entregada por la API incluye el año de cohorte, identificadores de carrera, conteo de matriculados iniciales, conteo de retenidos y la tasa porcentual de retencion.