

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 soporta dos modos de operación:

1. **Base de Datos (MongoDB):** Datos persistidos cargados previamente.
2. **Archivo Local (JSON):** Cálculo al vuelo leyendo directamente un archivo de origen.

Definicion de Retencion (Regla de Negocio)

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

1. **Primera matricula valida:** Un estudiante se considera 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:** Porcentaje 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 (MongoDB) o lectura de archivos (File System).
- **Capa de Negocio (Service):** Contiene la logica pura de calculo de cohortes y validacion de estados de retencion, independiente de la fuente de datos.
- **Capa de API (Controller):** Expone los endpoints REST y gestiona los filtros y parámetros de consulta.

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 MongoDB necesaria.

```
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 los archivos de datos (ej. `output.json`) se encuentren en el directorio `backend/database/seed/` para poder cargarlos o leerlos directamente.
4. **Ejecutar Aplicacion:** Inicie el servidor en modo desarrollo.

```
npm run start:dev
```

5. **Acceder a la Documentacion:** Una vez en ejecucion, puede acceder a la documentacion interactiva (Swagger) en:

```
http://localhost:5000/api
```

Endpoints Principales y Uso

Cálculo de Retención

Los endpoints de retención aceptan parámetros opcionales para filtrar o cambiar la fuente de datos.

Parámetros Query (Opcionales):

- `from` (number): Año de inicio de la cohorte (ej: 2018).
- `to` (number): Año de fin de la cohorte (ej: 2022).
- `catalogo` (string): Filtrar por plan de estudios específico.
- `filename` (string): **Nuevo feature.** Si se especifica (ej: `output.json`), el sistema ignora la base de datos y calcula la retención leyendo directamente ese archivo local.

Rutas:

1. **Resumen Global**

- `GET /api/retencion/resumen`
- *Ejemplo de uso:* Obtener resumen entre 2015 y 2020 leyendo desde archivo:
`GET /api/retencion/resumen?from=2015&to=2020&filename=output.json`

2. Por Carrera Específica

- `GET /api/retencion/por-carrera?cod_programa={codigo}`
- *Ejemplo de uso:* Ver retención de Ingeniería Civil (cod: 1234) del plan 2010:
`GET /api/retencion/por-carrera?cod_programa=1234&catalogo=2010`

3. Listado de Carreras

- `GET /api/retencion/carreras`
- Lista todas las carreras disponibles en la fuente de datos.

Gestión de Datos (Seeding)

- `GET /api/student/filenames`: Lista los archivos `.json` disponibles en la carpeta de semillas.
- `POST /api/student/seed/:filename`: Lee un archivo específico e inserta sus registros en la base de datos (acumulativo).

Vistas y Salida

La información entregada por la API incluye para cada registro:

- Año de cohorte
- Identificador de carrera y Nombre estándar
- Conteo de matriculados iniciales (Cohorte)
- Conteo de retenidos al año siguiente
- Tasa de retención (decimal).