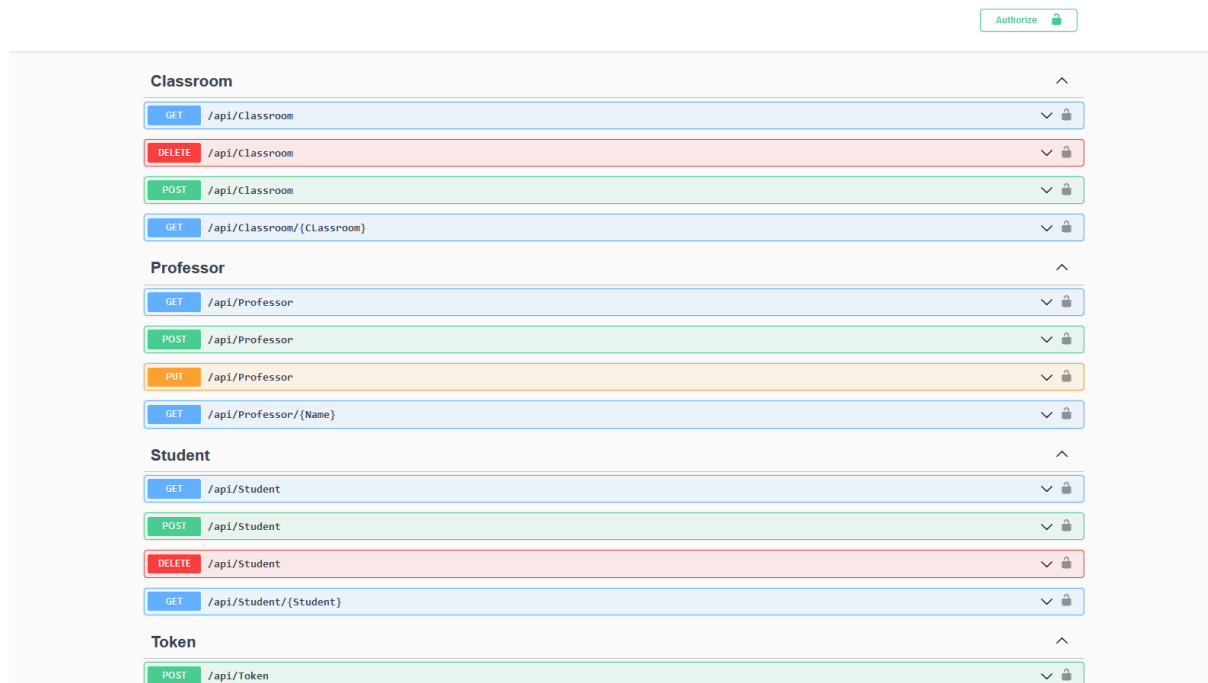


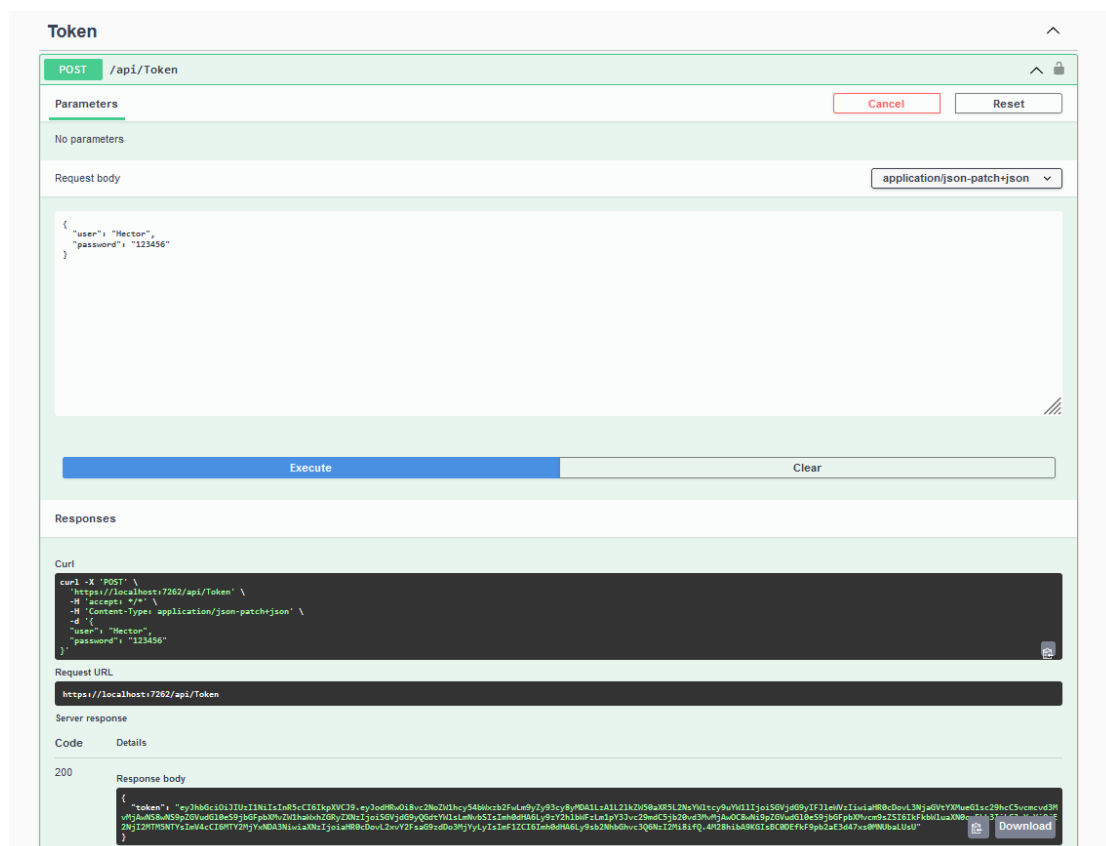
Documento técnico

Como Usarlo:

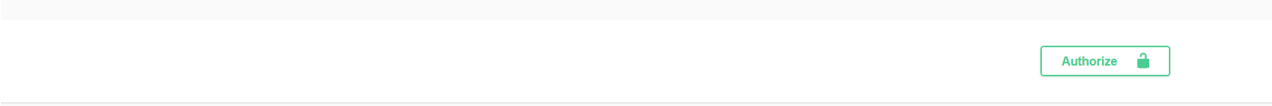
Al ejecutar el proyecto se abrirá el navegador donde se podrá encontrar con la IU de Swagger.



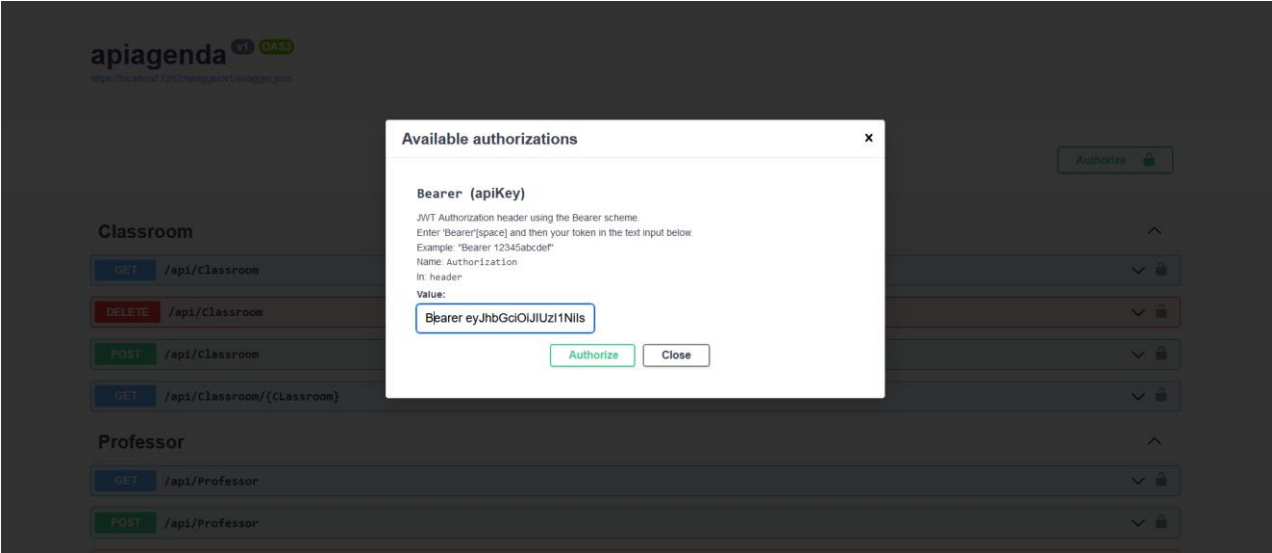
Para poder probar la funcionalidad de cada uno de los endpoints tendrá que ejecutar el endpoint de Token, el mismo solicitará un usuario y una contraseña, la introduce y devolverá un token.



Luego de obtener el Token, debe copiarlo, dar click en Authorize

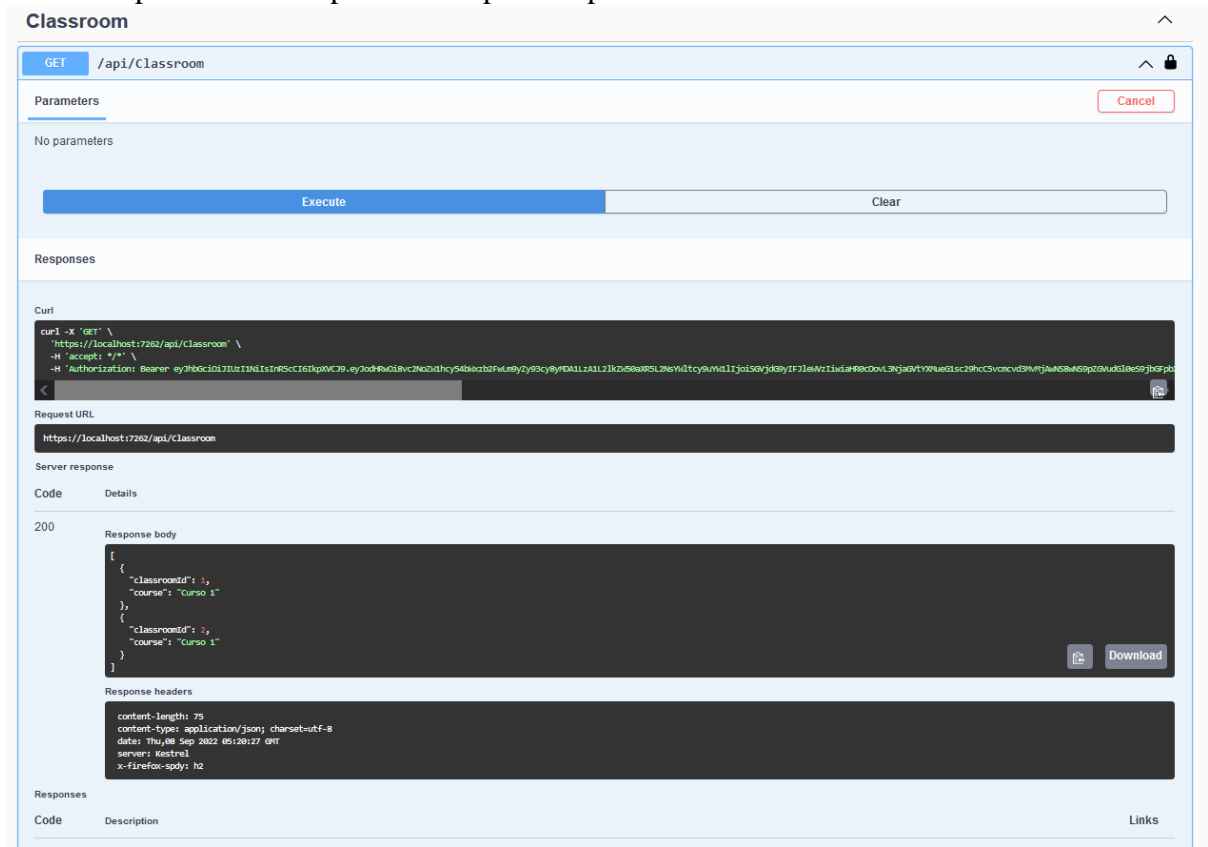


Saldrá un menú de dialogo donde debe escribir Bearer seguido de un espacio y luego pegue el token.

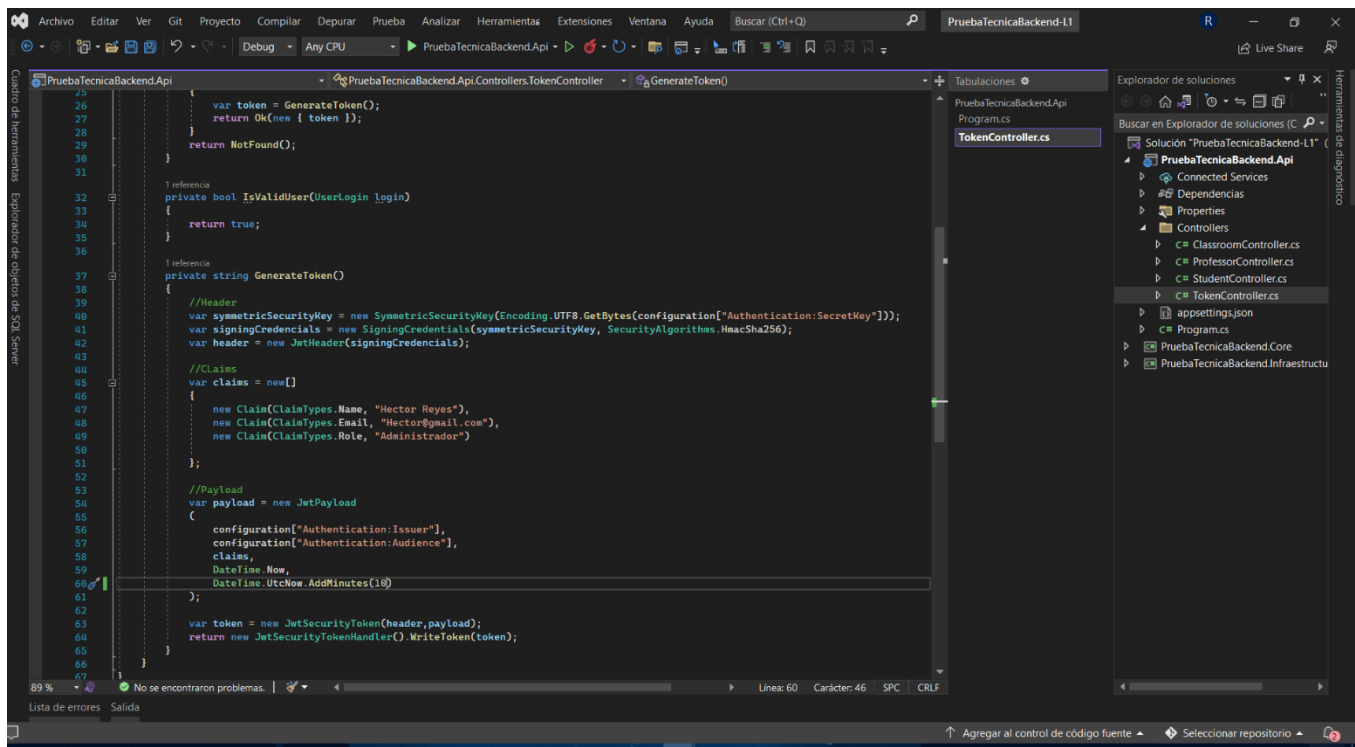


Dar click en Authorize y luego a Close.

Con estos pasos nuestro api estará disponible para utilizarse:



El token tiene un tiempo de caducidad de 10 minutos, si le quiere poner un tiempo mayor, vaya al controlador TokenController y en el método GenerateToken donde dice AddMinutes cambie el 10 por el numero deseado.



Funcionalidades:

Classroom

| Classroom | | | |
|-----------|----------------------------|--|--|
| GET | /api/Classroom | | |
| DELETE | /api/Classroom | | |
| POST | /api/Classroom | | |
| GET | /api/Classroom/{Classroom} | | |

- Get: obtiene todos los cursos
- Delete: borrar cursos por Id
- Post: crea cursos
- Get: busca cursos por curso

Professor

| Professor | | | |
|-----------|-----------------------|--|--|
| GET | /api/Professor | | |
| POST | /api/Professor | | |
| PUT | /api/Professor | | |
| GET | /api/Professor/{Name} | | |

Get: obtiene todos los profesores.

Post: crea profesores. No se puede insertar un profesor en el mismo curso mas de una vez, no puede dar clases en el mismo turno, puede dar clases en diferentes aulas con turnos diferentes, los profesores pueden dar clases en el mismo turno, pero en cursos diferentes.

Put: elimina los usuarios actualizando su campo States a activo y su campo DateDelete a la fecha actual al momento de actualizarlo.

Get: busca profesores por nombre

Student

| Student | | | ^ |
|---------|------------------------|---|---|
| GET | /api/Student | ✓ | 🔒 |
| POST | /api/Student | ✓ | 🔒 |
| DELETE | /api/Student | ✓ | 🔒 |
| GET | /api/Student/{Student} | ✓ | 🔒 |

Get: obtiene todos los estudiantes

Post: crea estudiantes. No se pueden crear mas de 30 estudiantes por aula

Delete: borrar estudiantes por Id

Get: busca estudiantes por nombre

Token

| Token | | | ^ |
|-------|------------|---|---|
| POST | /api/Token | ✓ | 🔒 |

Post: se inserta usuario y contraseña para obtener el token.

Herramientas Utilizadas

Visual Studio Community

SQL Server

Librerías Utilizadas:

AutoMapper: lo utilizo para mapear las entidades en DTO, para copiar un objeto en otro.

Newtonsoft.Json: la utilizo para resolver o mas bien ignorar las referencias circulares.

EntityFrameworkCore: convierte estructuras de datos en clases para poder trabajar con esa información usando objetos.