

# Evidencias Backend HORUS

## Swagger UI (Interfaz de pruebas)

### Capturas Swagger (Administrador)

**Admin**

GET

/api/admin/usuarios/search

Busca usuarios por nombre o email, filtrando por rol (Administrador, Profesor, Estudiante)

🔒

⌵

GET

/api/admin/usuarios/{idUserario}

Obtiene un usuario (Profesor o Estudiante) por ID.

🔒

⌵

POST

/api/admin/usuarios

Crea un nuevo usuario (Profesor/Estudiante) para la institución del Admin.

🔒

⌵

PUT

/api/admin/usuarios/{id}

Actualiza el nombre y correo de un usuario de la institución del Admin.

🔒

⌵

PUT

/api/admin/usuarios/{idUserario}/carrera

Actualiza la carrera principal, semestre y opcionalmente la sede de un estudiante.

🔒

⌵

POST

/api/admin/usuarios/{idUserario}/inscripciones

Inscribe a un estudiante en una o más secciones.

🔒

⌵

DELETE

/api/admin/usuarios/{idUserario}/inscripciones

Elimina la inscripción de un estudiante en una o más secciones (pasando IDs en query string).

🔒

⌵

GET

/api/admin/usuarios/{idUserario}

Obtiene un usuario (Profesor o Estudiante) por ID.

🔒

⌵

Parameters

Cancel

Name	Description
idUserario * required	
Integer(int32)	5
(path)	

Execute

Clear

Responses

Curl

curl -X 'GET' \n 'https://localhost:7289/api/admin/usuarios/5' \n -H 'accept: application/json'

Request URL

https://localhost:7289/api/admin/usuarios/5

Server response

Code

Details

200

Response body

```
{
  "idUserario": 5,
  "nombre": "Daniel Frias",
  "correo": "dani@duocur.cl",
  "rol": "Profesor",
  "nombreInstitucion": "DUOC UC",
  "tipoInstitucion": "Universidad",
  "carrera": null,
  "semestre": null,
  "asignaturas": [
    {
      "nombre": "Modelamiento de Base de Datos",
      "codigo": "MOD101",
      "seccion": "MOD101-0010",
      "dia": "Lunes",
      "horaInicio": "19:00",
      "horaFin": "20:20"
    },
    {
      "nombre": "Modelamiento de Base de Datos",
      "codigo": "MOD101",
      "seccion": "MOD101-0020",
      "dia": "Lunes",
      "horaInicio": "20:30",
      "horaFin": "21:50"
    }
  ]
}
```

Download

Response headers

## Capturas Swagger (Asignaturas)

Asignaturas			^
GET	/api/asignatura/list	Asignaturas	🔒
GET	/api/asignatura/carreras	Obtiene el listado completo de carreras con su escuela asociada.	🔒
PUT	/api/asignatura/{id}	Editar asignatura	🔒
GET	/api/asignatura/{idCurso}/secciones/detalle	Detalle de secciones por curso	🔒
GET	/api/asignatura/carreras/list	Listado de carreras	🔒
POST	/api/asignatura/Crear	Crea un curso y lo vincula a una carrera.	🔒
POST	/api/asignatura/cursos/{idCurso}/secciones	Crea una sección usando el código proporcionado por el frontend y asigna profesor/bloques/sala.	🔒
DELETE	/api/asignatura/secciones/{idSeccion}		🔒

Server response	
Code	Details
201	Response body
Unauthenticated	<pre>{  "message": "Curso creado correctamente",  "idCurso": 123,  "sigla": "PR0203"}</pre>

## Capturas Swagger (Catálogo)

Catalogo			^
GET	/api/catalog/sedes	Obtener todas las sedes	🔒
POST	/api/catalog/sedes		🔒
GET	/api/catalog/sedes/{id}	Obtener sede por ID	🔒
GET	/api/catalog/cursos/disponibles/{idUserario}/{semestre}		🔒
GET	/api/catalog/api/sedes/{idSede}/carreras	Obtener carreras en cada sede	🔒
GET	/api/catalog/api/carreras/{idCarrera}/sedes	Obtener sedes de una carrera	🔒
GET	/api/catalog/api/regiones/{idRegion}/sedes	Obtener sedes por región	🔒
GET	/api/catalog/oferta-academica/{sedeId}/{carreraId}	Obtener oferta académica de una sede y carrera	🔒
GET	/api/catalog/oferta-academica/{sedeId}	Obtener oferta académica por sede	🔒
GET	/api/catalog/api/catalog/filtros	Obtener filtros de oferta académica	🔒

GET

/api/catalog/sedes

Obtener todas las sedes

Devuelve los datos de todas las sedes.

Parameters

Cancel

No parameters

ExecuteClear

Responses

Curl

```
curl -X 'GET' \
'https://localhost:7289/api/catalog/sedes' \
-H 'accept: */*'

```

Request URL

```
https://localhost:7289/api/catalog/sedes

```

Server response

CodeDetails

200

Response body

```
{
  "idSede": 15,
  "nombre": "Chillán",
  "region": "Región de Ñuble",
  "entidad": "DUOC UC",
  "fechaInicio": "2026-01-16T00:00:00",
  "diurna": "08:00-18:00",
  "vespertina": "18:00-22:00",
  "disponibleEn": "Presencial",
  "imagenUrl": ""
},
{
  "idSede": 9,
  "nombre": "Concepción",
  "region": "Región del BíoBío",
  "entidad": "DUOC UC",
  "fechaInicio": "2026-01-16T00:00:00",
  "diurna": "08:00-18:00",
  "vespertina": "18:00-22:00",
  "disponibleEn": "Presencial",
  "imagenUrl": ""
},
{
  "idSede": 13,
  "nombre": "La Serena",
  "region": "Región de Coquimbo",
  "entidad": "DUOC UC",

```

Download

## Capturas Swagger (Autenticación)

Autenticación

GET

/api/auth/usuarios

Obtener todos los usuarios

POST

/api/auth/login

Obtener datos y generar Token

GET

/api/auth/me

Permite probar el token

# Swagger URL

## Rutas probadas (Swagger/OpenAPI)

**Base:** http://localhost:7289 · **Docs:** /swagger

**Auth:** Bearer JWT (roles: Admin/Profesor/Estudiante).

**https://localhost:7289/api/auth/login**

**https://localhost:7289/api/catalog/sedes**

**https://localhost:7289/api/catalog/api/carreras/1/sedes**

**https://localhost:7289/api/catalog/oferta-academica/1**

**https://localhost:7289/api/catalog/api/sedes/1/carreras**

**https://localhost:7289/api/asignatura/list**

**https://localhost:7289/api/admin/usuarios/search?q=Cristobal&NombreRol=Estudiante**

**https://localhost:7289/api/admin/usuarios/1**

**https://localhost:7289/api/asignatura/Crear**

**https://localhost:7289/api/asignatura/carreras/list**

## Qué valida cada ejemplo

- **/api/catalog/sedes** → catálogo público (GET).
- **/api/catalog/api/carreras/1/sedes** → relación sede–carrera (GET con path param).
- **/api/catalog/api/sedes/1/carreras** → relación inversa (GET).
- **/api/asignatura/list** → listado paginado/filtrado (GET).
- **/api/admin/usuarios/search?q=Cristobal&NombreRol=Estudiante** → búsqueda + query params (GET, rol Admin).
- **/api/admin/usuarios/1** → detalle por ID (GET, rol Admin).
- **/api/asignatura/Crear** → creación (POST, body JSON, rol autorizado).
- **/api/asignatura/carreras/list** → listado asociado (GET).