

[Repo](#)

Integrantes:

Ares Ortiz Botello A01747848

Arturo Montes González A01798012

Videojuego (Modelo relacional)

1. Post para crear videojuego

The terminal on the left shows the execution of a Node.js application. It starts by running 'npm run start', which launches a server on port 8085. The application then performs a database migration, creating a table named 'VideoJuego' in a database named 'steam'. It then inserts two records: one for 'Cocoon' with a price of 340 and genre 'god-like', and another for 'Roblox Giga Mansion' with a price of 3000000 and genre 'El mejor'. Finally, it queries the database to show the created records.

The REST client on the right shows a POST request to the endpoint '52.91.158.214:8085/videojuego/crear'. The request body is a JSON object:

```
{  "nombre": "Roblox Giga Mansion",  "precio": 3000000,  "genero": "El mejor"}
```

. The response is a 200 OK status with a body of 'Videojuego creado'.

2. Get para obtener videojuegos

The terminal on the left shows the same application running. It performs the same database migration and insert operations as in the first screenshot. Then, it queries the database to retrieve all records from the 'VideoJuego' table. The output shows the details of the three games: 'Cocoon', 'Roblox Giga Mansion', and 'El mejor'.

The REST client on the right shows a GET request to the endpoint '52.91.158.214:8085/videojuego/consultar'. The response is a 200 OK status with a body containing a JSON array of three game objects:

```
[  {    "id": 1,    "nombre": "Hades",    "precio": 340,    "genero": "Rogue-like"  },  {    "id": 2,    "nombre": "Stardew",    "precio": 340,    "genero": "kalsdjf-like"  },  {    "id": 3,    "nombre": "Cocoon",    "precio": 340,    "genero": "god-like"  },  {    "id": 4,    "nombre": "Roblox Giga Mansion",    "precio": 3000000,    "genero": "El mejor"  }]
```

Mascota (Modelo no relacional)

1. Post para crear mascota

The screenshot shows two side-by-side windows. The left window is a terminal running an AWS CLI command to start a server. The right window is a Postman client showing a successful POST request to the endpoint `52.91.158.214:8085/mascota/crear`.

Terminal Output:

```
[ec2-user@ip-172-31-18-232 pipelineTS]$ npm run start
> pipeline-ts@1.0.0 start
> node -r dotenv/config dist/index.js

VideoJuegoModel.js
VideoJuego
Server running on port 8085
(node:5197) NOTE: The AWS SDK for JavaScript (v2) will enter maintenance mode on September 8, 2024 and reach end-of-support on September 8, 2025.

Please migrate your code to use AWS SDK for JavaScript (v3).
For more information, check blog post at https://a.co/cUPnyil
(Use 'node --trace-warnings ...' to show where the warning was created)
Executing (default): SELECT TABLE_NAME FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_TYPE = 'BASE TABLE' AND TABLE_NAME = 'VideoJuego' AND TABLE_SCHEMA = 'steam'
Executing (default): SHOW INDEX FROM 'VideoJuego'
Tablas creadas
{ nombre: 'Pepillo', edad: 8, raza: 'Golden' }
Mascota creada
```

Postman Request:

- Method: POST
- URL: `52.91.158.214:8085/mascota/crear`
- Body (JSON):

```
{  "nombre": "Pepillo",  "edad": 8,  "raza": "Golden"}
```
- Status: 200 OK
- Response Body: `Mascota creada`

2. Get para obtener mascotas

The screenshot shows two side-by-side windows. The left window is a terminal running an AWS CLI command to start a server. The right window is a Postman client showing a successful GET request to the endpoint `52.91.158.214:8085/mascota/consultar`.

Terminal Output:

```
[ec2-user@ip-172-31-18-232 pipelineTS]$ npm run start
> pipeline-ts@1.0.0 start
> node -r dotenv/config dist/index.js

VideoJuegoModel.js
VideoJuego
Server running on port 8085
(node:5197) NOTE: The AWS SDK for JavaScript (v2) will enter maintenance mode on September 8, 2024 and reach end-of-support on September 8, 2025.

Please migrate your code to use AWS SDK for JavaScript (v3).
For more information, check blog post at https://a.co/cUPnyil
(Use 'node --trace-warnings ...' to show where the warning was created)
Executing (default): SELECT TABLE_NAME FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_TYPE = 'BASE TABLE' AND TABLE_NAME = 'VideoJuego' AND TABLE_SCHEMA = 'steam'
Executing (default): SHOW INDEX FROM 'VideoJuego'
Tablas creadas
{ nombre: 'Pepillo', edad: 8, raza: 'Golden' }
Mascota creada

{  Count: 4,  ScannedCount: 4,  Items: [ [Model], [Model], [Model], [Model] ] }
```

Postman Request:

- Method: GET
- URL: `52.91.158.214:8085/mascota/consultar`
- Status: 200 OK
- Response Body (JSON):

```
[  {    "nombre": "Pepillo",    "raza": "Golden",    "MascotaID": "98cb1144-4587-4146-ae95-967d3ebcc94f",    "edad": 8  },  {    "nombre": "Jack",    "raza": "Muskyy",    "MascotaID": "c0419821-e172-410a-89aa-0c5a232e4c2f",    "edad": 10  },  {    "nombre": "Bonny",    "raza": "Cocker",    "MascotaID": "1cf1306e-b681-4d5f-9f91-2b141388bed0",    "edad": 5  }]
```