

Lab7 GTICS - 20191450

NOTA: He cambiado el nombre del atributo playerId por player_id ya que me daba problemas para actualizar un jugador, estoy adjuntando la bd con el nombre del atributo cambiado.

Consumo API REST para listar leaderboard

URL	Method
http://localhost:8080/dota/jugadores/leaderboard	GET

1. Argumentos de entrada:

Ninguno

2. Argumentos de salida:

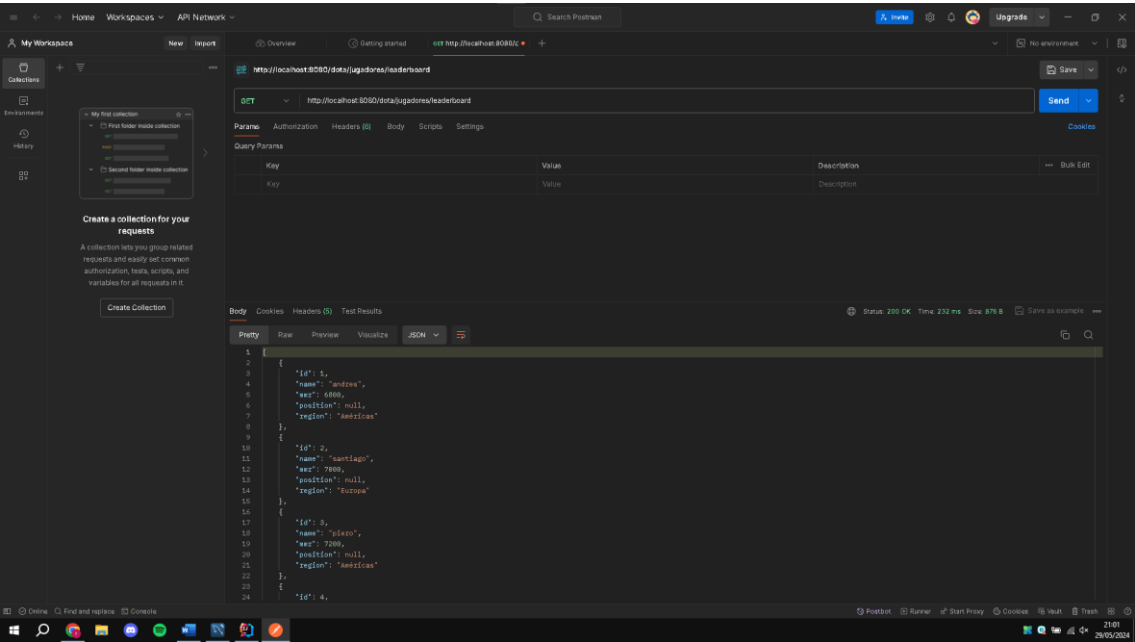
```
[
  {
    "id": 1,
    "name": "andres",
    "mmr": 6800,
    "position": null,
    "region": "Américas"
  },
  {
    "id": 2,
    "name": "santiago",
    "mmr": 7000,
    "position": null,
    "region": "Europa"
  },
  {
    "id": 3,
    "name": "piro",
    "mmr": 7200,
    "position": null,
    "region": "Américas"
  },
]
```

```
{
  "id": 4,
  "name": "juan",
  "mmr": 6700,
  "position": null,
  "region": "China"
},
{
  "id": 5,
  "name": "manuel",
  "mmr": 6550,
  "position": null,
  "region": "Europa"
},
{
  "id": 7,
  "name": "paul",
  "mmr": 7100,
  "position": null,
  "region": "Américas"
},
{
  "id": 9,
  "name": "pablo",
  "mmr": 8000,
  "position": null,
  "region": "Américas"
},
{
  "id": 10,
  "name": "paco",
  "mmr": 7500,
  "position": null,
  "region": "Europa"
},
{
  "id": 11,
  "name": "shisha",
  "mmr": 8200,
  "position": null,
  "region": "SE Asiático"
},
{
  "id": 16,
  "name": "aoki",
  "mmr": 8003,
  "position": null,
  "region": "China"
}
]
```

- Datos:

Campo	Tipo de Dato	Observaciones
playerId	entero	-
name	cadena	-
mmr	gran entero	-
position	entero	-
region	cadena	-

Ejemplo:



Consumo API REST para crear o actualizar un jugador

URL	Method
http://localhost:8080/dota/jugadores/save	POST
http://localhost:8080/dota/jugadores/save	PUT

1. Argumentos de entrada:

Opción 1: Metodo POST

- **Body:**

```
{
  "name": "jorge",
  "mmr": 6500,
  "region": "Américas"
}
```

Opción 2: Metodo PUT

- **Body:**

```
{
  "id": 17,
  "name": "jorge",
  "mmr": 7100,
  "region": "Américas"
}
```

2. Datos de argumentos de entrada:

POST:

Campo	Tipo de Dato	Observaciones
name	cadena	-
mmr	gran entero	-
region	cadena	-

PUT:

Campo	Tipo de Dato	Observaciones
PlayerId	entero	-
name	cadena	-
mmr	gran entero	-
region	cadena	-

3. Argumentos de salida:

Opción 1: Solicitud correcta – Metodo POST

- **Http Status:** HTTP 201 Created
- **Body:**

```
{
  "jugador agregado con exito": {
    "id": 17,
    "name": "jorge",
    "mmr": 6500,
    "position": 0,
    "region": "Américas"
  }
}
```

Opción 2: Solicitud correcta – Metodo PUT

- **Http Status:** HTTP 201 Created
- **Body:**

```
{
  "jugador actualizado con exito": {
    "id": 17,
    "name": "jorge",
    "mmr": 7100,
    "position": 0,
    "region": "Américas"
  }
}
```

Opción 3: Error – Metodo POST y PUT

- **Http Status:** HTTP 400 Bad Request
- **Body:**

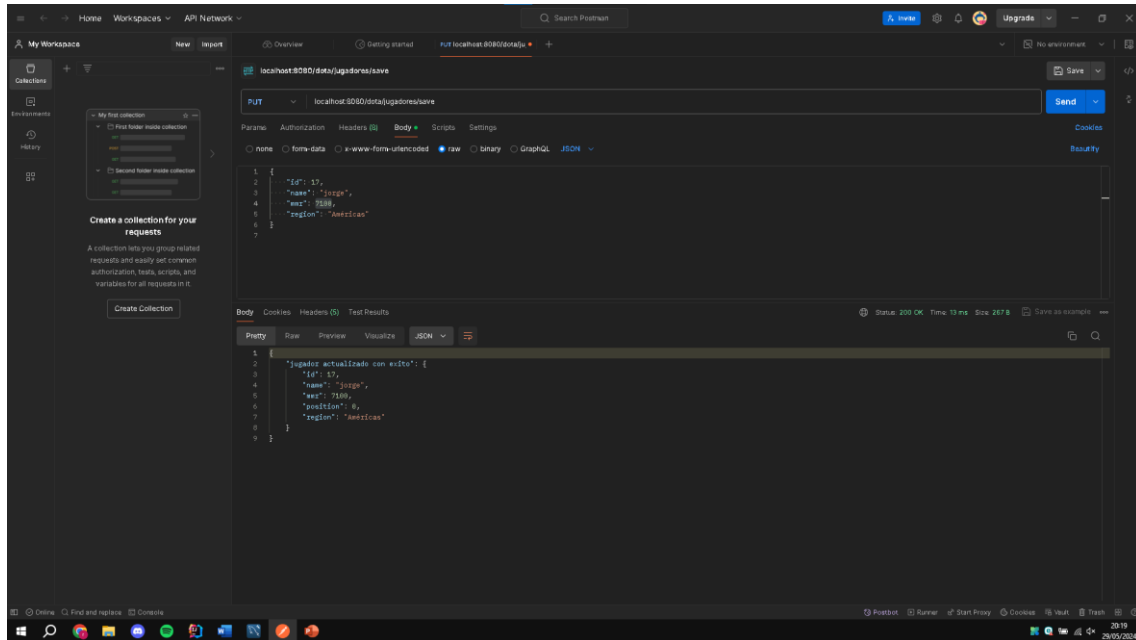
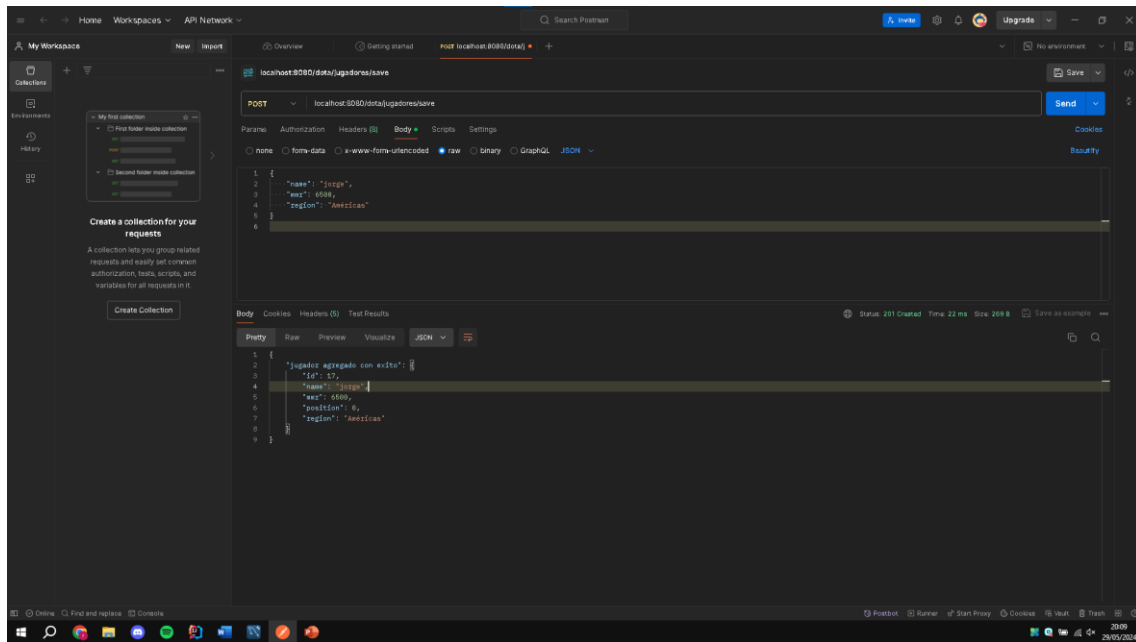
```
{  
  "date": "2024-05-25T17:42:34.253733900",  
  "error": "Error en validación de Datos."  
}
```

4. Datos argumentos de salida

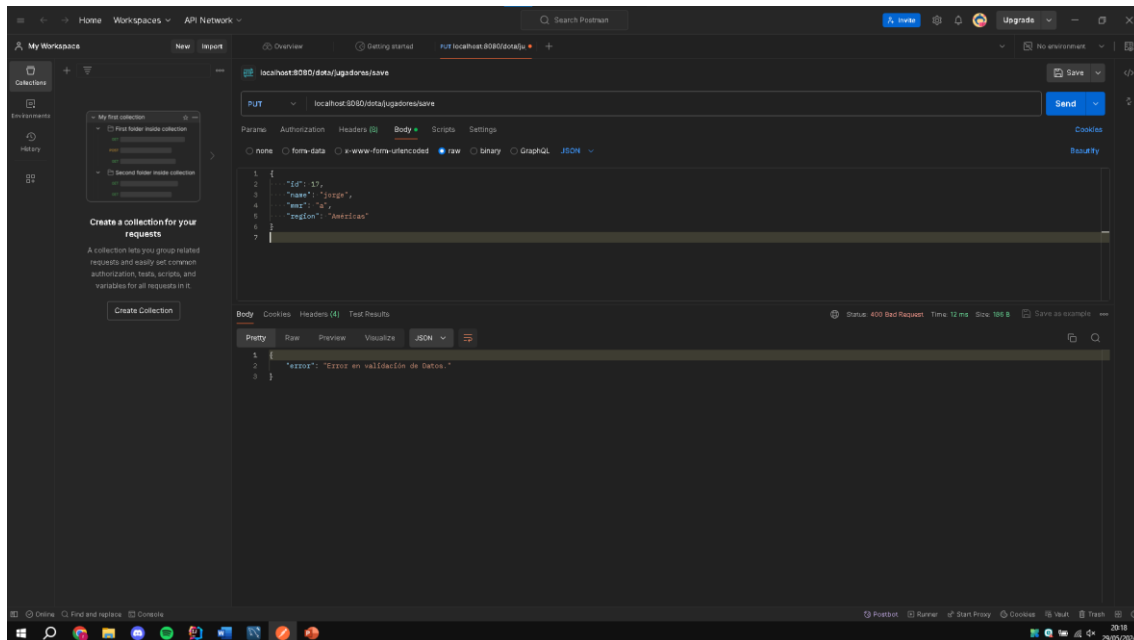
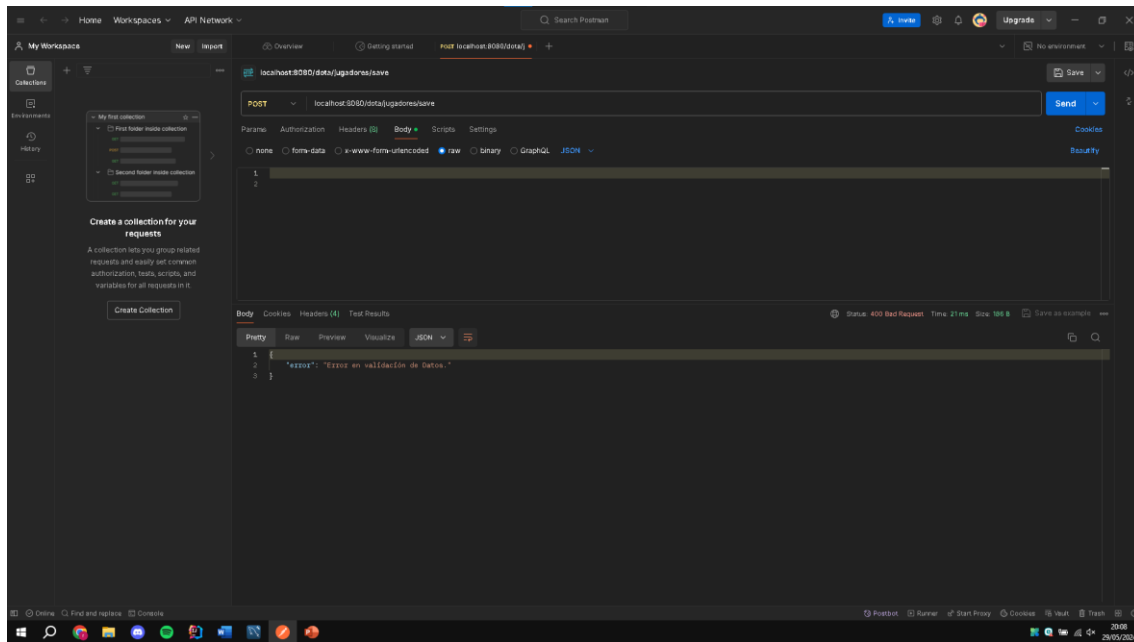
Campo	Tipo de Dato	Observaciones
playerId	entero	-
name	cadena	-
mmr	gran entero	-
position	entero	-
region	cadena	-

5. Ejemplos

Solicitud correcta:



Solicitud incorrecta:



Consumo API REST para eliminar un jugador

URL	Method
<code>http://localhost:8080/dota/jugadores/delete/{id}</code>	DELETE

6. Argumentos de entrada:

Campo	Tipo de Dato	Observaciones
playerId	entero	-

7. Argumentos de salida:

Opción 1: Solicitud correcta

- **Http Status:** HTTP 200 OK
- **Body:**

```
{
  "delete": "jugador eliminado con exito."
}
```

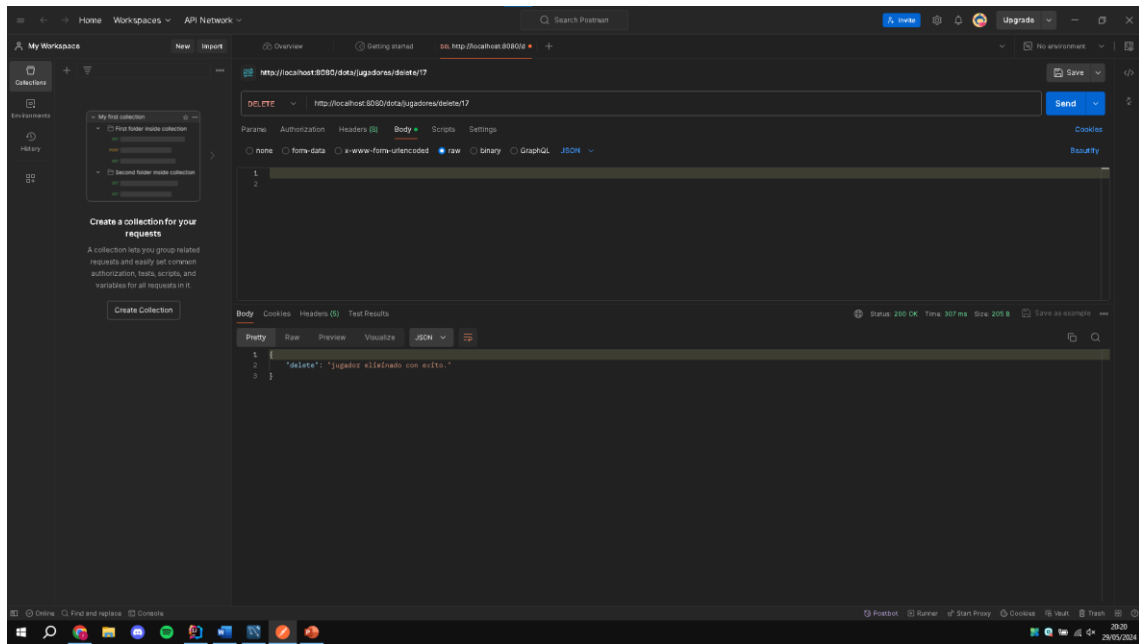
Opción 2: ID ingresado no esta en la lista de ID's existentes

- **Http Status:** HTTP 400 NOT FOUND
- **Body:**

```
{
  "error": "ID de jugador no encontrado."
}
```

8. Ejemplos

Solicitud correcta:



Solicitud incorrecta:

