



**UNIVERSIDAD
FRANCISCO GAVIDIA**
Tecnología, Innovación y Calidad

Facultad: Ingeniería y Sistemas

Módulo: Desarrollo de aplicaciones web

Docente: Carlos Boris Martinez Calzadia

Alumno: Manuel Alberto Serramo Reyes

Actividad: Laboratorio 03

Fecha de Entrega: 28 de octubre de 2023

Laboratorio 03

- 1- Se crea la base de datos

- ▼  Databases (2)

- mediapp15

- 2- Luego se crearon los modelos correspondientes de jugador, Posición, Camisa.

The screenshot displays an IDE with a Java file named `JugadorDTO.java` and a console window at the bottom.

Java Code:

```
1 package com.mitocode.model;
2
3 import java.util.*;
4
5 // no usages
6 @Entity
7 @Getter
8 @Setter
9 @AllArgsConstructor
10 @NoArgsConstructor
11 public class Jugador {
12     // no usages
13     @Id
14     @GeneratedValue(strategy = GenerationType.IDENTITY)
15     private Integer id;
16     // no usages
17     private String nombre;
18     // no usages
19     @OneToMany(cascade = CascadeType.MERGE)
20     @JoinColumn(name = "camisa_id", referencedColumnName = "id")
21     private Camisa camisa;
22     // no usages
23     @ManyToMany
24     @JoinColumn(name = "posicion_id", referencedColumnName = "id")
25     private Posicion posicion;
26 }
```

Console Output:

```
main] o.s.b.w.embedded.tomcat.TomcatWebServer : Will secure org.springframework.security.oauth2.config.annotation.web.configuration.ResourceServerConfiguration$NotOAuth2
main] o.s.b.w.DefaultSecurityFilterChain : Will secure any request with [org.springframework.security.web.session.DisableEncodeurFilter@8b94430e, org.springframework
main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 8080 (http) with context path ''
main] com.mitocode.MediappBackendApplication : Started MediappBackendApplication in 6.52 seconds (JVM running for 7.2s)
```

The console output shows the application starting successfully on port 8080. The status bar at the bottom indicates the build is successful and dependencies are resolved.

```
ExamDTO.java MedicoDTO.java AgendadorDTO.java MedappBackendApplication.java ResourceServerConfig.java ExamController.java JugadorRepos.java Jugador.java Posicion.java

package com.mitocode.model;

import lombok.AllArgsConstructor;
import lombok.Getter;
import lombok.NoArgsConstructor;
import lombok.Setter;

import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;

9 usages
@Entity
@Getter
@Setter
@AllArgsConstructor
@NoArgsConstructor
public class Posicion {

    no usages
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Integer id;
    no usages
    private String nombre;
}

s.s.web.DefaultSecurityFilterChain : Will secure URI [Ant [pattern= /oauth/token ], Ant [pattern= /oauth/token_key ], Ant [pattern= /oauth/check_token ]] with [
main] o.s.web.DefaultSecurityFilterChain : Will secure org.springframework.security.oauth2.config.annotation.web.configuration.ResourceServerConfiguration$NotOAuthRe
main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 8080 (http) with context path ''
main] com.mitocode.MediappBackendApplication : Started MediappBackendApplication in 6.52 seconds (JVM running for 7.261)
```

```
package com.mitocode.model;

import lombok.AllArgsConstructor;
import lombok.Getter;
import lombok.NoArgsConstructor;
import lombok.Setter;

import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;

9 usages
@Entity
@Getter
@Setter
@AllArgsConstructor
@NoArgsConstructor
public class Camisa {

    no usages
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Integer id;
    no usages
    private Integer numero;
    no usages
    private String color;
}

s.s.web.DefaultSecurityFilterChain : Will secure URI [Ant [pattern= /oauth/token ], Ant [pattern= /oauth/token_key ], Ant [pattern= /oauth/check_token ]] with [
s.s.web.DefaultSecurityFilterChain : Will secure org.springframework.security.oauth2.config.annotation.web.configuration.ResourceServerConfiguration$NotOAuthRe
s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 8080 (http) with context path ''
m.mitocode.MediappBackendApplication : Started MediappBackendApplication in 6.52 seconds (JVM running for 7.261)
```

```
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;
import org.springframework.web.servlet.support.ServletUriComponentsBuilder;

import javax.validation.Valid;
import java.util.List;
import java.util.stream.Collectors;

no usages
@RestController
@RequestMapping("/api/jugadores")
public class JugadorController {

    2 usages
    @Autowired
    private IJugadorRepo jugadorRepository;

    2 usages
    @Autowired
    private ModelMapper mapper;

    @GetMapping
    public ResponseEntity<List<JugadorDTO>> findAll() {
        List<JugadorDTO> list = jugadorRepository.findAll().stream().map(p -> mapper.map(p, JugadorDTO.class)).collect(Collectors.toList());
        return new ResponseEntity<>(list, HttpStatus.OK);
    }

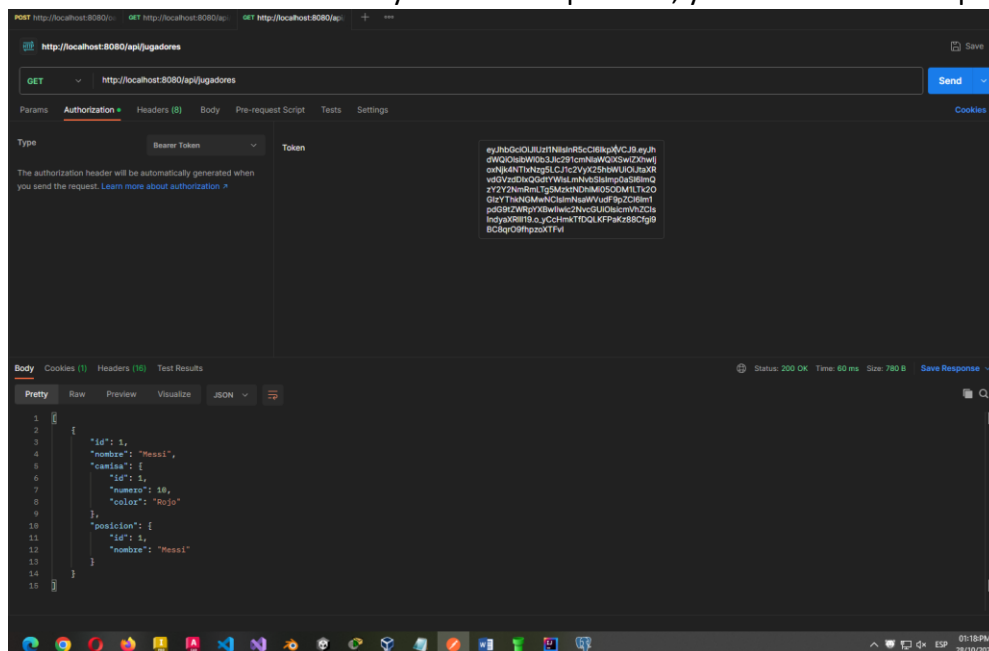
    @PostMapping
    public ResponseEntity<Void> save(@Valid @RequestBody JugadorDTO dto) {
        Jugador p = jugadorRepository.save(mapper.map(dto, Jugador.class));
        //localhost:8080/medics/3
        URI location = ServletUriComponentsBuilder.fromCurrentRequest().path("/{id}").buildAndExpand(p.getId()).toUri();
        return ResponseEntity.created(location).build();
    }
}
```

[illegible]

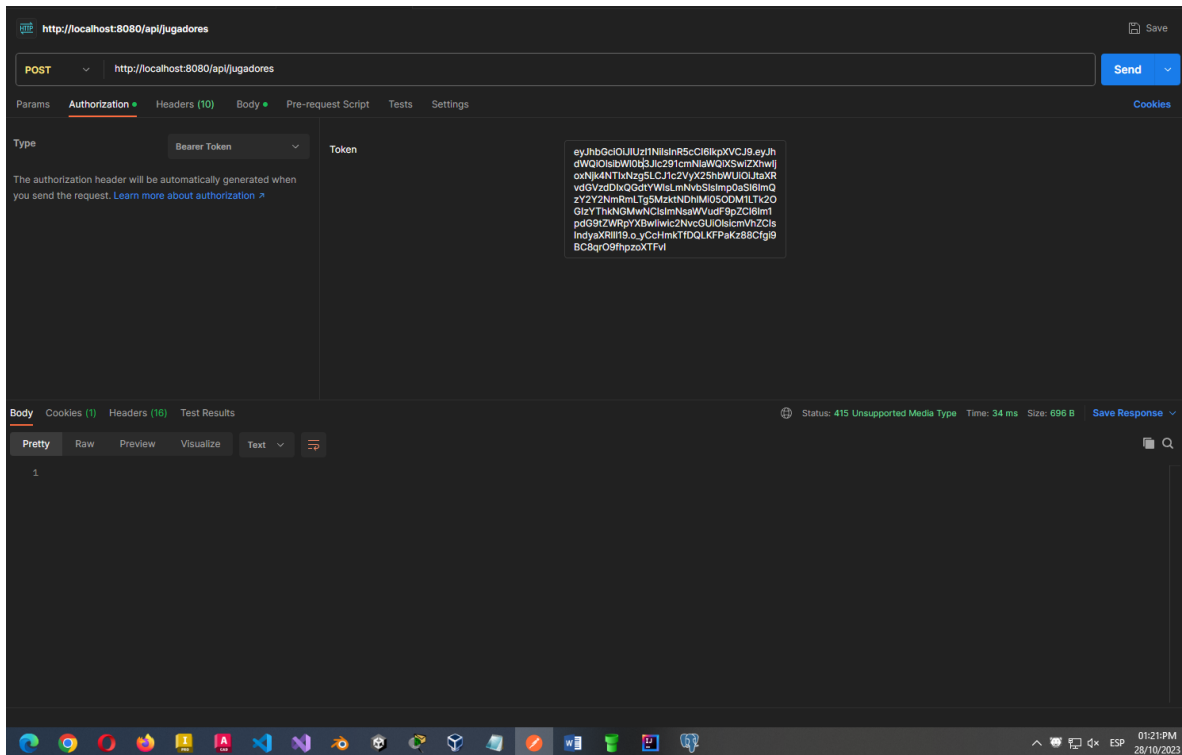
client-secret estos:

[illegible]

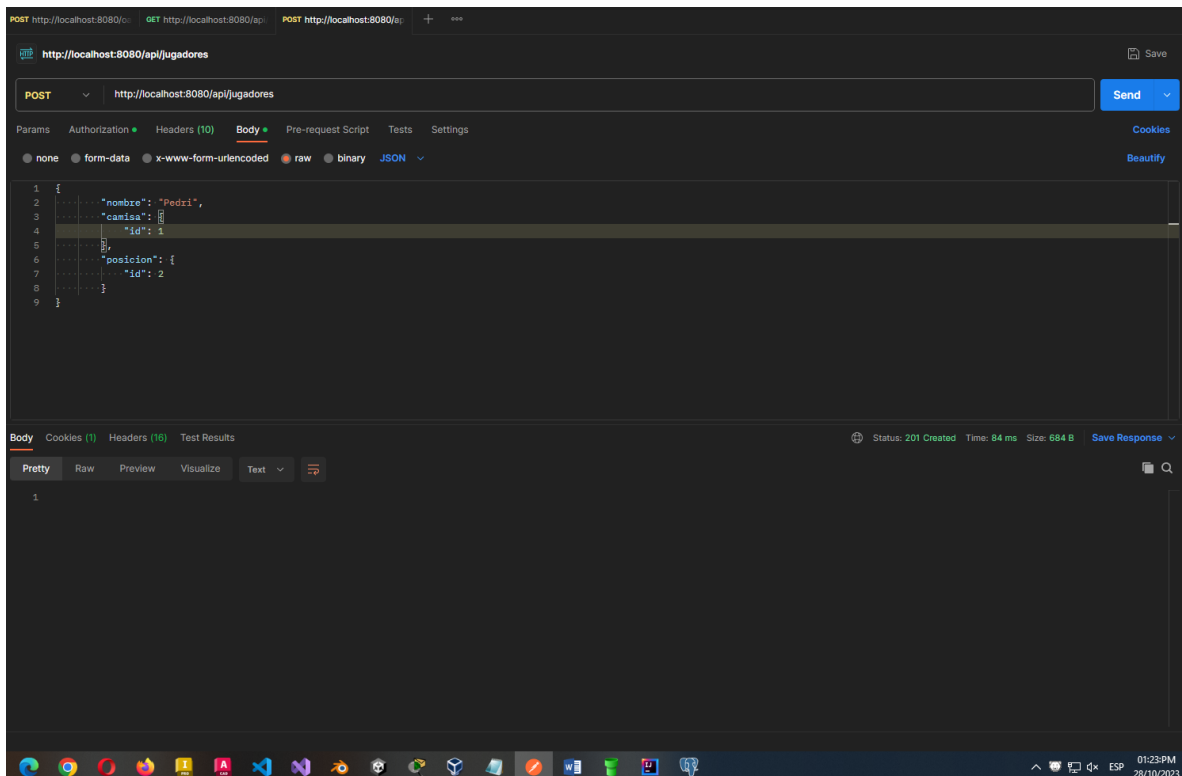
Luego de esto para probar las api, tomamos ese token secret en autorización seleccionamos token bearer y hacemos la petición, y de esta manera responde



Al probar con el post hacemos lo mismo autorización y agregamos el token realizamos la petición



Agregamos el json necesario para realizar el post



Y como se puede ver se realizo de forma exitosa el registro

The screenshot shows a PostgreSQL client interface with the following components:

- Top Bar:** Displays the connection name 'mediapp15/postgres@PostgreSQL 15' and various tool icons.
- Query Editor:** Contains the SQL query:

```
1 SELECT id, nombre, camisa_id, posicion_id
2 FROM public.jugador;
```
- Data Output Panel:** Shows the results of the query in a table format:

	id [PK] integer	nombre character varying (255)	camisa_id integer	posicion_id integer
1	1	Messi	1	1
2	2	Pedri	1	2
- Status Bar:** Indicates 'Total rows: 2 of 2' and 'Query complete 00:00:00.052'. The cursor is at 'Ln 2, C'.
- Taskbar:** Shows the Windows taskbar at the bottom with various application icons and the system clock displaying '01:22:PM 28/10/2023'.