



Educación de **CLASE MUNDIAL** ahora en El Salvador

Estudiante:

Michael Fernando Torres Callejas,
TC100220.

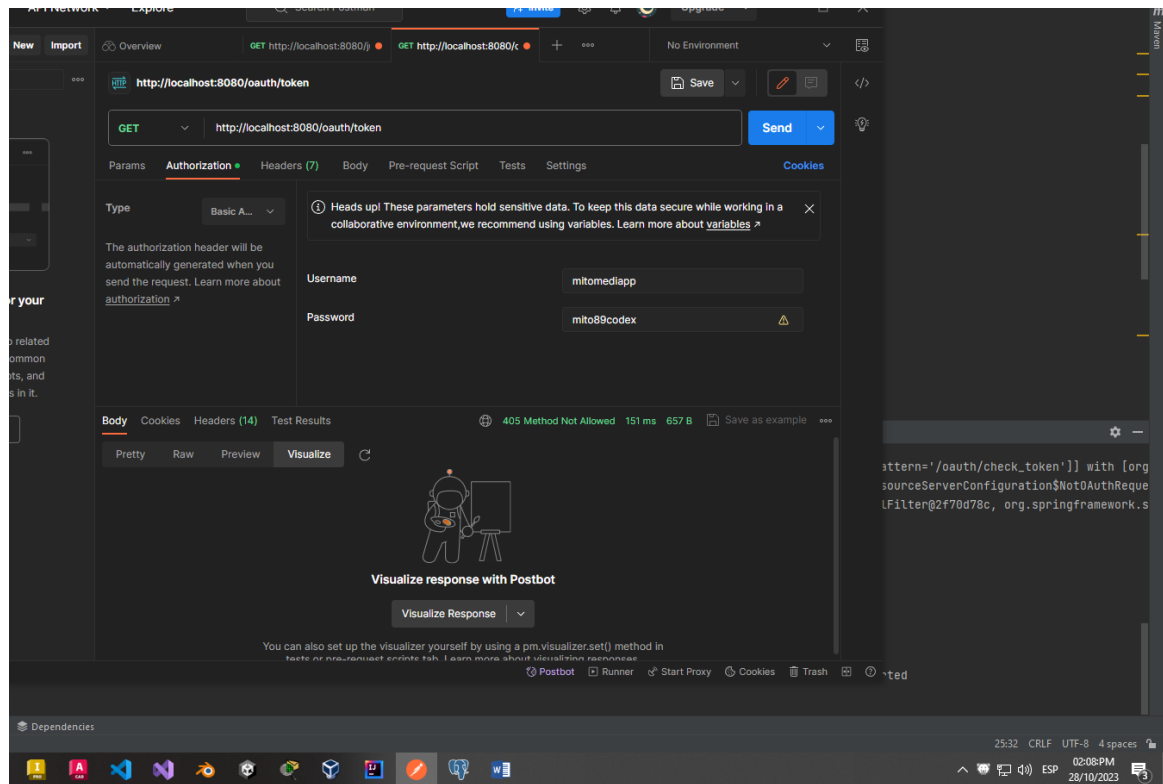
Docente: Carlos Boris Martinez Calzadia.

Actividad: LABORATORIO 03.

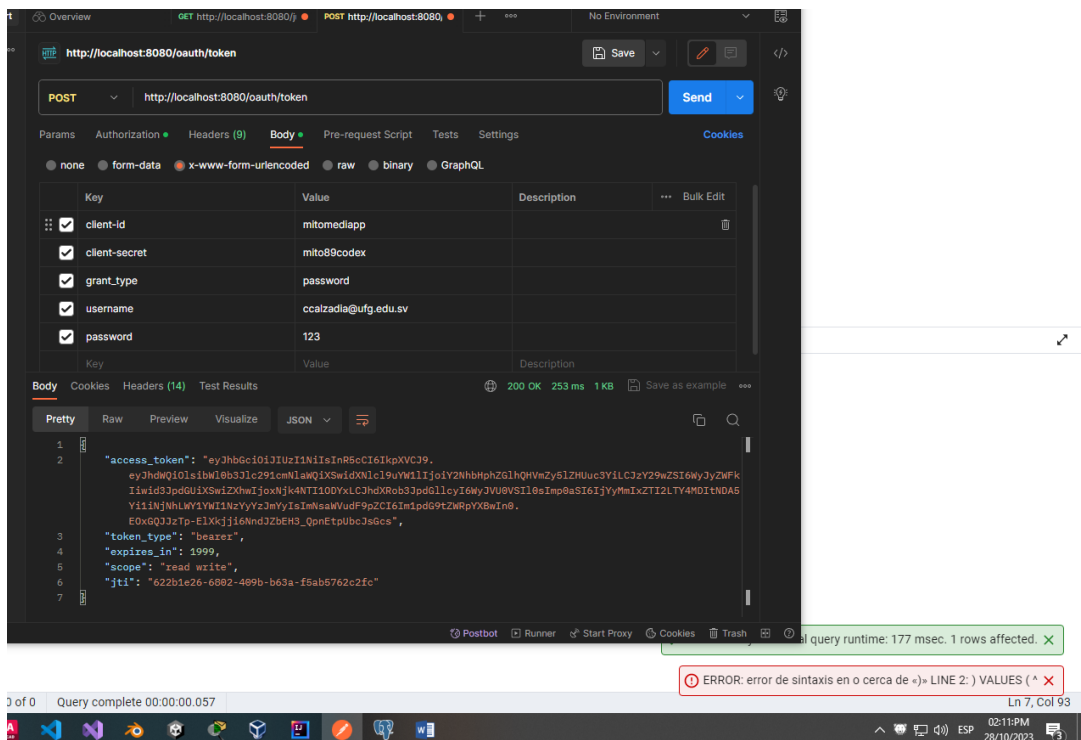
Modulo: Desarrollo de Aplicaciones Web 01

Fecha: 28/10/2023

1. Uso las credenciales correctas:



2. Obtengo el token.



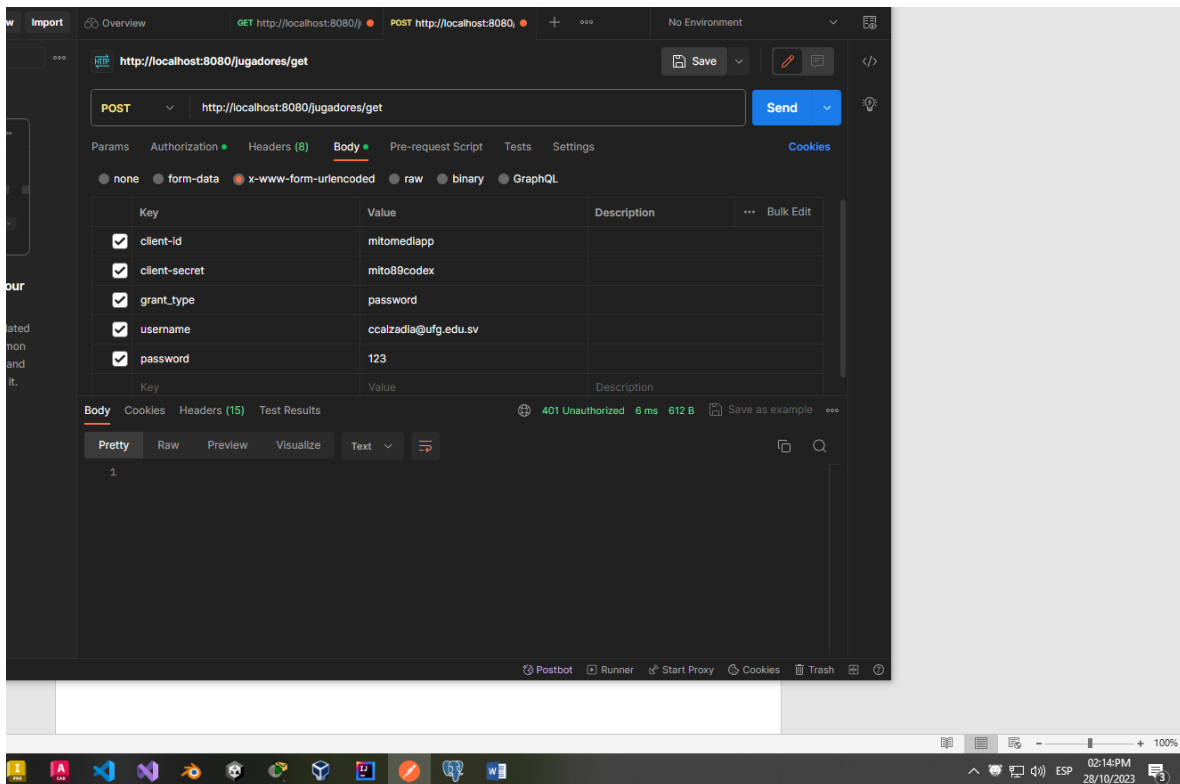
```
41 INSERT INTO menu_role (id_menu, id_role) VALUES (0, 4);
42
43 INSERT INTO jugador (numero_camisa) VALUES (1);
44 INSERT INTO jugador (nombre, posicion, id_camisa) VALUES (1,1,1,1);
45
46 select m.* from menu_role mr
47 inner join user_role ur on ur.id_role = mr.id_role
48 inner join menu m on m.id_menu = mr.id_menu
49 inner join user_data u on u.id_user = ur.id_user
50 where u.username = 'mitocode21@gmail.com';
```

Data Output Messages Notifications

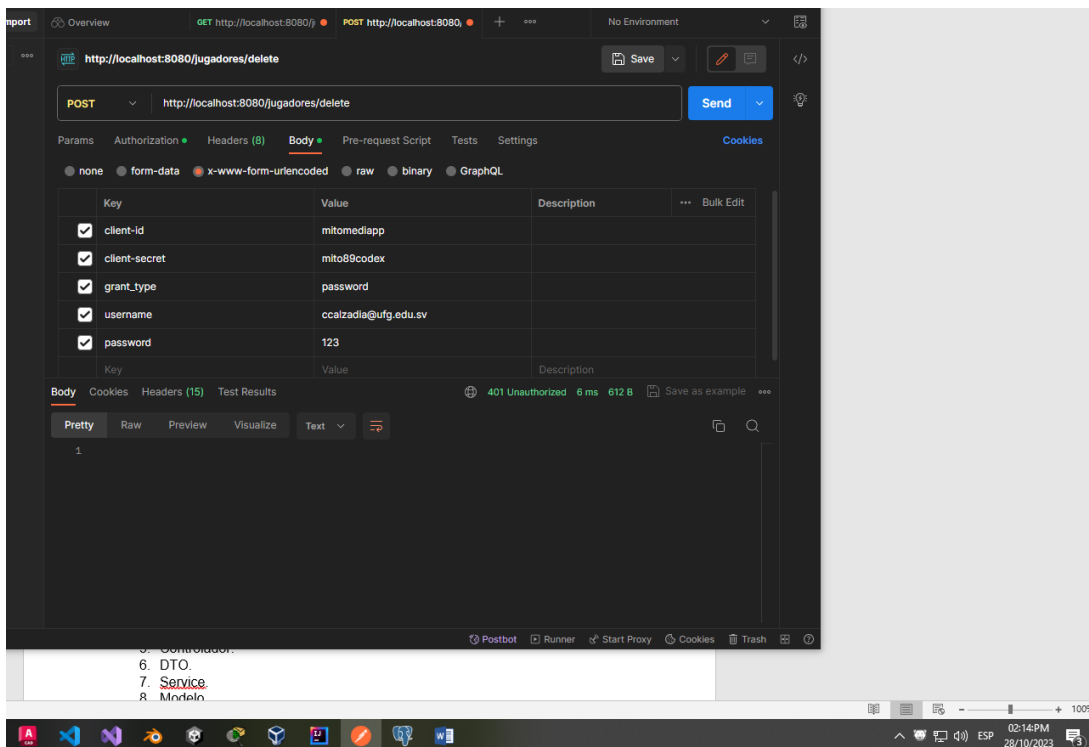
ERROR: no existe la columna «numero_camisa» en la relación «jugador»
LINE 1: INSERT INTO jugador (numero_camisa) VALUES (1);
^

SQL state: 42703
Character: 22

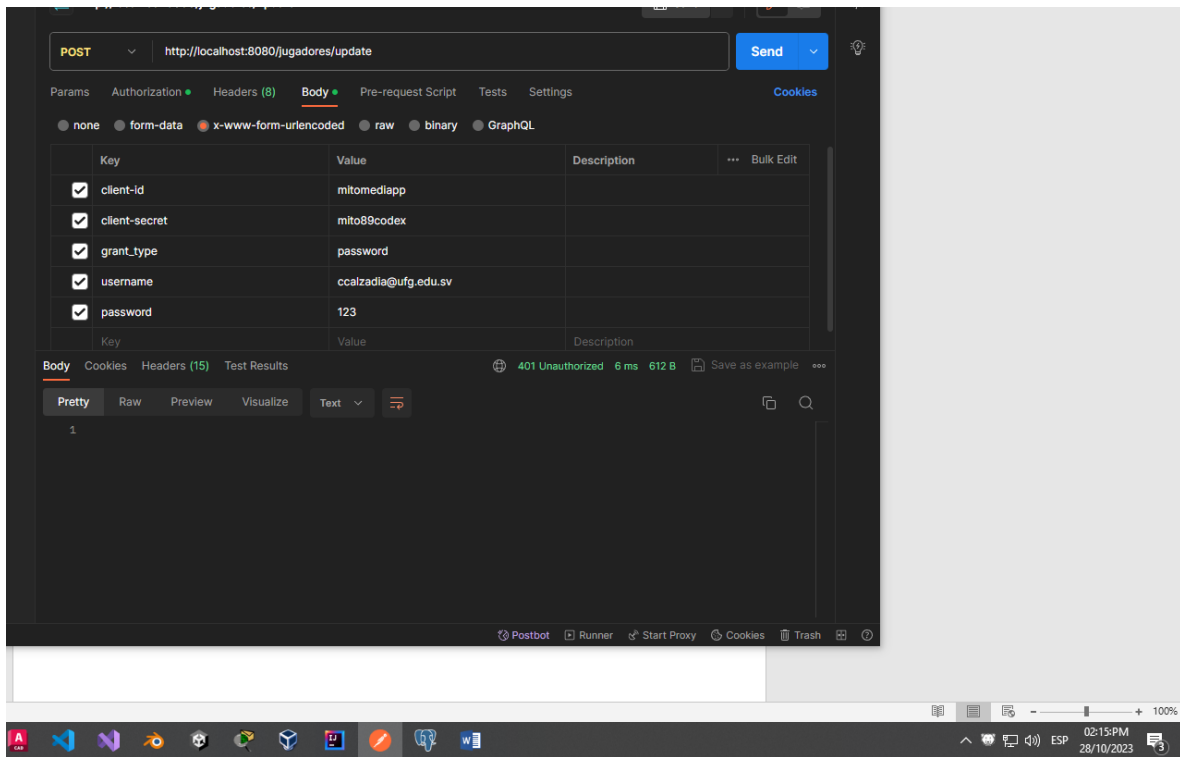
READ



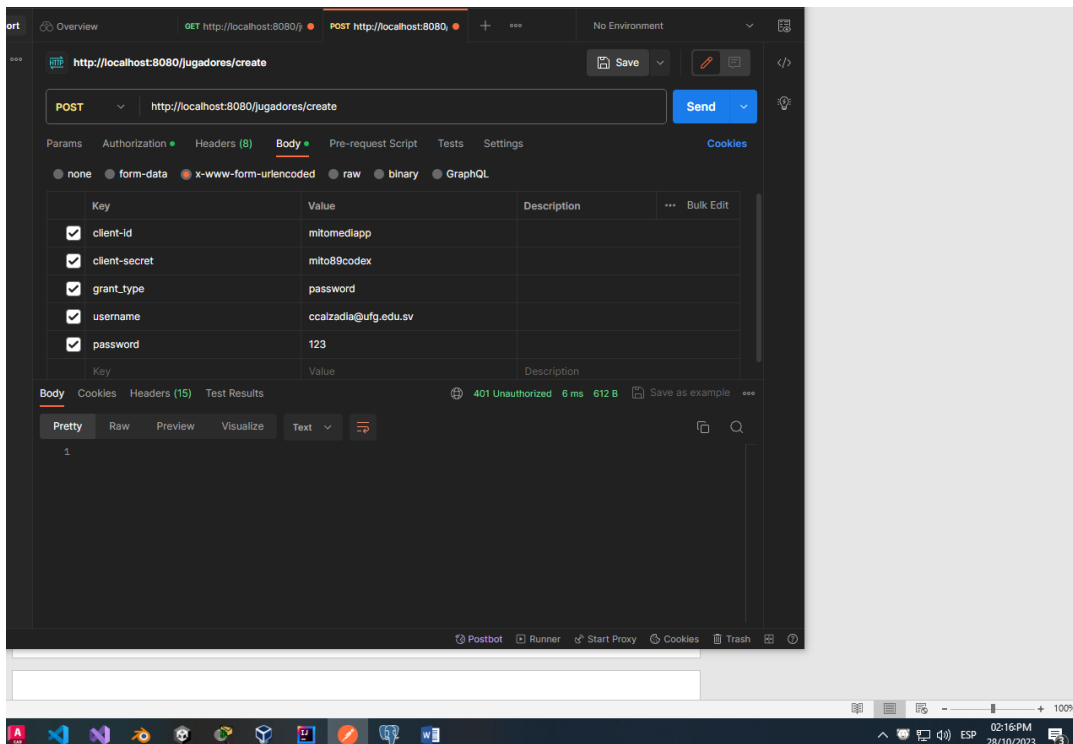
DELETE



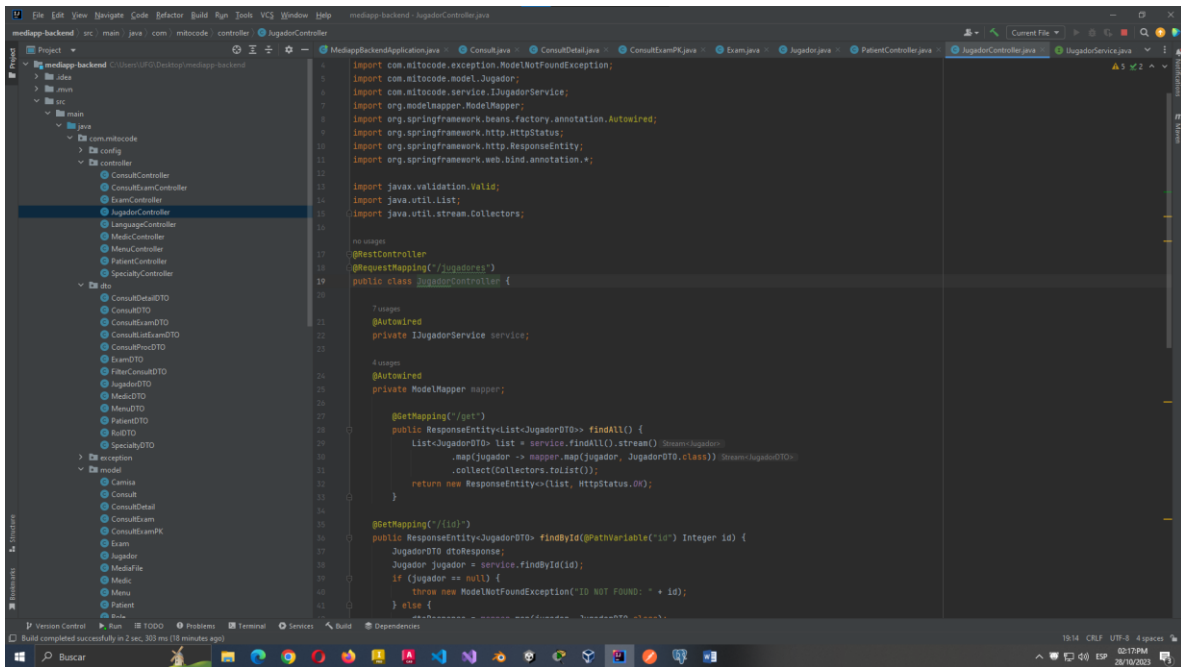
UPDATE



CREATE

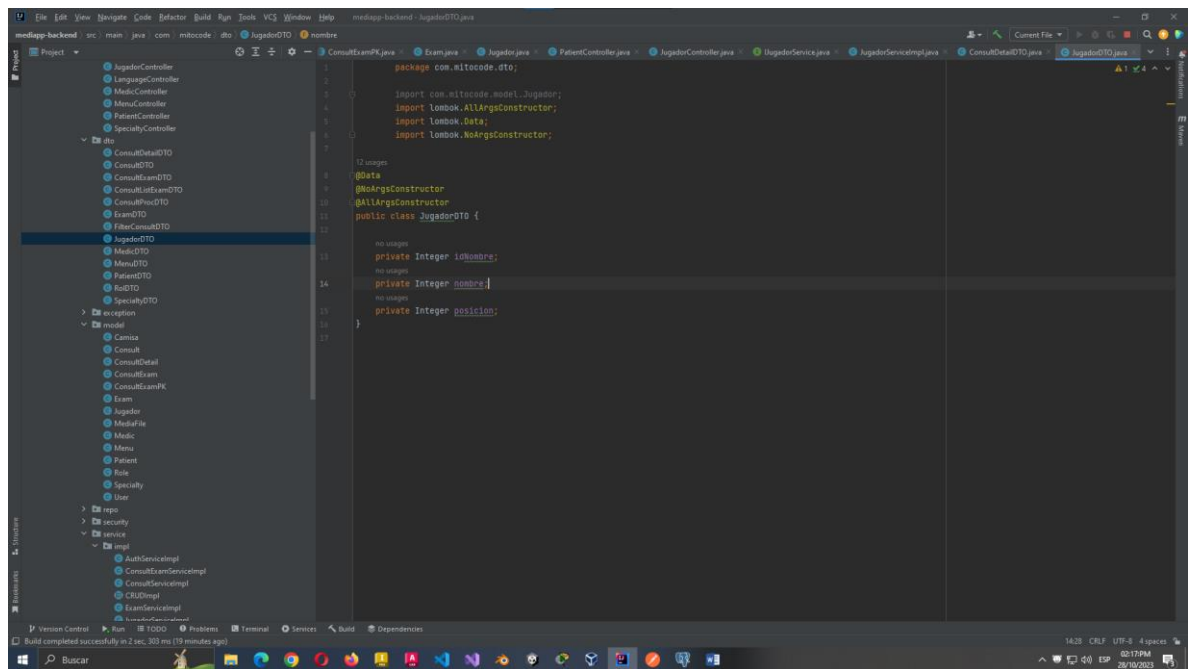


5. Controlador.



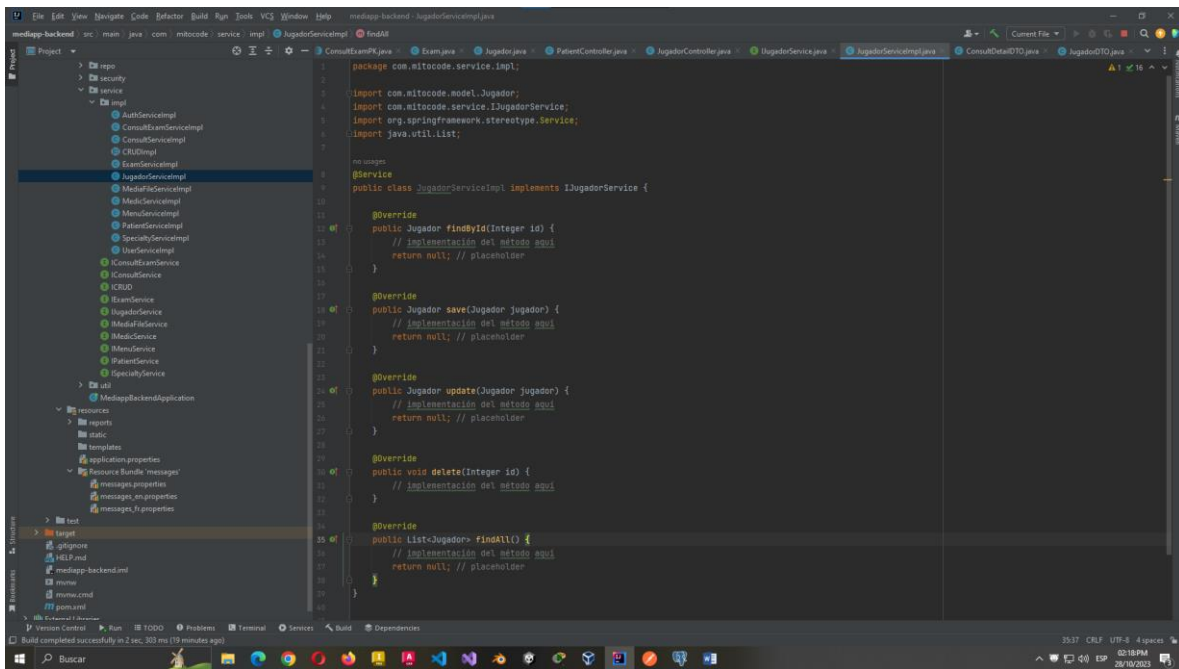
```
1 import com.mitocode.exception.ModelNotFoundException;
2 import com.mitocode.model.Jugador;
3 import com.mitocode.service.IJugadorService;
4 import org.springframework.beans.factory.annotation.Autowired;
5 import org.springframework.http.HttpStatus;
6 import org.springframework.http.ResponseEntity;
7 import org.springframework.web.bind.annotation.*;
8
9 import javax.validation.Valid;
10 import java.util.List;
11 import java.util.stream.Collectors;
12
13 @RestController
14 @RequestMapping("/jugadores")
15 public class JugadorController {
16
17     @Autowired
18     private IJugadorService service;
19
20     @Autowired
21     private ModelMapper mapper;
22
23     @GetMapping("/get")
24     public ResponseEntity<List<JugadorDTO>> findAll() {
25         List<JugadorDTO> list = service.findAll().stream().map(jugador -> mapper.map(jugador, JugadorDTO.class)).collect(Collectors.toList());
26         return new ResponseEntity<>(list, HttpStatus.OK);
27     }
28
29     @GetMapping("/{id}")
30     public ResponseEntity<JugadorDTO> findById(@PathVariable("id") Integer id) {
31         JugadorDTO dtoResponse;
32         Jugador jugador = service.findById(id);
33         if (jugador == null) {
34             throw new ModelNotFoundException("ID NOT FOUND: " + id);
35         } else {
36             dtoResponse = mapper.map(jugador, JugadorDTO.class);
37         }
38         return new ResponseEntity<>(dtoResponse, HttpStatus.OK);
39     }
40 }
```

a. DTO.



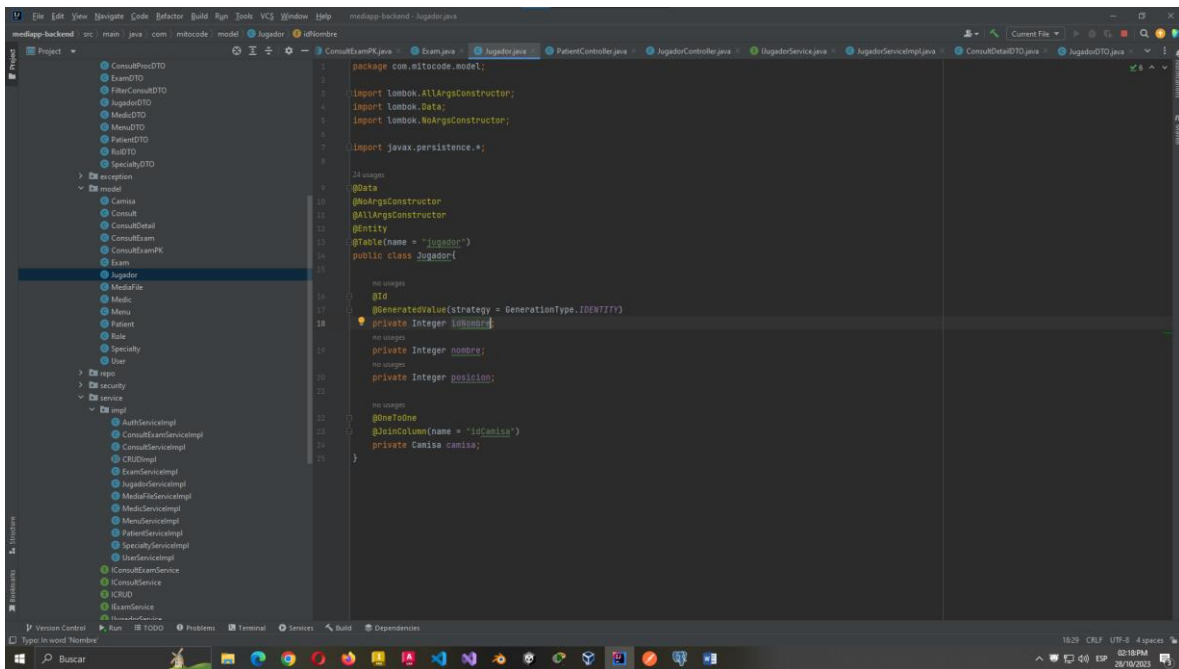
```
1 package com.mitocode.dto;
2
3 import com.mitocode.model.Jugador;
4 import lombok.AllArgsConstructor;
5 import lombok.Data;
6 import lombok.NoArgsConstructor;
7
8 @Data
9 @NoArgsConstructor
10 @AllArgsConstructor
11 public class JugadorDTO {
12
13     private Integer idnombre;
14     private Integer nombre;
15     private Integer position;
16 }
```

6. Service.



7. Modelo.

JUGADOR



CAMISA

