



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

Estudiante:

Kevin Giovanni Flores Arana

FA100620.

Docente: Carlos Boris Martinez Calzadia.

Actividad: LABORATORIO 03.

Modulo: Desarrollo de Aplicaciones Web 01

Fecha: 28/10/2023

POSTMAN

(INSERT)

The screenshot displays the Postman application interface. The top navigation bar includes 'Home', 'Workspaces', 'API Network', and 'Explore'. The left sidebar shows 'My Workspace' with a 'New Collection' button. The main area is set to 'Overview' for a POST request to 'http://localhost:8080/jugador/create'. The request body is a JSON object:

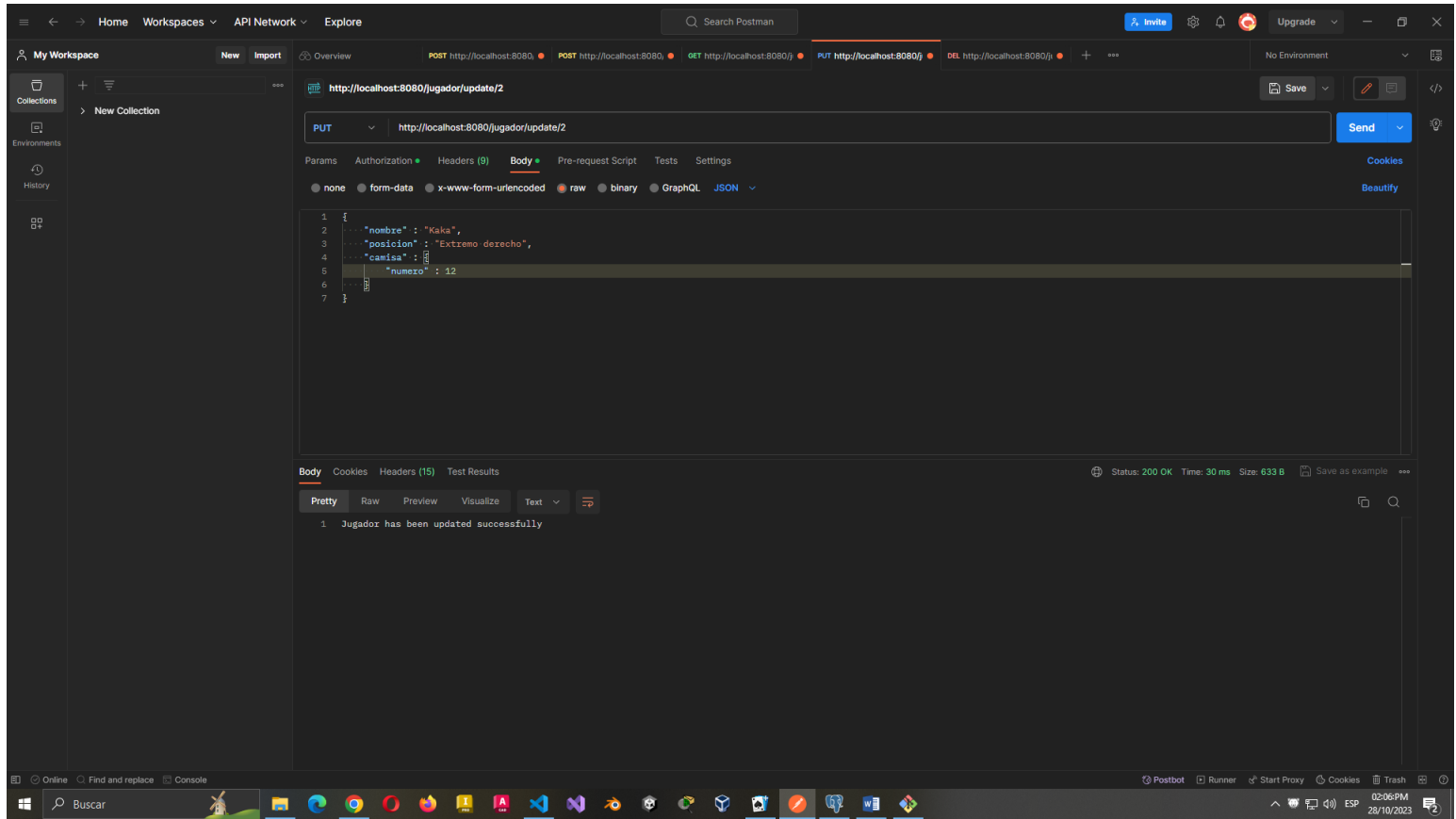
```
1 {
2   "nombre": "Ronaldo",
3   "posicion": "Delantero",
4   "camisa": {
5     "numero": 7
6   }
7 }
8
9
```

The bottom section shows the 'Body' tab with the response in 'Pretty' format:

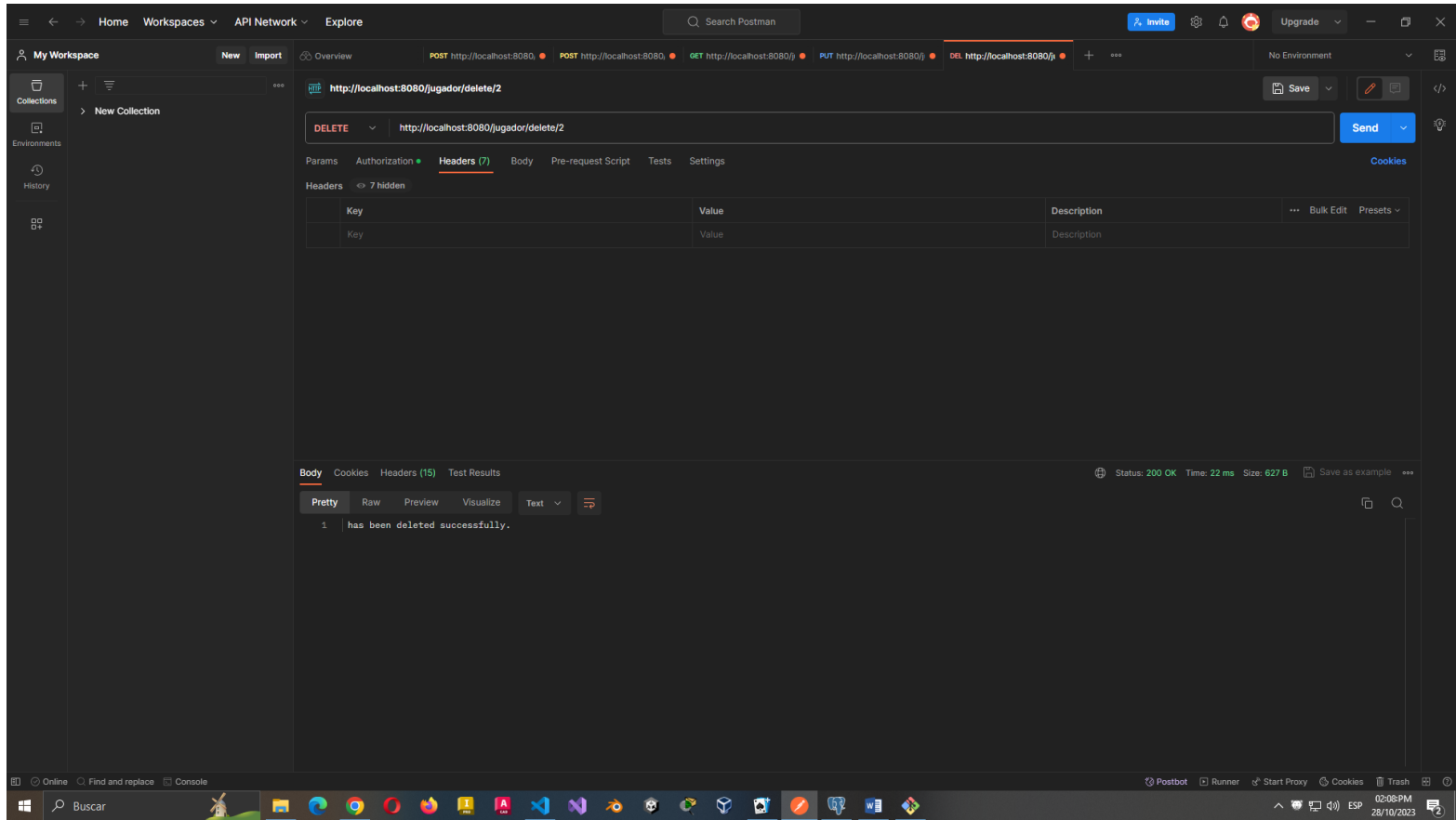
```
1 {
2   "idJugador": 2,
3   "camisa": {
4     "idCamisa": 2,
5     "numero": 7
6   },
7   "nombre": "Ronaldo",
8   "posicion": "Delantero"
9 }
```

The status bar at the bottom indicates 'Status: 200 OK', 'Time: 17 ms', and 'Size: 688 B'. The Windows taskbar is visible at the very bottom.

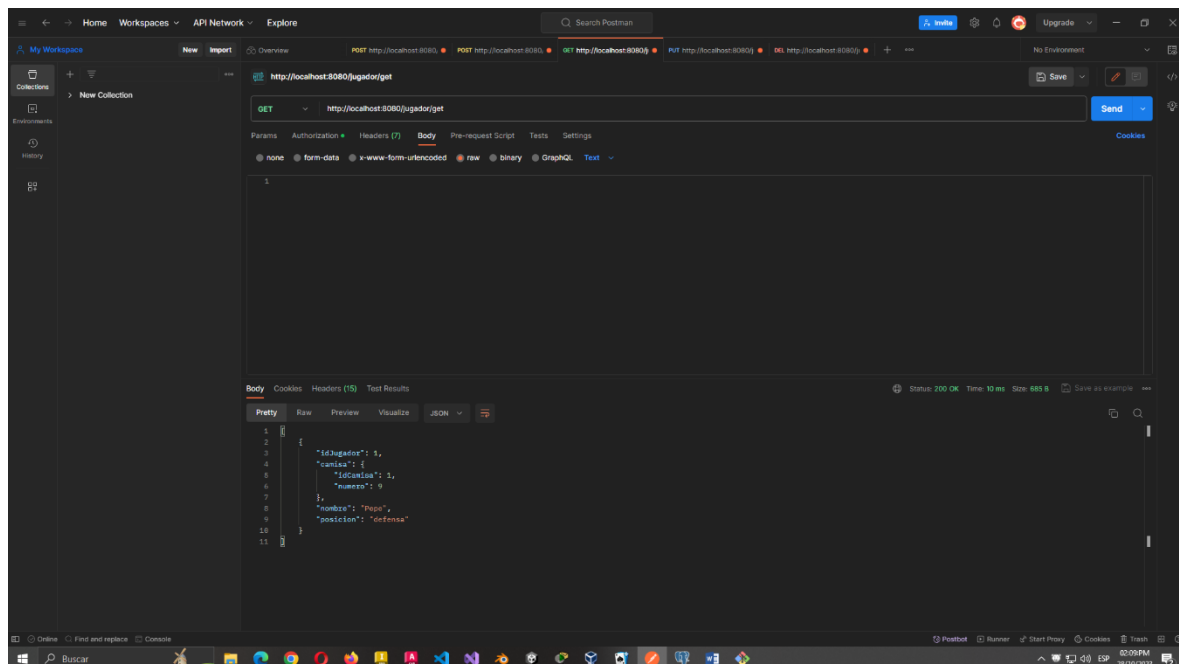
(Update)



(DELETE)



(GET)



Modelo Jugador

```
src > main > java > com > mitocode > model > J Jugador.java > Jugador > camisa
15
16 @Data
17 @NoArgsConstructor
18 @AllArgsConstructor
19 @Entity
20 public class Jugador {
21
22     @Id
23     @GeneratedValue(strategy = GenerationType.IDENTITY)
24     private Integer idJugador;
25
26     private String Nombre;
27
28     private String Posicion;
29
30     @OneToOne(cascade = CascadeType.ALL)
31     @JoinTable(name = "camisa_ent",
32         joinColumns =
33             {@JoinColumn(name = "idJugador", referencedColumnName = "idJugador") },
34         inverseJoinColumns =
35             {@JoinColumn(name = "idCamisa", referencedColumnName = "idCamisa")})
36     private Camisa camisa;
37
38 }
39
```

Ln 31, Col 34 Spaces: 4 UTF-8 CRLF {} Java Port: 5500 12 Spell 02:10 PM 28/10/2023

Modelo Camisa

```
JugadorController.java 6  JugadorService.java  Camisa.java X  Jugador.java
src > main > java > com > mitocode > model >  Camisa.java > ...
5  import javax.persistence.GenerationType;
6  import javax.persistence.Id;
7
8  import lombok.AllArgsConstructor;
9  import lombok.Data;
10 import lombok.NoArgsConstructor;
11
12 @Data
13 @NoArgsConstructor
14 @AllArgsConstructor
15 @Entity
16 public class Camisa {
17
18     @Id
19     @GeneratedValue(strategy = GenerationType.IDENTITY)
20     private Integer idCamisa;
21
22     private Integer Numero;
23
24 }
25
```

Ln 6, Col 29 Spaces: 4 UTF-8 CRLF {} Java Port: 5500 4 Spell R C

02:11:PM 28/10/2023

Controlador

```
J JugadorController.java 6 x J JugadorService.java J Camisa.java J Jugador.java
src > main > java > com > mitocode > controller > J JugadorController.java > JugadorController > updateJugador(Integer, Jugador)
26
27 @RestController
28 @RequestMapping("/jugador")
29 @RequiredArgsConstructor
30 public class JugadorController {
31
32     @Autowired
33     private IJugadorService repo;
34
35     @GetMapping("/get")
36     public List<Jugador> getJugador() {
37         return (List<Jugador>) repo.getJugador();
38     }
39
40     @PostMapping("/create")
41     public Jugador createJugador(@RequestBody Jugador jugador) {
42         return repo.createJugador(jugador);
43     }
44
45     @PutMapping("/update/{id}")
46     public String updateJugador(@PathVariable Integer id, @RequestBody Jugador updatedJugador) {
47         Jugador updated = repo.updateJugador(id, updatedJugador);
48
49         if (updated != null) {
50             return "Jugador has been updated successfully";
51         } else {
52             return "not found";
53         }
54     }
55
56     @DeleteMapping(value = "/delete/{id}")
57     public String deleteJugador(@PathVariable Integer id) {
58         repo.deleteJugador(id);
59         return "has been deleted successfully.";
60     }
61 }
62
63
```

Ln 45, Col 25 (6 selected) Spaces: 4 UTF-8 CRLF {} Java Port: 5500 9 Spell 02:12PM 28/10/2023