

# CAPTURAS DE PANTALLA

## Enfermedades-Services

The screenshot displays an IDE with two panels. The top panel shows the project structure for 'enfermedad-service'. The bottom panel shows the code for two files: 'TratamientoClient.java' and 'EnfermedadController.java'.

**Project Structure:**

- src
  - main
    - java\com\example\e...
      - client
        - TratamientoClient.j...
      - controller
        - EnfermedadControl...
      - entity
        - Enfermedad.java
      - model
        - Tratamiento.java
      - repository
        - EnfermedadReposit...
      - service
        - EnfermedadService....
        - EnfermedadServiceA...

**Code for TratamientoClient.java:**

```
1 package com.example.enfermedad_service.client;
2
3 import java.util.List;
4
5 import org.springframework.cloud.openfeign.FeignClient;
6 import org.springframework.web.bind.annotation.GetMapping;
7 import org.springframework.web.bind.annotation.PathVariable;
8
9 import com.example.enfermedad_service.model.Tratamiento;
10
11 @FeignClient(name = "tratamiento-service", url = "http://localhost:8081")
12 public interface TratamientoClient {
13
14     @GetMapping("/tratamientos/enfermedad/{id}")
15     List<Tratamiento> obtenerTratamientosPorEnfermedad(@PathVariable("id") Long enfermedadId);
16 }
17
```

**Code for EnfermedadController.java:**

```
11
12 import java.util.List;
13
14 @RestController
15 @RequestMapping("/api/enfermedades")
16 public class EnfermedadController {
17
18     private final EnfermedadService enfermedadService;
19
20     @Autowired
21     public EnfermedadController(EnfermedadService enfermedadService) {
22         this.enfermedadService = enfermedadService;
23     }
24
25     // Endpoint para obtener los tratamientos de una enfermedad
26     @GetMapping("/{enfermedadId}/tratamientos")
27     public List<Tratamiento> obtenerTratamientosPorEnfermedad(@PathVariable Long enfermedadId) {
28         return enfermedadService.obtenerTratamientosPorEnfermedad(enfermedadId);
29     }
30 }
31
```

```
application.properties x Enfermedad.java x
enfermedad-service > src > main > java > com > example > enfermedad_service > entity > J Enfermedad.java > ...
1 package com.example.enfermedad_service.entity;
2
3 import jakarta.persistence.*; // Correcto para Spring Boot 3.x
4
5 @Entity
6 @Table(name = "enfermedades")
7 public class Enfermedad {
8
9     @Id
10    @GeneratedValue(strategy = GenerationType.IDENTITY)
11    private Long id;
12
13    @Column(nullable = false, unique = true)
14    private String nombre;
15
16    // Constructor sin argumentos (requerido por JPA)
17    public Enfermedad() {
18    }
19
20    // Getters y Setters
21    public Long getId() {
```

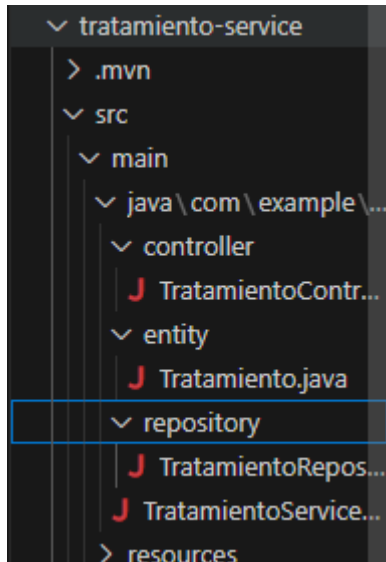
```
application.properties x Tratamiento.java x
C:\Users\Usuario\Desktop\ESPE\7mo\Distribuidas\Segundo Parcial\EXAMEN\Codigo\enfermedad-service\enfermedad-service\src\main\resources\application.properties > J Tratamiento.java > ...
3 public class Tratamiento {
4     private Long id;
5     private String nombre;
6
7     // Getters y setters
8     public Long getId() {
9         return id;
10    }
11
12    public void setId(Long id) {
13        this.id = id;
14    }
15
16    public String getNombre() {
17        return nombre;
18    }
19
20    public void setNombre(String nombre) {
21        this.nombre = nombre;
```

```
application.properties x EnfermedadRepository.java x
enfermedad-service > src > main > java > com > example > enfermedad_service > repository > J EnfermedadRepository.java
1 package com.example.enfermedad_service.repository;
2
3 import com.example.enfermedad_service.entity.Enfermedad;
4 import org.springframework.data.jpa.repository.JpaRepository;
5
6 public interface EnfermedadRepository extends JpaRepository<Enfermedad, Long> {
7 }
8
```

```
application.properties x EnfermedadService.java x
enfermedad-service > src > main > java > com > example > enfermedad_service > service > J EnfermedadService.java > ...
1 import java.util.List;
2
3 import org.springframework.stereotype.Service;
4
5 import com.example.enfermedad_service.client.TratamientoClient;
6 import com.example.enfermedad_service.model.Tratamiento;
7
8 @Service
9 public class EnfermedadService {
10
11     private final TratamientoClient tratamientoClient;
12
13     public EnfermedadService(TratamientoClient tratamientoClient) {
14         this.tratamientoClient = tratamientoClient;
15     }
16
17     public List<Tratamiento> obtenerTratamientosPorEnfermedad(Long enfermedadId) {
18         return tratamientoClient.obtenerTratamientosPorEnfermedad(enfermedadId);
19     }
20 }
21
22
23
```

```
application.properties EnfermedadServiceApplication.java x
enfermedad-service > src > main > java > com > example > enfermedad_service > EnfermedadServiceApplication.java
1 package com.example.enfermedad_service;
2
3 import org.springframework.boot.SpringApplication;
4 import org.springframework.boot.autoconfigure.SpringBootApplication;
5 import org.springframework.cloud.openfeign.EnableFeignClients;
6
7 @SpringBootApplication
8 @EnableFeignClients // Habilita Feign en la aplicación
9 public class EnfermedadServiceApplication {
10
11     public static void main(String[] args) {
12         SpringApplication.run(EnfermedadServiceApplication.class, args);
13     }
14 }
15
```

## Tratamiento-Service



```
application.properties TratamientoController.java x
tratamiento-service > src > main > java > com > example > tratamiento_service > controller > TratamientoController.java > Language Support for Java(TM) by Red Hat > TratamientoCont
18 @RestController
19 @RequestMapping("/api/tratamientos")
20 public class TratamientoController {
21
22     private final TratamientoRepository tratamientoRepository;
23
24     // Constructor para inyección de dependencia
25     public TratamientoController(TratamientoRepository tratamientoRepository) {
26         this.tratamientoRepository = tratamientoRepository;
27     }
28
29     // Obtener todos los tratamientos
30     @GetMapping
31     public List<Tratamiento> listarTratamientos() {
32         return tratamientoRepository.findAll();
33     }
34
35     // Obtener un tratamiento por ID
36     @GetMapping("/{id}")
37     public ResponseEntity<Tratamiento> obtenerTratamientoPorId(@PathVariable Long id) {
38         return tratamientoRepository.findById(id)
39             .map(ResponseEntity::ok) // Si se encuentra, devolver 200 OK
40             .orElse(ResponseEntity.notFound().build()); // Si no se encuentra, devolver 404 Not Found
41     }
42
43     // Crear un nuevo tratamiento
44     @PostMapping
45     public Tratamiento crearTratamiento(@RequestBody Tratamiento tratamiento) {
46         return tratamientoRepository.save(tratamiento); // Guardar el tratamiento en la base de datos
47     }
48 }
```

```
application.properties  Tratamiento.java X
tratamiento-service > src > main > java > com > example > tratamiento_service > entity > J Tratamiento.java
1 package com.example.tratamiento_service.entity;
2
3 import jakarta.persistence.*;
4
5 @Entity
6 public class Tratamiento {
7
8     @Id
9     @GeneratedValue(strategy = GenerationType.IDENTITY)
10    private Long id;
11
12    @Column(nullable = false, unique = true)
13    private String nombre;
14
15    // Getters y Setters
16    public Long getId() {
17        return id;
18    }
19
20    public void setId(Long id) {
21        this.id = id;
22    }
23
24    public String getNombre() {
25        return nombre;
26    }
27
28    public void setNombre(String nombre) {
29        this.nombre = nombre;
30    }
31 }
32

application.properties  TratamientoRepository.java X
tratamiento-service > src > main > java > com > example > tratamiento_service > repository > J TratamientoRepository.java
1 package com.example.tratamiento_service.repository;
2
3 import com.example.tratamiento_service.entity.Tratamiento;
4 import org.springframework.data.jpa.repository.JpaRepository;
5
6 public interface TratamientoRepository extends JpaRepository<Tratamiento, Long> {
7 }
8

application.properties  TratamientoServiceApplication.java X
tratamiento-service > src > main > java > com > example > tratamiento_service > J TratamientoServiceApplication.java > Language Support for Java(TM) by Red Hat > { } com.example.tratam
1 package com.example.tratamiento_service;
2
3 import org.springframework.boot.SpringApplication;
4 import org.springframework.boot.autoconfigure.SpringBootApplication;
5
6 @SpringBootApplication
7 public class TratamientoServiceApplication {
8
9     public static void main(String[] args) {
10        SpringApplication.run(TratamientoServiceApplication.class, args);
11    }
12
13 }
14
```