ECOLE NORMALE SUPÉRIEURE DE L'ENSEIGNEMENT
TECHNIQUE DE MOHAMMEDIA
UNIVERSITÉ HASSAN II DE CASABLANCA

ENSET

المدرسة العليا لأساتذة التعليم
التقني المحمدية جامعة الحسن
الثاني بالدار البيضاء

2021-2022

Département Mathématiques et Informatique
GLSID 2 – S4

Architecture JEE et Middlewares

Rapport TP 5
Mapping des associations (suite)

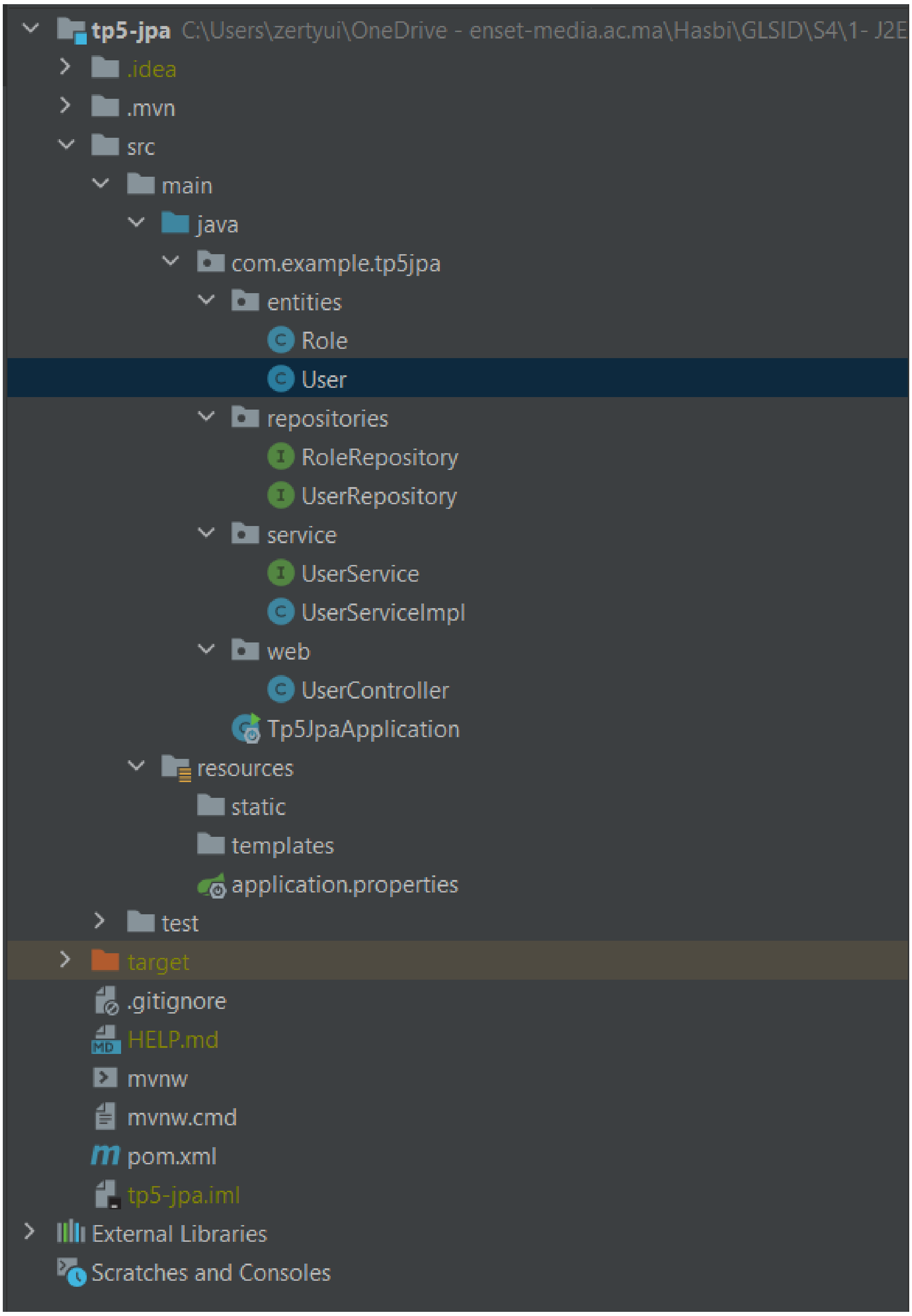
Préparé par : Fatima Zahra HASBI
Encadré par : Mr Mohamed YOUSSFI

Introduction

Ce tp est une continuité de Mapping des associations en JPA Hibernate Spring Data.

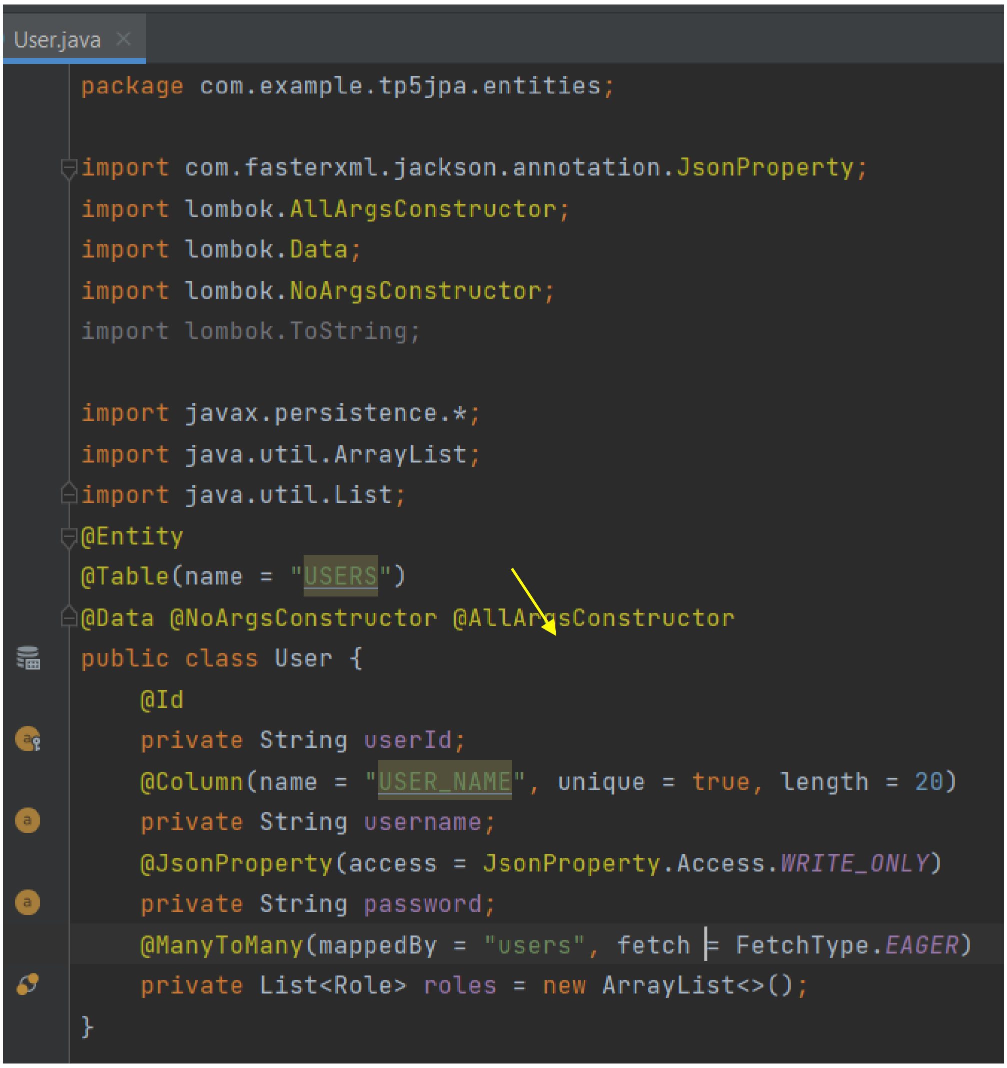
Réalisation

La structure de code



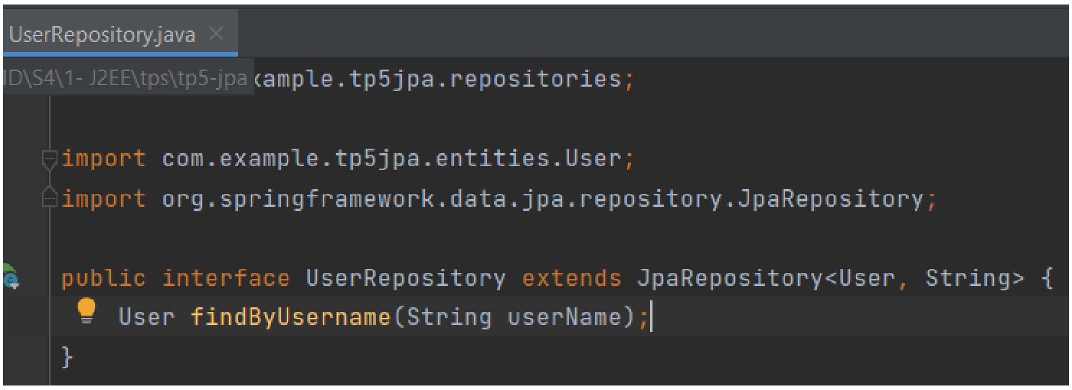
```
tp5-jpa  C:\Users\zertyui\OneDrive - enset-media.ac.ma\Hasbi\GLSID\S4\1- J2E
  .idea
  .mvn
  src
    main
      java
        com.example.tp5jpa
          entities
            Role
            User
          repositories
            RoleRepository
            UserRepository
          service
            UserService
            UserServiceImpl
          web
            UserController
          Tp5JpaApplication
      resources
        static
        templates
        application.properties
    test
  target
  .gitignore
  HELP.md
  mvnw
  mvnw.cmd
  pom.xml
  tp5-jpa.iml
External Libraries
Scratches and Consoles
```

L’entité User dans le package entities



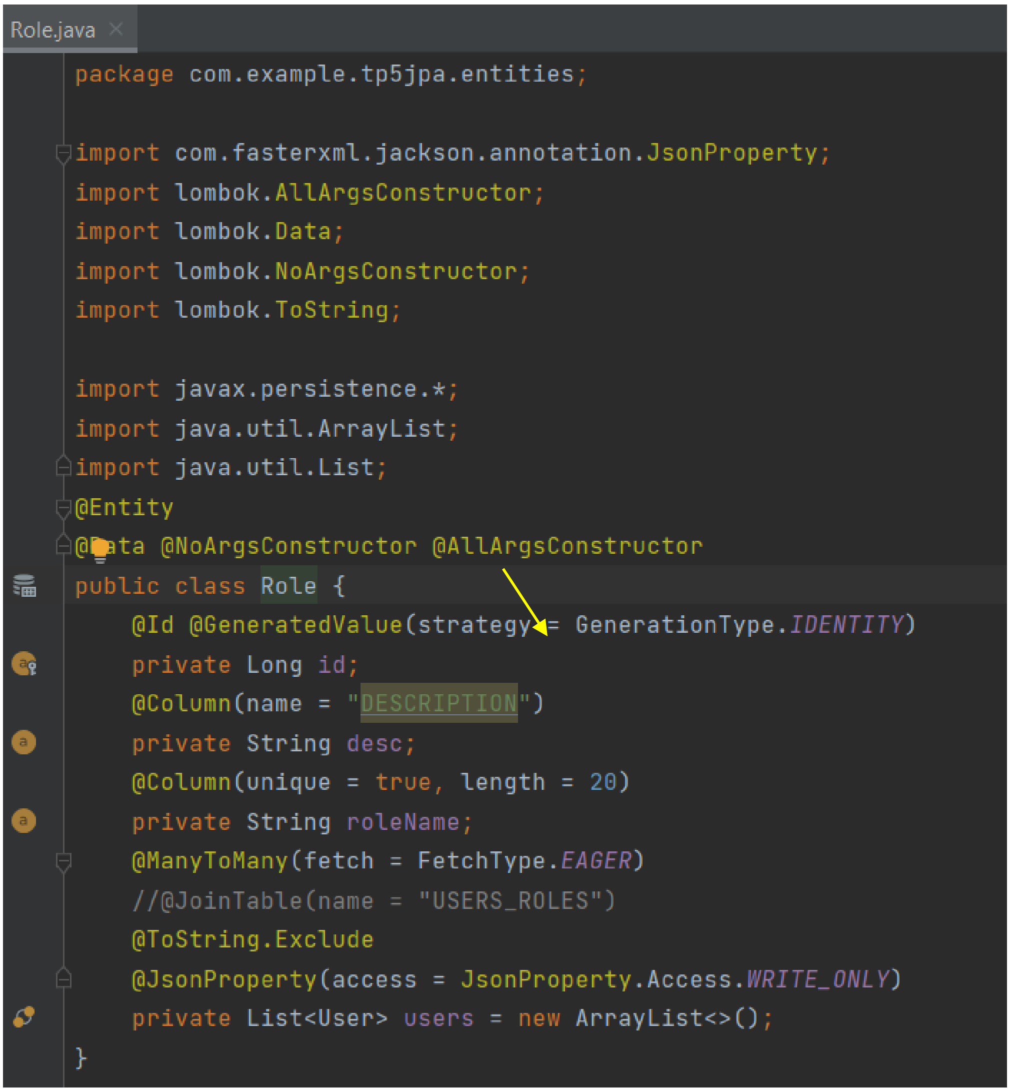
```java
package com.example.tp5jpa.entities;

import com.fasterxml.jackson.annotation.JsonProperty;
import lombok.AllArgsConstructor;
import lombok.Data;
import lombok.NoArgsConstructor;
import lombok.ToString;

import javax.persistence.*;
import java.util.ArrayList;
import java.util.List;
@Entity
@Table(name = "USERS")
@Data @NoArgsConstructor @AllArgsConstructor
public class User {
    @Id
    private String userId;
    @Column(name = "USER_NAME", unique = true, length = 20)
    private String username;
    @JsonProperty(access = JsonProperty.Access.WRITE_ONLY)
    private String password;
    @ManyToMany(mappedBy = "users", fetch = FetchType.EAGER)
    private List<Role> roles = new ArrayList<>();
}
```

L’interface UserRepository dans le package repositories



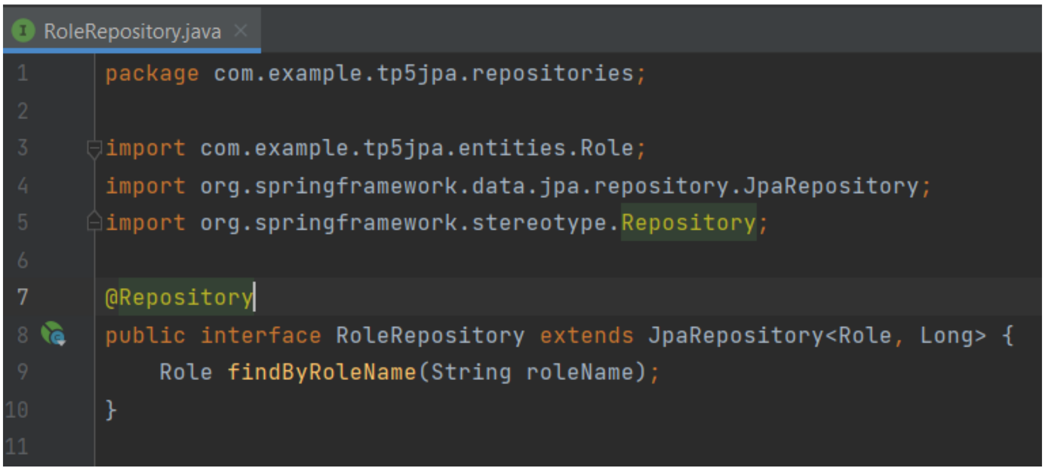
```java
ID\S4\1- J2EE\tps\tp5-jpa xample.tp5jpa.repositories;

import com.example.tp5jpa.entities.User;
import org.springframework.data.jpa.repository.JpaRepository;

public interface UserRepository extends JpaRepository<User, String> {
    User findByUsername(String userName);
}
```

L’entité Role dans le package entities



```java
package com.example.tp5jpa.entities;

import com.fasterxml.jackson.annotation.JsonProperty;
import lombok.AllArgsConstructor;
import lombok.Data;
import lombok.NoArgsConstructor;
import lombok.ToString;

import javax.persistence.*;
import java.util.ArrayList;
import java.util.List;
@Entity
@Data @NoArgsConstructor @AllArgsConstructor
public class Role {
    @Id @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;
    @Column(name = "DESCRIPTION")
    private String desc;
    @Column(unique = true, length = 20)
    private String roleName;
    @ManyToMany(fetch = FetchType.EAGER)
    //@JoinTable(name = "USERS_ROLES")
    @ToString.Exclude
    @JsonProperty(access = JsonProperty.Access.WRITE_ONLY)
    private List<User> users = new ArrayList<>();
}
```

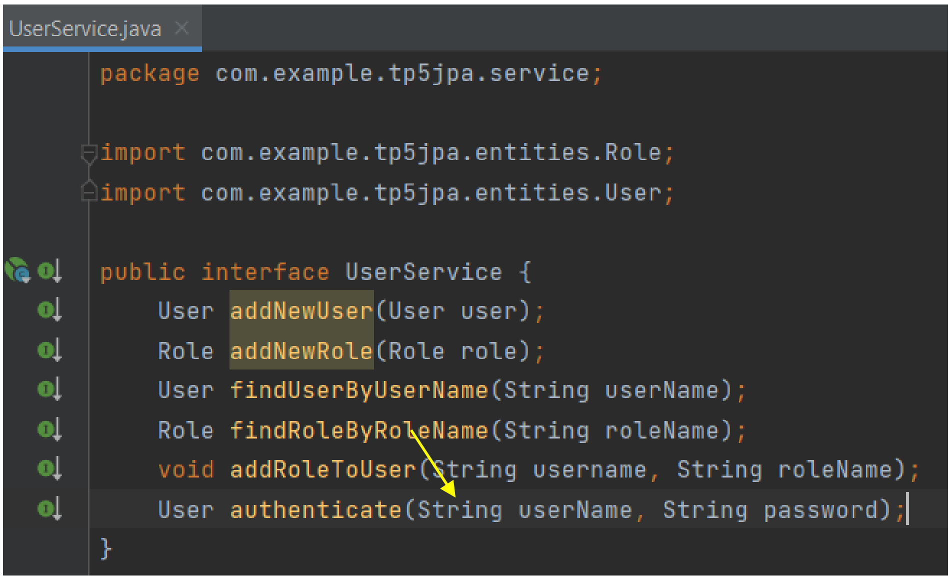
L’interface RoleRepository dans le package repositories



```java
package com.example.tp5jpa.repositories;

import com.example.tp5jpa.entities.Role;
import org.springframework.data.jpa.repository.JpaRepository;
import org.springframework.stereotype.Repository;

@Repository
public interface RoleRepository extends JpaRepository<Role, Long> {
    Role findByRoleName(String roleName);
}

```

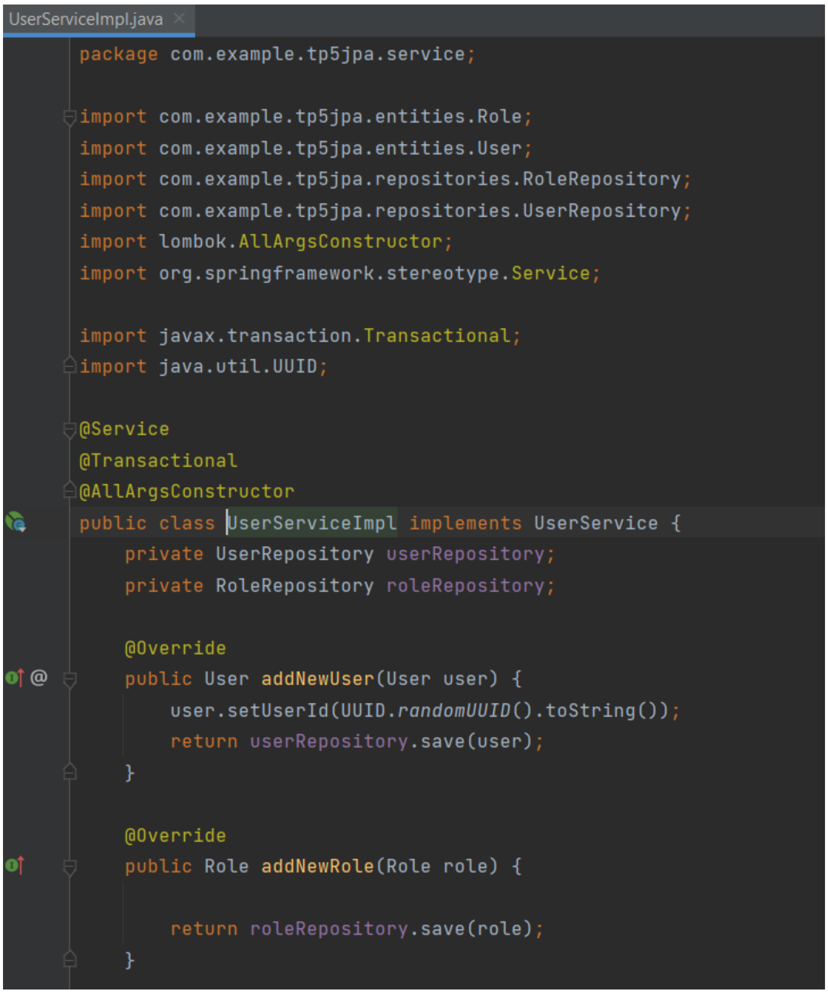
La couche service (couche metier)

L’interface UserService



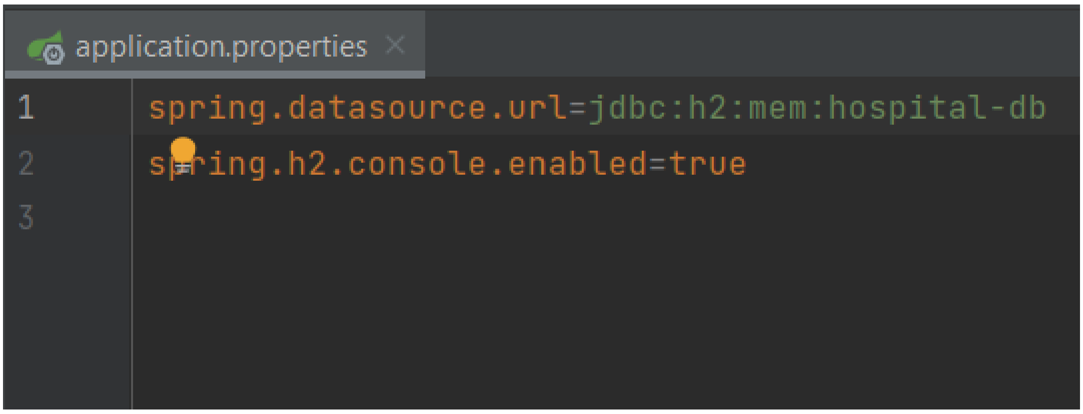
```java
package com.example.tp5jpa.service;

import com.example.tp5jpa.entities.Role;
import com.example.tp5jpa.entities.User;

public interface UserService {
    User addNewUser(User user);
    Role addNewRole(Role role);
    User findUserByUserName(String userName);
    Role findRoleByRoleName(String roleName);
    void addRoleToUser(String username, String roleName);
    User authenticate(String userName, String password);
}
```

L’implémentation de l’interface UserService



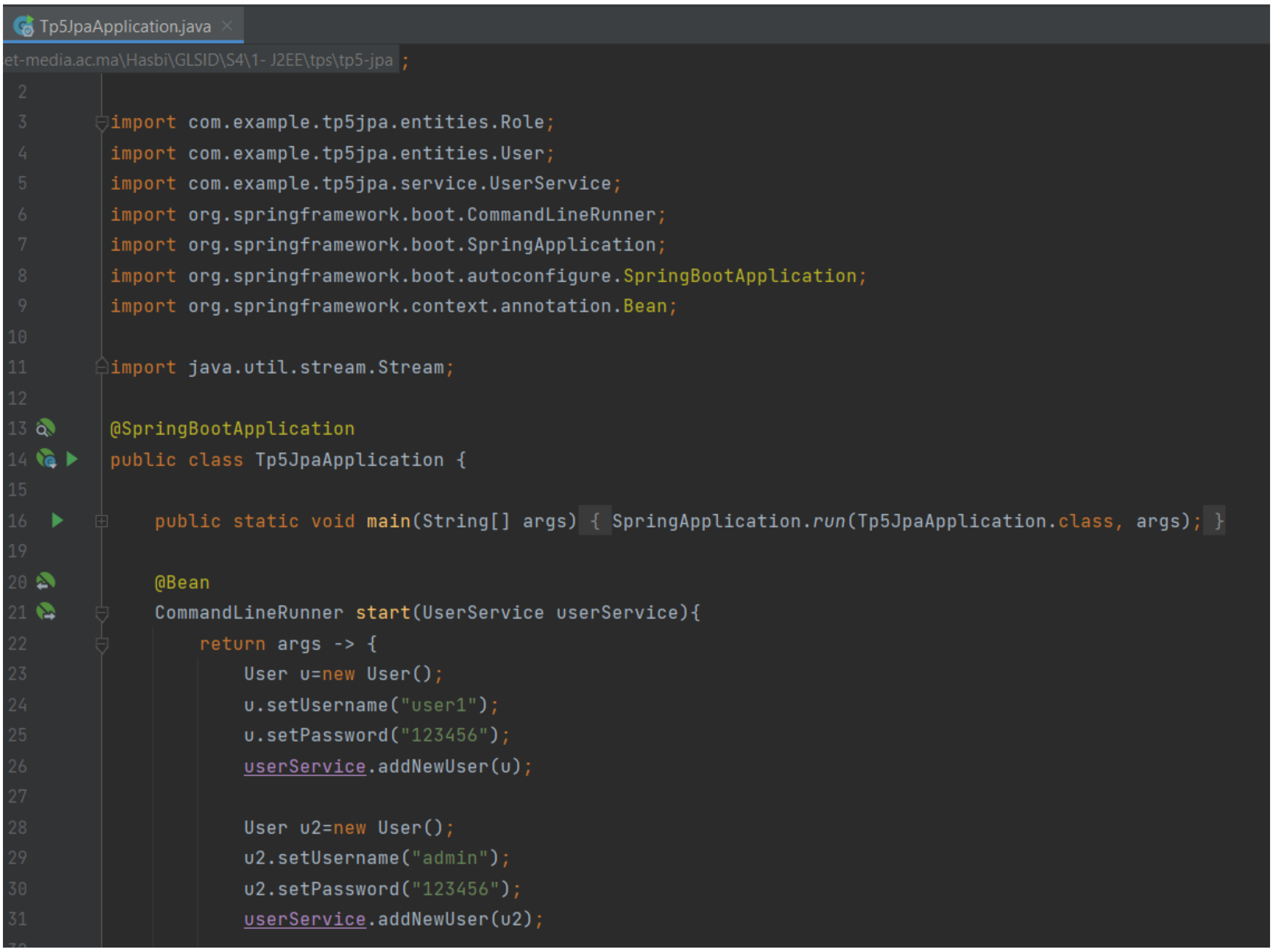
```java
package com.example.tp5jpa.service;

import com.example.tp5jpa.entities.Role;
import com.example.tp5jpa.entities.User;
import com.example.tp5jpa.repositories.RoleRepository;
import com.example.tp5jpa.repositories.UserRepository;
import lombok.AllArgsConstructor;
import org.springframework.stereotype.Service;

import javax.transaction.Transactional;
import java.util.UUID;

@Service
@Transactional
@AllArgsConstructor
public class UserServiceImpl implements UserService {
    private UserRepository userRepository;
    private RoleRepository roleRepository;

    @Override
    public User addNewUser(User user) {
        user.setUserId(UUID.randomUUID().toString());
        return userRepository.save(user);
    }

    @Override
    public Role addNewRole(Role role) {

        return roleRepository.save(role);
    }
```

Configuration du fichier application.properties



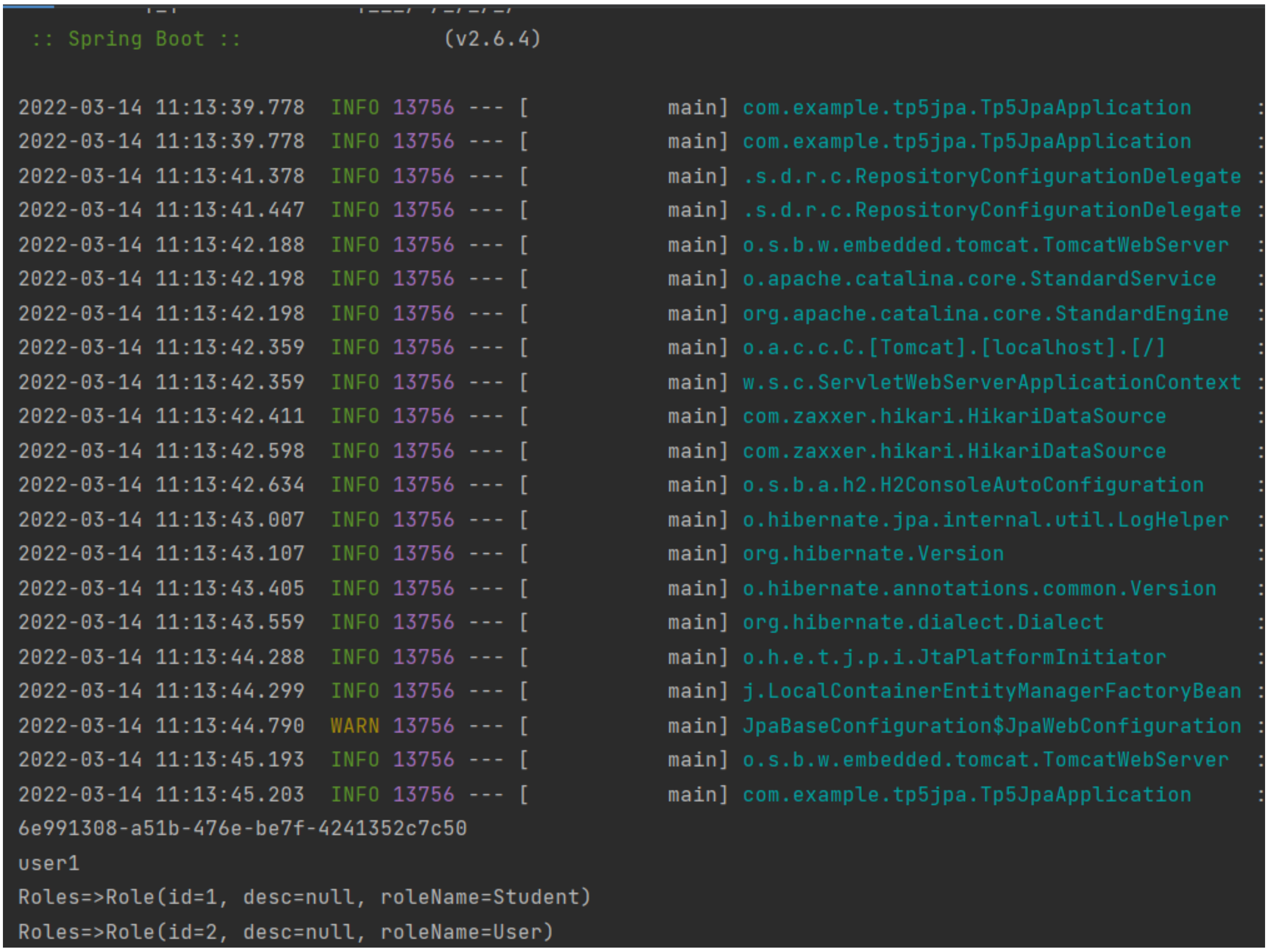
```
spring.datasource.url=jdbc:h2:mem:hospital-db
spring.h2.console.enabled=true
```

La fonction main() pour tester l'application



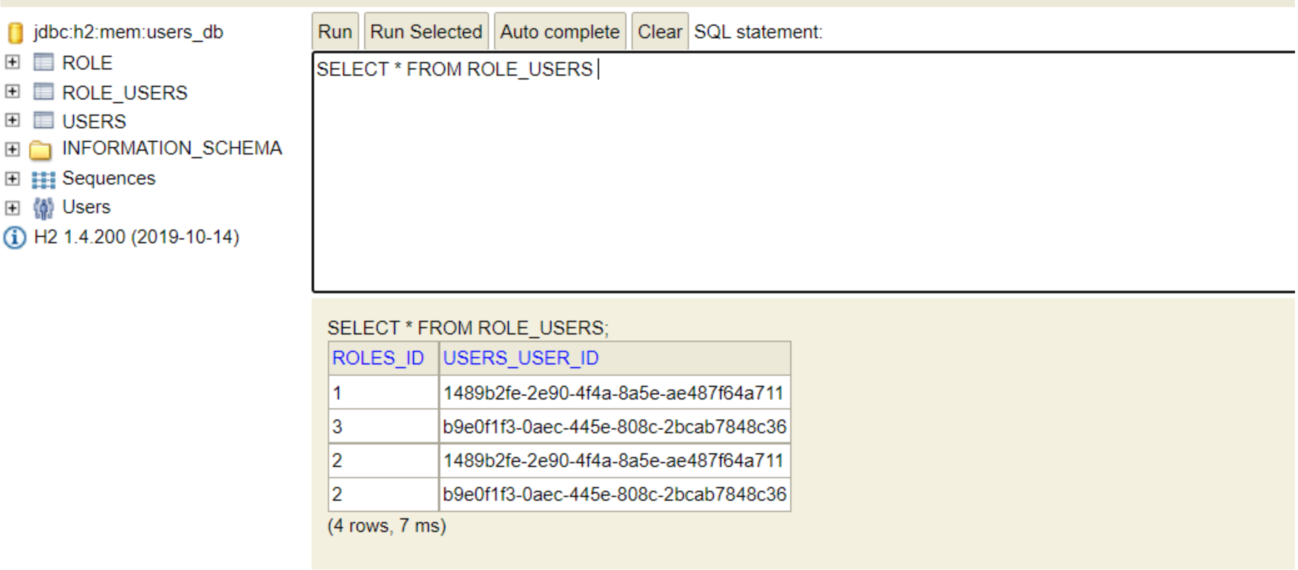
```
et-media.ac.ma\Hasbi\GLSID\S4\1- J2EE\tps\tp5-jpa ;

import com.example.tp5jpa.entities.Role;
import com.example.tp5jpa.entities.User;
import com.example.tp5jpa.service.UserService;
import org.springframework.boot.CommandLineRunner;
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.context.annotation.Bean;

import java.util.stream.Stream;

@SpringBootApplication
public class Tp5JpaApplication {

    public static void main(String[] args) { SpringApplication.run(Tp5JpaApplication.class, args); }

    @Bean
    CommandLineRunner start(UserService userService){
        return args -> {
            User u=new User();
            u.setUsername("user1");
            u.setPassword("123456");
            userService.addNewUser(u);

            User u2=new User();
            u2.setUsername("admin");
            u2.setPassword("123456");
            userService.addNewUser(u2);
```

Résultat dans la console



```
 :: Spring Boot ::                (v2.6.4)

2022-03-14 11:13:39.778  INFO 13756 --- [           main] com.example.tp5jpa.Tp5JpaApplication     :
2022-03-14 11:13:39.778  INFO 13756 --- [           main] com.example.tp5jpa.Tp5JpaApplication     :
2022-03-14 11:13:41.378  INFO 13756 --- [           main] .s.d.r.c.RepositoryConfigurationDelegate :
2022-03-14 11:13:41.447  INFO 13756 --- [           main] .s.d.r.c.RepositoryConfigurationDelegate :
2022-03-14 11:13:42.188  INFO 13756 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  :
2022-03-14 11:13:42.198  INFO 13756 --- [           main] o.apache.catalina.core.StandardService   :
2022-03-14 11:13:42.198  INFO 13756 --- [           main] org.apache.catalina.core.StandardEngine  :
2022-03-14 11:13:42.359  INFO 13756 --- [           main] o.a.c.c.C.[Tomcat].[localhost].[/]       :
2022-03-14 11:13:42.359  INFO 13756 --- [           main] w.s.c.ServletWebServerApplicationContext :
2022-03-14 11:13:42.411  INFO 13756 --- [           main] com.zaxxer.hikari.HikariDataSource       :
2022-03-14 11:13:42.598  INFO 13756 --- [           main] com.zaxxer.hikari.HikariDataSource       :
2022-03-14 11:13:42.634  INFO 13756 --- [           main] o.s.b.a.h2.H2ConsoleAutoConfiguration    :
2022-03-14 11:13:43.007  INFO 13756 --- [           main] o.hibernate.jpa.internal.util.LogHelper  :
2022-03-14 11:13:43.107  INFO 13756 --- [           main] org.hibernate.Version                    :
2022-03-14 11:13:43.405  INFO 13756 --- [           main] o.hibernate.annotations.common.Version   :
2022-03-14 11:13:43.559  INFO 13756 --- [           main] org.hibernate.dialect.Dialect            :
2022-03-14 11:13:44.288  INFO 13756 --- [           main] o.h.e.t.j.p.i.JtaPlatformInitiator       :
2022-03-14 11:13:44.299  INFO 13756 --- [           main] j.LocalContainerEntityManagerFactoryBean :
2022-03-14 11:13:44.790  WARN 13756 --- [           main] JpaBaseConfiguration$JpaWebConfiguration :
2022-03-14 11:13:45.193  INFO 13756 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  :
2022-03-14 11:13:45.203  INFO 13756 --- [           main] com.example.tp5jpa.Tp5JpaApplication     :
6e991308-a51b-476e-be7f-4241352c7c50
user1
Roles=>Role(id=1, desc=null, roleName=Student)
Roles=>Role(id=2, desc=null, roleName=User)
```

Ouvrant la base de données h2 sur localhost



```
SELECT * FROM ROLE_USERS
```

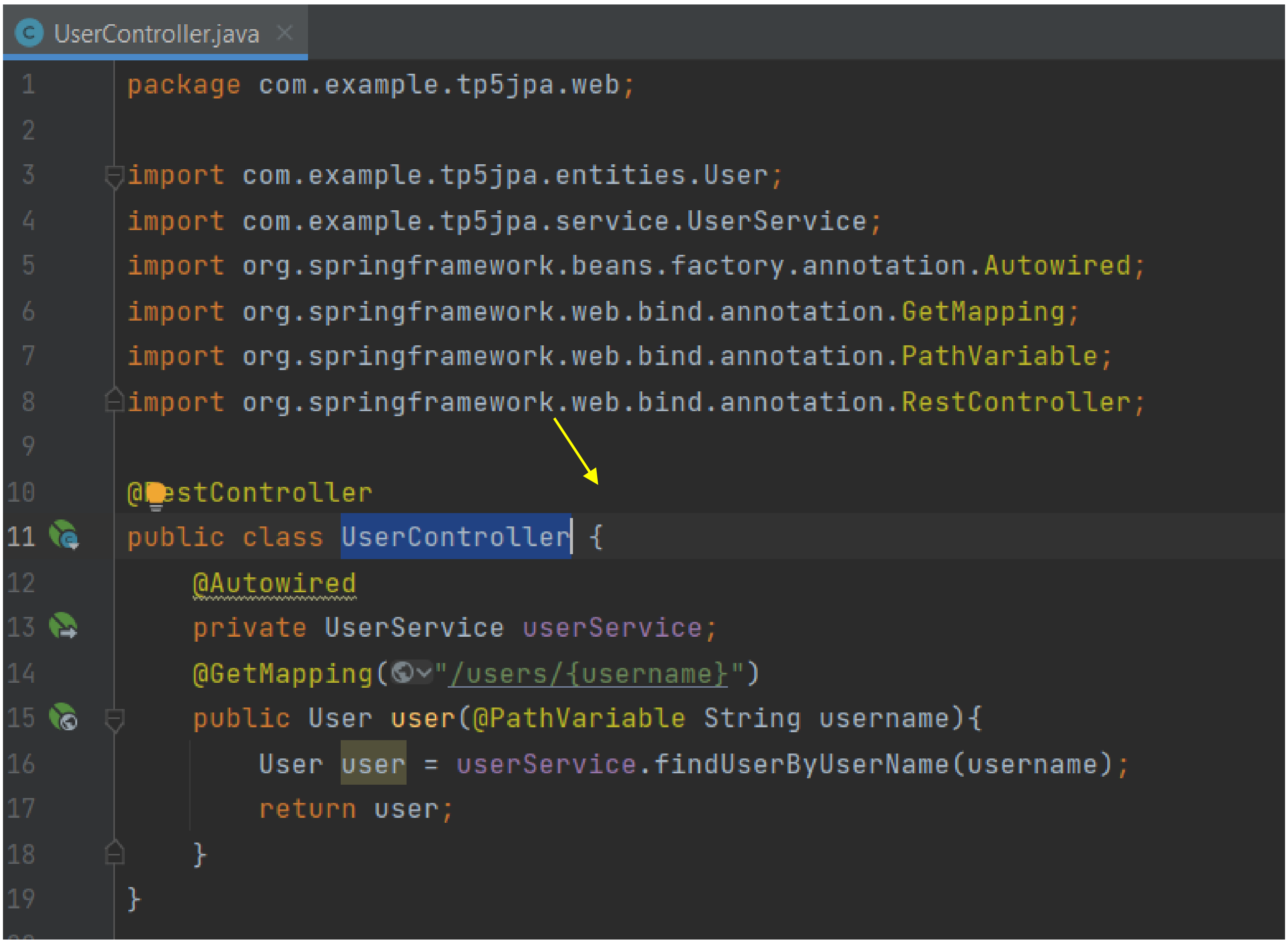
SELECT * FROM ROLE_USERS;

ROLES_ID	USERS_USER_ID
1	1489b2fe-2e90-4f4a-8a5e-ae487f64a711
3	b9e0f1f3-0aec-445e-808c-2bcab7848c36
2	1489b2fe-2e90-4f4a-8a5e-ae487f64a711
2	b9e0f1f3-0aec-445e-808c-2bcab7848c36

(4 rows, 7 ms)

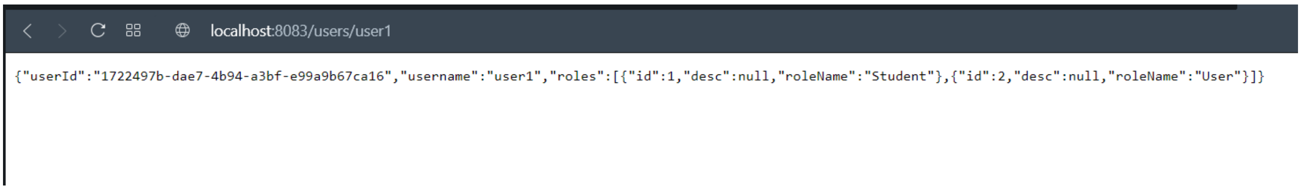
La couche web (couche metier)

La classe UserController



```java
package com.example.tp5jpa.web;

import com.example.tp5jpa.entities.User;
import com.example.tp5jpa.service.UserService;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.PathVariable;
import org.springframework.web.bind.annotation.RestController;

@RestController
public class UserController {
    @Autowired
    private UserService userService;
    @GetMapping("/users/{username}")
    public User user(@PathVariable String username){
        User user = userService.findUserByUserName(username);
        return user;
    }
}
```

Sur http://localhost:8083/users/user1



{"userId":"1722497b-dae7-4b94-a3bf-e99a9b67ca16","username":"user1","roles":[{"id":1,"desc":null,"roleName":"Student"},{"id":2,"desc":null,"roleName":"User"}]}