

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ - ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΠΜΣ «Πληροφορική»



Εργασία Μαθήματος
«Ιατρική Πληροφορική»

Θέμα: Ραντεβού Ασθενών-Ιατρών	
Επιβλέποντες Καθηγητές	Ευθύμιος Αλέπης
	Διονύσιος Σωτηρόπουλος
Όνομα φοιτητή (Αρ. Μητρώου)	Φώτιος Τσιούμας (ΜΠΠΛ21079)
Εξάμηνο - Ακαδημαϊκό Έτος	Εαρινό Εξάμηνο (4ο) – 2022/23

Η εργασία μου αποτελεί μια εφαρμογή για ραντεβού μεταξύ γιατρών και ασθενών. Οι τεχνολογίες και τα πακέτα που έχουν χρησιμοποιηθεί στην εφαρμογή είναι τα παρακάτω:

- Spring Boot (Spring Web, Spring Data JPA, Spring Security, Lombok, I/O Validation)
- MySQL
- Thymeleaf
- Bootstrap
- CSS

Οι χρήστες στους οποίους απευθύνεται η εφαρμογή είναι οι γιατροί και οι ασθενείς και έχουν τις παρακάτω δυνατότητες:

- Ο user ως **γιατρός**:
 - Εγγραφή/Σύνδεση/Αποσύνδεση
 - Δημιουργία νέου ραντεβού
 - Προβολή/Διαγραφή των ραντεβού του
 - Προβολή/Επεξεργασία των στοιχείων του
- Ο user ως **ασθενής**:
 - Εγγραφή/Σύνδεση/Αποσύνδεση
 - Κλείσιμο ραντεβού από τα διαθέσιμα
 - Προβολή/Ακύρωση των ραντεβού του
 - Προβολή/Επεξεργασία των στοιχείων του

Οι οντότητες (**entities**) της εφαρμογής είναι τρεις:

- **User**
- **Role**
- **Appointment**

Από αυτές τις οντότητες δημιουργούνται τέσσερις πίνακες:

- Οι **τρεις πίνακες** είναι ένας για κάθε οντότητα.
- Ο **τέταρτος** πίνακας δημιουργείται από την **σχέση many to many** μεταξύ **user** και **role**.

Για τις παραπάνω οντότητες έχουν δημιουργηθεί επιπλέον:

- Από ένα **repository** για την κάθε οντότητα, που κάνει implement το interface JpaRepository.
 - Στην περίπτωση του **role repository** έχουμε μια επιπλέον custom μέθοδο για την εύρεση role by name.
 - Στην περίπτωση του **user repository** έχουμε μια επιπλέον custom μέθοδο για την εύρεση user by email και μια ακόμα για update του user με custom query.
- Από ένα **service** για την κάθε οντότητα όπου με dependency injection των αντίστοιχων repositories, υλοποιεί το business logic, δηλαδή της μεθόδους για crud operations.
 - Στην περίπτωση του **user** υπάρχει ένα ακόμα service, όπου κάνει implement το interface UserDetailsService του Spring Security.
- Από ένας **controller** για τις οντότητες **user** και **appointment**, όπου διαχειρίζεται όλα τα http request για επικοινωνία μεταξύ του backend με το frontend.
- Δύο **config files**, ένα για το **web security** των urls της εφαρμογής και ένα για την δημιουργία **seed data** κατά την πρώτη φόρτωση της εφαρμογής.

```

▼ src
  ▼ main
    ▼ java
      ▼ unipi.fotistsiou.doctorappointment
        ▼ config
          ● SeedDataConfig
          ● WebSecurityConfig
        ▼ controller
          ● AppointmentController
          ● UserController
        ▼ entity
          ● Appointment
          ● Role
          ● User
        ▼ repository
          ● AppointmentRepository
          ● RoleRepository
          ● UserRepository
        ▼ service
          ● AppointmentService
          ● RoleService
          ● UserDetailsServiceImpl
          ● UserService

```

Για το **frontend** μέρος της εφαρμογής έχει χρησιμοποιηθεί:

- **Thymeleaf** για τη δημιουργία των templates.
- **Bootstrap** και **CSS** για την μορφοποίηση των templates.
- **Images** για πιο ευχάριστη παρουσίαση των φορμών.

Όλα τα παραπάνω βρίσκονται στα resources του project όπου εκεί υπάρχει και ένα file με το schema της βάσης και κάποια samples data.

