

Compte Rendu



REALISER PAR :

BENABDESSELAM AYMEN

G22 EMSI ORANGER

ENCADRE PAR:

MOHAMED YOUSSEFI

1- Cahier des charges :

Concernant le prochain contrôle JEE

Boîte de réception x



Mohamed Youssfi <med@yousfi.net>

mer. 13 avr. 12:42



À isaacbikwaku2001, ismail.lahbabi17, elalaouiomar1998, lahrach6yasser, l.mehdii, adiouaneyoussef, ayman99dakir, bentarghiilyass, boudfoustmohammed, ▾

Bonjour,

Voici les consignes concernant votre prochain contrôle :

- Présenter l'application web JEE Gestion des patients de l'activité pratique N°3 et N°4 avec ses quatre parties :
 - Partie 1 : Rechercher les patients, pagination et suppression des patients
 - Partie 2 : Page template, Ajout des patients, validation des formulaires, édition et mise à jour des patients
 - Partie 3 : Spring Security (Stratégies : InMemoryAuthentication et JDBCAuthentication)
 - Partie 4 : Spring Security (Stratégie : UserDetailsService)
- Présenter des fonctionnalités et améliorations apportées à l'application (Recherche Multi Critère, Ajouter d'autres attributs à la classe Patient, Améliorer le design, corriger quelques imperfections, améliorer la sécurité, etc.)
- Présenter le repository GitHub de votre application ainsi que les différentes branches et commits effectués
- Répondre aux questions que je vous poserai au niveau du code source de votre application
- Bonus : Ajouter la gestion d'autres entités en plus de Patient , comme : Médecin, Rendez-vous, Consultation, etc...)
- Il aussi important de montrer l'état d'avancement de votre projet dans les séances TP avant le contrôle
- Remettre un compte rendu dans un repository GitHub (Code source et Rapport (README.MD ou PDF))

Bon courage à tous

2-Mise en place :

Pour commencer, nous avons tout d'abord choisi les différentes technologies que nous allons utiliser. Premièrement, notre tuteur nous a demandé de réaliser le coeur du projet en Java EE (abrégié en JEE dans la suite du rapport). Nous en avons profité pour utiliser le framework Spring qui permet de s'affranchir de la gestion des servlets ou encore de la gestion direct des bases de données (couche DAO, requêtes SQL brutes) en utilisant un framework de persistance des objets, Hibernate dans notre cas. Nous avons également choisi d'utiliser Spring Boot qui permet de s'affranchir des problèmes de dépendances et d'automatiser la création des .jar Pour gérer la sécurité de notre site internet et les types d'utilisateurs, nous utiliserons Spring Security, un autre framework. Puis pour gérer la communication entre le front et le back, nous avons choisi le moteur de template Thymeleaf possédant une syntaxe plus souple que les jsp traditionnelles. Celui-ci va permettre d'afficher à l'utilisateur des informations venant de la partie back-office, comme des valeurs persistées en base de données par exemple. Ensuite, pour la partie front du site, c'est Bootstrap que nous utiliserons comme en association avec HTML 5, CSS 3 et JavaScript. Bootstrap est un framework qui permet de faciliter la mise en page de notre site et d'uniformiser le rendu. Son principale intérêt réside dans son système de grille particulièrement modulaire et responsive, c'est à dire s'adaptant à tous les types d'appareils.

REALISER PAR :

ENCADRE PAR:

BENABDESSELAM AYMEN

MOHAMED YOUSSEFI

G22 EMSI ORANGER

Il fournit également un ensemble de composants restylés pour faire plaisir aux yeux. Enfin, pour la programmation nous avons décidé d'utiliser l'IDE IntelliJ que nous avons déjà utilisé en cours et qui va nous permettre de ne pas avoir à nous soucier du conteneur web Apache Tomcat qui va accueillir notre site. Il offre également tout un ensemble d'outils tels que l'autocomplétion ou la structuration du projet.

Pour finir, nous tenons à jour un projet GIT.

REALISER PAR :

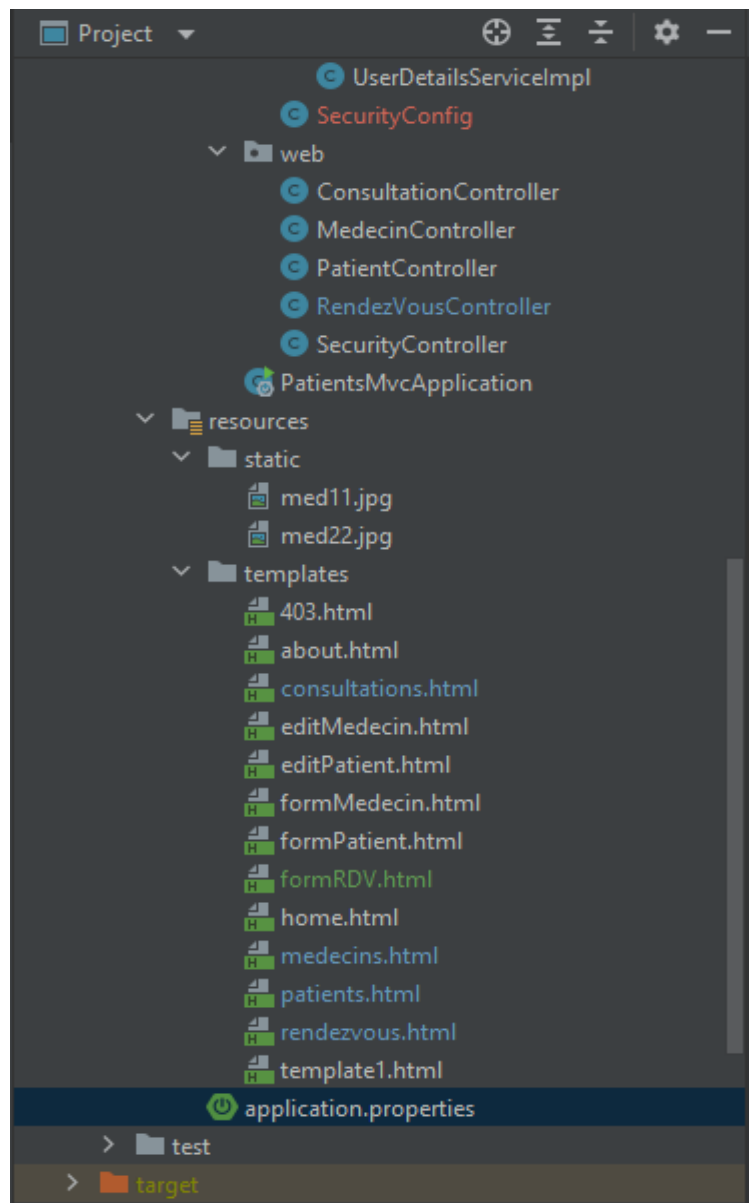
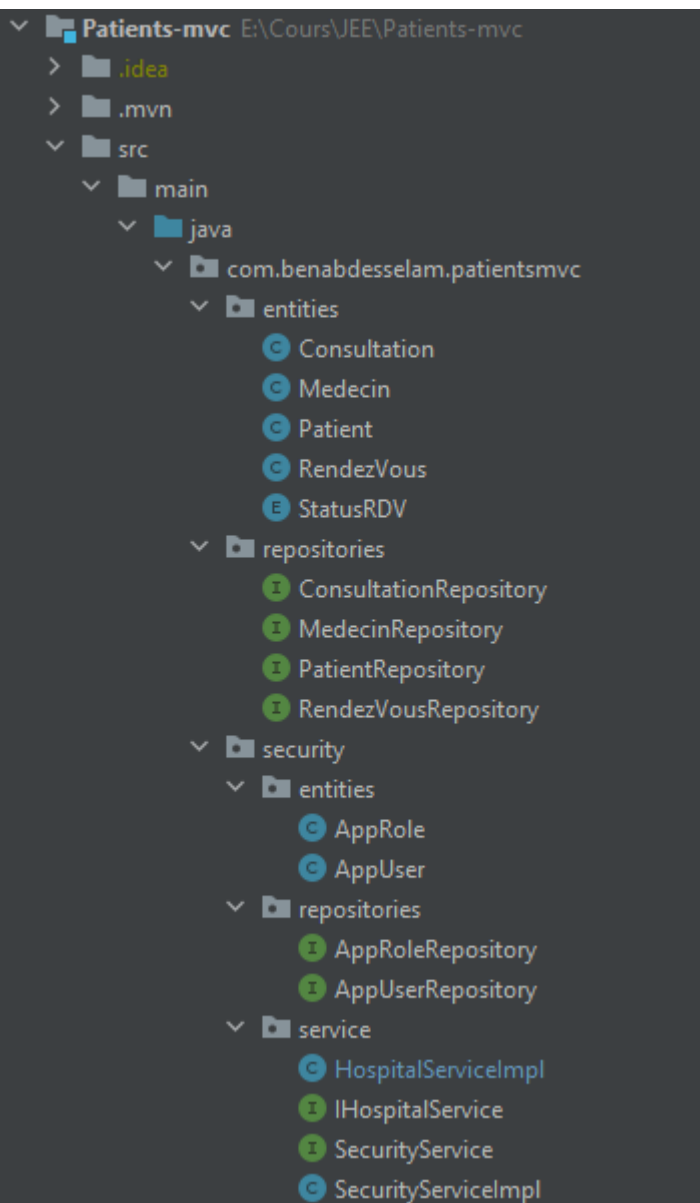
BENABDESSELAM AYMEN

G22 EMSI ORANGER

ENCADRE PAR:

MOHAMED YOUSSEFI

3-Structure du projet :



REALISER PAR :

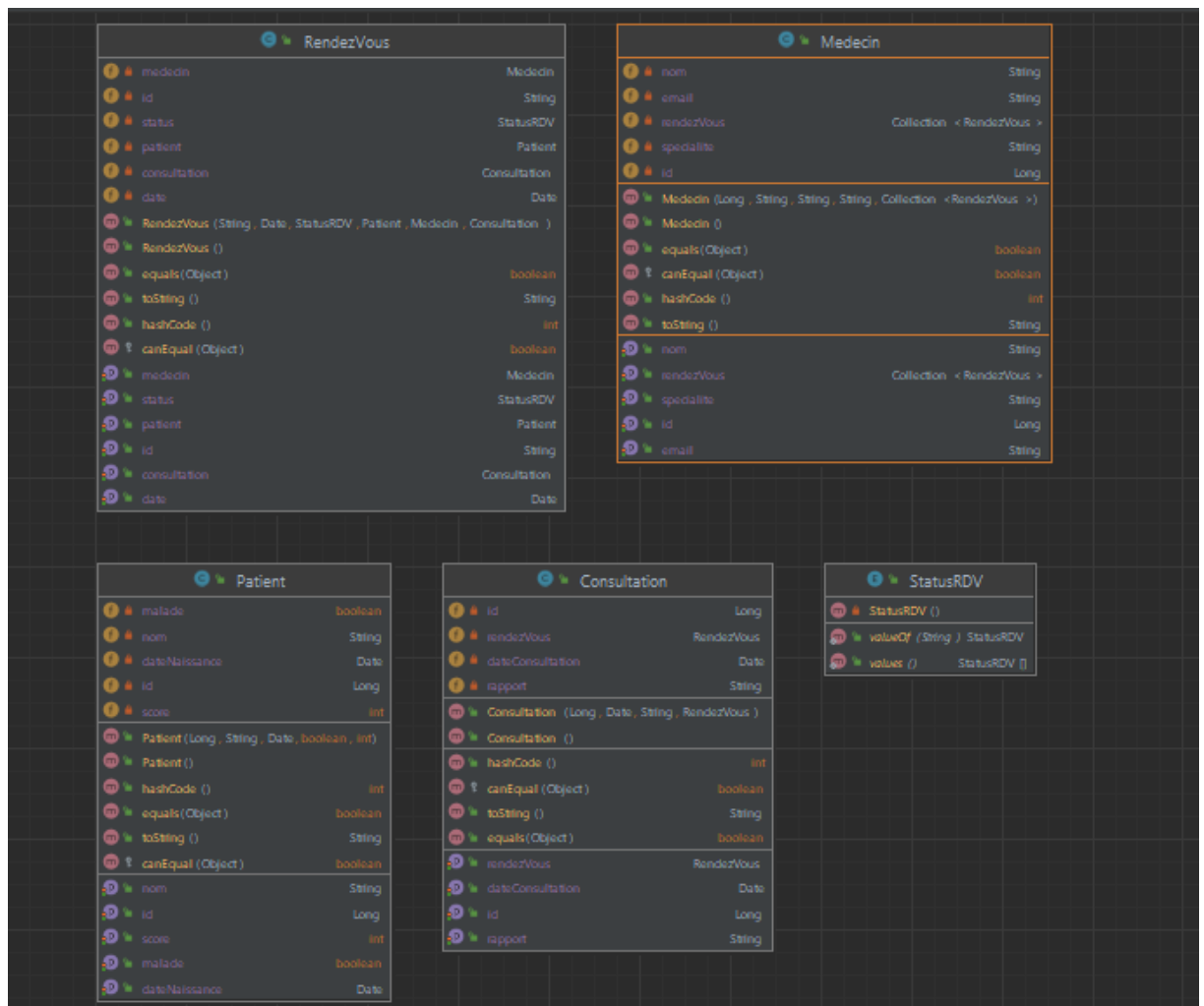
BENABDESSELAM AYMEN

G22 EMSI ORANGER

ENCADRE PAR:

MOHAMED YOUSSEFI

4-Les entités :



REALISER PAR :

BENABDESSELAM AYMEN

G22 EMSI ORANGER

ENCADRE PAR:

MOHAMED YOUSSEFI

1. Code source :

Le lien du code source : <https://github.com/Benabdesselam/Patients-mvc>

Capture d'écran

a. Capture du code source :

Entité Patient :

```
package com.benabdesselam.patientsmvc.entities;

import ...

@Entity
@Data @AllArgsConstructor @NoArgsConstructor
public class Patient {
    @Id @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;
    @NotEmpty
    @Size(min = 4,max = 20)
    private String nom;
    @Temporal(TemporalType.DATE)
    @DateTimeFormat(pattern = "yyyy-MM-dd")
    private Date dateNaissance;
    private boolean malade ;
    @DecimalMin("10")
    private int score;
}
```

Entité Médecin

```
package com.benabdesselam.patientsmvc.entities;

import ...
@Entity
@Data @NoArgsConstructor @AllArgsConstructor
public class Medecin {
    @Id @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;
    private String nom;
    private String email;
    private String specialite;
    @OneToMany(mappedBy = "medecin",fetch = FetchType.LAZY)
    @JsonProperty(access = JsonProperty.Access.WRITE_ONLY)
    private Collection<RendezVous> rendezVous;
}
```

REALISER PAR :

BENABDESSELAM AYMEN

G22 EMSI ORANGER

ENCADRE PAR:

MOHAMED YOUSSEFI

Entité Rendez-vous

```
package com.benabdesselam.patientsmvc.entities;

import ...
@Entity
@Data
@NoArgsConstructor @AllArgsConstructor

public class RendezVous {
    @Id
    private String id;
    private Date date;
    @Enumerated(EnumType.STRING)
    private StatusRDV status;
    @ManyToOne
    @JsonProperty(access = JsonProperty.Access.WRITE_ONLY)
    private Patient patient;
    @ManyToOne
    private Medecin medecin;
    @OneToOne(mappedBy = "rendezVous")
    private Consultation consultation;
}
```

REALISER PAR :

BENABDESSELAM AYMEN

G22 EMSI ORANGER

ENCADRE PAR:

MOHAMED YOUSSEFI

Entité Consultation

```
package com.benabdesselam.patientsmvc.entities;

import ...
@Entity
@Data
@NoArgsConstructor
@AllArgsConstructor
public class Consultation {
    @Id @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;
    private Date dateConsultation;
    private String rapport;
    @OneToOne
    @JsonProperty(access = JsonProperty.Access.WRITE_ONLY)
    private RendezVous rendezVous;
}
```

REPOSITORIES :

Patient

```
public interface PatientRepository extends JpaRepository<Pat, Long> {
    Page<Pat> findByNomContains(String kw, Pageable pageable);
    // Pat findByNom(String kw, Pageable pageable);
}
```

Médecin

```
public interface MedecinRepository extends JpaRepository<Medecin, Long> {
    Page<Medecin> findByNomContains(String kw, Pageable pageable);
}
```

REALISER PAR :

BENABDESSELAM AYMEN

G22 EMSI ORANGER

ENCADRE PAR:

MOHAMED YOUSSEFI

Rendez-vous

```
@Repository
public interface RendezVousRepository extends JpaRepository<RendezVous,Long> {
    @Override
    List<RendezVous> findAll();
    //Optional<RendezVous> findById(Long id);
    Page<RendezVous> findByPatientNomContains(String kw,Pageable page);
    //Page<RendezVous> findByDate(Date date,Pageable page);
}
```

Consultation

```
public interface ConsultationRepository extends JpaRepository<Consultation,Long> {
    Consultation findByRendezVousId(Long id);
}
```

PATIENT Controller

```
@Controller
@AllArgsConstructor
public class PatientController {
    private PatientRepository patientRepository;

    @GetMapping(path = "/user/index")
    public String patients(
        Model model,
        @RequestParam(name = "page",defaultValue = "0") int page,
        @RequestParam(name = "size",defaultValue = "5")int size,
        @RequestParam(name = "keyword",defaultValue = "")String keyword){
        Page<Pat> pagePatients = patientRepository.findByNomContains(keyword,PageRequest.of(page,size));
        model.addAttribute( attributeName: "listPatients", pagePatients.getContent());
        model.addAttribute( attributeName: "pages",new int[pagePatients.getTotalPages()]);
        model.addAttribute( attributeName: "currentPage",page);
        model.addAttribute( attributeName: "keyword",keyword);

        return "patients";
    }

    @GetMapping("/admin/delete")
    public String delete(Long id,String keyword,String page){
        patientRepository.deleteById(id);
        return "redirect:/user/index?page="+page+"&keyword="+keyword;
    }

    @GetMapping("/")
    public String home(){ return "home"; }
}
```

REALISER PAR :

BENABDESSELAM AYMEN

G22 EMSI ORANGER

ENCADRE PAR:

MOHAMED YOUSSEFI

Médecin Controller

```
package com.benabdeslam.patientsmvc.web;

import ...

@Controller
@AllArgsConstructor
public class MedecinController {

    MedecinRepository medecinRepository;
    @Autowired
    @GetMapping(path = "/user/medecins")
    @ResponseBody
    public List<Medecin> listMedecins() { return medecinRepository.findAll(); }

    @GetMapping(path = "/user/talaba")
    public String medecins(Model model,
        @RequestParam(name = "page",defaultValue = "0") int page ,
        @RequestParam(name = "size",defaultValue = "5") int size,
        @RequestParam(name = "keyword",defaultValue = "") String keyword){
        Page<Medecin> pageMedecins = medecinRepository.findByNomContains(keyword,PageRequest.of(page,size));
        model.addAttribute( attributeName: "listMedecins",pageMedecins.getContent());
        model.addAttribute( attributeName: "pages",new int[pageMedecins.getTotalPages()] );
        model.addAttribute( attributeName: "currentPage",page);
        model.addAttribute( attributeName: "keyword",keyword);
        return "medecins";
    }
    @GetMapping(path = "/admin/deleteMedecin")
```

Rendez vous

```
@Controller
@AllArgsConstructor
public class RendezVousController {
    private RendezVousRepository rendezVousRepository;
    private PatientRepository patientRepository;
    private MedecinRepository medecinRepository;

    @GetMapping("/user/rendezvous")
    public String rendezvous(Model model, @RequestParam(name="page",defaultValue = "0") int page,
        @RequestParam(name="size",defaultValue = "5") int size,
        @RequestParam(name="keyword",defaultValue = "") String keyword
    ){
        Page<RendezVous> rendezvous = rendezvousRepository.findByPatientNomContains(keyword,PageRequest.of(page,size));
        if (rendezvous == null) throw new RuntimeException("*****null");
        model.addAttribute( attributeName: "listRendezvous",rendezvous);
        model.addAttribute( attributeName: "pages",new int[rendezvous.getTotalPages()]);
        model.addAttribute( attributeName: "currentPage",page);
        model.addAttribute( attributeName: "keyword",keyword);

        return "rendezvous";
    }

    @GetMapping("/admin/rendezvous/delete")
    public String delete(Long id,String keyword,String page){
        rendezvousRepository.deleteById(id);
        return "redirect:/user/rendezvous?page="+page+"&keyword="+keyword;
    }
}
```

REALISER PAR :

BENABDESSELAM AYMEN

G22 EMSI ORANGER

ENCADRE PAR:

MOHAMED YOUSSEFI

Consultation Controller

```
package com.benabdeslam.patientsmvc.web;

import ...

@Controller
@AllArgsConstructor
public class ConsultationController {
    private ConsultationRepository consultationRepository;

    @GetMapping(path = "/user/listConsultations")
    @ResponseBody
    public List<Consultation> listConsultations() { return consultationRepository.findAll(); }
    @GetMapping(path = "/user/consult")
    public String consultations(Model model){
        // Page<Consultation> pageConsultations = consultationRepository.findByDateConsultation(date, PageRec
        model.addAttribute( attributeName: "listConsultations" ,listConsultations());
        // model.addAttribute("pages",new int[pageConsultations.getTotalPages()] );
        // model.addAttribute("currentPage",page);
        // model.addAttribute("date",date);
        return "consultations";
    }
}
```

Security Controller

```
package com.benabdeslam.patientsmvc.web;

import ...

@Controller
public class SecurityController {

    @GetMapping("/403")
    public String notAuthorized() { return "403"; }
}
```

REALISER PAR :

BENABDESSELAM AYMEN

G22 EMSI ORANGER

ENCADRE PAR:

MOHAMED YOUSSEFI

REALISER PAR :

BENABDESSELAM AYMEN

G22 EMSI ORANGER

ENCADRE PAR:

MOHAMED YOUSSEFI