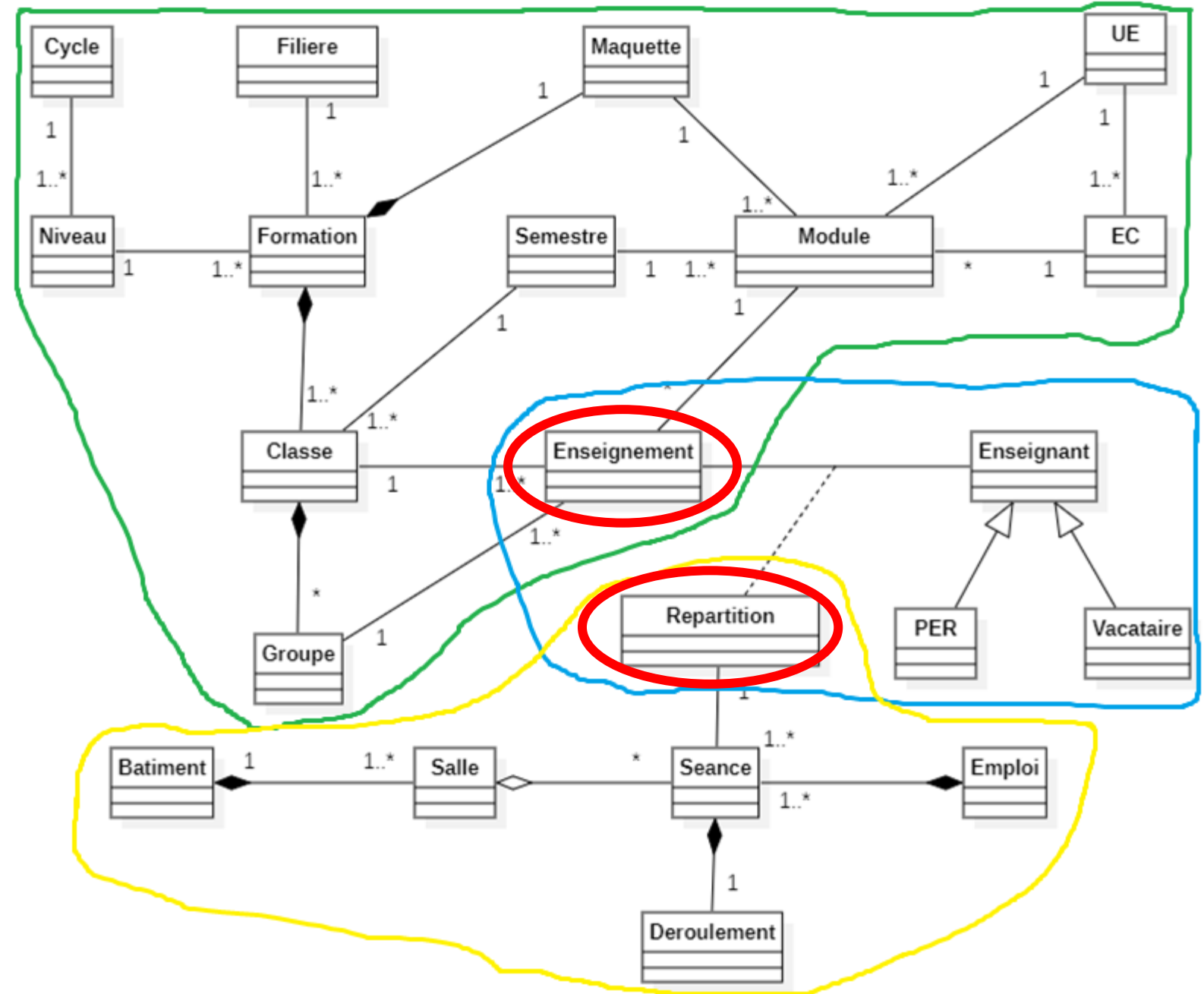


DAOS

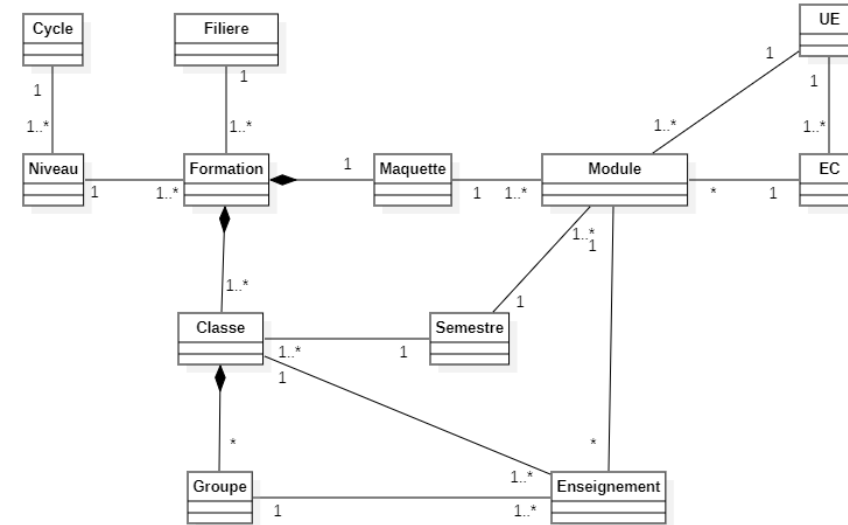
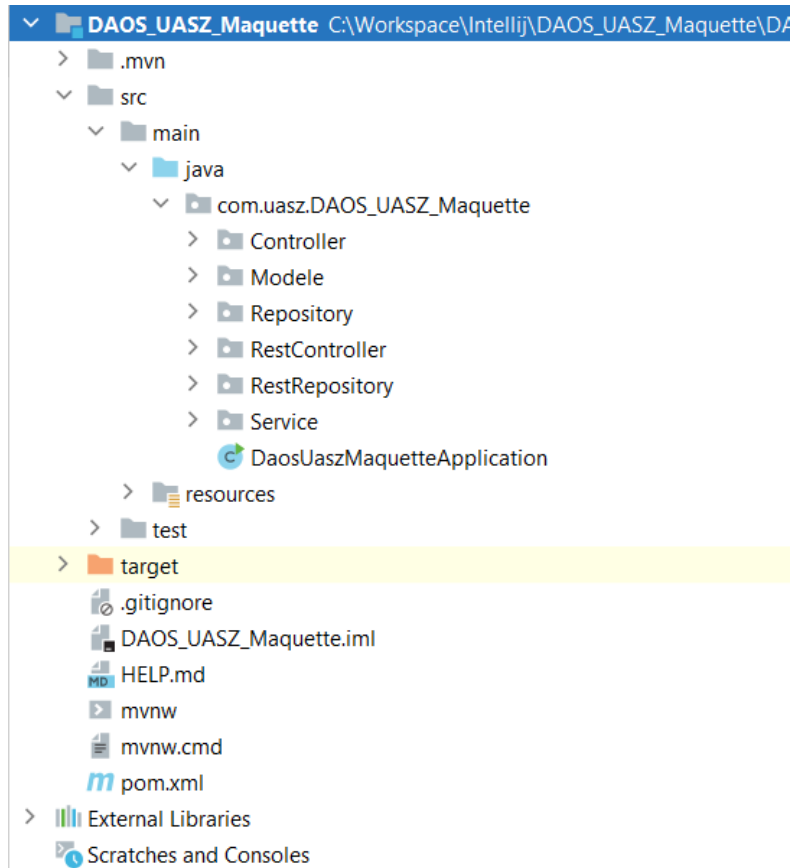
■ package Maquette

■ package Répartition

■ package Emploi du temps

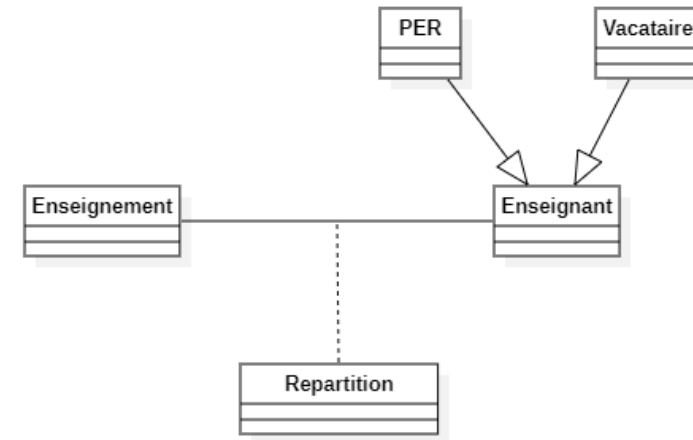
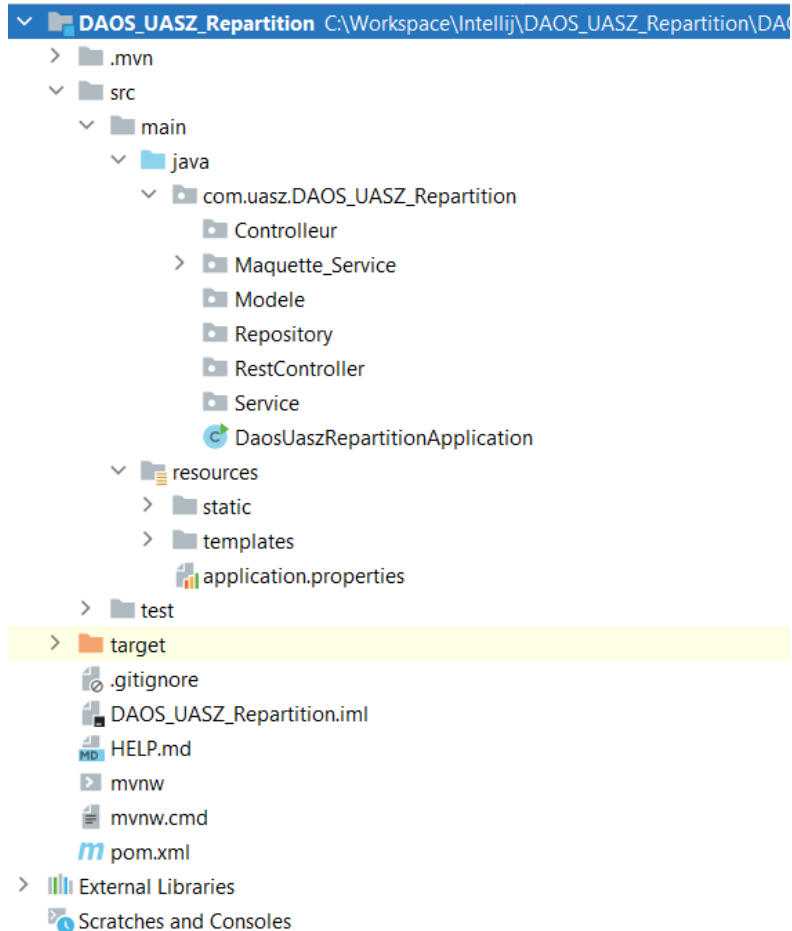


Micro service Maquette



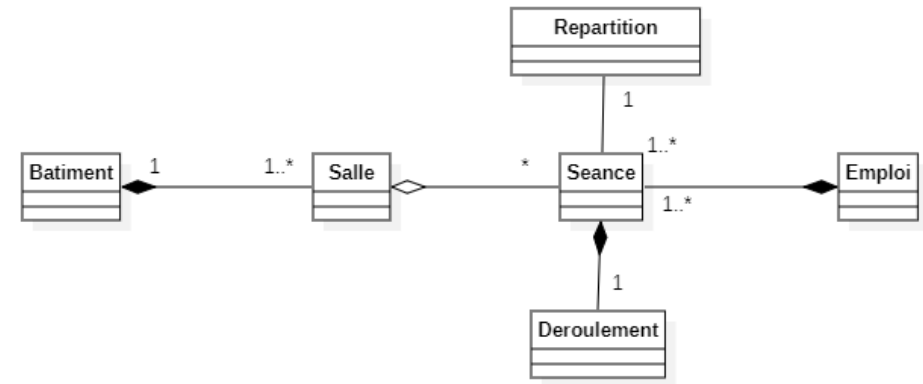
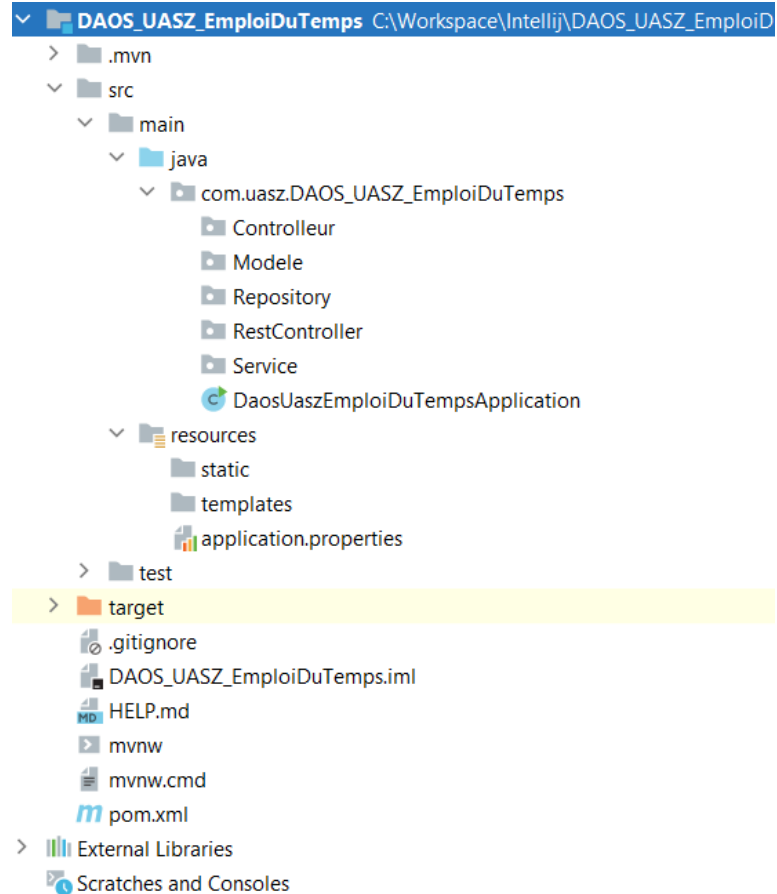
```
spring.application.name=maquette-service
server.port=8081
spring.datasource.driverClassName=org.h2.Driver
spring.datasource.url=jdbc:h2:mem:maquette-db
#SSSSpring.jpa.hibernate.ddl-auto=update
spring.jpa.show-sql=true
spring.h2.console.enabled=true
spring.h2.console.path=/h2
```

Micro service Répartition



```
spring.application.name=repartition-service
server.port=8082
spring.datasource.driverClassName=org.h2.Driver
spring.datasource.url=jdbc:h2:mem:repartition-db
#SSSspring.jpa.hibernate.ddl-auto=update
spring.jpa.show-sql=true
spring.h2.console.enabled=true
spring.h2.console.path=/h2
```

Micro service EmploiDuTemps



```
spring.application.name=emploiDuTemps-service
server.port=8083
spring.datasource.driverClassName=org.h2.Driver
spring.datasource.url=jdbc:h2:mem:emploiDuTemps-db
#SSSSpring.jpa.hibernate.ddl-auto=update
spring.jpa.show-sql=true
spring.h2.console.enabled=true
spring.h2.console.path=/h2
```

Feign Client

```
<dependency>  
  <groupId>org.springframework.cloud</groupId>  
  <artifactId>spring-cloud-starter-openfeign</artifactId>  
</dependency>
```

```
<dependencyManagement>  
  <dependencies>  
    <dependency>  
      <groupId>org.springframework.cloud</groupId>  
      <artifactId>spring-cloud-dependencies</artifactId>  
      <version>${spring-cloud.version}</version>  
      <type>pom</type>  
      <scope>import</scope>  
    </dependency>  
  </dependencies>  
</dependencyManagement>
```

Ajouter ces dépendances dans le pom.xml de chaque micro service

MS Maquette

```

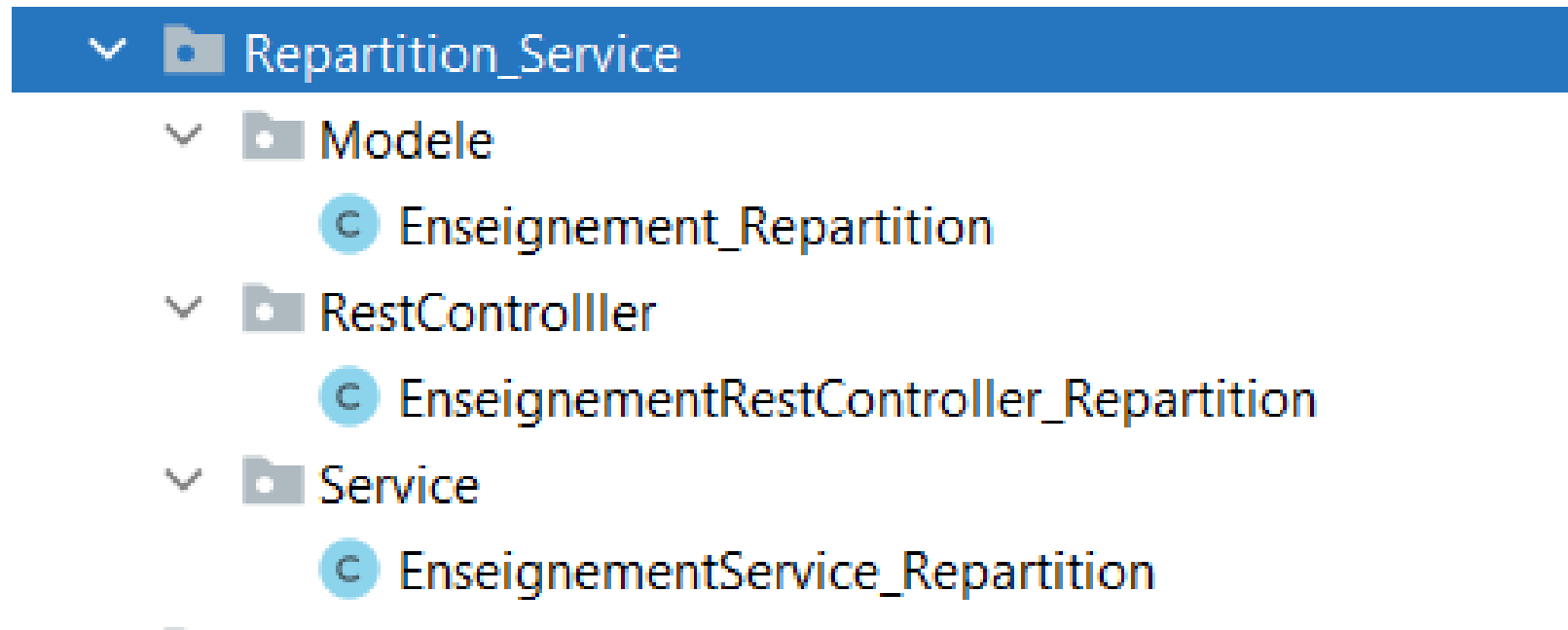
▼ DAOS_UASZ_Maquette C:\Workspace\Intellij\DAOS_
  > .mvn
  ▼ src
    ▼ main
      ▼ java
        ▼ com.uaszdAOS_UASZ_Maquette
          > Authentication
          > Controller
          > Modele
          > Repartition_Service
          > Repository
          > RestController
          > RestRepository
          > Service
          DaosUaszMaquetteApplication
        ▼ resources
          > static
          > templates
          application.properties
      > test

```

Modele : Enseignement

```
@Entity
@Data @NoArgsConstructor @AllArgsConstructor
public class Enseignement {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;
    private String libelle;
    private String description;
    @ManyToOne
    private Module module;
    @ManyToOne
    private Classe classe;
    @ManyToOne
    private Groupe groupe;
}
```


Répartition_Service dans le MS Maquette



Modele : Enseignement_Repartition

```
@Data
@NoArgsConstructor
@AllArgsConstructor
public class Enseignement_Repartition {
    private Long id;
    private String libelle;
    private String description;
    private Long idModule;
    private String cours;
    private int coefficient;
    private int nbreHeure;
    private Long idClasse;
    private String classe;
    private Long idGroupe;
    private String groupe;
    private String semestre;
}
```

Service : Enseignement_Repartition

```
@Service
public class EnseignementService_Repartition {
    1 usage
    @Autowired
    private EnseignementService enseignementService;
    2 usages
    public List<Enseignement_Repartition> lister_Enseignement_Repartition(){
        List<Enseignement> enseignements=enseignementService.lister_Enseignement();
        List<Enseignement_Repartition> enseignement_repartitions= new ArrayList<>();
        for (int i=0;i<enseignements.size();i++){
            Enseignement_Repartition enseignement_repartition=new Enseignement_Repartition();
            enseignement_repartition.setId(enseignements.get(i).getId());
            enseignement_repartition.setLibelle(enseignements.get(i).getLibelle());
            enseignement_repartition.setDescription(enseignements.get(i).getDescription());
            enseignement_repartition.setIdModule(enseignements.get(i).getModule().getId());
            enseignement_repartition.setCours(enseignements.get(i).getModule().getCours());
            enseignement_repartition.setCoefficient(enseignements.get(i).getModule().getCoefficient());
            enseignement_repartition.setNbreHeure(enseignements.get(i).getModule().getNbreHeure());
            enseignement_repartition.setIdClasse(enseignements.get(i).getClasse().getId());
            enseignement_repartition.setClasse(enseignements.get(i).getClasse().getLibelle());
            if(enseignements.get(i).getGroupe() != null){
                enseignement_repartition.setIdGroupe(enseignements.get(i).getGroupe().getId());
                enseignement_repartition.setGroupe(enseignements.get(i).getGroupe().getLibelle());
            }
            enseignement_repartition.setSemestre(enseignements.get(i).getClasse().getSemestre().getLibelle());
            enseignement_repartitions.add(enseignement_repartition);
        }
        return enseignement_repartitions;
    }
}
```

RestController : Enseignement_Repartition

```
@CrossOrigin(origins = "*", allowedHeaders = "*")
@RestController
@RequestMapping("/maquette/api")
public class EnseignementRestController_Repartition {

    1 usage
    @Autowired
    private EnseignementService_Repartition enseignementService_repartition;

    @GetMapping(path = "/enseignements_Repartitions")
    public List lister_Enseignement(){
        return enseignementService_repartition.lister_Enseignement_Repartition();
    }
}
```

Le fichier Run du MS Maquette

```
Classe semestre_5_L3_L2I = classeRepository.save(new Classe( id: null, libelle: "Licence 3 L2I Semestre 5", effectif: 30, nbreGroupe: 1, description: null, l3_L2I, semestre_5, groupe));
Classe semestre_6_L3_L2I = classeRepository.save(new Classe( id: null, libelle: "Licence 3 L2I Semestre 6", effectif: 30, nbreGroupe: 1, description: null, l3_L2I, semestre_6, groupe));
Classe semestre_3_L2_MPI = classeRepository.save(new Classe( id: null, libelle: "Licence 2 MPI Semestre 3", effectif: 70, nbreGroupe: 2, description: null, l2_MPI, semestre_3, groupe));
Classe semestre_4_L2_MI = classeRepository.save(new Classe( id: null, libelle: "Licence 2 MI Semestre 4", effectif: 30, nbreGroupe: 1, description: null, l2_MPI, semestre_4, groupes));
Classe semestre_4_L2_MP = classeRepository.save(new Classe( id: null, libelle: "Licence 2 MP Semestre 4", effectif: 30, nbreGroupe: 1, description: null, l2_MPI, semestre_4, groupes));

UE ue_1 = ueRepository.save(new UE( id: null, libelle: "Reseaux et Telecoms", code: "INF351", credit: 8, coefficient: 4, description: "", new Date(), ecs: null));
UE ue_2 = ueRepository.save(new UE( id: null, libelle: "Genie Logiciel 1", code: "INF352", credit: 8, coefficient: 4, description: "", new Date(), ecs: null));
UE ue_3 = ueRepository.save(new UE( id: null, libelle: "Technologies embarques et Mobiles", code: "INF353", credit: 8, coefficient: 4, description: "", new Date(), ecs: null));
UE ue_4 = ueRepository.save(new UE( id: null, libelle: "Gestions de donnees structurees", code: "INF354", credit: 8, coefficient: 4, description: "", new Date(), ecs: null));
UE ue_5 = ueRepository.save(new UE( id: null, libelle: "Humanites et Entreprise", code: "INF355", credit: 6, coefficient: 3, description: "", new Date(), ecs: null));

Module module_1 = moduleRepository.save(new Module( id: null, libelle: "Genie Logiciel 1", cours: "CM", coefficient: 2, nbreHeure: 2, description: null, objectifs: null, semestre_5, maquette));
Module module_2 = moduleRepository.save(new Module( id: null, libelle: "Genie Logiciel 1", cours: "TD", coefficient: 2, nbreHeure: 2, description: null, objectifs: null, semestre_5, maquette));
Module module_3 = moduleRepository.save(new Module( id: null, libelle: "Reseaux et Telecoms", cours: "CM", coefficient: 2, nbreHeure: 2, description: null, objectifs: null, semestre_5, maquette));
Module module_4 = moduleRepository.save(new Module( id: null, libelle: "Reseaux et Telecoms", cours: "TD", coefficient: 2, nbreHeure: 2, description: null, objectifs: null, semestre_5, maquette));

enseignementRepository.save(new Enseignement( id: null, libelle: "Genie Logiciel 1", description: null, module_1, semestre_5_L3_L2I, groupe: null));
enseignementRepository.save(new Enseignement( id: null, libelle: "Genie Logiciel 1", description: null, module_2, semestre_5_L3_L2I, groupe: null));
enseignementRepository.save(new Enseignement( id: null, libelle: "Reseaux et Telecoms", description: null, module_3, semestre_5_L3_L2I, groupe: null));
enseignementRepository.save(new Enseignement( id: null, libelle: "Reseaux et Telecoms", description: null, module_4, semestre_5_L3_L2I, groupe: null));
```

MS Maquette : Après exécution

SELECT * FROM ENSEIGNEMENT;

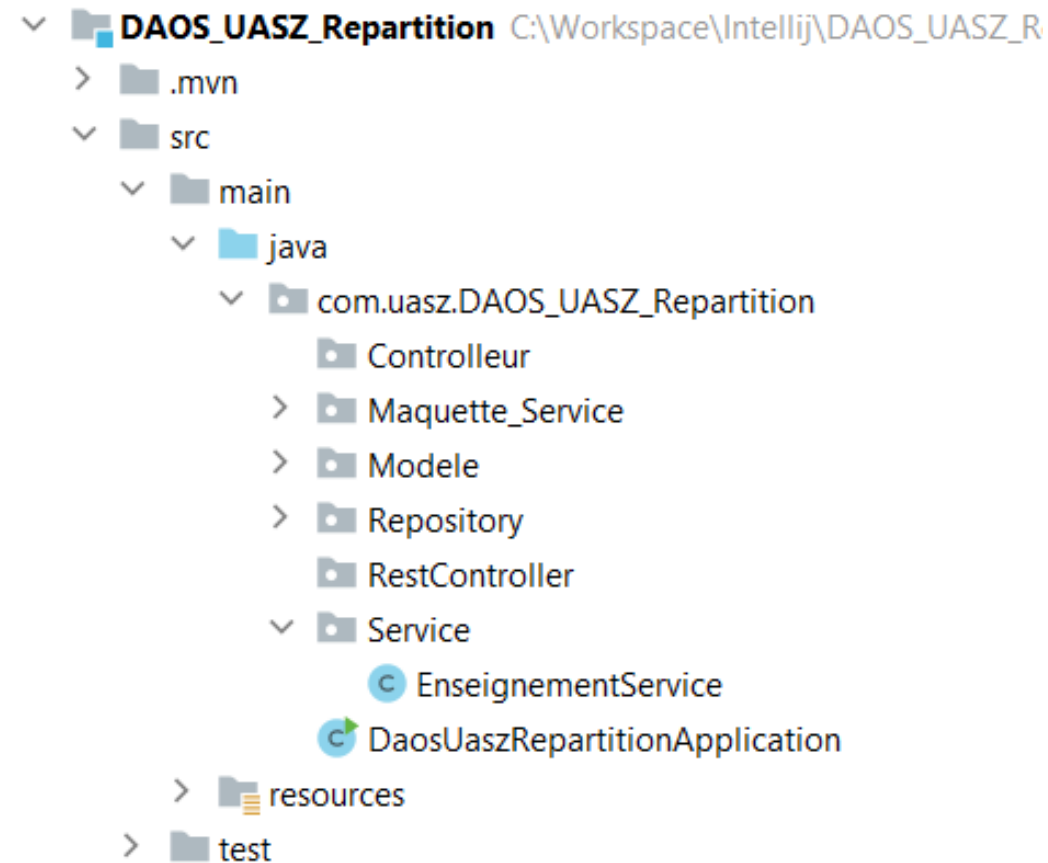
| ID | DESCRIPTION | LIBELLE | CLASSE_ID | GROUPE_ID | MODULE_ID |
|----|-------------|---------------------|-----------|-------------|-----------|
| 1 | <i>null</i> | Genie Logiciel 1 | 1 | <i>null</i> | 1 |
| 2 | <i>null</i> | Genie Logiciel 1 | 1 | <i>null</i> | 2 |
| 3 | <i>null</i> | Reseaux et Telecoms | 1 | <i>null</i> | 3 |
| 4 | <i>null</i> | Reseaux et Telecoms | 1 | <i>null</i> | 4 |

Enseignement

```
{
  "id": 1,
  "libelle": "Genie Logiciel 1",
  "description": null,
  "idModule": 1,
  "cours": "CM",
  "coefficient": 2,
  "nbreHeure": 2,
  "idClasse": 1,
  "classe": "Licence 3 L2I Semestre 5",
  "idGroupe": null,
  "groupe": null,
  "semestre": "Semestre 5"
},
```

Enseignement_Repartition

MS Repartition



Modele : Enseignement

```
@Entity
@Data @NoArgsConstructor @AllArgsConstructor
public class Enseignement {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;
    private String libelle;
    private String description;
    private Long idModule;
    private String cours;
    private int coefficient;
    private int nbreHeure;
    private Long idClasse;
    private String classe;
    private Long idGroupe;
    private String groupe;
    private String semestre;
    @OneToMany(mappedBy = "enseignement", fetch=FetchType.LAZY)
    private Collection<Repartition> repartitions;
}
```


Service : EnseignementService

```
@Service
@Transactional
public class EnseignementService {

    5 usages
    @Autowired
    private EnseignementRepository enseignementRepository;

    public List<Enseignement> lister_Enseignement(){ return enseignementRepository.findAll(); }

    public Enseignement rechercher_Enseignement(Long id){ return enseignementRepository.findById(id).get();}






    1 usage
    public Enseignement ajouter_Enseignement(Enseignement enseignement){ return enseignementRepository.save(enseignement);}

    public Enseignement modifier_Enseignement(Enseignement enseignement, Long id){
        enseignement.setId(id);
        return enseignementRepository.save(enseignement);
    }

    public void supprimer_Enseignement(Long id){ enseignementRepository.deleteById(id); }

}
```

Maquette_Service dans le MS Repartition

- ▼  Maquette_Service
 - >  Controller
 - ▼  Interface
 -  MaquetteProxy
 - >  Modele

Interface : MaquetteProxy

```
@FeignClient(name = "maquette-service", url = "localhost:8080")
public interface MaquetteProxy {

    2 usages
    @GetMapping(path = "/maquette/api/enseignements_Repartitions")
    List<Enseignement> lister_enseignements_Repartitions();

    1 usage
    @GetMapping(path = "/maquette/api/ues")
    List<UE> lister_ue();

    @GetMapping(path = "/maquette/api/ue/{id}")
    UE rechercher_ue(@PathVariable Long id);

}
```

Le fichier Run du MS Repartition

3 usages

```
private final MaquetteProxy maquetteProxy;
```

1 usage

```
@Autowired
```

```
private EnseignementService enseignementService;
```

```
@Override
```

```
public void run(String... args) throws Exception {  
    System.out.println(maquetteProxy.lister_enseignements_Repartitions());  
    List<Enseignement> enseignements=maquetteProxy.lister_enseignements_Repartitions();  
    for (int i=0;i<enseignements.size();i++){  
        enseignementService.ajouter_Enseignement(enseignements.get(i));  
    }  
}
```

MS Repartition : Après exécution

SELECT * FROM ENSEIGNEMENT;

| ID | CLASSE | COEFFICIENT | COURS | DESCRIPTION | GROUPE | ID_CLASSE | ID_GROUPE | ID_MODULE | LIBELLE | NBRE_HEURE | SEMESTRE |
|----|--------------------------|-------------|-------|-------------|-------------|-----------|-------------|-----------|---------------------|------------|------------|
| 1 | Licence 3 L2I Semestre 5 | 2 | CM | <i>null</i> | <i>null</i> | 1 | <i>null</i> | 1 | Genie Logiciel 1 | 2 | Semestre 5 |
| 2 | Licence 3 L2I Semestre 5 | 2 | TD | <i>null</i> | <i>null</i> | 1 | <i>null</i> | 2 | Genie Logiciel 1 | 2 | Semestre 5 |
| 3 | Licence 3 L2I Semestre 5 | 2 | CM | <i>null</i> | <i>null</i> | 1 | <i>null</i> | 3 | Reseaux et Telecoms | 2 | Semestre 5 |
| 4 | Licence 3 L2I Semestre 5 | 2 | TD | <i>null</i> | <i>null</i> | 1 | <i>null</i> | 4 | Reseaux et Telecoms | 2 | Semestre 5 |

(4 rows, 25 ms)