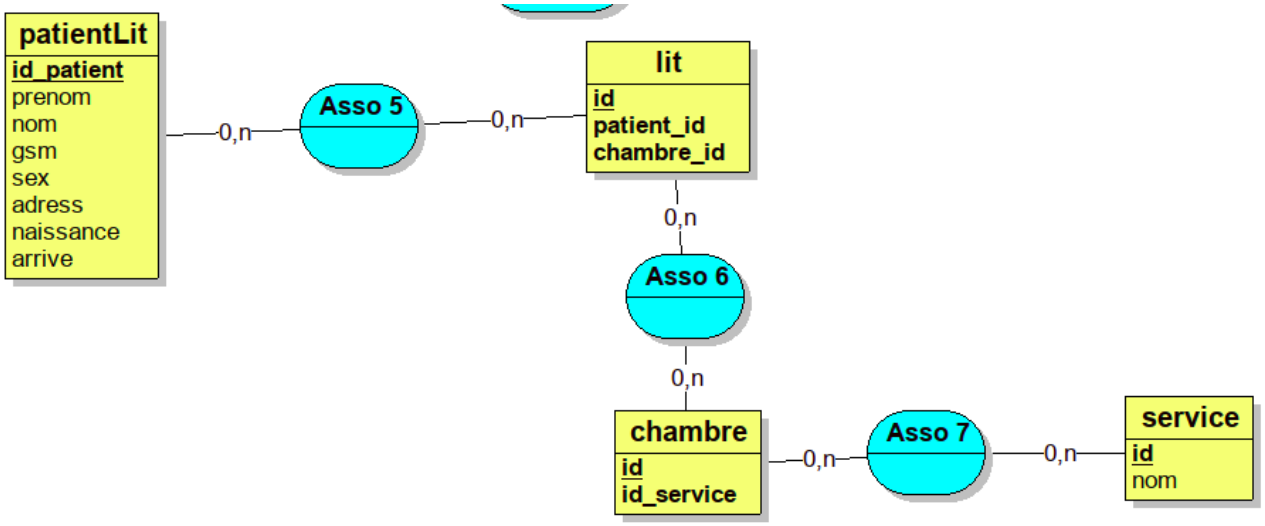
Documentation Technique

MLD-

l'architecture de la base de donnée



1- Service Controller

1.1) La première requête à la base de donnée va interroger la base de donnée et lui demande de lui renvoyer toute les données dans la table services.

La deuxième requête demande à la base de donnée de renvoyer toutes les données de la table chambre.

```
$services = $connection->fetchAll('SELECT * FROM services');

$chambres = $connection->fetchAll('SELECT * FROM chambre');
```

1.2) Les données du controller sont récupéré par le langage Twig. Par services un container est créer ou le nom du services est insérer dans le titre de la card.

```
{% for service in services %}
<div class="card border-dark mb-3 item" style="max-width: 18rem;">
<div class="card-header">{{service['name']}}</div>
```

1.3)Par services on ajoute les chambres qui sont associé aux services.

Chaque chambres est un bouton qui renvois vers la bonne chambre.

```
{% set indexChambre = 0 %}
 {% for chambre in chambres %}
    {% if chambre['id_service'] == service['id'] %}
    {% set indexChambre = indexChambre +1 %}
    <a href="chambre?numService={{service['id']}}&numChambre={{chambre['id']}}"><button type="
    {% endif %}
 {% endfor %}
```

2 – Chambre Controller

2.1) La première requête récupère le nom et l'id du service ou se trouve l'utilisateur.

La deuxièmes requête récupère toutes les colonnes de la table lit, seulement les lit qui se trouve dans la chambre ou se trouve l'utilisateur.

La troisièmes requête récupère toutes les colonnes des patient et des lits, seulement les patients associé au lit et seulement les lit associé à la chambre où se trouve l'utilisateur.

```
$services = $connection->fetchAll('SELECT name, id FROM services WHERE id = '. $_GET['numServi

$lits = $connection->fetchAll('SELECT * FROM lit WHERE chambre_id = '. $_GET['numChambre'] .''

$patients = $connection->fetchAll('SELECT * FROM patient, lit WHERE patient.id_patient = lit.p
                                   lit.chambre_id = '.$_GET['numChambre'] .'');
```

2.2) Si aucun lits on était créer dans la chambre donnée alors ont initialise la variable lits à -1 pour traiter l'erreur.

```
if (empty($lits)){
    $lits[0]['chambre_id'] = -1;
}
```

2.3) Dans le fichier template, 5 container qui représente les lits sont créer.

```
{% for i in 0..4 %}

<div id="lit{{loop.index}}">
    <h2>Lit n°{{loop.index}}</h2>
```

2.4) On vérifie qu'il y a bien des patients dans la chambre donné.

On traduit les valeurs en donnée compréhensible et on affiche les différentes information du patient.

```
{% if patient[i] is defined %}
<script>
  sex = {{ patient[i]['sex'] }};
 if (sex == 1) sex = "homme";
 else sex = "femme";

 </script>
 {{patient[i]['nom']}} {{patient[i]['prenom']}}
```

2.5) Deux boutons sont générer par lit,

Le premier permet de supprimer un patient seulement si il y a un patient. Une requête est envoyé sur la route supprimer.

Le deuxième bouton permet de ajouter un patient à un lit, les données de la chambre et du lit sont envoyé à la page ajoutPatient.

```
            <a href="supprimer?patient={{patient[i]['id_patient']}}&numChambre={{numChambre}}&numServi
            <button class="btn btn-danger">Supprimer</button></a>
           {% else %}
            <a href="ajoutpatient?numChambre={{numChambre}}&numService={{service_name[0]['id']}}">
            <button class="btn btn-success">ajouter</button></a>
```

2.6) La route qui permet de supprimer un patient d'un lit.

Une requête est envoyé qui supprime le lit où l'id du patient correspond avec l'id du patients assigné à la table lit.

Puis redirige l'utilisateur vers la chambre où l'utilisateur à était supprimé.

```
        $db = $this->getDoctrine()->getManager();
        $query = 'DELETE FROM `lit` WHERE patient_id = '. $_GET['patient'] .'';
        $stmt = $db->getConnection()->prepare($query);
        $stmt->execute();
        //Redirige l'utilisateur vers la chambre, pour recharger les modifications
    return $this->redirectToRoute('chambre', ['numService' => $_GET['numService'], 'numChambre' =>
```

3 – AjoutPatient Controller

3.1) On initialise les variables numChambre et numService.

```
    $numChambre = $_GET['numChambre'];
    $numService = $_GET['numService'];

    //Si le formulaire à était remplie alors ajoute un patient à un nouveau lit
    if (!empty($_POST['nom'])) {
```

3.2) On stocke les données du formulaire dans des variables.

```
    //Si le formulaire à était remplie alors ajoute un patient à un nouveau lit
    if (!empty($_POST['nom'])) {
        $nom = $_POST['nom']; $prenom = $_POST['prenom']; $gsm = $_POST['gsm']; $adress = $_POST['
        $date_naissance = $_POST['naissance'];
        $arrive = $date_naissance;
        $sex = $_POST['sex'];
```

3.3) On prépare la requête SQL pour ajouter un patient avec les bon attributs.

```
        $db = $this->getDoctrine()->getManager();
        //Ajoute le patient dans la base de donné
        $query = "INSERT INTO patient (nom,prenom,adress,naissance,gsm,arrive,sex)
                VALUE (:nom, :prenom, :adress, :naissance, :gsm, :arrive, :sex)";
        $stmt = $db->getConnection()->prepare($query);
```

3.4) On map les données de la requête SQL avec les variables initialisé au début du controller.

```
            $params = array(
                                'nom'=>$nom,
                                'prenom'=>$prenom,
                                'adress'=>$adress,
                                'naissance'=>$date_naissance,
                                'gsm'=>$gsm,
                                'arrive'=>$arrive,
                                'sex'=>$sex
                            );

            $stmt->execute($params);
```

3.5) Une requête SQL est envoyer à la base de donnée pour récupérer l'id du patient créé.

3.6) Un lit est ajouté avec le bon id patient. Puis l'utilisateur est redirigé vers la chambre où le

```
            $patient_id = $connection->fetchAll('SELECT id_patient FROM patient
            WHERE nom = "'. $nom .'" AND prenom = "'. $prenom .'" AND gsm = "'. $gsm .'" ');
```

patient à était ajouté.

```
            //Ajoute un lit dans la base de donné et le lit avec l'id du patient ajouté
            $db = $this->getDoctrine()->getManager();
            $query = "INSERT INTO lit (patient_id,chambre_id)
                    VALUE (:patient_id, :chambre_id)";
            $stmt = $db->getConnection()->prepare($query);
            $params = array(
                            'patient_id'=>$patient_id[0]['id_patient'],
                            'chambre_id'=>$numChambre,
                        );

            $stmt->execute($params);

            //redirige l'utilisateur ver la chambre ou le patient à était ajouté
            return $this->redirectToRoute('chambre', ['numService' => $numService, 'numChambre' => $nu
```