**Rapport JEE**

Page de garde

**Sommaire**

[I. Structuration de code, découplage et injection de dépendance 3](#_Toc97558382)

[II. Utilisation de Servlet et/ou JSP pour l’implémentation du module de service 3](#_Toc97558383)

[III. Transformation objet-relationnel 3](#_Toc97558384)

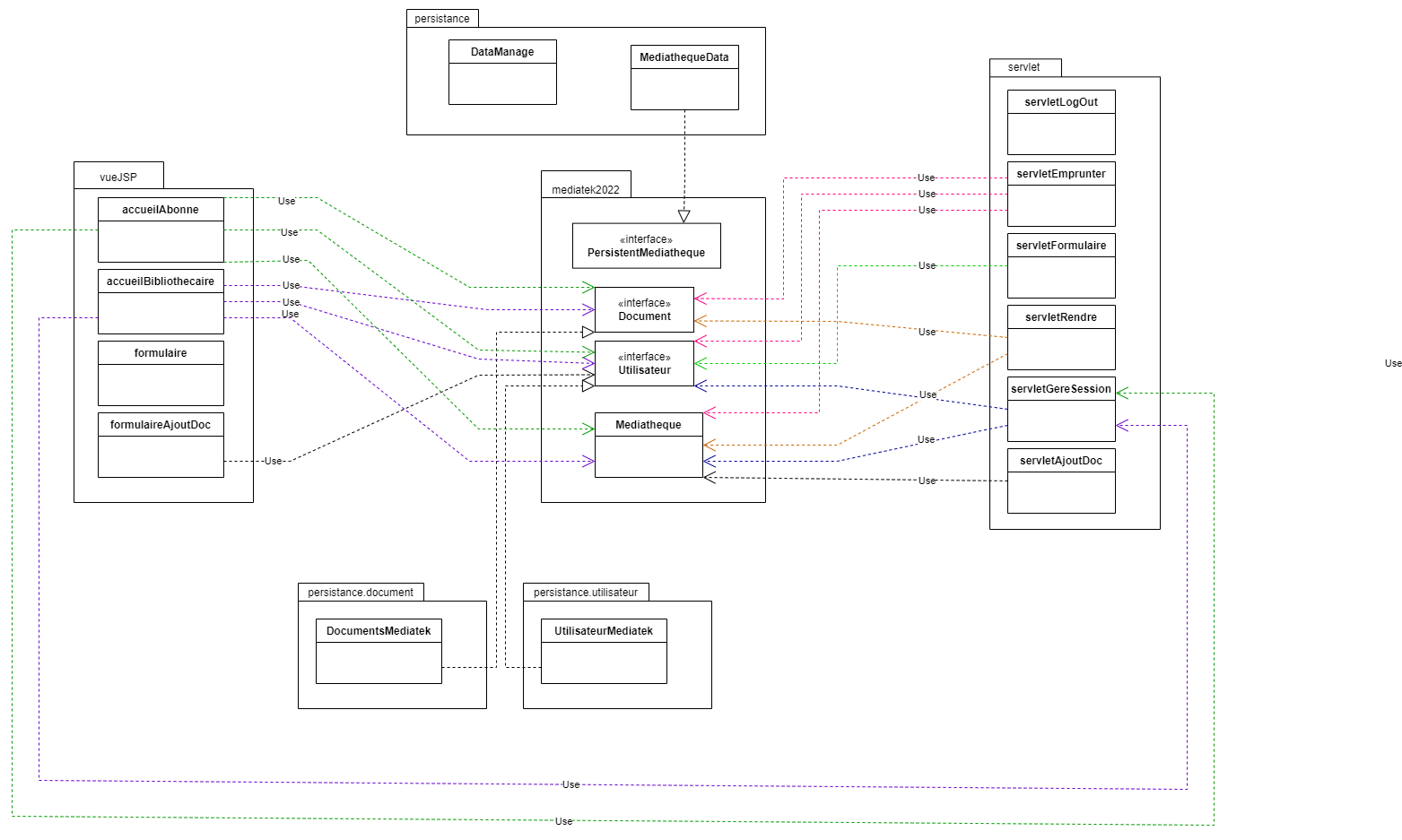
[IV. Variables sessions 3](#_Toc97558385)

[V. Concurrence 3](#_Toc97558386)

[VI. Efficacité des requêtes d’accès à la base de données 3](#_Toc97558387)

1. Structuration de code, découplage et injection de dépendance

Voici le diagramme d’architecture de notre projet. On peut voir les différentes liaisons entre les packages et les classes.



1. Utilisation de Servlet et/ou JSP pour l’implémentation du module de service

Nous avons choisi d’utiliser des Servlet et des JSP pour implémenter les différents services disponibles de notre application.

Nous utilisons des Servlets à la suite d’une requête HTML, par exemple, lorsqu’un utilisateur souhaite se connecter, ou bien qu’un bibliothécaire ajoute un nouveau document dans la médiathèque. Les requêtes ont lieu lors d’un changement de page généralement.

Ces servlets récupèrent les informations grâce aux méthodes GET et POST, ce qui nous permet d’effectuer les actions demandées.

Nous utilisons des JSP pour effectuer les autres actions, celles qui ne sont pas issues d’une requête HHTP.

Grâce aux JSP, nous pouvons afficher le contenu de la base de données (tous les documents d’un utilisateurs, ou de la médiathèque) dès le chargement (ou le rechargement) de la page. Cela permet d’avoir un site dynamique qui prend directement en compte un changement (un emprunt de document par exemple).

1. Transformation objet-relationnel

Concernant le passage d’objets de l’application Document et Utilisateur aux tables relationnelles de la base de données, nous effectuons se changement à l’aide de requêtes SQL. Les documents et utilisateurs ont différents attributs, attributs que l’on retrouve dans les tables de la base de données. De cette manière, lors d’une insertion ou d’une requête pour récupérer ces objets, nous avons une correspondance, ce qui permet de passer des objets à la base de données très facilement.

1. Variables sessions

Dans le cadre de notre application, nous avons mis en place des variables de sessions. Cette variable est créée. Elle contient un minimum d’informations sur l’utilisateur,

1. Concurrence
2. Efficacité des requêtes d’accès à la base de données

/!\ Nous avons convenus ensemble que nous pouvions rajouter deux nouvelles méthodes dans l’interface Médiathèque. Ces nouvelles méthodes nous permet de visualiser tous les documents qu’un utilisateur à emprunter et de consulter tous les documents de la bibliothèque