Théo CLERE, Maxime BROSSARD, Sandro SPINA, Ceif-Edine MAROUANI, Julien BUC

cpe LYON

Web Statique & Dynamique

2023-2024

Table des matières

[I. Diagrammes de conception 2](#_Toc167641540)

[A. Création d’un utilisateur : 2](#_Toc167641541)

[B. Connexion d’un utilisateur 3](#_Toc167641542)

[C. Diagramme de classe User-> Card 4](#_Toc167641543)

[D. Diagramme des instances réel 5](#_Toc167641544)

[E. Diagramme de l’application 3 Tiers 6](#_Toc167641545)

[F. Table des illustrations 7](#_Toc167641546)

# Diagrammes de conception

Les différents diagrammes ont été réalisé dans un but de compréhension et/ou faisant suite à une demande précise de la part du corp enseignant.

## Création d’un utilisateur :

Lorsqu’un nouvelle utilisateur arrive sur la page de l’application web, celui-ci va créer un compte. La page de connexion/création de compte est afficher grâce au service rendu par l’authentication manager qui va vérifier le cache du navigateur et chercher si une session active existe (via l’existence d’un TOKEN). Le service appelé pour la création d’un compte est dans le UserManager, celui-ci va créer un utilisateur en base de données (via l’ORM) et passer à l’étape suivante, qui est l’attribution aléatoire de 5 carte, donc une boucle de 5 tour est exécuter sur CardManager. Une fois les cartes créer, pour assurer la persistance, celle-ci sont ajouter en base de donnes et l’utilisateur (qui viens d’être créé) finis par être retourner à notre client.

Une image contenant texte, diagramme, ligne, Parallèle

Description générée automatiquement

Figure 1 - conception du workflow de création d'un utilisateur

Le diagramme précèdent s’accompagne d’un diagramme de séquence précisent mieux l’intervention et l’ordre des différents tiers de l’applications.

Une image contenant texte, diagramme, ligne, Parallèle

Description générée automatiquement

Figure 2 - Diagramme de séquence du workflow de création d'un utilisateur

## Connexion d’un utilisateur

Ci-dessous le diagramme de séquence de connexion d’un utilisateur, comme préciser précédemment, un utilisateur a besoin d’un TOKEN afin de réaliser n’importe quel action sur l’application Web. Le diagramme suivant explique le processus d’obtention du TOKEN (stocke dans le navigateur du client final).

Une image contenant texte, diagramme, ligne, Plan

Description générée automatiquement

Figure 3 - Diagramme de séquence d'une connexion d'un utilisateur

## Diagramme de classe User-> Card

Chaque utilisateur obtient des cartes lors de la création de son compte. Pour éviter les doublons et permettre à chacun de posséder des exemplaires d’une carte ou d’en avoir plusieurs sur le marché, nous avons mis en place des CardInstance, instance d’une carte (Card étant unique).

Une image contenant texte, diagramme, ligne, Plan

Description générée automatiquement

Figure 4 - Diagramme de classe Utilisateur -> Card

## Diagramme des instances réel

Une image contenant texte, diagramme, Plan, capture d’écran

Description générée automatiquementLe diagramme suivant décris et distingue les instances en jeux lors de l’exécution de l’application. On retrouve en premier lieux une porte qui va filtrer les interactions possible de la part du client. Toute interaction doit être authentifier et l’utilisateur n’interagit qu’avec le front.  
Le front fait appel a des routes gérer par le UserController et le CardController. En fonction de la demande, les différentes briques de l’application Spring interagissent afin de fournir la reponse adapter à la demande du front. Une fois les données récupérer depuis la base de données, la logique et le traitement exécuter, une réponse est retournée et le front se met a jour afin de remplir les différents champs coter UI.

Figure 5 - Diagramme des briques en interractions dans l'application web

## Diagramme de l’application 3 Tiers

Une image contenant texte, diagramme, capture d’écran, Plan

Description générée automatiquementDiagramme décrivant l’architecture 3 Tiers adopte par le projet dans le cadre de l’utilisation d’une base de données PostgreSQL dans le cloud, un serveur Spring Boot et un front-end React.

Figure 6 - Schema de l'application 3 Tiers

## Table des illustrations

[Figure 1 - conception du workflow de création d'un utilisateur 2](#_Toc167641425)

[Figure 2 - Diagramme de séquence du workflow de création d'un utilisateur 3](#_Toc167641426)

[Figure 3 - Diagramme de séquence d'une connexion d'un utilisateur 3](#_Toc167641427)

[Figure 4 - Diagramme de classe Utilisateur -> Card 4](#_Toc167641428)

[Figure 5 - Diagramme des briques en interractions dans l'application web 5](#_Toc167641429)

[Figure 6 - Schema de l'application 3 Tiers 6](#_Toc167641430)