Perlini Nilo et Kazanjian Adrian

Documentation

Module 133 – Forum\_Doudix

Table des matières

[Introduction 2](#_Toc193785206)

[Description du projet 2](#_Toc193785207)

[Fonctionnalités 2](#_Toc193785208)

[Analyse et conception 3](#_Toc193785209)

[Maquette 3](#_Toc193785210)

[Use case 6](#_Toc193785211)

[Activity Diagram 9](#_Toc193785212)

[Séquence System global 9](#_Toc193785213)

[Développement 11](#_Toc193785214)

[Tests 11](#_Toc193785215)

[Déploiement 11](#_Toc193785216)

[Maintenance et évolution 11](#_Toc193785217)

[Conclusion 11](#_Toc193785218)

# Introduction

Dans le cadre du module 133, nous avons réalisé une application WEB utilisant les points suivants :

* Les sessions Utilisateurs
* Les requêtes SQL
* Des bases de données
* Dockers : (DockerFiles, DockerHUB..)
* GitHub : (Repository..)
* Java

Voici donc la documentation du projet que nous avons réalisé (Adrian Kazanjian et Nilo Perlini) durant ce module :

## Description du projet

Voici la description de notre projet :

|  |  |  |  |
| --- | --- | --- | --- |
| **Nom du Projet** | | | **Doudix** |
| **Langage** | | Java-17 | **Description** |
| **Emplacement** | | WEB | Notre projet est de réaliser un forum WEB pour que nous puissions poser des questions et avoir notre propre forum de l’EMF. Crée et managé par des élèves de l’écoles. |
| **Développeurs** | | |
| Adrian | Kazanjian | |
| Nilo | Perlini | |

## Fonctionnalités

Voici les fonctionnalités principales de notre projet :

|  |
| --- |
| **Fonctionnalités** |
| Créer un compte, se connecter, se déconnecter, personnalisé ses informations |
| Créer des postes : (Proposer un thème, donner son avis..etc...) |
| Répondre à des postes : (Affichage utilisateurs, afficher des avis…etc.) |
| Admin : Contrôle des postes (Supprimer, modifier ) |
| Poster des images |
| Admin : Bannissement des utilisateurs |

Une image contenant Police, Graphique, logo, symbole

Le contenu généré par l’IA peut être incorrect.

# Analyse

## Maquette

Page Login : Au lancement du site, nous allons avoir deux options différentes. Soit se connecter à un compte existant ou bien en créer un directement en cliquant sur le bouton de la partie verte.

Une image contenant texte, capture d’écran, logiciel, Système d’exploitation

Le contenu généré par l’IA peut être incorrect.

Page Register : Cette page s’affiche suite à l’interaction avec le bouton « Sign Up » dans la première page de base. Dans cet affichage, il sera possible de créer un compte afin d’accéder au forum. Il faut donc remplir les champs avec un nom d’utilisateur et un mot de passe.

Une image contenant texte, capture d’écran, logiciel, Police

Le contenu généré par l’IA peut être incorrect.

Page Home User : Voilà la page utilisateur après la connexion à un utilisateur client. Il sera possible de créer un poste, de cliquer sur un poste déjà existant afin d’y réponde et interagir avec l’auteur du post. L’utilisateur pourra aussi choisir une catégorie et afficher tous les postes affiliés à cette catégorie.

Une image contenant texte, capture d’écran, logiciel, Icône d’ordinateur

Le contenu généré par l’IA peut être incorrect.

Page Post : Voici la page lorsque l’utilisateur veut ajouter un poste, il doit ajouter un titre, une description, une catégorie et une couleur. Il peut s’il veut ajouter une photo au poste mais cela n’est pas obliger.

Une image contenant texte, capture d’écran, logiciel, affichage

Le contenu généré par l’IA peut être incorrect.

Page Discussing : Cette page s’affiche lorsque l’utilisateur clique sur le poste publié d’un autre utilisateur. Il pourra répondre en écrivant un message et cela va permettre d’avoir une discussion sur n’importe quel sujet.

Une image contenant texte, capture d’écran, logiciel, Page web

Le contenu généré par l’IA peut être incorrect.

Page Home Administrator : Cette page s’affiche lors de la connexion à un compte administrateur. Il aura la possibilité de supprimer n’importe quel poste et de rechercher un utilisateur dans la liste de tous les utilisateurs créer. Selon son envie il peut aussi bannir un utilisateur en supprimant son compte.

Une image contenant texte, capture d’écran, logiciel, Icône d’ordinateur

Le contenu généré par l’IA peut être incorrect.

## Use case

Use Case Global : Voici le Use Case Global.

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

Le contenu généré par l’IA peut être incorrect.

Usercase\_client1 : Voici le Use Case de la partie utilisateur. Nous avons comme acteur l’utilisateur de base et l’utilisateur authentifié à l’application. Le premier acteur ne peut que se connecter et créer un compte. Le deuxième peut ajouter un poste et y répondre.

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

Le contenu généré par l’IA peut être incorrect.

Usercase\_client2 : Voici le Use Case de la partie admin. Nous avons comme acteur l’utilisateur de base et l’administrateur authentifié à l’application. Le premier acteur ne peut que se connecter et créer un compte. Le deuxième peut supprimer un poste et bannir un utilisateur.

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

Le contenu généré par l’IA peut être incorrect.

Usercase\_APIGateway : Ce diagramme affiche toutes les redirections des APIGateway vers les Ctrl Rest. Il y a donc l’ajout de message, ajout d’un poste, la déconnexion, la connexion et l’inscription pour le client 1. Pour le 2e client, il y a la suppression d’un poste et le bannissement d’un utilisateur.

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

Le contenu généré par l’IA peut être incorrect.

Usercase\_rest1 : Ce diagramme affiche toutes les redirections des APIGateway vers les Ctrl Rest afin d’interagir avec la bonne base de données. Il y a donc l’ajout de message, ajout d’un poste, la déconnexion, la connexion et l’inscription pour le client 1.

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

Le contenu généré par l’IA peut être incorrect.

Usercase\_rest2 : Ce diagramme affiche toutes les redirections des APIGateway vers les Ctrl Rest afin d’interagir avec la bonne base de données. Pour le 2e client, il y a la suppression d’un poste et le bannissement d’un utilisateur.

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

Le contenu généré par l’IA peut être incorrect.

## Activity Diagram

Activity Diagram client-gw-rest1 : Voici le diagramme d’activité pour le client1 afin de créer un poste.

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

Le contenu généré par l’IA peut être incorrect.

Activity Diagram client-gw-rest2 : Voici le diagramme d’activité pour le client2 afin de supprimer un poste.

Une image contenant texte, capture d’écran, Rectangle, Parallèle

Le contenu généré par l’IA peut être incorrect.

## Séquence System global

Séquence System client-apigw-rest1 : Voici le diagramme de séquence du client1 pour créer un poste.

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

Le contenu généré par l’IA peut être incorrect.

Séquence System client-apigw-rest2 : Voici le diagramme de séquence du client2 pour supprimer un poste.

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

Le contenu généré par l’IA peut être incorrect.

# Conception

## Diagramme de navigation

Dans ce diagramme, tout le site web est conceptualiser sous forme de schéma et de liens. Il montre le processus d’authentification des utilisateurs, la différence des rôles client et administrateur et la redirection en fonction des permissions. L’API Gateway va avoir comme rôle un intermédiaire afin de traiter les requêtes et les envoyés au bon services.

Une image contenant texte, diagramme, Plan, Parallèle

Le contenu généré par l’IA peut être incorrect.

## Diagramme Architecte

Ce diagramme illustre la navigation et les interactions entre les différentes pages, les fichier JavaScript et l’API Gateway. Les utilisateurs commencent par se connecter depuis l’index.jsp ou s’inscrire depuis register.jsp, avec la validation qui est gérée par des scripts JavaScript. Après que les informations ont bien été interprété à l’API Gateway, celle-ci va rediriger l’utilisateur vers la page html lui étant attribué (forumClientSide.html pour les utilisateurs classique et forumAdminSide.html pour les administrateurs).

## Une image contenant texte, capture d’écran, diagramme, ligne Le contenu généré par l’IA peut être incorrect.

## Architecte API Gateway

Dans ce diagramme, nous montrons l’architecture d’une API Gateway qui est utilisé pour la gestion des utilisateurs et la communication avec un backend. L’API Gateway va centralisé l’authtification et l’inscription des utilisateurs depuis un UserManager. Elle interagit avec le cas d’utilisation UseCaseRest1 qui va dépendre de l’interface ItfWrkCtrl. Le composant Wrk, va implémenter l’interface ItfCtrlWrk et communique avec un TCPClient qui va lui permettre de faire des interactions avec d’autre services.

Une image contenant texte, capture d’écran, Post-it, Rectangle

Le contenu généré par l’IA peut être incorrect.

## Architecte Rest

Ce diagramme va montrer le fonctionnement de la partie Rest du projet avec les requêtes pour interagir avec la base de données. Dans ce diagramme nous pouvons voir notre restController\_client1 qui va le Wrk UserManager car c’est ce dernier qui va faire la requête de création d’utilisateur dans la base de données.

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

Le contenu généré par l’IA peut être incorrect.

# Développement

# Tests

# Déploiement

# Maintenance et évolution

# Conclusion