

Quentin BOULET
Sébastien DOS SANTOS
Romain DOS SANTOS

Rapport de Développement : Système de Revue de Code (Backend)

Sommaire:

Introduction.....	3
Contexte du Projet.....	3
Objectif du Rapport.....	3
Structure du Rapport.....	3
Architecture.....	4
Front-end :.....	4
Back-end :.....	4
Base de données :.....	4
API RESTful :.....	4
Système de Gestion de Versions :.....	5
Organisation du Développement.....	6
Répartition des Tâches :.....	6
Méthodologie de Développement :.....	6
Outils de Gestion de Projet :.....	7
Gestion des Doubles Quotes :.....	8
Gestion des Erreurs :.....	8
Synchronisation des Données :.....	8
Approche de Résolution des Problèmes :.....	9
Technologies Utilisées :.....	9
Collaborations Internes et Externes :.....	9
Conclusion.....	10

Introduction

Ce rapport décrit notre projet de développement, le système de revue de code. Notre équipe, composée de Quentin Boulet, Sébastien Dos Santos et Romain Dos Santos, a travaillé sur ce projet.

Contexte du Projet

Le système de revue de code vise à faciliter la collaboration entre développeurs en offrant une plateforme permettant de soumettre et de reviewer du code Java avec des tests unitaires associés.

Objectif du Rapport

Ce rapport a pour objectif de documenter notre processus de développement, en mettant en lumière les différentes étapes de conception, les défis rencontrés et les solutions apportées.

Structure du Rapport

Le rapport est organisé en plusieurs sections pour couvrir les aspects clés de notre projet. Nous commencerons par présenter l'architecture générale du système, puis nous discuterons de notre méthodologie de développement et de l'organisation de notre équipe. Ensuite, nous aborderons les difficultés techniques rencontrées et les solutions adoptées. Enfin, nous conclurons avec une réflexion sur les perspectives d'avenir du projet.

Architecture

L'architecture du système de revue de code est conçue de manière à garantir sa modularité, sa maintenabilité et sa scalabilité. Voici une vue d'ensemble des différents composants et de leur interaction :

Front-end :

Nous avons opté pour le développement de l'interface utilisateur en utilisant la plateforme Android pour offrir une expérience utilisateur optimale sur les appareils mobiles. Nous avons utilisé le langage Kotlin ainsi que les outils de développement Android Studio pour créer une application native performante et intuitive.

Back-end :

Pour le back-end, nous avons choisi le framework Spring, une plateforme de développement d'applications Java qui offre une gamme complète de fonctionnalités pour créer des applications robustes et évolutives. Plus spécifiquement, nous avons utilisé Spring Boot pour simplifier le processus de configuration et de déploiement, et Spring MVC pour la création d'API RESTful. Cette combinaison nous a permis de développer rapidement des services web performants pour gérer les requêtes du front-end et manipuler les données de manière efficace.

Base de données :

Nous avons utilisé la base de données H2, une base de données relationnelle en mémoire, pour stocker les informations utilisateur, les codes soumis, les reviews, les votes et autres données pertinentes pour le fonctionnement de l'application. Spring Data JPA a été utilisé pour simplifier l'accès et la manipulation des données au sein de l'application.

API RESTful :

Le back-end expose une API RESTful qui permet au front-end de communiquer avec le serveur de manière standardisée. Cette API fournit des endpoints pour toutes les fonctionnalités principales de l'application, telles que la création d'un compte, la création de reviews, les votes, etc.

Système de Gestion de Versions :

Nous utilisons Git comme système de gestion de versions pour contrôler les différentes versions du code source du projet. Cela nous permet de travailler efficacement en équipe et de gérer les modifications apportées au code de manière organisée.

Cette architecture permet une séparation claire des responsabilités entre les différents composants du système, favorisant ainsi la réutilisabilité du code, la facilité de maintenance et l'évolutivité de l'application.

Organisation du Développement

Dans cette section, nous décrirons l'organisation adoptée pour le développement du système de revue de code, y compris la répartition des tâches au sein de l'équipe, la méthodologie de travail et les outils utilisés pour la gestion du projet.

Répartition des Tâches :

Au début du projet, nous avons effectué une analyse des différentes fonctionnalités à implémenter et des compétences individuelles de chaque membre de l'équipe. En fonction de cela, nous avons réparti les tâches de manière équilibrée afin d'exploiter au mieux les forces de chacun. Voici un aperçu de la répartition des tâches :

- Quentin Boulet : BroadcastReceiver pour notification, affichage des reviews, comments, etc... d'un certain user, ...
- Sébastien Dos Santos : gestion de l'authentification, création du token, ...
- Romain Dos Santos : recherche d'une revue, changer le mot de passe..

Méthodologie de Développement :

Nous avons opté pour une approche agile dans laquelle nous avons divisé le projet en différentes itérations, appelées sprints, de deux semaines chacune. Avant le début de chaque sprint, nous avons organisé une réunion de planification au cours de laquelle nous avons défini les objectifs à atteindre et les fonctionnalités à développer. Pendant le sprint, nous avons tenu des réunions quotidiennes de stand-up pour discuter de l'avancement du travail et des éventuels obstacles rencontrés. À la fin de chaque sprint, nous avons organisé une réunion de rétrospective pour évaluer notre performance et identifier des pistes d'amélioration pour les sprints suivants.

Outils de Gestion de Projet :

Pour la gestion du projet, nous avons utilisé plusieurs outils :

- GitLab : Nous avons hébergé notre code source sur GitLab et utilisé les fonctionnalités de gestion de projet intégrées pour suivre les problèmes, les fonctionnalités et les tâches.
- Google Drive Excel: Nous avons utilisé le excel de google drive pour pouvoir suivre l'avancé des tâches. Chaque ligne représentant une tâche avec son statut, le responsable de la tâche, la date de commencement, etc...
- Discord : Nous avons utilisé Discord comme outil de communication principale pour les discussions en temps réel et les réunions virtuelles.

Cette organisation nous a permis de travailler de manière collaborative et efficace, en assurant une communication fluide entre les membres de l'équipe et en favorisant un développement itératif et incrémental du système.

Difficultés Rencontrées

Au cours du développement du système de revue de code, plusieurs difficultés ont émergé, nécessitant des solutions créatives et des ajustements dans notre approche de travail. Voici quelques-unes des principales difficultés rencontrées :

Gestion des Doubles Quotes :

Lors de la création d'un commentaire, le front envoyait une string au back afin d'ajouter le commentaire à la base de données. Cependant, pour la partie web cette fonctionnalité fonctionnait parfaitement, mais pour android, le back ajoutait des doubles quotes (") au début et à la fin du texte. Cela était dû au fait que le back récupérait directement une string et non un objet DTO. Retrofit envoyait donc un JSON qui contenait les quotes et le serveur au parsing, parfait donc les doubles quotes. On a donc modifié le backend pour qu'il ne récupère pas directement une string mais bien un objet DTO.

Gestion des Erreurs :

Un autre défi majeur a été la gestion des erreurs et des cas imprévus tout au long du développement. Parfois, des erreurs inattendues se produisaient lors de l'exécution de certaines fonctionnalités, ce qui pouvait entraîner des comportements indésirables ou des plantages de l'application. Pour résoudre ces problèmes, nous avons ajouté des logs au endroit important afin de trouver les différents problèmes et d'où il venait afin de les corriger rapidement.

Synchronisation des Données :

La synchronisation des données entre le front-end et le back-end a également posé des défis, en particulier dans les cas où les données devaient être mises à jour en temps réel ou de manière asynchrone. Nous avons dû mettre en place des mécanismes de gestion de l'état et des notifications pour assurer une communication fluide entre les différentes parties de l'application.

En surmontant ces difficultés, nous avons acquis une expérience précieuse en matière de développement logiciel et avons renforcé notre capacité à résoudre des problèmes techniques de manière efficace et collaborative.

Solutions Adoptées

Pour résoudre les défis rencontrés lors du développement du système de revue de code, nous avons adopté plusieurs approches et solutions créatives.

Approche de Résolution des Problèmes :

Nous avons adopté une approche itérative et pragmatique pour résoudre les problèmes rencontrés. Plutôt que de chercher des solutions complexes dès le départ, nous avons préféré des solutions simples et efficaces. Nous avons également encouragé la collaboration au sein de l'équipe, en partageant nos idées et nos expériences pour trouver ensemble les meilleures solutions.

Collaborations Internes et Externes :

En interne, nous avons maintenu une communication ouverte et transparente entre les membres de l'équipe. Nous avons organisé des réunions régulières pour discuter des progrès, des défis et des solutions possibles. De plus, nous avons sollicité des retours d'autres étudiants et enseignants sur notre travail, ce qui nous a permis d'identifier des problèmes potentiels et d'apporter des améliorations tout au long du processus de développement.

Conclusion

Le développement du système de revue de code a été une expérience enrichissante qui nous a permis d'explorer et de maîtriser divers aspects du développement mobile. Tout au long de ce projet, nous avons été confrontés à des défis techniques et organisationnels auxquels nous avons dû trouver des solutions créatives et efficaces.

En adoptant une approche itérative et collaborative, nous avons pu surmonter ces obstacles et livrer un système fonctionnel et robuste. L'utilisation de technologies modernes telles que Spring pour le back-end et Android pour le front-end nous a permis de développer une application performante et adaptée aux besoins de nos utilisateurs.

La collaboration au sein de l'équipe a également joué un rôle crucial dans le succès du projet. En maintenant une communication ouverte et en partageant nos connaissances et nos expériences, nous avons pu résoudre rapidement les problèmes et avancer efficacement dans le développement de l'application.

En conclusion, ce projet nous a permis d'acquérir de nouvelles compétences et de renforcer notre compréhension des bonnes pratiques de développement logiciel. Nous sommes fiers du travail accompli et confiants dans la qualité du système que nous avons livré. Nous espérons que notre application sera utile à ses utilisateurs et qu'elle contribuera à promouvoir la collaboration et l'excellence dans le domaine du développement de logiciels.