ECOLE MAROCAINE DES SCIENCES DE L'INGENIEUR
Membre de HONORIS UNITED UNIVERSITIES

MIC Consulting Group

PROJET DE STAGE

Année 2025-2026

Conception et réalisation d'une plateforme intelligente de formation en ligne et d'orientation pédagogique

Ingénierie Informatique et Réseaux (DDSI)

Réalisé par :

KOUBRI HATIM

Encadré par :

Kenza Bengoud

Dédicace

Je dédie ce travail à mes chers parents,

pour leurs sacrifices, leur soutien inconditionnel, leur patience et leurs encouragements tout au long de mon parcours académique. Leur confiance et leurs conseils ont toujours constitué une source essentielle de motivation et de persévérance.

À ma famille,

pour son affection, sa présence et son soutien permanent durant les différentes étapes de ma formation.

À mes enseignants de l'EMSI Marrakech, et tout particulièrement à mon encadrante académique, Mme Kenza Bengoud,

pour les connaissances transmises, leur disponibilité et leur contribution à mon développement académique et professionnel.

À mes amis et à toutes les personnes qui m'ont accompagné, conseillé et encouragé, de près ou de loin, dans la réalisation de ce projet.

Hatim KOUBRI

Remerciements

Au terme de ce travail, je tiens à exprimer ma profonde gratitude à toutes les personnes qui ont contribué, directement ou indirectement, à la réalisation de ce projet de fin d'année.

Je remercie tout particulièrement mon encadrante professionnelle, **Oumaima RAHIL**, pour son accompagnement, sa disponibilité, ses orientations et les conseils qu'elle m'a apportés tout au long de cette expérience. Son suivi m'a permis de mieux organiser mon travail, de surmonter les difficultés rencontrées et d'améliorer progressivement la qualité de la solution réalisée.

J'adresse également mes sincères remerciements à mon encadrante académique à l'EMSI Marrakech, **Mme Kenza Bengoud**, pour son suivi, ses conseils et ses remarques tout au long de ce projet.

J'adresse également mes sincères remerciements à l'ensemble de l'équipe de **MIC Consulting Group**, et plus particulièrement au service informatique, pour son accueil, sa confiance et les conditions favorables mises à ma disposition durant la période du stage. Cette expérience m'a permis de renforcer mes compétences techniques, organisationnelles et professionnelles.

Je remercie aussi le corps professoral et administratif de l'**École Marocaine des Sciences de l'Ingénieur**, campus de Marrakech, pour la qualité de la formation dispensée au cours de mon parcours en **Ingénierie Informatique et Réseaux**, spécialité **Développement Digital et Systèmes d'Information**. Les connaissances acquises durant cette formation ont constitué une base essentielle pour la conception et le développement de Khotwa.

Mes remerciements s'adressent également aux membres du jury pour l'attention portée à ce travail, le temps consacré à son évaluation et les remarques constructives qui contribueront à son amélioration.

Enfin, je tiens à remercier chaleureusement mes parents, ma famille et mes proches pour leur soutien moral, leur patience et leurs encouragements permanents tout au long de mon parcours académique et durant la réalisation de ce projet.

Résumé

Dans le cadre du projet de fin d'année réalisé au sein du service informatique de **MIC Consulting Group**, ce travail porte sur la conception et le développement d'une plateforme E-learning nommée **Khotwa**. L'objectif principal consiste à proposer une solution centralisée, sécurisée et accessible, répondant aux besoins des participants, des formateurs et des administrateurs.

La plateforme permet aux participants de consulter un catalogue de formations, d'accéder à des contenus pédagogiques organisés en modules et en chapitres, de suivre leur progression, de passer des quiz et d'obtenir des certificats. Elle intègre également des fonctionnalités complémentaires telles que les notes, les signets, les favoris, les notifications et les classes virtuelles. Les formateurs disposent d'outils pour créer et gérer leurs formations, organiser les ressources pédagogiques, préparer les évaluations et suivre les participants. Les administrateurs peuvent gérer les utilisateurs, traiter les candidatures des formateurs, modérer les avis et consulter les informations de supervision.

Sur le plan technique, Khotwa repose sur un backend développé avec Java et Spring Boot, une application Web réalisée avec Next.js, React et TypeScript, ainsi qu'une application mobile Flutter. MySQL assure la persistance des données, Flyway gère les migrations et MinIO est utilisé pour le stockage des ressources. La sécurité repose sur Spring Security, les jetons JWT et une gestion des autorisations selon les rôles.

La qualité de la solution est vérifiée par des tests automatisés et une analyse SonarQube. Cette dernière présente une porte de qualité validée, une couverture de 75,1 % et un taux de duplication de 0,9 %. Les résultats obtenus montrent que Khotwa constitue une base fonctionnelle, modulaire et évolutive pour la gestion de formations en ligne.

Mots-clés : E-learning, Khotwa, Spring Boot, Next.js, Flutter, JWT, MySQL, MinIO, formation en ligne, sécurité, quiz, certificat.

Abstract

As part of the final-year project carried out within the IT department of **MIC Consulting Group**, this work focuses on the design and development of an E-learning platform called **Khotwa**. Its main objective is to provide a centralized, secure and accessible solution that addresses the needs of participants, trainers and administrators.

The platform allows participants to browse a training catalogue, access educational content organized into modules and chapters, monitor their progress, take quizzes and obtain certificates. It also provides complementary features such as notes, bookmarks, favorites, notifications and virtual classes. Trainers have access to tools for creating and managing courses, organizing educational resources, preparing assessments and monitoring participants. Administrators can manage users, process trainer applications, moderate reviews and consult supervision information.

From a technical perspective, Khotwa is based on a backend developed with Java and Spring Boot, a Web application built with Next.js, React and TypeScript, and a Flutter mobile application. MySQL ensures data persistence, Flyway manages database migrations and MinIO is used to store educational resources. Security is based on Spring Security, JWT tokens and role-based authorization.

The quality of the solution is assessed through automated tests and a SonarQube analysis. The analysis reports a passed Quality Gate, test coverage of 75.1% and a duplication rate of 0.9%. The obtained results show that Khotwa provides a functional, modular and scalable foundation for managing online learning activities.

Keywords : E-learning, Khotwa, Spring Boot, Next.js, Flutter, JWT, MySQL, MinIO, online learning, security, quiz, certificate.

Glossaire

Principaux termes et abréviations utilisés dans le rapport

Terme ou acronyme	**Description**
API	*Application Programming Interface* : interface permettant à plusieurs applications ou services de communiquer.
CI/CD	*Continuous Integration / Continuous Deployment* : pratiques d'intégration et de déploiement continus.
CRUD	*Create, Read, Update, Delete* : opérations fondamentales de création, de consultation, de modification et de suppression des données.
Docker	Plateforme permettant d'exécuter une application et ses dépendances dans des conteneurs.
DTO	*Data Transfer Object* : objet utilisé pour transporter les données entre les différentes couches de l'application.
E-learning	Mode d'apprentissage reposant sur l'utilisation de technologies et de contenus numériques.
EMSI	École Marocaine des Sciences de l'Ingénieur.
Flyway	Outil utilisé pour versionner et exécuter les migrations du schéma de la base de données.
Flutter	Framework développé par Google pour créer des applications multiplateformes.
HTTP	*Hypertext Transfer Protocol* : protocole utilisé pour les échanges de données sur le Web.
HTTPS	Version sécurisée du protocole HTTP utilisant le chiffrement des communications.
JPA	*Java Persistence API* : spécification Java utilisée pour gérer la persistance des données.

Terme ou acronyme	**Description**
Jitsi	Solution de visioconférence utilisée dans Khotwa pour l'organisation des classes virtuelles.
JSON	*JavaScript Object Notation* : format textuel utilisé pour l'échange de données.
JWT	*JSON Web Token* : format de jeton utilisé pour l'authentification et la sécurisation des échanges.
MinIO	Solution de stockage d'objets compatible avec l'API S3, utilisée pour conserver les ressources pédagogiques.
MVP	*Minimum Viable Product* : version initiale d'un produit contenant les fonctionnalités essentielles.
ORM	*Object-Relational Mapping* : technique permettant d'établir une correspondance entre les objets d'une application et les tables d'une base de données.
PDF	*Portable Document Format* : format utilisé pour les documents pédagogiques et les certificats.
PFA	Projet de fin d'année.
QCM	Questionnaire à choix multiple.
RBAC	*Role-Based Access Control* : méthode de contrôle d'accès fondée sur les rôles des utilisateurs.
REST	*Representational State Transfer* : style d'architecture utilisé pour concevoir les services Web.
S3	*Simple Storage Service* : modèle d'API utilisé pour le stockage d'objets.
SQL	*Structured Query Language* : langage utilisé pour manipuler les bases de données relationnelles.
Swagger	Ensemble d'outils permettant de documenter et de vérifier les services d'une API.
UI	*User Interface* : interface permettant à l'utilisateur d'interagir avec l'application.

Terme ou acronyme	**Description**
UML	*Unified Modeling Language* : langage standard utilisé pour la modélisation et la conception des systèmes informatiques.
URL	*Uniform Resource Locator* : adresse utilisée pour identifier une ressource.

Liste des figures

Figure 1.1	Organigramme de MIC Consulting Group	4
Figure 1.2	Cycle de développement selon la méthodologie Scrum	9
Figure 1.3	Diagramme de Gantt des sprints du projet Khotwa	10
Figure 2.1	Diagramme de cas d'utilisation du participant	16
Figure 2.2	Diagramme de cas d'utilisation du formateur	17
Figure 2.3	Diagramme de cas d'utilisation de l'administrateur	18
Figure 2.4	Diagramme de séquence de la soumission d'un quiz	20
Figure 2.5	Diagramme de séquence de la génération d'un certificat	21
Figure 2.6	Diagramme de classes de Khotwa	21
Figure 3.1	Architecture générale de la plateforme Khotwa	25
Figure 4.1	Page d'accueil de Khotwa	36
Figure 4.2	Catalogue des formations de Khotwa	37
Figure 4.3	Consultation des détails d'une formation	38
Figure 4.4	Interface de connexion sécurisée	39
Figure 4.5	Tableau de bord du participant	40
Figure 4.6	Lecteur de contenus pédagogiques	40
Figure 4.7	Résultat d'un quiz réussi	41
Figure 4.8	Certificat de réussite généré par Khotwa	42
Figure 4.9	Centre de notifications du participant	43
Figure 4.10	Tableau de bord du formateur	44
Figure 4.11	Tableau de bord de l'administrateur	45
Figure 4.12	Traitement d'une candidature de formateur	45
Figure 4.13	Authentification et consultation du catalogue sur l'application mobile	47
Figure 4.14	Gestion des formations et du parcours pédagogique sur mobile	48
Figure 4.15	Quiz intégré et fonctionnement hors connexion	49
Figure 4.16	Consultation et participation aux classes virtuelles	50
Figure 5.1	Résultat de l'analyse de qualité avec SonarQube	59

Liste des tableaux

Tableau 1.1 Fiche signalétique de MIC Consulting Group 3

Tableau 2.1 Principaux besoins fonctionnels de Khotwa 13
Tableau 2.2 Acteurs principaux de la plateforme Khotwa 14
Tableau 2.3 Description synthétique des principaux cas d'utilisation 19

Tableau 3.1 Principaux modules applicatifs de Khotwa 28
Tableau 3.2 Principales technologies utilisées dans Khotwa 33

Tableau 5.1 Niveaux de test appliqués à Khotwa 53
Tableau 5.2 Validation des principaux parcours fonctionnels 57
Tableau 5.3 Synthèse des indicateurs SonarQube 60

Tableau A.1 Organisation des composants de Khotwa 67

Tableau B.1 Matrice synthétique des rôles de Khotwa 69

Table des matières

Dédicace **I**

Remerciements **II**

Résumé **III**

Abstract **IV**

Glossaire et liste des abréviations **V**

Liste des figures **VIII**

Liste des tableaux **IX**

Introduction générale **1**

Chapitre 1 : Contexte général du projet **2**
1.1 Introduction . . . 2
1.2 Présentation de l'organisme d'accueil . . . 2
1.2.1 Fiche signalétique . . . 3
1.2.2 Domaines d'activité . . . 3
1.2.3 Une équipe pluridisciplinaire . . . 3
1.2.4 Organisation interne . . . 4
1.2.5 Principales références . . . 4
1.3 Contexte du stage . . . 5
1.4 Problématique . . . 5
1.5 Objectifs du projet . . . 6
1.5.1 Objectif général . . . 6
1.5.2 Objectifs spécifiques . . . 6
1.6 Acteurs concernés . . . 7
1.7 Périmètre fonctionnel . . . 7
1.7.1 Fonctionnalités incluses . . . 7
1.7.2 Limites du périmètre . . . 8
1.8 Méthodologie de gestion du projet . . . 8
1.8.1 Organisation du travail en sprints . . . 9
1.8.2 Gestion du Product Backlog . . . 10
1.8.3 Suivi et validation des itérations . . . 10
1.9 Conclusion . . . 11

Chapitre 2 : Analyse et conception **12**
2.1 Introduction . . . 12

2.2 Analyse des besoins . . . 12
2.2.1 Besoins fonctionnels . . . 12
2.2.2 Besoins non fonctionnels . . . 13
2.3 Identification des acteurs . . . 14
2.4 Diagramme UML . . . 14
2.5 Diagrammes de cas d'utilisation . . . 15
2.5.1 Cas d'utilisation du participant . . . 15
2.5.2 Cas d'utilisation du formateur . . . 16
2.5.3 Cas d'utilisation de l'administrateur . . . 17
2.6 Description des principaux cas d'utilisation . . . 18
2.7 Diagrammes de séquence . . . 20
2.7.1 Soumission d'un quiz . . . 20
2.7.2 Génération d'un certificat . . . 20
2.8 Diagramme de classes . . . 21
2.9 Conclusion . . . 23

Chapitre 3 : Architecture technique et technologies utilisées 24
3.1 Introduction . . . 24
3.2 Architecture générale de Khotwa . . . 24
3.3 Couche présentation . . . 25
3.3.1 Application Web . . . 26
3.3.2 Application mobile . . . 26
3.4 Couche applicative . . . 27
3.4.1 Principaux modules métier . . . 27
3.4.2 Communication par API REST . . . 28
3.4.3 Documentation de l'API . . . 29
3.5 Couche de données . . . 29
3.5.1 Base de données MySQL . . . 29
3.5.2 Spring Data JPA et Hibernate . . . 29
3.5.3 Gestion des migrations avec Flyway . . . 30
3.5.4 Stockage des fichiers avec MinIO . . . 30
3.6 Intégration des services externes . . . 30
3.6.1 Classes virtuelles avec Jitsi . . . 30
3.6.2 Service de courrier électronique . . . 31
3.6.3 Passerelle de paiement . . . 31
3.7 Architecture de sécurité . . . 31
3.7.1 Authentification avec Spring Security et JWT . . . 31
3.7.2 Gestion des rôles et des autorisations . . . 32
3.7.3 Protection des ressources pédagogiques . . . 32
3.8 Infrastructure et environnement d'exécution . . . 32
3.9 Synthèse des technologies utilisées . . . 33
3.10 Justification des choix technologiques . . . 34
3.11 Conclusion . . . 34

Chapitre 4 : Réalisation de la plateforme Khotwa 35
4.1 Introduction . . . 35
4.2 Réalisation de l'interface publique . . . 35
4.2.1 Page d'accueil . . . 35
4.2.2 Catalogue des formations . . . 36

4.2.3 Consultation des détails d'une formation . . . 37
4.3 Authentification et contrôle d'accès . . . 38
4.4 Réalisation de l'espace participant . . . 39
4.4.1 Tableau de bord du participant . . . 39
4.4.2 Consultation des contenus pédagogiques . . . 40
4.4.3 Évaluations et résultats . . . 41
4.4.4 Génération du certificat . . . 41
4.4.5 Centre de notifications . . . 42
4.5 Réalisation de l'espace formateur . . . 43
4.5.1 Tableau de bord du formateur . . . 43
4.6 Réalisation de l'espace administrateur . . . 44
4.6.1 Tableau de bord de l'administrateur . . . 44
4.6.2 Validation des candidatures des formateurs . . . 45
4.7 Réalisation de l'application mobile . . . 46
4.7.1 Authentification et consultation du catalogue . . . 46
4.7.2 Formations et parcours pédagogique . . . 47
4.7.3 Quiz et fonctionnement hors connexion . . . 48
4.7.4 Classes virtuelles et intégration de Jitsi Meet . . . 49
4.8 Conclusion . . . 50

Chapitre 5 : Tests, qualité et validation 52
5.1 Introduction . . . 52
5.2 Stratégie de test . . . 52
5.3 Tests du backend Spring Boot . . . 53
5.3.1 Tests unitaires . . . 53
5.3.2 Tests d'intégration . . . 54
5.3.3 Tests des opérations concurrentes . . . 54
5.3.4 Tests des migrations de données . . . 55
5.4 Tests de l'application Web . . . 55
5.4.1 Tests des composants . . . 55
5.4.2 Tests des parcours fonctionnels Web . . . 56
5.4.3 Commandes de vérification . . . 56
5.5 Validation de l'application mobile . . . 57
5.6 Validation fonctionnelle des principaux parcours . . . 57
5.7 Validation de la sécurité . . . 58
5.8 Analyse de la qualité avec SonarQube . . . 59
5.9 Limites de la validation actuelle . . . 60
5.10 Améliorations prévues . . . 61
5.11 Conclusion . . . 61

Conclusion générale et perspectives 62

Références 64

Annexe A : Organisation technique du projet 67

Annexe B : Matrice synthétique des rôles et autorisations 69

Introduction générale

La transformation numérique a profondément modifié les méthodes d'enseignement et de formation. Grâce aux technologies Web et mobiles, les plateformes E-learning permettent désormais d'accéder aux contenus pédagogiques à distance, de suivre une formation de manière progressive et de favoriser l'autonomie des apprenants. Cependant, une plateforme moderne ne doit pas se limiter à la diffusion de ressources : elle doit proposer une expérience interactive, sécurisée et adaptée aux besoins des participants, des formateurs et des administrateurs.

C'est dans ce contexte que s'inscrit le présent projet de fin d'année, réalisé au sein du service informatique de MIC Consulting Group. Il consiste à concevoir et à développer une plateforme E-learning sécurisée et multiplateforme, nommée Khotwa, accessible depuis une interface Web et une application mobile.

La plateforme permet aux participants de consulter un catalogue de formations, d'accéder à des modules, des chapitres, des vidéos et des documents PDF, de suivre leur progression, de passer des quiz et d'obtenir des certificats. Elle propose également des fonctionnalités complémentaires telles que les favoris, les notes, les signets, les notifications et les classes virtuelles. Les formateurs disposent d'outils pour organiser leurs contenus pédagogiques et suivre les activités des participants, tandis que les administrateurs peuvent gérer les utilisateurs, modérer les avis et superviser la plateforme à travers des tableaux de bord. Les recommandations personnalisées constituent une perspective d'évolution.

Sur le plan technique, Khotwa repose sur un backend développé avec Java et Spring Boot, une interface Web basée sur Next.js, React et TypeScript, ainsi qu'une application mobile réalisée avec Flutter. La solution utilise également MySQL, Flyway, Docker Compose, MinIO, Mailpit et Jitsi. La sécurité est assurée notamment par Spring Security, l'authentification JWT et un contrôle d'accès fondé sur les rôles. Le projet est conduit selon une démarche Agile inspirée de Scrum.

Le présent rapport est organisé en cinq chapitres. Le premier expose le contexte général du projet, notamment l'organisme d'accueil et le cadrage de Khotwa. Le deuxième présente l'analyse des besoins et la conception. Le troisième décrit l'architecture et les choix technologiques. Le quatrième présente la réalisation de Khotwa. Enfin, le cinquième est consacré aux tests, à la qualité et à la validation de la solution.

Chapitre 1

Contexte général du projet

1.1 Introduction

Ce chapitre présente l'organisme au sein duquel le stage a été effectué ainsi que le contexte général du projet. Il expose tout d'abord le profil de MIC Consulting Group, ses principaux domaines d'activité, son organisation interne et certaines de ses références. Il présente ensuite le cadre du projet Khotwa, la problématique identifiée, les objectifs recherchés, les acteurs concernés, le périmètre fonctionnel et la méthodologie adoptée pour conduire sa réalisation.

1.2 Présentation de l'organisme d'accueil

MIC Consulting Group est une société créée en 2008 par des chercheurs spécialisés dans différents domaines, notamment le droit, le management et l'informatique, en collaboration avec des hommes d'affaires.

L'entreprise intervient dans plusieurs activités de conseil et d'accompagnement. Sa stratégie repose sur la mobilisation de compétences techniques et humaines complémentaires afin de proposer des solutions adaptées aux besoins de ses clients.

1.2.1 Fiche signalétique

Le tableau 1.1 regroupe les principales informations administratives et les coordonnées de l'organisme d'accueil.

Tableau 1.1 – *Fiche signalétique de MIC Consulting Group*

Élément	**Information**
Raison sociale	MIC Consulting Group
Statut juridique	Société à responsabilité limitée (SARL)
Date de création	1er janvier 2008
Registre de commerce	24400
Identifiant fiscal	06544521
Patente	830098
Adresse	Avenue My Abdellah, Résidence Safaa, Bloc B, 3e étage, appartement 12, Marrakech
Téléphone	(+212) 24 29 00 66
Fax	(+212) 24 29 00 69
Adresse électronique	admin@micconsulting.ma
Site Web	www.micconsulting.ma

1.2.2 Domaines d'activité

MIC Consulting Group intervient dans plusieurs domaines complémentaires. Ses principales activités sont les suivantes :

— les études économiques ;

— le management et l'ingénierie ;

— le conseil juridique et fiscal ;

— les services d'intérim ;

— les travaux de comptabilité.

Cette diversité permet à l'entreprise de répondre à des besoins variés et de réunir, selon la nature des missions, des profils issus de plusieurs spécialités.

1.2.3 Une équipe pluridisciplinaire

MIC Consulting Group dispose d'une équipe composée de chercheurs, de consultants, d'ingénieurs et de techniciens confirmés dans leurs domaines respectifs. Ces profils se réunissent au sein de groupes de travail pluridisciplinaires afin d'analyser les besoins des clients et de proposer des solutions adaptées.

Cette organisation favorise la collaboration entre les différentes spécialités de l'entreprise. Elle permet également d'associer les compétences en management, en ingénierie, en droit et en comptabilité dans le cadre de missions nécessitant plusieurs domaines d'expertise.

L'entreprise accorde par ailleurs une importance particulière à la participation de ses collaborateurs aux décisions et aux orientations des projets. Cette démarche contribue à encourager la créativité, l'échange d'idées et la recherche de solutions pertinentes.

1.2.4 Organisation interne

MIC Consulting Group est organisée autour d'une direction générale supervisant quatre principaux domaines : le management, l'ingénierie, le juridique et la comptabilité.

La figure 1.1 présente l'organisation interne de l'entreprise.

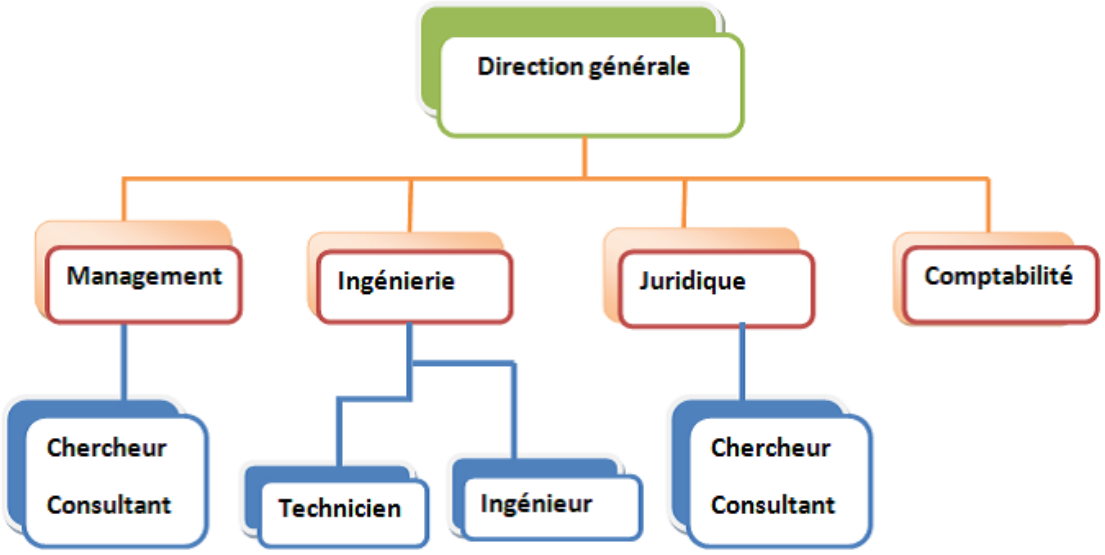


Figure 1.1 – *Organigramme de MIC Consulting Group*

Le pôle de management réunit notamment des chercheurs et des consultants. Le domaine de l'ingénierie mobilise des techniciens et des ingénieurs, tandis que le pôle juridique s'appuie sur des chercheurs et des consultants spécialisés. Le domaine de la comptabilité complète cette organisation en assurant les prestations relevant de sa spécialité.

1.2.5 Principales références

Au cours de ses activités, MIC Consulting Group a collaboré avec plusieurs organismes et institutions. Parmi les principales références mentionnées dans la documentation de l'entreprise figurent :

— le ministère de l'Intérieur ;

— le ministère de la Santé ;

— le ministère de l'Éducation nationale ;

— le Centre Hospitalier Universitaire de Marrakech ;

— le Conseil municipal de Marrakech ;

— le Conseil d'arrondissement de la Ménara ;

— les Forces Armées Royales.

Ces références témoignent de la diversité des structures accompagnées par MIC Consulting Group et de son intervention auprès d'organismes publics et institutionnels.

1.3 Contexte du stage

Le stage a été réalisé au sein du service informatique de MIC Consulting Group. Il s'inscrit dans le cadre du projet de fin d'année et a permis de mettre en pratique les connaissances acquises durant la formation en développement digital et systèmes d'information.

Le travail réalisé porte sur la conception et le développement d'une plateforme d'apprentissage numérique appelée Khotwa. Cette solution vise à centraliser la gestion des formations, des contenus pédagogiques, des participants, des évaluations et des classes virtuelles.

Le projet couvre plusieurs dimensions du développement d'un système d'information, notamment l'analyse des besoins, la modélisation, le développement du backend, la réalisation des applications Web et mobile, la sécurisation des accès, la gestion des données, le stockage des ressources et la validation de la qualité du code.

1.4 Problématique

La gestion numérique d'un parcours de formation nécessite plusieurs mécanismes complémentaires. La simple publication de documents ou de vidéos ne permet pas de répondre à l'ensemble des besoins liés à l'apprentissage en ligne.

Une plateforme complète doit notamment permettre de structurer les formations, de contrôler les accès, de suivre la progression, d'organiser les évaluations, de faciliter les échanges entre les participants et les formateurs et de superviser les opérations administratives.

La problématique du projet peut ainsi être formulée de la manière suivante :

> *Comment concevoir et réaliser une plateforme E-learning sécurisée, accessible depuis le Web et les appareils mobiles, permettant de centraliser la gestion des formations et d'assurer un suivi cohérent des participants, des formateurs et des activités pédagogiques ?*

Pour répondre à cette problématique, la solution doit proposer une architecture modulaire, protéger les données et les ressources pédagogiques et offrir une expérience adaptée aux responsabilités de chaque utilisateur.

1.5 Objectifs du projet

1.5.1 Objectif général

L'objectif général consiste à concevoir et à développer une plateforme E-learning multiplateforme permettant de gérer les principales étapes d'un parcours de formation en ligne, depuis la découverte d'une formation jusqu'à l'obtention d'un certificat.

La solution doit être accessible à travers une application Web et une application mobile, tout en utilisant un backend commun pour centraliser les règles métier et les données.

1.5.2 Objectifs spécifiques

Les objectifs spécifiques du projet sont les suivants :

— mettre en place une authentification sécurisée ;
— gérer les accès selon les rôles des utilisateurs ;
— proposer un catalogue de formations consultable et interrogeable ;
— permettre aux formateurs de créer et d'organiser leurs formations ;
— structurer les parcours en modules, chapitres et ressources pédagogiques ;
— permettre aux participants de s'inscrire aux formations ;
— assurer le suivi de la progression pédagogique ;
— intégrer des quiz associés aux modules et aux formations ;
— générer des certificats après la satisfaction des conditions de réussite ;
— permettre la planification et la participation aux classes virtuelles ;
— centraliser les notifications liées aux événements de la plateforme ;
— proposer une application mobile adaptée au parcours du participant ;
— permettre la consultation hors connexion de certaines ressources ;
— offrir aux administrateurs des outils de supervision et de contrôle ;
— garantir la sécurité, la maintenabilité et l'évolutivité de la solution.

1.6 Acteurs concernés

Khotwa prend principalement en charge trois catégories d'utilisateurs :

Participant : il consulte le catalogue, s'inscrit aux formations, accède aux contenus pédagogiques, suit sa progression, passe les quiz, participe aux classes virtuelles et récupère ses certificats.

Formateur : il crée les formations, organise les modules et les chapitres, ajoute les ressources pédagogiques, prépare les quiz, planifie les classes virtuelles et suit les participants.

Administrateur : il supervise le fonctionnement de la plateforme, gère les utilisateurs, valide les candidatures des formateurs, modère les avis et consulte les informations d'audit.

Le système exécute également certains traitements automatiques, tels que l'envoi des courriers électroniques, la correction des questionnaires à choix multiple, la mise à jour de la progression, la génération des certificats et l'envoi des notifications.

1.7 Périmètre fonctionnel

Le périmètre retenu couvre les principales fonctionnalités nécessaires à la gestion d'une plateforme de formation en ligne.

1.7.1 Fonctionnalités incluses

Les fonctionnalités intégrées au projet comprennent :

— la gestion des comptes et de l'authentification ;
— la gestion des rôles et des autorisations ;
— la consultation et la recherche des formations ;
— la gestion des formations et de leurs contenus ;
— la gestion des inscriptions ;
— la lecture des vidéos et des documents PDF ;
— le suivi de la progression ;
— la gestion des quiz et des résultats ;
— la génération des certificats ;
— la gestion des classes virtuelles ;
— l'envoi des notifications et des courriers électroniques ;
— la supervision administrative ;

— l'accès depuis une application Web ;

— l'accès depuis une application mobile ;

— la consultation hors connexion des ressources autorisées.

1.7.2 Limites du périmètre

Certaines fonctionnalités restent limitées dans la version actuelle du projet. Le paiement est simulé afin de valider le parcours fonctionnel sans traiter de véritables données bancaires. Une utilisation en production nécessiterait l'intégration d'une passerelle de paiement certifiée.

Certaines opérations de l'application mobile nécessitent également une connexion au backend, notamment la soumission des quiz, la synchronisation de la progression, l'achat d'une formation et l'accès aux classes virtuelles.

Le déploiement public à grande échelle et la publication de l'application mobile sur les magasins d'applications ne font pas partie du périmètre actuel.

1.8 Méthodologie de gestion du projet

Pour organiser la réalisation de Khotwa, une démarche Agile inspirée de la méthodologie Scrum a été adoptée. Cette approche permet de diviser un projet complexe en plusieurs cycles courts et progressifs, appelés sprints. Chaque sprint est consacré à un ensemble cohérent de fonctionnalités et aboutit à une version pouvant être vérifiée.

L'utilisation d'une démarche itérative a permis de développer progressivement les différentes composantes de Khotwa, de vérifier régulièrement leur fonctionnement et d'intégrer les ajustements identifiés au cours de la réalisation.

La figure 1.2 présente le fonctionnement général d'un cycle Scrum.

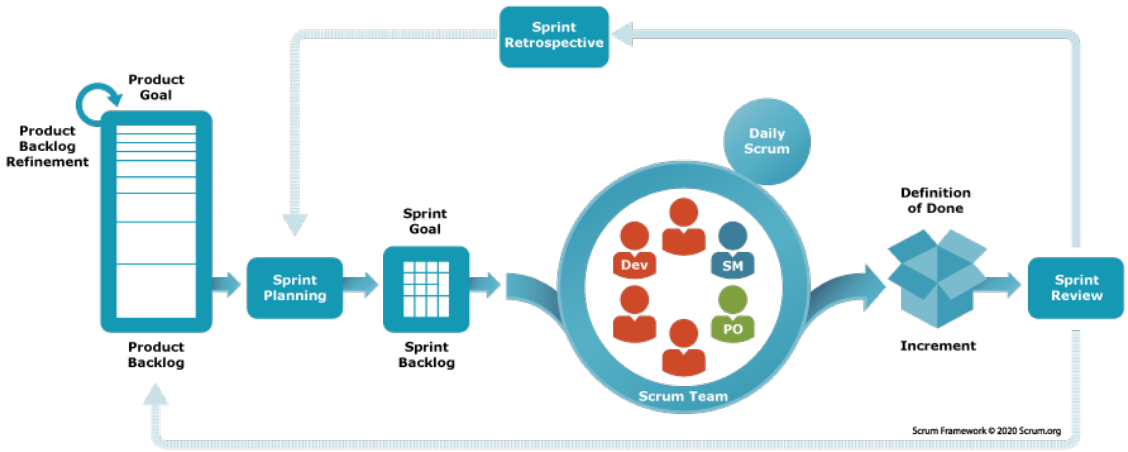


Figure 1.2 – *Cycle de développement selon la méthodologie Scrum*

Comme le montre la figure 1.2, les besoins sont d'abord regroupés et priorisés dans un Product Backlog. Les éléments sélectionnés sont ensuite préparés pendant la planification du sprint. L'équipe réalise les fonctionnalités retenues pendant le sprint, puis présente les résultats obtenus lors de la revue. Une rétrospective permet enfin d'identifier les améliorations à appliquer au cycle suivant.

Dans le cadre de ce projet, Scrum a servi principalement de cadre d'organisation itératif. Son application a été adaptée aux conditions du stage et ne correspond pas nécessairement à la mise en place de tous les rôles et de toutes les cérémonies d'une équipe Scrum complète.

1.8.1 Organisation du travail en sprints

Le développement a été réparti en plusieurs étapes correspondant aux principaux modules de la plateforme :

— préparation de l'environnement technique et de l'architecture ;
— mise en place de l'authentification et de la gestion des utilisateurs ;
— réalisation du catalogue et de la gestion des formations ;
— développement du parcours pédagogique et de la progression ;
— intégration des quiz et des certificats ;
— réalisation des classes virtuelles et des notifications ;
— développement des espaces formateur et administrateur ;
— développement de l'application mobile ;

— réalisation des tests et amélioration de la qualité.

Pour chaque étape, les fonctionnalités attendues ont été identifiées, développées puis vérifiées avant de poursuivre avec les éléments suivants. Cette progression a permis de limiter les dépendances entre les modules et de détecter les problèmes au fur et à mesure de la réalisation.

Le diagramme de Gantt présenté à la figure 1.3 synthétise la planification des sept sprints, depuis les fondations et la sécurité jusqu'à la qualité et à la stabilisation de la plateforme. Il permet de visualiser leur enchaînement ainsi que la durée prévue pour chaque étape.

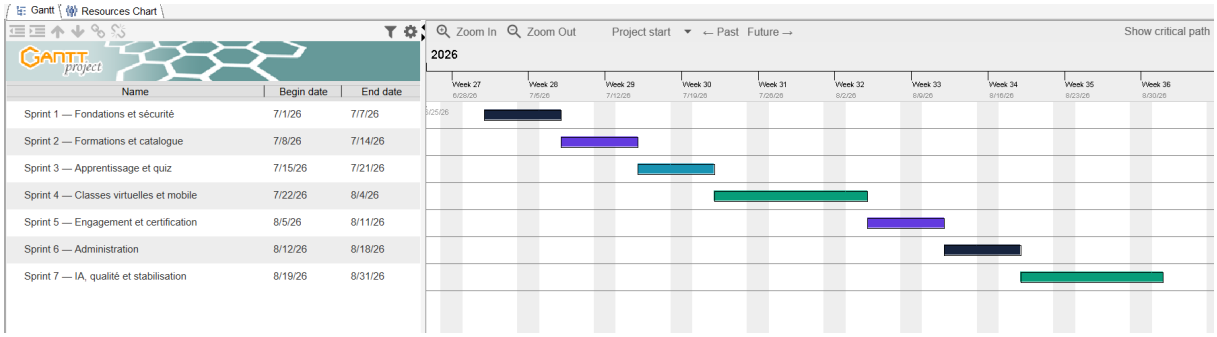


Figure 1.3 – *Diagramme de Gantt des sprints du projet Khotwa*

1.8.2 Gestion du Product Backlog

Les besoins fonctionnels ont été organisés dans un Product Backlog. Celui-ci regroupe les fonctionnalités attendues sous la forme d'éléments à réaliser, classés selon leur importance et leurs dépendances techniques.

Les éléments prioritaires, notamment l'authentification, la gestion des rôles, les formations et les inscriptions, ont été traités avant les fonctionnalités complémentaires telles que les notifications, les classes virtuelles et le fonctionnement hors connexion.

Le Product Backlog a évolué au cours du projet afin de prendre en considération les difficultés techniques rencontrées et les améliorations identifiées pendant le développement.

1.8.3 Suivi et validation des itérations

À la fin de chaque étape, les fonctionnalités développées ont été contrôlées afin de vérifier leur conformité avec les besoins définis. Les anomalies détectées ont été corrigées avant l'intégration des fonctionnalités suivantes.

Dans le cadre de ce projet, j'ai assuré seul la conception et le développement de la plateforme. À la fin de chaque sprint, j'ai présenté à mon encadrant les fonctionnalités réalisées et le livrable préparé afin de recueillir ses remarques et de valider les résultats.

Après la réalisation de chaque sprint et la préparation de son livrable, une réunion de suivi a été organisée pour présenter le travail accompli et recueillir les remarques sur les fonctionnalités livrées. Ces échanges ont permis de valider les résultats obtenus et d'identifier les éventuels ajustements à prendre en compte lors du sprint suivant. Cette pratique s'apparente à une revue de sprint, dans le cadre adapté du stage.

Le suivi régulier du projet a notamment porté sur :

— l'avancement des fonctionnalités ;

— la communication entre les applications clientes et le backend ;

— la cohérence des données ;

— le respect des règles d'autorisation ;

— la validation des principaux parcours utilisateurs ;

— la qualité et la maintenabilité du code.

Cette démarche a favorisé une évolution progressive de Khotwa et une validation continue des différents composants de la solution.

1.9 Conclusion

Ce chapitre a permis de présenter MIC Consulting Group, ses domaines d'activité, son équipe pluridisciplinaire, son organisation interne et ses principales références. Il a également défini le contexte du stage ainsi que le cadrage du projet Khotwa.

La problématique, les objectifs, les acteurs concernés et le périmètre fonctionnel ont ainsi été précisés. La démarche Agile inspirée de Scrum a permis d'organiser le développement sous la forme d'itérations successives et de valider progressivement les fonctionnalités réalisées.

Ces éléments constituent la base nécessaire à l'analyse détaillée des besoins et à la conception de la plateforme, qui feront l'objet du chapitre suivant.

Chapitre 2

Analyse et conception

2.1 Introduction

La phase d'analyse et de conception constitue une étape essentielle dans le développement d'une application informatique. Elle permet d'identifier les besoins des utilisateurs, de définir les fonctionnalités attendues et de représenter les principales interactions avec le système avant de commencer sa réalisation technique.

Dans le cadre de Khotwa, cette phase s'appuie sur l'étude des besoins des trois acteurs principaux de la plateforme : le participant, le formateur et l'administrateur.

Ce chapitre présente les besoins fonctionnels et non fonctionnels de Khotwa, les acteurs du système ainsi que les diagrammes de cas d'utilisation, les séquences de soumission d'un quiz et de génération d'un certificat, et le diagramme de classes.

2.2 Analyse des besoins

L'analyse des besoins permet de déterminer les services que la plateforme doit proposer ainsi que les contraintes qu'elle doit respecter. Les besoins identifiés sont répartis en deux catégories : les besoins fonctionnels, qui décrivent les fonctionnalités attendues, et les besoins non fonctionnels, qui concernent principalement la sécurité, la performance, la disponibilité et la facilité d'utilisation.

2.2.1 Besoins fonctionnels

Les besoins fonctionnels décrivent les opérations que les différents acteurs doivent pouvoir effectuer au sein de Khotwa. Le tableau 2.1 présente une synthèse des principaux besoins fonctionnels de la plateforme.

Tableau 2.1 – *Principaux besoins fonctionnels de Khotwa*

Module	**Fonctionnalités principales**
Authentification	Création de compte, connexion sécurisée, gestion de la session et contrôle des accès selon le rôle de l'utilisateur.
Catalogue	Consultation du catalogue, recherche d'une formation et affichage des informations détaillées.
Formations	Création et gestion des formations, des modules, des chapitres et des ressources pédagogiques.
Inscriptions	Inscription d'un participant à une formation et consultation du nombre de participants inscrits.
Ressources	Ajout et consultation de vidéos, de documents PDF et de liens vers des contenus externes.
Progression	Suivi de l'avancement du participant et reprise du contenu pédagogique.
Quiz	Création des quiz, passage des évaluations, correction automatique des QCM et calcul du résultat.
Certificats	Validation de la réussite, génération du certificat et notification du participant.
Classes virtuelles	Planification des séances, génération du lien Jitsi et notification des participants.
Administration	Gestion des utilisateurs, validation des candidatures des formateurs, consultation des audits, modération des avis et supervision des activités.
Notifications	Envoi de notifications internes et de courriers électroniques de confirmation, d'invitation et de rappel.

2.2.2 Besoins non fonctionnels

En complément des fonctionnalités métier, Khotwa doit respecter plusieurs exigences non fonctionnelles afin de garantir la qualité globale de la plateforme.

— **Sécurité :** protéger les comptes, les données personnelles et les ressources pédagogiques grâce à une authentification sécurisée, un contrôle d'accès par rôles et une gestion rigoureuse des autorisations.

— **Performance :** assurer un temps de réponse acceptable lors de la consultation du catalogue, du chargement des formations et de l'exécution des opérations courantes.

— **Disponibilité :** garantir l'accès aux principales fonctionnalités de la plateforme et limiter les interruptions de service.

— **Compatibilité :** proposer une expérience cohérente depuis l'application Web et l'application mobile.

— **Ergonomie :** offrir des interfaces simples, compréhensibles et adaptées aux différents profils d'utilisateurs.

— **Maintenabilité :** adopter une organisation modulaire facilitant la correction des anomalies et l'ajout de nouvelles fonctionnalités.

— **Intégrité des données :** préserver la cohérence des utilisateurs, des formations, des inscriptions, de la progression, des quiz et des certificats.

— **Traçabilité :** conserver les événements importants afin de faciliter le suivi administratif et la consultation des opérations d'audit.

— **Évolutivité :** permettre l'intégration future de nouveaux services pédagogiques sans remettre en cause l'ensemble de la solution.

2.3 Identification des acteurs

Khotwa repose sur trois acteurs principaux : le participant, le formateur et l'administrateur. Le tableau 2.2 présente ces acteurs et résume leurs principaux rôles au sein de la plateforme.

Tableau 2.2 – *Acteurs principaux de la plateforme Khotwa*

Acteur	**Rôle dans le système**
Participant	Consulte les formations, s'inscrit aux contenus disponibles, suit les ressources pédagogiques, participe aux classes virtuelles, passe les quiz et télécharge ses certificats.
Formateur	Crée et gère ses formations, ajoute les ressources pédagogiques, prépare les quiz, consulte les inscriptions et les résultats, puis organise les classes virtuelles.
Administrateur	Supervise la plateforme, gère les utilisateurs, traite les candidatures des formateurs, consulte les audits et modère les avis signalés.

2.4 Diagramme UML

Pour modéliser Khotwa, nous utilisons UML (*Unified Modeling Language*), un langage de modélisation graphique qui permet de représenter un système sous différents points de vue [1]. Les diagrammes de cas d'utilisation décrivent les interactions entre les acteurs et la plateforme, les diagrammes de séquence montrent l'enchaînement des échanges au cours d'un scénario, et le diagramme de classes présente la structure des principales entités et leurs relations. Ces représentations complémentaires facilitent la compréhension et la conception de la solution avant sa réalisation.

2.5 Diagrammes de cas d'utilisation

Les diagrammes de cas d'utilisation permettent de représenter les services proposés par Khotwa et les interactions entre les acteurs et la plateforme. Pour améliorer la lisibilité, les cas d'utilisation sont présentés séparément pour chaque acteur.

2.5.1 Cas d'utilisation du participant

Le participant représente l'utilisateur principal de la plateforme du point de vue de l'apprentissage. Il peut créer un compte, consulter le catalogue, rechercher une formation et afficher ses informations détaillées. Après authentification, il peut s'inscrire à une formation, accéder aux ressources pédagogiques, participer aux classes virtuelles, passer les quiz et télécharger les certificats obtenus.

La plateforme permet également au participant d'échanger avec le conseiller pédagogique lorsque cette fonctionnalité est disponible. Les opérations qui manipulent des informations personnelles ou pédagogiques nécessitent une authentification préalable.

La figure 2.1 présente les principaux cas d'utilisation associés au participant.

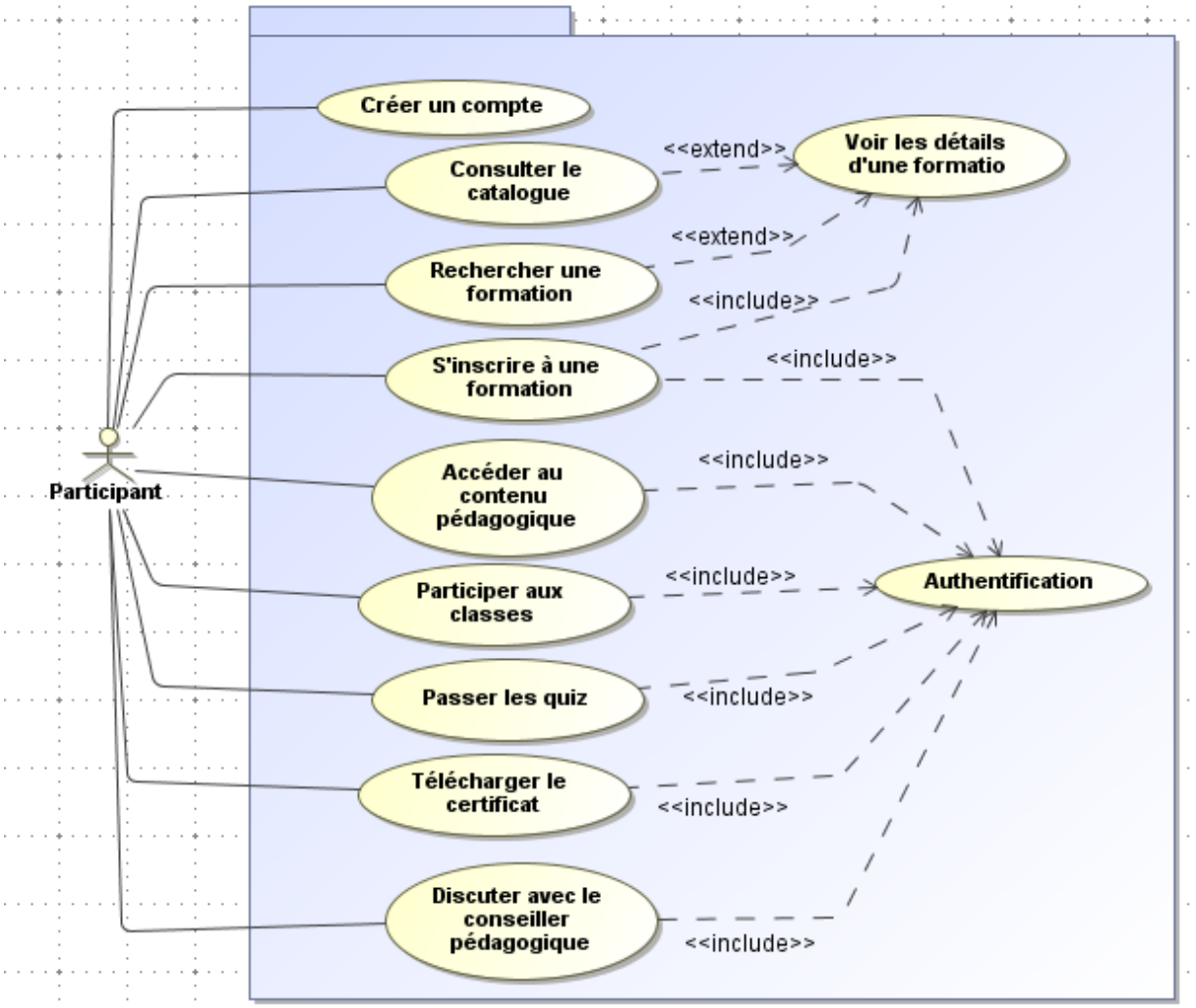


Figure 2.1 – *Diagramme de cas d'utilisation du participant*

Ce diagramme met en évidence le parcours général du participant, depuis la découverte du catalogue jusqu'à l'obtention du certificat. Il montre également que l'authentification constitue une condition nécessaire pour accéder aux fonctionnalités associées au suivi d'une formation.

2.5.2 Cas d'utilisation du formateur

Le formateur est responsable de la préparation et de l'organisation des contenus pédagogiques. Après authentification, il peut gérer ses formations, créer des quiz et ajouter différentes ressources, notamment des vidéos, des documents PDF ou des liens YouTube.

Il peut également consulter le nombre de participants inscrits, suivre le taux de réussite et organiser des classes virtuelles. Ces fonctionnalités lui permettent d'assurer le suivi pédagogique des formations dont il est responsable.

La figure 2.2 présente les principaux cas d'utilisation du formateur.

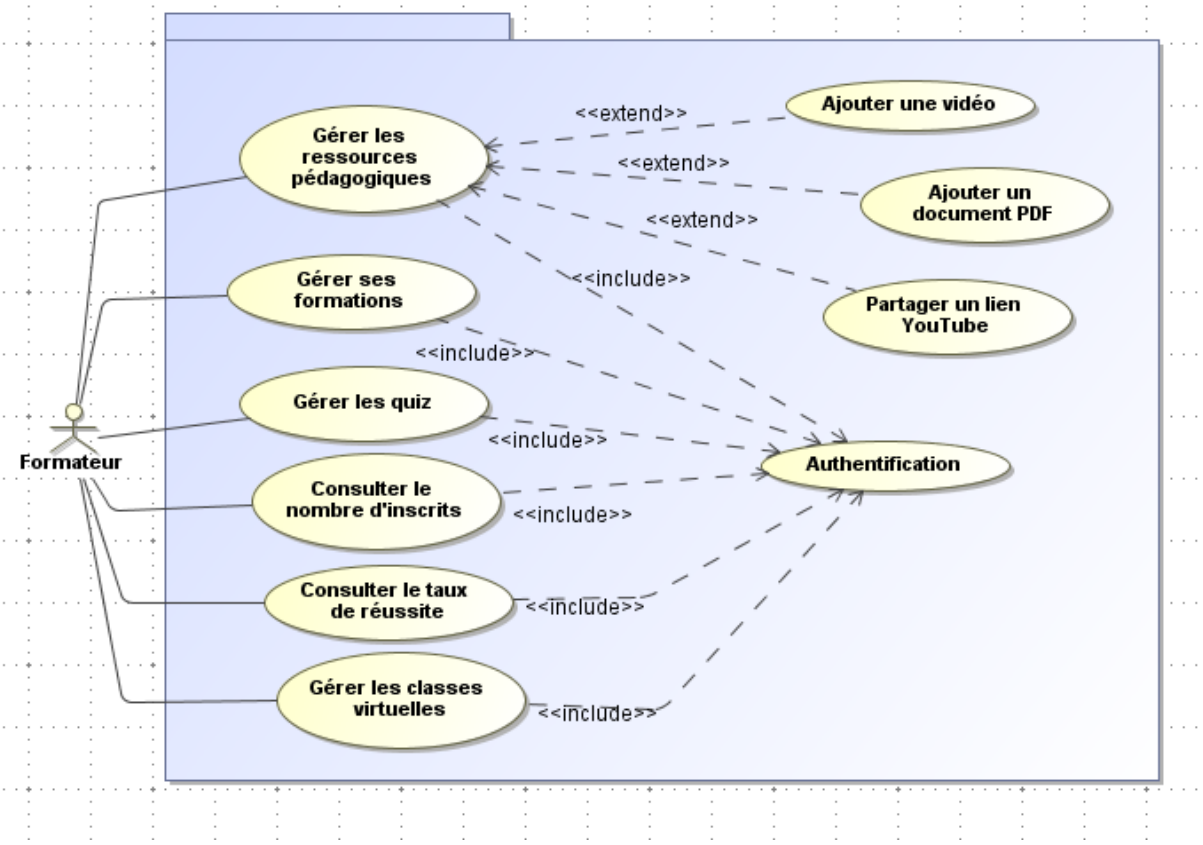


Figure 2.2 – *Diagramme de cas d'utilisation du formateur*

Le diagramme montre que l'ensemble des opérations de gestion du formateur nécessite une authentification. La gestion des ressources pédagogiques peut être complétée par l'ajout de vidéos, de documents PDF et de liens vers des contenus externes.

2.5.3 Cas d'utilisation de l'administrateur

L'administrateur assure la supervision générale de Khotwa. Il peut gérer les utilisateurs, accepter ou refuser les candidatures des formateurs, consulter le journal d'audit, accéder au tableau de bord et modérer les avis signalés.

Ces opérations sont sensibles, car elles concernent l'administration des comptes et la gouvernance de la plateforme. Elles sont donc accessibles uniquement après authentification et vérification du rôle administrateur.

La figure 2.3 présente les principaux cas d'utilisation associés à l'administrateur.

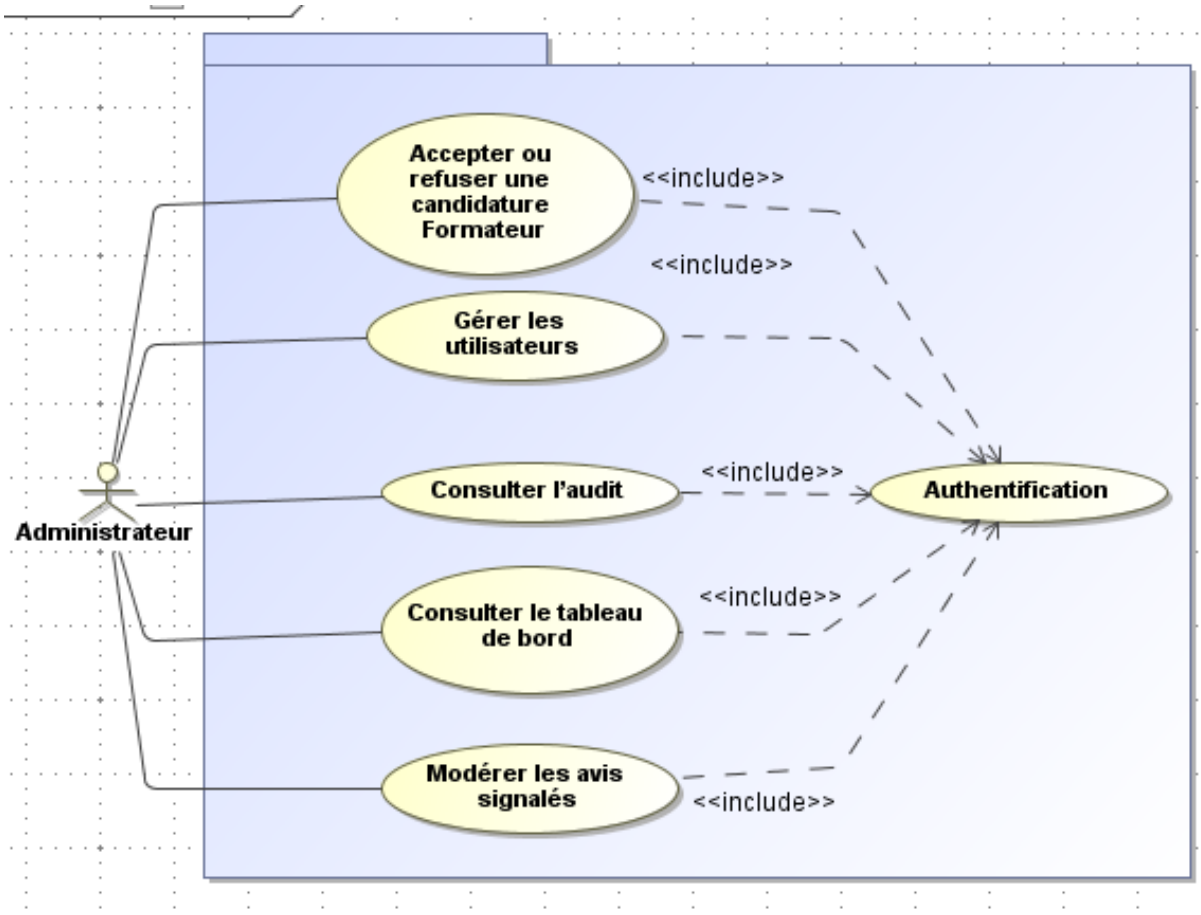


Figure 2.3 – *Diagramme de cas d'utilisation de l'administrateur*

Ce diagramme illustre le rôle central de l'administrateur dans le contrôle et la supervision de Khotwa. Il met notamment en évidence la gestion des utilisateurs, la validation des formateurs et la modération des contenus signalés.

2.6 Description des principaux cas d'utilisation

Afin de compléter les diagrammes précédents, le tableau 2.3 présente une description synthétique de quelques cas d'utilisation importants.

Tableau 2.3 – *Description synthétique des principaux cas d'utilisation*

Cas d'utilisation	**Acteur**	**Description**
Consulter le catalogue	Participant	Le participant parcourt les formations disponibles et consulte leurs informations générales.
S'inscrire à une formation	Participant	Après authentification, le participant sélectionne une formation et confirme son inscription.
Accéder au contenu	Participant	Le participant consulte les modules, les chapitres et les ressources pédagogiques accessibles.
Passer un quiz	Participant	Le participant répond aux questions d'une évaluation et soumet ses réponses pour correction.
Télécharger un certificat	Participant	Le participant télécharge le certificat généré après la validation des conditions de réussite.
Gérer une formation	Formateur	Le formateur crée ou met à jour une formation et organise son contenu pédagogique.
Gérer les ressources	Formateur	Le formateur ajoute des vidéos, des documents PDF ou des liens vers des ressources externes.
Gérer les quiz	Formateur	Le formateur prépare les évaluations associées aux modules ou à la formation.
Gérer les utilisateurs	Administrateur	L'administrateur consulte et administre les comptes enregistrés sur la plateforme.
Valider un formateur	Administrateur	L'administrateur examine une candidature et décide de l'accepter ou de la refuser.
Modérer un avis	Administrateur	L'administrateur examine un avis signalé et applique l'action de modération appropriée.

2.7 Diagrammes de séquence

Les diagrammes de séquence détaillent l'ordre des échanges entre un utilisateur, les interfaces, le backend et la base de données pour un scénario donné. Ils complètent les cas d'utilisation en montrant le déroulement des traitements.

2.7.1 Soumission d'un quiz

La figure 2.4 présente le scénario de soumission d'un quiz. Après l'envoi des réponses par le participant, le contrôleur transmet la demande au service métier. Celui-ci vérifie les prérequis et la limite des tentatives, corrige les réponses, calcule le score et enregistre le résultat avant son affichage dans l'interface.

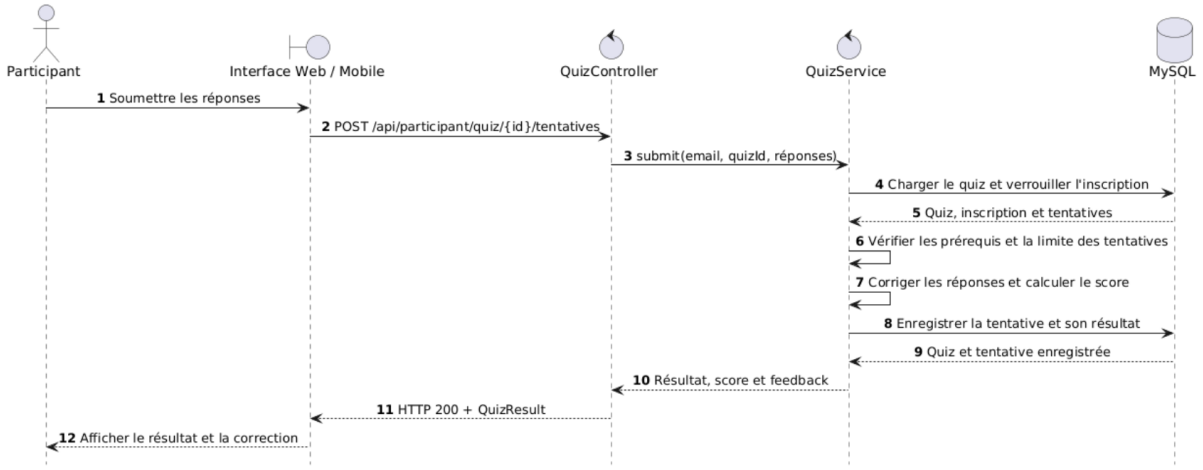


Figure 2.4 – *Diagramme de séquence de la soumission d'un quiz*

2.7.2 Génération d'un certificat

La figure 2.5 présente le processus de génération d'un certificat. Le participant en fait la demande depuis l'interface Web ou mobile, qui transmet une requête authentifiée à l'API. Le service récupère les informations de la formation et de l'inscription, puis vérifie l'éligibilité du participant à partir de sa progression et des résultats des quiz obligatoires. Si les conditions sont satisfaites, un certificat PDF nominatif, accompagné d'un code de vérification, est généré et transmis au participant. Dans le cas contraire, l'API retourne un refus motivé indiquant les éléments restant à valider.

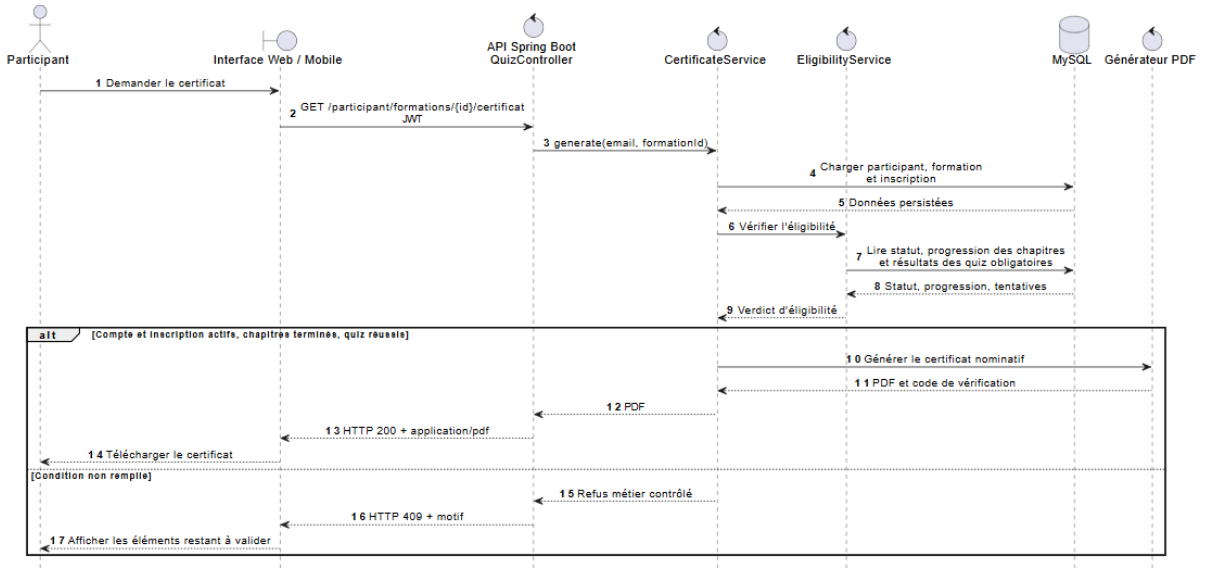


Figure 2.5 – *Diagramme de séquence de la génération d'un certificat*

2.8 Diagramme de classes

Le diagramme de classes représente la structure statique de Khotwa. Il présente les principales entités métier, leurs attributs ainsi que les relations qui assurent la cohérence fonctionnelle de la plateforme.

La figure 2.6 présente le diagramme de classes général de Khotwa.

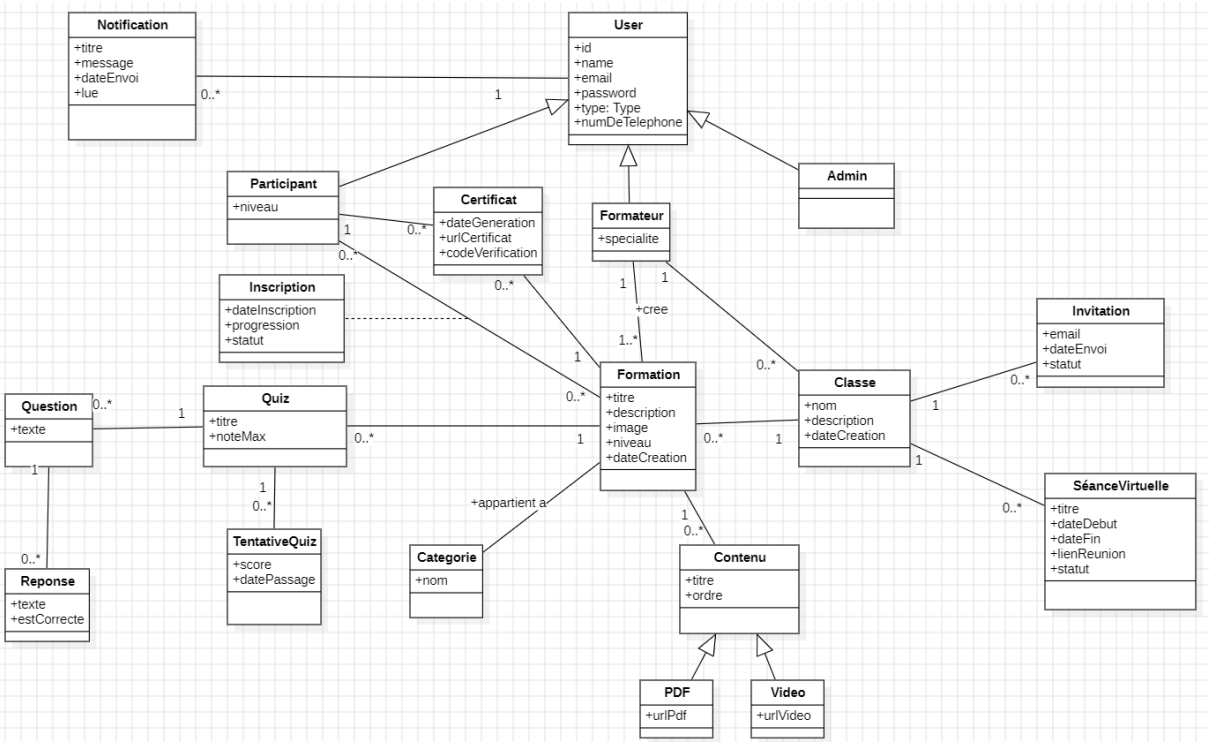


Figure 2.6 – *Diagramme de classes de Khotwa*

Comme le montre la figure 2.6, la classe `User` représente l'entité générale associée aux utilisateurs de la plateforme. Elle contient les informations communes, notamment l'identifiant, le nom, l'adresse électronique, le mot de passe, le type du compte et le numéro de téléphone.

Les classes `Participant`, `Formateur` et `Admin` spécialisent la classe `User`. Cette organisation permet de centraliser les informations communes tout en conservant les propriétés et les responsabilités particulières de chaque profil.

Un participant peut s'inscrire à plusieurs formations. Cette relation est représentée par la classe `Inscription`, qui conserve la date d'inscription, la progression et le statut du parcours. Une formation peut ainsi accueillir plusieurs participants, tandis qu'un participant peut suivre plusieurs formations.

Le formateur peut créer et gérer plusieurs formations. Chaque formation contient des informations telles que le titre, la description, l'image, le niveau et la date de création. Elle appartient également à une catégorie permettant d'organiser le catalogue et de faciliter la recherche.

Une formation est composée de plusieurs contenus pédagogiques ordonnés. La classe générale `Contenu` est spécialisée par les classes `PDF` et `Video`. Cette spécialisation permet de représenter différents types de ressources tout en conservant une structure commune.

Les évaluations sont représentées par les classes `Quiz`, `Question` et `Reponse`. Une formation peut contenir plusieurs quiz, chaque quiz étant composé de plusieurs questions. Une question possède plusieurs propositions de réponse, dont certaines peuvent être indiquées comme correctes.

La classe `TentativeQuiz` représente chaque passage d'un quiz et conserve le score obtenu ainsi que la date de la tentative.

Lorsqu'un participant satisfait les conditions de réussite d'une formation, un certificat peut être généré. La classe `Certificat` conserve notamment la date de génération, l'adresse du document et son code de vérification.

La gestion des activités synchrones repose sur les classes `Classe`, `SéanceVirtuelle` et `Invitation`. Une classe peut regrouper plusieurs séances virtuelles et plusieurs invitations. Les séances contiennent les dates de début et de fin, le lien de réunion et le statut de la session.

Enfin, la classe `Notification` représente les messages adressés aux utilisateurs. Elle conserve le titre, le contenu du message, la date d'envoi et l'état de lecture. Cette entité permet d'informer les utilisateurs des principaux événements survenus dans la plateforme.

2.9 Conclusion

Dans ce chapitre, nous avons présenté l'analyse fonctionnelle de Khotwa à travers l'identification des besoins fonctionnels et non fonctionnels, la description des acteurs ainsi que les diagrammes de cas d'utilisation du participant, du formateur et de l'administrateur. Cette analyse permet de définir clairement les responsabilités de chaque acteur et les principales fonctionnalités attendues de la plateforme.

Les diagrammes de séquence de la soumission d'un quiz et de la génération d'un certificat complètent cette analyse en décrivant le déroulement de deux traitements importants. Le diagramme de classes présente la structure des entités métier. Le chapitre suivant sera consacré à l'architecture technique de Khotwa et aux technologies retenues pour assurer la réalisation de la solution Web, mobile et backend.

Chapitre 3

Architecture technique et technologies utilisées

3.1 Introduction

Après avoir présenté les besoins et la conception fonctionnelle de Khotwa, il convient de définir l'architecture technique permettant de concrétiser la solution. Le choix d'une architecture adaptée est essentiel pour assurer la séparation des responsabilités, la sécurité des échanges, la maintenabilité du code et l'évolution progressive de la plateforme.

Khotwa adopte une architecture organisée en plusieurs couches. Elle associe une application Web, une application mobile, un backend centralisé exposant des services REST, une base de données relationnelle et un système de stockage d'objets. La plateforme communique également avec plusieurs services externes, notamment Jitsi pour les classes virtuelles et un service de courrier électronique pour les confirmations, les invitations et les rappels.

Ce chapitre présente l'architecture générale de Khotwa, les différentes couches qui la composent, les mécanismes de sécurité ainsi que les principales technologies utilisées dans sa réalisation.

3.2 Architecture générale de Khotwa

L'architecture de Khotwa est conçue pour prendre en charge trois catégories principales d'utilisateurs : l'administrateur, le formateur et le participant. Chaque acteur accède uniquement aux services correspondant à son rôle et à ses autorisations.

La solution est structurée autour de quatre ensembles principaux :

— la couche présentation, constituée de l'application Web et de l'application mobile ;

— la couche applicative, développée avec Spring Boot ;

— la couche de données, reposant sur MySQL, Spring Data JPA, Hibernate et MinIO ;

— les services externes, notamment Jitsi et le service d'envoi des courriers électroniques.

Les applications clientes communiquent avec le backend par l'intermédiaire d'une API REST. L'authentification est assurée au moyen de jetons JWT, tandis que les notifica-

tions peuvent être transmises aux utilisateurs par des mécanismes adaptés aux interactions en temps réel.

La figure 3.1 présente l'architecture générale de la plateforme Khotwa.

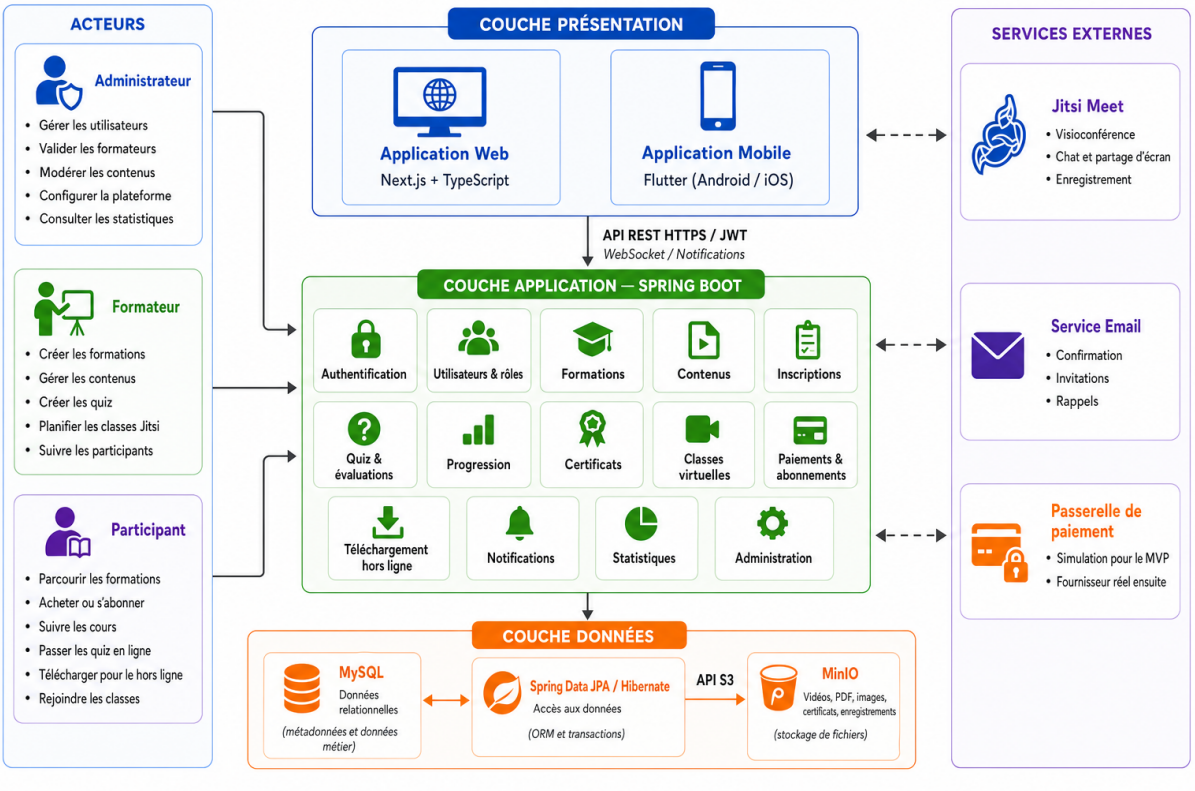


Figure 3.1 – *Architecture générale de la plateforme Khotwa*

Comme le montre la figure 3.1, les interfaces Web et mobile utilisent les services fournis par le backend Spring Boot. Celui-ci centralise les règles métier relatives aux utilisateurs, aux formations, aux contenus pédagogiques, aux inscriptions, aux quiz, à la progression, aux certificats, aux classes virtuelles et aux notifications.

La couche applicative communique avec MySQL pour les données métier structurées et avec MinIO pour les fichiers volumineux tels que les vidéos, les documents PDF, les images et les certificats. Les services externes complètent les fonctionnalités principales sans être directement intégrés aux applications clientes.

3.3 Couche présentation

La couche présentation constitue le point d'interaction entre les utilisateurs et Khotwa. Elle est composée d'une application Web et d'une application mobile. Les deux applications consomment les services fournis par le même backend, ce qui permet de centraliser les règles métier et d'assurer la cohérence des données.

3.3.1 Application Web

L'application Web est développée avec **Next.js**, **React** et **TypeScript** [7, 8, 9]. Elle fournit les interfaces nécessaires aux participants, aux formateurs et aux administrateurs.

Next.js facilite l'organisation des pages et des composants de l'application. React permet de construire des interfaces interactives et réutilisables, tandis que TypeScript renforce la fiabilité du code grâce à la vérification des types.

L'application Web permet notamment :

— la création de compte et l'authentification ;

— la consultation et la recherche des formations ;

— l'accès aux contenus pédagogiques ;

— le suivi de la progression ;

— le passage des quiz ;

— la gestion des formations par les formateurs ;

— la supervision administrative ;

— la consultation des notifications et des tableaux de bord.

Les appels au backend sont réalisés à travers l'API REST. Pour les opérations protégées, l'application transmet le jeton JWT de l'utilisateur afin que le serveur puisse vérifier son identité et ses autorisations.

3.3.2 Application mobile

L'application mobile est développée avec **Flutter** et le langage **Dart** [10, 11]. Ce framework permet de construire une application multiplateforme à partir d'une base de code commune et de cibler principalement les systèmes Android et iOS.

L'application mobile reprend les fonctionnalités essentielles destinées au participant, notamment la consultation du catalogue, l'accès aux formations, le suivi des contenus pédagogiques, le passage des quiz et la consultation de la progression.

Elle prend également en charge la consultation hors ligne de certaines ressources préalablement téléchargées. Cette possibilité améliore l'accessibilité des contenus dans les situations où la connexion Internet est instable ou temporairement indisponible.

Comme l'application Web, l'application Flutter communique avec le backend à travers les services REST sécurisés. Cette organisation évite la duplication des règles métier entre les différentes plateformes clientes.

3.4 Couche applicative

La couche applicative représente le cœur fonctionnel de Khotwa. Elle est développée avec **Java** et **Spring Boot** [2]. Elle reçoit les requêtes provenant des applications clientes, applique les règles métier, contrôle les autorisations et communique avec les systèmes de persistance.

Le backend adopte une architecture monolithique : les fonctionnalités métier sont regroupées dans une seule application Spring Boot. Ce choix simplifie le développement et le déploiement dans le cadre du projet, tout en permettant aux applications Web et mobile de partager la même API. L'organisation interne en couches, décrite ci-dessous, ne doit pas être confondue avec une architecture en microservices : elle sépare les responsabilités au sein de cette application unique.

Le backend est organisé selon une architecture en couches afin de séparer clairement les différentes responsabilités :

— les contrôleurs reçoivent les requêtes HTTP et exposent les points d'accès de l'API ;

— les services contiennent les règles métier de la plateforme ;

— les repositories assurent l'accès aux données ;

— les entités représentent les données persistées ;

— les objets de transfert assurent les échanges entre l'API et les applications clientes ;

— les composants de sécurité contrôlent l'authentification et les autorisations.

Cette séparation facilite la compréhension du code, les tests, la maintenance et l'évolution des différents modules.

3.4.1 Principaux modules métier

Le backend regroupe plusieurs modules correspondant aux principales fonctionnalités de Khotwa.

Tableau 3.1 – *Principaux modules applicatifs de Khotwa*

Module	**Responsabilités principales**
Authentification	Création des comptes, connexion sécurisée, gestion des jetons JWT et contrôle de l'accès aux fonctionnalités.
Utilisateurs et rôles	Gestion des comptes des participants, des formateurs et des administrateurs ainsi que de leurs autorisations.
Formations	Création, modification, publication et consultation des formations.
Contenus pédagogiques	Organisation des modules, des chapitres, des vidéos, des documents PDF et des ressources externes.
Inscriptions	Gestion de l'inscription des participants aux formations et suivi des apprenants inscrits.
Progression	Enregistrement de l'avancement du participant et reprise du contenu pédagogique.
Quiz et évaluations	Création des questions, soumission des réponses, correction automatique et calcul des résultats.
Certificats	Vérification des conditions de réussite, génération et mise à disposition des certificats.
Classes virtuelles	Planification des séances en ligne, génération des liens et interaction avec Jitsi.
Notifications	Gestion des notifications internes, des confirmations, des invitations et des rappels.
Administration	Supervision des utilisateurs, validation des formateurs, modération des avis, audit et statistiques.

3.4.2 Communication par API REST

La communication entre les applications clientes et le backend est assurée par une API REST. Cette API expose les opérations nécessaires à la consultation et à la gestion des ressources de Khotwa.

Les requêtes utilisent les méthodes HTTP adaptées aux opérations effectuées :

— `GET` pour consulter les ressources ;

— `POST` pour créer une nouvelle ressource ;

— `PUT` ou `PATCH` pour mettre à jour une ressource existante ;

— `DELETE` pour supprimer une ressource lorsque l'utilisateur possède les autorisations nécessaires.

Les données échangées sont principalement représentées au format JSON. Cette approche permet aux applications Web et mobile d'utiliser les mêmes services, indépendamment de leur technologie d'implémentation.

3.4.3 Documentation de l'API

L'API est documentée à l'aide d'**OpenAPI** et de **Swagger** [17, 18]. Cette documentation fournit une description structurée des points d'accès, des paramètres, des corps de requête et des réponses retournées par le backend.

L'interface Swagger facilite également les vérifications manuelles pendant le développement. Elle permet d'exécuter les requêtes directement depuis une interface Web et de contrôler le comportement des différents services avant leur intégration dans les applications clientes.

3.5 Couche de données

La couche de données assure la persistance des informations métier ainsi que le stockage des ressources pédagogiques. Khotwa distingue les données structurées, conservées dans MySQL, et les fichiers, stockés dans MinIO.

3.5.1 Base de données MySQL

MySQL est utilisé comme système de gestion de base de données relationnelle [12]. Il assure la conservation des comptes, des rôles, des formations, des modules, des chapitres, des inscriptions, des résultats, de la progression, des certificats, des notifications et des autres informations métier.

Le modèle relationnel permet de définir des associations cohérentes entre les différentes entités et d'appliquer les contraintes nécessaires à l'intégrité des données.

3.5.2 Spring Data JPA et Hibernate

L'accès aux données relationnelles est assuré par **Spring Data JPA** et **Hibernate** [4, 5]. Ces technologies permettent de représenter les tables de la base de données sous forme d'objets Java et de simplifier les opérations de création, de lecture, de mise à jour et de suppression.

Les repositories fournissent les méthodes d'accès nécessaires aux services métier. Cette approche limite la quantité de code répétitif et permet de centraliser les règles de persistance.

3.5.3 Gestion des migrations avec Flyway

Flyway est utilisé pour gérer les évolutions du schéma de la base de données [13]. Chaque modification structurelle est décrite dans une migration versionnée et exécutée dans un ordre déterminé.

Ce mécanisme garantit que les différents environnements utilisent une structure cohérente de la base de données. Il facilite également le suivi des changements apportés au modèle de données pendant les différentes étapes du projet.

3.5.4 Stockage des fichiers avec MinIO

Les ressources volumineuses ne sont pas directement enregistrées dans la base relationnelle. Elles sont stockées dans **MinIO**, une solution de stockage d'objets compatible avec l'API S3 [14].

MinIO est utilisé pour conserver notamment :

— les vidéos pédagogiques ;

— les documents PDF ;

— les images associées aux formations ;

— les ressources téléchargeables ;

— les certificats générés.

Les informations descriptives et les références vers ces fichiers sont conservées dans MySQL. Cette séparation améliore l'organisation du stockage et évite d'enregistrer des fichiers volumineux directement dans les tables relationnelles.

L'accès aux ressources protégées est contrôlé par le backend. Cette approche évite d'exposer directement les fichiers privés et permet de vérifier les autorisations de l'utilisateur avant leur consultation ou leur téléchargement.

3.6 Intégration des services externes

Khotwa communique avec plusieurs services externes afin de compléter ses fonctionnalités principales.

3.6.1 Classes virtuelles avec Jitsi

Jitsi est utilisé pour la mise en place des classes virtuelles [15]. Lorsqu'une séance est planifiée, la plateforme conserve les informations nécessaires et met à la disposition des utilisateurs autorisés un lien leur permettant de rejoindre la réunion.

Cette intégration permet aux formateurs et aux participants d'accéder aux séances de visioconférence depuis la plateforme. Jitsi fournit notamment les fonctions de communication audio et vidéo, de discussion et de partage d'écran nécessaires au déroulement d'une classe virtuelle.

3.6.2 Service de courrier électronique

Le service de courrier électronique est utilisé pour transmettre les communications importantes aux utilisateurs. Il intervient notamment lors de la création d'un compte, de l'inscription à une formation, de la planification d'une classe virtuelle ou de la génération d'un certificat.

Durant le développement, **Mailpit** permet de capturer et de visualiser les courriers électroniques envoyés par l'application sans les transmettre à de véritables destinataires [16]. Cette solution facilite les tests des modèles de messages et des scénarios de notification.

3.6.3 Passerelle de paiement

L'architecture prévoit une séparation entre la plateforme et le service de paiement. Dans la version actuelle du projet, le processus de paiement est simulé afin de valider le parcours fonctionnel sans dépendre d'un fournisseur financier réel.

Cette séparation permettra d'intégrer ultérieurement une passerelle de paiement réelle sans modifier profondément les autres modules de Khotwa. Le backend conservera la responsabilité de vérifier les résultats retournés par le fournisseur avant de confirmer une opération.

3.7 Architecture de sécurité

La sécurité constitue une exigence importante de Khotwa en raison de la présence de données personnelles, de ressources pédagogiques protégées et de fonctionnalités réservées à certains rôles.

3.7.1 Authentification avec Spring Security et JWT

L'authentification est mise en œuvre avec **Spring Security** et des jetons **JWT** [3, 6]. Après une connexion réussie, le backend génère un jeton permettant d'identifier l'utilisateur pendant les échanges avec l'API.

Pour accéder à une ressource protégée, l'application cliente transmet ce jeton dans la requête HTTP. Le backend vérifie sa validité, identifie l'utilisateur et contrôle ses autorisations avant d'exécuter l'opération demandée.

L'utilisation de JWT permet d'adopter une authentification sans session conservée directement sur le serveur. Cette approche est adaptée à une plateforme consommée simultanément par une application Web et une application mobile.

3.7.2 Gestion des rôles et des autorisations

Le contrôle d'accès est fondé sur les rôles de l'utilisateur. Khotwa distingue principalement les rôles suivants :

— **Participant**, pour les fonctionnalités d'apprentissage ;

— **Formateur**, pour la création et le suivi des formations ;

— **Administrateur**, pour la supervision et l'administration de la plateforme.

Le rôle ne suffit pas toujours à autoriser une opération. Le backend vérifie également la relation de l'utilisateur avec la ressource concernée. Par exemple, un formateur ne doit pouvoir modifier que les formations pour lesquelles il possède les droits nécessaires.

3.7.3 Protection des ressources pédagogiques

Les fichiers stockés dans MinIO ne doivent pas être accessibles sans contrôle. Le backend vérifie l'identité de l'utilisateur et ses droits avant de lui permettre de consulter ou de télécharger une ressource protégée.

Ce mécanisme contribue à limiter le partage non autorisé des contenus et garantit que les ressources privées restent accessibles uniquement aux utilisateurs concernés.

3.8 Infrastructure et environnement d'exécution

L'environnement de Khotwa est organisé à l'aide de **Docker** et de **Docker Compose** [19, 20]. Ces technologies permettent de définir et d'exécuter les différents services nécessaires au fonctionnement du projet dans des conteneurs séparés.

L'environnement de développement peut notamment regrouper :

— le serveur MySQL ;

— le service de stockage MinIO ;

— Mailpit pour la vérification des courriers électroniques ;

— les composants applicatifs nécessaires au projet.

La conteneurisation facilite la préparation de l'environnement, limite les différences de configuration entre les machines et simplifie le lancement des dépendances techniques.

3.9 Synthèse des technologies utilisées

Le tableau 3.2 présente les principales technologies retenues pour la réalisation de Khotwa.

Tableau 3.2 – *Principales technologies utilisées dans Khotwa*

Domaine	**Technologie**	**Rôle dans le projet**
Backend	Java et Spring Boot	Développement des services métier et exposition de l'API REST.
Sécurité	Spring Security et JWT	Authentification, autorisation et protection des points d'accès.
Persistance	Spring Data JPA et Hibernate	Correspondance entre les objets Java et les données relationnelles.
Base de données	MySQL	Conservation des données métier de la plateforme.
Migration	Flyway	Versionnement et exécution des modifications du schéma de données.
Application Web	Next.js, React et TypeScript	Création des interfaces Web destinées aux différents utilisateurs.
Application mobile	Flutter	Développement de l'application mobile multiplateforme.
Stockage	MinIO	Stockage des vidéos, des documents, des images et des certificats.
Classes virtuelles	Jitsi	Organisation des séances de formation à distance.
Courrier électronique	Mailpit	Vérification des courriers électroniques dans l'environnement de développement.
Documentation	OpenAPI et Swagger	Description et vérification interactive des services REST.
Infrastructure	Docker Compose	Organisation et exécution des services techniques du projet.

3.10 Justification des choix technologiques

Les technologies sélectionnées répondent aux besoins fonctionnels et techniques de Khotwa. Spring Boot offre un environnement structuré pour développer les règles métier et les services REST. Spring Security et JWT permettent de centraliser la protection des fonctionnalités accessibles depuis les applications Web et mobile.

Next.js, React et TypeScript facilitent la création d'une interface Web dynamique, modulaire et maintenable. Flutter permet de proposer une application mobile multiplateforme tout en partageant une même base de code.

MySQL répond aux besoins de persistance relationnelle, tandis que Spring Data JPA et Hibernate facilitent l'accès aux données. Flyway assure le suivi des évolutions du schéma et MinIO prend en charge les ressources pédagogiques volumineuses. Enfin, Docker Compose simplifie l'organisation de l'environnement technique et favorise la reproductibilité de son exécution.

L'ensemble de ces technologies permet de construire une solution cohérente, sécurisée et évolutive, adaptée aux besoins d'une plateforme E-learning accessible depuis plusieurs types de terminaux.

3.11 Conclusion

Dans ce chapitre, nous avons présenté l'architecture technique de Khotwa et les principales technologies utilisées pour sa réalisation. La solution repose sur une séparation entre la couche présentation, la couche applicative, la couche de données et les services externes. Les applications Web et mobile communiquent avec un backend Spring Boot à travers une API REST sécurisée par Spring Security et JWT.

La persistance des données est assurée par MySQL, Spring Data JPA et Hibernate, tandis que Flyway gère les évolutions du schéma et MinIO assure le stockage des ressources pédagogiques. Jitsi, Mailpit, OpenAPI, Swagger et Docker Compose complètent l'environnement technique de la plateforme. Le chapitre suivant présentera la réalisation concrète de Khotwa et les principaux modules développés.

Chapitre 4

Réalisation de la plateforme Khotwa

4.1 Introduction

Après avoir présenté l'analyse fonctionnelle, la conception et l'architecture technique de Khotwa, ce chapitre expose la réalisation concrète de la solution. Celle-ci comprend une application Web destinée aux participants, aux formateurs et aux administrateurs, ainsi qu'une application mobile principalement consacrée au parcours d'apprentissage du participant.

Les interfaces communiquent avec le backend Spring Boot au moyen d'une API REST sécurisée. Les captures présentées dans ce chapitre proviennent de la version fonctionnelle de Khotwa et illustrent les principaux résultats obtenus.

4.2 Réalisation de l'interface publique

L'interface publique représente le premier point de contact avec Khotwa. Elle permet de découvrir la plateforme, de consulter les formations disponibles et d'accéder aux fonctions d'authentification.

4.2.1 Page d'accueil

La page d'accueil présente l'identité visuelle de Khotwa et sa proposition principale : permettre aux utilisateurs de développer leurs compétences à leur propre rythme. Elle donne un accès direct au catalogue, à l'orientation, à la présentation du fonctionnement de la plateforme, à la connexion et à la création d'un compte.

La figure 4.1 présente la page d'accueil publique.

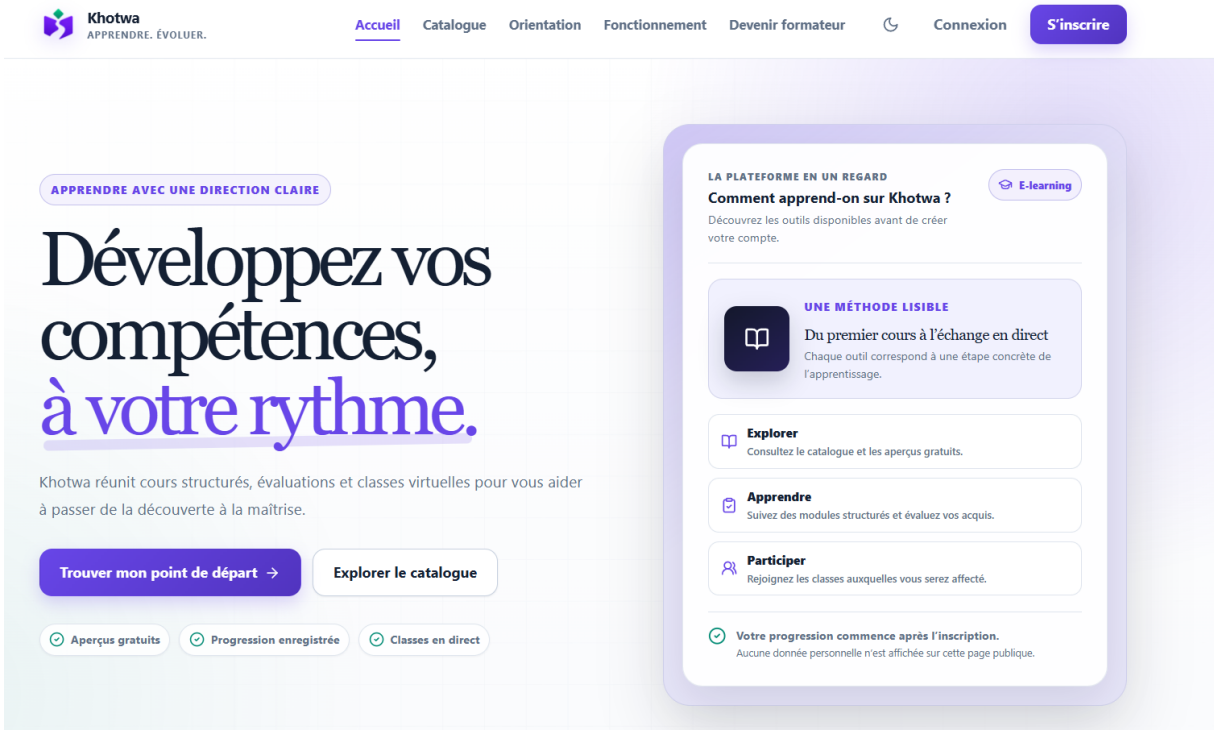


Figure 4.1 – *Page d'accueil de Khotwa*

Comme le montre cette interface, le visiteur peut comprendre rapidement les principales étapes du parcours proposé : explorer les formations, suivre les contenus et participer aux activités pédagogiques.

4.2.2 Catalogue des formations

Le catalogue regroupe les formations publiées sur la plateforme. Une zone de recherche permet de trouver une formation à partir de son titre, de sa catégorie ou d'un mot-clé. Des filtres permettent également de limiter les résultats selon le niveau, la langue ou la catégorie.

Chaque formation est présentée sous la forme d'une carte affichant les principales informations nécessaires à la comparaison des offres disponibles.

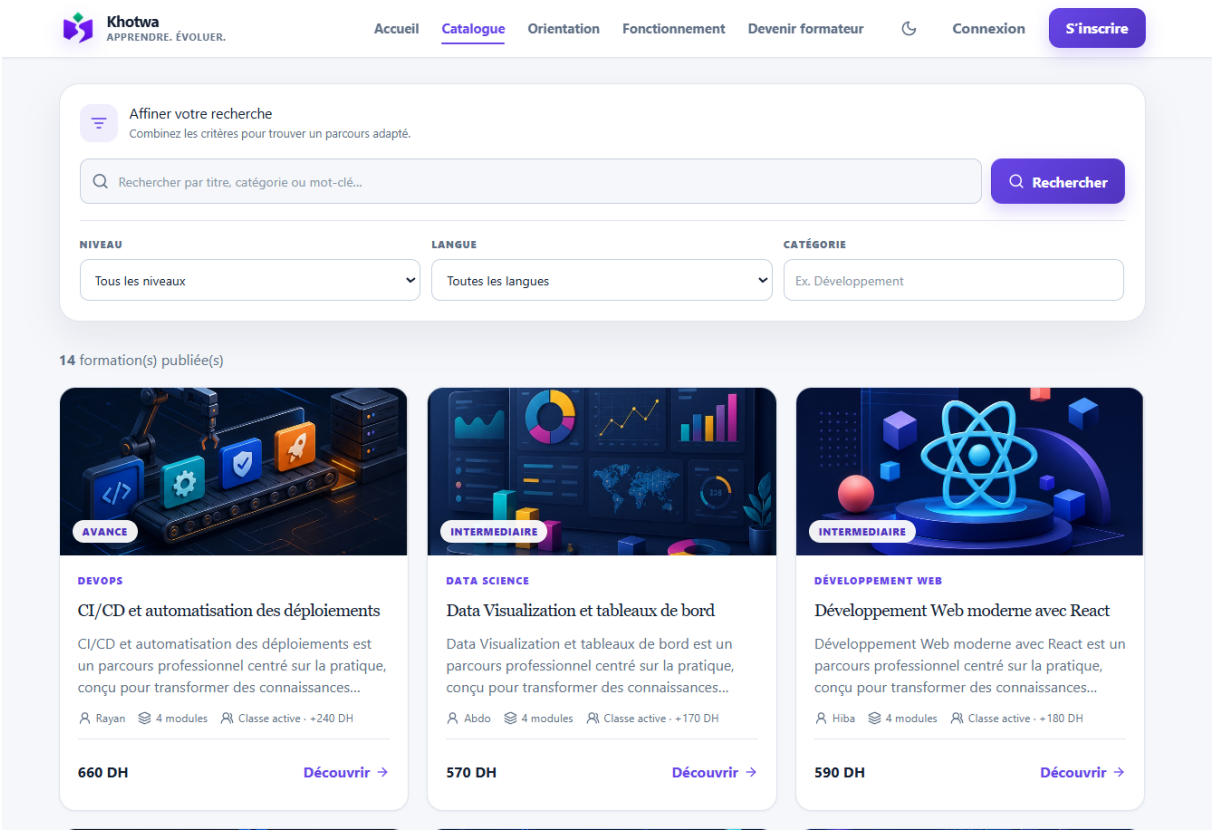


Figure 4.2 – *Catalogue des formations de Khotwa*

À partir de la figure 4.2, l'utilisateur peut sélectionner une formation et accéder à sa page détaillée.

4.2.3 Consultation des détails d'une formation

La page détaillée présente le titre, la catégorie, le niveau, la description, les informations relatives au formateur et l'organisation générale du parcours. Elle permet à l'utilisateur de prendre connaissance du contenu avant de choisir une formule d'accès.

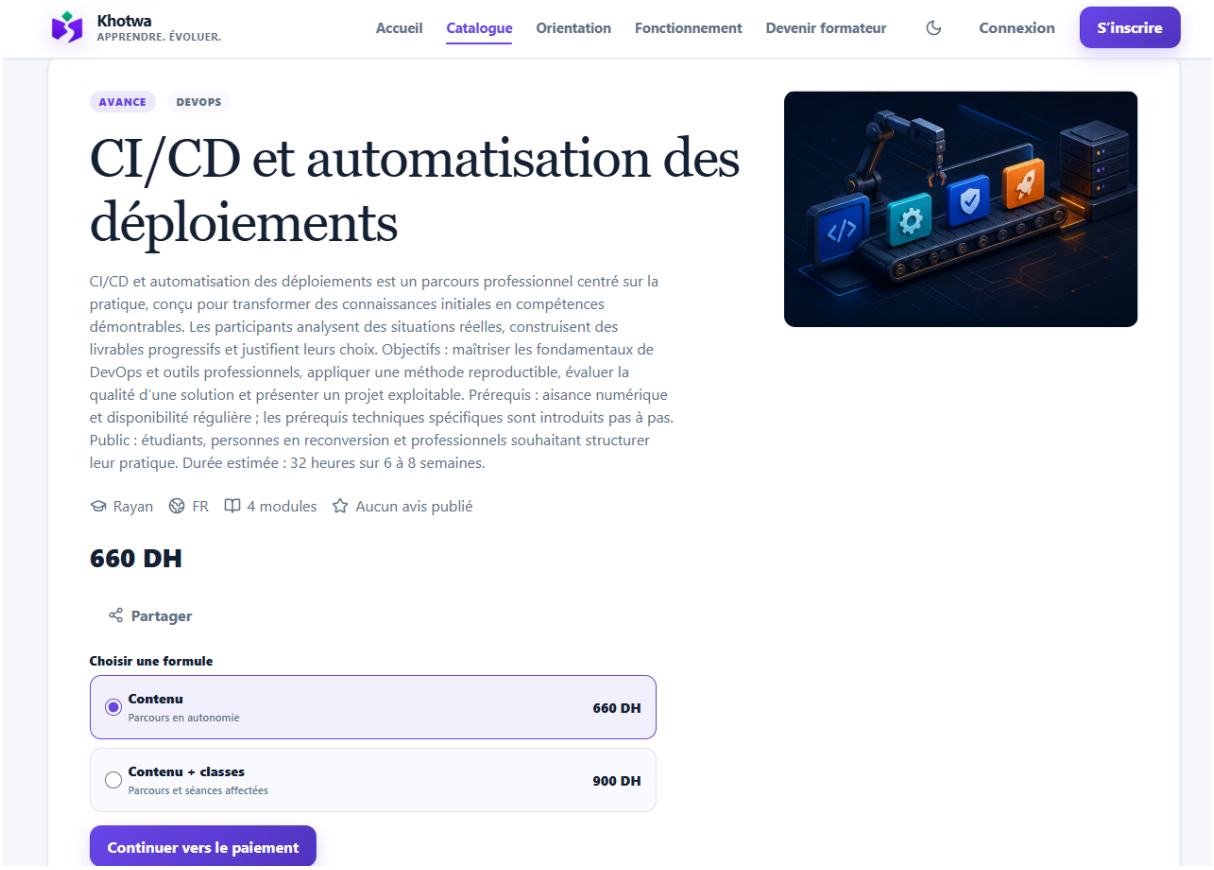


Figure 4.3 – *Consultation des détails d'une formation*

Dans la version actuelle du projet, le paiement est simulé afin de valider le parcours fonctionnel sans dépendre d'une passerelle financière réelle. Les données bancaires de test ne sont ni enregistrées ni transmises au backend.

4.3 Authentification et contrôle d'accès

L'accès aux espaces privés de Khotwa nécessite une authentification. L'utilisateur saisit son adresse électronique et son mot de passe dans l'interface présentée par la figure 4.4.

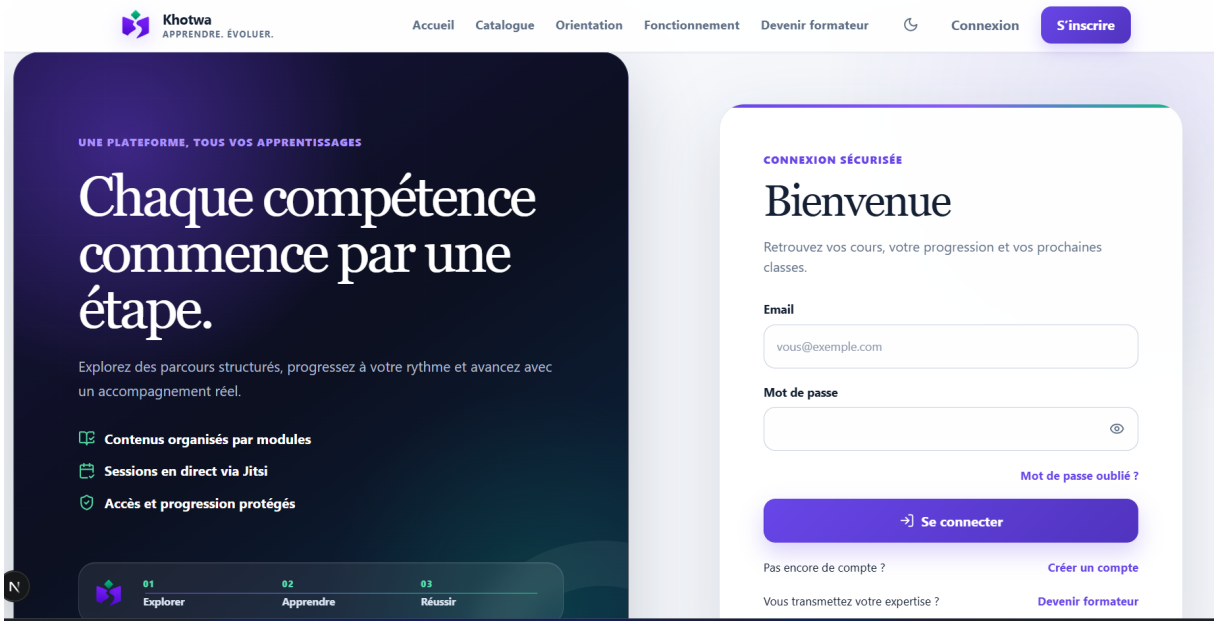


Figure 4.4 – *Interface de connexion sécurisée*

Après validation des identifiants, le backend génère un jeton JWT utilisé pour sécuriser les requêtes suivantes. L'utilisateur est ensuite orienté vers l'espace correspondant à son rôle : participant, formateur ou administrateur.

4.4 Réalisation de l'espace participant

L'espace participant regroupe les fonctions nécessaires au suivi du parcours d'apprentissage : consultation des formations, accès aux contenus, progression, quiz, certificats et notifications.

4.4.1 Tableau de bord du participant

Le tableau de bord présente une vue synthétique de l'activité pédagogique du participant. Il affiche notamment le nombre de formations suivies, la progression globale, les quiz disponibles et la prochaine étape du parcours.

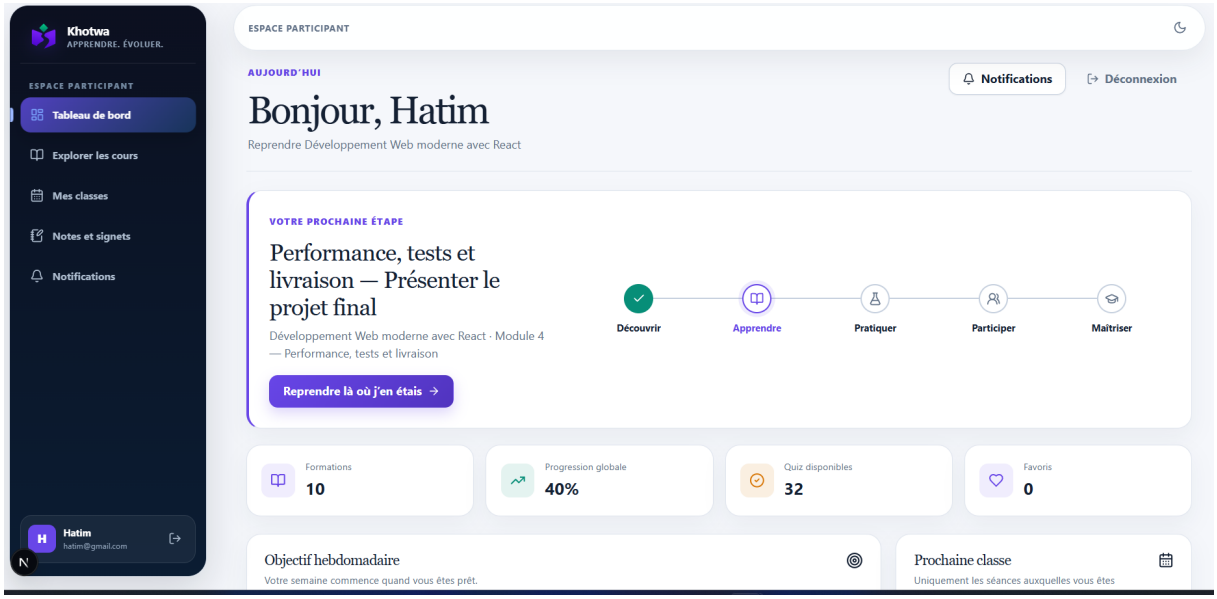


Figure 4.5 – *Tableau de bord du participant*

La zone principale de la figure 4.5 permet au participant de reprendre directement le chapitre où il avait interrompu son apprentissage. Les étapes affichées l'aident également à comprendre sa position dans le parcours.

4.4.2 Consultation des contenus pédagogiques

Le lecteur pédagogique organise les ressources selon une hiérarchie composée de modules et de chapitres. Il distingue les contenus disponibles, terminés ou verrouillés et permet au participant de reprendre son apprentissage au bon emplacement.

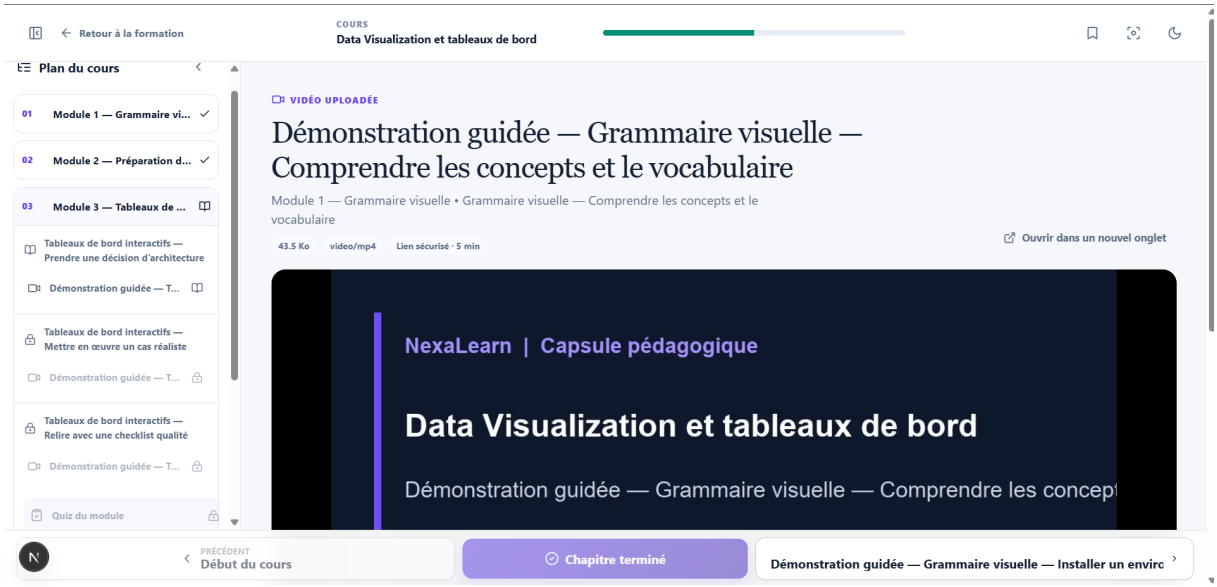


Figure 4.6 – *Lecteur de contenus pédagogiques*

Comme le montre la figure 4.6, la ressource sélectionnée est affichée dans la zone principale, tandis que le plan du cours facilite la navigation. Une fois le contenu consulté, le participant peut marquer le chapitre comme terminé, ce qui entraîne la mise à jour de sa progression.

4.4.3 Évaluations et résultats

Les quiz permettent de vérifier l'acquisition des connaissances à la fin d'un module ou d'une formation. Les questions sont chargées depuis le backend, qui assure également la correction et le calcul du résultat.

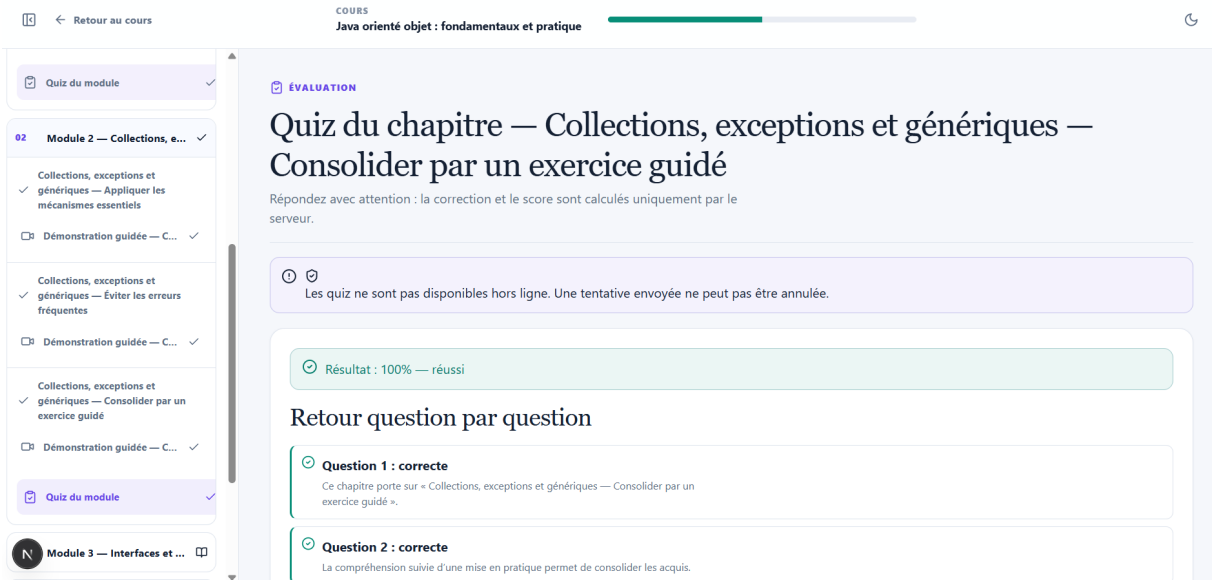


Figure 4.7 – *Résultat d'un quiz réussi*

La figure 4.7 présente clairement le score obtenu et l'état de réussite. Le traitement du résultat côté serveur contribue à préserver l'intégrité de l'évaluation.

4.4.4 Génération du certificat

Lorsqu'un participant termine les contenus obligatoires et réussit les évaluations requises, Khotwa génère un certificat de réussite.



Figure 4.8 – *Certificat de réussite généré par Khotwa*

Le certificat présenté par la figure 4.8 contient le nom du participant, le titre de la formation, la date de délivrance et un code de vérification. Il peut ensuite être consulté ou téléchargé depuis l'espace du participant.

4.4.5 Centre de notifications

Le centre de notifications centralise les événements importants du parcours, comme la disponibilité d'un quiz, l'ajout d'un contenu, la planification d'une classe virtuelle ou la génération d'un certificat.

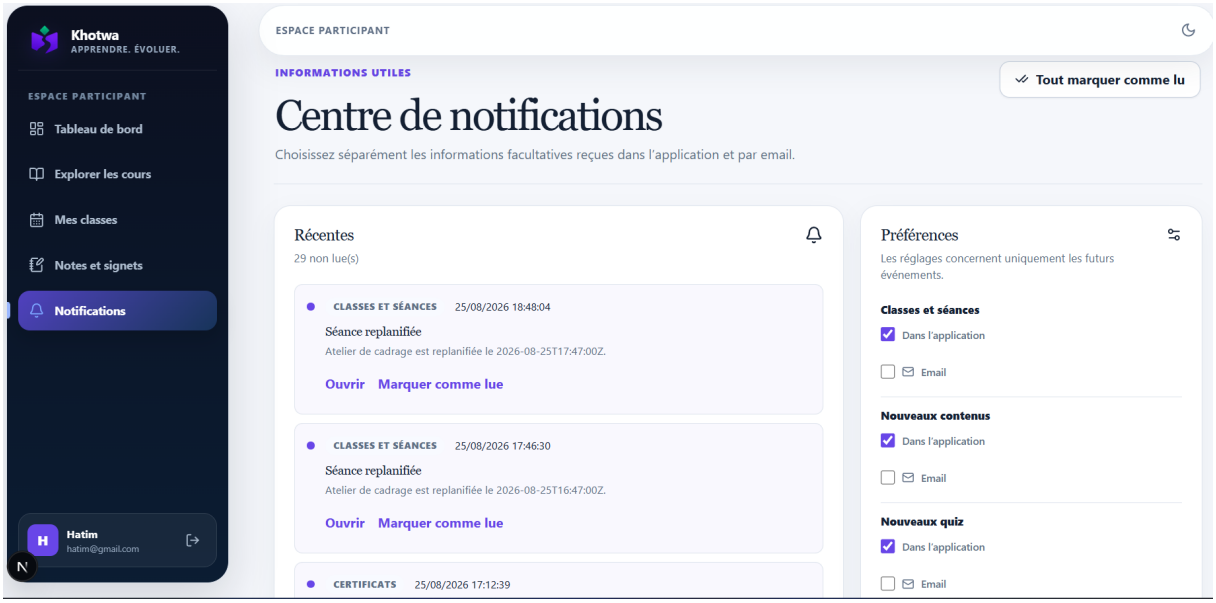


Figure 4.9 – *Centre de notifications du participant*

À partir de la figure 4.9, le participant peut consulter les événements récents, ouvrir la ressource concernée et marquer les notifications comme lues. Des préférences permettent également de choisir les catégories reçues dans l'application ou par courrier électronique.

4.5 Réalisation de l'espace formateur

L'espace formateur regroupe les outils nécessaires à la création des formations, à l'organisation des ressources pédagogiques et au suivi des participants.

4.5.1 Tableau de bord du formateur

Le tableau de bord présente une synthèse des activités du formateur, notamment les formations publiées, les participants inscrits, les séances à venir et les avis reçus.

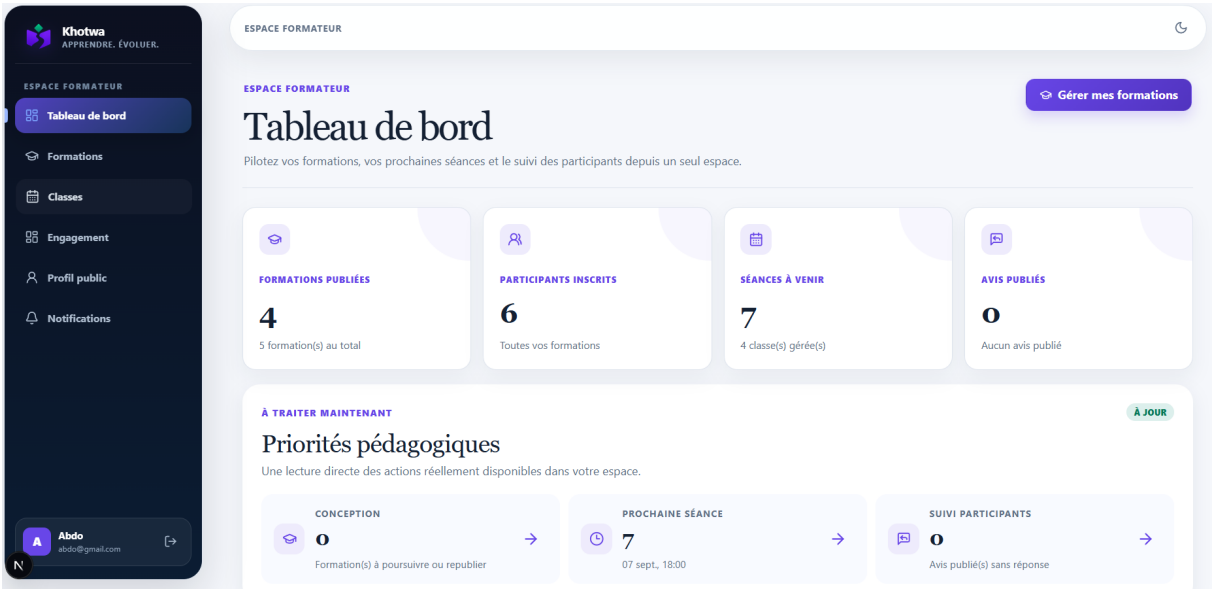


Figure 4.10 – *Tableau de bord du formateur*

La figure 4.10 met en évidence les priorités pédagogiques et fournit un accès direct à la gestion des formations, aux classes virtuelles, aux indicateurs d'engagement, au profil public et aux notifications.

4.6 Réalisation de l'espace administrateur

L'espace administrateur permet de superviser le fonctionnement général de Khotwa. Il regroupe les outils de gestion des utilisateurs, de validation des formateurs, de modération et d'audit.

4.6.1 Tableau de bord de l'administrateur

Le tableau de bord administratif fournit une vue globale de l'état de la plateforme. Il met en évidence les indicateurs principaux ainsi que les décisions nécessitant une intervention.

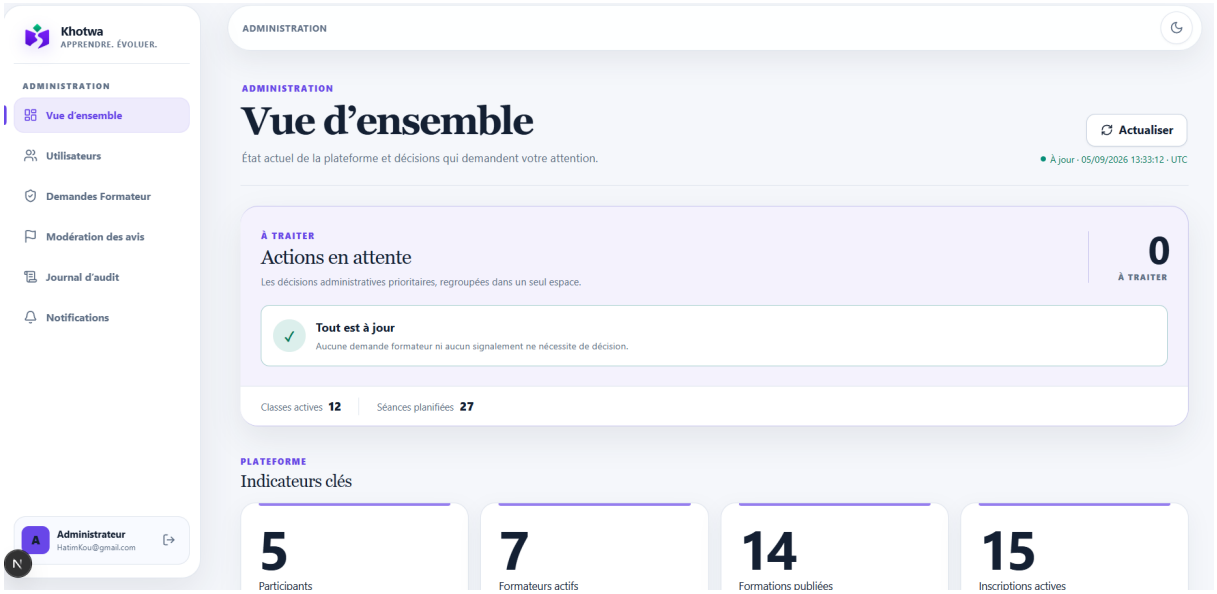


Figure 4.11 – *Tableau de bord de l'administrateur*

La figure 4.11 présente notamment le nombre de participants, de formateurs actifs, de formations publiées et d'inscriptions. La zone des actions en attente regroupe les opérations administratives prioritaires.

4.6.2 Validation des candidatures des formateurs

L'accès au rôle de formateur nécessite une validation administrative. Lorsqu'un utilisateur soumet une candidature, celle-ci apparaît dans l'espace réservé aux demandes des formateurs.

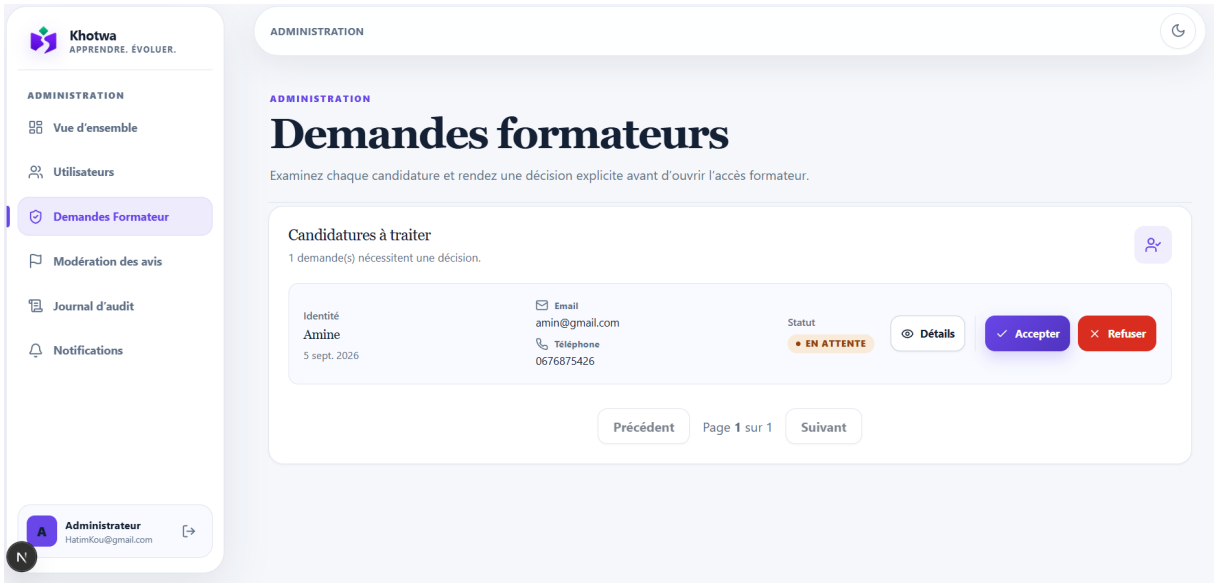


Figure 4.12 – *Traitement d'une candidature de formateur*

Comme le montre la figure 4.12, l'administrateur peut consulter les informations du candidat, accepter sa demande ou la refuser. Cette validation évite l'attribution automatique des privilèges permettant de créer et de publier des formations.

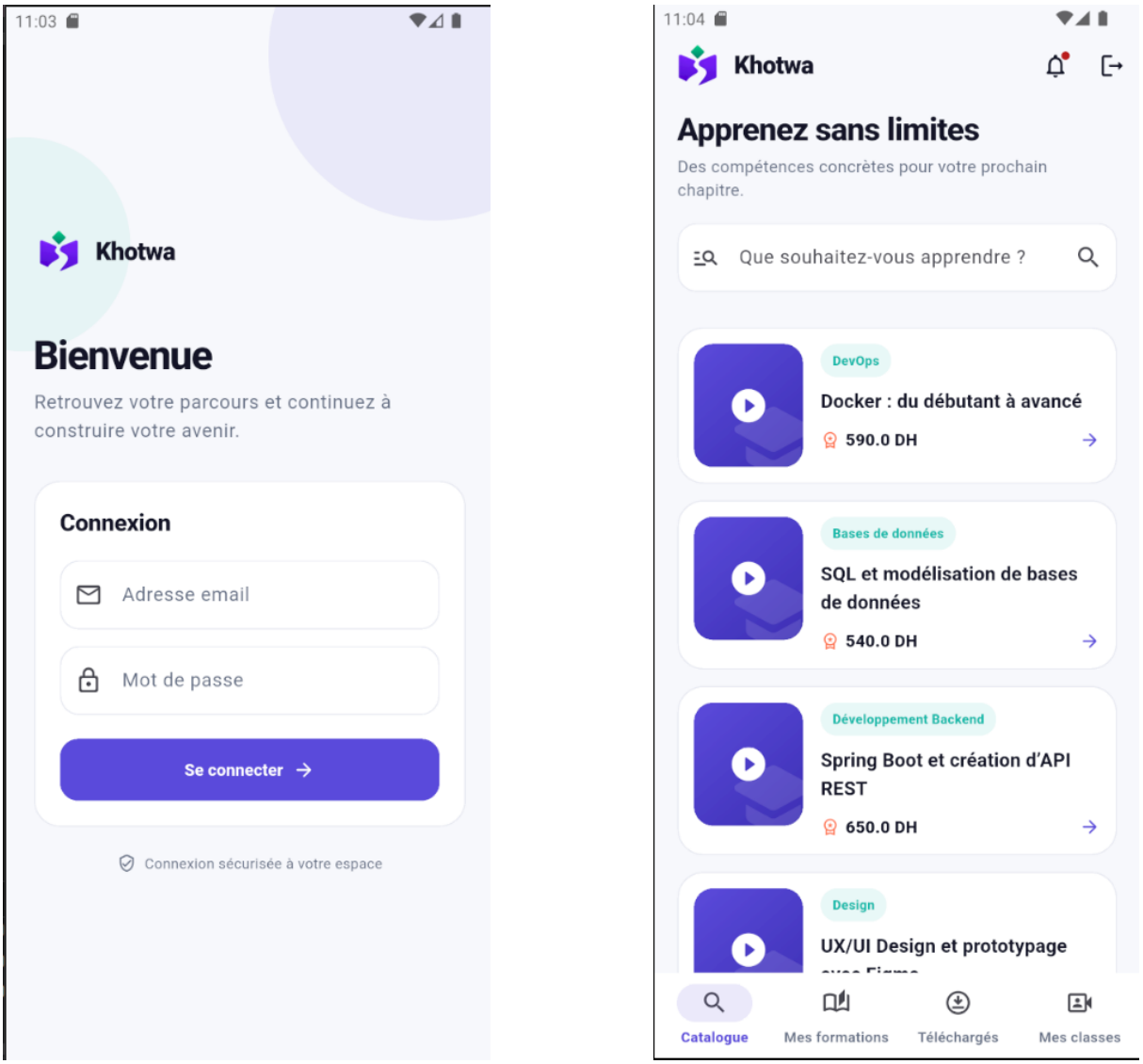
4.7 Réalisation de l'application mobile

En complément de l'application Web, Khotwa propose une application mobile développée avec Flutter et Dart. Principalement destinée aux participants, elle communique avec le même backend Spring Boot et reprend les fonctionnalités essentielles du parcours d'apprentissage.

4.7.1 Authentification et consultation du catalogue

L'application mobile utilise le même système d'authentification JWT que l'application Web. Après une connexion réussie, le jeton est enregistré au moyen d'un mécanisme de stockage sécurisé et ajouté automatiquement aux requêtes protégées.

Le catalogue récupère les formations publiées depuis le backend. Il affiche leur titre, leur catégorie, leur prix et leur état, tout en permettant une recherche par mot-clé.



(a) Interface de connexion (b) Catalogue des formations

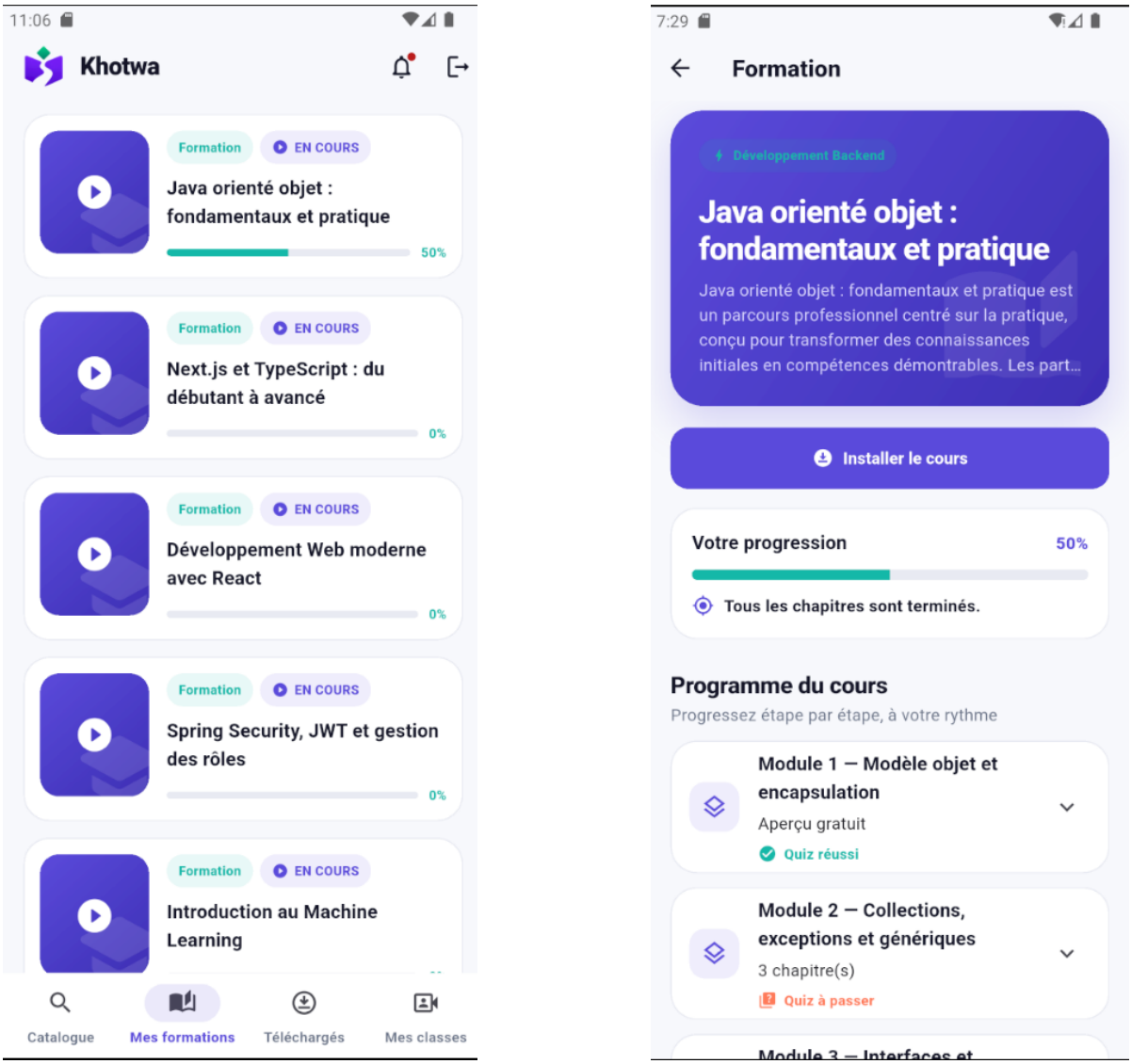
Figure 4.13 – *Authentification et consultation du catalogue sur l'application mobile*

La figure 4.13 illustre l'entrée dans l'espace mobile et l'accès aux formations disponibles.

4.7.2 Formations et parcours pédagogique

La rubrique « Mes formations » regroupe les formations acquises par le participant. Elle présente leur état et le pourcentage de progression enregistré par le backend.

La page détaillée affiche les informations générales du parcours ainsi que son organisation en modules et en chapitres. Les différents états permettent de repérer les éléments terminés, le chapitre actuel et les contenus encore verrouillés.



(a) Formations acquises (b) Détails d'une formation

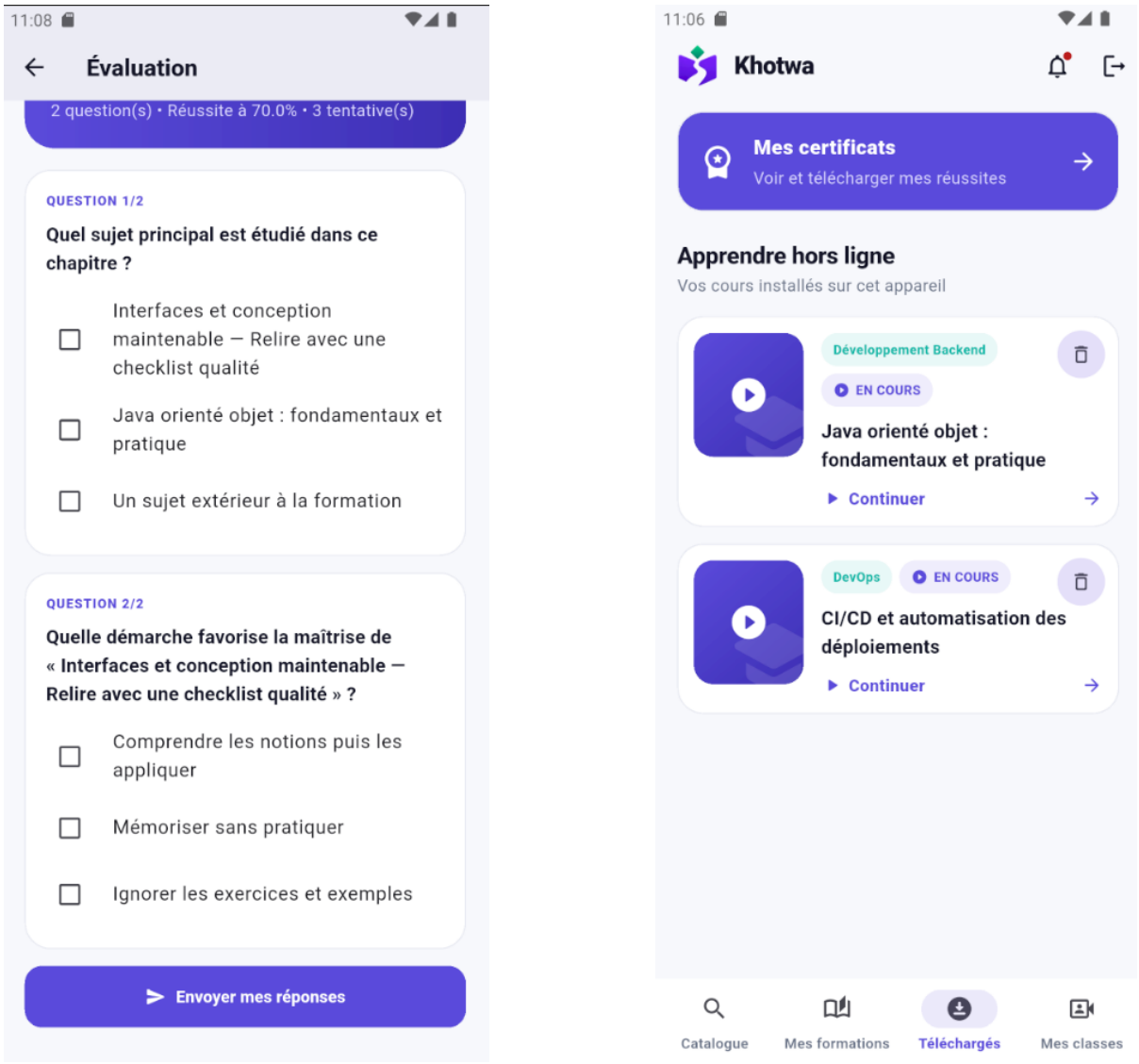
Figure 4.14 – *Gestion des formations et du parcours pédagogique sur mobile*

Comme le montre la figure 4.14, l'interface facilite la reprise de l'apprentissage et permet au participant de suivre son avancement.

4.7.3 Quiz et fonctionnement hors connexion

Les quiz sont directement intégrés à l'application. Le participant peut lire les questions, sélectionner ses réponses et soumettre sa tentative au backend.

L'application permet également d'installer localement certaines formations acquises. Les vidéos, documents PDF et images autorisés sont enregistrés dans le répertoire privé de l'application afin de rester accessibles sans connexion.



(a) Passage d'un quiz (b) Formations téléchargées

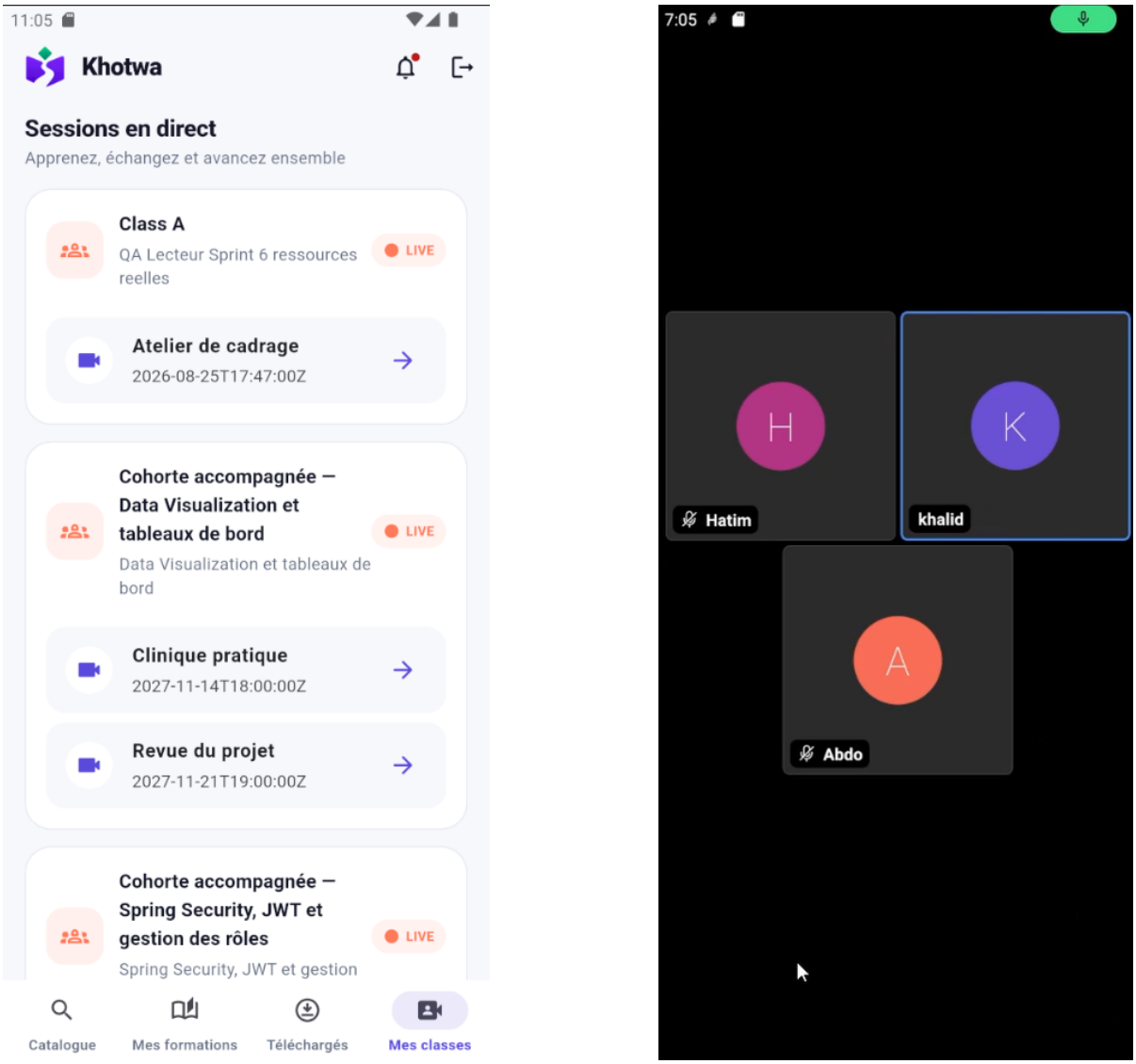
Figure 4.15 – *Quiz intégré et fonctionnement hors connexion*

La figure 4.15 illustre ces deux usages. Les actions nécessitant une validation du backend, comme la soumission d'un quiz ou la synchronisation de la progression, restent disponibles uniquement lorsque l'application dispose d'une connexion.

4.7.4 Classes virtuelles et intégration de Jitsi Meet

La rubrique « Mes classes » présente les classes associées aux formations du participant ainsi que les séances programmées. Avant de fournir l'accès, le backend vérifie l'inscription, l'appartenance à la classe et la disponibilité de la séance.

L'intégration native du SDK Jitsi Meet permet de rejoindre une visioconférence en utilisant la caméra et le microphone du téléphone, sans ouvrir une application externe.



(a) Classes du participant (b) Classe virtuelle avec Jitsi Meet

Figure 4.16 – *Consultation et participation aux classes virtuelles*

Lorsque la réunion présentée par la figure 4.16 est terminée, l'application ferme automatiquement l'interface de visioconférence et ramène le participant vers l'écran Khotwa précédent.

4.8 Conclusion

Ce chapitre a présenté les principales interfaces réalisées pour les applications Web et mobile de Khotwa. L'application Web propose un espace public ainsi que des espaces adaptés aux participants, aux formateurs et aux administrateurs. Elle prend en charge la consultation des formations, l'apprentissage, les évaluations, les certificats, les notifications et la supervision de la plateforme.

L'application mobile complète cette réalisation en permettant au participant de s'authentifier, de consulter le catalogue, de suivre ses formations, de passer les quiz, de télécharger des contenus et de rejoindre les classes virtuelles depuis un smartphone.

Les captures présentées confirment la réalisation des principaux besoins définis pendant la phase d'analyse. Le chapitre suivant sera consacré aux tests, à la qualité et à la validation de la solution.

Chapitre 5

Tests, qualité et validation

5.1 Introduction

La réalisation des fonctionnalités ne suffit pas à garantir la fiabilité d'une application. Une plateforme E-learning comme Khotwa traite des données personnelles, des contenus pédagogiques, des résultats d'évaluation et des autorisations associées à plusieurs profils. Elle doit donc faire l'objet de vérifications permettant de contrôler son comportement, sa sécurité et la qualité de son code.

Ce chapitre présente la stratégie de test adoptée pour Khotwa. Il décrit les tests réalisés sur le backend Spring Boot, les contrôles de l'application Web, la portée de la validation mobile, les scénarios fonctionnels vérifiés ainsi que les résultats de l'analyse de qualité obtenus avec SonarQube.

5.2 Stratégie de test

La stratégie adoptée repose sur plusieurs niveaux de validation complémentaires. Chaque niveau permet de détecter une catégorie particulière d'anomalies.

Tableau 5.1 – *Niveaux de test appliqués à Khotwa*

Niveau	**Moyen utilisé**	**Objectif**
Tests unitaires	JUnit et tests automatisés	Vérifier individuellement les services, les fonctions et les composants.
Tests d'intégration	Spring Boot Test	Contrôler les échanges entre les contrôleurs, les services, la sécurité et la persistance.
Tests des interfaces	Tests automatisés frontend	Vérifier le rendu des composants et les principales interactions de l'utilisateur.
Tests fonctionnels	Scénarios de validation	Vérifier les parcours des participants, des formateurs et des administrateurs.
Analyse statique	SonarQube et JaCoCo	Mesurer la couverture, les duplications et les problèmes liés à la qualité du code.

Cette combinaison permet de vérifier les règles métier du backend, le comportement des interfaces et le fonctionnement général des principaux parcours utilisateurs.

5.3 Tests du backend Spring Boot

Le backend représente le cœur fonctionnel de Khotwa. Il contient les règles métier, les mécanismes de sécurité, les contrôles d'autorisation et les opérations de persistance.

Le projet contient 39 fichiers Java de test regroupant 182 méthodes annotées avec `@Test`. Ces tests couvrent plusieurs domaines fonctionnels et techniques de la plateforme.

5.3.1 Tests unitaires

Les tests unitaires, exécutés avec **JUnit**, vérifient le comportement des services de manière isolée [21]. Ils permettent notamment de contrôler :

— l'authentification des utilisateurs ;

— la réinitialisation du mot de passe ;

— la gestion des formations ;

— les règles relatives aux quiz ;

— les conditions d'obtention des certificats ;

— la gestion des notifications ;

— la validation des fichiers téléversés ;

— la gestion des classes virtuelles ;

— la modération des avis ;

— la validation administrative des formateurs.

Chaque test prépare les données nécessaires, exécute l'opération concernée puis compare le résultat obtenu avec le comportement attendu.

Cette approche facilite la détection des anomalies dans les règles métier sans dépendre directement de l'interface Web.

5.3.2 Tests d'intégration

Les tests d'intégration vérifient le fonctionnement de plusieurs composants exécutés ensemble. Ils contrôlent principalement les échanges entre les contrôleurs REST, les services métier, les composants de sécurité et la couche de persistance.

Les principaux scénarios vérifiés concernent :

— la création d'un compte ;

— la connexion d'un utilisateur ;

— la consultation des formations ;

— la gestion des formations ;

— la visibilité des modules pour les participants ;

— la gestion administrative des utilisateurs ;

— le traitement des candidatures des formateurs ;

— la modération des avis signalés ;

— la protection des fonctionnalités administratives.

Ces tests permettent de vérifier que le backend respecte le comportement attendu par les applications clientes.

5.3.3 Tests des opérations concurrentes

Certaines actions peuvent être déclenchées simultanément par plusieurs utilisateurs. Des tests spécifiques vérifient que ces situations ne provoquent pas d'incohérences dans les données.

Ces contrôles concernent notamment :

— la validation simultanée d'une candidature de formateur ;

— la modération concurrente d'un avis ;

— le signalement d'un avis ;

— la vérification de l'éligibilité à un certificat.

L'objectif est d'éviter qu'une même opération soit confirmée plusieurs fois ou que deux décisions incompatibles soient enregistrées simultanément.

5.3.4 Tests des migrations de données

Flyway est utilisé pour gérer les évolutions du schéma de la base de données. Plusieurs tests permettent de vérifier que les migrations conservent la cohérence des données et restent compatibles avec les règles métier.

Les vérifications portent notamment sur :

— les données administratives ;

— les notifications ;

— la modération des avis ;

— les informations liées au stockage ;

— les évolutions du modèle de données.

Ces tests limitent les risques de régression lors de la modification de la structure de la base de données.

5.4 Tests de l'application Web

L'application Web fait l'objet de tests automatisés permettant de vérifier le comportement des composants, la navigation entre les pages et les principales interactions de l'utilisateur.

Le projet contient 27 fichiers de test Web couvrant les interfaces publiques, les espaces authentifiés et les composants pédagogiques.

5.4.1 Tests des composants

Les tests des composants contrôlent le rendu des éléments graphiques et leurs réactions aux actions de l'utilisateur.

Ils couvrent notamment :

— les menus de navigation ;

— les formulaires d'authentification ;

— les cartes du catalogue ;

— les boutons et les messages d'état ;

— le lecteur vidéo ;

— le lecteur de documents PDF ;

— les interfaces des quiz ;

— le centre de notifications ;

— les tableaux de bord.

Ces tests permettent de détecter les régressions pouvant affecter plusieurs pages de l'application.

5.4.2 Tests des parcours fonctionnels Web

Les tests automatisés couvrent plusieurs parcours associés aux acteurs de Khotwa. Pour le participant, les vérifications concernent notamment :

— la consultation du catalogue ;

— l'affichage des détails d'une formation ;

— l'authentification ;

— la lecture des ressources pédagogiques ;

— la progression dans les modules ;

— le passage des quiz ;

— les notifications ;

— les classes virtuelles.

Pour le formateur, les contrôles portent sur le tableau de bord, la gestion des formations, le suivi des participants et l'organisation des classes virtuelles.

Pour l'administrateur, les tests vérifient le tableau de bord, la gestion des utilisateurs, le traitement des candidatures des formateurs et la modération des avis.

5.4.3 Commandes de vérification

Les commandes suivantes permettent d'exécuter les principales vérifications de l'application Web :

```
npm run test
npm run lint
npm run typecheck
npm run build
```

Listing 5.1 *– Commandes de vérification de l'application Web*

La première commande exécute les tests automatisés. La commande `lint` vérifie les règles de qualité du code. La commande `typecheck` contrôle la cohérence des types TypeScript. Enfin, la commande `build` vérifie que l'application peut être générée pour un environnement de production.

5.5 Validation de l'application mobile

Les fonctionnalités de l'application Flutter présentées au chapitre précédent sont réalisées. Un premier socle de tests est présent dans le projet ; toutefois, le rapport ne documente pas encore une validation exhaustive de tous les parcours mobiles sur appareil ou émulateur. L'achèvement du développement doit donc être distingué de l'étendue des vérifications effectuées.

Les vérifications complémentaires devraient notamment porter sur :

— l'authentification ;

— la conservation de la session ;

— la consultation des formations ;

— la lecture des ressources pédagogiques ;

— le suivi de la progression ;

— le téléchargement des ressources ;

— la consultation hors ligne ;

— la synchronisation après le retour de la connexion.

Les résultats de ces vérifications devront être consignés pour établir précisément le niveau de validation de l'application mobile.

5.6 Validation fonctionnelle des principaux parcours

En complément des tests automatisés, les principales fonctionnalités de Khotwa ont été vérifiées à travers des scénarios représentatifs.

Tableau 5.2 – *Validation des principaux parcours fonctionnels*

Acteur	**Scénario**	**Résultat attendu**
Visiteur	Consulter le catalogue et rechercher une formation.	Les formations correspondant aux critères sont affichées.
Participant	Se connecter avec des informations valides.	L'utilisateur accède à son tableau de bord.
Participant	Saisir des informations d'authentification incorrectes.	L'accès est refusé et un message adapté est affiché.

Acteur	**Scénario**	**Résultat attendu**
Participant	S'inscrire à une formation puis consulter son contenu.	La formation est ajoutée à son parcours et les ressources autorisées deviennent accessibles.
Participant	Terminer un chapitre pédagogique.	La progression est enregistrée et le parcours est mis à jour.
Participant	Soumettre les réponses d'un quiz.	Le serveur calcule le résultat et retourne un bilan question par question.
Participant	Terminer les modules et réussir les évaluations obligatoires.	Le certificat devient disponible lorsque les conditions sont satisfaites.
Formateur	Créer et organiser une formation.	Les modules, les chapitres et les ressources sont enregistrés dans le bon ordre.
Formateur	Planifier une classe virtuelle.	La séance est enregistrée et devient accessible aux participants concernés.
Administrateur	Accepter une candidature de formateur.	Le compte obtient les autorisations associées au rôle formateur.
Administrateur	Refuser une candidature de formateur.	La décision est enregistrée sans attribuer les privilèges du formateur.
Administrateur	Modérer un avis signalé.	L'action est enregistrée et l'état de l'avis est mis à jour.

5.7 Validation de la sécurité

Les tests de sécurité vérifient que les ressources protégées ne sont pas accessibles sans authentification et qu'un utilisateur ne peut pas exécuter une opération réservée à un autre rôle.

Les principaux contrôles portent sur :

— le refus des requêtes sans jeton valide ;

— la vérification de la validité des jetons JWT ;

— la séparation des droits entre les différents rôles ;

— la protection des fonctionnalités administratives ;

— la validation des fichiers envoyés vers le stockage ;

— le contrôle de l'accès aux ressources pédagogiques ;

— la correction des quiz par le serveur ;

— la prévention des décisions administratives concurrentes.

Ces validations réduisent les risques d'accès non autorisé aux comptes, aux contenus pédagogiques et aux outils d'administration.

5.8 Analyse de la qualité avec SonarQube

Une analyse statique a été réalisée avec **SonarQube** afin d'évaluer la sécurité, la fiabilité, la maintenabilité, la couverture des tests et la duplication du code [23].

La figure 5.1 présente le tableau de bord obtenu pour Khotwa.

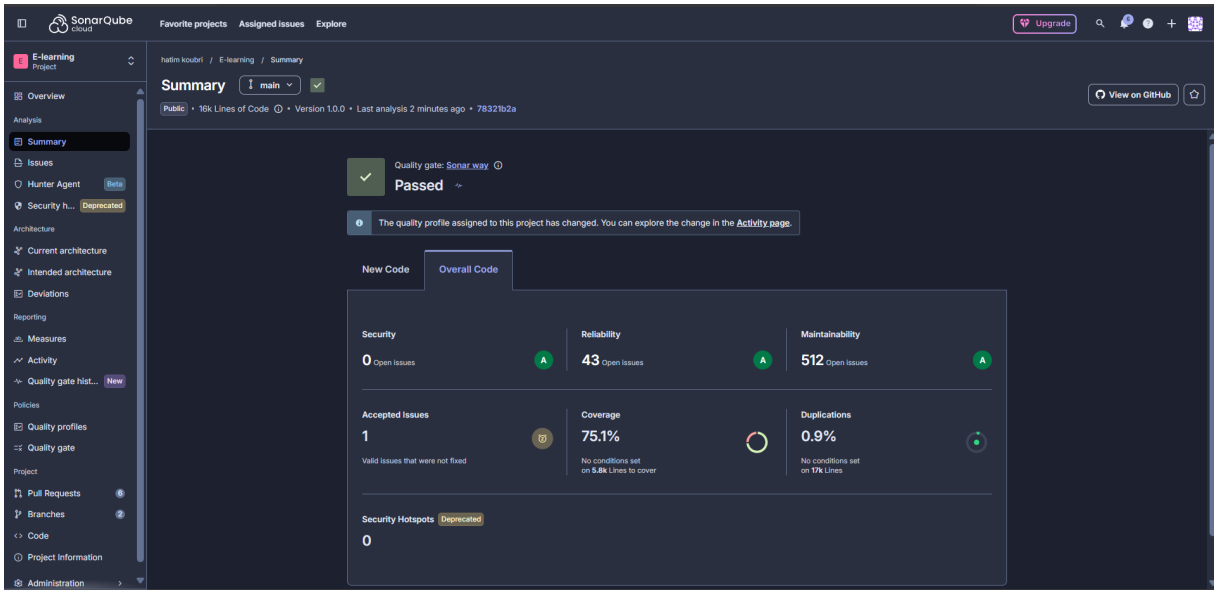


Figure 5.1 – *Résultat de l'analyse de qualité avec SonarQube*

Comme le montre la figure 5.1, la porte de qualité, appelée *Quality Gate*, est validée [24]. L'analyse porte sur environ 16 000 lignes de code et la couverture est mesurée à l'aide de **JaCoCo** [22].

Les principaux résultats sont présentés dans le tableau 5.3.

Tableau 5.3 – *Synthèse des indicateurs SonarQube*

Indicateur	**Résultat**	**Interprétation**
Quality Gate	Réussi	Les conditions définies par la porte de qualité sont respectées.
Problèmes de sécurité ouverts	0	Aucun problème de sécurité ouvert n'est indiqué.
Points sensibles de sécurité	0	Aucun point sensible de sécurité n'est affiché.
Note de sécurité	A	Le code obtient la meilleure note affichée pour la sécurité.
Problèmes de fiabilité	43	Des remarques restent ouvertes malgré une note globale A.
Problèmes de maintenabilité	512	Des améliorations restent possibles malgré une note globale A.
Couverture	75,1 %	Une part importante du code analysé est couverte par les tests.
Duplication	0,9 %	Le taux de duplication du code reste faible.
Problèmes acceptés	1	Un problème a été explicitement accepté dans l'analyse.

La validation de la porte de qualité représente un résultat positif. La couverture de 75,1 % indique qu'une partie importante du code mesuré est exécutée par les tests. Le taux de duplication de 0,9 % montre que la répétition du code reste limitée.

Cependant, la réussite du Quality Gate ne signifie pas que le projet est exempt de toute amélioration. L'analyse affiche encore 43 problèmes associés à la fiabilité et 512 problèmes associés à la maintenabilité. Ces éléments devront être examinés selon leur impact, leur priorité et les modules concernés.

5.9 Limites de la validation actuelle

Malgré les résultats obtenus, certaines limites doivent être prises en considération :

— une couverture de 75,1 % ne garantit pas que tous les scénarios métier sont testés ;

— les résultats SonarQube correspondent à un état précis du code ;

— les problèmes de fiabilité et de maintenabilité encore ouverts doivent être examinés ;

— la validation mobile documentée reste partielle, malgré l'achèvement des fonctionnalités présentées ;

— les performances devront être testées avec un volume plus important d'utilisateurs et de ressources ;

— les services externes devront être vérifiés dans un environnement proche de la production.

Ces limites ne remettent pas en cause les fonctionnalités déjà validées, mais elles définissent les travaux nécessaires avant un déploiement à plus grande échelle.

5.10 Améliorations prévues

Plusieurs actions pourront être mises en place pour renforcer la qualité de Khotwa :

— augmenter la couverture des modules les plus critiques ;

— examiner les problèmes de fiabilité et de maintenabilité ;

— compléter les tests de l'application Flutter ;

— ajouter des tests couvrant les parcours complets ;

— mettre en place des tests de charge et de performance ;

— automatiser les contrôles dans une chaîne d'intégration continue ;

— exécuter régulièrement l'analyse SonarQube ;

— renforcer les tests des ressources protégées et des opérations concurrentes.

5.11 Conclusion

Dans ce chapitre, nous avons présenté la stratégie de test et de validation appliquée à Khotwa. Le backend dispose de tests unitaires, de tests d'intégration et de vérifications portant sur la sécurité, les migrations et les opérations concurrentes. L'application Web dispose également de tests automatisés couvrant ses principaux composants et parcours fonctionnels.

Les principales fonctionnalités ont été vérifiées à travers des scénarios représentatifs pour les participants, les formateurs et les administrateurs. L'analyse SonarQube confirme la réussite de la porte de qualité, avec une couverture de 75,1 % et un taux de duplication limité à 0,9 %.

L'analyse met néanmoins en évidence plusieurs possibilités d'amélioration relatives à la fiabilité et à la maintenabilité. Les travaux futurs devront également compléter la validation mobile, renforcer les tests des parcours complets et poursuivre le traitement des remarques signalées par l'analyse statique.

Conclusion générale et perspectives

Le présent projet avait pour objectif de concevoir et de développer une plateforme E-learning sécurisée et multiplateforme, capable de répondre aux besoins des participants, des formateurs et des administrateurs. Réalisée dans le cadre du projet de fin d'année au sein du service informatique de **MIC Consulting Group**, la solution développée, nommée **Khotwa**, centralise les principales activités liées à la création, à la diffusion et au suivi des formations en ligne.

La réalisation du projet a commencé par une analyse des besoins fonctionnels et non fonctionnels. Cette étape a permis d'identifier les responsabilités des différents acteurs et de définir les principaux parcours de la plateforme. La phase de conception a ensuite permis de représenter les interactions avec le système à travers les diagrammes de cas d'utilisation et de préparer la structure des données et des composants applicatifs.

Sur le plan technique, Khotwa repose sur une architecture organisée en plusieurs couches. Le backend, développé avec **Java et Spring Boot**, centralise les règles métier et expose une API REST sécurisée avec Spring Security et JWT. L'interface Web est réalisée avec **Next.js, React et TypeScript**, tandis que l'application mobile repose sur **Flutter**. Les données structurées sont conservées dans **MySQL**, les migrations sont gérées avec **Flyway** et les ressources pédagogiques sont stockées dans **MinIO**. L'environnement utilise également Docker Compose, Mailpit, OpenAPI, Swagger et Jitsi.

La plateforme offre aux participants un catalogue de formations, un lecteur de contenus pédagogiques, un système de progression, des quiz, des certificats, des notifications et des classes virtuelles. Les formateurs disposent d'outils leur permettant d'organiser leurs formations, leurs modules, leurs chapitres et leurs ressources, tout en assurant le suivi des participants. Les administrateurs peuvent superviser la plateforme, gérer les utilisateurs, traiter les candidatures des formateurs, modérer les avis et consulter les événements d'audit.

Une attention particulière a été accordée à la sécurité et à la qualité de la solution. La gestion des rôles limite l'accès aux fonctionnalités selon les responsabilités de chaque utilisateur. Les ressources pédagogiques sont protégées et les résultats des quiz sont calculés côté serveur. Des tests unitaires, des tests d'intégration et des tests automatisés de l'application Web ont été mis en place afin de vérifier les principales règles métier et les parcours fonctionnels.

L'analyse réalisée avec SonarQube a permis d'obtenir une porte de qualité validée, une couverture de tests de **75,1 %** et un taux de duplication limité à **0,9 %**. Ces

résultats montrent que le projet repose sur une base fonctionnelle et mesurable. Certaines remarques de fiabilité et de maintenabilité restent néanmoins à examiner afin d'améliorer progressivement la qualité du code.

La réalisation de Khotwa a permis de mettre en pratique plusieurs compétences liées au développement Web, aux applications mobiles, à la conception d'API REST, à la sécurité, à la gestion des données, au stockage de fichiers, aux tests logiciels et à la conteneurisation. Elle a également permis d'approfondir l'organisation d'un projet complexe selon une démarche Agile inspirée de Scrum.

Perspectives

Bien que les principaux objectifs fonctionnels aient été atteints, plusieurs améliorations peuvent être envisagées pour faire évoluer Khotwa :

- approfondir les tests et la validation de l'application mobile Flutter ;
- étendre la consultation hors ligne et améliorer la synchronisation des ressources pédagogiques ;
- intégrer une véritable passerelle de paiement à la place du processus simulé ;
- développer des mécanismes de recommandation personnalisée et d'orientation pédagogique ;
- renforcer les outils d'analyse de la progression et de l'engagement des participants ;
- enrichir les tableaux de bord avec de nouveaux indicateurs pédagogiques et administratifs ;
- compléter les tests de bout en bout et les tests de performance ;
- traiter les remarques de fiabilité et de maintenabilité identifiées par SonarQube ;
- mettre en place une chaîne complète d'intégration et de déploiement continus ;
- déployer la plateforme dans un environnement de production sécurisé et supervisé ;
- renforcer la traçabilité des certificats et faciliter leur vérification ;
- améliorer l'accessibilité des interfaces Web et mobile.

En conclusion, Khotwa constitue une base solide pour une plateforme E-learning moderne et évolutive. Son architecture modulaire, ses mécanismes de sécurité et la séparation claire des responsabilités permettent d'envisager son enrichissement progressif. Les perspectives identifiées pourront transformer cette première version en une solution plus complète, capable de répondre à des contextes de formation variés et à un nombre croissant d'utilisateurs.

Références

[1] Object Management Group (OMG), *Unified Modeling Language (UML), version 2.5.1*. Spécification officielle du langage de modélisation UML. Disponible sur : https://www.omg.org/spec/UML/. Consulté le 20 septembre 2026.

[2] Spring, *Spring Boot Reference Documentation*. Documentation officielle de Spring Boot. Disponible sur : https://docs.spring.io/spring-boot/reference/. Consulté le 16 août 2026.

[3] Spring, *Spring Security Reference Documentation*. Documentation officielle relative à l'authentification, à l'autorisation et à la protection des applications Spring. Disponible sur : https://docs.spring.io/spring-security/reference/. Consulté le 16 août 2026.

[4] Spring, *Spring Data JPA Reference Documentation*. Documentation officielle de la couche de persistance Spring Data JPA. Disponible sur : https://docs.spring.io/spring-data/jpa/reference/. Consulté le 16 août 2026.

[5] Hibernate, *Hibernate ORM Documentation*. Documentation officielle de l'outil de correspondance objet-relationnel Hibernate. Disponible sur : https://hibernate.org/orm/documentation/. Consulté le 16 août 2026.

[6] M. Jones, J. Bradley et N. Sakimura, *JSON Web Token (JWT)*, RFC 7519, Internet Engineering Task Force, mai 2015. Disponible sur : https://datatracker.ietf.org/doc/html/rfc7519. Consulté le 16 août 2026.

[7] Vercel, *Next.js Documentation*. Documentation officielle du framework Next.js. Disponible sur : https://nextjs.org/docs. Consulté le 16 août 2026.

[8] Meta Open Source, *React Documentation*. Documentation officielle de la bibliothèque React. Disponible sur : https://react.dev/. Consulté le 16 août 2026.

[9] Microsoft, *TypeScript Documentation*. Documentation officielle du langage TypeScript. Disponible sur : https://www.typescriptlang.org/docs/. Consulté le 16 août 2026.

[10] Google, *Flutter Documentation*. Documentation officielle du framework Flutter. Disponible sur : https://docs.flutter.dev/. Consulté le 16 août 2026.

[11] Google, *Dart Documentation*. Documentation officielle du langage Dart. Disponible sur : https://dart.dev/guides. Consulté le 16 août 2026.

[12] Oracle, *MySQL 8.4 Reference Manual*. Manuel officiel du système de gestion de base de données MySQL. Disponible sur : https://dev.mysql.com/doc/refman/8.4/en/. Consulté le 16 août 2026.

[13] Redgate, *Flyway Documentation*. Documentation officielle de l'outil de gestion des migrations de bases de données. Disponible sur : https://documentation.red-gate.com/flyway. Consulté le 16 août 2026.

[14] MinIO, *MinIO Documentation*. Documentation officielle de la solution de stockage d'objets compatible avec l'API S3. Disponible sur : https://min.io/docs/minio/container/index.html. Consulté le 16 août 2026.

[15] Jitsi, *Jitsi Meet Handbook*. Documentation officielle de la solution de visioconférence Jitsi Meet. Disponible sur : https://jitsi.github.io/handbook/docs/intro/. Consulté le 16 août 2026.

[16] Mailpit, *Mailpit Documentation*. Documentation officielle de l'outil de test des courriers électroniques. Disponible sur : https://mailpit.axllent.org/docs/. Consulté le 16 août 2026.

[17] OpenAPI Initiative, *OpenAPI Specification*. Spécification officielle pour la description des API HTTP. Disponible sur : https://spec.openapis.org/oas/latest.html. Consulté le 16 août 2026.

[18] SmartBear Software, *Swagger Documentation*. Documentation des outils de description et de vérification des API. Disponible sur : https://swagger.io/docs/. Consulté le 16 août 2026.

[19] Docker, *Docker Documentation*. Documentation officielle de la plateforme de conteneurisation Docker. Disponible sur : https://docs.docker.com/. Consulté le 16 août 2026.

[20] Docker, *Docker Compose Documentation*. Documentation officielle pour la définition et l'exécution d'applications multiconteneurs. Disponible sur : https://docs.docker.com/compose/. Consulté le 16 août 2026.

[21] JUnit Team, *JUnit 5 User Guide*. Guide officiel du framework de test Java JUnit. Disponible sur : https://docs.junit.org/5/current/user-guide/. Consulté le 16 août 2026.

[22] JaCoCo, *JaCoCo Java Code Coverage Library*. Documentation officielle de l'outil de mesure de la couverture du code Java. Disponible sur : https://www.jacoco.org/jacoco/trunk/doc/. Consulté le 16 août 2026.

[23] SonarSource, *SonarQube Documentation*. Documentation officielle relative à l'analyse de la qualité et de la sécurité du code. Disponible sur : https://docs.sonarsource.com/sonarqube-server/. Consulté le 16 août 2026.

[24] SonarSource, *Understanding Quality Gates*. Documentation officielle relative aux portes de qualité SonarQube. Disponible sur : https://docs.sonarsource.com/sonarqube-server/quality-standards-administration/managing-quality-gates/introduction-to-quality-gates. Consulté le 16 août 2026.

Annexe A

Organisation technique du projet

Cette annexe présente l'organisation générale des différents composants de Khotwa. La plateforme est constituée d'un backend Spring Boot, d'une application Web Next.js, d'une application mobile Flutter et de plusieurs services techniques.

Tableau A.1 – *Organisation des composants de Khotwa*

Composant	**Technologies**	**Responsabilité**
Backend	Java et Spring Boot	Gestion des règles métier, de la sécurité et des services REST.
Application Web	Next.js, React et TypeScript	Interfaces Web des participants, des formateurs et des administrateurs.
Application mobile	Flutter et Dart	Accès mobile aux principales fonctionnalités de la plateforme.
Base de données	MySQL	Conservation des données métier de Khotwa.
Migrations	Flyway	Versionnement et évolution du schéma de la base de données.
Stockage	MinIO	Stockage des vidéos, des documents PDF, des images et des certificats.
Courrier électronique	Mailpit	Vérification des messages électroniques dans l'environnement de développement.
Classes virtuelles	Jitsi	Organisation des séances de visioconférence.
Conteneurisation	Docker Compose	Exécution et coordination des services techniques.

L'organisation logique du projet peut être représentée de la manière suivante :

```
Khotwa/
|-- backend/
|   |-- src/main/java/
```

```
|   |-- src/main/resources/
|   |-- src/test/
|   `-- pom.xml
|
|-- web/
|   |-- app/
|   |-- components/
|   |-- lib/
|   |-- tests/
|   `-- package.json
|
|-- mobile/
|   |-- lib/
|   |-- test/
|   `-- pubspec.yaml
|
|-- docker-compose.yml
`-- README.md
```

Listing A.1 *– Organisation générale du projet Khotwa*

Annexe B

Matrice synthétique des rôles et autorisations

La matrice suivante résume les principales fonctionnalités accessibles selon le rôle de l'utilisateur.

Tableau B.1 – *Matrice synthétique des rôles de Khotwa*

Fonctionnalité	**Participant**	**Formateur**	**Administrateur**
Consulter le catalogue	✓	✓	✓
S'inscrire à une formation	✓	–	–
Consulter les contenus pédagogiques	✓	✓	–
Suivre sa progression	✓	–	–
Passer les quiz	✓	–	–
Télécharger un certificat	✓	–	–
Créer et gérer une formation	–	✓	–
Ajouter des ressources pédagogiques	–	✓	–
Créer et gérer les quiz	–	✓	–
Planifier une classe virtuelle	–	✓	–
Consulter le suivi des participants	–	✓	–
Gérer les utilisateurs	–	–	✓
Valider les candidatures des formateurs	–	–	✓
Modérer les avis signalés	–	–	✓
Consulter le journal d'audit	–	–	✓