

IAAI

RAPPORT DE STAGE

*Conception et Développement d'une Plateforme
E-learning*



Présenté par :
Formation : [Formation]

Établissement : [Établissement]
Zakaria el Khaldi

Période de stage :

Du 5 mai 2025 au [Date de fin]

Sous la supervision de :

[Nom du superviseur]

[Fonction du superviseur]

Année Académique [Année Académique]

Remerciements

Je tiens à exprimer ma profonde gratitude à toutes les personnes qui ont contribué à la réussite de ce stage et à l'élaboration de ce rapport.

Je remercie tout d'abord M. Ahmed Benali, Directeur Technique de LearnTech Solutions, pour son accueil au sein de l'entreprise, sa disponibilité et ses précieux conseils qui m'ont guidé tout au long de cette expérience professionnelle.

Mes remerciements s'adressent également à l'ensemble de l'équipe de LearnTech Solutions pour leur accueil chaleureux, leur soutien et leur partage de connaissances qui ont rendu cette expérience particulièrement enrichissante.

Je suis également reconnaissant envers mon université et mes professeurs pour la qualité de l'enseignement reçu qui m'a permis d'aborder ce stage avec les compétences nécessaires.

Enfin, je remercie ma famille et mes amis pour leur soutien inconditionnel tout au long de mon parcours.

Table des matières

Remerciements	1
1 Introduction	4
1.1 Contexte du Stage	4
1.2 Présentation de l'Entreprise	4
1.3 Objectifs du Stage	4
2 Semaine 1 : Conception et Architecture de la Plateforme E-learning	6
2.1 Vision de la Plateforme	6
2.1.1 Idée Principale et Offres	6
2.1.2 Acteurs Utilisateurs Clés	6
2.1.3 Fonctionnalités Principales	7
2.2 Diagrammes UML	7
2.2.1 Diagramme de Classes	7
2.3 Architecture Backend et Base de Données	7
2.3.1 Choix de l'Architecture Microservices	7
2.3.2 Pile Technologique	8
2.3.3 Architecture Microservices Proposée	9
2.3.4 Communication Inter-Services	9
2.3.5 Modèles de Données des Services	10
3 Semaine 2 : Conception de l'Interface d'Accueil	21
3.1 Conception de la Page d'Accueil	21
3.1.1 Hero Section	21
3.1.2 Section "Where We Are"	22
3.1.3 Sections de Fonctionnalités	22
3.1.4 Section FAQ	23
3.1.5 Appel à l'Action Final	23
3.1.6 Barre de Navigation et Pied de Page	24
3.2 Mise en Place du Système de Paiement	24
3.3 Technologies Utilisées	25
3.4 Optimisations et Performances	26
3.5 Méthodologie de Conception	26
3.5.1 Analyse des Besoins et Recherche	26
3.5.2 Wireframing et Prototypage	26
3.5.3 Développement Itératif	27
3.6 Résultats et Impacts	27
3.7 Conclusion	27

4	Semaine 3 : Développement des Interfaces Utilisateur et Traitement des Données	28
4.1	Intégration et Déploiement de la Base de Données	28
4.1.1	Collecte et Préparation des Données	28
4.1.2	Transfert des Données vers MongoDB	28
4.1.3	Déploiement Cloud sur MongoDB Atlas	29
4.1.4	Connexion de l'Application à la Base de Données	29
4.1.5	Évolution vers une Solution JSON Locale	29
4.2	Conception et Développement de l'Interface Utilisateur	29
4.2.1	Interface d'Accueil des Utilisateurs Connectés	29
4.2.2	Barre Latérale de Navigation	29
4.2.3	Interface de Consultation des Cours	30
4.3	Traitement Avancé des Données de Contenu	30
4.3.1	Défis du Traitement des Données	30
4.3.2	Problématique du Scraping et Nettoyage des Données	32
4.3.3	Implémentation d'un Pipeline de Traitement LLM	33
4.3.4	Stratégie d'Optimisation Multi-LLM	33
4.4	Intégration Frontend-Backend	34
4.4.1	Développement des Routes d'API	34
4.4.2	Optimisation des Requêtes	34
4.4.3	Adaptation à l'Architecture JSON	34
4.5	Tests et Validation	34
4.5.1	Tests d'Interface Utilisateur	35
4.5.2	Tests Fonctionnels et d'Intégration	35
4.5.3	Planification Stratégique pour l'Avenir	35
4.6	Conclusion	35
5	Semaine 4 : Finalisation, Tests et Déploiement	36
6	Conclusion et Perspectives	37
6.1	Synthèse des Réalisations	37
6.2	Compétences Acquisées	37
6.3	Perspectives	37
A	Annexes	38

1 Introduction

Ce rapport présente le travail réalisé lors de mon stage de fin d'études effectué au sein de LearnTech Solutions du 10 mai au 10 juin 2025. Durant cette période, j'ai participé à la conception et au développement d'une plateforme e-learning complète destinée aux entreprises et aux professionnels individuels.

L'objectif de ce stage était de mettre en pratique les connaissances théoriques acquises durant ma formation et de les approfondir dans un contexte professionnel. Ce projet m'a permis de développer des compétences en développement web, en conception d'architecture logicielle, en gestion de bases de données et en traitement de données.

Ce rapport détaille l'ensemble des travaux réalisés durant les différentes semaines du stage, les méthodologies employées, les difficultés rencontrées et les solutions apportées, ainsi que les résultats obtenus.

1.1 Contexte du Stage

LearnTech Solutions est une entreprise spécialisée dans le développement de solutions d'apprentissage en ligne. Face à une demande croissante pour des plateformes de formation à distance, l'entreprise a décidé de développer une nouvelle plateforme e-learning innovante combinant des cours structurés et des services de consultation personnalisés.

Le projet s'inscrit dans un contexte de digitalisation accélérée de la formation professionnelle et de besoin croissant en solutions permettant un apprentissage flexible et adapté aux besoins spécifiques des entreprises et des individus.

1.2 Présentation de l'Entreprise

LearnTech Solutions est une entreprise tech spécialisée dans le développement de solutions éducatives numériques. Fondée en 2020, elle se positionne comme un acteur innovant dans le domaine de l'e-learning et des technologies éducatives. L'entreprise compte une vingtaine de collaborateurs, principalement des développeurs, des designers UX/UI, des spécialistes en contenu pédagogique et des experts en expérience utilisateur.

La mission de LearnTech Solutions est de démocratiser l'accès à une éducation de qualité grâce à des plateformes technologiques avancées et intuitives. L'entreprise se distingue par son approche qui combine apprentissage structuré et interaction humaine à la demande.

1.3 Objectifs du Stage

Les objectifs principaux de ce stage étaient les suivants :

- Participer à la conception initiale de l'architecture de la plateforme e-learning

- Développer des composants back-end pour la gestion des utilisateurs et des cours
- Mettre en place un système de traitement et de nettoyage de données pour les contenus pédagogiques
- Implémenter des fonctionnalités front-end pour l'expérience utilisateur
- Contribuer à l'intégration des différents services et microservices
- Participer aux phases de test et d'évaluation de la plateforme

Ces objectifs s'inscrivent dans le cadre plus large du développement d'une plateforme e-learning complète et innovante, destinée à être commercialisée auprès d'entreprises et de professionnels individuels.

2 Semaine 1 : Conception et Architecture de la Plateforme E-learning

La première semaine du stage a été consacrée à la conception initiale du projet, à la définition de l'architecture backend et des bases de données, ainsi qu'à la création des premiers diagrammes UML.

2.1 Vision de la Plateforme

La phase initiale de conception s'est concentrée sur la définition de la vision, des offres, des utilisateurs cibles et de la valeur unique de la plateforme.

2.1.1 Idée Principale et Offres

La plateforme e-learning conçue repose sur une vision claire :

- **Idée Principale** : Une plateforme e-learning complète pour les entreprises et les professionnels individuels.
- **Offres Clés** :
 - Cours en ligne auto-rythmés (vidéos, images, quiz)
 - Services de support, de conseil et de prestation via une plateforme de réunion personnalisée intégrée
- **Utilisateurs Cibles** : Entreprises (pour la formation des employés) et apprenants individuels (pour l'amélioration des compétences)
- **Valeur Unique** : Un mélange harmonieux d'apprentissage structuré et d'interaction avec des experts à la demande

2.1.2 Acteurs Utilisateurs Clés

Les principaux types d'utilisateurs identifiés pour la plateforme sont :

- **Apprenant Individuel** : S'inscrit, s'abonne, apprend, demande des consultations
- **Employé d'Entreprise** : Apprend via l'abonnement de l'entreprise, demande des consultations
- **Administrateur/Manager d'Entreprise** : Gère le compte de l'entreprise, les employés, les abonnements, attribue les cours, consulte les analyses
- **Créateur de Cours** : Conçoit et élabore le contenu des cours (modules, leçons, quiz)
- **Consultant/Fournisseur de Prestations** : Gère sa disponibilité, anime les sessions via la plateforme de réunion
- **Agent de Support Plateforme** : Assiste les utilisateurs pour les problèmes liés à la plateforme et les questions-réponses

- **Administrateur de la Plateforme** : Supervise l'ensemble de la plateforme, les utilisateurs, le contenu, les paramètres

2.1.3 Fonctionnalités Principales

Les fonctionnalités essentielles se divisent en trois catégories :

- **E-Learning** :
 - Catalogue de cours (filtrable, consultable)
 - Modules, Leçons (vidéo, basées sur des images)
 - Quiz et Évaluations
 - Suivi de la Progression
 - Certificats de Réussite
 - Support Multilingue (EN/FR)
 - Paramètres Utilisateur et Entreprise
- **Consultation et Prestation** :
 - Navigation des services
 - Profils des consultants et disponibilité
 - Système de réservation/demande
 - Intégration API avec la plateforme de réunion personnalisée
 - Facturation des sessions
- **Monétisation** :
 - Abonnements pour utilisateurs individuels (accès illimité)
 - Abonnements pour entreprises (par utilisateur)
 - Réductions et Démonstrations

2.2 Diagrammes UML

La modélisation UML a permis de formaliser la structure et les relations entre les différentes entités du système.

2.2.1 Diagramme de Classes

Le diagramme de classes a été élaboré pour représenter les principales entités du système et leurs relations.

Ce diagramme de classes illustre les relations entre les différentes entités du système, comme les utilisateurs, les cours, les modules, les leçons, les abonnements et les services de consultation. Les cardinalités et les types de relations (composition, agrégation, héritage) ont été définies pour refléter précisément la structure du modèle de données.

2.3 Architecture Backend et Base de Données

2.3.1 Choix de l'Architecture Microservices

L'architecture envisagée pour la plateforme repose sur une approche de microservices, justifiée par plusieurs avantages clés :

- **Scalabilité** : Mise à l'échelle indépendante des services individuels selon les besoins

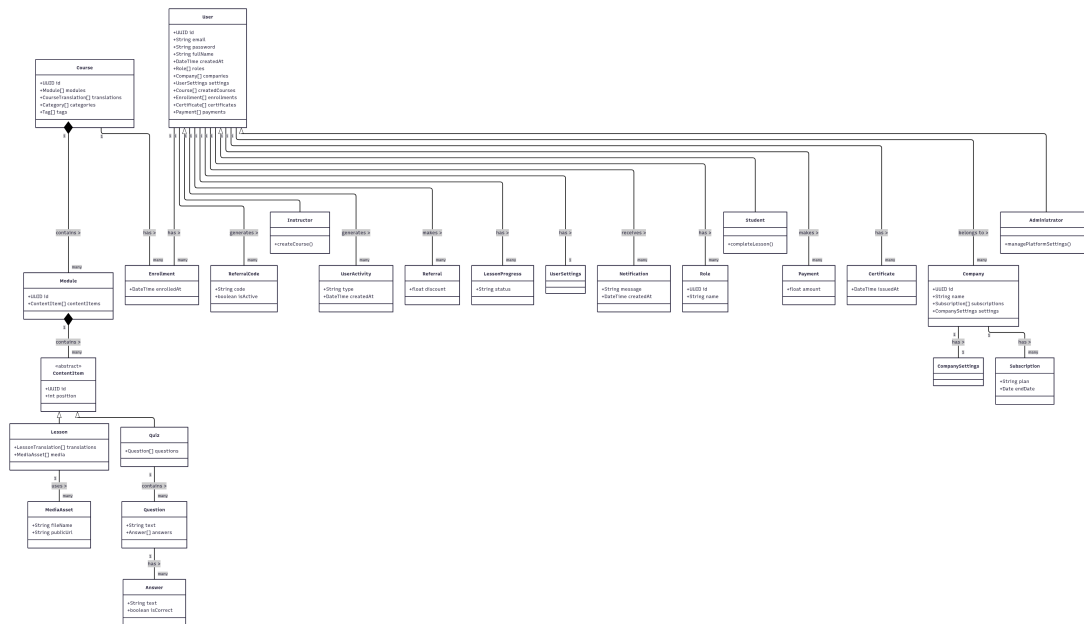


FIGURE 2.1 – **Diagramme de classes** de la plateforme e-learning montrant les principales entités et leurs relations.

- **Maintenabilité** : Possibilité de modifier des parties spécifiques sans impacter l'ensemble du système
- **Résilience** : Limitation de l'impact des défaillances à des services spécifiques
- **Autonomie des Équipes** : Développement, tests et déploiement indépendants par différentes équipes
- **Flexibilité Technologique** : Utilisation des technologies les plus adaptées pour chaque service

2.3.2 Pile Technologique

La pile technologique définie pour le développement comprend :

- **Frontend** : Next.js (React)
- **Microservices Backend** :
 - Go (pour les services critiques en performance et concurrents comme les Notifications, le backend de la plateforme de réunion)
 - Python avec FastAPI (pour les services gourmands en données, développement rapide d'API, par exemple, Catalogue de Cours, Facturation)
 - Node.js avec Express (TypeScript) (pour les opérations I/O intensives, interaction Supabase, par exemple, IAM, Gestion des Médias)
- **Bases de Données** :
 - PostgreSQL (stockage relationnel principal pour la plupart des services)
 - Supabase (pour l'Authentification, le Stockage, et son Postgres géré pour des services spécifiques)
- **Passerelle API** : Nginx (en tant que reverse proxy et passerelle)
- **Broker de Messages** : Apache Kafka (pour une gestion d'événements asynchrones robuste et scalable)
- **Conteneurisation et Orchestration** : Docker (Kubernetes serait une étape logique suivante pour l'orchestration)

2.3.3 Architecture Microservices Proposée

Une décomposition en microservices a été proposée pour répondre aux besoins fonctionnels de la plateforme :

- **Service IAM** : Comptes utilisateurs, rôles, entreprises (Supabase Auth + Node.js/Go)
- **Service Catalogue de Cours** : Structure des cours, métadonnées du contenu (Python/FastAPI + PostgreSQL)
- **Service de Progression de l'Apprenant** : Inscriptions, progression des leçons/quiz, certificats (Python/FastAPI ou Go + PostgreSQL)
- **Service de Réservation** : Demandes de consultation, planification (Node.js/Python + PostgreSQL)
- **Service de Facturation** : Abonnements, paiements, factures (Python/FastAPI + PostgreSQL)
- **Service Média** : Téléchargements de vidéos/images et métadonnées (Supabase Storage + Node.js/Go)
- **Service de Notification** : Envoi d'emails, notifications in-app (Go/Node.js + Kafka)
- **Autres services** : Feedback, Configuration Plateforme, Analytics, etc.

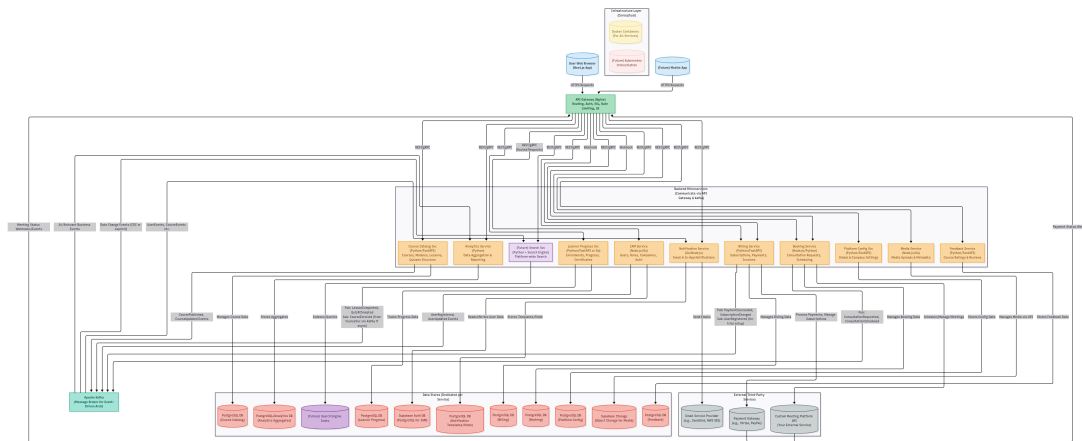


FIGURE 2.2 – Diagramme d'architecture backend et base de données de la plateforme e-learning.

Le diagramme ci-dessus illustre l'architecture microservices proposée, avec les différents services et leurs interactions. Cette architecture permettra un développement modulaire, une maintenance facilitée et une évolution progressive de la plateforme.

2.3.4 Communication Inter-Services

Pour illustrer le fonctionnement de l'architecture, voici un exemple de flux de communication entre les services pour un scénario d'inscription et d'abonnement :

1. **Client (Next.js)** → **Passerelle API (Nginx)** → **Service IAM** (Création de l'utilisateur)
2. **Service IAM** → **Kafka** (Publie 'UserRegisteredEvent')
3. **Service de Notification** (Consomme l'événement) → Envoie un Email de Bienvenue
4. **Client** → **Passerelle API** → **Service de Facturation** (Demande d'abonnement)
5. **Service de Facturation** → Passerelle de Paiement et Mise à jour de la BD interne
6. **Service de Facturation** → **Kafka** (Publie 'SubscriptionActivatedEvent')

7. **Service IAM** (Consomme, met à jour le statut utilisateur) & **Service de Notification** (Consomme, envoie une confirmation)

Cette approche combine des appels API synchrones lorsque nécessaire et des flux événementiels asynchrones pour une meilleure résilience et scalabilité.

2.3.5 Modèles de Données des Services

Dans le cadre de cette première semaine, des modèles de données préliminaires ont été conçus pour chacun des services identifiés. Voici quelques exemples des structures de données principales :

Service IAM (Identity and Access Management)

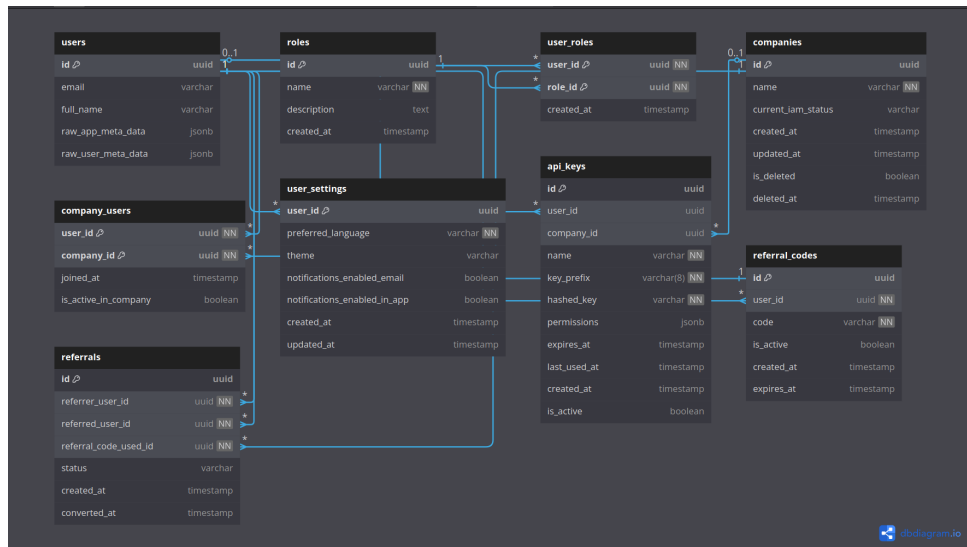


FIGURE 2.3 – Modèle de données du service IAM pour la gestion des utilisateurs et des accès.

Points clés du modèle IAM :

- **Gestion centralisée des utilisateurs** avec tables dédiées aux utilisateurs, rôles et permissions
- **Support multi-tenant** via la table des entreprises (companies)
- **Système de référence** permettant le suivi des recommandations et affiliations
- **Paramètres utilisateurs** stockés de manière structurée pour personnaliser l'expérience
- **Jetons d'authentification** permettant une gestion sécurisée des sessions

Service de Contenu

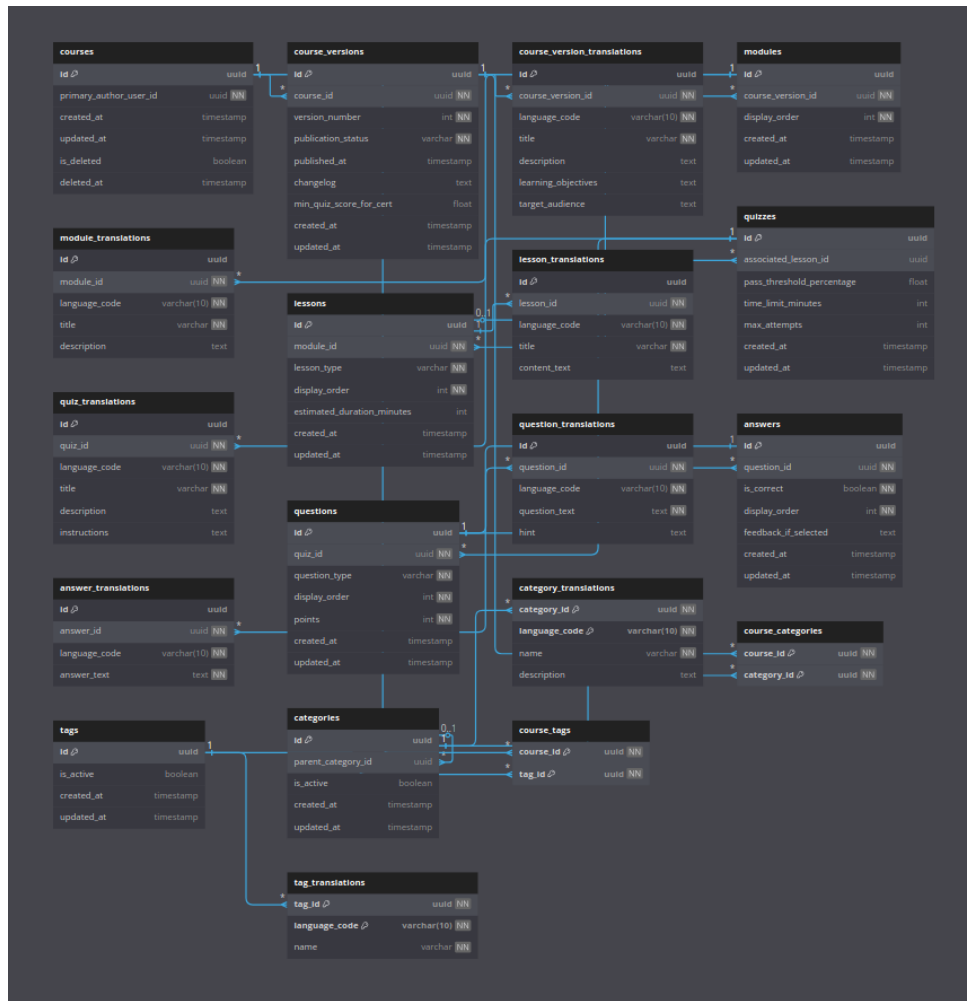


FIGURE 2.4 – Modèle de données du service de contenu pour la gestion des cours et du matériel pédagogique.

Points clés du service de Contenu :

- **Structure hiérarchique** des cours, modules et leçons
- **Support multimédia** avec gestion des vidéos, documents et quizz
- **Métadonnées riches** pour faciliter la recherche et la catégorisation
- **Gestion des versions** permettant la mise à jour du contenu sans perte d'historique
- **Support multilingue** pour internationaliser le contenu pédagogique

Service de Facturation et d'Abonnement

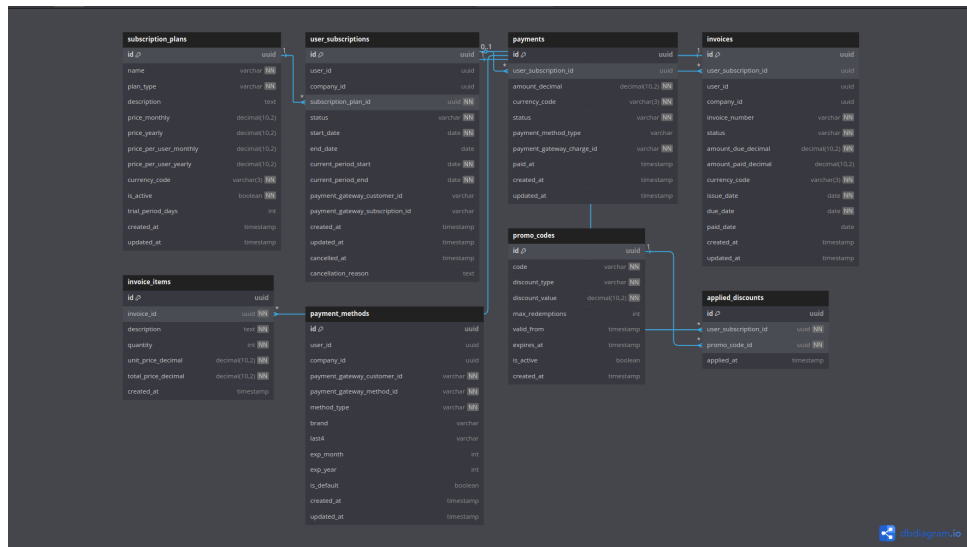


FIGURE 2.5 – Modèle de données du service de facturation pour la gestion des abonnements et des paiements.

Points clés du service de Facturation :

- **Gestion des plans d'abonnement** avec différents niveaux de service
- **Suivi des factures et paiements** pour les utilisateurs individuels et entreprises
- **Support des promotions et réductions** temporaires ou permanentes
- **Historique de facturation** complet pour analyses financières
- **Intégration** avec les passerelles de paiement externes

Service de Certification

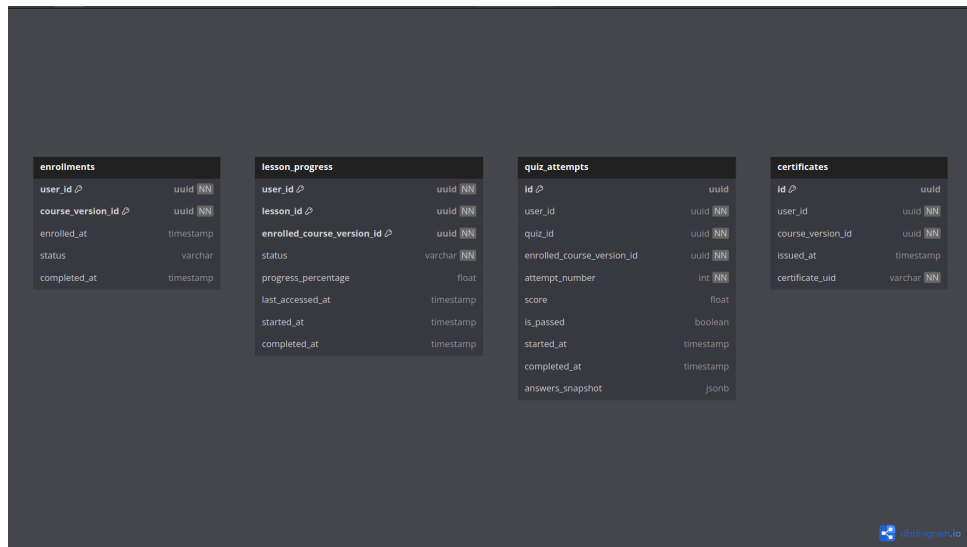


FIGURE 2.6 – Modèle de données du service de certification pour la délivrance et la validation des certificats.

Points clés du service de Certification :

- **Création de certificats** à l'achèvement des cours et modules
- **Validation et vérification** des compétences acquises
- **Badges et récompenses** pour motiver les apprenants
- **Système de validation externe** permettant aux entreprises de vérifier l'authenticité
- **Historique des certifications** pour chaque utilisateur

Service d'Analytique et Reporting

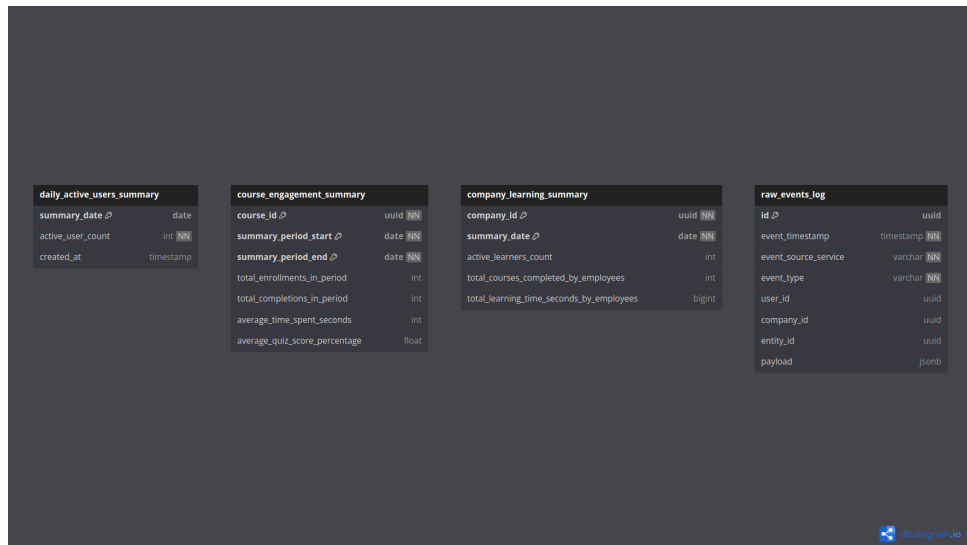


FIGURE 2.7 – Modèle de données du service d'analytique pour le suivi des performances et la génération de rapports.

Points clés du service d'Analytique :

- **Collecte de données d'utilisation** sur toutes les interactions utilisateurs
- **Métriques de performance** pour les cours et modules
- **Rapports personnalisés** pour les administrateurs et entreprises
- **Tableaux de bord en temps réel** pour le suivi des indicateurs clés
- **Système d'alerte** pour identifier les anomalies ou opportunités

Service de Feedback

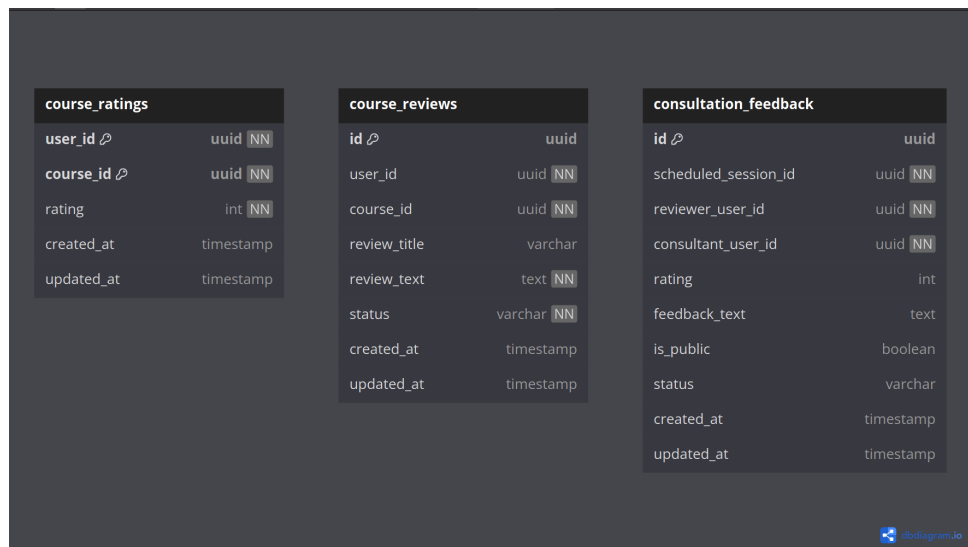


FIGURE 2.8 – Modèle de données du service de feedback pour la collecte et la gestion des retours utilisateurs.

Points clés du service de Feedback :

- **Évaluations et avis** sur les cours et modules
- **Système de notation** permettant une évaluation quantitative
- **Commentaires et suggestions** pour améliorer le contenu
- **Analyse de sentiment** sur les retours textuels
- **Boucle de rétroaction** pour les créateurs de contenu

Service de Notification

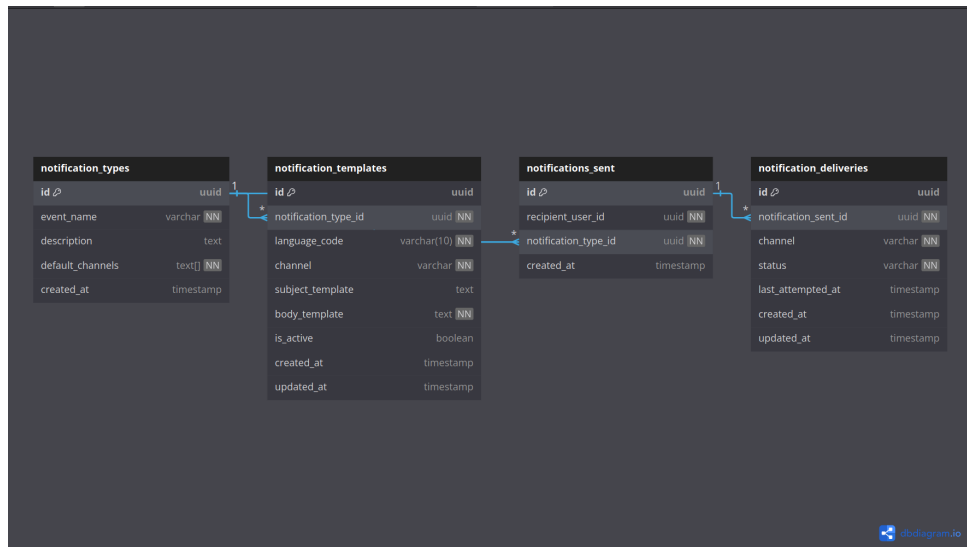


FIGURE 2.9 – Modèle de données du service de notification pour l’envoi et la gestion des notifications.

Points clés du service de Notification :

- **Notifications temps réel** pour les événements importants
- **Préférences de notification** configurables par utilisateur
- **Support multi-canal** : email, push, in-app, SMS
- **Modèles de messages** personnalisables
- **Planification et automatisé** des campagnes de notification

Service de Gestion des Médias

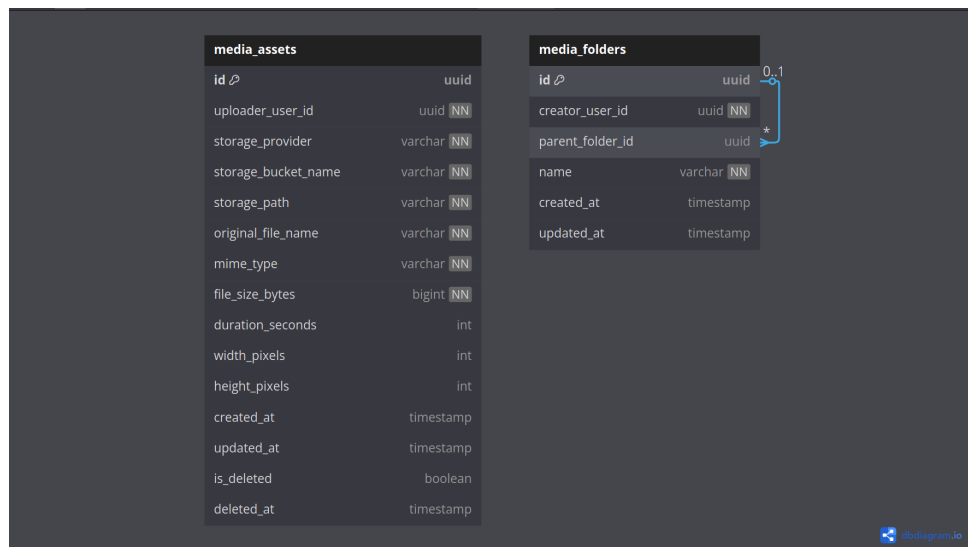


FIGURE 2.10 – Modèle de données du service de gestion des médias pour le stockage et l'accès aux ressources multimédias.

Points clés du service de Médias :

- **Gestion centralisée** des ressources audio, vidéo et images
- **Métadonnées enrichies** pour faciliter la recherche et l'organisation
- **Support de multiples formats** et résolutions
- **Optimisation automatique** pour différents appareils
- **Contrôle des droits d'accès** pour les ressources protégées

Service de Configuration de la Plateforme

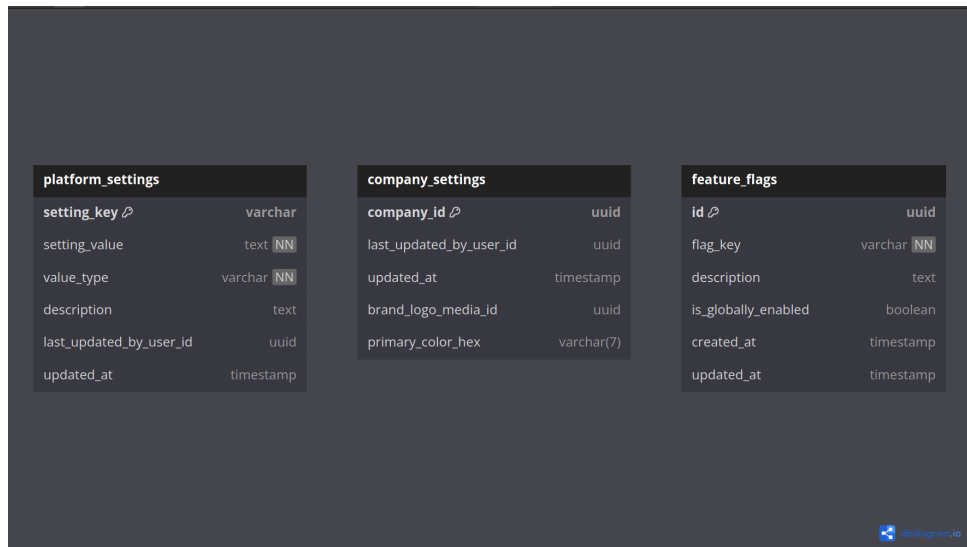


FIGURE 2.11 – Modèle de données du service de configuration pour la gestion des paramètres globaux de la plateforme.

Points clés du service de Configuration :

- **Paramètres système** centralisés et accessibles à tous les services
- **Gestion des drapeaux de fonctionnalités** (feature flags)
- **Configuration par environnement** (développement, test, production)
- **Personnalisation de l'interface** selon les besoins spécifiques
- **Paramètres multilingues** pour localisation de la plateforme

Service de Recherche

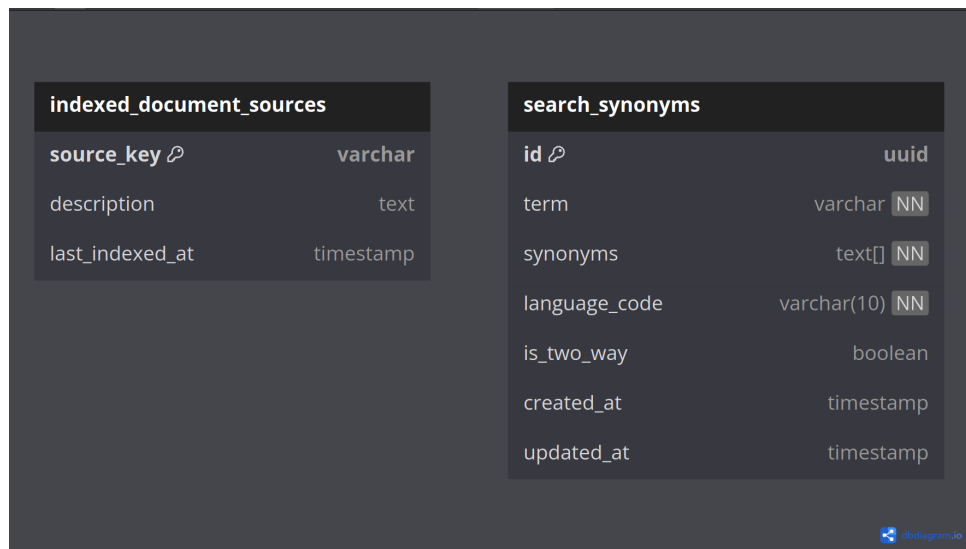


FIGURE 2.12 – Modèle de données du service de recherche pour l’indexation et la recherche de contenu.

Points clés du service de Recherche :

- **Indexation intelligente** du contenu des cours et ressources
- **Recherche full-text** avec prise en charge des synonymes
- **Système de tags** pour améliorer la pertinence des résultats
- **Historique des recherches** pour l’analyse des tendances
- **Algorithme de recommandation** basé sur les comportements de recherche

TABLE 2.1 – Comparaison des caractéristiques techniques des services

Service	Technologie principale	Type de données	Complexité
IAM	Node.js/Go	Utilisateurs	Élevée
Contenu	Python/FastAPI	Éducation	Moyenne
Facturation	Python/FastAPI	Financières	Élevée
Certification	Python/Go	Validation	Moyenne
Analytique	Python	Statistiques	Élevée
Feedback	Node.js	Évaluation	Basse
Notification	Go	Messaging	Moyenne
Médias	Node.js	Binaires	Élevée
Configuration	Go	Paramètres	Basse
Recherche	Python/Elasticsearch	Index	Élevée

3 Semaine 2 : Conception de l'Interface d'Accueil

La deuxième semaine du stage a été consacrée au développement de l'interface d'accueil de la plateforme e-learning. Cette étape était cruciale pour établir l'identité visuelle du projet et créer une première impression positive auprès des utilisateurs.

3.1 Conception de la Page d'Accueil

La page d'accueil a été conçue pour présenter clairement la valeur ajoutée de la plateforme et inciter les utilisateurs à s'inscrire. Plusieurs sections ont été développées pour structurer l'information de manière efficace.

3.1.1 Hero Section

La section principale (Hero Section) a été conçue pour capturer immédiatement l'attention des visiteurs avec un message clair et une incitation à l'action.

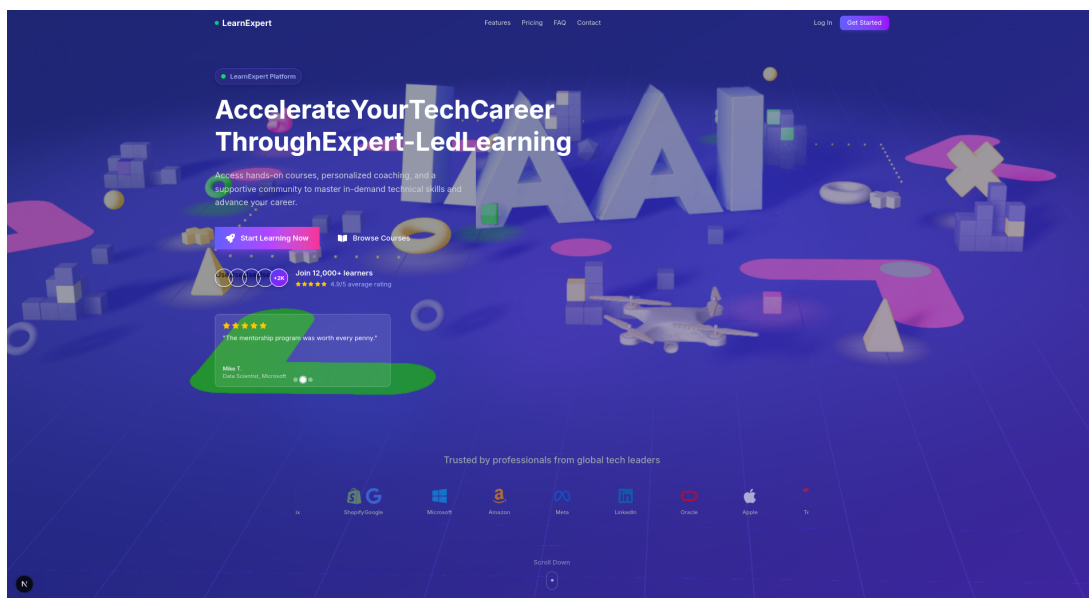


FIGURE 3.1 – **Hero Section améliorée** de la page d'accueil avec effets 3D et animations interactives.

Les éléments clés de cette section comprennent :

- Un titre accrocheur qui met en avant la proposition de valeur principale
- Un sous-titre explicatif sur les avantages de la plateforme
- Un bouton d'appel à l'action bien visible pour l'inscription

- Une illustration moderne avec des effets 3D représentant le concept d'apprentissage en ligne
- Un design responsive qui s'adapte à tous les types d'écrans
- Des animations interactives qui s'activent au survol ou au défilement

3.1.2 Section "Where We Are"

Une section spécifique a été développée pour illustrer la portée géographique et l'influence de la plateforme, renforçant ainsi sa crédibilité auprès des utilisateurs potentiels.

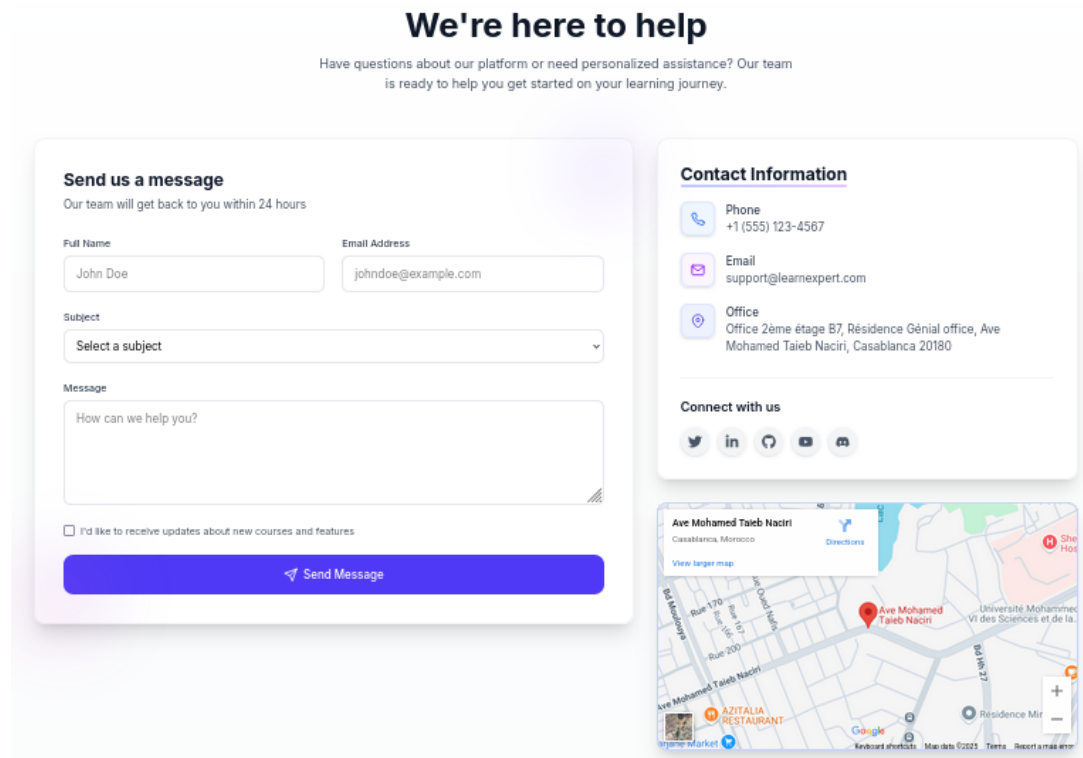


FIGURE 3.2 – Section "Where We Are" présentant la présence globale de la plateforme.

Cette section met en évidence :

- Une carte interactive montrant la présence internationale
- Des statistiques sur le nombre d'utilisateurs par région
- Des témoignages d'utilisateurs de différentes zones géographiques
- Une visualisation de la croissance de la communauté mondiale

3.1.3 Sections de Fonctionnalités

Pour mettre en évidence les principales fonctionnalités de la plateforme, trois sections distinctes ont été développées, chacune mettant en avant un aspect spécifique de l'offre.

Chaque section de fonctionnalités comprend :

- Une illustration ou icône représentative
- Un titre explicite de la fonctionnalité
- Une description détaillée des avantages
- Une mise en page alternée (image à gauche, texte à droite et inversement) pour créer un rythme visuel

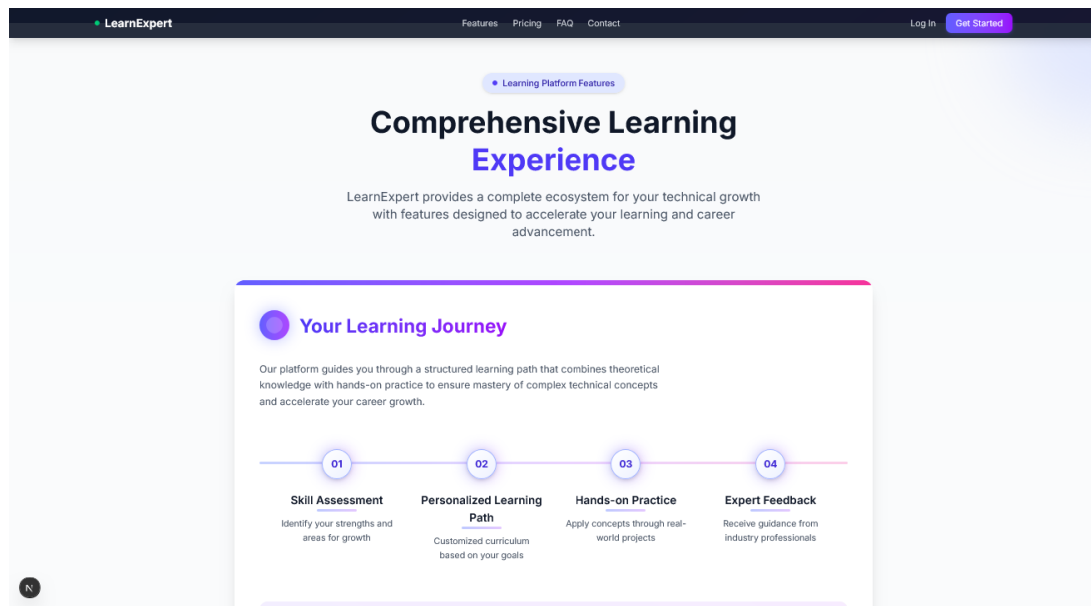


FIGURE 3.3 – Première section de fonctionnalités présentant les cours en ligne.

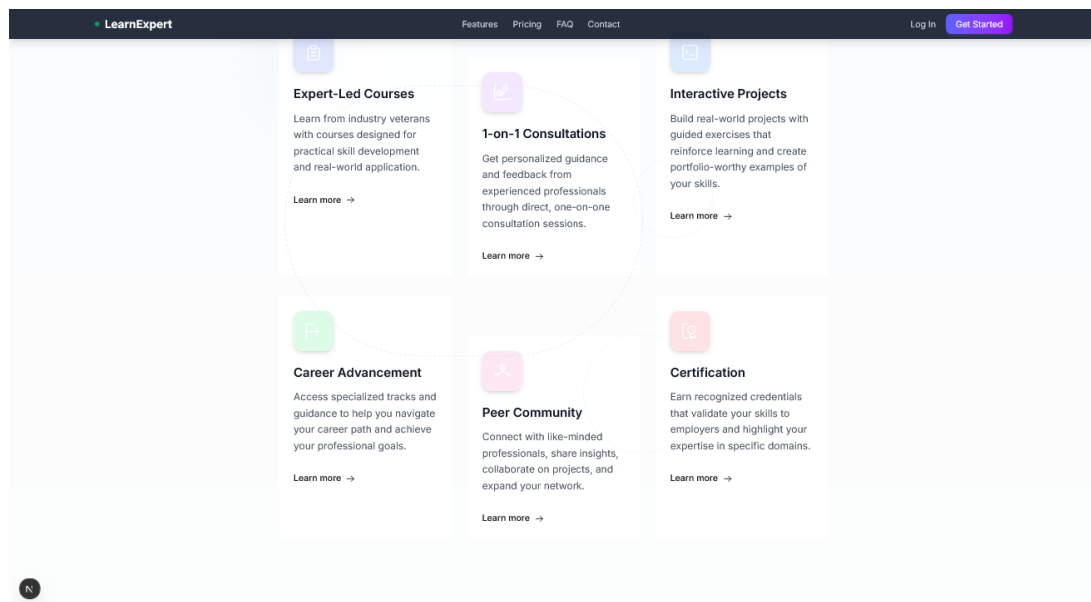


FIGURE 3.4 – Deuxième section de fonctionnalités mettant en avant les services de consultation.

3.1.4 Section FAQ

Pour répondre aux questions fréquentes des utilisateurs potentiels, une section FAQ interactive a été développée avec un système d'accordéon permettant de masquer/afficher les réponses.

3.1.5 Appel à l'Action Final

Une seconde section d'appel à l'action a été ajoutée en bas de page pour inciter les utilisateurs ayant parcouru l'ensemble du contenu à s'inscrire.

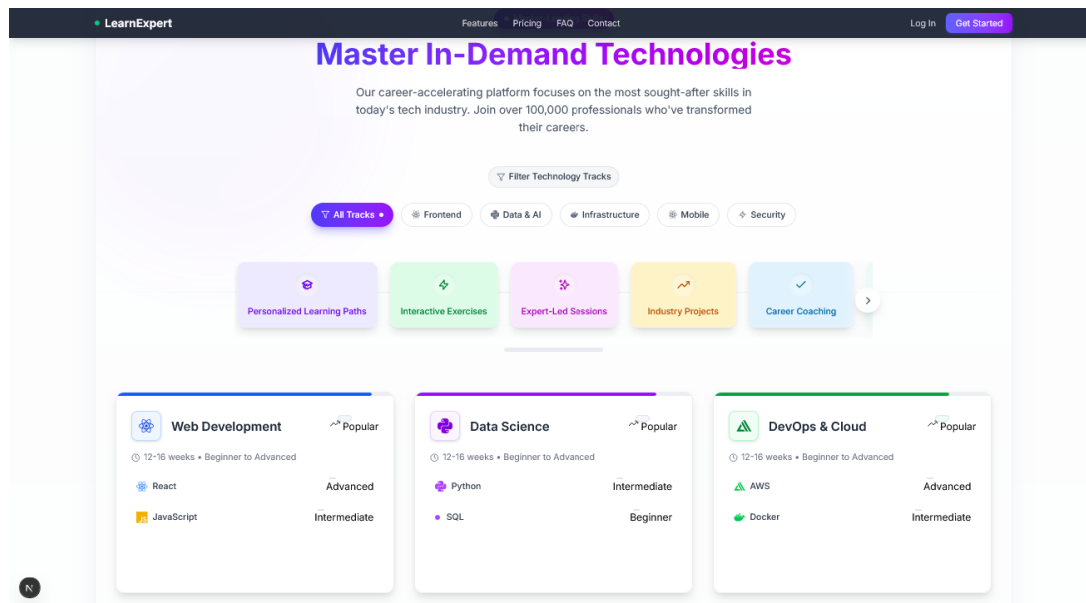


FIGURE 3.5 – Troisième section de fonctionnalités présentant les outils d’analyse et de suivi.

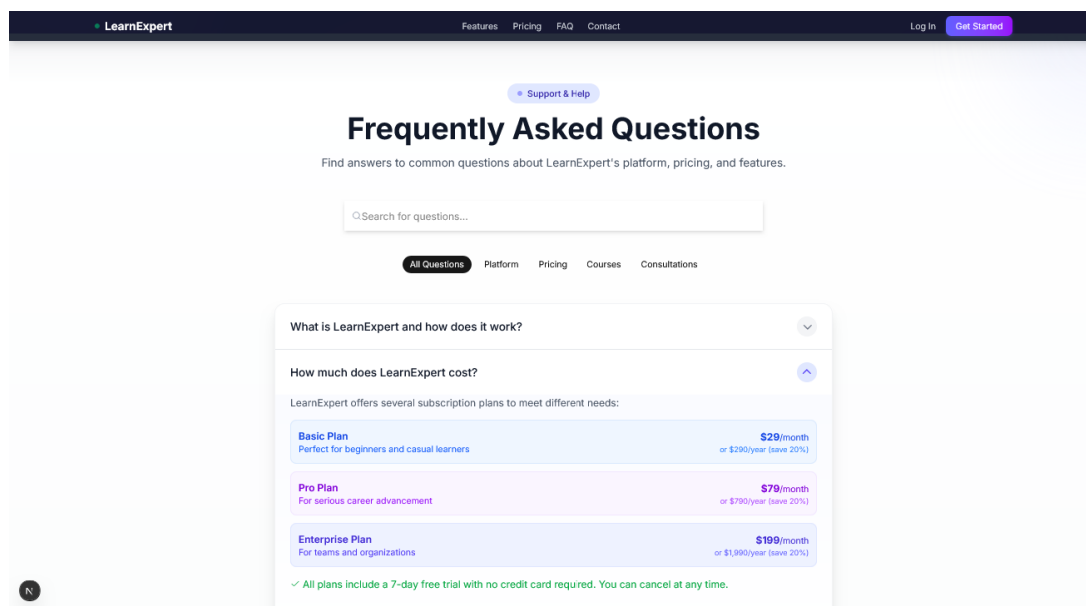


FIGURE 3.6 – Section FAQ avec questions-réponses interactives.

3.1.6 Barre de Navigation et Pied de Page

La barre de navigation et le pied de page ont été conçus pour offrir un accès facile aux différentes sections du site et renforcer la crédibilité de la plateforme.

3.2 Mise en Place du Système de Paiement

Un système de paiement sécurisé a également été intégré à la plateforme pour permettre l’achat d’abonnements et de services.

Les fonctionnalités clés du système de paiement incluent :

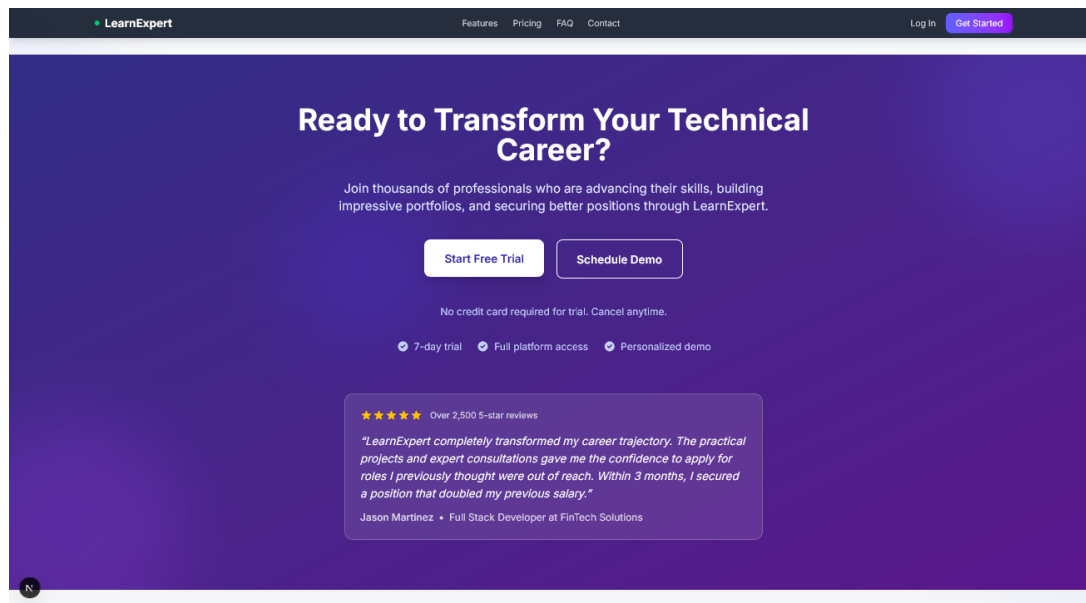


FIGURE 3.7 – Section d’appel à l’action finale incitant à l’inscription.



FIGURE 3.8 – Barre de navigation avec les liens vers les principales sections.

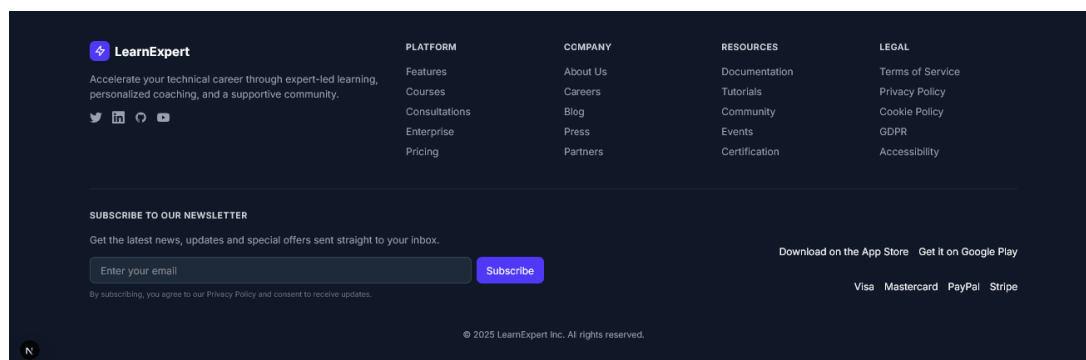


FIGURE 3.9 – Pied de page avec informations légales et liens supplémentaires.

- Sélection de différentes formules d’abonnement
- Interface de saisie des informations de carte bancaire sécurisée
- Intégration avec un service de paiement externe (Stripe)
- Confirmation de paiement et génération de facture automatique
- Gestion des abonnements récurrents et des annulations

3.3 Technologies Utilisées

Pour le développement front-end de cette page d’accueil, les technologies suivantes ont été utilisées :

- **Next.js** : Framework React pour le rendu côté serveur et la génération de sites statiques
- **Tailwind CSS** : Framework CSS utilitaire pour un design responsive et personnalisable
- **Framer Motion** : Bibliothèque d’animations pour ajouter des transitions et effets

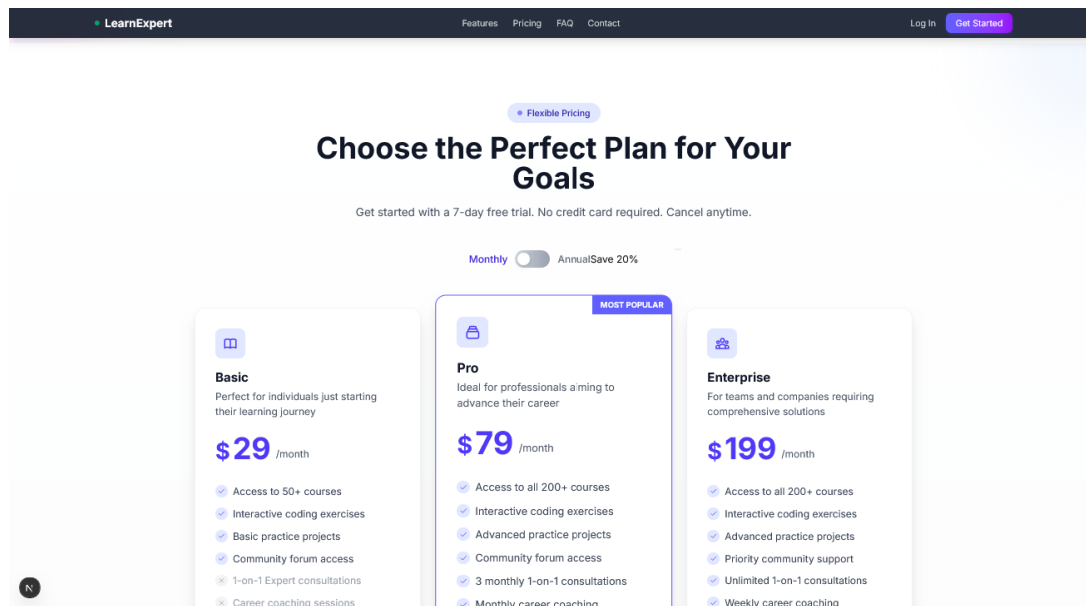


FIGURE 3.10 – Interface de paiement pour les abonnements et services.

visuels

- **React Hook Form** : Gestion des formulaires avec validation
- **TypeScript** : Pour un code plus robuste avec typage statique

3.4 Optimisations et Performances

Une attention particulière a été portée à l'optimisation des performances :

- Utilisation de l'optimisation d'images intégrée à Next.js
- Lazy loading des images et composants non critiques
- Minification du code CSS et JavaScript
- Mise en cache efficace pour réduire les temps de chargement
- Design responsive optimisé pour tous les appareils

3.5 Méthodologie de Conception

Le processus de conception de l'interface a suivi une approche méthodique :

3.5.1 Analyse des Besoins et Recherche

- Étude des plateformes concurrentes pour identifier les meilleures pratiques
- Analyse des besoins des utilisateurs cibles
- Définition des objectifs principaux de la page d'accueil

3.5.2 Wireframing et Prototypage

- Création de wireframes simples pour définir la structure de la page
- Développement de prototypes interactifs avec Figma
- Tests utilisateurs préliminaires pour valider les concepts

3.5.3 Développement Itératif

- Implémentation progressive des différentes sections
- Sessions quotidiennes de feedback avec l'équipe
- Améliorations continues basées sur les retours

3.6 Résultats et Impacts

La conception de cette interface d'accueil a établi une base solide pour le développement des autres composants de la plateforme e-learning, en définissant une identité visuelle cohérente et en présentant clairement la proposition de valeur aux utilisateurs potentiels.

3.7 Conclusion

Cette deuxième semaine de stage a été très productive, avec des avancées significatives dans le développement frontend de la plateforme. La création d'une page d'accueil attrayante et fonctionnelle constitue une étape importante du projet. Les différentes sections développées (Hero Section améliorée, Where We Are, fonctionnalités, FAQ, etc.) permettent de présenter efficacement la valeur ajoutée de la plateforme et d'inciter les utilisateurs à s'inscrire. Les prochaines étapes se concentreront sur le développement des interfaces internes pour les utilisateurs connectés et l'intégration des fonctionnalités backend.

4 Semaine 3 : Développement des Interfaces Utilisateur et Traitement des Données

La troisième semaine du stage a été consacrée au développement des interfaces utilisateur pour la plateforme d'apprentissage en ligne, ainsi qu'à la mise en place d'un système avancé de traitement des données de contenu. Ces avancées ont permis de construire les fondations interactives de la plateforme et d'optimiser le traitement des ressources pédagogiques.

4.1 Intégration et Déploiement de la Base de Données

La première tâche de cette semaine a été de structurer et pérenniser les données collectées lors des phases précédentes du projet.

4.1.1 Collecte et Préparation des Données

La semaine a débuté par la finalisation du processus de web scraping initié précédemment. Ce processus a permis de collecter un volume important de données éducatives provenant principalement de W3Schools. Les données brutes ont ensuite été soumises à un nettoyage préliminaire pour éliminer les éléments non pertinents et standardiser les formats.

4.1.2 Transfert des Données vers MongoDB

Les fichiers JSON contenant les données scrappées et préalablement nettoyées ont été migrés vers une base de données MongoDB. Chaque type de contenu a été organisé dans des collections distinctes, créant ainsi une structure cohérente et facilement interrogeable.

Collection Name	Documents
course_categories	71
tutorials	6134

FIGURE 4.1 – Interface MongoDB Atlas montrant les différentes collections de données.

4.1.3 Déploiement Cloud sur MongoDB Atlas

Pour assurer une accessibilité optimale et une scalabilité future, la base de données a été déployée sur MongoDB Atlas, une plateforme de base de données en tant que service (DBaaS). Cette solution cloud offre plusieurs avantages :

- Haute disponibilité avec réplication automatique
- Surveillance en temps réel des performances
- Sauvegardes automatisées
- Sécurité renforcée avec authentification et chiffrement
- Scalabilité horizontale et verticale selon les besoins

4.1.4 Connexion de l'Application à la Base de Données

Une fois la base de données déployée, la connexion entre l'application backend et l'instance MongoDB hébergée sur Atlas a été établie. Cette étape cruciale a permis d'assurer que l'application puisse lire et écrire des données de manière fiable et sécurisée.

4.1.5 Évolution vers une Solution JSON Locale

Après des tests approfondis, une décision stratégique a été prise de migrer d'une architecture MongoDB vers une solution basée sur des fichiers JSON locaux. Cette transition a été motivée par :

- La réduction de la complexité architecturale pour la phase actuelle du projet
- L'amélioration de l'expérience de développement avec un accès direct aux données
- La compatibilité optimale avec le framework Next.js et ses capacités de rendu côté serveur
- La simplification du déploiement et de la maintenance pour les premiers stades du projet

Cette migration a nécessité l'adaptation des modèles de données et le développement d'une nouvelle logique d'accès aux données, mais a considérablement simplifié le processus de développement.

4.2 Conception et Développement de l'Interface Utilisateur

Après avoir développé la page d'accueil la semaine précédente, l'accent a été mis sur la création des interfaces principales que les utilisateurs utiliseront quotidiennement pour accéder aux contenus et suivre leur progression.

4.2.1 Interface d'Accueil des Utilisateurs Connectés

Une interface d'accueil spécifique pour les utilisateurs connectés a été développée, offrant un accès rapide aux cours en cours, aux recommandations personnalisées et aux dernières activités.

4.2.2 Barre Latérale de Navigation

Une barre latérale de navigation intuitive a été implémentée pour faciliter l'accès aux différentes sections de la plateforme.

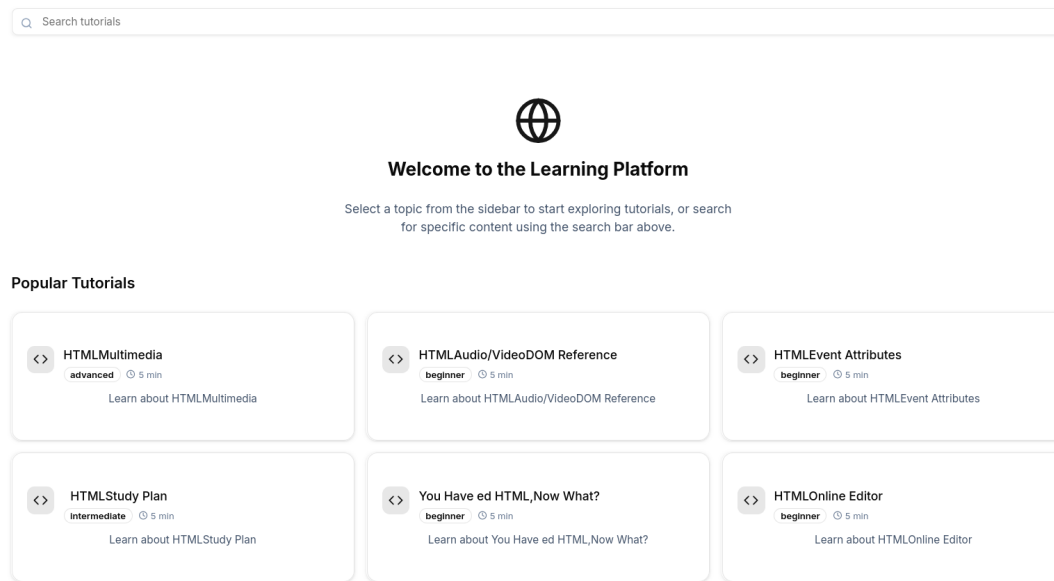


FIGURE 4.2 – Interface d’accueil personnalisée pour les utilisateurs connectés.

Cette barre latérale comprend :

- Accès au tableau de bord personnel
- Catalogue de cours et formations
- Suivi de progression
- Calendrier des sessions de consultation
- Paramètres du compte
- Centre de notifications

4.2.3 Interface de Consultation des Cours

L’interface de consultation des cours a été conçue pour offrir une expérience d’apprentissage immersive et efficace.

Les caractéristiques principales de cette interface incluent :

- Affichage clair du contenu théorique avec mise en forme optimisée
- Navigation intuitive entre les différentes sections du cours
- Exercices interactifs intégrés directement dans l’interface
- Suivi de progression en temps réel
- Possibilité de prendre des notes contextuelles
- Mode sombre/clair pour améliorer le confort de lecture

4.3 Traitement Avancé des Données de Contenu

Une partie importante de cette semaine a été consacrée à l’optimisation du traitement des données de contenu éducatif.

4.3.1 Défis du Traitement des Données

Le traitement des données collectées présentait plusieurs défis :

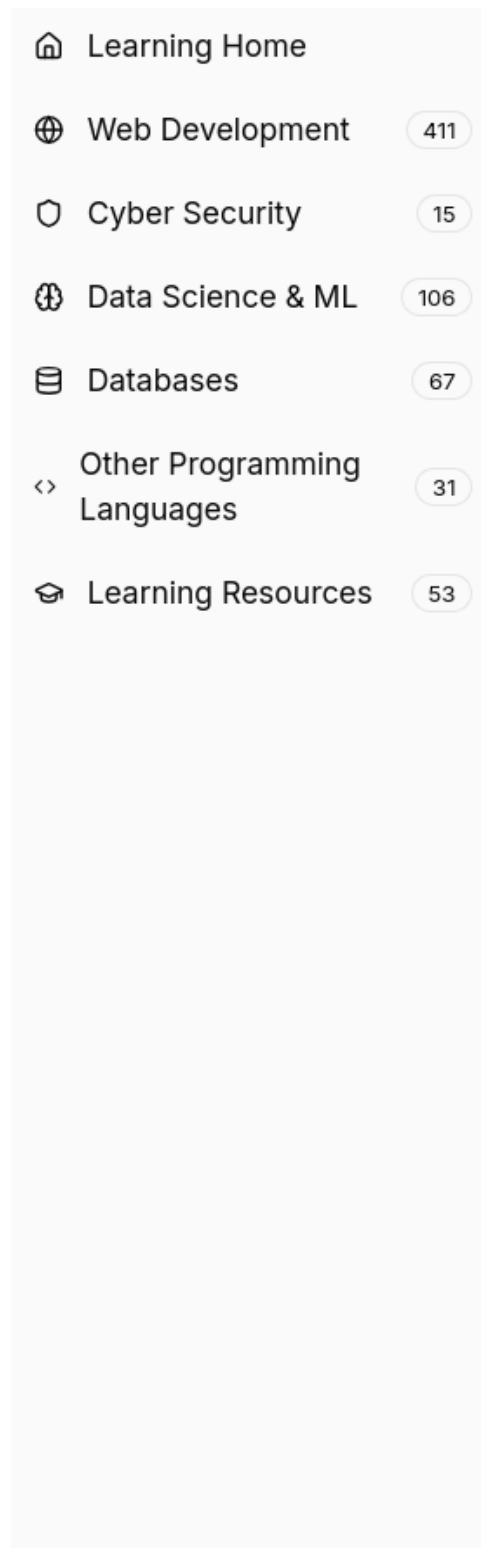


FIGURE 4.3 – **Barre latérale de navigation** avec accès aux principales fonctionnalités.

- Distinction entre le contenu textuel explicatif et les exemples de code
- Structuration cohérente des exemples et exercices
- Préservation des formatages spécifiques (tableaux, listes, etc.)
- Gestion des caractères spéciaux et encodages
- Extraction des métadonnées pertinentes

[← Back to Tutorials](#)

HTMLMultimedia

Advanced

⌚ 5 min read

LearnHTML

Learn about HTMLMultimedia with clear explanations and practical examples.

📖 Contents

[Introduction](#)[Details](#)

Introduction

HTML

Multimedia

FIGURE 4.4 – Interface de consultation des cours - Partie théorique.

☆ Practice Exercises

Create a Layout

Difficulty: easy

Build a simple layout using the techniques from this tutorial.

```
<!-- Add your code here -->
```

Quiz

Question 1

What is HTML Multimedia Multimedia on the web?

☐ None of the above.☒ sound☐ None of the above.☐ Multimedia

Explanation: The correct definition of HTML Multimedia Multimedia on the web is 'sound'.

FIGURE 4.5 – Interface de consultation des cours - Partie pratique avec exercices interactifs.

4.3.2 Problématique du Scraping et Nettoyage des Données

L'un des défis majeurs rencontrés a été la séparation du contenu textuel explicatif et des extraits de code dans les sections détaillées des cours. Plusieurs approches ont été explorées :

— Tentatives initiales avec des expressions régulières et des algorithmes de traitement de

texte, atteignant une précision maximale de 80%

- Expérimentation avec un agent IA basé sur des modèles de langage large (LLM) pour effectuer cette séparation de manière plus intelligente
- Découverte finale que la structure HTML particulière de W3Schools nécessitait une approche de scraping spécifique

La solution définitive a impliqué une refonte du script de scraping pour mieux cibler les éléments HTML spécifiques, permettant ainsi d'obtenir des données brutes de meilleure qualité dès le départ.

4.3.3 Implémentation d'un Pipeline de Traitement LLM

Pour surmonter ces défis, un pipeline de traitement basé sur des modèles de langage large (LLM) a été conçu et implémenté.

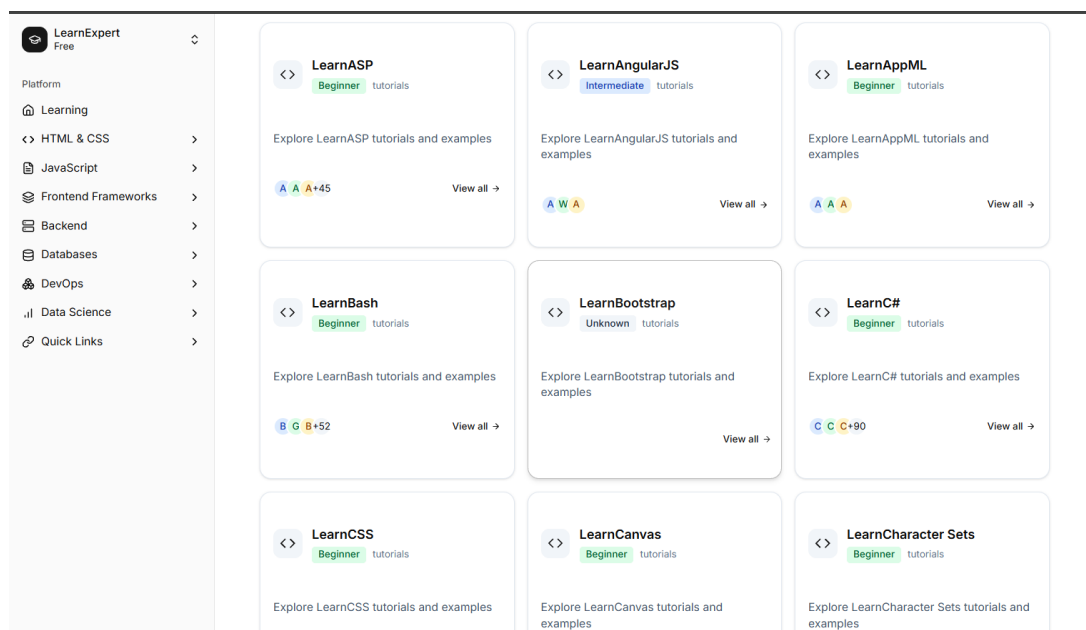


FIGURE 4.6 – Interface d'administration du pipeline LLM montrant les statistiques de traitement.

4.3.4 Stratégie d'Optimisation Multi-LLM

Le traitement initial utilisant un LLM unique ayant démontré une excellente précision mais des performances limitées en termes de vitesse, une stratégie d'optimisation a été mise en place :

- **Utilisation simultanée de plusieurs fournisseurs LLM** : Parallélisation des requêtes vers différentes API
- **Traitement asynchrone** : Optimisation de la latence perçue en traitant plusieurs segments de données simultanément
- **Segmentation intelligente** : Division des données en unités de traitement optimales pour chaque type de modèle
- **Gestion des requêtes** : Assignment statique de segments spécifiques à des modèles dédiés pour simplifier la gestion de la concurrence

Cette approche a permis de réduire significativement le temps de traitement tout en maintenant une qualité optimale, passant d'environ 15 heures à 7-8 heures pour l'ensemble des données.

4.4 Intégration Frontend-Backend

Une attention particulière a été portée à l'intégration efficace entre le frontend et le backend de la plateforme.

4.4.1 Développement des Routes d'API

Des routes d'API RESTful ont été développées pour permettre à l'interface utilisateur d'accéder aux données structurées dans MongoDB :

- Routes d'authentification et de gestion des utilisateurs
- Routes d'accès au catalogue de cours
- Routes de suivi de progression
- Routes de gestion des consultations et réservations
- Routes d'analytics et de rapports

4.4.2 Optimisation des Requêtes

Les requêtes MongoDB ont été optimisées pour garantir des performances optimales :

- Création d'index appropriés pour accélérer les recherches fréquentes
- Limitation des champs retournés aux données strictement nécessaires
- Utilisation de projections pour alléger les transferts de données
- Mise en place de pagination pour les listes volumineuses
- Implémentation de mécanismes de mise en cache pour les données fréquemment consultées

4.4.3 Adaptation à l'Architecture JSON

Suite à la migration vers des fichiers JSON locaux, l'architecture d'accès aux données a été adaptée :

- Développement de fonctions de récupération (fetch) optimisées pour les fichiers JSON
- Mise en place d'un système de chargement dynamique des données pour les différentes sections de l'interface
- Intégration des capacités de rendu côté serveur de Next.js pour améliorer les performances

Cette adaptation a permis de maintenir toutes les fonctionnalités prévues tout en simplifiant l'architecture globale du système.

4.5 Tests et Validation

Pour garantir la qualité et la fiabilité des interfaces et systèmes développés, plusieurs types de tests ont été mis en place :

4.5.1 Tests d'Interface Utilisateur

- **Tests de compatibilité** : Vérification du rendu sur différents navigateurs (Chrome, Firefox, Safari, Edge)
- **Tests de responsive design** : Validation de l'adaptation aux différentes tailles d'écran
- **Tests d'accessibilité** : Conformité aux standards WCAG pour garantir l'inclusivité
- **Tests d'utilisabilité** : Évaluation de l'intuitivité et de la fluidité de navigation

4.5.2 Tests Fonctionnels et d'Intégration

- **Tests des routes d'API** : Validation des entrées/sorties et gestion des erreurs
- **Tests d'intégration** : Vérification de la communication correcte entre frontend et backend
- **Tests de performance** : Mesure des temps de réponse et optimisation des goulots d'étranglement
- **Tests de charge** : Simulation d'utilisation intensive pour évaluer la robustesse

4.5.3 Planification Stratégique pour l'Avenir

En fin de semaine, une réflexion stratégique a été menée concernant l'enrichissement du contenu et la différenciation de la plateforme :

- Identification du besoin de diversifier les sources de contenu au-delà de W3Schools
- Planification de l'intégration d'éléments interactifs avancés, comme un éditeur de code basé sur les technologies VSCode
- Initiation à l'apprentissage des techniques d'optimisation pour les moteurs de recherche (SEO) pour améliorer la visibilité future de la plateforme

4.6 Conclusion

Cette troisième semaine a marqué des avancées significatives dans le développement de la plateforme e-learning. La mise en place d'interfaces utilisateur intuitives et ergonomiques, couplée à un système sophistiqué de traitement des données éducatives, a permis de poser les fondations solides pour l'expérience utilisateur finale. L'utilisation innovante de technologies comme MongoDB Atlas et les modèles de langage large (LLM) a apporté des solutions efficaces aux défis techniques rencontrés.

Les interfaces développées durant cette semaine offrent maintenant aux utilisateurs un environnement d'apprentissage fluide et immersif, tandis que l'optimisation du traitement des données garantit une qualité exceptionnelle du contenu éducatif proposé. Les décisions stratégiques prises, notamment la simplification de l'architecture de données et la planification d'éléments différenciateurs, posent les bases d'une plateforme compétitive et évolutive.

5 Semaine 4 : Finalisation, Tests et Déploiement

Contenu de la semaine 4 à venir. Cette section sera complétée ultérieurement.

6 Conclusion et Perspectives

Ce stage de quatre semaines chez LearnTech Solutions m'a permis de participer activement au développement d'une plateforme e-learning innovante. Au cours de cette période, j'ai pu mettre en pratique mes connaissances théoriques et acquérir de nouvelles compétences techniques dans un environnement professionnel stimulant.

6.1 Synthèse des Réalisations

Durant ces quatre semaines, les principales réalisations ont été :

- La conception de l'architecture microservices pour la plateforme
- La participation au développement des interfaces utilisateur
- L'implémentation de systèmes de traitement et nettoyage de données
- La contribution à l'optimisation des performances et à l'amélioration de l'expérience utilisateur

6.2 Compétences Acquisées

Ce stage m'a permis de développer de nombreuses compétences :

- **Compétences techniques** : Approfondissement des connaissances en développement web moderne (Next.js, API REST), en architecture microservices et en traitement de données avec des modèles LLM
- **Compétences méthodologiques** : Maîtrise des outils de gestion de projet agile, de versionnement et de déploiement continu
- **Compétences transversales** : Amélioration des capacités de communication, de travail en équipe et d'adaptation à un environnement professionnel

6.3 Perspectives

Ce projet ouvre plusieurs perspectives intéressantes :

- **Pour l'entreprise** : Finalisation et commercialisation de la plateforme e-learning, avec des fonctionnalités supplémentaires comme l'intelligence artificielle pour la personnalisation des parcours d'apprentissage
- **Pour ma carrière** : Approfondissement des compétences en développement full-stack et en architecture de systèmes complexes

En conclusion, ce stage a été une expérience extrêmement enrichissante qui m'a permis d'approfondir mes connaissances techniques tout en développant ma compréhension des enjeux liés au développement de solutions éducatives numériques. Les compétences acquises et l'expérience professionnelle seront des atouts précieux pour ma future carrière.

A Annexes

[Diagrammes UML, exemples de code, captures d'écran de l'interface]