

Onglet 1

Je veux creer une plateforme de formations et j'ai besoin que tu m'aide avec une api en laravel en fonctions des instructions suivantes:

1. Un systeme d'auth (create account, login, forgot password)
2. Le module de formations (un titre, une description, des chapitres, des cours, etc.. y ajouter des options de securite, d'access, pouvoir configurer les informations lie a une formation par exemple si elle est payante, si certains chapitres et/ou cours sont gratuits, prevoir une image de couverture, une video/texte de presentation optionnel, et autre options lie au formations que tu pourra connaitre) aussi les video doivent etre stockes soit sur vimeo via api ou youtube selon le besoin donc creer egalement un module de gestion de medias), une formations peut avoir des ressource telechargeable (pdf, image, etc(gratuit ou pas a configurer)); un temps de disponibilite de la formation doit egalement pouvoir etre configurable
3. Un module de tests (Donner la possibilite d'ajouter des test a une formation, soit apres un cours, un chapitre ou a la fin de la formation, les test doivent etre configurable completement par exemple de le nombre de tentatives, le temps, le score pour valider si la validation est automatique, le possibiliter d'avoir une validation manuelle donc donner la possibilite au createur de la formation d'évaluer lui même un test, y mettre une note, une mention, des commentaires et autre options importante); les tests peuvent etre de plusieurs formats (QCM, Text, audio, etc..) prevoir des options de configurations modulable pour les differents cas)
4. Un module de gestion des permissions et roles afin de controller les differents actions
5. un Dashboard pour l'étudiant où il pourra voir ses formations en cours, terminer, etc et pouvoir télécharger des recus, et tout le nécessaire; Evidemment pour passer une formation si disponible
6. Pour le module de paiement intégrer Stripe et paypal pour le moment et donner une flexibilité de pouvoir ajouter d'autres moyens aisement

Voilà une base n'hésite pas à ajouter des éléments que tu juges pertinents

Et avant de commencer fait moi un fichier md dans lequel tu décris le projet, les fonctionnalités à développer etc.. et une roadmap du dev du projet

Onglet 2



Plateforme de Formations en Ligne - Documentation Complète



Vue d'ensemble du projet

Une plateforme complète de formations en ligne permettant aux instructeurs de créer, gérer et monétiser leurs cours, et aux étudiants d'accéder à un contenu éducatif de qualité avec un système de certification.



Architecture Technique

Stack Technologique

- **Backend:** Laravel 10.x
- **Base de données:** MySQL 8.0+
- **Authentification:** Laravel Sanctum (API Token)
- **Stockage:** Laravel Filesystem (Local/S3)
- **Paiements:** Stripe, PayPal
- **Médias externes:** Vimeo API, YouTube API
- **Cache:** Redis
- **Queue:** Redis/Database

Principes de conception

- Architecture RESTful
 - Repository Pattern
 - Service Layer Pattern
 - Event-Driven Architecture
 - SOLID Principles
-



Fonctionnalités Détaillées

1. 🔒 Système d'Authentification & Autorisation

Authentification

- Inscription utilisateur avec validation email
- Connexion (email/password)

- Récupération de mot de passe (reset via email)
- Vérification email obligatoire
- Authentification à deux facteurs (2FA) - optionnel
- Déconnexion et révocation des tokens
- Gestion des sessions multiples

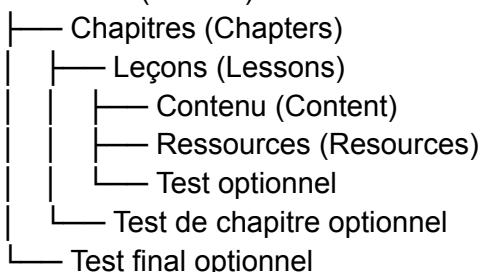
Gestion des Rôles et Permissions

- **Rôles prédéfinis:**
 - **Super Admin:** Contrôle total de la plateforme
 - **Admin:** Gestion des utilisateurs et courses
 - **Instructor:** Crédit et gestion de formations
 - **Student:** Accès aux formations
 - **Permissions granulaires:**
 - `courses.create, courses.edit, courses.delete, courses.view`
 - `chapters.manage, lessons.manage`
 - `tests.create, tests.evaluate`
 - `users.manage, roles.manage`
 - `payments.view, analytics.view`
 - **Système de permissions:**
 - Attribution de permissions par rôle
 - Permissions personnalisées par utilisateur
 - Héritage de permissions
 - Middleware de vérification des permissions
-

2. Module de Formations

Structure hiérarchique

Formation (Course)



Entité Formation (Course)

Informations générales:

- Titre, description, slug
- Image de couverture (upload)
- Vidéo/texte de présentation
- Catégorie et tags
- Niveau (débutant, intermédiaire, avancé)
- Langue
- Durée estimée totale

Configuration d'accès:

- Statut: Brouillon, Publié, Archivé
- Visibilité: Publique, Privée, Sur invitation
- Période de disponibilité (date début/fin)
- Limite d'inscriptions
- Pré-requis (formations obligatoires)
- Approbation manuelle des inscriptions

Configuration commerciale:

- Gratuit/Payant
- Prix et devise
- Réductions (pourcentage/montant fixe)
- Période promotionnelle
- Abonnement mensuel/annuel
- Accès à vie ou limité dans le temps
- Politique de remboursement

Options avancées:

- Certificat de fin de formation
- Support instructeur (forum, Q&A)
- Téléchargement du contenu autorisé
- Accès mobile/hors ligne
- Watermark sur les vidéos

Chapitres (Chapters)

- Titre et description
- Ordre d'affichage
- Accès gratuit ou payant
- Publication différée
- Durée estimée

Leçons (Lessons)

- Titre et description
- Type de contenu:
 - Vidéo (YouTube/Vimeo)
 - Texte enrichi (WYSIWYG)

- PDF
- Audio
- Quiz interactif
- Présentation (slides)
- Ordre d'affichage
- Durée
- Accès gratuit (preview)
- Téléchargeable ou non
- Progression automatique

Ressources téléchargeables

- Fichiers attachés (PDF, images, ZIP, etc.)
 - Titre et description
 - Taille maximale: 50 MB par fichier
 - Gratuit ou payant
 - Protection par watermark
 - Limitation de téléchargements
-

3. Module de Gestion des Médias

Intégration Vimeo

- Upload de vidéos via API Vimeo
- Gestion de la confidentialité
- Récupération des thumbnails
- Statistiques de visionnage
- Protection du contenu (domain whitelist)

Intégration YouTube

- Import de vidéos YouTube existantes
- Récupération des métadonnées
- Embed sécurisé
- Playlists automatiques

Stockage local/cloud

- Support AWS S3, DigitalOcean Spaces
- Compression automatique des images
- Génération de miniatures
- URLs signées pour sécurité
- CDN pour performance

Gestion des fichiers

- Upload multi-fichiers

- Validation des types MIME
 - Scan antivirus
 - Organisation par dossiers
 - Versionning des fichiers
-

4. Module de Tests & Évaluations

Types de tests

- **QCM (Questionnaire à Choix Multiple):**
 - Choix unique ou multiple
 - Points par question
 - Explication des réponses
 - Ordre aléatoire des questions/réponses
- **Questions ouvertes:**
 - Texte court ou long
 - Évaluation manuelle obligatoire
 - Grille de notation
- **Vrai/Faux:**
 - Questions binaires
 - Justification optionnelle
- **Questions audio/vidéo:**
 - Enregistrement de réponses
 - Évaluation manuelle
- **Exercices pratiques:**
 - Upload de fichiers
 - Code source
 - Projets

Configuration des tests

Paramètres généraux:

- Titre et instructions
- Position: Après leçon, chapitre ou formation
- Type d'évaluation: Formative ou sommative
- Durée limitée ou illimitée
- Nombre de tentatives (1 à illimité)
- Affichage des réponses après soumission
- Score minimum de validation (%)

Options avancées:

- Questions aléatoires depuis une banque
- Ordre aléatoire des questions
- Affichage page par page ou tout en une fois
- Retour arrière autorisé
- Enregistrement automatique des brouillons
- Anti-triche (désactivation copier-coller, surveillance webcam)

Validation:

- Automatique (QCM, Vrai/Faux)
- Manuelle (Questions ouvertes, projets)
- Mixte

Évaluation et notation

- Note sur 100 ou personnalisée
- Mentions (Excellent, Bien, Passable, Échec)
- Commentaires de l'instructeur
- Feedback détaillé par question
- Historique des tentatives
- Certificat si validation réussie

Banque de questions

- Réutilisation des questions
 - Catégorisation par sujet
 - Niveau de difficulté
 - Tags et recherche
-

5. Module de Paiement

Intégrations

Stripe:

- Paiements par carte
- Webhooks pour confirmation
- Gestion des remboursements
- Abonnements récurrents
- 3D Secure

PayPal:

- PayPal Standard
- PayPal Express Checkout
- Gestion des litiges

Architecture extensible:

- Interface [PaymentGateway](#)
- Ajout facile de nouveaux gateways (Flutterwave, Paystack, etc.)
- Configuration par pays

Fonctionnalités

- Panier d'achat
 - Codes promo
 - Facturation automatique
 - Reçus PDF
 - Historique des transactions
 - Gestion des taxes
 - Remboursements partiels/complets
 - Split payment (commission plateforme/instructeur)
-

6. Dashboard Étudiant

Vue d'ensemble

- Formations en cours
- Formations terminées
- Prochaines échéances
- Progression globale
- Statistiques personnelles

Mes formations

- Liste des formations inscrites
- Barre de progression
- Dernier cours consulté
- Notes des tests
- Certificats obtenus

Gestion des achats

- Historique des paiements
- Téléchargement des reçus/factures
- Demandes de remboursement
- Abonnements actifs

Certificats & Badges

- Téléchargement PDF
- Partage sur LinkedIn
- Vérification par QR code
- Galerie de badges

Paramètres

- Profil utilisateur
 - Préférences de notification
 - Confidentialité
 - Connexion et sécurité
-

7. Dashboard Instructeur

Gestion des formations

- Création/édition de cours
- Statistiques par cours
- Revenus générés
- Évaluations et commentaires

Gestion des étudiants

- Liste des inscrits
- Progression individuelle
- Communication (messages, annonces)
- Certifications délivrées

Évaluation des tests

- File d'attente des tests à corriger
- Interface de notation
- Historique des corrections

Analytics

- Taux d'achèvement
- Temps moyen par leçon
- Sections problématiques
- Taux de réussite aux tests

Revenus

- Tableau de bord des ventes
 - Historique des paiements
 - Demandes de retrait
-

8. Dashboard Admin

Gestion des utilisateurs

- Liste et recherche
- Édition des profils
- Attribution des rôles
- Suspension/activation

Gestion des formations

- Validation des nouvelles formations
- Modération du contenu
- Statistiques globales

Gestion financière

- Transactions
- Commissions
- Remboursements
- Rapports fiscaux

Configuration plateforme

- Paramètres généraux
- Gateways de paiement
- Templates d'emails
- Traductions

Analytics globales

- KPIs (utilisateurs, revenus, courses)
- Graphiques de tendances
- Rapports personnalisés

Roadmap de Développement

Phase 1: Fondations (Semaines 1-2)

- [] Configuration du projet Laravel
- [] Structure de la base de données
- [] Système d'authentification (Sanctum)
- [] Gestion des rôles et permissions (Spatie Permission)
- [] API Users (CRUD)
- [] Middleware de permissions
- [] Tests unitaires auth

Phase 2: Module Formations (Semaines 3-4)

- [] Modèles: Course, Chapter, Lesson

- [] API CRUD formations complète
- [] Relations et contraintes
- [] Upload d'images (cover)
- [] Configuration commerciale
- [] Configuration d'accès
- [] Système de slug unique
- [] Tests fonctionnels

Phase 3: Gestion Médias (Semaine 5)

- [] Service d'upload de fichiers
- [] Intégration Vimeo API
- [] Intégration YouTube API
- [] Génération de thumbnails
- [] Stockage S3/local
- [] Protection des URLs
- [] Tests d'intégration

Phase 4: Module Tests (Semaines 6-7)

- [] Modèles: Test, Question, Answer, Submission
- [] Types de questions (QCM, texte, etc.)
- [] Configuration des tests
- [] Soumission et évaluation
- [] Notation automatique/manuelle
- [] Historique des tentatives
- [] Certificats
- [] Tests complets

Phase 5: Inscriptions & Progression (Semaine 8)

- [] Système d'inscription aux cours
- [] Tracking de progression
- [] Marquage des leçons complétées
- [] Calcul de progression
- [] Notifications
- [] Tests

Phase 6: Paiements (Semaines 9-10)

- [] Architecture Payment Gateway
- [] Intégration Stripe
- [] Intégration PayPal
- [] Webhooks et callbacks
- [] Génération de factures PDF
- [] Gestion des remboursements
- [] Codes promo
- [] Tests de paiement

Phase 7: Dashboards (Semaines 11-12)

- Dashboard Étudiant
 - Vue d'ensemble
 - Mes formations
 - Mes achats
 - Certificats
- Dashboard Instructeur
 - Gestion formations
 - Analytics
 - Revenus
 - Évaluation tests
- Dashboard Admin
 - Gestion utilisateurs
 - Gestion formations
 - Analytics globales
 - Configuration

Phase 8: Fonctionnalités Avancées (Semaines 13-14)

- Système de notifications (email, in-app)
- Recherche et filtres avancés
- Système de notes et avis
- Wishlist
- Recommandations
- Forum/Q&A
- Messagerie instructeur-étudiant

Phase 9: Optimisation & Sécurité (Semaine 15)

- Mise en cache (Redis)
- Optimisation des requêtes
- Rate limiting
- Protection CSRF
- Validation des inputs
- Logs et monitoring
- Backup automatique

Phase 10: Documentation & Déploiement (Semaine 16)

- Documentation API (Swagger/Postman)
- Guide d'installation
- Guide utilisateur
- Tests de charge
- CI/CD Pipeline
- Déploiement production
- Monitoring et alertes



Structure du Projet

```
app/
  └── Console/
  └── Events/
    ├── UserRegistered.php
    ├── CoursePublished.php
    ├── TestSubmitted.php
    └── PaymentCompleted.php
  └── Exceptions/
  └── Http/
    └── Controllers/
      └── Api/
        └── Auth/
          ├── LoginController.php
          ├── RegisterController.php
          └── ForgotPasswordController.php
        └── CourseController.php
        └── ChapterController.php
        └── LessonController.php
        └── TestController.php
        └── EnrollmentController.php
        └── PaymentController.php
        └── DashboardController.php
        └── MediaController.php
      └── Middleware/
        ├── CheckRole.php
        ├── CheckPermission.php
        └── CheckCourseAccess.php
    └── Requests/
      ├── CourseStoreRequest.php
      └── TestSubmissionRequest.php
    └── ...
  └── Listeners/
  └── Models/
    ├── User.php
    ├── Role.php
    ├── Permission.php
    ├── Course.php
    ├── Chapter.php
    ├── Lesson.php
    ├── Resource.php
    ├── Test.php
    ├── Question.php
    ├── Answer.php
    └── TestSubmission.php
```

```
    └── Enrollment.php
    └── Progress.php
    └── Payment.php
    └── Certificate.php
    └── Media.php
    └── Notifications/
        └── WelcomeNotification.php
        └── CourseEnrolled.php
        └── TestGraded.php
        └── PaymentReceived.php
    └── Policies/
        └── CoursePolicy.php
        └── TestPolicy.php
        └── ...
    └── Providers/
    └── Repositories/
        └── CourseRepository.php
        └── TestRepository.php
        └── ...
    └── Services/
        └── AuthService.php
        └── CourseService.php
        └── MediaService.php
        └── PaymentService.php
            └── StripePaymentGateway.php
            └── PayPalPaymentGateway.php
            └── PaymentGatewayInterface.php
        └── TestEvaluationService.php
        └── CertificateService.php
        └── NotificationService.php
    └── Traits/
        └── HasPermissions.php
        └── Searchable.php
```

```
database/
    └── factories/
    └── migrations/
    └── seeders/
        └── RoleSeeder.php
        └── PermissionSeeder.php
        └── DatabaseSeeder.php
    └── schema/
```

```
routes/
    └── api.php
    └── web.php
    └── channels.php
```

```
tests/
└── Feature/
    ├── Auth/
    ├── Course/
    ├── Test/
    └── Payment/
└── Unit/
    ├── Services/
    └── Models/
```

Sécurité

- Validation stricte des inputs
 - Protection CSRF
 - Rate limiting sur les endpoints sensibles
 - Chiffrement des données sensibles
 - Authentification à deux facteurs
 - Logs d'audit
 - Protection contre les injections SQL
 - Sanitization des fichiers uploadés
 - HTTPS obligatoire en production
 - Gestion sécurisée des tokens
-

Métriques de Performance

- Temps de réponse API: < 200ms (95e percentile)
 - Upload de vidéos: Support jusqu'à 2GB
 - Concurrence: 1000+ utilisateurs simultanés
 - Disponibilité: 99.9%
 - Backup: Quotidien automatique
-

Internationalisation

- Support multi-langues (FR, EN, ES)
 - Traductions des emails
 - Formats de date/heure localisés
 - Devises multiples
-

Support & Maintenance

- Logs centralisés (Laravel Telescope)
 - Monitoring (Laravel Horizon pour les queues)
 - Alertes automatiques
 - Documentation technique complète
 - Tests automatisés (>80% coverage)
-

Fonctionnalités Bonus (Phase 11+)

- [] Gamification (points, badges, leaderboard)
 - [] Webinaires en direct
 - [] Transcription automatique des vidéos
 - [] Traduction automatique
 - [] Intelligence artificielle (recommandations personnalisées)
 - [] Application mobile (React Native/Flutter)
 - [] Mode hors ligne
 - [] Intégration calendrier (Google Calendar, Outlook)
 - [] Intégration Zoom/Teams
 - [] Certificats blockchain
-



Notes Importantes

1. **Évolutivité:** Architecture conçue pour supporter des milliers de cours et millions d'utilisateurs
 2. **Modularité:** Chaque module peut être développé et testé indépendamment
 3. **Standards:** Respect des conventions Laravel et PSR
 4. **Documentation:** Code commenté et documentation API complète
 5. **Tests:** TDD encouragé pour une couverture maximale
-

Durée estimée totale: 16 semaines (4 mois) **Équipe recommandée:** 2-3 développeurs backend Laravel + 1 DevOps

Cette roadmap est flexible et peut être ajustée selon les priorités business et les ressources disponibles.