



Cahier des Charges

—

EHBLO

Contents

Liste des Abréviations.....	3
1. Objectif	4
2. Acteurs et Rôles	4
3. Fonctionnalités globales.....	4
Gestion des utilisateurs	4
Gestion des cours	4
Gestion des contenus.....	4
Gestion des abonnements.....	4
Suivi analytique :	5
Accessibilité :	5
4. Exigences techniques	5
Backend	5
Frontend	5
Stockage des contenus	5
Sécurité	5
Performance	5
Accessibilité.....	5
5. Contraintes	6
Techniques	6
Opérationnelles	6
Budgétaires	6
Délais	6
6. Livrables	6
Code source.....	6
Documentation.....	6
Base de données	6
Tests	6
Déploiement.....	7
Support.....	7
Tableau de Budget — MVP EHBLO.....	7
Glossaire pour le système EHBLO	7

Liste des Abréviations

Abréviation	Signification	Contexte
API	Interface de Programmation d'Application	Intégration de services externes (paiement, géolocalisation).
FCFA	Franc de la Communauté Financière Africaine	Monnaie utilisée pour le budget (100 000 FCFA).
MVP	Produit Minimum Viable	Version initiale du site avec fonctionnalités de base.
RGPD	Règlement Général sur la Protection des Données	Conformité légale pour la protection des données utilisateurs.
SaaS	Software as a Service	Plateformes comme Shopify ou Wix utilisées pour réduire les coûts.
SSL	Secure Sockets Layer	Protocole de sécurité pour le cryptage des données.
UI	Interface Utilisateur	Conception visuelle et interactive du site.
2FA	Authentification à Deux Facteurs	Sécurité renforcée pour l'accès utilisateur.

1. Objectif

Le système **EHBLO** est une plateforme d'apprentissage en ligne (e-learning) visant à fournir des contenus éducatifs (vidéos, PDF) à des utilisateurs, principalement des étudiants et des instructeurs. L'objectif principal est de permettre la gestion de cours, l'accès à des contenus éducatifs variés, le suivi de la progression des utilisateurs et la gestion des abonnements, tout en assurant une accessibilité partielle sans compte pour certains contenus (ex. : contenus pour enfants). Le système doit être intuitif, sécurisé et évolutif pour répondre aux besoins d'une audience diversifiée, y compris l'intégration de dialectes locaux pour une meilleure accessibilité.

2. Acteurs et Rôles

- **Étudiant** : Utilisateur principal qui consulte les cours, accède aux contenus (vidéos, PDF), suit sa progression et interagit avec la plateforme.
- **Instructeur** : Utilisateur spécialisé qui crée, met à jour et gère les cours, télécharge des contenus et consulte les statistiques de ses cours.
- **Administrateur** : Gère les utilisateurs, les catégories de cours, les abonnements et les données analytiques de la plateforme.
- **Visiteur anonyme** : Accède à certains contenus publics (ex. : contenus pour enfants) sans nécessiter de compte.
- **Système (Backend)** : Gère les interactions entre les utilisateurs, la base de données, l'authentification et les analyses.

3. Fonctionnalités globales

Gestion des utilisateurs

- Inscription et authentification (email/mot de passe).
- Gestion des profils (mise à jour des informations personnelles).
- Suivi de la progression des étudiants par cours.

Gestion des cours

- Création, modification et suppression de cours par les instructeurs.
- Organisation des cours par catégories (ex. : Technologie, Sciences).
- Support de différents types de contenus (vidéos, PDF).

Gestion des contenus

- Téléversement et suppression de fichiers (vidéos, PDF).
- Streaming de vidéos et téléchargement de PDF.
- Accessibilité sans compte pour certains contenus.

Gestion des abonnements

- Support des abonnements gratuits et premium.

- Vérification des accès aux contenus premium.

Suivi analytique :

- Enregistrement des interactions (pages vues, actions comme téléchargements).
- Génération de rapports pour les administrateurs.

Accessibilité :

- Support multilingue, incluant des dialectes locaux.
- Interface utilisateur intuitive adaptée à différents dispositifs (mobile, desktop).

4. Exigences techniques

Backend

- Framework web : Django, Laravel ou Node.js
- Base de données : MySQL (relationnelle) ou MongoDB (NoSQL) pour stocker les utilisateurs, cours, contenus et sessions.
- API REST ou GraphQL pour les interactions frontend-backend.
- Gestion des sessions avec JWT ou OAuth pour l'authentification.

Frontend

- Framework JavaScript : React, Vue.js ou HTML/CSS/JavaScript classique.
- Interface responsive pour compatibilité mobile et desktop.
- Gestion des cookies pour le suivi de navigation (conformément aux mentions légales).

Stockage des contenus

- Utilisation d'un CDN (ex. : AWS S3, Cloudflare) pour les vidéos et PDF.
- Support des fichiers volumineux (optimisation pour le streaming/téléchargement).

Sécurité

- Chiffrement des mots de passe (ex. : bcrypt).
- Protection contre les attaques XSS, CSRF et injections SQL.
- Gestion des autorisations pour restreindre l'accès (ex. : contenus premium).

Performance

- Temps de chargement des pages < 2 secondes.
- Capacité à gérer 10 000 utilisateurs simultanés (évolutif).

Accessibilité

- Conformité aux normes WCAG 2.1 pour l'accessibilité.

- Support des langues et dialectes locaux via une configuration dynamique.

5. Contraintes

Techniques

- Le système doit être compatible avec les navigateurs modernes (Chrome, Firefox, Safari).
- Les contenus doivent être accessibles même avec une connexion à faible bande passante.
- Pas de dépendance à des technologies propriétaires spécifiques.

Opérationnelles

- Respect des réglementations sur les données personnelles (ex. : RGPD pour l'Europe).
- Gestion des cookies conforme aux mentions légales du site.
- Support de contenus sans compte pour des cas spécifiques (ex. : enfants).

Budgétaires

- Optimisation des coûts d'hébergement (ex. : utilisation de solutions cloud scalables).
- Minimisation des dépendances à des services tiers coûteux.

Délais

- Développement et déploiement de la version initiale dans un délai de 6 mois (à ajuster selon les ressources).

6. Livrables

Code source

- Backend (API, gestion de la base de données, logique métier).
- Frontend (interface utilisateur responsive).

Documentation

- Documentation technique (architecture, classes, méthodes, API).
- Guide utilisateur pour les étudiants, instructeurs et administrateurs.

Base de données

- Schéma de base de données (relationnelle ou NoSQL).
- Scripts d'initialisation pour les tables/collections.

Tests

- Tests unitaires pour les classes principales (Utilisateur, Cours, Contenu).

- Tests d'intégration pour les API.
- Tests de charge pour évaluer la performance.

Déploiement

- Environnement de production configuré (serveur, CDN, base de données).
- Pipeline CI/CD pour les mises à jour continues.

Support

- Plan de maintenance pour les 12 premiers mois.
- Formation pour les administrateurs de la plateforme.

Tableau de Budget — MVP EHBLO

Nom du produit/service	Prix en FCFA	Fonction
Hébergement Annuel (Supabase + Domaine)	20 000 FCFA/an	Hébergement cloud et nom de domaine.
Connexion Internet Développeur	10 000 FCFA/mois	Connexion pour dev et tests.
AI (ChatGPT+, support AI projet)	10 000 FCFA/mois	Assistance AI pour docs, code et automatisations.
Total mensuel	20 000 FCFA	(Connexion + AI)
Total initial (Hébergement)	20 000 FCFA	(Hébergement annuel)

Glossaire pour le système EHBLO

Ce glossaire définit les termes clés utilisés dans le cahier des charges du système **EHBLO**, une plateforme d'apprentissage en ligne, pour assurer une compréhension claire des concepts techniques, fonctionnels et organisationnels.

- **Administrateur** : Utilisateur avec des privilèges élevés, responsable de la gestion des utilisateurs, des catégories de cours, des abonnements et des rapports analytiques de la plateforme.
- **API (Application Programming Interface)** : Interface logicielle permettant la communication entre le frontend et le backend, utilisant des protocoles comme REST ou GraphQL.
- **Authentification** : Processus de vérification de l'identité d'un utilisateur via email et mot de passe, souvent sécurisé par des jetons JWT ou OAuth.
- **Backend** : Partie serveur du système, gérant la logique métier, la base de données et les API. Exemples de technologies : Django, Laravel, Node.js.

- **Base de données** : Système de stockage des données, relationnel (ex. : MySQL) ou non relationnel (ex. : MongoDB), contenant les informations sur les utilisateurs, cours, contenus, etc.
- **Catégorie** : Regroupement logique de cours par sujet ou thème (ex. : Technologie, Sciences, Religion).
- **CDN (Content Delivery Network)** : Réseau de serveurs distribués pour optimiser la livraison de contenus (vidéos, PDF) aux utilisateurs avec un temps de chargement réduit.
- **Contenu** : Matériel éducatif associé à un cours, tel que des vidéos ou des fichiers PDF, accessible via streaming ou téléchargement.
- **Cours** : Unité d'apprentissage sur la plateforme, comprenant un titre, une description, un instructeur et des contenus associés.
- **CSRF (Cross-Site Request Forgery)** : Type d'attaque où un utilisateur malveillant tente d'exécuter des actions non autorisées sur un site en exploitant une session active.
- **Étudiant** : Utilisateur principal qui consulte les cours, accède aux contenus et suit sa progression dans l'apprentissage.
- **Frontend** : Interface utilisateur de la plateforme, développée avec des technologies comme React, Vue.js ou HTML/CSS/JavaScript, accessible sur navigateur ou mobile.
- **Instructeur** : Utilisateur spécialisé qui crée, gère et met à jour les cours, et peut consulter les statistiques relatives à ses cours.
- **JWT (JSON Web Token)** : Standard pour la gestion sécurisée des sessions utilisateur, utilisé pour l'authentification et l'autorisation.
- **OAuth** : Protocole d'autorisation permettant aux utilisateurs de se connecter via des services tiers (ex. : Google, Facebook) sans partager leurs identifiants.
- **PDF** : Format de fichier utilisé pour les documents téléchargeables sur la plateforme, souvent pour des supports de cours.
- **Pipeline CI/CD** : Système automatisé pour l'intégration continue (CI) et le déploiement continu (CD) du code, facilitant les mises à jour de la plateforme.
- **Premium** : Type d'abonnement payant offrant un accès exclusif à certains cours ou contenus non disponibles pour les utilisateurs gratuits.
- **Progression** : Suivi de l'avancement d'un étudiant dans un cours, mesuré par exemple en pourcentage de complétion.
- **RGPD (Règlement Général sur la Protection des Données)** : Réglementation européenne sur la protection des données personnelles, imposant des règles strictes sur la gestion des cookies et des données utilisateurs.
- **Session** : Période pendant laquelle un utilisateur est connecté à la plateforme, gérée par un jeton d'authentification avec une durée de validité.
- **Streaming** : Technologie permettant la lecture en continu de contenus vidéo sans téléchargement complet, optimisée via un CDN.
- **Visiteur anonyme** : Utilisateur non authentifié pouvant accéder à certains contenus publics (ex. : contenus pour enfants) sans compte.

- **WCAG 2.1 (Web Content Accessibility Guidelines)** : Normes internationales pour rendre le contenu web accessible, notamment aux personnes en situation de handicap.
- **XSS (Cross-Site Scripting)** : Type d'attaque où un script malveillant est injecté dans une page web pour voler des données ou compromettre la sécurité.