

# Cahier des charges

## **PLATEFORME E-LEARNING « KNOWLEDGE LEARNING »**

### **SOMMAIRE**

1. Présentation du projet
2. Analyse des besoins
3. Spécifications fonctionnelles
4. Architecture des contenus
5. Spécifications techniques
6. Identité graphique
7. Backlog produit
8. Livrables attendus

# Cahier des charges

## 1. Présentation du projet

### 1.1. Contexte

La société Knowledge, spécialisée dans l'édition de livres pédagogiques, souhaite élargir son offre en proposant une plateforme e-learning accessible à distance, sur laquelle ils pourront étudier de chez eux et en toute autonomie.

Elle a fait appel à vous pour concevoir le back-end de son site e-learning/e-commerce.

### 1.2. Objectif du projet

Créer une application web permettant :

- l'achat de cursus ou de leçons,
- la consultation des leçons achetées,
- la validation des leçons,
- l'obtention de certifications,
- la gestion des contenus et des utilisateurs via un backoffice.

### 1.3. Mission confiée

Le projet consiste à développer :

- le back-end (Symfony),
- le front-end (Angular),
- la base de données,
- les tests unitaires,
- les composants e-commerce (Stripe sandbox),
- la documentation et le support de soutenance.

# Cahier des charges

## 2. Analyse des besoins

### 2.1. Personas

Persona 1 — L'Apprenant (Client)

- Objectif : acheter des formations, progresser en autonomie.
- Motivations : accès simple, interface fluide.
- Attentes :
  - parcours d'achat intuitif
  - accès immédiat aux contenus achetés
  - suivi de progression
  - certifications claires

Persona 2 — L'Administrateur Knowledge

- Objectif : gérer les utilisateurs, les contenus, les achats.
- Motivations : interface de backoffice claire.
- Attentes :
  - pouvoir créer / modifier / supprimer des contenus
  - consulter les utilisateurs
  - superviser les achats

Persona 3 — Le Correcteur / Jury

- Objectif : tester l'application.
- Actions prioritaires :
  - inscription
  - activation de compte
  - achat d'un cursus et d'une leçon
  - validation des leçons
  - obtention d'une certification

# Cahier des charges

## 2. Analyse des besoins

### 2.2. Parcours utilisateurs

Parcours Apprenant

1. Consultation des thèmes
2. Choix d'un cursus
3. Inscription
4. Activation via mail (Mailtrap)
5. Achat (Stripe sandbox)
6. Accès aux leçons
7. Validation des leçons
8. Certification automatique

Parcours Administrateur

1. Connexion
2. Accès au backoffice
3. Gestion utilisateurs
4. Gestion contenus
5. Gestion achats

Parcours Correcteur

1. Création de compte
2. Activation
3. Achat
4. Consultation des contenus
5. Validation leçons
6. Obtention certification

# Cahier des charges

## 3. Spécifications fonctionnelles

### 3.1. Fonctionnalités Utilisateur

- Création de compte
- Activation par email
- Connexion
- Achat d'un cursus
- Achat d'une leçon
- Accès aux contenus achetés
- Validation de la leçon
- Certification automatique
- Rubrique "Mes certifications"

### 3.2. Fonctionnalités Administrateur

- Gestion utilisateurs
- Gestion thèmes
- Gestion cursus
- Gestion leçons
- Gestion achats
- Accès au backoffice sécurisé

### 3.3. Règles métier

- Un compte non activé ne peut pas acheter.
- Un achat donne accès uniquement aux contenus correspondants.
- La validation d'un cursus est automatique lorsque toutes ses leçons sont validées.
- Une certification est générée pour chaque cursus validé.

# Cahier des charges

## 4. Architecture des contenus

### 4.1. Structure des contenus

- Thème → Cursus → Leçons
- Chaque leçon contient :
  - un texte (Lorem Ipsum)
  - une vidéo fictive

### 4.2. Structuration imposée

- Thèmes : Musique, Informatique, Jardinage, Cuisine
- Chaque thème comporte un ou plusieurs cursus
- Chaque cursus contient 2 leçons

# Cahier des charges

## 5. Spécifications techniques

### 5.1. Technologies utilisées

- Back-end : Symfony
- Front-end : Angular
- BDD : MySQL
- Paiement : Stripe (sandbox)
- Emails : Mailtrap
- Hébergement : Render

### 5.2. Architecture logicielle

- Architecture MVC
- Code orienté objet
- Séparation claire des couches (contrôleurs, services, repository, entités)

### 5.3. Exigences de sécurité

- Mots de passe hashés
- Vérification du format des mots de passe
- Middleware CSRF actif
- APIs sécurisées

### 5.4. Exigences base de données

Chaque table doit intégrer :

- created\_at
- updated\_at
- created\_by
- updated\_by

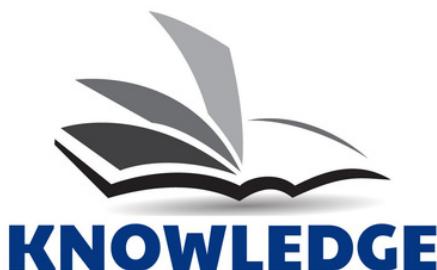
# Cahier des charges

## 6. Identité graphique

### 6.1. Police

- Comic Sans MS (imposée)

### 6.2. Logo et favicon

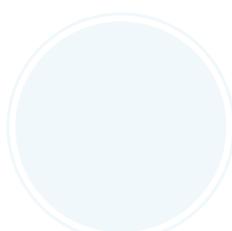


Logo

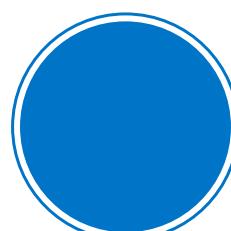


Favicon

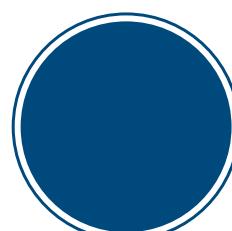
### 6.3. Palette de couleurs



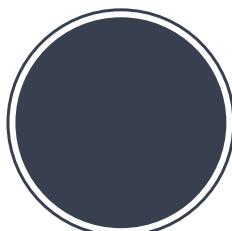
#f1f8fc



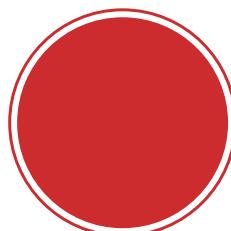
#0074c7



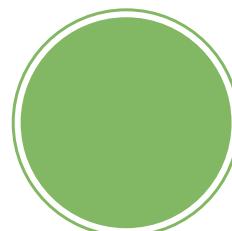
#00497c



#384050



#cd2c2e



#82b864

# Cahier des charges

## 7. Backlog produit

### 7.1. User Stories principales

#### Authentification

- US01 : Création de compte
- US02 : Activation par mail
- US03 : Connexion utilisateur

#### Catalogue

- US04 : Voir les thèmes
- US05 : Voir un cursus
- US06 : Voir une leçon

#### E-commerce

- US07 : Acheter un cursus
- US08 : Acheter une leçon

#### Progression

- US09 : Valider une leçon
- US10 : Certification automatique
- US11 : Afficher mes certifications

#### Administration

- US12 : Gestion utilisateurs
- US13 : Gestion contenus
- US14 : Gestion achats

# Cahier des charges

## 7. Backlog produit

### 7.2. Tableau Kanban

ID	Rôle	Besoin	Fonction	Critère d'acceptation
US01	Visiteur	S'inscrire	Créer compte	Email unique, mot de passe OK
US02	Utilisateur	Activer compte	Lien mail	Activation OK
US03	Utilisateur	Se connecter	Accès	JWT / Session
US04	Visiteur	Voir thèmes	Catalogue	Liste affichée
US05	Visiteur	Voir cursus	Détails	Prix + leçons
US06	Client	Voir leçon	Consultation	Contenu affiché
US07	Client	Acheter cursus	Paiement	Stripe OK
US08	Client	Acheter leçon	Paiement	Stripe OK
US09	Client	Valider leçon	Progression	Bouton valide
US10	Client	Certifier cursus	Auto-validation	Si toutes validées
US11	Client	Voir certifications	Suivi	Liste affichée
US12	Admin	Gérer users	Backoffice	CRUD
US13	Admin	Gérer contenus	Catalogue	CRUD
US14	Admin	Voir achats	Backoffice	Liste affichée

# Cahier des charges

## 8. Livrables attendus

- Dépôt GitHub public
- Application hébergée
- Modèle physique de données (MCD/MLD)
- Documentation du code
- Tests unitaires
- Supports de présentation (slides de soutenance)