

# Cahier des Charges

## Projet CINÉMA

### 1. Présentation générale du projet

Le projet CINÉMA consiste à développer une application web simple permettant de gérer une liste de films regardés.

L'application doit permettre d'ajouter un film, le modifier, le supprimer, et consulter ses détails.

Elle constitue un exercice pédagogique visant à appliquer les fondamentaux du PHP procédural, du CRUD et de la sécurité applicative.

### 2. Objectifs pédagogiques

- Mettre en pratique les notions essentielles du développement PHP procédural.
- Comprendre et implémenter un CRUD complet.
- Apprendre à protéger une application contre les failles courantes (CSRF, XSS, injections SQL, validations...).
- Structurer un mini-projet comme dans un contexte professionnel.
- Se familiariser avec la documentation, la charte graphique et les users stories.

### 3. Commençons par comprendre ce qu'est le CRUD :

#### [CRUD – Wikipédia](#)

- **C** pour **CREATE**. C'est-à-dire récupérer les données de la part du client, les traiter et les insérer dans une table de la base de données.

- **R** pour **READ**. C'est-à-dire récupérer les enregistrements d'une table de la base de données et les afficher à l'écran de l'utilisateur.
- **U** pour **UPDATE**. C'est-à-dire procéder à la modification d'un enregistrement d'une table de la base de données.
- **D** pour **DELETE**. C'est-à-dire supprimer un enregistrement d'une table de la base de données.
- **S** pour **SHOW**. C'est-à-dire, afficher les détails d'un enregistrement en particulier de la liste.

## 4. Contexte et simulation du client

Vous êtes mandatés par un ami surnommé **Mr POPCORN**.

Mr POPCORN souhaite une petite application interne lui permettant de lister les films qu'il a déjà visionnés, y ajouter une note personnelle, un commentaire et garder un historique.

Mr POPCORN veut une interface simple et agréable.

## 5. Besoins fonctionnels (Features)

Rédigés sous forme de **Users Stories** (méthode agile)

### ❖ **US1 – Ajouter un film**

En tant qu'utilisateur,  
Je veux enregistrer un film dans ma liste,  
Afin de garder une trace de mes visionnages.

### ❖ **US2 – Voir la liste des films**

En tant qu'utilisateur,  
Je veux voir tous les films enregistrés,  
Afin d'avoir une vue d'ensemble de mon répertoire.

### ❖ **US3 – Modifier un film**

En tant qu'utilisateur,  
Je veux modifier les informations d'un film,  
Afin de corriger ou améliorer les données.

#### ❖ **US4 – Supprimer un film**

En tant qu'utilisateur,  
Je veux supprimer un film de ma liste,  
Afin de garder un répertoire propre.

#### ❖ **US5 – Voir les détails d'un film**

En tant qu'utilisateur,  
Je veux consulter les détails complets d'un film,  
Afin de relire mes commentaires et voir l'historique.

## **6. Charte graphique**

#### ❖ **Palette de couleurs**

<b>Élément</b>	<b>Couleur</b>
Couleur primaire	Bleu : <b>#0D6EFD</b>
Couleurs secondaires	Noir : #000000, Blanc : #FFFFFF
Accent	Gris clair : #F8F9FA

#### ❖ **Typographies**

- Pacifico pour les titres
- Poppins pour le contenu restant

#### ❖ **Style général**

- Interface simple, moderne, lisible
- Boutons arrondis, contraste fort
- Mise en avant du titre du film et de la note

#### ❖ **Contraintes graphiques**

- Responsive minimal (mobile-friendly)
- Utilisation de Bootstrap autorisée

## 7. Spécifications techniques

### ❖ Technologies

- **FrontEnd**
  - HTML pour la mise en page
  - CSS pour la mise en forme
    - Bootstrap pour aller plus vite
  - JAVASCRIPT
- **Backend**
  - PHP
- **Base de données**
  - SQL
  - MySQL
  - PHPMYADMIN

### ❖ Structure des dossiers et fichiers

```
❖
❖
❖   |- documentation
❖   |   |- 1_projet_films_repository.pdf
❖   |   |- cahier-des-charges.docx
❖   |   |- conception.png
❖   |   |- image_to_favicon.png
❖   |   |- ~$hier-des-charges.docx
❖   |
❖   |- functions
❖   |   |- db.php
❖   |   |- helpers.php
❖   |
❖   |- partials
❖   |   |- foot.php
❖   |   |- footer.php
❖   |   |- head.php
❖   |   |- nav.php
❖   |
❖   |- public
❖   |   |- styles
❖   |   |   |- app.css
❖   |   |- .htaccess
❖   |   |- apple-touch-icon.png
❖   |   |- create.php
❖   |   |- delete.php
```

```

❖ └── edit.php
❖ └── favicon-96x96.png
❖ └── favicon.ico
❖ └── favicon.svg
❖ └── index.php
❖ └── robots.txt
❖ └── show.php
❖ └── site.webmanifest
❖ └── web-app-manifest-192x192.png
❖ └── web-app-manifest-512x512.png
❖ └── serve
❖ ``
❖ *Generated by FileTree Pro Extension*

```

## 8. Structure de la base de données

**Table : film**

<b>Champ</b>	<b>Type</b>	<b>Description</b>
id	INT, PK, AI	Identifiant unique
title	VARCHAR(255)	Titre du film
rating	DECIMAL(2,1), NULL	Note entre 0 et 5
comment	TEXT, NULL	Commentaire personnel
created_at	DATETIME	Date d'ajout
updated_at	DATETIME	Date de modification

## 9. Sécurité

L'application doit respecter les bonnes pratiques suivantes :

- Protection contre les **CSRF**
- Protection contre les **XSS**
- Usage de requêtes préparées (PDO) contre les **injections SQL**
- Validation des données côté serveur
- Nettoyage des données POST

## **10. Livrables attendus**

- Code source complet (**GitHub**)
- Base de données exportée