Cahier des charges

Contexte et objectifs

Contexte actuelle haut niveau

CineManage est un site web de gestion de cinéma qui comporte à la fois un front-office client pour la consultation de films et une interface de gestion interne sécuriser (1FA) soit un nom d'utilisateur et mot de passe ceux-là n'inclut pas de vérification de la force du mot de passe ou de désinfection des entrées de l'utilisateur. Après l'authentification l'utilisateur unique “admin” peut performer un CRUD simple dans la table Films.

Utilisateurs cible

On peut distinguer quatre catégories d’utilisateurs : * Les clients : visiteurs du site qui doivent pouvoir naviguer et consulter les films.

- les membre : un fois le client enregistrer et connecter, il peut réserver un ou des siège pour une séance. Ils doivent également pouvoir gérer leur profil.
- les administrateurs du cinéma : employés par le cinéma qui se connectent via un espace sécurisé pour gérer les contenus soit films, séances, salles, profils des membres et suivre les réservations. Ils peuvent gèrer manuellement des sièges, valider ou annuler des réservations, et consulter des statistiques de réservation.
- les développeurs tierces de la plateform : ils doivent pourvoir développer le site et testé les fonctionnalité sans avoir avoir accès aux données confidentielles.

Objectif du projet de refonte du system

Objectif fonctionnels

L’objectif principal est de simplifier ou développé ainsi qu’automatiser plusieur aspect du site actuel par exemple :

- Implémenté un processus d'ajout de film incluant le recherche de meta-donnée via un API.
- un systeme de réservation d'un ou plusieurs siège pour le visionement d'un film
- offrir aux clients et membre une interface claire, responsive et accessible, ainsi que compatible avec tous les navigateurs récents
- offir une interface avec de nouvelle fonctionalités comme :
    - un catalogue de films à jour et sans erreur.
    - offir plus de type de donnée via le catalogue de films ex: image converture, acteurs, bande d'annonce et horraire de séance.
    - une fonction de filtrage des films
    - la réservation de siège en ligne
        - avec confirmation envoyer par courriel
    - le langue du site sera le Français

  À noté, qu'une fonctionnalité de reservation par utiliseur nécésite une refonte de la base de donnée et une implémentation d'une espace ‘membre’ avec des fonctionnalitées de gestion de profile et des réglementation légal à respecté. Ainsi qu'un focus sur la sécurité acrue pour la protection des PII des membres.

  À noté, que aucune fonctionnalité payement en ligne n'a été discuté.

  À noté, qu'une fonctionnalité de recherche de film par API ou par moteur de recherche pour les clients est discuté si celle-ci est dans le prérimêtre du projet.

- offrir aux administrateurs du cinéma une interface centralisé avec des fonctionnalités administratives efficaces. Plus précisément :
    - la gestion des :
        - films faite manuellement par API
        - réservations
        - salles
        - sièges

Objectif non-fonctionnels

- Sécurisé les données des utilisateurs via le chiffrement (https/SSL) du site, le hashage avec sha-256 avant le transport des passwords et données sensible ainsi que le hashage avec sel dans pour le stockage dans la base de donnée.
- le maintien de session via cookie et jeton bearer.
- Le site doit pouvoir absorber les ‘peaks’ de visites sans lenteurs excessives.
- Ainsi qu'une base de données normalisé, index avec vue et optimisée
- Code éfficace et sécuritaire.
- Utilisation du ‘web stack LEMP’ avec docker avec possibilité intégré Laravel pour la future maintenance et scalabilité du site.
- implémentation de test unitaire ainsi d'intégration et de régression.

Contrainte et et qualité attendue

Contraintes techniques

- Utilisation de solutions open source pour limiter les coûts.
- Utilisation du ‘web stack LAMP’ avec XAAMP possible migration vers LEMP (en discudion) pour la maintenance et scalabilité du site.
- l'hébergement hybride du site cloud parti public et sur-prémise pour l’environement de développement, test et controle qualité.
- le choix du domaine pour avoir le site en ligne et accessible
- intégration de fonctionnalité pour connexion à des API distant pour les meta-données des films
- assuré la conformité légal avec GPDR

Contraintes organisationnelles (à venir)

À definir avec le client prochainement

Maintenance et soutien future (à venir)

- Définision de SLA et SLO et des horraire et procédure de support via SOP

Cloture du projet

Le projet sera considéré terminé si toutes les fonctionnalités essentielles sont implémentées et validées (accès sécurisé des utilisateurs, réservation complète, interface de gestion opérationnelle) et si le système respecte les contraintes données (sécurité, performances, adaptation mobile).

Analyse détailé des modules

Analyse des modules existants haut niveau

L'analyse du code existant des modules : * Module affichage de films : affiche les film et leurs meta-données au public. * Module authentification : Gère l'authentification, le login, gestion des sessions. * Module Films : Contient la logique pour la gestion des films avec CRUD

Analyse détailé des modules existants

Constatation générale

- le siteweb ne point vers le document root. Il utilise une constante BASE_URL "hard coded" ce qui dégrade le site si le nom du dossier est modifié.
- Plusieurs fichiers PHP autonomes ferment le tag ?>, ce qui n'est pas une bonne pratique.
- Les URL affichent l'extension du fichier (ex. .php).
- Que se passe-t-il avec la session si l'utilisateur ne clique pas sur Déconnexion et revient sur Accueil ?

login.php et logout.php

- Prévoir un flag d'environnement (production / développement).
- Supprimer les commentaires inutiles. Comme appris par Olivier.
- le mot de passe
    - Aucun salage ou poivrage (hash) avant l'insertion dans la base de données.
    - Le mot de passe est stocké en clair dans la base.
- il n'a pas de module de création/inscription pour les utilisateurs
- Les noms de tables et de colonnes sont en français, tandis que le code est en anglais ceci est incohérence.
- Erreur : "Header already sent" lors d'une connexion avec des identifiants valides.
- Les entrées utilisateur (nom d'utilisateur, mot de passe) ne sont pas désinfectées
- Le champ username est un simple champ texte sans validation.
- Faut-il vérifier si une session existe avant de la détruire ?
- login.php ligne 21: le code et la DB ne valide pas si le nom existe deja donc si il y a plusieurs utilisateur avec le même nom_utilisateur leur mot de passe sont tous valide pour cette utilisateur.

db_connect.php

- L'utilisation d'un bloc try/catch serait plus appropriée pour gérer les erreurs.
- Si la base de données est mal configurée, aucune gestion d'erreur n'est faite.

index.php, header.php et footer.php

- Serait-il préférable de créer une vue dédiée pour alléger le trafic et le traitement de la base de données, surtout pour la page d'accueil ?

config.php

- La constante BASE_URL est définie en dur, ce qui cause des problèmes si le nom du dossier change.
- Aucun mot de passe défini pour la base de données.
- Les variables sont en clair, accessibles si quelqu'un obtient l'accès au backend.

add_film.php

- Les données sont saisies manuellement uniquement, sans automatisation ni contrôle.

cinemanage_db.sql

- La table administrateurs est très basique et ne permet pas d'assigner un jeton à l'utilisateur ou un délais de connection ou autre données.
- la table films accepte tout les valeurs. aucune valeur prédefinis. Aussi, aucune validation niveau du code
- Aucun lien entre les tables
- Aucune clé étrangère
- Aucune normalisation
- Aucune gestion d'utilisateurs SQL
- Pas de contrainte UNIQUE sur les nom_utilisateur.
- Pas de contrainte UNIQUE sur les titres de films.
- l'utilisateur admin est hard codé dans la DB

Limites potentielles du système actuel

Les points faibles probables sont : * Interfaces utilisateur (UI/UX) : * l'interface actuelle n'est pas responsive et l'expérience sur mobile sera dégradé * L'ergonomie : le site ne prend pas en charge le role de utilisateur donc il affiche un lien admin en tout temps ceci peut confondre l'utilisateur. * L'absence d'accessibilité et précis en charge des animation du site et thême alternatifs.

- Base de données : simplise, sans normalisation ni indexation, ce qui limite les performances ou risque des duplications et incohérences.

- Architecture du code : l'application est codé de façon procédural sans séparation eg: pas de MVC. elle sera difficile à maintenir et étendre.

- La documentation : du code est simpliste, ce qui complique le 'onboarding' de nouveaux développeurs.

- Sécurité et légal :

    - l'architecture actuelle n'emploie pas de chiffrement ou hashage pour les mots de passe.
    - authentification simple sans autorisation.

Suggestions d’amélioration et modules supplémentaires

Pour répondre aux besoins, plusieurs améliorations sont requis : * Module affichage de films : ajout de filtrage et meta-données. * Module authentification : mettre la sécurité a niveau. * Module Films : automatiser la gestion des films * Refonte du code : Appliquer des principes MVC * Module de notifications : Ajout de notifications par email * Module amélioration du tableau de bord : Développer un dashboard avec visuel sur les réservations * Documentation : Créer une documentation technique pour le ‘onboarding’