

# Rapport Analyse Site

## 1 Contexte :

Ce projet est un projet personnelle lancée en début d'année en classe de TSI, à la demande de Victor Coindet.

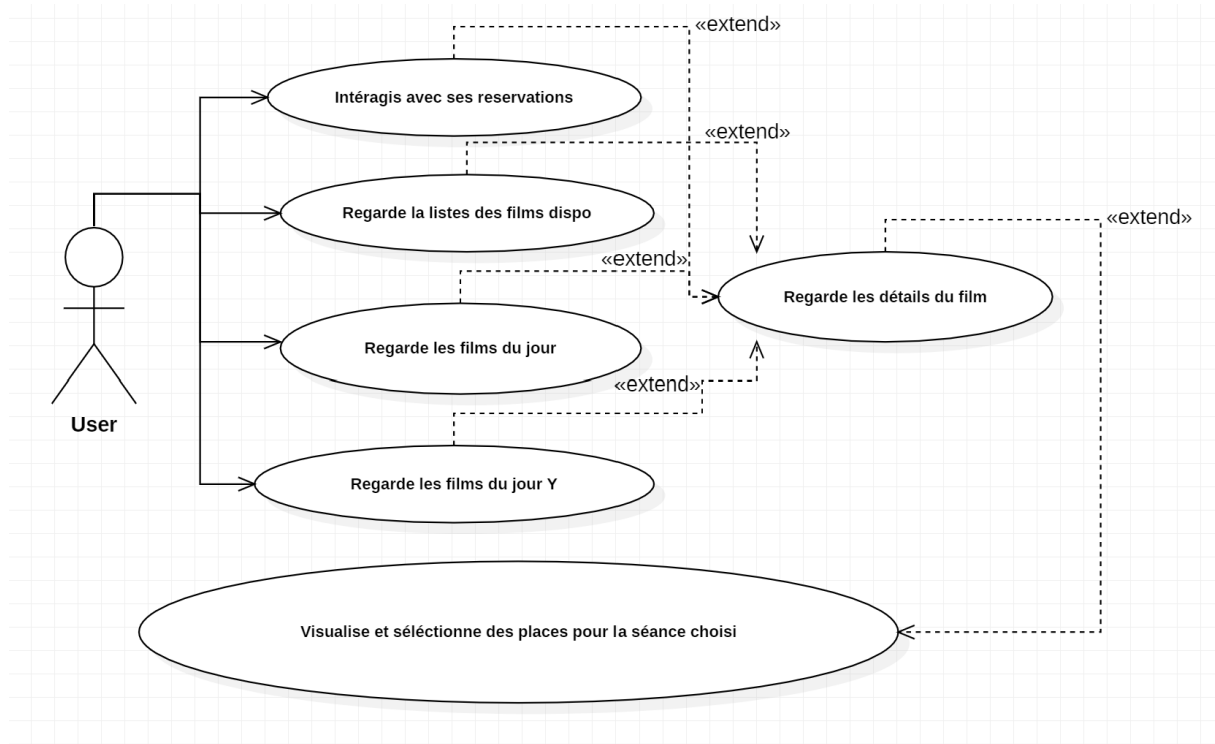
## 2 Objectif et reformulation du besoin :

Objectif : Concevoir un site web destiné à une salle de cinéma indépendante qui propose quotidiennement la diffusion d'un film du studio Ghibli et d'un film de la saga Potter.

Le site doit offrir une interface conviviale et simple, permettant de répertorier les films et leurs affiches en fonction des jours de projection. Les clients doivent pouvoir réserver leurs places tout en visualisant leur emplacement sur le plan de la salle. Une option "Covid" doit être intégrée, limitant l'occupation à une place sur deux.

Contraintes : Le site doit être susceptible d'être mis en ligne et disponible gratuitement.

Les acteurs : La plateforme ne cible pas un public spécifique ; elle doit être accessible à tous les internautes qui souhaitent réserver un film via une interface fluide et intuitive.



## 3 Analyse fonctionnelle :

Pour réaliser une interface fluide, nous avons opté pour le développement en utilisant JavaScript, en particulier avec le framework React. Nous avons intégré des API accessibles en ligne pour obtenir des informations sur les films Ghibli et Harry Potter. Le site comprendra plusieurs pages pour offrir une expérience complète aux utilisateurs :

Page de Connexion : Les utilisateurs peuvent s'authentifier sur le site à l'aide de leurs identifiants.

Page d'Accueil : Cette page présente une introduction à la salle de cinéma indépendante, ses offres, et met en évidence les films à l'affiche.

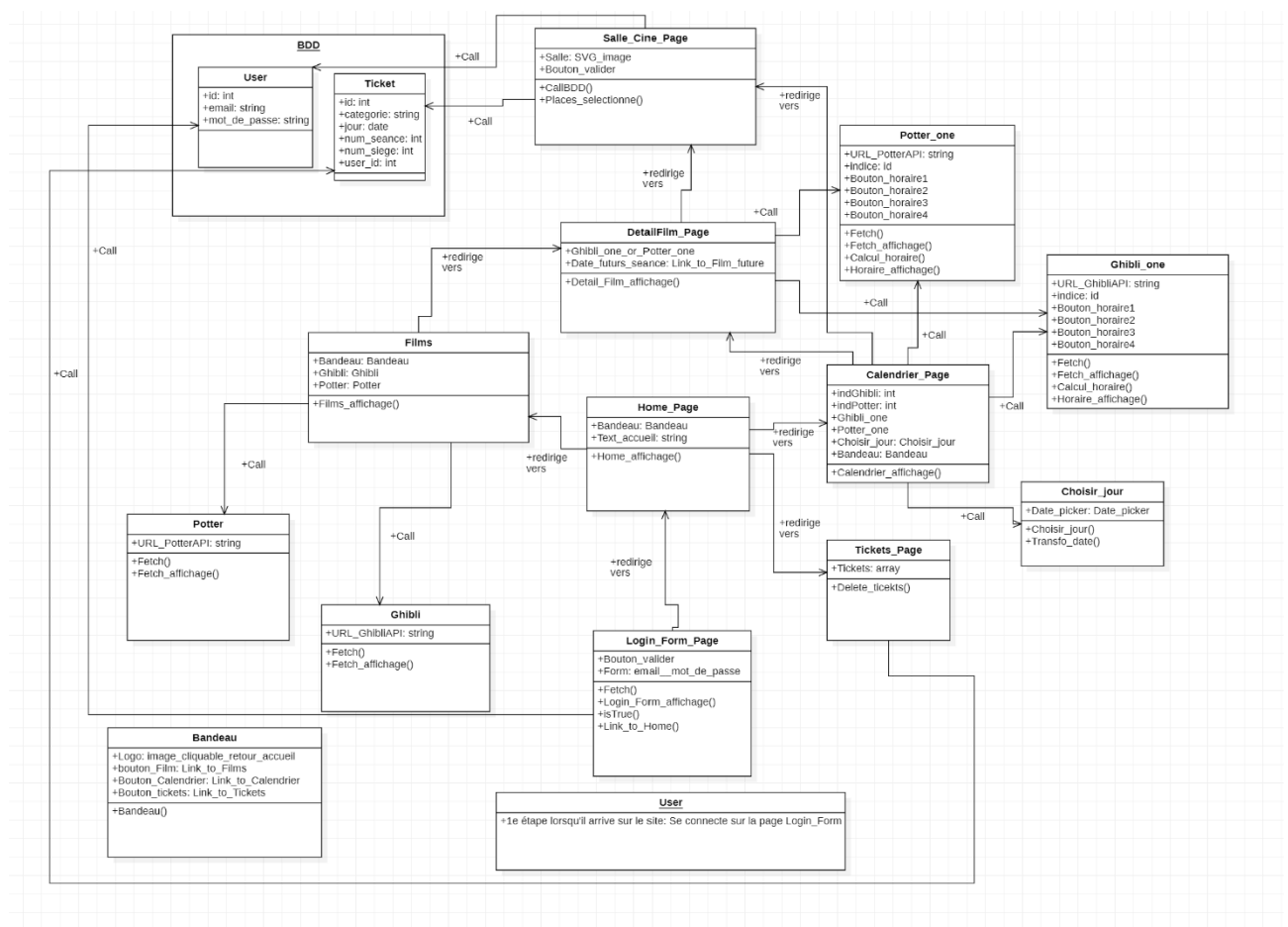
Page des Films Disponibles : Cette page liste les films proposés par la salle, avec des informations sur chaque film, comme le synopsis, les acteurs, la durée, etc.

Calendrier des Programmes : Les utilisateurs peuvent consulter un calendrier interactif pour voir quels films sont programmés pour quelles dates. Ils peuvent ainsi planifier leur visite en fonction des films qu'ils souhaitent voir.

Page de Visualisation de la Salle : Cette page permet aux utilisateurs de choisir leurs places en fonction du film choisi. Ils verront un plan de la salle avec les places disponibles et peuvent sélectionner celles qui les intéressent.

Page des Tickets : Les utilisateurs peuvent voir leurs tickets réservés, y compris les informations sur les films, la date, l'heure et les sièges sélectionnés.

En utilisant React, il est possible de créer des composants réutilisables pour maintenir une interface fluide et réactive. Les API de Ghibli et Harry Potter fourniront les données nécessaires pour afficher les films. L'intégration d'un calendrier interactif et la visualisation de la salle aideront à offrir une expérience utilisateur complète et conviviale.



Les pages :

- Login\_Form (1<sup>e</sup> page)
- Home (2<sup>e</sup> page)
- Ticket
- Films
- Calendrier
- DetailFilm en fonction du film
- Salle du film en question

Composant par page :

Page LoginForm :

- LoginForm.js : Fait le formulaire et l'appel à l'API express en « PUT »
- PageAuth : affiche le formulaire.

Page Home :

- Bandeau.js : affiche un bandeau en haut de page, « sticky », avec un logo qui redirige vers la page d'accueil, un bouton « Film » qui redirige vers la page Film et un bouton « Calendrier » qui redirige vers la page « Calendrier » avec 2 indices : un pour Potter et un pour Ghibli, qui correspond au film à afficher pour le jour en question.
- Home.js, simple page d'accueil.

Page Films:

- Bandeau.js
- Affiche les films par l'intermédiaire des composant « Potter » et Ghibli » qui affiches tous les films.
- Potter\_request.js : Fais une requête à l'API puis mets toutes les données dans un dictionnaire et les affiches, rend cliquable l'image pour aller dans le détail de ce film.
- Ghibli\_request : pareil que Potter.

Page DetailFilm :

- Bandeau.js
- Affiche la variable « film », lorsque cette page est affiché, elle aura en mémoire l'indice du film dans la liste des films du même type, et son type, on pourra alors appeler Ghibli\_one ou Potter\_one pour afficher le film seul.
- Ghibli\_one / Potter\_one : affiche le film et des horaire, la première à 15h, et ensuite les 3 autres sont affiché en fonction de la durée du film.
- Les horaires rediriges vers la page « Salle du film »

Page Calendrier :

- Calendrier.js : Affiche les Film du jour : Calcule la date du jour et affiche les films correspondant en faisant un modulo sur la liste des films respectifs.
- Catest.js : Dans le diagramme de classes : « Choisir\_jour » lorsque l'on clique sur le calendrier pour choisir un jour, Catest.js s'affiche, un calendrier qui fera appel à Film du jour avec l'indice du jour choisi.
-

Page Salle ciné :

- Fais appel à la bdd pour le film en question, regarde si des places sont déjà prises, après cela, montre le svg en fonction des places déjà prises ou non. User peut cliquer sur les places, et ensuite valider ses places. Possède ensuite ses places et peut les annuler.

Page Ticket :

- Affiche les tickets possédés par le user avec possibilité de les supprimer. En faisant appel à la bdd.

## 4 Etude technique

Pour l'étude technique, j'ai décidé d'explorer une approche innovante qui pourrait avoir des applications futures. C'est pourquoi j'ai opté pour l'utilisation de React avec JavaScript pour le développement de l'interface utilisateur. En ce qui concerne la gestion des données, j'ai également choisi une approche novatrice en utilisant un serveur Express pour établir la communication en JavaScript avec une base de données gérée par PHPMyAdmin.

## 5 Réalisation et suivi de projet

Risques : Les risques étaient plutôt limités. Les principaux concernaient les scénarios où je pourrais rencontrer des obstacles majeurs, notamment en cas d'architecture de base incohérente. Pour atténuer ces risques, j'ai pris soin de concevoir des diagrammes avant de me lancer dans le projet, tout en veillant constamment à ce que les interactions entre les composants ne posent pas de problèmes au niveau de l'architecture globale.

Planning : Le planning s'est élaboré progressivement en fonction des grandes fonctionnalités du site. Tout d'abord, j'ai consacré du temps à la découverte de React et à la compréhension des composants. Ensuite, j'ai abordé la création des pages une par une, en veillant à afficher les composants appropriés. J'ai ensuite travaillé sur la création de la base de données, l'intégration des appels aux API externes, et enfin la mise en place des routes entre les pages. Enfin, j'ai ajouté la gestion des images SVG en relation avec la base de données et la gestion de session utilisateur. J'ai trouvé que l'utilisation de React et des composants était très intuitive et permettait une structuration naturelle du code.