Rapport de projet de web

Radouane ELBISSIS,Imad BOUHOU,Deng YUFENG,Bihe DONG,Haowen WU

12 mai 2019

Table des matières

Chapitre 1

Présentation du projet

1.1 L'objectif du projet

CineEvry :
est un site web d'une nouvelle salle de cinéma à Evry (CineEvry) , ce site permet aux clients apés la création d'un nouveau compte de consulter les différents informations et les acutalités des prochains films ,des voir les prochains plannings , de réserver des places en ligne et aussi réserver la salle principal. CineEvry permet aux administrateurs de gérer les films,les réservations,les clients et les planings et de présenter les meilleurs films , des evenements , de proposer des offres avec un prix de place différents selon des catégories à savoir cat1,cat2,cat3,cat4,cat5 et cat6 .

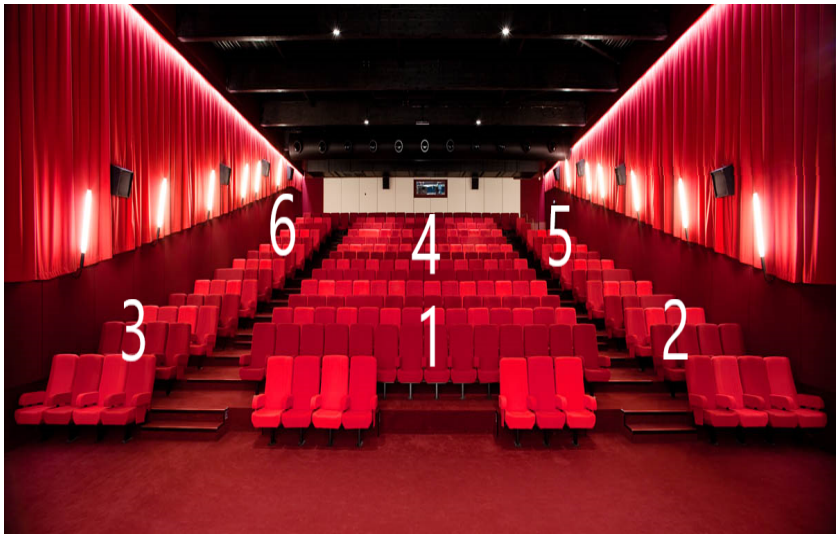

FIGURE 1.1 – l'organisation des fauteuils de CineEvry

Ces fonctionnalités ont objectif de gérer (les problématiques) :
pour les clients

— déplacement des clients le déplacement des clients vers la salle du cinéma pour obtenir des informations sur les films,les horraires,les tarifs,le plan d'accès...

— location de salles La location de salles pour organiser les manifestations privées .
— réservation des places La réservation des places et s'assurer d'avoir des places pour un film donné à une séance donnée (passer en caisse ou aux bornes = perte du temps).

pour l'entreprise

— Faciliter l'accès Faciliter l'accès des clients à la salle ..

— solution economique diminuer le nombre d'employés .

1.2 Le diagramme de cas d'utilisation et Le diagramme de classes

— Le diagramme de classes
Le diagramme de classes est un schéma utilisé en génie logiciel pour présenter les classes et les

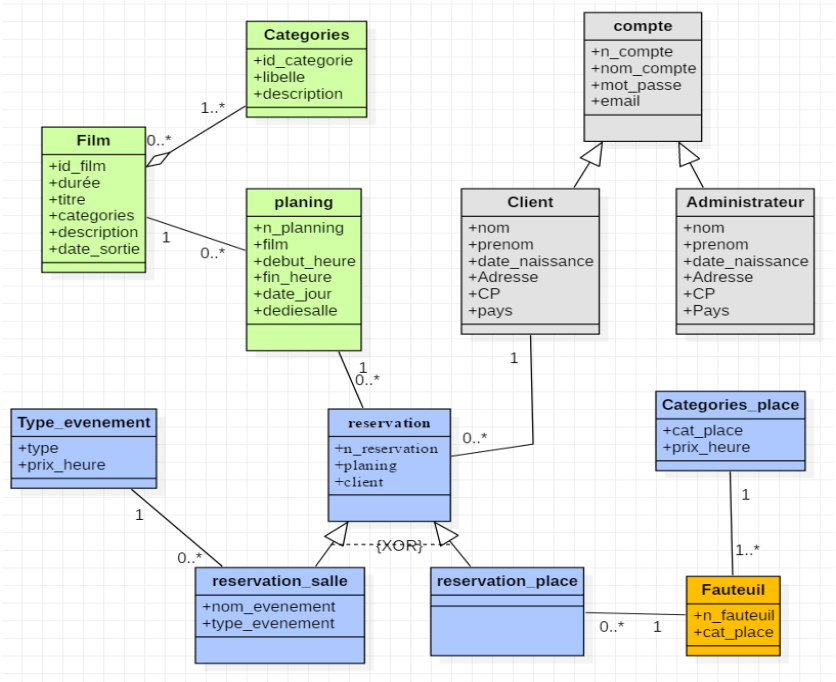

FIGURE 1.2 – diagramme de classes

interfaces des systèmes ainsi que les différentes relations entre celles-ci.
Ce diagramme fait partie de la partie statique d'UML car il fait abstraction des aspects temporels et dynamiques.
Puisque on travaille avec l'architecture MVC en Programmation Orientée Objet (POO) :
chaque table de ce diagramme sera une classe qui présente les différents attributs des tables de notre base de donnée

— Le diagramme de cas d'utilisation
Les diagrammes de cas d'utilisation sont des diagrammes UML utilisés pour donner une vision globale du comportement fonctionnel d'un système logiciel .
il est utilisé pour illustrer les fonctionnalités de site par rapport aux clients et aux administrateurs.

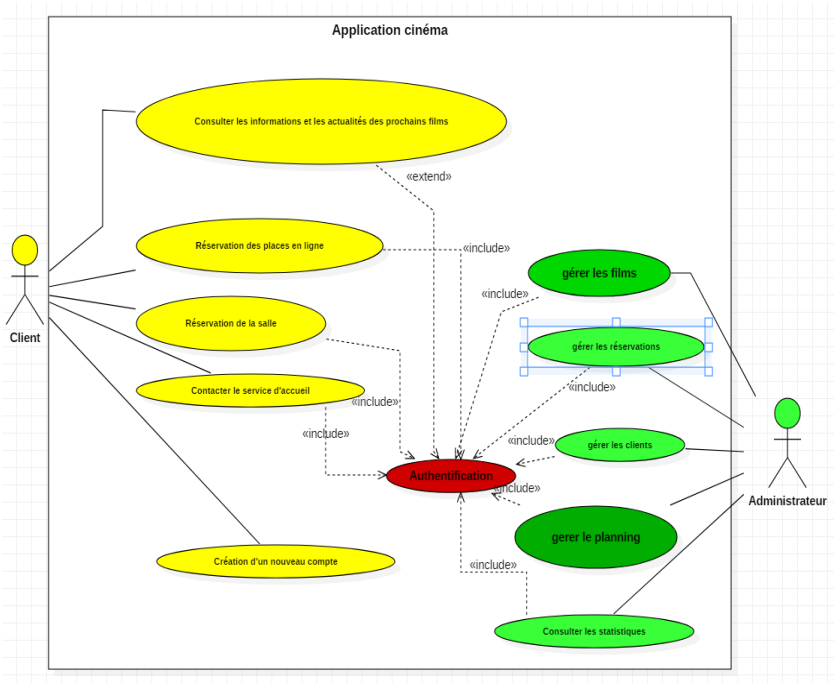

FIGURE 1.3 – diagramme de cas d'utilisation

Chapitre 2

le Modèle MVC

2.1 les modèles

2.1.1 Définition de modèle

Un modèle (Model) contient les données à afficher.

2.1.2 l'utilisation de modéle dans notre projet

on a utilisé un couple de deux classe pour chaque table de la base de données : 1-le premier élément du couple est la classe qui représente les attributs de la table et aussi les accesseurs et mutateurs (puisque les attributs sont en mode privé) 2-le deuxieme élément du couple est la classe qui contient les différents méthodes qui interviennent dans les controlleurs , par exemple (ajouter,suprimmer,modifier et récuperer les différents attributs du premier élément).

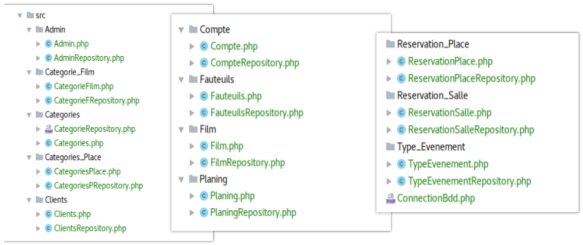

FIGURE 2.1 – les différents fichiers modèles

2.2 les vues

2.2.1 Définition de vue

Une vue (View) contient la présentation de l'interface graphique.

2.2.2 l'utilisation des vues dans notre projet

on a utilisé plusieurs fichiers php qui correspondants aux besoin des différents méthodes du contrôleurs , ces fichiers retourne des codes htmls (des formulaires,des tableaux...) qui seront stocké dans une variable ($body) du routeur , cette dernière sera utilisé dans la vue principale qui contient un contenu fixe (navbar,footer..) et des variable.

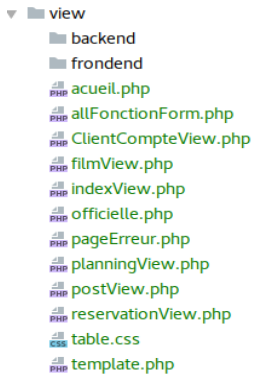

FIGURE 2.2 – les différents fichiers modèles

2.3 les contrôleurs

2.3.1 Définition des contrôleurs

Un contrôleur (Controller) contient la logique concernant les actions effectuées par l'utilisateur.

2.3.2 l'utilisation des contrôleurs

on a utilisé deux contrôleurs principals , un qui se charge de les méthodes demandé par le client et l'autre se charge de les méthodes demandé par les administrateurs ,c'est dans ces contrôleurs ou

FIGURE 2.3 – les différents fichiers modèles

2.4 la page principale officelle.php selon les différents utilisateurs

elle s'affiche pour chaque page de notre site mais par contre elle ne s'affiche pas de la même manière pour tout le monde

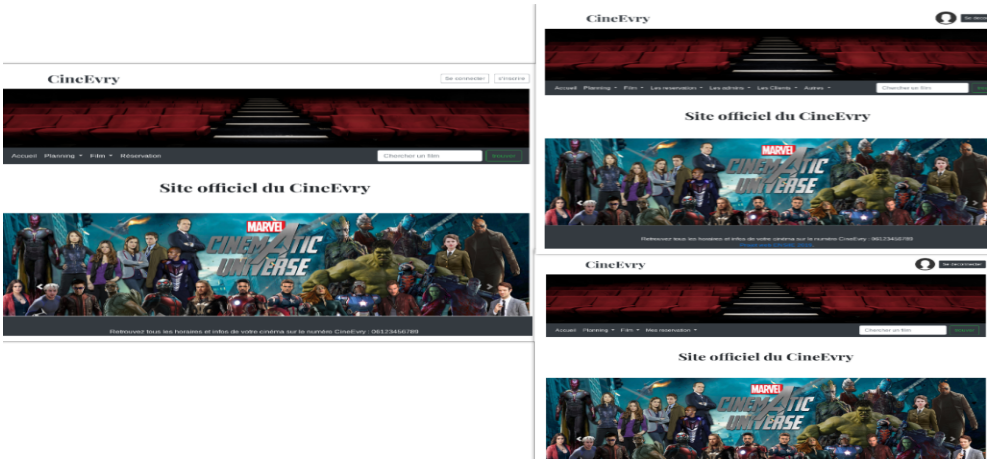

FIGURE 2.4 – la page principale

2.5 le routeur

notre routeur (index.php) est le premier fichier qu'on appelle sur le site CineEvry . Il va se charger d'appeler le contrôleur avec ses différents méthode selon l'URL .

Chapitre 3

la programmation orientée objet

3.1 Les fichiers de la partie modèle

Ces fichiers contients généralement un constructeur (methode _construct) qui prend en arguments un variable de type PDO , ce varaible permet de crée un connexion avec la base de données entre la classe et la base .
les autres méthodes on pour objectif d'ajouter,suprimmer,modifier et de récupérer des données grace à les différents requettes SQL et aussi grace à la variable $connection

Exemple :

```php
<?php

namespace Planing;

class PlaningRepository
{
    private $connection;
    public function __construct(\PDO $connection){...}
    public function add(Planing $planing){...}
    public function delete($planing){...}
    public function update(Planing $planing){...}
    //voir planing selon date;
    //voir planing selon film
    public function showPlaningOfDate($date){...}
    public function showPlaningOfWeek($date,$date_f){...}
    public function showPlaningOfFilm($film){...}

    public function showPlaningById($id){...}

    public function showAllPlaning(){...}
    public function showAllPlaningSalle($date){...}
}
```

FIGURE 3.1 – exemple de modèle

3.2 Les fichiers de la partie vue

Ces fichiers contients géneralement des méthodes (ou fonctions) qui contiennent du code HTML , ce code il sera par la suite manipuler par les contôleur **Exemple** :

FIGURE 3.2 – exemple de Vue

3.3 Les fichiers de la partie contrôleur

les deux contrôleurs de notre projets sont UserController qui s'occupe des méthodes coté clients , c'est à dire tout les actions demandé par un client , par contre , le contrôleur ChefController a pour objectif de gérer les demandes d'un administrateur.
les contrôleurs seront appelé par le routeur [index.php] ,ils récuperent les données grâce à les méthodes des modèles puis ils passent ces données aux vues. **Exemple** :

FIGURE 3.3 – exemple de contrôleur

Chapitre 4

Conclusion

généralement le projet répond à la majorité des problèmatiques . par contre on n'a pas pu réalisé des petites méthodes du contrôleur des administrateurs (ChefController) à cause d'une mauvaise gestion du temps.