# Présentation orale

Durée: 40 min

# Présentation orale: 30 min

## Présentation de l’entreprise et/ou du service et contexte du projet (cahier des charges, environnement humain et technique)

### .Contexte

* Geek et développeur web
* Une compagne passionnée de cinéma
* Exercice préparatoire à un projet personnel

### .Qu’est-ce que MoviesSpace ?

* Application locale sans Framework back-end
* Gérer les informations relatives à des films
* Application dépendante des utilisateurs

### .Autres Objectifs

* Mettre en œuvre la persistance des données (CRUD) dans une application web
* Programmation fonctionnelle -> POO

### .Responsive

* Interface adaptable aux mobiles et tablettes

### . Accessibilité

* Police sans empattement
* Navigation au clavier
* Application proposant un équivalent textuel au contenu non textuel

### .langue

* L’application est uniquement en Anglais.

Intéressant pour la pratique mais aussi pour les besoins de la certification.

### .Principales fonctionnalités

* Créer, modifier et supprimer un utilisateur
* Se connecter / déconnecter
* Ajouter, modifier et supprimer un film
* Visualiser les films présents en base de données

### .Contraintes

* Vérifier les données entrées par l’utilisateur
* Pas de doublons de nom d’utilisateur
* Accès aux fonctionnalités uniquement si connecté
* Images au format JPEG / JPG uniquement
* Mots de passe chiffrés
* Changement des paramètres du compte validés par mot de passe

### .Charte graphique

* Logo (FontAwesome)
* Couleurs (thème darkly de bootswatch)
* Typographie (Police sans empattement Montserrat)

### .Gestion de projet

* Méthodologie AGILE
* Trello
* Kanban
* Framework SCRUM

### .Suivi du code source

* Git
* Github

## Conception et codage des composants front-end et des composants back-end

### .Langages et Frameworks front-end utilisés

* HTML5
* CSS3
* Bootstrap
* Javascript
* Twig

### .Organisation général des pages

* Sections, conteneurs
* Cartes (Bootstrap)
* Gestion des marges (Bootstrap)

### .Adaptabilité de l’interface

* Bootstrap grid system

### .Composants Twig : default

* Template parent des autres templates
* Intègre l’ensemble des fichiers CSS du site
* Intègre également les scripts Javascript

### .Composants Twig : barre de navigation

* Présente sur toutes les pages
* Le menu change en fonction de l’état de l’utilisateur (connecté ou anonyme)

### .Langages et Frameworks back-end utilisés

* MySQL
* PHP

### .Structure de l’application

* FOLDERS TREE

### .Organisation de la base de données

* 3 tables
* movies : contient les films.
* account : contient les utilisateurs.
* session : enregistre les sessions.

### .baseController

* Controller de base
* Les contrôleurs réels de l’application héritent de BaseController
* Centralise les fonctionnalités partagées par les contrôleurs

### .DataBase et App

DataBase est une instance de PDO (Singleton).

App est le conteneur de l’ensemble des constantes de l’application.

### .AppController et MoviesManager

AppController gère le rendu graphique des pages relatives aux films.

MoviesManager gère les opérations en base de données concernant les films.

### .User et AuthController

User est une entité qui représente un utilisateur.

AuthController gère les opérations et le rendu des pages relatives à un utilisateur.

## Présentation des éléments les plus significatifs de l’interface de l’application

### .Workflow de l’application

* Création d’un compte
* Connexion au compte créé
* Présentation page d’accueil + responsive + menu
* Création d’un film sans jaquette
* Mise à jour de la jaquette du film
* Mise à jour des informations du film
* Suppression d’un film
* Page de visualisation d’un compte + responsive
* Changer le nom d’utilisateur (changer mot de passe)
* Déconnection

## Présentation du jeu d’essai de la fonctionnalité la plus représentative (données en entrée, données attendues, données obtenues) et analyse des écarts éventuels

### .Jeu d’essais de la fonctionnalité d’ajout de film

* Tests unitaires effectués avec PHP Unit
* Objectif : tester la fonction qui prépare les données et tester les filtres de nettoyage PHP (FILTER\_SANITARIZE)

## Présentation d’un exemple de recherche effectuée à partir de site anglophone

### .Comment effectuer un test unitaire avec PHP Unit

## Synthèse et conclusion (satisfactions et difficultés rencontrées)

### .Pourquoi pas de framework back-end ?

* Programmation fonctionnelle -> POO
* Apprendre à structurer un projet
* Mieux comprendre certains choix fait par les Frameworks
* Mettre en évidence mes points forts et mes points faible

### .Fonctionnalités à venir

### .Synthèse

### .Conclusion