Rapport de Stage

**2012**

Pierre EVERS et Maxime DESMAUTS

L3 informatique

31/05/2012



Remerciement

Nous tenons tous d’abord à remercier Monsieur Adrien Goëffon pour le temps qu’il nous a consacrée afin de mener à bien notre stage ainsi qu’a l’université d’Angers.

Sommaire

[1-Introduction 4](#_Toc325901368)

[2-Présentation du Projet 5](#_Toc325901369)

[2.1) Rappel du contexte 5](#_Toc325901370)

[2.2) Présentation du sujet 5](#_Toc325901371)

[2.3) Les besoins 5](#_Toc325901372)

[2.4) Langages utilisés 6](#_Toc325901373)

[III-Réalisation du Projet 7](#_Toc325901374)

[3.1) Les contraintes techniques 7](#_Toc325901375)

[3.2) Conception de la plateforme 7](#_Toc325901376)

[3.3) Réalisation de la base de donnée 8](#_Toc325901377)

[3.5) Mise en place de la partie Film 8](#_Toc325901378)

[3.6) Mise en place de l’inscription 10](#_Toc325901379)

[3.6) Mise en place de la gestion des utilisateurs 10](#_Toc325901380)

[3.7) Mise en place de l’ajout de film 11](#_Toc325901381)

[3.8) Mise en place de la liste de favoris 12](#_Toc325901382)

[3.9) Mise en place de la notation d’un film 13](#_Toc325901383)

[IV-Test effectués 14](#_Toc325901384)

[V-Conclusion 14](#_Toc325901385)

# 1-Introduction

Durant la période du 23 Avril 2012 au 31 Mai 2012 nous avons eu la chance de réaliser un stage au sein de l’université d’Angers afin de valider notre dernière année de licence Informatique.

Au cours de cette période, notre tâche à été de réaliser une application web collaboratif permettant l’ajout, la modification et l’affichage des données de films. Cette application doit être évolutive.

Ce document a pour objectif de présenter notre travail effectué et de décrire le déroulement de celui-ci durant toute la période de notre stage.

En vue de rendre compte de manière fidèle et analytique de ces cinq semaines de stage, il nous apparaît logique de commencer par présenter notre sujet de stage et le contexte, avant de nous focaliser sur le développement de notre application et d’en expliquer le déroulement.

# 2-Présentation du Projet

## 2.1) Rappel du contexte

Nous sommes deux étudiants de l’université d’Angers en 3ème année de Licence mention informatique. Afin de valider notre diplôme nous avons un stage à effectuer. En ce qui concerne notre stage nous sommes encadrés par Monsieur Goëffon.

## 2.2) Présentation du sujet

Il s'agit de réaliser une application / site web permettant de gérer de manière collaborative un catalogue de films. En fonction des droits des utilisateurs, ceux-ci pourront ajouter / modifier des données relatives aux films (qui pourront être récupérées automatiquement sur imdb.com par exemple), ou bien saisir une note d'appréciation. Il devra être possible ensuite de générer des listes ou des statistiques à partir de l'ensemble des données. L'application devra être simple d'utilisation et facilement évolutive.

## 2.3) Les besoins

Les objectifs de notre stage sont de concevoir et réaliser une plateforme web collaborative afin de permettre l’ajout, la suppression, la modification de film. On doit pouvoir accéder aux données au film de façon simple. On doit pouvoir rechercher des films en fonctions de leur titre. Il doit être possible de noter les films.

Un film possède différentes informations tels que le réalisateur, son année de sortie ou encore ses acteurs, un résumer et les notes attribué par les utilisateurs, il nous faut pouvoir afficher de manière simple ces données.

Il nous est alors demandé de réaliser un système d’authentification avec différents niveaux d’utilisateurs. En fonction de son niveau un utilisateur aura accès à différente partie de la plateforme. Un utilisateur non connecté ne pourra que avoir accès au donner des films sans pouvoir les noter. Les utilisateurs connectés eux sont répartie sur 3 niveaux :

-Niveau un, il est possible de noter les films

-Niveau deux, possède les fonctionnalités du niveau un avec en plus la possibilité d’ajouter, et de modifier un film.

-Niveau trois, possède également les fonctionnalités du Niveaux deux mais il peut également gérer les niveaux des autres utilisateurs, on peut donc le voir comme l’administrateur du site.

Les utilisateurs doivent pouvoir ajouter un film à leurs favoris. Ils doivent ainsi posséder une liste de leurs films favoris.

Il est également demandé de mettre en valeur un aspect communautaire. Afin de réaliser ce besoin il nous est demandé de mettre en place une fonction de groupe. Chaque utilisateur doit donc pouvoir rejoindre un groupe, ou créer un groupe. Une fois dans un groupe il doit pouvoir voir les autres membres du groupe. Ils doivent également pouvoir voir les listes des films favoris des autres membres.

Une fois ces données recueillis sur les utilisateurs et les notations des films, il nous est demandée de générer des statistiques.

Dans un dernier temps, si le temps nous le permet, notre maitre stage souhaiterais pouvoir modifier les base de donnée directement à partir de la plateforme.

Le dernier objectif est que la plateforme soit simple d’utilisation et facilement évolutives.

## 2.4) Langages utilisés

Les langages utilisés pour réaliser cette plateforme sont principalement le PHP, le JavaScript, l’Ajax ainsi que l’utilisation de frameworks tel que dojo pour réaliser les formulaires ou hightchar pour la partie statistique. C’est framework utilise du javascript, dojo permet de vérifier les données entrer par l’utilisateur dans un formulaire. Highchart génère des diagrammes à partir de donnée envoyer en norme Json. La partie PHP est codé en orienté objet. Nous avons également utilisé du css 3 pour la réalisation du design de la plateforme.

# 3-Réalisation du Projet

## Les contraintes techniques

Afin de mener à bien notre stage, nous avons dû respecter plusieurs contraintes techniques. Premièrement ils nous étaient demandés de réaliser une application web et donc de savoir maîtriser les langages de programmations web. L’application devant utiliser une large base de donnée nous avons donc d’abord du passer par une phase de conception de la base de données. Comme nous étions deux à travailler sur le projet il à également fallut que l’on mette en place un partage des tâches. Nous nous sommes également servit d’un svn via Google code.

## Conception de la plateforme

Afin de concevoir une plateforme simple d’utilisation et surtout qui soit facilement modifiable, nous avons adopté plusieurs normes.

-Modèle MVC

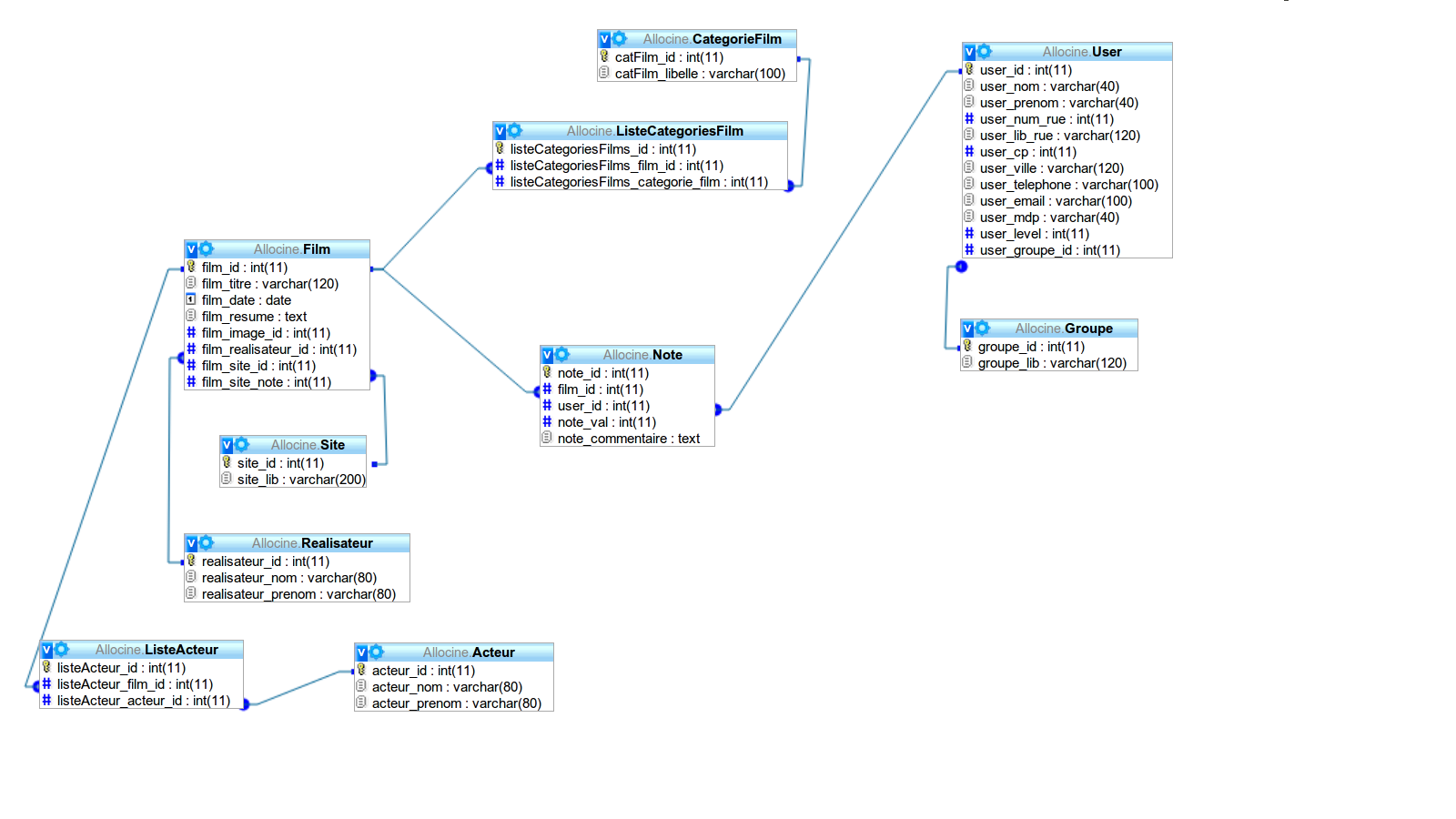
Nous avons décider de développer la plateforme sous un modèle MVC( Modèle, Vue , Controller ). Le mvc permet d’avoir une certaine rigeur dans l’écrutre du code et celui-ci permet en plus d’avoir un code structuré. Les parties qui concernent la vue sont situées dans le dossier « vue » tandis que le contrôler fait le lien entre les différentes vus et le modèle afin de gérer les données. Le modèle lui contient tous les objets de la plateforme.

-Utilisation d’un orm :Doctrine

Afin de facilitée l’accès au données et de permettre une meilleur synchronisation entre la base de donnée et le modèle nous avons décidée d’utiliser un orm. Notre choix c’est porté sur doctrine car il est simple d’utilisation et permet de faire des requêtes complètes.

## Réalisation de la base de donnée

Dans un premier temps il à fallut concevoir la base de données en fonction des besoins demandés.



Schémas BDD

## Mise en place de la partie Film

Dans un premier temps nous nous sommes concentrés sur l’accès à une liste de films et à la possibilité de voir les données du film.

* Contrainte

Il a d’abord fallut mettre en place une couche de persistance avec des pages php dao étant composé de setters et de getters. C’est méthodes permettent de récuprer facilement les données d’une table ou d’un objet du model. Le lien entre les table et le model est gérer par l’orm, donc dans ces méthodes on fait directement appelle à l’orm pour récuperer les données.

Exemple :

Si l’on veut récupérer le titre d’un film par rapport à son identifiant, on appelle la méthode getfilmTitrebyId, dans la page film\_dao.php . Cette méthode envoie une requête à l’orm afin de trouver le film qui correspond à l’identifiant. L’orm retourne ensuite un objet, ou un tableau d’objet s’il existe plusieurs films. Il suffit ensuite de récupérer les données qui nous intéresse ici se serait donc le titre du film.

* Réalisation

Nous avons réalisé une page qui gère la recherche des films demander par l’utilisateur, il existe 3 types de recherche :

La première est général par rapport à l’ajout du film, la deuxième permet de rechercher par genre, la dernière est la recherche par rapport à un titre. Il est possible d’éffectuer une recherche par titre à partir de n’importe quel endroit dans la plateforme.

C:\Users\etudiant\Desktop\rechercher.PNG

Barre de recherche par titre qui se situe dans la bannière.

L’affichage de la recherche se fait sous forme de liste on peut alors avoir déjà un aperçut des données du film, on peut voir le titre, l’année de sortie, les acteurs principaux, et l’affiche du film.



Résultat d’une recherche.

Une fois la recherche effectué il est possible de consulter la fiche d’un film en cliquant sur un bouton. On demande alors au contrôleur de nous afficher une nouvelle page fiche.php qui permet d’afficher la page du film demandé. Sur cette page toutes les informations du films sont affiché, ses informations sont récupérés via un appelle à la persistance qui appelle l’orm pour récupéré les donnée.

## Mise en place de l’inscription

* Réalisation technique

Afin de rendre l’inscription dynamique, nous avons utilisé de l’Ajax. Les informations sont envoyer à l’ORM qui créer un nouvel utilisateur dans la base de données et dans le model. Les différentes informations demandé sont le nom prénom, l’adresse postal, l’adresse mail et un mot de passe. Afin de se connecter l’utilisateur doit utiliser sont adresse mail et son mot de passe.

Une fois enregistré un utilisateur est par défaut de niveau un sur la plateforme.

## Mise en place de la gestion des utilisateurs

* Contraintes

Afin de mettre en place différent niveau d’utilisateur, il à fallut gérer des variables de sessions permettant de connaitre le niveau de chaque utilisateur. La page gestion est divisée en deux ou trois partie en fonction du niveau de l’utilisateur (la gestion de groupe, l’ajout de film et la gestion de groupe). Nous voulions que le changement de partie soit dynamique que l’utilisateur n’est pas à recharger la page à chaque changement.

* Solutions adoptées

Afin d’accéder au différentes partie et de respecter la contrainte liée au dynamisme, nous avons opté pour la réalisation de fonction javascript permettant d’afficher ou de cacher une partie en fonction du menu de la gestion.

Tous les utilisateurs enregistrés sur la plateforme ont accès à une gestion de leur compte et de la plateforme. La gestion diffère selon le niveau de l’utilisateur. Toute fois tous les utilisateurs enregistrés ont le droit de noter un film une unique fois. Ils peuvent également rejoindre des groupes.

La partie groupe permet de voir les membres du groupe, on ne peut rejoindre que un seul groupe, par contre tout les groupe sont publique et l’utilisateur peut changer très simplement de groupe. Lorsque qu’un utilisateur n’est dans aucun groupe deux choix s’offre à lui, il à la possibilité de créer un nouveau groupe ou bien de rejoindre un groupe déjà existant. Pour que la gestion de groupe soit dynamique nous avons également opter pour des fonctions javascript et d’ajax permettant de rejoindre, créer ou quitter un groupe. Une fois que l’utilisateur se situe dans un groupe il peut avoir accès à la liste des nom et prénoms des personnes étant dans le même groupe que lui. Il peut également à tout moment quitter le groupe.



Page de la gestion.

## Mise en place de l’ajout de film

Il existe différente façon d’ajouter des films sur la plateforme. On peut choisir une méthode dites manuel qui oblige l’utilisateur à rentrer toute les données manuellement ou bien en récupérant les l’id d’un film sur le site allociné ou IMDB.

* Contrainte

Lors de l’ajout d’un film il existe différentes contraintes auxquels nous avons dut nous plier. En effet il faut effectuer différentes vérifications afin de savoir si le film n’est pas déjà enregistré dans la base de données ou bien si le réalisateur ou les acteurs n’existe pas déjà dans la base de données. De plus une des contraintes est de pouvoir récupérer les films sur imdb et allocinée nous avons donc dut passer par des api qui permette de le faire. Nous avons trouvées deux api différentes une pour Allociné et une autre pour imdb. C’est deux api retourne des données de façon différents. Une des contraintes est de pouvoir maitriser ces api et comprendre leur fonctionnement.

* Solution adopté

D’un point de vue technique l’ajout de film manuellement ne présente pas beaucoup de difficulté, on récupéré les données soumise par l’utilisateur, après vérification, on les transmets à l’orm qui les insère dans la base de donnée et dans le model.

La partie insertion via l’identifiant d’un film d’une plateforme est plus complexe.  En effet les api retourne des données mais de types différentes en fonction du site. L’api de imdb retourne un objet tandis que l’api Allociné retourne du Json. Une fois les données récupérées il faut également faire les tests sur celle-ci puis ensuite on ajoute le films et ces données.

* Difficulté

La vrai difficulté à été de comprendre comment manipuler les api et ensuite de sélectionner uniquement les données qui nous intéresse. Nous avons également eux des problèmes avec les ajout de réalisateur ou d’acteurs. En effet certain films possède plusieurs réalisateurs mais la conception de notre base ne nous permettait dans avoir uniquement un seul par film. Après en avoir parlé avec notre maitre de stage nous avons de décider de sélectionner uniquement un seul réalisateur. Nous avons également rencontrer des problèmes pour l’ajout des nom des acteurs n’ayant prévue qu’un acteur possède uniquement un prénom ou nom. Mais certains acteur peuvent possède un nom composé tel que Tommy Lee Jones nous avons donc tronqué son nom.

## Mise en place de la liste de favoris

Afin de rendre la plateforme plus communautaire nous avons mit en place une liste de film favoris pour chaque utilisateur enregistré.

* Contrainte

Il à fallut modifier la base de donnée pour que chaque utilisateur puisse posséder une liste de film favoris. De plus il afin de mettre en place un aspect communautaire il à fallut mettre en place un système pour que les autres utilisateurs est également accès à la liste de chacun. L’utilisateur devait également pouvoir également gérer ça liste de favoris de manière simple.

* Solution Adopté

Lorsqu’un utilisateur est connecté et se situe sur une page d’un film, il à alors la possibilité d’ajouter ce film à sa liste de film favoris en cliquant sur un bouton « plus » situé en haut a droite. Dans le cas ou le film figure déjà dans la liste des films favoris de l’utilisateur nous avons remplacé le bouton « plus » par un bouton « moins » signifiant le retrait de ce film de la liste de favoris. Pour ce faire, il a fallut faire des testes à chaque fois qu’un utilisateur charge une fiche d’information d’un film, on regarde alors en fonction de l’id du film et l’id de l’utilisateur si le film est parmi les favoris ou non de l’utilisateur.

Afin de mettre en place un gestion des favoris et un accès simple nous avons ajouter un lien dans l’entete de la plateforme afin que l’utilisateur puisse rejoindre sa page des films favoris à tout moment. Dans la page de gestion des favoris, l’utilisateur peut voir l’affiche du film , le titre de celui-ci et il à également la possibilité de supprimer le film de la liste de ses faovris. L’orsque l’utilisateur clique sur le titre il accède directement à la fiche d’information du film

Pour voir la liste des films favoris des autres membres de la communauté nous avons mit en place dans la gestion des groupe la liste des membres de notre groupe avec leur nom, leur prénom et un lien menant direcement à la liste des favoris si celle-la existe.

.

Gestion des favoris.

## Mise en place de la notation d’un film

Pour répondre au besoin de notation du film nous avons mit en place dans l’affichage du film un système de notation par étoile. Le système de notation est compris entre z&ro et cinq , zéro étant la note la plus faible et cinq la note la plus forte.

* Contrainte

Une des contraintes pour la notation d’un film est la possibilité de noter une seul fois un film. En effet seuls les utilisateurs enregistrés doivent pouvoir noter un film. Une autre contrainte est que nous souhaitions avoir une notation dynamique. Nous avons donc dut effectuer des recherches afin de trouver une notation par étoile, une fois que l’utilisateur clique sur une étoile il doit pouvoir voir la note qu’il à attribuer et ne doit pas pouvoir revoter.

* Solution adopté

La solution adoptée pour le problème d’unique notation est d’effectuer une vérification, on exécute une requête pour savoir en fonction du film et de l’utilisateur afin de savoir si celui-ci a déjà noter le film. Dans le cas ou l’utilisateur a déjà noter le film on affiche la note qu’il à attribuer et la note général du film. Dans le cas ou l’utilisateur n’a pas noté le film on affiche les étoiles de notations. Afin de palier au problème d’effet dynamique, nous avons mit en place un script javascript et d’ajax. Le script javascript donne l’effet des étoiles lorsque l’utilisateur clique sur une étoiles, qui correspond à la notation qu’il souhaite donnée au film, une requête ajax est envoyé, cette requète retourne comme réponse la note donnée par l’utilisateur et l’affiche ensuite dans la page.

C:\Users\etudiant\Desktop\note etoile.PNG

C:\Users\etudiant\Desktop\votre note.PNG

## Mise en place des statistiques

Le système de statistique permet de voir la notation d’un film.

* Contrainte

Les contraintes de la mise en place des statistique était d’un point de vue graphique, il à d’abord fallut trouver une bibliothèque externe permettant de générer des statistiques. Dans un deuxième temps il à fallut maitriser c’est bibliothèque pouvoir convertir les données de la base de donnée pour quel soit accepté par la bibliothèque. Il fallait également gérer des statistiques par groupe.

* Solution

Les statistiques sont accessible à partir de la liste de recherche des films, un lien situer en dessous de la note du film permet d’accéder à la page des statistiques .Nous avons choisit d’utlisée hichart un frameworks javascript pour générer des diagrammes. Afin de créer les diagrammes highchart récupère les données passé par du json et génère les diagrammes correspondant. Un utlisateur peut avoir accès au maximum a 3 diagramme , par défaut seul deux diagramme sont affichée. Les diagrammes corresponde à la notation général du film , à la note de l’utilisateur pour ce film et enfin si il est dans un groupe à la note que le groupe à attribuée a ce film. Pour le dernier diagramme nous avons du récupérer tout les membres d’un groupe qui on noter ce film et ensuite en faire la moyenne.

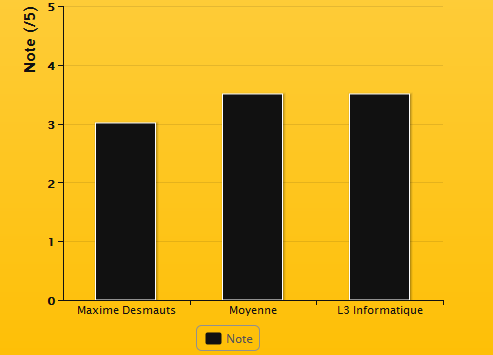


Diagramme de notation pour un film.

4-Test effectuer

Afin de régler les derniers problèmes nous avons du effectuer des séries de test. Nous avons entre autre rencontrée des problèmes lors de l’ajout de film et avec imdb ou allocinée. En effet certain acteur ou film était identifier comme null dans la base de donnée. Cela engendrait des erreurs lors de la consultation de la fiche détaillé du film.

# 4-Evlution possible de la plateforme

# 5-Conclusion