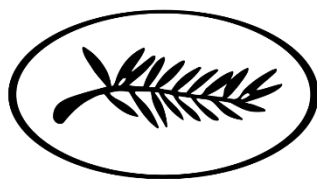


CPOA

Rapport de programmation



FESTIVAL DE CANNES

Julien Giraud - Mélodie Guérin
21/12/2018

Table des matières

Volet technique	3
État d'avancement et points d'amélioration / extensions	3
Description de l'architecture	3
Retour à l'analyse	4
Volet Gestion du projet	5
Point GIT	5
Méthode de test	5
Planning prévisionnel	5
Répartition des tâches prévisionnelle	5
Planning réel des tâches	6
Bilan personnel : Julien	7
Bilan personnel : Mélodie	7

Volet technique

État d'avancement et points d'amélioration / extensions

Notre site est terminé. Il permet de visualiser clairement les VIP, de les filtrer à l'aide d'une recherche, de les consulter en détail, de les supprimer, de les modifier et d'en créer de nouveaux. Il reste cependant un bug sur les photos des VIP : avec un serveur web locale PHP 7 (que cela soit sur Linux ou Windows) la photo est bien ajoutée au dossier photo mais sur le serveur de l'IUT (PHP 5) la photo ne s'ajoute pas. Pourtant elle est bien enregistrée dans la base de donnée. Nous pensons qu'il s'agit d'un problème de version PHP. Il resterait l'ergonomie du site à améliorer.

L'application Java est terminée mais certaines fonctionnalités pourraient être améliorées. Elle permet tout d'abord de récupérer le planning enregistré en base, de le supprimer ou de le régénérer en vérifiant les contraintes à partir des films en base.

Ensuite l'affichage du planning permet de voir pour chaque salle les séances utilisables ou utilisées. En sélectionnant une séance on peut soit supprimer la projection qui est dessus, soit y ajouter une projection (y ajouter un film car chaque séance possède un créneau horaire attribué).

Il est impossible de projeter un film dans une salle si la catégorie du film est incompatible avec celle de la salle. Cependant si le responsable modifie les projections, l'application ne vérifie pas que le nombre de projection du film est respecté, ni si les films sont soumis à des contraintes de date (dans les 9 premiers jours du festival par exemple). Il resterait les vérifications de contraintes à améliorer et les filtres sur les projections (voir par catégorie/date/séances libres) à rendre fonctionnel. La gestion automatique de l'ordre des jours de la semaine en fonction de la date de la première projection serait aussi à ajouter. Enfin il faudrait plus de souplesse sur les créneaux horaires et pouvoir afficher le planning des projections par date-heure plutôt que par salle/jour.

Description de l'architecture

Nous utilisons une architecture MVC.

En PHP nous avons donc les entités pour les classes métier, les contrôleurs, les modèles pour les accès à la base de données et les vues. Les contrôleurs des pages sont appelés par le fichier index.php à la racine du site, si le contrôleur existe il va vérifier que la vue existe avant d'effectuer les traitements puis il appelle la vue correspondante. Nous utilisons un model DAO qui possède une connexion à la base de données. Les modèles propres à chaque entité sont des extensions du modèle DAO.

En Java nous avons un package métier pour les classes métier ainsi que pour le traitement des actions des boutons de l'IHM, un package vues pour l'affichage et un package persistance pour la connexion et le DAO. Dans la persistance il y a également un package model avec les modèles correspondants aux classes métier et un package interface. Chaque modèle métier implémente les méthodes de l'interface correspondante. Ainsi nous n'avons qu'à ajouter les fonctionnalités dont nous avons besoin dans les interfaces puis nous les implémentons.

Les attributs informations enregistrées en base sont récupérées par les modèles (Java et PHP) qui les stockent dans des objets qu'ils retournent. À chaque modification de ces objets, nous enregistrons les modifications en base pour sauver le travail de l'utilisateur

Retour à l'analyse

Nous avons eu des difficultés à réaliser l'analyse bien que nous y ayons passé cinq séances. Au final les maquettes nous ont beaucoup aidé mais il y avait des incohérences dans nos diagrammes de classes et nos diagrammes de séquence / cas d'utilisation.

Lors du développement nous nous sommes rendu qu'il manquait certains éléments dans la base de donnée comme la gestion de plusieurs réalisateurs pour un même film. Nous avons donc ajouté une table et modifié certains champs. De même avec les classes en Java, certains attributs manquaient ou n'étaient pas du bon type comme le nom des salles ou les heures de projection en NUMBER à la place de VARCHAR. Pour l'IHM nous nous sommes basé sur les maquettes plutôt que sur les diagrammes de cas d'utilisation, au final nous avons bien fait car nos diagrammes avaient été réalisés à partir de ces maquettes sauf qu'ils étaient incomplets.

Les diagrammes de séquence nous ont aidé bien qu'ils ne soient pas très précis.

Volet Gestion du projet

Point GIT

GIT a été très pratique tout au long du projet. Il nous a servi à mettre en commun nos fichiers lors du travail d'analyse et lors du développement il nous a permis d'avoir un suivi de l'avancement du projet et de pouvoir revenir sur le code plusieurs fois à cause de différents problèmes. Il y aura eu un peu plus de 70 commit au total.

Méthode de test

Notre méthode de test consistait à tester toutes les fonctionnalités que nous avons implémenté. Cette méthode était très efficace pour ce projet car le nombre de fonctionnalités était très réduit il était très rapide pour nous de voir si cela fonctionnait. Nous utilisons le mode debug en PHP sur les modèles pour afficher plus de messages d'erreur (model = new model(true) pour activer le debug). Et nous regardions les valeurs dans la base de données depuis phpmyadmin pour vérifier que les traitements avaient bien été effectués. De même pour le Java, nous avons beaucoup utilisé les messages d'erreur de la console pour régler les problèmes à la source. Le dernier étant un problème de synchronisation entre la fenêtre de l'IHM et le traitement.

Planning prévisionnel

Date	Travail prévu
Avant le 14 décembre	Analyse et rédaction du rapport d'analyse
Du 14 au 20 décembre	Création de la base de données Création des classes PHP Création des classes Java Remplissage de la base de données
Pendant les vacances	Ajouter la connexion à la base de données en PHP Ajouter la connexion à la base de données en Java Ajouter les modèles d'accès aux données Avancer sur les fonctions en Java
De la reprise au 10 janvier	Implémenter les fonctionnalités en PHP Avancer sur les traitements en Java
Du 10 janvier au 16 janvier	Gérer l'affichage et l'ergonomie du PHP Implémenter l'IHM Java
Du 16 au 21 janvier	Préparer la présentation

Répartition des tâches prévisionnelle

Développeur(euse)	Tâches prévues
Julien	Gestion de la base de données et adaptation des anciens projets pour avoir une structure MVC en PHP et en Java Ajout des classes métier, modèles DAO / connexion et traitements algorithmiques
Mélodie	Ajout des contrôleurs et des vues en PHP Gestion de l'aspect graphique et responsive des vues Ajout des fenêtres et composants de l'IHM en Java Gestion de l'aspect graphique et responsive de l'IHM Java

Planning réel des tâches

Date	Développeur(euse)	Travail réalisé
Pendant les vacances	Julien	Ajout des données dans la base de données Réalisation de la connexion avec DAO en PHP Réalisation de la connexion avec DAO en Java Ajout des classes auto-générées en Java
Séance du 07/01/2019	Julien	Ajout des classes en PHP Début de l'ajout des modèles d'accès aux données en PHP
	Mélodie	Début de l'interface graphique en PHP Création contrôleur Connexion, Accueil en PHP Création vues Connexion et Accueil en PHP
Séance du 08/01/2019	Julien	Fin de l'ajout des modèles en PHP Ajout de quelques nouvelles valeurs en BD pour les tests Quelques correctifs sur le projet en Java
	Mélodie	Modification et création des vues Accueil (table VIP), VIP (liste des caractéristiques, boutons Modifier et Mupprimer), AjouterVIP (inputs et bouton Ajouter), ModifierVIP (inputs et bouton modifier), Header (menu) et Footer en PHP Modification et création des contrôleurs Accueil et VIP en PHP
Séance du 10/01/2019	Julien	Avancement dans les fonctions Java + correction de bugs Modification de la base de données (table de réalisateurs) Ajout des fonctions de recherche fonctionnelles en PHP et Java
	Mélodie	Création et modification contrôleurs Header, Footer et ModifierVIP en PHP Modification vue ModifierVIP en PHP Gestion de l'erreur de recherche de VIP non existant en PHP
Séance du 14/01/2019	Julien	Ajout des fonctions sur les projections en Java (fonctionnelles) Début de l'IHM Java Correctifs sur les fonctions du model VIP en PHP
	Mélodie	Création contrôleurs AjouterVIP, SupprimerVIP en PHP Modification et création vues AjouterVIP, SupprimerVIP en PHP
Séance du 15/01/2019	Julien	Début de la génération automatique du planning en Java Ajout du système d'affichage des projections en IHM Correctifs model VIP PHP
	Mélodie	Modification du modèle VIPDAO en PHP Modification contrôleurs AjouterVIP, SupprimerVIP et Header en PHP
Séance du 16/01/2019	Julien	Avancement sur la génération automatique du planning en Java Modification des contrôleurs en PHP
	Mélodie	Modification de Bootstrap en PHP Ajout de la fonctionnalité pour modifier/ajouter la photo d'un VIP en PHP
Séance du 21/01/2019	Julien	Java terminé : Génération automatique du planning Ajout / Suppression / Modification Améliorations de l'interface (responsive) Optimisation du code (moins d'objets, attente passive) Ajout des commentaires
	Mélodie	PHP terminé : Corrections des modèles PersonneDAO et ImageDAO Affichage / Ajout / Modification / Suppression VIP Améliorations du responsive design et du graphisme

Bilan personnel : Julien

De tous les projets que nous avons réalisés à l'IUT, celui-ci est de loin celui qui aura été le plus enrichissant. Depuis le S1 nous avons souvent travaillé en groupe et avons réalisé divers projets en Web, algorithmie, Java, IHM, Java/mysql, UML et PHP/mariaDB mais nous n'avons jamais vraiment eu l'occasion de combiner ces technologies.

Lors du développement j'ai pu d'un côté renforcer mes compétences mais aussi comprendre beaucoup mieux pourquoi nous avons vu ces technologies, notamment la SQL. En effet, lors des TP nous avions juste des questions sur des tables données dans l'énoncé mais là nous avons un cas réel. En TP je ne comprenais pas pourquoi nous devons savoir utiliser telle ou telle commande et quel était l'intérêt de nous poser des questions complexes sur des conditions avec des jointures de tables... À présent non seulement j'ai compris que cela avait une réelle utilité mais surtout j'ai réussi à le faire (il faut savoir que j'étais vraiment très mauvais en SQL au S1 et S2). Il s'agissait de l'exemple le plus pertinent mais j'ai aussi compris l'utilité de la séparation entre l'IHM et les traitements alors que je ne savais absolument pas faire d'IHM en Java avant ce projet (mon module d'IHM au S2 ne s'est pas très bien déroulé...).

Le travail avec Mélodie n'a pas été évident, surtout au début car nous n'avons pas les mêmes méthodes de travail mais nous avons su nous adapter. L'avantage était que Mélodie est très portée sur l'aspect graphique et l'ergonomie alors que j'ai une préférence pour l'algorithmie et le développement back-end. Nous étions donc plutôt équilibrés à ce niveau là.

Ayant déjà géré des petites équipes j'ai pris le rôle de chef de projet. Mon objectif était d'impliquer Mélodie au maximum dans le projet même si notre différence de niveau en développement était importante. Au début nous avons travaillé ensemble sur le PHP et j'ai aidé Mélodie à se familiariser avec la structure MVC. Elle a vite compris comment se servir de la partie back-end (classes / modèles) et j'ai pu m'impliquer davantage sur le Java. Une fois cette phase passée nous avons pu tous deux travailler ensemble très efficacement. J'ai constaté que Mélodie s'impliquait de plus en plus et était bien plus à l'aise et confiante sur son travail. Nous faisions des points à chaque séance sur notre avancement et en cas de besoin sur la partie back-end du PHP elle m'en parlait et je m'occupais d'implémenter les fonctions dont elle avait besoin. À la fin du projet elle a d'ailleurs réussi à implémenter de A à Z la gestion des photos que nous pensions ne pas faire par manque de temps. De mon côté j'ai plusieurs fois eu besoin d'elle sur l'IHM pour avoir sa vision et des idées d'améliorations. Nous étions donc faibles en nombre mais redoutables en créativité et réactivité face aux problèmes.

J'estime que ce projet est une excellente expérience et qu'il m'a beaucoup apporté tant au niveau des connaissances techniques que du travail en équipe et de la gestion de groupe.

Bilan personnel : Mélodie

Au premier abord, le thème de cette année m'a enjoué, étant intéressée par l'audiovisuel, mais les langages de programmations m'ont moins plu. En effet, je doutais de mes compétences en informatique et j'avais peur de ne rien pouvoir rendre. Lors des choix des groupes, je n'ai pas forcément eu le choix et j'ai été assignée à un binôme. Cela m'a d'autant plus stressé car je savais qu'en étant deux nous serions forcément désavantagés par rapport aux autres trinômes. Certes il y avait moins de contraintes à gérer mais la tâche de travail me paraissait tout de même plus importante.

Les 5 premières semaines d'analyses ont été laborieuses car mon binôme et moi-même ne savions pas réellement où se diriger avec les diagrammes. Les diagrammes de classes évoluaient constamment, pareillement à ceux de séquence et il a fallu tout recommencer de zéro plusieurs fois car nos esprits s'entremêlaient. Pour les IHM en revanche c'était différent, il était bien plus simple pour moi de savoir comment organiser l'espace pour optimiser au maximum les applications. Lorsque nous avons rendu le rapport d'analyse, j'étais satisfaite à moitié : les diagrammes que nous avions proposés ne me parlaient pas mais j'étais contente du résultat des IHM.

Nous avons commencé la programmation par l'application en PHP, en reprenant les bases du projet de Julien. Au départ j'ai été un peu perdue et il m'a fallu quelques explications pour comprendre certaines parties du code, mais après m'être plongée dedans j'ai trouvé un réel intérêt pour ce langage. Après avoir fait un affichage brouillon des fonctions principales (les boutons, le tableau de VIP, etc), j'ai voulu gérer le CSS, mais ce n'était pas une bonne idée. Je n'étais pas familière avec Bootstrap et c'est pourtant le Framework qui était implémenté, j'ai dû me renseigner sur comment bien l'utiliser et j'ai perdu du temps au début en voulant aller trop vite. Je me suis résignée à travailler le CSS et j'ai implémenter les fonctions pour ajouter, modifier et supprimer un VIP. Après m'être assurée qu'elles fonctionnaient, j'ai repris là où je m'étais arrêté en CSS pour avoir un rendu qui me convenait mieux. Je ne suis pas 100% satisfaite du résultat, mais je suis contente d'avoir compris la manière d'utiliser Bootstrap et ça m'a vraiment motivé à continuer ce projet jusqu'au dernier moment. Je me suis ensuite mise au défi de permettre d'ajouter une photo sur les VIP, chose que j'ai réussie et dont je suis très satisfaite.

Finalement, en PHP, je considère que j'ai réussi à implémenter tout ce qui était demandé, l'application reste encore à développer mais je suis satisfaite du résultat. Faire ce projet de CPOA m'a beaucoup apporté : il m'a réconforté quant à mes capacités et m'a permis de découvrir le PHP sous un autre angle, beaucoup plus ludique que ce à quoi je m'attendais. J'avais été déçue de la plupart des matières informatique à l'IUT, mais travailler un projet en groupe tel que celui-ci m'a permis de regagner cette curiosité que j'avais à l'arrivée à l'IUT. Le travail de groupe s'est quant à lui plutôt bien passé, avec Julien nous avons eu quelques tensions dû au fait que nous étions tout les deux stressés quant à l'avenir du projet, mais je pense que l'on a réussi à faire ce que l'on pouvait comme on le voulait.