

A dark blue vertical bar runs along the left edge of the page. A blue arrow points to the right from this bar, containing the date.

07/01/2016

Rapport de programmation

CPOA

Several thin, curved lines in dark blue and light grey originate from the bottom left corner and sweep upwards and to the right.

Pablo Gabolde, Amaury Petrini, Alexandre
Decraene
G3S3

Mots de passes nécessaires :

Vous pouvez retrouver le site web à l'adresse suivante :

<http://iutdoua-webetu.univ-lyon1.fr/~p1406314/AppliWebCPOA/billetterie/controller/index.php>

Pour vous connecter à l'espace administrateur, il faut utiliser les logs suivants :

- Login : PETRINI
- Mdp : 0000

Eléments techniques

1) Etat d'avancement

- **Pour la billetterie :**
 - *Ce qui fonctionne :*
 - Vérification du numéro de licence du client
 - Gestion des erreurs (aucun numéro saisi, numéro incorrect)
 - Achat de plusieurs places, choix des emplacements libres
 - Saisi d'un code promotionnel, et nouveau calcul de prix
 - Connexion d'un administrateur
 - Gestion des erreurs (aucun login ou mot de passe saisi, mot de passe incorrect)
 - Affichage des billets vendu, ainsi que leur prix, leur emplacement, le match associé
 - Possibilité d'ajouter ou de supprimer un billet
 - Affichage des promotions, avec leur code associé et leur pourcentage de réduction
 - Possibilité d'ajouter ou de supprimer un code promotionnel.
 - *Ce qui ne fonctionne pas :*
 - Pas d'achat plusieurs place à la fois, seulement une par une
 - Pas de promotion pour un client licencié

- **Pour le planning des matchs :**

- *Ce qui fonctionne :*

- Interface de connexion
 - Interface Management Matches d'Entrainement (cote admin) : Les matchs qui ont été réservés côté joueur sont enregistrés dans une base de données, et l'admin peut les supprimer.
 - Interface Management Matches de tournoi (cote joueur) : Les matchs qui ont été réservés (hors API) sont enregistrés dans une base de données, et l'admin peut les supprimer
 - L'admin peut aussi ajouter les joueurs dans une Jliste. (une Jliste pour les matchs en Solo et une Jliste pour les matchs en duo).

- *Ce qui ne fonctionne pas :*

- Réservation (coté joueur) : Le Terrain et le créneau horaire ne s'adaptent pas en fonction de la date choisie. Le nom de la réservation devrait être choisi dans un tableau de joueurs inscrits relié à la base de donnée qui contient les joueurs inscrits cette base existe, mais n'est pas reliée.
 - Interface Management Matches d'Entrainement (cote admin) : L'admin ne peut pas modifier la date ni le terrain.
 - Interface Management Matches de tournoi (cote joueur) : L'admin ne peut pas modifier les matchs.
 - Interface Management Planning (cote admin) : Le planning n'est pas géré.

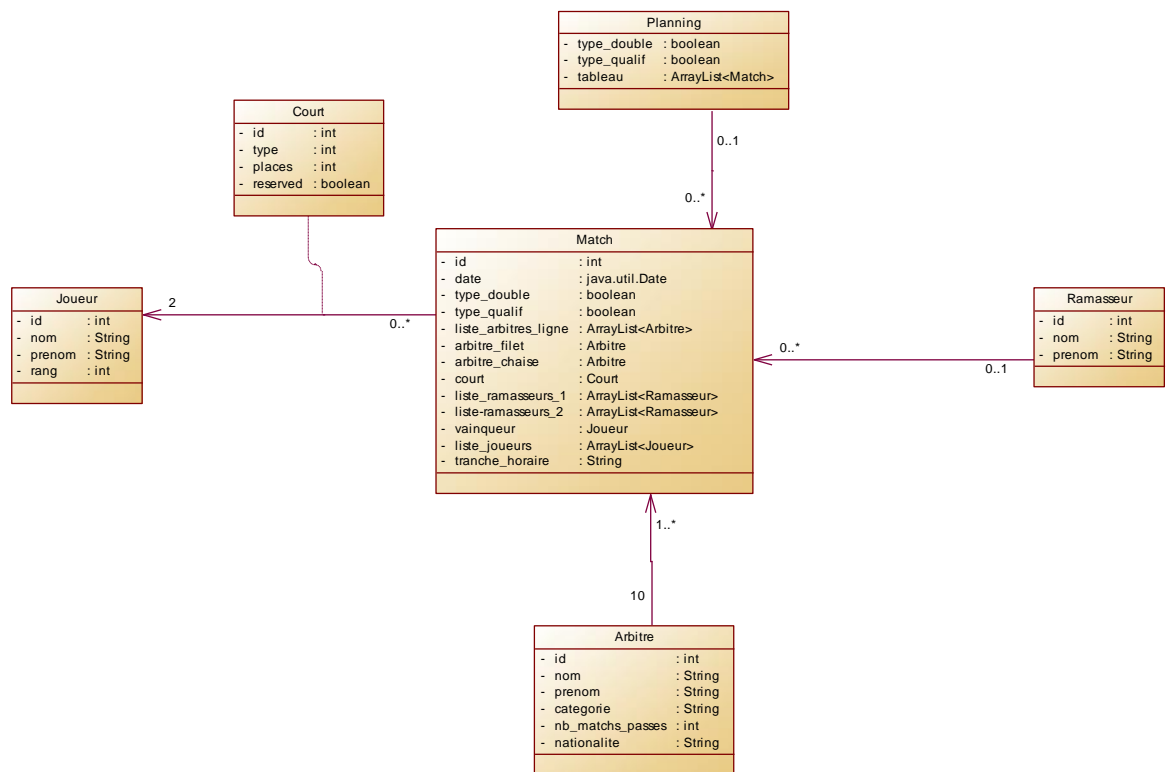
2) Description de l'architecture

Notre méthode utilisée pour coder a été le développement itératif incrémental d'UML car nous avons suivi nos diagrammes et notre IHM bien qu'il ait fallu les modifier et les repenser pendant le développement.

Ensuite, nous avons organisé (au départ) le projet (WEB et JAVA) à l'aide des couches MVC. Suite au manque de temps, nous avons délaissé la couche « Model » pour parvenir à un MVC « personnalisé ». En effet, il y avait les contrôleurs et les vues.

3) Retour à l'analyse

Modèle orienté objet	
Modèle : Projet_GPTL	
Package : Planning des matchs	
Diagramme : DiagrammeClassesPlanningDesMatches	
Auteur : p1402054	Date: 07/12/2015
Version:	



Nous avons utilisé ce diagramme de classe afin de créer une base de données (MySQL) qui soit cohérente avec notre projet. Nous avons tout de même dû modifier quelques éléments car notre diagramme était au final incomplet.

Reserver Terrain d'entrainement :

Reservation

Nom Réservation :

Date :

Terrain Disponible :

Créneau Horaire Disponible :

Dans cet exemple, on récupère les différentes dates, les terrains possibles et les créneaux horaires dans la base de données à l'aide de la classe Match.

Un autre exemple de l'interface JAVA :

Ici on a l'interface qui permet de Gérer le premier tour du tournoi, elle n'est pas fonctionnelle à 100%.

Tournament

Joueurs Inscrits

Joueurs Match Simple :

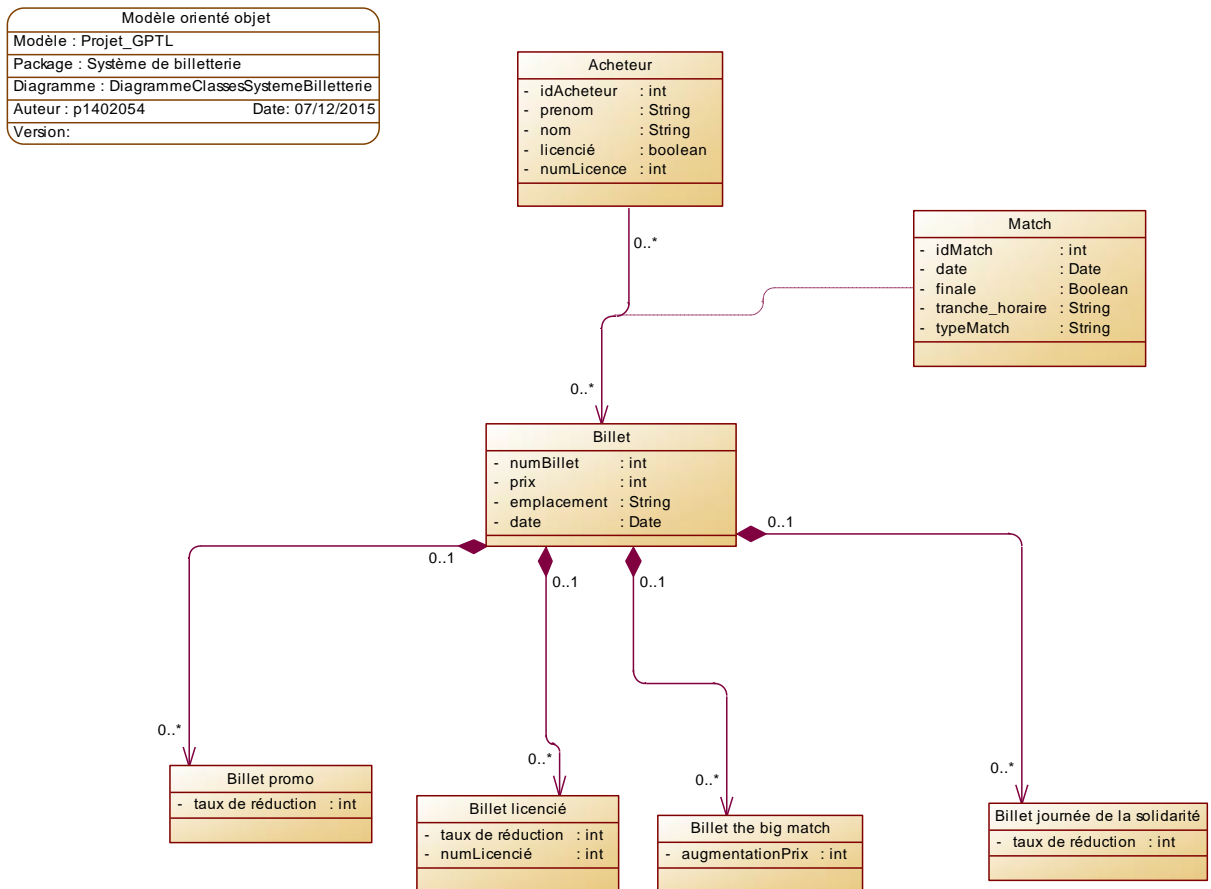
Momo PETRINI
Pablo GABOLDE
Rafael NADAL
Novak DJOKOVIC

Joueurs Match Double :

Roger FEDERER / Stanislas WAWRINKA
Jo-Wilfried TSONGA / Richard GASQUET

Joueurs Choisis

Match en Solo	Equipe 1	Equipe 2	Match en Equipe	Equipe 1	Equipe 2
1	Pablo GABOLDE	Momo PETRINI			



Pour donner un exemple pour le système de billetterie : nous avons utilisé la classe « Match » pour récupérer la liste des matchs dans la base de donnée, pour ensuite l'afficher. Cela sert à celui qui veut acheter une place de sélectionner le bon, et éventuellement de voir les détails de celui-ci :

Infos tournoi	Tableaux	BILLETTERIE	Offres VIP	Partenaires	Contacts	Administrateur
ACCUEIL						
LISTE DES MATCHS						
Joueur/Equipe 1	Joueur/Equipe 2	Date	Détails	Réserver		
Rafael NADAL	Novak DJOKOVIC	16/01/16 à 13:00	Détails	Réserver		
Momo PETRINI	Pablo GABOLDE	17/01/16 à 18:30	Détails	Réserver		
Roger FEDERER - Stanislas WAWRINKA	Jo-Wilfried TSONGA - Richard GASQUET	18/01/16 à 19:00	Détails	Réserver		

4) Points d'amélioration

- 1) Tout d'abord, pour la billetterie, on pourrait rediriger le client vers un site de paiement sécurisé une fois ses achats terminés, avec un mail de confirmation et une copie des billets.
- 2) De plus, au lieu de vérifier le numéro de licence par rapport à une base de données, on pourrait vérifier par rapport au fichier national des licences.
- 3) Au moment du choix de l'emplacement des places, on aurait pu ajouter une carte des tribunes interactive, comme sur le vrai site de l'open sud de France.
- 4) On aurait également pu ajouter des promotions pour les étudiants, avec vérification, ou pour les personnes à mobilités réduites avec un choix adéquat des places.
- 5) La génération des places automatiques, c'est-à-dire que l'administrateur n'a pas besoin de les créer tous manuellement.
- 6) La génération du premier tour du planning des matchs automatique.

5) Rétro-conception

Notre phase d'analyse n'était pas assez poussée. En effet, au fur et à mesure de l'avancement de notre projet, nous avons dû revoir nos IHM ainsi que nos diagrammes, car ceux-ci n'étaient pas adaptés. En revanche, les diagrammes de cas d'utilisation nous ont aidés pour cerner les besoins des utilisateurs.

Au niveau développement, nous avons tout d'abord eu une phase de réflexion entre nous trois afin de discerner les différentes fonctions de l'application. Ensuite, nous avons eu une phase de pratique où nous avons codé à la fois de façon individuelle et collective puisque nous savions quoi faire même s'il a fallu se mettre d'accord sur certains points d'ombre.

Éléments de gestion de projet

1) Planning

Planning prévisionnel :

Date	15-déc	17-déc	04-janv	05-janv	06-janv	13-janv	13-janv	13-janv	14-janv	15-janv
Heures cumulées	2	4	6	8	10	12	14	16	18	20
Création des bases de données										
Jeux de données										
Web :										
Modèle MVC, création de classes										
Interface connexion admin										
Interface accueil client										
Programmation pages										
Finition / Esthétique										
Java :										
Modèle MVC, création de classes										
Interface de connexion admin										
Interface connexion joueur										
Codage des interfaces										
Finition / Esthétique										
Présentation/Diapo										

Planning réel :

Date	15-déc	17-déc	04-janv	05-janv	06-janv	13-janv	13-janv	13-janv	14-janv	15-janv
Heures cumulées	2	4	6	8	10	12	14	16	18	20
Création des bases de données										
Jeux de données										
Web :										
Modèle MVC, création de classes										
Interface connexion admin										
Interface accueil client										
Programmation pages										
Finition / Esthétique										
Java :										
Modèle MVC, création de classes										
Interface de connexion admin										
Interface connexion joueur										
Codage des interfaces										
Finition / Esthétique										
Présentation/Diapo										

Répartition des tâches

Amaury et Pablo se sont mis d'accord pendant toute la durée du codage pour savoir comment réaliser le site, avec quelle fonctionnalités, tout en restant en contact avec Alexandre qui développe la partie java pour que ce soit en harmonie.

Pablo s'est occupé de la partie client, qui concerne principalement l'achat des billets, tandis qu'Amaury a réalisé la partie administrateur, qui concerne la gestion des billets et des codes promotionnels.

Lorsque le projet Web était finalisé, Amaury et Pablo ont terminé l'application JAVA avec Alexandre.

Nous avons également créé ensemble la base de données commune à l'application et au site Web pour être en cohérence.

Afin de se mettre d'accord sur nos deux applications et que tout soit cohérent il y a eu un échange permanent entre les 3 membres du groupe.

2) Système de gestion de version et test

En ce qui concerne le CVS, nous avons utilisé Dropbox, qui a été très utile pour partager notre travail, en revanche, nous reconnaissons que l'utilisation de GIT aurait été plus approprié pour travailler en collaboration.

Enfin, en ce qui concerne les tests, nous avons fait tester les applications (WEB et JAVA) à 10 personnes de notre entourage. Cela nous a permis d'améliorer l'ergonomie et l'efficacité de celles-ci.

3) Bilan personnel

Ce système de répartition des tâches nous a fait penser à ce que l'on peut voir dans une entreprise, ce qui pour nous a été très enrichissant car nous avons appris beaucoup de nouvelles choses aussi bien en terme de gestion du temps qu'en terme de notions.

Si cela était à refaire nous nous impliquerions beaucoup plus sur l'analyse pour ainsi ne pas perdre de temps dans le développement et ainsi optimiser notre temps. De plus, il est clair qu'il aurait fallu se mettre à plusieurs sur l'application JAVA dès le début car celle-ci était la partie la plus concise du projet.

Au final, nous avons déterminé une technique d'analyse propre à nous : il s'agirait de faire les maquettes de l'IHM de notre côté, pour ainsi faire une analyse UML plus complète que l'on proposera au client. Celle-ci s'avère plus flexible pour le client plutôt qu'une analyse IHM qui figerait les choses.