

Compte Rendu

Déroulement du projet de CPOA : partie Gestion Hébergement

De Bryan Boni | Emile Bex

20 janvier 2016



Ce document fait un compte rendu sur le déroulement de la partie développement du cours de CPOA du semestre 3 pour la gestion des hébergements pour le groupe de Bryan Boni, David Vivier et Emile Bex.

Table des Matières

Introduction	2
Gestion des Hébergements.....	2
Organisation	2
Etat d'avancement	3
Description de l'architecture en 3 couches	3
Présentation	3
Métier	3
Données.....	3
Analyse	4
Problèmes rencontrés	5
Améliorations possibles	5
Bilan	5

Introduction

Afin de répondre aux attentes définies dans le **cahier des charges**, lors de la partie conception et modélisation, nous avons passé plus de 22 heures de travail en groupe sur ce projet.

Notre groupe est composé de :

- David Vivier
- Emile Bex
- Bryan Boni

Voici le compte rendu faisant état du déroulement du module développement pour la partie gestion des hébergements, effectuée par Emile Bex et Bryan Boni.

Gestion des Hébergements

Organisation

Pour cette partie, les personnes en charge sont Bryan Boni (en tant que développeur front-end) et Emile Bex (en tant que développeur back-end).

En premier lieu Emile s'est occupé de créer les différentes classes du dossier modèle pendant que Bryan crée l'index, le layout et la fenêtre de connexion.

Ensuite Emile reprenait les interfaces faites par Bryan (au niveau contrôleur), afin d'y implémenter la partie back-end, pendant que celui-ci continuait à faire celles qui n'avaient pas encore été faites.

Et enfin après un bref retour du client sur une présentation du code, des améliorations ergonomiques et des corrections ont été l'ordre du jour des dernières séances.

Chacun avait les 11 séances pour finir son travail.

Etat d'avancement

Cette partie a été terminée le 18/01/2016,

Le site fonctionne et est en ligne à l'adresse : <http://bedtennis.ptut.comlu.com/>.

Les identifiants **pour un membre du Staff** sont :

Login : login3 mot de passe : mdp3

Et pour **un hébergeant** :

Login : login1 mot de passe : mdp1

Description de l'architecture en 3 couches

Pour la réalisation de ce projet, **un modèle MVC** (Modèle Vue Contrôleur) a été choisi.

Présentation

Pour la présentation, Bryan a choisi d'utiliser un layout (design pattern) pour la vue, par lequel toutes les autres classes dans la partie contrôleur passeront pour **la mise en forme**.

Il a aussi privilégié **la sobriété** et la **simplicité** pour les interfaces, afin de ne pas perdre l'utilisateur dans les décors et les couleurs et de rendre plus ergonomique le site web.

Métier

Les contrôleurs prennent en compte les informations saisies par l'utilisateur dans les différentes pages. Des redirections sont faites sur la plupart des pages en fonction de ces paramètres saisis. Le site adopte alors un aspect dynamique, alors qu'il est entièrement codé en **PHP**, sans utilisation de **Javascript**.

Données

MySQL a été choisi pour gérer la base de données pour les deux modules. Nous y accédons par le code Java, par le PHP ainsi que via l'interface **phpMyAdmin** pour son administration.

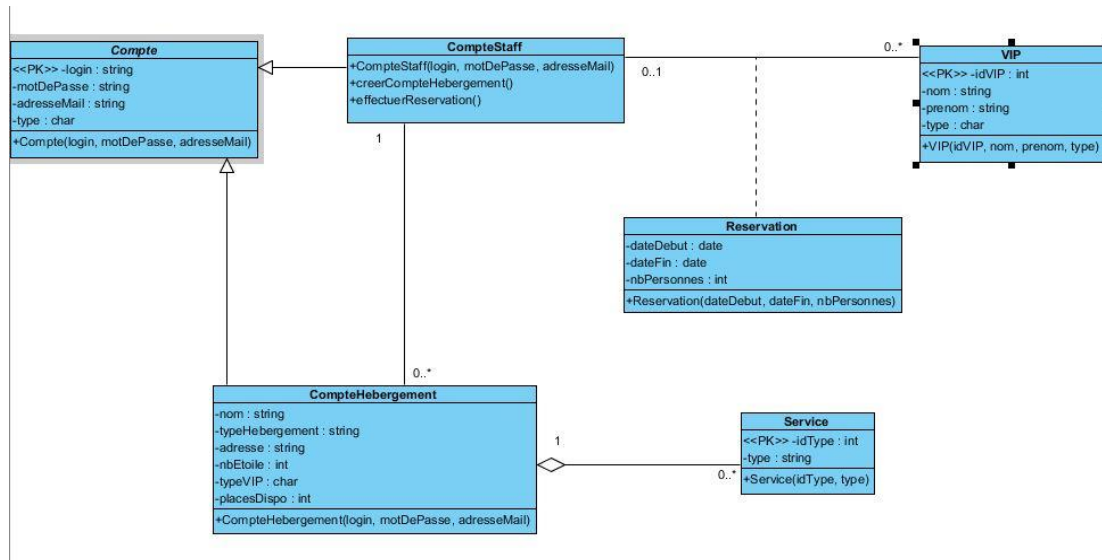
Afin de créer des données utilisables pour notre code dans nos tables, il nous a fallu créer des données sur à peu près toutes les tables. Au départ nous avons choisi de générer ces données automatiquement via une application web ; cependant du fait que les logiciels trouvés ne nous permettaient pas simplement de générer certaines variables, la plupart des données ont été **créées à la main**.

Retour sur l'Analyse

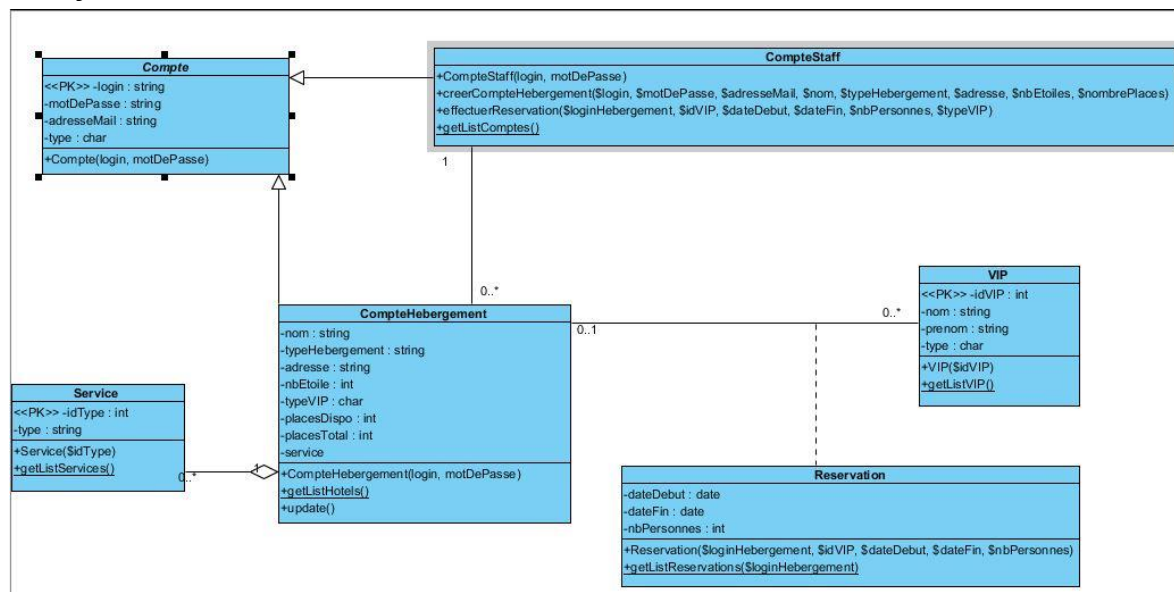
Bryan : l'analyse m'a beaucoup aidé, notamment les maquettes qui m'ont permis de m'orienter dans le développement de l'interface.

Emile : Tous les cas d'utilisations étant assez simples pour ce site, c'est majoritairement le diagramme de classes qui m'a aidé à implémenter toutes les classes, le nom des attributs et des méthodes, malgré les changements qui ont été apportés. Ceux-ci sont les suivants :

Original :



Modifié :



La classe d'association *Reservation* se trouve désormais entre le *CompteHebergement* et le *VIP*, ce qui est logique car une réservation se fait entre un hébergement et un client, le staff ne se retrouvera pas dans la table *Reservation*. Quelques méthodes ont été ajoutées, modifiées ou supprimées.

Problèmes rencontrés

Au début du projet, une base de données sous Oracle avait été choisie, cependant, il nous a été impossible de l'exploiter en PHP. Nous avons dû alors migrer vers une base de données MySQL.

Au milieu du développement nous avons eu un problème avec GitHub Desktop (une alternative aux lignes de commandes) qui ne pouvait plus synchroniser les fichiers de certains et avait même fini par supprimer le contenu d'un des répertoires, ce nous a obligé à créer une nouvelle branche à partir du précédent commit, effectué avant les erreurs de synchronisation et de passer aux lignes de commandes traditionnelles.

Améliorations possibles

La mise à jour des disponibilités ne peut s'effectuer que pour le jour actuel, et non pas jour par jour. Une table de disponibilité aurait dû être créée avec les disponibilités en fonction de la date.

Bilan

Ce projet nous a permis de nous mettre en situation professionnelle, car il nous était demandé de répondre à des attentes client qui n'étaient pas forcément ce que l'on apprécie. Ce qui finalement nous a permis d'acquérir certaines compétences et de comprendre diverses méthodologies utilisées dans le cadre d'un projet informatique professionnel.