**Rapport : application réservation**

**1. Introduction**

Ce projet web consiste à créer un site internet permettant d’effectuer une réservation pour un voyage. Lors de la navigation sur le site, les données entrées par l’utilisateur doivent être gardées en session pour être afficher lors d’un retour à une page précédente par exemple.  
L’application doit utiliser le design pattern MVC, nous avons donc bien une application orienté objet avec un modèle, des vues et un contrôleur qui gère la navigation sur les différentes pages web.   
Le site doit également être relié à une base de données permettant à l’administrateur de voir la liste des réservations et ainsi pouvoir en supprimer ou en éditer. Un deuxième contrôleur a donc été créé pour cette gestion de base de données.

**2. Fonctionnement de l’application**

**2.1 Généralités**

L’application est structurée en 4 parties, pour correspondre au design pattern MVC. Nous avons tout d’abord le modèle (information.php) qui va permettre de créer un objet réservation (appelé info dans le code). Ce modèle contiendra une série de méthode lui permettant de récupérer les valeurs et les mettre à jour. Ensuite nous avons les différentes vues (pages web) qui présentent les données au client en les demandant au modèle.  
Enfin nous avons les contrôleurs, nous en avons deux dans notre cas, un pour la gestion du site en tant que client (Controler.php) et l’autre pour l’administrateur. Ces contrôleurs sont au centre de l’application car ce sont eux qui vont analyser les requêtes du client et coordonner la mise à jour des données du modèle ainsi que l’affichage des vues adéquates.

**2.2 Analyse de la partie client**

Nous allons maintenant nous intéresser plus précisément au fonctionnement du contrôleur qui sera utilisé lors de l’utilisation du site par un client voulant réserver un voyage.  
Tout d’abord, pour se connecter au site l’utilisateur n’entre dans l’url que « Controler.php », la navigation entre les vues se fait automatiquement par la suite, tout l’intérêt d’un contrôleur.   
Il faut tout d’abord lancer une session, étant donné que nous avons deux contrôleurs et qu’ils ont tous deux besoin qu’une session soit lancé, nous vérifierons tout d’abord si une n’est pas déjà lancée.  
On récupère l’objet $info de la session si il existe, sinon on le crée. Pour savoir quel page est à *include* on vérifie à chaque fois quel bouton a été appuyé avec !empty($\_POST(‘nom du bouton’)).

La 1ère page par défaut est la page d’accueil (homepage.php) où l’on entre la destination, le nombre de place et si on désire une assurance annulation ou non. Lorsque l’utilisateur a bien entrée les données et appuie sur « suivant » le contrôleur va enregistrer ces données en session. Il va donc pouvoir les afficher sur cette même page par la suite, par exemple $info->getDestination() permet de récupérer la destination qui a été précédemment enregistré dans $info. Si l’utilisateur a fait une erreur d’encodage on affiche une erreur (homepageError).

Nous sommes maintenant sur la 2ème page (detail.php), celle-ci permet d’entrer le nom et l’âge de chaque voyageur. Si l’utilisateur fait une erreur d’encodage on affiche également une erreur.   
Le contrôleur se charger à nouveau d’enregistrer les différents noms et âges. S’il n’y a pas d’erreur, on affiche la 3ème page (summary.php) qui va tout simplement récapituler toute les informations entrées précédemment et ainsi demander à l’utilisateur de confirmer ou non sa réservation.

Lorsque l’utilisateur confirme sa réservation, la dernière page (confirmation.php) apparait. On peut y voir le montant total du voyage à verser. Lors de cette étape nous allons également devoir enregistrer cette réservation dans la base de données pour pouvoir ensuite l’afficher dans la liste des réservations. Pour se faire on va donc d’abord se connecter à la base de données (reservationbd) qui a été précédemment créé dans phpmyadmin. On écrit les valeurs enregistrées en session dans les tables « reservationinfo » et  « passenger ». Ensuite il y a une la partie d’édition mais nous y reviendront dans la section suivante du rapport.

Le bouton « Annuler » va permettre de revenir à la page d’accueil et de supprimer toutes les valeurs enregistré dans la session. Nous avons également les différents boutons « précédent » qui affichent la page précédente correspondante.