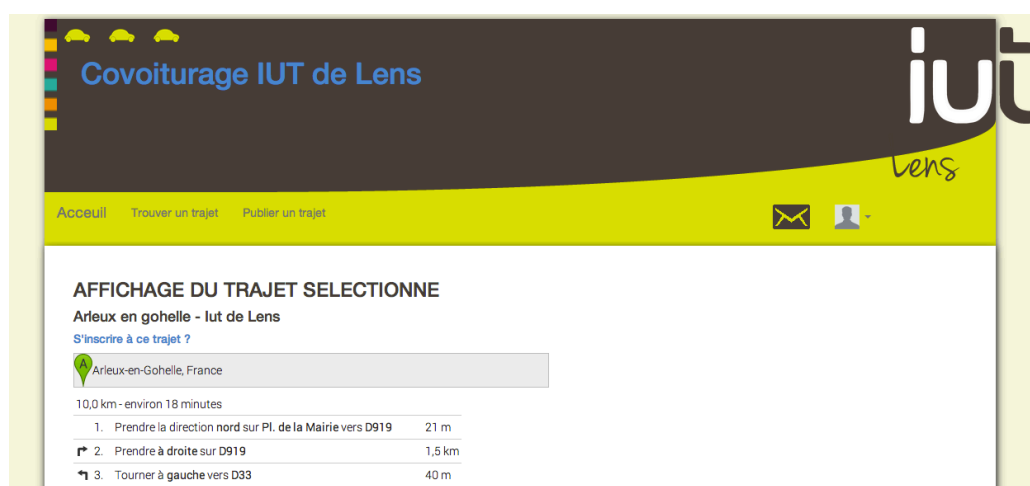


RAPPORT DEVELOPPEMENT WEB



04/04/2013

Site de covoiturage PHP/HTML/CSS/JS

INTRODUCTION

Pourquoi cette application ?

Dans le cadre de la démarche Développement Durable mise en place par l'IUT de Lens et l'université d'Artois nous avons développé une application web permettant de mettre en relation les personnes de l'IUT de Lens. Ce site internet a pour but de poursuivre cette démarche étant donné que depuis deux ans l'accent a été mis sur la promotion de la marche et celle du vélo.

Cette application a plusieurs objectifs qui ciblent différentes catégories de personnes :

- Les utilisateurs du site (Diminution des frais, moins de transport en commun)
- L'IUT (Réduction du nombre de places de parking utilisées, solidarité, réduction des émissions de CO2)
- L'ensemble de la population (moins de pollutions, qualité de l'air améliorée, moins de bouchons sur les routes)

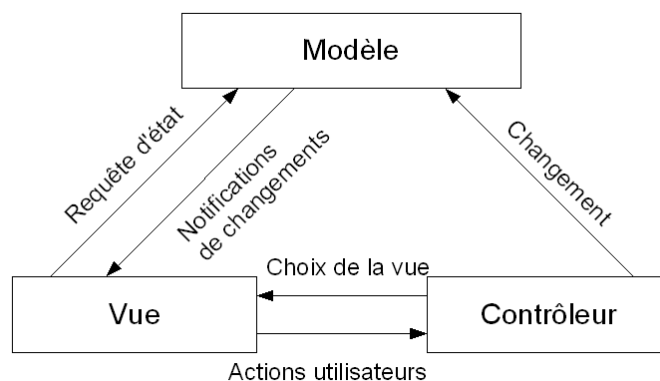
Cette démarche s'inscrit donc totalement comme la continuité des actions déjà entreprises par l'IUT de Lens et l'université d'Artois et elle a donc suscité notre intérêt. En plus de proposer un service de covoiturage nous avons voulu mettre en avant les différentes démarches mises en place précédemment. Durant le développement de l'application nous avons voulu réserver un espace dédié aux objectifs du covoiturage et de ce site afin de sensibiliser les visiteurs.

ARCHITECTURE

Général

Notre objectif était d'établir un site simple d'utilisation, nous avons donc décidé d'utiliser un système d'inscription/connexion afin de pouvoir fidéliser le visiteur. Le visiteur a donc la possibilité de chercher un trajet qui correspond à ses critères et de s'y inscrire. La page d'accueil quant à elle sert à pouvoir visualiser les derniers trajets publiés.

Le site utilise le modèle MVC :



Le principal avantage de cette méthode de développement est qu'elle permet de bien différencier les parties du code :

- Le modèle permet de faire le lien avec la base de données, de récupérer et d'envoyer des données vers celle-ci. Les données récupérées sont transmises au contrôleur afin d'effectuer les traitements nécessaires.
- Le contrôleur c'est les parties du code qui effectue le traitement des données. Une fois les données traitées on charge la vue en la permettant d'utiliser et d'afficher les données traitées.
- La vue c'est ce que l'utilisateur du site va voir, elle est en grande partie composée de code HTML / JavaScript.

Choix techniques

Technologies utilisées

- Pour notre base de données nous avons utilisé MySQL, c'est un SGBD simple à mettre en place en local et la nature de notre application ne nous demandait pas de traiter une énorme quantité de données, nous n'avons donc pas d'intérêt à utiliser des SGBD plus puissants comme PostgreSQL.
- Pour faire le lien avec la base de données et effectuer les traitements nous avons utilisé du PHP.

- L'affichage est fait avec un mélange d'HTML et de JavaScript, ce dernier nous permettant d'avoir des pages plus dynamiques pour l'utilisateur.
- En plus de ces différents outils nous avons décidé d'utiliser un framework : CodeIgniter. Ce framework nous a permis de bien comprendre le fonctionnement de l'architecture et étant assez simple à utiliser, bien documenter, léger et comportant de nombreuses méthodes très utiles pour réduire le temps de développement et s'assurer de générer du code de bonne qualité.
- Nous avons utilisé l'API de Google Maps couplé à une bibliothèque* dédiée à notre framework permettant d'utiliser l'API fournie par Google Maps plus simplement et avec une quantité de code réduite et très peu de redondance. Cette API nous permettait par des appels de fonctions PHP dans nos contrôleurs de pouvoir très facilement charger des itinéraires et la carte Google Maps dans notre vue et cela implique une réduction drastiquement la quantité de JavaScript à coder.
- Nous avons développé une petite bibliothèque pour le framework afin de pouvoir générer des liens, afficher des images, inclure des feuilles de styles dans une vue avec un simple appel à une fonction PHP.
- Pour les formulaires nous avons utilisé une bibliothèque PHP permettant de vérifier des formulaires de façon très sûre (caractères spéciaux, longueur minimale et maximale, validité d'une adresse e-mail)
- Les sessions sont stockées sur la base de données. En fonction de si l'utilisateur coche qu'il souhaite resté connecté ou non la session dans la base de données a une durée de vie illimitée ou bien elle est détruite à la fermeture du navigateur. Ces sessions stockées dans la base de données permettent de stocker un tableau associatif et donc de pouvoir réutiliser ces valeurs le temps de la session.
- Le framework nous permet de générer des URL très claires et nous n'avons plus besoin d'utiliser d'extensions pour nos fichiers. Les pages s'appellent en utilisant le nom du contrôleur.
- Une interface d'administration a été créée afin de pouvoir empêcher l'accès au site à certains utilisateurs, de pouvoir redonner l'accès au site à cet utilisateur banni et aussi afin de pouvoir supprimer des trajets.

Design, code couleur et en mise en page :

Nous avons voulu que notre site soit au couleur de l'IUT afin qu'une personne qui le visite ne soit pas dépaycé par rapport au site officiel. C'est pourquoi on peut voir une prédominance de vert, de marron et de blanc.

Afin de symboliser que le service proposé par le site s'adresse à l'ensemble des départements de l'IUT et non uniquement au département informatique nous avons réutilisé, à la manière du site de l'IUT, les carrés de couleur en haut à gauche de la page, chaque carré correspond à un département.

Nous avons utilisé un footer et un header qui se charge dans chaque vue afin d'éviter la redondance de code HTML dans les vues au moment de générer les affichages. Le header permet notamment d'inclure les fichiers JavaScript et les feuilles de style.

L'utilisation du JavaScript afin d'afficher un calendrier ou un menu cliquable nous permettent de générer des pages plus dynamiques que le PHP seul.

Le site est donc assez sobre et les différents affichages sont présentés dans des containers ombragés qui permettent de bien séparer les informations proposées à l'utilisateur.

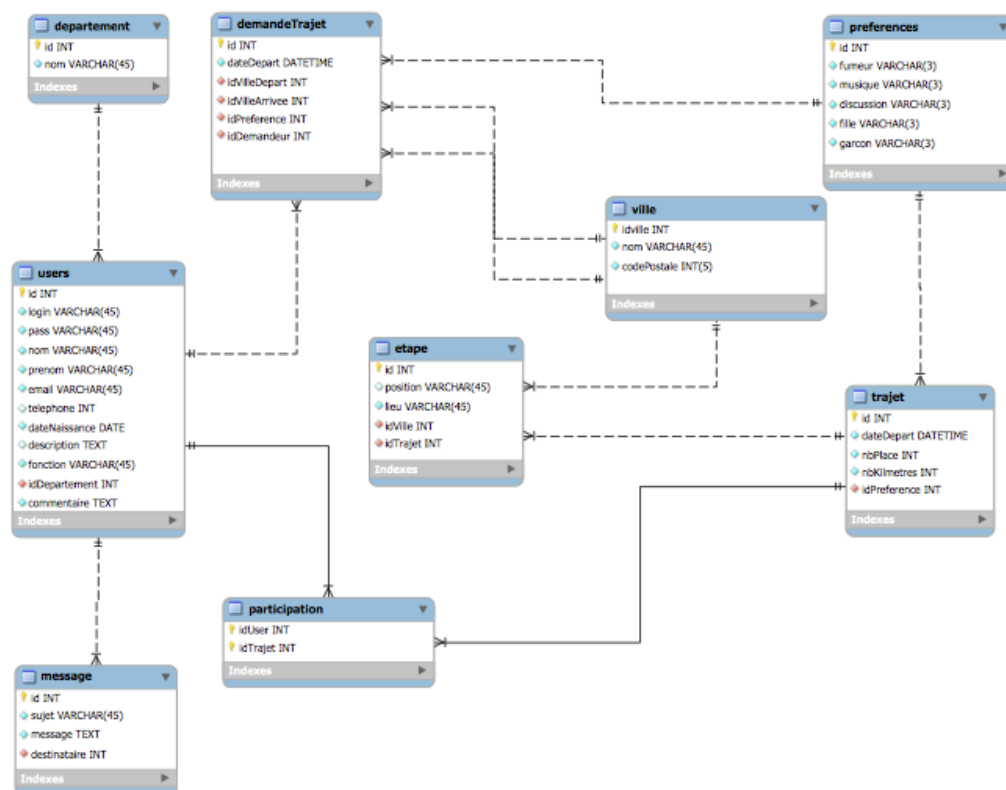
DIFFICULTES RENCONTREES

Gestion de versions

Nous avons eu quelques difficultés quant à la gestion des versions de nos fichiers avec Git Hub. Le temps de comprendre le fonctionnement des branches, du merging et cela a peut être causé un peu de redondance pour certaines parties du code.

Base de données

La base de données prévue initialement s'est révélée un peu trop compliquée à utiliser :



Nous avons donc dû se passer de certaines parties de la table et donc de réduire certaines fonctionnalités.

BILAN

Dans un premier temps, on peut dire que ce projet nous a donné la possibilité d'acquérir des compétences sur certains domaines :

- Le fonctionnement du MVC.
- L'utilisation de Google Maps dans le cadre d'un site web.
- Proposer un design efficace pour tous.
- Utilisation de certaines bibliothèques PHP.
- Utilisation d'un logiciel de versioning.
- Essayer de mieux sécuriser une application web.
- L'utilisation d'un framework, chose qui n'avait pas du tout été vu lors de notre formation.

Dans un second temps, nous avons tout de même rencontré certaines difficultés, surtout pour la conception même de la solution ce qui nous a parfois particulièrement ralenti, peut être a cause d'une conception trop ambitieuse de la base de données de départ en voulant inclure trop de fonctionnalités dès le début.

Fonctionnalités non terminées

Au moment de la rédaction du rapport certaines fonctionnalités ne sont pas encore terminées :

- Recherche
- Itinéraires des étapes

Pour conclure, le sujet du projet était intéressant et s'ancrait bien l'idée de faire de l'IUT de Lens un lieu plus convivial. Les technologies appréhendées nous ont permis de mieux cerner certains aspects du développement web.

SOURCE(S)

API Google Maps pour CodeIgniter : <http://biostall.com/codeigniter-google-maps-v3-api-library>