

Rapport du projet L007

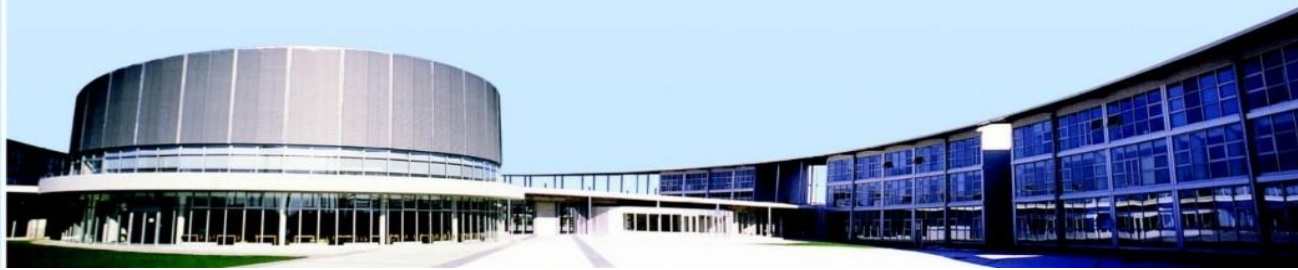
Site de covoiturage

Réalisé par: Guodong HE

Jie HOU

Responsable de l'UV: Marc LEMERCIER

Semestre: P2015



SOMMAIRE

Introduction.....	3
Analyse des contraintes et des besoins du cahier des charges.....	4
a. Création d'un nouveau compte et des différences parmi trois comptes	4
b. Information sur le véhicule	4
c. Ajout d'un nouveau trajet	4
d. Sélection d'un trajet	5
e. Réservation d'un trajet.....	5
f. Suppression d'un trajet	5
g. Préparation des trajets.....	6
h. Validation des trajets	6
i. Appréciations des trajets	6
j. Visualisation de la liste des comptes.....	6
k. Visualisation de la liste des trajets	7
Fonctionnalité et implémentation	8
a. Inscription de l'utilisateur	8
b. Connexion de l'utilisateur	9
c. Fonctionnalité pour le passager	11
1. Affiche info privé	12
2. Ajouter un véhicule	13
3. Recherche un trajet.....	13
4. Réserver un trajet.....	14
5. Apprécier un trajet et affiche appréciation	16
6. Messagerie	17
d. Fonctionnalité pour le conducteur.....	18
1. Affiche info véhicule.....	18
2. Propose un trajet.....	18
3. Préparer un trajet.....	19
4. Supprimer un trajet.....	21
5. Valider un trajet.....	22
e. Fonctionnalité pour l'administrateur.....	22

1. Afficher tous les compte	23
2. Affiche info trajet	23
Etape de conception du site	24
a. Diagramme de cas d'utilisation.....	24
b. Conception du site	26
Arborescence du site.....	26
c. Outils utilisés	27
Etape de conception dE la base de données.....	28
a. Schéma de la base de données	28
b. Modèle physique des données	29
Conclusion	33
Bibliographie.....	35
annexe	36
Script de création de la base de données	36

INTRODUCTION

Dans le cadre de l'UV LO07 concernant une introduction aux technologies du web, nous avons réalisé le projet dont l'objectif était de concevoir, de développer et de tester un site Web dynamique de covoiturage.

Dans ce projet, on doit résoudre quelques problématiques. Premièrement, des conducteurs disposant de véhicule proposent des places sur des trajets publiés sur un site web. Deuxièmement, des passagers pourront ensuite consulter le site et sélectionner des trajets en fonction de plusieurs critères de recherche. Troisièmement, seuls les conducteurs et les passagers préalablement enregistrés ont le droit d'effectuer des offres ou des réservations de trajets. Quatrièmement, le site dispose d'un système de messagerie interne à l'application. Cinquièmement, il dispose aussi d'un système d'appréciation permettant aux utilisateurs d'exprimer un avis sur le déroulement d'un trajet.

ANALYSE DES CONTRAINTES ET DES BESOINS DU CAHIER DES CHARGES

a. Création d'un nouveau compte et des différences parmi trois comptes

S'il n'y a pas de compte, on ne peut pas faire le login. Quand on s'est inscrit en soumettant des informations de nom, de prénom, de sexe, d'année de naissance, de mot de passe et d'adresse d'e-mail, on devient justement un passager, parce qu'il faut distinguer le passager et le conducteur. Après avoir ajouté un véhicule, l'identification de passager est changée en conducteur. Actuellement, le conducteur peut faire toutes les choses qu'un passager peut faire sauf la fonctionnalité d'ajouter un véhicule du fait qu'il ne possède qu'un seul véhicule. Le conducteur peut afficher des informations de véhicule et proposer des trajets, au contraire, le passager ne peut pas les faire.

De plus, le compte d'administrateur est déjà fixé, donc on n'a pas de droit pour créer le compte d'administrateur. Comme un administrateur, il a le droit de regarder l'ensemble des comptes d'utilisateur et l'ensemble de toutes les informations de chaque trajet.

Pour ce cahier, on utilise le formulaire dynamique de PHP pour transmettre des informations dans la TABLE utilisateur et la TABLE passager dans la base de données et la valeur de l'attribut `is_conducteur` dans la TABLE utilisateur est 0.

b. Information sur le véhicule

Quand un passager soumet des infos de voiture, le serveur va les transmettre dans la TABLE véhicule, et dans le même temps des données de lui va être transmises dans la TABLE conducteur et la valeur de l'attribut `is_conducteur` dans la TABLE utilisateur va être changée comme 1.

c. Ajout d'un nouveau trajet

Juste un utilisateur ayant le statut de conducteur et identifié par son login et password peut proposer des trajets. Et on ne peut que proposer un trajet à l'avenir du fait que du temps est monodimensionnel.

Ici, on utilise aussi le formulaire dynamique de PHP pour transmettre des infos dans la TABLE trajet. Pour l'attribut statut, il y a 4 possibles valeurs : non effectué, effectué par client, préparation, effectué. Si on n'a pas manipulé le trajet, la statue est « non effectué » par défaut. Pour d'autres 3 valeurs, on va en parler dans la situation correspondante.

d. Sélection d'un trajet

N'importe qui identifié peut rechercher des trajets qui sont dans la TABLE trajet dans la base de données et dont la valeur de l'attribut statut n'est pas « effectué ». On doit rechercher un trajet en précisant la ville de départ, la ville d'arrivée et une date. Ici, on utilise l'attribut « min » pour contrôler du temps que l'utilisateur choit à l'avenir. Tous les trajets trouvés sont basés sur la TABLE trajet dans la base de données.

e. Réservation d'un trajet

Un passager peut réserver des places sur l'un des trajets présentés à l'écran. Il n'est pas possible de réserver un nombre de places supérieur au nombre de places disponibles. Le nombre de places disponibles est défini par défaut quand le conducteur a proposé le trajet.

Quand un passager a réservé la place n'importe combien mais inférieur à le nombre original, le nombre de place dans les informations présentes va être réduit la valeur correspondante, aussi dans le même temps la valeur correspondante de l'attribut nombre_place dans la TABLE trajet va être réduite et l'attribut statut va être modifié en « effectué par client ».

Hors cela, pour notre projet, quand un passager a réservé le trajet, l'argent dans le compte va être décompté, c'est-à-dire, la valeur de l'attribut compte dans la TABLE utilisateur va réduit.

Comme pour un trajet, il y aura quelques passagers, donc on a créé une nouvelle TABLE réservation pour mémoriser des identifications de passager, de conducteur et le trajet correspondant. Ça c'est plus facile pour administrateur et conducteur de consulter des informations de trajet.

f. Suppression d'un trajet

Si le conducteur supprime son trajet, il va rembourser des passagers déjà inscrits en $10 * \text{nombre de places réservées}$. Il va transmettre des messages via des passagers

inscrits par le système interne de messagerie pour leur informer. Le serveur va faire le virement automatiquement du compte de conducteur aux comptes des passagers.

A cause de la clé étrangère de la TABLE trajet, il va d'abord supprimer le trajet dans la TABLE réservation, et après supprimer le trajet dans la TABLE trajet.

g. Préparation des trajets

Un conducteur peut afficher la liste de ses propres trajets avec la liste des passagers inscrits. Quand il veut valider le trajet, il va transmettre des messages via des passagers inscrits par le système interne de messagerie. Et ce moment-là, l'attribut statut dans la TABLE trajet va être changé comme « préparation ».

h. Validation des trajets

Pour cette action, quand le conducteur clique le bouton « valider », l'argent des passagers va être versé dans son compte, c'est-à-dire dans la TABLE utilisateur, le compte de ce conducteur va obtenir la valeur correspondante. Et aussi, le pareil que faire la préparation des trajets, ça va permettre de transmettre des messages via des passagers inscrits par le système interne de messagerie.

De plus, l'attribut statut dans la TABLE trajet va être changé comme « effectué ». Et depuis ce moment, d'autres passagers ne trouvent plus ce trajet. Ici, on utilise le jugement de PHP et l'interaction entre base de données.

i. Appréciations des trajets

On a créé une nouvelle TABLE appréciation. Chaque passager et chaque conducteur à la possibilité d'attribuer une évaluation et donner la note sur le déroulement du trajet.

j. Visualisation de la liste des comptes

Comme un administrateur, il a un compte particulier lui permettant d'avoir la fonctionnalité à présenter la liste des tous utilisateurs avec l'ensemble des données de leur compte à l'écran.

k. Visualisation de la liste des trajets

Comme un administrateur, il a un compte particulier lui aussi permettant d'avoir une autre fonctionnalité à présenter l'ensemble des trajets (non effectués) présents dans la base de données avec les informations sur le conducteur et les passagers inscrits.

FONCTIONNALITE ET IMPLEMENTATION

a. Inscription de l'utilisateur

Dans cette page, on a un formulaire à remplir pour s'inscrire. Il y a huit champs dans cette formulaire : sexe, nom, prénom, année de naissance, adresse e-mail, mot de passe, confirmation de mot de passe, le bouton de soumettre.

Inscription | BlaBlaCar

localhost/LO07projet/page_inscrit.php

BlaBlaCar

S'inscrire | Se connecter | Like 2.5k

Pas encore membre ? Inscrivez-vous gratuitement

en 30 secondes avec une adresse email

Civilité Homme

Votre nom *

Votre prénom *

Votre mail

Mot de passe

Confirmer le mot de passe

Année de naissance 1920

Submit

Grâce à l'utilisation de JavaScript, nous pouvons effectuer l'examen de forme de nom et prénom, aussi l'examen de mot de passe confirmé. Si la forme de nom ou prénom est incorrect, ou le mot de passe confirmé ne correspond pas au mot de passe écrit, il va donner un avertissement. Quand on clique le bouton « Submit », des informations sont soumises dans la base de données. Pour chaque utilisateur inscrit, on

lui donne un compte de login pour se connecter. Et ce login est composé par son nom et prénom sans espace en minuscule.

The screenshot shows a web browser window with the address bar displaying 'localhost/LO07projet/page_inscrit.php'. The page content includes the text 'Pas encore membre ? Inscrivez-vous gratuitement' and 'en 30 secondes avec une adresse email'. Below this is a registration form with the following fields and errors:

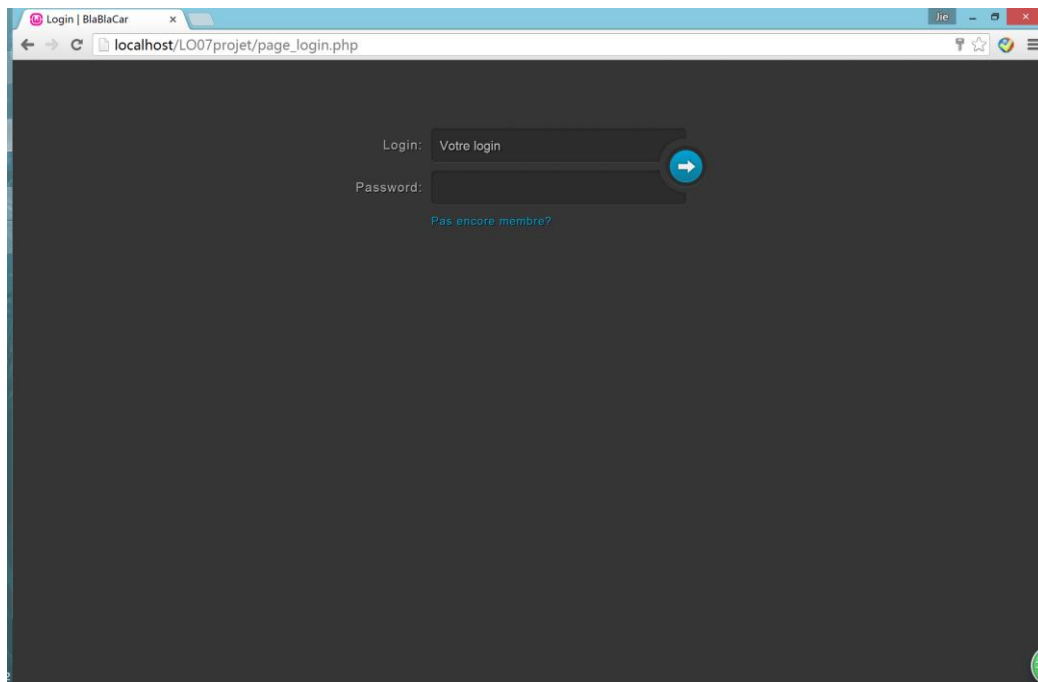
- Civilité:** A dropdown menu set to 'Homme'.
- Votre nom:** A text input field with a red error message 'Le forme est faux' below it.
- Votre prénom:** A text input field with a red error message 'Le forme est faux' below it.
- Email:** A text input field containing '@utt.fr'.
- Password:** A text input field with a red error message 'Cette valeur n'est pas valide.' below it.
- Année de naissance:** A dropdown menu set to '1920'.
- Submit:** A button at the bottom of the form.

b. Connexion de l'utilisateur

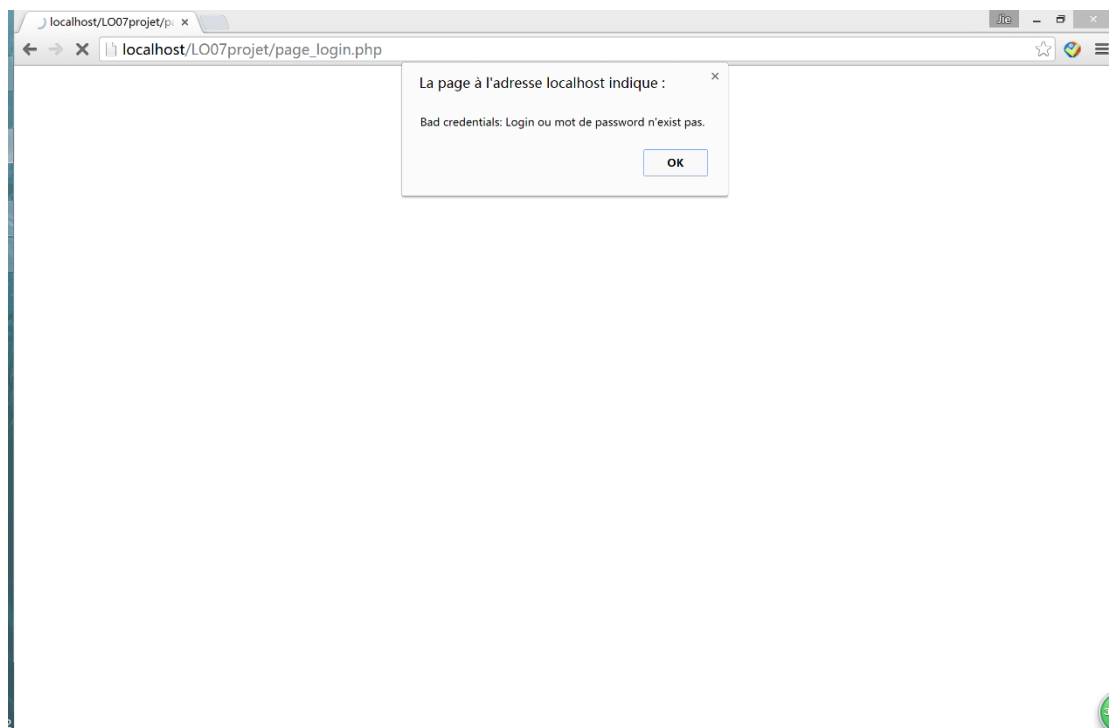
Quand on arrive la page de login, il faut utiliser correct login et password. En utilisant la fonction de Javascript, nous pouvons effectuer une pré-vérification pour les informations de connexion avant de l'authentification.

Si les informations sont correctes, l'authentification au serveur sera faite pour l'utilisateur afin de stocker une session qui possède d'identification de l'utilisateur: s'il

est un passager, un conducteur, ou l'administrateur. Après avoir identifié, le navigateur va diriger le site vers la page correspondante selon l'identification.

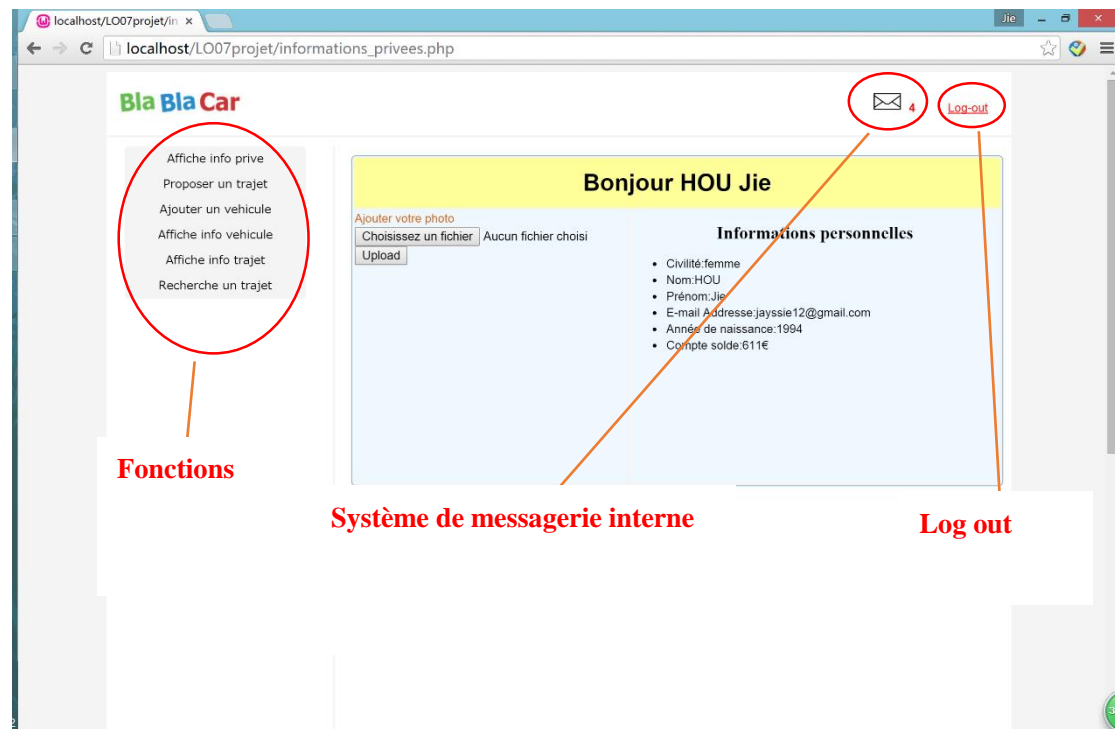


Si le mot de passe est incorrect, ou le login qu'on a entré n'existe pas, ça va sauter une alerte.



c. Fonctionnalité pour le passager

Ci-dessous, c'est l'interface graphique d'information privée pour passager et aussi conducteur :

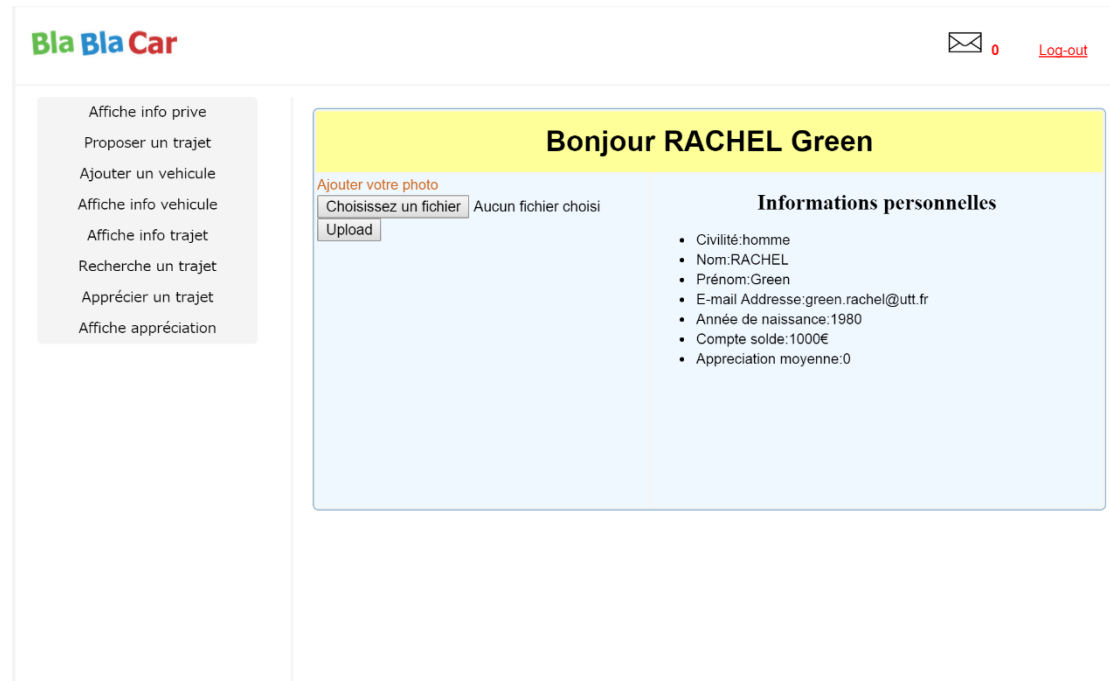


Dans cette page, la colonne à gauche contient les actions que utilisateurs peuvent faire, il y a « afficher info privé », « proposer un trajet », « ajouter un véhicule », « affiche info trajet », « recherche un trajet », « affiche appréciation ». Dans le coin supérieur droit, on peut accéder système de messagerie interne ou log out.

Quand on demande d'accéder cette page, le serveur va examiner la session de l'utilisateur, si il n'y en a pas, la page va être rédigée vers la page connexion. S'il y a la session, le serveur va lire des informations de l'utilisateur correspondantes dans la base de données et les transmettent vers le navigateur pour présenter.

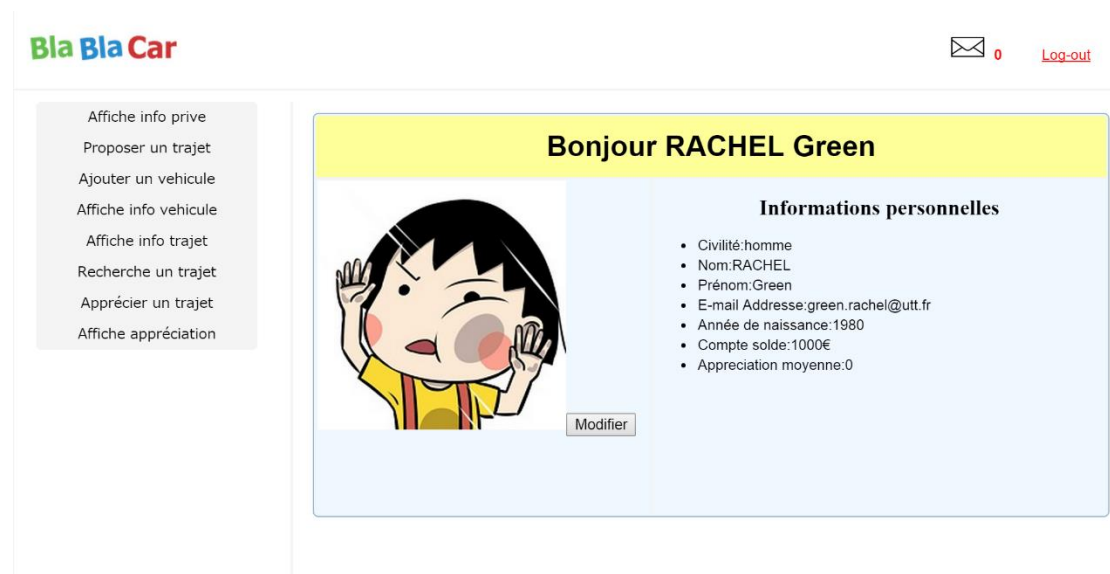
1. Afficher info privée

Quand on arrive la page info privé, on a déjà effectué la fonction “affiche info privé”. Si quand on reste dans d’autre page et clique ce bouton, la page va retourner info privé.



The screenshot shows the 'Bla Bla Car' user interface. On the left is a sidebar menu with options: 'Affiche info prive', 'Proposer un trajet', 'Ajouter un vehicule', 'Affiche info vehicule', 'Affiche info trajet', 'Recherche un trajet', 'Apprécier un trajet', and 'Affiche appréciation'. The main content area is titled 'Bonjour RACHEL Green'. Below the title, there's a section 'Ajouter votre photo' with a 'Choisissez un fichier' button, the text 'Aucun fichier choisi', and an 'Upload' button. To the right of this is the 'Informations personnelles' section, which lists: 'Civilité: homme', 'Nom: RACHEL', 'Prénom: Green', 'E-mail Adresse: green.rachel@utt.fr', 'Année de naissance: 1980', 'Compte solde: 1000€', and 'Appreciation moyenne: 0'. At the top right, there's a mail icon with '0' and a 'Log-out' link.

Et ici, on peut réaliser transmettre la photo vers le serveur et dans le même temps afficher la photo. Bien sûr, on peut changer la photo n’importe quand. Pour cette étape, on juste transmet l’URL de la photo dans le serveur et la base de données par le formulaire dynamique de PHP.



This screenshot shows the same 'Bla Bla Car' user interface as the previous one, but with a cartoon image of a girl with black hair and a yellow shirt in the 'Ajouter votre photo' section. A 'Modifier' button is now visible below the image. The 'Informations personnelles' section and the top navigation elements remain the same.

2. Ajouter un véhicule

On a un formulaire dynamique à remplir. Pour le domaine de « année de mise en service », on a utilisé la boucle de *for* pour construire « option ».

The screenshot shows the 'Ajouter un véhicule' (Add a vehicle) form in the BlaBlaCar application. The left sidebar contains a menu with options: 'Affiche info prive', 'Proposer un trajet', 'Ajouter un vehicule' (highlighted), 'Affiche info vehicule', 'Affiche info trajet', 'Recherche un trajet', 'Apprécier un trajet', and 'Affiche appréciation'. The main form area is titled 'Véhicule' and contains several dropdown menus and input fields: 'Marque' (Renault), 'Modele' (Zoe), 'Confort' (Basique), 'Nb de places' (1), 'Annee de mise en service' (1990), 'Couleur' (noir), and 'Catégorie' (Véhicule de tourisme). A 'Valider' button is at the bottom.

3. Recherche un trajet

Ci-dessous est la page de recherche trajet, on peut trouver tous les trajets en juste entrant la point de départ.

The screenshot shows the 'Recherche un trajet' (Search a trip) form in the BlaBlaCar application. The left sidebar is the same as in the previous screenshot, with 'Recherche un trajet' highlighted. The main form area is titled 'Recherche un trajet' and contains two input fields: 'Point de depart' (Exemple: Paris, Lyon) and 'Point d'arrivee' (Exemple: Troyes, Lile). Below these is a section titled 'Date et Horaire' with a 'Date aller:' label and two input fields: 'jj/mm/aaaa' and '--:--'. A 'Valider' button is at the bottom.

Quand on valide le formulaire de PHP, on a des infos de trajet correspondant à la commande d'utilisateur.

Bla Bla Car 0 [Log-out](#)

- Affiche info prive
- Proposer un trajet
- Ajouter un vehicule
- Affiche info vehicule
- Affiche info trajet
- Recherche un trajet**
- Apprécier un trajet
- Affiche appréciation

Trajet list [Reservez](#)

Trajet ID : 1
Ville deart : Troyes
Ville arrive : Paris
Date depart : 2015-07-06 Time depart : 16:00
Prix : 11 Nombre de place rest : 1
Statut : effectue_par_client
Nombre de passager: 1 ▼

Trajet list [Reservez](#)

Trajet ID : 2
Ville deart : Troyes
Ville arrive : Reimis
Date depart : 2015-07-13 Time depart : 08:00
Prix : 10 Nombre de place rest : 3
Statut : non_effectue
Nombre de passager: 1 ▼

Trajet list [Reservez](#)

Trajet ID : 3
Ville deart : Troyes
Ville arrive : Lille
Date depart : 2015-07-18 Time depart : 13:00
Prix : 17 Nombre de place rest : 1
Statut : effectue_par_client
Nombre de passager: 1 ▼

4. Réserver un trajet

Après avoir cliqué le bouton « Réserver », ça signifie que l'utilisateur a choisi ce trajet, son compte va être décompté de l'argent correspondant.

Bla Bla Car 0 [Log-out](#)

- Affiche info prive
- Proposer un trajet
- Ajouter un vehicule
- Affiche info vehicule
- Affiche info trajet
- Recherche un trajet
- Apprécier un trajet
- Affiche appréciation

Bonjour BUFFAY Phoebe

Ajouter votre photo
Choisissez un fichier Aucun fichier choisi
Upload

Informations personnelles

- Civilité: femme
- Nom: BUFFAY
- Prénom: Phoebe
- E-mail Adresse: phoebe.buffay@utt.fr
- Année de naissance: 1984
- Compte solde: 989€
- Appréciation moyenne: 0

La valeur de compte était par défaut 1000. Et maintenant il reste 989 euro. Ici, on utilise \$_post pour obtenir le nombre de places et le prix d'unité de place, et après avoir calculé la valeur, connecter la base de données pour modifier la TABLE utilisateur.

Après avoir réservé la place, des informations de trajet correspondantes sont aussi changées.

The screenshot shows the BlaBlaCar website interface. On the left is a sidebar with navigation links. The main content area displays a list of three travel offers, each with a 'Trajet list' header and a 'Reservez' button. The second offer is circled in red, and a red arrow points to it with the text 'Devenir 3, ici !'.

BlaBlaCar 0 [Log-out](#)

Trajet list [Reservez](#)

Trajet ID : 1
Ville deart : Troyes
Ville arrive : Paris
Date depart : 2015-07-06 Time depart : 16:00
Prix : 11 Nombre de place rest : 4
Statut : non_effectue
Nombre de passager : 1 ▼

Trajet list [Reservez](#)

Trajet ID : 2
Ville deart : Troyes
Ville arrive : Reims
Date depart : 2015-07-13 Time depart : 08:00
Prix : 10 Nombre de place rest : 3
Statut : non_effectue
Nombre de passager : 1 ▼

Trajet list [Reservez](#)

Trajet ID : 3
Ville deart : Troyes
Ville arrive : Lille
Date depart : 2015-07-18 Time depart : 13:00
Prix : 17 Nombre de place rest : 4
Statut : non_effectue
Nombre de passager : 1 ▼

Devenir 3, ici !

5. Apprécier un trajet et affiche appréciation

Quand on clique le bouton « apprécier un trajet », on a la page :

BlaBlaCar [Email icon] 1 [Log-out](#)

- Affiche info prive
- Proposer un trajet**
- Ajouter un vehicule
- Affiche info vehicule
- Affiche info trajet
- Recherche un trajet
- Apprécier un trajet
- Affiche appréciation

Trajet Inscrit

Trajet ID	Depart	Arrive	Date	Time	PT	Prix	PR	Conducteur	Status
1	Troyes	Paris	2015-07-06	16:00	4	11	1	RACHEL Green	effectue

Trajet Propose

Choisissez un trajet:


Appréciation: Note moyennes:

Après avoir cliqué « appréciation », la note va être mémorisée dans la base de données et aussi présentée dans la page d'info privé de conducteur correspondant. S'il y a plusieurs passagers qui ont déjà apprécié, la note présentée est la note moyenne. Ici, on utilise PHP à obtenir toutes les notes et la commande « `Note moyenne = sum(a.note)/count(a.a_id)` » pour calculer.

BlaBlaCar [Email icon] 1 [Log-out](#)

- Affiche info prive
- Proposer un trajet
- Ajouter un vehicule
- Affiche info vehicule
- Affiche info trajet
- Recherche un trajet
- Apprécier un trajet
- Affiche appréciation

Bonjour RACHEL Green

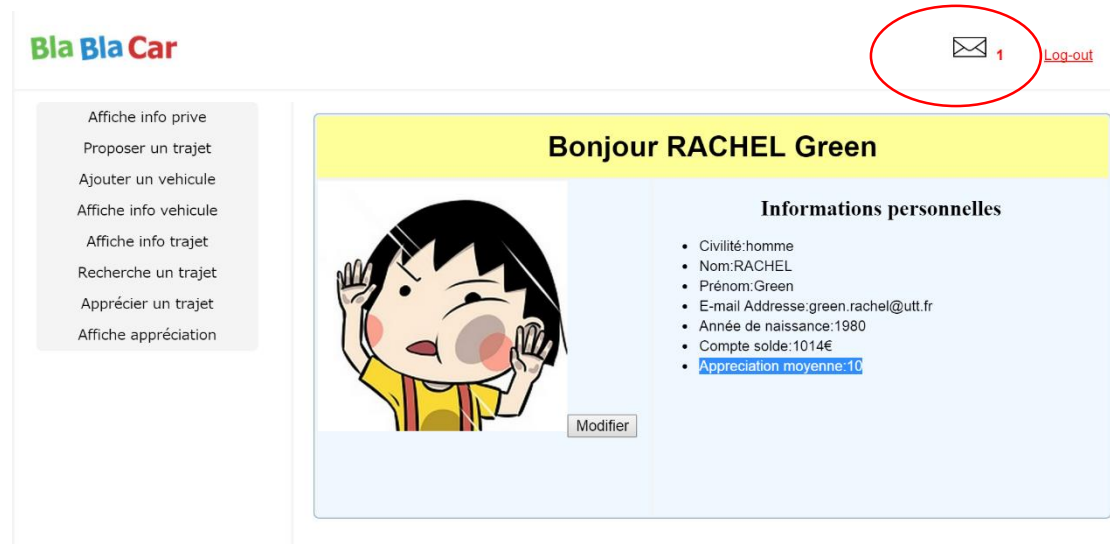


Informations personnelles

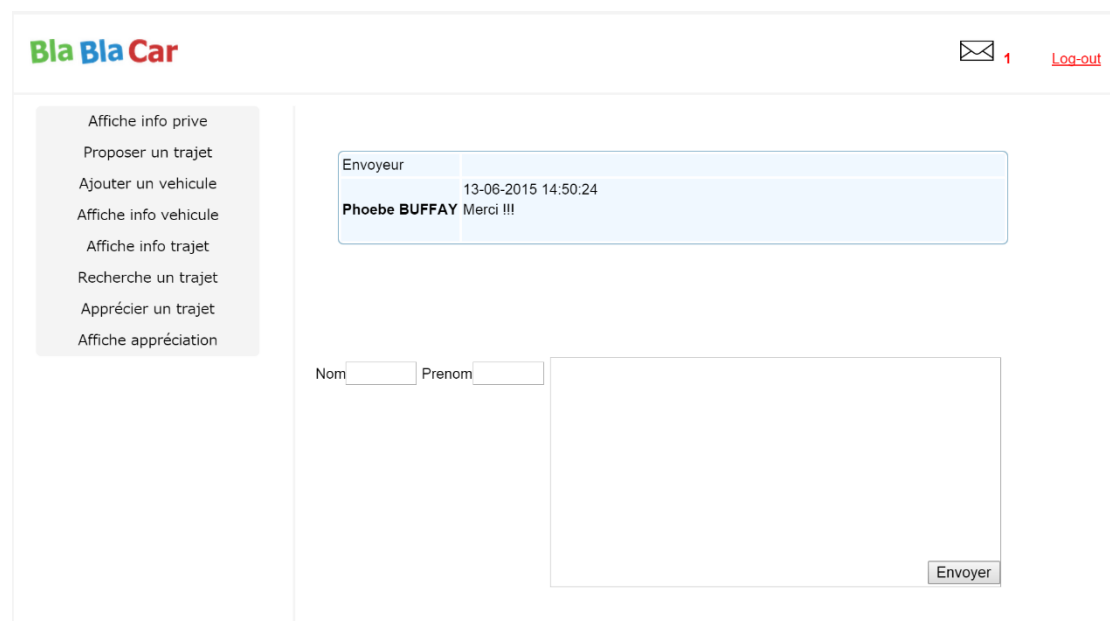
- Civilité: homme
- Nom: RACHEL
- Prénom: Green
- E-mail Adresse: green.rachel@utt.fr
- Année de naissance: 1980
- Compte solde: 1014€
- Appreciation moyenne: 10**

6. Messagerie

Quand le conducteur supprime un trajet, préparer un trajet, valider un trajet ou d'autre personne envoie un message, on peut être informé dans n'importe quelle page. On a aussi utilisé la langue PHP pour réaliser. Chaque fois qu'on accède le compte personnel, le serveur va contrôler la TABLE messagerie de la base de données s'il y a le message pas lu ou pas.



Quand on entre le système de messagerie, on peut regarder tous les messages reçus. Lorsqu'on clique le nom dans la colonne de « Envoyeur », à droite, il va afficher tous les messages de lui et du temps de envoi. Ici. On utilise l'événement de HTML 'onclick' et la commande de `document.getElementById().innerHTML` pour réaliser cette fonction. On peut aussi envoyer message à n'importe qui s'est inscrit.



d. Fonctionnalité pour le conducteur

1. Affiche info véhicule

Lors de cliquer « affiche info véhicule », on a toutes les informations de véhicule. Etant donné qu'un conducteur possède une seule voiture, on n'a pas de capacité de modifier l'info.

The screenshot displays the BlaBlaCar web application interface. At the top left is the 'BlaBlaCar' logo. At the top right, there is a mail icon with a '0' next to it, and a 'Log-out' link. On the left side, there is a vertical menu with the following items: 'Affiche info prive', 'Proposer un trajet', 'Ajouter un vehicule', 'Affiche info vehicule' (highlighted in yellow), 'Affiche info trajet', 'Recherche un trajet', 'Apprécier un trajet', and 'Affiche appréciation'. The main content area on the right is titled 'Véhicule' in a dark purple header. Below this header, the following vehicle details are listed: 'Vehicule ID : 1', 'Proprietaire ID : 1', 'Marque : Renault', 'Modele : Zoe', 'Confort : Basique', 'Nombre de place : 5', 'Couleur : Bleu', 'Anne de mise en service : 2014', and 'Catagorie: Vehicule de tourisme'.

2. Propose un trajet

C'est un juste formulaire dynamique à remplir. Ici, dans le domaine « date aller », nous utilisons la commande « `min=<?php echo date('Y-m-d') ?>` `max="2115-09-`

12" » pour contrôler la date réel. Après avoir proposé le trajet, le passager peut rechercher le trajet, sinon, il n'y a aucun résultat à afficher.

The screenshot shows the 'Proposer un trajet' (Propose a trip) form on the BlaBlaCar website. The interface includes a sidebar with navigation links, a main form area with three sections: 'Trajet', 'Itineraire', and 'Date et Horaire', and a 'Log-out' link in the top right corner.

Navigation Sidebar:

- Affiche info prive
- Proposer un trajet**
- Ajouter un vehicule
- Affiche info vehicule
- Affiche info trajet
- Recherche un trajet
- Apprécier un trajet
- Affiche appréciation

Trajet Section:

- Je fais le trajet une fois
- Je fais le trajet régulièrement

Itineraire * Section:

- Point de depart: Exemple: Paris, Lyon
- Point d'arrivee: Exemple: Troyes, Lile

Date et Horaire * Section:

- Date aller:
 - jj/mm/aaaa
 - : --
- Nombre de places disponibles:
 - 1 ▼
- Prix/personne:
 -
- Valider

3. Préparer un trajet

Si le conducteur veut préparer le trajet, il faut d'abord afficher info trajet. Et choisir un numéro de trajet pour manipuler. Il y a trois boutons à choisir, chaque bouton correspond sa fonction. Ici, on utilise \$_post pour lire le numéro de trajet à manipuler.

0 [Log-out](#)

Affiche info prive
Proposer un trajet
Ajouter un vehicule
Affiche info vehicule
Affiche info trajet
Recherche un trajet
Apprécier un trajet
Affiche appréciation

Trajet Inscrit

Trajet_ID	Depart	Arrive	Date	Time	PT	Prix	PR	PD	Statut
1	Troyes	Paris	2015-07-06	16:00	4	11	4	0	effectue_par_client

Trajet Propose

Trajet_ID	Depart	Arrive	Date	Time	PT	Prix	PR	PD	Statut
1	Troyes	Paris	2015-07-06	16:00	4	11	4	0	effectue_par_client

Passager	Place_reserve	Message
MONICA Geller	3	Contacter
BUFFAY Phoebe	1	Contacter

Trajet_ID	Depart	Arrive	Date	Time	PT	Prix	PR	PD	Statut
2	Troyes	Reimis	2015-07-13	08:00	3	10	0	3	non_effectue

Passager	Place_reserve	Message
MONICA Geller	3	Contacter

Trajet_ID	Depart	Arrive	Date	Time	PT	Prix	PR	PD	Statut
3	Troyes	Lille	2015-07-18	13:00	4	17	3	1	effectue_par_client

Passager	Place_reserve	Message
MONICA Geller	3	Contacter

Choisissez un trajet: 1 ▼

*Si vous supprimez un trajet pour lequel passagers sont inscrits
Vous serez pénalisé et devrez payer à chaque passager une pénalité de 10 €.

Quand il a cliqué « préparer », le système de messagerie va envoyer des messages à informer des passagers.

4 [Log-out](#)

Affiche info prive
Proposer un trajet
Ajouter un vehicule
Affiche info vehicule
Affiche info trajet
Recherche un trajet
Apprécier un trajet
Affiche appréciation

Envoyeur	Message
	13-06-2015 14:40:44 On vous informe que votre trajet de Troyes a Paris le 2015-07-06 a 16:00 est pret
	13-06-2015 14:40:51 On vous informe que votre trajet de Troyes a Lille le 2015-07-18 a 13:00 est pret
Green RACHEL	13-06-2015 14:43:15 On vous informe que votre trajet de Troyes à Lille le 2015-07-18 à 13:00 est supprimé
	13-06-2015 14:45:22 On vous informe que votre trajet de Troyes à Paris le 2015-07-06 à 16:00 est effectué

4. Supprimer un trajet

Le principe est la pareille que préparer un trajet, mais quand il supprime le trajet, il va être puni de rembourse des passagers.

4 [Log-out](#)

Affiche info prive
Proposer un trajet
Ajouter un vehicule
Affiche info vehicule
Affiche info trajet
Recherche un trajet
Apprécier un trajet
Affiche appréciation

Envoyeur	Message
	13-06-2015 14:40:44 On vous informe que votre trajet de Troyes a Paris le 2015-07-06 a 16:00 est pret
	13-06-2015 14:40:51 On vous informe que votre trajet de Troyes a Lille le 2015-07-18 a 13:00 est pret
Green RACHEL	13-06-2015 14:43:15 On vous informe que votre trajet de Troyes à Lille le 2015-07-18 à 13:00 est supprimé
	13-06-2015 14:45:22 On vous informe que votre trajet de Troyes à Paris le 2015-07-06 à 16:00 est effectué

0 [Log-out](#)

Affiche info prive
Proposer un trajet
Ajouter un vehicule
Affiche info vehicule
Affiche info trajet
Recherche un trajet
Apprécier un trajet
Affiche appréciation

Bonjour RACHEL Green

Modifier

Informations personnelles

- Civilité:homme
- Nom:RACHEL
- Prénom:Green
- E-mail Adresse:green.rachel@utt.fr
- Année de naissance:1980
- Compte solde:970€**
- Appreciation moyenne:0

Le compte original est 1000 euro, à cause de la remboursement, il est réduit 30 euro pour 3 places.

5. Valider un trajet

Le principe est la pareille que préparer un trajet, mais quand il valide le trajet, il va obtenir de l'argent des passagers. Et en plus, du fait de la statue de trajet modifiée comme « effectué », des passagers ne peuvent pas le trouver.

The screenshot shows the BlaBlaCar website interface. On the left is a sidebar with navigation links. On the right, a list of messages is displayed under the header 'Envoyeur'. The messages are from 'Green RACHEL'. The last message, dated '13-06-2015 14:45:22', states 'On vous informe que votre trajet de Troyes à Paris le 2015-07-06 à 16:00 est effectué' and is circled in red.


Envoyeur	Date	Message
Green RACHEL	13-06-2015 14:40:44	On vous informe que votre trajet de Troyes a Paris le 2015-07-06 a 16:00 est pret
	13-06-2015 14:40:51	On vous informe que votre trajet de Troyes a Lille le 2015-07-18 a 13:00 est pret
	13-06-2015 14:43:15	On vous informe que votre trajet de Troyes à Lille le 2015-07-18 à 13:00 est supprimé
	13-06-2015 14:45:22	On vous informe que votre trajet de Troyes à Paris le 2015-07-06 à 16:00 est effectué

e. Fonctionnalité pour l'administrateur

Pour le compte d'administrateur, c'est facile de réaliser sa fonction. Seulement connecter la base de données pour lire des informations. C'est important de faire l'interaction entre la page PHP et la base de données.

1. Afficher tous les comptes

Quand l'administrateur clique le bouton « affiche info compte », l'écran à droite va présenter tous comptes n'importe qui s'est inscrit, le passager et le conducteur.


Vous êtes un administrateur.

Affiche info compte
Affiche info trajet

Compte List

UserID	Login	Mot_de_password	Nom	Prenom	Sexe	Email	Naissance	VehiculeID	Marque	Place	Annee_service
1	rachelgreen	secret	RACHEL	Green	homme	green.rachel@utt.fr	1980	1	Renault	5	2014
2	monicageller	secret	MONICA	Geller	femme	geller.monica@utt.fr	1982	Pas de vehicule definie			
3	buffayphoebe	secret	BUFFAY	Phoebe	femme	phoebe.buffay@utt.fr	1984	Pas de vehicule definie			

2. Afficher info trajet

Quand l'administrateur clique le bouton « affiche info trajet », l'écran à droite va présenter toutes les informations de trajet.


Vous êtes un administrateur.

Affiche info compte
Affiche info trajet

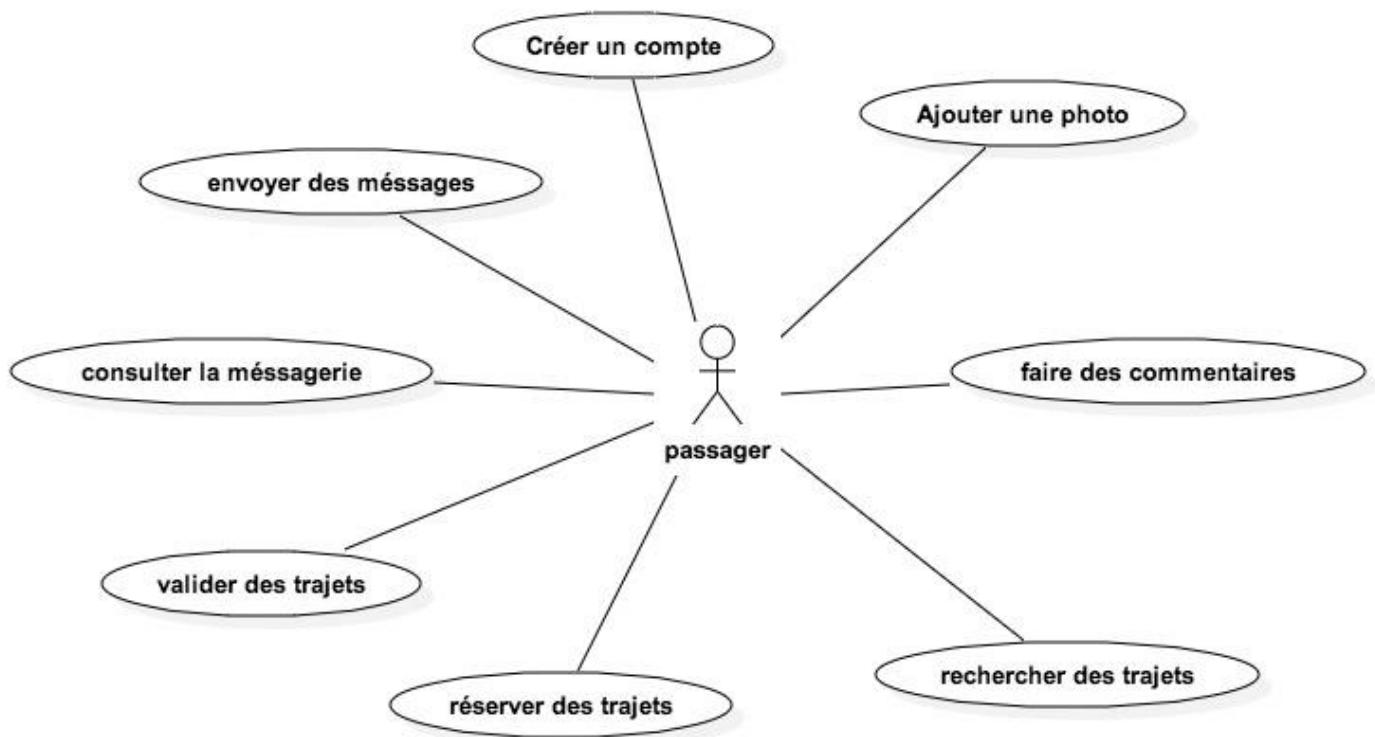
Trajet List

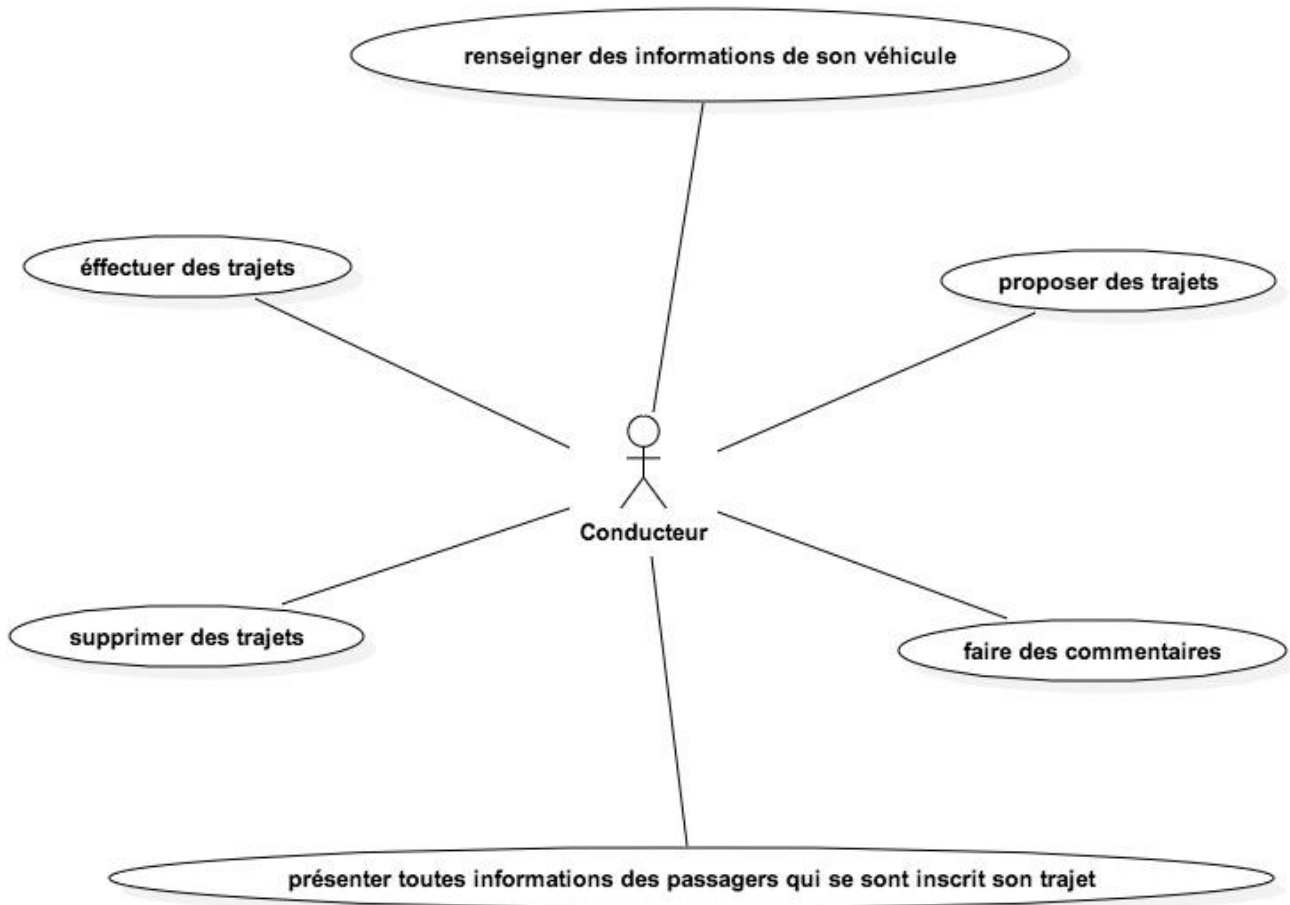
TrajetID	Ville_Depart	Ville_Arrive	Date_depart	Time_Depart	Prix	Nb_passager	Nb_reste	Status	Conducteur
2	Troyes	Reimis	2015-07-13	08:00	10	0	3	preparation	RACHELGreen

Passager	Place_reserve
----------	---------------

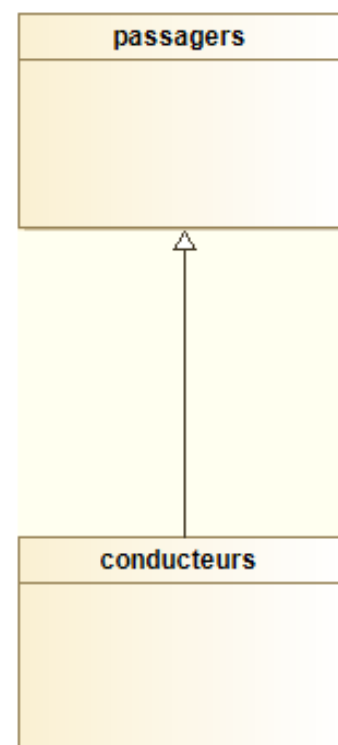
ETAPE DE CONCEPTION DU SITE

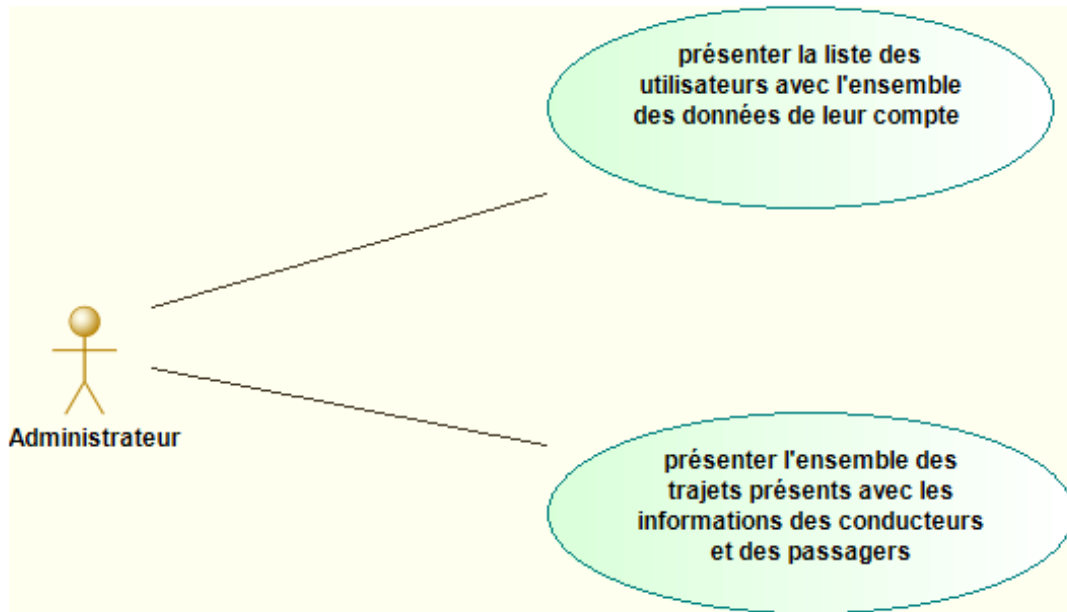
a. Diagramme de cas d'utilisation





Pour notre projet, quand le passager soumet des informations de sa voiture, il va devenir un conducteur. Donc, un conducteur est aussi un passager. C'est-à-dire, comme un conducteur, il peut aussi réserver et valider des trajets que d'autres conducteurs proposent.



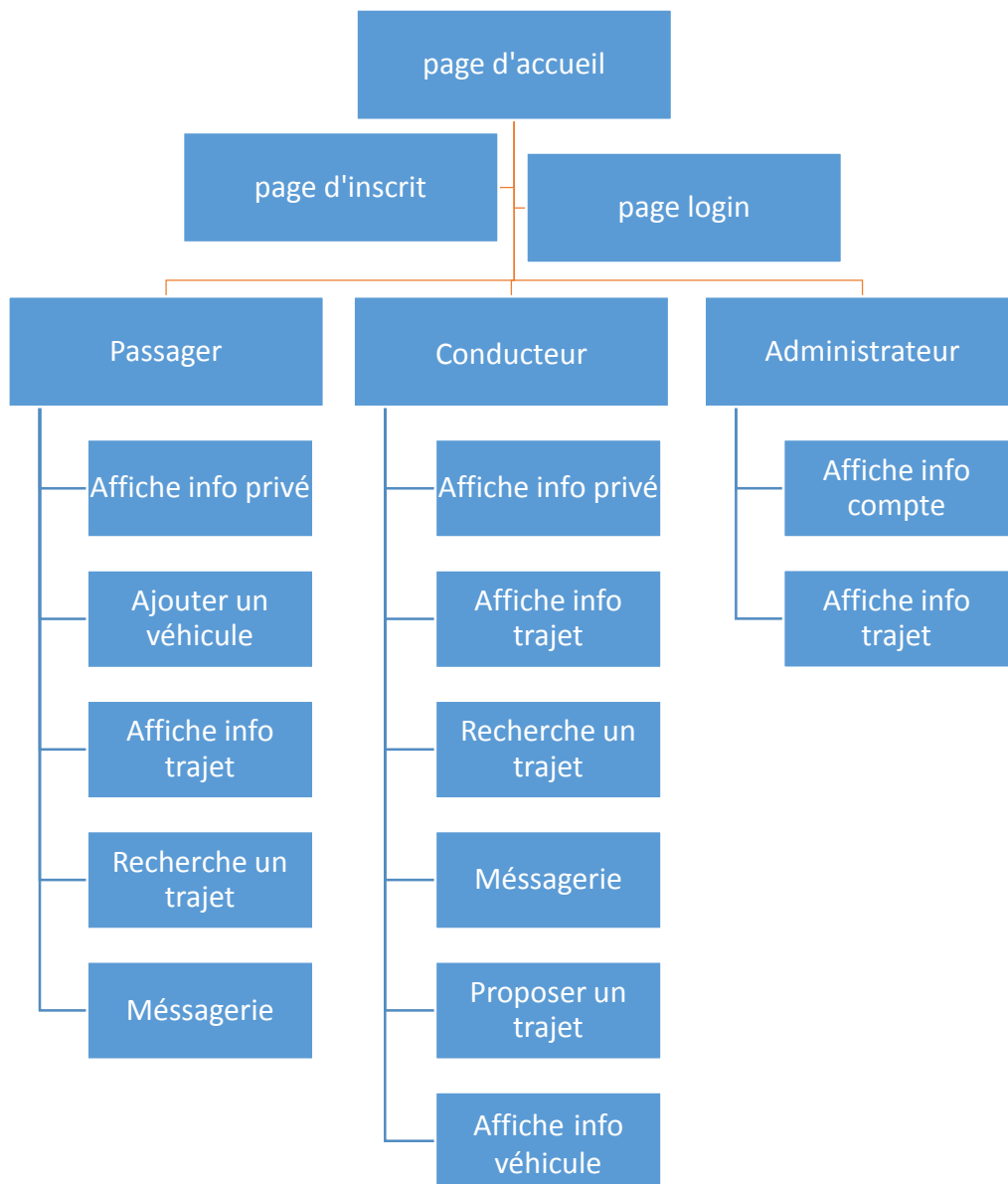


b. Conception du site

Arborescence du site

Quand on ouvre le site, on a d'accord de page d'accueil. Dans ce page, on peut accéder à la page d'inscrit. Après avoir un compte, c'est-à-dire, un utilisateur créé, on peut se connecter dans la page login. Pour notre projet, la page de passager et la page de conducteur est la pareille. Mais pour le passager, il n'a pas de droit à proposer des trajets et il ne peut pas consulter des infos de véhicule. Une fois que le passager a ajouté sa voiture dans la base de données, il devient un conducteur. Comme un conducteur, il peut faire toutes les actions quoi un passager peut sauf ajouter un véhicule. En plus, il peut proposer des trajets et consulter des infos de véhicule.

Excepté les comptes de passager et de conducteur, on a aussi un compte d'administrateur fixé qui peut consulter toutes les informations des trajets et toutes les informations des comptes créés dans la base de données. Le diagramme d'arborescence du site est ci-dessous:



c. Outils utilisés

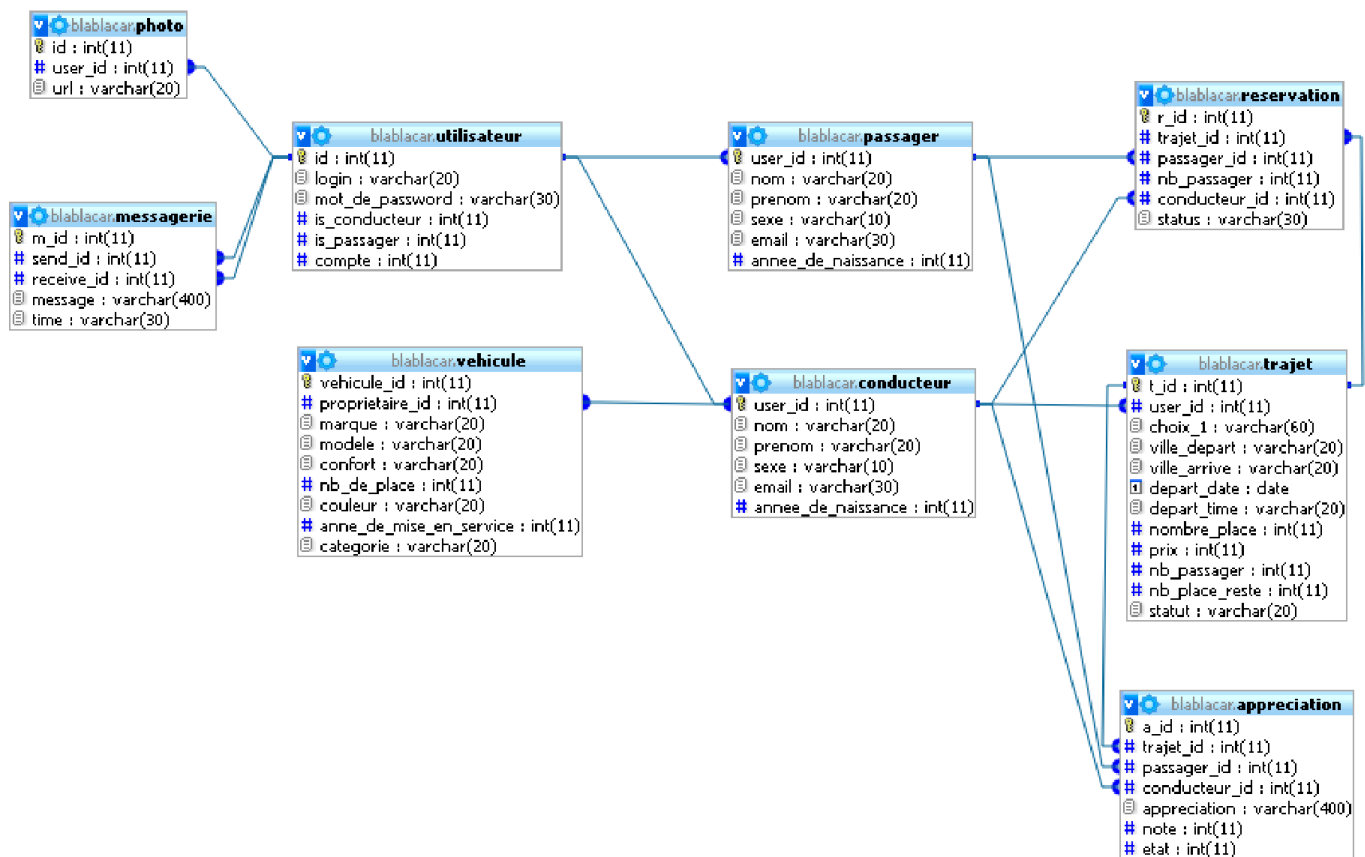
Nous créons ce projet en utilisant les éléments :

- A. Le serveur Apache
- B. Les scripts PHP
- C. Les formats HTML
- D. Les styles CSS
- E. Les scripts JavaScript
- F. La base de données MySQL et les instructions SQL
- G. L'utilisation de variables de session pour l'authentification des utilisateurs

ETAPE DE CONCEPTION DE LA BASE DE DONNEES

a. Schéma de la base de données

Établissement de la base de données :



b. Modèle physique des données

La TABLE utilisateur :

#	Nom	Type	Interclassement	Attributs	Null	Défaut	Extra
1	<u>id</u>	int(11)			Non	Aucune	AUTO_INCREMENT
2	login	varchar(20)	latin1_swedish_ci		Non	Aucune	
3	mot_de_password	varchar(30)	latin1_swedish_ci		Non	Aucune	
4	is_conducteur	int(11)			Non	Aucune	
5	is_passager	int(11)			Non	Aucune	
6	compte	int(11)			Non	Aucune	

La TABLE passager :

#	Nom	Type	Interclassement	Attributs	Null	Défaut	Extra
1	<u>user_id</u>	int(11)			Non	Aucune	
2	nom	varchar(20)	latin1_swedish_ci		Non	Aucune	
3	prenom	varchar(20)	latin1_swedish_ci		Non	Aucune	
4	sexe	varchar(10)	latin1_swedish_ci		Non	Aucune	
5	email	varchar(30)	latin1_swedish_ci		Non	Aucune	
6	annee_de_naissance	int(11)			Non	Aucune	

La TABLE conducteur :

#	Nom	Type	Interclassement	Attributs	Null	Défaut	Extra
1	<u>user_id</u>	int(11)			Non	Aucune	
2	nom	varchar(20)	latin1_swedish_ci		Non	Aucune	
3	prenom	varchar(20)	latin1_swedish_ci		Non	Aucune	
4	sexe	varchar(10)	latin1_swedish_ci		Non	Aucune	
5	email	varchar(30)	latin1_swedish_ci		Non	Aucune	
6	annee_de_naissance	int(11)			Non	Aucune	

La TABLE véhicule :

#	Nom	Type	Interclassement	Attributs	Null	Défaut	Extra
1	<u>vehicule_id</u>	int(11)			Non	Aucune	AUTO_INCREMENT
2	proprietaire_id	int(11)			Non	Aucune	
3	marque	varchar(20)	latin1_swedish_ci		Non	Aucune	
4	modele	varchar(20)	latin1_swedish_ci		Non	Aucune	
5	confort	varchar(20)	latin1_swedish_ci		Non	Aucune	
6	nb_de_place	int(11)			Non	Aucune	
7	couleur	varchar(20)	latin1_swedish_ci		Non	Aucune	
8	anne_de_mise_en_service	int(11)			Non	Aucune	
9	categorie	varchar(20)	latin1_swedish_ci		Non	Aucune	

La TABLE trajet :

#	Nom	Type	Interclassement	Attributs	Null	Défaut	Extra
1	<u>t_id</u>	int(11)			Non	Aucune	AUTO_INCREMENT
2	user_id	int(11)			Non	Aucune	
3	choix_1	varchar(60)	latin1_swedish_ci		Non	Aucune	
4	ville_depart	varchar(20)	latin1_swedish_ci		Non	Aucune	
5	ville_arrive	varchar(20)	latin1_swedish_ci		Non	Aucune	
6	depart_date	date			Non	Aucune	
7	depart_time	varchar(20)	latin1_swedish_ci		Non	Aucune	
8	nombre_place	int(11)			Non	Aucune	
9	prix	int(11)			Non	Aucune	
10	nb_passager	int(11)			Non	Aucune	
11	nb_place_reste	int(11)			Non	Aucune	
12	statut	varchar(20)	latin1_swedish_ci		Non	Aucune	

La TABLE photo :

#	Nom	Type	Interclassement	Attributs	Null	Défaut	Extra
1	<u>id</u>	int(11)			Non	Aucune	AUTO_INCREMENT
2	user_id	int(11)			Non	Aucune	
3	url	varchar(20)	latin1_swedish_ci		Non	Aucune	

La TABLE réservation :

#	Nom	Type	Interclassement	Attributs	Null	Défaut	Extra
1	<u>r_id</u>	int(11)			Non	Aucune	AUTO_INCREMENT
2	trajet_id	int(11)			Non	Aucune	
3	passager_id	int(11)			Non	Aucune	
4	nb_passager	int(11)			Non	Aucune	
5	conducteur_id	int(11)			Non	Aucune	
6	status	varchar(30)	latin1_swedish_ci		Non	Aucune	

La TABLE messagerie :

#	Nom	Type	Interclassement	Attributs	Null	Défaut	Extra
1	<u>m_id</u>	int(11)			Non	<i>Aucune</i>	AUTO_INCREMENT
2	send_id	int(11)			Non	<i>Aucune</i>	
3	receive_id	int(11)			Non	<i>Aucune</i>	
4	message	varchar(400)	latin1_swedish_ci		Non	<i>Aucune</i>	
5	time	varchar(30)	latin1_swedish_ci		Non	<i>Aucune</i>	

La TABLE appréciation :

#	Nom	Type	Interclassement	Attributs	Null	Défaut	Extra
1	<u>a_id</u>	int(11)			Non	<i>Aucune</i>	AUTO_INCREMENT
2	trajet_id	int(11)			Non	<i>Aucune</i>	
3	passager_id	int(11)			Non	<i>Aucune</i>	
4	conducteur_id	int(11)			Non	<i>Aucune</i>	
5	appreciation	varchar(400)	latin1_swedish_ci		Non	<i>Aucune</i>	
6	note	int(11)			Non	<i>Aucune</i>	
7	etat	int(11)			Non	<i>Aucune</i>	

CONCLUSION

Après un mois de développement, on a finalement réussi à concevoir et réaliser le site dynamique de covoiturage qui s'appelle « Blablacar » aussi. Le serveur Apache nous a permis de réaliser aisément l'interaction entre des pages de PHP et la base de données. En utilisant les langues HTML et CSS, le site est bien embelli. On a utilisé la session d'identification à mémoriser le login d'utilisateur pour lui attribuer le droit d'accéder des fonctionnalités.

Les points clés pour nous, premièrement, c'est l'interaction entre la page PHP et la base de données. Deuxièmement, c'est de concevoir et établir la base de données du fait qu'il faut réfléchir des relations permis le passager, le conducteur et le trajet. Quand on développait le site, on a aussi rencontré quelques problèmes :

1. L'examen de mot de passe confirmé : quand on est arrivé à l'inscription, on ne savait pas comment lire et comparer le mot de passe et le mot de passe confirmé. Après avoir appris la langue JavaScript, c'était facile de faire ça en utilisant l'élément DOCUMENT.
2. Le système de messagerie interne : l'utilisateur ne peut pas présenter des messages quand il clique le nom dans la table de réception. On l'a examiné et a essayé de modifier des codes plusieurs fois, mais ça faisait rien. Finalement, on a trouvé l'erreur, ce qu'on a eu ajouté le label « script » dans l'attribut « onclick ».

Même si on a déjà bien effectué toutes les fonctionnalités, on peut encore faire des améliorations pour notre site :

- On peut utiliser le patron MVC de construire ce sujet plus claire et mieux organisé.
- Un conducteur peut disposer de plusieurs véhicules et modifier des informations des véhicules n'importe quand.
- Lorsque le conducteur propose des trajets, il peut aussi offrir des escales (des villes passées) entre la ville de départ et la ville d'arrivé.

- Ajouter un plan dynamique dans le site qui va permettre aux utilisateurs de saisir des trajets plus clairement.

Ce projet nous a permis d'apprendre beaucoup sur la technologie du web comme HTML, PHP, PHP5, CSS, Javascript, donc il est un très bon cas de nous permettre de pratiquer toutes nos connaissances sur les langages du web.

BIBLIOGRAPHIE

- <http://www.w3school.com/>
- <http://www.cssflow.com/snippets/dark-login-form/> CSS de page_login est original d'auteur Thibaut Courouble

ANNEXE

Script de création de la base de données

■ Créer la TABLE utilisateur

```
create table utilisateur (  
    id INTEGER NOT NULL AUTO_INCREMENT,  
    login varchar(20) NOT NULL,  
    mot_de_password varchar(30) NOT NULL,  
    is_conducteur INTEGER NOT NULL,  
    is_passager INTEGER NOT NULL,  
    compte INTEGER DEFAULT 1000,  
    PRIMARY KEY(id)  
);
```

■ Créer la TABLE passager

```
create table passager (  
    user_id INTEGER NOT NULL,  
    nom varchar(20) NOT NULL,  
    prenom varchar(20) NOT NULL,  
    sexe varchar(10) NOT NULL,  
    email varchar(30) NOT NULL,  
    annee_de_naissance INTEGER NOT NULL,  
    PRIMARY KEY(user_id)  
);
```

■ Créer la TABLE conducteur

```
create table conducteur (  
    user_id INTEGER NOT NULL,  
    nom varchar(20) NOT NULL,  
    prenom varchar(20) NOT NULL,  
    sexe varchar(10) NOT NULL,  
    email varchar(30) NOT NULL,
```

```
annee_de_naissance INTEGER NOT NULL,  
PRIMARY KEY(user_id)  
);
```

■ Créer la TABLE photo

```
create table photo (  
    id INTEGER NOT NULL AUTO_INCREMENT,  
    user_id INTEGER NOT NULL,  
    url varchar(20) NOT NULL,  
    PRIMARY KEY(id)  
);
```

■ Créer la TABLE véhicule

```
create table vehicule (  
    vehicule_id INTEGER NOT NULL AUTO_INCREMENT,  
    proprietaire_id INTEGER NOT NULL,  
    marque varchar(20) NOT NULL,  
    modele varchar(20) NOT NULL,  
    confort varchar(20) NOT NULL,  
    nb_de_place INTEGER NOT NULL,  
    couleur varchar(20) NOT NULL,  
    annee_de_mise_en_service INTEGER NOT NULL,  
    categorie varchar(20) NOT NULL,  
    PRIMARY KEY(vehicule_id)  
);
```

■ Créer la TABLE trajet

```
create table trajet (  
    t_id integer not null auto_increment,  
    user_id integer not null,  
    choix_1 varchar(60) not null,  
    ville_depart varchar(20) not null,  
    ville_arrive varchar(20) not null,  
    depart_date date not null,  
    depart_time varchar(20) not null,  
    nombre_place integer not null,  
    prix integer not null,  
    nb_passager integer not null,
```

```
        nb_place_reste integer not null,  
        statut varchar(20) not null,  
        primary key(t_id)  
    );
```

■ Créer la TABLE réservation

```
create table reservation (  
    r_id INTEGER NOT NULL AUTO_INCREMENT,  
    trajet_id integer not null,  
    passager_id integer not null,  
    nb_passager integer not null,  
    conducteur_id integer not null,  
    status varchar(30) not null,  
    PRIMARY KEY(r_id)  
);
```

■ Créer la TABLE messagerie

```
Create table messagerie(  
    m_id integer AUTO_INCREMENT,  
    send_id integer NOT NULL,  
    receive_id integer NOT NULL,  
    message varchar(400) NOT NULL,  
    time varchar(30) NOT NULL,  
    PRIMARY KEY(m_id)  
);
```

■ Créer la TABLE appréciation

```
Create table appreciation(  
    a_id integer AUTO_INCREMENT,  
    trajet_id integer NOT NULL,  
    passager_id integer NOT NULL,  
    conducteur_id integer NOT NULL,  
    appreciation varchar(400) NOT NULL,  
    note integer NOT NULL,  
    etat integer NOT NULL,  
    PRIMARY KEY(a_id)  
);
```

■ Contrainte pour la TABLE trajet

```
alter table trajet add constraint foreign key(user_id) references conducteur(user_id);
```

■ Contrainte pour la TABLE réservation

```
alter table reservation add constraint foreign key(conducteur_id) references  
conducteur(user_id);
```

```
alter table reservation add constraint foreign key(passager_id) references  
passager(user_id);
```

```
alter table reservation add constraint foreign key(trajet_id) references trajet(t_id);
```

■ Contrainte pour la TABLE messagerie

```
alter table messagerie add constraint foreign key(send_id) references utilisateur(id);
```

```
alter table messagerie add constraint foreign key(receive_id) references  
utilisateur(id);
```

■ Contrainte pour la TABLE photo

```
alter table photo add constraint foreign key(user_id) references passager(user_id);
```

■ Contrainte pour la TABLE véhicule

```
alter table vehicule add constraint foreign key(proprietaire_id) references  
conducteur(user_id);
```

■ Contrainte pour la TABLE appréciation

```
alter table appreciation add constraint foreign key(passager_id) references  
passager(user_id);
```

```
alter table appreciation add constraint foreign key(conducteur_id) references  
conducteur(user_id);
```

```
alter table appreciation add constraint foreign key(trajet_id) references trajet(t_id);
```