

Mini projet

L'université de Bab-Ezzouar sollicite votre compétence en orientée objet afin de lui concevoir une application de covoiturage permettant de réduire au mieux les difficultés de transport pour les étudiants, les enseignants et le personnels ATS

Le but principal autour de cette application est de permettre aux individus au sein de l'université de s'entre aider afin de réduire au mieux les problèmes de transport à travers le partage de courses sur le trajet habituel et quotidien pour la majorité d'entre eux.

Les utilisateurs de cette application peuvent être des Etudiants, des Enseignants ou appartenant au corps administratif communément notée ATS. Chaque utilisateur est caractérisé par son nom, son prénom, son matricule, et sa réputation au sein de l'université (qui sera détaillée plus loin).

Un Etudiant est en plus caractérisé par son année d'admission à l'université, sa faculté et sa spécialité.

Un enseignant est en plus caractérisé par son année de recrutement, sa faculté

Un ATS est en plus caractérisé par son année de recrutement son service de rattachement

Chaque utilisateur défini dans son profil :

- Son statut ; Chaque utilisateur peut être à un instant donné ; soit « Passager » d'une course soit « Chauffeur ». Un Passager est un individu qui souhaite trouver une place avec un Chauffeur qui partage son itinéraire tandis qu'un Chauffeur est un individu qui propose une ou plusieurs place dans son itinéraire
- Son itinéraire (Un seul point de dépôt ou ramassage s'il s'agit d'un demandeur et un ou plusieurs point de dépôt ou ramassage dans le cas d'un offre de courses).
- Ses préférences durant la course (Exemple filles ou garçons uniquement ou bien indifférant, avec ou sans musique, avec ou sans bagages supplémentaires,etc),
- Ses horaires de disponibilités qui peuvent être journalier, hebdomadaires ou bien quotidien.
- Son type de course en « aller et retour » ou bien « aller simple » ou « retour simple » uniquement.

Lorsqu'un utilisateur demande une course, l'application devra lui fournir les chauffeurs disponibles en fonction de l'adéquation entre son itinéraire et ceux des chauffeurs, ainsi que ses préférences et ceux des chauffeurs.

A l'issu de la course, les passagers sont amenés à noter le chauffeur avec une évaluation numérique sur 5 et un commentaire qui vont servir à définir sa réputation dans toute l'application par la suite. Le chauffeur devra également faire de même pour tous les passagers d'une course donnée. Ces annotations permettront à l'administration de veiller continuellement au bon déroulement de l'opération.

L'administration souhaite également :

- Gérer l'ensemble des utilisateurs de l'application qu'ils soient des étudiants, des enseignants ou des ATS
- Visualiser les courses en cours à un instant donnée,
- Visualiser le planning journalier, hebdomadaire des courses prévues.
- Visualiser l'historique des courses passées.
- Recevoir les statistiques d'utilisation de l'application : Exemple, Nombre d'étudiants, nombre d'enseignants, nombre d'utilisateurs actifs, quelles catégories proposent plus de courses ? Quelles facultés ? top 10 des chauffeurs ? pire chauffeurs/ passagers à bannir
- Bannir les utilisateurs qui ont acquis une mauvaise réputation à travers l'application

Objectifs :

- Travailler en équipe de 3 à 4 étudiants
- Concevoir en oriente objet cette description du monde réelle.
- Concrétiser (implémenter) en JAVA votre conception
- Remettre un rapport qui décrit et justifie votre conception
- Remettre le code source de votre implémentation

Remarques générales :

1. Le rapport et le code source devront être déposée sur la classroom au plus tard le vendredi 09 mai 23h59.
2. Aucun travail remis par email avant ou après ce délai ne sera pris en compte par l'assistante de TP ou moi-même.

3. Une séance de démonstration sera organiser après les examens et vous serez informer du déroulement de cette étape ultérieurement.
4. Le travail se fait en équipe mais vous devez vous partager les tâches et toute tâche dans le projet doit être justifie avec le nom de son auteur dans le rapport et également dans le code
5. Le rapport contient les détails de conception en termes de classes, encapsulation, types, attributs, signature des méthodes Etc
6. Aucune interfaces graphiques n'est demande ni même une base de données et si faites elles ne seront pas prise en compte dans la note.