

NF18: Société Vehicle Share

Note de clarification

Groupe 4

Responsable: Alessandro Victorino

Sujet:

(disponible sur le site internet)

“La société vehicleShare propose un site internet qui permet de gérer la location de véhicules entre particuliers en libre-service. On souhaite mettre en place une application web pour gérer son système d'information (parc automobile, réservation en ligne, les commentaires des clients,...).

Un client de vehicleShare a un nom, un prénom, un âge, une photo, un pseudo et des coordonnées. Un client peut être un propriétaire et/ou un locataire. Un locataire doit fournir, en plus des informations précédentes, une copie de son permis. Ce dernier doit être valide pendant la période de la location.

Le site propose également la location de véhicules pour les entreprises. Dans ce cas-là, des informations supplémentaires sur l'entreprise doivent être fournies ainsi que la liste des conducteurs. Une entreprise peut louer donc plusieurs véhicules en même temps.

Un véhicule a un numéro d'immatriculation, catégorie, marque, modèle, couleur, carburant, options, l'année de mise en circulation, le kilométrage et le niveau du carburant ainsi qu'une brève description. Le propriétaire précise le secteur dans lequel le véhicule peut circuler (liste des pays). Un propriétaire peut préciser la période, voire les périodes, durant laquelle sa voiture sera disponible à la location. Un contrat d'assurance est associé avec chaque véhicule afin de le protéger en cas de dommage causé par le locataire ou par un tiers au cours de la location. La location d'une voiture est caractérisée par un contrat de location entre le locataire et le propriétaire. Le locataire doit effectuer un état des lieux de la voiture en prenant quelques photos et signale le moindre dégât sur l'état du véhicule en remplissant une check-list. En cas d'accident, le locataire paye les coûts de réparation dans la limite de la franchise. Plusieurs options de réduction de franchise sont proposées aux locataires au moment de la réservation : sans réduction, franchise réduite et Zéro franchise.

Au moment de check-out, le propriétaire effectue à son tour un état des lieux et signale tous dégâts constatés afin de pouvoir réclamer des dédommagements.

Un contrat de location donne lieu à une facture. Il y est inscrit le kilométrage, la date et le moyen de paiement. En plus, pour chaque durée de location, il y a un seuil de kilométrage. Si ce seuil est dépassé, le kilomètre est facturé à x€. Également, si le véhicule revient avec moins de carburant qu'au départ, le litre est facturé à y€.

Après la fin de la location, le locataire a la possibilité de donner son avis sur le véhicule utilisé et peut attribuer une note entre 0 et 5. Un locataire peut également signaler un véhicule. Si un véhicule est signalé plusieurs fois (03 fois), l'annonce du véhicule sera automatiquement désactivée dans le site. La réactivation de l'annonce nécessite l'intervention du responsable qualité.”

Besoins

Le site de la société devrait assurer les fonctionnalités suivantes (liste non exhaustive) :

- Ajouter/modifier/supprimer une annonce de véhicule.
- Consulter les annonces de véhicules et faire des recherches personnalisées selon plusieurs critères.
- Commenter, noter et signaler des véhicules
- Réserver une voiture

- Récupérer des statistiques à partir du parc automobile : moyenne d'âge, kilométrage, les catégories les plus utilisées...

Note de clarification :

Légende :

gras : clés

Pour les photos, on choisit de mettre un lien vers l'url d'une photo auquel cas le type de est string

Le mot louer possède un **double-sens** : on peut mettre à disposition sa voiture (ex : un propriétaire loue sa voiture), on peut utiliser une voiture qui n'est pas à nous (un locataire loue une voiture pour les vacances).

Classes:

Client :

- Attributs : nom (string), prénom (string), âge (int), photo(string), **pseudo(string)**, téléphone(string), email(string), type(enum{professionnel,particulier})
- Méthodes:
- Associations :
 - Classe mère de Locataire et Propriétaire
 - Un client peut appartenir ou non à une entreprise
- Remarques :
 - Un client peut posséder plusieurs voitures et les mettre en location en même temps (propriétaire)
 - Un client ne peut pas utiliser plusieurs voitures empruntées en même temps mais plusieurs au cours de sa vie (locataire)
 - Un client peut appartenir à une entreprise (c'est un employé qui loue une voiture pour le compte de l'entreprise)
 - tous les attributs sont not null
 - les coordonnées sont le téléphone et l'adresse mail

Locataire :

- Attributs : permis(string), validité (date)
- Méthodes: est_une_entreprise(), est_valide()
- Associations :
 - Classe fille de Client
 - Un locataire peut soussigner à un ou plusieurs contrats de location

Propriétaire :

- Attributs :
- Méthodes:
- Associations :
 - Classe fille de Client
 - Un propriétaire possède un ou plusieurs véhicules

- Un propriétaire peut soussigner à un ou plusieurs contrats de location

Entreprise :

- Attributs : nom(string), adresse(string), ville(string), code_postal(integer)
- Méthodes:
- Associations :
 - Une entreprise possède une liste de conducteurs
 - Une entreprise possède un ou plusieurs conducteurs
 - Une entreprise est un client
- Remarques :
 - Une entreprise peut posséder plusieurs voitures et les mettre en location en même temps
 - Une entreprise peut utiliser plusieurs voitures empruntées en même temps (grâce à ses conducteurs)

Conducteur:

- Attributs : nom(string), prenom(string), age(integer), photo(string), téléphone(string), email, permis(string), validité(date)
- Méthodes:
- Associations : Un conducteur appartient à une unique entreprise

Véhicule :

- Attributs : **numéro d'immatriculation (string)**, catégorie (string), marque (string), modèle (string), couleur (string), carburant (string), année de mise en circulation (year), kilométrage (int), niveau de carburant (float), description (text)
- Méthodes: est_disponible()
- Associations :
 - Un véhicule peut circuler dans un ou plusieurs pays
 - Un véhicule est disponible pendant une ou plusieurs périodes
 - Un véhicule est soumis à un et un seul contrat d'assurance
 - Un véhicule possède 0 ou une seule annonce
 - Un véhicule appartient à un unique propriétaire
 - Un véhicule possède 0 ou plusieurs options

Option :

- Attributs : intitulé(string)
- Méthodes:
- Associations :
 - Une option est possédée par 1 ou plusieurs véhicules

Pays :

- Attributs : **nom(string)**
- Méthodes:
- Associations :
 - Dans un pays peuvent circuler plusieurs voitures

Période :

- Attributs : debut(date), fin(date)
- Méthodes:
- Associations :
 - Lors d'une période, plusieurs véhicules peuvent être disponibles

Contrat_assurance :

- Attributs : nom_assurance(string), type(string)
- Méthodes:
- Associations :
 - Un contrat d'assurance est relatif à un unique véhicule

Contrat_location :

- Attributs : option_franchise(string), seuil_kilometrage(int), début (datetime), fin (datetime),
- Méthodes:
- Associations :
 - Un contrat de location est établi avec un unique locataire
 - Un contrat de location est établi avec un unique propriétaire
 - Un contrat de location donne lieu à une unique facture
 - Un contrat de location possède deux états des lieux (un d'entrée et un de sortie)
 - enum option_franchise : "sans réduction", "franchise réduite", "Zéro franchise"
 - Un état des lieux peut donner lieu ou non à un commentaire
- Remarques :
 - Nous avons décidé de mettre dans cette classe l'attribut seuil kilométrage de manière à pouvoir comparer les kilométrages lors de l'état des lieux de sortie

Etat_des_lieux:

- Attributs : type(enum:{debut,fin}), photo(string), kilométrage(int), carburant(float), checklist(string)
- Méthodes:
- Associations :
 - Un état des lieux est relatif à un unique contrat de location
- Remarques :
 - L'attribut checklist correspond aux dégâts constatés par le locataire lors de l'état des lieux d'entrée et par le propriétaire lors de l'état des lieux de sortie
 - Les commentaires retours sont les dégâts constatés par le propriétaire pour ensuite réclamer les dédommagements
 - kilométrage indique le nombre de kilomètre effectués entre le début et la fin de la location
 - carburant indique le niveau de carburant au début et à la fin de la location

Facture :

- Attributs : date(date), kilométrage(int), carburant(float), moyen de paiement(string), montant(float)

- Méthodes: calcul_montant()
- Associations :
 - Une facture est relative à un unique contrat de location
- Remarques :
 - date indique la date de la facture
 - kilométrage indique le nombre de kilomètres effectués lors du contrat
 - carburant indique le niveau du carburant manquant que le locataire doit payer
 - le montant correspond au montant total de la facture : le prix de la location + le montant du dépassement de kilomètres + prix de l'essence manquante + prix des dégâts effectués que nous effectuons dans la méthode calcul_montant()

Commentaire:

- Attributs : note(int), signaler(bool), description(string)
- Méthodes: changer_nombre_signalements()
- Associations :
 - Un commentaire est émis suite à un unique contrat de location
- Remarques :
 - une note est entre 0 et 5

Annonce:

- Attributs : activé(bool), intitulé(string), nombre_signalements(int), note(float)
- Méthodes: moyenne(), changer_signalements()
- Associations :
 - Une annonce peut donner lieu à un ou plusieurs contrats de location
 - Une annonce est relative à un unique véhicule
- Remarques :
 - La méthode moyenne() permet de calculer la note de l'annonce suite aux différents commentaires
 - La méthode changer_signalements() permet de changer le nombre de signalements de l'annonce
 - Lorsqu'un contrat est émis, nous avons supposé que le locataire répondait à une annonce, d'où la relation entre annonce et contrat de location.

UML

Lien PlantUML:

https://www.plantuml.com/plantuml/uml/hLLDRICs4Dth58HMRV4vQBeqWY15aAuAjAjk1LeQsuFmXoM7GOMqT-asvyZ5EfPCcl9h0ocQUJUHcv t-3Xw22GzHQsQNiaGnAD2C2JU6i5As6iNej07itk574Uwg3SmLgWeOJPMx9bNe0ub4421GmsYspG4bK157Hm8DO2JDJhJyPvkuKGq1e3wfiZ1jX7XU3VHtpptfpe 6fxIH8zp8mSU8sXAFyY5Gv8FCOW2T8SWRfee0D37fp70DzzdxqQSpXqRIQxjmo3r6-EFyq5hkWg1m-14ULag5GLxEq0dRE1f1AZSh03V-xtQCqGUuBe_wzp FgH9Q75TQNdcV-0 V48pmgXrfByCIO9hlaLaCpPUeGYhgN CrP1baj1fl5tYxwgwpV HlqEbCCO0TCm7kXvz6eh97K3w4-OPbTL0IX3Hu0l8s6MLjyhAz7Q0q7jq4qM2LyeSDc1mrk16BLpssmYUPJA453EBP7QKX_1l1X0P13pQulpw09j8YqkmHLCU5TkBILB7pqT2FUHDpj7lDmStX-fQolQJegck7LbsssEfFOQoMe28gYiL_mgB9-Ash7wd4E9hwhISCsEXRvpyVkl2N1ZYmddtU-YV5GQgjVX5zXHzTXWoNPF_r4dR0vZEIGGDbP-rDLJF9XKxkpkzopttQ3KNMlvIOsanHi2TaGhuHcsiLRMzQy8_6sDp-p1_V462kK9ky8A0iNhZeJirdHWoD9-jq5sQ7VYkpWiJqURqvMj_IBdveHMrVXqs-RRctqHI81soXu-FyHzaxrhRwUassDwje8vIJedFOtLxrARkxzMvuzJ5bmjlY9D_gDiKLuTLiZ6S6G6ulDISLydfzoqRHCZDm8_FqOa9ywlsqa9_ZUIYYbZN4sNaF8G6EQWafYM9VWSEjDegrMloZCHZO5PBFFoa-6dQHCUMzaGO006bw9alTmIEQv9odzTqmgHT6td-yjtrXh6AuHuBOcpVchBFSJyjrIWGXVOvCxCRsNTzCGrbpj9cKbsrpm0MrMhVm00

UML:

@startuml

class Client {

- nom: string
- prenom: string
- age: int
- photo: string
- pseudo: string
- telephone: int
- email: string
- type : enum:{

professionnel,particulier}

}

class Locataire {

- permis: string
- valide: date
- est_une_entreprise()
- est_valide()

}

class Proprietaire {

}

class Entreprise {

- nom: string
- adresse : string

```
- ville : string  
- code_postal : integer  
}
```

```
class Conducteur {  
  - nom: string  
  - prenom: string  
  - age: int  
  - photo: string  
  - telephone: int  
  - email: string  
  - permis : string  
  - valide: date  
}
```

```
class Vehicule {  
  - immatriculation: string  
  - categorie: string  
  - marque: string  
  - modele: string  
  - couleur: string  
  - carburant: string  
  - annee_mise_circulation : year  
  - kilometrage: int  
  - niveau_carburant: float  
  - description: text  
  - est_disponible()  
}
```

```
class Option{  
  - intitule : string  
}
```

```
class Pays {  
  - nom: string  
}
```

```
class Periode {  
  - debut: date  
  - fin: date  
}
```

```
class Contrat_assurance {  
  - nom_assurance: string  
  - type: string  
}
```



```
class Contrat_location {  
  - option_franchise: string  
  - seuil_kilometrage: int  
  - debut: date  
  - fin: date  
}
```

```
class Etat_des_lieux {  
  - type : enum:{debut,fin}  
  - photo: string  
  - kilometrage: int  
  - carburant: float  
  - checklist: string  
}
```

```
class Facture {  
  - date: date  
  - kilometrage: int  
  - carburant: float  
  - moyen_paiement: string  
  - montant: float  
  - calcul_montant()  
}
```

```
class Commentaire {  
  - note: int  
  - signaler: bool  
  - description: string  
}
```

```
class Annonce {  
  - activite: bool  
  - intitule: string  
  - nombre_signalement: int  
  - note: float  
  - changer_nombre_signalements()  
}
```

```
Client "0..*" -- "0..1" Entreprise : est employé >  
Entreprise "1" -- "1..*" Conducteur : appartient <  
Client <|- Locataire  
Client <|-- Proprietaire  
Proprietaire "1" -- "1..*" Contrat_location : soussigne à >  
Proprietaire "1" - "1..*" Vehicule : est propriétaire >  
Vehicule "*" -- "*" Periode : est dispo >  
Vehicule "*" -- "*" Pays : peut circuler >
```

Vehicule "1" -- "1" Contrat_assurance : est soumis à >
 Vehicule "1..*" -- "0..*" Option : possede >
 Contrat_location "1..*" -- "1" Annonce
 Locataire "1" -- "1..*" Contrat_location : soussigne à >
 Contrat_location "1" - "1" Facture : donne lieu à >
 Contrat_location "1" -- "2" Etat_des_lieux : effectue >
 Contrat_location "1" -- "0..1" Commentaire : donne lieu à >
 Annonce "1" -- "1" Vehicule : possede <
 @enduml

Visualisation:

