

Bases de données

INFO-H303

Trottinettes

Préparation et ajustement:

BAKKALI Yahya (000445166)
FALLAHI Amirmohammad (000460073)
HAUWAERT Maxime (000461714)

(Groupe AA)

Mars/Avril/Mai 2019

UNIVERSITÉ LIBRE DE BRUXELLES (ULB)

1 Introduction

Le but du projet est de modéliser un programme qui gère un ensemble de trottinettes électriques se déplaçant dans Bruxelles. Les trottinettes peuvent être déposées sur le bord de la route. Pour réaliser cela on commence par modéliser le problème en utilisant un diagramme entité-association et une traduction relationnelle pour pouvoir implémenter et manipuler la base de données correspondante.

2 Le diagramme entité-association et ses contraintes

2.1 Le diagramme entité-association

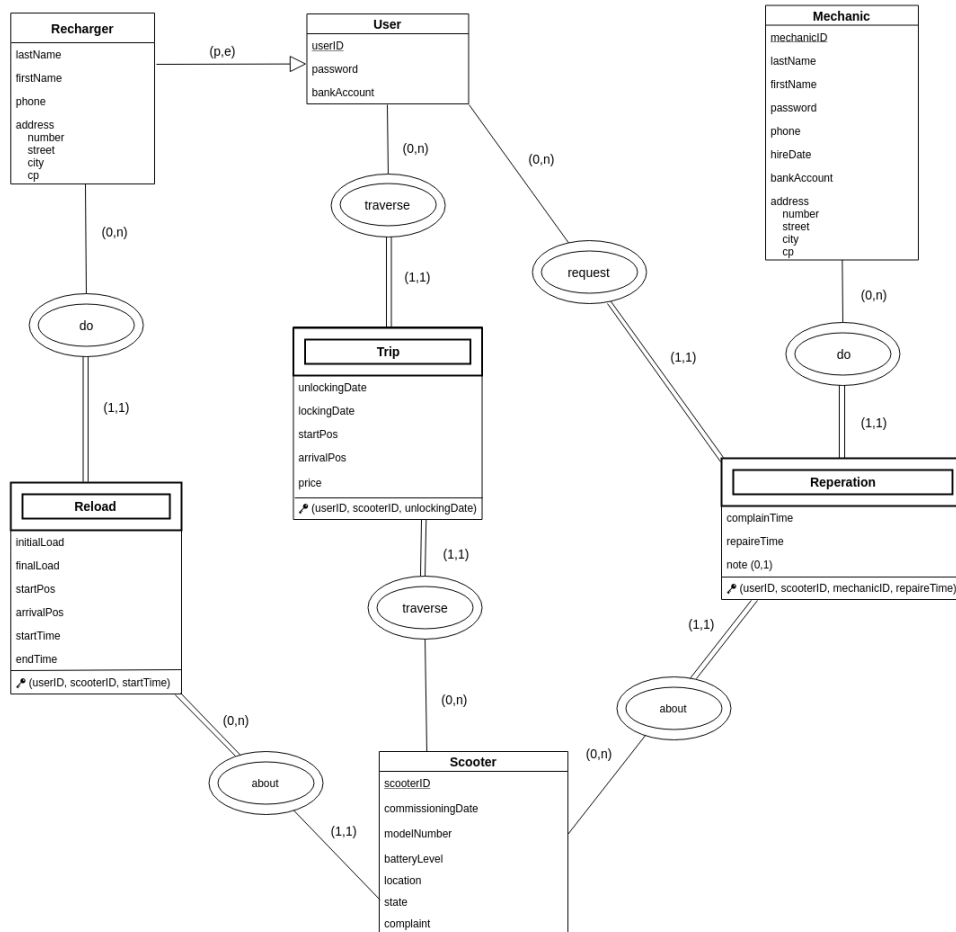


Figure 1: Le diagramme entité-association modélisant le problème

2.2 Les contraintes du diagramme entité-association

1. Pour chaque trajet, la société doit bloquer une partie de l'argent du compte de l'utilisateur en cas de dégradation ou de vol.
2. A chaque déverrouillage d'une trottinette, la société doit retirer 1€ du compte de l'utilisateur.
3. A chaque minute d'utilisation d'une trottinette, la société doit retirer 0.15€ du compte de l'utilisateur.
4. Si la durée d'un trajet dépasse une heure, le coût sera réduit à 6.5€ par heure.
5. Si le trajet dépasse un jour, le coût sera réduit à 36€ par jour.
6. L'utilisateur qui charge une trottinette doit la prendre après 22h, puis, la redéposer quelque part avant 7h du matin.
7. Pour chaque niveau de charge rechargé, l'utilisateur recevra 2€ .
8. Toutes les réparations doivent être effectuées entre 22h et 7h.
9. Dans une séance de rechargement, l'état de charge du début doit être un état plus faible que l'état de fin.
10. Le moment de déverrouillage doit être plus petit que le moment de verouillage.
11. Le moment de dépôt de la plainte doit être plus petit que le moment de réparation.

3 La traduction relationnelle et ses contraintes

3.1 Traduction du modèle entité-association vers le modèle relationnel

Scooter(scooterID, commissioningDate, modelNumber, batteryLevel, location, state, complaint)

Mechanic(mechanicID, lastName, firstName, password, phone, hireDate, bankAccount, adrsNumber, adrsStreet, adrsCity, adrsCp)

User(userID, password, bankAccount)

Recharger(userID, lastName, firstName, phone, adrsNumber, adrsStreet, adrsCity, adrsCp)
userID référence User.userID

Reperation(userID, scooterID, mechanicID, repaireTime, complainTime, repaireTime, note)
userID référence User.userID
scooterID référence Scooter.scooterID

mechanicID référence Mechanic.mechanicID

Reload(userID, scooterID, startTime, initialLoad, finalLoad, startPos, arrivalPos, startTime, endTime)

userID référence User.userID

scooterID référence Scooter.scooterID

Trip(userID, scooterID, unlockingDate, unlockingDate, lockingDate, startPos, arrivalPos, price)

userID référence User.userID

scooterID référence Scooter.scooterID

3.2 Les contraintes de la traduction relationnelle

1. $\text{Reload.initialLoad} \leq \text{Reload.finalLoad}$
2. $1 \leq \text{Scooter.batteryLevel} \leq 4$
3. $\text{Trip.unlockingDate} < \text{Trip.lockingDate}$
4. $\text{Reperation.complainTime} < \text{Reperation.repaireTime}$

4 Les hypothèses

1. Un utilisateur ne peut utiliser qu'une et une seule trottinette à la fois.
2. Une réparation ne concerne qu'une seule trottinette.