

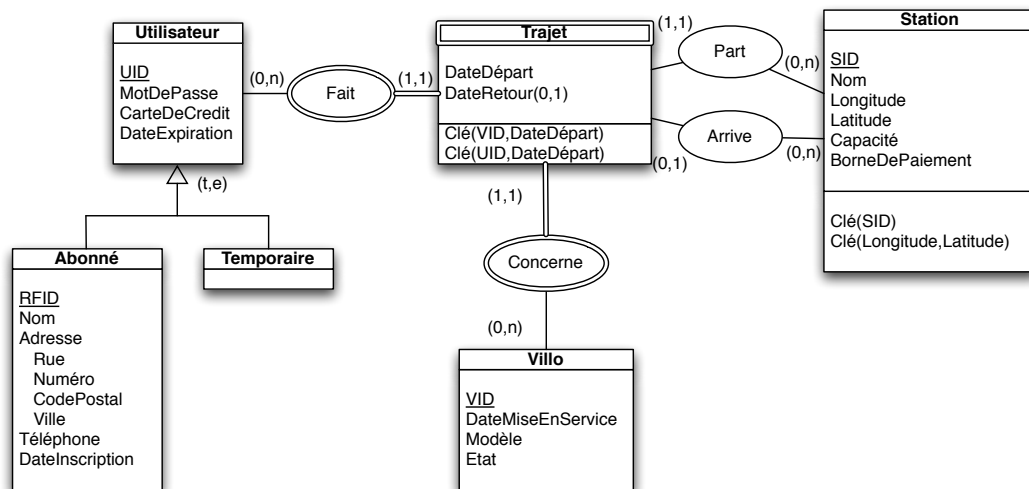
INFO-H-303 : Bases de données

Projet *Villo!* : Correction de la première partie

<http://cs.ulb.ac.be/public/teaching/infh303>

Ce document contient une solution pour la première partie du projet. Il en existe d'autres.

Modèle entité-association



Contraintes d'intégrité

- La *DateExpiration* d'un *Abonné* doit être strictement supérieure à sa *DateInscription*.
- La *DateDépart* d'un *Trajet* doit être strictement supérieure à la *DateInscription* de l'*Abonné* qui le fait.
- La *DateDépart* d'un *Trajet* doit être strictement inférieure à la *DateExpiration* de l'*Utilisateur* qui le fait.
- La *DateDépart* d'un *Trajet* doit être strictement inférieure à la *DateExpiration* de l'*Utilisateur* qui le fait.
- La *DateRetour* d'un *Trajet* doit être strictement supérieure à la *DateDépart* de ce même trajet.
- La *DateDépart* d'un *Trajet* doit être strictement supérieure à la *DateMiseEnService* du *Villo* concerné.
- A l'instant *DateRetour*, la *Station* de retour d'un *Trajet* doit contenir moins de *Villos* que sa *Capacité*. Le calcul du nombre de *Villos* présents dans une *Station* à l'aide de l'entité *Trajet*.
- Un même *Villo* ne peut pas concerner deux *Trajets* en même temps.
- Un même *Utilisateur* ne peut pas faire deux *Trajets* en même temps.
- Un *Trajet* *i* qui suit directement un autre *Trajet* *j* pour un même vélo doit avoir la même *Station* de départ que la *Station* d'arrivée du *Trajet* *j*.

Remarques

Il faut gérer la mise en service des *Villos*, c'est à dire le moment où ils arrivent dans une *Station* pour la première fois. Une solution que nous proposons dans ce modèle est la suivante :

Le premier *Trajet* d'un *Villo* sera effectué par un *Utilisateur* "bidon" (dont la *DateExpiration* est "infinie"). Ce *Trajet* aura la même *Station* de départ que celle d'arrivée et ses *DateDépart* et *DateRetour* seront égales à la *DateMiseEnService* du *Villo* concerné.

Une autre solution serait de matérialiser la *Station* de départ d'un *Villo* à l'aide d'une association entre ces deux entités. Cette solution serait plus lourde à gérer car il faudrait notamment ajouter des contraintes.

Dans les données fournies, ces *Trajets* n'ont pas d'*Utilisateur*, de *Station* de départ ni de *DateDépart*. Libre à vous de les insérer de façon à ce que vos insertions respectent votre modèle.

Modèle relationnel

Utilisateur(UID, MotDePasse, CarteDeCredit, DateExpiration)

Abonné(UID, RFID, Nom, Rue, Numéro, CodePostal, Ville, Téléphone, DateInscription)

– UID référence Utilisateur.UID

Villo(VID, DateMiseEnService, Modèle, Etat)

Station(SID, Nom, Longitude, Latitude, Capacité, BorneDePaiement)

Trajet(VID, DateDépart, UID, StationDépart, *DateRetour*, *StationRetour*)

– VID référence Villo.VID

– UID référence Utilisateur.UID

– StationDépart référence Station.SID

– StationRetour référence Station.SID

– (UID, DateDépart) est unique et donc est également une clé de cette relation

Remarque

– Un Utilisateur.UID n'existant pas dans Abonné.UID est un utilisateur temporaire.