



Rapport projet Bases de données

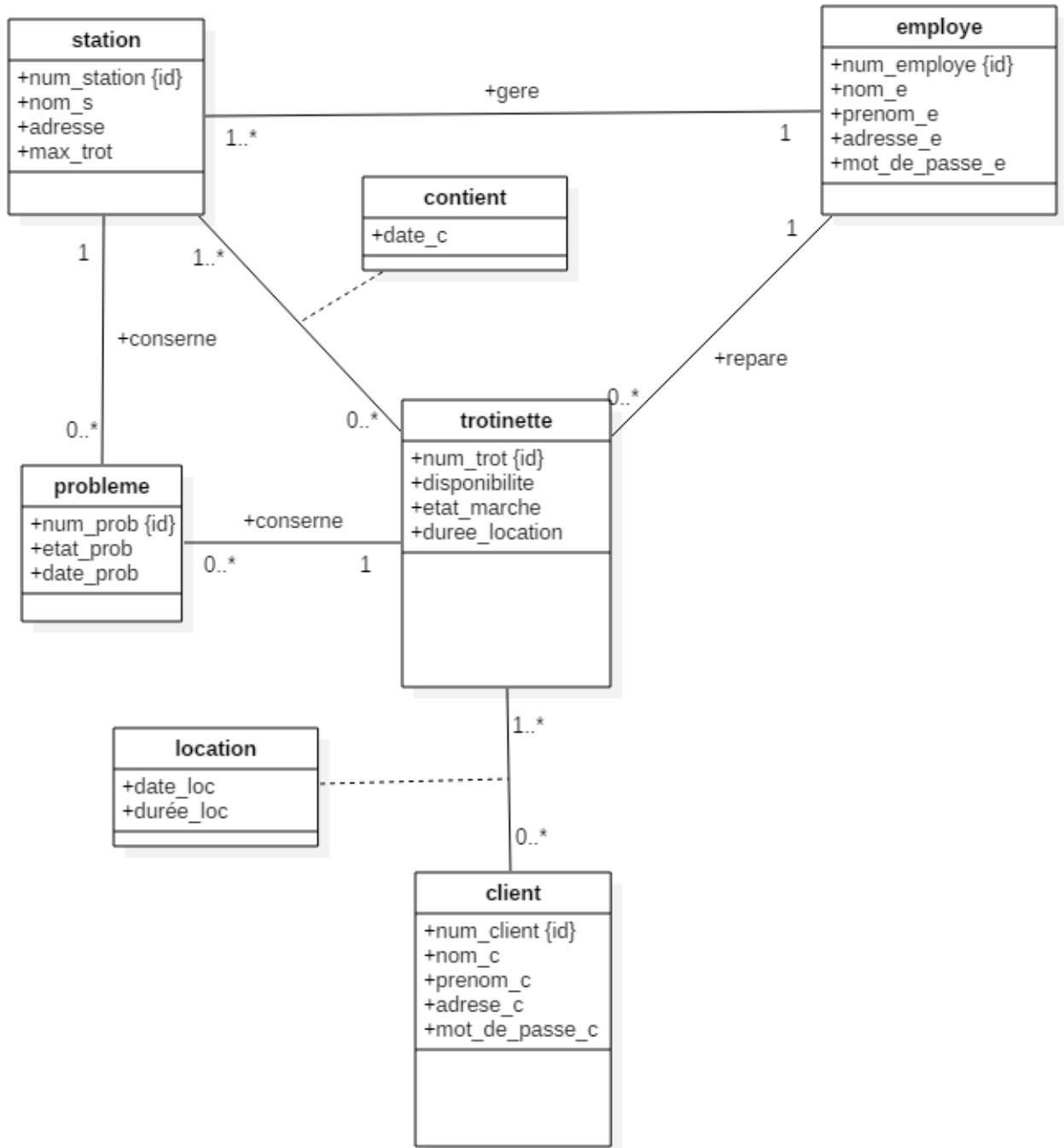
Membres du groupe :

-Lougani Faouzi

-Tafat Aghiles

1-LE MODELE ENTITE /ASSOCIATION

Grace a l'outil de modélisation StarUML on a pu réalisé notre modèle ci-joint



Etapes de réalisation du modèle Entité/Association :

Pour la conception de notre modèle on a commencé a créer les entités les plus claires comme :client ,trotinette ,station et employe .

-D'après l'énoncé le client peut louer et rendre plusieurs trottinettes dans une journée donnée pour cela on a pu ajouter une entité association appelée location qui nous permettra de gérer ce cas.

-Comme la trottinette occupe plusieurs stations dans une journée(un client peu la louer dans une station et la rendre dans une autre) on a pu ajouter une table Contient pour savoir les stations pleines et gérer le cas que la même trottinette peut apparaitre dans plusieurs stations.

-L'entité problème nous permettra de gérer dans une journée donnée les problèmes de notre station (station pleine) et aussi les trottinettes en pannes.

Les contraintes sur les tables :

Puisque les trottinettes Lorsqu'elles tombent en panne elles informent elles mêmes le serveur qu'elles ne sont plus fonctionnelles or ca peut qu'il soit une trottinette en panne et en mêmes temps louée pour cela on a :

Disponibilite :prend soit « quai » si la trottinette est disponible ou «loue » sinon .

- **etat_marche** :prend soit « fonction » si la trottinette est en bonne etat de fonction ou « enpanne »

Pour problème elle possède une variable :

-**etat_prob** : prend soit « regle » ou « nonregle »

Puisque on a utilisé *PhpMyadmin* la première difficulté est l'implémentation de ses contraintes avec check constraint que ce dernier n'accepte pas pour cela on a utilisé des **triggers**

Pour **Disponibilite**:

```
1 CREATE TRIGGER `tr1` AFTER INSERT ON `trotinette`  
2 FOR EACH ROW  
3 BEGIN  
4 if NOT(new.disponibilite="quai" or new.disponibilite="loue" ) then  
5 INSERT into trotinette(num_trot,disponibilite,etat_marche,num_employe)  
6 values (new.num_trot,new.disponibilite,new.etat_marche,new.num_employe);  
7 end IF; END
```

Pour **etat_marche** :

```
1 CREATE TRIGGER `tr2` BEFORE INSERT ON `trotinette`  
2 FOR EACH ROW  
3 BEGIN  
4 if NOT(new.etat_marche="fonction" or new.etat_marche="enpanne" ) then  
5 INSERT into trotinette(num_trot,disponibilite,etat_marche,num_employe)  
6 values (new.num_trot,new.disponibilite,new.etat_marche,new.num_employe);  
7 end IF; END
```

Pour **etat_prob** :

```
1 CREATE TRIGGER `tr3`  
2 AFTER INSERT ON `probleme`  
3 FOR EACH ROW  
4 BEGIN if NOT(new.etat_prob="regle" or new.etat_prob="nonregle" )  
5 then  
6 INSERT INTO probleme(num_prob,etat_prob,date_prob,num_station,num_trot)  
7 VALUES (new.num_prob,new.etat_prob,new.date_prob,new.num_station,new.num_trot);  
8 end IF;  
9 END
```

- les variables num_client , num_prob , num_trot, num_employe doivent être incrémenté automatiquement a chaque ajout .

Modèle relationnel :

station(num_station, nom_s, adresse_s, max_trot, num_employé)

client(num_client, nom_c, prenom_c, adresse_c, mot_de_passe_c)

employe(num_employe, nom_e, prenom_e, adresse_e, mot_de_passe_e)

trotinette(num_trot, disponibilibilite, etat_marche, num_employe)

contient(num_station, num_trot, date_c)

location(num_client, num_trot, date_loc, durée_location)

probleme(num_prob, etat_prob, date_prob, num_station, num_trot)

les requêtes requises dans l'algèbre relationnelle.

1. Calculer le montant de la location d'une trotinette pour un client (en euro):

R1=JOIN (client ,location).

R2=RESTRICT (R1, num_client='num notre client' et date_loc='aujourd'hui')

R3= PROJECT(R2, duree_location*0.1) .

$\pi_{\text{durée_location} \times 0.1} (\sigma_{\text{num_client} = \text{'num notre client'} \wedge \text{date} = \text{'aujourd'hui'}} (\text{Client} \bowtie \text{location}))$

Remarque : en SQL on va récupérer la date grâce à DATE (NOW()).

2. Lister les stations qui ont des trotinettes a louer :

R1=RESTRICT(trotinette, disponibilibilite='quai' and etat_marche='fonction').

R2=JOIN(R1,contient).

R3=JOIN(R2,station).

R4=Project(R3, num_s, adresse_s).

$\pi_{\text{num_s}, \text{adresse_s}} (\text{station} \bowtie (\text{contient} \bowtie (\sigma_{\text{disponibilibilite} = \text{'quai'} \wedge \text{etat_marche} = \text{'fonction'}} (\text{trotinette}))))$

3. Louer une trotinette dans une station donnée (choisissons comme exemple la station numéro 4 :

R1=RESTRICT(station,num_s='4').

R2=JOIN(R1,contient).

R3=JOIN(R2,trotinette).

R4=RESTRICT(R3, disponibilibilite='quai' and etat_marche='fonction') .

R5=PROJECT(R4,num_trot)

$\pi_{\text{num_trot}} (\sigma_{\text{disponibilibilite} = \text{'quai'} \wedge \text{etat_marche} = \text{'fonction'}} (\text{trotinette} \bowtie (\text{contient} \bowtie (\sigma_{\text{num_s} = \text{'4'}} (\text{station}))))))$

4. Trottinette en panne (puis on affiche la station ou elle se trouve car pour un employé le but n'est pas juste de savoir la trottinette en panne mais le lieu ou elle se trouve aussi afin de la réparer)

$R1 = \text{RETRICT}(\text{trottinette}, \text{etat_marche} = \text{'en panne'})$.

$R2 = \text{JOIN}(R1, \text{contient})$

$R3 = \text{JOIN}(R2, \text{station})$

$R4 = \text{Project}(R3, \text{Num_trot}, \text{nom_s}, \text{adresse_s})$

$\pi_{\text{num_trot}, \text{nom_s}, \text{adresse_s}} (\text{station} \bowtie (\text{contient} \bowtie (\sigma_{\text{etat_marche} = \text{'en panne'}}(\text{trottinette}))))$

5. Nombre de trottinettes louées dans une journée donnée :

$R1 = \text{RESTRIC}(\text{location}, \text{date_loc} = \text{'notre date'})$.

$R2 = \text{COUNT}(R1)_{\text{DISTINCT}(\text{num_trot})}$

$\text{num_trot } \gamma \text{count}(\sigma_{\text{date} = \text{'notre date'}}(\text{location}))$

6. Station pleine

$\pi_{\text{nom_s}, \text{adresse_s}} (\sigma_{\text{max_trot} = ((\text{station}) \bowtie (\text{num_station} \bowtie \text{count}(\text{num_trot})(\text{contient}))))$

la relation group by sera noté comme suite : <attributs de regroupement> \bowtie <Liste de fonctions> (**RELATION**)

7. Nombre de trottinettes en panne pour une journée donnée:

$R1 = \text{RESTRIC}(\text{contient}, \text{'notre date'})$.

$R2 = \text{JOIN}(R1, \text{trottinette})$.

$R3 = \text{RESTRIC}(R2, \text{etat_marche} = \text{'en panne'})$.

$R4 = \text{COUNT}(R3)$

$\text{num_trot } \text{count} (\sigma_{\text{etat_marche} = \text{'en panne'}} (\text{trottinette} \bowtie (\sigma_{\text{date_c} = \text{'notre date'}} (\text{contient}))))$

8. Nombre de problèmes non réglés dans une journée donnée :

$R1 = \text{RESTRIC}(\text{probleme}, \text{etat_prob} = \text{'nonregle'} \text{ and } \text{date_prob} = \text{'notre date'})$.

$R2 = \text{COUNT}(R1)$.

$\text{Num_prob } \text{count} (\sigma_{\text{etat_prob} = \text{'nonregle'}} \wedge \text{date_prob} = \text{'notre date'}} (\text{probleme}))$

9. Nombre de problèmes réglés dans une journée donnée :

```
R1=RESTRICT(probleme,etat_prob='regle' and date_prob='notre date').
```

```
R2=COUNT(R1)
```

```
Num_prob count ( $\sigma_{\text{etat\_prob}='regle' \wedge \text{date\_prob}='notre date'}$  (probleme))
```

Requêtes pour la création des tables :

1/table client :

use trotinette;

```
CREATE TABLE IF NOT EXISTS client (
```

```
    num_client int not null AUTO_INCREMENT,
```

```
    nom_c varchar(20) ,
```

```
    prenom_c varchar (20) ,
```

```
    adresse_c varchar(50),
```

```
    mot_de_passe_c varchar(20)not null,
```

```
    montant int not null,
```

```
    primary key(num_client));
```

2/table employe:

use trotinette;

```
CREATE TABLE IF NOT EXISTS employe(
```

```
    num_employe int not null AUTO_INCREMENT,
```

```
    nom_e varchar(20),
```

```
    prenom_e varchar(20),
```

```
    adresse_e varchar(50),
```

```
    mot_de_passe_e varchar(20)not null,
```

```
    primary key (num_employe));
```

3/table station :

```
create table IF NOT EXISTS station (  
    num_station int not null AUTO_INCREMENT,  
    nom_s var char(20),  
    adresse varchar(50),  
    max_trot int not null ,  
    num_employe int not null,  
    PRIMARY key(num_station),  
    FOREIGN key (num_employe)REFERENCES employe(num_employe));
```

4/table trotinette :

```
use trotinette;  
  
create table IF NOT EXISTS trotinette(  
    Num_trot int not null AUTO_INCREMENT,  
    Disponiblite char(15) not null,  
    etat_marche char(15) not null,  
    duree_location int,  
    num_employe int not null,  
    num_client int not null,  
    primary key (num_trot)  
    FOREIGN key (num_employe)REFERENCES employe(num_employe),  
    FOREIGN key (num_client) REFERENCES client(num_client),  
    );
```


5/table contient

use trotinette;

```
create table IF NOT EXISTS contient(  
    num_station int not null,  
    num_trot int not null,  
    date_c date not null,  
    primary key(num_station,num_trot),  
    FOREIGN key (num_station) REFERENCES client (num_station),  
    FOREIGN key (num_trot) REFERENCES trotinette(num_trot));
```

6/location

use trotinette;

```
create table IF NOT EXISTS location(  
    num_station int not null,  
    num_trot int not null,  
    primary key(num_station,num_trot),  
    FOREIGN key (num_station) REFERENCES station(num_station),  
    FOREIGN key (num_trot) REFERENCES trotinette(num_trot));
```

7/probleme

use trotinette;

```
create table IF NOT EXISTS probleme(  
    num_prob int not null AUTO_INCREMENT,  
    etat_prob varchar(15) not null,  
    date_prob date ,  
    num_station int not null,
```

num_trot int not null,
primary key(num_prob),
FOREIGN key (num_station) REFERENCES station(num_station),
FOREIGN key (num_trot) REFERENCES trotinette(num_trot));

Les requêtes SQL nécessaires aux formulaires PHP:

1. Calculer le montant de la location d'une trotinette pour un client exemple num 2 :

```
1 select duree_loc*0.1 from client,location
2 where
3 client.num_client=location.num_client
4 and client.num_client=2
5 and location.date_loc="2019-05-01"
```

2. Lister les stations qui ont des trotinettes a louer :

```
1 select nom_s,adresse_s from trotinette,contient,station
2 where
3 trotinette.disponibilite='quai'
4 and trotinette.etat_marche='fonction'
5 and trotinette.num_trot=contient.num_trot
6 and contient.num_station=station.num_station
```

3. Louer une trotinette dans une station donnée (choisissons comme exemple la station numéro 1 :

```
1 select trotinette.num_trot from station ,contient ,trotinette
2 where station.num_station=1
3 and station.num_station=contient.num_station
4 and contient.num_trot=trotinette.num_trot
5 and trotinette.disponibilite='quai'and
6 trotinette.etat_marche='fonction'
```

4. Trotinette en panne :

```

1 select trotinette.num_trot,station.nom_s,station.adresse_s
2 from trotinette,contient,station
3 where trotinette.etat_marche='enpanne'
4 and trotinette.num_trot=contient.num_trot
5 and contient.num_station=station.num_station
6

```

5. Nombre de trottinettes louées dans une journée donnée :

```

1 select count(distinct(num_trot)) from location
2 where location.date_loc="2019-05-01"

```

6. Station pleine :

```

1 SELECT * from station
2 where station.max_trot IN (
3     SELECT count(contient.num_trot)from contient,station
4     where date_c=date(now())
5     and contient.num_station=station.num_station
6     group by contient.num_station
7 )
8

```

7. Nombre de trottinettes en panne pour une journée donnée:

```

1 select count(trotinette.num_trot)from contient,trotinette
2 where
3 contient.date_c="2019-05-01"
4 and contient.num_trot=trotinette.num_trot
5 and trotinette.etat_marche='enpanne'

```

8. Nombre de problèmes non réglés dans une journée donnée :

```

1 SELECT COUNT(*) FROM probleme
2 WHERE Etat_prob='nonregle'
3 and date_prob="2019-05-01"

```

9. Nombre de problèmes réglés dans une journée donnée :

```

1 SELECT COUNT(*) FROM probleme
2 WHERE Etat_prob='regle'
3 and date_prob="2019-05-01"
4

```

Remarque : toutes ses requêtes sont incluses en format compatible (afin qu'elle puissent être exécutées en PHP)

-la requête sql nécessaire pour l'ajout d'un employé

```
//ajouter un employe
$stafat = mysqli_query($connect, "INSERT INTO employe(nom_e, prenom_e,adresse_e, mot_de_passe_e) VALUES( '$nom_e', '$prenom_e', '$adresse_e', '$mot_de_passe_e')");//ajouter

/*****/
}
```

-Lorsque la trottinette est réparée on met à jour la variable **etat_marche** en « fonction »

```
/*****/
//mise à jour la table trottinette(modifier l'état de marche de la trottinette)

$test = $db->query('update trottinette set trottinette.etat_marche = "fonction" where trottinette.etat_marche="en panne" and trottinette.num_trot='.$numT_rep ');

/*****/
```

-Si un client loue une trottinette on doit mettre à jour la variable disponibilité

```
/*****/
//mise à jour la table trottinette(disponibilite)

$db->query('update trottinette set trottinette.disponibilite = "loue" where trottinette.disponibilite="quai" and trottinette.num_trot='.$loct);

/*****/
//mise à jour du durée location
/*****/

$inst=$db->query('update location set location.duree_loc='.$temp_loc.' where location.num_trot='.$loct);
```

-Mise à jour de la disponibilité de la table trottinette .

-Mise à jour la de table contient (la requête va mettre à jour la station et la date de la trottinette rendu).

```
/*****/
//Modifier la disponibilite de trottinette
$req= $db->query('update trottinette set trottinette.disponibilite = "quai" where trottinette.disponibilite="loue" and trottinette.num_trot='.$r_troti);

/*****/


/*****/
//mise à jour de la table contient
$req=$db->query('UPDATE contient SET contient.num_station="'.$r_statn."', contient.date_c='".$temps_j.'" where contient.num_trot='.$r_troti);
```

Explication de l'étape formulaires en PHP et HTML :

Bienvenue !

Vous etes employe

Vous etes client



La page d'accueil permet au client et a l'employé de se connecter ou s'enregistrer

Bienvenue Dans L'Espace Inscription Client

NOM

Prenom

Adresse

Mot de passe



Une fois le client a choisit l'enregistrement il se trouve dans l'espace d'inscription pour s'enregistrer s'il n'est pas déjà client

Bienvenue Dans L'Espace Inscription Employe

NOM
Votre nom

Prenom
Votre Prenom

Adresse
Votre Adresse

Mot de passe
Votre Mot de passe

S'inscrire

Une fois l'employé a choisit l'enregistrement il se trouve dans l'espace d'inscription pour s'enregistrer s'il n'est pas déjà employé

Mon Espace Client

Bienvenue dans votre Espace Client

Numero client
Votre numero

Mot de passe
Votre mot de passe

Connexion



Le client doit saisir son numéro unique et son mot de passe pour se connecter

Bienvenue dans votre espace location

Les trotinettes disponibles sont :

Trottinette numero:1

Trottinette numero:5

choisez le numero de trotinette à louer

choisez le duree en minutes

louer

[clique ici](#) pour revenir à la page principale.php.

Merci de rendre la trottinette que vous avez loué

Prix de location: 4.5 EURO

Vous avez loué la trottinette numero: 1 avec succès

Rendre la trottinette que vous avez loué

Saisissez numero de station

Número station

Saisissez numero de trottinette

Número trottinette

[clique ici](#) pour revenir à la page d'accueil

Dans cette espace le client trouve le prix de location et la rubrique pour rendre la trottinette qu'il a loué .

Merci de rendre la trottinette que vous avez loué !!

Merci d'avoir rendu la trottinette numero 12

[clique ici](#) Pour revenir à la page d'accueil.php.

ce message s'affichera après la remise de la trottinette

Pour l'employé :

Mon Espace Employe

Bienvenue dans votre Espace Employe

Numero employe
22
Mot de passe

Connexion



L'employé doit saisir son numéro unique et son mot de passe pour se connecter

Mon Espace Du Travail

Pour lister les informations suivantes:

- *le nombre de trutinette loues dans une journée
- *les numeros des trutinette en pannes dans une journée
- *le nombre des problemes regles dans une journée
- *le nombre des problemes non regles dans une journée

Entrez une date :

Si tu veux changer de prénom, [clique ici](#) pour revenir à la page formulaire.php.



L'employé doit entrer une date pour lister tout les informations sur le réseau

Mon Espace Du Travail

Pour la date :2019-05-01

le nombre de trutinette loue est :4
le nombre de trutinette en pannes est:2
le nombre des problemes regles est: 1
le nombre des problemes non regles est :1
les numeros des trutinette en pannes:
numero :4
nom station :c
adresse station :canet

numero :6
nom station :d
adresse station :marseille

Entrez le numero de trutinette a reparer :

Pour lister les stations pleins [clique ici](#)

[clique ici](#) pour revenir à la page d'accueil



l'employé a toutes les informations sur le réseau comme il peut choisir un problème a régler

la trottinette numero 6 est repare avec succes !

Merci!

[clique ici](#) pour revenir à la page d'accueil