# Projet bases données

Ibrahima diao

# 1 Introduction

L'objectif de ce projet est de créer un base de données sous sql . Le sujet concerne la gestion du village gaulois(habitant, combats, casques...).
La base de données à créer doit permettre de recenser tous les habitants du village , leurs adresses, les combats et les casques prises durant les combats.

Dans le cahier de charge( discussion entre informaticien et le chef), on nous demande de gérer :

- Recenser les habitants d'un quartier, en effet il arrive que le chef convoque les habitants du même quartier..
- les participants dans un combats.
- les habitants qui ont pris des casques sur les combats.

# 2 Modèle conceptuel de données

La réalisation du modèle conceptuel de données est la première étape pour réaliser les tables de notre base de données. Cette étape est une sorte de traduction de l'énonce qui permet de référencer tous les données que l'on souhaite stocker dans la base .(voir figure)

# 3 le dictionnaire de données

| Relation | Attribut 2 | clé |
|---|---|---|
| quartier | (nomQ, distanceC,couleur1,couleur2,couleur3) | nomQ |
| habitant | (identite, nom, prenom, specialite, voie, numerovoie, statut) | identite |
| combat | (nomC, date, lieu) | nomC |
| casque | (numero, categories, disponibilte, etat, forme,grade ) | numero |
| prise | (habitant ,combat, casque, nbrepris) | (habitant ,combat, casque) |
| participe | (habitant, combat) | (habitant, combat) |
| habite | (habitant,quartier) | (habitant,quartier) |

## 3.1 Optimisation

Cela consiste à déplacer Toutes les classes d'associations reliées à une classe d 'entités avec une cardinalité de type $0,1$ ou $1,1$ peuvent être fusionnée avec la classe d 'entités. Dans ce cas on déplace les attributs de la classe d 'associations vers ceux de la relation traduisant la classe d 'entités.

- Notre schéma peut être optimiser car il contient une association de type $0,1$ et $1,1$.
- La table qui traduit l'association habite n'a pas lieu d'être car elle est de type $1,1$ car un habitant peut avoir un maximum une adresse.

- C'est pareil pour l'association prise car une casque peut être pris au plus par un habitant .

Cependant notre dictionnaire devient :

| Relation | Attribut 2 | clé |
|---|---|---|
| quartier | (nomQ, distanceC, couleur1, couleur2, couleur3) | nomQ |
| habitant | (identite, nom, prenom, specialite, voie, numerovoie, statut, quartier) | identite |
| combat | (nomC, date, lieu) | nomC |
| casque | (numero, categories, disponibilte, etat, forme, grade, habitant , combat, nbrpris) | numero |
| participe | (habitant, combat) | (habitant, combat) |

# 4 Modèle logique de données et 3 forme normale

## 4.1 Modèle logique des données

Le modèle logique est déduit du modèle conceptuel de données ,on obtient les tables suivantes .
Les clés primaires sont celle qui sont en gras et soulignées les clés étrangères en gras.
quartier(**nomQ**, distanceC,couleur1, couleur2, couleur3)

habitant (**identite**, nom, prenom, specialite, voie, numerovoie, statut, **quartier**)
combat (**nomC**, date, lieu)
casque (**numero**, categories, disponibilte, etat, forme, grade, **habitant** , **combat**, nbrpris)
participe **( habitant, combat)**

## 4.2 3 forme normale

On va vérifier pour toutes les tables que tous les attributs dépendent directement de la clé et aucun attribut ne doit dépendre de la clé par transitivité c'est à dire aucun attribut ne doit dépendre d'un autre attribut non clé.
quartier(**nomQ**, distanceC,couleur1, couleur2, couleur3)
habitant (**identite**, nom, prenom, specialite, voie, numerovoie, statut, **quartier**)
combat (**nomC**, date, lieu)
participe **( habitant, combat)**
Ces relations sont en 3éme forme normale car tous les attributs dépendent pleinement des clés pour chaque relation. Par contre pour la relation casque on a :
$forme \rightarrow grade$
Pour se ramener en 3éme forme normale la relation casque doit être décomposer , on obtient :
casque (**numero**, categories, disponibilte, etat, forme, **habitant** , **combat**, nbrpris)

equival(**forme**,grade)
Finalement on obtient les relations suivantes qui sont en 3 formes normale.
quartier(**nomQ**, distanceC,couleur1, couleur2, couleur3)
habitant (**identite**, nom, prenom, specialite, voie, numerovoie, statut, **quartier**)
combat (**nomC**, date, lieu)
participe **( habitant, combat)**
casque (**numero**, categories, disponibilte, etat, **forme**, **habitant** , **combat**, nbrpris)
equival(**forme**, grade)

### 4.2.1 Les contraintes d'intégrité référentielle

$proj_{habitant}(participe) \subseteq proj_{identite}(habitant)$
$proj_{combat}(participe) \subseteq proj_{nomC}(combat)$
$proj_{habitant}(casque) \subseteq proj_{identite}(habitant)$
$proj_{combat}(casque) \subseteq proj_{nomC}(combat)$
$proj_{forme}(casque) \subseteq proj_{forme}(equival)$
$proj_{quatier}(habitant) \subseteq proj_{nomQ}(quartier)$

### 4.2.2 les contraintes de domaine

Nous avons imposé quelques conditions aux attributs pour qu'ils répondent à la cohérence de la base .Cependant ces attributs pourront contenir que ces valeurs sinon ils sont vide si ils sont non nuls.
etat (Casque) = (bon ,mauvais,moyen)
statut(habitant) = (combattant ,no combattant ,ancien combattant)
disponibilite(casque) = (detenu, en service)
couleur=les couleurs standards