

# Projet Informatique Appliqué

Carmania :  
achat/location de véhicules

Groupe 3  
BESSAI Sofiane  
BIEMAR Alexandre  
FAUCONNIER Axel

Université d'Aix-Marseille

# Plan de la présentation

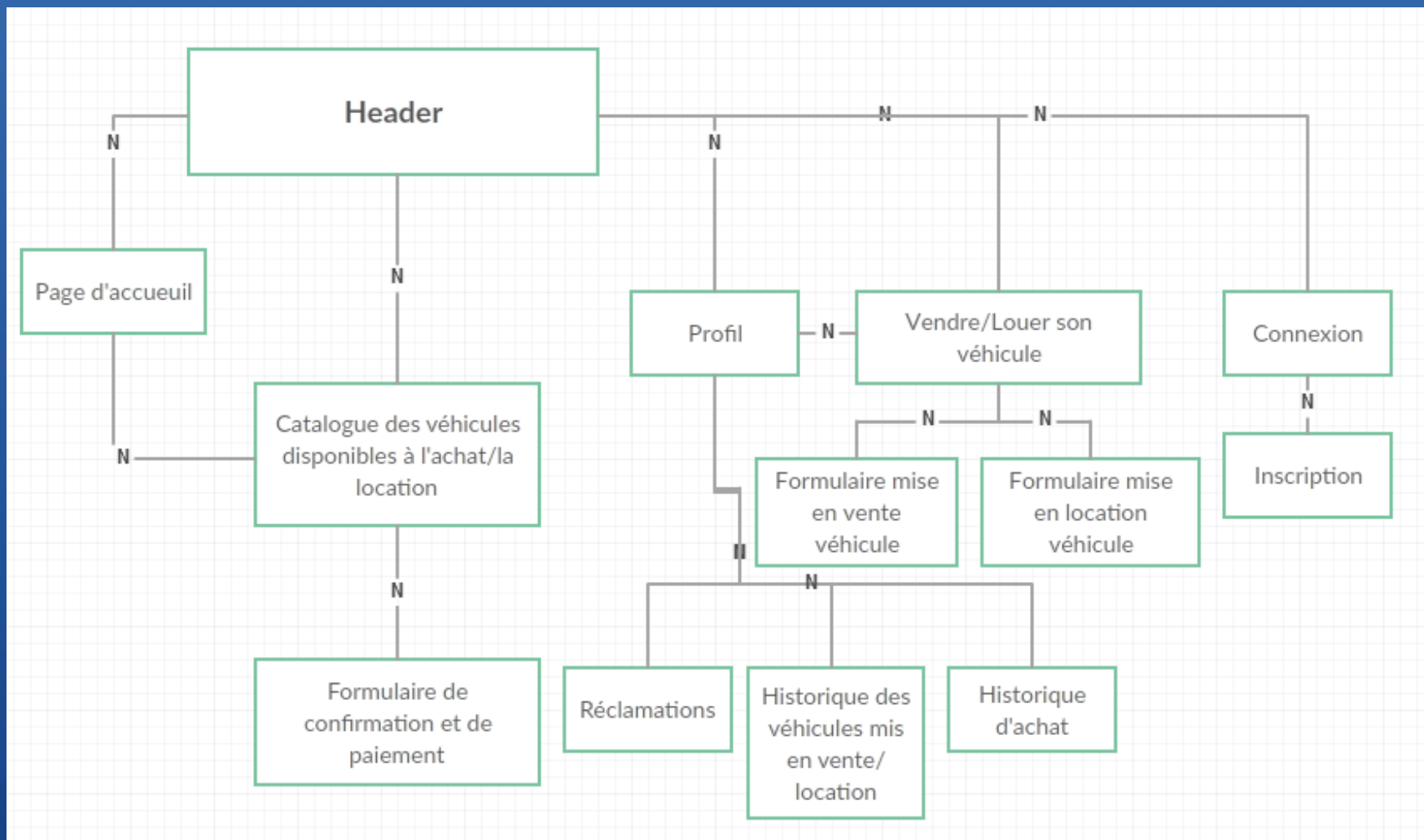
- **I. Présentation du projet**
  - Objectifs
  - Workflow
  - Base de données
  - Cahier des charges
  - Remplissage du cahier des charges
- **II. Démonstration du site**
  - Structure du site
  - Comptes
  - Catalogue
  - CRUD
- **III. Réalisation du projet**
  - Outils utilisés
  - Répartition du travail
  - Conclusion et perspectives d'évolution
- **IV. Références**

## Objectifs

- Nous voulions concevoir un site de vente simple mais fonctionnel, nous avons choisi un site de vente et de location de véhicules
- Un utilisateur peut acheter ou louer un véhicule parmi un catalogue de véhicules disponibles et a également la possibilité de mettre en vente ou de louer son propre véhicule via le site
- Le site est conçu en HTML/CSS et PHP pour communiquer avec sa base de données

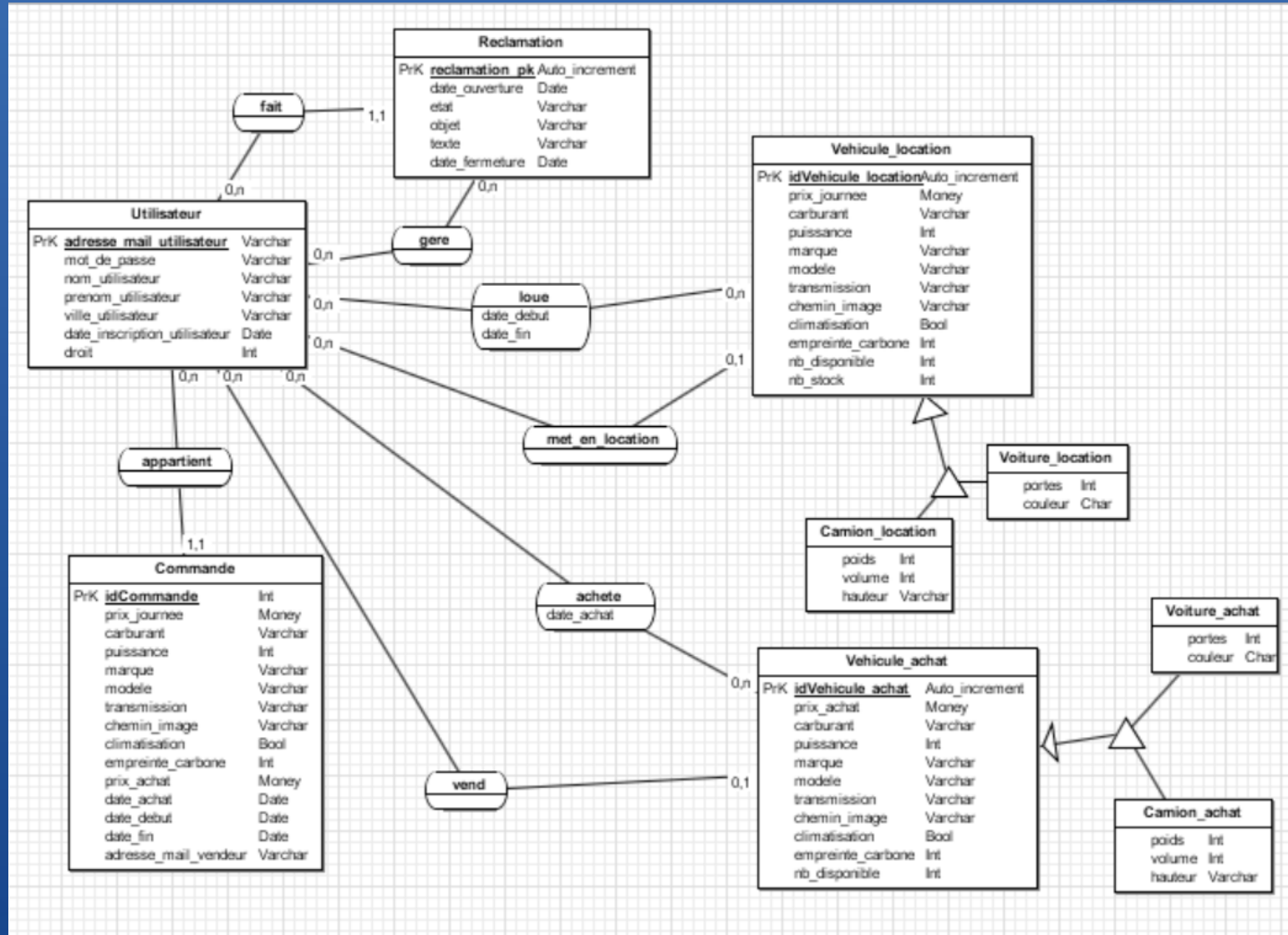
# Workflow

Figure 1 :



# Base de données

Figure 2 :



# Cahier des charges (1)

Nom	Importance	Avancement
MCD BDD véhicules	120	Terminé
MCD BDD comptes	120	Terminé
MCD BDD locations/ventes	120	Terminé
MCD final	120	Terminé
Conception BDD	120	Terminé
Page principale site	120	Terminé
Page de recherche de véhicule (achat)	120	Terminé
Page de recherche de véhicule (location)	120	Terminé
Formulaire création de compte utilisateur	120	Terminé
Lien page de recherche → BDD véhicule	120	Terminé
Page de connexion au compte utilisateur	120	Terminé
Lien création/connexion → BDD utilisateur	120	Terminé
Fiche information du véhicule	110	Terminé
Connexion compte administrateur	110	Terminé
Interface achat véhicule	100	Terminé
Interface location véhicule	100	Terminé

# Cahier des charges (2)

Nom	Importance	Avancement
Page informations de paiement	100	Terminé
Profil utilisateur	95	Terminé
Historique activité utilisateur	90	Terminé
Ajout critères de recherche supplémentaires	70	Terminé
CSS site	70	Terminé
Hachage mdp	50	Terminé
Page SAV	45	Terminé
Option de mise en vente/location pour un particulier	40	Terminé
Favoris utilisateur	25	Non terminé
Choix d'assurances	25	Non terminé
Système de notation et commentaire	25	Non terminé
Offres promotionnelles	25	Non terminé
Système de fidélité	25	Non terminé
Système de forfaits de location	25	Non terminé
Pagination	25	Terminé
Suppression de véhicules	25	Terminé

## Remplissage du cahier des charges

- Toute les fonctionnalités d'importance très hautes (en rouge) et hautes (en jaune) sont opérationnelles
- Pour ce qui concerne les fonctionnalités d'importance moyenne et basse (en vert), nous n'avons réalisé que la mise en vente/location d'un véhicule de particulier, l'envoi de réclamations au SAV et la pagination des pages de catalogue ou d'historique



## Structure du site

- Le site a une structure assez simple, pouvant être simplifiée en deux blocs organisés autour de la page d'accueil :
  - d'une part le catalogue (liste des véhicules et permettant d'acheter/louer)
  - d'autre part l'interface de gestion de compte (connexion, inscription, profil, mise en vente/location de véhicule personnel, réclamations)
- Un header permet l'accès rapide aux principales pages depuis n'importe quelle page du site

## Compte (1)

- Inscription: On vérifie si toute les informations sont valides et on inscrit le tout dans la table utilisateur, en hachant le mot de passe avec password\_hash
- Connexion: On vérifie si l'adresse mail correspond au mot de passe, en utilisant password\_verify pour lire le mot de passe dans la base de données
- Un utilisateur peut modifier son mot de passe et sa ville via la page profil, ainsi que créer une réclamation, consulter la liste des annonces qu'il a créé ou l'historique de ses achats/locations

## Compte (2)

- Si le compte est flag en tant que compte administrateur, son profil lui donnera la possibilité de gérer les réclamations et il pourra supprimer des véhicules du catalogue
- L'administrateur peut aussi voir l'historique de l'ensemble des véhicules vendus ou loués sur le site

# Catalogue

- Les catalogues d'achat et de location diffèrent peu l'un de l'autre visuellement, affichant juste quelques informations différemment. Les véhicules sont cependant stockés dans deux tables différentes. La liste des véhicules est créée par des requêtes SQL à la base de données
- On peut filtrer les véhicules affichés dans le catalogue par marque, type de transmission et prix. Le tout affiché dans un menu créé grâce au framework de w3.css
- Pour chaque véhicule est affiché ses caractéristiques et un bouton acheter/louer, qui redirige vers la page de paiement.

## CRUD (1)

Create :

- Création de compte
- Création d'annonce

Read :

- Affichage d'une annonce
- Affichage d'informations liées au compte
- Affichage d'historique

## CRUD (2)

Update:

- Mise à jour profil
- Mise à jour d'une réclamation (état)

Delete :

- Suppression d'une annonce

## Outils utilisés

- Pour la gestion du projet nous avons surtout utilisé Github
- Le MCD de la base de données et son code SQL fut créé à l'aide de Jmerise
- Nous avons utilisé MySQL et PHPMyAdmin via uwAmp pour le coté serveur
- Le reste fut fait par simple éditeur de texte, principalement Notepad++ et Geany

## Répartition du travail

- Nous avons essayé de nous repartir de façon égale, en matière de charge de travail, les différentes tâches, en fonction des points forts de chacun
- Bien que ayant chacun sa tâche, nous communiquions beaucoup et nous nous aidions l'un l'autre sur nos tâches respectives, par exemple pour ce qui est de la base de données, qui fut plus un travail commun et fut par la suite ajustée plusieurs fois au besoin de chacun d'entre nous
- Le travail fut surtout réalisé le vendredi en séance de TP et le soir, mais fut accéléré en fin d'UE pour rattraper notre retard sur le cahier des charges



## Conclusions et perspectives d'évolution

### Conclusion :

- Apprentissage du php
- Apprentissage de nouvelles méthodes de travail
- Organisation difficile

### Améliorations pour le futur :

- Programmation orientée objet
- Sécurité du site ( injection sql, contrainte de mot de passe..)
- Utilisation de framework

# Références

- Github du projet: <https://github.com/Ganti7/Carmania>
- Référence HTML/PHP/CSS: <https://www.w3schools.com/> - <https://openclassrooms.com>
- Framework: <https://www.w3schools.com/w3css/>
- Bibliographie:
  - Scrum and XP depuis les tranchées, par Henrik Kniberg
  - Progit, Second Edition, de Scott Chacon & Ben Straub