

PROJET



SOMMAIRE

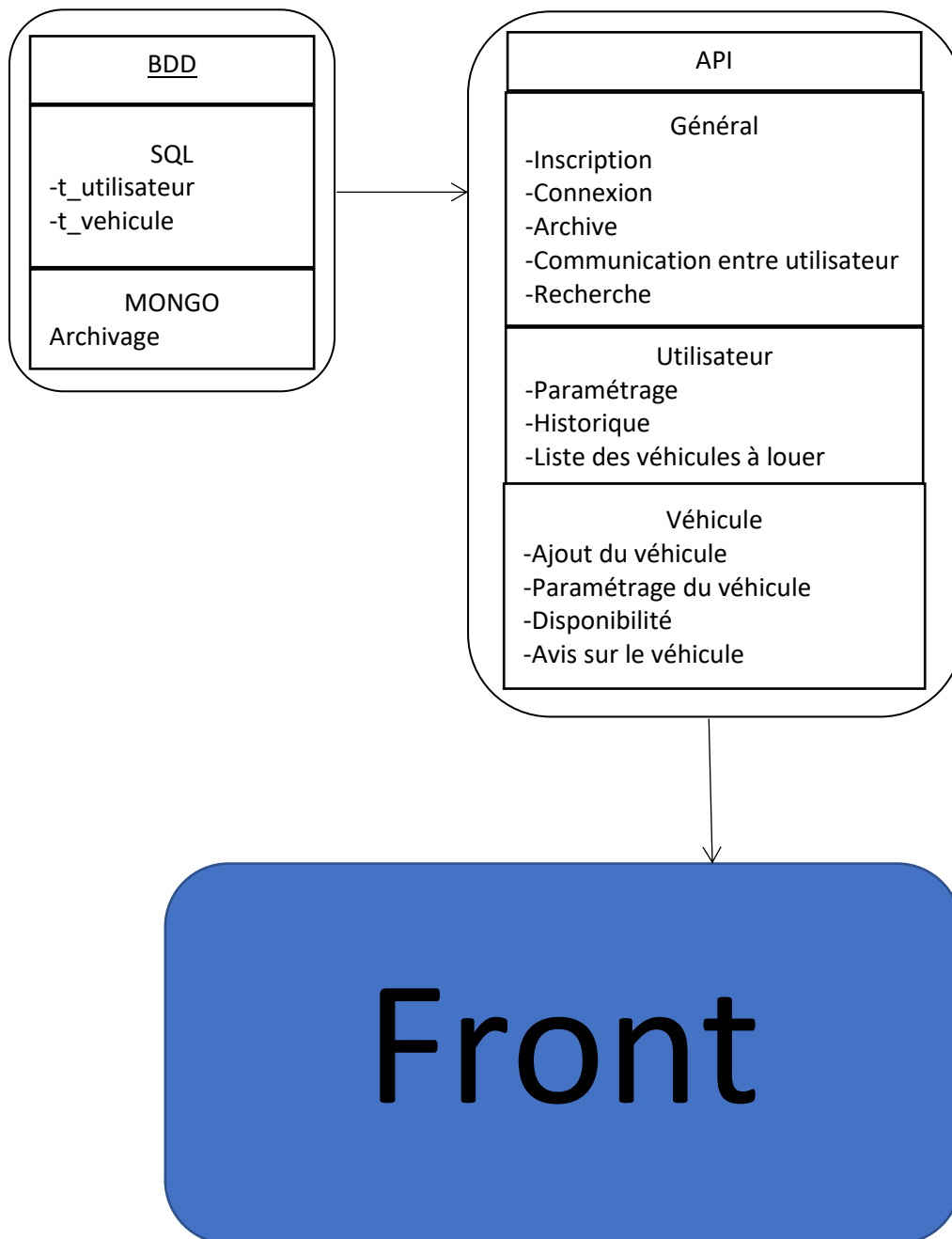
1 – Description.....	2
2 – Le diagramme des composants.....	3
3 – Les fonctionnalités.....	4
Compte utilisateur	4
Véhicule	4
Pour louer son véhicule :	4
Pour le choix du véhicule	5
Règlement Bancaire.....	5
Règlement direct.....	5
Règlement en ligne	5
Durée de location	5
Trajet.....	5
4 – La structure de la base de données.....	6
SQL	6
MongoDB	6
5 – Les routes	7

1 - Description

Ce projet a pour but de créer une web app qui permettra la location de véhicule en France. Pour y parvenir, nous nous inspirerons du site internet de [drivy](#).

Il sera divisé en deux parties. Une partie concernant le back-end essentiellement en NodeJS pour le serveur, SQL et MongoDB pour la/les base(s) de données. Pour le front-end, il sera élaboré avec du ReactJS.

2 - Le diagramme des composants



3 - Les fonctionnalités

Compte utilisateur

Les données demandées à la création du compte :

- Nom Prénom
- Adresse email
- Mot de Passe

Les données obligatoires pour proposer ou faire une location :

- Adresse
- Permis de conduire
- Date d'obtention
- Pays d'obtention

Les données optionnelles :

- Téléphone
- Date de Naissance
- Lieu de Naissance
- Présentation
- Réseaux sociaux
- Photo

Les fonctionnalités :

- Paramétrage : servira à configurer son compte et à rajouter ou mettre ses données.
- Historique : permettra de consulter ses anciennes locations ou les locations de son véhicule.

Véhicule

Pour louer son véhicule :

Les données obligatoires :

- Marque
- Modèle
- Année d'immatriculation
- Kilométrage
- Prix
- Lieux
- Temps minimum
- Temps maximum

Les données optionnelles :

- Photo
- Carburant
- Caractéristique technique
- Option
- Etat du véhicule
- Condition souhaité

Pour le choix du véhicule

- Lieux
- Type de véhicule
- Prix
- Nombre de place
- Age du véhicule
- Boite
- Moteur
- Siège bébé
- GPS
- Climatisation
- Porte-vélo
- Coffre toit
- Régulateur de vitesse
- Limiteur de vitesse
- Pneu neige
- Chaîne

Règlement Bancaire

Règlement direct

- Espèce
- Chèque

Règlement en ligne

- Paypal
- Par carte

Durée de location

- Début et Fin de la location

Trajet

- Secteur

4 - La structure de la base de données

SQL

```
t_utilisateur (  
id INT AUTO_INCREMENT NOT NULL,  
nom VARCHAR(50) NOT NULL,  
prenom VARCHAR(20) NOT NULL,  
email VARCHAR(60) NOT NULL,  
password VARCHAR(16) NOT NULL, (chiffrer avant insertion dans la base)  
adresse VARCHAR(100) NOT NULL,  
permisDeConduire SET("AM","A1","A2","A","B1","B","BE","D1") NOT NULL,  
dateOptention DATE NOT NULL,  
paysOptention VARCHAR(20) NOT NULL,  
telephone CHAR(10),  
dateNaissance DATE,  
lieuNaissance VARCHAR(30),  
presentation TEXT,  
resauxSociaux SET("facebook","twitter","Instagram"),  
photo VARCHAR(255));  
  
t_vehicule (  
id INT AUTO_INCREMENT NOT NULL,  
marque VARCHAR(30) NOT NULL,  
modele VARCHAR(30) NOT NULL,  
kilometrage INT NOT NULL,  
prix FLOAT(10,2) NOT NULL,  
lieu VARCHAR(100) NOT NULL,  
tempsMin INT NOT NULL,  
tempsMax INT NOT NULL,  
photo VARCHAR(255),  
carburant VARCHAR(10),  
caracteristiqueTechnique TEXT,  
boiteAVitesse VARCHAR(6),  
option SET("Siège bébé","GPS","Climatisation","Porte-vélo","Coffre toit","Régulateur de  
vitesse","Limiteur de vitesse","Pneu neige","Chaine"),  
etat TEXT,  
conditionSupplementaire TEXT);
```

MongoDB

```
ARCHIVE :  
id INT AUTO_INCREMENT NOT NULL,  
nomPrenom VARCHAR(100),  
log JSON  
date DATE;
```

5 - Les routes

Listes des routes :

Inscription & Paramétrage

create a user : post
delete a user : delete
update a user : put

Connexion

connexion : post

Véhicule

create a location-user : post
delete a location-user : delete

create a location-offer : post
delete a location-offer : delete
update a location-offer : put

search a location : get

navigation : get

view cars-user : get

availability cars-offer : get

Archive

archive-history-user : post
view-history-user : get