

# **Node.js Fundamentaux: Creation des API**

*Regis Atemengue  
@regis\_ate  
[www.regisatemengue.com](http://www.regisatemengue.com)*

**Si vous avez besoin d'aide !!**

*Udemy Q/A Discussion*

*Envoyez moi sur Udemy*

*Twitter: @regis\_ate*

*Site Web: [www.regisatemengue.com/contacts](http://www.regisatemengue.com/contacts)*

# Ressources pour suivre le Cours

<https://github.com/atemengue/nodejs-api-fondamentaux>

**Diagrammes**

**Code de Base / Snippet du projet**

<https://github.com/atemengue/nodejs-api-fondamentaux>

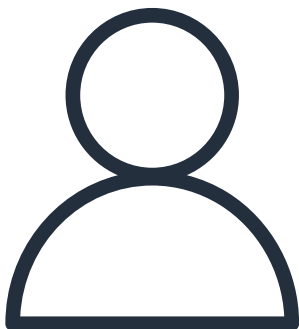
**Télécharger  
le code sur github**

**Accéder aux  
Snippets**

<https://github.com/atemengue/nodejs-api-fondamentaux>

**Node.js 22.12.0**

**Express 4.21.2**



Développeur

## SCENARIO

Vous travaillez pour l'entreprise **Propelize**

Vous avez des connaissances sur le langage de programmation Javascript

Vous n'avez jamais courstruit une API

**Propelize  
Entreprise**



Objectif: Lancer une application de location de véhicules

Location de voitures et de camionnettes

Application web

Applications mobiles

**Comment mettre en place notre environnement ?**

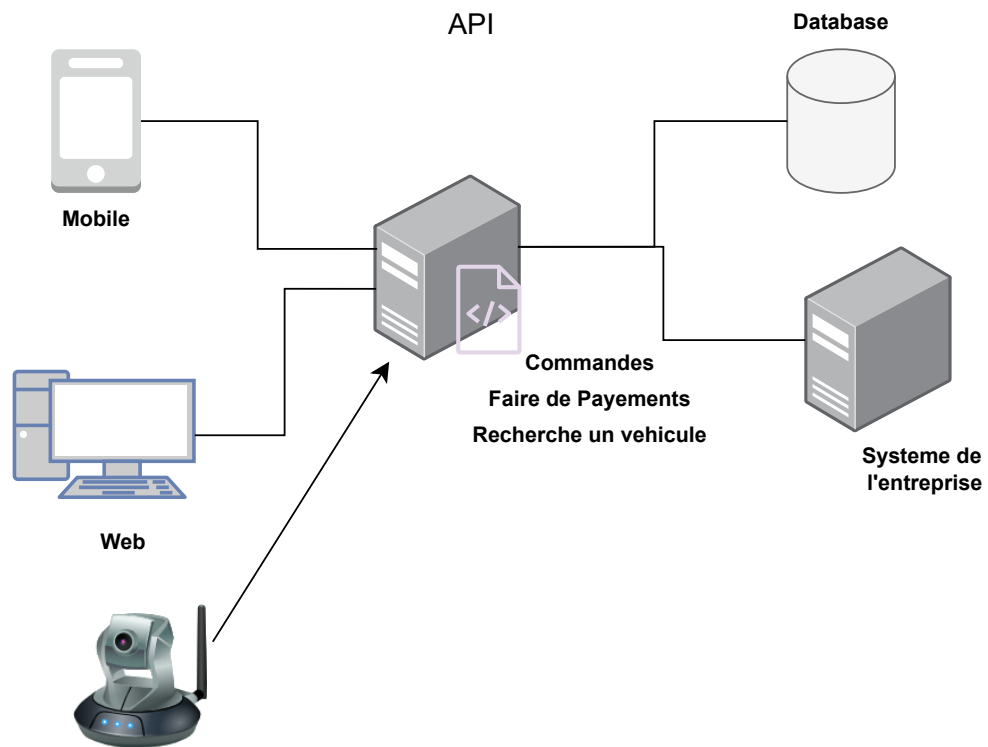
**De quels outils avons-nous besoin ?**

**Comment écrire du code pour Node.js ?**

**Pourquoi utiliser Node.js ?**

**Qu'est-ce qu'une API ?**

# Architecture Système





# Vehicule API

## *Méthodes API :*

- Obtenir / lire d' un véhicule
- Créer un véhicule
- Mettre à jour un véhicule
- Supprimer un véhicule
- Recherche par immatriculation
- Recherche par prix

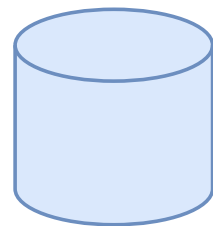
# Vehicule Service



- Obtenir / Lire les info un véhicule
- Créer un véhicule
- Mettre à jour un véhicule
- Supprimer un véhicule
- Recherche par immatriculation



Node.js Express.js



Mongo DB /  
PostgreSQL

# Qu'est ce que Node.js ?

## Un environnement d'exécution

- Exécute JavaScript en dehors du navigateur

Fournit

- des outils
- Paramètres
- Conditions d'utilisation



# Principales caractéristiques de Node.js

Non-blocking, asynchronous I/O

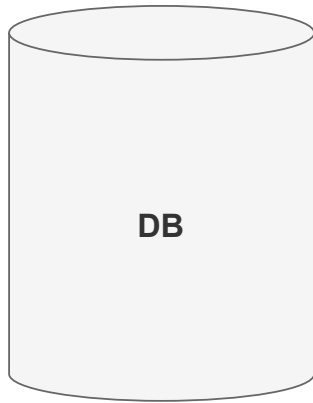
Moteur JavaScript V8

Architectures pilotées par événements

Gestionnaire de paquets NPM

**DEMO API**

## UTILISATION D'UNE BASE DE DONNEES



**Indépendance des services**

**Évolutivité**

**Isolation des données**

**Optimisation des performances**

**Flexibilité technique**

**RELATIONNELLE(SQL)**

**NON RELATIONNELLE(NoSQL)**



**NoSQL DB**

**MongoDB**

Prise en charge de différents types de modèles de données

Très flexible

**CouchDB**

**AzureStorage**

**AWS DynamonDB**

**SQL DB**

Modèle Strict de données

**MySQL**

**PostgreSQL**

**SQL Server**

## Choix de la DB

NoSQL DB	SQL DB
Structure de données flexible	Requêtes complexes
Évolutivité	Intégrité des données
Performances	Rapports et analyses
Développement simplifié	
Haute disponibilité et résilience	

## **INSTALLATION DE MONGO DB**

**ATLAS Cloud**

**Fichier.exe**

**Docker**

## **MULTI-CLOUD SERVICE ATLAS**

Initialisation

Creation du Cluster

## RELATIONNELLE(SQL)

FABRICANT	
ID	NOM
1	TESLA
2	TOYOTA

VEHICULES				
ID	IDFABRICANT	IMMATRICULATION	ANNEE	PRIX
1	2	ABC-23	2025	200000
2	1	OC5-3	2022	486520
3	1	TRP-54	2045	48684

# NoSQL Databases

VEHICULES	
Key	Value
de5f47ffe39d4291b0e25e6f7279036b Object	Object
de5f47ffe39d4291b0e25e6f7279036b Object	Object

```
{
  _id: ObjectId("de5f47ffe39d4291b0e25e6f7279036b"),
  marque: "Tesla",
  model: "Model Y",
  prix: 2000000,
  couleur: 'rouge'
}
```

```
{
  _id: ObjectId
  marque: "Te
  model: "Mod
  prix: 200000
  couleur: 'rou
  annee: 2025
}
```



```
l("de5f47ffe39d4291b0e25e6f7279036b "),  
sla",  
lel Y",  
0,  
ge',  
;
```





## Collections de Documents

Documents regroupés en collections

Pas de schéma strict

Chaque document peut avoir des champs différents



## Comprehension du Document

Paire Clé-Valeur

Champ-nom

Valeur

String

Number

Array

Document

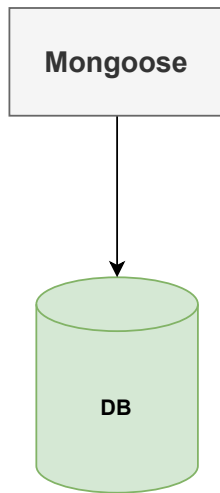
## CONFIGURATION ET CONNEXION MONGO DB

Create  
InsertOne()  
InsertMany()

Read  
find()  
findOne()

Update  
\$Set()

Delete  
DeleteOne()  
DeleteMany()



## Comprehension de Mongoose(ORM)

Definition des Schema

Validation des donnees

Mapping d'Objet

Facile à utiliser

## **Variables d'Environnement (ENV)**

**VEHICULE MODELE /SCHEMA**

## **CRUD AVEC MONGOOSE**