



CLINT Agency

Test Dev
NodeJS

Introduction

Ce test a pour but d'avoir un aperçu de tes compétences en NodeJS en vue du poste de développeur produit au sein de Clint Agency.

Le but de cet exercice est de réaliser une API avec des tests unitaires en NodeJS.

L'ensemble du code devra être, le plus possible, commenté pour faciliter la relecture.

Il est fortement déconseiller de copier/coller du code provenant d'internet.
Le but n'est pas de voir si tu sais faire du c/c ;-)

L'organisation, l'indentation, la clarté et la propreté du code auront le droit à une attention toute particulière.

Pré-requis

Voici une liste de package NPM à utiliser pour réaliser ce test :

- Express : <http://expressjs.com/>
- MongooseJS : <http://mongoosejs.com/>
- MochaJS : <https://mochajs.org/>

Livrable

Une fois l'exercice terminé, merci de retourner l'ensemble de votre code archivé accompagné d'un Readme indiquant les procédures à suivre pour installer, configurer et lancer le script à l'adresse suivante : **laurent@clintagency.com**

Exercice

Développement d'une API

Tu dois réaliser une API Rest qui va permettre de gérer des objets de type « Véhicule » et de type « Marque ».

Chaque « Véhicule » doit appartenir à une « Marque ». Une « Marque » peut avoir plusieurs « Véhicule ».

Par le biais de l'API, tu dois créer deux CRUD (Create, Read, Update, Delete), un pour les « Véhicule » et un pour les « Marque ».

Il doit également être possible d'associer un « Véhicule » à une « Marque ».

Développement des tests unitaires

Il t'es également demandé de réaliser les tests unitaires par le biais de « MochaJS » de chacune des routes des différents CRUD.

L'ensemble des tests doivent bien évidemment fonctionner à la livraison du code source.

Chaque « Véhicule » et « Marque » seront enregistré dans une base de données MongoDB (<https://www.mongodb.org/>). Penses à créer une BDD dédiée aux tests.

L'ensemble des échanges se feront au format JSON.

NB : Il n'est pas demandé d'avoir une interface graphique (type Back-office) pour gérer les objets. Seuls les logiciels « Postman » ou « Paw » seront utilisés pour vérifier le fonctionnement du code et les retours de l'API.

