

## Formation Node.js : Construire une API REST avec Express.js

<b>Durée :</b>	3 jours
<b>Public :</b>	Développeurs Javascript
<b>Pré-requis :</b>	Bonne connaissance de javascript et des fondamentaux de Node.js
<b>Objectifs :</b>	Apprendre à construire une API REST sécurisée avec Express.js
<b>Sanction :</b>	Attestation de fin de stage mentionnant le résultat des acquis
<b>Taux de retour à l'emploi:</b>	Aucune donnée disponible
<b>Référence:</b>	SIT102237-F
<b>Note de satisfaction des participants:</b>	4,93 / 5

### Comprendre l'architecture

- Architecture en couches : du monolythe au microservices
- Contraintes d'architecture des microservices
- Gestion de l'authentification centralisée dans une architecture microservices
- Intérêt d'une passerelle d'API
- Gestion centralisée des traces

### Développement d'API REST avec Express.js

- Express.js : principe, fonctionnalités, pré-requis
- Configuration du projet : packages et modules additionnels : Morgan, Helmet, Cors,...
- Configuration de morgan pour la gestion des logs de requêtes HTTP
- Organisation des couches du projet : routes, services, repositories
- Intérêt d'une couche DTO, utilisation d'un mapper d'objets : automapper ou autre
- Implémentation de contrôleurs REST : gestion des routes, traitement asynchrone
- Gestion des paramètres de méthodes et du mapping
- Gestion du download
- Gestion de l'upload, configuration
- Gestion des services et des transactions associées
- Gestion du cross origin avec Cors et restriction des domaines appelants
- Test de l'api REST avec Postman
- Écriture de tâches asynchrones, planification
- Cache web

**Atelier : Écriture de micro-services avec Express.js - Test des méthodes de services avec Postman ou autre**

### Documenter une API REST

Open API Specification (Swagger) : présentation, outil  
Utilisation de express-openapi  
Visualisation avec Swagger Editor  
Documentation du code Java, génération de javadoc

### **Atelier : Documentation de l'api**

## **Intercepter des requêtes et gérer les erreurs**

Gestion globale des exceptions  
Capture d'exceptions personnalisées  
Intercepteurs de requêtes/réponses

### **Atelier : Gestion des exceptions et implémentation d'intercepteurs**

## **Appeler d'autres API REST (écriture de clients)**

http\_request : méthodes et paramètres  
Ecriture de requêtes GET, POST, PUT, DELETE  
Gestion des paramètres et du corps de la requête  
Gestion des headers  
Gestion des réponses et utilisation de mappers

### **Atelier : Implémentation de clients pour un service REST**

## **Sécuriser l'API REST**

Configuration de Helmet  
Gestion des données d'entête  
Gestion des utilisateurs et des rôles : mise en place de JWT (passport-jwt)  
Hachage des mots de passe avec bcryptjs

### **Atelier : Intégration de Helmet, jsonwebtoken, dotenv et bcryptjs**

## **Tester une application Express.js avec Jest et SuperTest**

Stratégies de tests, types supportés  
Installation de jest, supertest et cross-env, configuration de l'application  
Mocking des couches de l'application  
Tests auto-configurés  
Exécution et reporting

### **Atelier : implémentation et exécution de tests avec Jest et supertest**