DAWAN

Plan de la formation

www.dawan.fr

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

Durée :	3 jours (21 heures)
Tarif inter-entreprises :	2 175,00 € HT (Présentiel) 1 740,00 € HT (Distanciel)
Public :	Développeurs Javascript
Pré-requis :	Bonne connaissance de javascript et des fondamentaux de Node.js
Objectifs :	Apprendre à construire une API REST sécurisée avec Express.js
Modalités pédagogiques, techniques et d'encadrement :	• Formation synchrone en présentiel et distanciel. • Méthodologie basée sur l'Active Learning : 75 % de pratique minimum. • Un PC par participant en présentiel, possibilité de mettre à disposition en bureau à distance un PC et l'environnement adéquat. • Un formateur expert.
Modalité d'évaluation :	• Définition des besoins et attentes des apprenants en amont de la formation. • Auto-positionnement à l'entrée et la sortie de la formation. • Suivi continu par les formateurs durant les ateliers pratiques. • Évaluation à chaud de l'adéquation au besoin professionnel des apprenants le dernier jour de formation.
Sanction :	Attestation de fin de formation mentionnant le résultat des acquis
Référence :	SIT102237-F
Note de satisfaction des participants :	4,93 / 5
Contacts :	commercial@dawan.fr - 09 72 37 73 73

Modalités d'accès :	Possibilité de faire un devis en ligne (www.dawan.fr, moncompteformation.gouv.fr, maformation.fr, etc.) ou en appelant au standard.
Délais d'accès :	Variable selon le type de financement.
Accessibilité :	Si vous êtes en situation de handicap, nous sommes en mesure de vous accueillir, n'hésitez pas à nous contacter à referenthandicap@dawan.fr, nous étudierons ensemble vos besoins

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
Ecriture 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