

Mis à jour le 26/07/2023

S'inscrire

Formation Hapi.JS

2 jours (14 heures)

Présentation

[hapi.js](#) (également appelé hapi) est un framework simple à utiliser, centré sur la configuration, avec un support intégré pour la validation des entrées, la mise en cache, l'authentification et d'autres fonctionnalités essentielles pour la création d'applications Web et de services. hapi permet aux développeurs de se concentrer sur l'écriture d'une logique d'application réutilisable dans une approche hautement modulaire et normative.

La version 17.x ne prend en charge que les nœuds v8.9.0 et supérieurs. Pour les versions plus anciennes du nœud, veuillez utiliser la version 16.x.

Comme tous nos programmes, notre formation porte sur la toute dernière version stable en date de cette plateforme ([hapi v20](#) à la date de l'article).

Objectifs

- Construire des serveurs API sécurisés
- Exploitez les plugins hapi.js pour mieux structurer votre codebase
- Simplifiez vos flux de travail de sécurité avec la fonctionnalité intégrée d'authentification et d'autorisation de hapi.js
- Assurer la fiabilité des applications avec des tests et une couverture de code
- Réduisez la complexité du code en utilisant une logique de validation réutilisable avec joi

Public visé

- Développeur
- Architecte
- Chefs de projet

Pré-requis

Avoir des connaissances sur Node.JS est un plus.

Programme de la formation hapi.js

Présentation de hapi.js

- Node.js, une condition préalable à l'apprentissage de hapi.js
- Comparaison avec d'autres frameworks
 - Séparation de la logique métier de la couche transport
 - L'écosystème
 - Petits modules
- Création de notre premier serveur hapi.js
 - Installer hapi
 - Notre premier serveur hapi
 - Plugins hapi
 - Configuration hapi
 - Le guide de style hapi

Ajout de fonctionnalités par des demandes de routage

- Routage du serveur
 - Configuration Route
 - Method
 - Path
 - Description
 - Validate
 - Pre
 - Handler
 - Cache
 - Algorithme de routage hapi
- Cycle de vie
 - Extension du cycle de vie des requêtes
- Objet de requête hapi
- L'interface de réponse
 - Gestionnaires personnalisés
- Servant des fichiers statiques avec inerte
- Modèles de service avec vision
 - Configuration de la vision
 - Les moteurs
 - Les chemins
 - compileMode

Structurer votre codebase avec des plugins

- Applications structurantes
- Plugins hapi
 - Fonctionnalité d'encapsulation dans les plugins
 - Options du plugin

- Exposer et consommer des plugins
 - Gestion des dépendances des plugins
 - Exposer les fonctionnalités du sandbox
 - Combiner des plugins
- L'écosystème du plugin hapi
 - Authentification
 - Génération de documentation
 - Enregistrement
 - Utilitaires
- Composer des applications à partir de plugins

Ajout de tests et importance de la couverture du code à 100%

- Pourquoi l'importance des tests est-elle négligée?
- Les avantages et l'importance du code de test
- Présentation des utilitaires de test de hapi
 - Installation du laboratoire et du code
 - Modules locaux versus globaux
 - Notre premier script de test
 - Approches de test
- Test d'applications hapi avec lab
 - Couverture de code
 - Linting

Sécurisation des applications avec authentification et autorisation

- Authentification
 - Configuration de l'authentification
 - Authentification de cookie
 - Authentification par un tiers
- Autorisation

La joi de la validation réutilisable

- Une introduction à la validation
- Introduction à joi
- Validation des routes hapi avec joi
 - Validation des réponses Route
- Génération de documentation

Rendre votre application prête à l'emploi

- Données persistantes
 - MongoDB
 - Le 'M' dans MVC
- Introduction à la mise en cache
 - Méthodes serveur
 - Mise en cache avec des méthodes de serveur

- Introduction aux logs
 - Applications de débogage
 - La console
 - Le débogueur de noeud
 - Le mode de débogage dans hapi
 - Profilage heapdumps
 - TV
- Conseil général de production
- Continuer l'apprentissage
 - Démarrer avec hapi.js repo GitHub
 - Hapijs.com
 - Makemehapi
 - La communauté hapi

Sociétés concernées

Cette formation s'adresse à la fois aux particuliers ainsi qu'aux entreprises, petites ou grandes, souhaitant former ses équipes à une nouvelle technologie informatique avancée ou bien à acquérir des connaissances métiers spécifiques ou des méthodes modernes.

Méthodes pédagogiques

Stage Pratique : 60% Pratique, 40% Théorie. Support de la formation distribué au format numérique à tous les participants.

Organisation

Le cours alterne les apports théoriques du formateur soutenus par des exemples et des séances de réflexions, et de travail en groupe.

Validation

À la fin de la session, un questionnaire à choix multiples permet de vérifier l'acquisition correcte des compétences.

Sanction

Une attestation sera remise à chaque stagiaire qui aura suivi la totalité de la formation.