

Editions ENI

Node.js

Exploitez la puissance
de JavaScript côté serveur

Collection
Expert IT

Table des matières

Les éléments à télécharger sont disponibles à l'adresse suivante :
<http://www.editions-eni.fr>
Saisissez la référence de l'ouvrage **EINOD** dans la zone de recherche
et validez. Cliquez sur le titre du livre puis sur le bouton de téléchargement.

Avant-propos

1. Pourquoi ce livre ?	11
2. À qui s'adresse cet ouvrage ?	12
3. Structure de l'ouvrage	12

Chapitre 1

Présentation

1. Introduction	15
2. Définition de Node	16
3. Historique	16
4. Success-stories	18
5. Programmation orientée composant	21

Chapitre 2

Installation

1. Introduction	23
2. GNU/Linux	23
2.1 n : simple et efficace	24
2.2 Via les paquets	25
2.2.1 Ubuntu/Debian	25
2.2.2 Gentoo	26
2.2.3 ArchLinux	26
2.2.4 RHEL, CentOS, Fedora, etc.	26

2.3	Via les sources	27
3.	OpenBSD et FreeBSD	27
4.	Mac OS	28
5.	Windows	28
6.	Vérification de l'installation	30

Chapitre 3

Gestionnaire de paquets npm

1.	Introduction	31
1.1	Qu'est-ce qu'un paquet ?	31
1.2	Exemple d'installation	32
1.3	Registre npmjs.org	32
1.4	Paquet global	33
2.	Recherche du bon paquet	34
2.1	Recherche en ligne de commande	34
2.2	Recherche sur npmjs.org	35
2.3	Critères de confiance	36
2.3.1	Popularité sur GitHub	36
2.3.2	Intégration continue	36
3.	Versionnage	37
3.1	Numéro de version	37
3.2	Contrainte de version	38
4.	Gestion des dépendances	39
4.1	Ajout	40
4.1.1	Dépendance de production	40
4.1.2	Dépendance optionnelle	40
4.1.3	Développement	41
4.2	Mise à jour	41
4.3	Suppression	42
4.4	Listage des dépendances	42

- 4.5 Installation des dépendances manquantes 43
- 5. Publication d'un paquet 43
 - 5.1 Création d'un compte sur le registre 44
 - 5.2 Saisie des métadonnées 45
 - 5.3 Exécutables 45
 - 5.4 Publication 46
- 6. Gestion d'un paquet publié 47
 - 6.1 Mise à jour 47
 - 6.2 Ajout des mainteneurs 48

Chapitre 4
Concepts

- 1. Introduction 49
- 2. Modules 49
 - 2.1 Modules de base 51
 - 2.1.1 Stabilité 52
 - 2.1.2 Exemple 52
- 3. Variables globales 53
 - 3.1 global 53
 - 3.2 process 54
 - 3.2.1 Événements 54
 - 3.2.2 Signaux 54
 - 3.2.3 Flux 55
 - 3.2.4 Contexte d'exécution 56
 - 3.2.5 Métriques 57
 - 3.3 console 58
 - 3.4 Buffer 60
 - 3.4.1 Construction 60
 - 3.4.2 Conversion 61
 - 3.4.3 Manipulation 61
 - 3.5 require() 63

3.6	__filename et __dirname	63
3.7	module	63
3.8	exports	63
3.9	Minuteurs	64
4.	Programmation asynchrone	64
4.1	API asynchrone dans Node	65
4.2	Programmation événementielle	69
4.3	Boucle d'événements	71
4.3.1	Fonctionnement simplifié	71
4.3.2	Fonctionnement détaillé	72

Chapitre 5

Flux

1.	Introduction	75
1.1	Qu'est-ce qu'un flux ?	75
1.1.1	Mode de flux	76
1.2	Pourquoi utiliser un flux ?	77
2.	Construction	78
2.1	Readable	78
2.2	Writable	80
2.3	Duplex	81
2.4	Transform	81
3.	Utilisation	83
3.1	Lecture	83
3.1.1	Mode flot	83
3.1.2	Mode à la demande	84
3.1.3	Sélection du mode de lecture	85
3.2	Écriture	85
3.3	Connexion	85

- 4. Omniprésence dans Node 86
 - 4.1 Réseau 86
 - 4.2 Fichiers 87
- 5. Implémentations et compatibilité..... 87
 - 5.1 Support des flux version 1 88
 - 5.2 S'affranchir de la version de Node 88

Chapitre 6

Manipulation de fichiers

- 1. Introduction 89
- 2. Manipulation de chemin 90
- 3. Manipulation de dossiers 92
- 4. Manipulation de fichiers 94
 - 4.1 Métadonnées..... 94
 - 4.2 Surveillance..... 100
- 5. Manipulation de liens symboliques..... 101

Chapitre 7

Promesses

- 1. Introduction 103
- 2. Callbacks vs promesses..... 103
- 3. Notion de promesse 106
 - 3.1 États d'une promesse 108
 - 3.2 Similarité avec du code synchrone 108
 - 3.3 Gestion des erreurs 110
- 4. Création d'une promesse 111
 - 4.1 À la main..... 111
 - 4.2 À partir d'une fonction Node..... 112
 - 4.3 À partir d'un événement 113

- 5. Intégration avec Node 114
- 6. Intégration avec les générateurs 115

Chapitre 8

Application en ligne de commande

- 1. Introduction 119
- 2. Gestion des paramètres 120
- 3. Testabilité 122
- 4. Exécution directe 123
- 5. Installation du programme 124
- 6. Exemple complet 125

Chapitre 9

Application web

- 1. Introduction 129
- 2. Générateur Express 130
- 3. Application 132
- 4. Architecture 132
- 5. Requête 133
- 6. Réponse 134
- 7. Distribution de fichiers statiques 135
- 8. Routage 136

Chapitre 10

Bases de données

1. Introduction	139
2. SQL	140
2.1 PostgreSQL	140
3. NoSQL	141
3.1 MongoDB	142
3.1.1 Mongoose	142
3.2 Redis	143
3.3 LevelDB	144

Chapitre 11

Tests

1. Introduction	149
2. Lanceur de tests - Mocha	151
2.1 Installation	151
2.2 Création d'une suite de tests	152
3. Assertions - Chai	156
4. Espions et bouchons - Sinon.js	160
4.1 Espions	160
4.2 Bouchons	162

Chapitre 12

Outils de développement

1. Introduction	165
2. Automatisation de tâches - Gulpjs	166
2.1 Installation	166
2.2 Création de tâches	167

3. Analyse du code - JSHint	171
3.1 Installation et utilisation	172
3.2 Configuration	173
3.3 Environnements	174
3.4 Options générales	175
3.5 Options de mise en conformité	178
3.6 Options assouplissantes	186
4. Redémarrage automatique - node-dev	197
5. Browserify	198

Chapitre 13

Débogage

1. Introduction	201
2. Ajout de traces	201
3. Node-Inspector	203
3.1 Installation et lancement	203
3.2 Contrôle de l'exécution	205
3.3 Inspection	209
3.4 Intervention	211

Chapitre 14

Mise en production

1. Introduction	215
2. Forever	215
3. Supervision	217
4. Recettes	218
4.1 Reverse proxy	218
4.1.1 Avec Apache	219
4.1.2 Avec Nginx	220

4.2	Intégration avec systemd (GNU/Linux)	220
4.2.1	Fichier de configuration	221
4.2.2	Activation	222
4.2.3	Utilisation	222
4.3	Intégration avec SaltStack	222
4.3.1	Installation et déploiement	223
4.3.2	npm via SaltStack	224
	Index	225

Editions ENI

AngularJS

Développez aujourd'hui
les applications web de demain

Collection
Expert IT

Table des matières

Les éléments à télécharger sont disponibles à l'adresse suivante :
<http://www.editions-eni.fr>
Saisissez la référence ENI de l'ouvrage **EIANG** dans la zone de recherche
et validez. Cliquez sur le titre du livre puis sur le bouton de téléchargement.

Avant-propos

1. Pourquoi ce livre ?	11
2. À qui s'adresse cet ouvrage ?	12
3. Structure de l'ouvrage	12
4. Remerciements	13

Chapitre 1

Introduction à AngularJS

1. Introduction	15
2. Single Page Application	16
3. AngularJS	20
3.1 Organisation de l'application	20
3.2 Mécanismes techniques	22
3.3 Versions	23

Chapitre 2

Ma première application AngularJS

1. Introduction	25
2. Référencer AngularJS	25
3. Définir la portée d'AngularJS	26

2 **AngularJS**

Développez aujourd'hui les applications web de demain

4. Création d'une application	27
4.1 Modèle	27
4.2 Contrôleur	27
4.3 Vue	28
5. Application complète	29

Chapitre 3 **Fondamentaux d'AngularJS**

1. Introduction	31
2. Initiation aux modules	31
3. Contrôleur et modèle	32
4. Vues et Binding	35
4.1 Binding	35
4.2 Afficher et cacher des éléments	39
4.3 Liens	41
4.4 Styles	42
4.5 Listes	43
4.6 Évènements HTML	46
4.7 Réagir aux changements du modèle	48
4.8 Directives	51
4.9 Expressions	52
4.10 Communication dans une application	53
4.11 Unobtrusive JavaScript	55
5. Formatage des données	57
5.1 Utilisation des filtres	58
5.2 Création de filtres	65
5.3 Exemples de filtres	66

Chapitre 4 Structurer son application

1. Introduction	73
2. Modules	73
2.1 Création et utilisation d'un module	74
2.2 Combien de modules utiliser ?	75
2.3 Chargement d'un module	76
3. Injection de dépendances	77
3.1 Déclaration et résolution des dépendances	77
3.2 Utilisation du service d'injection	79
4. Services	81
4.1 Utilisation de services	81
4.2 Création de services	82
4.3 Providers	85
5. Constantes	87
6. Structure des fichiers	88
6.1 Application simple	89
6.2 Application complexe	90

Chapitre 5 Promises et requêtage HTTP

1. Introduction	93
2. Promise, la fin des callbacks	93
2.1 Promise	95
2.2 Création d'une promise	96
2.3 Interactions avec une promise	99
3. Communication avec le serveur	101
3.1 Service \$http	101
3.1.1 Effectuer une requête HTTP	101
3.1.2 Méthodes raccourcies	103

3.1.3	Provider <code>\$httpProvider</code>	106
3.1.4	Headers HTTP	106
3.1.5	Transformation de la requête et de la réponse	107
3.1.6	Cache	109
3.1.7	Intercepteurs	110
3.1.8	Timeout	112
3.1.9	Encapsulation des appels dans un service	114
3.2	Service <code>\$resource</code>	115
3.2.1	Création d'une ressource	116
3.2.2	Récupération des ressources	116
3.2.3	Interaction avec une ressource	119
3.2.4	Méthodes spécifiques	121
3.2.5	Encapsulation dans un service	122

Chapitre 6

Naviguer dans une application AngularJS

1.	Introduction	123
2.	Navigation au sein de l'application	123
2.1	Déclaration des routes	124
2.1.1	Passage de paramètres dans une route	126
2.1.2	Exécution d'une action avant la navigation	127
2.2	Définir l'emplacement des vues	129
2.3	Naviguer dans l'application	130
2.3.1	Liens de navigation	130
2.3.2	Service <code>\$location</code>	130
2.4	Évènements de navigation	134
2.5	Mode HTML 5	136
3.	Navigation hors de l'application	138

Chapitre 7
Formulaires et validation

- 1. Introduction 139
- 2. Formulaires 140
 - 2.1 Directive form 140
 - 2.2 Soumission d'un formulaire 142
 - 2.2.1 Directive ngSubmit 143
 - 2.2.2 Directive ngClick 143
 - 2.2.3 ngClick ou ngSubmit ? 144
- 3. Validation 145
 - 3.1 Déclaration des règles de validation 145
 - 3.2 Contrôle de validation des champs 147
 - 3.3 Détection des modifications 150
 - 3.4 Module ngMessages 151
- 4. Formulaires avec listes 153
- 5. Exemple 155

Chapitre 8
Internationalisation et localisation

- 1. Introduction 159
- 2. Internationalisation des données non textuelles 160
 - 2.1 Règles de localisation 160
 - 2.2 Filtres et directives 163
- 3. Internationalisation des données textuelles 167
 - 3.1 Principe général 167
 - 3.2 Création des fichiers de ressources 167
 - 3.3 Création du service 169
 - 3.4 Création du filtre 170
 - 3.5 Gestion du multilinguisme 170

Chapitre 9

Les directives

1. Introduction	175
2. Création d'une directive	176
3. Nommage d'une directive	177
4. Restriction des modes de déclaration	178
5. Utilisation des templates	180
5.1 Propriété template	180
5.2 Propriété templateUrl	182
5.3 Préchargement des templates	183
5.4 Insertion du template dans la vue	183
5.5 Mode d'insertion	184
6. Transclusion	186
7. Phases d'exécution d'une directive	189
7.1 Link	194
7.2 Compile	200
8. Définition des priorités	201
8.1 Propriété priority	201
8.2 Propriété terminal	205
9. Utilisation des scopes	207
9.1 Scope courant	207
9.2 Nouveau scope	208
9.3 Scope isolé	209
10. Contrôleurs	217
10.1 Déclaration d'un contrôleur	217
10.2 Déclaration d'un lien vers une directive	218
10.3 Communication avec une directive	220
10.4 Interagir avec la directive ngModel	221

- 11. Exemples de directives 224
 - 11.1 Intégration d'un composant jQuery 224
 - 11.2 Création d'un composant de rating 229
 - 11.3 Création d'un composant accordéon 233
 - 11.4 Optimisation avec compile 238

Chapitre 10
Notions avancées

- 1. Introduction 247
- 2. Fonctionnement du binding 248
 - 2.1 Cycle digest 248
 - 2.2 Quand et comment déclencher un cycle digest ? 253
- 3. Performances et optimisations 255
 - 3.1 JavaScript : un langage event-driven 255
 - 3.2 Trop de watch tue le watch 257
 - 3.2.1 Avoir une fonction watch simple 259
 - 3.2.2 Privilégier le ngChange au watch 260
 - 3.2.3 Privilégier le binding one-time 262
 - 3.3 Limiter l'utilisation de filtres dans les vues 262
 - 3.4 Manipuler le DOM dans les directives 264
 - 3.5 Désactiver les informations de debug 267
 - 3.6 Comment anticiper les problèmes de performance ? 268
- 4. Animation de l'application 268
 - 4.1 Animations CSS 3 268
 - 4.1.1 Transitions 269
 - 4.1.2 Keyframes 270
 - 4.2 Module ngAnimate 273
- 5. Insertion de contenu HTML 279
- 6. Indexation par les moteurs de recherche 282
 - 6.1 Indiquer que l'application web possède du contenu AJAX 283
 - 6.2 Renvoyer une page statique depuis le serveur 283

6.3 Configuration de l'application AngularJS	284
7. Rendre son application accessible	285

Chapitre 11

Tester son application

1. Introduction	287
2. Tests unitaires.	288
2.1 Outils.	288
2.2 Karma	289
2.3 Fonctionnement de Jasmine	289
2.4 Vérification du résultat d'un test.	291
2.5 Installation du framework Jasmine.	292
2.6 Test simple	293
2.7 Espions	294
2.8 Module ngMocks	296
2.9 Test d'un contrôleur simple.	296
2.10 Tester une directive	298
2.11 Tester un service.	300
3. Introduction aux mocks.	301
3.1 Mocker \$http	301
3.2 Mocker \$httpBackend	302
4. Tests d'intégration	308
4.1 Utilisation de Protractor	308
4.2 Test simple	309
4.3 Manipulation du DOM	311
4.4 Configuration avancée.	312

Chapitre 12
Au-delà d'AngularJS

- 1. Introduction 315
- 2. Environnement JavaScript compilé..... 315
 - 2.1 Fonctionnement de TypeScript..... 317
 - 2.2 Intégration avec AngularJS 323
- 3. Choix de l'environnement de développement 325
- 4. Déboguer une application AngularJS 327
 - 4.1 Developer Tools 327
 - 4.2 Utilisation de la console..... 330
 - 4.3 Batarang 333
- 5. Rendre son application utilisable hors connexion 338
 - 5.1 Cache HTML 5 338
 - 5.1.1 Mettre en cache des ressources 339
 - 5.1.2 Comment fonctionne le cache ? 341
 - 5.1.3 Interagir avec la mise en cache 342
 - 5.2 Storage HTML 5..... 343
 - 5.2.1 WebSQL 343
 - 5.2.2 Web Storage 344
 - 5.2.3 IndexedDB 345
- 6. Création d'applications natives 347
 - 6.1 Principe de Cordova 348
 - 6.2 Accès aux fonctionnalités de l'appareil 349
 - 6.3 Intégration avec une application AngularJS 350

- Index 353