

Angular - Fondamentaux

 3 j (21 heures)

Ref : ANGU



Public

Développeurs Javascript, architectes et chefs de projets web



Pré-requis

Bonne connaissance de Javascript, pratique du développement web



Moyens pédagogiques

Formation présentielle
Présentation des concepts, discussion technique, démonstrations, exercices et TP
Un poste informatique par stagiaire connecté à internet, à une imprimante en réseau et au réseau informatique
Les salles sont équipées d'un tableau interactif ou d'un vidéoprojecteur et d'un paperboard
Support de cours fourni à chaque stagiaire



Modalités de suivi et d'évaluation

Questionnaire d'évaluation de la satisfaction en fin de stage, feuille de présence émarginée par demi-journée par les stagiaires et le formateur, Attestation de fin de formation

Description

Action Collective FAFIEC N°27987

AELION a été retenue par le FAFIEC dans le cadre de l'action collective "Développer avec les langages .NET JAVA et C++". Si vous êtes salarié d'une ESN/SSII cotisant au FAFIEC, vous pouvez bénéficier d'une prise en charge à 100% du coût de cette formation dans la limite des fonds disponibles.

En passe de devenir la référence des frameworks front-end dans le développement d'applications web, Angular apporte, au-delà du support ES6 et du typescript, un véritable écosystème qui permet d'améliorer les performances des applications mais aussi la productivité des équipes de développement. Nouvelle approche de conception, capacités d'industrialisation et réutilisation accrues, intégration des standards du développement web et outillage spécialisé rendent cette plateforme plus attractive et l'impose comme la solution de prédilection pour les projets applicatifs web.

Objectifs

- ⊗ Concevoir une Single Page Application
- ⊗ Etre productif avec Angular
- ⊗ S'intégrer avec d'autres bibliothèques Javascript
- ⊗ Eviter les pièges pour garder un code structuré et maintenable

Programme détaillé

Rappel optionnel : ES6 et Typescript

Qu'est-ce qu'Angular ?

Vue d'ensemble de l'architecture

- ▶ Modules
- ▶ Composants
- ▶ Templates
- ▶ Metadonnées
- ▶ Data Binding - Liaison de données dynamique
- ▶ Directives
- ▶ Services
- ▶ Injection de dépendance

Introduction en pratique

- ▶ Création d'une mini-application
- ▶ ng-cli l'outil ad-hoc en ligne de commande
- ▶ Les décorateurs Typescript
- ▶ Equivalence avec Javascript

Organisation des composants

Syntaxe des templates

- ▶ Interpolation
- ▶ Liaison de propriétés
- ▶ Liaison d'événements
- ▶ Variables locales
- ▶ Directives structurelles ngIf, ngFor, ngSwitch...

Communication inter-composants

- ▶ Entrées
- ▶ Sorties
- ▶ Alternatives avancées

Les formulaires

- ▶ Pilotés par template
- ▶ Pilotés par le code

Injection de dépendances et services

- ▶ Injecter un service
- ▶ Fournir un service
- ▶ Partager un service
- ▶ Configuration du moteur de DI

Directives

- ▶ Les directives fournies par Angular
- ▶ Ecrire une directive

Pipes

- ▶ Les pipes fournies par Angular
- ▶ Ecrire son pipe

Tester son code

- ▶ Tests unitaires
- ▶ Tests d'intégration

Inclusion de bibliothèques externes

Http

- ▶ Utilisation du module Http
- ▶ Bouchonnage

Routeur

Utilisation avancée

- ▶ Style d'encapsulation CSS
- ▶ Cycle de vie et hooks

- ▶ Zones
- ▶ Animations
- ▶ Transclusion
- ▶ Fonctionnement interne du data-binding
- ▶ Création d'un composant de façon programmatique
- ▶ Rendu côté serveur avec Angular Universal
- ▶ Compilation statique de l'application' (AOT)