
Programme de Formation

Angular : Concevoir et structurer des front-ends professionnels

Description

Formation pratique pour maîtriser Angular et TypeScript, créer des applications web modulaires, gérer formulaires, composants et API, et préparer vos projets front-end à la production.

Points Forts

Apprenez à créer des applications web robustes et modulaires avec Angular grâce à des ateliers pratiques sur la CLI, les composants, la modularisation, les formulaires, RxJS et les tests pour améliorer productivité et qualité.



Public visé

- Développeurs front-end
- Développeurs web full-stack
- Chefs de projet techniques
- Architectes web



Objectifs pédagogiques

- Comprendre les concepts clés d'Angular et ses dernières évolutions
- Développer une application web Angular avec Javascript et Typescript
- Tester et sécuriser son développement
- Créer des composants réutilisables et modulaires
- Utiliser les API Angular
- Connaître les bonnes pratiques de développement

Organisation

Durée : 21 heures

Mode d'organisation : Mixte

Contenu pédagogique



Description

Jour 1 :

Comprendre les concepts clés d'Angular et ses dernières évolutions

- JS, EcmaScript et TypeScript
- Notion de Single Page Application
- Les Framework JS (Vue, React, Angular)
- Créer une SPA "from scratch" en JS
- Exemple de travaux pratiques : Le participant doit développer une mini-application SPA (ex : liste de tâches) avec JavaScript natif, permettant d'ajouter et supprimer des éléments dynamiquement.

Développer une application web Angular avec Javascript et Typescript

- Commandes Angular-CLI et scaffolding
- Structure d'un workspace Angular et outils intégrés



- Ecosystème Angular et librairies annexes
- Environnement de développement (IDEs et extensions utiles)
- Pratique
- Création d'un workspace Angular
- Utilisation des commandes Angular CLI
- Exemple de travaux pratiques : le participant doit initialiser un projet Angular, créer un premier composant via la CLI et visualiser le rendu dans le navigateur.

Jour 2 :

Identifier et utiliser les fondamentaux du Framework Angular et ses nouveautés

- Les syntaxes de base avec angular
- Les modules
- Les composants et les templates
- Le data binding
- Les directives
- Les pipes
- Les services
- Pratique
- Création de composants standalone ou modulaires
- Découvrir les directives de comportement (hidden, class, ...) et créer une directive
- Découvrir les directives de structures (*ngif, *ngfor, ...) et structures de contrôle (@if, @for...)
- Découvrir les pipes (dates, chaînes, etc) et créer un pipe
- Exemple de travaux pratiques : participant doit créer un composant de liste d'utilisateurs avec des directives *ngIf et *ngFor, et un pipe de formatage pour l'affichage.

Créer des composants réutilisables et modulaires

- Rappels sur le principe des modules Angular ainsi que l'approche standalone
- Installation et utilisation de librairies (utilisation d'une librairie de composants graphiques)
- Création de modules et patterns connus (Shared Module pattern)
- Communication entre composants
- Fonctionnalités de Routing et chargement de modules
- Pratique
- Installer Angular Material et créer un thème ou module pour regrouper les éléments graphiques
- Découper son application en plusieurs modules à partir d'un diagramme UML
- Créer un Shared Module pour partager les fonctionnalités transverses de son application
- Exemple de travaux pratiques : le participant doit créer un module partagé avec des composants réutilisables (boutons, modales) et intégrer Angular Material avec un thème personnalisé.

Jour 3 :

Utiliser les API Angular

- Formulaires
- Template driven forms vs Reactive Forms
- Services
- Notion d'injection de dépendance
- Utiliser le module HttpClient
- Créer et utiliser ses propres services
- Pratique
- Création d'un Reactive Form
- Consommation d'une API REST (introduction RxJS)
- Exemple de travaux pratiques : le participant doit créer un formulaire réactif de création d'utilisateur et consommer une API REST pour enregistrer les données.

Tester et sécuriser son développement

- Outils et API de test Angular (Protractor, Karma, Testbed,)
- Ecrire des tests
- Outils annexes
- Pratique
- Tester un composant
- Tester un service
- Exemple de travaux pratiques : le participant doit écrire un test unitaire pour un service de gestion de tâches et vérifier son bon fonctionnement avec Jasmine

Connaître les bonnes pratiques de développement

- Outils de documentation
- Mises à jour Angular
- Générer une documentation à partir de la JS DOC et d'outils comme COMPODOC
- Exemple de travaux pratiques : le participant doit documenter un composant à l'aide de commentaires JSDoc et générer une documentation HTML avec COMPODOC.

★ **Prérequis**

- Avoir des connaissances pratiques sur les technologies du Web, les outils actuels de développement Front-End et JavaScript
- Un questionnaire de positionnement sera réalisé au préalable pour adapter la formation à vos besoins



Modalités pédagogiques

- Exercices concrets
- Cas pratiques
- Quiz d'évaluation des connaissances



Moyens et supports pédagogiques

- Mise à disposition de postes de travail
- Vidéoprojecteur, tableau blanc, supports numériques
- Documentation pédagogique numérique



Modalités d'évaluation et de suivi

- Positionnement en amont de la formation :
 - Un quiz de consolidation des pré-requis sera administré en amont de la formation
- Suivi « pendant » :
 - Feuilles de présence
 - Exercices pratiques
 - Évaluation « fin de formation »
 - Évaluation des acquis en fin de formation
 - Formulaires d'évaluation de la formation
- Évaluation à froid :
 - Suivi post-formation : Questionnaire de satisfaction à j+30