
Programme de Formation

Angular avancé : Concevoir et optimiser des applications web complexes

Description

Approfondissement d'Angular pour maîtriser les concepts avancés, l'architecture, les formulaires, RxJS, tests et optimisation dans des applications web complexes.

Points Forts

Approche pratique centrée sur Angular avancé : modularisation, RxJS, formulaires réactifs, tests unitaires, optimisation de performance et intégration d'outils pour structurer vos applications web évolutives.



Public visé

- Développeurs JavaScript et Angular



Objectifs pédagogiques

- Connaître l'architecture et les fonctionnalités avancées d'Angular
- Maîtriser les grands principes de la programmation réactive
- Tester et sécuriser son développement
- Développer une application web dynamique et performante
- Réaliser des tests unitaires

Organisation

Durée : 21 heures

Mode d'organisation : Mixte

Contenu pédagogique



Description

Jour 1 :

Connaître l'architecture et les fonctionnalités avancées d'Angular

- Comprendre entre application modulaire et composants standalone
- Configurer l'application avec des standalone components
- Créer un premier composant standalone
- Exemples de travaux pratiques : Le participant crée une application Angular simple configurée avec un composant standalone, et développe ce composant pour afficher un message personnalisé dans l'interface utilisateur.

Maîtriser les grands principes de la programmation réactive

- Comprendre la difficulté de la gestion des états
- Comment Signal fonctionne
- Créer un composant utilisant Signal
- Signal et les effets de bord
- Tester un composant avec Signal
- Exemples de travaux pratiques : Le participant développe un composant Angular utilisant Signal



pour gérer dynamiquement l'état d'un élément affiché, puis réalise des tests unitaires pour valider le comportement réactif du composant.

Jour 2 :

Développer une application web dynamique et performante

- Utiliser la CLI pour ajouter une librairie
- Créer un module spécifique pour l'intégration de Material
- Customiser une feuille de style pour Material
- Ajouter le support PWA à une application Angular
- Exemples de travaux pratiques : Le participant intègre Angular Material via la CLI, crée un module dédié pour ce framework, personnalise son thème graphique, puis active les fonctionnalités PWA dans une application Angular existante.

Créer une librairie Angular

- Retour sur la richesse et la puissance des Directive Angular
- Utiliser la CLI Angular pour créer la structure d'une librairie
- Comprendre l'API publique
- Ajouter une directive à la librairie et la déployer
- Ajouter la librairie à un développement existant
- Exemples de travaux pratiques : Le participant crée une librairie Angular via la CLI, y développe une directive personnalisée, déploie la librairie, puis l'ajoute et l'utilise dans une application Angular existante.

Jour 3 :

Tester et sécuriser son développement

- Le principe des Token Angular
- Intercepter les requêtes http avec HTTP_INTERCEPTOR
- Agir avant le chargement du premier composant avec APP_INITIALIZER
- Comprendre et bien utiliser les interfaces OnInit, OnDestroy, ...
- Exemples de travaux pratiques : Le participant crée un intercepteur HTTP pour gérer l'authentification, configure une initialisation préalable à l'application, et implémente des méthodes liées au cycle de vie des composants pour gérer les ressources.

Réaliser des tests unitaires

- Comprendre le pattern Observer
- Utiliser les opérateurs from, of pour générer des Observables
- Souscrire efficacement à un Observable
- Comprendre les Subject
- Exemples de travaux pratiques : Le participant écrit des tests unitaires vérifiant le comportement d'un service basé sur RxJS, crée et teste un composant Angular réactif utilisant des Observables, et simule différentes sources de données à l'aide de Subject.

Utiliser le pipeline d'opérateurs RxJS

- Comprendre les opérateurs RxJS : map, switchMap, mergeMap,
- Choisir le bon opérateur en fonction du contexte d'exécution
- Envoyer plusieurs requêtes HTTP en parallèle ou enchaînées
- Agréger les résultats de plusieurs flux de manière cohérente
- Exemples de travaux pratiques : Le participant implémente un service Angular qui interroge plusieurs endpoints via HttpClient, combine les réponses à l'aide des opérateurs RxJS, et affiche le résultat final dans un composant Angular.

★ **Prérequis**

- Connaissances en JS et concepts MVC
- Expérience préalable sur des applications Angular (concepts de base)
- 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