**Formation**

**Framework Angular 9**

Angular est devenu une référence en développement Frontend, c’est un Framework Javascript complet qui structure vos applications. De nos jours nous ne travaillons plus avec Javascript comme avant, des Frameworks comme Angular tire parti de la puissance du langage Javascript en utilisant TypeScript.

Mis à part quelques spécificités acquises au cours de l’évolution du Framework cette formation reste compatible Angular 4, 5, 6, 7 et 8.

**3 jours**

**Public concerné**

La formation s’adresse aux développeurs n’ayant pas encore ou peu travaillé avec Angular.

**Objectifs Pédagogiques :**

Le but de la formation est de donner les clés de la compréhension de la structure d’un projet Frontend Angular et de maitriser les bases de la création de pages et de formulaires, les bases de la communication avec un Backend et la compréhension de la programmation réactive. A l’issue de la formation le développeur sera capable d’intégrer un projet Build ou Run en Angular.

**Prérequis :**

* Structure d’un projet MVC
* HTML
* XML
* Javascript
* GIT

**Intervenant**

► **Jacques BUCHAILLOT**

ARCHITECTE APPLICATIF

**Programme**

**Présentation théorique d’Angular et premières manipulations**

* Théorie
  + Présentation d’Angular
  + L’interface de commandes Angular CLI
  + L’outil Visual Studio Code
  + Le langage TypeScript
  + La structure d’une application Angular
    - Définition d’un module, d’un composant et d’une directive
    - Définition du routing
* Pratique
  + Créer une nouvelle application « Gérer mes tâches »
  + Accéder à l’application et au module principal : routing
  + Créer le composant « Liste de tâches »
  + Définir la directive d’attribut « Surligner la tâche »
  + Créer le composant « Tâche » et l’intégrer à la liste : directive avec template

**Formulaires, Data binding et communication entre les composants**

* Théorie
  + Communication entre les composants
    - Du parent vers l’enfant
    - De l’enfant vers le parent
    - Introduction aux services
* Pratique
  + Créer le formulaire « Saisir une tâche »
    - Saisir les données
    - Mettre à jour le DOM en fonction des données saisies
  + Ajouter l’action au bouton « Ajouter une tâche »
    - Ajouter une tâche à la liste
  + Déclencher l’évènement « Supprimer une tâche »
    - Supprimer la tâche de la liste

**Services & Programmation réactive**

* Théorie
  + Services
    - Injecter un service dans un module
    - Utiliser des Interfaces
    - Se connecter au Backend : HttpClient
  + RxJs
    - Observables et souscripteurs
* Pratique
  + Créer le service « taches.service »
  + Brancher le service au module « taches.module »
  + Appeler le service dans le composant « Liste de tâches »

**Mocks, tests unitaires, fichiers de configuration et Guards**

* Théorie
  + Tester son application : quoi tester et comment ?
  + Gérer la configuration de plusieurs environnements
  + Droits d’accès aux composants : les Guards
* Pratique
  + Tester la génération des composants
  + Tester le service « taches.service.ts »
  + Créer une configuration pour un environnement de DEV et de PROD
* Exercices supplémentaires
  + Modifier une tâche
    - Ajouter un bouton "Modifier" sur chaque tâche
    - Ouvrir un écran de modification
    - Valider les modifications
  + Ajouter un message de notification en bas de page qui affiche les messages suivants :
    - Tâche <libellé> ajoutée avec l'identifiant <id>
    - Tâche <libellé> supprimée
    - Tâche <libellé> modifiée
    - Les erreurs d'accès à l'API