

Architecture du Framework Angular

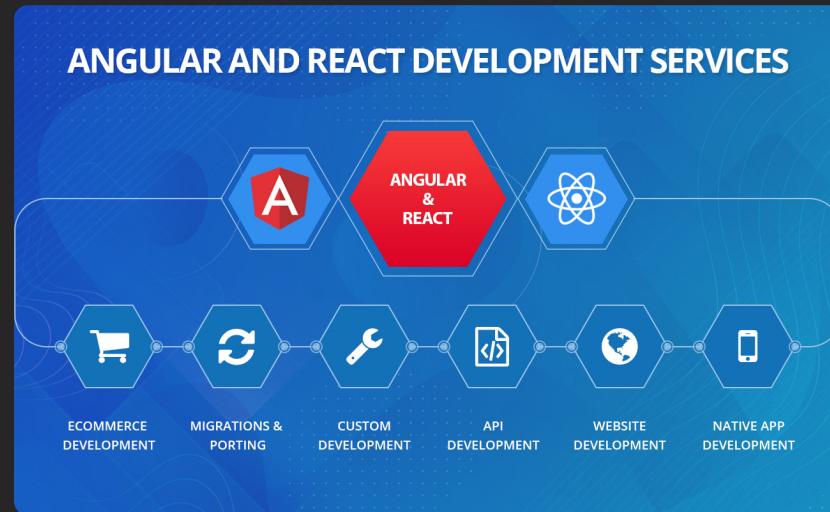
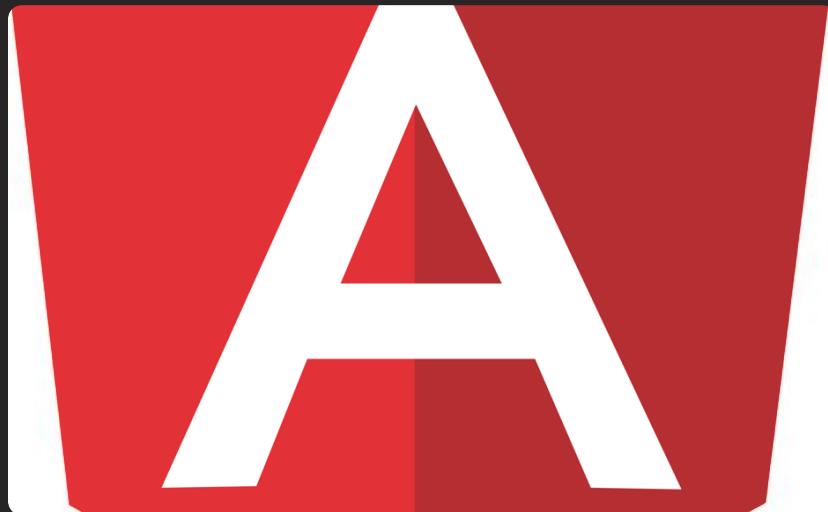
Le framework Angular est un puissant outil de développement web utilisé pour créer des applications interactives et dynamiques. Découvrez son architecture et ses principaux concepts.



by Haythem Hemaissi

Qu'est-ce qu'Angular ?

Angular est un framework open-source développé par Google pour la création d'applications web. Il utilise le langage TypeScript et offre une architecture basée sur les composants, les services et le routage.

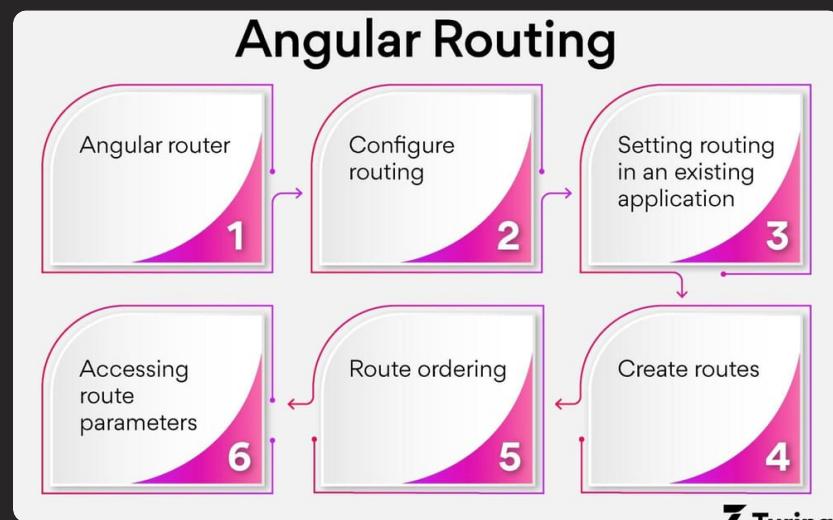


Composants

Les composants sont les blocs de construction fondamentaux d'une application Angular. Ils encapsulent la logique et l'interface utilisateur d'une partie spécifique de l'application.

Services et Injections de Dépendances

Les services sont utilisés pour partager des données et des fonctionnalités entre différents composants. L'injection de dépendances permet de créer des relations entre les composants et les services.



Routage dans Angular

Le routage permet de naviguer entre les différentes vues d'une application Angular. Il facilite la gestion de l'état de l'application et l'affichage dynamique du contenu.

Principaux Concepts d'Angular

1 Modularité

Angular encourage la modularité en divisant l'application en fonctionnalités distinctes appelées modules.

2 Binding de Données

Angular offre différents types de binding de données pour lier de manière dynamique les données du modèle à l'interface utilisateur.

3 Directives

Les directives sont des instructions qui modifient le comportement de certaines parties de l'interface utilisateur en fonction des données ou des actions de l'utilisateur.

4 Pipes

Les pipes permettent de formater et de transformer les données affichées dans l'interface utilisateur d'une manière simple et réutilisable.



Architecture du Framework Angular

- Angular se base sur une architecture de type MVC (Modèle-Vue-Contrôleur) pour organiser et structurer les applications.
- Le modèle représente les données et la logique métier de l'application.
- La vue affiche les données et interagit avec l'utilisateur.
- Le contrôleur gère les interactions entre la vue et le modèle.



Composants d'Angular

Composants Enfants

Les composants peuvent être imbriqués les uns dans les autres pour créer une structure hiérarchique et réutilisable.

1

Création des Composants

Les composants sont créés à partir d'une classe TypeScript avec une annotation `@Component` et un template HTML qui définit l'interface utilisateur.

2

Communication entre Composants

Angular offre différents mécanismes de communication entre composants, tels que les propriétés d'entrée et de sortie, les observables et les services.

3

Services et Injections de Dépendances

Création de Services

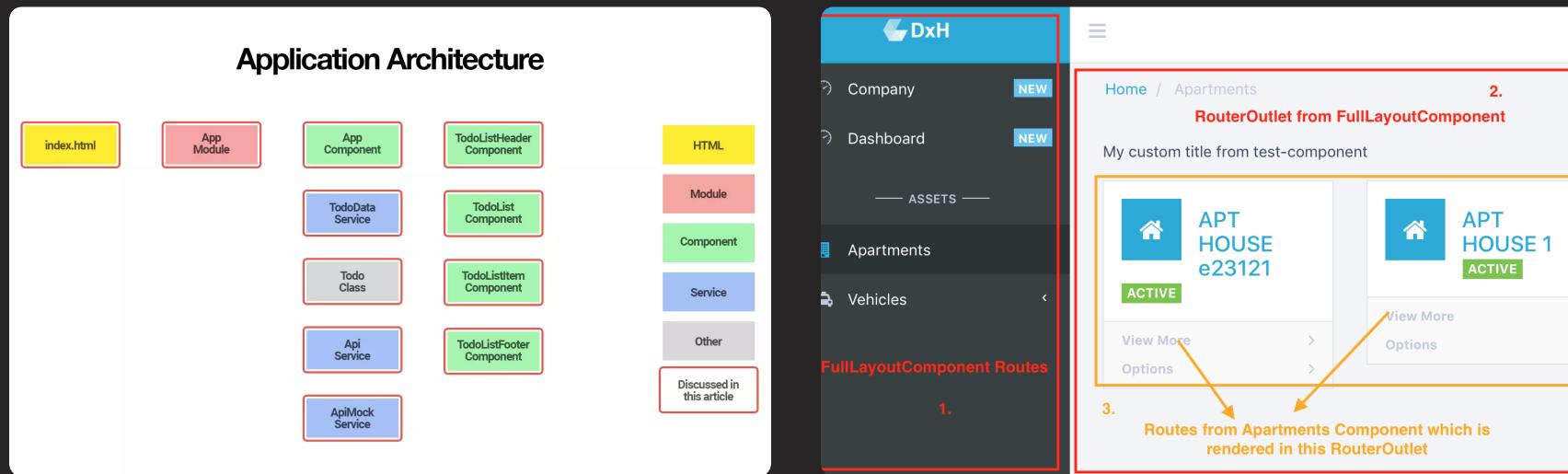
Les services sont créés à partir d'une classe TypeScript avec une annotation `@Injectable`. Ils peuvent être utilisés pour encapsuler la logique métier et partager des données entre les composants.

Injections de Dépendances

L'injection de dépendances permet de fournir une instance d'un service à un composant qui en demande une, facilitant ainsi la réutilisabilité et la testabilité des composants.

Routage dans Angular

Le routage dans Angular permet de naviguer entre les différentes vues de l'application. Il utilise une approche basée sur les URL pour définir les différentes routes et les associer à des composants spécifiques.



Configuration des Routes

La configuration des routes se fait dans un fichier de configuration qui associe une URL à un composant.

Outlet Router

L'élément d'insertion du routeur, `<router-outlet>`, est utilisé pour afficher les composants correspondant aux différentes routes.