

Angular Universal

Server Side Rendering (SSR)

Une application angulaire normale s'exécute dans le navigateur, rendant les pages dans le DOM en réponse aux actions de l'utilisateur.

Angular Universal s'exécute sur le serveur, générant des pages d'application statiques qui sont ensuite amorcées sur le client.

Cela permet aux visiteurs de visualiser la mise en page de l'application avant que celle-ci ne soit entièrement interactive

Ajouter le module server

```
ng add @nguniversal/express-engine
```

Le module Angular Universal s'appuie sur la librairie ExpressJS

Les modifications :

```
/server.ts
```

```
/tsconfig.server.json
```

```
/src/main.server.ts
```

```
/src/app/app.server.module.ts
```

Ajout de paramètres dans le fichier /angular.json.

Démarrage du serveur

```
npm run dev:ssr
```

Pourquoi utiliser le rendu côté serveur ?

Il existe trois raisons principales de créer une version universelle de votre application.

- Faciliter l'exploration des crawler (SEO)
- Améliorer les performances sur les appareils mobiles et à faible puissance
- Afficher rapidement la première page

Faciliter les robots d'indexation (SEO)

Les robots des moteurs de recherche ou réseaux sociaux (Google, Bing, Facebook, Twitter) peuvent être incapables de naviguer et d'indexer une application Angular hautement interactive.

Angular Universal génère une version statique de l'application pour la rendre facilement consultable et navigable sans JavaScript

Angular Universal propose également un aperçu du site car chaque URL renvoie une page entièrement rendue.

Améliorer les performances sur les appareils mobiles et à faible puissance

Certains appareils ne prennent pas en charge JavaScript ou exécutent JavaScript si mal que l'expérience utilisateur est inacceptable.

Dans ces cas, vous pouvez avoir besoin d'une version de l'application rendue par le serveur et sans JavaScript.

Cette version, bien que limitée, pourrait être la seule alternative pratique pour les personnes qui autrement ne pourraient pas du tout utiliser l'application.

Afficher la première page rapidement

L'affichage rapide de la première page peut être essentiel pour l'engagement des utilisateurs.

Les pages qui se chargent plus rapidement ont un meilleur taux d'engagement client

Votre application devra peut-être se lancer plus rapidement pour engager ces utilisateurs avant qu'ils ne décident de faire autre chose.

Avec Angular Universal, vous pouvez générer des pages de destination pour l'application qui ressemblent à l'application complète.

Les pages sont en HTML pur et peuvent s'afficher même si JavaScript est désactivé. Les pages ne gèrent pas les événements du navigateur, mais elles prennent en charge la navigation sur le site à l'aide de [routerLink](#).

En pratique, vous servirez une version statique de la page de destination pour retenir l'attention de l'utilisateur. En même temps, vous chargerez l'application Angular complète derrière. L'utilisateur perçoit des performances quasi instantanées à partir de la page de destination et bénéficie d'une expérience interactive complète après le chargement complet de l'application.

Commandes Utiles

`npm run dev:ssr`

Semblable à [ng serve](#), qui offre un rechargement en direct pendant le développement, mais utilise le rendu côté serveur. L'application s'exécute en mode veille et actualise le navigateur après chaque modification. Cette commande est plus lente que la `ng serve` commande réelle.

`ng build && ng
run
app-name:server`

Génère à la fois le script du serveur et l'application en mode production. Utilisez cette commande lorsque vous souhaitez générer le projet pour le déploiement.

npm run
serve:ssr

Démarre le script serveur pour servir l'application localement avec un rendu côté serveur. Il utilise les artefacts de construction créés par `ng run build:ssr`, alors assurez-vous d'avoir également exécuté cette commande.

REMARQUE :

`serve:ssr` n'est pas destiné à être utilisé pour servir votre application en production, mais uniquement pour tester localement l'application rendue côté serveur.

npm run
prerender

Utilisé pour pré-afficher les pages d'une application. En savoir plus sur le pré-rendu [ici](#).