

Build et déploiement Web

Vue d'ensemble

Le frontend Angular et le backend NestJS sont déployés ensemble. Le build Angular de production écrit directement dans le backend, puis le backend est compilé et démarré avec l'environnement voulu.

Pipeline de build

flowchart TD

A[uniksimFrontEnd] -->|npm run build| B[uniksim-back-end/src/front]

B -->|npm run build dans le backend| C[uniksim-back-end/dist]

C -->|npm run start:demo| D[Node + demo.env]

D --> E[sim.dev.uniksim.com]

Pourquoi le frontend apparaît dans le backend

Dans `uniksimFrontEnd/angular.json`, la configuration de production utilise :

```
outputPath: ../uniksim-back-end/src/front
```

Le dossier `src/front` contient donc le **résultat compilé d'Angular**. Ce n'est pas un dossier frontend à modifier manuellement.

Lors du build NestJS, ces fichiers sont ensuite disponibles dans le build backend, et `ServeStaticModule` les sert au navigateur.

Serveur de développement hébergé

L'instance de développement est accessible via `sim.dev.uniksim.com` sur l'infrastructure DigitalOcean.

Le workflow serveur utilisé actuellement est :

1. Se connecter au serveur DigitalOcean.
2. Passer sur l'utilisateur `forge` avec `sudo -u forge` ou la commande équivalente utilisée sur le serveur.
3. Aller dans le dossier du site avec `cd sim.dev.uniksim.com`.
4. Mettre le code à jour.
5. Builder le frontend Angular avec `npm run build` dans `uniksimFrontEnd`.
6. Builder le backend NestJS avec `npm run build` dans `uniksim-back-end`.
7. Démarrer le backend avec `npm run start:demo`.

Ordre de build

sequenceDiagram

```
participant D as Développeur
participant F as Angular
participant B as Backend NestJS
participant N as Node demo
participant W as sim.dev.uniksim.com
```

```
D->>F: npm run build
F-->>B: Génère src/front
D->>B: npm run build
B-->>N: Génère dist
D->>N: npm run start:demo
N->>N: Charge demo.env
N-->>W: Sert API + frontend compilé
```

Scripts backend disponibles

- `npm run build` : compile NestJS.
- `npm run start:dev` : démarre `dist/main` avec `dev.env`.
- `npm run start:dev2` : démarre avec `dev2.env`.
- `npm run start:demo` : démarre avec `demo.env`.
- `npm run start:live` : démarre avec `live.env`.

Points d'attention

- Le frontend doit être buildé **avant** le backend si on veut embarquer la nouvelle version Angular dans le build backend.
- Après une modification uniquement backend, le build Angular n'est pas toujours nécessaire.
- Après une modification frontend, vérifier que les fichiers générés sont présents dans `uniksim-back-end/src/front` avant le build backend.
- Vérifier l'environnement utilisé avant de lancer le serveur : `demo`, `dev`, `dev2` ou `live`.

Revision #1

Created 2026-09-18 17:42:53 UTC by Mathis

Updated 2026-09-18 17:42:56 UTC by Mathis