

Cadriciel Web côté client

Cours 8 - Routage

Steve Lévesque, Tous droits réservés © où applicables

Table des matières

- 1 Le principe de routage par répertoires
 - Définition - Structure
 - Définition - URL
 - Routage par répertoire
 - Convention de fichiers
 - Hiérarchie des Components (Précédence/Priorité)
 - Colocation de fichiers auxiliaires

Définition - Structure

Arbre (Tree) : Une convention pour visualiser une structure hiérarchique. Par exemple, une arborescence de composants avec des composants parents et enfants, une structure de dossiers, etc.

Sous-arbre (Subtree) : Partie d'un arbre, commençant à une nouvelle racine (premier) et se terminant aux feuilles (dernier).

Racine (Root) : premier nœud d'une arborescence ou d'une sous-arborescence, telle qu'une disposition racine.

Feuille (Leaf) : nœuds d'une sous-arborescence qui n'ont pas d'enfants, tels que le dernier segment d'un chemin d'accès URL.

Définition - Structure

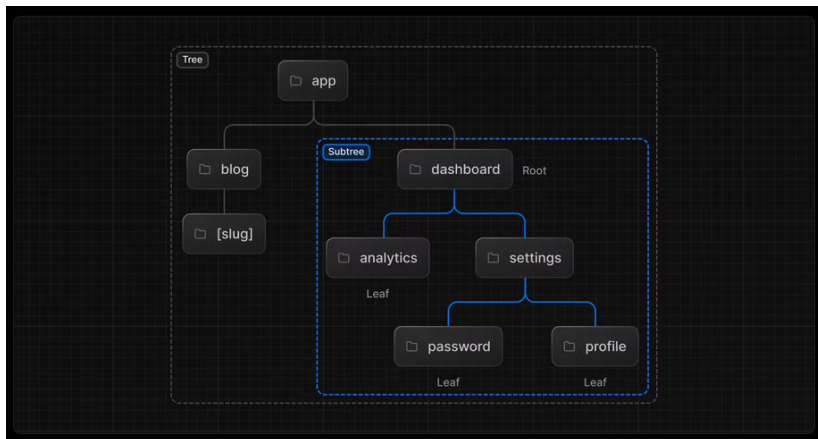


Figure: <https://nextjs.org/docs/app/building-your-application/routing>

Définition - URL

Segment d'URL (URL Segment) : partie du chemin d'url délimitée par des barres obliques.

Chemin d'accès à l'URL (URL Path) : partie de l'URL qui vient après le domaine (composé de segments).

Définition - URL

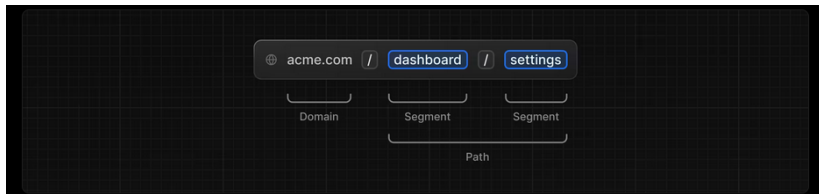


Figure: <https://nextjs.org/docs/app/building-your-application/routing>

Routage par répertoire

Les dossiers sont utilisés pour définir des itinéraires. Un itinéraire est un chemin unique de dossiers imbriqués, suivant la hiérarchie du système de fichiers du dossier racine jusqu'à un dossier feuille final qui inclut un fichier `page.js/jsx/tsx`.

Des fichiers sont utilisés pour créer l'interface utilisateur qui est affichée pour un segment de route.

Sur ce, chaque dossier représente un segment de route, pouvant être imbriqué l'un dans l'autre.

Routage par répertoire

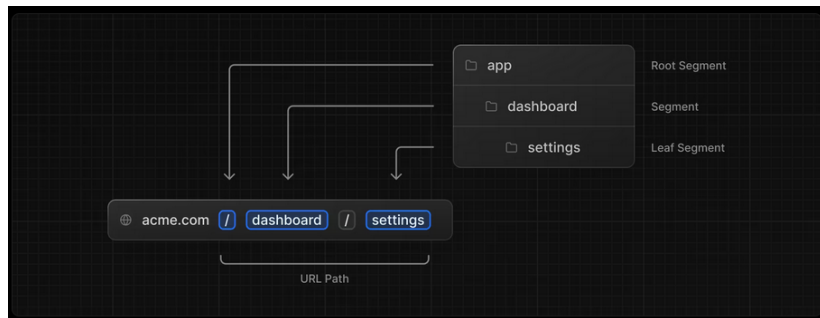


Figure: <https://nextjs.org/docs/app/building-your-application/routing>

Convention de fichiers

Next.js fournit un ensemble de fichiers spéciaux pour créer une interface utilisateur avec un comportement spécifique dans les itinéraires imbriqués.

Voir plus d'information : <https://nextjs.org/docs/app/building-your-application/routing#file-conventions>

NB : Comme `page.js/jsx/tsx`, les extensions de fichier `.js`, `.jsx` ou `.tsx` peuvent être utilisées pour les fichiers spéciaux.

Convention de fichiers

- **layout** : Interface utilisateur partagée pour un segment et ses enfants
- **page** : Interface utilisateur unique d'un itinéraire et rendre les itinéraires accessibles au public
- **loading** : Chargement de l'interface utilisateur pour un segment et ses enfants
- **not-found** : Interface utilisateur introuvable pour un segment et ses enfants
- **error** : Interface utilisateur d'erreur pour un segment et ses enfants
- **global-error** : Interface utilisateur d'erreur globale
- **route** : Point de terminaison d'API côté serveur
- **template** : Specialized re-rendered Layout UI
- **default** : Interface utilisateur de secours pour les itinéraires parallèles

Hiérarchie des Components (Précédence/Priorité)

Les composants React définis dans des fichiers spéciaux d'un segment d'itinéraire sont rendus dans une hiérarchie spécifique.

NB : une erreur est fatale et est considérée dans la frontière des erreurs respectivement, loading est avant not-found dans le cas qu'un chargement survient peu importe le bon ou mauvais chemin URL, puis not-found est dans la frontière des erreurs puisque ce n'est pas un cas de situation complètement intègre (404), mais considéré comme un scénario contrôlé dans les utilisations d'interfaces.

- layout.js
- template.js
- error.js (React error boundary)
- loading.js (React suspense boundary)
- not-found.js (React error boundary)
- page.js ou imbrication layout.js

Hiérarchie des Components (Précédence/Priorité)

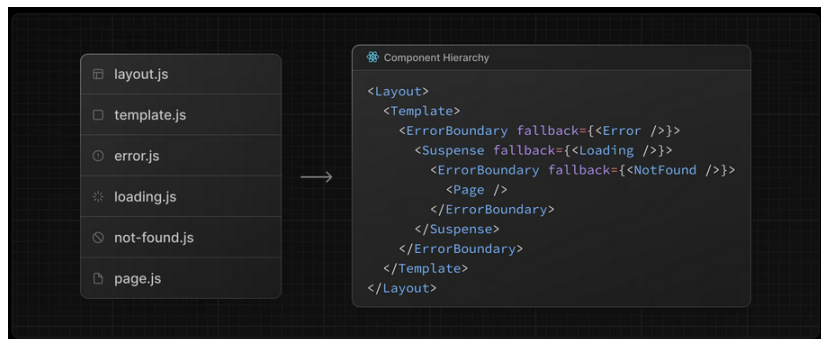


Figure: <https://nextjs.org/docs/app/building-your-application/routing>

Hiérarchie des Components (Précédence/Priorité)

Dans une route imbriquée, les Components d'un segment seront imbriqués dans les composants de son segment parent.

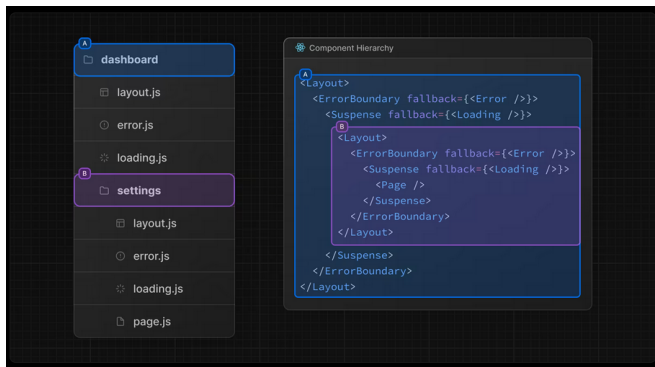


Figure: <https://nextjs.org/docs/app/building-your-application/routing>

Colocation de fichiers auxiliaires

Sachant que seulement les fichiers de `page.js` ou `route.js` sont disponibles au grand public, il est possible de mettre d'autres fichiers en colocation sans danger d'accès direct par les utilisateurs finaux.

Colocation de fichiers auxiliaires

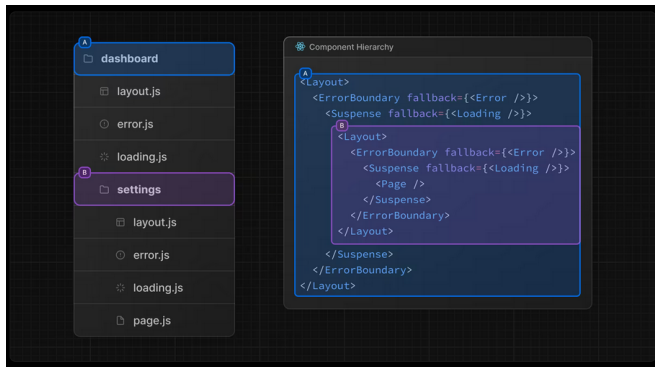


Figure: <https://nextjs.org/docs/app/building-your-application/routing>

Bibliographie

- `https://nextjs.org/docs/app/building-your-application/routing`