Frontend and Backend Folder Structure Guide

# Frontend Structure

This structure is intended for React/Next.js projects.

src/  
├── api/ # Axios or fetch-based API calls (e.g., documentTypeAPI.ts)  
├── components/ # Reusable UI components  
│ └── document-type/ # Specific folder for DocumentType components  
│ ├── DocumentTypeForm.tsx  
│ └── DocumentTypeTable.tsx  
├── constants/ # Static configuration and constant values  
├── hooks/ # Custom hooks  
│ └── use-document-type/  
│ ├── formDefaults.ts  
│ ├── index.ts  
│ └── types.ts  
├── pages/ or app/ # Routing or page-based structure (depends on framework)  
├── store/ # Redux store slices  
│ └── slices/  
│ └── documentTypeSlice.ts  
├── styles/ # CSS/SCSS or Tailwind configs  
└── utils/ # Utility functions and helpers

# Backend Structure

This structure is intended for Node.js + Express projects with Sequelize or TypeORM.

src/  
├── api/ # API contracts (optional for clarity)  
├── config/ # Configuration files (e.g., db config)  
├── controllers/ # Handles business logic  
│ └── documentType.controller.ts  
├── middlewares/ # Custom middleware (auth, error handling)  
├── models/ # Sequelize/TypeORM models  
│ └── documentType.model.ts  
├── repositories/ # Database interaction logic  
│ └── documentType.repository.ts  
├── routes/ # Express routers  
│ └── documentType.route.ts  
├── services/ # Service layer (core logic)  
│ └── documentType.service.ts  
├── utils/ # Utility functions (e.g., validators)  
└── index.ts # Entry point of the application