Developer Documentation for Century Cinema Ticket backend
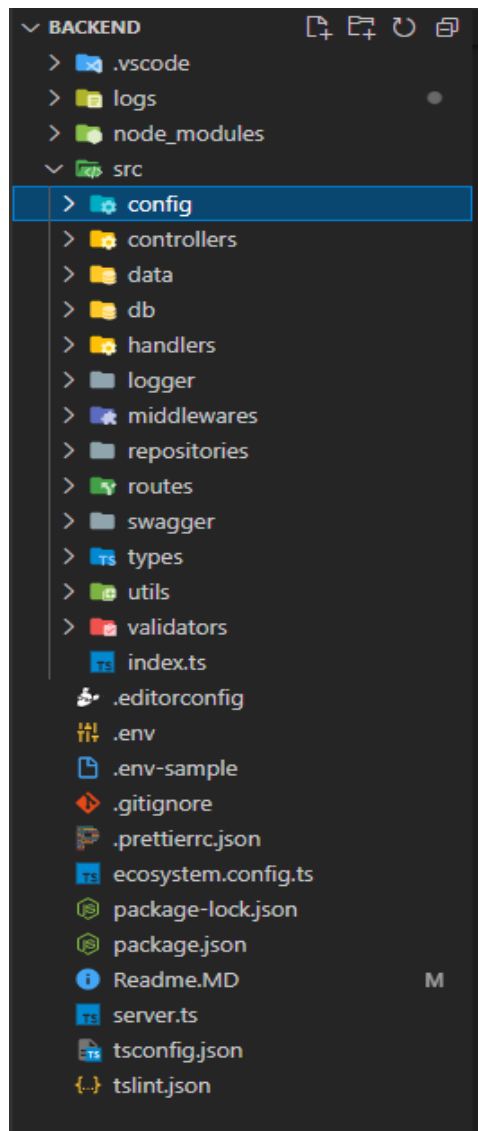
This developer documentation contains two sections, One for the Backend API and one for the Admin dashboard.

1. Backend API

a) Tech stack
- Typescript
- Express.JS
- Winston
- Prisma
- Swagger-ui-express
- postgresql

b) Folder structure

The application code is contained in the src directory
config - contains configurations imported from the environment variables
controllers - contains the api controllers
data - contains application constants exported from modules for better code readability
db - contains the schema for the database as a .prisma file
handlers - contains the error handler for the application
logger - contains the logger service of the application (uses winston)
middleware - contains the api middlewares like authentication & authorization middlewares
repositories - contains the code repositories for most common database operations
routes - contains the api routes
swagger - contains the configuration for swagger-js-doc
types - contains the typescript Interfaces/Types that are used in the system
utils - contains the various utilities used for example like messaging, authentication, etc…
validators - contains the request validators for the api routes
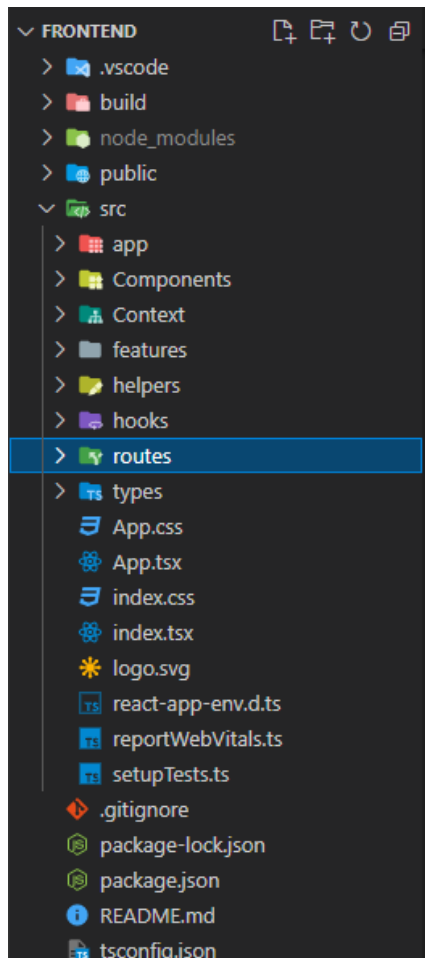
C) How to run the project locally
  - **npm i** (installs the required modules)
  - **npx prisma migrate deploy** (apply the pending migrations to your database)
  - **npx prisma generate client** (generate the models ORM code for type checking)
  - Then preferably restart your IDE
  - **npm run dev** (run the application in development mode)
  - **now you can go to http://localhost:3000/api-docs/ to check the api docs**

2. Admin Dashboard
a) Tech stack
  - Typescript
  - React.JS
  - Redux

b) Folder structure

The application code is contained in the src directory

app - contains the redux toolkit configuration files
Components - contains the react components
Context - contains the configuration for react context api
features - contains the slices for our redux toolkit
helpers - contains helper functions like http calls to backend api
hooks - contains the custom hooks contained in the application
routes - contains the routes needed for react-router-dom
types - contains the typescript interfaces/types used in the application