TABLE OF CONTENTS

► Design-Documentation.md

Design-Documentation.md

Table of Contents

Technical Design Overview

Software Architecture

Frontend

Backend

Database

API Specification

Detailed List of Technology

Technical Design Overview

Our industry partner specified requirements for the rough technology stack. This included React for the frontend and .Net with C# in the backend.

Based on the requirements, we decided to go for **next.js** as a React framework, as it provides an efficient environment for developing React applications. It provides an intuitive API for quickly creating production-ready React applications, with features like server-side rendering, static optimization, and automatic routing, which were very helpful in the course of the project.

For the authentication we used **Next-auth**, which is a popular and powerful authentication library for React applications. It provides features such as built-in support for multiple authentication providers, secure session management, and an intuitive and extensible API that makes it easy to customize authentication for a project.

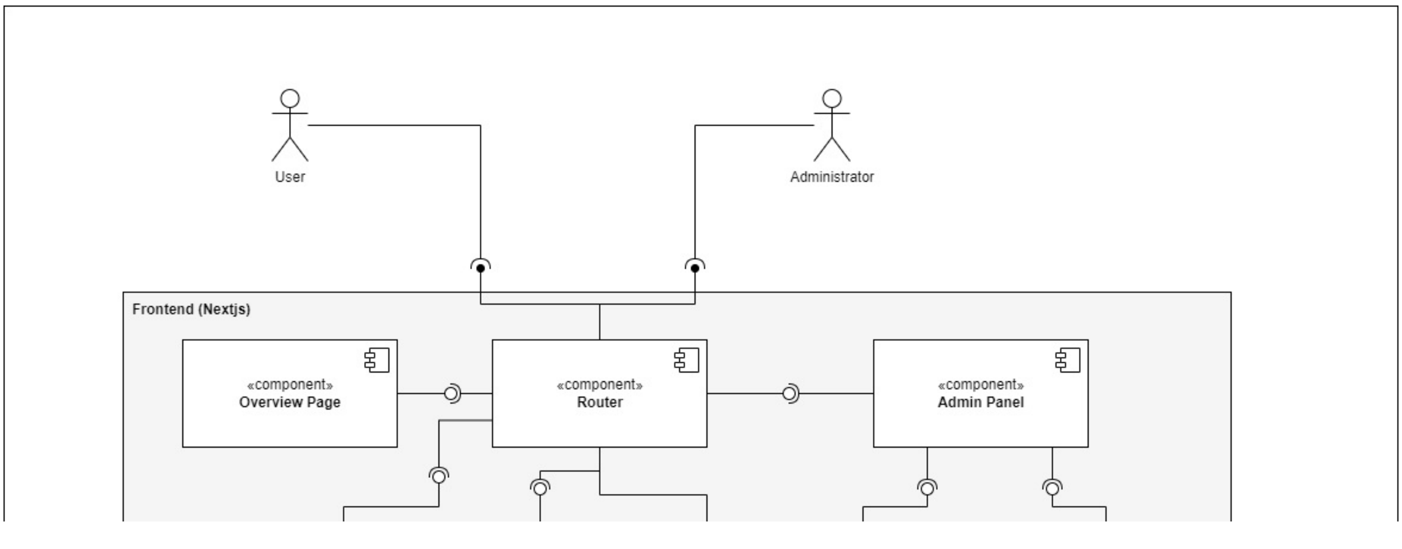
For designing our application, we used **DaisyUI** on top of **Tailwind CSS**. Tailwind CSS is a utility-first CSS framework that provides low-level styling classes that make it easy to create custom designs quickly. Daisy UI is a highly-customizable UI library built on top of Tailwind CSS, providing a set of ready-made components that can be quickly adapted to fit the design of our application. Using these two together allowed for rapid styling of the Deskstar Frontend, while still providing a great level of customization.

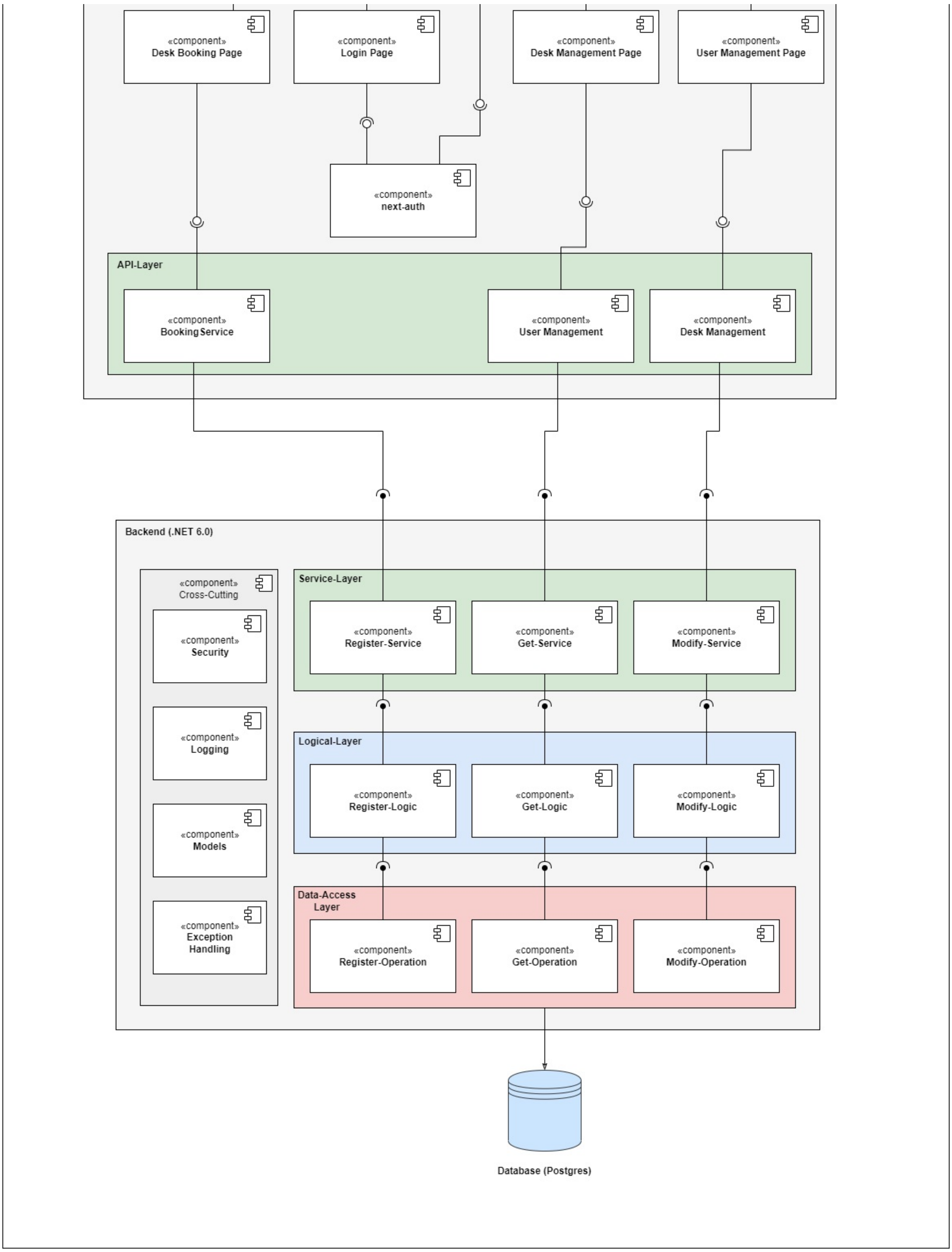
PostgresSQL was chosen for the database, as it is a powerful and reliable relational database.

To ensure the application ran smoothly on different machines, we set up a **devcontainer**. Creating a devcontainer for a project allowed us to ensure consistent development across different machines. A devcontainer provides a development environment that is pre-configured and tailored to the specific needs of our project, and it also allows for easy setup and configuration of the development environment. Additionally, a devcontainer can be used to package the application and its dependencies into a single package, which can be easily deployed to our production server. This helps to ensure that the application will run consistently and reliably on different machines, regardless of their configuration.

Software architecture

The following image provides an overview of our runtime and code components.




«component» Desk Booking Page
«component» Login Page
«component» Desk Management Page
«component» User Management Page
«component» next-auth
API-Layer
«component» Booking Service
«component» User Management
«component» Desk Management
Backend (.NET 6.0)
«component» Cross-Cutting
«component» Security
«component» Logging
«component» Models
«component» Exception Handling
Service-Layer
«component» Register-Service
«component» Get-Service
«component» Modify-Service
Logical-Layer
«component» Register-Logic
«component» Get-Logic
«component» Modify-Logic
Data-Access Layer
«component» Register-Operation
«component» Get-Operation
«component» Modify-Operation
Database (Postgres)

Frontend

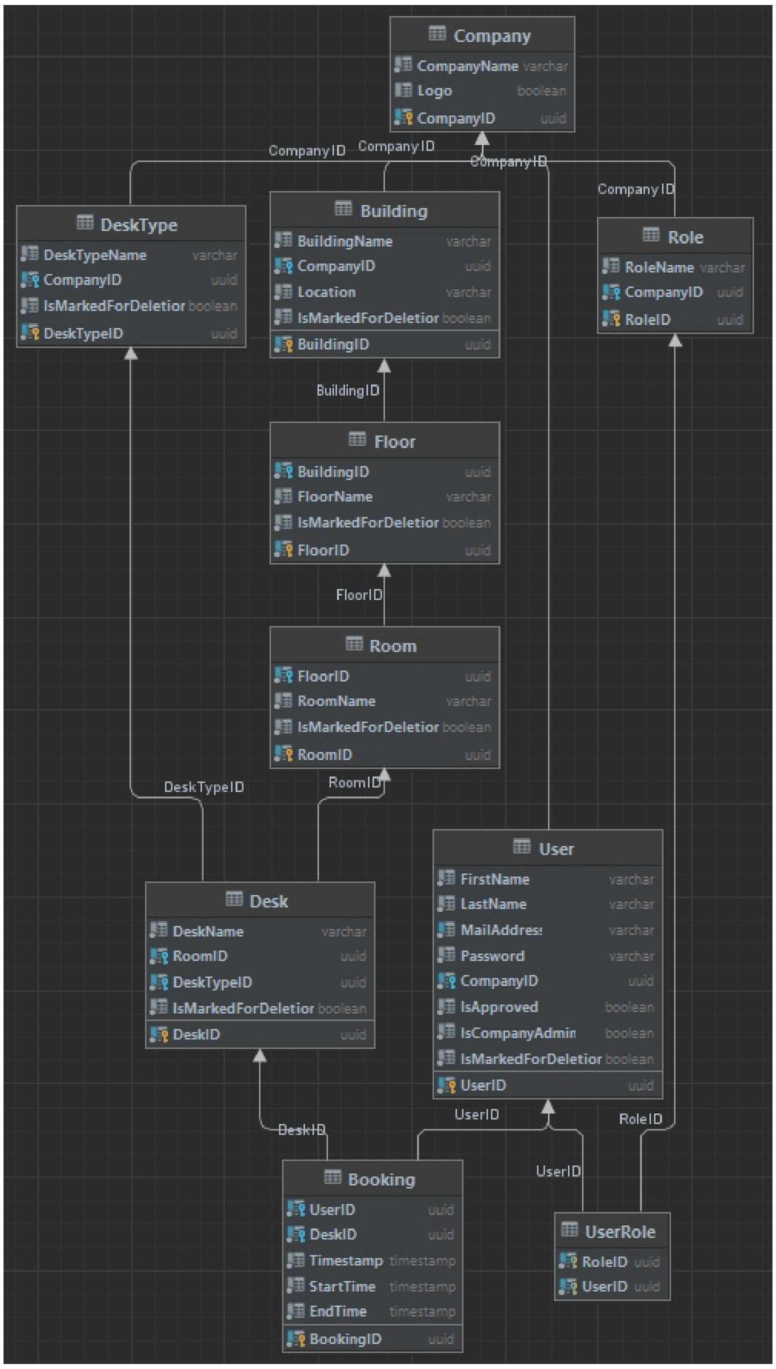
The user interface of our applications consists of multiple pages, all following a design theme we established to ensure consistency. We developed many custom React components that we could reuse on several of our pages to ensure efficient, yet custom development. The frontend communicates with the backend through several API services we defined, which include response handling to ensure the user is always shown what they need to know.

Backend

The backend consists of the Service Layer, where we defined all controllers, the Logical Layer with all UseCases and the DataAccess Layer, which handles all interactions with our Postgres database.

Database

The following image shows our Data Model.



API Specification

Swagger Documentation (https://amosproj.github.io/amos2022ws05-shared-desk-mgmt/)

Detailed List of Technology

Bill of materials (https://docs.google.com/spreadsheets/d/1LUbmaLufCelq6UPbuvRlQrnZx9TGKXqAJfG-utjM0dc/edit#gid=927854276)