

SENG 513 Docker Deployment Instructions

1. Introduction

This document provides the necessary steps to set up and run the project using Docker Compose. The project includes three services: a React frontend, a Node.js backend, and a PostgreSQL database. This guide also outlines the strategy for managing sensitive information like database credentials.

2. Setup Instructions

Step 1: Clone the Repository

Run the following commands to clone the repository and navigate to the project directory:

```
git clone <repo-url>
```

```
cd project-directory
```

Step 2: Configure Environment Variables

Create a .env file at the project root with the following environment variables. Refer to the provided .env.example for guidance:

```
POSTGRES_DB=postgres
```

```
POSTGRES_USER=postgres.dfqtwodpxzjylkwphxji
```

```
POSTGRES_PASSWORD=74K3_Y0Ur_71M3
```

```
SUPABASE_URL=https://dfqtwodpxzjylkwphxji.supabase.co
```

```
SUPABASE_KEY=your_supabase_key
```

```
SUPABASE_AUTH_KEY=your_supabase_auth_key
```

```
DATABASE_URL=postgresql://postgres.dfqtwodpxzjylkwphxji:74K3_Y0Ur_71M3@aws-0-us-east-1.
```

pooler.supabase.com:5432/postgres

REACT_APP_SUPABASE_URL=https://dfqtwodpxzjylkwphxji.supabase.co

REACT_APP_SUPABASE_KEY=your_supabase_key

Step 3: Build and Run the Docker Containers

Run Docker Compose to build and start the containers:

docker-compose up --build

Step 4: Access the Application

Once the containers are running, you can access the services through your browser:

- Frontend: <http://localhost:3000>
- Backend: <http://localhost:4000>

3. Password Strategy

Sensitive information such as database passwords and Supabase keys are stored in environment variables. These variables are defined in a `.env` file that is excluded from version control via the `.gitignore` file. For developers setting up the project, an `.env.example` file is provided to guide them in creating their own `.env` file with the appropriate credentials.