Building a User-Friendly Frontend for a RESTful API with CRUD Operations

Deadline 14-5-2023, 23:45

Problem Statement

You have a RESTful API (from Assignment 1) that performs CRUD operations on a `person` object, which has the following attributes: `name`, `age`, `id`, `gender`, and `email`. You would like to add a frontend to this API that presents the data to users in a user-friendly way. Your task is to create a frontend for your RESTful API that meets the following requirements:

- Display a list of all people in the database

- Allow users to add a new person to the database

- Allow users to update an existing person in the database

- Allow users to delete a person from the database

Frontend Requirements

The main frontend app requirements can be enumerated as follows:

1. Choose a suitable frontend technology (*e.g.* React, Vue.js, Angular, traditional CSS/JS/HTML, etc.) and prepare a virtual environment or container.

2. Create a frontend app that interacts with your RESTful API to retrieve and display data.

3. Write code in your frontend app that allows users to add, update, and delete people in the database using your API.

4. Create a Dockerfile that describes how to build a Docker image for your frontend app.

5. Build a Docker image for your frontend app and run it as a container.

Docker Image Specifications

You should write a Dockerfile that builds a Docker image containing the frontend code and its dependencies. You should then run a container from this Docker image to verify that the frontend app is working correctly. Your submission should include the following:

- The Dockerfile used to build the Docker image.

- You entire project as a git repository.

- A YAML file that specifies the configuration for the Docker container, the RESTful API container, including any environment variables, port mappings, or other settings.

The Dockerfile should include the necessary instructions to install any required dependencies, copy the frontend app code to the container, build the app if necessary, and start the app inside the container. The Docker image should be published to Docker Hub or another container registry to make it accessible to others.

The YAML file should specify the configuration for the Docker container, including any environment variables, port mappings, or other settings. This file should be included in your submission and should be well-documented to explain the purpose and usage of each configuration setting.

Grading Criteria

Your submission will be graded based on the following criteria:

- The correctness of the Frontend implementation.

- The correctness of the Dockerfile and the resulting Docker image.

- The correctness and completeness of the YML file.

- The clarity and completeness of the documentation.