

RadDiffChecker Documentation

This documentation provides an in-depth overview of the RadDiffChecker project, encompassing both the front-end (built with Next.js) and the back-end (developed using Flask). The project aims to assist radiologists in comparing reports and templates efficiently.

Table of Contents

1. Project Overview
2. Frontend (Next.JS)
 - Description
 - Usage
3. Backend (Flask)
 - Description
 - API Documentation
4. Setup Instructions
 - Frontend Setup
 - Backend Setup
5. Usage

Project Overview

The RadDiffChecker project is designed to assist radiologists in comparing a report with a template. It consists of two main components:

1. **Frontend (Next.JS):** A web application that allows users to input a Template ID and a Report. The application fetches the associated template text from the backend and visualizes the differences between the Report and the Template using the **react-diff-viewer** component.
2. **Backend (Flask):** An API built with Flask, serving as the data source for the front-end. It retrieves template text based on a provided Template ID and returns it as JSON.

Frontend (Next.JS)

Description

The front-end is developed using Next.js, providing an interactive user interface for the RadDiffChecker project. It enables users to input a Template ID, fetch the template text from the back-end, and visualize the differences between the Report and the Template.

Usage

1. Enter a Template ID in the designated input field.
2. Click the "Find Diff" button to retrieve the template text from the backend.
3. Enter a Report in the textarea.
4. View the differences between the Report and the Template text using the **react-diff-viewer** component.

Backend (Flask)

Description

The back-end is constructed with Flask and serves as the data provider for the RadDiffChecker project. It offers a simple API that retrieves template text based on a given Template ID, which is later visualized in the front-end.

API Documentation

Get Template Text

- **Endpoint:** `/api/template/:template_id`
- **Method:** GET
- **Parameters:**
- ``<template_id>`` (integer) - The Template ID for which the template text is requested.

Response:

- If the template is found:
 - Status: 200 OK
 - Body: JSON object with ``template_text``.
- If the template is not found:
 - Status: 404 Not Found
 - Body: JSON object with an ``error`` message.

Setup Instruction

Frontend Setup

1. Open the frontend repository from RadDiffChecker folder.
2. In the project directory, run:
npm install --save --legacy-peer-deps
npm install react-diff-viewer --legacy-peer-deps
3. Start the frontend application:
npm run dev
4. Access the application at <http://localhost:3000> in your web browser.

Note: legacy-peer-deps command is used to prevent conflicts of dependencies version

Backend Setup

1. Open the backend repository from RadDiffChecker folder in Terminal (in windows 'CMD' and in Mac/Linux OS 'Bash terminal').
2. Create a virtual environment and activate it (Linux/macOS):
python3 -m venv venv
source venv/bin/activate
On Windows, use the following:
python -m venv venv
.\venv\Scripts\activate
3. Install Flask:
pip install Flask flask-cors
4. Change the directory to:
cd flask-app
5. Start the backend server:
flask run --port 8080

The backend server will run on <http://localhost:8080>.

Usage

1. Ensure both the front-end and back-end servers are running.
2. Access the front-end application in your web browser at **`http://localhost:3000`**.
3. Enter a Template ID and click the "Find Diff" button.
4. Enter a Report in the textarea.
5. View the differences between the Report and the Template text.