**Project Documentation**

**Caution : main I’m using my local database server so change configuration of database and then test the application I got good result or else not work api’s**

**Overview**

This project is an employee management application built using Django. It allows for CRUD (Create, Read, Update, Delete) operations on employee records with AJAX interactions for a seamless user experience.

**Features**

* **Create Employee**: Add new employees with details including personal information, work experience, qualifications, and projects.
* **Read Employee**: View employee details in a list format.
* **Update Employee**: Edit employee information.
* **Delete Employee**: Remove employee records with a confirmation prompt.

**Models**

* **UserProfile**: Stores employee personal details (e.g., name, email, address).
* **WorkExperience**: Captures work history related to each employee.
* **Qualification**: Records educational qualifications of employees.
* **Project**: Details about projects associated with employees.

**Views**

* **EmployeeListView**: Displays a list of employees.
* **EmployeeDetailView**: Shows detailed information for a specific employee.
* **EmployeeUpdateView**: Allows editing of employee details.
* **EmployeeDeleteView**: Handles deletion of employee records.

**URLs**

* **List Employees**: path('employees/', EmployeeListView.as\_view(), name='employee\_list')
* **Employee Detail**: path('employee/<int:pk>/', EmployeeDetailView.as\_view(), name='employee\_detail')
* **Update Employee**: path('employee/update/<int:pk>/', EmployeeUpdateView.as\_view(), name='employee\_update')
* **Delete Employee**: path('employee/delete/<int:pk>/', EmployeeDeleteView.as\_view(), name='employee\_delete')

**AJAX Integration**

* **Delete Employee via AJAX**:
  + URL: /employee/delete/<employee\_id>/
  + Method: DELETE
  + Data: None
  + Response: JSON message indicating success or failure.

**CSRF Protection**

* **CSRF Token Handling**: The CSRF token is included in AJAX requests for security. The token is retrieved from cookies using JavaScript.

**Installation**

1. **Clone the Repository**:

bash

Copy code

git clone <repository-url>

1. **Install Dependencies**:

bash

Copy code

pip install -r requirements.txt

1. **Apply Migrations**:

bash

Copy code

python manage.py migrate

1. **Run the Development Server**:

bash

Copy code

python manage.py runserver

1. **Access the Application**:
   * Open a web browser and navigate to http://127.0.0.1:8000/ to use the application.

**Contributing**

* Fork the repository and create a feature branch.
* Submit a pull request with a description of your changes.

**License**

* This project is licensed under the MIT License. See the LICENSE file for details.

Feel free to modify or expand this documentation based on the specific details of your project and any additional features you may have.