End User Documentation for Department Management System

1. Overview

The Department Management System is a Django-based web application designed to help users manage departmental information efficiently. It enables users to perform the following actions:
- View a list of active departments
- Add a new department
- Edit an existing department's details
- Delete (soft delete) a department

This document provides detailed instructions on how to use the system effectively.
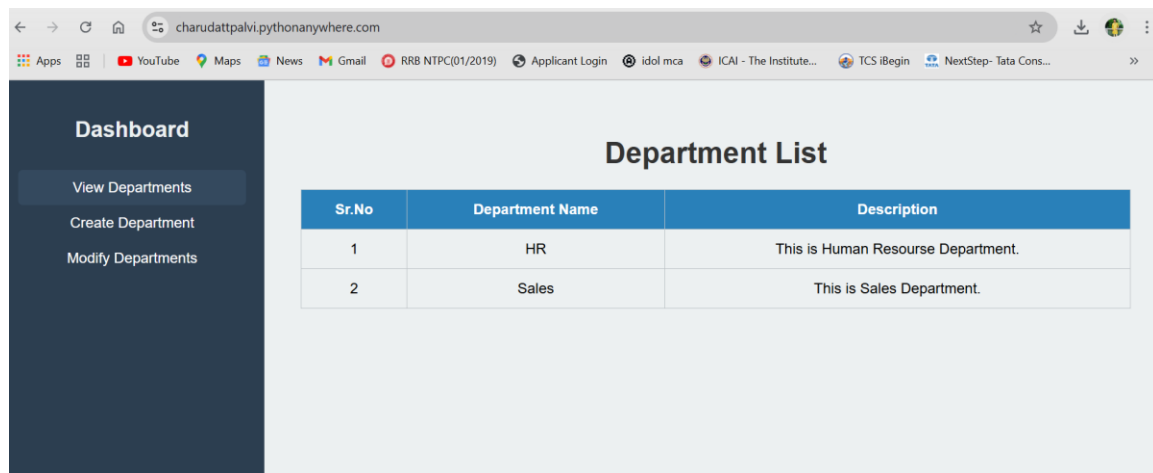
2. Accessing the System

The system is hosted locally or on a server. To access it, follow these steps:
1. Open a web browser (Chrome, Firefox, Edge, etc.).
2. Enter the following URL in the address bar:

  `https://charudattpalvi.pythonanywhere.com/` (for live server)
  `http://127.0.0.1:8000/` (for local server)

This will open the homepage of the Department Management System.

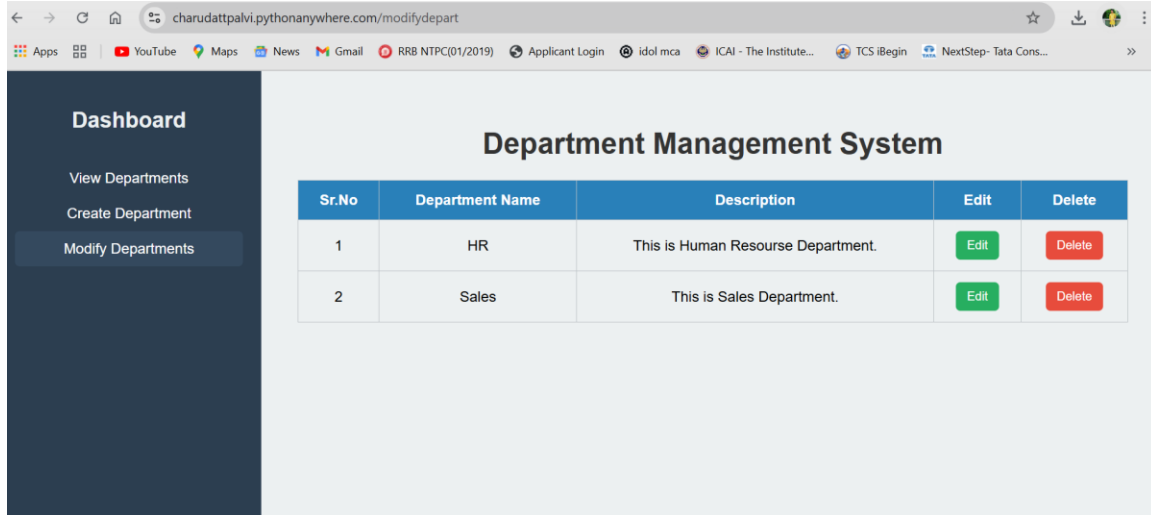## 3. Features and Instructions of Department:-

3.1 View Departments

The homepage displays a list of all active departments in a tabular format. To view departments:
1. Go to the homepage (`https://charudattpalvi.pythonanywhere.com/` or `http://127.0.0.1:8000/`).

2. The table will display the following details:
   - Department ID
   - Department Name
   - Description
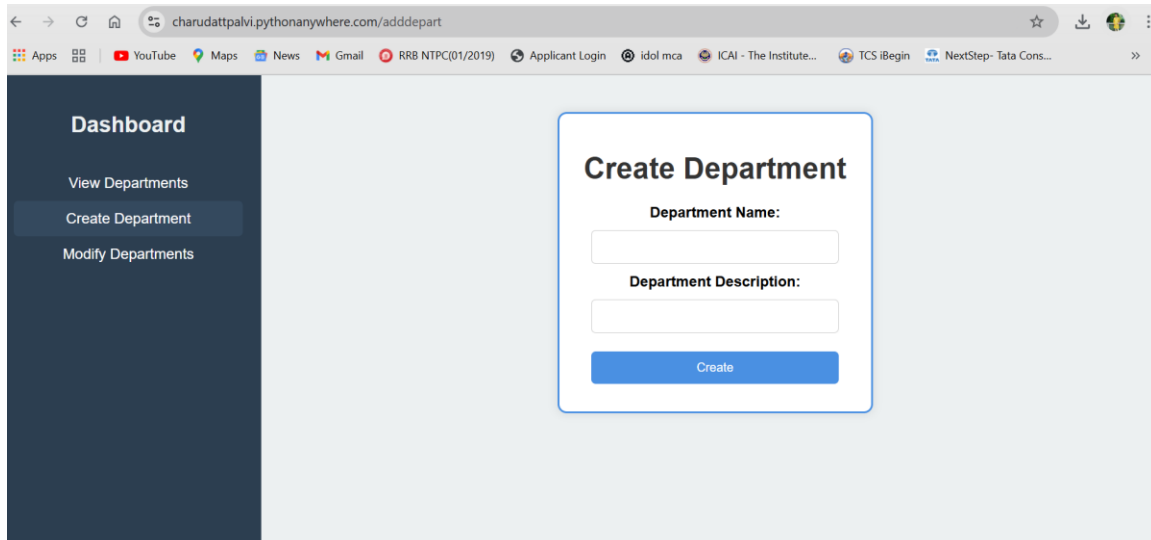3. Options to Edit or Delete a department are also displayed.

3.2 Add a New Department

To create a new department:
1. Navigate to the 'Create Department' page by clicking on **'Create Department'** in the sidebar.
2. Fill in the form fields:
   - **Department Name**: Enter the name of the department.
   - **Department Description**: Enter a short description.
3. Click on the **Create** button.
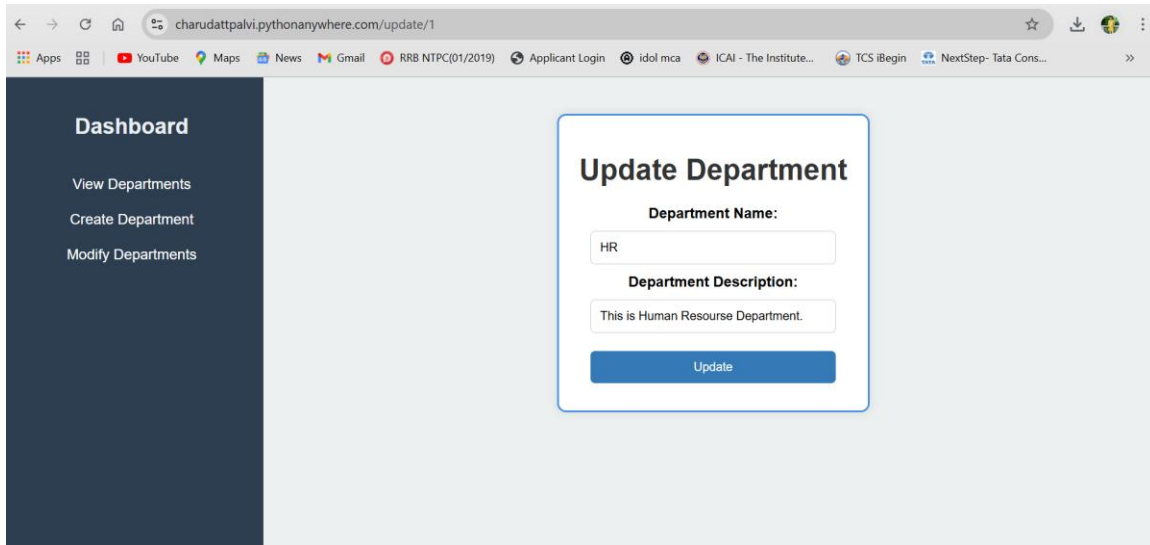4. The new department will now appear in the list.

3.3 Edit/Update a Department

To update a department's details:

1. On the homepage, locate the department you want to update.
2. Click on the **Edit** button in the corresponding row.
3. Modify the fields in the form:
   - Department Name
   - Department Description
4. Click on the **Update** button to save the changes.
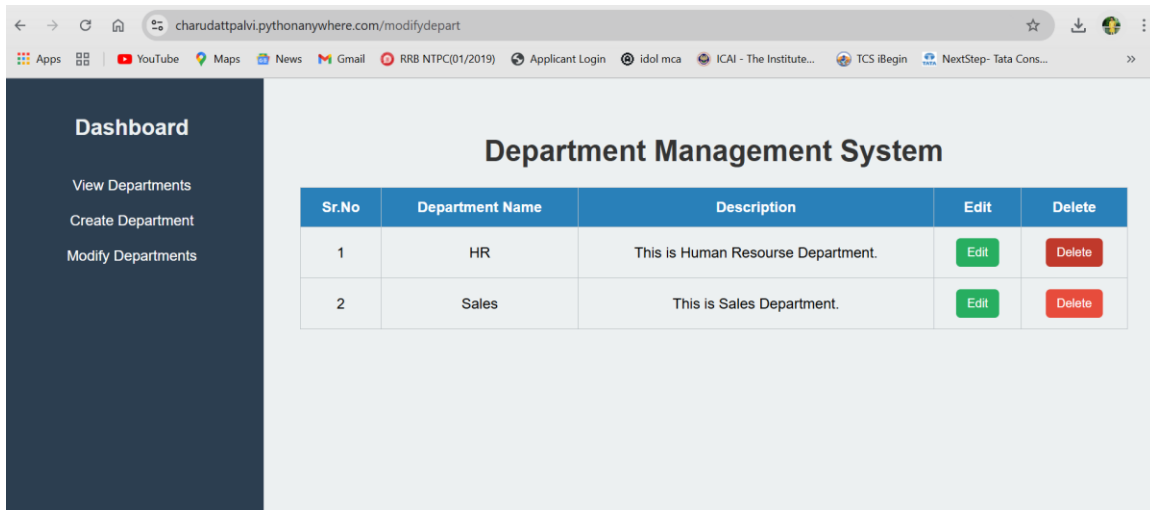5. The updated details will be reflected in the list.

3.4 Delete a Department

To delete (soft delete) a department:

1. On the homepage, locate the department you want to delete.
2. Click on the **Delete** button in the corresponding row.
3. The department will no longer appear in the list.

4. Features and Instructions of Role:-

4.1 View Roles

The homepage displays a list of all active roles in a tabular format. To view roles:
1. Go to the homepage (`https://charudattpalvi.pythonanywhere.com/` or `http://127.0.0.1:8000/`).
2. The table will display the following details:
   - Role ID
   - Role Name
   - Description
3. Options to Edit or Delete a role are also displayed.
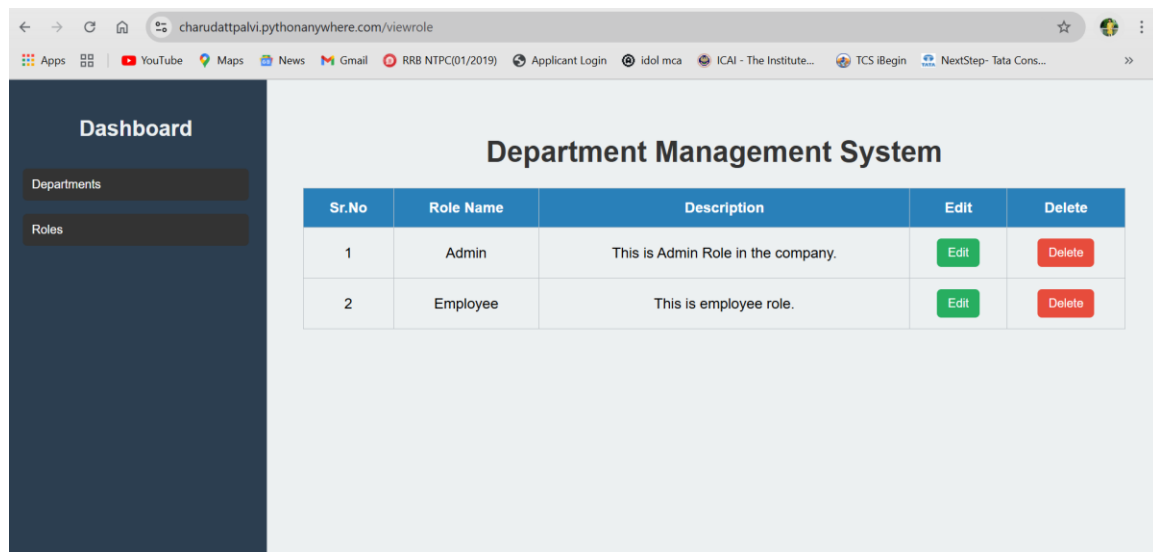
4.2 Add a New Role

To create a new role:
1. Navigate to the 'Create Role page by clicking on **'Create Role** in the sidebar.
2. Fill in the form fields:
   - **Role Name**: Enter the name of the role.
   - **Role Description**: Enter a short description.
3. Click on the **Create** button.
4. The new role will now appear in the list.

3.3 Edit/Update a Role

To update a role's details:

1. On the homepage, locate the role you want to update.
2. Click on the **Edit** button in the corresponding row.
3. Modify the fields in the form:
   - Role Name
   - Role Description
4. Click on the **Update** button to save the changes.
5. The updated details will be reflected in the list.

4.4 Delete a Role

To delete (soft delete) a Role:

1. On the homepage, locate the role you want to delete.
2. Click on the **Delete** button in the corresponding row.
3. The role will no longer appear in the list.
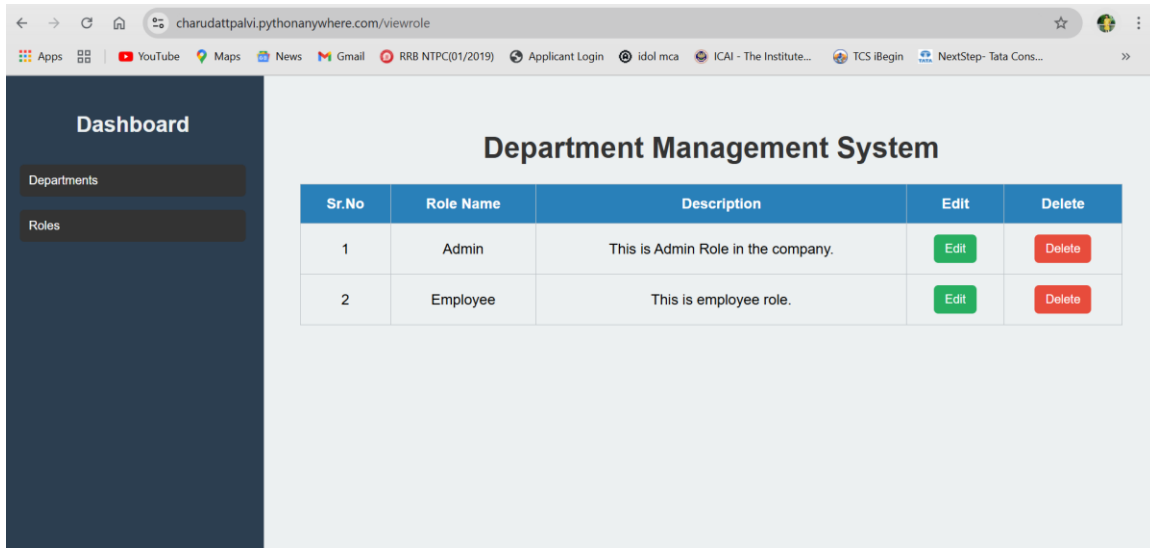
5. Troubleshooting

If you encounter issues, try the following steps:

- **Page Not Loading**: Ensure the server is running. Start it using the command:
  `python manage.py runserver` for local server or check internet connection for live server.
- **Data Not Displayed**: Confirm that the database has the data you are looking for.
- **Form Errors**: Ensure all required fields are filled before submitting.

6. Notes

- This system uses a soft delete mechanism, meaning deleted departments or roles can be restored manually from the database.
- For security, ensure CSRF tokens are included in all form submissions.
- Static files (CSS, JS) must be served correctly for a smooth user interface.