**TODO Task management system:-**

**Overview**

The TODO Management System is a CRUD (Create, Read, Update, Delete) application built using the Django framework. This application allows users to manage TODO tasks, including creating new tasks, viewing a list of tasks, updating task status, and deleting tasks. The frontend of the application is built using Bootstrap for a responsive and modern user interface. Authentication is implemented to ensure secure access to the application, and server-side validations are applied to maintain data integrity.

Index

1 – features

2 – technologies used

3 -

**Features**

* **Authentication and Authorization:** Secure login and logout functionality.
* **TODO Management:**
  + Create new TODO tasks.
  + View a list of tasks ordered by priority.
  + Update the status of existing tasks.
  + Delete tasks.
* **Validation:** Server-side validations to ensure data integrity.
* **Responsive Design:** Built with Bootstrap for a mobile-friendly UI.

**Technologies Used**

* **Backend:** Django
* **Frontend:** Bootstrap
* **Database:** SQLite (default for Django projects)

**Installation**

**Prerequisites**

* Python 3.x
* Django 3.x or higher

**Steps**

1. **Clone the repository:**

git clone <repository\_url>

cd <repository\_directory>

1. **Create a virtual environment:**

python -m venv venv

venv\Scripts\activate

1. **Install dependencies:**

pip install -r requirements.txt

1. **Apply migrations:**

python manage.py migrate

1. **Run the development server:**

python manage.py runserver

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

**Usage**

**Authentication**

* **Login:** Access the login page at /login/ and enter your credentials to log in.
* **Register:** Access the registration page at /signup/ to create a new account.
* **Logout:** Use the logout button to securely log out of the application.

**TODO Management**

* **Home Page:** After logging in, you will be redirected to the home page where you can see a list of your TODO tasks.
* **Add Task:** Fill out the form on the home page and submit to create a new task.
* **Update Task Status:** In the task list, click "Update" next to a task to change its status.
* **Delete Task:** In the task list, click "Delete" next to a task to remove it from the database.

**Server-Side Validations**

The following validations are implemented on the server side:

* **Title:** Required.
* **Priority:** Required, must be a valid integer.
* **Status:** Required, must be either 'C' (Completed) or 'P' (Pending).

**Key Files**

* **views.py:** Contains the view functions for handling requests and rendering templates.
* **models.py:** Defines the TODO model for task data.
* **urls.py:** Maps URL routes to the corresponding view functions.
* **templates/:** Contains the HTML templates for rendering the web pages.