Trackify : Smart and intuitive task management system

Submitted in partial fulfillment of the requirements of the degree of

Bachelor of Engineering (Information Technology)

By

Anushka Shahane Roll No – 54

Department of Information Technology

VIVEKANAND EDUCATION SOCIETY'S INSTITUTE OF TECHNOLOGY,

Chembur, Mumbai 400074

(An Autonomous Institute, Affiliated to University of Mumbai)

April 2024

Contents :

TRACKIFY

Name of student	Anushka Shahane
Class_Roll no	D15A_54
D.O.P	3/04/25
D.O.S	17/04/25
Sign and Grade	

<u>Title :</u> Trackify

<u>Project Description :</u>

Trackify is a smart and intuitive task management system designed to boost productivity by organizing and tracking daily tasks efficiently. Built using **Angular** for the frontend, **Flask** for the backend, and **MongoDB** as the database, Trackify offers a seamless user experience for individuals and teams to stay on top of their schedules and progress. With responsive design and powerful REST APIs, Trackify ensures accessibility and usability across all devices.

<u>Requirement gathering :</u> The requirement gathering for Trackify was conducted through user interviews, observation of daily task management challenges, and analysis of common productivity tools. The primary goal was to identify pain points in existing systems, such as poor UI/UX, lack of real-time tracking, and absence of categorization or visual indicators. Both individual users and small team workflows were considered to ensure the solution remains flexible and scalable. Functional requirements included CRUD operations for tasks, visual progress tracking, and a responsive design, while non-functional requirements focused on performance, security, and scalability. Feedback from early users helped shape the core features and user experience of Trackify.

System Requirements :

1. Hardware Requirements:

- **Processor:** Intel Core i5 / AMD Ryzen 5 or higher (dual-core, 2.0 GHz or faster)

- **RAM:** Minimum 8GB (16GB recommended)

- **Storage:** At least 1GB free space (256GB SSD recommended)

- **Network:** Stable internet connection (especially for MongoDB Atlas users)

2. Software Requirements:

- **Operating System:** Windows 10/11, macOS 10.15+, or Ubuntu 20.04+

- **Code Editor:** Visual Studio Code or compatible IDE

- **Version Control:** Git 2.25+

Technologies Used :

Development	VS Code , Postman , Git
Frontend	Angular (v19)
Backend	Flask (Python 3.8+)
Database	MongoDB
Styling	SCSS / Bootstrap
APIs	RESTful Flask APIs

Setup Instructions :

- **Node.js** and **Angular CLI :** To set up Trackify, first ensure that Node.js and Angular CLI are installed. Visit the official Node.js website and download the LTS version suitable for your operating system. After installation, verify it using node -v and npm -v in your terminal. Once Node.js is installed, open a terminal or command prompt and install Angular CLI globally by running **npm install -g @angular/cli**, then confirm the installation using ng version

- **Python 3.8+ :** Next, install Python 3.8 or higher by visiting the official Python website. Download the appropriate installer for your OS and during installation (especially on Windows), ensure you check the option "Add Python to PATH." After installation, verify it by running python --version and pip --version in the terminal. Optionally, for better environment management, you can create a virtual environment using python -m venv venv' and activate it with venv\Scripts\activate on Windows or source venv/bin/activate on macOS/Linux.

- **MongoDB** (Local or Cloud - **MongoDB Atlas**) : Finally, set up **MongoDB** either locally or via **MongoDB Atlas**. For a local setup, download MongoDB Community Edition from the MongoDB website and follow the platform-specific installation instructions. To use MongoDB Atlas, go to https://www.mongodb.com/cloud/atlas, create a free cluster, set up a database, and obtain the connection string for use in your Flask backend.

Backend Setup :

1. Navigate to backend folder:

   cd project

2. (Optional) Create a virtual environment:

   python -m venv venv

   venv\Scripts\activate   # For Windows
3. Install dependencies:

   pip install -r requirements.txt
4. Start the Flask server:

   cd api
   python app.py

Backend will run at: http://localhost:5000

Frontend Setup

1. Navigate to frontend folder:
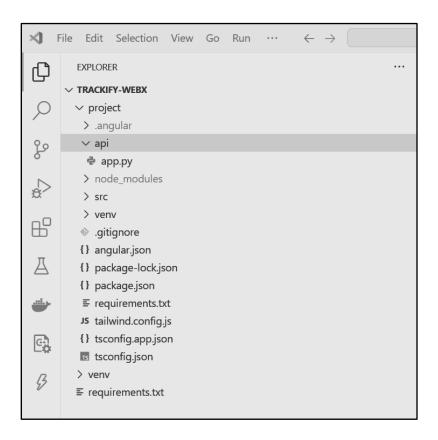
   cd project
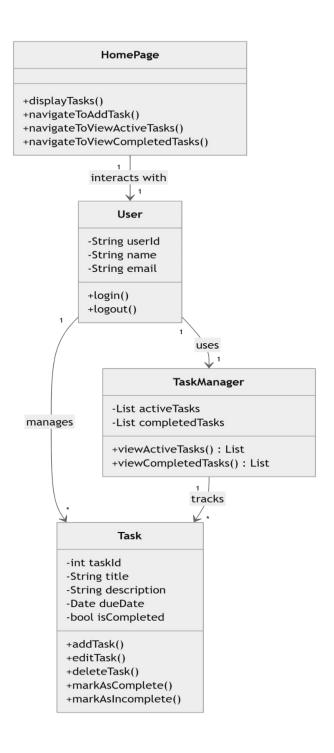
2. Install dependencies:

   npm install

3. Start Angular development server:

    ng serve

Frontend will run at: http://localhost:4200
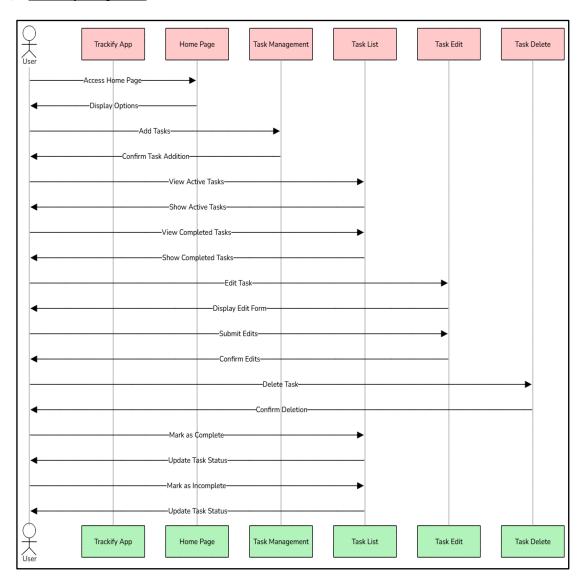
Project Structure
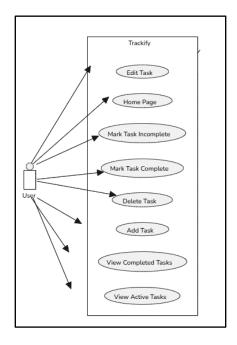
Architectural Diagrams :

a) Class Based Diagram -

HomePage

+displayTasks()
+navigateToAddTask()
+navigateToViewActiveTasks()
+navigateToViewCompletedTasks()

1
interacts with
1

User

-String userId
-String name
-String email

+login()
+logout()

1

manages

1

uses

1

TaskManager

-List activeTasks
-List completedTasks

+viewActiveTasks() : List
+viewCompletedTasks() : List

1
tracks
*

*

Task

-int taskId
-String title
-String description
-Date dueDate
-bool isCompleted

+addTask()
+editTask()
+deleteTask()
+markAsComplete()
+markAsIncomplete()

b) Swimlane Diagram :

c) <u>Activity diagram :</u>

Features Implemented :

1. **Task Management**: Create, update, delete, and organize tasks.

2. **Progress Tracking**: Visual indicators for task completion.

3. **Responsive Design**: Mobile-first UI design using SCSS/Bootstrap.

4. **REST APIs**: Communication between Angular frontend and Flask backend

5. **Google Analytics**: Tracked user activity and engagement.

6. **Email OTP Authentication**: Secured login/registration with one-time password.

7. **AI Task Assistance:** Smart suggestions and support for task management.

8. **Task Breakdown:** Split tasks into subtasks for better tracking and clarity.

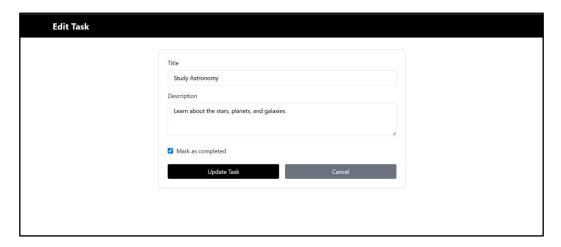## Screenshots of implementation :

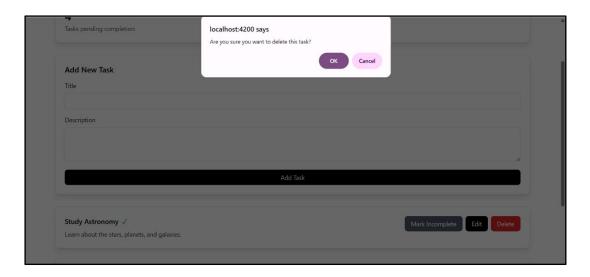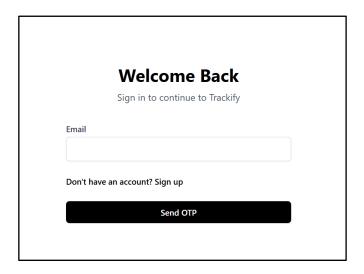Home Page -

Add tasks page -

View active tasks page -

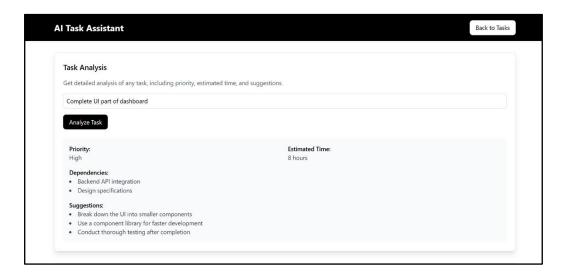View completed tasks page -
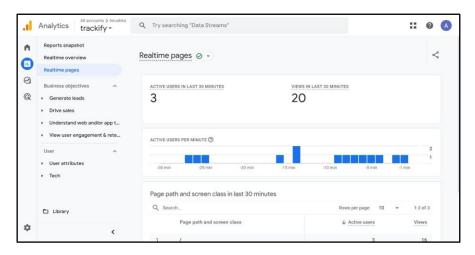
Edit task page -

Delete task -

Login via OTP :
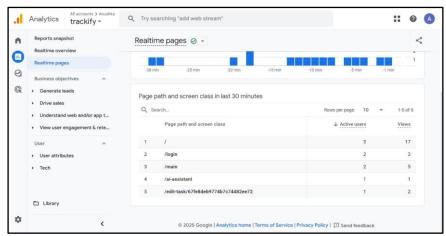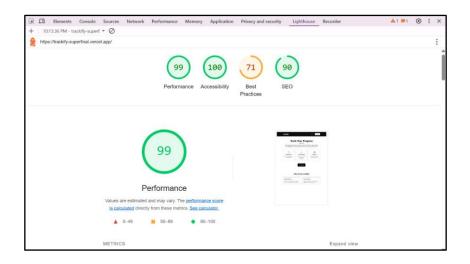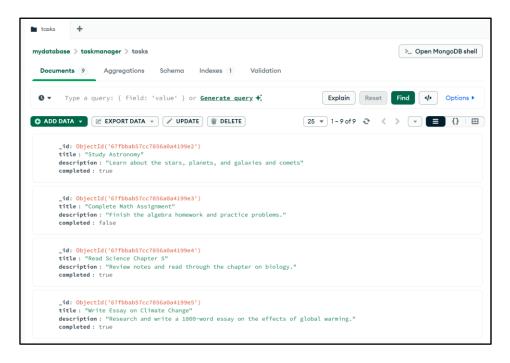

AI Task Assistance :

Google Analytics :

Database (MongoDB) -

Future Scope : Trackify has the potential to evolve into a comprehensive productivity ecosystem. In the future, it can integrate AI-powered features such as intelligent task recommendations, automatic prioritization based on user habits, and smart reminders using natural language processing. Adding voice assistant support can enable users to manage tasks hands-free, improving accessibility and ease of use. Furthermore, cross-platform synchronization with calendar tools, email clients, and wearable devices can provide real-time updates and seamless productivity tracking across devices. With enhanced collaboration tools, Trackify can also be expanded to support project management features for small to mid-sized teams.

Github Link : https://github.com/Anushka3204/Trackify_AI

<u>Conclusion:</u>

Trackify streamlines task management through a well-integrated tech stack comprising Angular, Flask, and MongoDB. The setup process involves installing essential tools like Node.js, Angular CLI, Python, and MongoDB, ensuring a robust development environment. With a responsive interface and intuitive features such as real-time task tracking and seamless CRUD operations, Trackify enhances productivity for both individuals and teams. The successful configuration and deployment of this system demonstrate practical implementation of full-stack development principles, making Trackify a valuable solution for efficient task management in modern workspaces.