**Task Flow: Project Management Tool**

**1. Problem Statement**

Managing multiple projects and tasks can be overwhelming, especially when working on various platforms like **Green Grid**, **Shelter Soul**, **Cinema Sphere**, and **Weather Dashboard**. The challenges include:

* **Task Management**: Difficulty in tracking tasks, their progress, and deadlines.
* **Project Organization**: Lack of a centralized platform to manage all projects in one place.
* **Collaboration**: Inefficient communication and task assignment among team members.
* **Progress Tracking**: No clear way to visualize the status of tasks (To-Do, In-Progress, Done).

**2. Solution: Task Flow**

**Task Flow** is a **project management tool** designed to help users organize, track, and manage their projects and tasks efficiently. It provides a **centralized dashboard** where users can create projects, add tasks, and track their progress. The tool is especially useful for developers, students, and teams working on multiple projects simultaneously.

**3. Key Features**

**3.1 Dashboard**

* **Centralized View**: Users can view all their recent projects (e.g., Green Grid, Shelter Soul, Cinema Sphere, Weather Dashboard) in one place.
* **Quick Access**: Easy access to create new projects or view existing ones.

**3.2 Task Management**

* **Task Categories**: Tasks are categorized into **To-Do**, **In-Progress**, and **Done** for easy tracking.
* **Task Actions**: Users can **add**, **update**, **edit**, and **delete** tasks dynamically.
* **Progress Tracking**: Visual representation of task progress helps users stay on top of their work.

**3.3 Project Creation**

* **New Project**: Users can create new projects by providing a **project name** and **description**.
* **Recent Projects**: Easy access to recently worked-on projects for quick navigation.

**3.4 User Profile**

* **Profile Management**: Users can edit their profile details, including **full name**, **job**, **institution**, **email**, and **location**.
* **Recent Projects**: Display of the most recent projects worked on by the user.

**4. Technologies Used**

**4.1 Frontend**

* **React.js**: For building a dynamic and responsive user interface.
* **React Router DOM**: For client-side routing and navigation.
* **Flowbite & Flowbite-React**: For pre-designed UI components and responsive design.
* **FontAwesome & Phosphor Icons**: For scalable vector icons.

**4.2 Backend**

* **Express.js**: A Node.js framework for building the backend server.
* **MongoDB**: A NoSQL database for storing project and task data.
* **Mongoose**: An ODM (Object Data Modeling) library for MongoDB.
* **GraphQL**: For efficient data querying and manipulation.
* **Apollo Server**: For implementing the GraphQL API.
* **JSON Web Tokens (JWT)**: For secure user authentication and authorization.
* **Bcrypt**: For password hashing and secure storage.

**4.3 Other Tools**

* **Dotenv**: For managing environment variables.
* **CORS**: For enabling cross-origin resource sharing.
* **Vitest**: For testing the application.
* **Happy DOM**: For simulating the browser environment during testing.

**5. Impact of Task Flow**

**5.1 For Developers and Students**

* **Improved Productivity**: Centralized task management helps users stay organized and focused.
* **Better Collaboration**: Teams can easily assign tasks and track progress, improving collaboration.
* **Enhanced Time Management**: Clear visualization of task status helps users prioritize and manage their time effectively.

**5.2 For Project Management**

* **Streamlined Workflow**: Task Flow simplifies the process of creating, updating, and tracking tasks.
* **Increased Efficiency**: Users can quickly access their recent projects and tasks, reducing time spent on navigation.

**6. Hackathon-Ready Features**

If you're presenting **Task Flow** at a hackathon, here are some **key highlights** to focus on:

* **Dynamic Task Management**: Showcase how users can add, update, edit, and delete tasks dynamically.
* **Progress Tracking**: Highlight the **To-Do**, **In-Progress**, and **Done** categories for task tracking.
* **User Profile**: Demonstrate the profile management feature, including editing personal details and viewing recent projects.
* **Project Creation**: Show how users can create new projects and manage them efficiently.

**7. Resume Points**

Here’s how you can include **Task Flow** in your resume:

**Project: Task Flow – Project Management Tool**

* **Role**: Full-Stack Developer
* **Technologies**: React.js, Express.js, MongoDB, GraphQL, JWT, Bcrypt, Flowbite, FontAwesome, Phosphor Icons.
* **Key Contributions**:
  + Designed and implemented a **centralized project management tool** for tracking tasks and projects.
  + Developed **dynamic task management** features, including adding, updating, editing, and deleting tasks.
  + Integrated **GraphQL APIs** for efficient data querying and manipulation.
  + Implemented **JWT-based authentication** for secure user login and authorization.
* **Impact**: Improved productivity and collaboration for developers and students by providing a streamlined task management system.

**8. PPT Structure**

Here’s a suggested structure for your **PPT presentation**:

**Slide 1: Title Slide**

* **Title**: Task Flow – Project Management Tool
* **Subtitle**: Streamlining Task and Project Management
* **Your Name**: [Your Name]
* **Date**: [Presentation Date]

**Slide 2: Problem Statement**

* Challenges in task and project management (e.g., lack of organization, inefficient collaboration).

**Slide 3: Solution Overview**

* Introduction to Task Flow and its key features.

**Slide 4: Key Features**

* Dashboard, Task Management, Project Creation, User Profile.

**Slide 5: Technologies Used**

* Frontend: React.js, Flowbite, FontAwesome.
* Backend: Express.js, MongoDB, GraphQL, JWT.

**Slide 6: Impact**

* Benefits for developers, students, and teams.

**Slide 7: Demo**

* Live demo of the platform (if possible) or screenshots of key features.

**Slide 8: Future Enhancements**

* Potential future features (e.g., team collaboration, mobile app).

**Slide 9: Conclusion**

* Recap of the problem, solution, and impact.
* Call to action: Encourage users to try Task Flow for better project management.

**9. PDF Structure**

For a **PDF document**, you can follow a similar structure as the PPT but with more detailed explanations. Include:

* **Cover Page**: Title, your name, and date.
* **Table of Contents**: List all sections.
* **Problem Statement**: Detailed explanation of the challenges.
* **Solution**: In-depth description of Task Flow and its features.
* **Technologies Used**: Detailed breakdown of the tech stack.
* **Impact**: Metrics and success stories (if available).
* **Future Work**: Potential enhancements and next steps.
* **Conclusion**: Summary and call to action.