**Question: Build a Simple Task Management App**

**Time Estimate: ~3 Hours**

**Objective:**

Create a simple Task Management web application where users can create, view, update, and delete tasks.

**Requirements**

1. **Frontend:**
   * Build a simple UI to display a list of tasks.
   * Allow users to add a new task, edit an existing task, and delete a task.
   * Each task should have:
     + A title (short text)
     + A description (longer text)
     + A status (e.g., "To Do", "In Progress", "Done")
   * Show basic validation (e.g., title cannot be empty; the more validation the more **bonus**).
   * Add filtering (e.g., view only "In Progress" tasks; the more filtering the more **bonus**).
2. **Backend:**
   * Implement a REST API or GraphQL API that:
     + Allows CRUD operations on tasks (Create, Read, Update, Delete).
     + Stores tasks in a simple in-memory database, JSON file, or SQLite.
   * Keep the backend modular and structured (avoid monolithic files).
   * Implement at least basic authentication (e.g., users must log in to manage tasks; **bonus** for advanced authentication).
3. **General Guidelines:**
   * Use any frontend and backend tech stack you are comfortable with.
   * The application does not need to be fully functional or polished. Focus on clean, readable, and well-structured code but **bonus** for functional app.
   * Use version control (Git), and include README.md with setup instructions.
   * Write meaningful commit messages if using Git.
   * If you can’t complete it fully, write comments on how you would proceed.

**Evaluation Criteria**

* **Code Quality:** Is the code structured, readable, and maintainable?
* **Coding Patterns:** Are good practices followed (e.g., modularisation, reusable components, clean API design)?
* **Error Handling:** Are potential errors considered and handled gracefully?
* **Basic UI/UX Considerations:** Does the frontend provide a clear, intuitive interface?
* **Use of Version Control:** Are commits meaningful and structured?
* **Bonus Considerations:** Any extra effort (e.g., filtering, authentication) is a plus!