

Task Board – Full Stack Developer Assignment

Time Expectation: 2–4 Hours

Purpose: This assignment is designed to quickly assess a developer's core full-stack knowledge, including backend logic, database design, authentication, and frontend integration. Polish is not the priority; clarity and correctness are.

Objective

Build a very small Task Board application where a user can log in, create tasks, and update task statuses. The focus is on clean implementation and understanding of fundamentals.

Tech Stack Requirements

- Next.js (App Router preferred)
- TypeScript
- Relational Database (PostgreSQL / MySQL / SQLite)
- Any ORM (Prisma / Drizzle / TypeORM)
- Tailwind CSS (basic usage is sufficient)

Restrictions

- No Firebase or MongoDB
- No website builders or AI-generated sites
- Application must include backend + database (no static-only apps)

Features to Build

1. Authentication (Minimal)

- User signup and login
- Single role: User
- Secure password hashing

2. Task Management

- Create a task with title and status (Todo / In Progress / Done)
- View all tasks created by the logged-in user
- Update task status

“No delete, no priority, no filters required.”

3. User Interface

- Single page after login
- Simple task list with status update option
- Basic loading and empty states
- Responsive layout

Backend Expectations

- Clean database schema (User → Task relationship)
- Clean API routes or server actions
- Input validation and error handling
- Sensible project structure

Deployment

Deployment is optional. If completed, deploy on Vercel or Render. Local setup is acceptable if time runs out.

Documentation (README.md)

- Tech stack used
- Short explanation of authentication flow
- Database schema explanation (text or diagram)
- Steps to run the project locally
- Live URL (if deployed)

Evaluation Criteria

We evaluate clarity of thought, correctness, and understanding of full-stack fundamentals. Design polish and extra features are not required.



Digitally Next

Think. Act. Disrupt.

