

## BetterFlow - 3-Hour Technical Test Task

### Objective

Develop a **Task Management API** using **PHP, Laravel, and MySQL**. This API should allow users to create, retrieve, update, and delete tasks while following best practices for clean and maintainable code.

### Task Overview

You will build a UI **RESTful API** with the following endpoints:

1. **Create a Task**
  - Fields: title (string, required), description (string, optional), priority (low, medium, high - default: medium), due\_date (datetime, optional).
2. **Retrieve All Tasks**
  - Return a **list of tasks**, sorted by creation date (descending).
3. **Retrieve a Single Task**
  - Return task details by ID.
4. **Update a Task**
  - Allow modifying title, description, priority, OR due\_date.
5. **Delete a Task**
  - Soft delete (bonus if implemented).

### Technical Requirements

- Use **Laravel** as the backend framework.
- Store data in a **MySQL** database.
- Implement **proper request validation**.
- Follow **RESTful principles** (HTTP methods, status codes).
- Write **clean, maintainable code** using Laravel best practices.
- Authentication is **not required**.
- Testing:
  - Manual testing using **Postman** or similar tools (unit tests are optional).
  - Include test cases for **at least two API endpoints**.

### Expected Submission

1. **Code Repository**: Submit your code via **GitHub (preferred)** or as a **ZIP file**.
2. **Database Schema**: Provide the **SQL schema** or **Laravel migration files**.
3. **Video Explanation (5-10 minutes)**:
  - Explain your coding approach and logic.
  - Highlight best practices (clean code, security, performance).
  - Show how you tested/debugged the API.

### Time Breakdown (3 Hours)

Task	Estimated Time
Setup Laravel & MySQL Configuration	15 min
Create Task Model & Migration	15 min
Implement CRUD API (Controller & Routes)	45 min
Add Validation & Error Handling	20 min
Test API with Postman & Debug	30 min
Record Video Explanation	10 min
Buffer Time (Unexpected Issues, Review, Wrap-up)	30 min
Total	~3 hours

### Bonus (If Time Allows)

- Implement **soft delete** for tasks.
- Add **pagination** for the task list.
- Implement **sorting & filtering** (e.g., filter by priority or due\_date).
- Write **unit tests** for at least one endpoint.