

# Hiring Test — Global Project Management System

## Introduction

You will build a **Project Management System** that allows authenticated users to manage projects and related tasks.

The purpose is to evaluate your ability to:

- Design and implement a **secure, maintainable RESTful API** in Laravel
- Build a **modern, responsive frontend** in Vue.js
- Apply **authentication, authorization, advanced queries, and validation**
- Write **tests** and document your design choices

You may choose **any relational database** (MySQL, PostgreSQL, SQLite, etc.) and **any CSS framework** (Tailwind, Bootstrap, Bulma, custom CSS).

---

## Requirements Overview

You may implement the solution as:

- **A single Laravel app** using Inertia + Vue.js, or
- **Two separate projects** (Laravel API + Vue SPA).

If split, use **Laravel Sanctum** for authentication.

---

## Backend Requirements

### Entities

#### Projects

- name (string, required, unique per creator)
- start\_date (datetime, required)

- end\_date (datetime, required, after or equal to start\_date)
- value (decimal, required, min: 0)
- status (enum: Active | Inactive)
- creator\_id (authenticated user who created the project)
- timestamps

### **Tasks** (related to projects)

- project\_id (foreign key to projects)
  - title (string, required)
  - due\_date (datetime, required)
  - status (enum: Pending | Completed | Inactive)
  - timestamps
- 

## **Features**

### **1. Authentication & Authorization**

- Auth required for all endpoints
- Use **Laravel Policies** to ensure:
  - Only project owners can edit/delete their own projects and tasks
  - Other users can only view

### **2. CRUD**

- Projects: create, list (paginated), show, update, delete
- Tasks: create, list (paginated per project), update, delete

### **3. Business Rules**

- Setting a project to Inactive automatically sets all its tasks to Inactive
- A task can only be edited by the project owner or the task creator

### **4. Query Features**

- Search projects by name

- Filter by status
- Sort by start\_date, end\_date, or value
- Combine filters, search, and sorting in the same request
- Allow page\_size parameter for pagination

#### 5. **Validation**

- Clear, user-friendly error messages
- Use custom rules where necessary

#### 6. **Database**

- Include migrations and seeds
  - At least one seeded user with credentials in README
- 

## **Frontend Requirements**

#### 1. **Auth**

- Simple login/logout flow

#### 2. **Project Management Page**

- Paginated list of projects with search/filter/sort controls
- Create/Edit forms with validation messages
- Delete with confirmation dialog

#### 3. **Task Management**

- List tasks inside a project (paginated)
- Create/Edit/Delete tasks with inline validation

#### 4. **UI**

- Responsive design
  - Any CSS framework of your choice
  - Clear feedback (loading states, empty states, success/error toasts)
-

# Testing Requirements

- At least **one feature test** for the project API
  - At least **one unit test** for a business rule
  - Tests must pass (php artisan test)
- 

## Deliverables

### 1. Public GitHub Repository

- Public or with access granted to reviewers

### 2. README (Required)

- Setup & run instructions
- Environment config (.env.example)
- Seed user credentials
- API endpoint list with examples
- Explanation of architecture and trade-offs
- Next steps you would take for production readiness

### 3. Code Quality

- Consistent naming and formatting
  - Clear commit messages
  - Proper use of Laravel features (Resources, Policies, Validation, Scopes)
- 

## Acceptance Criteria

- Meets **all functional requirements**
- Code is **clean, maintainable, and follows Laravel/Vue best practices**
- Backend and frontend run with minimal setup following the README
- UI is responsive and user-friendly
- Tests pass

---

## Nice to Have (Optional Bonus)

- State management in frontend (e.g., Pinia, Vuex)
  - API documentation (Swagger/OpenAPI)
  - Docker/Sail setup for easy run
  - CI/CD pipeline (GitHub Actions)
  - Accessibility improvements (ARIA, keyboard navigation)
- 

## Submission

- Share the **GitHub repository URL**
- Ensure the README allows a reviewer to run everything quickly