

Laravel Project Report Template

| | Name | ID | Section |
|------------|------------------------|----------|---------|
| Student 1: | Mahdi Khazaal Khanafer | 10120246 | D |
| Student 2: | Malak Khanafer | 10121681 | C |

Project Title

Collaborative Task Management System

Project Description

The Collaborative Task Management System is designed to help users efficiently manage tasks and projects. The platform allows users to create projects, add tasks (which may or may not belong to a project), assign tasks to other users (only by the project owner), and track task completion status. It also features authentication, sorting, filtering, and progress tracking to ensure effective task management.

Core Features

- User authentication and registration (login/logout)
- Project and task management (CRUD operations)
- Task assignment (only by project creator)
- Task status tracking (To Do, In Progress, Done)
- Task priority and due date management
- Task sorting by priority or deadline
- Project completion tracking based on task progress
- Role-based access control (project owners assign tasks)

Models and Migrations

Models:

- **User:** Manages authentication and user-related data.
- **Project:** Stores project details including name, description, owner_id, and progress.
- **Task:** Contains task details such as name, description, due_date, priority, status, project_id, assigned_to, and created_by.

Migrations:

- Users table (default Laravel migration)

- Projects table: Contains id, name, description, owner_id (foreign key to users), progress.
- Tasks table: Contains id, name, description, due_date, priority, status, project_id (nullable foreign key to projects), assigned_to (nullable foreign key to users), created_by.

Frontend Overview

Technologies Used:

- HTML, CSS (Bootstrap for styling)
- JavaScript (for smooth page transitions)
- Blade templates (Laravel's templating engine)

Views:

- welcome.blade.php - Landing page
- home.blade.php - Home after login
- dashboard.blade.php - User dashboard
- projects/index.blade.php - List of projects
- projects/show.blade.php - Project details with tasks
- tasks/index.blade.php - List of all tasks
- tasks/create.blade.php - Form for adding a task
- auth/login.blade.php - Login page
- auth/register.blade.php - Registration page

Backend Overview

Routes:

- / - Landing page
- /home - Redirects to dashboard
- /dashboard - Dashboard view
- /projects - Project list (CRUD)
- /tasks - Task list (CRUD)

Controllers:

- **ProjectController:** index, show, create, store, edit, update, destroy

- **TaskController:** index, show, create, store, edit, update, destroy
- **DashboardController:** index

Middleware Strategy:

- **Authentication:** Laravel's built-in auth middleware to restrict access.
- **Authorization:** Middleware ensuring that only the project owner can assign tasks.
- **Validation Middleware:** Laravel Form Request validation for inputs.

Form Handling & Validation

Forms in the Application:

- User registration
- Login
- Project creation
- Task creation
- Task assignment (only by project owner)

Validation Methods:

- Laravel request validation (`$request->validate([...])`)
- Custom validation rules for ensuring tasks belong to a valid project

Roles and Permissions

Roles:

- **Project Owner:** Can create projects, add tasks, assign tasks, delete tasks and projects.
- **Regular User:** Can only manage tasks assigned to them.

Implementation:

- **Middleware-based Role Checking:** Restrict assignment actions to project owners only.
- **Laravel Gates & Policies:** To restrict task assignments.

Web API (if applicable)

Planned API Endpoints:

- GET /projects - Retrieve all projects
- POST /projects - Create a new project
- GET /tasks - Retrieve all tasks
- POST /tasks - Create a task
- PUT /tasks/{id} - Update task status
- DELETE /tasks/{id} - Delete a task

Authentication Method:

- Laravel Sanctum for API authentication

Third-Party Integrations (if any)

- Bootstrap for UI styling
- Laravel Sanctum for API authentication
- Optional: Email notifications using Laravel Mail for task assignments