



# Laravel Backend Assignment

## Task Management API

Task management is the process of creating, prioritizing, delegating, and monitoring tasks to ensure they are completed within given deadlines. It also involves organizing the workflow and resolving bottlenecks to ensure efficient task completion. Task management is an essential component of effective project management and successful business operations. Some of the task management software tools like Trello, Asana, Jira, ClickUp, ...etc.

The program includes three user roles: Product owner, developer, and tester. Each role has distinct responsibilities and can perform specific tasks.

### Product Owner

The Product Owner manages tasks by creating, updating, and deleting them. They assign tasks to developers and review completed tasks, marking them as DONE or returning them to IN PROGRESS. They also manage users by creating, updating, and deleting profiles and assigning titles.

### Developer

They are responsible to move only their tasks from TODO to IN PROGRESS and from IN PROGRESS to READY FOR TEST.

### Tester

They are responsible to move only their tasks from READY FOR TEST to PO REVIEW.

### Task

A task is a specific work or duty that someone needs to complete. It may have one or more subtasks. The Product Owner is responsible for creating tasks, which are then assigned to a developer. The developer updates the task status, and each task can have only one status at a time. The program includes various statuses for each task:

- TODO: Someone has to do/implement this task (it is the default status for every task)
- IN PROGRESS: when the developer starts the task.
- READY FOR TEST: when the developer finished the task
- PO REVIEW: after the tester tests the task. The Product Owner has to review it.
- DONE: when the Product Owner confirms that the task is finished.
- REJECTED: only the Product Owner can change a task to reject from any status.

The program should save logs (when the status changes or the task is updated) in the database and it should be returned when someone requests the task detail.



# Fastlink company

## Business Development Department



Each task should have these properties:

- Title.
- Description.
- Assigned to (this can be null when the task is from TODO).
- Created at.
- Due date.
- Current status.
- Log of the change on tasks: changing status, changing the title or changing the description.
- List of subtasks (id, title, description, status, assigned to).
- Parent Id (if it has a parent).

The mentioned details should be returned when the user requested to get the task (e.g. /task/1).

This automation should be implemented through the tasks when their status has been changed:

- When the task is moved to PO REVIEW, it should automatically be assigned to the Product Owner (the one who created the task)
- When the task is moved to READY FOR TEST, it should be assigned to a tester who has less number of tasks.
- When the Product Owner returned the task to IN PROGRESS or moved to DONE, it should be assigned to the developer who did the task.

## Features

Regarding the above requirements and program description you also need to implement these things:

- Simple token-based authentication.
- Users can be created through an API and only Product Owners have this access, also they can be created through a Laravel artisan command.
- An API to return a list of the tasks with those details (id, title, description, status, assigned to), all users should have access to this API and can see the list of tasks, they can search through them by title, description, or ID or filter them by assigned to (it is multi-select). In addition the tasks can be searched using the name of the assignee.
- An API to return the details of a task as mentioned above.
- A Laravel command to export tasks to a CSV file.
- A Laravel command to import tasks from a CSV file into a database asynchronously using batch jobs.
- Provide an API that returns the progress of the above job import process.
- The program should send an alert to the product owner (who created the task) using web-socket when the task passed its due date.



# Fastlink company

## Business Development Department



- The program should send an email to the user when a task is assigned to them. If the "assigned to" field is null, no email needs to be sent. The email should be sent in the background.

### Required

- You are only in charge of implementing the backend of this project. (APIs).
- The project should be implemented using:
  - Laravel PHP framework.
  - Any DBMS. (MySQL, PostgreSQL, Oracle..etc).
- Using postman: structure all your APIs in postman and export the collection and include it in the root of your project before pushing to GitHub.
- Populate the database with some fake data during project installation.
- Please ensure the system is completed within 72 hours. Before the deadline, share the source code in a public GitHub repository.

### Nice to have

- Provide API documentation as well as setup instructions for the project.
- Unit/Feature testing.
- Dockerizing your app.
- Deploying your code on a free hosting service or a server.

### What's important to us?

- Solving the given problem.
- Input validation.
- Good database design.
- Clean Code.
- OOP.
- API structure.