Conditions

You have to prepare a solution to the proposed problem in the defined period of time. The solution must comply with the requirements. For anything not explicitly listed, you are free to choose whatever technology/library/tool you feel comfortable with.

Once ready, you must send a link to access the source code of the solution in any git repository you choose, so it can be built and reviewed. Instructions how to use the solution must also be provided (resource names, SQL scripts to import test data, other scripts, etc.).

*If you have completed the task after the deadline has expired, you are still encouraged to submit a solution.*

Software Requirements

Programming languages: PHP, TypeScript

Framework: Lumen  
Database: PostgreSQL

UI: Angular

Description

This project involves creating a RESTful API to manage TODO lists and their associated items. The API will allow users to store, retrieve, update, and delete TODO lists and their items. The data will be stored in a database.

**Requirements**

1. Create a TODO List
   1. Users can create a new TODO list.
   2. Fields for a TODO list:
      1. Name (string, unique): The name of the TODO list.
      2. Description (string, optional): A brief description of the list (e.g., “Work tasks,” “Personal goals”).
      3. Date to Complete (date-time): The deadline for completing the entire list.
      4. Items: A list of items associated with this TODO list.
   3. Validation rules:
      1. The name must be unique across all TODO lists.
      2. The date to complete must be a valid date-time format.
      3. No more than 10 items are allowed per TODO list.
2. Retrieve All TODO Lists
   1. Users can retrieve information about all stored TODO lists, including their associated items.
   2. The response should include:
      1. TODO list names.
      2. Descriptions (if available).
      3. Item count for each list.
3. Retrieve Details for a Single TODO List
   1. Users can retrieve detailed information about a specific TODO list.
   2. The response should include:
      1. TODO list name.
      2. Description (if available).
      3. Date to complete.
      4. List of associated items with their names and statuses.
4. Add an Item to a TODO List
   1. Users can add a new item to an existing TODO list.
   2. Fields for an item:
      1. Name (string, unique): The name of the item.
      2. Status (string): The status of the item
   3. Validation rules:
      1. The item name must be unique within the TODO list.
      2. The available status should be: “To Do,” “In Progress,” “Done”
5. Remove an Item from a TODO List
   1. Users can remove an item from an existing TODO list.
6. Remove a TODO List
   1. Users can remove an existing TODO List and its associated items.

Other considerations

Please, provide

* Basic UI build on Angular
* Readme file with installation guides.