**Backend​**

In this assessment, you will build a backend API for a to-do list application. Focus only on developing the API and data processing logic without creating a user interface. Use your preferred programming language and framework. ​

**1.**

**Create a RESTful API for a to-do list application:​**

◦

Create new tasks with title, description, due date, and priority​

◦

Read/list all tasks with filtering and pagination​

◦

Update task details and status​

◦

Delete tasks​

**2.**

**Task Dependencies:​**

◦

Allow tasks to depend on other tasks (e.g., "Task B can only be started after Task A is completed")​

◦

Implement endpoints to add/remove dependencies between tasks​

◦

Implement an endpoint that allows users to view all dependencies of a task, including both direct dependencies (first level) and indirect dependencies (all subsequent levels). ​

**3.**

**Dependency Validation:​**

◦

Implement an algorithm to detect circular dependencies​

◦

When a user attempts to create a dependency, validate that it won't create a circular reference​

◦

Return appropriate error messages when circular dependencies are detected​

**4.**

**Caching and Performance Optimization (Optional):​**

◦

Implement a caching mechanism for frequently accessed data (ex: memory cache, ..etc)​

◦

Optimize database queries for performance (In case your application is famous with millions of records)​

**5.**

**Notification System (Optional):​**

◦

Create a notification system for upcoming/overdue tasks​

◦

Implement a background job scheduler for periodic tasks​

**6.**

**Deployment CI/CD (Optional)​**

◦

Create a Dockerfile for your application​

◦

Set up a docker-compose.yml file that includes your application and database​

**Important Note​**

You don't need to complete all the requirements. Submit what you can finish within the timeframe. Your progress will be evaluated based on what you submit. ​