Assignment 1

TASK

You have to create following APIs:

1. Create task - input is title, description and due_date with jwt auth token
2. Create sub task - input is task_id
3. Get all user task(with filter like priority, due date and proper pagination etc)
4. Get all user sub tasks (with filter like task_id if passed)
5. Update task- due_date, status-"TODO" or "DONE" can be changed
6. Update subtask - only status can be updated - 0,1
7. Delete task(soft deletion)
8. Delete sub task(soft deletion)

And the following cron jobs

1. Cron logic for changing priority of task based on due_date of task (refer below for priority)
2. Cron logic for voice calling using twilio if a task passes its due_date. Calling should be based on priority of the user, i.e. first the user with priority 0 should be called, then 1 and then 2. The user should only be called if the previous user does not attend the call. This priority should be fetched from the user table.

Instructions:

- Proper validation should be there while taking input and authenticating user for api calls
- Error handling should be implemented wherever necessary and user friendly error should be thrown
- You can use https://jwt.io/ for creating a jwt token with user_id and only corresponding decoding logic should be there
- You should also update the corresponding sub tasks in case of task updation and deletion
- Sub task model and user table is given, you have to make task model accordingly
- Task should also have priority and status (refer below for both)
- You can use postman to demonstrate all the apis

Sub Task model

id (int, unique identifier)

task_id (int)//references task table
status (0,1) //0- incomplete, 1- complete
created_at (date/string)
updated_at (date/string)
deleted_at (date/string)

User model

id (int, unique identifier)
phone_number (num)
priority (0,1,2) //for twilio calling priority

Priority for task model

0 - Due date is today //0
1 - Due date is between tomorrow and day after tomorrow // 1-2
2 - 3-4
3 - 5+

Status for task model

“TODO” - when no sub task is finished
“IN_PROGRESS” - when at least 1 sub task is finished
“DONE” - when every sub task is completed

Assignment submission details

When you’re ready, please go ahead and start the assignment.

- Use your own IDE to write the code. Once you are done upload the same on GitHub.
- Use mongodb to store the data and use Nodejs and ExpressJs to create api.
- Create a readme file explaining your project. Deeply test your project using Post-Man.

This comprehensive task list ensures that the prospective intern can showcase their skills in frontend and backend development, database integration, security implementation, and documentation. The live changes during the interview assess their ability to adapt and make modifications in real-time, aligning with the dynamic nature of the development environment at TechnoVerse.

ALL THE BEST