

Assignment 1

About Listed

We are a lab building products for the creator economy. Why so, because we are on a mission to empower the individual to realise/unleash their full potential. Following are some of our flagship products:

OpeninApp

Almost all your favourite creators on Instagram share links to their latest YouTube videos or YouTube channels in their bio or stories. But whenever you tap on those links, you get redirected to the YouTube website instead of the YouTube app. And here you can't watch the video in full screen nor can you like it, comment on it or subscribe to the channel.

Ever wondered why it is so? That's because Instagram has something called a Built-IN-Browser that opens up instead of opening the app you wanted to open. Which is not cool at all....And this is not just so for IG but for all your favourite social media apps eg. YouTube, LinkedIn, Twitter, Snapchat, Telegram, Quora etc.

So we fixed this problem of sub-optimal user experiences by building a tool that ensures all social media links open in the apps they should. It's called OpeninApp. And it is already loved by over 20,000 influencers & brands including the likes of Munawar Faruqui, Raj Shamani, Riva Arora, Aishwarya (Ranjanikaant-j's daughter), Trakin Tech, Mamaearth etc. And has been used by 55Mn+ users.

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](https://www.twilio.com) 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.

- After the challenge is uploaded on GitHub record the demonstration of the completed challenge, and share the video and GitHub link by following the steps mentioned below for the review:

1. Kindly download a screen recorder of your choice
2. Record a video demonstrating the functioning of the program including a code walkthrough
3. The video should be ideally 5 to 6 minutes long
4. Submit video via google drive link with public access
5. Reply back to the same assignment email with the drive link of the video demonstration as well as the GitHub link for the code

By When can you complete the task:

24 to 48 hours - this is an urgent opening and if you have received the assignment you are already ahead of many candidates, hence the faster you submit, the faster we can move ahead with your application.

Feel free to write, if you have any doubts - careers@openinapp.freshteam.com