



TECHNICAL TEST

Mini task

Hi 🙌😊

Here you will find the mini-task.

If you have any questions, please contact Hr ConceptRecall
(hr@conceptrecall.com)

Assessment Task 1:

Develop a standalone application using React/Next, Nodejs and Typescript.

- Please upload the code into a public GitHub repository and share it with us. Include the time needed for all Tasks in the README

Objective:

Design and develop a full-stack application using React/Next, Typescript, and a suitable backend technology (e.g., Node.js, Express). An application is capable of creating tasks accompanied by a specific checklist. Every task should come with its individual checklist items that

can be checked off. While the main emphasis isn't on the user interface, ensure the app's functionality along with the user authentication

Sample Checklists:

- **Checklist 1:**
 - I have
 - created
 - something big!
- **Checklist 2:**
 - Hello
 - I am
 - Ready!

Key Requirements:

- latest React/Next is used
- Any State management library should be used
- Node js for backend, i.e user authentication

Frontend Requirements:

1. User can login to system with their username and password
2. User can view all tasks
3. Tasks should be associated with a selected checklist.
4. Display created tasks in a list along with their respective checklist items.
5. Users should be able to check off items in the checklist.
6. There can be multiple users using this application
7. If the user doesn't exist then create new user from user module
8. The data from the users are separated so that user A can't access user's B Checklist
9. Basic Routing is required.

10. Basic Design approach

Task Name

Ticket progress is blocked

Checklist

CI

Light Bulb 150S

7 STEPS

Check item 1

Not started

Electrical connection, general, 3-pin. Electrical

Blocked :Part installation done

Check item 3

Final installation done

L3.1 LED surface-mounted wall light Architech

L3.1 LED surface-mounted wall light

Electrical connection, general, 3-pin.

Done: Part installation done

Check item 3

Final installation done

L3.1 LED surface-mounted wall light

L3.1 LED surface-mounted wall light

ADD NEW ITEM

Backend Requirements:

- Develop a backend using a suitable technology (e.g., Node.js, Express).
- Implement user authentication: users can log in with a username and password.
- Secure all routes: only authenticated users can access their tasks and checklists.
- Store user and task data in a database (e.g., MongoDB, PostgreSQL).
- Integrate appropriate security measures (e.g., password hashing, token-based authentication).
- Ensure user data is isolated: users should only have access to their own tasks and checklists.

Feedback/Questions

Add answers to these question in README

Considering you have an intricate backend with numerous functionalities and a frontend that currently interfaces with the backend, please address the following points:

1. Drawing from your own experience with functionalities, what challenges have you encountered when implementing or maintaining modes?
2. How did you overcome these challenges, or what solutions did you implement?
3. Are there any particular tools, libraries, or practices you've found especially helpful or problematic in the context?

