



Vertex UI Technical Take Home Assignment

Background

For this technical exercise, we ask you to timebox yourself to no more than **two (2) hours of your time**, it is valuable to you and we don't want to take away from that. We are not looking for completeness, we are looking for how you think and how you solve problems, We also want to see you explain your solution so make sure you understand what you do submit completely.

Work on these requirements in order. If you don't know how to solve a requirement, skip it instead of taking time to learn a new skill specifically for this exercise.

General Instructions

- Create a public GitHub Repository for this exercise
- Do NOT mention Vertex anywhere in the repository, the code, the README.md, anywhere
- Ensure all code is on main, but if you develop with branches do not delete them, again we like to see how you think and work so those branches give us valuable insight. Do not squash merge your commits to main, do a simple merge. Working on a single branch is fine, just commit often.
- When you are ready to submit, send an email to your recruiter with the link to your GitHub repository
- Provide proper instructions in your README.md of how to setup prerequisites, build your code, and ultimate run and test your code

General Thoughts

- Use documentation to your advantage not for us
- Show off your skills. Do not try and match a language or skill you perceive, show us what you know
- Don't let deficiencies stop you, we all have them. Our growth is based on identifying our own deficiencies and improving on them, not hiding them or running from them

Requirements

The following requirements should be executed in order based on the above instructions and background information.

Step 1: Create a CRUD web application

You will need to build a Web API that provides basic CRUD to manage your notes. The note has the following fields:

Field	Type
Id	UUID
Title	String
Content	String
CreatedTime	Date

we want you to use react (next.js is also fine) + your favorite ui library to implement this web application which has the following features

1. create a page to list all notes you created (you only need to show Title and Created Time in this list)
2. create a popup modal to add new note
3. create a page to show details of your notes (you will show Title, CreatedTime and Content on this page)
 1. you can edit your note and save your note here
 2. you have a delete button to delete note
4. you are free to use localStorage to store your data or you can also use a mock backend to store the state of data
5. you need to use frontend router to navigate between pages

Step 2: Add form checking

we want you to add form checking to create/edit note page:

1. for fields
 1. title can't be larger than 50
 2. content can't be larger than 200
2. error message should be shown in a user friendly way

Step 3: Add responsive design

Your app can be seen well on both mobile device and desktop. this is a open questions, just add the responsive design you think is reasonable, you are free to leverage framework to help on responsive design part, some hint could be

1. give some padding on desktop version so notes didn't span too wide while on mobile padding could be small

Step 4: Protect your Route

we want to add protection on route so only after user login, they can access create/edit notes route

1. you will implement a simple user login page to allow user login with username + passowrd
2. for login, it can be a mock one, you don't need to call a real backend api for this
3. when user try to access create/edit notes route without login, proper error message shown.

Step 5: Add Auto Testing

use the testing framework you are familiar with to create following auto test cases:

1. test case 1: create new note
2. test case 2: delete an existing note

Step 6: Add CI/CD

Create a simple github action to test, build and deploy your application

1. you need to run your Auto Testing created in Step 5
2. you need to build your application which is ready to deploy
3. [optional] you can choose to deploy your web application to a remote env (e.g. vercel, netlify, heroku, s3, your own server). this step is optional, it's a bonus to have

As an Equal Opportunity Employer, Vertex Provides reasonable accommodations for applicants with disabilities. If you are an individual with a disability and would like to request a reasonable accomodation at any time during the employment selection process, including pre-employment testing, please contact 610-640-4200 or AskHR@vertexinc.com