Frontend Task Assessment

//TO DO generate the access token for the candidate and include in the mail

In this task assessment, you must develop a React.js application UI that allows CRUD operations on user profiles based on a Figma design. The application should allow the user to switch between themes (dark and light) and include pagination, sorting, and searching (with debounce) functionality. You will be evaluated on your ability to develop a scalable and maintainable React.js application that meets the requirements.

Task Details:
You are required to develop a React.js application UI that performs the following tasks:

Display a list of profiles: Display the list of user profiles(Card View and Grid format) that matches the Figma design. On page load Card View of profiles should be the first page to display and have 16 items fetched from the API. Note: Card view should be shrinkable(Video has been attached for reference.)

Pagination: Implement pagination functionality that displays a limited number of profiles per page. The user should be able to navigate to the next or previous page. The card view should be an infinite scroll, and pagination should work on a scroll.

Switch between themes: Implement functionality that allows the user to switch between themes (dark and light). The theme should be applied to the entire application.

Sorting: Implement sorting functionality that allows the user to sort the list of profiles by email or is_verified. The user should be able to sort the list in ascending or descending order.

Searching: Implement searching functionality that allows the user to search for a profile by name. The search should use debounce functionality(Note: should have 500 ms delay once user is done typing) to prevent multiple API requests .

CRUD Operations: Implement Create, Read, Update, and Delete operations on user profile based on the Figma design. The user should be able to add new profiles, edit an existing profile, and delete an existing profile. Proper error handling should be implemented for all CRUD operations, with appropriate error messages displayed to the user.

Responsive design: Application should be mobile compatible and include responsive design for various screen sizes as shown in the figma design.

Code quality: Your code should follow best practices for React.js development, including proper use of state and props, functional and reusable components, and clean and readable code.

Evaluation Criteria:
Your task assessment will be evaluated on the following criteria:

1. Compliance with the given requirements and Figma design, including responsive design and error handling.
2. Code quality and adherence to best practices for React.js development.
3. User interface design and usability, including responsive design.
4. Ability to implement CRUD operations on user profiles.

Figma -
https://www.figma.com/file/14NNJKOCQbWXkNFxEfbOKo/Front-End-Challenge?node-id=0%3A1&t=vgK2vzlfWvBHRyiq-1

Video Link -
https://drive.google.com/file/d/1eFprXaQugXsvBZn9WoR3pXt_G7iDeshY/view?usp=share_link

Postman collection for the API'S -
https://drive.google.com/file/d/1dCCZdDgnQJ3y5FMfN3CAEXjCNdQDggdR/view?usp=share_link
//Examples, input and expected output has been added for reference.

Access-token-