

Problems:

- The biggest issue will be the synchronization issue. One page of the application won't know about other pages. For example, If a user changes his profile information, this should be reflected on other pages too, or if a user adds a product to a cart, some other pages that are related to products, must know about this. But in our approach, one page will not have context about the pages. There will be state management chaos.
- Another issue will be in the UI. Different pages may produce different themes. For example, logo, button or any other elements that are shared between different pages, should be the same in terms of styling, but in our approach, the same button or logo may contain inconsistent styles or themes across different pages. Layouts may differ in different pages. For example, the header or sidebar section should be the same for all pages, in our approach, these layouts may be inconsistent.
- Configuring the project in this approach is another problem. Authentication, protected routes, public routes, routes linking etc. Must be configured after all the pages have been defined and requirements are known, in our approach, this kind of configuration will be an issue without the whole project context.
- Fetching is an issue here. If user authentication is implemented, there must be a global authentication handler, token must be accessible across all pages. Other pages, would not be about user or token.