Task: Data Fetching with SSR and SSG in Next.js

Task: Create a Next.js application with Server-Side Rendering (SSR), Static Site Generation (SSG), search functionality, and pagination using the reqres.in API.

Objective: Build a Next.js application that fetches user data from the reqres.in API, uses SSG for paginated user pages, SSR for the search page, and includes a search functionality with client-side routing.

Steps:

1. Set up a new Next.js project and install the required dependencies by running the following commands:

npx create-next-app your-app-name

cd your-app-name

npm install axios next-router

1. Create a **components** folder and a **UserCard.js** file inside it. Implement the **UserCard** component that displays the user's avatar, first name, and last name.
2. Update the **pages/index.js** file to fetch the first page of users from the reqres.in API using **getServerSideProps**. Display the users using the **UserCard** component and include pagination links for navigating to other pages.
3. Add a search form to the home page that includes an input field and a submit button. Implement the search functionality using client-side routing with the **next-router** library to navigate to the **/search** route with the search query as a parameter.
4. Create a dynamic SSG page inside the **pages/page** folder with the file name **[page].js**. Implement **getStaticProps** and **getStaticPaths** functions to fetch user data for each page and generate static pages for all available pages using the total\_pages information from the reqres.in API.
5. Add the search form to the paginated user pages with the same functionality as in the home page.
6. Create a new file **pages/search.js** to handle search results. Use the **getServerSideProps** function to fetch the list of users from the reqres.in API and filter them based on the search query.
7. Test the application to ensure that the search functionality works correctly, the paginated user pages are generated statically, and the data is fetched from the reqres.in API.

By following these steps, you will create a Next.js application that combines SSG, SSR, search functionality, and pagination while fetching user data from the reqres.in API. The application will display paginated user pages and allow users to search for other users by their names.