



Technical Test: React Developer with Next.js and Tailwind CSS

Task 1: Authentication with Next.js

- Secure routes using NextAuth.js for authentication.
- Evaluate design patterns in the implementation (Explain which ones you have used and why).
- Create your own domain model.

Task 2: Component development

Animated UI Components Create a UI Kit for the project.

- Implement a theme system with multiple theme options.
- Integrate Tailwind CSS JIT for optimizing the styling workflow and reducing the final bundle size.

Custom Hook

- Create a custom hook for handling complex state management scenarios (e.g., data fetching, caching, and real-time updates).
- Use the hook in at least two different components to highlight its versatility.

Task 3: Dashboard Generation

- Develop a dashboard to show a summary of information.
- This dashboard should include a graph, you can use libraries such as D3.js, Chart.js, etc.

Task 4: Large data set handling

- Display a set of no less than 1000 records on the screen.
- Implements an optimal way of displaying information to the user.
- Implement server-side rendering (SSR) for critical pages to improve the initial loading time.
- Ensure efficient resource loading using strategies like pre-fetching or pre-loading.

Task 5: Performance Optimisation and CSS

- Optimize the application for mobile performance using Google Lighthouse metrics.
- Implement server-side rendering (SSR) for critical pages to improve the initial loading time.
- Ensure efficient resource loading using strategies like pre-fetching or pre-loading.

Task 6: Testing

- Performs unit testing of the application with cypress.
- Performs E2E testing of the application, with jest or with React Testing Library.

Task 7: Documentation

- Produces a detailed documentation of the generated application.

Submission Guidelines:

Share the link to the Git repository containing your code.

Provide a detailed README with instructions, architecture overview, and any additional information.

Ensure that your application is deployed, and share the deployment link (e.g., Netlify, Vercel).