Estas son preguntas adicionales que nos gustaría saber sobre tu experiencia con este stack.

1. ¿Cuál es tu nivel de experiencia en el desarrollo de aplicaciones utilizando React? ¿Has trabajado en proyectos de producción utilizando React? Por favor, proporciona algunos ejemplos.
2. ¿Cuál es el propósito y el uso principal de Redux en una aplicación React? ¿Has utilizado Redux en proyectos anteriores? ¿Cómo gestionas el estado en tus aplicaciones?
3. ¿Estás familiarizado(a) con TypeScript y has trabajado con él en proyectos anteriores? ¿Cuáles son las ventajas que encuentras al utilizar TypeScript en comparación con JavaScript?
4. ¿Tienes experiencia en el desarrollo de sitios web estáticos utilizando Gatsby? ¿Cuál es tu opinión sobre las ventajas y desventajas de utilizar Gatsby en comparación con otros frameworks de React?
5. ¿Has utilizado GraphQL para la comunicación entre el cliente y el servidor? ¿Cuáles son las ventajas de utilizar GraphQL en comparación con las API REST tradicionales? ¿Qué herramientas has utilizado para trabajar con GraphQL?
6. ¿Has trabajado con Tailwind CSS en proyectos anteriores? ¿Cuál es tu opinión sobre su enfoque utility-first en comparación con otros enfoques de CSS, como los frameworks de CSS más tradicionales?
7. ¿Has colaborado con equipos de desarrollo utilizando herramientas de control de versiones, como Git? ¿Puedes describir cómo te has organizado y colaborado eficientemente en un proyecto de equipo?
8. ¿Has tenido que enfrentarte a desafíos de rendimiento en aplicaciones front-end? ¿Qué estrategias o técnicas has utilizado para optimizar el rendimiento de una aplicación web?
9. ¿Has tenido la oportunidad de utilizar este stack tecnológico en conjunto en un solo proyecto? Si es así, ¿puedes proporcionar detalles sobre el proyecto y las tecnologías específicas que utilizaste?