

Descripción detallada de las tecnologías y componentes propuestos:

1. Frontend: React/Vue para el desarrollo de la aplicación web, ya que son frameworks modernos y eficientes para crear aplicaciones escalables y de alto rendimiento.
2. Backend: Node.js con Express para crear una API RESTful que permita la comunicación entre el frontend y la base de datos.
3. Base de datos: MongoDB/DynamoDB, una base de datos NoSQL para almacenar información de usuarios, paquetes y notificaciones.
4. Servidor de autenticación: Utilización de JWT (JSON Web Tokens) para la autenticación y autorización de usuarios y empleados.
5. Mensajería: Integración de servicios como Twilio para enviar SMS y SendGrid o Amazon SES para enviar correos electrónicos.
6. CDN (Content Delivery Network): Cloudflare/Amazon CloudFront para mejorar la velocidad de carga y la disponibilidad de la aplicación.
7. Plataforma en la nube: AWS (Amazon Web Services) para hospedar y escalar la aplicación web y sus componentes.

Razones para elegir dichas tecnologías y componentes:

1. Las tecnologías seleccionadas son populares, bien documentadas y cuentan con amplio soporte en la comunidad de desarrolladores.
2. Las tecnologías propuestas permiten crear aplicaciones escalables y de alto rendimiento que pueden manejar un crecimiento en el número de usuarios y paquetes rastreados.
3. La arquitectura en la nube de AWS ofrece alta disponibilidad y resiliencia, lo que garantiza un tiempo de actividad óptimo y la capacidad de recuperarse rápidamente de posibles problemas.

Explicación de cómo se abordan los desafíos técnicos y cómo se garantizan los requisitos no funcionales:

1. Resiliencia: La utilización de servicios en la nube de AWS proporciona redundancia y copias de seguridad automáticas, lo que garantiza la recuperación rápida de la aplicación en caso de fallos.
2. Escalabilidad: La arquitectura propuesta utiliza servicios en la nube de AWS, lo que permite aumentar o disminuir los recursos fácilmente en función de las necesidades. Además, el uso de un CDN garantiza una distribución rápida y eficiente del contenido a nivel mundial.
3. Seguridad: La autenticación y autorización basadas en JWT garantizan la protección de los datos de los usuarios y los paquetes. Además, el uso de HTTPS y la integración de un CDN como Cloudflare/Amazon CloudFront proporciona protección adicional contra ataques DDoS y otros riesgos de seguridad.
4. Rendimiento: La aplicación web utiliza frameworks de frontend y backend eficientes y optimizados para proporcionar una experiencia de usuario rápida y fluida. Además, el uso de un CDN garantiza tiempos de carga rápidos para los usuarios en todo el mundo.