**Instalación y configuración de servicios de mensajería instantánea (OpenFire, XMPP, Strophe.Js, NODEJS).**

**Autores: Adrián Lecrevisse y David Martín Huerta**

1. **Abstract .**

La comunicación digital se basa en la mensajería instantánea, favoreciendo la interacción en tiempo real entre los usuarios. Este proyecto pone en marcha un sistema de mensajería que utiliza Node.js, Express.js, SQLite, Socket.io, React y Vite, ofreciendo una solución actualizada y eficaz para la transmisión de mensajes.

El procesamiento de datos, la comunicación con la base de datos y otros servicios se realiza con Node.js y Express.js. El almacenamiento de mensajes en SQLite y la gestión de conexiones en tiempo real mediante Socket.io

La interfaz de usuario está desarrollada con React y Vite. La comunicación en tiempo real con el procesamiento de datos, la comunicación con la base de datos y otros servicios se realiza mediante Socket.io, asegurando la sincronización de los mensajes.

La aplicación garantiza una funcionalidad fácil, ideal para aplicaciones de chat en tiempo real.

1. **Introducción.**

Hoy en día, la mensajería instantánea se ha convertido en una herramienta fundamental para la comunicación de los seres humanos, permitiéndonos intercambiar información en tiempo real. En el día a día usamos aplicaciones como WhatsApp, Instagram, Twitter.. Las cuales utilizan este tipo de mensajería.

Este proyecto tiene como objetivo el desarrollo de una aplicación de mensajería en tiempo real, utilizando herramientas para garantizar una comunicación efectiva entre usuarios. Para ello, hemos construido una arquitectura basada en los siguientes componentes:

La parte de la interfaz gráfica, con la que interactúan los usuarios (Frontend), lo hemos construido con React y Vite, proporcionando una interfaz de usuario simple para una buena experiencia en la comunicación.

La parte que controla la lógica de negocio, procesa las solicitudes de los clientes y los mensajes, en definitiva la parte que se encarga de que la lógica de la aplicación funcione (Backend), está desarrollada con Node.js y Express.js.

Para almacenar los mensajes e información de los usuarios hemos utilizado SQLite, una base de datos que está integrada en todos los teléfonos móviles y en la mayoría de computadoras.

La comunicación en tiempo real es implementada mediante Socket.io. Es una biblioteca que permite la transmisión de datos en tiempo real entre clientes y servidor.

Esta aplicación permite a los usuarios enviar y recibir mensajes instantáneamente, manteniendo una conexión estable y segura. Además, la herramienta Vite que es utilizada para el desarrollo del Frontend, permite una carga rápida y Socket.io permite una comunicación en ambos sentidos sin necesidad de actualizar la aplicación.