



UTT

UNIVERSIDAD TECNOLÓGICA DE TIJUANA

GOBIERNO DE BAJA CALIFORNIA

TEMA:

Framework Selection

PRESENTADO POR:

Valenzuela Esparza Javier Ivan

GRUPO:

10B

MATERIA:

Desarrollo Movil Integral

PROFESOR:

Ray Brunett Parra Galaviz

FECHA:

10/01/2025

WHY CHOOSE REACT NATIVE FOR MOBILE DEVELOPMENT AND MOLECULER.JS FOR MICROSERVICES?

Why Choose React Native for Mobile Development and Moleculer.js for Microservices?

For a robust and scalable application architecture, the combination of React Native for mobile development and Moleculer.js for backend microservices is a powerful choice. Below, I explain why this stack is ideal, highlighting their key features, advantages, and how they complement each other.

React Native for Mobile Development

1. Cross-Platform Development

React Native allows developers to build mobile apps for both iOS and Android using a single codebase. This saves time, effort, and costs compared to native development. Its native-like performance ensures smooth animations and a seamless user experience.

2. Strong Ecosystem and Community

Backed by Meta (Facebook) and supported by a large developer community, React Native benefits from regular updates, third-party libraries, and extensive documentation. Tools like Expo make it easier to manage builds and deploy apps.

3. Hot Reloading for Faster Iterations

React Native's hot reloading feature accelerates the development process by allowing developers to see changes in real time, reducing debugging time and enhancing productivity.

4. Used by Leading Companies

React Native powers applications for companies like Facebook, Airbnb, and Uber Eats, showcasing its reliability for high-performance, scalable applications.

Moleculer.js for Microservices

- **Flexible Microservices Framework**

Moleculer.js is a modern and fast framework for building microservices in Node.js. It simplifies the creation of scalable and maintainable distributed systems by providing built-in tools for service communication, load balancing, fault tolerance, and caching.

- **Ease of Service Communication**

Moleculer.js supports multiple communication protocols, including **HTTP**, **WebSockets**, and **NATS**, enabling seamless integration between services. This makes it easy to build APIs, handle events, and connect various parts of your backend system.

- **Built-In Features**

Moleculer.js comes with built-in support for features like service discovery, caching, and metrics. This reduces the need for additional dependencies and makes the framework lightweight yet powerful.

- **Scalability and Fault Tolerance**

Moleculer.js ensures high availability with service replication and automatic failover. It supports horizontal scaling, which is essential for handling increasing workloads in a distributed system.

- **Community and Support**

Moleculer.js has an active community and well-maintained documentation, making it easy for developers to adopt and implement in production systems.

Bibliographic Sources.

React Native · Learn once, write anywhere. (s/f). Reactnative.dev. Recuperado el 11 de enero de 2025, de <https://reactnative.dev/>

Moleculer - Progressive microservices framework for Node.js. (s/f). Moleculer - Progressive Microservices Framework for Node.js. Recuperado el 11 de enero de 2025, de <https://moleculer.services/>