COMUNICACIÓN ENTRE BACKEND Y FRONTEND

CLAUDIA LORENA DIAZ GOMEZ

**Investigar sobre la comunicación entre BackEnd y FrontEnd:**

Lee artículos o tutoriales que expliquen cómo funciona la comunicación entre el BackEnd y el FrontEnd en una aplicación web. Busca ejemplos de implementación de esta comunicación utilizando diferentes tecnologías y protocolos, como REST, GraphQL o WebSockets.

Conceptos Básicos

**FrontEnd:** La parte de una aplicación con la que interactúan los usuarios. Está desarrollada utilizando tecnologías como HTML, CSS y JavaScript.

**BackEnd:** La parte de la aplicación que maneja la lógica del negocio, el almacenamiento de datos y otras operaciones del servidor. Se desarrolla con tecnologías como Node.js, Python (Django, Flask), Ruby on Rails, Java (Spring), etc.

API (Application Programming Interface): Un conjunto de reglas que permite que el frontend y el backend se comuniquen. Las APIs pueden usar diferentes protocolos y tecnologías para esta comunicación.

Tecnologías y Protocolos

1. REST (Representational State Transfer)

REST es un estilo de arquitectura para diseñar servicios web. Utiliza HTTP para realizar operaciones CRUD (Create, Read, Update, Delete).

HTTP Methods:

GET: Obtener datos del servidor.

POST: Enviar datos al servidor.

PUT: Actualizar datos en el servidor.

DELETE: Eliminar datos del servidor.

Ejemplo de Implementación:

Backend (Node.js con Express):



Captura de pantalla de un celular

Descripción generada automáticamente

**2. GraphQL**

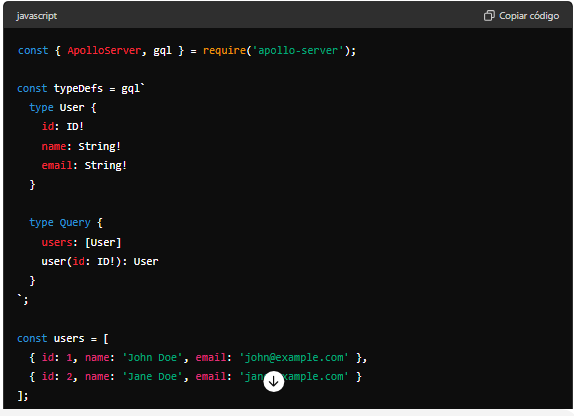
GraphQL es un lenguaje de consulta para APIs que permite a los clientes solicitar exactamente los datos que necesitan.

Forma, Rectángulo

Descripción generada automáticamente

* **Ejemplo de Implementación:**

**Backend (Node.js con Apollo Server):**



Captura de pantalla de computadora

Descripción generada automáticamente

Frontend (JavaScript con Apollo Client):

Captura de pantalla de un celular

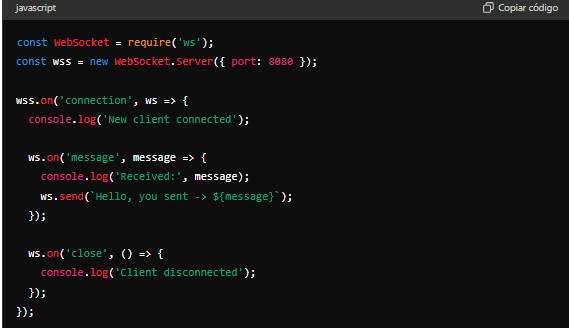
Descripción generada automáticamente

**3. WebSockets**

WebSockets permiten una comunicación bidireccional en tiempo real entre el cliente y el servidor.

* Ejemplo de Implementación:

Backend (Node.js con ws):



Frontend (JavaScript con WebSocket API):

Captura de pantalla de computadora

Descripción generada automáticamente

**Recursos Adicionales**

Documentación REST:

\*[REST API Tutorial](https://restfulapi.net/)

\*[Mozilla Developer Network (MDN) REST](https://developer.mozilla.org/en-US/docs/Web/Guide/REST/Introduction)

\*Documentación GraphQL:

\*[GraphQL Official Website](https://graphql.org/" \t "_new)

\*Apollo GraphQL Documentation

\*Documentación WebSockets:

\*[MDN WebSockets](https://developer.mozilla.org/en-US/docs/Web/API/WebSockets_API)

\*[WebSockets Tutorial](https://www.websocket.org/" \t "_new)

Estas tecnologías y ejemplos proporcionan una base sólida para entender cómo se comunica el frontend con el backend en una aplicación web moderna.

4o