**Parte 4: Preguntas Teóricas (Breves)**

**Objetivo:**

Responde las siguientes preguntas para evaluar tus conocimientos teóricos en programación y arquitectura.

1. **Spring Boot**: ¿Cómo manejarías la inyección de dependencias en Spring Boot?

**R**: Bueno la inyección de dependencias se maneja con la anotación @Autowired que es la forma más común en spring y se usa cuando queremos inyectar una clase, o un constructor y los métodos de una clase.

1. **Hibernate**: Explica brevemente cómo funcionan las relaciones entre entidades en Hibernate. ¿Cómo se maneja una relación One-to-Many?

**R**: Las relaciones entre entidades se modelan utilizando anotaciones específicas que definen cómo se relacionan las tablas de la base de datos. Una relación uno a muchos (One-to-Many) es como su nombre lo dice una entidad que está relacionada con muchas instancias de otra entidad y generalmente se usa la anotación @OneToMany además de usarse mappedBy para especificar la propiedad en la otra entidad que gestiona la relación.

1. **Angular**: ¿Cómo se maneja la comunicación entre componentes padre e hijo en Angular?

**R**: La comunicación entre esos componentes se usa @Input() para enviar los datos del padre al hijo y @Output()y EventEmitter para los datos del hijo al padre. Estos dos mecanismos permiten pasar información entre componentes, siguiendo un patrón de comunicación unidireccional, es decir, siempre que el flujo de datos sea desde el padre al hijo o viceversa.

1. **Node.js**: ¿Cuál es la principal ventaja de usar **Node.js** en el desarrollo backend? ¿Cómo manejas peticiones asíncronas?

**R**: La ventaja es que permite ejecutar código JavaScript en el servidor. Su uso en el desarrollo backend tiene varias ventajas clave, siendo las más destacadas es alta escalabilidad y rendimiento, el uso de hilos para manejo de múltiples solicitudes, gran ecosistemas de paquetes, es ideal para aplicaciones en tiempo real.

Y el manejo de peticiones asíncronas en Node.js es una de las características clave que permite a este entorno manejar grandes volúmenes de tráfico de forma eficiente. Entre ellos están Callbacks, Promises, Async/Await.