**JSON (Java Script Object Notation)**

    Es una notación de objetos de objetos de javascript, es un estándar utilizado en librerías y base de datos.

Se pueden introducir atributos y métodos dentro de los json. Además, se pueden meter json dentro de otro json.

**API (Application Programing Interface)**

    Una API es una interface que sirve para conectar aplicaciones con sistemas sin la necesidad de saber que es lo que está pasando en el interior del sistema; pero utilizando todos los recursos que el sistema posee.

    Las apis pueden tener varias arquitecturas:

* REST ( representational state transfer ): es una representación de transferencia de estado. Implica que los datos pueden guardarse en caché, los  estados no se envían en las peticiones y que se pueden definir los datos y permisos que desea enviar.
* Un formato popular  para enviar datos es el XML, pero hoy en día se utiliza el JSON
* Las api pueden ser públicas o privadas. En el caso de las apis privadas, el servidor devuelve un token en formato jwt para que pueda acceder a la api.
* Las apis pueden ser locales(usar vibrador del teléfono o sonidos) o remotas(servicios web con http con protocolo soap o rest).

    Para desarrollar una api rest se necesita comprender varios conceptos.

    En primer lugar, cada recurso consultado tiene un identificador URI. Cuando se le envía una petición al servidor, este envía un código de estado (2xx , 3xx ,4xx, 5xx). Los métodos HTTP se utilizan para interactuar con el servidor (GET, POST, PUT, DELETE).. Las apis pueden devolver la información en diferentes formatos (xml, json, texto plano)

    Para desarrollar APIs es necesario utilizar buenas prácticas:

* HATEOAS es que la api se describe a sí misma, cada recurso y la cantidad de recursos que hay.
* Seguridad, se deben proteger las apis privadas
* Testear, verificar que todo funcione correctamente. Está ligado con la documentación.
* La documentación es necesaria para que las personas puedan utilizar los recursos de las apis correctamente.