
Tecnicatura Superior en Telecomunicaciones

Materia: Arquitectura y Conectividad

Profesor: JORGE E. MORALES

Ciclo lectivo: 2023

Alumnos : Grupo 4

- Santillan Maximo <https://github.com/maxii-sc>
- Carolina NIS: <https://github.com/Mayte2008>
- Fernando Vexenat: <https://github.com/fvexe82>
- Emilio Moyano: <https://github.com/TerraWolf>
- Esteban Carrizo: <https://github.com/estebancarrizo>

1) ¿Que es una Comunicación REST? ¿Para que se usan? Ejemplifique.

La comunicación REST (Representational State Transfer) es un estilo arquitectónico para sistemas distribuidos que permite la comunicación entre diferentes aplicaciones web utilizando el protocolo HTTP.

En la comunicación REST, las aplicaciones web interactúan a través de una API (Interfaz de Programación de Aplicaciones) que utiliza un conjunto de verbos HTTP (GET, POST, PUT, DELETE, entre otros) para enviar y recibir datos en formato JSON (JavaScript Object Notation) o XML (Extensible Markup Language).

En este modelo de comunicación, cada recurso (como un objeto o una colección de objetos) se representa con una URL (Uniform Resource Locator) única y es posible acceder a él mediante el uso de los verbos HTTP mencionados anteriormente. Esto permite a las aplicaciones web solicitar y recibir información específica, actualizarla o eliminarla según sea necesario.

La comunicación REST se utiliza ampliamente en el desarrollo de aplicaciones web y móviles, ya que permite una comunicación eficiente y escalable entre sistemas distribuidos, independientemente del lenguaje de programación utilizado para desarrollarlos.

En resumen, las comunicaciones REST se utilizan para establecer una arquitectura de comunicación sencilla, escalable e interoperable entre sistemas distribuidos a través de Internet, aprovechando el protocolo HTTP y promoviendo la independencia de plataforma y lenguaje.

Supongamos que tienes un servicio web REST que proporciona información sobre usuarios. El endpoint (punto final) del servicio es <https://api.example.com/users>, y puedes realizar una solicitud GET para obtener información de un usuario específico o de todos los usuarios. Para obtener todos los usuarios, realizarías una solicitud GET a <https://api.example.com/users>. La respuesta del servidor sería una lista de usuarios en formato JSON o XML, por ejemplo:

```
[
  {
    "id": 1,
    "name": "John Doe",
    "email": "john.doe@example.com"
  },
  {
    "id": 2,
    "name": "Jane Smith",
    "email": "jane.smith@example.com"
  }
]
```

Si deseas obtener información sobre un usuario en particular, puedes hacer una solicitud GET a `https://api.example.com/users/{id}`, donde `{id}` es el identificador único del usuario. Por ejemplo, para obtener información del usuario con ID 1, realizarías una solicitud GET a `https://api.example.com/users/1`. La respuesta del servidor sería un objeto JSON o XML con los detalles del usuario:

```
{  
  "id": 1,  
  "name": "John Doe",  
  "email": "john.doe@example.com"  
}
```