

TALLER DE DESARROLLO WEB

Taller IV – Node JS

BACKEND

- Toda la información sensible y la bases de datos de nuestro servidor web deben ser manejado por nuestro backend. Desde el backend, se enviará la información solicitada por parte del frontend.
- Existen varios lenguajes y herramientas para crear nuestro backend, algunos ejemplos son Javascript, PHP, .Net
- Nuestro servidor backend debe tener rutas de acceso disponibles para que el frontend pueda utilizarlos cuando se genere una solicitud.

NODE JS

- Ideado como un entorno de ejecución de JavaScript orientado a eventos asíncronos, Node.js está diseñado para crear aplicaciones network escalables.
- Node JS nos será útil para instalar dependencias como Express, Mongoose, Cors, etc, para manejar nuestro servidor web.
- Para inicializar nuestro proyecto utilizaremos el código en consola:
`npm init -y`
- Nuestro archivo principal de Javascript deberá llamarse index.js

REST

- Representational State Transfer (Transferencia de Estado Representacional) es una interfaz de comunicación web que hace uso de protocolos HTTP (HyperText Transfer Protocol o Protocolo de Transferencia de HiperTexto).
- Las peticiones REST más comunes son:
- PUT (editar)
- GET (obtener)
- POST (crear)
- DELETE (eliminar)

EXPRESS

- Express es una dependencia que instalaremos en nuestro proyecto con NODE JS.
- Express hace uso de la interfaz REST para que podamos generar nuestras APIs (Application Program Interface o Interfaz de Programación de Aplicaciones) dado una ruta en la que se pueda acceder.
- Para instalar express en nuestro proyecto utilizaremos el siguiente código:
`npm install express`

ABRIENDO NUESTRO SERVIDOR WEB

- Para abrir nuestro servidor web utilizaremos Express y le designaremos un puerto que NO esté ocupado por nuestro sistema. Los números de puertos existentes se enumeran del 0 al 65535.
- Nuestro servidor debe tener asignado un URI (Uniform Resource Identifier o Identificador de Recursos Uniforme). Un URI consta de: Esquema, Autoridad, Ruta, Consulta y Fragmento. Debido a que haremos uso de protocolos HTTP nuestro Esquema será "http:"
- Nuestro proyecto será ejecutado por un ambiente local, o sea, nuestra computadora, por lo cual la autoridad será "//localhost:<puerto>"
- Las rutas serán definidas por nosotros en nuestro proyecto junto con las consultas.