



MANUAL TÉCNICO

Practica 4: “Desarrollo Web”

“Califica a tu ingeniero”

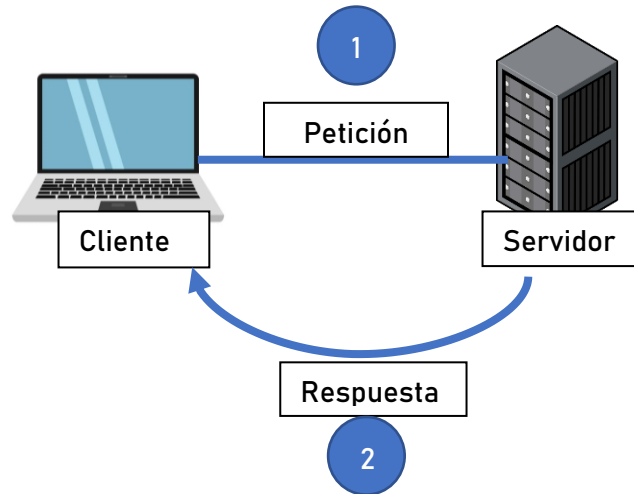
Grupo.	Integrante.	Carnet.
10	José David Panaza Batres	202111478
	Geovanny Sebastián Herrera Claudio	202110588
	Julio Alejandro Zaldaña Ríos	202110206
https://github.com/julizaldana/Practica4_PracticasInicialesC_G10		

Guatemala, 22 de septiembre de 2022

Descripción Aplicación:

Se desarrolló una aplicación web, con base a una arquitectura cliente-servidor que permite a los estudiantes de la carrera de Ingeniería en Ciencias y Sistemas; poder acceder a un récord de catedráticos que imparten los diferentes cursos del área profesional. Para que así, se puedan informar de qué catedrático es el más viable para llevar un curso específico.

Modelo Cliente – Servidor:



Fuente: Elaboración propia, 2022.

La idea es comprender que el servidor es un proveedor que brinda una serie de recursos los cuales serán consumidos por el cliente.

En la práctica se trabajó para:

- La capa del cliente (visualización de aplicación web) fue desarrollada con Angular. (*Frontend*)
- La capa del servidor fue desarrollada con ayuda de Node.js. (*Backend*)
- Toda la información que se maneja en el servidor se almacena en MySQL.

API Rest:

La API Rest, permite y utiliza requisiciones HTTP, responsables de las operaciones básicas necesarias para la manipulación de datos.

Las principales solicitudes de datos son:

- **POST:** Crea datos en el servidor
- **GET:** Lectura y obtención de datos
- **DELETE:** Borra información o datos
- **PUT:** Registro de actualizaciones

Funcionalidades principales:

○ **Servidor:**

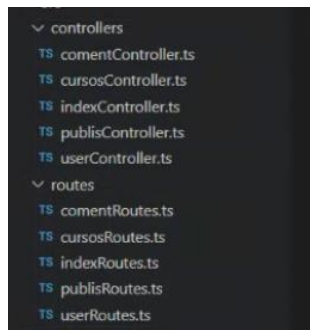
Para poder inicializar el servidor se necesitan distintos módulos que se debieron de instalar, entre ellos están:

- Express
- Morgan
- Promise-mysql
- Cors

Se necesitan distintos controladores y rutas, que se conectan con el servidor para poder manipular y utilizar la base de datos y poder conectar directamente con la API.

Las rutas, sirven como punto de entrada para el usuario en la aplicación, en otras palabras, son la primera capa.

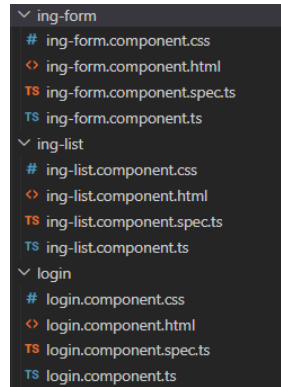
Mientras que los controladores interactúan con los modelos, u otras clases y servicios para generar respuestas que se enviará al usuario, ya sea como llave/JSON.



También serán necesarios de llaves de datos para inicializar las bases de datos.

- **Front-End:**

Para el desarrollo de la interfaz y la visualización de la página web, se usan varios componentes de Angular, utilizando HTML y TypeScript para hacer una conexión con la API, y así poder desplegar bien las pantallas en la web.



Así que, con estos, el usuario podrá interactuar fácilmente con los componentes del backend, que proveerán datos y fundamentos de lógica de la aplicación.

Descripción de herramientas utilizadas:

1. **Angular:** Es una plataforma de desarrollo, construido sobre TypeScript. Es un framework basado en componentes para crear aplicaciones web escalables.
2. **JavaScript:** Es un lenguaje de programación que los desarrolladores utilizan para hacer páginas web interactivas. Ayuda como lenguaje de scripting del lado del servidor.
3. **Node.js:** Es un entorno de tiempo de ejecución de JavaScript. Este entorno de tiempo de ejecución en tiempo real incluye todo lo que se necesita para ejecutar un programa escrito en JavaScript.
4. **TypeScript:** Está construido a un nivel superior de JavaScript (JS), permite que se pueda escribir código de manera mucho más sencilla y fácil de probar. Se puede desarrollar en cualquier tipo de framework.
5. **MySQL:** Es un sistema de gestión de bases de datos relacional de forma open-source.

Glosario:

1. **API:** Las API son mecanismos que permiten a dos componentes de software comunicarse entre sí mediante un conjunto de definiciones y protocolos.
2. **Back-end:** Es la parte de una aplicación, que se encarga de que toda la lógica detrás haga que una página web funcione correctamente.
3. **Open-source:** Es un tipo de software cuyo código fuente y otros recursos que normalmente son exclusivos, son parte del dominio público (libres de uso).
4. **Framework:** Es una estructura o entorno de trabajo que se utiliza para desarrollar aplicaciones.
5. **Front-end:** Es la parte de una aplicación donde se permite a los usuarios acceder, solicitar prestaciones y servicios de forma directa.