

CURSO de DESARROLLO WEB

MODULOS	LARAVEL	3 semanas y media
	NODE.JS CON NESTJS	4 semanas y media
	ANGULAR BASICO - AVANZADO	4 semanas y media
	VUE.JS	3 semanas y media
Requisitos	Conocimientos en Programación orientada a objetos, bases de PHP, bases de HTML, CSS y JavaScript	
Duración	4 MESES martes y jueves	

LARAVEL

Contenido

1.
 - 1.1. ¿Qué es Laravel y como lo puedo utilizar?
 - 1.2. Instalación y configuración de Laravel
 - 1.2.1. Requerimientos mínimos
 - 1.2.2. Instalación vía Composer
 - 1.3. Estructura del Proyecto Laravel
 - 1.3.1. Configuraciones Fundamentales
 - 1.4. Consola Artisan
 - 1.5. Enrutamiento
 - 1.6. Modelado de datos
 - 1.6.1. Query Builders
 - 1.6.2. Migraciones
 - 1.6.3. Seeding
 - 1.6.4. Eloquent ORM
 - 1.6.5. Relaciones entre tablas
 - 1.7. Vistas
 - 1.7.1. Plantillas Blade (Integración con Bootstrap)
 - 1.8. Controladores
 - 1.9. Validaciones (Requests)
 - 1.9.1. Requests
 - 1.9.2. CSRF
 - 2.10. Autenticación y Autorización
 - 2.10.1. Autenticación Scaffolding
 - 2.11. Middleware y su implementación en el enrutamiento seguro
 - 2.12. Políticas de seguridad
 - 2.13. Servicios de Laravel
 - 2.13.1 Servicios o Paquetes de Terceros

NODEJS

con base en TypeScript con PostgreSQL y NestJS

Contenido

1. Bases de TypeScript
 - Instalación
 - El compilador
 - Los tipos en TypeScript
 - Clases e Interfaces
2. NPM
 - Comandos básicos
 - Las dependencias en NPM
 - Package.json
 - Los @types para TypeScript
3. NestJS
 - Instalación y creación de proyecto
 - Manejo de módulos
 - Manejo de controladores
 - Manejo de servicios
4. Instalación del controlador de PostgreSQL para NodeJS y TypeORM
 - Instalación y creación de nuestra BBDD
 - El modelo relacional
 - Entidades
 - Relaciones
5. Servicios
 - Inyección de dependencias (repositorio)
 - Métodos de repositorio
 - Query builders
 - Consultas SQL directas
6. Módulos static y Modulo config
 - Configuración de variables de entorno
 - Archivos públicos con static
7. Manejo de archivos en NestJS
 - Decoradores para el manejo de archivos
 - Validaciones
 - Servicio para el almacenamiento de archivos
8. Manejo de errores
 - Manejo mediante try catch

- Clases para el manejo de errores
- 9. Validaciones
 - Class validator
 - Class transformer
 - Validaciones personalizadas
- 10. Controladores
 - Data Transfer Object
 - Manejo de decoradores
 - Agregar OpenAPI (Swagger)

ANGULAR

Contenido

1. Bases de Angular
 - Conceptos básicos
 - Instalación y creación de un primer proyecto
 - Manejo de Angular CLI
 - Entendiendo nuestro primer proyecto
2. Enrutamiento
 - Definición y manejo de rutas
 - Rutas mediante lazy load
 - Parámetros y parámetros de consulta
 - Guardianes de rutas
3. Componentes Standalone
 - Creación de componentes
 - Partes de un componente
 - Directivas principales
 - Eventos
4. Manejo de Formularios
 - ReactiveFormModule
 - FormBuilder y FormGroup
 - Validación de formularios
 - Validaciones personalizadas
5. Variables de entorno en Angular
 - Creación mediante Angular CLI
 - Creación de diferentes entornos
 - Ejecución y compilación en base a un entorno definido
6. Manejo de servicios
 - Servicios para centralizar información
 - Modulo HTTP
 - Servicios para comunicarse con un servidor REST API
 - Manejo de servicios con los operadores RXJS
 - Subject

- Observer, Observable, subscription
- 7. Manejo de errores
 - Manejo de error al suscribirse
- 8. Manejo de archivos en Angular
 - Manejo de archivos
 - Validaciones para archivos
- 9. Component Lifecycle
 - ngOnInit
 - ngOnChange
 - ngOnDestroy
- 10. Input y Output
 - Modularizar componentes con entradas y salidas
- 11. Manejo de Pipes
 - Conceptos básicos de los Pipes
 - Manejo de Pipes
 - Creación de pipe personalizado
 - Manejo desde un componente
- 12. Interceptores
 - Creación de un interceptor
 - Configuración de interceptores
- 13. Manejo de Guards
 - Conceptos básicos
 - Creación de Guards y tipos de Guards
 - Los Guards y las rutas
- 14. PrimeNG y PrimeFlex
 - Instalación y configuración
 - Diseño con PrimeNG
 - Manejo de la disposición mediante PrimeFlex
 - Manejo de componentes PrimeNG

Curso Vue JS

CONTENIDO

1. Reforzamiento de Javascript
 - 1.1 Bases de Javascript
 - 1.2 Let vs Var vs Const
 - 1.3 Template Literals
 - 1.4 Object Literal
 - 1.5 Arrays
 - 1.6 Arrow functions
 - 1.7 Desestructuración de objetos
 - 1.8 Desestructuración de Arreglos

- 1.9 Importaciones y exportaciones
- 1.10 Promesas
- 1.11 Axios
- 1.12 Async – Await
- 1.13 Ternarios y null check

- 2. Introducción a Vue JS
 - 2.1 Hola Mundo VueJS
 - 2.2 Representacion declarativa
 - 2.3 Estado del componente – Data
 - 2.4 Introducción a los eventos
 - 2.5 Directiva v-for
 - 2.6 Directiva v-model
 - 2.7 Directiva v-bind
 - 2.8 Modificadores de eventos
 - 2.9 Directivas v-if y v-show

- 3. Vue + Vite – Scaffolding Application (Option API y Composition API)
 - 3.1 Inicio de proyecto
 - 3.2 Estructura de directorios generada por defecto
 - 3.3 Mi primero componente (Single File Component)
 - 3.4 Componentes modulares
 - 3.5 Option API y Composition API
 - 3.6 Entendiendo el Option API (data, methods, computed, props)
 - 3.7 Entendiendo composition API
 - 3.8 Lifecycle hooks (en Option API y Composition API)
 - 3.9 **Ref**
 - 3.10 Propiedades computadas – Computed
 - 3.11 Properties – props
 - 3.12 Define props
 - 3.13 Define emit

- 4. Axios en Vue
 - 4.1 Instalar Axios
 - 4.2 Configuraciones generales de Axios
 - 4.3 Consumir servidor REST API con Axios desde componente
 - 4.4 Manejo de errores con try catch
 - 4.5 Interceptores en Axios

- 5. Manejo de validaciones
 - 5.1 Entendiendo las validaciones en Vue
 - 5.2 Vee validate y Yup
 - 5.3 Validaciones personalizadas

6. Vue Router

- 6.1 RouterLink
- 6.2 RouterView
- 6.3 Definición y manejo de rutas
- 6.4 Definición de rutas mediante (Lazy Load)
- 6.5 Parametros y parámetros de consulta
- 6.6 Rutas hijas
- 6.7 Guard – Protección de rutas

7. Pinia

- 7.1 Concepto de Pinia y su uso en Vue
- 7.2 Creación estado centralizado o almacen (store)
- 7.3 Propiedad “state”
- 7.4 Actions
- 7.5 Getters
- 7.6 Su llamada y uso en componentes
- 7.7 Como puedo obtener información de mi “store”
- 7.8 Como puedo establecer información en mi “store” desde el componente
- 7.9 Uso aplicado desde un componente que usa Option API y Composition API