

TEMARIO

Módulo I

Unidad I: Clase 1 (Nivelación)

Lenguajes de Programación (Compilados e Interpretados)

Comandos de Git

Instalación de Docker

Protocolos de Comunicación y Servicios

Navegación Web

Trabajo Práctico de Nivelación

Clase 2 (Introducción a Docker con PHP)

¿Qué es Docker? ¿Para qué me puede servir?

Imágenes en Docker

Contenedores

Docker Compose

Unidad II:

Clase 3 (Introducción a PHP)

¿Qué es PHP? - Historia

Servidores para PHP

Conectores de PHP al servidor

Configuración con Docker

Versionado de PHP

Uso de PHP

Clase 4 (Condicionales y Bucles)

Declaración de condicionales

Declaración de ciclos

Comentarios de código

Práctica de variables, condicionales y ciclos

Unidad III:

Clase 5 (Funciones en PHP)

Semántica de funciones

Palabra reservada function y return

Argumentos en funciones

Documentación de Funciones

Scope de variables en las

funciones Funciones anónimas

Modularización de código

Práctica de funciones usando array callbacks

Clase 6 (Introducción a POO con PHP)

¿Qué es POO? ¿Para qué me puede servir?

Sintaxis de clases en PHP

Unidad IV:

Clase 7 (Herencia y Abstracción)

¿Qué es la Herencia? ¿Qué es la Abstracción?

Práctica sobre herencia simple y herencia múltiple

Clase 8 (Polimorfismo)

¿Qué es el Polimorfismo?

¿Qué son las interfaces?

Clases abstractas

Práctica sobre polimorfismo con vectores polimórficos

Módulo II

Unidas I:

Clase 9 (Namespace, Control de Excepciones y Autoloading)

¿Qué es el namespacing?

¿Cómo se controlan los errores?

¿Qué es el autoloading?

Manejador de Código para PHP - [PEAR, PECL and Composer](#)

Composer

Práctica para hacer uso de autoloading con `use` para las clases o funciones

Clase 10 (Configuración de Proyecto con Composer)

Atributos de composer.json

Agregando dependencias al proyecto

Práctica para manejar peticiones por HTTP realizando testing con script `composer test`

Unidad II:

Clase 11 (Configuración del entorno de desarrollo)

¿Qué es un linter? ¿Para qué me puede servir?

Analizador de código

Test unitarios con PHPUnit

Generador de documentación de código PHP

Clase 12 (Wrappers de PHP)

¿Qué es un wrapper?

Wrapper [php://](#)

Práctica para extraer imagen del usuario usando un formulario

Unidad III:

Clase 13 (Enrutamiento dinámico)

¿Qué es enrutamiento dinámico?

Habilitar enrutamiento dinámico en apache con archivo .htaccess

Introducción a API REST

Práctica para crear enrutamiento dinámico usando peticiones `GET` y `POST`

Unidad IV:

Clase 14 (Introducción a MVC)

¿Qué es MVC?

¿Qué es un ORM? ¿Qué es un ODM? [Diferencias](#)

Introducción a Eloquent ORM

¿Qué son las migraciones?

Clase 15 (Modelos)

Modelo para usuario

Modelo para tareas y su relación

Práctica CRUD para una lista de tareas

Unidad V:

Clase 16 (Clase de Repaso y Consultas)

Inconvenientes con Docker y MySQL

Dudas sobre Eloquent ORM

Clase 17 (Clase de Repaso y Consultas)

Realización de mini proyecto usando CRUD para una lista de tareas

MODULO III

Unidad I:

Clase 18 (Seguridad en Aplicaciones)

- ¿Cómo hacer un registro de usuarios?
- ¿Cómo hacer un login de usuarios?
- ¿Cómo validar credenciales validas?

Clase 19 (Vinculación de Registro, Login y Todos)

- Incorporación de Frameworks de diseño
- Incorporación de Imágenes
- Manejo de cierre de sesión

Unidad II:

Clase 20 (Transformación a API REST)

- Registro en una API REST
- Inicio de sesión en una API REST

Clase 21 (Consumo de API REST)

- Uso de fetch con Javascript
- Peticiones por POST con fetch
- Práctica de consumo de API REST con Javascript Vanilla

Unidad III:

Clase 22 (Despliegue con Heroku)

Configurando Base de Datos

Configurando Aplicación

Realizando despliegue

Clase 23 (Clase de Repaso y Consultas)

Inconvenientes con Heroku

Inconvenientes con PostgreSQL

Dudas sobre API Rest

Dudas sobre Modelos

Dudas sobre Eloquent

Unidad IV:

Clase 24 (Teoría de Servidores)

Protocolo IP

DNS

Clase 25 (Github Actions)

¿Qué es CI/CD?

Integración continua con testing unitarios y de integración

Despliegue continuo con testing e2e

Workflows

Despliegue continuo con Heroku

[Telegram Notify](#) y [Slack Notify](#)

Clase 26 Repaso y entrega Proyecto final

