

Framework PHP Online

Código: FRM-102

Propuesta de Valor: DESARROLLO - PROGRAMACIÓN - METODOLOGÍAS

Duración: 32 Horas



Los frameworks de desarrollo backend, son muy populares en la industria por sus múltiples ventajas para construir proyectos web. Descubre la estructura, patrones y buenas prácticas que nos otorgan estas tecnologías y prepárate para convertirte en una desarrolladora profesional.

AUDIENCIA

- El curso está dirigido a Dirigido a programadores, webmasters, desarrolladores web y en general personas que desean capacitarse a nivel profesional en programación, con conocimientos previos al menos en lenguajes de programación web.

PRE REQUISITOS

- Conocimientos en programación básica y orientada a objetos con PHP.

OBJETIVOS

- Crear tu propio Front Controller capaz de procesar y responder peticiones.
- Conocer el estado del arte de las tecnologías de desarrollo backend.
- Entender la estructura, patrones y conceptos comunes de un framework PHP.
- Entender la complejidad de desarrollar proyectos web complejos.

CERTIFICACIÓN DISPONIBLE

- Certificado emitido por **Cognos**



CONTENIDO

1. INTRODUCCIÓN A PHP

- 1.1. SINTAXIS
- 1.2. OPERADORES
- 1.3. VARIABLES
- 1.4. CONSTANTES
- 1.5. ESTRUCTURAS DE CONTROL
- 1.6. CONSTRUCTORES DE LENGUAJE Y FUNCIONES
- 1.7. EXTENSIONES

2. FUNCIONES

- 2.1. SINTAXIS
- 2.2. PARÁMETROS
- 2.3. VARIABLES
- 2.4. REFERENCIAS
- 2.5. RETORNO
- 2.6. ÁMBITO DE LAS VARIABLES

3. ARRAYS

- 3.1. ARRAYS NUMÉRICOS
- 3.2. ARRAYS ASOCIATIVOS
- 3.3. RECORRER UN ARRAY
- 3.4. ARRAYS MULTIDIMENSIONALES
- 3.5. OTRAS OPERACIONES SOBRE ARRAYS
- 3.6. OBJETOS COMO ARRAYS (USO DE LA SPL)

4. CADENAS Y EXPRESIONES REGULARES

- 4.1. DELIMITADORES DE CADENAS
- 4.2. COMPARACIONES
- 4.3. EXTRACCIONES
- 4.4. BÚSQUEDAS
- 4.5. SUSTITUCIONES
- 4.6. FORMATEO
- 4.7. EXPRESIONES REGULARES COMPATIBLES DE PERL (PCRE)
- 4.8. HEREDOC Y NOWDOC
- 4.9. CODIFICACIONES DE CARACTERES

5. ENTRADA / SALIDA

- 5.1. ARCHIVOS
- 5.2. LECTURA / ESCRITURA / OTRAS FUNCIONES DE ARCHIVOS
- 5.3. CODELGNITER
- 5.4. ESTRUCTURA EN CODEIGNITER 4
- 5.5. CONTROLADOR Y MÉTODO POR DEFECTO
- 5.6. CREAR MÉTODO
- 5.7. TRABAJAR CON PARÁMETROS Y RUTAS
- 5.8. CONEXIÓN BASE DE DATOS
- 5.9. MODELOS
- 5.10. MIGRACIONES
- 5.11. REFRESH DE MIGRACIONES
- 5.12. SEMILLAS
- 5.13. CONTROLADOR DE RECURSOS
- 5.14. AGREGANDO VISTAS
- 5.15. RENDERIZADO DE VISTAS
- 5.16. AGREGANDO BOOTSTRAP 5
- 5.17. FINDALL
- 5.18. ENTIDADES
- 5.19. TABLE
- 5.20. HELPER FORM
- 5.21. INSERT Y ENTIDADES
- 5.22. DELETE
- 5.23. SOFTDELETE
- 5.24. REDIRECT
- 5.25. SHOW
- 5.26. SHOW ESTILOS
- 5.27. EDIT
- 5.28. UPDATE Y ENTIDADES
- 5.29. VALIDACIONES BACK-END 1
- 5.30. VALIDACIONES BACK-END 2
- 5.31. VALIDACIONES BACK-END 3
- 5.32. SEPARAR VALIDACIONES EN VISTA
- 5.33. SECURITY CLASS
- 5.34. TESTING DE REQUEST
- 5.35. CSRF TOKENS Y REGENERACIÓN
- 5.36. NOTIFICACIONES 1
- 5.37. NOTIFICACIONES 2
- 5.38. FILTRAR CON DATATABLES
- 5.39. SET SESSION
- 5.40. LOGIN FORM
- 5.41. DASHBOARD INICIO
- 5.42. LOGOUT DE USUARIO

6. LARAVEL FUNDAMENTALS

- 6.1. INSTALACIÓN DE COMPOSER Y LARAVEL
- 6.2. INTRODUCCIÓN A LARAVEL.
- 6.3. RUTAS
- 6.4. PRUEBAS
- 6.5. CONTROLADORES
- 6.6. VISTAS
- 6.7. PLANTILLAS CON BLAD
- 6.8. LAYOUTS CON BLADE

7. INTRODUCCIÓN A LOS MODELOS DE LARAVEL

- 7.1. CONFIGURACIÓN DE LA BASE DE DATOS DEL PROYECTO
- 7.2. PRESENTAMOS EL ORM ELOCUENTE
- 7.3. CREANDO EL PRIMER MODELO
- 7.4. INTRODUCCIÓN A LAS MIGRACIONES
- 7.5. DEFINICIÓN DE ACCESORES, MUTADORES Y MÉTODOS
- 7.6. VALIDACIÓN DE MODELOS
- 7.7. CREACIÓN DE UN CONTROLADOR RESTFUL
- 7.8. SEMBRANDO LA BASE DE DATOS
- 7.9. BÚSQUEDA DE DATOS
- 7.10. INSERCIÓN DE NUEVOS REGISTROS
- 7.11. ACTUALIZACIÓN DE REGISTROS EXISTENTES
- 7.12. ELIMINACIÓN DE REGISTROS
- 7.13. INTRODUCCIÓN AL GENERADOR DE CONSULTAS
- 7.14. CREACIÓN DE URL SLUGGABLE
- 7.15. MODELOS DE PRUEBA

8. RELACIONES DE MODELOS, ALCANCES Y OTRAS CARACTERÍSTICAS AVANZADAS

- 8.1. INTRODUCCIÓN A LAS RELACIONES
- 8.2. INTRODUCCIÓN A LAS RELACIONES UNO A UNO
- 8.3. INTRODUCCIÓN A LA RELACIÓN PERTENECE A
- 8.4. INTRODUCCIÓN A LAS RELACIONES DE UNO A MUCHOS
- 8.5. INTRODUCCIÓN A LAS RELACIONES DE MUCHOS A MUCHOS
- 8.6. PRESENTANDO TIENE MUCHOS A TRAVÉS DE LAS RELACIONES
- 8.7. INTRODUCCIÓN A LAS RELACIONES POLIMÓRFICAS
- 8.8. CARGA ANSIOSA
- 8.9. INTRODUCCIÓN A LOS ÁMBITOS

9. SYMFONY INTRODUCCIÓN

- 9.1. QUE ES SYMFONY
- 9.2. POR QUÉ SYMFONY
- 9.3. SIMPLICIDAD
- 9.4. ARQUITECTURA
- 9.5. COMPONENTES
- 9.6. INSTALACIÓN Y CONFIGURACIONES
- 9.7. CREAR UN PROYECTO DESDE CERO
- 9.8. ARQUITECTURA MVC
- 9.9. ARCHIVOS DE CONFIGURACIÓN YML
- 9.10. ESTRUCTURA DE DIRECTORIOS
- 9.11. USO Y DE CONTROLADORES, CLASES DE MODELO Y LA VISTA
- 9.12. CLASE CONTROLLER
- 9.13. REQUEST HTTP
- 9.14. PARÁMETROS DEL REQUEST
- 9.15. USO DE ANOTACIONES ROUTE
- 9.16. RUTAS/ROUTES
- 9.17. RESPONSE Y REDIRECT
- 9.18. CONTENEDOR DE SERVICIOS
- 9.18. CONTENEDOR E INYECCIÓN DE DEPENDENCIA
- 9.19. LAYOUT TWIG, EVENT LISTENERS O HOOKS
- 9.20. TWIG LAYOUT
- 9.21. VIEW HELPERS
- 9.21. INTERECPTORES O LISTENERS (HTTP REQUEST)
- 9.22. MANEJO DE EVENTOS (PATRÓN OBSERVADOR OBSERVABLE)
- 9.23. FORMULARIOS Y VALIDACIONES
- 9.24. COMPONENTE FORM
- 9.25. TIPOS DE ELEMENTOS
- 9.26. ETIQUETAS Y HELPERS FORM DE VISTAS
- 9.27. VALIDADORES USANDO ANOTACIONES

10. MODELO (BASE DE DATOS)

- 10.1. CONEXIÓN A LA BASE DE DATOS
- 10.2. CONFIGURACIÓN Y ADAPTADORES
- 10.3. ORM DOCTRINE
- 10.4. USO DE ANOTACIONES
- 10.5. SENTENCIAS DQL CON QUERY BUILDER
- 10.6. MAPEO A TABLAS DE LA BASE DE DATOS
- 10.7. RELACIONES DE OBJETOS Y TABLAS RELACIONADAS
- 10.8. CLASES ENTITY DE PERSISTENCIA
- 10.9. CLASES DE MODELO ENTITY Y DAO
- 10.10. QUERY LOG (REVISAR CONSULTAS GENERADAS)
- 10.11. CRUD: FUNCIONALIDADES DE UN SISTEMA: ABMS
- 10.12. COMPONENTES DE NEGOCIO, SERVICE CONTAINER E INYECCIÓN DE DEPENDENCIAS
- 10.13. EJEMPLOS CRUD CON FORMULARIOS Y TABLAS RELACIONADAS

★ BENEFICIOS

- Los estudiantes que finalicen el curso podrán crear e implementar aplicaciones de forma más rápida y segura. Con los métodos tradicionales, el código se desarrolla en un entorno informático específico que, cuando se transfiere a una nueva ubicación, a menudo genera fallas y errores.