

# LISTADO COMPLETO DE CONTENIDOS - TEMA 0 Y TEMA 1

## TEMA 0: Introducción al Desarrollo Web en Entorno Servidor

### 1. Interfaces Web y Diseño en Aplicaciones

- Concepto de interfaz web como comunicación usuario-servidor
- Tipos de interfaces web:
  - Interfaz web tradicional (HTML/CSS/JS)
  - Interfaz de administración (paneles de control)
  - Interfaz basada en formularios web
  - API REST (JSON/XML)
  - Single Page Applications (SPAs)
  - Progressive Web Apps (PWAs)
  - Interfaces de terminal web

### 2. Tipos de Interfaces de Usuario

- **Interfaz gráfica de usuario (GUI):** Elementos visuales
- **Terminal/Línea de comandos (CLI):** Comandos de texto
  - `php -S localhost:8000`: servidor web local
  - `composer install`: instalar dependencias PHP
  - `npm start`: servidor frontend
  - `php artisan serve`: servidor Laravel

### 3. APIs (Application Programming Interface)

- **Definición:** Conjunto de reglas para comunicación entre aplicaciones web
- **Función:** Intermediario entre aplicaciones (cliente-servidor)
- **Tipos de APIs:**
  - **API de Biblioteca Web:** Funciones predefinidas (ej: Google Maps)
  - **API de Sistema:** Acceso a funcionalidades del SO
  - **API de Servicios Web:** Comunicación online vía HTTP

### 4. API REST vs API SOAP

- **API RESTful:**
  - Operaciones HTTP: GET, POST, PUT, DELETE
  - Formato JSON (más ligero)
  - Simplicidad y escalabilidad
  - Ideal para aplicaciones web/móviles
- **API SOAP:**

- Formato XML estructurado
- Mayor seguridad (WS-Security)
- Transacciones complejas
- Aplicaciones empresariales

## 5. Unidades de Medida en Diseño Web

- **Píxeles (px)**: Unidad básica fija
- **Porcentaje (%)**: Relativo al contenedor
- **em**: Relativo al tamaño de fuente del padre
- **rem**: Relativo al tamaño de fuente raíz (html)
- **vw/vh**: Viewport width/height (1vw = 1% ventana)
- **Densidad de píxeles (dpi)**: Calidad de pantalla
- **Píxeles independientes (dp)**: Para responsive design

## 6. Fórmulas de Conversión CSS

- $rem = px / 16$
- $em = px / tamaño\_fuente\_padre$
- Media queries: `@media (max-width: 768px) { ... }`

## 7. Usabilidad y Experiencia de Usuario (UX)

- **Usabilidad**: "No me hagas pensar"
- **Principios de diseño**:
  - Consistencia visual y funcional
  - Interfaz predecible
  - Menos es más (pantallas móviles)
  - Ayuda contextual
- **Causas de abandono**: Notificaciones excesivas, registro obligatorio, lentitud

## 8. Tipografía y Colores

- **Tipografía web**:
  - Tamaño mínimo: 16px
  - Fuentes recomendadas: Open Sans, Roboto, Lato
  - Contraste alto texto-fondo
- **Sistemas de colores**:
  - **RGB**: Valores 0-255 por canal (8 bits)
  - **Hexadecimal**: Notación compacta (#FF5733)
  - **ARGB/RGBA**: Incluye transparencia (Alpha)

## 9. Cálculos de Color

- **RGB combinaciones**:  $256^3 = 16,777,216$  colores
- **Bits por color**: 24 bits (8 bits × 3 canales)
- **Con transparencia**: 32 bits (ARGB)

## 10. Imágenes y Multimedia

- **Mapa de bits:** Matriz de píxeles (JPEG, PNG)
- **Vectoriales:** Puntos y curvas matemáticas (SVG)
- **Relación de aspecto:** Altura / Anchura
- **Formatos de audio:** WAV (sin pérdida), MP3/OGG (con pérdida)
- **Bitrate audio:** 128-320 kbps (recomendado 192 kbps)

## 11. Iconos y Navegación

- **Iconos Material Design:** Vectoriales gratuitos
- **Buenas prácticas:** No repetir iconos para acciones distintas
- **Tipos de menú:**
  - Estándar: App sigue funcionando
  - Modal: Ocupan toda pantalla

## 12. Accesibilidad

- **Definición:** Usabilidad para todos los usuarios
- **Estadísticas:** 4.5% población daltónica, 4% visión reducida
- **Recomendaciones:**
  - Contraste alto en textos
  - Controles mínimos 48x48 dp
  - Descripciones para elementos
  - Gestos estándar

## 13. Frameworks y Tecnologías

- **PHP Frameworks:**
  - **Symfony:** Robusto, Twig templates, Doctrine ORM
  - **Laravel:** Popular, Eloquent ORM, Blade templates
  - **CodeIgniter:** Ligero y simple
- **CSS Frameworks:**
  - **Bootstrap:** Componentes predefinidos, grid system
  - **Tailwind:** Clases de utilidad, personalización

## 14. Terminal y Comandos

- **Windows (CMD/PowerShell):**
  - `cd carpeta`: cambiar directorio
  - `dir`: listar archivos
  - `copy`: copiar archivos
- **Ubuntu/Linux (Bash):**
  - `ls -la`: listar detallado
  - `sudo`: ejecutar como admin
  - `chmod 755`: permisos ejecución

## 15. Servidores Web y Entornos

- **Apache HTTP Server:**
  - Configuración .htaccess
  - Módulos: mod\_php, mod\_rewrite
- **Nginx:**
  - Mayor rendimiento
  - Configuración centralizada
- **Entornos locales:**
  - **XAMPP:** Apache + MySQL + PHP
  - **Docker:** Contenedores consistentes

## 16. Conceptos de Servidor

- **Puertos:**
    - 80: HTTP
    - 443: HTTPS
    - 3306: MySQL
    - 8000: Desarrollo local
  - **Logs:** access.log, error.log
  - **Gestión servicios:** systemctl start/stop/status
- 

# TEMA 1: Introducción a PHP y Entorno de Desarrollo

## 1. Introducción a PHP

- **Definición:** PHP: Hypertext Preprocessor
- **Características:**
  - Interpretado en el servidor
  - Multiplataforma (Windows, Linux, macOS)
  - Sintaxis sencilla
  - Código abierto
  - Gran comunidad
- **Estadísticas:** 77.4% de sitios web usan PHP

## 2. Instalación y Configuración

- **Editores recomendados:**
  - **Visual Studio Code** (gratuito)
  - **PhpStorm** (profesional)
  - **Sublime Text, Atom, NetBeans**
- **Extensiones VS Code:**
  - PHP Intelephense
  - PHP Debug
  - HTML CSS Support
  - Live Server

### 3. XAMPP - Entorno de Desarrollo

- **Componentes XAMPP:**
  - **X:** Multiplataforma
  - **A:** Apache (servidor web)
  - **M:** MariaDB/MySQL (base de datos)
  - **P:** PHP (lenguaje)
  - **P:** Perl
- **Servicios principales:**
  - Apache: Puerto 80, 443
  - MySQL: Puerto 3306
  - FileZilla: Puerto 21

### 4. Docker para Desarrollo

- **Ventajas Docker vs XAMPP:**
  - Mayor consistencia entre entornos
  - Mejor portabilidad
  - Despliegues reproducibles
- **Docker Compose:** Configuración con servicios web + db
- **Comandos básicos:**
  - `docker-compose up -d`
  - `docker ps`
  - `docker-compose down`

### 5. Estructura de Proyectos Web

- **Directorio htdocs:** `C:\xampp\htdocs\`
- **Estructura recomendada:**

mi\_proyecto/

— index.php	# Página principal
— config/	# Configuración
— css/	# Hojas de estilo
— js/	# JavaScript
— images/	# Imágenes
— includes/	# Archivos reutilizables
— pages/	# Páginas adicionales
— README.md	# Documentación

## 6. Archivos de Configuración

- **config.php**: Configuración general
  - Constantes: `define('APP_NAME', 'Mi Web')`
  - Base de datos: `DB_HOST`, `DB_NAME`, `DB_USER`
  - Modo desarrollo: `error_reporting(E_ALL)`
- **.htaccess**: Configuración Apache
  - Reescritura URLs
  - Seguridad básica
  - Compresión GZIP
- **composer.json**: Gestión dependencias

## 7. Sintaxis Básica PHP

- **Tags PHP**:
  - `<?php ... ?>`: Etiquetas completas
  - `<?= ... ?>`: Echo corto
- **Comentarios**:
  - `//`: Línea única
  - `/* */`: Múltiples líneas
  - `#`: Estilo shell
- **Separación instrucciones**: Punto y coma (;)

## 8. Variables y Tipos de Datos

- **Declaración**: Siempre con `$`
- **Tipos escalares**:
  - `string`: "texto"
  - `int`: 42
  - `float`: 3.14
  - `bool`: true/false
- **Tipos compuestos**:
  - `array`: [1, 2, 3]
  - `object`: Instancias de clases
- **Tipos especiales**:
  - `null`: Ausencia de valor
  - `resource`: Referencias externas

## 9. Inclusión de Archivos

- **include vs require**:
  - `include`: Continúa si archivo no existe
  - `require`: Detiene ejecución si no existe
  - `include_once/require_once`: Incluye solo una vez
- **Uso típico**:
  - `header.php`: Encabezado común

- `footer.php`: Pie de página
- `config.php`: Configuración

## 10. Integración HTML + PHP

- **Métodos de integración:**
  - Sintaxis completa: `<?php echo $variable; ?>`
  - Sintaxis corta: `<?= $variable ?>`
- **Separación lógica/presentación:**
  - Lógica en PHP (variables, cálculos)
  - Presentación en HTML con variables PHP

## 11. CSS Frameworks Integration

- **Bootstrap 5:**
  - CDN para inclusión rápida
  - Grid system responsive
  - Componentes predefinidos
- **Tailwind CSS:**
  - Clases de utilidad
  - Personalización granular
  - Integración con build tools

## 12. Responsive Design Básico

- **Media Queries:**
  - Móviles: `@media (max-width: 576px)`
  - Tablets: `@media (min-width: 768px)`
  - Desktop: `@media (min-width: 992px)`
- **Viewport:** `<meta name="viewport" content="width=device-width">`
- **Flexbox:** Layout flexible y responsive

## 13. Buenas Prácticas Desarrollo

- **Organización código:**
  - Nombres descriptivos
  - Comentarios claros
  - Indentación consistente
  - Encoding UTF-8
- **Versionado:** Git para control de versiones
- **Estructura:** Separar por tipo y funcionalidad

## 14. Testing y Debugging

- **URLs de prueba:**
  - `http://localhost/`: Página principal
  - `http://localhost/phpmyadmin/`: Gestión BD
- **Herramientas navegador:**

- F12: Developer tools
- Console: Errores JavaScript
- Network: Análisis HTTP requests
- **PHP Debug:** `var_dump()`, `print_r()`

## 15. Recursos y Documentación

- **Documentación oficial:** <https://www.php.net/manual/es/>
  - **Comunidades:** Stack Overflow, Reddit r/PHP
  - **Herramientas adicionales:**
    - **Composer:** Gestor de paquetes
    - **Xdebug:** Depurador avanzado
    - **phpMyAdmin:** Administración MySQL
- 

## RESUMEN EJECUTIVO POR CATEGORÍAS

### DESARROLLO EN SERVIDOR

- PHP como lenguaje servidor-side
- XAMPP/Docker para entornos locales
- Apache/Nginx como servidores web
- Estructura de proyectos organizados
- Configuración y archivos de entorno

### TECNOLOGÍAS WEB

- APIs REST/SOAP para comunicación
- Frameworks PHP (Symfony, Laravel)
- CSS Frameworks (Bootstrap, Tailwind)
- Responsive design y media queries
- Integración frontend-backend

### HERRAMIENTAS Y COMANDOS

- Terminal/CLI para desarrollo
- Composer para dependencias
- Git para versionado
- Docker para contenedores
- Debugging y testing tools

### DISEÑO Y UX

- Principios de usabilidad
- Sistemas de colores (RGB, Hex)
- Tipografía web responsive
- Accesibilidad e inclusión
- Optimización de imágenes





## RESPONSIVE Y MÓVIL

- Unidades CSS (px, rem, em, %)
- Densidad de píxeles
- Navegación móvil
- Touch interfaces
- Performance optimization

---

*Este listado cubre todo el contenido evaluable de los Temas 0 y 1, enfocándose en desarrollo del lado del servidor y tecnologías web fundamentales.*