



Aprendizaje sobre el Proyecto Portafolio

Programación

1. Elementos fundamentales del código

- Variables (`let proyecto`), constantes (`const VERSION`)
- Operadores: `+`, `-`, `*`, `/`, `===`
- Tipos: string, number, boolean, array, objeto

```
const VERSION = "1.0";  
let proyecto = "Portafolio";  
let proyectos = ["CRM", "Voltereta", "Portafolio"];  
let activo = true;
```

2. Estructuras de control

- Selección: `if`, `else`
- Repetición: `for`, `forEach`

```
for (let p of proyectos) {  
  if (p === "Portafolio") {  
    console.log("Proyecto actual");  
  }  
}
```

3. Control de excepciones y gestión de errores

- Uso de `try-catch` en scripts JS para manejo de errores

```
try {  
  // Código que puede fallar  
} catch (error) {  
  console.error(error);  
}
```

4. Documentación del código

- Comentarios `//` en JS y CSS
- Archivos markdown para documentación

5. Paradigma aplicado

- Programación estructurada y modular
- Separación de lógica en funciones y archivos

6. Clases y objetos principales

- Objetos: `proyecto`, `usuario`
- No se usan clases, pero sí objetos literales

7. Conceptos avanzados

- Modularidad y reutilización de funciones
- Sin herencia ni polimorfismo

8. Gestión de información y archivos

- No hay backend ni base de datos
- Toda la información se gestiona en el frontend

9. Estructuras de datos utilizadas

- Arrays y objetos

```
let proyectos = [  
  { nombre: "CRM", stack: "Node.js" },  
  { nombre: "Voltereta", stack: "Node.js" },  
];
```

10. Técnicas avanzadas

- Validación de formularios con JS
- Manipulación del DOM

Sistemas Informáticos

1. Características del hardware

- Desarrollo: MacBook, 8GB+ RAM
- Producción: Hosting web estático

2. Sistema operativo

- macOS para desarrollo
- Linux en servidores de hosting

3. Configuración de redes

- Acceso público vía HTTP/HTTPS
- Dominio personalizado

4. Copias de seguridad

- Backups manuales y repositorio Git

5. Integridad y seguridad de datos

- Validación de entradas en formularios
- HTTPS en producción

6. Usuarios, permisos y accesos

- No hay gestión de usuarios en el sistema
- Acceso público

7. Documentación técnica

- Archivos markdown y comentarios en el código

Entornos de Desarrollo

1. Entorno de desarrollo (IDE)

- Visual Studio Code
- Extensiones: HTML, CSS, JavaScript, Live Server

2. Automatización de tareas

- Live Server para recarga automática
- No hay scripts de build

3. Control de versiones

- Git y GitHub
- Ramas para pruebas y producción

4. Refactorización

- Mejoras periódicas de código y estilos

5. Documentación técnica

- Markdown (`README.md`), comentarios

6. Diagramas

- Diagramas de flujo para navegación (opcional)

Bases de Datos

1. Sistema gestor

- No se usa base de datos
- Toda la información es estática o en arrays JS

2. Modelo entidad-relación

- No aplica

3. Funcionalidades avanzadas

- No aplica

4. Protección y recuperación de datos

- Backups en GitHub

Lenguajes de Marcas y Gestión de Información

1. Estructura de HTML

- Etiquetas semánticas (`<header>`, `<main>`, `<footer>`)
- Buenas prácticas para accesibilidad y SEO

```
<header>
  <h1>Portafolio</h1>
</header>
<main>
  <!-- Proyectos -->
</main>
<footer>
  <p>© 2025</p>
```

2. Tecnologías frontend

- HTML, CSS, JavaScript

3. Interacción con el DOM

- JavaScript para mostrar/ocultar secciones y validar formularios

```
document.getElementById("btnVer").addEventListener("click", function() {  
    document.getElementById("proyectos").style.display = "block";  
});
```

4. Validación de HTML y CSS

- Validadores online y extensiones del IDE

5. Conversión de datos (XML, JSON)

- No aplica, pero se podría usar JSON para guardar datos en localStorage

6. Integración con sistemas de gestión

- No aplica

Proyecto Intermodular

1. Objetivo del software

- Mostrar proyectos y habilidades de forma visual y profesional

2. Necesidad o problema que soluciona

- Facilita la presentación y búsqueda de información personal y profesional

3. Stack de tecnologías

- HTML, CSS, JavaScript
- Git y GitHub para control de versiones

4. Desarrollo por versiones

- v1: Estructura básica y presentación de proyectos
- v2: Mejoras visuales, animaciones, formulario de contacto

