

Examen Trimestre I

Programación Web - EPRENDIA

Desarrollo del Sitio Web Eprendia

Informe Técnico de Implementación

Fecha de inicio: Jueves 11-12
Fecha de entrega: Lunes 15-12
Tecnologías: HTML5, CSS3, JavaScript

15 de diciembre de 2025

Índice

| | |
|---|-----------|
| 1. Introducción | 3 |
| 1.1. Objetivo del Examen | 3 |
| 1.2. Consigna del Examen | 3 |
| 1.3. Tecnologías Utilizadas | 3 |
| 2. Estructura del Proyecto | 3 |
| 2.1. Organización de Archivos | 3 |
| 2.2. Páginas Desarrolladas | 4 |
| 2.2.1. index.html - Página Principal | 4 |
| 2.2.2. novedades.html - Novedades | 4 |
| 2.2.3. quienes-somos.html - Información Institucional | 4 |
| 2.2.4. contacto.html - Formulario de Contacto | 4 |
| 3. Implementación de Requisitos | 5 |
| 3.1. Requisito 1: Estructura Básica del Sitio | 5 |
| 3.2. Requisito 2: Elementos Visuales | 5 |
| 3.2.1. Logo de la Empresa | 5 |
| 3.2.2. Barra de Navegación | 5 |
| 3.2.3. Carrusel con Mínimo 3 Slides | 6 |
| 3.2.4. Diseño Responsive | 7 |
| 3.3. Requisito 3: Secciones Específicas | 7 |
| 3.3.1. Sección de Interés en Home | 7 |
| 3.3.2. Footer Genérico | 8 |
| 3.3.3. Formulario de Contacto con Validaciones | 8 |
| 4. Desarrollo del JavaScript | 9 |
| 4.1. Carrusel Automático (carousel.js) | 9 |
| 4.2. Menú Hamburguesa (menu.js) | 10 |
| 4.3. Validación de Formulario (form.js) | 11 |
| 5. Desarrollo del CSS | 12 |
| 5.1. Sistema de Variables | 12 |
| 5.2. Efectos Visuales | 12 |
| 5.2.1. Glassmorphism en Navbar | 12 |
| 5.2.2. Animaciones | 13 |
| 5.3. Diseño Responsive | 13 |
| 5.3.1. Menú Móvil | 13 |
| 6. Desafíos y Soluciones | 14 |
| 6.1. Desafío 1: Carrusel Automático | 14 |
| 6.2. Desafío 2: Validación de Formulario | 14 |
| 6.3. Desafío 3: Diseño Responsive | 14 |

| | |
|--|-----------|
| 7. Resultados y Conclusiones | 14 |
| 7.1. Cumplimiento de Requisitos | 14 |
| 7.2. Funcionalidades Adicionales | 14 |
| 7.3. Aprendizajes | 15 |
| 7.4. Conclusión | 15 |

1 Introducción

1.1 Objetivo del Examen

El presente informe documenta el desarrollo del sitio web **Eprenda** como parte del examen del Trimestre I de Programación Web. El objetivo principal fue replicar la página de e-prenda (<https://eprendaplay.com/home>) aplicando los conocimientos adquiridos durante el trimestre sobre desarrollo web con HTML, CSS y JavaScript.

1.2 Consigna del Examen

Se solicitó crear un sitio web que incluya:

- 4 páginas: Home, Novedades, Quiénes Somos y Contacto
- Barra de navegación responsive
- Carrusel con mínimo 3 slides
- Diseño responsive
- Formulario de contacto con validaciones
- Footer en todas las páginas

1.3 Tecnologías Utilizadas

Para el desarrollo de la pagina se utilizaron exclusivamente tecnologías web fundamentales:

- **HTML5**: Estructura semántica del contenido
- **CSS3**: Diseño visual, animaciones y responsive design
- **JavaScript (Vanilla)**: Interactividad y validaciones

2 Estructura del Proyecto

2.1 Organización de Archivos

El examen se organizó de manera modular para facilitar el mantenimiento:

```
1 EntregaExamen-Rinesi/  
2 |-- index.html           # Pagina principal  
3 |-- novedades.html       # Pagina de novedades  
4 |-- quienes-somos.html   # Pagina institucional  
5 |-- contacto.html        # Pagina de contacto  
6 |-- css/  
7 |   |-- styles.css       # Estilos globales  
8 |-- js/  
9 |   |-- carousel.js      # Logica del carrusel  
10 |   |-- menu.js          # Menu hamburguesa  
11 |   |-- form.js          # Validacion de formulario
```

```
12 |-- images/  
13 |-- logo.png  
14 |-- slide1.png, slide2.png, slide3.png  
15 |-- slide4.png, slide5.png, slide6.png  
16 |-- slide7.png, slide8.png, slide9.png  
17 |-- curso1.png, curso2.png, curso3.png  
18 |-- Instagram.png, Facebook.png, Whatsapp.png
```

Listing 1: Estructura de directorios

2.2 Páginas Desarrolladas

2.2.1. index.html - Página Principal

Contiene:

- Navbar con logo y menú de navegación
- Carrusel automático con 3 slides
- Sección de cursos con 3 tarjetas
- Footer con información de contacto y redes sociales

2.2.2. novedades.html - Novedades

Presenta:

- Mismo navbar y footer
- Sección de anuncios con tarjetas informativas
- Información sobre inscripciones y actualizaciones

2.2.3. quienes-somos.html - Información Institucional

Incluye:

- Descripción de la empresa
- Misión, visión y valores
- Sección 'Nuestro enfoque' con tarjetas

2.2.4. contacto.html - Formulario de Contacto

Contiene:

- Formulario completo con validaciones
- Campos: nombre y apellido, email, teléfono, país, comentarios
- Mensajes de éxito y error

3 Implementación de Requisitos

3.1 Requisito 1: Estructura Básica del Sitio

Cumplimiento: ✓ Completado

Se crearon las 4 páginas solicitadas con estructura HTML5 semántica:

```
1 <!DOCTYPE html>
2 <html lang="es">
3 <head>
4     <meta charset="UTF-8">
5     <meta name="viewport" content="width=device-width, initial-scale=1.0">
6     <title>Titulo</title>
7     <link rel="stylesheet" href="css/styles.css">
8 </head>
9 <body>
10     <nav class="navbar">...</nav>
11     <main>...</main>
12     <footer class="footer">...</footer>
13     <script src="js/script.js"></script>
14 </body>
15 </html>
```

Listing 2: Estructura base HTML

3.2 Requisito 2: Elementos Visuales

3.2.1. Logo de la Empresa

Cumplimiento: ✓ Completado

El logo se implementó en el navbar de todas las páginas:

```
1 <nav class="navbar">
2     <div class="container">
3         
4         <!-- Menu -->
5     </div>
6 </nav>
```

Listing 3: Implementación del logo

3.2.2. Barra de Navegación

Cumplimiento: ✓ Completado

Se implementó un navbar sticky con menú hamburguesa para móviles:

```
1 <nav class="navbar" id="navbar">
2     <div class="container">
3         
4
5         <ul class="nav-menu" id="nav-menu">
6             <li><a href="index.html">Home</a></li>
7             <li><a href="novedades.html">Novedades</a></li>
8             <li><a href="quienes-somos.html">Quienes Somos</a></li>
9             <li><a href="contacto.html">Contacto</a></li>
10        </ul>
```

```

11         <button class="menu-toggle" id="menuToggle">☰</button>
12     </div>
13 </nav>
14

```

Listing 4: Navbar responsive

Características implementadas:

- Posición sticky (permanece visible al hacer scroll)
- Efecto glassmorphism con backdrop-filter
- Menú hamburguesa para dispositivos móviles
- Animaciones suaves en hover

3.2.3. Carrusel con Mínimo 3 Slides**Cumplimiento:** ✓ Completado (3 slides)

Se implementó un carrusel completamente funcional:

```

1 <section class="carousel">
2   <div class="carousel-container">
3     <div class="slide active">
4       
5       <div class="slide-text">
6         <h2>Certificados con aval Internacional</h2>
7       </div>
8     </div>
9     <!-- Slides 2 y 3 -->
10  </div>
11
12  <button class="prev" onclick="changeSlide(-1)"></button>
13  <button class="next" onclick="changeSlide(1)"></button>
14
15  <div class="dots">
16    <span class="dot active" onclick="currentSlide(0)"></span>
17    <span class="dot" onclick="currentSlide(1)"></span>
18    <span class="dot" onclick="currentSlide(2)"></span>
19  </div>
20 </section>

```

Listing 5: Estructura del carrusel

Funcionalidades del carrusel:

- Rotación automática cada 5 segundos
- Botones de navegación (anterior/siguiente)
- Dots para navegación directa
- Transiciones suaves con CSS
- Reinicio del timer al interactuar manualmente

3.2.4. Diseño Responsive

Cumplimiento: ✓ Completado

Se implementó diseño responsive con enfoque mobile-first:

```

1 /* Celular por defecto */
2 .course-grid {
3     grid-template-columns: 1fr;
4 }
5
6 /* Tablet */
7 @media (min-width: 640px) {
8     .course-grid {
9         grid-template-columns: repeat(2, 1fr);
10    }
11 }
12
13 /* Escritorio */
14 @media (min-width: 1024px) {
15     .course-grid {
16         grid-template-columns: repeat(3, 1fr);
17    }
18 }

```

Listing 6: Media queries implementadas

Breakpoints utilizados:

- Celular: $< 640\text{px}$
- Tablet: $640\text{px} - 1023\text{px}$
- Escritorio: $\geq 1024\text{px}$

3.3 Requisito 3: Secciones Específicas

3.3.1. Sección de Interés en Home

Cumplimiento: ✓ Completado

Se implementó una sección de cursos con tarjetas:

```

1 <section class="courses">
2     <div class="container">
3         <h2>Nuestros cursos</h2>
4
5         <div class="course-grid">
6             <div class="course-card">
7                 
8                 <h3>Reparacion de plaquetas electronicas</h3>
9                 <p>Para cualquier persona con o sin conocimientos</p>
10            </div>
11            <!-- Mas tarjetas -->
12        </div>
13    </div>
14 </section>

```

Listing 7: Seccion de cursos

3.3.2. Footer Genérico

Cumplimiento: ✓ Completado en todas las páginas

```

1 <footer class="footer">
2   <div class="container">
3     <div class="footer-content">
4       <div class="footer-section">
5         
6         <p>17 de agosto 752, CP 1500 CABA, Argentina</p>
7       </div>
8
9       <div class="footer-section">
10        <p>cursos@eprenda.com</p>
11        <div class="social-icons">
12          <a href="#"></a>
13          <a href="#"></a>
14          <a href="#"></a>
15        </div>
16      </div>
17    </div>
18
19    <p class="footer-bottom">
20      Terminos y condiciones - Eprenda V 1.1.69
21    </p>
22  </div>
23 </footer>

```

Listing 8: Footer implementado

3.3.3. Formulario de Contacto con Validaciones

Cumplimiento: ✓ Completado

Se implementó un formulario completo con validaciones en JavaScript:

```

1 <form id="contactForm" class="contact-form">
2   <div class="form-group">
3     <label for="nombre">Nombre y Apellido *</label>
4     <input type="text" id="nombre" required>
5   </div>
6
7   <div class="form-group">
8     <label for="email">Email *</label>
9     <input type="text" id="email" required>
10  </div>
11
12  <div class="form-group">
13    <label for="telefono">Telefono</label>
14    <input type="tel" id="telefono">
15  </div>
16
17  <div class="form-group">
18    <label for="pais">Pais</label>
19    <select id="pais">
20      <option value="">Seleccione un pais</option>
21      <option value="argentina">Argentina</option>
22      <!-- Mas opciones -->
23    </select>
24  </div>

```

```
25
26     <div class="form-group">
27         <label for="comentarios">Comentarios</label>
28         <textarea id="comentarios"></textarea>
29     </div>
30
31     <button type="submit" class="btn-submit">Enviar Mensaje</button>
32 </form>
33
34 <div id="mensaje" class="mensaje"></div>
```

Listing 9: Formulario de contacto

4 Desarrollo del JavaScript

4.1 Carrusel Automático (carousel.js)

Se implementó la lógica completa del carrusel:

```
1 // Variables globales
2 let currentIndex = 0;
3 const slides = document.querySelectorAll('.slide');
4 const dots = document.querySelectorAll('.dot');
5 let autoSlideInterval;
6
7 // Mostrar slide especifica
8 function showSlide(index) {
9     // Remover clase active de todos
10    slides.forEach(slide => slide.classList.remove('active'));
11    dots.forEach(dot => dot.classList.remove('active'));
12
13    // Manejar limites
14    if (index >= slides.length) {
15        currentIndex = 0;
16    } else if (index < 0) {
17        currentIndex = slides.length - 1;
18    } else {
19        currentIndex = index;
20    }
21
22    // Activar slide actual
23    slides[currentIndex].classList.add('active');
24    dots[currentIndex].classList.add('active');
25 }
26
27 // Cambiar slide con botones
28 function changeSlide(direction) {
29     showSlide(currentIndex + direction);
30     resetAutoSlide();
31 }
32
33 // Ir a slide especifica (dots)
34 function currentSlide(index) {
35     showSlide(index);
36     resetAutoSlide();
37 }
38
```

```
39 // Auto-slide cada 5 segundos
40 function startAutoSlide() {
41     autoSlideInterval = setInterval(() => {
42         showSlide(currentIndex + 1);
43     }, 5000);
44 }
45
46 // Reiniciar timer
47 function resetAutoSlide() {
48     clearInterval(autoSlideInterval);
49     startAutoSlide();
50 }
51
52 // Inicializar
53 showSlide(0);
54 startAutoSlide();
```

Listing 10: Logica del carrusel

Explicación del código:

1. Se declaran variables globales para el índice actual y elementos del DOM
2. `showSlide()` maneja la lógica de mostrar/ocultar slides
3. `changeSlide()` permite navegación con botones
4. `startAutoSlide()` inicia la rotación automática
5. `resetAutoSlide()` reinicia el timer al interactuar

4.2 Menú Hamburguesa (menu.js)

Implementación del menú responsive:

```
1 // Seleccionar elementos
2 const menuToggle = document.getElementById('menuToggle');
3 const navMenu = document.querySelector('.nav-menu');
4
5 // Toggle del menu
6 menuToggle.addEventListener('click', function() {
7     navMenu.classList.toggle('active');
8 });
9
10 // Cerrar menu al hacer click en enlace
11 const navLinks = document.querySelectorAll('.nav-menu a');
12 navLinks.forEach(link => {
13     link.addEventListener('click', function() {
14         navMenu.classList.remove('active');
15     });
16 });
```

Listing 11: Menu hamburguesa

Funcionalidad:

- Al hacer click en el botón hamburguesa, se muestra/oculta el menú
- Al seleccionar un enlace, el menú se cierra automáticamente
- Mejora la experiencia en dispositivos móviles

4.3 Validación de Formulario (form.js)

Sistema completo de validación:

```
1 // Elementos del DOM
2 const form = document.getElementById('contactForm');
3 const mensaje = document.getElementById('mensaje');
4 const telefonoInput = document.getElementById('telefono');
5
6 // Limites de caracteres
7 const LIMITES = {
8     nombre: 100,
9     email: 100,
10    telefono: 20,
11    comentarios: 500
12 };
13
14 // Validacion en tiempo real para telefono
15 telefonoInput.addEventListener('input', function(e) {
16     // Solo numeros, +, -, espacios y parentesis
17     e.target.value = e.target.value.replace(/[\d+\-\s()]/g, '');
18
19     // Aplicar limite
20     if(e.target.value.length > LIMITES.telefono) {
21         e.target.value = e.target.value.slice(0, LIMITES.telefono);
22     }
23 });
24
25 // Validacion al enviar
26 form.addEventListener('submit', function(e) {
27     e.preventDefault();
28
29     // Obtener valores
30     const nombre = document.getElementById('nombre').value.trim();
31     const email = document.getElementById('email').value.trim();
32
33     // Validar nombre
34     if (nombre === '') {
35         mostrarMensaje('Por favor, ingrese su nombre', 'error');
36         return;
37     }
38
39     // Validar email
40     if (email === '') {
41         mostrarMensaje('Por favor, ingrese su email', 'error');
42         return;
43     }
44
45     const emailRegex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;
46     if(!emailRegex.test(email)) {
47         mostrarMensaje('Email invalido', 'error');
48         return;
49     }
50
51     // Exito
52     mostrarMensaje('Mensaje enviado con exito!', 'exito');
53     form.reset();
54 });
55
```

```
56 // Mostrar mensaje al usuario
57 function mostrarMensaje(texto, tipo) {
58     mensaje.textContent = texto;
59     mensaje.className = 'mensaje ${tipo}';
60     mensaje.scrollIntoView({ behavior: 'smooth' });
61 }
```

Listing 12: Validaciones del formulario

Validaciones implementadas:

1. **Campos obligatorios:** Nombre y email son requeridos
2. **Formato de email:** Validación con expresión regular
3. **Teléfono:** Solo permite números y caracteres especiales (+, -, espacios)
4. **Límites de caracteres:** Todos los campos tienen límites
5. **Validación en tiempo real:** El teléfono se valida mientras se escribe
6. **Mensajes visuales:** Feedback inmediato de éxito o error

5 Desarrollo del CSS

5.1 Sistema de Variables

Se utilizaron variables CSS para mantener consistencia:

```
1 :root {
2     /* Colores */
3     --primary: #0b63ce;
4     --primary-dark: #084fa3;
5     --accent: #ffbf2f;
6     --text: #1f2937;
7     --bg: #f6f7fb;
8
9     /* Sombras */
10    --shadow-sm: 0 2px 8px rgba(0,0,0,.08);
11    --shadow-md: 0 4px 16px rgba(0,0,0,.12);
12
13    /* Radios */
14    --radius: 14px;
15    --radius-sm: 10px;
16
17    /* Animaciones */
18    --ease-smooth: cubic-bezier(0.4, 0, 0.2, 1);
19 }
```

Listing 13: Variables CSS

5.2 Efectos Visuales

5.2.1. Glassmorphism en Navbar

```
1 .navbar {  
2     background: rgba(255,255,255,.95);  
3     backdrop-filter: blur(12px) saturate(180%);  
4     border-bottom: 1px solid var(--border);  
5 }
```

Listing 14: Efecto glassmorphism

5.2.2. Animaciones

```
1 @keyframes fadeInUp {  
2     from {  
3         opacity: 0;  
4         transform: translateY(15px);  
5     }  
6     to {  
7         opacity: 1;  
8         transform: translateY(0);  
9     }  
10 }  
11  
12 .course-card {  
13     animation: fadeInUp 0.5s ease both;  
14 }  
15  
16 .course-card:hover {  
17     transform: translateY(-8px);  
18     box-shadow: var(--shadow-lg);  
19 }
```

Listing 15: Animaciones CSS

5.3 Diseño Responsive

5.3.1. Menú Móvil

```
1 @media (max-width: 767px) {  
2     .menu-toggle {  
3         display: inline-grid;  
4     }  
5  
6     .nav-menu {  
7         display: none;  
8     }  
9  
10    .nav-menu.active {  
11        display: flex;  
12        flex-direction: column;  
13        position: absolute;  
14        top: var(--nav-h);  
15        left: 0;  
16        right: 0;  
17        background: rgba(255,255,255,.98);  
18        animation: slideDown 0.3s ease;  
19    }
```

20 }

Listing 16: Menu responsive

6 Desafíos y Soluciones

6.1 Desafío 1: Carrusel Automático

Problema: Implementar un carrusel que rote automáticamente pero que también permita control manual.

Solución: Se utilizó `setInterval` para la rotación automática y `clearInterval` + reinicio del timer cuando el usuario interactúa manualmente.

6.2 Desafío 2: Validación de Formulario

Problema: Validar el formulario sin recargar la página y mostrar mensajes específicos.

Solución: Se utilizó `preventDefault()` para evitar el envío del formulario y se implementaron validaciones con expresiones regulares y condicionales.

6.3 Desafío 3: Diseño Responsive

Problema: Adaptar el diseño a diferentes tamaños de pantalla.

Solución: Se implementó un enfoque mobile-first con media queries y CSS Grid para layouts flexibles.

7 Resultados y Conclusiones

7.1 Cumplimiento de Requisitos

| Requisito | Estado |
|-----------------------------|--------|
| 4 páginas HTML | ✓ |
| Barra de navegación | ✓ |
| Carrusel con 3+ slides | ✓ |
| Diseño responsive | ✓ |
| Formulario con validaciones | ✓ |
| Mensajes de éxito/error | ✓ |
| Footer en todas las páginas | ✓ |

Cuadro 1: Cumplimiento de requisitos del examen

7.2 Funcionalidades Adicionales

Además de los requisitos, se implementaron:

- Animaciones CSS suaves
- Efectos hover en tarjetas y botones

- Glassmorphism en navbar y carrusel
- Validación en tiempo real del teléfono
- Auto-cierre del menú móvil
- Scroll suave al mensaje de feedback

7.3 Aprendizajes

Durante el desarrollo de la pagina se aplicaron los siguientes conceptos:

1. **HTML Semántico:** Uso correcto de etiquetas como `<nav>`, `<main>`, `<section>`, `<footer>`
2. **CSS Moderno:** Variables CSS, Grid, Flexbox, animaciones y media queries
3. **JavaScript Vanilla:** Manipulación del DOM, eventos, validaciones y timers
4. **Responsive Design:** Enfoque mobile-first y adaptación a diferentes dispositivos
5. **UX/UI:** Feedback visual, animaciones suaves y accesibilidad

7.4 Conclusión

La pagina cumple exitosamente con todos los requisitos del examen. Se desarrolló un sitio web completamente funcional, responsive y con interactividad mediante JavaScript vanilla. El código está organizado de manera modular, facilitando el mantenimiento y futuras mejoras.