

UNIVERSIDAD PRIVADA DEL VALLE



# ***MANUAL DE USUARIO – UNIVERSITY OF NEW HAVEN***

**MATERIA:** PROGRAMACION WEB 1

**INTEGRANTES:** PAUCARA MAMANI OLIVER YASSIR

**DOCENTE:** HENRY MIRANDA ORDONEZ

**SEMESTRE:** SEGUNDO SEMESTRE

**CARRERA:** ING. EN SISTEMAS INFORMÁTICOS

**LA PAZ – BOLIVIA**

# **GUIA DE NAVEGACION DE LA PLATAFORMA UNIVERSITY OF NEW HAVEN**

**Versión 1.0**

**Fecha : 08/12/2025**

## **1. Introduccion y Requisitos**

### **1.1 Propósito del Documento**

Le damos la más cordial bienvenida al Manual de Usuario de la plataforma web University of New Haven.

Este documento ha sido creado para servir como una guía fundamental y accesible, cuyo propósito es facilitar a usted, el usuario final, la comprensión total de la estructura y el contenido de nuestro sitio web. Nuestro objetivo es asegurar que la experiencia de navegación sea intuitiva, eficiente y libre de contratiempos.

Dada la naturaleza del proyecto un sitio web estático, informativo y sin necesidad de cuentas de usuario o transacciones complejas, este manual se enfoca en explicar cómo acceder, interpretar y moverse a través de las diferentes secciones de contenido (Proyectos, Programas, Catálogo y Organizaciones).

### **1.2 Requisitos Mínimos**

Para la experiencia optima, recomendamos:

- Navegacion Web: Chrome, Firefox, Edge o Safari (Versiones recientes).
- Conexion: Acceso a internet estable.
- Dispositivo: Funciona en computadoras de escritorio, tabletas y teléfonos mpviles (diseño responsive).

### **1.3 Objetivos de la Plataforma**

La plataforma Web de University of New Haven ha sido desarrollada como un recurso puramente informativo. Su función primordial es la difusión clara y organizada de la información relativa: cursos, proyectos académicos, organizaciones colaboradoras, etc.

#### **Puntos clave:**

**No requiere registro:** La información está disponible al público en general sin necesidad de crear una cuenta o iniciar sesión.

**Consulta Rápida:** La navegación ha sido diseñada para permitirle encontrar la información que busca en el menor número de clics posible.

**Simplicidad y Estabilidad:** Al ser una plataforma sin bases de datos ni lógicas de servidor complejas, garantiza una alta velocidad de carga y una excepcional estabilidad.

## 2. Navegación general

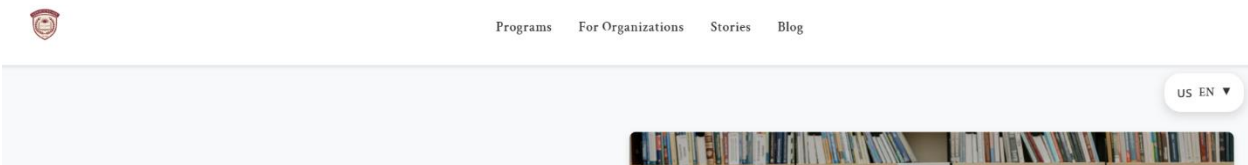
La estructura de la página es simple y se basa en la navegación directa a las secciones clave:

Sección Principal	Propósito	Archivo HTML
Inicio / Proyecto	Presentación principal y visión del proyecto.	proyecto.html
Programas	Muestra la lista o detalles de los programas ofrecidos.	programs.html
Catálogo	Listado de los ítems o cursos disponibles.	catalogo.html
Organizaciones	Información sobre las organizaciones asociadas o relevantes.	organizations.html

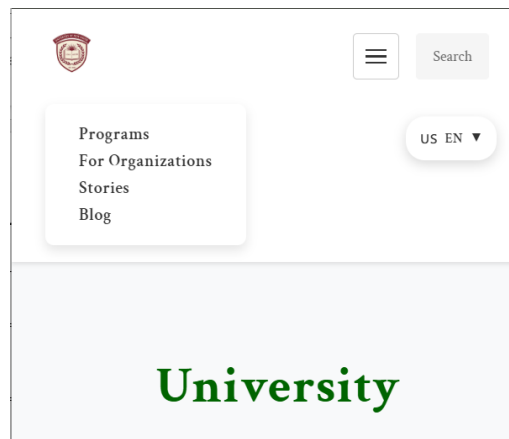
### 2.1. Uso del Menú

El menú de navegación principal se encuentra en la parte superior (Header) de todas las páginas.

- **En Escritorio:** Los enlaces son visibles permanentemente.



- **En Móviles:** El menú se contrae detrás de un **icono de hamburguesa** (tres líneas horizontales). Toca el icono para desplegar las opciones.

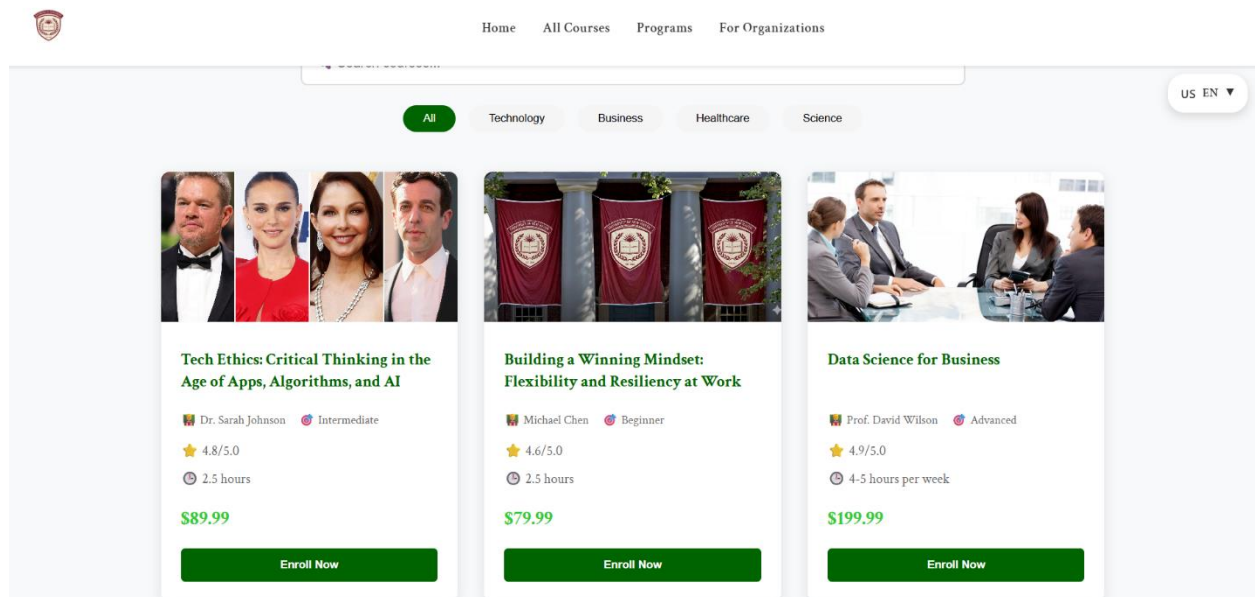


### 3. Guía Detallada de Funcionalidades

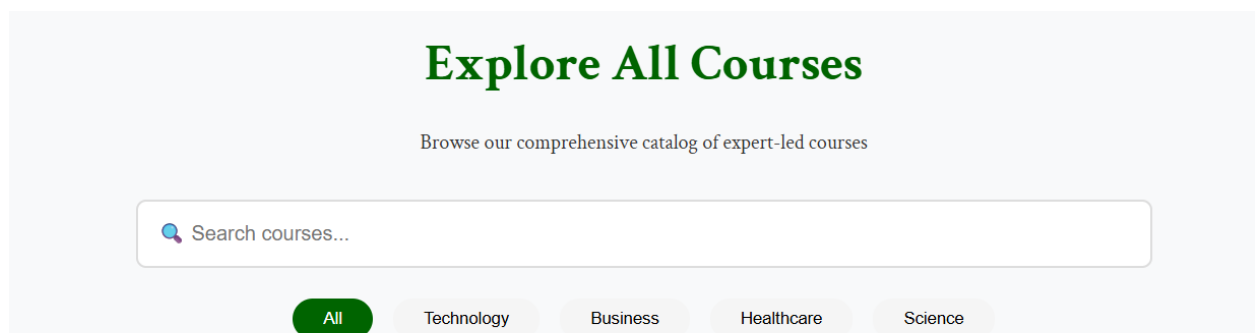
#### 3.1. Visualización del Catálogo (catalogo.html)

Esta sección presenta una lista de elementos (cursos, productos, etc.).

- **Visualización:** Los ítems se muestran en formato de cuadrícula o lista para facilitar la exploración.



- **Búsqueda/Filtro:** Si existe un campo de texto o botones de filtro, úsalos para acotar la lista de resultados. (La funcionalidad de JavaScript reside en `catalogo.js`).



### 3.2. Detalles de un Ítem (curso-detalle.html)

Al hacer clic en cualquier elemento del **Catálogo**, serás dirigido a la página de detalle:

- Aquí encontrarás la información ampliada del ítem (descripción completa, requisitos, imágenes adicionales, etc.).
- Para volver al catálogo, usa el enlace de **"Volver al Catálogo"** o el botón de atrás de tu navegador.

US EN ▼

## Tech Ethics: Critical Thinking in the Age of Apps, Algorithms, and AI

This course explores ethical frameworks for technology development and deployment. Learn to identify and address ethical challenges in AI, algorithms, and digital platforms.

2.5 hours

Dr. Sarah Johnson

Intermediate

4.8/5.0 (250 reviews)

### About This Course

This course explores ethical frameworks for technology development and deployment. Learn to identify and address ethical challenges in AI, algorithms, and digital platforms.

### What You'll Learn

- ✓ Understand core ethical principles in technology
- ✓ Identify ethical challenges and AI systems


### 3.3. Cursos y adaptación de el cursor

Una gran adaptación de Cursos promocionados por parte de la Universidad y a la vez un diseño o animación responsiva al pasar cursor al pasar por el anuncio del curso detallado

Power Skills for Career Growth: Developed by leading faculty and brought to life with insights from award-winning celebrities


US EN ▼

Featured




### Building a Winning Mindset: Flexibility and Resiliency at Work

2.5 hours • Start today



### Tech Ethics: Critical Thinking in the Age of Apps, Algorithms, and AI

2.5 hours • Start today



### Cultivating Power for Positive Impact

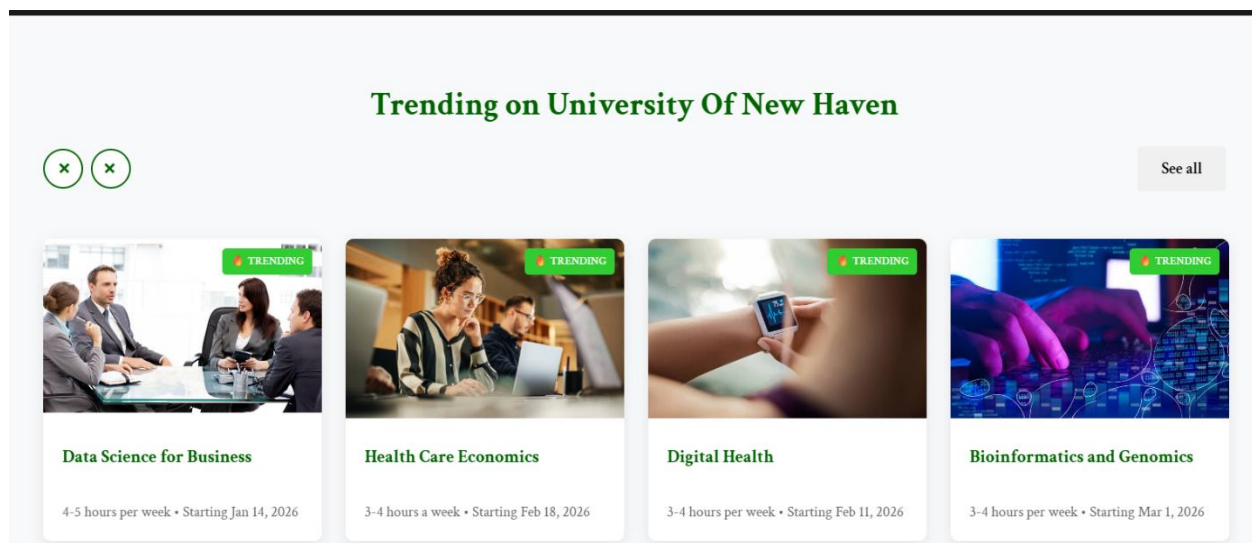
2.5 hours • Start today

### 3.4. Catálogo y Visualización de Programas

La información central del sitio se presenta a través de Tarjetas de Programa o Tarjetas de Curso. Estas tarjetas están diseñadas para ofrecer un resumen conciso antes de acceder al detalle completo.

#### 3.4.1. Interpretación de las Tarjetas Trending

En la sección titulada "Trending on University Of New Haven" (y potencialmente en otras listas de programas), cada tarjeta de programa contiene la siguiente información clave:



**Etiqueta de Tendencia:** El distintivo verde "TRENDING" indica que este programa es actualmente popular o está siendo destacado por la universidad.

**Título del Programa:** El nombre completo del curso o programa (ej., "Data Science for Business" o "Digital Health").

**Dedicatoria Semanal:** Indica la carga horaria estimada que el usuario deberá dedicar al curso por semana (ej., "4-5 hours per week").

**Fecha de Inicio:** Muestra la fecha en la que comenzará la próxima cohorte o sesión del curso (ej., "Starting Jan 14, 2026").

#### Cómo utilizar esta sección:

**Visualización Rápida:** Utilice las tarjetas para identificar rápidamente los programas más relevantes por título y carga horaria.

**Acceso al Detalle:** Haga clic en cualquier parte de la tarjeta para navegar a la página Detalle de Curso (curso-detalle.html), donde encontrará el temario completo, requisitos, y más información.

### **3.5. Opciones de Navegación Secundaria (Header)**

Aunque el menú principal (Hamburguesa/Barra) maneja la navegación clave, en el encabezado superior puede encontrar enlaces secundarios para acceder a contenido más específico:

Programs: Acceso directo a la sección de programas.

For Organizations: Información dirigida específicamente a organizaciones.

Stories: Acceso a historias o casos de éxito relacionados con la universidad.

Blog: Enlaces a artículos y publicaciones informativas de la plataforma.

### **3.6. Suscripción a la Newsletter**

El sitio web ofrece una sección destacada "Stay tuned for more" (Mantente al tanto para más) para recibir actualizaciones directas. Esta es una función de entrada de datos simple:

**Propósito:** La suscripción sirve para "no perderse nada" y "obtener actualizaciones sobre contenido exclusivo" dirigido a los estudiantes de la University of New Haven.

#### **Proceso de Suscripción:**

**Paso 1:** Escriba su dirección de correo electrónico en el campo etiquetado como "Email Address".

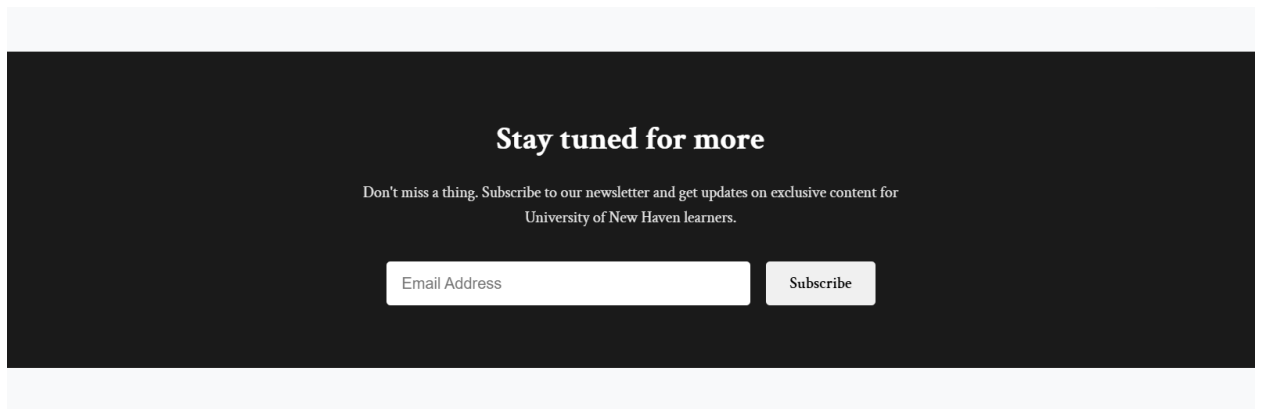
**Paso 2:** Haga clic en el botón "Subscribe".

**Resultado:** Al hacer clic en "Subscribe", su dirección de correo electrónico se enviará a la lista de correo del proyecto para recibir futuras comunicaciones.

### **3.7. Artículos y Webinars**

Justo debajo de la sección de suscripción, encontrará un llamado a la acción: "Learn more with free articles and webinars" (Aprenda más con artículos y seminarios web gratuitos).

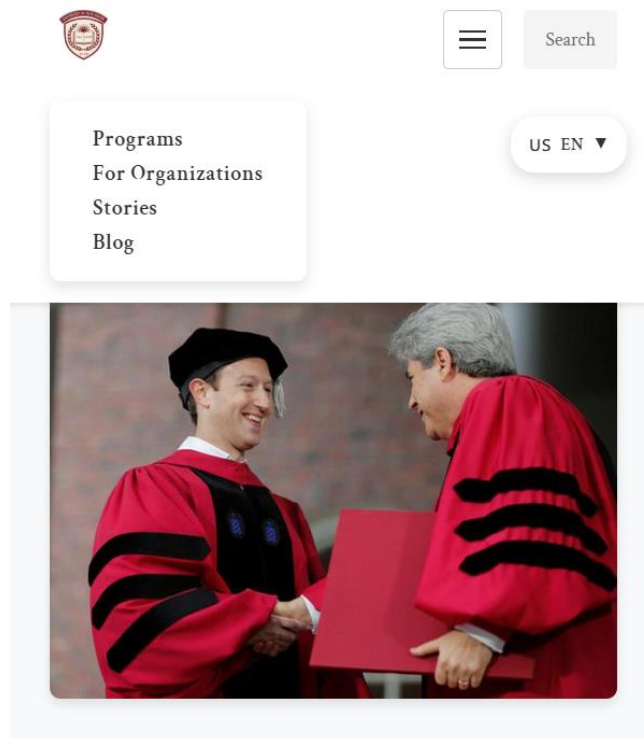
**Botón "See all":** Este botón lo redirigirá a la sección completa de recursos gratuitos (probablemente al blog o a una página de recursos, complementando el enlace "Blog" del Header).



#### 4. Diseño y Adaptabilidad

El sitio web fue desarrollado con un diseño **Responsive** (adaptable). Esto significa que la interfaz se ajusta automáticamente al tamaño de tu pantalla:

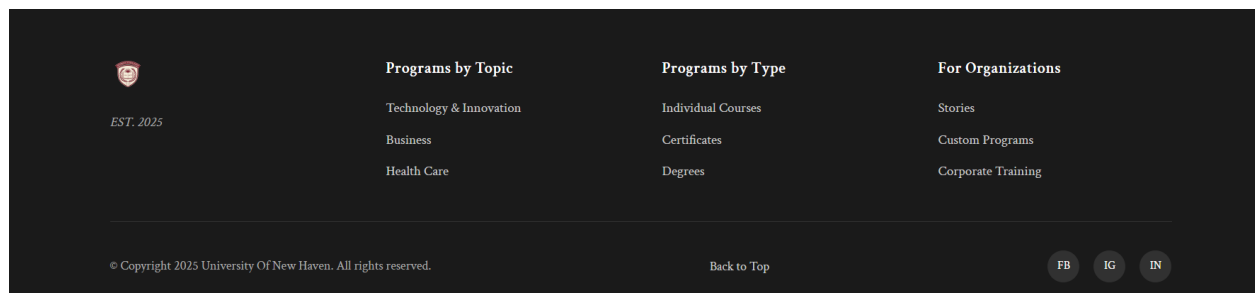
- El tamaño de las imágenes, el texto y la disposición de las columnas cambian para asegurar que la lectura sea cómoda en cualquier dispositivo.





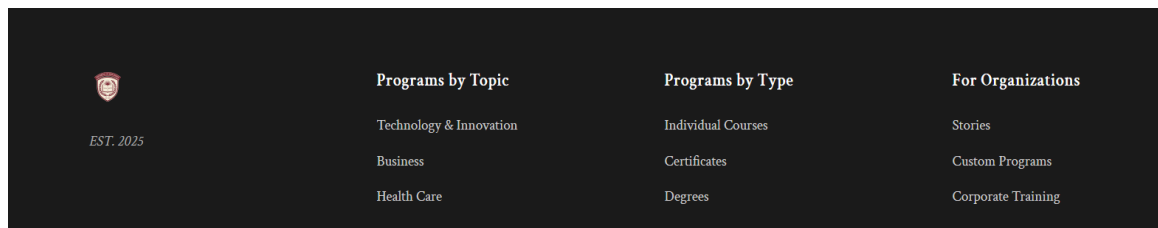
## 5. Uso y Funcionalidad del Pie de Página (Footer)

El Pie de Página, ubicado en la parte inferior de todas las páginas del sitio web, cumple dos funciones principales: proporcionar navegación secundaria y ofrecer información de contacto y legal.



### 5.1. Enlaces de Navegación Rápida

Esta sección replica enlaces importantes del menú principal o incluye accesos directos a páginas clave que complementan la información del sitio, asegurando que siempre pueda saltar a una sección principal sin tener que desplazarse hasta el encabezado.



### 5.2. Información de Contacto y Redes Sociales

Aquí se localizan los medios para contactar al equipo responsable del proyecto:

Email o Dirección: Si aplica, se muestra un correo electrónico de contacto o una dirección física.

**Iconos de Redes Sociales:** Encontrará los enlaces directos a las cuentas de redes sociales del proyecto (como LinkedIn, Twitter o Instagram). Simplemente haga clic en el ícono correspondiente para ser redirigido.



### 5.3. Avisos Legales y Derechos de Autor

El pie de página es el lugar estándar para encontrar la información de cumplimiento legal:

**Derechos de Autor (Copyright):** Indica el año y la entidad que posee los derechos del contenido.

© Copyright 2025 University Of New Haven. All rights reserved.

UNIVERSIDAD PRIVADA DEL VALLE



# ***MANUAL TECNICO – UNIVERSITY OF NEW HAVEN***

**MATERIA:** PROGRAMACION WEB 1

**INTEGRANTES:** PAUCARA MAMANI OLIVER YASSIR

**DOCENTE:** HENRY MIRANDA ORDONEZ

**SEMESTRE:** SEGUNDO SEMESTRE

**CARRERA:** ING. EN SISTEMAS INFORMÁTICOS

**LA PAZ – BOLIVIA**

# **MANUAL TÉCNICO - PLATAFORMA UNIVERSITY OF NEW HAVEN**

**Versión 1.0 - Para desarrolladores y mantenedores**

## **1. INTRODUCCIÓN**

### **1.1 Propósito del Documento**

Este manual técnico proporciona información detallada sobre la arquitectura, estructura de archivos, tecnologías utilizadas y funcionalidades implementadas en la plataforma de educación en línea "Universidad 1970 Online". Está dirigido a desarrolladores, mantenedores y administradores técnicos.

### **1.2 Descripción del Sistema**

- **Nombre:** Universidad 1970 Online
- **Tipo:** Plataforma de cursos en línea (e-learning)
- **Arquitectura:** Frontend estático (HTML/CSS/JavaScript)
- **Estado:** Plataforma funcional de demostración
- **Objetivo:** Permitir exploración, filtrado y visualización de cursos online

## **2. ARQUITECTURA Y TECNOLOGÍAS**

### **2.1 Stack Tecnológico**

#### **Frontend:**

- └─ HTML5
- └─ CSS3 (con variables CSS y Flexbox/Grid)
- └─ JavaScript Vanilla (ES6+)
- └─ No backend (datos embebidos en JavaScript)

#### **Librerías externas:**

- └─ Font Awesome 6.0.0 (iconos)
- └─ Google Fonts (Crimson Text, Merriweather)
- └─ Google Fonts API

#### **Hosting recomendado:**

- └─ Servidor estático (Netlify, Vercel, GitHub Pages, Apache, Nginx)

## 2.2 Variables CSS Globales

css

```
:root {  
  --primary-color: #006400;    /* Verde oscuro */  
  --accent-color: #32CD32;     /* Verde lima */  
  --text-color: #333;  
  --bg-light: #f8f9fa;  
  --bg-dark: #1a1a1a;  
  --shadow: rgba(0, 0, 0, 0.1);  
  --shadow-detailed: 0 4px 8px rgba(0, 0, 0, 0.1);  
}
```

## 2.3 Estructura de Carpetas (Inferida)

universidad-1970-online/

```
|  
|— index.html          (renombrado a proyecto.html)  
|— catalogo.html  
|— curso-detalle.html  
|— programs.html  
|— organizations.html  
|  
|— estilos.css         (CSS principal)  
|— catalogo.css        (CSS específico catálogo)  
|— curso-detalle.css   (CSS específico detalle)  
|  
|— funcionalidad.js     (JS principal - funcionalidades generales)  
|— catalogo.js          (JS específico catálogo)  
|— curso-detalle.js     (JS específico detalle curso)
```

```

|
|— Images/                (Carpeta de imágenes)
|   |— Universidad1970-Photoroom.png
|   |— mosaico.jpeg
|   |— certificado.jpeg
|   |— universityofnewhaven.png
|   |— famososharvad.jpg
|   |— discusioncolaborativa.png
|   |— ... (todas las imágenes del sitio)
|
|— README.md              (Documentación)

```

### 3. ESTRUCTURA DE ARCHIVOS DETALLADA

#### 3.1 Archivos HTML

Archivo	Descripción	Dependencias
proyecto.html	Página principal / Home	estilos.css, funcionalidad.js
catalogo.html	Catálogo completo de cursos	estilos.css, catalogo.css, funcionalidad.js, catalogo.js
curso-detalle.html	Detalle individual de curso	estilos.css, curso-detalle.css, funcionalidad.js, curso-detalle.js
programs.html	Página de programas	estilos.css, funcionalidad.js
organizations.html	Página para organizaciones	estilos.css, funcionalidad.js

### 3.2 Archivos CSS

Archivo	Responsabilidad	Especificidad
estilos.css (2,237 líneas)	Estilos globales y diseño responsivo del sitio	<b>Alta</b> – Contiene la base visual completa del sitio
catalogo.css (66 líneas)	Estilos específicos del catálogo de cursos	<b>Media</b> – Complementa estilos.css
curso-detalle.css (58 líneas)	Estilos exclusivos para la página de detalle de curso	<b>Media</b> – Extiende los estilos globales
programs.css (si aplica)	Estilos propios para la sección de programas	<b>Media/Baja</b> – Solo para una página concreta
organizations.css (si aplica)	Estilos para la página de organizaciones	<b>Media/Baja</b> – Estilos adicionales no globales
responsive.css (si existe)	Ajustes para diferentes resoluciones	<b>Alta</b> – Afecta múltiples páginas

### 3.3 Archivos JavaScript

Archivo	Responsabilidad	Datos / Módulos
funcionalidad.js (418 líneas)	Funcionalidades globales y navegación	Menú hamburguesa, carruseles, notificaciones
catalogo.js (171 líneas)	Lógica del catálogo	Objeto cursosCatalogo con 10 cursos
curso-detalle.js (303 líneas)	Lógica para página de detalle de curso	Objeto cursosCompletos con 10 cursos detallados

## 4. BASE DE DATOS EMBEBIDA

### 4.1 Estructura de Datos de Cursos

### En `catalogo.js` (vista resumida):

javascript

```
const cursosCatalogo = {  
  'tech-ethics': {  
    titulo: 'Tech Ethics: Critical Thinking...',  
    precio: '$89.99',  
    duracion: '2.5 hours',  
    categoria: 'technology',  
    imagen: 'Images/famososharvad.jpg',  
    rating: '4.8',  
    instructor: 'Dr. Sarah Johnson',  
    nivel: 'Intermediate'  
  },  
  // ... 9 cursos más  
};
```

### En `curso-detalle.js` (vista detallada):

javascript

```
const cursosCompleto = {  
  'tech-ethics': {  
    id: 'tech-ethics',  
    titulo: 'Tech Ethics: Critical Thinking...',  
    descripcion: 'This course explores ethical frameworks...',  
    precio: '$89.99',  
    precioOriginal: '$149.99',  
    duracion: '2.5 hours',  
    instructor: 'Dr. Sarah Johnson',  
  },  
  // ... 9 cursos más  
};
```



```
nivel: 'Intermediate',
rating: '4.8',
reviews: '250 reviews',
imagen: 'Images/famososharvad.jpg',
modulos: ['Introduction to Tech Ethics', ...], // 7 módulos
objetivos: ['Understand core ethical principles...', ...], // 5 objetivos
requisitos: ['Basic understanding of technology...', ...] // 4 requisitos
},
// ... 9 cursos más con estructura similar
};
```

## 4.2 Categorías de Cursos

1. **technology (Tecnología)**
2. **business (Negocios)**
3. **healthcare (Salud)**
4. **science (Ciencia)**

## 4.3 Niveles de Dificultad

- **Beginner (Principiante)**
- **Intermediate (Intermedio)**
- **Advanced (Avanzado)**

## 5. FUNCIONALIDADES PRINCIPALES

### 5.1 Sistema de Navegación

javascript

// Funcionalidades implementadas en funcionalidad.js:

1. Menú hamburguesa responsivo
2. Smooth scroll para enlaces internos
3. Navegación entre páginas
4. Cierre automático de menú al seleccionar opción

## 5. Indicador visual de página activa

### 5.2 Sistema de Cursos

javascript

// Flujo de datos:

Página Principal (proyecto.html)

(carga cursos vía funcionalidad.js)

Catálogo (catalogo.html) , Filtrado y búsqueda

(selección de curso)

Detalle Curso (curso-detalle.html?course=id)

### 5.3 Filtrado y Búsqueda

Implementado en `catalogo.js`:

- Filtrado por categoría (botones: All, Technology, Business, Healthcare, Science)
- Búsqueda en tiempo real por título o instructor
- Persistencia de filtros vía URL parameters (`?filter=trending`)

### 5.4 Sistema de Páginas de Detalle

Implementado en `curso-detalle.js`:

- Parámetro URL: `course=course-id`
- Renderizado dinámico basado en ID
- Funcionalidades: Inscripción, carrito, wishlist, compartir

### 5.5 Componentes Reutilizables

1. **Header/Navbar:** Responsivo con menú hamburguesa
2. **Course Card:** Tarjeta de curso con hover effects
3. **Carousel:** Carrusel horizontal con controles
4. **Footer:** Con columnas responsivas y redes sociales
5. **Notification System:** Sistema de notificaciones temporales

## 6. FLUJOS DE DATOS Y COMUNICACIÓN

### 6.1 Pasar Datos entre Páginas

javascript

// De catálogo a detalle:

```
window.location.href = `curso-detalle.html?course=${courseId}`;
```

// En detalle para obtener ID:

```
const params = new URLSearchParams(window.location.search);
```

```
const courseId = params.get('course') || 'tech-ethics';
```

### 6.2 Estructura de Eventos

javascript

// Eventos principales:

1. DOMContentLoaded / Inicialización
2. click en .filter-btn / Filtrado de cursos
3. input en .search-input / Búsqueda en tiempo real
4. click en .enroll-btn / Redirección a detalle
5. click en tarjetas / Redirección a detalle
6. resize window / Ajustes responsivos

## 7. DISEÑO RESPONSIVO

### 7.1 Breakpoints Definidos

css

En estilos.css

- **Pantallas grandes:** > 1024px (Desktop)
- **Pantallas medianas:** 768px - 1024px (Tablet)
- **Pantallas pequeñas:** 480px - 768px (Mobile grande)
- **Móviles:** < 480px (Mobile pequeño)

## 7.2 Patrones Responsivos

CSS

### Grids responsivos

```
.courses-grid {  
  grid-template-columns: repeat(auto-fill, minmax(300px, 1fr));  
}
```

### Flexbox para alineación

```
.main-header {  
  display: flex;  
  justify-content: space-between;  
  align-items: center;  
}
```

### Media queries específicas

```
@media (max-width: 768px) {  
  .main-nav { display: none; }  
  .main-nav.active { display: block; }  
}
```

## 8. SISTEMA DE ESTADOS Y NOTIFICACIONES

### 8.1 Estados Visuales

CSS

#### Estados de botones

```
.filter-btn.active { background: var(--primary-color); color: white; }
```

#### Estados hover

```
.btn:hover { transform: translateY(-2px); box-shadow: ...; }
```

## Estados focus

```
.btn:focus { outline: 2px solid var(--accent-color); }
```

## 8.2 Sistema de Notificaciones

javascript

```
// En funcionalidad.js - función showNotification()
```

### Características:

- **Posición:** fixed bottom-right
- **Animación:** slideIn/slideOut
- **Duración:** 3 segundos
- **Estilos:** Usa variables CSS
- **Uso:** Para feedback de usuario

## 9. PATRONES DE CÓDIGO Y CONVENCIONES

### 9.1 Nomenclatura

javascript

```
// Variables y funciones en inglés
```

```
const cursosCatalogo; // snake_case para objetos complejos
```

```
function renderCourses(); // camelCase para funciones
```

```
const courseId; // camelCase para variables
```

```
// IDs y clases en inglés
```

```
<div class="course-card-detailed">
```

```
<button class="enroll-now-btn">
```

```
<section class="global-testimonial-section">
```

### 9.2 Estructura de Componentes

html

```
<!-- Patrón típico de componente -->
```

```
<section class="component-section">
```

```
  <div class="section-header">
```

```
<h2>Título</h2>
<p>Descripción</p>
</div>
<div class="component-container">
  <article class="component-item">
    
    <div class="component-content">
      <h3>Subtítulo</h3>
      <p>Contenido</p>
      <button class="action-btn">Acción</button>
    </div>
  </article>
</div>
</section>
```

## 9.3 Comentarios y Documentación

javascript

// Comentarios en español (consistentes)

- **Función:** cargarCursosEnHome
- **Descripción:** Carga dinámicamente los cursos en la página principal
- **Dependencias:** cursosCatalogo, DOM cargado
- **Retorna:** void

```
function cargarCursosEnHome() { ... }
```

## 10. MANTENIMIENTO Y ACTUALIZACIONES

### 10.1 Agregar un Nuevo Curso

javascript

// 1. Agregar al catálogo (catalogo.js)

```
const nuevoCurso = {
  'nuevo-id': {
```

```
    titulo: 'Nuevo Curso',
    precio: '$99.99',
    duracion: '3 horas',
    categoria: 'technology', // technology|business|healthcare|science
    imagen: 'Images/nueva-imagen.jpg',
    rating: '4.5',
    instructor: 'Nombre Instructor',
    nivel: 'Intermediate' // Beginner|Intermediate|Advanced
  }
};
```

#### **// 2. Agregar detalles completos (curso-detalle.js)**

```
const nuevoDetalle = {
  'nuevo-id': {
    id: 'nuevo-id',
    titulo: 'Nuevo Curso',
    descripcion: 'Descripción completa...',
    // ... completar todos los campos como otros cursos
  }
};
```

#### **// 3. Agregar imagen a carpeta Images/**

#### **// 4. Actualizar funcionalidad.js si se muestra en home**

### **10.2 Modificar Estilos**

css

**Para cambios globales: estilos.css**

**Para cambios específicos: archivo CSS correspondiente**

**Ejemplo: Cambiar color primario**

```
:root {  
    --primary-color: #NUEVO_COLOR; /* Actualizar todos los usos */  
}
```

### 10.3 Extender Funcionalidades

javascript

// Para nuevas funcionalidades JavaScript:

// 1. Evaluar si va en funcionalidad.js (global) o archivo específico

// 2. Usar patrón existente de event listeners

// 3. Mantener consistencia de nombres y estilos

## 11. SISTEMA DE DEPURACIÓN Y LOGS

### 11.1 Consola de Desarrollo

javascript

**// Logs implementados en funcionalidad.js**

console.log("🔄 Cargando cursos en página principal...");

console.log("📊 Encontrado contenedor Trending");

console.log(`✅ Añadido curso: \${curso.titulo}`);

console.warn('⚠️ Advertencia: elemento no encontrado');

### 11.2 Validaciones y Fallbacks

javascript

**// Validación de elementos DOM**

if (!grid) return; // Salir si elemento no existe

**// Fallback para imágenes**





## // Valores por defecto

```
const courseId = params.get('course') || 'tech-ethics';
```

```
const curso = cursosCompletos[courseId] || cursosCompletos['tech-ethics'];
```

## 12. OPTIMIZACIONES Y MEJORES PRÁCTICAS

### 12.1 Performance

1. **CSS:** Todas las reglas en un archivo principal (estilos.css)
2. **JavaScript:** Cargado al final del body
3. **Imágenes:** Optimizadas y con dimensiones específicas
4. **Fuentes:** Cargadas desde CDN (Google Fonts)

### 12.2 Accesibilidad

html

```
<!-- Mejoras implementadas -->
```

```
<button aria-label="Previous slide"> <!-- Etiquetas ARIA -->
```

```
<img alt="Descripción significativa"> <!-- Alt text descriptivo -->
```

```
<nav role="navigation"> <!-- Roles semánticos -->
```

### 12.3 SEO Básico

html

```
<!-- Metadatos en head -->
```

```
<meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
<title>Descriptivo por página</title>
```

## 13. DESPLIEGUE Y HOSTING

### 13.1 Requisitos de Servidor

Requisitos mínimos:

- Servidor web estático (Apache, Nginx, etc.)
- Soporte para HTML5, CSS3, ES6+
- No requiere base de datos ni backend
- Certificado SSL recomendado

### Estructura mínima de archivos:

- Todos los archivos HTML
- Carpeta CSS con archivos CSS
- Carpeta JS con archivos JavaScript
- Carpeta Images con todas las imágenes

## 13.2 Pasos para Desplegar

bash

### 1. Preparar archivos

universidad-1970-online/

```
|— proyecto.html (renombrar a index.html para producción)
|— catalogo.html
|— curso-detalle.html
|— programs.html
|— organizations.html
|— css/
|   |— estilos.css
|   |— catalogo.css
|   |— curso-detalle.css
|— js/
|   |— funcionalidad.js
|   |— catalogo.js
|   |— curso-detalle.js
|— Images/ (todos los archivos de imagen)
```

# 2. Subir a servidor

# 3. Configurar reglas de reescritura si es necesario

# 4. Probar todas las funcionalidades

### 13.3 Plataformas Recomendadas

- GitHub Pages: Gratuito para repositorios públicos
- Netlify: Despliegue continuo desde GitHub
- Vercel: Optimizado para proyectos estáticos
- Hosting tradicional: Cualquier servidor web

## 14. SOLUCIÓN DE PROBLEMAS COMUNES

### 14.1 Problemas Frecuentes

Problema	Causa Probable	Solución
Menú no se despliega	JavaScript no cargado o errores en consola	Revisar consola del navegador; verificar funcionalidad.js
Cursos no se muestran	Error en los objetos de datos en JS	Revisar sintaxis en catalogo.js o curso-detalle.js
Estilos no aplican	Rutas incorrectas a los archivos CSS	Verificar rutas en HTML y la estructura de carpetas
Imágenes no cargan	Rutas incorrectas o archivos faltantes	Revisar carpeta Images/ y nombres exactos de las imágenes
Filtros no funcionan	Event listeners con errores o no vinculados	Revisar consola; depurar funciones en catalogo.js

### 14.2 Herramientas de Depuración

1. **Consola del Navegador:** Para errores JavaScript
2. **Inspector de Elementos:** Para problemas CSS/HTML
3. **Network Tab:** Para verificar carga de recursos
4. **Lighthouse:** Para análisis de performance y accesibilidad

## 15. ROADMAP Y MEJORAS FUTURAS

### 15.1 Mejoras Priorizadas

1. **Backend básico:** API REST para cursos (Node.js + Express)
2. **Sistema de usuarios:** Registro, login, perfil
3. **Carrito de compras:** Persistencia con localStorage
4. **Panel de administración:** CRUD para cursos

## 5. Sistema de pagos: Integración con Stripe/PayPal

### 15.2 Escalabilidad

javascript

// Migración a arquitectura más escalable:

Opción 1: Frontend Framework (React/Vue)

Opción 2: Backend + Base de datos (Node.js + MongoDB)

Opción 3: Headless CMS (Strapi, Contentful) para gestionar cursos

## 16. CONTACTOS Y SOPORTE

### 16.1 Estructura del Proyecto

Propietario: [Nombre del cliente]

Desarrollador: [Tu nombre/información]

Versión Actual: 1.0

Última Actualización: [Fecha]

### 16.2 Archivos Clave para Modificaciones

- **Datos de cursos:** `catalogo.js` y `curso-detalle.js`
- **Estilos principales:** `estilos.css`
- **Funcionalidades globales:** `funcionalidad.js`
- **Estructura HTML:** Archivos .html individuales

### 16.3 Recomendaciones para Nuevos Desarrolladores

- Leer completamente este manual técnico
- Explorar la estructura de archivos
- Probar todas las funcionalidades
- Revisar la consola del navegador para logs
- Seguir las convenciones establecidas

## APÉNDICE A: GLOSARIO DE TÉRMINOS

<b>Término</b>	<b>Descripción</b>
curso-detalle.htmlcourse=id	Patrón de URL usado para cargar la página de detalle de un curso según su ID
cursosCatalogo	Objeto JavaScript que contiene la vista resumida de los cursos
cursosCompletos	Objeto JavaScript que contiene la información detallada de los cursos
--primary-color	Variable CSS que define el color principal (#006400)
.active	Clase CSS utilizada para indicar un elemento activo o seleccionado
DOMContentLoaded	Evento JS que se ejecuta cuando el DOM ha terminado de cargarse

## **APÉNDICE B: CÓDIGOS DE ERROR COMUNES**

javascript

// Error: Cannot read property 'innerHTML' of null

// Solución: Verificar que elemento existe antes de manipularlo

// Error: cursosCatalogo is not defined

// Solución: Verificar que archivo catalogo.js está cargado

// Error: Failed to load resource (imágenes)

// Solución: Verificar rutas en carpeta Images/

**Documento generado automáticamente basado en análisis de código**

**© 2025 Universidad 1970 Online - Manual Técnico v1.0**

**Última revisión: 08/12/2025**