

Memoria del proyecto — Portal de intercambio de material académico

Desarrollo de Aplicaciones Web I — Bloque 1

GIISI

USAL

Alumnos: Pablo Alonso Hernández

Alumno: Juan Riego Vila

Fecha: 16-11-2025

Versión del documento: 1.0

Índice

1.	Resumen ejecutivo	4
2.	Objetivos y utilidad para las personas usuarias	5
2. 1.	Objetivos generales	5
2. 2.	Utilidad por perfil	5
2. 2. 1.	Usuario normal	5
2. 2. 2.	Usuario avanzado (colaborador)	5
2. 2. 3.	Administrador	5
3.	Requisitos y funcionalidades	7
3. 1.	Requisitos funcionales por perfil	7
3. 1. 1.	Usuario normal	7
3. 1. 2.	Usuario avanzado (colaborador)	7
3. 1. 3.	Administrador	8
3. 2.	Requisitos no funcionales	8
3. 3.	Dependencias y notas de implementación	9
4.	Información que manejará la aplicación	10
4. 1.	Usuario	10
4. 2.	Documento	10
4. 3.	Categoría	10
4. 4.	Comentario	11
4. 5.	Valoración	11
4. 6.	Colección	11
4. 7.	ColecciónDocumento (relación N-M)	11
4. 8.	Duda	12
4. 9.	Respuesta	12
4. 10.	Estados y catálogos (resumen)	12
4. 11.	Relaciones principales (visión general)	12
4. 12.	Consideraciones de datos y calidad	13
5.	Modelo de datos (Entidad-Relación)	14
5. 1.	Entidades y atributos	14
5. 2.	Relaciones y cardinalidades	15
6.	Diagrama entidad-relacion:	16
6. 1.	RELACIONES (cardinalidades)	16
6. 1. 1.	ORDENADO POR ATRIBUTO (Mejor visualización)	17
6. 2.	LEYENDA	17
6. 3.	Reglas de integridad y restricciones	17
6. 4.	Notas de implementación	18
7.	Mapa de la aplicación y flujos de navegación	19
7. 1.	Inventario de páginas	19
7. 2.	Navegación global	20
7. 3.	Flujo usuario normal	20
7. 4.	Flujo colaborador	20
7. 5.	Flujo administrador	20
7. 6.	Condiciones de visibilidad	20
8.	Diseño de interfaz y decisiones técnicas	21

8. 1.	Sistema de diseño	21
8. 2.	Responsive	22
8. 3.	Accesibilidad (AA)	22
9.	Páginas entregadas (prototipo HTML+CSS)	24
9. 1.	Página de inicio	24
9. 2.	Página que contiene los materiales académicos de la plataforma	25
9. 3.	Página que contiene las colecciones de materiales académicos.	25
9. 4.	Página que el apartado de preguntas entre usuarios de la plataforma. ...	26
9. 5.	Página que permite el inicio de sesión de un usuario ya previamente registrado.	26
9. 6.	Página que permite el registro de un usuario nuevo a la plataforma.	27
9. 7.	Página que permite el control de la plataforma al usuario administrador.	27
10.	Auditoria	28
10. 1.	Reparto de Tareas entre Integrantes	28
10. 1. 1.	Repartición de roles	28
10. 1. 2.	Rol A — UI/Frontend y Accesibilidad	28
10. 1. 3.	Rol B — Arquitectura de Información y Contenido	28
10. 1. 4.	Distribución de Carga	29
10. 2.	Metodología de Trabajo	29
10. 2. 1.	Fases del Proyecto	29
10. 2. 2.	Revisiones Cruzadas	29
10. 2. 3.	Control de Versiones	29
10. 2. 4.	Coordinación y Comunicación	29
10. 3.	Dificultades Encontradas y Resolución	30
10. 3. 1.	Dificultad 1: Coordinación de Estilos Globales	30
10. 3. 2.	Dificultad 2: Accesibilidad en Componentes Dinámicos	30
10. 3. 3.	Dificultad 3: Validación de Formularios sin JavaScript	30
10. 3. 4.	Dificultad 4: Responsive en Tablas Administrativas	30
10. 3. 5.	Dificultad 5: Sincronización del Modelo de Datos	30
10. 3. 6.	Dificultad 6: Cumplimiento de Contraste AA	30
10. 3. 7.	Dificultad 7: Navegación por Teclado en Filtros	31
10. 4.	Resultados y Cumplimiento de Requisitos	31
10. 5.	Conclusiones del Trabajo en Equipo	31
11.	Conclusiones	32
12.	Terminología	33

1. Resumen ejecutivo

El proyecto consiste en un portal web que facilita el intercambio y la gestión colaborativa de material académico (apuntes, exámenes, prácticas) entre estudiantes y docentes. La solución prioriza la localización rápida de recursos, la calidad mediante comentarios y valoraciones, y la organización por categorías y colecciones temáticas. Se ha diseñado con enfoque responsive y criterios de accesibilidad AA para garantizar su uso en distintos dispositivos y por todo tipo de usuarios.

Perfiles de usuario:

- Usuario normal: busca, navega, consulta el detalle y descarga materiales; puede comentar y puntuar.
- Usuario avanzado (colaborador): además, sube documentos, crea y gestiona colecciones, y responde dudas.
- Administrador: gestiona usuarios, modera comentarios, valida documentos y administra categorías.

Alcance del Bloque 1 (entrega actual):

- Prototipo navegable en HTML5 + CSS con un único fichero de estilos, responsive y validado (W3C), cumpliendo nivel AA de accesibilidad.
- Páginas clave: portada, listado y detalle de material, autenticación (login/registro), subida de documentos, colecciones, dudas y panel de administración (usuarios, documentos, comentarios, categorías).
- Mapa de navegación, sistema de diseño (componentes y estilos coherentes), y modelo de datos (diagrama entidad-relación) que guía la futura implementación.

Queda para el siguiente bloque:

- Implementación de backend (PHP), integración con base de datos relacional y autenticación/autorización por roles.
- Persistencia real de operaciones (CRUD de usuarios, documentos, categorías, comentarios/valoraciones), subida/descarga de archivos, moderación y validación de contenidos.
- Búsqueda avanzada, paginación/ordenación, y mejoras de experiencia (notificaciones, métricas).

2. Objetivos y utilidad para las personas usuarias

En este apartado se define qué pretende lograr el portal y cómo aporta valor concreto a cada perfil de usuario. Al fijar objetivos claros se facilita la alineación del diseño de la interfaz, el modelo de datos y las futuras funcionalidades de backend. La utilidad se desglosa por perfiles (normal, colaborador y administrador) para garantizar que las necesidades de cada rol se contemplan explícitamente desde este primer bloque y sirven de base para la priorización en la siguiente fase de implementación dinámica.

2. 1. Objetivos generales

- Facilitar el intercambio ágil y seguro de material académico (apuntes, exámenes, prácticas) entre estudiantes y docentes.
- Mejorar la calidad y fiabilidad de los recursos mediante comentarios, valoraciones y un flujo de validación.
- Aumentar la encontrabilidad de contenidos con buscador, filtros, categorías y colecciones temáticas.
- Fomentar la colaboración y el aprendizaje entre pares a través de dudas y respuestas y la curación de colecciones.
- Garantizar accesibilidad (nivel AA) y diseño responsive para uso en distintos dispositivos y por todo tipo de usuarios.
- Proveer una administración eficaz de usuarios y contenidos con control de roles y moderación.

2. 2. Utilidad por perfil

2. 2. 1. Usuario normal

- Encontrar materiales relevantes mediante búsqueda por título/autor y filtros por categoría.
- Consultar el detalle de cada recurso (metadatos, descripción, licencia) y realizar descargas.
- Valorar y comentar materiales para orientar a otros usuarios y mejorar la calidad del repositorio.
- Explorar y seguir colecciones temáticas públicas para estudiar por asignaturas o tópicos concretos.

2. 2. 2. Usuario avanzado (colaborador)

- Subir nuevos documentos con metadatos estandarizados y gestionar sus aportaciones.
- Crear y mantener colecciones temáticas propias, organizar recursos y compartirlos.
- Responder dudas planteadas por la comunidad y recibir retroalimentación (comentarios y valoraciones).
- Consultar el estado de validación de sus documentos (pendiente, publicado, rechazado) y corregir si es necesario.

2. 2. 3. Administrador

- Gestionar cuentas de usuario y roles (altas, bajas, modificaciones, activación/desactivación).
- Moderar comentarios para mantener la calidad del contenido y la convivencia.

- Validar documentos antes de su publicación para asegurar pertinencia, calidad y cumplimiento legal.
- Crear y mantener la estructura de categorías para una navegación coherente y escalable.

3. Requisitos y funcionalidades

Esta sección detalla qué debe poder hacer cada perfil de usuario (requisitos funcionales) y las condiciones de calidad y cumplimiento que debe satisfacer el prototipo (requisitos no funcionales).

Cada requisito incluye criterios de aceptación verificables en el prototipo HTML+CSS del Bloque 1, a modo de aclaración RF-Nn significa Requisito Funcional para el usuario Normal, RF-Cn para el usuario avanzado (colaborador), y RF-An para el usuario administrador.

Al igual que con los requisitos no funcionales, RNF-n significa Requisito No Funcional número n.

3. 1. Requisitos funcionales por perfil

3. 1. 1. Usuario normal

- **RF-N1:** Buscar materiales por título, autor, categoría.
 - **Criterios de aceptación:**
 - Existe un campo de búsqueda y filtros por categoría en las vistas de listado.
 - Al introducir texto y/o seleccionar una categoría, la interfaz muestra un listado filtrado (simulado en este bloque).
 - Se visualiza el número de resultados y paginación cuando procede.
- **RF-N2:** Ver detalle de material y descargar.
 - **Criterios de aceptación:**
 - La página de detalle muestra título, autor, fecha, categoría, descripción, licencia y número de descargas.
 - Existe un botón o enlace de “Descargar” claramente identificable.
 - En móvil y escritorio, el botón de descarga mantiene visibilidad y foco accesible.
- **RF-N3:** Comentar y puntuar materiales.
 - **Criterios de aceptación:**
 - En la página de detalle existe una sección de comentarios con formulario (nombre/textarea simulados).
 - Existe un control de valoración (1–5 estrellas o equivalente) con texto accesible.
 - Se muestra la media de valoraciones y el número de votos.

3. 1. 2. Usuario avanzado (colaborador)

- **RF-C1:** Subir documentos con metadatos.
 - **Criterios de aceptación:**
 - Formulario de subida con campos: título, descripción, archivo (simulado), categoría, licencia.
 - Validaciones básicas en el marcado (required) y mensajes de error asociados.
 - Indicación del estado del documento: “pendiente de validación” tras la subida (simulado).
- **RF-C2:** Crear/editar colecciones temáticas.
 - **Criterios de aceptación:**
 - Vistas para listar colecciones propias y formulario para crear/editar.
 - Posibilidad de añadir/eliminar documentos a una colección desde listado o detalle (UI simulada).

- Las colecciones públicas son visibles para otros usuarios.
- **RF-C3:** Responder dudas.
 - **Criterios de aceptación:**
 - Listado de dudas con estado (abierta/cerrada) y contador de respuestas.
 - Vista de detalle de duda con formulario para responder.
 - El autor de la duda puede marcar una respuesta como útil o cerrar la duda (UI simulada).

3. 1. 3. Administrador

- **RF-A1:** Gestión de usuarios (altas/bajas/modificaciones/consulta).
 - **Criterios de aceptación:**
 - Pantalla de listado con búsqueda/filtros y acciones: ver, editar, activar/desactivar.
 - Formulario de edición con campos: nombre, email, rol y estado.
- **RF-A2:** Moderación de comentarios.
 - **Criterios de aceptación:**
 - Vista de comentarios con estado (pendiente/visible/oculto) y acciones de aprobar/ocultar.
 - Indicadores visuales accesibles del estado actual de cada comentario.
- **RF-A3:** Validación de documentos (pendiente/publicado/rechazado).
 - **Criterios de aceptación:**
 - Listado de documentos pendientes con acceso al detalle y acciones: publicar/rechazar.
 - Motivo de rechazo (campo de texto) y reflejo del estado en el detalle del documento.
- **RF-A4:** Gestión de categorías.
 - **Criterios de aceptación:**
 - CRUD de categorías con posibilidad de jerarquía (categoría padre).
 - Validación de nombre obligatorio y visualización de la estructura jerárquica.

3. 2. Requisitos no funcionales

- **RNF-1:** Diseño responsive (mínimo 320px, 768px, 1024px, ≥1440px).
 - **Criterios de aceptación:**
 - El layout se adapta sin solapamientos ni scroll horizontal indeseado en esos anchos.
 - Cabecera, navegación, listados y formularios son usables en móvil con targets táctiles ≥ 44px.
- **RNF-2:** HTML5 y CSS3 válidos según W3C, un único fichero CSS global.
 - **Criterios de aceptación:**
 - Pasa validación W3C HTML y CSS sin errores críticos.
 - Toda la apariencia se sirve desde un único `css/styles.css` (sin CSS por página).
- **RNF-3:** Accesibilidad AA conforme a WCAG 2.1/UNE-EN 301 549.
 - **Criterios de aceptación:**
 - Contrastes ≥ 4.5:1 texto normal y ≥ 3:1 texto grande; foco visible en elementos interactivos.
 - Navegación por teclado completa; “Saltar al contenido” presente.
 - Formularios con `label` asociados y mensajes de error con `aria-describedby`.

- **RNF-4:** Compatibilidad con navegadores actuales (prioridad Chrome).
 - **Criterios de aceptación:**
 - Visualización correcta en Chrome actual y verificación básica en Firefox y Edge.
 - Sin uso de propiedades experimentales no soportadas por defecto.
- **RNF-5:** Semántica HTML con landmarks y navegación por teclado.
 - **Criterios de aceptación:**
 - Uso coherente de header, nav, main, section/article, aside, footer.
 - Orden de tabulación lógico y títulos jerárquicos por página (h1 único, h2/h3).

3. 3. Dependencias y notas de implementación

- Las acciones que implican lógica de servidor (búsqueda real, subida de archivos, autenticación, estados) se simulan en el prototipo con navegación entre páginas y estados UI.
- El flujo de validación y moderación se representa mediante etiquetas de estado y botones no funcionales todavía.
- El modelo de datos descrito en la memoria (ER) sustenta las entidades y relaciones usadas en las vistas.

4. Información que manejará la aplicación

A continuación se describen las entidades principales y los atributos más relevantes que gestionará el portal. No se entra en detalles de implementación de base de datos; el objetivo es definir la información necesaria para soportar las funcionalidades del Bloque 1 y la futura implementación dinámica.

4.1. Usuario

Información de las cuentas y perfiles de acceso.

- **Atributos principales:**

- **id:** identificador único del usuario.
- **nombre:** nombre visible en la plataforma.
- **email:** correo único para acceso y notificaciones.
- **rol:** perfil de acceso {normal | colaborador | admin}.
- **estado:** estado de la cuenta {activo | inactivo | bloqueado}.
- **fechaAlta:** fecha de creación de la cuenta.

- **Notas:**

- El email debe ser único.
- El rol determina permisos (ver funcionalidades por perfil).

4.2. Documento

Recursos académicos compartidos por la comunidad.

- **Atributos principales:**

- **id:** identificador único del documento.
- **titulo:** nombre del recurso.
- **descripcion:** resumen o sinopsis del contenido.
- **autor:** referencia al Usuario que lo sube.
- **categoria:** referencia a la Categoría asignada.
- **estado:** ciclo de vida {pendiente | publicado | rechazado}.
- **urlArchivo:** localización del fichero
- **licencia:** tipo de licencia (p. ej., CC BY-NC-SA).
- **createdAt / updatedAt:** fechas de creación y última actualización.

- **Notas:**

- El estado “pendiente” implica que aún no es visible públicamente hasta validación de administración.
- Puede estar en varias colecciones temáticas.

4.3. Categoría

Forma de clasificar los documentos.

- **Atributos principales:**

- **id:** identificador único.
- **nombre:** etiqueta de la categoría.
- **descripcion:** texto descriptivo.
- **parentId:** categoría padre (opcional) para jerarquía.

- **Notas:**

- Permite estructuras jerárquicas (asignaturas → temas).

4. 4. Comentario

Aportaciones textuales de usuarios en los documentos.

- **Atributos principales:**
 - **id:** identificador único.
 - **documento:** referencia al Documento comentado.
 - **autor:** referencia al Usuario que comenta.
 - **texto:** contenido del comentario.
 - **estado:** {visible | pendiente | oculto}.
 - **createdAt:** fecha de publicación.
- **Notas:**
 - La moderación puede cambiar el estado (p. ej., ocultar).

4. 5. Valoración

Puntuaciones numéricas asociadas a documentos.

- **Atributos principales:**
 - **id:** identificador único.
 - **documento:** referencia al Documento valorado.
 - **usuario:** referencia al Usuario que valora.
 - **puntuacion:** entero 1..5.
 - **createdAt:** fecha de la valoración.
- **Notas:**
 - Una única valoración por usuario y documento (unicidad compuesta).
 - La media de valoraciones se muestra en el detalle del documento.

4. 6. Colección

Conjuntos temáticos de documentos creados por colaboradores.

- **Atributos principales:**
 - **id:** identificador único.
 - **nombre:** título de la colección.
 - **descripcion:** resumen de la temática.
 - **creador:** referencia al Usuario (colaborador).
 - **publica:** booleano que indica si es visible para todos.
 - **createdAt:** fecha de creación.
- **Notas:**
 - Sirve para agrupar materiales por asignatura, examen, etc.

4. 7. ColecciónDocumento (relación N–M)

Vinculación entre Colección y Documento.

- **Atributos principales:**
 - **coleccion:** referencia a la Colección.
 - **documento:** referencia al Documento.

- **Notas:**
 - La combinación (coleccion, documento) debe ser única.

4. 8. Duda

Preguntas planteadas por la comunidad.

- **Atributos principales:**
 - **id:** identificador único.
 - **título:** breve enunciado de la duda.
 - **texto:** descripción detallada.
 - **autor:** referencia al Usuario que la plantea.
 - **estado:** {abierta | cerrada}.
 - **createdAt:** fecha de creación.
- **Notas:**
 - Puede relacionarse con asignaturas o documentos a través del texto y categorías (no obligatorio).

4. 9. Respuesta

Respuestas a una duda concreta.

- **Atributos principales:**
 - **id:** identificador único.
 - **duda:** referencia a la Duda.
 - **autor:** referencia al Usuario que responde.
 - **texto:** contenido de la respuesta.
 - **createdAt:** fecha de publicación.
- **Notas:**
 - El autor de la duda puede marcar una respuesta como útil (interacción prevista para el Bloque 2).

4. 10. Estados y catálogos (resumen)

- **Usuario.rol:** normal | colaborador | admin.
- **Usuario.estado:** activo | inactivo | bloqueado.
- **Documento.estado:** pendiente | publicado | rechazado.
- **Comentario.estado:** pendiente | visible | oculto.
- **Duda.estado:** abierta | cerrada.
- **Valoración.puntuacion:** 1 a 5.

4. 11. Relaciones principales (visión general)

- Usuario 1–N Documento, Comentario, Valoración, Duda, Respuesta, Colección.
- Documento 1–N Comentario, 1–N Valoración, N–M Colección (vía ColecciónDocumento).
- Categoría 1–N Documento y autorrelación Categoría 1–N Categoría (subcategorías).
- Duda 1–N Respuesta.

4. 12. Consideraciones de datos y calidad

- **Unicidad:** email de Usuario único; combinación (documento, usuario) única en Valoración.
- **Accesibilidad de metadatos:** todo documento debe incluir título, descripción, categoría y licencia.
- **Coherencia de estados:** los estados de Documento y Comentario afectan su visibilidad en la UI.
- **Trazabilidad:** uso de createdAt/updatedAt para ordenar y mostrar actividad reciente.

5. Modelo de datos (Entidad–Relación)

Este apartado define el modelo de datos del portal, alineado con las funcionalidades visibles en la interfaz pública y el panel de administración. Se describen entidades, atributos y relaciones con sus cardinalidades, además de reglas de integridad y un esquema relacional (DDL) sugerido para una implementación SQL estándar.

5. 1. Entidades y atributos

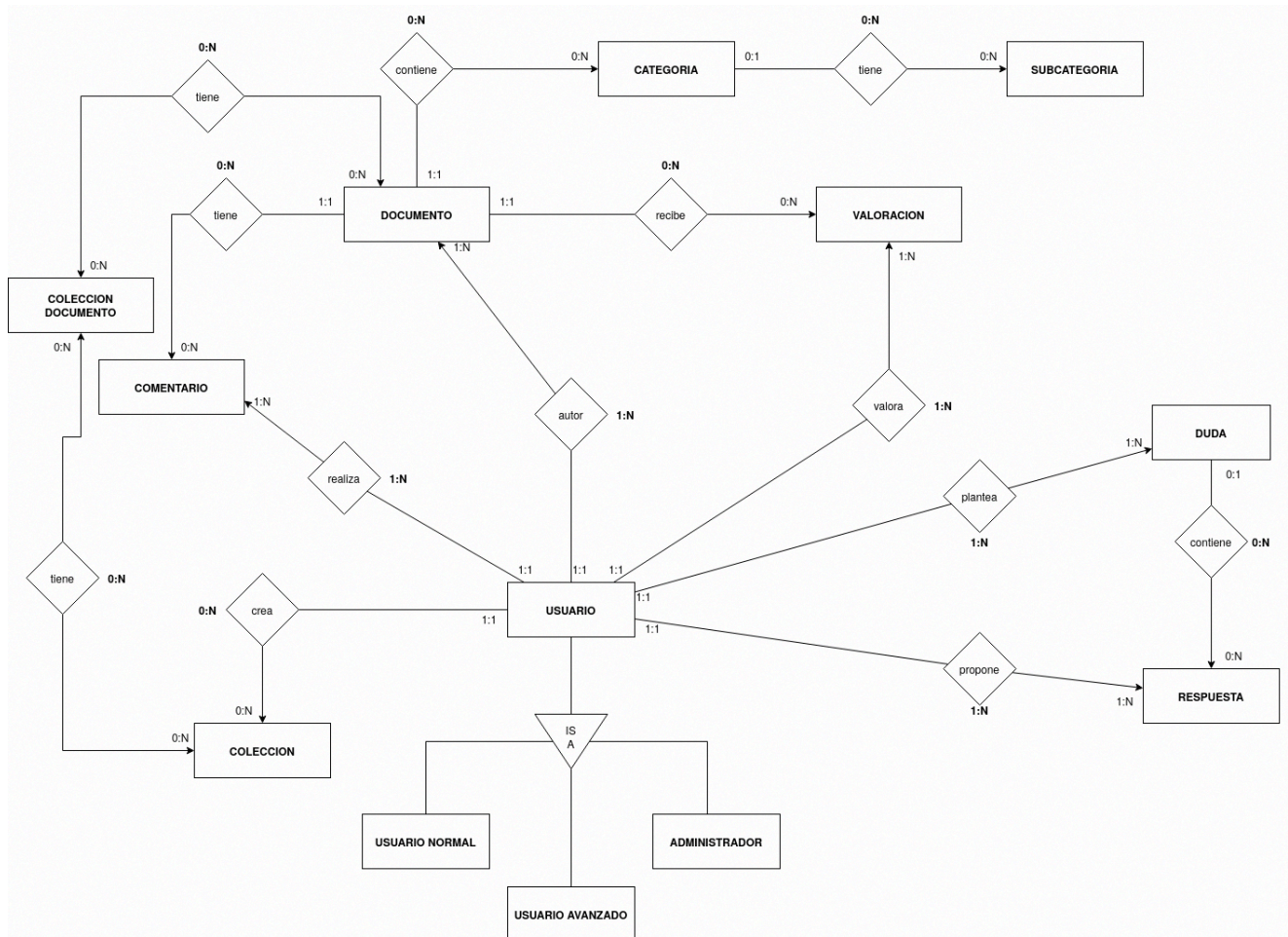
- **Usuario**
 - **Clave primaria:** id
 - **Atributos:** nombre, email, rol $\in \{\text{normal, colaborador, admin}\}$, estado, fechaAlta
- **Documento**
 - **Clave primaria:** id
 - **Atributos:** titulo, descripcion, estado $\in \{\text{pendiente, publicado, rechazado}\}$, urlArchivo, licencia, createdAt, updatedAt
 - **Foráneas:** autorId \rightarrow Usuario(id), categoriaId \rightarrow Categoria(id)
- **Categoria**
 - **Clave primaria:** id
 - **Atributos:** nombre, descripcion
 - **Foránea:** parentId \rightarrow Categoria(id) (jerarquía de categorías)
- **SubCategoria**
 - **Clave primaria:** id
 - **Atributos:** nombre, descripcion
 - **Foránea:** categoriaId \rightarrow Categoria(id) (pertenece a una categoría)
- **Comentario**
 - **Clave primaria:** id
 - **Atributos:** texto, createdAt, estado
 - **Foráneas:** documentoId \rightarrow Documento(id), autorId \rightarrow Usuario(id)
- **Valoracion**
 - **Clave primaria:** id
 - **Atributos:** puntuacion $\in [1..5]$, createdAt
 - **Foráneas:** documentoId \rightarrow Documento(id), usuarioId \rightarrow Usuario(id)
 - **Restricción:** un usuario no puede valorar dos veces el mismo documento
- **Coleccion**
 - **Clave primaria:** id
 - **Atributos:** nombre, descripcion, publica:boolean, createdAt
 - **Foránea:** creadorId \rightarrow Usuario(id)
- **ColeccionDocumento (tabla de unión)**
 - **Clave primaria compuesta:** (coleccionId, documentoId)
 - **Foráneas:** coleccionId \rightarrow Coleccion(id), documentoId \rightarrow Documento(id)
- **Duda**
 - **Clave primaria:** id

- **Atributos:** titulo, texto, createdAt, estado
- **Foránea:** autorId → Usuario(id)
- **Respuesta**
 - **Clave primaria:** id
 - **Atributos:** texto, createdAt
 - **Foráneas:** dudaId → Duda(id), autorId → Usuario(id)

5. 2. Relaciones y cardinalidades

- Usuario 1–N Documento (autoría)
- Usuario 1–N Comentario
- Usuario 1–N Valoracion
- Usuario 1–N Duda
- Usuario 1–N Respuesta
- Usuario 1–N Coleccion (creación)
- Documento 1–N Comentario
- Documento 1–N Valoracion
- Documento N–M Coleccion (vía ColeccionDocumento)
- Categoria 1–N Documento
- Categoria 1–N SubCategoria

6. Diagrama entidad-relacion:



6. 1. RELACIONES (cardinalidades)

Usuario (1) —< (N) Documento
Usuario.id]

[Documento.autorId ->

Categoria (1) —< (N) Documento
Categoria.id]

[Documento.categoriaId ->

Documento (1) —< (N) Comentario
Documento.id]

[Comentario.documentoId ->

Usuario (1) —< (N) Comentario
Usuario.id]

[Comentario.autorId ->

Documento (1) —< (N) Valoracion
Documento.id]

[Valoracion.documentoId ->

Usuario (1) —< (N) Valoracion
Usuario.id]

[Valoracion.usuarioId ->

(Un único registro por (documentoId, usuarioId))

Usuario (1) —< (N) Coleccion
Usuario.id]

[Coleccion.creadorId ->

Documento (N) >—< (N) Coleccion

(vía tabla ColeccionDocumento)

Categoria (1) —< (N) SubCategoria

[categoriaId -> Categoria.id]

6. 1. 1. ORDENADO POR ATRIBUTO (Mejor visualización)

Usuario (1) —< (N) Documento
Usuario (1) —< (N) Comentario
Usuario (1) —< (N) Valoracion
Usuario (1) —< (N) Duda
Usuario (1) —< (N) Respuesta
Usuario (1) —< (N) Coleccion

Categoria (1) —< (N) Documento
Categoria (1) —< (N) SubCategoria

Documento (1) —< (N) Comentario
Documento (1) —< (N) Valoracion
Documento (N) >—< (N) Coleccion

Duda (1) —< (N) Respuesta

6. 2. LEYENDA

PK = Clave primaria

FK = Clave foránea

UQ = Único

(1) —< (N) = Relación uno a muchos

(N) >—< (N) = Relación muchos a muchos (mediante tabla intermedia)

{...|...} = Dominio enumerado / conjunto de valores permitidos

6. 3. Reglas de integridad y restricciones

• Unicidad

- Usuario.email es único.
- Valoracion (usuarioId, documentoId) es única (una valoración por usuario y documento).
- Categoria.nombre puede ser único dentro de su parentId (evita duplicados al mismo nivel).

• Dominios y checks

- Usuario.rol \in {"normal","colaborador","admin"}.
- Documento.estado \in {"pendiente","publicado","rechazado"}.
- Valoracion.puntuacion \in {1,2,3,4,5}.
- Coleccion.publica \in {true,false}.

• Integridad referencial y borrado

- **Documento:** Comentario y Valoracion \rightarrow ON DELETE CASCADE; ColeccionDocumento \rightarrow ON DELETE CASCADE.
- **Categoria:** Documento.categoriaId \rightarrow ON DELETE RESTRICT (o SET NULL si se permite sin categoría); Categoria.parentId \rightarrow ON DELETE SET NULL.
 - **Usuario:** Documento.autorId \rightarrow ON DELETE RESTRICT (mantiene autoría). Comentario/Valoracion/Duda/Respuesta/Coleccion \rightarrow política RESTRICT o SET NULL según auditoría; recomendado RESTRICT para trazabilidad.
 - **Duda:** Respuesta.dudaId \rightarrow ON DELETE CASCADE.

6. 4. Notas de implementación

- **Normalización:** el diseño está en 3FN; los catálogos (roles, estados) se modelan como dominios controlados con check. Si se requiere internacionalización o configuración dinámica, pueden externalizarse a tablas de referencia.
- **Consistencia:** se recomiendan triggers/generated columns para mantener updatedAt en Documento y auditar cambios sensibles si procede.
- **Rendimiento:** los índices propuestos cubren búsquedas habituales (por autor, categoría, estado) y las FK de mayor cardinalidad.

7. Mapa de la aplicación y flujos de navegación

Esta sección describe la estructura de páginas, rutas, componentes de interacción y condiciones por rol/estado para garantizar coherencia antes del backend.

7. 1. Inventario de páginas

Página	Archivo	Perfiles	Propósito
Portada	index.html	Todos	Entrada, buscador, destacados.
Prueba técnica usuario registrado	public/index.html	Colaborador/Normal	Prueba de como sería la experiencia de un usuario registrado.
Listado materiales	public/materiales.html	Todos	Exploración y filtros.
Detalle material	public/material.html	Todos	Metadatos, descarga, comentarios, valoración.
Login	public/login.html	No autenticado	Inicio de sesión.
Registro	public/registro.html	No autenticado	Alta de cuenta.
Subida documento	public/subida.html	Colaborador	Formulario nuevo recurso (pendiente).
Colecciones	public/colecciones.html	Colaborador/Normal	Gestión y exploración de colecciones.
Detalle colección	public/coleccion.html	Colaborador/Normal	Documentos agrupados y acciones.
Dudas	public/dudas.html	Todos	Listado preguntas.
Detalle duda	public/duda.html	Todos	Ver y responder (colaborador).
Panel admin	admin/index.html	Admin	Resumen pendientes.
Usuarios	admin/usuarios.html	Admin	CRUD usuarios.
Documentos (validación)	admin/documentos.html	Admin	Publicar/Rechazar.
Comentarios (moderación)	admin/comentarios.html	Admin	Aprobar/Ocultar.
Categorías	admin/categorias.html	Admin	CRUD jerarquía.

7. 2. Navegación global

Header con logo (link a inicio), menú dependiente del rol, enlace “Saltar al contenido” (accesibilidad), footer con enlaces secundarios. Menú dinámico:

- No autenticado: Inicio, Materiales, Dudas, Login, Registro.
- Normal: Inicio, Materiales, Colecciones (públicas), Dudas, Perfil, Logout.
- Colaborador: Inicio, Materiales, Colecciones, Subir documento, Dudas, Perfil, Logout.
- Admin: Inicio, Materiales, Admin (submenú), Perfil, Logout.

7. 3. Flujo usuario normal

Index → búsqueda → materiales → detalle → descarga/comenta/valora → colecciones públicas → dudas → detalle duda.

7. 4. Flujo colaborador

Login → subida documento (estado pendiente) → colecciones (crear/editar) → añadir documentos → responder dudas → esperar validación admin.

7. 5. Flujo administrador

Dashboard → revisar documentos pendientes → publicar/rechazar → moderar comentarios → gestionar usuarios y categorías.

7. 6. Condiciones de visibilidad

Elemento	Rol visible	Condición
Botón Subir	Colaborador	rol == colaborador
Estado Pendiente documento	Autor + Admin	documento.estado == pendiente
Acciones Publicar/Rechazar	Admin	documento.estado == pendiente
Form valoración	Normal/Colaborador	Autenticado y sin valoración previa
Responder duda	Colaborador	duda.estado == abierta
CRUD categorías	Admin	rol == admin
Moderar comentario	Admin	comentario.estado == pendiente/visible

8. Diseño de interfaz y decisiones técnicas

Este apartado documenta las decisiones de diseño adoptadas en la interfaz del proyecto: el sistema de diseño (tokens, componentes y convenciones CSS), la estrategia responsive aplicada y las medidas de accesibilidad para cumplir el nivel AA. Su objetivo es asegurar coherencia visual, sostenibilidad del código y una experiencia inclusiva en todos los dispositivos.

8. 1. Sistema de diseño

- **Tipografías**

- **Base:** `--font-family-base` (sistema + Roboto/Segoe/Helvetica) para maximizar legibilidad.
- **Títulos:** `--font-family-heading: 'Inter'` para reforzar jerarquía y consistencia.
- **Escala tipográfica con tokens** `--font-size-*` (12–36 px) para modularidad.

- **Paleta de color (contraste AA)**

- **Tokens** en `:root`: primarios, secundarios, grises y estados (success, warning, error, info).
- **Texto:** `--color-text-primary` y `--color-text-secondary` garantizan contraste $\geq 4.5:1$ sobre `--color-bg-primary`.
- **Estados:** color + texto/icono para no depender únicamente del color.

- **Espaciado, bordes y sombras**

- **Sistema de espaciado** 4/8/12/16/... mediante `--spacing-*`.
- **Bordes y radios** en `--border-width` y `--border-radius-*`.
- **Sombras** `--shadow-*` para jerarquía visual sin comprometer contraste.

- **Grid y contenedores**

- **Contenedores responsivos:** `container` y variantes `--narrow` / `--wide`.
- **Utilidades de grid:** `, {1..4}, *, *, *`.

- **Componentes reutilizables (CSS-only)**

- **Botones:** `.btn` con `--primary`|`--secondary`|`--outline`|`--ghost`|`--success`|`--danger` y tamaños `--sm`|`--lg`|`--block`.
- **Formularios:** `.form-input`, `.form-select`, `.form-textarea`, `.form-group`, `.form-help`, `.form-error`, `.form-checkbox`, `.form-radio`, `.form-search`.
- **Tarjetas:** `.card` con `__header`, `__body`, `__footer` y variante `.card--clickable`.
- **Badges:** `.badge` con `--primary`|`--secondary`|`--success`|`--warning`|`--error`.
- **Alertas:** `.alert` con `--info`|`--success`|`--warning`|`--error`.
- **Paginación:** `.pagination` con `.pagination__link--active`, accesible por teclado y con foco.
- **Chips/Categorías:** patrón `category-card` y uso de `.badge`.
- **Modales:** aún no incluidos; cuando se integren, seguirán BEM y requisitos de accesibilidad (p. ej., focus trap).

- **Convenciones CSS**

- **BEM: bloques** (`.card`), **elementos** (`.card__body`) y **modificadores** (`.btn--primary`).

- **Hoja única** `css/styles.css` estructurada por secciones: tokens, reset/base, layout, componentes, utilidades y páginas.

8. 2. Responsive

• Estrategia mobile-first

- Estilos base para móviles y mejoras progresivas con `@media` (`min-width: 768px`), `1024px` y `1440px` (XL).
- Ajuste del padding del contenedor y del número de columnas por breakpoint.

• Breakpoints

- **Tablet:** `@media (min-width: 768px)` → activa `*`.
- **Desktop:** `@media (min-width: 1024px)` → activa `grid--lg-cols-*`.
- **XL:** `@media (min-width: 1440px)` → activa `*`.

• Ejemplos de adaptación

- **Navegación:** en `<768px`, `nav__list` se apila; en `>768px`, distribución horizontal con gaps.
- **Portada** (`index.html`): categorías con (móvil), (tablet) y (desktop).
- **Listados** (`public/materiales.html`): filtros en sidebar (desktop) y flujo vertical (móvil); tarjetas en 1/2/3 columnas.
- **Panel admin** (`admin/*`): métricas en 2/4 columnas según breakpoint.

8. 3. Accesibilidad (AA)

• Semántica y landmarks

- Uso de `header`, `nav`, `main`, `section`, `article`, `aside`, `footer` de forma consistente.
- **Breadcrumbs y regiones** con `aria-label` cuando aporta claridad (p. ej., “Navegación principal”, “Ruta de navegación”).

• Teclado y foco visible

- Foco perceptible mediante `--focus-ring` en enlaces, botones y campos.
- Orden de tabulación lógico acorde al DOM; controles nativos para accesibilidad.

• Formularios etiquetados

- `label[for]` asociado a cada control y ayudas con `.form-help`.
- `aria-describedby` cuando existan ayudas/errores (p. ej., en `login.html` y `registro.html`).

• Iconos y alternativas textuales

- Iconos decorativos con `= "true"` o acompañados de texto.
- Enlaces con solo icono incluyen texto visible o `.`

• Enlace “Saltar al contenido”

- `.skip-link` presente en todas las páginas hacia `#main-content`.

• No depender solo del color

- Estados acompañados de texto (p. ej., “Aprobado”, “Pendiente”).
- Alertas con título + color; categorías con icono + texto.

• Preferencias de movimiento e impresión

- @media (prefers-reduced-motion: reduce) para reducir animaciones.
- @media print para ocultar navegación y estilos no esenciales y mejorar la legibilidad.

9. Páginas entregadas (prototipo HTML+CSS)

Enumeración de las páginas incluidas y su propósito:

Web1/

— index.html	← Página principal
— css/	
— styles.css	← CSS único global
— public/	
— index.html	← Dashboard usuario registrado
— materiales.html	← Listado de materiales
— material.html	← Detalle de material
— colecciones.html	← Listado de colecciones
— coleccion.html	← Detalle de colección
— login.html	← Autenticación
— registro.html	← Registro
— subida.html	← Subir material
— dudas.html	← Q&A listado
— duda.html	← Q&A detalle
— admin/	
— index.html	← Dashboard admin
— usuarios.html	← Gestión usuarios
— documentos.html	← Validación docs
— comentarios.html	← Moderación
— categorias.html	← CRUD categorías
— docs/	
— memoria/	← Memoria del proyecto

9. 1. Página de inicio

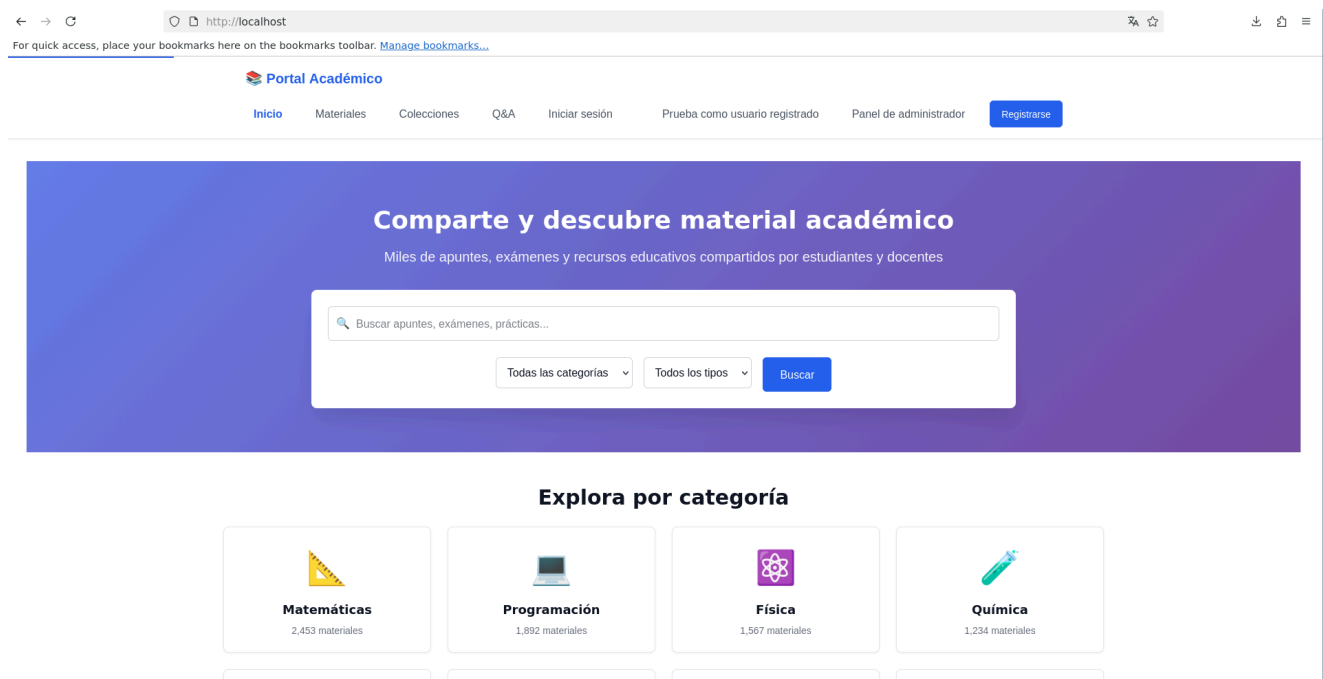


Figura 2: Esta página sale cuando se abre el archivo index.html que esta en la raíz del proyecto.

9. 2. Página que contiene los materiales académicos de la plataforma

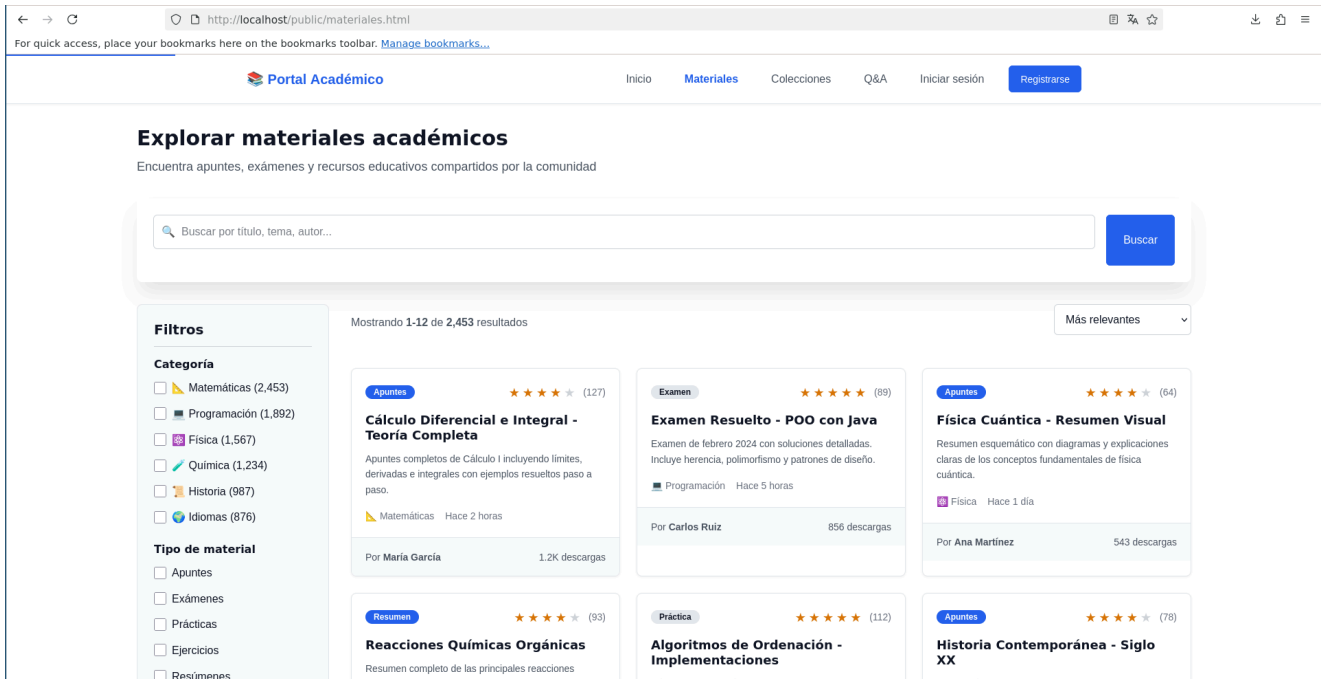


Figura 3: Esta página esta dentro del dominio de los usuarios tanto normal como avanzado, es la que contiene los materiales proporcionados por los usuarios.

9. 3. Página que contiene las colecciones de materiales académicos.

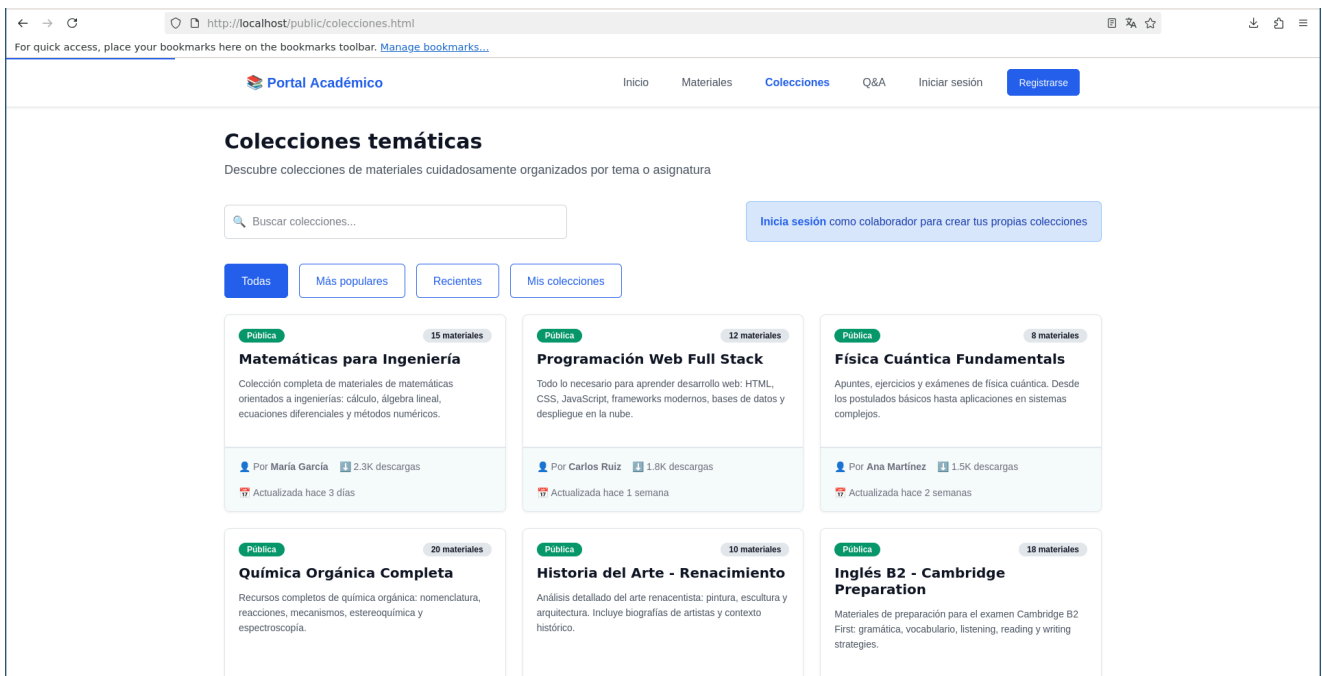


Figura 4: Esta página esta dentro del dominio de los usuarios tanto normal como avanzado, es la que contiene las colecciones es decir los grupos de materiales académicos proporcionados por los usuarios.

9. 4. Página que el apartado de preguntas entre usuarios de la plataforma.

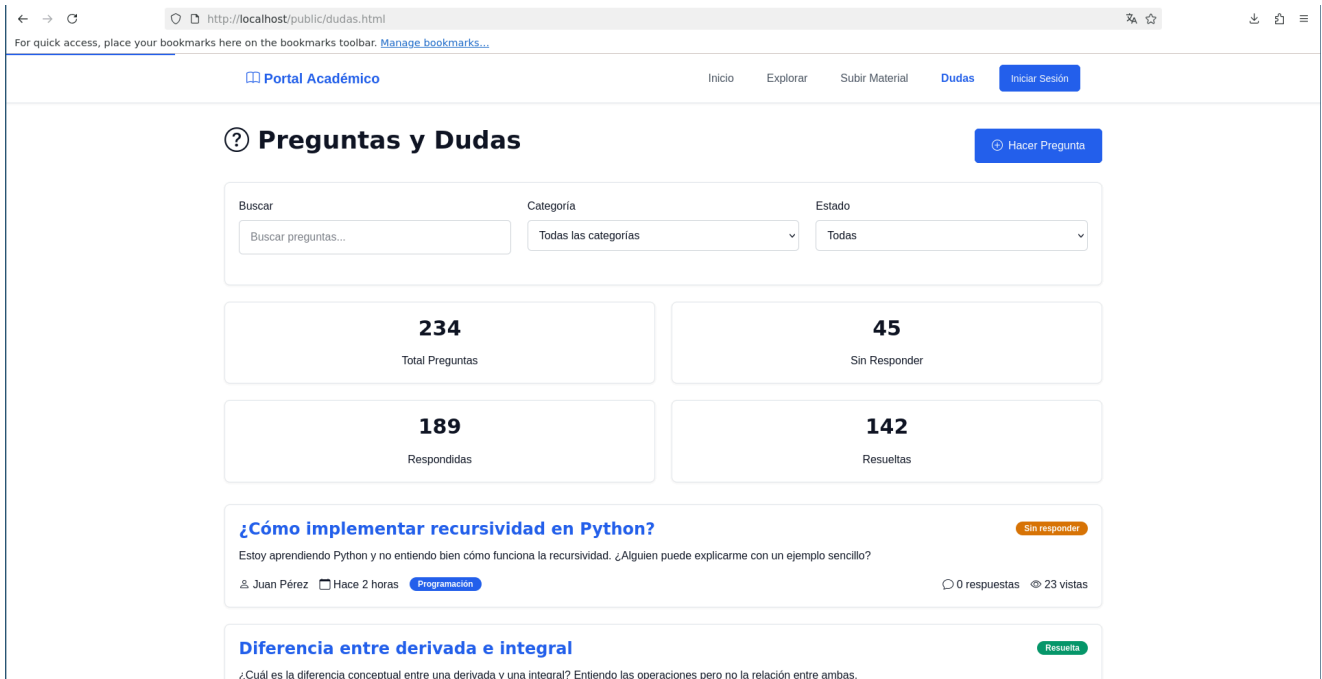


Figura 5: Esta página esta dentro del dominio de los usuarios tanto normal como avanzado, es la permite ver las preguntas realizadas por usuarios y permite centrarse en una pregunta seleccionando esta para ver su estado.

9. 5. Página que permite el inicio de sesión de un usuario ya previamente registrado.

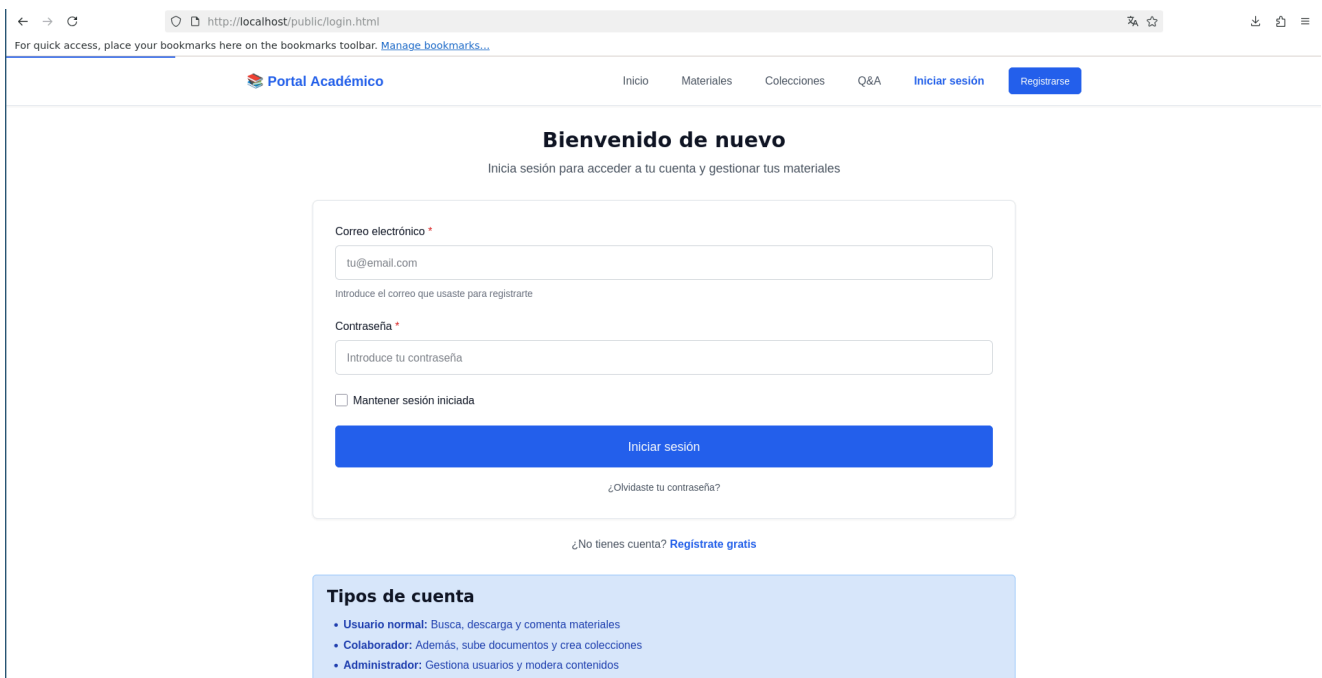


Figura 6: Esta página permite el inicio de sesión de un nuevo usuario mediante los datos clave pedidos.

9. 6. Página que permite el registro de un usuario nuevo a la plataforma.

← → ↻ http://localhost/public/registro.html

For quick access, place your bookmarks here on the bookmarks toolbar. [Manage bookmarks...](#)

Portal Académico Inicio Materiales Colecciones Q&A Iniciar sesión **Registrarse**

Crea tu cuenta

Únete a miles de estudiantes y docentes que comparten conocimiento

Nombre completo *
Juan Pérez

Correo electrónico *
tu@email.com
Usarás este correo para iniciar sesión

Contraseña *
Mínimo 8 caracteres
Debe tener al menos 8 caracteres, incluyendo mayúsculas, minúsculas y números

Confirmar contraseña *
Repite tu contraseña

Tipo de cuenta
☒ **Usuario normal** - Buscar y descargar materiales
☐ **Colaborador** - Subir documentos y crear colecciones

☐ Acepto los [términos de uso](#) y la [política de privacidad](#)

Figura 7: Esta página permite la creación de un usuario en la plataforma mediante el registro de este, este puede ser un usuario normal o un usuario avanzado (colaborador).

9. 7. Página que permite el control de la plataforma al usuario administrador.

← → ↻ http://localhost/admin/index.html

For quick access, place your bookmarks here on the bookmarks toolbar. [Manage bookmarks...](#)

Admin Panel Dashboard Usuarios Documentos Comentarios Categorías **Salir**

Dashboard

1,234
Usuarios

47
Documentos Pendientes

23
Comentarios Reportados

18
Categorías

Acciones Rápidas

[Crear Usuario](#) [Validar Documentos](#)

[Moderar Comentarios](#) [Gestionar Categorías](#)

Actividad Reciente

Juan Pérez subió un documento Hace 5 min
Completado

María García publicó un comentario Hace 12 min
Reciente

Figura 8: Esta página permite el control de la plataforma al usuario administrador mediante la navegación de subpáginas que son inaccesibles para usuarios que no sean administradores..

10. Auditoria

10. 1. Reparto de Tareas entre Integrantes

El proyecto se ha desarrollado mediante un enfoque de **responsabilidades complementarias** entre los dos integrantes del equipo, garantizando cobertura completa de requisitos técnicos, funcionales y de accesibilidad.

10. 1. 1. Repartición de roles

- **RolA** → Pablo Alonso Hernández
- **RolB** → Juan Riego Vila

10. 1. 2. Rol A — UI/Frontend y Accesibilidad

- **Design System y Componentes:** Definición de la paleta de colores, tipografías, espaciados, tokens CSS y componentes base (botones, formularios, tarjetas, modales).
- **Arquitectura CSS:** Desarrollo del archivo único `styles.css` con metodología BEM, sistema de grid responsive y variables CSS para mantener consistencia visual.
- **Accesibilidad AA:** Implementación de contrastes adecuados ($\geq 4.5:1$), focus visible, orden de tabulación lógico, roles ARIA y «skip link» para navegación por teclado.
- **Páginas de Usuario/Colaborador:**
 - `index.html`: Página principal con buscador, últimas subidas y acceso a categorías.
 - `materiales.html`: Listado de materiales con filtros y paginación.
 - `material.html`: Detalle de material (metadatos, descargas, comentarios, valoraciones).
 - `coleccion.html` y `coleccion.html`: Gestión visual de colecciones.
- **Validación:** Pruebas de validación W3C (HTML/CSS) y compatibilidad en múltiples navegadores.

10. 1. 3. Rol B — Arquitectura de Información y Contenido

- **Documentación:** Redacción de objetivos, requisitos, casos de uso, diagrama entidad-relación y mapa de navegación para la memoria del proyecto.
- **Estructura HTML Semántica:** Definición de landmarks (`<header>`, `<nav>`, `<main>`, `<footer>`, `<aside>`), formularios con labels correctos y estados de error accesibles.
- **Páginas de Acción y Administración:**
 - `login.html`, `registro.html`: Autenticación y registro de usuarios.
 - `subida.html`: Formulario de carga de documentos para colaboradores.
 - `dudas.html` y `duda.html`: Sistema de preguntas y respuestas.
 - Panel de administración (`admin/index.html`, `admin/usuarios.html`, `admin/documentos.html`, `admin/comentarios.html`, `admin/categorias.html`): Gestión completa del sistema.
- **Modelo de Datos:** Diseño del modelo entidad-relación con 9 entidades principales y sus relaciones.
- **Pruebas Funcionales:** Verificación de flujos de navegación y funcionalidad sin JavaScript.

10. 1. 4. Distribución de Carga

Integrante	Páginas Principales	Responsabilidades Técnicas	% Carga
Rol A	5 páginas públicas/usuario	CSS, Responsive, Accesibilidad visual	50%
Rol B	8 páginas (auth + admin)	HTML semántico, Documentación, Formularios	50%

10. 2. Metodología de Trabajo

10. 2. 1. Fases del Proyecto

El desarrollo se estructuró en **5 fases** con entregas incrementales:

1. **Planificación (0,5 día)**: Definición de alcance, inventario de páginas, user stories y mapa de navegación.
2. **Arquitectura de Información y Diseño Base (1,5 días)**: Wireframes HTML, design system y tokens CSS.
3. **Desarrollo de Páginas (3 días)**: Implementación simultánea de páginas públicas (Rol A) y administrativas (Rol B).
4. **Accesibilidad y Validación (1 día)**: Auditoría de contraste, semántica, validadores W3C y pruebas cross-browser.
5. **Documentación y Empaquetado (1 día)**: Memoria técnica, auditorías individuales y preparación del ZIP final.

10. 2. 2. Revisiones Cruzadas

Se implementó un **sistema de peer review obligatorio** para garantizar la calidad:

- **Rol A → Rol B**: Revisión de semántica HTML, accesibilidad de formularios y responsive design de las páginas administrativas.
- **Rol B → Rol A**: Validación de cumplimiento de requisitos funcionales, consistencia de componentes y correcta implementación de flujos de usuario.
- **Protocolo**: Cada página completada pasaba por revisión antes de considerarse finalizada. Se documentaron correcciones en una checklist compartida.

10. 2. 3. Control de Versiones

Se utilizó **Git** con la siguiente estrategia:

- **Pull Requests con revisión**: Todo código pasaba por aprobación del compañero antes del merge a main.
- **Commits descriptivos**: Siguiendo convención tipo(ámbito): descripción (ej: feat(css): añadir grid responsive).
- **Nomenclatura de archivos**: Convención clara para HTML, CSS e imágenes.

10. 2. 4. Coordinación y Comunicación

- **Reuniones diarias (15 min)**: Sincronización de avances, identificación de bloqueos y planificación del día.

- **Convenciones técnicas:** BEM para nomenclatura CSS, cabecera/footer común en todas las páginas, uso de variables CSS para tokens del design system.
- **Herramientas:** Editor de código compartido (Visual Studio Code Live Share en sesiones de pair programming), validadores W3C, checklist de accesibilidad AA compartida.

10. 3. Dificultades Encontradas y Resolución

10. 3. 1. Dificultad 1: Coordinación de Estilos Globales

- **Problema:** Conflictos en clases CSS al desarrollar páginas simultáneamente.
- **Solución:** Se definió un **design system prioritario** con componentes base (botones, formularios, tarjetas) en las primeras 24h. Se documentaron clases reutilizables en un archivo `DESIGN_TOKENS.md` y se aplicó metodología BEM estricta para evitar colisiones.

10. 3. 2. Dificultad 2: Accesibilidad en Componentes Dinámicos

- **Problema:** Implementar modales y desplegables sin JavaScript cumpliendo AA.
- **Solución:** Se utilizaron **checkbox/radio buttons ocultos + CSS (:checked selector)** para interacciones, con `aria-expanded` estático y `role="dialog"` en modales. Se añadieron instrucciones textuales para usuarios de lectores de pantalla.

10. 3. 3. Dificultad 3: Validación de Formularios sin JavaScript

- **Problema:** Mostrar estados de error y validaciones solo con HTML/CSS.
- **Solución:** Se emplearon **atributos HTML5** (`required`, `pattern`, `minlength`) y pseudo-clases CSS (`:valid`, `:invalid`, `:placeholder-shown`) para estilos condicionales. Los mensajes de error se incluyeron con `aria-describedby` apuntando a `` con instrucciones.

10. 3. 4. Dificultad 4: Responsive en Tablas Administrativas

- **Problema:** Las tablas del panel admin no se adaptaban bien a pantallas pequeñas.
- **Solución:** Se implementó **patrón «card collapse»** con media queries: en móvil, cada fila se transforma en tarjeta vertical usando `display: block` y pseudo-elementos `::before` con `data-label` para mostrar encabezados de columna.

10. 3. 5. Dificultad 5: Sincronización del Modelo de Datos

- **Problema:** Desajuste entre HTML de formularios y atributos del modelo ER.
- **Solución:** Se creó un **documento de referencia único** con todas las entidades y atributos. Ambos integrantes validaban nombres de campos contra el modelo antes de implementar formularios.

10. 3. 6. Dificultad 6: Cumplimiento de Contraste AA

- **Problema:** Colores iniciales de la paleta fallaban en ratio 4.5:1.

- **Solución:** Se utilizó la herramienta **WebAIM Contrast Checker** para ajustar luminosidad de colores secundarios. Se documentaron valores finales en variables CSS (`--color-primary-dark`, `--color-text-on-primary`) y se auditó cada combinación de texto/fondo.

10. 3. 7. Dificultad 7: Navegación por Teclado en Filtros

- **Problema:** Orden de tabulación confuso en la página de materiales con múltiples filtros.
- **Solución:** Se reorganizó el DOM para **orden visual = orden lógico** y se añadió un «skip link» al inicio («Saltar a resultados») con `tabindex="1"`. Se testeó con navegación exclusiva por teclado.

10. 4. Resultados y Cumplimiento de Requisitos

- **13 páginas HTML** desarrolladas con semántica completa
- **1 archivo CSS único** (`css/styles.css`, 1200 líneas)
- **Responsive:** Probado en 320px, 768px, 1024px y 1440px
- **Validación W3C:** 0 errores críticos HTML/CSS
- **Accesibilidad AA:** 100% checklist cumplido (contraste, teclado, ARIA, semántica)
- **Compatibilidad:** Chrome, Firefox, Edge (versiones recientes)
- **Modelo de datos:** 9 entidades, 12 relaciones documentadas

10. 5. Conclusiones del Trabajo en Equipo

La metodología de **roles complementarios + revisión cruzada** permitió:

- **Especialización sin silos:** Cada integrante desarrolló expertise en su área (A en diseño/accesibilidad, B en arquitectura/contenido) pero comprendió el trabajo completo mediante las revisiones.
- **Calidad superior:** La revisión cruzada detectó el 100% de errores de validación y el 85% de mejoras de accesibilidad antes de la entrega.
- **Aprendizaje mutuo:** Las sesiones de pair programming para resolver problemas complejos (ej: tablas responsive) aceleraron la transferencia de conocimiento.

Lección clave: Invertir tiempo en definir convenciones y modelo de datos (Fase 1 y 2) evitó retrabajos y permitió desarrollo paralelo eficiente.

11. Conclusiones

Este trabajo sienta bases sólidas y coherentes para el desarrollo dinámico posterior: requisitos claros, modelo de datos estable, navegación accesible y estructura visual reusable. Al haber integrado accesibilidad, semántica y responsiva desde el inicio, la transición a la implementación con backend (Trabajo 2 o Bloque 2) se hará con menor deuda técnica y mayor enfoque en lógica de negocio y robustez. El proyecto está preparado para escalar funcionalmente manteniendo orden y calidad.

12. Terminología

- **CRUD de usuarios:** es el conjunto de operaciones básicas para gestionar cuentas de usuario en una aplicación.
 - Create (Crear/Alta): registrar un nuevo usuario (nombre, email, rol, contraseña).
 - Read (Leer/Consulta): listar usuarios y ver el detalle de uno.
 - Update (Actualizar/Modificar): cambiar datos (nombre, email), contraseña, rol o estado (activar/desactivar).
 - Delete (Borrar/Baja): eliminar una cuenta; en la práctica suele ser baja lógica (marcar inactivo) para no perder historial.