

**Universidad Espíritu Santo**

**Modalidad en Línea**

Ingeniería en Ciencias de la Computación

**PLATAFORMA DE TRUEQUE DE LIBROS ENTRE ESTUDIANTES**

Estudiante : Alex Mendoza Morante

Oscar Vallejo Mino

Bryan Cuenca Guerrero

Materia : Diseño de Software

Docente : MTI Vanessa Jurado

Fecha de Entrega : 18 de mayo del 2025

Contenido

[**PROPÓSITO DEL DOCUMENTO** 1](#_Toc196668559)

[**COMPONENTES PRINCIPALES** 1](#_Toc196668560)

[**ESTRUCTURA GENERAL DE COMPONENTES** 1](#_Toc196668561)

[**DETALLE DE COMPONENTES** 2](#_Toc196668562)

[**BASE DE DATOS** 3](#_Toc196668563)

[**ADMINISTRACIÓN DEL SISTEMA** 3](#_Toc196668564)

[**CONSIDERACIONES DE SEGURIDAD** 3](#_Toc196668565)

[**PRINCIPIOS DE DISEÑO APLICADOS** 4](#_Toc196668566)

# **PROPÓSITO DEL DOCUMENTO**

Este documento describe los componentes principales (o módulos) que conforman la estructura lógica y técnica del sistema. Se busca garantizar la organización modular, facilitar el desarrollo colaborativo, y permitir una escalabilidad limpia en futuras fases. La arquitectura se basa en el patrón Modelo-Template-Vista (MTV) de Django.

# **COMPONENTES PRINCIPALES**

* **Usuario Web** (cliente externo que interactúa vía navegador).
* **Servidor Django** (coordinador principal, framework backend).
* **Apps Django** (cada funcionalidad empaquetada como componente):
  + usuarios (registro/login)
  + libros (publicar/buscar libros)
  + catálogos (categorías, estados, tipos contacto)
  + perfiles (roles de usuarios)
  + contacto (mensajes entre usuarios)
* **Base de Datos MySQL** (almacenamiento permanente).

Interfaz de usuario gráfica, Diagrama

El contenido generado por IA puede ser incorrecto.

# **ESTRUCTURA GENERAL DE COMPONENTES**

|  |  |
| --- | --- |
| Módulo | Descripción |
| Usuarios | Gestión de autenticación y administración de usuarios registrados. |
| Libros | Publicación, edición, eliminación y visualización de libros para intercambio. |
| Búsqueda | Búsqueda de libros por criterios. |
| Contacto | Envío de mensajes entre estudiantes. |
| Catálogos | Administración de categorías, estados de libros y tipos de contacto. |
| Perfiles | Definición de tipos de usuario (administrador, estudiante). Solo gestionable desde el panel de administración. |
| Moderación *(futuro)* | Revisión y validación de publicaciones antes de ser públicas. |
| Reportes *(futuro)* | Estadísticas básicas del sistema. |

.

# **DETALLE DE COMPONENTES**

**Catalogos**

* **Categorías de libros:** Historia, Geografía, Ciencias, Matemáticas, etc.
* **Estados de libros:** Nuevo, Buen estado, Aceptable, etc.
* **Tipos de contacto:** Formulario, Correo electrónico, WhatsApp.
* Accesible y modificable exclusivamente desde el **panel administrativo**.

**Perfiles**

* Definición de tipos de usuario:
  + **Administrador:** Accede a funciones de moderación y administración de catálogos.
  + **Estudiante registrado:** Accede a publicación, búsqueda y contacto de libros.
* Gestión interna a través del panel de administración (/admin/).
* No disponible para modificación desde la plataforma pública.

**Usuarios**

* Registro, inicio y cierre de sesión de estudiantes.
* Recuperación de contraseñas.
* Manejo de sesiones seguras.
* Asociación de cada usuario a un perfil de tipo de usuario

L**ibros**

* Creación de publicaciones de libros.
* Edición y eliminación de publicaciones propias.
* Asociación del libro a categorías y estados definidos en catálogos.

**Contacto**

* Sistema interno de mensajes para gestionar solicitudes de trueque.
* Asociación entre el libro y el interesado (Usuario).

**Busqueda**

* Búsqueda de libros por título, autor o categoría.
* Visualización de resultados filtrados dinámicamente.

**Moderacion *(futuro)***

* Validación de nuevas publicaciones de libros por un usuario administrador antes de hacerse públicas.

**Reportes *(futuro)***

* Estadísticas simples como:
  + Cantidad de libros publicados.
  + Categorías más activas.
  + Usuarios más participativos.

# **BASE DE DATOS**

* TipoContacto → id, descripción,estado
* Perfil → id, nombre del perfil (Administrador, Estudiante), Estado
* Contacto → id, Tipo de contato, Estado
* Usuario → id, nombre, correo, contraseña, perfil, Fecha Acceso,Aceptación de Políticas, Intentos de acceso, Status (FK hacia Perfil)
* Libro → id, título, autor, categoría (FK), descripción, estado (FK), usuario (FK)
* CategoriaLibro → id, nombre, estado
* EstadoLibro → id, descripción
* Peticiones → id, libro (FK), usuario\_interesado (FK), libro, fecha petición, Fecha Limite, Mensaje, respuesta,

# **ADMINISTRACIÓN DEL SISTEMA**

* **Usuarios** (estudiantes) gestionan su información pública básica.
* **Administradores** gestionan:
  + Catálogos (categorías, estados, tipos de contacto).
  + Perfiles de usuarios (sólo lectura asignada).
  + Publicaciones moderadas (futuro).
* **Django Admin Panel**: Gestión ágil y segura de los datos.

# **CONSIDERACIONES DE SEGURIDAD**

* Uso de Django Sessions para autenticación segura.
* Protecciones integradas: CSRF, ORM contra inyección SQL.
* Validación de datos en formularios de usuario.
* Gestión de contraseñas cifradas.
* Control de accesos basado en tipo de perfil (is\_staff, groups, o campo personalizado en Usuario).

# **PRINCIPIOS DE DISEÑO APLICADOS**

* **Separación de responsabilidades.**
* **Modularidad total entre apps.**
* **Escalabilidad horizontal:** fácil expansión a nuevos módulos.
* **Mantenibilidad:** código limpio, siguiendo buenas prácticas de Django.