



MANUAL TÉCNICO

Sistema de Registro y
Control de Tesis de la
Biblioteca de la
Universidad de Oriente



Mayo de 2025

Ricardo Álvarez

ÍNDICE GENERAL

ÍNDICE GENERAL	2
ÍNDICE DE FIGURAS	3
ÍNDICE DE TABLAS	4
DIAGRAMA CONCEPTUAL DEL SISTEMA	5
Diccionario de Datos	5
Diagrama Entidad — Relación	14
Diagrama de Clases	14
REQUERIMIENTOS DE SOFTWARE Y HARDWARE	15
Requerimientos de Software	15
Requerimientos de Hardware	16
ruta de UBICACIÓN	16
CONFIGURACIÓN	17
RUTINAS CLAVE	17
Conexión a la Base de Datos	17
Validación de Acceso al Sistema	18
Manejo de Rutas	19
LISTA DE ERRORES	21

ÍNDICE DE FIGURAS

Figura 1. Diagrama Entidad—Relación	14
Figura 2. Diagrama de Clases	15
Figura 3. Parámetros de Conexión a la Base de Datos en “.env”	17
Figura 4. Extracto del controlador "LoginController.php"	18
Figura 5. Rutas del Sistema (1/2)	19
Figura 6. Rutas del Sistema (2/2)	20

ÍNDICE DE TABLAS

Tabla 1. Diccionario de datos de la Entidad autores.....	5
Tabla 2. Diccionario de Datos de la Entidad autor_ficha	5
Tabla 3. Diccionario de Datos de la Entidad carreras	6
Tabla 4. Diccionario de Datos de la Entidad fichas.....	6
Tabla 5. Diccionario de Datos de la Entidad infopers	7
Tabla 6. Diccionario de Datos de la Entidad prestamos	8
Tabla 7. Diccionario de Datos de la Entidad prestatarios	9
Tabla 8. Diccionario de Datos de la Entidad users	10
Tabla 9. Diccionario de Datos de la Entidad permissions.....	10
Tabla 10. Diccionario de Datos de la Entidad roles	11
Tabla 11. Diccionario de Datos de la Entidad role_has_permissions	11
Tabla 12. Diccionario de Datos de la Entidad model_has_roles.....	11
Tabla 13. Diccionario de Datos de la Entidad preguntas_user	12
Tabla 14. Diccionario de Datos de la Entidad model_has_permissions.....	13
Tabla 15. Diccionario de Datos de la Entidad sessions	13
Tabla 16. Requerimientos de Software.....	16
Tabla 17. Requerimientos de Hardware	16

DIAGRAMA CONCEPTUAL DEL SISTEMA

Diccionario de Datos

En el diccionario de datos se realiza una descripción de las entidades o tablas presentes en el sistema, y cada uno de sus atributos correspondientes, como tipo, cotejamiento, nulo y extra. Las entidades del sistema se presentan a continuación:

Tabla 1. Diccionario de datos de la Entidad *autores*

Nombre de la Tabla		autores			
Descripción		Almacena la información de los autores			
Nombre	Tipo	Cotejamiento	Nulo	Extra	Descripción
id	bigint(20)		No	auto_increment	Identificador del autor
ci_autor	varchar(8)	utf8mb4_unico de_ci	No		Cédula del autor
nombre_autor	varchar(250)	utf8mb4_unico de_ci	No		Nombres del autor
apellido_autor	varchar(250)	utf8mb4_unico de_ci	No		Apellidos del autor
created_at	timestamp		Sí		Fecha de creación
updated_at	timestamp		Sí		Fecha de actualización

Fuente: Base de Datos del Sistema de Registro y Control de Tesis.

Tabla 2. Diccionario de Datos de la Entidad *autor_ficha*

Nombre de la Tabla		autor_ficha			
Descripción		Tabla pivote entre autores y fichas			
Nombre	Tipo	Cotejamiento	Nulo	Extra	Descripción
id	bigint(20)		No	auto_increment	Identificador de la relación
ficha_id	bigint(20)		No		Identificador de la ficha
autor_id	bigint(20)		No		Identificador del autor
created_at	timestamp		Sí		Fecha de creación
updated_at	timestamp		Sí		Fecha de actualización

Fuente: Base de Datos del Sistema de Registro y Control de Tesis.

Tabla 3. Diccionario de Datos de la Entidad carreras

Nombre de la Tabla		carreras			
Descripción		Contiene la información de las carreras registradas			
Nombre	Tipo	Cotejamiento	Nulo	Extra	Descripción
id	bigint(20)		No	auto_increment	Identificador de la carrera
nombre	varchar(255)	utf8mb4_unico de_ci	No		Nombre de la carrera
created_at	timestamp		Sí		Fecha de creación
updated_at	timestamp		Sí		Fecha de actualización

Fuente: Base de Datos del Sistema de Registro y Control de Tesis.

Tabla 4. Diccionario de Datos de la Entidad fichas

Nombre de la Tabla		fichas			
Descripción		Almacena las fichas bibliográficas de los trabajos			
Nombre	Tipo	Cotejamiento	Nulo	Extra	Descripción
id	bigint(20)		No	auto_increment	Identificador de la ficha
titulo	varchar(900)	utf8mb4_unico de_ci	No		Título del trabajo
fecha	date		No		Fecha de elaboración
carrera_id	bigint(20)		No		Carrera relacionada
resumen	text	utf8mb4_unico de_ci	No		Resumen del trabajo
created_at	timestamp		Sí		Fecha de creación
updated_at	timestamp		Sí		Fecha de actualización

Fuente: Base de Datos del Sistema de Registro y Control de Tesis.

Tabla 5. Diccionario de Datos de la Entidad infopers

Nombre de la Tabla		infopers			
Descripción		Almacena información personal asociada a los usuarios			
Nombre	Tipo	Cotejamiento	Nulo	Extra	Descripción
id	bigint(20)		No	auto_increment	Identificador de registro
ci_us	varchar(8)	utf8mb4_unico de_ci	No		Cédula del usuario
nombre	varchar(255)	utf8mb4_unico de_ci	No		Nombre del usuario
apellido	varchar(255)	utf8mb4_unico de_ci	No		Apellido del usuario
user_id	bigint(20)		No		Relación con la tabla de usuarios
created_at	timestamp		Sí		Fecha de creación
updated_at	timestamp		Sí		Fecha de actualización

Fuente: Base de Datos del Sistema de Registro y Control de Tesis.

Tabla 6. Diccionario de Datos de la Entidad prestamos

Nombre de la Tabla		prestamos			
Descripción		Registra los préstamos de las fichas en el sistema			
Nombre	Tipo	Cotejamiento	Nulo	Extra	Descripción
id	bigint(20)		No	auto_increment	Identificador del préstamo
ficha_id	bigint(20)		No		Ficha prestada
prestatario_id	bigint(20)		No		Prestatario asociado
fecha_prestito	date		No		Fecha del préstamo
fecha_devolucion	date		No		Fecha límite de devolución
fecha_entrega	date		Sí		Fecha real de entrega
estado	enum('prestado','devuelto')		No	default 'prestado'	Estado del préstamo
created_at	timestamp		Sí		Fecha de creación
updated_at	timestamp		Sí		Fecha de actualización

Fuente: Base de Datos del Sistema de Registro y Control de Tesis.

Tabla 7. Diccionario de Datos de la Entidad prestatarios

Nombre de la Tabla		prestatarios			
Descripción		Almacena los datos de quienes solicitan préstamos			
Nombre	Tipo	Cotejamiento	Nulo	Extra	Descripción
id	bigint(20)		No	auto_increment	Identificador del prestatario
ci_prestatario	varchar(8)	utf8mb4_unico de_ci	No		Cédula del prestatario
nombre_prestatario	varchar(255)	utf8mb4_unico de_ci	No		Nombre del prestatario
apellido_prestatario	varchar(255)	utf8mb4_unico de_ci	No		Apellido del prestatario
tlf_prestatario	varchar(11)	utf8mb4_unico de_ci	No		Teléfono del prestatario
created_at	timestamp		Sí		Fecha de creación
updated_at	timestamp		Sí		Fecha de actualización

Fuente: Base de Datos del Sistema de Registro y Control de Tesis.

Tabla 8. Diccionario de Datos de la Entidad users

Nombre de la Tabla		users			
Descripción		Información básica de los usuarios del sistema			
Nombre	Tipo	Cotejamiento	Nulo	Extra	Descripción
id	bigint(20)		No	auto_increment	Identificador del usuario
name	varchar(20)	utf8mb4_unico de_ci	No		Nombre de usuario
email	varchar(255)	utf8mb4_unico de_ci	No		Correo electrónico
email_verified_at	timestamp		Sí		Verificación de correo
password	varchar(255)	utf8mb4_unico de_ci	No		Contraseña encriptada
remember_token	varchar(100)	utf8mb4_unico de_ci	Sí		Token de sesión
created_at	timestamp		Sí		Fecha de creación
updated_at	timestamp		Sí		Fecha de actualización

Fuente: Base de Datos del Sistema de Registro y Control de Tesis.

Tabla 9. Diccionario de Datos de la Entidad permissions

Nombre de la Tabla		permissions			
Descripción		Almacena cada uno de los permisos del sistema			
Nombre	Tipo	Cotejamiento	Nulo	Extra	Descripción
id	bigint(20)		No	auto_increment	Identificador del permiso
name	varchar(255)	utf8mb4_unico de_ci	No		Nombre del permiso
guard_name	varchar(255)	utf8mb4_unico de_ci	No		Guard asociado (web)
created_at	timestamp		Sí		Fecha de creación
updated_at	timestamp		Sí		Fecha de actualización

Fuente: Base de Datos del Sistema de Registro y Control de Tesis.

Tabla 10. Diccionario de Datos de la Entidad roles

Nombre de la Tabla		roles			
Descripción		Registra los roles existentes en el sistema			
Nombre	Tipo	Cotejamiento	Nulo	Extra	Descripción
id	bigint(20)		No	auto_increment	Identificador del rol
name	varchar(255)	utf8mb4_unico de_ci	No		Nombre del rol
guard_name	varchar(255)	utf8mb4_unico de_ci	No		Guard asociado
created_at	timestamp		Sí		Fecha de creación
updated_at	timestamp		Sí		Fecha de actualización

Fuente: Base de Datos del Sistema de Registro y Control de Tesis.

Tabla 11. Diccionario de Datos de la Entidad role_has_permissions

Nombre de la Tabla		role_has_permissions			
Descripción		Tabla pivote entre roles y permisos			
Nombre	Tipo	Cotejamiento	Nulo	Extra	Descripción
permission_id	bigint(20)		No		Permiso asignado al rol
role_id	bigint(20)		No		Rol que recibe el permiso

Fuente: Base de Datos del Sistema de Registro y Control de Tesis.

Tabla 12. Diccionario de Datos de la Entidad model_has_roles

Nombre de la Tabla		model_has_roles			
Descripción		Establece la relación entre los modelos (usuarios) y roles			
Nombre	Tipo	Cotejamiento	Nulo	Extra	Descripción
role_id	bigint(20)		No		Rol asignado al modelo
model_type	varchar(255)	utf8mb4_unico de_ci	No		Tipo de modelo (ej. usuario)
model_id	bigint(20)		No		ID del modelo (usuario) asociado

Fuente: Base de Datos del Sistema de Registro y Control de Tesis.

Tabla 13. Diccionario de Datos de la Entidad preguntas_user

Nombre de la Tabla		preguntas_user			
Descripción		Almacena preguntas y respuestas de seguridad de los usuarios			
Nombre	Tipo	Cotejamiento	Nulo	Extra	Descripción
id	bigint(20)		No	auto_increment	Identificador de registro
user_id	bigint(20)		No		Relación con el usuario
pregunta_uno	varchar(255)	utf8mb4_unico de_ci	No		Primera pregunta de seguridad
respuesta_uno	varchar(255)	utf8mb4_unico de_ci	No		Respuesta de la primera pregunta
pregunta_dos	varchar(255)	utf8mb4_unico de_ci	No		Segunda pregunta de seguridad
respuesta_dos	varchar(255)	utf8mb4_unico de_ci	No		Respuesta de la segunda pregunta
created_at	timestamp		Sí		Fecha de creación
updated_at	timestamp		Sí		Fecha de actualización

Fuente: Base de Datos del Sistema de Registro y Control de Tesis.

Tabla 14. Diccionario de Datos de la Entidad model_has_permissions

Nombre de la Tabla		model_has_permissions			
Descripción		Establece la relación entre modelos (usuarios) y permisos específicos			
Nombre	Tipo	Cotejamiento	Nulo	Extra	Descripción
permission_id	bigint(20)		No		ID del permiso asignado
model_type	varchar(255)	utf8mb4_unicode_ci	No		Tipo de modelo (ej. usuario)
model_id	bigint(20)		No		ID del modelo (usuario) asociado

Fuente: Base de Datos del Sistema de Registro y Control de Tesis.

Tabla 15. Diccionario de Datos de la Entidad sessions

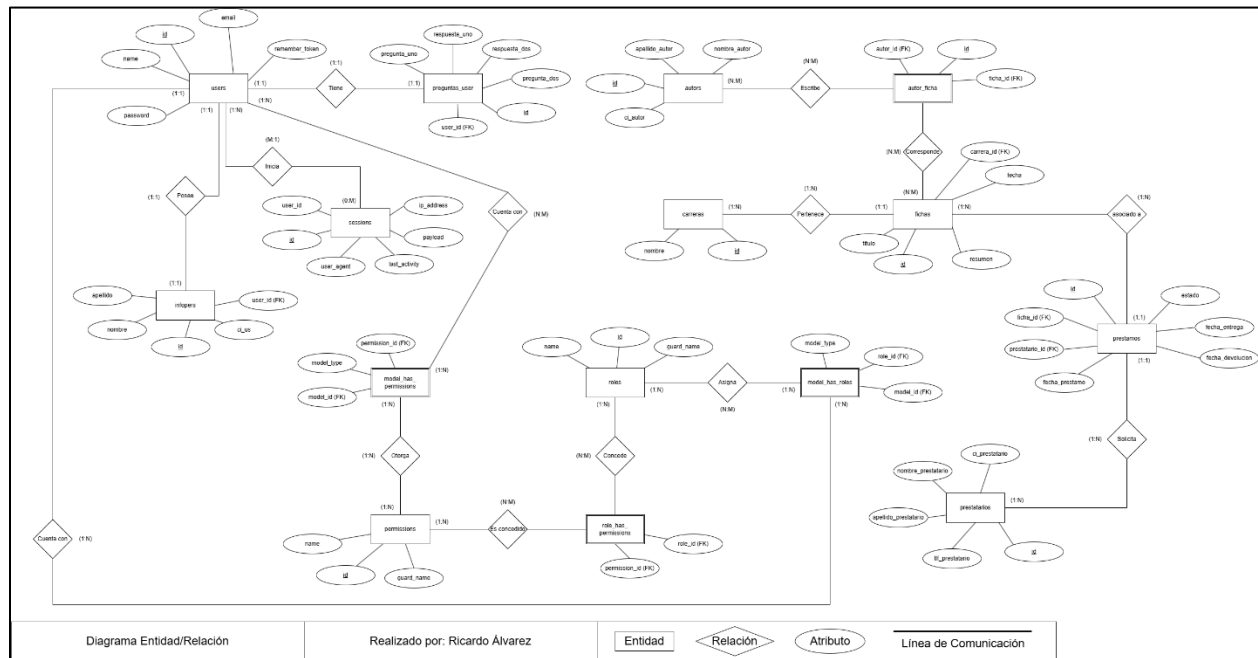
Nombre de la Tabla		sessions			
Descripción		Guarda las sesiones activas de los usuarios			
Nombre	Tipo	Cotejamiento	Nulo	Extra	Descripción
id	varchar(255)	utf8mb4_unicode_ci	No		Identificador único de sesión
user_id	bigint(20)		Sí		ID del usuario (si este aplica)
ip_addresses	varchar(45)	utf8mb4_unicode_ci	Sí		Dirección IP de la sesión
user_agent	text	utf8mb4_unicode_ci	Sí		Navegador utilizado
payload	longtext	utf8mb4_unicode_ci	No		Información serializada de la sesión
last_activity	int(11)		No		Timestamp de la última actividad

Fuente: Base de Datos del Sistema de Registro y Control de Tesis.

Diagrama Entidad — Relación

Un diagrama entidad-relación es una herramienta con la cual representamos visualmente las entidades y las relaciones entre estas y sus atributos, y el cómo se organizan en un sistema en conjunto. Aquí se muestra el modelo E/R del Sistema de Registro y control de Tesis (*Figura 1*).

Figura 1. Diagrama Entidad—Relación

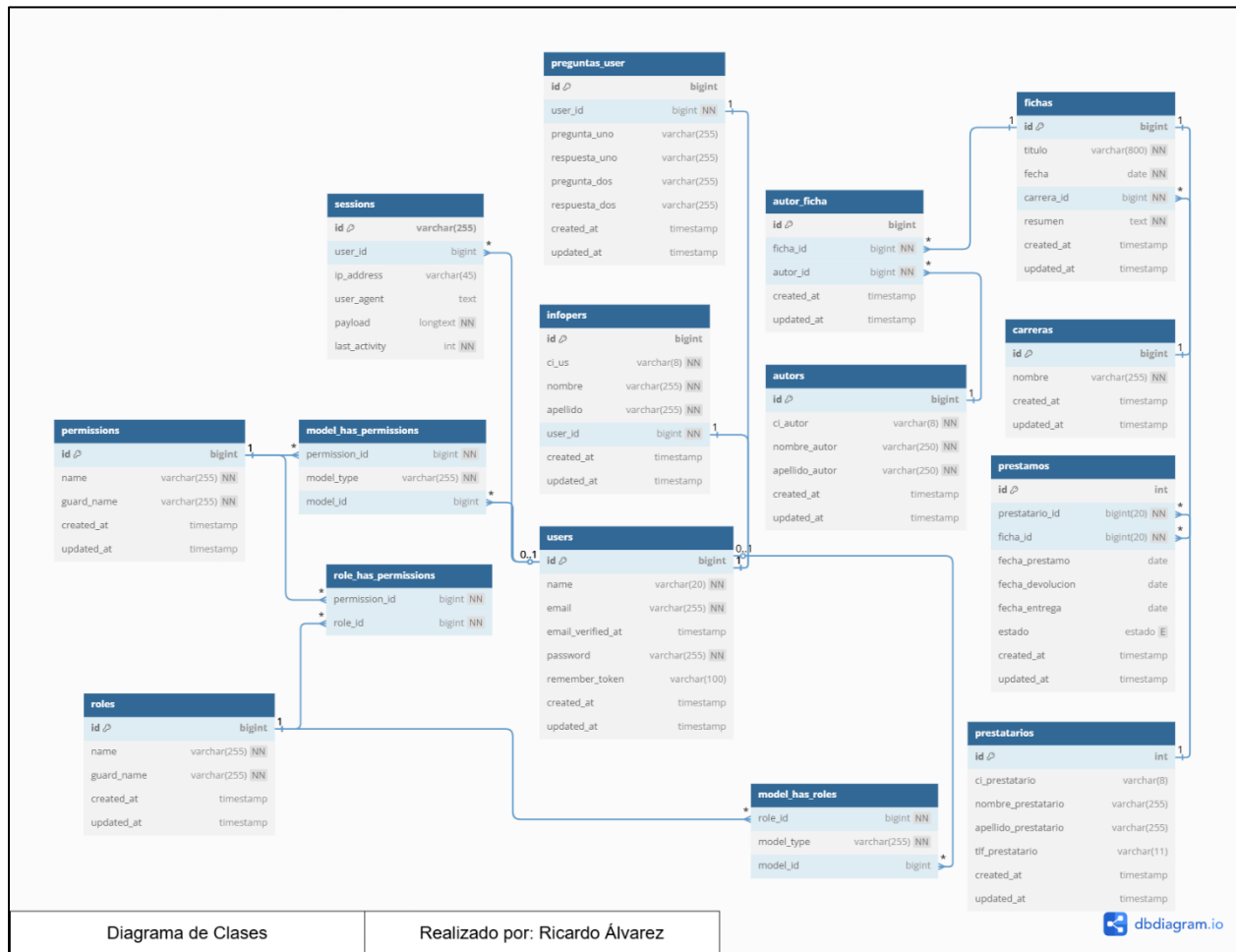


Fuente: Álvarez R. (2025)

Diagrama de Clases

El diagrama de clases es una representación visual en estático de la estructura del sistema, haciendo énfasis en sus atributos y relaciones entre ellos.

Figura 2. Diagrama de Clases



Fuente: Álvarez R. (2025)

REQUERIMIENTOS DE SOFTWARE Y HARDWARE

Requerimientos de Software

Se dejan en claro los requisitos mínimos que debe cumplir el software del equipo en el que será instalado el sistema para que este pueda realizar cada una de sus funciones de manera efectiva. Estos requerimientos los encontramos en la siguiente tabla (*Tabla 16*).

Tabla 16. Requerimientos de Software

Software	Características Mínimas
Sistema Operativo	Windows 10
Servidor Local	Apache
Sistema Gestor de Base de Datos	MySQL
Navegador	Chrome, Firefox, Opera, Microsoft Edge

Fuente: Álvarez R. (2025)

Requerimientos de Hardware

Se detallan los requerimientos mínimos con los que debe contar el hardware para que se haga uso del sistema sin complicaciones en su funcionalidad, características que cumple el equipo en el cual se encuentra instalado el sistema en el departamento. Estos requerimientos los encontramos en la siguiente tabla (*Tabla 17*).

Tabla 17. Requerimientos de Hardware

Dispositivo	Características Mínimas
Computador	Con sus componentes y periféricos.
Procesador	Intel i3 de 2 GHz o superior
Disco duro	10 GB de Espacio Libre
RAM	4 GB
Resolución de pantalla	1024 x768

Fuente: Álvarez R. (2025)

RUTA DE UBICACIÓN

Al tratarse de un sistema manejado de forma local, es preciso acceder al mismo a través de la URL del servidor local, esta viene siendo: <http://localhost/laravel/biblio/public>.

CONFIGURACIÓN

Cuando se realiza la instalación del sistema en un equipo nuevo, una vez ya instalado e iniciado el servidor local, es necesario dirigirse a la ruta <http://localhost/phpmyadmin/> y desde allí crear una nueva base de datos con el nombre “biblioteca_udo” y cotejamiento “utf8mb4_unicode_ci”, para luego proceder a importar el archivo de base de datos, generalmente en formato *.sql*. Luego de esto se procede a copiar la carpeta del Sistema a la carpeta “htdocs” del servidor local.

RUTINAS CLAVE

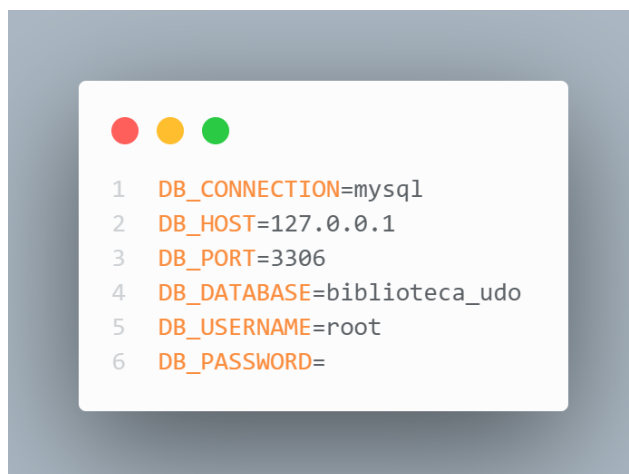
Dentro del Sistema de Registro y Control de Tesis hay ciertas rutinas que cuentan con relevancia por sobre otras para la gestión de la información que se encuentra almacenada dentro del mismo, estas son 3:

- Conexión a la Base de Datos.
- Validación de Acceso al Sistema.
- Manejo de Rutas.

Conexión a la Base de Datos

Para establecer la conexión con la base de datos almacenada en el servidor local y poder hacer uso del sistema, este almacena en el archivo “.env” los siguientes parámetros.

Figura 3. Parámetros de Conexión a la Base de Datos en “.env”



Fuente: Álvarez R. (2025)

Validación de Acceso al Sistema

Las tareas de autenticación de datos ingresados al realizar un intento de inicio de sesión para así dar acceso al interior del sistema se encuentran expresadas en el código presente en la siguiente figura.

Figura 4. Extracto del controlador "LoginController.php"



```

1  <?php
2
3  namespace App\Http\Controllers\Auth;
4
5  use App\Http\Controllers\Controller;
6
7  use Illuminate\Foundation\Auth\AuthenticatesUsers;
8
9  class LoginController extends Controller
10 {
11
12     use AuthenticatesUsers;
13
14     protected function authenticated($request, $user)
15     {
16         if ($user->hasRole('MASTER')) {
17             return redirect('/home');
18         }
19         return redirect()->route('admin.perfil', $user->id);
20     }
21
22     public function __construct()
23     {
24         $this->middleware('guest')->except('logout');
25         $this->middleware('auth')->only('logout');
26     }
27 }
28

```

Fuente: Álvarez R. (2025)

Manejo de Rutas

A partir del archivo “web.php” se manejan las rutas de acceso a todas las secciones del sistema.

Figura 5. Rutas del Sistema (1/2)



```
1 <?php
2
3 use Illuminate\Support\Facades\Auth;
4 use Illuminate\Support\Facades\Route;
5
6 Route::get('/', function () {
7     $user = Auth::user();
8     if ($user && $user->can('Ver Estadísticas del Sistema')) {
9         return redirect()->route('home');
10    } elseif ($user) {
11        return redirect()->route('admin.perfil');
12    }
13    return redirect('/login');
14 })->middleware('auth');
15
16 Route::get('/home', function () {
17     return view('home');
18 })->middleware(['auth', 'verified', 'can:Ver Estadísticas del Sistema'])->name('home');
19
20 //rutas de recuperación de contraseña
21 Route::get('/recover', [App\Http\Controllers\UsuarioController::class, 'recover'])->name('password.recuperar');
22 Route::post('/admin/usuarios/recover', [App\Http\Controllers\UsuarioController::class, 'recoverPost'])->name('password.recuperar.post');
23 Route::post('/admin/usuarios/verificarPreguntas', [App\Http\Controllers\UsuarioController::class, 'verificarPreguntas'])->name('admin.usuarios.verificarPreguntas');
24 Route::post('/admin/usuarios/resetPassword', [App\Http\Controllers\UsuarioController::class, 'resetPassword'])->name('admin.usuarios.resetPassword');
```

Fuente: Álvarez R. (2025)

Figura 6. Rutas del Sistema (2/2)

```
1 Route::middleware(['auth'])->group(function () {
2
3     // rutas usuario
4     Route::get('/admin/usuarios', [App\Http\Controllers\UsuarioController::class, 'index'])->middleware('can:Ver Lista de Usuarios')->name('admin.usuarios.index');
5     Route::get('/admin/usuarios/register', [App\Http\Controllers\UsuarioController::class, 'create'])->middleware('can:Registrar Usuario')->name('admin.usuarios.register');
6     Route::post('/admin/usuarios/register', [App\Http\Controllers\UsuarioController::class, 'store'])->middleware('can:Registrar Usuario')->name('admin.usuarios.store');
7     Route::get('/admin/usuarios/export', [App\Http\Controllers\UsuarioController::class, 'exportarUsuarios'])->middleware('can:Exportar Reporte de Usuarios')->name('usuarios.exportar');
8     Route::get('/admin/usuarios/exportPdf', [App\Http\Controllers\UsuarioController::class, 'exportPdf'])->middleware('can:Exportar Reporte de Usuarios')->name('usuarios.exportar.pdf');
9     Route::get('/admin/usuarios/{id}', [App\Http\Controllers\UsuarioController::class, 'show'])->middleware('can:Ver Información de Usuario')->name('admin.usuarios.show');
10    Route::get('/admin/usuarios/{id}/edit', [App\Http\Controllers\UsuarioController::class, 'edit'])->middleware('can:Editar Usuario')->name('admin.usuarios.edit');
11    Route::put('/admin/usuarios/{id}', [App\Http\Controllers\UsuarioController::class, 'update'])->middleware('can:Editar Usuario')->name('admin.usuarios.update');
12    Route::delete('/admin/usuarios/{id}', [App\Http\Controllers\UsuarioController::class, 'destroy'])->middleware('can:Eliminar Usuario')->name('admin.usuarios.destroy');
13    Route::get('/admin/perfil', [App\Http\Controllers\UsuarioController::class, 'perfil'])->middleware('can:Ver Perfil de Usuario')->name('admin.perfil');
14
15
16    // rutas autor
17    Route::get('/admin/autores', [App\Http\Controllers\AutorController::class, 'index'])->middleware('can:Ver Lista de Autores')->name('admin.autores.index');
18    Route::get('/admin/autores/register', [App\Http\Controllers\AutorController::class, 'create'])->middleware('can:Registrar Autor')->name('admin.autores.register');
19    Route::post('/admin/autores/register', [App\Http\Controllers\AutorController::class, 'store'])->middleware('can:Registrar Autor')->name('admin.autores.store');
20    Route::get('/admin/autores/export', [App\Http\Controllers\AutorController::class, 'exportarAutores'])->middleware('can:Exportar Reporte de Autores')->name('autores.exportar');
21    Route::get('/admin/autores/exportPdf', [App\Http\Controllers\AutorController::class, 'exportPdf'])->middleware('can:Exportar Reporte de Autores')->name('autores.exportar.pdf');
22    Route::get('/admin/autores/{id}/edit', [App\Http\Controllers\AutorController::class, 'edit'])->middleware('can:Editar Autor')->name('admin.autores.edit');
23    Route::put('/admin/autores/{id}', [App\Http\Controllers\AutorController::class, 'update'])->middleware('can:Editar Autor')->name('admin.autores.update');
24    Route::delete('/admin/autores/{id}', [App\Http\Controllers\AutorController::class, 'destroy'])->middleware('can:Eliminar Autor')->name('admin.autores.destroy');
25
26    //rutas Carreras
27    Route::get('/admin/carreras', [App\Http\Controllers\CarreraController::class, 'index'])->middleware('can:Ver Lista de Carreras')->name('admin.carreras.index');
28    Route::get('/admin/carreras/register', [App\Http\Controllers\CarreraController::class, 'create'])->middleware('can:Registrar Carrera')->name('admin.carreras.register');
29    Route::post('/admin/carreras/register', [App\Http\Controllers\CarreraController::class, 'store'])->middleware('can:Registrar Carrera')->name('admin.carreras.store');
30    Route::get('/admin/carreras/export', [App\Http\Controllers\CarreraController::class, 'exportarCarreras'])->middleware('can:Exportar Reporte de Carreras')->name('carreras.exportar');
31    Route::get('/admin/carreras/exportPdf', [App\Http\Controllers\CarreraController::class, 'exportPdf'])->middleware('can:Exportar Reporte de Carreras')->name('carreras.exportar.pdf');
32    Route::get('/admin/carreras/{id}/edit', [App\Http\Controllers\CarreraController::class, 'edit'])->middleware('can:Editar Carrera')->name('admin.carreras.edit');
33    Route::put('/admin/carreras/{id}', [App\Http\Controllers\CarreraController::class, 'update'])->middleware('can:Editar Carrera')->name('admin.carreras.update');
34    Route::delete('/admin/carreras/{id}', [App\Http\Controllers\CarreraController::class, 'destroy'])->middleware('can:Eliminar Carrera')->name('admin.carreras.destroy');
35
36    //rutas fichas
37    Route::get('/admin/fichas', [App\Http\Controllers\FichaController::class, 'index'])->middleware('can:Ver Lista de Fichas')->name('admin.fichas.index');
38    Route::get('/admin/fichas/register', [App\Http\Controllers\FichaController::class, 'create'])->middleware('can:Registrar Ficha')->name('admin.fichas.register');
39    Route::post('/admin/fichas/register', [App\Http\Controllers\FichaController::class, 'store'])->middleware('can:Registrar Ficha')->name('admin.fichas.store');
40    Route::get('/admin/fichas/export', [App\Http\Controllers\FichaController::class, 'exportarFichas'])->middleware('can:Exportar Reporte de Fichas')->name('fichas.exportar');
41    Route::get('/admin/fichas/exportPdf', [App\Http\Controllers\FichaController::class, 'exportPdf'])->middleware('can:Exportar Reporte de Fichas')->name('fichas.exportar.pdf');
42    Route::get('/admin/fichas/{id}', [App\Http\Controllers\FichaController::class, 'show'])->middleware('can:Ver Información de Ficha')->name('admin.fichas.show');
43    Route::get('/admin/fichas/pdf/{id}', [App\Http\Controllers\FichaController::class, 'pdf'])->middleware('can:Exportar Reporte de Ficha Unica')->name('admin.fichas.pdf');
44    Route::get('/admin/fichas/{id}/edit', [App\Http\Controllers\FichaController::class, 'edit'])->middleware('can:Editar Ficha')->name('admin.fichas.edit');
45    Route::get('/admin/fichas/{cedula}/get', [App\Http\Controllers\FichaController::class, 'buscarAutor'])->middleware('can:Registrar Ficha')->name('admin.fichas.buscarAutor');
46    Route::get('/admin/fichas/{cedula}/fichasBuscar', [App\Http\Controllers\FichaController::class, 'fichasBuscar'])->middleware('can:Registrar Ficha')->name('fichas.buscar');
47    Route::put('/admin/fichas/{id}', [App\Http\Controllers\FichaController::class, 'update'])->middleware('can:Editar Ficha')->name('admin.fichas.update');
48    Route::delete('/admin/fichas/{id}', [App\Http\Controllers\FichaController::class, 'destroy'])->middleware('can:Eliminar Ficha')->name('admin.fichas.destroy');
49    Route::delete('/admin/quitar_autor/{id}', [App\Http\Controllers\FichaController::class, 'quitar'])->middleware('can:Editar Ficha')->name('fichas.quitar');
50
51    //rutas prestamos
52    Route::get('/admin/prestamos', [App\Http\Controllers\PrestamoController::class, 'index'])->middleware('can:Ver Lista de Préstamos')->name('admin.prestamos.index');
53    Route::get('/admin/prestamos/register', [App\Http\Controllers\PrestamoController::class, 'create'])->middleware('can:Registrar Préstamo')->name('admin.prestamos.register');
54    Route::post('/admin/prestamos/register', [App\Http\Controllers\PrestamoController::class, 'store'])->middleware('can:Registrar Préstamo')->name('admin.prestamos.store');
55    Route::get('/admin/prestamos/export', [App\Http\Controllers\PrestamoController::class, 'exportarPrestamos'])->middleware('can:Exportar Reporte de Préstamos')->name('prestamos.exportar');
56    Route::get('/admin/prestamos/exportPdf', [App\Http\Controllers\PrestamoController::class, 'exportPdf'])->middleware('can:Exportar Reporte de Préstamos')->name('prestamos.exportar.pdf');
57    Route::get('/admin/prestamos/pdf/{id}', [App\Http\Controllers\PrestamoController::class, 'pdf'])->middleware('can:Exportar Reporte de Préstamos')->name('admin.prestamos.pdf');
58    Route::get('/admin/prestamos/{id}', [App\Http\Controllers\PrestamoController::class, 'show'])->middleware('can:Ver Información de Préstamo')->name('admin.prestamos.show');
59    Route::put('/admin/prestamos/{id}/devolver', [App\Http\Controllers\PrestamoController::class, 'devolver'])->middleware('can:Marcar Préstamo como Devuelto')->name('admin.prestamos.devolver');
60    Route::put('/admin/prestamos/{id}/edit', [App\Http\Controllers\PrestamoController::class, 'edit'])->middleware('can:Editar Préstamo')->name('admin.prestamos.edit');
61    Route::put('/admin/prestamos/{id}', [App\Http\Controllers\PrestamoController::class, 'update'])->middleware('can:Editar Préstamo')->name('admin.prestamos.update');
62    Route::delete('/admin/prestamos/{id}', [App\Http\Controllers\PrestamoController::class, 'destroy'])->middleware('can:Eliminar Préstamo')->name('admin.prestamos.destroy');
63
64    //rutas de prestatario
65    Route::post('/admin/prestatarios/register', [App\Http\Controllers\VprestatarioController::class, 'store'])->middleware('can:Registrar Préstamo')->name('prestatarios.store');
66    Route::get('/admin/prestatarios/buscar', [App\Http\Controllers\VprestatarioController::class, 'buscarPorCedula'])->middleware('can:Registrar Préstamo')->name('prestatarios.buscar');
67    Route::put('/prestatarios/{id}', [App\Http\Controllers\VprestatarioController::class, 'update'])->middleware('can:Editar Préstamo')->name('prestatarios.update');
68    Route::delete('/admin/prestatarios/{id}', [App\Http\Controllers\VprestatarioController::class, 'destroy'])->middleware('can:Eliminar Préstamo')->name('prestatarios.destroy');
69
70    //rutas roles
71    Route::get('/admin/roles', [App\Http\Controllers\RoleController::class, 'index'])->middleware('can:Ver Lista de Roles')->name('admin.roles.index');
72    Route::get('/admin/roles/register', [App\Http\Controllers\RoleController::class, 'create'])->middleware('can:Registrar Rol')->name('admin.roles.register');
73    Route::post('/admin/roles/register', [App\Http\Controllers\RoleController::class, 'store'])->middleware('can:Registrar Rol')->name('admin.roles.store');
74    Route::get('/admin/roles/export', [App\Http\Controllers\RoleController::class, 'exportarRoles'])->middleware('can:Exportar Reporte de Roles')->name('roles.exportar');
75    Route::get('/admin/roles/exportPdf', [App\Http\Controllers\RoleController::class, 'exportPdf'])->middleware('can:Exportar Reporte de Roles')->name('roles.exportar.pdf');
76    Route::get('/admin/roles/{id}/edit', [App\Http\Controllers\RoleController::class, 'edit'])->middleware('can:Editar Rol')->name('admin.roles.edit');
77    Route::put('/admin/roles/{id}', [App\Http\Controllers\RoleController::class, 'update'])->middleware('can:Editar Rol')->name('admin.roles.update');
78    Route::put('/admin/roles/{id}/asignar', [App\Http\Controllers\RoleController::class, 'asignar'])->middleware('can:Editar Rol')->name('admin.roles.asignar');
79    Route::put('/admin/roles/asignar/{id}', [App\Http\Controllers\RoleController::class, 'otorgar'])->middleware('can:Editar Rol')->name('admin.roles.otorgar');
80    Route::delete('/admin/roles/{id}', [App\Http\Controllers\RoleController::class, 'destroy'])->middleware('can:Eliminar Rol')->name('admin.roles.destroy');
81
82    //rutas de respaldo
83    Route::get('/admin/backup', [App\Http\Controllers\BackupController::class, 'index'])->middleware('can:Ver Lista de Respaldos')->name('admin.backup.index');
84    Route::get('/admin/backup/create', [App\Http\Controllers\BackupController::class, 'create'])->middleware('can:Crear Respaldo')->name('admin.backup.create');
85    Route::post('/admin/backup/restore', [App\Http\Controllers\BackupController::class, 'restore'])->middleware('can:Restaurar Respaldo')->name('admin.backup.restore');
86    Route::post('/admin/backup/upload', [App\Http\Controllers\BackupController::class, 'uploadRestore'])->middleware('can:Restaurar Respaldo desde Dispositivo')->name('admin.backup.upload');
87    Route::get('/admin/backup/descargar/{nombreFile}', [App\Http\Controllers\BackupController::class, 'descargar'])->middleware('can:Descargar Respaldo')->name('admin.backup.descargar');
88    Route::delete('/admin/backup/eliminar/{nombreFile}', [App\Http\Controllers\BackupController::class, 'eliminar'])->middleware('can:Eliminar Respaldo')->name('admin.backup.eliminar');
89
90    });
91
92    require __DIR__ . '/auth.php';
93
94    Auth::routes();
95
96    Route::get('/home', [App\Http\Controllers\HomeController::class, 'index'])->middleware(['auth', 'can:Ver Estadísticas del Sistema'])->name('home');
```

Fuente: Álvarez R. (2025)

LISTA DE ERRORES

- **Error de Conexión con la Base de Datos:** es posible encontrarse con este error si al intentar acceder al sistema instalado no se han puesto en marcha los servicios de MySQL del servidor local o bien si la base de datos no existe dentro del mismo.
- **404 | Página No Encontrada:** puede encajar en la categoría de errores de ruta, ya que este es específico al intentar acceder a una dirección que no se encuentra definida dentro de las rutas del sistema en el archivo “web.php”, por tanto, al no existir, el sistema no la reconoce y arroja el error.
- **403 | Esta Acción No Está Autorizada:** otro de los posibles errores de ruta es el de acción no autorizada, este se presenta por lo general cuando un usuario autenticado trata de acceder a una ruta dentro del sistema a la cual no le corresponde según sus permisos definidos, el sistema bloquea el intento de acceso y arroja el error.