

Documentación

Completa del Proyecto

AppConsultaHogar

Índice

1. [Introducción](#)
2. [Estructura del Proyecto](#)
3. [Rutas](#)
4. [Controladores](#)
5. [Modelos](#)
6. [Vistas](#)
7. [Componentes](#)
8. [Flujo de Datos](#)
9. [Validaciones](#)
10. [Base de Datos](#)
11. [Funcionalidades Principales](#)

Introducción

AppConsultaHogar es una aplicación web desarrollada en Laravel que permite consultar información detallada sobre hogares, integrantes y logros asociados a cada hogar. La aplicación está diseñada para facilitar el seguimiento y gestión de logros para cada integrante de un hogar registrado en el sistema.

La aplicación permite buscar hogares por folio o documento, visualizar detalles del hogar, información de sus integrantes y los logros asociados a cada integrante. Además, permite generar reportes en PDF con la información del hogar.

Estructura del Proyecto

El proyecto sigue la estructura estándar de Laravel:

```
appConsultaHogar/  
├─ app/  
│   ├─ Http/  
│   │   └─ Controllers/  
│   │       ├─ ConsultaHogarController.php  
│   │       ├─ SearchController.php  
│   │       └─ SearchMefController.php  
│   └─ Models/  
│       ├─ ConsultaHogar.php  
│       ├─ Integrante.php  
│       ├─ Search.php  
│       └─ SearchMef.php  
├─ config/  
├─ database/  
├─ public/  
├─ resources/  
│   └─ views/  
│       └─ components/  
│           ├─ detalle-logro.blade.php  
│           ├─ hogar-details.blade.php  
│           ├─ integrantes-list.blade.php  
│           ├─ logros-integrante.blade.php  
│           ├─ logros-integrante-modal.blade.php  
│           ├─ logros-list.blade.php  
│           └─ tabla-resultados.blade.php  
│       └─ consultaHogar.blade.php  
│       └─ search.blade.php  
│       └─ searchMef.blade.php  
├─ routes/  
│   └─ web.php  
└─ storage/
```

Rutas

Las rutas de la aplicación están definidas en el archivo `routes/web.php` y se organizan de la siguiente manera:

Rutas Principales

```
// Ruta principal redirige a la búsqueda
Route::get('/', function () {
    return redirect()->route('search');
})->name('home');

// Rutas de búsqueda general
Route::get('/search', [SearchController::class, 'index'])
Route::post('/search/process', [SearchController::class,

// Rutas de búsqueda MEF
Route::get('/searchMef', [SearchMefController::class, 'i
Route::post('/searchMef/process', [SearchMefController::

// Ruta para obtener detalles de la búsqueda
Route::post('/search/details', [SearchController::class,
```

Rutas de Consulta de Hogar

```
// Rutas principales de ConsultaHogar
Route::get('/consultaHogar', [ConsultaHogarController::c
Route::post('/consultaHogar/process', [ConsultaHogarCont
Route::post('/consultaHogar/details', [ConsultaHogarCont
Route::get('/consultaHogar/download', [ConsultaHogarCont
Route::post('/consultaHogar/download-ficha', [ConsultaHo
```

Rutas de Logros

```
// Rutas para los logros
Route::post('/consulta-hogar/logros-integrante', [Consul
Route::post('/consulta-hogar/detalle-logro', [ConsultaHo
Route::get('/getDimensiones', [ConsultaHogarController::
```

Ruta de Cierre de Sesión

```
// Ruta de logout
Route::post('/logout', function () {
    Auth::logout();
    return redirect()->route('home');
})->name('logout');
```

Controladores

ConsultaHogarController

Este es el controlador principal de la aplicación y maneja todas las funcionalidades relacionadas con la consulta de hogares, integrantes y logros.

Métodos Principales:

1. **index()**: Muestra la vista principal de consulta de hogar.
2. **process(Request \$request)**: Procesa la búsqueda de hogares por folio o documento.

- Valida que se ingrese al menos un folio o documento.
 - Realiza la búsqueda en la base de datos.
 - Devuelve los resultados para mostrar en la vista.
3. **getDetails(Request \$request):** Obtiene los detalles de un hogar específico.
- Recibe el folio del hogar.
 - Consulta la información detallada del hogar.
 - Obtiene la lista de integrantes del hogar.
 - Devuelve una vista parcial con los detalles.
4. **getLogrosIntegrante(Request \$request):** Obtiene los logros asociados a un integrante.
- Recibe el ID del integrante y el folio del hogar.
 - Consulta los logros agrupados por dimensión.
 - Devuelve los datos para mostrar en la interfaz.
5. **getDetalleLogro(Request \$request):** Obtiene el detalle de un logro específico.
- Recibe el ID del logro, ID de dimensión, folio e ID del integrante.
 - Consulta el detalle del logro y la información del integrante.
 - Devuelve una vista parcial con el detalle del logro.
6. **download(Request \$request):** Genera un archivo PDF con la información del hogar.
- Recibe el folio del hogar.
 - Genera un PDF con los detalles del hogar y sus integrantes.
7. **downloadFicha(Request \$request):** Genera una ficha detallada del hogar en formato PDF.

- Recibe el folio del hogar.
 - Genera un PDF con información más detallada del hogar.
8. **getDimensiones()**: Obtiene la lista de dimensiones disponibles para los logros.

SearchController y SearchMefController

Estos controladores manejan las funcionalidades de búsqueda general y búsqueda MEF respectivamente, con métodos similares para procesar búsquedas y mostrar resultados.

Modelos

ConsultaHogar

Este modelo representa la tabla `t1_principalthogar` en la base de datos y maneja las consultas relacionadas con los hogares.

Atributos y Configuración:

```
protected $connection = 'mysql';
protected $table = 'familiar_modulo_cif.t1_principalthoga
protected $primaryKey = 'folio';
public $timestamps = false;
```

Métodos Principales:

1. **integrantes()**: Relación con el modelo Integrante.

```
public function integrantes()
{
    return $this->hasMany(Integrante::class, 'folio'
}
```

2. **buscarPorFolioODocumento(\$folio, \$documento)** Método estático para buscar hogares por folio o documento.

- Realiza consultas SQL para obtener información del hogar.
- Devuelve una colección con los resultados.

Integrante

Este modelo representa la tabla `t1_principalintegrantes` en la base de datos y maneja las consultas relacionadas con los integrantes de un hogar.

Atributos y Configuración:

```
protected $connection = 'mysql';
protected $table = 'familiar_modulo_cif.t1_principalinte
protected $primaryKey = 'idintegrante';
public $timestamps = false;
```

Métodos Principales:

1. **hogar()**: Relación con el modelo ConsultaHogar.

```
public function hogar()
{
    return $this->belongsTo(ConsultaHogar::class, 'f
}
```


2. **getIntegrantesByFolio(\$folio)**: Método estático para obtener todos los integrantes de un hogar.
 - Recibe el folio del hogar.
 - Devuelve una colección con los integrantes.

Vistas

consultaHogar.blade.php

Esta es la vista principal de la aplicación y contiene:

1. **Formulario de Búsqueda**: Permite buscar hogares por folio o documento.
2. **Tabla de Resultados**: Muestra los resultados de la búsqueda.
3. **Modal de Detalles**: Muestra los detalles del hogar seleccionado.
4. **Modal de Logros**: Muestra los logros de un integrante específico.
5. **Modal de Detalle de Logro**: Muestra el detalle de un logro específico.

La vista incluye scripts JavaScript para manejar las interacciones del usuario, como:

- Envío de formularios mediante AJAX.
- Carga dinámica de detalles del hogar.
- Carga dinámica de logros de integrantes.
- Visualización de detalles de logros.

search.blade.php y searchMef.blade.php

Estas vistas manejan las funcionalidades de búsqueda general y búsqueda MEF respectivamente, con estructuras similares a la vista principal.

Componentes

`detalle-logro.blade.php`

Este componente muestra el detalle de un logro específico, incluyendo:

- Lista de integrantes a los que aplica el logro.
- Estado inicial y actual del logro para cada integrante.

```

<div class="modal-dialog modal-lg">
  <div class="modal-content">
    <div class="modal-header bg-primary text-white">
      <h5 class="modal-title">Detalle del Logro</h5>
      <button type="button" class="btn-close" data-
    </div>
    <div class="modal-body">
      <!-- Tabla de integrantes a los que aplica e
      <div class="card">
        <div class="card-header bg-info text-whi
          <h6 class="mb-0">Integrantes a los q
        </div>
        <div class="card-body">
          <div class="table-responsive">
            <table class="table table-stripe
              <!-- Contenido de la tabla -
            </table>
          </div>
        </div>
      </div>
    </div>
    <div class="modal-footer">
      <button type="button" class="btn btn-seconda
    </div>
  </div>
</div>

```

hogar-details.blade.php

Este componente muestra los detalles de un hogar, incluyendo:

- Información general del hogar (dirección, comuna, barrio, etc.).
- Lista de integrantes del hogar.
- Opciones para ver los logros de cada integrante.

integrantes-list.blade.php

Este componente muestra la lista de integrantes de un hogar en formato de tabla, con opciones para ver los logros de cada integrante.

logros-integrante.blade.php

Este componente muestra los logros de un integrante específico, agrupados por dimensión, y permite:

- Seleccionar un integrante del hogar.
- Ver los logros organizados por dimensiones en un acordeón.
- Ver el estado inicial y actual de cada logro.

```
<div class="mb-3">
    <select class="form-select" id="selectIntegrante">
        <option value="">Seleccione un integrante...</option>
        @foreach($integrantes as $integrante)
            <option value="{{ $integrante->idintegrante }} - {{ $integrante->nombrecompleto }}">
                {{ $integrante->nombrecompleto }}
            </option>
        @endforeach
    </select>
</div>

<div id="logrosContainer" style="display: none;">
    <div class="accordion" id="dimensionesAccordion">
        <!-- Las dimensiones y logros se cargarán dinámicamente -->
    </div>
</div>
```

logros-integrante-modal.blade.php

Este componente muestra un modal con los logros de un integrante, similar a `logros-integrante.blade.php` pero adaptado para mostrarse en un modal.

tabla-resultados.blade.php

Este componente muestra los resultados de la búsqueda de hogares en formato de tabla, con opciones para ver los detalles de cada hogar.

Flujo de Datos

El flujo de datos en la aplicación sigue el siguiente patrón:

1. Búsqueda de Hogar:

- El usuario ingresa un folio o documento en el formulario de búsqueda.
- La solicitud se envía al método `process` del controlador `ConsultaHogarController`.
- El controlador valida los datos y realiza la búsqueda en la base de datos.
- Los resultados se devuelven y se muestran en la tabla de resultados.

2. Visualización de Detalles del Hogar:

- El usuario hace clic en "Ver Detalles" para un hogar específico.
- La solicitud se envía al método `getDetails` del controlador.
- El controlador obtiene los detalles del hogar y sus

integrantes.

- Los detalles se muestran en un modal.

3. Visualización de Logros de Integrante:

- El usuario hace clic en "Ver Logros" para un integrante específico.
- La solicitud se envía al método `getLogrosIntegrante` del controlador.
- El controlador obtiene los logros del integrante agrupados por dimensión.
- Los logros se muestran en un modal con un acordeón de dimensiones.

4. Visualización de Detalle de Logro:

- El usuario hace clic en un logro específico.
- La solicitud se envía al método `getDetalleLogro` del controlador.
- El controlador obtiene el detalle del logro y la información del integrante.
- El detalle se muestra en un modal.

5. Generación de PDF:

- El usuario hace clic en "Descargar PDF" para un hogar específico.
- La solicitud se envía al método `download` o `downloadFicha` del controlador.
- El controlador genera un PDF con la información del hogar.
- El PDF se descarga en el navegador del usuario.

Validaciones

Las validaciones en la aplicación se realizan principalmente en el controlador `ConsultaHogarController`:

1. Validación de Búsqueda:

```
if (empty($folio) && empty($documento)) {  
    if ($request->ajax()) {  
        return response()->json([  
            'error' => 'Por favor ingrese un folio o documento',  
        ], 400);  
    }  
    return back()->with('error', 'Por favor ingrese un folio o documento');  
}
```

2. Validación de Parámetros para Detalles de Logro:

```
if (empty($idlogro) || empty($folio)) {  
    return response()->json([  
        'success' => false,  
        'error' => 'ID de logro y folio son requeridos',  
    ], 400);  
}
```

3. Manejo de Errores en Consultas:

```
try {  
    // Código de consulta  
} catch (\Exception $e) {  
    \Log::error('Error en getDetalleLogro: ' . $e->getMessage());  
    \Log::error('Stack trace: ' . $e->getTraceAsString());  
    return response()->json([  
        'success' => false,  
        'error' => 'Error al obtener detalle del logro',  
    ], 500);  
}
```

Base de Datos

La aplicación utiliza múltiples tablas y procedimientos almacenados para gestionar la información:

Tablas Principales:

1. **t1_principalhogar**: Almacena la información general de los hogares.
 - Campos: folio, direccion, comuna, barrio, departamento, municipio, estado, etc.
2. **t1_principalintegrantes**: Almacena la información de los integrantes de cada hogar.
 - Campos: idintegrante, folio, documento, nombre1, nombre2, apellido1, apellido2, telefono, etc.

Procedimientos Almacenados:

1. **sp4logroresultado**: Obtiene los resultados de logros para un hogar específico.
 - Parámetros: folio, idlogro.
2. **spdatosintegrante**: Obtiene los datos de un integrante específico.
 - Parámetros: folio, idintegrante.

Funcionalidades Principales

1. Consulta de Hogares

La aplicación permite buscar hogares por folio o documento, mostrando los resultados en una tabla con opciones para ver detalles, descargar PDF y ver logros de los integrantes.

2. Visualización de Detalles del Hogar

Se pueden ver los detalles de un hogar específico, incluyendo información general y lista de integrantes.

3. Gestión de Logros

La aplicación permite ver los logros de cada integrante, agrupados por dimensión, y visualizar el detalle de cada logro, incluyendo:

- Estado inicial y actual del logro.
- Lista de integrantes a los que aplica el logro.
- Información detallada del logro.

4. Generación de Reportes

Se pueden generar reportes en formato PDF con la información del hogar y sus integrantes, incluyendo:

- Ficha básica del hogar.
- Ficha detallada con información completa.

5. Visualización de Estados de Logros

Los logros se muestran con indicadores visuales (colores) que representan su estado:

- Rojo: No cumple
- Verde: Cumple
- Gris: No aplica
- Azul: En proceso
- Café: Pendiente
- Blanco: No evaluado

Estos indicadores permiten identificar rápidamente el estado de cada logro para cada integrante.