

CODIGO JOAQUIN GARCIA HERNANDEZ

En este archivo voy a explicar el código mas relevante de la página.

Archivo de rutas.

```
Route::get('/', 'HomeController@welcome')->name('welcome');

Auth::routes();

Route::get('/activity', 'ProductController@updatedActivity');

Route::get('/logout', 'HomeController@logout')->name('logout');

Route::get('/admin', 'AdminController@index');

Route::get('/admin/productos', 'AdminController@productos')->name('admin.productos');

Route::get('/admin/users', 'AdminController@usuarios')->name('admin.usuarios');

Route::get('/admin/rooms', 'AdminController@rooms')->name('admin.rooms');

Route::get('/admin/categorias', 'AdminController@categorias')->name('admin.categorias');

Route::get('/perfil', 'HomeController@perfil')->name('perfil');

Route::get('/reservar', 'ProductController@reservar');

Route::get('/devolver/{id}', 'ProductController@devolver');


//categorias

Route::get('category/destroy/{id}', 'CategoryController@destroy');
Route::post('category/guardar/{id}', 'CategoryController@guardar');
Route::resource('/category', 'CategoryController');


//productos
Route::get('product/destroy/{id}', 'ProductController@destroy');
Route::post('product/guardar/{id}', 'ProductController@guardar');
Route::resource('/product', 'ProductController');


//rooms
Route::get('room/destroy/{id}', 'RoomController@destroy');
Route::post('room/guardar/{id}', 'RoomController@guardar');
Route::resource('/room', 'RoomController');


//User
Route::get('user/editar/{id}', 'UserController@editar');
Route::get('user/destroy/{id}', 'UserController@destroy');
Route::post('user/guardar/{id}', 'UserController@guardar');
Route::resource('/User', 'UserController');
```

En esta funcion welcome del HomeController llamamos a la vista welcome donde se muestran los productos disponibles , en esta funcion tambien realizamos la busqueda de los productos tanto por name , serial , categorias e id. Tambien gestionamos la paginación mediante ajax para que si hay una cantidad grande de productos se carguen poco a poco para no relentizar la página.

```

public function welcome(Request $request){

    $productos = Product::where('status', 'NO RESERVADO')
    ->where(function($query) use ($request){
        $query->where('name', 'LIKE', '%' . $request->dato . '%')
        ->orWhere('serial', 'LIKE', '%' . $request->dato . '%')
        ->orWhere('id', 'LIKE', '%' . $request->dato . '%');
    })->paginate(8);

    $rooms = Room::all();
    $categorias = Category::all();
    if (isset($request->category_id)){
        $productos = Product::where('status', 'NO RESERVADO')->where('category_id', $request->category_id)
        ->paginate(10);
    }
    if ($request->ajax()) {
        return response()->json(view('tablaproductos', compact('productos', 'rooms', 'categorias'))->render());
    }
    if ($request->ajax()) {
        return response()->json(view('tablaproductos', compact('productos', 'rooms', 'categorias'))->render());
    }

    return view('welcome', compact('productos', 'rooms', 'categorias'));

}

```

En la siguiente funcion llamada reservar es donde introducimos en la tabla rent los datos de la reserva , tambien realizamos el envio del mensaje al grupo de telegram.

```

public function reservar(Request $request){

    $fecha = Carbon::now()->format('d M Y');

    $new['start_date'] = $fecha;
    $new['end_date'] = Carbon::parse($request->end_date)->format('d M Y');
    $new['user_id'] = Auth::user()->id;
    $new['product_id'] = $request->idproducto;
    $new['room_id'] = $request->room_id;

    $rent = Rent::create($new);

    $producto = Product::where('id', $request->idproducto)->first();
    $producto->status = 'RESERVADO';
    $producto->save();

    $text = 'Producto ' . $producto->name . ' reservado por ' . Auth::user()->name . ' hasta ' . $request->end_date;

    Telegram::sendMessage([
        'chat_id' => '-1001353881659',
        'parse_mode' => 'HTML',
        'text' => $text
    ]);

    $productos = Product::where('status', 'NO RESERVADO')->paginate(5);
    $categorias = Category::all();

    return view('welcome', compact('productos', 'categorias'));

}

```

La funcion devolver coge el id del producto que quieres devolver, lo busca y le cambia el estado del producto a no reservado.

```
public function devolver($id){  
  
    $fecha = Carbon::now()->format('d M Y');  
  
    $reserva = Rent::find($id);  
    $reserva->real_end_date = $fecha;  
    $reserva->save();  
  
    $producto = Product::find($reserva->product_id);  
    $producto->status = 'NO RESERVADO';  
    $producto->save();  
  
    $reservados = Rent::where('user_id',Auth::user()->id)->get();  
  
    return redirect()->route('perfil');  
  
}
```

Hasta aquí las funciones mas relevantes del proyecto las demas funciones son crud.