

Sistema de Ventas laravel 8 y livewire

Sergi Castillo Tiñena

Contenido

Herramientas necesarias:.....	2
Instalación	2
Base de Datos.....	4
GitHub	6
Plantilla del Proyecto	8
Plugins y Complementos.....	12
Migraciones.....	13
Seeders.....	15
Componente Categories.....	18
Componentes:.....	21
Categorías:.....	21
Productos	23
Denominations/Monedas	25
Ventas.....	27
Usuarios:.....	27
Corte de Caja	29
Roles y Permisos.....	30
Reportes	31
Configurar Permisos	32
Middlewares.....	34

Herramientas necesarias:

Composer

Git

NodeJS

Laragon

Editor de Texto (Sublime Text/Visual Studio)

Instalación

Pasos para crear el proyecto.

```
Creating project: [ventaslite]. Please wait...

***** Database:
Created database: [ventaslite]

***** Hint: In Terminal, you can type:
-----
cd C:\laragon\www
composer create-project laravel/laravel ventaslite --prefer-dist
-----


Running.....
Creating a "laravel/laravel" project at "./ventaslite"
Info from https://repo.packagist.org: #StandWithUkraine
Installing laravel/laravel (v8.6.11)
- Downloading laravel/laravel (v8.6.11)
- Installing laravel/laravel (v8.6.11): Extracting archive
Created project in C:\laragon\www\ventaslite
> @php -r "file_exists('.env') || copy('.env.example', '.env');"
Loading composer repositories with package information
Info from https://repo.packagist.org: #StandWithUkraine
Updating dependencies
Lock file operations: 110 installs, 0 updates, 0 removals
- Locking asm89/stack-cors (v2.1.1)
- Locking brick/math (0.9.3)
- Locking dflydev/dot-access-data (v3.0.1)
- Locking doctrine/inflector (2.0.4)
- Locking doctrine/instantiator (1.4.1)
- Locking doctrine/lexer (1.2.3)
```

```
C:\laragon\www\ventaslite>composer require laravel/ui
Info from https://repo.packagist.org: #StandWithUkraine
Using version ^3.4 for laravel/ui
./composer.json has been updated
Running composer update laravel/ui
Loading composer repositories with package information
Info from https://repo.packagist.org: #StandWithUkraine
Updating dependencies
Lock file operations: 1 install, 0 updates, 0 removals
- Locking laravel/ui (v3.4.5)
Writing lock file
Installing dependencies from lock file (including require-dev)
Package operations: 1 install, 0 updates, 0 removals
- Downloading laravel/ui (v3.4.5)
- Installing laravel/ui (v3.4.5): Extracting archive
Package swiftmailer/swiftmailer is abandoned, you should avoid using it. Use symfony/mailer instead.
Generating optimized autoload files
```

```
> @php artisan vendor:publish --tag=laravel-assets --ansi --force
No publishable resources for tag [laravel-assets].
Publishing complete.

C:\laragon\www\ventaslite>php artisan ui bootstrap
Bootstrap scaffolding installed successfully.
Please run "npm install && npm run dev" to compile your fresh scaffolding.

C:\laragon\www\ventaslite>php artisan ui bootstrap --auth
Bootstrap scaffolding installed successfully.
Please run "npm install && npm run dev" to compile your fresh scaffolding.
Authentication scaffolding generated successfully.

C:\laragon\www\ventaslite>npm install
npm [WARN] deprecated querystring@0.2.0: The querystring API is considered Legacy. new code should
API instead.

added 769 packages, and audited 770 packages in 38s
```

```
C:\laragon\www\ventaslite>npm run dev

> dev
> npm run development

> development
> mix

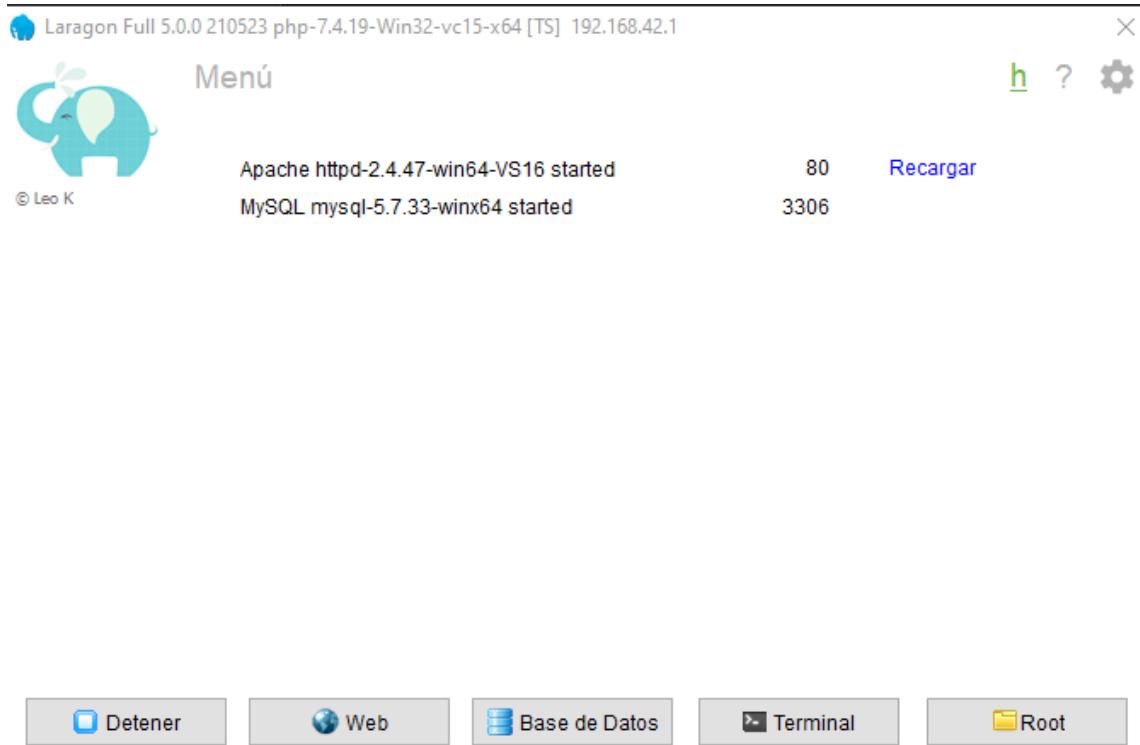
• Mix [██████████] emitting (98%)
  after emit

✓ Mix
  Compiled successfully in 4.84s
```

```

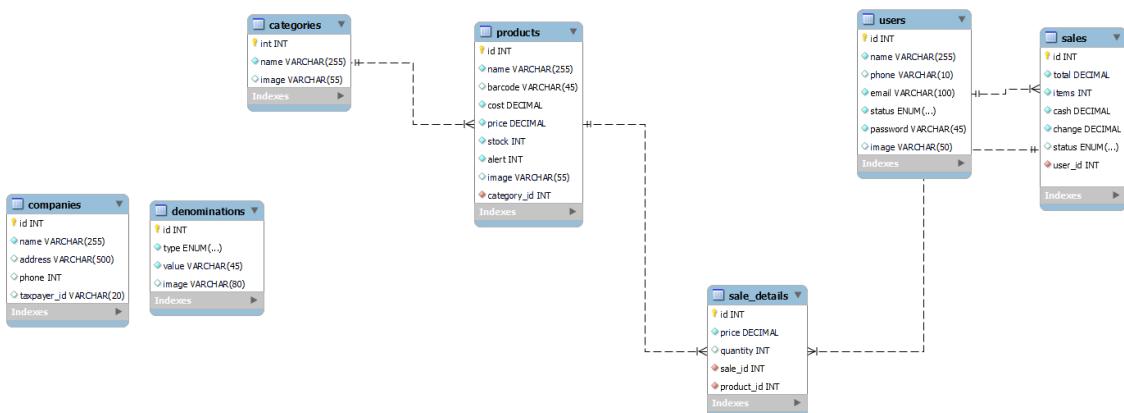
Laravel Mix v6.0.43
Compiled Successfully in 4821ms
File Size
/js/app.js 2.23 MiB
css/app.css 202 Kib
webpack compiled successfully
C:\laragon\www\ventaslite>

```



Base de Datos

Vamos a crear 7 tablas (categories, products, users, sales, companies, denominations y sale_details) con las relaciones correspondientes.



Para hacer las migraciones a la base de datos, debemos escribir en la terminal (dentro de la carpeta del proyecto) **php artisan make:model** y el nombre de la table en cuestión, acabando con -m al final.

```
C:\laragon\www\ventaslite (master)
λ php artisan make:model Denomination -m
Model created successfully.
Created Migration: 2022_03_31_172037_create_denominations_table

C:\laragon\www\ventaslite (master)
λ php artisan make:model Category -m
Model created successfully.
Created Migration: 2022_03_31_172729_create_categories_table

C:\laragon\www\ventaslite (master)
λ php artisan make:model Product -m
Model created successfully.
Created Migration: 2022_03_31_173104_create_products_table
```

Una vez creadas todas las migraciones, en la carpeta Models, ubicada dentro de la carpeta app, nos aparece un fichero por cada migración hecha, todos con su respectivo nombre. Debemos abrirlos y escribir todas las tablas que queremos crear.

```
<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class Product extends Model
{
    use HasFactory;

    protected $fillable = ['name', 'barcode', 'cost', 'price', 'stock', 'alerts', 'image', 'category_id'];
}
```

Finalmente en la carpeta migrations dentro de database, también se nos deben haber creado tantos archivos como migraciones hemos hecho. Allí debemos escribir cada tabla con su tipo de datos, el máximo de caracteres si es posible y alguna característica necesaria como nullable o los enums.

```
public function up()
{
    Schema::create('products', function (Blueprint $table) {
        $table->id();
        $table->string('name',255);
        $table->string('barcode',25)->nullable();
        $table->decimal('cost',10,2)->default(0);
        $table->decimal('price',10,2)->default(0);
        $table->integer('stock');
        $table->integer('alerts');
        $table->string('stock',100)->nullable();

        $table->unsignedBigInteger('category_id');
        $table->foreign('category_id')->references('id')->on('categories');

        $table->timestamps();
    });
}
```

GitHub

Comandos mas

GIT INIT -> crear el repositorio (.git)

GIT STATUS -S -> ver estatus de los archivos

TOUCH .GITIGNORE -> archivos no sincronizables al repositorio

GIT REFLG ->ver log histórico de los commits

GIT ADD . ->movemos todos los archivos pendientes al stage area

GIT COMMIT -M "COMENTARIO" ->ver log histórico de los commits

GIT RESET –HARD COMMIT ID ->regresar a un commit en el tiempo

```
sergicastin@PCSG-DAW-07 MINGW64 /c/laragon/www/ventaslite
$ git init
Initialized empty Git repository in C:/laragon/www/ventaslite/.git/
sergicastin@PCSG-DAW-07 MINGW64 /c/laragon/www/ventaslite (master)
$ |
```

```
sergicastin@PCSG-DAW-07 MINGW64 /c/laragon/www/ventaslite (master)
$ git add .
warning: LF will be replaced by CRLF in .editorconfig.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in .env.example.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in .gitattributes.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in .gitignore.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in .styleci.yml.
```

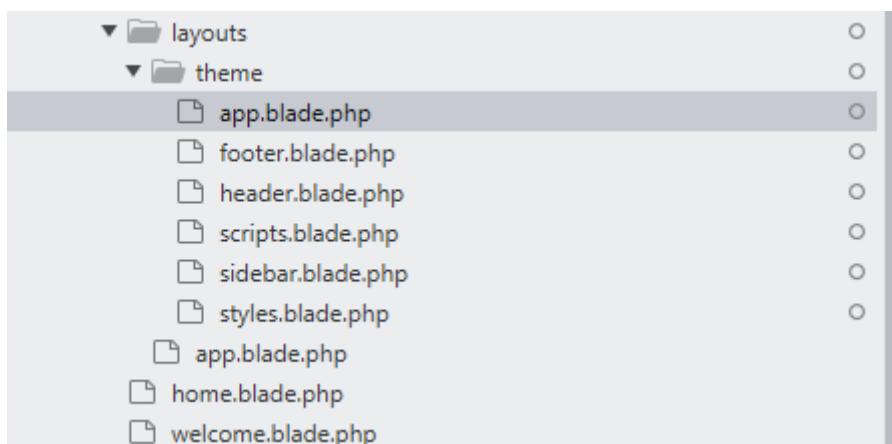
```
sergicastin@PCSG-DAW-07 MINGW64 /c/laragon/www/ventaslite (master)
$ git status -s
A .editorconfig
A .env.example
A .gitattributes
A .gitignore
A .styleci.yml
A README.md
A app\Console\Kernel.php
A app\Exceptions\Handler.php
A app\Http\Controllers\Auth\ConfirmPasswordController.php
A app\Http\Controllers\Auth\ForgotPasswordController.php
```

```
sergicastin@PCSG-DAW-07 MINGW64 /c/laragon/www/ventaslite (master)
$ git commit -m "primer commit del proyecto vacio"
[master (root-commit) 3fd62f9] primer commit del proyecto vacio
 105 files changed, 40852 insertions(+)
 create mode 100644 .editorconfig
 create mode 100644 .env.example
 create mode 100644 .gitattributes
 create mode 100644 .gitignore
 create mode 100644 .styleci.yml
 create mode 100644 README.md
 create mode 100644 app\Console\Kernel.php
```

```
sergicastin@PCSG-DAW-07 MINGW64 /c/laragon/www/ventaslite (master)
$ git log --oneline
3fd62f9 (HEAD -> master) primer commit del proyecto vacio
sergicastin@PCSG-DAW-07 MINGW64 /c/laragon/www/ventaslite (master)
$ ...
```

Plantilla del Proyecto

Dividimos nuestra plantilla en diferentes secciones, app.blade, footer.blade, header.blade, scripts.blade, sidebar.blade y styles.blade. En esos archivos pegamos su parte correspondiente del index que hemos descargado.



Donde hemos cortado el código escribimos un include para decir que la parte de código que está buscando realmente está en otro archivo.

```
<!-- BEGIN CONTENT AREA -->
<div id="content" class="main-content">
    <div class="layout-px-spacing">
        @yield('content')
    </div>

    @include('layouts.theme.footer')
</div>
<!-- END CONTENT AREA -->

</div>
<!-- END MAIN CONTAINER -->
```

Finalmente, copiamos las carpetas de la plantilla y las ponemos en la carpeta de nuestro proyecto.

equipo > SISTEMA (C:) > laragon > www > ventaslite > public

Nombre	Fecha de modificación
assets	05/04/2022 18:44
bootstrap	05/04/2022 18:45
css	05/04/2022 18:45
js	29/03/2022 18:37
plugins	05/04/2022 18:44
Resources	05/04/2022 18:45
sass	05/04/2022 18:45
.htaccess	29/03/2022 18:28
favicon.ico	29/03/2022 18:28
index.php	29/03/2022 18:28
mix-manifest.json	29/03/2022 18:37
robots.txt	29/03/2022 18:28

Editamos el sidebar y ponemos los nombres correspondientes a las secciones:

```
sidebar.blade.php x header.blade.php x

        </div>
        <span>CATEGORIAS</span>
    </div>
    </a>
    <svg xmlns="http://www.w3.org/2000/svg" width="24" height="24" viewBox="0 0 24 24" fill="none" stroke="currentColor" stroke-width="2" stroke-linecap="round" stroke-linejoin="round" class="feather feather-chevron-left"><polyline points="15 18 9 12 15 6"></polyline></svg>
</li>

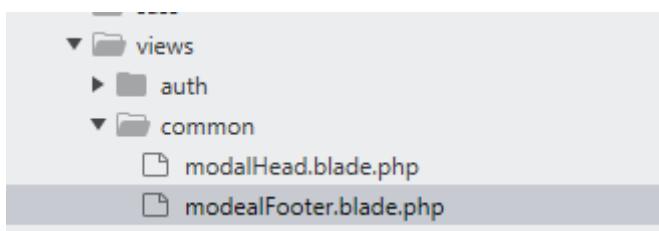
<li class="menu">
    <a href="#app" data-active="false" class="menu-toggle">
        <div class="base-menu">
            <div class="base-icons">
                <svg xmlns="http://www.w3.org/2000/svg" width="24" height="24" viewBox="0 0 24 24" fill="none" stroke="currentColor" stroke-width="2" stroke-linecap="round" stroke-linejoin="round" class="feather feather-cpu"><rect x="4" y="4" width="16" height="16" rx="2"></rect><rect x="9" y="9" width="6" height="6"></rect><line x1="9" y1="1" x2="9" y2="4"></line><line x1="15" y1="1" x2="15" y2="4"></line><line x1="9" y1="20" x2="9" y2="23"></line><line x1="15" y1="20" x2="15" y2="23"></line><line x1="20" y1="9" x2="23" y2="9"></line><line x1="20" y1="20" x2="23" y2="20"></line><line x1="15" y1="23" x2="15" y2="14"></line><line x1="1" y1="9" x2="4" y2="9"></line><line x1="1" y1="14" x2="4" y2="14"></line></svg>
            <span>PRODUCTOS</span>
        </div>
    </a>
    <svg xmlns="http://www.w3.org/2000/svg" width="24" height="24" viewBox="0 0 24 24" fill="none" stroke="currentColor" stroke-width="2" stroke-linecap="round" stroke-linejoin="round" class="feather feather-chevron-left"><polyline points="15 18 9 12 15 6"></polyline></svg>
</li>

<li class="menu">
    <a href="#UIKit" data-active="false" class="menu-toggle">
        <div class="base-menu">
            <div class="base-icons">
                <svg xmlns="http://www.w3.org/2000/svg" width="24" height="24" viewBox="0 0 24 24" fill="none" stroke="currentColor" stroke-width="2" stroke-linecap="round" stroke-linejoin="round" class="feather feather-zap"><polygon points="13 2 3 14 12 14 11 22 21 10 12 10 13 2"></polygon></svg>
            </div>
            <span>VENTAS</span>
        </div>
    </a>

```



En la carpeta views, creamos una nueva carpeta llamada common, en ella creamos dos archivos: modalHead.blade y modalFooter.blade. Tal como muestra la siguiente imagen.



modalHead.blade.php:

```

modalHead.blade.php      sidebar.blade.php      header.blade.php      blank.blade.php
<div wire:ignore.self class="modal fade" tabindex="-1" role="dialog">
  <div class="modal-dialog modal-lg" role="document">
    <div class="modal-content">
      <div class="modal-header">
        <h5 class="modal-title text-white">
          <b>{$componentName}</b> | {$selected_id > 0 ? 'EDITAR' : 'CREAR'}
        </h5>
        <h6 class="text-center text-warning" wire:loading>POR FAVOR ESPERE</h6>
      </div>
      <div class="modal-body">

```

modalFooter.blade.php:

```

modalFooter.blade.php      modalHead.blade.php      sidebar.blade.php      header.blade.php      blank.blade.php      init.php
</div>
<div class="modal-footer">
  <button type="button" wire:click.prevent="resetUI()" class="btn btn-dark close-btn text-info" data-dismiss="modal">CERRAR</button>
  @if($selected:id < 1)
    <button type="button" wire:click.prevent="Store()" class="btn btn-dark close-modal">GUARDAR</button>
  @else
    <button type="button" wire:click.prevent="Update()" class="btn btn-dark close-modal">CERRAR</button>
  @endif
</div>
</div>
</div>

```

Creamos también la carpeta livewire con el archivo blank.blade.php. Junto a ese archivo, creamos otra carpeta startpage con otro archivo llamado init.php en su interior.

DESCRIPCIÓN	IMAGEN	ACTIONS
Category Name		View Edit

Para hacer el buscador, creamos un archivo llamado searchbox.blade.php en la carpeta common dentro de views y escribimos:

A screenshot of a code editor showing a file structure. At the top, there are tabs for 'searchbox.blade.php', 'modalFooter.blade.php', 'modalHead.blade.php', 'sidebar.blade.php', and 'app.blade.php — layout'. Below the tabs, the code editor displays a portion of a blade template:

```
<div class="row justify-content-between">
    <div class="col-lg-4 col-md-4 col-sm-12">
        <div class="input-group mb-4">
            <div class="input-group-prepend">
                <span class="input-group-text input-gp">
                    <i class="fas fa-search"></i>
                </span>
            </div>
            <input type="text" wire:model="search" placeholder="Buscar" class="form-control">
        </div>
    </div>
</div>
```

Plugins y Complementos

Instalamos el siguiente paquete:

```
\ composer require livewire/livewire
Info from https://repo.packagist.org: #StandWithUkraine
Using version ^2.10 for livewire/livewire
./composer.json has been updated
Running composer update livewire/livewire
Loading composer repositories with package information
Info from https://repo.packagist.org: #StandWithUkraine
Updating dependencies
Lock file operations: 1 install, 0 updates, 0 removals
- Locking livewire/livewire (v2.10.4)
Writing lock file
Installing dependencies from lock file (including require-dev)
Package operations: 1 install, 0 updates, 0 removals
- Downloading livewire/livewire (v2.10.4)
- Installing livewire/livewire (v2.10.4): Extracting archive
Package swiftmailer/swiftmailer is abandoned, you should avoid
```

Modificamos el archivo de los scripts y el de los styles:

Scripts:

```
<script>
    function notify(msg, option = 1){
        Snackbar.show({
            text: msg.toUpperCase(),
            actionText: 'Cerrar',
            actionTextColor: '#fff',
            backgroundColor: option == 1 ? '#3b3f5c' : '#e7515a',
            pos: 'top-right'
        });
    }
</script>

@livewareScripts
```

Styles:

```
<style>
    aside{
        display: none!important;
    }
    .page-item.active .page-link{
        z-index: 3;
        color: white;
        background-color: #3b3f5c;
        border-color: #3b3f5c;
    }
    @media (max-width: 480px) {
        .mtmobile{
            margin-bottom: 20px!important;
        }
        .mtmobile{
            margin-bottom: 10px!important;
        }
        .mtmobile{
            display: none!important;
        }
        .mtmobile{
            display: block;
        }
    }
</style>

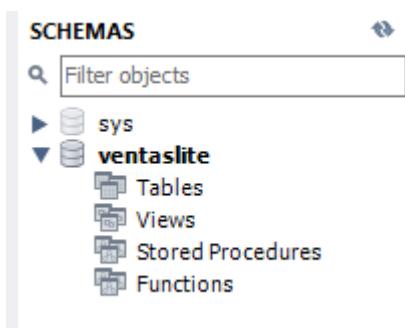
@livewareStyles
```

Y finalmente publicamos el archivo de configuración con el siguiente comando:

```
C:\laragon\www\ventaslite (master)
λ php artisan livewire:publish --config
Copied File [\vendor\livewire\livewire\config\livewire.php] To [\config\livewire.php]
Publishing complete.
```

Migraciones

Automáticamente nuestro proyecto lleva ya una base de datos creada con el mismo nombre que el proyecto. El usuario es root y no lleva contraseña.



Para hacer la migración a la base de datos, debemos escribir en una consola dentro del proyecto php artisan migrate.

```
C:\laragon\www\ventaslite (master)
λ php artisan migrate
Migration table created successfully.
Migrating: 2014_10_12_000000_create_users_table
Migrated: 2014_10_12_000000_create_users_table (54.72ms)
Migrating: 2014_10_12_100000_create_password_resets_table
Migrated: 2014_10_12_100000_create_password_resets_table (68.38ms)
Migrating: 2019_08_19_000000_create_failed_jobs_table
Migrated: 2019_08_19_000000_create_failed_jobs_table (47.62ms)
Migrating: 2019_12_14_000001_create_personal_access_tokens_table
Migrated: 2019_12_14_000001_create_personal_access_tokens_table (70.96ms)
Migrating: 2022_03_31_170921_create_companies_table
Migrated: 2022_03_31_170921_create_companies_table (30.69ms)
Migrating: 2022_03_31_172037_create_denominations_table
Migrated: 2022_03_31_172037_create_denominations_table (65.58ms)
Migrating: 2022_03_31_172729_create_categories_table
Migrated: 2022_03_31_172729_create_categories_table (24.75ms)
Migrating: 2022_03_31_173104_create_products_table
Migrated: 2022_03_31_173104_create_products_table (87.10ms)
Migrating: 2022_04_05_155839_create_sales_table
Migrated: 2022_04_05_155839_create_sales_table (86.99ms)
Migrating: 2022_04_05_160515_create_sale_details_table
Migrated: 2022_04_05_160515_create_sale_details_table (147.26ms)
```

```
C:\laragon\www\ventaslite (master)
λ |
```

Probamos a registrar un usuario:

Register

Name	Sergi Castillo
Email Address	sergicastin@monlau.com
Password	*****
Confirm Password	*****

Register

El usuario se crea perfectamente y se inicia sesión automáticamente.



Si cerramos sesión e intentamos iniciar sesión también funciona.

Seeders

Creamos los seeders con el siguiente comando

```
C:\laragon\www\ventaslite (master)
λ php artisan make:seeder DenominationSeeder
Seeder created successfully.
```

Creamos cada categoría de productos añadiendo un nombre y una imagen acorde a la categoría.

```
use Illuminate\Database\Seeder;
use App\Models\Category;

class CategorySeeder extends Seeder
{
    /**
     * Run the database seeds.
     *
     * @return void
     */
    public function run()
    {
        Category::create([
            'name' => 'CURSOS',
            'image' => 'https://dummyimage.com/200x150/5c5756/fff'
        ]);

        Category::create([
            'name' => 'TENIS',
            'image' => 'https://dummyimage.com/200x150/5c5756/fff'
        ]);

        Category::create([
            'name' => 'CELULARES',
            'image' => 'https://dummyimage.com/200x150/5c5756/fff'
        ]);

        Category::create([
            'name' => 'COMPUTADORAS',
            'image' => 'https://dummyimage.com/200x150/5c5756/fff'
        ]);
    }
}
```

Creamos también cada forma de pago, ya sean en billetes, monedas u otros, asignando el tipo y valor.

The screenshot shows a code editor with three tabs at the top: 'DenominationSeeder.php' (active), 'ProductSeeder.php', and 'CategorySeeder.php'. The code in the 'DenominationSeeder.php' tab is as follows:

```
DenominationSeeder.php  x  ProductSeeder.php  x  CategorySeeder.php  x

public function run()
{
    Denomination::create([
        'type' => 'BILLETE',
        'value' => '500'
    ]);

    Denomination::create([
        'type' => 'BILLETE',
        'value' => '200'
    ]);

    Denomination::create([
        'type' => 'BILLETE',
        'value' => '100'
    ]);

    Denomination::create([
        'type' => 'BILLETE',
        'value' => '50'
    ]);

    Denomination::create([
        'type' => 'BILLETE',
        'value' => '20'
    ]);
}
```

Creamos un producto por categoría asignando el nombre, el coste, su precio, el código de barras, nuestro stock, una alerta, el id de la categoría del producto y una imagen del producto.

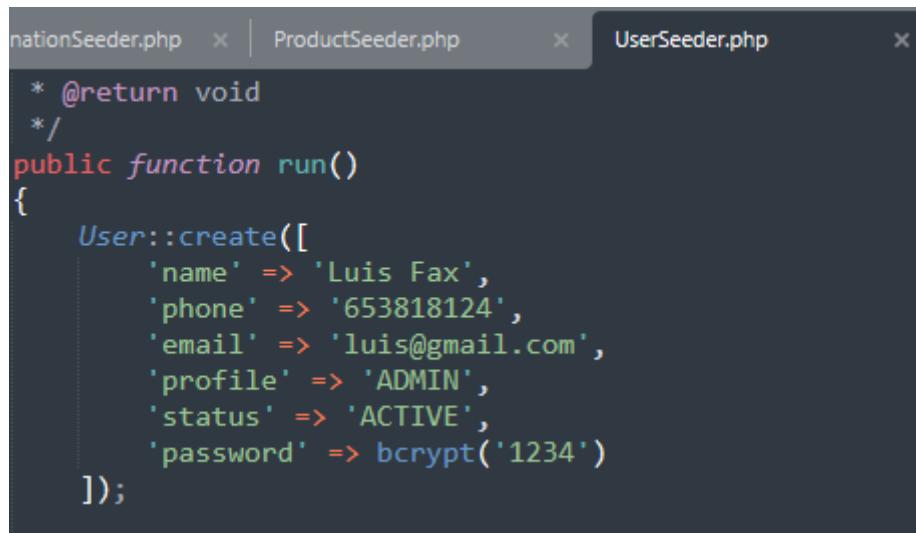
The screenshot shows a code editor with three tabs at the top: 'DenominationSeeder.php', 'ProductSeeder.php' (active), and 'CategorySeeder.php'. The code in the 'ProductSeeder.php' tab is as follows:

```
DenominationSeeder.php  x  ProductSeeder.php  x  CategorySeeder.php  x

*/
public function run()
{
    Product::create([
        'name' => 'LARAVEL Y LIVEWIRE',
        'cost' => 200,
        'price' => 350,
        'barcode' => 7845694365,
        'stock' => 1000,
        'alerts' => 10,
        'category_id' => 1,
        'image' => 'curso.png'
    ]);
}
```

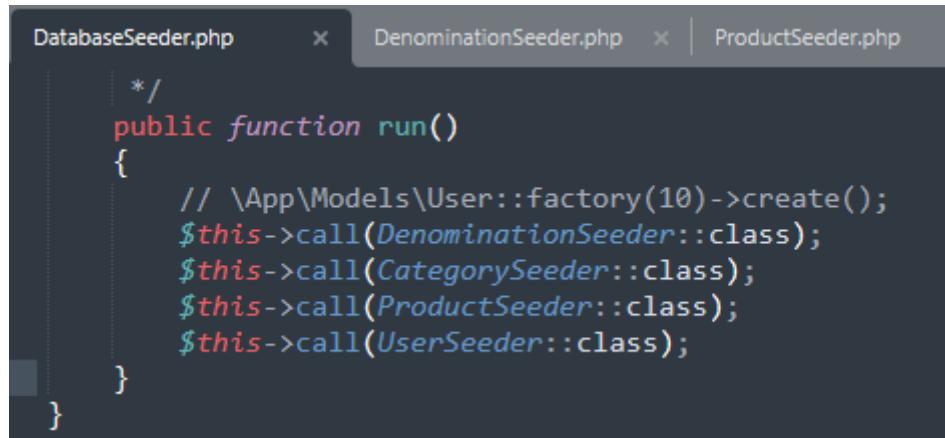
También creamos un seed de los usuarios y dos usuarios en él.

```
C:\laragon\www\ventaslite (master)
λ php artisan make:seeder UserSeeder
Seeder created successfully.
```



```
nationSeeder.php  × | ProductSeeder.php  × | UserSeeder.php  ×
* @return void
*/
public function run()
{
    User::create([
        'name' => 'Luis Fax',
        'phone' => '653818124',
        'email' => 'luis@gmail.com',
        'profile' => 'ADMIN',
        'status' => 'ACTIVE',
        'password' => bcrypt('1234')
    ]);
}
```

Ejecutamos los seeders:



```
DatabaseSeeder.php  × | DenominationSeeder.php  × | ProductSeeder.php  ×
*/
public function run()
{
    // \App\Models\User::factory(10)->create();
    $this->call(DenominationSeeder::class);
    $this->call(CategorySeeder::class);
    $this->call(ProductSeeder::class);
    $this->call(UserSeeder::class);
}
```

```
C:\laragon\www\ventaslite (master)
└ php artisan migrate:fresh --seed
Dropped all tables successfully.
Migration table created successfully.
ns , function (Blueprint $table) {
Migrating: 2014_10_12_00000_create_users_table
Migrated: 2014_10_12_00000_create_users_table (47.94ms) [R:0]
Migrating: 2014_10_12_100000_create_password_resets_table
Migrated: 2014_10_12_100000_create_password_resets_table (58.39ms)
Migrating: 2019_08_19_000000_create_failed_jobs_table
Migrated: 2019_08_19_000000_create_failed_jobs_table (43.21ms)
Migrating: 2019_12_14_000001_create_personal_access_tokens_table
Migrated: 2019_12_14_000001_create_personal_access_tokens_table (102.36ms)
Migrating: 2022_03_31_170921_create_companies_table
Migrated: 2022_03_31_170921_create_companies_table (28.10ms)
Migrating: 2022_03_31_172037_create_denominations_table
Migrated: 2022_03_31_172037_create_denominations_table (29.20ms)
Migrating: 2022_03_31_172729_create_categories_table
Migrated: 2022_03_31_172729_create_categories_table (20.55ms)
Migrating: 2022_03_31_173104_create_products_table
Migrated: 2022_03_31_173104_create_products_table (77.91ms)
Migrating: 2022_04_05_155839_create_sales_table
Migrated: 2022_04_05_155839_create_sales_table (70.74ms)
Migrating: 2022_04_05_160515_create_sale_details_table
Migrated: 2022_04_05_160515_create_sale_details_table (213.17ms)
Seeding: Database\Seeders\DenominationSeeder
Seeded: Database\Seeders\DenominationSeeder (32.31ms)
Seeding: Database\Seeders\CategorySeeder
Seeded: Database\Seeders\CategorySeeder (19.34ms)
Seeding: Database\Seeders\ProductSeeder
Seeded: Database\Seeders\ProductSeeder (15.54ms)
Seeding: Database\Seeders\UserSeeder
Seeded: Database\Seeders\UserSeeder (194.79ms)
Database seeding completed successfully.

C:\laragon\www\ventaslite (master)
```

Y vemos el resultado:

The screenshot shows a MySQL Workbench interface with a query editor and a result grid. The query editor contains the following SQL code:

```
1 select * from categories;
```

The result grid displays the following data:

	id	name	image	created_at	updated_at
▶	1	CURSOS	https://dummyimage.com/200x150/5c5756/fff	2022-04-19 16:16:48	2022-04-19 16:16:48
▶	2	TENIS	https://dummyimage.com/200x150/5c5756/fff	2022-04-19 16:16:48	2022-04-19 16:16:48
▶	3	CELULARES	https://dummyimage.com/200x150/5c5756/fff	2022-04-19 16:16:48	2022-04-19 16:16:48
▶	4	COMPUTADORAS	https://dummyimage.com/200x150/5c5756/fff	2022-04-19 16:16:48	2022-04-19 16:16:48
*	NULL	NULL	NULL	NULL	NULL

Componente Categories

Editamos el archivo CategoriesController.php:

```
<?php  
namespace App\Http\Livewire;  
use Livewire\Component;  
use App\Models\Category;  
use Illuminate\Support\Facades\Storage;  
use Livewire\WithFileUploads;  
use Livewire\WithPagination;  
  
class CategoriesController extends Component  
{  
  
    use WithFileUploads;  
    use WithPagination;  
  
    public $name, $search, $image, $selected_id, $pageTitle, $componentName;  
    private $pagination = 5;  
  
    public function render()  
    {  
        $data = Category::all();  
        return view('livewire.category.categories', ['categories' => $data])  
            ->extends('layouts.theme.app')  
            ->section('content');  
    }  
}
```

Ahora la página de categorías ya funciona.

Editando el archivo categories.blade.php conseguimos configurar el botón de eliminar registro tal como muestro en las siguientes imágenes:

```
<script>  
document.addEventListener('DOMContentLoaded', function(){  
  
    window.livewire.on('category-added', msg =>{  
        $('#theModal').modal('hide');  
        noty(msg)  
    })  
  
    window.livewire.on('category-updated', msg =>{  
        $('#theModal').modal('hide');  
        noty(msg)  
    })  
  
    window.livewire.on('category-deleted', msg =>{  
        noty(msg)  
    })  
  
    window.livewire.on('hide-modal', msg =>{  
        $('#theModal').modal('hide');  
    })  
  
    window.livewire.on('show-modal', msg =>{  
        $('#theModal').modal('hide');  
    })  
});</script>
```

```

categories.blade.php      x  searchbox.blade.php      x  form.blade.php      x

function Confirm(id) {
    swal({
        title: 'CONFIRMAR',
        text: '¿CONFIRMAS ELIMINAR EL REGISTRO?',
        type: 'warning',
        showCancelButton: true,
        cancelButtonText: 'Cerrar',
        cancelButtonColor: '#fff',
        confirmButtonText: 'Aceptar',
        confirmButtonColor: '#3B3F5C'
    }).then(function(result){
        if (result.value) {
            window.livewire.emit('deleteRow', id)
            swal.close()
        }
    })
}

</script>

```



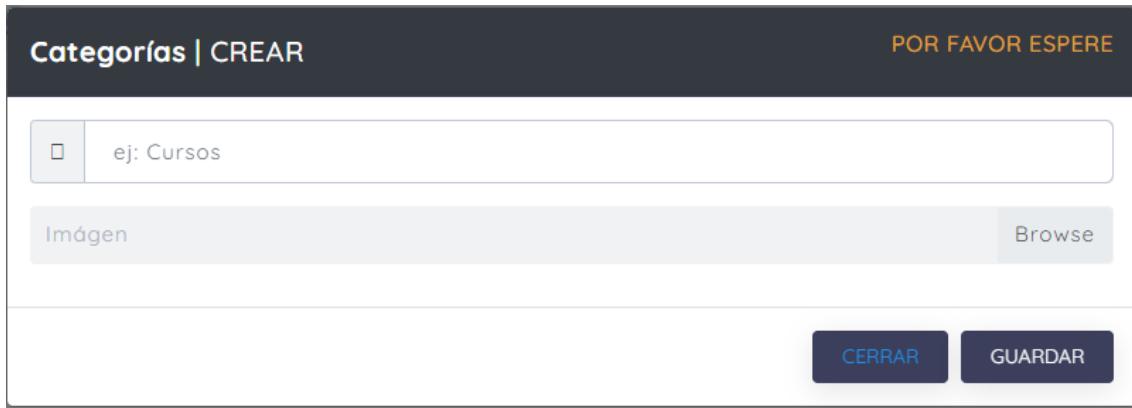
Ahora configuraremos el formulario de agregar:

```

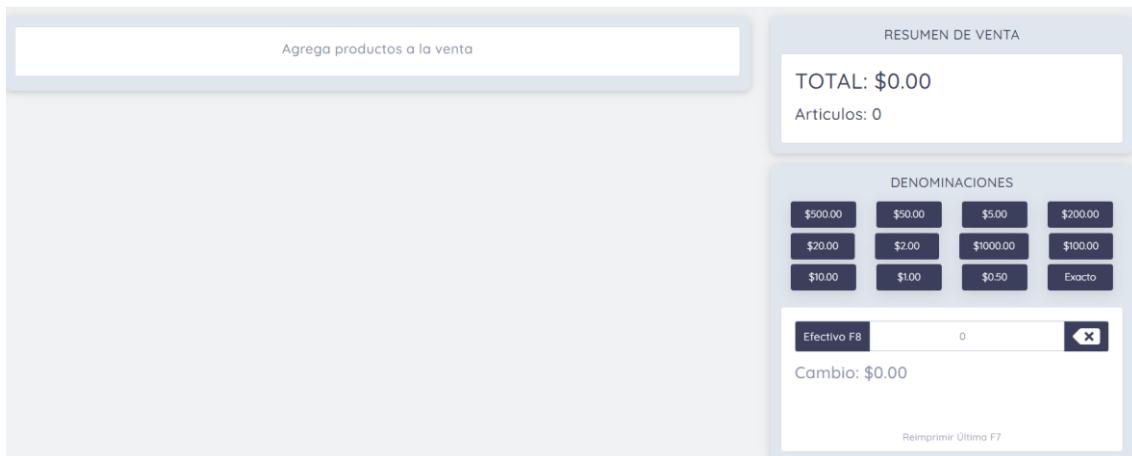
<div class="row">
    <div class="col-sm-12">
        <div class="input-group">
            <div class="input-group-prepend">
                <span class="input-group-text">
                    <span class="fas fa-edit">
                        </span>
                </span>
            </div>
            <input type="text" wire:model.lazy="name" class="form-control" placeholder="ej: Cursos">
        </div>
        @error('name') <span class="text-danger er">{{ $message }}</span> @enderror
    </div>

    <div class="col-sm-12 mt-3">
        <div class="form-group custom-file">
            <input type="file" class="custom-file-input form-control" wire:model="image" accept="image/x-png, image/gif, image/jpeg">
            <label class="custom-file-label">Imagen {{ $image }}</label>
            @error('image') <span class="text-danger er">{{ $message }}</span> @enderror
        </div>
    </div>
</div>

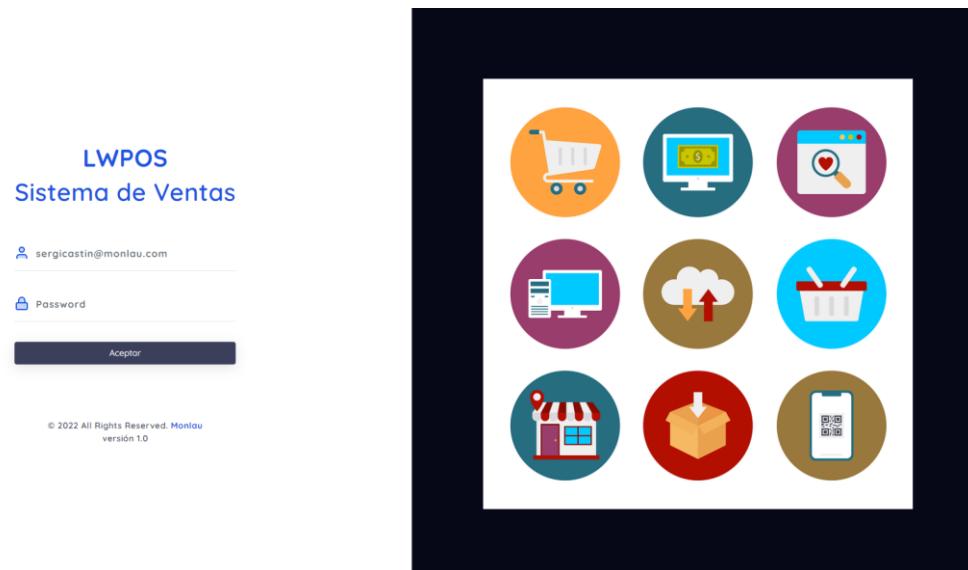
```



Con la plantilla terminada, podemos ver ya los productos a la venta, el resumen de venta y las denominaciones.



Igual pasa con el login, totalmente funcional:



Componentes:

Categorías:

Como el nombre indica, podemos ver las categorías de los productos que vendemos.

Categorías | Listado

Agregar

DESCRIPCIÓN	IMAGEN	ACTIONS
NUEVA		
COMPUTADORAS		
CELULARES2		
TENIS		
CURSOS		

```
<div class="table-responsive">
    <table class="table table-bordered table-striped mt-1">
        <thead class="text-white" style="background: #3B3F5C">
            <tr>
                <th class="table-th text-white">DESCRIPCIÓN</th>
                <th class="table-th text-white text-center">IMAGEN</th>
                <th class="table-th text-white text-center">ACTIONS</th>
            </tr>
        </thead>
        <tbody>
            @foreach($categories as $category)
            <tr>
                <td>
                    <h6>$category->name</h6>
                </td>
                <td class="text-center">
                    <span>
                        
                    </span>
                </td>
                <td class="text-center">
                    @can('category_update')
                    <a href="javascript:void(0)" wire:click="Edit({{$category->id}})" class="btn btn-dark mtmobile" title="Edit">
                        <i class="fas fa-edit"></i>
                    </a>
                    @endcan

                    @if($category->products->count() < 1 ) @can('category_destroy')
                    <a href="javascript:void(0)" onclick="Confirm('{{ $category->id }}')" class="btn btn-dark" title="Delete">
                        <i class="fas fa-trash"></i>
                    </a>
                    @endcan
                    @endif
                </td>
            </tr>
        </tbody>
    </table>
</div>
```

```
<script>
document.addEventListener('DOMContentLoaded', function() {
    window.livewire.on('show-modal', msg => {
        $('#theModal').modal('show')
    });
    window.livewire.on('category-added', msg => {
        $('#theModal').modal('hide')
    });
    window.livewire.on('category-updated', msg => {
        $('#theModal').modal('hide')
    });
});

function Confirm(id) {
    swal({
        title: 'CONFIRMAR',
        text: '¿CONFIRMAS ELIMINAR EL REGISTRO?',
        type: 'warning',
        showCancelButton: true,
        cancelButtonText: 'Cerrar',
        cancelButtonColor: '#fff',
        confirmButtonColor: '#3B3F5C',
        confirmButtonText: 'Aceptar'
    }).then(function(result) {
        if (result.value) {
            window.livewire.emit('deleteRow', id)
            swal.close()
        }
    })
}
</script>
```

Controlador:

Productos

Tenemos un listado con todos los productos creados anteriormente:

Productos Listado							Agregar
DESCRIPCIÓN	BARCODE	CATEGORÍA	PRECIO	STOCK	INV.MÍN.	IMAGEN	ACTIONS
CURSO PHP	98745	CURSOS	100.00	100	10		 
IPHONE 15	7709876541	CELULARES2	1400.00	1000	10		 
LARAVEL Y LIVEWIRE	75010065987	CURSOS	350.00	1000	10		 
PC GAMMER	790654812	COMPUTADORAS	1350.00	1000	10		 
RUNNING NIKE	7609872014	TENIS	1500.00	1000	10		 

Y otro menú para crear un producto nuevo:

Productos | CREAR

Nombre	Código	
<input type="text" value="ej: Curso Laravel"/>	<input type="text" value="ej: 025974"/>	
Costo	Precio	Stock
<input type="text" value="ej: 0.00"/>	<input type="text" value="ej: 0.00"/>	<input type="text" value="ej: 0"/>
Alertas	Categoría	
<input type="text" value="ej: 10"/>	<input type="text" value="Elegir"/>	
Imagen	Browse	
CERRAR GUARDAR		

```

<div class="table-responsive">
  <table class="table table-bordered table-striped mt-1">
    <thead class="text-white" style="background: #3B3F5C;">
      <tr>
        <th class="table-th text-white">DESCRIPCIÓN</th>
        <th class="table-th text-white text-center">BARCODE</th>
        <th class="table-th text-white text-center">CATEGORÍA</th>
        <th class="table-th text-white text-center">PRECIO</th>
        <th class="table-th text-white text-center">STOCK</th>
        <th class="table-th text-white text-center">INV. MIN</th>
        <th class="table-th text-white text-center">IMAGEN</th>
        <th class="table-th text-white text-center">ACTIONS</th>
      </tr>
    </thead>
    <tbody>
      @foreach($data as $product)
      <tr>
        <td>
          <h6 class="text-left">{{ $product->name }}</h6>
        </td>
        <td>
          <h6 class="text-center">{{ $product->barcode }}</h6>
        </td>
        <td>
          <h6 class="text-center">{{ $product->category }}</h6>
        </td>
        <td>
          <h6 class="text-center">{{ $product->price }}</h6>
        </td>
      </tr>
    
```

```

<script>
  document.addEventListener('DOMContentLoaded', function() {
    window.livewire.on('product-added', msg => {
      $('#theModal').modal('hide')
    });
    window.livewire.on('product-updated', msg => {
      $('#theModal').modal('hide')
    });
    window.livewire.on('product-deleted', msg => {
      // noty
    });
    window.livewire.on('modal-show', msg => {
      $('#theModal').modal('show')
    });
    window.livewire.on('modal-hide', msg => {
      $('#theModal').modal('hide')
    });
    window.livewire.on('hidden.bs.modal', msg => {
      $('.er').css('display', 'none')
    });
    $('#theModal').on('hidden.bs.modal', function(e) {
      $('.er').css('display', 'none')
    })
    $('#theModal').on('shown.bs.modal', function(e) {
      $('.product-name').focus()
    })
  });
}

```

```

function Confirm(id) {
    swal({
        title: 'CONFIRMAR',
        text: '¿CONFIRMAS ELIMINAR EL REGISTRO?',
        type: 'warning',
        showCancelButton: true,
        cancelButtonText: 'Cerrar',
        cancelButtonColor: '#fff',
        confirmButtonColor: '#3B3F5C',
        confirmButtonText: 'Aceptar'
    }).then(function(result) {
        if (result.value) {
            window.livewire.emit('deleteRow', id)
            swal.close()
        }
    })
}

```

Denominaciones/Monedas

Tenemos una lista con todos los métodos de pago y sus variantes, ya sean monedas, billetes u otros. Junto a todas sus variables como los distintos valores.

Denominaciones Listado			
TIPO	VALOR	IMAGEN	ACTIONS
OTRO	\$0.00		
MONEDA	\$0.50		
MONEDA	\$1.00		
MONEDA	\$2.00		
MONEDA	\$5.00		

« 1 2 3 4 5 »

Podemos agregar otro valor a nuestro grupo de monedas o billetes con el siguiente formulario:

Denominaciones | CREAR

Tipo <div style="border: 1px solid #ccc; padding: 5px; width: 100%;"> Elegir </div> <div style="border: 1px solid #ccc; padding: 5px; width: 100%; background-color: #f0f0f0; margin-top: 5px;"> <ul style="list-style-type: none"> <li style="border-bottom: 1px solid #ccc; padding: 2px 0; display: flex; align-items: center;"> Elegir BILLETE <li style="padding: 2px 0;">MONEDA <li style="padding: 2px 0;">OTRO </div>	Value <div style="border: 1px solid #ccc; padding: 5px; width: 100%; position: relative;"> <div style="display: flex; align-items: center;"> Edit <input style="width: 100%; height: 30px; border: none; padding: 0 10px; font-size: 14px;" type="text" value="ej: 100.00"/> </div> <div style="position: absolute; right: -10px; top: 0; width: 10px; height: 10px; background-color: #f0f0f0; border-radius: 50%;"></div> </div> <div style="display: flex; justify-content: space-between; width: 100%; margin-top: 10px;"> Browse </div>
CERRAR GUARDAR	

```

@include('common.searchbox')

<div class="widget-content">

    <div class="table-responsive">
        <table class="table table-bordered table-striped mt-1">
            <thead class="text-white" style="background: #3B3F5C">
                <tr>
                    <th class="table-th text-white">TIPO</th>
                    <th class="table-th text-white text-center">VALOR</th>
                    <th class="table-th text-white text-center">IMAGEN</th>
                    <th class="table-th text-white text-center">ACTIONS</th>
                </tr>
            </thead>

```

```

<tbody>
    @foreach($data as $coin)
        <tr>
            <td><h6>{{$coin->type}}</h6></td>
            <td><h6 class="text-center">${{number_format($coin->value,2)}}</h6></td>
            <td class="text-center">
                <span>
                    
                </span>
            </td>
            <td class="text-center">
                <a href="javascript:void(0)" wire:click="Edit({{$coin->id}})"
                   class="btn btn-dark mtmobile" title="Edit">
                    <i class="fas fa-edit"></i>
                </a>

                <a href="javascript:void(0)" onclick="Confirm('{{ $coin->id }}')"
                   class="btn btn-dark" title="Delete">
                    <i class="fas fa-trash"></i>
                </a>
            </td>
        </tr>
    @endforeach

```

```

<script>
    document.addEventListener('DOMContentLoaded', function(){
        window.livewire.on('item-added', msg =>{
            $('#theModal').modal('hide')
        });
        window.livewire.on('item-updated', msg =>{
            $('#theModal').modal('hide')
        });
        window.livewire.on('item-deleted', msg =>{
            // noty
        });
        window.livewire.on('show-modal', msg =>{
            $('#theModal').modal('show')
        });
        window.livewire.on('modal-hide', msg =>{
            $('#theModal').modal('hide')
        });
        $('#theModal').on('hidden.bs.modal', function (e) {
            $('.er').css('display', 'none')
        })
    })

```

```

function Confirm(id)
{
    swal({
        title: 'CONFIRMAR',
        text: '¿CONFIRMAS ELIMINAR EL REGISTRO?',
        type: 'warning',
        showCancelButton: true,
        cancelButtonText: 'Cerrar',
        cancelButtonColor: '#ffff',
        confirmButtonColor: '#3B3F5C',
        confirmButtonText: 'Aceptar'
    }).then(function(result) {
        if(result.value){
            window.livewire.emit('deleteRow', id)
            swal.close()
        }
    })
}

```

Ventas

Podemos llamarlo carrito, se ven todos los productos añadidos al carrito y podemos calcular el cambio del cliente.

The screenshot shows a two-panel interface. On the left, a 'RESUMEN DE VENTA' panel displays a table of items in the cart:

DESCRIPCIÓN	PRECIO	CANT	IMPORTE	ACTIONS
IPHONE 15	\$1,000.00	1	\$1,000.00	- +
PC GAMMER	\$850.00	1	\$850.00	- +
RUNNING NIKE	\$500.00	1	\$500.00	- +

To the right, a 'RESUMEN DE VENTA' panel shows payment details:

- TOTAL: \$2,350.00**
- Artículos: 0**
- DENOMINACIONES** (Buttons for \$500.00, \$50.00, \$5.00, \$200.00, \$20.00, \$2.00, \$12.00, \$100.00, \$100.00, \$10.00, \$1.00, \$0.50)
- Exento**
- Efectivo F8** (Buttons for 0, X)
- Cambio: \$0.00**
- CANCELAR F4**

Usuarios:

En el listado de los usuarios nos aparecen todos los usuarios de la base de datos con los datos que contiene, como el teléfono, el email, el estatus de la cuenta o si es administrador o trabajador.

The screenshot shows a 'Usuarios | Listado' page with a search bar and a table of users:

USUARIO	TELÉFONO	EMAIL	ESTATUS	PERFIL	IMÁGEN	ACTIONS
casti	653818127	casti@monlau.com	ACTIVE	ADMIN ROLES:		editar borrar
Luis Fox		luisfaax@gmail.com	ACTIVE	ADMIN ROLES:ADMIN		editar borrar
Melisa Albohat		melisa@gmail.com	ACTIVE	EMPLOYEE ROLES:EMPLOYEE		editar borrar

Pagination controls at the bottom: < 1 2 >

Entonces tenemos dos botones por usuario que nos permite editar o eliminar dicho usuario.

Arriba tenemos también un botón para agregar usuarios:

Usuarios | CREAR

Nombre	Teléfono
ej: Rob Seu	ej: 351 115 9550
Email	Contraseña
ej: mancar@monlau.com	
Estatus	Asignar Role
Elegir	Elegir
Imagen de Perfil	
<input type="button" value="Seleccionar archivo"/> Ninguno a...hivo selec.	
CERRAR GUARDAR	

```
@include('common.searchbox')

<div class="widget-content">

  <div class="table-responsive">
    <table class="table table-bordered table-striped mt-1">
      <thead class="text-white" style="background: #3B3F5C">
        <tr>
          <th class="table-th text-white">USUARIO</th>
          <th class="table-th text-white text-center">TELÉFONO</th>
          <th class="table-th text-white text-center">EMAIL</th>
          <th class="table-th text-white text-center">ESTATUS</th>
          <th class="table-th text-white text-center">PERFIL</th>
          <th class="table-th text-white text-center">IMÁGEN</th>
          <th class="table-th text-white text-center">ACTIONS</th>
        </tr>
      </thead>
      <tbody>
        @foreach($data as $r)
          <td><h6>{{ $r->name }}</h6></td>
          <td class="text-center"><h6>{{ $r->phone }}</h6></td>
          <td class="text-center"><h6>{{ $r->email }}</h6></td>
          <td class="text-center">
            <span class="badge {{ $r->status == 'Active' ? 'badge-success' : 'badge-danger' }} text-uppercase">{{ $r->status }}</span>
          </td>
          <td class="text-center text-uppercase">
            <h6>{{ $r->profile }}</h6>
            <small><b>Roles:</b><br>{{ implode(', ', $r->getRoleNames()->toArray()) }}</small>
          </td>
          <td class="text-center">
            @if($r->image != null)
              
            @endif
          </td>
        @endforeach
      </tbody>
    </table>
  </div>
</div>
```

Corte de Caja

Podemos consultar las ventas hechas de un usuario en concreto, seleccionando el usuario y las fechas de inicio y final. Entonces nos muestra la cantidad de ventas con el total gastado, los ítems y la fecha.

Corte de Caja

Usuario: Luis Fox Fecha inicial: 01/06/2021 Fecha final: 30/06/2021 Consultar Imprimir

Ventas Totales: \$11,250.00 Artículos: 15

FOLIO	TOTAL	ITEMS	FECHA
5	\$2,350.00	3	2021-06-03 04:15:25
6	\$2,350.00	3	2021-06-03 04:18:37
7	\$2,350.00	3	2021-06-01 00:33:27
8	\$2,700.00	3	2021-06-23 22:33:45
9	\$1,500.00	3	2021-06-24 22:34:52

```
<div class="widget-content">
    <div class="row">
        <div class="col-sm-12 col-md-3">
            <div class="form-group">
                <label>Usuario</label>
                <select wire:model="userid" class="form-control">
                    <option value="0" disabled>Elegir</option>
                    @foreach($users as $u)
                    <option value="{{ $u->id }}>{{ $u->name }}</option>
                    @endforeach
                </select>
                @error('userid') <span class="text-danger">{{$message}}</span>@enderror
            </div>
        </div>

        <div class="col-sm-12 col-md-3">
            <div class="form-group">
                <label>Fecha inicial</label>
                <input type="date" wire:model.lazy="fromDate" class="form-control">
                @error('fromDate') <span class="text-danger">{{$message}}</span>@enderror
            </div>
        </div>
    </div>
```

```
<div class="col-sm-12 col-md-3">
    <div class="form-group">
        <label>Fecha final</label>
        <input type="date" wire:model.lazy="toDate" class="form-control">
        @error('toDate') <span class="text-danger">{{$message}}</span>@enderror
    </div>
</div>

<div class="col-sm-12 col-md-3 align-self-center d-flex justify-content-around">
    @if($userid > 0 && $fromDate != null && $toDate != null)
        <button wire:click.prevent="Consultar" type="button"
            class="btn btn-dark">Consultar</button>
        @endif

        @if($total > 0)
            <button wire:click.prevent="Print()" type="button"
                class="btn btn-dark">Imprimir</button>
        @endif
    </div>
```

Roles y Permisos

```
C:\laragon\www\ventaslite (master)
$ composer require spatie/laravel-permission
Using version ^4.1 for spatie/laravel-permission
./composer.json has been updated
Running composer update spatie/laravel-permission
Loading composer repositories with package information
Updating dependencies
Lock file operations: 1 install, 0 updates, 0 removals
- Locking spatie/laravel-permission (4.1.0)
Writing lock file
Installing dependencies from lock file (including require-dev)
Package operations: 1 install, 0 updates, 0 removals
- Installing spatie/laravel-permission (4.1.0): Extracting archive

    protected $fillable = [
        'name',
        'email',
        'password',
        'profile',
        'phone',
        'status',
        'image'
    ];

    /**
     * The attributes that should be hidden for arrays.
     *
     * @var array
     */
    protected $hidden = [
        'password',
        'remember_token',
    ];

    /**
     * The attributes that should be cast to native types.
     *
     * @var array
     */
    protected $casts = [
        'email_verified_at' => 'datetime',
    ];

```

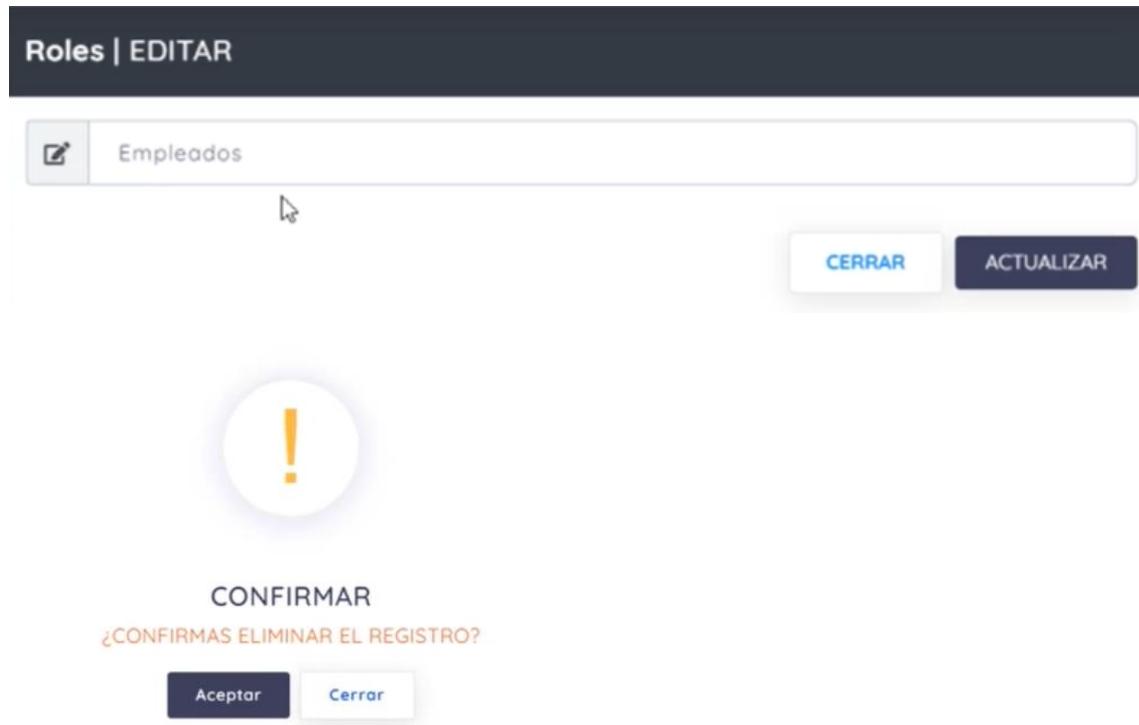
Roles Listado		
ID	DESCRIPCIÓN	ACTIONS
1	Admin	
2	Employee	

Podemos crear un nuevo rol pulsando el botón de añadir:

Roles | CREAR

	ej: Admin
	<input type="button" value="CERRAR"/> <input type="button" value="GUARDAR"/>

También podemos editar o eliminar el rol.



Reportes

Reports de Venta

FOLIO	TOTAL	ITEMS	ESTATUS	USUARIO	FECHA
6	\$5,000.00	1	Paid	Luis Fox	01-06-2021
7	\$350.00	1	Paid	Luis Fox	10-06-2021
8	\$1,200.00	1	Paid	Luis Fox	24-06-2021

Elige el usuario: Luis Fox

Elige el tipo de reporte: Ventas por Fecha

Fecha desde: 2021-06-01

Fecha hasta: 2021-06-30

Buttons at the bottom:

- Consultar
- Generar PDF
- Exportar a Excel

```

λ composer require barryvdh/laravel-dompdf
Using version ^0.9.0 for barryvdh/laravel-dompdf
./composer.json has been updated
Running composer update barryvdh/laravel-dompdf
Loading composer repositories with package information
Updating dependencies
Lock file operations: 5 installs, 0 updates, 0 removals
- Locking barryvdh/laravel-dompdf (v0.9.0)
- Locking dompdf/dompdf (v1.0.2)
- Locking phenx/php-font-lib (0.5.2)
- Locking phenx/php-svg-lib (v0.3.3)
- Locking sabberworm/php-css-parser (8.3.1)
Writing lock file
Installing dependencies from lock file (including require-dev)
Package operations: 5 installs, 0 updates, 0 removals
- Downloading sabberworm/php-css-parser (8.3.1)
- Downloading phenx/php-svg-lib (v0.3.3)
- Downloading phenx/php-font-lib (0.5.2)
- Downloading dompdf/dompdf (v1.0.2)
- Downloading barryvdh/laravel-dompdf (v0.9.0)
1/5 [=====>-----] 20%

```

```

</thead>
<tbody>
@foreach($data as $coin)
<tr>
    <td><h6>{{$coin->type}}</h6></td>
    <td><h6 class="text-center">${{number_format($coin->value,2)}}</h6></td>
    <td class="text-center">
        <span>
            
        </span>
    </td>

    <td class="text-center">
        <a href="javascript:void(0)" wire:click="Edit({{$coin->id}})" class="btn btn-dark mtmobile" title="Edit">
            <i class="fas fa-edit"></i>
        </a>

        <a href="javascript:void(0)" onclick="Confirm('{{{$coin->id}}})" class="btn btn-dark" title="Delete">
            <i class="fas fa-trash"></i>
        </a>
    </td>

```

Configurar Permisos

Permisos | Listado

ID	DESCRIPCIÓN	ACTIONS
2	Category_Create	
4	Category_Destroy	
1	Category_Index	
5	Category_Search	
3	Category_Update	

```

public function render()
{
    if(strlen($this->search) > 0)
        $data = User::where('name', 'like', '%' . $this->search . '%')
        ->select('*')->orderBy('name', 'asc')->paginate($this->pagination);
    else
        $data = User::select('*')->orderBy('name', 'asc')->paginate($this->pagination);

    return view('livewire.users.component',[

        'data' => $data,
        'roles' => Role::orderBy('name', 'asc')->get()
    ])
    ->extends('layouts.theme.app')
    ->section('content');
}

```

```

public function destroy(User $user)
{
    if($user) {
        $sales = Sale::where('user_id', $user->id)->count();
        if($sales > 0) {
            $this->emit('user-withsales','No es posible eliminar el usuario porque tiene ventas registradas');
        } else {
            $user->delete();
            $this->resetUI();
            $this->emit('user-deleted','Usuario Eliminado');
        }
    }
}

```

Usuarios | CREAR

Nombre

Teléfono

Email

Contraseña

Estatus

Asignar Role

Elegir

- Elegir
- Admin
- Cliente

Imagen de Perfil

No se eligió archivo

CERRAR
GUARDAR

Middlewares

```
protected $routeMiddleware = [
    'auth' => \App\Http\Middleware\Authenticate::class,
    'auth.basic' => \Illuminate\Auth\Middleware\AuthenticateWithBasicAuth::class,
    'cache.headers' => \Illuminate\Http\Middleware\SetCacheHeaders::class,
    'can' => \Illuminate\Auth\Middleware\Authorize::class,
    'guest' => \App\Http\Middleware\RedirectIfAuthenticated::class,
    'password.confirm' => \Illuminate\Auth\Middleware\RequirePassword::class,
    'signed' => \Illuminate\Routing\Middleware\ValidateSignature::class,
    'throttle' => \Illuminate\Routing\Middleware\ThrottleRequests::class,
    'verified' => \Illuminate\Auth\Middleware\EnsureEmailIsVerified::class,
];

```