

UNIVERSIDAD DE EL SALVADOR

FACULTAD MULTIDISCIPLINARIA DE OCCIDENTE

DEPARTAMENTO DE INGENIERÍA



TÉCNICAS DE PROGRAMACIÓN EN INTERNET (TPI)

REPORTE DE PRACTICA#3

DOCENTE: Ing. Xenia Ivette Peñate Godoy

<u>Alumnos</u>	<u>Carnet</u>
Br. Interiano Estrada, Elmer Eduardo	IE18001
Br. Martinez Figueroa, Guillermo Enrique	MF17006
Br. Oliva Maravilla, Christian Alexander	OM18003

PRACTICA NUMERO 3 UNIDAD 3 DE TPI

Para la siguiente práctica, se utilizará el framework de Laravel, un framework que está basado en PHP, para este framework se utilizara el Laravel en su versión 8.44.0 y un PHP en su versión más reciente (v8.0.5)



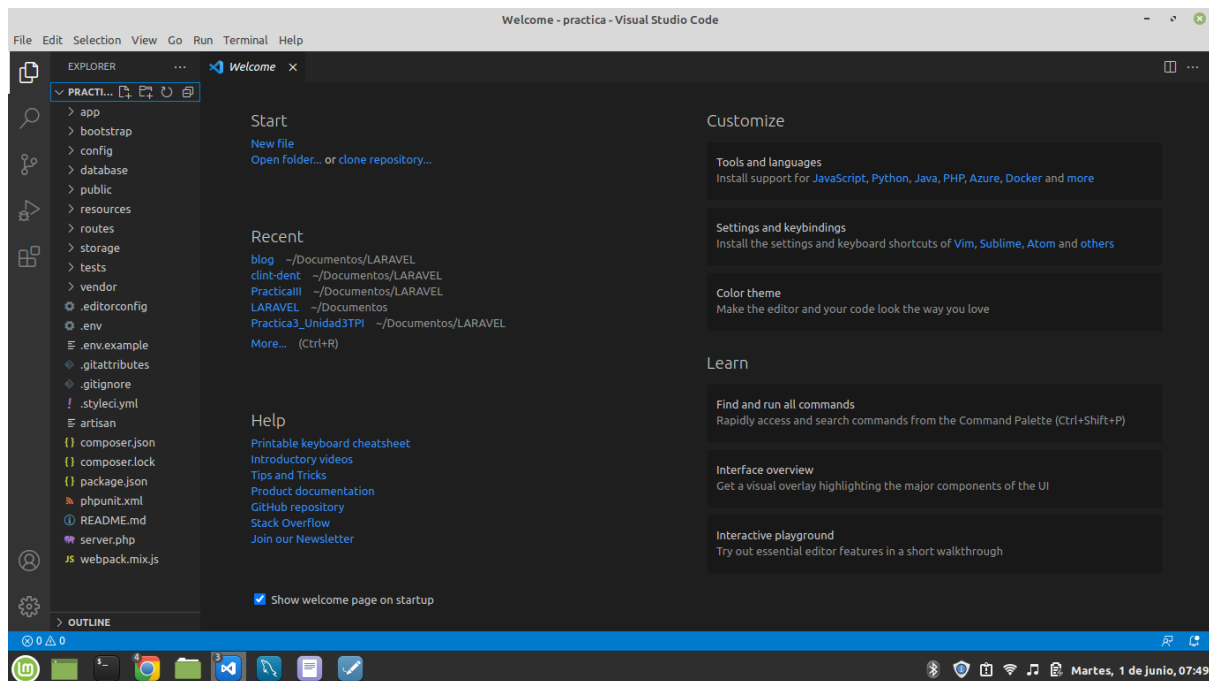
Para ello, vamos a comenzar un proyecto desde cero, en el que queremos instalar Laravel para comenzar a desarrollar.

Nos dirigimos a la carpeta de nuestro ordenador donde queremos instalar Laravel. Puede ser cualquier carpeta para guardar los proyectos. El comando para crear el proyecto depende de Composer, es cual es el siguiente:

`composer create-project --prefer-dist laravel/laravel nombre del proyecto (en nuestro caso, nuestro proyecto se llama practicaIII)`

En el siguiente comando tenemos "composer" que es el programa que se encarga de instalar Laravel en el nuevo proyecto. "create-project" es el subcomando de composer para crear un nuevo proyecto. "laravel/laravel" es el nombre del proyecto de base que vamos a usar para este nuevo proyecto creado en local, indica que es un proyecto llamado "laravel" que pertenece a la organización "laravel". Por último "mi-proyecto-laravel" es el nombre del proyecto que estamos creando.

Al ya tener creado el proyecto, se abre el proyecto desde un entorno de desarrollo (en nuestro caso usaremos Visual Studio Code), y tendríamos el siguiente aspecto:



Al ya tener abierto el proyecto creado procederemos en configurar el archivo “.env”, la cual es el archivo relacionado a las variables de entorno, y luego se configuran las opciones de las Databases, en este caso agregamos que nuestro usuario de la DB es “user” y la contraseña de la DB es “password”

```
.env
.env
1 APP_NAME=Laravel
2 APP_ENV=local
3 APP_KEY=base64:TcnCpZ9eDNWog051vbnj fLyfK1I+yGLfHwoZ3UqqEFg=
4 APP_DEBUG=true
5 APP_URL=http://localhost
6
7 LOG_CHANNEL=stack
8 LOG_LEVEL=debug
9
10 DB_CONNECTION=mysql
11 DB_HOST=127.0.0.1
12 DB_PORT=3306
13 DB_DATABASE=laravel
14 DB_USERNAME=user
15 DB_PASSWORD=password
16
17 BROADCAST_DRIVER=log
18 CACHE_DRIVER=file
19 QUEUE_CONNECTION=sync
20 SESSION_DRIVER=file
21 SESSION_LIFETIME=120
```

Al ya tener configurado esto, se selecciona la terminal del Visual Code, y se procede a el comando **"npm install"**, el cual es un código que instala las dependencias en la carpeta local node_modules, ante ello instala el contexto del paquete actual como un paquete global.

```

chris@chrisoliva:~/Documentos/LARAVEL/practica$ npm install
npm WARN deprecated chokidar@2.1.8: Chokidar 2 will break on node v14+. Upgrade to chokidar 3 with 15x less dependencies.
npm WARN deprecated fsevents@1.2.13: fsevents 1 will break on node v14+ and could be using insecure binaries. Upgrade to fsevents 2.
npm WARN deprecated uuid@3.4.0: Please upgrade to version 7 or higher. Older versions may use Math.random() in certain circumstances, which is known to be problematic. See https://v8.dev/blog/math-random for details.
npm WARN deprecated resolve-url@0.2.1: https://github.com/lydell/resolve-url#deprecated
npm WARN deprecated urix@0.1.0: Please see https://github.com/lydell/urix#deprecated
npm notice created a lockfile as package-lock.json. You should commit this file.
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.7 (node_modules/@types/browser-sync/node_modules/chokidar/node_modules/fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.13: wanted {"os":"darwin","arch":"any"} (current: {"os":"linux","arch":"x64"})
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@~2.3.1 (node_modules/chokidar/node_modules/fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@2.3.2: wanted {"os":"darwin","arch":"any"} (current: {"os":"linux","arch":"x64"})

added 1148 packages from 509 contributors and audited 1152 packages in 63.199s

95 packages are looking for funding
  run `npm fund` for details

found 33 moderate severity vulnerabilities
  run `npm audit fix` to fix them, or `npm audit` for details
chris@chrisoliva:~/Documentos/LARAVEL/practica$

```

Luego, se escribe el comando `"npm run dev"` el cual es un compilador de scripts

```
chris@chrisoliva:~/Documentos/LARAVEL/practica$ npm run dev  
  
> @ dev /home/chris/Documentos/LARAVEL/practica  
> npm run development  
  
> @ development /home/chris/Documentos/LARAVEL/practica  
> mix  
  
● Mix ██████████ emitting (95%)  
emit  
  
● Mix ██████████ done (99%) plugins  
WebpackBar:done
```

```
Laravel Mix v6.0.19

✓ Compiled Successfully in 1671ms



| File        | Size    |
|-------------|---------|
| /js/app.js  | 597 KiB |
| css/app.css | 1 bytes |



webpack compiled successfully
```

Al ya tener configurada la parte del NPM, se procede a ejecutar el comando **"composer install"**, esto para instalar composer dentro del proyecto. Y luego se ejecuta el comando **"composer update"**, para la actualización del composer dentro del proyecto.

```

chris@chrisoliva:~/Documentos/LARAVEL/practica$ composer install
Installing dependencies from lock file (including require-dev)
Verifying lock file contents can be installed on current platform.
Nothing to install, update or remove
Generating optimized autoload files
> Illuminate\Foundation\ComposerScripts::postAutoloadDump
> @php artisan package:discover --ansi
Discovered Package: facade/ignition
Discovered Package: fideloper/proxy
Discovered Package: fruitcake/laravel-cors
Discovered Package: laravel/sail
Discovered Package: laravel/tinker
Discovered Package: nesbot/carbon
Discovered Package: nunomaduro/collision
Package manifest generated successfully.
74 packages you are using are looking for funding.
Use the 'composer fund' command to find out more!
chris@chrisoliva:~/Documentos/LARAVEL/practica$ composer update
Composer is operating significantly slower than normal because you do not have the PHP curl extension enabled.
Loading composer repositories with package information
Updating dependencies
Nothing to modify in lock file
Installing dependencies from lock file (including require-dev)
Nothing to install, update or remove
Generating optimized autoload files
> Illuminate\Foundation\ComposerScripts::postAutoloadDump
> @php artisan package:discover --ansi
Discovered Package: facade/ignition
Discovered Package: fideloper/proxy
Discovered Package: fruitcake/laravel-cors
Discovered Package: laravel/sail
Discovered Package: laravel/tinker
Discovered Package: nesbot/carbon
Discovered Package: nunomaduro/collision
Package manifest generated successfully.
74 packages you are using are looking for funding.
Use the 'composer fund' command to find out more!

```

Al ya tener configurado lo del composer, vamos a configurar la base de datos del proyecto, para ello dentro de la terminal se escribirá el comando `“sudo mysql -u root”`, este comando es para ingresar al monitor del mysql y desde allí hacer las configuraciones de la base de datos.

```

chris@chrisoliva:~/Documentos/LARAVEL/practica$ sudo mysql -u root
[sudo] password for chris:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 23
Server version: 5.7.34-0ubuntu0.18.04.1 (Ubuntu)

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> 

```

Luego se procederá a crear la base de datos respectiva del proyecto, para ello dentro de la terminal del mysql se escribirá el comando `“create database nombre del proyecto;”` (en nuestro caso el nombre sería practica3), y al ya tener creada la base de datos se escribirá el comando `“show databases;”`, para efectuar que fue creado la base de datos.

```
mysql> create database practica3;
Query OK, 1 row affected (0.00 sec)

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| clint-dent |
| mysql |
| performance_schema |
| practica3 |
| practicaIII |
| sys |
+-----+
7 rows in set (0.00 sec)

mysql> 
```

Luego de tener creada la base de datos, vamos al archivo “.env” y configuramos el nombre de la base de datos con el nombre de la DB que creamos anteriormente.

```
.env
1 APP_NAME=Laravel
2 APP_ENV=local
3 APP_KEY=base64:TcnCpZ9eDNWog051vbnjfLyfK1I+yGLfHwoZ3UqqEFg=
4 APP_DEBUG=true
5 APP_URL=http://localhost
6
7 LOG_CHANNEL=stack
8 LOG_LEVEL=debug
9
10 DB_CONNECTION=mysql
11 DB_HOST=127.0.0.1
12 DB_PORT=3306
13 DB_DATABASE=practica3
14 DB_USERNAME=user
15 DB_PASSWORD=password
```

Después, se abrirá otra terminal para ejecutar el comando “php artisan migrate”, el cual hará las respectivas migraciones de información hacia la base de datos. Y luego se aplica el comando “php artisan migrate:fresh”, esto para volver a ejecutar las migraciones.

```
chris@chrisoliva:~/Documentos/LARAVEL/practica$ php artisan migrate
Migration table created successfully.
Migrating: 2014_10_12_000000_create_users_table (772.26ms)
Migrated: 2014_10_12_000000_create_users_table (772.26ms)
Migrating: 2014_10_12_100000_create_password_resets_table (636.43ms)
Migrated: 2014_10_12_100000_create_password_resets_table (636.43ms)
Migrating: 2019_08_19_000000_create_failed_jobs_table (793.22ms)
Migrated: 2019_08_19_000000_create_failed_jobs_table (793.22ms)
chris@chrisoliva:~/Documentos/LARAVEL/practica$ php artisan migrate:fresh
Dropped all tables successfully.
Migration table created successfully.
Migrating: 2014_10_12_000000_create_users_table (659.61ms)
Migrated: 2014_10_12_000000_create_users_table (659.61ms)
Migrating: 2014_10_12_100000_create_password_resets_table (1,017.89ms)
Migrated: 2014_10_12_100000_create_password_resets_table (1,017.89ms)
Migrating: 2019_08_19_000000_create_failed_jobs_table (896.99ms)
Migrated: 2019_08_19_000000_create_failed_jobs_table (896.99ms)
chris@chrisoliva:~/Documentos/LARAVEL/practica$ 
```

Al ya tener configurada la parte de las migraciones, se procederá a poner en marcha el proyecto Laravel con el servidor integrado en PHP, para ello se ejecuta el comando “php artisan serve”

```
chris@chrisoliva:~/Documentos/LARAVEL/practica$ php artisan serve
Starting Laravel development server: http://127.0.0.1:8000
[Tue Jun 1 09:23:37 2021] PHP 8.0.5 Development Server (http://127.0.0.1:8000) started
```

Al ver esto, se dirigirá a la dirección 127.0.0.1:8000, usando la combinación de ctrl+click, y luego nos aparecerá que ya está funcionando el proyecto en laravel en su versión 8.0.5



Documentation

Laravel has wonderful, thorough documentation covering every aspect of the framework. Whether you are new to the framework or have previous experience with Laravel, we recommend reading all of the documentation from beginning to end.



Laracasts

Laracasts offers thousands of video tutorials on Laravel, PHP, and JavaScript development. Check them out, see for yourself, and massively level up your development skills in the process.



Laravel News

Laravel News is a community driven portal and newsletter aggregating all of the latest and most important news in the Laravel ecosystem, including new package releases and tutorials.



Vibrant Ecosystem

Laravel's robust library of first-party tools and libraries, such as [Forge](#), [Vapor](#), [Nova](#), and [Envoyer](#) help you take your projects to the next level. Pair them with powerful open source libraries like [Cashier](#), [Dusk](#), [Echo](#), [Horizon](#), [Sanctum](#), [Telescope](#), and more.

[Shop](#) [Sponsor](#)

Laravel v8.44.0 (PHP v8.0.5)

CONSTRUCCIÓN DEL PROYECTO

Al ya tener ya ejecutado el proyecto de Laravel funcionando, vamos a hacer las respectivas configuraciones del entorno del proyecto. Para ello iremos al archivo “composer.json” y cambiaremos la parte de laravel/ui, la cual es la parte de utilidades y ajustes de la interfaz del usuario, y le modificaremos a una version mas reciente (3.3)

```
composer.json > {} require > laravel/ui
1  {
2      "name": "laravel/laravel",
3      "type": "project",
4      "description": "The Laravel Framework.",
5      "keywords": ["framework", "laravel"],
6      "license": "MIT",
7      "require": {
8          "php": "^7.3|^8.0",
9          "fideloper/proxy": "^4.4",
10         "fruitcake/laravel-cors": "^2.0",
11         "guzzlehttp/guzzle": "^7.0.1",
12         "laravel/framework": "^8.40",
13         "laravel/tinker": "^2.5",
14         "laravel/ui": "^3.3"
```

Luego, configuraremos algunas opciones predeterminadas del archivo “composer.lock”, la cual es un archivo de bloqueo del composer; dichas modificaciones serán actualizaciones acorde a las especificaciones del proyecto. Los campos que serán modificados son:

“content-hash”

```
composer.lock > ...
1  {
2      "_readme": [
3          "This file locks the dependencies of your project to a known state",
4          "Read more about it at https://getcomposer.org/doc/01-basic-usage.md#installing-dependencies",
5          "This file is @generated automatically"
6      ],
7      "content-hash": "445b2b56036f4fb3c253035674db75e4",
8      "packages": [
```

“version” del laravel/ui

```
{ } composer.lock > [ ] packages > { } 14
1094     },
1095     {
1096         "name": "laravel/ui",
1097         "version": "v3.3.0",
1098         "source": {
```

“reference” del source

```
{ } composer.lock > [ ] packages > { } 14 > { } source > [ ] reference
1099         "type": "git",
1100         "url": "https://github.com/laravel/ui.git",
1101         "reference": "07d725813350c695c779382cbd6dac0ab8665537",
1102     },
```

“url” y “reference” del dist

```
{ } composer.lock > [ ] packages > { } 14 > { } dist
1102     },
1103     "dist": {
1104         "type": "zip",
1105         "url": "https://api.github.com/repos/laravel/ui/zipball/07d725813350c695c779382cbd6dac0ab8665537",
1106         "reference": "07d725813350c695c779382cbd6dac0ab8665537",
1107         "shasum": ""
1108     }
1109 }
```

Las variables “illuminate” del require

```
{ } composer.lock > [ ] packages > { } 14 > { } require
1107     "shasum": ""
1108 },
1109 "require": {
1110     "illuminate/console": "^8.42",
1111     "illuminate/filesystem": "^8.42",
1112     "illuminate/support": "^8.42",
1113     "illuminate/validation": "^8.42",
1114     "php": "^7.3|^8.0"
```

Y el “source” y “time” de support

```
{ } composer.lock > [ ] packages > { } 14 > { } support > [ ] source
1140     },
1141     "support": {
1142         "source": "https://github.com/laravel/ui/tree/v3.3.0",
1143     },
1144     "time": "2021-05-25T16:45:33+00:00"
1145 }
```

Después, configuraremos algunas opciones de versiones del archivo “package.json”, mas que todo, las versiones del “laravel-mix” en su versión 6.0.19 y el “resolve-url-loader” en su versión 4.0.0

```
{
  "bootstrap": "4.0.0",
  "jquery": "^3.6",
  "laravel-mix": "6.0.19",
  "lodash": "^4.17.19",
  "popper.js": "^1.16.1",
  "postcss": "^8.1.14",
  "resolve-url-loader": "4.0.0",
  "sass": "^1.32.11",
  "sass-loader": "11.0.1"
}
```

Al ya tener configurado estos archivos, nos dirigiremos al directorio config/databases.php a configurar la gestión de las migraciones en la base de datos, para ello se modificarán las opciones de “charset” y “collation” en dentro del parámetro del utf8

```
> database.php
...
'unix_socket' => env('DB_SOCKET', ''),
'charset' => 'utf8',
'collation' => 'utf8_unicode_ci',
'prefix' => '',
```


Al acabar con esa cuestión, configuraremos las opciones del login y el welcome. Para configurar el login nos dirigimos al directorio `resources/views/auth/login.blade.php`; y en dicho archivo modificaremos algunas variables, como el “card-header” del div class “card”, algunos labels y nombre de comentarios predeterminados por unos conforme a las exigencias del proyecto; y tendríamos el script del login de la siguiente manera:

```
@extends('layouts.app')
@section('content')
<div class="container">
    <div class="row justify-content-center">
        <div class="col-md-8">
            <div class="card">
                <div class="card-header">{{ __('Inicio de Sesión') }}</div>

                <div class="card-body">
                    <form method="POST" action="{{ route('login') }}">
                        @csrf

                        <div class="form-group row">
                            <label for="email" class="col-md-4 col-form-label text-md-right">{{ __('Correo Electrónico') }}</label>

                            <div class="col-md-6">
                                <input id="email" type="email" class="form-control @error('email') is-invalid @enderror"
name="email" value="{{ old('email') }}" required autocomplete="email" autofocus>

                                @error('email')
                                    <span class="invalid-feedback" role="alert">
                                        <strong>{{ $message }}</strong>
                                    </span>
                                @enderror
                            </div>
                        </div>

                        <div class="form-group row">
                            <label for="password" class="col-md-4 col-form-label text-md-right">{{ __('Contraseña') }}</label>

                            <div class="col-md-6">
                                <input id="password" type="password" class="form-control @error('password') is-invalid @enderror"
name="password" required autocomplete="current-password">

                                @error('password')
                                    <span class="invalid-feedback" role="alert">
                                        <strong>{{ $message }}</strong>
                                    </span>
                                @enderror
                            </div>
                        </div>
                    </form>
                </div>
            </div>
        </div>
    </div>
</div>
```



```

<!DOCTYPE html>
<html lang="{ str_replace('-', '-', app()->getLocale()) }">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">

    <title>Práctica III – Técnicas de Programación de Internet</title>

    <!-- Fonts -->
    <link href="https://fonts.googleapis.com/css2?family=Nunito:wght@400;600;700&display=swap"
rel="stylesheet">

    <!-- Styles -->
    <style>
      /* normalize.css v8.0.1 | MIT License | github.com/necolas/normalize.css
*/html{line-height:1.15;-webkit-text-size-adjust:100%}body{margin:0}{background-color:transparent}[hid
den]{display:none}html{font-family:system-ui,-apple-system,BlinkMacSystemFont,Segoe
UI,Roboto,Helvetica Neue,Arial,Noto Sans,sans-serif,Apple Color Emoji,Segoe UI Emoji,Segoe UI
Symbol,Noto Color Emoji;line-height:1.5}*,:after,:before{box-sizing:border-box;border:0 solid
#e2e8f0}{color:inherit;text-decoration:inherit}svg,video{display:block;vertical-align:middle}video{max-widt
h:100%;height:auto}.bg-white{--bg-opacity:1;background-color:#fff;background-color:rgba(255,255,255,
var(--bg-opacity))}.bg-gray-100{--bg-opacity:1;background-color:#f7fafc;background-color:rgba(247,25
0,252,var(--bg-opacity))}.border-gray-200{--border-opacity:1;border-color:#edf2f7;border-color:rgba(2
37,242,247,var(--border-opacity))}.border-t{border-top-width:1px}.flex{display:flex}.grid{display:grid}.hid
den{display:none}.items-center{align-items:center}.justify-center{justify-content:center}.font-semibold{fon
t-weight:600}.h-5{height:1.25rem}.h-8{height:2rem}.h-16{height:4rem}.text-sm{font-size:.875rem}.text-
lg{font-size:1.125rem}.leading-7{line-height:1.75rem}.mx-auto{margin-left:auto;margin-right:auto}.ml-1{
margin-left:.25rem}.mt-2{margin-top:.5rem}.mr-2{margin-right:.5rem}.ml-2{margin-left:.5rem}.mt-4{mar
gin-top:1rem}.ml-4{margin-left:1rem}.mt-8{margin-top:2rem}.ml-12{margin-left:3rem}.-mt-p{margin-top:
-1px}.max-w-6xl{max-width:72rem}.min-h-screen{min-height:100vh}.overflow-hidden{overflow:hidden}.p-6{padding:1.5rem}.py-4{padding-top:1rem;padding-bottom:1rem}.px-6{padding-left:1.5rem;padding-r
ight:1.5rem}.pt-8{padding-top:2rem}.fixed{position:fixed}.relative{position:relative}.top-0{top:0}.right-0{rig
ht:0}.shadow{box-shadow:0 1px 3px 0 rgba(0,0,0,.1),0 1px 2px 0
rgba(0,0,0,.06)}.text-center{text-align:center}.text-gray-200{--text-opacity:1;color:#edf2f7;color:rgba(23
7,242,247,var(--text-opacity))}.text-gray-300{--text-opacity:1;color:#e2e8f0;color:rgba(226,232,240,va
r(--text-opacity))}.text-gray-400{--text-opacity:1;color:#cbd5e0;color:rgba(203,213,224,var(--text-opa
city))}.text-gray-500{--text-opacity:1;color:#a0aec0;color:rgba(160,174,192,var(--text-opacity))}.text-gr
ay-600{--text-opacity:1;color:#718096;color:rgba(113,128,150,var(--text-opacity))}.text-gray-700{--t
ext-opacity:1;color:#4a5568;color:rgba(74,85,104,var(--text-opacity))}.text-gray-900{--text-opacity:1;c
olor:#1a202c;color:rgba(26,32,44,var(--text-opacity))}.underline{text-decoration:underline}.antialiased{-w
ebkit-font-smoothing:antialiased;-moz-osx-font-smoothing:grayscale}.w-5{width:1.25rem}.w-8{width:2r
em}.w-auto{width:auto}.grid-cols-1{grid-template-columns:repeat(1,minmax(0,1fr))}@media
(min-width:640px){.sm\:rounded-lg{border-radius:.5rem}.sm\:block{display:block}.sm\:items-center{align-it
ems:center}.sm\:justify-start{justify-content:flex-start}.sm\:justify-between{justify-content:space-between
}.sm\:h-20{height:5rem}.sm\:ml-0{margin-left:0}.sm\:px-6{padding-left:1.5rem;padding-right:1.5rem}.sm\:
pt-0{padding-top:0}.sm\:text-left{text-align:left}.sm\:text-right{text-align:right}}@media
(min-width:768px){.md\:border-t-0{border-top-width:0}.md\:border-l{border-left-width:1px}.md\:grid-col

```

```

s-2{grid-template-columns:repeat(2,minmax(0,1fr))}}@media
(min-width:1024px){.lg\:px-8{padding-left:2rem;padding-right:2rem}}@media
(prefers-color-scheme:dark){.dark\:bg-gray-800{--bg-opacity:1;background-color:#2d3748;background-c
olor:rgba(45,55,72,var(--bg-opacity))}.dark\:bg-gray-900{--bg-opacity:1;background-color:#1a202c;bac
kground-color:rgba(26,32,44,var(--bg-opacity))}.dark\:border-gray-700{--border-opacity:1;border-color:
#4a5568;border-color:rgba(74,85,104,var(--border-opacity))}.dark\:text-white{--text-opacity:1;color:#ff
f;color:rgba(255,255,255,var(--text-opacity))}.dark\:text-gray-400{--text-opacity:1;color:#cbd5e0;color:r
gba(203,213,224,var(--text-opacity))}}
</style>

```

```

<style>
  body {
    font-family: 'Nunito', sans-serif;
    background-color: #f7fafc;
  }
  .text-right{
    display: flex;
    /* flex-direction: column; */
    justify-content: space-between;
  }
</style>
</head>
<body class="antialiased">
  <div class="relative flex items-top justify-center min-h-screen sm:items-center py-4 sm:pt-0">
    @if (Route::has('login'))
      <div class="hidden fixed top-0 right-0 px-6 py-4 sm:block">
        @auth
          <a href="{{ url('/home') }}" class="text-sm text-gray-700 underline"></a>
        @else
          <a href="{{ route('login') }}" class="text-sm text-black-700 underline">Iniciar Sesión</a>

          @if (Route::has('register'))
            <a href="{{ route('register') }}" class="ml-4 text-sm text-black-700
underline">Regístrate</a>
          @endif
        @endauth
      </div>
    @endif

    <div class="max-w-6xl mx-auto sm:px-6 lg:px-8">
      <div class="flex justify-center pt-8 sm:justify-between sm:pt-0">
        <svg viewBox="0 0 651 192" fill="none" xmlns="http://www.w3.org/2000/svg" class="h-16
w-auto text-gray-700 sm:h-20">
          <g clip-path="url(#clip0)" fill="#EF3B2D">
                                                    <path d="M248.032
44.676h-16.466v100.23h47.394v-14.748h-30.928V44.676zM337.091
87.202c-2.101-3.341-5.083-5.965-8.949-7.875-3.865-1.909-7.756-2.864-11.669-2.864-5.062

```

0-9.69.931-13.89 2.792-4.201 1.861-7.804 4.417-10.811 7.661-3.007 3.246-5.347 6.993-7.016
 11.239-1.672 4.249-2.506 8.713-2.506 13.389 0 4.774.834 9.26 2.506 13.459 1.669 4.202 4.009
 7.925 7.016 11.169 3.007 3.246 6.609 5.799 10.811 7.66 4.199 1.861 8.828 2.792 13.89 2.792 3.913
 0 7.804-.955 11.669-2.863 3.866-1.908 6.849-4.533
 8.949-7.875v9.021h15.607V78.182h-15.607v9.02zm-1.431 32.503c-.955 2.578-2.291 4.821-4.009
 6.73-1.719 1.91-3.795 3.437-6.229 4.582-2.435 1.146-5.133 1.718-8.091 1.718-2.96
 0-5.633-.572-8.019-1.718-2.387-1.146-4.438-2.672-6.156-4.582-1.719-1.909-3.032-4.152-3.9
 38-6.73-.909-2.577-1.36-5.298-1.36-8.161 0-2.864.451-5.585 1.36-8.162.905-2.577
 2.219-4.819 3.938-6.729 1.718-1.908 3.77-3.437 6.156-4.582 2.386-1.146 5.059-1.718
 8.019-1.718 2.958 0 5.656.572 8.091 1.718 2.434 1.146 4.51 2.674 6.229 4.582 1.718 1.91 3.054
 4.152 4.009 6.729.953 2.577 1.432 5.298 1.432 8.162-.001 2.863-.479 5.584-1.432
 8.161zM463.954
 87.202c-2.101-3.341-5.083-5.965-8.949-7.875-3.865-1.909-7.756-2.864-11.669-2.864-5.062
 0-9.69.931-13.89 2.792-4.201 1.861-7.804 4.417-10.811 7.661-3.007 3.246-5.347 6.993-7.016
 11.239-1.672 4.249-2.506 8.713-2.506 13.389 0 4.774.834 9.26 2.506 13.459 1.669 4.202 4.009
 7.925 7.016 11.169 3.007 3.246 6.609 5.799 10.811 7.66 4.199 1.861 8.828 2.792 13.89 2.792 3.913
 0 7.804-.955 11.669-2.863 3.866-1.908 6.849-4.533
 8.949-7.875v9.021h15.607V78.182h-15.607v9.02zm-1.432 32.503c-.955 2.578-2.291 4.821-4.009
 6.73-1.719 1.91-3.795 3.437-6.229 4.582-2.435 1.146-5.133 1.718-8.091 1.718-2.96
 0-5.633-.572-8.019-1.718-2.387-1.146-4.438-2.672-6.156-4.582-1.719-1.909-3.032-4.152-3.9
 38-6.73-.909-2.577-1.36-5.298-1.36-8.161 0-2.864.451-5.585 1.36-8.162.905-2.577
 2.219-4.819 3.938-6.729 1.718-1.908 3.77-3.437 6.156-4.582 2.386-1.146 5.059-1.718
 8.019-1.718 2.958 0 5.656.572 8.091 1.718 2.434 1.146 4.51 2.674 6.229 4.582 1.718 1.91 3.054
 4.152 4.009 6.729.953 2.577 1.432 5.298 1.432 8.162 0 2.863-.479 5.584-1.432 8.161zM650.772
 44.676h-15.606v100.23h15.606V44.676zM365.013
 144.906h15.607V93.538h26.776V78.182h-42.383v66.724zM542.133 78.182l-19.616
 51.096-19.616-51.096h-15.808l25.617 66.724h19.614l25.617-66.724h-15.808zM591.98
 76.466c-19.112 0-34.239 15.706-34.239 35.079 0 21.416 14.641 35.079 36.239 35.079 12.088 0
 19.806-4.622 29.234-14.688l-10.544-8.158c-.006.008-7.958 10.449-19.832 10.449-13.802
 0-19.612-11.127-19.612-16.884h51.777c2.72-22.043-11.772-40.877-33.023-40.877zm-18.713
 29.28c.12-1.284 1.917-16.884 18.589-16.884 16.671 0 18.697 15.598 18.813
 16.884h-37.402zM184.068
 43.892c-.024-.088-.073-.165-.104-.25-.058-.157-.108-.316-.191-.46-.056-.097-.137-.176-.203
 -.265-.087-.117-.161-.242-.265-.345-.085-.086-.194-.148-.29-.223-.109-.085-.206-.182-.327-
 .252l-.002-.001-.002-.002-35.648-20.524a2.971 2.971 0 00-2.964 0l-35.647
 20.522-.002.002-.002.001c-.121.07-.219.167-.327.252-.096.075-.205.138-.29.223-.103.103-.178
 .228-.265.345-.066.089-.147.169-.203.265-.083.144-.133.304-.191.46-.031.085-.08.162-.104.25
 -.067.249-.103.51-.103.776v38.979l-29.706 17.103V24.493a3 3 0
 00-.103-.776c-.024-.088-.073-.165-.104-.25-.058-.157-.108-.316-.191-.46-.056-.097-.137-.17
 6-.203-.265-.087-.117-.161-.242-.265-.345-.085-.086-.194-.148-.29-.223-.109-.085-.206-.182
 -.327-.252l-.002-.001-.002-.002L40.098 1.396a2.971 2.971 0 00-2.964 0L1.487
 21.919l-.002.002-.002.001c-.121.07-.219.167-.327.252-.096.075-.205.138-.29.223-.103.103-.17
 8.228-.265.345-.066.089-.147.169-.203.265-.083.144-.133.304-.191.46-.031.085-.08.162-.104.2
 5-.067.249-.103.51-.103.776v122.09c0 1.063.568 2.044 1.489 2.575l71.293
 41.045c.156.089.324.143.49.202.078.028.15.074.23.095a2.98 2.98 0 001.524
 0c.069-.018.132-.059.2-.083.176-.061.354-.119.519-.214l71.293-41.045a2.971 2.971 0
 001.489-2.575v-38.979l34.158-19.666a2.971 2.971 0 001.489-2.575V44.666a3.075 3.075 0

00-.106-.774zM74.255 143.167I-29.648-16.779 31.136-17.926.001-.001 34.164-19.669 29.674
17.084-21.772 12.428-43.555
24.863zm68.329-76.259v33.841I-12.475-7.182-17.231-9.92V49.806I12.475 7.182 17.231
9.92zm2.97-39.335I29.693 17.095-29.693 17.095-29.693-17.095 29.693-17.095zM54.06
114.089I-12.475 7.182V46.733I17.231-9.92 12.475-7.182v74.537I-17.231 9.921zM38.614
7.398I29.693 17.095-29.693 17.095L8.921 24.493 38.614 7.398zM5.938 29.632I12.475 7.182
17.231 9.92v79.676I.001.005-.001.006c0
.114.032.221.045.333.017.146.021.294.059.434I.002.007c.032.117.094.222.14.334.051.124.088.25
5.156.371a.036.036 0
00.004.009c.061.105.149.191.222.288.081.105.149.22.244.314I.008.01c.084.083.19.142.284.215.1
06.083.202.178.32.247I.013.005.011.008 34.139 19.321v34.175L5.939
144.867V29.632h-.001zm136.646 115.235I-65.352 37.625V148.31I48.399-27.628
16.953-9.677v33.862zm35.646-61.22I-29.706 17.102V66.908I17.231-9.92
12.475-7.182v33.841z"/>

</g>

</svg>

</div>

<div class="mt-8 md:border-t border-gray-200 dark:border-gray-700 overflow-hidden shadow
sm:rounded-lg">

<div class="">

<div class="p-6 ">

<h2>Alumnos:</h2>

<div>

<p>Interiano Estrada, Elmer Eduardo</p>

<p>IE18001</p>

<p>Martinez Figueroa, Guillermo Enrique </p>

<p>MF17006</p>

<p>Oliva Maravilla, Christian Alexander</p>

<p>OM18003</p>

</div>

</div>

</div>

<div class="border-t text-gray-900 text-center">

<p>Práctica Correspondiente a la Unidad 3 de Técnicas de Programación de Internet (TPI)</p>

</div>

</div>

<div class="flex justify-center mt-4 sm:items-center sm:justify-between">

```

<div class="ml-4 texto-p text-sm text-black-500 sm:text-right sm:ml-0">
    Hecho en Laravel v{{ Illuminate\Foundation\Application::VERSION }} (PHP v{{ PHP_VERSION
  }})
</div>
</div>
</div>
</div>
</body>
</html>

```

Al ya tener configurada la parte del welcome, procederemos a crear el apartado del registro dentro del directorio `resources/views/auth/register.blade.php`, y dentro de este archivo se va configurar la parte del registro de usuarios dentro del proyecto como una extensión de un `layout.app` y dentro del apartado de “section” que aparece en la parte del Frontend, y dicho script estará configurado para almacenar la información del usuario que se está registrando, el cual sería de la siguiente manera:

```
@extends('layouts.app')

@section('content')

<div class="container">

    <div class="row justify-content-center">

        <div class="col-md-8">

            <div class="card">

                <div class="card-header">{{ __('Registro de tu cuenta') }}</div>

                <div class="card-body">

                    <form method="POST" action="{{ route('register') }}">

                        @csrf

                        <div class="form-group row">

                            <label for="name" class="col-md-4 col-form-label text-md-right">{{ __('Tú nombre') }}</label>

                            <div class="col-md-6">

                                <input id="name" type="text" class="form-control @error('name') is-invalid @enderror"

                                name="name" value="{{ old('name') }}" required autocomplete="name" autofocus>

                                @error('name')

                                    <span class="invalid-feedback" role="alert">

                                        <strong>{{ $message }}</strong>

                                    </span>

                                @enderror

                            </div>

                        </div>

                    </div>

                </div>

            <div class="form-group row">
```

```

    <label for="email" class="col-md-4 col-form-label text-md-right">{{ __('Tú correo')
}}</label>

    <div class="col-md-6">
        <input id="email" type="email" class="form-control @error('email') is-invalid @enderror"
name="email" value="{{ old('email') }}" required autocomplete="email">

        @error('email')
            <span class="invalid-feedback" role="alert">
                <strong>{{ $message }}</strong>
            </span>
        @enderror
    </div>
</div>

<div class="form-group row">
    <label for="password" class="col-md-4 col-form-label text-md-right">{{ __('Contraseña')
}}</label>

    <div class="col-md-6">
        <input id="password" type="password" class="form-control @error('password') is-invalid
@enderror" name="password" required autocomplete="new-password">

        @error('password')
            <span class="invalid-feedback" role="alert">
                <strong>{{ $message }}</strong>
            </span>
        @enderror
    </div>
</div>

<div class="form-group row">
    <label for="password-confirm" class="col-md-4 col-form-label text-md-right">{{
__('Confirmar Contraseña') }}</label>

    <div class="col-md-6">
        <input id="password-confirm" type="password" class="form-control"
name="password_confirmation" required autocomplete="new-password">
    </div>
</div>

<div class="form-group row mb-0">
    <div class="col-md-6 offset-md-4">
        <button type="submit" class="btn btn-primary">
            {{ __('Regístrate') }}
        </button>
    </div>
</div>

```



```

        </div>
    </form>
</div>
</div>
</div>
</div>
</div>
</div>
@endsection

```

Al acabar de configurar la opción del register, vamos a configurar las opciones del layout de nuestro proyecto, para ello crearemos las opciones de diseño de la app en el directorio `resources/views/layouts/app.blade.php`, en dicho archivo estarán los parámetros de diseño que se mostrarán en el entorno de la app, y su script sería el siguiente:

```

<!doctype html>
<html lang="{{ str_replace('_', '-', app()->getLocale()) }}">
<head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">

    <!-- CSRF Token -->
    <meta name="csrf-token" content="{{ csrf_token() }}">

    <title>{{ config('app.name', 'Laravel') }}</title>

    <!-- Scripts -->
    <script src="{{ asset('js/app.js') }}" defer></script>

    <!-- Fonts -->
    <link rel="dns-prefetch" href="//fonts.gstatic.com">
    <link href="https://fonts.googleapis.com/css?family=Nunito" rel="stylesheet">

    <!-- Styles -->
    <link href="{{ asset('css/app.css') }}" rel="stylesheet">
</head>
<body>
    <div id="app">
        <nav class="navbar navbar-expand-md navbar-light bg-white shadow-sm">
            <div class="container">
                <a class="navbar-brand" href="{{ url('#') }}">
                    {{ config('app.name', 'Laravel') }}
                </a>
                <button class="navbar-toggler" type="button" data-toggle="collapse"
data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false"
aria-label="{{ __('Toggle navigation') }}">
                    <span class="navbar-toggler-icon"></span>
                </button>
            </div>
        </nav>
    </div>
</body>
</html>

```

```

<div class="collapse navbar-collapse" id="navbarSupportedContent">
  <!-- Left Side Of Navbar -->
  <ul class="navbar-nav mr-auto">

    </ul>

    <!-- Right Side Of Navbar -->
    <ul class="navbar-nav ml-auto">
      <!-- Authentication Links -->
      @guest
        @if (Route::has('login'))
          <li class="nav-item">
            <a class="nav-link" href="{{ route('login') }}">{{ __('Inicio de Sesión') }}</a>
          </li>
        @endif

        @if (Route::has('register'))
          <li class="nav-item">
            <a class="nav-link" href="{{ route('register') }}">{{ __('Regístrate') }}</a>
          </li>
        @endif
      @else
        <li class="nav-item dropdown">
          <a id="navbarDropdown" class="nav-link dropdown-toggle" href="#" role="button"
data-toggle="dropdown" aria-haspopup="true" aria-expanded="false" v-pre>
            {{ Auth::user()->name }}
          </a>

          <div class="dropdown-menu dropdown-menu-right" aria-labelledby="navbarDropdown">
            <a class="dropdown-item" href="{{ route('logout') }}"
              onclick="event.preventDefault();
                document.getElementById('logout-form').submit();">
              {{ __('Cerrar Sesión') }}
            </a>

            <form id="logout-form" action="{{ route('logout') }}" method="POST" class="d-none">
              @csrf
            </form>
          </div>
        </li>
      @endguest
    </ul>
  </div>
</div>
</nav>

```

```

        <main class="py-4">
            @yield('content')
        </main>
    </div>
</body>
</html>

```

Al culminar la configuración del diseño de la app, vamos a crear las migraciones y modelos que tendrá nuestro proyecto, pero antes de iniciar con esto procederemos a ir al directorio database/.gitignore, esto para evitar inconvenientes que podrían surgir durante el funcionamiento del programa.

Luego, se comenzará a crear el modelo relacionado a las ventas, para ello nos dirigiremos a app/models/ventas.php, en este archivo será el que va interactuar con su respectiva tabla que estará dentro de la base de datos del proyecto, y su script sería de la siguiente manera:

```

<?php

namespace App\Models;

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

class Venta extends Model
{
    use HasFactory;
}

```

Con respecto a las migraciones, crearemos un archivo que se encargará de las migraciones del entorno de las ventas, y se creará en database/migrations/2021_05_30_052509_create_ventas_table.php, en este archivo se almacenará todo lo que pase con el entorno de las ventas, con ello haciendo la respectiva conexión con la base de datos, el script de este archivo sería el siguiente:

```

<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

class CreateVentasTable extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
}

```

```

public function up()
{
    Schema::create('ventas', function (Blueprint $table) {
        $table->id();
        $table->string('nombre_venta');
        $table->float('precio');

        $table->timestamps();

    });
}

/**
 * Reverse the migrations.
 *
 * @return void
 */
public function down()
{
    Schema::dropIfExists('ventas');
}
}

```

Al ya tener configurado lo relacionado con los modelos y las migraciones del proyecto, se creará el controlador del entorno de las ventas, para ello crearemos el archivo del controller en el directorio `app/Http/Controllers/Venta_Controller.php`, este archivo se encargará de la lógica de manejo de solicitudes relacionadas en el entorno de las ventas, y su script sería el siguiente:

```

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

class VentaController extends Controller
{

}

```

Al culminar con el controller, procederemos a configurar todo el entorno del home del proyecto, este archivo se creará en el directorio `resources/views/home.blade.php`, en este archivo estará toda la parte del proceso que se obtenga del sistema de ventas del programa, y por ser un motor de plantillas que viene incluido en Laravel solo configuraremos y modificaremos el script predeterminado conforme a la estructura y diseño del proyecto del sistema de ventas.

En dicho archivo estarán las configuraciones referentes al entorno del Frontend como la introducción de scripts del lenguaje Javascript, las configuraciones de la selección de opciones mediante select, la configuración del botón que estará configurado para hacer efectiva la compra de las opciones que se tienen en venta, y las configuraciones de los precios dentro de los parámetros establecidos anteriormente, ante ello, el script sería el siguiente:

```
@extends('layouts.app')
```

```
@section('content')
```

```
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.1/dist/css/bootstrap.min.css" rel="stylesheet"
integrity="sha384-+On0xVW2eSR5OomGNYDnhzAbDsOXxcvSN1TPprVMTNDbiYZCxYbOOI7+AMvyTG2
x" crossorigin="anonymous">
```

```
<div class="container">
```

```
  <div class="row justify-content-center">
```

```
    <div class="col-md-8">
```

```
      <div class="card">
```

```
        <div class="card-header">{{ __('Sistema de ventas:')} </div>
```

```
      <div class="card-body">
```

```
        @if (session('status'))
```

```
          <div class="alert alert-success" role="alert">
```

```
            {{ session('status') }}
```

```
          </div>
```

```
        @endif
```

```
        {{ __('Llena los siguientes campos a continuación:')} >>
```

```
        <form method="POST" action="{{ route('venta.request') }}">
```

```
          @csrf
```

```
          <div class="form-group row">
```

```
            <label for="name" class="col-md-4 col-form-label text-md-right">{{ __('Selecciona una
opcion: ') }}</label>
```

```
          <div class="col-md-6">
```

```
            <select id="nombre_venta" type="select" onchange="ShowSelected();" class="form-control
```

```
@error('nombre_venta') is-invalid @enderror" name="nombre_venta" value="{{ old('nombre_venta') }}"
```

```
required autocomplete="nombre_venta" autofocus>
```

```
            <option disabled="" selected="">Elija un producto</option>
```

```
            <option >Camisa</option>
```

```
            <option >Short</option>
```

```
            <option >Pantalon</option>
```

```
            <option >Calcetines</option>
```

```
          </select>
```

```
          @error('nombre_venta')
```

```
            <span class="invalid-feedback" role="alert">
```

```
              <strong>{{ $message }}</strong>
```

```
            </span>
```

```
          @enderror
```

```
</div>
```

```

</div>

<div class="form-group row">
  <label for="precio" class="col-md-4 col-form-label text-md-right">{{ _ ("Precio: $")
}}</label>

  <div class="col-md-6">
    <input id="precio" class="precio form-control @error('precio') is-invalid @enderror"
name="precio" value="{{ old('precio') }}" required autocomplete="precio" autofocus>

  </input>
  @error('precio')
  <span class="invalid-feedback" role="alert">
    <strong>{{ $message }}</strong>
  </span>
  @enderror
</div>
</div>
<div class="text-right">
  <button type="submit" class="btn btn-success ">Comprar</button>
</div>
</form>

</div>
</div>
</div>
</div>
</div>
@endsection

```

Al ya tener configurado el entorno del home del proyecto, se hará la creación del entorno del motor de plantilla para la parte de las compras, para ello se creará el archivo en el directorio `resources/views/compra.blade.php`; y en dicho archivo estarán los parámetros relacionados con la configuración del proceso de la compra que se dé dentro del proyecto, y su script sería el siguiente:

```

<!DOCTYPE html>
<html lang="en">

<head>
  <!-- CSS only -->
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.1/dist/css/bootstrap.min.css" rel="stylesheet"
integrity="sha384-+0n0xVW2eSR5OomGNYDnhzAbDsOXxcvSN1TPprVMTNDbiYZCxBbOOI7+AMvyTG2
x" crossorigin="anonymous">

  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">

```

```

<title>Compra</title>
</head>

<body>
<div class="card text-md-center">
  <div class="card-body col-auto">
    <h3 class="text-success text-center">
      @if(session('info'))
      <div class="alert alert-success">
        {{session('info')}}
      </div>
      @endif
    </h3>
    <h3 class="text-center text-danger">Historial de compras:</h3>
    <table class="table table-hover table-sm">
      <thead>
        <th>Artículo:</th>
        <th>Precio(USD):</th>
      </thead>
      <tbody>
        @foreach($ventas as $venta)
        <tr>
          <td>{{ $venta->nombre_venta}}</td>
          <td>{{ $venta->precio}}</td>
        </tr>
        @endforeach
      </tbody>
    </table>

    <div class="text-center">
      <a href="/home"><button type="submit" class="btn btn-warning">Regresar a comprar</button></a>
    </div>
  </div>
</div>
</body>

</html>

```

Y por último, se aplicará en la creación de la ruta web del proyecto, para ello se creará dicho archivo en routes/web.php, allí estara toda la parte del enrutamiento correspondiente para la interfaz web del proyecto, y en dicho archivo estarán los parámetros relacionados con todo el entorno de la venta y compra que han sido configurados anteriormente, ante ello, su script sería el siguiente:

```
<?php
```

```

use Illuminate\Support\Facades\Route;
use Illuminate\Http\Request;

```

```

use App\Models\Venta;
/*
|-----
| Web Routes
|-----
|
| Here is where you can register web routes for your application. These
| routes are loaded by the RouteServiceProvider within a group which
| contains the "web" middleware group. Now create something great!
|
*/

```

```

Route::get('/', function () {
    return view('welcome');
});

```

```

Auth::routes();

```

```

Route::get('/home', [App\Http\Controllers\HomeController::class, 'index'])->name('home');

```

```

Auth::routes();

```

```

Route::get('/home', [App\Http\Controllers\HomeController::class, 'index'])->name('home');

```

```

Auth::routes();

```

```

Route::get('/home', [App\Http\Controllers\HomeController::class, 'index'])->name('home');

```

```

Route::get('compra', function(){
    $ventas = Venta::orderBy('created_at','desc')->get();
    return view('compra', compact('ventas'));
})->name('compra');

```

```

Route::post('venta', function(Request $request){
    //return $request->all();
    $nueva_venta = new Venta;
    $nueva_venta -> nombre_venta = $request->input('nombre_venta');
    $nueva_venta -> precio = $request->input('precio');
    $nueva_venta -> save();
    return redirect()-> route('compra')->with('info','¡¡Compra hecha con éxito!!');

}) -> name('venta.request');

```


PROCESO DE COMPROBACIÓN

Al ya tener configurado todo el entorno interno del proyecto, se procederá a la comprobación de que el proyecto compile y se ejecute como se debe. Para ello, se abre una terminal dentro del Visual Studio Code, y se escribe el comando “php artisan serve” para arrancar el servidor del proyecto.

```
chris@chrisoliva:~/Documentos/LARAVEL/PracticaIII$ php artisan serve
Starting Laravel development server: http://127.0.0.1:8000
[Tue Jun 11 21:07:25 2021] Failed to listen on 127.0.0.1:8000 (reason: Address already in use)
```

Después que se haya ejecutado ese comando, se tiene que dirigir a la dirección que aparece allí usando la combinación ctrl+click.



Como se puede ver, aparece el entorno del Frontend que fue configurado anteriormente, y para comprobar que funciona el sistema de ventas del proyecto nos dirigiremos a la opción de “Regístrate”

The screenshot shows a registration form titled 'Registro de tú cuenta'. It has four input fields: 'Tú nombre' with the value 'Jenniffer Morales', 'Tú correo' with the value 'CM18045@ues.edu.sv', 'Contraseña' with masked characters, and 'Confirmar Contraseña' also with masked characters. A blue 'Regístrate' button is located at the bottom of the form.

Dentro del campo del registro se procede a ingresar los datos que se solicitan para quedar registrado dentro del sistema, y luego seleccionar la opción de “Registrate”
Luego, aparecerá la siguiente ventana con la configuración de compra.

Laravel

Jennifer Morales ▾

Sistema de ventas:

Llena los siguientes campos a continuacion:

Selecciona una opcion:

Precio: \$

Comprar

Y para comprobar que la configuración de la compra funciona correctamente, seleccionamos un producto, y le configuramos el precio de dicho producto, y luego se selecciona la opción de “Comprar”

Laravel

Jennifer Morales ▾

Sistema de ventas:

Llena los siguientes campos a continuacion:

Selecciona una opcion:

Precio: \$

Comprar

Despues nos aparecera otra ventana que nos confirmara que la compra fue realizada con éxito, mostrando el articulo comprado y su precio; y si seleccionamos la opcion de Regresar a comprar nos devolverá al lugar donde estuvimos antes

¡¡Compra hecha con éxito!!

Historial de compras:

Articulo:	Precio(USD):
Camisa	4

Regresar a comprar

Y para salir del sistema se selecciona la opción del usuario, y alli aparecera la opcion de Cerrar Sesión, y al seleccionarlo nos dirigirá al inicio del sistema

Laravel

Jennifer Morales ▾

Sistema de ventas:

Llena los siguientes campos a continuacion:

Selecciona una opcion:

Precio: \$

Comprar

Cerrar Sesion