



Análisis y desarrollo de software.

ID: 2556456



www.sena.edu.co

@SENAComunica

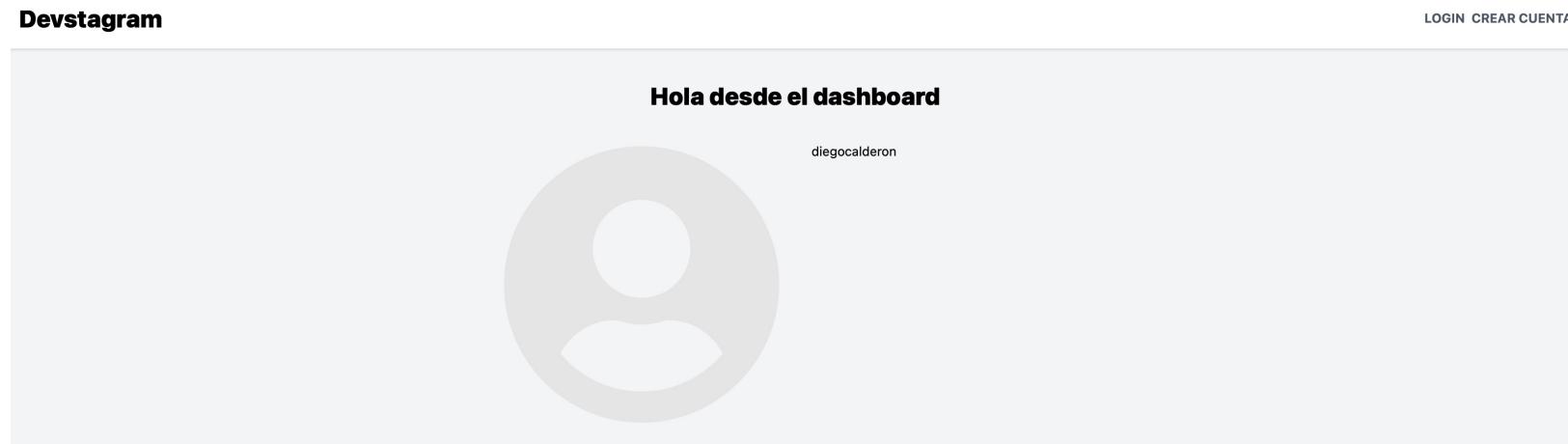
Instructor

Diego Fernando Calderón Silva
Correo: dfcalderon@sena.edu.co

Autenticación



Observando más de cerca nuestra aplicación podemos notar que si bien, hemos iniciado sesión y creado un login, al estar en el muro seguimos viendo la opción de iniciar sesión y registrarse y no la de cerrar sesión:



Por lo tanto, vamos a ver como corregir esto.

Autenticación



Nos dirigimos a app.blade.php que es nuestro archivo principal y un poco antes de la navegación podemos ayudarnos de los helper para saber si el usuario inicio sesión, por esto emepezamos poniendo un if.

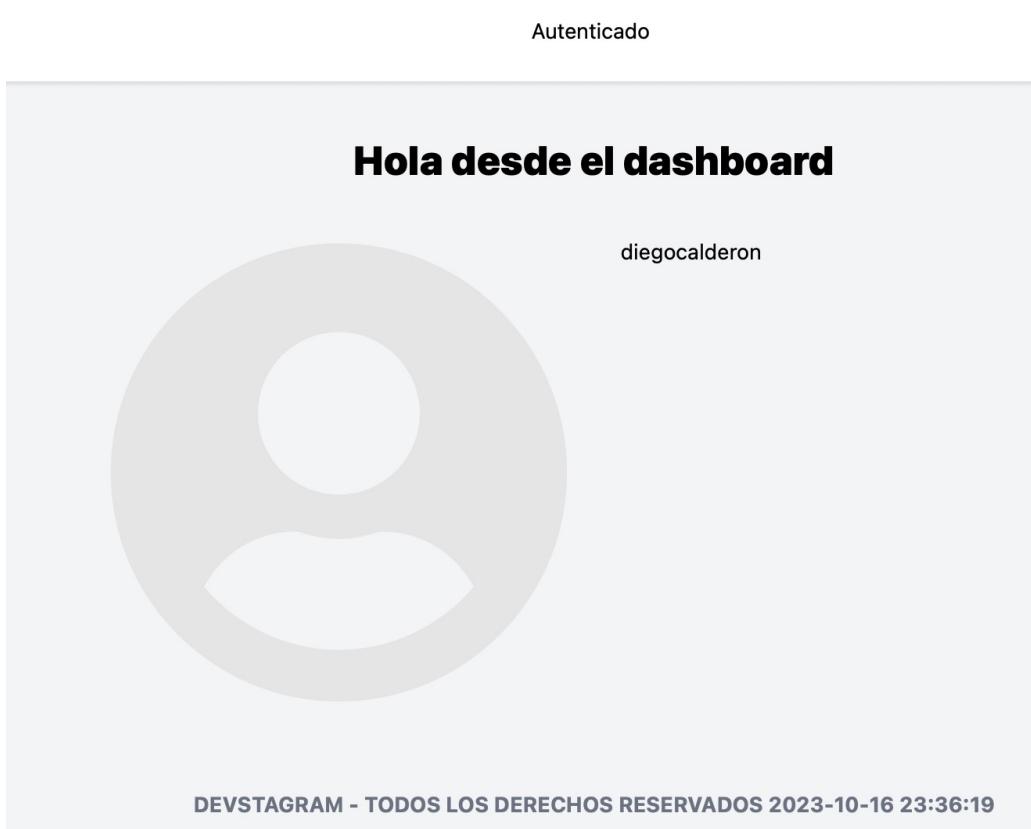
```
<header class="p-5 border-b bg-white shadow flex justify-between items-center">
    <h1 class="text-3xl font-black">
        Devstagram
    </h1>
    {{--Vamos a verificar si un usuario esta autenticado--}}
    @if(auth()->user())
        <p>Autenticado</p>
    @else
        <p>No autenticado</p>
    @endif
```

Autenticación



Y se evidencia como cuando iniciamos sesión esta autenticado y todo lo contrario de no haberlo hecho.

Autenticado

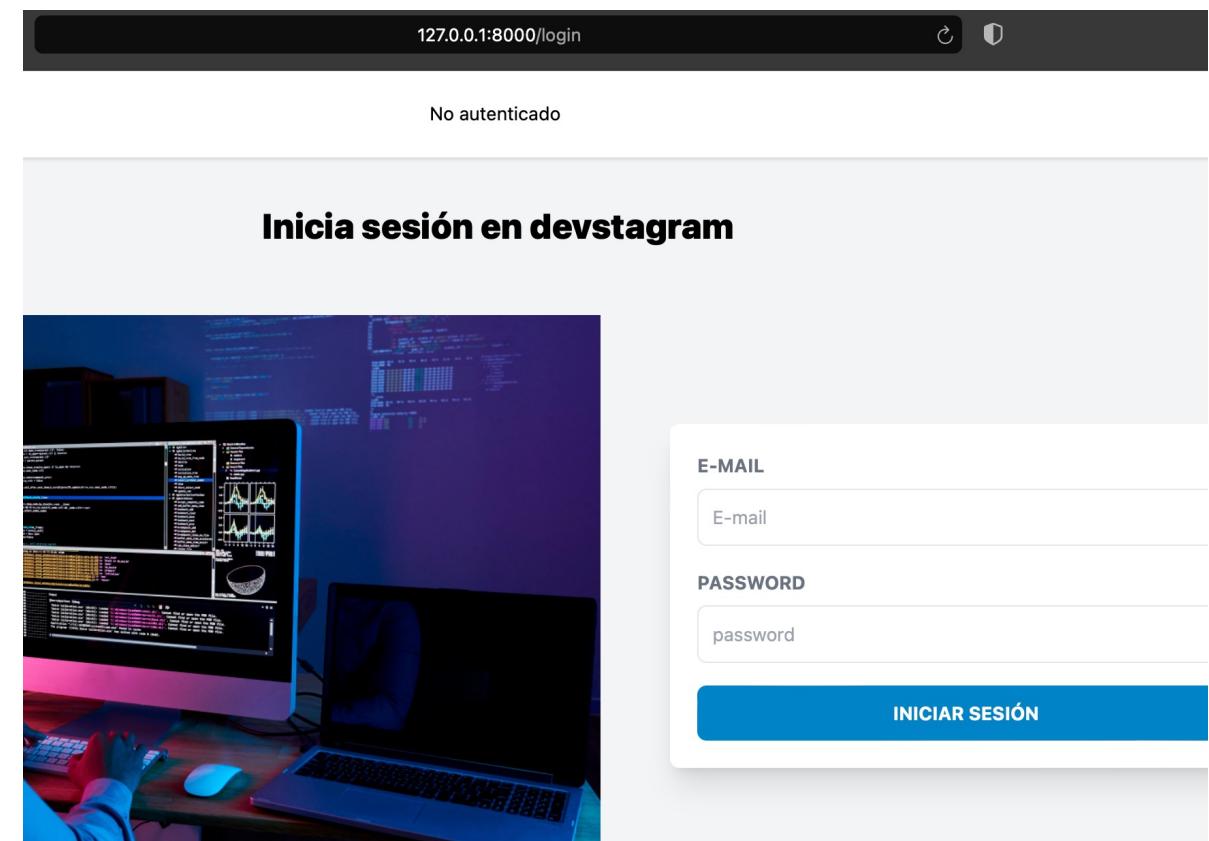


A screenshot of a web application dashboard. At the top, it says "Hola desde el dashboard". Below that is a large, light-gray circular placeholder for a user profile picture. To the right of the placeholder, the text "diegocalderon" is displayed. The background of the page is white.

DEVSTAGRAM - TODOS LOS DERECHOS RESERVADOS 2023-10-16 23:36:19

127.0.0.1:8000/login

No autenticado



A screenshot of a login form. At the top, it says "Inicia sesión en devstagram". Below that is a background image of a computer setup with multiple monitors displaying code and data. To the right of the background, there is a form with fields for "E-MAIL" and "PASSWORD", each containing placeholder text ("E-mail" and "password"). At the bottom is a large blue button labeled "INICIAR SESIÓN".

Autenticación



La forma anterior es como lo hacíamos hace algún tiempo, pero laravel nos ofrece una ayuda o helper llamada @auth la cual nos ahorra tiempo y se usa así:

```
</h1>
{{--Vamos a verificar si un usuario esta autenticado--}}
@auth()
    <p>Autenticado</p>
@endauth
```

Y para los usuarios no identificados podríamos usar @guest:

Recargue ambas páginas y notara que no hay cambios.

Ahora lo que debemos hacer es que cuando el usuario no este autenticado, me permita ver el login y crear cuenta por lo tanto...

Autenticación

Dándole forma al código para que me muestre cerrar sesión cuando este logueado tenemos:

```
<header class="p-5 border-b bg-white shadow flex justify-between items-center">
  <h1 class="text-3xl font-black">
    Devstagram
  </h1>
  {{--Vamos a verificar si un usuario esta autenticado--}}
  @auth()
    <nav class="flex gap-2 items-center">
      <a class="font-bold uppercase text-gray-600" href="{{ route('login') }}>Cerrar sesión</a>
    </nav>
  @endauth

  @guest()
    <nav class="flex gap-2 items-center">
      <a class="font-bold uppercase text-gray-600" href="{{ route('login') }}>Login</a>
      <a class="font-bold uppercase text-gray-600" href="{{route('register')}}>
        Crear cuenta
      </a>
    </nav>
  @endguest
</header>
```

Autenticación

También quiero que se mire el nombre del usuario, para esto usare auth() así:

```
<header class="p-5 border-b bg-white shadow flex justify-between items-center">
  <h1 class="text-3xl font-black">
    Devstagram
  </h1>
  {{--Vamos a verificar si un usuario esta autenticado--}}
  @auth()
    <nav class="flex gap-2 items-center">
      <a class="font-bold uppercase text-gray-600" href="#">
        Hola <span class="font-bold" > {{ auth() -> user() -> username }} </span>
      </a>
      <a class="font-bold uppercase text-gray-600" href="#">
        Cerrar sesión
      </a>
    </nav>
  @endauth

  @guest()
    <nav class="flex gap-2 items-center">
      <a class="font-bold uppercase text-gray-600" href="{{ route('login') }}">Login</a>
      <a class="font-bold uppercase text-gray-600" href="{{ route('register') }}">
        Crear cuenta
      </a>
    </nav>
  @endguest

</header>
```

Cerrar sesión



Ahora enfoquemos todo en destruir la sesión que iniciamos así:

Crearemos un controlador llamado: logoutController

Ahora nos dirigimos a web.php para nombrar la ruta de tipo get como: logout donde el método será 'store' y el name->(logout)

Ahora en el controller en un método store podremos un dd para comprobar el endpoint:

Y finalmente en app.Blade.php apuntaremos el cerrar sesión hacia la ruta:

```
 {{ route('logout') }}
```

Recargamos y presionamos el botón cerrar sesión y observamos que pasa.

Cerrar sesión

Con esto que acabamos de hacer no cerramos sesión, solo verificamos que el endpoint esté funcionando. Para cerrar usaremos el helper logout de auth así:

```
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

2 usages

class LogoutController extends Controller
{
    public function store(){
        auth()->logout();
        return redirect()->route( route: 'login' );
    }
}
```

Cerrar sesión

Si bien la forma anterior destruye nuestra sesión, no es la más segura del mundo ya que dejamos vulnerabilidades y podríamos sufrir un ataque cruzado como nos pasó en clases anteriores.

La primera medida que no es eficiente es que estamos usando el método get para tener un request de la base de datos lo cual ya es bastante malo, por lo tanto, lo cambiaremos a post

```
Route::get('uri: '/logout', [LogoutController::class, 'store'])->name(name: 'logout');
```

Así:

```
Route::post('uri: '/logout', [LogoutController::class, 'store'])->name(name: 'logout');
```

Ahora que lo cambiamos, tenemos acceso a la directiva @csrf; pero antes debemos hacer unos cambios en la barra de navegación

Cerrar sesión

Crearemos un formulario y usaremos el método post en la opción de cerrar sesión, cambiando también el a por un botón:

```
 {{--Vamos a verificar si un usuario esta autenticado--}}
@auth()
    <nav class="flex gap-2 items-center">
        <a class="font-bold uppercase text-gray-600" href="#">
            Hola <span class="font-bold" > {{ auth() -> user() -> username }} </span>
        </a>

        <form action="{{ route('logout') }}" method="post">
            @csrf
            <button type="submit" class="font-bold uppercase text-gray-600" >
                Cerrar sesión
            </button>
        </form>

    </nav>
@endauth
```

Mantener sesión abierta



Para lograr esto, iremos a login.blade.php y debajo de la validación de error pondremos un div demás así:

```
<div class="mb-5">
    <input type="checkbox" name="remember"> <label class="text-gray-500 text-sm" for="">Mantener mi sesión abierta</label>
</div>
```

Ahora en LoginController en donde está el store pondremos un dd para validar así:

```
dd($request->remember);
```

¿Qué pasa si dejo el checkbox vacío y le doy a iniciar sesión?

¿Qué pasa si selecciono el checkbox y le doy a iniciar sesión?

```
public function store(Request $request){
    dd($request->remember);

    $this->validate($request, [
        'email' => 'required|email',
        'password' => 'required'
    ]);
}
```

Mantener sesión abierta



Como pudimos observar, se obtiene un 'on' cuando esta seleccionado el campo, esto nos indica que nuestro endpoint sigue funcionando correctamente. Ahora, utilicemos este request a favor dentro del attempt ya que este como segundo parámetro tiene un request por default en false:

```
if (!auth()->attempt($request->only( keys: 'email', 'password'))){
    return back()->with(StatefulGuard)
}
return redirect()->with(StatefulGuard)
```

Attempt to authenticate a user using the given credentials.

Parameters: `array $credentials`
`bool $remember`

Returns: `bool`

Source: `vendor/laravel/framework/src/Illuminate/Contracts/Auth/StatefulGuard.php` :

Mantener sesión abierta



Siguiendo lo anterior tenemos el método store del LoginController así:

```
public function store(Request $request){
    $this->validate($request, [
        'email' =>'required|email',
        'password' =>'required'
    ]);

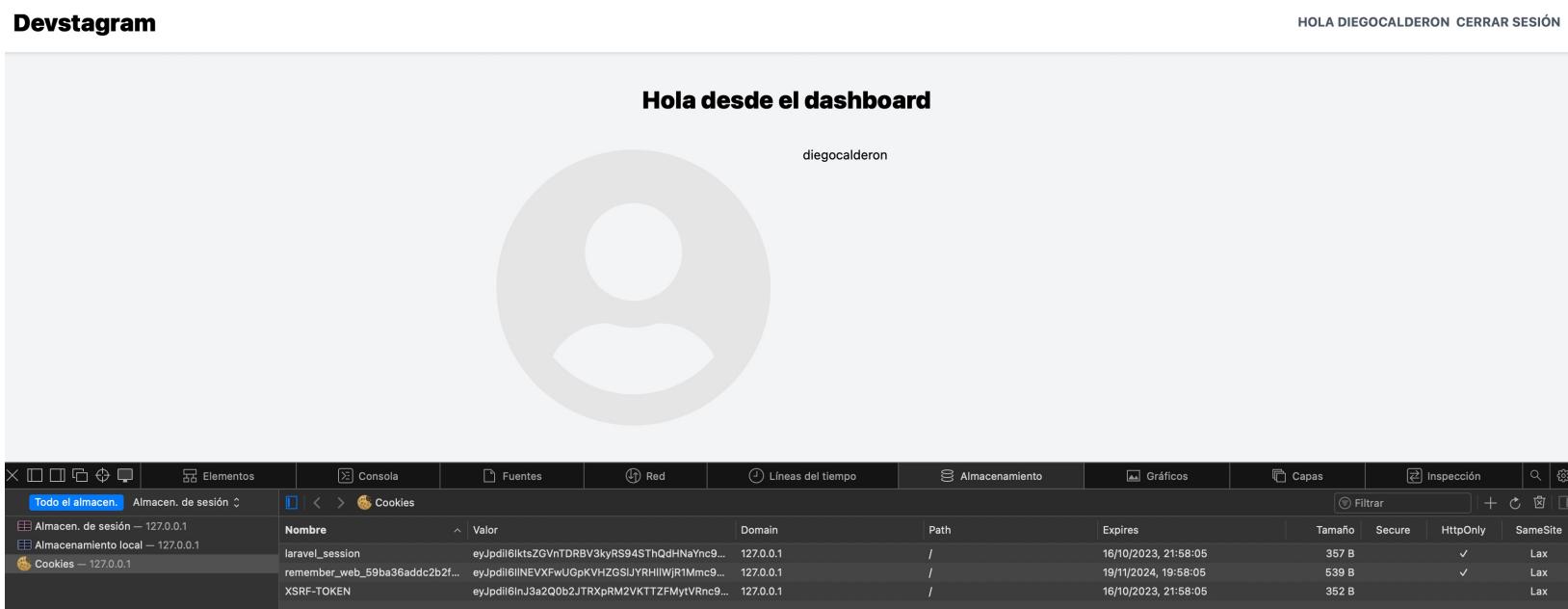
    if (!auth()->attempt($request->only( keys: 'email', 'password'),$request->remember)){
        return back()->with('mensaje','Credenciales incorrectas');
    }

    return redirect()->route( route: 'post.index');
}
```

Abriendo el inspector de página y buscando las cookies, analicemos un poco que esta pasando cuando le damos iniciar sesión con el checkbox activo.

Mantener sesión abierta

Podemos observar que se ha creado una nueva a diferencia de cuando iniciamos sin seleccionar el checkbox.



The screenshot shows the Devstagram dashboard with the message "Hola desde el dashboard". Below it is a placeholder for a user profile picture. At the top right, there are links for "HOLA DIEGOCALDERON" and "CERRAR SESIÓN". The browser's developer tools are open, specifically the "Cookies" tab under the "Almacen. de sesión" section. The table lists three cookies:

Nombre	Valor	Domain	Path	Expires	Tamaño	Secure	HttpOnly	SameSite
laravel_session	eyJpdi6lkt...RS94SThQd-HaYnc9...	127.0.0.1	/	16/10/2023, 21:58:05	357 B	✓	Lax	
remember_web_59ba36addc2b2f...	eyJpdi6lNEVXFwUGpKVHZGSJYRHiiWjR1Mmc9...	127.0.0.1	/	19/11/2024, 19:58:05	539 B	✓	Lax	
XSRF-TOKEN	eyJpdi6lnJ3a2Q0b2JTRxpRM2VKTZFMtV...Rnc9...	127.0.0.1	/	16/10/2023, 21:58:05	352 B		Lax	

Pero eso no es todo, observemos la tabla users, donde evidenciaremos un remember_token

name	email	email_verified_at	password	remember_token	created_at	updated_at	username
motato	motato23@gmail.com	NULL	\$2y\$10\$.59UMHae8DUnKKPnPeyneQB20WVfZnrJMdrir6ZRy3...	NULL	2023-10-13 16:42:28	2023-10-13 16:42:28	motato123
diego calderon	diego@info.com	NULL	\$2y\$10\$7W1XpS7z8LDRSckFomCeHO1EmsNQ8z8y11TcsEMpAS...	LqZpubsfV0YMXICVeg2LG8vqlTzP2FasqUdtfnGwxJf7alSJMu...	2023-10-16 23:29:34	2023-10-16 23:29:34	diegocalderon

Mantener sesión abierta

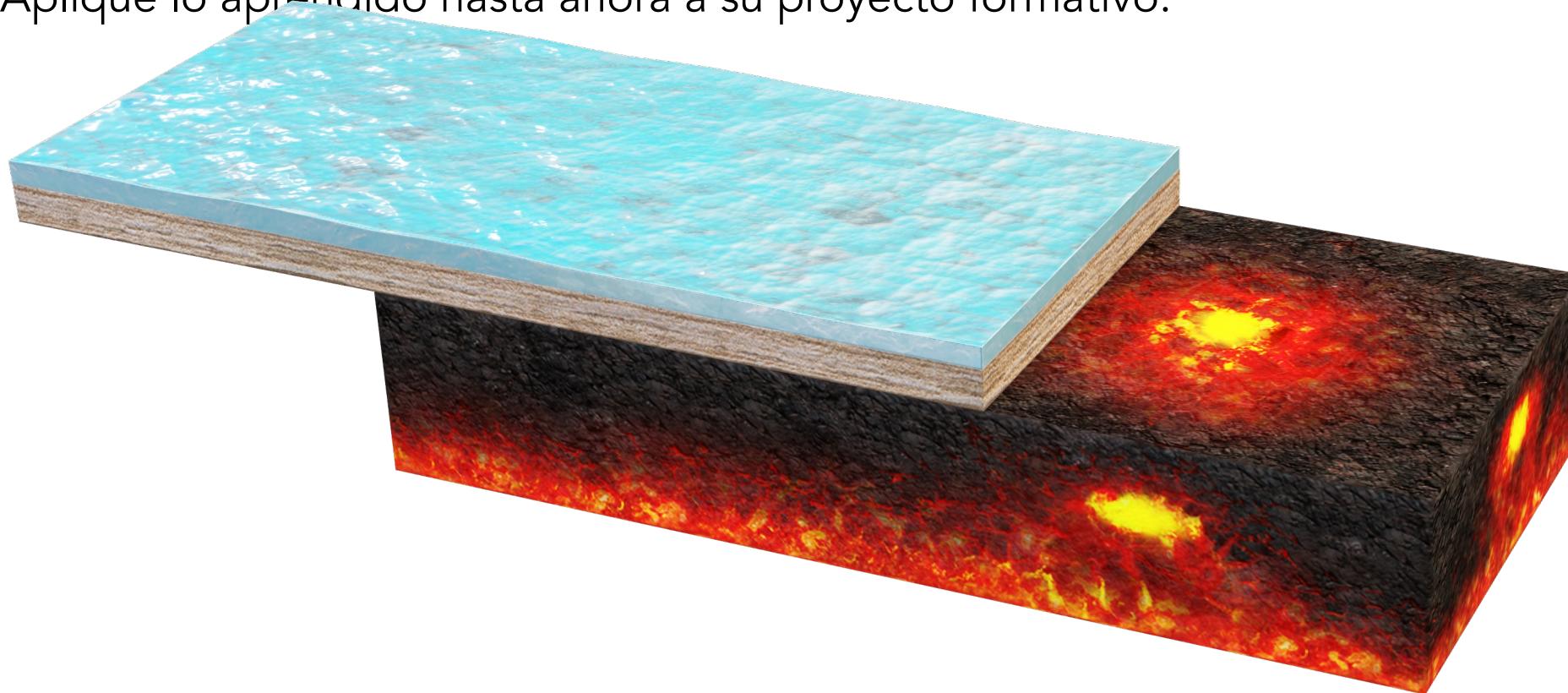


De esta forma nuestro registro se guarda en la base de datos y se estará comparando con la cookie, donde si es el mismo que esta inicializado, mantendrá al usuario todo el tiempo dentro del aplicativo y su sesión no se cerrará.



DESARROLLE

Aplique lo aprendido hasta ahora a su proyecto formativo.





G R A C I A S

Línea de atención al ciudadano: 01 8000 910270
Línea de atención al empresario: 01 8000 910682



www.sena.edu.co