

UNIVERSIDAD AUTÓNOMA GABRIEL RENÉ MORENO

**FACULTAD DE INGENIERÍA EN CIENCIAS DE LA
COMPUTACIÓN Y TELECOMUNICACIONES**



INVESTIGACION U4: JET STREAM EN LARAVEL

MATERIA Tecnología Web

NOMBRE Pablo Michael Tardio Ventura

CODIGO 217064957 **GRUPO** SC

FECHA 13/02/2021

DOCENTE Ing. Evans Balcázar Veizaga

CONTENIDO

Introducción	4
Objetivos	4
Objetivo general	4
Objetivos específico.....	4
Desarrollo.....	4
Autenticación	4
Introducción	4
Laravel Forify	5
Vistas, Páginas	5
Personalización de las vistas de autenticación	5
Personalización del proceso de autenticación	5
Restablecimiento de contraseña	9
Perfiles	11
Introducción	11
Comportamiento	11
Vistas / Páginas	12
Fotos de perfil.....	12
Gestión de fotos de perfil.....	12
Eliminación de cuenta.....	13
Seguridad	13
Introducción.....	13
Comportamiento	14
Vistas / Páginas	14
Desactivación de la autenticación de dos factores	15
Equipos.....	15
Creación de equipo	16
Comportamiento	16
Vistas / Páginas	17
Inspeccionar equipos de usuarios	17
El equipo actual	17
El objeto del equipo	17
Gestión de miembros.....	18
Vistas / páginas de administración de miembros	18

Invitaciones	19
Roles / Permisos	20
Autorización	20
Combinando permisos de equipo con permisos de API	21
CRUD con tailwind css.....	21
CRUD: la base de la gestión de datos	21
Tailwind	22
Resultado	24
Conclusiones	27
Bibliografía	28

INTRODUCCION

Es un scaffolding diseñado especialmente para el framework de PHP. Jetstream ofrece una mesa de trabajo prediseñada para comenzar a desarrollar aplicaciones con Laravel.

Alguna de las funcionalidades que trae consigo Jetstream son las siguientes:

- Verificación por correo electrónico
- Autenticación de dos factores
- Administrador de sesiones
- Soporte de API
- Gestión de equipos
- Etc.

OBJETIVOS

OBJETIVO GENERAL

Investigar sobre Laravel usado en su conjunto con Jetstream

OBJETIVOS ESPECIFICO

Describir la funcionalidad y conceptos siguientes:

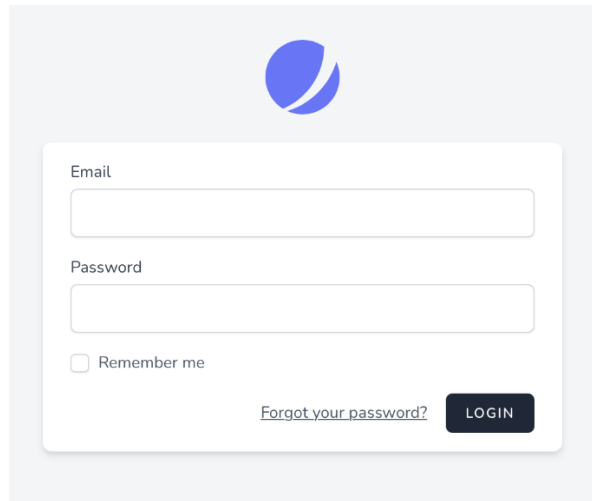
- Autenticación
- Perfiles
- Seguridad
- Equipos
- CRUD (Persona) con Tailwind CSS

DESARROLLO

AUTENTICACION

INTRODUCCION

Laravel Jetstream integra automáticamente las funciones de inicio de sesión, inicio de sesión de dos factores, registro, restablecimiento de contraseña y verificación de correo electrónico para su proyecto, lo que le permite comenzar a crear las funciones que le interesan en lugar de preocuparse por los detalles esenciales de la autenticación del usuario.



LARAVEL FORIFY

las partes de autenticación de Jetstream están impulsadas por [Laravel Fortify](#), que es un backend de autenticación independiente de front-end para Laravel. Básicamente, Fortify define las rutas y los controladores para implementar las funciones de autenticación de la aplicación, mientras que la interfaz de usuario de Jetstream realiza solicitudes a esas rutas.

Cuando se instala Jetstream, el `config/fortify.php` archivo de configuración se instala en su aplicación. Dentro de este archivo de configuración, puede personalizar varios aspectos del comportamiento de Fortify, como la protección de autenticación que se debe usar, a dónde se debe redirigir a los usuarios después de la autenticación, y más.

Dentro del `fortify` archivo de configuración, también puede deshabilitar funciones completas de Fortify, como la capacidad de actualizar la información del perfil o las contraseñas.

VISTAS, PAGINAS

Cuando se usa el Livewire Stack, la vista de inicio de sesión se muestra usando la `resources/views/auth/login.blade.php` plantilla Blade. Cuando se usa la pila de inercia, esta vista se muestra usando la `resources/js/Pages/Auth/Login.vue` plantilla. Los directorios que contienen estas vistas también contienen otras vistas / páginas relacionadas con la autenticación para su aplicación.

PERSONALIZACIÓN DE LAS VISTAS DE AUTENTICACIÓN

Laravel Jetstream generará automáticamente las vistas adecuadas para el inicio de sesión de su aplicación y otras pantallas de autenticación. Sin embargo, a veces es posible que desee personalizar cómo se representa una vista de autenticación en particular o los datos que recibe la vista.

Toda la lógica de representación de la vista de autenticación se puede personalizar utilizando los métodos apropiados disponibles a través de la `Laravel\Fortify\Fortify` clase. Por lo general, debe llamar a este método desde el bootmétodo de la `App\Providers\JetstreamServiceProvider` clase de su aplicación :

PERSONALIZACIÓN DEL PROCESO DE AUTENTICACIÓN

PERSONALIZACIÓN DE LA AUTENTICACIÓN DE USUARIO

A veces, es posible que desee tener una personalización completa sobre cómo se autentican las credenciales de usuario y cómo se recuperan los usuarios de la base de datos de su aplicación. Afortunadamente, Jetstream te permite lograr esto fácilmente usando el *Fortify::authenticateUsing* método.

El *authenticateUsing* método acepta un cierre que recibe la solicitud HTTP entrante. El cierre es responsable de validar las credenciales de inicio de sesión adjuntas a la solicitud y devolver la instancia de usuario asociada. Si las credenciales no son válidas o no se puede encontrar ningún usuario, nullo falso deben ser devueltas por el cierre. Normalmente, este método debe llamarse desde el **boot** método de su *JetstreamServiceProvider*:

```
use App\Models\User;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Hash;
use Laravel\Fortify\Fortify;

/**
 * Bootstrap any application services.
 *
 * @return void
 */
public function boot()
{
    // ...

    Fortify::authenticateUsing(function (Request $request) {
        $user = User::where('email', $request->email)->first();

        if ($user &&
            Hash::check($request->password, $user->password)) {
            return $user;
        }
    });
}
```

php

Si prefiere encapsular su proceso de autenticación personalizado dentro de una clase en lugar de un cierre, puede pasar una matriz "invocable" de PHP al *authenticateUsing* método:

```
use App\Actions\AuthenticateLoginAttempt;
use Laravel\Fortify\Fortify;

Fortify::authenticateUsing([new AuthenticateLoginAttempt, '__invoke']);
```

php

PERSONALIZACIÓN DE LA CANALIZACIÓN DE AUTENTICACIÓN

Laravel Fortify, que es la biblioteca de autenticación subyacente de Jetstream, autentica las solicitudes de inicio de sesión a través de una canalización de clases invocables.

Si lo desea, puede definir una canalización personalizada de clases por las que se deben canalizar las solicitudes de inicio de sesión. Cada clase debe tener un método `__invoke` que reciba la instancia `Illuminate\Http\Request` entrante y, como el middleware, una variable `$next` que se invoca para pasar la solicitud a la siguiente clase en la canalización.

Para definir su canalización personalizada, puede utilizar el método `Fortify::authenticateThrough`. Este método acepta un cierre que debería devolver la matriz de clases para canalizar la solicitud de inicio de sesión. Normalmente, este método debe llamarse desde el método `boot` de su clase `App\Providers\JetstreamServiceProvider`.

El siguiente ejemplo contiene la definición de canalización predeterminada que puede utilizar como punto de partida al realizar sus propias modificaciones:

```
use Laravel\Fortify\Actions\AttemptToAuthenticate;
use Laravel\Fortify\Actions\EnsureLoginIsNotThrottled;
use Laravel\Fortify\Actions\PrepareAuthenticatedSession;
use Laravel\Fortify\Actions\RedirectIfTwoFactorAuthenticatable;
use Laravel\Fortify\Fortify;
use Illuminate\Http\Request;

Fortify::authenticateThrough(function (Request $request) {
    return array_filter([
        config('fortify.limiters.login') ? null : EnsureLoginIsNotThrottled::class,
        RedirectIfTwoFactorAuthenticatable::class,
        AttemptToAuthenticate::class,
        PrepareAuthenticatedSession::class,
    ]);
});
```

php

RESTABLECIMIENTO DE CONTRASEÑA

Además de las vistas y las acciones relacionadas con el inicio de sesión / autenticación, Jetstream también ofrece los recursos necesarios para que un usuario restablezca su contraseña en caso de que se olvide. Por supuesto, esta función utiliza las funciones de restablecimiento de contraseña subyacentes de Laravel .

Correo Laravel

Antes de usar la función de restablecimiento de contraseña, debe asegurarse de que su aplicación Laravel esté configurada para [enviar correos electrónicos](#) . De lo contrario, Laravel no podrá enviar enlaces de restablecimiento de contraseña a los usuarios de su aplicación.

COMPORTAMIENTO

Como es típico de la mayoría de las funciones de Jetstream, la lógica ejecutada para satisfacer las solicitudes de restablecimiento de contraseña se puede encontrar en una clase de acción dentro de su aplicación. Recuerde, las acciones son clases granulares que son responsables de realizar una sola tarea relacionada con una característica de Jetstream o Fortify.

Específicamente, la `App\Actions\Fortify\ResetUserPassword` clase se invocará cuando un usuario restablezca su contraseña. Esta acción es responsable de validar la nueva contraseña del usuario y actualizar la contraseña en la instancia del usuario. Por lo tanto, cualquier personalización que desee realizar en la lógica de restablecimiento de la contraseña del usuario debe realizarse en esta clase. La acción recibe una matriz de `$input` que contiene toda la entrada de la solicitud entrante.

REGLAS DE VALIDACIÓN DE CONTRASEÑA

Los `App\Actions\Fortify\CreateNewUser`, `App\Actions\Fortify\ResetUserPassword` y `App\Actions\Fortify\UpdateUserPassword` todas las acciones utilizan el rasgo `App\Actions\Fortify\PasswordValidationRules`.

Como habrá notado, el `App\Actions\Fortify\PasswordValidationRules` rasgo utiliza un objeto `Laravel\Fortify\Rules\Password` de regla de validación personalizado . Este objeto le permite personalizar fácilmente los requisitos de contraseña para su aplicación. De forma predeterminada, la regla requiere una contraseña de al menos ocho caracteres. Sin embargo, puede utilizar los siguientes métodos para personalizar los requisitos de la contraseña:


```

use Laravel\Fortify\Rules\Password;

// Require at least 10 characters...
(new Password)->length(10)

// Require at least one uppercase character...
(new Password)->requireUppercase()

// Require at least one numeric character...
(new Password)->requireNumeric()

// Require at least one special character...
(new Password)->requireSpecialCharacter()

```

php

Por supuesto, estos métodos pueden encadenarse para definir las reglas de validación de contraseña para su aplicación:

```

(new Password)->length(10)->requireSpecialCharacter()

```

php

PERSONALIZACIÓN DE VISTAS DE AUTENTICACIÓN POR INERCIA

Si su aplicación utiliza la pila de inercia, puede devolver las páginas de inercia desde sus cierres de personalización de vista:

```

use Illuminate\Support\Facades\Route;
use Inertia\Inertia;
use Laravel\Fortify\Fortify;

Fortify::loginView(function () {
    return Inertia::render('Auth/Login', [
        'canResetPassword' => Route::has('password.request'),
        'status' => session('status'),
    ]);
});

```

php

RESTABLECIMIENTO DE CONTRASEÑA

Además de las vistas y las acciones relacionadas con el inicio de sesión / autenticación, Jetstream también ofrece los recursos necesarios para que un usuario restablezca su contraseña en caso de que se olvide. Por supuesto, esta función utiliza las [funciones de restablecimiento de contraseña](#) subyacentes de Laravel.

Correo Laravel

Antes de usar la función de restablecimiento de contraseña, debe asegurarse de que su aplicación Laravel esté configurada para [enviar correos electrónicos](#). De lo contrario, Laravel no podrá enviar enlaces de restablecimiento de contraseña a los usuarios de su aplicación.

COMPORTAMIENTO

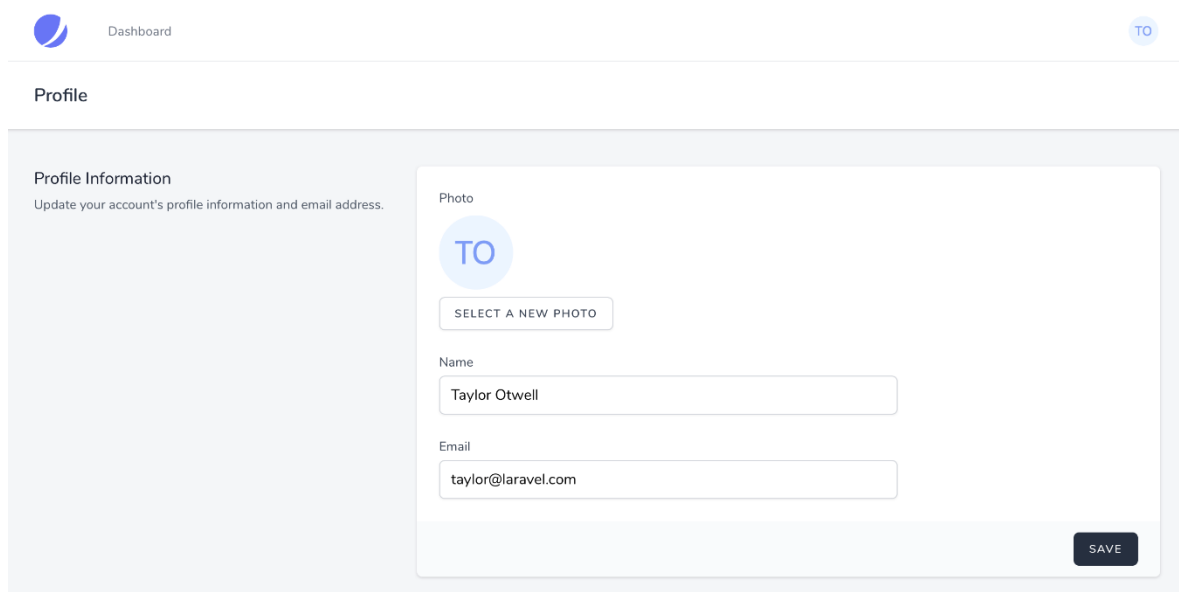
Como es típico de la mayoría de las funciones de Jetstream, la lógica ejecutada para satisfacer las solicitudes de restablecimiento de contraseña se puede encontrar en una clase de acción dentro de su aplicación. Recuerde, las acciones son clases granulares que son responsables de realizar una sola tarea relacionada con una característica de Jetstream o Fortify.

Específicamente, la `App\Actions\Fortify\ResetUserPassword` se invocará cuando un usuario restablezca su contraseña. Esta acción es responsable de validar la nueva contraseña del usuario y actualizar la contraseña en la instancia del usuario. Por lo tanto, cualquier personalización que desee realizar en la lógica de restablecimiento de la contraseña del usuario debe realizarse en esta clase. La acción recibe una matriz de *\$input* que contiene toda la entrada de la solicitud entrante.

PERFILES

INTRODUCCION

El usuario accede a las funciones de gestión de perfiles de Laravel Jetstream utilizando el menú desplegable de navegación de perfil de usuario superior derecho. Jetstream crea vistas y acciones que permiten al usuario actualizar su nombre, dirección de correo electrónico y, opcionalmente, su foto de perfil.



The screenshot shows the 'Profile' page in a Laravel Jetstream application. At the top, there's a navigation bar with a 'Dashboard' link and a user profile icon labeled 'TO'. Below the navigation bar, the 'Profile' section is active. The main content area is titled 'Profile Information' with a subtitle 'Update your account's profile information and email address.' To the right, there's a form with three fields: 'Photo' (with a circular profile picture placeholder labeled 'TO' and a 'SELECT A NEW PHOTO' button), 'Name' (with a text input field containing 'Taylor Otwell'), and 'Email' (with a text input field containing 'taylor@laravel.com'). A 'SAVE' button is located at the bottom right of the form.

COMPORTAMIENTO

Como es típico de la mayoría de las funciones de Jetstream, la lógica ejecutada para satisfacer las solicitudes de actualización de perfil se puede encontrar en una clase de acción dentro de su aplicación. Específicamente, la `App\Actions\Fortify\UpdateUserProfileInformation` se invocará cuando el usuario actualice su perfil. Esta acción es responsable de validar la entrada y actualizar la información del perfil del usuario.

Por lo tanto, cualquier personalización que desee realizar en la administración de esta información de su aplicación debe realizarse en esta clase. Cuando se invoca, la acción recibe el actualmente

autenticado `$user` una matriz `$input` que contiene toda la entrada de la solicitud entrante, incluida la foto de perfil actualizada, si corresponde.

Gestión de información adicional

Si necesita administrar información adicional sobre el usuario, no se sienta obligado a agregar información adicional a la tarjeta "Información de perfil" dentro de la interfaz de usuario de su aplicación Jetstream. Puede agregar elementos y formularios de la interfaz de usuario adicionales al panel del perfil del usuario que sean específicos de la información que desea administrar.

VISTAS / PÁGINAS

Cuando se usa la pila Livewire, el formulario de información del perfil del usuario se muestra usando la `resources/views/profile/update-profile-information-form.blade.php` plantilla Blade. Cuando se usa la pila de inercia, esta vista se muestra usando la plantilla `resources/js/Pages/Profile/UpdateProfileInformationForm.vue`

Cada una de estas plantillas recibirá el objeto de usuario autenticado completo para que pueda agregar campos adicionales a estos formularios según sea necesario. Cualquier entrada adicional agregada a los formularios se incluirá en la `$input` matriz que se pasa a su acción `UpdateUserProfileInformation`

FOTOS DE PERFIL

Si desea permitir que los usuarios carguen fotos de perfil personalizadas, debe habilitar la función en el `config/jetstream.php` archivo de configuración de su aplicación .

GESTIÓN DE FOTOS DE PERFIL

La función de foto de perfil de Jetstream es compatible con el `Laravel\Jetstream\HasProfilePhoto` rasgo que se adjunta automáticamente a su `App\Models\User` clase durante la instalación de Jetstream.

Este rasgo contiene métodos tales como `updateProfilePhoto`, `getProfilePhotoUrlAttribute`, `defaultProfilePhotoUrl`, y `profilePhotoDisk` que todos pueden ser sobrescritos por su propia `App\Models\User` clase si necesita personalizar su comportamiento. Se le anima a leer el código fuente de este rasgo para que tenga una comprensión completa de las funciones que proporciona a su aplicación.

El `updateProfilePhoto` método es el método principal utilizado para almacenar fotos de perfil y es llamado por la `App\Actions\Fortify\UpdateUserProfileInformation` clase de acción de su aplicación .

Vapor de Laravel

De forma predeterminada, el `s3` disco se usará para almacenar fotos de perfil cuando su aplicación Jetstream se esté ejecutando dentro de [Laravel Vapor](#) .

ELIMINACIÓN DE CUENTA

La pantalla de gestión de perfiles también incluye un panel de acción que permite al usuario eliminar su cuenta de aplicación. Cuando el usuario elige eliminar su cuenta, `App\Actions\Jetstream\DeleteUserse` invoca la clase de acción. Puede personalizar la lógica de eliminación de cuenta de su aplicación dentro de esta clase.

La función de eliminación de cuenta se puede deshabilitar eliminando la función del `config/jetstream.php` archivo de configuración de su aplicación :

```
use Laravel\Jetstream\Features;

'features' => [
    Features::termsAndPrivacyPolicy(),
    Features::profilePhotos(),
    Features::api(),
    Features::teams(),
    // Features::accountDeletion(),
],
```

php

SEGURIDAD

INTRODUCTION

Laravel Jetstream incorpora automáticamente el soporte de autenticación de dos factores para todas las aplicaciones Jetstream. El usuario accede a las funciones de seguridad de Laravel Jetstream utilizando el menú desplegable de navegación de perfil de usuario superior derecho. Dentro de este tablero, Jetstream diseña vistas que permiten al usuario habilitar y administrar la autenticación de dos factores para su cuenta.

Cuando un usuario habilita la autenticación de dos factores para su cuenta, debe escanear el código QR proporcionado utilizando una aplicación de autenticación TOTP gratuita como Google Authenticator. Además, deben almacenar los códigos de recuperación enumerados en un administrador de contraseñas seguro como Password .

Si el usuario pierde el acceso a su dispositivo móvil, la página de inicio de sesión de Jetstream le permitirá autenticarse utilizando uno de sus códigos de recuperación en lugar del token temporal proporcionado por la aplicación de autenticación de su dispositivo móvil.

Update Password

Ensure your account is using a long, random password to stay secure.

Current Password

New Password

Confirm Password

SAVE


Two Factor Authentication

Add additional security to your account using two factor authentication.

You have enabled two factor authentication.

When two factor authentication is enabled, you will be prompted for a secure, random token during authentication. You may retrieve this token from your phone's Google Authenticator application.

Two factor authentication is now enabled. Scan the following QR code using your phone's authenticator application.



Store these recovery codes in a secure password manager. They can be used to recover access to your account if your two factor authentication device is lost.

Yn8gp4eeyj-RtucpjK5AK

iRZc550ebK-q0ZzUsdIMv

emR6P6tKX0-utdxCyn8m5

M1QCT60urV-5NvxYjmvA0

hhK5knYHwu-jz1uhYLOu3

IZTHCqs2Wn-zo5MM0FM5b

52h06MtVEs-0uYOL8godE

C2mHzWQeK0-EkqhaXXm0c

REGENERATE RECOVERY CODES

DISABLE

COMPORTAMIENTO

La mayoría de las funciones de Jetstream se pueden personalizar mediante clases de acción. Sin embargo, por motivos de seguridad, los servicios de autenticación de dos factores de Jetstream están encapsulados dentro de Jetstream y no deberían requerir personalización.

VISTAS / PÁGINAS

Por lo general, las vistas y páginas correspondientes de la función de autenticación de dos factores no deberían requerir personalización, ya que ya están completas. Sin embargo, sus ubicaciones se describen a continuación en caso de que necesite realizar pequeños ajustes de presentación en estas páginas.

Cuando se usa la pila Livewire, la vista de administración de autenticación de dos factores se muestra usando la `resources/views/profile/two-factor-authentication-form.blade.php` plantilla Blade. Cuando se usa

la pila de inercia, esta vista se muestra usando
la `resources/js/Pages/Profile/TwoFactorAuthenticationForm.vue` plantilla.

DESACTIVACIÓN DE LA AUTENTICACIÓN DE DOS FACTORES

Si lo desea, puede deshabilitar la compatibilidad con la autenticación de dos factores eliminando la función de la `featuresmatriz` del `config/fortify.php` archivo de configuración de su aplicación :

```
use Laravel\Fortify\Features; php

'features' => [
    Features::registration(),
    Features::resetPasswords(),
    Features::emailVerification(),
    Features::updateProfileInformation(),
    Features::updatePasswords(),
    // Features::twoFactorAuthentication([
    //     'confirmPassword' => true,
    // ]),
],
```

EQUIPOS

Si instaló Jetstream usando la opción `--teams`, su aplicación tendrá un scaffolding para apoyar la creación y administración del equipo.

Las características del equipo de Jetstream permiten que cada usuario registrado cree y pertenezca a varios equipos. Por defecto, cada usuario registrado pertenecerá a un equipo "Personal". Por ejemplo, si un usuario llamado "Sally Jones" crea una nueva cuenta, será asignado a un equipo llamado "Equipo de Sally". Después del registro, el usuario puede cambiar el nombre de este equipo o crear equipos adicionales.



Team Settings

Team Name

The team's name and owner information.

Team Owner



Taylor Otwell
taylor@laravel.com

Team Name

Taylor's Team

SAVE

Add Team Member

Add a new team member to your team, allowing them to collaborate with you.

Please provide the email address of the person you would like to add to this team. The email address must be associated with an existing account.

Email

Role

Administrator

Administrator users can perform any action.

Editor

Editor users have the ability to read, create, and update.

ADD

Team Members

All of the people that are part of this team.



Abigail Otwell

[Administrator](#)

[Remove](#)

Equipos Jetstream

Los andamios y las opiniones del equipo de Jetstream pueden no funcionar para todas las aplicaciones. Si no funciona para su caso de uso, no dude en crear una aplicación Jetstream que no sea de equipo y agregar manualmente la funcionalidad de equipo a su aplicación según sus propias necesidades.

CREACIÓN DE EQUIPO

Se accede a la vista de creación del equipo a través del menú desplegable de navegación del usuario en la parte superior derecha.

COMPORTAMIENTO

Como muchas otras funciones de Jetstream, la lógica de creación y eliminación de equipos se puede personalizar modificando las clases de acción relevantes dentro de

su `app/Actions/Jetstreamdirectorio`. Estas acciones incluyen `CreateTeam`, `UpdateTeamName` y `DeleteTeam`. Cada una de estas acciones se invoca cuando el usuario realiza su tarea correspondiente en la interfaz de usuario de la aplicación. Puede modificar estas acciones según sea necesario en función de las necesidades de su aplicación.

VISTAS / PÁGINAS

Cuando se usa la pila Livewire, la vista de creación del equipo se muestra usando la `resources/views/teams/create-team-form.blade.php` plantilla Blade. Cuando se usa la pila de inercia, esta vista se muestra usando la `resources/js/Pages/Teams/CreateTeamForm.vue` plantilla. Los campos de entrada adicionales que se especifican en los formularios de creación de equipos se proporcionarán a la `App\Actions\Jetstream\CreateTeam` clase de acción cuando el usuario cree un equipo.

INSPECCIONAR EQUIPOS DE USUARIOS

Se puede acceder a la información sobre los equipos de un usuario a través de los métodos proporcionados por el `Laravel\Jetstream\HasTeams` rasgo. Este rasgo se aplica automáticamente al `App\Models\User` modelo de su aplicación durante la instalación de Jetstream. Este rasgo proporciona una variedad de métodos útiles que le permiten inspeccionar los equipos de un usuario:

EL EQUIPO ACTUAL

Cada usuario dentro de una aplicación Jetstream tiene un "equipo actual". Este es el equipo para el que el usuario visualiza activamente los recursos. Por ejemplo, si está creando una aplicación de calendario, su aplicación mostraría los próximos eventos del calendario para el equipo actual del usuario.

Puede acceder al equipo actual del usuario utilizando la `$user->currentTeam` relación Eloquent. Esta relación puede usarse para determinar el alcance de sus otras consultas de Eloquent por parte del equipo actual del usuario:

```
use App\Models\Calendar;

return Calendar::where(
    'team_id', $request->user()->currentTeam->id
)->get();
```

php

Cambiar de equipo

Un usuario puede cambiar su equipo actual a través del menú de "cambio de equipo" disponible dentro de la barra de navegación Jetstream.

EL OBJETO DEL EQUIPO

El objeto de equipo al que se accede a través **\$user->currentTeam** de otras consultas de Eloquent relacionadas con el equipo de Jetstream proporciona una variedad de métodos útiles para inspeccionar los atributos y las relaciones del equipo:

```
// Access the team's owner...
$team->owner : App\Models\User

// Get all of the team's users, including the owner...
$team->allUsers() : Illuminate\Database\Eloquent\Collection

// Get all of the team's users, excluding the owner...
$team->users : Illuminate\Database\Eloquent\Collection

// Determine if the given user is a team member...
$team->hasUser($user) : bool

// Determine if the team has a member with the given email address...
$team->hasUserWithEmail($emailAddress) : bool

// Determine if the given user is a team member with the given permission...
$team->userHasPermission($user, $permission) : bool
```

GESTIÓN DE MIEMBROS

Los miembros del equipo se pueden agregar y eliminar de un equipo a través de la vista "Configuración del equipo" de Jetstream.

ACCIONES DE GESTIÓN DE MIEMBROS

Al igual que el proceso de personalización para otras funciones de Jetstream, la lógica de adición de miembros del equipo se puede personalizar modificando la **App\Actions\Jetstream\AddTeamMember** clase de acción. El `add` método de la clase se invoca con el usuario actualmente autenticado, la **Laravel\Jetstream\Team** instancia, la dirección de correo electrónico del usuario que se agrega al equipo y el rol (si corresponde) del usuario que se agrega al equipo.

Esta acción es responsable de validar que el usuario realmente se puede agregar al equipo y luego agregar el usuario al equipo. Puede personalizar esta acción según las necesidades de su aplicación particular.

La eliminación de miembros del equipo se puede personalizar modificando la **App\Actions\Jetstream\RemoveTeamMember** clase de acción.

VISTAS / PÁGINAS DE ADMINISTRACIÓN DE MIEMBROS

Cuando se usa la pila Livewire, la vista del administrador de miembros del equipo se muestra usando la `resources/views/teams/team-member-manager.blade.php` plantilla Blade. Cuando se usa la pila de inercia, esta vista se muestra usando la `resources/js/Pages/Teams/TeamMemberManager.vue` plantilla. Generalmente, estas plantillas no deberían requerir personalización.

INVITACIONES

De forma predeterminada, Jetstream simplemente agregará a su equipo cualquier usuario de aplicación existente que especifique. Sin embargo, muchas aplicaciones optan por enviar correos electrónicos de invitación a los usuarios que están invitados a los equipos. Si el usuario no tiene una cuenta, el correo electrónico de invitación puede indicarle que cree una cuenta y acepte la invitación. O, si el usuario ya tiene una cuenta, puede aceptar o ignorar la invitación.

Afortunadamente, Jetstream le permite habilitar invitaciones de miembros del equipo para su aplicación con solo unas pocas líneas de código. Para comenzar, pase la `invitations` opción cuando habilite la función "equipos" para su aplicación. Esto se puede hacer modificando la `features` matriz del `config/jetstream.php` archivo de configuración de su aplicación :

```
use Laravel\Jetstream\Features;

'features' => [
    Features::termsAndPrivacyPolicy(),
    Features::profilePhotos(),
    Features::api(),
    Features::teams(['invitations' => true]),
    Features::accountDeletion(),
],
```

Una vez que haya habilitado la función de invitaciones de Jetstream, los usuarios que estén invitados a los equipos recibirán un correo electrónico de invitación con un enlace para aceptar la invitación del equipo. Los usuarios no serán miembros de pleno derecho del equipo hasta que se acepte la invitación.

ACCIONES DE INVITACIÓN

Cuando se invita a un usuario al equipo, la `App\Actions\Jetstream\InviteTeamMember` acción de su aplicación se invocará con el usuario actualmente autenticado, el equipo al que está invitado el nuevo usuario, la dirección de correo electrónico del usuario invitado y, opcionalmente, el rol que debe asignarse. el usuario una vez que se une al equipo. Puede revisar esta acción o modificarla según las necesidades de su propia aplicación.

Correo Laravel

Antes de usar la función de invitación al equipo, debe asegurarse de que su aplicación Laravel esté configurada para **enviar correos electrónicos** [↗](#). De lo contrario, Laravel no podrá enviar correos electrónicos de invitación al equipo a los usuarios de su aplicación.

ROLES / PERMISOS

A cada miembro del equipo agregado a un equipo se le puede asignar una función determinada, y a cada función se le asigna un conjunto de permisos. Los permisos de rol se definen en la **App\Providers\JetstreamServiceProvider** clase de su aplicación usando el **Jetstream::role** método. Este método acepta un "slug" para el rol, un nombre de rol fácil de usar, los permisos del rol y una descripción del rol. Esta información se utilizará para mostrar el rol dentro de la vista de administración de miembros del equipo.

Por ejemplo, imagina que estamos construyendo una aplicación de administración de servidores como [Laravel Forge](#). Podríamos definir los roles de equipo de nuestra aplicación así:

```
Jetstream::defaultApiTokenPermissions(['read']);

Jetstream::role('admin', 'Administrator', [
    'server:create',
    'server:read',
    'server:update',
    'server:delete',
])->description('Administrator users can perform any action.');
```

```
Jetstream::role('support', 'Support Specialist', [
    'server:read',
])->description('Support specialists can read server information.');
```

php

Soporte de API de equipo

Cuando Jetstream se instala con el soporte del equipo, los permisos API disponibles se derivan automáticamente al combinar todos los permisos únicos disponibles para los roles. Por lo tanto,

`Jetstream::permissions` no es necesaria una llamada separada al método.

AUTORIZACIÓN

Por supuesto, necesitará una forma de autorizar que las solicitudes entrantes iniciadas por un miembro del equipo puedan ser realizadas por ese usuario. Los permisos del equipo de un usuario se pueden inspeccionar utilizando el `hasTeamPermission` método disponible a través del `Laravel\Jetstream\HasTeams` rasgo.

Por lo general, no es necesario inspeccionar el rol de un usuario. Solo necesita inspeccionar que el usuario tenga un permiso granular determinado. Los roles son simplemente un concepto de presentación utilizado para agrupar permisos granulares. Normalmente, ejecutará llamadas a este método dentro de las políticas de autorización de su aplicación :

```
return $user->hasTeamPermission($server->team, 'server:update');
```

php

COMBINANDO PERMISOS DE EQUIPO CON PERMISOS DE API

Al crear una aplicación Jetstream que proporcione soporte de API y soporte de equipo, debe verificar los permisos de equipo de una solicitud entrante y los permisos de token de API dentro de las políticas de autorización de su aplicación. Esto es importante porque un token de API puede tener la capacidad teórica de realizar una acción, mientras que un usuario en realidad no tiene esa acción otorgada a través de los permisos de su equipo:

```
/**
 * Determine whether the user can update a flight.
 *
 * @param \App\Models\User $user
 * @param \App\Models\Flight $flight
 * @return bool
 */
public function view(User $user, Flight $flight)
{
    return $user->belongsToTeam($flight->team) &&
           $user->hasTeamPermission($flight->team, 'flight:view') &&
           $user->tokenCan('flight:view');
}
```

php

CRUD CON TAILWIND CSS

CRUD: LA BASE DE LA GESTIÓN DE DATOS

El concepto CRUD está estrechamente vinculado a la gestión de datos digitales. CRUD hace referencia a un acrónimo en el que se reúnen las primeras letras de las cuatro operaciones fundamentales de aplicaciones persistentes en sistemas de bases de datos:

- Create (Crear registros)
- Read bzw. Retrieve (Leer registros)
- Update (Actualizar registros)
- Delete bzw. Destroy (Borrar registros)

En pocas palabras, CRUD resume las funciones requeridas por un usuario para crear y gestionar datos. Varios procesos de gestión de datos están basados en CRUD, en los que dichas operaciones están específicamente adaptadas a los requisitos del sistema y de usuario, ya sea para la gestión de bases de datos o para el uso de aplicaciones. Para los expertos, las operaciones son las herramientas de acceso típicas e indispensables para comprobar, por ejemplo, los problemas de la base de datos, mientras que para los usuarios, CRUD significa crear una cuenta (create) y utilizarla (read), actualizarla (update) o borrarla (delete) en cualquier momento. Dependiendo de la configuración regional, las operaciones CRUD pueden implementarse de diferentes maneras, como lo muestra la siguiente tabla:

En informática, CRUD es el acrónimo de "Crear, Leer, Actualizar y Borrar" (del original en inglés: Create, Read, Update and Delete), que se usa para referirse a las funciones básicas en bases de datos o la capa de persistencia en un software.

En algunos lugares, se utilizan las siglas ABM para lo mismo ("Alta, Baja y Modificación"), obviando la operación de "obtener"; el acrónimo ABC para "Altas, Bajas y Cambios"; ABML siendo la última letra (L) de "listar, listado o lectura"; ABMC siendo la 'C' de "Consulta"; o bien CLAB que sería la traducción literal del acrónimo ("Crear, Leer, Actualizar y Borrar") también se llega a usar el acrónimo ABCC ("Altas, Bajas, Cambios y Consultas").

También es usado el ABCDEF: "Agregar, Buscar, Cambiar, Desplegar (listar), Eliminar, Fichar (Ficha, cédula o Reporte de un registro)".

En programación, crear, leer, actualizar y borrar (con el acrónimo CRUD) son las cuatro funciones básicas de la persistencia de Bases de Datos. Términos alternativos son usados a veces cuando se definen las cuatro funciones básicas de CRUD, como "recuperar" en vez de "leer", "modificar" en vez de "actualizar" o "destruir" en vez de "borrar". CRUD se usa también a veces para describir convenciones de interfaz de usuario que facilita la vista, búsqueda y modificación de la información; a menudo se usa en programación de formularios (forms) e informes (reports). El término fue popularizado por primera vez por James Martin en su libro del año 1980 *Managing the Data-base Environment*. El acrónimo puede extenderse a CRUDL para cubrir el listado de gran cantidad de datos que conllevan una complejidad tal como paginación cuando los registros de datos son demasiado grandes para alojarse fácilmente en memoria.

TAILWIND

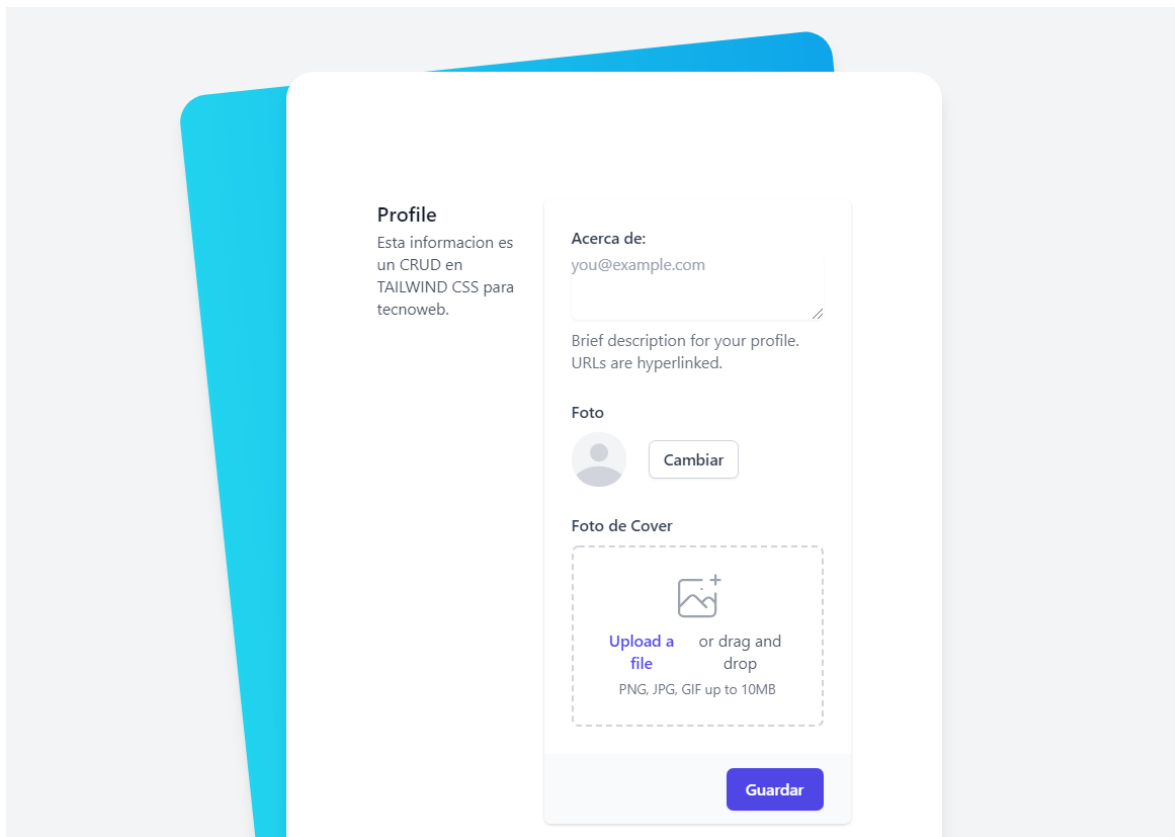
Tailwind CSS es un framework CSS que permite un desarrollo ágil, basado en clases de utilidad que se pueden aplicar con facilidad en el código HTML y unos flujos de desarrollo que permiten optimizar mucho el peso del código CSS.

Tailwind CSS es una potente herramienta para el desarrollo frontend. Está dentro de la clasificación de los frameworks CSS o también llamados frameworks de diseño. Permite a los desarrolladores y diseñadores aplicar estilos a los sitios web de una manera ágil y optimizada.

Tailwind permite escribir los estilos por medio de clases que se incluyen dentro del código HTML y que afectan a un aspecto muy concreto y específico de las CSS, por ejemplo, el fondo de un elemento, el color del texto o simplemente el margen por la parte de arriba. Este enfoque se conoce como "Atomic CSS", por aplicarse mediante estilos muy determinados y simples. En Tailwind CSS a estas clases se les llama "utility classes" o clases de utilidad en español.

Por tanto, Tailwind CSS no aporta muchos componentes. De hecho ofrece muy pocos componentes y son los desarrolladores los que los tienen que realizar bajo demanda del proyecto. En cambio lo que propone es entregar una enorme cantidad de clases de utilidad que combinadas en distintas variantes ofrecen prácticamente un número ilimitado de variantes de diseño, que permite una personalización del aspecto realmente única para cada proyecto.

Además Tailwind es una herramienta que se apoya en PostCSS para todo lo que es la generación del código CSS. Gracias a PostCSS se alcanza un flujo de desarrollo muy avanzado, personalizable, ágil y sobre todo, extremadamente optimizado. Con ello, el código CSS resultante de un proyecto es realmente ajustado, porque con PostCSS se consigue hacer que las clases que finalmente estén en el código de producción sean solamente las que el proyecto está usando realmente, ni una más.



Informacion Personal

Estos son sus datos personales

Nombre

Apellido

Direccion de Email

Pais / Region

Estados ▼

Calle

Ciudad

Estado /
Provincia

ZIP /
Postal

Guardar

RESULTADO



Name

user

Email

user@user.com


Password

Confirm Password

[Already registered?](#)


REGISTER


Dashboard


 **Laravel Jetstream**


Welcome to your Jetstream application!

Laravel Jetstream provides a beautiful, robust starting point for your next Laravel application. Laravel is designed to help you build your application using a development environment that is simple, powerful, and enjoyable. We believe you should love expressing your creativity through programming, so we have spent time carefully crafting the Laravel ecosystem to be a breath of fresh air. We hope you love it.

 **Documentation**
Laravel has wonderful documentation covering every aspect of the framework. Whether you're new to the framework or have previous experience, we recommend reading all of the documentation from beginning to end.
[Explore the documentation →](#)

 **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.
[Start watching Laracasts →](#)

 **Tailwind**
Laravel Jetstream is built with Tailwind, an amazing utility first CSS framework that doesn't get in your way. You'll be amazed how easily you can build and maintain fresh, modern designs with this wonderful framework at your fingertips.

 **Authentication**
Authentication and registration views are included with Laravel Jetstream, as well as support for user email verification and resetting forgotten passwords. So, you're free to get started what matters most: building your application.

Profile

Profile Information
Update your account's profile information and email address.

Name

user

Email

user@user.com

SAVE

Update Password
Ensure your account is using a long, random password to stay secure.

Current Password

New Password

Confirm Password

Two Factor Authentication

Add additional security to your account using two factor authentication.

You have not enabled two factor authentication.

When two factor authentication is enabled, you will be prompted for a secure, random token during authentication. You may retrieve this token from your phone's Google Authenticator application.

ENABLE

Browser Sessions

Manage and logout your active sessions on other browsers and devices.

If necessary, you may logout of all of your other browser sessions across all of your devices. Some of your recent sessions are listed below; however, this list may not be exhaustive. If you feel your account has been compromised, you should also update your password.

 OS X - Safari
127.0.0.1, [This device](#)

LOGOUT OTHER BROWSER SESSIONS

Delete Account

Permanently delete your account.

Once your account is deleted, all of its resources and data will be permanently deleted. Before deleting your account, please download any data or information that you wish to retain.

DELETE ACCOUNT



Dashboard

user ▾

API Tokens

Create API Token

API tokens allow third-party services to authenticate with our application on your behalf.

Token Name

Permissions

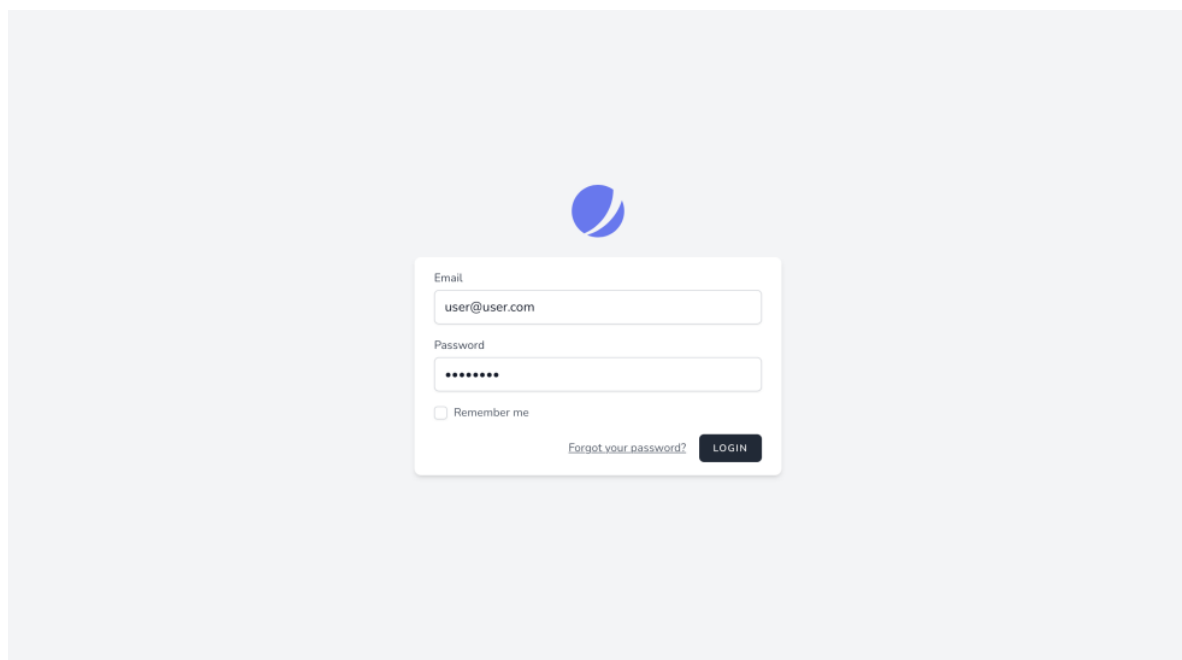
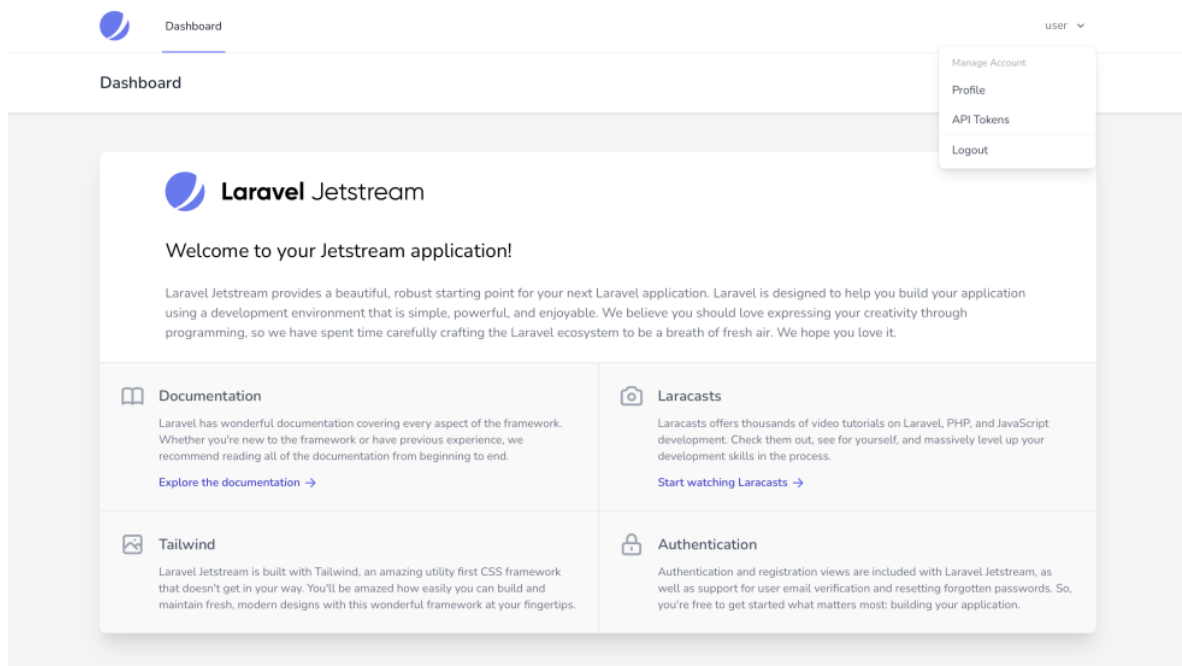
☐ create

☐ update

☒ read

☐ delete

CREATE



CONCLUSIONES

Se ha podido aprender acerca del uso de Jetstream y Tailwind Y poder aprender el concepto de un CRUD y su respectiva implementación en laravel y con esto poder concluir la investigación dada para la materia de Tecnología web

BIBLIOGRAFIA

1. Que es laravel JetStream: <https://gabrielchavez.me/que-es-laravel-jetstream/>
2. Autenticacion de laravel Jetstream:
<https://jetstream.laravel.com/2.x/features/authentication.html>
3. Gestion de perfiles <https://jetstream.laravel.com/2.x/features/profile-management.html>
4. Tailwind Css <https://tailwindui.com/>