**ApliSports**, es una tienda de aplicaciones deportivas.

Te registras como cliente, para comprar las apps, o también podes crear un usuario desarrollador, para además de comprar aplicaciones podes poner las tuyas a la venta.

***Un cliente, solo puede comprar aplicaciones.***  
El ***desarrollador***, además de comprar, puede ***subir*** sus app, ***editarlas*** y ***borrarlas***. Ambos tipos de usuario pueden ver las aplicaciones que compraron y el desarrollador tiene habilitada una pestaña para ver las apps que subió a la web.

*ApliSports está desarrollado con lenguaje HTML 5, PHP, JS y CSS y se utilizó Laravel, que es framework de código abierto para desarrollar aplicaciones y servicios web.*

*Otras herramientas que se usaron fueron:*

*\* Google fonts.  
\** *Fontawesome  
\** *Scrollreveal.  
\* El playgroung de DH y las clases virtuales.  
\* Php myadmin.  
\* Bootstrap.  
\* FlatIcon.  
\* YouTube.  
\* W3School.  
\* GitHub -> se creó el repositorio, y se fue subiendo diariamente los avances.*

**SITIO.**Desde el home lo primero que aparece es un jumbotron de bootstrap con información y un botón para registrarte. Cuando dejas de ser guest, esto no aparece. También se pueden ver las categorías a las que pertenecen las aplicaciones, la breve historia del sitio y un formulario de contacto para enviar una mensaje a ApliSports, y la información de las redes sociales, teléfono, mail y dirección.   
  
En el navbar se ven los links para poder iniciar sesión y en el caso de no tener una, registrarte en la web y anclados están los links de lo que se puede ver en el home.

Cuando ya estas registrado, retorna al home ya logueado. Sin necesidad de iniciar sesión.   
Ya con la sesión iniciada, el cliente/desarrollador puede ver el detalle de las aplicaciones y comprarlas.  
  
Para ver el detalle de cada aplicación, es necesario estar logueado. Si no lo estas, al ingresar te redirige a la vista de login. Una vez logueado el usuario podrá comprar la aplicación.

Cuando se compra una aplicación el usuario tiene un panel para ver el detalle de dicha compra, y puede dajar un comentario al respecto de la compra y calificarla. La calificación y comentario es único, ya que una vez registrado el comentario ese campo queda deshabilitado.  
  
El cliente/desarrollador, además del panel para ver el detalle de la compra, tiene uno para ver el listado de aplicaciones suyas. Puede subir apps, editarlas y borrarlas.  
  
Ningún usuario puede comprar dos veces la misma app. El sistema ya avisa que esa aplicación la tiene disponible.

***ApliSports es un sitio responsive, se adapta a todos los dispositivos***  mediante media querys y usando el sistema de grilla de bootstrap.

**BASE DE DATOS.**  
Se utilizó PHP MYADMIN. Las tablas fueron creadas por migraciones. Las información de la base se completo de esta manera:  
\* Users -> Factory y seeders.  
\* Applications -> Factory y seeders, además de las aplicaciones que fueron subiendo los desarrolladores.  
\* Categories-> Insert.  
\* Comments, Orders, Contacts -> se rellenan mediante los formularios que tiene la web.

Para la creación, edición y eliminación de un producto se tuvo en cuenta la clase virtual de brindó Alejandro Villafañe por Zoom. Desde esa clase se tomó la base para realizar esos pasos.

**MODELOS Y CONTROLLERS.**Los modelos y sus respectivos controladores se realizaron mediante la terminal que otorga VSC.   
*php artisan make: model.  
php artisan make:controller CreateNombreController.*

**VISTAS.**Se realizaron vistas para categorías, aplicaciones, usuarios y webside. Todas con usan una plantilla integral que se reutiliza en todas las vistas.

**MIDDLEWARE.**Se uso un middleware para la ruta Developer. Lo que hace este filtro es que si no usos usuario desarrollador no te deje ingresar a las rutas de crear, editar y borrar archivos.  
Si el usuario logueado, es distino a NULL y de user\_type 1 (que es el desarrollador) te deja ingresar a ese grupo de rutas. Si no, te devuelve un 403- Negándote el ingreso.

**AUTENTIFICACION DE USUARIOS.**Se realizó con el sistema que otorga Laravel. Se crean los modelos, los controladores y las vistas. Estas vistas se adaptaron al navbar para que se vea el contenido necesario. Si estas logueado podes ingresar a los paneles de usuario (ver tus compras, o tus apps) y poder cerrar la sesión.

**BUSCADOR.**La página cuenta con un buscador en la vista de aplicaciones. Las aplicaciones están ordenadas de forma ascendente por el nombre que tiene en la base de datos y con una paginación de 3. Para que funcione este buscador se agrego en el modelo Application una public function scopeName($query, $name) y en ControllerApplications una función search para pedir los parámetros requeridos y retornarlos.

**RETURN CONFIRM.**Se agrego para la compra y para eliminar una app, un botón de alerta para confirmar la compra a realizar y otro para confirmar la eliminación de un producto del listado del desarrollador y por ende de la base de datos.

**MUCHAS GRACIAS!**