**Manual Técnico**

**Historias de Usuario**

|  |  |
| --- | --- |
| **Como un…** | **Quiero…** |
| Css, administrador, cliente, agente de ventas, producción | Iniciar sesión |
| Administrador | Dar de alta usuarios |
| Administrador | CRUDs administrativos |
| Cliente | Hacer nuevas órdenes |
| Cliente | Visualizar información de la empresa, como correo y teléfono |
| Cliente | Visualizar detalles de mi orden |
| Administrador | Editar, crear, quitar y ver textiles |
| Administrador | Editar, crear, quitar y ver estructuras |
| Administrador | Editar precios |
| Css, administrador, cliente, agente de ventas, producción | Cerrar mi sesión |
| Cliente | Eliminar, guardar y enviar órdenes |
| Administrador | Ver, editar y eliminar ordenes |
| Cliente | Ver mis órdenes |
| Administrador | Dar de baja usuarios y/o editarlos, además de visualizarlos |
| Cliente | Ver productos en una orden |
| Cliente | Eliminar productos de una orden |
| Cliente | Ver detalles de producto al crearlo |
| Cliente | Ver el formulario de producto en una sola pagina |
| Administrador | Registro de usuarios |
| Cliente | Ver precio dinámico al cotizar |
| Administrador | Autorizar o denegar cualquier orden |
| Administrador | Visualizar ofertas y pedidos |
| Css, agente de ventas | Crear, enviar, autorizar, ver y eliminar órdenes a producción |
| Css, agente de ventas | Ver información de los usuarios |
| Administrador | Descuento fijo a usuario |
| Administrador | Correo cuando se manda a producción |
| Administrador | CRUD mecanismos |
| Administrador | CRUD tubos |
| Administrador | CRUD paneles |
| Cliente | Opciones de direccion de envio |
| Cliente | Cotizaciones correctas de cortinas |

**Tecnologías utilizadas**

* **Laravel**

Framework open source basado en PHP para el desarrollo de aplicaciones web

* **PHP**

Lenguaje para desarrollo en web capaz de mezclarse con HTML

* **HTML**

Lenguaje para desarrollo de interfaces en web

* **CSS**

Lenguaje para el diseño web

* **MySQL**

Query Languange más común.

* **Java y Selenium**

Software de pruebas automatizadas.

**Patrón de arquitectura**

El sistema web está basado en un patrón de arquitectura MVC, o bien, Modelo-Vista-Controlador. Los modelos siendo la base de datos y las relaciones, la vista es la interfaz y el controlador el backend, conectando ambas.

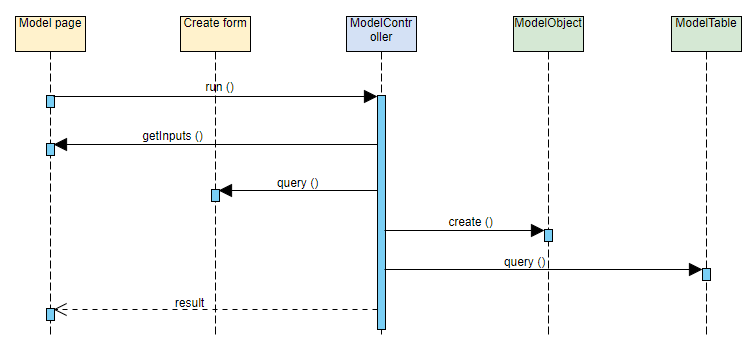
En este caso se utilizó Eloquent para las relaciones, ya que es muy práctico y fácil de utilizar.

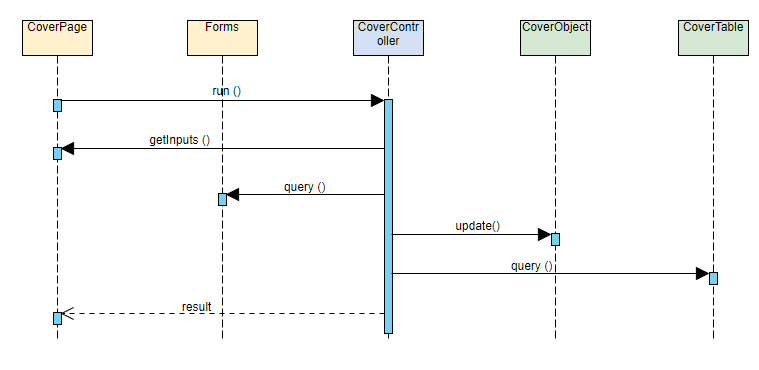
Para vistas se utilizó una tecnología llamada Blade, que facilita la integración de código PHP en HTML.

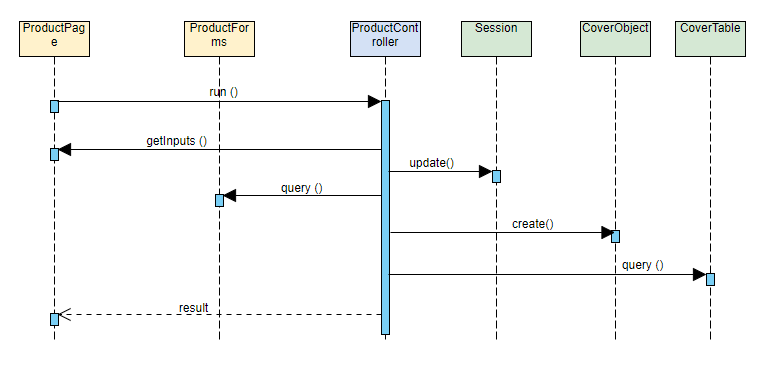
Todos los controladores estás programados enteramente en PHP.

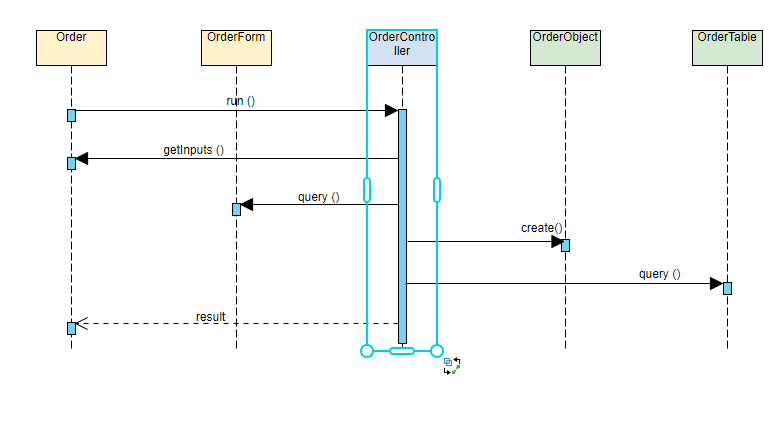


**Diagramas de secuencia**









Diagram

Description automatically generated

Diagram

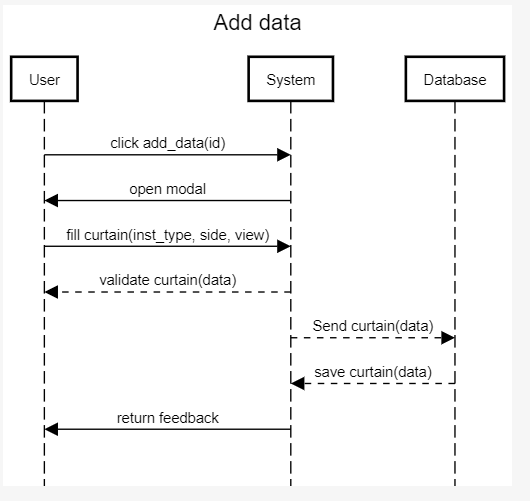
Description automatically generated

Diagram

Description automatically generated

Diagram

Description automatically generated



Diagram

Description automatically generated

Diagram

Description automatically generated

**Modelo de base de datos**

