

UNIVERSIDAD POLITÉCNICA DE PUEBLA
Ingeniería en Informática



Proyecto de Estadía Profesional

“Desarrollo de un Sistema E-Commerce”

Área temática del CONACYT: VII
Ingenierías y tecnologías

Presenta:
Héctor León Palillero

Asesor técnico
Ing. Guadalupe de Jesús Raúl Aguilera Cabrera

Asesor académico
MC. Rebeca Rodríguez Huesca

Juan C. Bonilla, Puebla, México.

18 de Diciembre de 2019

Resumen

Este documento describe el proceso que se siguió para el desarrollo de algunos de los módulos del sistema E-Commerce, desarrollado para la empresa Crystal Media. Para esto fue necesario identificar la problemática y plantear una solución, la cual consistió de elegir una metodología, en este caso fue Scrum.

Una vez seleccionada se siguieron los pasos que ésta marca, los cuales son: Planificación de la iteración (Sprint Planning), Ejecución de la iteración (Sprint) e Inspección y adaptación.

Índice

1. Introducción	6
1.1. Descripción del problema o necesidad	6
1.2 Justificación	6
1.3 Objetivo General y Específicos	7
2. Metodología y herramientas.....	8
2.1 Descripción de la metodología empleada.....	8
2.2 Herramientas	12
3. Resultados	21
3.1 Etapa 1: Planificación de la iteración (Sprint Planning).....	21
3.2 Etapa 2: Ejecución de la iteración (Sprint).....	25
4. Conclusiones y recomendaciones	49
5. Referencias bibliográficas	50

Índice de figuras

Figura 1. Metodología Scrum	9
Figura 2. Base de datos	30
Figura 3. Login Administrador	31
Figura 4. Registro de transportistas	32
Figura 5. Consulta de transportistas.....	32
Figura 6. Modificar transportistas	33
Figura 7. Desactivar transportistas.....	33
Figura 8. Registro de Marcas.....	34
Figura 9. Consulta de marcas.....	34
Figura 10. Modificar marcas	35
Figura 11. Desactivar marca	35
Figura 12. Login Cliente	36
Figura 13. Registro de clientes.....	36
Figura 14. Página de inicio de la tienda	37
Figura 15. Página de inicio de productos	37
Figura 16. Página de inicio información	38
Figura 17. Consulta de categorías y subcategorías.....	38
Figura 18. Consulta de productos.....	39
Figura 19. Consulta del producto por el cliente o visitante.....	39
Figura 20. Carrito.....	40
Figura 21. Pago	40
Figura 22. PayPal.....	41
Figura 23. Inicio de sesión del cliente.....	43
Figura 24 Inicio de sesión correctamente.....	43
Figura 25. Inicio de sesión del WebMaster	44
Figura 26. Encriptación la contraseña	44
Figura 27. Token CSRF	45
Figura 28. Ingreso de datos	45
Figura 29. Compras de productos.....	46
Figura 30. Validación de inicio de usuarios.....	46
Figura 31. Validación de campos	47

Índice de Tablas

Tabla 1. Roles	21
Tabla 2. Historias de Usuario	23
Tabla 3. Priorización de Historias de usuario.....	24
Tabla 4. Distribución de las HU	24
Tabla 5. Sprint 1.....	27
Tabla 6. Sprint 2.....	28
Tabla 7. Pruebas de compatibilidad.....	42

1. Introducción

El desarrollo de sistemas de software es un proceso necesario para crear un nuevo sistema, ya que se necesita llevar a cabo una metodología para poder estructurar y trabajar de una manera ordenada y sin confusiones.

En este capítulo se presentará la descripción del problema, la justificación y los objetivos del proyecto a realizar en la empresa Crystal Media.

1.1. Descripción del problema o necesidad

La empresa Crystal Media, es un equipo de multidisciplinarios creativos integrando una agencia de Marketing Digital que asesora en todo lo relativo a marcas y presencias en línea de los negocios: Identidad corporativa, Estrategia de Marca Diseño de páginas web, SEO, Google Awords, Social Media, Marketing de contenidos, Impresión digital y consultoría. Por lo que en el desarrollo de la creación de las tiendas en línea se ocupa el CMS E-Commerce PrestaShop, sin embargo, este E-Commerce tiene funciones que suelen alentar el desarrollo y la administración de las tiendas en línea, así como funciones que no son necesarias para la creación de las tiendas y cuando se requieren de módulos extras es necesario un pago monetario, el cual es una inversión que causa problemas para la empresa. El problema descrito anteriormente es tedioso y/o costoso debido a las funciones innecesarias, pagos de módulos extras que suelen ser ocupadas para algunas tiendas y lentos en el proceso de administración de datos. Para solucionar estos problemas se propone crear un Sistema E-Commerce a medida, el cual permitirá que el sistema se adapte a las necesidades de la empresa, optimizar el proceso de administración de los datos y que la empresa no demore tiempo y funcionalidades al estar adaptándose al Sistema.

1.2 Justificación

El comercio electrónico, conocido también como eCommerce, es el intercambio de productos o servicios usando redes computacionales, específicamente el Internet. El eCommerce tiene ramificaciones y usa tecnologías como el E-Commerce que se refiere a las compras online vía dispositivos móviles, fintech, manejo de la cadena de suministros, Internet marketing, procesamiento de transacciones, intercambio de datos electrónicos, manejo de sistemas de inventario y sistemas de recolección de datos.

El e-Commerce puede realizarse a través de una variedad de aplicaciones, como email, catálogos y carritos de compras, intercambio electrónico de datos (EDI), protocolo de transferencia de datos (FTP) y servicios web.[1]

Este proyecto tiene como fin de resolver la problemática que tiene la empresa en las etapas de desarrollo y administración de las tiendas en línea utilizando un Sistema E-Commerce, el cual tiene funciones que no se ocupan, es tedioso

el manejo y administración de los datos y así como el costo excesivo de módulos extra, por lo que para solucionarlo se desarrollará un sistema E-Commerce a medida, el cual contendrá sólo las funciones necesarias y así acelerando la creación y/o administración de dichas tiendas por la empresa; evitando el pago de módulos extra, las funciones innecesarias, la tediosa administración y el costoso desarrollo de nuevas tiendas.

1.3 Objetivo General y Específicos

Objetivo General

Desarrollar un sistema E-Commerce a medida el cual permita adaptarse a las necesidades de la empresa, en las etapas de desarrollo y administración de las tiendas online.

1.3.2 Objetivos Específicos

- Identificar los requerimientos de cada uno de los roles que participarán en el sistema.
- Diseñar la base de datos con los requerimientos identificados.
- Programar el acceso para el Webmaster y los clientes.
- Programación de los siguientes módulos:
 - Catálogo de Marcas.
 - Catálogo de transportistas.
 - Interface de productos, para que el cliente pueda visualizar la Información de los productos disponibles.
 - Interface de los productos con ofertas, para que el cliente pueda visualizar la información de los productos con ofertas disponibles.
 - Interface para consultar los productos por categoría, nombre o referencia.
 - Interface para consular la información completa de un producto.
 - Un carrito de compras, para que los clientes puedan cargar los productos a comprar.
 - Interface de orden de cobro para los diferentes tipos de clientes (cliente: Usuario con una cuenta en el sistema, Invitado: usuario sin una cuenta en el sistema).
 - Interface para la forma de pago.
 - Interface para la forma de envió
- Realizar pruebas de funcionalidad, seguridad, usabilidad y accesibilidad para verificar que el sistema cumple con los requisitos planeados inicialmente.

2. Metodología y herramientas

En este capítulo se presentará la metodología que será empleada para el desarrollo del sistema E-Commerce, así como las características, ventajas y desventajas de las herramientas tecnológicas utilizadas.

La metodología que se usará para el desarrollo del sistema E-Commerce se seleccionó de acuerdo con la forma en la que se está llevando a cabo el desarrollo del sistema.

2.1 Descripción de la metodología empleada

Metodología SCRUM

La metodología utilizada para el desarrollo del proyecto E-Commerce será la metodología Scrum, ya que es la metodología que utiliza la empresa para el desarrollo de sus proyectos.

Scrum es un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto. Estas prácticas se apoyan unas a otras y su selección tiene origen en un estudio de la manera de trabajar de equipos altamente productivos.

En Scrum se realizan entregas parciales y regulares del producto final, priorizadas por el beneficio que aportan al receptor del proyecto. Por ello, Scrum está especialmente indicado para proyectos en entornos complejos, donde se necesita obtener resultados pronto, donde los requisitos son cambiantes o poco definidos, donde la innovación, la competitividad, la flexibilidad y la productividad son fundamentales. [2]

2.1.1 Proceso

En Scrum un proyecto se ejecuta en ciclos temporales cortos y de duración fija (iteraciones que normalmente son de 2 semanas, aunque en algunos equipos son de 3 y hasta 4 semanas, límite máximo de feedback de producto real y reflexión). Cada iteración tiene que proporcionar un resultado completo, un incremento de producto final que sea susceptible de ser entregado con el mínimo esfuerzo al cliente cuando lo solicite. En la figura. 1 se muestra el funcionamiento de la metodología Scrum.

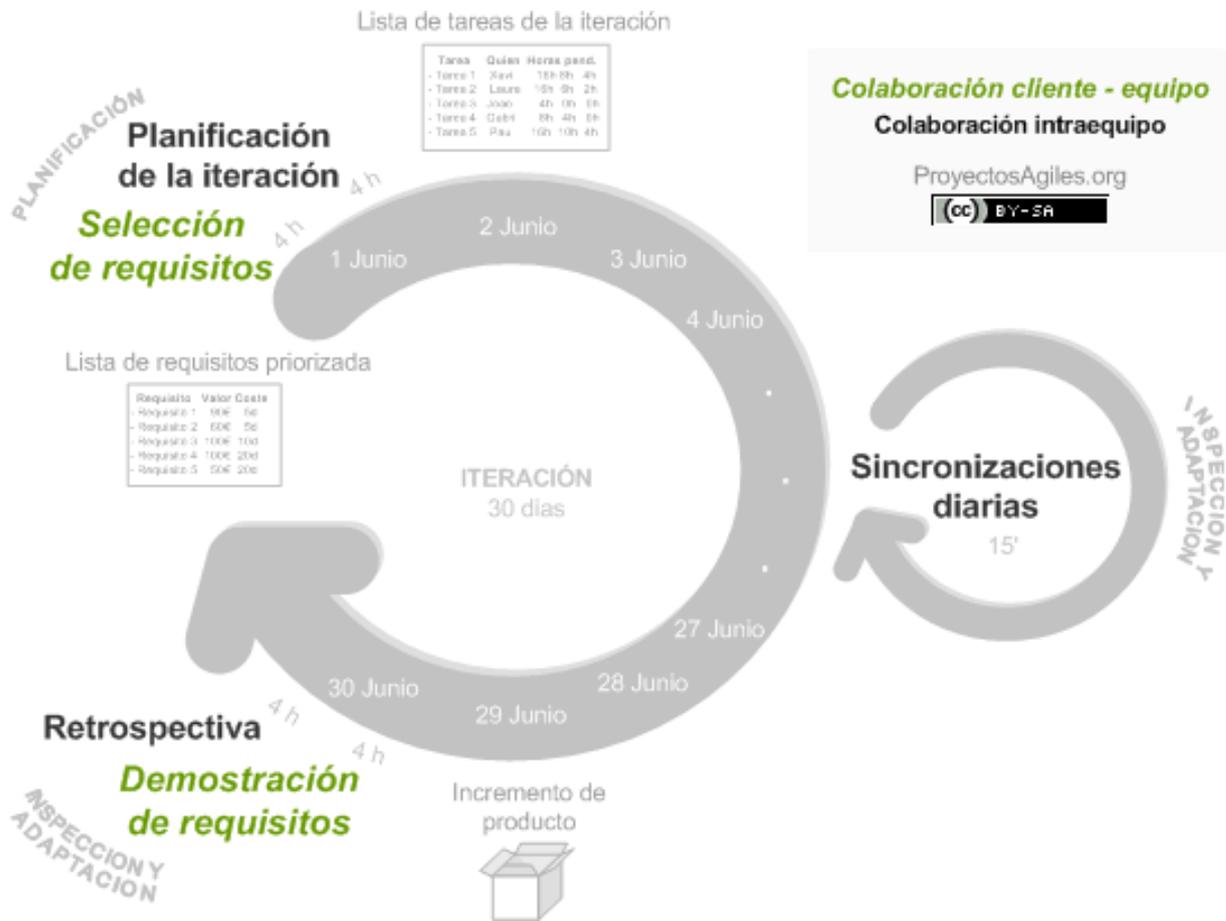


Figura 1. Metodología Scrum [2]

El proceso parte de la lista de objetivos/requisitos priorizada del producto, que actúa como plan del proyecto. En esta lista el cliente (Product Owner) prioriza los objetivos balanceando el valor que le aportan respecto a su coste (que el equipo estima considerando la Definición de Hecho) y quedan repartidos en iteraciones y entregas. [2]

2.1.2 Etapas de la metodología

1. Planificación de la iteración (Sprint Planning)

El primer día de la iteración se realiza la reunión de planificación de la iteración. Tiene dos partes: [2]

- **Selección de requisitos** (2 horas). El cliente presenta al equipo la lista de requisitos priorizada del producto o proyecto. El equipo pregunta al cliente las dudas que surgen y selecciona los requisitos más prioritarios que prevee que podrá completar en la iteración, de manera que puedan ser entregados si el cliente lo solicita.
- **Planificación de la iteración** (2 horas). El equipo elabora la lista de tareas de la iteración necesarias para desarrollar los requisitos seleccionados. La estimación de esfuerzo se hace de manera conjunta y los miembros del equipo se autoasignan las tareas, se autoorganizan

para trabajar incluso en parejas (o grupos mayores) con el fin de compartir conocimiento o para resolver juntos objetivos especialmente complejos.

2. Ejecución de la iteración (Sprint)

Cada día el equipo realiza una reunión de sincronización (15 minutos), normalmente delante de un tablero físico o pizarra (Scrum Taskboard). El equipo inspecciona el trabajo que el resto está realizando (dependencias entre tareas, progreso hacia el objetivo de la iteración, obstáculos que pueden impedir este objetivo) para poder hacer las adaptaciones necesarias que permitan cumplir con la previsión de objetivos a mostrar al final de la iteración. En la reunión cada miembro del equipo responde a tres preguntas: [2]

- ¿Qué he hecho desde la última reunión de sincronización para ayudar al equipo a cumplir su objetivo?
- ¿Qué voy a hacer a partir de este momento para ayudar al equipo a cumplir su objetivo?
- ¿Qué impedimentos tengo o voy a tener que nos impidan conseguir nuestro objetivo?

Durante la iteración el Facilitador (Scrum Master) se encarga de que el equipo pueda mantener el foco para cumplir con sus objetivos.

- Elimina los obstáculos que el equipo no puede resolver por sí mismo.
- Protege al equipo de interrupciones externas que puedan afectar el objetivo de la iteración o su productividad.

Durante la iteración, el cliente junto con el equipo refinan la lista de requisitos (para prepararlos para las siguientes iteraciones) y, si es necesario, cambian o replanifican los objetivos del proyecto (10%-15% del tiempo de la iteración) con el objetivo de maximizar la utilidad de lo que se desarrolla y el retorno de inversión.

3. Inspección y adaptación

El último día de la iteración se realiza la reunión de revisión de la iteración. Tiene dos partes: [2]

- **Revisión** (Sprint Review) (1,5 horas). El equipo presenta al cliente los requisitos completados en la iteración, en forma de incremento de producto preparado para ser entregado con el mínimo esfuerzo. En función de los resultados mostrados y de los cambios que haya habido en el contexto del proyecto, el cliente realiza las adaptaciones necesarias de manera objetiva, ya desde la primera iteración, replanificando el proyecto.
- **Retrospectiva** (Sprint Retrospective) (1,5 horas). El equipo analiza cómo ha sido su manera de trabajar y cuáles son los problemas que

podrían impedirle progresar adecuadamente, mejorando de manera continua su productividad. El Facilitador se encargará de eliminar o escalar los obstáculos identificados que estén más allá del ámbito de acción del equipo.

2.1.3 Roles en la metodología Scrum

La metodología Scrum tiene unos roles y responsabilidades principales, asignados a sus procesos de desarrollo. Estos son: [2]

- Cliente o Dueño del Proyecto. Se asegura de que el proyecto se esté desarrollando acorde con la estrategia del negocio. Escribe historias de usuario, las prioriza, y las coloca en el proyecto en desarrollo.
- Scrum master o Facilitador. Elimina los obstáculos que impiden que el equipo cumpla con su objetivo.
- Development team Member (Miembros del equipo de desarrollo). Los encargados de crear el producto para que pueda estar listo con los requerimientos necesarios.

2.1.4 Herramientas de metodología Scrum

- Lista de objetivos / requisitos priorizada (Product Backlog) [3]
La lista priorizada de objetivos/requisitos representa la visión y expectativas del cliente respecto a los objetivos y entregas del producto o proyecto
El orden de sus ítems viene determinado por el valor que aporta al cliente final respecto a riesgos y coste estimado de completarlo (ROI). Es una planificación estratégica que evoluciona a lo largo de toda la vida del producto/proyecto, debido a cambios de necesidades del cliente, feedback del mercado, aparición de nuevas ideas, dificultades tecnológicas, etc.
El cliente es el responsable de crear y gestionar la lista (con la ayuda del Facilitador y del equipo, quien proporciona el coste estimado de completar cada requisito). Dado que reflejar las expectativas del cliente, esta lista permite involucrarle en la dirección de los resultados del producto o proyecto
- Lista de tareas de la iteración (Sprint Backlog) [4]
Subconjunto de objetivos/requisitos del Product Backlog seleccionado para la iteración actual y su plan de tareas de desarrollo. El equipo lo elabora en la reunión de planificación de la iteración (Sprint planning) seleccionando lo que prevé que podrá completar y demostrar al cliente al finalizar la iteración, en forma de incremento de producto preparado para ser entregado.

- Gráficos de trabajo pendiente (Burndown charts) [5]
Un gráfico de trabajo pendiente a lo largo del tiempo muestra la velocidad a la que se está completando los objetivos/requisitos. Permite extrapolar si el Equipo podrá completar el trabajo en el tiempo estimado.

Se pueden utilizar los siguientes gráficos de esfuerzo pendiente:

Días pendientes para completar los requisitos del producto o proyecto (product burndown chart), realizado a partir de la lista de requisitos priorizada (Product Backlog).

Horas pendientes para completar las tareas de la iteración (sprint burndown chart), realizado a partir de la lista de tareas de la iteración (Iteration Backlog).

2.2 Herramientas

Las herramientas para llevar a cabo el desarrollo del sistema E-Commerce se explicarán a continuación:

2.2.1 Visual Studio Code

Visual Studio Code. Es un editor de código fuente desarrollado por Microsoft para Windows, Linux y macOS. Incluye soporte para depuración, control de Git integrado, resaltado de sintaxis, finalización de código inteligente, fragmentos de código y refactorización de código. También es personalizable, de modo que los usuarios pueden cambiar el tema del editor, los métodos abreviados de teclado y las preferencias. Es gratuito y de código abierto.[3]

Ventajas

- Libre.
- Fantástico Debugging.
- Preparado. Lo básico ya viene preestablecido.
- Ideal para Javascript. Está realizado en Typescript.
- Gran variedad de plugins.
- Fácil al instalar plugins, configurar y cambiar tema.
- Git integrado.[4]

Desventajas

- Pesado. Su base es Electron (Chrome). Prácticamente es como tener un navegador abierto.
- No tan rápido. Tiene un ligero retraso cuando se abren archivos, se mueven con el scroll o se editan ficheros grandes.
- No funciona en terminal, aunque sí contiene un terminal.[4]

2.2.2 Apache HTTP Server

Apache es un software de servidor web gratuito y de código abierto con el cual se ejecutan el 46% de los sitios web de todo el mundo. El nombre oficial es

Apache HTTP Server, y es mantenido y desarrollado por la Apache Software Foundation.

Le permite a los propietarios de sitios web servir contenido en la web, de ahí el nombre de “servidor web”. Es uno de los servidores web más antiguos y confiables, con la primera versión lanzada hace más de 20 años, en 1995.[5]

Ventajas

- De código abierto y gratuito, incluso para uso comercial.
- Software confiable y estable.
- Parches de seguridad regulares y actualizados con frecuencia.
- Flexible debido a su estructura basada en módulos.
- Fácil de configurar para principiantes.
- Multiplataforma (funciona tanto en servidores Unix como en Windows).
- Viene listo para trabajar con sitios de WordPress.
- Enorme comunidad y soporte fácilmente disponible en caso de cualquier problema.[5]

Desventajas

- Problemas de rendimiento en sitios web con demasiado tráfico.
- Demasiadas opciones de configuración pueden generar vulnerabilidades de seguridad.[5]

2.2.3 PHP

PHP es un lenguaje de código abierto muy popular, adecuado para desarrollo web y que puede ser incrustado en HTML. Es popular porque un gran número de páginas y portales web están creadas con PHP. Código abierto significa que es de uso libre y gratuito para todos los programadores que quieran usarlo. Incrustado en HTML significa que en un mismo archivo vamos a poder combinar código PHP con código HTML, siguiendo unas reglas.[6]

Ventajas

- Es un lenguaje multiplataforma.
- Completamente orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una Base de Datos.
- El código fuente escrito en PHP es invisible al navegador y al cliente ya que es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML al navegador. Esto hace que la programación en PHP sea segura y confiable.
- Capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y PostgreSQL.
- Capacidad de expandir su potencial utilizando la enorme cantidad de módulos (llamados ext's o extensiones).[7]

Desventajas

- un inconveniente que el código fuente no pueda ser ocultado. La ofuscación es una técnica que puede dificultar la lectura del código, pero no la impide y, en ciertos casos, representa un costo en tiempos de ejecución
- Debes saber cuándo menos HTML para poder hacer un trabajo medianamente funcional
- Si no lo configuras correctamente dejas abiertas muchas brechas de seguridad
- Se necesita instalar un servidor web.[7]

2.2.4 Laravel

Laravel es uno de los frameworks de código abierto más fáciles de asimilar para PHP. Es simple, muy potente y tiene una interfaz elegante y divertida de usar. Fue creado en 2011 y tiene una gran influencia de frameworks como Ruby on Rails, Sinatra y ASP.NET MVC.[8]

Ventajas

- Reducción de costos y tiempos en el desarrollo y mantenimiento
- La documentación existente es sencilla y muy completa
- Es de código abierto
- Facilita el manejo de ruteo de una aplicación [9]

Desventajas

- Utilizar la consola para realizar funciones
- La instalación se realiza a través de composer
- Está pensado solo para las últimas versiones de PHP a partir de la versión 5.4 [10]

2.2.5 Composer

Composer es un gestor de dependencias en proyectos, para programación en PHP. Eso quiere decir que nos permite gestionar (declarar, descargar y mantener actualizados) los paquetes de software en los que se basa nuestro proyecto PHP. Se ha convertido en una herramienta de cabecera para cualquier desarrollador en este lenguaje que aprecie su tiempo y el desarrollo ágil.[10]

Ventajas

- Permite indicar exactamente la librería que necesitas y su versión
- Fomenta el uso de estándares
- Puedes utilizar paquetes PEAR

- Permite también el uso de repositorios privados
- Permite la definición de procedimientos o llamadas a funciones que se ejecutarán después de
- realizar ciertas operaciones como actualizaciones, borrados, instalaciones[10]

Desventajas

- Debes descargar Composer para cada proyecto que crees, lo cual no es muy productivo y te hará malgastar espacio en el disco duro.
- Como Composer se actualiza muy frecuentemente, mantenerlo actualizado en todos tus proyectos será casi imposible y te costará mucho tiempo.[11]

2.2.6 SQL Power Architect

SQL Power Architect es una herramienta de modelado de datos creados por los diseñadores de almacenamiento de datos y tiene muchas características únicas, dirigidas específicamente para el arquitecto de almacenamiento de datos.

Permite a los usuarios realizar ingeniería inversa de base de datos existentes, realizar datos de perfiles en base de datos fuente y auto-generar metadatos ETL. [12]

Ventajas:

- Acceso a datos vía JDBC fuente.
- Se puede conectar múltiples fuentes de datos al mismo tiempo.
- Compara datos de modelos y estructuras de base de datos e identifica las discrepancias.
- Genera diccionarios y scripts SQL para crear las bases de datos en múltiples lenguajes.
- Soporte para dominios de datos y tipos (Enterprise Edition).
- Multiplataforma, incluye una versión en java.
- Código abierto.
- Posee diferentes tipos de datos. [12]

Desventajas:

- No existe una versión a la medida en Linux, si no que ocupa la versión creada en java para correrla en máquina virtual. [12]

2.2.7 HTML

HTML significa "Lenguaje de Marcado de Hipertexto" por sus siglas en inglés "HyperText Markup Language", es un lenguaje que pertenece a la familia de los

"lenguajes de marcado" y es utilizado para la elaboración de páginas web. El estándar HTML lo define la W3C (World Wide Web Consortium) y actualmente HTML se encuentra en su versión HTML5.

Cabe destacar que HTML no es un lenguaje de programación ya que no cuenta con funciones aritméticas, variables o estructuras de control propias de los lenguajes de programación, por lo que HTML genera únicamente páginas web estáticas, sin embargo, HTML se puede usar en conjunto con diversos lenguajes de programación para la creación de páginas web dinámicas.

Básicamente el lenguaje HTML sirve para describir la estructura básica de una página y organizar la forma en que se mostrará su contenido, además de que HTML permite incluir enlaces (links) hacia otras páginas o documentos.

HTML es un lenguaje de marcado descriptivo que se escribe en forma de etiquetas para definir la estructura de una página web y su contenido como texto, imágenes, entre otros, de modo que HTML es el encargado de describir (hasta cierto punto) la apariencia que tendrá la página web. [13]

Ventajas:

- Es un lenguaje sencillo, fácil de aprender y fácil de leer e interpretar.
- Existen numerosas aplicaciones y editores de páginas web (WYSIWYG) que generan el código automáticamente, por lo que no es necesario ser un experto informático para hacer páginas basadas en HTML.
- Su sencillez hace que pueda diseñarse y desplegarse un sitio web en muy poco tiempo.
- Es el lenguaje más extendido, todos los navegadores lo admiten.
- Código visible e interpretable por los buscadores. [14]

Desventajas:

- Es un lenguaje estático que no permite manejar bases de datos.
- Las páginas pueden variar su aspecto dependiendo del navegador (es un lenguaje interpretado).
- Su funcionalidad es limitada, hay muchas cosas que no se pueden hacer solo con HTML.
- El diseño de las páginas también está limitado, aunque con HTML5 ha mejorado bastante. [14]

2.2.8 MySQL

MySQL es un sistema de gestión de bases de datos relacional, fue creada por la empresa sueca MySQL AB, la cual tiene el copyright del código fuente del servidor SQL, así como también de la marca.

MySQL es un software de código abierto, licenciado bajo la GPL de la GNU, aunque MySQL AB distribuye una versión comercial, en lo único que se diferencia de la versión libre, es en el soporte técnico que se ofrece, y la posibilidad de integrar este gestor en un software propietario, ya que de otra manera, se vulneraría la licencia GPL. [15]

Características principales

Inicialmente, MySQL carecía de algunos elementos esenciales en las bases de datos relacionales, tales como integridad referencial y transacciones. A pesar de esto, atrajo a los desarrolladores de páginas web con contenido dinámico, debido a su simplicidad, de tal manera que los elementos faltantes fueron complementados por la vía de las aplicaciones que la utilizan. Poco a poco estos elementos faltantes, están siendo incorporados tanto por desarrolladores internos, como por desarrolladores de software libre.

En las últimas versiones se pueden destacar las siguientes características principales:

- El principal objetivo de MySQL es velocidad y robustez.
- Soporta gran cantidad de tipos de datos para las columnas.
- Gran portabilidad entre sistemas, puede trabajar en distintos sistemas operativos.
- Cada base de datos cuenta con 3 archivos: Uno de estructura, uno de datos y uno de índice y soporta hasta 32 índices por tabla.
- Aprovecha la potencia de sistemas multiproceso, gracias a su implementación multihilo.
- Flexible sistema de contraseñas (passwords) y gestión de usuarios, con un muy buen nivel de seguridad en los datos.
- El servidor soporta mensajes de error en distintas lenguas. [15]

Ventajas:

- Velocidad al realizar las operaciones, lo que le hace uno de los gestores con mejor rendimiento.
- Bajo costo en requerimientos para la elaboración de bases de datos, ya que debido a su bajo consumo puede ser ejecutado en una máquina con escasos recursos sin ningún problema.
- Facilidad de configuración e instalación.
- Soporta gran variedad de Sistemas Operativos.
- Baja probabilidad de corromper datos, incluso si los errores no se producen en el propio gestor, sino en el sistema en el que está.
- Conectividad y seguridad. [9]

Desventajas:

- Un gran porcentaje de las utilidades de MySQL no están documentadas.
- No es intuitivo, como otros programas (ACCESS). [15]

2.2.9 GitHub

Git es uno de los sistemas de control de versiones más populares entre los desarrolladores. Y parte culpa de su popularidad la tiene GitHub, un excelente servicio de alojamiento de repositorios de software con este sistema, que lejos de quedarse en esta funcionalidad, ofrece hoy en día un conjunto de características muy útiles para el trabajo en equipo. [16]

Características

Pero hoy en día GitHub es mucho más que un servicio de alojamiento de código. Además de éste, se ofrecen varias pequeñas herramientas en línea muy útiles para el trabajo en equipo. Entre ellas, cabe destacar: [16]

- Un wiki que funciona con gollum, el cual opera con Git para el mantenimiento de las distintas versiones de las páginas.
- Un sistema de seguimiento de problemas, que al estilo del clásico sistema de tickets, permiten a los miembros de tu equipo (o a cualquier usuario de GitHub si tu repositorio es público) abrir un ticket detallando un problema que tenga con tu software o una sugerencia que desee hacer al mismo.
- Una herramienta de revisión de código, donde se pueden añadir anotaciones en cualquier punto de un fichero (ej.: "Esto es mejor que lo extraigamos a una nueva clase"), y debatir sobre determinados cambios realizados en un commit específico.
- Un visor de ramas donde se pueden comparar los progresos realizados en las distintas ramas de nuestro repositorio.

Ventajas

- **Compartición selectiva:** El desarrollo de la aplicación serán únicamente nuestro, pudiendo decidir qué parte de nuestro proyecto compartimos y con quién, restringiendo a que sólo pueda verlo, que tenga la posibilidad de añadir notas, comentarios o que pueda añadir cambios. No todo tiene porque ser público (aunque en la compartición encontramos la riqueza y agilidad que fundamenta el software libre ;).
- **Velocidad:** Muchas empresas deciden implementar GIT como servicio local en su infraestructura física, por lo tanto el control de versiones se realizaría dentro de la propia red con la consiguiente ganancia en velocidad de acceso y escritura, así como eliminando el requisito de contar con una conexión a internet obligatoria. No obstante (como ya se verá en el curso de Git, GitHub y GitLab, contar con el respaldo de un hosting para estos menesteres siempre es un plus de seguridad pues ganaremos la despreocupación para con respecto de la seguridad y accesibilidad de nuestro trabajo.
- **Ramificación:** Ya hablamos de la ramificación en una entrada anterior, y como vimos ofrece un amplio abanico de posibilidades a la hora de

realizar cambios en la estructura principal, pudiendo crear diferentes ramas sobre las que aplicar nuestras modificaciones en entornos aislados de la línea principal de desarrollo.

- **Convergencia:** Si en la creación de una rama del proyecto encontramos que uno de los cambios incluidos se integra tal y como deseamos, sin presentar conflictos con las diferentes partes de nuestra aplicación, podremos incluir o hacer converger dicha ramificación con el desarrollo principal de forma sencilla y segura, contando así con una nueva versión o revisión de nuestro proyecto lista para ser distribuida, compartida, liberada.
- **Sandbox:** Esta sería una ventaja de una ventaja, ya que las ramificaciones nos preparan un entorno aislado de pruebas sobre el desarrollo de la línea central de nuestra App. Los cambios realizados en una de las ramas del proyecto no tendrán consecuencias para los usuarios que actualmente usen o accedan a la versión sin modificar o principal. Muy útil si lo que vamos a hacer es incluir servicios que antes no existían en nuestra aplicación y esto pudiese poner en peligro la estabilidad de otros componentes.
- **Flujo de trabajo adaptable:** En el sector de los controladores de versiones encontraremos diferentes formas para gestionar el flujo de desarrollo de la aplicación, destacando entre éstos los modelos centralizados y los modelos de libre configuración. Con esto encontramos que trabajemos como trabajemos encontraremos un control de versiones que se adapte a nosotros o nuestra empresa, haciendo uso desde un simple sistema jerárquico hasta un ligeramente más complejo sistema centralizado. En este apartado Git gana puntos sobre la competencia, admitiendo multitud de configuraciones que nos permitirán dentro de su estructura organizar el trabajo tal y como deseemos nosotros, nuestro equipo de desarrollo, etc...
- **Seguridad:** Pero... ¿y si tengo un sistema de control de versiones instalado en local y tengo una caída de la tensión eléctrica que provoca el apagado de la infraestructura? ¿Los datos se habrán corrompido en el proceso de escritura? La respuesta es que resultaría muy complejo, ya que en su inmensa mayoría, los controladores de versiones cuentan con sistemas de cifrado y otros tipos de medidas de seguridad que se aplicarán para que nuestros datos permanezcan lo más íntegros posible. Por poner un ejemplo, Git hace uso de sistemas de árbol SHA1, lo que asegurará que hasta que no se realice la comprobación del cifrado o firma, los cambios no se escribirán en el servidor.
- **Coste:** Obviamente podremos encontrar software de control de versiones que nos ofrezcan las mismas o similares características que nos otorga Git o BitBucket entre otros, pero lo que será complejo es que alguna de estas alternativas sea gratuita. Los sistemas de hosting pueden llegar a presentar algún coste, pero entrará en nuestro juicio contemplar si realizar una pequeña inversión en infraestructura online

(asegurando ya de paso un poco más el acceso y seguridad de nuestro proyecto) o adquirir una costosa solución similar a los ejemplos ya citados. Poniéndonos en la piel de un empresario, creo que cuanto más consigamos ahorrar a la empresa mejor para la empresa (y para nosotros demostrando preocupación por los recursos económicos de la misma. [17]

Desventaja

Afortunadamente, la desventaja presentada tiene fácil solución con el curso de Git, GitHub y GitLab guiados por profesionales con amplia experiencia en cada rama de enseñanza, siempre orientada a la formación de cara al mercado laboral actual.

Aprendizaje: Es algo por lo que tendremos que pasar cada vez que queramos incluir alguna nueva tecnología en un flujo de trabajo ya establecido. Deberemos formar al equipo de desarrollo o a aquellas personas a las que tendremos que dar acceso a nuestro trabajo en el servicio de control de versiones; para evitar así posibles errores en la realización de cambios [17]

3. Resultados

En este capítulo se muestran los resultados obtenidos durante el desarrollo del proyecto de estadía, los resultados serán mostrados conforme a las etapas de la metodología Scrum.

3.1 Etapa 1: Planificación de la iteración (Sprint Planning)

En esta primera etapa se realizó la asignación de tareas del equipo de trabajo después de cada reunión con el dueño del proyecto, en este caso la asignación de roles de cada integrante del equipo de desarrollo se muestra en la tabla 1.

Persona	Rol
Lic. Francisco Ibo	Product owner
Ing. Guadalupe de Jesús Raúl Aguilera Cabrera	Scrum master
Héctor León Palillero	Team
Rey David Juárez Mejía	

Tabla 1. Roles

El rol que tomó cada uno de los integrantes del proyecto fue designado por el asesor del proyecto, así como también describió la problemática planteada para el desarrollo del proyecto.

Desde el inicio del desarrollo del proyecto hasta la fecha se han tenido 2 reuniones con el dueño del proyecto, una se realizó el 17 de septiembre de 2019 la otra el 21 de octubre, en éstas se establecieron los requisitos para el desarrollo del proyecto, los cuales se listan en la tabla 1 como historias de usuarios

Historias de Usuario	
HU1-Login	Como: Administrador Quiero: Ingresar al sistema del administrador Para: Poder realizar las configuraciones pertinentes para el sitio web Condiciones: Contar con un usuario y contraseña, contar con internet
HU2-Inicio	Como: Cliente/visitante Quiero: ver los que ofrece el sitio web Para: Saber que comprar Condiciones Contar con internet
HU3-Registrarse	Como: Cliente/visitante Quiero: Ingresar en la tienda Para: Paga agilizar el método de pago, así como obtener los beneficios de ser cliente

	Condiciones: Ingresar la información obligatoria y de forma correcta
HU4-Login Cliente	Como: Cliente Quiero: Ingresar a la tienda Para: Poder realizar compras Condiciones: Contar con un usuario y contraseña, contar con internet
HU5-Agregar Transportista	Como: Administrador Quiero: Insertar transportista Para: Poder Registrar su información de un transportista nuevo Condiciones: Ingresar los datos obligatorios, contar con internet
HU6-ConsultarTransportista	Como: Administrador Quiero: Buscar transportistas Para: Poder Consultar los transportistas y su información Condiciones: Tener registros, contar con internet
HU7-ModificarTransportista	Como: Administrador Quiero: Editar transportista Para: Poder Modificar su información de los transportistas. Condiciones: Contar con transportistas, contar con internet
HU8-Desactivar Transportista	Como: Administrador Quiero: Desactivar transportista Para: Poder desactivar transportes que ya no se utilicen Condiciones: Contar con transportistas, contar con internet
HU9-Agregar Marca	Como: Administrador Quiero: Insertar marca Para: Poder Registrar su información de una marca nueva Condiciones: Ingresar los datos obligatorios, contar con internet
HU10-Consultar Marca	Como: Administrador Quiero: Buscar marcas Para: Poder Consultar las marcas y su información Condiciones: Tener registros, contar con internet
HU11-Modificar Marca	Como: Administrador Quiero: Editar marca Para: Poder Modificar su información de las marcas Condiciones: Contar con marcas, contar con internet
HU12-Desactivar Marca	Historia: Desactivar marca

	<p>Como: Administrador Quiero: Desactivar marca Para: Poder Desactivar marcas que ya no se utilicen Condiciones: Contar con marcas, contar con internet</p>
HU13-Consultar Categorías y subcategorías	<p>Como: Cliente / Visitante Quiero: Quiero consultar las categorías y subcategorías de la tienda Para: Consultar las categorías que se encuentran disponibles y así poder buscar los productos Condiciones: Contar con categorías disponibles, contar con internet</p>
HU14-Consusltar Productos	<p>Como: Cliente / Visitante Quiero: Quiero consultar los Productos por categoría Para: Consultar los productos que se encuentran disponibles y así poder realizar compras Condiciones: Contar con productos disponibles, contar con internet</p>
HU15-Carrito	<p>Como: Cliente/visitante Quiero: ver mis artículos comprados Para: Ver la información detallada de los productos comprados y el total de compra Condiciones: Tener productos comprados, contar con internet</p>
HU16-Pago	<p>Como: Cliente/Visitante Quiero: Seleccionar método de pago Para: Realizar un pago de acuerdo a la disponibilidad Condiciones: Tener métodos de pago, contar con internet</p>

Tabla 2. Historias de Usuario

3.1.1 Selección de requisitos

A continuación, se muestra en la Tabla 3 la prioridad de las HU que se llevaron a cabo en el desarrollo del E-commerce, para realizar en cada sprint.

Prioridad Alta	HU1-Login HU2-Inicio HU3-Registrarse HU4-Login Cliente HU5-Agregar Transportista HU9-Agregar Marca HU14-Consusltar Productos HU15-Carrito HU16-Pago
Prioridad Mediana	HU7-ModificarTransportista HU8-Desactivar Transportista HU11-Modificar Marca HU12-Desactivar Marca HU13-Consultar Categorías y subcategorías
Prioridad Baja	HU6-ConsultarTransportista HU10-Consultar Marca

Tabla 3. Priorización de Historias de usuario

3.1.2 Planificación de la iteración

A continuación, se muestra en la tabla 3 la distribución de las Historias de Usuario realizadas en cada Sprint.

Sprint 1	Sprint 2
HU1-Login HU2-Inicio HU3-Registrarse HU4-Login Cliente HU5-Agregar Transportista HU6-ConsultarTransportista	HU7-ModificarTransportista HU8-Desactivar Transportista HU9-Agregar Marca HU10-Consultar Marca HU11-Modificar Marca HU12-Desactivar Marca HU13-Consultar Categorías y subcategorías HU14-Consusltar Productos HU15-Carrito HU16-Pago

Tabla 4. Distribución de las HU

3.2 Etapa 2: Ejecución de la iteración (Sprint)

En esta etapa se llevaron a cabo reuniones entre los integrantes del equipo y el scrum master diariamente, aproximadamente cada reunión tuvo una duración de 10 minutos, en las cuales se llevaban las siguientes actividades:

- Cada integrante le explicaba al scrum master los avances que tuvo durante el día, así como los problemas que impedían avanzar con lo que se estaba realizando.
- Entre los integrantes del equipo y el scrum master se buscaba dar solución a los problemas que se establecieron durante la reunión.
- El scrum master aclaraba las dudas que se tenían desacuerdo al requerimiento a desarrollar.

A continuación, en la tabla 4 y 5 se describe brevemente lo que pasaba en las reuniones diarias.

Sprint 1					
Día		¿Qué se hizo el día de ayer?	¿Qué se realizara el día de hoy?	Obstáculos	Responsable
S E P T I E M B R E	17	Reunión con el dueño del proyecto	Llevar acabo el análisis de los requerimientos	Ninguno	Equipo de Desarrolladores
	18	Se hizo el análisis de los requerimientos	Seleccionar los requisitos más prioritarios	Ninguno	Equipo de Desarrolladores
	19	Seleccionar los requisitos más prioritarios	Asignar tareas que llevara a cabo cada integrante	Ninguno	Equipo de Desarrolladores
	20	Asignar tareas que llevara a cabo cada integrante	Analizar los requerimientos de las tareas	Ninguno	Equipo de Desarrolladores
	21	Analizar los requerimientos de las tareas	Priorización de los requerimientos	Problemas en la creación de las lista de Priorización	Equipo de Desarrolladores
	23	Priorización de los requerimientos	Priorización de los requerimientos	Ninguno	Equipo de Desarrolladores
	24	Priorización de los requerimientos	Se hizo el análisis de la base de datos	Ninguno	Equipo de Desarrolladores
	25	Se hizo el análisis de la base de datos	Se hizo el análisis de la base de datos	Ninguno	Equipo de Desarrolladores
	26	Se hizo el análisis de la base de datos	Llevar acabo el análisis de la base de datos	Ninguno	Equipo de Desarrolladores
	27	Llevar acabo la creación de la base	Llevar acabo la creación de la	Problemas en la	Equipo de Desarrolladores

		de datos	base de datos	creación de la base de datos	
	28	Llevar acabo la creación de la base de datos	Llevar acabo la creación de la base de datos	Ninguno	Equipo de Desarrolladores
	30	Llevar acabo la creación de la base de datos	Llevar acabo la creación de la base de datos	Ninguno	Equipo de Desarrolladores
O C T U B R E	1	Se creó el diseño de inicio de sesión del administrador	Se creó el diseño de inicio de sesión del administrador	Problemas con la vista	Héctor León Palillero
	2	Se creó el diseño de inicio de sesión del administrador	Se creó el diseño de inicio de sesión del administrador	Ninguno	Héctor León Palillero
	3	Crear login del administrador	Crear login del administrador	Problemas con la conexión a la base de datos	Héctor León Palillero
	4	Crear login del administrador	Crear login del administrador	Problemas con la conexión a la base de datos	Héctor León Palillero
	5	Crear login del administrador	Crear login del administrador	Problemas con la autenticación	Héctor León Palillero
	7	Crear login del administrador	Crear login del administrador	Problemas con la autenticación	Héctor León Palillero
	8	Crear login del administrador	Crear login del administrador	Problemas con la autenticación	Héctor León Palillero
	9	Crear login del administrador	Crear login del administrador	Problemas con la encryptacion de la contraseña	Héctor León Palillero
	10	Crear login del administrador	Crear login del administrador	Problemas con la encryptacion de la contraseña	Héctor León Palillero
	11	Crear login del administrador	Pruebas para acceder como administrador	Ninguno	Héctor León Palillero
	12	Crear vista para el registro de transportistas	Crear vista para el registro de transportistas	Problemas con la ruta	Héctor León Palillero
13	Crear vista para el	Crear vista para el	Ninguno	Héctor León	

		registro de transportistas	registro de transportistas		Palillero
14		Registro de transportistas en la base de datos	Registro de transportistas en la base de datos	Problemas con la base de datos	Héctor León Palillero
15		Registro de transportistas en la base de datos	Registro de transportistas en la base de datos	Problemas con la base de datos	Héctor León Palillero
16		Registro de transportistas en la base de datos	Registro de transportistas en la base de datos	Ninguno	Héctor León Palillero
18		Crear la vista para la consulta de transportistas	Crear la vista para la consulta de transportistas	Ninguno	Héctor León Palillero
19		Crear la vista para la consulta de transportistas	Crear la vista para la consulta de transportistas	Ninguno	Héctor León Palillero

Tabla 5. Sprint 1

Sprint 2					
Día		¿Qué se hizo el día de ayer?	¿Qué se realizara el día de hoy?	Obstáculos	Responsable
O C T U B R E	21	Reunión y Creación de método de búsqueda de transportistas en la vista de consulta	Segunda reunión con el dueño del proyecto y Creación del método de búsqueda de transportistas en la vista de consulta	Ninguno	Héctor León Palillero
	22	Se creó botón modificar transportistas	Se creó botón modificar, una ruta y un controlador	Ninguno	Héctor León Palillero
	23	Se creó una vista para modificar los transportistas	Crear una vista para modificar los transportistas	Problemas con la obtención de los datos desde la base de datos	Héctor León Palillero
	25	Se creó una vista para modificar los transportistas	Crear una vista para modificar los transportistas y modificarlos en la base de datos	Ninguno	Héctor León Palillero
	26	Se creó botón desactivar transportistas	Se creó botón desactivar, una ruta y un js	Problemas con la función java scrip	Héctor León Palillero
	27	Se creó botón	Se creó botón	Ninguno	Héctor León

		desactivar transportistas	modificar, una ruta y un js		Palillero
	28	Crear vista para el registro de marcas	Crear vista para el registro de transportistas	Ninguno	Héctor León Palillero
	29	Registro de marcas en la base de datos	Registro de marcas en la base de datos	Ninguno	Héctor León Palillero
	30	Se creó botón modificar marcas	Se creó botón modificar, una ruta y un controlador	Ninguno	Héctor León Palillero
	31	Se creó una vista para modificar los transportistas y un botón desactivar marcas	Crear una vista para modificar las marcas, modificarlos y desactivarlos en la base de datos	Ninguno	Héctor León Palillero
N O V I E M B R E	4	Crear la vista para la consulta de marcas	Crear la vista para la consulta de marcas y un método de búsqueda	Ninguno	Héctor León Palillero
	5	Crear login del cliente	Crear una vista y login del cliente	Ninguno	Héctor León Palillero
	6	Crear Registro del cliente	Crear una vista y registro del cliente en la base de datos	Ninguno	Héctor León Palillero
	7	Crear una consulta de categorías en la vista de inicio de cliente	Crear una consulta de categorías en la vista de inicio de cliente	Ninguno	Héctor León Palillero
	8	Crear vista de consulta de productos por parte del cliente	Crear vista de consulta de productos por parte del cliente	Problemas con las pestañas de categorías	Héctor León Palillero
	9	Crear vista de consulta de productos por parte del cliente	Crear vista de consulta de productos por parte del cliente	Ninguno	Héctor León Palillero
	11	Crear carrito de compras	Crear carrito de compras	Problemas con variables de sesión	Héctor León Palillero
	12	Crear carrito de compras	Crear carrito de compras	Problemas con variables de sesión	Héctor León Palillero
	13	Crear método de pago	Crear el método de pago	ninguno	Héctor León Palillero

Tabla 6. Sprint 2

3.3 Etapa 3: Inspección y adaptación

3.3.1 Revisión

En esta etapa se mostraba al scrum master los avances de los requerimientos ya programados, se verificaba que:

- Cada sección programada cumpliera el objetivo del requerimiento original.
- Cada sección programada fuera entendible para el usuario que lo usará.

Si algún requerimiento no fue programado conforme a su objetivo, entre el scrum master y los integrantes del equipo se replanteaba el requerimiento para entender objetivo principal y así tener una idea más clara de lo que se quiere visualizar en la interfaz.

3.3.1.1 Resultados del primer Sprint

1.- Diseño Base de datos

Para el funcionamiento del E-commerce es necesario agregar 19 tablas las cuales son:

Lista de tablas

- [categorías](#)
- [cliente](#)
- [comentarios](#)
- [compras](#)
- [descuentos](#)
- [det_pedido](#)
- [devoluciones](#)
- [direcciones](#)
- [empresa](#)
- [marcas](#)
- [pedidos](#)
- [producto](#)
- [rol](#)
- [roles_categoria](#)
- [transporte_rol](#)
- [transportes_producto](#)
- [transportistas](#)
- [usuarios](#)
- [ventas](#)

En la figura 2 se muestra el modelo de las tablas de base de datos.

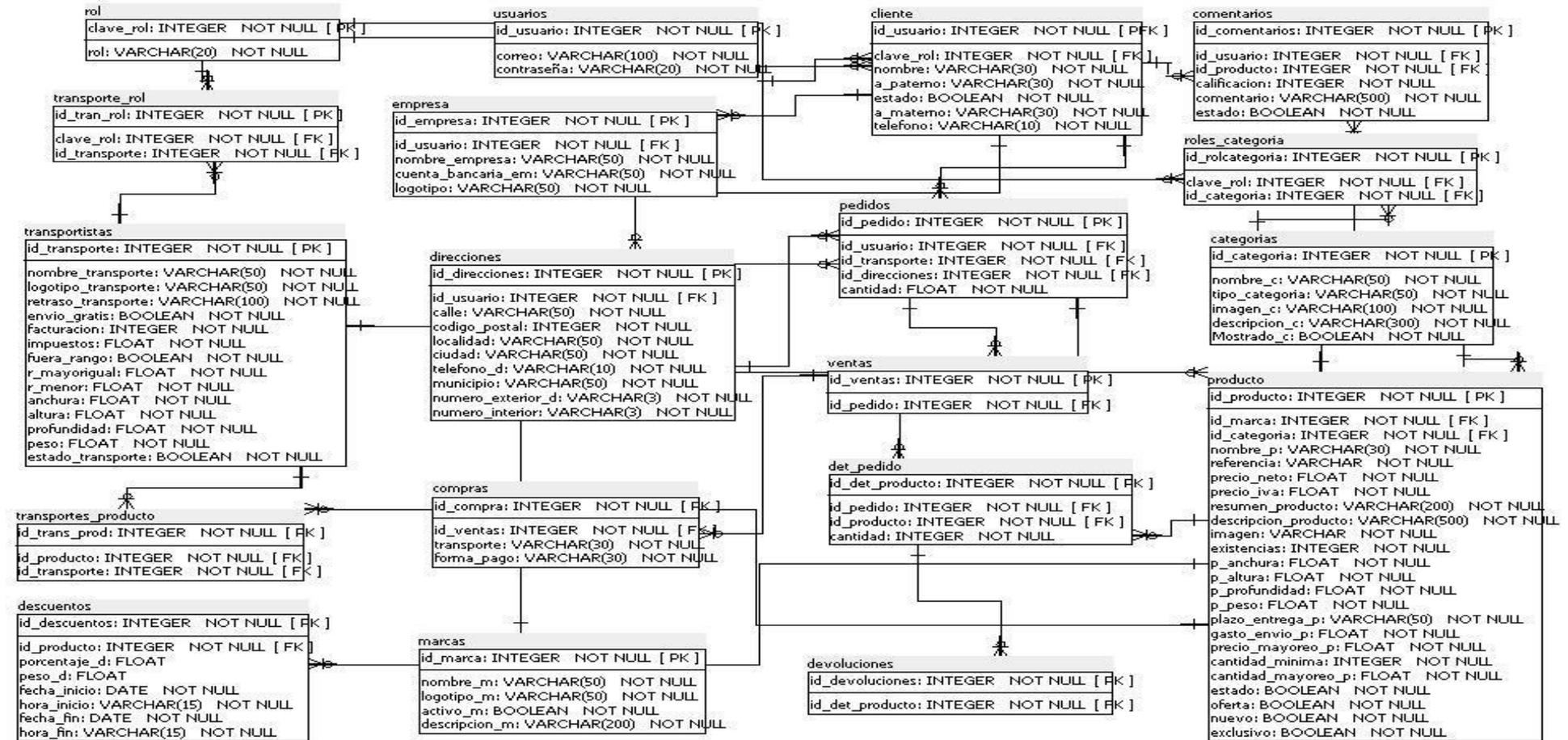


Figura 2. Base de datos

2- Diseño de las vistas

Se muestran las vistas realizadas en el primer Sprint

Login Administrador

Se creó un login para administrador para redimirlo al panel del administrador, sólo este rol tendrá acceso a dicho panel.

Se muestran dos campos de entrada los cuales son correo electrónico y su contraseña. (Figura. 3 Login Administrador)

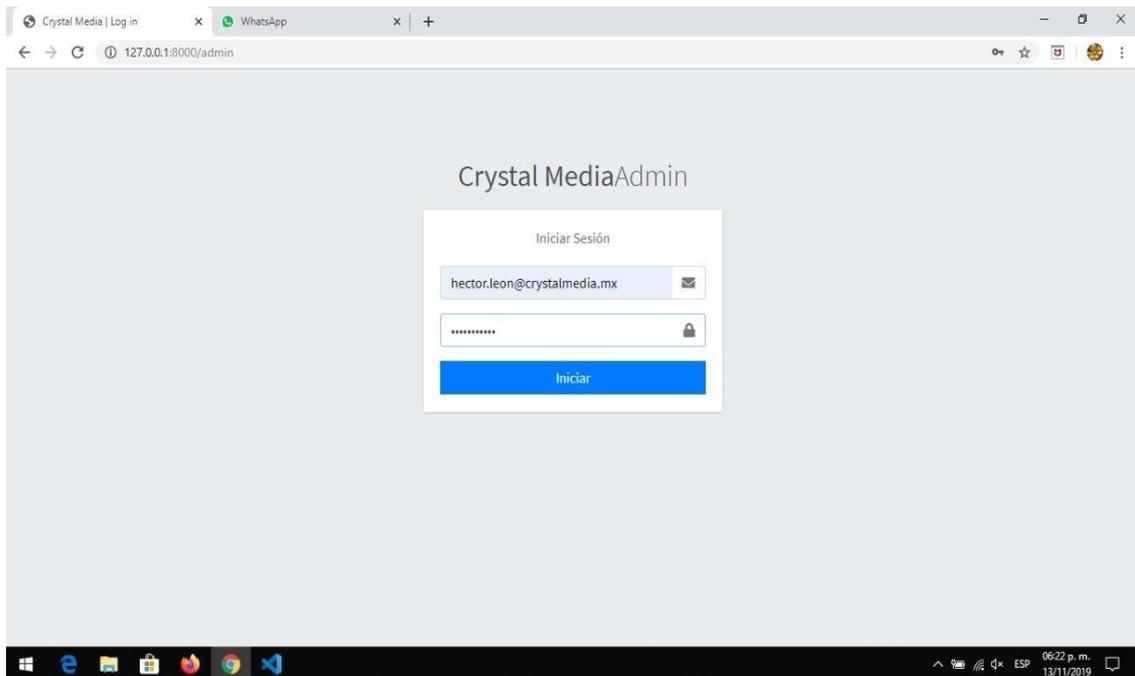


Figura 3. Login Administrador

Registro de transportistas

Se creó la vista para registrar transportistas desde el panel del administrador, esta vista tiene que ser llenada con nombre del transportista, tiempo de tránsito, logotipo, rango, altura, profundidad, peso, selección de impuestos, envío gratuito, facturación, impuestos, fuera de la gama de comportamiento, acceso de grupo y estado de transportista. (Figura. 4 Registro de transportistas)

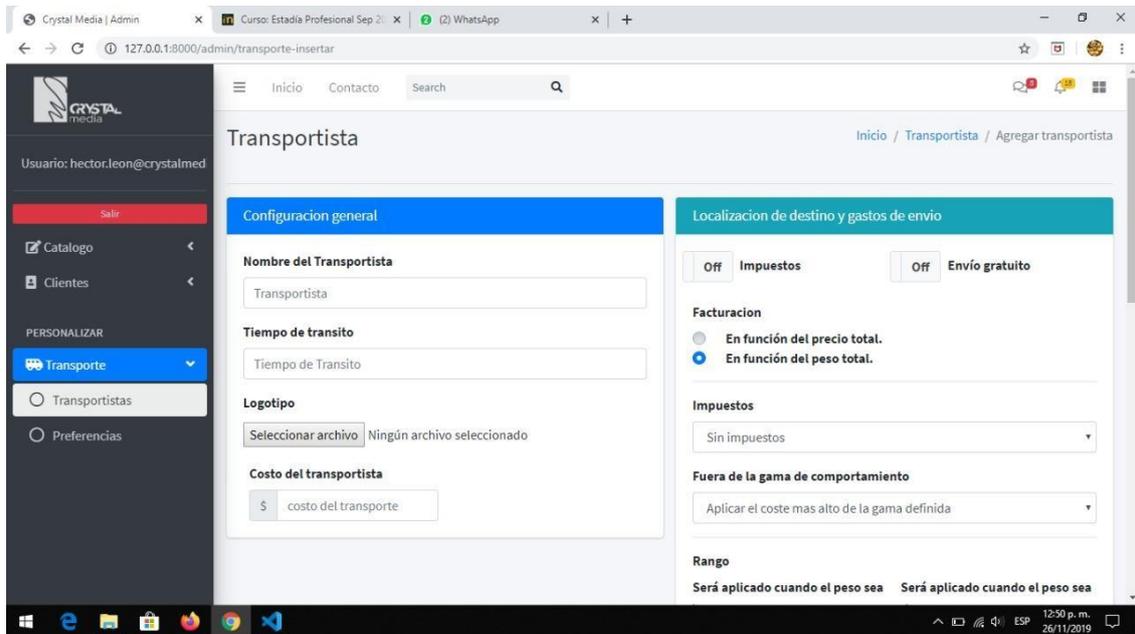


Figura 4. Registro de transportistas

3.3.1.2 Resultados del segundo Sprint

Se muestran las vistas realizadas en el segundo Sprint

Consulta de transportistas

Se creó la vista para consultar transportistas el cual mostrará el id, nombre, logotipo, retraso, estado, envió gratis, y también permitirá obtener una búsqueda ingresando en los campos de formulario. (Figura. 5 Consulta de transportistas)

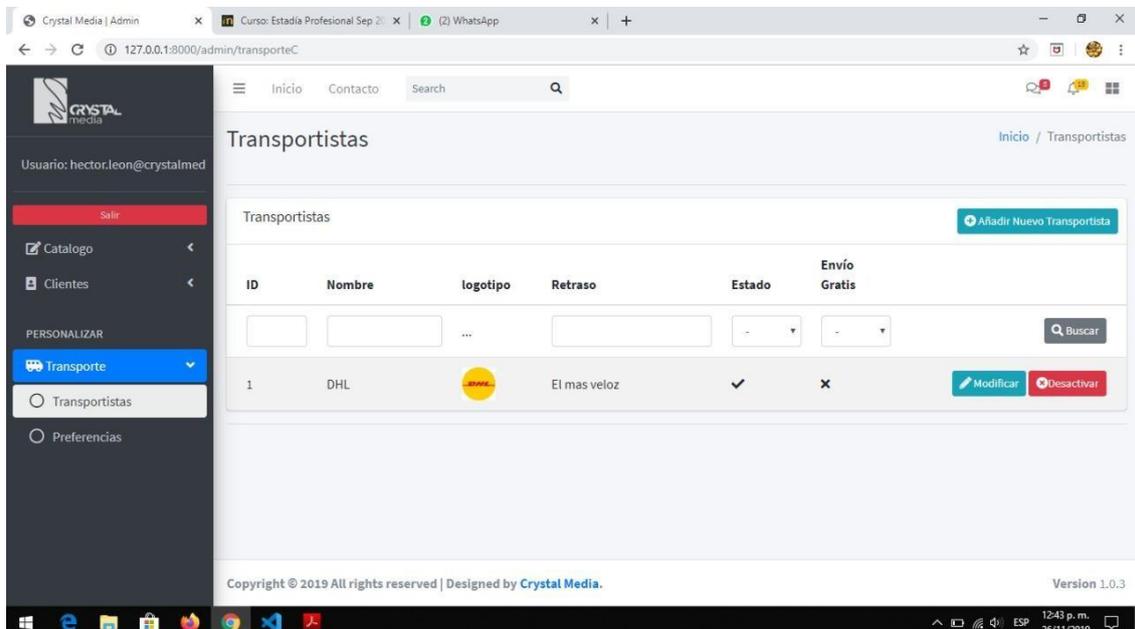


Figura 5. Consulta de transportistas

Modificar transportistas

Se creó una vista para modificar los datos de un transportista seleccionado por el usuario, editando y mostrando los campos nombre del transportista, tiempo de tránsito, logotipo, rango, altura, profundidad, peso, selección de impuestos, envío gratuito, facturación, impuestos, fuera de la gama de comportamiento, acceso de grupo y estado de transportista. (Figura 6. Modificar transportistas)

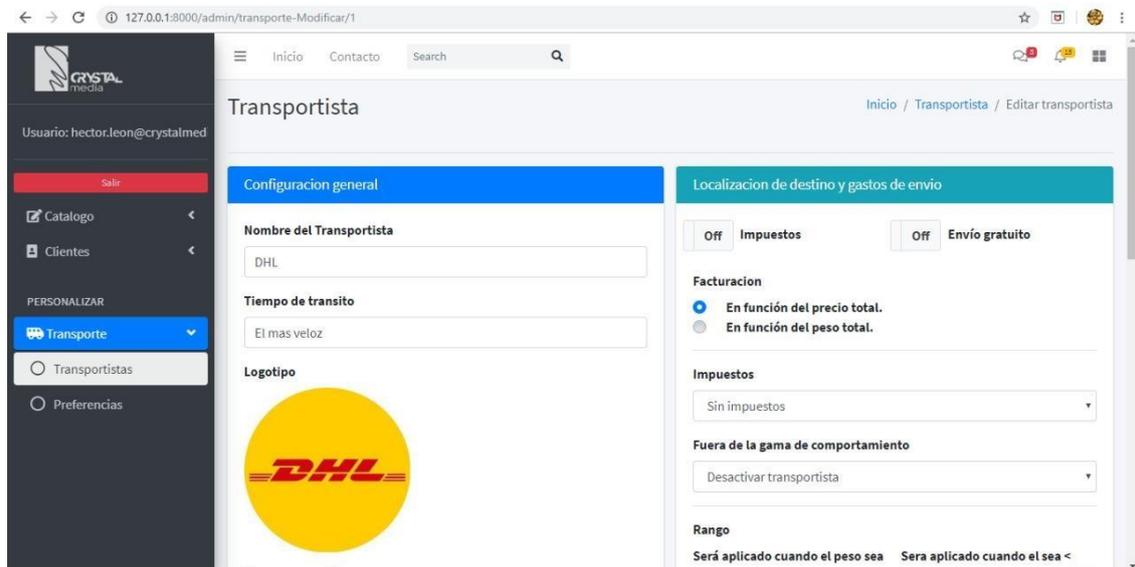


Figura 6. Modificar transportistas

Desactivar transportistas

Se creó un botón y una función que desactivará y activará el estado del transportista según fuera el caso. (Figura 7. Desactivar transportistas)

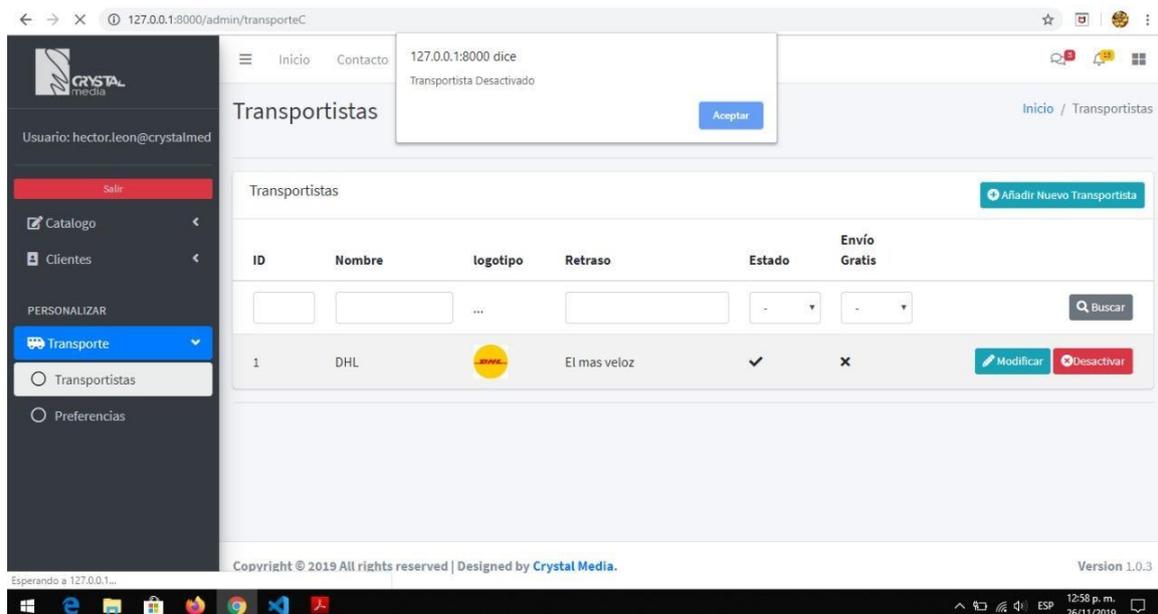


Figura 7. Desactivar transportistas

Registro de Marcas

Se creó la vista para registrar marcas desde el panel del administrador, esta vista tiene que ser llenada con nombre, descripción, imagen de la marca y seleccionar su estado. (Figura 8. Registro de Marcas)

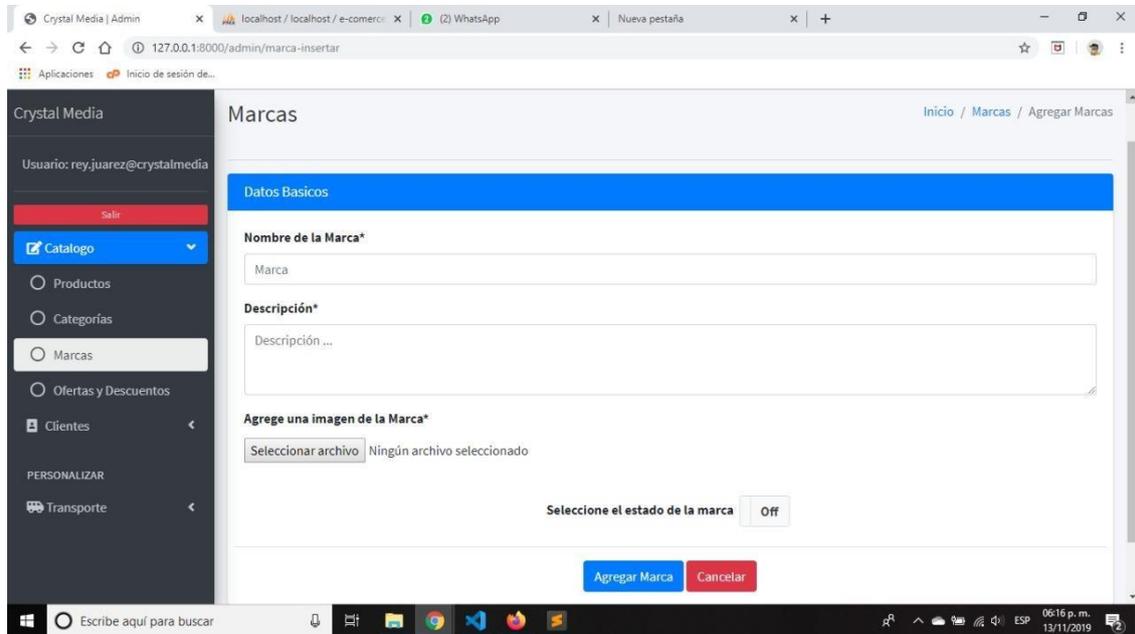


Figura 8. Registro de Marcas

Consulta de marcas

Se creó la vista para consultar marcas el cual mostrará el id, nombre, logotipo, estado, y también permitirá obtener una búsqueda ingresando datos en los campos del formulario. (Figura. 9 Consulta de marcas)

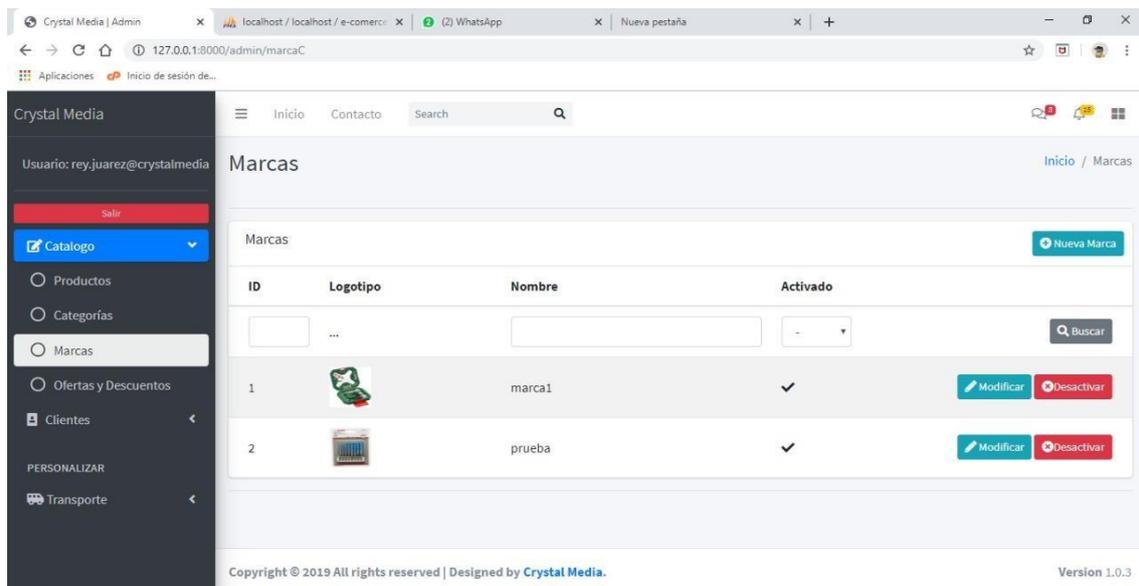


Figura 9. Consulta de marcas

Modificar marcas

Se creó una vista para modificar los datos de una marca seleccionada por el usuario, editando y mostrando los campos nombre, descripción, imagen de la marca y seleccionar su estado. (Figura 10. Modificar marcas)

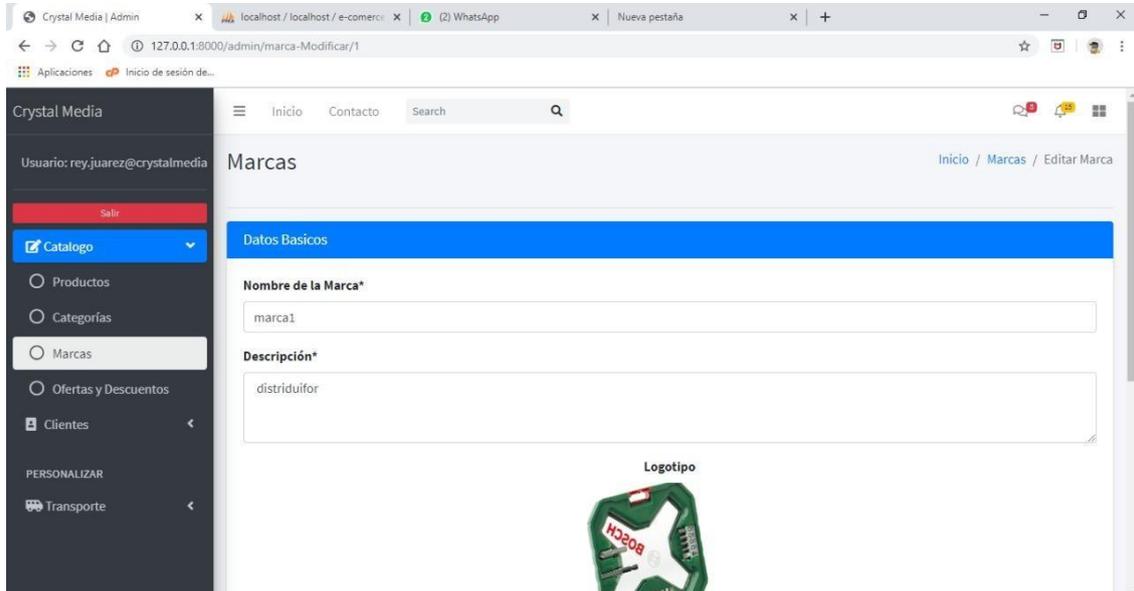


Figura 10. Modificar marcas

Desactivar marca

Se creó un botón y una función que desactivará y activará el estado de la marca según fuera el caso. (Figura 11. Desactivar marca)

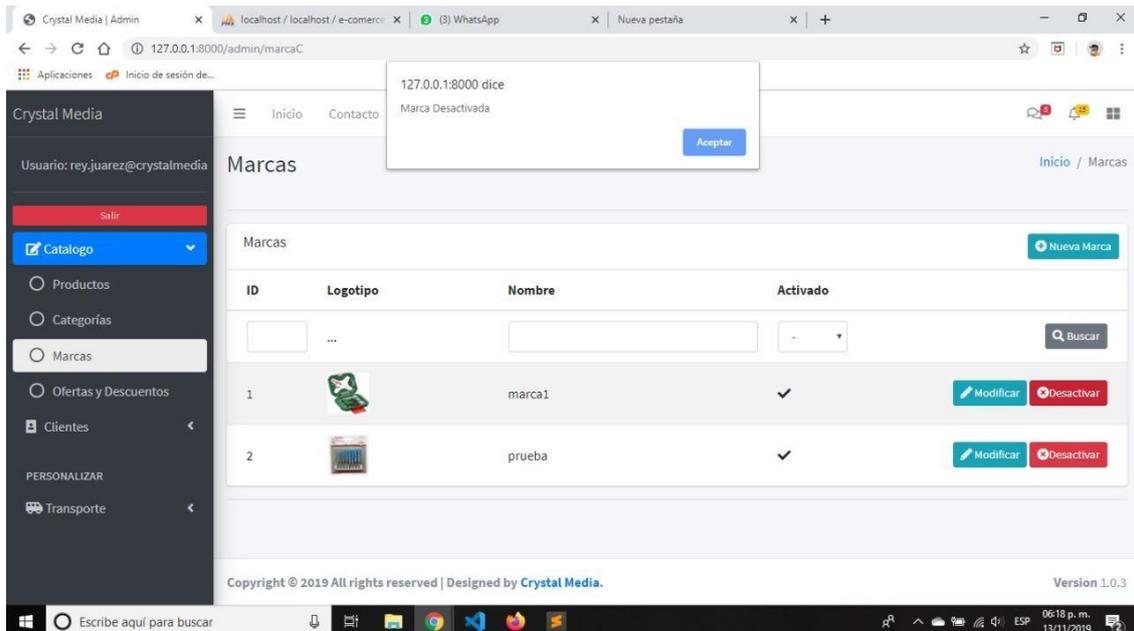


Figura 11. Desactivar marca

Login Cliente

Se creó un login para cliente.

Se muestran dos campos de entrada los cuales son correo electrónico y su contraseña. (Figura. 12 Login Cliente)

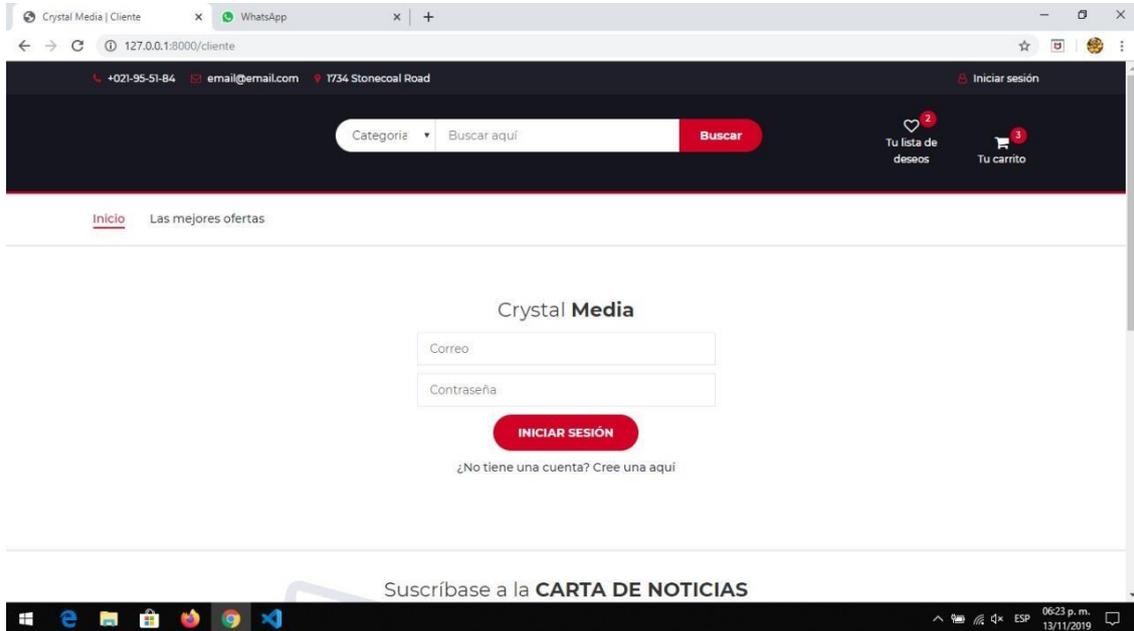


Figura 12. Login Cliente

Registro de clientes

Se creó el registro de clientes el cual tiene que ser llenado en los campos nombre, apellido paterno, apellido materno, teléfono, correo electrónico y contraseña. (Figura 13. Registro de clientes)

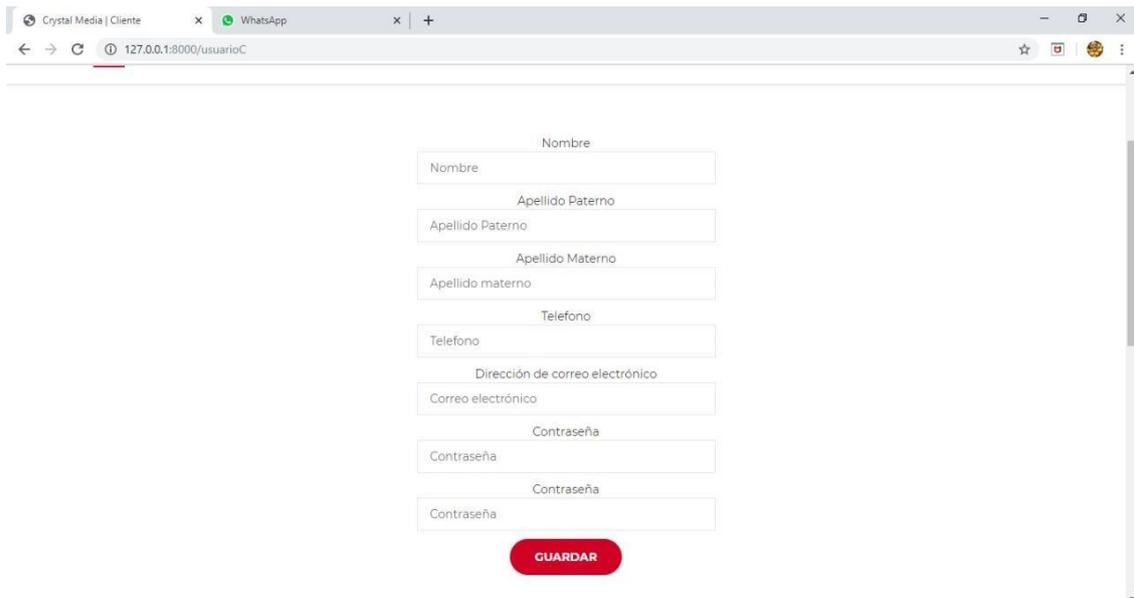


Figura 13. Registro de clientes

Página de inicio de la tienda

Se creó la página de inicio de la tienda la cual tendrán acceso invitados y clientes ya registrados. (Figura 14. Página de inicio de la tienda), (Figura 15. Página de inicio productos), (Figura 16. Página de inicio información)

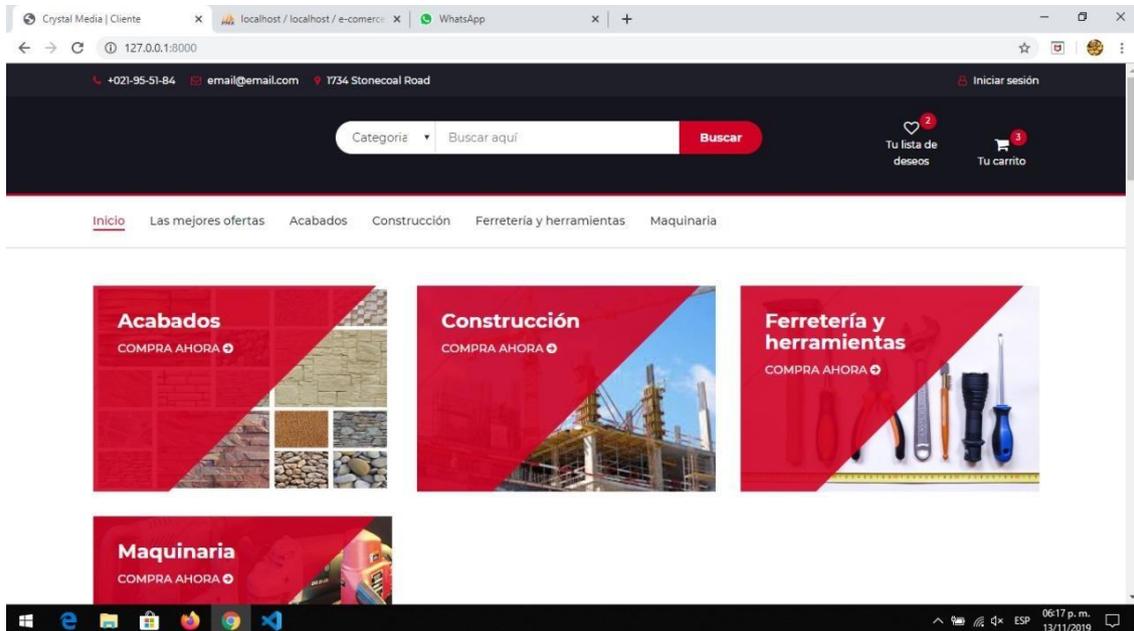


Figura 14. Página de inicio de la tienda

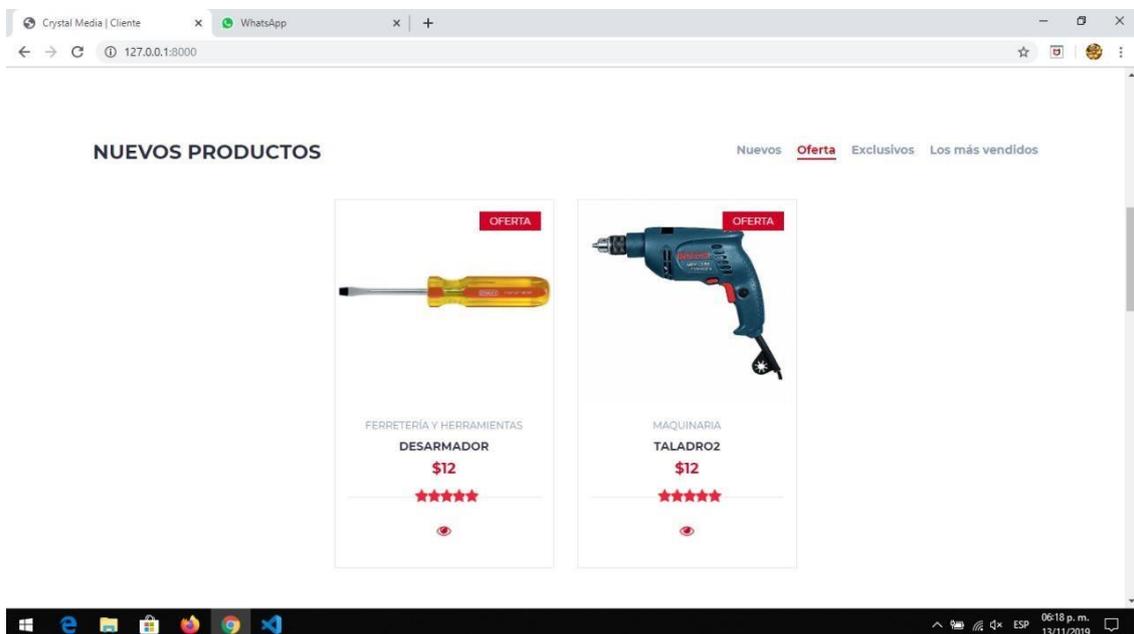


Figura 15. Página de inicio productos

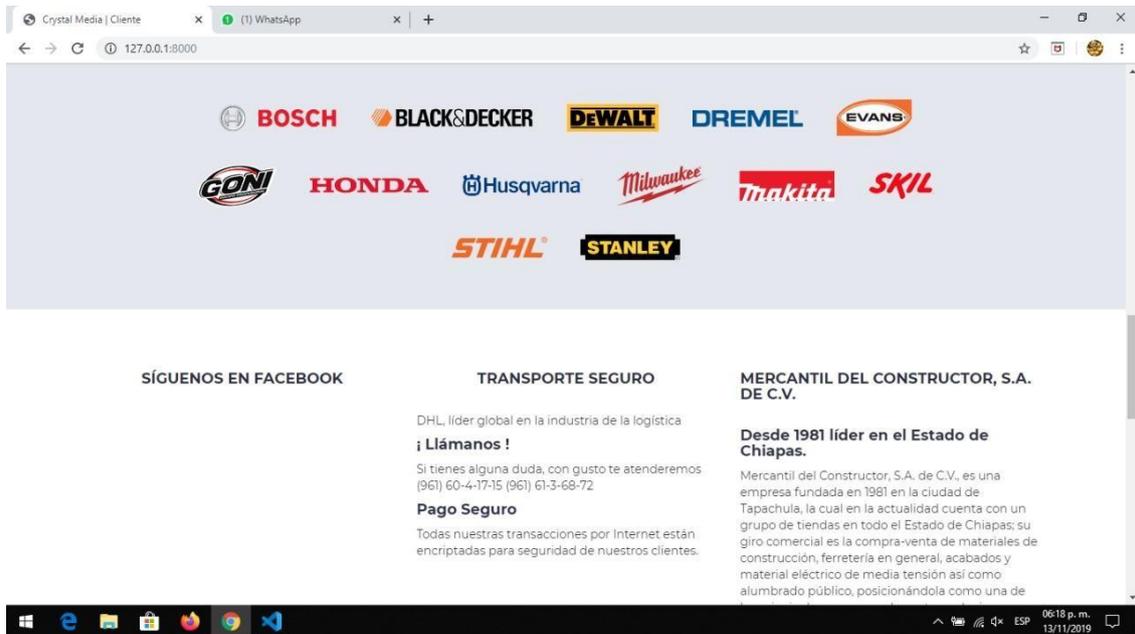


Figura 16. Página de inicio información

Consulta de categorías, subcategorías y sus productos

Se creó una vista que consulta las categorías en el cual se podrán ver sus subcategorías y sus productos de dichas subcategorías y/o categorías. Figura 17. Consulta de categorías, subcategorías y sus productos), (Figura 18. Consulta de categorías, subcategorías y sus productos)

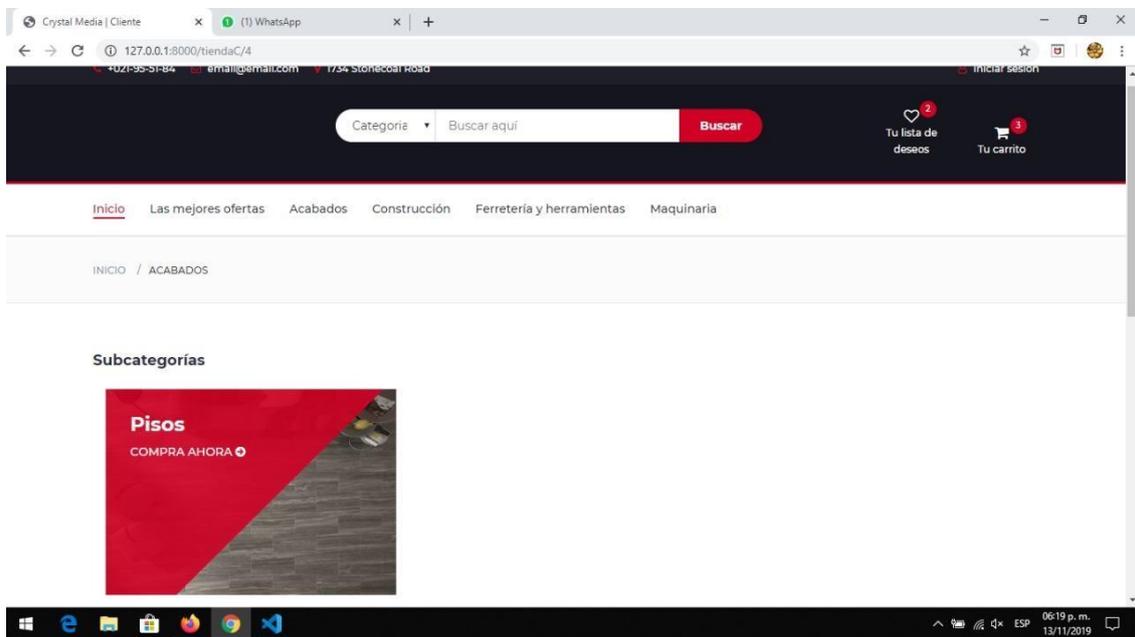


Figura 17. Consulta de categorías y subcategorías

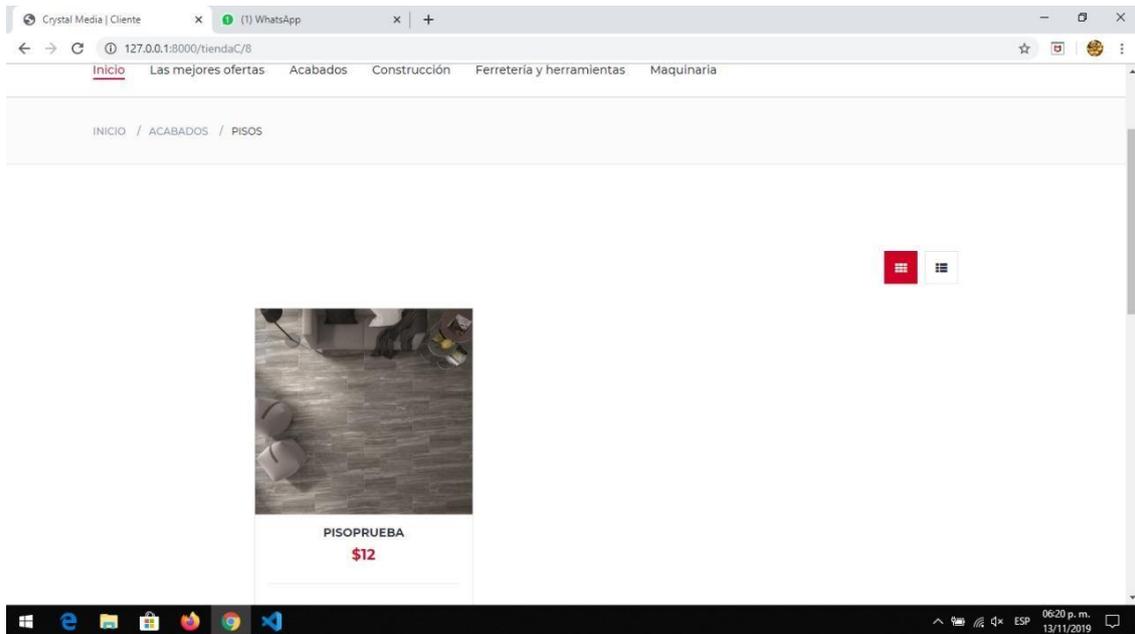


Figura 18. Consulta de productos

Consulta del producto por el cliente o visitante

Se creó una vista que consulta los detalles del producto. (Figura. 19 Consulta del producto por el cliente o visitante)

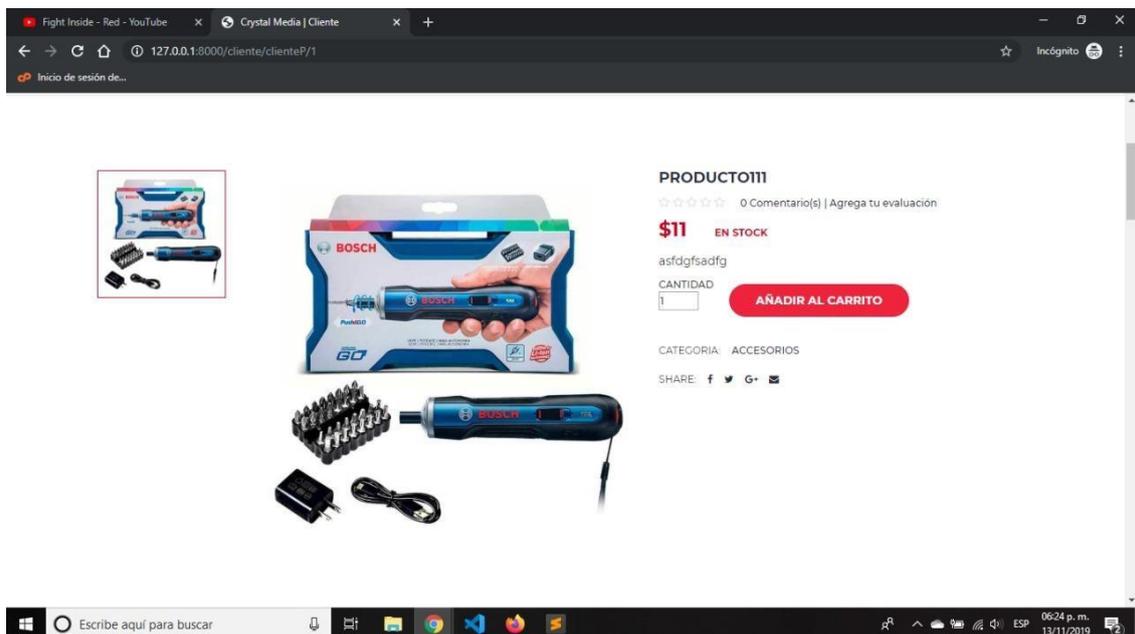


Figura 19. Consulta del producto por el cliente o visitante

Carrito

Se creó el carrito el cual podrá editar las cantidades del producto, eliminar el producto del carrito, consultar el total de artículos, subtotal y total. (Figura 20. Carrito)

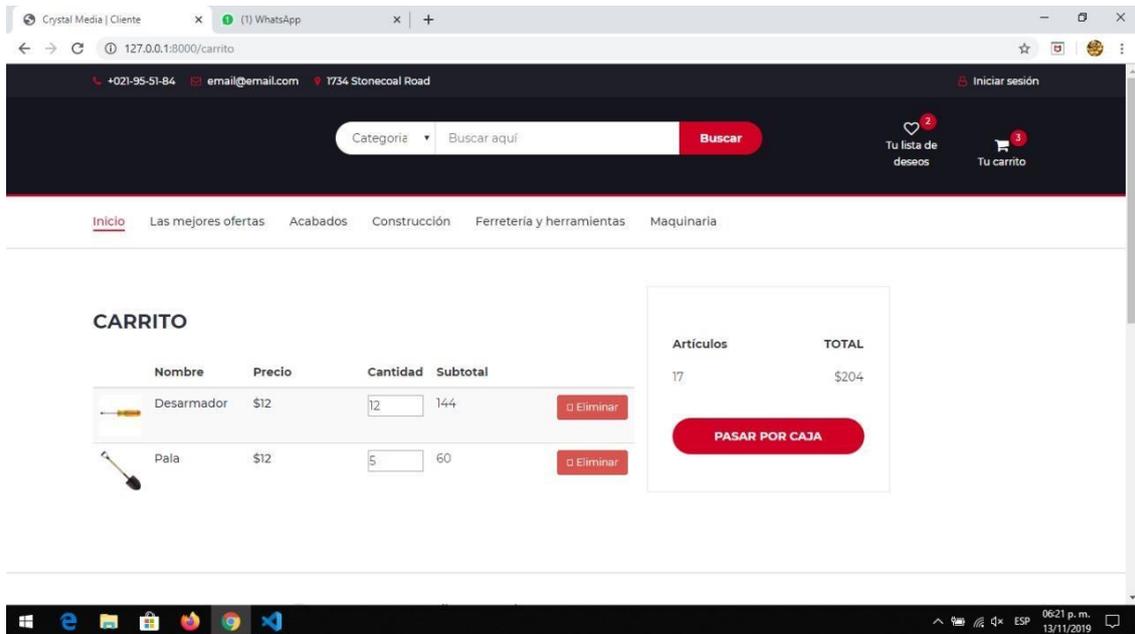


Figura 20. Carrito

Pago

Se creó el método de pago el cual pedirá los datos de la dirección del envío los cuales son calle, código postal, localidad, ciudad, estado, número interior, número exterior y teléfono, también mostrará los datos de la orden como su producto, total, permitirá elegir el transportista y realizar el pago mediante PayPal. (Figura 21. Pago), (Figura 22. PayPal)

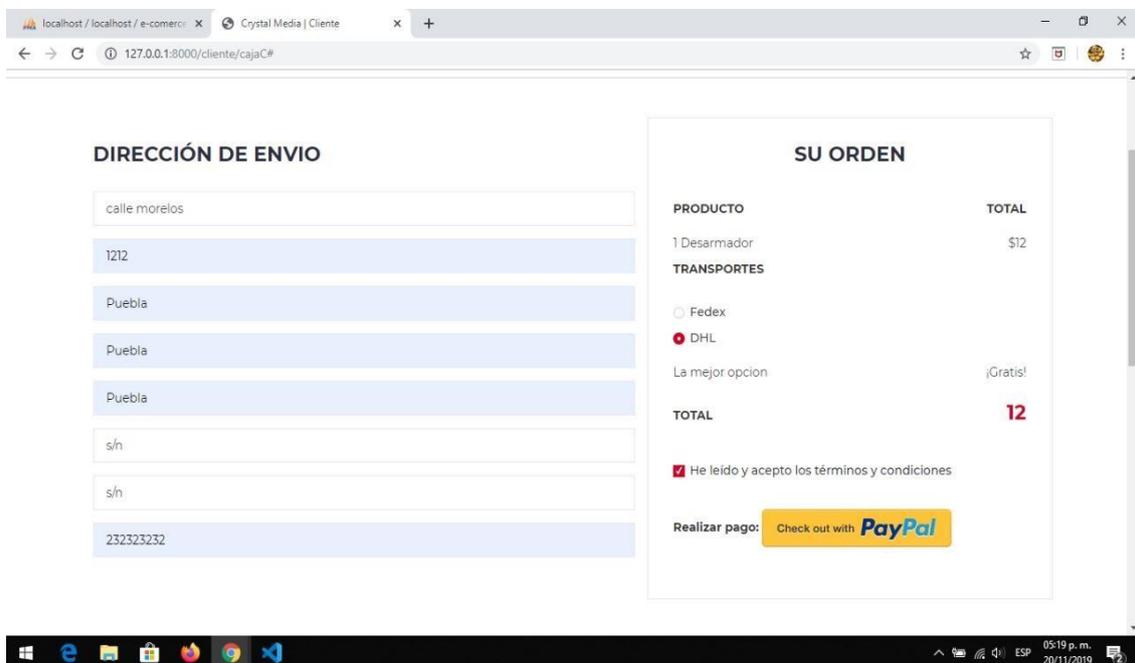


Figura 21. Pago

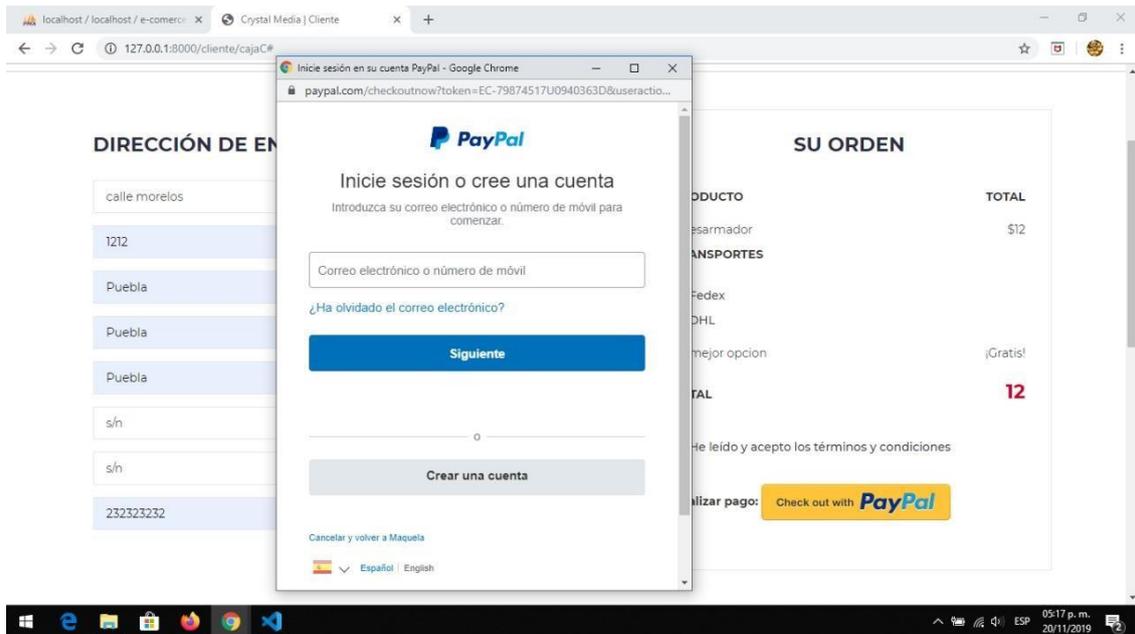


Figura 22. PayPal

3.3.1.3 Pruebas

La fase de pruebas son básicamente un conjunto de actividades dentro del desarrollo del sistema, el cual nos ayudarán a encontrar errores de seguridad, funcionalidad, usabilidad y compatibilidad.

Pruebas de usabilidad

Son procedimientos con los que se puede probar la usabilidad de una web. Utilizando una prueba de usabilidad ofrece posibilidades para optimizar la experiencia de usuario. El resultado de estas pruebas puede cambiar el diseño de la web o la redacción, sin embargo, aún esta prueba no se ha podido llevar a cabo ya que aún se encuentra en un estado local.

Pruebas de compatibilidad

Se realizaron pruebas en los siguientes 3 navegadores ya que por a hora se encuentra de forma local, de los cuales se obtuvo los siguientes resultados: (Tabla 7)

S.O.	 Chrome	 Firefox	 le 8, 9, 10, 11
Windows 7, 10			
Linux			
Android			

Tabla 7. Pruebas de compatibilidad

Pruebas de seguridad

La seguridad es un tema de gran importancia cuando desarrollamos sistemas, por lo que se busca constantemente que se realicen sistema con la mayor seguridad posible.

Las pruebas de seguridad fueron realizadas de forma local ya que el sistema se encuentra en desarrollo.

Las pruebas que se realizaron fueron:

- Autenticación: se comprobó que los usuarios registrados pudieran acceder al sistema con su contraseña y correo de usuario. En la figura 23 se muestra el inicio de sesión de un usuario y en la figura 24 se muestra que el usuario pudo iniciar correctamente.

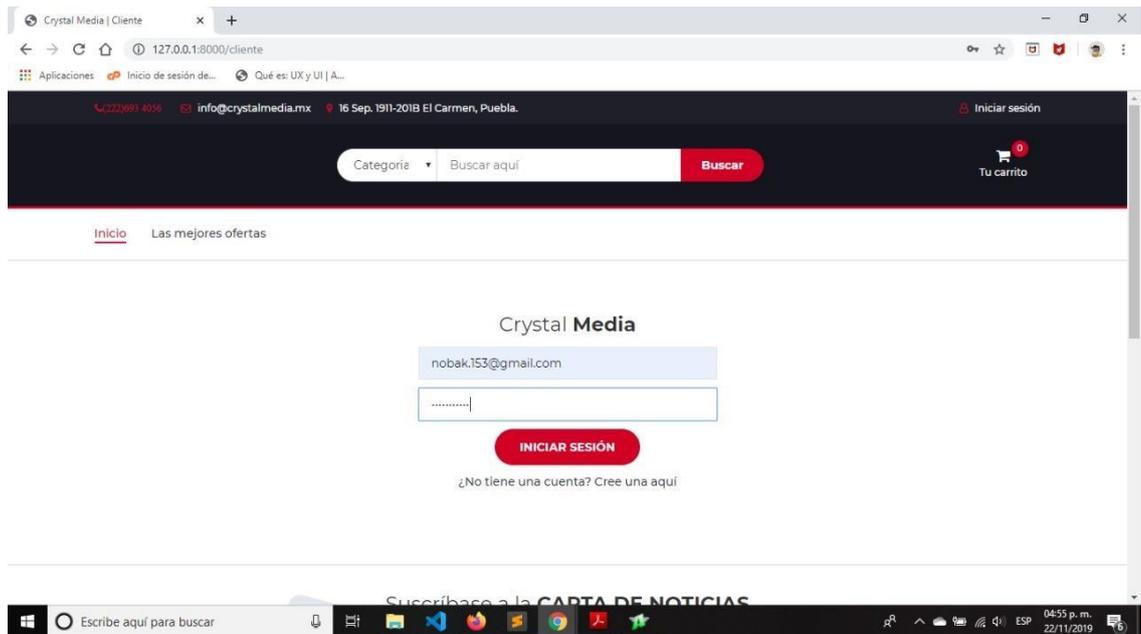


Figura 23. Inicio de sesión del cliente

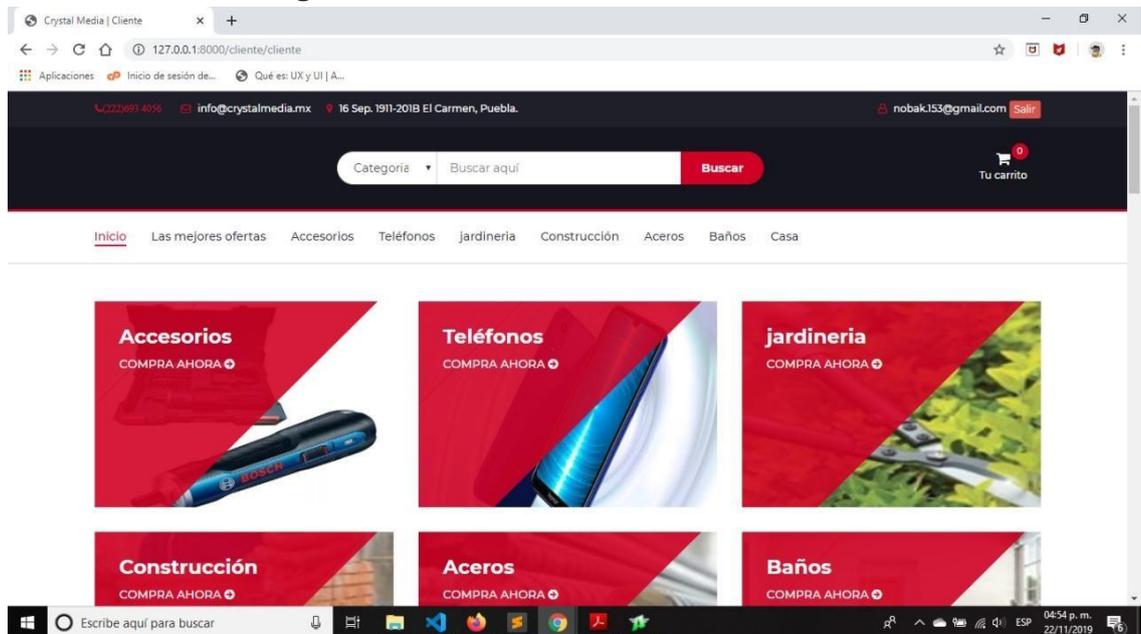


Figura 24. Inicio de sesión correctamente.

- Autorización: se comprobó que los usuarios de acuerdo a su rol tuvieran acceso solo al sistema o partes del sistema dependiendo su autorización.

En la figura 25 se muestra el inicio de sesión del WebMarter (Administrador).

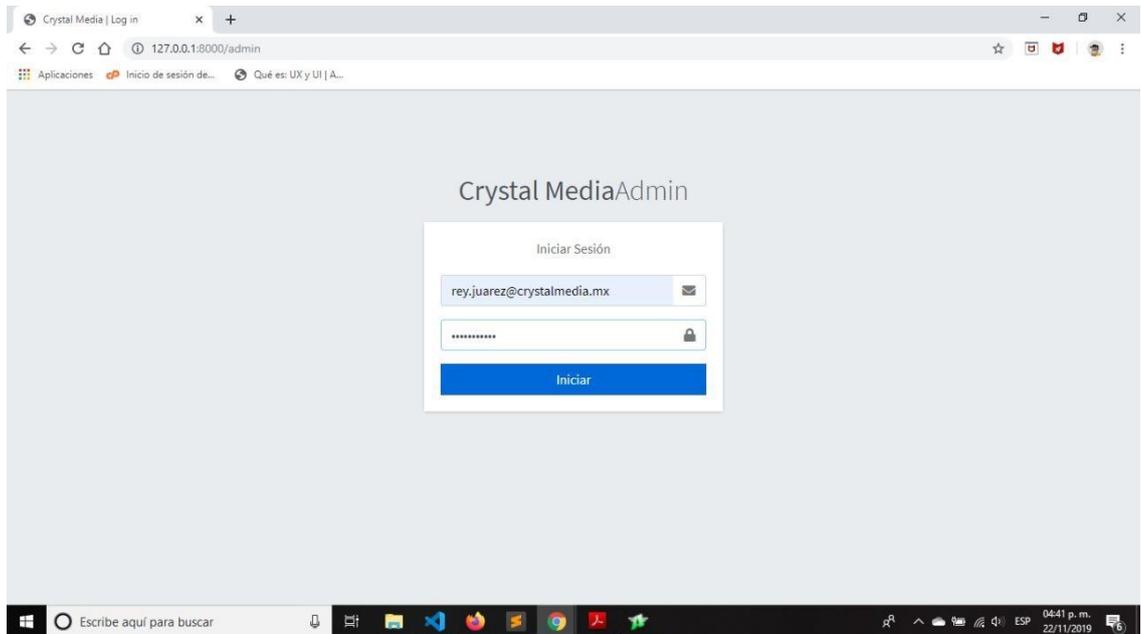


Figura 25. Inicio de sesión del WebMaster.

- **Cifrado:** Laravel nos brinda una ayuda para encriptar nuestros datos, de esta forma podemos hacer que nuestras contraseñas se ingresen a la base de datos encriptadas y así aumentar la confidencialidad de los datos, se comprobó que nuestros datos se ingresaran la base de datos de forma encriptada.

En la figura 26 se muestra como el sistema guarda la contraseña de forma encriptada.

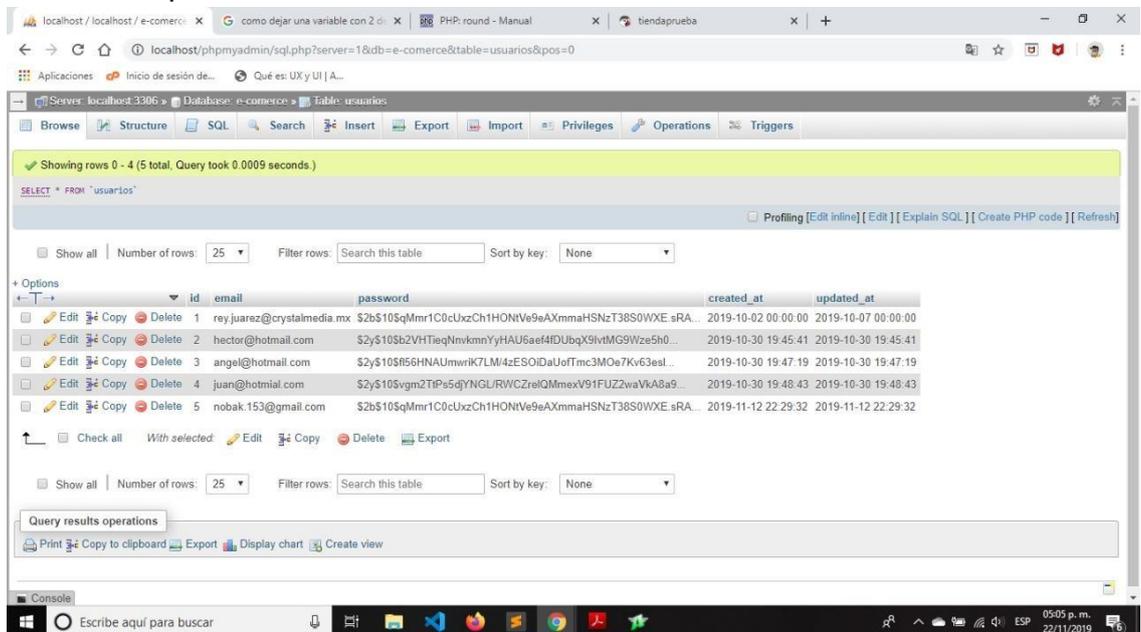


Figura 26. Encriptación la contraseña.

- **Seguridad con CSRF:** Con la ayuda de Laravel nos brinda una seguridad con CSRF el cual consiste en no permitir la inyección de SQL dentro de

las vistas, se realizaron pruebas por parte del líder del proyecto para comprobar el funcionamiento de la seguridad de CSRF. En la figura 27 se comprobó que en cada vista contenga un token CSRF.

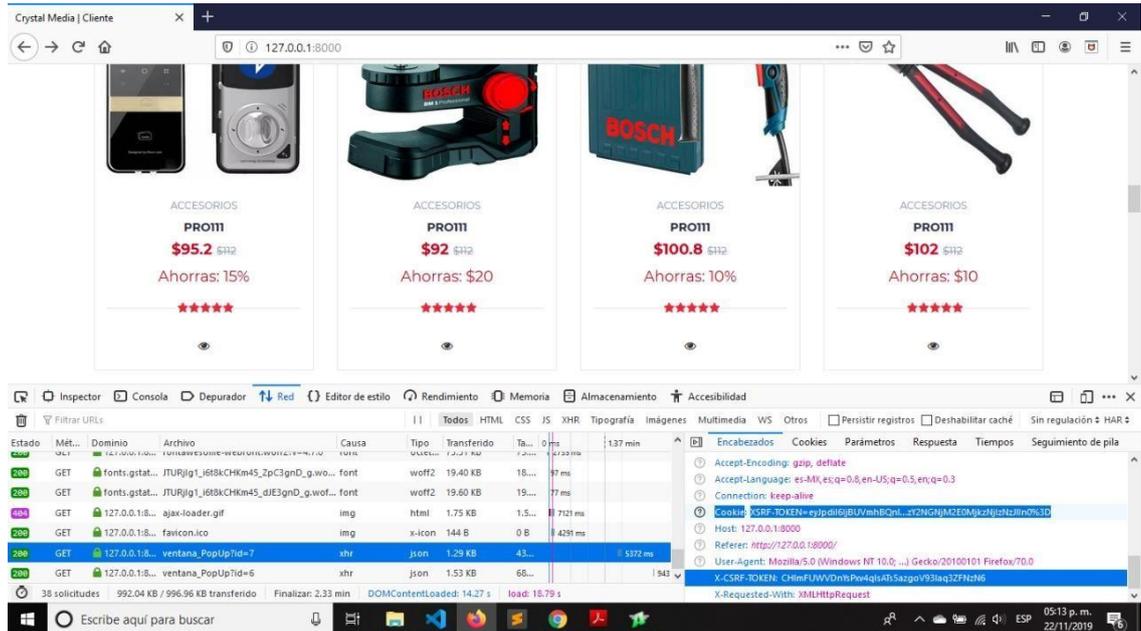


Figura 27. Token CSRF.

Pruebas de funcionalidad

Esta prueba fue fundamental para verificar y validar que los módulos desarrollados cumplieran con su objetivo respectivo:

- Se llevaron a cabo diferentes registros de usuarios, productos, marcas, categorías, transportistas y comentarios.

En la figura 28 se muestran cómo se llevó el ingreso de datos de un producto.

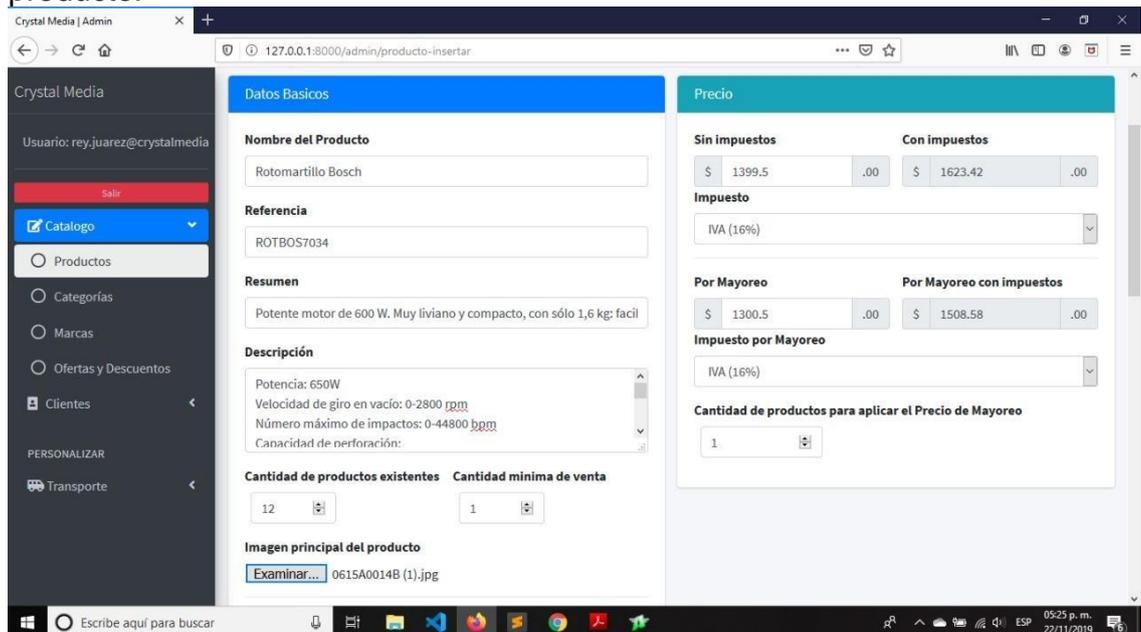


Figura 28. Ingreso de datos.

- Se agregaron datos de nuevos pedidos y compras del cliente. En la figura 29 se muestran cómo se realizó una compra de productos.

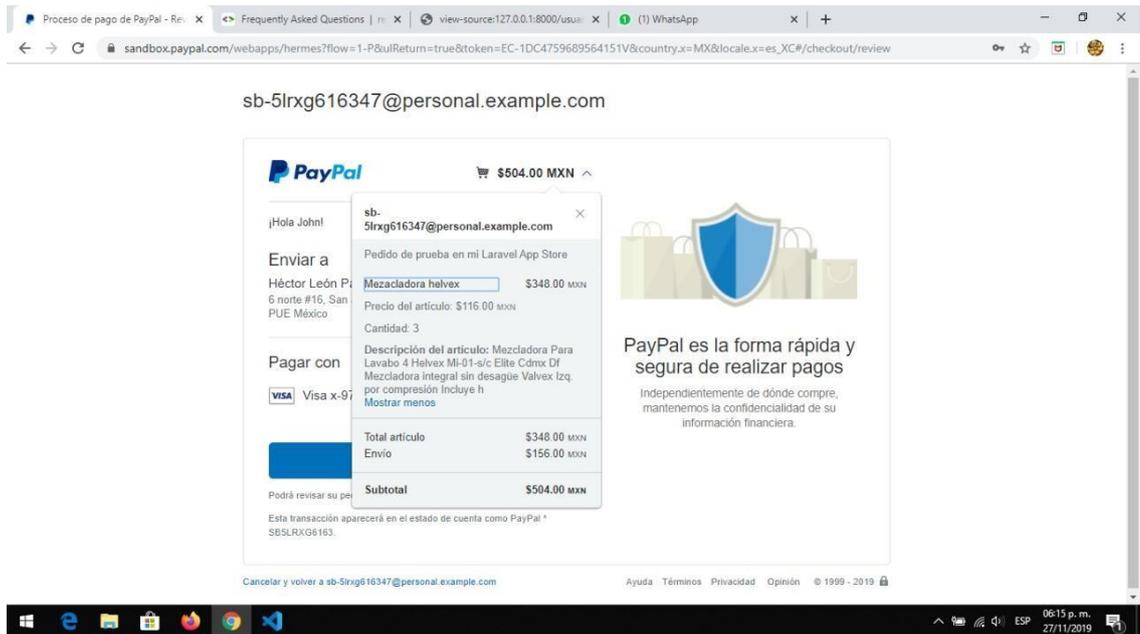


Figura 29. Compras de productos.

- Se verificó que no se rompiera el inicio de sesión con cualquier tipo de usuario. En la figura 30 se validó que sólo los usuarios firmados pudieran entrar al sistema de acuerdo a su rol.

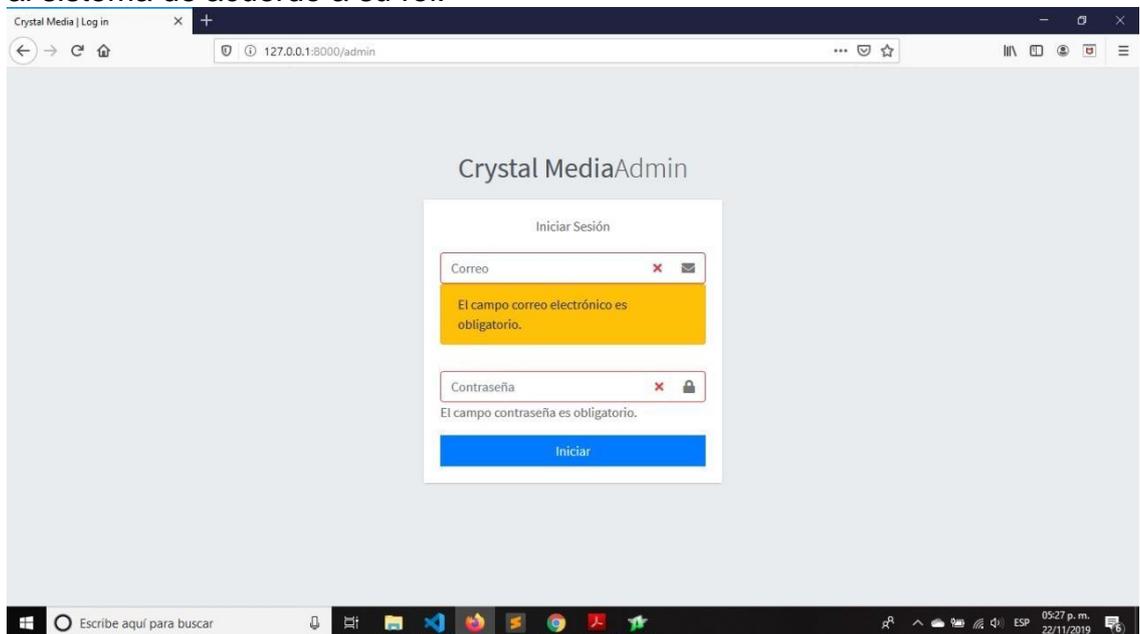


Figura 30. Validación de inicio de usuarios.

- Se verificó que todos los campos de formularios estuvieran validados. En la figura 31 se muestra la validación de los campos.

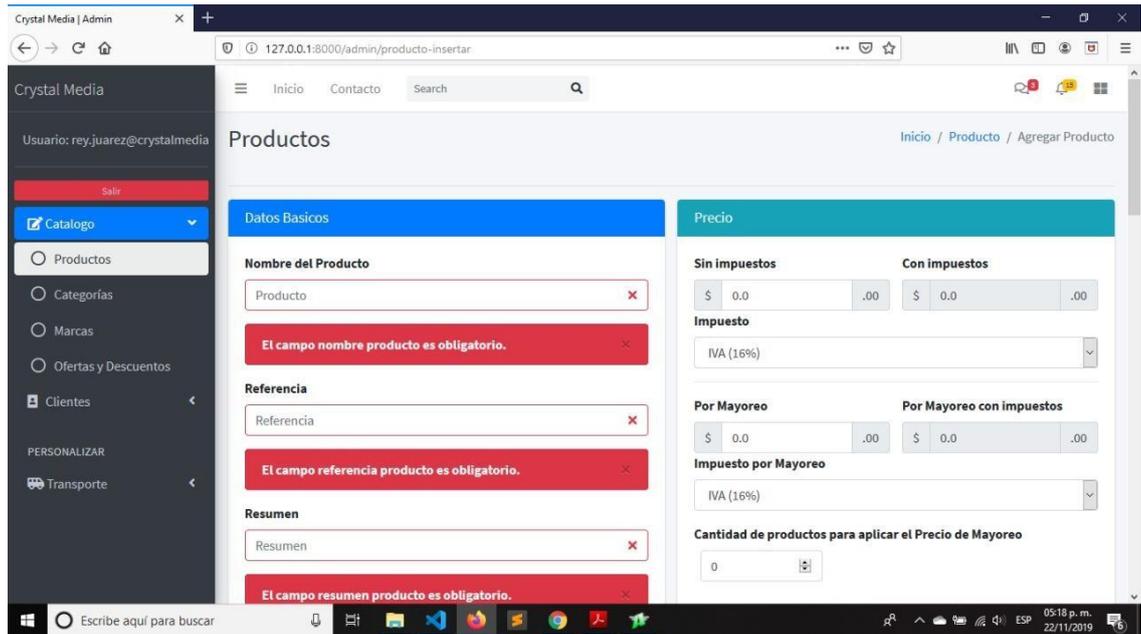


Figura 31. Validación de campos.

3.3.2 Retrospectiva Sprint 1

En el sprint 1 se llevó a cabo una reunión entre el equipo de desarrollo y el scrum master, Al cual se le explicó que problemas se tuvieron y como lo fuimos solucionando.

Problemas que se tuvieron:

- No se conocía el framework que se iba a utilizar.
- El framework que se eligió era nuevo para nosotros, el cual se tenía que aprender desde cero.
- Tiempos cortos para entrega de los módulos.
- Errores en el diseño de la base de datos.
- Errores en el diseño de las vistas.

Como fueron solucionados:

- Se investigaron sobre framework que se adecuaban al sistema y que nos pudieran ayudar para realizarlo.
- Se vieron videos tutoriales para aprender sobre el Framework Laravel.
- Se propuso al scrum master que se diera más tiempo para realizar las tareas ya que se estaba aprendiendo a la par que se desarrollaban.
- Se analizó la base de datos para realizar los nuevos cambios que se solicitaron.

- De acuerdo a los errores del diseño de las vistas se realizaron los cambios propuestos por el scrum master.

Sprint 2

En el sprint 2 se llevó a cabo una reunión entre el equipo de desarrollo y el scrum master, Al cual se le explicó que problemas se tuvieron y como lo fuimos solucionando.

Problemas que se tuvieron:

- Errores con JavaScript, JQuery y Ajax.
- Errores con sesiones.
- Errores con el módulo de carrito y pago con PayPal.

Como fueron solucionados:

- Se investigó de como realzar funciones con JavaScript, JQuery y Ajax para realizar algunas funcionalidades que lo requerían.
- Se vieron tutoriales de cómo solucionar los problemas marcados por Laravel en las consultas de usuarios en la base de datos.
- Laravel tiene su propio carrito de compras el cual se tuvo que comprender como funcionaba y así implementar ese modulo que trae Laravel.
- Se buscaron tutoriales de cómo implementar el pago con PayPal en el sistema.

4. Conclusiones y recomendaciones

Como conclusión se cumplieron totalmente con los objetivos planteados al inicio de la estadía, éstos fueron presentados al asesor y al dueño del proyecto cabe mencionar que este proyecto aún no está completo por lo que se pretende que en el futuro este proyecto sea concluido.

El haber llevado acabo el desarrollo de este proyecto durante la estadía profesional, al inicio fue algo difícil porque se ocupó un framework, el cual fue Laravel; sin embargo, yo nunca había utilizado uno, por lo que tuve que investigar en varias fuentes para poder llevar a cabo este proyecto, poco a poco fui comprendiendo Laravel y gracias a que trabaja con un modelo vista controlador se me facilitó más comprenderlo.

En este proceso de estadía profesional he aprendido demasiadas cosas en el ámbito laboral, como relacionarme con mis compañeros, ser puntual, trabajar en equipo y a comprender y hablar en juntas, también me ayudó a tener mayor conocimiento en el lenguaje de programación PHP y el framework Laravel.

El hacer la estadía profesional me ayudó a tener mayor experiencia en el ámbito laboral y mejores conocimientos que me ayudarán en el futuro para obtener un puesto laboral.

5. Referencias bibliográficas

- [1] URL: <https://marketing4ecommerce.mx/que-es-el-ecommerce/> Página donde se puede consultar información sobre que es un Sistema E-Commerce.
- [2] URL: <https://proyectosagiles.org/que-es-scrum/> Página donde se puede consultar información sobre la Metodología SCRUM.
- [3] URL: http://www.ecured.cu/Visual_Studio_Code Página donde se puede consultar información sobre Visual Code.
- [4] URL: <https://programadorwebvalencia.com/vim-vs-vscode/> Página donde se puede consultar información sobre las ventajas y desventajas de Visual Code.
- [5] URL: <https://www.hostinger.mx/tutoriales/que-es-apache/> Página donde se puede consultar información sobre Apache.
- [6] URL: <https://www.aprenderaprogramar.com/> Página donde se puede consultar información sobre PHP.
- [7] URL: <https://adictoalcodigo.blogspot.com/2016/07/ventajas-y-desventajas-de-programar-en-40.html> Página donde se puede consultar información sobre ventajas y desventajas de PHP.
- [8] URL: <https://www.arsys.es/blog/programacion/que-es-laravel/> Página donde se puede consultar información sobre Laravel.
- [9] URL: <https://es.slideshare.net/RafaelLobo4/framework-laravel> Página donde se puede consultar información sobre ventajas y desventajas de Laravel.
- [10] URL: <https://www.acens.com/wp-content/images/2014/12/wp-composer-acens.pdf> Página donde se puede consultar información sobre Composer
- [11] URL: <https://uniwebsidad.com/tutoriales/como-instalar-composer-globalmente> Página donde se puede consultar información sobre desventajas de Composer.
- [12] URL: <http://www.software.com.mx/p/sql-power-architect> Sitio web en donde se puede obtener el software SQL Power Architect y describen sus características.
- [13] URL: <http://www.acercadehtml.com/manual-html/que-es-html.html> Página web donde se puede obtener información de HTML.
- [14] URL: <http://lenguaje-html-cufinho.blogspot.com/p/ventajas-sencillo-que-permite-describir.html> Página web donde se puede obtener información de las ventajas y desventajas de HTML.

[15] URL: [https://www.ecured.cu/Sistema de Gestión de Bases de Datos: MySQL](https://www.ecured.cu/Sistema_de_Gesti%C3%B3n_de_Bases_de_Datos:_MySQL) Página web donde se puede obtener información Mysql.

[16] URL: <https://www.genbeta.com/desarrollo/conociendo-github-el-servicio-donde-alojar-tus-repositorios-git-como-el-nuestro> Página web donde se puede obtener información de GitHub.

[17] URL: <https://openwebinars.net/blog/ventajas-del-uso-de-git/> Página web donde se puede obtener información de las ventajas y desventajas de GitHub.



Universidad Politécnica de Puebla
Ingeniería en Informática

Héctor León Palillero
Guadalupe de Jesús Raúl Aguilera Cabrera
Rebeca Rodríguez Huesca

Este documento se distribuye para los términos de la
Licencia 2.5 Creative Commons (CC-BC-NC-ND 2.5 MX)