



Pontificia Universidad
Católica del Ecuador

**PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR
FACULTAD DE INGENIERÍA
ESCUELA DE SISTEMAS Y COMPUTACIÓN**

**DESARROLLO DE UN SISTEMA PARA LA GESTIÓN DE
TALLERES AUTOMOTRICES, ORIENTADO A LA WEB Y A
DISPOSITIVOS MÓVILES. CASO DE ESTUDIO: TALLER
AUTOMOTRIZ AUTOTREK**

CARLOS ANDRÉS PINTO VACACELA

DIRECTOR: ING. FABIÁN DE LA CRUZ

QUITO, 2020

Dedicatoria

Le dedico esta disertación a mi madre, padrastro, hermano, abuela y padre que en paz descansen, que siempre estuvieron ahí para apoyarme incondicionalmente en cualquier circunstancia de mi vida personal y estudiantil.

Agradecimientos

Quiero dar las infinitas gracias a mi madre, quien estuvo para mí no solo en mi vida personal, sino también en mi vida académica, brindándome su apoyo en todos los sentidos y siendo el pilar fundamental de mi vida.

De igual manera, agradezco a mi director de tesis Ing. Fabián de la Cruz, y a mis revisores Ing. Charles Escobar e Ing. Henry Roa, quienes me han ayudado a culminar mi trabajo de titulación y mi carrera universitaria.

Agradezco a mi familia, quienes han inculcado en mí todo el amor y valores para convertirme en la persona que soy.

Le agradezco a Pamela y mis amigos que me ayudaron en varios de los aspectos personales y estudiantiles, y con los que pude contar siempre para todas las ocasiones que se presentaron.

Tabla de contenidos

| | |
|---|----|
| Resumen..... | 8 |
| Capítulo 1 Introducción | 9 |
| Tema | 9 |
| Razones que motivaron al estudio | 9 |
| Planteamiento del problema..... | 10 |
| Objetivos:..... | 10 |
| Objetivo general..... | 10 |
| Objetivos específicos | 10 |
| Capítulo 2 Caso de estudio: Taller Automotriz AUTOTREK..... | 11 |
| Introducción | 12 |
| Situación de la entidad | 13 |
| Información del taller..... | 13 |
| Visión..... | 14 |
| Misión. | 15 |
| Problemática | 15 |
| Administración..... | 15 |
| Orden de prioridad. | 15 |
| Movilidad..... | 15 |
| Notificaciones. | 16 |
| Procesos | 16 |
| Ingreso de Vehículos..... | 16 |
| Asignación de Cita..... | 18 |
| Post Venta. | 19 |
| Capítulo 3 Marco Teórico..... | 20 |
| Aplicaciones web | 20 |
| Historia de la WWW e Internet. | 20 |
| Redes de comunicación..... | 22 |
| Sitios web..... | 24 |
| Aplicaciones Web. | 26 |
| Aplicaciones para Dispositivos Móviles..... | 31 |
| Historia de los Dispositivos Móviles. | 31 |
| Aplicaciones móviles. | 35 |

| | |
|--|-----|
| Tipos de aplicaciones móviles | 37 |
| Herramientas de Desarrollo | 40 |
| PHP | 40 |
| HTML | 41 |
| CSS | 42 |
| JavaScript..... | 44 |
| Laravel | 45 |
| React | 45 |
| Ionic | 46 |
| Bases de Datos | 46 |
| MySQL | 47 |
| OOHDM | 48 |
| Capítulo 4: Fase inicial | 51 |
| Análisis y obtención de requerimientos | 51 |
| Especificación de requerimientos de software..... | 52 |
| Análisis de herramientas de desarrollo | 110 |
| Visual Studio Code | 111 |
| Mozilla Firefox | 111 |
| PhpMyAdmin..... | 112 |
| Power Designer..... | 112 |
| Análisis de la base de datos | 112 |
| Capítulo 5: Diseño, implementación y pruebas | 114 |
| Consideraciones | 114 |
| Diseño Conceptual | 115 |
| Diseño navegacional | 117 |
| Clases Navegacionales..... | 117 |
| Contexto Navegacional..... | 118 |
| Diseño de Interfaz Abstracta..... | 119 |
| Ingreso al sistema..... | 119 |
| Pantalla principal | 120 |
| Gestión de clientes | 120 |
| Diagrama de la Base de Datos | 122 |
| Modelo conceptual de datos..... | 122 |
| Modelo lógico de datos..... | 123 |
| Implementación..... | 124 |

| | |
|--|-----|
| Conexión a la Base de Datos: | 124 |
| Conexión con el proveedor de E-mail | 125 |
| Rutas para consumir servicio | 126 |
| Modelo y Controlador..... | 126 |
| Vista del Cliente..... | 127 |
| Pruebas del cliente | 129 |
| Capítulo 6: Conclusiones y Recomendaciones | 146 |
| Conclusiones | 146 |
| Recomendaciones | 147 |
| Bibliografía | 148 |

Tablas

| | |
|--|-----|
| Tabla 1 Java vs JavaScript. | 44 |
| Tabla 2 Etapas de la Metodología OOADM..... | 49 |
| Tabla 3 Personal Involucrado. | 53 |
| Tabla 4 Características de los Usuarios. | 55 |
| Tabla 6 Prueba Ingreso al Sistema..... | 130 |
| Tabla 7 Prueba Gestión de Usuarios. | 130 |
| Tabla 8 Prueba Gestión de Clientes. | 131 |
| Tabla 9 Prueba Gestión de Vehículos. | 133 |
| Tabla 10 Prueba Gestión de Servicios. | 134 |
| Tabla 11 Prueba Gestión Cotizaciones. | 135 |
| Tabla 12 Prueba Gestión de Compras..... | 136 |
| Tabla 13 Prueba Gestión de Facturas..... | 137 |
| Tabla 14 Prueba Gestión de Turnos..... | 139 |
| Tabla 15 Prueba Gestión de Notificaciones..... | 140 |
| Tabla 16 Prueba Gestión de Promociones y Descuentos..... | 142 |
| Tabla 17 Prueba Gestión de Reportes Financieros. | 143 |

Figuras

| | |
|---|-----|
| Figura 1 Logo del Taller Automotriz AUTOTREK. | 14 |
| Figura 2 Diagrama de actividades de Ingreso de Vehículos. | 17 |
| Figura 3 Diagrama de actividades de Asignación de Cita. | 18 |
| Figura 4 Diagrama de actividades de Post Venta. | 19 |
| Figura 5 Los pilares de la Web. | 21 |
| Figura 6 Esquema de una Red LAN. | 23 |
| Figura 7 Esquema de una Red MAN. | 23 |
| Figura 8 Ejemplos de diagramas de red WAN con varias sedes sociales. | 24 |
| Figura 9 Esquema básico del servicio web. | 27 |
| Figura 10 Separación de funciones. | 28 |
| Figura 11 Tecnologías empleadas en el cliente y en el servidor web. | 31 |
| Figura 12 Motorola DynaTAC 8000X, uno de los primeros teléfonos móviles fabricados en gran cantidad. | 33 |
| Figura 13 Teléfono móvil de la segunda generación. | 33 |
| Figura 14 Compaq iPaq 3630, el primer PocketPC de éxito. | 34 |
| Figura 15 Dispositivo móvil de la última generación. | 34 |
| Figura 16 Web pública. | 38 |
| Figura 17 Juegos para móviles. | 39 |
| Figura 18 Aplicación de Facebook para móviles. | 40 |
| Figura 19 Diagrama general de casos de uso. | 58 |
| Figura 20 Gestión de cotizaciones. | 66 |
| Figura 21 Gestión de compras. | 73 |
| Figura 22 Gestión de facturas. | 80 |
| Figura 23 Gestión de turnos. | 85 |
| Figura 24 Gestión de notificaciones. | 92 |
| Figura 25 Gestión de promociones y descuentos. | 100 |
| Figura 26 Diagrama conceptual del sistema. | 110 |
| Figura 27 Diagrama de clases conceptual. | 116 |
| Figura 28 Diagrama de clases navegacional. | 117 |
| Figura 29 Diagrama de contexto navegacional. | 118 |
| Figura 30 Diagrama de vista abstracta de Ingreso al Sistema. | 119 |
| Figura 31 Diagrama de vista abstracta Pantalla Principal. | 120 |
| Figura 32 Diagrama de vista abstracta Ingreso Cliente. | 120 |
| Figura 33 Diagrama de vista abstracta Consulta Cliente. | 121 |
| Figura 34 Diagrama de vista abstracta Editar Cliente. | 121 |
| Figura 35 Diagrama conceptual de la base de datos. | 122 |
| Figura 36 Diagrama lógico de la base de datos. | 123 |
| Figura 37 Modelos, controladores y conexión a la base de datos. | 125 |
| Figura 38 Conexión con el proveedor de E-mail. | 126 |
| Figura 39 Rutas para el consumo de servicios. | 126 |
| Figura 40 Modelo vehículo. | 127 |
| Figura 41 Controlador vehículo. | 127 |
| Figura 42 Código JavaScript con React. | 129 |

Resumen

La presente disertación tiene como finalidad el desarrollo de un sistema que sirva para la gestión del taller automotriz AUTOTREK, este sistema está orientado a la web y a dispositivos móviles, por lo que se aplican las tecnologías de la información pertinentes para la sustentación de las necesidades y construcción del sistema. Para lograr el cometido, se realizó una previa investigación acerca de todos los conceptos necesarios, así como también, se estudió a la entidad que en este caso es el taller automotriz, tomando en cuenta sus procesos y problemas. Por motivo de cubrir todos los requerimientos del cliente, la disertación consta de varias etapas contempladas en la Ingeniería de Software, como son un levantamiento de requerimientos, el cual se apoya en base a los estándares IEEE. Se utiliza la metodología para desarrollo web OOHDM, que brinda un esquema basado en el paradigma de orientación a objetos y en el diseño de hipermedia. Para culminar, se realiza un proceso de pruebas del cliente para corroborar la funcionalidad del sistema generalmente y por cada uno de los módulos que lo comprenden.

Capítulo 1 Introducción

Tema

Desarrollo de un Sistema para la Gestión de Talleres Automotrices, Orientado a la Web y a Dispositivos Móviles. Caso de Estudio: Taller Automotriz AUTOTREK.

Razones que motivaron al estudio

En la actualidad existe una gran variedad de talleres automotrices a nivel nacional, sobre todo talleres pequeños que encontramos en cualquier ciudad. Se puede dar cuenta a simple vista que este tipo de talleres no cuentan con un sistema adecuado para poder gestionar su negocio, o simplemente la gran mayoría no lo tiene. Se puede enfocar en “mecánicas” pequeñas a las cuales acude un gran porcentaje de la población cuando tiene un desperfecto en su vehículo. Estos establecimientos no cuentan y/o no tienen acceso a un sistema que les facilite la correcta administración de su negocio, por ejemplo: administrar la entrada y salida vehicular, los desperfectos a corregir, posibles piezas que se van a reemplazar, datos personales de los clientes, el dinero que ingresa, entre otros aspectos sumamente importantes para el funcionamiento productivo de su negocio.

Otro punto de importancia es el orden que se debería mantener en estos talleres. En varias ocasiones los clientes no se encuentran satisfechos debido a la falta de cumplimiento por parte del taller, no se cumplen con los tiempos establecidos, existen descuidos de tareas y no se realizan, se generan discusiones entre los mismos clientes por su llegada, entre otros factores que además de afectar al mismo cliente, disminuyen la credibilidad, confiabilidad y calidad del trabajo realizado por el taller. Por lo que se necesita gestionar citas previas, orden de llegada de vehículos, fechas de entrega y más aspectos que ayudarán a mantener un negocio organizado y más eficiente.

En muchos casos en este tipo de talleres, el administrador del negocio es también mecánico, o cumple varias labores dentro del negocio, por lo que no puede estar todo el tiempo en la oficina, sino en varias partes del establecimiento. También es necesario que la persona encargada pueda tener completa accesibilidad al sistema en cualquier lugar ya sea de la mecánica como en lugares externos, esto sin contar los talleres que no tienen acceso a una computadora. Debido a esto, se implementará el sistema también para dispositivos móviles, lo que facilitará gestionar

las necesidades de los clientes de una manera más eficiente y cómoda sin preocuparse de los aspectos antes mencionados.

El sistema brindará un gran apoyo a la gestión del negocio, proporcionando mecanismos para su administración y organización, sin mencionar el aumento de satisfacción del cliente. También brindará una gran accesibilidad ya que, hoy en día un gran porcentaje de la población tiene acceso a un dispositivo móvil, lo que facilitará su portabilidad y comodidad.

Planteamiento del problema

Dado que los talleres automotrices necesitan de una gestión óptima para mejorar su productividad y mejorar su competitividad, se plantea realizar la implementación de este sistema, utilizando metodologías, técnicas y herramientas investigadas, que nos proporcionen la ayuda necesaria para el desarrollo. Los administradores y trabajadores de los talleres automotrices necesitan estar en constante movimiento, por lo que un aplicativo móvil va a ser de gran ayuda para manejar el sistema desde cualquier parte en la que se encuentren.

Los talleres automotrices pequeños, no poseen un sistema que contenga los módulos adecuados para solventar sus necesidades no solo administrativas sino también procesos que muchas veces son necesarios para un buen funcionamiento del negocio y experiencia con el cliente, por ejemplo: agendar citas, registrar repuestos, administrar fechas de entrega, entre otros. Debido a esto, el sistema proveerá de las funcionalidades adecuadas para lograr solucionar dichas falencias y facilitar y mejorar la gestión.

De igual manera muchas veces el negocio no prospera por la falta de constancia por parte del cliente, por lo que se necesitará de una funcionalidad que permita enviar recordatorios para que los clientes no pasen por alto las prevenciones que deben de realizarle a su automóvil.

Objetivos:

Objetivo general

Desarrollar un sistema para la gestión de talleres automotrices, orientado a la web y a dispositivos móviles. Caso de estudio: Taller Automotriz AUTOTREK.

Objetivos específicos

- Analizar los requerimientos del sistema utilizando diagramas que ayuden al entendimiento.

- Analizar el diseño de la base de datos tomando como base los requerimientos del sistema.
- Seleccionar las herramientas adecuadas para el desarrollo del sistema.
- Diseñar el sistema en cuanto al backend y el frontend.
- Implementar el sistema tomando en cuenta los análisis y diseños antes realizados.
- Realizar pruebas para corroborar el completo y buen funcionamiento del sistema.

Capítulo 2 Caso de estudio: Taller Automotriz AUTOTREK

En este capítulo se dará una breve introducción de lo que será esta disertación en sí, cómo la tecnología se ha convertido en la herramienta fundamental y como ha ido evolucionando con el tiempo. Se conocerá como la automatización de procesos mediante la creación de sistemas puede ayudar a los talleres automotrices. También se conocerá acerca del Taller Automotriz AUTOTREK, aspectos generales, a que se dedica la organización, que esperan lograr a futuro, los factores de influencia en su negocio, como se encuentra en la actualidad no solo desde una perspectiva tecnológica si no también internamente, tomando en cuenta las actividades que se realizan para que el negocio funcione. Se revisarán los puntos fuertes y los débiles que tiene el

taller para saber que se puede mejorar, dándole una clasificación a la problemática que existe y así poder tener un rumbo claro de lo que se va a realizar con la presente disertación.

Introducción

“La tecnología por sí sola no basta. También tenemos que poner el corazón” (Jane Goodall).

Desde tiempos remotos, el ser humano ha buscado la manera de facilitar sus labores diarias para disminuir el trabajo realizado y tener un mejor estilo de vida. Como hemos podido observar con el pasar de los años, la tecnología ha sido una herramienta primordial para poder cumplir con este objetivo ya que ha logrado facilitarnos la vida y más que esto, mejorarla de manera exponencial. La mayoría de las tecnologías que han surgido, lo han hecho como perfeccionamiento de la mente humana con el fin de lograr satisfacer las necesidades que nos rodean.

Con el nacimiento de la tecnología, aparecen las computadoras, máquinas que se convierten rápidamente en una herramienta necesaria en la vida diaria del ser humano, esto gracias a su completo beneficio y rapidez para cumplir con tareas que antes podían tardar mucho y ser complejas. Hoy en día la utilidad de las computadoras es extremadamente importante para poder sobrellevar los negocios, estudios, trabajo y todos los campos en los que se desenvuelva el hombre y la industria.

Después de la llegada de las computadoras, empieza el surgimiento de las Tecnologías de la Información, que ayudan al hombre a transmitir, almacenar y procesar información mediante el uso de la tecnología. Lo que se ha buscado es automatizar procesos que realizaban los humanos para que puedan ser más eficientes y exactos. Esta innovación ha sido utilizada en los últimos años por la mayoría de las empresas, industrias, entidades, entre otros. Se ha podido observar un crecimiento relativamente alto en cuanto a producción y facilidad para los empleadores. Facturación, contabilidad, registro de datos, etc., son actividades que antes se realizaban a mano con gran dificultad y no muy buena efectividad, pero con las Tecnologías de la Información todo esto se ha logrado automatizar para cumplir con el propósito del ser humano, facilitar y mejorar su estilo de vida.

Por otro lado, según estadísticas del INEC, en el año 2015 se registraron 1'925,368 autos en el Ecuador, una cifra muy alta considerando el volumen de población que tiene el país. Muchos de los dueños de estos vehículos no tienen la facilidad para ingresar su auto a un taller de

prestigio o a las concesionarias, ya que el precio es relativamente elevado. Por esta razón, prefieren ingresar sus autos en talleres pequeños multimarca, donde se les ofrece un buen servicio a un precio moderado. En el Ecuador existen muchos de estos talleres mejor nombrados como “mecánicas”, y a pesar de la capacidad que poseen en cuanto a su área, muchas veces se manejan aún de la manera tradicional, facturas en hojas, pedidos anotados en papel, etc. Estos factores influyen mucho en la administración del taller y dan un gran espacio para errores.

El taller automotriz AUTOTREK forma parte de este tipo de talleres, por lo que se ha visto en la necesidad de una herramienta que le facilite la gestión para así poder seguir a flote y mejorar su negocio. Varios de los procesos que realiza el taller en cuanto a la administración y otros factores se los lleva en hojas de Excel, que al comienzo fueron útiles, pero conforme el negocio ha ido avanzando, se han empezado a volver no tan eficaces para el negocio.

Debido a esto, el presente trabajo de disertación busca crear un sistema orientado a los talleres automotrices para su gestión, pero centrándonos en las necesidades del taller AUTOTREK. De esta manera podrá tener una superioridad comercial y una mayor productividad, así como también más facilidad para sus empleados y una brecha estrecha para errores que puedan surgir. Esto gracias a la aplicación de las tecnologías de la información llevadas a este negocio y personalizadas para satisfacer sus demandas.

Situación de la entidad

Información del taller.

En la ciudad de Quito, con el crecimiento acelerado del parque automotriz, la necesidad de aumentar empresas que se dediquen a la suplir las necesidades de mantenimiento, reparación, cuidado, lavado de autos es imprescindible y un nicho de mercado que a pesar de ser muy amplio nunca está debidamente atendido.

AUTOTREK nace de la idea de un Ingeniero Mecánico que después de realizar un estudio de factibilidad en el sector norte de la capital; la parroquia de Calderón-Carapungo, determina que la población que actualmente supera los 300.000 habitantes al menos un 30%, de esas familias tienen un vehículo. Por lo que montar una mecánica automotriz integral multimarca, representaría un negocio viable que en poco tiempo supliría con creces las expectativas de su propietario.

La empresa inició actividades en enero del presente año, al momento con 6 meses de funcionamiento, está en el proceso de maduración del negocio, con una ocupación del 45%. La etapa de arranque que es de una forma la más dura de una empresa está ya concluyendo, las actividades de publicidad en redes sociales y medios físicos están rindiendo sus frutos. Se espera que hasta finalizar el año se llegue a una ocupación de al menos el 60%. En la Figura 1 se puede visualizar el logo del taller.



Figura 1 Logo del Taller Automotriz AUTOTREK.

Fuente: Taller Automotriz AUTOTREK.

Por otro lado, en los que se refiere a la gestión del taller, los responsables llevan registro del ingreso y salida de vehículos, facturas, notas de entrega, entre otros documentos que les ayuda a mantener el taller organizado y así poder tener un control de éste. Varios de estos documentos se manejan por medio de hojas de Excel y hojas físicas. Las citas agendan por vía telefónica al igual que las entregas de vehículos. De esta manera el taller busca un orden y legalidad en todas las actividades que se realizan dentro de él para poder al final de ciertos períodos de tiempo tener una base de información que les sirva como herramienta de gestión.

Visión.

Empoderarse del mercado del sector norte de la capital, llegando a ser más que un taller automotriz, una marca que proporcione tranquilidad a sus clientes satisfaciendo cada una de sus necesidades y sobre todo que brinde seguridad a sus familias, conociendo que sus vehículos están totalmente respaldados por los servicios que brinda AUTOTREK, basados en nuestros conceptos de calidad, profesionalismo y eficiencia.

Misión.

Brindar calidad en cada uno de nuestros servicios, atendiendo las necesidades de nuestros clientes de forma rápida y oportuna. Estamos conscientes que su vehículo transporta vidas y nosotros somos responsables de su seguridad.

Problemática

Debido a que el taller automotriz AUTOTREK es relativamente nuevo y está en su etapa de crecimiento como negocio rentable, no posee aún las herramientas adecuadas para una gestión óptima que le permita mejorar sus procesos y al mismo tiempo facilitarlos para el personal. De igual manera se podría mejorar su productividad, eficacia y eficiencia implementando tecnologías que ayuden con todo el negocio.

Administración.

Como ya se observó anteriormente, muchos de los documentos que maneja el taller para llevar un registro formal de las actividades realizadas, aún son físicos, lo que en la actualidad ya no es factible y se considera un poco obsoleto. Mientras que los demás que están llevados en hojas de Excel carecen de organización y tienen mucha apertura a fallos humanos, por esta razón el taller ha tenido algunos problemas en cuanto al dinero que ingresa, registros de autos, tiempos de respuesta, etc.

Orden de prioridad.

De una manera similar, han existido algunos inconvenientes en cuanto al orden que se maneja con los autos que ingresan y su salida. Muchas veces por la falta de un registro y orden, no se ha respetado la prioridad de los clientes que llegan antes que otros, por lo que los técnicos despachan los autos que ya han culminado sin control, ocasionando insatisfacciones en los clientes y varios reclamos. Esto además de perjudicar los ingresos del taller, representa un daño al prestigio, confianza y seguridad que tienen los clientes.

Movilidad.

Otro punto importante que tomar en cuenta es la accesibilidad al sistema. El área de trabajo del taller AUTOTREK es relativamente amplia con relación a los empleados que laboran en este lugar, y la mayoría de las veces se necesita tomar el registro de los diferentes tipos de datos

mientras el vehículo está siendo revisado previamente a su reparación y/o mantenimiento, por lo que se dificulta el hecho de utilizar el sistema en una computadora ya que por lo general se encuentran en las oficinas o lugares que no estén a la intemperie. Dichas complicaciones crean la necesidad de una aplicación para dispositivos móviles que facilite las actividades antes mencionadas.

Notificaciones.

El taller AUTOTREK tiene como uno de sus objetivos primordiales la satisfacción a sus clientes frecuentes no solo mientras se utilizan sus servicios, sino también recordándoles el mantenimiento que deben realizarles a sus vehículos posterior a la visita que ya han realizado. Muchos de estos recordatorios son manejados vía telefónica por el administrador, lo que representa un gran esfuerzo y tiempo perdido para el negocio. Siendo otra necesidad una herramienta para gestionar estas notificaciones automatizándolas.

Procesos

Para que el taller automotriz AUTOTREK esté en funcionamiento necesita de algunos procesos específicos que ya han sido definidos para poder realizar las actividades principales. Permiten que las actividades del taller tengan un cierto orden y no existan grandes confusiones en cuanto a cómo se maneja el negocio.

Como el negocio es un tanto temprano en cuanto a su creación, no posee una gran cantidad de procesos, además son muy generales y se puede observar que algunos podrían mejorar mediante la implementación de las tecnologías de la información, de esta manera los involucrados tendrán mayor facilidad para realizar sus tareas y una mejor experiencia.

A continuación, se definen los procesos con los que cuenta el taller.

Ingreso de Vehículos.

Se puede decir que es proceso fundamental del taller, ya que conlleva el negocio en sí. Este proceso consiste en todo lo que se realiza desde que un cliente ingresa al taller, hasta que sale una vez que se le hayan corregidos los desperfectos a su vehículo. Como se ha podido observar sigue una cadena de actividades distintas pero que en conjunto conforman la cadena que se debe seguir para que un vehículo sea reparado. Esto quiere decir que aparte de las actividades

técnicas que se realizan al vehículo en sí, se encuentran también actividades de facturación, de cotización, entre otras.

En la Figura 2 se presenta el diagrama de actividades del proceso de Ingreso de Vehículos.

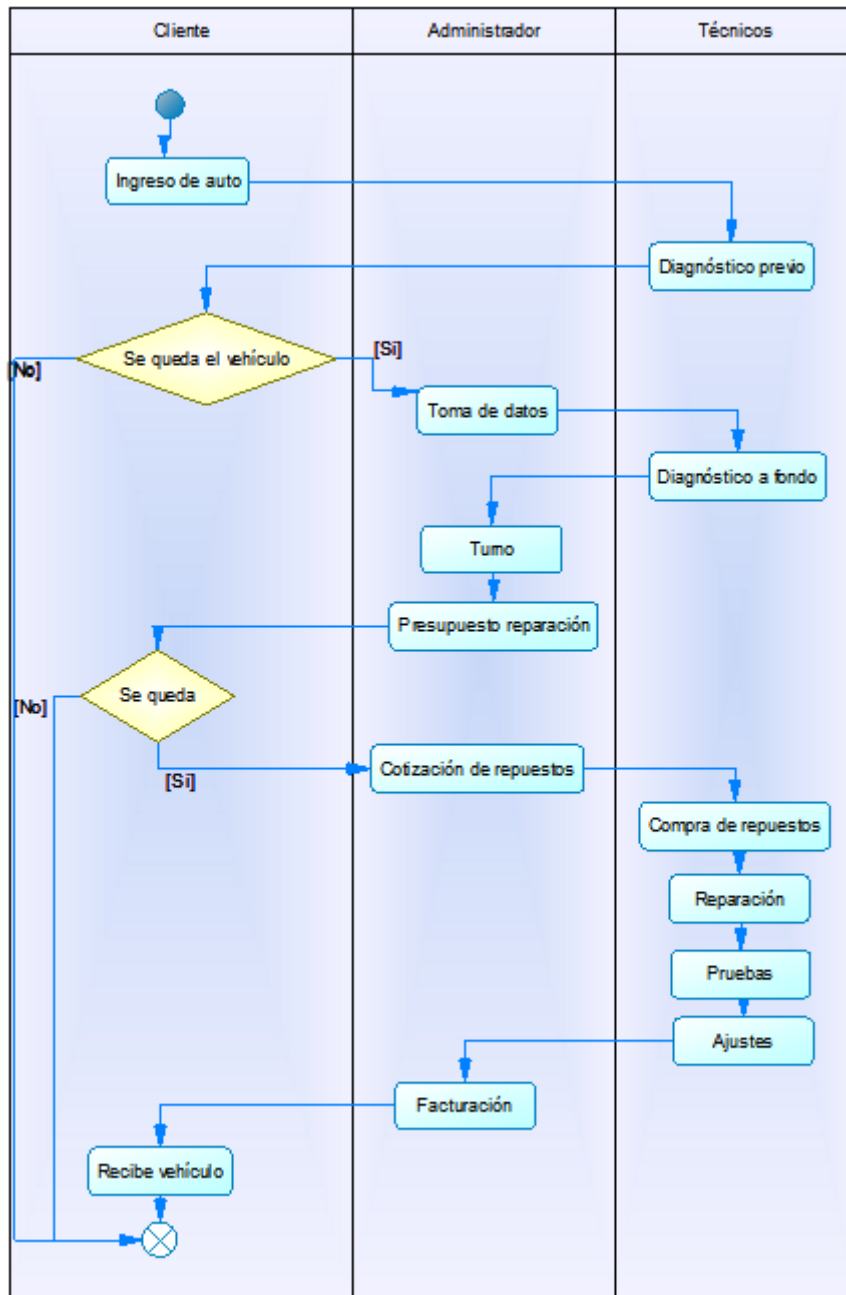


Figura 2 Diagrama de actividades de Ingreso de Vehículos.

Fuente: Propia.

Asignación de Cita.

Como en cualquier negocio, los clientes son parte fundamental por no decir lo más importante, siempre su comodidad y confianza debería ser la mayor prioridad. Este proceso tiene el fin de establecer un orden en cuanto a los clientes para de esta manera brindar un mejor servicio y satisfacer a los clientes dándoles la confianza que necesitan con respecto a los tiempos de entrega de sus vehículos. Cada cliente tiene una prioridad dependiendo el orden en el que llega y esto se logra mediante la asignación de citas, para que de esta manera los técnicos puedan atender a cada cliente de una manera organizada y se cumpla con las necesidades de estos.

En la Figura 3 se presenta el diagrama de actividades del proceso de Asignación de Cita.

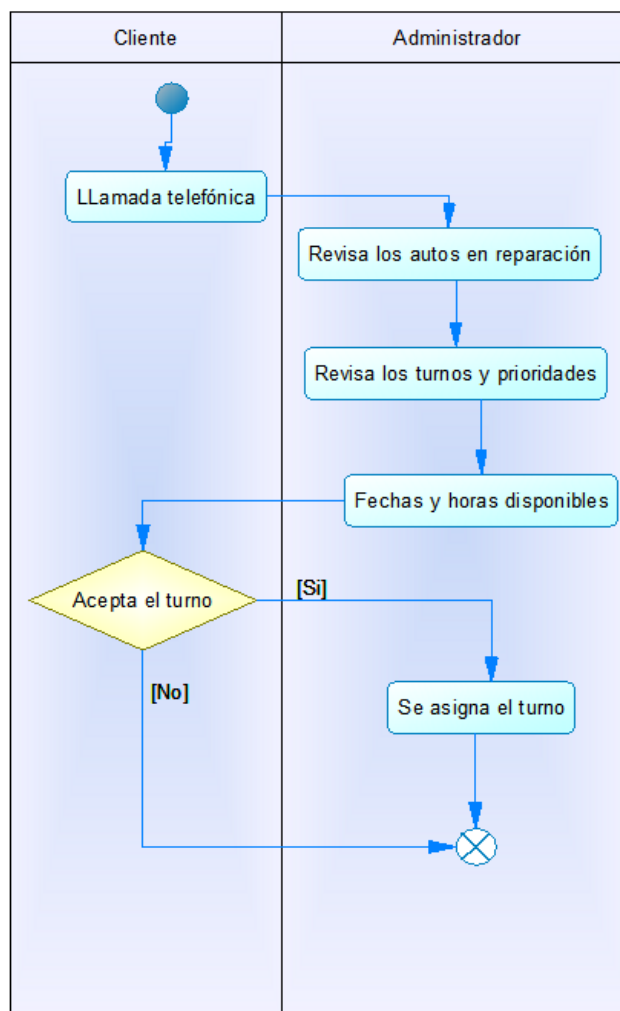


Figura 3 Diagrama de actividades de Asignación de Cita.

Fuente: Propia.

Post Venta.

Como se habló anteriormente, la mayor prioridad del taller es la satisfacción y la confianza que se le brinda al cliente, por lo que su punto de vista del servicio que le fue brindado es muy importante para poder reconocer los factores en los que se debe mejorar. Este proceso aparte de buscar una retroalimentación también recuerda a los clientes los mantenimientos que se le deben realizar al vehículo después. Las fechas de cambios de aceite, reparaciones que se deben realizar después de la visita al taller, etc.

Este es un proceso muy importante para que los clientes regresen al taller y se conviertan en frecuentes, de esta manera el negocio crecerá y se aumentará la confianza y satisfacción de los clientes.

En la Figura 4 se presenta el diagrama de actividades del proceso de Post Venta.

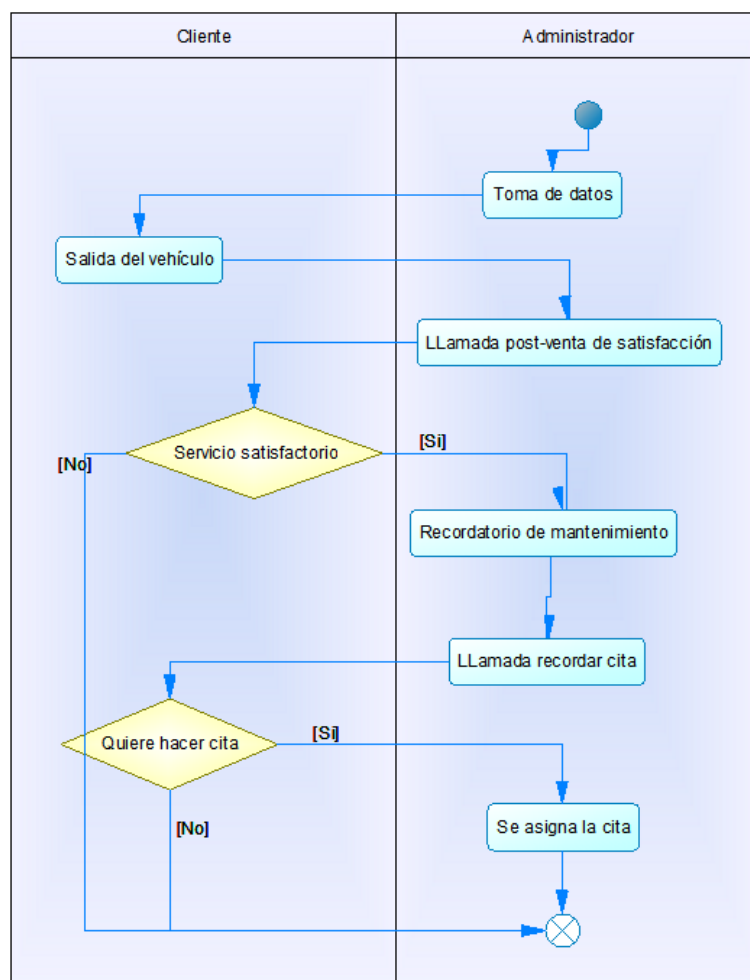


Figura 4 Diagrama de actividades de Post Venta.

Fuente: Propia.

Capítulo 3 Marco Teórico

En este capítulo se recopila la información necesaria para el desarrollo del sistema producto de la disertación. Toda la información que se revisará a continuación promueve una base para el entendimiento del proceso que se llevará a cabo con la aplicación en cuestión.

Se comenzará con conceptos básicos acerca de la historia en la que se basa la aplicación, continuando con el funcionamiento previsto para la misma. Seguido de las tecnologías que se van a utilizar tomando en cuenta el avance de los últimos años.

También se explicará la metodología que será utilizada en el proyecto y que brindará la dirección que la aplicación deberá tomar conforme vaya avanzando en el proceso de desarrollo.

Por otra parte, se hablará acerca de las herramientas que serán imprescindibles para lograr un éxito en el desarrollo de la aplicación y que han sido seleccionadas por su acoplamiento con el tipo de ambiente en el que se va a trabajar.

Aplicaciones web

Historia de la WWW e Internet.

Timothy John Berners-Lee fue la primera persona que conectó dos ordenadores para que puedan comunicarse entre sí, por lo que se lo podría considerar como el padre de la web.

Las computadoras al igual que los humanos necesitan un sistema para poder comunicarse. Dos personas se comunican mediante señas, gestos, el habla, etc. Necesitan también un protocolo que les indique quien debe hablar primero y quien responde, turnos de comunicación para que sea ordenada y no existan confusiones. Por último, necesitan saber dónde está el otro, ya que si se desconoce la ubicación no se podrá realizar la comunicación entre ambos. En las computadoras el caso es similar, para que dos computadoras se comuniquen necesitan un lenguaje (HTML), necesitan un protocolo de comunicación ordenado (HTTP) y necesitan saber la ubicación de la otra (URL). Toda esta estructura fue creada por Timothy John Berners-Lee mientras realizaba una investigación con su equipo. Se vio en la necesidad de intercambiar información de manera efectiva, creando la WWW (World Wide Web). En la Figura 5 se presentan los 3 elementos clave en el nacimiento de la web y cómo interactúan entre sí.

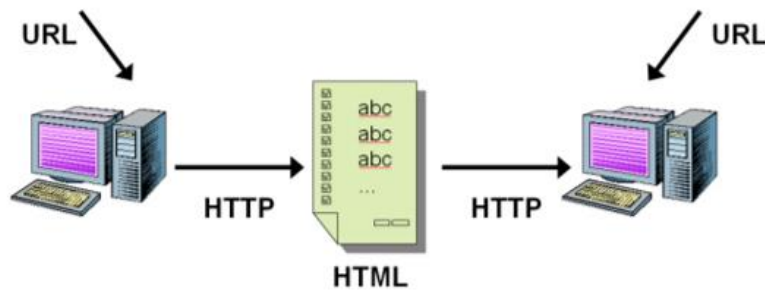


Figura 5 Los pilares de la Web.

Fuente: (Mora, 2002, pág. 16)

Se ha dicho que la guerra ha contribuido a desarrollar invenciones que luego resultaron útiles para la Humanidad. En el caso de la Guerra Fría (el periodo de tensiones entre Estados Unidos y el bloque soviético que siguió a la 2ª Guerra Mundial) es bastante cierto. (Millán, 2006, pág. 1)

La información era vital en los tiempos en los que se libraba la guerra a una escala atómica, nacía la necesidad de un sistema de comunicaciones que no fuera afectado por los conflictos que existían en la época, y la información se mantuviera a salvo a pesar de la guerra. La solución que se halló fue crear una red de ordenadores que tuvieran un valor idéntico, de manera que si uno de ellos se desconectaba no afectara el intercambio de información. Esto dio cabida a varios proyectos en los años 60, involucrando importantes organizaciones como El Pentágono, el Instituto de Tecnología de Massachussets, entre otros.

Financiada por El Pentágono, surgió la red ARPANET, la que constituía una de las primeras formas de lo que hoy en día es la Internet. Por otro lado, otros equipos se desarrollaban en otros campos como acceso a almacenes de datos, la creación del ratón y el trabajo en colaboración a través del Hipertexto, dando lugar a algo más grande que la comunicación entre ordenadores, la comunicación entre personas. Esto fue el inicio de lo que sería una conglomeración de investigaciones para formar una red global de comunicaciones en la que participaría todo el mundo.

La red ARPANET desaparece en 1989, cuando fue separada la parte militar de la civil en la red, pero varias instituciones ya habían creado sus propias redes para la comunicación interna

superando los 100.000 servidores ese mismo año. Timothy Berners-Lee propuso un sistema de hipertexto compartido creando las bases de lo que era la WWW.

Con la aparición y masificación de los ordenadores personales, el pacto de los protocolos de comunicación y el lanzamiento del primer navegador, se creaba la Internet, una red de comunicación que hoy en día constituye una de las herramientas más populares y utilizadas a nivel mundial. Pronto se extendió a nivel mundial y se posicionó como la red más utilizada globalmente, formando hoy en día parte fundamental de la mayoría de los negocios, entidades, organizaciones, comunicaciones, etc.

Hoy en día la web y la Internet son conceptos que casi todo el mundo ha escuchado, la Internet como la red global más grande y la web como la colección de páginas que está asentada sobre la Internet. Cuando se navega en el ordenador o smartphone, se utiliza la red de Internet para acceder a la web.

Redes de comunicación.

Se puede considerar a una red como la interconexión de un número determinado de ordenadores mediante distintos dispositivos que proporcionan el envío y la recepción de información entre éstos, formando un conjunto organizado.

Estas redes permiten no solo el intercambio de información, sino que constituyen una herramienta sumamente importante en comunicaciones de todo tipo. Por ejemplo, con una red se puede administrar las comunicaciones internas, controlar periféricos como impresoras, escáneres, etc. Muchas de las organizaciones manejan este tipo de sistema de enjambre para el procesamiento de datos y procesos de administración.

Ya que no siempre se tendrá un número de ordenadores fijos, se han clasificado a las redes dependiendo de su dimensión o el área que ocupan.

Redes de Área Local (LAN). Sus siglas en inglés significan Local Area Network, constituyen las redes que no sobrepasan los kilómetros de distancia, en ellas no existe una gran cantidad de ordenadores conectados y son las redes de menor envergadura. Si un ordenador o un smartphone está conectado a una red, ya se considera una red LAN. En la Figura 6 se puede apreciar el esquema de una red de área local.

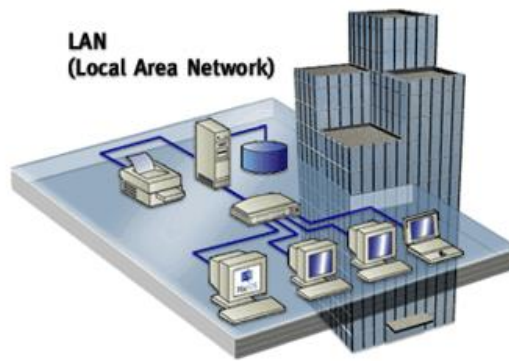


Figura 6 Esquema de una Red LAN.

Fuente: (Lepre)

Redes de Área Metropolitana (MAN). Sus siglas en inglés significan Metropolitan Area Network. Estas redes son de tamaño mediano, se las puede considerar como un conjunto de redes LAN dispersas en un área de extensión grande. Un campus universitario, una porción de ciudad son consideradas como redes MAN. En la Figura 7 se puede apreciar el esquema de una red de área metropolitana.

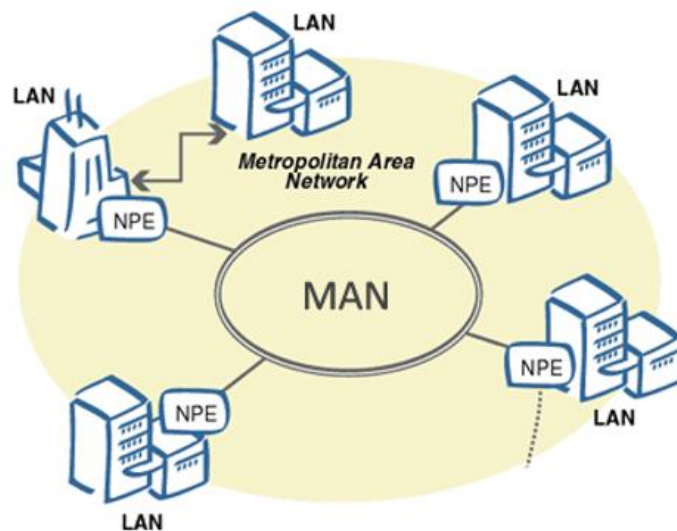


Figura 7 Esquema de una Red MAN.

Fuente: (Carrascal)

Redes de Área Amplia (WAN). Sus siglas en inglés significan Wide Area Network. Estas redes abarcan grandes extensiones de terreno y un número gigantesco de ordenadores, se refieren a redes que conglomeran ciudades, países, incluso continentes enteros. Esto quiere

decir que los usuarios que utilizan esta red no se encuentran en una misma ubicación física. El Internet es un claro ejemplo de una red WAN, ya que abarca todos los continentes del planeta. En la Figura 8 se puede apreciar el esquema de una red de área amplia.

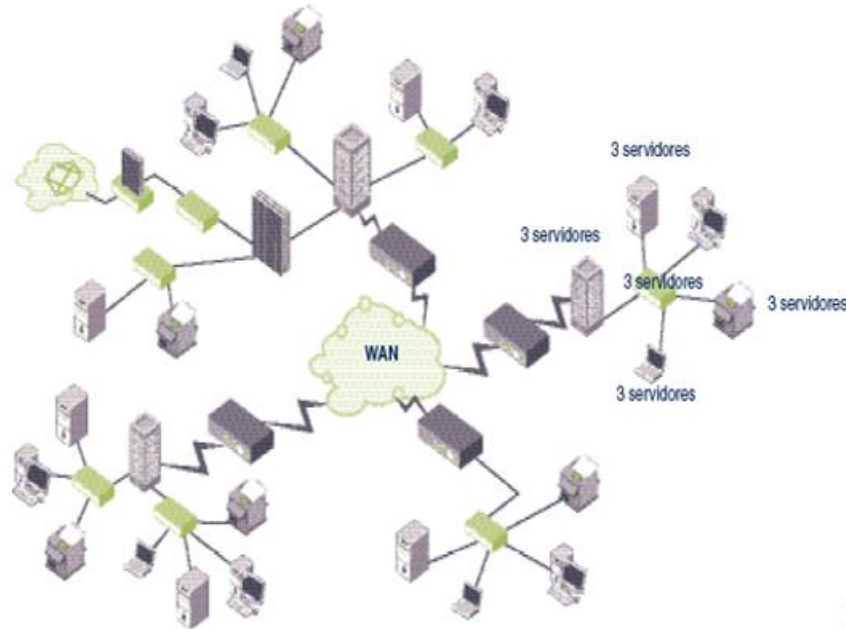


Figura 8 Ejemplos de diagramas de red WAN con varias sedes sociales.

Fuente: (McGraw, 2008)

Muchas WAN son construidas por organizaciones o empresas para su uso privado, otras son instaladas por los proveedores de Internet (ISP) para proveer conexión a sus clientes, por ejemplo, Telefónica, Jazztel, Orange, Ono, etc. Son empresas (ISP – Internet Services Provider) por las que nos conectamos desde nuestra red LAN (nuestro router, PC, Smartphone y Tablet) a la WAN y a su vez a Internet. (Castañeda, 2016, pág. 27)

Sitios web.

Con el surgimiento de la Internet o también nombrada web, surgió una nueva manera de obtener, observar, conseguir, etc. Información, solo hay que imaginar el simple hecho de poder presentar y acceder a toneladas de información desde la comodidad del hogar, fue un cambio radical pero aún se podía hacer más. “Inicialmente la web era simplemente una colección de páginas estáticas, documentos, etc., que podían consultarse o descargarse.” (Mateu, 2004, pág. 20). Pero con el avance del tiempo surgieron nuevas tecnologías que ampliaron el campo de desarrollo de la web.

Al comienzo, cuando surgió la web, se empezaron a desarrollar páginas que tenían limitaciones debido a la tecnología del tiempo, pero con la evolución se han llegado a clasificar a las páginas por generaciones, debido a las diferencias notorias que existen por las tecnologías nuevas que se fueron implementando.

Primera generación (1992-1994). Constituían páginas escritas, casi no había imágenes y peor aún contenidos multimedia. Eran desorganizadas, como que se hubieran escrito en papel, la navegación era muy poco estructurada y poseían listas interminables de enlaces a diferentes sitios web. En esta época existía muy poco ancho de banda por ello las páginas no podían ser pesadas y eran estáticas.

Segunda Generación (1995-Actualidad). La principal diferencia de esta generación y la primera es la incorporación de elementos gráficos, lo cual fue una enorme novedad en la época. Se emplean colores de fondo e incluso imágenes, se llegan a utilizar íconos en vez de palabras, se emplean tablas para tabular datos. Debido al contenido y las páginas tenían una carga lenta y a pesar de tener una jerarquía en su contenido aún no existía el concepto de navegación. Estas páginas aún siguen siendo estáticas. Los creadores se centraban en la presentación de la página mas no en el contenido, por esto la lentitud de carga.

Tercera generación (1996-Actualidad). Estas páginas constituyen las más utilizadas en la actualidad, los creadores de las páginas le dan mayor importancia al contenido que a la presentación, minimizando el tiempo de carga utilizando pocos recursos gráficos y optimizando el código HTML. La información se puede visualizar completamente en la pantalla para no desplazarse, se crean los sitios web tomando en cuenta para qué se necesitan, esto quiere decir que estaban enfocados a las ventas, informar, ofrecer servicios, etc. Se emplean estrategias visuales para seducir al usuario creando una experiencia en todo el sitio web. Se emplean las páginas dinámicas gracias a la navegación y se empiezan a utilizar tecnologías como CSS.

Cuarta generación (1999-Actualidad). Esta generación marca una pequeña diferencia gracias al ancho de banda que empieza a existir, el cual permite el uso de medios más atractivos. Los recursos gráficos vuelven a ser el punto principal de estas páginas, se extiende el uso de tecnologías como CSS, DHTML, Macromedia Flash Player, entre otras. Se enfocan en una mejor experiencia al usuario desde la primera hasta la última página que sea visitada. Aparece el streaming de audio y video en tiempo real.

Todas estas generaciones de sitios web han ido evolucionando, pero a pesar de la tecnología que se haya implementado, aún siguen siendo sitios que nos brindan información, gráficos, tal vez videos, etc. Contenido para que el usuario pueda observar. No se puede hacer mayor interacción con estos como ingresar datos, consultar información de una base de datos, etc.

Aplicaciones Web.

Con la evolución de los sitios web, viene la incorporación de CGI (Common Gateway Interface), que era un método que ayudaba a solicitar datos de un programa o aplicación que se ejecutaba en un servidor web, permitiendo crear contenido dinámico para los sitios web. Debido a la sencillez que poseen los CGI siguen siendo utilizados en la actualidad, esto gracias a que la mayoría de los servidores web pueden soportarlos. Se lo puede considerar como un sistema de comunicación para envío y recepción de datos, esto quiere decir que un cliente realiza una petición mediante un *request*, el servidor recibe la petición, la lee y envía la respuesta. A toda esta labor de compartir información, interoperabilidad, diseño concentrado en experiencia para el usuario y colaboración con la WWW, se le considera como una aplicación web. Resumiendo, una aplicación web consiste en el software que reside en un ordenador considerado como servidor web, al cual los usuarios pueden acceder mediante un navegador web, utilizando una intranet o el mismo Internet para poder consumir sus servicios.

Al parecer el sistema era bueno, pero tenía un punto débil, cada vez que se recibía una petición, el servidor web lanzaba un proceso que ejecutaba el programa CGI. Como, por otro lado, la mayoría de CGI estaban escritos en algún lenguaje interpretado (Perl, Python, etc.) o en algún lenguaje que requería run-time environment (VisualBasic, Java, etc.), esto implicaba una gran carga para la máquina del servidor. (Mateu, 2004, pág. 20)

Buscando soluciones para este problema de rendimiento, surgen dos alternativas principales, la primera consistía en el diseño de sistemas más integrados con el servidor para evitar la instancia y ejecución de varios programas, mientras que la segunda radicaba en dar al servidor un intérprete de lenguaje de programación para incluir las páginas en el código y así sean ejecutadas por el mismo servidor. De esta manera el tiempo de respuesta se reducía considerablemente optimizando el rendimiento. En la Figura 9 se puede apreciar cómo se conectan las diferentes partes de una aplicación web.

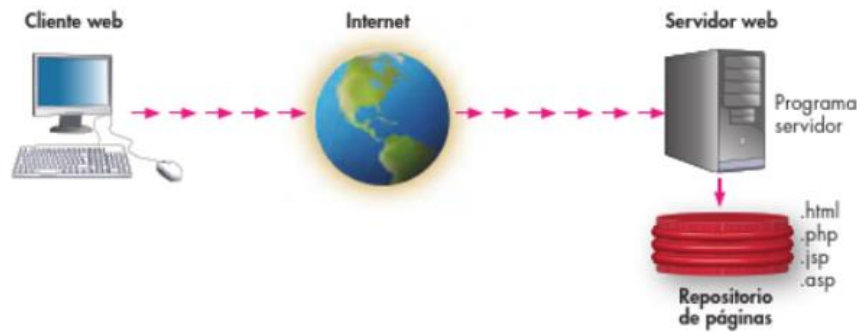


Figura 9 Esquema básico del servicio web.

Fuente: (Raül V. Lerma-Blasco, 2013, pág. 11)

A partir de estos conceptos, empieza un boom referente a arquitecturas y lenguajes de programación para desarrollo de aplicaciones web, pero cabe recalcar que todas estas siguen alguna de las dos alternativas descritas anteriormente, aunque la mayoría optó como mejor opción la segunda. La mayoría de las arquitecturas más útiles son aquellas que poseen un lenguaje de programación integrado para que el servidor pueda interpretar comandos incrustados en las páginas HTML, y también un sistema para ejecutar programas que esté más enlazado con el servidor y así no tenga dificultades de rendimiento.

Como se habló anteriormente, la arquitectura de una aplicación web básicamente consta de ordenadores conectados, mediante el esquema cliente-servidor, a una red, ya sea una red interna como una intranet corporativa o a la Internet. Este tipo de arquitectura se mantiene por roles, el cliente y el servidor. Si existe un proceso que solicita servicios es el denominado cliente. Por otro lado, aquel proceso que responde a estas peticiones es el servidor. Estos dos tipos de procesos pueden ser ejecutados en el mismo equipo o en diferentes, creando aplicaciones distribuidas facilitando la separación de funciones según su servicio.

Este tipo de arquitecturas se manejan separadas en tres capas fundamentales para el correcto funcionamiento.

Lógica de presentación. Se podría decir que esta capa consiste en la vista de la aplicación, todo lo que puede ver el usuario, contiene la interfaz mediante el usuario puede interactuar, presenta las salidas que se hayan solicitado. Captura las acciones realizadas por el cliente para obtener la información, luego envía esta información a la capa de lógica de negocio para que sea procesada y luego presenta la información que fue devuelta al usuario.

Lógica de negocio. Esta capa es el puente que existe entre el cliente y los datos brutos. Se encarga de recibir la entrada que envió la capa de presentación para que luego se puedan aplicar las reglas del negocio, es decir gestionar las funcionalidades que se esperan de la aplicación o sistema (facturación, cálculos, controles, etc.) interactuando con la lógica de datos para retornar la respuesta pedida. Esta capa se puede programar tanto en el entorno del cliente como del servidor.

Lógica de datos. En esta capa se encuentran los datos que utiliza la aplicación o sistema. Se encarga de mantener los datos, recuperarlos, almacenarlos y asegurar su integridad. Por lo general está formada por los gestores de bases de datos con los que se trabaja para poder enviar la información a la lógica de negocio.

En la Figura 10 se presenta la relación entre las capas que existen para que exista una correcta comunicación.

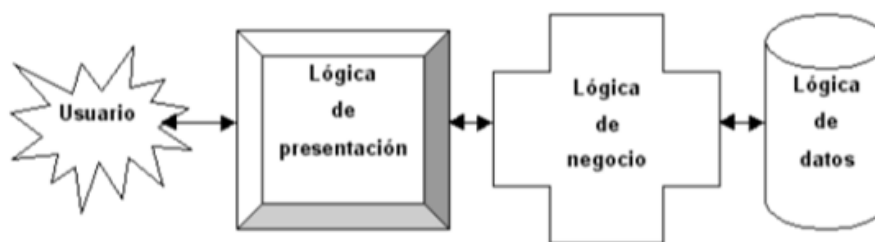


Figura 10 Separación de funciones.

Fuente: (Mora, 2002, pág. 42)

Sistema Cliente-Servidor

Como ya se habló anteriormente, el entorno cliente servidor es utilizado para el desarrollo de aplicaciones web, pero a continuación se explicará con mayor detalle a que se refiere en sí este sistema.

Cuando se hace referencia a la parte del cliente (también llamada front-end), consiste en la interfaz entre el usuario y la aplicación, es aquella que maneja la interacción con el usuario, en pocas palabras es todo lo que puede visualizar el usuario y todo con lo que se puede comunicar con el sistema. Por otro lado, cuando se habla de la parte del servidor (también llamada back-end), hace referencia al servidor, todos los recursos compartidos que utiliza la aplicación como las bases de datos, módems, impresoras, etc.

A pesar de ser partes complementarias, cada una de estas necesitan diferentes recursos para su correcto funcionamiento, por ejemplo, la memoria, velocidad, capacidad de discos duros, dispositivos, etc. (Mora, 2002, pág. 44). Por lo que ya es de imaginarse el desarrollar el front-end y back-end por separado teniendo en cuenta las herramientas óptimas para cada uno dependiendo de sus necesidades. El ambiente en el que se desarrollan estas aplicaciones suele ser heterogéneo y utilizando software de distintos fabricantes. También se debe tomar en cuenta que generalmente el hardware y sistema operativo del front-end y back-end suelen no ser los mismos por lo que la comunicación entre estas dos partes se realiza mediante una interfaz de programación de aplicaciones (API o Application Program Interface) y una Llamada a Procedimiento Remoto (RPC o Remote Procedure Call). Estos conceptos se abordarán más adelante.

Entonces gracias a la información anterior, se puede definir a una aplicación web como un tipo de sistema cliente-servidor el cual es accesible desde cualquier navegador, ya sea a través de Internet o red local. Esto quiere decir que no necesitan ningún tipo de instalación, y su potencia no depende del dispositivo mediante el cual se accede por lo que a pesar de no tener una supercomputadora se obtendrá una excelente experiencia, todo lo que consta en sí de la aplicación se encuentra en el servidor donde reside.

Ventajas

- Una de las principales ventajas y por la que este tipo de aplicaciones son muy usadas en la actualidad es gracias al hecho de que no necesitan de instalación, debido a que se accede a ellas mediante un navegador.
- El hecho de ser multiplataforma y multidispositivo es otra de las grandes ventajas de este tipo de aplicaciones, gracias a esto no se debe preocupar del software y dispositivo por el que se accede.
- Como ya se habló anteriormente, la potencia de las aplicaciones reside en el servidor mas no en el ordenador que se tenga.
- Estas aplicaciones pueden residir en la nube, por lo que pueden ser accesibles mediante internet por medio de cualquier ordenador.
- Son muy adaptables, se pueden actualizar con gran facilidad y son muy intuitivas visualmente para los usuarios.
- Otra de las más grandes ventajas de las aplicaciones web es el precio. Debido a su tiempo de desarrollo estas aplicaciones tienen precios más bajos que las demás.

Desventajas

- Las aplicaciones web deben estar programadas para poder ser ejecutadas en los diferentes dispositivos, esto quiere decir que tiene que ser responsive o si no se tendrán problemas al momento de visualizarlas.
- El hecho de que sean web implica que el ahorro en precio que se logre obtener se lo debe invertir en mejorar la página web, no sirve de nada una aplicación web si su front-end no es el apropiado.
- Se necesita obligatoriamente una conexión a la red para que se pueda utilizar, lo que equivale una gran desventaja en lugares donde no existe o no se puede conectar.
- Debido a que no se encuentra en tiendas de aplicaciones, no es tan visible para las personas que no la conocen.

El cliente

Se considera al cliente web como un programa que sirve para que el usuario pueda interactuar y así se solicite a un servidor web el envío de información mediante HTTP. Esta parte de las aplicaciones web suele estar programada en código HTML y código ejecutable hecho en lenguaje script para que interprete el navegador. A parte de estos lenguajes también se suelen construir con pequeños applets construidas en java y plugin para que se pueda visualizar multimedia. Estos dos últimos recursos ya son considerados un poco antiguos y suelen presentar problemas de compatibilidad con las aplicaciones actuales. En resumen, el cliente lo que hace es interpretar las páginas HTML y los recursos que se utilicen.

El servidor

Este es un programa que se encuentra escuchando solicitudes de conexión mediante el protocolo HTTP por parte de los clientes web. Esta escucha de peticiones es continua, permanente, y difiere del sistema operativo para el método en la que se realiza.

El servidor está formado por páginas estáticas que muestran la misma información, recursos como multimedia, documentos, etc. Que están en la capacidad de ser descargados y ejecutados en el lado del cliente para su visualización. También existen programas que se ejecutan por el servidor web en los que su respuesta suele ser una página en HTML enviada al navegador en el que se visualiza la aplicación.

Existen varias tecnologías que se utilizan para el desarrollo de servidores web, todas ellas son muy similares y están basadas en los mismos principios ofreciendo las prestaciones y resultados

muy similares. (Mora, 2002, pág. 50). En la Figura 11 se puede apreciar la relación entre el cliente y el servidor.

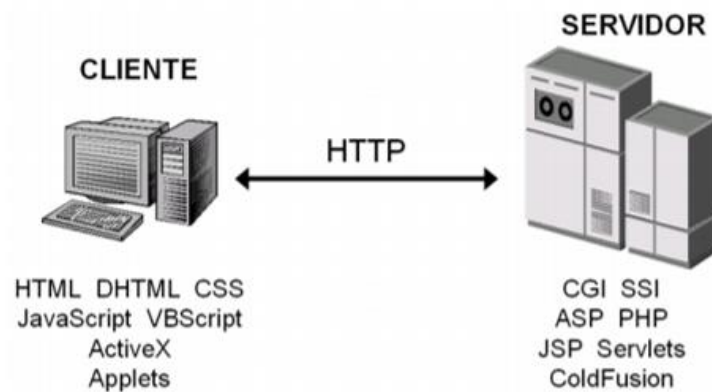


Figura 11 Tecnologías empleadas en el cliente y en el servidor web.

Fuente: (Mora, 2002, pág. 51)

Aplicaciones para Dispositivos Móviles

Historia de los Dispositivos Móviles.

Los equipos móviles ya son parte de la vida de las personas que tienen uno, y son cada vez más sofisticados, ya que son más poderosos y permiten realizar tareas inimaginables hace pocos años. (Lisandro Delía, pág. 766)

Con el pasar de los tiempos y el surgimiento y gran éxito de las computadoras personales, la tecnología necesitaba evolucionar, computadores más potentes, mejores prestaciones, mayor performance, movilidad, tamaño y potencia. Estos factores no solo han cambiado a la tecnología en sí, sino también los hábitos de consumo de las personas y las rutinas a las que estaban acostumbradas.

Esta revolución tan solo tiene 20 años, viene a darse desde la voz transmitida, hasta la información inalámbrica, convirtiendo a los dispositivos móviles en una herramienta fundamental y necesaria hoy en día.

Toda esta historia se remonta en 1860, cuando el matemático James Clerk Maxwell, quien predijo la propagación de las ondas electromagnéticas a la velocidad de la luz. Después de 20

años se logró confirmar esta teoría y a lo largo de otros 20 años se creó la primera aplicación móvil.

Como se ha podido observar, la evolución viene desde grandes e incómodos celulares que tenían baterías que duraban una hora y eran muy costosos, hasta los teléfonos inteligentes de hoy en día que les falta poco para ser computadoras actuales. Gracias a todos estos avances se puede hablar de la computación Ubicua, un concepto invertido al de la realidad virtual, el cual se basa en el hecho de ingresar a la computadora en el mundo del humano, mientras que el de la realidad virtual se basa en ingresar al humano en el mundo de la computadora.

Este paradigma de la computación Ubicua generó tendencias, como la de ofrecer conectividad en tiempo real 24/7 de los móviles, los precios ya sean de dispositivos, así como también de las operadoras sean más bajos, tener conectividad y accesibilidad a la red móvil y a otras redes como Internet desde cualquier parte y que se abran las puertas a un mundo de aplicaciones para dispositivos móviles. (Francisco Domínguez Mateos, 2014, pág. 13)

Se puede considerar a las calculadoras programables como los primeros dispositivos móviles en crearse, ya que a pesar de que no podían sincronizar sus datos con un ordenador, poseían capacidades gráficas y accesorios que hoy en día son tan pequeños, pero en sus tiempos eran tecnologías de gigantesca ayuda, como memorias, impresoras, etc.

Después del surgimiento de las calculadoras nace el primer dispositivo móvil, el Newton, creado como un asistente personal, el cual tenía grandiosas herramientas para su época. Esta era una tecnología muy revolucionaria y de gran ayuda, sin embargo, debido a su alta tecnología su precio era muy alto por lo que muy pocas personas tenían acceso a esta, ocasionando una poca aceptación del producto y su muerte. Pero con el deceso de este dispositivo empiezan a surgir nuevas tecnologías que se apoderarían del mercado: la Pilot y el Pocket PC. Estas eran tecnologías más avanzadas que incorporaban conexión a Windows, Bluetooth y Wifi.

Para comprender mejor esta evolución que ha surgido en el tiempo se ha separado en generaciones dependiendo el surgimiento de nuevas tecnologías.

Primera Generación. Estos dispositivos eran de gran tamaño, tenían un gran peso, lo que dificultaba su movilidad. Funcionaban de manera analógica, por lo que solo podían recibir y transmitir datos. En la Figura 12 se puede observar dos ejemplares de los teléfonos móviles de esta generación.



Figura 12 Motorola DynaTAC 8000X, uno de los primeros teléfonos móviles fabricados en gran cantidad.

Fuente: (Moro, 2009, pág. 5)

Segunda Generación. Conlleva al inicio de la tecnología digital, viene una notable evolución en el manejo de las llamadas, al igual que la implementación de mensajes cortos. Otro de los puntos importantes es que se permiten más enlaces simultáneos en el mismo ancho de banda. En la Figura 13 se puede apreciar un ejemplo de teléfono móvil de la segunda generación.



Figura 13 Teléfono móvil de la segunda generación.

Fuente: (Basterretche, 2007, pág. 8)

Tercera Generación. Empieza la tecnología para poder trabajar con correos electrónicos y descargar programas. Estos dispositivos podían ser utilizados como módems, lo que facilitaba las conexiones. Surgen redes más avanzadas y se incrementa la velocidad de los dispositivos

para poder soportar estas cargas. En la Figura 14 se pueden observar ejemplares de dispositivos móviles de la tercera generación.



Figura 14 Compaq iPaq 3630, el primer PocketPC de éxito.

Fuente: (Moro, 2009, pág. 10)

Cuarta generación. Esta es la generación de los teléfonos inteligentes, poseen un ancho de banda muy superior. Las prestaciones de pantalla, cámara fotográfica y de video son superiores en todo sentido. Dispositivos táctiles con sistemas operativos sofisticados que permiten el uso de aplicaciones de computadora o muy parecidas. La memoria y la velocidad del procesador se expanden notablemente para tener una mejor experiencia al usuario. En la Figura 15 se puede observar un smartphone, dispositivo móvil de la última generación.



Figura 15 Dispositivo móvil de la última generación.

Fuente: Google Pixel.

Gracias a esta última generación podemos dar paso al mundo de las aplicaciones para dispositivos móviles.

Aplicaciones móviles.

Como se ha podido observar, paralelamente al crecimiento de las redes de telefonía móvil, ha evolucionado las tecnologías implementadas en los dispositivos móviles, como el wifi, bluetooth, GPS, touchscreen, etc., han sido un punto clave para el avance y demanda del uso de los servicios móviles, creando con siglo una industria extremadamente grande a nivel de necesidades para las personas. (Gasca Mantilla, Camargo Ariza, & Delgado, 2014, pág. 22)

Entonces a continuación se define lo que es una aplicación móvil. Esta consiste en un programa que se puede ejecutar en dispositivos móviles como smartphones, tabletas, video consolas, entre otros. Se considera como un software para ejecutar tareas en un dispositivo móvil.

Antes de continuar, se debe hacer un paréntesis para conocer el contexto de las aplicaciones móviles. Estas pueden aprovechar mucho más el entorno en el que se ejecutan, debido a varios factores como las capacidades de los nuevos dispositivos, la transmisión de información, entre otras. Que difieren mucho de las aplicaciones tradicionales para computadora.

Para conocer más acerca de esto, se considerarán los factores más influyentes para las aplicaciones móviles.

Capacidad de los dispositivos.

Gracias a la evolución de la tecnología, se ha logrado incorporar varios elementos de gran aporte e importancia en los dispositivos móviles. A continuación, se conocerán los más destacados.

- El GPS (Sistema de geoposicionamiento), sirve para poder conocer la posición geográfica y saber que se debe mostrar.
- La brújula sirve para conocer la orientación tomando en cuenta los puntos cardinales (Norte, Sur, Este y Oeste).
- El acelerómetro sirve para conocer cuál es la orientación del dispositivo y superponer capas de información.
- La cámara sirve para poder captar el entorno y ampliar la información.

- La conexión a internet sirve para poder obtener información y ampliar la realidad. Esta conexión puede venir de internet móvil, wifi, etc.
- Los dispositivos móviles ya cuentan con la capacidad de procesamiento gráfico mejorada, con chips de aceleración potentes que ayudan a la experiencia que tiene el usuario.

Ubicuidad.

La ubicuidad se refiere al poder de tener acceso a toda la información o servicios necesarios para el usuario en cualquier momento o circunstancia, mediante su dispositivo.

Los beneficios que aporta la ubicuidad son muy propios de los dispositivos móviles.

- Es el primer medio de comunicación masivo, debido a la capacidad que tiene de llegar a la gran mayoría de usuarios en todo momento.
- Siempre está encendido, por lo que es capaz de enviar información aun cuando se encuentra en reposo.
- Es el primer dispositivo que está siempre con el usuario.

Contexto Social.

Este es uno de los puntos más importantes al momento de desarrollar una aplicación móvil, ya que el triunfo social significa un potencial cambio para cualquier usuario. Como ya se ha podido observar en las redes sociales, muchas de ellas han llegado a convertirse en necesidades de las personas hoy en día.

Costos.

Esta es una parte crítica y se la puede considerar una desventaja para las aplicaciones móviles.

A continuación, se revisa las necesidades para poder tener un entorno funcional.

- Acceso a internet.
- Dispositivos cercanos para la transmisión de información.
- La constante conexión a las redes sociales.
- Conexiones inalámbricas como bluetooth, GPS NFC, etc.
- Buena capacidad del procesador en el dispositivo.
- Constantes actualizaciones para la aplicación.

Todos estos factores significan costos que se necesitan para que una aplicación funcione óptimamente en un dispositivo.

Entonces se puede decir que una aplicación y su éxito depende del contexto en el que funcione. Este es el valor diferenciador con respecto a las aplicaciones web, ya que solo necesitan de un navegador y una conexión a la red para poder funcionar.

Tipos de aplicaciones móviles.

Como ya se conoce existen algunas plataformas sobre las que funcionan los dispositivos móviles actuales, cada una de estas posee una infraestructura diferente, convirtiendo a esto en el reto primordial para los desarrolladores, ya que lo que se busca es una aplicación que pueda funcionar en todos los celulares.

Debido a esto se han creado enfoques que permiten a las aplicaciones móviles funcionar en los distintos medios.

Existen algunos tipos de aplicaciones que aportan a conseguir este objetivo, pero se hablará de las más importantes a continuación.

Aplicaciones Web.

Al igual que las aplicaciones web para ordenador, estas aplicaciones tienen el mismo funcionamiento, son ejecutadas en el navegador del dispositivo. Generalmente son hechas con HTML, CSS y JavaScript.

Al igual que en las aplicaciones web para ordenadores, la principal fortaleza de este tipo de aplicaciones es la independencia que posee, ya que no necesita ningún entorno para operar, solo un navegador. De igual manera no hay que instalar ningún componente para su funcionamiento, solo se necesita acceso a internet.

El desperfecto de estas aplicaciones es la velocidad de ejecución y su interfaz, ya que pueden ser no muy atractivas como las aplicaciones nativas. Otro factor negativo es el hecho de no ser capaces de usar el hardware del dispositivo, como cámara, GPS, entre otros. Además, como necesitan conexión a internet, implica posibles problemas para los usuarios si no pueden acceder a internet en algún momento. En la Figura 16 se puede apreciar una aplicación web móvil, una web pública es un ejemplo.



Figura 16 Web pública.

Fuente: (Vique, pág. 23)

Aplicaciones Nativas.

Se puede decir que estas aplicaciones no aportan al hecho de servir en las distintas plataformas, ya que son aquellas que se construyen para una específica. Esto quiere decir que se debe considerar el sistema operativo, el dispositivo y las versiones.

Para funcionar, el código fuente es compilado y convertido a código ejecutable, de esta manera el dispositivo puede leerlo. Algo muy similar pasa en las aplicaciones de escritorio creadas para ordenadores.

Como ya es de suponer, la mayor ventaja de este tipo de aplicaciones es el poder acceder y utilizar a todos los elementos de hardware que posee un dispositivo, además a experiencia visual es mejor que en las aplicaciones web para móviles, incluyendo que el tiempo de ejecución es menor.

Por otra parte, si se desea desplegar una aplicación de este tipo, el proceso para incluirla a una tienda es más tedioso y obviamente más costoso, ya que se debe desarrollar para cada plataforma a la que se desee pueda ser ejecutada. En la Figura 17 se puede apreciar aplicaciones nativas móviles. Los juegos son un excelente ejemplo de este tipo de aplicaciones ya que aprovechan los recursos de los dispositivos.



Figura 17 Juegos para móviles.

Fuente: (Vique, pág. 31)

Aplicaciones Híbridas.

Se puede decir que estas aplicaciones como su nombre lo indica, son la combinación de las aplicaciones web y las nativas móviles, de manera que poseen las mejores características de estas.

Son desarrolladas con tecnologías multiplataforma como HTML, CSS JavaScript, pero tienen la capacidad de utilizar una buena parte de los elementos de hardware del teléfono.

Estas aplicaciones son ejecutadas en un contenedor web sobre el dispositivo móvil, permitiéndole funcionar en diferentes plataformas. Así pueden ser distribuidas en las tiendas sin una larga revisión.

Una desventaja es la apariencia de la aplicación, aún no es igual a la de una aplicación nativa al igual que su tiempo de ejecución, es todavía más lento. En la Figura 18 se puede observar una aplicación móvil, Facebook es un buen ejemplo.



Figura 18 Aplicación de Facebook para móviles.

Fuente: (Vique, pág. 26)

Herramientas de Desarrollo

Como ya se habló con anterioridad el desarrollo web y de aplicaciones móviles viene dado con la evolución de las tecnologías que antes surgieron y conocíamos. Cuando se desarrolla cualquier aplicación, es necesario poseer y conocer las herramientas que ayuden y faciliten la creación de aplicaciones, y hay que escoger las óptimas para lo que estamos haciendo, ya que existen varias herramientas que apoyan, pero no todas son iguales.

Ahora se revisarán algunas que ayudarán al desarrollo de esta disertación.

PHP.

Sus siglas significan Hypertext Preprocessor (Preprocesador de Hipertexto) y es uno de los lenguajes de programación más utilizados para el desarrollo web. Según estadísticas, este lenguaje es utilizado en más de 20 millones de sitios web y en más de un tercio de los servidores web en el mundo, lo que conlleva a darse cuenta de lo potente que es, sin contar que es gratis. (Evans, 2010, pág. 4)

Este lenguaje es una opción muy buena para las aplicaciones web orientadas a bases de datos gracias a su gran soporte, escalabilidad y facilidad para los diferentes motores.

El concepto de PHP consiste en ser un lenguaje de secuencia de comandos diseñado específicamente para el desarrollo web, he ahí la razón del porque es tan fuerte. Cuando se crea

una página web, el código PHP será interpretado en el servidor web y éste generará código HTML para que la persona que está visitando el sitio pueda verlo.

Otro de los fuertes de PHP se debe a que es un lenguaje de código abierto, lo que implica que se puede acceder a su código, modificarlo y redistribuirlo sin ningún costo.

En cuanto a si es una buena idea utilizar PHP para desarrollo web, debemos poner en consideración sus mejores ventajas frente a otros lenguajes orientados a la web (por ejemplo: ASP, JSP Allaire ColdFusion, etc.), entre estas las más notorias y que le dan ventaja a PHP son:

- El rendimiento de PHP es muy eficaz ya que con el uso de un solo servidor puede abastecer a millones de accesos al día.
- Gracias a que utiliza ODBC (Open Database Connectivity Standard), que es un estándar de conectividad general, puede conectarse a cualquier base de datos que posea un controlador ODBC, lo que brinda un muy amplio campo de compatibilidad.
- Debido a que es un lenguaje diseñado para la web, PHP incorpora un amplio contenido de bibliotecas con una gran cantidad de funciones integradas para que se puedan realizar tareas útiles en este entorno.
- PHP es gratis por lo que cualquiera puede descargarlo.
- Su sintaxis es similar a la de C y Perl, por lo que es de fácil entendimiento
- Para los que disfrutan de la orientación a objetos, PHP incorpora funciones con sintaxis del tipo de lenguajes que utilizan este paradigma.
- Tiene una gran portabilidad ya que está disponible para una gran cantidad de sistemas operativos.
- Este lenguaje es de código abierto, por lo que se tiene disponibilidad de cambiarlo.

HTML.

Como ya se habló con anterioridad, el front-end se puede decir que consiste en todo lo que ve el usuario, esto quiere decir que las tecnologías que se utilizan para desarrollar esta parte de un sistema se ejecutan en el navegador para poder ser visualizadas e interpretadas. De igual manera como se explicó antes el servicio web consta de tres partes fundamentales, el protocolo

HTTP, el esquema de direcciones URL y el lenguaje HTML. Este último es el que facilita la inclusión de recursos mediante el uso de etiquetas.

Entonces HTML es un lenguaje de programación que utiliza etiquetas las cuales definen los elementos que componen una página web como texto, imágenes, enlaces, etc. El navegador de internet es el encargado de interpretar estas etiquetas para que puedan ser visualizadas de manera correcta. HTML no es un lenguaje de programación tipo JAVA o C++ o Perl, ni un lenguaje de descripción de páginas tipo PostScript o PDF, sino que es un lenguaje que describe la estructura y la semántica del documento. (Anciano, 2010, pág. 18)

A pesar de que las versiones anteriores de HTML tenían varias limitaciones por el hecho de ser un lenguaje de etiquetas, la versión de HTML 5 abrió un mundo de posibilidades gracias a la incorporación de otras tecnologías como CSS, JavaScript entre otros. Esto quiere decir que ya no solo servía para definir elementos básicos si no utilizar terceros en el código y aumentar la funcionalidad.

Debido a que HTML no es más que un archivo de texto, es posible crear o modificar el código con cualquier editor de texto y abrirlo mediante cualquier navegador web. Por otra parte, existen herramientas que facilitan el desarrollo brindando un entorno más amigable.

Ya que HTML es un lenguaje de marcado, siempre existirá una marca de apertura y una que cerrará el contenido. Estas marcas pueden anidarse formando complejas estructuras para visualización en el navegador.

CSS.

El estilo siempre es una parte fundamental en cuanto a la creación de páginas web, es lo que el cliente observará y mientras mejor se vea, la experiencia del usuario será más agradable.

Cascading Style Sheets (Hojas de Estilo en Cascada) o más conocido como CSS, viene a ser un lenguaje de programación parecido a HTML que ayuda a integrar estilos a las páginas web para que tengan un orden y más amplias herramientas mejorando la manera en que se ven.

Anteriormente, se mezclaba el contenido del documento HTML con el estilo, modificando este en cada etiqueta que sea necesaria. Pero esto no era muy factible ya que, si se escribían grandes cantidades de código y se les aplicaba un color, y posteriormente se deseaba cambiar el color, se debía de modificar todas las líneas en donde se encuentre lo que se va a modificar.

Para poder corregir el anterior problema, CSS vincula los contenidos del documento HTML con la hoja de estilos CSS de manera que, si se realiza un cambio en la hoja de CSS a algún contenido, se realizará el cambio en todos los contenidos de la misma instancia en el documento HTML.

Las hojas de estilos abarcan grandes herramientas que permiten y ayudan a mejorar la maquetación de las páginas web. Cabe recalcar que las posibilidades que brinda CSS son mucho más amplias comparando con los documentos impresos de tipo Word, PowerPoint, etc.

Una hoja de estilo puede encontrarse dentro del mismo documento HTML o estar en un documento aparte, pero con la debida vinculación a al documento HTML que va a utilizar dicha hoja.

Las hojas de estilo según su aplicación se pueden dividir en dos partes:

Interna o residente. Este tipo de hojas solo afectan al documento en el que son creadas. Se definen mediante etiquetas dentro del HTML que se integran en la cabecera.

Externa. Este tipo de hojas pueden afectar a cualquier documento que se vincule con ellas. Se crean en un documento independiente con extensión .css.

Las hojas de estilo también se pueden dividir según el contenido al que afectan:

De clase. Afecta a un bloque definido de contenido. Para que funcione se debe instanciar la clase dentro de la etiqueta, y de esta manera se puede afectar a una etiqueta en específica y no a todas de la misma denominación.

De etiqueta. Se aplica a una etiqueta determinada y afecta al contenido de ésta. Debe estar vinculada con la hoja de estilo.

De identificador. Se integra un identificador a los textos que se quiere afectar por la hoja de estilos.

CSS ayuda a minimizar código, es muy sencillo gracias al uso de propiedades y es uno de los pilares fundamentales del DHTML y puede acoplarse con JavaScript. A pesar de sus grandes ventajas, se puede considerar como el mayor inconveniente la incompatibilidad. Varios navegadores soportan el lenguaje, el problema son las irregularidades que existen entre las distintas versiones de los navegadores.

JavaScript.

Hasta ahora constamos de HTML y CSS, lenguajes que permiten la visualización de una página web, el primero dando el contenido y el segundo el estilo. Ahora para poder tener un sitio funcional, que pueda modificar su contenido y formato cuando el usuario interactúe con este, nos hace falta un complemento más.

Se podría decir que JAVA dio paso a la creación de JavaScript. En JAVA se podían crear applets, que eran pequeños programas que se incrustaban en la página. De aquí nace Netscape que también serviría para crear programas sencillos en páginas. Más tarde con la alianza de Sun Microsystems y Netscape nace JavaScript. A pesar de estos datos, JAVA y JavaScript son dos lenguajes muy diferentes. En la Tabla 1 se muestran las diferencias que tienen JAVA y JavaScript.

Tabla 1 Java vs JavaScript.

| JavaScript | Java |
|---|---|
| Es un lenguaje sencillo. | Es un lenguaje más complejo y completo con una <i>estricta</i> orientación a objetos. |
| Diseñado para desarrollar aplicaciones Web. | Diseñado para desarrollar aplicaciones de propósito general incluyendo tanto aplicaciones de escritorio como Web. |
| Lenguaje interpretado por el navegador. No requiere compilador. | Lenguaje compilado a código intermedio que luego es interpretado. |
| Más flexible. Por ejemplo, no requiere declarar variables ni tipos (aunque es aconsejable hacerlo). | Tiene reglas mucho más rígidas. Por ejemplo, hay que declarar todas las variables con sus tipos. |

Fuente: (Jorge Mohedano, 2012, pág. 10)

JavaScript es un lenguaje ejecutado por el navegador, su sintaxis es parecida a la del lenguaje JAVA, por esto su nombre. El fin primordial de este lenguaje es el de definir acciones cuando ocurren eventos en el navegador. Estos eventos pueden ser ejecutados por el usuario o por el mismo navegador. (García, 2016, pág. 24)

Debido a que es un lenguaje de script, este no necesita ser compilado, los navegadores se encargan de interpretar el código de manera directa, lo que proporciona un trabajo rápido con resultados muy satisfactorios. Al igual que HTML y CSS, este lenguaje puede ser escrito en un archivo de texto plano para su ejecución.

A pesar de que JavaScript se ha convertido en uno de los lenguajes más utilizados en cuando al desarrollo web, posee todavía algunas desventajas. Una de las más importantes es la seguridad, muchas veces código malicioso puede ser ejecutado en la máquina del cliente con el objetivo de sacar provecho de algunas vulnerabilidades. Otra desventaja de este lenguaje es la tendencia a introducir grandes fragmentos de código en los sitios web.

Cabe recalcar que JavaScript, hoy en día es soportado por la mayoría de los navegadores a nivel mundial que vienen a ser más del 97% de los navegadores existentes. Solo los navegadores de texto que utilizan algunos usuarios del sistema operativo Linux no soportan de manera correcta el lenguaje.

Laravel.

Laravel es un framework para aplicaciones web que contiene una sintaxis expresiva y elegante. Es de código abierto y gracias a su facilidad de uso se convierte en una herramienta de fácil aprendizaje. Laravel utiliza lenguaje PHP convirtiéndolo en una herramienta simple y potente.

Uno de los objetivos más importantes de Laravel es evitar el famoso código espagueti por lo que su organización está formada por dependencias. Cabe recalcar que utiliza el patrón de diseño del Modelo Vista-Controlador. La arquitectura de las aplicaciones que se crean con este marco tiene múltiples capas, en la estructura web de tres capas, la base de datos no es un servicio directo para cada cliente, sino que se conectan a la web del servidor para obtener servicios dinámicos en tiempo real para el que lo necesita.

En conclusión, Laravel es un marco para desarrollo web muy potente y fácil, además es elegante y divertido de utilizar. Diseñado para facilitar el trabajo a los desarrolladores de manera que puedan evitar el código desordenado y así mantener un modelo que respete la organización.

React.

Es una biblioteca de JavaScript que ayuda y facilita la creación de interfaces para el usuario de una manera interactiva y sencilla. Permite el diseño de vistas para cada estado de una aplicación y las renderiza y actualiza de manera eficiente, tomando en cuenta todos los componentes cuando los datos cambien.

React tiene como objetivo desarrollar aplicaciones web más organizadamente y con menos código que si se utilizara JavaScript puro o librerías como jQuery que están centradas en la manipulación del DOM.

Una de las grandes ventajas de usar React es el hecho de que ofrece muchas cosas ya listas, de manera que nos facilita el trabajo sin tomar en cuenta que evita el código espagueti.

Ionic.

Es un framework para desarrollo de aplicaciones móviles multiplataforma, además es de código abierto para quienes lo desean. Este marco está desarrollado sobre AngularJS y facilita de una manera absoluta la creación de aplicaciones proporcionando una estructura sobre la cual se puede trabajar las interfaces de usuario que se requieran.

Cabe recalcar que ni Ionic ni angular tienen orientación a objetos, por lo que los conceptos de clases no se pueden utilizar para este framework.

Ionic nos permite crear aplicaciones híbridas brindando una apariencia nativa y manejando lenguajes convencionales como son HTML, CSS y JavaScript.

Estas aplicaciones son construidas sobre Apache Cordova, lo que quiere decir que tienen acceso a todos los elementos de éste y así poder utilizar funciones nativas como notificaciones, cámara, acelerómetro, etc.

Bases de Datos

Se puede definir a una base de datos como el conjunto de datos interrelacionados y administrados como una misma unidad. Las bases de datos sirven para almacenar grandes y pequeñas cantidades de información de una manera ordenada y así poder utilizar esa información para algún fin.

Las bases de datos relacionales son aquellas que están estructuradas en registros, más conocidos como filas, y campos, más conocidos como columnas, estos se vinculan entre sí por campos en común con características similares entre uno o varios bloques de información llamados tablas. Se dice que un registro es la unidad elemental de la información de la tabla o conjunto de datos.

Después surgieron los sistemas gestores de bases de datos, los cuales apoyan en la manipulación de las tablas sin tener que crear programas que procesen los datos. De esta manera las acciones básicas de consultar, crear, actualizar, eliminar y editar estos datos se volvieron más fácil.

Con el avance de la tecnología, los sistemas gestores de bases de datos ya no solo son utilizados para manipular la información, sino también para apoyar a aplicaciones externas y que estas puedan utilizar la información contenida en estas estructuras.

Para poder manejar la información y manipular las bases de datos, se creó un lenguaje estandarizado llamado SQL (Structured Query Language). Este lenguaje es utilizado en bases de datos relacionales, lo que significa que utiliza un modelo de datos relacional. (Sanz, 2014, pág. 16)

MySQL.

Este es un sistema gestor de bases de datos ideal para la creación de aplicaciones web ya que es rápido, sólido y flexible. Se lo utiliza mucho en transacciones on-line y en soluciones profesionales que impliquen el almacenamiento de datos, así como consultas veloces. Debido a su gran versatilidad es uno de los sistemas gestores más conocidos y utilizados a nivel mundial en lo que se refiere al entorno web, esto se debe a las ventajas que posee en este ambiente las cuales se explican a continuación.

- Es un sistema público que permite la modificación de su código, lo que ayuda mucho en cuanto a la personalización de este dependiendo de las necesidades que se tengan.
- Ya que fue desarrollado con C y C++, se acopla muy bien con aplicaciones que también estén desarrolladas con estos lenguajes.
- Posee una licencia gratuita y una comercial.
- Es de tipo cliente-servidor lo que permite trabajar con multiusuarios y de esta manera la seguridad estará anclada a cada usuario dependiendo de los permisos que tengan.
- Es multiplataforma, lo que quiere decir que posee la característica de portabilidad. Está disponible en más de veinte plataformas.

MySQL maneja archivos y datos con un tamaño menor al de otros motores de bases de datos, por lo que el tiempo de respuesta en lo que se refiere a localizar y grabar datos es mucho menor en comparación con otros sistemas gestores. Debido a esto muchas empresas utilizan MySQL,

no solo pequeñas empresas, si no algunas muy reconocidas como son Yahoo, Google, CISCO, Motorola, NASA, Facebook, YouTube, Twitter, entre otras.

OOHDM

Debido al gran crecimiento del Internet en los últimos años, el desarrollo de las aplicaciones web ha estado necesitando de un método de ingeniería de software que sea de gran ayuda. Uno de los métodos utilizado para el diseño de aplicaciones web es el OOHDM (Object Oriented Hypermedia Design Model ó en español Modelo de Diseño de Hipermedia Orientado a Objetos), que proporciona un modelo orientado a objetos para que se pueda implementar el diseño obtenido. Pero cabe recalcar que por el hecho de que se utilizan lenguajes que no son orientados a objetos, se pierden algunas de las propiedades de este paradigma como la reutilización de código, herencia, entre otros. (Germán Terrazas, pág. 1)

Debido a que las aplicaciones web suelen incluir información compleja y comportamiento de navegación, la metodología OOHDM utiliza mecanismos de abstracción y composición enmarcados en la orientación a objetos, lo que permite una descripción concisa de los elementos de información y la especificación de patrones de navegación con transformaciones de interfaz.

La metodología OOHDM está basada en cuatro etapas.

Diseño conceptual: Esta etapa también conocida como análisis de dominio, se construye todo lo que tiene que ver con los esquemas que permiten definir los objetos del dominio de la aplicación, las clases y objetos, y sus relaciones.

Diseño navegacional: En esta fase de la metodología se determina el perfil del usuario tomando en cuenta el aspecto cognitivo. Gracias a esto, se describe la estructura navegacional de la aplicación utilizando nodos, enlaces, índices, tours, etc. Estos contextos nos permiten definir lo usuarios potenciales que tendrá la aplicación. El modelo navegacional puede evolucionar independientemente del modelo conceptual facilitando el mantenimiento.

Este diseño se representa con modelos o esquemas.

- Esquema de clases navegacionales: Muestran las vistas de la aplicación, estas vistas son representadas a su vez con nodos o enlaces que son un tipo de clases navegacionales, también se consideran índices y tours guiados.

- Esquema de contexto navegacional: Este esquema permite estructurar el hiperespacio de navegación en subespacios, los cuales muestran la información a presentar y los enlaces disponibles cuando se acceda a un objeto o nodo específico.

Diseño abstracto de la interfaz: En esta etapa lo que se busca es definir los objetos que serán vistos por los usuarios, la representación de los objetos navegacionales, la sincronización, entre otros. De igual manera se describen los objetos que activarán la interfaz, y las funcionalidades de la aplicación.

Las interfaces de clases se describen como botones, cuadros de texto, etc. Los objetos de interfaz representan los objetos navegacionales, brindándoles una apariencia que se puede percibir. Se declara como manejar los eventos externos y generados por el usuario.

Implementación: Esta es la fase final en la cual se traducen los objetos de la interfaz en un lenguaje de programación, esta puede incluir arquitecturas elaboradas con la de cliente servidor. En conclusión, se desarrolla todos los diseños realizados para que sean funcionales.

A continuación, en la Tabla 2 se muestra un cuadro explicativo de las etapas de la metodología OOHDL.

Tabla 2 Etapas de la Metodología OOHDM.

| Etapas | Productos | Formalismos | Mecanismos | Descripción |
|-------------------------------------|--|--|---|---|
| Obtención de Requerimientos | Casos de Uso (actores, escenarios) | Plantillas del formato del documento, Diagramas de Interacción de Usuario (UIDs) | Técnicas de Observación, Entrevistas | Se crea un documento que describe actividades y requerimientos de los usuarios |
| Diseño Conceptual | Clases, subsistemas, relaciones, atributos | Modelos Orientados a Objetos | Clasificación, agregación, generalización y especialización | Se modela la semántica del dominio de la aplicación |
| Diseño Navegacional | Nodos, enlaces, estructuras de acceso, contextos navegacionales, transformaciones de navegación | Vistas Orientadas a Objetos, Cartas de navegación orientadas a objetos, Clases de Contexto | Clasificación, agregación, generalización y especialización | Se tiene en cuenta el perfil del usuario y las tareas. Se enfatiza en los aspectos cognitivos. Se crea la estructura de navegación de la aplicación |
| Diseño de Interfaz Abstracta | Objetos de la interfaz abstracta, respuestas a eventos externos, transformaciones de la interfaz | Vistas Abstractas de Datos (ADV), Diagramas de Configuración, Cartas de navegación de los ADVs | Mapeado entre la navegación y los objetos visibles | Se modelan los objetos visibles. Se describe la interfaz para los objetos de navegación. Se define el aspecto de los objetos de la interfaz |
| Implementación | Aplicación en funcionamiento | Los soportados por el entorno | Los que provea el entorno | Se realiza la puesta en producción del sistema |

Fuente: (Vinueza, 2013, pág. 58)

Ventajas de utilizar la metodología OOADM:

- Se puede diferenciar los tres niveles del diseño.
- Se pueden representar gráficamente y de manera clara los elementos propios de la aplicación.
- Se pueden satisfacer las necesidades del usuario en cada etapa.
- Gracias a su documentación se puede llevar un control del desarrollo de las etapas.

Desventajas:

- Se requiere cierto conocimiento debido a los modelos que utiliza.
- Es un poco complicado debido a las reglas que la metodología demanda.
- Se tienen una gran cantidad de diagramas.

Capítulo 4: Fase inicial

En este capítulo se realizará el análisis y especificación de cómo será construido el sistema, tomando en cuenta la problemática del taller analizada anteriormente. El objetivo principal que se va a tratar en esta parte será brindar una clara descripción de los requerimientos del sistema en un aspecto entendible para el cliente y para el desarrollador, de esta manera se tendrá una visión clara de las funcionalidades que poseerá el software, así como también otros aspectos importantes que definirán su correcta funcionalidad y que apoyarán al ingeniero en la construcción de éste.

También, este capítulo definirá la primera etapa de desarrollo del proyecto para sentar una clara idea del funcionamiento del sistema basándose en cada uno de los módulos que lo conformarán. Además, se especificarán los recursos necesarios para la construcción, funcionamiento, usabilidad, entre otros aspectos para la interacción de todo el sistema con sus partes y con el elemento humano. Se definirán aspectos claves para delimitar el proyecto y así no existan confusiones futuras cuando ya esté terminado.

Análisis y obtención de requerimientos

El análisis de requerimientos es una de las partes más importantes en cualquier proceso de desarrollo de software ya que se estudian las necesidades del usuario para poder llegar a una definición de los requerimientos del sistema que se va a crear. Esta etapa debe producir algún tipo de documento que describa lo que la aplicación hará cuando ya esté terminada.

Debido a esto se utilizará el estándar de especificación de requerimientos de la norma IEEE 830, ya que provee un refinamiento de los requerimientos de un software mediante un documento muy completo, legible, entendible y específico. Dando lugar a la participación de los usuarios involucrados en esta etapa de definición, ya que ellos conocen mejor los procesos a automatizar en el sistema.

Especificación de requerimientos de software

Introducción

Propósito

El presente trabajo de disertación tiene como objetivo el desarrollo de un sistema para gestionar talleres automotrices. Se escogió este tema debido a que en el Ecuador existe un gran índice de talleres que no poseen una administración óptima y por lo tanto se generan fallos en varias de sus áreas de trabajo. Estas fallas son causantes de pérdidas en cuanto a los ingresos de cada taller, así como inseguridad en la información, falta de confianza por el cliente, entre otros aspectos que perjudican al negocio directamente.

Gracias a las tecnologías de la información, es posible automatizar procesos dentro de cada taller, lo que facilitará la gestión de éstos. Además, de brindar orden, formalismo, facilidades, prevención de errores humanos, seguridad de información, etc. la información será más segura, ya que en la actualidad los datos de cada taller son guardados físicamente o con instrumentos que no proporcionan la seguridad y confiabilidad adecuada que el cliente necesita.

Este documento tiene como objetivo principal brindar una descripción clara de lo que podrá hacer el sistema una vez que esté finalizado. De igual forma, se aclararán los puntos y criterios para que el sistema sea evaluado y satisfaga todas las necesidades requeridas por el cliente.

Esta especificación está dirigida hacia el cliente y al estudiante ya que, podrán entender cómo funcionará el sistema y se podrá crear una base sobre la cual se desarrollará.

Alcance

El sistema tendrá como objetivo la automatización de los procesos principales del taller, de esta manera se podrá asegurar un orden en la información, ya que se guardará en una base de datos, y dependiendo de las funciones del usuario, se tendrá acceso o no a ésta. Por otro lado gracias a los protocolos de seguridad y control de errores que se implementarán, existirá fiabilidad y facilidad, asegurando el uso correcto del mismo.

El sistema presentará los siguientes módulos:

Módulo 1: Ingreso al sistema (Inicio de sesión, cerrar sesión).

Módulo 2: Usuarios (Ingreso, modificación, eliminación y consulta).

Módulo 3: Clientes (Ingreso, modificación, eliminación y consulta).

Módulo 4: Vehículos (Ingreso, modificación, eliminación y consulta).

Módulo 5: Servicios (Ingreso, modificación, eliminación y consulta).

Módulo 5: Cotizaciones (Ingreso, modificación, eliminación y consulta).

Módulo 6: Compras (Ingreso, modificación, eliminación y consulta).

Módulo 7: Facturas (Ingreso, impresión y consulta).

Módulo 8: Turnos (Ingreso, modificación, eliminación y consulta).

Módulo 9: Notificaciones (Ingreso, modificación, eliminación, consulta y envío de correo).

Módulo 10: Promociones y descuentos (Ingreso, modificación, eliminación y consulta).

Módulo 11: Reportes financieros (Consulta).

Los módulos especificados anteriormente se implementarán para el sistema web. En cuanto a los dispositivos móviles, se incluirán los módulos 1, 3, 4 y el envío de correo del módulo 9.

Personal involucrado

Tabla 3 Personal Involucrado.

| | |
|-------------------------|--|
| Nombre | Andrés Pinto |
| Rol | Desarrollador |
| Categoría profesional | Estudiante |
| Responsabilidades | Desarrollo del sistema |
| Información de contacto | Email: pitoandres2014a@gmail.com Número: 0992626973 |

| | |
|-------------------------|---|
| Nombre | Fernando Vinueza |
| Rol | Auspiciante |
| Categoría profesional | Ing. Mecánico |
| Responsabilidades | Responsable del taller automotriz |
| Información de contacto | Email: fernando.vinueza.esacero@gmail.com Número: 0998229542 |

| | |
|-------------------------|---|
| Nombre | Fabián de la Cruz |
| Rol | Director de Tesis |
| Categoría profesional | Ingeniero |
| Responsabilidades | Revisión de proyecto de disertación |
| Información de contacto | Email: FIDELACRUZ@puce.edu.ec Número: 0984617742 |

Fuente: Propia.

Resumen

A continuación, se describen todas las funcionalidades del sistema que permitirán proveer al cliente un producto de calidad y que satisfaga sus necesidades para poder mejorar su negocio implementando las tecnologías de la información. Cada una de estas automatizaciones que se realizarán, están claramente definidas y especificadas de manera secuencial para que el cliente pueda tener un pleno entendimiento de lo que abarca el sistema.

Por otro lado, se presentan aspectos fundamentales del sistema que cubren cualquier aspecto no funcional del mismo y que permitirán corroborar las dudas que posee el cliente, de esta manera la descripción del sistema será muy completa, clara y concisa.

Descripción general

Perspectiva del producto

Este sistema para el taller AUTOTREK, estará orientado a la web y a dispositivos móviles, esto quiere decir que su utilización será descentralizada y a más de eso no tendrá interacción con otros sistemas externos, por lo que será independiente, únicamente para el taller. También el sistema tendrá un solo backend que servirá para que se consuman sus servicios desde el frontend web y móvil.

Este sistema almacenará la información del taller automotriz, la que podrá ser utilizada y procesada para brindar valor al negocio. De esta manera se podrá omitir las fallas humanas que se tenían antes de la implementación de las tecnologías de la información.

Funcionalidad del producto

El sistema contendrá las siguientes funcionalidades.

- F1. Ingreso al sistema.
- F2. Gestión de usuarios.
- F3. Gestión de clientes.
- F4. Gestión de vehículos.
- F5. Gestión de servicios.
- F6. Gestión de cotizaciones.
- F7. Gestión de compras.
- F8. Gestión de facturas.
- F9. Gestión de turnos.
- F10. Gestión de notificaciones.
- F11. Gestión de promociones y descuentos.
- F12. Gestión de reportes financieros.

Características de los usuarios

El sistema permitirá la interacción de 4 tipos de usuarios, los que serán:

Tabla 4 Características de los Usuarios.

| | |
|-----------------|--|
| Tipo de usuario | Gerente |
| Formación | Ing. Mecánico y/o Automotriz, Gerencia |
| Habilidades | Conocimiento y manejo del negocio de taller automotriz |
| Actividades | Gestionar todo el sistema desde las interfaces creadas |

| | |
|-----------------|------------------------------------|
| Tipo de usuario | Jefe de taller |
| Formación | Mecánico automotriz |
| Habilidades | Conocimiento y reparación de autos |
| Actividades | Gestionar vehículos, repuestos |

| | |
|-----------------|---|
| Tipo de usuario | Administrador de negocio |
| Formación | Administración general |
| Habilidades | Conocimiento sobre administración de negocios |
| Actividades | Gestionar clientes, facturación, turnos, citas, proformas, notificaciones, promociones y descuentos, ingresos y egresos |

| | |
|-----------------|---|
| Tipo de usuario | Administrador de sistema |
| Formación | Ingeniero en sistemas y computación |
| Habilidades | Conocimientos sobre sistemas, bases de datos y el negocio |
| Actividades | Gestionar todo el sistema |

Fuente: Propia.

Restricciones

- El sistema estará orientado a la web, por lo que es necesario que las computadoras en las que funcione el sistema tengan un navegador para que pueda funcionar.
- Se utilizarán los lenguajes PHP, JavaScript, TypeScript y CSS para el desarrollo del sistema.
- Los frameworks y librerías consideradas para facilitar el trabajo serán Laravel para el backend, React para el frontend, y Ionic para la aplicación móvil.
- El motor de base de datos que se utilizará será MySQL ya que aporta grandes beneficios en lo que se refiere a sistemas web.
- Aparte, se utilizarán librerías como Bootstrap y Material UI para el desarrollo de las interfaces.

- El sistema obedecerá las normativas de bases de datos que se conocen para un óptimo funcionamiento, así se prevendrán posibles errores.

Suposiciones y dependencias

- Se dependerá de una conexión a la red para que el sistema pueda funcionar ya que está orientado a la web.
- Debido a que se utilizarán diferentes librerías y tecnologías para las interfaces, dependerá del sistema operativo en el que se ejecute para que sus componentes se vean diferentes. Esto no afecta en la funcionalidad del sistema, solo cambiarán elementos como íconos, cuadros de texto, comboBox, etc. En su forma visual mas no en la funcional.
- Se deberá instalar la aplicación móvil en cada dispositivo para que pueda ser utilizado. Cabe recalcar que las versiones de sistemas operativos de cada celular deberán ser compatibles con la aplicación.

Evolución previsible del sistema

- El sistema puede evolucionar conforme se incorporen más funcionalidades que satisfagan necesidades futuras que requiera el taller automotriz. Se pueden seguir creando más módulos e incorporarlos al mismo, tomando en cuenta el no afectar a la estructura del sistema.
- Gracias a que está orientado a la web, no se necesitaría de instalaciones por actualizaciones de versión, esto en lo que se refiere a la web. Por otro lado, en el caso de la aplicación móvil si se necesitará de una actualización de la aplicación, ya que se instalan en el sistema operativo del dispositivo.

Requisitos específicos

Diagrama General de Casos de Uso

Los diagramas de Casos de Uso refieren una técnica para recopilar los requerimientos y necesidades de los usuarios, tomando en cuenta los escenarios (los que se llaman casos) en los que los usuarios tienen interacción con el sistema y así logran un objetivo común el cual por lo general es satisfacer una de sus necesidades planteadas.

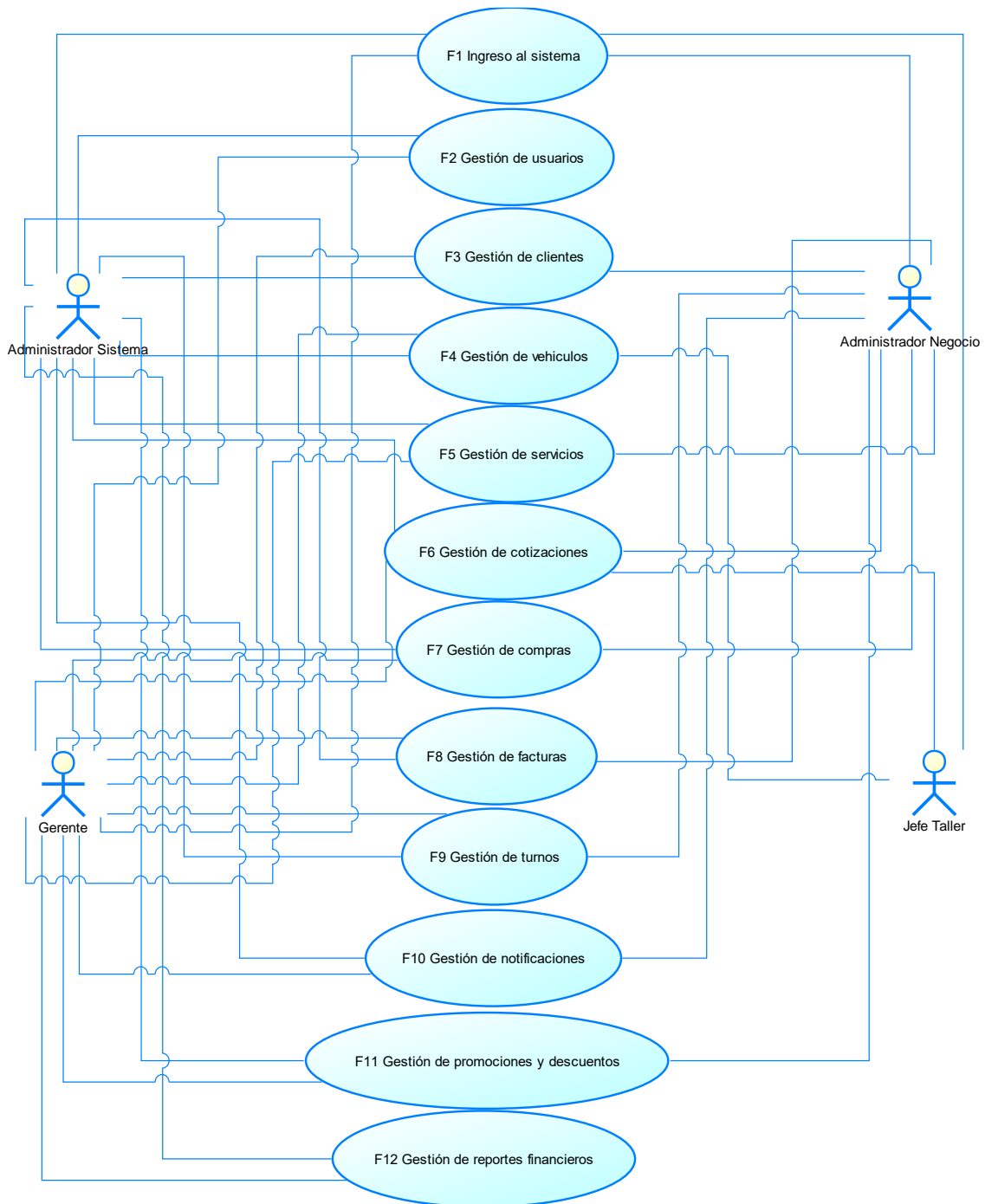


Figura 19 Diagrama general de casos de uso.

Fuente: Propia.

| | | |
|----------------------------|----|--|
| Número de funcionalidad | de | F1 |
| Nombre de funcionalidad | de | Ingreso al sistema |
| Descripción | | Permite ingresar al sistema dependiendo el cargo que tenga el usuario. |
| Prioridad de funcionalidad | de | <input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional |
| Autores | | Gerente, Jefe de taller, Administrador de negocio y Administrador del sistema. |

| | | |
|----------------------------|----|---|
| Número de funcionalidad | de | F2 |
| Nombre de funcionalidad | de | Gestión de usuarios |
| Descripción | | Permite la gestión de los usuarios que podrán ingresar al sistema. Refiriéndose al ingreso, modificación, eliminación y consulta de los mismos. |
| Prioridad de funcionalidad | de | <input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional |
| Autores | | Gerente, Administrador del sistema. |

| | | |
|----------------------------|----|---|
| Número de funcionalidad | de | F3 |
| Nombre de funcionalidad | de | Gestión de clientes |
| Descripción | | Permite la gestión de los clientes del taller que son dueños de los vehículos. Esto se refiere a ingreso, modificación, eliminación y consulta. |
| Prioridad de funcionalidad | de | <input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional |

| | |
|---------|--|
| Autores | Gerente, Administrador de negocio y Administrador del sistema. |
|---------|--|

| | |
|----------------------------|---|
| Número de funcionalidad | F4 |
| Nombre de funcionalidad | Gestión de vehículos |
| Descripción | Permite la gestión de los vehículos que ingresan al taller. Esto se refiere a ingreso, modificación, eliminación y consulta. |
| Prioridad de funcionalidad | <input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional |
| Autores | Administrador de sistema, Gerente, Jefe de taller |

| | |
|----------------------------|---|
| Número de funcionalidad | F5 |
| Nombre de funcionalidad | Gestión de servicios |
| Descripción | Permite la gestión de los servicios que brinda el taller. Refiriéndose al ingreso, modificación, eliminación y consulta. |
| Prioridad de funcionalidad | <input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional |
| Autores | Gerente, Administrador del sistema, Administrador de negocio. |

| | |
|-------------------------|-------------------------|
| Número de funcionalidad | F6 |
| Nombre de funcionalidad | Gestión de cotizaciones |

| | |
|----------------------------|---|
| Descripción | Permite la gestión de las cotizaciones que se realizarán a cada uno de los vehículos del taller. Refiriéndose al ingreso, modificación, eliminación y consulta. |
| Prioridad de funcionalidad | <input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional |
| Autores | Gerente, Administrador del sistema, Administrador de negocio y Jefe de taller. |

| | |
|----------------------------|--|
| Número de funcionalidad | F7 |
| Nombre de funcionalidad | Gestión de compras |
| Descripción | Permite la gestión de las compras que se necesitarán para cada automóvil y para el taller. Refiriéndose al ingreso modificación, eliminación y consulta de las mismas. |
| Prioridad de funcionalidad | <input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional |
| Autores | Gerente, Administrador del sistema y Administrador de negocio. |

| | |
|----------------------------|--|
| Número de funcionalidad | F8 |
| Nombre de funcionalidad | Gestión de facturas |
| Descripción | Permite manejar las facturas que el taller emita a sus clientes. Refiriéndose al ingreso, anulación y consulta de las mismas. |
| Prioridad de funcionalidad | <input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional |
| Autores | Gerente, Administrador de negocio y Administrador del sistema. |

| | | |
|----------------------------|----|---|
| Número de funcionalidad | de | F9 |
| Nombre de funcionalidad | de | Gestión de turnos |
| Descripción | | Permite la gestión de los turnos para que los clientes sean atendidos. Refiriéndose al ingreso, modificación, eliminación y consulta de los mismos. |
| Prioridad de funcionalidad | de | <input type="checkbox"/> Alta/Esencial <input checked="" type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional |
| Autores | | Gerente, Administrador de negocio y Administrador del sistema. |

| | | |
|----------------------------|----|---|
| Número de funcionalidad | de | F10 |
| Nombre de funcionalidad | de | Gestión de notificaciones |
| Descripción | | Permite la gestión de las notificaciones que se realizaran paulatinamente a los clientes dependiendo de las necesidades. |
| Prioridad de funcionalidad | de | <input type="checkbox"/> Alta/Esencial <input checked="" type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional |
| Autores | | Gerente, Administrador de negocio y Administrador del sistema. |

| | | |
|-------------------------|----|---|
| Número de funcionalidad | de | F11 |
| Nombre de funcionalidad | de | Gestión de promociones y descuentos |
| Descripción | | Permite la gestión de las promociones que puede tener el taller en distintos periodos de tiempo. Refiriéndose al ingreso, modificación, eliminación y consulta de las mismas. |

| | | |
|----------------------------|----|--|
| Prioridad de funcionalidad | de | <input type="checkbox"/> Alta/Eencial <input checked="" type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional |
| Autores | | Gerente, Administrador de negocio y Administrador del sistema. |

| | | |
|----------------------------|----|---|
| Número de funcionalidad | de | F12 |
| Nombre de funcionalidad | de | Gestión de reportes financieros |
| Descripción | | Permite la observación de los ingresos y egresos que posea el taller al finalizar un periodo de tiempo determinado. Así se podrá verificar las ganancias. |
| Prioridad de funcionalidad | de | <input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional |
| Autores | | Gerente, Administrador del sistema. |

Fuente: Propia.

Requisitos comunes de las interfaces

- Las entradas que se le deberán proporcionar al sistema deberán ser con componentes adecuados dependiendo del tipo de entrada. Se manejará la base de datos asociada a estos componentes para que no se queme código en ninguna parte del sistema.
- De igual manera, las salidas del sistema estarán visibles al usuario utilizando los componentes más adecuados para su pleno entendimiento y fácil interpretación. Estas salidas generalmente estarán asociadas a la base de datos y a un proceso de información que genere la respuesta requerida.

Interfaces de usuario

- Los colores que se utilizarán como base de las interfaces serán el Azul como color principal y el amarillo como color secundario.
- Los componentes que se manejarán serán los más adecuados para que el usuario no tenga confusiones y no ingrese información que pueda ocasionar errores en el sistema.
- Se utilizarán tablas generalmente cuando se refiera a consultas de información del taller.
- Se utilizarán íconos de la misma categoría para que se tenga una visibilidad más amigable y compacta.
- De igual manera los botones que se utilizarán serán similares entre sí dependiendo su funcionalidad y también su ubicación.
- Se utilizarán menús para navegar entre las diferentes pantallas del sistema, esto gracias también a la utilización de una barra de navegación.

Interfaces de hardware

- Es imprescindible que el cliente disponga de computadores en un estado aceptable para que no existan fallos.
- Se necesita que los computadores posean puertos de red para conectarse a la red.
- No obstante, se necesitan equipos con mouse, teclado, monitor e impresora.
- Se necesitará de igual manera dispositivos móviles que se encuentren en un estado aceptable y funcional para que la aplicación pueda correr de manera normal.

- Los dispositivos deberán tener almacenamiento suficiente para que soporte la aplicación, aparte de un sistema operativo Android o IOS.

Interfaces de software

- Sistema operativo Windows, Linux o MacOS, en el que se instalan las dependencias necesarias para el sistema como el navegador.
- Explorador o navegador de internet, en el que se ejecutará la aplicación.
- Servidor apache, donde se implementará el protocolo HTTP y la noción de página web.
- Motor de bases de datos MySQL, en donde se almacenarán los datos del taller.

Interfaces de comunicación

El servidor, el cliente, la aplicación móvil y la base de datos se comunicarán entre si utilizando los protocolos estándares de transmisión de datos ya existentes para el funcionamiento de todo el sistema, ya que será desarrollado por capas, se considerará un servidor que será el desarrollado en Laravel y el cliente, que será desarrollado en React.

Requisitos funcionales

A continuación, se presentarán los requisitos que tendrá el sistema para que pueda satisfacer las demandas del cliente. Se describirán cada uno de estos tomando en cuenta su nombre, una breve descripción, los actores que estarán sujetos a cada uno, el flujo principal que se debe seguir para interactuar con la funcionalidad, el flujo alterno y la excepciones que se pueden tener.

1. Gestión de cotizaciones

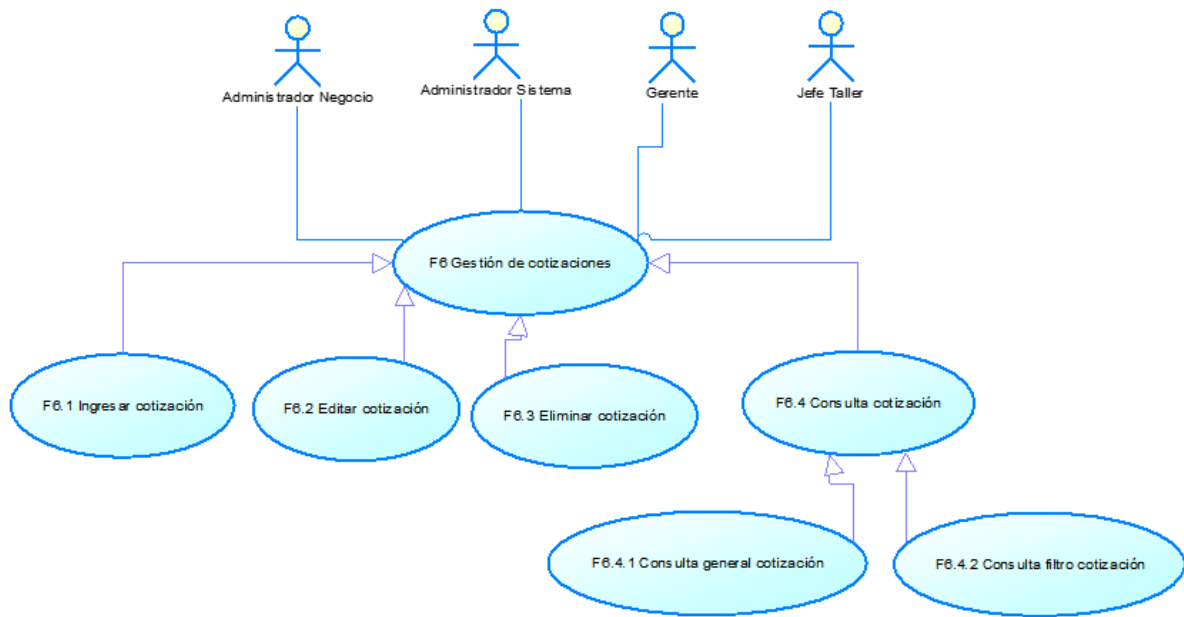


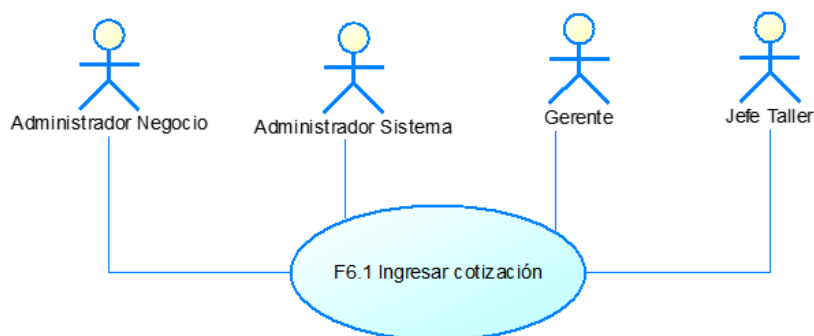
Figura 20 Gestión de cotizaciones.

Fuente: Propia.

Nombre: F6.1 Ingresar cotización.

Descripción: Permite que el usuario registre una cotización para un vehículo del taller con sus datos. Para tener un mejor acercamiento del proceso observe la *Figura 2*.

Actores: Gerente, Administrador del sistema, Administrador de negocio y Jefe de taller.



Flujo principal:

1. El usuario da clic en el menú de cotizaciones.
2. El usuario escoge la opción ingresar cotización.
3. El software muestra la pantalla de vehículos.
4. El usuario da clic en el botón de ingresar cotización del vehículo que desee.
5. El software muestra la pantalla de ingreso de cotización.
6. El usuario ingresa y selecciona los datos de la orden especificados en cada campo.
7. El usuario da clic en el botón de guardar.
8. El software verifica que los datos sean apropiados.
9. El software guarda el registro con los datos de la cotización.
10. El software retorna un mensaje de verificación.

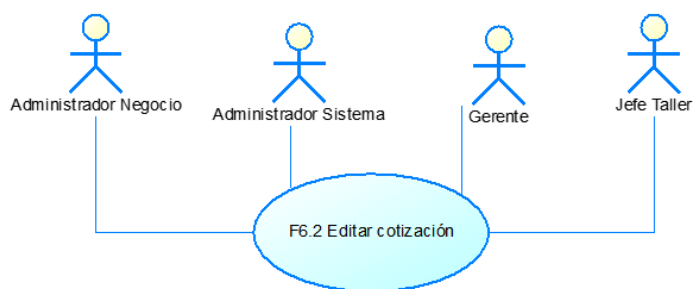
Flujo alterno:**Excepciones:**

| Causa | Mensaje |
|--|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |
| E3: Inconveniente con la validación de formato | Ingrese los datos de manera correcta. |
| E4: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

Nombre: F6.2 Editar cotización.

Descripción: Permite que el usuario modifique el registro de una cotización ya ingresada.

Actores: Gerente, Administrador del sistema, Administrador de negocio y Jefe de taller.



Flujo principal:

1. El usuario da clic en el menú de cotizaciones.
2. El usuario escoge la opción de consulta de cotizaciones.
3. El software presenta la pantalla de consulta.
4. El usuario consulta el registro por el filtro que desee.
5. El software presenta el registro deseado.
6. El usuario da clic en el botón de editar.
7. El software muestra la pantalla de edición con los datos precargados del registro seleccionado.
8. El usuario cambia los campos de datos que desee.
9. El usuario da clic en el botón editar.
10. El software cambia el registro.
11. El software presenta un mensaje de verificación.

Flujo alterno:

5) Ver caso de uso F6.3

Excepciones:

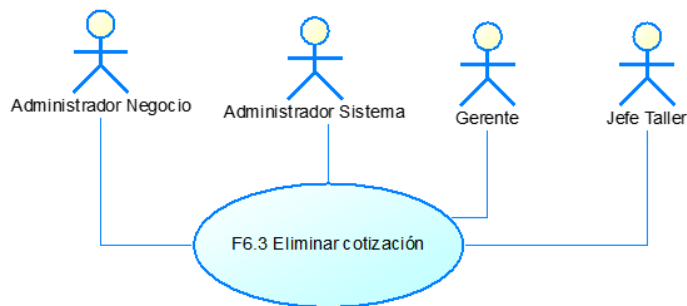
| Causa | Mensaje |
|---------------------------------------|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |

| | |
|--|---|
| E3: Inconveniente con la validación de formato | Ingrese los datos de manera correcta. |
| E4: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

Nombre: F6.3 Eliminar cotización.

Descripción: Permite que el usuario elimine el registro de información de una cotización.

Actores: Gerente, Administrador del sistema, Administrador de negocio y Jefe de taller.



Flujo principal:

1. El usuario da clic en el menú de cotizaciones.
2. El usuario escoge la opción de consulta de cotizaciones.
3. El software muestra la pantalla de consulta.
4. El usuario busca el registro por el filtro que desee.
5. El software muestra el registro deseado.
6. El usuario da clic en el botón de eliminar.
7. El software elimina el registro.
8. El software presenta un mensaje de verificación.

Flujo alternativo:

- 5) Ver caso de uso F6.2

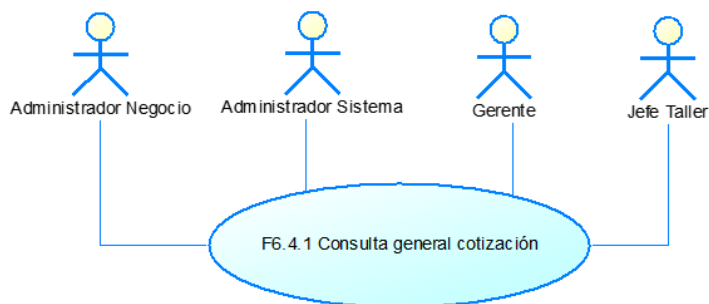
Excepciones:

| Causa | Mensaje |
|--|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |
| E3: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

Nombre: F6.4.1 Consulta general cotización.

Descripción: Permite que el usuario pueda consultar todos los registros que se tienen en el taller de cotizaciones.

Actores: Gerente, Administrador del sistema, Administrador de negocio y Jefe de taller.



Flujo principal:

1. El usuario da clic en el menú de cotizaciones.
2. El usuario escoge la opción de consulta cotizaciones.
3. El software presenta todos los registros de cotizaciones que tiene el taller.

Flujo alternativo:

3) Ver caso de uso F6.4.2

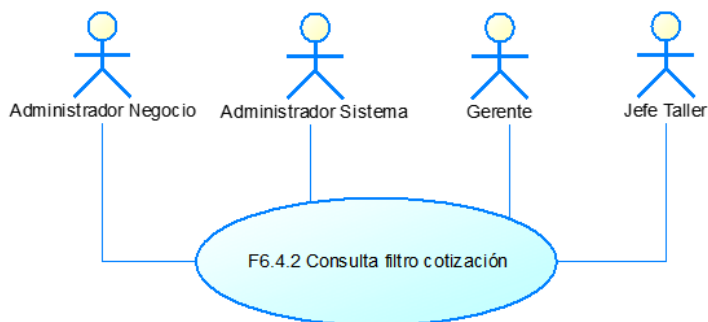
Excepciones:

| Causa | Mensaje |
|--|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |
| E3: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

Nombre: F6.4.2 Consulta filtro cotización.

Descripción: Permite que el usuario pueda consultar los registros de cotizaciones, filtrándolos por un campo seleccionado.

Actores: Gerente, Administrador del sistema, Administrador de negocio y Jefe de taller.



Flujo principal:

1. El usuario da clic en el menú de cotizaciones.
2. El usuario escoge la opción de consulta de cotizaciones.
3. El software presenta todos los registros de cotizaciones que tiene el taller.
4. El usuario escoge el campo por el que desea filtrar los registros.
5. El usuario ingresa la información del campo para filtrar los registros.

6. El usuario da clic en el botón buscar.
7. El software valida la información del campo.
8. El software muestra los registros filtrados.

Flujo alterno:

3) Ver caso de uso F6.4.1

Excepciones:

| Causa | Mensaje |
|--|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |
| E3: Inconveniente con la validación de formato | Ingrese los datos de manera correcta. |
| E4: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

2. Gestión de compras

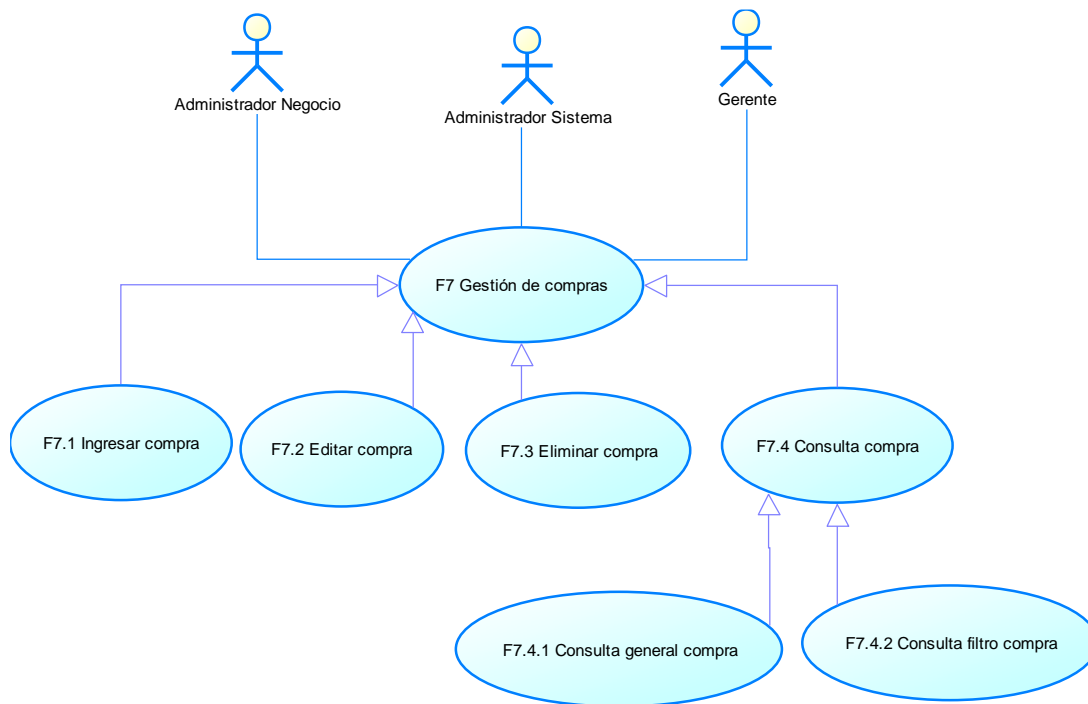


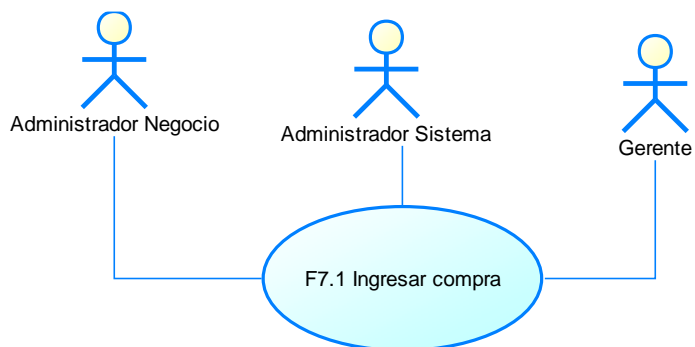
Figura 21 Gestión de compras.

Fuente: Propia.

Nombre: F7.1 Ingresar compra.

Descripción: Permite que el usuario registre una compra que se haya realizado. Para tener un mejor acercamiento del proceso observe la *Figura 2*.

Actores: Gerente, Administrador de negocio y Administrador del sistema.



Flujo principal:

1. El usuario da clic en el menú de compras.
2. El usuario escoge la opción ingresar compra.
3. El software muestra la pantalla de ingreso de compra.
4. El usuario registra los datos del cliente especificados en cada campo.
5. El usuario da clic en el botón de guardar.
6. El software valida que los datos sean correctos.
7. El software guarda los datos de la compra.
8. El software retorna un mensaje de verificación.

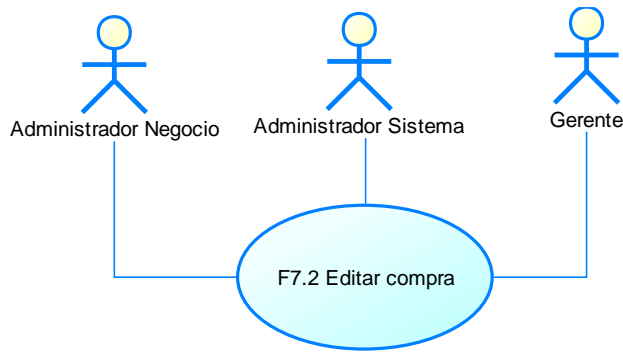
Flujo alternativo:**Excepciones:**

| Causa | Mensaje |
|--|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |
| E3: Inconveniente con la validación de formato | Ingrese los datos de manera correcta. |
| E4: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

Nombre: F7.2 Editar compra.

Descripción: Permite que el usuario modifique el registro de una compra del taller ya ingresada.

Actores: Gerente, Administrador de negocio y Administrador del sistema.



Flujo principal:

1. El usuario da clic en el menú de compras
2. El usuario escoge la opción de consulta de compras.
3. El software presenta la pantalla de consulta.
4. El usuario consulta el registro por el filtro que desee.
5. El software presenta el registro deseado.
6. El usuario da clic en el botón de editar.
7. El software muestra la pantalla de edición con los datos precargados del registro.
8. El usuario edita los campos de datos que desee.
9. El usuario da clic en el botón editar.
10. El software cambia el registro.
11. El software presenta un mensaje de verificación.

Flujo alternativo:

- 4) Ver caso de uso F7.3

Excepciones:

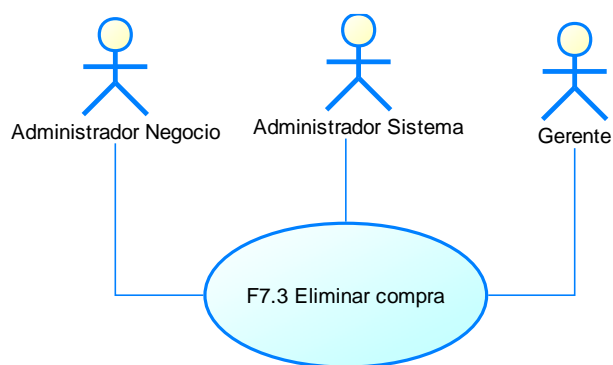
| Causa | Mensaje |
|---------------------------------------|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |

| | |
|--|---|
| E3: Inconveniente con la validación de formato | Ingrese los datos de manera correcta. |
| E4: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

Nombre: F7.3 Eliminar compra.

Descripción: Permite que el usuario elimine el registro de información de una compra.

Actores: Gerente, Administrador de negocio y Administrador del sistema.



Flujo principal:

1. El actor da clic en el menú de compras.
2. El usuario escoge la opción de consulta de compras.
3. El software presenta la pantalla de consulta.
4. El actor consulta el registro por el filtro que desee.
5. El software presenta el registro deseado.
6. El usuario da clic en el botón de eliminar.
7. El software muestra un mensaje de confirmación de eliminación.
8. El usuario da clic en el botón aceptar.
9. El software elimina el registro de la compra.
10. El software presenta un mensaje de verificación.

Flujo alternativo:

5) Ver caso de uso F7.2

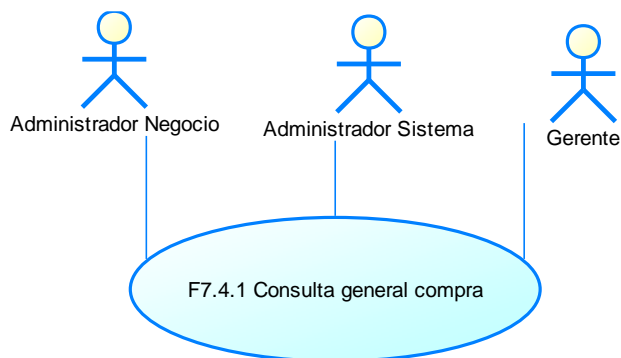
Excepciones:

| Causa | Mensaje |
|--|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |
| E3: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

Nombre: F7.4.1 Consulta general compra.

Descripción: Permite que el usuario pueda consultar todos los registros que se tienen en el taller de compras.

Actores: Gerente, Administrador de negocio y Administrador del sistema.



Flujo principal:

1. El usuario da clic en el menú de compras.
2. El usuario escoge la opción de consulta de compras.
3. El software presenta todos los registros de compras que tiene el taller.

Flujo alterno:

3) Ver caso de uso F7.4.2

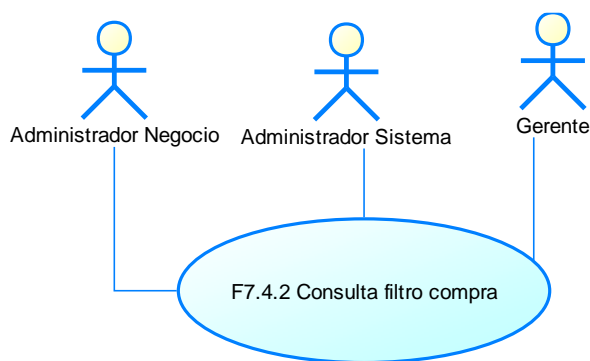
Excepciones:

| Causa | Mensaje |
|--|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |
| E3: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

Nombre: F7.4.2 Consulta filtro compra.

Descripción: Permite que el usuario pueda consultar los registros de compras filtrándolos por un campo seleccionado.

Actores: Gerente, Administrador de negocio y Administrador del sistema.



Flujo principal:

1. El usuario da clic en el menú de compras.

2. El usuario escoge la opción de consulta de compras.
3. El software presenta todos los registros de compras que tiene el taller.
4. El usuario escoge el campo por el que desea filtrar los registros.
5. El usuario ingresa la información del campo para filtrar los registros.
6. El usuario da clic en el botón buscar.
7. El software valida la información del campo.
8. El software muestra los registros filtrados.

Flujo alterno:

3) Ver caso de uso F7.4.1

Excepciones:

| Causa | Mensaje |
|--|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |
| E3: Inconveniente con la validación de formato | Ingrese los datos de manera correcta. |
| E4: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

3. Gestión de facturas

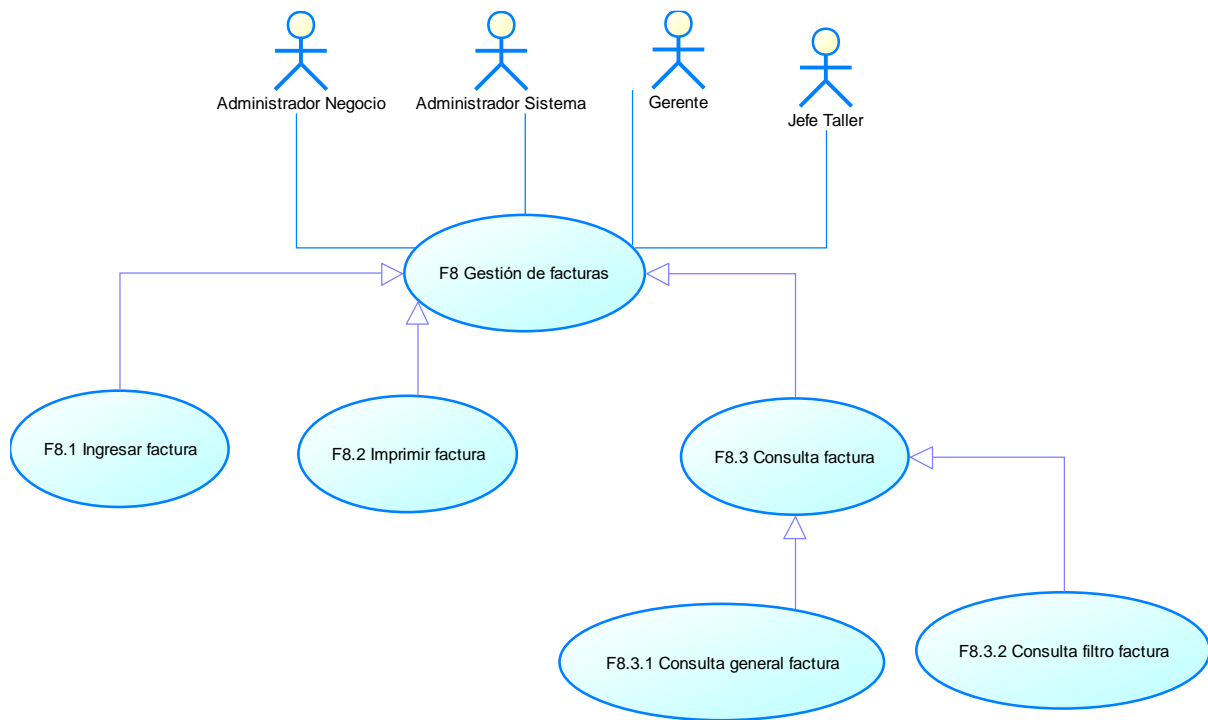


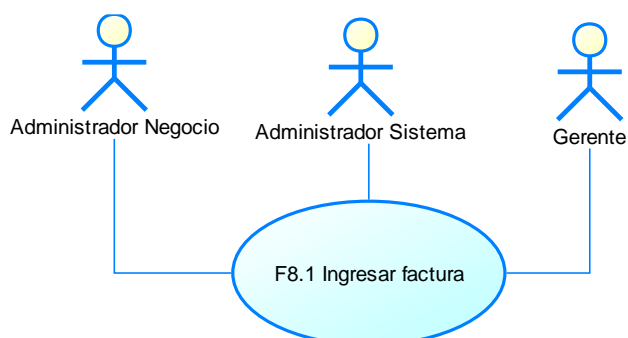
Figura 22 Gestión de facturas.

Fuente: Propia.

Nombre: F8.1 Ingresar factura.

Descripción: Permite que el usuario registre una factura para el cliente. Para tener un mejor acercamiento del proceso observe la *Figura 2*.

Actores: Gerente, Administrador de negocio y Administrador del sistema.



Flujo principal:

1. El usuario da clic en el menú de facturas.
2. El usuario escoge la opción generar factura.
3. El software muestra la pantalla de consulta de cotizaciones.
4. El usuario elije la cotización que desea.
5. El software muestra la cotización seleccionada.
6. El usuario da clic en el botón de factura.
7. El software muestra la factura generada.
8. El usuario da clic en el pulsa guardar.
9. El software valida la información.
10. El software guarda el registro de la factura.

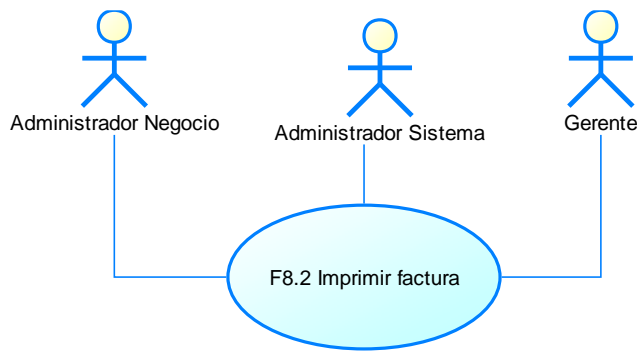
Flujo alterno:**Excepciones:**

| Causa | Mensaje |
|--|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |
| E3: Inconveniente con la validación de formato | Ingrese los datos de manera correcta. |
| E4: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

Nombre: F8.2 Imprimir factura.

Descripción: Permite que el usuario imprima el registro de una factura.

Actores: Gerente, Administrador de negocio y Administrador del sistema.



Flujo principal:

1. El usuario da clic en el menú de facturas.
2. El usuario escoge la opción de consulta de facturas.
3. El software muestra la pantalla de consulta.
4. El usuario busca el registro por el filtro que desee.
5. El software muestra el registro deseado.
6. El usuario da clic en el botón de ver.
7. El software muestra la factura precargada.
8. El usuario da clic en el botón imprimir.
9. El software muestra el cuadro de impresión.

Flujo alterno:

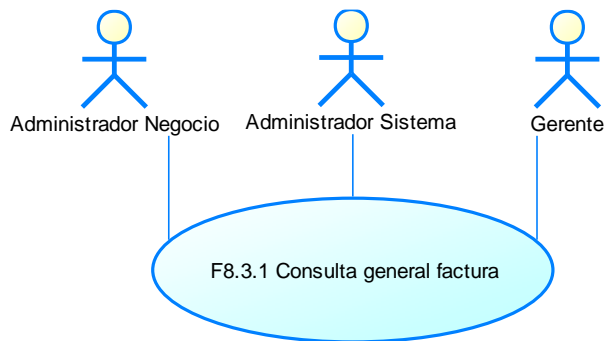
Excepciones:

| Causa | Mensaje |
|--|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |
| E3: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

Nombre: F8.3.1 Consulta general factura.

Descripción: Permite que el usuario pueda consultar todos los registros que se tienen en el taller de facturas.

Actores: Gerente, Administrador de negocio y Administrador del sistema.



Flujo principal:

1. El usuario da clic en el menú de facturas.
2. El usuario escoge la opción de consulta.
3. El software muestra todos los registros de facturas que tiene el taller.
4. El usuario da clic en sobre botón de ver.
5. El software muestra la pantalla con la información de la factura seleccionada.

Flujo alterno:

- 3) Ver caso de uso F8.3.2

Excepciones:

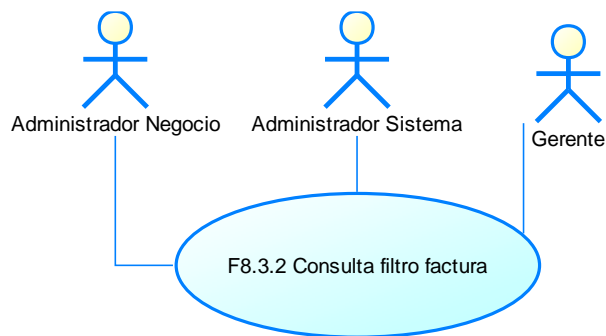
| Causa | Mensaje |
|---------------------------------------|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |

| | |
|--|---|
| E3: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |
|--|---|

Nombre: F8.3.2 Consulta filtro factura.

Descripción: Permite que el usuario pueda consultar los registros de facturas filtrándolos por un campo seleccionado.

Actores: Gerente, Administrador de negocio y Administrador del sistema.



Flujo principal:

1. El usuario da clic en el menú de facturas.
2. El usuario escoge la opción de consulta.
3. El software presenta todos los registros de facturas que tiene el taller.
4. El usuario elije el campo por el que desea filtrar los registros.
5. El usuario ingresa la información del campo para filtrar los registros.
6. El usuario da clic en el botón buscar.
7. El software valida la información del campo.
8. El software muestra los registros filtrados.
9. El usuario da clic en el botón de ver.
10. El software muestra la pantalla con la información de la factura seleccionada.

Flujo alterno:

3) Ver caso de uso F8.3.1

Excepciones:

| Causa | Mensaje |
|--|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |
| E3: Inconveniente con la validación de formato | Ingrese los datos de manera correcta. |
| E4: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

4. Gestión de turnos

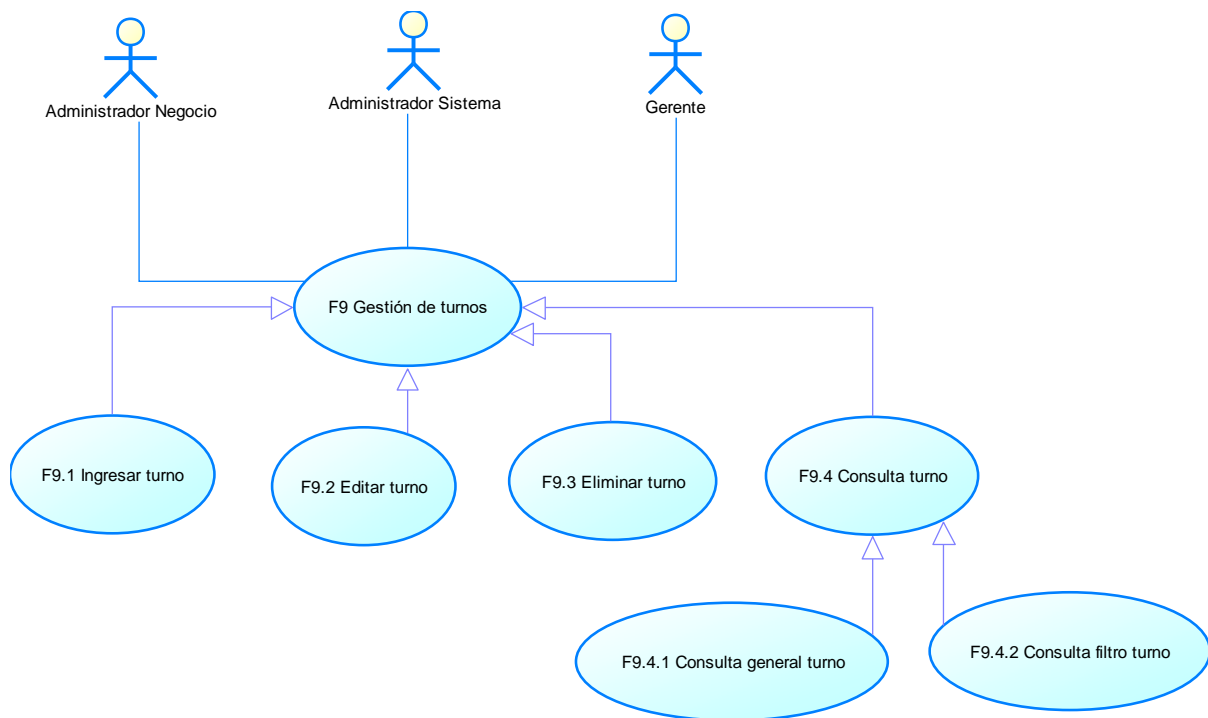


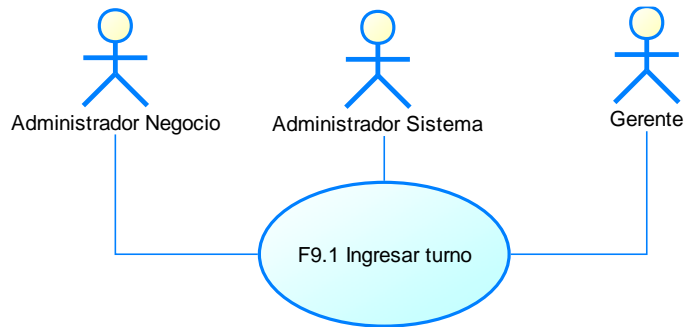
Figura 23 Gestión de turnos.

Fuente: Propia.

Nombre: F9.1 Ingresar turno.

Descripción: Permite que el usuario registre un turno asignado a un vehículo del taller. Para tener un mejor acercamiento del proceso observe la *Figura 3*.

Actores: Gerente, Administrador de negocio y Administrador del sistema.



Flujo principal:

1. El actor da clic en el menú de turnos.
2. El usuario escoge la opción asignar turno.
3. El software muestra la pantalla de consulta de clientes.
4. El usuario da clic en el botón de turnos del cliente seleccionado.
5. El software muestra un calendario para que se le asigne la fecha.
6. El usuario elije la fecha y presiona el botón aceptar.
7. El software muestra la pantalla de ingreso de turno.
8. El usuario ingresa y selecciona los datos para el turno.
9. El usuario da clic en el botón guardar.
10. El software valida los datos.
11. El software muestra una notificación de verificación.

Flujo alternativo:

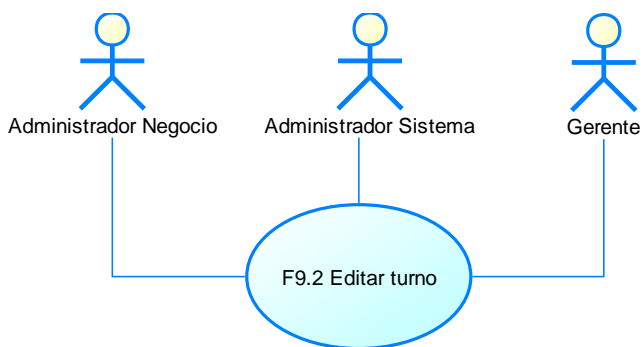
Excepciones:

| Causa | Mensaje |
|--|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |
| E3: Inconveniente con la validación de formato | Ingrese los datos de manera correcta. |
| E4: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

Nombre: F9.2 Editar turno.

Descripción: Permite que el usuario modifique el registro de un turno de un vehículo ya ingresado.

Actores: Gerente, Administrador de negocio y Administrador del sistema.

**Flujo principal:**

1. El usuario da clic en el menú de turnos.
2. El usuario escoge la opción de consulta de turnos.
3. El software muestra la pantalla de consulta.
4. El usuario busca el registro por el filtro que desee.
5. El software muestra el registro deseado.
6. El usuario da clic en el botón de editar.

7. El software muestra la pantalla de edición con los datos precargados del registro.
8. El usuario cambia los datos que desee del turno.
9. El usuario da clic en el botón editar.
10. El software cambia el registro del turno.
11. El software muestra un mensaje de verificación.

Flujo alternativo:

4) Ver caso de uso F9.3

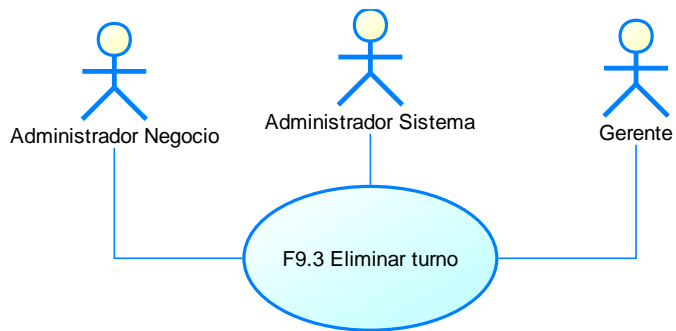
Excepciones:

| Causa | Mensaje |
|--|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |
| E3: Inconveniente con la validación de formato | Ingrese los datos de manera correcta. |
| E4: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

Nombre: F9.3 Eliminar turno.

Descripción: Permite que el usuario elimine el registro de un turno ya registrado.

Actores: Gerente, Administrador de negocio y Administrador del sistema.



Flujo principal:

1. El usuario da clic en el menú de turnos.
2. El usuario escoge la opción de consulta de turnos.
3. El software muestra la pantalla de consulta.
4. El usuario busca el registro por el filtro que desee.
5. El software muestra el registro deseado.
6. El usuario da clic en el botón de eliminar.
7. El software elimina el registro del turno.
8. El software muestra un mensaje de verificación.

Flujo alterno:

- 4) Ver caso de uso F9.2

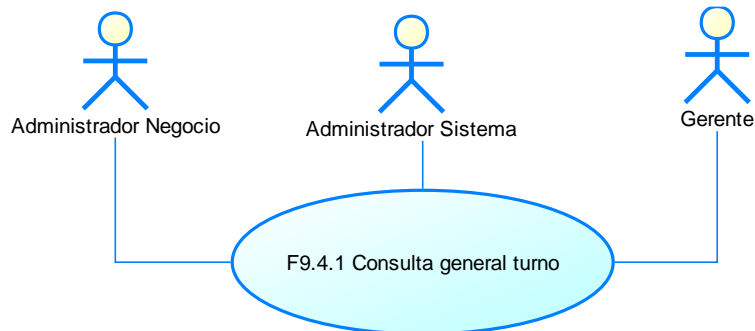
Excepciones:

| Causa | Mensaje |
|--|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |
| E3: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

Nombre: F9.4.1 Consulta general turno.

Descripción: Permite que el usuario pueda consultar todos los turnos que se tienen en el taller de vehículos.

Actores: Gerente, Administrador de negocio y Administrador del sistema.



Flujo principal:

1. El usuario da clic en el menú de turnos.
2. El usuario escoge la opción de consulta de turnos.
3. El software presenta todos los registros de turnos que tiene el taller.

Flujo alternativo:

- 3) Ver caso de uso F9.4.2

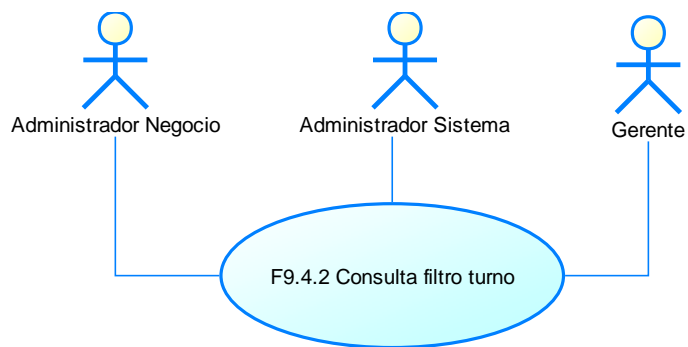
Excepciones:

| Causa | Mensaje |
|--|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |
| E3: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

Nombre: F9.4.2 Consulta filtro turno.

Descripción: Permite que el usuario pueda consultar los registros de turnos filtrándolos por un campo seleccionado.

Actores: Gerente, Administrador de negocio y Administrador del sistema.



Flujo principal:

1. El usuario da clic en el menú de turnos.
2. El usuario escoge la opción de consulta de turnos.
3. El software muestra todos los registros de turnos que tiene el taller.
4. El usuario elije el campo por el que desea filtrar los registros.
5. El usuario ingresa la información del campo para filtrar los registros de turnos.
6. El usuario da clic en el botón buscar.
7. El software valida la información del campo.
8. El software muestra los registros filtrados.

Flujo alternativo:

- 3) Ver caso de uso F9.4.1

Excepciones:

| Causa | Mensaje |
|--|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |
| E3: Inconveniente con la validación de formato | Ingrese los datos de manera correcta. |
| E4: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

5. Gestión de notificaciones para clientes

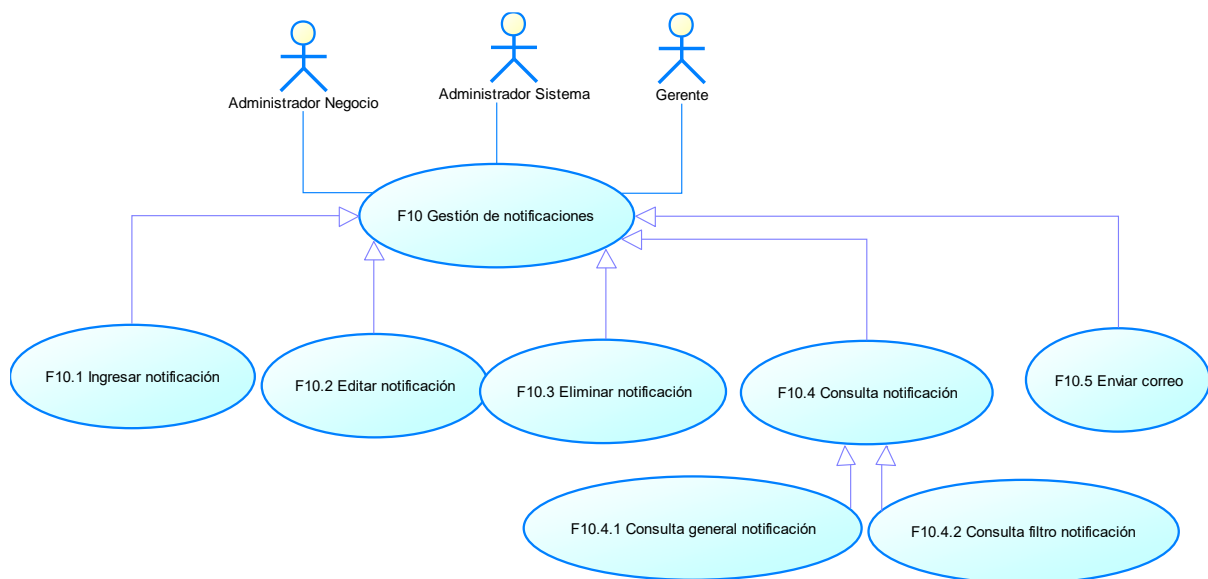


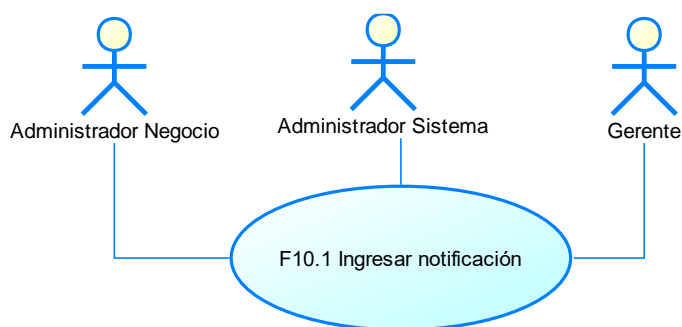
Figura 24 Gestión de notificaciones.

Fuente: Propia.

Nombre: F10.1 Ingresar notificación.

Descripción: Permite que el usuario registre una notificación asignado a un vehículo del taller. Para tener un mejor acercamiento del proceso observe la *Figura 4*.

Actores: Gerente, Administrador de negocio y Administrador del sistema.



Flujo principal:

1. El usuario genera una factura con los pasos anteriormente especificados.
2. El software presenta la pantalla con la factura
3. El usuario da clic en el botón de notificación.
4. El software presenta el recuadro de la notificación.
5. El usuario ingresa y selecciona los datos necesarios.
6. El usuario da clic en el botón aceptar.
7. El software verifica los datos.
8. El software muestra una notificación de verificación.

Flujo alternativo:

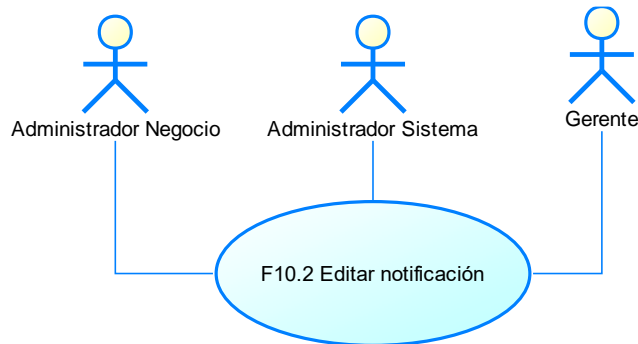
Excepciones:

| Causa | Mensaje |
|--|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |
| E3: Inconveniente con la validación de formato | Ingrese los datos de manera correcta. |
| E4: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

Nombre: F10.2 Editar notificación.

Descripción: Permite que el usuario modifique el registro de una notificación de un vehículo ya ingresado.

Actores: Gerente, Administrador de negocio y Administrador del sistema.



Flujo principal:

1. El usuario da clic en el menú de notificaciones.
2. El usuario escoge la opción de consulta de notificaciones.
3. El software muestra la pantalla de consulta.
4. El usuario busca el registro por el filtro que desee.
5. El software muestra el registro deseado.
6. El usuario da clic en el botón de editar.
7. El software muestra el recuadro de edición con los datos precargados del registro.
8. El usuario cambia los campos que desee de la notificación.
9. El usuario da clic en el botón aceptar.
10. El software cambia el registro de la notificación.
11. El software muestra un mensaje de verificación.

Flujo alternativo:

- 4) Ver caso de uso F10.3

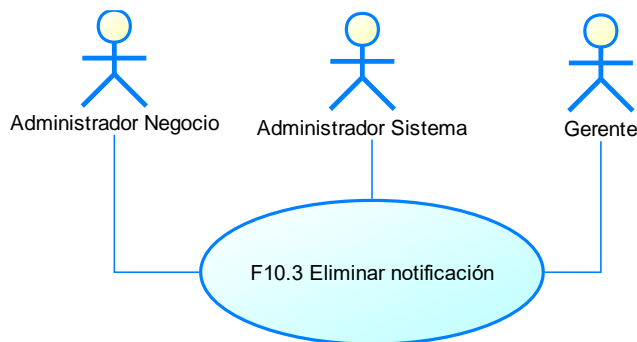
Excepciones:

| Causa | Mensaje |
|--|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |
| E3: Inconveniente con la validación de formato | Ingrese los datos de manera correcta. |
| E4: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

Nombre: F10.3 Eliminar notificación.

Descripción: Permite que el usuario elimine el registro de una notificación ya registrada.

Actores: Gerente, Administrador de negocio y Administrador del sistema.



Flujo principal:

1. El usuario da clic en el menú de notificaciones
2. El usuario escoge la opción de consulta de notificaciones.
3. El software muestra la pantalla de consulta.
4. El usuario busca el registro por el filtro que desee.
5. El software muestra el registro deseado.

6. El usuario da clic en el botón de eliminar.
7. El software elimina el registro de la notificación.
8. El software muestra un mensaje de verificación.

Flujo alternativo:

4) Ver caso de uso F10.2

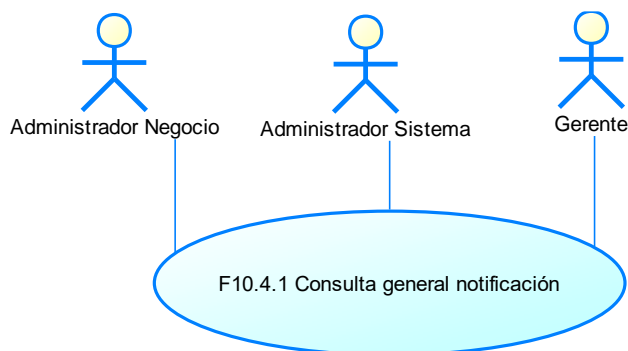
Excepciones:

| Causa | Mensaje |
|--|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |
| E3: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

Nombre: F10.4.1 Consulta general notificación.

Descripción: Permite que el usuario pueda consultar todas las notificaciones que se tienen en el taller de vehículos.

Actores: Gerente, Administrador de negocio y Administrador del sistema.



Flujo principal:

1. El usuario da clic en el menú de notificaciones.
2. El usuario escoge la opción de consulta de notificaciones.
3. El software presenta todos los registros de notificaciones que tiene el taller.

Flujo alternativo:

- 3) Ver caso de uso F10.4.2

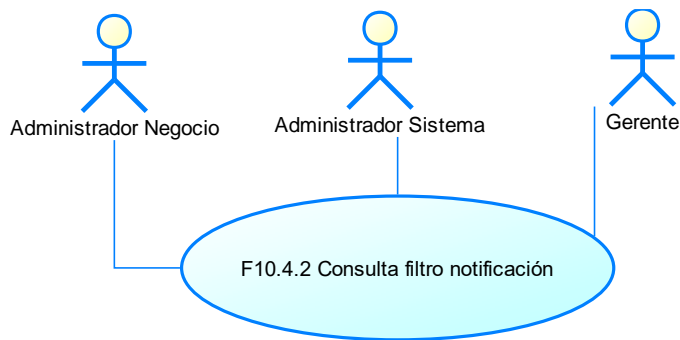
Excepciones:

| Causa | Mensaje |
|--|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |
| E3: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

Nombre: F10.4.2 Consulta filtro notificación.

Descripción: Permite que el usuario pueda consultar los registros de notificaciones filtrándolos por un campo seleccionado.

Actores: Gerente, Administrador de negocio y Administrador del sistema.



Flujo principal:

1. El usuario da clic en el menú de notificaciones.
2. El usuario escoge la opción de consulta de notificaciones.
3. El software muestra todos los registros de notificaciones que tiene el taller.
4. El usuario elije el campo por el que desea filtrar los registros.
5. El usuario escribe la información del campo para filtrar los registros.
6. El usuario da clic en el botón buscar.
7. El software valida la información del campo.
8. El software muestra los registros filtrados.

Flujo alterno:

- 3) Ver caso de uso F10.4.1

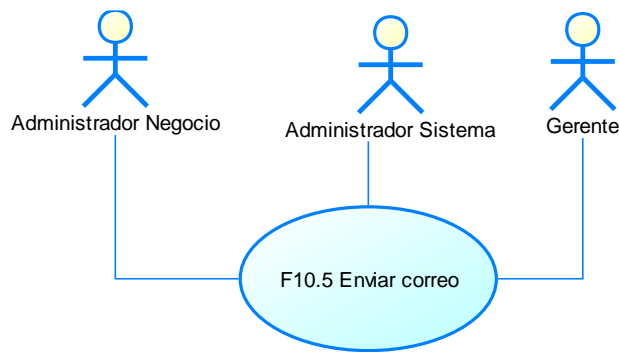
Excepciones:

| Causa | Mensaje |
|--|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |
| E3: Inconveniente con la validación de formato | Ingrese los datos de manera correcta. |
| E4: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

Nombre: F10.5 Enviar correo.

Descripción: Permite que el usuario pueda enviar un correo electrónico desde el sistema.

Actores: Gerente, Administrador de negocio y Administrador del sistema.



Flujo principal:

1. El usuario da clic en el menú de notificaciones.
2. El usuario escoge la opción de enviar correo.
3. El software muestra la pantalla de envío de correo.
4. El usuario ingresa los datos para enviar el correo electrónico.
5. El usuario da clic en el botón enviar.
6. El software valida los datos.
7. El software muestra una notificación de verificación.

Flujo alterno:

Excepciones:

| Causa | Mensaje |
|--|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |
| E3: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

6. Gestión de promociones y descuentos

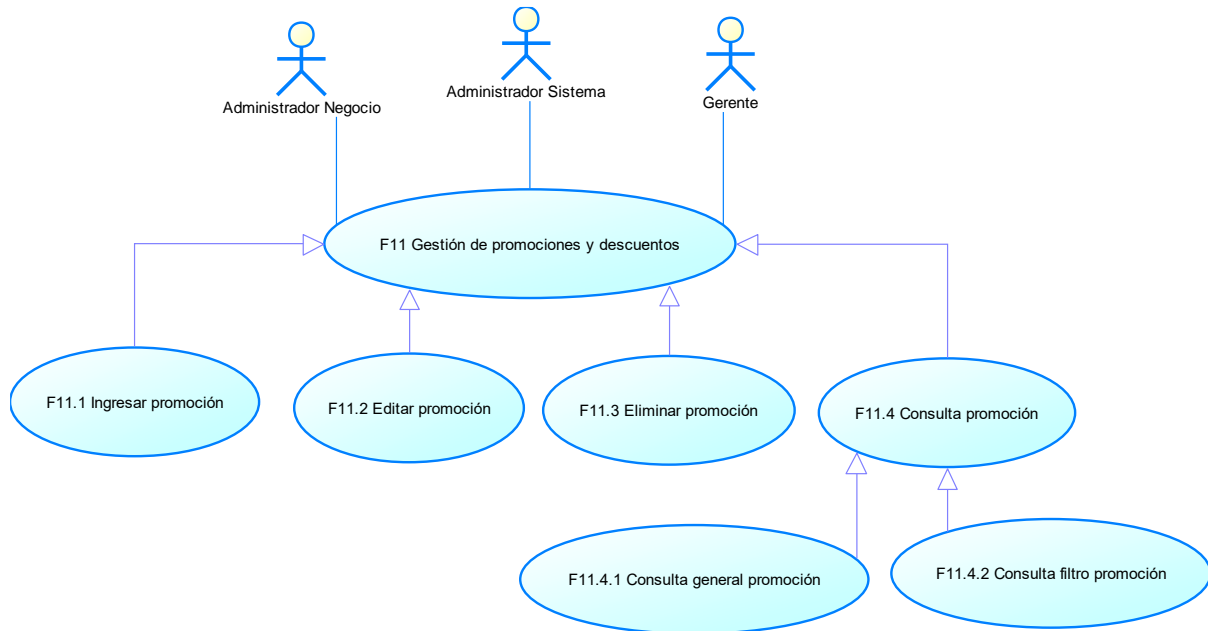


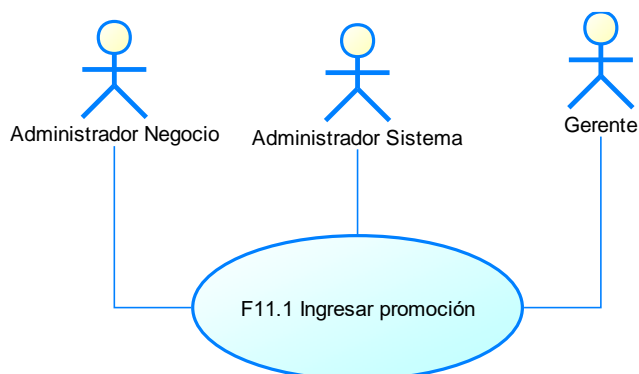
Figura 25 Gestión de promociones y descuentos.

Fuente: Propia.

Nombre: F11.1 Ingresar promoción.

Descripción: Permite que el usuario registre una promoción del taller. Para tener un mejor acercamiento del proceso observe la *Figura 4*.

Actores: Gerente, Administrador de negocio y Administrador del sistema.



Flujo principal:

1. El usuario da clic en el menú de promociones.
2. El usuario escoge la opción ingresar.
3. El software muestra la pantalla de ingreso de promociones.
4. El usuario ingresa y selecciona los datos para el registro de la notificación especificados en cada campo.
5. El usuario da clic en el botón de ingresar.
6. El software valida que los datos sean correctos.
7. El software guarda el registro de la promoción.
8. El software retorna un mensaje de verificación.

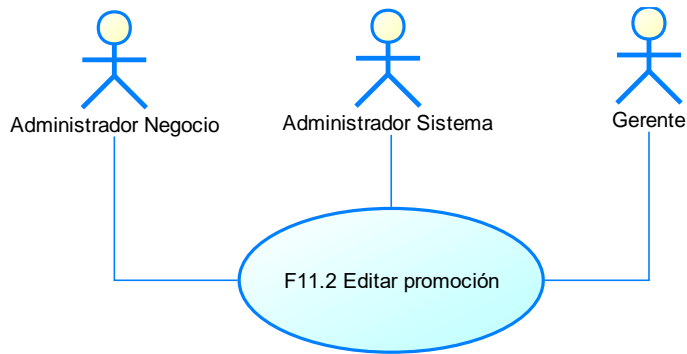
Flujo alterno:**Excepciones:**

| Causa | Mensaje |
|--|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |
| E3: Inconveniente con la validación de formato | Ingrese los datos de manera correcta. |
| E4: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

Nombre: F11.2 Editar promoción.

Descripción: Permite que el usuario modifique el registro de una promoción ya ingresada.

Actores: Gerente, Administrador de negocio y Administrador del sistema.



Flujo principal:

1. El usuario da clic en el menú de promociones.
2. El usuario escoge la opción de consulta de promociones.
3. El software muestra la pantalla de consulta.
4. El usuario busca el registro por el filtro que desee.
5. El software muestra el registro deseado.
6. El usuario da clic en el botón de editar.
7. El software muestra la pantalla de edición con los datos precargados del registro.
8. El usuario cambia los campos que desee de la promoción.
9. El usuario da clic en el botón guardar.
10. El software cambia el registro de la promoción.
11. El software muestra un mensaje de verificación.

Flujo alterno:

- 4) Ver caso de uso F11.3

Excepciones:

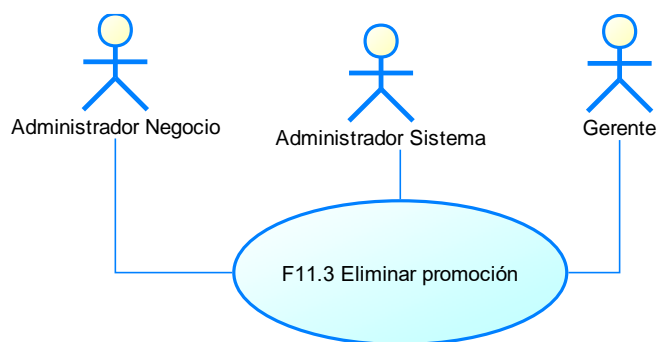
| Causa | Mensaje |
|---------------------------------------|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |

| | |
|--|---|
| E3: Inconveniente con la validación de formato | Ingrese los datos de manera correcta. |
| E4: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

Nombre: F11.3 Eliminar promoción.

Descripción: Permite que el usuario elimine el registro de una promoción ya registrada.

Actores: Gerente, Administrador de negocio y Administrador del sistema.



Flujo principal:

1. El usuario da clic en el menú de promociones
2. El usuario escoge la opción de consulta de promociones.
3. El software muestra la pantalla de consulta.
4. El usuario busca el registro por el filtro que desee.
5. El software muestra el registro deseado.
6. El usuario da clic en el botón de eliminar.
7. El software elimina el registro de la promoción.
8. El software presenta un mensaje de verificación de eliminación.

Flujo alternativo:

4) Ver caso de uso F11.2

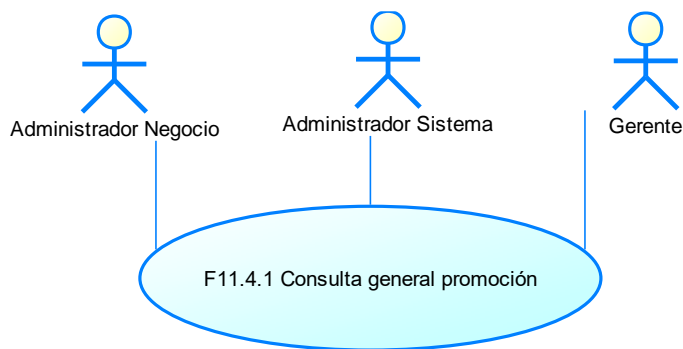
Excepciones:

| Causa | Mensaje |
|--|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |
| E3: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

Nombre: F11.4.1 Consulta general promoción.

Descripción: Permite que el usuario pueda consultar todas las promociones que se tienen en el taller de vehículos.

Actores: Gerente, Administrador de negocio y Administrador del sistema.



Flujo principal:

1. El usuario da clic en el menú de promociones.
2. El usuario escoge la opción de consulta.
3. El software presenta todos los registros de promociones que tiene el taller.

Flujo alterno:

3) Ver caso de uso F11.4.2

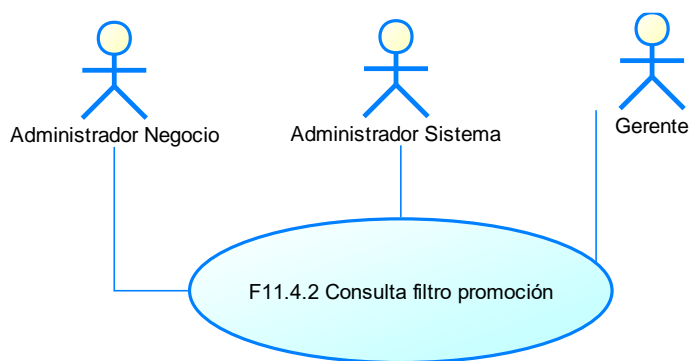
Excepciones:

| Causa | Mensaje |
|--|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |
| E3: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

Nombre: F11.4.2 Consulta filtro promoción.

Descripción: Permite que el usuario pueda consultar los registros de promociones filtrándolos por un campo seleccionado.

Actores: Gerente, Administrador de negocio y Administrador del sistema.



Flujo principal:

1. El usuario da clic en el menú de promociones.

2. El usuario escoge la opción de consulta.
3. El software muestra todos los registros de promociones que tiene el taller.
4. El usuario elige el campo por el que desea filtrar los registros.
5. El usuario escribe la información del campo para filtrar los registros de promociones.
6. El usuario da clic en el botón buscar.
7. El software valida la información del campo.
8. El software muestra los registros filtrados.

Flujo alterno:

3) Ver caso de uso F11.4.1

Excepciones:

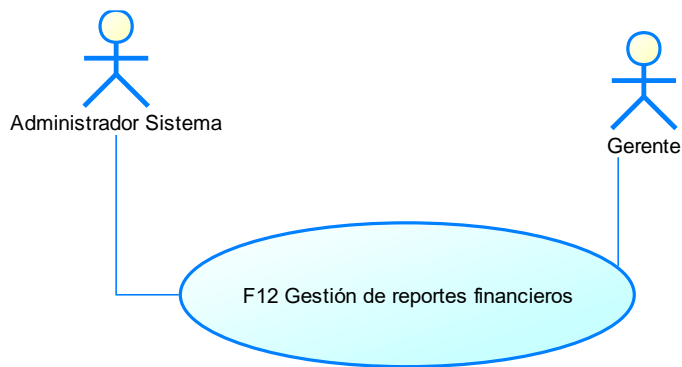
| Causa | Mensaje |
|--|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |
| E3: Inconveniente con la validación de formato | Ingrese los datos de manera correcta. |
| E4: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

7. Gestión de reportes financieros

Nombre: F12 Consulta de reportes.

Descripción: Permite que el usuario pueda visualizar los ingresos y egresos del taller en un período de tiempo determinado.

Actores: Administrador de sistema y Gerente.



Flujo principal:

1. El usuario da clic en el menú de control.
2. El software presenta la pantalla de control.
3. El usuario escoge el filtro de tiempo necesitado.
4. El software presenta el registro de ingresos y egresos en el tiempo delimitado.

Flujo alterno:

Excepciones:

| Causa | Mensaje |
|--|---|
| E1: Inconveniente con la BBDD | Comuníquese con el administrador del sistema. |
| E2: Inconveniente con la verificación | Ingrese sus credenciales correctamente. |
| E3: Inconveniente con la validación de formato | Ingrese los datos de manera correcta. |
| E4: Inconveniente con la ruta de la pantalla | Comuníquese con el administrador del sistema. |

Requisitos no funcionales

Requisitos de rendimiento

- Se necesitará de una conexión a red estable para que el sistema no presente errores en cuanto a todo su funcionamiento ya que está orientado a la web y a dispositivos móviles.
- Debido a que el sistema será usado por los empleados que disponga el taller, dependerá de este número de personas las prestaciones que necesite el servidor y la conexión a la red.
- Cualquier transición entre ventanas, despliegue de mensajes, gráficos, contenedores, etc. Debe tomar un tiempo muy reducido como respuesta para una buena experiencia para el usuario.

Seguridad

- Se modificarán y/o desactivarán los mensajes de error que se envíen desde el servidor y que puedan representar una inseguridad para los datos del sistema.
- Se implementará un Ingreso al sistema para proteger el acceso al sistema por parte de usuarios no admitidos. De esta manera la información que ingrese y salga no se verá afectada.
- Se filtrarán los datos que entren al sistema mediante el control de éstos, esto quiere decir que se obligará a que el usuario ingrese los datos con las medidas proporcionadas para evitar posibles fallos.
- Las credenciales de acceso solo estarán ubicadas en el equipo en el que se encuentre el servidor por lo que no podrán ser visibles en ningún sentido.

Fiabilidad

- Las conexiones entre el cliente y el servidor deben ser continuas para que no exista información desechada cuando se esté enviando y recibiendo información.

Disponibilidad

- Debido a que el sistema será utilizado durante el tiempo de trabajo del taller, el servidor deberá estar funcionando en este lapso. No se necesita que el servidor se encuentre encendido todo el día ya que el sistema no va a ser público.

- De igual manera, el motor de base de datos debe estar activo antes que el servidor, para que este pueda interactuar con la base de datos del sistema.
- Cabe recalcar que el servidor de apache también debe estar activo antes que se inicie el servidor ya que sin este no tendrá funcionamiento.

Mantenibilidad

- Las interfaces del sistema deben estar claramente complementadas con mensajes de ayuda para que el personal que no tenga experiencia en el uso de éste pueda hacer uso de sus funcionalidades sin mayor problema.
- Se deberán de seguir estándares en cuanto a la codificación, nombramiento de variables, funciones, etc. Para que la mantenibilidad del sistema a futuro no sea complicada y se siga según los lineamientos.

Portabilidad

- El servidor debe estar instalado en un equipo con apache para su funcionamiento.
- Los dispositivos móviles que vayan a hacer uso del sistema deberán de tener sistema operativo Android o IOS, y ser compatibles con la versión del aplicativo móvil.
- En todo momento se necesita conexión a internet para que el sistema pueda funcionar.

Diagrama conceptual

Gracias a este tipo de diagramas, se logra modelar la estructura posible del sistema a un nivel de abstracción alto. Describen lo que debe estar dentro de la arquitectura sin tomar en cuenta detalles.

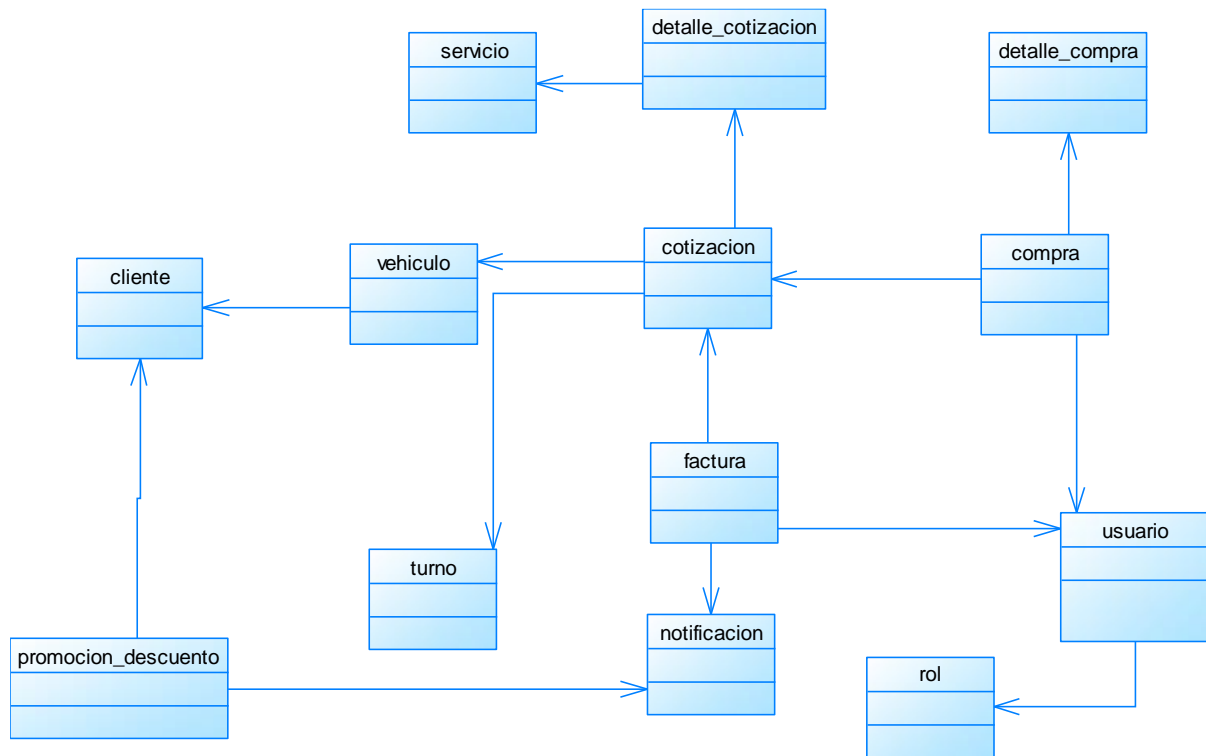


Figura 26 Diagrama conceptual del sistema.

Fuente: Propia.

Análisis de herramientas de desarrollo

Para poder desarrollar el sistema se utilizarán herramientas que nos ayudarán en cuanto a la codificación, administración de la base de datos, interpretar el código, entre otras acciones necesarias para la culminación del software.

Basándose en varias de las ventajas y facilidades que aportan, se han escogido las siguientes herramientas que serán de gran apoyo para la realización del sistema.

Visual Studio Code

Esta herramienta es un editor de código muy poderoso y liviano, también es multiplataforma por lo que se puede utilizar en varios sistemas operativos. Este software tiene como ventaja su soporte de JavaScript, TypeScript y Node.js, aparte de que posee extensiones para varios lenguajes incluyendo PHP. Gracias a estas particularidades, Visual Studio Code será de gran ayuda en cuanto a la codificación del sistema en sí. Como ya se sabe, debido a que es desarrollo orientado a la web, solo se necesita de un editor de texto y un navegador para que el sistema pueda funcionar, por lo que esta herramienta será perfecta para codificar todo el sistema tomando en cuenta los lenguajes que se utilizaran que son: JavaScript, PHP, CSS y HTML. Esta herramienta es gratuita y muy accesible para cualquier persona.

Otra ventaja que ofrece este programa es su apoyo en cuanto a la escritura, contiene patrones de colores que ayudarán a identificar palabras reservadas de los distintos lenguajes, además de un corrector de estas lo cual facilita muchísimo la codificación.

Este software solo estará instalado en el o los equipos en los que se desarrollará el sistema, mas no en los equipos que lo utilizarán.

Mozilla Firefox

Este es un navegador web de código abierto multiplataforma el cual posee varias herramientas que aportarán mucho en cuanto al desarrollo del sistema. Como ya se sabe para que el sistema sea interpretado se necesita un navegador web y Mozilla es uno de los más conocidos y utilizados en la actualidad.

Gracias a su compatibilidad con varios lenguajes de programación, podrá ejecutar el sistema perfectamente al igual que cada uno de sus componentes. También posee una interfaz muy completa y fácil de utilizar, la cual tiene funciones para el desarrollador que ayudarán para verificar la funcionalidad del software.

Además, es compatible con dispositivos móviles y tiene una gran utilidad para la visualización de aplicaciones móviles, manteniendo todas las características que tiene para el desarrollo web.

PhpMyAdmin

Es una herramienta de software que está escrita en PHP, permite la administración de la base de datos MySQL en la web, permitiendo realizar varias operaciones además de las frecuentes en otros administradores de motores de bases de datos.

Una de sus grandes ventajas es la amplia documentación que posee y su interfaz intuitiva, características que facilitarán el manejo de la base de datos y permitirán un gran desempeño para el desarrollo del sistema. Se puede crear gráficos para el diseño de las bases de datos y así, poder formar una arquitectura correcta de la misma.

Debido a que previamente al desarrollo del sistema se deberá instalar el servidor de apache, este ya viene incluido con el administrador de PhpMyAdmin, por lo que no se necesitará de una instalación particular. Cabe recalcar que este programa si debe estar instalado en las computadoras en las que se utilizará el sistema, ya que será de gran ayuda para el manejo de la base de datos directamente.

Power Designer

Esta es una potente herramienta para el modelado y administración de datos, arquitecturas, entre otros, que ayudarán en cuanto a los diseños de sistema. Power Designer es un software muy potente que posee grandes funcionalidades y permitirán construir las bases del sistema para poder tener un esqueleto y una guía para su desarrollo.

Este software servirá en gran mayoría para el modelado del sistema como son los casos de usos, múltiples diagramas que se necesiten para el desarrollo, el diseño de la base de datos, etc. Gracias a su gran conjunción de técnicas de modelización y relación entre las mismas, esta herramienta será de gran ayuda para formar una arquitectura sólida del sistema y llevar un mejor análisis de ésta.

Análisis de la base de datos

Como se habló en el capítulo 2, para el desarrollo de esta disertación se planea utilizar las bases de datos relacionales, por lo que hay que tomar en cuenta algunas especificaciones previas a la creación de éstas. Lo que se necesitará será crear los modelos fundamentales que permitirán tener una guía clara y una estructura sobre la cual se desarrollará la parte de datos del sistema.

Para esto se tomarán en cuenta las siguientes consideraciones que constituirán los cimientos para crear la base de datos.

Modelo relacional: Una vez que se hayan identificado las entidades que serán partícipes en el sistema, se procederá a relacionarlas y diseñar el modelo relacional, el cual es una herramienta comprensible para las dos partes del cliente y el administrador de base de datos. Este modelo es de fácil comprensión gracias a la lógica de predicados, y hay que tomar en cuenta el grado de normalización que se debe implementar en la base de datos el cual será, en este caso de nivel 2 para prevenir posibles fallos en la base de datos y sus relaciones, no se planea implementar un nivel más alto de normalización ya que no es necesario por los datos que se estarán manejando en la base. Este modelo conformará los cimientos para construir la base de datos física del sistema.

Nomenclatura: Para la construcción y nombrado de la base de datos y sus componentes se utilizarán las siguientes especificaciones que permitirán en cualquier circunstancia la expansión, comprensión y formalidad.

Se han predispuesto las siguientes normas en cuanto a la nomenclatura de la base de datos:

- El nombre de cada tabla debe tener una clara relación en cuanto a los datos que va a abarcar.
- El nombre de cada tabla se debe escribir en su forma plural, esto quiere decir que todas las tablas terminarán con una letra “s”.
- Si se da el caso de que el nombre de la tabla tiene más de dos palabras, se las separará mediante el símbolo “_”.
- No se deben de usar ningún tipo de caracteres especiales, numéricos o distintos a los alfabéticos a excepción del “_”.
- No se utilizarán tildes ni la letra “ñ” para nombrar las tablas, en el caso de la letra se utilizará la sílaba “ni”.
- El nombre de las tablas debe estar escrito en español.
- En cuanto a los campos de cada tabla, se aplicarán las mismas reglas, agregando la que cada campo debe estar nombrado con letras minúsculas.
- En los campos se exceptuará la regla de nombrar con el plural.
- En los campos se podrá utilizar abreviaturas siempre y cuando sean claras y entendibles.

Capítulo 5: Diseño, implementación y pruebas

Este capítulo abarca el diseño y la construcción del sistema en sí, por lo que hay que tomar gran énfasis en el mismo. Se revisarán las fases que contempla la metodología utilizada para el desarrollo del software, así como también se realizarán las pruebas correspondientes al sistema para verificar su funcionalidad en cada uno de los módulos.

Se puede decir que esta es la cúspide de la disertación en sí, ya que en esta parte se diseñará el sistema tomando en cuenta los lineamientos previamente establecidos, así como también los requerimientos levantados en el capítulo anterior. Se pondrá en marcha el proceso de desarrollo aplicando el marco de trabajo que brinda la metodología, utilizando las técnicas y herramientas que dicta, para así lograr productos que denotarán la estructura y arquitectura del sistema. Estos ayudarán en el manejo de la base de datos, la codificación y la construcción de todo el sistema, dando como resultado el software final.

Por último, se realizarán las pruebas pertinentes por el cliente para corroborar cada una de las funcionalidades definidas, y para verificar que los requerimientos se hayan solventado de manera correcta. De esta manera el proceso que se ha realizado desde el inicio quedará finalizado, con el producto esperado y la satisfacción del cliente.

Consideraciones

En la metodología OOHDMM (), se presentan 4 fases principales para el desarrollo de un sistema web, las cuales están contempladas en los siguientes apartados. Pero antes, cabe recalcar algunas características que apoyarán al diseño del sistema, y por las cuales se utiliza dicha metodología.

OOHDMM, a diferencia de otras metodologías de desarrollo web, se enfoca en el paradigma orientado a objetos, por lo que se utilizarán los conocidos diagramas UML para la representación del sistema.

Esta metodología no contempla una fase de análisis de requerimientos, pero para poder tener un gran acercamiento de las necesidades del cliente, se realizó el respectivo levantamiento en el capítulo anterior.

Para comenzar el proceso de desarrollo, OOADM parte del diseño de un modelo conceptual, el cual ya es conocido por otras metodologías basadas en el paradigma orientado a objetos. A partir del modelo conceptual, se creará un nuevo modelo basándose en las clases del anterior, pero en un ámbito navegacional. Y de igual manera, a partir de éste, se creará un diseño de interfaces abstracto que representará la vista que tendrá el cliente del sistema y las interfaces.

Diseño Conceptual

En esta parte del desarrollo se construye el modelo conceptual en base a las técnicas utilizadas para la orientación de objetos. Esto definirá un dominio del sistema en un entorno semántico brindando un primer acercamiento a lo que será el sistema en sí en cuanto a su diseño.

El diagrama contiene los atributos y funciones que tendrá cada clase del sistema y las relaciones entre estas.

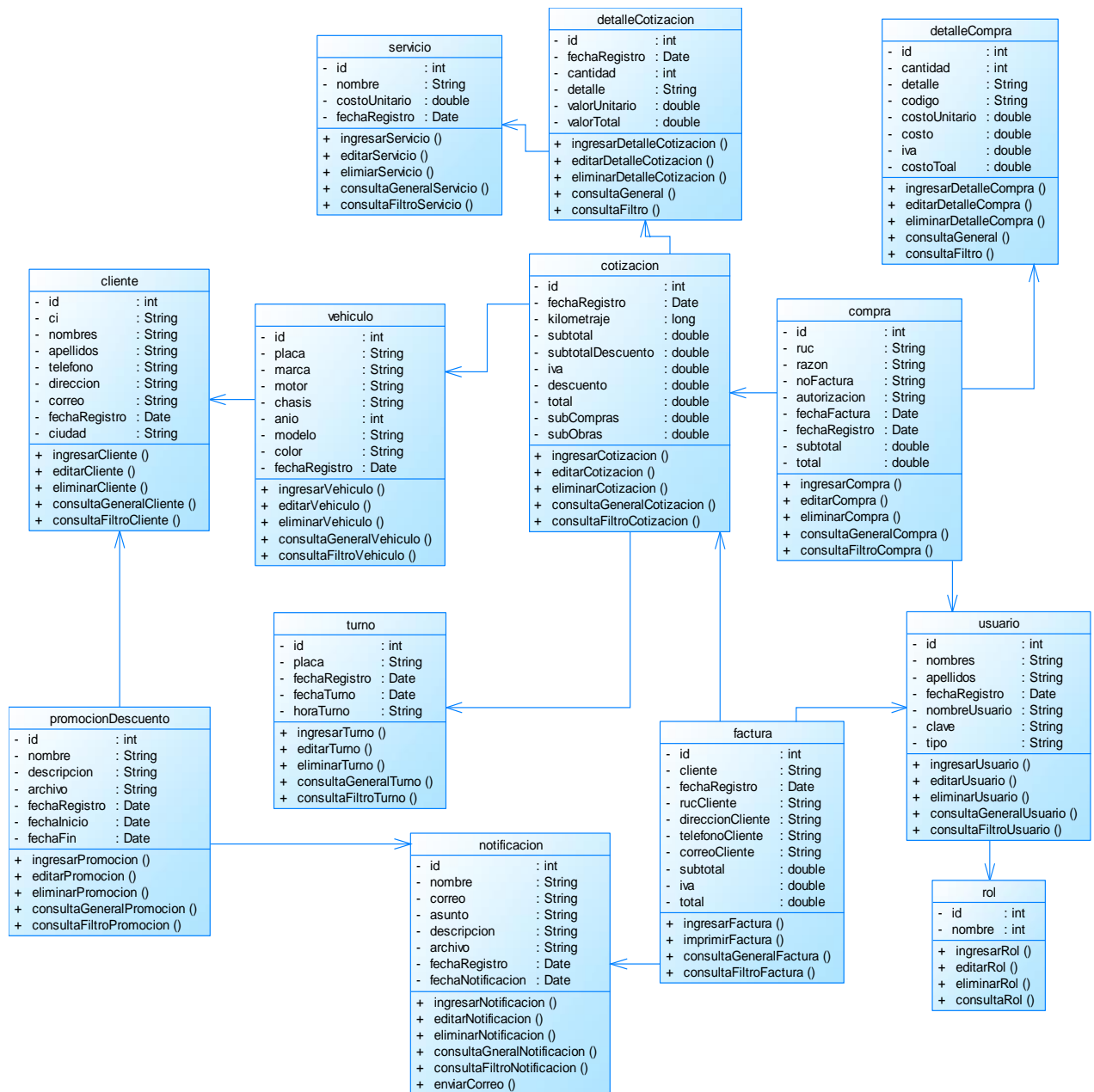


Figura 27 Diagrama de clases.

Fuente: Propia

En la Figura 27 se puede apreciar el modelo conceptual del sistema con todas sus clases, atributos, métodos y relaciones. Este es el primer paso para empezar la construcción del sistema en sí.

Diseño navegacional

En esta fase de desarrollo, se plantea crear un diseño navegacional de la aplicación basándose en las tareas que el usuario realizará juntamente con las clases definidas en el diseño conceptual.

La semántica que se utilizará para este esquema será la utilizada en las aplicaciones hipertexto, contemplando los nodos y enlaces de las vistas que tendrá la aplicación.

Para esta parte se dividirá el diseño en dos tipos de diagramas: el de clases navegacionales y el de contexto navegacional. Estos irán de la mano para una especificación del sistema.

Debido a la extensión del sistema y cada una de sus vistas y clases, se utilizará una simplificación de éstas para no extender el contenido.

Clases Navegacionales

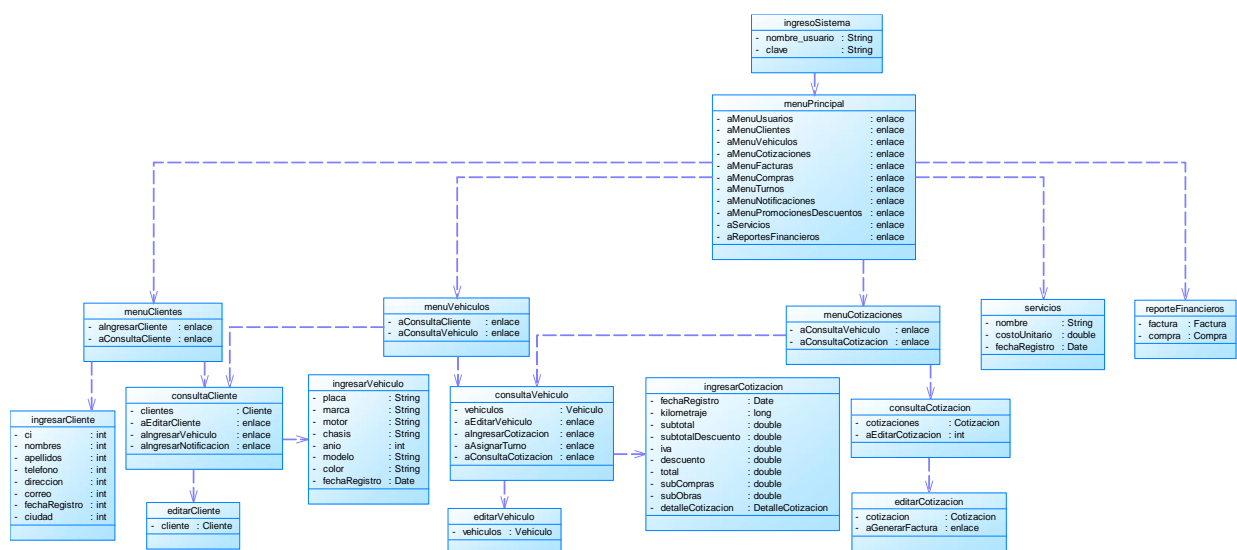


Figura 28 Diagrama de clases navegacional.

Fuente: Propia

En la Figura 28 se pueden apreciar las diferentes clases en su modelo navegacional. Se aprecian los enlaces que tendrá cada una con las demás para el flujo de información entre ellas.

Las clases definidas en el diagrama navegacional de clases contemplan las diferentes posibilidades de navegación, las demás clases faltantes predisponen una navegación similar a éstas, por lo que no se las incluyó para una mejor visualización.

Contexto Navegacional

La navegación no puede estar completamente definida sin un contexto en el cual se desenvuelva, es por esto, que el diagrama de contexto navegacional toma gran importancia en el diseño para poder tener una visualización del flujo que debe seguir el usuario para poder culminar una tarea o acción en el sistema.

De igual manera, para poder tener una mejor apreciación del contexto navegacional, se tomaron en cuenta las clases anteriores, con el fin de no extender el contenido.

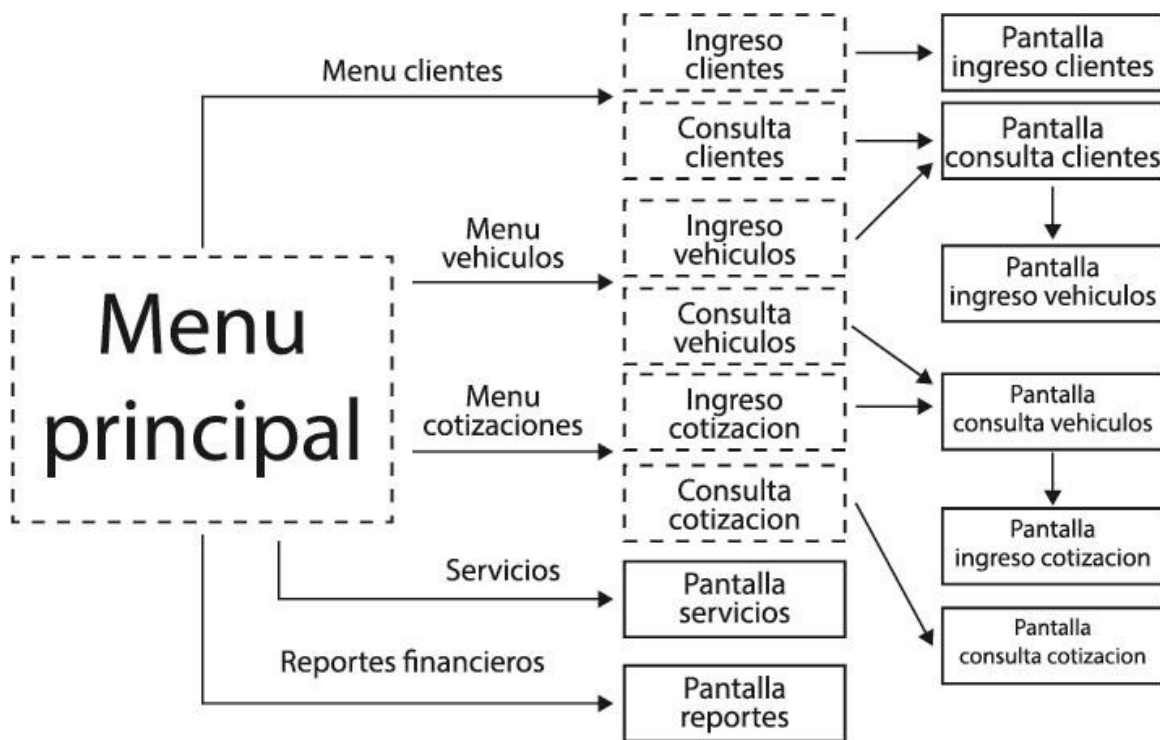


Figura 29 Diagrama de contexto navegacional.

Fuente: Propia

En la Figura 29 se puede apreciar el diagrama del Contexto Navegacional, en el que se representan las acciones que debe seguir el usuario por entre las vistas del sistema para poder llegar a lograr una determinada tarea.

Diseño de Interfaz Abstracta

Como se observó en los diagramas anteriores, se definió la estructura conceptual y navegacional del sistema, pero estas no son tan bien percibidas por el usuario. Es por lo que el diseño de la interfaz abstracta ayuda a identificar claramente las vistas que se tendrán del sistema.

Este diseño presenta un acercamiento claro de los objetos que poseerán las interfaces para que el usuario pueda verificar la solvencia de sus necesidades. Se muestran las posiciones de los elementos en pantalla que se acomodarán para la facilidad y el manejo del sistema.

Los diagramas utilizados se denominan ADV (Modelos de Vistas Abstractas), los cuales son representaciones formales utilizados para representar la manera en la que la interfaz se estructura, como se relaciona con las clases navegacionales y el cómo reacciona la aplicación a eventos externos.

Al igual que antes, se utilizará una simplificación de las vistas, que representará las interfaces de los demás restantes debido a que son muy similares.

Ingreso al sistema

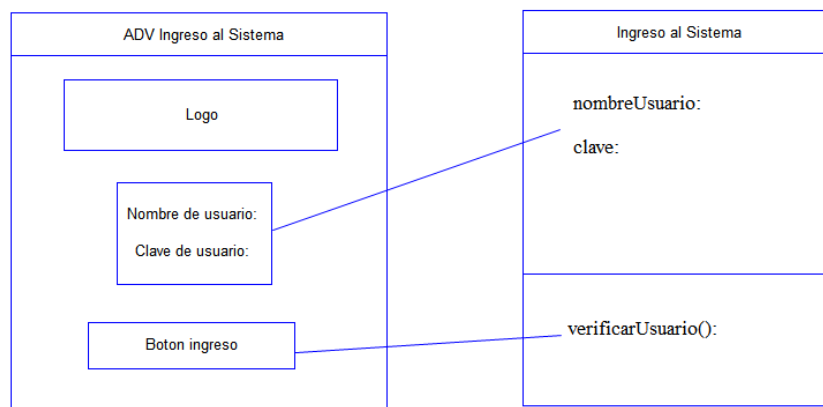


Figura 30 Diagrama de vista abstracta de Ingreso al Sistema.

Fuente: Propia

Pantalla principal

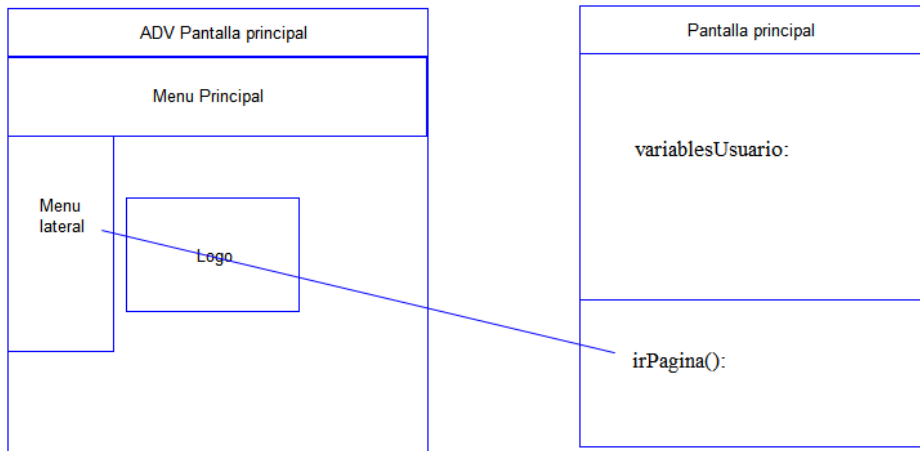


Figura 31 Diagrama de vista abstracta Pantalla Principal.

Fuente: Propia

Gestión de clientes

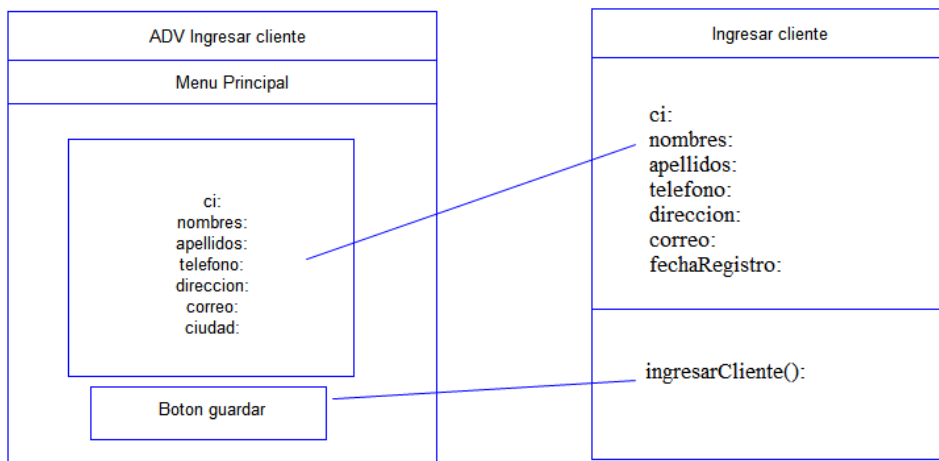


Figura 32 Diagrama de vista abstracta Ingreso Cliente.

Fuente: Propia

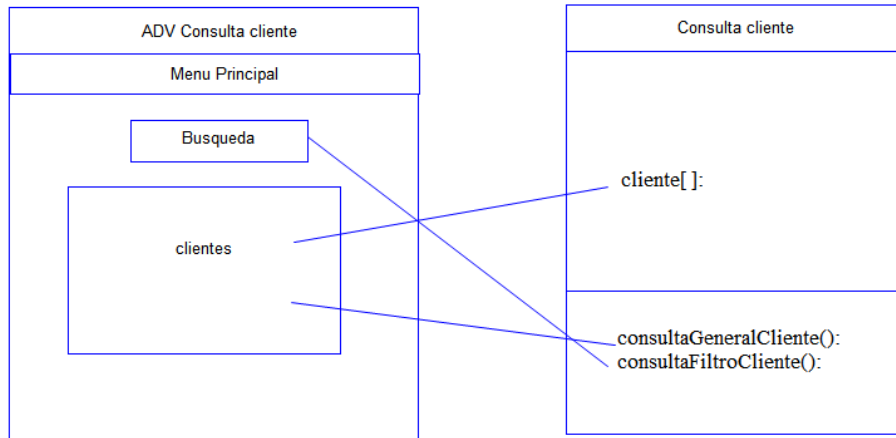


Figura 33 Diagrama de vista abstracta Consulta Cliente.

Fuente: Propia

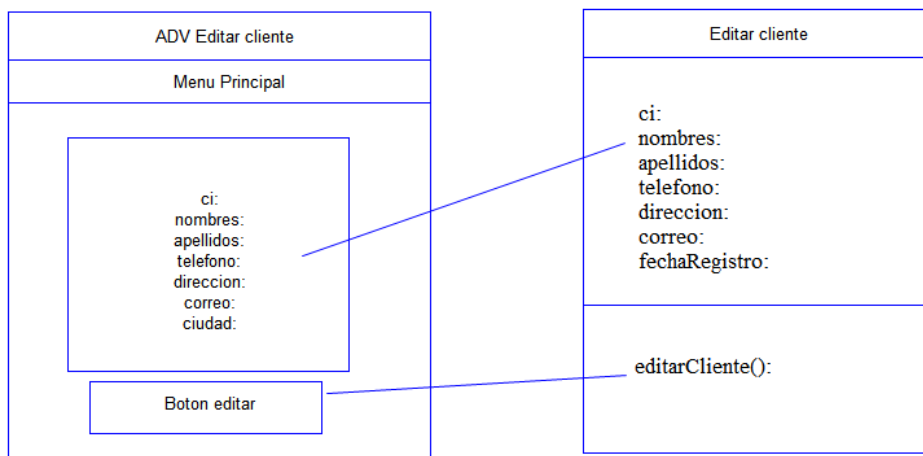


Figura 34 Diagrama de vista abstracta Editar Cliente.

Fuente: Propia

Diagrama de la Base de Datos

Como se explicó anteriormente, la base de datos que se utilizará para el sistema será del tipo relacional. Se tomarán en cuenta las relaciones que se necesiten entre cada tabla y las claves primarias y foráneas para mantener un modelo robusto que no permita fallos por parte del usuario en lo que refiere al ingreso de datos.

Se presentará el modelo conceptual de la base de datos y el modelo lógico de la misma para tener una mayor apreciación de la estructura de los datos.

Modelo conceptual de datos

Este modelo identifica las entidades que existen en la base de datos y las relaciones de más alto nivel, sin brindar un claro detalle de éstas.

En la Figura 35 se puede observar el diagrama del modelo conceptual de la base de datos para el sistema con los atributos de cada tabla y sus relaciones.

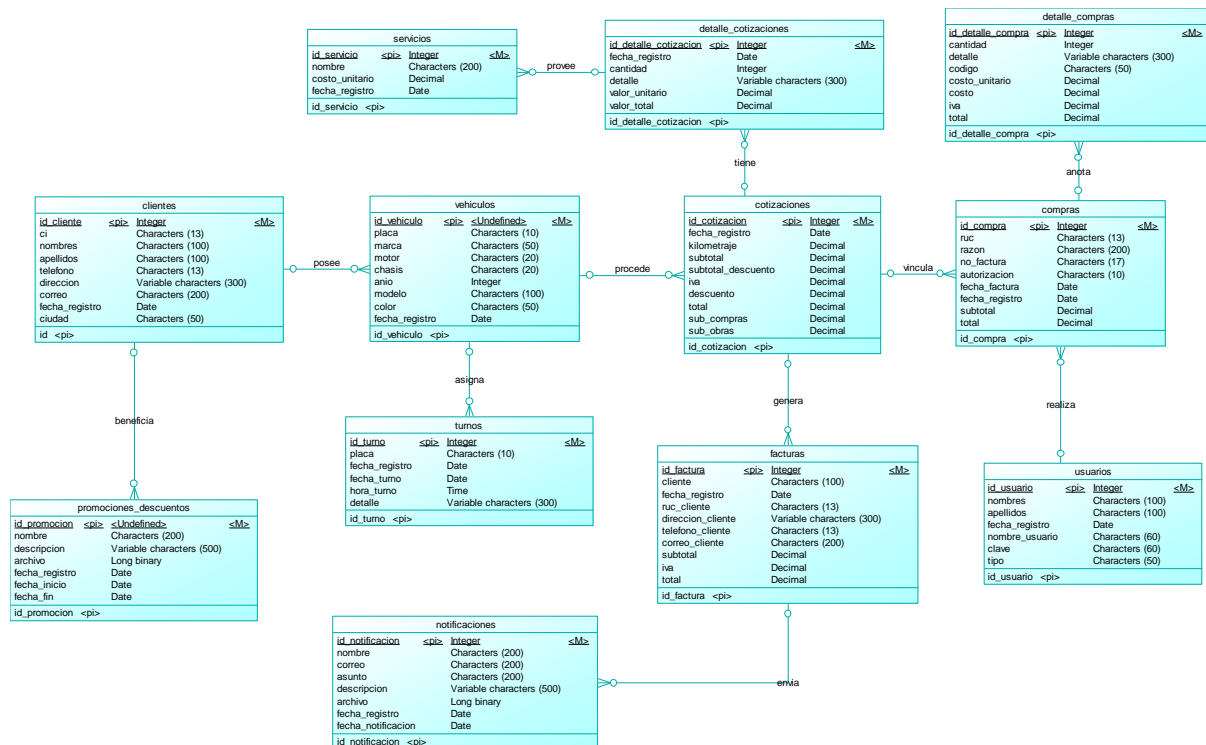


Figura 35 Diagrama conceptual de la base de datos.

Fuente: Propia

Modelo lógico de datos

Este modelo describe los datos a un nivel de detalle alto, independientemente de cómo se implementará en la base de datos real. Aquí se observan las claves foráneas y tablas intermedias que puedan existir entre las relaciones.

Se utiliza un nivel de normalización para crear un modelo robusto y fiable.

En la Figura 36 se observa el diagrama del modelo lógico de la base de datos para el sistema, tomando en cuenta sus atributos, relaciones y claves correspondientes.

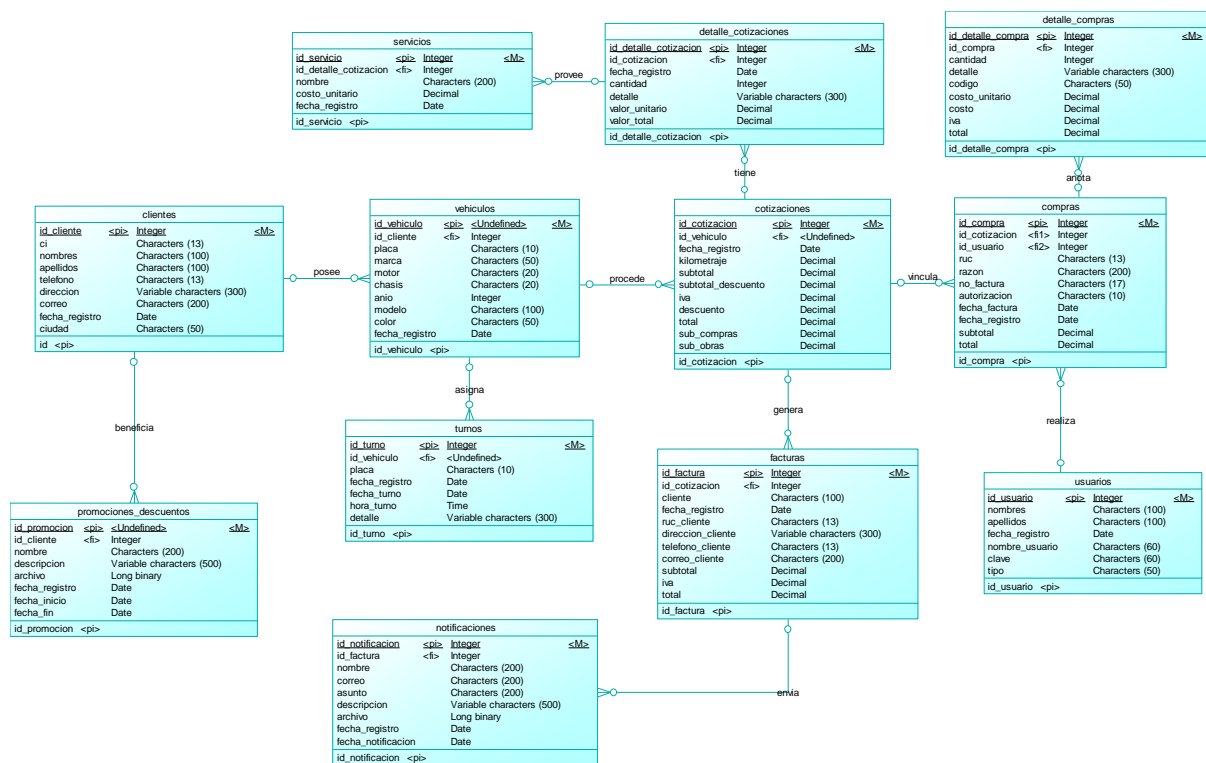


Figura 36 Diagrama lógico de la base de datos.

Fuente: Propia

A partir de estos modelos se construirá la base de datos del sistema previo a su codificación. Es por esta la importancia del modelado correcto tomando en cuenta los atributos correctos y sus relaciones, para evitar futuras modificaciones.

Implementación

Después de que se realizó el debido levantamiento de requerimientos del usuario, se escogieron las herramientas de desarrollo que se van a utilizar juntamente con los criterios, se diseñó el sistema en base a las fases y conceptos de la metodología escogida (conceptual navegacional y diseño abstracto) y se modeló la base de datos con sus entidades y relaciones. Se procede a codificar el sistema, contemplando los lineamientos, formalidades y especificaciones definidas con anterioridad.

En esta etapa del desarrollo el sistema es construido en los lenguajes seleccionados y con las técnicas apropiadas para el mismo. Esto se realizará parcialmente codificando los módulos de las clases básicas, seguido por los módulos de los procesos que consumirán información de dichas clases.

Una vez codificados todos los módulos, y después de haber verificado la funcionalidad correcta de los mismos, se procede a unirlos manteniendo los parámetros ya preestablecidos. Obteniendo como resultado el producto de esta disertación, un sistema para la gestión de talleres mecánicos.

En cuanto a las clases básicas del sistema, el comportamiento es muy similar, por lo que la codificación no difiere en sí. Es por esto que se mostrarán los detalles más específicos del código.

Conexión a la Base de Datos:

La conexión de base de datos que se realiza en Laravel consta en la creación de controladores y modelos. Estos modelos deben tener el nombre en singular década tabla de la base de datos, e irán de la mano con los controladores que contendrán las funciones de la lógica del negocio.

Los parámetros de la base de datos están definidos en un documento de tipo .env que permite la conexión a la base de datos.

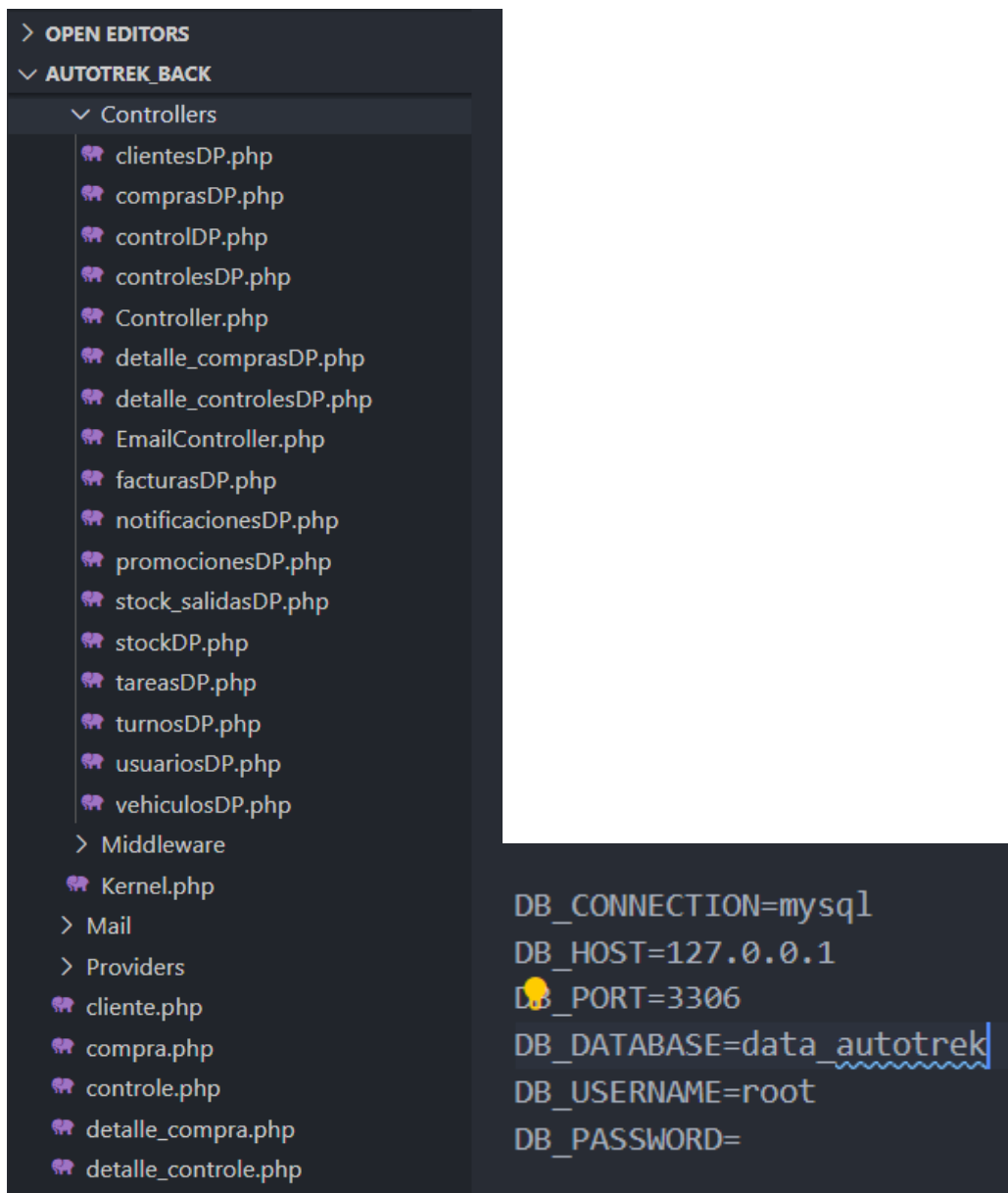


Figura 37 Modelos, controladores y conexión a la base de datos.

Fuente: Propia.

Conexión con el proveedor de E-mail

En cuanto a la conexión con el proveedor de E-mail, se debe realizar un proceso similar al de la base de datos, en el archivo .env ya definido. Y el código de implementación se muestra a continuación.

```
Mail::to($r->input('correo'))->send(new SendEmail($data,$subStrtokn[0]));
```

```
MAIL_MAILER=smt  
MAIL_HOST=smt.googlemail.com  
MAIL_PORT=465  
MAIL_USERNAME=[redacted]@gmail.com  
MAIL_PASSWORD=[redacted]  
MAIL_ENCRYPTION=ssl  
MAIL_FROM_ADDRESS=cuyrc68@gmail.com  
MAIL_FROM_NAME="{APP_NAME}"
```

Figura 38 Conexión con el proveedor de E-mail.

Fuente: Propia.

Rutas para consumir servicio

Para que el lado del cliente se conecte con el servidor, se necesitan de rutas específicas del tipo REST, las cuales se definieron después de crear cada una de las funciones de la lógica del negocio. A continuación, se presenta las rutas de vehículos para tener un claro acercamiento.

```
//vehiculos  
Route::get('/consultaGeneralVehiculo','vehiculosDP@consultaGeneralVehiculo');  
Route::get('/seleccionarVehiculoId/{id}','vehiculosDP@seleccionarVehiculoId');  
Route::get('/seleccionarVehiculoCliente/{id_cliente}','vehiculosDP@seleccionarVehiculoCliente');  
Route::get('/seleccionarVehiculoPlaca/{placa}','vehiculosDP@seleccionarVehiculoPlaca');  
Route::get('/seleccionarVehiculoMarca/{marca}','vehiculosDP@seleccionarVehiculoMarca');  
Route::get('/seleccionarVehiculoModelo/{modelo}','vehiculosDP@seleccionarVehiculoModelo');  
Route::get('/seleccionarVehiculoAnio/{anio}','vehiculosDP@seleccionarVehiculoAnio');  
Route::get('/seleccionarVehiculoDiaRegistro/{fecha_registro}','vehiculosDP@seleccionarVehiculoDiaRegistro');  
Route::get('/seleccionarVehiculoMesRegistro/{mes}','vehiculosDP@seleccionarVehiculoMesRegistro');  
Route::get('/seleccionarVehiculoAnioRegistro/{anio}','vehiculosDP@seleccionarVehiculoAnioRegistro');  
Route::post('/ingresarVehiculo','vehiculosDP@ingresarVehiculo');  
Route::put('/editarVehiculo/{id}','vehiculosDP@editarVehiculo');  
Route::delete('/eliminarVehiculo/{id}','vehiculosDP@eliminarVehiculo');
```

Figura 39 Rutas para el consumo de servicios.

Fuente: Propia.

Modelo y Controlador

Para la programación del modelo y el controlador, se utiliza la herramienta de Eloquent que proporciona Laravel, así se definen los modelos y se importan en los controladores para poder utilizarlos como objetos de datos.

```

use Illuminate\Database\Eloquent\Model;

class vehiculo extends Model
{
    public $timestamps = false;

    protected $fillable = ['id','id_cliente','placa','marca','motor','chasis','anio','modelo','color','fecha_registro'];
}

```

Figura 40 Modelo vehículo.

Fuente: Propia.

```

use Illuminate\Http\Request;
use App\vehiculo;

class vehiculosDP extends Controller
{
    public $mensaje='Comuníquese con el administrador';

    function consultaGeneralVehiculo (){
        try{
            $vehiculo = \App\vehiculo::all();

            return json_encode ($vehiculo);
        }
        catch (Exception $e) {
            report($e);
        }

        return $mensaje;
    }
}

```

Figura 41 Controlador vehículo.

Fuente: Propia.

Vista del Cliente

Se utilizará programación de JavaScript con el uso de las herramientas de React para la programación del Front-End. A continuación, se presenta un pedazo de código para demostrar la codificación.

```

class ingresarVehiculo extends Component{

  constructor(props){

    super(props);

    const {idEdit, nombreCliente, apellidoCliente}=this.props.location.state;
    this.state={
      mensaje:'',
      open: false,
      open2: false,
      open3: false,
      vehiculos: [],
      controles: [],
      clienteNombre: nombreCliente,
      clienteApellido: apellidoCliente,
      vehiculo1: '',
      id_cliente1: '',
      id_cliente: idEdit,
      placa: '',
      marca: '',
      motor: '',
      chasis: '',
      anio: '',
      modelo: '',
      color: '',
      fecha_registro: '',
      fecha_selec: '',
      fecha_mostrar1: '',
      seleccionado: false,
      env: new Entorno().getRuta(),
    }

    this.handleSubmit=this.handleSubmit.bind(this);
  }

```

```

  handleSubmit(){
    console.log(this.state);
    axios.post(this.state.env+'ingresarVehiculo', {
      id_cliente: this.state.id_cliente,
      placa: this.state.placa,
      marca: this.state.marca,
      motor: this.state.motor,
      chasis: this.state.chasis,
      anio: this.state.anio,
      modelo: this.state.modelo,
      color: this.state.color,
      fecha_registro: this.state.fecha_registro
    })
    .then(res=>{
      this.setState({mensaje: res.data});
      this.setRes();
    })
    .catch(function (error) {
      this.setState({mensaje: error});
    });
    this.handleOpen();
  }
}

```

```

render(){
  return(
    <div>
      <NavBar/>
      <div align="center">
        <br/>
        <br/>
        <div>
          <Paper >
            <Table >
              <TableHead>
                <TableRow>
                  <TableCell align='center' >PLACA</TableCell>
                  <TableCell align='center' >MARCA</TableCell>
                  <TableCell align='center' >MODELO</TableCell>
                  <TableCell align='center' >MOTOR</TableCell>
                  <TableCell align='center' >CHASIS</TableCell>
                  <TableCell align='center' >AÑO</TableCell>
                  <TableCell align='center' >COLOR</TableCell>
                  <TableCell align='center' >FECHA REGISTRO</TableCell>
                  <TableCell align='center' >ACCIÓN</TableCell>
                </TableRow>
              </TableHead>
              <TableBody id = 'cuerpoTabla'>
                {
                  Array.isArray(this.state.vehiculos) && this.state.vehiculos.map(vehiculo => {
                    return (
                      <TableRow>
                        <TableCell>
                          {vehiculo.placa}
                        </TableCell>
                        <TableCell>
                          {vehiculo.marca}

```

Figura 42 Código JavaScript con React.

Fuente: Propia.

Pruebas del cliente

Para esta etapa, se realizarán pruebas de funcionalidad del sistema por parte del cliente, para verificar el correcto funcionamiento de éste. Se tomarán en cuenta todos los lineamientos que se especificaron en el Documento de Requerimientos presentado anteriormente.

El usuario debe seguir el flujo que se presentará en el plan de pruebas, el cual servirá como camino para que se realice la verificación de una manera más armónica y sin dejar atrás cualquier funcionalidad.

Se presenta el plan de pruebas de cada módulo, seguido de las capturas de pantallas respectivas para corroborar el éxito o fracaso del módulo.

Caso de prueba: F1 Ingreso al Sistema.

Precondiciones: Tener al menos un usuario registrado en la base de datos.

Tabla 5 Prueba Ingreso al Sistema.

| Caso de uso | Entradas | Resultado esperado | Aceptación |
|--------------------|---|---|-------------------|
| F1 | Ingrese al sistema | Despliegue de la pantalla de Ingreso al Sistema. | |
| F1.1 | Ingresar los datos del usuario. -Válidos -No válidos | -Ingreso al sistema con la verificación de datos. -Mensaje de error. | |
| F1.2 | Seleccionar opción de Cerrar sesión en el menú principal de la barra de navegación. | La sesión se termina, se despliega la pantalla de Ingreso al Sistema. | |

Postcondiciones: Variables de sesión eliminadas.

Caso de prueba: F2 Gestión de Usuarios.

Precondiciones: Tabla de usuarios vacía.

Tabla 6 Prueba Gestión de Usuarios.

| Caso de uso | Entradas | Resultado esperado | Aceptación |
|--------------------|--|---|-------------------|
| F2 | Seleccione Usuarios en la barra de navegación y seleccione Ingresar Usuario. | Despliegue de la pantalla de Ingreso de Usuario. | |
| F2.1 | Ingresar los datos del usuario. -Válidos -No válidos | -Registro guardado en la BDD. -Mensaje de error. | |

| | | | |
|--------|---|--|--|
| F2.4.1 | Seleccione Usuarios en la barra de navegación y seleccione Consulta Usuarios. | Despliegue de la pantalla de Consulta de usuarios. | |
| F2.2 | Presione sobre el botón de editar del registro. | Despliegue de la pantalla de Editar Usuario con los datos precargados. | |
| F2.2 | Modifique los datos del registro y presione guardar. -Válidos -No válidos | -Registro modificado en la BDD. -Mensaje de error. | |
| F2.4 | Seleccione Usuarios en la barra de navegación y seleccione Consulta Usuarios. | Despliegue de la pantalla de Consulta de usuarios. | |
| F2.4.2 | Seleccione el filtro por el que desea buscar el registro e ingrese el dato correspondiente. | Despliegue del registro modificado anteriormente. | |
| F2.3 | Presione sobre el botón eliminar. | Mensaje de confirmación, registro eliminado de la BDD. | |

Postcondiciones: Variables de sesión eliminadas, tabla de usuarios vacía.

Caso de prueba: F3 Gestión de Clientes.

Precondiciones: Tabla de clientes vacía.

Tabla 7 Prueba Gestión de Clientes.

| Caso de uso | Entradas | Resultado esperado | Aceptación |
|--------------------|--|--|-------------------|
| F3 | Seleccione Clientes en la barra de navegación y seleccione Ingresar Cliente. | Despliegue de la pantalla de Ingreso de Cliente. | |

| | | | |
|--------|---|--|--|
| F3.1 | Ingresar los datos del cliente. -Válidos -No válidos | -Registro guardado en la BDD. -Mensaje de error. | |
| F3.4.1 | Seleccione Clientes en la barra de navegación y seleccione Consulta Clientes. | Despliegue de la pantalla de Consulta de clientes. | |
| F3.2 | Presione sobre el botón de editar del registro. | Despliegue de la pantalla de Editar Cliente con los datos precargados. | |
| F3.2 | Modifique los datos del registro y presione guardar. -Válidos -No válidos | -Registro modificado en la BDD. -Mensaje de error. | |
| F3.4 | Seleccione Clientes en la barra de navegación y seleccione Consulta Clientes. | Despliegue de la pantalla de Consulta de clientes. | |
| F3.4.2 | Seleccione el filtro por el que desea buscar el registro e ingrese el dato correspondiente. | Despliegue del registro modificado anteriormente. | |
| F3.3 | Presione sobre el botón eliminar. | -Mensaje de confirmación, registro eliminado de la BDD. | |

Postcondiciones: Tabla de clientes vacía.

Caso de prueba: F4 Gestión de Vehículos.

Precondiciones: Tabla de vehículos vacía, tener un cliente registrado.

Tabla 8 Prueba Gestión de Vehículos.

| Caso de uso | Entradas | Resultado esperado | Aceptación |
|--------------------|---|---|-------------------|
| F4 | Seleccione Vehículos en la barra de navegación y seleccione Ingresar Vehículo. | Despliegue de la pantalla de Consulta de Cliente. | |
| F4.1 | Presione sobre el botón de ingreso vehículo en el cliente propietario. | Despliegue de la pantalla de Ingreso de Vehículo | |
| F4.1 | Ingresar los datos del vehículo. -Válidos -No válidos | -Registro guardado en la BDD. -Mensaje de error. | |
| F4.4.1 | Seleccione Vehículos en la barra de navegación y seleccione Consulta Vehículos. | Despliegue de la pantalla de Consulta de vehículos. | |
| F4.2 | Presione sobre el botón de editar del registro. | Despliegue de la pantalla de Editar Vehículo con los datos precargados. | |
| F4.2 | Modifique los datos del registro y presione guardar. -Válidos -No válidos | -Registro modificado en la BDD. -Mensaje de error. | |
| F4.4 | Seleccione Vehículos en la barra de navegación y seleccione Consulta Vehículos. | Despliegue de la pantalla de Consulta de Vehículos. | |

| | | | |
|--------|---|---|--|
| F4.4.2 | Seleccione el filtro por el que desea buscar el registro e ingrese el dato correspondiente. | Despliegue del registro modificado anteriormente. | |
| F4.3 | Presione sobre el botón eliminar. | -Mensaje de confirmación, registro eliminado de la BDD. | |

Postcondiciones: Tabla de vehículos vacía.

Caso de prueba: F5 Gestión de Servicios.

Precondiciones: Tabla de servicios vacía.

Tabla 9 Prueba Gestión de Servicios.

| Caso de uso | Entradas | Resultado esperado | Aceptación |
|--------------------|---|---|-------------------|
| F5.4.1 | Seleccione Servicios en la barra de navegación. | Despliegue de la pantalla de Servicios con los campos de Ingreso y Consulta. | |
| F5.1 | Ingresar los datos del servicio en los campos designados. -Válidos -No válidos | -Registro guardado en la BDD, registro aparece en la tabla de consulta. -Mensaje de error. | |
| F5.2 | Presione sobre el botón de editar del registro. | Despliegue de los campos de editar servicio. | |
| F5.4.2 | Seleccione el filtro por el que desea buscar el registro e ingrese el dato correspondiente. | Despliegue del registro modificado anteriormente. | |
| F5.2 | Modifique los datos del registro y presione guardar. | | |

| | | | |
|------|-----------------------------------|---|--|
| | -Válidos -No válidos | -Registro modificado en la BDD. -Mensaje de error. | |
| F5.3 | Presione sobre el botón eliminar. | -Mensaje de confirmación, registro eliminado de la BDD. | |

Postcondiciones: Tabla de servicios vacía.

Caso de prueba: F6 Gestión de Cotizaciones.

Precondiciones: Tabla de cotizaciones vacía, tener un vehículo registrado.

Tabla 10 Prueba Gestión Cotizaciones.

| Caso de uso | Entradas | Resultado esperado | Aceptación |
|--------------------|---|---|-------------------|
| F6 | Seleccione Cotizaciones en la barra de navegación y seleccione Ingresar Cotización. | Despliegue de la pantalla de Consulta de Vehículos. | |
| F6.1 | Presione sobre el botón de ingreso cotización en el vehículo. | Despliegue de la pantalla de Ingreso de Cotización | |
| F6.1 | Ingresar los datos de la cotización. -Válidos -No válidos | -Registro guardado en la BDD. -Mensaje de error. | |
| F6.4.1 | Seleccione Cotizaciones en la barra de navegación y seleccione Consulta Cotizaciones. | Despliegue de la pantalla de Consulta de cotizaciones. | |

| | | | |
|--------|---|---|--|
| F6.2 | Presione sobre el botón de editar del registro. | Despliegue de la pantalla de Editar Cotización con los datos precargados. | |
| F6.2 | Modifique los datos del registro y presione guardar. -Válidos -No válidos | -Registro modificado en la BDD. -Mensaje de error. | |
| F6.4 | Seleccione Cotizaciones en la barra de navegación y seleccione Consulta Cotizaciones. | Despliegue de la pantalla de Consulta de Cotizaciones. | |
| F6.4.2 | Seleccione el filtro por el que desea buscar el registro e ingrese el dato correspondiente. | Despliegue del registro modificado anteriormente. | |
| F6.3 | Presione sobre el botón eliminar. | -Mensaje de confirmación, registro eliminado de la BDD. | |

Postcondiciones: Tabla de cotizaciones vacía.

Caso de prueba: F7 Gestión de Compras.

Precondiciones: Tabla de compras vacía, tener una cotización registrada.

Tabla 11 Prueba Gestión de Compras.

| Caso de uso | Entradas | Resultado esperado | Aceptación |
|--------------------|--|---|-------------------|
| F7 | Seleccione Compras en la barra de navegación y seleccione Ingresar Compra. | Despliegue de la pantalla de Ingreso de Compra. | |
| F7.1 | Ingresar los datos de la compra. | | |

| | | | |
|--------|---|---|--|
| | -Válidos -No válidos | -Registro guardado en la BDD. -Mensaje de error. | |
| F7.4.1 | Seleccione Compras en la barra de navegación y seleccione Consulta Compras. | Despliegue de la pantalla de Consulta de clientes. | |
| F7.2 | Presione sobre el botón de editar del registro. | Despliegue de la pantalla de Editar Compra con los datos precargados. | |
| F7.2 | Modifique los datos del registro y presione guardar. -Válidos -No válidos | -Registro modificado en la BDD. -Mensaje de error. | |
| F7.4 | Seleccione Compras en la barra de navegación y seleccione Consulta Compras. | Despliegue de la pantalla de Consulta de compras. | |
| F7.4.2 | Seleccione el filtro por el que desea buscar el registro e ingrese el dato correspondiente. | Despliegue del registro modificado anteriormente. | |
| F7.3 | Presione sobre el botón eliminar. | -Mensaje de confirmación, registro eliminado de la BDD. | |

Postcondiciones: Tabla de compras vacía.

Caso de prueba: F8 Gestión de Facturas.

Precondiciones: Tabla de facturas vacía, tener una cotización registrada.

Tabla 12 Prueba Gestión de Facturas.

| Caso de uso | Entradas | Resultado esperado | Aceptación |
|-------------|----------|--------------------|------------|
|-------------|----------|--------------------|------------|

| | | | |
|--------|---|---|--|
| F8 | Seleccione Facturas en la barra de navegación y seleccione Generar Factura. | Despliegue de la pantalla de Consulta de Cotizaciones. | |
| F8.1 | Presione sobre el botón de editar en la cotización. | Despliegue de la pantalla de Editar Cotización | |
| F8.1 | Presione el botón Factura | Despliegue de la pantalla de Factura | |
| F8.1 | Presione sobre el botón guardar. -Válidos -No válidos | -Registro guardado en la BDD. -Mensaje de error. | |
| F8.2 | Presione sobre el botón imprimir. | Despliegue de vista de impresión. | |
| F8.4.1 | Seleccione Cotizaciones en la barra de navegación y seleccione Consulta Cotizaciones. | Despliegue de la pantalla de Consulta de cotizaciones. | |
| F8.2 | Presione sobre el botón de editar del registro. | Despliegue de la pantalla de Editar Cotización con los datos precargados. | |
| F8.2 | Modifique los datos del registro y presione guardar. -Válidos -No válidos | -Registro modificado en la BDD. -Mensaje de error. | |
| F8.4 | Seleccione Cotizaciones en la barra de navegación y seleccione Consulta Cotizaciones. | Despliegue de la pantalla de Consulta de Cotizaciones. | |
| F8.4.2 | Seleccione el filtro por el que desea buscar el registro e | Despliegue del registro modificado anteriormente. | |

| | | | |
|------|-----------------------------------|---|--|
| | ingrese el dato correspondiente. | | |
| F8.3 | Presione sobre el botón eliminar. | -Mensaje de confirmación, registro eliminado de la BDD. | |

Postcondiciones: Tabla de facturas con un registro, impresión de la factura.

Caso de prueba: F9 Gestión de Turnos.

Precondiciones: Tabla de turnos vacía, tener un vehículo registrado.

Tabla 13 Prueba Gestión de Turnos.

| Caso de uso | Entradas | Resultado esperado | Aceptación |
|--------------------|---|--|-------------------|
| F9 | Seleccione Turnos en la barra de navegación y seleccione Asignar Turno. | Despliegue de la pantalla de Consulta de Vehículos. | |
| F9.1 | Presione sobre el botón de turno en el registro del vehículo. | Despliegue de la ventana de Selección de fecha de turno. | |
| F9.1 | Seleccione la fecha del turno para el vehículo. | Despliegue de la pantalla de Ingreso de Turno. | |
| F9.1 | Ingresar los datos del turno. -Válidos -No válidos | -Registro guardado en la BDD. -Mensaje de error. | |
| F9.4.1 | Seleccione Turnos en la barra de navegación y seleccione Consulta Turnos. | Despliegue de la pantalla de Consulta de Turnos. | |

| | | | |
|--------|---|--|--|
| F9.2 | Presione sobre el botón de editar del registro. | Despliegue de la pantalla de Editar Turno con los datos precargados. | |
| F9.2 | Modifique los datos del registro y presione guardar. -Válidos -No válidos | -Registro modificado en la BDD. -Mensaje de error. | |
| F9.4 | Seleccione Turnos en la barra de navegación y seleccione Consulta Turnos. | Despliegue de la pantalla de Consulta de Turnos. | |
| F9.4.2 | Seleccione el filtro por el que desea buscar el registro e ingrese el dato correspondiente. | Despliegue del registro modificado anteriormente. | |
| F9.3 | Presione sobre el botón eliminar. | -Mensaje de confirmación, registro eliminado de la BDD. | |

Postcondiciones: Tabla de turnos vacía.

Caso de prueba: F10 Gestión de Notificaciones.

Precondiciones: Tabla de notificaciones vacía, tener un cliente registrado.

Tabla 14 Prueba Gestión de Notificaciones.

| Caso de uso | Entradas | Resultado esperado | Aceptación |
|--------------------|---|--|-------------------|
| F10 | Ingrese a la factura del cliente que desea notificar, o al cliente. | Despliegue de la ventana de notificaciones. | |
| F10.1 | Presione sobre el botón de turno en el registro del vehículo. | Despliegue de la ventana de Ingreso de Notificación. | |

| | | | |
|---------|---|--|--|
| F10.1 | Ingresar los datos de la notificación. -Válidos -No válidos | -Registro guardado en la BDD. -Mensaje de error. | |
| F10.4.1 | Seleccione Notificaciones en la barra de navegación y seleccione Consulta Notificaciones. | Despliegue de la pantalla de Consulta de Notificaciones. | |
| F10.2 | Presione sobre el botón de editar del registro. | Despliegue de la ventana de Editar Notificación con los datos precargados. | |
| F10.2 | Modifique los datos del registro y presione guardar. -Válidos -No válidos | -Registro modificado en la BDD. -Mensaje de error. | |
| F10.4 | Seleccione Notificaciones en la barra de navegación y seleccione Consulta Turnos. | Despliegue de la pantalla de Consulta de Notificaciones. | |
| F10.4.2 | Seleccione el filtro por el que desea buscar el registro e ingrese el dato correspondiente. | Despliegue del registro modificado anteriormente. | |
| F10.3 | Presione sobre el botón eliminar. | -Mensaje de confirmación, registro eliminado de la BDD. | |
| F10.5 | Seleccione Notificaciones en la barra de navegación y seleccione Enviar Correo. | Despliegue de la pantalla de correo. | |
| F10.5 | Ingresar los datos del correo. -Válidos -No válidos | -Envío de correo exitoso. -Mensaje de error. | |

Postcondiciones: Tabla de notificaciones vacía.

Caso de prueba: F11 Gestión de Promociones y Descuentos.

Precondiciones: Tabla de promociones vacía.

Tabla 15 Prueba Gestión de Promociones y Descuentos.

| Caso de uso | Entradas | Resultado esperado | Aceptación |
|--------------------|--|--|-------------------|
| F11 | Seleccione Promociones y Descuentos en la barra de navegación y seleccione Ingresar Promoción. | Despliegue de la pantalla de Ingreso de Promoción. | |
| F11.1 | Ingresar los datos de la promoción. -Válidos -No válidos | -Registro guardado en la BDD. -Mensaje de error. | |
| F11.4.1 | Seleccione Promociones y Descuentos en la barra de navegación y seleccione Consulta Promociones. | Despliegue de la pantalla de Consulta de promociones. | |
| F11.2 | Presione sobre el botón de editar del registro. | Despliegue de la pantalla de Editar Promoción con los datos precargados. | |
| F11.2 | Modifique los datos del registro y presione guardar. -Válidos -No válidos | -Registro modificado en la BDD. -Mensaje de error. | |
| F11.4 | Seleccione Promociones y Descuentos en la barra de | Despliegue de la pantalla de Consulta de Promociones. | |

| | | | |
|---------|---|---|--|
| | navegación y seleccione Consulta Promoción. | | |
| F11.4.2 | Seleccione el filtro por el que desea buscar el registro e ingrese el dato correspondiente. | Despliegue del registro modificado anteriormente. | |
| F11.3 | Presione sobre el botón eliminar. | -Mensaje de confirmación, registro eliminado de la BDD. | |

Postcondiciones: Tabla de promociones vacía.

Caso de prueba: F12 Gestión de Reportes Financieros.

Precondiciones: Tener registros de factura y compra.

Tabla 16 Prueba Gestión de Reportes Financieros.

| Caso de uso | Entradas | Resultado esperado | Aceptación |
|--------------------|--|---|-------------------|
| F12 | Seleccione Reportes. | Despliegue de la pantalla de Ingreso de Reportes. | |
| F12 | Seleccione el criterio de búsqueda de período de tiempo. | Despliegue de los datos encontrados en ese período. | |

Postcondiciones: Ninguna

Verificación:

VERIFICACIÓN DE FUNCIONALIDAD DEL SISTEMA

Este documento, tiene como objetivo la verificación y validación de las pruebas funcionales del sistema para corroborar que se hayan solventado todas las necesidades que el Ing. Fernando Vinueza Maldonado, gerente general del Taller Automotriz AUTOTREK requirió para el sistema de gestión de éste.

A continuación, se presentan las funcionalidades del sistema para la correspondiente verificación y validación. Cabe recalcar que, si la funcionalidad no cumple en cualquier aspecto con lo solicitado, se debe marcar la casilla de no cumple.

| No. | Funcionalidad | Si cumple | No cumple |
|-----|-------------------------------------|-------------------------------------|--------------------------|
| F1 | Ingreso al Sistema | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| F2 | Gestión de Usuarios | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| F3 | Gestión de Clientes | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| F4 | Gestión de Vehiculos | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| F5 | Gestión de Servicios | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| F6 | Gestión de Cotizaciones | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| F7 | Gestión de compras | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| F8 | Gestión de facturas | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| F9 | Gestión de Turnos | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| F10 | Gestión de Notificaciones | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| F11 | Gestión de Promociones y Descuentos | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| F12 | Gestión de Reportes Financieros | <input checked="" type="checkbox"/> | <input type="checkbox"/> |

A continuación, se presentan otros puntos claves para verificación en el sistema.

| Consideración | Si cumple | No cumple |
|---|-------------------------------------|--------------------------|
| La interfaz de usuario es la correcta, y tiene un diseño cómodo y de fácil uso. | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| Se aplicaron los colores y componentes solicitados. | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| El sistema cumple con los estándares definidos. | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| El sistema cumple con las funcionalidades y utilidades predichas. | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| El usuario está conforme con el sistema en su totalidad. | <input checked="" type="checkbox"/> | <input type="checkbox"/> |

Observaciones

.....
.....
.....



Ing. Fernando Vinueza



Andrés Pinto

Capítulo 6: Conclusiones y Recomendaciones

Conclusiones

- El levantamiento de requerimientos fue fundamental para la comprensión de las necesidades del cliente y para el inicio del proceso de desarrollo del sistema. Así se tuvo una grata comprensión de los dos lados: cliente y desarrollador. Esto gracias a la implementación de las normativas que dicta la Especificación de Requerimientos de la IEEE.
- La base de datos cumplió con las necesidades del sistema, ya que se la diseñó con los lineamientos y normalización requeridos. Cada una de las entidades fue abstraída conforme al negocio y su funcionalidad. De esta manera se garantizó seguridad en la información y persistencia.
- Cada una de las herramientas seleccionadas cumplió de manera efectiva con su cometido en cada una de las fases del proceso de desarrollo. Estas herramientas brindaron un gran apoyo y facilidad en cuanto al diseño y codificación del software. Por lo que la elección fue primordial para la culminación exitosa del software.
- Se realizó con éxito el desarrollo del sistema, tomando en cuenta todos los lineamientos previstos en la investigación y en la etapa de levantamiento de requerimientos. De esta forma cada una de las necesidades especificadas por el cliente fueron solventadas basándose en el esquema que brindó la metodología seleccionada juntamente con las herramientas seleccionadas para el desarrollo de la disertación.
- Gracias a la implementación del sistema, se mejoraron los procesos del taller y se brindó una mejoría en cuanto a la gestión, organización y seguridad, no solo de la información que se producía en el día a día, sino también en el negocio como tal. Así es como las tecnologías de la información cumplen con su cometido en la automatización y optimización de un negocio.

- El desarrollo de una aplicación web permitió la utilización del sistema en un entorno amigable para el usuario, ya que, gracias a la conexión por la red, no se necesitó instalar las dependencias que utiliza el sistema en cada computadora, sino solo en el equipo que sirve como servidor. De esta manera la portabilidad fue de gran ayuda para el manejo de software.
- El proceso de desarrollo de software fue esencial para la construcción del sistema, ya que gracias a todos los puntos que fueron tomados en cuenta, el sistema tuvo éxito en su funcionalidad y en su diseño previo a la implementación de éste. La secuencialidad realizada no solo ayudó, sino también permitió que cada uno de los módulos funcionaran en armonía con sus adyacentes. De esta manera no se tuvo mayores inconvenientes y errores en la codificación y ejecución de cada una de las partes.
- Las pruebas reflejaron la correcta funcionalidad del sistema en conjunto y de cada uno de los módulos que se desarrollaron. Se pudo observar la ausencia de errores funcionales y el óptimo comportamiento en cada una de las vistas presentadas al cliente.
- La metodología que se escogió para la construcción del sistema fue un marco muy acoplado en cuanto a las necesidades de software, ya que, cada una de sus etapas sirvió como base para las consecuentes y para el éxito de la disertación. Gracias al paradigma de la orientación a objetos y el desarrollo web, esta metodología apoyó acertadamente en la arquitectura y diseño que el sistema presenta.

Recomendaciones

- Al momento de la implementación del sistema en cuanto al código, se recomienda mantener una organización en carpetas dependiendo de cada uno de los módulos. De esta manera se mantendrá una mejor visión y orden en cada una de las vistas del front-end y de las partes del back-end.
- Cuando se inicie el back-end, hay que tener en cuenta tomar la dirección IP del computador que servirá como servidor y reflejarlo en las variables de entorno del front-end para que no solo sirva localmente en el equipo. Así cada una de las computadoras conectadas a la red tendrán acceso al sistema.

- Ya que este sistema fue diseñado para talleres pequeños que aún no poseen varias sucursales, es ejecutado en un servidor propio del negocio. Pero gracias a las necesidades suscritas y satisfechas, puede funcionar efectivamente en negocios más grandes con varias sucursales. Pero para que esto sea posible, se debe subir a un servidor en la web para que se pueda tener acceso a éste.
- En cuanto a la gestión de notificaciones y promociones, ya que el sistema utiliza un proveedor de correos electrónicos de Gmail, se debe crear una excepción para que el antivirus del equipo no afecte el funcionamiento de éste. No se recomienda desactivarlo permanentemente por la inseguridad que esto representa.
- Mantener un código limpio y bien organizado es una buena práctica en cuando al desarrollo de software, se recomienda estrictamente utilizar los lineamientos de codificación, nombramiento de variables, organización de archivos, encapsulación, etc. para que el sistema pueda ser actualizado e inclusive acoplado a otros sistemas. De esta manera la mantenibilidad y evolución pueden ser posibles.

Bibliografía

- Agut, R. M. (2001). *Especificación de Requisitos Software según el estándar de IEEE 830*. Castellón de la Plana: Universitat Jaume I.
- Alba, P. R. (2011). *Manual de Javascript*. Madrid: CEP, S.L.
- Anciano, J. T. (2010). *Manual de Introducción al lenguaje HTML*. Madrid: CEP.
- Antonio Navarro Martín, A. F.-V. (2006). *Conceptualización, prototipado y proceso de aplicaciones hipermedia*. Madrid: Universidad Complutense de Madrid.
- Basterretche, J. F. (2007). *Dispositivos Móviles*. Argentina.
- Bean, M. (2015). *Laravel 5 Essentials*. Birmingham: PACKT.

- Cabello, J. C. (2014). *Diseño de páginas Web con XHTML, JavaScript y CSS*. Madrid: RA-MA.
- Carrascal, J. A. (s.f.).
- Castañeda, M. Á. (2016). *Desarrollo de aplicaciones web multiplataforma*. Ministerio de Educación de España.
- Daniel Schwabe, G. R. (s.f.). *The Object-Oriented Hypermedia Design Model (OOHDM)*. Rio de Janeiro: Pontificia Universidade Católica.
- Enriquez Juan Gabriel, C. S. (2013). *USABILIDAD EN APLICACIONES MÓVILES*. UNPA.
- Evans, C. (2010). *Fundamentos de PHP*. México: McGraw-Hill Interamericana.
- Francisco Domínguez Mateos, M. P. (2014). *Programación multimedia y dispositivos móviles*. Madrid: RA-MA Editorial.
- García, J. A. (2016). *HTML5, CSS3 y JQuery: curso práctico*. Madrid: RA-MA.
- Gasca Mantilla, M. C., Camargo Ariza, L. L., & Delgado, M. (2014). Metodología para el desarrollo de aplicaciones móviles. *Tecnura*, 20-35.
- Germán Terrazas, G. R. (s.f.). *Aplicando Estrategias de Mapeo OOHDM–WCML para el desarrollo de Aplicaciones Web*. La Plata: Universidad Nacional de La Plata.
- Jiménez, J. Z. (2013). *Aplicaciones web*. Macmillan Iberia, S.A.
- Jorge Mohedano, J. M. (2012). *Iniciación a javascript*. Ministerio de Educación de España.
- Juan Miguel Aguado, C. F. (2013). *La comunicación móvil*. Barceloca: Editorial Gedisa.
- Kolbeck, R. (1997). *El gran libro de JavaScript*. Barcelona: marcombo.
- laravel. (s.f.). *Laravel*. Obtenido de <https://laravel.com/>
- Lepre, E. (s.f.). *Esquema de una Red LAN*.
- Lisandro Delía, N. G. (s.f.). *Un Análisis Experimental de Tipo de Aplicaciones para Dispositivos Móviles*. Obtenido de https://digital.cic.gba.gob.ar/bitstream/handle/11746/2091/11746_2091.pdf-PDFA.pdf?sequence=1&isAllowed=y
- Luke Welling, L. T. (2005). *Desarrollo Web con PHP*. Madrid: ANAYA.
- Luna, A. C. (2014). *Creación de páginas web: HTML 5*. Málaga: ICB.
- Marini, E. (Octubre de 2012). *Linuxito*. Obtenido de <https://www.linuxito.com/docs/el-modelo-cliente-servidor.pdf>
- Martínez, J. F. (2014). *Aplicaciones web*. Madrid: RA-MA Editorial.
- Mateu, C. (2004). *Desarrollo de aplicaciones web*. Barcelona: Eureka Media, SL.
- McGraw, H. (22 de 10 de 2008). *mailxmail*. Obtenido de <http://www.mailxmail.com/curso-red-informatica/red-area-local-conexiones-externas-red>

- Millán, J. A. (21 de 01 de 2006). *Universitat de Lleida*. Obtenido de <http://cv.udl.cat/cursos/elsmitjans/t1/docs/internet2.pdf>
- Mobile Marketing Association (MMA). (2011). *Libro Blanco de Apps*. Madrid: Mobile Marketing Association (MMA).
- Mora, S. L. (2002). *Programación de Aplicaciones Web: Historia. Principios Básicos y Clientes Web*. San Vicente: Club Universitario.
- Moro, C. T. (2009). *openaccess*. Obtenido de http://openaccess.uoc.edu/webapps/o2/bitstream/10609/9164/1/dispositivos_moviles_y_multimedia.pdf
- Overflow, S. (s.f.). *Aprendizaje Ionic-framework*. Obtenido de <https://riptutorial.com/Download/ionic-framework-es.pdf>
- Quijado, J. L. (2011). *Domina JavaScript*. México D.F.: Ra-Ma.
- Quijado, J. L. (2014). *Domine PHP y MySQL (2a. ed.)*. Madrid: RA-MA.
- Rafael León Sanz, R. G. (2014). *Introducción a la movilidad: 4G/LTE y el desarrollo de aplicaciones Android*. Dextra Editorial.
- Raül V. Lerma-Blasco, J. A. (2013). *Aplicaciones web*. Madrid: McGraw-Hill España.
- reactjs. (s.f.). *reactjs*. Obtenido de <https://es.reactjs.org/docs/getting-started.html>
- Rodríguez, M. D. (2012). *JavaScript (2a. ed.)*. Málaga: ICB.
- Sanz, P. V. (2014). *Gestión de bases de datos*. Madrid: RA-MA.
- Suehring, S. (2002). *MySQL Bible*. New York: Wiley Publishing.
- Vaswani, V. (2009). *Fundamentos de PHP*. Mexico: McGrawHill.
- Vinueza, M. D. (2013). *Aplicacion de la metodología OOADM y técnicas de la inteligencia artificial en la solución del desarrollo de un Videojuego enfocado a niños de 6 a 10 años , utilizando la tecnología GDI+ Basado en C# y Wimote ára su aplicación en la Empresa Virtual Learni*. Sangolquí: Pontificia Universidade Católica.
- Vique, R. R. (s.f.). *Métodos para el desarrollo de aplicaciones móviles*. Madrid: FUOC.
- web, P. y. (2005). *Ángel Cobo, Patricia Gómez, Daniel Pérez, Rocío Rocha*. Madrid: Díaz Santos.
- YU, H. R. (2014). ICCSET. *Design and implementation of web based on Laravel framework* (págs. 301-304). Binhai: ICCSET.