



UNIVERSIDAD  
Privada  
DR. RAFAEL BELLOSO CHACÍN

# Capítulo II

---

## MARCO TEÓRICO

## **CAPÍTULO II**

### **MARCO TEÓRICO**

En este capítulo, se hace referencia a las investigaciones previas que sustenta la investigación actual y tienen un aporte sustancioso. Asimismo, se plasman los conceptos y las definiciones fundamentales de las tecnologías y el entorno concerniente a este trabajo.

#### **1. ANTECEDENTES DE LA INVESTIGACIÓN**

Como referencia para realizar este trabajo se tomaron una serie de investigaciones que servirán de pilares fundamentales para llegar a las conclusiones pertinentes gracias a sus aportes teóricos y metodológicos; sirviendo así estos trabajos como antecedentes del estudio entre los que se encuentran:

En primera instancia, se tiene el trabajo especial de grado para optar al título de Ingeniero en Informática en la Universidad Privada Dr. Rafael Bellosillo Chacín realizado por Bastidas, Faria y Zárraga (2016) llamado “Aplicación bajo ambiente web para la gestión de proyectos informáticos en una empresa de

desarrollo de software” con el objeto de optimizar la comunicación entre empresa-cliente y el monitoreo en la gestión de proyectos informáticos. Las bases teóricas fueron soportadas por Lujan (2012), Llorens (2005) y Rodríguez (2007).

El tipo de investigación fue proyectiva, descriptiva, de campo y documental con un diseño transaccional. Se aplicaron diferentes instrumentos de recolección de datos como la entrevista, encuesta, observación directa y revisión documental, así como también la guía de visita, lista de cotejos, guía de entrevista y cuestionarios.

La metodología seleccionada fue ecléctica de Senn (2003), Sommerville (2002) y Powell (2001), y a su vez estuvo conformada por seis fases: investigación preliminar, definición de requerimientos del sistema, diseño del sistema, disección de la fase del sistema, disección de la fase de diseño, desarrollo de los subsistemas y pruebas; para el desarrollo de la misma, las herramientas utilizadas fueron los lenguajes de programación PHP, JavaScript, como lenguaje de etiquetado HTML5 y CSS3 y como gestor de base de datos MySQL; luego del análisis de los resultados obtenidos de las diferentes pruebas elaboradas se obtuvo una aplicación bajo ambiente web adaptada a las necesidades que se presentan durante la gestión de proyectos informáticos en las empresas de desarrollo de software, así optimizando la comunicación empresa-cliente y el monitoreo de los distintos proyectos llevados a cabo.

Esta investigación **Aporta** información valiosa referente al desarrollo de

aplicaciones web, las herramientas y tecnologías utilizadas en el desarrollo de software, así como también la metodología utilizada para llevar acabo el mismo, así como también importante contenido referente a bases teóricas.

Por otra parte, se tiene el trabajo especial de grado para optar al título de Ingeniero en Informática en la Universidad Privada Dr. Rafael Beloso Chacín realizado por Bellorín, Castillo y González denominado “Aplicación informática bajo ambiente web para la automatización de los procesos operativos de la Marmolería Bellorín C.A.” con la finalidad de agilizar los procesos administrativos que se realizaban manualmente.

Las bases teóricas se fundamentaron en los aportes de Villar (2006), Cobo, Gómez y otros (2005), Levy (2004), entre otros. El tipo de investigación fue proyectiva, descriptiva, con un diseño de campo y bibliográfico; de acuerdo a los autores Balestrini (2006), Tamayo y Tamayo (2007), Namakforoosh (2005), entre otros. Se aplicaron técnicas de recolección de datos como entrevista, observación directa y encuesta, usándose como instrumentos la guía de visita, guía de entrevista, lista de cotejo y cuestionario, mediante los cuales se logró recolectar toda la información relacionada a los procesos administrativos actuales.

En cuanto a la población, fue finita compuesta por cincuenta y tres (53) personas: tres directivos y cincuenta clientes frecuentes de la Marmolería Bellorin C.A. La metodología seleccionada fue de tipo híbrida, tomada de los autores Kent Beck (2006) y Roger Pressman (2006) y se estructuró en cinco fases: comunicación con el cliente, planeación, diseño, codificación y pruebas.

Para el desarrollo de la aplicación informática de la Marmolería Bellorin C.A. se utilizaron los lenguajes HTML y CSS para la interfaz del sistema junto con PHP, JavaScript y MySQL para la programación de las tareas administrativas.

Dicha propuesta se sometió a las pruebas establecidas en la última fase de la metodología seleccionada, obteniendo como conclusión que la Aplicación de los Procesos Operativos de la Marmolería Bellorin C.A. cumplió con el objetivo propuesto, considerándose totalmente recomendable, no solo para dicha empresa sino también para otras que presenten las mismas necesidades.

La investigación anteriormente expuesta **Aporta** información importante concerniente a las herramientas y tecnologías utilizadas en el desarrollo de la aplicación informática bajo ambiente web; del mismo modo, la metodología utilizada para llevar a cabo dicha investigación sirve de referencia e igualmente es referente para las bases teóricas.

De igual manera, Bracho, Cárdenas, Corobo y Delgado, realizaron el trabajo especial de grado denominado “Aplicación Web con capacidad responsiva para la gestión de procesos de un centro de recursos audiovisuales de una Universidad Privada” para optar al título de Ingeniero en Informática en la Universidad Privada Dr. Rafael Belloso Chacín.

El objetivo principal de la investigación fue agilizar los procesos llevados a cabo en el centro de recursos audiovisuales con la finalidad de solucionar los inconvenientes existentes. La investigación se desarrolló en base a los aportes de Senn (2003), Kendall y Kendall (2005), Powell (2002), ente otros. Esta

investigación se clasificó como proyectiva, descriptiva y de campo por el método y la forma de obtener los datos.

Como técnicas de recolección de datos se utilizó un guion de sondeo, dirigido al personal docente de una universidad privada, así como también a los operadores que trabajan con los equipos, al igual que los encargados de dichos centros.

La metodología se desarrolló en base a los aportes de Senn (2003) y Pressman (2009) y consta de ocho (8) fases (Investigación Preliminar, Formulación, Planificación, Análisis, Ingeniería, Diseño de la estructura de datos, Generación de páginas, Puesta a prueba y Evaluación al cliente.

Se implementó como lenguaje de marcado HTML, para la programación se acudió a PHP y JavaScript para la elaboración de la aplicación web y como administrador de base de datos MySQL y finalizando con librerías de JQuery y Bootstrap, lográndose una aplicación web que contiene una interfaz sencilla, interactiva, beneficiando el centro de recursos audiovisuales, modificando la gestión de procesos de manual a automatizada.

El antecedente aquí referido **Aporta** información valiosa referente al desarrollo de aplicaciones web responsiva, herramientas y tecnologías utilizadas durante el desarrollo. Aunado a esto, la metodología utilizada ofrece experiencia valiosa que puede ser de utilidad en el desarrollo de la presente investigación, así como también su contenido teórico.

A su vez, se tiene el trabajo especial de grado para optar al título de Ingeniero en Informática en la Universidad Privada Dr. Rafael Bellosillo Chacín

realizado por Castro, Fonseca, Ocantó y Sánchez denominado “Aplicación informática bajo ambiente web para la gestión de mantenimiento del sistema de aire comprimido en el Centro Médico Docente Paraíso” con el propósito de mejorar el proceso de registro de indicadores en el área de mantenimiento dentro del centro de salud referido.

La investigación está basada en aportes metodológicos ofrecidos por Pressman (2006), Senn (2003), Kendall y Kendall (2009), Beck (2004), entre otros. El tipo de investigación se consideró proyectiva según la finalidad, descriptiva considerando el método de campo según la forma de obtener los datos y con un diseño no experimental. El proceso de recolección de datos se llevó a cabo valiéndose de técnicas como la entrevista, observación directa, encuestas y revisión documental, usando como instrumentos una guía de observaciones y un guion de entrevistas.

La población estuvo conformada por dos (2) gerentes, tres (3) supervisores y quince (15) auxiliares de mantenimiento. La metodología utilizada fue la de Desarrollo de sistemas Dinámicos (DSDM) desarrollada por el consorcio DSDM (2006) la cual está estructurada de la siguiente manera: Fase 1: Estudio de la viabilidad, Fase II: Estudio del negocio, Fase III: Iteración del modelo funcional, Fase IV: Iteración del diseño y de la construcción, Fase V: Implementación. La solución propuesta en esta investigación se basó en el análisis de la situación actual del proceso de registro de indicadores del Centro Médico Docente Paraíso, específicamente en el área de mantenimiento. Para el desarrollo de esta aplicación informática se utilizaron herramientas como

HTML para la codificación de la página Web, utilizando comandos basados en PHP, MySQL para la base de datos. Luego el sistema fue sometido a pruebas de aceptación, garantizando un buen funcionamiento.

Esta investigación **Aporta** información concerniente al desarrollo de aplicaciones bajo ambiente web, herramientas y las tecnologías utilizadas para los diferentes elementos del software como lo son la aplicación y la base de datos. Adicionalmente, la metodología utilizada en ella sustenta el trabajo actual.

Finalmente, Hernández, Pierini y Sánchez (2015) realizaron el trabajo especial de grado denominado “Aplicación Informática Inteligente bajo Ambiente Web para la Gestión de los Procesos Operativos en las Empresas de Papelería” para optar al título de Ingeniero en Informática en la Universidad Privada Dr. Rafael Belloso Chacín, cuyo propósito fue la automatización de los procesos operativos de las diferentes empresas de papelería.

El tipo de investigación se clasificó en proyectiva, descriptiva, de campo y documental con un diseño no experimental y bibliográfico. La población fue finita compuesta por veinticuatro (24) personas entre empleados y clientes que frecuentan las empresas de papelería. En cuanto a las técnicas de recolección de datos se utilizaron la observación directa, revisión documental, entrevista y encuesta. Se utilizó como instrumento la guía de visita, guía de observación, guía de entrevistas y el cuestionario, con la finalidad de identificar los procesos actuales y los requerimientos para la aplicación informática inteligente.

Por otra parte, para el desarrollo de la aplicación informática inteligente, se



planteó una metodología que se combina entre la propuesta por los autores Beck (2003) para la fase I, II y III Planificación, Diseño y Codificación y Powell (2003) para la fase IV pruebas. Dicha aplicación fue desarrollada en PHP, HTML, y JavaScript como lenguajes de programación y MySQL para la gestión de base de datos. Como resultado se obtuvo una aplicación informática inteligente bajo ambiente web adaptada a las necesidades y requerimientos de las empresas de papelería cumpliendo los objetivos de la investigación.

Dicha investigación realizó **Aportes** en lo referido al desarrollo de aplicaciones bajo ambiente web, así como también las herramientas y tecnologías utilizadas en el diseño y ejecución. La metodología utilizada para llevar a cabo esta investigación se basa en varios autores, por lo que tendrá un aporte importante. Adicionalmente, sus bases teóricas ayudarán a dar cuerpo a este trabajo.

## **2. BASES TEÓRICAS**

### **2.1. APLICACIÓN**

El término “aplicación” puede referirse a dos conceptos totalmente diferentes y es necesario diferenciarlos bien pues ambas definiciones son concernientes a la ingeniería del software. Primeramente, la palabra “aplicación” se utiliza para designar el propósito o dominio general que va a poseer un software; según Pressman (2010, p. 6), estos dominios son:

software de sistemas, el cual consiste en la utilización del mismo para dar servicio a otros programas; software de ingeniería y ciencias o “devoradores de números”, que esencialmente son programas utilizados para generar una gran cantidad de cálculos; software incrustados, como el tablero de un horno de microondas o el de una cocina eléctrica; software de línea de productos, diseñados para proporcionar una capacidad específica para uso de muchos consumidores diferentes como lo son los procesadores de texto y las hojas de cálculo; software de ingeniería artificial, software de aplicación y aplicaciones web. Los dos últimos son los más concernientes a la presente investigación.

Igualmente, Pressman (2010, p. 6) define software de aplicación de la siguiente manera:

programas aislados que resuelven una necesidad específica de negocios. Las aplicaciones en esta área procesan datos comerciales o técnicos en una forma que facilita las operaciones de negocios o la toma de decisiones administrativas o técnicas. Además de las aplicaciones convencionales de procesamiento de datos, el software de aplicación se usa para controlar funciones de negocios en tiempo real (por ejemplo, procesamiento de transacciones en punto de venta, control de procesos de manufactura en tiempo real).

Cabe destacar, que los softwares de aplicación son muy populares hoy en día dado que están orientados directamente a mejorar el funcionamiento de los negocios o a resolver problemas específicos que residen en los mismos; en otras palabras, constituyen una herramienta que impacta directamente en el funcionamiento de los mercados y la sociedad.

### 2.1.1 AMBIENTE WEB

Es imposible hablar de la Web sin hablar primero de internet, ya que la primera tecnología no puede existir sin la segunda; de acuerdo con Ramos, A. y Ramos (2014, p. 2), internet es “una gran red mundial de ordenadores conectados que, como todas las redes, permite compartir recursos e intercambiar información”. A su vez, Rodríguez (2007, p. 2) establece que internet “no es una simple red de ordenadores, sino una red de redes, es decir, un conjunto de redes interconectadas a escala mundial con la particularidad de que cada una de ellas es independiente y autónoma”.

Asimismo, internet está compuesta por diferentes espacios que ofrecen distintos servicios a los usuarios entre los que destacan: e-mail o servicio de correos, FTP (*File Transfer Protocol*) o protocolo de transferencia de archivos, boletines informativos, servicios de telecomunicaciones de red como Telnet y WWW (*World Wide Web*), que es lo que se conoce comúnmente como la Web. Según Ramos (2014, p. 2):

La web permite al usuario acceder a gran cantidad de información y a distintos tipos de documentos de manera sencilla; se le llama “la gran tela de araña” y es la herramienta más utilizada en internet. Básicamente, permite visualizar en la pantalla del usuario “páginas” (llamadas generalmente “sitios”) con información alojada en ordenadores remotos.

Adicionalmente, (Rodríguez, 2007, p. 16) asegura que la Web es el servicio que mayor aceptación por parte de los usuarios de la red. El éxito radica en su

facilidad de uso, ya que con una simple pulsación del ratón sobre una palabra o gráfico se puede pasar de una información o página a otra.

Actualmente, la Web se caracteriza principalmente por el gran despliegue de contenido multimedia y la gran interactividad con la que el usuario cuenta dentro del sitio, los cuales no están limitados a textos fijos, si no que ofrecen un amplio margen de operabilidad y posibilidades, que van desde la utilización de bases de datos para la administración de valiosa información como lo son los datos personales de un usuario, reproducción de videos, películas, noticias en tiempo real en incluso videojuegos. La Web ha trascendido de tal manera, que es difícil encontrar un espacio que no haya sido modificado directa o indirectamente por la presencia de la misma.

### **2.1.2 APLICACIÓN WEB**

Las aplicaciones Web se dividen en tres niveles: el primer nivel o nivel superior que interacciona con el usuario, cliente o navegador; luego si tiene el nivel inferior que proporciona los datos (comúnmente conocido como base de datos) y finalmente el nivel intermedio es aquel que procesa los datos, frecuentemente denominado servidor web (Luján, 2002, p. 47).

Consecuentemente, el mismo autor anteriormente citado expresa que:

Una aplicación web es un tipo especial de aplicación cliente/servidor, donde tanto el cliente (el navegador, explorador o visualizador) como el servidor (el servidor web) y el protocolo mediante el que se comunican (HTTP) están estandarizados y no han de ser creados por el programador de aplicaciones. El

protocolo HTTP forma parte de la familia de protocolos de comunicaciones TCP/IP, que son los empleados en Internet. Estos protocolos permiten la conexión de sistemas heterogéneos, lo que facilita el intercambio de información entre distintos ordenadores.

Por otra parte, Aumaille (2002, p. 26) estipula que una aplicación Web “ es un conjunto de recursos Web que participan en el funcionamiento de la propia aplicación Web”; entre dichos recursos destacan componentes de servidor dinámicos, bibliotecas de clases del lenguaje o los lenguajes de programación utilizados, elementos Web estáticos como lo son páginas HTML, imágenes, sonido y, por último, un descriptor de desarrollo y de configuración de la aplicación Web, que permita la ejecución de la aplicación Web, así como también la relación entre los componentes.

### **2.1.3 APLICACIÓN WEB RESPONSIVA**

En la actualidad, existen una gran variedad de dispositivos que poseen una pantalla y la capacidad de acceder y mostrar diferentes páginas Web. Sin embargo, estos dispositivos no siempre son similares entre sí; la variedad va desde ordenadores de escritorios o portátiles hasta dispositivo más livianos como lo son las tabletas y los teléfonos inteligentes.

Debido a esto, es muy importante para los desarrolladores de aplicaciones prever las características de los diferentes monitores que estarán haciendo uso de la aplicación Web, de manera que el producto final se comporte de una

manera deseada sin importar la pantalla o dispositivo que esté siendo utilizado. La capacidad que posee una aplicación Web para adaptarse a diferentes tamaños de pantalla se denomina responsiva.

Continuando con lo establecido anteriormente, (LaGrone, 2013, p. i) muestra una serie de técnicas para lograr este cometido entre la que destacan: ajuste de tamaño de imágenes, videos y tipografía; ajuste de la disposición de los diferentes elementos en una pantalla, estableciendo si se van a mostrar o no mediante herramientas de CSS3 y JavaScript; la utilización de las herramientas de desarrollo presente en los exploradores modernos para garantizar el comportamiento responsivo, entre otras tecnologías.

Por otra parte, (Marcotte, 2014, p. 11) determina que los ingredientes principales necesarios para el desarrollo de una aplicación Web responsiva son: un sistema de disposición basado en rejilla flexible, imágenes flexibles e instrucciones para el comportamiento multimedia utilizando CSS3 (*media queries*).

## **2.2 ANÁLISIS DE REQUERIMIENTOS**

En el proceso de desarrollo de cualquier software, es importante establecer clara y concisamente los requerimientos que debe poseer el software que se quiere desarrollar; así lo establece Kendal (2011, p. 10), quien establece que análisis de requerimientos busca:

determinar las necesidades de los usuarios involucrados,

mediante el uso de varias herramientas, para comprender la forma en que interactúan en el contexto laboral con sus sistemas de información actuales. El analista utilizará métodos interactivos como entrevistas, muestreos e investigación de datos duros, además de los cuestionarios y los métodos discretos, como observar el comportamiento de los encargados al tomar las decisiones y sus entornos de oficina, y los métodos integrales como la creación de prototipos.

Del mismo modo, Pressman (2010, p. 109) indica que el análisis de requerimientos tiene la finalidad de:

identificar el problema, proponer elementos de la solución, negociar distintos enfoques y especificar un conjunto preliminar de requerimientos de la solución en una atmósfera que favorezca el logro de la meta. Para entender mejor el flujo de eventos conforme ocurren, se presenta un escenario breve que bosqueja la secuencia de hechos que llevan a la reunión para obtener requerimientos, a lo que sucede durante ésta y a lo que sigue después de ella.

En conformidad con lo anteriormente presentado, la presente investigación primeramente indagará las necesidades y requerimientos del software, a la vez que propondrá varios prototipos para así poder obtener la mejor solución posible; este proceso se describe con más detalle en el capítulo 3 de esta investigación.

## **2.3 ARQUITECTURA PARA APLICACIONES WEB**

Al principio de los años 90, los sistemas Web consistían en una característica o herramienta adicionales que las empresas de entonces

poseían, por lo que no representaban ese factor decisivo que marcaba la diferencia en la competencia; esto permitía a los desarrolladores sacar de servicio la aplicación sin mayores consecuencias, realizar los cambios necesarios para satisfacer los requerimientos de usuarios de entonces para, finalmente, volver a colocar el sitio Web en línea.

Con el paso del tiempo, los sistemas Web requirieron ser más complejos, más robustos, de manera que no ya no era una opción sacar el sistema de línea para realizar alguna modificación, o desactivar la total funcionalidad de dicho sistema; es a partir de aquí, cuando se comienzan a estudiar las arquitecturas para aplicaciones Web monolíticas o clásica y la arquitectura basada en microservicios. Las mismas se describen a continuación.

### **2.3.1 ARQUITECTURA MONOLÍTICA**

Esta arquitectura es la más antigua que se conoce, y la que se utilizaba en los inicios de la Web, debido a su simplicidad y la rapidez con que permitía llevar a cabo el desarrollo de la aplicación Web. López y Maya (2017, p. 6) determinan que:

en una aplicación monolítica toda la lógica se ejecuta en un único servidor de aplicaciones. Las aplicaciones monolíticas típicas son grandes y construidas por múltiples equipos, requiriendo una orquestación cuidadosa del despliegue para cada cambio. También se consideran aplicaciones monolíticas cuando existen múltiples servicios de API que proporcionan la lógica de negocio, toda la capa de presentación es una sola aplicación web grande.



No obstante, aunque la arquitectura monolítica tiene la ventaja de otorgar soluciones rápidas a los problemas de desarrollo, su mantenimiento se complica a medida que aumenta la complejidad de la aplicación; así lo reseña Mora (2015, p. 87), quien establece que:

Tradicionalmente los servicios IT se han desarrollado utilizando una arquitectura monolítica, el cliente tiene una necesidad y se construye una solución, lo que se conoce como entregable. El gran problema de los servicios monolíticos es que tanto su mantenimiento como su evolución se complican con el tiempo.

Como solución a los problemas de escalabilidad presentados anteriormente, surge la arquitectura basada en microservicios, la cual se explica a continuación.

### **2.3.2 ARQUITECTURA BASADA EN MICROSERVICIOS**

Según López y Maya (2017, p. 4), definen arquitectura basada en microservicios de la siguiente manera:

es un enfoque para el desarrollo de una aplicación única como un conjunto de pequeños servicios, cada uno ejecutándose en su propio proceso y mecanismos ligeros de comunicación, a menudo un recurso de una interfaz de programación de aplicaciones (API) sobre protocolo de transferencia de hipertexto (HTTP). Estos servicios están contruidos alrededor de las capacidades del negocio y con independencia de despliegue e implementación totalmente automatizada. Existe un mínimo de gestión centralizada de estos servicios, los que pueden estar escritos en lenguajes de programación diferentes y utilizar diferentes tecnologías de almacenamiento de datos.

En concordancia con la idea anterior, (Mora 2015, p. 87) plantea que, aunque el modelo monolítico ha demostrado ser una arquitectura sólida que ha funcionado durante años, ha surgido una nueva tendencia de diseñar la arquitectura de un servicio. Se trata de dividir el servicio en microservicios que puedan interaccionar entre ellos, para fortalecer las mismas funcionalidades de una solución monolítica, pero sin los inconvenientes de ésta.

**Cuadro 1**  
**Diferencias entre arquitecturas monolíticas y microservicios**

<b>Categoría</b>	<b>Arquitectura Monolítica</b>	<b>Arquitectura de microservicios</b>
Código	Una base de código única para toda la aplicación.	Múltiples bases de código. Cada microservicio tiene su propia base de código.
Comprensibilidad	A menudo difuso y difícil de entender.	Despliegue sencillo ya que cada microservicio se puede implementar de forma individual, con un tiempo de inactividad mínimo, si no es cero.
Despliegue	Implementaciones complejas con ventanas de mantenimiento y paradas programadas.	Despliegue sencillo ya que cada microservicio se puede implementar de forma individual, con un tiempo de inactividad mínimo, si no es cero.
Lenguaje	Típicamente totalmente desarrollado en un lenguaje de programación.	Cada microservicio puede desarrollarse en un lenguaje de programación diferente.
Escalamiento	Requiere escalar la aplicación entera, aunque los cuellos de botella estén localizados.	Cada microservicio puede desarrollarse en un lenguaje de programación diferente.

**Fuente: López y Maya (2017, p. 5)**

Una arquitectura de microservicios promueve el desarrollo y despliegue de aplicaciones compuestas por unidades independientes, autónomas, modulares y auto contenidas, lo cual difiere de la forma tradicional o monolítico. Una de las ventajas de utilizar microservicios es la capacidad de publicar una aplicación grande como un conjunto de pequeñas aplicaciones que se pueden desarrollar, desplegar, escalar, manejar y visualizar de forma independiente.

Los microservicios permiten a las empresas gestionar las aplicaciones de código base grande usando una metodología más práctica donde las mejoras incrementales son ejecutadas por pequeños equipos en bases de código y despliegues independientes (López y Maya 2017, p. 5).

## **2.4 SERVIDORES WEB**

Con el auge de las nuevas tecnologías de telecomunicaciones e internet, han surgido una gran cantidad de aplicaciones con servicios concentrados en la web; algunos ejemplos son: aplicaciones asociadas a la banca, mediante las cuales existe una gestión centralizada de datos personales y datos asociados a los balances de los usuarios; aplicaciones de servicios en el ámbito automotriz, donde se almacenan datos referentes a uno o varios automóviles en equipos centralizados y, más conocida por la mayoría, las redes sociales, que permiten la interacción entre individuos separados por grandes distancias.

Las aplicaciones anteriormente mencionadas, y muchas otras existentes, no podrían operar sin la presencia de los servidores. Según Mateu (2004, p. 23) un servidor web “es un programa que atiende y responde a las diversas peticiones de los navegadores, proporcionándoles los recursos que solicitan mediante el protocolo HTTP o el protocolo HTTPS (la versión segura, cifrada y autenticada de HTTP)”. Por otra parte, (Luján, 2002, p. 39) determina que los servidores son ordenadores potentes dedicados a gestionar unidades de disco, impresoras, tráfico de red, datos o incluso aplicaciones.

Adicionalmente, (Aumaille, 2002, p. 12) indica que el servidor debe encargarse de la creación y de la carga en memoria de las instancias de los componentes, así como de la gestión de una de espera para satisfacer las peticiones de los clientes. Además, para satisfacer las exigencias de las aplicaciones corporativas, el servidor de aplicación debe ser potente y fiables.

De los autores previamente citados, se puede extraer que los servidores son equipos con alto poder de cómputo y almacenamiento, cuya finalidad es dar respuesta a las peticiones de clientes referentes a gestión y almacenamiento de datos.

#### **2.4.1 APACHE**

Para poder operar, todos los equipos de servidores necesitan un software que les permita operar como tal; entre las diferentes soluciones que hay disponibles en el mercado se encuentra el servidor Apache, el cual, según

(Mateu, 2002, p. 29) es un servidor web de código libre, robusto, cuya implementación se realiza de forma colaborativa, con prestaciones y funcionalidades equivalentes a las de los servidores comerciales. El mismo autor también plantea que el proyecto está dirigido y controlado por un grupo de voluntarios de todo el mundo que, mediante el uso de internet y la web, se comunican, planifican y desarrollan el servidor, así como a su vez la respectiva documentación relacionada.

El servidor web Apache no tiene costo monetario asociado a su utilización puesto que el proyecto se mantiene a flote gracias a la colaboración monetaria de individuos en instituciones interesadas y la participación de una amplia comunidad que vela por su correcto funcionamiento

## **2.5 PROGRAMACIÓN CLIENTE (FRONTEND)**

El incremento de la demanda de la producción de aplicaciones web, ha hecho que estas deban producirse con más velocidad y de una manera más ordenada y estructurada, sin importar la complejidad del proyecto en cuestión. Este hecho ha generado un nuevo paradigma en el que, el trabajo total a realizar, se separa en dos grandes campos: lado cliente o frontend y lado servidor o backend.

Consecuentemente, el lado cliente corresponde con la interfaz con la que el usuario estará interactuando y el lado servidor se refiere a la programación de base de datos y actividades que no corresponde que ocurran en el dispositivo

del cliente. A continuación, se describen algunas de las tecnologías más populares utilizadas hoy en día para la programación del lado cliente.

### **2.5.1 HYPER TEXT MARKUP LANGUAGE (HTML)**

En términos generales, un sitio web posee elementos que son generalmente comunes, conformados por texto tales como títulos, bloques de texto o párrafos y listas; otros tipos de contenido funcionales como lo son los formularios y los botones de ingresar información y elementos multimedia entre los que destacan imágenes, música y videos.

Dado que el tipo de contenido está más o menos predeterminado, un desarrollador generalmente no comienza “desde cero” a la hora de programar un sitio web, puesto que los distintos elementos mencionados con anterioridad ya están predefinidos en el lenguaje de etiquetas HTML.

Según Luján (2002, p. 91) “HTML es un lenguaje de marcas (etiquetas) que se emplea para dar formato a los documentos que se quieren publicar en la web. Los navegadores son capaces de interpretar las etiquetas y mostrar los documentos con el formato deseado”; a su vez, Mateu (2004, p. 51) indica que el lenguaje de etiquetas HTML (*hypertext markup language*) por sus siglas en inglés, “se utiliza para crear documentos que muestren una estructura de hipertexto”.

No obstante, dadas las ventajas que aporta la tecnología anteriormente mencionada al proceso de creación y funcionamiento de un sitio web, las

mismas limitan en el sentido que no permiten al desarrollador configurar la estética del sitio en cuestión; para esto, se utiliza la tecnología que se describe a continuación.

### **2.5.2 CASCADE STYLE SHEET (CSS)**

La hoja de estilos en cascada (*cascade style sheets* por sus siglas en inglés) es la herramienta especial utilizada para aportar estilos a un sitio web; de acuerdo con Mateu (2004, p.82), “las hojas de estilo son un mecanismo para separar el formato de representación y presentación del contenido. Eso se consigue asociando atributos de presentación a cada una de las etiquetas de HTML o a subclases de éstas”.

Esto permite, a la vez que se obtiene un entorno de trabajo más ordenado al realizar la separación de la hoja de etiquetas y la hoja de estilos, aplicar la misma hoja de estilos a múltiples hojas de etiqueta, lo que acelera extraordinariamente el proceso de producción de software. A su vez, Luján (2002, p. XXII), define CSS como:

tecnología empleada en la creación de páginas web, que permite un mayor control sobre el lenguaje HTML. Permite crear hojas de estilo que definen como cada elemento, como por ejemplo los encabezados o los enlaces, se tiene que mostrar. El término “en cascada” indica que diferentes hojas de estilo se pueden aplicar sobre la misma página.

Como nota adicional, cabe destacar que las hojas de estilo en cascadas son

muy versátiles, de manera que la capacidad de aplicarse a cualquier hoja de etiquetas HTML y la opción de aplicar múltiples hojas de estilos a la misma hoja de etiquetas, dan un sinnúmero de posibilidades que pueden ser aprovechadas por los desarrolladores.

### **2.5.3 JAVASCRIPT**

Al momento del desarrollo de un sitio web, el lenguaje de etiqueta HTML y el lenguaje de estilos CSS aportan un gran conjunto de herramientas, pero no permiten resolver todos los problemas que puedan aparecer dado que no son lenguajes de programación; por lo anteriormente mencionado, existe JavaScript, el cual, según Mateu (2004, p. 93) “es un lenguaje de programación interpretado (un lenguaje de tipo script). A pesar de que existen intérpretes no dependientes de ningún navegador, es un lenguaje de script que suele encontrarse vinculado a páginas web”.

Por otra parte, JavaScript es un lenguaje interpretado, basado en objetos y multiplataforma, inventado por Netscape Communications Corporation. Los navegadores de Netscape fueron los primeros que usaron JavaScript. El primer nombre oficial de este lenguaje fue LiveScript y apareció por primera vez en la versión beta de Netscape Navigator 2.0 en septiembre de 1995, pero poco después fue rebautizado JavaScript en un comunicado conjunto con Sun Microsystems (Luján, 2002, p. 182).

Como punto a favor, se tiene que JavaScript contiene una serie de objetos



predefinidos (arreglos, objetos matemáticos y de fecha), y un conjunto de elementos del lenguaje como operadores, estructuras de control y sentencias (Luján, 2002, p. 182).

Hoy JavaScript es uno de los lenguajes de programación más utilizados, dado que todo o gran parte del desarrollo web gira entorno a dicho lenguaje; aunado a esto, el aumento proporcional de los navegadores web, desde los de computadores personales hasta los que se usan en teléfonos inteligentes, han hecho que JavaScript esté más presente que nunca, sin mencionar la inmensa inversión que ha recibido por parte de grandes empresas como Facebook y Google para mejorar sus características y su desempeño.

#### **2.5.4 BOOTSTRAP**

En el ámbito del desarrollo de software, el término “framework” (que en inglés significa marco o armazón) es utilizado frecuentemente para referirse a librerías, que no forman parte de las tecnologías fundamentales del entorno web, sino que, basándose en ellas, ofrecen un paquete que permite aligerar, estandarizar y hacer más robusto el producto final.

Tal es el caso de Bootstrap, que de acuerdo a lo publicado en su página oficial Bootstrap (04/07/19), es un conjunto de herramientas, creadas bajo un código abierto, utilizado para el desarrollo con HTML, CSS y JavaScript, que permite el prototipado rápido de ideas y la construcción aplicaciones completas mediante el uso de diversas herramientas como sistemas de rejillas responsivos y

componentes preconstruidos.

El conjunto de herramientas Bootstrap es mantenido por un grupo pequeño de desarrolladores en la plataforma GitHub (que es una base de datos que alberga gran cantidad de proyectos de código abierto). Originalmente fue diseñado y creado en Twitter, y, con el pasar del tiempo, se ha convertido en el conjunto de herramientas para el desarrollo del lado cliente más popular del mundo. (Bootstrap ,04/07/19).

### **2.5.5 REACT**

La librería React, de acuerdo con su sitio web oficial, es una librería de JavaScript utilizada para el desarrollo de interfaces gráficas. Mediante el diseño, administración y utilización de componentes, React hace más sencillo del proceso de desarrollo de interfaces gráficas, permitiendo la compilación en tiempo de real de datos que cambian al instante.

Adicionalmente, permite la encapsulación de unos componentes dentro de otros, lo que otorga una gran versatilidad a la hora del desarrollo; los mismos componentes tienen la capacidad de manejar su propio estados o datos, lo que aporta un entorno de desarrollo más confiable.

Por otra parte, puesto que React es una librería creada con JavaScript, la misma es configurable y versátil, lo que permite agregar funcionalidades adicionales que dañar las ya existentes. Aunado a esto, esta librería permite compilarse del lado del servidor y posee una extensión denominada React Native,

para el desarrollo de aplicaciones móviles, sin la necesidad de tener que desarrollar en código nativo de los dispositivos (React, 04/07/19).

## **2.6 PROGRAMACIÓN DE SERVIDORES (BACKEND)**

A continuación, se definen las herramientas, los lenguajes de programación y el framework a utilizar para la programación del lado del servidor de la aplicación, así como también el sistema de base de datos seleccionado.

### **2.6.1 PYTHON**

Existen diferentes lenguajes de programación que se pueden utilizar del lado servidor en el desarrollo de aplicaciones web, tal es el caso de Python, que no solo es un lenguaje orientado a la web, sino que tiene un carácter multipropósito, sencillo y poderoso, con la cualidad de ser orientado a objetos. Entre alguna de las características más notables de Python destacan:

- Utiliza una sintaxis elegante, que hace que los programas sean más fáciles de leer y entender por los desarrolladores.
- Es un lenguaje fácil y rápido de utilizar, lo que lo convierte ideal para realizar prototipos, sin comprometer la mantenibilidad.
- Incluye una extensa librería estándar que soporta muchas de las actividades comunes a la hora de programar, tales como conexión a

servidores web, búsqueda de texto y expresiones regulares y lectura y modificación de archivos.

- El modo interactivo de Python permite que la prueba de pequeños bloques de código sea rápida y fácil. También existen varios entornos de desarrollo para pruebas más exhaustivas.
- Se puede extender fácilmente agregando nuevos módulos compilados en lenguajes tales como C y C++.
- Los programas corren en cualquier plataforma, incluyendo Mac OS X, Windows, Linux y Unix, con paquetes no oficiales también para Android y iOS.
- Es un software libre en dos sentidos; no tiene ningún costo asociado a la descarga y su utilización, y puede ser libremente modificado y redistribuido, dado que opera bajo la licencia de código abierto (Python, 04/07/19).

Las características anteriormente nombradas son solo algunas, pero suficiente para entender por que Python se ha vuelto uno de los lenguajes de programación multipropósitos más utilizados en el mundo, cuya utilización abarca desde el desarrollo de videojuegos (pygame), minería de datos (pandas y numpy), desarrollo de inteligencia artificial (scikit-learn) y servidores web (django y flask) (Matthes, 2016, p. 5).

### **2.6.2 DJANGO**

Django es un framework de entorno web de alto nivel de Python que se enfoca en un desarrollo rápido y limpio de aplicaciones, utilizando un diseño pragmático; el mismo ha sido construido por desarrolladores experimentados, y resuelve mucho de los problemas más comunes en lo que se refiere al desarrollo web, lo que permite a los desarrolladores concentrarse en el trabajo importante sin que tengan la necesidad de inventar la rueda. Además, el mismo funciona bajo las políticas de código abierto.

Este framework se caracteriza por ser extremadamente rápido, puesto que fue diseñado teniendo como meta incrementar la velocidad en el desarrollo de aplicaciones. Así mismo, es extremadamente seguro, puesto que toma los aspectos de seguridad con mucha seriedad, ayudando a los desarrolladores a evitar los errores más comunes en este ámbito.

Aunado a lo anterior, Django posee la cualidad de ser escalable, lo que permite expandir cualquier sitio web, desde el de menos tráfico hasta el más saturado, en un periodo de tiempo relativamente corto, evitando así largos periodos de fuera de línea de los sitios en cuestión (Django, 04/07/19).

## **2.7 BASES DE DATOS**

Todo sistema web completo debe incluir un sistema gestor de base de datos, dado que su funcionalidad usualmente se enfoca en el correcto almacenamiento

y manejo de datos. Según Silberschatz, Korth y Sudarshan (2006, p. 1), un sistema gestor de bases de datos:

consiste en una colección de datos interrelacionados y un conjunto de programas para acceder a dichos datos. La colección de datos, normalmente denominada base de datos, contiene información relevante para una empresa. El objetivo principal de un SGBD es proporcionar una forma de almacenar y recuperar la información de una base de datos de manera que sea tanto práctica como eficiente.

En concordancia, los sistemas de bases de datos están diseñados para gestionar grandes cantidades de información, lo que implica determinada definición de estructuras para almacenar la información como la provisión de mecanismos para su correcta manipulación. Por otra parte, los sistemas de bases de datos deben garantizar la fiabilidad de la información almacenada, sin importar las caídas de sistemas o intentos de acceso no autorizados (Silberschatz y otros, 2006, p. 1).

Por otra parte, según Coronel, Morris y Ros (2011, p. 7), un sistema de administración de bases de datos (DBMS por sus siglas en inglés):

es un conjunto de programas que maneja la estructura de la base de datos y controla el acceso a los datos guardados en ella. En cierto sentido, una base de datos se asemeja a un muy bien organizado gabinete electrónico de archivos en el que un poderoso software, conocido como sistema de administración de base de datos, ayuda a manejar el contenido del gabinete.

Existen dos grandes grupos de bases de datos: las bases de datos relacionales y las no relacionales. Para el presente proyecto, se utilizó un sistema de bases de datos no relacional, el cual se basa en simples archivos de texto en

los que almacenan los datos bajo una estructura predefinida. El sistema a utilizar será MongoDB.

### **2.7.1 MONGODB**

Mongodb es un sistema de administración de bases de datos de propósito general basada en documentos (no relacional), utilizada mayormente para el desarrollo de aplicaciones modernas, ideada para la era de la virtualización. Este sistema almacena los datos mediante formato JSON (*Javascript Object Notation* por sus siglas en inglés), que es altamente utilizado por aplicaciones modernas, el cual, según los desarrolladores de esta tecnología, es la manera más natural de pensar acerca de los datos, y es mucho más expresiva y poderosa que la manera tradicional basada en filas y columnas (MongoDB, 07/07/19).

Adicionalmente, MongoDB posee un lenguaje bastante expresivo que permite filtrar los datos utilizando cualquier campo, si importar que tan indentado pueda estar el mismo dentro del documento; también soporta agregación de otras utilidades modernas como búsquedas basadas en geolocalización, búsquedas gráficas y, por supuesto, de texto (MongoDB, 07/07/19).

Cabe destacar, que la librería Django posee las herramientas prescindibles para el desarrollo integrado con el entorno de MongoDB, lo que promete, como resultado, el desarrollo de una aplicación robusta y segura, confiable y rápida.

### **3 SISTEMA DE VARIABLES**

En el sistema de variables, se definen nominal, conceptual y operacionalmente las variables presentadas en esta investigación son:

- Aplicación Web Responsiva
- Cálculo de Tarifa

#### **3.1 DEFINICIÓN NOMINAL**

Aplicación Web Responsiva para el cálculo de tarifas de servicios de transporte terrestre.

#### **3.2 DEFINICIÓN CONCEPTUAL**

##### **APLICACIÓN WEB RESPONSIVA**

Según Luján (2002, p. 47), es, “un tipo especial de aplicación cliente/servidor, donde tanto el cliente (el navegador, explorador o visualizador) como el servidor (el servidor web) y el protocolo mediante el que se comunican (HTTP o *Hyper Text Transfer Protocol* por sus siglas en inglés) están estandarizados y no han de ser creados por el programador de aplicaciones. El protocolo HTTP forma parte de la familia de protocolos de



comunicaciones TCP/IP, que son los empleados en Internet. Estos protocolos permiten la conexión de sistemas heterogéneos, lo que facilita el intercambio de información entre distintos ordenadores”.

Para el presente trabajo, la aplicación web posee capacidad responsiva, lo que implica que los elementos visuales se adaptan a las distintas características de las diferentes pantallas de los dispositivos que la utilizarán.

## **CÁLCULO DE TARIFA**

De acuerdo con Todd y Doherty (2002, p. 9), una tarifa se refiere a un costo interno que varía en función de la percepción del usuario o un costo que tiende a incrementarse y que el usuario debe abordar para poder disfrutar de un servicio. Estos costos afectan directamente las decisiones de consumo; por ejemplo, el cambio en el precio del combustible, tarifas de estacionamiento y el costo de los peajes afectan la decisión de hacer un viaje de un consumidor. Una economía eficiente requiere que los precios reflejen el costo total de producción de un bien para así, aportar marcadores de mercados adecuados”.

### **3.3 DEFINICIÓN OPERACIONAL**

#### **APLICACIÓN WEB RESPONSIVA**

Una aplicación web es un software que tiene la finalidad de ofrecer servicios

de información descentralizados mediante la utilización de servidores y bases de datos, a un grupo diversos de usuarios que utilizan distintos dispositivos con conexión a internet y navegadores integrados que tienen la capacidad de operar con protocolo de hipertexto. La misma puede poseer capacidad responsiva, lo que permite que dicha aplicación se muestre y se comporte diferentemente en función de las dimensiones de la pantalla y la naturaleza del dispositivo que la utiliza.

## **CÁLCULO DE TARIFA**

El cálculo de tarifa se refiere a la valoración económica de un servicio prestado, la cual deberá ser cancelado por un usuario para el disfrute del mismo; el cálculo de determinada tarifa es variante desde varios puntos de vista.

Primeramente, se tiene el entorno de oferta y demanda, el cual es independiente del costo de producción del bien en cuestión, y este recae exclusivamente en la valoración que el usuario tenga del servicio; por otra parte, está el entorno de producción, donde el costo del servicio otorgado se realiza en función de los costos derivados que implica otorgar el servicio prestado. Ambos costos son importantes a la hora de realizar el cálculo de la tarifa del servicio que se desea prestar.