DESARROLLO DE UN SISTEMA FACTURADOR WEB PARA PEQUEÑAS Y MEDIANAS EMPRESAS

DAYANA KATHERINE SILVA GAITÁN

COD.: 20181578064

UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS

FACULTAD TECNOLÓGICA

TECNOLOGÍA EN SISTEMATIZACIÓN DE DATOS

BOGOTÁ D.C.

2022

DESARROLLO DE UN SISTEMA FACTURADOR WEB PARA PEQUEÑAS Y MEDIANAS EMPRESAS

DAYANA KATHERINE SILVA GAITÁN

COD.: 20181578064

Trabajo de grado para obtener el título de tecnólogo en sistematización de datos

Tutora: Sonia Alexandra Pinzón Núñez

UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS

FACULTAD TECNOLÓGICA

TECNOLOGÍA EN SISTEMATIZACIÓN DE DATOS

BOGOTÁ D.C.

2022

Nota de aceptación

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tutora del proyecto

Ing. Sonia Alexandra Pinzón Núñez

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Jurado

Ing. Johanna Del Pilar Dueñas Galindo

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Segundo Jurado

**Dedicatoria**

Primeramente, le dedico este logro a Dios quien es mi fortaleza día a día; a mis padres y hermana por su apoyo y motivación en todo momento y a mis docentes universitarios quienes fueron un pilar esencial en todo el camino de mi aprendizaje.

**Agradecimientos**

Le agradezco a Dios por darme la sabiduría, la disciplina y el entendimiento necesario para desarrollar este trabajo de grado, a mi familia por su incondicional apoyo, amor y motivación; a la Universidad Distrital Francisco José de Caldas por permitirme tener una excelente formación académica, a los docentes por darme la oportunidad de ser mejor cada día y por compartir su aprendizaje para conmigo; y a mi tutora por el tiempo y las pautas que me brindó para la realización del mismo.

**Tabla de contenido**

Resumen

Abstract

Introducción

1. Fase de definición, planeación y organización
   1. Titulo
   2. Tema
   3. Planteamiento del problema
      1. Descripción
      2. Formulación
   4. Objetivos
      1. Objetivo general
      2. Objetivos específicos
   5. Justificación
   6. Alcances y delimitaciones
      1. Alcances
      2. Delimitaciones
   7. Marco de referencia
      1. Fuentes primarias
   8. Marco teórico
      1. Marco histórico
      2. Aplicaciones web
         1. Características generales de las aplicaciones web
         2. Componentes de las aplicaciones web
            1. Bases de datos
            2. Código de compilación
            3. Acceso a través del navegador
      3. Sistema facturador web
         1. Características de un sistema facturador web
            1. Gestión de facturas
            2. Pagos en línea
            3. Gestión de clientes
            4. Gestión de personal
      4. Servicios web
         1. REST
            1. Principios en los que se basa REST
            2. Ventajas de REST para el desarrollo
      5. Herramientas de desarrollo
         1. Java
         2. Spring boot
         3. JavaScript
            1. Características de JavaScript

Tipos de datos

Operadores

Comentarios

Uso de un archivo externo

* + - * 1. Librería ReactJS
      1. HTML
      2. CSS
      3. Bootstrap
      4. MySQL
      5. Eclipse IDE
      6. Visual Studio Code
  1. Marco conceptual
     1. Pyme
     2. Cliente
     3. Producto
     4. Factura
     5. Usuario
     6. Gestión de la información
  2. Factibilidad
     1. Factibilidad técnica
     2. Factibilidad operativa
     3. Factibilidad legal
     4. Factibilidad económica
        1. Presupuesto y financiación
        2. Recursos humanos
        3. Recursos tecnológicos
        4. Recursos varios
        5. Costo total
  3. Metodología
  4. Cronograma de actividad

1. Desarrollo
   1. Product backlog
      1. Sprint backlog
   2. Sprints
      1. Sprint 1: Creación de base de datos, paquetes y clases.
         1. Planning
         2. Execution
         3. Dailys
         4. Review
         5. Retrospective
      2. Sprint 2: Página principal y Login
         1. Planning
         2. Execution
         3. Dailys
         4. Review
         5. Retrospective
      3. Sprint 3: Registro de Pymes
         1. Planning
         2. Execution
         3. Dailys
         4. Review
         5. Retrospective
      4. Sprint 4: Módulo Usuarios
         1. Planning
         2. Execution
         3. Dailys
         4. Review
         5. Retrospective
      5. Sprint 5: Módulo Productos
         1. Planning
         2. Execution
         3. Dailys
         4. Review
         5. Retrospective
      6. Sprint 6: Módulo Clientes
         1. Planning
         2. Execution
         3. Dailys
         4. Review
         5. Retrospective
      7. Sprint 7: Módulo Facturas
         1. Planning
         2. Execution
         3. Dailys
         4. Review
         5. Retrospective
      8. Sprint 8: Módulo de consulta de facturas por referencia y descarga
         1. Planning
         2. Execution
         3. Dailys
         4. Review
         5. Retrospective

**Lista de tablas**

*Tabla 1. Delimitación del proyecto*

*Tabla 2. Herramientas que se usarán para el desarrollo del proyecto*

*Tabla 3. Recursos físicos para el desarrollo del proyecto*

*Tabla 4. Recursos humanos para el desarrollo del proyecto*

*Tabla 5. Presupuesto en recursos humanos*

*Tabla 6. Presupuesto en recursos tecnológicos*

*Tabla 7. Presupuesto en recursos varios*

*Tabla 8. Costo total del presupuesto para el proyecto*

*Tabla 9. Sprints del proyecto*

*Tabla 10. Historias de usuario*

*Tabla 11. Planning Sprint 1*

*Tabla 12. Tareas Sprint 1*

*Tabla 13. Diccionario de datos - Ciudad*

*Tabla 14. Diccionario de datos - Empresa*

*Tabla 15. Diccionario de datos - Usuario*

*Tabla 16. Diccionario de datos - Categoría*

*Tabla 17. Diccionario de datos - Impuesto*

*Tabla 18. Diccionario de datos - Estado*

*Tabla 19. Diccionario de datos – Forma Pago*

*Tabla 20. Diccionario de datos – Documento*

*Tabla 21. Diccionario de datos – Producto*

*Tabla 22. Diccionario de datos – Cliente*

*Tabla 23. Diccionario de datos – Factura*

*Tabla 24. Diccionario de datos - Impuesto Factura*

*Tabla 25. Diccionario de datos - Detalle*

*Tabla 26. Diccionario de datos – Categoría Empresa.*

**Lista de figuras**

*Figura 1. Arquitectura de una aplicación web*

*Figura 2. Arquitectura de una API REST*

*Figura 3. Arquitectura de microservicios JAVA con Spring Boot*

*Figura 4. Cronograma de actividades, parte 1*

*Figura 5. Cronograma de actividades, parte 2*

*Figura 6. Cronograma de actividades, parte 3*

*Figura 7. Cronograma de actividades, parte 4*

*Figura 8. Modelo relacional de base de datos*

*Figura 9. Estructura proyecto backend*

*Figura 10. Dependencias backend*

*Figura 11. Estructura proyecto frontend*

*Figura 12. Dependencias frontend*

**Resumen**

Por lo general, pequeñas y medianas empresas realizan sus procesos de facturación por medio de formatos establecidos en papel; donde en muchas ocasiones se suelen cometer errores en la diligencia de las facturas, de modo que conlleva un mayor tiempo realizarlas. Además, suelen generarse pérdidas de las facturas realizadas o de los datos de los clientes, impidiendo una buena organización con dichos procesos.

Asimismo, este procedimiento genera mayores gastos en papelería para las empresas, no garantiza la preservación de la información y no es amigable con el medio ambiente; de este modo, el proceso para facturar y gestionar resulta ser tedioso y poco práctico para las pymes.

Con el objetivo de evitar las situaciones anteriormente mencionadas, se propone desarrollar un sistema facturador web que permita procesar la información que requieren las pequeñas y medianas empresas para mejorar organizacionalmente el proceso de facturar.

Para el desarrollo del sistema facturador web se realizó un análisis de aplicaciones similares a la desarrollada y de diferentes tecnologías que brindaran una solución eficiente a los objetivos propuestos. Por ello, se hizo uso de servicios web REST que se comunican por medio del lenguaje JAVA y la librería React de JavaScript.

Adicionalmente, el proyecto se desarrolló siguiendo los lineamientos de la metodología ágil Scrum.

**Abstract**

In general, small and medium-sized companies carry out their billing processes through established paper formats; where errors are often made in the diligence of invoices, so it takes a longer time to perform them. In addition, losses of invoices or customer data are often generated, preventing a good organization with such processes.

In addition, this procedure generates higher costs for companies in the form of stationery, does not guarantee the preservation of information and is not environmentally friendly; In this way, the process of invoicing and managing turns out to be tedious and impractical for small and medium-sized companies.

In order to avoid the aforementioned situations, it is proposed to develop a web billing system that allows processing the information required by small and medium-sized companies to improve the billing process.

For the development of the web billing system, an analysis was made of applications similar to the one proposed and of different technologies that would provide an efficient solution to the proposed objectives. Therefore, REST web services were used that communicate through the JAVA language and the React library of JavaScript.

Additionally, the project was developed following the guidelines of the Scrum agile methodology.

**Introducción**

Con el paso del tiempo, el uso de las tecnologías web ha incrementado gracias a las diferentes soluciones que brindan en diversas áreas según las necesidades de una empresa o persona; es por esto, que pueden ser de carácter informativo, aplicativo, publicitario, interactivo, entre otros. Son bastantes los beneficios que se pueden obtener con el uso de las mismas; sin embargo, aún se continúan empleando métodos tradicionales como el papel, el cual puede generar ciertas dificultades en distintos procesos organizacionales; por ejemplo, en las empresas puede ocasionar impedimentos en el control de documentos primordiales, como lo son las facturas.

Actualmente, existen algunos sistemas que permiten la creación y consulta de facturas; sin embargo, algunos no son web, no son gratuitos, no cuentan con exportación de datos, no brindan la funcionalidad de consultar facturas a los clientes y no cuentan con módulos para administrar la información que implica la realización de una factura.

Con el fin de proporcionar una herramienta web para facturación, se desarrolla el sistema facturador web ISSMC; el cual es un sistema aplicativo para pequeñas y medianas empresas, realizado con el fin de mitigar el tiempo que tiene el proceso de facturar para las pymes; en donde se registra la pyme por medio de un formulario adjuntando logo, información básica, datos del primer usuario administrador y categorías de los productos que maneja. Posteriormente al registro, permite la administración de usuarios, clientes, productos, categorías, crear, anular, y exportar facturas.

En el presente documento, se encontrará el desarrollo de un conjunto de servicios que conformarán el sistema ISSMC, el cual implementa el uso de diferentes tecnologías y herramientas; entre ellas, servicios web REST, del lado del servidor el uso de Java, JPA, Spring Boot, del lado del cliente uso de React, Bootstrap, HTML, CSS y para la gestión de datos el sistema gestor relacional MySQL; para el aseguramiento de la clave de registro e ingreso para cada usuario, se empleó doble encriptamiento con AES y SHA256.

Para obtener consultas agiles, se da la posibilidad de filtrar la información por campos, el ordenamiento de cada columna, paginación con número de registros configurables y un color en específico para algunos registros según parámetros determinados en los datos. Así mismo, las inserciones y ediciones cuentan con validaciones a fin de mitigar errores en la información; y para las eliminaciones de registros que cuenten con dependencias relacionales se usaron borrados lógicos para evitar perdida de la misma. Por último, se incorporó exportación en PDF de las facturas, exportación en archivos Excel para clientes, usuarios y productos.

1. **Fase de definición, planeación y organización**
   1. **Título**

Desarrollo de un sistema facturador web para pequeñas y medianas empresas.

* 1. **Tema**

Sistema web apoyado con servicios web Rest como herramienta para facturar en pymes.

* 1. **Planteamiento del problema**
     1. **Descripción**

Actualmente algunas pequeñas y medianas empresas, manejan los procesos de facturación de manera tradicional; donde por medio del papel realizan dichos procedimientos para emitir las facturas de los productos que ofrecen en el mercado. El proceso que se lleva a cabo para dicha gestión, consiste en tener una lista de clientes con sus datos básicos de contacto por medio libretas o herramientas ofimáticas, como lo son Word o Excel. Posteriormente, a partir de dicha lista se crean las facturas en papel con una copia para el cliente; donde esta contiene los productos a facturar con su respectivo valor, datos del cliente, fecha de emisión, fecha de vencimiento, observaciones, costo en impuestos y valor total.

“Hacer la facturación en papel se ha convertido en algo tedioso y obsoleto a la vez que se emplean numerosas horas a lo largo del año para llevar a cabo esta gestión.” [27] Por ello, la tarea de facturar mediante el proceso anteriormente descrito produce pérdida de información, mayores gastos en papel y material de oficina, mayor riesgo de cometer errores y consume más tiempo para su realización. Consecuentemente, ese modo de funcionamiento genera dificultades en diferentes aspectos organizacionales para las pymes.

“Existen muchas empresas que por miedo a los cambios siguen emitiendo las facturas en papel y se quedan sin saber los beneficios de la facturación online.” [27]

* + 1. **Formulación**

¿Cómo a través del desarrollo de un sistema facturador web se logran mejorar los procesos de facturación para pequeñas y medianas empresas?

* 1. **Objetivos**
     1. **Objetivo general**

Desarrollar un sistema facturador web para pequeñas y medianas empresas.

* + 1. **Objetivos específicos**
* Realizar el levantamiento de información para definir el diseño de la persistencia de los datos.
* Implementar un módulo que permita gestionar los datos de los usuarios del sistema.
* Implementar un módulo que permita gestionar los datos de los productos de diferentes categorías.
* Implementar un módulo que permita gestionar las facturas y generarlas en formato PDF.
* Implementar un módulo que permita gestionar los datos de los clientes.
* Implementar un módulo que permita consultar a los clientes las facturas por su referencia y realizar su descarga.
  1. **Justificación**

El sistema facturador web propuesto servirá como herramienta para pequeñas y medianas empresas, ya que facilitará la de la información que requiere la realización de una factura. Por ello, brindará organización y evitará el riesgo de perder la información como suele suceder con facturas realizadas en papel.

Así mismo, contribuirá a disminuir el daño ecológico que genera la elaboración de facturas en papel; ya que, al estar computarizada la información, el sistema facturador web será amigable con el medio ambiente. De este modo, reduce costos en gastos papelería y evita el riesgo de cometer errores en el momento de diligenciar una factura.

Al ser una aplicación web, permitirá el acceso a la información en todo momento desde cualquier lugar; posibilitará la gestión de clientes, productos, facturas y usuarios, por lo que será una solución eficiente para las pymes.

* 1. **Alcances y delimitaciones**
     1. **Alcances**

Se desarrollará un sistema facturador web que facilite el proceso de facturar para pequeñas y medianas empresas.

* Un usuario podrá registrar la pyme con los datos, donde debe especificar las categorías de productos que maneja la misma y definir las credenciales del primer usuario administrador.
* El usuario administrador podrá gestionar productos por medio de las categorías que especificó en el registro; sin embargo, podrá tendrá acceso a un módulo adicional para administrar las categorías en el caso de que la pyme requiera vender productos con categorías diferentes a las especificadas.
* Se implementarán módulos para gestionar clientes, productos y usuarios; en los cuales se permitirá consultar los registros por medio de filtro, ordenamiento y paginación en las tablas, exportarlos a Excel, agregar, actualizar sus datos básicos o darlos de baja.
* Se implementará un módulo para gestionar las facturas; en este apartado se permitirá realizar consultas de la información por medio de filtro con ordenamiento y paginación, crear nuevas facturas validando los clientes y productos registrados, exportarlas en formato PDF y anularlas.
* Los clientes podrán realizar la consulta de una factura por medio de la referencia de la misma con exportación en formato PDF; esta funcionalidad se encontrará en la página principal del aplicativo.
* Los usuarios con nivel 0 serán administradores y tendrán acceso a facturas, clientes, productos, usuarios y categorías. Por otro lado, los usuarios con nivel 1 tendrán acceso al módulo de clientes y facturas.
* El sistema facturador ISSMC no tendrá la funcionalidad de pago online para las facturas generadas a los clientes. Además, su desarrollo, diseño y estructura estará orientado a pequeñas y medianas empresas de Colombia. Adicionalmente, el idioma predeterminado será el español.
  + 1. **Delimitaciones**

|  |  |
| --- | --- |
| **Aplicación** | **Descripción** |
| **JAVA** | Lenguaje de programación para el backend |
| **Spring boot** | Herramienta de desarrollo para el backend. |
| **ReactJS** | Librería JavaScript para desarrollo para el frontend. |
| **Visual Studio Code** | Editor de texto para código front. |
| **Eclipse IDE** | Entorno de desarrollo para el backend. |
| **HTML** | Lenguaje marcado para definir el contenido de la aplicación. |
| **CSS** | Lenguaje para definir los estilos del aplicativo |
| **Bootstrap** | Framework de diseño. |
| **MySQL** | Sistema gestor de bases de datos relacionales |
| **Microsoft Office Professional Plus 2019** | Herramientas ofimáticas para documentación y prueba de funcionalidades del aplicativo. |
| **Windows 10** | Sistema operativo. |

*Tabla 1. Delimitación del proyecto.*

* 1. **Marco de referencia**

Como material de apoyo para el desarrollo del sistema facturador web se realizó una investigación de sistemas facturadores que comparten ciertas características con ISSMC como lo son: Facturador online, FACTURAgem y PymeFree.

* + 1. **Fuentes primarias**

A continuación, se referencian las paginas web en las que se consultaron los sistemas facturadores relacionados con la funcionalidad de ISSMC.

* Facturador Online. (2018). [en línea] Consultado: 11 de abril de 2021, Disponible en: <https://facturadoronline.com/>
* Saasmanía. (2021) Saas de la semana. [en línea] Consultado: 29 de agosto de 2021, Disponible en: <http://www.saasmania.com/page/7/?s=%22saas+de+la+semana%22>
* IONOS. (2019). ¿Qué es un sistema de facturación? [en línea] Consultado: 29 de agosto de 2021, Disponible en: <https://www.ionos.mx/startupguide/gestion/sistemas-de-facturacion/>
  1. **Marco teórico**

A continuación, se evidencian los antecedentes de facturas, investigaciones y temáticas que soportan teóricamente el sistema:

* + 1. **Marco histórico**

A lo largo del tiempo, el proceso de facturación ha evolucionado en gran medida; para empezar, en el siglo XIII surge el mercantilismo gracias al auge de la baja Edad Media y a las leyes que imponían estrictos castigos con el fin de proteger la actividad mercantil que realizaban los banqueros, mercaderes y burgueses. Desde ese momento, se impone el concepto de título valor a fin de reemplazar las tradicionales y clásicas prácticas de intercambio en mercados y ferias; por ejemplo, el trueque. [34]

Luego a través de la edad moderna y contemporánea, junto con la evolución de los mercados y del comercio; el pensamiento abstracto del valor sobre el papel fue ajustándose cada vez más a los cambios económicos, comerciales y sociales Puesto que, se veía cada vez más la necesidad de sustentar y demostrar el dinero por medio un documento en el que el cliente y el vendedor estuvieran de acuerdo. [34] Sin embargo, al evidenciarse cada vez más la necesidad de facturar, con el paso del tiempo comenzaron a surgir diferentes problemáticas para las empresas; principalmente se generaron dificultades con el orden, tiempo, costos y errores de escritura. Pues, en muchas ocasiones se extravían las facturas, se diligencian mal los valores produciendo discordias con el cliente y “Hacer las facturas en papel supone un gasto considerable en papel y material de oficina.” [27]

“El desorden es otro de nuestros mayores enemigos cuando hacemos las facturas en papel.” [27]

Debido a la problemática descrita anteriormente, como solución se da la llegada de avances tecnológicos, gracias a ellos se permitió que las empresas emplearan una mejor organización, lo que posibilitó la reducción de costos y la mejora continua conforme con las mejoras tecnológicas. A este tipo de avances se les llamó facturadores online, los cuales brindan organización, son amigables con el medio ambiente, automatizan tiempo que puede ser empleado en otras tareas y permiten el acceso a la información en cualquier momento desde cualquier lugar. [27]

“La organización y rapidez que nos va a proporcionar una facturación online no la vamos a encontrar nunca en el papel.” [27]

A continuación, se describen los sistemas investigados que comparten una relación estrecha en cuanto a características implementadas en el sistema facturador desarrollado:

* **Facturador online**

Se basa en un sistema en línea que permite gestionar las facturas en una empresa, permitiendo visualizarlas en formato PDF y enviarlas por correo electrónico a los clientes que la compañía registre; así mismo se tiene la posibilidad de tener un control sobre los productos que se manejan. Es un software privado y creado en México, por lo que solo permite el manejo de valores en pesos mexicanos. [1]

* **FACTURAgem**

Aplicación web privada orientada para pymes y emprendedores, cuenta con una interfaz intuitiva y sencilla de manejar. Permite consultar facturas que se encuentren pendientes de pago, crearlas, modificarlas, eliminarlas y ver el histórico de operaciones. Aun así, es de uso limitado; puesto que, las primeras 10 facturas son gratis y no permite gestionar los productos que la pyme o emprendedor comercializa; por ello, el proceso de facturación puede consumir mayor tiempo. [2]

* **PymeFree**

Es una aplicación gratuita de escritorio diseñada para pymes de México, permite la creación de facturas y la gestión de clientes de forma automática, por lo que ofrece un mejor rendimiento y una agradable experiencia de usuario. Sin embargo, al ser una aplicación de escritorio tiene acceso limitado, puesto que su descarga e instalación depende del sistema operativo del ordenador y en este caso, PymeFree solo se encuentra disponible para Windows 7,8 y 10. [3]

Como parte fundamental para el desarrollo del proyecto, a continuación, se describen brevemente las temáticas que soportan teóricamente el sistema propuesto.

* + 1. **Aplicaciones web**

Son herramientas que se codifican en un determinado lenguaje, el cual es soportado y ejecutado por diferentes navegadores; es decir, se pueden usar accediendo a cierto servidor web. Ofrecen diferentes ventajas como lo son: practicidad, independencia del sistema operativo, facilidad para actualizar y mantenimiento [4] por lo que ocupan menos requerimientos de memoria; no crean problemas técnicos para el usuario, permitiendo el uso de las mismas en un tiempo concurrente por diferentes clientes.

Las aplicaciones web brindan diferentes beneficios, entre ellos: el trabajo a distancia se puede realizar con mucha facilidad, ofrecen total disponibilidad pues para trabajar sobre ellas solo es necesario un dispositivo con un navegador instalado y conexión a internet, ya que se pueden acceder a ellas en cualquier momento y lugar. Por último, las aplicaciones web permiten centralizar todas las áreas de trabajo. [20]

* + - 1. **Características generales de las aplicaciones web** [20]
* Son prácticas, al permitir una comunicación fluida y dinámica gracias a que manejan una arquitectura Cliente-Servidor.
* Posibilita su actualización y mantenimiento por medio de internet sin requerir de instalaciones, sistemas o softwares específicos.
* Brinda al usuario el acceso a los datos de manera interactiva, respondiendo a las peticiones que se requieran.
* Generan dinámicamente una serie de páginas en formato estándar, como lo es HTML o XHTML, los cuales son soportados por la mayoría de navegadores web.
* Para el desarrollo de las aplicaciones web, se utilizan lenguajes que están diseñados del lado del cliente; por ejemplo,JavaScript con el fin de crear elementos dinámicos para la interfaz del usuario.

Las aplicaciones web están compuestas por tres componentes fundamentales como lo son:

* + - 1. **Componentes de las aplicaciones web** [4]
         1. **Base de datos**

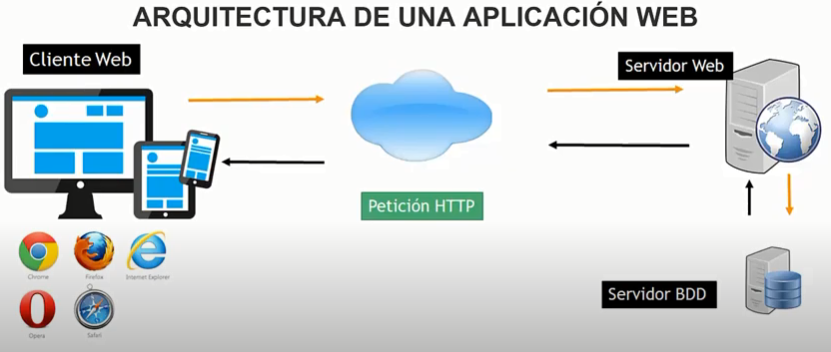
En esta se almacena toda la información correspondiente a la aplicación desarrollada; es decir, contiene la información que interactúa con la misma, como lo son: datos, usuarios, permisos, contenidos, entre otros.

* + - * 1. **Código de compilación**

Se basa en todo el desarrollo de la aplicación web, por medio de este se logra el funcionamiento deseado y así mismo permite la interacción con la base de datos. Este puede almacenarse en un servidor en la nube o local.

* + - * 1. **Acceso a través del navegador**

Contiene la información correspondiente a las URL, las cuales permiten el acceso de los usuarios finales a la aplicación a través de los diferentes navegadores web.



*Figura 1. Arquitectura de una aplicación web [35]*

* + 1. **Sistema facturador web**

Un sistema de facturación online es un software informático que ayuda a optimizar diferentes tareas, las cuales pueden realizarse de manera manual, como lo son gestionar y controlar el envío y recepción de facturas de los servicios que presta una empresa, los clientes, los usuarios y los productos. [5]

* + - 1. **Características de un sistema facturador web [5]**
         1. **Gestión de facturas**

Permite crear facturas a partir de una lista de clientes registrados, actualizar, anular y consultar, eliminando los procedimientos manuales y automatizando los procesos.

* **Beneficios de la facturación web** [21]
* Reducción de costos por facturación.
* Evita errores en la creación y entrega de facturas.
* Simplificación de procesos administrativos.
* Brinda un mayor control.
* Ayuda a preservar el medio ambiente, ya que disminuye el uso del papel.
* Brinda organización, al tener disponible la información en todo momento.
* Disminuye el riesgo de pérdida de la información,
* Las facturas pueden ser consultadas desde cualquier navegador web.
* Ofrece un mejor servicio al cliente.
  + - * 1. **Pagos en línea**

Con los softwares de facturación web, los clientes tienen la posibilidad de efectuar los pagos de manera online de sus facturas directamente a la empresa. Solo basta con conocer el número de factura a cancelar para darle la posibilidad al cliente de efectuar el pago de la misma.

* + - * 1. **Gestión de clientes**

Proporciona la administración de clientes registrados y no registrados, para así llevar un buen manejo y evitar datos erróneos en la creación de facturas.

* + - * 1. **Gestión de personal**

Son una tecnología que permite la interacción entre máquinas conectadas en red; dicha interacción se basa en el envío de solicitudes y respuestas entre un cliente y un servidor, las cuales incluyen datos. Para la comunicación de dichos datos, se usan lenguajes escritos en archivos de texto plano como lo son XML y JSON. [6]

* + 1. **Servicios web**

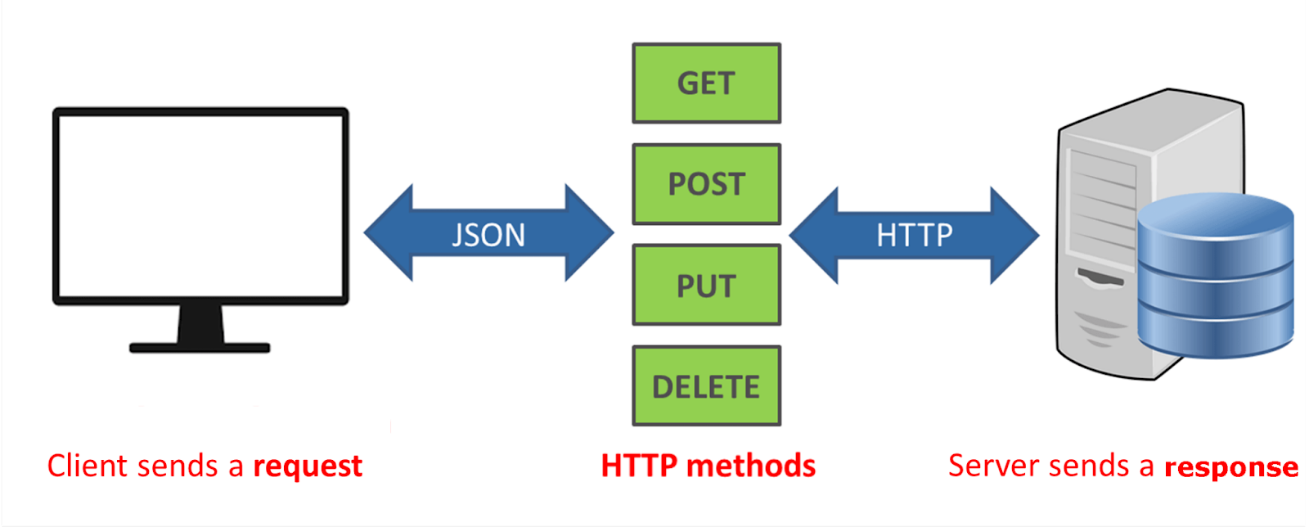
Son una tecnología que permite la interacción entre máquinas conectadas en red; dicha interacción se basa en el envío de solicitudes y respuestas entre un cliente y un servidor, las cuales incluyen datos. Para la comunicación de dichos datos, se usan lenguajes escritos en archivos de texto plano como lo son XML y JSON. [6]

* + - 1. **REST**

REST (Representational State Transfer) se basa en un estilo de arquitectura que se usa para realizar la comunicación entre cliente y servidor. Las solicitudes se suelen hacer a través de un protocolo de transferencia de hipertexto, denominado HTTP; una vez recibida la solicitud, tiene la capacidad de enviar la respuesta en distintos mensajes; sin embargo, el formato de preferencia es JSON. Es por ello, que las API de Restful son más flexibles y se pueden configurar con mayor facilidad. [7]

* + - * 1. **Principios en los que se basa REST** [22]
* Usa los métodos HTTP: Emplea cuatro métodos específicos que cumplen con una funcionalidad determinada: en primera parte, el método GET se utiliza para obtener un recurso desde el servidor; es decir, por medio de este se accede a una determinada información. El método POST es empleado para crear un nuevo recurso; PUT realiza actualizaciones a la información y por último, el método DELETE elimina un recurso del servidor. [8]
* No tiene estado: Es stateless, cada petición HTTP contiene toda la información para ser ejecutada, lo que quiere decir que ni el cliente ni el servidor requiera recordar de un estado para realizarla o ejecutarla.
* Manejo de URIs: La URI es el único elemento que contiene la información para acceder a un servicio de la aplicación REST, ya que por medio de esta se nos facilita el uso de las funcionalidades que ofrece el sistema desarrollado.
* Sistema de capas: Maneja una arquitectura jerárquica entre los componentes que conforman la API.
  + - * 1. **Ventajas de REST para el desarrollo** [22]
* Separación entre cliente y servidor: Gracias a esto, se mejora la portabilidad de la interfaz del usuario a otras plataformas, aumenta la escalabilidad de los proyectos y permite la independencia del desarrollo en los componentes.
* Fiabilidad y escalabilidad: Se puede escalar la aplicación sin inconvenientes, migrar a otros servidores, realizar cambios sobre la base de datos y demás. Por lo tanto, la separación de cliente y servidor permite tener el front y el back en distintos servidores, para así trabajar con aplicaciones más flexibles.
* Es independiente: Es adaptable a cualquier tipo de sintaxis independientemente de la plataforma en la que se esté desarrollando.

**ARQUITECTURA DE UNA API REST**

*Figura 2. Arquitectura de una API REST [36]*

* + 1. **Herramientas de desarrollo**

Para el desarrollo del sistema facturador web, se van a emplear diferentes herramientas tecnológicas como lo son:

* + - 1. **Java**

Java es un lenguaje de programación orientado a objetos, diseñado para crear aplicaciones seguras y potentes. Entre sus características, se encuentra que es un lenguaje pequeño y práctico, ya que cuenta con las herramientas necesarias para un desarrollo eficiente. Además, es de propósito general es decir que las funciones que ofrecen los demás lenguajes, también las brinda Java; es independiente de la plataforma, por lo tanto, se puede ejecutar en cualquier sistema operativo y equipo. Finalmente, ofrece funcionalidades por medio de la inserción de bibliotecas. [9]

Existen variedad de programas desarrollados en Java que habitualmente usamos, ya que este es uno de los lenguajes de programación más populares y usados en proyectos de gran tamaño en la actualidad. Entre sus usos, se encuentran: aplicaciones de educación, entretenimiento, comunicación, entre otras. [24]

**Ventajas de desarrollar en Java [24]**

* Lenguaje simple: No es un lenguaje complejo de aprender, por lo que rápidamente se puede familiarizar con su sintaxis.
* Lenguaje orientado a objetos: Por medio de los objetos se permite encapsular información, clases y funciones.
* Lenguaje interpretado y compilado: Su compilación es tan eficiente que se asimila con la de un lenguaje ensamblador, es decir, desde la base puede ser interpretado. Esto beneficia a la ejecución de aplicaciones, ya que esta se puede realizar en cualquier lugar.
* Seguro: Ofrece seguridad máxima en las aplicaciones desarrolladas en él, por lo cual brinda comodidad para el usuario.
  + - 1. **Spring boot**

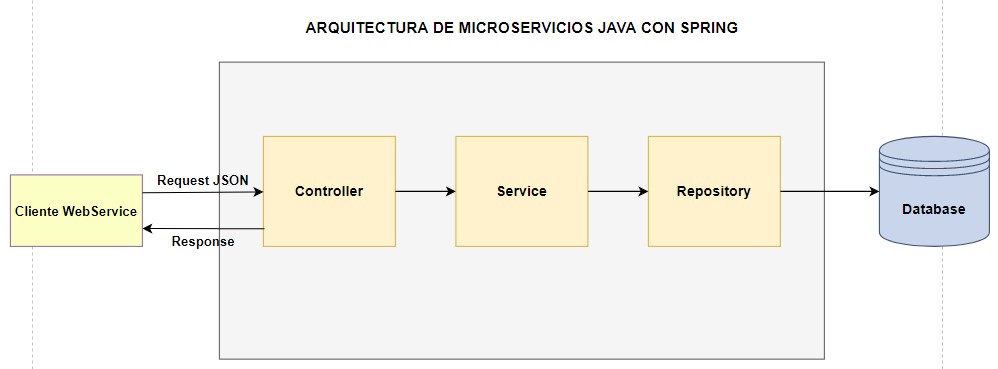
Se basa en una herramienta tecnológica creada por Spring Framework, que facilita la creación de aplicaciones independientes para varios lenguajes, entre ellos JAVA. Se basa en el uso de anotaciones, que brindan configuración del comportamiento en la aplicación. Para su uso, es necesario crear un proyecto Maven/Gradle y allí agregar todas las dependencias necesarias para el uso de las funcionalidades que ofrece esta herramienta. [10]

Es un framework basado en el Modelo,Vista,Controlador, mediante el cual se facilita el desarrollo de servicios REST, por medio del manejo de componentes y librerías que brinda. [8]

**Características de Spring boot [23]**

* Contiene servidores embebidos, como lo son: Tomcat, Jetty y Undertow.
* No genera código, y así mismo, no requiere configuración XML.
* Permite configurar bibliotecas automáticamente, ya sean de Spring o de otras tecnologías.
* Brinda soporte para la automatización con Maven o Gradle.

A continuación, se evidencia la arquitectura de microservicios de JAVA con Spring Boot.



*Figura 3. Arquitectura de microservicios JAVA con Spring Boot*

* + - 1. **JavaScript**

Se basa en un lenguaje de programación ligero del lado del cliente, el cual es muy importante ya que es el único lenguaje de programación que entienden los navegadores, es por esto que con JavaScript se desarrollan funcionalidades del frontend en las aplicaciones web modernas. Es un lenguaje levemente tipado, en el cual están basadas populares librerías como jQuery o ReactJS. [11]

* + - * 1. **Características de JavaScript** [26]

**Tipos de datos**

Los distintos tipos de datos podrán tener valores que pueden ser reconocidos por el lenguaje de la computadora, donde un programador puede dar instrucciones mediante JavaScript; el cual se reconoce hasta seis tipos de valores distintos, como lo son: números, cadenas, booleanos, objetos, nulos e indefinidos.

**Operadores**

Estos elementos permiten realizar operaciones con los datos que se deseen operar, dichos operadores se clasifican en operadores aritméticos, lógicos y de comparación.

**Comentarios**

JavaScript posibilita documentar el código con comentarios por medio de // o /\*\*/, estos comentarios se suelen usar para entender mejor el código y en caso de una modificación futura sea más fácil realizarla.

**Uso de un archivo externo**

JavaScript permite utilizar un archivo que haya sido escrito por separado, este archivo debe contener la extensión .js donde este contendrá todo el código de JavaScript.

* + - * 1. **Librería ReactJS**

Es una librería basada en JavaScript que maneja isomorfismo, es decir que interactúa tanto del lado del cliente como del servidor. Se basa en la composición de varios componentes; los cuales pueden ser con estado (statefull) que se manejan con funciones, o sin estado (stateless) que se escriben por medio de clases. Además, permite pasar funciones como parámetros entre componentes y ofrece rendimiento gracias a que cuenta con un DOM virtual en el que se ejecutan las aplicaciones desarrolladas. [12]

* + - 1. **HTML**

HTML es un lenguaje marcado de hipertexto, el cual permite definir el contenido de una página web. Ofrece diferentes funcionalidades que se utilizan por medio de etiquetas y atributos propios del lenguaje con la sintaxis <etiqueta></etiqueta>; la estructura básica de un documento se compone de cuatro etiquetas, como lo son: HTML que es el elemento en donde se abarca el contenido, head hace referencia a la cabecera, body es el cuerpo y por último footer es el pie de la página a desarrollar. [13]

* + - 1. **CSS**

Lenguaje de marcas enfocado para definir, crear y mejorar la presentación de elementos escritos en un lenguaje marcado como lo es HTML. Funciona por medio de propiedades y selectores, que posibilitan localizar los sectores y personalizarlos con un diseño específico. [14]

* + - 1. **Bootstrap**

Framework de diseño que permite crear interfaces para una página o aplicación web con JavaScript y CSS; facilita la creación de sitios responsive design, es decir que un documento puede adaptarse fácilmente a cualquier tamaño de dispositivo en el que se acceda al mismo. Ofrece diferentes elementos con estilos predefinidos, como lo son botones, menús desplegables, formularios, entre otros. [15]

* + - 1. **MySQL**

Gestor de bases de datos relacional de código abierto; se caracteriza por ser flexible, fácil de manejar, seguro y cuenta con un alto rendimiento gracias a que maneja una arquitectura de cliente-servidor, lo que permite que dichas partes se comuniquen entre sí. [16]

**Ventajas del usar MySQL** [25]

A continuación, se mencionan algunas de las ventajas que brinda el uso de MySQL:

* Permite la creación de todo tipo de aplicaciones.
* Cuenta con una alta capacidad de soporte técnico.
* Tiene alta capacidad para manipular un gran volumen de datos, MySQL puede manejar más de cincuenta millones de registros.
* Provee alta seguridad para la información almacenada.
* Al ser de código abierto, se permite ser libre de usarlo y modificarlo.
* Su distribución es gratuita.
* Permite la realización de todo tipo de consultas, las cuales son respondidas rápidamente.
  + - 1. **Eclipse IDE**

Plataforma de desarrollo de proyectos de código abierto. Proporciona herramientas para la gestión de espacios de trabajo, escribir, desplegar, ejecutar y depurar aplicaciones; basado en perspectivas, soporta diversos lenguajes de programación y permite el uso de aplicaciones cliente. [17]

* + - 1. **Visual Studio Code**

Se basa en un editor de código fuente que proporciona facilidad para la escritura y la administración del código. Se maneja por medio de esquematización, es decir que permite expandir y contraer diferentes bloques de código; además enmarca por medio de colores definidos la sintaxis de un lenguaje en específico, marca errores y advertencias con agilidad, maneja seguimiento de cambios, entre otros. [18]

* 1. **Marco conceptual**
     1. **Pyme**

Pyme hace referencia a pequeñas y medianas empresas, las cuales suelen contar con un bajo número de trabajadores. Son pequeñas organizaciones con ánimo de lucro que tienen operaciones de baja escala. [28]

* + 1. **Cliente**

“Es una persona física o jurídica que recibe un servicio o adquiere un bien a cambio de un dinero u otro tipo de retribución.” [29]

* + 1. **Producto**

Se define como el conjunto de características que pueden ser tangibles en cuanto a forma, tamaño, color, textura; o intangibles en cuanto a servicios. Es decir, algo que se ofrece en el mercado para su adquisición que tiene determinadas características y que el cliente acepta para satisfacer sus necesidades. [30]

* + 1. **Factura**

“Una factura es un documento comercial que registra la información relativa a la compra o venta de un bien o servicio.” [31]

Una factura debe contener determinada información; como lo es, el número de factura, fecha de emisión, datos de la empresa y cliente, descripción de los productos (cantidad, nombre, valor unitario, valor total), impuestos de la factura y el valor total de la misma. [31]

* + 1. **Usuario**

Un usuario es un individuo que hace uso de determinado servicio al cual tiene acceso. Existe una clasificación de usuarios, la cual se divide en grupos como lo son: invitados, usuarios registrados y administradores. [32]

* + 1. **Gestión de la información**

“La gestión de la información se puede identificar como la disciplina que se encargaría de todo lo relacionado con la obtención de la información adecuada, en la forma correcta, para la persona indicada, al coste adecuado, en el momento oportuno, en el lugar apropiado y articulando todas estas operaciones para el desarrollo de una acción correcta.” [33]

* 1. **Factibilidad**

A continuación, se describen los factores que contribuirán en el desarrollo del proyecto planteado, en aspectos técnicos, operativos y legales.

* + 1. **Factibilidad técnica**

Para lograr un óptimo desarrollo del sistema facturador web, se requerirá de un computador que cuente con adecuadas características que soporten las herramientas de desarrollo descritas anteriormente.

|  |  |
| --- | --- |
| **Nombre de herramienta** | **Descripción** |
| **Eclipse IDE for Enterprise Java and Web Developers** | Plataforma de desarrollo de código abierto multiplataforma para el desarrollo de aplicaciones cliente. |
| **Visual Studio Code** | Editor de código compatible con varios lenguajes de programación. |
| **MySQL** | Gestor de bases de datos relacionales. |

*Tabla 2. Herramientas que se usarán para el desarrollo del proyecto.*

|  |  |
| --- | --- |
| **Recurso físico** | **Características** |
| **Computador** | Computador de escritorio que cuente con las siguientes especificaciones:  Procesador: Intel Corei7 Memoria RAM: 12 GB Disco duro: 500 SSD Sistema operativo: Windows 10 Herramienta ofimática: Microsoft Office Professional Plus 2019 |

*Tabla 3. Recursos físicos para el desarrollo del proyecto.*

|  |  |
| --- | --- |
| **Recurso físico** | **Características** |
| **Desarrollador** | Estudiante de Tecnología en Sistematización de datos, con conocimientos en: servicios web REST, lenguaje orientado a objetos JAVA, framework de desarrollo Spring Boot, lenguaje de programación ligero JavaScript, librería ReactJS, HTML, CSS, Bootstrap y sistema gestor de bases de datos MySQL. |
| **Tutor** | Ingeniero de sistemas con bastantes conocimientos en el desarrollo de aplicaciones web con servicios web REST. |

*Tabla 4. Recursos humanos para el desarrollo del proyecto.*

* + 1. **Factibilidad operativa**

El sistema facturador web que se desarrollará, estará enfocado para pequeñas y medianas empresas con el fin de realizar procesos de facturación, además será una herramienta organizacional para las mismas. Brindará orden y optimización de tiempo, puesto que una vez el administrador haya registrado la empresa, procede a registrar los productos que manejan y posteriormente, los clientes activos de la misma; y así, se posibilitará la gestión de la información junto con otras funcionalidades que se ofrecerán, en cuanto a consultas, ediciones, eliminaciones y claramente, nuevos registros.

Con ayuda de la tecnología de Servicios web REST, se obtendrá una mayor eficiencia del aplicativo; gracias a que consume menos recursos del servidor y así brindará una mayor optimización de tiempo al ejecutar las funcionalidades que el usuario requiera.

Una parte importante del aplicativo, estará basada en el módulo de registro de nuevas facturas; puesto que al momento de facturar se digita el nombre del cliente usando filtros y por medio de una rápida búsqueda, se autocompletarán los campos correspondientes al mismo. Así mismo, funcionará con los productos a facturar, se podrá filtrar su búsqueda e ir agregando rápidamente especificando la cantidad del producto; se establecerá si la factura incluye IVA y automáticamente se sumará o restará el valor correspondiente, el campo de valor en letras será dinámico con el campo de valor total.

* + 1. **Factibilidad legal**

Para el desarrollo del aplicativo propuesto, se utilizará la herramienta Spring boot que se ejecuta bajo la licencia gratis Apache 2.0; en cuanto al entorno de desarrollo para aplicaciones Eclipse, se usará bajo Eclipse Public License que también es una licencia libre. Para el sistema gestor de bases de datos MySQL se usará Licencia Community, que es gratuita y está distribuida bajo la licencia pública general GNU; por último, Visual Studio Code se distribuye mediante la licencia MIT y se ejecuta con una licencia gratuita.

Por lo anterior, no deberían generarse problemas de derechos de autor en el momento de hacer uso de las licencias anteriormente mencionadas; ya que son gratuitas y libres.

En términos de protección de datos, la información de la empresa estará almacenada en MySQL; donde se encontrará protegida y a la cual solo se puede acceder por medio de las credenciales de los usuarios registrados según el nivel de permisos de los mismos; sus contraseñas pasarán por dos fases de encriptación: la primera que corresponde a SHA-256 , luego se protegerá con cifrado AES. Por lo tanto, se preserva la seguridad de las empresas registradas.

* + 1. **Factibilidad económica**
       1. **Presupuesto y financiación**

La factibilidad económica del proyecto es alta; puesto que para lograr un óptimo desarrollo, en términos financieros se requiere de asesorías por parte del tutor, un equipo de trabajo, servicio de energía e internet,papelería y demás gastos varios que pueden presentarse.

Por ello, se ha realizado una proyección por medio de tablas en donde se describe la factibilidad económica, identificando costos en diferentes aspectos como lo son: recursos humanos, recursos técnicos, otros recursos y un porcentaje estimado para imprevistos.

* + - 1. **Recursos humanos**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tipo** | **Descripción** | **Valor de hora** | **Cantidad** | **Total** |
| **Tutor** | Asesorías y orientación para la realización del proyecto | $ 50.000 | 64 horas en 4 meses | $ 3.200.000 |
| **Desarrollador** | Un programador que elabore el desarrollo y el diseño del aplicativo. | $ 45.000 | 8 horas por semana | $ 5.760.000 |
| **Total de Recursos Humanos** | | | | $ 8.960.000 |

*Tabla 5. Presupuesto en recursos humanos.*

* + - 1. **Recursos tecnológicos**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Recurso** | **Descripción** | **Valor Unitario** | **Cantidad** | **Total** |
| **Computador (Servidor - clientes)** | Equipo de escritorio para el desarrollo y las pruebas del sistema; el cual debe contar con ciertas características técnicas como lo son: Procesador IntelCorei7, DD 500GB y RAM 12GB. | $ 2’699.000 | 1 | $ 2’699.000 |
| **Total de Recursos Tecnológicos** | | | | $ 2’699.000 |

*Tabla 6. Presupuesto en recursos tecnológicos.*

* + - 1. **Recursos varios**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Recurso** | **Descripción** | **Valor Unitario** | **Cantidad** | **Total** |
| **Internet** | Servicio de internet de 30 megas. | $ 120.000 | 4 meses | $ 480.000 |
| **Energía** | Servicio de energía eléctrica | $ 60.000 | 4 meses | $ 240.000 |
| **Papelería** | Materiales de papelería para realizar el diseño y el análisis del sistema. | $ 50.000 | 1 block,  6 lapices,  3 esferos,  1 carpeta,  1 paq, de Post-its,  1 cuaderno, 1 paq. de marcadores | $ 50.000 |
| **Total de Recursos Varios** | | | | $ 770.000 |

*Tabla 7. Presupuesto en recursos varios.*

|  |  |
| --- | --- |
| **Recurso** | **Valor** |
| **Total Recursos Humanos** | $  8.960.000 |
| **Total Recursos Técnicos** | $  2’699.000 |
| **Total Otros recursos** | $     770.000 |
| **Costos imprevistos (10%)** | $  1.242.900 |
| **TOTAL COSTO** | **$13.671.900** |

* + - 1. **Costo total**

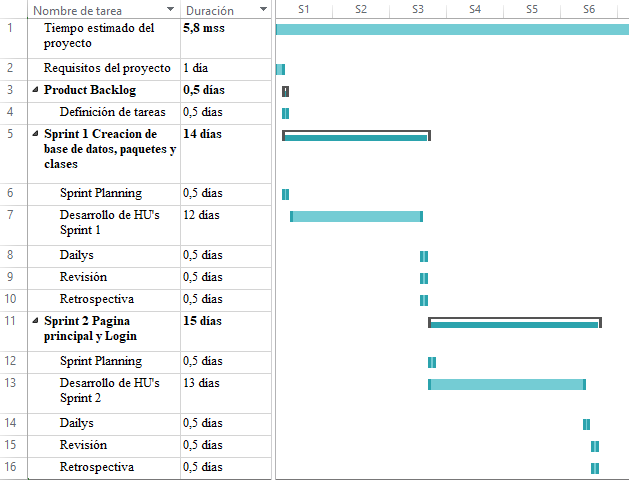
*Tabla 8. Costo total del presupuesto para el proyecto.*

* 1. **Metodología** [19]

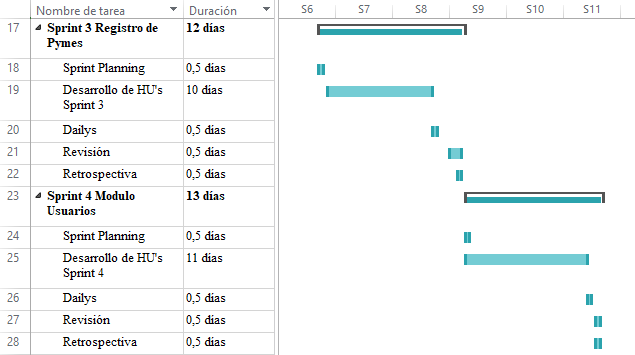
Con el fin de lograr un desarrollo eficiente para el sistema facturador web, se establece como metodología de desarrollo a Scrum; ya que es ágil, eficaz, de fácil entendimiento y permite el trabajo en equipo.

Inicia con el Product Backlog, donde en este se definen las tareas a realizar según los requisitos que se identificaron para el proyecto a desarrollar.

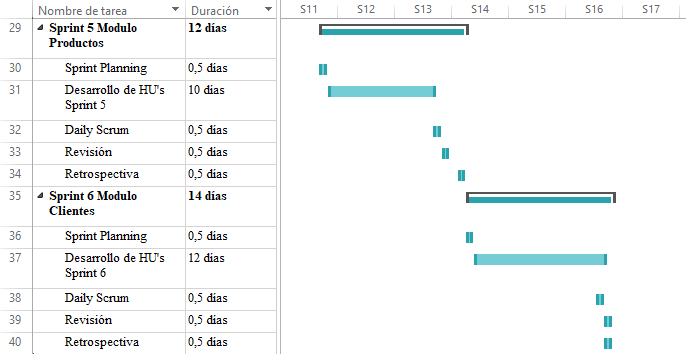
* Definición de componentes de un sprint
  + Planning: En esta fase se planean, estructuran, organizan y se planean y plantean las tareas que se ejecutarán en el sprint según el estado del backlog.
  + Execution: Ejecución de las tareas planeadas en la fase anterior. Tiene una duración de 1 a 4 semanas.
  + Dailys: Se basa en una reunión diaria, donde en esta se tratarán tres puntos esenciales como lo son: tareas que se lograron realizar el día anterior, tareas que se realizarán para ese día y obstáculos que están impidiendo avanzar en el desarrollo de las mismas.
  + Review: En esta fase se realiza una revisión del trabajo efectuado y el alcance del mismo según las tareas que se comprometieron en la planning.
  + Retrospective: En esta fase, se analiza la forma de trabajo y se expone lo observado en las fases anteriores; esto con el fin de buscar una mejoría y lograr de manera eficiente los objetivos propuestos.
  1. **Cronograma de actividades**

**

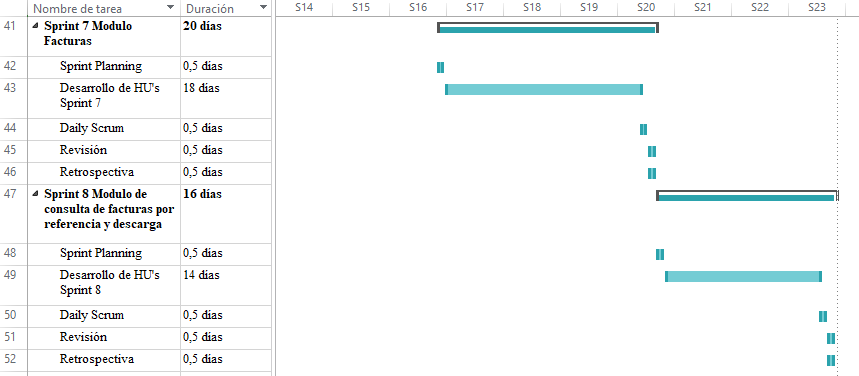
*Figura 4. Cronograma de actividades, parte 1.*



*Figura 5. Cronograma de actividades, parte 2.*



*Figura 6. Cronograma de actividades, parte 3.*



*Figura 7. Cronograma de actividades, parte 4.*

1. **Desarrollo**
   1. **Product backlog**

A continuación, se evidencia la lista priorizada y ordenada de los sprints que abarcarán el desarrollo del sistema facturador web ISSMC.

|  |  |  |
| --- | --- | --- |
| **No. Sprint** | **Nombre** | **Descripción** |
| 1 | Creación de base de datos, paquetes y clases. | Se realizará el levantamiento de información definiendo el diseño de la persistencia de datos. Se crea el proyecto del backend con Java - Spring boot; definiendo las dependencias, entidades con JPA, controladores, repositorios, mappers, modelos y servicios. Se crea proyecto en react, se separan carpetas según los módulos a realizar y se instalan las dependencias necesarias. |
| 2 | Página principal y Login | Se realizará la página principal del sistema facturador web; incluyendo un apartado para darle acceso a la página del Login construido por un formulario donde se validarán las credenciales del usuario para darle acceso a las funcionalidades del menú en el sistema. |
| 3 | Registro de Pymes | En el formulario del login se tendrá un link para permitir el registro de nuevas pymes; el cual redirigirá a un formulario que solicita el logo, los datos básicos, categorías y primer usuario administrador; incluyendo validaciones de datos dinámicas. |
| 4 | Módulo Usuarios | Se permitirán las funcionalidades para registrar, actualizar, consultar, exportar a Excel y dar de baja a los usuarios de la pyme. |
| 5 | Módulo Productos | Se permitirán las funcionalidades para registrar, actualizar, consultar, exportar a Excel y dar sacar del mercado los productos de la pyme. |
| 6 | Módulo Clientes | Se permitirán las funcionalidades para registrar, actualizar, consultar, exportar a Excel y dar de baja a los clientes de la pyme. |
| 7 | Módulo Facturas | Se permitirá realizar nuevas facturas con funcionamiento y comportamiento dinámico. Además, se podrán consultar y desde la tabla de resultados se podrá anular y exportar. |
| 8 | Módulo de consulta de facturas por referencia y descarga | Se implementará un campo en la página principal del ISSMC para permitir la consulta de facturas a los clientes por medio de la referencia y en caso de que exista redireccionará al visualizador de PDF para permitir la descarga de la misma. |

*Tabla 9. Sprints del proyecto.*

* + 1. **Sprint backlog**

Partiendo del producto backlog realizado anteriormente, se realizó un tasking de las historias de usuario que se consideran necesarias para cumplir a cabalidad de manera satisfactoria los objetivos propuestos.

* + - 1. **Historias de usuario**

A continuación, se presentan las historias que se identificaron para el desarrollo de ISSMC; cada una de ellas se compone de un id donde su primer digito es el numero del sprint al que pertenece seguido del nombre, cuentan con el rol, funcionalidad, resultado, estado y prioridad.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Id** | **Rol** | **Funcionalidad** | **Resultado esperado** | **Estado** | **Prioridad** |
| 101 - Persistencia de datos | Equipo de desarrollo | Brindar una persistencia de datos adecuada. | Diseño de la persistencia de datos y creación de base de datos en MySQL. | **Realizado** | **Alta** |
| 102 - Creación de proyecto. | Equipo de desarrollo | Maquetación de carpetas y archivos repositorios de backend y frontend. | Desacoplamiento de carpetas y clases, componentes del proyecto e instalación de dependencias. | **Realizado** | **Alta** |
| 201 – Creación de página principal | Usuario particular | Visualizar página principal de ISSMC. | Un usuario podrá visitar la página principal. | **Realizado** | **Alta** |
| 202 – Creación de Login | Administrador / Usuario básico | Permitir acceso a las funcionalidades del sistema validando pyme, credenciales y permisos. | Los usuarios registrados podrán ingresar al sistema y tener acceso al menú para manipular la información de la pyme a la que pertenecen. | **Realizado** | **Alta** |
| 303 – Registro de pymes | Usuario particular | Permitir el registro de pymes. | Un usuario podrá registrar una pyme, proporcionando los datos básicos, logo, primer administrador y categorías. | **Realizado** | **Alta** |
| 401 – Registro de nuevos usuarios | Administrador | Permitir el registro de nuevos usuarios. | Un administrador podrá registrar la información de un usuario para la pyme a la que pertenece. | **Realizado** | **Alta** |
| 402 – Actualizar o eliminar usuarios | Administrador | Permitir la actualización de datos o eliminación de usuarios. | Un administrador podrá actualizar o eliminar la información a un usuario de la pyme. | **Realizado** | **Alta** |
| 403 – Consultar usuarios | Administrador | Permitir el registro de nuevos usuarios. | Un administrador podrá consultar la información de los usuarios pertenecientes a la pyme y exportarlos a Excel. | **Realizado** | **Alta** |
| 501 – Registro de nuevos productos | Administrador | Permitir el registro de nuevos productos. | Un administrador podrá registrar la información de un producto para la pyme a la que pertenece. | **Realizado** | **Alta** |
| 502 – Actualizar o eliminar productos | Administrador | Permitir la actualización de datos o eliminación de productos. | Un administrador podrá actualizar o eliminar la información a un producto de la pyme. | **Realizado** | **Alta** |
| 503 – Consultar productos | Administrador | Permitir el registro de nuevos productos. | Un administrador podrá consultar la información de los productos pertenecientes a la pyme y exportarlos a Excel. | **Realizado** | **Alta** |
| 601 – Registro de nuevos clientes | Administrador / Usuario básico | Permitir el registro de nuevos clientes. | Un administrador o usuario básico podrá registrar la información de un cliente para la pyme a la que pertenece. | **Realizado** | **Alta** |
| 602 – Actualizar o eliminar clientes | Administrador / Usuario básico | Permitir la actualización de datos o la eliminación de clientes. | Un administrador o usuario básico podrá actualizar o eliminar la información a un cliente de la pyme. | **Realizado** | **Alta** |
| 603 – Consultar clientes | Administrador / Usuario básico | Permitir el registro de nuevos clientes. | Un administrador o usuario básico podrá consultar la información de los clientes pertenecientes a la pyme y exportarlos a Excel. | **Realizado** | **Alta** |
| 701 – Crear facturas | Administrador / Usuario básico | Permitir el registro de facturas. | Un administrador o usuario básico podrá crear facturas para la pyme. | **Realizado** | **Alta** |
| 702 – Consultar y anular facturas | Administrador / Usuario básico | Permitir la consultar, anulación y exportación en PDF de facturación. | Un administrador o usuario básico podrá consultar, anular y exportar facturas a formato PDF. | **Realizado** | **Alta** |
| 801 – Consultar factura por referencia con exportación a PDF | Usuario particular | Permitir la consulta de una factura desde el home. | Un usuario particular podrá consultar una factura por medio de la referencia desde la página principal y exportarla a PDF. | **Realizado** | **Alta** |

*Tabla 10. Historias de usuario.*

* 1. **Sprints**

En la metodología Scrum, un sprint hace referencia a un intervalo de tiempo que se emplea por el equipo de desarrollo para efectuar las tareas definidas en el producto backlog y en las historias de usuario. Se divide en 5 etapas, las cuales son: planning, execution, dailys, review y retrospective.

* + 1. **Sprint 1: Creación de base de datos, paquetes y clases.**

Tiene como objetivo principal definir el diseño de la persistencia de datos y crear los proyectos de backend y frontend con las dependencias necesarias.

* + - 1. **Planning**

Se seleccionan las siguientes historias de usuario correspondientes a este Sprint:

|  |  |  |
| --- | --- | --- |
| **Id Historia** | **Nombre** | **Actor** |
| 101 | Persistencia de datos | Equipo de desarrollo |
| 102 | Creación de proyecto | Equipo de desarrollo |

*Tabla 11.* *Planning Sprint 1.*

* + - 1. **Execution**

Se definen los criterios de aceptación para el Sprint 1, aquí se relaciona la historia de usuario, el número, la descripción y el tiempo que duró la implementación de cada uno de ellos:

|  |  |  |  |
| --- | --- | --- | --- |
| **Id Historia** | **Numero de tarea** | **Descripción tarea** | **Tiempo** |
| **101** | 1 | Diseño e implementación de la base de datos a usar en MySQL. | 72 horas |
| 2 | Diccionario de datos | 24 horas |
| **102** | 3 | Creación y estructura de proyecto backend | 96 horas |
| 4 | Implementación de dependencias backend | 24 horas |
| 5 | Creación y estructura de proyecto frontend | 48 horas |
| 6 | Implementación de dependencias frontend | 24 horas |

*Tabla 12. Tareas Sprint 1.*

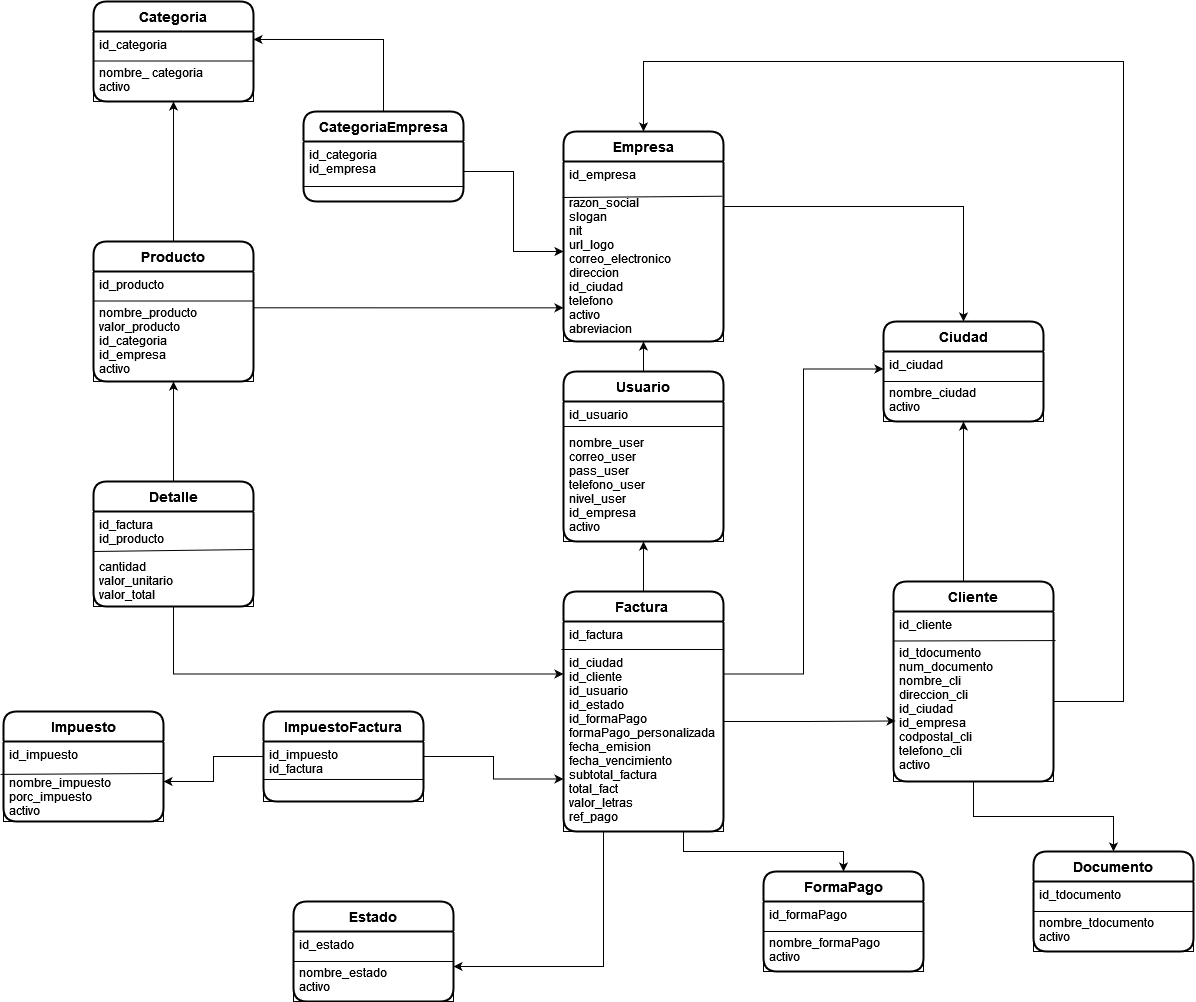
* + - 1. **Dailys**

Se evidenciaron grandes avances y pocos impedimentos en las tareas desarrolladas durante el sprint.

* + - 1. **Review**

En esta etapa se realiza la entrega de las tareas comprometidas en las historias de usuario 101 y 102, de las cuales su resultado fue:

* Se realizó el levantamiento de información con el fin de realizar el diseño de la persistencia de los datos (modelo relacional) teniendo en cuenta reglas de normalización. (*Ver Figura 8. Modelo relacional de base de datos*)
* Se realizó el diccionario de datos y la creación del script de la base de datos según el modelo relacional diseñado en la tarea anterior. (*Ver desde Tabla 13. Diccionario de datos – Ciudad hasta Tabla 26. Diccionario de datos – Categoría Empresa*)
* Se crea el proyecto backend de Java usando Sprint Boot, se divide en 11 carpetas que contienen clases e interfaces con una responsabilidad en especifica; como lo son entidades creadas con ORM JPA, controladores, servicios, modelos, repositorios y mappers. (*Ver Figura 9. Estructura proyecto backend*)
* Se implementan dependencias del backend en archivo pom.xml como lo son: Sprint boot con JPA, conector MySQL para la base de datos, herramientas de mapeo, librería lombok para evitar realizar constructores, getters y setters en modelos. (*Ver Figura 10.* *Dependencias backend*)
* Se crea proyecto frontend con librería React de JavaScript y se implementa estructura desacoplada con 4 carpetas para: componentes, estilos, rutas e imágenes. (*Ver Figura 11. Estructura proyecto frontend*)
* A través del uso del sistema de gestión de paquetes de Node.js (npm) se instalaron dependencias necesarias para el desarrollo de ISSMC, las cuales se evidencias en archivo package.son. (*Ver* *Figura 12. Dependencias frontend*)



*Figura 8. Modelo relacional de base de datos*

A continuación, se evidencia el diccionario de datos para las tablas identificadas en el modelo anteriormente presentado.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ciudad** | | | | | |
| Guarda las ciudades de clientes, empresas y facturas | | | | | |
| **Nombre** | **Tipo** | **PK** | **FK** | **NULL** | **Descripción** |
| id\_ciudad | INT AUTO\_INCREMENT | **X** |  |  | Numero único que define la ciudad |
| nombre\_ciudad | VARCHAR (150) |  |  |  | Nombre de la ciudad asociada al id\_ciudad |
| activo | BIT |  |  |  | Borrado lógico de la ciudad relacionada al id\_ciudad |

*Tabla 13. Diccionario de datos - Ciudad.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Empresa** | | | | | |
| Almacena las pymes registradas en el sistema | | | | | |
| **Nombre** | **Tipo** | **PK** | **FK** | **NULL** | **Descripción** |
| id\_empresa | INT AUTO\_INCREMENT | **X** |  |  | Numero único que define la empresa |
| razon\_social | VARCHAR (100) |  |  |  | Razón social de la pyme asociada al id\_empresa |
| slogan | VARCHAR (250) |  |  | **X** | Slogan de la pyme |
| nit | VARCHAR (20) |  |  |  | Define la identificación de la pyme |
| url\_logo | VARCHAR (350) |  |  |  | Ruta donde se guardará el logo de la pyme. |
| correo\_electronico | VARCHAR (100) |  |  |  | Define el correo electrónico asociado al id\_empresa |
| direccion | INT |  |  |  | Define la dirección asociada al id\_empresa |
| id\_ciudad | INT |  | **X** |  | Define la ciudad de la pyme |
| telefono | VARCHAR (20) |  |  |  | Define el teléfono de la pyme |
| activo | BIT |  |  |  | Borrado lógico de la empresa relacionada al id\_empresa |
| abreviacion | VARCHAR (10) |  |  |  | Abreviación única de la razón social para facturación. |

*Tabla 14. Diccionario de datos - Empresa.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Usuario** | | | | | |
| Almacena los usuarios registrados según su pyme | | | | | |
| **Nombre** | **Tipo** | **PK** | **FK** | **NULL** | **Descripción** |
| id\_usuario | INT AUTO\_INCREMENT | **X** |  |  | Numero único que define el usuario |
| nombre\_user | VARCHAR (300) |  |  |  | Nombre del usuario asociado al id\_usuario |
| correo\_user | VARCHAR (100) |  |  |  | Correo del usuario, parte de las credenciales para tener acceso al sistema |
| pass\_user | VARCHAR (200) |  |  |  | Contraseña del usuario, parte de las credenciales para tener acceso al sistema |
| telefono\_user | VARCHAR (30) |  |  |  | Teléfono del usuario asociado al id\_usuario |
| nivel\_user | INT |  |  |  | Define el tipo de permiso que tendrá el usuario, ya sea 1 o 0. |
| id\_empresa | INT |  | **X** |  | Define la empresa a la que pertenece el usuario |
| activo | BIT |  |  |  | Borrado lógico del usuario relacionado al id\_usuario |

*Tabla 15. Diccionario de datos - Usuario.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Categoria** | | | | | |
| Almacena las categorías a las que pertenecen los productos | | | | | |
| **Nombre** | **Tipo** | **PK** | **FK** | **NULL** | **Descripción** |
| id\_categoria | INT AUTO\_INCREMENT | **X** |  |  | Numero único que define la categoria |
| nombre\_categoria | VARCHAR (60) |  |  |  | Nombre de la ciudad asociada al id\_categoria |
| activo | BIT |  |  |  | Borrado lógico de la categoria relacionada al id\_categoria |

*Tabla 16. Diccionario de datos - Categoría.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Impuesto** | | | | | |
| Almacena los impuestos aplicados a las facturas | | | | | |
| **Nombre** | **Tipo** | **PK** | **FK** | **NULL** | **Descripción** |
| id\_impuesto | INT AUTO\_INCREMENT | **X** |  |  | Numero único que define el impuesto |
| nombre\_impuesto | VARCHAR (30) |  |  |  | Nombre del impuesto asociado al id\_impuesto |
| porc\_impuesto | DOUBLE |  |  |  | Define el porcentaje del impuesto asociado al id\_impuesto |
| activo | BIT |  |  |  | Borrado lógico del impuesto relacionado al id\_impuesto |

*Tabla 17. Diccionario de datos - Impuesto.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Estado** | | | | | |
| Almacena los tipos de estados que pueden tener las facturas | | | | | |
| **Nombre** | **Tipo** | **PK** | **FK** | **NULL** | **Descripción** |
| id\_estado | INT AUTO\_INCREMENT | **X** |  |  | Numero único que define el estado |
| nombre\_estado | VARCHAR (50) |  |  |  | Nombre del estado asociado al id\_estado |
| activo | BIT |  |  |  | Borrado lógico del estado relacionado al id\_estado |

*Tabla 18. Diccionario de datos - Estado.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **FormaPago** | | | | | |
| Almacena los tipos de forma de pago usados cotidianamente en una factura. | | | | | |
| **Nombre** | **Tipo** | **PK** | **FK** | **NULL** | **Descripción** |
| id\_formapago | INT AUTO\_INCREMENT | **X** |  |  | Numero único que define la forma de pago. |
| nombre\_formapago | VARCHAR (80) |  |  |  | Nombre de la forma de pago asociado al id\_formapago |
| activo | BIT |  |  |  | Borrado lógico de la forma de pago relacionado al id\_formapago |

*Tabla 19. Diccionario de datos – Forma Pago.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Documento** | | | | | |
| Almacena los tipos de documentos que pueden tener los clientes | | | | | |
| **Nombre** | **Tipo** | **PK** | **FK** | **NULL** | **Descripción** |
| id\_tdocumento | INT AUTO\_INCREMENT | **X** |  |  | Numero único que define el tipo de documento. |
| nombre\_tdocumento | VARCHAR (80) |  |  |  | Nombre del tipo de documento asociado al id\_tdocumento |
| activo | BIT |  |  |  | Borrado lógico del tipo de documento relacionado al id\_tdocumento |

*Tabla 20. Diccionario de datos – Documento.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Producto** | | | | | |
| Almacena los productos que pertenecen a cada pyme | | | | | |
| **Nombre** | **Tipo** | **PK** | **FK** | **NULL** | **Descripción** |
| id\_producto | INT AUTO\_INCREMENT | **X** |  |  | Numero único que define el producto |
| nombre producto | VARCHAR (300) |  |  |  | Nombre del producto asociado al id\_producto |
| valor\_producto | DOUBLE |  |  |  | Define el valor del producto |
| id\_categoria | INT |  | **X** |  | Define la categoria del producto |
| id\_empresa | INT |  | **X** |  | Define la pyme a la que pertenece el producto |
| activo | BIT |  |  |  | Borrado lógico del producto relacionado al id\_producto |

*Tabla 21. Diccionario de datos – Producto.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Cliente** | | | | | |
| Almacena los clientes | | | | | |
| **Nombre** | **Tipo** | **PK** | **FK** | **NULL** | **Descripción** |
| id\_cliente | INT AUTO\_INCREMENT | **X** |  |  | Numero único que define el cliente |
| id\_tdocumento | INT |  | **X** |  | Define el tipo de documento que tiene el cliente |
| num\_documento | VARCHAR (20) |  |  |  | Numero de documento del cliente |
| nombre\_cli | VARCHAR (100) |  |  |  | Nombre del cliente asociado al id\_cliente |
| direccion\_cli | VARCHAR (100) |  |  |  | Dirección del cliente |
| id\_ciudad | INT |  | **X** |  | Define la ciudad a la que pertenece el cliente |
| id\_empresa | INT |  | **X** |  | Define la pyme a la que pertenece el cliente |
| codpostal\_cli | VARCHAR (10) |  |  |  | Código postal del cliente |
| telefono\_cli | VARCHAR (30) |  |  |  | Teléfono de contacto del cliente |
| activo | BIT |  |  |  | Borrado lógico del cliente relacionado al id\_cliente |

*Tabla 22. Diccionario de datos – Cliente.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Factura** | | | | | |
| Almacena las facturas realizadas. | | | | | |
| **Nombre** | **Tipo** | **PK** | **FK** | **NULL** | **Descripción** |
| id\_factura | INT AUTO\_INCREMENT | **X** |  |  | Numero único que define la factura. |
| id\_ciudad | INT |  | **X** |  | Define la ciudad de la factura. |
| id\_cliente | INT |  | **X** |  | Define el cliente de la factura. |
| id\_usuario | INT |  | **X** |  | Define el usuario que realiza la factura. |
| id\_estado | INT |  | **X** |  | Define el estado de la factura. |
| id\_formapago | INT |  | **X** | **X** | Define la forma de pago de la factura. |
| formapago\_personalizada | VARCHAR (100) |  |  | **X** | Define la forma personalizada de pago de la factura. |
| fecha\_emision | DATE |  |  |  | Fecha en la que se realiza la factura |
| fecha\_vencimiento | DATE |  |  |  | Fecha de vencimiento de la factura |
| subtotal\_factura | DOUBLE |  |  |  | Suma de los productos facturados sin impuesto |
| total\_fact | DOUBLE |  |  |  | Suma de los productos facturados incluyendo impuestos |
| valor\_letras | VARCHAR (300) |  |  |  | Valor en letras del total |
| ref\_pago | VARCHAR (300) |  |  |  | Referencia de la factura (Dato único) |

*Tabla 23. Diccionario de datos – Factura.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ImpuestoFactura** | | | | | |
| Relaciona los impuestos con las facturas | | | | | |
| **Nombre** | **Tipo** | **PK** | **FK** | **NULL** | **Descripción** |
| id\_impuesto | INT | **X** | **X** |  | Numero único identifica el impuesto. |
| id\_factura | INT | **X** | **X** |  | Numero único identifica la factura. |

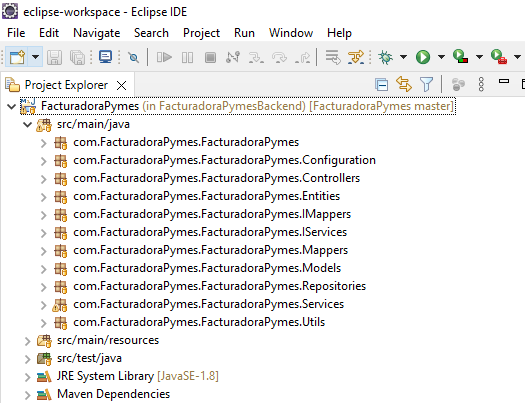
*Tabla 24. Diccionario de datos - Impuesto Factura.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Detalle** | | | | | |
| Almacena los productos a facturados en una factura | | | | | |
| **Nombre** | **Tipo** | **PK** | **FK** | **NULL** | **Descripción** |
| id\_factura | INT | **X** | **X** |  | Define la factura del detalle |
| id\_producto | INT | **X** | **X** |  | Define el producto del detalle |
| cantidad | INT |  |  |  | Cantidad de productos |
| valor\_unitario | DOUBLE |  |  |  | Valor unitario del producto |
| valor\_total | DOUBLE |  |  |  | Valor total multiplicado la cantidad por el valor unitario. |

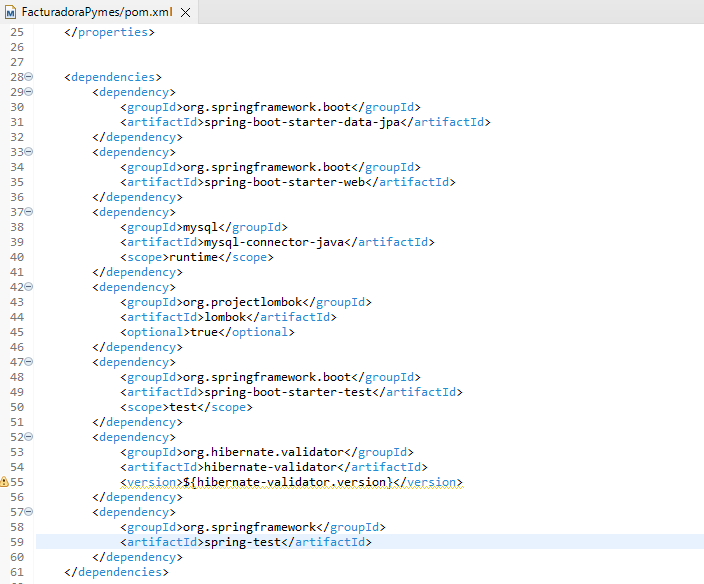
*Tabla 25. Diccionario de datos - Detalle.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **CategoriaEmpresa** | | | | | |
| Relaciona las categorías de los productos que maneja cada pyme registrada | | | | | |
| **Nombre** | **Tipo** | **PK** | **FK** | **NULL** | **Descripción** |
| id\_categoria | INT | **X** | **X** |  | Numero único identifica la categoria. |
| id\_empresa | INT | **X** | **X** |  | Numero único identifica la pyme. |

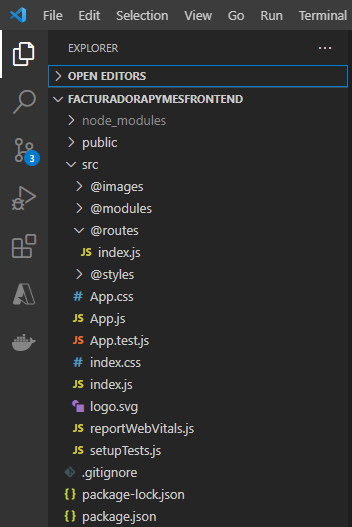
*Tabla 26.* *Diccionario de datos – Categoría Empresa.*



*Figura 9.* *Estructura proyecto backend*



*Figura 10. Dependencias backend*



*Figura 12. Dependencias frontend*

*Figura 11.* *Estructura proyecto frontend*

* + - 1. **Retrospective**

Se evidenció un buen desempeño por parte del equipo de desarrollo debido a que se lograron cumplir las tareas comprometidas en su totalidad del Sprint 1; por ello, se concluye el Sprint.

* + 1. **Sprint 2: Página principal y Login**
       1. **Planning**
       2. **Execution**
       3. **Dailys**
       4. **Review**
       5. **Retrospective**
    2. **Sprint 3: Registro de Pymes**
       1. **Planning**
       2. **Execution**
       3. **Dailys**
       4. **Review**
       5. **Retrospective**
    3. **Sprint 4: Módulo Usuarios**
       1. **Planning**
       2. **Execution**
       3. **Dailys**
       4. **Review**
       5. **Retrospective**
    4. **Sprint 5: Módulo Productos**
       1. **Planning**
       2. **Execution**
       3. **Dailys**
       4. **Review**
       5. **Retrospective**
    5. **Sprint 6: Módulo Clientes**
       1. **Planning**
       2. **Execution**
       3. **Dailys**
       4. **Review**
       5. **Retrospective**
    6. **Sprint 7: Módulo Facturas**
       1. **Planning**
       2. **Execution**
       3. **Dailys**
       4. **Review**
       5. **Retrospective**
    7. **Sprint 8: Módulo de consulta de facturas por referencia y descarga**
       1. **Planning**
       2. **Execution**
       3. **Dailys**
       4. **Review**
       5. **Retrospective**