

“Implementación de una aplicación web para la recepción de pimienta”

INSTITUTO TECNOLÓGICO SUPERIOR DE TEZIUTLÁN

Residencia Profesional

ALUMNO (A):

Manuel Francisco Peña

**Licenciatura en:**

Ingeniería en Sistemas Computacionales.

**Número de Control:**

**18TE0158**



ASESOR (((((((((((((((A9(A):

Heriberto Hernández Rodríguez

**Teziutlán, Puebla; diciembre de 2022**

“La Juventud de hoy, Tecnología del Mañana”

# PRELIMINARES

## Agradecimientos

**A mi Familia**

Por estar en mis días difíciles, ver por mi salud y mi felicidad,  
quererme siempre por encima de cualquier discusión, por el   
gran esfuerzo que hacen para apoyarme a lograr formarme   
en todas mis fases estudiantiles, por estar presente en mi camino.

**A mis Amigos, Colegas y Profesores**

Martín González, Nicolas Moreno, Iván Córdova, y todos   
aquellos que a lo largo de este proyecto de vida me apoyaron   
a no quedarme atrás, los llevare siempre en mi mente y corazón.   
Con especial mención a el maestro Heriberto Hernández Rodríguez,   
que me ha apoyado a lo largo de mi carrera universitaria.

**A querida madre**

Por siempre apoyarme a lo largo de mi vida, ser

mi motivo para nunca rendirme y sobre todo siempre ver

por mí y estar ahí siempre que lo necesite.

## Resumen

El presente documento tiene como finalidad la realización de una aplicación web para la recepción de pimienta en la empresa Asociaciones Agroindustriales Serranas S.A de C.V.

Los sistemas web están alojados en un servidor de Internet o intranet (red local/privada), lo que significa que son independientes de estar instalados en una plataforma o sistema operativo específico. Se parece mucho a los sitios web que solemos ver, pero en realidad los sistemas web van un paso más allá porque son muy potentes y pueden dar respuesta a situaciones muy concretas.

Para el desarrollo de la aplicación web se consideró una plataforma en la cual los colaboradores de la empresa Asociaciones Agroindustriales Serranas S.A de C.V. puedan crear, consultar, editar y eliminar información.

Mencionando las principales herramientas:

* Node Js 18.
* NPM 8.18.0.
* React JS 16.8.
* Axios 0.27.
* Express Node JS.
* Boostrap.
* Prisma ORM.

## Introducción

Este documento presenta una propuesta de creación de una aplicación web para la administración de los procesos durante recepción de pimienta de la empresa Asociaciones Agroindustriales Serranas S.A de C.V. Incluye capítulos VIII en los cuales se detalla los procesos necesarios en búsqueda de cumplir con el objetivo general y los objetivos específicos.

Inicialmente se enuncian los preliminares del proyecto. En el capítulo I se señala el por qué se determinó elaborar una aplicación web, los datos generales de la empresa, los objetivos que se plantearon al iniciar el proyecto, así como los alcances y limitaciones para realizarlo. El capítulo II nos muestra la investigación realizada que dio a conocer la importancia que tiene realizar el proyecto. Dentro del capítulo III se presentan los procedimientos y descripción de las actividades realizadas, módulos del proyecto, los casos de uso y el diagrama de la base de datos utilizada. En el capítulo IV se presenta la propuesta de maquetado del sistema web con base a la investigación previamente realizada en el capítulo II, además de evidencia de la implementación del sistema web, por último, contiene la interpretación de los datos recabados tras implementar el instrumento de investigación seleccionado. El capítulo V hace referencia a las conclusiones que se hicieron de acuerdo a lo observado durante la realización del proyecto. En el capítulo VI se describen las competencias desarrolladas al realizar el proyecto, mismas que son un diagnóstico para conocer las fortalezas y áreas de oportunidad que como egresado de un tecnológico se posee y que serán la base del aprendizaje permanente. El capítulo VII hace mención a las fuentes consultadas para la realización del trabajo y, por último, el capítulo VIII presenta los anexos que se integraron al proyecto.

**Índice general**

[PRELIMINARES 2](#_Toc115200890)

[Agradecimientos 3](#_Toc115200891)

[Resumen 4](#_Toc115200892)

[Introducción 5](#_Toc115200893)

[CAPITULO I GENERALIDADES DE ASOCIACIONES AGROINDUSTRIALES SERRANAS S.A DE C.V. 9](#_Toc115200894)

[1.1 Descripción de la empresa 10](#_Toc115200895)

[1.1.1 Antecedentes 10](#_Toc115200896)

[1.1.2 Misión 11](#_Toc115200897)

[1.1.3 Visión 11](#_Toc115200898)

[1.1.4 Objetivos filosóficos 11](#_Toc115200899)

[1.1.5 Objetivos estratégicos 11](#_Toc115200900)

[1.1.6 Política de calidad 12](#_Toc115200901)

[1.1.7 Política de responsabilidad social 12](#_Toc115200902)

[1.2 Problemática a resolver 12](#_Toc115200903)

[1.3 Pregunta de investigación 12](#_Toc115200904)

[1.4 Objetivos 13](#_Toc115200905)

[1.4.1 Objetivo general 13](#_Toc115200906)

[1.4.2 Objetivos específicos 13](#_Toc115200907)

[1.5 Justificación 13](#_Toc115200908)

[CAPITULO I MARCO TEÓRICO 14](#_Toc115200909)

[2.1 Fundamentos teóricos 15](#_Toc115200910)

[2.1.1 Estado del arte 15](#_Toc115200911)

[2.2 Tecnologías 15](#_Toc115200912)

[2.2.1 Node JS 18 15](#_Toc115200913)

[2.2.2 NPM 8.18.0 18](#_Toc115200914)

[2.2.3 React JS 16.8 19](#_Toc115200915)

[2.2.4 Axios 0.27 20](#_Toc115200916)

[2.2.5 Express Node JS 20](#_Toc115200917)

[2.2.6 Bootstrap 22](#_Toc115200918)

[2.2.7 Prisma ORM 25](#_Toc115200919)

[CAPITULO II DESARROLLO Y METODOLOGÍA 26](#_Toc115200920)

[3.1 Procedimiento y descripción de actividades realizadas. 27](#_Toc115200921)

[3.1.1 Descripción del proyecto 27](#_Toc115200922)

[3.1.2 Cronograma de actividades 29](#_Toc115200923)

[3.1.3 Infraestructura de la empresa 30](#_Toc115200924)

[3.1.4 Infraestructura mínima deseada de la empresa 30](#_Toc115200925)

[3.2 Alcances y enfoque de investigación. 31](#_Toc115200926)

[3.2.1 Alcances 31](#_Toc115200927)

[3.2.2 Enfoque de la investigación 31](#_Toc115200928)

[3.2.3 Limitaciones de la aplicación 31](#_Toc115200929)

[3.2.4 Características 32](#_Toc115200930)

[3.2.5 Ventajas 32](#_Toc115200931)

[3.3 Hipótesis 32](#_Toc115200932)

[3.4 Diseño y metodología de la investigación 32](#_Toc115200933)

[3.4.1 Herramientas de apoyo 32](#_Toc115200934)

[3.4.2 Fases 32](#_Toc115200935)

[3.4.3 Selección de muestra 32](#_Toc115200936)

[3.4.4 Recolección de datos 33](#_Toc115200937)

[3.5 Desarrollo de la metodología 33](#_Toc115200938)

[3.5.1 Requerimientos funcionales 33](#_Toc115200939)

[3.5.2 Diagramas de caso de usos 36](#_Toc115200940)

[3.5.3 Casos de uso 36](#_Toc115200941)

[3.5.4 Diagrama de navegación 42](#_Toc115200942)

[3.5.5 Diagrama de la base de datos 42](#_Toc115200943)

**Índice de ilustraciones**

[Ilustración 1 Tiempo de vida Node JS 15](#_Toc115200944)

**Índice de tablas**

[Tabla 1 Versiones de Node JS 16](#_Toc115200945)

[Tabla 2 Descripción de la aplicación 27](#_Toc115200946)

[Tabla 3 Cronograma de actividades 29](#_Toc115200947)

[Tabla 4 REQ-01 Inicio de sesión 33](#_Toc115200948)

[Tabla 5 REQ-02 Panel principal 34](#_Toc115200949)

[Tabla 6 REQ-03 Nuevo reporte de inspección 34](#_Toc115200950)

[Tabla 7 REQ-04 Creación de comprobante de entrada 34](#_Toc115200951)

[Tabla 8 REQ-05 Historial de entradas de pimienta 35](#_Toc115200952)

[Tabla 9 REQ-06 Creación de orden de compra 35](#_Toc115200953)

[Tabla 8 REQ-07 Historial de órdenes de compra 35](#_Toc115200954)

[Tabla 9 CU-01 Inicio de sesión 36](#_Toc115200955)

[Tabla 10 CU-02 Panel principal 37](#_Toc115200956)

[Tabla 11 CU-03 Nuevo reporte de inspección 38](#_Toc115200957)

[Tabla 12 CU-04 Creación de comprobante de entrada 39](#_Toc115200958)

[Tabla 13 CU-05 Historial de entradas de pimienta 39](#_Toc115200959)

[Tabla 14 CU-06 Creación de orden de compras 40](#_Toc115200960)

[Tabla 15 CU-07 Historial de órdenes de compra 41](#_Toc115200961)

# CAPITULO I GENERALIDADES DE ASOCIACIONES AGROINDUSTRIALES SERRANAS S.A DE C.V.

## 1.1 Descripción de la empresa

### 1.1.1 Antecedentes

Asociaciones Agroindustriales Serranas S.A de C.V. surge como una opción de servicio de comercialización a productores agropecuarios, inicia operaciones el 25 de junio de 1994 únicamente con café y pimienta; más adelante en el año de 1999 incursiona exitosamente en el cacao.

Fue fundada en el municipio de Tlapacoyan Veracruz siendo un punto estratégico para la comercialización de café y pimienta gorda. Al situarse en este municipio ha superado un elemento de innovación en la forma de hacer en el sector agrario. Sin dudarlo su filosofía industrial los ha convertido en líderes indiscutibles dentro del sector, buscando organizar y capacitar a los productores con el propositito de establecer un programa de desarrollo de proveedores que les permita mejorar los procesos de manejo poscosecha y obtener prejuicios justos para sus productos. La empresa reconoce y valora la calidad.

La comercialización de mayores volúmenes de café, pimienta gorda y cacao se logró con el establecimiento de dos sucursales; el de Comalcalco, Tabasco en 1999, donde se recolectaba y procesaba cacao y pimienta gorda; y otro en 2004 en Huixtla, Chiapas, donde se recolectaba café. Asimismo, existe un almacén de productos de café terminado y una planta-almacén de procesamiento de pimienta gorda en Perote, Veracruz.

En el otoño de 2016, Asociaciones Agroindustriales Serranas SA de CV. Recibió el Premio al Mérito Exportador de la Región Sur de México dentro de la categoría de pequeña y mediana empresa agrícola (PYME) del Consejo Mexicano de Comercio Exterior. El Premio Nacional de Exportación es el máximo reconocimiento anual al esfuerzo, tendencia, eficiencia, talento, dinamismo y liderazgo otorgado en aquellas empresas que han logrado convertirse en sinónimo de conectividad global, posicionando la oferta exportable de México, convirtiéndose en uno de los motores de la economía.

En el transcurso de estos exitosos 25 años han logrado consolidar un exitoso equipo de trabajadores multidisciplinarios con compromiso con los valores de la empresa los permitan seguir logrando su misión de ser los proveedores por excelencia de café mexicano en grano, pimienta gorda y cacao.

### 1.1.2 Misión

Garantizar un buen manejo de los productos que se adquieren en la empresa organizándolos en una base de datos, para tener un registro detallado, así como de la recolección, venta y uso de los productos. Al mismo tiempo se mostrará en bloques la calidad de los productos y la zona que se obtienen para dichas referencias en la adquisición de dichos productos.

### 1.1.3 Visión

Mejorar la consulta y registro de datos en el menor tiempo posible, teniendo la opción de poder sacar registros fácilmente sin la necesidad de buscar al personal especificado. Se realiza con responsabilidad y excelencia, con la discreción y seguridad de los datos con los que se cuentan.

### 1.1.4 Objetivos filosóficos

Manipulación de datos, registro y seguridad de los mismos.

### 1.1.5 Objetivos estratégicos

Acceso a los datos de la empresa desde cualquier parte de la misma.

### 1.1.6 Política de calidad

Nos comprometemos a cuidad los datos que se ingresen en nuestras bases de datos para no comprometer los datos personales de los compradores o vendedores de la misma, así mismo los datos de los trabajadores del plantel.

### 1.1.7 Política de responsabilidad social

Nos encontramos comprometidos con la responsabilidad social a través de:

* Respeto, responsabilidad y principios de ética.
* Cero tolerancias a la corrupción de datos y notificarlo en caso de presentarse.
* Acciones que promuevan un crecimiento social y/o económico para los trabajadores.

## 1.2 Problemática a resolver

Los encargados de la administración de la recepción de pimienta en la empresa Asociaciones Agroindustriales Serranas S.A de C.V, tienen diversos problemas en cuanto a el tiempo de recepción de la pimienta, entre las necesidades se encuentran:

* Mejorar el tiempo de intercambio de información entre los diversos departamentos involucrados.
* Evitar retrasos en el registro de información.
* Reducir el costo de impresión de talonarios para el proceso.

Todos estos problemas presentados anteriormente dieron lugar al planteamiento del desarrollo de una aplicación web que permita optimizar tiempos en la recepción de la pimienta en la empresa a trabajar.

## 1.3 Pregunta de investigación

¿En qué medida el desarrollo de una aplicación Web mejorará tiempo de recepción de pimienta en la empresa Asociaciones Agroindustriales Serranas S.A. de C.V.?

## 1.4 Objetivos

### 1.4.1 Objetivo general

Desarrollar un sistema receptor de información basado en web para automatizar los procesos de recepción de café, pimienta y cacao.

### 1.4.2 Objetivos específicos

* Desarrollar el módulo de recepción de pimienta.
* Desarrollar una base de datos que concentre la información de la pimienta recibida.
* Garantizar la seguridad y persistencia de los datos recolectados.
* Contar con un control adecuado de la pimienta recibida.

## 1.5 Justificación

Actualmente es necesario procesar y almacenar la información de manera eficiente para mejorar los procesos de gestión de apoyos y así obtener un control total sobre estas actividades.

Diseñar y desarrollar una aplicación web que mejore el tiempo de recepción de pimienta, nace de la creciente necesidad de los diferentes departamentos para registrar de manera más ágil y eficaz la información de cada nueva entrada de pimienta a la empresa.

# CAPITULO I MARCO TEÓRICO

## 2.1 Fundamentos teóricos

### 2.1.1 Estado del arte

## 2.2 Tecnologías

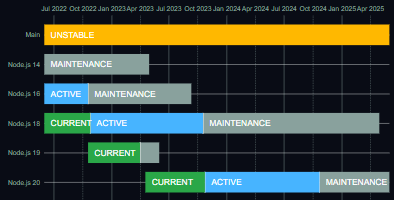
### 2.2.1 Node JS 18

Ideado como un entorno de ejecución de JavaScript orientado a eventos asíncronos, Node.js está diseñado para crear aplicaciones network escalables. Node.js es similar en diseño y está influenciado por sistemas como Event Machine de Ruby y Twisted de Python. Pero Node.js lleva el modelo de eventos un poco más allá. Incluye un bucle de eventos como runtime de ejecución en lugar de una biblioteca.

HTTP es un elemento destacado en Node.js, diseñado teniendo en cuenta la transmisión de operaciones con streaming y baja latencia. Esto hace que Node.js sea muy adecuado para la base de una librería o un framework web.

Las versiones de Node.js ingresan en el estado Current release durante seis meses, lo que proporciona a los autores librerías tiempo para añadir soporte a ella. Después de seis meses, los lanzamientos con números impares (9, 11, etc.) dejan de ser compatibles, y los lanzamientos con números pares (10, 12, etc.) pasan al estado LTS activo y están listos para uso general. El estado de lanzamiento de LTS es "long-term support", de soporte a largo plazo, que generalmente garantiza que los errores críticos se solucionarán por un total de 30 meses. Las aplicaciones de producción solo deben usar versiones Active LTS o Maintenance LTS.

Ilustración 1  
Tiempo de vida Node JS



Fuente: Node JS (2016)

A continuación, se muestran las versiones actuales de Node JS:

Tabla 1  
Versiones de Node JS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **VERSIÓN** | **ESTADO** | **LANZAMIENTO INICIAL** | **INICIO LTS ACTIVO** | **INICIO MANTENIMIENTO LTS** | **FIN DE VIDA** |
| **v14** | Mantenimiento LTS | 21/04/2020 | 27/10/2020 | 19/10/2021 | 30/04/2023 |
| **v16** | LTS Activo | 20/04/2021 | 26/10/2021 | 18/10/2022 | 11/09/2023 |
| **v18** | Actual | 19/04/2022 | 25/10/2022 | 18/10/2023 | 30/04/2025 |
| **v19** | Pendiente | 18/10/2022 |  | 01/04/2023 | 01/06/2023 |
| **v20** | Pendiente | 18/04/2023 | 24/10/2023 | 22/10/2024 | 30/04/2026 |

Fuente: Node JS (2022)

Desde una perspectiva de desarrollo de servidor web, Node tiene un gran número de ventajas:

* Gran rendimiento. Node ha sido diseñado para optimizar el rendimiento y la escalabilidad en aplicaciones web y es un muy buen complemento para muchos problemas comunes de desarrollo web (ej, aplicaciones web en tiempo real).
* El código está escrito en "simple JavaScript", lo que significa que se pierde menos tiempo ocupándose de las "conmutaciones de contexto" entre lenguajes cuando estás escribiendo tanto el código del explorador web como del servidor.
* JavaScript es un lenguaje de programación relativamente nuevo y se beneficia de los avances en diseño de lenguajes cuando se compara con otros lenguajes de servidor web tradicionales (ej, Python, PHP, etc.) Muchos otros lenguajes nuevos y populares se compilan/convierten a JavaScript de manera que puedes también usar CoffeeScript, ClosureScript, Scala, LiveScript, etc.
* El gestor de paquetes de Node (NPM del inglés: Node Packet Manager) proporciona acceso a cientos o miles de paquetes reutilizables. Tiene además la mejor en su clase resolución de dependencias y puede usarse para automatizar la mayor parte de la cadena de herramientas de compilación.
* Es portable, con versiones que funcionan en Microsoft Windows, OS X, Linux, Solaris, FreeBSD, OpenBSD, WebOS, y NonStop OS. Además, está bien soportado por muchos de los proveedores de hospedaje web, que proporcionan infraestructura específica y documentación para hospedaje de sitios Node.
* Tiene un ecosistema y comunidad de desarrolladores de terceros muy activa, con cantidad de gente deseosa de ayudar.
* Puedes crear de forma sencilla un servidor web básico para responder cualquier petición simplemente usando el paquete HTTP de Node.

### 2.2.2 NPM 8.18.0

NPM es parte esencial de Node.js, el entorno de ejecución de javaScript en el lado del servidor basado en el motor V8 de Google. Es muy seguramente, la principal razón del gran éxito de Node permitiendo que cientos de desarrolladores puedan compartir paquetes de software entre distintos proyectos.

NPM responde a las siglas de Node Package Manager o manejador de paquetes de node, es la herramienta por defecto de JavaScript para la tarea de compartir e instalar paquetes.

NPM se compone de al menos dos partes principales:

* Un repositorio online para publicar paquetes de software libre para ser utilizados en proyectos Node.js
* Una herramienta para la terminal (command line utility) para interactuar con dicho repositorio que te ayuda a la instalación de utilidades, manejo de dependencias y la publicación de paquetes.

Es decir, los proyectos basados en Node (actualmente incluye los proyectos de aplicaciones web que utilizan Node para su proceso de compilación y generación de archivos) utilizan la utilidad de linea de comandos (cli) para consumir paquetes desde el repositorio online, un listado gigantesco de soluciones de software para distintos problemas disponibles públicamente en npmjs.com y para manejar dependencias, y para ello necesitas un archivo de configuración que le diga a NPM que este es un proyecto node.

NPM es la solución que ofrece el ecosistema de JavaScript para manejar dependencias, auditar paquetes y mantener un repositorio de paquetes disponible para todos los usuarios, es la herramienta central de todo proyecto JavaScript sea este backend o frontend.

El sistema de NPM se basa en un archivo centralizado que describe los metadatos del proyecto, scripts y dependencias tanto de producción como de desarrollo.

### 2.2.3 React JS 16.8

React usa una sintaxis HTML en JavaScript llamada JSX (JavaScript y XML). Estar familiarizado con HTML y JavaScript ayuda a aprender JSX, y a identificar, en una mejor manera, si los errores en las aplicaciones están relacionados con JavaScript o con el más específico dominio de React.

Como su eslogan oficial señala, React es una biblioteca para construir interfaces de usuario. React no es un framework, ni siquiera se limita a la web. React es utilizado con otras bibliotecas para renderizar en ciertos entornos.

El objetivo principal de React es minimizar los errores que ocurren cuando los desarrolladores construyen interfaces de usuario. Esto lo hace mediante el uso de componentes, piezas de código lógicas y auto-contenidas que describen una parte de la interfaz del usuario. Estos componentes se pueden juntar para crear una interfaz de usuario completa, y React abstrae la mayor parte del trabajo de renderizado, permitiéndote enfocarte en el diseño de la interfaz.

A diferencia de los otros frameworks vistos en este módulo, React no impone reglas estrictas sobre convenciones de código u organización de archivos. Esto permite a los equipos establecer las convenciones que funcionen mejor para ellos y adoptar React de la manera en que deseen. React puede manejar un solo botón, algunas piezas de una interfaz o la interfaz de usuario completa de una aplicación.

Si bien React puede usarse para pequeñas piezas de una interfaz, no resulta tan sencillo "introducirlo" en una aplicación como sería el caso de una biblioteca como jQuery, o incluso de un framework, como Vue, por lo que viene a ser más abordable cuando construimos una aplicación completamente con React.

Además, muchos de los beneficios de la experiencia del desarrollador de una aplicación React, como codificar interfaces con JSX, requieren un proceso de compilación. Agregar un compilador como Babel a un sitio web hace que el código se ejecute de manera lenta, por lo que los desarrolladores a menudo configuran dichas herramientas con un paso de compilación. Podrá decirse que React tiene un gran requisito de herramientas, pero se puede aprender.

### 2.2.4 Axios 0.27

Muchos proyectos en la web deben interactuar con una API REST en algún momento en su desarrollo. Axios es un cliente HTTP ligero basado en el servicio $http en Angular.so v1.x y es similar a la API Fetch nativa de JavaScript. Axios se basa en promesas, lo que le permite aprovechar async y await de JavaScript para obtener un código asíncrono más legible. También puede interceptar y cancelar solicitudes, y hay una protección integrada del lado del cliente contra la falsificación de solicitudes entre sitios.

Algunas de sus principales características son:

* Hace XMLHttpRequests desde el navegador.
* Hace peticiones http desde node.js.
* Soporta el API de Promesa.
* Intercepta petición y respuesta.
* Transforma petición y datos de respuesta.
* Cancela peticiones.
* Transformación automática de datos JSON.
* Soporte para proteger al cliente contra XSRF.

### 2.2.5 Express Node JS

“Es una infraestructura de aplicaciones web Node.js mínima y flexible que proporciona un conjunto sólido de características para las aplicaciones web y móviles” (Express JS, 2022).

Con miles de métodos de programa de utilidad HTTP y middleware a disposición, la creación de una API sólida es rápida y sencilla. Express proporciona una delgada capa de características de aplicación web básicas, que no ocultan las características de Node.js

Proporciona mecanismos para:

* Escritura de manejadores de peticiones con diferentes verbos HTTP en diferentes caminos URL (rutas).
* Integración con motores de renderización de "vistas" para generar respuestas mediante la introducción de datos en plantillas.
* Establecer ajustes de aplicaciones web como qué puerto usar para conectar, y la localización de las plantillas que se utilizan para renderizar la respuesta.
* Añadir procesamiento de peticiones "middleware" adicional en cualquier punto dentro de la tubería de manejo de la petición.

A pesar de que Express es en sí mismo bastante minimalista, los desarrolladores han creado paquetes de middleware compatibles para abordar casi cualquier problema de desarrollo web. Hay librerías para trabajar con cookies, sesiones, inicios de sesión de usuario, parámetros URL, datos POST, cabeceras de seguridad y muchos más.

La popularidad de un framework web es importante porque es un indicador de se continuará manteniendo y qué recursos tienen más probabilidad de estar disponibles en términos de documentación, librerías de extensiones y soporte técnico.

No existe una medida disponible de inmediato y definitiva de la popularidad de los frameworks de lado servidor (aunque sitios como Hot Frameworks intentan asesorar sobre popularidad usando mecanismos como contar para cada plataforma el número de preguntas sobre proyectos en GitHub y StackOverflow). Una pregunta mejor es si Node y Express son lo "suficientemente populares" para evitar los problemas de las plataformas menos populares.

De acuerdo con el número de compañías de perfil alto que usan Express, el número de gente que contribuye al código base, y el número de gente que proporciona soporte tanto libre como pagado.

En sitios web o aplicaciones web dinámicas, que accedan a bases de datos, el servidor espera a recibir peticiones HTTP del navegador (o cliente). Cuando se recibe una petición, la aplicación determina cuál es la acción adecuada correspondiente, de acuerdo a la estructura de la URL y a la información (opcional) indicada en la petición con los métodos POST o GET. Dependiendo de la acción a realizar, puede que se necesite leer o escribir en la base de datos, o realizar otras acciones necesarias para atender la petición correctamente. La aplicación ha de responder al navegador, normalmente, creando una página HTML dinámicamente para él, en la que se muestre la información pedida, usualmente dentro de un elemento específico para este fin, en una plantilla HTML.

Express posee métodos para especificar que función ha de ser llamada dependiendo del verbo HTTP usado en la petición (GET, POST, SET, etc.) y la estructura de la URL ("ruta"). También tiene los métodos para especificar que plantilla ("view") o gestor de visualización utilizar, donde están guardadas las plantillas de HTML que han de usarse y como generar la visualización adecuada para cada caso. El middleware de Express, puede usarse también para añadir funcionalidades para la gestión de cookies, sesiones y usuarios, mediante el uso de parámetros, en los métodos POST/GET. Puede utilizarse además cualquier sistema de trabajo con bases de datos, que sea soportado por Node (Express no especifica ningún método preferido para trabajar con bases de datos).

### 2.2.6 Bootstrap

Bootstrap es un framework CSS utilizado en aplicaciones front-end (es decir, en la pantalla de interfaz con el usuario) para desarrollar aplicaciones que se adaptan a cualquier dispositivo. El propósito del framework es ofrecerle al usuario una experiencia más agradable cuando navega en un sitio. Por esta razón, tiene varios recursos para configurar los estilos de los elementos de la página de una manera simple y eficiente, además de facilitar la construcción de páginas que, al mismo tiempo, están adaptadas para la web y para dispositivos móviles. Lo anterior demuestra por qué es importante conocer una estructura potencial de este tipo.

Bootstrap es un framework CSS desarrollado por Twitter en 2010, para estandarizar las herramientas de la compañía. Inicialmente, se llamó Twitter Blueprint y, un poco más tarde, en 2011, se transformó en código abierto y su nombre cambió para Bootstrap. Desde entonces fue actualizado varias veces y ya se encuentra en la versión 4.4. El framework combina CSS y JavaScript para estilizar los elementos de una página HTML. Permite mucho más que, simplemente, cambiar el color de los botones y los enlaces. Esta es una herramienta que proporciona interactividad en la página, por lo que ofrece una serie de componentes que facilitan la comunicación con el usuario, como menús de navegación, controles de página, barras de progreso y más. Además de todas las características que ofrece el framework, su principal objetivo es permitir la construcción de sitios web responsive para dispositivos móviles. Esto significa que las páginas están diseñadas para funcionar en desktop, tablets y smartphones, de una manera muy simple y organizada.

Bootstrap está constituido por una serie de archivos CSS y JavaScript responsables de asignar características específicas a los elementos de la página. Hay un archivo principal llamado bootstrap.css, que contiene una definición para todos los estilos utilizados. Básicamente, la estructura del framework se compone de dos directorios:

* CSS: contiene los archivos necesarios para la estilización de los elementos y una alternativa al tema original;
* JS: contiene la parte posterior del archivo bootstrap.js (original y minificado), responsable de la ejecución de aplicaciones de estilo que requieren manipulación interactiva.

Para asignarle una característica a un elemento, simplemente debemos informar la clase correspondiente en la propiedad “class” del elemento que será estilizado.

Bootstrap ocupa el segundo lugar como la biblioteca de JavaScript más utilizada, con un 20,6% de los sitios web que utilizan dicha tecnología. A continuación, se enlistan las principales razones por lo cual se usará Bootstrap para el presente proyecto:

* Mobile-first. Bootstrap sigue el concepto de mobile first. Esto significa que la preocupación del framework es, primero, desarrollar una página que funcione perfectamente en dispositivos móviles y luego en el desktop. La ventaja de esta estrategia es la garantía de tener un sitio web al que se pueda acceder desde cualquier dispositivo, lo cual es esencial debido a la cantidad de personas que usan smartphone.
* Estándar visual. Las funciones disponibles en Bootstrap le ofrecen una experiencia muy rica al usuario. Esto se debe a que el estándar visual de estilización sigue las tendencias de diseño utilizadas en este momento. Además, hay numerosos temas de Bootstrap gratuitos o pagos que se pueden descargar de Internet.
* Reutilización de código. Para desarrollar muchas de las características existentes en Bootstrap, sería necesario escribir una gran cantidad de líneas de código lo que provoca un aumento en el tamaño de los archivos y en la cantidad de datos transferidos a la carga de la página. Estos factores pueden dificultar una buena clasificación en Google y toda una estrategia de SEO completa, ya que el tiempo para que la página se cargue es una de las razones por las que el usuario deja de esperar y abandona el sitio. Cuando se usa Bootstrap, simplemente se debe vincular la clase al elemento en el que quieres aplicar el recurso. Por lo tanto, su uso elimina la necesidad de escribir muchas líneas de código y también contribuye a reducir el tamaño de los archivos.
* Comunidad activa. Como framework de código abierto, Bootstrap tiene una comunidad activa de desarrolladores. Además de contribuir con las actualizaciones de las versiones (lo que es muy importante para tener una herramienta siempre actualizada). La comunidad también mantiene al día la documentación, administra un blog con consejos y noticias sobre la herramienta y ayuda en una página en el sitio web de Stack Overflow.

### 2.2.7 Prisma ORM

Prisma es un kit de herramientas utilizado en Back-End (herramienta que se ejecuta en el servidor) destinado a base de datos de código abierto, es un generador de consultas, creaciones, actualizaciones y eliminaciones adaptado a las necesidades de nuestro esquema de base de datos.

Prismas se compone de las siguientes herramientas para el desarrollo:

* Prisma Client: Es el encargado de realizar la conexión hacia la base de datos haciendo el intercambio de datos mas seguro evitando el uso directo de inyecciones SQL.
* Prisma Migrate: Es el encargado de migrar el esquema a la base de datos a través de un modelo definido que contiene los datos declarados creando una base de datos con tablas y relaciones de entidad completamente funcional para el consumo de información.

Prisma ORM está construido por una serie de archivos prisma y SQL responsables de migrar nuestro modelo a la base de datos y poder consumirlo a través de nuestro servicio REST API. El archivo principal llamado schema.prisma, es el archivo que contiene declarado todos nuestros modelos de tablas, básicamente es la estructura de toda nuestra base de datos.

Características principales de Prisma ORM:

* Agilidad en el desarrollo: El uso de Prisma ORM optimiza los tiempos en que el desarrollador diseña una base de datos y ejecuta su implementación.
* Abstracción de base de datos: Al emplear este sistema conseguimos un acceso transparente al sistema de base de datos, pudiendo cambiar en un futuro el motor sin notar efecto sobre nuestro sistema. Esto ese debe que no se emplea una sintaxis para una base de datos concreta para acceder a nuestro modelo, sino la sintaxis propia que nos proporcione el ORM, ya que este será el encargado de traducir para los distintos sistemas de bases de datos.
* Seguridad: Los ORM implementan sistemas para evitar tipos de ataques maliciosos como pueden ser inyecciones SQL, ya que no atacamos directamente a nuestra base de datos con una consulta SQL.

# CAPITULO II DESARROLLO Y METODOLOGÍA

## 3.1 Procedimiento y descripción de actividades realizadas.

### 3.1.1 Descripción del proyecto

La aplicación web es un sistema desarrollado en React utilizando JavaScript como lenguaje de programación. Utiliza una base de datos desarrollada en MySql enlazada a través de Prisma ORM. Esta aplicación tiene como finalidad optimizar el procesamiento de datos que la empresa genera en cada nueva entrada de pimienta.

A continuación, se presenta los módulos que puede manipular los usuarios de la aplicación, describiendo las operaciones que se pueden realizar en cada apartado:

Tabla 2  
Descripción de la aplicación

|  |  |
| --- | --- |
| **USUARIO** | |
| **MODULO** | **DESCRIPCIÓN** |
| Inicio de sesión | El usuario podrá acceder al panel administrativo por medio de sus credenciales generadas por el departamento de sistemas. |
| Panel principal | Es la vista principal de la aplicación la cual mediante una tabla refleja las entradas de pimienta pendientes de ser procesadas, en cada registro pendiente. |
| Nuevo reporte de inspección | El usuario ingresa los valores de la inspección realizada a la pimienta entrante por cliente, por el cual registra que tipo de pimienta se recibe, si es pimienta soplada o sin soplar, además de factores adicionales como humedad, porcentaje de pimienta madura, porcentaje de basura, tamaño del grano, calidad de secado, tipo de secado, olor de la pimienta, características de la pimienta, entre otros. |
| Creación de comprobante de entrada. | el usuario a través de información heredada del módulo de reporte de inspección, el usuario complementa los datos ya obtenidos fijando precio que pagaran por kilogramo, ajustando los porcentajes de descuento de esa entrada de pimienta, todo esto reflejando el recalculo de kilogramos netos y importe total en tiempo real para una mejor experiencia de usuario. |
| Historial de entradas de pimienta | El usuario encontrara todas las entradas de pimienta que ya han sido procesadas (Una vez que el usuario complete el módulo de comprobante de entrada se mueve el registro a este módulo) , en el cual se tendrán filtros por nombre de proveedor y fecha para encontrar datos más rápidos además de poder imprimir “Reporte de inspección, comprobante de entrada y orden de compra de la misma”. |
| Creación de orden de compra | El usuario registra la orden de compra anticipada, en donde el usuario establece el precio a comprar y se estipula los kilos de pimienta al que el usuario se compromete a entregar. |
| Historial de órdenes de compra | El usuario puede visualizar las órdenes de compra que existen, creadas a partir del módulo de “Creación de orden de compra”, en el cual podrá filtrar por fecha y nombre para encontrar un reporte de una manera más rápida además de tener la opción de reimprimir el comprobante que se requiera. |

Fuente: Creación propia (2022)

### 3.1.2 Cronograma de actividades

Tabla 3  
Cronograma de actividades

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ACTIVIDADES** | **FECHA** | | | | | | | | | | | | | | |
| **AGOSTO** | | **SEPTIEMBRE** | | | | **OCTUBRE** | | | | **NOVIEMBRE** | | | | |
| **S1** | **S2** | **S1** | **S2** | **S3** | **S4** | **S1** | **S2** | **S3** | **S4** | | **S1** | **S2** | **S3** | **S4** | |
| Diseño del contenido del temario |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| Establecimiento del marco teórico |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| Conformación del primer capítulo |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| Interpretación de resultados |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| Conformación del segundo capítulo |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| Correcciones y sugerencias |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| Conformación del tercer capítulo |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| Conformación del cuarto capítulo |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| Conclusión, anexos y bibliografía |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| Codificación de la aplicación web |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| Periodo de pruebas de funcionalidad |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| Corrección de errores |  |  |  |  |  |  |  |  |  |  | |  |  |  |  | |
| Implementación de la aplicación |  |  |  |  |  |  |  |  |  |  | |  |  |  | G | |

Fuente: Creación propia (2022)

### 3.1.3 Infraestructura de la empresa

…

### 3.1.4 Infraestructura mínima deseada de la empresa

…

## 3.2 Alcances y enfoque de investigación.

### 3.2.1 Alcances

A continuación, se describen los principales alcances para este proyecto.

* La aplicación web servirá para \*\*\*

### 3.2.2 Enfoque de la investigación

El enfoque de este trabajo de investigación es de tipo cualitativo ya que recolecta datos no estandarizados de una muestra que no es representativa (aplicada solo a \*\*\*) con el objetivo de alcanzar una mejor comprensión de la información recaudada en el proceso de \*\*\* como parte del tema de investigación.

Con relación al alcance de este trabajo se destaca que es de tipo descriptivo pues pretende que a partir de la información recopilada de las entrevistas realizadas a \*\*\* se establezcan las ventajas y desventajas de la implementación de la aplicación web.

### 3.2.3 Limitaciones de la aplicación

La aplicación web cuenta con las siguientes restricciones:

* La aplicación web depende de la respuesta del navegador al cual se ejecutará. En caso de fallar es posible que la aplicación web no pueda ejecutar con éxito alguna de sus funciones.
* Para utilizar la aplicación, necesita un dispositivo informático y una conexión a internet estable.
* Para utilizar la aplicación, debe tener conocimientos básicos de informática.
* \*\*\*

### 3.2.4 Características

* La aplicación no debe descargarse e instalarse en los dispositivos que acceden a ella, ya que está alojada en un servidor.
* Las solicitudes por parte de la aplicación serán consideradas únicamente por personal autorizado de la empresa Asociaciones Agroindustriales Serranas S.A. de C.V
* \*\*\*

### 3.2.5 Ventajas

\*\*\*

## 3.3 Hipótesis

El diseño, desarrollo e implementación de un sistema de información basado en web, optimizará el proceso de recepción de pimienta en la empresa Asocioaciones Agroindustriales Serranas S.A. de C.V.

## 3.4 Diseño y metodología de la investigación

### 3.4.1 Herramientas de apoyo

\*\*\*

### 3.4.2 Fases

\*\*\*

### 3.4.3 Selección de muestra

\*\*\*

### 3.4.4 Recolección de datos

\*\*\*

## 3.5 Desarrollo de la metodología

### 3.5.1 Requerimientos funcionales

Tabla 4  
REQ-01 Inicio de sesión

|  |  |
| --- | --- |
| **REQ-01** | **INICIO DE SESIÓN** |
| **TIPO DE REQUERIMIENTO** | Funcional. |
| **DESCRIPCIÓN** | La aplicación necesita recibir el nombre de usuario y la contraseña como entrada y verificarlos. Si los datos son correctos, puede ingresar al sistema. |
| **PRECONDICIÓN** | Los datos y la contraseña deben estar almacenados en la base de datos para verificarse cada vez que se requiera un inicio de sesión. |
| **VOCABULARIO Y FORMATOS** | Contraseña: Será un conjunto de letras, números y/o caracteres que el usuario haya asignado y dado de alta por parte del administrador para ingresar al sistema.  Base de Datos: “Recopilación organizada de información o datos estructurados, que normalmente se almacena de forma electrónica en un sistema informático. Normalmente, una base de datos está controlada por un sistema de gestión de bases de datos (DBMS)” (Oracle, 2022). |

Fuente:Creación Propia, 2022

Tabla 5  
REQ-02 Panel principal

|  |  |
| --- | --- |
| **REQ-02** | **PANEL PRINCIPAL** |
| **TIPO DE REQUERIMIENTO** | Funcional. |
| **DESCRIPCIÓN** |  |
| **PRECONDICIÓN** |  |
| **VOCABULARIO Y FORMATOS** |  |

Fuente:Creación Propia, 2022

Tabla 6  
REQ-03 Nuevo reporte de inspección

|  |  |
| --- | --- |
| **REQ-03** | **NUEVO REPORTE DE INSPECCIÓN** |
| **TIPO DE REQUERIMIENTO** | Funcional. |
| **DESCRIPCIÓN** |  |
| **PRECONDICIÓN** |  |
| **VOCABULARIO Y FORMATOS** |  |

Fuente:Creación Propia, 2022

Tabla 7  
REQ-04 Creación de comprobante de entrada

|  |  |
| --- | --- |
| **REQ-04** | **CREACIÓN DE COMPROBANTE DE ENTRADA** |
| **TIPO DE REQUERIMIENTO** | Funcional. |
| **DESCRIPCIÓN** |  |
| **PRECONDICIÓN** |  |
| **VOCABULARIO Y FORMATOS** |  |

Fuente:Creación Propia, 2022

Tabla 8  
REQ-05 Historial de entradas de pimienta

|  |  |
| --- | --- |
| **REQ-05** | **HISTORIAL DE ENTRADS DE PIMIENTA** |
| **TIPO DE REQUERIMIENTO** | Funcional. |
| **DESCRIPCIÓN** |  |
| **PRECONDICIÓN** |  |
| **VOCABULARIO Y FORMATOS** |  |

Fuente:Creación Propia, 2022

Tabla 9  
REQ-06 Creación de orden de compra

|  |  |
| --- | --- |
| **REQ-06** | **CREACIÓN DE ORDEN DE COMPRA** |
| **TIPO DE REQUERIMIENTO** | Funcional. |
| **DESCRIPCIÓN** |  |
| **PRECONDICIÓN** |  |
| **VOCABULARIO Y FORMATOS** |  |

Fuente:Creación Propia, 2022

Tabla 8  
REQ-07 Historial de órdenes de compra

|  |  |
| --- | --- |
| **REQ-07** | **HISTORIAL DE ÓRDENES DE COMPRA** |
| **TIPO DE REQUERIMIENTO** | Funcional. |
| **DESCRIPCIÓN** |  |
| **PRECONDICIÓN** |  |
| **VOCABULARIO Y FORMATOS** |  |

Fuente:Creación Propia, 2022

### 3.5.2 Diagramas de caso de usos

\*\*\*

### 3.5.3 Casos de uso

Tabla 9  
CU-01 Inicio de sesión

|  |  |  |  |
| --- | --- | --- | --- |
| **CU-01** | **INICIAR SESIÓN** | | |
| **Descripción** | La aplicación deberá permitir al usuario entrar al sistema para poder realizar ciertas acciones que le competen al mismo. | | |
| **Precondición** | El usuario deberá estar registrado en la base de datos previamente por el administrador para poder iniciar sesión. | | |
| **Secuencia Normal** | **Paso** | **Acción** | |
| 1 | Ingresar el nombre de usuario. | |
| 2 | Ingresar la contraseña. | |
| 3 | Validar datos. | |
|  | 3a | Si los datos ingresados son correctos, la aplicación dará acceso al usuario. |
| 3b | En caso contrario la aplicación deberá mostrar un mensaje que diga “Usuario y/o contraseña incorrectos”. |
| 4 | Salir. | |
| **Postcondición** | El usuario podrá hacer todas las acciones que se encuentran a su disposición. | | |
| **Excepciones** | **Paso** | **Acción** | |
| 1.1 | El usuario podrá recuperar su usuario y/o contraseña al contactar al administrador. | |
| **Rendimiento** | La aplicación realizará las acciones descritas, dependiendo de la velocidad de la conexión a Internet y las características de la computadora. Tiempo estimado de treinta segundos para la autenticación. | | |
| **Frecuencia** | Cada vez que se requiera. | | |
| **Importancia** | Vital. | | |
| **Urgencia** | Inmediatamente. | | |
| **Comentarios** | Ninguno. | | |

Fuente: Creación Propia, 2022

Tabla 10  
CU-02 Panel principal

|  |  |  |
| --- | --- | --- |
| **CU-02** | **PANEL PRINCIPAL** | |
| **Descripción** | La aplicación deberá de ser capaz de \*\*\* | |
| **Precondición** | El usuario deberá contar con \*\*\* | |
| **Secuencia Normal** | **Paso** | **Acción** |
| 1 | Ingresar \*\*\* |
|  |  |
|  |  |
| **Postcondición** | \*\*\*\* | |
| **Excepciones** | **Paso** | **Acción** |
| 1.1 | No existen registros. |
| **Rendimiento** | La aplicación realizará las acciones descritas, dependiendo de la velocidad de la conexión a Internet y las características de la computadora. | |
| **Frecuencia** | En caso de ser necesario. | |
| **Importancia** | Vital. | |
| **Urgencia** | Inmediatamente. | |
| **Comentarios** | Ninguno. | |

Fuente: Creación Propia, 2022

Tabla 11  
CU-03 Nuevo reporte de inspección

|  |  |  |
| --- | --- | --- |
| **CU-03** | **NUEVO REPORTE DE INSPECCIÓN** | |
| **Descripción** | La aplicación deberá de ser capaz de \*\*\* | |
| **Precondición** | El usuario deberá contar con \*\*\* | |
| **Secuencia Normal** | **Paso** | **Acción** |
| 1 | Ingresar \*\*\* |
|  |  |
|  |  |
| **Postcondición** | \*\*\*\* | |
| **Excepciones** | **Paso** | **Acción** |
| 1.1 | No existen registros. |
| **Rendimiento** | La aplicación realizará las acciones descritas, dependiendo de la velocidad de la conexión a Internet y las características de la computadora. | |
| **Frecuencia** | En caso de ser necesario. | |
| **Importancia** | Vital. | |
| **Urgencia** | Inmediatamente. | |
| **Comentarios** | Ninguno. | |

Fuente: Creación Propia, 2022

Tabla 12  
CU-04 Creación de comprobante de entrada

|  |  |  |
| --- | --- | --- |
| **CU-04** | **CREACIÓN DE COMPROBANTE DE ENTRADA** | |
| **Descripción** | La aplicación deberá de ser capaz de \*\*\* | |
| **Precondición** | El usuario deberá contar con \*\*\* | |
| **Secuencia Normal** | **Paso** | **Acción** |
| 1 | Ingresar \*\*\* |
|  |  |
|  |  |
| **Postcondición** | \*\*\*\* | |
| **Excepciones** | **Paso** | **Acción** |
| 1.1 | No existen registros. |
| **Rendimiento** | La aplicación realizará las acciones descritas, dependiendo de la velocidad de la conexión a Internet y las características de la computadora. | |
| **Frecuencia** | En caso de ser necesario. | |
| **Importancia** | Vital. | |
| **Urgencia** | Inmediatamente. | |
| **Comentarios** | Ninguno. | |

Fuente: Creación Propia, 2022

Tabla 13  
CU-05 Historial de entradas de pimienta

|  |  |  |
| --- | --- | --- |
| **CU-05** | **HISTORIAL DE ENTRADAS DE PIMIENTA** | |
| **Descripción** | La aplicación deberá de ser capaz de \*\*\* | |
| **Precondición** | El usuario deberá contar con \*\*\* | |
| **Secuencia Normal** | **Paso** | **Acción** |
| 1 | Ingresar \*\*\* |
|  |  |
|  |  |
| **Postcondición** | \*\*\*\* | |
| **Excepciones** | **Paso** | **Acción** |
| 1.1 | No existen registros. |
| **Rendimiento** | La aplicación realizará las acciones descritas, dependiendo de la velocidad de la conexión a Internet y las características de la computadora. | |
| **Frecuencia** | En caso de ser necesario. | |
| **Importancia** | Vital. | |
| **Urgencia** | Inmediatamente. | |
| **Comentarios** | Ninguno. | |

Fuente: Creación Propia, 2022

Tabla 14  
CU-06 Creación de orden de compras

|  |  |  |
| --- | --- | --- |
| **CU-06** | **CREACIÓN DE ORDEN DE COMPRAS** | |
| **Descripción** | La aplicación deberá de ser capaz de \*\*\* | |
| **Precondición** | El usuario deberá contar con \*\*\* | |
| **Secuencia Normal** | **Paso** | **Acción** |
| 1 | Ingresar \*\*\* |
|  |  |
|  |  |
| **Postcondición** | \*\*\*\* | |
| **Excepciones** | **Paso** | **Acción** |
| 1.1 | No existen registros. |
| **Rendimiento** | La aplicación realizará las acciones descritas, dependiendo de la velocidad de la conexión a Internet y las características de la computadora. | |
| **Frecuencia** | En caso de ser necesario. | |
| **Importancia** | Vital. | |
| **Urgencia** | Inmediatamente. | |
| **Comentarios** | Ninguno. | |

Fuente: Creación Propia, 2022

Tabla 15  
CU-07 Historial de órdenes de compra

|  |  |  |
| --- | --- | --- |
| **CU-07** | **HISTORIAL DE ÓRDENES DE COMPRA** | |
| **Descripción** | La aplicación deberá de ser capaz de \*\*\* | |
| **Precondición** | El usuario deberá contar con \*\*\* | |
| **Secuencia Normal** | **Paso** | **Acción** |
| 1 | Ingresar \*\*\* |
|  |  |
|  |  |
| **Postcondición** | \*\*\*\* | |
| **Excepciones** | **Paso** | **Acción** |
| 1.1 | No existen registros. |
| **Rendimiento** | La aplicación realizará las acciones descritas, dependiendo de la velocidad de la conexión a Internet y las características de la computadora. | |
| **Frecuencia** | En caso de ser necesario. | |
| **Importancia** | Vital. | |
| **Urgencia** | Inmediatamente. | |
| **Comentarios** | Ninguno. | |

Fuente: Creación Propia, 2022

### 3.5.4 Diagrama de navegación

\*\*\*\*

### 3.5.5 Diagrama de la base de datos

\*\*\*\*

## 7.1 Bibliografía

Axios (2021). “Empezando Axios”. Agosto 2022. <https://axios-http.com/es/docs/intro>

Express (2022). “Infraestructura web rápida, minimalista y flexible para Node.js”. Agosto 2022. <https://expressjs.com/es/>

Guaña J. (2022). “Como usar Axios con React: La guía definitiva (2021)”. Agosto 2022, de freeCodeCamp: <https://www.freecodecamp.org/espanol/news/como-usar-axios-con-react/>

Halliday P. (2020). “Como usar Axios con React”. Agosto 2022, de Digital Ocean: <https://www.digitalocean.com/community/tutorials/react-axios-react-es>

Hernández M. (2021). “¿Qué es NPM?”. Agosto 2022, de freeCodeCamp: <https://www.freecodecamp.org/espanol/news/que-es-npm/>

MDN Contributors (2022). “Introducción a Express/Node”. Agosto 2022, de MDN WEB DOCS: <https://developer.mozilla.org/es/docs/Learn/Server-side/Express_Nodejs/Introduction>

MDN Contributors (2022). “Primeros pasos en React”. Agosto 2022, de MDN WEB DOCS: <https://developer.mozilla.org/es/docs/Learn/Tools_and_testing/Clientside_JavaScript_frameworks/React_getting_started>

Node JS (2016). “Acerca de Node.js®”. Agosto 2022. <https://nodejs.org/es/about/>

Node JS (2016). “Versiones”. Agosto 2022. <https://nodejs.org/es/about/releases/>

RockContent (2020). “Bootstrap: guía para principiantes de qué es, por qué y cómo usarlo”. Agosto 2022. <https://rockcontent.com/es/blog/bootstrap/>

https://www.analyticaweb.com/desarrollo-web/aprende-ya-que-es-un-orm

https://www.prisma.io/docs/concepts/components/prisma-schema

https://dev.to/denispixi/prisma-un-toolkit-para-bases-de-datos-orm-para-typescript-y-node-js-3g9#:~:text=Prisma%20es%20un%20kit%20de,esquema%20de%20base%20de%20datos.