

DISEÑO DE SISTEMA WEB DE TRAZABILIDAD PARA AUTOMATIZAR LA GESTIÓN DE LOS PROCESOS DE PRODUCCIÓN Y VENTA DE LA NUEZ DE MACADAMIA EN EL RANCHO CHANTORO

Romero Ruiz Abel, Ruiz Tadeo Ana Claudia, Garcia Virgen Juan, Mata Lopez Walter Alexander, Garcia Diaz Noel, Verduzco Ramírez Jesús Alberto

Instituto Tecnológico de México campus Colima

División de Estudios de Posgrado e Investigación

Av. Tecnológico No.1 Villa de Álvarez, Col. C.P. 28976 Villa de Álvarez, colonia Liberación

Tel. +52312312639,

g2346015@colima.tecnm.mx, aruiz@colima.tecnm.mx, jgarcia@colima.tecnm.mx, wmata@ucol.mx,
ngarcia@colima.tecnm.mx, averduzco@colima.tecnm.mx

RESUMEN.

La nuez de macadamia es un fruto seco altamente valorado por su sabor y beneficios nutricionales. En México se cosecharon 380 toneladas de nuez de macadamia entre 2009 y 2010, estos datos muestran la importancia de la automatización de los procesos de trazabilidad. Por lo cual la falta de información en la cadena de suministros es una falla crítica, que afecta a productores y clientes finales. El resultado que se presenta en esta investigación es el diseño de un sistema de trazabilidad para el registro de datos, seguimiento de siembras, cosechas y ventas, mejorando con esto la inocuidad de los cultivos, para esto se utilizó la metodología en espiral y se realizó el diseño y la evaluación de las interfaces a implementar. También se diseñó la arquitectura del software que soporta la implementación del sistema, que automatiza la trazabilidad de un proceso que impacta positivamente en la agricultura.

Palabras Clave: Trazabilidad, Sistema, Nuez de Macadamia, Producción, Inocuidad.

ABSTRACT.

Macadamia nuts are highly valued for their flavor and nutritional benefits. In Mexico, 380 tons of macadamia nuts were harvested between 2009 and 2010, showing the importance of automating traceability processes. Therefore, the lack of information in the supply chain is a critical failure that affects producers and end customers. The result presented in this research is the design of a traceability system for data recording, tracking of planting, harvesting, and sales, thereby improving the safety of crops. For this, the spiral methodology was used, and the design and evaluation of the interfaces to be implemented were carried out. The software architecture that supports the implementation of the system, which automates the traceability of a process that positively impacts agriculture, was also designed.

Keywords: Traceability, System, Macadamia Nut, Production, Food Safety.

1. INTRODUCCIÓN

En 2021, a nivel mundial, los principales países exportadores de nueces de macadamia, frescas o secas, con cáscara, fueron Sudáfrica (MX\$1,908M), Australia (MX\$1,708.2M) y Guatemala (MX\$460.8M) [5].

La nuez de macadamia llegó a México en 1965. En México, hay un gran potencial para cultivar esta nuez debido a las condiciones climatológicas y lo óptimo del terreno [3].

Teniendo en cuenta que el cultivo de la nuez de macadamia es poco conocido en México, este puede tener el potencial para crecer aceleradamente y exportarse a los países con mayor demanda. La

trazabilidad juega un papel clave, ya que es una herramienta muy importante para el seguimiento de los procesos de los productos, proporcionando pautas para lograr mejoras y reducir los errores en la cadena de suministro.

La trazabilidad es una herramienta mediante la cual se pueden conocer todos los elementos que intervienen en la elaboración de un producto, desde las materias primas, aditivos, empaques y seguimiento en los procesos de siembra. Por lo tanto, es una herramienta importante para utilizar en las industrias que requieren más información y proporcionará una forma de alcanzar los objetivos finales [14].

En este documento se presenta un modelo para la implementación de trazabilidad interna y externa para el rancho Chantoro utilizando herramientas tecnológicas, además de pruebas de usabilidad para comprobar que se cumplieron los requerimientos del sistema.

1.1. Antecedentes

En los últimos años, se han desarrollado diversos sistemas de trazabilidad. En el trabajo desarrollado en [7], se presenta una solución para mejorar la trazabilidad de los productos con una tecnología móvil, en la cual usaron códigos QR (quick-response code) en la producción de banano. Asimismo, el sistema implementado por Cervantes para las huertas de aguacate fue hecho con una aplicación web [9]. Del mismo modo, Gilces y Velastegui implementaron un sistema de trazabilidad para la granja "10 Hermanos", el cual dio seguimiento a 15 hectáreas de producto [4]. Otro de los sistemas que se implementaron fue un sistema de trazabilidad para el suministro de cacao en la cordillera nariñense colombiana; el resultado fue una aplicación móvil [1]. Por otro lado, Sánchez diseñó un sistema para el seguimiento de la producción y control de materias primas de una empresa llamada Robottions [8].

En el trabajo de Velaverde se presenta la trazabilidad para garantizar la inocuidad de los productos, a través del diseño de un sistema que permite dar seguimiento a los procesos durante la creación de los productos, siguiendo la norma ISO 22000:2005 [15].

En este artículo, se describe el diseño de un sistema de trazabilidad utilizando una aplicación web, el cual dará seguimiento a la

producción de nuez de macadamia en el apartado de trazabilidad interna y externa, para lo cual se implementará una base de datos que almacenará la información y proporcionará al usuario la posibilidad de tomar decisiones informadas.

El problema que se presenta en la granja Chantoro es que no existe ningún tipo de tecnología que pueda dar seguimiento al proceso de producción de la nuez de macadamia, por lo cual se realizó el diseño del sistema que automatizará la trazabilidad interna y externa para su implementación en la empresa Chantoro.

2. Métodos y herramientas

A continuación, se muestran las herramientas tecnológicas aplicadas para el desarrollo del sistema objeto de esta investigación que se muestra en la figura 1 el cual se integra de los siguientes elementos: Base de datos, REST API, Frontend, gestión de reportes, gestión de almacén, gestión de cosechas, gestión de clientes y gestión de ventas y las que se utilizarán para la codificación e implementación del mismo.

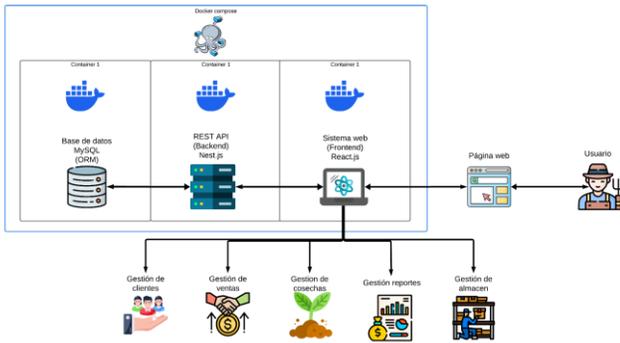


Fig. 1. Modelo conceptual (Autoría propia).

2.1. Enterprise architect

Enterprise Architect es una herramienta gráfica multiusuario diseñada para ayudar a los equipos de desarrolladores de software a crear sistemas robustos y fáciles de mantener, la cual ofrece una gran variedad de módulos con los cuales se pueden definir proyectos de manera eficiente. Con esta herramienta, Architect permitirá ayudar a simplificar la manera en la cual se presenta la información por medio de la estructura que proporciona esta herramienta entre las cuales podemos ver que son: diagrama de secuencia, diagrama de interfaces, diagrama de casos de uso, diagrama de despliegue, diagrama de clases los cuales usaremos en este artículo [12].

2.2. Figma

Figma sirve para crear, compartir y evaluar diseños de sitios web, aplicaciones móviles y otros productos. Es una herramienta muy popular entre diseñadores, jefes de producto, redactores y desarrolladores [20]. Figma es una herramienta que se utiliza para plasmar el diseño de las interfaces web, porque tiene la ventaja de concluir las interfaces de manera rápida y sencilla.

2.3. UML

UML es un lenguaje para definir diseños de software, el cual se puede usar para visualizar, especificar, construir y documentar las piezas de software. Algunos de los diagramas que se implementarán son los diagramas de clases y de interfaz de usuario. Esta aplicación brinda las herramientas para modelar sistemas de manera eficiente, documentando lo que se está diseñando [18]. Y para definir cómo se deben de visualizar las representaciones del sistema objeto de esta investigación.

2.4. Node.js

Node.js es una aplicación de tiempo de ejecución asíncrona de JavaScript controlada por eventos, está diseñada para crear aplicaciones de red escalables en las cuales puede controlar conexiones simultáneamente. Este entorno de JavaScript permitirá ejecutar código en un entorno tipo servidor [24]. Ya que provee las ventajas de poder ejecutar código de JavaScript en el servidor, con esto se brinda muchas posibilidades a diferencia de ejecutar JavaScript puro.

2.5. TypeScript

TypeScript es una sintaxis adicional a JavaScript para admitir una integración más estrecha con su editor, donde se pueda detectar errores antes de ejecutar la aplicación, dando confianza a la hora de crear código. El código de TypeScript se convierte en JavaScript, que se ejecuta en cualquier lugar donde se ejecute JavaScript: en un navegador, en Node.js o en sus aplicaciones [11]. Se pretende utilizar este lenguaje de programación para el desarrollo del sistema en su totalidad (Backend y Frontend). Los beneficios que ofrece, a diferencia de solo usar JavaScript, es que tiene tipado estático, el cual previene los posibles errores antes de ejecutar el código.

2.6. React.js

React.js es una librería que permite crear componentes juntos y crear interfaces de usuario. Con esta tecnología podremos crear toda la parte del Frontend [17]. Se pretende usar React.js en el apartado del Frontend porque es una librería que tiene mucha popularidad y da la posibilidad de crear nuestros propios componentes desde cero. Otra de las ventajas es que el creador de esta librería es Facebook, una empresa con gran repercusión en internet.

2.7. Nest.js

Nest.js es un Framework para construir de manera eficiente y escalable aplicaciones de Node.js usando JavaScript, y soporta TypeScript, y combina elementos de POO (Programación Orientada a Objetos), PF (Programación Funcional), y PFR (Programación Funcional Reactiva) [21]. Se eligió Nest.js como el Framework que se utilizará en el apartado del Backend. Las principales razones por las cuales se propone su aplicación debido a que su licencia es gratuita.

2.8. ORM

ORM por sus siglas en inglés Object Relational Mappings, es un modelo de programación el cual permite mapear o estructurar las bases de datos relacionales sobre una estructura de entidades, con el objetivo de acelerar el desarrollo de las aplicaciones [10]. Se decidió usar ORM por su flexibilidad para la creación de la base de datos y la facilidad que brinda para el mantenimiento futuro.

2.9. TypeORM

TypeORM por sus siglas en inglés Object-Relational Mapper para TypeScript y JavaScript, el cual permite interactuar con bases de datos a través de objetos y clases, facilita la manipulación de datos y la implementación de relaciones entre tablas [11]. Se propone utilizar TypeORM debido a su facilidad para migrar datos a cualquier tipo de base de datos sin necesidad de cambiar la estructura. Esto se debe a que emplea clases y decoradores.

2.10. MYSQL

MySQL es una base de datos de código abierto que es administrada, mantenida, desarrollada y distribuida por Oracle Corporation. Se eligió esta tecnología para la base de datos ya que es de código abierto y porque su sintaxis no difiere mucho de los motores de otras bases de datos [16]. Además es código abierto y, por lo tanto, no tiene ningún costo de licencia.

2.11. Manual de buenas prácticas agrícolas

El Manual de Buenas Prácticas Agrícolas es una de las principales ayudas que manejan los lineamientos para el empaque de los productos, los formatos para registrar las operaciones, los reglamentos y las normas existentes para la agricultura. Esta información ayuda a alcanzar la inocuidad de los alimentos [23]. El uso de las buenas prácticas agrícolas ayudará a establecer qué lineamientos y reglas a seguir para el diseño del sistema de trazabilidad, sin costo alguno.

2.12. Trazabilidad interna

La trazabilidad interna es el registro y seguimiento de los productos que se desarrollan dentro de la empresa, la forma en que se producen, los tiempos de fabricación y el control de existencia [13]. Se decidió diseñar este tipo de trazabilidad en el sistema, dado que es importante establecer un registro de los datos en todas las etapas del proceso de producción y permitir que los productos puedan garantizar la inocuidad.

2.13. Trazabilidad externa

La trazabilidad externa incluye los productos preparados para su expedición y venta para el cliente inmediato al que se le entregan. Un ejemplo sería a quien se le vendió, cuando y cuanto se le vendió [13]. La trazabilidad externa forma parte de las acciones que realiza la empresa cuando el producto sale al mercado. Un ejemplo serían las ventas realizadas, las cuales llevan un registro de cuánto se vendió y a quién se le vendió. Es importante tener un historial de las actividades realizadas y con esto apoyar en la toma de

decisiones a los usuarios del sistema a implementar en la empresa el Rancho Chantoro.

2.14. Lucidchart

Es una herramienta en línea de diagramación y visualización de datos que permite a los usuarios crear una amplia variedad de diagramas, como diagramas de flujo, organigramas, mapas conceptuales, mapas mentales y modelos de entidad-relación [22]. Esta herramienta permitió construir los diagramas de flujo de la aplicación facilitando el diseño y acotando los tiempos de creación de los diagramas.

2.15. Docker

Docker es una plataforma de contenedores de software que permite a los desarrolladores empaquetar aplicaciones junto con sus dependencias y entornos en contenedores aislados [28]. Se propone el uso de Docker para crear contenedores que alojan el Frontend, Backend y la base de datos, permitiendo una replicación eficiente y sin complicaciones de toda la aplicación. La contenedorización con Docker facilita la portabilidad y despliegue de la aplicación en cualquier entorno, garantizando consistencia entre los entornos de desarrollo, prueba y producción.

2.16. Docker compose

Docker Compose es una herramienta que facilita la definición y ejecución de aplicaciones multicontenedor en Docker. A través de un archivo YAML, se puede configurar la infraestructura completa de una aplicación, especificando los servicios, volúmenes y redes necesarios [29]. Para gestionar y orquestar estos contenedores, se sugiere el uso de Docker Compose, una herramienta que permite levantar múltiples contenedores de manera sencilla y simultánea mediante un archivo de configuración (docker-compose.yml). Docker Compose no solo facilita el inicio de los servicios, sino que también abstrae los contenedores dentro de una red privada, asegurando que solo puedan comunicarse entre ellos, lo que añade una capa adicional de seguridad al sistema.

2.17. Principios de usabilidad

Estos principios de usabilidad los propuso Jakob Nielsen en 1995, que son 10 principios generales de usabilidad Web, que construyen la cimentación para diseñar sitios "user friendly", en pocas palabras centrados y optimizados para los usuarios para mejorar sus experiencias [26]. Los parámetros de evaluación de usabilidad de una interfaz de usuario, que establece el autor son: Visibilidad del estado del sistema, Coincidencia entre el sistema y el mundo real, Control y libertad del usuario, Coherencia y estándares, Prevención de errores, Reconocimiento en lugar de recuerdo, Flexibilidad y eficiencia de uso, Diseño estético y minimalista, Ayuda a los usuarios a reconocer, diagnosticar y recuperarse de errores y Permite la documentación. Estos parámetros se utilizaron para evaluar la usabilidad de las interfaces a integrar en el sistema de trazabilidad a implementar en el Rancho Chantoro.

3. DESARROLLO

La metodología seleccionada del autor Barry W. Boehm para realizar esta investigación fue la espiral la cual ofrece una mejora continua en todas las etapas de desarrollo del sistema. En este modelo se realizan interacciones que son llamadas espirales, que pasan por cuatro fases, que son: Planificación, Análisis de Riesgos, Implementación y Evaluación, como se muestra en la figura 2 [18]. En esta investigación se realizó el diseño de un sistema de trazabilidad para lo cual se procedió a recopilar requerimientos por medio de entrevistas con los agricultores, posteriormente se diseñaron los diagramas de clases, diagrama de flujo de la aplicación, el modelo conceptual y por último se procedió a la creación de las interfaces del sistema.

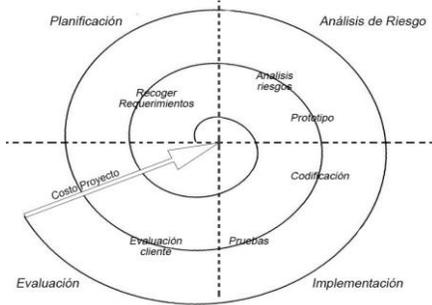


Fig. 2. Modelo en espiral.

3.1. Planificación

En la granja Chantoro, ubicada en La Montrica, Colima, Col., México, se tienen problemas para dar seguimiento a la producción de la nuez de macadamia. Por esta razón, se realizó el diseño de un sistema de trazabilidad para la producción de este Rancho. Actualmente, la empresa realiza el seguimiento de manera tradicional, utilizando papel, lo que genera errores y pérdidas de información.

Para implementar la metodología aplicada en esta investigación como primer paso se realizó la recopilación de los requisitos funcionales y los no funcionales del sistema mediante consultas con el productor. Dado que el productor no cuenta con un sistema para registrar los datos de las operaciones de la siembra, cosecha y venta. Después de esta planificación se procedió a realizar el diseño del sistema. El manual de buenas prácticas agrícolas apoyó en el diseño del sistema de trazabilidad para la determinación de los datos a registrar de las actividades los cuales son: nombre, área donde se realiza, material y equipo, la frecuencia y tiempo de realización, responsable y descripción de los pasos de cada actividad.

Como resultado del análisis realizado, se definieron los siguientes requisitos funcionales y no funcionales que se muestran en la Tabla 1.

Tabla. 1. Lista de requisitos funcionales y no funcionales del sistema (Autoría propia).

Requisitos funcionales	Requisitos no funcionales
RF1 El sistema debe gestionar un catálogo de clientes, lo cual permite visualizar, ingresar, actualizar y eliminar clientes.	RNF1 El sistema debe cargarse en 2 segundos.
RF2 El sistema debe gestionar las ventas, lo cual permite visualizar, ingresar, actualizar y finalizar la venta.	RNF2 El sistema debe recuperarse automáticamente después de un fallo.
RF3 El sistema debe gestionar el catálogo de productos, lo cual permite visualizar, ingresar, eliminar y actualizar.	RNF3 El sistema debe ser intuitivo y fácil de aprender para los usuarios nuevos.
RF4 El sistema debe permitir generar los reportes de ventas mensualmente y anualmente.	RNF4 El sistema debe estar compuesto por módulos independientes para facilitar futuras actualizaciones.
RF5 El sistema debe permitir gestionar la sesión del usuario.	RNF5 El sistema debe poder adaptarse a diferentes entornos de hardware o software.
RF6 El sistema debe gestionar un catálogo de tipos de unidad, lo cual permitirá visualizar, ingresar, actualizar y eliminar los tipos de unidades de medida.	RNF6 La interfaz de usuario debe ser amigable y fácil de navegar.
RF7 El sistema debe gestionar los insumos, lo cual permite visualizar, ingresar, actualizar y eliminar.	RNF7 El sistema debe estar disponible el 99.9% del tiempo.
RF8 El sistema debe gestionar los tipos de actividades, lo cual permite visualizar, ingresar, actualizar y eliminar.	RNF8 El sistema debe soportar diferentes plataformas, como Windows, Linux y macOS.

3.2. Análisis de riesgo

El siguiente paso de la metodología aplicada para el desarrollo del sistema a implementar en el Rancho Chantoro es el Análisis de riesgo. En el cual se diseñaron los diagramas de casos de uso e interfaces del sistema. Se consideró para este paso el tipo de usuario "Agricultor", quien tendría las opciones de registrar una cosecha, para registrarla es necesario primero registrar un cultivo y después registrar los insumos que se utilizaron en la misma.

En el sistema, se tomó en cuenta que existen dos actores: el "Agricultor", que se muestra en la figura 3, y el "Administrador" que está representado en la figura 4, en el cual se definieron las acciones necesarias para el buen funcionamiento, separando las actividades que corresponden únicamente a un Administrador. Estas serían de las más importantes, como eliminar datos de los catálogos. Un ejemplo sería eliminar un cultivo, que es una acción que solo puede realizar el usuario con mayores privilegios.

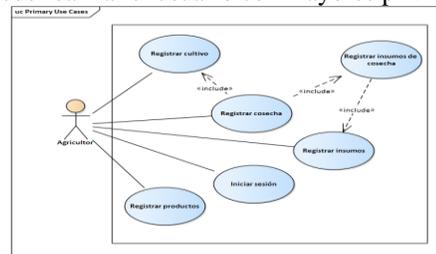


Fig. 3. Diagrama de casos de uso de productor (Autoría propia).

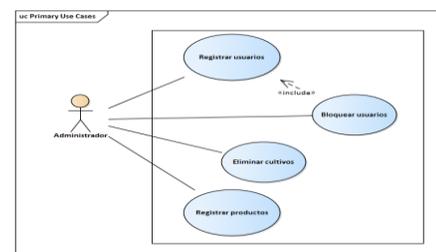


Fig. 4. Diagrama de casos de uso del administrador (Autoría propia).

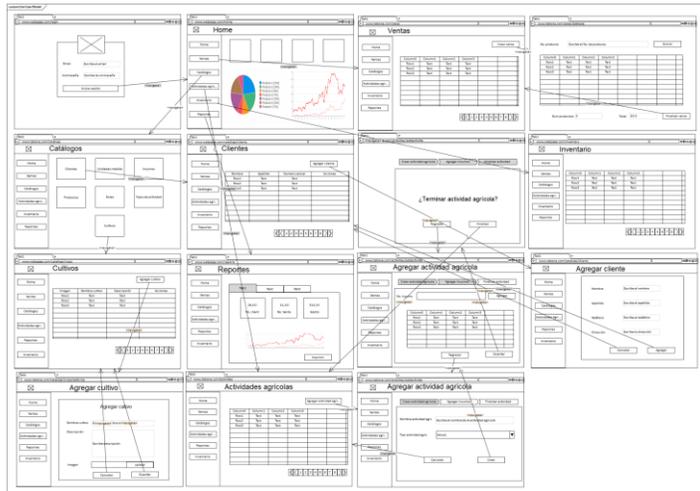


Fig. 5. Diagrama de interacción de la aplicación (Autoría propia).

En la figura 5 se describe el flujo de navegación entre las diferentes funcionalidades del sistema de gestión agrícola. El usuario comienza en la pantalla de inicio, desde donde puede acceder a módulos como ventas, catálogos, clientes, cultivos, reportes y actividades agrícolas. Cada módulo permite realizar operaciones específicas, como crear o consultar ventas, gestionar el inventario de cultivos y productos, o generar reportes sobre el rendimiento de las ventas y actividades agrícolas. A lo largo del proceso, las conexiones entre las pantallas indican cómo el usuario puede navegar fácilmente entre las diferentes áreas del sistema, proporcionando una interfaz eficiente para la administración de todas las actividades agrícolas involucradas.

La siguiente actividad del análisis de riesgo fue el diseñar el diagrama de clases, mostrado en la figura 6, para la base de datos, en el que se ven todas las clases, comenzando con la clase de Usuario, que tiene dos tipos de herencia, que son el Agricultor y el Cliente. Cabe destacar que el cliente sólo formará parte de un catálogo de clientes administrado por el agricultor que es el usuario con mayores privilegios en el sistema.

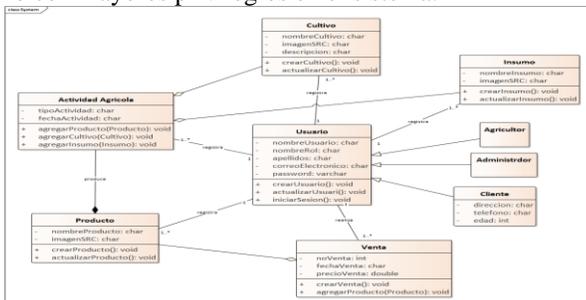


Fig. 6. Diagrama de clases (Autoría propia).

3.3. Implementación

La fase de la implementación del sistema se realizará con las herramientas de programación siguientes TypeScript, HTML,

CSS, JavaScript, esta implementación está planeada para la siguiente etapa de desarrollo del sistema.

3.4. Evaluación

En el último apartado, se realiza la evaluación de las interfaces con el agricultor para validar las pantallas del sistema y recibir propuestas de mejoras. Para realizar la evaluación de usabilidad de las interfaces se tomó de base el autor Jakob Nielsen con sus 10 principios [26].

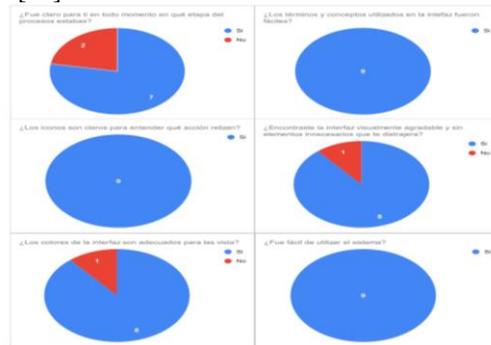


Fig. 7. Gráficas preguntas de usabilidad (Autoría propia).

El autor de los principios de usabilidad menciona en uno de sus artículos, que para medir el nivel de usabilidad de las interfaces se debe de aplicar como mínimo a cinco personas para poder obtener el 85% de los errores de usabilidad [25], en base a esto se aplicaron encuestas a nueve usuarios que probaron el sistema, esta encuesta contó con 10 preguntas basadas en los principios de usabilidad del autor, las cuales se muestran a en la figura 7.

El 90% de los usuarios indicaron que los colores de las interfaces son adecuados para la vista. Y con respecto a los términos y conceptos utilizados en la interfaz el 100% de los usuarios indicaron que fueron fáciles de comprender. Así como el 100% indicó que fue fácil de usar el sistema. Y también indicaron que los iconos son claros para entender, qué acción se está realizando. El 80% y 90% indicaron que la interfaz fue visualmente agradable.

4. Resultados

Los productos finales obtenidos son: el diseño del modelo conceptual basado en los requerimientos funcionales, el diagrama de interfaces, diagrama de interacción de la aplicación y los diagramas de casos de uso. Otro de los productos finales obtenidos fue el diseño de la base de datos y la evaluación de las interfaces del sistema.

4.1. Modelo conceptual del sistema

En la figura 8 se muestra el modelo conceptual del sistema que se desarrollará, el cual está compuesto por dos aplicaciones y una base de datos. La primera aplicación es el Frontend, que será implementado con la librería React.js. Esta aplicación permitirá la consulta de reportes, la visualización de los clientes actuales, la administración del inventario y la gestión de cosechas.

La segunda aplicación será una API REST, que se implementará utilizando el framework Nest.js para el desarrollo de aplicaciones de backend. Esta API se encargará de recibir las peticiones del frontend y realizar las consultas a la base de datos mediante un ORM (Object Relational Mapper), facilitando así el uso de la Programación Orientada a Objetos (POO). Por último, la base de datos se diseñará utilizando el motor MySQL. Los íconos utilizados en este modelo se tomaron de [27]. Tanto el frontend como el backend se implementarán en contenedores independientes de Docker, al igual que la base de datos, que estará en otro contenedor. Estos contenedores estarán conectados entre sí a través de una red interna, accesible únicamente entre ellos. Para gestionar el despliegue y la ejecución de estos contenedores, se utilizará la tecnología de Docker Compose. Finalmente, la aplicación será accesible a través del navegador web, proporcionando una interfaz amigable y fácil de usar para el usuario final.

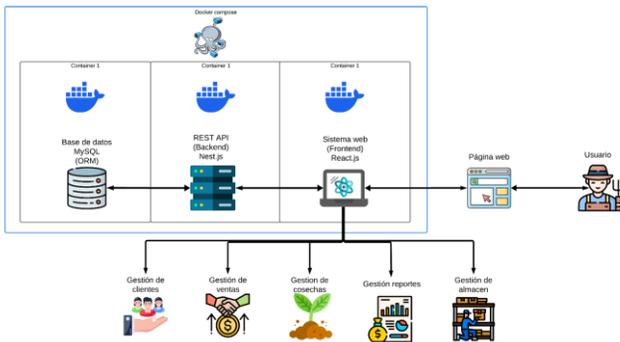


Fig. 8. Modelo conceptual del sistema (Autoría propia).

4.2. Interfaces del usuario

En la figura 9 se muestran las interfaces, diseñadas en el software Figma, donde se presentan las diferentes pantallas que se plantearon. Al inicio de la aplicación, habrá un login en el cual se validan las credenciales para entrar al sistema. Después de iniciar sesión, la siguiente pantalla será un menú donde se visualizarán los módulos de la aplicación, entre ellos, el módulo de catálogos, el módulo de entradas, el módulo de salidas, el módulo de reportes, módulo de actividades agrícolas y el módulo de inventario. A continuación, se describen detalladamente las funciones de cada módulo.

- Módulo de catálogos: Aquí se gestionan los catálogos existentes, clientes, cultivos, insumos, productos, roles de usuario, unidades de medida, tipo de actividad.
- Módulo de entradas: Este apartado permite gestionar las entradas realizadas por el usuario.
- Módulo de salidas: Este apartado permite gestionar las salidas realizadas por el usuario.
- Módulo de actividades agrícola (Gestión de cosechas): Este apartado permite gestionar las actividades que se le realizan a los cultivos.

- Módulo de reportes: En esta sección se pueden generar reportes de varios escenarios los cuales se mencionan a continuación:
 - Reporte de stock y producción
 - Productos más vendidos y menos vendidos
 - Reporte de gastos
 - Reporte de clientes
 - Reporte de ventas por fecha y periodo
- Módulo de inventario: En este se puede ver la existencia de cada uno de los productos o insumos en el sistema.

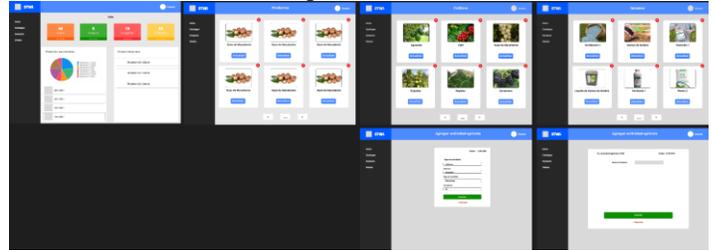


Fig. 9. Interfaces del sistema (Autoría propia).

4.3. Tablas de base de datos

Tomando en cuenta el diagrama de clases que se desarrolló, se crearon las siguientes tablas: detalleVenta, tipoActividad, usuarios, rolUsuario, cultivos, actividadesAgricola, actividadInsumo, insumos, unidadMedida, ventas, productos y tipoActividad para la base de datos, como se muestra en la figura 10.

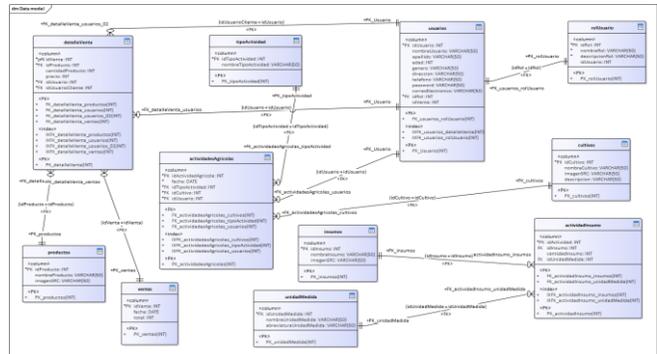


Fig. 10. Modelo del sistema (Autoría propia).

5. Conclusión

Este trabajo de investigación brinda los elementos y las bases para la codificación e implementación de un sistema de trazabilidad interna y externa para el registro de los datos de siembra, cosecha y venta de nuez de macadamia en el rancho Chantoro. También se presentan los resultados de la evaluación de las interfaces del sistema, el diagrama de base de datos y se definió la tecnología de programación a aplicar en el desarrollo del sistema, así como los datos a almacenar en cada una de las tablas y las interfaces finales. Se propone como trabajo futuro la implementación de un módulo especializado para predecir la producción de un cultivo utilizando lógica difusa.

Referencias

- [1] D. Y. Cañar Serna, J. L. Sepúlveda Forero, E. Martínez Pachón, L. P. Tibaduiza Castañeda, "Diseño de un sistema de trazabilidad como soporte al modelo productivo de cacao en la Cordillera Nariñense Colombiana", 2022.
- [2] D. Hernández, "Modelos del ciclo de vida del software, Julio 1, 2022. [Online], Disponible: <https://www.linkedin.com/pulse/modelos-del-ciclo-de-vida-software-daniel-herandez/>
- [3] MACAMEX, "Manual Técnico para productores de nuez de Macadamia", [Online], Disponible: <https://macadamiamexico.com/wp-content/uploads/2016/06/Manualmacadamia.pdf>
- [4] Y. N. Gilces Ruiz, W. K. Velastegui Paredes, "Implementación de un sistema web de trazabilidad para el cultivo de arroz en la hacienda '10 hermanos'", Trabajo de titulación, Universidad Agraria del Ecuador, Guayaquil, Guayas, 2022.
- [5] Gobierno de México, "Nueces de Macadamia, Frescas o Secas, con Cáscara", 2024, [Online], Disponible: <https://www.economia.gob.mx/datamexico/es/profile/product/macadamia-nuts-fresh-or-dried-shelled>
- [6] R. Hernández Sampieri, C. Fernández Collado, P. Baptista Lucio, Metodología de la investigación. McGraw-Hill Interamericana de España, 2014, pp. 8.
- [7] G. E. Silva Peñafiel, A. A. Flores Lescano, D. A. Jines Espín, A. D. Paredes Achatipán, "Implementación de un aplicativo móvil con códigos QR para mostrar información de trazabilidad", 2022.
- [8] J. Sánchez Jiménez, "Sistema de trazabilidad para el seguimiento de producción y control de stocks de materias primas", Trabajo de titulación, Universidad Jaume I, España, 2022.
- [9] R. Ochoa Ornelas, D. Fajardo Delgado, F. Zambrano Cervantes, R. Gudiño Venegas, "Trazabilidad para la cadena de suministro agrícola desde un enfoque informático", 2020.
- [10] J. A. Muro, "¿Qué es un ORM?", 2024, [Online], Disponible: <https://www2.deloitte.com/es/es/pages/technology/articles/que-es-orm.html>
- [11] TypeORM, "TypeORM - Amazing ORM for TypeScript and JavaScript (ES7, ES6, ES5)." TypeORM, 2023. [Online]. Available: <https://typeorm.io>
- [12] Sparx Systems, "Enterprise Architect", 2023, [Online], Disponible: <https://sparxsystems.com/products/ea/index.html/>
- [13] Servicio Nacional de Sanidad y Calidad Agroalimentaria, "Trazabilidad", 2024, [Online], Disponible: <https://www.senasa.gob.ar/cadena-animal/equinos/industria/trazabilidad>
- [14] A. L. Manco, "Importancia de la trazabilidad en las certificaciones sostenibles", 2022.
- [15] A. F. Velaverde Santoyo, "Diseño de un sistema de inocuidad alimentaria para la exportación de pasta de ají rocoto a los Estados Unidos bajo la Norma ISO 22000:2005", Trabajo de titulación, Universidad Nacional Pedro Ruiz Gallo, Perú, 2019.
- [16] MySQL, "What is MySQL", 2024, [Online], Disponible: <https://dev.mysql.com/doc/refman/8.0/en/what-is-mysql.html>
- [17] Meta Open Source, "React development", 2024, [Online], Disponible: <https://react.dev/>
- [18] R. S. Pressman, Ingeniería del software un enfoque práctico, Mc Graw Hill, 2010.
- [19] TypeScript, "TypeScript is JavaScript with syntax for types", 2024, [Online], Disponible: <https://www.typescriptlang.org/>
- [20] Figma, "Figma", 2024. [Online], Disponible: <https://www.figma.com>
- [21] Nest.js, "NestJS - A progressive Node.js framework", 2024, [Online], Disponible: <https://docs.nestjs.com/>
- [22] Lucidchart, "¿Qué es un diagrama de flujo? | Lucidchart," Lucidchart, 2024, [Online]. Disponible: <https://www.lucidchart.com>
- [23] J. H. Siller Cepeda, M. A. Báez Sañudo, A. Sañudo Barajas, R. Báez Sañudo, "Manual de buenas prácticas agrícolas", 2022.
- [24] OpenJS Foundation, "About Node.js", 2024, [Online], Disponible: <https://nodejs.org/en/about>
- [25] Nielsen, J., & Landauer, T. K. (1993). A mathematical model of the finding of usability problems. In Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems (pp. 206-213). ACM
- [26] J. Nielsen, "10 Usability Heuristics for User Interface Design," Nielsen Norman Group 2024. [Online]. Disponible: <https://www.nngroup.com/articles/ten-usability-heuristics/>.
- [27] Iconos, Freepik, Flaticon, <https://www.flaticon.com>
- [28] Docker, "What is Docker?", Docker Official Website. <https://www.docker.com/resources/what-container> (accedido: 11-sep-2024).
- [29] Docker, "Overview of Docker Compose," Docker Documentation. <https://docs.docker.com/compose/> (accedido: 11-sep-2024).