****



**Construcción de Software y Toma de Decisiones**

**CoreLink**

Ayetza Yunnuen Infante García A01709011

Fernanda Ríos Juárez A01656047

Lou Parra Camargo A01733551

### 

### 

### 

**14 de octubre, 2024**

**1. Introducción**

#### **1.1 Análisis del Problema**

La industria restaurantera enfrenta un gran desafío en sostenibilidad ambiental, específicamente en reducir su huella de carbono. Según la ONU, el sector alimentario consume aproximadamente el 30% de la energía mundial y es responsable del 22% de las emisiones globales de gases de efecto invernadero (GEI). Aunque hay avances en prácticas sostenibles, los restaurantes aún necesitan mejorar para reducir su impacto ambiental. La falta de herramientas efectivas para medir y gestionar la huella de carbono dificulta la implementación de estrategias para disminuir las emisiones de GEI.

Oxtron, una empresa enfocada en la reducción del impacto ambiental, ha identificado la necesidad de desarrollar un sistema integral para que los restaurantes puedan calcular y visualizar la huella de carbono de sus platillos. Este sistema tiene como objetivo sensibilizar a los restaurantes sobre su impacto en el cambio climático y proporcionarles una herramienta práctica para gestionar y reducir sus emisiones.

#### **1.2 Desarrollo del Problema y Soluciones Propuestas**

Aunque existen tecnologías avanzadas para medir la huella de carbono en diversas industrias, la industria restaurantera carece de soluciones específicas adaptadas a sus necesidades. La mayoría de los restaurantes no tienen los recursos ni el conocimiento para implementar sistemas de medición de carbono sofisticados.

Oxtron propone desarrollar un simulador animado que permita a los restaurantes visualizar la huella de carbono de cada platillo, desde la obtención de ingredientes hasta la preparación final. Este simulador incluirá un sistema de gestión de recetas que permitirá a los usuarios administrar ingredientes, tiempos de cocción y métodos de preparación, considerando su impacto en la huella de carbono. Además, generará reportes detallados para ayudar a los restaurantes a comprender y reducir su impacto ambiental de manera efectiva.

#### **1.3 Alcance del Sistema**

El sistema que se desarrollará incluirá las siguientes funcionalidades clave:

* **Gestión de Recetas:** Los usuarios podrán crear, editar y eliminar recetas, asignando a cada ingrediente un valor específico de huella de carbono basado en datos verificados.
* **Simulación de Preparación:** El sistema permitirá simular la preparación de un platillo, mostrando de manera animada la generación de CO2 en cada etapa del proceso.
* **Generación de Reportes:** Se generarán reportes detallados que incluirán la huella de carbono total generada por cada platillo, así como un resumen de las emisiones diarias, semanales y mensuales del restaurante.
* **Adaptabilidad a Nuevas Recetas:** El sistema será capaz de incorporar nuevas recetas y realizar simulaciones basadas en estas, permitiendo una expansión continua de la base de datos de ingredientes y recetas.

#### **1.4 Limitaciones del Sistema**

* **Dependencia de la Precisión de los Datos:** La precisión de las simulaciones dependerá en gran medida de la exactitud de los datos ingresados en el sistema. Si los valores asignados a los ingredientes o métodos de preparación no son precisos, los resultados de la simulación pueden no reflejar fielmente la realidad.
* **Requerimientos Técnicos:** La implementación del sistema requerirá una infraestructura tecnológica adecuada, incluyendo una base de datos robusta y servidores capaces de manejar el procesamiento de las simulaciones en tiempo real.
* **Uso Limitado a la Industria Restaurantera:** El sistema está diseñado específicamente para restaurantes, lo que limita su aplicabilidad en otros sectores de la industria alimentaria o en otras industrias que también generan huella de carbono.

#### **1.5 Modelo de Ciclo de Desarrollo**

Para el desarrollo del sistema se ha elegido la metodología ágil SCRUM, que facilita una adaptación continua a las necesidades del proyecto y promueve la colaboración estrecha entre todos los stakeholders. SCRUM se basa en ciclos de desarrollo iterativos llamados sprints, donde se entregarán partes del sistema para ser evaluadas y mejoradas según el feedback recibido. Esta metodología asegura que el desarrollo del software sea flexible y capaz de adaptarse rápidamente a cambios en los requisitos o en el mercado, promoviendo la mejora continua del producto y cumpliendo con todos los requerimientos funcionales y no funcionales desde el inicio del proyecto.

### **2. Descripción General**

#### **2.1 Identificación de Stakeholders**

* **Oxtron (Principal Stakeholder):** Oxtron, como la entidad principal para la cual se desarrollará este sistema, juega un rol central en la definición de los requerimientos técnicos y funcionales. Su experiencia en la medición de huella de carbono y su conocimiento del mercado son esenciales para guiar el proceso de desarrollo.
* **Restaurantes y Empresas del Sector Alimentario:** Los principales usuarios del sistema serán los restaurantes y otras empresas del sector alimentario interesadas en reducir su impacto ambiental, ya que necesitan una solución intuitiva y eficaz que les permita gestionar sus recetas y entender cómo cada una contribuye a su huella de carbono global.
* **Gobiernos y Organismos Reguladores:** Dado que la medición de la huella de carbono será obligatoria a partir de 2025, el sistema deberá cumplir con las normativas vigentes y ser capaz de generar reportes que faciliten la auditoría y verificación por parte de las autoridades.
* **Clientes y Consumidores Finales:** Los clientes de los restaurantes también son stakeholders, ya que la implementación de este sistema podría influir en las decisiones de compra basadas conscientemente en la sostenibilidad de los alimentos que consumen.

#### 

#### 

#### **2.2 Identificación de Actores**

* **Chef o Encargado de Cocina:** Este actor interactúa con el sistema principalmente para consultar y ajustar las recetas según las necesidades diarias del restaurante. También puede utilizar el sistema para explorar el impacto de nuevas recetas o ingredientes en la huella de carbono total.

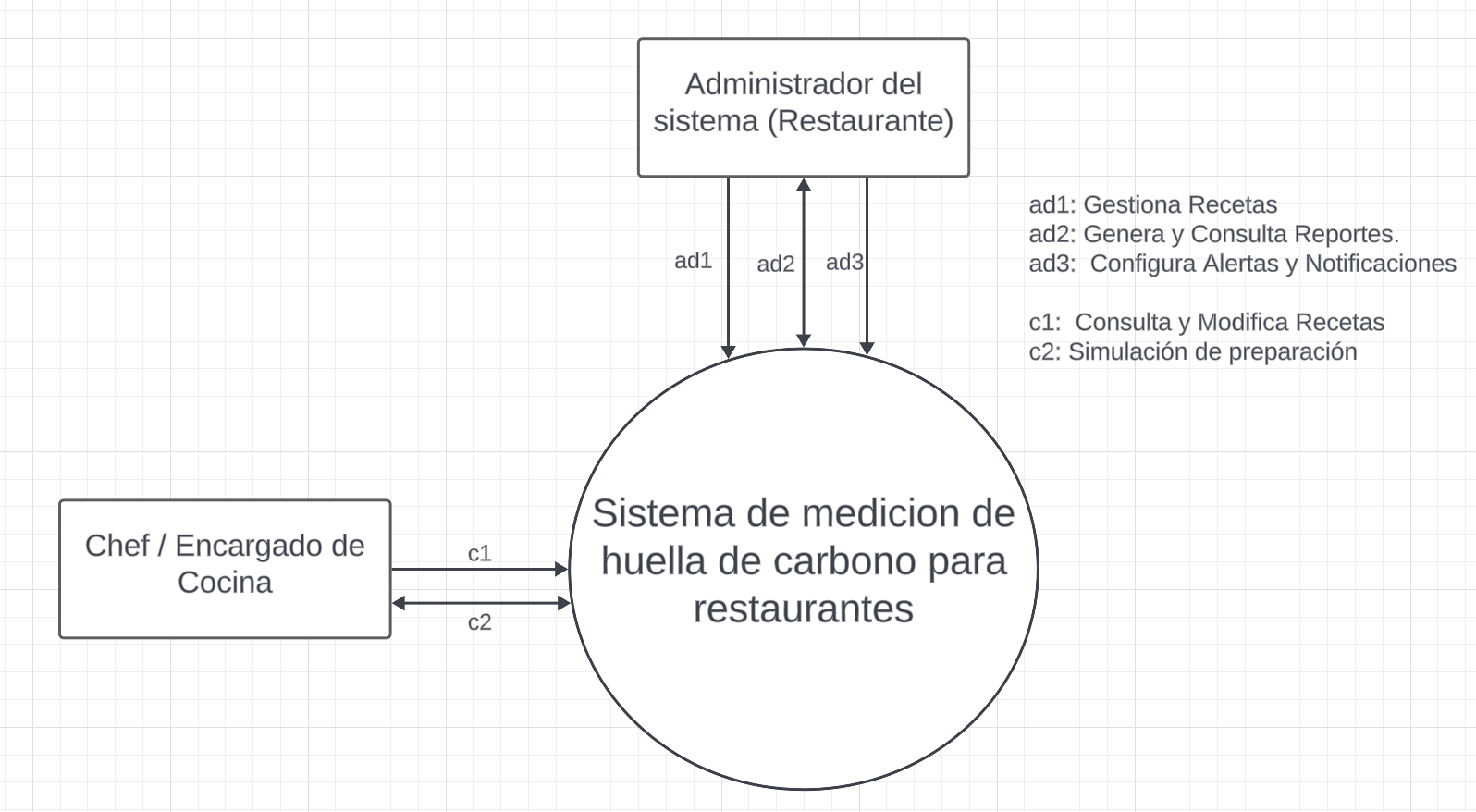
#### **2.3 Requerimientos de Usuario**

* **Gestión de Recetas:** El sistema permite eliminar recetas de manera sencilla. Cada receta debe estar asociada a una lista de ingredientes, cada uno con un valor de huella de carbono predefinido, que se utilizará para calcular el impacto total de la receta.
* **Simulación en Tiempo Real:** Los usuarios deben poder simular la preparación de cualquier receta en tiempo real, observando cómo se genera la huella de carbono en cada etapa del proceso.
* **Generación de Reportes Detallados:** El sistema es capaz de generar reportes de manera detallada mediante gráficas semanales que miden la cantidad de platillos preparados por día y el CO2 total producido tanto en la semana como por día.
* **Interfaz Intuitiva:** La interfaz del sistema debe ser fácil de usar, permitiendo a los usuarios acceder rápidamente a todas las funcionalidades principales.

#### **2.4 Perfiles de Usuario**

* **Chef o Encargado de Cocina:** Este usuario está enfocado en la parte operativa de la cocina. Necesita una herramienta que le permita acceder y modificar recetas rápidamente, así como obtener información sobre el impacto de los ingredientes en la huella de carbono para tomar decisiones informadas sobre el menú.

**Diagrama de contexto**



### **3. Requerimientos Funcionales**

#### **3.1 Gestión de Recetas**

El sistema debe permitir a los usuarios gestionar de manera eficiente las recetas utilizadas en el restaurante.

* **Creación de Recetas:** El sistema proporciona una interfaz intuitiva que permite a los usuarios crear nuevas recetas, seleccionando ingredientes de una base de datos predefinida o agregando nuevos ingredientes. Cada ingrediente seleccionado debe incluir detalles como el origen, el método de producción, y su impacto en la huella de carbono,
* **Eliminación de Recetas:** El sistema permite la eliminación segura de recetas que ya no son necesarias, asegurando que los datos históricos asociados se mantengan para propósitos de auditoría y análisis.

#### **3.2 Simulación de Preparación de Platillos**

El sistema incluye una funcionalidad avanzada a manera de videojuego de simulación para la preparación de platillos en tiempo real. Esta simulación permite a los usuarios visualizar cómo se genera la huella de carbono en cada etapa del proceso de preparación, desde la obtención de los ingredientes hasta la cocción final.

* **Simulación en Tiempo Real:** La simulación muestra de manera animada el impacto de cada ingrediente y método de preparación en la huella de carbono total del platillo. Esto incluye el consumo de energía, la emisión de gases durante la cocción, y otros factores relevantes.
* **Comparación de Recetas:** Los usuarios comparan diferentes recetas y ver cómo varían sus huellas de carbono. Esta funcionalidad es crucial para ayudar a los restaurantes a tomar decisiones informadas sobre qué platillos incluir en su menú.

#### **3.3 Cálculo Automático de la Huella de Carbono**

El sistema es capaz de calcular automáticamente la huella de carbono de cada receta, utilizando algoritmos predefinidos y datos de la base de datos de ingredientes.

* **Algoritmos de Cálculo:** El sistema calcula la huella de carbono de cada ingrediente y receta.
* **Actualización de Datos:** Los cálculos se actualizan en tiempo real cada vez que se modifiquen las recetas o se agreguen nuevos ingredientes.
* **Compatibilidad con Normativas:** El sistema asegura que los cálculos realizados cumplan con las normativas nacionales e internacionales en materia de medición de huella de carbono.

**3.5 Gestión de Usuarios**

El sistema tiene un módulo de gestión de usuarios para definir el acceso a los usuarios con credenciales válidas, garantizando una experiencia personalizada.

* **Seguridad de la Información:** Implementación de mecanismos de autenticación robustos, como contraseñas hash para proteger los datos sensibles.

#### **3.6 Integración con Bases de Datos y APIs**

El sistema es capaz de integrarse con bases de datos externas y APIs que proporcionan datos en tiempo real sobre la huella de carbono de los ingredientes y otros factores relevantes.

* **Sincronización de Datos:** El sistema garantiza que los datos obtenidos a través de bases de datos están sincronizados de manera continua, asegurando que los usuarios siempre tengan acceso a la información más actualizada.

#### **3.7 Interfaz de Usuario (UI) Intuitiva**

El sistema ofrece una interfaz de usuario (UI) intuitiva que permite a los usuarios interactuar con todas las funcionalidades de manera fluida y eficiente.

* **Diseño de la Interfaz:** La UI está diseñada para ser accesible y fácil de usar, incluso para usuarios con poca experiencia en tecnología. Permite una navegación fácil entre las diferentes funcionalidades del sistema.

#### **3.8 Soporte Multidispositivo y Multiplataforma**

* El sistema es compatible con múltiples dispositivos y plataformas.

**3.10 Definición Detallada de Recetas**

Además de la creación y eliminación de recetas, el sistema permite una definición detallada de cada receta

* **Cantidad de Ingredientes:** Al definir una receta, el sistema debe permitir a los usuarios especificar la cantidad exacta de cada ingrediente en unidades estándar (gramos, mililitros, piezas, etc.). Esta cantidad debe influir en el cálculo de la huella de carbono total de la receta.
* **Método de Preparación:** El sistema debe incluir opciones para definir el método de cocción.
* **Tiempo de Preparación:** El sistema permite a los usuarios especificar el tiempo de preparación y cocción de cada receta. El tiempo de cocción debe ser un factor en el cálculo de la huella de carbono, ya que influye en el consumo de energía.

### **3.11 Generación de Reportes Detallados por Periodo**

El sistema es capaz de generar reportes detallados mediante gráficas interactivas las cuales desglosan información semanal ofreciendo una visión clara y detallada de cómo se distribuyen las emisiones.

**Requisitos Funcionales**

***Usuario***

**Iniciar sesión:**

* El administrador inicia sesión en el sistema utilizando credenciales seguras.

**Gestión de recetas:**

* Crear nuevas recetas seleccionando ingredientes de una base de datos predefinida o agregando nuevos ingredientes.
* Eliminar recetas que ya no son necesarias.
* Acceder a la lista de recetas disponibles para su consulta.

**Gestión de ingredientes:**

* Añadir nuevos ingredientes a la base de datos.
* Eliminar ingredientes.
* Simulación de huella de carbono: Ejecutar simulaciones en tiempo real para visualizar el impacto ambiental de la preparación de diferentes platillos.

**Generación de reportes:**

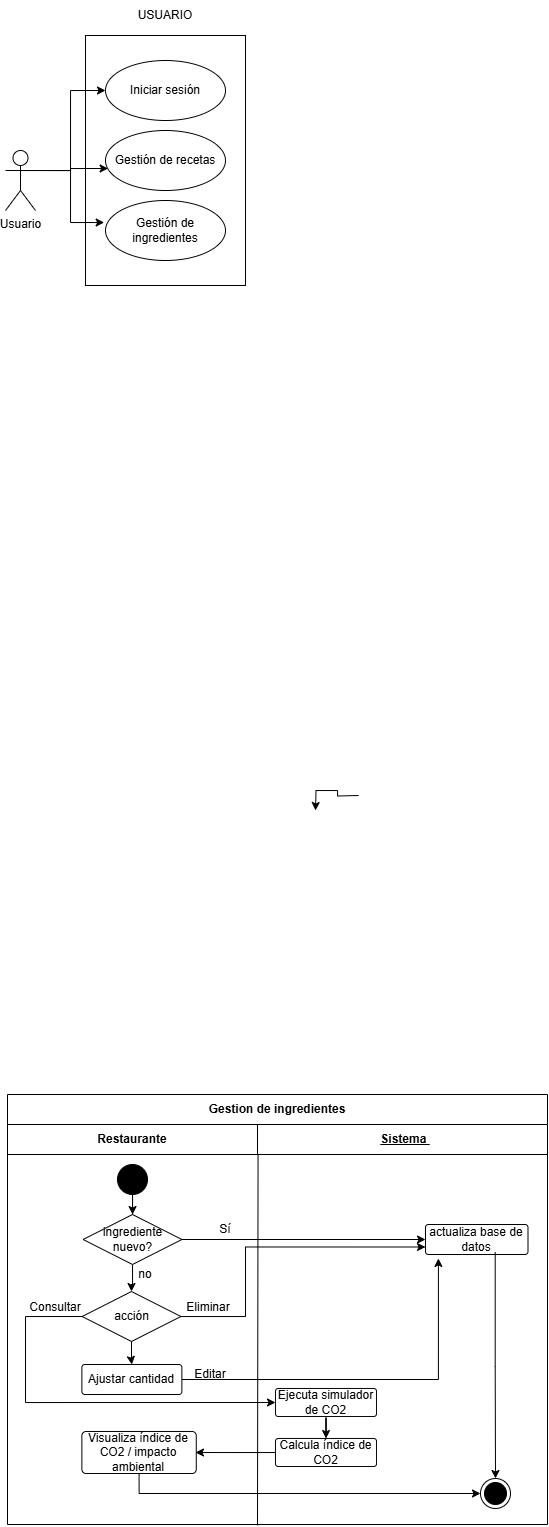
* Generar reportes detallados de huella de carbono semanal (con sección de que se realizó cada día), mediante gráficos interactivos.

**Diagrama de Casos de Uso**

El diagrama de casos de uso que se muestra a continuación incluye las principales funcionalidades del sistema de medición de huella de carbono para restaurantes y cómo interactúan los actores con el sistema; ya que proporciona una visión clara de las funciones disponibles en el sistema de medición de huella de carbono, mostrando cómo los diferentes usuarios interactúan con estas funciones.

Casos de Uso:

* Iniciar sesión: Permite a los actores acceder al sistema utilizando credenciales seguras.
* Gestión de recetas: Los usuarios pueden gestionar (crear y eliminar) recetas en el sistema.
* Gestión de ingredientes: Permite a los usuarios gestionar ingredientes, asignarles valores de huella de carbono y otros detalles.



**Detalle de los casos de Uso**

**CU01: Iniciar sesión**

**Descripción:** Este caso de uso describe el proceso mediante el cual el usuario registrado inicia sesión en el sistema.

**Actores:**

Usuario registrado

**Precondiciones:**

El usuario debe estar registrado en el sistema.

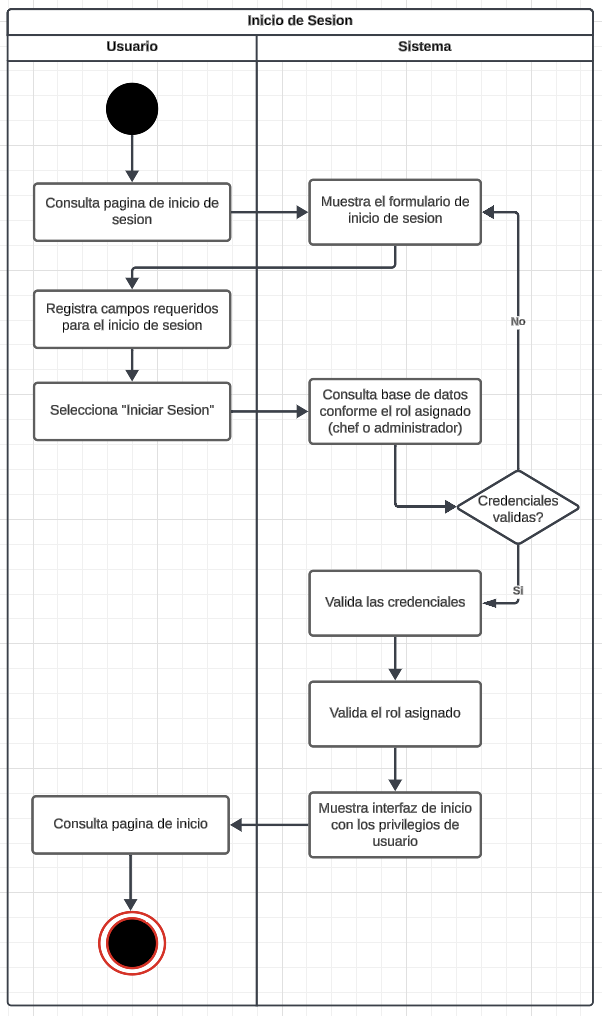
**Flujo Básico:**

* El usuario accede a la página de inicio de sesión.
* El sistema muestra interfaz de inicio de sesión solicitando al usuario que ingrese su número de usuario y contraseña.
* El usuario ingresa su número de usuario y contraseña.
* El usuario selecciona Iniciar Sesión.
* El sistema consulta la base de datos (Tabla Chef) (Usuario y Contraseña).
* El sistema valida las credenciales del usuario.
* Credenciales no válidas, el sistema muestra mensaje de error y redirige a la página principal de Inicio de Sesión.
* El sistema muestra interfaz de inicio con los privilegios de usuario.
* El usuario consulta página de inicio.

**Postcondiciones:**

El usuario inicia sesión en el sistema y se otorgan los privilegios.

**Diagrama de Actividad**



**CU02: Gestión de Recetas**

**Descripción:**

Este caso de uso describe cómo el usuario puede gestionar recetas dentro del sistema, incluyendo la creación de nuevas recetas, y la eliminación de recetas.

**Actores:**

* Chef

**Precondiciones:**

* El usuario debe haber iniciado sesión en el sistema.

**Flujo Básico:**

* El usuario accede a la interfaz de gestión de recetas desde la página de inicio.
* El sistema muestra las opciones de gestión de recetas: Crear nueva receta o Eliminar receta.
* El usuario selecciona la opción "Crear nueva receta".
* El sistema presenta una interfaz para ingresar los detalles de la receta (nombre, descripción, ingredientes, método de cocción).
* El usuario selecciona los ingredientes de una base de datos de ingredientes.
* El usuario selecciona el método de cocción.
* El sistema calcula el índice de CO2 basado en los ingredientes y el método de cocción seleccionados.
* El sistema muestra el índice de CO2 calculado y el impacto ambiental final.
* El usuario revisa la información y selecciona "Guardar receta".
* El sistema guarda la receta en la base de datos.
* El sistema confirma la creación exitosa de la receta.

**Flujo Alternativo:**

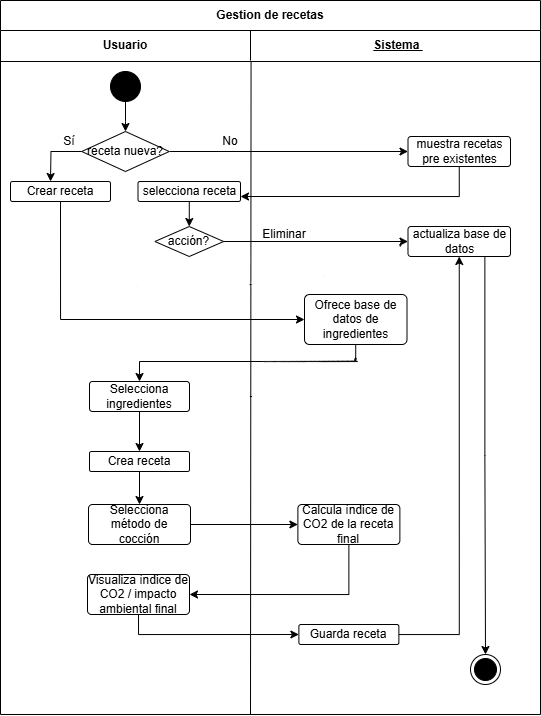
*Eliminar Receta:*

* El usuario selecciona una receta de la lista de recetas existentes.
* El usuario selecciona la opción "Eliminar".
* El sistema pide confirmación para eliminar la receta.
* El usuario confirma la eliminación.
* El sistema elimina la receta de la base de datos y muestra un mensaje de confirmación.

**Postcondiciones:**

* La receta ha sido creada o eliminada exitosamente en el sistema.
* Los cambios se reflejan en la base de datos de recetas.

**Diagrama de Actividad**



**CU03: Gestión de Ingredientes**

**Descripción:**

Este caso de uso describe cómo el usuario puede gestionar los ingredientes en el sistema, incluyendo la creación de nuevos ingredientes, la consulta de ingredientes existentes, la edición de detalles de ingredientes y la eliminación de ingredientes.

**Actores:**

* Chef

Precondiciones:

* El chef debe haber iniciado sesión en el sistema.
* El chef debe tener los permisos necesarios para gestionar ingredientes.

**Flujo Básico:**

1. Iniciar Gestión de Ingredientes:

* El chef accede a la interfaz de gestión de ingredientes desde la página principal del sistema.

*Decisión:* Ingrediente Nuevo.

* El sistema pregunta si el ingrediente que se desea gestionar es nuevo.

1. Si el ingrediente es nuevo:

* El chef selecciona la opción de crear un nuevo ingrediente.
* El sistema presenta un formulario para ingresar los detalles del nuevo ingrediente (nombre, descripción, valor de carbono, estado).
* El chef completa el formulario con los detalles requeridos.
* El chef selecciona "Guardar".
* El sistema guarda el nuevo ingrediente en la base de datos y muestra un mensaje de confirmación de que el ingrediente ha sido creado exitosamente.

1. Si el ingrediente no es nuevo:

* El chef selecciona un ingrediente de la lista de ingredientes existentes.
* El sistema muestra los detalles actuales del ingrediente.

*Decisión:* Acción sobre Ingrediente.

El sistema pregunta qué acción desea realizar el chefr:

1. Consultar:

* El chef revisa los detalles del ingrediente seleccionado sin realizar cambios.
* El sistema permite la visualización de detalles, pero no se realizan modificaciones.
* El chef puede optar por salir de la consulta o realizar otra acción.

1. Eliminar:

* El chef selecciona la opción de eliminar el ingrediente.
* El chef solicita una confirmación para la eliminación del ingrediente.
* El chef confirma la eliminación.
* El sistema elimina el ingrediente de la base de datos y muestra un mensaje de confirmación de que el ingrediente ha sido eliminado.

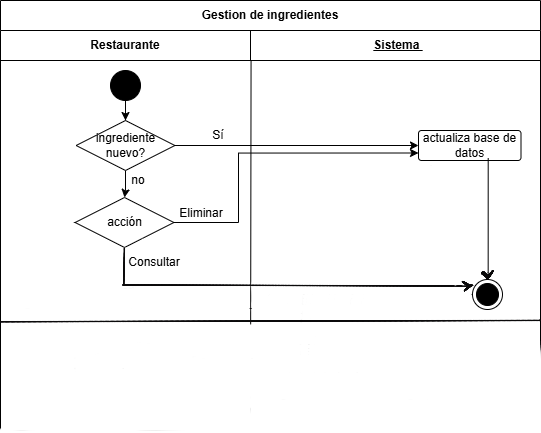
1. Editar:

* El chef selecciona la opción de editar el ingrediente.
* El sistema permite al administrador ajustar la cantidad del ingrediente o editar otros detalles (nombre, valor de carbono, etc.).
* El sistema valida los datos ingresados para asegurar que sean correctos.
* El chef selecciona "Guardar cambios".
* El sistema guarda los cambios en la base de datos.
* El sistema ejecuta una simulación de CO2 basada en los cambios realizados.
* El sistema calcula el nuevo índice de CO2 y muestra el impacto ambiental final basado en los ajustes.
* El sistema muestra un mensaje de confirmación de que los cambios han sido guardados exitosamente.

**Postcondiciones:**

* El ingrediente ha sido creado, consultado, editado o eliminado exitosamente en el sistema.
* Los cambios realizados se reflejan en la base de datos de ingredientes.
* El índice de CO2 y el impacto ambiental se actualizan y muestran correctamente si se realizaron ajustes en los ingredientes.

**Diagrama de Actividad**



**CU04: Generación de Reportes de CO2**

**Descripción:**

Este caso de uso describe el proceso mediante el cual un chef puede generar un reporte semanal del consumo de CO2 basado en los platillos elaborados en el restaurante, consultar datos específicos.

**Actores:**

* Chef

**Precondiciones:**

* El chef debe haber iniciado sesión en el sistema.
* Deben existir datos de consumo de CO2 disponibles en la base de datos para el periodo seleccionado (semana).

**Flujo Básico:**

1. Iniciar Generación de Reporte:

* El chef accede a la interfaz de generación de reportes desde la página principal del sistema.

1. Consulta de Datos:

* El sistema automáticamente consulta la base de datos de la semana para recopilar información relevante.
* El sistema consulta los platillos elaborados en cada día de la semana seleccionada.
* El sistema calcula el total de CO2 emitido por los platillos elaborados en ese periodo.

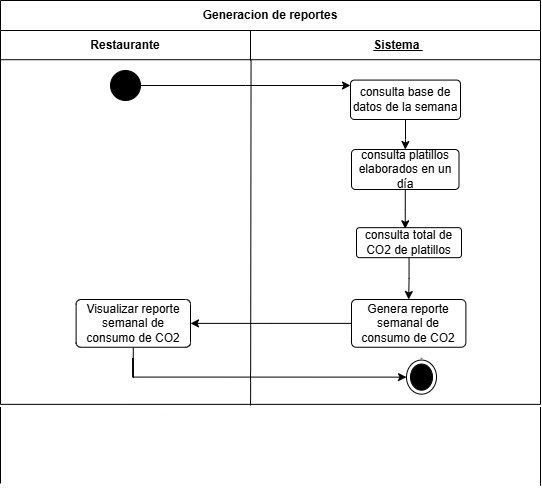
1. Generación de Reporte:

* El sistema utiliza la información recopilada para generar un reporte semanal de consumo de CO2.

1. Visualización del Reporte:

* El sistema presenta al administrador el reporte semanal de consumo de CO2 en la interfaz de usuario.
* El administrador revisa el reporte generado.

**Diagrama de Actividad**



**Modelo Entidad-Relación (MER)**

El MER proporcionado a continuación abarca la gestión de recetas, ingredientes, simulaciones y reportes; los cuales son mencionados en los requisitos funcionales.

****

**Explicación detallada del Modelo Entidad Relación (MER)**

1. ***Entidades Principales***

**Administrador**

Atributos:

* idChef: Identificador único del administrador.
* nombrech: Nombre del administrador.
* apellidopach: Apellido paterno del administrador.
* apellidomach: Apellido materno del administrador.
* correoch: Correo electrónico del administrador.
* contraseñach: Contraseña para el acceso seguro al sistema.
* UUID: Autenticacion del usuario

**Receta**

Atributos:

* idReceta: Identificador único de la receta.
* NombreRe: Nombre de la receta.
* descripcionRe: Descripción detallada de la receta.
* fechaCreacion: Fecha en la que se creó la receta.
* estadoRe: Estado actual de la receta (activa, inactiva, etc.).
* repredeterminada: Recetas globales en el sistema.
* CO2re: Valor de carbono de la receta.
* status: Estado actual de la receta (1 / 0)

Función: Almacena la información de las recetas que se preparan en el restaurante, incluyendo detalles de ingredientes y métodos de cocción.

**Ingrediente**

Atributos:

* idIngrediente: Identificador único del ingrediente.
* nombreIn: Nombre del ingrediente.
* valorcabono: Valor de CO2 del ingrediente.
* unidades: kilogramos (Kg) o litros (L) del ingrediente.

Función: Almacena información sobre los ingredientes utilizados en las recetas y su impacto en términos de emisiones de carbono.

**Simulación**

Atributos:

* idSimulacion: Identificador único de la simulación.
* fecha: Fecha en que se realizó la simulación.
* ResultCarbono: Resultado de la simulación en términos de CO2 emitido.

Función: Realiza simulaciones para calcular la huella de carbono de las recetas, considerando los ingredientes y métodos de cocción.

1. ***Relaciones entre Entidades***

**Chef - Gestiona - Receta**

Relación de tipo N:1, donde un chef puede gestionar múltiples recetas. Los chefs son responsables de crear, modificar y optimizar recetas en función de la sostenibilidad.

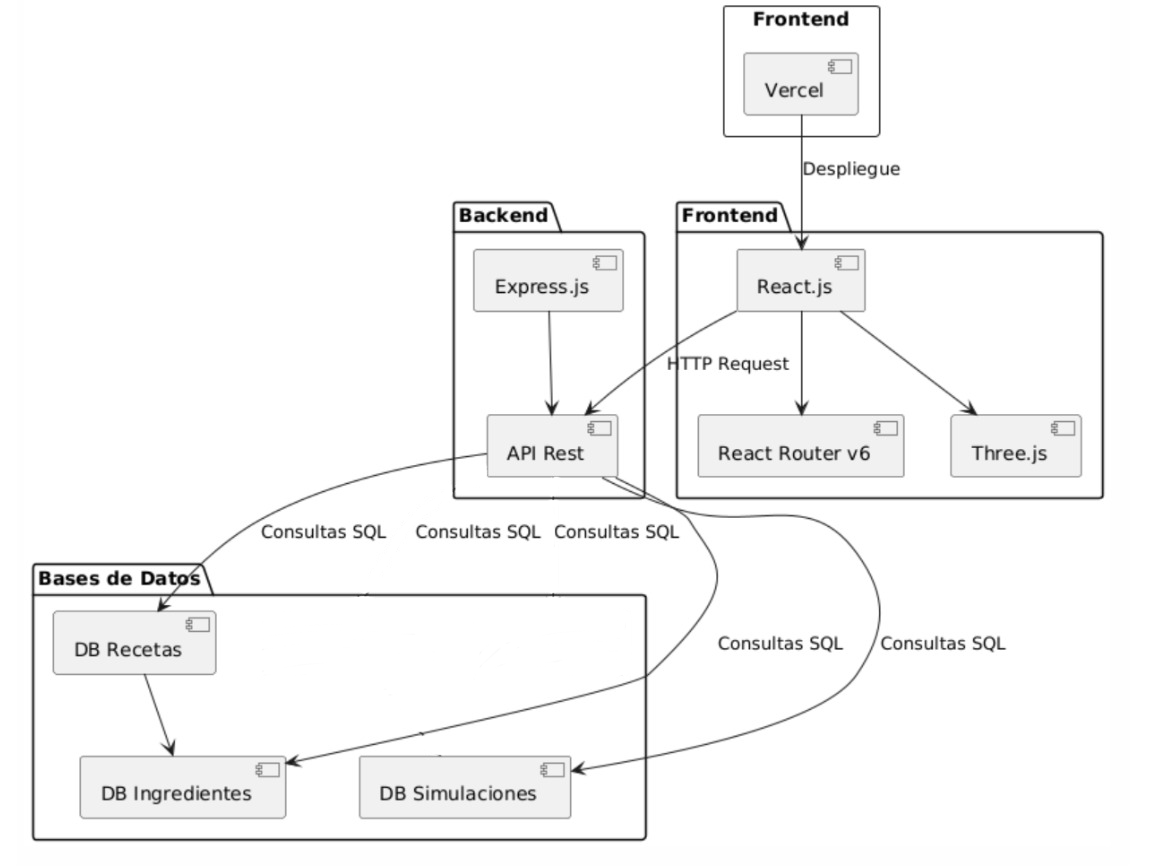
**Receta - Contiene - Ingredientes**

Relación de tipo N (muchos a muchos), donde una receta puede contener varios ingredientes y un ingrediente puede formar parte de múltiples recetas. Esta relación es clave para calcular el impacto de cada receta en términos de huella de carbono.

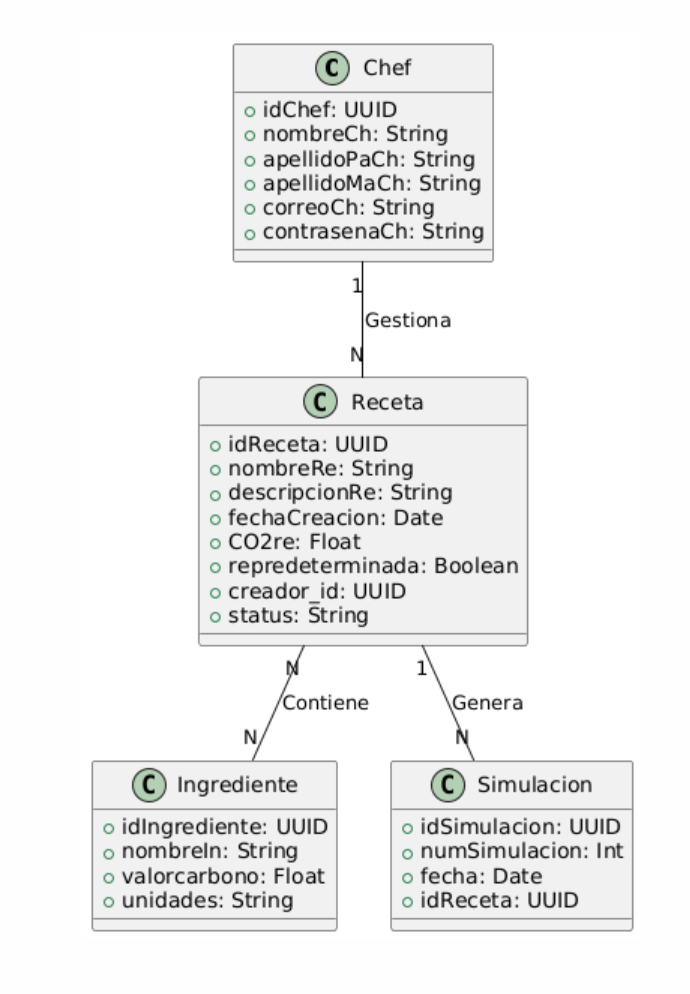
**Receta - Genera - Simulación**

Relación de tipo 1, donde una receta puede tener múltiples simulaciones asociadas. Las simulaciones permiten evaluar diferentes escenarios de preparación y sus efectos en la huella de carbono.

**Diagrama de componentes**

****

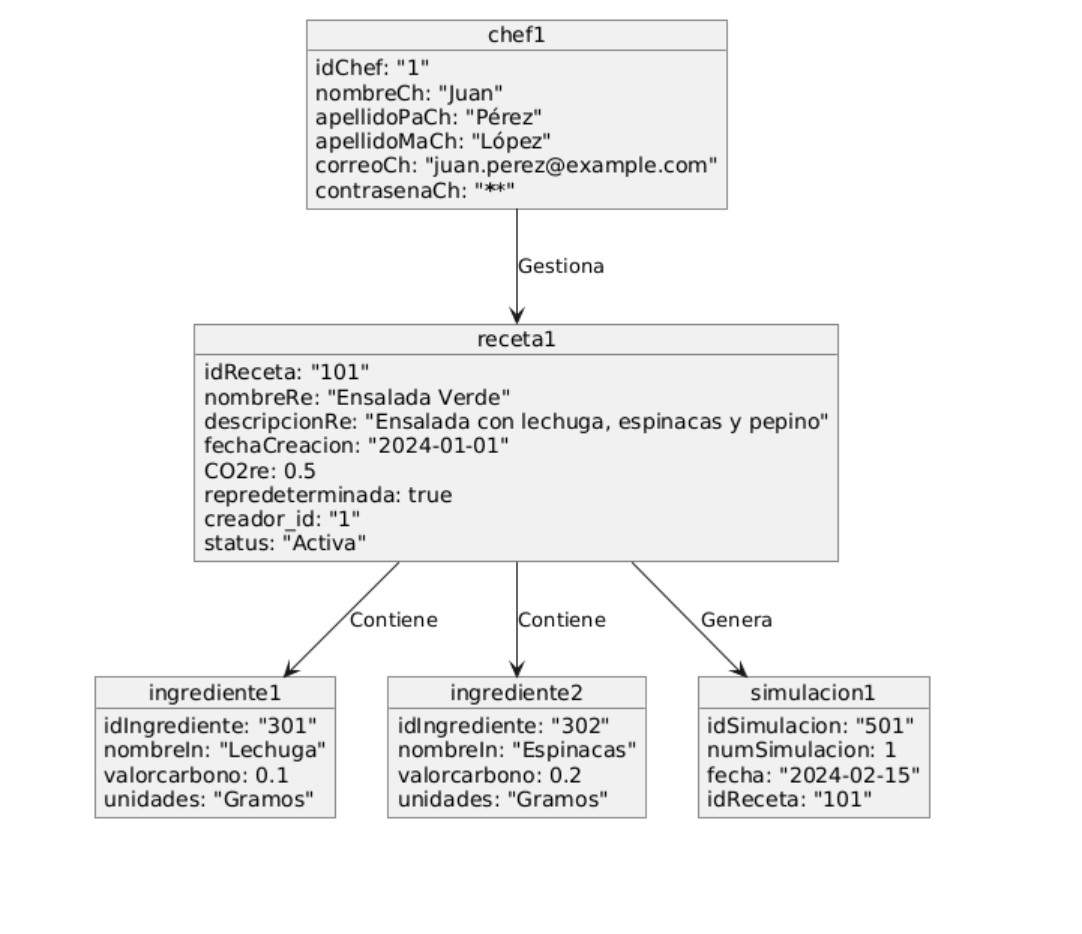
**Diagrama de clases**



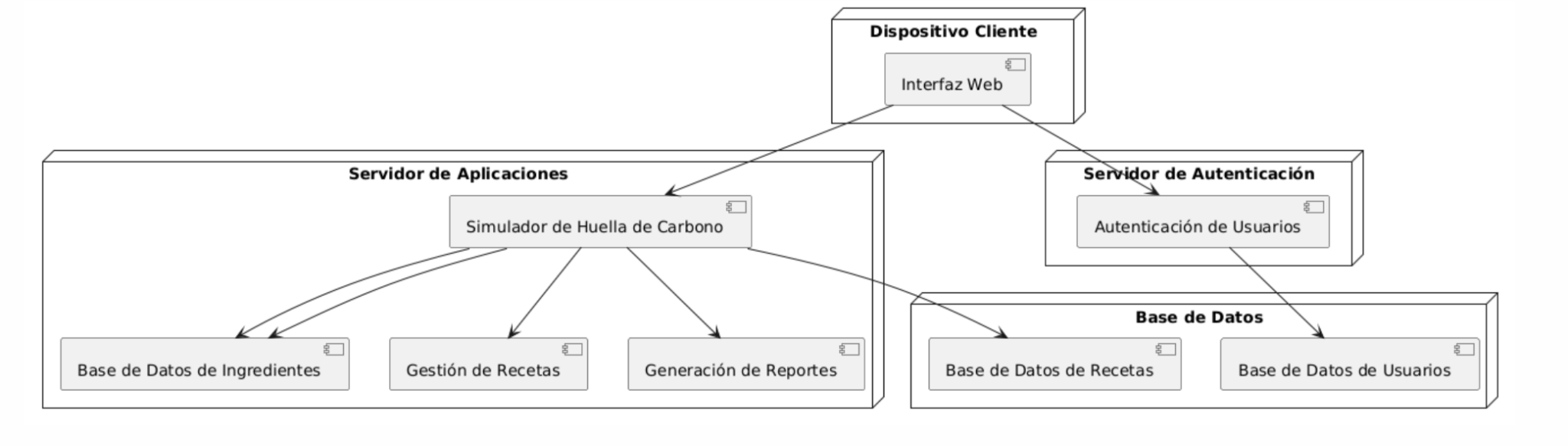
**Diccionario de Datos**

[**diccionario\_de\_datos\_clasificado.xlsx**](https://docs.google.com/spreadsheets/d/17gTnZQCmyXdy8inHL6AP42d5vR5lidqQ/edit?usp=sharing&ouid=113889973898849713701&rtpof=true&sd=true)

**Diagrama de Objetos**

****

**Diagrama de Despliegue**

****

**4. Requerimientos No - Funcionales**

#### **4.1 Rendimiento y Escalabilidad**

El sistema debe ser capaz de manejar múltiples usuarios y grandes volúmenes de datos sin comprometer su rendimiento. Además, debe estar diseñado para escalar de manera efectiva a medida que aumenta la demanda.

* **Tiempo de Respuesta:** El sistema debe garantizar un tiempo de respuesta rápido, no superior a 3 segundos en el 95% de las interacciones, incluso durante picos de uso. Esto es especialmente crucial durante la generación de reportes y la simulación en tiempo real.
* **Escalabilidad Horizontal y Vertical:** El sistema debe ser escalable tanto horizontal como verticalmente. Esto significa que debe ser capaz de añadir más recursos (como servidores) o mejorar los recursos existentes (como CPU y memoria) para manejar un mayor número de usuarios o datos sin pérdida de rendimiento.
* **Manejo de Grandes Volúmenes de Datos:** El sistema debe ser capaz de procesar y almacenar grandes volúmenes de datos relacionados con recetas, ingredientes, y reportes de emisiones de CO₂ sin pérdida de integridad o eficiencia.

#### **4.2 Seguridad y Protección de Datos**

La seguridad es un aspecto fundamental del sistema, especialmente dado que maneja datos sensibles relacionados con la sostenibilidad y las operaciones comerciales de los restaurantes.

* **Autenticación y Autorización:** El sistema debe implementar un mecanismo robusto de autenticación y autorización que asegure que solo los usuarios autorizados puedan acceder a las funcionalidades críticas, como la gestión de recetas y la generación de reportes.
* **Cifrado de Datos:** Todos los datos, tanto en tránsito como en reposo, deben estar cifrados utilizando estándares de cifrado actuales. Esto incluye datos de usuario, recetas, reportes, y cualquier otro dato almacenado en el sistema.
* **Cumplimiento con Normativas de Protección de Datos:** El sistema debe cumplir con las normativas locales e internacionales de protección de datos, como el GDPR en Europa, asegurando que los datos personales de los usuarios estén protegidos y que se respeten los derechos de privacidad.
* **Respaldo y Recuperación de Datos:** El sistema debe incluir mecanismos automáticos de respaldo de datos y planes de recuperación en caso de fallos del sistema, para minimizar la pérdida de datos y el tiempo de inactividad.

#### **4.3 Usabilidad y Experiencia de Usuario**

La usabilidad del sistema es clave para asegurar que los usuarios puedan interactuar con él de manera eficiente y efectiva, sin necesidad de una extensa curva de aprendizaje.

* **Interfaz de Usuario Intuitiva:** El sistema debe tener una interfaz de usuario (UI) intuitiva y fácil de navegar, diseñada de acuerdo con los principios de diseño centrado en el usuario. La interfaz debe permitir a los usuarios realizar tareas complejas con el mínimo esfuerzo.
* **Accesibilidad:** El sistema debe cumplir con los estándares de accesibilidad, como las directrices WCAG 2.1, para garantizar que sea utilizable por personas con discapacidades, incluyendo soporte para lectores de pantalla, colores contrastantes, y navegación con teclado.

#### **4.4 Mantenibilidad y Facilidad de Actualización**

El sistema debe ser diseñado de manera que sea fácil de mantener y actualizar, permitiendo que las mejoras y correcciones se implementen sin interrumpir su funcionamiento.

* **Código Modular y Documentado:** El sistema debe estar desarrollado con una arquitectura modular, donde cada componente esté claramente separado y documentado. Esto facilita la identificación y corrección de errores, así como la implementación de nuevas funcionalidades.
* **Compatibilidad hacia Adelante:** Las actualizaciones y mejoras del sistema deben ser compatibles con versiones anteriores, evitando que las actualizaciones rompan la funcionalidad existente o causen problemas de compatibilidad con datos antiguos.
* **Documentación Completa:** Toda la documentación técnica, desde la arquitectura del sistema hasta las guías de usuario, debe estar completa y actualizada, facilitando el mantenimiento y la capacitación de nuevos desarrolladores.

#### **4.5 Confiabilidad y Disponibilidad**

El sistema debe ser confiable y estar disponible para los usuarios en todo momento, minimizando los tiempos de inactividad y asegurando que esté operativo incluso en situaciones críticas.

* **Alta Disponibilidad:** El sistema debe estar diseñado para ser altamente disponible, con un tiempo de actividad de al menos el 99.9%. Esto puede lograrse mediante la implementación de redundancia en los servidores, balanceo de carga, y mecanismos de failover.
* **Tolerancia a Fallos:** El sistema debe ser tolerante a fallos, lo que significa que debe continuar funcionando de manera efectiva incluso si un componente individual falla. Esto incluye la capacidad de recuperarse automáticamente de fallos sin pérdida de datos o funcionalidad.
* **Monitorización Continua:** El sistema debe ser monitorizado continuamente para detectar y responder rápidamente a cualquier problema de rendimiento o seguridad. Esto incluye el uso de herramientas de monitorización que alerten a los administradores sobre posibles problemas antes de que afecten a los usuarios.

#### **4.6 Interoperabilidad y Compatibilidad**

El sistema debe ser interoperable con otros sistemas y plataformas utilizados por los usuarios, permitiendo una integración fluida en entornos tecnológicos existentes.

* **Compatibilidad con APIs Estándar:** El sistema debe ser compatible con APIs estándar utilizadas en la industria, permitiendo la integración con otras aplicaciones de gestión de restaurantes, contabilidad, y sostenibilidad.
* **Soporte Multiplataforma:** El sistema debe ser compatible con múltiples plataformas, incluidos sistemas operativos como Windows, macOS, Linux, así como dispositivos móviles iOS y Android.
* **Integración con Herramientas de Análisis:** El sistema debe ser capaz de exportar datos a herramientas de análisis de terceros, como Microsoft Power BI o Tableau, permitiendo a los usuarios realizar análisis avanzados de la información generada por el sistema.

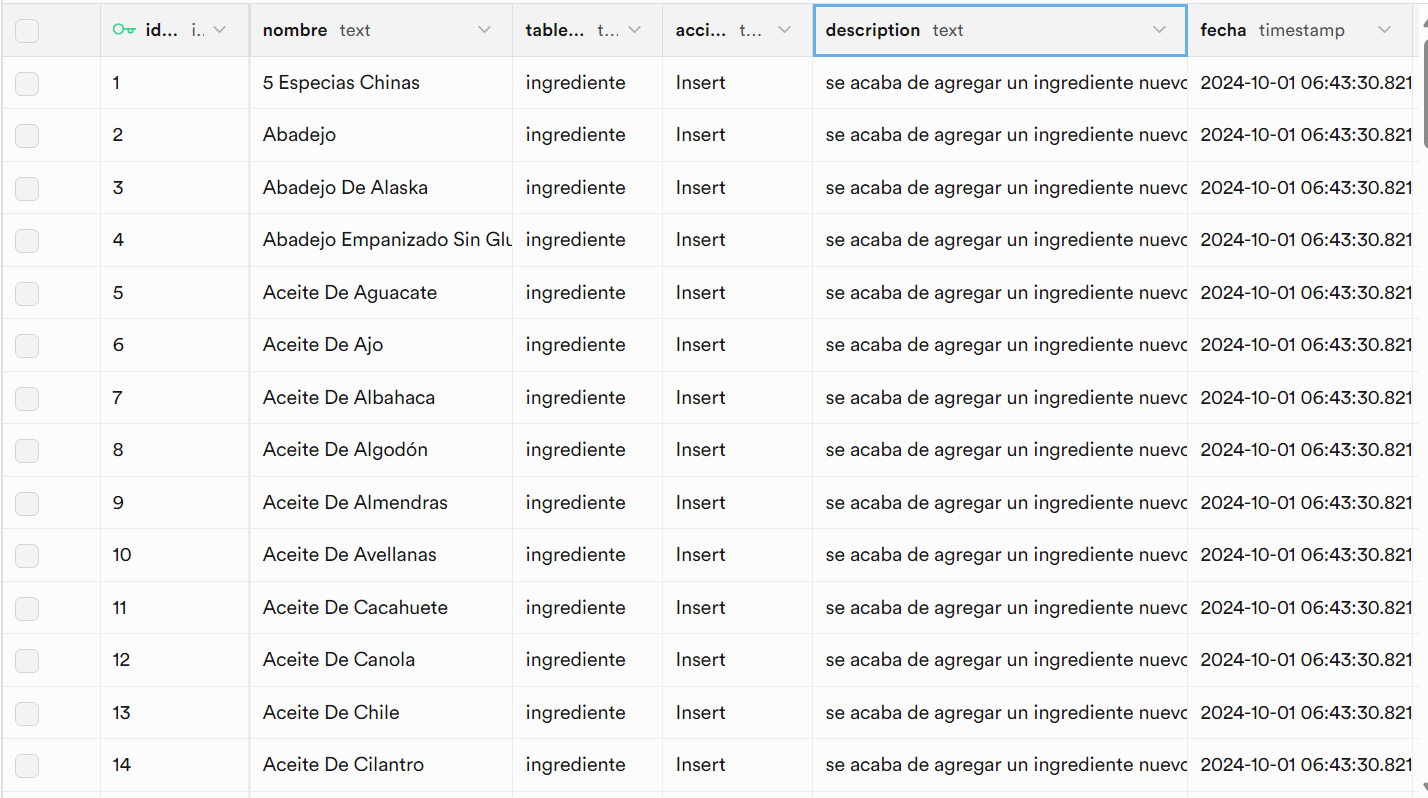
#### **4.7 Responsabilidad Ambiental y Eficiencia Energética**

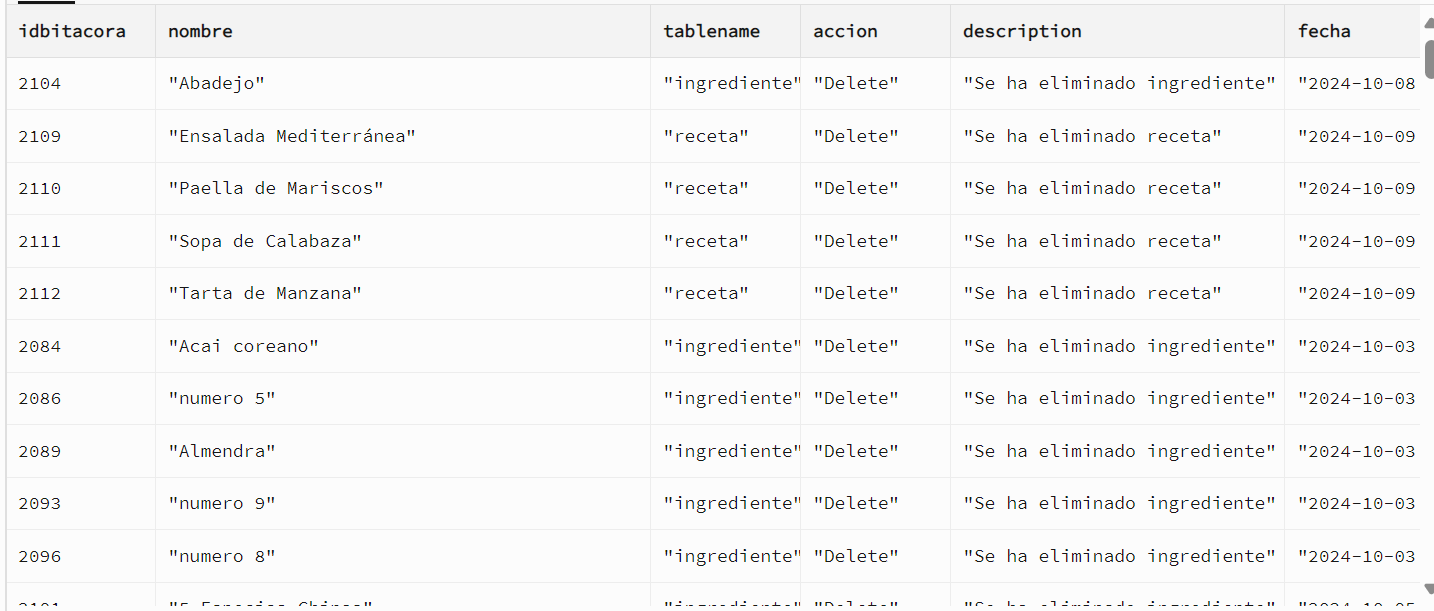
Dado el enfoque del sistema en la sostenibilidad, es crucial que el desarrollo y operación del sistema también reflejen un compromiso con la responsabilidad ambiental.

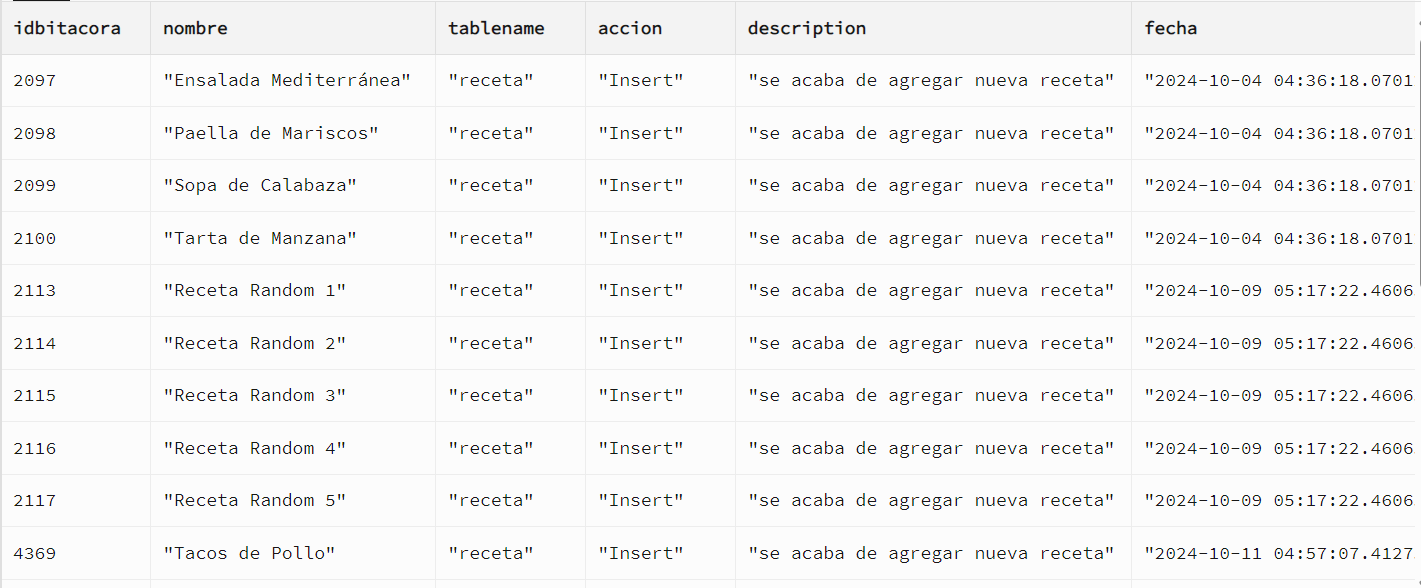
* **Optimización de Recursos:** El sistema debe estar optimizado para consumir la menor cantidad posible de recursos informáticos (CPU, memoria, ancho de banda), reduciendo así su huella de carbono digital.
* **Uso de Energía Renovable:** Siempre que sea posible, el sistema debe ser desplegado en servidores que utilicen energía renovable o que estén ubicados en centros de datos con políticas de sostenibilidad.
* **Impacto Ambiental del Software:** El sistema debe estar diseñado para minimizar su propio impacto ambiental, utilizando técnicas de desarrollo sostenible y priorizando la eficiencia en el uso de recursos.

**Pruebas Unitarias**

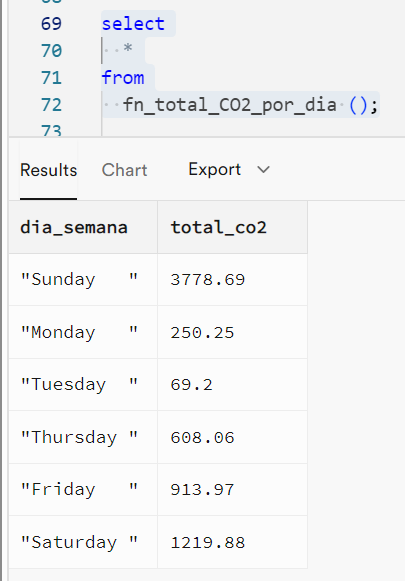
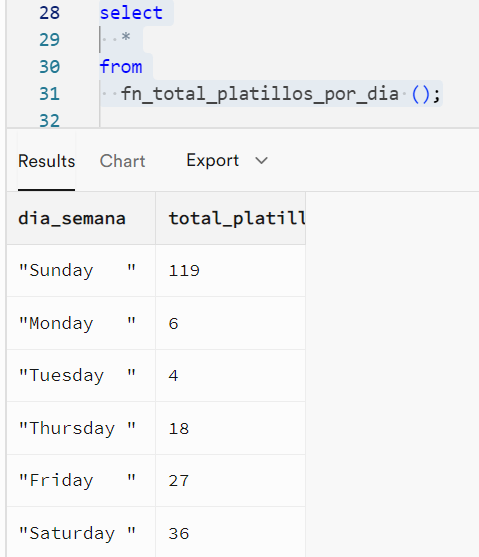
Muestran los cambios y pruebas principales ocurridos dentro de la base de datos como es la inserción y la eliminación de las recetas e ingredientes, las cuales están registradas en una bitácora.

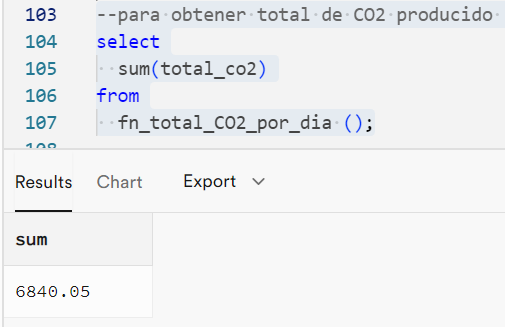
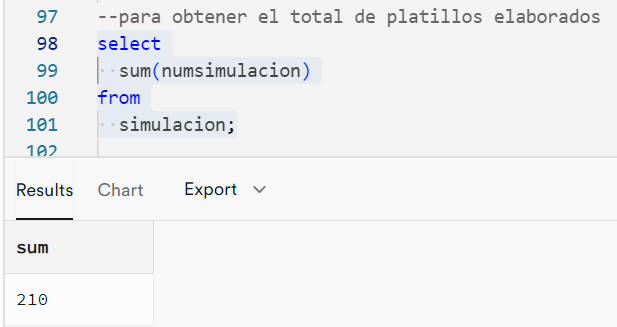




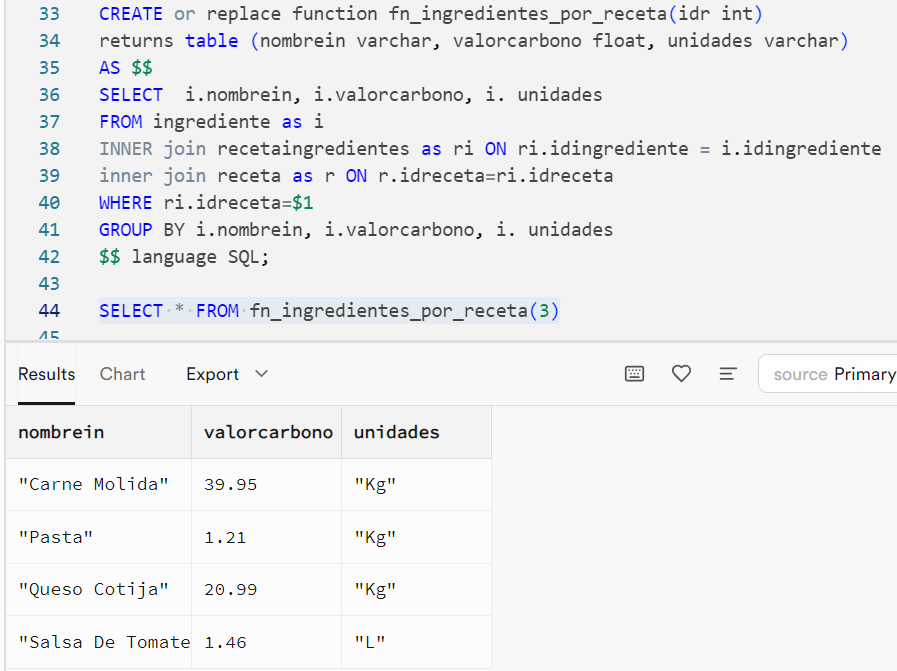


Muestran también cómo funcionan algunas de las funciones generadas para obtener datos específicos.

****

****

****

****

**Pruebas Funcionales**

**Inicio de sesión y Login**

Si el usuario ingresa credenciales válidas lo manda directamente a la página de inicio de sesión, de lo contrario el sistema no le permitirá ingresar.



****

En la vista de inicio de sesión dependiendo del usuario que ingrese le da una bienvenida personalizada basándose en sus credenciales y correo electrónico de la base de datos.





Al igual que se pueden visualizar las gráficas generadas con los datos y especificación de cada uno de la base de datos.

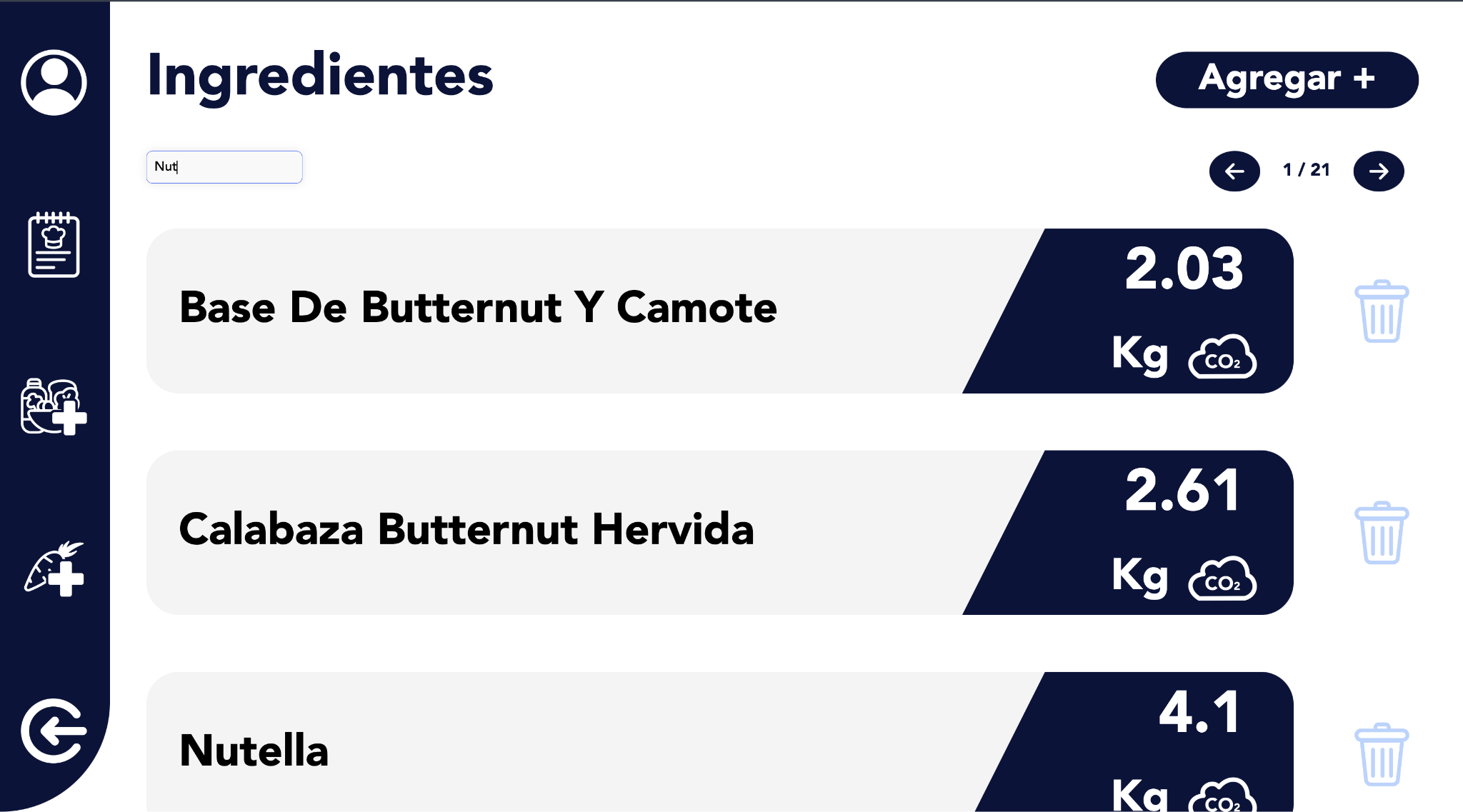


**Ingredientes**

En la vista de ingredientes el sistema muestra todos los datos de la tabla ingredientes de la base de datos.



Tiene search bar para buscar el ingrediente en toda la base de datos (sin importar la paginación), además, al momento de buscar el ingrediente aparecen todos los datos que contengan información similar (como un command F, pero en este caso la búsqueda es global)



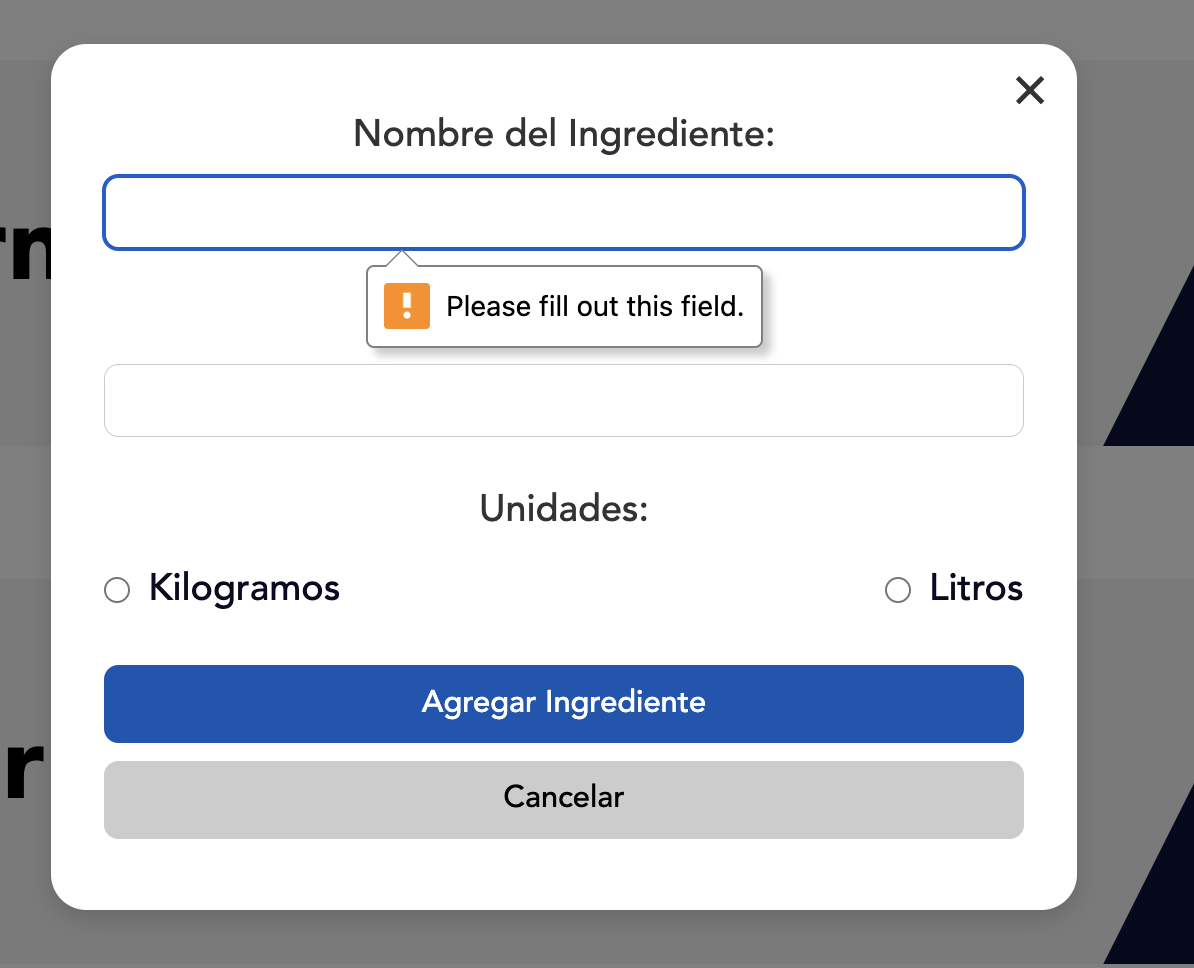
Si un ingrediente no se encuentra en la base de datos no aparece (en este “queso sueco”)



Por lo tanto se puede agregar un nuevo ingrediente rellenando los campos específicos.



De lo contrario no se puede agregar un nuevo ingrediente.



Una vez que se agrega el ingrediente se actualiza en el sistema y en la base de datos.



También se puede eliminar el ingrediente.



**Recetas**

En la vista de recetas se pueden observar todas las recetas predeterminadas de la base de datos.

\

Al igual que si no se ha seleccionado un ingrediente aparece un campo interactivo de selección (sad cat), una vez que se hace la selección (la cual se hace notoria por la transición de la inversión de los colores) aparece la información de cada ingrediente que contiene la receta seleccionada desde la base de datos.



Igualmente tiene la opción de eliminar una receta (como se puede ver, antes habían 3 y ahora 2)



Lo que se solicita en esta sección es que se describa y documente formalmente la licencia de software bajo la cual se distribuirá o utilizará el producto. Además, se requiere una identificación clara de los derechos de autor y una aclaración sobre la autoría y los permisos para el uso de recursos externos, como imágenes, diagramas, componentes, etc.

**Autorizaciones, Licencias y Derechos de Autor**

**Licencia de Software:**

Este software es un producto propietario desarrollado por Ayetza Yunnuen Infante Garcia y Fernanda Rios Juarez y está protegido por las leyes de derechos de autor. Queda prohibido su uso, modificación, copia, distribución o ingeniería inversa sin autorización previa y expresa del titular de los derechos de autor.

**Derechos de Autor:**

El software y su código fuente son propiedad de

Ayetza Yunnuen Infante Garcia

Fernanda Rios Juarez

© 2024. Todos los derechos reservados.

El uso no autorizado del software, en parte o en su totalidad, está prohibido y puede derivar en sanciones legales.

**Recursos Externos:**

Las imágenes utilizadas en la interfaz gráfica son de [Fuente], bajo la licencia [tipo de licencia].

Los diagramas fueron creados utilizando LucidChart, PlantUML y por individual; y están sujetos a las condiciones de uso de dicha herramienta.

El software utiliza React, un framework de JavaScript licenciado bajo la licencia MIT. React es desarrollado y mantenido por Meta Platforms, Inc. y su uso en este proyecto está permitido bajo los términos de dicha licencia.

Otros componentes utilizados en el software pueden estar sujetos a licencias adicionales, que han sido debidamente respetadas.

Licencia de React:

* React (c) Meta Platforms, Inc.
* Licenciado bajo la Licencia MIT.