



# **Fuego y Pan**

**Yeray Ventura Núñez  
German Alvarenga Erroa  
Diego Pertíñez Cabo**

# Índice

<b>Descripción.....</b>	<b>3</b>
<b>Justificación.....</b>	<b>3</b>
<b>Alcance.....</b>	<b>4</b>
<b>Alternativas existentes en el mercado.....</b>	<b>4</b>
<b>El stack tecnológico elegido está detallado para que podamos probar los avances.....</b>	<b>5</b>
<b>Objetivos.....</b>	<b>5</b>
<b>Objetivo general.....</b>	<b>5</b>
<b>Objetivos específicos.....</b>	<b>5</b>
<b>Requisitos del sistema.....</b>	<b>6</b>
<b>Requisitos funcionales:.....</b>	<b>6</b>
<b>Requisitos no funcionales:.....</b>	<b>6</b>
<b>Casos de uso.....</b>	<b>6</b>
<b>Modelo y diseño cerrado de la base de datos.....</b>	<b>7</b>

## Descripción

El trabajo tiene como objetivo principal el diseño, desarrollo e implementación de una aplicación para gestionar el inventario en una hamburguesería. Con este proyecto buscamos digitalizar la manera de controlar el stock de los ingredientes de los alimentos del establecimiento que se hace manualmente mediante una libreta de forma que puede llevar a muchos errores, pérdida de tiempo y una mala planificación de las compras.

Nuestra solución propone una aplicación multiplataforma que será accesible desde móviles que permite a los camareros registrar las comandas de forma rápida y sencilla. Cada vez que confirme una comanda y automáticamente reducirá el inventario de los ingredientes que se necesite para elaborar cada plato, según las cantidades definidas previamente para cada receta que definió el gerente. Por ejemplo, si un cliente pide una "Hamburguesa Clásica", el sistema identificará que dicha hamburguesa requiere 1 pan de brioche, 120 gramos de carne de ternera, 20 gramos de lechuga, 15 gramos de tomate, 10 gramos de cebolla y 15 gramos de queso cheddar. Estas cantidades se restará del stock actual de cada ingrediente en tiempo real.

Además, la aplicación incluirá un sistema de alertas que notificará al gerente cuando el stock de algún ingrediente esté por debajo de una cantidad predefinida. Esto permite reponer a tiempo ante que se acabe este producto, evitando que se no haya stock suficiente en el servicio así también llevando mejor un control.

## Justificación

Este proyecto se basa en la necesidad de modernizar la gestión de stock en pequeños establecimientos de hostelería, como la hamburguesería Oculto. Actualmente, muchos locales dependen de métodos manuales como apuntes de una libreta o en una hoja de excel para llevar el control de inventario, lo que genera varios problemas:

- Riesgo de stock vacío: Sin alertas automáticas, es muy fácil quedarse sin ingredientes necesarios en mitad del servicio.
- Compras innecesarias: Algunas veces se compra ingredientes de memoria o por creencias y no por datos reales, lo que puede generar exceso o escasez.

Una solución digital, económica y fácil de usar permite a estos negocios mejorar su eficiencia, reducir costos y ofrecer un mejor servicio al cliente. Además, al centrarse exclusivamente en apuntar las comandas y su impacto en el inventario, la aplicación evita caer en sistemas genéricos, adaptándose perfectamente a las necesidades de una hamburguesería.

# Alcance

Funcionalidades incluidas:

- Autenticación de usuarios con dos roles: *camarero* y gerente.
- Gestión de ingredientes por parte del administrador (nombre, unidad de medida, stock actual, stock mínimo).
- Registro de comandas por parte de los camareros, con confirmación automática de deducción de stock.
- Sistema de alertas visuales cuando un ingrediente cae por debajo del stock mínimo por mensaje.
- Sistemas de alertas cuando un producto está próximo a caducar
- Interfaz intuitiva, optimizada para uso en dispositivos táctiles.
- Generar Informes de desperdicios y productos más vendidos

## Alternativas existentes en el mercado

En el mercado existen varias soluciones de gestión para restaurantes, pero la mayoría están diseñadas para grandes cadenas o tienen limitaciones importantes para pequeños negocios como "Oculto" y no utilizan la integración del mensaje cuando el stock es bajo:

- Loyverse POS: Ofrece gestión de inventario y ventas, pero su versión gratuita tiene límites en funciones y reportes. Además, su lógica de stock no permite definir recetas con ingredientes medidos en gramos o mililitros de forma intuitiva.
- SumUp POS: Se centra más en el cobro que en la gestión de inventario profunda. No automatiza la reducción de ingredientes por comanda.
- Toast (EE.UU.): Potente, pero orientado a grandes restaurantes, con costos mensuales elevados y configuración compleja.
- Hoja de cálculo (Excel): Usada por muchos pequeños negocios, pero no es automatizada, es propensa a errores y no genera alertas en tiempo real.

Ninguna de estas alternativas ofrece una solución gratuita, específica para hamburgueserías pequeñas. Nuestra aplicación llena ese vacío al centrarse exclusivamente en pequeños comercios y en una hamburguesería, con una interfaz simple, sin costos de licencia y con un modelo de datos adaptado a lo que realmente se gasta por gramos o unidades. Esto la convierte en una opción más eficiente, económica y práctica para los negocios pequeños.

# El stack tecnológico elegido está detallado para que podamos probar los avances.

Utilizaremos React Native + Tailwind CSS para crear la aplicación, utilizaremos la librería nativewind y con el typescript haremos el backend de esta misma.

Para las aplicaciones nos apoyaremos de una API externa para WhatsApp ( Estamos considerando Twilio pero en caso de dar fallos de implementación o que no sea tan moldeable como buscamos y podríamos cambiar a otra).

Para la base de datos utilizaremos MySQL ya que al ser un par de bases de datos y necesitar que interactúen entre ellas de manera frecuente elegiremos una base de datos relacional.

## Objetivos

### Objetivo general

Gestionar el stock de productos en tiempo real mediante notificaciones mediante la aplicación de whatsapp avisando cuando el inventario alcance niveles críticos o los productos estén próximos a caducar.

### Objetivos específicos

- Descuento automático de stock de ingredientes al confirmar una venta.
- Notificar al gerente cuando el stock alcance el porcentaje mínimo.
- Enviar alertas cuando un ingrediente del stock esté a punto de expirar para priorizar su uso.
- Informar cuando un producto esté caducado para retirar del stock.
- Permitir al gerente consultar el stock en cualquier momento.
- Facilitar el reabastecimiento del inventario.
- Implementar el método de gestión FIFO, es decir primero en entrar, primero en salir.
- Generar informes automáticos de productos más vendidos y fechas de caducidad.

# Requisitos del sistema

## Requisitos funcionales:

- El sistema debe permitir la autenticación de usuarios, mediante el usuario y contraseña.
- Al registrar una comanda, el sistema debe descontar automáticamente del inventario los productos asociados.
- El sistema debe priorizar los productos más próximos a caducar.
- El gerente debe poder consultar el inventario completo con cantidades, porcentajes y fechas de caducidad.
- El gerente debe poder configurar el porcentaje mínimo de stock de cada producto.
- El gerente debe poder actualizar el inventario al reponer productos.
- El sistema debe enviar una notificación vía WhatsApp al gerente cuando un producto alcance el porcentaje mínimo configurado.
- El sistema debe notificar cuando un producto está por caducar.
- El sistema debe enviar una alerta cuando un producto esté ya caducado e indicar tirarlo.
- El camarero y el gerente deben poder modificar manualmente el inventario para reflejar el stock real al final del día.
- El sistema debe disponer de un servicio automático que compruebe diariamente la fecha actual y revise los productos caducados.
- El sistema debe generar informes de los productos más vendidos.
- El sistema debe generar informes sobre productos próximos a caducar y productos caducados.
- El gerente debe poder consultar los informes por fecha o por rango de fechas.

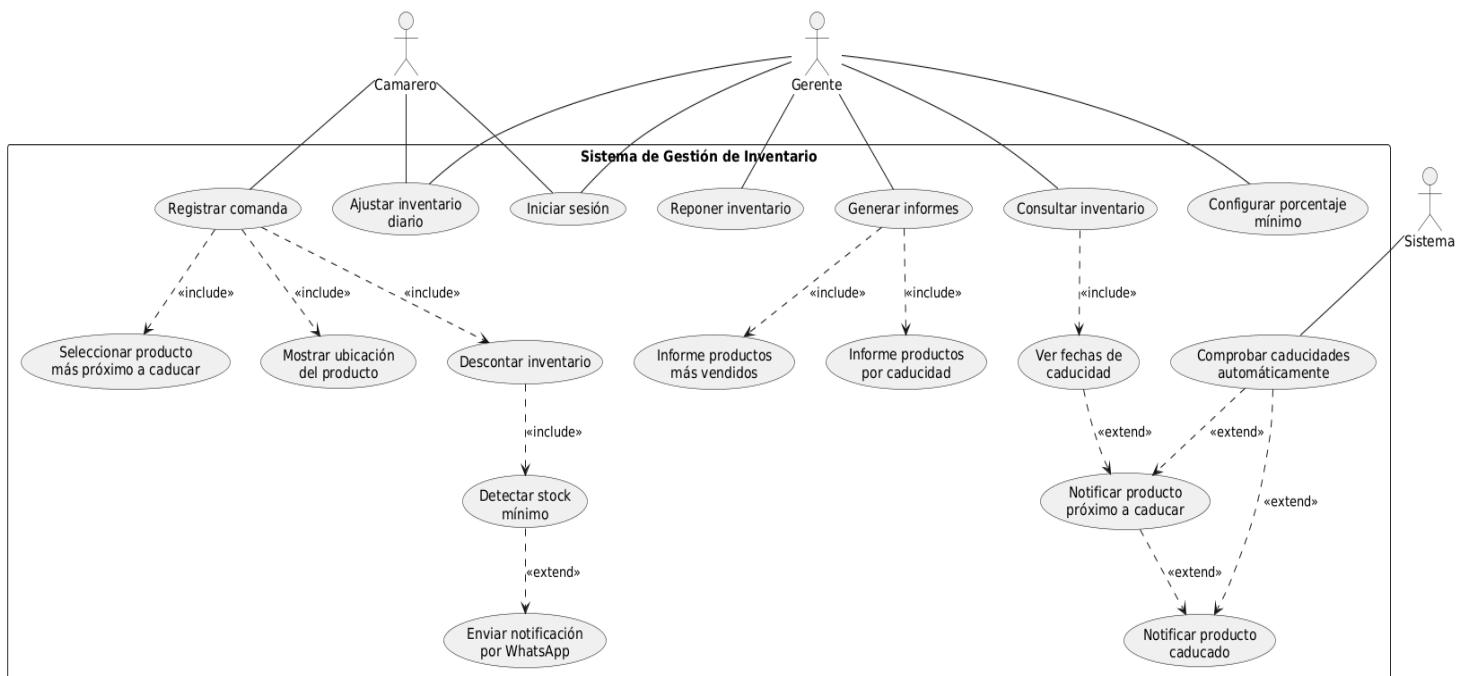
## Requisitos no funcionales:

- El sistema debe responder a las peticiones en menos de 2 segundos.
- El sistema debe permitir el uso simultáneo por varios camareros.
- Las contraseñas deben guardarse de forma cifrada.
- Se debe poder acceder al sistemas desde un móvil o una tableta.
- El inventario se debe actualizar en tiempo real.
- Las notificaciones del stock a punto de caducar deben ser consistentes y de forma clara.
- El sistema debe ejecutar automáticamente los procesos de comprobación de caducidad sin intervención del usuario.
- El sistema debe utilizar triggers en la base de datos para garantizar la coherencia del inventario ante cambios en los datos.

## Requisitos Interfaz:

- La aplicación tendrá tres pantallas principales.
  - La primera pantalla será la de inicio de sesión, donde el usuario introducirá sus credenciales.
  - Una vez autenticado, el sistema redirigirá al usuario según su rol:
  - Si es camarero, accederá a la pantalla de registro de comandas, donde podrá:
    - Seleccionar productos del menú.
    - Confirmar pedidos.
  - Si es gerente, accederá a la pantalla de gestión de stock, desde la cual podrá:
    - Consultar los niveles actuales de ingredientes.
    - Realizar acciones de restock.
  - En ambas pantallas posteriores al inicio de sesión se incluirá:
    - Una barra de navegación o menú lateral.
    - Opción para cerrar sesión en cualquier momento.
    - Posibilidad de regresar al menú principal correspondiente a su rol.

# Casos de uso



# Descripción de casos de uso:

## CU-001 Registrar comanda

Versión: 1.0 (16/02/2026)

### Dependencias:

- RG-001 Gestionar inventario
- RG-002 Gestionar productos próximos a caducar
- RN-001 Fecha de caducidad de los productos

### Precondición:

El usuario debe estar registrado y haber iniciado sesión en el sistema.

### Descripción:

El sistema permite al camarero registrar las comandas realizadas por los clientes. Al registrar una comanda, se descuentan automáticamente los productos del inventario, se priorizan los productos próximos a caducar y se muestra la ubicación de cada producto.

### Secuencia normal:

Paso	Acción
1	El camarero solicita al sistema comenzar el proceso de registrar la comanda.
2	El camarero selecciona los productos que forman parte de la comanda.
3	El sistema identifica los productos seleccionados y verifica las unidades disponibles en el inventario.
4	El sistema selecciona automáticamente los productos próximos a caducar.
5	El sistema descuenta las unidades correspondientes del inventario.
6	El sistema verifica si algún producto alcanza el stock mínimo.

7	Si algún producto alcanza un stock mínimo, el sistema envía una notificación al gerente.
8	El sistema confirma al camarero que la comanda ha sido registrada correctamente.

**Postcondición:**

La comanda queda registrada en el sistema y el inventario se actualiza automáticamente.

**Excepciones:**

<b>Paso</b>	<b>Acción</b>
E.1	Si algún producto no tiene unidades suficientes en el inventario, el sistema informa al camarero y no permite registrar ese producto.
E.2	Si ocurre un error de conexión con la base de datos, el sistema notifica al usuario y no registra la comanda hasta que se resuelva el problema.

## CU-002 Reponer inventario

**Versión:** 1.0 (16/02/2026)

**Dependencias:**

- RG-001 Gestionar inventario
- RN-001 Fecha de caducidad de los productos
- RN-002 Umbral mínimo de stock por producto

**Precondición:**

El gerente debe estar autenticado en el sistema y haber recibido una notificación de stock mínimo o haber accedido manualmente al módulo de gestión de inventario.

**Descripción:**

El sistema permite al gerente registrar la reposición de productos en el inventario tras una compra o entrega de proveedores, actualizando las cantidades y fechas de caducidad. Durante la reposición, el sistema valida que no se dupliquen lotes o fechas erróneas, y registra el movimiento para trazabilidad.

#### **Secuencia normal:**

Paso	Acción
1	El sistema muestra al gerente una lista de productos con stock bajo (o permite buscar productos específicos).
2	El gerente selecciona el producto a reponer y especifica la cantidad y lote (opcional).
3	El sistema solicita la fecha de caducidad del nuevo lote (si aplica) y verifica que sea posterior a la fecha actual.
4	El sistema valida que el producto exista en catálogo y que no haya errores en formato (ej. fecha inválida, cantidad $\leq 0$ ).
5	El sistema registra el movimiento en el historial de inventario (producto, cantidad, fecha entrada, lote, proveedor — si se indica).
6	El sistema actualiza el stock disponible del producto.
7	Si el producto tenía alerta de stock mínimo, el sistema cancela dicha alerta.
8	El sistema confirma al gerente que la reposición se ha realizado correctamente.

#### **Postcondición:**

El inventario del producto se actualiza, se registra el movimiento y se elimina cualquier alerta activa de stock mínimo para ese producto.

#### **Excepciones:**

Paso	Acción
E.1	Si la fecha de caducidad es anterior a la fecha actual, el sistema rechaza la reposición y solicita corregirla.
E.2	Si la cantidad ingresada es $\leq 0$ , el sistema muestra error y no permite continuar.
E.3	Si el producto no existe en el catálogo, el sistema sugiere crearlo (opcional) o cancela la operación.

## CU-003 Consultar inventario y estado de caducidad

**Versión:** 1.0 (16/02/2026)

### Dependencias:

- RG-001 Gestionar inventario
- RN-001 Fecha de caducidad de los productos
- RN-003 Plazo de alerta (ej. 3 días antes de caducar = “próximo a caducar”)

### Precondición:

El usuario (gerente o camarero con permisos) debe estar autenticado.

### Descripción:

El sistema permite consultar el estado actual del inventario, incluyendo niveles de stock, fechas de caducidad próximas y productos ya caducados. El sistema también puede mostrar alertas automáticas generadas por el proceso de verificación diaria.

### Secuencia normal:

Paso	Acción
1	El usuario inicia sesión y accede al módulo Consultar inventario.
2	El sistema carga y muestra una tabla con todos los productos activos, mostrando: nombre, stock actual, fecha de caducidad más próxima, estado (normal / próximo a caducar / caducado).
3	El usuario puede filtrar por categoría, estado (ej. “próximo a caducar”), o rango de fechas.

4	El sistema resalta visualmente los productos próximos a caducar ( $\leq 3$ días) y los caducados (fecha $\leq$ hoy).
5	El usuario puede hacer clic en un producto para ver su historial de entradas/salidas y lotes.
6	Si el usuario selecciona Ver alertas automáticas, el sistema muestra las notificaciones generadas por el proceso Comprobar caducidades automáticamente.
7	El sistema registra la consulta (para auditoría), pero no modifica datos.

**Postcondición:**

El usuario obtiene información actualizada y filtrable del inventario, sin alterar el estado del sistema.

**Excepciones:**

Paso	Acción
E.1	Si el sistema no puede acceder a la base de datos, muestra mensaje de error y ofrece reintentar.
E.2	Si no hay productos registrados, muestra mensaje informativo y opción para ir a Gestionar productos.

## CU-004 Generar informes de inventario

**Versión:** 1.0 (16/02/2026)

**Dependencias:**

- RG-001 Gestionar inventario
- RN-001 Fecha de caducidad de los productos
- RN-002 Umbral mínimo de stock por producto

**Precondición:**

El gerente debe estar autenticado en el sistema.

**Descripción:**

El sistema genera informes personalizados: productos más vendidos, por caducidad, y niveles de stock.

**Secuencia normal:**

Paso	Acción
1	El gerente selecciona tipo de informe: a) Productos más vendidos (últimos 7/30 días), b) Productos por caducidad, c) Stock bajo.
2	El sistema solicita rango de fechas (si aplica).
3	El sistema procesa datos y genera informe en formato tabular (PDF/Excel).
4	El usuario puede previsualizar o descargar.
5	El sistema registra generación del informe (fecha, usuario, tipo).

**Postcondición:**

Informe generado y disponible para descarga.

**Excepciones:**

Paso	Acción
E.1	Rango inválido → ajusta automáticamente al válido.
E.2	Sin datos en rango → informe vacío con mensaje.
E.3	Error de exportación → notifica y ofrece reintentar.

## CU-005 Configurar porcentaje mínimo de stock

**Versión:** 1.0 (16/02/2026)

**Dependencias:**

- RG-001 Gestionar inventario
- RN-002 Umbral mínimo de stock por producto

**Precondición:**

El gerente debe estar autenticado; el sistema debe tener productos registrados.

**Descripción:**

El gerente define el umbral mínimo (en % o unidades) para cada producto o por categoría, usado para activar alertas.

**Secuencia normal:**

Paso	Acción
1	El gerente accede a Configurar porcentaje mínimo.
2	El sistema muestra lista de productos con valor actual (ej. 10% o 5 unidades).
3	El gerente edita valor para uno o varios productos (puede aplicar por categoría).
4	El sistema valida que el valor sea $\geq 1$ unidad o $\geq 1\%$ .
5	El sistema guarda cambios y actualiza reglas de detección de stock mínimo.
6	Confirma configuración guardada.

**Postcondición:**

Umbrales actualizados y aplicables en futuras comandas/reposiciones.

**Excepciones:**

Paso	Acción
E.1	Valor $< 1 \rightarrow$ rechaza y sugiere mínimo permitido.
E.2	Cambio masivo sin confirmación $\rightarrow$ solicita confirmación explícita.

## CU-006 Comprobar caducidades automáticamente

Versión: 1.0 (16/02/2026)

### Dependencias:

- RG-001 Gestionar inventario
- RG-002 Gestionar productos próximos a caducar
- RN-001 Fecha de caducidad de los productos
- RN-003 Plazo de alerta

### Precondición:

El sistema está en funcionamiento; existen productos con fecha de caducidad registrada.

### Descripción:

El sistema revisa diariamente las fechas de caducidad y genera alertas según plazos configurados.

### Secuencia normal:

Paso	Acción
1	El sistema ejecuta proceso (automático a las 06:00 o manual al presionar Comprobar ahora).
2	Recorre todos los productos con fecha de caducidad registrada.
3	Clasifica: Caducados (fecha ≤ hoy) y Próximos (fecha ≤ hoy + N días).
4	Para cada producto afectado: registra alerta, marca como no usable (si caducado), y notifica.
5	Genera resumen ejecutivo (cantidad de productos afectados)
6	Termina proceso.

### Postcondición:

Alertas actualizadas y notificaciones disparadas según configuración.

### Excepciones:

Paso	Acción

E.1	Fallo al leer fechas → registra error y continúa con resto.
E.2	Sin productos con caducidad → termina sin alertas.

## CU-007 Notificar por WhatsApp

**Versión:** 1.0 (16/02/2026)

### Dependencias:

- RG-002 Gestionar productos próximos a caducar
- RN-003 Plazo de alerta

### Precondición:

El sistema ha detectado una condición de alerta (stock mínimo, caducidad próxima o producto caducado) y el gerente tiene número de WhatsApp registrado.

### Descripción:

El sistema envía notificaciones automáticas al gerente vía WhatsApp cuando se cumplen ciertas condiciones críticas.

### Secuencia normal:

Paso	Acción
1	El sistema detecta una alerta (ej. stock mínimo tras comanda, producto caducado).
2	Verifica que el gerente tenga número de WhatsApp configurado.
3	Formatea mensaje con detalles: tipo de alerta, producto(s), cantidad, fecha relevante.
4	El sistema envía mensaje mediante API de WhatsApp Business (o integración externa)
5	Registra el envío en historial de notificaciones (fecha, destinatario, contenido).
6	Confirma internamente que la notificación fue enviada.

### Postcondición:

El gerente recibe alerta en tiempo real vía WhatsApp.

**Excepciones:**

Paso	Acción
E.1	Número de WhatsApp no configurado → registra error y notifica en panel interno.
E.2	Fallo en API de WhatsApp → reintentar 2 veces, luego registrar error persistente.
E.3	Mensaje vacío o mal formado → cancelar envío y loguear excepción.

## Modelo y diseño cerrado de la base de datos

