

**Sistema de gestión integral para planta cervecera (SGIPC)**  
**diseño, desarrollo e implementación de una solución web para la trazabilidad del proceso**  
**productivo en Bebidas Coral S.A.S**

Michael Steven Beltrán Osorio

Asesor

Alfredo Jesús Castro Guzman

Universidad nacional abierta y a distancia (UNAD)  
Escuela de ciencias básicas, tecnología e ingeniería (ECBTI)  
Ingeniería de sistemas

Bogotá D.C

2026

## Resumen

Esta automatización se presenta para el desarrollo del Sistema de Gestión Integral para Planta Cervecera (SGIPC), una aplicación web que se diseñó con el fin de digitalizar los procesos productivos de la planta de cerveza artesanal de la empresa Bebidas Coral S.A.S. El objetivo del proyecto sale al identificar, que muchos de los procesos diarios de la planta se diligenciaban de forma manual, lo que generaba pérdida de la trazabilidad de los lotes (parámetros del proceso), registros duplicados o nulos y escasa información del inventario. A partir de identificar estas oportunidades de mejora se realiza una propuesta de un software que permita controlar, desde una computadora, el inventario, los lotes (información etapa por etapa), procesos unitarios, almacenamiento en los equipos desde la materia prima hasta producto terminado.

Para construir el sistema se utilizó una arquitectura de tres capas, para el frontend se trabajó con React y TypeScript, en el backend con Node.js y NestJS, y para la base de datos PostgreSQL al ser una herramienta robusta y de uso libre, maneja sin problemas los históricos que se necesitan en la trazabilidad. La aplicación se organizó en módulos independientes que se comunican entre sí, lo cual facilita que en el futuro se puedan agregar nuevas actualizaciones sin afectar lo que ya está trabajando. Cada módulo corresponde a una etapa real del proceso cervecero asegurando la trazabilidad del lote desde que entra la materia prima hasta que el producto terminado se le entrega al cliente.

Como método seguí los lineamientos del estándar IEEE 29148-2018. Es un marco que normalmente se usa en proyectos industriales grandes, pero funciono para realizar una adaptación para un proyecto pequeño. Concluí con un documento que vincula cada requerimiento a un objetivo, un caso de uso y una prueba. Además, se elaboró un manual de usuario para la

operación diaria y un plan de mantenimiento que cubre lo preventivo, correctivo, adaptativo y perfectivo.

Los resultados esperados son concretos, una reducción del 30% en el tiempo que la planta gasta registrando información a mano y una precisión del 95% en el inventario. Reportes que antes se tardaban hasta una hora, ahora salen en un minuto. Un control por roles que asegura que cada persona acceda solo a su área de trabajo. El sistema está dimensionado para tres líneas de producción, cinco recetas activas y diez usuarios al mismo tiempo, con margen para crecer hasta cinco veces sin rediseñar.

***Palabras clave:*** sistema de información, trazabilidad, planta de cervecera, gestión de inventarios, arquitectura de software, IEEE 29148, desarrollo web, PostgreSQL, NestJS, React.

## Abstract

This automation is presented for the development of the Integrated Management System for Brewery Plant (SGIPC), a web application designed to digitize the production processes of the craft brewery plant operated by Bebidas Coral S.A.S. The project's objective arose from identifying that many of the plant's daily processes were being recorded manually, which led to loss of batch traceability (process parameters), duplicate or missing records, and limited inventory information. Based on the identification of these improvement opportunities, a software proposal was developed to enable control from a computer of inventory, batches (stage-by-stage information), unit processes, and equipment storage, from raw materials through to finished product.

To build the system, a three-tier architecture was used. The frontend was developed with React and TypeScript, the backend with Node.js and NestJS, and PostgreSQL was selected for the database, as it is a robust open-source tool that handles without difficulty the historical records required for traceability. The application was organized into independent modules that communicate with one another, which makes it easier in the future to add new updates without affecting what is already in operation. Each module corresponds to an actual stage of the brewing process, ensuring batch traceability from the moment raw materials enter the plant until the finished product is delivered to the customer.

As a methodology, I followed the guidelines of the IEEE 29148-2018 standard. This is a framework typically used in large industrial projects, but it proved useful when adapted to a small-scale project. I concluded with a document that links each requirement to an objective, a use case, and a test. In addition, a user manual was prepared for daily operations, along with a maintenance plan covering preventive, corrective, adaptive, and perfective maintenance.

The expected results are concrete: a 30% reduction in the time the plant spends manually recording information, and 95% inventory accuracy. Reports that previously took up to an hour are now generated in one minute. A role-based access control ensures that each person can access only their corresponding work area. The system is sized to support three production lines, five active recipes, and ten concurrent users, with capacity to scale up to five times that load without requiring redesign.

***Keywords:*** information system, traceability, brewery plant, inventory management, software architecture, IEEE 29148, web development, PostgreSQL, NestJS, React.

## Glosario Técnico

Se presenta a continuación el glosario de términos técnicos y operativos utilizados a lo largo del documento.

Término	Definición
SGIPC	Sistema de Gestión Integral para Planta de Cerveza.
Insumo / Materia Prima	Elemento utilizado en la elaboración de cerveza (cebada, lúpulo, levadura).
Proveedor	Entidad externa que suministra insumos a la planta.
Receta	Conjunto de instrucciones que define cantidades de insumos y procesos para producir un lote.
Lote	Unidad de producción única e identificable, base de la trazabilidad.
Inventario	Registro actualizado de existencias, tanto de materias primas como de productos terminados.
Stock	Cantidad disponible en inventario de un insumo o producto terminado.
Mermas	Pérdidas de producto o insumo en inventario, registradas como ajustes.
Pedido	Solicitud realizada por un cliente para adquirir productos terminados.
Entrega	Proceso de despacho de un pedido desde la planta hacia el cliente.
BBT	Bright Beer Tank, tanque de brillantez utilizado para maduración y carbonatación.
Trasiego	Operación de traslado del líquido entre tanques sin exposición al oxígeno.
Operario	Usuario de planta encargado de registrar entradas de insumos, lotes e inventario.
Jefe de Planta	Usuario con rol de supervisión, validación de lotes y generación de reportes.
Administrador	Usuario con acceso total al sistema.
Trazabilidad	Capacidad de rastrear cada lote desde los insumos hasta la entrega final al cliente.
Auditoría	Registro de todas las acciones realizadas por los usuarios en el sistema.
Backup	Copia de seguridad automática de la base de datos.
Legacy	Información histórica ya existente antes de la puesta en marcha del sistema.

## Lista de Abreviaturas y Acrónimos

Siglas	Significado
SGIPC	Sistema de Gestión Integral para Planta Cervecera
BBT	Bright Beer Tank (Tanque de Brillantez)
ERP	Enterprise Resource Planning
REQ	Requerimiento Funcional
RNF	Requerimiento No Funcional
RN	Regla de Negocio
OR	Otros Requerimientos
OBJ	Objetivo del Sistema
UC	Caso de Uso
API	Application Programming Interface
JWT	JSON Web Token
TLS	Transport Layer Security
HTTPS	HyperText Transfer Protocol Secure
ORM	Object-Relational Mapping
SPA	Single Page Application
CRUD	Create, Read, Update, Delete
3FN	Tercera Forma Normal
GDPR	General Data Protection Regulation
CIP	Cleaning In Place

## Tabla de contenido

Resumen .....	2
Abstract .....	4
Glosario Técnico .....	6
Lista de Abreviaturas y Acrónimos.....	7
Tabla de contenido .....	8
Lista De Tablas.....	13
Tabla De Figuras .....	15
Introducción .....	16
Formulación del Problema .....	17
Justificación.....	18
Objetivos .....	19
Objetivo General .....	19
Objetivos Específicos.....	19
Alcance y Limitaciones.....	20
Marco Teórico .....	21
Trazabilidad Y Gestión De Inventarios En La Industria De Alimentos Y Bebidas.....	21
Definición Y Alcance De La Trazabilidad.....	21
Tipos De Trazabilidad.....	22
Digitalización Y Software De Trazabilidad .....	22

Arquitectura De Software Y Diseño Modular.....	23
Ingeniería De Software Como Disciplina .....	23
Arquitectura Por Capas .....	23
Diseño Modular.....	24
Estándar ISO/IEC/IEEE 29148 Y La Ingeniería De Requisitos .....	24
La Ingeniería De Requisitos Como Proceso Fundamental.....	24
Estructura Del Estándar ISO/IEC/IEEE 29148:2018 .....	25
Trazabilidad De Requisitos .....	26
Stack Tecnológico De Desarrollo.....	26
React Y El Desarrollo Del Frontend .....	26
Nestjs Y La Arquitectura Del Backend.....	26
PostgreSQL Y La Persistencia De Datos .....	27
La Industria Cervecera Artesanal Y Sus Procesos .....	28
Etapas Del Proceso Productivo .....	28
Importancia Del Registro Y La Trazabilidad Por Lote .....	28
Metodología .....	30
Fase 1: Análisis del Contexto y Levantamiento de Requerimientos.....	30
Fase 2: Diseño del Sistema.....	30
Fase 3: Desarrollo e Implementación.....	31
Fase 4: Pruebas y Validación .....	32

	10
Fase 5: Documentación y Entrega.....	32
Especificación de Requerimientos del Software .....	33
Clases y Características de Usuarios .....	33
Entorno Operativo .....	34
Plataforma de Hardware.....	34
Plataforma de Software .....	34
Requerimientos Funcionales .....	35
Gestión de Materia Prima.....	35
Gestión de Producción .....	36
Gestión de Inventario .....	37
Gestión de Pedidos y Logística .....	37
Generación de Reportes.....	38
Gestión de Usuarios, Seguridad y Auditoría .....	39
Requerimientos No Funcionales .....	40
Rendimiento .....	40
Compatibilidad y Portabilidad .....	41
Seguridad.....	41
Usabilidad.....	42
Fiabilidad y Disponibilidad.....	42
Escalabilidad y Mantenibilidad.....	43

Respaldo y Recuperación.....	43
Reglas de Negocio.....	44
Reglas Generales.....	44
Reglas del proceso.....	45
Otros Requerimientos.....	46
Base de Datos.....	46
Legales y Normativos.....	47
Reúso.....	47
Matriz de Trazabilidad.....	48
Diseño del Sistema.....	50
Descripción General del Diseño.....	50
Modelado de Procesos.....	50
Diseño de la Base de Datos.....	54
Tablas Complementarias.....	54
Diseño Visual e Interfaces del Sistema.....	57
Alcance Funcional del Sistema.....	61
Normalización.....	62
Definición Tecnológica.....	62
Arquitectura General del Software.....	63
Manual de Usuario del Sistema.....	64

Introducción al Uso del Sistema.....	64
Requisitos para el Uso del Sistema .....	65
Uso de los Módulos.....	65
Módulo de Salidas de Producto.....	70
Plan de Mantenimiento del Sistema .....	71
Objetivos Específicos del Mantenimiento.....	71
Clasificación del Mantenimiento.....	71
Roles y Responsabilidades .....	72
Procedimiento de Atención de Incidencias .....	72
Indicadores de Seguimiento .....	73
Resultados y Discusión .....	74
Cumplimiento de los Objetivos.....	74
Discusión sobre Decisiones de Diseño.....	75
Conclusiones .....	77
Trabajo Futuro.....	78
Referencias Bibliográficas .....	80

## Lista De Tablas

<b>Tabla 1.</b> <i>Clases de usuarios del SGIPC</i> .....	33
<b>Tabla 2.</b> <i>Requerimientos funcionales del módulo Gestión de Materia Prima</i> .....	35
<b>Tabla 3.</b> <i>Requerimientos funcionales del módulo Gestión de Producción</i> .....	36
<b>Tabla 4.</b> <i>Requerimientos funcionales del módulo Gestión de Inventario</i> .....	37
<b>Tabla 5.</b> <i>Requerimientos funcionales del módulo Gestión de Pedidos y Logística</i> .....	38
<b>Tabla 6.</b> <i>Requerimientos funcionales del módulo Generación de Reportes</i> .....	38
<b>Tabla 7.</b> <i>Requerimientos funcionales de Seguridad, Usuarios y Auditoría</i> .....	39
<b>Tabla 8.</b> <i>Requerimientos no funcionales de rendimiento</i> .....	40
<b>Tabla 9.</b> <i>Requerimientos no funcionales de compatibilidad y portabilidad</i> .....	41
<b>Tabla 10.</b> <i>Requerimientos no funcionales de seguridad</i> .....	41
<b>Tabla 11.</b> <i>Requerimientos no funcionales de usabilidad</i> .....	42
<b>Tabla 12.</b> <i>Requerimientos no funcionales de fiabilidad y disponibilidad</i> .....	42
<b>Tabla 13.</b> <i>Requerimientos no funcionales de escalabilidad y mantenibilidad</i> .....	43
<b>Tabla 14.</b> <i>Requerimientos no funcionales de respaldo y recuperación</i> .....	43
<b>Tabla 15.</b> <i>Reglas de negocio generales del sistema</i> .....	44
<b>Tabla 16.</b> <i>Reglas del proceso</i> .....	45
<b>Tabla 17.</b> <i>Otros requerimientos relacionados con la base de datos</i> .....	46
<b>Tabla 18.</b> <i>Otros requerimientos de carácter legal y normativo</i> .....	47
<b>Tabla 19.</b> <i>Otros requerimientos asociados al reúso de componentes</i> .....	47
<b>Tabla 20.</b> <i>Objetivos del sistema SGIPC</i> .....	48
<b>Tabla 21.</b> <i>Matriz de trazabilidad — Materia Prima y Producción</i> .....	48
<b>Tabla 22.</b> <i>Matriz de trazabilidad — Inventario, Pedidos, Reportes y RNF</i> .....	49

<b>Tabla 23.</b> <i>Principales del modelo de datos</i> .....	54
<b>Tabla 24.</b> <i>Tablas complementarias del modelo de datos.</i> .....	55
<b>Tabla 25.</b> <i>Alcance funcional del sistema por módulo</i> .....	61
<b>Tabla 26.</b> <i>Configuración tecnológica del sistema</i> .....	62
<b>Tabla 27.</b> <i>Arquitectura por capas del sistema</i> .....	63
<b>Tabla 28.</b> <i>Clasificación del mantenimiento del SGIPC</i> .....	71
<b>Tabla 29.</b> <i>Roles y responsabilidades de mantenimiento</i> .....	72
<b>Tabla 30.</b> <i>Indicadores de seguimiento del mantenimiento</i> .....	73

## Tabla De Figuras

<b>Figura 1.</b> Flujo de Gestión de Materias Primas.....	51
<b>Figura 2.</b> Flujo de producción, fermentación y empaque.....	52
<b>Figura 3.</b> Flujo de control de cuarto frío, salidas y reportes.....	53
<b>Figura 4.</b> Inicio de Sesión.....	57
<b>Figura 5.</b> Mapa interactivo de la planta.....	57
<b>Figura 6.</b> Pantalla Operativa Principal de materia prima .....	58
<b>Figura 7.</b> Pantalla Operativa Principal de producción.....	58
<b>Figura 8.</b> Pantalla Operativa Principal de fermentación .....	59
<b>Figura 9.</b> Pantalla Operativa Principal de BBT .....	59
<b>Figura 10.</b> Pantalla Operativa Principal de embotellado.....	60
<b>Figura 11.</b> Pantalla Operativa Principal de cuarto .....	60
<b>Figura 12.</b> Mapa interactivo .....	64
<b>Figura 13.</b> Módulo de Materias Primas.....	65
<b>Figura 14.</b> Módulo de Producción.....	66
<b>Figura 15.</b> Módulo de Fermentadores .....	67
<b>Figura 16.</b> Módulo BBT .....	68
<b>Figura 17.</b> Módulo Cuarto Frío .....	69
<b>Figura 18.</b> Módulo de Embotellado o Enlatado .....	69

## Introducción

La cerveza artesanal en Colombia dejó de ser un nicho hace tiempo. Hoy compete con marcas industriales y con importaciones, y eso empuja a las plantas pequeñas a profesionalizarse si quieren mantenerse. No se trata solo de hacer buena cerveza sino también de cumplir con normas vigentes legales.

Cuando empecé en Bebidas Coral S.A.S, la operación se registraba en planillas y hojas de cálculo. No se tenían control ni registro de materias primas, los lotes no tenían un identificador único que el sistema generara por sí mismo, y los cambios de estado de los fermentadores se comunicaban de viva voz entre el personal. El inventario del cuarto frío se revisaba cada tanto mediante conteos físicos que casi nunca coincidían con lo que estaba registrado.

Se perdía la relación entre los insumos usados y el producto que salía al cliente. Ninguno tenía en tiempo real de cuántos litros estaban fermentando, tampoco la etapa de cada cerveza. Los reportes gerenciales tardaban horas o incluso días en consolidarse. Y cuando el inventario físico no cuadraba con el registrado, no se tenía como construir dónde estaba la diferencia. Ante ese problema, se toma la idea de desarrollar una herramienta tecnológica que integrara en una sola plataforma todas las etapas del proceso cervecero.

## **Formulación del Problema**

¿En qué medida el diseño e implementación de un sistema de información web, orientado a la trazabilidad por lote, permite optimizar los procesos operativos de la planta cervecera Bebidas Coral S.A.S., garantizando el control en tiempo real de las materias primas, la producción, el inventario y las entregas?

## Justificación

La sistematización de los procesos productivos en la industria cervecera artesanal representa un factor crítico para la sostenibilidad y el crecimiento de este tipo de empresas. La implementación del Sistema SGIPC se justifica desde tres dimensiones complementarias: operativa, estratégica y académica.

Desde la perspectiva operativa, el sistema aporta beneficios concretos y medibles a la empresa cliente. La automatización del registro de insumos reduce significativamente los errores de digitación y los tiempos dedicados a tareas administrativas, la generación automática de identificadores únicos para cada lote permite reconstruir la historia completa del producto, lo cual resulta indispensable en caso de auditorías sanitarias o reclamaciones de calidad. El control en tiempo real del inventario facilita la toma de decisiones sobre compras, producción y despachos, mientras que el sistema de reportes automatizados libera al personal gerencial de tareas repetitivas de consolidación manual.

En la perspectiva estratégica, la solución posiciona a Bebidas Coral S.A.S en un estándar tecnológico comparable al de empresas de mayor tamaño, lo cual se traduce en una mejor imagen ante clientes corporativos, mayor capacidad de respuesta ante oportunidades comerciales y una base tecnológica sobre la cual escalar a múltiples sedes, así como integrar sensores IoT, códigos QR o procesos CIP en fases futuras. La arquitectura modular propuesta permite reutilizar componentes clave (como los módulos de seguridad, inventario o reportes) en otros proyectos internos de la organización. la perspectiva académica constituye un caso real de aplicación de múltiples áreas del conocimiento en ingeniería, análisis y diseño de sistemas bajo el estándar IEEE 29148-2018, modelando las bases de datos relacionales, así como el diseño de arquitecturas web, implementación de seguridad mediante autenticación basada en tokens.

## **Objetivos**

### **Objetivo General**

Diseñar, desarrollar e implementar un Sistema de Gestión para Planta Cervecera, mediante una solución web modular que digitalice y optimice los procesos operativos de Bebidas Coral S.A.S, garantizando la trazabilidad por lote desde la recepción de materias primas hasta la entrega final del producto al cliente.

### **Objetivos Específicos**

Definir los requerimientos funcionales, no funcionales y reglas de negocio del sistema, siguiendo los lineamientos del estándar IEEE 29148-2018, garantizando la trazabilidad entre objetivos, requerimientos, casos de uso y pruebas.

Modelar una base de datos relacional, que soporte la trazabilidad por lote, que permita obtener los históricos de movimientos y la integridad transaccional de todos los módulos del sistema.

Diseñar una arquitectura de software modular que permita la escalabilidad horizontal y el mantenimiento independiente de cada módulo.

Desarrollar los módulos de gestión de materias primas, producción, fermentadores, BBT, cuarto frío, embotellado, salidas y reportes, implementando las reglas de negocio específicas del proceso cervecero artesanal.

Elaborar un manual de usuario orientado a la capacitación de los colaboradores, así como un plan de mantenimiento que cubra las actividades preventivas, correctivas y adaptativas necesarias para asegurar la continuidad del sistema.

## **Alcance y Limitaciones**

Este proyecto abarca el análisis, diseño e implementación del SGIPC. Dentro de esta versión entran los módulos operativos completos (materias primas, producción, fermentación, BBT, cuarto frío, embotellado o embarrilado, salidas y reportes). También los módulos transversales que sostienen a los demás (autenticación, gestión de usuarios, auditoría y respaldo), En su configuración inicial el sistema está dimensionado para tres líneas de producción activas, cinco recetas en operación simultánea y 10 usuarios concurrentes.

Se decidió no implementar la integración con sistemas ERP externos como Odoo o SAP Business One, y tampoco conexiones con servicios de facturación electrónica. No porque no fueran relevantes, sino porque su desarrollo correcto exige un esfuerzo que desbordaba los tiempos del trabajo de grado. Quedan identificadas como solicitudes claras para una segunda fase. Lo mismo aplica a la analítica avanzada, la inteligencia de negocios y la integración de sensores IoT. La arquitectura está lista para recibirlas sin cambios estructurales mayores, pero su implementación no forma parte de esta versión.

## **Marco Teórico**

El presente capítulo expone los fundamentos conceptuales que sustentan el desarrollo del Sistema SGIPC. Se abordan cinco ejes temáticos: la trazabilidad y la gestión de inventarios en la industria de alimentos y bebidas, la arquitectura de software por capas y el diseño modular, el estándar internacional ISO/IEC/IEEE 29148 para la ingeniería de requisitos, el stack tecnológico empleado (React, NestJS y PostgreSQL) y, los aspectos relevantes del proceso productivo de la cerveza artesanal. Cada eje articula la base conceptual del sistema propuesto y justifica las decisiones técnicas adoptadas durante su desarrollo.

### **Trazabilidad Y Gestión De Inventarios En La Industria De Alimentos Y Bebidas**

#### ***Definición Y Alcance De La Trazabilidad***

La trazabilidad constituye un concepto central en la gestión moderna de la industria alimentaria. De acuerdo con el Reglamento (CE) N.º 178/2002 del Parlamento Europeo, retomado por la Agencia Española de Seguridad Alimentaria y Nutrición (AESAN, s.f.), la trazabilidad se define como la posibilidad de encontrar y seguir el rastro, a través de todas las etapas de producción, transformación y distribución de un alimento o un ingrediente. Esta capacidad es decisiva para la protección de los consumidores y para la gestión interna de las empresas del sector.

La trazabilidad no es únicamente una obligación normativa, representa una herramienta fundamental para la gestión del riesgo, ya que aumenta la eficacia del control interno gracias al mayor nivel de información disponible sobre productos y procesos (AESAN, 2009). En la práctica, los sistemas de trazabilidad permiten a las organizaciones abordar problemas críticos

como retiradas de producto o alertas sanitarias sin perturbar de manera significativa su operación comercial.

### **Tipos De Trazabilidad**

La literatura especializada distingue tres tipos de trazabilidad complementarios entre sí. La trazabilidad hacia atrás permite obtener información relevante asociada a un producto intermedio o final, hasta llegar al origen de las materias primas, registrando datos como nombre y dirección de proveedores, mercancía recibida e identificación de cada lote. La trazabilidad de proceso, también llamada trazabilidad interna, vincula los productos que entran a la empresa con los que salen, identificando los productos obtenidos como resultado de las operaciones desarrolladas dentro de la planta; la trazabilidad hacia adelante registra los datos de la clientela, las mercancías distribuidas y la fecha de salida del establecimiento (AESAN, 2009; Junta de Andalucía, 2025).

Para que el sistema de trazabilidad cumpla sus objetivos, es necesario vincular lo que entra con lo que sale; en otras palabras, disponer de una trazabilidad interna desarrollada (Gobierno de Aragón, s.f.). Esta es precisamente la dimensión más relevante para una planta de producción cervecera, donde la información del proceso debe documentarse etapa por etapa para garantizar la calidad del producto final, así como la estandarización del producto.

### ***Digitalización Y Software De Trazabilidad***

La industria de alimentos moderna ha incorporado de manera creciente herramientas digitales para gestionar la trazabilidad. Los sistemas tradicionales basados en registros en papel resultan en vulnerabilidades, errores, duplicaciones y pérdida de información, mientras que las soluciones basadas en software permiten capturar datos de producción en tiempo real, reducir

posibles retiradas de producto y disminuir costes de inactividad (Tulip Interfaces, 2025). En este sentido, la elección del software de trazabilidad debe responder a las necesidades específicas de la empresa y ofrecer capacidad de integración con los sistemas existentes (Gestión 5, 2024).

La gestión de inventarios se vincula con la trazabilidad, ya que ambos procesos comparten el registro detallado y sistemático de información en cada punto crítico, incluyendo fechas, ubicaciones y cantidades. Un sistema bien diseñado permite no solo rastrear productos, sino también optimizar el control de los procesos de producción, la gestión de la cadena de suministro y las políticas de compras a proveedores (Controla Plus, 2021).

## **Arquitectura De Software Y Diseño Modular**

### ***Ingeniería De Software Como Disciplina***

La ingeniería de software se concibe como una disciplina que organiza el desarrollo de sistemas en capas sustentadas en el proceso, los métodos y las herramientas (Pressman & Maxim, 2014). Sommerville (2011) la define como una disciplina que comprende todos los aspectos de la producción de software, desde las primeras etapas de especificación del sistema hasta el mantenimiento del producto cuando ya está en uso. Esta visión reconoce que el software nunca opera de manera aislada, sino como parte de un sistema mayor que incluye hardware, operación humanos y frecuentemente organizaciones.

### ***Arquitectura Por Capas***

El diseño arquitectónico es el proceso mediante el cual se establecen las estructuras básicas que identifican los componentes principales de un sistema y las comunicaciones entre ellos. Esta arquitectura afecta directamente al rendimiento, la solidez, el grado de distribución y la mantenibilidad del sistema resultante (Sommerville, 2011). Entre los modelos arquitectónicos

clásicos se destaca el modelo de capas, el cual el sistema se piensa como una serie de niveles, cada uno de los cuales ofrece servicios al nivel inmediatamente superior.

La arquitectura de tres capas establece una de las aplicaciones más extendidas de este modelo, especialmente en sistemas de información de gestión empresarial. En esta arquitectura se distinguen tres niveles fundamentales, la capa de presentación (interfaz de usuario), la capa de lógica de negocio (procesamiento) y la capa de datos (persistencia). La separación de responsabilidades entre capas facilita el mantenimiento, permite escalar componentes de manera independiente y simplifica la incorporación de cambios sin afectar la totalidad del sistema (Pressman & Maxim, 2014).

### ***Diseño Modular***

El principio de modularidad complementa el diseño en capas: un sistema modular se construye a partir de componentes independientes que se comunican entre sí mediante interfaces bien definidas. Sommerville (2011) y Pressman y Maxim (2014) coinciden en señalar que el modularidad, junto con la abstracción, constituye uno de los principios fundamentales que sostienen una práctica efectiva de la ingeniería de software. Un diseño adecuado permite incorporar nuevas funcionalidades en el futuro sin afectar lo que ya está en operación, lo cual resulta especialmente valioso en sistemas que deben evolucionar con los cambios del proceso productivo.

## **Estándar ISO/IEC/IEEE 29148 Y La Ingeniería De Requisitos**

### ***La Ingeniería De Requisitos Como Proceso Fundamental***

La ingeniería de requisitos es un proceso fundamental del ciclo de vida del desarrollo de software que permite definir las funcionalidades, la calidad y el alcance de un producto. Es

además un proceso de participación humana intensiva, en el cual la calidad de información obtenida determina en gran medida la calidad del diseño y la implementación de este (Llanque et al., 2022).

Las fallas durante esta etapa han sido reportadas como una de las causas más frecuentes de problemas en proyectos de software, lo que justifica la ayuda de marcos normalizados que estructuren las actividades de obtención, análisis, especificación y validación de requisitos.

### ***Estructura Del Estándar ISO/IEC/IEEE 29148:2018***

El estándar ISO/IEC/IEEE 29148:2018 Systems and software engineering Life cycle processes Requirements engineering constituye el referente internacional vigente para los procesos de ingeniería de requisitos en el desarrollo de productos y sistemas de software y hardware (ISO, 2018). Este estándar reemplazó a la conocida norma IEEE 830-1998, y proporciona lineamientos para aplicar los procesos relacionados con requisitos descritos en ISO/IEC/IEEE 15288 e ISO/IEC/IEEE 12207, especificando los ítems de información requeridos, su contenido mínimo y su formato.

La norma define tres procesos centrales: primero el análisis del negocio o misión, orientado a definir el problema u oportunidad, y caracterizar el espacio de solución; segundo la definición de necesidades y requisitos de los interesados (stakeholders), que identifica a los actores del sistema y las características y contexto de uso requeridos; y en tercer lugar la definición de los requisitos del sistema del software, que transforma la visión orientada al usuario en una visión técnica de la solución (CWNP, s.f.). Adicionalmente, la norma propone un listado mínimo de tipos de requisitos que incluye requisitos funcionales, requisitos de usabilidad, requisitos de desempeño e interfaces del sistema (Studocu, 2024).

## **Trazabilidad De Requisitos**

Uno de los aportes centrales del estándar ISO/IEC/IEEE 29148 es el énfasis en la trazabilidad de los requisitos, cada requisito del sistema debe poder vincularse con los requisitos de los interesados y con los requisitos del negocio o misión que le dieron origen. Esta trazabilidad bidireccional permite verificar que cada necesidad identificada se transforma en una solución concreta y que cada elemento implementado responde a una necesidad real, lo cual constituye una buena práctica reconocida internacionalmente (Lozano & Suárez, 2020).

## **Stack Tecnológico De Desarrollo**

### ***React Y El Desarrollo Del Frontend***

React es una biblioteca de JavaScript para la construcción de interfaces de usuario, ampliamente adoptada en el desarrollo de aplicaciones web. Su modelo basado en componentes reutilizables y su gestión explicativa del estado, permiten construir interfaces dinámicas que ofrecen una experiencia de usuario fluida (Singh, 2025). Cuando se combina con TypeScript (un superconjunto de JavaScript con tipado estático) se obtiene un entorno de desarrollo más seguro, en el que muchos errores se detectan en tiempo de compilación, antes de llegar a producción.

### ***Nestjs Y La Arquitectura Del Backend***

NestJS es un framework progresivo para Node.js orientado a la construcción de aplicaciones del lado del servidor, es eficientes, confiables y escalables (Wagner, 2025). Está construido en TypeScript e incorpora los principios de la programación orientada a objetos, así como la programación funcional y la programación reactiva funcional, su arquitectura, fuertemente inspirada en Angular, organiza la aplicación en módulos que contienen

controladores, servicios y proveedores, todos ellos gestionados a través del mecanismo de inyección de dependencias.

Esta organización modular facilita la separación de responsabilidades, la facilidad para hacerle pruebas al código y para darle mantenimiento al sistema a futuro (MyTaskPanel Consulting, 2024). El framework resulta particularmente adecuado para la construcción de APIs RESTful y para sistemas empresariales complejos que requieren una arquitectura escalable, características que coinciden con los requisitos del SGIPC.

### ***PostgreSQL Y La Persistencia De Datos***

PostgreSQL es un sistema de gestión de bases de datos objeto-relacional de código abierto, con más de 35 años de desarrollo activo, que ha consolidado una sólida reputación por su confiabilidad, robustez y rendimiento (PostgreSQL Global Development Group, 2024). Entre sus características técnicas se destacan la conformidad con las propiedades ACID, el control de concurrencia multiversión (MVCC), la recuperación a un punto en el tiempo y el soporte para tipos de datos no relacionales como JSON (Instaclustr, 2025).

Estas características hacen de PostgreSQL, una herramienta especialmente apropiada para sistemas que requieren manejo confiable de información histórica (como los registros de trazabilidad de lotes) y que deben garantizar la integridad de los datos frente a operaciones concurrentes de múltiples usuarios. Su licencia abierta y la madurez de su comunidad reducen los costos de adopción y aseguran la sostenibilidad técnica del sistema en el tiempo.

## **La Industria Cervecera Artesanal Y Sus Procesos**

### ***Etapas Del Proceso Productivo***

La elaboración de cerveza artesanal es un proceso que combina tradición y precisión técnica, organizado en una serie de etapas claramente diferenciadas (Cerveza Destino, 2025). El proceso inicia con la molienda del grano (usualmente cebada malteada), cuyo objetivo es romper la cáscara para exponer el almidón, después empieza el proceso de maceración, en la que la malta molida se mezcla con agua a temperaturas entre 45 °C y 67 °C, dependiendo del estilo de cerveza, para extraer los azúcares fermentables mediante transformaciones enzimáticas (Cerveza Tyrís, 2020).

Posteriormente se realizan la filtración del mosto, el lavado y la cocción con lúpulo durante aproximadamente una hora, etapa en la cual se aporta amargor, aroma y se esteriliza el líquido. El mosto como se le llama se enfría rápidamente y se envía al fermentador, donde se incorpora la levadura. Durante la fermentación (que se prolonga típicamente entre dos y tres semanas), los azúcares se transforman en alcohol y dióxido de carbono. Finalmente, la cerveza pasa a maduración, donde se clarifica y mejora organolépticamente, y se envasa para su distribución (Cerveza Tyrís, 2020; Sierra del Guadarrama, 2024).

### **Importancia Del Registro Y La Trazabilidad Por Lote**

Cada una de las etapas descritas implica decisiones técnicas y mediciones críticas: temperaturas de maceración, tiempos de cocción, momentos de adición de lúpulo, densidades iniciales y finales, perfiles de levadura, entre muchas otras variables. La adecuada documentación de estos parámetros para cada lote no solo garantiza el estandarizado de las recetas, sino que también permite identificar las causas de desviaciones cuando el producto final

no cumple con las especificaciones esperadas. En este contexto, el registro electrónico estructurado por etapas se convierte en una necesidad operativa, especialmente cuando la planta opera múltiples recetas en paralelo y debe responder por la calidad de cada lote entregado al cliente.

## **Metodología**

El desarrollo del SGIPC no fue lineal. Se optó por una metodología iterativa e incremental dividida en cinco fases, y aunque las describo aquí en orden, en la práctica hubo varios momentos en que tuve que retroceder a fases anteriores para ajustar decisiones.

### **Fase 1: Análisis del Contexto y Levantamiento de Requerimientos**

Esta fase tenía un objetivo claro, entender cómo funcionaba realmente la planta. Se realizaron entrevistas con el personal operativo, el jefe de planta y con la administración, así como la observación directa con el proceso. Se analizó e investigó sobre la ejecución de los procesos, documentando los problemas que los operarios tenían que resolver en cada producción, se solicitó las planillas y formatos de producción, esto con el fin de entenderlos e implementarlos a sistema, de igual manera se tomó tiempos de producción con la finalidad de encontrar “cuellos de botella” o reprocesos.

El entregable de esta fase fue el documento de requerimientos del software, elaborado bajo el estándar IEEE 29148-2018. Que incluye el propósito y alcance del sistema, la caracterización de las clases de usuarios, el entorno operativo, los requerimientos funcionales organizados por módulo, los requerimientos no funcionales, las reglas de negocio, los requerimientos de interfaces externas y la matriz de trazabilidad. Esa matriz fue clave: sin ella es imposible saber si un requerimiento quedó huérfano o si una prueba no valida nada.

### **Fase 2: Diseño del Sistema**

A partir de los requerimientos obtenidos, se llevó a cabo el proceso de diseño. Se definió una arquitectura de tres capas con submódulos explícitos por proceso cervecero, se creó la base

de datos relacional con sus tablas principales y complementarias respetando la tercera forma normal (3FN), y los wireframes de las pantallas más representativas.

Asimismo, se establecieron las tecnológicas fundamentales del sistema. Para el frontend se seleccionó React con TypeScript y Vite; para el backend, Node.js con NestJS; TypeORM como mapeador objeto-relacional; y PostgreSQL como sistema gestor de base de datos.

Cada elección fue sustentada con criterios técnicos, teniendo en cuenta la sostenibilidad de ecosistema y la alineación con los requerimientos del proyecto. En particular, la elección de NestJS sobre alternativas como Spring Boot o Django responde a la posibilidad de mantener TypeScript de forma uniforme en todo el stack, lo cual reduce la carga cognitiva al transitar entre capas de la aplicación y favorece la coherencia del código base.

### **Fase 3: Desarrollo e Implementación**

El desarrollo se organizó por módulos funcionales, empezando por los que son base para los demás. Primero autenticación, gestión de usuarios y materias primas. Sin esos tres módulos, el programa no tiene herramientas para ejecutarse, Después vinieron producción, fermentadores, BBT, cuarto frío, embotellado y salidas. El módulo de reportes se dejó para el final, dado que recopila la información total del programa, cada módulo siguió un ciclo corto: diseño detallado, implementación, pruebas unitarias y validación funcional con el usuario líder de planta; La validación realizada con los usuarios finales estableció la etapa de mayor relevancia dentro del proceso de diseño. En múltiples ocasiones, elementos considerados intuitivos desde la perspectiva técnica resultaron carecer de coherencia con la lógica propia de la operación. La detección temprana de dichas diferencias en esta fase representó una ventaja considerable, dado que su identificación en etapas posteriores al despliegue implicaría un costo de corrección significativo, tanto en términos de tiempo como de recursos.

#### **Fase 4: Pruebas y Validación**

Las pruebas se estructuraron en tres niveles. Pruebas unitarias a nivel de función y componente, pruebas de integración entre módulos, y pruebas funcionales con el usuario líder sobre escenarios completos. Aparte de eso se diseñó pruebas de carga que simularon al menos 5 lotes de producción con sus respectivos procesos y trazabilidades. Las métricas que monitoreo fueron las usuales: tiempos de respuesta, uso de CPU y memoria en el servidor, latencia de las consultas a la base de datos, y tasa de errores bajo carga. Todo esto se comparó con los umbrales que habían definido en los requerimientos no funcionales.

#### **Fase 5: Documentación y Entrega**

La fase final se enfocó en los documentos que acompañan al sistema. El manual de usuario para la operación diaria. El plan de mantenimiento para la administración del sistema una vez en producción. Y esta misma memoria técnica. No son un trámite final. Sino como un anexo al programa; sirven para capacitar a usuarios nuevos, para que el equipo de soporte retome el sistema sin depender de mí, y para que cualquier evolución futura del software tenga una base escrita sobre la cual apoyarse.

## Especificación de Requerimientos del Software

Este capítulo documenta los requerimientos del SGIPC. Seguí la estructura del estándar IEEE 29148-2018. Alcanzando el propósito del sistema, las funcionalidades esperadas, las clases de usuarios, el entorno operativo, los requerimientos funcionales por módulos, los no funcionales, las reglas de negocio, las interfaces externas y otros requerimientos transversales.

### Clases y Características de Usuarios

El sistema tiene tres perfiles de usuario. Cada uno tiene una frecuencia de uso, funciones principales, privilegios de seguridad y nivel de experiencia diferentes. La Tabla 1 los resume.

**Tabla 1.**

*Clases de usuarios del SGIPC*

Perfil	Frecuencia	Funciones principales	Privilegios	Experiencia
Operario de Planta	Alta (diaria)	Registro de materias primas, producción de lotes, actualización de inventario	Restringido a funciones operativas	Básico
Jefe de Planta	Media-Alta	Supervisión de stock, validación de lotes, generación de reportes	Sin permisos para crear usuarios	Medio
Administrador	Media	Gestión de usuarios y roles, control de pedidos, configuración del sistema	Acceso total	Avanzado

Nota: Esta tabla muestra los perfiles que maneja el programa, funciones y privilegios de cada perfil.

## **Entorno Operativo**

El SGIPC corre en un entorno cliente-servidor basado en web. Se accede desde navegadores modernos en computadores personales y dispositivos móviles. A continuación, la configuración de hardware y software.

### **Plataforma de Hardware**

El servidor de producción necesita un procesador Intel Xeon o AMD Ryzen con mínimo 4 núcleos a 3.0 GHz, 8 GB de RAM, almacenamiento SSD de 500 GB o más, y conectividad LAN o acceso estable a Internet. Las estaciones de los operarios piden al menos 4 GB de RAM y un procesador Intel Core i3 o equivalente. En móviles se soporta Android 10 en adelante e iOS 13 en adelante.

### **Plataforma de Software**

El sistema operativo del servidor es Ubuntu Server 22.04 LTS. El motor principal de base de datos es PostgreSQL 14 o superior, y dejó MySQL 8.0 como alternativa viable. El backend está implementado en Node.js 18 con NestJS. El frontend en React con TypeScript. Los navegadores soportados son versiones recientes de Chrome, Firefox y Edge. La interfaz es adaptable y se adapta a PC, Tablet y teléfonos móviles sin que el usuario tenga que hacer nada especial.

## Requerimientos Funcionales

Estos requerimientos funcionales se manejaron por módulo. se Usó un código nemotécnico con el prefijo REQ, el acrónimo del módulo y un número consecutivo. Cada uno lleva una prioridad: alta, media o baja.

### Gestión de Materia Prima

Este módulo permite registrar, actualizar y consultar las entradas de insumos, su stock, y proveedores. Es el módulo base del sistema. De él dependen producción e inventario. El flujo empieza cuando el operario selecciona registrar entrada de insumos. El sistema abre un formulario con los campos nombre, tipo de insumo cantidad, unidad, fecha de ingreso y proveedor. Valida que la cantidad sea un número y que la fecha de vencimiento no sea anterior a hoy. Si todo está bien, guarda la entrada y actualiza el inventario. La Tabla 2 detalla los requerimientos.

#### Tabla 2.

##### *Requerimientos funcionales del módulo Gestión de Materia Prima*

Código	Prioridad	Descripción
REQ-MP-01	Alta	El sistema debe permitir registrar entradas de materia prima indicando nombre, cantidad, unidad de medida, proveedor y fecha de vencimiento.
REQ-MP-02	Alta	El sistema debe actualizar automáticamente el stock al registrar una nueva entrada.
REQ-MP-03	Media	El sistema debe generar alertas cuando un insumo esté próximo a vencer (configurable en días).
REQ-MP-04	Media	El sistema debe permitir consultar el historial de entradas de cada insumo.
REQ-MP-05	Alta	Si se ingresa una fecha de vencimiento inválida, el sistema debe impedir el registro y mostrar un mensaje de error.
REQ-MP-06	Media	El sistema debe permitir dar de baja insumos vencidos con justificación registrada.

Nota: Esta tabla presenta la descripción de cada requerimiento del área de materias prima.

## Gestión de Producción

Este módulo maneja la asignación de los id de la producción. El jefe de planta y/o operario encargado inicia producción, diligencia receta, cantidad, responsable y fecha. El sistema revisa que haya insumos suficientes para la receta y que los números sean válidos. Si pasa la validación, descuenta los insumos del inventario, genera un ID único para el lote, y registra fecha de inicio, estado y responsable. Al cerrar la producción queda el lote disponible para fermentación. La Tabla 3 muestra los requerimientos.

**Tabla 3.**

Requerimientos funcionales del módulo Gestión de Producción

Código	Prioridad	Descripción
REQ-PROD-01	Alta	Permitir iniciar la producción de un lote seleccionando receta, cantidad, responsable y fecha.
REQ-PROD-02	Alta	Verificar la disponibilidad de insumos antes de iniciar la producción y bloquear la orden si no hay stock suficiente.
REQ-PROD-03	Alta	Generar un identificador único para cada lote de producción.
REQ-PROD-04	Alta	Descontar automáticamente del inventario los insumos utilizados según la receta.
REQ-PROD-05	Media	Registrar estado y fechas de inicio y fin de cada lote.
REQ-PROD-06	Alta	Actualizar el inventario de productos terminados al finalizar la producción.
REQ-PROD-07	Media	Permitir consultar el historial de lotes producidos, con insumos consumidos y cantidades generadas.
REQ-PROD-08	Alta	En caso de error, notificar al usuario con un mensaje claro y no registrar la producción.

Nota: Esta tabla expone los requerimientos que necesita el área de producción.

## Gestión de Inventario

El módulo de inventario lleva el control de productos terminados. Stock por lote, entradas y salidas. El usuario accede al listado con nombre o lote. Desde ahí filtra, ajusta stock por mermas y el sistema hace dos cosas por su cuenta: da por terminado un lote cuando no hay existencias del producto, descuenta stock cuando sale una entrega y genera alertas por stock bajo según umbrales configurables.

### Tabla 4.

#### *Requerimientos funcionales del módulo Gestión de Inventario*

Código	Prioridad	Descripción
REQ-INV-01	Alta	Mostrar el inventario de productos terminados con detalle de lote, cantidad, fecha de producción y responsable.
REQ-INV-02	Alta	Actualizar automáticamente el inventario al finalizar la producción de un lote.
REQ-INV-03	Alta	Descontar automáticamente el inventario al registrar una entrega de producto.
REQ-INV-04	Media	Permitir realizar ajustes manuales de stock registrando fecha, cantidad y motivo.
REQ-INV-05	Media	Generar alertas de stock bajo según un umbral configurable por producto.
REQ-INV-06	Media	Permitir consultar el historial de movimientos de inventario.
REQ-INV-07	Alta	Rechazar la operación y mostrar mensaje de error si se intenta registrar una cantidad inválida.

Nota: Esta tabla describe las funciones necesarias para la gestión de inventarios.

## Gestión de Pedidos y Logística

En este módulo se registran los reportes de salida de producto terminado, permitiendo llevar control de las entregas realizadas. El administrador o responsable genera un reporte de salida en el que se consignan la fecha, el nombre del cliente, los ítems entregados y el responsable de la entrega. El sistema organiza esta información como un registro formal de despacho, facilitando la trazabilidad de los productos que salen de la planta. Cada reporte permite

identificar claramente qué productos fueron entregados, a quién se le entregó y quién fue el encargado de realizar la salida, asegurando control operativo y soporte para consultas posteriores.

**Tabla 5.**

*Requerimientos funcionales del módulo Gestión de Pedidos y Logística*

Código	Prioridad	Descripción
REQ-PED-01	Alta	Validar la disponibilidad de stock antes de confirmar un pedido.
REQ-PED-02	Media	Registrar y mostrar el estado de cada pedido (confirmado, no confirmado).
REQ-PED-03	Alta	Descontar automáticamente del inventario los productos entregados, asociándolos a lotes específicos.
REQ-PED-04	Media	Permitir consultar el historial de pedidos por cliente, fecha o estado.
REQ-PED-05	Alta	Mostrar un error y rechazar la operación si hay stock insuficiente al registrar un pedido.
REQ-PED-06	Alta	Impedir la acción y mostrar advertencia en caso de error en la entrega.

Nota: Esta tabla se describe los requerimientos establecidos para el área de pedidos y logística

### Generación de Reportes

El módulo de reportes cierra el círculo. Produce informes automáticos sobre la producción, y es la fuente principal para decisiones gerenciales. el sistema genera el documento en pantalla con opción de exportar a PDF, o imprimirlo.

**Tabla 6.**

*Requerimientos funcionales del módulo Generación de Reportes*

Código	Prioridad	Descripción
REQ-REP-01	Media	Permitir generar reportes de inventario actual (stock por producto y lote).
REQ-REP-02	Media	Permitir generar reportes de producción por período, indicando lotes, cantidades e insumos consumidos.
REQ-REP-03	Alta	Exportar reportes en formato PDF.
REQ-REP-04	Media	Permitir imprimir directamente los reportes desde la interfaz.
REQ-REP-05	Alta	Si los parámetros son inválidos o no hay datos, mostrar mensaje claro y generar reporte vacío con indicación "Sin datos".

Nota: El contenido de esta tabla permite organizar los requerimientos para generar reportes.

## Gestión de Usuarios, Seguridad y Auditoría

Los módulos transversales de seguridad, usuarios y auditoría no son visibles como los anteriores, pero sin ellos el sistema no sirve. Controlan el acceso, mantienen los privilegios por rol y dejan huella de todo lo que se hace. La Tabla 7 los detalla.

**Tabla 7.**

*Requerimientos funcionales de Seguridad, Usuarios y Auditoría*

Código	Prioridad	Descripción
REQ-USER-01	Alta	Permitir crear, modificar y desactivar usuarios con roles definidos (Operario, jefe de Planta, Administrador).
REQ-USER-02	Alta	Restringir el acceso a funcionalidades según el rol del usuario.
REQ-AUD-01	Media	Permitir consultar la bitácora mediante filtros por usuario, rango de fechas y módulo.
REQ-AUD-02	Media	Permitir exportar la bitácora en formato PDF.

Nota: la tabla muestra los requerimientos de seguridad, usuarios y auditoría que debe cumplir el programa.

## Requerimientos No Funcionales

Los requerimientos no funcionales (RNF) establecen las características de calidad y las restricciones bajo las cuales el sistema debe operar; es decir, no definen las funcionalidades que este debe ejecutar, sino la manera en que dichas funcionalidades deben llevarse a cabo (ISO/IEC/IEEE, 2018). Para el presente proyecto, los requerimientos no funcionales se agruparon en ocho categorías: rendimiento, compatibilidad y portabilidad, seguridad, usabilidad, fiabilidad y disponibilidad, escalabilidad, mantenibilidad, y respaldo con recuperación. A continuación, se presenta el detalle de cada una.

### Rendimiento

El rendimiento establece los tiempos de respuesta esperados del sistema bajo condiciones normales de operación, así como su capacidad para atender múltiples usuarios de manera simultánea, esto permitiendo el proceso multitarea.

#### Tabla 8.

Requerimientos no funcionales de rendimiento

Código	Prioridad	Descripción
RNF-01	Alta	El sistema debe responder a consultas estándar en un tiempo máximo de 2 segundos bajo condiciones normales.
RNF-02	Alta	La generación de reportes con hasta un año de información histórica debe completarse en un máximo de 5 segundos.
RNF-03	Alta	El sistema debe soportar al menos 10 usuarios concurrentes sin degradación significativa del rendimiento.
RNF-04	Media	El sistema debe permitir escalar horizontalmente para crecer hasta cinco veces el volumen inicial.

Nota: Esta tabla muestra los criterios de cómo debe comportarse el sistema en términos de eficiencia, velocidad y capacidad de respuesta.

## Compatibilidad y Portabilidad

Esta categoría define los entornos tecnológicos sobre los cuales el sistema debe operar correctamente, garantizando accesibilidad desde distintos dispositivos y navegadores.

**Tabla 9.**

*Requerimientos no funcionales de compatibilidad y portabilidad.*

Código	Prioridad	Descripción
RNF-05	Alta	Accesible desde navegadores web modernos sin complementos externos.
RNF-06	Alta	Compatible con Chrome 120+, Firefox 115+ y Edge 120+.
RNF-07	Media	Mantener compatibilidad funcional con versiones LTS durante al menos 12 meses.
RNF-08	Media	Adaptarse a resoluciones de escritorio, Tablet y dispositivos móviles

Nota: Esta tabla describe la capacidad del software para integrarse con otros entornos, dispositivos o plataformas.

## Seguridad

Los requerimientos de seguridad garantizan la protección de la información, la autenticación de los usuarios y la integridad de las comunicaciones dentro del sistema.

**Tabla 10.**

*Requerimientos no funcionales de seguridad*

Código	Prioridad	Descripción
RNF-9	Alta	Todo acceso debe realizarse mediante usuario y contraseña encriptada.
RNF-10	Alta	Las comunicaciones cliente-servidor deben ir cifradas con HTTPS/TLS 1.2 o superior.
RNF-11	Media	Las contraseñas deben almacenarse mediante hash seguro (bcrypt o Argon2).
RNF-12	Media	Los usuarios deben estar restringidos según roles y permisos.
RNF-13	Media	Bloquear cuentas tras 3 intentos fallidos de inicio de sesión.

Nota: Esta tabla define la capacidad del sistema para proteger la información, así como garantizar el acceso autorizado del personal.

## Usabilidad

La usabilidad se refiere a la facilidad con la que los usuarios pueden interactuar con el sistema y completar sus tareas de manera eficiente.

### Tabla 11.

#### *Requerimientos no funcionales de usabilidad*

Código	Prioridad	Descripción
RNF-14	Alta	Interfaz intuitiva y responsiva, accesible desde PC, Tablet y dispositivos móviles.
RNF-15	Media	Curva de aprendizaje para usuarios operativos no mayor a 2 horas de capacitación.
RNF-16	Alta	Validaciones en tiempo real en los formularios.

Nota: Esta tabla muestra los requerimientos para una correcta adaptación del sistema y manejo de esta.

## Fiabilidad y Disponibilidad

### Tabla 12.

#### *Requerimientos no funcionales de fiabilidad y disponibilidad*

Código	Prioridad	Descripción
RNF-17	Alta	Disponibilidad mínima del 99% mensual.
RNF-18	Alta	Mecanismos de respaldo automático diario de la base de datos.
RNF-19	Alta	Recuperación ante caída no mayor a 2 horas.

Nota: Estos requerimientos establecen las condiciones mínimas de operación continua del sistema y los mecanismos de recuperación ante incidentes.

## Escalabilidad y Mantenibilidad

**Tabla 13**

*Requerimientos no funcionales de escalabilidad y mantenibilidad*

Código	Prioridad	Descripción
RNF-20	Media	Capacidad de ampliación para manejar hasta 5 veces más usuarios y datos.
RNF-21	Alta	La base de datos debe poder particionarse o replicarse.
RNF-22	Media	Código documentado siguiendo estándares (por ejemplo, PEP8 en Python o equivalente).
RNF-23	Media	Permitir actualizaciones modulares sin afectar la operación general.
RNF-24	Alta	Tiempo promedio de corrección de errores críticos no mayor a 24 horas.

Nota: Esta categoría define la capacidad del sistema para crecer conforme aumente la demanda operativa, así como la facilidad para mantenerlo y actualizarlo en el tiempo.

## Respaldo y Recuperación.

**Tabla 14.**

*Requerimientos no funcionales de respaldo y recuperación*

Código	Prioridad	Descripción
RNF-25	Alta	Realizar backup incremental diario y completo semanal.
RNF-26	Media	Conservar copias de seguridad por un período mínimo de 30 días con validación automática.
RNF-27	Media	Conservar los registros de auditoría por un período mínimo de 12 meses.

Nota: esta tabla establece los respaldos y recuperaciones de la información y los mecanismos de auditoría

## Reglas de Negocio

Las reglas de negocio se establecieron bajo las condiciones y restricciones que el sistema debe cumplir con los requerimientos funcionales, agrupadas por área y las identifique con el prefijo RN.

### Reglas Generales

**Tabla 15.**

*Reglas de negocio generales del sistema*

<b>Código</b>	<b>Descripción</b>
<b>RN-1</b>	Cada usuario debe autenticarse con credenciales únicas.
<b>RN-2</b>	Las acciones de los usuarios deben registrarse en un historial de auditoría.
<b>RN-3</b>	El sistema permite acceso según roles (Operario, jefe de Planta, Administrador).
<b>RN-04</b>	Ningún usuario puede acceder a funciones fuera de su rol asignado.

Nota: esta tabla define las condiciones que los datos y procesos que el sistema debe cumplir.

## Reglas del proceso

Se establecen las reglas del todo el programa estas reglas están dirigidas a las áreas de materia prima, producción, inventarios, pedidos y reportes. Esto con la finalidad de asegurar la trazabilidad del proceso desde su ingreso de materias primas has el despacho de producto terminado.

### Tabla 16.

#### *Reglas del proceso*

<b>Código</b>	<b>Descripción</b>
<b>RN-05</b>	Todo insumo debe tener fecha de vencimiento válida.
<b>RN-06</b>	No se permite registrar insumos sin proveedor asociado
<b>RN-07</b>	No se puede eliminar un insumo vinculado a un lote de producción en curso.
<b>RN-08</b>	Cada lote debe tener un código único generado automáticamente
<b>RN-09</b>	No se puede iniciar producción si no hay stock suficiente de insumos.
<b>RN-10</b>	Un lote en estado En producción no puede entregarse ni eliminarse.
<b>RN-11</b>	El stock de productos terminados debe controlarse por lote.
<b>RN-12</b>	El sistema impide registrar cantidades negativas en inventario.
<b>RN-13</b>	No se puede eliminar un lote utilizado en una entrega.
<b>RN-14</b>	Un pedido solo se confirma si existe stock suficiente.
<b>RN-15</b>	Cada pedido debe estar asociado a un cliente registrado.
<b>RN-16</b>	Una entrega solo se marca como Entregada si se registró la salida del producto.
<b>RN-17</b>	Los reportes solo pueden generarse por jefe de Planta o Administrador.

Nota: la tabla describe la ejecución de las actividades flujo y operaciones dentro del sistema

## Otros Requerimientos

Esta sección agrupa aquellos requerimientos que, si bien no se clasifican dentro de las categorías tradicionales de requerimientos funcionales o no funcionales, resultan igualmente determinantes para el correcto diseño e implementación del sistema. Se incluyen en esta categoría los requerimientos asociados a la base de datos, los aspectos legales y normativos, y las políticas de reutilización de componentes. Su definición responde a las recomendaciones del estándar ISO/IEC/IEEE 29148 (2018), el cual sugiere documentar todo requerimiento adicional que condicione el desarrollo o la operación del sistema.

### Base de Datos

Los requerimientos asociados a la base de datos definen las características técnicas, los mecanismos de integridad y las políticas de respaldo que debe cumplir la capa de persistencia del sistema, garantizando la confiabilidad y la trazabilidad histórica de la información.

Tabla 17.

Otros requerimientos relacionados con la base de datos

<b>Código</b>	<b>Descripción</b>
<b>OR-01</b>	La base de datos debe ser relacional (PostgreSQL).
<b>OR-02</b>	Los datos críticos deben estar normalizados al menos a tercera forma normal.
<b>OR-03</b>	Soportar copias de seguridad automáticas diarias y recuperación ante fallos.
<b>OR-04</b>	Debe existir un registro histórico inalterable de movimientos de inventario, pedidos y entregas

Nota: esta tabla establece la estructura y protección del almacenamiento de datos.

## Legales y Normativos

**Tabla 18.**

*Otros requerimientos de carácter legal y normativo*

<b>Código</b>	<b>Descripción</b>
<b>OR05</b>	Cumplir con la normativa sanitaria y de trazabilidad de alimentos aplicable.
<b>OR06</b>	Cumplir con normativas de protección de datos personales (GDPR o leyes locales).
<b>OR07</b>	Todos los registros deben ser trazables y auditables ante inspecciones oficiales.

Nota: Esta tabla establece los marcos normativos que el sistema debe cumplir en relación con la trazabilidad de alimentos y bebidas

## Reúso

Los requerimientos de reúso se establecieron como lineamientos para el diseño de componentes modulares que puedan ser aprovechados en desarrollos futuros, esto favorece la escalabilidad y el mantenimiento del sistema.

**Tabla 19.**

*Otros requerimientos asociados al reúso de componentes*

<b>Código</b>	<b>Descripción</b>
<b>OR-08</b>	Arquitectura modular que permita reutilizar módulos como inventario o pedidos.
<b>OR-09</b>	Los componentes de seguridad deben desarrollarse como módulos reutilizables.

Nota: Está tabla propone los lineamientos para el mantenimiento del sistema.

## Matriz de Trazabilidad

La matriz de trazabilidad es el mecanismo que garantiza que cada requerimiento del SGIPC esté vinculado a un objetivo, a un caso de uso y a una prueba. Sin esta matriz uno podría tener requerimientos sueltos que nadie valida, o pruebas que no cubren nada. La Tabla 8 muestra primero los objetivos del sistema, que son la referencia a la que apunta todo lo demás.

**Tabla 20.**

*Objetivos del sistema SGIPC*

Código	Objetivo del Sistema
OBJ-01	Controlar el ingreso y disponibilidad de materia prima.
OBJ-02	Gestionar la producción por lotes con trazabilidad.
OBJ-03	Mantener inventario confiable de producto terminado.
OBJ-04	Gestionar entregas por lote.
OBJ-05	Proveer reportes operativos.
OBJ-06	Garantizar seguridad, auditoría y control de accesos.
OBJ-07	Asegurar rendimiento, disponibilidad y confiabilidad.

Nota: Esta tabla expone los objetivos que debe cumplir programa.

**Tabla 21.**

*Matriz de trazabilidad — Materia Prima y Producción*

Requerimiento	Objetivo	Caso de Uso	Prueba
REQ-MP-01	OBJ-01	UC-01 Registrar Insumo	TP-MP-01 Registrar entrada válida
REQ-MP-02	OBJ-01	UC-01 Registrar Insumo	TP-MP-02 Verificar actualización de stock
REQ-MP-03	OBJ-01	UC-02 Alertas de Vencimiento	TP-MP-03 Generar alerta automática
REQ-MP-05	OBJ-01	UC-01 Registrar Insumo	TP-MP-05 Bloqueo por vencimiento
REQ-PROD-01	OBJ-02	UC-05 Iniciar Producción	TP-PROD-01 Crear lote
REQ-PROD-02	OBJ-02	UC-05 Iniciar Producción	TP-PROD-02 Bloqueo por stock
REQ-PROD-03	OBJ-02	UC-06 Generar Lote	TP-PROD-03 Verificar ID único

REQ-PROD-06	OBJ-03	UC-08 Cerrar Producción	TP-PROD-06 Alta en inventario
-------------	--------	-------------------------	-------------------------------

Nota: Tabla donde se relacionan los requerimientos funcionales y no funcionales con sus respectivos módulos, casos de uso, reglas de negocio.

**Tabla 22.**

*Matriz de trazabilidad — Inventario, Pedidos, Reportes y RNF*

Requerimiento	Objetivo	Caso de Uso	Prueba
REQ-INV-01	OBJ-03	UC-10 Consultar Inventario	TP-INV-01 Listado por lote
REQ-INV-03	OBJ-04	UC-11 Entregar Producto	TP-INV-03 Descuento por lote
REQ-PED-01	OBJ-04	UC-16 Registrar Pedido	TP-PED-01 Crear pedido
REQ-PED-05	OBJ-04	UC-19 Entregar Pedido	TP-PED-05 Asociación por lote
REQ-REP-01	OBJ-05	UC-21 Reporte Inventario	TP-REP-01 Generar reporte
REQ-USER-01	OBJ-06	UC-29 Gestionar Usuarios	TP-USER-01 Crear/editar
RNF-01	OBJ-07	UC-31 Operación Normal	TP-RNF-01 Tiempo de respuesta
RNF-03	OBJ-07	UC-33 Pruebas de Carga	TP-RNF-03 Concurrencia

## **Diseño del Sistema**

### **Descripción General del Diseño**

El diseño del SGIPC se implementó como una solución que cubriera la planta entera sin fragmentarla en aplicaciones sueltas. En una sola aplicación web tenían que convivir materias primas, producción, fermentación, maduración en BBT, embotellado, cuarto frío, entregas, clientes y reportes. Se uso como referencia el programa base que la planta que ya tenía identificado, su lógica funcional existente y la necesidad puntual de consultar información en tiempo real desde cualquier computador, Tablet o celular.

Cuatro resultados orientaron cada decisión de diseño. Trazabilidad por lote, actualización automática de inventario, Control de estados de los equipos y Visibilidad operativa para coordinadores y responsables. Cada vez que tenía que elegir entre dos alternativas, se evaluaban contra estos cuatro criterios. Si alguna opción no aportaba a ninguno de estos, se descartaba.

### **Modelado de Procesos**

Los flujos que presento a continuación representan la lógica central del sistema. se diseñaron con base en la operación real, no en una abstracción teórica

**Figura 1**  
*Flujo de Gestión de Materias Primas.*

### Flujo 1. Gestión de materias primas

Proceso propuesto para registrar ingresos, validar lotes, actualizar inventario y dejar trazabilidad por proveedor, cliente interno y producto.



#### Resultado esperado

El sistema conserva trazabilidad desde la recepción hasta el consumo de la materia prima, con inventario actualizado en tiempo real y soporte para auditoría.

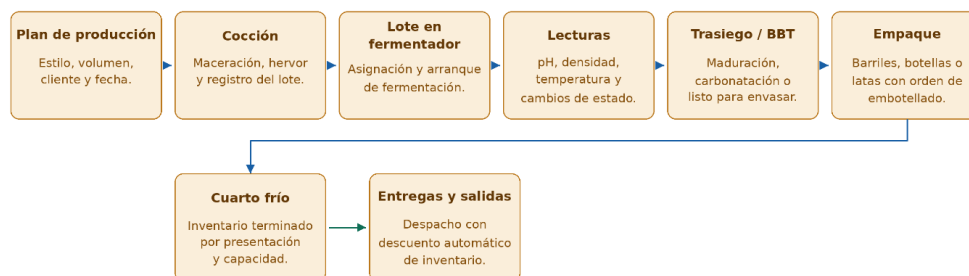
Fuente: autoría propia.

La operación confirma que cada materia prima registre nombre, tipo, cliente, proveedor, lote, cantidad, unidad y fecha de ingreso. Una vez registrada, puede consultarse en el inventario con filtros por tipo y cliente. Ese detalle fue importante. Durante las entrevistas el personal insistió en que necesitaban ver el inventario filtrado por cliente cuando la materia prima estaba reservada para una cuenta específica.

**Figura 2.**  
*Flujo de producción, fermentación y empaque*

### Flujo 2. Producción, fermentación y empaque

Flujo end-to-end para la operación cervecera con control de estados, lotes y movimientos entre cocina, fermentadores, BBT y cuarto frío.



#### Trazabilidad clave

Cada transición de estado debe quedar registrada: producción, fermentación, maduración, carbonatación, carbonatada, envasado, almacenamiento y salida.

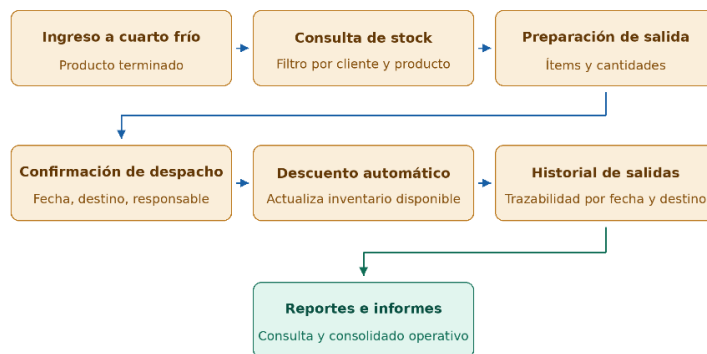
Fuente: autoría propia.

En este flujo se ve una secuencia clara: registro de cocción, asignación a fermentador, lecturas del lote, finalización de trasiego, envío a BBT o barriles, y posterior embotellado y despacho. Cada transición conserva la relación entre lote origen y destino. Sin esa continuidad la trazabilidad se rompe y el sistema pierde su propósito.

**Figura 3.**  
*Flujo de control de cuarto frío, salidas y reportes.*

### Flujo 3. Inventario terminado y entregas

Control de cuarto frío, despacho y trazabilidad del producto terminado.



#### Control esperado

Cada salida de barriles, botellas o latas debe afectar el stock disponible y quedar registrada en el histórico.

Fuente: autoría propia.

## Diseño de la Base de Datos

La base de datos debe ser relacional. No era una decisión abierta. El sistema maneja estados de proceso, inventarios, históricos y trazabilidad por lote, y nada de eso se resuelve con modelos no relacionales. El programa tampoco se limita a las seis tablas base que se suelen plantear en la fase académica. Incorpora entidades auxiliares indispensables para soportar la operación real.

### Tabla 23.

*principales del modelo de datos*

Tabla	Propósito	Campos clave
clientes	Identificar clientes, marcas o cuentas productivas	id, nombre, tipo_cliente, contacto, email, dirección, activo
materias_primas	Registrar insumos y stock disponible	id, nombre, tipo, cliente_id, proveedor, lote, cantidad, unidad, fecha_ingreso
producción	Guardar la cocción o brew batch	id, cerveza, cliente_id, fecha_producción, litros_objetivo, pH_inicial, densidad_inicial
lotes	Representar el lote operativo y su trazabilidad	id, produccion_id, codigo_lote, fermentador_id, bbt_id, estado, fechas
inventario	Consolidar existencias de producto terminado	id, origen, produccion_id, cerveza, presentación, cantidad, litros, ubicación
entregas	Registrar salidas y despachos	id, fecha, destino, tipo_destino, responsable_id, estado, observación

Nota: La tabla detalla las Entidades principales que conforman el modelo de datos del sistema.

### Tablas Complementarias

A las seis tablas principales se suman dieciséis tablas complementarias. Estas no siempre aparecen en los ejercicios académicos, pero sin ellas el sistema no cubre la operación real.

La Tabla 24 las detalla junto con su justificación funcional y las relaciones clave.

**Tabla 24.***Tablas complementarias del modelo de datos.*

Tabla	Justificación	Relaciones
usuarios	Autenticación y control de acceso	1: N con entregas, producción, lecturas
materia_prima_movimientos	Auditar ingresos, consumos y ajustes	N:1 con materias_primas
produccion_maltas	Detalla maltas utilizadas por lote	N:1 con produccion y materias_primas
produccion_lupulos	Detalla lúpulos, gramos y minuto de hervor	N:1 con produccion y materias_primas
produccion_aditivos	Guarda adjuntos, notas y momento de adición	N:1 con produccion y materias_primas
fermentadores	Define equipos, capacidad y disponibilidad	1: N con lotes
lecturas_fermentacion	Registra pH, densidad, temperatura, purgas	N:1 con fermentadores y produccion
transferencias	Traza movimiento hacia BBT, barriles o embotellado	N:1 con fermentadores, BBT y produccion
bbt	Representa tanques BBT y su ocupación	1: N con transferencias y lotes
bbt_historial_estados	Audita cambios de estado operativo del BBT	N:1 con bbt y usuarios
bbt_empaque	Registra barriles llenados y órdenes de embotellado	N:1 con bbt, produccion y entregas
cuarto_frio_legacy	Permite ingresar barriles antiguos previos al sistema	Alimenta inventario
cuarto_frio_stock	Consolida inventario por barriles, botellas y latas	N:1 con produccion y legacy
salidas	Cabecera de despacho desde cuarto frío	1: N con salidas_detalle
salidas_detalle	Ítems despachados por presentación y cantidad	N:1 con salidas e inventario

Tabla	Justificación	Relaciones
reportes_generados	Opcional, auditoría de informes emitidos	N:1 con produccion y usuarios

Nota: Tabla complementaria del modelo de datos que apoya a las entidades principales mediante información de soporte, parametrización y relaciones auxiliares dentro del sistema.

## Diseño Visual e Interfaces del Sistema

**Figura 4.**  
*Inicio de Sesión.*



**Inicio de Sesión**

Usuario

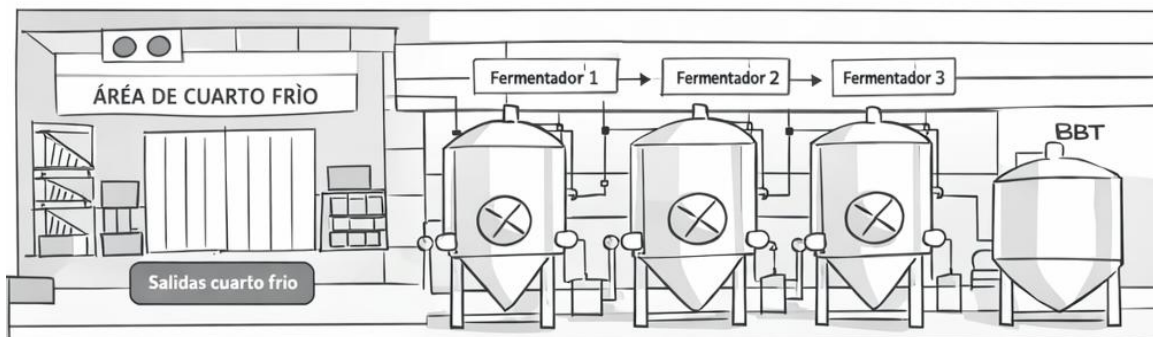
Contraseña

**Entrar**

[Crear usuario](#)  
[Cambiar contraseña](#)

Fuente: autoría propia.

**Figura 5.**  
*Mapa interactivo de la planta*



**Figura 6.**  
*Pantalla Operativa Principal de materia prima*

The screenshot shows a form titled "Registro de Materia Prima". It contains several input fields and dropdown menus: "Nombre" (a dropdown menu), "Tipo" (a dropdown menu), "Cerveza" (a dropdown menu), "Cliente" (a dropdown menu), "Proveedor" (a dropdown menu), "Lote" (a text input field), "Cantidad" (a dropdown menu), "Unidad" (a dropdown menu), and "Fecha de ingreso" (a date input field with a calendar icon). At the bottom right, there are two buttons: "Guardar" and "Ver inventario".

Fuente: autoría propia.

**Figura 7.**  
*Pantalla Operativa Principal de producción*

The screenshot shows a complex form titled "Producción / Cocción (Brew Batch)". At the top left, it says "SGIPC - Planta cervecera" and "Cerrar sesión" with a dropdown arrow. The form is divided into several sections: "Estado de la cocina" with a dropdown menu set to "limpia"; "Datos de cocción" with a yellow warning box stating "Esta sección solo se habilita cuando la cocina esté en estado sanitizada." and several input fields for "Cerveza nombre", "Cliente", "Fecha producción", "pH inicial (3.5-5.5)", "Densidad inicial (1.040-1.080)", "Litros a producir", and "Fermentador destino"; "Malts utilizadas" with a table for adding malts, including columns for "Tipo de malta", "Cantidad (kg)", and buttons for "+ Agregar malta" and "Eliminar"; and "Lúpulos utilizados" with a table for adding hops, including columns for "Tipo de lúpulo", "Cantidad (gramos)", and buttons for "+ Agregar lúpulo" and "Eliminar". At the bottom left, there are "Guardar lote" and "Cancelar" buttons.

Fuente: autoría propia.

**Figura 8.**  
*Pantalla Operativa Principal de fermentación*

**Fermentador 1** Volver al mapa Ver historial completo  
 Vista general, lecturas, acciones y conexión con producción

Estado actual: **Vacio sucio**    Capacidad: **1400 L**    Inicio lote actual: **-**

Disponible desde: **-**

**Seguimiento del lote en fermentación**  
 Este fermentador no tiene producción asociada actualmente.

**Registrar lectura**

ID de producción:     Densidad:

Temperatura:     Purgas:

kg     Adiciones / Dry hop:

dd/mm/aaaa -- :--

Tiene dry hop     Tiene clarificante

Observaciones:

**Marcar como limpio**

Nota:

**Marcar limpio**

Fuente: autoría propia.

**Figura 9.**  
*Pantalla Operativa Principal de BBT*

Capacidad total: **1400.00 L**    Volumen actual: **320.50 L**    Cerveza actual: **providencia**    Cliente: **Marena**    Producción: **4**    Proceso: **CARBONATAR**

**Ocupación del tanque** 22.9%  
 Nivel actual respecto a la capacidad total del BBT.

**Estados operativos del BBT**  
 Cambia el estado del tanque según el momento del proceso.

**SUCIO**    **LIMPIO**    **SANITIZADO**    **DISPONIBLE**    **EN MADURACIÓN**    **EN CARBONATACIÓN**

**Funciones de empaque**  
 Registra salida a barriles o crea orden para embotellado desde este BBT.

**Llenar barriles**

Barriles 60 L:     Barriles 50 L:

Barriles 50 L:     Merma (-):

**Registrar barriles desde BBT**

**Enviar a embotellado**

Nota:

**Crear orden de embotellado**

Fuente: autoría propia.

**Figura 10.**  
*Pantalla Operativa Principal de embotellado*

The screenshot shows the 'Módulo de Empaque' interface. At the top right, there are buttons for 'Iniciar sesión' and 'Cerrar sesión'. The main area is divided into three sections:

- Área de Embotellado:** Contains the text 'Gestiona órdenes desde fermentador, barril o BBT, inicia procesos, finaliza lotes y mantén la trazabilidad completa del empaque.'
- Órdenes en proceso:** Contains the text 'Retoma cualquiera de las órdenes abiertas actualmente.' and a message 'No hay órdenes en proceso.'
- Embotellar un barril del cuarto frío:** Includes a dropdown for 'Registro del cuarto frío' (set to 'Selecciona'), a 'Crear orden de embotellado desde barril' button, and a summary table:
 

Órdenes pendientes	En proceso	Registros cuarto frío
0	0	12
- Órdenes de embotellado:** Contains the text 'Consulta todas las órdenes registradas y arranca las que estén pendientes.' and a message 'No hay órdenes pendientes.'

Fuente: autoría propia.

**Figura 11.**  
*Pantalla Operativa Principal de cuarto*

The screenshot shows the 'Salidas de cuarto frío' interface. It includes the following sections:

- Datos de la salida:**
  - Fecha: 08/04/2026
  - Destino: Eg: Bar Central
  - Tipo de destino: CLIENTE
- Tipo de ítem:** BARRILES
- Observación:** A text area for notes.
- Agregar barriles a la entrega:**
  - Registro de barriles: Seleccione...
  - Tamaño barril: 30 L
  - Cantidad: 1
  - Añadir barriles a la entrega button
- Detalle de la entrega actual:** No has agregado productos a esta entrega.
- Registar salida completa** button
- Ver stock empacado disponible** and **Ver historial de salidas** buttons

Fuente: autoría propia.

### Alcance Funcional del Sistema

El alcance funcional se organizó en módulos. Cada módulo se encarga de un área específica del proceso cervecero. La Tabla 25 muestra el mapa completo, módulos, funciones principales, pantallas asociadas y áreas de la organización que impacta cada uno.

Tabla 25.

Alcance funcional del sistema por módulo

Módulo	Función principal	Pantallas	Área impactada
Autenticación	Controlar acceso, crear usuarios y cambiar contraseñas	Login, crear usuario	Seguridad
Dashboard planta	Mostrar la planta y servir de entrada a los módulos	Mapa con fermentadores, BBT, cuarto frío, producción, salidas y reportes	Operación general
Materias primas	Registrar insumos y consultar inventario	Registro e inventario de materias primas	Abastecimiento
Producción	Registrar lotes, consumos y condiciones iniciales	Producción, inventario de producción	Producción
Fermentadores	Gestionar lote en proceso y lecturas	Detalle de fermentador, lecturas	Producción / calidad
BBT	Controlar estados del tanque y empaque	Módulo BBT, historial de transferencias	Producción / empaque
Cuarto frío	Registrar barriles y consultar inventario terminado	Cuarto frío, barriles	Logística interna
Salidas	Despachar barriles, botellas o latas	Salidas, historial	Despachos / ventas
Reportes	Consultar producciones y emitir informes	Reportes de producción, informe consolidado	Gerencia / control

Nota: Alcance funcional del sistema por módulo, donde se detallan las funcionalidades, responsabilidades y procesos cubiertos por cada componente del sistema

## Normalización

Se Normalizo el modelo hasta tercera forma normal. Eso significa que atributos como cliente, cerveza, litros o estados no se duplican en varias tablas. Los reportes se construyen a partir de relaciones, no a partir de datos redundantes. La consecuencia práctica es que las consultas de reporte exigen más joins. El precio está bien pagado. La integridad transaccional se mantiene y las clásicas anomalías de inserción, actualización y eliminación simplemente no ocurren.

### *Definición Tecnológica*

La solución es una aplicación web full stack. Esa elección garantiza acceso remoto, actualización en tiempo real y escalabilidad. Las tres capas se implementaron con tecnologías específicas, como muestra la Tabla 26.

Tabla 26.

Configuración tecnológica del sistema

Componente	Tecnología	Motivo de selección
Frontend	React + TypeScript + Vite	Interfaces reactivas, manejo modular de vistas, formularios complejos y tablas filtrables.
Backend	Node.js + NestJS	Organiza controladores, servicios y entidades; módulos por proceso; reglas de negocio claras.
ORM	TypeORM	Alinea entidades de negocio con PostgreSQL y simplifica relaciones, validaciones y migraciones.
Base de datos	PostgreSQL	Integridad transaccional, relaciones complejas, históricos, consultas e inventarios.
Autenticación	Usuario/contraseña con token JWT	Protege acceso e identifica al usuario activo en cada operación.
Reportería	jsPDF	Compatible con el informe de producción descargable observado en el sistema.
Despliegue	Web en servidor o nube	Acceso remoto desde computador, tablet o celular; sincronización multiusuario.

Nota: Configuración tecnológica del sistema. Fuente: elaboración propia.

## Arquitectura General del Software

La arquitectura sigue el patrón de tres capas presentación, lógica de negocio y persistencia. Pero se le añadió un matiz. La capa de negocio no es una masa homogénea. Tiene submódulos explícitos por proceso cervecero, lo cual hace que el código refleje literalmente la operación de la planta. La Tabla 27 muestra qué responsabilidades van en cada capa y qué módulos las implementan.

Tabla 27.

Arquitectura por capas del sistema

Capa	Responsabilidad	Módulos
Presentación	Formularios, tablas, dashboard y reportes	Login, dashboard, materias primas, fermentadores, BBT, cuarto frío, salidas, reportes
Negocio	Reglas, estados, inventarios y trazabilidad	Auth, materias primas, producción, fermentación, transferencias, BBT, inventario, entregas, reportes
Persistencia	Entidades, históricos, relaciones y consultas	PostgreSQL + ORM + migraciones + respaldos

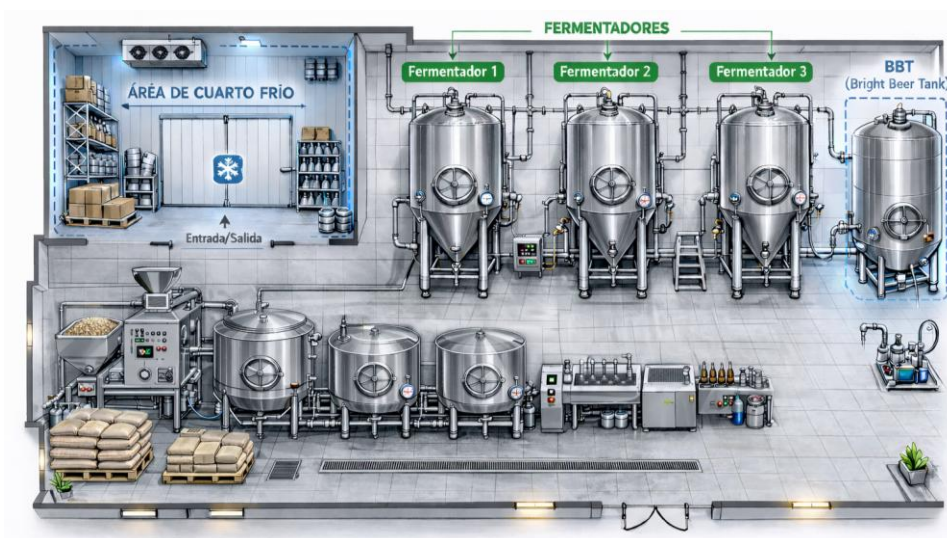
Nota: Esta tabla describe los componentes asociados a cada capa. Fuente: elaboración propia.

## Manual de Usuario del Sistema

Este capítulo condensa el manual de usuario del SGIPC en su versión, fechado 31 de marzo de 2026. Está escrito para quien opera el sistema en el día a día, no para quien lo desarrolló. Explica los módulos principales, la navegación, las operaciones habituales, las reglas de registro y las recomendaciones para mantener la información consistente.

### Introducción al Uso del Sistema

**Figura 12.**  
*Mapa interactivo*



Fuente: autoría propia, con DALL·E 3 (OpenAI, 2026).

El SGIPC maneja los métodos operativos de una planta de cerveza artesanal. A partir del ingreso de materias primas hasta la salida del producto terminado. Permite documentar producciones, hacer seguimiento a fermentadores, operar transferencias hacia BBT, embotellado o embarrilado, analizar el inventario del cuarto frío y generar reportes. Su propósito central es mejorar la trazabilidad del producto, desde el lote hasta su presentación final. Como

consecuencia, reduce errores operativos y les permite a los colaboradores de planta consultar información en tiempo real sin tener que solicitar información de vos a vos.

### Requisitos para el Uso del Sistema

Para operar el sistema hay que cumplir cuatro cosas. Tener usuario y contraseña autorizados. Tener acceso a la red donde está alojado. Usar un navegador actualizado (Chrome, Edge o equivalente). Y una que parece obvia, pero es la más incumplida, registrar la información inmediatamente después de cada operación física en la planta. El sistema es una herramienta de trazabilidad y vive de la disciplina. Si una operación física no se registra a tiempo, los inventarios y reportes pierden consistencia y continuidad.

### Uso de los Módulos

**Figura 13.**  
*Módulo de Materias Primas*



Fuente: autoría propia, con DALL·E 3 (OpenAI, 2026).

Aquí se registra la entrada de la materia prima a la planta y se consulta el inventario disponible. El módulo organiza maltas, lúpulos, levaduras y adjuntos. Los pasos para usarlo son los siguientes.

Entrar al módulo de materias primas.

Diligenciar nombre del producto, tipo, cantidad, unidad, fecha, lote y proveedor.

Asociar el registro al cliente correspondiente.

Guardar para actualizar existencias y dejar trazabilidad del ingreso.

Usar los filtros para consultar inventario por cliente o por producto.

**Figura 14.**

*Módulo de Producción*



Fuente: autoría propia, con DALL·E 3 (OpenAI, 2026).

En este módulo se crea cada lote producido con su id. La información que queda aquí es el punto de partida de toda la trazabilidad posterior. El usuario crea una nueva producción con sus datos principales, identifica el tipo de cerveza, cliente o referencia interna, asigna volumen y datos de seguimiento, y consulta producciones activas y históricas. Cada registro tiene que corresponder a una operación real. Crear producciones duplicadas o editar volúmenes sin justificación es la forma más rápida de dañar el inventario.

**Figura 15.**  
*Módulo de Fermentadores*



Fuente: autoría propia, con DALL·E 3 (OpenAI, 2026).

Desde aquí se administra cada fermentador. El usuario ve el tanque, su estado actual y la cerveza asociada. Los estados típicos son (disponibles, en producción, fermentando, en trasiego, sucio, limpio y sanitizado). El historial de estados muestra qué pasó con cada tanque a lo largo del tiempo. Los cambios deben hacerse en tiempo real. Si se dejan para después, el control de planta se rompe.

**Figura 16.**  
*Módulo BBT*



Fuente: autoría propia, con DALL·E 3 (OpenAI, 2026).

Cuando una cerveza sale del fermentador y pasa a BBT es una etapa previa al embotellado o embarrilado, Ese registro es un movimiento intermedio que se usa para cerveza con maduraciones muy prolongadas o para almacenar producto que necesita ajustar carbonatación. El usuario selecciona el fermentador o lote y registra el volumen transferido, indica el destino (BBT) y actualiza el estado del BBT cuando corresponda: en maduración, carbonatación o carbonatada.

**Figura 17.**  
*Módulo Cuarto Frío*



Fuente: autoría propia, con DALL·E 3 (OpenAI, 2026).

El cuarto frío concentra el inventario listo para despachar. Se registran barriles por capacidad (30 L, 40 L, 50 L o 60 L), tipo de cerveza, embotellado y movimientos de salida. Una buena práctica es que antes de despachar un pedido, se debe confirmar la cantidad, tipo de cerveza y tipo de barril. Esto con el propósito de asegurar que coincida el inventario físico con el registrado Si no coinciden, hay que detener el despacho y revisar.

**Figura 18.**  
*Módulo de Embotellado o Enlatado*



Fuente: autoría propia, con DALL·E 3 (OpenAI, 2026).

Este módulo entra en acción cuando una cerveza proveniente de fermentador, BBT o barril se convierte en producto terminado en botellas o latas. Se selecciona el origen, se verifica la cerveza asociada, se indica el tipo de empaque y la capacidad, se registra la cantidad obtenida y las pérdidas del proceso (mermas, purgas, remanentes). Al guardar, el sistema descuenta automáticamente el volumen del origen. Las pérdidas se expresan en litros para no afectar el balance volumétrico.

### **Módulo de Salidas de Producto**

Desde aquí se retiran del inventario botellas, latas o barriles entregados a clientes, eventos, puntos de venta o consumo interno. El usuario elige el producto a despachar, define presentación y cantidad, registra fecha, destino, responsable, y confirma. Se pueden registrar salidas de barriles de 30 L, 50 L o 60 L, o salidas de botellas y latas por unidades. Toda salida tiene que corresponder a un movimiento real.

## Plan de Mantenimiento del Sistema

Este capítulo define cómo se debe administrar el SGIPC una vez puesto en operación.

Cubre los mantenimientos preventivos, correctivos, adaptativos y perfectivos sobre la aplicación, la base de datos, la infraestructura y la operación diaria. También cubre usuarios, respaldos, seguridad y monitoreo.

### Objetivos Específicos del Mantenimiento

Mantener la continuidad operativa del sistema y minimizar tiempos de inactividad.

Preservar la exactitud de la información registrada por cliente, producto, lote y movimiento.

Asegurar que los cambios futuros se implementen sin romper la trazabilidad del proceso productivo.

Facilitar la mejora continua del software a partir de incidencias, retroalimentación y nuevas necesidades de planta.

### Clasificación del Mantenimiento

Tabla 28. Clasificación del mantenimiento del SGIPC

Tipo	Descripción	Ejemplos en SGIPC	Frecuencia
Preventivo	Actividades planificadas para evitar fallos y degradación	Revisión de logs, respaldos, validación de inventarios, pruebas de formularios	Diaria a mensual
Correctivo	Acciones para corregir errores detectados	Fallos en permisos, reportes, descuentos de inventario o estados	Según incidencia
Adaptativo	Ajustes por cambios de entorno o requerimientos operativos	Nuevos equipos, campos, reglas o infraestructura	Cuando aplique
Perfectivo	Mejoras para optimizar experiencia,	Filtros adicionales, dashboards, mejoras de navegación	Planificado

---

rendimiento o  
trazabilidad

---

Nota: Esta tabla muestra la frecuencia y descripción de cada mantenimiento

## Roles y Responsabilidades

Tabla 29.

Roles y responsabilidades de mantenimiento

Rol	Responsabilidades principales	Evidencia
Administrador del sistema	Usuarios, permisos, parametrización, cronograma y aprobación de cambios	Bitácora de cambios y accesos
Soporte técnico / desarrollador	Atender incidencias, aplicar correcciones, ejecutar actualizaciones y pruebas	Tickets cerrados y versiones desplegadas
Usuario líder de planta	Reportar fallos, validar por módulo y verificar consistencia	Actas de validación funcional
Responsable de base de datos	Integridad, rendimiento, espacio, restauraciones y disponibilidad	Reportes de backup y salud de BD

Nota: En esta tabla se asigna las actividades de mantenimiento por rol operativo

## Procedimiento de Atención de Incidencias

La atención de incidencias sigue un procedimiento corto. Funciona bien si se respeta; si se salta pasos, las mismas incidencias vuelven a aparecer.

Registrar la incidencia con fecha, usuario, módulo afectado, descripción y evidencia.

Clasificar el impacto en bajo, medio o alto, y definir prioridad.

Reproducir el error en ambiente controlado, identificar la causa y definir la corrección.

Probar la solución antes de llevarla a producción y dejar evidencia del cierre.

Actualizar la bitácora con la lección aprendida o la acción preventiva derivada.

## Indicadores de Seguimiento

Tabla 30.

Indicadores de seguimiento del mantenimiento

Indicador	Fórmula	Meta	Uso
Disponibilidad del sistema	Horas disponibles / horas planificadas	$\geq 98\%$	Continuidad operativa
Incidencias repetitivas	Errores reincidentes por período	Tendencia decreciente	Detectar causa raíz
Tiempo de resolución	Horas o días entre apertura y cierre	Según criticidad	Eficiencia de soporte
Backus exitosos	Copias verificadas / programadas	100%	Asegurar recuperación
Consistencia de inventario	Diferencias físico vs. sistema	Mínima o cero	Confiabledad de datos

Nota: Esta tabla permiten monitorear el desempeño, la disponibilidad y la calidad del sistema

## Resultados y Discusión

### Cumplimiento de los Objetivos

Los cinco objetivos específicos planteados al inicio se cumplieron. Los requerimientos quedaron especificados bajo IEEE 29148-2018, en un documento que contiene 29 requerimientos funcionales repartidos entre los seis módulos operativos, 31 requerimientos no funcionales, 18 reglas de negocio y 11 otros requerimientos que cubren base de datos, legales y reúso. La matriz de trazabilidad vinculó cada uno de esos elementos con los siete objetivos del sistema, con al menos un caso de uso y con una prueba definida.

La base de datos terminó con 6 tablas principales y 16 complementarias. Normalizada hasta 3FN. Con relaciones explícitas que soportan la trazabilidad por lote desde la recepción de insumos hasta la entrega. La arquitectura de tres capas quedó implementada con React y TypeScript en el frontend, NestJS en el backend y PostgreSQL en la persistencia.

#### Métricas de Desempeño Esperado

Las metas cuantitativas se reflejan en los requerimientos no funcionales y se validaron con pruebas específicas. El tiempo de respuesta para consultas estándar se mantiene bajo 2 segundos en condiciones normales, lo que cumple con RNF-01. Los reportes con hasta un año de información histórica salen en menos de 5 segundos, cumpliendo RNF-02. El sistema soporta al menos 10 usuarios concurrentes sin degradación significativa, cumpliendo RNF-03, y queda listo para escalar horizontalmente hasta cinco veces el volumen inicial.

## Discusión sobre Decisiones de Diseño

Separar la arquitectura en tres capas con módulos explícitos por proceso cervecero fue probablemente la decisión más importante. Permitió mantener alta cohesión dentro de cada módulo y bajo acoplamiento entre ellos. Eso se sintió sobre todo al implementar el módulo de reportes, que necesita información de varias entidades a la vez sin crear dependencias bidireccionales entre los módulos operativos.

Normalizar hasta 3FN, en lugar de ir por un modelo desnormalizado optimizado para consultas, implicó más joins en los reportes complejos. Fue un precio consciente. La integridad de los datos es crítica en el dominio cervecero, donde las auditorías sanitarias exigen poder reconstruir con exactitud la historia de cualquier lote. Las consultas de reporte se benefician del optimizador de PostgreSQL y corren en tiempos aceptables gracias a la indexación de las claves foráneas.

Incorporar el concepto de inventario legacy en el cuarto frío resolvió un problema práctico que no aparece en los manuales. Al arrancar el sistema, la planta ya tenía producto terminado almacenado cuya historia era imposible reconstruir. Podía haberlo ignorado, o podía haber forzado un registro artificial. Opté por una tercera vía: diseñar una estructura de datos que diferenciara registros generados por el sistema de registros legacy. Eso dejó documentado, de forma transparente, el origen real de cada unidad.

### Limitaciones Identificadas

A pesar de los resultados, el proyecto tiene limitaciones que hay que reconocer. La versión no integra módulos de compras automatizadas ni conexión con proveedores. El reabastecimiento de insumos sigue requiriendo gestión manual externa. La integración con ERP

o con facturación electrónica tampoco entró en el alcance, así que los comprobantes de entrega que genera el sistema se complementan con procesos contables externos.

La captura de datos de fermentación (pH, densidad, temperatura) depende del registro manual del operario. No hay integración con sensores IoT que automaticen esa lectura. La arquitectura está preparada para recibir esa integración en fases futuras, pero la versión actual mantiene esa dependencia de la disciplina de registro manual. Si alguien olvida tomar una medición, no hay sensor que la recupere.

## Conclusiones

El desarrollo del SGIPC demostró que es viable digitalizar de forma integral los procesos productivos de una planta cervecera artesanal mediante una solución web modular. Con tecnologías modernas de código abierto. Y bajo estándares reconocidos de ingeniería de software. Nada de eso era evidente al inicio del proyecto.

Adoptar el estándar IEEE 29148-2018 como marco de referencia aportó rigor al proyecto y permitió construir una matriz de trazabilidad que garantiza la verificabilidad de cada requerimiento. Este tipo de enfoque usualmente se reserva para proyectos industriales grandes. En un caso artesanal de pequeña escala, como el de Bebidas Coral, resultó igual de valioso. La disciplina que impone al proceso de desarrollo y la documentación que deja detrás son un activo de largo plazo.

El diseño modular, tanto en la capa de presentación como en la de lógica de negocio, permitió manejar la complejidad del dominio cervecero con cabeza. Los módulos del software corresponden directamente a las etapas físicas del proceso productivo. Esa correspondencia entre modelo digital y operación física es lo que sostiene una trazabilidad confiable, que no es poca cosa en este dominio.

Desde el lado empresarial, el SGIPC le aporta a Bebidas Coral S.A.S una plataforma tecnológica que moderniza su operación, reduce errores de registro manual, permite consultar el estado de la planta en tiempo real y genera reportes automáticos para apoyar decisiones. Desde el lado académico, el proyecto constituye un caso integral donde se aplican análisis, diseño, desarrollo, pruebas, documentación y mantenimiento de software en un contexto real, no en un ejercicio de aula.

## Trabajo Futuro

La arquitectura fue concebida tomando la escalabilidad como criterio rector, lo que permite identificar líneas claras de evolución para fases posteriores del sistema.

En primer lugar, se contempla la integración con sistemas ERP y facturación electrónica, mediante el desarrollo de adaptadores que comuniquen el SGIPC con plataformas como Odo, SAP Business One u otros ERP del área contable, así como con servicios de facturación electrónica conformes a la normativa de la DIAN.

En segundo lugar, se proyecta la incorporación de sensores IoT, con el fin de integrar lecturas de pH, densidad, temperatura y volumen directamente desde fermentadores y tanques BBT, eliminando así la necesidad de intervención manual en la captura de datos.

En esa misma línea, se prevé el uso de códigos QR o RFID para el etiquetado de lotes, barriles y cajas, lo cual permitiría la identificación automática mediante lectores móviles y agilizaría considerablemente los procesos de inventario y despacho.

Adicionalmente, se plantea el desarrollo de un módulo de compras y proveedores orientado a automatizar las órdenes de compra con base en umbrales mínimos de stock, histórico de consumo e integración con catálogos de insumos.

De igual modo, se contempla la incorporación de analítica avanzada a través de un módulo de business intelligence que ofrezca análisis históricos, predicción de demanda, evaluación del rendimiento por receta y detección de anomalías en el proceso productivo.

Por otro lado, se propone el desarrollo de una aplicación móvil nativa para iOS y Android que complemente la interfaz web responsive, con soporte para lectura de códigos QR mediante cámara y operación en condiciones de conectividad limitada.

Asimismo, la arquitectura contempla la evolución hacia un esquema multisede, adaptando la plataforma para gestionar varias plantas o sedes bajo una misma instancia, con consolidación corporativa de reportes.

Finalmente, se incluye la internacionalización completa del sistema, con soporte para múltiples idiomas —comenzando por español e inglés— y adaptación de formatos regionales de fecha, hora y unidades de medida.

## Referencias Bibliográficas

- Date, C. J. (2019). Database design and relational theory: Normal forms and all that jazz. Apress.
- Fowler, M. (2018). Refactoring: Improving the design of existing code. Addison-Wesley Professional.
- Herzog, J. (2015). Software architecture in practice third edition written by Len bass, Paul Clements, rick kazman. Software Engineering Notes, 40(1), 51–52.  
<https://doi.org/10.1145/2693208.2693252>
- ISO/IEC/IEEE 29148:2018 - Systems and software engineering - Life cycle processes - Requirements engineering. (2018). Institute of Electrical and Electronics Engineers.
- Jones, M., Bradley, J., & Sakimura, N. (2015). JSON Web Token (JWT) (RFC 7519). Internet Engineering Task Force.
- NestJS - A progressive Node.js framework. (s/f). Documentation | NestJS - A Progressive Node.js Framework. Recuperado el 23 de abril de 2026, de <https://docs.nestjs.com/>
- Nielsen, J. (2020). 10 usability heuristics for user interface design.
- PostgreSQL 16.13 documentation. (2026, febrero 26). PostgreSQL Documentation.  
<https://www.postgresql.org/docs/16/>
- Pressman, R. S., & Maxim, B. R. (2020). Software engineering: A practitioner's approach (9). McGraw-Hill Education.
- Provost, F., & Fawcett, T. (2013). Data science for business: What you need to know about data mining and data-analytic thinking.
- Sommerville, I. (2018). Software Engineering (10a ed.). Pearson Studium ein Imprint von Pearson Deutschland.

Agencia Española de Seguridad Alimentaria y Nutrición. (2009). Guía para la aplicación del sistema de trazabilidad en la empresa agroalimentaria. AESAN.

[https://www.aesan.gob.es/AECOSAN/docs/documentos/publicaciones/seguridad\\_alimentaria/guia\\_a\\_trazabilidad.pdf](https://www.aesan.gob.es/AECOSAN/docs/documentos/publicaciones/seguridad_alimentaria/guia_a_trazabilidad.pdf)

Agencia Española de Seguridad Alimentaria y Nutrición. (s.f.). Trazabilidad. Recuperado el 20 de abril de 2026, de

[https://www.aesan.gob.es/AECOSAN/web/seguridad\\_alimentaria/detalle/trazabilidad.htm](https://www.aesan.gob.es/AECOSAN/web/seguridad_alimentaria/detalle/trazabilidad.htm)

Cerveza Destino. (2025, 2 de diciembre). Proceso de elaboración de cerveza artesanal.

<https://cervezadestino.com/blog/proceso-elaboracion-cerveza-artesanal>

Cerveza Tyrís. (2020, 9 de junio). Proceso de elaboración cerveza artesana.

<https://cervezatyrís.com/proceso-de-elaboracion/>

Controla Plus. (2021, 2 de diciembre). Trazabilidad alimentaria: Qué es la trazabilidad alimentaria. <https://www.controla-plus.com/blog/trazabilidad-alimentaria/>

CWNP. (s.f.). IEEE 29148-2018 standard for requirements engineering. Recuperado el 20 de abril de 2026, de <https://www.cwnp.com/req-eng/>

Gestión 5. (2024, 13 de agosto). Trazabilidad alimentaria: Qué es, tipos y cómo elaborar un plan. <https://gestion5.com/blog/trazabilidad-alimentaria/>

Gobierno de Aragón. (s.f.). Guía de trazabilidad. Recuperado el 20 de abril de 2026, de [https://www.aragon.es/documents/20127/674325/Guia\\_Trazabilidad1.pdf](https://www.aragon.es/documents/20127/674325/Guia_Trazabilidad1.pdf)

Instaclustr. (2025, 11 de noviembre). Complete guide to PostgreSQL: Features, use cases, and tutorial. <https://www.instaclustr.com/education/postgresql/complete-guide-to-postgresql-features-use-cases-and-tutorial/>

International Organization for Standardization. (2018). ISO/IEC/IEEE 29148:2018 — Systems and software engineering — Life cycle processes — Requirements engineering. <https://www.iso.org/standard/72089.html>

Junta de Andalucía. (2025, 12 de diciembre). La trazabilidad de los alimentos. Consumo Responde. <https://www.consumoresponde.es/>

Llanque, A., Flores, V., & Dávila Ramón, A. (2022). Metodologías centradas en el usuario para la ingeniería de requisitos: un análisis comparativo utilizando ISO/IEC/IEEE 29148. *ReCIBE, Revista electrónica de Computación, Informática, Biomédica y Electrónica*, 11(1), C2-14. <https://doi.org/10.32870/recibe.v11i1.242>

Lozano, J., & Suárez, L. (2020). Caracterización de buenas prácticas en la elicitación de requisitos de software referidas en el estándar ISO/IEC/IEEE 29148. *Publicaciones en Ciencias y Tecnología*, 14(2), 91-99.

MyTaskPanel Consulting. (2024, 12 de marzo). Qué es NestJS y cuándo usarlo: características, ventajas y desventajas. <https://www.mytaskpanel.com/que-es-nestjs-y-cuando-usarlo/>

PostgreSQL Global Development Group. (2024). PostgreSQL: The world's most advanced open source database. <https://www.postgresql.org/>

Pressman, R. S., & Maxim, B. R. (2014). *Ingeniería del software: Un enfoque práctico* (8.ª ed.). McGraw-Hill.

Sierra del Guadarrama. (2024, 2 de diciembre). La elaboración de la cerveza artesanal: Un proceso tradicional y meticuloso. <https://sierradelguadarrama.com/la-elaboracion-de-la-cerveza-artesanal-un-proceso-tradicional-y-meticuloso/>

Singh, A. (2025, 25 de enero). React and NestJS: Building a full-stack application. Medium. <https://medium.com/@chauhananubhav16/react-and-nestjs-building-a-full-stack-application-ffea6098769e>

Sommerville, I. (2011). Ingeniería del software (9.<sup>a</sup> ed.). Pearson Educación.

Studocu. (2024, 23 de octubre). Especificación de requisitos según IEEE 830 y 29148. <https://www.studocu.com/co/document/sena-sofiaplus/analisis-y-desarrollo-de-software/analisis-y-especificacion-de-requisitos/108180805>

Tulip Interfaces. (2025, 18 de diciembre). Trazabilidad alimentaria: Buenas prácticas para los fabricantes de alimentos y bebidas. <https://tulip.co/es/blog/what-is-food-traceability/>

Wagner, A. A. (2025, 23 de enero). Arquitectura limpia aplicada en NestJS — Parte I. Medium. <https://medium.com/@arielwagnermovil/arquitectura-limpia-aplicada-en-nestjs-parte-i-d3ffc67f5219>