| DUOC UC - Escuela de informática y telecomunicaciones |
| --- |
| Propuesta de Proyecto y Especificación de Requisitos de Software |
| *Proyecto: [ProyectoAPT]* |
|  |
| **Revisión*: [01]*** |
| **[04/09/24]** |

| Planificación y Especificación de Requisitos según estándares; IEEE 830, ISO9000 y PMI. |
| --- |

**Contenido**

[*DUOC UC - Escuela de informática y telecomunicaciones 1*](#_heading=h.gjdgxs)

[**Ficha del documento 4**](#_heading=h.1fob9te)

[**1. Introducción 5**](#_heading=h.3znysh7)

[1.1.](#_heading=h.2et92p0) Propósito 5

[1.2.](#_heading=h.tyjcwt) Ámbito del Sistema 5

[1.3.](#_heading=h.3dy6vkm) Definiciones, Acrónimos y Abreviaturas 5

[1.4.](#_heading=h.1t3h5sf) Referencias 5

[1.5.](#_heading=h.4d34og8) Visión General del Documento 5

[**2.**](#_heading=h.2s8eyo1) **Descripción General 6**

[2.1.](#_heading=h.17dp8vu) Perspectiva del Producto 6

[2.2.](#_heading=h.3rdcrjn) Funciones del Producto 6

[2.3.](#_heading=h.26in1rg) Características de los Usuarios 6

[2.4.](#_heading=h.lnxbz9) Restricciones 6

[2.5.](#_heading=h.35nkun2) Suposiciones y Dependencias 7

[2.6.](#_heading=h.1ksv4uv) Requisitos Futuros 7

[**3.**](#_heading=h.44sinio) **Requisitos Específicos 8**

[3.1](#_heading=h.2jxsxqh) Requisitos comunes de las interfaces 8

[*3.1.1*](#_heading=h.z337ya) *Interfaces de usuario 8*

[*3.1.2*](#_heading=h.3j2qqm3) *Interfaces de hardware 8*

[*3.1.3*](#_heading=h.1y810tw) *Interfaces de software 8*

[3.2](#_heading=h.4i7ojhp) Requisitos funcionales 9

[3.3](#_heading=h.2xcytpi) Requisitos no funcionales (Organización) y de calidad (Producto) 9

[*3.3.1*](#_heading=h.1ci93xb) *Requisitos de Rendimiento o Eficiencia 9*

[*3.3.2*](#_heading=h.3whwml4) *Requisitos de Seguridad 9*

[*3.3.3*](#_heading=h.2bn6wsx) *Requisitos de Usabilidad 9*

[*3.3.4*](#_heading=h.qsh70q) *Requisitos de Disponibilidad 9*

[*3.3.5*](#_heading=h.3as4poj) *Requisitos de Portabilidad 9*

[*3.3.6*](#_heading=h.1pxezwc) *Requisitos de Mantenibilidad 9*

[*3.3.7*](#_heading=h.49x2ik5) *Requisitos de Funcionalidad 9*

[3.4](#_heading=h.2p2csry) Requisitos No funcionales Organizacionales 10

[**4. Propuesta de Planificación 11**](#_heading=h.147n2zr)

[4.1 Descripción general acerca de la Planificación 11](#_heading=h.3o7alnk)

[*4.1.2 Definición del Equipo de Trabajo 11*](#_heading=h.23ckvvd)

[*4.1.3 Definición de Actividades principales del Proyecto 11*](#_heading=h.ihv636)

[*4.1.4 Resumen Costos del Desarrollo del Proyecto 11*](#_heading=h.32hioqz)

[*4.1.6 Carta Gantt 11*](#_heading=h.1hmsyys)

[5. Anexos 12](#_heading=h.41mghml)

[*5.1 Acta de Proyecto 12*](#_heading=h.2grqrue)

[*5.2 Matriz Especificación de Requerimientos 12*](#_heading=h.vx1227)

[*5.3 Prototipado de Software 12*](#_heading=h.3fwokq0)

[*5.4 Matriz EDT. Planilla Detallada Cálculo de Esfuerzo 12*](#_heading=h.1v1yuxt)

[*5.5 Planilla Carta Gantt 12*](#_heading=h.4f1mdlm)

# Ficha del documento

| **Fecha** | **Revisión** | **Autor** | **Modificación** |
| --- | --- | --- | --- |
| *04/09/24* | *1* | *Pablo Toro*  *Franco ventura*  *Ignacio alvear* |  |
|  |  |  |  |

Documento validado por las partes en fecha:

**Integrantes:**

| **Nombre Integrante del Equipo** | **Rol Definido** |
| --- | --- |
| *Pablo Toro* | *Lider de Proyecto* |
| *Franco Ventura* | *DBA* |
| *Ignacio Alvear* | *Programador* |
|  |  |
|  |  |

# 1. Introducción

En esta sección se proporcionará una introducción a todo el documento de Especificación de Requisitos Software (ERS). Consta de varias subsecciones: propósito, ámbito del sistema, definiciones, referencias y visión general del documento.

## 1.1. Propósito

El presente documento de Especificación de Requisitos de Software (ERS) tiene como objetivo definir de manera detallada los requisitos del sistema de gestión de inventario para la bodega de aires acondicionados. Este documento servirá como guía fundamental para los equipos de desarrollo, pruebas y cualquier otro interesado en el proyecto, asegurando que se comprendan y se acuerden los requisitos antes de iniciar la fase de desarrollo. Además, el ERS actuará como un marco de referencia para validar que el producto final cumpla con las expectativas y necesidades del cliente

## 1.2. Ámbito del Sistema

1.2.2 Descripción del Sistema

Airescold permitirá a los usuarios realizar una gestión eficiente del inventario de productos de aire acondicionado, incluyendo las siguientes funcionalidades:

Gestión de Inventario: Los usuarios podrán agregar, actualizar, consultar y eliminar productos en el inventario, con información detallada sobre el stock, precios, y características técnicas.

Módulo de Ventas: El sistema integrará una tienda en línea donde los clientes podrán visualizar los productos disponibles y realizar compras directamente desde la web.

Integración con Servicios Web: Airescold permitirá la integración con otros sistemas mediante servicios web SOAP, facilitando la comunicación y el intercambio de datos con plataformas externas.

Gestión de Usuarios: El sistema incluirá roles y permisos para diferentes tipos de usuarios, como administradores y clientes, asegurando que solo los usuarios autorizados puedan acceder a determinadas funciones.

Lo que Airescold no hará:

Mantenimiento Físico de Productos: Airescold no gestionará el mantenimiento físico o las reparaciones de los productos, sino que se centrará únicamente en la gestión digital del inventario.

Logística de Entregas: El sistema no gestionará el proceso de envío o logística de los productos vendidos. Esto deberá ser manejado por sistemas externos o manualmente por el personal de la empresa.

1.2.3 Beneficios, Objetivos y Metas

Beneficios:

Eficiencia Operativa: Airescold optimizará el proceso de gestión de inventario, reduciendo errores manuales y tiempos de operación, lo que resultará en una mayor eficiencia en las operaciones diarias.

Aumento de Ventas: Con el módulo de ventas en línea, se espera un incremento en las ventas, ya que los clientes podrán realizar compras de manera rápida y sencilla desde cualquier lugar.

Mejora en la Toma de Decisiones: La disponibilidad de datos en tiempo real permitirá a la empresa tomar decisiones informadas sobre la gestión del inventario y las estrategias de ventas.

Objetivos:

Implementar un sistema centralizado de gestión de inventario que permita la administración eficiente de los productos de aire acondicionado.

Facilitar la venta en línea de productos mediante una plataforma web accesible y fácil de usar.

Integrar servicios web SOAP para permitir la comunicación con otros sistemas y mejorar la interoperabilidad.

Metas:

Reducir en un 50% el tiempo de gestión de inventario dentro de los primeros seis meses de implementación.

Incrementar las ventas en línea en un 30% durante el primer año de operación del módulo de ventas.

Garantizar la interoperabilidad con al menos tres sistemas externos mediante la integración exitosa de servicios web SOAP.

## 1.3. Definiciones, Acrónimos y Abreviaturas

1.3.1 Definiciones

Inventario: Conjunto de bienes o productos almacenados en una bodega, disponibles para la venta o uso.

CRUD: Conjunto de operaciones básicas en bases de datos que permite Crear, Leer, Actualizar y Eliminar registros en una base de datos.

Web Service: Servicio disponible en la web que permite la comunicación y el intercambio de datos entre diferentes sistemas a través de internet.

1.3.2 Acrónimos

API: Application Programming Interface (Interfaz de Programación de Aplicaciones). Permite la interacción entre diferentes sistemas de software a través de definiciones y protocolos establecidos.

SOAP: Simple Object Access Protocol (Protocolo de Acceso a Objetos Simple). Un protocolo estándar para el intercambio de información en servicios web mediante mensajes XML.

ERS: Especificación de Requisitos de Software. Documento que define los requisitos funcionales y no funcionales de un sistema de software.

1.3.3 Abreviaturas

DB: Base de Datos (Database). Sistema organizado para almacenar, gestionar y recuperar datos de manera eficiente.

UI: Interfaz de Usuario (User Interface). Parte del software que interactúa con el usuario, permitiendo la visualización y manipulación de la información.

HTTP: Hypertext Transfer Protocol (Protocolo de Transferencia de Hipertexto). Protocolo utilizado en la web para la transferencia de información entre un cliente y un servidor.

## 1.4. Referencias

En esta subsección se mostrará una lista completa de todos los documentos referenciados en la ERS.

## 1.5. Visión General del Documento

Sección 1: Descripción General

Esta sección proporciona una visión completa del contexto en el que se desarrollará el sistema. Se abordan temas como la perspectiva del producto, el alcance del sistema, las funciones del usuario, las restricciones del diseño y los supuestos y dependencias relacionados con el proyecto.

Sección 2: Requisitos Funcionales

Esta sección detalla los requisitos funcionales del sistema, describiendo cada una de las funcionalidades que debe cumplir. Aquí se definen las interacciones entre el usuario y el sistema, así como las reglas de negocio que rigen su comportamiento.

Sección 3: Requisitos No Funcionales

En esta sección se especifican los requisitos no funcionales, tales como rendimiento, seguridad, usabilidad, y otras características que afectan la calidad y el entorno operativo del sistema. Estos requisitos aseguran que el sistema funcione de manera eficiente y segura bajo diversas condiciones.

Sección 4: Modelos y Diagramas

Aquí se incluyen los modelos y diagramas que representan la arquitectura del sistema, su diseño, y la interacción entre sus componentes. Los diagramas de clases, de casos de uso, y de secuencia, entre otros, proporcionan una representación visual de cómo está estructurado el sistema y cómo operan sus componentes.

Sección 5: Requisitos de Interfaz de Usuario

Esta sección describe las interfaces de usuario que se desarrollarán como parte del sistema. Se especifican los requisitos de diseño de la interfaz, la disposición de las pantallas, y la interacción con el usuario.

Sección 6: Requisitos de Interfaz Externa

En esta sección se describen los requisitos para las interfaces con otros sistemas o componentes externos, incluyendo la integración con servicios web SOAP, bases de datos, y otras aplicaciones externas necesarias para el funcionamiento del sistema.

Sección 7: Aceptación y Validación

Esta sección detalla los criterios y métodos de aceptación que se utilizarán para verificar que el sistema cumple con todos los requisitos especificados. Incluye los planes de pruebas y validación necesarios para garantizar la calidad del software.

Sección 8: Anexos

Los anexos incluyen cualquier información adicional que sea relevante para la comprensión del documento o del proyecto en general, como glosarios, diagramas adicionales, y referencias a documentación técnica complementaria

# 2. Descripción General

El sistema de eCommerce para la venta de aire acondicionado es una plataforma en línea diseñada para permitir a los usuarios explorar, comparar y comprar diferentes unidades de aire acondicionado. El sistema tiene como objetivo principal facilitar una experiencia de compra intuitiva y eficiente para los clientes, mientras proporciona herramientas robustas para la gestión de productos y pedidos para los administradores del sitio. Este sistema debe integrarse con sistemas de pago en línea, gestionar el inventario en tiempo real y ofrecer soporte post-venta.

funciones:

**Catálogo de Productos:** Visualización con descripciones, especificaciones, imágenes y precios.

**Carrito de Compras:** Añadir, revisar y ajustar productos antes de pagar.

**Proceso de Pago:** Integración con sistemas de pago para transacciones seguras.

**Gestión de Pedidos:** Gestión de pedidos y actualización de estados.

**Gestión de Inventario:** Actualización y monitoreo del stock.

## 2.1. Perspectiva del Producto

El sistema de eCommerce para la venta de aire acondicionado es una solución independiente que proporciona una plataforma completa para la compra en línea de unidades de aire acondicionado. Sin embargo, se integra y se interrelaciona con varios sistemas y productos para funcionar de manera efectiva. A continuación, se detallan las relaciones e interfaces con otros sistemas y productos relevantes:

El sistema de eCommerce incluirá las siguientes funciones clave:

**Sistemas de Procesamiento de Pagos:**

* **Descripción:** El eCommerce se integra con plataformas de pago en línea (mercado pago) para procesar transacciones de manera segura.
* **Funcionalidad:** Permite a los usuarios pagar sus compras utilizando diversas opciones de pago y asegura que las transacciones se manejen de manera segura y conforme a las normativas.

## 2.2. Funciones del Producto

**Visualización de Productos**: Los usuarios podrán ver una lista de productos disponibles, con información detallada como descripción, características técnicas, imágenes, y precios.

**Filtrado y Búsqueda**: Los usuarios podrán buscar productos utilizando filtros como marca, tipo de aire acondicionado (split, portátil, etc.), capacidad (BTU), y rango de precios.

**Añadir al Carrito**: Los usuarios podrán añadir productos a un carrito de compras virtual con opciones para ajustar cantidades y eliminar artículos.

**Revisión del Carrito**: Los usuarios podrán revisar los productos en el carrito, ver el subtotal, aplicar cupones de descuento y calcular los costos de envío.

**Proceso de Pago**: Integración con sistemas de pago para procesar pagos seguros mediante tarjetas de crédito/débito, PayPal u otros métodos de pago en línea.

**Confirmación de Pedido**: Generación de una confirmación de pedido que incluya un resumen de la compra, detalles de envío y número de seguimiento.

## 2.3. Características de los Usuarios

#### Clientes

* **Nivel Educacional**: Varía desde secundaria hasta educación superior. Los clientes pueden incluir desde estudiantes hasta profesionales y ejecutivos.
* **Experiencia Técnica**: Generalmente, los clientes tienen un nivel básico a intermedio de habilidades técnicas. Pueden tener experiencia con el uso de eCommerce, pero no necesariamente con plataformas específicas de venta de aire acondicionado.
* **Experiencia con el Producto**: Los clientes pueden variar en su conocimiento sobre aire acondicionado, desde principiantes que buscan su primera unidad hasta expertos que conocen especificaciones técnicas detalladas.
* **Perfil**:
  + **Objetivo**: Comprar aire acondicionado para uso personal o profesional.
  + **Necesidades**: Facilidad de navegación, información clara sobre productos, opciones de comparación, y un proceso de compra simple y seguro.
  + **Intereses**: Comparar precios, revisar especificaciones, leer reseñas y encontrar las mejores ofertas.

## 2.4. Restricciones

Esta subsección describe aquellas limitaciones que se imponen sobre los desarrolladores del producto:

• Políticas de la empresa.

**Políticas de Privacidad y Protección de Datos:** El sistema debe cumplir con las políticas de privacidad y protección de datos de la empresa, así como con las leyes y regulaciones locales e internacionales (como GDPR o CCPA).

**Políticas de Seguridad:** Debe adherirse a las políticas internas de seguridad para proteger datos sensibles y garantizar la integridad del sistema.

• Limitaciones del hardware.

El sistema debe ser accesible para una gran variedad de dispositivos como notebooks o PC. y deberán cumplir con los requisitos mínimos:

* 4gm ram
* intel Gen 3
* 240 HDD
* Gráficos integrados

• Interfaces con otras aplicaciones.

**Integraciones Externas:** Debe integrar con sistemas externos como plataformas de pago como mercadopago, sistemas de gestión de inventario con soapUI siguiendo las especificaciones de las APIs correspondientes.

• Operaciones paralelas.

**Mantenimiento y Actualizaciones:** Las actualizaciones y el mantenimiento del sistema deben realizarse sin interrumpir las operaciones normales del sitio eCommerce. Debe implementarse un plan para realizar estos cambios de manera eficiente.

**Cargas Concurrentes:** El sistema debe manejar múltiples transacciones y consultas simultáneas sin degradar el rendimiento, especialmente durante picos de tráfico.

• Funciones de auditoría.

**Registro de Actividades:** El sistema debe incluir capacidades de auditoría para registrar actividades críticas como transacciones, accesos y cambios en la configuración del sistema.

**Accesos y Permisos:** Debe tener mecanismos para auditar accesos y cambios en el sistema, asegurando que solo personal autorizado pueda realizar modificaciones importantes.

• Funciones de control.

**Control de Acceso:** Implementar controles de acceso basados en roles para garantizar que los usuarios solo puedan acceder a las funciones y datos necesarios para sus responsabilidades.

**Gestión de Configuración:** Las configuraciones del sistema deben ser controladas para prevenir cambios no autorizados que puedan afectar la operación del eCommerce.

• Lenguaje(s) de programación.

**Lenguajes Permitidos:** El desarrollo del sistema debe utilizar lenguajes de programación aprobados por la empresa, como HTML, CSS, JavaScript (para el frontend), y Python, Django y Java (para el backend), según las políticas tecnológicas establecidas.

• Protocolos de comunicación.

**Protocolos de Seguridad:** Utilizar protocolos de comunicación seguros, como HTTPS, para proteger la transmisión de datos entre el cliente y el servidor.

**Intercambio de Datos:** Asegurar que las interfaces con sistemas externos utilicen protocolos y formatos de datos estandarizados (como RESTful APIs y JSON).

• Requisitos de habilidad.

**Habilidades Técnicas del Equipo:** El equipo de desarrollo debe contar con habilidades en las tecnologías y herramientas seleccionadas para el proyecto, como desarrollo web, bases de datos y análisis de negocio

• Criticidad de la aplicación.

**Disponibilidad:** El sistema debe estar disponible y operable 24/7, con una alta disponibilidad y un tiempo mínimo de inactividad para cumplir con las expectativas de los usuarios.

**Tiempo de Respuesta:** Debe cumplir con los requisitos de rendimiento establecidos, proporcionando tiempos de respuesta rápidos para transacciones y búsquedas.

• Consideraciones acerca de la seguridad.

**Autenticación y Autorización:** Asegurar que los mecanismos de autenticación y autorización sean robustos para prevenir accesos no autorizados. Estas son las contraseñas seguras y la edición de las credenciales.

## 2.5. Suposiciones y Dependencias

**Suposiciones**

**Recursos Tecnológicos:** Se asume que el sistema funcionará en servidores que cumplen con las especificaciones mínimas.

**Conectividad a Internet:** Los usuarios tienen acceso a Internet de alta velocidad.

**Cumplimiento Normativo:** El sistema cumplirá con leyes de protección de datos y comercio electrónico vigentes.

**Compatibilidad de Navegadores:** El sistema es compatible con navegadores web modernos (google, edge, safari, etc.).

**Interoperabilidad con Sistemas Externos:** Las integraciones con servicios externos mercadopago y soapUI) funcionarán según los estándares establecidos.

**Dependencias**

**Actualizaciones Tecnológicas:** Dependencia de tecnologías y bibliotecas de terceros que deben mantenerse actualizadas.

**Capacitación del Personal:** El personal debe estar capacitado para usar y mantener el sistema.

**Sistema Operativo:** Depende de un sistema operativo compatible (Linux o Windows Server) y versiones específicas de software.

## 2.6. Requisitos Futuros

### Funcionalidades

* **Chatbots:** Integrar chatbots para mejorar la atención al cliente en tiempo real.

### Análisis y Reportes Avanzados

* **Informes Personalizados:** Crear informes detallados para ventas e inventario.
* **Dashboards Interactivos:** Implementar dashboards para visualizar datos clave.

# 3. Requisitos Específicos

* Todos los sistemas con el que el usuario interactúa son a través de la página web , por medio del monitor, teclado y mouse.
* Control de las funciones manipulables del equipo.
* Sistema de ayuda interactivo.
* Login y registro de usuarios: El sistema deberá tener un Login (Inicio de sesión) y un registro de usuarios para cuántas personas usarán el sistema y además ayudará a futuras funcionalidades.
* Perfil de usuario: En esta sección se podrá encontrar la información básica del usuario.
* Perfil de pagos: Aquí el usuario podrá asociar sus métodos de pago con mercado pago y/o asociar tarjetas de crédito o débito por el mismo
* Control de las funciones manipulables del equipo.
* Sistema de ayuda interactivo.

## 3.1 Requisitos comunes de las interfaces

Descripción detallada de todas las entradas y salidas del sistema de software.

### 3.1.1 Interfaces de usuario

* Las pantallas son de color café moro
* Letra tamaña 10
* Los botones deben de color calido
* **Página de Inicio:** Incluye navegación principal y destacados productos.
* **Página de Producto:** Muestra detalles del producto, imágenes y opciones de compra.
* **Carrito de Compras:** Muestra los productos añadidos, opciones para modificar cantidades y proceder al pago.
* **Página de Pago:** Permite ingresar información de pago y revisar el pedido antes de finalizar.

### 3.1.2 Interfaces de hardware

**Servidor Web:** El sistema debe ser compatible con servidores que ejecutan Linux o Windows Server con al menos 4 GB de RAM y 4 núcleos de CPU.

**Dispositivos del Usuario:** El sistema debe funcionar en computadoras de escritorio y notebooks navegadores web modernos.

### 3.1.3 Interfaces de software

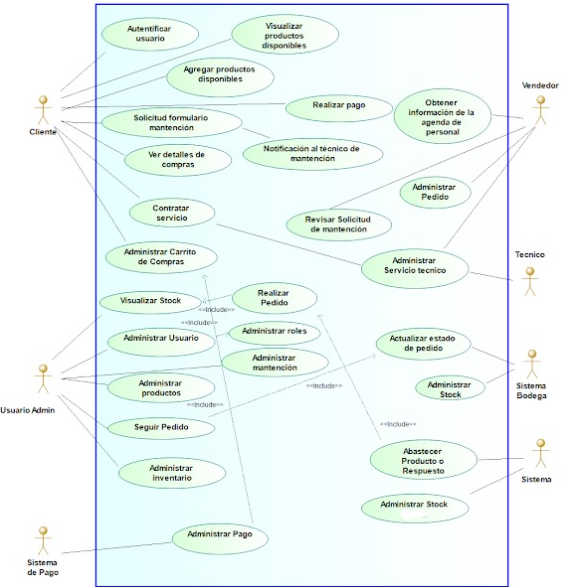
**Integraciones con Otros Productos de Software:**

* **Sistema de Pago:**
  + **Descripción:** Plataforma de pagos en línea (mercadopago).
  + **Propósito del Interfaz:** Permitir a los usuarios realizar pagos de manera segura.
  + **Definición del Interfaz:**
    - **Contenido:** Información del pedido y detalles del pago.
    - **Formato:** API RESTful con respuestas en JSON para transacciones y verificación.

## 3.2 Requisitos funcionales

| **[R-N°]** | **[Nombre del Requerimiento]** | **Tipo Requerimiento**  **[Funcional, No Funcional]** | **Actores Usuarios Relacionados** | **[Descripción corta del requerimiento]** | **Estado** |
| --- | --- | --- | --- | --- | --- |
| R.1 | Administración de roles y usuarios del sistema | Funcional | Usuario Admin | El sistema permitirá al Usuario Admin crear, editar permisos, editar roles, de Usuario Vendedor, Usuario Tecnicos, Usuarios Servicio Técnico, Usuario Logística del sistema | por hacer |
| R.2 | Registro usuarios "Comprador" desde la página web | Funcional | Usuario Cliente/ Usuario Vendedor | El sistema debe permitir el registro de usuarios para compra de productos y/o contratación de servicios ingresando un correo electrónico, datos personales asociados al comprador y contraseñas personalizadas para la autenticación de usuario cliente. | por hacer |
| R.3 | Búsqueda de productos o servicios para contratación | Funcional | Usuario Cliente/ Usuario Vendedor | Permite al usuario visualizar y buscar los diferentes productos (aires acondicionados) y servicios relacionados a contratar para el usuario cliente. | por hacer |
| R.4 | Selección de productos o servicios para contratación | Funcional | Usuario Cliente/ Usuario Vendedor | Habilita al usuario para seleccionar dentro del stock de productos y servicios expuestos la selección de estos para solicitud de compra o contratación. | por hacer |
| R.5 | Pago de productos y/o servicios para contratación | Funcional | Usuario Cliente/ Usuario Vendedor | Permite al usuario cliente realizar los pagos asociados a sus compras de productos o solicitud de servicios contratados. ya sea de manera web o física | por hacer |
| R.6 | Monitoreo de contratación de servicios | Funcional | Usuario Cliente/ Usuario Vendedor | El sistema permitirá al usuario el estado del proceso seleccionado tanto para la compra de productos como para la contratación de servicios contratados por el cliente de manera que pueda saber en que estadio se encuentra. | por hacer |
| R.7 | Visualización de records sobre contrataciones y compras asociadas al cliente | Funcional | Usuario Cliente / Usuario Vendedor | El sistema entregará un listado histórico de los movimientos realizados por el cliente, ya sea compras realizadas en el sistema como contrataciones de servicios asociadas. | por hacer |
| R.8 | monitoreo de pedidos de productos | Funcional | Usuario Cliente/ Usuario Servicio Al Cliente/ Usuario Logistica | Se despliega una vista que entrega el estado y ruta del pedido de productos realizado | por hacer |
| R.9 | Seguimiento de servicios contratados por cliente | Funcional | Usuario Admin/ Usuario Tecnico/ Usuario Servicio Tecnico/ Usuario Vendedor | Permitirá visualizar solicitudes de contratación de servicios detalladamente, nombre de cliente, servicio, equipamiento a instalar o mantener, fecha de contratación, fecha de instalación, estado de la solicitud | por hacer |
| R.10 | Visualización de solicitudes de contratación de servicio | Funcional | Usuario Admin/ Usuario Tecnico/ Usuario Servicio Al Cliente | EL sistema permitirá a los usuarios asociados aprobar solicitudes de contratación de servicios solicitadas tanto por Atención al cliente, vendedores y clientes | por hacer |
| R.11 | Aprobación de solicitud de contratación de servicios | Funcional | Usuario Admin/ Usuario Tecnico/ | El sistema permitirá a los usuarios asociados aprobar solicitudes de contratación de servicios solicitadas tanto por Atención al cliente, vendedores y clientes | por hacer |
| R.12 | Administración de Servicios | Funcional | Usuario Admin | El sistema permitirá crear, modificar y eliminar servicios que se entregarán por parte de los Técnicos registrados en el sistema detallando sus características, valores y elementos relacionados. | por hacer |
| R.13 | Visualización de Servicios | Funcional | Todos los Usuarios | Desplegara vistas de todos los servicios proporcionado para su contratacion, valores y detalles del mismo | por hacer |
| R.14 | Administracion stock de productos | Funcional | Usuario Admin/Usuario Logistica/ | Permite a los usuarios administrar elementos referentes al stock de productos, tanto como agregar producto, modificar producto, modificar cantidad | por hacer |
| R.15 | visualizacion de stock de productos | Funcional | Todos los usuarios | El sistema permite a los usuarios la visualización del stock de los productos | por hacer |
| R.16 | Creación de pedidos de productos | Funcional | Usuario Cliente/ Usuario Vendedor/ Usuario Logistica | Permite la creación de pedidos de productos para adquisición de estos por un cliente | por hacer |
| R.17 | administración del estado de pedidos de productos | Funcional | Usuario Logistica | Permite actualizar y modificar los estados de las etapas del pedido del producto (ingresado, elaborando pedido, en proceso, despachado, recepcionado) | por hacer |
| R.18 | visualización estado pedidos de productos | Funcional | Usuario Cliente/ Usuario Vendedor/ Usuario Logistica | Se visualiza en pantalla el estado de todos los pedidos realizados por el usuario | por hacer |
| R.19 | visualización pedidos de productos | Funcional | Usuario Cliente/Usuario Logistica | Se visualiza el detalle del pedido realizado por el usuario | por hacer |
| R.20 | seguimiento de pedido de productos | Funcional | Usuario Cliente/ Usuario Logistica | Permite monitorear la fecha y ruta del pedido realizado para su próxima entrega | por hacer |
| R.21 | Integración de datos | No Funcional |  | La solución debe ser capaz de integrar los datos de las aplicaciones existentes de manera eficiente y sin pérdida de datos. | por hacer |
| R.22 | Interoperabilidad de tecnologías | No Funcional |  | La solución debe ser compatible con los sistemas y aplicaciones existentes en la empresa. | por hacer |
| R.23 | Seguridad y privacidad de datos sensibles | No Funcional |  | La solución debe garantizar la seguridad de los datos y la privacidad de los usuarios, cumpliendo con los estándares y normas de seguridad de la industria. | por hacer |
| R.24 | Mantenimiento y soporte del aplicativo | No Funcional |  | La solución debe ser fácil de mantener y actualizar, con un mínimo impacto en los sistemas existentes. | por hacer |
| R.25 | Saturación controlada y optimizada de datos | No Funcional |  | La solución debe ser capaz de manejar grandes cantidades de datos y usuarios, sin comprometer el rendimiento. | por hacer |
| R.26 | Interfaz gráfica intuitiva | No Funcional |  | La solución debe ser fácil de usar e intuitiva para los usuarios finales, con una interfaz de usuario clara y sencilla. | por hacer |
| R.27 | Optimización de tiempos de respuesta | No Funcional |  | La solución debe tener un alto rendimiento y una respuesta rápida a las solicitudes del usuario. | por hacer |
| R.28 | Disponibilidad del sistema | No Funcional |  | La solución debe estar disponible las 24 horas del día, los 7 días de la semana, con una interrupción mínima del servicio. | por hacer |
| R.29 | Fiabilidad de funcionalidad y control de errores | No Funcional |  | La solución debe ser confiable y estable, minimizando los errores y las interrupciones en el servicio. | por hacer |
| R.30 | Compatibilidad del aplicativo a diferentes plataformas y tecnologías asociadas | No Funcional |  | La solución debe ser compatible con una amplia gama de dispositivos y navegadores, y cumplir con los estándares de la industria. | por hacer |
| R.31 | Adaptabilidad y escalabilidad del aplicativo | No Funcional |  | La solución debe ser escalable y capaz de crecer con la empresa, sin comprometer el rendimiento o la estabilidad | por hacer |
| R.32 | Garantizar la seguridad y protección de datos | No Funcional |  | La solución debe cumplir con los estándares y normas de seguridad de la industria, garantizando la protección de los datos y la privacidad de los usuarios. | por hacer |

Los requisitos funcionales deben estar enumerados (diagrama caso uso) y pueden ser divididos en sub-secciones.



## 3.3 Requisitos no funcionales (Organización) y de calidad (Producto)

### 3.3.1 Requisitos de Rendimiento o Eficiencia

| RNF-15 | Actualización base de datos | Los datos modificados en la base de datos deben ser actualizados para todos los usuarios que acceden en menos de 2 segundos. | Que no se demore más de 2 segundos en actualizar los datos | Eficiencia |
| --- | --- | --- | --- | --- |

### 3.3.2 Requisitos de Seguridad

| R.32 | Garantizar la seguridad y proteccion de datos | No Funcional |  | La solución debe cumplir con los estándares y normas de seguridad de la industria, garantizando la protección de los datos y la privacidad de los usuarios. | por hacer |
| --- | --- | --- | --- | --- | --- |

| RNF-04 | Servidor de base de datos | El servidor debe considerar la utilización de certificado de sitio seguro. | El servidor se encuentra implementado con certificado SSL de 256 bit.  https://www.hosting.cl/certificados-ssl | **Seguridad**  (Métrica: Integridad) |
| --- | --- | --- | --- | --- |

### 3.3.3 Requisitos de Usabilidad

| RNF-13 | Accesibilidad | Opción de pantalla para personas con daltonismo | escoger opciones>formato daltonico. | usabilidad |
| --- | --- | --- | --- | --- |

### 3.3.4 Requisitos de Disponibilidad

| R.28 | Disponibilidad del sistema | No Funcional |  | La solución debe estar disponible las 24 horas del día, los 7 días de la semana, con una interrupción mínima del servicio. | por hacer |
| --- | --- | --- | --- | --- | --- |

### 3.3.5 Requisitos de Portabilidad

| R.22 | Interoperabilidad de tecnologías | No Funcional |  | La solución debe ser compatible con los sistemas y aplicaciones existentes en la empresa. | por hacer |
| --- | --- | --- | --- | --- | --- |

La portabilidad se refiere a la capacidad del sistema AireCold para ser transferido y operado en diferentes entornos de hardware, software o plataformas sin necesidad de modificaciones extensivas. A continuación se detallan los requisitos de portabilidad necesarios para asegurar que el sistema pueda ser fácilmente adaptado y utilizado en distintos contextos:

Compatibilidad con Diversos Entornos de Ejecución:

Requisitos: El sistema debe ser capaz de operar en múltiples plataformas y entornos operativos sin necesidad de modificaciones significativas. Esto incluye:

Compatibilidad Multiplataforma: El sistema debe ser compatible con diferentes sistemas operativos (por ejemplo, Windows, Linux, macOS) y versiones relevantes.

Dependencias Mínimas: Minimizar las dependencias en tecnologías específicas de una plataforma, utilizando estándares y tecnologías que sean ampliamente soportados.

Portabilidad del Código Fuente:

Requisitos: El código del sistema debe estar escrito de manera que facilite la portabilidad a diferentes entornos de desarrollo y ejecución. Esto implica:

Uso de Estándares Abiertos: Utilizar lenguajes de programación, bibliotecas y frameworks que sean estándar y compatibles con múltiples plataformas.

Configuración Externa: Configurar parámetros específicos de la plataforma (como rutas de archivo, configuraciones de red) mediante archivos de configuración externos en lugar de hardcoding.

Portabilidad de Datos:

Requisitos: Los datos gestionados por el sistema deben ser fácilmente exportables e importables entre diferentes entornos y sistemas. Esto incluye:

Formatos de Datos Estándar: Utilizar formatos de datos abiertos y estandarizados (como CSV, JSON, XML) que sean compatibles con diferentes herramientas y sistemas.

Migración de Datos: Proporcionar herramientas o scripts que faciliten la migración de datos entre diferentes bases de datos o sistemas de almacenamiento.

Adaptación a Nuevas Versiones de Software:

Requisitos: El sistema debe poder adaptarse a nuevas versiones de software sin necesidad de grandes modificaciones. Esto incluye:

Actualizaciones de Dependencias: Asegurarse de que las bibliotecas y dependencias utilizadas estén actualizadas y sean compatibles con las versiones más recientes de los sistemas operativos y plataformas.

Pruebas de Compatibilidad: Realizar pruebas de compatibilidad periódicas con nuevas versiones de software y hardware para identificar y resolver problemas de portabilidad.

Interoperabilidad con Otros Sistemas:

Requisitos: El sistema debe ser capaz de integrarse y comunicarse con otros sistemas y plataformas de manera eficiente. Esto implica:

Interfaces y APIs Estándar: Implementar interfaces y APIs que utilicen estándares abiertos y protocolos comunes (como RESTful APIs) para facilitar la interoperabilidad.

Documentación de Integración: Proporcionar documentación clara y detallada sobre cómo integrar el sistema con otros servicios y plataformas.

Soporte para Entornos de Virtualización:

Requisitos: El sistema debe ser compatible con entornos de virtualización y contenedores para facilitar su despliegue en diferentes entornos de nube y locales. Esto incluye:

Contenedores y Virtualización: Proporcionar soporte para la ejecución en contenedores (por ejemplo, Docker) y entornos virtualizados, asegurando que el sistema funcione correctamente en estos contextos.

Scripts de Despliegue: Incluir scripts y configuraciones que faciliten el despliegue en entornos de virtualización y nubes.

### 3.3.6 Requisitos de Mantenibilidad

| R.24 | Mantenimiento y soporte del aplicativo | No Funcional | usuario cliente | La solución debe ser fácil de mantener y actualizar, con un mínimo impacto en los sistemas existentes. | por hacer |
| --- | --- | --- | --- | --- | --- |
| R.25 | Saturación controlada y optimizada de datos | No Funcional | usuario cliente | La solución debe ser capaz de manejar grandes cantidades de datos y usuarios, sin comprometer el rendimiento. | por hacer |

La mantenibilidad se refiere a la facilidad con la que el sistema AireCold puede ser modificado para corregir errores, adaptarse a cambios en el entorno, mejorar el rendimiento o agregar nuevas funcionalidades. A continuación se describen los requisitos de mantenibilidad necesarios para el sistema, especificando los tipos de mantenimiento que serán requeridos:

Mantenibilidad Correctiva:

Requisitos: El sistema debe ser diseñado para facilitar la identificación y corrección de errores. Esto implica:

Registro de Errores: Implementar un sistema de logging y seguimiento de errores para detectar, registrar y priorizar problemas.

Documentación de Código: El código debe estar bien documentado, facilitando la comprensión y modificación por parte de otros desarrolladores.

Pruebas Unitarias: Deben estar en su lugar pruebas unitarias y de integración que ayuden a identificar problemas rápidamente durante el proceso de corrección.

Mantenibilidad Adaptativa:

Requisitos: El sistema debe ser flexible para adaptarse a cambios en el entorno tecnológico o en las normativas externas. Esto incluye:

Arquitectura Modular: La arquitectura del sistema debe ser modular para permitir modificaciones sin afectar al resto del sistema.

Interfaz de Configuración: Las configuraciones y parámetros que pueden cambiar con el tiempo deben ser fácilmente ajustables sin necesidad de modificar el código fuente.

Compatibilidad con versiones: El sistema debe ser compatible con nuevas versiones de software y hardware, con actualizaciones documentadas y planificadas.

Mantenibilidad Perfectiva:

Requisitos: El sistema debe permitir la incorporación de mejoras y nuevas funcionalidades con facilidad. Esto se logra mediante:

Extensibilidad del Código: El código debe estar escrito de manera que facilite la extensión y adición de nuevas funcionalidades sin requerir reescrituras extensivas.

Modularidad de Componentes: Los componentes del sistema deben ser independientes y reemplazables, lo que permite la adición o mejora de funcionalidades sin impactar el sistema en su totalidad.

Gestión de Versiones: El uso de un sistema de control de versiones debe estar integrado en el flujo de trabajo para gestionar cambios y versiones del software eficientemente.

Mantenibilidad Preventiva:

Requisitos: Se deben implementar prácticas que minimicen la necesidad de mantenimiento reactivo. Esto incluye:

Actualizaciones Regulares: Planificar y aplicar actualizaciones periódicas para mantener el sistema al día con los últimos parches de seguridad y mejoras.

Revisión de Código: Realizar revisiones de código regulares para identificar y resolver problemas potenciales antes de que se conviertan en fallos críticos.

Documentación y Normas de Codificación: Seguir normas de codificación y mantener una documentación clara y actualizada para facilitar futuras modificaciones y mantenimiento.

Mantenibilidad Documental:

Requisitos: La documentación del sistema debe ser completa y accesible, facilitando el mantenimiento y la formación. Esto implica:

Manual de Usuario y Administrador: Documentar claramente cómo usar y administrar el sistema, incluyendo guías para la resolución de problemas comunes.

Documentación Técnica: Proporcionar documentación técnica detallada que describa la arquitectura, el diseño y el código del sistema.

Registro de Cambios: Mantener un registro detallado de todos los cambios realizados en el sistema, incluyendo mejoras, correcciones y actualizaciones.

### 3.3.7 Requisitos de Funcionalidad

Para asegurar la correcta operación y longevidad del sistema AireCold, se deben identificar y definir los tipos de mantenimiento necesarios. Estos requisitos de funcionalidad están orientados a garantizar que el sistema se mantenga en condiciones óptimas, adaptándose a cambios y evolucionando según las necesidades del negocio. A continuación se describen los tipos de mantenimiento que el sistema requerirá:

Mantenimiento Correctivo:

Descripción: Este tipo de mantenimiento se enfoca en la corrección de errores o defectos encontrados en el sistema después de su implementación. Los errores pueden incluir fallos en la funcionalidad, problemas de rendimiento, o cualquier otro problema que afecte la operatividad del sistema.

Procedimiento: El equipo de soporte técnico debe registrar y priorizar los errores reportados. Los errores críticos deben ser corregidos de inmediato, mientras que los menos urgentes pueden ser abordados en ciclos de mantenimiento programados. Las correcciones deben ser probadas exhaustivamente antes de ser implementadas en el entorno de producción.

Mantenimiento Adaptativo:

Descripción: Este mantenimiento se realiza para adaptar el sistema a cambios en el entorno externo, como actualizaciones en el software, cambios en la normativa legal, o modificaciones en la infraestructura tecnológica.

Procedimiento: Los cambios en el entorno externo deben ser monitoreados regularmente para identificar cualquier impacto potencial en el sistema. El equipo de desarrollo debe realizar las modificaciones necesarias y probar las adaptaciones para asegurar que el sistema siga funcionando correctamente con las nuevas condiciones.

Mantenimiento Perfectivo:

Descripción: Este tipo de mantenimiento busca mejorar el rendimiento del sistema o añadir nuevas funcionalidades basadas en las necesidades cambiantes de los usuarios o en nuevas oportunidades de negocio.

Procedimiento: Las solicitudes de mejoras y nuevas funcionalidades deben ser evaluadas y priorizadas por el equipo de gestión de proyectos. Las mejoras deben ser diseñadas, desarrolladas e implementadas de manera controlada, con pruebas rigurosas para asegurar que no se introduzcan nuevos problemas en el sistema.

Mantenimiento Preventivo:

Descripción: El mantenimiento preventivo se realiza para evitar posibles fallos del sistema y garantizar su estabilidad a largo plazo. Esto incluye actividades como actualizaciones de software, optimización de bases de datos, y revisión de la infraestructura.

Procedimiento: Se deben establecer y seguir calendarios de mantenimiento preventivo para realizar actualizaciones periódicas, realizar pruebas de rendimiento y llevar a cabo auditorías de seguridad. El equipo de mantenimiento debe estar al tanto de las mejores prácticas y las recomendaciones de los proveedores de software y hardware.

Mantenimiento de Documentación:

Descripción: Mantener la documentación del sistema actualizada es crucial para apoyar el mantenimiento y la evolución del sistema. Esto incluye manuales de usuario, documentación técnica y registros de cambios.

Procedimiento: La documentación debe ser revisada y actualizada de manera continua para reflejar cualquier cambio en el sistema. Esto asegura que todos los miembros del equipo y los usuarios finales tengan acceso a información precisa y actualizada.

Mantenimiento de Capacitación:

Descripción: Asegurar que los usuarios y el personal de soporte estén capacitados en el uso del sistema y en los procedimientos de mantenimiento es esencial para el éxito del proyecto.

Procedimiento: Se deben proporcionar sesiones de capacitación regulares y materiales educativos actualizados para el personal de soporte y los usuarios finales. Además, se debe ofrecer soporte continuo para resolver cualquier duda o problema relacionado con el uso del sistema

## 3.4 Requisitos No funcionales Organizacionales

Los siguientes requisitos no funcionales han sido solicitados por la empresa y están orientados a garantizar la alineación del sistema con la identidad corporativa y otras políticas internas, aunque no afectan directamente la calidad del software:

Identidad Corporativa:

El sistema AireCold debe incorporar el logo oficial de la empresa en la pantalla de inicio, así como en todas las páginas principales de la interfaz de usuario. Además, los colores corporativos de la empresa, que son rojo y azul, deben ser utilizados en el diseño de la interfaz gráfica. Estos colores deben estar presentes en los elementos clave de la interfaz, como los encabezados, botones y barras de navegación.

Política de Seguridad Interna:

El sistema debe cumplir con las políticas de seguridad de la información establecidas por la empresa. Esto incluye la gestión de contraseñas conforme a los estándares corporativos, con la obligación de cambiar las contraseñas cada 90 días, y la autenticación de dos factores para los usuarios con permisos administrativos.

Compatibilidad con Herramientas Existentes:

El sistema debe ser compatible con las herramientas de software ya existentes en la empresa, como los sistemas de correo electrónico y el software de gestión de proyectos utilizado por el equipo (ej., Jira, Confluence). Esto permitirá una integración fluida con los procesos y sistemas actualmente en uso.

Política de Accesibilidad:

El diseño del sistema debe seguir las normas de accesibilidad definidas por la empresa, asegurando que el software sea utilizable por personas con discapacidades. Esto incluye la implementación de funciones como textos alternativos para imágenes, compatibilidad con lectores de pantalla y la capacidad de ajustar el contraste y el tamaño del texto en la interfaz de usuario.

Documentación Estándar:

Toda la documentación relacionada con el sistema debe seguir el formato y las normas establecidas por la empresa. Esto incluye la plantilla de documentación técnica, informes de estado del proyecto, y manuales de usuario. La documentación debe estar disponible en los idiomas oficiales de la empresa y ser accesible a través del portal interno de la compañía.

Cumplimiento Legal y Normativo:

El sistema debe cumplir con todas las regulaciones legales y normativas relevantes, tanto locales como internacionales, que la empresa debe seguir. Esto incluye, entre otras, regulaciones sobre protección de datos, privacidad de la información y comercio electrónico.

# 4. Propuesta de Planificación

## 4.1 Descripción general acerca de la Planificación

El desarrollo del sistema AireCold se llevará a cabo en un período estimado de 90 días distribuidos en diferentes fases de trabajo. A continuación, se detallan las fases y las personas involucradas:

Fase 1: Planificación y Análisis de Requisitos (10 días)

Equipo:

Jefe de Proyecto

Analista de Requisitos

Stakeholders

Actividades: Recolección de requisitos, análisis inicial, y creación del documento ERS.

Fase 2: Diseño del Sistema (15 días)

Equipo:

Arquitecto de Software

Diseñador de UI/UX

Desarrolladores

Actividades: Diseño de la arquitectura del sistema, creación de modelos y diagramas, diseño de la interfaz de usuario.

Fase 3: Desarrollo e Integración (40 días)

Equipo:

Desarrolladores

Ingeniero de Integración

Administrador de Bases de Datos

Actividades: Codificación del sistema, desarrollo del CRUD, integración de servicios web SOAP, desarrollo de la tienda en línea.

Fase 4: Pruebas y Validación (15 días)

Equipo:

Ingeniero de Pruebas

QA (Aseguramiento de la Calidad)

Usuarios finales (para pruebas de aceptación)

Actividades: Pruebas unitarias, pruebas de integración, pruebas de aceptación del usuario, validación del sistema.

Fase 5: Despliegue y Mantenimiento (10 días)

Equipo:

Ingeniero de DevOps

Administrador de Sistemas

Soporte Técnico

Actividades: Despliegue del sistema en producción, formación del personal, soporte y mantenimiento inicial.

1.6.2 Buenas Prácticas

Para asegurar el éxito del proyecto, se implementarán las siguientes buenas prácticas:

Gestión Ágil del Proyecto: Utilización de Scrum para gestionar las tareas y sprints, garantizando la entrega continua de valor y la adaptación a cambios durante el desarrollo.

Control de versiones: Uso de Git para el control de versiones y la gestión de código fuente, permitiendo un seguimiento riguroso de los cambios y facilitando la colaboración entre desarrolladores.

Documentación Completa: Mantener una documentación detallada y actualizada en todas las fases del proyecto, asegurando que todas las decisiones, cambios y avances sean registrados adecuadamente.

Pruebas Continuas: Implementación de un ciclo de pruebas continuas desde el desarrollo hasta la fase de despliegue, para detectar y corregir errores de manera temprana.

Revisiones Periódicas: Realización de revisiones periódicas del proyecto con todos los miembros del equipo y los stakeholders, para asegurar el alineamiento con los objetivos del proyecto.

1.6.3 Condiciones Necesarias para la Implementación

Para la correcta implementación del sistema, se deben considerar las siguientes condiciones:

Recursos Tecnológicos: Disponibilidad de herramientas y tecnologías adecuadas, como servidores para pruebas y producción, entornos de desarrollo integrados (IDEs), y plataformas de integración continua.

Capacitación del Personal: Formación previa del equipo de desarrollo en tecnologías y metodologías específicas utilizadas en el proyecto, como SOAP, Django, y la base de datos Oracle.

Colaboración Activa de Stakeholders: Participación activa y continua de los stakeholders en la definición de requisitos y validación de entregables, asegurando que el sistema cumpla con sus expectativas y necesidades.

Acceso a Información y Sistemas Externos: Asegurar el acceso adecuado y oportuno a los sistemas externos con los que se integrará el proyecto, como servicios web y bases de datos, para evitar retrasos en la integración.

Plan de Contingencia: Definición de un plan de contingencia para abordar posibles riesgos, como retrasos en el cronograma, problemas técnicos, o cambios en los requisitos, garantizando que el proyecto pueda ser completado dentro del tiempo y presupuesto asignado.

### 4.1.2 Definición del Equipo de Trabajo

#### Descripción del Equipo

* **Pablo Toro - Líder de Proyecto:**Pablo es el responsable de la planificación, coordinación y supervisión general del proyecto. Su función principal es asegurar que todas las actividades del proyecto se realicen dentro de los plazos y el presupuesto establecido. Además, actúa como el principal enlace entre el equipo de desarrollo y los stakeholders, gestionando la comunicación y asegurando que los objetivos del proyecto se cumplan. También es responsable de la gestión de riesgos y de la asignación de recursos, garantizando que todas las áreas del proyecto estén alineadas y trabajando hacia los objetivos comunes.
* **Franco Ventura - Analista de Base de Datos:**Franco se encarga del diseño y gestión de la base de datos Oracle que soportará el sistema **Airescold**. Entre sus responsabilidades se encuentran la creación de esquemas de base de datos, la optimización de consultas SQL, y la garantía de la integridad y seguridad de los datos. Franco también juega un papel crucial en la integración de la base de datos con los servicios web SOAP, asegurando que la información se maneje de manera eficiente y segura a lo largo de todo el sistema.
* **Ignacio Alvear - Programador:**Ignacio es el desarrollador principal del sistema, encargado de la implementación del CRUD para la gestión de inventarios y la integración de servicios web SOAP que permitirán la comunicación entre el sistema y otras plataformas. Además, Ignacio lidera el desarrollo de la tienda en línea utilizando tecnologías como Python, Django, HTML, CSS y JavaScript. Su rol incluye la resolución de problemas técnicos y la colaboración con Franco para asegurar una integración fluida entre la base de datos y el sistema web.

### 4.1.3 Definición de Actividades principales del Proyecto

Enumeres las etapas principales del proyecto de acuerdo a carta Gantt y EDT

### 4.1.4 Resumen Costos del Desarrollo del Proyecto

* Costos
* Costo por Fase 1: $17.500.000
* Costo por Fase 1: $17.500.000
* Costo por Fase 1: $17.500.000
* Costos por Actor o Rol total Proyecto
* Costo total en Remuneraciones, honorarios e incentivos: $30.492.000
* Costo total del proyecto: $52.500.000

### 4.1.6 Carta Gantt

las fechas establecidas del proyecto con el cliente fueron las siguientes:

#### Fechas Importantes y Hitos del Proyecto

1. **Inicio del Proyecto**: 14/08/2024
2. **Hito 1: Entrega de Documentos Fase 1**
   * **Fecha**: 04/09/2024
3. **Hito 2: Entrega de Modelo Arquitectónico Fase 2**
   * **Fecha**: 16/10/2024
4. **Hito 3: Documentos Fase 3**
   * **Fecha**: 04/12/2024
5. **Fecha de Finalización del Proyecto**: 04/12/2024

## 5. Anexos

### 5.1 Acta de Proyecto

[Insertar Acta de Constitución del Proyecto](https://docs.google.com/document/d/1fmss8DixS9NgEUxNVqwa2Lt-9qrz_22q/edit?usp=drive_link)

### 5.2 Matriz Especificación de Requerimientos

[Insertar la Matriz en formato planilla sobre la especificación de Requerimientos](https://docs.google.com/spreadsheets/d/1eTet2MwgauAWVu4dw0_eeIGksQebvlJP/edit?usp=drive_link&ouid=117192360844811438578&rtpof=true&sd=true)

### 5.3 Prototipado de Software

[Insertar documento con Mockups de las interfaces de usuario del Sistema](https://docs.google.com/document/d/1yjeK7cZa1szjohFvprC9YDtPX_fmJ0Pv/edit?usp=drive_link&ouid=117192360844811438578&rtpof=true&sd=true)

### 5.4 Matriz EDT. Planilla Detallada Cálculo de Esfuerzo

[[Insertar matriz EDT en formato Planilla que nos permite realizar el cálculo de estimación de esfuerzo en base a jornadas laborales.]](https://docs.google.com/spreadsheets/d/1ONttofpg3bnXkDUvxfbmLUeek4eiyM1Q/edit?usp=drive_link&ouid=117192360844811438578&rtpof=true&sd=true)

### 5.5 Planilla Carta Gantt

[Insertar Excel de la carta Gantt](https://docs.google.com/spreadsheets/d/1End1vMHDQ0cP4BIZ9LVfluZLAprXltrw/edit?usp=drive_link)