Duoc Uc, sede San Joaquín, ingeniería en informática 2024.

Joudy Valdivia

Erick Fuentes

Moisés Sáez

Anthony Perkins

Arquitectura de software (das)

Proyecto Homologación

**Contenido**

[1 Alcance 2](#_Toc183385457)

[2 Arquitectura de Software 2](#_Toc183385458)

[● Vista de Casos de Uso y Escenarios de Calidad: 2](#_Toc183385459)

[● Vista de Metas y Restricciones: 3](#_Toc183385460)

[● Vista Lógica: 4](#_Toc183385461)

[● Vista de Procesos: 4](#_Toc183385462)

[● Vista de Implementación: 5](#_Toc183385463)

[● Nodos Principales 8](#_Toc183385464)

[● Despliegue General 9](#_Toc183385465)

[Aprobaciones 13](#_Toc183385466)

**Introducción**

El Documento de Arquitectura de Software (DAS) tiene como objetivo describir de manera detallada la estructura y los componentes del sistema de homologación para Dolphin Medical. Este sistema busca automatizar y optimizar el proceso de homologación de productos médicos, permitiendo una gestión eficiente de cotizaciones y la interacción entre usuarios y datos clave. El DAS servirá como una guía técnica para el equipo de desarrollo y garantizará que todos los miembros involucrados compartan una visión común sobre la arquitectura del sistema.

# Alcance

Este documento abarca la descripción de los módulos principales del sistema, incluyendo:

* El frontend, diseñado para que los usuarios interactúen con el sistema de manera intuitiva.
* El backend, que procesará las solicitudes, gestionará los datos y garantizará la seguridad del sistema.
* La base de datos, encargada de almacenar información clave como productos, usuarios y proveedores.
* El uso de herramientas específicas como PdfPlumber para extraer información de catálogos en formato PDF.

# Arquitectura de Software

La arquitectura del sistema **Sistema de Homologación** está representada siguiendo el enfoque del **framework 4+1** y las recomendaciones del **proceso unificado**. En esta versión del documento se incluyen las siguientes vistas:

# Vista de Casos de Uso y Escenarios de Calidad:

**Casos de uso**

**Actores principales:**

* Gerenta de Ventas: Consulta informes del sistema (dashboard) para evaluar la actividad de los usuarios, productos más solicitados e instituciones principales.
* Cotizadores: Buscan homologaciones de productos mediante códigos o descripciones y gestionan solicitudes de cotización.
* Vendedores en Terreno: Solicitan cotizaciones rápidas utilizando cajas predeterminadas.
* Asistente de Ventas: Recibe solicitudes de cotización enviadas a través del sistema para procesarlas en el sistema externo.

**Casos de Uso Asociados**

**Búsqueda de productos:**

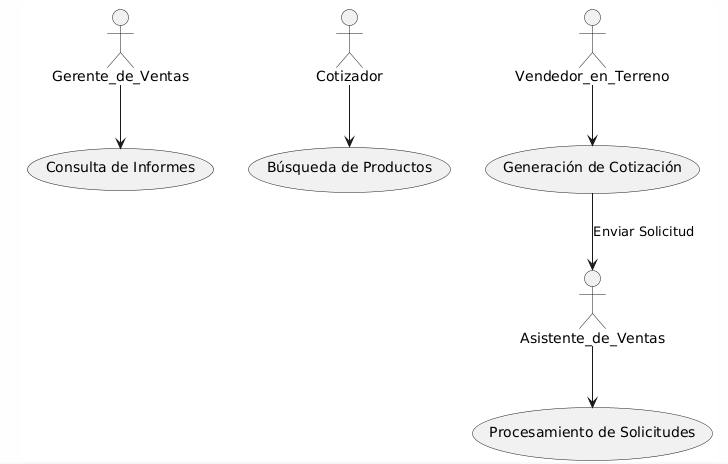
* Actor: Cotizador.
* Descripción: El sistema permite buscar productos por código o descripción, mostrando equivalencias, precios y tiempos de entrega de diferentes proveedores.

**Generación de Cotización:**

* Actor: Vendedor en terreno.
* Descripción: Selección de productos para generar solicitudes de cotización y envío al correo del asistente de ventas.

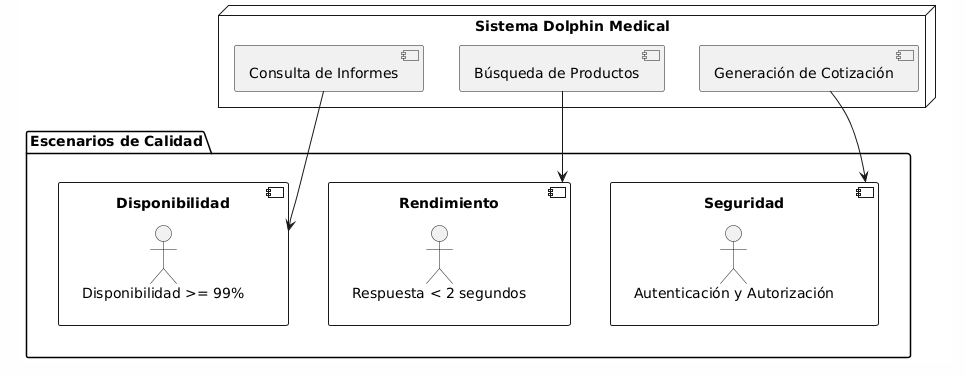
**Consulta de Informes:**

* Actor: Gerente de ventas.
* Descripción: Visualización de indicadores clave como productos más solicitados, usuarios activos e instituciones principales.



**Escenarios de calidad:**

* **Disponibilidad:** El sistema debe estar disponible el 99% del tiempo en horario laboral.
* **Rendimiento:** Tiempo de respuesta de menos de 2 segundos para búsquedas y homologaciones.
* **Seguridad:** Implementación de autenticación y autorización para prevenir accesos no autorizados.



# 

# Vista de Metas y Restricciones

**Restricciones tecnológicas:**

* Uso obligatorio de PostgreSQL como sistema de gestión de bases de datos relacionales.
* Herramientas específicas como PdfPlumber para extracción de datos de catálogos PDF.
* Desarrollo del frontend con React.js y backend con Node.js y Express.js.

**Normas y estándares:**

* Cumplir con normativas internas de Dolphin Medical para la protección de datos sensibles.
* Uso de estándares RESTful para la API.
* Implementación de seguridad mediante JWT y cifrado de contraseñas.

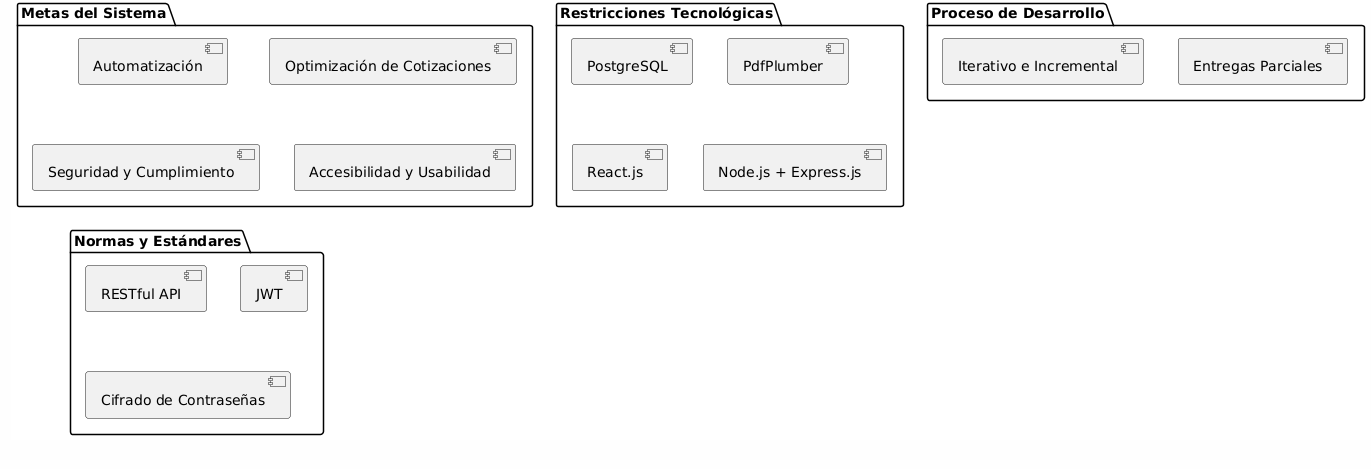
**Restricciones del Proceso de Desarrollo:**

* Metodología iterativa e incremental.
* Revisión continua de avances con entregas parciales funcionales en cada iteración.

Diagrama de Componentes:

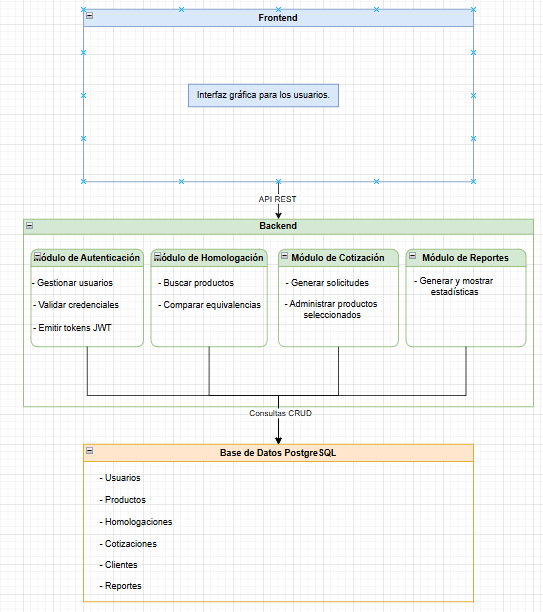


Diagrama de Paquetes



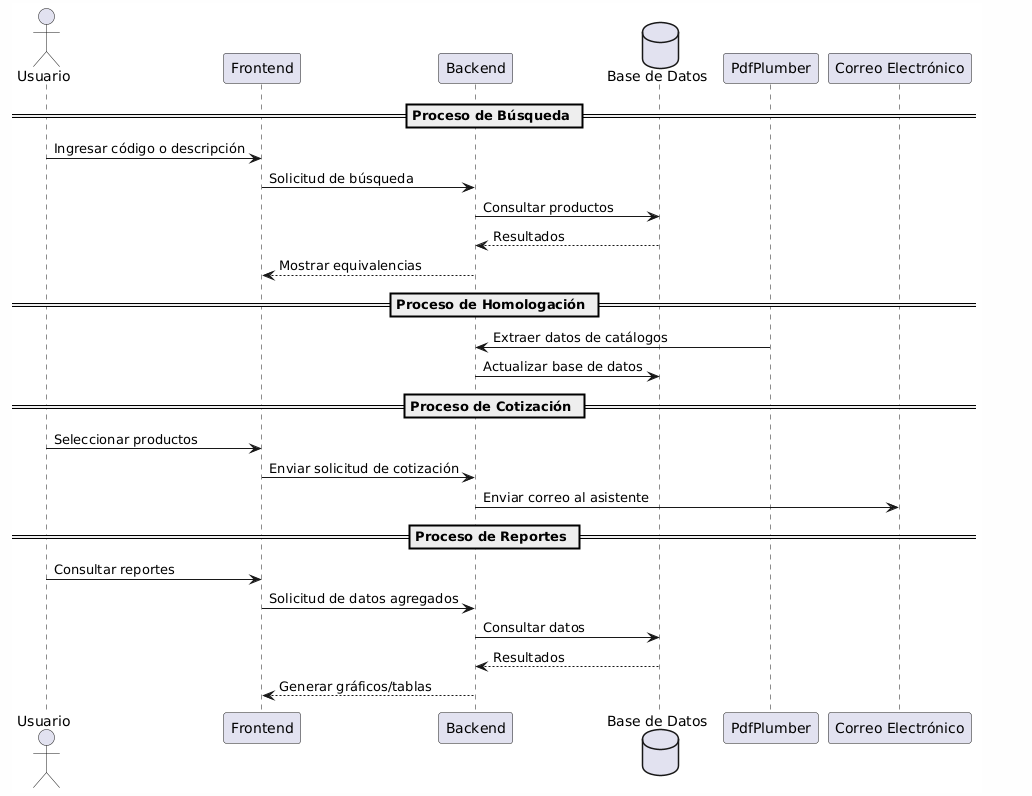
# Vista Lógica:

* Descripción General: La estructura lógica del sistema se organiza en módulos principales:
* Módulo de Autenticación: Gestiona el acceso al sistema y asegura la identidad de los usuarios mediante JWT.
* Módulo de Homologación: Permite la búsqueda y comparación de productos, mostrando equivalencias y referencias.
* Módulo de Cotización: Gestión de solicitudes de cotización, incluyendo selección de productos y generación de formularios.
* Módulo de Reportes: Proporciona estadísticas e informes para la gerencia de ventas.
* Responsabilidades y Dependencias:
* Los módulos del backend dependen de la base de datos PostgreSQL para almacenar y recuperar datos.
* El frontend interactúa con los módulos del backend mediante API REST.



# Vista de Procesos:

* Descripción de los procesos:
* Proceso de Búsqueda: El usuario ingresa un código o descripción, y el sistema busca equivalencias en la base de datos.
* Proceso de Homologación: Se extraen datos de catálogos PDF para enriquecer la base de datos.
* Proceso de Cotización: El usuario selecciona productos y genera una solicitud que se envía automáticamente a la asistente de ventas.
* Proceso de Reportes: Se generan gráficos y tablas con datos agregados para análisis.
* Relaciones de Comunicación y Sincronización:
* Las solicitudes del frontend son procesadas por el backend
* Los datos extraídos de los catálogos son sincronizados con la base de datos mediante procesos por lotes.



# Vista de Implementación:

Componentes Construidos:

* Interfaz:
  + Tecnología: React.js.
  + Dependencias: Axios, React Router DOM.

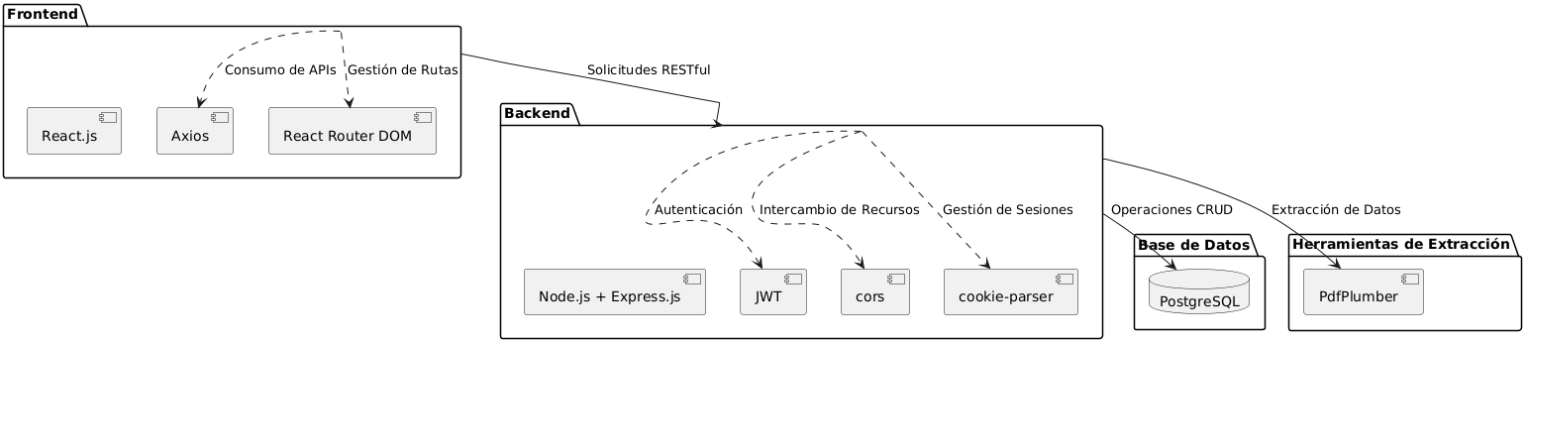
Parte posterior:

* Tecnología: Node.js con Express.js.
* Dependencias: Jwt, cors, cookie-parser.

Base de datos: PostgreSQL con índices para optimización.

Herramientas de extracción: PdfPlumber para análisis y extracción de datos de catálogos.

Dependencias: El backend depende de la base de datos para las operaciones CRUD y del PdfPlumber para extracción de datos.



1. **Objetivos y Restricciones de la Arquitectura**

**Objetivos de la arquitectura**

De acuerdo con las reuniones realizadas con los interesados ​​y el análisis de los requerimientos iniciales, se han identificado los siguientes conductores arquitectónicos, que corresponden a las metas arquitectónicas principales para el Sistema de Homologación:

Desempeño: El sistema debe garantizar tiempos de respuesta inferiores a 2 segundos en la mayoría de las operaciones, como búsquedas de productos y homologaciones.

La base de datos debe manejar consultas optimizadas para grandes volúmenes de productos y proveedores.

Razón: Los usuarios necesitan procesar solicitudes de homologación y cotizaciones de manera rápida para cumplir con los plazos de licitaciones y compras ágiles.

Tolerancia a Fallos: El sistema debe ser capaz de manejar errores inesperados, asegurando una interrupción mínima en las operaciones críticas.

Implementar mecanismos de respaldo para proteger la integridad de los datos en caso de fallas del servidor o interrupciones en la red.

Razón: La disponibilidad del sistema es crucial para que los usuarios puedan realizar homologaciones y cotizaciones sin retrasos significativos.

Seguridad: Implementar autenticación basada en JWT (JSON Web Tokens) para garantizar el acceso seguro.

Encriptar datos sensibles, como contraseñas y datos de usuarios, tanto en tránsito como en reposo.

Restringir el acceso a funciones avanzadas (como el tablero) únicamente a roles específicos (gerenta de ventas y gerente general).

Razón: Dolphin Medical maneja información sensible que debe protegerse contra accesos no autorizados o posibles ataques.

Modificabilidad/Reutilización: Diseñar una arquitectura modular que permita el uso de componentes y facilite la incorporación de nuevas funcionalidades en el futuro.

Crear módulos independientes, como el módulo de homologación, el módulo de cotización y el módulo de informes.

Razón: Esto asegura que el sistema pueda adaptarse a cambios o ampliaciones sin afectar su núcleo principal.

Operatividad: Garantizar que el sistema sea fácil de operar y mantener por el equipo técnico de Dolphin Medical.

Proveer documentación detallada y guías claras para el uso, implementación y mantenimiento del sistema.

Razón: La facilidad de uso y operación asegura una adopción más rápida del sistema por parte de los usuarios y un menor esfuerzo para resolver problemas técnicos en el futuro.

**Restricciones de la Arquitectura**

En las reuniones con los stakeholders, se han identificado y documentado las siguientes restricciones que influyen directamente en las decisiones arquitectónicas del sistema de homologación:

**Tiempo de Construcción**: El plazo para la construcción del sistema está ajustado, estableciéndose un máximo de 4 semanas para la entrega de la clave de funcionalidad, según lo definido en la planificación.

Impacto en la arquitectura:

* Es necesario priorizar las funcionalidades críticas del sistema (homologación, cotización e informes) y dejar las posibles extensiones o módulos adicionales para futuras iteraciones.
* Se adoptará una metodología iterativa e incremental para entregar valor al negocio dentro del tiempo establecido.

**Infraestructura**: La arquitectura debe aprovechar la infraestructura existente, que incluye: Servidores de aplicación replicados con balanceadores de carga. Una base de datos con estructura de clúster para soportar alta disponibilidad y rendimiento.

Impacto en la arquitectura:

* El sistema deberá ser compatible con esta infraestructura para garantizar escalabilidad y tolerancia a fallos.
* Los componentes del sistema deben diseñarse para aprovechar el equilibrio de carga y garantizar un uso eficiente de los recursos.

**Otros componentes de software:** No se contempla la adquisición ni el licenciamiento de componentes de software adicionales para el desarrollo del sistema.

Impacto en la arquitectura:

* Se utilizarán herramientas y tecnologías de código abierto, como:
  + Node.js y React.js para el desarrollo del backend y frontend, respectivamente.
  + PostgreSQL como base de datos relacionales.
  + PdfPlumber para extracción de datos desde catálogos PDF.
  + Las soluciones deben maximizar el uso de herramientas existentes y minimizar la dependencia de software comercial.

1. **Arquitectura física**

En esta vista se despliegan los nodos que participan con el sistema. Los nodos principales son los nodos Servidor de Integración. Características a continuación:

# Nodos Principales

1. Servidor de Aplicaciones:

* Aloja el backend del sistema, desarrollado en Node.js con Express.js, gestionando la lógica de negocio, las solicitudes de usuarios y la comunicación con la base de datos.
* Características:
* Replicado para alta disponibilidad.
* Balanceadores de carga para distribuir el tráfico entrante y garantizar escalabilidad.
* Configurado para manejar APIs RESTful con autenticación mediante JWT.

1. Servidor de Base de Datos:

* Aloja PostgreSQL, el sistema de gestión de bases de datos relacional utilizado para almacenar y recuperar datos de usuarios, productos y homologaciones.
* Características:
  + - Configuración en clúster para alta disponibilidad y rendimiento.
    - Optimización mediante índices y particionamiento de tablas para acelerar consultas.

1. Nodos Cliente (Usuarios Finales):

* Dispositivos de los usuarios que acceden al sistema web a través de navegadores.
* Características:
  + - Interacción con el frontend desarrollado en React.js.
    - Envío de solicitudes HTTP hacia el backend.

# Despliegue General

1. Flujo de Trabajo:
   * + Los usuarios acceden al sistema desde un navegador y envían solicitudes al servidor de aplicaciones a través de Internet.
     + El servidor de aplicaciones se comunica con el servidor de base de datos para realizar consultas y devolver resultados al usuario.
2. Seguridad:
   * + Cifrado de las comunicaciones entre los nodos mediante HTTPS.
     + Roles y permisos definidos para cada usuario, controlados desde el servidor de aplicaciones.
3. Alta Disponibilidad:
   * + Configuración de servidores en un entorno replicado con balanceadores de carga para soportar múltiples solicitudes concurrentes.
     + Clúster en la base de datos para evitar caídas del sistema ante fallos.



1. **Escenarios**

**Escenarios Funcionales Prioritarios**

Los escenarios funcionales se centran en las operaciones principales del sistema, que son críticas para su propósito.

Escenario de Búsqueda y Homologación de Productos

Un cotizador ingresa un código o descripción de producto en el sistema para buscar equivalencias. El sistema retorna opciones disponibles en el catálogo interno y de proveedores, incluyendo precios, tiempos de entrega y una imagen referencial.

Atributos de Calidad:

* + Usabilidad: La interfaz debe ser intuitiva para facilitar la búsqueda.
  + Rendimiento: Los resultados deben mostrarse en menos de 2 segundos.

Escenario de Generación de Cotizaciones

Un vendedor o cotizador selecciona productos o cajas predeterminadas, personaliza las opciones, y genera una solicitud de cotización. Esta solicitud se envía automáticamente al correo de la asistente de ventas.

Atributos de Calidad:

* + Eficiencia: El proceso debe ser rápido y evitar redundancias.
  + Fiabilidad: La información enviada debe ser precisa y completa.

Escenario de Consulta de Reportes

La gerenta de ventas accede al dashboard para analizar datos consolidados, como los productos más solicitados, usuarios activos e instituciones que cotizan con mayor frecuencia.

Atributos de Calidad:

* + Disponibilidad: El dashboard debe estar disponible en horario laboral.
  + Accesibilidad: Solo usuarios con roles de administrador pueden acceder.

Escenario de Gestión de Usuarios

Un administrador crea, edita o elimina cuentas de usuario según sea necesario. Esto incluye asignar roles específicos a los usuarios.

Atributos de Calidad:

* + Seguridad: Solo administradores autorizados pueden realizar cambios.
  + Escalabilidad: El sistema debe manejar múltiples usuarios sin degradar su rendimiento.

Escenario de Carga de Catálogos

El administrador carga catálogos en formato PDF. El sistema extrae automáticamente códigos, nombres y descripciones de productos, validando los datos antes de almacenarlos.

Atributos de Calidad:

* + Eficiencia: La carga debe ser automatizada y rápida.
  + Fiabilidad: Los datos extraídos deben ser validados para evitar errores.

**Escenarios No Funcionales Prioritarios**

Los escenarios no funcionales están relacionados con los atributos de calidad del sistema, como rendimiento, seguridad y disponibilidad.

Escenario de Rendimiento

Durante una búsqueda, el sistema debe procesar grandes volúmenes de datos y retornar los resultados en menos de 2 segundos.

Atributos de Calidad:

* + Rendimiento: Optimización de consultas SQL e índices en la base de datos.

Escenario de Seguridad

El sistema debe garantizar que solo usuarios autenticados puedan acceder a funcionalidades específicas, como el dashboard.

Atributos de Calidad:

* + Confidencialidad: Uso de JWT para autenticación y roles.
  + Integridad: Encriptación de datos sensibles.

Escenario de Disponibilidad

El sistema debe estar disponible el 99% del tiempo en horario laboral para garantizar el acceso constante de los usuarios.

Atributos de Calidad:

* + Disponibilidad: Infraestructura con balanceadores de carga y servidores replicados.

Escenario de Escalabilidad

El sistema debe soportar un aumento en el número de usuarios y productos sin afectar el rendimiento.

Atributos de Calidad:

* + Escalabilidad: Diseño modular y uso de infraestructura escalable.

Escenario de Usabilidad

Los usuarios deben ser capaces de realizar homologaciones y generar cotizaciones sin requerir capacitación extensa.

Atributos de Calidad:

* + Usabilidad: Diseño intuitivo de la interfaz en React.js.

**Diagrama de Casos de Uso**

Descripción del Diagrama:

* Actores principales:
  + - Gerenta de Ventas: Accede a reportes y análisis de datos.
    - Cotizador: Realiza búsquedas de homologación y genera cotizaciones.
    - Vendedor en Terreno: Solicita cotizaciones rápidas.
    - Administrador: Gestiona usuarios y carga catálogos.
* Casos de uso clave:
  + - Búsqueda de productos.
    - Generación de cotizaciones.
    - Gestión de usuarios.
    - Consulta de reportes.
    - Carga de catálogos.

**Casos de Uso Prioritarios**

Se describieron anteriormente (en la sección de casos de uso) los detalles de los casos más importantes. Estos representan funcionalidades críticas y deben ser completados en las primeras iteraciones del proyecto.

**Modelo de Casos de Uso**

El modelo de casos de uso puede ser encontrado en el documento “Casos de Uso”.

**Especificación de Casos de Uso Relevantes**

A continuación se listan los casos de uso relevantes, los cuales pueden ser encontrados con su especificación detallada en el documento “Casos de Uso”.

|  |  |  |  |
| --- | --- | --- | --- |
| **Código** | **Nombre** | **Actores** | **Prioridad** |
| CU-01 | Búsqueda de Productos | Cotizador | Alta |
| CU-02 | Homologación de Productos | Cotizador | Alta |
| CU-03 | Generación de Solicitudes de Cotización | Cotizador, Vendedor en Terreno | Alta |
| CU-04 | Visualización de Reportes | Gerenta de Ventas | Alta |
| CU-05 | Gestión de Usuarios | Administrador | Media |
| CU-06 | Carga de Catálogos | Administrador | Alta |
| CU-07 | Recuperación de Contraseñas | Usuario | Media |
| CU-08 | Modificación de Cajas Predeterminadas | Cotizador, Vendedor en Terreno | Media |
| CU-09 | Visualización de Historial de Homologaciones | Cotizador | Media |
| CU-10 | Verificación de Disponibilidad de Productos | Cotizador, Gerenta de Ventas | Alta |
| CU-11 | Generación de Informes Detallados | Gerenta de Ventas, Gerente General | Alta |
| CU-12 | Notificación de Cambios de Estado de Solicitudes | Cotizador, Vendedor en Terreno | Media |

**Aprobaciones**

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Fecha** | **Firma** |
| Marisol Tapia, gerente de ventas, Dolphin Medical. | 28-08-2024 |  |
| Joudy Valdivia, Líder de proyecto, Jelyb Solutions. | 28-08-2024 |  |