|  |
| --- |
|  |
| **DAS (Documento Arquitectura de Sistema)** |
| Cristian Reyes  Marcos Olmedo  Javier Pailahueque  Juan Peña |

|  |
| --- |
|  |

**Contenido**

Contenido

[1. Introducción 3](#_Toc183531523)

[1.1 Propósito 3](#_Toc183531524)

[1.2 Alcance 3](#_Toc183531525)

[2. Descripción General de la Arquitectura 3](#_Toc183531526)

[2.1 Visión General 3](#_Toc183531527)

[2.2 Componentes Clave 3](#_Toc183531528)

[2.3 Principios de Diseño 4](#_Toc183531529)

[3. Detalle de los Componentes del Sistema 4](#_Toc183531530)

[3.1 Aplicación Móvil 4](#_Toc183531531)

[3.2 Interfaz Web 5](#_Toc183531532)

[Visualización de inventarios con filtros personalizados. 5](#_Toc183531533)

[3.3 Backend con Firebase 5](#_Toc183531534)

[3.4 Visualización Arquitectura 6](#_Toc183531535)

# 1. Introducción

## 1.1 Propósito

El presente documento describe la arquitectura del sistema NFC Ignite, diseñado para optimizar la gestión de inventarios en pequeñas y medianas empresas (Pymes) mediante tecnología NFC, Firebase y Flutter. Este documento será una guía para el equipo técnico y stakeholders, proporcionando una visión detallada del diseño, implementación y escalabilidad del sistema.

## 1.2 Alcance

El sistema consta de:

* Una aplicación móvil que permite el escaneo de etiquetas NFC y la gestión básica de inventarios.
* Una interfaz web para funcionalidades avanzadas, como exportación de datos y análisis.
* Un backend basado en Firebase, que gestiona la autenticación, almacenamiento y sincronización en tiempo real.

# 2. Descripción General de la Arquitectura

## 2.1 Visión General

La arquitectura del sistema sigue un modelo cliente-servidor distribuido, con una integración estrecha entre componentes front-end (móvil y web) y servicios backend (Firebase).

## 2.2 Componentes Clave

Front-end móvil:

Implementado en Flutter.

Funcionalidades principales: escaneo NFC, operaciones CRUD en inventarios.

Frontend web:

Permite la gestión avanzada de inventarios.

Incluye opciones para aplicar filtros y exportar datos.

Backend:

Firebase proporciona servicios de autenticación, base de datos y reglas de seguridad.

## 2.3 Principios de Diseño

Modularidad: Cada componente está desacoplado, facilitando el mantenimiento.

Escalabilidad: Capacidad para manejar múltiples usuarios y datos crecientes.

Usabilidad: Interfaces intuitivas para minimizar errores y aprendizaje.

Seguridad: Reglas estrictas en Firestore y autenticación robusta.

# 3. Detalle de los Componentes del Sistema

## 3.1 Aplicación Móvil

Tecnología: Flutter (Dart).

Arquitectura: MVVM (Modelo-Vista-ViewModel).

Funciones principales:

Escaneo de etiquetas NFC.

Registro, consulta y actualización de inventarios.

Estructura de Directorios en Flutter:

lib/views/screens/: Pantallas principales (e.g., inventory\_screen.dart, scanner\_screen.dart).

lib/views/widgets/: Componentes reutilizables (botones, formularios).

lib/business\_logic/: Lógica de negocio, conectada a Firebase.

## 3.2 Interfaz Web

Tecnología: Framework FLET (Python).

Funciones principales:

Gestión avanzada de inventarios.

Exportación de datos en Excel.

## Visualización de inventarios con filtros personalizados.

## 3.3 Backend con Firebase

Servicios Utilizados:

Firebase Authentication: Gestión de usuarios autenticados.

Firestore: Base de datos NoSQL con sincronización en tiempo real.

Storage: Almacenamiento de archivos exportados.

Reglas de Seguridad:

Acceso restringido según roles de usuario.

## 3.4 Visualización Arquitectura

Diagrama

Descripción generada automáticamente

1. **El Usuario Interactúa con el Frontend (Flutter)**: El usuario realiza una acción en la aplicación (como escanear un código NFC o agregar un producto al inventario).
2. **El Frontend envía la Solicitud al Backend (Dart):** Los datos de la acción se envían desde el frontend al backend móvil, que se encarga de procesarlos.
3. **El Backend Interactúa con Firebase:** El backend móvil consulta o guarda datos en el servidor Firebase a través de las APIs de Firebase.
4. **Firebase Procesa y Almacena los Datos:** Los datos se almacenan en la base de datos en tiempo real de Firebase. Si hay actualizaciones, todos los usuarios conectados a la aplicación reciben los cambios de inmediato.
5. **El Backend Responde al Frontend:** El backend envía una respuesta al frontend con el resultado de la acción (por ejemplo, validación de un código NFC o la lista de productos).
6. **El Frontend Muestra la Respuesta al Usuario:** Finalmente, la aplicación muestra al usuario los resultados de la acción, como la confirmación de la operación o los datos actualizados.