***TABLA DE CONTENIDO***

[1. INTRODUCCIÓN 2](#_Toc165893464)

[1.1 Objetivo del Documento 2](#_Toc165893465)

[1.2 Definiciones 2](#_Toc165893466)

[2. ARQUITECTURA EMPRESARIAL 2](#_Toc165893467)

[2.1 Alcance general de la Solución 2](#_Toc165893468)

[2.2 Alineación Estratégica 3](#_Toc165893469)

[2.3 Procesos de Negocio 3](#_Toc165893470)

[3. ARQUITECTURA DE APLICACIÓN 3](#_Toc165893471)

[3.1 Objetivos de la Arquitectura 3](#_Toc165893472)

[3.2 Restricciones de Arquitectura 3](#_Toc165893473)

[3.3 Decisiones de Arquitectura 4](#_Toc165893474)

[3.4 Seguridad 4](#_Toc165893475)

[3.5 Vista de Contexto 4](#_Toc165893476)

[3.6 Vista de Desarrollo 4](#_Toc165893477)

[3.7 Diseño común – Requerimientos transversales 5](#_Toc165893478)

[3.8 Persistencia 6](#_Toc165893479)

[3.8.1 Requerimientos de Base de Datos 6](#_Toc165893480)

[3.8.2 Modelo de datos Proyecto 6](#_Toc165893481)

[3.8.3 Modelo de datos Sistemas Implicados 6](#_Toc165893482)

[3.8.4 Modelo de datos Transversales 7](#_Toc165893483)

[3.8.5 Conexiones 7](#_Toc165893484)

[3.9 Vista de Despliegue 7](#_Toc165893485)

[3.10 Procesos automáticos 7](#_Toc165893486)

[3.11 Vista de Infraestructura 7](#_Toc165893487)

[3.11.1 Características de plataforma en tiempo de ejecución 8](#_Toc165893488)

[3.11.2 Proyección de capacidad requerida 8](#_Toc165893489)

[3.11.3 Servicios en la nube 8](#_Toc165893490)

[3.12 Atributos De Calidad/Requerimientos No Funcionales 9](#_Toc165893491)

[4. DISEÑO DETALLADO 10](#_Toc165893492)

# INTRODUCCIÓN

# Objetivo del Documento

El objetivo de este documento es describir el diseño de la arquitectura solución para el proyecto [nombre del proyecto]

# Definiciones

Los siguientes acrónimos y definiciones se usarán en el [proyecto] y dentro de este documento:

|  |  |
| --- | --- |
| ***Acrónimo*** | ***Definición*** |
|  |  |
|  |  |
|  |  |
|  |  |

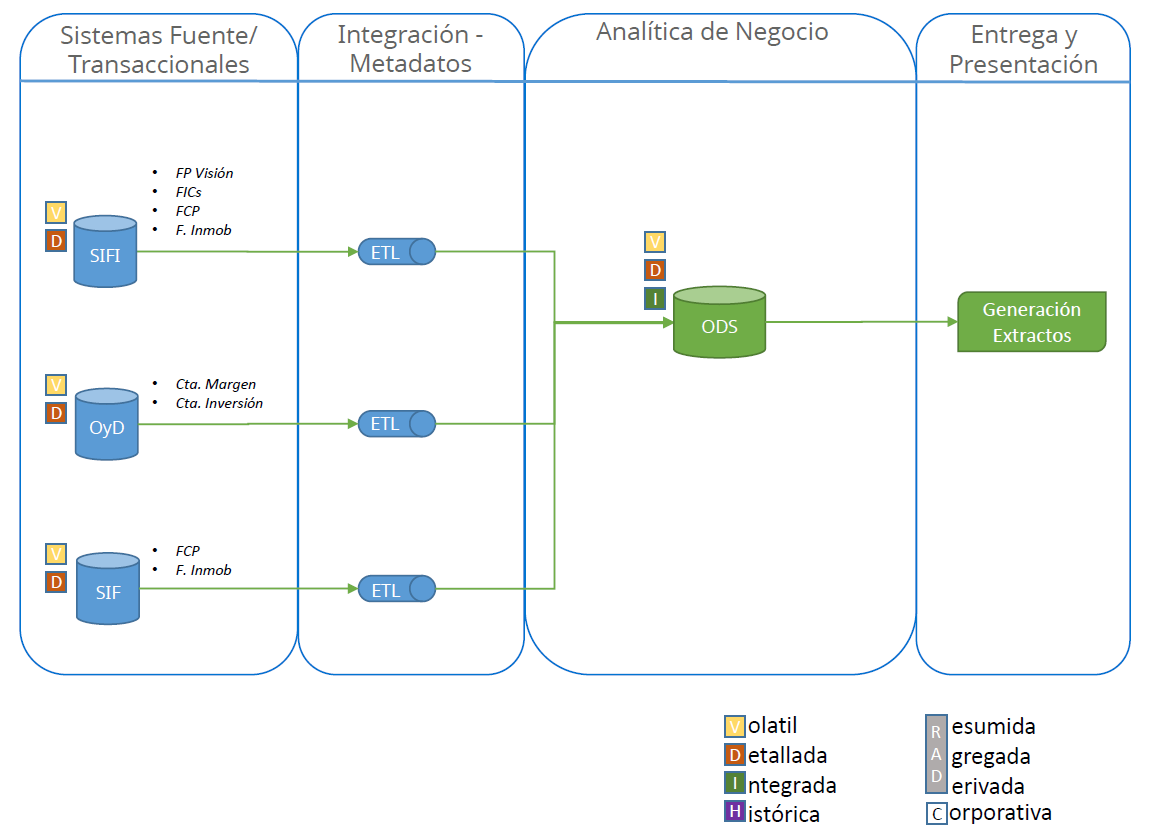
# ARQUITECTURA EMPRESARIAL

En esta sección se describen los ítems de arquitectura empresarial donde se permite identificar como se alinea el proyecto con la estrategia de negocio de la compañía.

# Alcance general de la Solución

El alcance permite describir los aspectos que están contemplados dentro de la solución y debe brindar una visión global de la problemática a resolver y de la arquitectura de solución del proyecto, con el fin de entregar las diferentes funcionalidades requeridas.

Esta sección debe hacer explícitos los objetivos que busca la implementación de la solución.



**Ilustración 1 Modelo Conceptual de la Solución**

# Alineación Estratégica

Esta sección debe identificar los objetivos estratégicos o metas con las que se relaciona la implementación de la solución, precisando como la puesta en producción de la solución busca alcanzar o mejorar el logro de indicadores de desempeño.

En esta sección se puede indicar también la relación con otros proyectos en curso.

# Procesos de Negocio

Identificación de los procesos de negocio impactados o que impactan la solución. Debe describirse de qué manera la solución debe ajustarse a los procesos o como los procesos deben ser modificados con la puesta en producción de la solución.

# ARQUITECTURA DE APLICACIÓN

# Objetivos de la Arquitectura

En esta sección se describen las especificaciones donde se relacionan la necesidad del proyecto y como estas son gestionadas por la definición de los objetivos que deben cumplirse con el planteamiento de arquitectura

# Restricciones de Arquitectura

En esta sección se describen los requerimientos de solución y objetivos que tienen un impacto arquitecturalmente significativo sobre el proyecto [nombre del proyecto]

# Decisiones de Arquitectura

Con el objetivo de realizar una organización racional del trabajo de arquitectura y cumplir con los requerimientos planteados se han tomado un conjunto de decisiones las cuales han sido justificadas en los siguientes puntos:

Ej.: Los elementos de solución de más alto significado serán los servicios, debido a que representan los artefactos contenedores de la lógica de negocio de [nombre del proyecto] estos encapsularan su funcionalidad.

# Seguridad

En esta sección se deben especificar los controles relacionados con la gestión y el cumplimiento de los permisos de acceso en sistemas, aplicaciones y recursos: métodos de autenticación y autorización, etc.

Esta sección debe ser validada y aprobada por la Gerencia de Ciberseguridad., aspectos tales como:

* + - * + Permisos en Firewall
        + Permisos en carpetas u otros recursos
        + Requisitos adicionales de seguridad (cifrado, llaves públicas-privadas)
        + Protocolos seguros

**NOTA**: Los aspectos mencionados anteriormente pueden aplicar o no sobre el proyecto en cuestión

# Vista de Contexto

Se utiliza un diagrama de contexto (y una explicación de apoyo) para explicar el entorno en el que opera el sistema y las entidades externas con las que interactúa. Define brevemente cada una de las entidades externas y las interacciones importantes que el sistema tiene con ellas.

El diagrama de contexto suele presentarse como una imagen simple de alto nivel que muestra los límites del sistema y sus entidades externas adyacentes. Las entidades externas suelen ser otros sistemas, pero también pueden ser dispositivos físicos, organizaciones externas de "caja negra" o componentes de software más granulares. Las interacciones pueden ser flujos de datos (interfaces) o flujos de control (por ejemplo, invocando un servicio o función expuesta).

# Vista de Desarrollo

La vista de desarrollo del sistema define las restricciones en el proceso de desarrollo de software que son requeridas por la arquitectura. Esto incluye la organización del sistema mediante módulos o paquetes, el procesamiento común que todos los módulos deben implementar, cualquier estandarización de diseño, codificación y pruebas requerida y la organización del código del sistema.

Aquí es necesario registrar algunas decisiones importantes desde el punto de vista arquitectónico, por ejemplo, acerca de la elección de librerías o marcos, o el enfoque y herramientas para la implementación de software o la gestión de configuración.

# Diseño común – Requerimientos transversales

Define el diseño común de algunas funciones (como el registro, la seguridad, el rastreo, etc.) que deben realizarse de manera estándar en todo el sistema y cómo debe realizarse (por ejemplo, mediante un patrón de diseño o una referencia a una librería de código o un ejemplo).

**Identificación:** Describir la forma como las conexiones se van a identificar.

**Autenticación**: Deben mencionarse las estrategias, tácticas o mecanismos de autenticación para evitar que el software sea vulnerable a ataques de suplantación de identidad, ataques de diccionario y otros tipos de ataques.

**Autorización**: Deben mencionarse las estrategias, tácticas o mecanismos de autorización para evitar que el software sea vulnerable a la divulgación de información, manipulación de datos y elevación de privilegios.

**Caché**: Deben incluirse las estrategias, tácticas o mecanismos para mejorar el rendimiento y la capacidad de respuesta del software a través del uso de caché.

**Manejo de Excepciones**: Deben documentarse en esta sub-sección las estrategias, tácticas o mecanismos para la gestión de excepciones de manera que se garantice la seguridad y confiabilidad del software. Dichas decisiones deben facilitar el diagnóstico y resolución de problemas sin revelar información sensible y crítica que haga el software vulnerable a ataques.

**Logs e Instrumentación**: Deben describirse las estrategias, tácticas o mecanismos para evitar que el software sea vulnerable a las amenazas de repudio, donde los usuarios niegan sus acciones, y los archivos de registro pueden servir de prueba de las acciones ejecutadas. Debe auditarse y registrarse la actividad en las capas del software para ayudar a detectar actividades sospechosas y proporcionar una indicación temprana de un ataque serio.

**Manejo de estados**: Deben describirse las estrategias, tácticas o mecanismos para administrar la persistencia de los datos que representan el estado de un componente, operación, o paso en un proceso. Los datos del estado pueden persistir usando diferentes formatos y tipos de almacenamiento.

**Validación**: Deben describirse las estrategias, tácticas o mecanismos para evitar que el software esté expuesto a datos incoherentes, violaciones de reglas de negocio y una experiencia de usuario deficiente. Dichas estrategias, tácticas o mecanismos deben evitar que el software sea vulnerable a problemas como ataques de cross-scripting, ataques de inyección SQL, desbordamientos de búfer, y otros tipos de ataques de entrada.

**Roles y Perfiles:** Deben describirse las estrategias, tácticas o mecanismos para gestionar los roles y perfiles sobre la aplicación y como se relaciona con los procesos de autenticación transversal. Confirmar y verificar si los roles y perfiles deben hacer parte del proceso de aprovisionamiento de usuarios y realizar la respectiva solicitud al equipo de aprovisionamiento.

# Persistencia

# Requerimientos de Base de Datos

En esta sección se deberá definir el motor de Base de Datos a utilizar, la versión, el espacio en disco, memoria RAM, CPU que usará la Base de Datos.

Crear tabla por cada ambiente con todas las bases de datos que aplican para proyecto.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Características Servidores Bases de Datos** | | | | | | |
| **Base de Datos** | **Máquina** | **Versión** | **Requerimiento** | **Cantidad** | **Fecha de Actualización** | **Periodicidad Backup** |
|  |  |  | Procesamiento CPU (Vcores) |  |  |  |
|  |  |  | Memoria RAM - GB |  |  |  |
|  |  |  | Espacio en Disco - GB |  |  |  |
|  |  |  | Procesador |  |  |  |
|  |  |  | Sistema Operativo |  |  |  |
|  |  |  | Ubicación Servidor |  |  |  |
|  |  |  | Contenedores |  |  |  |
|  |  |  | Versión Docker |  |  |  |

# Modelo de datos Proyecto

Se debe agregar el modelo entidad-relación de la base de datos que forma parte de la solución. Este diagrama debe incluir al menos: nombres de las tablas, sus atributos, un detalle de las relaciones entre cada una de las tablas, índices y llaves de las tablas.

NOTA: Detallar objetivo del modelo de dados (impacto, procesos de pruebas, etc)

# Modelo de datos Sistemas Implicados

Se debe agregar el modelo entidad-relación de las tablas relacionadas de los sistemas existentes en Alianza que hacen parte de la solución. Este diagrama debe incluir al menos: nombres de las tablas, sus atributos, y un detalle de las relaciones entre cada una de las tablas.

# Modelo de datos Transversales

Se debe agregar el modelo entidad-relación donde se especifican las tablas transversales para el proyecto en cuestión y otros proyectos que compartan el modelo de base de datos. Este diagrama debe incluir al menos: nombres de las tablas, sus atributos, y un detalle de las relaciones entre cada una de las tablas.

# Conexiones

En esta sección se relacionan todos los temas adicionales de las bases de datos que hacen parte del proyecto, especificando cantidad o listando nombres, campos de entrada o de salida, paquetes o ítems relevantes.

Se plantea la siguiente tabla como ejemplo o se podría gestionar la información como un anexo del documento.

|  |  |  |
| --- | --- | --- |
| **Base de Datos** | **Recurso** | **Cantidad** |
|  | Pool de conexiones |  |
|  | Servicios |  |
|  | Conexión por servicio |  |
|  | Procedimientos Almacenados |  |
|  | Funciones |  |

# Vista de Despliegue

La vista de despliegue del sistema define las características importantes del entorno de despliegue operativo del sistema. Esta vista incluye los detalles de los nodos de procesamiento que el sistema requiere para su instalación (es decir, su plataforma de ejecución), las dependencias de software en cada nodo (tales como las librerías necesarias) y los detalles de la red subyacente que el sistema requerirá. Adicionalmente se deben especificar los tipos y los puertos de conexión entre los componentes.

Crear vista por cada ambiente (ambientes previos, preproducción, producción, contingencia) si aplica.

# Procesos automáticos

En esta sección se relacionan todos los procesos automáticos que se ejecuten dentro de la solución arquitectónica como jobs, triggers, crons o cualquier tipo de proceso autónomo. Se debe especificar el detalle del proceso, objetos de la base de datos relacionados como procedimientos o vistas, otros componentes como servicios o servidores de archivos, campos de entrada, campos de salida, su periodicidad de ejecución y quien lo usa.

# Vista de Infraestructura

La vista de infraestructura de la aplicación permite visualizar los componentes físicos que soportan el despliegue de los componentes, se debe contemplan todos los componentes que se integran con el proyecto a nivel interno y externo, puertos y protocolos de conexión y la especificación de infraestructura general de Alianza como WAF o Balanceador de carga.

Crear vista por cada ambiente (ambientes previos, preproducción, producción, contingencia) si aplica.

# Características de plataforma en tiempo de ejecución

Muestra las especificaciones de la infraestructura que soporta la aplicación en tiempo de ejecución. Relacionar toda la información de los componentes de la aplicación Front-End, Back-End, Componentes de Almacenamiento, etc.

Crear tabla por cada ambiente.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Características Servidores Producción** | | | | | |
| **Componente** | **Máquina** | **Requerimiento** | **Cantidad** | **Fecha de Actualización** | **Periodicidad Backup** |
|  |  | Procesamiento CPU (Vcores) |  |  |  |
|  |  | Memoria RAM - GB |  |  |  |
|  |  | Espacio en Disco - GB |  |  |  |
|  |  | Procesador |  |  |  |
|  |  | Sistema Operativo |  |  |  |
|  |  | Ubicación Servidor |  |  |  |
|  |  | Contenedores |  |  |  |
|  |  | Versión Docker |  |  |  |

# 

# Proyección de capacidad requerida

Se debe especificar la capacidad requerida inicial y la proyección anual de crecimiento en términos de Procesamiento, Memoria, Almacenamiento, Licenciamiento, Usuarios y demás elementos indispensables para garantizar la operación.

# Servicios en la nube

En esta sección se deben especificar todas las características de los servicios del proyecto que se encuentran desplegados en la nube, así como el tipo de interacción que se maneja.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Detalle de Servicios en la Nube** | | | | | | | |
| **Recurso** | **Descripción** | **Proveedor** | **Nube** | **Normas** | **Disponibilidad** | **Comunicación** | **Contingencia** |
|  |  |  |  | *ISO 27001* |  |  |  |
|  |  |  |  | *ISO 27017* |  |  |  |
|  |  |  |  | *ISO 27018* |  |  |  |
|  |  |  |  | *SOC1* |  |  |  |
|  |  |  |  | *SOC2* |  |  |  |
|  |  |  |  | *SOC3* |  |  |  |

**NOTA:** Diligenciar esta información si aplica.

# Atributos De Calidad/Requerimientos No Funcionales

Un atributo de calidad (QA) es una propiedad medible o comprobable de un sistema que se utiliza para indicar qué tan bien el sistema satisface las necesidades de sus partes interesadas. Puede pensarse que un atributo de calidad mide la "bondad" de un producto en alguna dimensión de interés para un actor.

Un requerimiento para un atributo de calidad debe ser inequívoco y comprobable. Se requiere usar el siguiente formato común para especificar todos los requisitos de atributos de calidad *(Len, Paul, & Rick, 2012)*:

|  |  |
| --- | --- |
| **Identificador** |  |
| **Atributo(s) de calidad relevante(s)** |  |
| **Objetivo(s) de Negocio correspondiente(s)** |  |
| **Fuente** |  |
| **Estímulo** |  |
| **Artefacto** |  |
| **Ambiente** |  |
| **Respuesta** |  |
| **Medida de la respuesta** |  |

# DISEÑO DETALLADO

Esta se deberán relacionar los documentos bajo el formato ***Documento de diseño técnico*** por cada requerimiento.

|  |  |  |
| --- | --- | --- |
| **Requerimiento** | **Relación Anexo** | **Descripción del cambio.** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |