**DOCUMENTO DE ARQUITECTURAS DE SOFTWARE**

**<<DESCRIBIR EL NOMBRE DEL SISTEMA DE INFORMACIÓN>>**

**INSTITUTO COLOMBIANO DE BIENESTAR FAMILIAR**

**DIRECCIÓN DE INFORMACIÓN Y TECNOLOGÍA**

**SUBDIRECCIÓN DE SISTEMAS DE INTEGRADOS DE INFORMACIÓN**

**VERSIONAMIENTO DEL DOCUMENTO**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha**  **(De la Versión del documento que se está actualizando)** | **Versión** | **Autor** | **Descripción del cambio** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**TABLA DE CONTENIDO:** Estructura del documento que contiene en orden los temas que hacen parte del documento.

**Contenido**

[1. ABREVIATURAS Y DEFINICIONES 5](#_Toc18488923)

[2. INTRODUCCIÓN 6](#_Toc18488924)

[3. ALCANCE 6](#_Toc18488925)

[4. INTERESADOS 6](#_Toc18488926)

[5. REQUERIMIENTOS NO FUNCIONALES 7](#_Toc18488927)

[5.1 Seguridad 7](#_Toc18488928)

[5.2 Interoperabilidad 8](#_Toc18488929)

[5.3 Disponibilidad 8](#_Toc18488930)

[5.4 Desempeño 9](#_Toc18488931)

[5.5 Escalabilidad 9](#_Toc18488932)

[5.6 Mantenibilidad 10](#_Toc18488933)

[5.7 Auditabilidad 10](#_Toc18488934)

[6. PRIORIZACION DE ATRIBUTOS DE CALIDAD 10](#_Toc18488935)

[7. TÁCTICAS DE ARQUITECTURA 11](#_Toc18488936)

[8. VERFICACIÓN DE TÁCTICAS DE ARQUITECTURA 12](#_Toc18488937)

[9. LINEAMIENTOS TÉCNICOS 12](#_Toc18488938)

[9.1 Componentes 12](#_Toc18488939)

[10. VISTAS DE ARQUITECTURA 13](#_Toc18488940)

[10.1 Consideraciones Vista lógica 14](#_Toc18488941)

[10.1.1 Elementos de la Arquitectura 14](#_Toc18488942)

[10.1.2 BluePrint de Servicios de Negocio 14](#_Toc18488943)

[10.2 Vista de Implementación 15](#_Toc18488944)

[10.2.1 Elementos de la Arquitectura y Frameworks 16](#_Toc18488945)

[10.2.2 Diagrama de Componentes 16](#_Toc18488946)

[10.3 Vista de Integración 17](#_Toc18488947)

[10.3.1 Interfaces con Sistemas internos y externos 18](#_Toc18488948)

[10.3.2 Diagrama de integración portal Servicios – Sistemas Externos 18](#_Toc18488949)

[10.4 Vista de Despliegue 19](#_Toc18488950)

[10.4.1 Consideraciones Generales 20](#_Toc18488951)

[10.4.2 Elementos de la Arquitectura 20](#_Toc18488952)

[10.5 Vista de Amenazas 22](#_Toc18488953)

# ABREVIATURAS Y DEFINICIONES

Describe la definición de las abreviaturas y definiciones que contiene el documento, así:

* **Framework**: Es un marco de trabajo y/o un diseño reutilizable para un sistema de software, expresado como un conjunto de clases y la forma como colaboran entre ellas.
* **WSDL**: Web Service Description Language, por sus siglas en inglés. Es el lenguaje utilizado para describir servicios Web.
* **XSD**: XML Schema Definition, por sus siglas en inglés. Lenguaje de esquema utilizado para describir la estructura y las restricciones de los contenidos de los documentos XML.
* **JSON:** Java Script Object Notation.
* **API:** Interface de programación de aplicaciones, es un conjunto de funciones que se ofrecen para ser utilizado por otros sistemas y/o componentes de software.
* **Sistema de Información:** Conjunto de elementos relacionados que recolectan, procesan, almacenan y distribuyen información para apoyar la toma de decisiones y el control en una organización.
* **Interface:** Es la conexión entre dos sistemas de información y/o módulos y/o servicios y/o componentes que representa una comunicación que permite el intercambio de información.
* **Capa:** Agrupación lógica de elementos que permite estructurar responsabilidades y ofrecer funcionalidades que están relacionadas con un propósito particular.
* **REST**: Transferencia de Estado Representacional (Representational State Transfer)

Nota. Se deben adicionar las abreviaturas y/o definiciones que apliquen y den claridad al contenido del documento.

# INTRODUCCIÓN

Breve resumen donde se explican los aspectos más relevantes del proyecto para hacerse una idea de este, así:

El presente documento tiene como objetivo principal “La definición de la arquitectura de software para la construcción del aplicativo <<Nombre Aplicativo>>, permitiendo así identificar los requisitos funcionales y técnicos”, adicionalmente permite trazar el alcance y objetivo de la solución conceptual propuesta de cara a las necesidades y expectativas de los usuarios.

A partir de lo anterior, el presente documento estructura la arquitectura de software a través de las diferentes vistas de arquitectura que permita así, a todos los interesados e involucrados comprender como las soluciones de TI cumplen con los requerimientos funcionales.

El presente documento se formula bajo la necesidad de lograr la definición conceptual de la solución Informática.

# ALCANCE

Describe las características y nivel de aplicación del proyecto

# INTERESADOS

Describe los interesados que tienen relación directa con las soluciones de software, cuáles son sus necesidades de información de la arquitectura y que vistas de arquitectura pueden cubrir sus necesidades o expectativas.

Por ejemplo: a continuación, se identifican los interesados y sus necesidades sobre una solución de software.

**Responsables del seguimiento de la información en el ICBF**

Directivos del ICBF, Directivos del área, Líder funcional del aplicativo

**Profesional de la Subdirección de Sistemas Integrados de Información con rol Arquitecto**

Toma las principales decisiones de diseño de un sistema y de documentar esas decisiones para los distintos interesados que se relacionan de manera directa con la implementación del sistema.

Orienta la definición y documentación de los lineamientos a aplicar sobre el ciclo de vida de los proyectos, identificar sus riesgos técnicos y operativos, y brinda acompañamiento técnico para la realización de las actividades relacionadas con la estructuración de software de los sistemas de información aplicando las metodologías y estándares de acuerdo con los lineamientos de la SSII.

**Profesional de la Subdirección de Sistemas Integrados de Información con rol profesional responsable de los sistemas**

El responsable de los sistemas misionales, apoyo, estratégicos y de evaluación, se encarga de planificar, organizar y controlar el desarrollo de los sistemas de información.

**Profesional de la Subdirección de Sistemas Integrados de Información con rol desarrollador**

Genera el código de software necesario para obtener los resultados esperados en el cumplimiento de los requerimientos funcionales, de acuerdo con la arquitectura de software establecida y los atributos de calidad identificados.

Realiza las actividades requeridas y relacionadas para el desarrollo de software de los sistemas de información y la generación de material de apoyo relacionado, aplicando mejores prácticas, metodologías y estándares de acuerdo con los lineamientos de la SSII. Transforma los requerimientos suministrados en programas y desarrollos que hacen parte de un sistema de información.

**Profesional de la Subdirección de Sistemas Integrados de Información con rol profesional de pruebas**

Determina si la solución de software construida cumple con los requisitos funcionales y atributos de calidad del sistema.

Realiza el alistamiento, logística y ejecución de las pruebas funcionales y no funcionales de los diferentes sistemas de información cuando estas sean requeridas, reportando los resultados con sus respectivos soportes e informes sobre el mismo. Revisa que los productos suministrados por desarrollo correspondan a las necesidades documentadas en la etapa de análisis y especificación de requerimientos.

**Profesional de la Subdirección de Sistemas Integrados de Información con rol analista**

Estructura las especificaciones y requerimientos de software relacionadas con nuevos sistemas de información, nuevas funcionalidades y versiones de los sistemas de información con base en las necesidades del área solicitante.

**Profesional(es) de la Subdirección de Recursos Tecnológicos con rol que apoya la supervisión de los servicios de infraestructura y Bases de datos**

Asigna recursos de hardware y de ambiente productivo a los componentes de software construidos.

# REQUERIMIENTOS NO FUNCIONALES

Estos describen características relacionadas con la calidad del servicio, ofrecida por el sistema de información. Para poder medir el cumplimiento de estos requerimientos no funcionales, se deben definir métricas concretas. Por ejemplo: la seguridad de un sistema se puede especificar en términos del número y tipo de vulnerabilidades encontrados tras ejecutar pruebas sobre el mismo.

La priorización de requerimientos no funcionales es fundamental, ya que muchos de estos se oponen entre sí; por ejemplo, la seguridad del sistema va en contra de su desempeño.

Los Requerimientos no funcionales, los cuales se identificarán con el prefijo RNF, hacen relación al conjunto de atributos de calidad de software que definen las cualidades diferentes a las operativas del sistema, éstos se detallarán de tal manera que plasmen las necesidades expresadas por el usuario sobre lo que se espera de la aplicación, y así, poder medir la calidad técnica del software.

A continuación, se describe cada uno de los requerimientos asociados al atributo de calidad, la métrica para las pruebas se especifica en los escenarios de calidad.

## Seguridad

Nivel de protección de los recursos utilizados por los sistemas de información, contra riesgos (daño, pérdida, actividad criminal, etc.).

|  |  |
| --- | --- |
| **Módulo** | Nombre de la aplicación |
| **ID** | RNF—1 |
| **Nombre del Requerimiento** | Autenticación |
| **Descripción del Requerimiento** | Los actores del sistema se autenticarán a través del módulo existente para tal fin (Modulo Transversal de Seguridad) o a través del servicio de api management de azure. |

|  |  |
| --- | --- |
| **Módulo** | Nombre de la aplicación |
| **ID** | RNF—2 |
| **Nombre del Requerimiento** | Transferencia segura |
| **Descripción del Requerimiento** | El sistema debe transferir los datos de manera segura a través del protocolo de aplicación HTTPS. |

|  |  |
| --- | --- |
| **Módulo** | Nombre de la aplicación |
| **ID** | RNF—3 |
| **Nombre del Requerimiento** | Código Seguro |
| **Descripción del Requerimiento** | Cumplimiento de calidad de código seguro alineado al TOP 10 de OWASP[[1]](#footnote-2) |

|  |  |
| --- | --- |
| **Módulo** | Nombre de la aplicación |
| **ID** | RNF--<<Consecutivo>> |
| **Nombre del Requerimiento** |  |
| **Descripción del Requerimiento** |  |

Nota. Se deben adicionar los que apliquen a la solución tecnológica

## Interoperabilidad

Capacidad del sistema de información para integrarse con otros sistemas, construidos en plataformas distintas, con el objetivo de intercambiar información relevante al negocio.

|  |  |
| --- | --- |
| **Módulo** |  |
| **ID** | RNF--<<Consecutivo>> |
| **Nombre del Requerimiento** |  |
| **Descripción del Requerimiento** |  |

Nota. Se deben adicionar los que apliquen a la solución tecnológica

## Disponibilidad

El porcentaje de tiempo que un sistema se encuentra operable, durante un periodo determinado de tiempo. Por ejemplo, una disponibilidad del 99.5% anual se refiere a:

365 \* 24 = 8760

8760 \* 99.5% = 8716

Número de horas que el sistema puede estar por fuera de línea, al año = 44 horas

|  |  |
| --- | --- |
| **Módulo** |  |
| **ID** |  |
| **Nombre del Requerimiento** |  |
| **Descripción del Requerimiento** |  |

Nota. Se deben adicionar los que apliquen a la solución tecnológica

## Desempeño

Tiempos de respuesta de los diferentes servicios funcionales del sistema de información

|  |  |
| --- | --- |
| **Módulo** |  |
| **ID** |  |
| **Nombre del Requerimiento** |  |
| **Descripción del Requerimiento** |  |

Nota. Se deben adicionar los que apliquen a la solución tecnológica

## Escalabilidad

Habilidad de un sistema de información para manejar cantidades crecientes de carga o trabajo. Escalabilidad vertical es la capacidad del sistema para manejar mayores cantidades de carga (por ejemplo, más usuarios concurrentes) a través del aumento de recursos de maquina (CPU, memoria RAM, memoria física, etc.). Escalabilidad horizontal es la capacidad del sistema para manejar mayores cantidades de carga, haciendo uso de un número mayor de servidores.

Escalabilidad de Base de Datos indica su habilidad para reaccionar y adaptarse sin perder calidad y performance, o bien manejar el crecimiento continuo de trabajo de manera fluida, o bien para estar preparada para un crecimiento en el almacenamiento de los datos

|  |  |
| --- | --- |
| **Módulo** | Nombre de la aplicación |
| **ID** | RNF--<<Consecutivo>> |
| **Nombre del Requerimiento** | Autogestión de datos |
| **Descripción del Requerimiento** | El sistema debe tener la capacidad de auto gestionar sus datos a través de la generación de módulos de administración global. |

Nota. Se deben adicionar los que apliquen a la solución tecnológica

## Mantenibilidad

Facilidad para el mantenimiento de un sistema de información, lo cual involucra:

* Corregir errores
* Desarrollar nuevos requerimientos del cliente
* Adaptarse a cambios ambientales (reglas de negocio, reglamentación legal, etc.)
* Autogestión de datos

|  |  |
| --- | --- |
| **Módulo** |  |
| **ID** |  |
| **Nombre del Requerimiento** |  |
| **Descripción del Requerimiento** |  |

Nota. Se deben adicionar los que apliquen a la solución tecnológica

|  |  |
| --- | --- |
| **Módulo** |  |
| **ID** |  |
| **Nombre del Requerimiento** |  |
| **Descripción del Requerimiento** |  |

## Auditabilidad

Capacidad del sistema para llevar un control de auditora de acuerdo con las necesidades del negocio. Ejemplos del tipo de información que normalmente se audita son:

* Accesos al sistema
* Accesos y modificaciones a los recursos del sistema
* Transacciones

|  |  |
| --- | --- |
| **Módulo** |  |
| **ID** |  |
| **Nombre del Requerimiento** |  |
| **Descripción del Requerimiento** |  |

Nota. Se deben adicionar los que apliquen a la solución tecnológica

Nota. Incluir todas las calidades sistémicas que considere para el sistema o la solución a desarrollar.

# PRIORIZACION DE ATRIBUTOS DE CALIDAD

Para realizar la priorización cada uno de los requerimientos no funcionales, primero en conjunto con todas las partes involucradas del proyecto, se identifican los atributos de calidad más significativos de acuerdo con las necesidades del proyecto, para tener la claridad de que NO todos los requerimientos se pueden cumplir.

Priorización de los atributos de calidad seleccionados de acuerdo con los requerimientos no funcionales y su impacto en el sistema:

Luego estos requerimientos son calificados usando dos parámetros:

* El primer valor de prioridad determina la importancia para el éxito de la aplicación.
* El segundo es el riesgo o dificultad para conseguirlo.

Las Posibles calificaciones son: H: High; M: Medium; L: Low[[2]](#footnote-3). El resultado de esta priorización permite determinar los escenarios con prioridad alta, media y baja. Se presentan en orden de priorización:

|  |  |  |  |
| --- | --- | --- | --- |
| Atributo de Calidad | ID | REQUERIMIENTO NO FUNCIONAL | PRIORIDAD |
| Seguridad | RNF-01 | Autenticación | (H,M) |
| Seguridad | RNF-02 | Transferencia Segura | (H,L) |
| Seguridad | RNF-03 | Código Seguro | (H,M) |
|  |  |  |  |

Para el caso en que dos o más atributos de calidad tengan la misma importancia para su priorización, el desempate se realizara por medio de juicio de experto, entre los responsables de los sistemas y los arquitectos de software.

# TÁCTICAS DE ARQUITECTURA

A continuación, se enumeran las tácticas de arquitectura que podrán ser seleccionadas para atender los atributos de calidad que se definan.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atributo de Calidad | Táctica de Arquitectura | Patrón de Arquitectura | Descripción | Tecnología de Referencia |
| Seguridad  Requerimientos Atendidos  RNF-1  RNF-2  Se cubre el 100% de los requerimientos no funcionales de Seguridad | Resistir Ataques | Autenticación de Usuarios    Autorización de Servicios | Los usuarios deben autenticarse a través de un usuario y contraseña para poder hacer uso del sistema de información. | Integración con el componente de autorización del ICBF |
| No repudio | Auditoria de eventos de aplicación y base de datos | Se registran todos los eventos del sistema | Integración con componentes de auditoria |
| Resistir Ataques | Limitar la exposición | Utilización de herramientas Firewall para bloquear el acceso a los recursos del sistema. Los servidores de base de datos pueden estar en una zona militarizada, solo accesible desde la red interna | Top 10 OWASP. |

# VERFICACIÓN DE TÁCTICAS DE ARQUITECTURA

Se ilustra una matriz que muestra la relación entre las tácticas de arquitectura seleccionadas y los respectivos escenarios y atributos de calidad. De esta forma se logra verificar que la arquitectura este completa y balanceada, verificando que no haya escenarios sin tácticas, que no haya tácticas repetidas o que estén en conflicto.

| ATRIBUTO DE CALIDAD | ESCENARIO DE CALIDAD | TÁCTICA DE ARQUITECTURA | CRITERIOS DE SELECCIÓN |
| --- | --- | --- | --- |
| Seguridad | Autenticación  Autorización  TOP 10 OWAS | Resistir ataques | La autenticación para usuario externo e internos contra el módulo de autenticación que provee el ICBF que se incorporara en el sistema.  Validación de los lineamientos de OWASP y principalmente en el conjunto de 10 vulnerabilidades que debe resistir como mínimo en un sistema de información. |

# LINEAMIENTOS TÉCNICOS

En esta sección se enumeran los elementos centrales de la arquitectura, con el objetivo de facilitar el proceso de control de cambios solicitados por el cliente, evidenciando su relación con los requerimientos y casos de uso del proyecto.

Adicionalmente, este listado de componentes e interfaces tiene como propósito facilitar la construcción del diseño detallado, al enumerar explícitamente los elementos necesarios para construir los diagramas de dicho documento.

## Componentes

En esta sección se listan todos los componentes de la arquitectura, al igual que la definición de sus contratos o interfaces internas de comunicación, que pueden ser utilizadas por otros componentes, sistemas o usuarios.

* **Componentes de reúso**

A continuación, se listan los componentes de la arquitectura que van a ser reutilizados.

| TITULO | PROPÓSITO | INTERFAZ INTERNA COMUNICACIÓN |
| --- | --- | --- |
| Modulo Transversal de Seguridad | Autenticación de usuarios externos del sistema | API de programación (servicios REST) |
| Central de Mensajes | Componente que permite el envió de notificaciones vía correo electrónico a los usuarios. | API de programación (servicios REST) |
| Servicio de auditoria | Componente que permite administrar los logs de auditoria de la aplicación | Conexión directa al repositorio de datos de auditoria |

* **Componentes por desarrollar**

A continuación, se ve el listado de los componentes de la arquitectura que van a ser diseñados, implementados y/o desarrollados para este proyecto. Cada uno de los componentes listados se especificará con mayor detalle en las vistas de arquitectura

|  |  |  |
| --- | --- | --- |
| TITULO | PROPÓSITO | OBSERVACIONES |
|  |  |  |
|  |  |  |
|  |  |  |

# VISTAS DE ARQUITECTURA

Elementos arquitectónicos que se mantendrán como estándar durante el desarrollo del aplicativo y que responden a requerimientos funcionales y no funcionales críticos del aplicativo; la intención de esta sección es poder validar la arquitectura del sistema.

A continuación, se describen cada una de las vistas que son contempladas en la arquitectura del proyecto:

| VISTA | PREOCUPACIONES | UML | STAKEHOLDER |
| --- | --- | --- | --- |
| Vista Lógica | Capacidades de negocio del sistema de información  Servicios de negocio del sistema de información | Diagrama SOMF[[3]](#footnote-4) | Usuarios Finales  Desarrolladores |
| Vista de Implementación | Modelo de desarrollo de las aplicaciones  Capas y componentes que deben desarrollarse | Diagrama de Componentes | Desarrolladores |
| Vista de interoperabilidad  e integración | Modelo de desarrollo de las integraciones e interoperabilidad.  Integración e interoperabilidad del sistema Web con los sistemas transaccionales existentes y componentes reutilizables | Diagrama trivadis  Diagrama de interacción | Desarrolladores  Integradores |
| Vista Física | Instalación, Administración y operación del producto en los diferentes ambientes.  Sistemas de computadores, redes y comunicaciones requeridos | Diagrama de despliegue | Personal de Infraestructura |
| Vista de amenazas | Vista lógica de alto nivel de las posibles amenazas que puedan afectar al sistema de información | Diagrama de modelado de amenazas | Desarrolladores  Arquitectos |

## Consideraciones Vista lógica

Con el fin de identificar los servicios y capacidades que la solución requiere, se ha seguido un método de Arquitectura de Solución que permite generar el BluePrint de Arquitectura, la cual se ha construido acercándose a la notación SOMF4.

De esta manera, se han identificado los siguientes Canales y Servicios:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Notación de elementos del Diagrama Lógico de Servicios | | | | |
| Zonas, Capacidades, Servicios y/o elementos Internos de la solución | | | | |
|  |  |  |  |  |
| Consumidor | Capacidades o servicios compuestos de otros más granulares | Zona de servicios. Agrupa cohesivamente las capacidades | Relación desacoplada entre los elementos | Capacidades o servicios compuestos de otros más granulares externos |

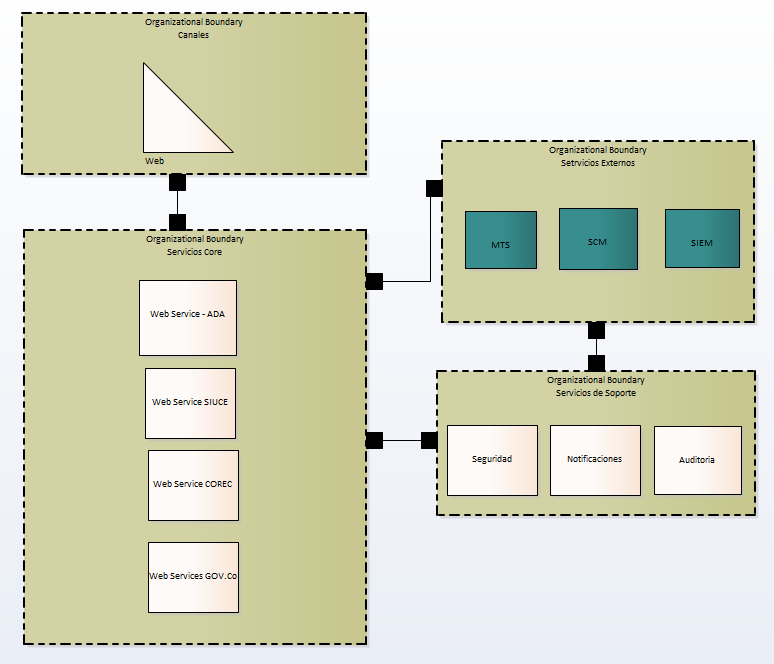
### Elementos de la Arquitectura

A continuación, se listan las diferentes zonas y servicios de negocio que integran el sistema:

|  |  |  |
| --- | --- | --- |
| ZONA | SERVICIO | DESCRIPCIÓN |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

### BluePrint de Servicios de Negocio

El siguiente diagrama ilustra los servicios de negocio identificados dentro de las zonas que sugieren una Arquitectura de Referencia. Este blue-print permitirá expresar las capacidades con las que cuenta la solución para que puedan mapearse a componentes tecnológicos durante el diseño



**Ilustración 1. Vista de Servicios de Negocio.**

## Vista de Implementación

La solución propuesta está basada en la estructuración de los componentes en capas lógicas basadas en una arquitectura de N capas, donde cada una de ellas tiene responsabilidades específicas y detalladas.

Como relevante para la arquitectura la construcción de una capa de integración que permitirá la comunicación entre los componentes de los sistemas y la comunicación con las aplicaciones externas basada en estándares de interoperabilidad SOA.

La arquitectura propuesta prioriza los atributos de alta disponibilidad y gran escalabilidad. De igual manera presenta escenarios de calidad que garantizan los tiempos de respuesta frente a la ejecución de las tareas y de los procesos, como de las consultas y reportes.

### Elementos de la Arquitectura y Frameworks

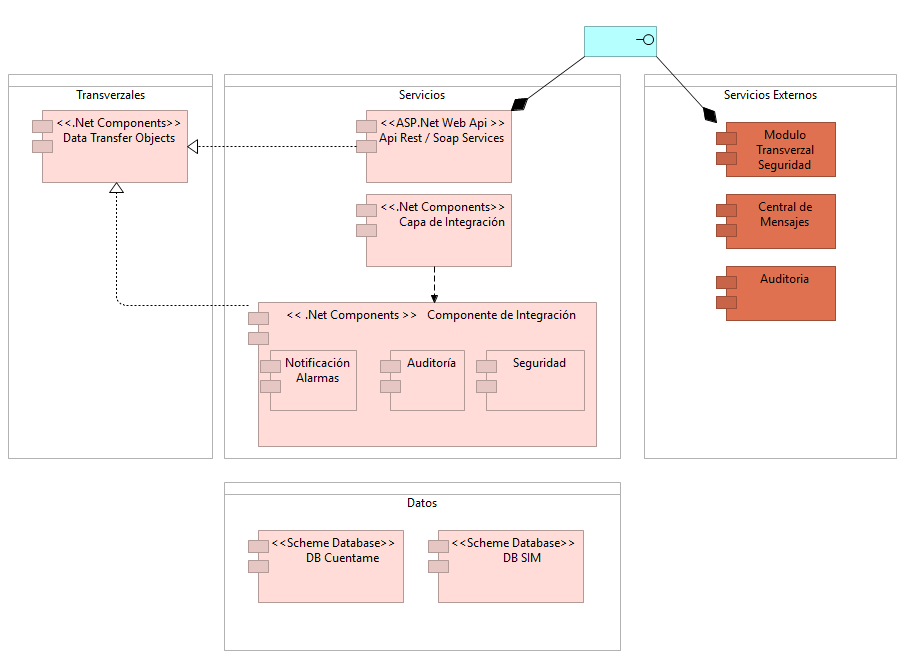
A Continuación, se listan los elementos de arquitectura que han sido seleccionados para abordar la implementación de los componentes que requieren construcción para el sistema. Los componentes de terceros que deben ser implementados requieren una evaluación razón por la cual para estos componentes no se identifican con una tecnología definida.

|  |  |
| --- | --- |
| ELEMENTO | DESCRIPCIÓN |
| WEBAPI ASP.NET - Servicios de Intercambio de información. | Se implementarán servicios REST como servicios de intercambio de información y componentes de lógica de negocio que requiere el sistema |
| .NET Framework – Capa de Integración | La capa de integración se construirá con .NET Framework como plataforma de integración y lograr desacoplar los componentes de negocio de la capa de presentación.  Esta capa de integración permitirá la implementación de una arquitectura SOA en el sistema. |
| DTO (Data Transfer Objects) | Se definirán a demanda modelos de entidades que representarán las entidades de transferencia de información y/o entidades de negocio transversales a todas las capas de la aplicación. |
| DB Transaccional | Modelos entidad relación de los sistemas CUENTAME y SIM. |

### Diagrama de Componentes

El objetivo de este diagrama es ilustrar las diferentes capas lógicas (paquetes) que se utilizarán para agrupar los componentes del sistema, logrando un nivel determinado de modularidad y mantenibilidad.

|  |  |  |  |
| --- | --- | --- | --- |
| Notación de los elementos del Diagrama de Componentes | | | |
|  |  |  |  |
| Componentes y frameworks requeridos en el sistema que requieren implementación | Componentes que requieren construcción dentro del alcance del sistema. | Componentes y servicios externos al sistema | Canales de comunicación entre componentes (Especifica protocolos y/o puertos) |



**Ilustración 2. Diagrama de Componentes.**

La organización y estructuración de estos componentes debe obedecer al principio de separación de responsabilidades en capas lógicas, donde cada una de ellas tiene responsabilidades específicas.

Capa de Servicios: componentes que encapsulan el acceso a recursos utilizados por el sistema. Tiene la responsabilidad de comunicar todos los componentes del sistema que se encuentran desacoplados. También provee la interfaz con los sistemas externos con los que debe interoperar el sistema.

Capa de datos: Contiene todos los recursos que con utilizados por el sistema. Bases de datos de configuración de la aplicación de servicios, almacenamiento de transacciones, Almacenamiento de archivos de auditoria y archivos.

Capa Transversal: Contiene los componentes de autenticación, autorización y auditoria del sistema, así como los value objects y entidades de negocio que requiere el sistema.

## Vista de Integración

La vista de integración se encarga de ilustrar el detalle de las integraciones entre el sistema por desarrollar y los sistemas legado interno o externos a la entidad.

### Interfaces con Sistemas internos y externos

A continuación, se identifican las interfaces requeridas por el sistema

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sistema Externo | Protocolo de Integración | Táctica de Arquitectura relacionada | Componentes Relacionados | Observaciones |
| Modulo Transversal de Seguridad | HTTPS | Patrones SOA | Capa de Integración, Servicios de Negocio | Sistema de Autenticación y Autorización del ICBF |
| Central de mensajes | HTTPS | Patrones SOA | Capa de Integración, Servicios de Negocio | Componente de ICBF que permite el envío de mensajes y notificaciones |
| Servicio de auditoria | TCP | ETL | Capa de datos | Componente que permite el registro de los logs de auditoria. |

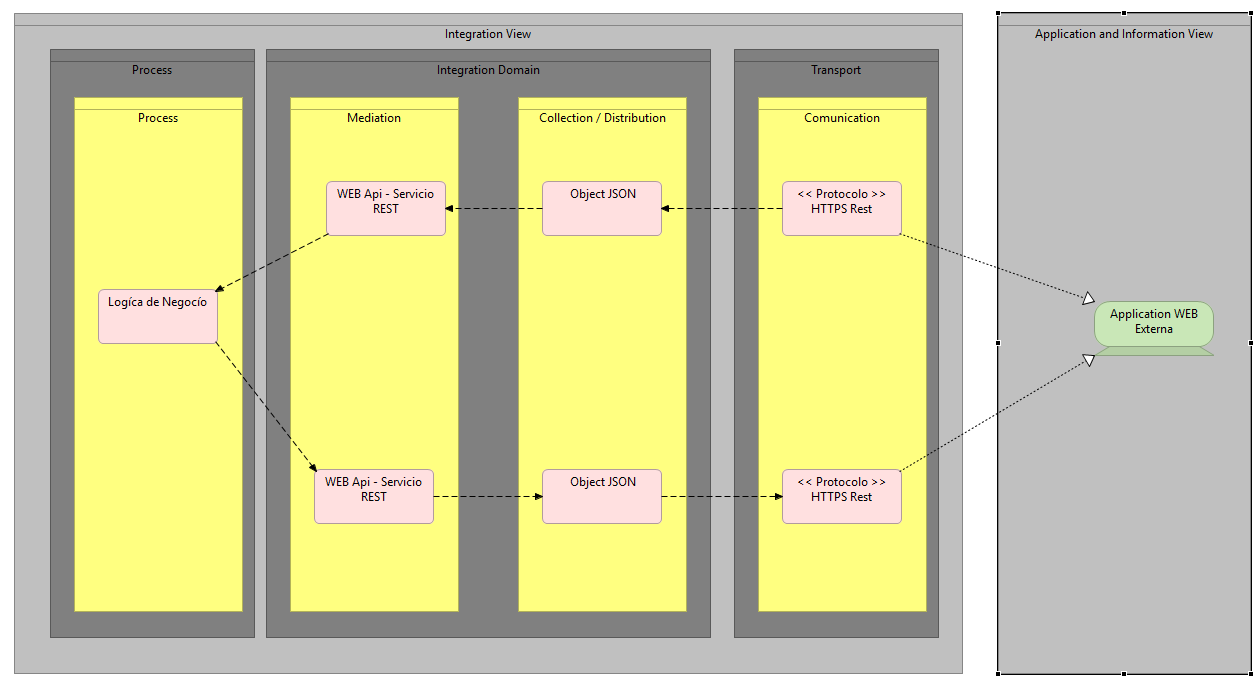
### Diagrama de integración portal Servicios – Sistemas Externos

El sistema debe estar en capacidad de interoperar a través de servicios REST y/o servicios SOAP, con los sistemas del ICBF que le proveen información.

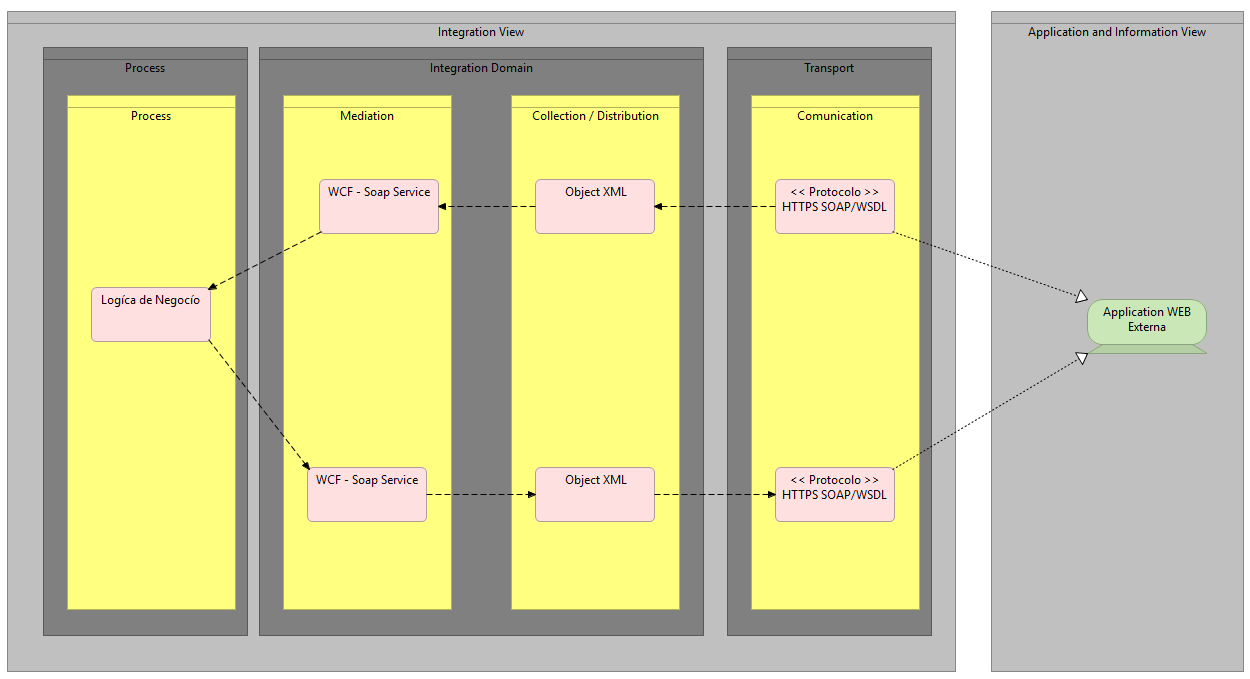
Los sistemas del ICBF con los cuales va a interoperar el pool de servicios a través de servicios REST y/o SOAP son:

* Modulo Transversal de Seguridad
* Central de Mensajes
* Servicio de Auditoria

1. Cada vez que el sistema requiera realizar consulta sobre un sistema externo consumirá un servicio REST y/o SOAP con la capacidad de consulta.
2. El portal se comunica a través de una solicitud HTTP con él envió de un objeto de negocio JSON y/o XML (Modelo Canónico construido)
3. El servicio realiza la comunicación con la lógica de negocio a través de la interfaz de consulta definida.
4. El componente de lógica de negocio devolverá una petición de consulta a través de un objeto JSON y/o XML.



**Ilustración 3. Diagrama de Integración vía HTTPS REST.**



**Ilustración 4. Diagrama de Integración vía HTTPS WSDL/SOAP.**

## Vista de Despliegue

La vista de despliegue (o vista física) describe la estructura del sistema en términos de la topología de los componentes en los equipos y la capa física, así como los mecanismos y equipos de comunicación que se usan.

### Consideraciones Generales

La infraestructura propuesta estará alojada en el centro de cómputo provisto por el Instituto Colombiano de Bienestar Familiar el cual ofrece la plataforma indicada en los supuestos y limitantes del proyecto.

Para el Sistema de Información, se proponen dos ambientes de despliegue:

* **Ambiente de pruebas**: Entorno para las pruebas de integración y las pruebas del sistema.
* **Ambiente de producción**: Entorno para la puesta en funcionamiento de las aplicaciones y la utilización por parte de usuarios internos y externos

### Elementos de la Arquitectura

A continuación, se describen los elementos físicos de la arquitectura para cada uno de los ambientes definidos, con la descripción de cantidad, función, componentes desplegados y las características físicas mínimas con las que se debe contar.

|  |  |
| --- | --- |
| ELEMENTO | DESCRIPCIÓN |
| Ambiente de Pruebas | Entorno para pruebas no funcionales y de usuario |
| Ambiente de Producción | Entorno para la puesta en funcionamiento de la solución |

* **Ambiente de Pruebas.**

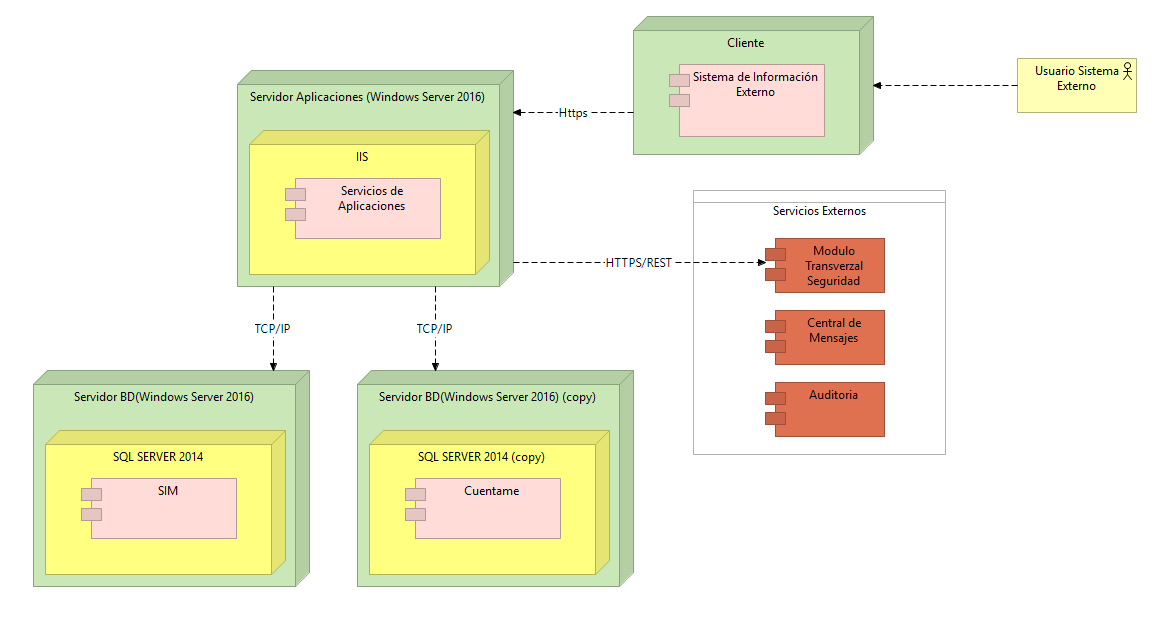
El ambiente de pruebas contiene la siguiente definición:

|  |  |  |  |
| --- | --- | --- | --- |
| EQUIPO | ESPECIFICACIONES | CANTIDAD | FUNCIÓN |
|  |  |  |  |

* **Diagrama Ambiente de Pruebas.**

El ambiente de pruebas representa una versión reducida (con un número menor de recursos, pero con la misma separación de componentes, y con los mismos mecanismos de comunicación entre ellos) del ambiente de producción y tiene como objetivo permitir lo siguiente: - Realizar pruebas funcionales por parte del equipo de calidad, interno y del cliente

* Realizar pruebas funcionales por parte del equipo de calidad, interno y del cliente.
* Realizar pruebas no funcionales (estrés, seguridad, desempeño, etc.) por parte del equipo de arquitectos. Con las métricas obtenidas en estas pruebas, se puede realizar el proceso de dimensionamiento de ambientes de producción.
* Documentar el proceso de despliegue e instalación del sistema de información.



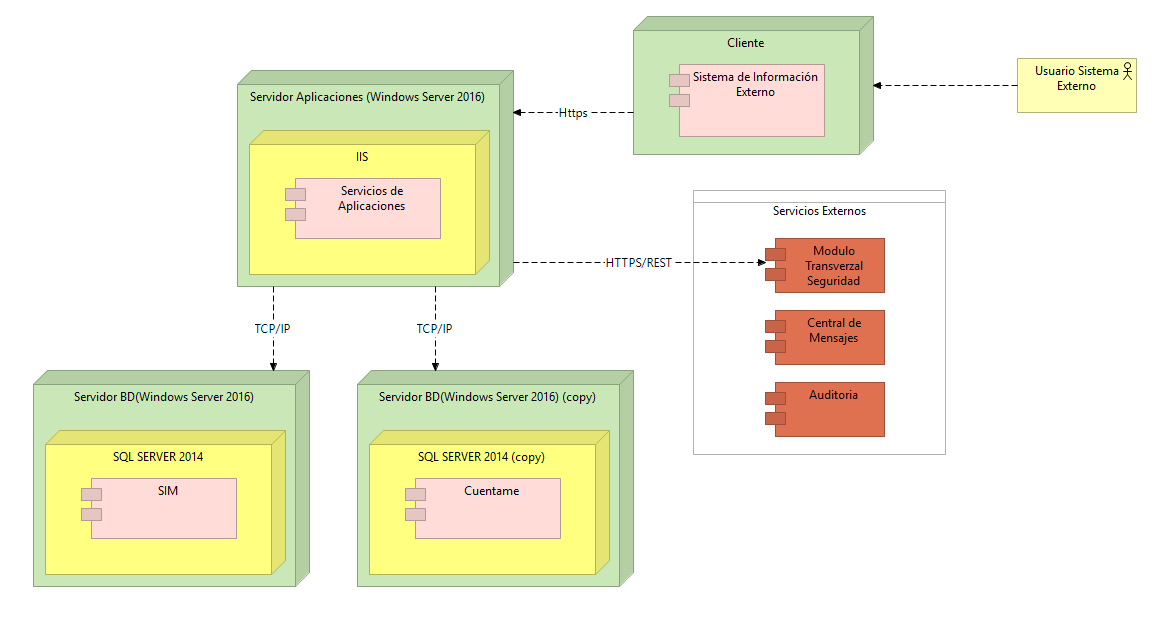
**Ilustración 4. Diagrama de Despliegue Ambiente Pruebas**

* **Ambiente de Producción.**

El ambiente de producción contiene la siguiente definición:

|  |  |  |  |
| --- | --- | --- | --- |
| EQUIPO | ESPECIFICACIONES | CANTIDAD | FUNCIÓN |
|  |  |  |  |

El ambiente de producción deberá soportar los niveles de calidad de servicio definidos por los requerimientos no funcionales. Para la definición de los diferentes componentes de hardware a ser utilizados se debe hacer uso del catálogo de patrones de arquitectura, definidos en el Documento de Diseño Detallado.



**Ilustración 5. Diagrama de Despliegue Ambiente Producción**

## Vista de Amenazas

La vista de posibles amenazas describe las posibles amenazas que pueden materializarse, a nivel de seguridad que puedan evidenciarse desde el diseño de la aplicación, para que estas puedan mitigarse en la etapa de construcción e implementación. Esta vista es opcional.

* **Diagrama de modelado de amenazas.**

1. <https://www.owasp.org/index.php/Main_Page> [↑](#footnote-ref-2)
2. Utility Trees. Consultado el 07 de Marzo de 2014. Disponible en: http://www.sei.cmu.edu/reports/00tr004.pdf [↑](#footnote-ref-3)
3. Notación SOMF para el lenguaje del modelo de descubrimiento de servicios de negocio en etapa de Análisis. https://sparxsystems.com/downloads/whitepapers/SOMF-2.1-Logical-Design-Model-LanguageSpecifications.pdf [↑](#footnote-ref-4)