# 

# DOCUMENTO DE ARQUITECTURA SAD

**Cliente**: <Nombre del cliente>

**Proyecto**: <Nombre del proyecto>

<dd de mes de aaaa>

# 

# 1 DESCRIPCIÓN

Los modelos y anotaciones definidos por un estándar de descripción de arquitecturas, como el modelo de vistas 4+1[[1]](#footnote-1), ayudan a capturar y comunicar los resultados del proceso de diseño. Por otra parte, como complemento a este documento, existe el documento de decisiones de arquitectura ya que responde por qué, quién y cómo.

Con este documento se describe la arquitectura de sistemas de software basado en el uso de múltiples vistas concurrentes ya que permite ofrecer distintas visualizaciones a los distintos stakeholders. Este documento se va refinando a medida que las decisiones arquitectónicas maduran y a medida que el proyecto va avanzando. Es decir, la documentación aquí plasmada no se hace una única vez.

# 2 PROPÓSITO

Este documento tiene como objetivo principal, resumir las funcionalidades básicas de la solución a construir y especificar las funcionalidades, restricciones técnicas y los atributos de calidad que resolverán todas las necesidades planteadas respecto a la arquitectura adaptada para el proyecto.

El documento de arquitectura de software proporciona una visión global de la arquitectura del sistema de software. Sirve como un medio de comunicación entre el arquitecto de software y otros miembros del equipo del proyecto respecto a las decisiones de gran importancia arquitectónica que se han hecho en el proyecto.

## 2.1 Impacto de no tener este producto de trabajo

Sin este producto de trabajo, sería muy complicado construir una solución sin haber tenido en cuenta los atributos de calidad y los posibles escenarios primarios en los cuales la solución debería comportarse de cierta manera. En este documento se define el comportamiento acordado de la solución en cuanto a atributos de calidad como desempeño, seguridad, escalabilidad, entre otros. Si esto no estuviera documentado antes de construir la solución, sin duda, no se podría satisfacer las necesidades de la misma.

De igual manera no se tendría las vistas generales de arquitectura en las cuales se pueda establecer el alcance general de la solución con el fin de cumplir con los requisitos funcionales y no funcionales.

## 2.2 Razones para no necesitar este producto de trabajo

Este producto de trabajo no es necesario si ya existe un documento similar donde se describa la generalidad del proyecto y los atributos de calidad que van a satisfacer la construcción de la solución y en el cual se genera una visualización clara para distintos stakeholders.

# 3 PASOS PARA ESTABLECER EL DOCUMENTO DE ARQUITECTURA

* Reuniones con el cliente para elicitar información de necesidades del negocio y requisitos funcionales.
* Reuniones con el cliente para elicitar información de restricciones técnicas identificadas tempranamente y que son claves para el buen desarrollo de la solución.
* Reuniones con los dueños de la solución para elicitar información acerca de restricciones de negocio previstos para la solución.
* Para la definición de la arquitectura, cada iteración debe tener:
  + Identificar requisitos principales.
  + Elección de escenarios críticos.
  + Arquitectura candidata (alto nivel) para descubrir elementos.
  + Los elementos se ubican dentro de las 4 vistas.
  + PoC de la arquitectura (implementación, pruebas, mediciones, análisis, planteamiento de mejoras).
  + Lecciones aprendidas.
  + Refinar arquitectura.
* Documentación de generalidades del proyecto (contexto y necesidad del negocio).
* Documentación de funcionalidades básicas de la solución.
* Documentación de restricciones técnicas y de negocio.
* Documentación de atributos de calidad para satisfacer las necesidades por medio de escenarios reales.

# 4 INTRODUCCIÓN

[Ingrese en esta sección la descripción del resto del contenido, explicando cómo el documento está organizado.]

Partiendo de las restricciones, requisitos y necesidades identificadas tempranamente, se busca contribuir a la documentación y a la toma de decisiones arquitectónicas de la solución. Este documento busca reflejar de una manera general, una visual a los distintos stakeholders.

# 5 GENERALIDADES DEL PROYECTO

## 5.1 Objetivos

[Ingrese los objetivos generales del proyecto.]

## 5.2 Alcance

[Ingrese el alcance del proyecto.]

# 6 DRIVERS DE ARQUITECTURA

Los Drivers de arquitectura que se definen en este documento pertenecen al alcance general del proyecto. No reemplazan a los que se definan en otros documentos del método aplicado. Sin embargo, los Drivers que se definan en otros documentos deben cumplir con los que aquí se documenten, pues éstos son aspectos generales a tener en cuenta.

## 6.1 Restricciones técnicas y de negocio.

## 6.2 Atributos de Calidad.

Los atributos de calidad son documentados por medio de escenarios, en los cuales se deja en evidencia, los posibles comportamientos del sistema teniendo en cuenta factores externos que en cualquier momento, pueden afectar negativamente o positivamente al mismo.

|  |  |
| --- | --- |
| **ID** | [ID del atributo] |
| **Atributo de calidad:** | [Tipo del atributo de calidad: Modificabilidad, Seguridad, Rendimiento, etc.] |
| **Descripción:** | [Descripción del atributo de calidad. Ej: Desempeño procesando la petición y generando la respuesta, sin incluir el tiempo de invocación del servicio externo.] |
| **Fuente del estímulo:** | [Quién realiza la estimulación a este atributo de calidad. Ej: Sistema Allus invocando el servicio expuesto por la mediación] |
| **Estímulo actual:** | [Aquellas acciones que se realizan y que impactan al atributo de calidad. Ej: Acciones definidas donde debe procesarse la petición: recibir petición, realizar transformaciones, ruteo de los servicios, filtros, excepciones] |
| **Artefacto afectado:** | [De acuerdo al estímulo recibido, los artefactos directamente afectados que impactan positivamente o negativamente al atributo de calidad. Ej: primitivas dentro de la mediación.] |
| **Ambiente:** | [Contexto general en el que este atributo se va a cumplir. Ej: sin falla en el hardware, sin falla en comunicaciones, un solo servicio invocado satisfactoriamente.] |
| **Efecto de la acción:** | [Se refiere a la unidad en que se realice la medición del comportamiento del sistema frente a este escenario. Ej: el tiempo en que el sistema procese la petición y genera una respuesta al consumidor del servicio expuesto por la mediación.] |
| **Medición de la respuesta:** | [Valor del efecto de la acción de acuerdo al escenario descrito. Ej: La respuesta no debe exceder los 3 segundos.] |

# 7 VISTA LÓGICA

[Roles interesados: usuarios finales.

Componentes: clases.

Resumen: funcionalidades.

Se busca que esta vista sea única para la solución]

## 7.1 Paquetes

## 7.2 Clases

# 8 VISTA DE DESARROLLO

[Roles interesados: usuarios desarrolladores.

Resumen: administración del software.

Componentes: módulos y capas de la solución]

## 8.1 Componentes

## 8.2 Paquetes

# 9 VISTA FÍSICA DE DESPLIEGUE

[Roles interesados: usuarios infraestructura.

Resumen: topología, comunicaciones.

Componentes: nodos, servidores, plataformas, puertos.

Mapear el software al hardware.

Utilice esta vista sólo si el sistema se distribuye a través de más de un nodo. Incluso en estos casos, sólo utilice la vista de despliegue si la distribución tiene implicaciones arquitectónicas. Por ejemplo, en los casos en que hay un solo servidor y muchos clientes, un punto de vista de despliegue sólo es necesario para delimitar las responsabilidades de los servidores y los clientes como una clase de nodos, no hay necesidad de mostrar todos los nodos cliente si todos tienen la mismas capacidades.]

## 9.2 Deployment (servidores, nodos, ambientes de ejecución, protocolos de comunicación).

# 10 VISTA DE DATOS

[Utilice esta vista sólo si la persistencia es un aspecto importante del sistema y la traducción del modelo de diseño para el modelo de datos no se realiza automáticamente por el mecanismo de persistencia. Por ejemplo un ORM (mapeo objeto relacional).]

# 11 +1 ESCENARIOS - VISTA DE CASOS DE USO

[Roles interesados: todos.

Resumen: secuencia de interacciones entre las capas.]

## 11.1 Excenario #n.

# 

# 

# 12 VERSIÓN DEL DOCUMENTO

## 12.1 Información de la versión <x.x>

|  |  |  |  |
| --- | --- | --- | --- |
| **Elaborado por** | <Nombre> | **Fecha** | <aaaa/mm/dd> |
| **Revisado por** | <Nombre> | **Fecha** | <aaaa/mm/dd> |
| **Aprobado por** | <Nombre> | **Fecha** | <aaaa/mm/dd> |

## 12.2 Registro de cambios del documento

|  |  |  |  |
| --- | --- | --- | --- |
| **¿Cuándo?** | **¿Qué?** | **¿Quién aprobó?** | **¿Para qué?** |
|  |  |  |  |

# 

# 

# VERSIÓN DEL FORMATO

## Información de la versión 2.1

|  |  |  |  |
| --- | --- | --- | --- |
| **Elaborado por** | Carlos Palacio | **Fecha** | 2014/04/15 |
| **Revisado por** | Jorge Quintero | **Fecha** | 2014/04/15 |
| **Aprobado por** | Jorge Quintero | **Fecha** | 2014/04/15 |

## Rationale

|  |  |  |  |
| --- | --- | --- | --- |
| **¿Cuándo?** | **¿Qué?** | **¿Quién aprobó?** | **¿Para qué?** |
| 2013/06/20 | Se genera la primera versión del formato | Gabriel Morris | Para resumir formalmente las funcionalidades básicas, restricciones técnicas y atributos de calidad. |
| 2014/04/15 | Se adaptó la versión 1.0 de acuerdo a la plantilla de formatos del método Pragma | Jorge Quintero | Para seguir los lineamientos de formatos del método Pragma. |
| 2014/0614 | Se hicieron ajustes de en el contenido del documento, se elimina la vista de procesos, teniendo en cuenta que esta se define en Blueworks Live | Jorge Quintero | Para adecuar el contenido del documento a las necesidades del proceso |

1. Modelo de vistas 4+1: Philippe Kruchten: [paper del modelo](https://docs.google.com/a/pragma.com.co/file/d/0Byr_Bgm1eJDkTm8zTF9YMjhweFE/edit?usp=sharing) [↑](#footnote-ref-1)