ki****

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Consecutivo:** | ***01*** | ***Fecha*** | ***22*** | ***05*** | ***2018*** |
| **Nombre del proyecto:** | ***ARQUITECTURA SOA*** | | | | |
| **Ciclo**: | **1** | | | | |
| **Responsable**: | Sebastián Gómez  Mario Silva  Fabián Capera  Víctor Alfonso Duarte Quintero | | | | |
| **Nombre**: |  | | | | |
| **Curso:** | **MODELAMIENTO Y VALIDACION DE ARQUITECTURA** | | | | |

Tabla de Contenido

1. Introducción 3

1.1 Objetivo 3

1.2 Definiciones, Acrónimos y Abreviaturas 3

1.3 Referencias 3

1.4 Alcance 3

2. Contexto 4

2.1 Fundamentos de la solución 4

2.1.1 Fundamentos 4

2.1.2 Decisiones arquitecturales 4

3. Drivers de Arquitectura 4

3.1 Objetivos de la arquitectura de la solución 4

3.1.1 Permitir el acceso a las facturas por medio de servicios por parte de proveedores, proveyendo servicios con alto grado de interoperabilidad (atributo calidad 3.4.1.2). 4

3.1.2 Proveer mecanismos de auditoría con el fin de poder revisar cualquier irregularidad detectada e informada por los usuarios de la plataforma (Decisión de arquitectura DESC\_ARQ\_002 y DESC\_ARQ\_003). 4

3.1.3 Habilitar los servicios requeridos para la integración con los proveedores, basándonos en una arquitectura lo suficientemente escalable (atributo calidad 3.4.1.1). 4

3.2 Requerimientos Funcionales Significativos 4

3.2.1 Validación de Usuario. 4

3.2.2 Selección de servicio a pagar. 4

3.2.3 Selección de medio de pago. 4

3.2.4 Verificación de saldos. 4

3.2.5 Realizar pago. 4

3.2.6 Notificar pago. 4

3.3 Restricciones 5

3.4 Atributos de calidad 5

3.4.1 Descripción de atributos de calidad 5

4. Vistas de arquitectura 6

4.1 Vistas de la arquitectura 6

4.1.1 Vista lógica 6

4.1.2 Vista de desarrollo 7

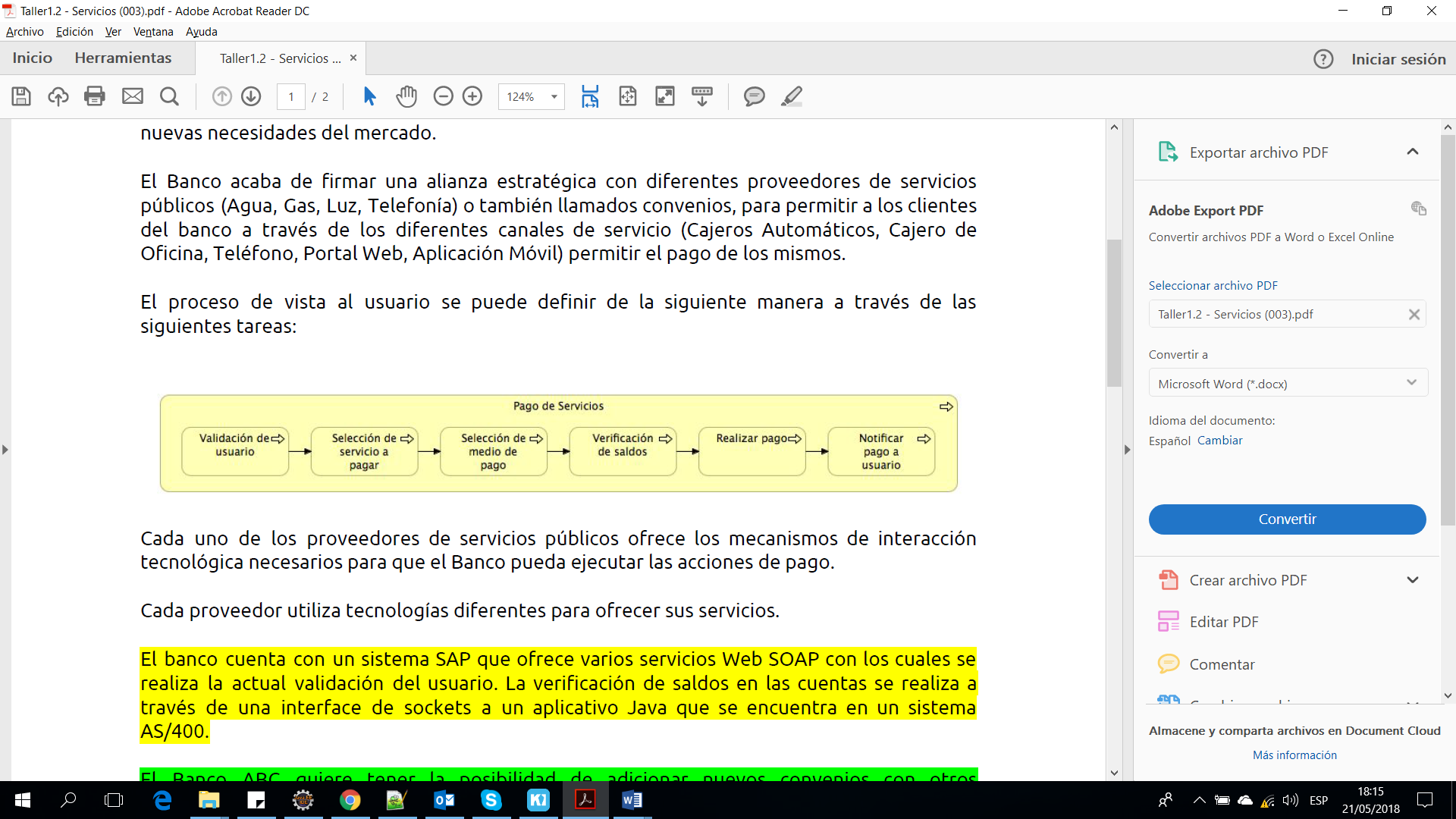
4.1.3 Vista de casos de uso 8

5. Riesgos de la arquitectura 8

# Introducción

El Banco ABC está realizando varios proyectos de actualización tecnológica los cuales le permiten ofrecer sus productos financieros de manera ágil parax responder a nuevas necesidades de mercado.

El Banco acaba de firmar una alianza estratégica con diferentes proveedores de servicios públicos (Agua, Gas, Luz, Telefonía) o también llamados convenios, para permitir a los clientes del banco a través de los diferentes canales de servicio (Cajeros Automáticos, Cajero de Oficina, Teléfono, Portal Web, Aplicación Móvil) permitir el pago de los mismos. El proceso de vista al usuario se puede definir de la siguiente manera a través de las siguientes tareas:



## Objetivo

El objetivo de este proyecto es crear un modelo de la arquitectura SOA (Service Oriented Architecture) planteado para lograr la solución donde se pueda ofrecer mecanismos de interacción tecnológica necesarios para el banco pueda ejecutar las acciones de pago con los servicios expuestos por los proveedores.

## Definiciones, Acrónimos y Abreviaturas

**Gifhub**: Plataforma de desarrollo para alojar proyectos utilizando el sistema de control de versiones.

## Referencias

* + <https://github.com/>*Conectarse a proyectos en github*
  + <https://docs.docker.com/compose/>, *Desplegar y consumir proyectos alojados en Docker*

## Alcance

Se va a crear una arquitectura SOA que permita soportar los pagos de servicio, consultar el saldo a pagar teniendo en cuenta la reusabilidad, interoperabilidad, escalabilidad, flexibilidad, elasticidad y eficiencia.

# Contexto

## Fundamentos de la solución

### Fundamentos

Se hace necesaria la implementación de una arquitectura que provea las funcionalidades suficientes para realizar la integración con proveedores de servicios públicos (Agua, Gas, Luz, telefonía) para permitir a los clientes del banco a través de sus diferentes canales de servicio (Cajeros Automáticos, cajero de oficina, teléfono, Portal Web, Aplicación Móvil) permitir el pago de los mismos.

### Decisiones arquitecturales

|  |  |
| --- | --- |
| IDENTIFICADOR | JUSTIFICACIÓN |
| DEC\_ARQ\_001 | Control de acceso de usuarios por medio de validaciones. |
| DEC\_ARQ\_002 | Patrón arquitectura SOA, bajo la premisa de orquestación de servicios para el consumo de diversos servicios expuestos por terceros. |
| DEC\_ARQ\_003 | Patrón Publisher Subscriber orientado a tópicos bajo el cual se informe a los distintos suscriptores expongan sus servicios. |

# Drivers de Arquitectura

## Objetivos de la arquitectura de la solución

### Permitir el acceso a las facturas por medio de servicios por parte de proveedores, a través de servicios que permitan interoperabilidad. (atributo calidad 3.4.1.2).

### Proveer mecanismos de auditoría con el fin de poder revisar cualquier irregularidad detectada e informada por los usuarios de la plataforma (Decisión de arquitectura DESC\_ARQ\_002 y DESC\_ARQ\_003).

### Habilitar los servicios requeridos para la integración con los proveedores, basándonos en una arquitectura lo suficientemente escalable (atributo calidad 3.4.1.1).

## Requerimientos Funcionales Significativos

### Validación de Usuario.

### Selección de servicio a pagar.

### Selección de medio de pago.

### Verificación de saldos.

### Realizar pago.

### Notificar pago.

## Restricciones

|  |  |  |
| --- | --- | --- |
| TIPO RESTRICCION | ID\_RESTRICCION | DESCRIPCION |
| NEGOCIO | RN001 | Los cambios que ocurran en las transacciones de negocio deberán reflejarse en tiempo real. |
| TÉCNICAS | RT001 | Se requiere que el número de intercambios de información entre los participantes sea el menor posible. |
| ARQUITECTURAL | RA001 | Integración con plataformas de proveedores por medio de configuración flexible (Realizar cambios sin necesidad de reiniciar el sistema). |

## Atributos de calidad

### Descripción de atributos de calidad

#### Escalabilidad

Se requiere que la aplicación pueda tener una escalabilidad vertical de cada uno de los servicios, siendo lo suficientemente adecuado a nivel de infraestructura como para soportar el crecimiento esperado de usuarios y transacciones en línea sin afectar el rendimiento o performance de la misma (atributo calidad 3.4.1.5).

#### Interoperabilidad

Es requerido que los servicios tengan la posibilidad de integrarse a los distintos proveedores indiferente de su protocolo de comunicación SOAP o Rest (restricción RA001).

#### Reusabilidad

Al tener una arquitectura SOA se tienen servicios que pueden ser utilizados para composiciones de otros servicios lo cual genera una reusabilidad dada por la arquitectura implementada.

Se espera que la cantidad de líneas de código sea reutilizable en un mínimo de 20%.

Este atributo se encuentra altamente ligado con otro atributo de calidad el cual es la mantenibilidad entre más reutilizable sea el código si llega a encontrarse un defecto en alguna parte, este será más fácil de modificar debido a que solo se deberá resolver en un punto central y no en varios puntos.

#### Flexibilidad

Se debe asignar recursos a sus propios identificadores (URI). Debido a que cada verbo tiene un significado independiente o especifico como (GET, POST, PUT Y DELETE) evitando ambigüedades.

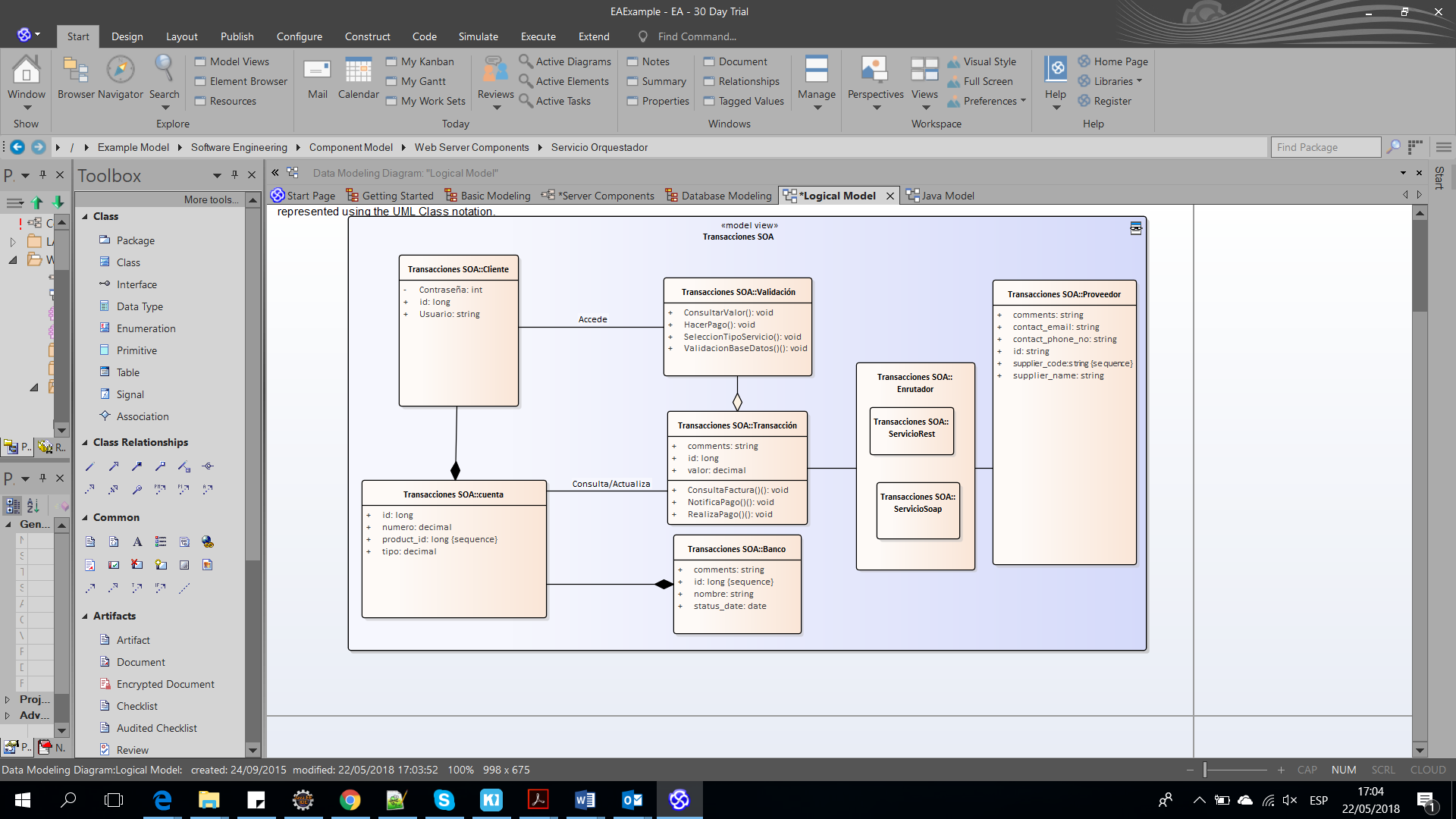
#### Eficiencia

Se requiere reducir costos con el tiempo al no tener un procesamiento intensivo utilizando formatos de mensaje más pequeños que XML.

# Vistas de arquitectura

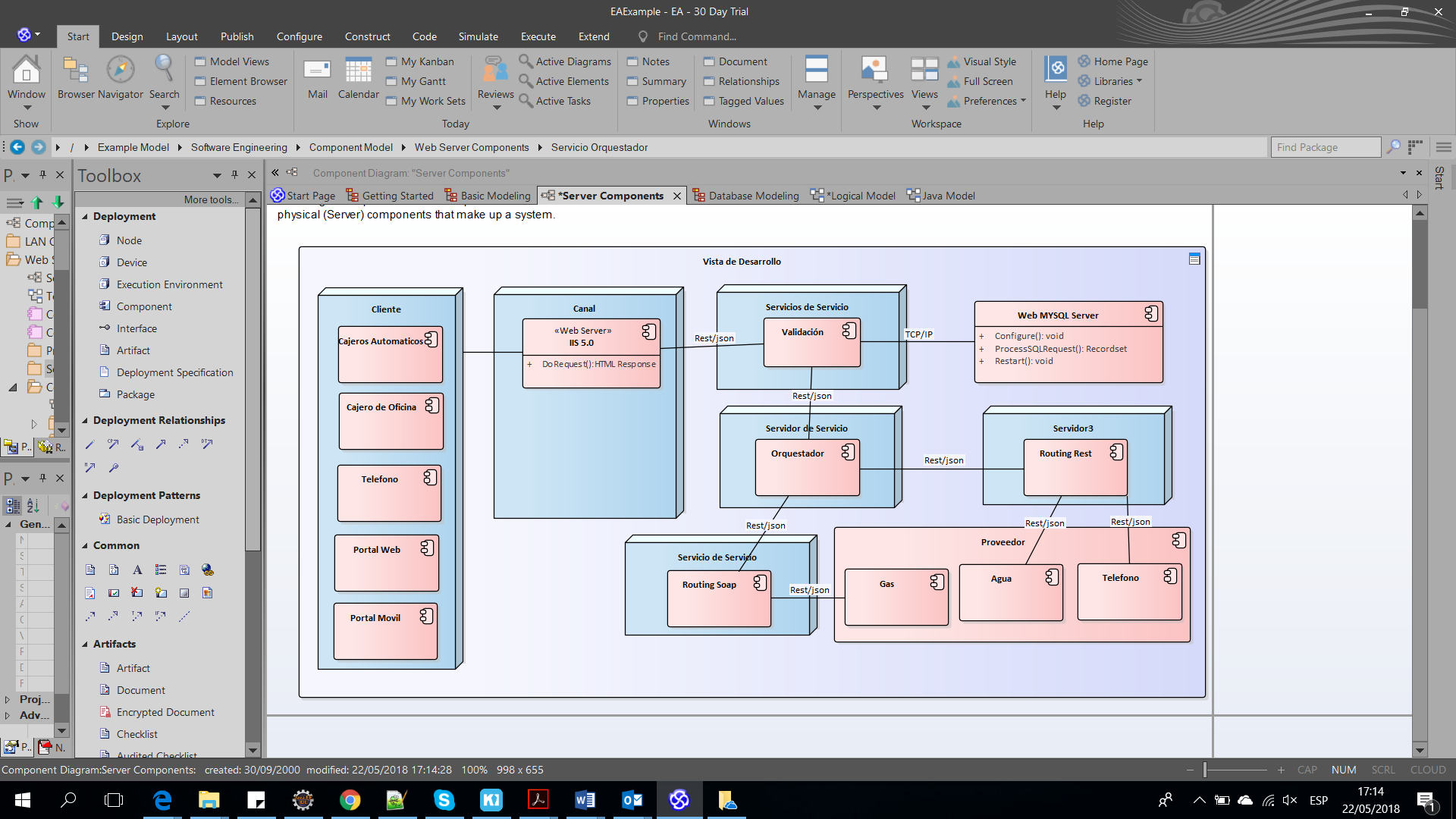
## Vistas de la arquitectura

### Vista lógica



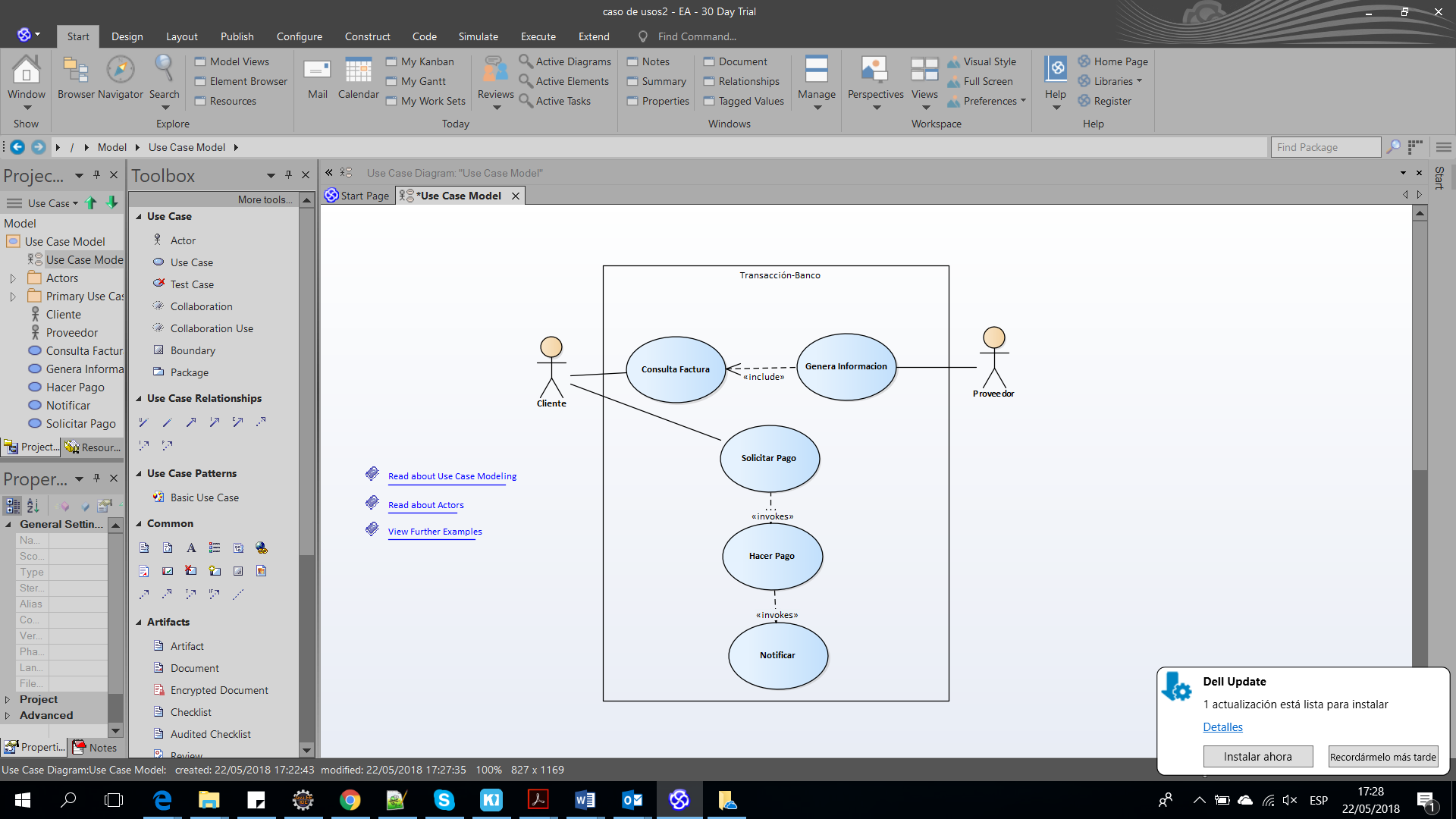
**Figura 1.** En este diagrama se pueden observar las diferentes clases que se van a implementar en el sistema, dentro de estas se encuentran: Cliente, Canal, Validación con base de datos, el orquestador que es el encargado de hacer las transacciones, Enrutador es donde se define servicios para contratos Rest y Soap y los proveedores. Este contempla la decisión de arquitectura DEC\_ARQ\_001.

### Vista de desarrollo



**Figura 2.** En este diagrama se pueden observar los diferentes componentes que se van a implementar. Este contempla las siguientes decisiones de arquitectura: DEC\_ARQ\_004, DEC\_ARQ-005 y DEC\_ARQ\_007.

### Vista de casos de uso



**Figura 5.** En este diagrama se visualiza el caso de uso principal el cual está vinculado a la realización de una cotización.

# Riesgos de la arquitectura

|  |  |  |  |
| --- | --- | --- | --- |
| RIESGOS | ESTRATEGIA DE MITIGACIÓN | PLAN DE CONTINGENCIA | IMPACTO |
| Debido a que el tipo de servicio de composición el cual no puede seguir con el proceso una vez se llegase a encontrar un problema ya que es síncrono. | Se cuenta con un sistema Registro de logs de auditoría en la mensajería y la aplicación. Sobre el cual se realiza un constante monitoreo de todos los servicios | Si se identifica algún tipo de irregularidad sobre alguno de los sistemas, será inmediata mente reportado al área de soporte, que cuenta con una disponibilidad de 7\*24 para la atención de estos casos | El impacto en caso de materialización del riesgo es de nivel medio, ya que a pesar de la indisponibilidad del sistema se cuentan con protocolos agiles para la activación del sistema de respaldo. |
| Aumento en los tiempos de atención debido al aumento de peticiones y transacciones por minuto | Se realiza un monitoreo sobre el número de las transacciones y cantidad de peticiones diario, donde también se tiene un umbral definido para generar una alerta que contempla una escalabilidad vertical | Cuando las transacciones y peticiones lleguen a un nivel máximo del 80%, se va a efectuar una escalabilidad vertical. | El impacto es de nivel medio debido a que, aunque aumente el tiempo de respuesta de la aplicación frente a los usuarios la disponibilidad no se va a ver afectada. |