**Documento de Arquitectura de Software**

**Sistema de Cuentas Bancarias (SN)**

Este documento incluye una descripción de la Arquitectura de Software para el Sistema Sistemas de Cuentas Bancarias de la empresa TuB@nco S.A.C.

**2JAV Software Consulting SAC**

Gerencia de Desarrollo

**Historial de Revisiones**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Versión** | **Descripción** | **Autor** |
| 17/07/2019 | 1.0 | Primera versión del documento. | Munares A. |
| 18/07/2019 | 2.0 | Segunda versión del documento. | Magallanes J. |
| 18/07/2019 | 2.01 | Revisión de componentes | Ramirez R. |
| 18/07/2019 | 2.02 | Revisión de diagramas | Espinoza V. |
| 19/07/2019 | 2.02 | Aprobación | Bautista E. |

**Tabla de Contenido**

[1. Introducción 6](#_Toc14472769)

[1.1 Propósito 6](#_Toc14472770)

[1.2 Alcance 6](#_Toc14472771)

[1.3 Público objetivo 6](#_Toc14472772)

[2. Contexto 6](#_Toc14472773)

[2.1 Objetivo general 6](#_Toc14472774)

[2.2 Objetivos específicos 6](#_Toc14472775)

[2.3 Restricciones 6](#_Toc14472776)

[2.4 Suposiciones 6](#_Toc14472777)

[3. Vista de casos de uso 6](#_Toc14472778)

[4. Atributos de calidad 7](#_Toc14472779)

[4.1 Descripción de escenarios 8](#_Toc14472780)

[4.2 Evaluación de las Tácticas. 8](#_Toc14472781)

[4.3 Obtención de los Constrains y Preocupaciones. 8](#_Toc14472782)

[4.4 Resultados de Arquitectura a implementar. 8](#_Toc14472783)

[5. C4 Model 8](#_Toc14472784)

[5.1 Vista de Contexto 8](#_Toc14472785)

[5.2 Vista de componente 9](#_Toc14472786)

[6. Vista lógica 10](#_Toc14472787)

[6.1 Diagrama de clases 10](#_Toc14472788)

[7. Vista física 10](#_Toc14472789)

[7.1 Infraestructura tecnológica 10](#_Toc14472790)

[7.2 Diagrama de implantación o despliegue 10](#_Toc14472791)

[8. Vista de desarrollo 10](#_Toc14472792)

[8.1 Lenguaje de programación 10](#_Toc14472793)

[8.2 Herramientas 10](#_Toc14472794)

[8.3 Diagrama de integración y despliegue continuo 11](#_Toc14472795)

[9. Vista de datos 11](#_Toc14472796)

[10. Validaciones adicionales de seguridad 11](#_Toc14472797)

[11. Mejores prácticas de desarrollo 11](#_Toc14472798)

[11.1 Principios de diseño y arquitectura 11](#_Toc14472799)

[11.2 Codificación 11](#_Toc14472800)

[12. Referencias 11](#_Toc14472801)

**Sistema de Cuentas Bancarias**

# Introducción

## Propósito

El presente documento tiene como propósito el afianzar la arquitectura de software existente del Banco, para tener precedentes de las decisiones que se han tomado y par que sirva como fuente de referencia para nuevas arquitecturas.

## Alcance

El documento contiene el análisis documentario de los diferentes componentes del sistema de cuentas bancarias, la aplicación web y otros, incluyendo los atributos de calidad que se deberían tener para cumplir los objetivos.

## Público objetivo

Está enfocado para todo el equipo de proyecto y de TI, la vista de contexto para la gerencia quien también debe conocer sobre los alcances de la arquitectura incorporada.

# Contexto

## Objetivo general

Brindar toda la fuente necesaria del trabajo realizado para la implementación de la arquitectura para el sistema de cuentas bancarias, sus diagramas y referencias.

## Objetivos específicos

## Restricciones

* Solo cambiar la documentación cuando el cambio afecte la arquitectura.

## Suposiciones

* Que el lector tenga conocimiento base de documentación de arquitectura de software.
* Que lo detallado hasta el presente se actualice conforme cambios del sistema.

# Vista de casos de uso

|  |  |
| --- | --- |
| **ID** | **Descripción** |
| CU-1 Acceso al Sistema | Para acceder al sistema debe ser por usuario y clave |
| CU-2 Registrar de acciones en el sistema | Todas las acciones de los usuarios deben registrase. |

# Atributos de calidad

|  |  |  |
| --- | --- | --- |
| **ID** | **Atributo** | **Descripción** |
| QA-1 | Seguridad | El usuario debe ingresar correctamente su usuario y clave |
| QA-2 | Seguridad | Al acceder al sistema todas las acciones del usuario deben registrarse en el archivo log. |
| QA-3 | Seguridad | La información manejada por el sistema está protegida de acceso no autorizado, divulgación, evitando la corrupción y estados inconsistentes |
| QA-4 | Disponibilidad | Se garantiza el acceso a información integrada, óptima y certera, que permite a los usuarios estar informados del estado de sus movimientos bancarios en tiempo real. |

# Descripción de escenarios

* **DISPONIBILIDAD**
* **DIS001:** El sistema debe estar disponible como mínimo un 99% durante el año.



* **DIS002:** El envío de la data estar disponible como mínimo en un 98% desde realizada la transacción por el usuario.
* 

**SEGURIDAD**

**SEG001: Acceso al sistema**



* **SEG002: Autenticación de usuarios**



# Evaluación de las Tácticas.

**DISPONIBILIDAD**

|  |  |
| --- | --- |
| **TÁCTICA** | **DESCRIPCIÓN** |
| Detección de fallas | |
| Monitoreo | Realizar el seguimiento del funcionamiento de los sensores, servidor web y base de datos. |
| ExceptionDetection | * Para el manejo de excepciones en el software se realizará a través de mensajes de alerta al usuario y registro de fallas en el log. * Para el manejo de excepciones en el hardware se utilizará el visor de errores del sistema de operativo. |

|  |  |
| --- | --- |
| **TÁCTICA** | **DESCRIPCIÓN** |
| Recuperación de fallas | |
| Redundancia | Clusterizar los servidores con un balanceador que asegure de alta disponibilidad sobre los servidores base de datos y el servidor de aplicaciones. |
| Rollback | Realizar los backup incremental diario a las 05:00 horas. |

|  |  |
| --- | --- |
| **TÁCTICA** | **DESCRIPCIÓN** |
| Prevención de fallas | |
| Transacciones | La base de datos MySql se encargará del manejo de las transacciones para la base de datos. |
| Prevención de excepciones | Envío automático de alertas al correo del administrador de red. |

**DISPONIBILIDAD : CHECKLIST**

|  |  |  |
| --- | --- | --- |
| N° | Categoría | DESCRIPCIÓN |
| 01 | Asignación de  Responsabilidades | * Registro de fallas * Notificación al administrador de sistema * Temporalmente fuera de servicio. * Arreglar la falla |
| 02 | Modelo de datos | BD MySql (Mirror) |
| 03 | Tecnología | * Cluster * Backup * Redudancia |

**SEGURIDAD**

|  |  |
| --- | --- |
| **TÁCTICA** | **DESCRIPCIÓN** |
| Resistencia Ataques | |
| Autentificación de usuarios | La autenticación se hace a nivel del servidor, los usuarios se autentican utilizando el mecanismo de autenticación basada en token. |
| Autorización de usuarios | Mediante la definición de los roles y permisos de acceso. |
| Limitar accesos | El servidor permite solo 2 sesiones en paralelo de un usuario |

|  |  |
| --- | --- |
| **TÁCTICA** | **DESCRIPCIÓN** |
| Recuperación de un Ataque | |
| Auditoría | Utilización de log de auditoria para la detección de cualquier eventualidad en el mantenimiento de los registros del sistema. |

**SEGURIDAD : CHECKLIST**

|  |  |  |
| --- | --- | --- |
| N° | Categoría | DESCRIPCIÓN |
| 01 | Asignación de  Responsabilidades | Módulo de seguridad |
| 02 | Modelo de Coordinación | Bidireccional entre el servidor de aplicaciones y el servidor web |
| 03 | Administración de Recursos | Módulo de seguridad:  - Habilitar el acceso de usuarios de acuerdo al perfil requerido.  - Notificar el acceso a los usuarios.  - Generar informe de auditoría de accesos.  - Generar informe de seguridad. |

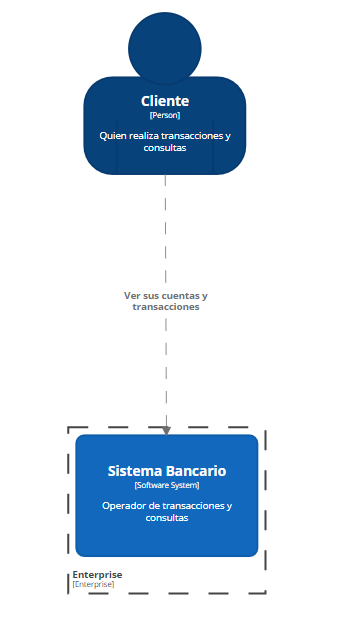
# Obtención de los Constrains y Preocupaciones.

|  |  |
| --- | --- |
| **ID** | **Restricción** |
| CON-1 | El personal solo puede ingresar al sistema y acceder a su información personal. |
| CON-2 | Los módulos a desarrollar deben ser accesible desde cualquier navegador web. |

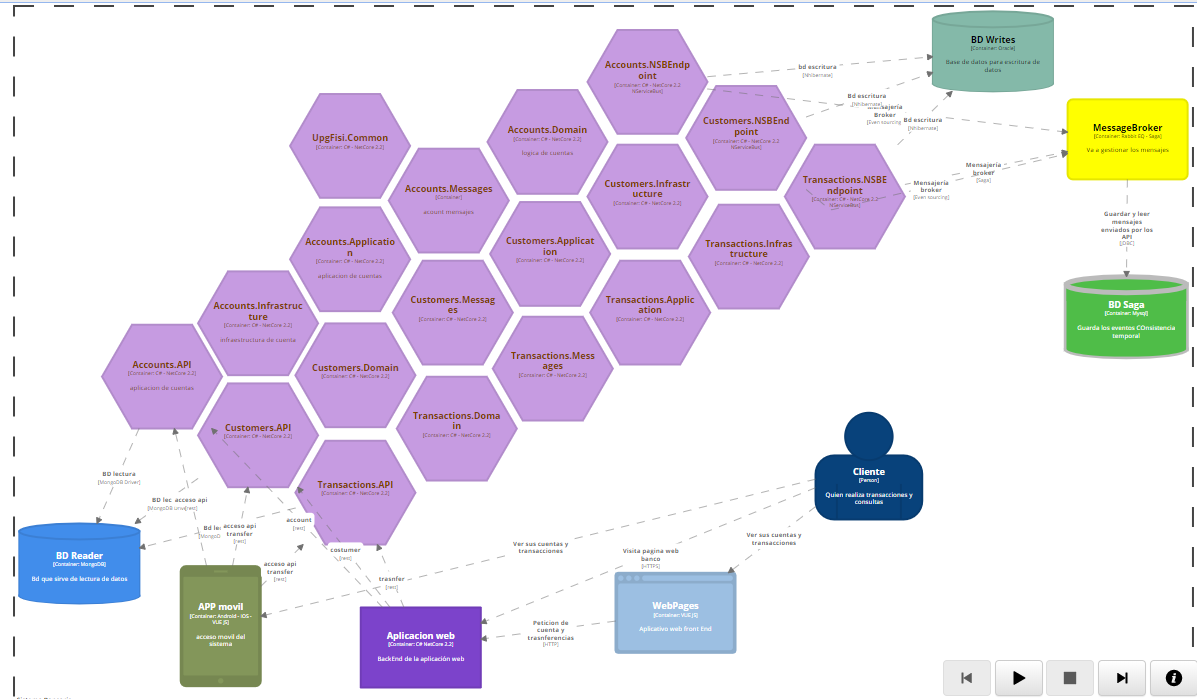
|  |  |
| --- | --- |
| **ID** | **Preocupaciones** |
| CR-1 | El sistema debe ser desarrollado en Net Core 2.2 |
| CR-2 | La base de datos debe ser MySql. |

# C4 Model

# Vista de Contexto



# Vista de componente



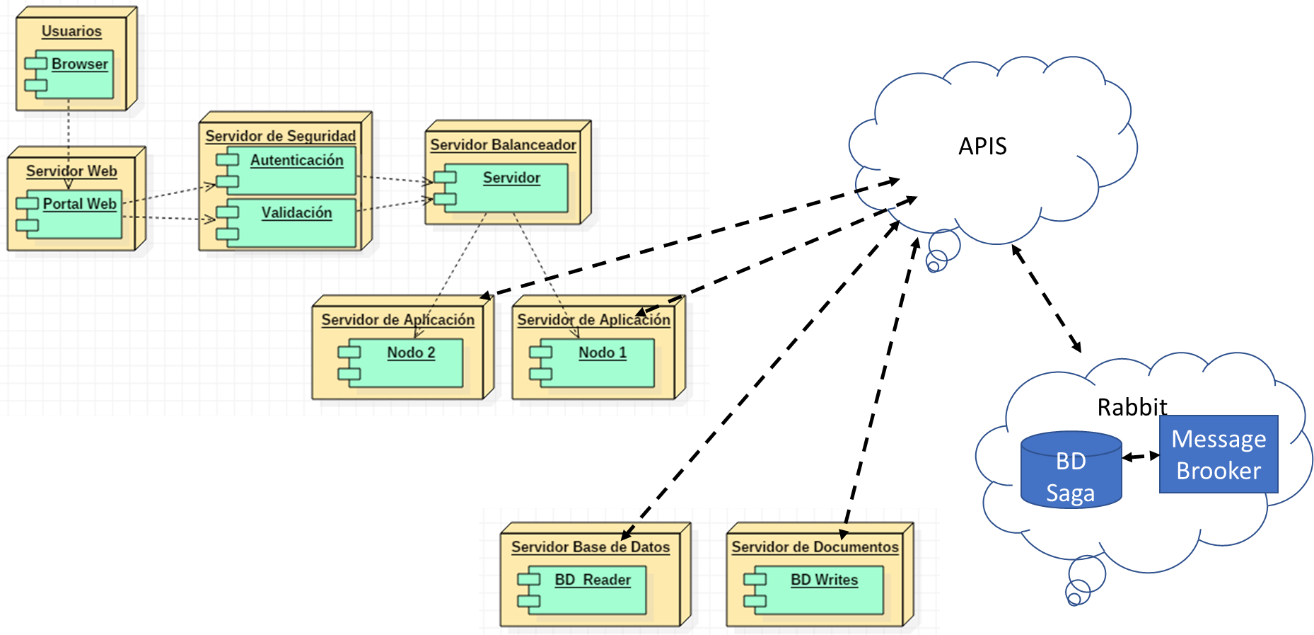
# Vista lógica

## Diagrama de clases

|  |  |
| --- | --- |
| **Clase** | **Descripción** |
|  |  |

# Vista física

## Infraestructura tecnológica



## Diagrama de implantación o despliegue

|  |  |
| --- | --- |
| **Componente** | **Descripción** |
| Servidor aplicaciones |  |
| Servidor de seguridad |  |
| Microservicios |  |
| Componente Cloud |  |

# Vista de desarrollo

## Lenguaje de programación

El software está desarrollado en C#.

## Herramientas

|  |  |  |
| --- | --- | --- |
| **Tipo** | **Herramienta** | **Compra de Licencia** |
| Manejador de mensajería | NServiceBus | Si |
| Message Broker | Rabbit EQ | Si |
| Base de datos | Mysql | Gratuito |
| Comunicación microservicios | Rest | Gratuito |
| Framework front end | VUE JS | Gratuito |

## Diagrama de integración y despliegue continuo

|  |  |
| --- | --- |
| **Servidor** | **Descripción** |
|  |  |
|  |  |
|  |  |

# Vista de datos

# Validaciones adicionales de seguridad

# Mejores prácticas de desarrollo

## Principios de diseño y arquitectura

|  |  |
| --- | --- |
| **Principio** | **Descripción** |
| DDD | Arquitectura diseñada por el dominio, que permite separar la lógica del negocio de los demás componentes del sistema. |
| Even Sourcing | Nos permite manejar de manera asíncrona nuestras APIS para que funcionen por independiente sin acoplamiento. |
| Saga | Patrón que nos permiten guardar en cola la mensajería para que tengamos una eventual consistencia. |
| Outbox | Para registrar las consultas y otros de la BD que realizan las APIS |

## Codificación

# Referencias