Microservicio MsAdaptadorBCP

Documento Versión: 1.0

Gestión de Seguridad Electrónica

Fecha generación: **30/11/2025**

CONTROL DE VERSIÓN

VERSIÓN	FECHA DE ACTUALIZACIÓN	CARGO	DESCRIPCIÓN
1.0	03/11/2025	Arquitecto de Software	Versión inicial del documento
1.1	29/11/2025	Control de Arquitectura	Corrección del documento basado en observaciones
1.2	30/11/2025	Control de Arquitectura	Revisión estructura de objetos JSON

Tabla de contenido

1 CONTEXTO

El Registro Nacional de Identificación y Estado Civil (RENIEC) es el organismo técnico autónomo encargado de la identificación de los peruanos, otorgar el Documento Nacional de Identidad (DNI) y registrar los hechos vitales. En el marco de la modernización y transformación digital del Estado peruano, RENIEC ha desarrollado el DNI Electrónico (DNIe), un documento de identidad que incorpora tecnología de chip y biometría, permitiendo la autenticación electrónica de ciudadanos y facilitando servicios digitales seguros.

2 INTRODUCCIÓN

Este documento describe el **catálogo de microservicios** identificados para la solución de Personalización del DNIe de RENIEC. El objetivo es establecer una arquitectura técnica moderna, escalable y mantenible que reemplace o complemente los sistemas monolíticos actuales mediante una transición ordenada hacia una arquitectura orientada a microservicios.

2.1 Justificación de la Arquitectura de Microservicios

La adopción de microservicios para esta solución responde a necesidades técnicas y operativas concretas:

Escalabilidad Independiente: Componentes con cargas diferenciadas pueden escalar de forma independiente según demanda real, optimizando recursos de infraestructura.

Resiliencia y Tolerancia a Fallos: El fallo de un microservicio no compromete servicios críticos. Los patrones Circuit Breaker y Retry garantizan continuidad operativa.

Agilidad en el Desarrollo: Equipos autónomos pueden desarrollar, probar y desplegar servicios de manera independiente, reduciendo tiempos de entrega.

Mantenibilidad y Evolución Tecnológica: Cada servicio puede evolucionar tecnológicamente sin afectar al ecosistema completo.

Trazabilidad y Observabilidad: Arquitectura distribuida permite implementar logging centralizado, distributed tracing y métricas granulares.

2.2 Arquitectura de Referencia

La solución se estructura en tres capas principales:

Capa de Exposición (API Management Layer): API Manager como punto único de entrada con gestión centralizada de seguridad, throttling y versionado.

Capa de Representación (Microservices Layer): Microservicios de negocio con lógica específica de dominio y responsabilidad única.

Capa de Integración (Integration Layer): Event Streaming para comunicación asíncrona y conectores a sistemas legados.

2.3 Códigos de Respuesta HTTP Estándar

Todos los microservicios implementan un conjunto estandarizado de códigos de respuesta HTTP para garantizar consistencia:

Código	Descripción
200	OK - Operación completada exitosamente
201	Created - Recurso creado exitosamente

400	Bad Request - Parámetros inválidos o datos incompletos
401	Unauthorized - Token JWT inválido, expirado o ausente
403	Forbidden - Sin permisos suficientes para ejecutar la operación
404	Not Found - Recurso no encontrado en el sistema
408	Request Timeout - Tiempo de espera agotado al conectar
409	Conflict - Conflicto con el estado actual del recurso
422	Unprocessable Entity - Datos válidos, pero no procesables por reglas de negocio
429	Too Many Requests - Límite de rate limit excedido
500	Internal Server Error - Error interno del servicio (información generalizada al exterior, detalle en logs)
502	Bad Gateway - Servicio externo no disponible o respuesta inválida
503	Service Unavailable - Servicio temporalmente no disponible o Circuit Breaker abierto
504	Gateway Timeout - Servicio externo no respondió en tiempo esperado

3 CATÁLOGO DE MICROSERVICIOS

3.1 Microservicio MsAdaptadorBCP

El Microservicio MsAdaptadorBCP es el componente adaptador que facilita la integración con la API del Banco de Crédito del Perú (BCP) para la validación de pagos realizados por los ciudadanos en los trámites de RENIEC. Este microservicio actúa como un intermediario que traduce las solicitudes del sistema SIIRC al formato requerido por la API del BCP, y transforma las respuestas del banco a un formato estandarizado que puede ser procesado por el sistema.

3.1.1 Endpoint: ValidarPagoBCP

Valida un pago realizado en el BCP consultando la API del banco con el código de operación proporcionado.

Atributo	Valor
Path	/api/v1/pago/MsAdaptadorBCP/validar-bcp
API Gateway	Interno
Método HTTP	POST
Protocolo	REST/HTTP
Headers	Authorization String (Bearer token JWT para autenticación) X-Correlation-ID UUID (ID de correlación para trazabilidad distribuida) X-Office-Code String (Código de oficina RENIEC, formato: ORG-LIMA-CENTRO) X-Request-Source String (Origen de la solicitud: CoreService, EvaluacionService, etc.) X-User-ID String (Identificador del usuario que realiza la consulta)

Entrada	{
	"codigoOperacion": "string",
	"numeroSolicitud": "string",
	"montoCobrar": "decimal",
	"moneda": "string",
	"fechaOperacion": "string",
	"tipoTramite": "string",
	"canalPago": "string",
	"metadata": {

	"dni": "string", "nombreCompleto": "string", "codigoOficina": "string", "usuarioRegistrador": "string", "numeroTarjeta": "string" } }

Respuesta	{
	"validacionId": "string",
	"codigoOperacion": "string",
	"numeroSolicitud": "string",
	"esValido": "boolean",
	"estadoTransaccion": "string",
	"detalleTransaccion": {
	"montoPagado": "decimal",
	"moneda": "string",
	"fechaPago": "string",
	"horaPago": "string",
	"canalPago": "string",
	"codigoAgencia": "string",
	"nombreAgencia": "string",
	"numeroTarjeta": "string",
	"tipoTarjeta": "string",
	"codigoAutorizacion": "string",
	"numeroReferencia": "string",
	"comercio": "string"
	},
	"coincidencias": {
	"montoCoincide": "boolean",
	"diferenciaMonto": "decimal",
	"fechaCoincide": "boolean",
	"diasDiferencia": "integer"
	},
	"auditoria": {
	"timestampConsulta": "string",
	"tiempoRespuestaMs": "integer",
	"ipOrigen": "string",
	"usuarioConsulta": "string",
	"intentosRealizados": "integer"
	},
	"seguridad": {
	"scoreAntifraude": "integer",
	"riesgo": "string"
	}
	}

Error Response	"error": {
	"tipo": "string",
	"titulo": "string",
	"estado": "integer",
	"errores": [
	{
	"detalleError": "string"
	}
	]
	}

3.1.1.1 Parámetros de Entrada

Dato	Atributo	Tipo	Obligatorio	Longitud Mínima	Longitud Máxima
Código Operacion	codigoOperacion	String	Sí	8	25
Numero Solicitud	numeroSolicitud	String	Sí	1	50
Monto Cobrar	montoCobrar	Decimal	Sí	-	-
Moneda	moneda	String	Sí	3	3
Fecha Operacion	fechaOperacion	String	No	-	-
Tipo Tramite	tipoTramite	String	Sí	1	100
Canal Pago	canalPago	String	No	1	50
Metadata	metadata	Object	No	-	-
Dni	metadata.dni	String	No	8	8
Nombre Completo	metadata.nombreCompleto	String	No	1	200
Código Oficina	metadata.codigoOficina	String	No	1	50
Usuario Registrador	metadata.usuarioRegistrador	String	No	1	100
Numero Tarjeta	metadata.numeroTarjeta	String	No	4	4

3.1.1.2 Parámetros de Respuesta

Nombre	Tipo	Obligatorio	Descripción
validacionId	String (UUID)	Sí	Identificador único de la validación
codigoOperacion	String	Sí	Código de operación consultado
numeroSolicitud	String	Sí	Número de solicitud asociada
esValido	Boolean	Sí	Indica si el pago es válido
estadoTransaccion	String	Sí	Estado de la transacción
detalleTransaccion	Object	Condicional	Detalle completo de la transacción
detalleTransaccion.montoPagado	Decimal	Sí	Monto efectivamente pagado
detalleTransaccion.moneda	String	Sí	Moneda de la transacción
detalleTransaccion.fechaPago	String	Sí	Fecha del pago
detalleTransaccion.horaPago	String	Sí	Hora del pago
detalleTransaccion.canalPago	String	Sí	Canal utilizado
detalleTransaccion.codigoAgencia	String	No	Código de agencia bancaria
detalleTransaccion.nombreAgencia	String	No	Nombre de agencia bancaria
detalleTransaccion.numeroTarjeta	String	No	Últimos 4 dígitos de la tarjeta
detalleTransaccion.tipoTarjeta	String	No	Tipo tarjeta

detalleTransaccion.codigoAutorizacion	String	No	Código de autorización bancaria
detalleTransaccion.numeroReferencia	String	No	Número de referencia único de BCP
detalleTransaccion.comercio	String	No	Nombre del comercio (RENIEC)
coincidencias	Object	Sí	Análisis de coincidencias
coincidencias.montoCoincide	Boolean	Sí	Indica si el monto pagado coincide con el esperado
coincidencias.diferenciaMonto	Decimal	Sí	Diferencia entre monto pagado y esperado
coincidencias.fechaCoincide	Boolean	Condicional	Indica si la fecha coincide
coincidencias.diasDiferencia	Integer	Condicional	Días de diferencia con fecha esperada
auditoria	Object	Sí	Información de auditoría
auditoria.timestampConsulta	String (DateTime)	Sí	Timestamp de la consulta (ISO 8601)
auditoria.tiempoRespuestaMs	Integer	Sí	Tiempo de respuesta en milisegundos
auditoria.ipOrigen	String	Sí	IP desde donde se realizó la consulta
auditoria.usuarioConsulta	String	Sí	Usuario que realizó la consulta
auditoria.intentosRealizados	Integer	Sí	Número de intentos realizados
seguridad	Object	Sí	Información de seguridad y antifraude
seguridad.scoreAntifraude	Integer	Sí	Score de antifraude del BCP
seguridad.riesgo	String	Sí	Nivel de riesgo
seguridad.factoresRiesgo	Array[String]	Sí	Lista de factores de riesgo detectados
error	Object	No	Objeto qué especifica algún error existente en la operación.
error.tipo	String	No	Tipo de error
error.titulo	String	No	Título del error
error.status	integer	No	Número del estado de error
error.errores	Array	No	Listado de errores
error.errores[].detalleError	String	No	Detalle del error generado

3.1.1.3 Valores para el Atributo statusCode

Código	Respuesta	Descripción
200	OK	Validación realizada exitosamente
400	Bad Request	Código de operación inválido o parámetros incorrectos
401	Unauthorized	Token JWT inválido, expirado o ausente
403	Forbidden	Sin permisos para validar pagos o token BCP expirado
404	Not Found	Código de operación no encontrado en el BCP
408	Request Timeout	Tiempo de espera agotado al consultar el banco
409	Conflict	Transacción reversada o en disputa
422	Unprocessable Entity	Datos válidos, pero inconsistencias detectadas (monto no coincide, riesgo alto)
500	Internal Server Error	Error interno del adaptador
502	Bad Gateway	API del BCP no disponible o respuesta inválida
503	Service Unavailable	Servicio temporalmente no disponible (Circuit Breaker abierto)
504	Gateway Timeout	API del BCP no respondió en tiempo esperado