

Patrones de Arquitectura de Software - parcial

Camilo Andrés Romero López, Luis Alejandro Mesa, Juan Pablo Guerrero, Alejandro López

29 de mayo de 2024

Índice

1. Introducción	5
2. Alcance	5
2.1. Funciones y Características Incluidas	5
2.2. Exclusiones y Fuera de Alcance	6
3. Supuestos	7
3.1. Banco ABC	7
3.2. Convenios / proveedores	7
3.3. Servicios públicos	7
3.4. Hogares	8
3.5. Usuarios del Banco ABC	10
3.6. Ciclos de facturación	10
3.7. Transacciones / Solicitudes	12
3.8. TPS	12
4. Atributos	13
4.1. Escalabilidad	13
4.1.1. Escenario	13
4.1.2. Estímulo	14
4.1.3. Ambiente	14
4.1.4. Respuesta	14
4.1.5. Medición	14
4.1.6. Atributo de Calidad	14
4.1.7. Atributos no cualitativos:	14
4.2. Modificabilidad	15
4.2.1. Escenario	15
4.2.2. Estímulo	15
4.2.3. Ambiente	15
4.2.4. Respuesta	15
4.2.5. Medición	15
4.2.6. Atributo de Calidad	15
5. ADR	15
5.1. Escalabilidad	15
5.1.1. Escenario	15
5.1.2. Fecha	15
5.1.3. Autor	15
5.1.4. Resumen	15
5.1.5. Decisiones	16
5.2. Modificabilidad	18
5.2.1. Escenario	18
5.2.2. Fecha	18
5.2.3. Autor	18
5.2.4. Resumen	18
5.2.5. Decisiones	18
6. 4 + 1	19
6.1. Escenario de escalabilidad	19
6.1.1. Vista lógica	19
6.1.2. Vista física	19
6.1.3. Vista de procesos	20
6.1.4. Vista de desarrollo	21
6.2. Escenario de Modificabilidad	24
6.2.1. Vista lógica	24

6.2.2. Vista física	24
6.2.3. Vista de procesos	24
6.2.4. Vista de desarrollo	24
7. Trade Off	27
8. Patrones utilizados	27
9. Idempotencia y garantías de mensajería	27
9.1. Servicio de identificación de usuario	27
9.2. Servicio consulta de información de pago	27
9.3. Servicio de hacer una orden de pago	27
10. Alternativas de arquitectura	27
10.1. Arquitectura usando estilo <i>Serverless</i>	27
10.1.1. Vista lógica funcional	27
10.1.2. Vista de procesos funcional	27
10.1.3. Vista física funcional	28
10.1.4. Vista de desarrollo funcional	28
10.1.5. Casos de uso	28
10.1.6. ADR	28

Índice de figuras

1. Disponibilidad de productos financieros, 2019	8
2. Instrumentos utilizados en los pagos habituales mensuales (Porcentaje de la población)	9
3. Tenencia de medios e instrumentos de pago	9
4. Las plataformas digitales cada vez ganan más espacio en el sistema financiero	10
5. Días disponibles para pago: Ene 2023 - Abr 2024 (Gas natural)	11
6. Días disponibles para pago: Ene 2023 - Abr 2024 (Energía eléctrica)	11
7. Días disponibles para pago: Ene 2023 - Abr 2024 (Acueducto)	11
8. Distribución de transacciones durante el ciclo de facturación	13
9. Vista Lógica - escenario de escalabilidad.	21
10. Vista Física - escenario de escalabilidad.	22
11. Vista de procesos. - escenario de escalabilidad	23
12. Vista de desarrollo. - escenario de escalabilidad	23
13. Vista Lógica - escenario de modificabilidad.	24
14. Vista Física - escenario de modificabilidad.	25
15. Vista de procesos. - escenario de modificabilidad	25
16. Vista de desarrollo. - escenario de modificabilidad	26
17. Vista Lógica Funcional.	29
18. Vista de procesos funcional, identificación de usuario.	30
19. Vista de procesos funcional, pago de servicios públicos.	30
20. Vista de procesos funcional, registro de convenio / proveedor.	30
21. Vista física funcional.	31
22. Vista de desarrollo funcional - identificación de usuario.	32
23. Vista de desarrollo funcional - pago.	33
24. Vista de desarrollo funcional - registro de proveedores.	33
25. Vista de escenarios funcional.	34

Índice de cuadros

1. Hogares, personas y promedio de personas por hogar Total nacional y área 2023.	7
2. Encuesta Nacional de Calidad de Vida (ECV) 2020.	7
3. Volumen de transacciones al mes, por instrumento de pago utilizado en 2019.	8

4.	Distribución de transacciones durante ciclo de facturación y transacciones promedio por día.	12
5.	Punto de cambio - Durante.	18
6.	Punto de cambio - Quién.	19
7.	Punto de cambio - Patrón.	19
8.	Servicios de Amazon utilizados.	20
9.	Estimado de tiempos y latencia <i>Load Balancer</i> solo vs. <i>Global Accelerator</i>	20
10.	Estimado de tiempos y latencia <i>Load Balancer</i> solo vs. <i>Global Accelerator</i>	20

Resumen

Este documento presenta los resultados del ejercicio de arquitectura llevado a cabo como parte del examen final de la asignatura. Se incluyen los siguientes elementos: la publicación en GitHub, la documentación basada en el modelo de vistas 4 + 1, el análisis de *trade-offs*, la explicación del uso de cada patrón arquitectónico, la justificación de la idempotencia, las garantías de mensajería, y por último, las alternativas a la arquitectura seleccionada.

1. Introducción

El Banco ABC está llevando a cabo proyectos de actualización tecnológica que le permiten destacarse frente a la competencia. Uno de estos proyectos es el sistema de pago de servicios de convenios, lo que amplía su portafolio de servicios y refuerza la fidelidad de sus clientes. El Banco planea establecer alianzas estratégicas con varios proveedores de servicios públicos (agua, gas, electricidad y telefonía), también llamados convenios, para facilitar a sus clientes el pago de facturas a través de diversos canales de servicio, como cajeros automáticos, oficinas bancarias, llamadas telefónicas, portales web y aplicaciones móviles. El Banco ABC busca la flexibilidad de agregar nuevos convenios con otros proveedores de servicios de forma rápida, así como la capacidad de eliminar o rescindir convenios existentes. En ambos casos, se espera que estas acciones no afecten la disponibilidad del servicio. Cada proveedor de servicios públicos proporciona los mecanismos tecnológicos necesarios para que el banco pueda procesar las transacciones de pago. Esto significa que cada proveedor puede utilizar tecnologías distintas para ofrecer sus servicios.

Se llegó a un acuerdo de las capacidades/primitivas básicas que se deben soportar para cada convenio:

- Consulta de saldo a Pagar.
- Pago del Servicio.
- Compensación de pago (Opcional).

Principalmente el banco necesita un conjunto de servicios de integración que representen sus necesidades internas de negocio, lo cual les permite desacoplar los servicios de los proveedores y así no depender de sus detalles [1][2].

Aquí se puede consultar el repositorio en GitHub.

2. Alcance

Esta sección establece el alcance del parcial en el contexto del proyecto de pago de servicios públicos para el Banco ABC. El alcance determina los límites del proyecto, las funciones y características que se implementarán, así como las exclusiones y las áreas que quedan fuera del alcance de esta entrega.

Nota: Las decisiones, implementaciones, análisis y cualquier otro tipo de resultado presentados aquí se centrarán, sin limitarse, en abordar el caso promedio. Sin embargo, se excluirán aquellas implementaciones que expongan los escenarios más desfavorables.

2.1. Funciones y Características Incluidas

El proyecto tiene como objetivo principal presentar los siguientes puntos:

1. Proyecto publicado en GitHub.
2. Documentación de vistas 4 + 1.
3. ADR.
4. Trade off.
5. Explicación del por qué de la utilización de cada patrón.

6. Justificación de Idempotencia y garantías de mensajería.
7. Alternativas de arquitectura (limitado a vistas funcionales).
8. Implementación que permita ver la capacidad de adicionar convenios sin re-desplegar el sistema.

A continuación, se detallan las funciones y características que se incluirán en esta entrega a nivel de arquitectura:

- Integración con Proveedores de Servicios Públicos: El sistema permitirá la integración con proveedores de agua, gas y electricidad para gestionar el pago de facturas a través de múltiples canales.
- Soporte para Diversos Canales de Servicio: Los clientes podrán realizar pagos a través de cajeros automáticos, oficinas bancarias, llamadas telefónicas, portales web y aplicaciones móviles.
- Consultas y Pagos de Servicios: Se tendrá la capacidad para consultar saldos y realizar pagos, teniendo en cuenta la idempotencia para evitar errores en transacciones.
- Notificaciones: Se incluirán notificaciones para informar a los clientes sobre el estado de sus pagos.
- Flexibilidad: El sistema debe permitir agregar y eliminar nuevos convenios sin afectar la disponibilidad del servicio.

2.2. Exclusiones y Fuera de Alcance

Algunas funciones y características no serán parte de esta entrega. A continuación, se detallan las exclusiones y áreas que quedan fuera del alcance del proyecto:

- Implementación de mecanismos de Compensación de Pago: Aunque es opcional, no se incluirá en esta entrega para enfocarse en las otras funciones.
- Módulos de Administración de Usuarios: Se asumirá que los usuarios ya existen y no se crearán nuevos módulos para su administración.
- Implementación de capacidades de Recuperación ante Desastres: Si bien se considerarán mecanismos de redundancia y alta disponibilidad, la implementación de métodos para la recuperación ante desastres no será parte del alcance de este proyecto.
- Implementaciones que Exploten Peores Casos: No se incluirán implementaciones que aborden escenarios de peor caso, centrándose en casos promedio.

3. Supuestos

3.1. Banco ABC

Se asume:

- El Banco ABC es un banco tradicional colombiano.
- El sistema de pagos es nuevo. Sin embargo, otros sistemas relacionados a la operación de un banco tradicional existen y su disponibilidad no será tomada en cuenta.

3.2. Convenios / proveedores

Se asume:

- Todos los proveedores tienen un *Service Level Agreement* (SLA) de 100 %

3.3. Servicios públicos

En el cuadro 1 se observa la cantidad de hogares, la cantidad de personas y el promedio de personas por hogar. En el Cuadro 2 se observa el porcentaje de hogares con acceso a servicios públicos del total nacional, siendo el de energía eléctrica el más alto (98,1 %).

Se asume:

- Los valores de los porcentajes de acceso a servicios públicos se mantienen constantes hasta el día de hoy.
- El acueducto administrará también los pagos de alcantarillado y recolección de basuras y el porcentaje de acceso a los tres será del 87 %.
- No existen otros tipos de servicios públicos.
- Solamente se tendrán en cuenta los servicios públicos para hogares, quedan excluidos los comercios.

Cuadro 1: Hogares, personas y promedio de personas por hogar Total nacional y área 2023.

Hogares	Personas	Personas por hogar
18 009 000	52 314 000	2,9

Fuente: ECV 2024 [3].

Cuadro 2: Encuesta Nacional de Calidad de Vida (ECV) 2020.

Servicio Público	Total nacional
Energía eléctrica	98,1
Gas natural	64,5
Acueducto	87,0
Alcantarillado	74,4
Recolección de basuras	81,1

Fuente: ECV 2020 [4].

3.4. Hogares

En el cuadro 3 se muestra la medición realizada por el Banco de la República de la preferencia de uso de la población en general de diferentes instrumentos de pago en 2019, al realizar pagos habituales mensuales, entre los que se encuentran los pagos por servicios públicos [5]. En el mismo año, la población colombiana con algún producto financiero corresponde al 47,7% [5], ver figura 1. Tras la pandemia de COVID-19, en 2022 el 12,6% de los pagos habituales mensuales se realizó por medio de transacciones electrónicas [6] y el porcentaje de la población que respondió tener al menos un producto financiero (cuentas de ahorro y corriente, tarjetas débito y crédito, chequera, depósito electrónico o cualquier tipo de crédito) corresponde al 56,9% [7], ver figuras 2 y 3.

Se asume:

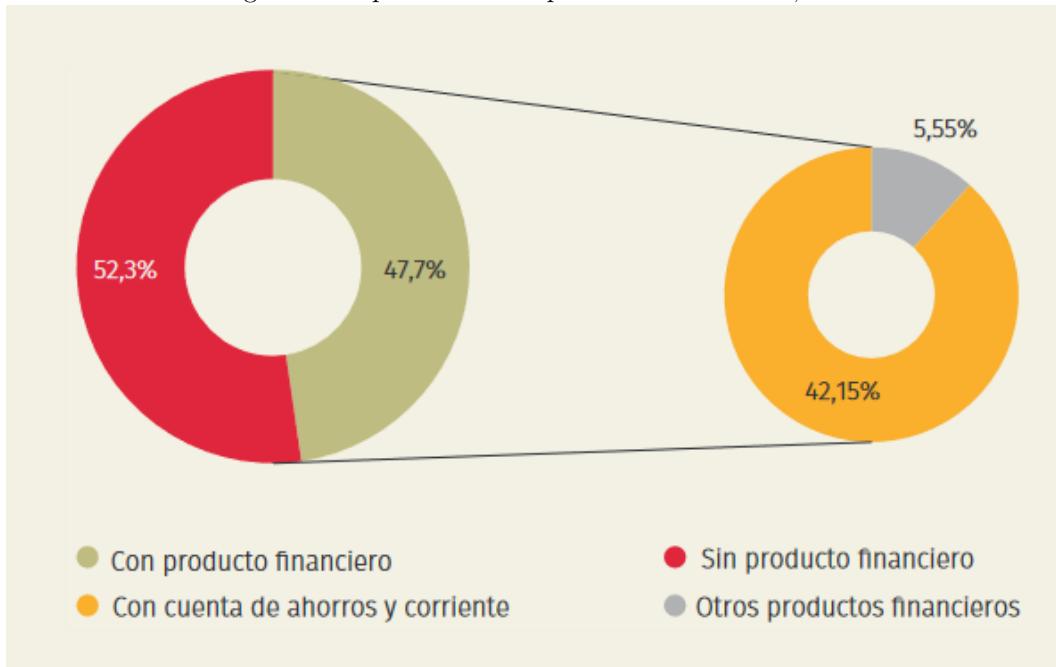
- El 50,8 % de la población que tiene cuenta de ahorros está repartido uniformemente en los hogares que en promedio tienen entre 2 y 3 personas. En otras palabras, todos los hogares tienen al menos una persona que podría pagar por a través de transferencia electrónica.
- Aunque se entiende que la cantidad de hogares es distinta a la población, el 12,6 % (2 269 134) de los hogares pagará a través de medios de pago electrónicos.

Cuadro 3: Volumen de transacciones al mes, por instrumento de pago utilizado en 2019.

Uso de Instrumento de pago	Número de operaciones (porcentaje)
Efectivo	88,1
Tarjeta débito	5,1
Tarjeta crédito	2,1
Transferencia	2,72
Otro	2,0

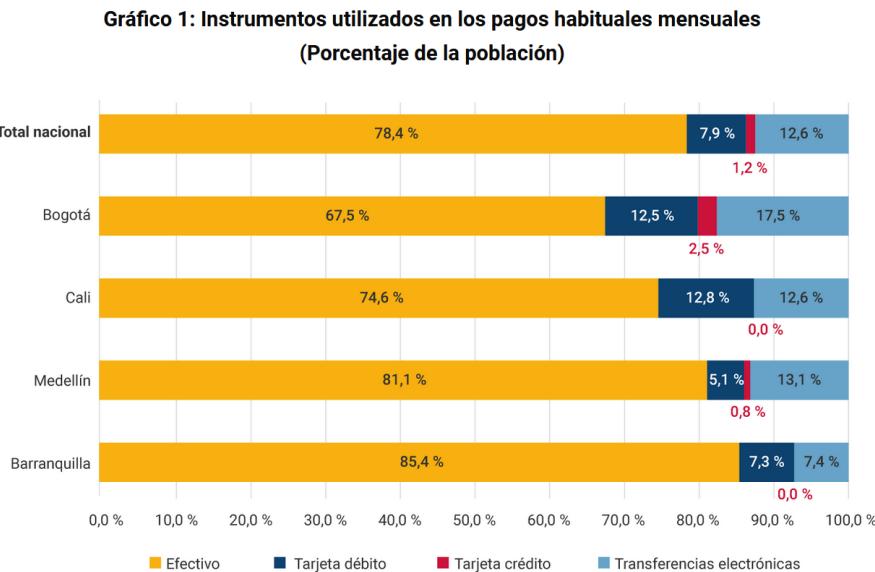
Fuente: Banco de la República 2022 [5].

Figura 1: Disponibilidad de productos financieros, 2019



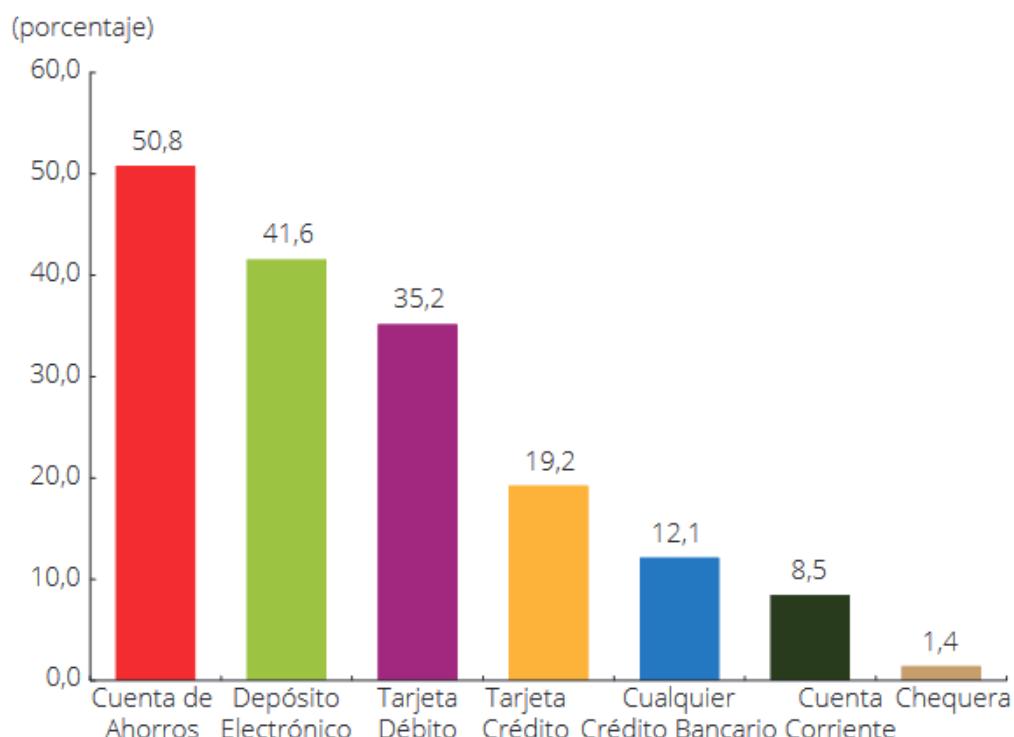
Fuente: Reporte de sistemas de pago, Banco de la República [5].

Figura 2: Instrumentos utilizados en los pagos habituales mensuales (Porcentaje de la población)



Fuente: El efectivo sigue siendo el instrumento de pago preferido por el público y el comercio en Colombia, Banco de la República [5].

Figura 3: Tenencia de medios e instrumentos de pago



Fuente: Encuesta de percepción sobre el uso de los instrumentos, Banco de la República [7].

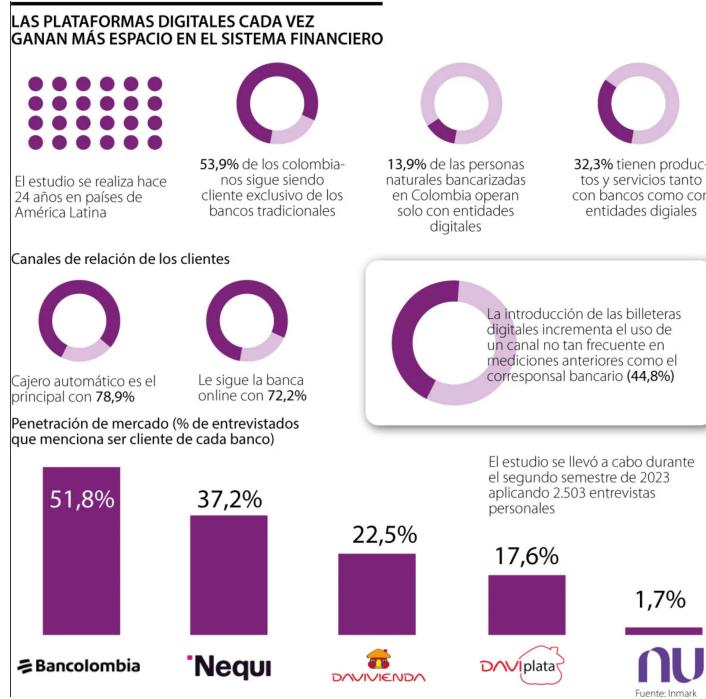
3.5. Usuarios del Banco ABC

En la figura 4 se puede ver que el 53,9 % de los colombianos son clientes exclusivos de bancos tradicionales. De esta población, el 51,8% es cliente de Bancolombia y el 22,5% de Davivienda [8].

Se asume:

- Los servicios públicos también pueden ser pagados por plataformas digitales distintas a los bancos tradicionales y entran en los porcentajes de transacciones realizadas a través de medios electrónicos.
- Del 12,6 % (2 269 134) de los hogares que pagará a través de medios de pago electrónicos, el 53,9 % (1 223 963) lo hará a través de bancos tradicionales, entre los que está el Banco ABC.
- Del 87,4 % (15 739 866) de los hogares que pagará en efectivo, el 53,9 % (8 483 787) lo hará a través de bancos tradicionales, entre los que está el Banco ABC.
- El Banco ABC tiene una cuota de mercado del 2 % para otros servicios distintos al pago de servicios públicos que será un sistema construido desde cero [1].
- Con su nuevo sistema y una estrategia agresiva, el Banco ABC busca captar el 2 % tanto de los hogares que realizan pagos a través de medios electrónicos (24 461) como de aquellos que lo hacen en efectivo (169 675). En total, representan 194 136 hogares, siendo el 12,6 % para pagos electrónicos y el 87,4 % en efectivo.

Figura 4: Las plataformas digitales cada vez ganan más espacio en el sistema financiero



Fuente: La República [8].

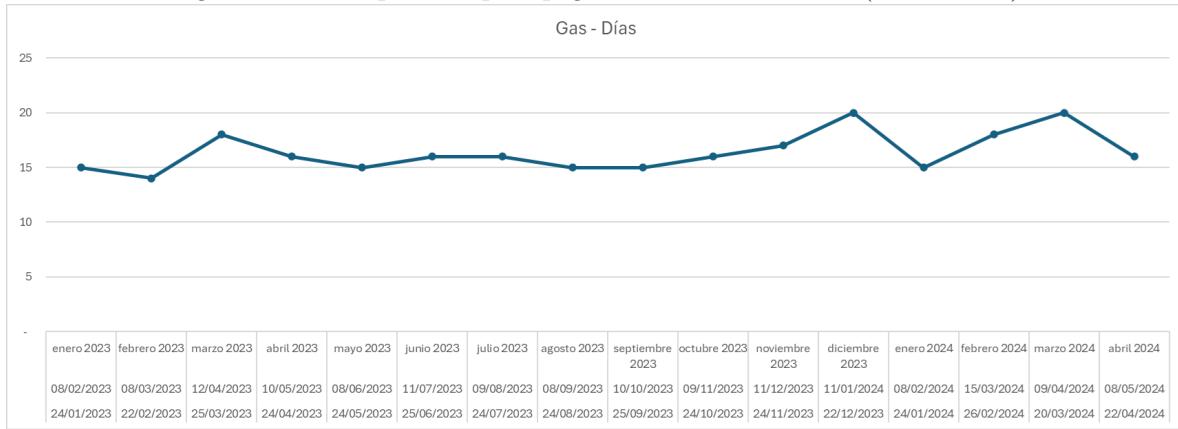
3.6. Ciclos de facturación

En las figuras 5, 6 y 7 se observa el número de días que toma un ciclo de facturación para los servicios de Energía eléctrica, Gas y acueducto. El promedio de días de los tres servicios es de 15 días.

Se asume:

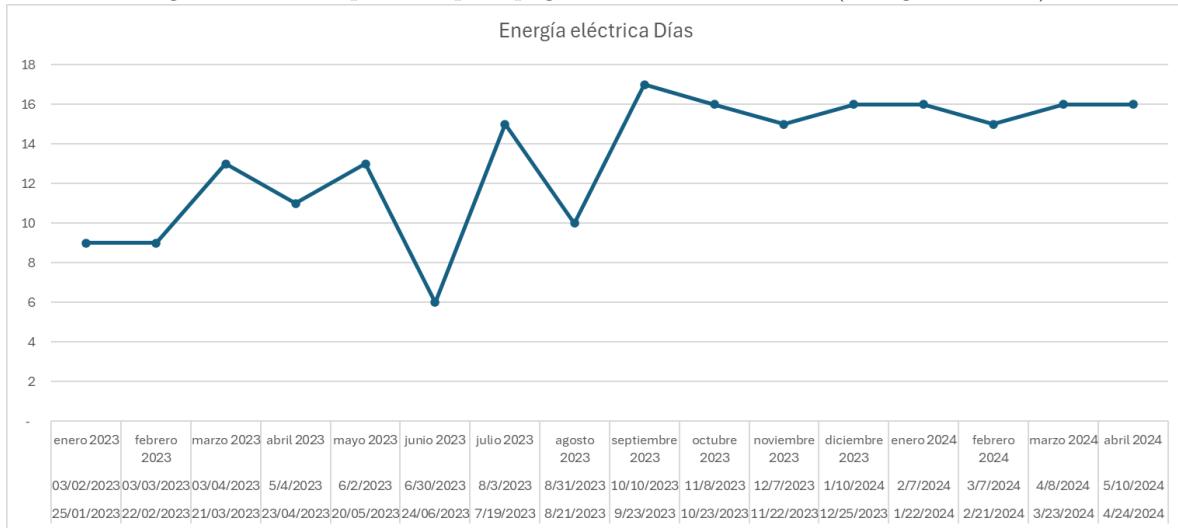
- Solamente existe un ciclo de facturación para todos los servicios.
- El ciclo de facturación dura 15 días.

Figura 5: Días disponibles para pago: Ene 2023 - Abr 2024 (Gas natural)



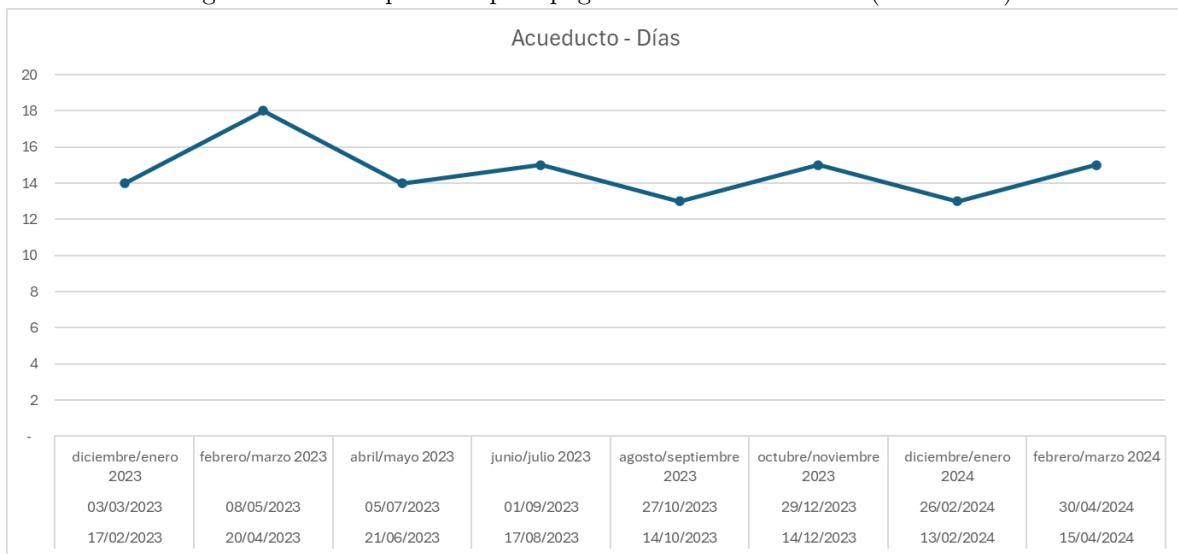
Fuente: Elaboración propia.

Figura 6: Días disponibles para pago: Ene 2023 - Abr 2024 (Energía eléctrica)



Fuente: Elaboración propia.

Figura 7: Días disponibles para pago: Ene 2023 - Abr 2024 (Acueducto)



Fuente: Elaboración propia.

3.7. Transacciones / Solicitudes

Existen tres tipos de servicios que puede consultar un cliente: identificarse, consultar información de pago y hacer una orden de pago. Siendo H la cantidad de hogares, **0,981** el porcentaje de hogares con acceso a energía eléctrica, **0,645** el porcentaje de hogares con acceso al gas natural, **0,87** el porcentaje de hogares con acueducto, alcantarillado y recolección de basuras, y S la cantidad de solicitudes, se cumple la siguiente ecuación:

$$S = 3(H * 0,981 + H * 0,645 + H * 0,87)$$

Simplificada:

$$S = 7,488H$$

Se asume:

- Para hogares que realizan pagos electrónicos (24 461) se tienen 183 163 solicitudes al sistema, mientras que para hogares que no realizan pagos electrónicos (169 675) se tienen 1 270 526 solicitudes al sistema durante el ciclo de facturación. En total serían **1 453 689** solicitudes que se pueden esperar dentro de los 15 días de duración.

3.8. TPS

La ley de Little se puede aplicar a cualquier sistema en equilibrio en el que los clientes llegan, pasan un cierto tiempo y se marchan. Suponemos que λ es la tasa media de llegada, W es el tiempo medio que un cliente pasa en el sistema y L es el número medio de clientes en el sistema [9].

$$\lambda = L/W$$

Acomodándolo al contexto de este ejercicio, podemos suponer que λ es la cantidad de solicitudes o transacciones por segundo que un sistema puede soportar, W Es el tiempo promedio que se tarda el sistema en procesar una solicitud y L Es el número de solicitudes procesadas simultáneamente.

En la figura 8 muestra la distribución del número de solicitudes / transacciones por día durante un ciclo normal de facturación. Se hace un promedio de las transacciones de cada día, suponiendo que en un día normal habrá una ventana de 12 horas donde los clientes realizarán las transacciones, por el principio de palomar [10] podemos saber que hay un 100 % de probabilidad de que al menos 7 solicitudes / transacciones ocurran en un segundo el último día del ciclo de facturación, ver cuadro 4

Cuadro 4: Distribución de transacciones durante ciclo de facturación y transacciones promedio por día.

Día	Número de transacciones	Promedio de transacciones por segundo en un día (12 h)
1	750	0,017361111
2	3600	0,083333333
3	5000	0,115740741
4	6000	0,138888889
5	7600	0,175925926
6	9800	0,226851852
7	12000	0,277777778
8	17010	0,393750000
9	29740	0,688425926
10	51900	1,201388889
11	90310	2,090509259
12	169805	3,930671296
13	249456	5,774444444
14	332111	7,687754630
15	468607	10,84738426

Fuente: Elaboración propia.

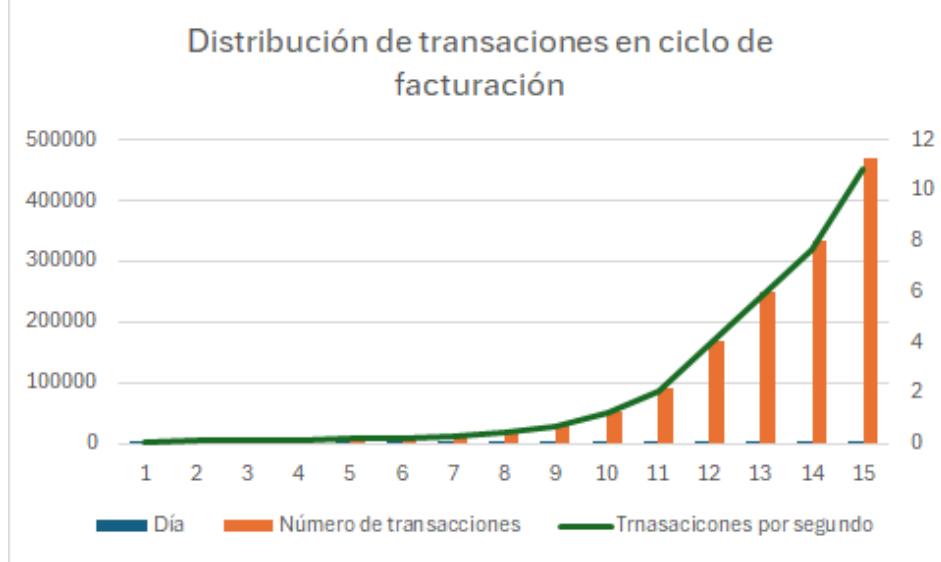
Con base en el Cuadro 4 podemos saber que el promedio de transacciones que se solicitan por segundo durante el ciclo de facturación es de 4,21 (L), que además cubre el 86,7% de los días (1 - 13). Adicionalmente, se conoce que existen 3 tipos de solicitudes: identificar cliente, consultar información de pago y hacer una orden de pago. Suponiendo que la identificación tarda 0,5 s; la consulta de información de pago tarda en realizarse 0,5 s; y hacer una orden de pago tarda 1 s, tenemos que: el promedio de tiempo que tarda en realizarse una transacción es de 0,67 s (W). Aplicando la ley de Little encontramos que el número de transacciones por segundo (TPS) o *Throughput* es de 6,28:

$$6,28 = 4,21/0,67$$

Se asume:

- El *Throughput* del sistema es de 7 TPS.
- El peor caso posible es en el que todas las transacciones ocurran el mismo día. Es decir: 33.65 transacciones por segundo durante 12 horas.

Figura 8: Distribución de transacciones durante el ciclo de facturación



Fuente: Elaboración propia.

4. Atributos

En esta sección, se presentarán las descripciones de los atributos de calidad para el proyecto de pago de servicios públicos del Banco ABC. Se abordarán los escenarios relevantes y las razones detrás de la elección de ciertas soluciones de arquitectura. Cada decisión se fundamentará en los requisitos del proyecto. Además, se destacarán las implicaciones de estas decisiones en aspectos relevantes para la arquitectura del sistema.

4.1. Escalabilidad

4.1.1. Escenario

El 2 % del mercado que utiliza bancos tradicionales, estimado en 194 136 hogares, podrá realizar pagos electrónicos y en efectivo al Banco ABC a lo largo de los 15 días del ciclo de facturación.

4.1.2. Estímulo

- Fuente: Tráfico de órdenes de pago durante los días del ciclo de facturación.
- Tipo: Aumento repentino de usuarios concurrentes.
- Volumen:
 - 183 163 solicitudes al sistema a través de medios de pago electrónicos.
 - 1 270 526 solicitudes al sistema a través de medios de pago en efectivo.
 - En total 1 453 689 solicitudes al sistema.
- Duración: 15 días.

4.1.3. Ambiente

- Cajero en oficina y portal web del Banco ABC.
- Infraestructura basada en la nube con escalabilidad automática.

4.1.4. Respuesta

- Mantener la disponibilidad del sistema al 99.99 %
- Mantener los tiempos de respuesta promedio (P50) por debajo de 500 ms.
- Mantener los tiempos de respuesta del P90 por debajo de 1 s.
- Soportar un volumen de transacciones de hasta 420 solicitudes por minuto.
- Garantizar que los sistemas de procesamiento de transacciones puedan soportar el aumento.

4.1.5. Medición

- Tiempo promedio de respuesta para el sistema (P50 y P90).
- Porcentaje de transacciones exitosas.
- Tiempo dedicado al procesamiento de transacciones.

4.1.6. Atributo de Calidad

Escalabilidad: El sistema debe ser capaz de soportar un aumento significativo de tráfico sin comprometer el rendimiento. Se espera que el sistema pueda escalar hasta 2 veces su capacidad normal durante un período de al menos 2 días.

4.1.7. Atributos no cualitativos:

- **Rendimiento**: El sistema debe mantener tiempos de respuesta rápidos incluso bajo una carga alta. Se espera que el sistema mantenga un tiempo de respuesta promedio (P50) por debajo de 500 ms y un tiempo de respuesta del P90 por debajo de 1 s incluso durante el pico de tráfico del ciclo de facturación.
- **Disponibilidad**: El sistema debe estar altamente disponible y poder soportar un aumento de tráfico sin tiempos de inactividad. Se espera que el sistema tenga una disponibilidad del 99.99 % durante el ciclo de facturación.

4.2. Modificabilidad

4.2.1. Escenario

Un convenio podrá ser agregado o eliminado del sistema sin hacer que algún componente existente del sistema sea re-desplegado.

4.2.2. Estímulo

- Fuente: El equipo de convenios introduce un nuevo proveedor que debe ser integrado en el sistema.
- Tipo: Esporádico.
- Volumen: 1 convenio.
- Duración: 5 minutos.

4.2.3. Ambiente

- Infraestructura basada en la nube con escalabilidad automática.

4.2.4. Respuesta

- El sistema permite la adición del nuevo convenio de manera dinámica.

4.2.5. Medición

- El nuevo convenio se añade sin necesidad de re-despliegue del componente del sistema.

4.2.6. Atributo de Calidad

Modificabilidad: El sistema debe ser capaz de soportar la incorporación dinámica de convenios sin re-despliegue.

5. ADR

5.1. Escalabilidad

5.1.1. Escenario

Escalabilidad a partir del P86 durante las fechas hábiles de facturación.

5.1.2. Fecha

2024-05-14.

5.1.3. Autor

Los Pollos Hermanos.

5.1.4. Resumen

La distribución de transacciones durante ciclo de facturación y transacciones promedio por día, experimenta un incremento fuerte en el volumen de peticiones que deben ser procesadas después del P86 o día 14, teniendo en cuenta que en ese intervalo se espera procesar 11 TPS, en caso contrario 7 TPS.

5.1.5. Decisiones

5.1.5.1. Pruebas de carga

Definición de objetivos de rendimiento

- Tiempo de respuesta P86: Establecer un tiempo de respuesta P86 objetivo promedio de 700 ms, asegurando que el 86 % de las solicitudes se atiendan dentro de este plazo. Esto garantizará una experiencia de usuario fluida para la mayoría de los usuarios, incluso durante los períodos de mayor tráfico.
- Tasa de error objetivo: Fijar una tasa de error objetivo inferior al 1 %, lo que significa que menos del 1 % de las solicitudes deberían fallar.
- Rendimiento objetivo: Establecer un objetivo de rendimiento de 420 transacciones por minuto, asegurando que el sistema pueda manejar el volumen de tráfico esperado.

Selección de una herramienta de prueba de carga

- **JMeter:** Una herramienta de prueba de carga de código abierto popular que puede simular una gran cantidad de usuarios concurrentes y medir los tiempos de respuesta.
- **Siege:** Una herramienta ligera de línea de comandos para generar tráfico de alto volumen.

Creación de un escenario de prueba de carga

- **Definir el perfil de usuario:** Identificación del usuario, solicitud de información de pago y enviar orden de pago.
- **Establecer el patrón de tráfico:** Patrón de tráfico establecido en el cuadro de “Distribución de transacciones durante ciclo de facturación y transacciones promedio por día”.
- **Tiempo de subida y bajada:** Aumentar gradualmente el número de usuarios concurrentes para simular el aumento natural del tráfico y luego disminuirlo gradualmente.

Ejecución de pruebas de carga

- **Ejecutar múltiples pruebas:** Realizar múltiples pruebas para recopilar datos consistentes e identificar cualquier valor atípico.
- **Monitorizar métricas de rendimiento:** Monitorizar continuamente métricas de rendimiento clave, incluidos los tiempos de respuesta P86, el rendimiento y las tasas de error.
- **Analizar resultados:** Analizar los resultados de la prueba para identificar cualquier cuello de botella de rendimiento y determinar si el sistema cumple con el objetivo de tiempo de respuesta P86.

Refinamiento y repetición

- **Identificar cuellos de botella:** Basándose en los resultados de la prueba, identificar cualquier cuello de botella de rendimiento que deba abordarse.
- **Realizar mejoras en el sistema:** Implementar cambios para abordar los cuellos de botella identificados y repetir las pruebas de carga para verificar las mejoras.
- **Monitorización continua:** Monitorizar continuamente el rendimiento del sistema y realizar pruebas de carga periódicas para garantizar que el sistema se mantenga con un buen rendimiento bajo diferentes condiciones de tráfico.

Consideraciones adicionales

- **Pruebas distribuidas:** Considere utilizar herramientas de prueba de carga distribuidas para simular el tráfico desde múltiples ubicaciones y regiones.
- **Herramientas de monitorización de rendimiento:** Utilice herramientas de monitorización de rendimiento para identificar el uso de recursos y los posibles cuellos de botella.
- **Monitorizar de usuarios reales (RUM):** Emplee herramientas de RUM para capturar datos reales de la experiencia del usuario y colecciónarlos con los resultados de las pruebas de carga.
- **Introducción en nuevo mercado:** El Banco ABC espera introducirse el mercado financiero de Brasil, se debe considerar estar preparados para soportar esta región.

Al seguir estos pasos y considerar las recomendaciones adicionales, podrá medir efectivamente los tiempos de respuesta P86 durante las pruebas de carga y garantizar que su sistema cumpla con las expectativas de rendimiento para el aumento del volumen del ciclo final de facturación determinado después del P86.

5.1.5.2. Infraestructura

Adoptar una infraestructura en la nube escalable

- Utilizar el servicio de nube Amazon Web Services (AWS). Este ofrece recursos escalables y elásticos que pueden adaptarse a las demandas cambiantes del tráfico.
- Implementar una arquitectura de microservicios: Dividir la aplicación en responsabilidades más pequeñas e independientes permite escalar cada microservicio de forma individual según sea necesario.
- Utilizar bases de datos escalables: Elegir bases de datos NoSQL como DynamoDB o Redis que puedan escalar horizontalmente agregando más nodos al clúster, en caso de gestión de usuario y proveedores elegir bases de datos SQL como PostgreSQL.

Implementar escalamiento automático

- Configurar grupos de escalamiento automático: Definir grupos de recursos, como servidores web o instancias de base de datos, que se escalan vertical y horizontal automáticamente según la demanda.
- Establecer umbrales de escalado: Determinar los indicadores clave de rendimiento (KPI) que desencadenarán el escalado automático, como el uso de la CPU, la memoria, el tiempo de respuesta o el número de peticiones.
- Utilizar políticas de escalado: Definir cómo se escalarán los grupos de recursos, por ejemplo, agregando o eliminando instancias de forma gradual o en incrementos específicos.

Utilizar un balanceador de carga

- Elegir un balanceador de carga adecuado: Seleccionar un balanceador de carga que pueda manejar el volumen de tráfico esperado y distribuirlo de manera eficiente entre los servidores.
- Configurar algoritmos de balanceo de carga: Implementar algoritmos como round-robin o basado en el rendimiento para distribuir el tráfico de manera uniforme entre los servidores disponibles.
- Monitorizar el estado del servidor: El balanceador de carga debe monitorizar el estado de los servidores y eliminar temporalmente aquellos que no estén disponibles o que no respondan.

Implementar una red de entrega de contenido (CDN)

- Seleccionar un proveedor de CDN: Elegir un proveedor de CDN que tenga una red global de servidores y pueda ofrecer contenido de baja latencia a los usuarios en dos regiones.

5.2. Modificabilidad

5.2.1. Escenario

Hace referencia al mismo escenario del atributo [4.2.1](#).

5.2.2. Fecha

2024-05-29.

5.2.3. Autor

Los Pollos Hermanos.

5.2.4. Resumen

El nuevo sistema requiere prescindir de un re-despliegue completo cada vez que se añade un nuevo convenio, debido a que esto resulta en tiempos de inactividad y en una carga adicional para el equipo de desarrollo y operaciones. El objetivo es diseñar una solución que permita la incorporación de nuevos convenios de manera dinámica y sin la necesidad de re-despliegue, manteniendo la continuidad operativa y minimizando el impacto en la disponibilidad del sistema.

5.2.5. Decisiones

5.2.5.1. Punto de cambio

Durante

- Debido a su implicación dentro de las decisiones de arquitectura, diseño y desarrollo, es necesario dejar claro dónde se encuentra el punto de cambio (modificabilidad) dentro del sistema que permita agregar convenios de forma dinámica. Por ello, se presenta el cuadro [5](#) comparando los posibles puntos de cambio:

Cuadro 5: Punto de cambio - Durante.		
Durante	Impacto	calificación
Implementación	re-despliegue.	0
Compilación	re-despliegue.	0
Despliegue	re-despliegue.	0
Ejecución	sin despliegue.	100

Fuente: *Elaboración propia.*

La decisión que se toma es la de tener el punto de cambio durante la ejecución.

¿Quién?

- Debido a la implicación en las decisiones de arquitectura, es necesario dejar claro quién será el encargado de realizar el cambio (agregar un convenio). En el cuadro [6](#) se muestra la calificación dada a cada rol posible.

Cuadro 6: Punto de cambio - Quién.

Rol	Impacto	calificación
Usuario final	El usuario final debería conocer cómo agregar un convenio.	0
Desarrollador	El tiempo para desarrollo se disminuye.	10
Administrador del sistema	Se asigna un administrador para el sistema de convenios.	100
Técnico de Sistemas	El tiempo para resolución de incidentes disminuye.	10

Fuente: Elaboración propia.

Se sabe que el Banco ABC cuenta con el personal capacitado para hacer frente a cualquier proyecto, mientras este sea viable [1]. Como el Banco ABC tiene las capacidades de ejecución de proyectos, la decisión es la de asignar a una persona del equipo que administra el sistema de convenios para que sea la encargada de agregar los convenios.

Patrón

- Debido a la implicación en las decisiones de arquitectura, diseño y desarrollo, es necesario dejar claro qué patrón deberá seguirse para que al agregar un convenio no haya re-despliegue de algún componente existente del sistema. En el cuadro 7 se muestra la calificación dada a cada patrón evaluado.

Cuadro 7: Punto de cambio - Patrón.

Patrón	Impacto	calificación
Cliente / Servidor	El convenio debe iniciar la comunicación.	0
Microkernel	Se extiende la funcionalidad a través de configuraciones.	100
Publicador / suscriptor	El convenio debería entender en cierto nivel los eventos del sistema.	0
Capas + N-Tiers	Separación física y/o lógica de los componentes de despliegue y <i>Software</i> .	90

Fuente: Elaboración propia.

Si bien la implementación del patrón Microkernel permitirá la exención de la funcionalidad (agregar convenios) a través de configuración, la implementación de capas / *tiers* permite que los módulos puedan ser desarrollados y adaptados de forma separada con poca o ninguna dependencia entre las partes. La combinación de ambos patrones permitirá seguir esta decisión de forma articulada junto con las demás. Por ejemplo, el ADR 5.1.

6. 4 + 1

6.1. Escenario de escalabilidad

Será el mismo contemplado en el ADR 5.1.1 y el atributo 4.1.1 de escalabilidad.

6.1.1. Vista lógica

En la figura 9 se puede ver la vista lógica, que permite ver cómo funciona el sistema ante un pico de transacciones por segundo (TPS) de 11, el cual podría presentarse en el último día del ciclo de facturación como se ve en el cuadro 4. Cuando se superen las 7 TPS, los sistemas dentro de la sección redundante empezarán a encenderse para evitar pérdidas de rendimiento, derivadas del límite de 7 TPS por sistema en cada sección. Teniendo esta sección redundante el sistema puede alcanzar aproximadamente 14 TPS, unas 840 transacciones por minuto (TPM) superiores a las 420 esperadas, garantizando que el sistema pueda soportar el aumento de TPS.

6.1.2. Vista física

En la figura 10 se puede ver la vista física, que permite ver la independencia de despliegues entre sistemas. Cuenta con los servicios de Amazon descritos en la tabla 8.

Cuadro 8: Servicios de Amazon utilizados.

RDS	SLA
EC2 + Available zone	99,99 %
DynamoDB	99,99 %
Load Balancer	99,99 %
Global accelerator	99,989 %

Fuente: Elaboración propia [5].

Para calcular el SLA total, multiplicamos cada SLA de cada servicio utilizado en una región:

$$SLA_{total} = SLA_{Dynamo} * SLA_{RDS} * SLA_{LoadBalancer} * SLA_{EC2} * SLA_{Route53}$$

Reemplazando:

$$99,99 \% * 99,99 \% * 99,99 \% * 99,99 \% * 99,989 \% = 99,94 \%$$

99,94 % sigue estando por debajo del 99,99 % esperado. Por esta razón, se añade una nueva región para mejorar el SLA final. Tenemos qué el porcentaje de indisponibilidad en una región es 0,06 %:

$$100 \% - 99,94 \% = 0,06 \%$$

Lo que quiere decir que el porcentaje de indisponibilidad en dos regiones combinadas es:

$$0,06 \% * 0,06 \% = 0,000036 \%$$

Lo que quiere decir que el SLA total utilizando las dos regiones es de 99.999964 %:

$$100 \% - 0,000036 \% = 99,999964 \%$$

De esta forma se asegura el 99,99 % de disponibilidad.

En los cuadros 9 y 10 se muestra la comparación de tiempos estimados, con base en el ejemplo de AWS para Global Accelerator [11], entre utilizar únicamente un balanceador de carga y utilizar el servicio de *Global Accelerator*. Con el uso de este último se garantiza que en P85 se puedan procesar 16,1 TPS con un tiempo menor a 600 ms:

Cuadro 9: Estimado de tiempos y latencia Load Balancer solo vs. Global Accelerator.

Endpoint	Tiempo tomado	TPS	TPT	velocidad de transferencia	tiempo de conexión (ms)
Global Accelerator	70s	16,1	70,9ms	1413,6Kbps	670
Load balancer	180s	7,26	190,21ms	627,01Kbps	1792

Fuente: Medición del rendimiento de AWS Global Accelerator y análisis de los resultados [11].

Cuadro 10: Estimado de tiempos y latencia Load Balancer solo vs. Global Accelerator.

Endpoint	P50 (ms)	P90 (MS)	P95 (MS)	P99 (ms)
Global Accelerator	433	575	688	1019
Load balancer	1512	2114	2625	5627

Fuente: Medición del rendimiento de AWS Global Accelerator y análisis de los resultados [11].

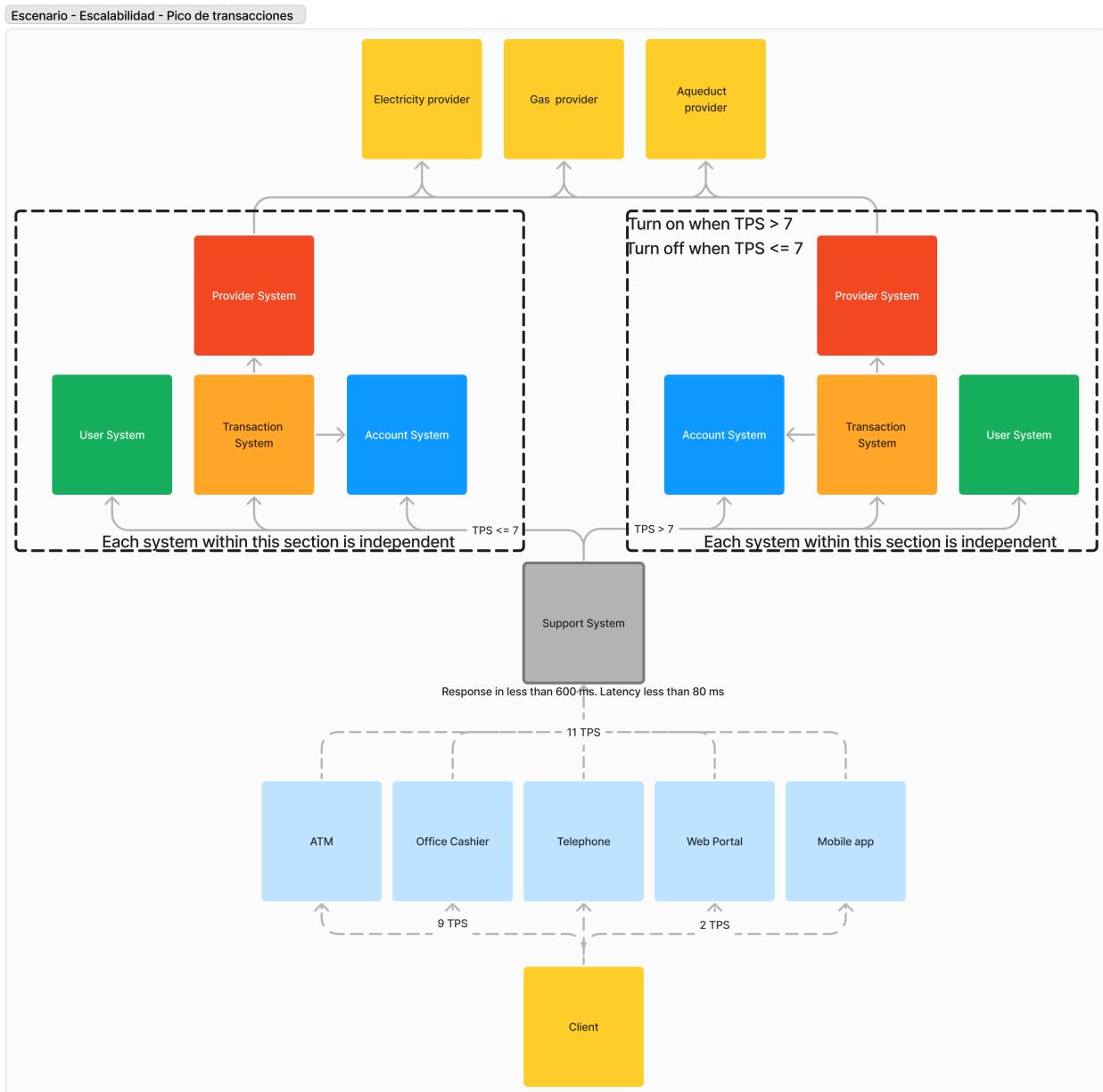
6.1.3. Vista de procesos

El diagrama realizado para la vista de procesos busca establecer la interacción de la solicitud realizada por el cliente -en este caso cualquier medio utilizado para el pago- que llega será gestionada por el *Gateway*, quien estará pendiente del volumen de solicitudes al que estará expuesto y definirá un valor que posteriormente será utilizado para determinar si debe mantener su estado actual, o debe empezar el proceso de escalamiento. Una vez ajustado los componentes, se continuará con el procesamiento de la petición validando su usuario quien le responderá al *Gateway* y le indicara en caso positivo continuar al proceso de pago y guardar información en la base de datos.

6.1.4. Vista de desarrollo

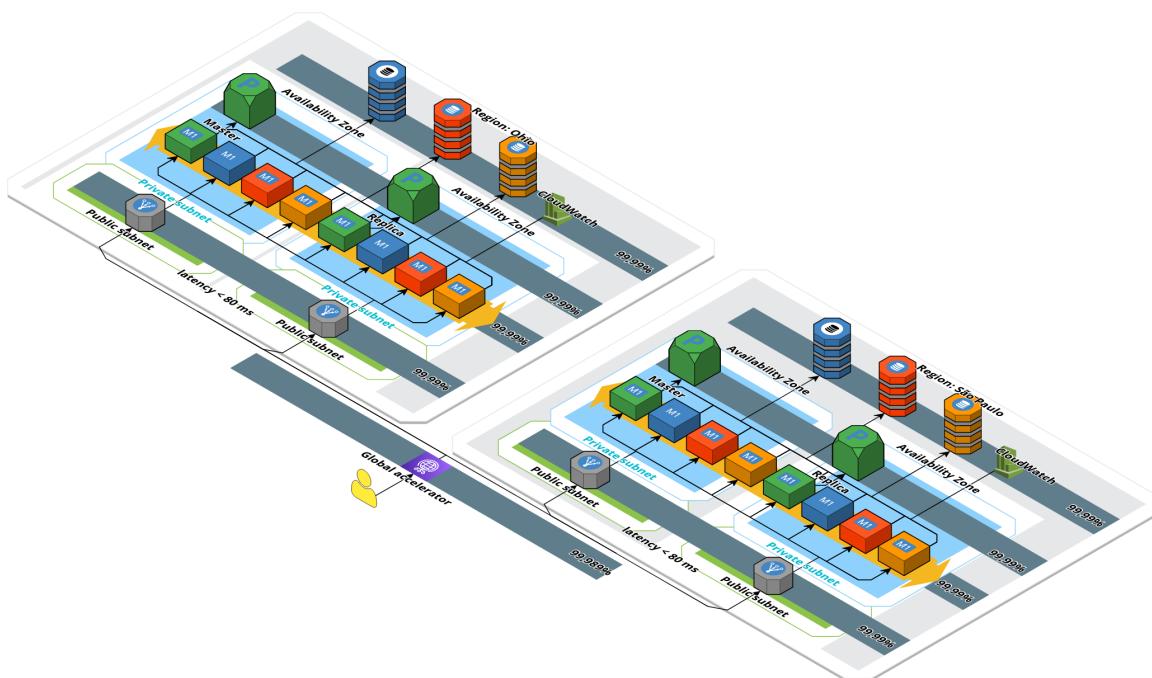
En la figura 12 se puede ver la vista de desarrollo, en la que se el *Software* y el *Hardware* encuentran divididos en subsistemas que pueden ser desarrollados / desplegados independientemente por grupos pequeños de desarrolladores, e incluso, ser tercerizados como puede ocurrir [1]. Los subsistemas están organizados en una jerarquía de capas, cada capa sirve a las capas superiores. De esta forma, se puede ver cómo la infraestructura que asegura una disponibilidad del 99,99 % y que puede escalar de 7 TPS a 14 TPS sirve al *software* que se encargará que, a su vez, sirve a los componentes externos (cajeros automáticos, aplicaciones móviles, etc.).

Figura 9: Vista Lógica - escenario de escalabilidad.



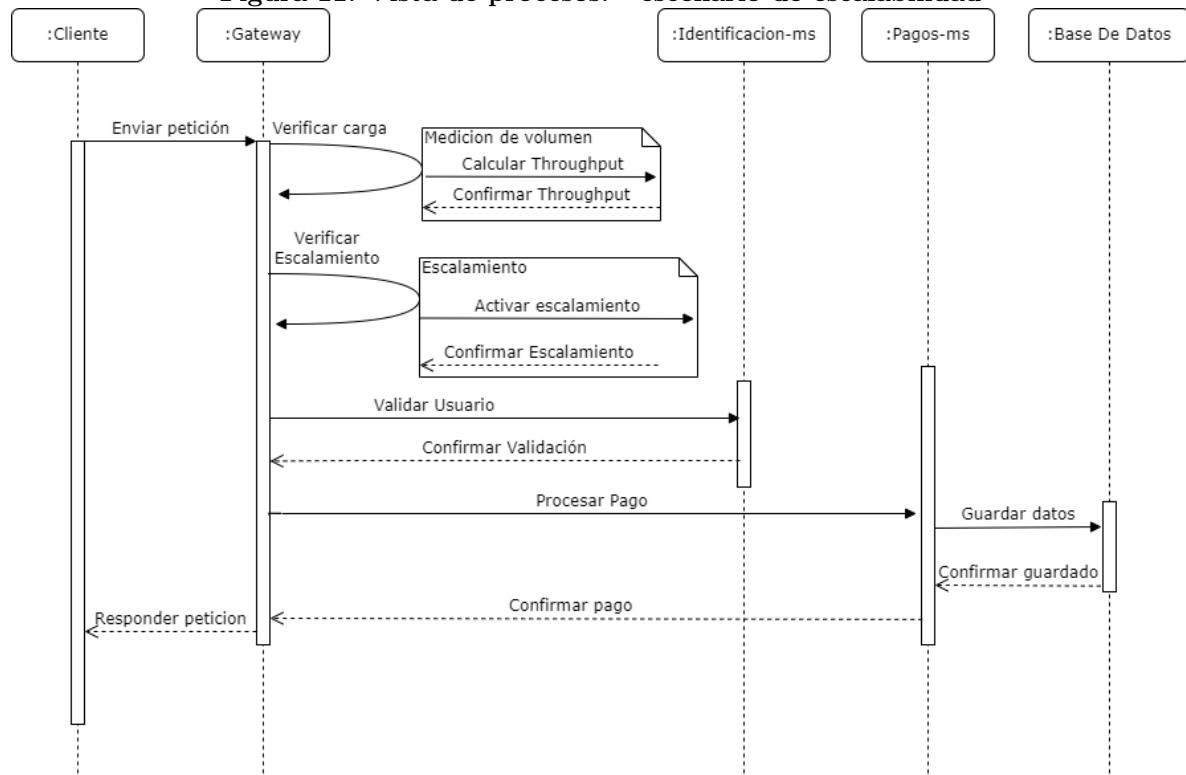
Fuente: Elaboración propia.

Figura 10: Vista Física - escenario de escalabilidad.



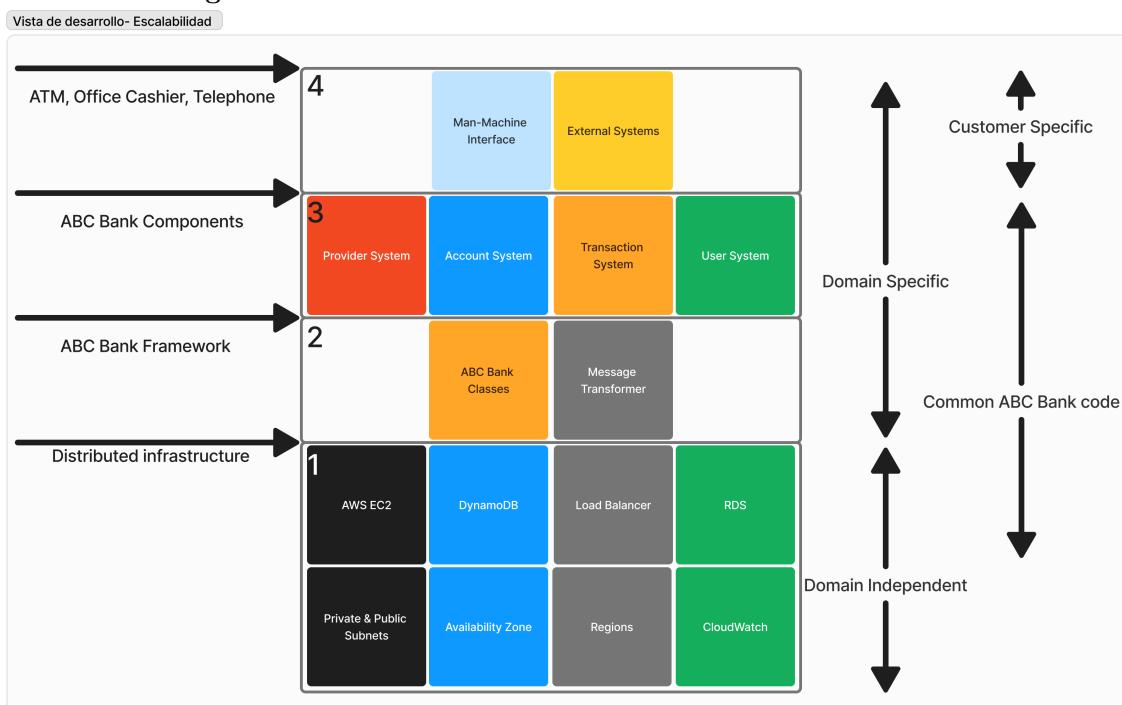
Fuente: Elaboración propia.

Figura 11: Vista de procesos. - escenario de escalabilidad



Fuente: Elaboración propia.

Figura 12: Vista de desarrollo. - escenario de escalabilidad



Fuente: Elaboración propia.

6.2. Escenario de Modificabilidad

Será el mismo contemplado en el ADR 5.2.1 de modificabilidad.

6.2.1. Vista lógica

En la figura 13 se puede ver la vista lógica, que permite ver cómo funciona el sistema frente al registro esporádico de algún convenio. Será un administrador del sistema de convenios el encargado de recolectar la información necesaria para llenar el formato de configuración de nuevo convenio y cargarlo en el sistema de proveedores. De esta forma se ve cómo, a través de la configuración, se puede extender la funcionalidad del sistema.

6.2.2. Vista física

En la figura 14 se puede ver la vista física, en donde se aprecia la independencia de despliegues entre sistemas que evita re-desplegar cada vez que se agregue una configuración.

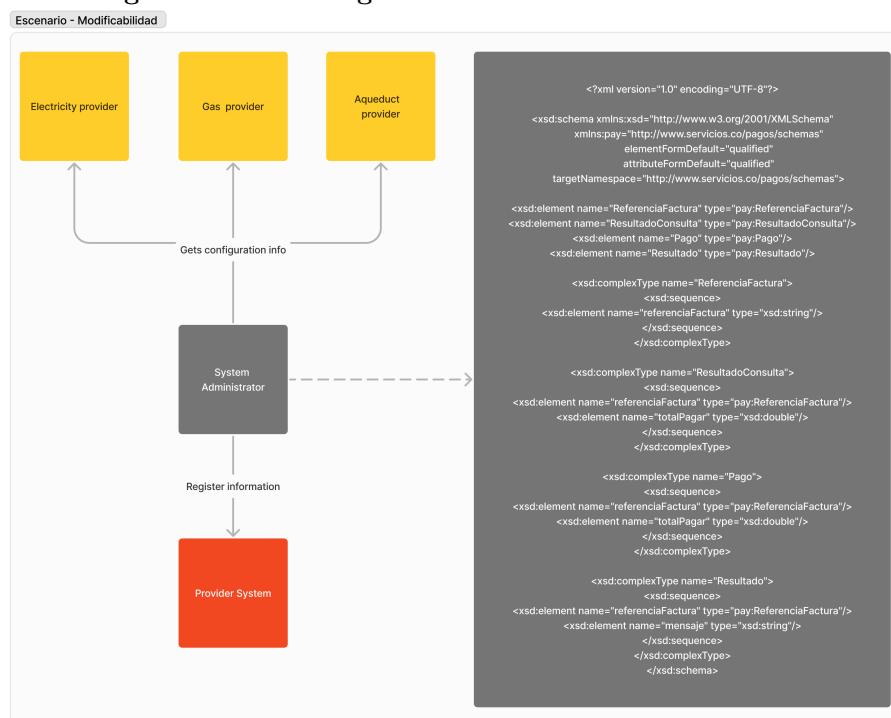
6.2.3. Vista de procesos

La figura 15 Aborda en una vista de procesos las principales abstracciones de la vista lógica dentro de la arquitectura de procesos. En este caso, a través de la externalización de configuraciones dirigida por el uso del patrón microkernel, donde a una configuración "base" se le adicionan nuevas configuraciones específicas, extendiendo la funcionalidad.

6.2.4. Vista de desarrollo

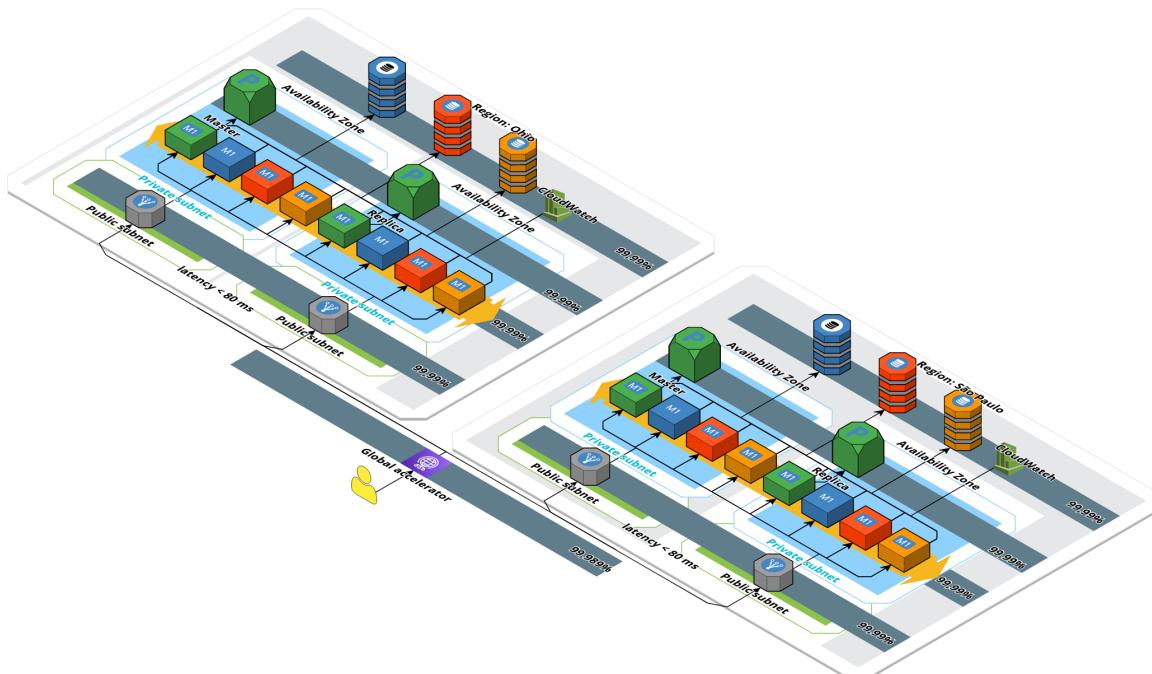
La figura 16 aborda la vista de desarrollo por capas / tiers, en la que se el *Software* y el *Hardware* encuentran divididos en subsistemas que pueden ser desarrollados / desplegados independientemente por grupos pequeños de desarrolladores, e incluso, ser tercerizados como puede ocurrir [1]. Los subsistemas están organizados en una jerarquía de capas, cada capa sirve a las capas superiores.

Figura 13: Vista Lógica - escenario de modificabilidad.



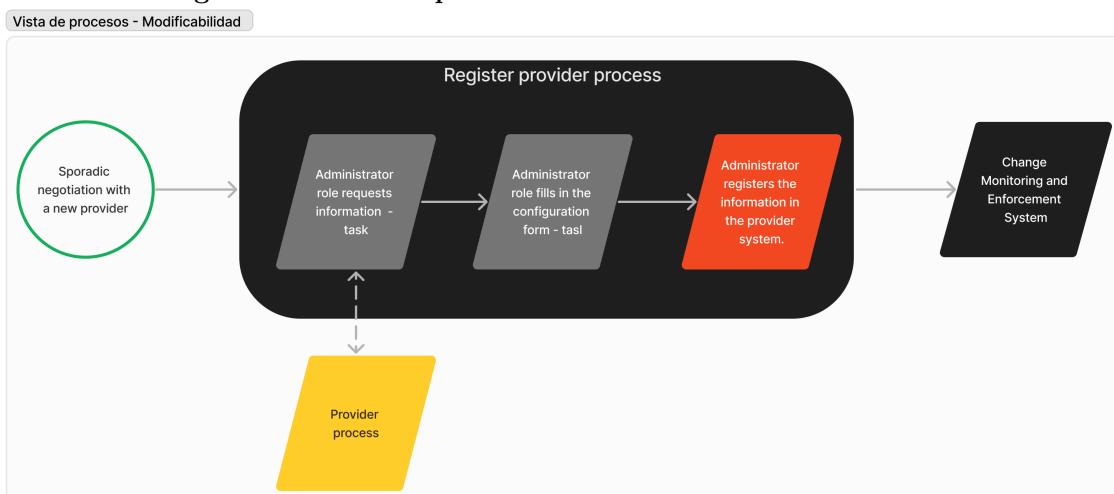
Fuente: Elaboración propia.

Figura 14: Vista Física - escenario de modificabilidad.



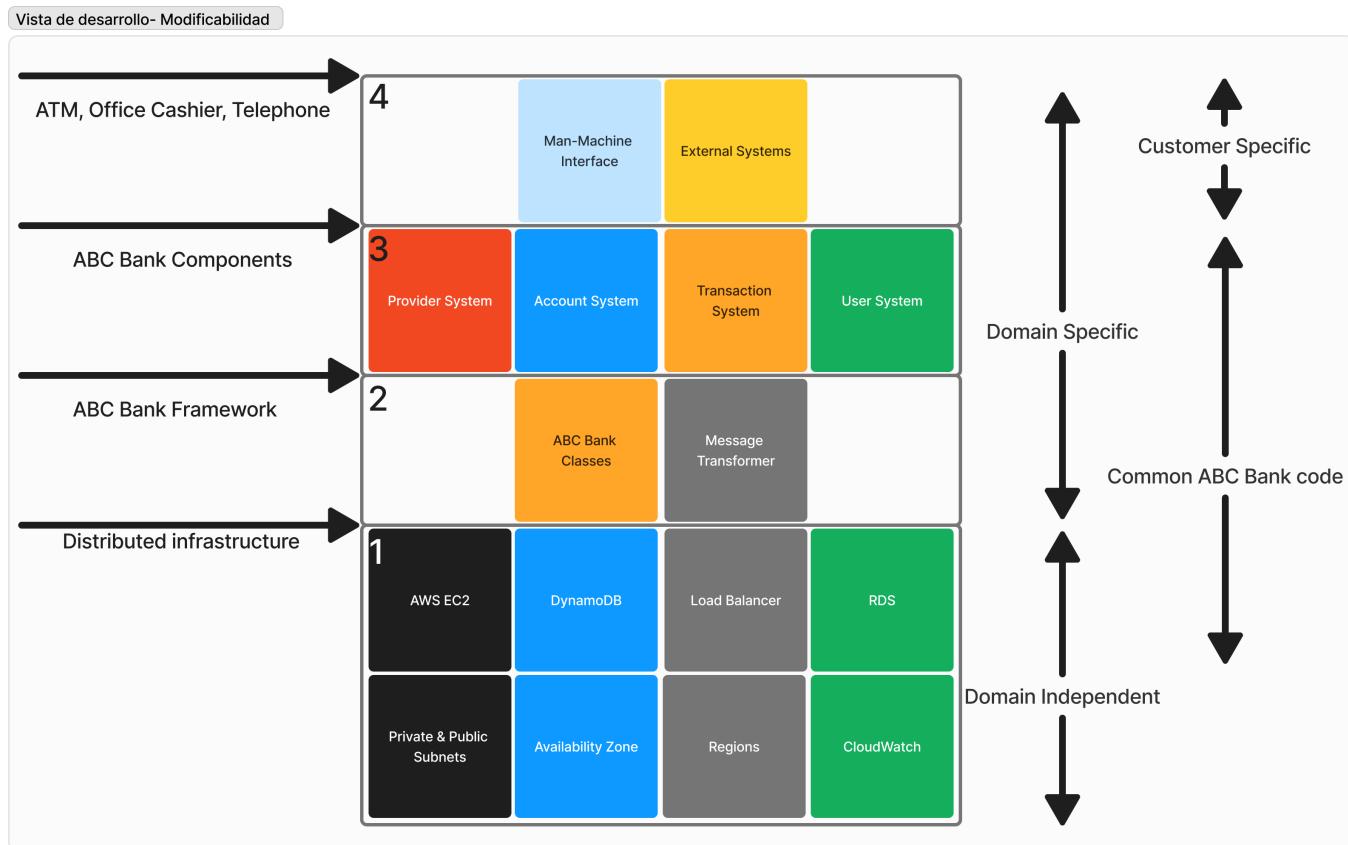
Fuente: Elaboración propia.

Figura 15: Vista de procesos. - escenario de modificabilidad



Fuente: Elaboración propia.

Figura 16: Vista de desarrollo. - escenario de modificabilidad



Fuente: Elaboración propia.

7. Trade Off

Los pollos hermanos han decidido abordar los requerimientos principalmente no funcionales como la Escalabilidad desde el enfoque de Arquitectura de Microservicios debido a que este nos permite concentrarnos en unidades específicas de responsabilidades e internamente separar componentes mediante *Layers*. Responsabilidades como la consulta de saldos, identificación de usuarios, pago de servicios y gestión de proveedores se comunicarán a través del *Gateway* como punto principal de entrada de peticiones. El *Gateway* contará con funcionalidades para medir el volumen de tráfico, en el escenario esperado manejará 7 TPS y en el escenario que se espera un volumen superior de hasta 11 TPS será el indicador que le permitirá determinar si debe empezar el proceso de escalamiento y enrutar las peticiones a los correspondientes microservicios. Si bien usar este enfoque requiere alta complejidad en la gestión de los microservicios, la escalabilidad y su despliegue independiente, permite tener un mejor manejo sobre los recursos, permite tener la información correspondiente a cada unidad de despliegue, lo que evita tener centralizado en un solo punto la información del sistema y usuario.

Algunos de los beneficios y costos pueden verse en la sección de ADR y vistas 4 + 1.

8. Patrones utilizados

Los patrones utilizados se pueden ver en los ADR y vistas 4 + 1.

9. Idempotencia y garantías de mensajería

9.1. Servicio de identificación de usuario

Es naturalmente idempotente, si recibe los mismos parámetros, el resultado será siempre el mismo y no habrán efectos secundarios que puedan cambiar el estado del sistema.

9.2. Servicio consulta de información de pago

Es idempotente, si recibe los mismos parámetros, el resultado será siempre el mismo y no habrán efectos secundarios que puedan cambiar el estado del sistema. El sistema de proveedores será el encargado de registrar en cache la información de los pagos para evitar latencia sobre las respuesta y evitar un poco probable error de los proveedores que desemboque en un fallo de la consulta.

9.3. Servicio de hacer una orden de pago

Aunque pueda parecer que naturalmente no es idempotente. Este servicio lo es. Por ejemplo, cuando una orden de pago se ha creado a través de la página web, si se envía una nueva orden de pago a través de la aplicación móvil, el sistema debería ser capaz de reconocer ambas y mantener la misma orden o transacción para las dos.

10. Alternativas de arquitectura

10.1. Arquitectura usando estilo *Serverless*

10.1.1. Vista lógica funcional

En la figura 17 se puede ver la vista lógica funcional para la alternativa de arquitectura utilizando *Serverless*. Se puede encontrar con mejor calidad [aquí](#).

10.1.2. Vista de procesos funcional

10.1.2.1. Identificación La figura 18 muestra el diagrama de procesos alternativo funcional para la identificación de un usuario. Se puede encontrar con mejor calidad [aquí](#).

10.1.2.2. Pago La figura 19 muestra el diagrama de procesos alternativo funcional para el pago de recibos públicos. Se puede encontrar con mejor calidad [aquí](#).

10.1.2.3. Registro de convenio / proveedor La figura 20 muestra el diagrama de procesos alternativo funcional para el registro de un convenio. Se puede encontrar con mejor calidad [aquí](#).

10.1.3. Vista física funcional

La figura 21 muestra el diagrama físico alternativo funcional. Se puede encontrar con mejor calidad [aquí](#).

10.1.4. Vista de desarrollo funcional

10.1.4.1. Identificación La figura 22 muestra el diagrama de desarrollo alternativo funcional para la identificación de un usuario. Se puede encontrar con mejor calidad [aquí](#).

10.1.4.2. Pago La figura 23 muestra el diagrama de desarrollo alternativo funcional para el pago de recibos públicos. Se puede encontrar con mejor calidad [aquí](#).

10.1.4.3. Registro de convenio / proveedor La figura 24 muestra el diagrama de desarrollo alternativo funcional para el registro de un convenio. Se puede encontrar con mejor calidad [aquí](#).

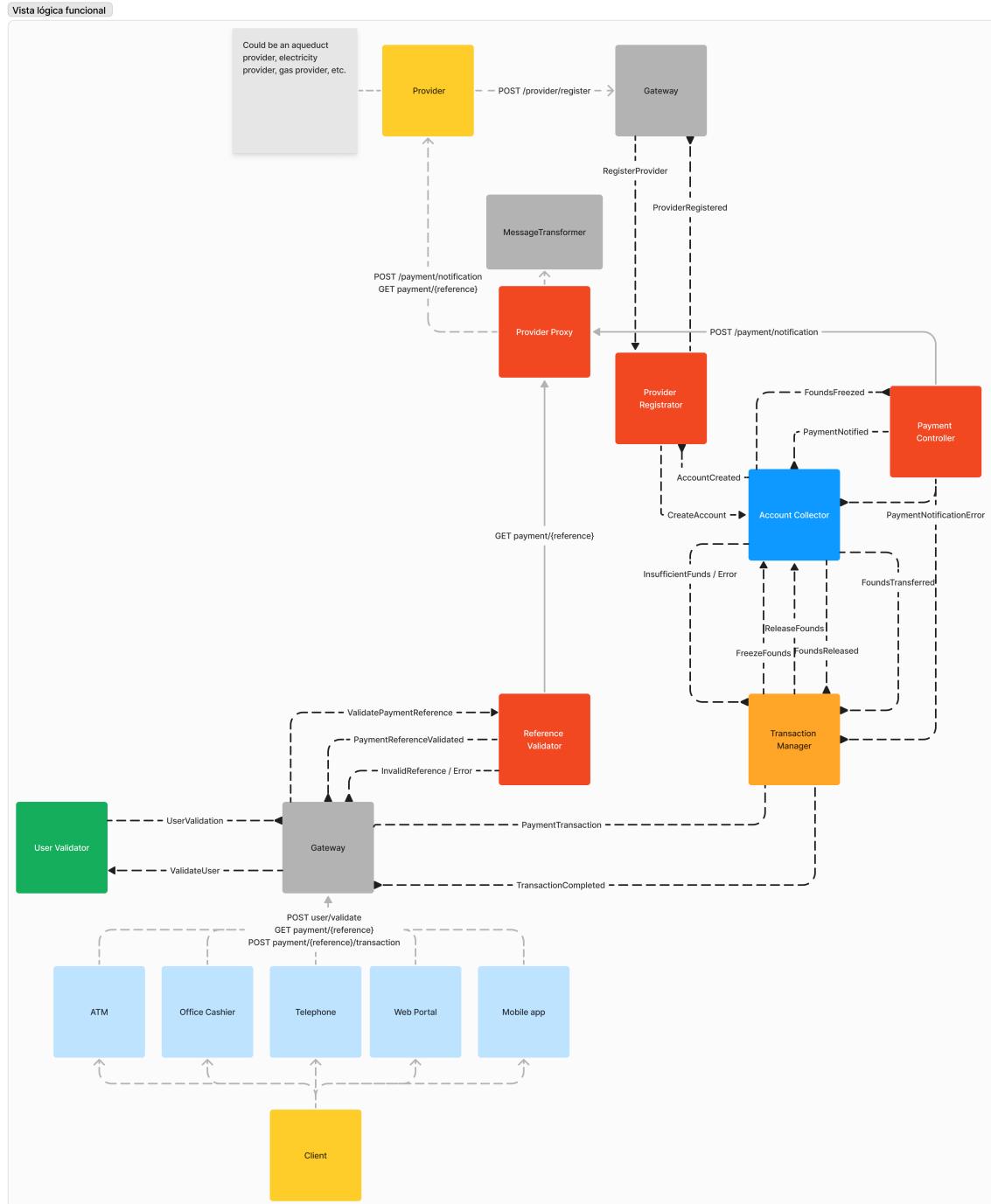
10.1.5. Casos de uso

La figura 25 muestra el diagrama de escenarios alternativo funcional para el registro de un convenio. Se puede encontrar con mejor calidad [aquí](#). También, se puede encontrar más información relacionada a los casos de uso [aquí](#).

10.1.6. ADR

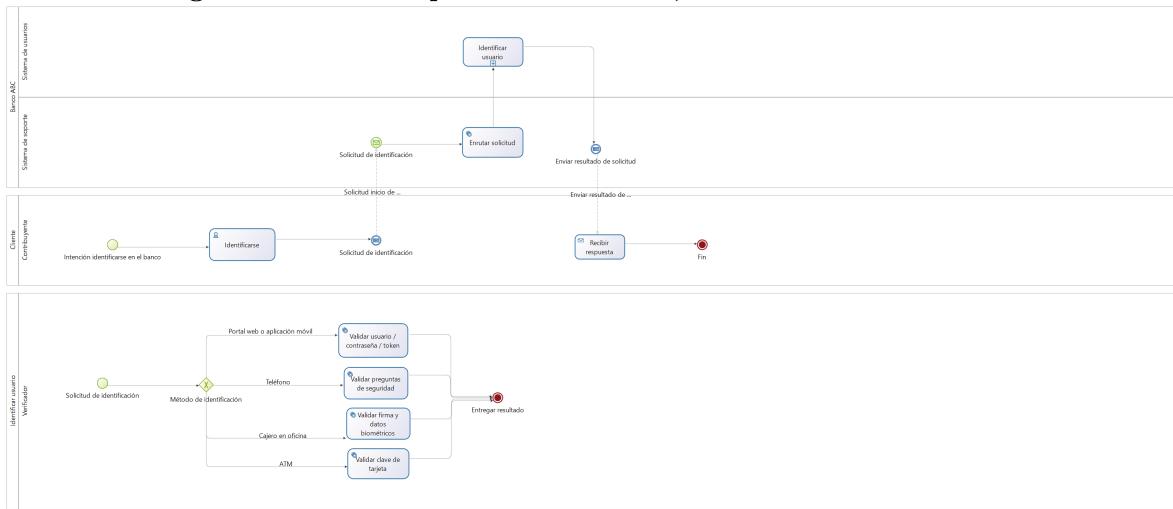
Los ADR funcionales para esta alternativa de arquitectura se pueden encontrar [aquí](#).

Figura 17: Vista Lógica Funcional.



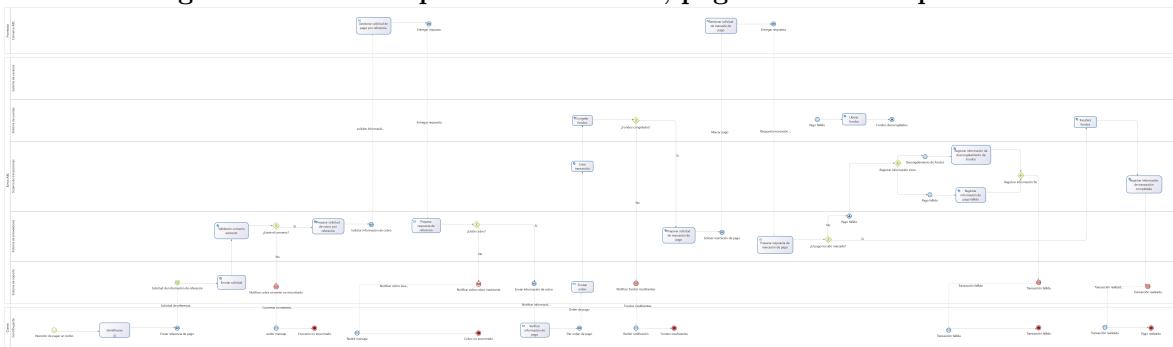
Fuente: Elaboración propia.

Figura 18: Vista de procesos funcionales, identificación de usuario.



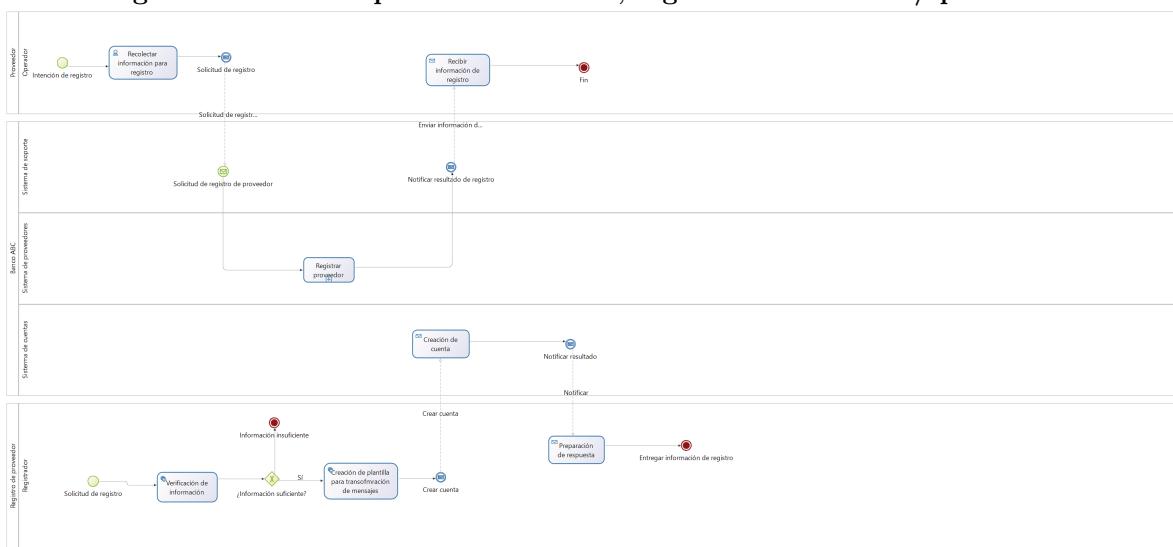
Fuente: Elaboración propia.

Figura 19: Vista de procesos funcionales, pago de servicios públicos.



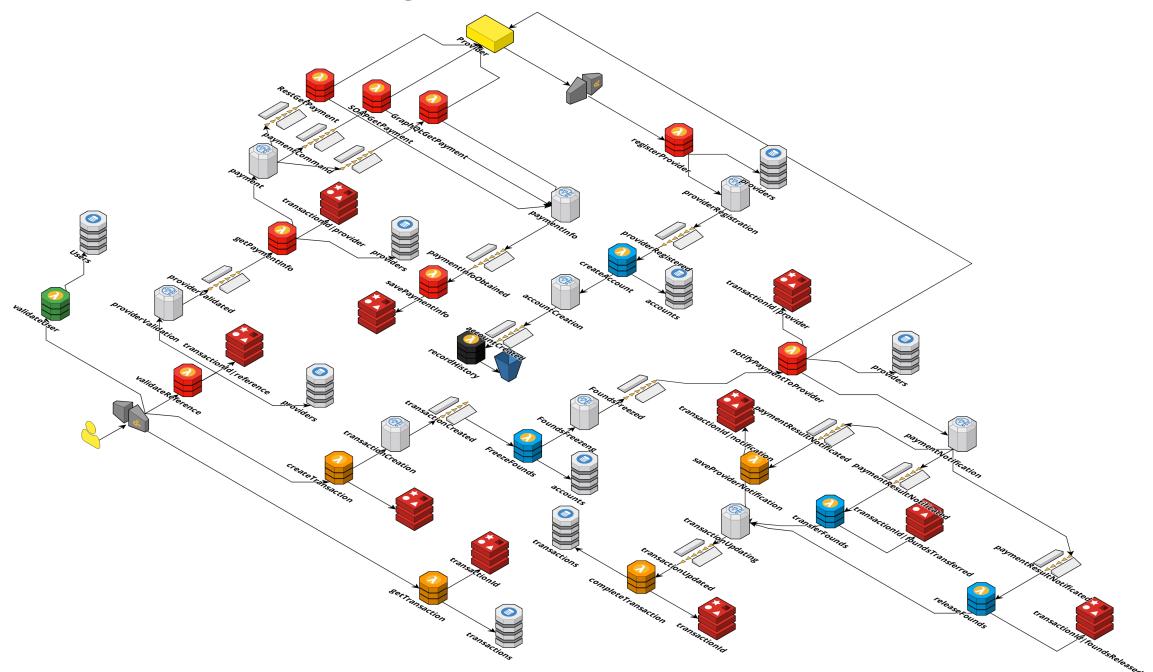
Fuente: Elaboración propia.

Figura 20: Vista de procesos funcionales, registro de convenio / proveedor.



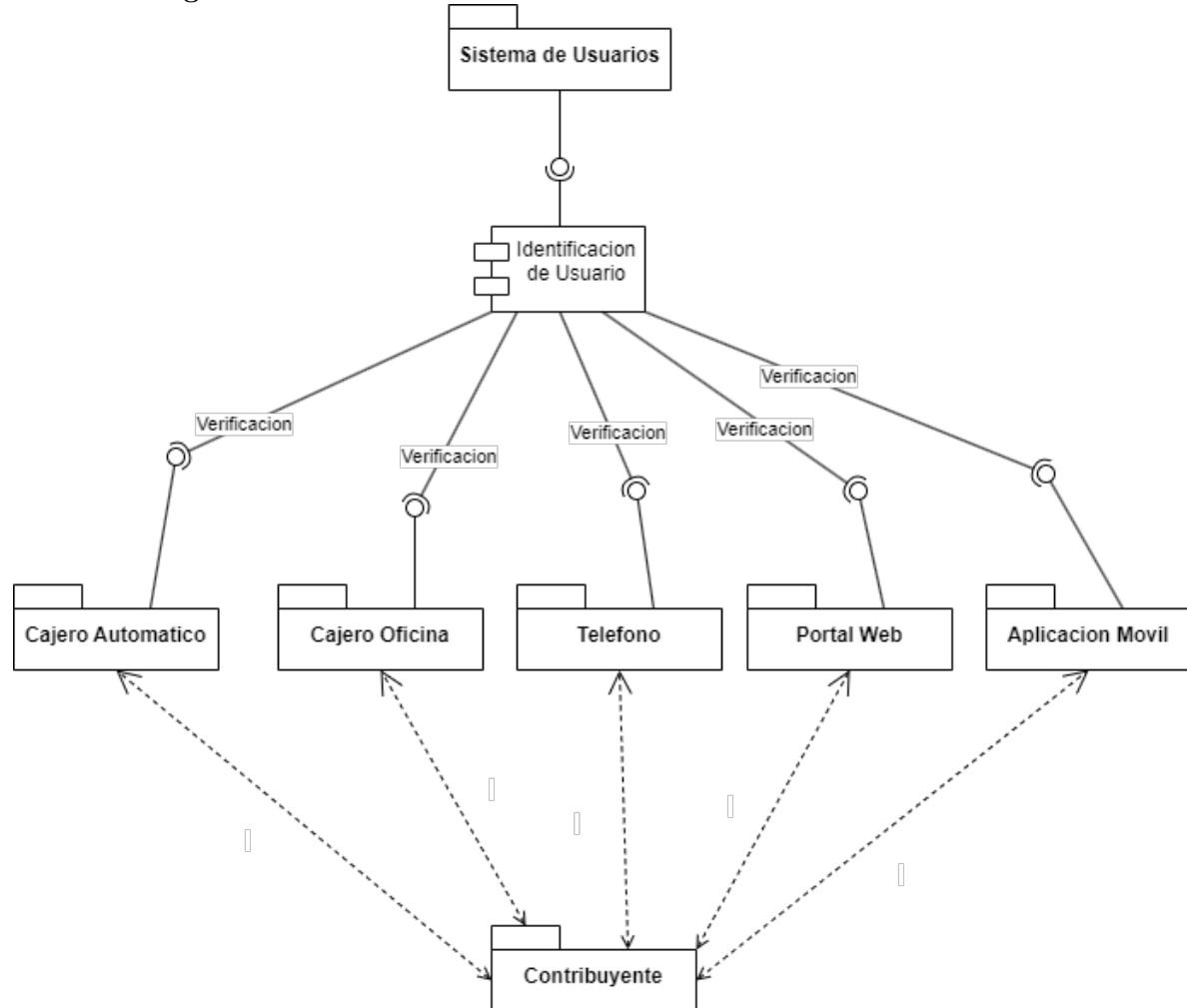
Fuente: Elaboración propia.

Figura 21: Vista física funcional.



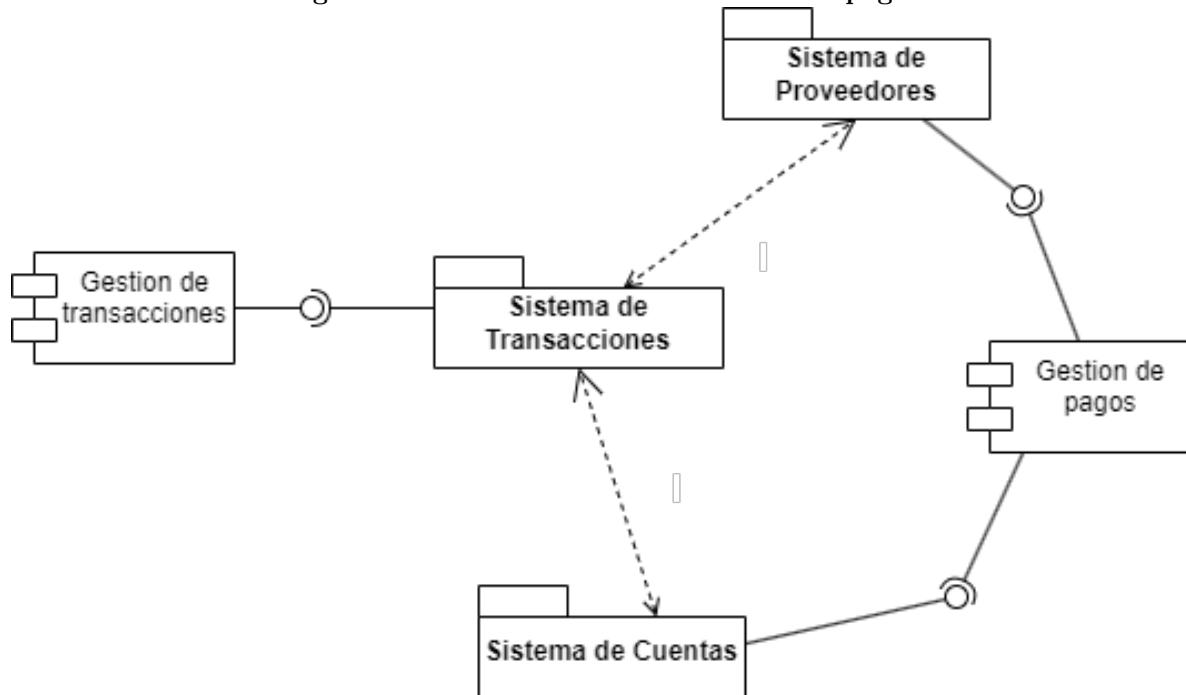
Fuente: Elaboración propia.

Figura 22: Vista de desarrollo funcional - identificación de usuario.



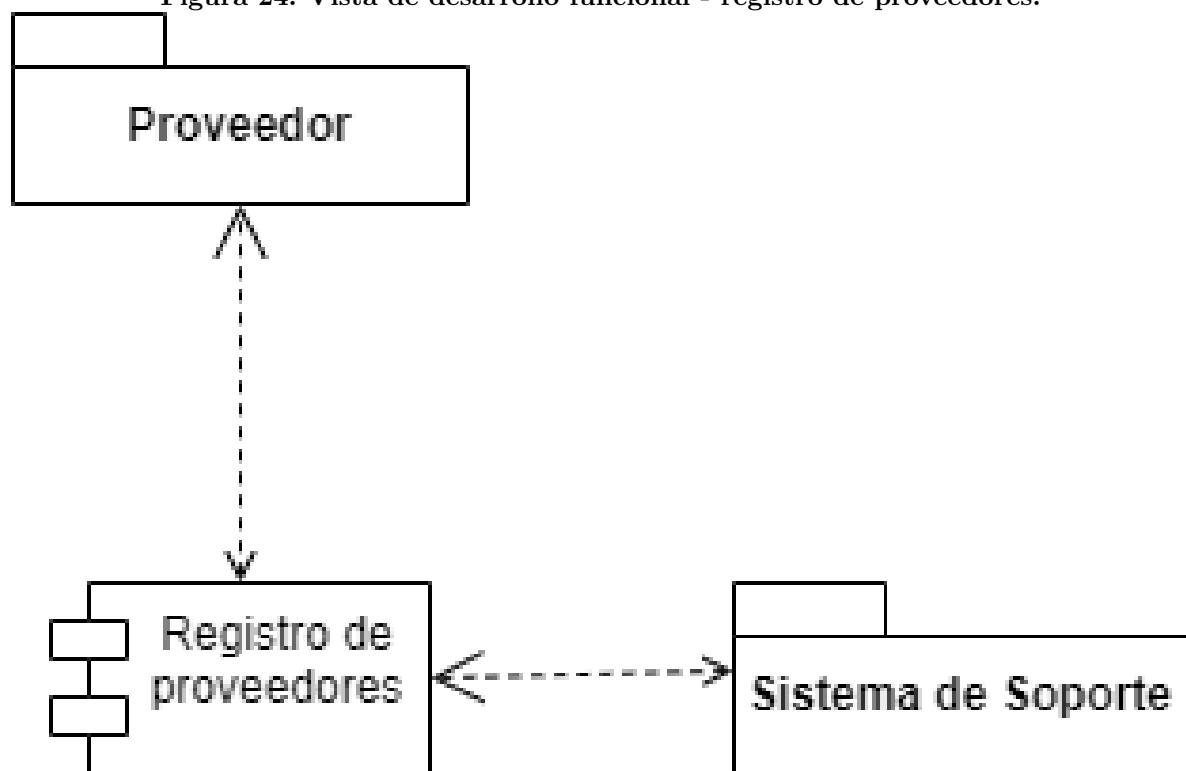
Fuente: Elaboración propia.

Figura 23: Vista de desarrollo funcional - pago.



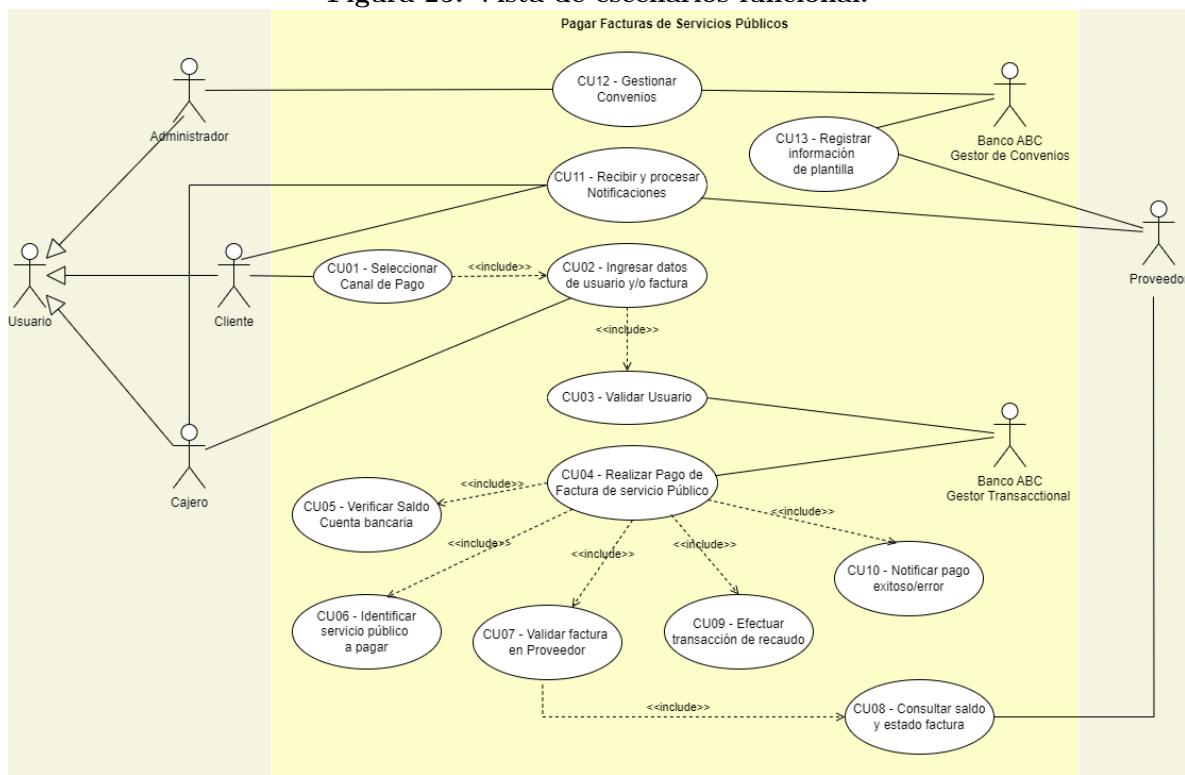
Fuente: Elaboración propia.

Figura 24: Vista de desarrollo funcional - registro de proveedores.



Fuente: Elaboración propia.

Figura 25: Vista de escenarios funcionales.



Fuente: Elaboración propia.

Referencias

- [1] G. A. Suarez, “preguntas,” 2024.
- [2] G. A. Suarez, “Parcial de patrones de arquitectura.”
- [3] DANE, “Boletín técnico encuesta nacional de calidad de vida (ecv) 2023,” 4 2024.
- [4] DANE, “Boletín técnico encuesta nacional de calidad de vida (ecv) 2020,” 9 2021.
- [5] B. de la República, “Reporte de sistemas de pago,” 2 2022.
- [6] B. de la República, “El efectivo sigue siendo el instrumento de pago preferido por el público y el comercio en colombia — banco de la república,” 2 2023.
- [7] B. de la República, “Resultados de la encuesta de percepción sobre el uso de los instrumentos para los pagos habituales en colombia - 2022,” 2 2023.
- [8] L. Repúblca and C. R. Salcedo, “Por los menos 14 % de los usuarios bancarizados ya opera solo con entidades digitales,” 1 2024.
- [9] A. O. Allen, *Probability, Statistics, and Queueing Theory: With Computer Science Applications*. 1990.
- [10] A. Engel, *Problem-Solving Strategies*. 1998.
- [11] AWS, “Measuring aws global accelerator performance and analyzing results — networking content delivery,” 2021.