

INTEGRACIÓN DE NUEVA EMBOZADORA EN EL FLUJO DE TARJETAS FÍSICAS

Nombre: Matías Meza Leyán

Docente: Nicolás Cenzano

Carrera: Ingeniería civil informática

Fecha: 11 de diciembre del 2023



Tabla de contenido

1.1 Resumen ejecutivo	3
1.2 Executive Summary	4
2. Introducción.....	5
2.1 La empresa.....	5
2.2 Contexto y problema.....	5
3. Objetivos y medidas de desempeño.....	7
3.1 Objetivo General	7
3.2 Objetivos específicos	8
3.3 Medidas de desempeño	8
4. Estado del Arte	9
4.1 Microservicios.....	9
4.2 Función serverless	10
4.3 JavaScript	10
4.4 TypeScript	10
5. Solución.....	11
5.1 Soluciones propuestas.....	11
5.2 Solución escogida	12
5.3 Plan de implementación.....	14
5.4 Desarrollo.....	16
5.5 Evaluación económica	18
5.6 Matriz de riesgo	18
6. Resultados.....	20
6.1 El proyecto	20
6.2 Desarrollo personal	20
6.3 Impacto	21
7. Conclusión.....	22
8. Referencias	23



1.1 Resumen ejecutivo

Dentro del desarrollo de las empresas chilenas, se está presentando una tendencia en el aumento de los productos financieros como aplicaciones, que facilitan la entrega de tarjetas de prepago o débito, junto a nuevas alternativas de plataformas de inversión. Mach es una de las empresas pioneras de Chile conocidas como Fintech o Neo Banco, y mediante este entorno altamente competitivo se encuentra en constante amenaza y necesidad de mejorar sus productos, sin perder el foco en su público objetivo e idea de que todos puedan acceder a productos bancarios.

Dado este mundo altamente competitivo es que se desarrollo está pasantía, enfocándose en el mejoramiento del flujo de solicitudes de tarjetas físicas dentro de Mach, eliminando el principal problema que se poseía al tener un solo punto de falla al contar con un solo proveedor de tarjetas físicas. Es por eso por lo que dentro de este ensayo se presentará el cómo se identificó ese problema, las diferentes alternativas de solución que se presentaron, la alternativa de solución que se terminó escogiendo, de tal forma que no se pierda la trazabilidad dentro del equipo de trabajo en el cual se trabajó, ya que este equipo es el encargado de las tarjetas y movimientos bancarios de los usuarios de Mach.

Esta solución se hizo con el enfoque de generar un impacto a largo plazo dentro de la empresa, haciendo posible el cambio de cualquier parámetro fácilmente en caso de que así se requiera en un futuro, o en caso de que se quiera incorporar aún más proveedores dentro del flujo de solicitudes de tarjetas físicas. Estando alineado también con los estándares actuales de las tecnologías y herramientas que se usan dentro de la empresa. Además, cabe destacar que dentro de esta solución el desarrollo que se hizo fue una investigación de que lo que se necesitaba para poder realizarla, se tuvo que aprender en profundidad arquitecturas y lenguajes de programación nuevos según se avanzaba en la creación e implementación de la solución.

También se presentará dentro de este informe el impacto económico que genera y beneficia a la compañía la implementación de un nuevo proveedor de tarjetas físicas, una matriz de riesgo con sus consecuencias y mitigaciones, y por último los resultados que se obtuvieron tras meses de trabajo con un equipo altamente competitivo aplicando la metodología ágil de Scrum.¹

¹ Marco de administración que los equipos utilizan para organizarse por cuenta propia y trabajar en aras de alcanzar un objetivo común (Fuente: <https://aws.amazon.com/es/what-is/scrum/>)



1.2 Executive Summary

Within the development of Chilean companies, there is a trend in the increase of financial products such as apps which facilitate the delivery of prepaid or debit cards, along with new alternative of investment platforms. Mach is one of the first Chilean companies known as Fintech or Neo Banco, and through this highly competitive environment is in constant threat and need to improve its product, without losing focus on its target audience and idea that everyone can access banking products.

Given this highly competitive world is that this internship was developed, focusing on improving the flow of physical card applications within Mach, eliminating the main problem of having a single point of failure by having a single supplier of physical cards. That is why in this essay we will present how this problem was identified, the different solution alternatives that were presented, the alternative solution that was finally chosen, so as not to lose the traceability within the team in which we worked, since this team is responsible for the cards and banking movements of Mach users.

This solution was developed with the focus of generating a long-term impact within the company, making it possible to change any parameter easily in case it is required in the future, or in case you want to incorporate even more suppliers into the flow of physical card requests. It is also aligned with the current standards of the technologies and tools used within the company. In addition, it should be noted that within this solution the development that was done was an investigation of what was needed to be able to realize it, it was necessary to learn in depth new architectures and programming languages as the creation and implementation of the solution progressed.

This report will also present the economic impact that the implementation of a new physical card provider generates and benefits the company, a risk matrix with its consequences and mitigations, and finally the results obtained after months of work with a highly competitive team applying the agile Scrum methodology.



2. Introducción

2.1 La empresa

MACH es un banco digital que combina la tecnología y el diseño centrado en las personas para crear productos financieros que todos puedan y quieran usar. Este nació en el año 2017 en el área de innovación del Banco BCI. La principal motivación que se tuvo fue crear un producto digital para todos los residentes en Chile que, al no estar bancarizados, no podían acceder a pagar servicios internacionales o comprar de forma online en el extranjero.

De esta forma empezó en el mercado con una tarjeta virtual prepago, creciendo en sus primeros tres años a más de dos millones de usuarios, formando parte de una de las innovaciones con mayor crecimiento de Latinoamérica.² En la actualidad cuenta con más de cuatro millones de usuarios, servicios de tanto tarjetas virtuales como físicas para cuentas prepago y cuentas corrientes, destacando que se encuentra en un proceso de migración de los usuarios de cuenta prepago a cuenta corriente, además de servicios como sistemas de inversión en fondos mutuos, pagos en cuotas, retiro de dinero en cajeros sin costos, pago a empresas en formato QR y envío de dinero al extranjero entre otros.

Mach al ser una empresa tecnológica cuenta con una gran variedad de equipos, estos dentro del contexto de la empresa tienen la denominación de BU (Business Units o Unidades de Negocio), dentro de éstas se encuentra la Business Unit de principalidad, cuyo fin es que los usuarios utilicen Mach como su cuenta bancaria principal. Dentro de esta unidad de negocio se encuentra el equipo de “tarjetas y movimientos”, que es el pilar responsable de la migración de usuarios de cuenta prepago a cuenta corriente y en el cual se desempeñará la pasantía.

2.2 Contexto y problema

Como se mencionó con anterioridad, dentro de la empresa se está llevando a cabo un proceso de migración, lo que está implicando hacer un cambio a todas las tarjetas físicas de cuenta prepago por tarjetas físicas de cuenta corriente, además, dada esta campaña de cuenta corriente

² Mach, (S/F). Somosmach.com. <https://www.somosmach.com/quienes-somos>



llegan en promedio más de cinco mil nuevas solicitudes de tarjeta física al día, lo que antes correspondía al promedio semanal, demostrando el éxito de la campaña. Dado todo este contexto que indica nuevo volumen de solicitud de tarjetas físicas, cabe profundizar en el área de fabricación, embozamiento y logística de todo esto. Desde la implementación de las tarjetas físicas hasta la actualidad, dentro de la empresa se ha contado con un único proveedor de tarjetas físicas, siendo esta la empresa la embozadora, que se ha encargado desde la fabricación a la logística de entrega de las tarjetas que recibe el usuario. Esta embozadora junto a Mach se encuentran en una comunicación constante mediante microservicios, de tal forma que el usuario recibe dentro de la aplicación un feedback que se encarga de mostrarle el estado de su tarjeta, esto desde que se inicia la solicitud hasta que el usuario la recibe y activa con la creación de su pin, siendo todo este proceso conocido como el flujo de tarjetas físicas.

Ahora bien, dado todo lo anteriormente mencionado es que podemos identificar un problema, y este se genera al poseer un único proveedor como embozadora, dado que en caso de cualquier problema que esta presente en la fabricación de las tarjetas, el usuario se verá afectado, dentro de la empresa el compromiso con el usuario es la entrega de la tarjeta física en un periodo máximo de 12 días hábiles desde que este la solicita.

Históricamente el proveedor actual dentro de los años que lleva trabajando junto con Mach ya ha presentado en más de una ocasión problemas que han impedido que el usuario reciba su tarjeta en el tiempo acordado. El primer incidente de este tipo fue en la época de pandemia, presentándose retrasos en la fabricación debido a faltas de materias primas para las tarjetas, de esta forma afectó en cadena al flujo y hubo retrasos de semanas. El segundo incidente importante se vivió en el año 2021, en el cual hubo problemas en un microservicio encargado de la generación del archivo CSV que se debe generar todos los días, este archivo es el que contiene la información de todas las solicitudes de tarjetas físicas que se realizaron en el día y este lo recibe el proveedor, sin embargo, dentro de esos ocho días no informaron que no se estaban recibiendo solicitudes, lo que terminó afectando entre tres mil a cinco mil usuarios. El tercer incidente es uno que se presentó recientemente, este consistió en que el proveedor tras procesar los archivos retornaba a Mach campos sin el formato que siempre se ha presentado, por lo que las solicitudes eran rechazadas por el servicio encargado de verificar la integridad del formato, esto debido a una actualización en el flujo del proveedor, sin verificar correctamente el que estuviera todo como corresponde.



En síntesis, se cuenta con una sola embozadora que históricamente ya ha presentado problemas tanto en la logística como en la fabricación, siendo este un único punto de falla, además, Mach se encuentra en un rubro sumamente competitivo, teniendo como grandes competidores a MercadoPago con más de dos millones seiscientos mil usuarios³ y Tenpo con más de dos millones doscientos mil usuarios⁴, por lo que no se deben tener problemas que afecten la reputación de la empresa.

Dado todo lo anterior es que surge la oportunidad del proyecto, que consiste en la integración de un nuevo proveedor al flujo de tarjetas físicas, robusteciendo el flujo para crear un sistema de alta disponibilidad para los usuarios, pudiendo recibir la mayor cantidad de solicitudes de tarjetas físicas que implicó la migración a cuenta corriente y entregarlas en el tiempo habitual a los usuarios, incluso si es que alguno de los proveedores tiene problemas, todo esto acompañado de que el usuario siga recibiendo dentro de la app el feedback del estado de su solicitud independiente del proveedor que le haya tocado para el embozamiento y entrega de su tarjeta. Todo esto actualizando los servicios que se encuentran desactualizado, de tal forma que queden alineados con los estándares actuales que se presentan en la empresa. Gracias a esto ya no se contará tan solo con un único punto de falla dentro del flujo de tarjetas físicas, disminuyendo el riesgo y posterior impacto reputacional de no cumplir con las expectativas del usuario.

3. Objetivos y medidas de desempeño

3.1 Objetivo General

Dada la metodología SMART (Específico, Medible, Alcanzable, Relevante, Duración Limitada) se define el siguiente objetivo general:

Eliminar el único punto de falla en el flujo de tarjetas físicas para convertirlo en

³ Osvaldo Gimenez. (2023, octubre 7). La Tercera.
<https://www.latercera.com/pulso/noticia/ceo-de-mercado-pago-adelanta-los-nuevos-productos-que-lanzaran-en-chile/YTEHMXMAPJFLPCKLDV2L7XG2TI/>

⁴ Fernando Araya. (2023, agosto 9). Forbes Chile.
<https://forbes.cl/negocios/2023-08-09/fernando-araya-de-tenpo-en-chile-hay-una-migracion-muy-grande-de-bancos-tradicionales-a-companias-como-tenpo>



un sistema de alta disponibilidad integrando un nuevo proveedor de tarjetas físicas antes del Q4 de 2023.

3.2 Objetivos específicos

Para cumplir de manera adecuada con el objetivo general se derivan los siguientes objetivos específicos:

- Integrar un nuevo proveedor que funcione en paralelo al actual.
- Levantamiento de la documentación para visibilizar los flujos actuales y habilitar a mejoras.
- Robustecer el stack tecnológico de las infraestructuras de procesos legacy.

3.3 Medidas de desempeño

Para poder medir el éxito de este proyecto se evaluará según los siguientes KPI relacionados cada uno a los objetivos específicos del proyecto:

KPI para determinar la integración del nuevo proveedor:

$$\text{Porcentaje de objetivos alcanzados} = \frac{\text{Número de story points}^5 \text{ completados}}{\text{Número total de story points planificados}} * 100$$

KPI para medir la calidad e integridad de la documentación:

Evaluación de documentación

$$= \frac{\text{Puntuación de proceso} + \text{Puntuación de integración} + \text{Puntuación de discovery}}{3}$$

Por último, para el objetivo de robustecer el stack tecnológico de la infraestructura de procesos legacy se definieron tres medidas de desempeño:

Mejora en tasa de falla

$$= \frac{\text{Tasa de falla tras cambio} - \text{Tasa de falla antes del cambio}}{\text{Tasa de falla antes del cambio}} * 100$$

⁵ Los puntos de historia son una forma de calcular la cantidad de esfuerzo necesario para completar una historia de usuario en la lista de trabajo pendiente del producto. (Fuente: <https://asana.com/es/resources/story-points>)



Mejora en uso de recursos

$$= \frac{\text{Uso de recursos tras cambio} - \text{Uso de recursos antes del cambio}}{\text{Uso de recursos antes del cambio}} * 100$$

Mejora de número de vulnerabilidades

$$= \left(\frac{\text{Número de vulnerabilidades tras cambio} - \text{Número de vulnerabilidades antes del cambio}}{\text{Número de vulnerabilidades antes del cambio}} \right) * 100$$

4. Estado del Arte

Para determinar las herramientas óptimas para poder hacer la implementación se realizó una investigación de diferentes elementos que podían ser utilizados, dentro de estos destacan los servicios en nube como los microservicios, de los cuales se cuenta ya con una gran cantidad dentro de la empresa, sin embargo, estos poseen un mayor costo que otras alternativas que también se encuentran dentro de la nube, esta alternativa más barata es conocida como función serverless.

Además de esto, hay que considerar cual es el mejor lenguaje de programación posible para la realización de este proyecto, dentro de la gama de posibilidades se encontraban dos alternativas que eran viables y se utilizaban ya dentro de la empresa, a la vez que son populares en la actualidad, una de estas es JavaScript, que es un lenguaje orientado a objetos y por otro lado se encuentra TypeScript, que forma parte del nuevo estándar que se está implementando dentro de Mach, además, cabe destacar que recientemente se aprobó su uso para poder ser utilizado como lenguaje base de una lambda dentro de la empresa.

4.1 Microservicios

Ahora bien, entrando en profundidad dentro de lo que son los microservicios, estos son un tipo de arquitectura de software que divide una aplicación en un conjunto de servicios independientes, donde cada uno de estos es responsable de una tarea o función. Dentro de sus ventajas destaca su escalabilidad, su facilidad de mantener y depurar, y su flexibilidad y adaptación. Dentro de sus desventajas destaca que puede ser complejo de implementar, junto con sus costos de



implementación y mantención. Dentro de los grandes proveedores de servidores para microservicios destacan Amazon Web Services, Microsoft Azure y Google Cloud Platform.⁶

4.2 Función serverless

Estas corresponden a un tipo de función sin servidor que se ejecuta en la nube, es decir, son un servicio informático que permite ejecutar código sin aprovisionar ni administrar servidores. El servicio de funciones serverless ejecuta la función solo cuando es necesario y escala automáticamente según sea requerido, además de que solo se paga por el tiempo de uso, reduciendo los costos en contraste a un microservicio, que poseen un gasto mínimo garantizado. Dentro de sus ventajas destaca que es más fácil de leer, posee un mejor uso de recursos acompañado de menores costos. Al igual que como con los microservicios hay proveedores que se destacan por ofrecer servicios de funciones sin servidor en sus nubes, estás son AWS Lambda, Azure Functions, Google Cloud Functions e IBM Cloud Functions entre otras.⁷ Cabe destacar que, dentro de este informe, dado el lenguaje que se aplica dentro de la empresa, a las funciones serverless se les llamará lambdas, ya que pertenecen a Amazon Web Services.

4.3 JavaScript

Este es un lenguaje de programación orientado a objetos que se utiliza principalmente para el desarrollo web, este cuenta con la característica de ser un lenguaje dinámico, lo que significa que los tipos de datos se determinan en tiempo de ejecución. Dentro de sus ventajas destacan que su poca complejidad, su versatilidad y que al ser popular hay mucha documentación al respecto. Dentro de sus desventajas destaca que el hecho de que sus datos se determinen en tiempo de ejecución puede dificultar la detección de errores y su menor eficiencia frente al tipado estático, lo que dificulta la comprensión del código a la hora de trabajar en conjunto con otras personas.⁸

4.4 TypeScript

⁶ Amazon. (S/F) aws.amazon.com. <https://aws.amazon.com/es/microservices/>

⁷ Amazon. (S/F). docs.aws.amazon.com. https://docs.aws.amazon.com/es_es/lambda/latest/dg/welcome.html

⁸ Mozilla. (2023, julio 24). developer.mozilla.com. <https://developer.mozilla.org/es/docs/Web/JavaScript>



Este corresponde a un lenguaje de programación de tipado estático basado en JavaScript, que agrega tipos de datos, clases y otras características de los lenguajes de programación orientados a objetos. Dentro de sus ventajas destacan que permite detectar errores a la hora de compilar, mejorando la calidad del código, facilita la colaboración al todos entender el tipo de datos con los que se está trabajando. Dentro de sus desventajas destacan que es un poco más complejo de JavaScript, y que puede aumentar el tamaño del código al requerir declarar los tipos.⁹

5. Solución

5.1 Soluciones propuestas

Tras el análisis del estado del arte, y también tras leer y estudiar la documentación que posee sobre sus propios servicios Mach, se vio que dentro de la empresa ya habían visibilizado el problema, y habían determinado dos propuestas de solución para la implementación del nuevo proveedor y mitigar ese único punto de falla dentro del flujo de tarjetas físicas.

La primera propuesta de solución que se tenía propuesta era el hacer una etiqueta que determine antes de generar el archivo de solicitudes a que proveedor debe ir esa solicitud, de tal forma que la mayoría del trabajo recaería en modificar el microservicio de logísticas, agregando ese nuevo campo y haciendo que lo reciba el proveedor, la generación del archivo la hace otro equipo de operaciones bancarias, y ellos también deberían modificar algunas cosas de tal forma que hagan entrega del archivo según a que proveedor corresponda. La principal ventaja de esta propuesta es que se mantiene trazabilidad por parte del equipo de tarjetas y movimientos de que solicitud va vinculada a cada proveedor.

La segunda propuesta de solución que tenía ya propuesta la empresa contemplaba que el equipo de operaciones se encargase de hacer la distinción de proveedor al momento de generar el archivo, teniendo solo que modificar la lambda actual agregando un nuevo flujo para que emita los estados que reciba de las solicitudes del nuevo proveedor.

La tercera propuesta fue la que yo le planteé a la empresa, dada toda la investigación que se hizo, una extensa cantidad de reuniones con el equipo refinando la idea y la aprobación del líder técnico del equipo. Esta consiste en lo siguiente, hacer dos nuevas lambdas en TypeScript, en la cual

⁹ Typescript. Lang (S/F). typescriptlang.org.
<https://www.typescriptlang.org/docs/handbook/typescript-from-scratch.html>



la primera lambda se encarga de recibir el archivo completo ya generado por el equipo de operaciones, y su única función es dividir en dos archivos diferentes según la proporción que está contractualmente acordada con los proveedores, de un 70% para el antiguo proveedor y un 30% para el nuevo proveedor, la ventaja de hacer esta lambda es que al igual que en la primera propuesta se mantiene la trazabilidad sobre a que proveedor se le asigna cada solicitud, al estar en una lambda es más fácil de editar al encargarse solo de eso, por lo que si en un futuro se debe cambiar un parámetro o se agrega un nuevo proveedor se tiene que modificar muy poco dentro del flujo, además, esta lambda no presentaría la mayor desventaja que posee la primera propuesta, ya que al en el microservicio se tenía que estar constantemente comprobando que se cumpla con la proporción 70/30, lo que implicaría un gasto excesivo de recursos y dinero, sin embargo con esta primera lambda tan solo se ejecutaría una vez al día la división, reduciendo al mínimo el gasto de recursos. La segunda lambda que contempla esta propuesta de solución consiste en recibir cada uno de los archivos de solicitud generados por la lambda anterior, encargándose de comprobar la integridad y formato de cada uno de los campos que vienen dentro del archivo de solicitudes, discriminando según el nombre de cada archivo a que proveedor pertenece y haciendo que con esto siga el flujo que corresponde hasta la publicación de los estados de la tarjeta que recibe el usuario en la aplicación. Dentro de las ventajas de esta segunda lambda se encuentran el que se está reemplazando y robusteciendo la lambda ya existente que se encarga de hacer emitir los estados del proveedor antiguo, ya que esa lambda antigua cuenta con más de cuatro años sin modificaciones, además de estar hecha en JavaScript, siendo este un lenguaje que se está intentando migrar a TypeScript dentro de la compañía al este ser su nuevo estándar, además se realizará con el enfoque de hacerla a prueba de cambios a futuro en caso de que se requiera incluir un tercer proveedor al flujo.

5.2 Solución escogida

Tras el análisis del estado del arte, y también tras una consulta junto al equipo de cuál de las tres propuestas correspondía a la mejor dirección que podía tomar el proyecto, se llegó a la conclusión de que la tercera propuesta, que básicamente y tal como se mencionó con anterioridad, corresponde a la realización de dos lambdas en TypeScript, de las cuales la función de la primera de éstas sería recibir un archivo CSV que contiene todas las solicitudes de tarjetas físicas del día, para posteriormente dividirla en dos archivos CSV diferentes, en los cuales el 70% de las solicitudes irán al proveedor ya existente, mientras que el 30% restante irán al nuevo proveedor, para



posteriormente depositar esos CSV generados en dos carpetas contenedoras en la nube que gatillarían el segundo lambda propuesto.

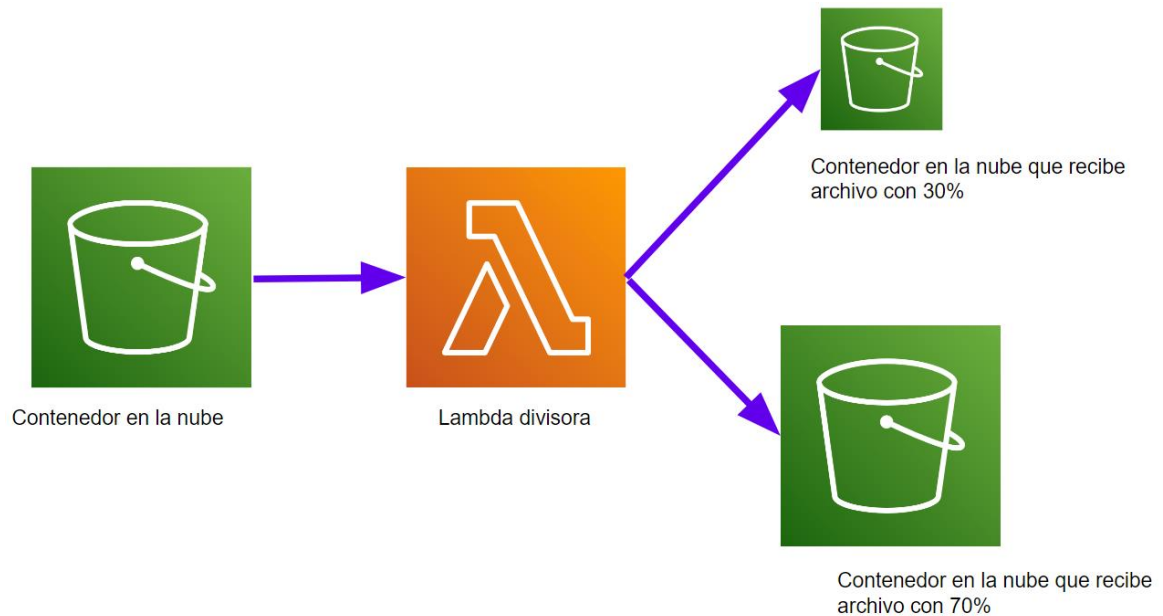


Ilustración 1. Elaboración propia. Correspondiente a diagrama de lambda divisora.

Esta segunda lambda, distinguiría según la carpeta de la que recibe el archivo el flujo que debe tomar, primero verificando la integridad del archivo, es decir, que cada columna corresponda al formato exigido al proveedor, que no se presenten filas en blanco, que no falte ningún campo requerido según el formato, que tenga toda la información esperada y posteriormente, tras esa verificación, según el prefijo en el nombre del archivo, se señalaría su estado (Si la solicitud fue recibida, si está en fabricación, si la tarjeta está en camino, etc.) mediante mensajes SNS¹⁰ le seguirá informando al microservicio tal como se hace hoy en día el estado de la solicitud de la tarjeta física como feedback al usuario dentro de la aplicación.

¹⁰ Simple Notification Service (Amazon SNS) es un servicio administrado con el que se ofrece la entrega de mensajes de los publicadores a los suscriptores (también conocido como *productores y consumidores*). (Fuente: https://docs.aws.amazon.com/es_es/sns/latest/dg/welcome.html)

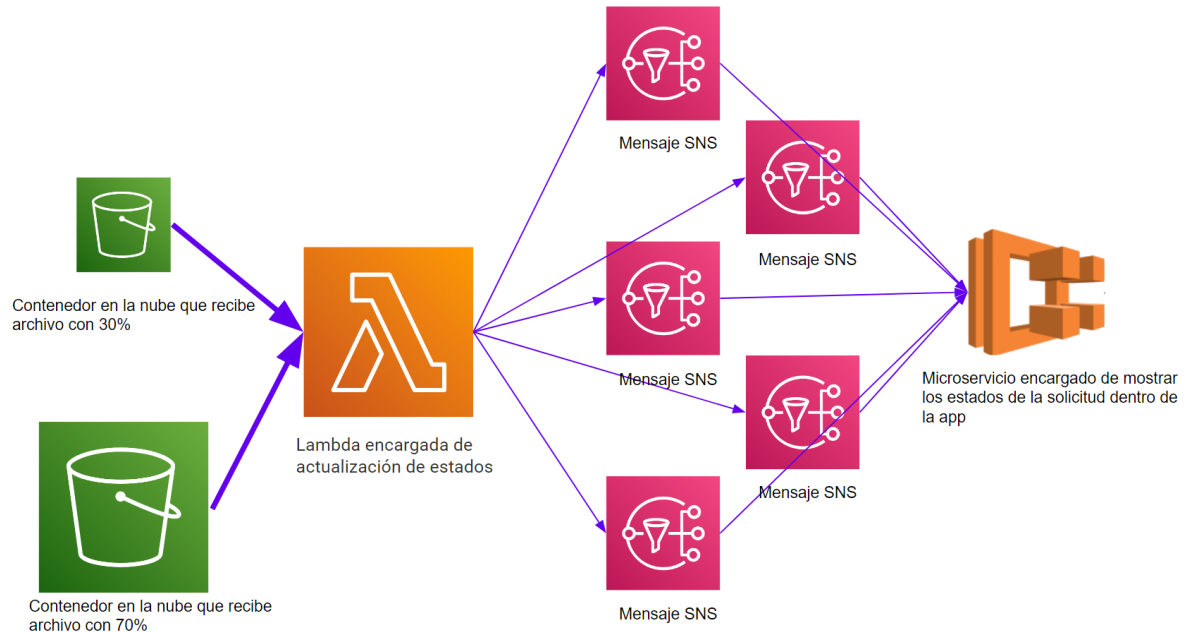


Ilustración 2. Elaboración propia. Correspondiente a diagrama de lambda encargada de actualización de estados

5.3 Plan de implementación

Para llevar a cabo este proyecto, se trabajó mediante la metodología ágil de Scrum, dado que esta corresponde al estándar que se maneja dentro de Mach para todo trabajo o proyecto que se quiera realizar, dentro de los marcos de esta metodología se hizo un plan de implementación para el proyecto. Primero que nada, se tuvo una reunión con el Product Owner, que es la persona que representa los intereses del proyecto dentro del equipo de tarjetas y movimiento, este presentó un cronograma de seis sprints (cada sprint corresponde a dos semanas de trabajo) dentro de las cuales tendríamos que investigar, refinar, desarrollar e implementar todo para que antes del 18 de Diciembre ya se pueda contar con un plan piloto dentro de la empresa, y posteriormente el 3 de Enero de 2024 ya esté disponible el nuevo proveedor para todos los usuarios que soliciten tarjetas físicas. Dentro de este cronograma de seis sprints se destaca lo siguiente que correspondería al plan de implementación.

La primera etapa, correspondió a una reunión con la nueva embozadora, dentro de esta se definieron los aspectos más técnicos, como el tipo de protocolo de transferencia de archivos que se tendría entre esta y Mach, los tiempos en los que esperamos tener ya en producción y funcionando



el nuevo proveedor para todos los usuarios, se definieron reuniones cada dos semanas dentro de las cuales se presentarían los avances y dificultades si es que llegasen a surgir, para así poder solucionarlas lo antes posible.

La siguiente etapa del plan se define como la etapa de investigación, en esta etapa fue que se decidió la mejor solución a implementar dentro de las responsabilidades que recaían en Mach para el desarrollo, en esta etapa se hizo un estudio del estado del arte, se tuvieron extensas reuniones de refinamiento de las posibles soluciones con diferentes personas dentro de la empresa, dentro de las cuales participaron los otros desarrolladores del equipo a pesar de no estar involucrados directamente en el desarrollo, el Tech Lead que nos orientó técnicamente sobre todo respecto a uso de recursos y efectividad de las herramientas que se terminaron usando para la solución escogida que se mencionó en los puntos anteriores. También se realizó una investigación y documentación de la solución, estudiando el flujo de solicitudes de tarjetas físicas actual para de esta forma determinar que parte del flujo es la que correspondía a mayores reestructuraciones para la implementación de la solución.

Posterior a esto se ingresó en la etapa más extensa del proyecto, que correspondió al desarrollo de la solución, al este proyecto presentar el desarrollo de dos lambdas en el lenguaje de programación de TypeScript, algo inédito y nuevo dentro de la empresa, en un principio se presentaron muchas fricciones, sin embargo, a medida que se iba desarrollando cada vez las fricciones con esta novedad fueron disminuyendo, llegando a los resultados esperados y terminando dentro de los tiempos propuestos el desarrollo.

Tras esta etapa, es decir, una vez concluido el desarrollo llegó la etapa de pruebas, en las cuales se comprobó mediante diferentes casos, ya sean casos esperados o casos bordes los diferentes comportamientos que presentaban las lambdas, viendo si arrojaban mensajes de errores cuando correspondía y demostrando de esta forma su correcto funcionamiento. Estos tests primero se hicieron de manera local, y posteriormente se hicieron las pruebas en los diferentes ambientes de prueba antes de ingresar a producción. Cabe destacar que, hoy en día esta etapa ya ha sido concluida exitosamente y ambas lambdas se encuentran en producción.

La quinta etapa del plan de implementación corresponde a una prueba de los tiempos de respuesta en un entorno de solicitud real para el nuevo proveedor, esto se hace con la implementación ya en producción, pero enviando manualmente al proveedor las solicitudes de



tarjetas físicas correspondientes a “Friends and Family” de la empresa, para de esta forma ver que se cumpla con los SLA esperados. Esta etapa se encuentra ad portas de suceder ya que como se señaló anteriormente será este 18 de diciembre.

La sexta y última etapa del plan de implementación corresponde a habilitar a todos los usuarios el nuevo proveedor, es decir que todos los días, el 30% de las solicitudes de tarjeta física estén a cargo de la nueva embozadora, esto sucederá el día 3 de enero del siguiente año tal como se mencionó anteriormente, dando fin al proyecto.

5.4 Desarrollo

Ahora bien, presentado ya en su mayoría el proyecto, esta parte se centrará en explicar que fue lo que se aplicó y realizó personalmente dentro del proyecto. Primero que nada, se ingresó a la empresa y mediante un periodo de dos semanas se estudió en profundidad mediante un sistema estandarizado para toda nueva incorporación, en la que se presentan todas las herramientas y conceptos de la empresa que un software engineer debe conocer. Tras esto se fue poco a poco entrando en detalle y estudiando la arquitectura y microservicios que están directamente a cargo del equipo de tarjetas y movimientos. Esta fue una etapa de mucha fricción, ya que los lenguajes de JavaScript y TypeScript eran muy poco familiares y se fue estudiando en paralelo al desarrollo. Gracias a la universidad se entendían los principios y lógica detrás de los lenguajes de programación, por lo que tras un par de semanas de adaptación se lograron dominar el nivel requerido por parte de la empresa para el desarrollo.

Tras esa primera etapa de desafíos, se empezó a trabajar plenamente en el desarrollo del proyecto, primero se leyó una documentación ya realizada con anterioridad por el equipo, ya que se requería entender el estudio tras lo que se quería hacer y las alternativas de solución a la que se habían llegado, para posteriormente encargarme el desarrollar una nueva propuesta en base a lo ya existente. Es por esto por lo que, tras un estudio de las lambdas, los microservicios, consultas a diferentes personas dentro de la empresa para obtener su punto de vista y reuniones constantes de ideas, se llegó a la solución propuesta que se desarrolló. Teniendo esta que ser documentada y



presentada por mi parte al equipo de desarrolladores backend, Tech Lead¹¹ y Sem¹² de tarjetas y movimientos, cabe destacar que en esta etapa es que se tuvieron que desarrollar los diagramas de la ilustración 1 e ilustración 2.

Después de eso, se empezó a trabajar netamente en el desarrollo de las lambdas, debiendo tener comunicación con el equipo de cloud¹³ para que habilitaran en la nube de la empresa los contenedores necesarios para la subida de archivos, las lambdas, etc. Una vez ya se habilitó un puerto al que enlazar las lambdas se comenzó a trabajar. Por mi parte fui el responsable de migrar toda la lógica de la lambda del proveedor antiguo que se encargaba de la validación de la integridad de las solicitudes de tarjetas físicas, además del enviar mensajes SNS al microservicio de logísticas, a la nueva que se encontraba en TypeScript, por lo que debí modificar muchos archivos, crear los tipos de las variables, además de desarrollar test unitarios¹⁴ ya que las funciones de esta lambda antigua no los poseía, a pesar del estándar actual de la empresa ser que todas las funciones tengan sus respectivos test.

Posteriormente, a este nuevo flujo migrado se le añadió lo necesario para que utilizando el mismo código y solo añadiendo la identificación del proveedor mediante una etiqueta se determine el como se deben mandar los mensajes SNS. Tras esto se debieron crear nuevos tests unitarios probando diferentes casos bordes y que el flujo no tenga ningún problema para el nuevo proveedor. Por lo que, una vez aprobado, se hicieron pruebas en ambientes de staging y terminó en un paso a producción.

Una vez se encontró esa lambda en producción y otra persona del equipo se encargó del desarrollo de la lambda divisora, realicé los cambios necesarios en el microservicio de logísticas, para que identifique el mensaje SNS del nuevo proveedor y siga ejecutándose el feedback visual en la aplicación.

¹¹ Un technical leader es un desarrollador de software que tiene a su mando un equipo completo de desarrollo. Cuenta con la responsabilidad de que los productos que se lleven a cabo cuenten con la calidad técnica correspondiente (Fuente: <https://www.inesdi.com/blog/tech-lead-funciones-salario/>)

¹² Un Software Engineer Manager es el responsable de alinear a sus equipos, a la tecnología y a los procesos productivos. (Fuente: <https://ericmora.me/organigrama-digital/que-hace-un-software-engineering-manager/>)

¹³ Dentro de la empresa es el equipo encargado de todos los permisos y desarrollos que se pueden hacer en la nube.

¹⁴ Las pruebas unitarias son el proceso en el que se prueba la unidad funcional de código más pequeña. Las pruebas de software ayudan a garantizar la calidad del código y son una parte integral del desarrollo de software. (Fuente: <https://aws.amazon.com/es/what-is/unit-testing/>)



5.5 Evaluación económica

Ya que no se puede dar ningún dato real de los costos que se manejan con cada proveedor, se creó una moneda ficticia denominada “M” y una cantidad arbitraria de solicitudes para poder hacer operaciones con los costos y demostrar los beneficios económicos de la implementación del nuevo proveedor.

Considerando que actualmente el costo de fabricación son 7791 M, de delivery 26372 M y custodia de 2880 M, se asciende a un total de 37043 M como costo final de la producción de esa cantidad arbitraria de tarjetas físicas. Tras la implementación del proyecto, considerando que un 70% quedaría a cargo del proveedor antiguo y un 30% a cargo del proveedor nuevo, los nuevos costos serían de 7004 M la fabricación, 21044 M el delivery y 3076 M la custodia, ascendiendo a un total de 31124 M, lo que implicaría una reducción de un 17,3% de los costos totales. Por otro lado, con una mirada aún más a futuro, en el cual tras una exitosa implementación y piloto se toma la decisión de aumentar a un 50% la carga del nuevo proveedor y disminuyendo a un 50% el antiguo proveedor, tenemos que los costos de fabricación serían de 6480 M, delivery 17492 M y custodia de 3207 M, ascendiendo a un total de 27179 M, implicando una disminución de un 30,7% de los costos comparándolo con lo que cuesta en la actualidad.

Cabe destacar que los costos totales del nuevo proveedor son un 53,25% más económicos que los costos actuales del proveedor actual, y al agregarlo al flujo y ya no ser 100% dependientes del proveedor antiguo, se genera un poder de negociación de disminuir los costos de este último.

Además, se debe señalar también que el desarrollo y éxito de este proyecto impactó directamente en las proyecciones presupuestarias del año 2024 de la empresa, demostrando la relevancia de este proyecto.

5.6 Matriz de riesgo

Como en todo proyecto existen riesgos asociados y diferentes formas de mitigarlos, tras un estudio de los posibles riesgos que pueden surgir se destacan los siguientes:



Riesgo	Probabilidad	Consecuencias	Impacto	Mitigación
Fallo de la empresa generadora de números de las tarjetas.	Posible que suceda, ya que se sabe dentro de la que presenta problemas semanalmente.	Moderada, ya que puede implicar un atraso en los SLA para la entrega de tarjetas físicas.	Medio, ya que solo se generaría un impacto reputacional si es que se llegase a no cumplir con los SLA de entrega.	Comunicarse con la empresa para que se informe el problema y se nos diga en cuanto tiempo aproximadamente se solucionará
Hackeo al archivo de solicitudes físicas al no estar encriptado, sin embargo, el protocolo de transferencia si lo está.	Muy poco probable, Mach jamás a presentado un hackeo o filtración a día de hoy, además se someten constantemente a una empresa ethical hacking que corrige vulnerabilidades.	Moderada, ya que los archivos no contienen ninguna información sensible del usuario, todos los datos son públicos en internet buscando mediante el rut.	Medio, acá más que nada entra en juego la noticia de que hackearon a Mach, ya que, si bien los datos no son relevantes, el impacto reputacional sí lo es.	Informarlo al equipo, confirmar cual fue la vulneración que se efectuó y que el equipo de ciberseguridad corrija la brecha.
Que no se mande la información diaria de las solicitudes de tarjeta física.	Poco probable, ya que es algo que muy rara vez ha sucedido y se tienen mensajes de alerta si es que falla algo así.	Moderada, ya que si no se corrige lo antes posible puede implicar en atrasos en las entregas de las tarjetas.	Medio, ya que el único impacto que se puede generar serían las quejas de los usuarios solo si es que la tarjeta no llegase a tiempo	Identificar mediante los logs que es lo que está fallando en el envío del archivo de solicitudes y de esta forma corregirlo.



Riesgo	Probabilidad	Consecuencias	Impacto	Mitigación
Que se mande un archivo CSV con algún error en el formato.	Poco probable, es algo que muy rara vez sucede, sobre todo por la cantidad de validadores que se poseen sobre el formato del archivo.	Moderada, ya que si no se corrige implicaría atrasos en las entregas de las tarjetas físicas.	Medio, ya que el único impacto que se puede generar serían las quejas de los usuarios solo si es que la tarjeta no llegase a tiempo	Identificar mediante los logs que es lo que está fallando en la generación del archivo de solicitudes y de esta forma corregirlo.

Tabla 1. Matriz de riesgo

6. Resultados

6.1 El proyecto

A lo largo de esta pasantía se logró cumplir con el objetivo principal de eliminar el único punto de falla en el flujo de tarjetas físicas para convertirlo en un sistema de alta disponibilidad integrando un nuevo proveedor de tarjetas físicas antes del Q4 de 2023.

Además de esto, se actualizó el sistema de lambdas de tal forma que se encuentra con los estándares actuales que contempla Mach para sus servicios, a la vez que se cuenta con una solución escalable¹⁵ y fácil de modificar en caso de que se quiera incorporar a futuro un nuevo proveedor, o cambiar algún parámetro de la división de los archivos.

Todo el proceso, desde la identificación del problema, hasta la investigación de la solución y el como se implementó quedó documentado dentro de la información que maneja la empresa, por lo que se logró documentar para todos el nuevo flujo de solicitudes de tarjetas físicas con sus mejores implementadas.

6.2 Desarrollo personal

Dentro de este proyecto de pasantía se logró profundizar y mejorar aún más las herramientas que ya se poseían gracias a la universidad, logrando aplicarlas en casos de uso reales,

¹⁵ Pandora Tech. (2023, marzo 20) LinkedIn.com. <https://es.linkedin.com/pulse/la-relaci%C3%B3n-cost-beneficio-de-usar-lambdas-damper>



aplicando también la metodología ágil, un eje central de las enseñanzas que se nos brindaron dentro de la universidad.

Cabe destacar, que se generaron nuevos conocimientos al tener que investigar y aplicar en profundidad en servicios en la nube como los microservicios, o funciones gatilladas por eventos como lo son las lambdas, además que la experiencia del trabajo en equipo fue sumamente enriquecedora para el crecimiento personal de lo que es un entorno real de trabajo, logrando aumentar de esta manera la red de contactos que se posee.

6.3 Impacto

Se logró un impacto positivo dentro de diversos ámbitos a lo largo de este proyecto, el primero y más destacable fue la eliminación del único punto de fallas, logrando hacer un sistema de alta disponibilidad acorde a lo que se espera de una empresa con más de 4 millones de usuarios, a la vez que se logró también un positivo impacto económico dentro de la compañía, logrando un importante ahorro de costos para el siguiente año con la posibilidad de que sea aún mayor al largo plazo, y sin lugar a dudas, cabe destacar el impacto presente dentro de las medidas de desempeño de los objetivos específicos.

Con respecto al porcentaje de objetivos alcanzado, que va directamente relacionado con la cantidad de StoryPoints completados se logró completar en un 100%, lo cual es muy destacable ya que demuestra que todo el desarrollo se vio efectuado según los plazos que se tenían previstos.

Con respecto a la evaluación de la documentación, se logró una puntuación de un 9.6 máximo posible de un 10, contando con un 10 en la puntuación del proceso, un 10 en la puntuación de la integración, y un 9 en la puntuación del discovery.

Con respecto a la mejora del stack tecnológico, cuya medida de desempeño se rige mediante tres métricas diferentes de mejora en tasa de falla, mejora en uso de recursos y mejora en el número de vulnerabilidades, debido a que aún no hay una implementación real hasta el día 3 de enero de 2024 aún no se pueden obtener estas métricas para determinar su impacto.

Sin embargo, dado todo lo anterior podemos ver que realmente se logró un impacto positivo dentro de la empresa con este proyecto de pasantía.



7. Conclusión

Con respecto a la solución implementada para el proyecto, se logra una división correcta de las solicitudes de tarjeta física, con un parámetro que es fácil de modificar en caso de cualquier cambio que se requiera a futuro, a la vez que se desarrolló una segunda lambda que valida y verifica correctamente la integridad de los archivos, con una estructura moderna y acorde a los estándares actuales de la compañía, todo esto se podrá habilitar sin que el usuario se vea afectado ni se percate siquiera de que existe una nueva empresa embozadora de tarjetas físicas, ya que el feedback que el usuario reciba en la aplicación está estandarizado para ambos proveedores.

A la vez que se logró la mejor solución, logrando un equilibrio entre carga de trabajo dentro del equipo de tarjetas y movimientos, se utilizó lo que es mejor tecnológicamente dentro de la vanguardia de lo que es hoy en día en la industria para funciones que se ejecutan pocas veces al día y requieren un menor uso de recursos, lo que aporta directamente a los costos. A la vez que al ser más barato este nuevo proveedor también se disminuyen los costos generales de la fabricación y entrega de las tarjetas físicas para los usuarios que la soliciten dentro de la aplicación.



8. Referencias

- Scrum: Amazon. (S/F) aws.amazon.com. <https://aws.amazon.com/es/what-is/scrum/>
- Mach, (S/F). Somosmach.com. <https://www.somosmach.com/quienes-somos>
- Osvaldo Gimenez. (2023, octubre 7). La Tercera.
<https://www.latercera.com/pulso/noticia/ceo-de-mercado-pago-adelanta-los-nuevos-productos-que-lanzaran-en-chile/YTEHMXMAPJFLPCKLDV2L7XG2TI/>
- Fernando Araya. (2023, agosto 9). Forbes Chile.
<https://forbes.cl/negocios/2023-08-09/fernando-araya-de-tenpo-en-chile-hay-una-migracion-muy-grande-de-bancos-tradicionales-a-companias-como-tenpo>
- Whitney Vige. (2022, diciembre 3). Asana.com.
<https://asana.com/es/resources/story-points>
- Microservicios: Amazon. (S/F) aws.amazon.com.
<https://aws.amazon.com/es/microservices/>
- Lambdas: Amazon. (S/F). docs.aws.amazon.com.
https://docs.aws.amazon.com/es_es/lambda/latest/dg/welcome.html
- JavaScript: Mozilla. (2023, julio 24). developer.mozilla.com.
<https://developer.mozilla.org/es/docs/Web/JavaScript>
- Typescript: Typescript. Lang (S/F). typescriptlang.org.
<https://www.typescriptlang.org/docs/handbook/typescript-from-scratch.html>
- SNS: Amazon. (S/F). docs.aws.amazon.com.
https://docs.aws.amazon.com/es_es/sns/latest/dg/welcome.html
- Tech Lead: Inesdi. (2022, mayo 31). Inesdi.com.
<https://www.inesdi.com/blog/tech-lead-funciones-salario/>
- Sem: Eric Mora. (S/F). ericmora.me
<https://ericmora.me/organigrama-digital/que-hace-un-software-engineering-manager>
- Test unitarios: Amazon. (S/F) aws.amazon.com.
<https://aws.amazon.com/es/what-is/unit-testing/>
- Pandora Tech. (2023, marzo 20) Linkedin.com.
<https://es.linkedin.com/pulse/la-relaci%C3%B3n-costo-beneficio-de-usar-lambdas-damper>