



Diseño de un sistema de autogestión de accesos y recursos de nube

Orlando Allan Rodríguez Pamplona

Informe de prácticas académicas para optar por título de Ingeniería de Sistemas

Asesor Interno Joan Andrés Hasper Tabares, Ingeniera en Sistemas UdeA

Asesor Externo Juan Felipe Gallo Rendón

Universidad de Antioquia
Facultad de Ingeniería
Ingeniería de Sistemas
Medellín, Antioquia, Colombia

2022

Cita

(Rodríguez Pamplona, 2022)

Referencia

- [1] Rodríguez Pamplona, O.A. (2022). Diseño de un sistema de autogestión de accesos y recursos de nube [Trabajo de grado profesional]. Universidad de Antioquia, Medellín, Colombia

Estilo IEEE (2020)



Rector: John Jairo Arboleda Céspedes.

Decano/Director: Jesús Francisco Vargas Bonilla.

Jefe departamento: Diego José Luis Botía Valderram.

El contenido de esta obra corresponde al derecho de expresión de los autores y no compromete el pensamiento institucional de la Universidad de Antioquia ni desata su responsabilidad frente a terceros. Los autores asumen la responsabilidad por los derechos de autor y conexos.

Dedicatoria

Al medio, que evoluciona constantemente y a las personas, que buscan evolucionar con este.

Agradecimientos

Primeramente, a mi madre por hacer todo lo posible para permitirme llegar hasta aquí. Luego, a la Universidad de Antioquia por permitir mi desarrollo integral como persona y profesional.

Finalmente a Experimentality S.A.S por ayudarme a crecer en el ámbito profesional.

TABLA DE CONTENIDO

RESUMEN	6
ABSTRACT	7
I. INTRODUCCIÓN	8
II. OBJETIVOS	9
A. Objetivo general	9
B. Objetivos específicos	9
III. MARCO TEÓRICO	10
IV. PLANTEAMIENTO DEL PROBLEMA	11
V. METODOLOGÍA	12
VI. DESARROLLO DEL PRODUCTO	19
A. Product Backlog	14
B. Casos de Uso	14
C. Principales retos	14
D. Diagrama de la solución planteada	15
E. Diagrama de la arquitectura	16
VII. RESULTADOS	19
VIII. CONCLUSIONES	22
REFERENCIAS	23

LISTA DE FIGURAS

Fig. 1. Ejemplo de tablero ScrumBan	12
Fig. 2. Diagrama de contenedores	15
Fig. 3. Diagrama arquitectura de la infraestructura final.	16
Fig. 4. Diagrama de arquitectura de la infraestructura MVP.	17
Fig. 5 Diagrama de procesos	17
Fig. 6 Tabla comparativa de soluciones	19

RESUMEN

Experimentality S.A.S. Es una compañía de desarrollo de software, que ha experimentado un gran crecimiento en poco tiempo a nivel de personal y procesos. En este rol de compañía, está sujeto a una serie de actividades recurrentes, que deben efectuarse para continuar con su funcionamiento interno, tales como: la creación de usuario en los sistemas de comunicación de la compañía, acceso a los servicios de terceros, entre otros. Además, su estatus como empresa de tecnología, revela flujos adicionales que se adquieren a nivel de compañía como: repositorios de código, proveedores de cómputo en la nube, etc. Los cuales, se traducen a integraciones con productos de terceros.

La elaboración constante de estos procedimientos es de naturaleza recurrente y genera cargas de trabajo extra en los colaboradores mientras la empresa va creciendo, haciendo de éste, un proceso manual que no escala.

Teniendo como idea primordial, el concepto de Automatización de procesos de negocio (BPA)[1], en la cuál se busca reducir la carga operativa por medio de automatización de procesos de negocio y la adición de nuevas necesidades por ser una compañía de tecnología, se observa la posibilidad de evaluar e incorporar un sistema que resuelva tanto necesidades de negocio como necesidades técnicas de manera automatizada, para así, lograr el objetivo de liberar carga productiva al transformar estos flujos.

Palabras clave — BPA, software, automatización, Computación en la nube

ABSTRACT

Experimentality S.A.S. is a software development company, it have experienced a big growth in human resources and processes in a short time. This role as a company, is attached to a variety of recurrent activities, which have to be executed in order to operate normally. Some of those are: user's creation in the communication software, access to third party services, etc. Besides, Experimentality's status as a technology enterprise, reveals additional flows which are acquired as: code's repositories, cloud computing's resources, etc. Generating a greater number of third party's integration.

The execution of these procedures are recurrent processes, so they generate additional workloads in the company's collaborators, becoming a manual process that is non scalable.

Having as a core concept "Business Process Automation", which wants to reduce the operative workload by automating business processes, and the addition new needs by consequence of being a technology enterprise, the possibility of the evaluation and implementation of a system that resolves these needs automatically borns, in order to release productive workload.

Keywords —BPA, software, automate, Cloud Computing

I. INTRODUCCIÓN

En este documento, se da a conocer los resultados de la práctica académica, en la modalidad semestre de industria. Donde, el estudiante dedicó el tiempo al diseño e implementación de un sistema interno, el cual tiene como objetivo ayudar a la compañía a nivel organizacional para reducir las cargas de trabajo.

Se realizó la evaluación de posibles soluciones en el mercado o la necesidad de la implementación de un sistema propio que responda a las necesidades de la compañía. Se definieron los procesos piloto y estos hicieron parte de la implementación preliminar o MVP (Mínimo producto viable) de la solución. Se entregó un prototipo funcional que se podrá utilizar en los flujos de la compañía a nivel productivo, siendo el limitante del alcance, los flujos más prioritarios.

Es inevitable tener procesos repetitivos en las compañías, ya que muchos de estos son tareas que se ejecutan con base en su crecimiento o funcionamiento general. Unos ejemplos de estos podrían ser *Onboarding* de nuevos colaboradores, solicitud de recursos o solicitudes recurrentes como vacaciones o incapacidades.

Estos procesos son bien conocidos en el entorno, de la misma manera, son abordados de diferentes formas. Debido a esto ya existen variedad de plataformas y servicios que solucionan la mayoría de interacciones de negocio y ya tienen casos de éxito. Lo que se evaluó en este proyecto fue la oportunidad de integrar un sistema como estos que también responda a los procesos reiterativos encontrados en una empresa enfocada en la tecnología o el desarrollo de software. Esta particularidad, nos hizo descartar varios de los competidores en el mercado y generó la oportunidad de explorar una solución íntegra para compañías en el área de tecnología que, a pesar de compartir muchas similitudes con las compañías convencionales, adiciona flujos específicos que también requieren una solución.

II. OBJETIVOS

A. Objetivo general

Implementar un sistema de automatización que permite ejecutar tanto tareas repetitivas de negocio como tareas de contexto tecnológico.

B. Objetivos específicos

- Generar un banco de procesos repetitivos que pueden ser automatizados.
- Generar necesidades particulares en base a los procesos repetitivos.
- Realizar una búsqueda de soluciones o servicios que respondan a las necesidades particulares.
- Escoger la solución o implementación necesaria para la realización del proyecto.
- Generar las tareas a ser ejecutadas en su orden de prioridad y con la respectiva priorización.
- Desarrollar una fase preliminar del producto que la empresa pueda usar con fines productivos.

III. MARCO TEÓRICO

En términos organizacionales, existen procesos que se realizan de manera recurrente y que involucran diferentes sistemas de TI tales como: Incorporación de nuevos colaboradores a la compañía, creación de accesos a sistemas de la compañía, etc. De estos, existen de varios tipos, pero todos cumplen el mismo objetivo, suplir necesidades de negocio.

Debido a esto, se crea la necesidad de una Gestión de procesos[2] que trae consigo operatividad para la ejecución de las mismas. Con esto en mente, nace el concepto de BPA (Business Process Automation), el cuál consiste en hacer uso de herramientas o plataformas para automatizar procesos recurrentes, creando también en consecuencia una responsabilidad de identificar estos procesos e incorporarlos a tiempo en los sistemas. Además, de los procesos recurrentes encontrados en las organizaciones en general, las compañías de software traen un reto adicional, el aprovisionamiento de recursos informáticos en plataformas de nube ya sea a nivel productivo o a nivel de experimentación. Con el constante crecimiento del concepto de DevOps[3], nace también el concepto de Infraestructura como código (IaC)[4], la cuál tiene el objetivo de poder crear recursos de nube de manera programática usando código y así poder automatizar la creación de los mismos.

IV. PLANTEAMIENTO DEL PROBLEMA

En el mundo empresarial, teniendo a las compañías como sujetos de estudio, se evidencia que estas están sustentadas sobre una serie de herramientas, tanto propias como de terceros. Estas herramientas, que a su vez resuelven necesidades tales como: la comunicación, el almacenamiento de archivos, etc. requieren de un proceso de creación de accesos y usuarios que se repite con cada nuevo colaborador en la compañía. Debido a esto se genera una carga que debe ser ejecutada por los colaboradores de la compañía, reduciendo así, la capacidad productiva de la misma.

Analizando estos procesos, se evidencia que tienen en común factores como integraciones con varios sistemas de terceros para resolver alguna necesidad de la compañía. El cómo interactúan estos sistemas con los procesos de la compañía y la necesidad de los mismos añade un peso más a la carga anteriormente mencionada.

Por último, entrando en un contexto más específico como lo son las empresas de tecnología, tenemos procesos o solicitudes tienen como efecto la creación de recursos de cómputo en la nube, o de permisos sobre la misma, añadiendo una capa más a nuestra carga operacional.

La compañía Experimentally S.A.S., requiere la capacidad de elaborar con éxito los flujos anteriormente mencionados con un enfoque más automático y de autogestión, reduciendo así la demanda operativa de la empresa y mejorando la productividad.

V. METODOLOGÍA

La metodología Scrumban, permite planificar los proyectos según se ejecuta. Se marcan los hitos, al final de los denominados sprints y, cuando se llega a esa fecha, se revisa el trabajo hecho y se planifica el siguiente sprint o iteración, según los nuevos requerimientos o cambios producidos. En esta metodología se tienen varias columnas, las cuales suelen ser: hecho, demorado, en proceso, por hacer. Cada columna está formada por actividades que todo el equipo comparte. De esta manera, todo el equipo revisa sus tareas, su prioridad y fecha de entrega.

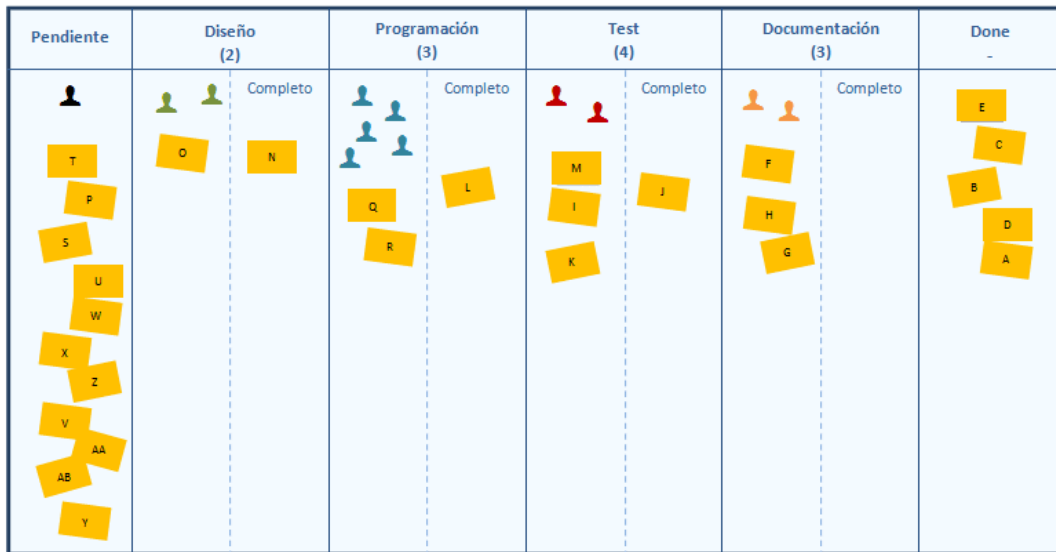


Fig. 1. Ejemplo de tablero ScrumBan

VI. DESARROLLO DEL PRODUCTO

A. Entendimiento de la necesidad

En el proceso de análisis de la necesidad se encontraron los siguientes procesos:

- Onboarding de nuevos colaboradores
- Reinicio de credenciales en los proveedores de nube utilizados por la empresa.
- Solicitud de permisos en los proveedores de nube utilizados por la empresa.
- Flujo de entrega de facturas

B. Casos de Uso

Para la elección de la solución se tuvieron en cuenta los siguientes requerimientos:

- *El sistema debe poder interactuar con los sistemas de terceros usados en la compañía.*
- *La ejecución de los flujos debe poder efectuarse tanto por personas técnicas como por personas de negocio.*
- *La construcción de los flujos debe ser mantenible y modificable en el tiempo.*
- *El precio de la solución no debe ser alto, debido a que es un proceso evolutivo que de momento no tendrá un presupuesto asignado.*
- *La solución deberá poder usarse a demanda sin necesidad de depender de tareas de aprovisionamiento o mantenimiento de servidores.*

Estas condiciones nacen tanto de la necesidad hablada con la compañía como de los resultados de la búsqueda de soluciones efectuada.

Luego de haber realizado la evaluación de las soluciones en el mercado, se tomó la decisión, tomando en cuenta los requisitos aquí planteados, de realizar una solución a la medida escalable para la realización de estas tareas.

C. Principales retos

En un principio, el principal reto fué contrastar las diferentes soluciones del mercado por sus funcionalidades. Se llegó a considerar varias opciones de las cuales ninguna cumplía con la cabalidad de los requerimientos. A manera de

observación, uno de los candidatos finales en la búsqueda de soluciones era la implementación de flujos automatizables en Jenkins, esto debido a su flexibilidad y facilidad de ejecución. Sin embargo, se generaba la necesidad de mantenimiento de un servidor, que a la larga, implica más carga operacional.

Ahora, en la creación del desarrollo a la medida también se tuvo en cuenta contrastar diferentes opciones. De nuevo entran en juego los requisitos aquí planteados, buscando siempre un sistema usable y escalable.

Otra de las dificultades de esta elaboración fué la disponibilidad de los sistemas de terceros para brindar la utilización de los mismos de manera programática, que si bien es necesario, es algo que no se puede controlar. Por esto, algunos pasos del proceso no se pudieron automatizar completamente.

D. Diagrama de la solución planteada

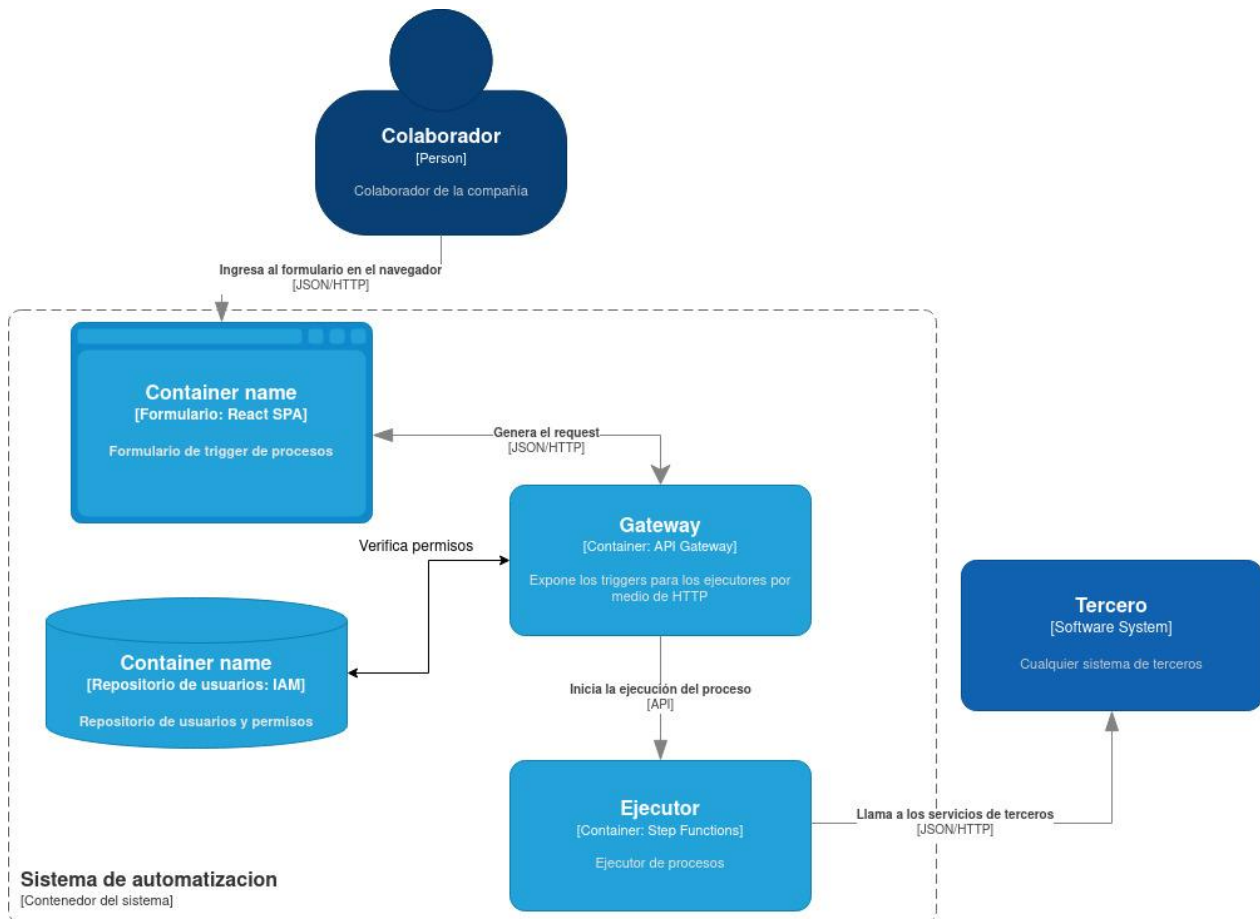


Fig. 2. Diagrama de contenedores

Podemos evidenciar que el sistema presenta un punto de entrada ejecutado por un usuario a través de una SPA. Esta SPA es simplemente encargada de mostrar los formularios para la ejecución de los distintos procesos y el envío da como resultado un REQUEST al sistema ejecutor. La seguridad se hará por medio del repositorio de usuarios en AWS IAM.

Se evidencia también el componente de sistema de terceros el cuál será modificado por los diferentes procesos.

E. *Diagramas de la arquitectura.*

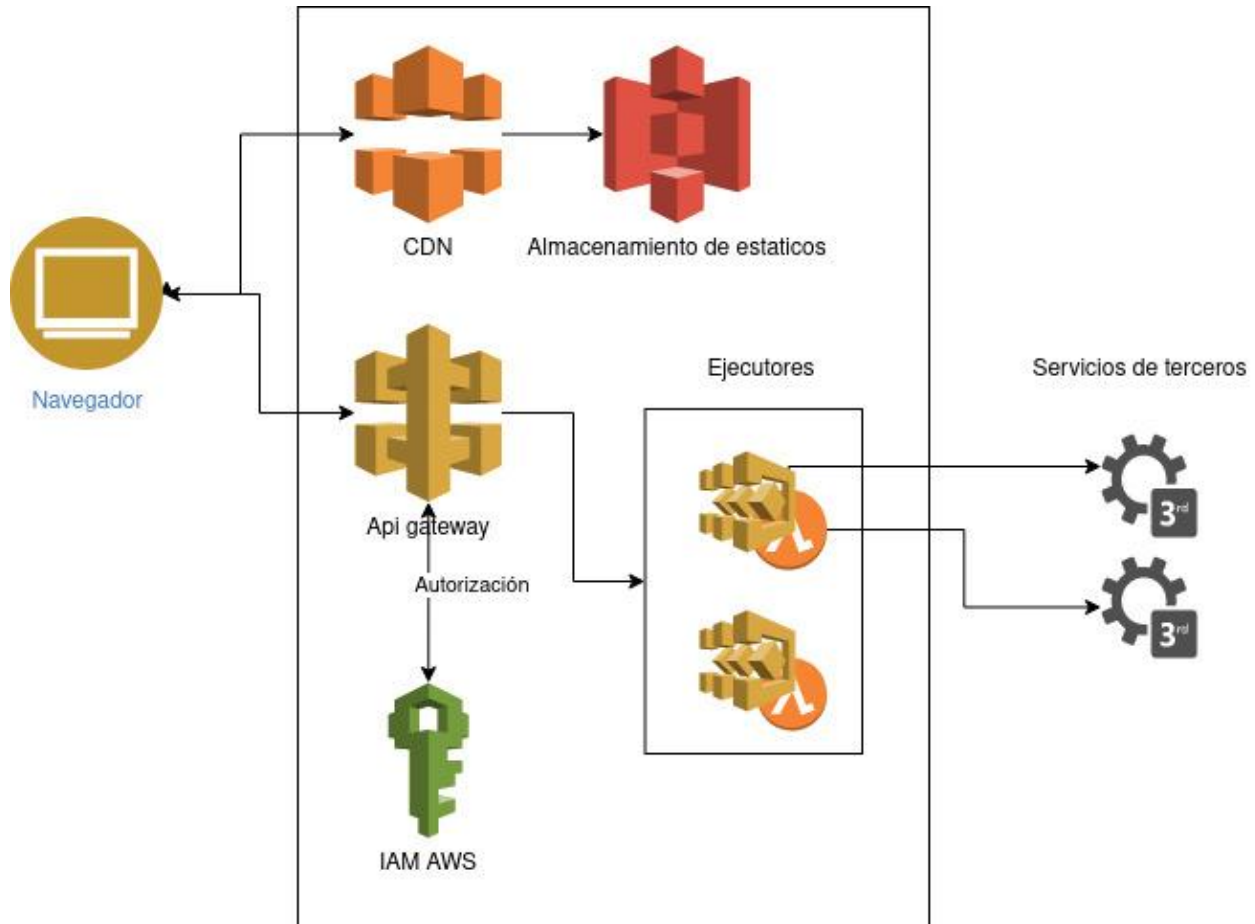


Fig. 3. Diagrama arquitectura de la infraestructura final.

El diseño planteado hace uso de la nube AWS, junto a varios de sus servicios. El fundamento principal de esta arquitectura es el de pagar solo por el consumo y no por aprovisionamiento. Con esto, se busca un sistema que sea administrado y permita concentrarse solo en la ejecución de las tareas puntuales.

Se aclara el repositorio de usuarios (IAM) como base de autenticación y autorización, siendo este un paso importante antes de poder disparar el proceso.

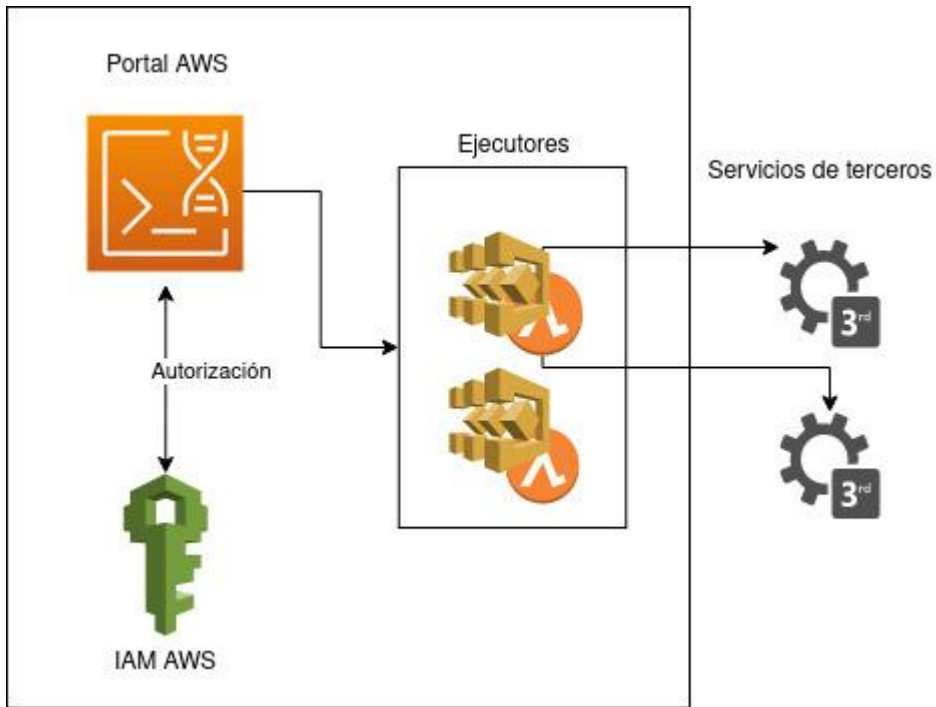


Fig. 4. Diagrama de arquitectura de la infraestructura MVP.

El entregable, por otro lado, hará uso de un disparador manual en la consola de AWS, facilitando así la autenticación y la ejecución de los procesos de manera sencilla.

Otro punto necesario, es la vista de la construcción de los procesos.

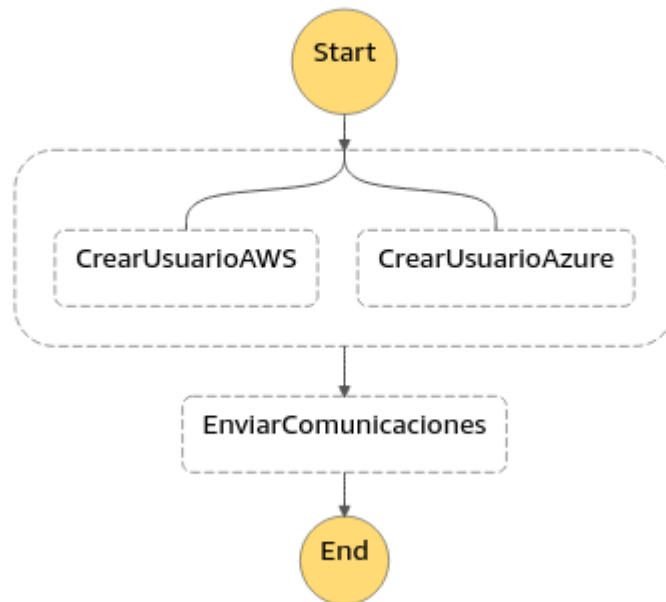


Fig. 5. Diagrama de procesos.

Los procesos pueden ser elaborados de una manera visual generando así una visión más clara de la responsabilidades del mismo y generando discusión en cuanto a la refinación de estos procesos.

VII. RESULTADOS

Luego de una serie de reuniones recolectando los procesos aptos para ser automatizados, se procedió a priorizar los mismos y luego a seleccionar cuales debían entrar en el producto mínimo viable.

Luego de esto, se realizó la búsqueda y la comparación de las posibles soluciones, dando como resultado esta tabla:

Herramienta	Pros	Contras
Nintex	Tiene grandes integradores y conectores además de una interfaz intuitiva para los procesos.	Su elevado precio de \$910 us por mes. El conocimiento adquirido con esta herramienta no es conocimiento que puedas reutilizar. No posee conectores para algunas integraciones con proveedores de nube.
Process Maker	Igual que su competencia, posee varias integraciones con aplicaciones de terceros. Posee una interfaz para el usuario final y tiene buena curva de aprendizaje.	En este caso también tenemos un precio alto de \$1,495 us por mes + precios por cada usuario adicional. Es una herramienta que usa protocolos propietarios y no se podrían implementar fuera de la misma.
Jenkins	Gran flexibilidad con sus plugins para la construcción de cualquier tipo de flujo. Se integra con el sistema de usuario que implemente la compañía. Tiene disparadores sencillos para usuarios no técnicos. Se pueden construir conectores a cualquier sistema externo.	Pese a ser free, Jenkins requiere de un servidor funcionando y esto aumenta el proceso operativo del mantenimiento del servidor.
Desarrollo a la medida	Gran flexibilidad en el diseño	Requiere mano de obra

	<p>de los procesos. Se construyen los conectores necesarios. Se puede crear como servicio y que cueste por su uso. Se pueden explorar diferentes maneras de iniciar los procesos.</p>	<p>técnica para la elaboración de los flujos.</p>
--	--	---

Fig. 6. Tabla comparativa de soluciones.

Con la información anteriormente mencionada, se tomó la decisión de realizar un desarrollo a la medida.

Se inició realizando pruebas de concepto en diferentes proveedores de funciones orquestadas (Azure, AWS), siendo el segundo el elegido con su servicio Step Functions. Se realizó una prueba de concepto con el proceso de Onboarding de la compañía, el cual requería de la integración con varios servicios de terceros.

Luego de realizar la primera conexión, se procedió a realizar la implementación de la infraestructura en un esquema de Infraestructura como Código, usando Cloud Formation(AWS). Esto con el objetivo de poder replicar la arquitectura en diferentes cuentas y ambientes.

Con este esquema se empezaron a construir los otros procesos priorizados, los cuales respondieron bien en ambiente de pruebas.

Productos próximos:

El producto será socializado con la compañía para su conocimiento y será puesto en productivo luego de esto. Se espera que se vuelva algo clave en la compañía y evolucione en base al feedback de la misma.

Inmediatamente después, se tiene pensado empezar con el desarrollo del aplicativo front end para el trigger web. Este se hará a manera de semillero para los nuevos talentos de la compañía.

Aspectos de mantenimiento, código, secuencias, repositorios, actualizaciones:

El producto está sujeto a múltiples validaciones y ajustes que serán necesarios en cuanto avance el tiempo y los procesos cambien. Estos cambios deberán poder ser ejecutados por cualquier persona del equipo técnico.

El código de la aplicación y los scripts de IaaS se encuentran en el repositorio de github empresarial, donde los colaboradores pueden agregar y mantener los cambios que sean requeridos.

Aspectos preponderantes de la aplicación:

Alta disponibilidad del servicio, arquitectura serverless y cloud native, Infraestructura como código y alta escalabilidad.

VIII. CONCLUSIONES

Experimentality S.A.S. Al ser una empresa en crecimiento, empieza a enfrentarse a nuevos retos y necesidades, no solo a nivel técnico. Esto genera oportunidades para abordar problemáticas que solo seguirá incrementando mientras la compañía siga su crecimiento.

El proyecto nace como una inversión de la compañía a nivel de procesos internos, la búsqueda de mejorar las bases sobre las que se cimienta el funcionamiento de la empresa. Además, la oportunidad de explorar nuevos conocimientos y maneras de solucionar problemas de software que luego podrán ser aplicados a clientes finales.

La implementación de soluciones utilizando AWS nos permite generar conocimientos prácticos que después pasarán a ser usados en proyectos con clientes. Además, pese a ser un proyecto de prácticas, es importante connotar las buenas prácticas y conceptos de desarrollo de software que se implementan desde aquí, una etapa de pruebas y diseño de producto, hasta el cliente final.

Este proyecto genera una manera más de solucionar necesidades de compañía, y brinda la posibilidad de generar un ambiente de ejecución seguro en el cuál se pueden empezar a considerar nuevos procesos o nuevas necesidades que en un principio no eran visibles.

REFERENCIAS

- [1]. What is business process automation, REDHAT [online] Available at: <https://www.redhat.com/es/topics/automation/what-is-business-process-automation/> [Accessed 10 January 2022].
- [2]. Procesos para la gestión en empresas de software. [online] Available at: <https://navegapolis.com/index.php/116-procesos-para-la-gestion-en-empresas-de-software> [Accessed 10 January 2022].
- [3]. What is DevOps, AWS[online] Available at: <https://aws.amazon.com/devops/what-is-devops/> [Accessed 10 January 2022]
- [4]. What is Infrastructure as code,[online] Available at: <https://www.redhat.com/es/topics/automation/what-is-infrastructure-as-code-iac> [Accessed 20 January 2022].