

**PLATAFORMA TECNOLÓGICA BASADA EN PROCESS MINING PARA
LA GESTIÓN Y MEJORAMIENTO DE PROCESOS
ORGANIZACIONALES**

EDWIN RAFAEL ALFARO IGUARÁN

UNIVERSIDAD DE PAMPLONA

**FACULTAD INGENIERÍAS Y ARQUITECTURA
PROGRAMA DE INGENIERÍA DE SISTEMAS**

PAMPLONA, NORTE DE SANTANDER

2022

**PLATAFORMA TECNOLÓGICA BASADA EN PROCESS MINING PARA
LA GESTIÓN Y MEJORAMIENTO DE PROCESOS
ORGANIZACIONALES**

EDWIN RAFAEL ALFARO IGUARÁN

**TRABAJO DE GRADO PARA OPTAR EL TÍTULO DE
INGENIERO DE SISTEMAS**

DIRECTOR

JORGE OMAR PORTILLA JAIMES

MAGISTER EN CIENCIAS DE LA COMPUTACIÓN

UNIVERSIDAD DE PAMPLONA

FACULTAD INGENIERÍAS Y ARQUITECTURA

PROGRAMA DE INGENIERÍA DE SISTEMAS

PAMPLONA, NORTE DE SANTANDER

2022

AGRADECIMIENTOS

A mis padres, quienes a pesar de que a lo largo de mi formación profesional surgieron varios inconvenientes, estuvieron siempre presentes brindándome todo su amor y paciencia, apoyándome para no rendirme y cumplir este gran logro.

A mis hermanos, a mi pareja y amigos que me aconsejaron como afrontar diversas situaciones, ya que, de no ser por ellos, no habría llegado tan lejos.

A la Universidad de Pamplona por haberme brindado la oportunidad de formarme como profesional. A todos y cada uno de mis docentes, quienes me brindaron todo su conocimiento y experiencias, y en especial a mi profesor y amigo Jorge Omar Portilla, quien vio mucho potencial en mi para lograr cosas grandes.

RESUMEN

Actualmente las Pyme representan aproximadamente el 98% de las empresas de toda Latinoamérica, y son generadoras de más del 60% de empleo formal. Lastimosamente muchas de estas pequeñas y medianas empresas se declaran en quiebra al no lograr adaptarse por completo al ritmo tan acelerado con el que se mueve el mercado de hoy en día.

Por otro lado, la minería de procesos (process mining) es una nueva disciplina que se encarga de comprender todos los procesos de una organización, y mediante el uso de diversas técnicas es posible descubrir la realidad de los procesos, analizar si son ejecutados de manera correcta e incluso es posible mejorarlos.

Debido a esto, el presente proyecto tiene como finalidad el desarrollar un prototipo de una de una plataforma tecnológica en la que sea posible implementar diversas técnicas de process mining, para que las pequeñas y medianas empresas de Latinoamérica que están en riesgo de quebrar tengan la oportunidad de gestionar y mejorar todos sus procesos de forma fácil, rápida y personalizada.

Palabras clave: Minería de Procesos, event log, redes de Petri, python

ÍNDICE DE CONTENIDO

INTRODUCCIÓN	1
PLANTEAMIENTO DEL PROBLEMA	2
JUSTIFICACIÓN	4
1. OBJETIVOS	5
1.1. Objetivo General	5
1.2. Objetivos Específicos	5
2. METODOLOGÍA	6
2.1. Cronograma y descripción de actividades	7
2.1.1. Cronograma	7
2.1.2. Descripción de actividades	7
3. MARCO TEÓRICO	9
3.1. Antecedentes	9
3.1.1. Minería de Procesos (Process Mining) según autores	9
3.1.2. Procesos y Event log	13
3.1.3. Descubrimiento de Procesos	14
3.1.4. Chequeo de Conformidad	15

3.1.5.	Mejoramiento de Procesos	15
3.1.6.	Cuellos de Botella	16
3.2.	PM4PY	16
3.2.1.	Descubrimiento de Procesos en PM4PY	16
3.2.2.	Chequeo de Conformidad en PM4PY	21
4.	PROCEDIMIENTO Y RESULTADOS	24
4.1.	Ciencia de Datos y PM4PY	24
4.2.	Desarrollo Backend de la herramienta (Pymes+Plus).....	29
4.2.1.	Preliminares.....	29
4.2.2.	Modelado de Procesos (Descubrimiento).....	32
4.2.3.	Chequeo de Conformidad.....	35
4.2.4.	Cuellos de Botella	38
4.3.	Desarrollo Frontend de Pymes+Plus.....	44
4.3.1.	Boceto.....	45
4.3.2.	Desarrollo de la interfaz	48
4.4.	Observaciones	52
5.	CONCLUSIONES	56

6.	TRABAJOS FUTUROS	57
	REFERENCIAS BIBLIOGRÁFICAS.....	59

ÍNDICE DE FIGURAS

Figura 1 <i>Cronograma de Actividades.</i>	7
Figura 2 <i>Ejemplo Event log</i>	14
Figura 3 <i>Minero Alpha</i>	19
Figura 4 <i>Minero Heurístico</i>	19
Figura 5 <i>Minero Inductivo</i>	20
Figura 6 <i>Extracto de event log de casos hospitalarios</i>	25
Figura 7 <i>Extracto de event log después de aplicar ciencia de datos</i>	26
Figura 8 <i>Gráficos de puntos</i>	26
Figura 9 <i>Comparación de modelos de redes de Petri descubiertos con PM4PY</i>	28
Figura 10 <i>Grafo de seguimiento directo (DFG)</i>	28
Figura 11 <i>Extracto de Event log modificado</i>	30
Figura 12 <i>Extracto de event log modificado luego de aplicar ciencia de datos</i>	30
Figura 13 <i>Valores únicos</i>	31
Figura 14 <i>Recursos por Acciones</i>	32
Figura 15 <i>Actividades como Transiciones</i>	33
Figura 16 <i>Lugares de Espera (entrada)</i>	34

Figura 17 <i>Lugares de Recursos</i>	35
Figura 18 <i>Variantes (Secuencias diferentes)</i>	35
Figura 19 <i>Agregando lugares de salida</i>	36
Figura 20 <i>Matriz de salidas</i>	37
Figura 21 <i>Representación del modelo terminado</i>	38
Figura 22 <i>Relación de Casos por Variante</i>	39
Figura 23 <i>Reconstrucción de los hechos</i>	40
Figura 24 <i>Selección de caminos</i>	41
Figura 25 <i>Extracto de la Marcación</i>	41
Figura 26 <i>Cuellos de Botella</i>	42
Figura 27 <i>API - Process Mining</i>	43
Figura 28 <i>Dashboard Railway</i>	44
Figura 29 <i>Boceto Login</i>	45
Figura 30 <i>Boceto Página de subida de Archivos</i>	46
Figura 31 <i>Boceto Página de Listado de Archivos</i>	46
Figura 32 <i>Boceto Página de Resultados 1</i>	47
Figura 33 <i>Boceto Página de Resultados 2</i>	47

Figura 34 *Login de Acceso*..... 48

Figura 35 *Subida de Archivos* 49

Figura 36 *Listado de Archivos* 49

Figura 37 *Formulario para procesamiento* 50

Figura 38 *Resultados: Generalidades y Mejor Variante* 51

Figura 39 *Resultados: Cuellos de Botella*..... 51

Figura 40 *Resultados: Ocupación de Recursos por Actividad* 52

ÍNDICE DE TABLAS

Tabla 1 <i>Comparación de Algoritmos de Descubrimiento de Procesos</i>	20
Tabla 2 <i>PM4PY: Ventajas y Desventajas</i>	53
Tabla 3 <i>PM4PY vs Pymes+Plus</i>	54

INTRODUCCIÓN

Día a día, dentro de las empresas y organizaciones se realizan miles de tareas, que son realizadas por personas o por sistemas automatizados con el fin de alcanzar un objetivo en específico. Con el pasar del tiempo, es muy probable que muchas veces surja la misma pregunta ¿Por qué se demora tanto este proceso? y comience la búsqueda de los posibles culpables.

La aplicación de minería de procesos resulta muy útil para estos casos, ya que puede ser utilizada para comprender el comportamiento de todo tipo de procesos, ayudando a identificar donde se producen los diferentes cuellos de botella, ya sean ocasionados por la falta de recursos, o por una posible desviación en la ejecución del proceso, permitiendo de esta manera, encontrar las partes del modelo de procesos que pueden ser optimizadas para obtener un mejor rendimiento de los estos en ejecuciones futuras.

De esta manera, con el presente trabajo se pretende crear una plataforma que pueda ser capaz de comprender los procesos organizacionales de las pymes, para ayudarlas a identificar y solucionar todas las posibles ineficiencias que afecten de manera negativa el rendimiento de sus procesos.

PLANTEAMIENTO DEL PROBLEMA

Actualmente las Pyme representan aproximadamente el 98% de las empresas de toda Latinoamérica, y son generadoras de más del 60% de empleo formal. Lastimosamente muchas de estas pequeñas y medianas empresas se declaran en quiebra al no lograr adaptarse por completo al ritmo tan acelerado con el que se mueve el mercado de hoy en día. Esto ocurre debido a que la mayoría de pymes en sus inicios cuentan con una capacidad de producción reducida, que les permite cumplir adecuadamente con sus tiempos de entrega, para así poder brindar productos y servicios de una buena calidad. Principalmente cuando su demanda comienza a aumentar, los tiempos de entrega y la calidad comienzan a verse afectados de manera negativa y muchas pymes no logran entender por qué ocurre esto. Este tipo de problemas, pueden ser solucionados mediante minería de procesos.

Sin embargo, autores como Berti, et al. (2019) expresan que al ser la Minería de Procesos (Process Mining) una disciplina emergente, esta ha experimentado muchos cambios en estos últimos años, ya que desde sus inicios a finales de los 90s ha resultado muy difícil de implementar, pues había soporte muy limitado o nulo a las herramientas disponibles en aquella época. En la actualidad se han desarrollado distintas herramientas de software para la implementación de minería de procesos, tanto de código abierto como ProM, PM4PY; y, comerciales como Disco, Celonis, entre otros. Aunque hoy en día existe esta variedad de herramientas, todavía resulta complicado utilizarlas, ya que, en algunos casos, la documentación existente no explica detalladamente la funcionalidad de los elementos que componen estas herramientas.

Por otra parte, Berti, et al. (2019), también comentan que estas herramientas brindan muy poco soporte para la implementación de algoritmos personalizados, por lo cual no se puede tener un “control total” del rendimiento en los entornos experimentales.

Al no contar con herramientas que contribuyan a la gestión y mejoramiento de sus procesos organizacionales, de manera fácil, rápida y personalizada, la mayoría de pymes quiebra por no ser capaces de entender por qué el rendimiento de sus procesos ha disminuido tanto que comienza a perjudicar sus tiempos de entrega y la calidad de sus productos y/o servicios, provocando de esta manera grandes pérdidas económicas.

Por esta razón, surge la necesidad de plantear la siguiente pregunta de investigación: ¿Cómo debe ser una plataforma tecnológica basada en process mining para la gestión y mejoramiento de procesos organizacionales?

JUSTIFICACIÓN

Según investigaciones de La República (2020), en Latinoamérica solo 45% de las Pyme no quiebra en sus dos primeros años, esto sucede debido a que no todas las pequeñas y medianas empresas logran adaptarse al ritmo del mercado actual, presentando demasiados fallos y retrasos en sus procesos de producción, lo que a su vez genera muchas pérdidas económicas.

Mediante el uso de herramientas enfocadas a la minería de procesos, estos problemas podrían ser solucionados, pero, autores como Berti, et al. (2019), manifiestan que debido a que varias de estas herramientas actuales enfocadas a la minería de procesos cuentan con interfaz gráfica de usuario (GUI), muchos usuarios inexpertos intentan conocer la forma de gestionar y optimizar sus procesos organizacionales, pero debido a la alta complejidad de estas herramientas y la poca documentación existente, su usabilidad se dificulta en gran medida, provocando que la mayoría de usuarios prefiera desistir.

Por esta razón resulta necesario el desarrollo de una plataforma tecnológica que cuente con una interfaz gráfica amigable con el usuario, y que además, sea capaz de identificar, gestionar y aconsejar formas de mejorar los procesos de las pequeñas y medianas empresas mediante la aplicación de las diversas técnicas de process mining.

1. OBJETIVOS

1.1. Objetivo General

Implementar una plataforma para la gestión y mejoramiento de procesos organizacionales soportada en process mining.

1.2. Objetivos Específicos

- Realizar un comparativo de los diferentes algoritmos existentes para descubrimiento, chequeo de conformidad y mejoramiento de procesos.
- Definir una arquitectura computacional que soporte la plataforma tecnológica para la gestión y mejoramiento de procesos organizacionales soportada en process mining.
- Desarrollar el backend requerido por la arquitectura computacional definida.
- Desarrollar el frontend requerido para la arquitectura computacional definida.
- Validar mediante un caso de uso la plataforma tecnológica desarrollada para la gestión y mejoramiento de procesos usando process mining.

2. METODOLOGÍA

La metodología aplicada para el presente trabajo puede dividirse en dos etapas principales, la etapa de Investigación y la etapa de Desarrollo.

Dentro de la etapa de investigación, se lleva a cabo una búsqueda para recopilar información, y experiencias de diferentes autores que estén involucrados con la investigación y aplicación de la Minería de Procesos. La información encontrada proviene en su mayoría de artículos científicos, libros, tesis de grado y la participación en cursos en modalidad de MOOC sobre la Minería de Procesos.

Por otra parte, dentro de la etapa de desarrollo, se llevan a cabo una serie de pruebas utilizando la librería PM4PY, la cual como su nombre lo indica (Process Mining For Python) está enfocada a la minería de procesos. De igual forma, en esta etapa se inicia con el desarrollo de Pymes+Plus, el prototipo de una plataforma capaz de aplicar técnicas de minería de procesos, para identificar, gestionar, y mejorar los procesos organizacionales de las pequeñas y medianas empresas.

Las etapas de las que consta este trabajo fueron realizadas siguiendo el cronograma, que se presenta a continuación.

2.1. Cronograma y descripción de actividades

2.1.1. Cronograma

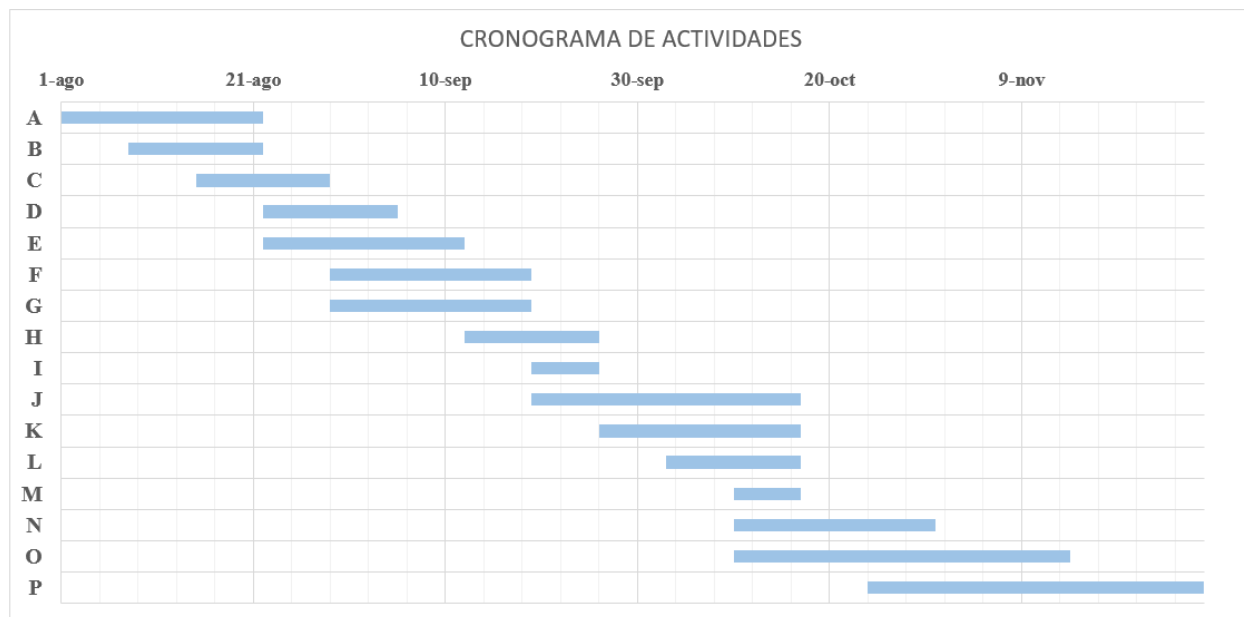


Figura 1 *Cronograma de Actividades*.
Elaboración propia.

2.1.2. Descripción de actividades

- A. Investigar los distintos algoritmos enfocados en el descubrimiento de procesos.
- B. Determinar que algoritmo es más eficaz para el descubrimiento de procesos.
- C. Investigar los distintos algoritmos enfocados en el chequeo de conformidad.
- D. Determinar que algoritmo es más eficaz para el chequeo de conformidad.
- E. Investigar los distintos algoritmos enfocados en el mejoramiento de procesos.
- F. Determinar que algoritmo es más eficaz para el mejoramiento de procesos.
- G. Diseño de arquitectura que tendrá la plataforma (¿Qué hará?, datos de entrada, datos de salida, etc.).

- H. Diseñar Backend de la plataforma (Algoritmos, estructuras de datos a utilizar).
- I. Determinar la tecnología Backend a utilizar.
- J. Implementar el diseño del Backend.
- K. Realizar pruebas para validar el correcto funcionamiento del Backend.
- L. Diseñar bocetos para el Frontend que tendrá la plataforma.
- M. Determinar la tecnología Frontend a utilizar.
- N. Implementar el diseño del Frontend.
- O. Realizar pruebas para validar la correcta integración del Frontend y el Backend.
- P. Redacción del documento.

3. MARCO TEÓRICO

En este capítulo se describirán algunos de los temas básicos necesarios para comprender la problemática y el desarrollo de esta investigación, con el objetivo de explicar y aclarar los conceptos a todas aquellas personas que no estén relacionadas con la temática tratada en este documento.

De esta manera, se abarcarán temas como la definición de Minería de Procesos, sus características y algunas de sus actuales aplicaciones; haciendo referencia a event logs, descubrimiento de procesos y chequeo de conformidad.

3.1. Antecedentes

3.1.1. *Minería de Procesos (Process Mining) según autores*

Garcia, et al. (2019), mencionan que:

La minería de procesos es un área de estudio creciente y prometedora centrada en comprender los procesos y ayudar a capturar los hallazgos más significativos durante la ejecución real en lugar de aquellos métodos que solo observan el modelo de proceso idealizado.

De esta manera, el profesor Wil van der Aalst (2016) quien es precursor de la minería de procesos dentro de su libro afirma que:

La minería de procesos puede verse como un medio para cerrar la brecha entre la ciencia de datos y la ciencia de procesos. Los enfoques de ciencia de datos tienden a ser independientes del proceso, mientras que los enfoques de ciencia de procesos tienden a basarse en modelos sin considerar la ‘evidencia’ oculta en los datos.

Esto también puede ser complementado con algunas de las ideas planteadas por autores como Molero, et, al., quienes afirman que el área de la minería de procesos proporciona diversas técnicas para extraer conocimiento a partir de los grandes volúmenes de datos, y a su vez servir como apoyo al momento de la toma de decisiones (2017). Para van der Aalst (2016), “La idea de la minería de procesos consiste en descubrir, monitorear y mejorar procesos reales (es decir, procesos no asumidos) mediante la extracción de conocimiento de los registros de eventos fácilmente disponibles en los sistemas actuales”. De la misma manera, en investigaciones recientes, se plantea que hoy en día la gran mayoría de las organizaciones recurren cada vez más a los sistemas del tipo PAIS (Process-Aware Information Systems - Sistemas de Información Conscientes de los Procesos), para que mediante su uso puedan administrar y ejecutar todos sus procesos, y al mismo tiempo ir guardando esta gran cantidad de información en registros de eventos (Pasquadibisceglie et al., 2022).

Para Aguirre y Rincón (2015) los registros de eventos (event logs) son el historial de ejecución de los procesos, estos contienen información heterogénea y de alta dimensión como, por ejemplo, que tareas fueron realizadas, quien o que realizó cierta tarea, distintas marcas de tiempo y muchos más datos en general.

Por otro lado, los autores Marín-Castro y Tello-Leal (2021), y Guzzo, et al. (2021), afirman que las organizaciones que cuentan con todos estos datos podrían implementar técnicas de process

mining, puesto que estas brindan la oportunidad de extraer todo el valor oculto que poseen los datos.

Las técnicas de minería de procesos se dividen en tres tipos, técnicas para el descubrimiento de procesos, las cuales utilizan los event logs para identificar y producir los modelos de procesos; así mismo, encontramos las técnicas para el chequeo de conformidad, estas comparan un modelo de proceso descubierto previamente con event logs del mismo proceso para verificar si la realidad que está almacenada en los event logs resulta semejante al modelo descubierto y viceversa; y por ultimo tenemos las técnicas para el mejoramiento de procesos, que ayudan a adaptar y optimizar los modelos de procesos existentes utilizando la información almacenada en los event logs sobre los procesos reales.

En ese mismo orden de ideas, Pérez (2015) plantea dentro de su tesis algunas de las principales razones para la aplicación de la minería de procesos, afirmando que la minería de procesos es un mecanismo de evaluación de los procesos que son realizados por los funcionarios encargados de cumplirlos, y se apoya en el registro real con el que cuentan las aplicaciones como soporte de sus transacciones; de esta manera se aprovechan todos los registros que solo son consultados en muy pocos casos (casos de error o situaciones anómalas) y se facilita la evaluación del comportamiento real de los procesos, donde se pueden identificar cuellos de botella, flujos de procesos alternativos, tareas no realizadas y repetitivas, etcétera.

Con el transcurrir de los años la tecnología ha evolucionado y se ha demostrado la eficacia de la minería de procesos al momento de recopilar, analizar y mejorar los procesos de las organizaciones, lo que ha conllevado a un crecimiento en el desarrollo de proyectos de práctica (en sectores de gran importancia para la sociedad como la salud, las finanzas, la automatización

de procesos, entre otros) y proyectos de investigación que permiten ampliar el campo de acción de esta disciplina, cabe mencionar algunos proyectos que se han destacado por la implementación de process mining:

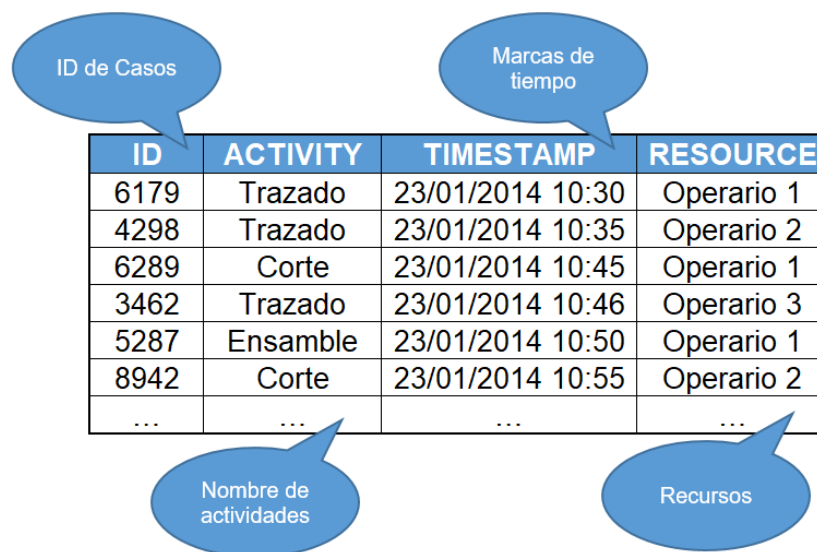
- Yang, et al. (2022), explican que las técnicas de Process Mining (Minería de Procesos) pueden ser aplicadas hasta el punto de poder derivar modelos organizativos a partir de event logs que contengan datos relacionados con los recursos; pero que estas aún no pueden determinar las agrupaciones de recursos involucradas en la ejecución de los procesos; razón por la cual proponen el framework OrdinoR, el cual es capaz de solucionar todos estos problemas.
- Kregel, et al. (2021), plantean que resulta muy necesario integrar Process Mining (PM) a la metodología de optimización de procesos Six Sigma, debido a que la complejidad de los procesos aumenta a pasos agigantados, y las herramientas de BI (Business Intelligence) no pueden analizarlos, ya que están enfocadas a la elaboración de dashboards e informes simples.
- Sippl, et. al. (2022), afirman que, debido a los constantes cambios y mejoras de la tecnología, a las empresas les corresponde adaptar, tanto sus productos como sus procesos de producción de forma permanente, y mediante la aplicación técnicas de Process Mining, es posible comprender el método de trabajo actual, además de identificar donde se producen los cuellos de botella, para así poder mejorar el desempeño de los procesos.
- Orellana (2018), demuestra que existe un gran interés en los procesos hospitalarios, ya que, estos procesos resultan ser muy caóticos por estar sometidos todo el tiempo a entornos cambiantes debido a las decisiones tomadas por los profesionales de la

salud, es decir, siempre existirán muchas variantes de un mismo proceso, lo que ocasiona que el servicio no sea del todo eficiente, por lo cual se hace necesaria la aplicación de process mining.

3.1.2. *Procesos y Event log*

C3d3 (2020) afirma que un proceso puede ser definido como una secuencia de actividades (o eventos) que han sido planificadas e implican la participación de cierta cantidad de actores y recursos para conseguir objetivos que han sido previamente identificados, por ejemplo, cada paso que realiza una persona al momento de retirar dinero de un cajero automático (introducir tarjeta, introducir pin, seleccionar retirar, etc.), es un evento que hace parte del proceso de retiro de dinero.

Por otra parte, un event log puede definirse como el historial de ejecuciones de los eventos de un proceso (Aguirre & Rincón, 2015), donde cada evento o acción, es asociado a un caso, a un grupo de recursos y a un instante en el tiempo, permitiendo así tener una idea superficial del comportamiento del proceso (Figura 2).



The diagram shows a table representing an Event log. Four blue callout bubbles point to the columns: 'ID de Casos' points to the ID column, 'Marcas de tiempo' points to the TIMESTAMP column, 'Nombre de actividades' points to the ACTIVITY column, and 'Recursos' points to the RESOURCE column.

ID	ACTIVITY	TIMESTAMP	RESOURCE
6179	Trazado	23/01/2014 10:30	Operario 1
4298	Trazado	23/01/2014 10:35	Operario 2
6289	Corte	23/01/2014 10:45	Operario 1
3462	Trazado	23/01/2014 10:46	Operario 3
5287	Ensamble	23/01/2014 10:50	Operario 1
8942	Corte	23/01/2014 10:55	Operario 2
...	...	...	...

Figura 2 *Ejemplo Event log*
Elaboración propia.

3.1.3. Descubrimiento de Procesos

Según Automation Anywhere (s. f.), el descubrimiento de procesos es un conjunto de técnicas y tecnologías utilizadas para poder identificar la forma en la que son realizados los procesos organizacionales.

El descubrimiento de procesos es una técnica de minería de procesos que describe la visualización basada en datos de un proceso, la cual genera un modelo de proceso en forma de gráfico de seguimiento directo (DFG) a partir de los registros de eventos (event logs), teniendo como objetivos principales una mayor transparencia y conocimiento de los procesos. (Appian, s. f.-d).

3.1.4. *Chequeo de Conformidad*

Según Appian (s. f.-b), la comprobación de conformidad o chequeo de conformidad es una técnica de minería de procesos utilizada para comparar registros de eventos (event logs) con un modelo que corresponda al comportamiento de un proceso objetivo, esto con el objetivo de garantizar la identificación de todas las desviaciones existentes tanto en el proceso objetivo como en el modelo, ya que estas desviaciones pueden generar impactos negativos en la calidad de los productos o servicios de las empresas, provocando de esta manera pérdidas financieras; por esta razón, las empresas tienen mucho interés en identificar todas las secuencias de actividades que ocurren y no están planificadas, para así poder tomar las medidas preventivas necesarias o modificar el proceso para disminuir la aparición de estas desviaciones.

3.1.5. *Mejoramiento de Procesos*

La mejora de los procesos es un procedimiento realizado por las organizaciones, con el cual se pretende encontrar ineficiencias en el flujo de los procesos para poder corregirlos y mejorar su rendimiento. (Ekon, 2022).

En la minería de procesos, al aplicar las técnicas de descubrimiento de procesos y chequeos de conformidad, es posible identificar varias debilidades en los procesos, incluidos cuellos de botella, actividades repetidas, o desviaciones no deseadas en el proceso; esto es información muy valiosa, ya que, cuenta con un alto potencial de optimización, y mediante la aplicación de cambios y mejoras en el modelo de procesos, se puede mejorar el rendimiento de estos. (Appian, s. f.-c).

3.1.6. Cuellos de Botella

Un cuello de botella tiene la capacidad de empeorar o retrasar el desempeño de un proceso, aumentando así, el tiempo de ciclo o de ejecución, o incluso siendo capaz de paralizarlo por completo, debido a la escasez de recursos como la mano de obra, maquinaria, o materiales de fabricación. Los cuellos de botella pueden ser encontrados y eliminados mediante el descubrimiento de procesos y análisis de chequeo de conformidad. (Appian, s. f.-a).

3.2. PM4PY

PM4PY (Process Mining for Python) es una biblioteca de Python de código abierto desarrollada por Fraunhofer Institute for Applied Information Technology (FIT) con el fin de poner a disposición del público algoritmos de minería de procesos, donde se pueden destacar los algoritmos enfocados al descubrimiento de procesos y el chequeo de conformidad, ya que actualmente no hay algoritmos para el mejoramiento de procesos.

3.2.1. Descubrimiento de Procesos en PM4PY

El descubrimiento de procesos consiste en la búsqueda de un modelo de red de Petri que pueda describir el comportamiento de un proceso, a partir del orden de los eventos o actividades que son ejecutadas durante la ejecución de dicho proceso.

3.2.1.1. Propiedades Importantes de los Modelos

En el descubrimiento de procesos con PM4PY, por lo general los modelos descubiertos son modelos de redes de Petri, y cuentan con las siguientes propiedades importantes.

- *SEGURIDAD (SAFENESS)*: Está basada en la idea de que todos los eventos o actividades presentes dentro del modelo están permitidos, y son mutuamente excluyentes, es decir, la probabilidad de que ambos ocurran al mismo tiempo es de cero.
- *AUSENCIA DE PARTES MUERTAS*: En modelos de redes de Petri, para cada transición debe existir una o varias secuencias que produzcan que ocurra, es decir, deben existir secuencias de disparos que la habiliten.
- *OPCIÓN PARA COMPLETAR*: Todo modelo de red de Petri, debe tener un lugar marcado como final, para que de esta manera exista la posibilidad de completar los procesos al momento de reproducirlos.
- *FINALIZACIÓN ADECUADA*: Al momento de reproducir una traza (caso) del event log y esta termina (finaliza el proceso), ninguna de las actividades que hacen parte de la secuencia de ejecución de dicha traza puede ejecutarse, es decir, las actividades solo pueden ejecutarse cuando una traza se está reproduciendo.
- *ROBUSTEZ*: Para que un modelo pueda ser considerado como robusto, debe reunir todas las propiedades anteriores, es decir, por lo menos la actividad inicial del proceso debe poder ejecutarse (ausencia de partes muertas) para que las actividades siguientes puedan ser ejecutadas en orden (seguridad) hasta su finalización de manera correcta (opción para completar, finalización adecuada).

3.2.1.2. Descubrimiento de Procesos en PM4PY

Actualmente dentro de la librería PM4PY de Python, posible encontrar varios algoritmos y sus variantes para el descubrimiento de procesos, algunos de los más utilizados son: el Minero Alpha, el Minero Alpha+ (el cual es una mejora del minero Alpha), el Minero Inductivo y el Minero Heurístico.

3.2.1.2.1. Minero Alpha

El minero Alpha es de los algoritmos más conocidos, ya que fue quizás el primer algoritmo de descubrimiento de procesos capaz de descubrir concurrencia. Al utilizar el minero Alpha pueden ser encontrados modelos de redes de Petri cuyas transiciones son visibles y únicas, además, estos modelos también cuentan con marcaciones iniciales y finales que describen cuando comienza y finaliza una ejecución.

Las principales limitaciones del Minero Alpha son:

- No tiene en cuenta la frecuencia de los eventos o actividades, por lo que no es posible el manejo de bucles.
- Necesita event logs sin ruido para poder encontrar un modelo.
- No puede descubrir transiciones ocultas.
- No garantiza robustez.

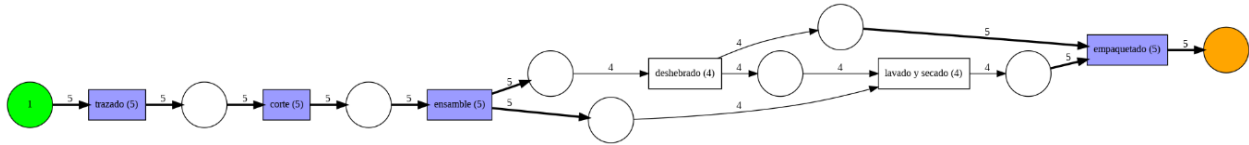


Figura 3 *Minero Alpha*
Elaboración propia.

3.2.1.2.2. *Minero Heurístico*

El minero heurístico, actúa sobre un gráfico de seguimiento directo (DFG) lo que le permite ser capaz de soportar el ruido y de encontrar modelos en forma de redes heurísticas las cuales son objetos que contienen todas las actividades de un proceso y las relaciones entre ellas; y así mismo, también es capaz de encontrar modelos de redes de Petri que cuentan con sus respectivas marcaciones.

Al aplicar el minero heurístico, se tiene en cuenta las frecuencias de los eventos o actividades, y es capaz de detectar bucles cortos, sin embargo, no garantiza robustez y es más adecuado cuando la cantidad de eventos es pequeña.

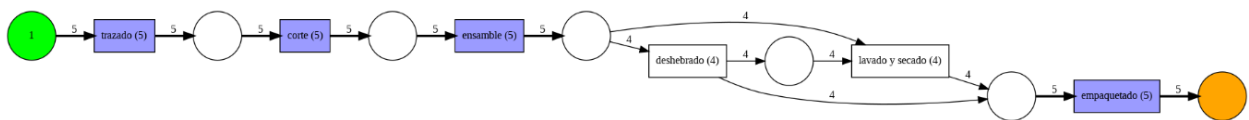


Figura 4 *Minero Heurístico*
Elaboración propia.

3.2.1.2.3. *Minero Inductivo*

El minero inductivo tiene la capacidad de encontrar dos modelos de procesos, redes de Petri y árboles de procesos. El minero inductivo puede dividir el event log, en sub-logs más pequeños hasta encontrar un caso base.

Este algoritmo tiene en cuenta la frecuencia de los eventos y es capaz de detectar transiciones ocultas o invisibles para evitar hacer bucles dentro del modelo, y como una mejora para los demás mineros, se garantiza la robustez.

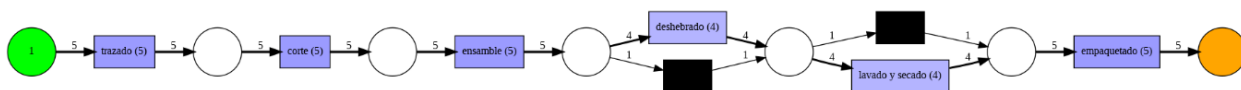


Figura 5 *Minero Inductivo*
Elaboración propia.

3.2.1.2.4. *Comparación entre Algoritmos de Descubrimiento de Procesos*

Tabla 1 *Comparación de Algoritmos de Descubrimiento de Procesos*

Características	Alpha	Heurístico	Inductivo
Encuentra modelos de redes de Petri con su respectiva marcación.	✓	✓	✓
Encuentra otro tipo de modelos de procesos.	X	✓	✓

Soporta el ruido.	X	✓	✓
Tiene en cuenta las frecuencias de los eventos	X	✓	✓
Garantiza la robustez.	X	X	✓

Elaboración propia.

Al momento de escoger que algoritmo utilizar para poder descubrir los modelos de procesos, los minero heurístico e inductivo arrojan los mejores resultados, ya que tienen en cuenta los bucles, las frecuencias, las transiciones ocultas y representan modelos más robustos a comparación de los modelos del minero alfa.

Por otra parte, resulta necesario examinar las métricas de precisión y de aptitud con algoritmos de chequeo de conformidad para determinar cuál de los dos resulta ser el más adecuado para representar el comportamiento de cualquier proceso.

3.2.2. Chequeo de Conformidad en PM4PY

La librería PM4PY cuenta con dos de los principales algoritmos para el chequeo de conformidad: la reproducción basada en tokens y las alineaciones. Los cuales pueden ayudar a

elegir que algoritmo de descubrimiento de procesos utilizar ya que ambos son muy utilizados para calcular las estadísticas de aptitud, precisión, generalización y simplicidad de los modelos.

3.2.2.1. Reproducción Basada en Tokens (TBR)

La reproducción basada en tokens hace coincidir una traza (o caso) con el modelo que ha sido descubierto anteriormente, partiendo desde el lugar inicial con la finalidad de descubrir que transiciones han sido ejecutadas y en qué lugares se encuentran tokens restantes o faltantes, siendo esto realizado para cada traza. Cuando las trazas lleguen al lugar final sin necesidad de agregar tokens faltantes, se dice que se ajustan al modelo.

3.2.2.2. Reproducción Basada en Alineaciones

La reproducción basada en alineaciones tiene como objetivo encontrar una de las mejores alineaciones entre las trazas y el modelo, arrojando como resultado listas de parejas con los resultados de los movimientos ya sean sincronizados o con desviaciones tanto en la traza como en el modelo lo que implicaría movimientos no adecuados.

3.2.2.3. Elegir entre Algoritmos de Chequeo de Conformidad

Al momento de elegir que algoritmo de chequeo de conformidad utilizar, hay que tener en cuenta que, aunque la reproducción basada en tokens es mucho más rápida, es un método heurístico y por lo tanto su resultado puede que no sea del todo exacto, mientras que las alineaciones son

exactas, pero mucho más lentas ya que representan un problema computacionalmente más costoso en modelos que contienen mucha concurrencia, pero estas pueden descomponerse en partes más pequeñas que ayudan a que la alineación sea más fácil.

4. PROCEDIMIENTO Y RESULTADOS

En el presente capítulo se describe en detalle el procedimiento realizado para cumplir con todos los objetivos propuestos para este trabajo, y mediante la aplicación de ejemplos de pruebas y evidencias se irá mostrando el avance del proyecto.

4.1. Ciencia de Datos y PM4PY

Para entender cómo funciona la minería de procesos, resulta útil experimentar en un ambiente de pruebas que no esté limitado por el hardware, para esto, se opta por utilizar la plataforma Colaboratory, también conocida como Google Colab, ya que permite programar y ejecutar código en Python desde el navegador, y brinda acceso a GPUs sin coste alguno, y además no requiere ningún tipo de configuración previa para aprovechar toda la potencia de librerías como Pandas, Numpy y Matplotlib, permitiendo también agregar nuevas librerías al entorno de trabajo como en este caso, PM4PY, lo que facilita mucho la experimentación.

Lo primero que se necesita para iniciar con la minería de procesos, es tener a la mano un event log con el que trabajar, razón por la cual para la realización de pruebas y algunos ejemplos que serán mostrados a continuación, se decide utilizar un event log (tomado de: <https://gitlab.com/healthcare2/process-mining-tutorial>) donde se puede apreciar el trayecto hospitalario de un grupo de pacientes (Figura 6), desde que ingresan hasta que son dados de alta, es decir, que para este caso en particular, toda cita a la que un paciente asista dentro del hospital es un evento que conforma este proceso, lo que permite tener una idea superficial del

comportamiento de este proceso. Cabe aclarar que, aunque el desarrollo de este trabajo tiene como finalidad ayudar a las pymes a gestionar y mejorar sus procesos, se toma como ejemplo un event log de casos hospitalarios, ya que es posible generalizar cualquier proceso de cualquier organización.

	patient	action	resource	datetime
0	patient 0	First consult	Dr. Anna	2017-01-02 11:40:11
1	patient 0	Blood test	Lab	2017-01-02 12:47:33
2	patient 0	Physical test	Nurse Jesse	2017-01-02 12:53:50
3	patient 0	Second consult	Dr. Anna	2017-01-02 16:21:06
4	patient 0	Surgery	Dr. Charlie	2017-01-05 13:23:09

Figura 6 *Extracto de event log de casos hospitalarios*

Tomado de “Process Mining with Python tutorial: A healthcare application — Part 1”, Medium, https://medium.com/@c3_62722/process-mining-with-python-tutorial-a-healthcare-application-part-1-ae02027a050.

Al ya tener definido el event log a utilizar, mediante ciencia de datos este puede ser preparado para poder analizar y extraer la mayor cantidad de información posible, dentro de la información obtenida podemos encontrar la cantidad de actividades que fueron realizadas para cada caso, los tiempos de ejecución total de cada caso, tiempos relativos entre actividades, la secuencia de actividades que sigue cada caso, etc. (Figura 7), lo que resulta útil para la generación de distintos tipos de gráficos que permitan visualizar el comportamiento de cada caso a lo largo del tiempo (Figura 8).

	patient	action	resource	datetime	caseend	casestart	relativetime	action_sequence	numactions	caselength	weekday	date	startdate	hour	relativetime_s	relativedays
0	patient 0	First consult	Dr. Anna	2017-01-02 11:40:11	2017-01-09 08:29:28	2017-01-02 11:40:11	0 days 00:00:00	First consult__Blood test__Physical test__S...	6	6 days 20:49:17	0	2017-01-02	2017-01-02	11	0	0
1	patient 0	Blood test	Lab	2017-01-02 12:47:33	2017-01-09 08:29:28	2017-01-02 11:40:11	0 days 01:07:22	First consult__Blood test__Physical test__S...	6	6 days 20:49:17	0	2017-01-02	2017-01-02	12	4042	0
2	patient 0	Physical test	Nurse Jesse	2017-01-02 12:53:50	2017-01-09 08:29:28	2017-01-02 11:40:11	0 days 01:13:39	First consult__Blood test__Physical test__S...	6	6 days 20:49:17	0	2017-01-02	2017-01-02	12	4419	0
3	patient 0	Second consult	Dr. Anna	2017-01-02 16:21:06	2017-01-09 08:29:28	2017-01-02 11:40:11	0 days 04:40:55	First consult__Blood test__Physical test__S...	6	6 days 20:49:17	0	2017-01-02	2017-01-02	16	16855	0
4	patient 0	Surgery	Dr. Charlie	2017-01-05 13:23:09	2017-01-09 08:29:28	2017-01-02 11:40:11	3 days 01:42:58	First consult__Blood test__Physical test__S...	6	6 days 20:49:17	3	2017-01-05	2017-01-02	13	265378	3

Figura 7 Extracto de event log después de aplicar ciencia de datos
Elaboración propia.

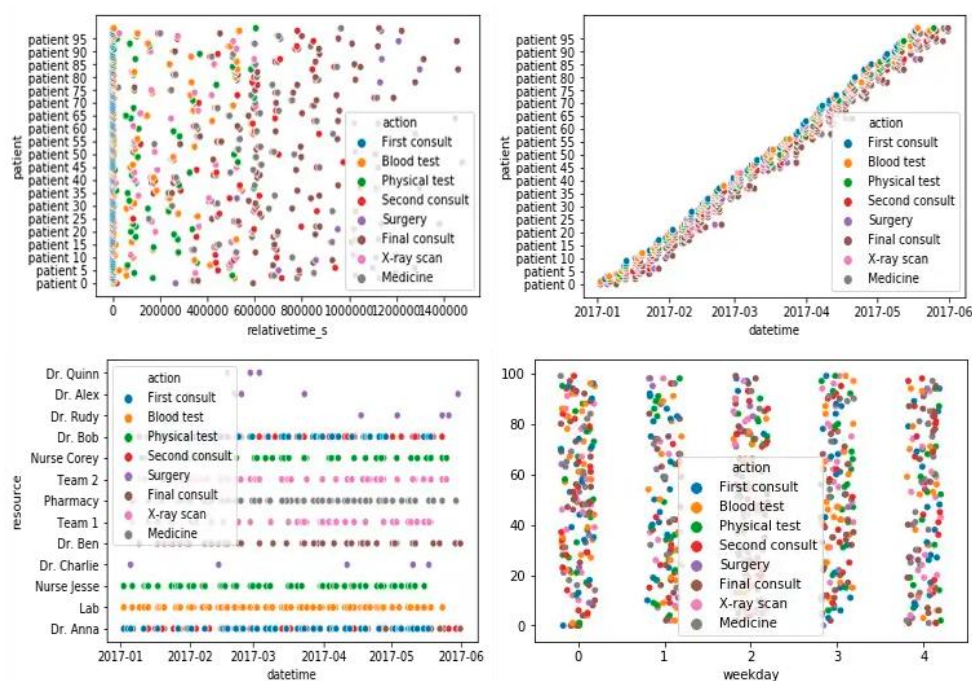


Figura 8 Gráficos de puntos

Adaptado de “Process Mining with Python tutorial: A healthcare application — Part 1”, Medium,
https://medium.com/@c3_62722/process-mining-with-python-tutorial-a-healthcare-application-part-1-ae02027a050.

Aunque para este momento se ha analizado gran parte de la información que puede brindar un event log, todavía no se han aplicado a profundidad las diversas técnicas de minería de procesos, para poder aplicarlas, se utiliza la librería PM4PY, pero primero se debe renombrar las columnas

principales del event log (id, action, resource, datetime) para que la librería pueda comprenderlo y así dar inicio con el descubrimiento de procesos.

En la etapa de descubrimiento de procesos es posible implementar de manera sencilla distintos algoritmos que al aplicarse sobre el event log pueden crear varios tipos de modelos (redes de Petri, redes heurísticas, arboles de procesos, entre otros) que sirven para representar el flujo de trabajo del proceso, cada uno con algunas diferencias con respecto a los otros ya que intentan adaptarse al proceso de maneras distintas (Figura 9). Para determinar qué modelo descubierto utilizar, es necesario aplicar algoritmos de chequeo de conformidad para encontrar cual cuenta con mejores estadísticas de aptitud, precisión, generalización y simplicidad.

PM4PY también cuenta con la capacidad de generar grafos de seguimiento directo (DFG) (Figura 10) para comprender que están capturando u omitiendo los diferentes modelos, ya que, estos son la representación más fiel al comportamiento real de un proceso, por lo que resulta muy útil compararlos con los modelos descubiertos.

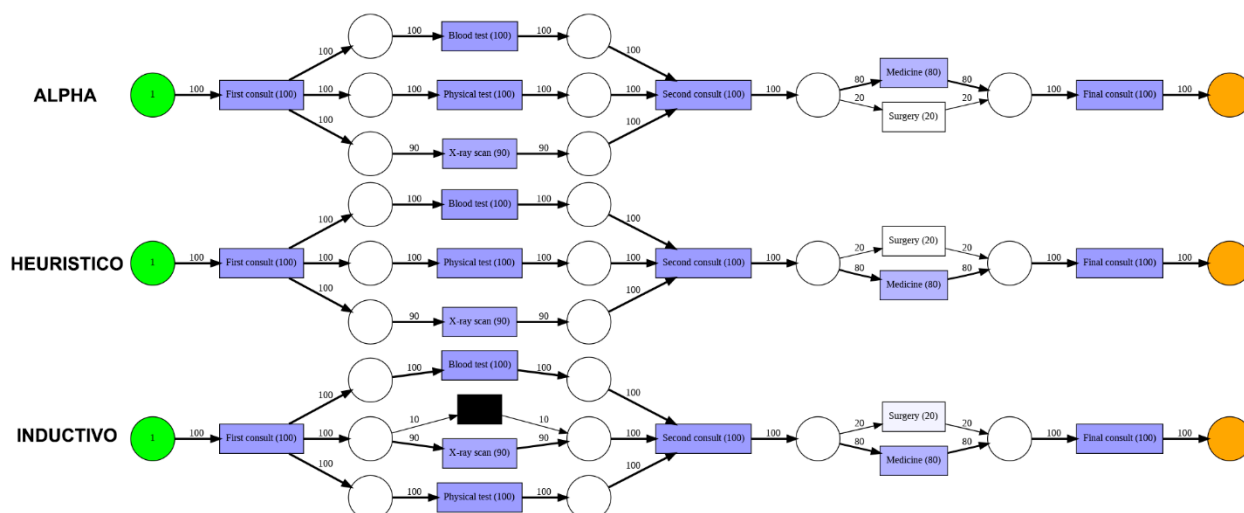


Figura 9 Comparación de modelos de redes de Petri descubiertos con PM4PY

Adaptado de https://medium.com/@c3_62722/process-mining-with-python-tutorial-a-healthcare-application-part-3-cc9af986c122.

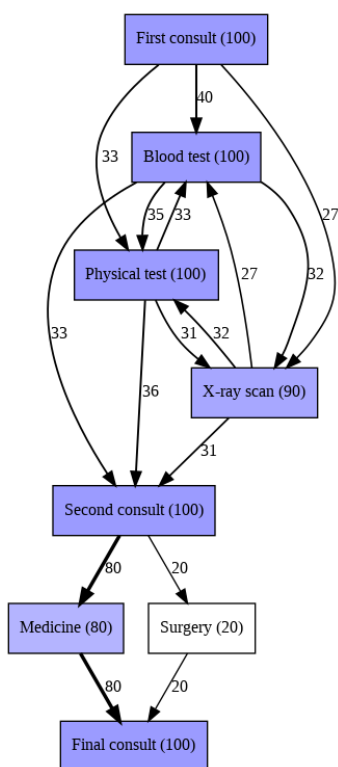


Figura 10 Grafo de seguimiento directo (DFG)

Tomado de https://medium.com/@c3_62722/process-mining-with-python-tutorial-a-healthcare-application-part-3-cc9af986c122.

4.2.Desarrollo Backend de la herramienta (Pymes+Plus)

Como se ha podido observar en figuras anteriores (Figuras 9, 10), la librería PM4PY cuenta con muchas ventajas al momento de su utilización, ya que brinda algoritmos para el descubrimiento de procesos y para el chequeo de conformidad que ayudan a reducir el tiempo a la hora de analizar el comportamiento de un proceso a través de un event log, pero a partir de diversas pruebas realizadas, fueron encontrados algunos inconvenientes como por ejemplo, que los modelos descubiertos no representan fielmente el comportamiento de los procesos aunque las métricas de chequeo de conformidad digan lo contrario, ya que, muchas veces no se tiene en cuenta la finalización de las actividades, por lo que se toman como instantáneas al momento de reproducir la secuencia de ejecución de los casos; y además, tampoco se tiene en cuenta cómo interactúan los recursos en la ejecución del proceso, lo que genera dificultades a la hora de hallar cuellos de botella.

Debido a esto, se decide desarrollar con python una herramienta que pueda ser capaz de adaptarse al comportamiento de los procesos (descubrir procesos y chequeo de conformidad), teniendo en cuenta las relaciones existentes entre las actividades y los recursos, y la cantidad disponible de estos últimos; y que además también sea capaz de hallar las posibles variantes existentes y determinar cuál es mejor económica y temporalmente.

4.2.1. Preliminares

El paso principal para poder llevar a cabo esta propuesta, sin lugar a duda fue modificar el event log para añadir marcas de tiempo de finalización a cada evento o actividad (Figura 11), ya

que, el poder conocer los momentos en los que una actividad está siendo realizada, logra aportar la información necesaria sobre los recursos que están siendo utilizados y posibles cuellos de botella.

	id	action	resource	datetime	endaction
0	paciente 0	Primera Consulta	Dra. Anna	2017-01-02 11:40:11	2017-01-02 11:47:11
1	paciente 1	Primera Consulta	Dra. Anna	2017-01-02 11:47:35	2017-01-02 11:53:35
2	paciente 0	Examen de Sangre	Bacte. Andrea	2017-01-02 11:49:33	2017-01-02 11:50:33
3	paciente 2	Primera Consulta	Dra. Anna	2017-01-02 11:53:49	2017-01-02 12:01:49
4	paciente 3	Primera Consulta	Dra. Anna	2017-01-02 12:02:00	2017-01-02 12:13:58

Figura 11 *Extracto de Event log modificado*

Elaboración propia.

De esta manera, mediante ciencia de datos este nuevo event log es analizado para encontrar solo la información que resulta ser la más importante, que para este caso son la duración de las actividades y la secuencia de actividades que fueron ejecutadas para cada caso (Figura 12).

	id	action	resource	datetime	endaction	duration	action_sequence
0	patient 0	First consult	Dr. Anna	2017-01-02 11:40:11	2017-01-02 11:47:11	0 days 00:07:00	First consult ⇒ Blood test ⇒ Physical test ⇒ S...
1	patient 0	Blood test	Lab	2017-01-02 11:49:33	2017-01-02 11:50:33	0 days 00:01:00	First consult ⇒ Blood test ⇒ Physical test ⇒ S...
2	patient 0	Physical test	Nurse Jesse	2017-01-02 12:53:50	2017-01-02 12:58:50	0 days 00:05:00	First consult ⇒ Blood test ⇒ Physical test ⇒ S...
3	patient 0	Second consult	Dr. Ben	2017-01-02 16:21:06	2017-01-02 16:28:06	0 days 00:07:00	First consult ⇒ Blood test ⇒ Physical test ⇒ S...
4	patient 0	Surgery	Dr. Charlie	2017-01-05 13:23:09	2017-01-05 13:33:09	0 days 00:10:00	First consult ⇒ Blood test ⇒ Physical test ⇒ S...
...	...	...	...	...	...	...	...

Figura 12 *Extracto de event log modificado luego de aplicar ciencia de datos*

Elaboración propia.

Al tener toda esta información dentro de un DataFrame, mediante el método `unique` de la librería de Pandas es posible hallar todos los casos, todas las actividades, todos los recursos y todas las posibles variantes diferentes que se encuentran dentro del event log, para poder almacenarlos en listas donde ningún valor se encuentre repetido, esto es representado en la Figura 13.

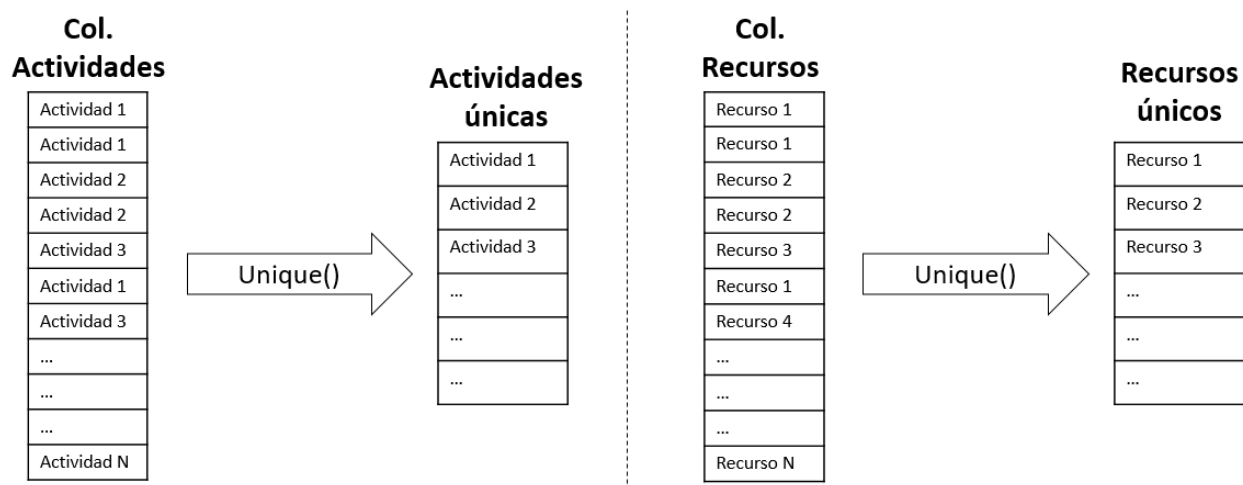


Figura 13 *Valores únicos*
Elaboración propia.

Por otra parte, también se debe determinar la cantidad de recursos disponibles y la relaciones que pueden tener con las distintas actividades, para esto, mediante una tabla cruzada (representada en la Figura 14) es posible observar que cada una de las actividades tiene relación con un grupo, de por lo menos un recurso, estos grupos de recursos también serán almacenados en varias listas que serán utilizadas más adelante. Llegado el caso de que se presenten actividades que compartan el mismo grupo de recursos estos son filtrados para evitar que existan grupos de recursos duplicados.

resource	Dr. Anna	Dr. Ben	Dr. Bob	Dr. Charlie	Dr. Rudy	Lab	Nurse Corey	Nurse Jesse	Pharmacy	Team 1	Team 2
action											
Blood test	0	0	0	0	0	10	0	0	0	0	0
Final consult	4	4	2	0	0	0	0	0	0	0	0
First consult	4	3	3	0	0	0	0	0	0	0	0
Medicine	0	0	0	0	0	0	0	0	6	0	0
Physical test	0	0	0	0	0	0	4	6	0	0	0
Second consult	5	2	3	0	0	0	0	0	0	0	0
Surgery	0	0	0	1	3	0	0	0	0	0	0
X-ray scan	0	0	0	0	0	0	0	0	0	4	4

```
[[ 'Dr. Anna', 'Dr. Ben', 'Dr. Bob'],
[ 'Lab'],
[ 'Nurse Corey', 'Nurse Jesse'],
[ 'Dr. Charlie', 'Dr. Rudy'],
[ 'Team 1', 'Team 2'],
[ 'Pharmacy']]
```

Figura 14 *Recursos por Acciones*

Elaboración propia.

4.2.2. Modelado de Procesos (Descubrimiento)

Para generar un modelo que pueda adaptarse al comportamiento de procesos, y pueda ser analizado correctamente, se decidió generar un modelo del proceso basado en la estructura de las redes de Petri.

4.2.2.1. Transiciones

Según la teoría de las redes de Petri, se necesita conocer la cantidad de transiciones, la cantidad de lugares y la marcación inicial, para elaborar una red. Las transiciones son las actividades que están encargadas de aplicar cambios a los estados de los lugares.

Para este caso, todas las transiciones necesarias pueden ser creadas (Figura 15), ya que previamente han sido halladas todas las actividades o eventos diferentes con los que cuenta el proceso.

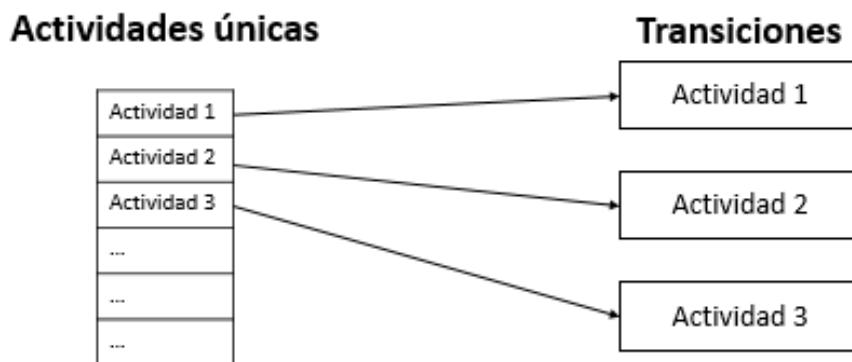


Figura 15 *Actividades como Transiciones*
Elaboración propia.

4.2.2.2. Lugares de Espera

Siguiendo con la teoría de las redes de Petri, cada Transición debe tener por lo menos un lugar de entrada y un lugar de salida, con esto en cuenta, por el momento hay que centrarse en los lugares de entrada, que para fines prácticos llamaremos como “Lugares de Espera”, ya que crearemos uno por cada Transición (Figura 16). Estos lugares servirán para controlar el flujo del proceso hasta llegar al lugar de finalización.

Para poder analizar más adelante el comportamiento del modelo, ya que una red de Petri es un grafo dirigido, se crea una matriz de entradas para asignar una arista entre cada Lugar de Espera y su respectiva Transición asociada.

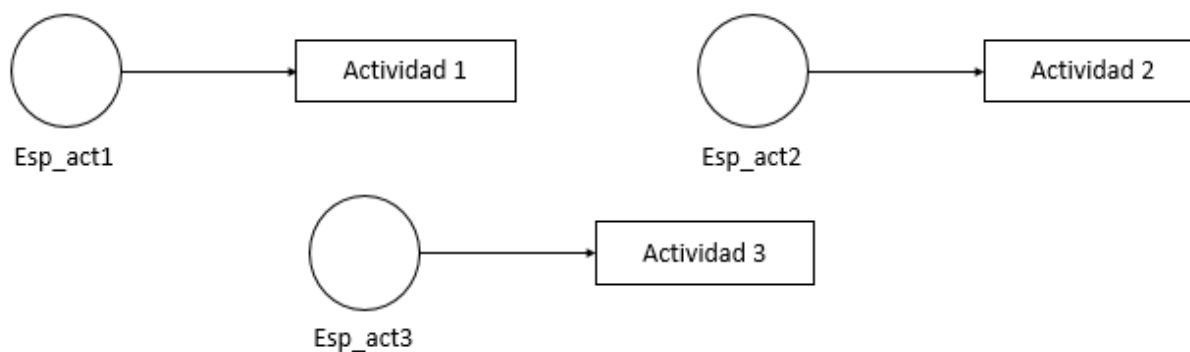


Figura 16 *Lugares de Espera (entrada)*

Elaboración propia.

4.2.2.3. Lugares de Recursos

Como se pudo observar en figuras anteriores, cada actividad cuenta con grupos de recursos asociados, estos grupos de recursos, también tienen sus lugares dentro de este modelo, ya que permiten la habilitación de las Transiciones cuando hay recursos disponibles, todo esto gracias a la implementación de marcas de tiempo de finalización, porque se puede saber en qué momento hay o no recursos disponibles.

Para poder analizar más adelante el comportamiento del modelo, en la matriz de entradas se asigna una arista entre cada Lugar de Recurso y su respectivas Transiciones asociadas, y así mismo, se crea la matriz de salidas, y se asignan las mismas aristas, ya que, en otras palabras, los Lugares de Recursos funcionan como lugares de entrada y de salida secundarios (Figura 17), ya que los recursos son ocupados y liberados varias veces a lo largo del tiempo de ejecución de los distintos procesos.

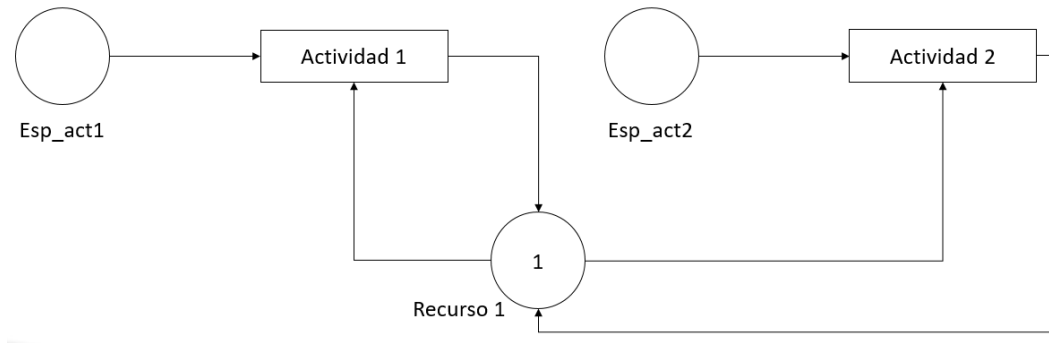


Figura 17 *Lugares de Recursos*

Representación de cómo puede interactuar el mismo grupo de recursos con varias actividades. Elaboración propia.

4.2.3. Chequeo de Conformidad

4.2.3.1. Secuencias

Para que un modelo pueda ser capaz de representar realmente el comportamiento de un proceso, hay que tener en cuenta que secuencias de actividades fueron realizadas, para esto, también fueron filtradas desde un principio todas las distintas variantes.

```

['First consult ⇒ Blood test ⇒ Physical test ⇒ Second consult ⇒ Surgery ⇒ Final consult',
'First consult ⇒ Physical test ⇒ Blood test ⇒ X-ray scan ⇒ Second consult ⇒ Medicine ⇒ Final consult',
'First consult ⇒ Physical test ⇒ X-ray scan ⇒ Blood test ⇒ Second consult ⇒ Medicine ⇒ Final consult',
'First consult ⇒ Blood test ⇒ X-ray scan ⇒ Physical test ⇒ Second consult ⇒ Medicine ⇒ Final consult',
'First consult ⇒ Blood test ⇒ Physical test ⇒ X-ray scan ⇒ Second consult ⇒ Medicine ⇒ Final consult',
'First consult ⇒ Physical test ⇒ X-ray scan ⇒ Blood test ⇒ Second consult ⇒ Surgery ⇒ Final consult',
'First consult ⇒ Physical test ⇒ Blood test ⇒ X-ray scan ⇒ Second consult ⇒ Surgery ⇒ Final consult',
'First consult ⇒ X-ray scan ⇒ Blood test ⇒ Physical test ⇒ Second consult ⇒ Medicine ⇒ Final consult',
'First consult ⇒ Physical test ⇒ Blood test ⇒ Second consult ⇒ Surgery ⇒ Final consult']

```

Figura 18 *Variantes (Secuencias diferentes)*

Ejemplo de las posibles variantes halladas por el programa. Elaboración propia.

4.2.3.2. Lugares de Salida

Debido que las variantes cuentan cada una con un orden de ejecución distinto (Figura 18), al momento de seguir variantes diferentes, cada Lugar de Espera puede convertirse en un lugar de salida para cualquier Transición anterior, lo que permite que el modelo pueda adaptarse de la mejor manera a todas las variantes de un proceso, esto es representado en la Figura 19.

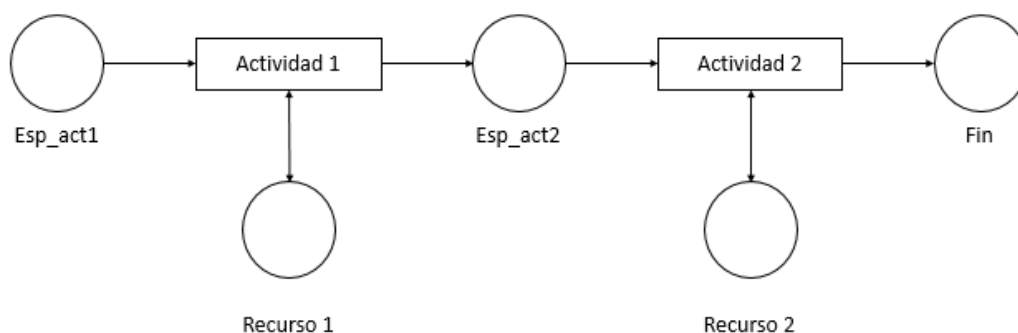


Figura 19 Agregando lugares de salida
Elaboración propia.

Para llevar el control de las conexiones que son asignadas, dentro de la matriz de salidas se coloca un 1 para resaltar que hay una arista entre una transición y sus posibles lugares de salida (Lugar de Espera de cualquier otra Transición) (Figura 20), este paso se repite con cada variante encontrada, para generar un modelo basado en la estructura de las redes de Petri que cuente con todos los caminos posibles al igual que los gráficos de seguimiento directo (DFG).

Si una variante cuenta con una sub-secuencia similar a otra y está ya ha sido señalada con un 1 dentro de la matriz, no hace falta volver a asignar la misma arista de nuevo, por lo que se puede seguir con la siguiente, de esta manera, al finalizar de analizar las variantes es necesario

agregar 0 a las celdas vacías para representar que no existe ninguna relación entre una Transición y un Lugar.

Variantes:

1. ~~T1~~ ⇒ ~~T3~~ ⇒ ~~T4~~ ⇒ ~~T2~~ ✓

2. ~~T1~~ ⇒ ~~T2~~ ⇒ ~~T3~~ ⇒ ~~T4~~ ✓

3. T1 ⇒ T4 ⇒ T2 ⇒ T3

		Lugares							
		Espera				Recursos			Fin
		ET1	ET2	ET3	ET4	R1	R2	R3	
Transiciones	T1		1	1		1			
	T2			1			1		1
	T3				1	1			
	T4		1					1	1

Figura 20 *Matriz de salidas*

Elaboración propia.

Al finalizar de analizar las variantes y terminar de asignar las aristas dentro de las matrices de entradas y salidas, el modelo ha sido terminado y representa todas las rutas de ejecución que un proceso puede seguir, lo que quiere decir que, el modelo puede reproducir todas las posibles variantes que presente un proceso. En la Figura 21, se muestra una representación visual de todos los posibles caminos que fueron encontrados a partir del event log de casos hospitalarios que fue tomado como ejemplo para la realización de este trabajo.

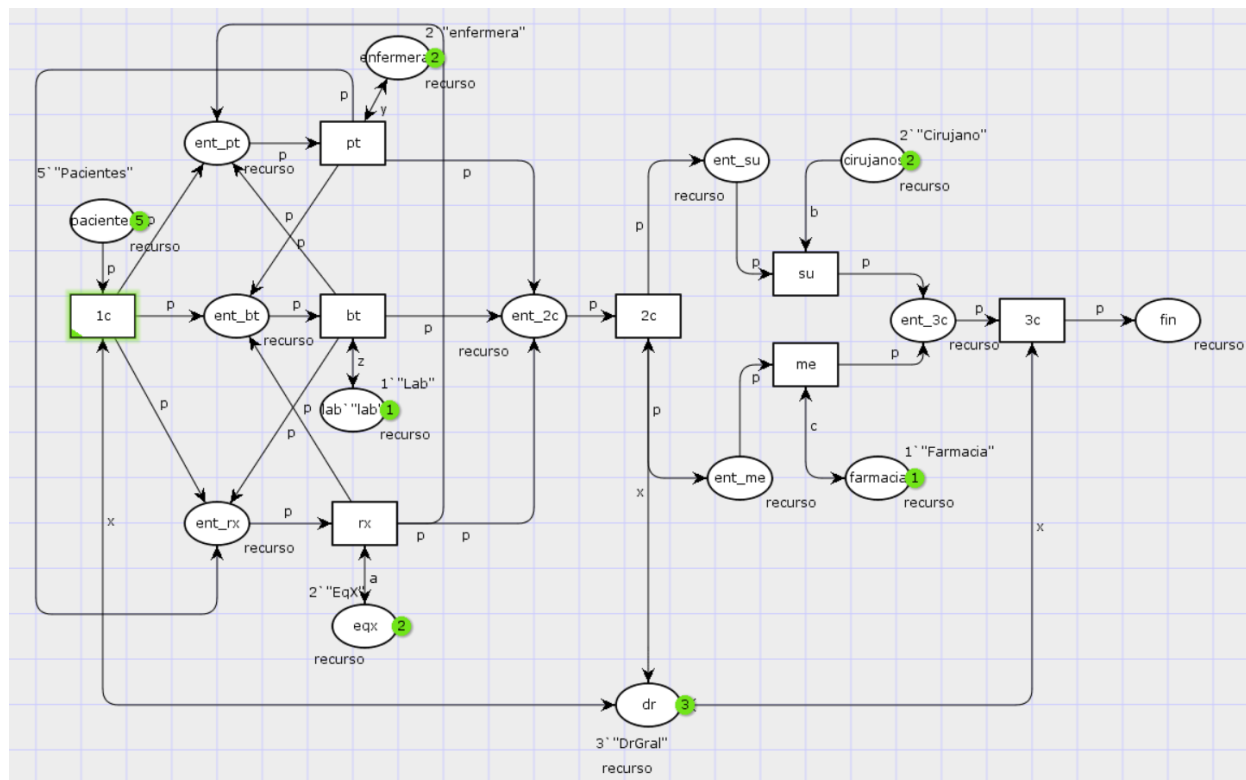


Figura 21 Representación del modelo terminado
Elaboración propia.

4.2.4. Cuellos de Botella

4.2.4.1. Casos por Variante

Para que el modelo pueda replicar el comportamiento del proceso adecuadamente para poder ser analizado, se necesita saber que secuencia sigue cada caso, cabe aclarar, que la relación que hay entre los casos y las variantes es 1 a muchos (Figura 22), por lo que un caso, solo puede seguir una variante, pero una variante puede ser seguida por muchos casos.

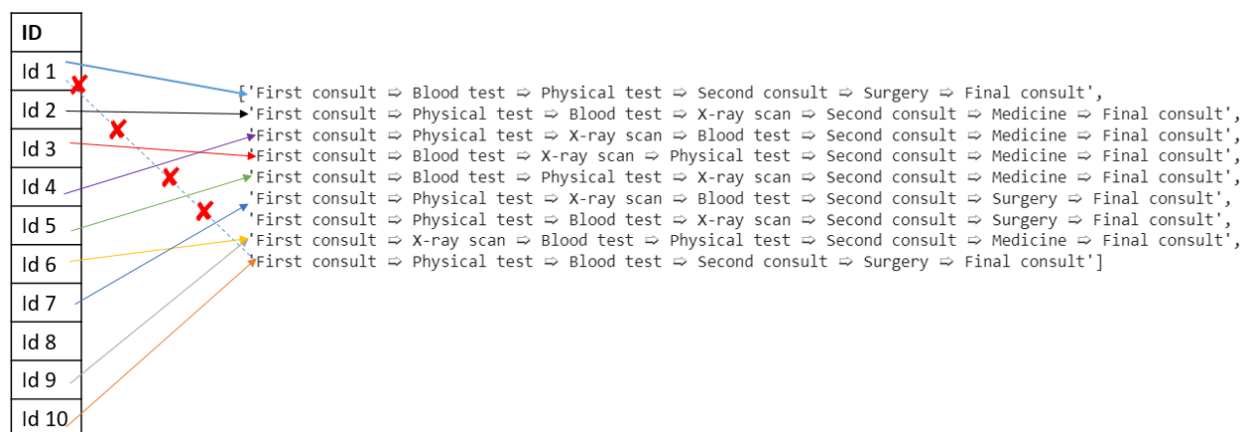


Figura 22 *Relación de Casos por Variante*
Elaboración propia.

4.2.4.2. Análisis Forense de Procesos

Debido a que se requiere replicar el comportamiento del proceso a lo largo del tiempo, es necesario hacer una reconstrucción de los hechos, de esta manera se pretende obtener el orden cronológico desde que el event log registra que el primer caso da inicio al proceso al instante de realizar su primera actividad, hasta que se registre que todos los casos han finalizado por completo con sus actividades, en la Figura 23 es representada la forma en la que se transforma el event log original a un nuevo DataFrame que contiene el orden cronológico de inicio y fin de cada una de las actividades.

Esto es posible ya que, al agregar marcas de tiempo de inicio y fin a cada actividad, se puede determinar en qué momento hay actividades en ejecución, ya sean o no, ejecuciones simultáneas de la misma o distintas actividades. De esta manera, también se puede determinar la disponibilidad de los recursos a lo largo del tiempo, lo que facilita en gran medida la forma de encontrar cuellos de botella.

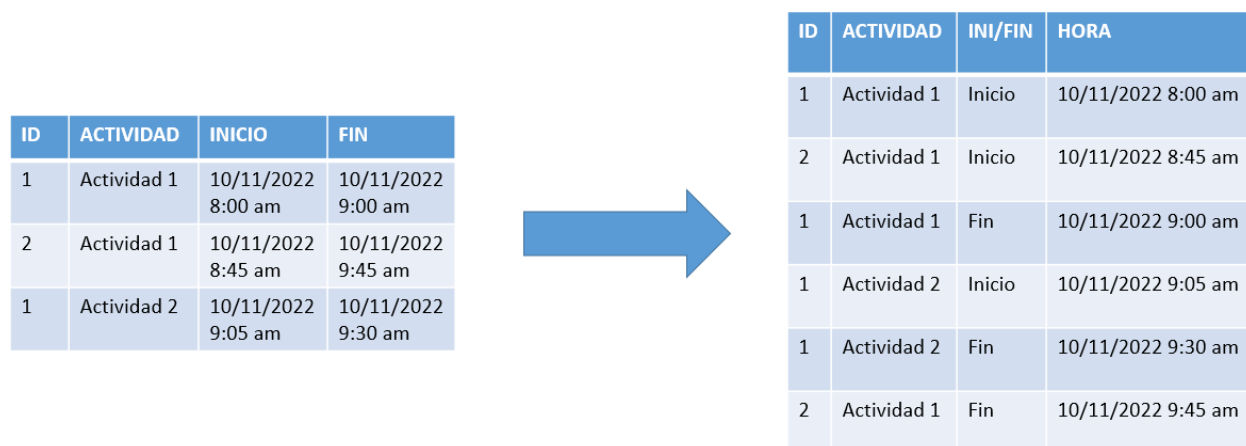


Figura 23 *Reconstrucción de los hechos*
Elaboración propia.

4.2.4.3. Motor de Ejecución Forense de Procesos

Al ya tener secuenciado por orden cronológico el event log, es momento de reproducirlo con el modelo que se generó anteriormente, este modelo, se ejecuta casi de la misma manera que una red de Petri normal.

El principal cambio, es que cuando una Transición se encuentra habilitada y cuenta con muchos lugares de salida, solo se elige un camino siguiendo la secuencia establecida para cada caso (Figura 24), de esta manera se va actualizando la respectiva marcación de los Lugares de Espera y de Recursos, teniendo en cuenta que cuando una actividad inicia, los recursos son ocupados y se descuentan de la cantidad disponible, para que al finalizar la actividad este vuelva a ser añadido, sin modificar la marcación de los demás lugares (Figura 25).

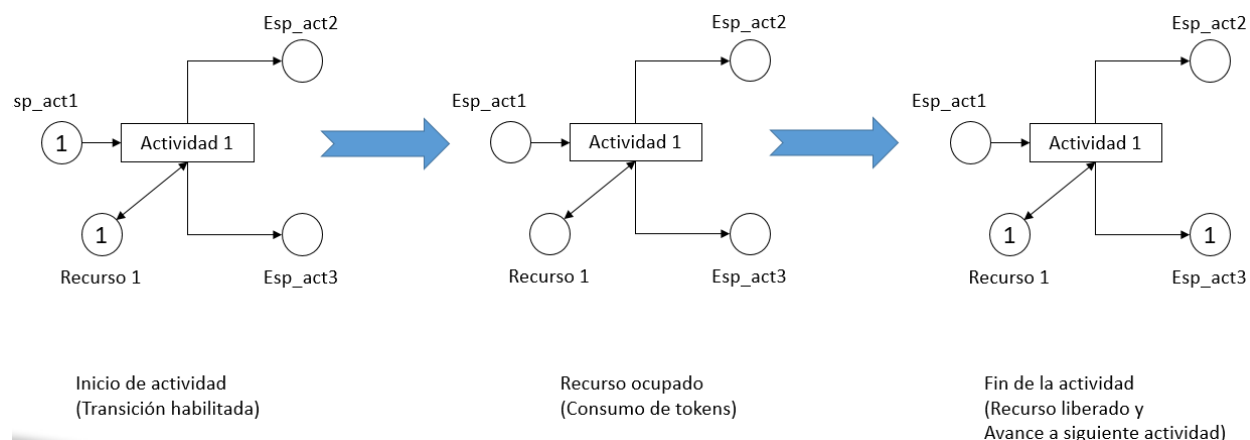


Figura 24 *Selección de caminos*
Elaboración propia.

transicion	InicioFin	esp_First consult	esp_Blood test	esp_Physical test	esp_Second consult	esp_Surgery	esp_Final consult	esp_X-ray scan	esp_Medicine	rec1	rec2	rec3	rec4	rec5	rec6	fin
0	u0 Marca Inicial	10	0	0	0	0	0	0	0	3	1	2	2	2	1	0
1	First consult Inicio	10	0	0	0	0	0	0	0	2	1	2	2	2	1	0
2	First consult Fin	9	1	0	0	0	0	0	0	3	1	2	2	2	1	0
3	First consult Inicio	9	1	0	0	0	0	0	0	2	1	2	2	2	1	0
4	Blood test Inicio	9	1	0	0	0	0	0	0	2	0	2	2	2	1	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...

Figura 25 *Extracto de la Marcación*
Elaboración propia.

Cuando se ha terminado de reproducir todas las secuencias de actividades de los distintos casos, se analiza la marcación para poder identificar los momentos en los que se encuentra en espera una gran cantidad casos y los recursos no son suficientes para atenderlos. En la Figura 26, se presenta una tabla con el resumen de cuellos de botella que fueron detectados a lo largo del análisis para cada actividad.

Al tener identificados estos momentos, es posible recomendar si es necesario o no modificar la cantidad de ciertos recursos, para obtener mejores resultados en futuras ejecuciones.

Accion	Cant. En Espera	Recurso	Cant. Recurso Libre
esp_First consult	7	rec1	1
esp_Blood test	3	rec2	0
esp_Physical test	3	rec3	1
esp_Second consult	0	rec1	1
esp_Surgery	2	rec4	1
esp_Final consult	0	rec1	1
esp_X-ray scan	4	rec5	1
esp_Medicine	1	rec6	0

Figura 26 *Cuellos de Botella*
Elaboración propia.

4.2.4.4.API

Para este punto ya han sido terminados métodos capaces de aplicar las principales técnicas de minería de procesos, pero que todavía resultarían muy complicados de utilizar para un usuario que no tenga los conocimientos necesarios de programación, ni de minería de procesos, razón por la cual se decide utilizar la librería FastAPI de Python para desarrollar una API, que mediante la utilización de JWT (Json Web Tokens) pueda autenticar a los usuarios registrados, y autorizarlos para que puedan hacer uso de la plataforma, y así controlar un poco la seguridad de la misma

De esta manera, cuando un usuario, ha sido autenticado y autorizado puede subir un archivo “.CSV” que contenga un event log de sus procesos, y realizar una solicitud para comenzar con el análisis. Cuando el análisis termina, el usuario recibirá información detallada sobre que variante podría ser la más óptima para su proceso, además de poder identificar en que parte de sus

procesos son detectados cuellos de botella, y, por último, el usuario también recibirá algunas recomendaciones que podrían ayudar a solucionar las ineficiencias detectadas en sus procesos.

AUTH			^
POST	/signup	Sign Up	✓ 
POST	/token	Login	✓
GET	/	Home	✓ 
PM			^
GET	/files	Get Files	✓ 
GET	/files/{id}	Get A File	✓ 
GET	/file_to_process/{id}/{days}/{hours}/{costo}	Get File To Process	✓ 
POST	/upload	Upload File	✓ 
DELETE	/delete/{id}	Delete File	✓ 

Figura 27 API - Process Mining
Elaboración propia.

4.2.4.5. Subida a Railway

Para la realización de pruebas, se optó por desplegar la API en la plataforma de Railway, ya que, permite el despliegue de múltiples aplicaciones con solo vincular un repositorio de GitHub y la realización de una serie de configuraciones básicas (versión del lenguaje de programación

utilizado, selección de puertos, el dominio donde se va a desplegar, etc.). Esto con el fin de poder utilizar la API en cualquier lugar.

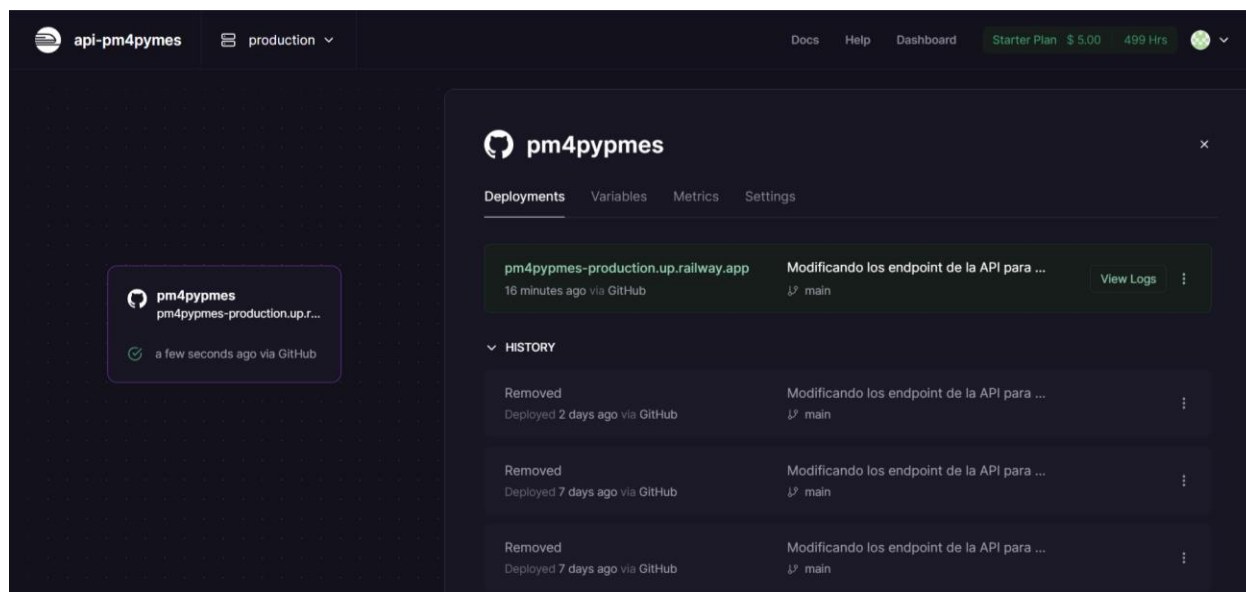


Figura 28 *Dashboard Railway*
Elaboración Propia.

Gracias a la utilización de Railway, la realización de cambios y modificaciones fue mucho más rápida, ya que, al estar vinculado a un repositorio de GitHub, al subir los cambios con un push, Railway desplegaba de manera instantánea la nueva versión, guardando también el historial de versiones, por si se requería comparar (Figura 28).

4.3.Desarrollo Frontend de Pymes+Plus

Debido a que las API devuelven texto en formato JSON, este debe ser procesado para poder leer y comprender la información de la manera correcta, por tal motivo, también ha sido diseñada

una página web para que los usuarios puedan acceder a la información de sus procesos de una manera más intuitiva y sencilla.

4.3.1. Boceto

Hacer un boceto por lo general tiende a pasarse por alto al momento de diseñar una página web, pero en realidad se trata de una etapa necesaria, ya que es una versión esquemática que visualmente permite la organización de las ideas y la aplicación de cambios de manera flexible, por otra parte, también se puede evitar el desarrollo de trabajos innecesarios.

A continuación, se presentan algunos de los bocetos que fueron diseñados para la página web mediante la plataforma de Balsamiq.

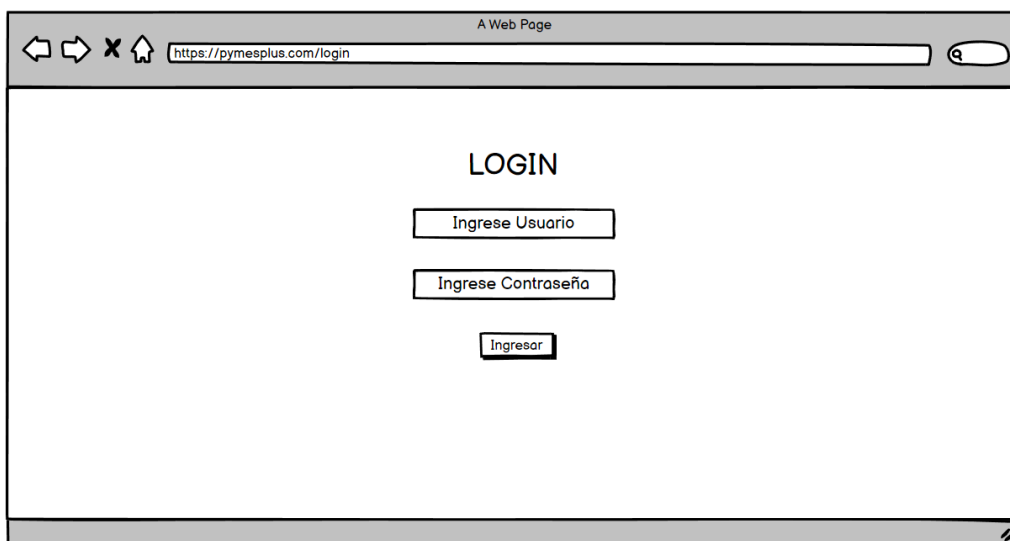


Figura 29 *Boceto Login*
Elaboración propia.

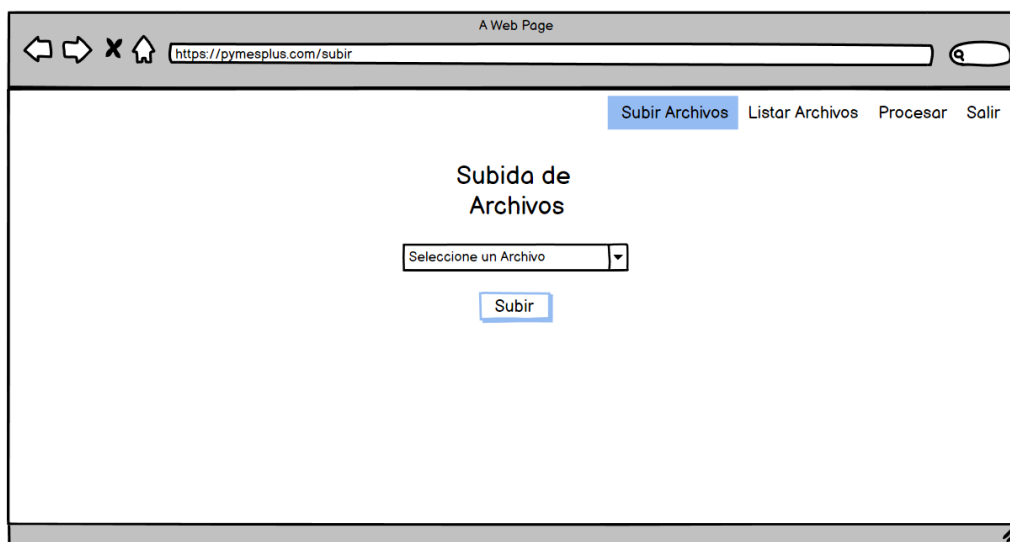


Figura 30 *Boceto Página de subida de Archivos*
Elaboración propia.

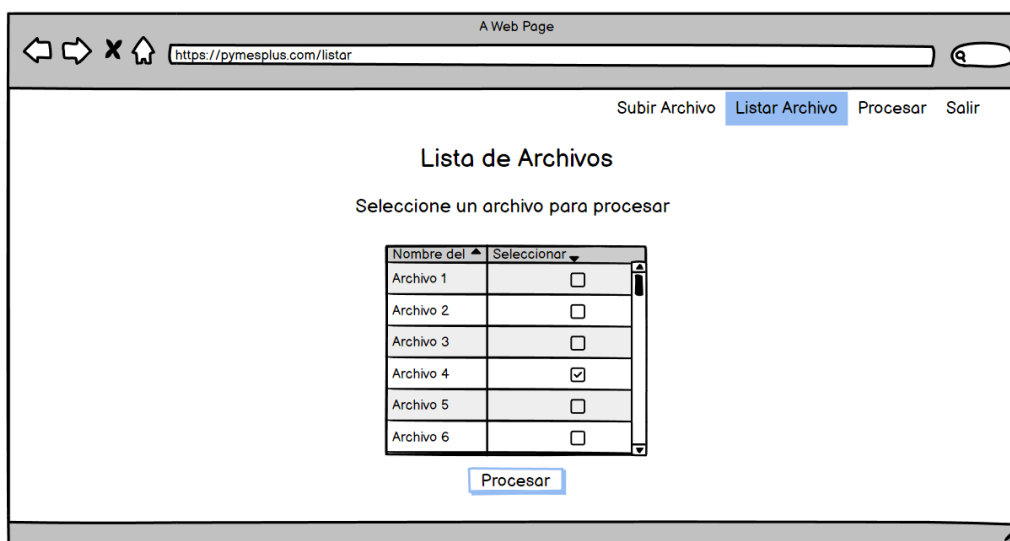


Figura 31 *Boceto Página de Listado de Archivos*
Elaboración propia.

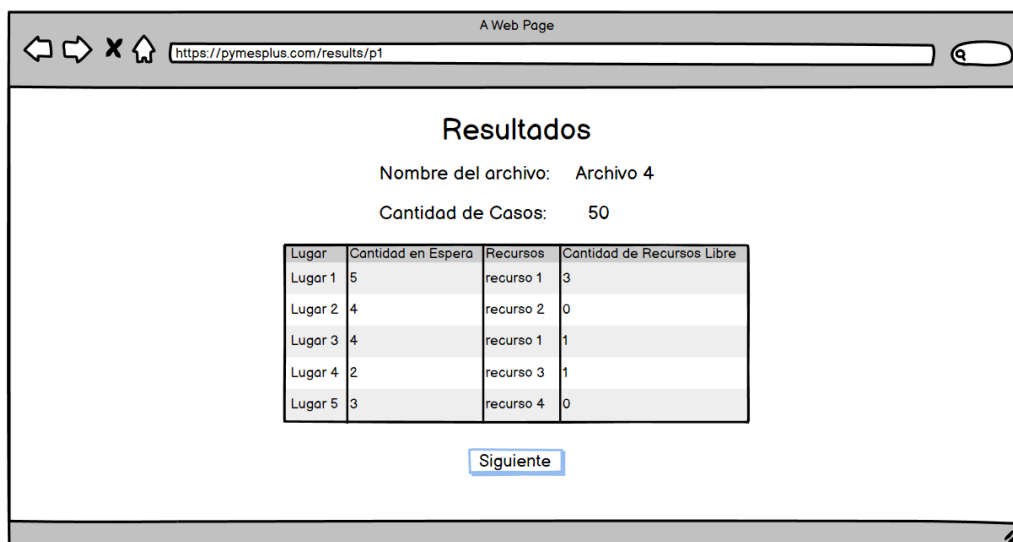


Figura 32 *Boceto Página de Resultados 1*
Elaboración propia.

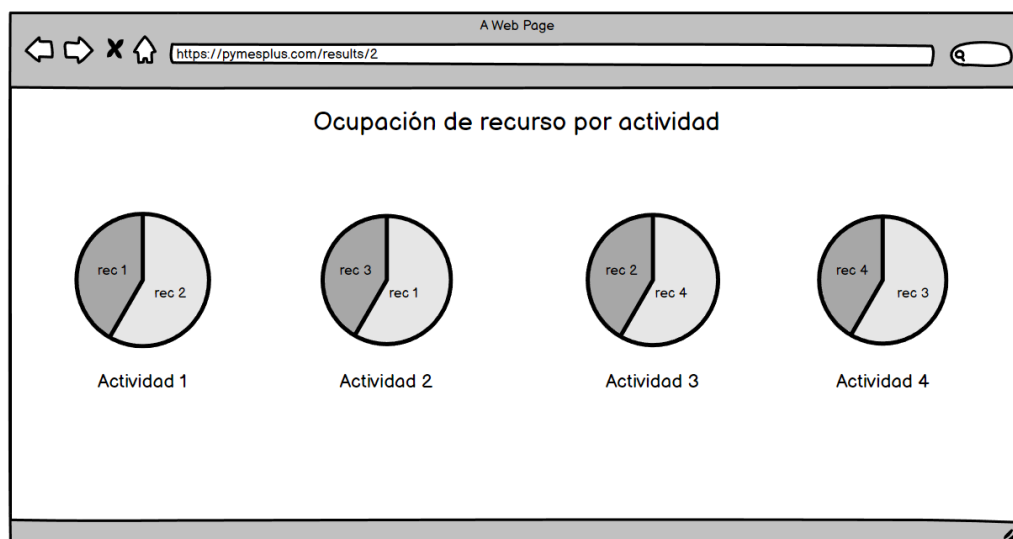
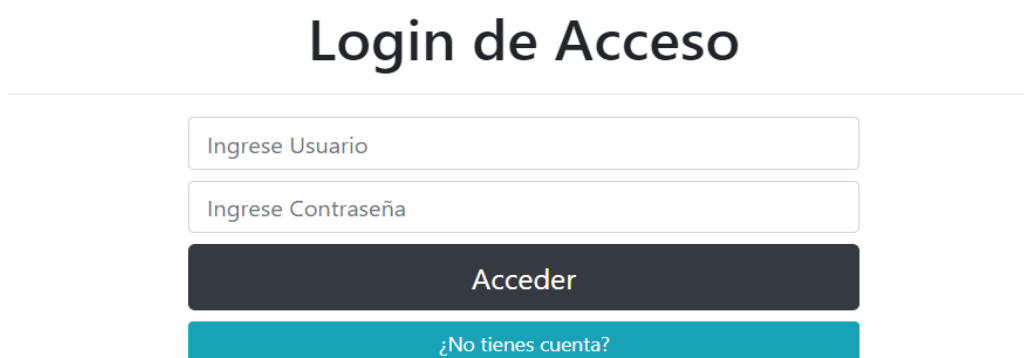


Figura 33 *Boceto Página de Resultados 2*
Elaboración propia.

4.3.2. Desarrollo de la interfaz

La interfaz gráfica de la plataforma fue desarrollada con React JS, ya que facilita mucho el desarrollo de interfaces de usuario; y por otra parte fue estilizada con clases de Bootstrap 4 para darle un acabado que sea más amigable con el usuario.

El Login de Acceso mostrado en la figura 34, resulta necesario ya que, al momento de desarrollar la API, se implementó un sistema de autenticación por tokens para tener seguridad dentro de la plataforma.



The image shows a login form titled "Login de Acceso". It consists of two input fields: "Ingrese Usuario" and "Ingrese Contraseña". Below these fields is a dark grey button labeled "Acceder". At the bottom is a teal button labeled "¿No tienes cuenta?".

Figura 34 *Login de Acceso*
Elaboración propia.

De esta manera, cuando un usuario ha sido autenticado, tiene la posibilidad de subir varios archivos “.CSV” (Figura 35) que contengan sus event log. Cuando sea requerido realizar un análisis a alguno de estos archivos, estos se encontrarán disponibles en la sección de listado de archivos (Figura 36), los cuales, al ser seleccionados dirigirán al usuario a la sección de procesamiento.



PYMES+Plus Inicio Listar Archivos **Subir Archivo** Salir

Subida de Archivos

Seleccione el archivo con extensión ".CSV" que desea subir

Seleccionar archivo log_textileria.csv

Subir Archivo

Figura 35 *Subida de Archivos*
Elaboración propia.



PYMES+Plus Inicio **Listar Archivos** Subir Archivo Salir

Lista de Archivos

Selecciona un archivo para empezar a procesar

Actualizar Lista

ID:
6387d469ab119d8512148f8
5
Nombre: pm_es.csv

Figura 36 *Listado de Archivos*
Elaboración propia.

Dentro de la sección de procesamiento (Figura 37), se encuentra un formulario para conocer cuál es la duración promedio del proceso (sin contar los tiempos muertos) cuando no hay cuellos de botella y cuánto cuesta esta ejecución, esto para hallar una variante que cuente con los tiempos y costos más cercanos a los entregados por el usuario, siendo esto el primer paso para poder entregar el análisis completo de resultados que además de la mejor variante incluye,

generalidades del archivo (Figura 38), resumen de cuellos de botella (Figura 39) y la ocupación de recursos por actividad (Figura 40), que mediante varias gráficas puede ayudar al momento de decidir si es necesario o no modificar la cantidad de los recursos para obtener mejores resultados en futuras ejecuciones del proceso.



The screenshot shows the 'Procesar Archivo' (Process File) interface of the PYMES+Plus application. The header bar is dark blue with the PYMES+Plus logo on the left and navigation links 'Inicio', 'Listar Archivos', 'Subir Archivo', and 'Salir' on the right. The main content area has a white background. At the top, it displays the title 'Procesar Archivo' and a unique ID: 'ID: 6387d469ab119d8512148f85'. Below this, a instruction reads: 'Ingrese la hora como se muestra a continuación, Ej: Horas "01:15:00"'. The form contains three input fields: 'Ingrese Cantidad de días' with the value '0', 'Ingrese Cantidad de horas' with the value '06:00:00', and 'Ingrese Costo Total' with the value '150000'. The 'Costo Total' field is highlighted with a blue border. At the bottom of the form is a dark blue button labeled 'Procesar'.

Figura 37 *Formulario para procesamiento*
Elaboración propia.



Figura 38 Resultados: Generalidades y Mejor Variante
 Elaboración propia.

Cuello de Botella			
ACCION	CANT. EN ESPERA	RECURSO	CANT. RECURSO LIBRE
esp_First consult	7	rec1	1
esp_Blood test	3	rec2	0
esp_Physical test	3	rec3	1
esp_Second consult	0	rec1	1
esp_Surgery	2	rec4	1
esp_Final consult	0	rec1	1
esp_X-ray scan	4	rec5	1
esp_Medicine	1	rec6	0

Figura 39 Resultados: Cuellos de Botella
 Elaboración propia.

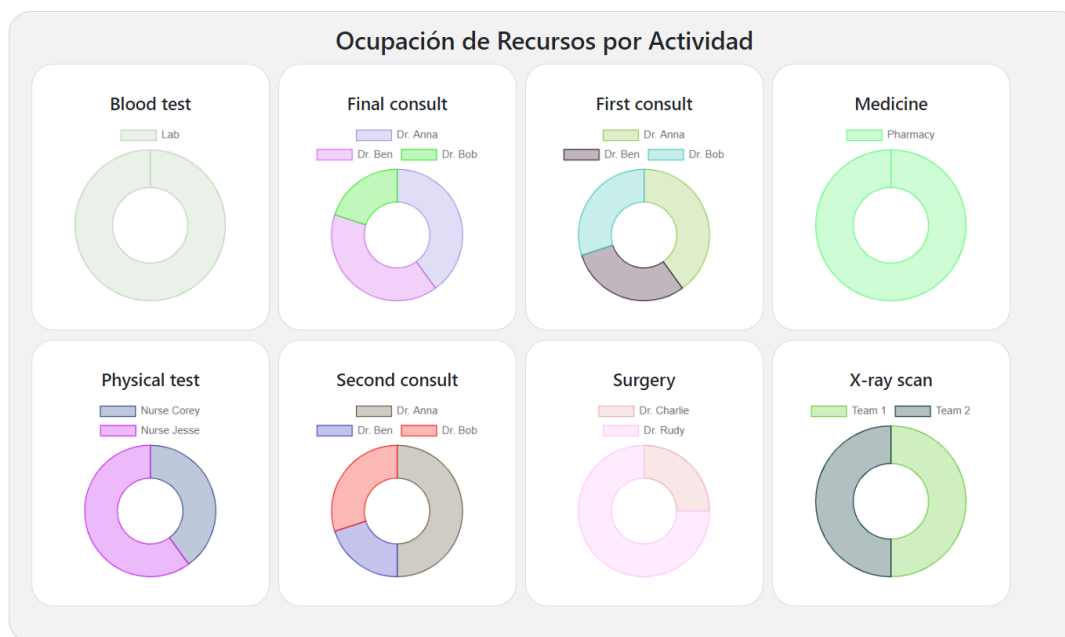


Figura 40 *Resultados: Ocupación de Recursos por Actividad*
Elaboración propia.

4.4.Observaciones

Actualmente, sin duda alguna la librería de PM4PY cuenta con algoritmos de minería de procesos de última generación, ya que sus desarrolladores (Fraunhofer Institute for Applied Information Technology), están afiliados al Profesor Wil van der Aalst, quien es uno de los precursores y el líder mundial en la investigación de minería de procesos, debido a esta relación, todos los desarrollos novedosos son integrados a esta librería en el menor tiempo posible. A lo largo de la realización de este proyecto, fue posible encontrar varias ventajas que trae consigo la utilización de PM4PY, pero del mismo modo, también fueron halladas varias desventajas que impedían el cumplimiento de los objetivos propuestos para este proyecto, razón por la cual, se tomó la difícil decisión de dejar de lado esta librería para comenzar desde cero el desarrollo de una herramienta que sí pudiera cumplir con todos los objetivos propuestos.

Tabla 2 *PM4PY: Ventajas y Desventajas*

Ventajas	Desventajas
Algoritmos de descubrimiento de procesos, capaces de generar diferentes modelos de procesos.	Los modelos descubiertos no representan exactamente el comportamiento de los procesos.
Algoritmos de chequeo de conformidad para la medición de aptitud, precisión, generalización y simplicidad de los modelos descubiertos.	No es tomada en cuenta la interacción entre actividades y recursos.
Análisis de rendimiento de actividades por trazas.	Rara vez se tiene en cuenta la finalización de las actividades.
Calculo de estadísticas: Proporción de llegada/dispersión de los casos, tiempos de ejecución y de espera, entre otros.	Dificultad para hallar cuellos de botella.
Redes de traspaso de recursos (redes complejas)	

Elaboración propia.

Esto no quiere decir que la librería PM4PY no funcione de la manera correcta, todo lo contrario, gracias a su utilización en muchas de las pruebas realizadas para este proyecto, fue posible entender cómo funcionan internamente los algoritmos de descubrimiento de procesos y de chequeo de conformidad, los cuales sirvieron de inspiración para el desarrollo de los algoritmos principales de la herramienta “Pymes+Plus” con la cual fue posible cumplir con la totalidad de los objetivos propuestos.

Tabla 3 *PM4PY vs Pymes+Plus*

Desventajas PM4PY	Herramienta Propuesta (Pymes+Plus)
Los modelos descubiertos no representan exactamente el comportamiento de los procesos.	Modelos generados inspirados en DFG y redes de Petri, que logran adaptarse al comportamiento de los procesos.
No es tomada en cuenta la interacción entre actividades y recursos.	Se logra detectar grupos de recursos y su respectiva cantidad disponible.
Rara vez se tiene en cuenta la finalización de las actividades.	Los modelos logran representar las relaciones entre actividades y recursos.
Dificultad para hallar cuellos de botella.	Se logra detectar las variantes existentes y se recomienda la mejor.

Se logra identificar cuellos de botella, gracias a
añadir marcas de tiempo de finalización de
actividades

Elaboración propia.

5. CONCLUSIONES

A partir de la investigación realizada sobre Minería de Procesos y los algoritmos de sus diversas técnicas, fue posible cumplir con el objetivo principal de este trabajo, que consistía en desarrollar “Pymes+Plus”, una plataforma capaz de aplicar técnicas de minería de procesos, con la finalidad de ayudar a las pequeñas y medianas empresas a comprender, gestionar y mejorar sus procesos organizacionales de manera fácil, rápida y personalizada, ya que las herramientas de minería de procesos disponibles actualmente tienden a ser muy complicadas de utilizar.

Aunque la plataforma desarrollada es una primera versión, es capaz de implementar técnicas de minería de procesos a partir de la aplicación de ciencia de datos, ya que fue desecheda la idea de utilizar la librería PM4PY, porque se presentaron varios inconvenientes que impedían el cumplimiento de los objetivos propuestos a la hora de realizar pruebas. El aplicar un poco de ciencia de datos para procesar los event logs por completo, resultó ser una muy buena idea, ya que con la librería de Pandas (enfocada en el manejo de datos), se facilitó mucho la elaboración de modelos de procesos, los cuales al momento de realizar un análisis forense de procesos, son capaces de representar tanto el comportamiento de las distintas trazas, como el comportamiento de los recursos a lo largo del tiempo; permitiendo de esta manera, identificar los cuellos de botella que afectan el rendimiento de los procesos.

De esta manera, quien utilice la plataforma, recibirá información sobre las ineficiencias que presentan sus procesos (cuellos de botella, actividades repetidas, etc.) y algunas recomendaciones (variante más óptima, o variaciones en la cantidad de recursos de algunas actividades) que ayudarían a las pequeñas y medianas empresas a mejorar el rendimiento de sus procesos para ejecuciones futuras.

6. TRABAJOS FUTUROS

Ya que con la realización de este trabajo se buscaba crear un prototipo de una plataforma capaz de aplicar técnicas de minería de procesos, para que todas las personas pudieran conocer, gestionar y mejorar sus procesos organizacionales de manera fácil, rápida y personalizada, en un futuro cercano, es posible implementar algunas de las siguientes mejoras:

- Optimización de los métodos para identificar cuellos de botella, ya sea, agregando un umbral mínimo de tiempo, para que el tiempo que se tarda un recurso para reponerse después de finalizar una actividad, no sea tomado en cuenta si el mismo recurso vuelve a ser requerido de manera inmediata. O, modificando el algoritmo para que este pueda comprender, cuando son utilizados varios recursos diferentes para una misma actividad
- Optimización en los métodos de recomendaciones, para que el sistema pueda detectar los momentos en los que podría ser necesario agregar recursos temporales para evitar cuellos de botella en “horarios pico”.
- Agregar algunas nuevas funcionalidades como la agrupación de variantes con resultados similares (mismas actividades, diferente orden) para la recomendación de rutas de procesos paralelos, para distribuir de mejor manera tanto los casos como los recursos para evitar que se generen cuellos de botella y para no tener recursos inactivos mucho tiempo.

Debido a que actualmente mucha gente no cuenta con computadores y realizan la gran mayoría de sus actividades diarias desde los dispositivos móviles (teléfonos, tablets), sería de gran

ayuda para ellos el poder tener una aplicación móvil que mediante el uso de la API que fue desarrollada, también tengan acceso a los beneficios aplicar minería de procesos.

Por otro lado, mediante métodos de clustering podría ser posible agrupar por áreas todos los procesos que hayan sido probados, para poder recomendar las distintas formas de realizar algún proceso en específico; y además, con la implementación de una inteligencia artificial que pueda comprender el lenguaje natural, también podría ser posible que un usuario busque la mejor manera de realizar cualquier proceso, y esta sea capaz de responder indicando la posible mejor variante que encuentre entre todos los procesos que hagan parte del clúster.

REFERENCIAS BIBLIOGRÁFICAS

- Aguirre, H. & Rincón, N. (2015). Minería de procesos: desarrollo, aplicaciones y factores críticos. *Cuadernos de Administración*, 28(50). <https://doi.org/10.11144/javeriana.cao28-50.mpda>
- Appian. (s. f.-a). *Bottleneck - The Process Mining Glossary*. <https://appian.com/process-mining/bottleneck.html>
- Appian. (s. f.-b). *Conformance Checking – The Process Mining Glossary*. <https://appian.com/process-mining/conformance-checking.html>
- Appian. (s. f.-c). *Model Enhancement - The Process Mining Glossary*. <https://appian.com/process-mining/model-enhancement.html>
- Appian. (s. f.-d). *Process Discovery – The Process Mining Glossary*. <https://appian.com/process-mining/process-discovery.html>
- Automation Anywhere. (s. f.). *Descubrimiento de procesos en la automatización*. <https://la.automationanywhere.com/rpa/process-discovery>
- Berti, A., van Zelst, S. J. & van der Aalst, W. M. P. (2019). Process Mining for Python (PM4Py): Bridging the Gap Between Process- and Data Science. *Cornell University - arXiv*. <https://doi.org/10.48550/arxiv.1905.06169>
- c3d3. (2020, 29 julio). Process Mining with Python tutorial: A healthcare application — Part 1. *Medium*. https://medium.com/@c3_62722/process-mining-with-python-tutorial-a-healthcare-application-part-1-ae02027a050

Editorial La República S.A.S. (2020, 6 julio). *En América Latina, solo 45% de las Pyme sobrevive los dos primeros años.* Diario La República.

<https://www.larepublica.co/globoeconomia/en-los-paises-de-la-region-45-de-las-pyme-sobreviven-los-dos-primeros-anos-3027735>

Ekon, E. (2022, 28 julio). *Qué es la mejora de procesos y cuál es su importancia.* Ekon.
<https://ekon.es/blog/mejora-de-procesos-empresas/>

Fraunhofer Institute for Applied Information Technology (FIT). (s. f.). *PM4Py - Process Mining for Python.* PM4PY. <https://pm4py.fit.fraunhofer.de/documentation>

Garcia, C. D. S., Meinheim, A., Faria Junior, E. R., Dallagassa, M. R., Sato, D. M. V., Carvalho, D. R., Santos, E. A. P. & Scalabrin, E. E. (2019). Process mining techniques and applications – A systematic mapping study. *Expert Systems with Applications*, 133, 260-295. <https://doi.org/10.1016/j.eswa.2019.05.003>

Guzzo, A., Joaristi, M., Rullo, A. & Serra, E. (2021). A multi-perspective approach for the analysis of complex business processes behavior. *Expert Systems with Applications*, 177, 114934. <https://doi.org/10.1016/j.eswa.2021.114934>

Kregel, I., Stemann, D., Koch, J. & Coners, A. (2021). Process Mining for Six Sigma: Utilising Digital Traces. *Computers & Industrial Engineering*, 153, 107083. <https://doi.org/10.1016/j.cie.2020.107083>

Marin-Castro, H. M. & Tello-Leal, E. (2021). An end-to-end approach and tool for BPMN process discovery. *Expert Systems with Applications*, 174, 114662. <https://doi.org/10.1016/j.eswa.2021.114662>

- Molero, G., Benítez, E. & Mezura, C. (2017). Interacción humano computadora y minería de datos para la generación y representación de conocimiento útil. *Ciencias de la Información*, 48(1), 3-10.
- Orellana, A. (2018). Integración de técnicas de Minería de procesos para la detección de la variabilidad en procesos hospitalarios desde sistemas automatizados. *Revista Cubana de Informática Médica*, 10(2). <http://scielo.sld.cu/pdf/rcim/v10n2/1684-1859-rcim-10-02-e15.pdf>
- Pasquadibisceglie, V., Appice, A., Castellano, G. & van der Aalst, W. (2022). PROMISE: Coupling predictive process mining to process discovery. *Information Sciences*, 606, 250-271. <https://doi.org/10.1016/j.ins.2022.05.052>
- Pérez, S. (2015). *Minería de procesos* [Tesis de Maestría]. Universidad Pontificia Bolivariana, Medellín.
- Sippl, F., Moriz, T. & Reinhart, G. (2022). A process mining-based approach for stakeholder identification in manufacturing and engineering change management. *Procedia CIRP*, 107, 978-983. <https://doi.org/10.1016/j.procir.2022.05.095>
- van der Aalst, W. (2016). *Process Mining: Data Science in Action* (2.^a ed.). Springer Berlín, Heidelberg.
- Yang, J., Ouyang, C., van der Aalst, W. M., ter Hofstede, A. H. & Yu, Y. (2022). OrdinoR: A framework for discovering, evaluating, and analyzing organizational models using event logs. *Decision Support Systems*, 158, 113771. <https://doi.org/10.1016/j.dss.2022.113771>