



**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**  
**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**  
**CARRERA DE INGENIERÍA EN SISTEMAS**

**DESARROLLO DE UNA PLATAFORMA PARA LA EJECUCIÓN  
DE PROCESOS DE ADECUACIÓN DE LA INFORMACIÓN DE  
FORMA AUTOMÁTICA PARA LA EMPRESA TRACKLINK  
UTILIZANDO WORKER SERVICES**

**Trabajo de integración curricular**

Tipo: Proyecto Técnico

Presentado para optar al grado académico de:

**INGENIERO EN SISTEMAS INFORMÁTICOS**

**AUTORES:**

**MARIO DAVID CHÁVEZ JARAMILLO**

**ISRAEL EDUARDO OÑA ORDOÑEZ**

Riobamba - Ecuador

2021



**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**  
**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**  
**CARRERA DE INGENIERÍA EN SISTEMAS**

**DESARROLLO DE UNA PLATAFORMA PARA LA EJECUCIÓN  
DE PROCESOS DE ADECUACIÓN DE LA INFORMACIÓN DE  
FORMA AUTOMÁTICA PARA LA EMPRESA TRACKLINK  
UTILIZANDO WORKER SERVICES**

**Trabajo de integración curricular**

Tipo: Proyecto Técnico

Presentado para optar al grado académico de:

**INGENIERO EN SISTEMAS INFORMÁTICOS**

**AUTORES: MARIO DAVID CHÁVEZ JARAMILLO**

**ISRAEL EDUARDO OÑA ORDOÑEZ**

**DIRECTOR: ING. JORGE ARIEL MENÉNDEZ VERDECIA**

Riobamba - Ecuador

2021

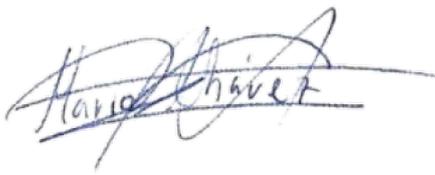
**©2021, Mario David Chávez Jaramillo, Israel Eduardo Oña Ordoñez**

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el Derecho de Autor.

Nosotros, **MARIO DAVID CHÁVEZ JARAMILLO** y **ISRAEL EDUARDO OÑA ORDOÑEZ** declaramos que el presente trabajo de titulación es de nuestra autoría y los resultados de este son auténticos. Los textos en el documento que provienen de otras fuentes están debidamente citados y referenciados.

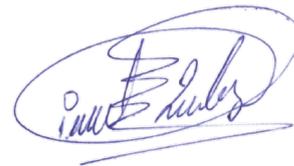
Como autores asumimos la responsabilidad legal y académica de los contenidos de este trabajo de titulación; El patrimonio intelectual pertenece a la Escuela Superior Politécnica de Chimborazo.

Riobamba, 29 de octubre de 2021



Mario David Chávez Jaramillo

0603311499



Israel Eduardo Oña Ordoñez

0603784653

**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**  
**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**  
**CARRERA DE INGENIERÍA EN SISTEMAS INFORMÁTICOS**

El Tribunal de Trabajo de Integración Curricular certifica que: El Trabajo de Integración Curricular; tipo: Proyecto Técnico. “**DESARROLLO DE UNA PLATAFORMA PARA LA EJECUCIÓN DE PROCESOS DE ADECUACIÓN DE LA INFORMACIÓN DE FORMA AUTOMÁTICA PARA LA EMPRESA TRACKLINK UTILIZANDO WORKER SERVICES.**”, realizado por los señores **MARIO DAVID CHÁVEZ JARAMILLO** e **ISRAEL EDUARDO OÑA ORDÓÑEZ**, ha sido minuciosamente revisado por los Miembros del Tribunal del Trabajo de Integración Curricular, el mismo que cumple con los requisitos científicos, técnicos, legales, en tal virtud el Tribunal autoriza su presentación.

	FIRMA	FECHA
Ing. Ivonne Elizabeth Rodríguez Flores <b>PRESIDENTE DEL TRIBUNAL</b>	_____	29-10-2021
Ing. Jorge Ariel Menéndez Verdecia <b>DIRECTOR DEL TRABAJO DE INTEGRACIÓN CURRICULAR</b>	_____	29-10-2021
Dr. Alonso Washington Álvarez Olivo <b>MIEMBRO DEL TRIBUNAL</b>	_____	29-10-2021

## **DEDICATORIA**

A mi madre quién a lo largo de mi vida ha sido mi pilar, solo me queda decir, lo logramos.

**Mario**

Este trabajo se lo dedico a mis padres quienes siempre me han apoyado para que pueda cumplir cada una de mis metas.

**Israel**

## **AGRADECIMIENTO**

Agradecemos a los buenos docentes de la Facultad de Informática y Electrónica de la Escuela Superior Politécnica de Chimborazo que demostraron una gran entrega en su trabajo e hicieron de este proceso algo valioso para la vida, en especial al Ingeniero Jorge Menéndez por su apoyo en la realización de este trabajo.

A la empresa Tracklink y al Ingeniero Edgar Toapanta por su apertura para la implementación de este proyecto.

**Mario e Israel**

## TABLA DE CONTENIDO

ÍNDICE DE TABLAS.....	ix
ÍNDICE DE FIGURAS.....	x
ÍNDICE DE GRÁFICOS.....	xi
ÍNDICE DE ANEXOS.....	xii
RESUMEN.....	xiii
SUMMARY.....	xiv
INTRODUCCIÓN.....	1

### CAPÍTULO I

1. DIAGNÓSTICO DEL PROBLEMA.....	2
1.1. Antecedentes.....	2
1.2. Formulación del problema.....	3
1.3. Sistematización del problema.....	3
1.4. Justificación.....	3
1.4.1. <i>Justificación teórica</i> .....	3
1.4.2. <i>Justificación aplicativa</i> .....	4
1.5. Objetivos.....	5
1.5.1. <i>Objetivo general</i> .....	5
1.5.2. <i>Objetivos específicos</i> .....	6

### CAPÍTULO II

2. FUNDAMENTOS TEÓRICOS.....	7
2.1. Procedimientos de ejecución de los procesos de adecuación de la información....	7
2.2. Automatización de procesos informáticos.....	9
2.2.1. <i>Ventajas de la automatización de procesos informáticos</i> .....	10
2.2.2. <i>Impactos de la automatización de procesos en los sistemas de información</i> .....	10
2.3. Daemons.....	11
2.3.1. <i>Características de los daemons</i> .....	11
2.3.2. <i>Tipos de daemons por sistema operativo</i> .....	12
2.3.3. <i>Evolución de los Windows services</i> .....	12
2.3.3.1. <i>Windows services</i> .....	12
2.3.3.2. <i>Worker services</i> .....	12

2.3.4.	<i>Automatización de procesos con daemons</i> .....	13
2.4.	<b>Aplicación web</b> .....	13
2.4.1.	<i>Tipos de aplicaciones por su arquitectura</i> .....	14
2.5.	<b>Calidad</b> .....	16
2.5.1.	<i>Normas de la calidad del software</i> .....	17
2.5.2.	<i>Norma ISO/IEC 25010</i> .....	17
2.5.2.1.	<i>Eficiencia de desempeño</i> .....	18
2.6.	<b>Trabajos Relacionados</b> .....	19

### CAPÍTULO III

3.	<b>MARCO METODOLÓGICO</b> .....	21
3.1.	<b>Métodos y tipo de la investigación</b> .....	21
3.2.	<b>Plataforma Sentinel</b> .....	22
3.2.1.	<i>Fase inicial</i> .....	22
3.2.1.1.	<i>Requerimientos</i> .....	23
3.2.2.	<i>Fase de planificación</i> .....	24
3.2.2.1.	<i>Miembros y roles del proyecto</i> .....	24
3.2.2.2.	<i>Estimación</i> .....	25
3.2.2.3.	<i>Plan de entrega</i> .....	26
3.2.3.	<i>Fase de desarrollo</i> .....	28
3.2.3.1.	<i>Interacción del usuario con la plataforma</i> .....	28
3.2.3.2.	<i>Arquitectura de la plataforma</i> .....	30
3.2.3.3.	<i>Diseño de la base de datos</i> .....	31
3.2.3.4.	<i>Diseño de la interfaz de usuario</i> .....	32
3.2.3.5.	<i>Estándar de codificación</i> .....	33
3.2.3.6.	<i>Desarrollo</i> .....	34
3.2.4.	<i>Fase de cierre</i> .....	36
3.2.5.	<i>Cumplimiento</i> .....	37
3.3.	<b>Tiempo de retraso del proceso actual de adecuación de la información</b> .....	38
3.4.	<b>Tiempo de retraso del proceso de adecuación de la información con Sentinel</b> ....	39

### CAPÍTULO IV

4.	<b>RESULTADOS</b> .....	42
4.1.	<b>Análisis del tiempo de retraso del proceso actual de adecuación de la información</b> .....	42

4.1.1.	<i>Tabulación de las encuestas</i> .....	42
4.1.2.	<i>Cantidad de scripts ejecutados por semana</i> .....	42
4.1.3.	<i>Cantidad de procedimientos almacenados ejecutados por semana</i> .....	43
4.1.4.	<i>Tiempo de preparación de scripts por semana</i> .....	44
4.1.5.	<i>Tiempo de preparación de procedimientos almacenados por semana</i> .....	45
4.1.6.	<i>Tiempo de acceso a los ambientes de ejecución por semana</i> .....	46
4.2.	<b>Análisis del tiempo de retraso del proceso de adecuación de la información con Sentinel</b> .....	48
4.2.1.	<i>Tabulación de los datos observados</i> .....	48
4.3.	<b>Análisis</b> .....	49
<b>CONCLUSIONES</b> .....		52
<b>RECOMENDACIONES</b> .....		53
<b>GLOSARIO</b>		
<b>BIBLIOGRAFÍA</b>		
<b>ANEXOS</b>		

## ÍNDICE DE TABLAS

<b>Tabla 1-2:</b> Subcaracterísticas de la eficiencia de desempeño .....	18
<b>Tabla 1-3:</b> Datos de los miembros y roles del proyecto.....	24
<b>Tabla 2-3:</b> Sentinel - Equivalencia de puntos estimados a tallas de camisetas. ....	25
<b>Tabla 3-3:</b> Sentinel - Product Backlog .....	25
<b>Tabla 4-3:</b> Sentinel - Sprint Backlog.....	27
<b>Tabla 5-3:</b> Sentinel - LOCs .....	35
<b>Tabla 6-3:</b> Preguntas de la encuesta .....	38
<b>Tabla 7-3:</b> Estratificación de la muestra de procesos ejecutados con Sentinel .....	40
<b>Tabla 8-3:</b> Procesos ejecutados por cada entorno.....	41
<b>Tabla 1-4:</b> Cantidad de scripts ejecutados por semana .....	43
<b>Tabla 2-4:</b> Cantidad de procedimientos almacenados ejecutados por semana .....	44
<b>Tabla 3-4:</b> Tiempo de preparación de scripts por semana .....	45
<b>Tabla 4-4:</b> Tiempo de preparación de procedimientos almacenados por semana .....	46
<b>Tabla 5-4:</b> Tiempo de acceso a los ambientes de ejecución por semana.....	47
<b>Tabla 6-4:</b> Tiempos de retraso en la ejecución por día con Sentinel.....	48
<b>Tabla 7-4:</b> Resumen de tiempos con Sentinel.....	49
<b>Tabla 8-4:</b> Tiempo medio en los dos entornos.....	49
<b>Tabla 9-4:</b> Pruebas de normalidad de la muestra.....	49
<b>Tabla 10-4:</b> Resultados del test Z.....	50

## ÍNDICE DE FIGURAS

<b>Figura 1-2:</b> Diagrama de procesos de adecuación de la información .....	8
<b>Figura 2-2:</b> Diagrama de arquitectura web en 2 capas .....	14
<b>Figura 3-2:</b> Diagrama de arquitectura de 2 niveles y 3 capas .....	15
<b>Figura 4-2:</b> Diagrama de la arquitectura hexagonal .....	15
<b>Figura 5-2:</b> Diagrama de separación de capas en la arquitectura limpia.....	16
<b>Figura 1-3:</b> Sentinel - Diagrama de casos de uso .....	29
<b>Figura 2-3:</b> Sentinel - Diagrama de despliegue de la plataforma.....	30
<b>Figura 3-3:</b> Sentinel - Modelo físico de la base de datos.....	31
<b>Figura 4-3:</b> Sentinel - Pantalla de inicio .....	32
<b>Figura 5-3:</b> Sentinel - Registro de proceso tipo SP .....	33

## ÍNDICE DE GRÁFICOS

<b>Gráfico 1-3:</b> Sentinel - Burndown chart .....	36
<b>Gráfico 2-3:</b> Sentinel - Pruebas unitarias exitosas vs fallidas .....	37
<b>Gráfico 1-4:</b> Cantidad de scripts ejecutados por semana .....	43
<b>Gráfico 2-4:</b> Cantidad de procedimientos almacenados ejecutados por semana .....	44
<b>Gráfico 3-4:</b> Tiempo de preparación de scripts por semana .....	45
<b>Gráfico 4-4:</b> Tiempo de preparación de procedimientos almacenados por semana .....	46
<b>Gráfico 5-4:</b> Tiempo de acceso a los ambientes de ejecución por semana .....	47
<b>Gráfico 6-4:</b> Representación de la distribución normal .....	51
<b>Gráfico 7-4:</b> Comparación de medias de tiempo de retraso en la ejecución .....	51

## **ÍNDICE DE ANEXOS**

**ANEXO A:** ENTREVISTA A LA DIRECTORA DE TI DE LA EMPRESA TRACKLINK

**ANEXO B:** ENCUESTA MODELO A LOS RESPONSABLES DE LA EJECUCIÓN DE LOS PROCESOS DE ADECUACIÓN DE LA INFORMACIÓN DE LA EMPRESA TRACKLINK

**ANEXO C:** RESULTADOS DE LAS ENCUESTAS A LOS RESPONSABLES DE LA EJECUCIÓN DE LOS PROCESOS DE ADECUACIÓN DE LA INFORMACIÓN DE LA EMPRESA TRACKLINK

**ANEXO D:** MEDICIÓN DE TIEMPOS DE RETRASO EN LA EJECUCIÓN DE LOS PROCESOS DE ADECUACIÓN DE LA INFORMACIÓN DE LA EMPRESA TRACKLINK EN LA PLATAFORMA SENTINEL

**ANEXO E:** CERTIFICADO DE ENTREGA DE MANUALES Y DE LA PLATAFORMA SENTINEL

## RESUMEN

En el presente proyecto se tuvo como objetivo modificar el tiempo de retraso en la ejecución de los procesos de adecuación de la información de la Empresa Tracklink por medio de una plataforma denominada “Sentinel”. Para lo cual se obtuvo de manera detallada el procedimiento que se seguía inicialmente mediante una entrevista a la directora del área de TI de Tracklink, mientras que los tiempos de las actividades involucradas se obtuvieron por medio de una encuesta dirigida a todo el personal responsable de la ejecución. Haciendo uso de la metodología SCRUM, se desarrollaron bajo una arquitectura hexagonal con el framework .NET los dos componentes principales de la plataforma, una aplicación web y un worker service. Una vez concluida la fase de desarrollo se procedió a recopilar los registros de ejecución de la plataforma para realizar una comparación estadística del tiempo de retraso de la ejecución de los procesos entre el procedimiento original respecto al uso de “Sentinel”. Al tabular los datos se obtuvo que el tiempo promedio del retraso evaluado en el procedimiento original es de 87 segundos con 715 milisegundos, mientras que con “Sentinel” este tiempo es de 2 segundos con 57 milisegundos, dando como resultado una diferencia significativa de 85 segundos con 658 milisegundos por proceso. Por tal motivo se concluye que fue posible reducir en un 97.65% aproximadamente el tiempo de retraso en la ejecución de los procesos de adecuación de la información de la empresa Tracklink con la implementación de la plataforma. Se recomienda cambiar el uso de worker services por Windows services para corroborar la reducción del tiempo de retraso en la ejecución de procesos con daemons especialmente en entornos Windows.

**Palabras clave:** <TECNOLOGÍA Y CIENCIAS DE LA INGENIERÍA>  
<AUTOMATIZACIÓN> <PROCESOS INFORMÁTICOS> <METODOLOGÍA SCRUM>  
<WORKER SERVICES>.



Firmado electrónicamente por:  
**HOLGER GERMAN  
RAMOS UVIDIA**

1825-DBRA-UPT-2021

2021-09-23

## SUMMARY

The objective of current project was to modify the delay time in the execution of the processes of adaptation of the information of the Tracklink Company through a platform called “Sentinel”. For which, we had an interview with the director of the IT area of Tracklink that allowed us to obtain in detail the procedure that was initially followed, while the times of the activities involved were obtained through a survey addressed to all the staff responsible for the execution. We developed, under a hexagonal architecture with the .NET framework, the two main components of the platform, a web application and a worker service using the SCRUM methodology. Once the development phase was concluded, we collected the platform execution logs to make a statistical comparison of the delay time of the execution of the processes between the original procedure compared to the use of “Sentinel”. When tabulating the data, we obtained that the average time of the delay evaluated in the original procedure is 87 seconds and 715 milliseconds, while with “Sentinel” this time is 2 seconds and 57 milliseconds, resulting in a significant difference of 85 seconds. with 658 milliseconds per process. Thus, we concluded that it was possible to reduce 97.65% approximately the delay time in the execution of the processes of adaptation of the information of the Tracklink company with the implementation of the platform. We recommend changing the use of worker services to Windows services to corroborate the reduction of the delay time in the execution of processes with daemons, especially in Windows environments.

**Key words:** <TECHNOLOGY AND ENGINEERING SCIENCE> <AUTOMATION>  
<INFORMATION TECHNOLOGY PROCESSES> <SCRUM METHODOLOGY>  
<WORKER SERVICES>.

## INTRODUCCIÓN

Con el constante avance de la tecnología es común que las tareas que se realizan de forma recurrente, siguiendo una secuencia de pasos, se intenten automatizar para optimizar recursos, principalmente el humano, logrando así una mejora de la productividad y posiblemente también una reducción de costos y/o tiempos en la realización de dichas tareas.

En este estudio se plantea desarrollar una plataforma que permita automatizar un procedimiento informático, a la que se ha denominado “Sentinel”, que hasta el momento se lo realiza de forma manual varias veces por semana involucrando una gran cantidad de horas hombre, y dada la naturaleza de este al estar compuesto por una serie de actividades previamente definidas, es posible pensar en que es viable su automatización.

En las empresas comúnmente se realizan procesos como el envío de boletas de pagos a fin de mes a los empleados, la generación de informes, el procesamiento de documentos recibidos, entre otros que son necesarios para el correcto funcionamiento de la institución, todos estos generalmente tienen una característica en común, se repiten de forma constante en el tiempo y existe al menos una persona encargada de realizarlo; es el caso de la empresa Tracklink, que debido a su tamaño y actividad, se ve en la necesidad de realizar una gran cantidad de procesos con los datos que genera diariamente en sus diferentes áreas para garantizar la continuidad de sus actividades. Por tal motivo se plantea el desarrollo de una aplicación web que permita calendarizar dichos procesos y un worker service que ejecute estos en el momento esperado.

El uso de herramientas tecnológicas como bases de datos, entornos de programación, navegadores web, entre otros son indispensables para la construcción de la plataforma propuesta, así como también el uso del lenguaje de programación C# con el framework .NET para agilizar el proceso de desarrollo de los componentes.

El presente trabajo está compuesto de cuatro capítulos: el primero contiene antecedentes, justificaciones y objetivos; el segundo, muestra los detalles teóricos y conceptos de las tecnologías que se utilizaron para el desarrollo del proyecto, el tercero muestra los detalles del diseño y construcción de la plataforma y el cuarto contiene el análisis de los resultados obtenidos con esta frente al procedimiento original.

# CAPÍTULO I

## 1. DIAGNÓSTICO DEL PROBLEMA

### 1.1. Antecedentes

Actualmente Tracklink es una empresa privada dedicada principalmente a proveer servicios de rastreo satelital de todo tipo de vehículos en varios países de Latinoamérica, por esta razón cuenta con cientos de miles de clientes, respecto a los cuales debe procesar información puntual mes a mes, para de esta manera asegurar tanto la prestación de un servicio de calidad a los mismos, como la correcta dirección de la empresa.

Hoy en día, empresas de todo el mundo generan enormes cantidades de datos que deben ser procesados y analizados con el objetivo de realizar una oportuna toma de decisiones, es por eso por lo que invierten en soluciones tecnológicas que faciliten estos procesos repetitivos, puesto que la demora en la disponibilidad, o peor aún, la falta de esta información puede provocar cuantiosas pérdidas económicas.

Día a día es necesario mantener el control de los datos que recibe y genera una empresa, dado que la información es muy valiosa siendo considerada un activo intangible, es por esto por lo que se apoyan en recursos de hardware y software, supervisado por el personal designado (Cazañas Gordón y Parra Mora, 2017).

En base a lo anterior es posible mencionar que en la empresa Tracklink, cada día se generan datos de más de cien mil usuarios, los mismos que cada cierto período de tiempo deben ser procesados para generar nueva información resumida y concisa para poner en conocimiento de las áreas gerenciales el estado de la empresa en dicho período de tiempo; estos procesos al momento consisten en adecuar para cada mes, en base a parámetros como fechas o localizaciones, la gran cantidad de datos de las diferentes fuentes de información con las que cuenta la empresa.

Una vez presentado este contexto se puede aclarar que dichos procesos de adecuación de la información se realizan de forma manual y repetitiva por un departamento de TI encargado, quienes, por diferentes causas, pueden generar información errónea o, en el peor de los casos, pasar por alto uno de los procesos y no proveer el reporte respectivo, es por ello que se plantea el desarrollo de una plataforma, utilizando worker services, denominada Sentinel, para que se puedan agendar y posteriormente ejecutar de forma automática estos procesos de adecuación de la información para así reducir los tiempos de retraso en la ejecución de estos.

## **1.2. Formulación del problema**

¿Es posible modificar los tiempos de retraso en la ejecución de los procesos de adecuación de la información que emplea la empresa Tracklink mediante la plataforma Sentinel?

## **1.3. Sistematización del problema**

- ¿Cómo son los procedimientos de ejecución de los procesos de adecuación de la información y sus tiempos de retraso?
- ¿Cómo trabajan los worker services para ejecutar de forma automática los procesos?
- ¿Es posible desarrollar la plataforma Sentinel utilizando worker services?
- ¿Cuáles son los tiempos de retraso en la ejecución de los procesos de adecuación de la información a través de la plataforma Sentinel?
- ¿De qué manera se puede evaluar el nivel de eficiencia de la plataforma propuesta?

## **1.4. Justificación**

### ***1.4.1. Justificación teórica***

Dentro del mundo de la informática existen diferentes tipos de procesos que se pueden clasificar de acuerdo varios factores, uno de estos es el plano en el que son ejecutados, dividiéndose en dos tipos, procesos en primer plano y procesos en segundo plano (McKelvey, 2018).

Según (Mayank y Garg, 2019) los procesos que se ejecutan en el primer plano son aquellos que permiten la interacción con el usuario, es decir que están a disposición de las acciones que este realice para completar el proceso, mientras que los segundos se conciben para tareas muy específicas y por tal razón no requieren de la intervención de un usuario para completarse, lo que en este trabajo de titulación se traduce en una eliminación de tiempos de espera para su ejecución.

Estos últimos se conocen también como “daemons” o demonios en español, dicho nombre fue acuñado en los sistemas UNIX, posteriormente cuando fueron adoptados en los ambientes Windows, aquí tomaron el nombre de “servicios de Windows” (Nowacki, 2021).

Con la tendencia en auge de los sistemas cross-platform o multiplataforma, Microsoft innovó sobre sus denominados servicios de Windows, que únicamente son compatibles con el sistema operativo homónimo, dando como resultado un nuevo tipo de programa multiplataforma de la familia de los demonios llamado ahora worker service (Condron, 2019).

Un worker service es un tipo de programa configurado especialmente para ejecutarse de forma persistente en un computador sin afectar al usuario en caso de que este se encontrara usando el mismo; estos se caracterizan por dos razones principales, primero por no poseer una interfaz de usuario y segundo por no hacer uso de entradas o salidas estándar de datos para iniciar su proceso o para comunicar estados, sino más bien usan archivos de registro especializados para registrar los detalles de su funcionamiento (Condron, 2019).

La aplicación de los demonios toma cada vez más relevancia en proyectos que requieran una capa de monitoreo constante, como en el caso del proyecto de adaptación de un módulo para garantizar la alta disponibilidad de un sistema IDS/IPS, donde cuyo objetivo se cumplió gracias al uso de demonios programados específicamente para verificar la disponibilidad de los nodos requeridos del sistema, si uno de estos falla, el demonio levanta automáticamente un nodo secundario que actúa como un respaldo del principal, consiguiendo así un servicio disponible las 24 horas del día durante los 365 días del año (Herrera, et al., 2012).

Otro caso de aplicación de esta tecnología es en el Big Data, puntualmente refiriéndonos al entorno de Hadoop que según (Vasuja, et al., 2018) en su publicación denominada “Daemons of Hadoop: An Overview”, explican los 5 demonios principales que Hadoop incorpora para administrar el gran volumen de datos que este maneja para realizar procesos de mapeo y seguimiento en el menor tiempo posible dado que cada uno de estos se ejecuta de manera independiente en su propio entorno aislado en una máquina virtual de java.

Por todo lo anterior se presume que se podrá garantizar la alta disponibilidad de la información en el tiempo que esta sea requerida con la aplicación de demonios para la automatización y ejecución de procesos.

#### ***1.4.2. Justificación aplicativa***

La incorporación de la tecnología a procesos manuales permite automatizar estos, liberando el tiempo que conlleva su ejecución (Dumas, et al., 2018).

La característica anterior es tomada a consideración en el presente proyecto, razón por la cual la plataforma presenta una solución innovadora para la mejora de los procesos de adecuación de la información, tal que el personal del área de TI de Tracklink no sienta una responsabilidad adicional, por el contrario, puedan deslindarse del proceso de ejecución y centrarse en el análisis de los resultados, mismos que ayudan a la pronta entrega de informes a las áreas gerenciales, para

esto, el trabajo de titulación propuesto permitirá agendar y posteriormente ejecutar de forma automática los procesos mencionados.

La plataforma estará compuesta por 4 módulos principales, los mismos que se detallan a continuación:

- Módulo de autenticación. – Permite el ingreso de los usuarios autorizados a la aplicación web de la plataforma Sentinel.
- Módulo de calendarización. – Permite gestionar el registro y la calendarización de los procesos de adecuación de la información, siendo este el módulo central de la plataforma puesto que es el punto de partida para solucionar una de las causas para la falta de información oportuna, seguido de la ejecución de los procesos, realizado por el siguiente módulo.
- Módulo de ejecución de los procesos. – Permite la ejecución de forma automática de los procesos de adecuación de la información previamente calendarizados.
- Módulo de panel de control. – Permite visualizar el registro detallado de la ejecución de los procesos de adecuación de la información.

Al ser este un proyecto técnico cuyo enfoque es el desarrollo de una plataforma, según las áreas de conocimiento planteadas por la UNESCO, el tema se encuentra en la sub área de Informática perteneciente al área de Ciencias; dentro del Plan Nacional de Desarrollo se enmarca dentro del objetivo número 5 que busca impulsar la productividad y competitividad para el crecimiento económico y sostenible, de manera redistribuida y solidaria; en cuanto a las líneas de investigación de la ESPOCH, se encuentra dentro de las líneas transversales, específicamente en la de Tecnologías de la Información y Comunicación en el programa de Ingeniería de Software, por último, dentro de las líneas de investigación de la Escuela de Ingeniería en Sistemas se encuentra en la de Gestión de los Sistemas de Información, cuyo objetivo es el mismo que el número 5 del Plan Nacional de Desarrollo descrito anteriormente.

## **1.5. Objetivos**

### ***1.5.1. Objetivo general***

Desarrollar una plataforma compuesta por una aplicación web y un worker service que permita ejecutar los procesos de adecuación de la información de forma automática para modificar los tiempos de retraso en la ejecución de estos en la empresa Tracklink.

### ***1.5.2. Objetivos específicos***

- Determinar los procedimientos de ejecución de los procesos de adecuación de la información y sus tiempos de retraso.
- Definir el modo de trabajo de los worker services para la ejecución automática de procesos.
- Desarrollar los módulos especificados que componen la plataforma Sentinel.
- Analizar los tiempos de retraso en la ejecución de los procesos de adecuación de la información.
- Evaluar el nivel de eficiencia de la plataforma propuesta en base a la norma ISO/IEC 25010

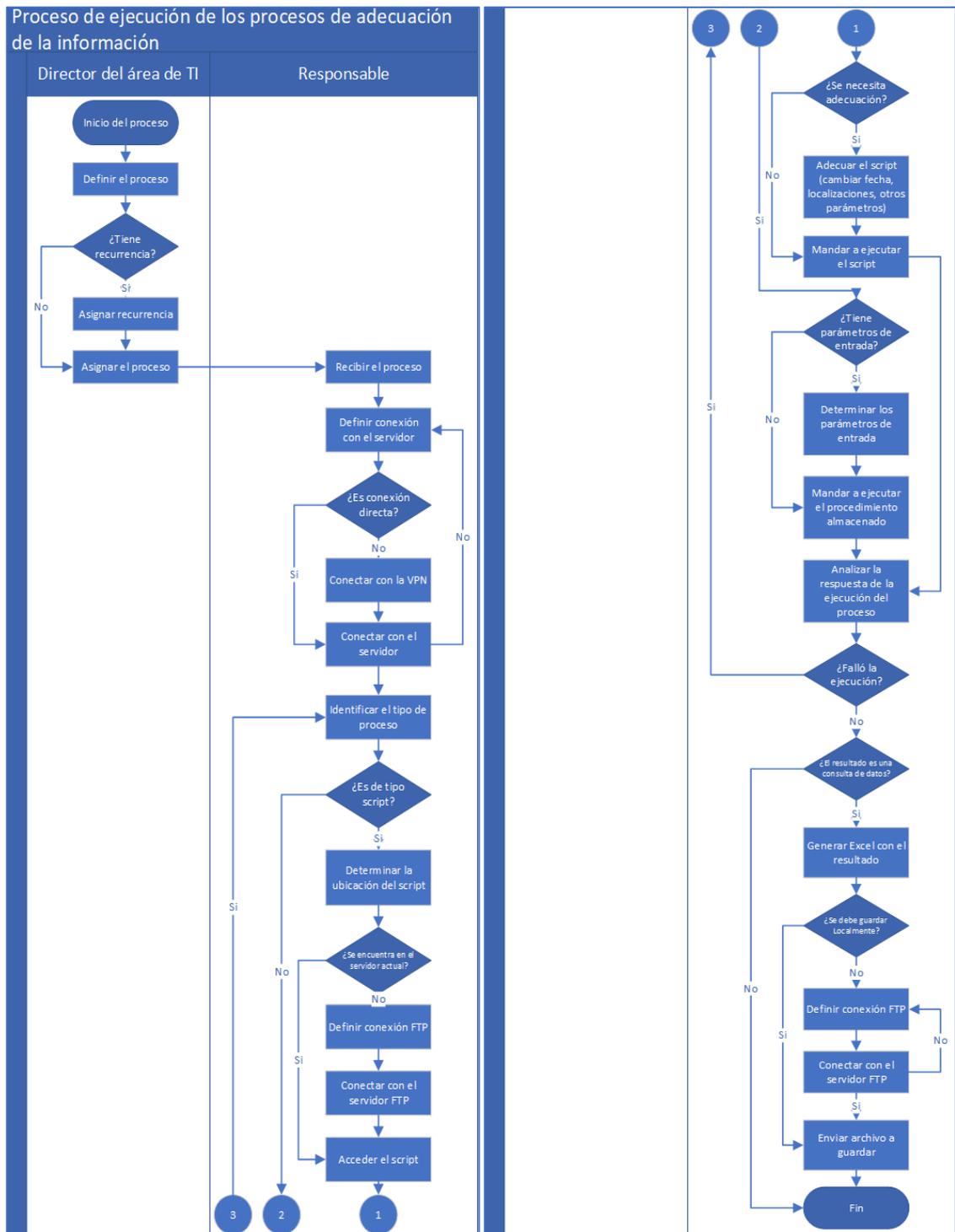
## CAPÍTULO II

### 2. FUNDAMENTOS TEÓRICOS

#### 2.1. Procedimientos de ejecución de los procesos de adecuación de la información

Según (RAE, 2020) “un procedimiento consiste en seguir ciertos pasos predefinidos para desarrollar una labor de manera eficaz”, tomando como referencia la definición anterior se puede mencionar que dentro del área de TI de la empresa Tracklink se cuenta con un protocolo para la ejecución de los procesos de adecuación de la información que asegura la correcta ejecución de estos y además sirve como base para el posterior desarrollo de la plataforma.

El protocolo anteriormente mencionado, fue facilitado por la directora del área de TI de Tracklink, quien, por medio de una entrevista (**Anexo A**), supo manifestar cada uno de los pasos a seguir, desde la identificación del proceso hasta la evaluación de los resultados de este, dichos pasos se presentan en el siguiente diagrama de procesos que se puede observar en la **Figura 1-2**.



**Figura 1-2:** Diagrama de procesos de adecuación de la información

Realizado por: Chávez, M.; Oña, I. 2021

Dentro del diagrama de procesos de la **Figura 1-2** se pueden identificar los siguientes 2 actores con sus funciones:

1. Director del área de TI: Encargado de la asignación de la recurrencia y la designación del responsable de la ejecución del proceso.

2. Responsable: Encargado de la preparación, ejecución y análisis de los resultados de la ejecución de los procesos.

De forma general, los procesos se ejecutan siempre sobre un sistema gestor de base de datos SQL Server y dichos procesos pueden ser únicamente de dos tipos:

1. Script
2. Procedimiento almacenado

Para la ejecución de los procesos se consideran también los tiempos de retraso, mismos que, dentro del contexto de estos procesos de la empresa Tracklink, responden al tiempo que pasa desde la hora planificada para que se empiece a ejecutar el proceso de adecuación de la información, hasta que realmente se empieza a ejecutar, estos tiempos generalmente se distribuyen entre las actividades anteriores que debe realizar el responsable de la ejecución, como:

- Establecer conexión con el servidor de base de datos
- Acceder al gestor de base de datos
- Identificar el proceso a ejecutarse
- Preparar el proceso a ejecutarse

Dentro del contexto de los tiempos de retraso especificados en el párrafo anterior, es necesario aclarar que para la empresa es importante poder reducirlos, debido a que mientras menor sea el tiempo que conlleva la ejecución de estas actividades, el proceso podrá empezar a ejecutarse antes y así el resultado de este podrá entregarse según lo planificado.

## **2.2. Automatización de procesos informáticos**

Un proceso corresponde a la representación de un conjunto de acciones (actividades) que se hacen, bajo ciertas condiciones (reglas) y que puede gatillar o ejecutar cosas (eventos) (Dumas, et al., 2018).

Dentro de este ámbito también se habla de otros temas como, como mejora de procesos, que incluye la reducción de tiempo, número de actividades a realizar, coste, carga de trabajo, etc., para mantener o mejorar la calidad del proceso durante su ejecución hasta su finalización (Undey, 2021).

Según (Schepler y Weber, 2020) la automatización de un proceso informático consiste en el uso de software para crear instrucciones y procesos repetibles para reemplazar o reducir la interacción hombre-máquina con los sistemas de TI. El proceso automatizado puede ejecutarse dentro del alcance de estos marcos, herramientas e instrucciones sin intervención manual para completar las tareas.

Para la mejora de los procesos, existe un beneficio principal que se abordó a lo largo del presente trabajo, este es el tiempo que toma ejecutar las actividades necesarias para llevar a cabo un proceso.

### ***2.2.1. Ventajas de la automatización de procesos informáticos***

Existen varias ventajas en la automatización de procesos, por ejemplo, se pueden disminuir los errores en los mismos, liberar recursos y sobre todo disminuir los tiempos que, según (Undey, 2021) los tiempos de ciclo de los procesos están relacionados con la eficiencia y la agilidad de negocio que tiene una organización, por lo que la reducción de estos mejorará el proceso. Sin embargo, esta reducción no debería reducir la calidad del proceso, ni tampoco debería aumentar su complejidad.

Por lo tanto, se deben considerar diferentes tiempos como el tiempo de proceso de inicio a fin, tiempo de transferencia, tiempo de ejecución por actividad, tiempo de ruta crítica, tiempo por ruta alternativa y tiempo de espera hasta iniciar una actividad, por lo que, analizando estos tiempos, se pueden realizar mejoras en el proceso (Schepler y Weber, 2020).

### ***2.2.2. Impactos de la automatización de procesos en los sistemas de información***

Con la globalización el mundo laboral ha evolucionado generando cambios notables a la hora de ejecutar procesos de forma más eficiente para reducir costos y tiempos, manteniendo la competencia dentro del mercado. La automatización de información principalmente pretende reducir costes en las actividades del negocio, cambiando la naturaleza en la que se realiza el trabajo abriendo oportunidades hacia la innovación, transformación digital, tecnologías de la información, etc. (Nolan, et al., 2021)

Como se menciona en el trabajo de investigación sobre el impacto de la automatización en el mejoramiento de procesos (Cazañas Gordón y Parra Mora, 2017), se concluye que el proceso sin automatizar no es eficiente. Los estándares de la industria señalan que un proceso de negocio es eficiente cuando 25% o más del tiempo total del proceso se dedica a la ejecución de las tareas que

contribuyen a satisfacer los requerimientos del cliente. La eficiencia del tiempo de ciclo del proceso sin automatizar es del 8%.

La automatización de tareas mejora el desempeño de los procesos al minimizar el tiempo invertido en tareas de control y mantenimiento de registros. De las acciones de mejora que se establecieron para minimizar los efectos de las causas de ineficiencia en el proceso actual 50% de las acciones de mejora corresponden a automatización de tareas, el resto de las acciones de mejora se distribuyen en: eliminación de tareas sin valor agregado (31%), cambios de políticas (11%), combinar tareas (1%) y eliminación de causas de reprocesamiento (1%) (Cazañas Gordón y Parra Mora, 2017).

Se ha demostrado que la automatización tiene un efecto positivo en cuanto a la reducción de tiempos como se menciona en los trabajos anteriormente descritos, además de la disponibilidad de datos a tiempo.

### **2.3. Daemons**

Según (McKelvey, 2018) un demonio es un tipo de programa en sistemas operativos Unix que se ejecuta en segundo plano, en lugar de estar bajo el control directo de un usuario, esperando ser activado por la ocurrencia de un evento o condición específica.

Un daemon en esencia, es una aplicación que se ejecuta sin la necesidad de la presencia humana siendo esta característica fundamental en el desarrollo del presente trabajo en la ejecución y automatización de procesos recurrentes (Mayank y Garg, 2019).

#### ***2.3.1. Características de los daemons***

Los daemons tienen una característica principal, no hacen uso de una interfaz de usuario para su funcionamiento y por ende sus entradas provienen de otras fuentes al igual que sus salidas se registran en archivos del sistema o en gestores de información externos (Mayank y Garg, 2019).

Los daemons cronológicos adicionalmente poseen la capacidad de realizar tareas programadas, generalmente estas son actividades del mantenimiento del sistema en segundo plano cada cierto tiempo (Condron, 2019).

### ***2.3.2. Tipos de daemons por sistema operativo***

Este tipo de programas se conocen con diferentes nombres según el sistema operativo de destino, por ejemplo:

- Linux: En sus diferentes distribuciones, estos mantienen el nombre de “daemons” pues su origen es en estas plataformas basadas en Unix (McKelvey, 2018).
- Windows: Inicialmente tomaron el nombre de “Windows services” y con la tendencia de aplicaciones multiplataforma, Microsoft lanzó una nueva versión de estos, ahora llamados “worker services” (Nowacki, 2021).
- MacOS: Principalmente se los denominó como “system extensions” aunque comúnmente eran conocidos también como “daemons” (Mayank y Garg, 2019).

### ***2.3.3. Evolución de los Windows services***

Los servicios Windows tienen sus orígenes en los sistemas Microsoft Windows NT según (Nowacki, 2021) estos iniciaron replicando la funcionalidad de los daemons y con las nuevas versiones de Windows se fueron incluyendo nuevas características hasta dar el paso a los worker services.

#### ***2.3.3.1. Windows services***

Fueron incluidos en las primeras versiones de Windows NT con el objetivo de correr procesos asociados al sistema, la red de servicios y otros procesos locales; con las nuevas versiones de Windows se fueron incluyendo nuevas características como la posibilidad de ejecutarlos o detenerlos a petición del usuario, finalmente cualquier aplicación de Windows es capaz de desempeñarse como un daemon y no requerir exclusivamente de un Windows service.

#### ***2.3.3.2. Worker services***

Con la llegada de .NET Core 3.0 se introdujo la plantilla de los Worker Services a la familia de aplicaciones de este entorno, según (Condrón, 2019) los worker services poseen características útiles que convierten una aplicación de consola en algo más poderoso, y que en la actualidad son usados generalmente en los siguientes escenarios:

- Procesamiento de mensajes/eventos provenientes de colas o flujos de eventos.
- Reacción ante cambios en archivos.

- Formateo y limpieza de conjuntos de datos.
- Calendarización de procesos.
- Ejecución de tareas en múltiples subprocesos

Además de lo mencionado anteriormente, una de las características que más diferencian a este tipo de servicios es su capacidad de ejecutarse en diferentes plataformas, dado que el framework sobre el cual se construyen es multiplataforma y puede ser compilado para ambientes Windows, Linux o MacOS (Gordon, 2020).

Según (Giretti, 2020) este tipo de aplicaciones se ajustan bien para entornos donde se necesiten realizar cargas de trabajo periódicas y de larga duración debido a que, por su construcción, incluyen características comunes de ASP.NET Core como la inyección de dependencias, registro de logs y configuración de dependencias haciendo uso de un host para mantener la vida útil de la aplicación hasta que se le indique al host que se apague.

Para el desarrollo del presente trabajo, se tomó en cuenta la definición de (Giretti, 2020) dado que es mandatorio generar un servicio que sea capaz de realizar cargas de trabajo periódicas y también se apega a la característica de procesamiento de mensajes provenientes de flujos de eventos expuestas por (Condron, 2019) debido a que pueden existir procesos que se envían a ejecutar manualmente, pero por medio del worker service.

#### ***2.3.4. Automatización de procesos con daemons***

Actualmente existen daemons en un gran número de herramientas y programas como los navegadores web, estos ejecutan procesos de sincronización de datos en múltiples dispositivos centrándose en el caso de ausencia de conexión a internet, el proceso en segundo plano se ejecuta cuando se recupera la conexión, otra aplicación es la gestión de notificaciones push que son utilizadas principalmente en aplicaciones móviles para mostrar información de relevancia al usuario, los cambios que se realizaron dentro de esta desencadenan la ejecución de estos procesos en segundo plano (Condron, 2019).

### **2.4. Aplicación web**

Una aplicación web es una aplicación que se invoca con un navegador web, generalmente, a través de Internet, pero estas se pueden construir tanto para ambientes públicos como privados. Este tipo de aplicaciones tienen sus orígenes en la década de los 90, donde, según (Yik Ern, 2019) en su tesis explica como la llegada de PHP, uno de los lenguajes precursores de muchas de las plataformas

que ocupamos hoy en día, dio paso al origen de estas, dado que hasta ese entonces se ocupaba nada más que HTML para crear documentos bajo un sistema de hipertexto con el objetivo de compartir documentos, pero no propiamente eran aplicaciones web.

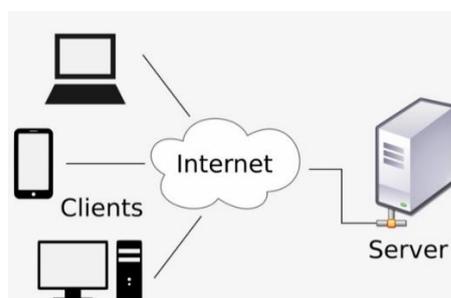
El desarrollo de aplicaciones web ha ido evolucionando con gran velocidad debido al surgimiento de nuevas tecnologías que permiten crear nuevas aplicaciones más rápidas, robustas, seguras, etc. Para (Strauss, 2021) los desarrollos futuros en las aplicaciones web estarán impulsados por los avances en la tecnología de los navegadores, la infraestructura de Internet web, los estándares de protocolo, los métodos de ingeniería de software y las tendencias de las aplicaciones.

Dentro de la construcción en sí de una aplicación web, es posible encontrar muchas similitudes y diferencias, como el lenguaje en que fueron construidas, los frameworks empleados, el propósito de estas, pero algo que generalmente puede tener pocas variaciones, es la arquitectura con la que fueron diseñadas (Yik Ern, 2019).

#### **2.4.1. Tipos de aplicaciones por su arquitectura**

Como lo mencionan (Deinum y Cosmina, 2021) en un inicio las aplicaciones se escribieron bajo una arquitectura que no era arquitectura dado que mezclaban un sin número de responsabilidades dentro de un mismo componente, cuestión que dificultaba tareas de mantenimiento futuras. Con el paso del tiempo se fueron realizando separaciones tanto físicas como lógicas dentro de los proyectos de software, consiguiendo así distintos tipos de arquitecturas como:

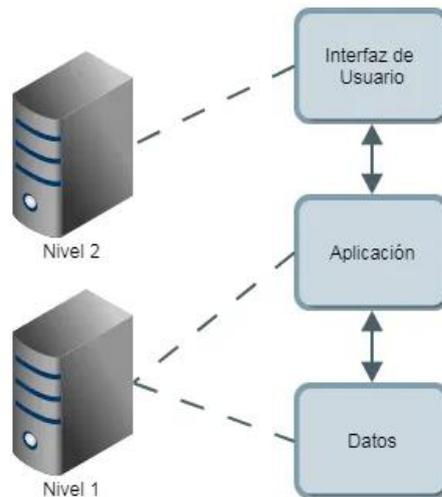
**N capas:** Según (Khenmour y Debbah, 2017) con la introducción de bases de datos a los sistemas, aparecieron las primeras arquitecturas de dos capas, capa de aplicación y capa de datos, para de esta forma poder realizar una separación tanto física como lógica del manejo de datos dentro de una aplicación, también es conocida tradicionalmente como arquitectura cliente servidor, en la **Figura 2-2** se observa una representación de esta arquitectura de 2 capas.



**Figura 2-2:** Diagrama de arquitectura web en 2 capas

**Fuente:** (Khenmour y Debbah, 2017)

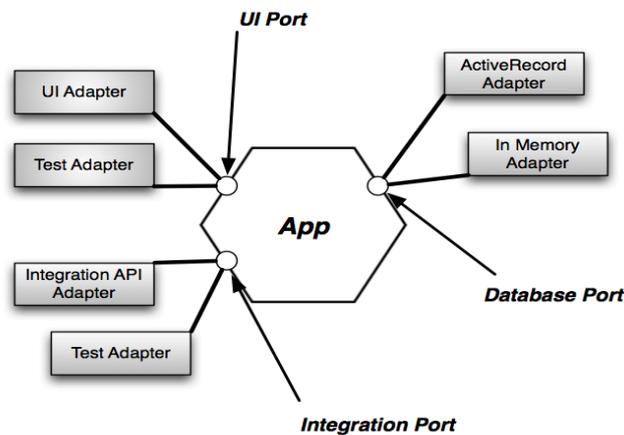
Una aplicación web bajo este esquema, puede estar compuesta por n cantidad de capas, en donde cada una tenga una única responsabilidad sobre el sistema, es importante tener en cuenta que una capa no es lo mismo que un nivel, bajo este concepto, capa es la separación lógica que se le puede dar a un sistema, mientras que nivel es la separación física de esas capas, como se puede observar una separación de 2 niveles y 3 capas en la **Figura 3-2** (Lamza, et al., 2015).



**Figura 3-2:** Diagrama de arquitectura de 2 niveles y 3 capas

Fuente: (Lamza, et al., 2015)

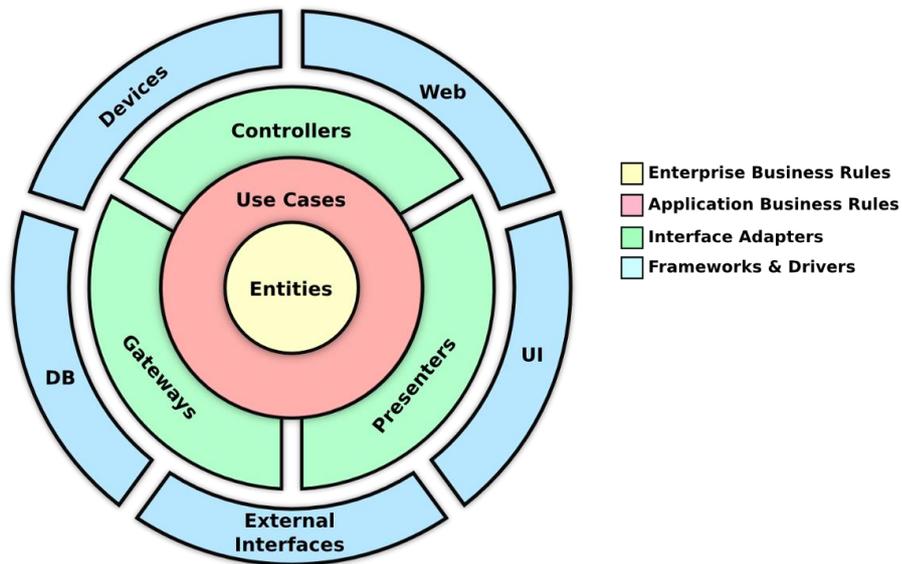
**Hexagonal:** Según (Mcclanahan, 2021) esta arquitectura también es conocida como puertos y adaptadores debido a que cada uno de sus componentes se acopla con el resto según se necesite tomando en cuenta las reglas de integración establecidas inicialmente para los denominados puertos y adaptadores, donde los primeros se definen como un contrato que estos últimos deben cumplir para ser capaces de implementarlos, como se puede observar en la **Figura 4-2** los puertos de la aplicación y los adaptadores que pueden integrarse.



**Figura 4-2:** Diagrama de la arquitectura hexagonal

Fuente: (Mcclanahan, 2021)

Uno de los conceptos claves en este tipo de arquitecturas es el control de las dependencias, en arquitecturas n capas, estas no pueden funcionar si no existe la capa inmediata inferior o superior, pero en el caso de las arquitecturas hexagonales, la dependencia como tal no solo es vertical, sino también puede ser horizontal pero nunca una capa interna va a depender de una capa externa, es aquí donde toma más sentido la premisa de que estas son una interpretación de la clean architecture propuesta por (Martin, 2017), como se puede observar en la **Figura 5-2** la separación de responsabilidades por parte de cada una de las capas y la forma en como estas se integran en la arquitectura limpia.



**Figura 5-2:** Diagrama de separación de capas en la arquitectura limpia

Fuente: (Martin, 2017)

## 2.5. Calidad

El término castellano calidad está definido por la Real Academia Española de la Lengua como: "conjunto de cualidades que constituyen la manera de ser de una persona o cosa" (RAE, 2020).

Al hablar de bienes o servicios de calidad, su significado sigue siendo ambiguo y muchas veces su uso depende de lo que cada uno entiende por calidad.

Las empresas según (Alam y Ramchand, 2021), pueden definir la calidad como “el logro de la satisfacción de los clientes a través del establecimiento adecuado de todos sus requisitos y el cumplimiento de estos con procesos eficientes, que permita así a la organización ser competitiva en la industria y beneficie al cliente con precios razonables.”

En la industria del software se pueden evidenciar necesidades de satisfacción del cliente de productos o servicios de software, de reducción de recursos invertidos en proyectos de software y de la efectiva asignación de recursos humanos.

La calidad del producto software se puede interpretar como el grado en que dicho producto satisface los requisitos de sus usuarios aportando de esta manera un valor (ISO, 2020).

Una característica que define la ISO 25010 es la eficiencia del desempeño misma que ha sido tomada en cuenta para el desarrollo del presente trabajo, razón por la cual se menciona a continuación breves conceptos sobre eficiencia.

Según (RAE, 2020) eficiencia es la “capacidad de disponer de alguien o algo para conseguir el cumplimiento adecuado de una función”.

Partiendo de esta definición general se pueden encontrar distintos tipos de definiciones dependiendo del campo en la que sea aplicada, por ejemplo, en el campo empresarial la eficiencia significa lograr una productividad favorable, es decir alcanzar los mejores resultados con la inversión mínima de recursos. Para (Patel, et al., 2021) al medir la eficiencia se toman en cuenta variables como el tiempo, el capital, esfuerzo, calidad, etc. Una empresa eficiente se resume en aquella que obtiene más por menos, transformándola en una entidad rentable en el negocio.

Ahora bien, en el campo de la ingeniería de software, según (Alam y Ramchand, 2021) la eficiencia se determina por la utilización de recursos del sistema, la configuración del hardware necesaria, entonces se evalúa el desempeño del sistema esperado teniendo en cuenta las necesidades del usuario.

### ***2.5.1. Normas de la calidad del software***

La norma ISO/IEC 25010 que es una familia de normas que tiene por objetivo la creación de un marco de trabajo común para evaluar la calidad del producto software (ISO, 2020).

### ***2.5.2. Norma ISO/IEC 25010***

La calidad es de fundamental en cada una de las fases del desarrollo software, hasta lograr obtener un producto útil y usable que cumpla con las expectativas del usuario.

Para (Peters y Aggrey, 2020) la calidad de software va de la mano a la concordancia con los requisitos funcionales y de rendimiento explícitamente establecidos con los estándares de desarrollo plenamente documentados, formalmente especificados dentro de alguna norma como la ISO, y con las características implícitas que se espera de todo software desarrollado profesionalmente.

Según (Hovorushchenko, 2018) debe existir una concordancia entre los requisitos planteados frente a los obtenidos para así satisfacer las necesidades del usuario, generando así un sentido de calidad que esté fuertemente ligado a la utilización de metodologías y procedimientos estandarizados bajo las normativas definidas en la ISO 25010.

No es posible medir la calidad de un software de forma precisa debido a la naturaleza en sí del mismo, debido a que, según (Haoues, et al., 2016), las certificaciones y el aval que se tiene de la norma ISO, se otorga a los procesos del desarrollo, mas no al software en sí, es por eso por lo que no se puede medir un software como tal, sino más bien a los atributos que lo conforman.

La ISO 25010 permite evaluar la calidad de un software tomando en cuenta los atributos de este, según la característica que se esté evaluando, razón por la cual es importante definir claramente los requisitos del software para así desarrollarlos acorde a lo esperado y finalmente poder evaluarlos según la normativa.

#### 2.5.2.1. Eficiencia de desempeño

Esta característica representa el desempeño relativo a la cantidad de recursos utilizados bajo determinadas condiciones. Dentro de esta se encuentran 3 subcaracterísticas más como se puede observar en la **Tabla 1-2**.

**Tabla 1-2:** Subcaracterísticas de la eficiencia de desempeño

<b>Subcaracterística</b>	<b>Definición</b>
Comportamiento temporal	Grado en el que los tiempos de respuesta y procesamiento y las tasas de rendimiento de un producto o sistema, al realizar sus funciones, cumplen con los requisitos.
Utilización de recursos	Grado en el que las cantidades y tipos de recursos utilizados por un producto o sistema, al realizar sus funciones, cumplen con los requisitos.
Capacidad	Grado en el que los límites máximos de un producto o parámetro del sistema cumplen los requisitos.

Fuente: (ISO, 2020)

Realizado por: Chávez, M.; Oña, I. 2021

Según (Yuniasri, et al., 2020) el comportamiento temporal de un sistema es un punto determinante cuando el objetivo de este es realizar procesos recurrentes que responden a un flujo de eventos, debido a que, si los procesos internos del sistema no son capaces de responder en el tiempo adecuado, todo el flujo podría colapsar con el transcurso del tiempo por un embotellamiento de solicitudes pendientes a ser atendidas.

Para (Mustari, et al., 2020) que un sistema presente variaciones en el rendimiento durante el tiempo es algo normal y aceptable dado que los datos a procesarse por un sistema no siempre van a ser ideales y más aún si el factor determinante es el volumen de estos, pero para mantener un estándar de calidad, se deben considerar los rangos en los cuales es aceptable una variación en los tiempos de procesamiento para llevar a cabo una acción.

En base a las definiciones planteadas en los párrafos inmediatamente anteriores, es posible indicar que para medir el nivel de eficiencia de la plataforma Sentinel se utilizará la característica de la eficiencia de desempeño de la ISO/IEC 25010, tomando en cuenta sus subcaracterísticas de comportamiento temporal para evaluar los tiempos de ejecución de los procesos de adecuación de la información.

## **2.6. Trabajos Relacionados**

En la actualidad con el avance y desarrollo constante de nuevas tecnologías y tendencias los usuarios optan por el uso de diferentes aplicaciones software que faciliten el desarrollo de procesos reduciendo principalmente factores como tiempo y costos, actualmente estos productos software son utilizados por todo tipo de empresas sin importar la actividad que desempeñan; todas manejan algún tipo de proceso, sea este simple o de alto grado de complejidad, siempre implica un tiempo desde su inicio hasta su fin, pero en el desarrollo de este pueden existir errores causados por el recurso humano que afectan el mismo, generando pérdidas de tiempo que generalmente desembocan subsecuentemente en una pérdida de capital, estas acumulaciones de tiempo perdido en plazos determinados, como mes a mes, pueden generar problemas a nivel gerencial afectando las actividades de la empresa (Undey, 2021).

Se han analizado cuatro trabajos con características similares a las que aspiramos resolver con el presente trabajo de titulación, a continuación, se mencionan los mismos.

En la Universidad Politécnica Salesiana de la ciudad de Guayaquil, se desarrolló un proyecto que propuso el desarrollo de un sistema para el control y monitoreo de tesis debido a que este proceso, hasta ese momento, se lo hacía de forma manual, de tal manera que el estudiante debía

comunicarse vía telefónica o email, resultando así en un proceso que conllevaba mucho tiempo, además de que interrumpía labores diarias. El trabajo mencionado hizo énfasis en identificar en qué medida demandaba tiempo el control y seguimiento de las tesis para así determinar las incidencias al llevar a cabo el proceso manual y no automatizado (Almeida y Cárdenas, 2013).

Un siguiente proyecto de titulación analizado habla sobre los beneficios de integrar una aplicación en el proceso de concesión de microcréditos en una cooperativa con el objetivo de reducir el tiempo que conlleva todo este proceso. Más a detalle, se explica sobre como el desarrollo de la aplicación, denominada SIAM, permitió mejorar y agilizar la gestión de microcréditos, reduciendo tiempos de espera en todas las operaciones, automatizando cálculos y registros, obteniendo así también una reducción de errores humanos (Chang, et al., 2014).

Dentro del artículo denominado “Influence of information systems on business performance” se analizan principalmente los beneficios que aportan los sistemas de información a las empresa, generalmente estos beneficios están relacionados al desempeño; uno de estos beneficios es la disponibilidad de la información y la agilidad que proporcionan al recurso humano dentro de la organización para incrementar su productividad, reduciendo así el tiempo necesario que estas personas deben emplear en tareas manuales, repetitivas o poco prácticas que fácilmente podrían ser automatizadas (Lipaj y Davidaviciene, 2013).

Por último, en el anuario número 38 publicado por la Universidad de Carabobo, en su sección denominada “¿Por qué los sistemas de información son esenciales?” se destaca el papel sumamente importante que tienen estos dentro de la sociedad actual, al ser estos una herramienta muy poderosa sobre la que cualquier persona puede apoyarse para resolver problemas habituales que están relacionados a dos aspectos fundamentales como la disponibilidad de la información, y el tiempo requerido para completar una tarea, ambos son problemas que se han podido superar gracias a la automatización de aquellos procesos, que en antaño se realizaban de forma manual, y que ahora gracias a los sistemas de información, ha sido posible retirar esa carga de las personas (Hamidian y Ospino, 2015).

Tomando en cuenta la información presentada se puede corroborar que los daemons influyen en el tiempo de ejecución de procesos, manteniendo una alta disponibilidad en la información además de la automatización en casos repetitivos minimizando adicionalmente errores por parte del factor humano.

## CAPÍTULO III

### 3. MARCO METODOLÓGICO

En este capítulo se especifica la metodología que se utilizó para el desarrollo de la plataforma Sentinel que permitirá ejecutar los procesos de adecuación de la información de la empresa Tracklink de forma automática para reducir los tiempos de retraso en la ejecución de estos, pasando por la recolección de información hasta el desarrollo mismo de la plataforma.

Para conocer los detalles de los procedimientos llevados a cabo por la empresa se aplicaron técnicas de investigación como la entrevista y la encuesta, mientras que para el desarrollo de la plataforma se aplicaron los conocimientos adquiridos previamente junto con la revisión bibliográfica de ciertos componentes primordiales para los análisis posteriores.

#### 3.1. Métodos y tipo de la investigación

- **Determinar los procedimientos de ejecución de los procesos de adecuación de la información y sus tiempos de retraso.**

Como punto inicial se procedió a la recolección información necesaria para conocer los distintos procedimientos llevados a cabo por el área de TI de la empresa Tracklink para que sus procesos de adecuación de la información sean ejecutados, se agendó una única entrevista a la directora de TI el 29 de octubre de 2021.

La entrevista constó de 10 preguntas (**Anexo A**) enfocadas en conocer los procesos de adecuación de la información que se ejecutan dentro de la empresa, para eso, se generó la misma siguiendo la modalidad no estructurada debido a que se buscaba conocer el procedimiento que se seguía dentro de la empresa, razón por la cual el entrevistado dispone de toda la libertad de contestar a las preguntas sin regirse a un dominio específico de respuestas.

Con la información recabada con la entrevista previa por medio del método analítico se procedió a identificar todos los aspectos involucrados dentro del proceso teniendo como interés principal identificar los tiempos generados.

- **Definir el modo de trabajo de los worker services para la ejecución automática de procesos.**

Es necesario comprender los worker services, sus características, propiedades, entornos, etc., para ser implementados en la etapa de desarrollo, para lo cual se hizo revisión bibliográfica, al ser un tipo de producto software creado por Microsoft, poseen documentación propia, razón por la cual, dicha documentación sirvió de base para identificar los modos de trabajo de estos.

Adicionalmente, a partir de varios autores se recabó información donde se observan casos reales donde se utilizaron worker services, tomando principalmente lo expuesto por (Condron, 2019) respecto a la capacidad de estos de responder ante eventos y funcionar bajo un esquema de calendarización.

En base a lo expuesto en el párrafo se consideró también lo expuesto por (Gordon, 2020) y (Giretti, 2020) respecto al funcionamiento de los worker services en ambientes de cargas de trabajo recurrentes.

### **3.2. Plataforma Sentinel**

Para el desarrollo de la plataforma se hizo uso de la metodología ágil SCRUM debido a que tiene como uno de sus principios, incluir al cliente como parte del equipo de trabajo para así generar un ambiente centrado en la comunicación y garantizar así el entendimiento de las necesidades del cliente.

Dentro del desarrollo de la plataforma se pasaron por distintas fases, mismas que se explican a continuación:

#### ***3.2.1. Fase inicial***

En la fase inicial se determina la información necesaria para iniciar el desarrollo de la plataforma Sentinel.

Dentro de la información necesaria se deben determinar los requerimientos, mismos que son tomados a partir de las necesidades expuestas por el cliente y posteriormente son redactadas como un requerimiento de software.

### 3.2.1.1. *Requerimientos*

Con el fin de determinar las funcionalidades que tendrá la plataforma y el alcance de este se procedió con la identificación de los requerimientos funcionales y no funcionales. Se mantuvo una reunión con la directora del departamento de TI y los responsables de la ejecución de procesos pertenecientes a la empresa Tracklink.

Los requisitos fueron documentados utilizando las tarjetas de historias de usuario, especificando el rol que la ejecuta, qué desea ejecutar y para qué, las mismas que constan en el manual técnico (Chávez y Oña, 2020).

#### Requerimientos Funcionales:

- Permitir el ingreso de administrador.
- Ingresar conexiones FTP.
- Ingresar conexiones SQL.
- Editar conexiones FTP.
- Editar conexiones SQL.
- Eliminar conexiones FTP.
- Eliminar conexiones SQL.
- Visualizar las conexiones registradas.
- Ingresar un proceso tipo Procedimiento almacenado.
- Ingresar un proceso tipo Script.
- Editar un proceso tipo Procedimiento almacenado.
- Editar un proceso tipo Script.
- Eliminar un proceso tipo Procedimiento almacenado.
- Eliminar un proceso tipo Script.
- Listar los procesos registrados.
- Calendarizar un proceso diariamente.
- Calendarizar un proceso semanalmente.
- Calendarizar un proceso mensualmente.
- Calendarizar un proceso en cualquier fecha.
- Visualizar los procesos listos para su ejecución.
- Ejecutar un proceso.
- Visualizar el registro de la ejecución de procesos.

## Requerimientos No Funcionales:

- Disponibilidad
- Seguridad

Una vez documentadas fueron validadas y verificadas en otra reunión de trabajo donde estuvieron presentes los involucrados en dichos procesos y fueron priorizadas por ellos.

Como resultado se obtuvieron 22 requerimientos funcionales y 2 no funcionales, todos los requerimientos serán implementados en la plataforma Sentinel.

### 3.2.2. Fase de planificación

Se definieron las personas que se encuentran involucradas en el desarrollo de la plataforma, se estimaron las historias y establecieron sus tiempos de entrega, todo esto se realizó con el fin de cumplir con la metodología propuesta y llevar un desarrollo óptimo.

#### 3.2.2.1. Miembros y roles del proyecto

Siguiendo la metodología SCRUM se definieron los roles de los involucrados en el desarrollo del proyecto como se presenta en la **Tabla 1-3**.

**Tabla 1-3:** Datos de los miembros y roles del proyecto.

Miembro	Rol	Correo	Institución
Ing. Edgar Toapanta	Product Owner	etoapanta@tracklink.ec	Tracklink
Ing. Jorge Menéndez	Scrum Master	jorge.menendez@esPOCH.edu.ec	ESPOCH
Israel Oña	Development Team	israel.ona@esPOCH.edu.ec	ESPOCH
Mario Chávez	Development Team	mario.chavez@esPOCH.edu.ec	ESPOCH

Realizado por: Chávez, M.; Oña, I. 2021

Para el desarrollo del proyecto se cuenta con un equipo de trabajo formado por 4 personas definidas con su correspondiente rol pertenecientes a la empresa Tracklink y a la Escuela Superior Politécnica de Chimborazo, el product owner también realizará el rol de cliente y el rol de cliente lo asumirán las personas responsables de ejecutar los procesos.

### 3.2.2.2. Estimación

La estimación es un proceso necesario que es aplicado al inicio del proyecto inmediatamente después de realizar la toma de requerimientos, con el objetivo de clarificar el tiempo requerido para desarrollar los mismos y por ende completar el desarrollo de un sistema. Este proceso se realizó mediante la aplicación de un método denominado tallas de camiseta (T-shirt), cuyos valores se detallan a continuación en la **Tabla 2-3**

**Tabla 2-3:** Sentinel - Equivalencia de puntos estimados a tallas de camisetas.

Puntos Estimados	Tallas
>40	XL
40	L
20	M
8	S
4	XS

**Realizado por:** Chávez, M.; Oña, I. 2021

Es importante mencionar que, dentro de los puntos equivalentes a cada talla, un punto corresponde a una hora de trabajo, un día de trabajo está compuesto por 8 horas y una semana de trabajo está compuesta por 5 días de trabajo. En el desarrollo se contó con dos personas trabajando 8 horas diarias teniendo como resultado un total de 80 puntos de esfuerzo a la semana.

Para identificar la prioridad de los requerimientos, se definieron 3 prioridades, alta, media y baja, mismas que fueron asignadas a cada historia de usuario después de realizada una negociación con el cliente. En la **Tabla 3-3** se detalla la lista de requerimientos con su respectiva talla y el valor que le corresponde a esta, finalmente se muestra una columna que indica la prioridad que ha sido dada por el cliente.

**Tabla 3-3:** Sentinel - Product Backlog

Requerimiento	Talla	Puntos Estimados	Prioridad
Determinar la arquitectura de la plataforma	M	20	Alta
Determinar el estándar de codificación de la plataforma	M	20	Alta
Determinar la interfaz de usuario de la plataforma	L	40	Alta
Determinar la base de datos de la plataforma	L	40	Alta
Permitir el ingreso de administrador	M	20	Alta
Ingresar conexiones FTP	M	20	Alta
Ingresar conexiones SQL	M	20	Alta
Editar conexiones FTP	M	20	Alta
Editar conexiones SQL	M	20	Alta
Eliminar conexiones FTP	S	8	Alta

Requerimiento	Talla	Puntos Estimados	Prioridad
Eliminar conexiones SQL	S	8	Alta
Visualizar las conexiones registradas	XS	4	Alta
Ingresar un proceso tipo Procedimiento almacenado	M	20	Alta
Ingresar un proceso tipo Script	M	20	Alta
Editar un proceso tipo Procedimiento almacenado	M	20	Media
Editar un proceso tipo Script	M	20	Media
Eliminar un proceso tipo Procedimiento almacenado	S	8	Media
Eliminar un proceso tipo Script	S	8	Media
Visualizar los procesos registrados	XS	4	Media
Calendarizar un proceso diariamente	M	20	Alta
Calendarizar un proceso semanalmente	M	20	Alta
Calendarizar un proceso mensualmente	M	20	Alta
Calendarizar un proceso en cualquier fecha	M	20	Alta
Visualizar los procesos listos para su ejecución	M	20	Alta
Ejecutar un proceso	L	40	Alta
Visualizar el registro de la ejecución de procesos	M	20	Alta
Redactar el manual de usuario	M	20	Baja
<b>TOTAL</b>		<b>520</b>	

Realizado por: Chávez, M.; Oña, I. 2021

Se contabilizaron un total de 3 requerimientos estimados con talla L (40), 18 requerimientos estimados con talla M (20), 4 requerimientos estimados con talla S (8) y 2 requerimientos estimados con talla XS (4).

Finalmente, con las entrevistas concluidas con el cliente, se determinó que, de las 27 historias de usuario generadas, 21 se asignaron con prioridad alta, 5 con prioridad media y 1 con prioridad baja.

### 3.2.2.3. Plan de entrega

El plan de entrega o Sprint Backlog es el conjunto de ítems del Product Backlog, de acuerdo con la prioridad que poseen, se plantea ser entregado avances al cliente en cada iteración.

Los Sprints casi en su totalidad cumplieron con 80 puntos de esfuerzo, tan sólo el último contó con 40 puntos debido a los requerimientos finales que no poseían mayor complejidad.

El proyecto cuenta con un total de 7 Sprints. El plan de entrega se puede apreciar de manera detallada en la **Tabla 4-3**.

**Tabla 4-3:** Sentinel - Sprint Backlog

HU/MS	Descripción	Inicio	Fin	Puntos Estimados
<b>Sprint 1</b>				
MS_01	Determinar la arquitectura de la plataforma	9/11/2020	11/11/2020	20
MS_02	Determinar el estándar de codificación de la plataforma	11/11/2020	13/11/2020	20
MS_02	Determinar la interfaz de usuario de la plataforma	9/11/2020	13/11/2020	40
<b>Total</b>				<b>80</b>
<b>Sprint 2</b>				
MS_04	Determinar la base de datos de la plataforma	16/11/2020	20/11/2020	40
HU_01	Permitir el ingreso de administrador	16/11/2020	18/11/2020	20
HU_02	Ingresar conexiones FTP	18/11/2020	20/11/2020	20
<b>Total</b>				<b>80</b>
<b>Sprint 3</b>				
HU_03	Ingresar conexiones SQL	23/11/2020	25/11/2020	20
HU_04	Editar conexiones FTP	25/11/2020	27/11/2020	20
HU_05	Editar conexiones SQL	23/11/2020	25/12/2020	20
HU_06	Eliminar conexiones FTP	25/11/2020	26/11/2020	8
HU_07	Eliminar conexiones SQL	26/11/2020	27/11/2020	8
HU_08	Visualizar las conexiones registradas	27/11/2020	27/11/2020	4
<b>Total</b>				<b>80</b>
<b>Sprint 4</b>				
HU_09	Ingresar un proceso tipo Procedimiento almacenado	30/11/2020	2/12/2020	20
HU_10	Ingresar un proceso tipo Script	2/12/2020	4/12/2020	20
HU_11	Editar un proceso tipo Procedimiento almacenado	30/11/2020	2/12/2020	20
HU_12	Editar un proceso tipo Script	2/12/2020	4/12/2020	20
<b>Total</b>				<b>80</b>
<b>Sprint 5</b>				
HU_13	Eliminar un proceso tipo Procedimiento almacenado	7/12/2020	7/12/2020	8
HU_14	Eliminar un proceso tipo Script	8/12/2020	8/12/2020	8
HU_15	Visualizar los procesos registrados	9/12/2020	9/12/2020	4
HU_16	Calendarizar un proceso diariamente	9/12/2020	11/12/2020	20
HU_17	Calendarizar un proceso semanalmente	7/12/2020	9/12/2020	20
HU_18	Calendarizar un proceso mensualmente	9/12/2020	11/12/2020	20
<b>Total</b>				<b>80</b>
<b>Sprint 6</b>				
HU_19	Calendarizar un proceso en cualquier fecha	14/12/2020	16/12/2020	20
HU_20	Visualizar los procesos listos para su ejecución	16/12/2020	18/12/2020	20
HU_21	Ejecutar un proceso	14/12/2020	18/12/2020	40
<b>Total</b>				<b>80</b>

HU/MS	Descripción	Inicio	Fin	Puntos Estimados
<b>Sprint 7</b>				
HU_22	Visualizar el registro de la ejecución de procesos	21/12/2020	23/12/2020	20
MS_05	Redactar el manual de usuario	21/12/2020	23/12/2020	20
<b>Total</b>				<b>40</b>

Realizado por: Chávez, M.; Oña, I. 2021

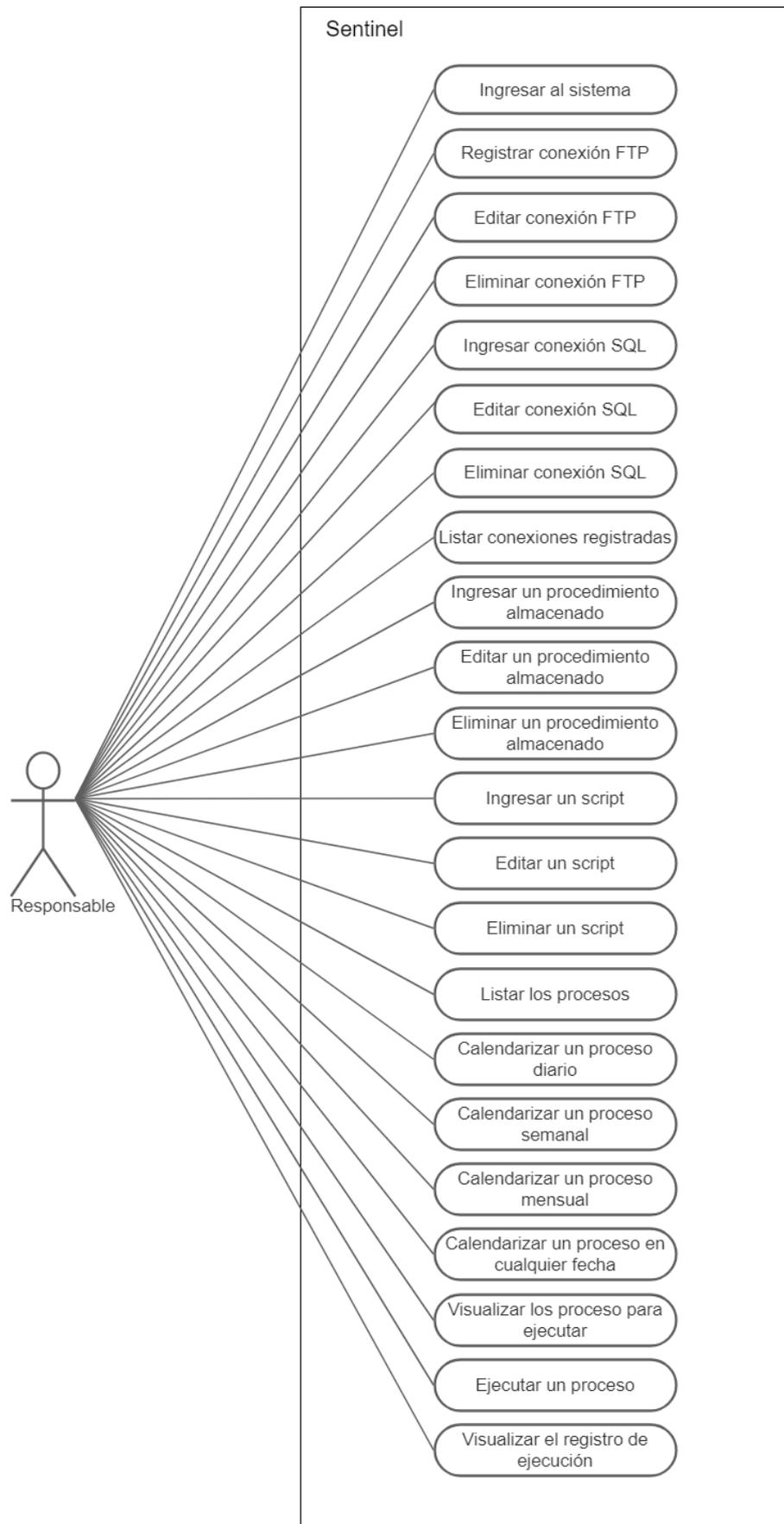
Como se muestra en la **Tabla 4-3**. Se obtuvo un total de 7 sprints mismos que muestra consistencia con 80 puntos de esfuerzo, cada historia de usuario o metáfora del sistema se detalla con su fecha correspondiente de inicio y fin junto con los puntos de esfuerzo que demandó su desarrollo.

El desarrollo de la plataforma tuvo una duración de siete semanas iniciando el 9 de noviembre de 2020 y concluyendo el 23 de diciembre de 2020.

### **3.2.3. Fase de desarrollo**

#### **3.2.3.1. Interacción del usuario con la plataforma**

Con la finalidad de comunicarse con el cliente se desarrolló un diagrama UML de Casos de Uso con la finalidad de obtener una representación visual del comportamiento que se espera de la plataforma en un caso de un usuario concreto.



**Figura 1-3:** Sentinel - Diagrama de casos de uso

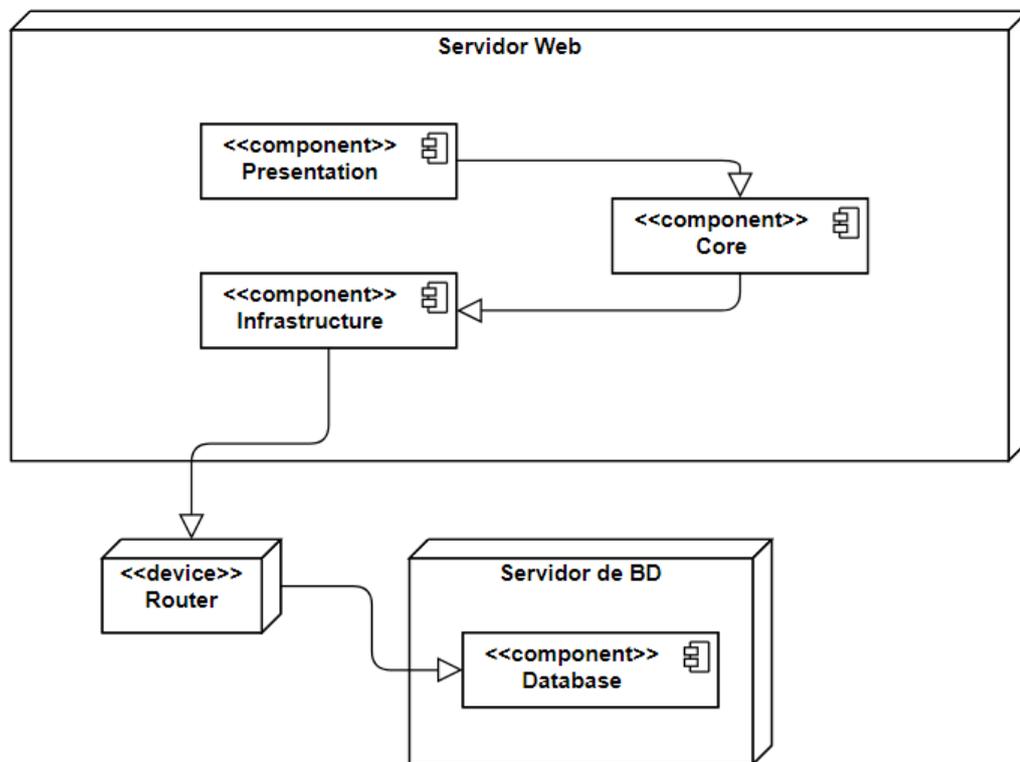
Realizado por: Chávez, M.; Oña, I. 2021

El usuario podrá hacer uso de todas las funcionalidades que se observan en la **Figura 1-3** sin restricción alguna.

Al momento solo se tienen un único rol, el de responsable, quien tiene el control total de la plataforma, a futuro es recomendable gestionar roles adicionales para la gestión específica de las conexiones a los distintos servidores.

### 3.2.3.2. Arquitectura de la plataforma

La arquitectura del software muestra la interacción que tienen los distintos componentes de la plataforma a nivel de hardware y software tomando en cuenta la infraestructura que dispone la empresa en la que será implementada la misma, para tener un mejor entendimiento se desarrolló un diagrama de despliegue que se muestra a continuación.



**Figura 2-3:** Sentinel - Diagrama de despliegue de la plataforma

**Realizado por:** Chávez, M.; Oña, I. 2021

En Tracklink se han destinado dos servidores para la implementación de la plataforma, un servidor web para montar la aplicación web y el worker service, y un servidor de base de datos SQL Server 2019.

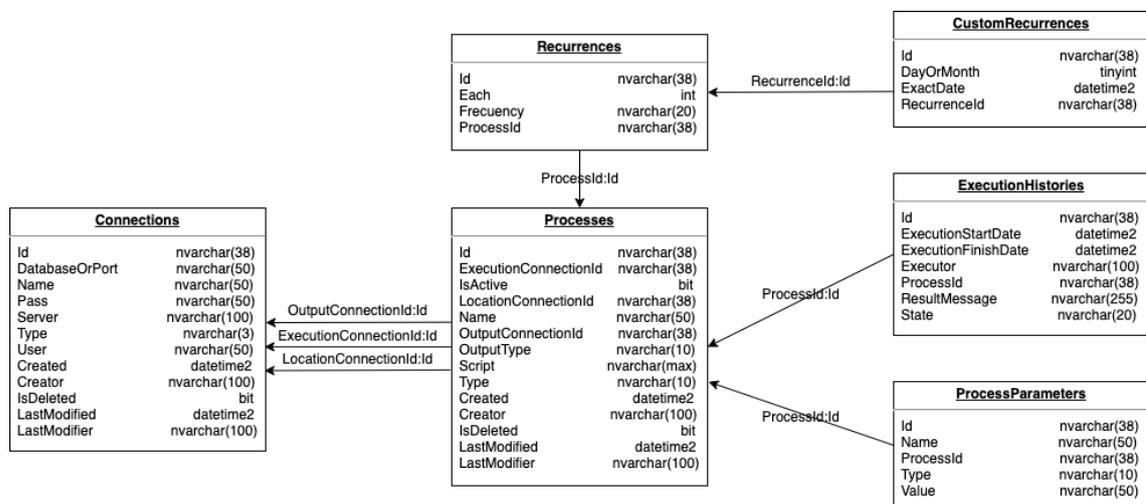
La arquitectura que se definió está compuesta por 2 niveles y 4 capas como se describe en la **Figura 2-3**.

Una vez concluido el desarrollo, se procedió con despliegue de la plataforma, misma que respetó la arquitectura planteada, la empresa facilitó los dos servidores necesarios para ejecutar dicha acción.

### 3.2.3.3. Diseño de la base de datos

Para el diseño de la base de datos, fue primordial el tema de registros de eventos, dado que se debe llevar un control de las ejecuciones de los procesos, razón por la cual se optó por una base de datos relacional.

El diagrama físico de la base de datos de la plataforma Sentinel se encuentra en la **Figura 3-3**, que se visualiza a continuación:



**Figura 3-3:** Sentinel - Modelo físico de la base de datos

Realizado por: Chávez, M.; Oña, I. 2021

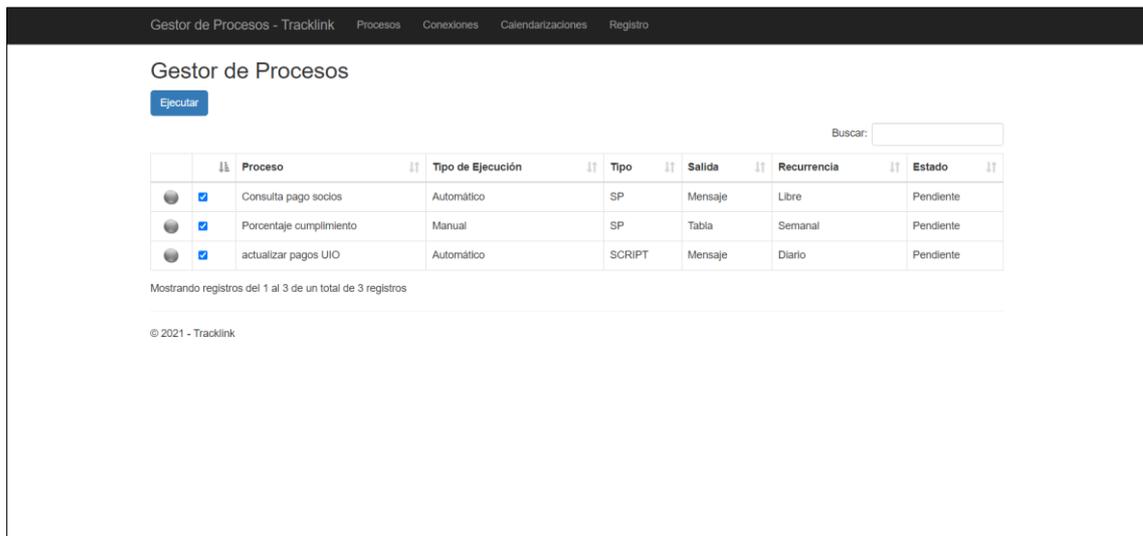
La base de datos resultante de la plataforma está compuesta por 6 tablas mismas que se relacionan por un identificador único por cada registro, en donde el mayor peso recae sobre la tabla Processes misma que almacena los datos de los procesos a ejecutarse, para ello, hace uso de los datos contenidos en la tabla Connections para acceder y/o ejecutar dichos procesos, en caso de que estos tengan recurrencia, los datos de las fechas de repetición se guardan en las tablas Recurrences y CustomRecurrences.

El modelo generado es capaz de soportar el manejo de conexiones a servidores de base de datos y directorios ftp para la localización, ejecución y/o salida de procesos, pero en caso de requerir acceder a otro tipo de fuentes, como rutas compartidas, se recomienda considerar una nueva tabla que maneje este nuevo tipo de conexiones.

#### 3.2.3.4. Diseño de la interfaz de usuario

La interfaz de usuario es de gran importancia debido a que el usuario estará en permanente interacción, sin embargo, además de mantener una buena y presentable interfaz es necesario centrarse también en la experiencia que genera el uso de la plataforma, sin una buena experiencia de usuario por muy buena que sea la interfaz no tendrá impacto.

La plataforma Sentinel mantiene una interfaz con colores neutros como principales juntamente con uno de acento propio de la empresa, también maneja un semáforo de colores para indicar el estado de ejecución demostrando una simplicidad de forma que el usuario aprenda con facilidad la funcionalidad de la plataforma generando una buena experiencia.



**Figura 4-3:** Sentinel - Pantalla de inicio

**Realizado por:** Chávez, M.; Oña, I. 2021

La interfaz se maneja bajo un estándar mismo que de forma general incluye una barra de navegación superior, una lista de forma tabular donde se muestra la información de manera ordenada, una entrada de búsqueda para facilitar al usuario ubicar la información requerida, botones de acción principal y finalmente en la parte inferior un copyright por parte de la empresa como se puede observar en la **Figura 4-3**.

Gestor de Procesos - Tracklink | Procesos | Conexiones | Calendarizaciones | Registro

### Nuevo Proceso

Nombre

Tipo

Tabla de salida  
 Desactivado

Nombre del procedimiento almacenado

Conexión de ejecución

Conexión de salida para la tabla

© 2021 - Tracklink

**Figura 5-3:** Sentinel - Registro de proceso tipo SP

**Realizado por:** Chávez, M.; Oña, I. 2021

Como se puede observar en la **Figura 5-3**, para el registro de datos se tienen campos de entrada de texto con sus respectivas etiquetas en la parte superior, cajas de confirmación y combos de selección de opciones, se mantiene la barra de navegación superior, un botón de acción principal y en la parte inferior un copyright por parte de la empresa.

#### 3.2.3.5. Estándar de codificación

Para tener un código limpio y fácil de manejar se emplearon varios estándares de nomenclatura y buenas prácticas, que, además, permitió mejorar su entendimiento. A continuación, se mencionan los estándares y buenas prácticas empleados en el desarrollo de la plataforma:

- **Notación Pascal:** La primera letra va en mayúscula y el resto en minúsculas, esto se lo hace por cada palabra como se muestra en el siguiente ejemplo: NombreDeUnaClase, NombreDeUnMétodo.

Esta nomenclatura se usó para clases, funciones y métodos.

- **Notación Camel:** La primera letra excepto de la primera palabra se escriben con mayúsculas, el resto con minúscula como en el siguiente ejemplo: nombreDeUnaVariable

La nomenclatura se usó para denominar variables locales, cabe mencionar que se usó nombres descriptivos para nombrar a cada una de las variables.

- **Notación para miembros privados:** EL inicio de un identificador parte con un `_` y el resto de los caracteres se apega a la notación camel como en el siguiente ejemplo:  
`_registrosPrivados`

La nomenclatura se usó para denominar variables globales privadas a nivel de clases.

Dentro de las buenas prácticas se emplearon las siguientes:

- Para el espaciado y sangría se usó un TAB
- Las llaves ‘`{}`’ están alineadas al código fuera de estas
- Una línea vacía para separar un bloque de código lógico dentro de un método
- Las llaves ‘`{}`’ están separadas de por una línea de cualquier estructura como `if`, `for`, `while`, etc.
- Un espacio simple separa variables de operadores lógicos como muestra el ejemplo: `(a > b)`.
- Los archivos con extensión `.cs` están separados por regiones, agrupando variables, propiedades, constructores, métodos, etc.

#### 3.2.3.6. *Desarrollo*

El desarrollo del proyecto tuvo como fin satisfacer todos los requerimientos funcionales previamente expuestos por el cliente para lo cual se definió el desarrollo de una aplicación web y un worker service como solución planteada.

Debido a la designación de tareas de desarrollo dentro del equipo, la codificación de cada historia de usuario se la ejecutó de manera individual, misma que para marcarse como completada debía subirse al repositorio remoto de código, junto con sus pruebas unitarias y su respectiva documentación.

Los dos componentes se desarrollaron con el framework `.NET`, tecnología que se definió en la fase de exploración, se hizo uso del IDE Visual Studio 2019 junto con el editor Visual Studio Code para el desarrollo completo de la plataforma.

Se codificaron un total de 3 módulos, conocidos como capas dentro de la arquitectura utilizada, el primer módulo es el de “Core” mismo que contiene las entidades y casos de uso de la plataforma, dentro de estos el worker service, el segundo módulo es el de “Infrastructure” que contiene principalmente los mecanismos de comunicación con la base de datos, mismos que se

manejan bajo un ORM, finalmente el tercer módulo es el de “Presentation” que contiene las interfaces gráficas que exponen al usuario final siendo esta el punto de entrada y salida de datos de la aplicación web.

El diseño de interfaz de usuario se planteó de la forma más sencilla, por conveniencia del cliente para que su uso sea simple, la presentación de información de forma tabular resultó intuitiva y ordenada, la gestión de información cuenta con la descripción necesaria y suficiente para que el usuario ingrese los datos correctamente.

El diseño resultante de la base de datos logró satisfacer las necesidades de la plataforma, debido a que se pudieron optimizar las relaciones entre las tablas de los procesos y conexiones, además de que también es posible realizar un seguimiento de los responsables en los cambios en la información dentro de la base de datos.

A lo largo del desarrollo el estándar de codificación que se definió fue una guía indispensable para llevar un código ordenado, la comprensión de este para agilizar, y facilitar el desarrollo de los requerimientos; es recomendable siempre definir y respetar un estándar de codificación y buenas prácticas para esta fase.

Como resultado del desarrollo de la plataforma se obtuvo la aplicación web junto con el worker service, a partir de los cuales se contabilizaron sus líneas de código haciendo uso del programa “cloc”, los resultados se muestran en la siguiente tabla.

**Tabla 5-3:** Sentinel - LOCs

Lenguaje	Archivos	Líneas en blanco	Líneas de comentarios	Líneas de código
JavaScript	9	5204	8338	18187
CSS	6	30	26	7451
C#	99	833	532	7166
Razor	24	30	3	1402
<b>TOTAL</b>	<b>138</b>	<b>6097</b>	<b>8899</b>	<b>34206</b>

Realizado por: Chávez, M.; Oña, I. 2021

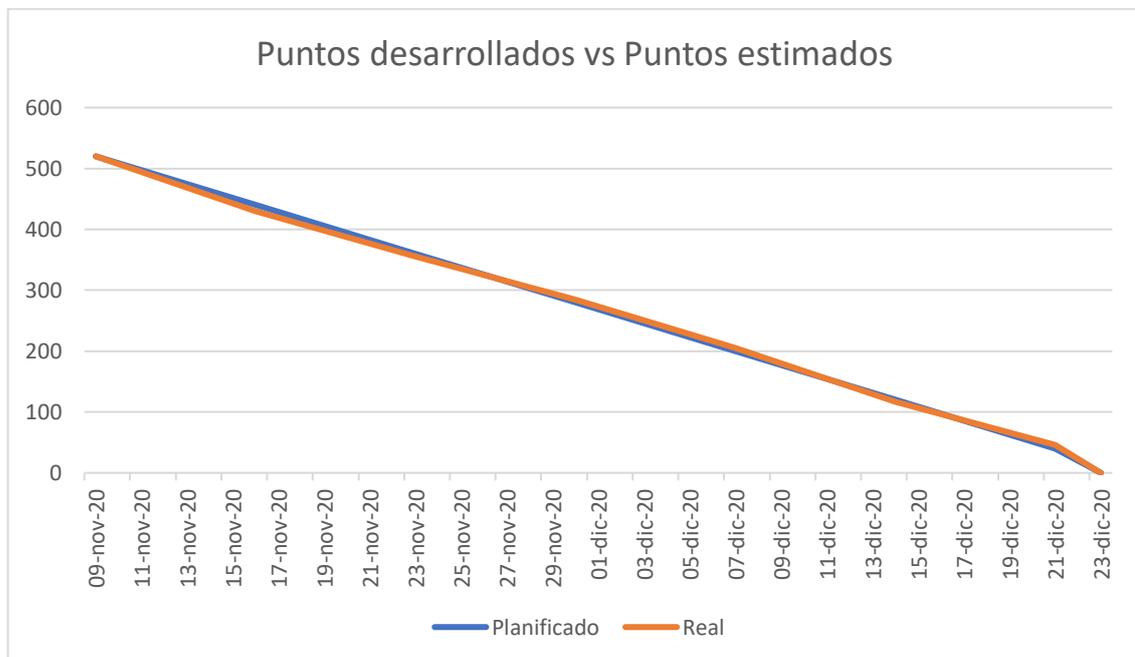
Al ser un desarrollo basado en .NET la mayor cantidad de archivos resultantes corresponden a archivos codificados en el lenguaje C#, como se puede observar en la **Tabla 5-3**, a pesar de que la mayor cantidad de líneas de código correspondan al lenguaje JavaScript, utilizado como lenguaje del lado del cliente en la aplicación web, esto se debe principalmente al uso de librerías de terceros para facilitar el desarrollo de funcionalidades.

Por otra parte, se puede observar, aunque en menor cantidad, archivos Razor y CSS para el maquetado y diseño de las interfaces de la aplicación web respectivamente, es importante mencionar que Razor es una combinación de HTML con C# puesto que permite embeber código C# en un archivo con etiquetas HTML.

### 3.2.4. Fase de cierre

Se mantuvo una comunicación cercana con el cliente a lo largo del desarrollo, se definieron reuniones diarias y en el caso de que alguno de los involucrados no pudiese asistir a l reunión todos los avances se manejaban vía mensaje de texto, estas decisiones mantuvieron un gran compromiso tanto en el equipo de desarrollo como en el cliente incrementando la confianza y fiabilidad, puntos importantes para mejorar la comunicación.

Adicionalmente se pactaron reuniones de entrega cada lunes de cada semana para presentar el desarrollo de cada sprint, en estas reuniones se obtuvo adicionalmente puntos de vista en retrospectiva por parte del cliente sobre el buen trabajo y cumplimiento que se mantenía, aunque también existieron observaciones, enfocadas principalmente a la seguridad de la información, que se tomaron en cuenta para mejorar el proceso, cabe mencionar que no existieron cambios en cuanto a los requerimientos por lo cual se respetó la planificación establecida y no hubo necesidad de recurrir a una replanificación.



**Gráfico 1-3:** Sentinel - Burndown chart

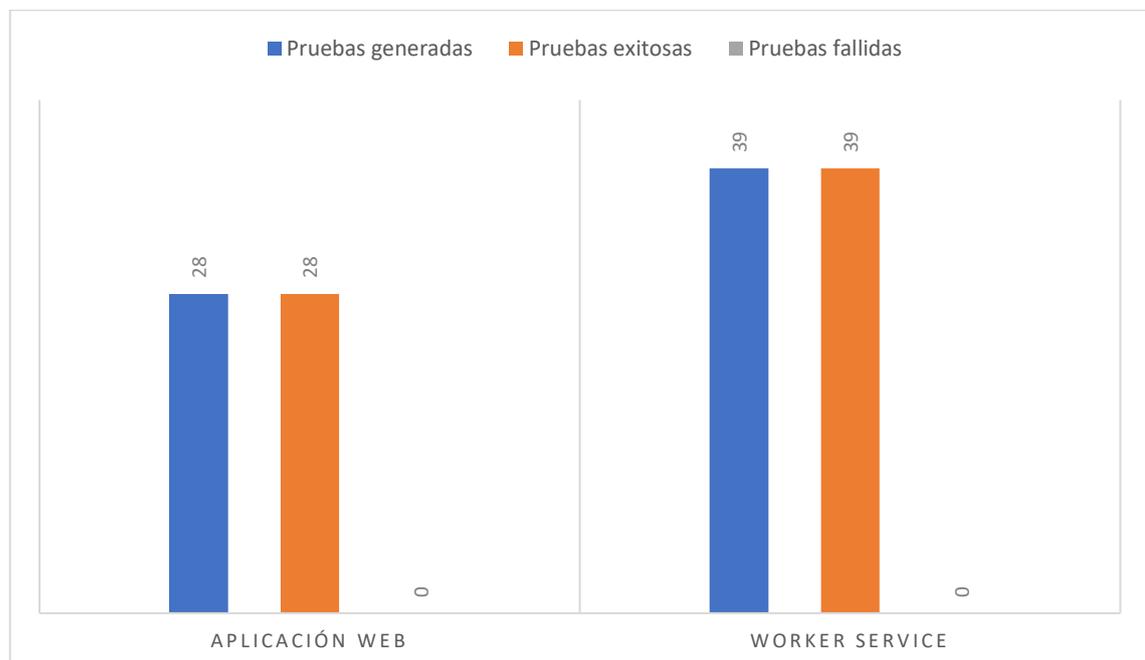
**Realizado por:** Chávez, M.; Oña, I. 2021

Como se muestra en el **Gráfico 1-3**, se cumplió en el tiempo que se estableció para la entrega final, aunque los puntos reales de cada sprint no fueron completamente acordes con los planificados, sin embargo, las variaciones no presentaron mayores inconvenientes para culminar en la fecha esperada

### 3.2.5. Cumplimiento

Los requisitos fueron desarrollados en su totalidad, en las reuniones de entrega se validó el cumplimiento de los requisitos tanto en su funcionalidad como en los tiempos de entrega pactados, además de las reuniones semanales se desarrollaron gráficos tipo burndown chart para mostrar el avance y observar si el desarrollo iba acorde a lo planificado, a pesar de haber desfases en algunos puntos estimados siempre se cumplió con la entrega semanal.

Adicionalmente se generó un proyecto de pruebas unitarias que, aunque no interviene directamente en la puesta en producción de la plataforma, cuenta con un total de 67 casos de prueba a partir de las cuales se obtuvieron los índices que se pueden observar en el **Gráfico 2-3**.



**Gráfico 2-3:** Sentinel - Pruebas unitarias exitosas vs fallidas

Realizado por: Chávez, M.; Oña, I. 2021

Como se puede observar en el **Gráfico 2-3**, del total de pruebas unitarias generadas, el 41.79% corresponde a la aplicación web y 58.21% al worker service, de las cuales el 100% fueron exitosas, asegurando de esta manera la calidad interna del producto desarrollado.

- **Analizar los tiempos de retraso en la ejecución de los procesos de adecuación de la información.**

Luego de concluir con el desarrollo de la plataforma se procede a determinar los métodos de evaluación, para determinar si existe o no diferencia en los tiempos de retraso de la ejecución de los procesos de adecuación de la información al usar la plataforma frente al mecanismo inicial.

### 3.3. Tiempo de retraso del proceso actual de adecuación de la información

Se construyó una encuesta longitudinal auto administrada, razón principal por la cual cabe destacar que esta es anónima pero dirigida exclusivamente a las 7 personas encargadas, con la intención de definir una semana como tiempo objetivo en la cual se observa los tiempos de retraso en la ejecución de los procesos, dicha encuesta (**Anexo B**) constó con 5 preguntas cerradas con respuestas de tipo numérica mismas que fueron evaluadas y verificadas por los tesisistas donde se descartaron preguntas con inconsistencias y términos con ambigüedad que puedan generar confusión en los encuestados y obtener resultados con mayor confianza, las preguntas de esta se pueden observar en la **Tabla 6-3**

**Tabla 6-3:** Preguntas de la encuesta

N°	Pregunta
1	¿Cuántos procesos de tipo script ejecuta a la semana?
2	¿Cuántos procesos de tipo procedimiento almacenado ejecuta a la semana?
3	¿Cuánto tiempo aproximado en segundos se demora en preparar todos los procesos de tipo script de la semana para ejecutarlos?
4	¿Cuánto tiempo aproximado en segundos se demora en preparar todos los procesos de tipo procedimiento almacenado de la semana para ejecutarlos?
5	¿Cuánto tiempo aproximado en segundos a la semana se demora en acceder al ambiente para ejecutar los procesos?

Realizado por: Chávez, M.; Oña, I. 2021

El objetivo principal de la encuesta fue recolectar información sobre el número de procesos que el personal ejecuta en una semana y el tiempo que conlleva su preparación, actividad que se realiza siempre antes de su ejecución.

La encuesta denominada “Encuesta de evaluación de procesos de adecuación de la información” fue aplicada por los tesisistas Israel Oña y Mario Chávez el 5 de noviembre de 2020 hacia los 7 responsables de ejecutar los procesos mismos que pertenecen al departamento de TI, al ser una población tan pequeña no se realizó ningún muestro, razón por la cual esta se aplicó a toda la población.

Para la obtención de las respuestas se clasificaron en dos grupos, el primero basado en el criterio de cantidad obteniendo un total de procesos ejecutados a la semana y el segundo por el criterio de tiempo, obteniendo así el total de tiempo, medido en segundos, que se tardan en preparar el total de los procesos semanales para posteriormente ejecutarlos.

### **3.4. Tiempo de retraso del proceso de adecuación de la información con Sentinel**

Para conocer los tiempos de retraso existentes en la ejecución de los procesos con la plataforma, se procedió a recuperar la información de los registros de ejecución de esta, en donde principalmente se encuentran datos como la calendarización de los procesos y la hora exacta del inicio de la ejecución del proceso en el motor de base de datos.

A nivel de la calendarización de los procesos dentro de la plataforma, todos se ejecutan a partir de las 23 horas, debido a que desde las 22 horas la carga de los servidores, web y de bases de datos, empieza a disminuir y se puede aprovechar la máxima capacidad de estos.

El servidor web donde se encuentran desplegados los componentes como el worker service, presenta las siguientes características:

- Procesador: Intel Xeon E5-2680 v2 2.80 GHz
- Memoria RAM: 8 GB
- Disco duro: HDD 160 GB
- Sistema operativo: Windows Server 2016

De los registros obtenidos es necesario obtener una muestra sobre la cual se aplicarán los análisis, para ello, dado a que la población es infinita debido que los procesos podrían ejecutarse infinidad de veces, se ha planteado obtener una muestra con un grado de confianza del 95% y con un margen de error que no supere el 5%.

Nivel de confianza del 95% (0.95)

$$\alpha = 1 - 0.95 = 0.05$$

$$z = 1 - \frac{\alpha}{2} = 0.975$$

$z = 1.96$ ; según la distribución normal para alfa 0.05 (nivel de significancia)

$e = 0.05$ ; error de estimación no supere el 5%

$p = 0.5$ ; proporción aproximada del fenómeno en estudio en la población de referencia

$q = 0.5$ ; proporción de la población de referencia que no presenta el fenómeno en estudio ( $1 - p$ )

$$n = \frac{z^2 * p * q}{e^2}$$

$$n = \frac{1.96^2 * 0.5 * 0.5}{0.05^2}$$

$$n = 384.16 = 385$$

El tamaño de la muestra resultante es de 385 registros, mismos que se tomaron de forma aleatoria por cada una de las semanas, esto debido a que todos los procesos se repiten cada semana, la cantidad de muestras por cada uno de los días está basada en el porcentaje correspondiente al total de procesos que se ejecutan en cada día con respecto al total de procesos calendarizados por semana como se puede observar en la **Tabla 7-3**.

**Tabla 7-3:** Estratificación de la muestra de procesos ejecutados con Sentinel

Días	Procesos	Porcentaje	Muestras diarias
Lunes	5	14.29%	55
Martes	3	8.57%	33
Miércoles	4	11.43%	44
Jueves	6	17.14%	66
Viernes	5	14.29%	55
Sábado	8	22.86%	88
Domingo	4	11.43%	44
<b>Total</b>	<b>35</b>	<b>100%</b>	<b>385</b>

Realizado por: Chávez, M.; Oña, I. 2021

Todas las semanas se ejecutan un total de 35 procesos, dato obtenido a través de la encuesta aplicada con anterioridad a los responsables de los procesos y que a su vez es consistente con la cantidad de procesos calendarizados en la aplicación web.

- **Evaluar el nivel de eficiencia de la plataforma propuesta en base a la norma ISO/IEC 25010**

Para evaluar la eficiencia de esta plataforma en base al comportamiento temporal según la ISO 25010, se realizó un estudio comparativo entre el tiempo que le toma a los responsables de estos

prepararlos manualmente frente al tiempo que le toma a la plataforma prepararlos automáticamente.

Semanalmente se ejecutan, de forma consistente en el tiempo, un total de 35 procesos, entre scripts y procedimientos almacenados, que se usaron en dos entornos para la evaluación, por el lado de la preparación a cargo de los responsables, se denomina “entorno sin sistema”, mientras que por el lado de la preparación a cargo del worker service, se denomina “entorno con sistema” como se puede observar en la **Tabla 8-3**

**Tabla 8-3:** Procesos ejecutados por cada entorno

<b>Número de procesos</b>	<b>Entorno</b>	<b>Procesos</b>
385	Con Sistema	Scripts y procedimientos almacenados
35	Sin Sistema	Scripts y procedimientos almacenados

**Realizado por:** Chávez, M.; Oña, I. 2021

## CAPÍTULO IV

### 4. RESULTADOS

En este capítulo se analizan los resultados obtenidos en la experimentación planteada. Estos resultados definen la aceptación o rechazo de una de las hipótesis planteadas.

Los resultados obtenidos fueron recopilados y analizados con la ayuda de hojas de cálculo, para una mejor comprensión de estos; en primer lugar, se presentan los del experimento sin el sistema, datos obtenidos de las encuestas aplicadas al personal del departamento de TI, y en segundo lugar los datos con el sistema, obtenidos por medio de la observación.

#### 4.1. Análisis del tiempo de retraso del proceso actual de adecuación de la información

Los datos que se obtuvieron en este experimento fueron el número de procesos que el personal ejecuta a la semana y el tiempo que les conlleva acceder al ambiente para posteriormente adecuarlos. Las respuestas fueron agrupadas en dos grupos: cantidad y tiempo, mismas que se encuentran en el (**Anexo C**)

##### 4.1.1. Tabulación de las encuestas

Las preguntas se dividieron en dos partes las cuales son: la pregunta 1 y 2 identifican la cantidad de procesos que se ejecutan y las preguntas 3, 4 y 5 determinan los tiempos de retraso en la ejecución, a continuación, se detalla cada una de las preguntas con sus resultados.

##### 4.1.2. Cantidad de scripts ejecutados por semana

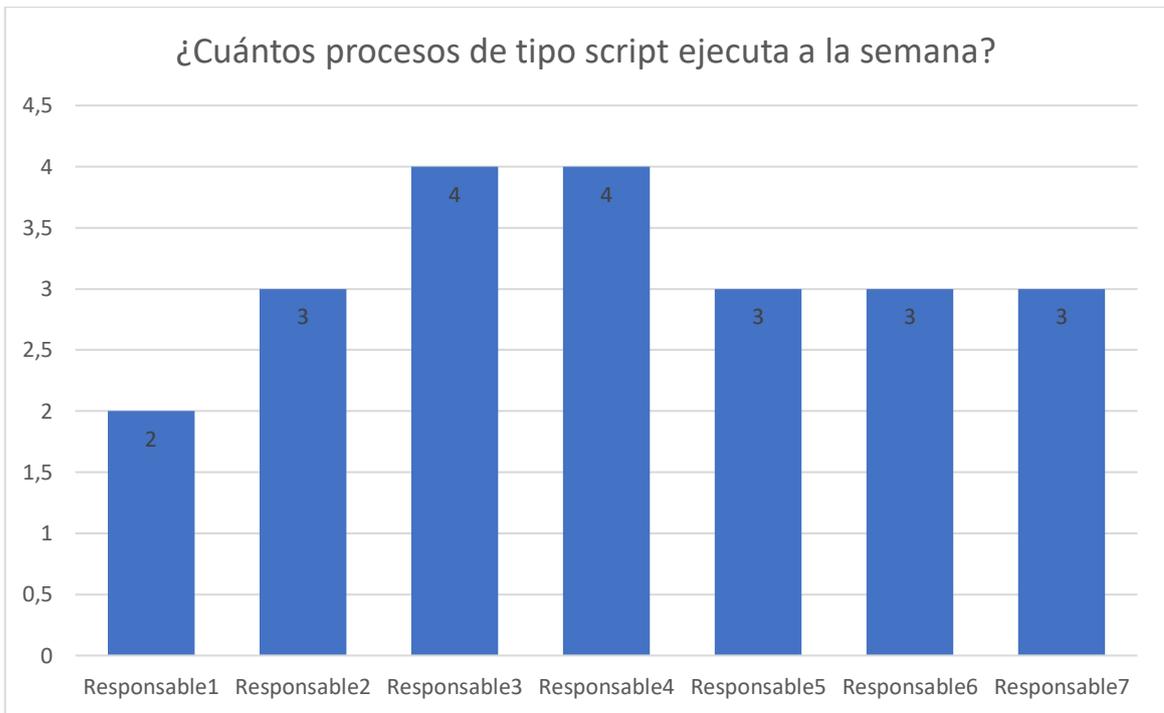
¿Cuántos procesos de tipo script ejecuta a la semana?

En esta pregunta se establece el número de script que cada responsable ejecuta en la semana como se puede apreciar en la **Tabla 1-4** dando un total de 22 procesos de tipo script ejecutados a la semana por los siete responsables en el departamento de TI.

**Tabla 1-4:** Cantidad de scripts ejecutados por semana

Responsable	Scripts
Responsable 1	2
Responsable 2	3
Responsable 3	4
Responsable 4	4
Responsable 5	3
Responsable 6	3
Responsable 7	3
<b>TOTAL</b>	<b>22</b>

Realizado por: Chávez, M.; Oña, I. 2021



**Gráfico 1-4:** Cantidad de scripts ejecutados por semana

Realizado por: Chávez, M.; Oña, I. 2021

#### 4.1.3. Cantidad de procedimientos almacenados ejecutados por semana

¿Cuántos procesos de tipo procedimiento almacenado ejecuta a la semana?

En esta pregunta se establece cuantos procedimientos de tipo almacenado están ejecutando los responsables como se ve en la **Tabla 2-4** dando un total de 13 procesos de tipo procedimiento almacenado ejecutados a la semana por los siete responsables en el departamento de TI.

**Tabla 2-4:** Cantidad de procedimientos almacenados ejecutados por semana

Responsable	Procedimientos almacenados
Responsable 1	1
Responsable 2	2
Responsable 3	1
Responsable 4	3
Responsable 5	3
Responsable 6	1
Responsable 7	2
<b>TOTAL</b>	<b>13</b>

Realizado por: Chávez, M; Oña Israel. 2021



**Gráfico 2-4:** Cantidad de procedimientos almacenados ejecutados por semana

Realizado por: Chávez, M.; Oña, I. 2021

#### 4.1.4. Tiempo de preparación de scripts por semana

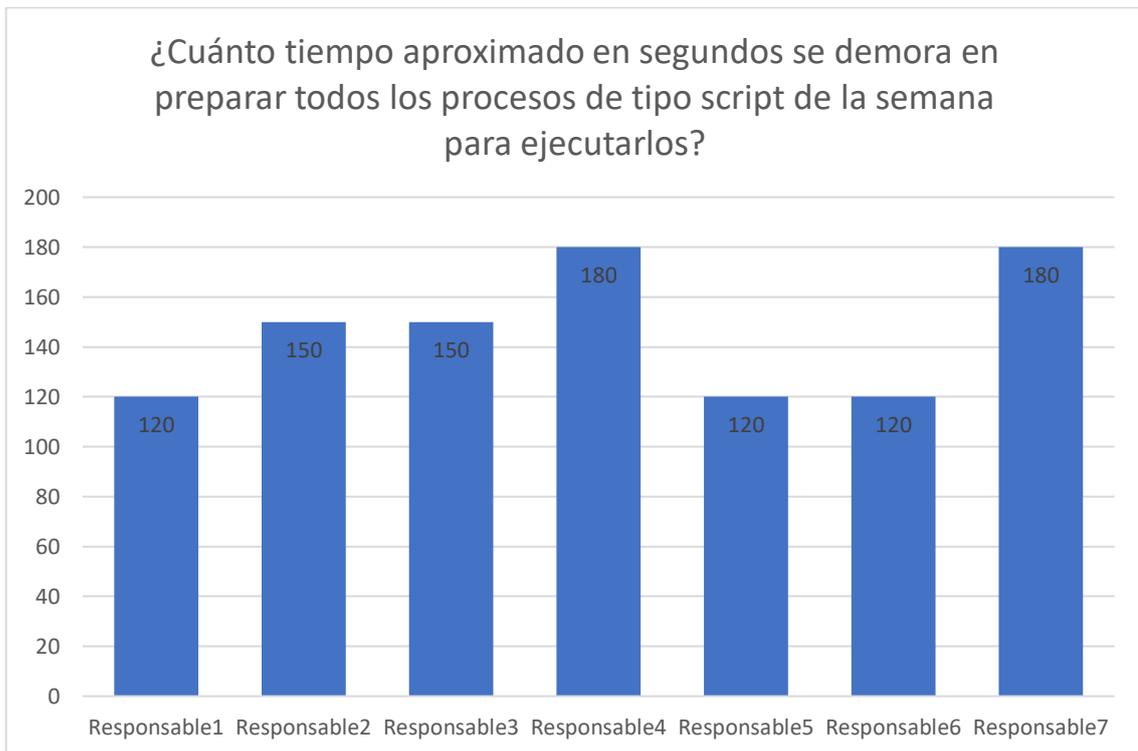
¿Cuánto tiempo aproximado en segundos se demora en preparar todos los procesos de tipo script de la semana para ejecutarlos?

A continuación, en la **Tabla 3-4** se observa el tiempo que le toma a los responsables preparar todos los procesos de tipo script para poder ejecutarlos dando como resultado 1020 segundos en preparar los procesos de tipo script a la semana por los 7 responsables en el departamento de TI.

**Tabla 3-4:** Tiempo de preparación de scripts por semana

<b>Responsable</b>	<b>Segundos</b>
Responsable 1	120
Responsable 2	150
Responsable 3	150
Responsable 4	180
Responsable 5	120
Responsable 6	120
Responsable 7	180
<b>TOTAL</b>	<b>1020</b>

Realizado por: Chávez, M.; Oña, I. 2021



**Gráfico 3-4:** Tiempo de preparación de scripts por semana

Realizado por: Chávez, M.; Oña, I. 2021

#### 4.1.5. Tiempo de preparación de procedimientos almacenados por semana

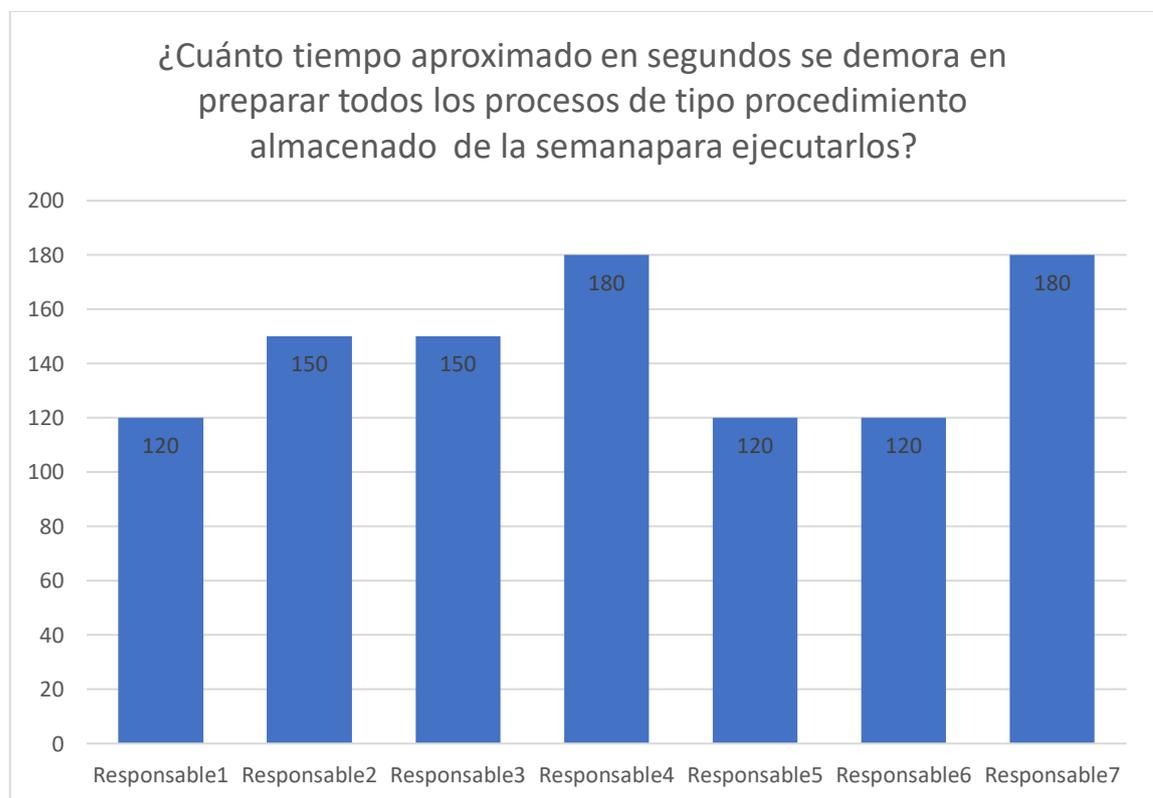
¿Cuánto tiempo aproximado en segundos se demora en preparar todos los procesos de tipo procedimiento almacenado de la semana para ejecutarlos?

De la misma manera que en la anterior pregunta aquí se establece el tiempo en segundos que le toma en preparar a los responsables los procesos de tipo procedimiento almacenado como se aprecia en la **Tabla 4-4** dando como resultado 810 segundos en preparar los procesos de tipo procedimiento almacenado a la semana por los 7 responsables en el departamento de TI.

**Tabla 4-4:** Tiempo de preparación de procedimientos almacenados por semana

<b>Responsable</b>	<b>Segundos</b>
Responsable 1	120
Responsable 2	150
Responsable 3	30
Responsable 4	150
Responsable 5	180
Responsable 6	60
Responsable 7	120
<b>TOTAL</b>	<b>810</b>

Realizado por: Chávez, M.; Oña, I. 2021



**Gráfico 4-4:** Tiempo de preparación de procedimientos almacenados por semana

Realizado por: Chávez, M.; Oña, I. 2021

#### 4.1.6. Tiempo de acceso a los ambientes de ejecución por semana

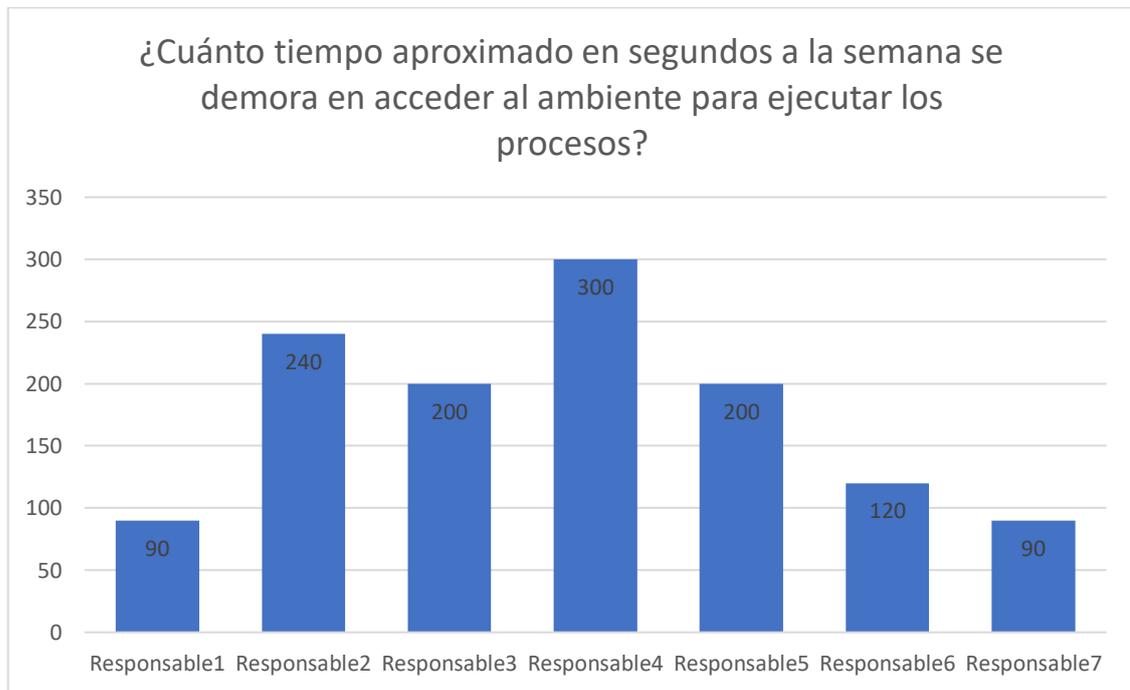
¿Cuánto tiempo aproximado en segundos a la semana se demora en acceder al ambiente para ejecutar los procesos?

En esta pregunta se plantea en cuantos segundos durante toda la semana se demora en acceder al ambiente para ejecutar los procesos descritos anteriormente dando como resultado un total de 1240 segundos a la semana que les toma a los 7 responsables acceder al ambiente **Tabla 5-4**.

**Tabla 5-4:** Tiempo de acceso a los ambientes de ejecución por semana

Responsable	Segundos
Responsable 1	90
Responsable 2	240
Responsable 3	200
Responsable 4	300
Responsable 5	200
Responsable 6	120
Responsable 7	90
<b>TOTAL</b>	<b>1240</b>

Realizado por: Chávez, M.; Oña, I. 2021



**Gráfico 5-4:** Tiempo de acceso a los ambientes de ejecución por semana

Realizado por: Chávez, M.; Oña, I. 2021

En base a las respuestas de las preguntas uno y dos se determinó que los responsables ejecutan un total de 35 procesos entre scripts y procedimientos almacenados a la semana, y con las respuestas de las preguntas tres, cuatro y cinco se obtuvo un total aproximado de 3070 segundos de retraso en la ejecución del total de estos procesos.

Con los datos obtenidos por cada uno de los responsables, se pueden determinar los siguientes tiempos promedios de retraso:

- Preparación de scripts: 145 segundos con 714 milisegundos
- Preparación de procedimientos almacenados: 131 segundos con 428 milisegundos
- Acceso al ambiente: 177 segundos con 142 milisegundos

De los tiempos promedios anteriores es posible establecer que el mayor tiempo de retraso se encuentra en la conexión a los ambientes, mismo que representa un 38.99% del total, principalmente debido a que todos los procesos deben ejecutarse en servidores a los cuales es posible acceder únicamente por medio de conexiones seguras a través de un punto de acceso en una red privada, es por esto por lo que establecer dicha conexión impacta directamente en los tiempos de retraso.

#### **4.2. Análisis del tiempo de retraso del proceso de adecuación de la información con Sentinel**

Los datos que se obtuvieron en este experimento fueron la diferencia de tiempo que existe entre la calendarización de un proceso y su ejecución mismos que se encuentra en el (**Anexo D**), y el tiempo total de retraso de ejecución de todos los procesos de la semana.

##### **4.2.1. Tabulación de los datos observados**

De la muestra de 385 procesos, estratificados por día como se muestra en la tabla 6-4, se tomaron dos valores de tiempo por cada uno, la hora de calendarización y la hora en que estos son ejecutados, siendo la diferencia entre estas dos horas el dato de interés; con el objetivo de facilitar la lectura de estos datos para su análisis, se los agruparon según la estratificación definida, como se puede observar a continuación.

**Tabla 6-4:** Tiempos de retraso en la ejecución por día con Sentinel

<b>Días</b>	<b>Segundos</b>
Lunes	99.934
Martes	66.233
Miércoles	78.025
Jueves	147.585
Viernes	98.482
Sábado	220.218
Domingo	81.677

**Realizado por:** Chávez, M.; Oña, I. 2021

A partir de estos datos se obtuvo un total de 792 segundos con 154 milisegundos de diferencia entre la calendarización y la ejecución de los 385 procesos, de las cuales se obtuvo un promedio

de 2 segundos con 58 milisegundos de tiempo de retraso por proceso como se muestra en la **Tabla 7-4**

**Tabla 7-4:** Resumen de tiempos con Sentinel

<b>Retraso</b>	<b>Segundos</b>
En el total de muestras	792.154
En promedio por proceso	2.057

**Realizado por:** Chávez, M.; Oña, I. 2021

Con la información obtenida del tiempo promedio de retraso en la ejecución de cada proceso, se determinó que a la semana esta plataforma se tarda 72 segundos con 14 milisegundos en empezar a ejecutar los 35 procesos calendarizados.

Para evaluar la eficiencia de la plataforma se comparó el promedio de segundos que tardan los responsables frente al promedio de segundos que tarda el worker service como se puede observar en la **Tabla 8-4**.

**Tabla 8-4:** Tiempo medio en los dos entornos

<b>Entorno</b>	<b>Segundos</b>
Sin sistema	87.715
Con sistema	2.057

**Realizado por:** Chávez, M.; Oña, I. 2021

### 4.3. Análisis

Una vez concluido el análisis descriptivo de los resultados se procede a realizar el análisis inferencial. Considerando el tamaño de la muestra que se debe evaluar, para el análisis inferencial se optó por el método test Z para una muestra.

Para saber si la distribución de una muestra tiene una distribución normal se dice que el nivel de significancia debe ser menor a 0.05. Para definir la distribución de la muestra se procedió a hacer dos pruebas de normalidad para verificar los resultados, como se puede apreciar en la **Tabla 9-4**.

**Tabla 9-4:** Pruebas de normalidad de la muestra

	<b>Pruebas de normalidad</b>					
	<b>Kolmogorov-Smirnov</b>			<b>Shapiro-Wilk</b>		
	Estadístico	gl	Sig.	Estadístico	gl	Sig.
Muestra con el Sistema	.145	385	.000	.921	385	.000

**Realizado por:** Chávez, M.; Oña, I. 2021

Como se puede observar en la **Tabla 9-4** de las pruebas de normalidad realizadas, se obtuvo que el nivel de significancia es menor al 0.05 dando como resultado que la muestra a trabajar tiene una distribución normal.

Dentro de la evaluación es importante conocer si existe o no variación en los tiempos de preparación de los procesos en los dos entornos planteados por lo que se plantearon las siguientes hipótesis:

- Hipótesis nula: No existe diferencia en los tiempos de retraso con y sin la plataforma Sentinel.
- Hipótesis alterna: Existe diferencia en los tiempos de retraso con y sin la plataforma Sentinel.

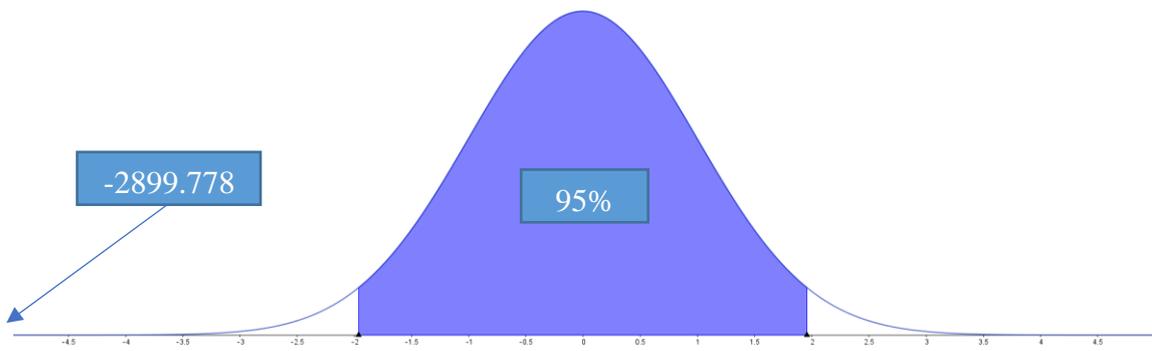
Al estar ya definidas las hipótesis de la investigación, haciendo uso de una herramienta estadística, se procedió a aplicar el método test Z para una muestra, para obtener así el valor de Z que representa la cantidad de desviación estándar respecto al promedio, y por otro lado el valor de p, mismo que determina la fiabilidad del estudio y a su vez si resulta ser menor a 0.05 se descarta la hipótesis nula, debido a que es ínfimamente probable que esta pueda ser cierta (Molina, 2017).

**Tabla 10-4:** Resultados del test Z

<b>Variables</b>	<b>Resultados</b>
Media	2.057
$\sigma$	0.5796
ES	0.0295
N	385
Z	-2899.7786
P	0

**Realizado por:** Chávez, M.; Oña, I. 2021

Como se puede observar en la **Tabla 10-4**, se obtuvo que el valor de p es cero, y al ser este menor a 0.05 se descarta la hipótesis nula ( $H_0$ ) que plantea que “No existe diferencia en los tiempos de retraso con y sin la plataforma Sentinel” por lo expuesto en el párrafo inmediatamente anterior; De forma adicional, se puede corroborar lo anterior por medio de la interpretación del valor obtenido de Z a través de su ubicación en la distribución normal para el nivel de confianza del 95% manejado en este experimento, por medio de su representación gráfica en la campana de Gauss.

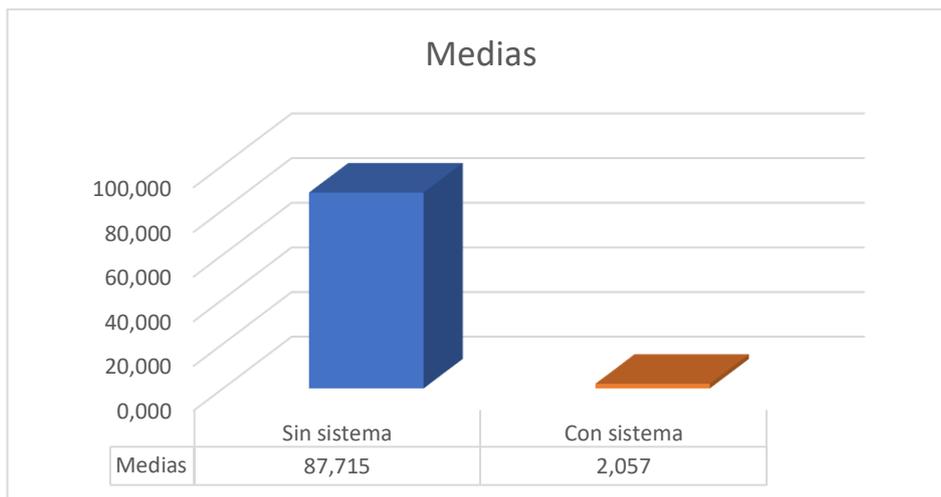


**Gráfico 6-4:** Representación de la distribución normal

Realizado por: Chávez, M.; Oña, I. 2021

Dado que el valor de Z obtenido no se encuentra dentro de los intervalos de confianza  $[-1.96$  y  $1.96]$  como se puede apreciar en el **Gráfico 6-4**, se puede corroborar que la hipótesis nula ( $H_0$ ) no se puede establecer como válida y en cambio se puede aceptar la hipótesis alterna ( $H_1$ ).

Una vez que se comprueba la aceptación de la hipótesis alterna ( $H_1$ ) se llega a la conclusión de que el tiempo de retraso se modificó con el uso de la plataforma Sentinel como se observa en el **Gráfico 7-4**.



**Gráfico 7-4:** Comparación de medias de tiempo de retraso en la ejecución

Realizado por: Chávez, M.; Oña, I. 2021

Finalmente, en base al análisis del **Gráfico 7-4** es posible establecer que la modificación de los tiempos de retraso en la ejecución de los procesos con el uso de la plataforma Sentinel responden a una reducción promedio de 85 segundos con 658 milisegundos con respecto al tiempo de retraso en la ejecución de estos sin la plataforma Sentinel, demostrando así un incremento en la eficiencia del 97.65% en la ejecución de estos procesos.

## CONCLUSIONES

- Con la plataforma denominada Sentinel se redujo en promedio 85 segundos y 658 milisegundos el tiempo de retraso en la ejecución de los procesos de adecuación de la información presentados por la empresa Tracklink.
- Una vez conocidos los procesos de adecuación de la información de la empresa Tracklink se determinó principalmente que estos son repetitivos, que se ejecutan en horarios establecidos y que presentan una media de 87 segundos y 715 milisegundos de retraso en su ejecución.
- Los worker services fueron apropiados para las calendarizaciones de procesos gracias a que pueden ejecutar tareas programadas en segundo plano, sin la necesidad de supervisión, de forma ininterrumpida.
- Se desarrolló una aplicación web que permite calendarizar procesos y un worker service con el que se automatizó la ejecución de estos observándose un tiempo de retraso promedio de 2 segundos y 57 milisegundos.
- Se determinó que la principal causa de demora en la ejecución de los procesos sin la plataforma Sentinel, responde a los 177 segundos con 142 milisegundos en promedio que toma acceder al ambiente de ejecución, mismo que representa un 38.99% del total del tiempo de retraso.
- Una vez analizados estadísticamente los datos obtenidos, en base a la subcaracterística de comportamiento temporal de la característica de eficiencia de desempeño de la norma ISO/IEC 25010, se concluye que con la implementación de la plataforma Sentinel se disminuyen los tiempos de retraso en un 97.65% aproximadamente, con un 95% de confiabilidad y un margen de error del 5%.

## RECOMENDACIONES

- Se recomienda mantener documentados todos los procedimientos, tanto actuales como futuros, que se lleven a cabo en la empresa para facilitar el acceso a estos y en caso de que requieran ser automatizados, se encuentren disponibles.
- Dado que la plataforma Sentinel actualmente se ejecuta en un sistema operativo Windows se recomienda determinar la diferencia en rendimiento entre los Worker services frente a los Windows services.
- Debido a que los procesos que se calendarizan en la plataforma Sentinel pueden ser procedimientos almacenados o scripts, se recomienda realizar un estudio comparativo entre sus tiempos de ejecución en la base de datos.
- Se recomienda continuar con la ejecución de los procesos de adecuación de la información en un horario donde la carga del servidor sea baja para que los tiempos de retraso no se incrementen.
- Para futuros estudios se recomienda analizar la eficiencia de la plataforma tomando en cuenta otras subcaracterísticas dadas por la ISO/IEC 25010 como la utilización de recursos.

## GLOSARIO

**Big Data:** Son datos que contienen una mayor variedad y que se presentan en volúmenes crecientes y a mayor velocidad. (Oracle, 2021)

**CSS:** De las siglas en inglés **Cascading Style Sheets** (Hojas de Estilo en Cascada), es un lenguaje declarativo que controla el aspecto de las páginas web en el navegador. (Mozilla, 2005)

**Daemon:** Un demonio es un tipo de programa en sistemas operativos Unix que se ejecuta en segundo plano, en lugar de estar bajo el control directo de un usuario, esperando ser activado por la ocurrencia de un evento o condición específica. (Mckelvey, 2018)

**FTP:** (File Transfer Protocol) es un protocolo de transferencia de archivos entre sistemas conectados a una red TCP basado en la arquitectura cliente-servidor, de manera que desde un equipo cliente nos podemos conectar a un servidor para descargar archivos desde él o para enviarle nuestros propios archivos independientemente del sistema operativo utilizado en cada equipo. (Networktech Net Services, 2021)

**HTML:** (Lenguaje de Marcas de Hipertexto, del inglés HyperText Markup Language) es el componente más básico de la Web. Define el significado y la estructura del contenido web. (Mozilla, 2005)

**IDE:** Un entorno de desarrollo integrado (IDE) o entorno de desarrollo interactivo es una aplicación que proporciona facilidades a los programadores para el desarrollo de software. Un IDE normalmente consta de un editor de código, herramientas automatizadas para producir el proyecto y un depurador. (Mozilla, 2005)

**IDS/IPS:** Son sistemas de protección de las comunicaciones que actúan monitorizando el tráfico que entra o sale de una red, pero cada uno tiene unas características que les confieren ventajas e inconvenientes. (Incibe, 2020)

**LOC:** Las líneas de medidas de código son las medidas más tradicionales utilizadas para cuantificar la complejidad del software. (Verifysoft, 2020)

**PHP:** Es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML. (PHP, 2020)

**Procedimiento almacenado (SP):** Un procedimiento almacenado es un conjunto de instrucciones de T-SQL que SQL Server compila, en un único plan de ejecución, los llamados "stored procedures" se encuentran almacenados en la base de datos, los cuales pueden ser ejecutados en cualquier momento. (USMP, 2020)

**Product Backlog:** El Product Backlog o pila de producto en un proyecto que sigue la metodología Scrum consiste en una lista con todos los requerimientos iniciales del producto que se va a desarrollar. Se trata de una lista dinámica, que irá evolucionando a medida que lo hace el producto y el entorno del proyecto. La finalidad de crear esta lista no es otra que identificar las necesidades del producto para lograr su máxima utilidad. (EALDE, 2019)

**Script:** En informática, un script, secuencia de comandos o guion (traduciendo desde inglés) es un término informal que se usa para designar a un programa relativamente simple. Los scripts regularmente no se compilan con anticipación a código máquina, sino que son ejecutados por un intérprete que lee el archivo de código fuente al momento; o incluso por una consola interactiva donde el usuario suministra el programa al intérprete paso a paso. (USMP, 2020)

**SCRUM:** Es una metodología de trabajo iterativa e incremental para la gestión de proyectos, desplegado principalmente en el desarrollo ágil de software. La metodología scrum hace hincapié en el software funcional, la flexibilidad para cambiar junto con las realidades de negocio emergentes como páginas web, la comunicación y la colaboración. (Arimetrics, 2020)

**Sprint:** El núcleo central de la metodología de trabajo 'scrum' es el 'sprint'. Se trata de un mini proyecto de no más de un mes (ciclos de ejecución muy cortos -entre una y cuatro semanas), cuyo objetivo es conseguir un incremento de valor en el producto que estamos construyendo. Todo 'sprint' cuenta con una definición y una planificación que ayudará a lograr las metas marcadas. (BBVA, 2019)

**Sprint Backlog:** Este concepto consiste en una lista de elementos seleccionados previamente del Product Backlog, para ser desarrollados en el día a día en los diferentes Sprints del proyecto. (EALDE, 2019)

**SQL:** Por sus siglas en inglés significa Lenguaje de Consulta Estructurada (Structured Query Language), es un lenguaje de programación diseñado para actualizar, obtener, y calcular información en bases de datos relacionales. (Mozilla, 2005)

**TI:** El término tecnología de la información (TI) fue creado con el propósito de hacer una distinción entre las máquinas de alcance limitado y otras con propósitos más generales.

Se basa en el estudio y desarrollo de sistemas de información como aplicaciones software y hardware de computadoras. (ETAC, 2019)

**UML:** El Lenguaje Unificado de Modelado (UML) fue creado para forjar un lenguaje de modelado visual común y semántica y sintácticamente rico para la arquitectura, el diseño y la implementación de sistemas de software complejos, tanto en estructura como en comportamiento. UML tiene aplicaciones más allá del desarrollo de software, p. ej., en el flujo de procesos en la fabricación. (Ludichart, 2020)

**Windows services:** Los servicios de Microsoft Windows, anteriormente conocidos como servicios NT, le permiten crear aplicaciones de larga ejecución que se ejecutan en sesiones propias de Windows. Estos servicios se pueden iniciar automáticamente al arrancar el equipo, se pueden pausar y reiniciar, y no muestran ninguna interfaz de usuario. (Microsoft, 2017)

**Worker service:** Son la evolución directa de los Windows services, con la compatibilidad para ejecutarse en entornos Windows, Linux y macOS. (McKelvey, 2018)

## BIBLIOGRAFÍA

**ALAM, Irtija; & RAMCHAND, Sonam.** *Computational Intelligence in Software Quality Assurance*. Singapur: World Scientific Publishing Company, 2021, pp. 30-32.

**ALMEIDA, Zaira, & CÁRDENAS, Roxana.** Estudio y Automatización del Proceso de Control y Monitoreo de Elaboración de Tesis de Estudiantes de la Carrera de Ingeniería de Sistemas (Trabajo de titulación) (Grado). Universidad Politécnica Salesiana. Guayaquil – Ecuador. 2013. pp. 39-44.

**ARIMETRICS.** *Qué es SCRUM* [en línea]. 2020. [Consulta: 26 diciembre 2020]. Disponible en: <https://www.arimetrics.com/glosario-digital/scrum>

**BBVA.** *Metodología 'scrum': ¿Qué es un 'sprint'?* [en línea]. 2019. [Consulta: 10 enero 2021]. Disponible en: <https://www.arimetrics.com/glosario-digital/scrum>

**CAZAÑAS GORDÓN, Alex.; & PARRA MORA, Esther.** “El impacto de la automatización en el mejoramiento de procesos”. *SATHIRI: Sembrador*, n° 12 (2017), (Ecuador) pp. 249-266.

**CHANG, Argenis; et al.** Automatización de los procesos de registro y procesamiento de información para la gestión de microcréditos del banco comunitario “Mujeres Emprendedoras” de la organización afroecuatoriana “Mujeres Progresistas” del barrio Nigeria-Isla Trinitaria/Guayaquil. (Trabajo de titulación) (Grado). Universidad Politécnica Salesiana. Guayaquil – Ecuador. 2014. pp. 13-18.

**CHÁVEZ, Mario; & OÑA, Israel.** *Manual técnico de la plataforma Sentinel*. 2020.

**CONDRON, Glen.** *NET Core Workers as Windows Services*. [blog]. [Consulta: 23 noviembre 2020]. Disponible en: <https://devblogs.microsoft.com/aspnet/net-core-workers-as-windows-services/>

**DEINUM, Marten; & COSMINA, Iuliana.** *Pro Spring MVC with WebFlux: Web Development in Spring Framework 5 and Spring Boot 2*. Nueva York: Apress, 2021, pp. 123-129.

**EALDE.** *En qué consiste el Product Backlog y el Sprint Backlog* [en línea]. 2019. [Consulta: 27 diciembre 2020]. Disponible en: <https://www.ealde.es/product-backlog-sprint-backlog/>

**ETAC.** *¿Qué es ti?: conoce este concepto y sus alcances* [en línea]. 2019. [Consulta: 27 diciembre 2020]. Disponible en: <https://www.ealde.es/product-backlog-sprint-backlog/>

**MENDLING, Jan; et al.** *Fundamentals of Business Process Management*. Heidelberg: Springer, 2018, pp. 44.

**GIRETTI, Anthony.** *Building a Windows service with Worker Services and .NET Core 3.1, part 1: Introduction*. [blog]. [Consulta: 23 noviembre 2020]. Disponible en: <https://anthonygiretti.com/2020/01/02/building-a-windows-service-with-worker-services-and-net-core-3-1-part-1-introduction/>.

**GORDON, Steve.** *What are .NET Worker Services*. [blog]. [Consulta: 23 noviembre 2020]. Disponible en: <https://www.stevejgordon.co.uk/what-are-dotnet-worker-services>.

**HAMIDIAN, Benito; & OSPINO, Germán.** *¿Por qué los sistemas de información son esenciales?* [blog]. [Consulta: 23 noviembre 2020]. Disponible en: <http://servicio.bc.uc.edu.ve/derecho/revista/idc38>

**HAOUES, Mariem; et al.** “A guideline for software architecture selection based on ISO 25010 quality related characteristics”. *Int J Syst Assur Eng Manag*, n° 8 (2016), (Estados Unidos) pp. 886-909.

**HERRERA, Nelson; et al.** Adaptación de un módulo que garantice la alta disponibilidad del IDS/IPS snort. (Trabajo de titulación) (Grado). Escuela Superior Politécnica del Litoral. Guayaquil – Ecuador. 2012. pp. 31-33.

**HOVORUSHCHENKO, Tetiana.** “Methodology of Evaluating the Sufficiency of Information for Software Quality Assessment According to ISO 25010”. *JIOS*, n° 42 (2018), (Croacia) pp. 63-85.

**INCIBE.** *¿Qué son y para qué sirven los SIEM, IDS e IPS?* [blog]. 2020. [Consulta: 6 febrero 2021]. Disponible en: <https://www.incibe.es/protege-tu-empresa/blog/son-y-sirven-los-siem-ids-e-ips>

**ISO 25010.** División para el modelo de calidad.

**KHENNOUR, Mohammed; & DEBBAH, Youcef.** Enterprise Web Application Architectures. (Trabajo de titulación) (Grado). Kasdi Merbah University. Ouargla– Argelia. 2017. pp. 9-14.

**LAMŽA, Aleksander; et al.** *Scalable and Flexible Web Application Architectures*. Heidelberg: Springer, 2015, pp 22-25.

**LIPAJ, Dmitrij.; & DAVIDAVICIENE, Vida.** “Influence of information systems on business performance”. *Business in XXI century*, vol. 5, n° 1 (2013), (Lituania) pp. 38-45.

**LUDICHART.** *Qué es el lenguaje unificado de modelado (UML)* [en línea]. 2020. [Consulta: 16 diciembre 2020]. Disponible en: <https://www.lucidchart.com/pages/es/que-es-el-lenguaje-unificado-de-modelado-uml>

**MARTIN, Robert.** *Clean Architecture: A Craftsman's Guide to Software Structure and Design*. Estados Unidos: Pearson, 2017, pp. 84-86.

**MAYANK, Manas.; & GARG, Mohit.** *Daemon Application Authentication*. Estados Unidos: Apress, 2019, pp 137-142.

**MCCLANAHAN, Craig.** *The Evolution of Web Application Architectures*. Estados Unidos: Apress, 2021, pp 244-258.

**MCKELVEY, Fenwick.** *Internet Daemons: Digital Communications Possessed*. Estados Unidos: University of Minesota Press, 2018, pp 107-118.

**MICROSOFT.** *Introducción a las aplicaciones de servicios de Windows* [en línea]. 2017. [Consulta: 6 diciembre 2020]. Disponible en: <https://docs.microsoft.com/es-es/dotnet/framework/windows-services/introduction-to-windows-service-applications>

**MOLINA, M.** “¿Qué significa realmente el valor de p?”. *Pediatría atención primaria* [en línea], 2017, (España) 19 (76), pp. 377-381. [Consulta: 4 febrero 2021]. ISSN 2174-4106. Disponible en: <https://pap.es/articulo/12589/que-significa-realmente-el-valor-de-p>

**MOZILLA.** *Glosario* [en línea]. 2005. [Consulta: 20 enero 2021]. Disponible en: <https://developer.mozilla.org/es/docs/Glossary>

**MUSTARI, Lamada; et al.** Pengujian Aplikasi Sistem Monitoring Perkuliahan Menggunakan Standar ISO 25010. *Jurnal MediaTIK* [en línea], 2021, (Indonesia) 3(3), pp. 1-7 [Consulta: 8 febrero 2021]. ISSN 2656-1247. Disponible en <https://ojs.unm.ac.id/mediaTIK/article/view/15172>

**NETWORKTECH NET SERVICES.** *¿Qué es FTP?* [en línea]. 2021. [Consulta: 23 enero 2021]. Disponible en: <https://goc.networktech.com.ar/index.php?rp=/knowledgebase/8/Glosario>

**NOLAN, Tristan; et al.** “Business Process Automation: Automating the Analysis of Anomaly Data”. *International Journal of Simulation*, 1, n°22 (2021), (Estados Unidos) pp. 1-5.

**NOWACKI, Zbigniew.** *File Duplicator: An Example of Windows Services*. *Zszyty Naukowe Politechniki Lodzkiej*. [blog]. [Consulta: 18 noviembre 2020]. Disponible en: [http://repozytorium.p.lodz.pl/bitstream/handle/11652/418/File\\_duplicator\\_for\\_the\\_Nowacki\\_2011.pdf?sequence=1&isAllowed=y](http://repozytorium.p.lodz.pl/bitstream/handle/11652/418/File_duplicator_for_the_Nowacki_2011.pdf?sequence=1&isAllowed=y)

**ORACLE.** *¿Qué es el big data?* [en línea]. 2021. [Consulta: 7 enero 2021]. Disponible en: <https://www.oracle.com/es/big-data/what-is-big-data/>

**PATEL, Shrikant; et al.** “Software Quality Assurance and its Requirements”. *Solid State Technology*, 2, n°60 (2020), (Estados Unidos) pp.12. 17.

**PETERS, Emmanuel.; & AGGREY, George. 2020.** “An ISO 25010 Based Quality Model for ERP Systems”. *Advances in Science, Technology and Engineering Systems Journal*, 2 , n°5 (2020), (Ghana) pp. 578-583.

**PHP.** *¿Qué es PHP?* [en línea]. 2020. [Consulta: 5 diciembre 2020]. Disponible en: <https://www.php.net/manual/es/intro-what-is.php>

**RAE.** *Diccionario de la lengua española* (23ª ed.).

**SCHEPPLER, Björn. & WEBER, Christian.** *Robotic Process Automation*. Heidelberg: Springer, 2020, pp. 152-156.

**STRAUSS, Dirk. 2021.** *Creating ASP.NET Core Web Applications: Proven Approaches to Application Design and Development*. España: Apress, 2021, pp. 257-284.

**UNDEY, Cenk.** “AI in Process Automation”. *SLAS TECHNOLOGY: Translating Life Sciences Innovation*. vol. 26, n° 1 (2021), (Estado Unidos) pp. 1-2.

**USMP.** *¿Qué son procedimientos almacenados?* [en línea]. 2020. [Consulta: 13 enero 2021]. Disponible en: <https://www.usmp.edu.pe/publicaciones/boletin/fia/info41/procedimiento.html>

**VASUJA, Richa; et al.** “Daemons of Hadoop: An Overview”. *International Journal of Engineering Research & Technology (IJERT)*, 7, n° 4 (2018), (India) pp. 545-548.

**VERIFYSOFT.** *Measurement of Lines of Code-Metrics with Testwell CMT++ and Testwell CMTJava* [en línea]. 2020. [Consulta: 27 diciembre 2020]. Disponible en: [https://www.verifysoft.com/en\\_linesofcode\\_metrics.html](https://www.verifysoft.com/en_linesofcode_metrics.html)

**YIK, Ern; et.al.** *Web Applications (Trabajo de titulación) (Maestría)*. Asia Pacific University of Technology and Innovations, Kuala Lumpur - Malasia 2019. pp. 6-32.

**YUNIASRI, Dini; et al.** “Performance Efficiency Evaluation Frameworks Based on ISO 25010”. *2020 10th Electrical Power, Electronics, Communications, Controls and Informatics Seminar (EECCIS)*, n° 10 (2020), (Indonesia) pp. 254-258.

## **ANEXOS**

### **ANEXO A: ENTREVISTA A LA DIRECTORA DE TI DE LA EMPRESA TRACKLINK**

Directora de TI, empresa Tracklink, (entrevistada el 12 de enero de 2021).

**Buenas tardes, gracias por atendernos, procederé con las preguntas**

Bunas tardes, sí adelante

**¿Qué tipo de procesos ejecutan?**

Actualmente solo ejecutamos dos tipos de procesos, scripts y procedimientos almacenados.

**¿Dónde ejecutan estos procesos?**

Los procesos son ejecutados en los servidores de bases de datos de la empresa que están ubicados tanto en Quito como en Guayaquil.

**¿Cómo realizan la conexión a los servidores?**

Dependiendo de la ubicación del servidor, si el servidor está desde donde se envía a ejecutar la conexión es directa caso contrario primero se accede al servidor por una VPN y se conecta al servidor.

**¿Dónde mantienen la ubicación de los Scripts?**

Los scripts se manejan en el mismo servidor o se accede a un servidor FTP y se los ubica dentro de este.

**¿Los procesos que se ejecutan necesitan algún tipo de adecuación?**

Si generalmente un script necesita cambiarse ciertos parámetros para las consultas y de igual forma en los procesos almacenados ingresar algún nuevo parámetro como cantidades que es más usual.

**¿Guardan los resultados de scripts que son tipo consulta?**

Si guardamos los resultados de las consultas en un archivo Excel.

**¿Dónde guardan el archivo generado?**

Generalmente en el mismo servidor donde se hizo la consulta o en algún FTP

**¿En el caso de falla al ejecutar algún proceso que acción toman?**

Se revisa el error que dio la ejecución y se re ejecuta el proceso

**¿La persona encargada de ejecutar el proceso debe quedarse mirándolo por así decirlo hasta que el proceso se ejecute correctamente?**

Si, la persona debe realizar la acción hasta que el proceso se ejecute de manera correcta

**¿Considera usted que el flujo que manejan actualmente genera tiempos de retraso en la ejecución de los procesos?**

Si, porque hay muchas acciones que son repetitivas como conectarse a los servidores y en el caso de fallos re ejecutar el proceso, entre acciones siempre hay segundos q hacen la diferencia.

**Bueno por nuestra parte no tenemos más preguntas, agradecemos su tiempo**

De nada, igualmente muchas gracias, cualquier inquietud estoy aquí para ayudarles.

**ANEXO B: ENCUESTA MODELO A LOS RESPONSABLES DE LA EJECUCIÓN DE LOS PROCESOS DE ADECUACIÓN DE LA INFORMACIÓN DE LA EMPRESA TRACKLINK**

## Encuesta de evaluación de procesos de adecuación de la información

La presente encuesta busca identificar los tiempos involucrados en la ejecución de los procesos de adecuación de la información que actualmente son ejecutados de forma manual por el personal responsable.

1. ¿Cuántos procesos de tipo script ejecuta a la semana?

Tu respuesta \_\_\_\_\_

2. ¿Cuántos procesos de tipo procedimiento almacenado ejecuta a la semana?

Tu respuesta \_\_\_\_\_

3. ¿Cuánto tiempo aproximado en segundos se demora en preparar todos los procesos de tipo script de la semana para ejecutarlos?

Tu respuesta \_\_\_\_\_

4. ¿Cuánto tiempo aproximado en segundos se demora en preparar todos los procesos de tipo procedimiento almacenado de la semana para ejecutarlos?

Tu respuesta \_\_\_\_\_

5. ¿Cuánto tiempo aproximado en segundos a la semana se demora en acceder al ambiente para ejecutar los procesos?

Tu respuesta \_\_\_\_\_

**Enviar**

Nunca envíes contraseñas a través de Formularios de Google.

Este contenido no ha sido creado ni aprobado por Google. [Notificar uso inadecuado](#) - [Términos del Servicio](#) - [Política de Privacidad](#)

Google Formularios

**ANEXO C: RESULTADOS DE LAS ENCUESTAS A LOS RESPONSABLES DE LA  
EJECUCIÓN DE LOS PROCESOS DE ADECUACIÓN DE LA INFORMACIÓN  
DE LA EMPRESA TRACKLINK**

<b>N°</b>	<b>Pregunta</b>	<b>Respuesta</b>
1	¿Cuántos procesos de tipo script ejecuta a la semana?	2 scripts
2	¿Cuántos procesos de tipo procedimiento almacenado ejecuta a la semana?	1 procedimientos almacenados
3	¿Cuánto tiempo aproximado en segundos se demora en preparar todos los procesos de tipo script para ejecutarlos?	120 segundos
4	¿Cuánto tiempo aproximado en segundos se demora en preparar todos los procesos de tipo procedimiento almacenado para ejecutarlos?	120 segundos
5	¿Cuánto tiempo aproximado en segundos a la semana se demora en acceder al ambiente para ejecutar los procesos?	90 segundos

<b>N°</b>	<b>Pregunta</b>	<b>Respuesta</b>
1	¿Cuántos procesos de tipo script ejecuta a la semana?	3 scripts
2	¿Cuántos procesos de tipo procedimiento almacenado ejecuta a la semana?	2 procedimientos almacenados
3	¿Cuánto tiempo aproximado en segundos se demora en preparar todos los procesos de tipo script para ejecutarlos?	150 segundos
4	¿Cuánto tiempo aproximado en segundos se demora en preparar todos los procesos de tipo procedimiento almacenado para ejecutarlos?	150 segundos
5	¿Cuánto tiempo aproximado en segundos a la semana se demora en acceder al ambiente para ejecutar los procesos?	240 segundos

<b>N°</b>	<b>Pregunta</b>	<b>Respuesta</b>
1	¿Cuántos procesos de tipo script ejecuta a la semana?	4 scripts
2	¿Cuántos procesos de tipo procedimiento almacenado ejecuta a la semana?	1 procedimientos almacenados
3	¿Cuánto tiempo aproximado en segundos se demora en preparar todos los procesos de tipo script para ejecutarlos?	150 segundos
4	¿Cuánto tiempo aproximado en segundos se demora en preparar todos los procesos de tipo procedimiento almacenado para ejecutarlos?	30 segundos
5	¿Cuánto tiempo aproximado en segundos a la semana se demora en acceder al ambiente para ejecutar los procesos?	200 segundos

<b>N°</b>	<b>Pregunta</b>	<b>Respuesta</b>
1	¿Cuántos procesos de tipo script ejecuta a la semana?	4 scripts
2	¿Cuántos procesos de tipo procedimiento almacenado ejecuta a la semana?	3 procedimientos almacenados
3	¿Cuánto tiempo aproximado en segundos se demora en preparar todos los procesos de tipo script para ejecutarlos?	180 segundos
4	¿Cuánto tiempo aproximado en segundos se demora en preparar todos los procesos de tipo procedimiento almacenado para ejecutarlos?	150 segundos
5	¿Cuánto tiempo aproximado en segundos a la semana se demora en acceder al ambiente para ejecutar los procesos?	300 segundos

<b>N°</b>	<b>Pregunta</b>	<b>Respuesta</b>
1	¿Cuántos procesos de tipo script ejecuta a la semana?	3 scripts
2	¿Cuántos procesos de tipo procedimiento almacenado ejecuta a la semana?	3 procedimientos almacenados
3	¿Cuánto tiempo aproximado en segundos se demora en preparar todos los procesos de tipo script para ejecutarlos?	120 segundos
4	¿Cuánto tiempo aproximado en segundos se demora en preparar todos los procesos de tipo procedimiento almacenado para ejecutarlos?	180 segundos
5	¿Cuánto tiempo aproximado en segundos a la semana se demora en acceder al ambiente para ejecutar los procesos?	200 segundos

<b>N°</b>	<b>Pregunta</b>	<b>Respuesta</b>
1	¿Cuántos procesos de tipo script ejecuta a la semana?	3 script
2	¿Cuántos procesos de tipo procedimiento almacenado ejecuta a la semana?	1 procedimientos almacenados
3	¿Cuánto tiempo aproximado en segundos se demora en preparar todos los procesos de tipo script para ejecutarlos?	120 segundos
4	¿Cuánto tiempo aproximado en segundos se demora en preparar todos los procesos de tipo procedimiento almacenado para ejecutarlos?	60 segundos
5	¿Cuánto tiempo aproximado en segundos a la semana se demora en acceder al ambiente para ejecutar los procesos?	120 segundos

<b>N°</b>	<b>Pregunta</b>	<b>Respuesta</b>
1	¿Cuántos procesos de tipo script ejecuta a la semana?	3 script
2	¿Cuántos procesos de tipo procedimiento almacenado ejecuta a la semana?	2 procedimiento almacenado
3	¿Cuánto tiempo aproximado en segundos se demora en preparar todos los procesos de tipo script para ejecutarlos?	180 segundos
4	¿Cuánto tiempo aproximado en segundos se demora en preparar todos los procesos de tipo procedimiento almacenado para ejecutarlos?	120 segundos
5	¿Cuánto tiempo aproximado en segundos a la semana se demora en acceder al ambiente para ejecutar los procesos?	90 segundos

**ANEXO D: MEDICIÓN DE TIEMPOS DE RETRASO EN LA EJECUCIÓN DE LOS PROCESOS DE ADECUACIÓN DE LA INFORMACIÓN DE LA EMPRESA TRACKLINK EN LA PLATAFORMA SENTINEL**

**Lunes**

<b>Proceso</b>	<b>Fecha</b>	<b>Calendarización</b>	<b>Ejecución</b>	<b>Diferencia</b>
2	2021-03-15	23:00:00:000	23:00:2:320	1.382
1	2021-02-15	23:00:00:000	23:00:1:657	1.657
3	2021-01-25	23:00:00:000	23:00:1:462	1.462
2	2021-03-29	23:00:00:000	23:00:2:568	2.568
3	2021-04-26	23:00:00:000	23:00:1:349	1.349
3	2021-01-18	23:00:00:000	23:00:1:769	1.769
2	2021-01-11	23:00:00:000	23:00:2:334	2.334
5	2021-06-07	23:00:00:000	23:00:2:640	2.64
5	2021-02-22	23:00:00:000	23:00:2:325	2.325
2	2021-02-15	23:00:00:000	23:00:2:527	2.527
4	2021-01-11	23:00:00:000	23:00:1:280	1.28
4	2021-05-03	23:00:00:000	23:00:1:478	1.478
2	2021-06-07	23:00:00:000	23:00:2:146	2.146
1	2021-01-11	23:00:00:000	23:00:1:482	1.482
2	2021-02-01	23:00:00:000	23:00:2:383	2.383
2	2021-02-08	23:00:00:000	23:00:2:273	2.273
1	2021-03-15	23:00:00:000	23:00:1:656	1.656
2	2021-05-17	23:00:00:000	23:00:2:408	2.408
3	2021-05-10	23:00:00:000	23:00:1:313	1.313
5	2021-01-25	23:00:00:000	23:00:2:649	2.649
4	2021-06-14	23:00:00:000	23:00:1:614	1.614
5	2021-03-08	23:00:00:000	23:00:2:207	2.207
1	2021-01-25	23:00:00:000	23:00:1:724	1.724
4	2021-05-17	23:00:00:000	23:00:1:363	1.363
2	2021-02-22	23:00:00:000	23:00:2:593	2.593
1	2021-03-01	23:00:00:000	23:00:1:641	1.641
4	2021-03-08	23:00:00:000	23:00:1:410	1.41
4	2021-03-01	23:00:00:000	23:00:1:285	1.285
1	2021-04-05	23:00:00:000	23:00:1:701	1.701
5	2021-05-10	23:00:00:000	23:00:2:287	2.287
4	2021-05-24	23:00:00:000	23:00:1:267	1.267
3	2021-02-22	23:00:00:000	23:00:1:708	1.708
3	2021-04-05	23:00:00:000	23:00:1:686	1.686
4	2021-04-12	23:00:00:000	23:00:1:461	1.461
3	2021-05-03	23:00:00:000	23:00:1:502	1.502
1	2021-03-08	23:00:00:000	23:00:1:518	1.518
3	2021-02-01	23:00:00:000	23:00:1:694	1.694
5	2021-05-31	23:00:00:000	23:00:2:521	2.521
1	2021-04-26	23:00:00:000	23:00:1:819	1.819
5	2021-03-22	23:00:00:000	23:00:2:387	2.387

5	2021-05-03	23:00:00:000	23:00:2:203	2.203
1	2021-05-10	23:00:00:000	23:00:1:533	1.533
5	2021-02-01	23:00:00:000	23:00:2:532	2.532
4	2021-03-29	23:00:00:000	23:00:1:242	1.242
3	2021-03-22	23:00:00:000	23:00:1:355	1.355
2	2021-04-05	23:00:00:000	23:00:2:293	2.293
1	2021-05-17	23:00:00:000	23:00:1:445	1.445
2	2021-03-01	23:00:00:000	23:00:2:416	2.416
4	2021-02-01	23:00:00:000	23:00:1:175	1.175
3	2021-05-24	23:00:00:000	23:00:1:401	1.401
4	2021-01-25	23:00:00:000	23:00:1:548	1.548
4	2021-02-15	23:00:00:000	23:00:1:351	1.351
1	2021-01-18	23:00:00:000	23:00:1:615	1.615
1	2021-05-31	23:00:00:000	23:00:1:695	1.695
3	2021-04-19	23:00:00:000	23:00:1:661	1.661

**Martes**

<b>Proceso</b>	<b>Fecha</b>	<b>Calendarización</b>	<b>Ejecución</b>	<b>Diferencia</b>
1	2021-02-02	23:00:00:000	23:00:1:728	1.728
2	2021-04-20	23:00:00:000	23:00:2:624	2.624
2	2021-02-02	23:00:00:000	23:00:2:434	2.434
1	2021-06-15	23:00:00:000	23:00:1:877	1.877
1	2021-06-08	23:00:00:000	23:00:1:883	1.883
2	2021-01-26	23:00:00:000	23:00:2:353	2.353
3	2021-01-26	23:00:00:000	23:00:1:699	1.699
2	2021-01-19	23:00:00:000	23:00:2:284	2.284
3	2021-05-18	23:00:00:000	23:00:1:779	1.779
2	2021-03-23	23:00:00:000	23:00:2:249	2.249
3	2021-04-13	23:00:00:000	23:00:1:822	1.822
1	2021-04-27	23:00:00:000	23:00:1:715	1.715
3	2021-02-09	23:00:00:000	23:00:1:740	1.74
2	2021-06-15	23:00:00:000	23:00:2:520	2.52
2	2021-05-11	23:00:00:000	23:00:2:639	2.639
2	2021-02-16	23:00:00:000	23:00:2:334	2.334
2	2021-03-09	23:00:00:000	23:00:2:415	2.415
2	2021-03-16	23:00:00:000	23:00:2:398	2.398
2	2021-05-04	23:00:00:000	23:00:2:521	2.521
1	2021-06-01	23:00:00:000	23:00:1:589	1.589
2	2021-06-08	23:00:00:000	23:00:2:407	2.407
1	2021-05-04	23:00:00:000	23:00:1:452	1.452
3	2021-03-16	23:00:00:000	23:00:1:806	1.806
1	2021-05-18	23:00:00:000	23:00:1:627	1.627
1	2021-05-25	23:00:00:000	23:00:1:648	1.648
3	2021-03-23	23:00:00:000	23:00:1:843	1.843
1	2021-01-12	23:00:00:000	23:00:1:876	1.876
2	2021-02-23	23:00:00:000	23:00:2:574	2.574
3	2021-04-06	23:00:00:000	23:00:1:470	1.47
3	2021-01-19	23:00:00:000	23:00:1:579	1.579
3	2021-02-02	23:00:00:000	23:00:1:918	1.918
3	2021-02-23	23:00:00:000	23:00:1:613	1.613
3	2021-04-20	23:00:00:000	23:00:1:817	1.817

**Miércoles**

<b>Proceso</b>	<b>Fecha</b>	<b>Calendarización</b>	<b>Ejecución</b>	<b>Diferencia</b>
1	2021-06-16	23:00:00:000	23:00:1:299	1.299
2	2021-06-16	23:00:00:000	23:00:1:635	1.635
1	2021-03-24	23:00:00:000	23:00:1:295	1.295
3	2021-04-14	23:00:00:000	23:00:1:947	1.947
1	2021-04-14	23:00:00:000	23:00:1:476	1.476
2	2021-04-28	23:00:00:000	23:00:1:824	1.824
4	2021-03-24	23:00:00:000	23:00:2:596	2.596
1	2021-02-17	23:00:00:000	23:00:1:623	1.623
2	2021-01-13	23:00:00:000	23:00:1:643	1.643
2	2021-02-10	23:00:00:000	23:00:1:530	1.53
3	2021-02-03	23:00:00:000	23:00:1:900	1.9
4	2021-04-28	23:00:00:000	23:00:2:482	2.482
4	2021-05-12	23:00:00:000	23:00:2:149	2.149
4	2021-05-26	23:00:00:000	23:00:2:385	2.385
3	2021-03-31	23:00:00:000	23:00:1:536	1.536
3	2021-01-13	23:00:00:000	23:00:1:779	1.779
3	2021-03-03	23:00:00:000	23:00:1:950	1.95
2	2021-05-19	23:00:00:000	23:00:1:604	1.604
2	2021-04-07	23:00:00:000	23:00:1:771	1.771
1	2021-04-28	23:00:00:000	23:00:1:466	1.466
1	2021-02-10	23:00:00:000	23:00:1:356	1.356
1	2021-01-27	23:00:00:000	23:00:1:775	1.775
4	2021-03-31	23:00:00:000	23:00:2:161	2.161
3	2021-01-20	23:00:00:000	23:00:1:709	1.709
3	2021-06-16	23:00:00:000	23:00:1:925	1.925
1	2021-04-07	23:00:00:000	23:00:1:619	1.619
1	2021-06-02	23:00:00:000	23:00:1:425	1.425
4	2021-01-27	23:00:00:000	23:00:2:258	2.258
1	2021-05-05	23:00:00:000	23:00:1:378	1.378
4	2021-03-17	23:00:00:000	23:00:2:304	2.304
2	2021-04-21	23:00:00:000	23:00:1:626	1.626
3	2021-05-26	23:00:00:000	23:00:1:983	1.983
1	2021-01-13	23:00:00:000	23:00:1:486	1.486
2	2021-03-10	23:00:00:000	23:00:1:715	1.715
3	2021-02-10	23:00:00:000	23:00:1:733	1.733
4	2021-05-19	23:00:00:000	23:00:2:426	2.426
2	2021-03-24	23:00:00:000	23:00:1:512	1.512
2	2021-03-31	23:00:00:000	23:00:1:455	1.455
2	2021-03-03	23:00:00:000	23:00:1:422	1.422
4	2021-04-21	23:00:00:000	23:00:2:461	2.461
1	2021-03-31	23:00:00:000	23:00:1:301	1.301
1	2021-05-19	23:00:00:000	23:00:1:656	1.656
3	2021-03-24	23:00:00:000	23:00:1:646	1.646
2	2021-06-09	23:00:00:000	23:00:1:803	1.803

**Jueves**

<b>Proceso</b>	<b>Fecha</b>	<b>Calendarización</b>	<b>Ejecución</b>	<b>Diferencia</b>
2	2021-01-21	23:00:00:000	23:00:2:367	2.367
5	2021-02-11	23:00:00:000	23:00:2:827	2.827
5	2021-05-27	23:00:00:000	23:00:2:909	2.909
6	2021-04-29	23:00:00:000	23:00:2:498	2.498
2	2021-03-18	23:00:00:000	23:00:2:415	2.415
6	2021-01-14	23:00:00:000	23:00:2:714	2.714
2	2021-02-25	23:00:00:000	23:00:2:668	2.668
4	2021-03-11	23:00:00:000	23:00:1:517	1.517
1	2021-03-11	23:00:00:000	23:00:2:368	2.368
1	2021-02-04	23:00:00:000	23:00:2:376	2.376
6	2021-01-21	23:00:00:000	23:00:2:546	2.546
6	2021-01-28	23:00:00:000	23:00:2:867	2.867
6	2021-02-18	23:00:00:000	23:00:2:636	2.636
4	2021-04-22	23:00:00:000	23:00:1:353	1.353
3	2021-03-18	23:00:00:000	23:00:1:612	1.612
4	2021-02-18	23:00:00:000	23:00:1:637	1.637
6	2021-06-10	23:00:00:000	23:00:2:732	2.732
4	2021-04-08	23:00:00:000	23:00:1:374	1.374
5	2021-03-18	23:00:00:000	23:00:2:637	2.637
2	2021-06-03	23:00:00:000	23:00:2:591	2.591
4	2021-05-13	23:00:00:000	23:00:1:459	1.459
1	2021-05-27	23:00:00:000	23:00:2:608	2.608
5	2021-02-18	23:00:00:000	23:00:2:864	2.864
5	2021-05-13	23:00:00:000	23:00:2:608	2.608
3	2021-04-29	23:00:00:000	23:00:1:412	1.412
3	2021-03-04	23:00:00:000	23:00:1:476	1.476
5	2021-03-04	23:00:00:000	23:00:2:663	2.663
3	2021-05-20	23:00:00:000	23:00:1:343	1.343
4	2021-03-25	23:00:00:000	23:00:1:734	1.734
3	2021-01-28	23:00:00:000	23:00:1:722	1.722
4	2021-01-21	23:00:00:000	23:00:1:501	1.501
4	2021-05-06	23:00:00:000	23:00:1:499	1.499
3	2021-02-11	23:00:00:000	23:00:1:650	1.65
2	2021-04-29	23:00:00:000	23:00:2:354	2.354
5	2021-02-04	23:00:00:000	23:00:2:715	2.715
2	2021-03-25	23:00:00:000	23:00:2:490	2.49
3	2021-05-27	23:00:00:000	23:00:1:713	1.713
6	2021-03-04	23:00:00:000	23:00:2:622	2.622
3	2021-02-18	23:00:00:000	23:00:1:744	1.744
6	2021-04-22	23:00:00:000	23:00:2:845	2.845
6	2021-03-18	23:00:00:000	23:00:2:551	2.551
5	2021-03-11	23:00:00:000	23:00:2:612	2.612
3	2021-05-06	23:00:00:000	23:00:1:339	1.339
6	2021-02-11	23:00:00:000	23:00:2:726	2.726

1	2021-02-25	23:00:00:000	23:00:2:382	2.382
5	2021-04-15	23:00:00:000	23:00:2:531	2.531
2	2021-02-04	23:00:00:000	23:00:2:236	2.236
3	2021-01-14	23:00:00:000	23:00:1:724	1.724
2	2021-03-04	23:00:00:000	23:00:2:433	2.433
3	2021-06-17	23:00:00:000	23:00:1:408	1.408
5	2021-06-10	23:00:00:000	23:00:2:825	2.825
1	2021-06-03	23:00:00:000	23:00:2:412	2.412
2	2021-05-13	23:00:00:000	23:00:2:219	2.219
6	2021-05-13	23:00:00:000	23:00:2:784	2.784
6	2021-02-25	23:00:00:000	23:00:2:765	2.765
1	2021-01-28	23:00:00:000	23:00:2:379	2.379
6	2021-05-27	23:00:00:000	23:00:2:544	2.544
6	2021-06-03	23:00:00:000	23:00:2:885	2.885
3	2021-04-08	23:00:00:000	23:00:1:701	1.701
5	2021-04-22	23:00:00:000	23:00:2:854	2.854
1	2021-04-01	23:00:00:000	23:00:2:355	2.355
4	2021-06-10	23:00:00:000	23:00:1:633	1.633
6	2021-04-15	23:00:00:000	23:00:2:866	2.866
4	2021-04-29	23:00:00:000	23:00:1:828	1.828
3	2021-03-25	23:00:00:000	23:00:1:568	1.568
1	2021-06-17	23:00:00:000	23:00:2:359	2.359

**Viernes**

<b>Proceso</b>	<b>Fecha</b>	<b>Calendarización</b>	<b>Ejecución</b>	<b>Diferencia</b>
3	2021-04-23	23:00:00:000	23:00:1:773	1.773
3	2021-01-29	23:00:00:000	23:00:1:952	1.952
5	2021-04-09	23:00:00:000	23:00:2:330	2.33
2	2021-03-26	23:00:00:000	23:00:1:697	1.697
3	2021-02-05	23:00:00:000	23:00:1:476	1.476
2	2021-04-09	23:00:00:000	23:00:1:591	1.591
1	2021-06-11	23:00:00:000	23:00:1:649	1.649
1	2021-03-12	23:00:00:000	23:00:1:541	1.541
4	2021-01-15	23:00:00:000	23:00:1:457	1.457
4	2021-01-29	23:00:00:000	23:00:1:397	1.397
3	2021-01-22	23:00:00:000	23:00:1:676	1.676
4	2021-02-19	23:00:00:000	23:00:1:531	1.531
1	2021-01-22	23:00:00:000	23:00:1:923	1.923
5	2021-02-12	23:00:00:000	23:00:2:180	2.18
2	2021-04-30	23:00:00:000	23:00:1:893	1.893
1	2021-03-05	23:00:00:000	23:00:1:485	1.485
1	2021-04-16	23:00:00:000	23:00:1:845	1.845
1	2021-02-05	23:00:00:000	23:00:1:904	1.904
5	2021-03-19	23:00:00:000	23:00:2:320	2.32
3	2021-03-05	23:00:00:000	23:00:1:549	1.549
2	2021-01-22	23:00:00:000	23:00:1:493	1.493
3	2021-04-09	23:00:00:000	23:00:1:690	1.69
5	2021-04-23	23:00:00:000	23:00:2:299	2.299
5	2021-03-05	23:00:00:000	23:00:2:286	2.286
5	2021-05-07	23:00:00:000	23:00:2:247	2.247
1	2021-02-19	23:00:00:000	23:00:1:699	1.699
3	2021-01-15	23:00:00:000	23:00:1:822	1.822
2	2021-02-12	23:00:00:000	23:00:1:459	1.459
1	2021-04-30	23:00:00:000	23:00:1:738	1.738
4	2021-02-26	23:00:00:000	23:00:1:338	1.338
2	2021-03-05	23:00:00:000	23:00:1:465	1.465
4	2021-03-12	23:00:00:000	23:00:1:561	1.561
4	2021-05-21	23:00:00:000	23:00:1:284	1.284
3	2021-05-28	23:00:00:000	23:00:1:678	1.678
5	2021-02-26	23:00:00:000	23:00:2:318	2.318
5	2021-04-02	23:00:00:000	23:00:2:224	2.224
2	2021-04-16	23:00:00:000	23:00:1:861	1.861
5	2021-04-30	23:00:00:000	23:00:2:377	2.377
3	2021-06-04	23:00:00:000	23:00:1:621	1.621
4	2021-04-23	23:00:00:000	23:00:1:590	1.59
3	2021-03-12	23:00:00:000	23:00:1:917	1.917
2	2021-04-23	23:00:00:000	23:00:1:475	1.475
2	2021-06-18	23:00:00:000	23:00:1:822	1.822
1	2021-04-09	23:00:00:000	23:00:1:931	1.931

2	2021-01-29	23:00:00:000	23:00:1:431	1.431
2	2021-04-02	23:00:00:000	23:00:1:634	1.634
3	2021-04-16	23:00:00:000	23:00:1:610	1.61
4	2021-04-09	23:00:00:000	23:00:1:592	1.592
5	2021-04-16	23:00:00:000	23:00:2:641	2.641
3	2021-02-12	23:00:00:000	23:00:1:809	1.809
1	2021-05-07	23:00:00:000	23:00:1:874	1.874
3	2021-05-14	23:00:00:000	23:00:1:672	1.672
2	2021-03-19	23:00:00:000	23:00:1:771	1.771
5	2021-01-15	23:00:00:000	23:00:2:323	2.323
3	2021-05-07	23:00:00:000	23:00:1:761	1.761

**Sábado**

<b>Proceso</b>	<b>Fecha</b>	<b>Calendarización</b>	<b>Ejecución</b>	<b>Diferencia</b>
4	2021-06-12	23:00:00:000	23:00:3:431	3.431
6	2021-01-30	23:00:00:000	23:00:1:654	1.654
6	2021-02-13	23:00:00:000	23:00:1:564	1.564
5	2021-06-19	23:00:00:000	23:00:2:237	2.237
3	2021-02-06	23:00:00:000	23:00:3:318	3.318
1	2021-02-06	23:00:00:000	23:00:1:691	1.691
8	2021-06-12	23:00:00:000	23:00:2:586	2.586
5	2021-04-03	23:00:00:000	23:00:2:594	2.594
4	2021-05-15	23:00:00:000	23:00:3:592	3.592
1	2021-06-05	23:00:00:000	23:00:1:641	1.641
7	2021-01-23	23:00:00:000	23:00:2:440	2.44
5	2021-06-05	23:00:00:000	23:00:2:678	2.678
1	2021-05-29	23:00:00:000	23:00:1:362	1.362
3	2021-01-30	23:00:00:000	23:00:3:274	3.274
7	2021-06-05	23:00:00:000	23:00:2:335	2.335
1	2021-02-27	23:00:00:000	23:00:1:309	1.309
4	2021-03-13	23:00:00:000	23:00:3:452	3.452
7	2021-03-27	23:00:00:000	23:00:2:693	2.693
6	2021-05-22	23:00:00:000	23:00:1:629	1.629
6	2021-05-08	23:00:00:000	23:00:1:465	1.465
4	2021-04-24	23:00:00:000	23:00:3:387	3.387
8	2021-01-30	23:00:00:000	23:00:2:628	2.628
4	2021-06-05	23:00:00:000	23:00:3:272	3.272
4	2021-01-16	23:00:00:000	23:00:3:323	3.323
6	2021-06-19	23:00:00:000	23:00:1:692	1.692
6	2021-03-13	23:00:00:000	23:00:1:774	1.774
2	2021-03-13	23:00:00:000	23:00:2:691	2.691
1	2021-06-12	23:00:00:000	23:00:1:553	1.553
1	2021-05-15	23:00:00:000	23:00:1:661	1.661
8	2021-05-15	23:00:00:000	23:00:2:540	2.54
6	2021-03-06	23:00:00:000	23:00:1:419	1.419
4	2021-04-03	23:00:00:000	23:00:3:392	3.392
4	2021-04-17	23:00:00:000	23:00:3:537	3.537
4	2021-03-20	23:00:00:000	23:00:3:418	3.418
6	2021-04-17	23:00:00:000	23:00:1:747	1.747
1	2021-05-01	23:00:00:000	23:00:1:620	1.62
6	2021-02-20	23:00:00:000	23:00:1:587	1.587
6	2021-02-06	23:00:00:000	23:00:1:435	1.435
8	2021-03-20	23:00:00:000	23:00:2:686	2.686
8	2021-05-08	23:00:00:000	23:00:2:769	2.769
2	2021-04-17	23:00:00:000	23:00:2:546	2.546
2	2021-06-05	23:00:00:000	23:00:2:767	2.767
1	2021-01-16	23:00:00:000	23:00:1:538	1.538
6	2021-02-27	23:00:00:000	23:00:1:460	1.46

2	2021-03-20	23:00:00:000	23:00:2:892	2.892
7	2021-02-27	23:00:00:000	23:00:2:510	2.51
6	2021-05-01	23:00:00:000	23:00:1:788	1.788
5	2021-01-16	23:00:00:000	23:00:2:338	2.338
3	2021-05-15	23:00:00:000	23:00:3:511	3.511
5	2021-05-01	23:00:00:000	23:00:2:202	2.202
6	2021-01-16	23:00:00:000	23:00:1:869	1.869
8	2021-02-20	23:00:00:000	23:00:2:840	2.84
3	2021-06-05	23:00:00:000	23:00:3:219	3.219
4	2021-02-27	23:00:00:000	23:00:3:503	3.503
1	2021-05-08	23:00:00:000	23:00:1:541	1.541
5	2021-05-08	23:00:00:000	23:00:2:397	2.397
2	2021-02-06	23:00:00:000	23:00:2:806	2.806
2	2021-06-19	23:00:00:000	23:00:2:956	2.956
8	2021-01-23	23:00:00:000	23:00:2:462	2.462
5	2021-04-17	23:00:00:000	23:00:2:660	2.66
7	2021-05-15	23:00:00:000	23:00:2:341	2.341
8	2021-01-16	23:00:00:000	23:00:2:524	2.524
4	2021-04-10	23:00:00:000	23:00:3:546	3.546
4	2021-06-19	23:00:00:000	23:00:3:275	3.275
4	2021-02-06	23:00:00:000	23:00:3:486	3.486
3	2021-02-27	23:00:00:000	23:00:3:280	3.28
2	2021-02-13	23:00:00:000	23:00:2:629	2.629
6	2021-03-27	23:00:00:000	23:00:1:404	1.404
5	2021-03-13	23:00:00:000	23:00:2:421	2.421
7	2021-03-06	23:00:00:000	23:00:2:455	2.455
3	2021-05-01	23:00:00:000	23:00:3:459	3.459
7	2021-03-13	23:00:00:000	23:00:2:796	2.796
7	2021-06-19	23:00:00:000	23:00:2:677	2.677
7	2021-01-30	23:00:00:000	23:00:2:532	2.532
4	2021-05-29	23:00:00:000	23:00:3:613	3.613
2	2021-05-01	23:00:00:000	23:00:2:534	2.534
8	2021-02-13	23:00:00:000	23:00:2:503	2.503
7	2021-05-08	23:00:00:000	23:00:2:830	2.83
2	2021-05-08	23:00:00:000	23:00:2:629	2.629
8	2021-05-22	23:00:00:000	23:00:2:960	2.96
8	2021-04-24	23:00:00:000	23:00:2:773	2.773
5	2021-06-12	23:00:00:000	23:00:2:652	2.652
6	2021-06-05	23:00:00:000	23:00:1:665	1.665
4	2021-03-06	23:00:00:000	23:00:3:650	3.65
3	2021-03-06	23:00:00:000	23:00:3:489	3.489
7	2021-01-16	23:00:00:000	23:00:2:558	2.558
1	2021-03-13	23:00:00:000	23:00:1:420	1.42
1	2021-01-30	23:00:00:000	23:00:1:236	1.236

**Domingo**

<b>Proceso</b>	<b>Fecha</b>	<b>Calendarización</b>	<b>Ejecución</b>	<b>Diferencia</b>
4	2021-06-13	23:00:00:000	23:00:2:379	2.379
1	2021-05-16	23:00:00:000	23:00:1:607	1.607
1	2021-02-21	23:00:00:000	23:00:1:879	1.879
3	2021-02-14	23:00:00:000	23:00:1:581	1.581
3	2021-04-11	23:00:00:000	23:00:1:527	1.527
3	2021-05-02	23:00:00:000	23:00:1:609	1.609
1	2021-01-17	23:00:00:000	23:00:1:383	1.383
1	2021-05-23	23:00:00:000	23:00:1:867	1.867
2	2021-03-21	23:00:00:000	23:00:2:140	2.14
4	2021-05-23	23:00:00:000	23:00:2:449	2.449
1	2021-03-21	23:00:00:000	23:00:1:803	1.803
4	2021-04-18	23:00:00:000	23:00:2:595	2.595
2	2021-06-20	23:00:00:000	23:00:2:138	2.138
2	2021-05-09	23:00:00:000	23:00:2:297	2.297
4	2021-02-21	23:00:00:000	23:00:2:488	2.488
2	2021-05-23	23:00:00:000	23:00:2:583	2.583
3	2021-06-20	23:00:00:000	23:00:1:427	1.427
1	2021-03-14	23:00:00:000	23:00:1:840	1.84
2	2021-01-24	23:00:00:000	23:00:2:260	2.26
4	2021-01-24	23:00:00:000	23:00:2:369	2.369
2	2021-02-28	23:00:00:000	23:00:2:262	2.262
1	2021-03-28	23:00:00:000	23:00:1:431	1.431
1	2021-05-02	23:00:00:000	23:00:1:487	1.487
3	2021-03-14	23:00:00:000	23:00:1:531	1.531
3	2021-06-06	23:00:00:000	23:00:1:477	1.477
3	2021-05-23	23:00:00:000	23:00:1:639	1.639
2	2021-04-25	23:00:00:000	23:00:2:524	2.524
3	2021-04-25	23:00:00:000	23:00:1:509	1.509
3	2021-02-07	23:00:00:000	23:00:1:502	1.502
4	2021-05-09	23:00:00:000	23:00:2:250	2.25
3	2021-05-16	23:00:00:000	23:00:1:465	1.465
3	2021-01-31	23:00:00:000	23:00:1:401	1.401
3	2021-02-28	23:00:00:000	23:00:1:622	1.622
3	2021-01-24	23:00:00:000	23:00:1:595	1.595
2	2021-05-16	23:00:00:000	23:00:2:514	2.514
3	2021-04-18	23:00:00:000	23:00:1:525	1.525
3	2021-02-21	23:00:00:000	23:00:1:452	1.452
1	2021-06-13	23:00:00:000	23:00:1:621	1.621
1	2021-06-20	23:00:00:000	23:00:1:410	1.41
2	2021-06-06	23:00:00:000	23:00:2:571	2.571
1	2021-02-14	23:00:00:000	23:00:1:406	1.406
1	2021-05-09	23:00:00:000	23:00:1:690	1.69
3	2021-03-21	23:00:00:000	23:00:1:396	1.396
2	2021-04-11	23:00:00:000	23:00:2:176	2.176

# ANEXO E: CERTIFICADO DE ENTREGA DE MANUALES Y DE LA PLATAFORMA SENTINEL

## TRACKLINK

Control y seguridad para tu tranquilidad

### CERTIFICADO DE CUMPLIMIENTO

En mi calidad de Gerente de Innovación de la empresa Tracklink,

### CERTIFICO

Que ISRAEL EDUARDO OÑA ORDOÑEZ con número de cédula 060378465-3 y MARIO DAVID CHÁVEZ JARAMILLO con número de cédula 060331149-9, han implementado la plataforma denominada "Sentinel" en nuestra empresa Tracklink, misma que permite ejecutar de forma automática los procesos de adecuación de nuestra información, para el respaldo de esta adjuntaron los documentos "Manual técnico de la plataforma Sentinel" y "Manual de usuario de la plataforma Sentinel", por lo cual han cumplido con total satisfacción la entrega del proyecto y a su vez han beneficiado en gran medida a esta dependencia.

Es todo cuanto puedo certificar en honor a la verdad, se les autoriza para que usen el presente certificado de la forma que más les convenga.

Quito, 01 agosto del 2021

Atentamente,

**Audioauto S.A.**  
*Edgar Toapanta*  
FIRMA AUTORIZADA

Edgar Toapanta  
Gerente de Innovación  
Tracklink

#### SUCURSALES TRACKLINK

Guayaquil (04)8017300	Quito (Gaspar) (02)2976200	Quito (Paris) (02)2976255	CC. Quicentro Sur (02)4000263	Santo Domingo (02)3775317	Loja 0997391545
Ambato (03)2414000	Manta (06)2923017	Machala (07)2989105	Ibarra (08)8003283	Cuenca (07)2870716	



WWW.TRACKLINK.EC



ESCUELA SUPERIOR POLITÉCNICA DE  
CHIMBORAZO

DIRECCIÓN DE BIBLIOTECAS Y RECURSOS DEL  
APRENDIZAJE



UNIDAD DE PROCESOS TÉCNICOS  
REVISIÓN DE NORMAS TÉCNICAS, RESUMEN Y BIBLIOGRAFÍA

Fecha de entrega: 05 / 11 / 2021

<b>INFORMACIÓN DEL AUTOR/A (S)</b>
<b>Nombres – Apellidos:</b> MARIO DAVID CHÁVEZ JARAMILLO ISRAEL EDUARDO OÑA ORDOÑEZ
<b>INFORMACIÓN INSTITUCIONAL</b>
<b>Facultad:</b> INFORMÁTICA Y ELECTRÓNICA
<b>Carrera:</b> INGENIERÍA EN SISTEMAS
<b>Título a optar:</b> INGENIERO EN SISTEMAS INFORMÁTICOS
<b>f. Analista de Biblioteca responsable:</b> Lcdo. Holger Ramos, MSc.



Firmado electrónicamente por:  
**HOLGER GERMAN  
RAMOS UVIDIA**

1825-DBRA-UPT-2021