



Facultad de Ingeniería
Ingeniería de Sistemas e Informática

Programa Especial de Titulación:
**“ANÁLISIS, DESARROLLO E
IMPLEMENTACIÓN DE UN SISTEMA
AGENTE PARA LA TOMA DE DATOS
EN EL ÁREA DE OPERACIONES DE
UNA ENTIDAD FINANCIERA.”**

Autor: Arnold Gutiérrez Contreras

Para optar el Título Profesional de
Ingeniero de Sistemas e Informática

Asesor: Yamil Alexander Quiñones Nieto

Lima - febrero 2020

INDICE DE CONTENIDO

INDICE DE FIGURAS	4
INDICE DE TABLAS	5
INTRODUCCION.....	6
CAPITULO 1.....	7
ASPECTOS GENERALES	7
1.1. Definición del Problema	7
1.1.1. Descripción del Problema.....	7
1.1.2. Formulación del Problema.....	9
1.2. Definición de objetivos	10
1.2.1. Objetivo general	10
1.2.2. Objetivos específicos	10
1.3. Alcances y limitaciones	10
1.3.1. Alcances.....	10
1.3.2. Limitaciones	11
1.4. Justificación	12
CAPITULO 2.....	13
FUNDAMENTO TEÓRICO	13
2.1. Antecedentes	13
2.1.1. Nacional.....	13
2.1.2. Internacional	16
2.2. Marco teórico	17
2.2.1. Sistema Agente	17
2.2.2. Arquitectura REST	18
2.3. Marco metodológico	20
2.3.1. El Marco de Trabajo de SCRUM	20
2.4. Marco conceptual.....	29
CAPITULO 3.....	31
DESARROLLO DE LA SOLUCIÓN.....	31
3.1. INICIACIÓN	33
3.1.1. Creación de acta de constitución del proyecto	33
3.1.2. Identificación de Stakeholders.....	33
3.2. PLANIFICACIÓN.....	34

3.3. IMPLEMENTACION	42
3.4. LANZAMIENTO	70
CAPITULO 4.....	71
RESULTADOS	71
4.1.1. Implementar un sistema agente para reducir el tiempo de registro de las tareas realizadas en el área de operaciones.	71
4.1.2. Implementar un sistema agente para salvaguardar la autenticación en el proceso de registro de las tareas realizadas en el área de operaciones.	74
4.1.3. Implementar un sistema Agente para mantener una trazabilidad de las actividades realizadas por un trabajador determinado en el área de operaciones.	75
4.2. Análisis de costos.....	79
4.2.1. Costos de Personal.....	79
4.2.2. Costos de Hardware.....	79
4.2.3. Costos de Software	80
4.2.4. Costos Totales	80
4.2.5. Costos variables.....	81
4.3. Análisis de Beneficios	82
4.4. Análisis de Flujo de Caja	83
4.4.1. Flujo de caja	83
4.4.2. Calculo del VAN	85
4.4.3. Calculo del TIR	85
CONCLUSIONES	87
RECOMENDACIONES	88
REFERENCIA BIBLIOGRÁFICA.....	89
ANEXOS	90

INDICE DE FIGURAS

Figura 1. Árbol de problemas.....	8
Figura 2. Framework Scrum.....	20
Figura 3. Principios de Scrum.....	22
Figura 4. Procesos de Scrum.....	24
Figura 5. Fases del proyecto.....	32
Figura 6. Cronograma	35
Figura 7. Arquitectura de la aplicación.....	37
Figura 8. Diagrama de Base de datos.....	44
Figura 9. Prueba en Postman.....	45
Figura 10. Lógica de captura de usuario de Windows.....	47
Figura 11. Tabla de usuarios.....	47
Figura 12. Interfaz Selección de Flujo.....	48
Figura 13. Lógica para obtener los flujos de la base de datos.....	48
Figura 14. Lógica para obtener la lista de tareas a realizar.....	49
Figura 15. Lógica para leer las aplicaciones activas del sistema.....	50
Figura 16. Lógica para saber si se ha completado una tarea.....	51
Figura 17. Lógica para registrar las tareas completadas.....	52
Figura 18. Lógica para registrar el tiempo total de la tarea terminada.....	53
Figura 19. Pruebas Unitarias Sprint 1	53
Figura 20. Lógica para contabilizar el total de tareas registradas.....	55
Figura 21. Interfaz para mostrar las notificaciones de tareas terminadas.....	56
Figura 22. Lógica para mostrar notificaciones de tareas terminadas.....	57
Figura 23. Lógica para mostrar como notificación mensaje de no se está siguiendo con el patrón.....	58
Figura 24. Lógica para mostrar como notificación las sub-tareas que continúan.....	59
Figura 25. Lógica de creación de archivo de registro de errores.....	60
Figura 26. Pruebas Unitarias Sprint 2.....	61
Figura 27. Interfaz para el registro tareas manuales.....	63
Figura 28. Lógica para el registro de tareas manuales.....	64
Figura 29. Lógica para el monitoreo de los estados de agentes.....	65
Figura 30. Interfaz para mostrar lista de tareas.....	65
Figura 31. Lógica para mostrar lista de tareas seleccionadas.....	66
Figura 32. Lógica de intercambio de flujos de manera automática.....	67
Figura 33. Lógica para poder deshacer acción.....	68
Figura 34. Pruebas Unitarias Sprint 3.....	69
Figura 35. Lógica autenticación de usuario.....	75
Figura 36. Datos de trazabilidad.....	77
Figura 37. Costos de personal.....	79
Figura 38. Costos de Hardware.....	80
Figura 39. Costos de Software.....	80
Figura 40. Costos Totales.....	81
Figura 41. Costos variables.....	82
Figura 42. Beneficios Tangibles.....	82
Figura 43. Flujo de caja.....	84
Figura 44. Fórmula del VAN.....	85
Figura 45. Fórmula del TIR.....	85

INDICE DE TABLAS

Tabla 1. Árbol de problemas.	8
Tabla 2. Relación entre grupo de procesos y áreas de conocimiento.	28
Tabla 3. Identificación de Stakeholders.	33
Tabla 4. Product Backlog.	34
Tabla 5. Arquitectura de servidores.	36
Tabla 6. Riesgos del proyecto.	38
Tabla 7. Sprint Cero.	42
Tabla 8. Sprint 1.	46
Tabla 9. Sprint 2.	54
Tabla 10. Sprint 3.	62
Tabla 11. Encuesta de tiempo de registro antes del sistema.	72
Tabla 12. Encuesta de tiempo de registro después del sistema.	73
Tabla 13. Encuesta de seguimiento de la información antes del sistema.	76
Tabla 14. Encuesta de seguimiento de la información después del sistema.	78

INTRODUCCION

El presente trabajo se trata del análisis, desarrollo e implementación de un sistema agente en una entidad financiera de tipo bancario, este informe abarca desde la descripción del problema, el planteamiento de la solución, la aplicación de una metodología adecuada y el desarrollo de la solución.

En el primer capítulo se detalla la descripción del problema, se definen los objetivos generales y específicos, alcances limitaciones y la justificación del proyecto.

En el segundo capítulo se refiere al marco teórico, marco conceptual y marco metodológico.

En el tercer capítulo se detalla el desarrollo de la solución aplicando la metodología utilizada.

En el cuarto capítulo se describe la manera de medir los resultados del proyecto y el análisis de costos.

CAPITULO 1

ASPECTOS GENERALES

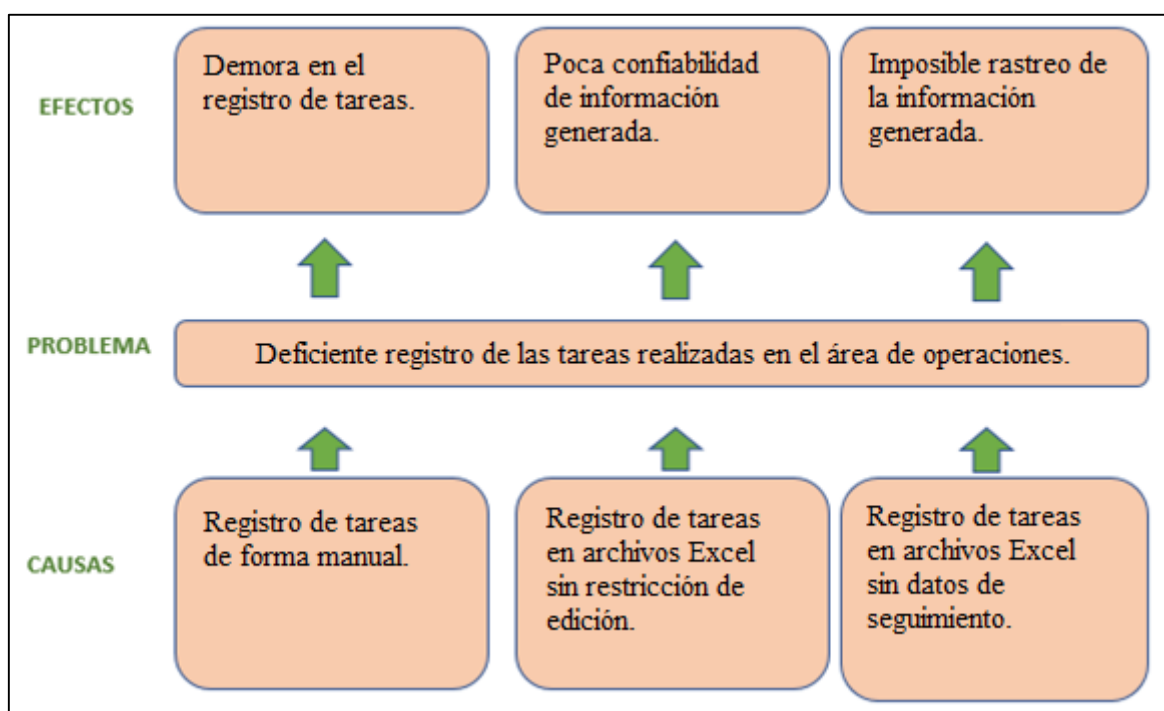
1.1. Definición del Problema

1.1.1. Descripción del Problema

En el 2007 la entidad financiera y una empresa Outsourcing establecieron un contrato que consiste en la externalización del área de operaciones de la entidad financiera, cada personal de la empresa Outsourcing registra sus tareas realizadas del día manualmente en archivos Excel de esta forma se mide el trabajo realizado y se procede con el pago por el servicio.

Esta forma de registrar las tareas es susceptible a errores por ser registro de forma manual por lo tanto no refleja el esfuerzo real ejecutado por el personal de la empresa Outsourcing debido a eso se produce pagos incorrectos y el monitoreo del trabajo no es exacto.

Figura 1. Árbol de problemas.



Fuente: Elaboración propia

Tabla 1. Árbol de problemas.

Problema: Registro de las tareas realizadas en el área de operaciones es deficiente	
CAUSAS	EFECTOS
Registro de las tareas realizadas es de forma manual.	Demora en el registro de las tareas.
Registro de tareas en archivos de Excel sin restricción de edición.	Poca confiabilidad de la información generada.
Registro de tareas en archivos Excel sin datos de seguimiento.	Imposible rastreo de la información generada.

Fuente: Elaboración propia

1.1.2. Formulación del Problema

1.1.2.1. Problema general

¿De qué manera el registro de las tareas realizadas en el área de operaciones dejará de ser deficiente?

1.1.2.2. Problemas específicos

- ¿Cómo hacer para que el registro de las tareas realizadas en el área de operaciones que se hace de forma manual deje de ocasionar demora?
- ¿De qué manera el registro de tareas en archivos de Excel sin restricción de edición deje de ocasionar poca confiabilidad de la información generada?
- ¿Cómo hacer para que el registro de tareas en archivos Excel sin datos de seguimiento ocasione el imposible rastreo de la información generada?

1.2. Definición de objetivos

1.2.1. Objetivo general

Desarrollar un Sistema Agente para la toma de datos en el área de operaciones de una entidad financiera.

1.2.2. Objetivos específicos

- Implementar un sistema agente para reducir el tiempo de registro de las tareas realizadas en el área de operaciones.
- Implementar un sistema agente para salvaguardar la autenticación en el proceso de registro de las tareas realizadas en el área de operaciones.
- Implementar un sistema agente para mantener una trazabilidad de las actividades realizadas por un trabajador determinado en el área de operaciones.

1.3. Alcances y limitaciones

1.3.1. Alcances

- La solución debe realizar el conteo de transacciones y registro de los tiempos de las tareas ejecutadas por el usuario en su PC.
- Para el proyecto se utilizará una metodología ágil, con una duración aproximada de 5 meses.
- La solución será utilizado por los usuarios del área de operaciones de la entidad financiera.
- La solución debe poder leer una matriz de patrones el cual estarán las tareas de manera secuencial de lo que debe realizar en el día el trabajador.

- La solución será construida en Visual Studio y con el lenguaje C#, será una aplicación de tipo winforms.

1.3.2. Limitaciones

- El proveedor no se hace responsable de la pérdida de datos ya sea por un problema de Hardware o Software. Es responsabilidad del cliente la ejecución de los procedimientos de respaldo.
- El proveedor no proporciona servidores para la solución. Es responsabilidad del cliente tener la infraestructura de hardware necesaria a tiempo para no afectar los tiempos del proyecto.
- El proveedor no proporciona el servicio de instalación física de servidores, Cableado, Rack, y otros.

1.4. Justificación

La entidad financiera en el área de operaciones en específico y para el tema de registrar sus tareas diarias por el personal que son utilizadas para justificar el pago del servicio por ser una área manejada por un Outsourcing, la realización de este proceso es de forma manual, es por eso que necesita usar sistemas automatizados que le permita tener una información más exacta, más segura y con un mejor monitoreo de lo que se está haciendo, y no incurrir en gastos en exceso por el pago de los servicios.

Este proyecto tiene como finalidad implementar un sistema agente que le permita hacer el registro de las tareas realizadas por el personal en el área de operaciones de forma automatizada y trabajando en modo transparente al usuario, almacenar estos datos en una base de datos que posteriormente será administrada desde una plataforma web (el desarrollo de la plataforma web no forma parte del informe), esto le permitirá a la entidad financiera tener una información más precisa, seguridad de su información y un mejor control que le ayude a tomar mejores decisiones.

CAPITULO 2

FUNDAMENTO TEÓRICO

2.1. Antecedentes

2.1.1. Nacional

Título

Godoy Vilca, Erik Raziél. (2017) Agente virtual inteligente para la orientación vocacional en el hogar Virgen de Fátima de la ciudad de Puno, Tesis presentada para obtener el Título de Ingeniero Estadístico e Informático. Universidad Nacional del Altiplano - Puno.

Objetivos

- Determinar cómo se interrelacionan la Orientación Vocacional con el Agente Virtual Inteligente.
- Implementar el sistema Agente Virtual Inteligente para la Orientación Vocacional.
- Determinar el efecto del sistema Agente Virtual Inteligente en el servicio, comunicación y control de datos de la orientación vocacional.

Situación problemática

En esta institución se presentan anualmente problemas sobre el entendimiento de las diferentes carreras profesionales que existen, por eso es necesario que las alumnas al salir de la institución tengan una mejor orientación de estos temas y puedan tomar una mejor decisión de lo que quieren hacer con sus vidas realmente.

Metodología

Para el desarrollo del sistema Agente Virtual Inteligente se utilizó HTML5, CSS3, JQuery, el lenguaje de programación JSP y como gestor de base de datos el MySQL 5.5.6.

En cuanto al agente inteligente conversacional se utilizó el AIML (lenguaje de marcas para la inteligencia artificial).

Resultados y conclusiones

- Las actividades del sistema Agente Virtual Inteligente ayuda a obtener una orientación vocacional de calidad.
- La conversación con el sistema Agente Virtual Inteligente y la interrelación de las actividades hacen que las internas tengan muchos materiales multimedia y lograron encontrar su verdadera vocación.
- El sistema Agente Virtual Inteligente implementado logró que en la institución, los temas de orientación vocacional alcance un nivel de calidad óptimo.

Aporte

El aporte de este antecedente al proyecto actual es ver como los sistemas Agentes pueden ser útiles en diferentes sectores y áreas, y otro factor importante su disponibilidad que es algo que caracterizan a este tipo de sistemas por su autonomía no requieren la intervención de una persona.

Título

Harry Miguel Garro Portales. (2016) Desarrollo de un sistema informático aplicando SCRUM para mejorar la gestión académica del Instituto Manuel Banda de Guadalupe - La Libertad, tesis para optar el título profesional de ingeniero, Guadalupe- Perú

Objetivos

- Realizar el análisis y diseño del Sistema Informático.
- Realizar el desarrollo del Sistema Informático.
- Ejecutar las pruebas del Sistema Informático.

Situación problemática

La manera de trabajar del instituto en el área académica es de forma manual eso ocasiona problemas como tareas repetitivas, registros con inconsistencias y pérdida de tiempo en la tarea de registrar la información ocasionando malestar.

Metodología

La metodología utilizada es ágil con el Framework SCRUM debido a que el sistema del proyecto requiere entregas en periodos cortos, cumpliendo con los requerimientos y las necesidades del cliente.

Resultados y conclusiones

- El 83% de los miembros del Instituto se encuentra conforme con diseño de la interfaz y el fácil uso del Sistema Informático
- El 87% afirma que el tiempo de respuesta de bueno.
- El 68% de los miembros del Instituto está Satisfecho con las funcionalidades de Sistema por las grandes ventajas, ya que ofrecen gran velocidad el procesamiento y el manejo de los datos y con ello generar reportes en muy poco tiempo.

Aporte

El aporte de este antecedente es ver y entender la manera de aplicar la metodología ágil con el Framework SCRUM.

2.1.2. Internacional

Título

John Patricio Solano Cabrera. (2015) Desarrollo de un Agente Inteligente para dispositivos móviles Android que permita llevar los ingresos y egresos económicos de una persona, Tesis de Grado previo a la obtención del título de Ingeniería en Sistemas. Universidad Nacional de Loja - ECUADOR.

Objetivos

Implementar una aplicación que ayude a llevar el control de ingresos y egresos económicos de las personas a través de un dispositivo que cuente con el sistema operativo Android y que le permita pronosticar sus gastos diarios, presupuesto y notificaciones mediante alertas.

Situación problemática

El poco control de los gastos de las personas los lleva a una situación de pobreza o un estilo de vida ajustado y el avance de la tecnología y los sistemas ayuda a solucionar estos problemas con sistemas de control de gastos por ejemplo, pero en el sector de dispositivos móviles es escaso este tipo de aplicaciones.

Metodología

Se hace uso de la biblioteca JADE en JAVA, que permite desarrollar sistemas de tipo Agente para dispositivos Android.

Resultados y conclusiones

- La aplicación de la metodología de Programación Extrema permitió documentar las partes importantes que ayudan a determinar las entradas tanto de la aplicación como definir las creencias con las que trabaja el Agente.

- El uso del entorno de desarrollo eclipse permitió el manejo eficaz en la creación de la aplicación para los dispositivos móviles.
- La librería JADE-LEAP de java en Android para la implementación del sistema Agente inteligente permitió agilizar el desarrollo del mismo.

Aporte

El aporte de este antecedente al presente proyecto es ver y entender la importancia de los sistemas Agentes y que existen diferentes librerías para su construcción.

2.2. Marco teórico

2.2.1. Sistema Agente

(Wooldridge. M & Jennings. R, 1994) Según los autores se puede distinguir dos conceptos para el término “Agente” desde el punto de vista informático.

Un concepto general de Agente.

Es un hardware o software que tiene las siguientes propiedades:

- **Autonomía:** Significa que tienen control propio sobre sus acciones sin la intervención de alguna persona.
- **Capacidad Social:** Tiene la habilidad de interactuar con otros agentes y personas.
- **Reactividad:** Tienen la capacidad de percibir su entorno y responder ante cambios en su entorno.
- **Pro actividad:** Un agente no solo tiene que poder responder ante cambios en su entorno si no también debe realizar las tareas para lo cual fue diseñado.

Un concepto avanzado de Agente.

Para algunos investigadores sobre todo los que están relacionados con el mundo de la Inteligencia artificial, le dan al término Agente propiedades más relacionadas con el ser humano como conocimiento, creencia, intención y obligación además de los ya mencionados anteriormente.

Otros atributos de un Agente

Adicionalmente a los atributos ya mencionados también se tiene en cuenta las siguientes propiedades que no son indispensables:

- **Movilidad:** La capacidad de un agente de moverse por una red de computadores en busca de recursos que le permitan continuar con su tarea.
- **Veracidad:** Un agente no comunicará información falsa.
- **Benevolencia:** Un agente llevara a cabo su tarea fielmente.
- **Racionalidad:** Un Agente siempre actuará para lograr sus objetivos.

2.2.2. Arquitectura REST

(Flanders. J, 2009) REST son las siglas de Representational state transfer, es un estilo arquitectónico que se presentó por primera vez en una exposición de Roy Fielding en la Universidad de California Irvine en el año 2000. Presenta las siguientes características:

Recursos y URI

Un servicio REST está formado por recursos que son accedidos a través de URL, si se desea acceder a un recurso se deberá ingresar la URL y como resultado retornará el recurso buscado, la URL no podrá ser de tipo llamadas a archivos ejecutables en el servidor como por ejemplo archivos de tipo jsp, php, asp.

Interfaz uniforme

Los servicios REST utilizan los métodos HTTP (HyperText Transfer Protocol)

estos métodos son los que se mencionan a continuación:

- GET: Retorna la información de un recurso.
- PUT: Modifica un recurso.
- POST: Crea un nuevo recurso.
- DELETE: Elimina un recurso.

Representaciones de los recursos

La forma de representar los recursos es variable, esto se especifica en el tipo de medios (Content-Type) de la respuesta HTTP.

Estos son algunos de los tipos de medios más usados:

- XML (Extensible Markup Language)
- XHTML (Extensible HyperText Markup Language)
- JSON (JavaScript Object Notation)

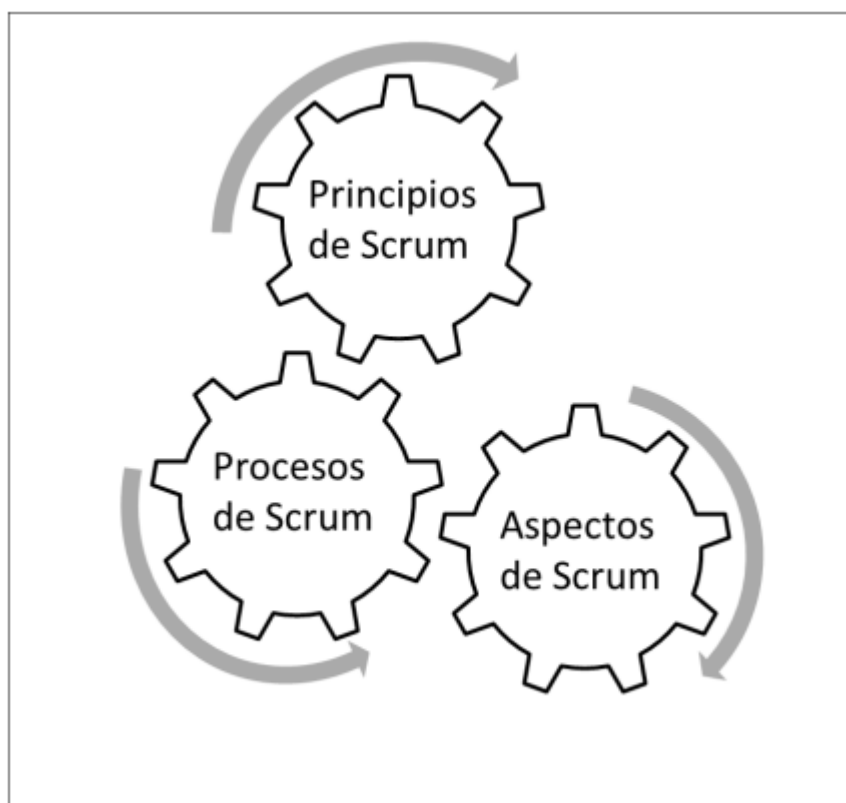
2.3.Marco metodológico

2.3.1. El Marco de Trabajo de SCRUM

(SBOK, 2017) Scrum es un Framework que se usa en equipos de trabajo para desarrollar proyectos de cualquier tipo, basado en una metodología de trabajo ágil, tiene como meta realizar entregables en periodos cortos de tiempo.

La guía SBOK se divide en tres áreas, las cuales se muestran en la siguiente figura.

Figura 2. Framework Scrum.



Fuente: SBOK 2017

a. **PRINCIPIOS.** Los principios de Scrum son los siguientes.

Control del proceso empírico.

Indica que las decisiones se fundamentan en la observación y la experimentación y no en la planificación inicial.

Este principio se basa en tres ideas principales; Transparencia, inspección y adaptación.

- **Transparencia.** Esto permite que todas las actividades y facetas del proceso Scrum sean vistas por cualquiera y crea una cultura de trabajo libre.
- **Inspección.** Permite tener un mejor control del proyecto mediante herramientas como pueden ser; Scrumboard, creación del Backlog priorizado.
- **Adaptación.** Ocurre cuando el equipo aprende mediante la transparencia y la inspección y se logran adaptar durante las mejoras que se realizan en el proyecto.

Auto-organización.

Significa que cada integrante del equipo trabaja de manera ordenada y coordinada para alcanzar los objetivos del proyecto.

Colaboración.

Significa que debe existir una unión en la forma de trabajo entre los miembros del equipo principal y los demás Stakeholders.

Priorización basada en valor.

Scrum utiliza la priorización para poder obtener una parte del producto funcionando durante cada iteración en el menor tiempo posible.

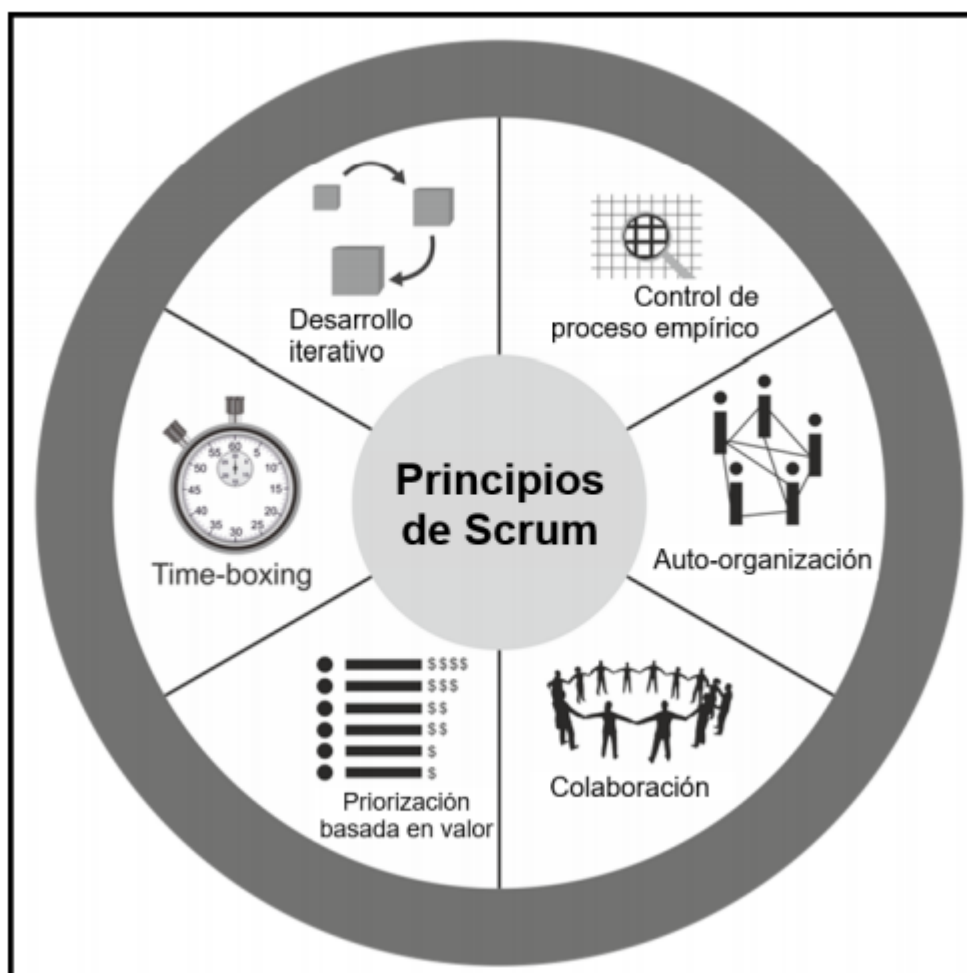
Time-Boxing.

Significa bloqueo de tiempo y en Scrum se utiliza con la finalidad de establecer una cantidad de tiempo a una determinada actividad y así el equipo no ocupe demasiado o muy poco tiempo en la actividad.

Desarrollo Iterativo.

Es una técnica en donde toda la etapa de desarrollo se segmenta en pequeñas etapas completas y cada una de estas teniendo como resultado un producto software funcionando.

Figura 3. Principios de Scrum.



Fuente: SBOK 2017

b. ASPECTOS DE SCRUM

Los cinco aspectos de Scrum deben realizarse y gestionarse durante todo el proyecto.

Organización

Se refiere a los roles y responsabilidades definidos en un proyecto de Scrum. Los roles de Scrum se dividen en dos categorías:

- **Roles Centrales.** Son los que se requieren obligatoriamente que estén en el proyecto. Estos roles son:
 - Product Owner.
 - Scrum Master.
 - Equipo Scrum.
- **Roles no Centrales.** No son obligatorios en el proyecto:
 - Stakeholders

Justificación del negocio. Se refiere a que se debe realizar una evaluación adecuada del negocio antes de la iniciación del proyecto y verificar si es realmente justificable la creación del producto o servicio.

Calidad. Se refiere a que el producto o servicio debe tener la capacidad para cumplir con los criterios de aceptación y de alcanzar el valor de negocio que el cliente espera.

Cambio. Se refiere a que el equipo debe estar preparado para dar una respuesta rápida ante los cambios que aparecen durante la realización de un proyecto.

Riesgo. Se refiere a que se deben identificar los posibles riesgos del proyecto desde el inicio de este y así poder tenerlos identificados y saber qué hacer cuando suceda alguno.

c. PROCESOS DE SCRUM

Los procesos de Scrum son los siguientes agrupados cada uno por la fase que le corresponde.

Figura 4. Procesos de Scrum.

Fase	Procesos fundamentales de Scrum
Inicio	<ol style="list-style-type: none"> 1. Crear la visión del proyecto 2. Identificar al Scrum Master y Stakeholder(s) 3. Formar Equipos Scrum 4. Desarrollar épica(s) 5. Crear el Backlog Priorizado del Producto 6. Realizar la planificación de lanzamiento
Planificación y estimación	<ol style="list-style-type: none"> 7. Crear historias de usuario 8. Estimar historias de usuario 9. Comprometer historias de usuario 10. Identificar tareas 11. Estimar tareas 12. Crear el Sprint Backlog
Implementación	<ol style="list-style-type: none"> 13. Crear entregables 14. Realizar Daily Standup 15. Refinar el Backlog Priorizado del Producto
Revisión y retrospectiva	<ol style="list-style-type: none"> 16. Demostrar y validar el sprint 17. Retrospectiva del sprint
Lanzamiento	<ol style="list-style-type: none"> 18. Enviar entregables 19. Retrospectiva del proyecto

Fuente: SBOK 2017

2.3.2. PMBOK

(PMBOK, 2017) El PMBOK es una guía de buenas prácticas para la dirección de proyectos, es producido por una entidad llamada PMI, no es una metodología pues no es un conjunto de pasos a seguir, es una guía y el director de proyecto debe decidir que partes de la guía aplicarlas en su proyecto. Fue publicado en 2017, se divide en tres partes.

PARTE 1 Guía de los fundamentos para la dirección de proyectos.

Contiene la identificación de las áreas de conocimientos, que son campos de conocimiento que todo director de proyectos debe conocer para aplicarlo a su proyecto como debe de ser.

Áreas de conocimiento.

- **Gestión de la integración.**

Esta área de conocimiento se incluye las actividades para identificar, definir, unificar y coordinar los diferentes procesos de la dirección de proyectos que se aplican desde el inicio del proyecto hasta su término.

- **Gestión del alcance del proyecto.**

En esta área de conocimiento se gestiona todo lo que debe tener y que no debe tener el proyecto.

- **Gestión del cronograma del proyecto.**

En esta área de conocimiento se gestiona todo las actividades requeridas para controlar la finalización del proyecto a tiempo.

- **Gestión de los costos del proyecto.**

En esta área de conocimiento se gestiona todos los costos referentes a los recursos necesarios para completar las actividades del proyecto.

- **Gestión de la calidad del proyecto.**

En esta área de conocimiento se realiza la comparación de los resultados de las actividades (según se va avanzando) con los requisitos de calidad con la finalidad de obtener un resultado aceptable.

- **Gestión de los recursos del proyecto.**

Se encarga de gestionar para que todos los recursos físicos o humanos se encuentren disponibles a lo largo del proyecto.

- **Gestión de las comunicaciones del proyecto.**

Esta parte consiste en desarrollar y llevar a cabo una estrategia para asegurar que la comunicación sea eficaz para los interesados.

- **Gestión de los Riesgos del proyecto.**

Se encarga de aumentar la probabilidad y/o el impacto de los riesgos positivos y disminuir la probabilidad y/o el impacto de los riesgos negativos con la finalidad de optimizar las probabilidades de éxito del proyecto.

- **Gestión de las adquisiciones del proyecto.**

En esta área de conocimiento se gestiona los acuerdos de compras de productos o servicios que se requieran en el proyecto.

- **Gestión de los interesados del proyecto.**

En esta área de conocimiento se identifica a las personas, grupos u organizaciones que pueden afectar o ser afectados por el proyecto.

PARTE 2 Estándar para la dirección de proyectos.

Contiene los grupos de procesos.

- Grupo de procesos de inicio.
- Grupo de procesos de planificación.

- Grupo de procesos de ejecución.
- Grupo de procesos de Monitoreo y control
- Grupo de procesos de Cierre.

PARTE3 Apéndice, glosario e índice.

Contiene el apéndice, glosario y el índice de la guía.

Tabla 2. Relación entre grupo de procesos y áreas de conocimiento.

Áreas de Conocimiento	Grupos de Procesos de la Dirección de Proyectos				
	Grupo de Procesos de Inicio	Grupo de Procesos de Planificación	Grupo de Procesos de Ejecución	Grupo de Procesos de Monitoreo y Control	Grupo de Procesos de Cierre
4. Gestión de la Integración del Proyecto	4.1 Desarrollar el Acta de Constitución del Proyecto	4.2 Desarrollar el Plan para la Dirección del Proyecto	4.3 Dirigir y Gestionar el Trabajo del Proyecto 4.4 Gestionar el Conocimiento del Proyecto	4.5 Monitorear y Controlar el Trabajo del Proyecto 4.6 Realizar el Control Integrado de Cambios	4.7 Cerrar el Proyecto o Fase
5. Gestión del Alcance del Proyecto		5.1 Planificar la Gestión del Alcance 5.2 Recopilar Requisitos 5.3 Definir el Alcance 5.4 Crear la EDT/WBS		5.5 Validar el Alcance 5.6 Controlar el Alcance	
6. Gestión del Cronograma del Proyecto		6.1 Planificar la Gestión del Cronograma 6.2 Definir las Actividades 6.3 Secuenciar las Actividades 6.4 Estimar la Duración de las Actividades 6.5 Desarrollar el Cronograma		6.6 Controlar el Cronograma	
7. Gestión de los Costos del Proyecto		7.1 Planificar la Gestión de los Costos 7.2 Estimar los Costos 7.3 Determinar el Presupuesto		7.4 Controlar los Costos	
8. Gestión de la Calidad del Proyecto		8.1 Planificar la Gestión de la Calidad	8.2 Gestionar la Calidad	8.3 Controlar la Calidad	
9. Gestión de los Recursos del Proyecto		9.1 Planificar la Gestión de Recursos 9.2 Estimar los Recursos de las Actividades	9.3 Adquirir Recursos 9.4 Desarrollar el Equipo 9.5 Dirigir al Equipo	9.6 Controlar los Recursos	
10. Gestión de las Comunicaciones del Proyecto		10.1 Planificar la Gestión de las Comunicaciones	10.2 Gestionar las Comunicaciones	10.3 Monitorear las Comunicaciones	
11. Gestión de los Riesgos del Proyecto		11.1 Planificar la Gestión de los Riesgos 11.2 Identificar los Riesgos 11.3 Realizar el Análisis Cualitativo de Riesgos 11.4 Realizar el Análisis Cuantitativo de Riesgos 11.5 Planificar la Respuesta a los Riesgos	11.6 Implementar la Respuesta a los Riesgos	11.7 Monitorear los Riesgos	
12. Gestión de las Adquisiciones del Proyecto		12.1 Planificar la Gestión de las Adquisiciones	12.2 Efectuar las Adquisiciones	12.3 Controlar las Adquisiciones	
13. Gestión de los Interesados del Proyecto	13.1 Identificar a los Interesados	13.2 Planificar el Involucramiento de los Interesados	13.3 Gestionar la Participación de los Interesados	13.4 Monitorear el Involucramiento de los Interesados	

Fuente: PMBOK

2.4. Marco conceptual

Base de datos-SQL Server

Un sistema gestor de bases de datos (SGBD) consiste en una colección de datos interrelacionados y un conjunto de programas para acceder a dichos datos.

Directorio activo

Es un servicio de Microsoft que se aloja en un servidor y sirve para crear usuarios, grupos y poder administrar los inicios de sesión y políticas en un entorno de red.

Entidad Financiera

Una entidad financiera es un intermediario del mercado financiero. Las entidades financieras pueden ser bancos, cajas de ahorros o cooperativas de crédito, es decir, intermediarios que administran y prestan dinero; o empresas financieras, un tipo distinto de intermediarios financieros que, sin ser bancos, ofrecen préstamos o facilidades de financiamiento en dinero.

Flujos de trabajo

Es una definición utilizada en el proyecto que describirá una categoría mayor a la de los servicios, también es subida a la plataforma web mediante la matriz de servicio que es donde se especifica.

Matriz de patrones

Matriz utilizada en el proyecto para almacenar el detalle de las tareas realizadas en la entidad financiera, estas se enlazarán con una aplicación en específico para que pueda ser reconocida por el agente y se pueda saber en qué tarea se encuentra.

Postman

Es una aplicación que sirve para realizar el testeo de servicios web de tipo REST.

Pruebas Unitarias

Es un tipo de prueba de software que sirve para comprobar que el funcionamiento de una unidad de código sea correcto.

Visual Studio

Es un entorno de desarrollo integrado, permite crear aplicaciones de tipo escritorio, web y servicios web, entre otros, y es compatible con lenguajes como C#, Visual Basic, etc.

CAPITULO 3

DESARROLLO DE LA SOLUCIÓN

En este capítulo se explicará cada uno de los pasos que se realizó para la implementación de la solución, utilizando una metodología basada en el Framework Ágil Scrum para administrar el desarrollo del producto y la guía PMBOK para la gestión del proyecto.

Donde se utilizó PMBOK

PMBOK es utilizado en la fases de iniciación, planificación, monitoreo y cierre. Para gestionar la integración, el alcance, los riesgos, el cronograma, los interesados, la puesta en marcha.

Donde se utilizó Scrum

Scrum es utilizado para la fase de ejecución y monitoreo, en la creación de historias de usuario, tareas, reuniones, incrementos de producto, pruebas.

En el siguiente cuadro tenemos un resumen de cómo se llevó a cabo la metodología en el presente proyecto.

Figura 5. Fases del proyecto.



Fuente: Elaboración propia

3.1.INICIACIÓN

En esta fase se incluyen las actividades relacionadas con el inicio del proyecto, creación del acta de constitución, formación del equipo de trabajo y elaboración del Product Backlog.

3.1.1. Creación de acta de constitución del proyecto

Ver Anexo 1.

3.1.2. Identificación de Stakeholders

A continuación se presenta los interesados del proyecto.

Tabla 3. Identificación de Stakeholders.

Stakeholders			
ROL EN EL PROYECTO	CARGO	DEPARTAMENTO	INSTITUCION
Patrocinador del Proyecto	Gerente de empresa proveedor del sistema.	Tecnología de Información	Proveedor del sistema
Gerente del Proyecto	PMP	Tecnología de Información	Proveedor del sistema
Gerente del área operativa	Gerente del área operativa	Servicios Operativos	Entidad Bancaria
Product Owner	Especialista de Servicios Operativos de entidad bancaria	Servicios Operativos	Entidad Bancaria
SCRUM Master	Especialista en TI	Tecnología de Información	Proveedor del sistema
Programador 1	Especialista en desarrollo de sistemas	Tecnología de Información	Proveedor del sistema
Programador 2	Especialista en integración de sistemas	Tecnología de Información	Proveedor del sistema
Analista de calidad.	Especialista en TI	Tecnología de Información	Proveedor del sistema

Fuente: Elaboración propia

3.2.PLANIFICACIÓN

En esta fase se incluyen las actividades relacionadas con la planificación y estimación de las tareas, los cuales incluyen, definición del alcance, elaborar cronograma, definición de la arquitectura, elaboración de cuadro de riesgos.

3.2.1. Definición del alcance y EDT. A continuación se muestra el Product Backlog, el cual está constituido por las historias de usuario. Esta parte lo define el Product Owner. El EDT se muestra en el Anexo 2.

Tabla 4. Product Backlog.

Código	Historias de usuario	Prioridad
HU01	Como usuario del sistema necesito autenticación automática con usuario del sistema operativo registrado en un directorio activo.	Media
HU02	Como usuario del sistema necesito poder elegir un flujo para poder empezar a trabajar.	Media
HU03	Como usuario del sistema, necesito poder leer las tareas en un orden siguiendo un patrón.	Alta
HU04	Como Usuario del sistema, necesito registrar las tareas que realizo en el día.	
HU05	Como usuario del sistema, necesito registrar el tiempo que me toma en hacer cada tarea.	Alta
HU06	Como usuario del sistema, necesito contabilizar el total de tareas terminadas.	Alta
HU07	Como administrador del sistema, necesito poder visualizar notificaciones de tareas terminadas	Alta
HU08	Como usuario del sistema, necesito saber si no se está siguiendo con el patrón.	Alta
HU09	Como usuario del sistema, necesito saber cuál es la siguiente sub-tarea que continua.	Alta
HU10	Como administrador del sistema necesito ver un registro de errores	Media
HU11	Como usuario del sistema necesito poder registrar las tareas manuales	Media
HU12	Como administrador del sistema necesito poder monitorear el estado de los agentes	Media
HU13	Como usuario del sistema necesito ver la lista de tareas del flujo que he seleccionado y en la que estoy trabajando.	Media
HU14	Como usuario del sistema necesito poder salir y poder cambiar entre flujos de manera automática.	Media
HU15	Como usuario del sistema necesito poder retroceder a un paso anterior por si me equivoco	Media

Fuente: Elaboración propia

3.2.2. Elaboración del cronograma del proyecto. A continuación se muestra el cronograma del proyecto, el cual está categorizado por las fases y cada una con sus tareas, en la fase de implementación se puede ver los Sprints que lo conforman.

Figura 6. Cronograma
Desde 07/01/2019 Hasta 27/05/2019

Nombre de tarea	Duración	Comienzo	Fin
INICIO			
Creación de acta de constitución del proyecto	1 día	lun 07/01/19	lun 07/01/19
Formación de equipo de trabajo	1 día	mar 08/01/19	mar 08/01/19
Elaboración de Product Backlog	4 días	mié 09/01/19	lun 14/01/19
PLANIFICACIÓN			
Elaboración de Sprint Backlog	4 días	mar 15/01/19	vie 18/01/19
Elaboración de cronograma de proyecto	5 días	lun 21/01/19	vie 25/01/19
Elaborar arquitectura del proyecto	3 días	lun 28/01/19	mié 30/01/19
Elaborar cuadro de riesgos	2 días	jue 31/01/19	vie 01/02/19
IMPLEMENTACIÓN Y CONFIGURACIÓN			
Sprint 0	14 días		
Planificación	1 día	lun 04/02/19	lun 04/02/19
Desarrollo	9 días	mar 05/02/19	vie 15/02/19
Pruebas	3 días	lun 18/02/19	mié 20/02/19
Sprint retrospective	1 día	jue 21/02/19	jue 21/02/19
Sprint 1	20 días		
Planificación	1 día	vie 22/02/19	vie 22/02/19
Desarrollo	15 días	lun 25/02/19	vie 15/03/19
Pruebas	3 días	lun 18/03/19	mié 20/03/19
Sprint retrospective	1 día	jue 21/03/19	jue 21/03/19
Sprint 2	20 días		
Planificación	1 día	vie 22/03/19	vie 22/03/19
Desarrollo	15 días	lun 25/03/19	vie 12/04/19
Pruebas	3 días	lun 15/04/19	mié 17/04/19
Sprint retrospective	1 día	jue 18/04/19	jue 18/04/19
Sprint 3	20 días		
Planificación	1 día	vie 19/04/19	vie 19/04/19
Desarrollo	15 días	lun 22/04/19	vie 10/05/19
Pruebas	3 días	lun 13/05/19	mié 15/05/19
Sprint retrospective	1 día	jue 16/05/19	jue 16/05/19
CIERRE	7 días		
Manual de usuario	5 días	vie 17/05/19	jue 23/05/19
Acta de cierre del proyecto	2 días	vie 24/05/19	lun 27/05/19

Fuente: Elaboración propia

3.2.3. Arquitectura de la solución. La solución se ejecutara desde las pc de los operadores tomara datos de estas y la información será enviada a una base de datos (a través de servicios web) para luego ser explotada por una aplicación web (No está contemplada su desarrollo en el presente proyecto).

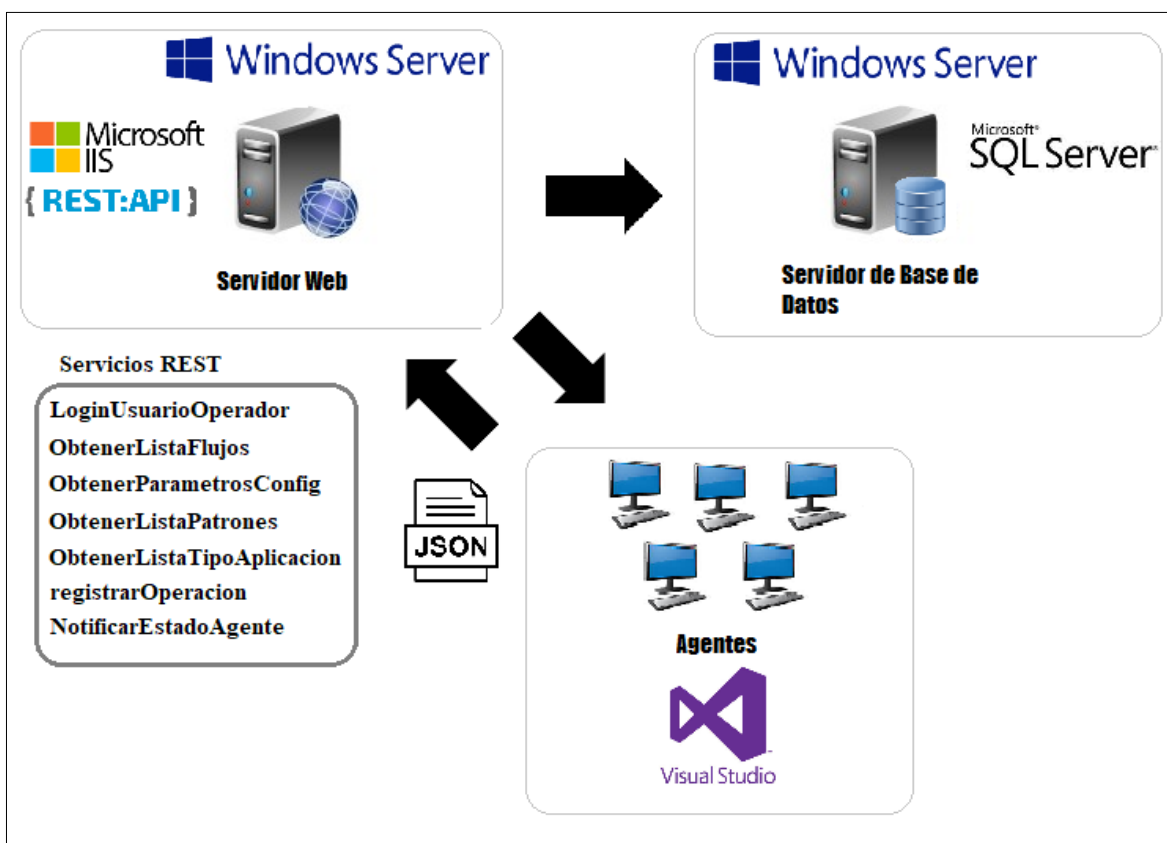
La arquitectura propuesta para los entornos de desarrollo y producción es la siguiente:

Tabla 5. Arquitectura de servidores.

SERVIDOR DE BASE DE DATOS	
Sistema Operativo	Windows Server 2016
Memoria	32 GB
Almacenamiento	2 × 2TB HDD
Procesador	2-3 GHz, 4 núcleos
Aplicaciones	SQL Server 2016
SERVIDOR DE APLICACIONES	
Sistema Operativo	Windows Server 2016
Memoria	32 GB
Almacenamiento	2 × 2TB HDD
Procesador	2-3 GHz, 4 núcleos
Aplicaciones	IIS
PCS OPERARIOS (AGENTE)	
Sistema Operativo	Windows 10
Memoria	4 GB
Almacenamiento	1 TB HDD
Procesador	2 GHz
Aplicaciones	Office, Outlook, etc.

Fuente: Elaboración propia

Figura 7. Arquitectura de la aplicación.



Fuente: Elaboración propia

3.2.4. Riesgos del proyecto. A continuación se muestra los riesgos del proyecto, indicando la probabilidad que ocurra, el impacto, y el plan de acción a tomar.

Tabla 6. Riesgos del proyecto.

N°	RIESGO IDENTIFICADO	PROBABILIDAD	IMPACTO	PLAN ACCION
1	Retraso en la implementación por parte del proveedor.	Media	Media	Se solicitarán más recursos a trabajar en paralelo en ambos casos. Recuperación de horas.
2	Retraso en la entrega de entornos desarrollo, QA y producción.	Alto	Alto	El proveedor realizará el despliegue en su ambiente de desarrollo.
3	Dependencia con la entrega de información de parte del cliente, de los flujos de trabajo (procedimiento y aplicaciones informáticas)	Alto	Alto	Simulación con algunos datos de prueba.
4	Disponibilidad de Líder y usuarios para validación de información de flujos de trabajo.	Medio	Medio	Involucramiento del área de producción desde el inicio del proyecto y estimación de fechas de atención.

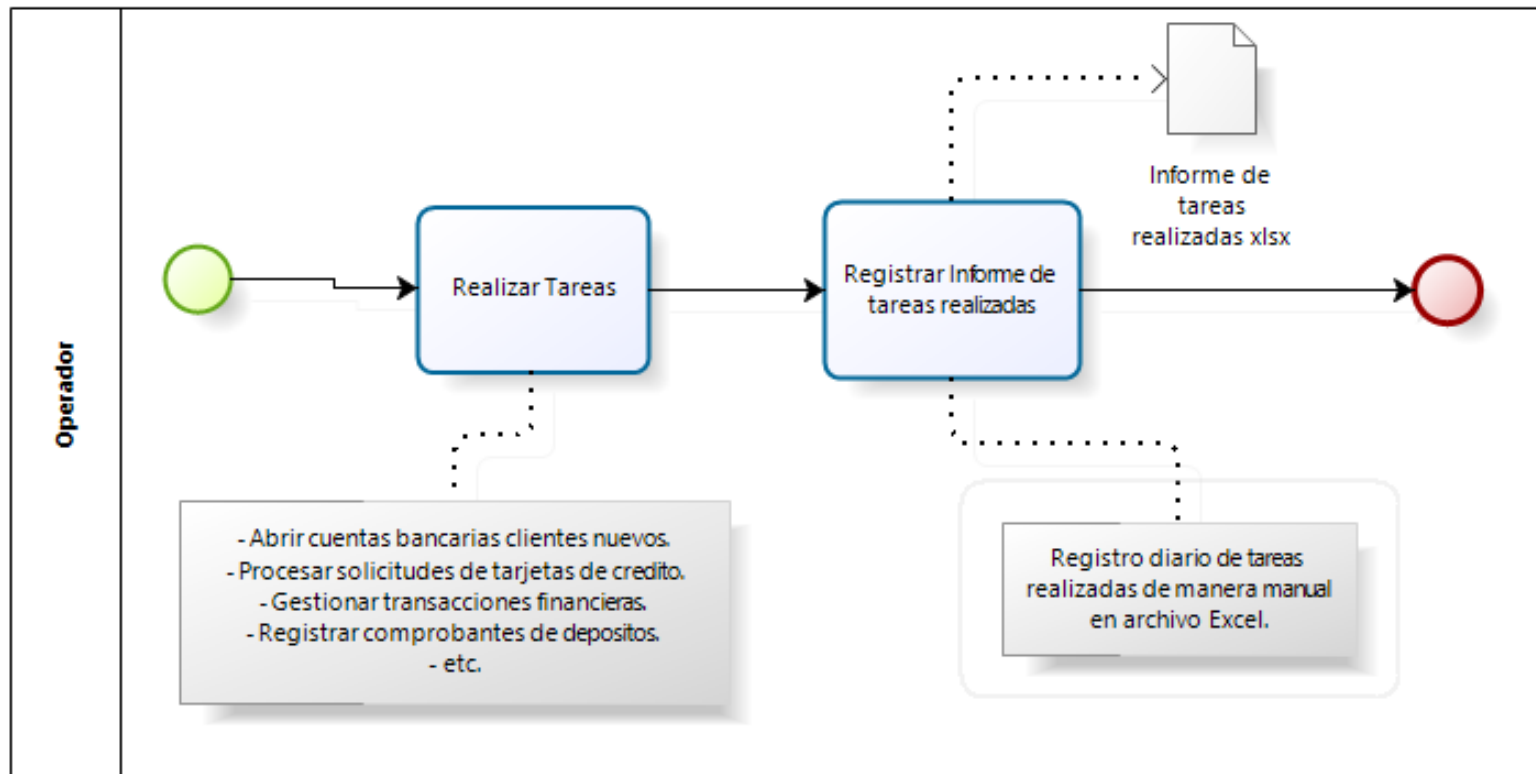
Fuente: Elaboración propia

3.2.5. Identificación de procesos.

Proceso actual manual.

Aquí se identifica el proceso actual manual.

Figura 8. Modelamiento de negocio ASIS

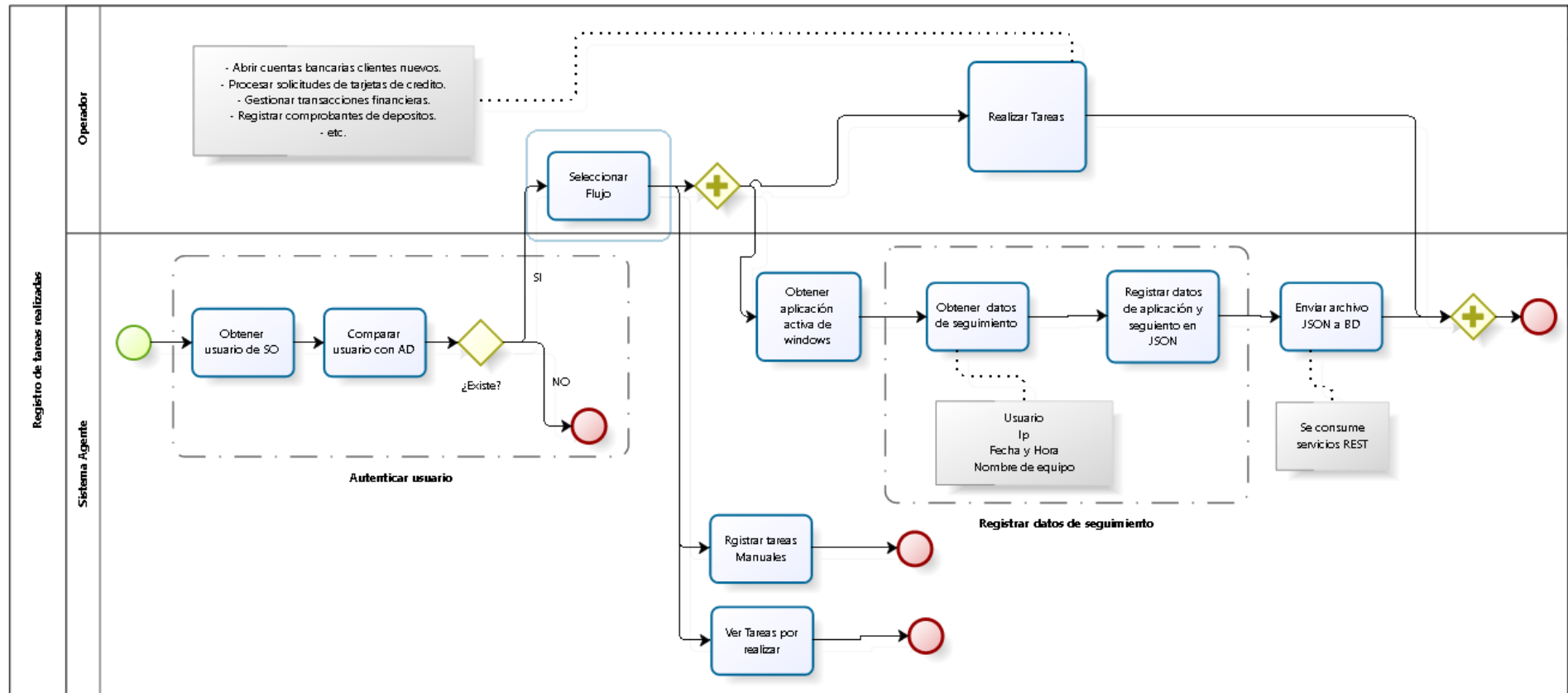


Fuente: Elaboración propia

Proceso automatizado.

Luego de analizar los requerimientos se plantea una solución en términos de proceso de negocio.

Figura 9. Modelamiento de negocio TOBE



Fuente: Elaboración propia

Diagrama de Casos de uso:

En este diagrama podemos apreciar la interacción entre el usuario y el sistema.

Figura 10. Diagrama de casos de uso



Fuente: Elaboración propia

3.3. IMPLEMENTACION

En esta fase se desarrollarán a detalle los tres Sprint definidos.

3.3.1. Sprint cero. A continuación se muestra a detalle el desarrollo del Sprint cero.

a. Planificación. En la etapa de planificación del Sprint se define las tareas a realizar y el tiempo que tomará para su desarrollo.

Tabla 7. Sprint Cero.

Código	Tareas	Estimación	Sprint
SP01	Creación de Base de datos, análisis y diseño.	14 días	0
SP02	Creación de servicios Web.		

Fuente: Elaboración propia

b. Ejecución del sprint. En esta etapa se desarrollarán las tareas especificadas en el Sprint cero.

SP01 Análisis y diseño de la Base de datos. En esta parte se toma en cuenta los requerimientos del cliente, y se analiza y diseña la base de datos, en el siguiente grafico se puede observar cómo está diseñado y las tablas:

Procedimiento de diseño.

- Organizar la información recopilada.
- Dividir la información en tablas.
- Crear las columnas.
- Especificar claves principales.
- Relacionar las tablas.
- Verificar si está bien estructurado.
- Realizar diccionario de datos. Ver Anexo 3.

Tablas creadas.

TipoAplicacion: Se encarga de almacenar los diferentes tipos de aplicación con lo que trabajará el sistema Agente.

Pe_mtzpatrones: Se encarga de almacenar las sub-tareas del sistema.

Pe_frmmtzservicios: Se encarga de almacenar las tareas del sistema.

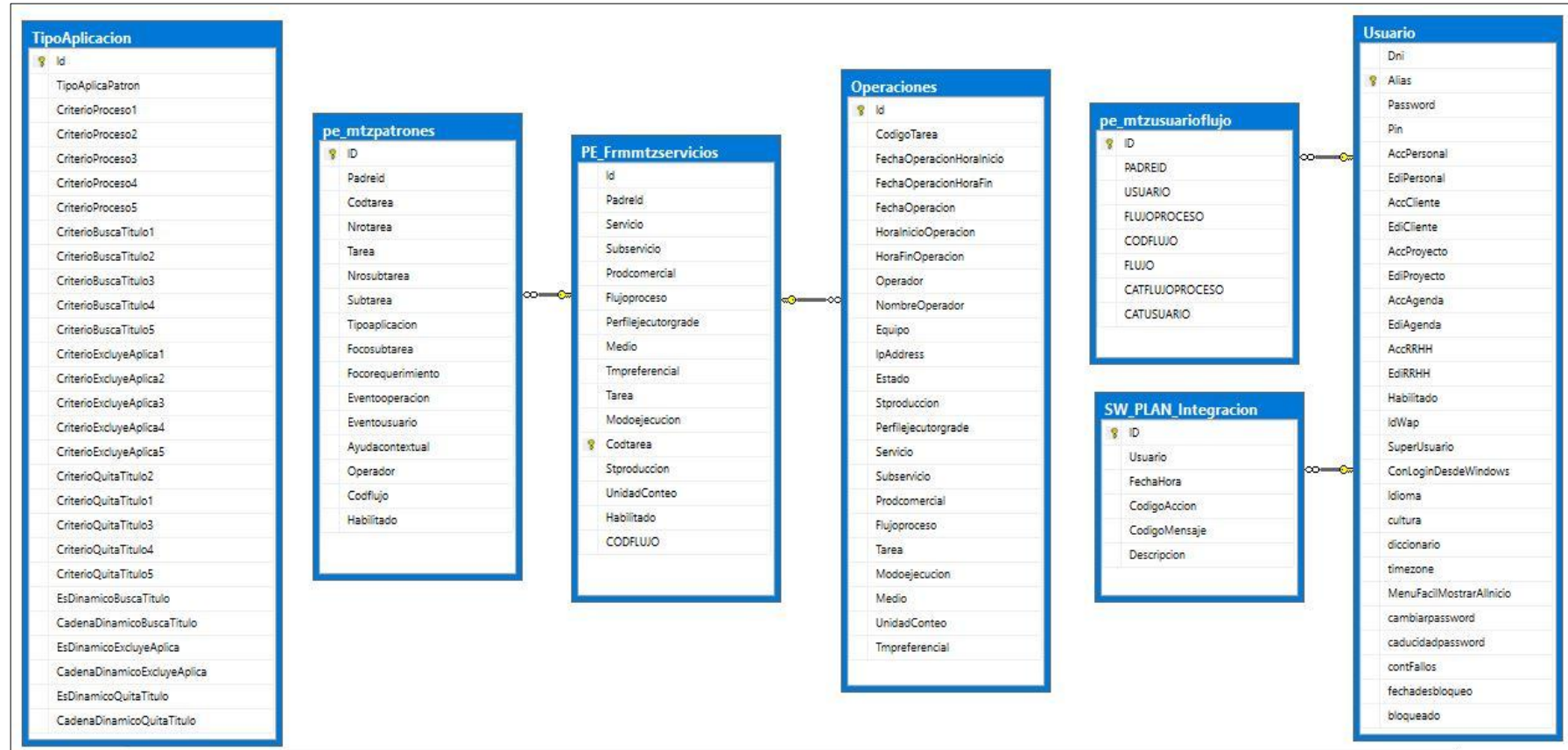
Operaciones: Se encarga de almacenar los registros de las operaciones.

Pe_mtzusuarioflujo: Se encarga de almacenar la relación de usuario con el flujo.

Sw_plan_integracion: Se encarga de almacenar los registros de originados cuando se realiza el llamado a cualquiera de los servicios.

Usuario: Se encarga de almacenar a los usuarios que tienen acceso al sistema.

Figura 11. Diagrama de Base de datos.



Fuente: Elaboración propia

- **SP02 Creación de servicios Web.** En esta parte se realiza la creación de los servicios Web de tipo REST.

LoginUsuarioOperador

ObtenerListaFlujos

ObtenerParametrosConfig

ObtenerListaPatrones

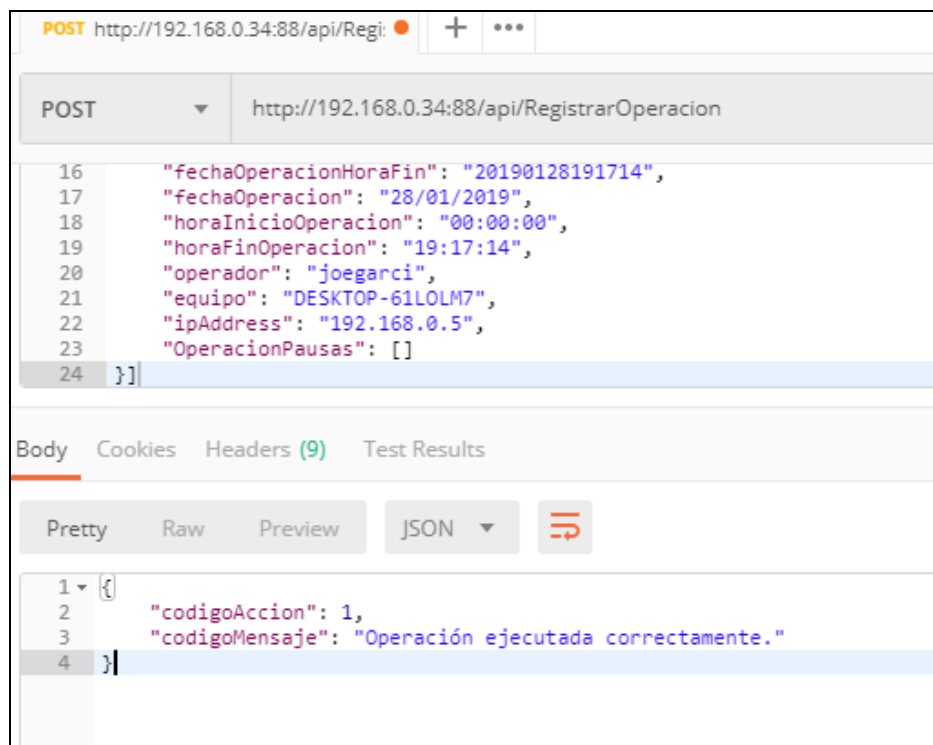
ObtenerListaTipoAplicacion

registrarOperacion

NotificarEstadoAgente

- c. **Pruebas.** El equipo realiza las pruebas funcionales según las tareas del sprint. Para las pruebas se utiliza la aplicación Postman.

Figura 12. Prueba en Postman



Fuente: Elaboración propia

- d. Sprint Review Meeting.** El equipo se reúne para hacer una demostración de lo que se ha logrado realizar y sobre las nuevas características de la aplicación, los participantes son; el equipo de desarrollo, Scrum Master, Product Owner y usuarios del módulo en cuestión. Para las actas de reunión ver Anexo 4.
- e. Sprint Retrospective.** El equipo se reúne y presenta problemas o inconvenientes ocurridos en el desarrollo del Sprint y se propone opciones de soluciones o mejoras. Para las actas de reunión ver Anexo 4.

3.3.2. Primer Sprint. En esta parte se muestra a detalle el desarrollo del primer Sprint.

- a. Planificación.** En la etapa de planificación del Sprint se define las tareas a realizar y el tiempo que tomará para su desarrollo. Si han ocurrido cambios se tomaran en cuenta.

Tabla 8. Sprint 1.

Código	Tareas	Estimación	Sprint
HU01-01	Crear la lógica necesaria para autenticar el usuario del sistema operativo.	20 días	1
HU02-01	Crear la interfaz que permita seleccionar el flujo		
HU02-02	Crear la lógica para obtener los flujos de la base de datos		
HU03-01	Crear la lógica para obtener la lista de tareas a realizar (patrón)		
HU03-01	Crear la lógica para leer las aplicaciones activas del sistema		
HU04-01	Crear la lógica para saber si se ha completado una tarea.		
HU04-02	Crear la lógica para registrar las tareas completadas.		
HU05-01	Crear la lógica para capturar y registrar el tiempo total de la tarea terminada.		

Fuente: Elaboración propia

- b. Ejecución del sprint.** En esta etapa se desarrollarán las tareas especificadas en el primer Sprint.

HU01-01 Crear la lógica necesaria para autenticar el usuario del sistema operativo. La autenticación del sistema agente consiste de la siguiente manera; El agente al iniciarse obtiene el usuario del sistema operativo, ese dato lo compara con la base de datos en la tabla usuario (previamente cargado con usuarios del directorio activo), si lo encuentra le permite el ingreso.

Figura 13. Lógica de captura de usuario de Windows.

```
WindowsIdentity currentUser = WindowsIdentity.GetCurrent();
Program.strUsuario = currentUser.Name.Substring(currentUser.Name.IndexOf(@"\") + 1);
```

Fuente: Elaboración propia

Figura 14. Tabla de usuarios.

	Dni	Alias	Password	Pin	AccPersonal
1	1	1	1	NULL	10000000000000000000000000000000
2	2	mlopez	1	NULL	
3	92	rlopez	1	NULL	
4	11	rtubon	1	NULL	
5	13	joegarci	1	NULL	
6	15	pdavilap	1	NULL	
7	17	lcando	1	NULL	
8	19	pcastillo	1	NULL	
9	21	jmrecinas	1	NULL	
10	23	jatituan	1	NULL	

Fuente: Elaboración propia

HU02-01 Crear la interfaz que permita seleccionar el flujo. Se construyó esta interfaz para mostrar los flujos del usuario autenticado, los flujos son mostrados en la grilla y se deberá seleccionar uno con el cual se empezará a trabajar.

Figura 15. Interfaz Selección de Flujo.

Código	Descripción
824	Requerimiento Tarjetas - Aumento de Cupo / Levante de Cancelación

Fuente: Elaboración propia

HU02-02 Crear la lógica para obtener los flujos de la base de datos. La aplicación obtiene la información de un servicio web, esa información se almacena en una lista y se carga a la grilla de la ventana mostrada anteriormente.

Figura 16. Lógica para obtener los flujos de la base de datos.

```

var rptaFlujo = cUtilitario.HttpGetJson(Program.strServerWeb + "/api/ObtenerListaFlujos?usuarioRedOperador=" + Program.strUsuario, "");
List<EFlujo> _LFlujos = JsonConvert.Deserialize<List<EFlujo>>(rptaFlujo);
this._LFlujos_Aux = _LFlujos.ToList();

if (_LFlujos_Aux.Count <= 0)
{
    MessageBox.Show("No existen flujos en la matriz de servicios asociados a este usuario. Contáctese con el administrador.",
        "GESDATA",
        MessageBoxButtons.OK,
        MessageBoxIcon.Exclamation,
        MessageBoxDefaultButton.Button1);
    Application.Exit();
}

```

Fuente: Elaboración propia

HU03-01 Crear la lógica para obtener la lista de tareas a realizar (patrón). Se consume el servicio “ObtenerListaPatrones” para obtener el listado de tareas con el cual se va a trabajar. Tal como se ve en la imagen se obtiene el listado y se realizan una serie de validaciones de si existe data o si la data tiene el formato correcto.

Figura 17. Lógica para obtener la lista de tareas a realizar.

```

var rptaPatrones = Utilitario.HttpGetJson(Program.strServerWeb + "/api/ObtenerListaPatrones?usuarioRedOperador=" + Program.strUsuario
+ "&codigoFlujo=" + strCodFlujo, "");
List<WsPatrones> _LPatrones = JsonConvert.Deserialize<List<WsPatrones>>(rptaPatrones);

if (_LPatrones.Count <= 0)
{
    MessageBox.Show("El flujo seleccionado no tiene tareas asociadas. Contáctese con el administrador.",
        "GESDATA",
        MessageBoxButtons.OK,
        MessageBoxIcon.Exclamation,
        MessageBoxDefaultButton.Button1);
    return;
}

Boolean ValidaFormato = ValidaFormatoPatrones(_LPatrones);
if(!ValidaFormato)
{
    MessageBox.Show("La matriz de patrones del flujo seleccionado no tiene el formato correcto. Contáctese con el administrador.",
        "GESDATA",
        MessageBoxButtons.OK,
        MessageBoxIcon.Exclamation,
        MessageBoxDefaultButton.Button1);
    return;
}

Program.listatareas = RetornaPatrones(_LPatrones);
Program.listatareasCab = RetornaPatronesCab(_LPatrones);

```

Fuente: Elaboración propia

HU03-02 Crear la lógica para leer las aplicaciones activas del sistema. Se crea la lógica “RetornaDatosProcesoActivo” que se utiliza para leer las aplicaciones activas del sistema el cual se utilizará para determinar que aplicaciones son las que está ingresando el operador.

Para esta lógica se hace uso del Api de Windows llamado “User32” el cual tiene funciones como por ejemplo para obtener el nombre de la ventana activa, entre otras.

Figura 18. Lógica para leer las aplicaciones activas del sistema.

```
private List<string> RetornaDatosProcesoActivo()
{
    string NombreProceso = string.Empty;
    string TituloVentana = string.Empty;
    string Omitir = string.Empty;

    var rpta = new List<string>();

    IntPtr hWnd = User32.GetForegroundWindow();//obtiene la ventana activa
    uint lpdwProcessId;
    User32.GetWindowThreadProcessId(hWnd, out lpdwProcessId);//obtiene el proceso id
    IntPtr hProcess = User32.OpenProcess(0x0410, false, lpdwProcessId);
    StringBuilder text = new StringBuilder(1000);
    User32.GetModuleBaseName(hProcess, IntPtr.Zero, text, text.Capacity);//obtiene el nombre del proceso

    NombreProceso = text.ToString().Trim().ToLower();
    NombreProceso = NombreProceso.ToLower().Replace(".exe", string.Empty);

    int length = User32.GetWindowTextLength(hWnd);
    StringBuilder sb = new StringBuilder(length + 1);
    User32.GetWindowText(hWnd, sb, sb.Capacity);

    TituloVentana = sb.ToString().Trim();

    User32.CloseHandle(hProcess);

    string strTipoAplica = string.Empty;
    Boolean bolEncontrado = false;

    rpta.Add(NombreProceso);//Nombre de proceso
    rpta.Add(TituloVentana);//Titulo de ventana
}
```

Fuente: Elaboración propia

HU04-01 Crear la lógica para saber si se ha completado una tarea. Se crea la lógica para “BuscarTarea” que permite saber si una tarea está en ejecución y si se está siguiendo con la secuencia o patrón.

Figura 19. Lógica para saber si se ha completado una tarea.

```

List<ETarea> listatareasBusq;
private Boolean BuscarTarea(string TipoAplica, string Url, string Teclas)
{
    Boolean rpta = false;
    try
    {
        listatareasBusq = Program.listatareas;

        //verifica si existen posibles siguientes, para hacer la busqueda
        int cta = Program.listatareas.Where(x => x.Patron.PosiblesSiguintes == 1).Count();
        if (cta > 0)
            listatareasBusq = Program.listatareas.Where(x => x.Patron.PosiblesSiguintes == 1).ToList();
        else
            listatareasBusq = Program.listatareas.ToList();

        string strUrl = Url;
        string strTipoAplica = TipoAplica;

        strUrl = AcotarTitulos(strTipoAplica, strUrl);

        //Filtro de la lista de tareas a partir de los 3 parametros
        List<ETarea> lista01 = Busqueda(listatareasBusq, 1, strTipoAplica, strTipoAplica);
        lista01 = Busqueda(lista01, 2, strUrl, strTipoAplica);
        lista01 = Busqueda(lista01, 3, Teclas, strTipoAplica);

        if (lista01.Count() > 0)
        {
            rpta = true;
            bolDeshacer = false;
        }
    }
}

```

Fuente: Elaboración propia

HU04-02 Crear la lógica para registrar las tareas completadas. Se crea la lógica para registrar tareas completadas, el registro el almacenado temporalmente en archivos de texto en formato JSON y posteriormente serán enviados a la base de datos mediante un servicio web.

Figura 20. Lógica para registrar las tareas completadas.

```
private void AgregaTareaRealizada(string CodigoTarea)
{
    try
    {
        string TiempoIniMin = "";
        DateTime minTiempo = Program.listatareas.Where(x => x.CodigoTarea == CodigoTarea).Where(x => x.FechaInicio < minTiempo).Min(x => x.FechaInicio);
        DateTime maxTiempo = Program.listatareas.Where(x => x.CodigoTarea == CodigoTarea).Where(x => x.FechaFin > maxTiempo).Max(x => x.FechaFin);
        //foreach (var item in Program.listatareas.Where(x => x.CodigoTarea == CodigoTarea))
        {
            TiempoIniMin = minTiempo.ToString();
            int intStproduccion = Program.listatareas.Where(x => x.CodigoTarea == CodigoTarea).Max(x => x.Stproduccion);

            string Perfiljecedorgrade = Program.listatareas.Where(x => x.CodigoTarea == CodigoTarea).Max(x => x.Perfiljecedorgrade);
            string Servicio = Program.listatareas.Where(x => x.CodigoTarea == CodigoTarea).Max(x => x.Servicio);
            string Subservicio = Program.listatareas.Where(x => x.CodigoTarea == CodigoTarea).Max(x => x.Subservicio);
            string Prodcomercial = Program.listatareas.Where(x => x.CodigoTarea == CodigoTarea).Max(x => x.Prodcomercial);
            string Flujoproceso = Program.listatareas.Where(x => x.CodigoTarea == CodigoTarea).Max(x => x.Flujoproceso);
            string Tarea = Program.listatareas.Where(x => x.CodigoTarea == CodigoTarea).Max(x => x.Tarea);
            string Modoejecucion = Program.listatareas.Where(x => x.CodigoTarea == CodigoTarea).Max(x => x.Modoejecucion);
            string Medio = Program.listatareas.Where(x => x.CodigoTarea == CodigoTarea).Max(x => x.Medio);
            string UnidadConteo = Program.listatareas.Where(x => x.CodigoTarea == CodigoTarea).Max(x => x.UnidadConteo);
            string Tmpreferencial = Program.listatareas.Where(x => x.CodigoTarea == CodigoTarea).Max(x => x.Tmpreferencial);
        }
    }
}
```

Fuente: Elaboración propia

HU05-01 Crear la lógica para capturar y registrar el tiempo total de la tarea terminada. Se obtiene la fecha y hora inicial y final de cada tarea completada, ese dato se almacena también en el archivo con estructura JSON.

Cada vez que se van realizando las tareas, el agente registra los datos de la tarea y el tiempo que le tomo en realizar cada tarea, lo almacena en archivo en el sistema para posteriormente enviarlo a través de un servicio web a la base de datos.

Figura 21. Lógica para registrar el tiempo total de la tarea terminada.

```

cOperacion obj = new cOperacion();

obj.codigoTarea = CodigoTarea;
obj.fechaOperacionHoraInicio = Convert.ToDateTime(TiempoIniMin).ToString("yyyyMMddHHmmss");
obj.fechaOperacionHoraFin = Convert.ToDateTime(maxTiempo).ToString("yyyyMMddHHmmss");
obj.fechaOperacion = DateTime.Now.ToString("dd/MM/yyyy");
obj.horaInicioOperacion = Convert.ToDateTime(TiempoIniMin).ToString("HH:mm:ss");
obj.horaFinOperacion = Convert.ToDateTime(maxTiempo).ToString("HH:mm:ss");
obj.operador = Program.strUsuario;
obj.equipo = Program.strNombreEquipo;
obj.ipAddress = Program.strIpAddress;
obj.Stproduccion = intStproduccion;

```

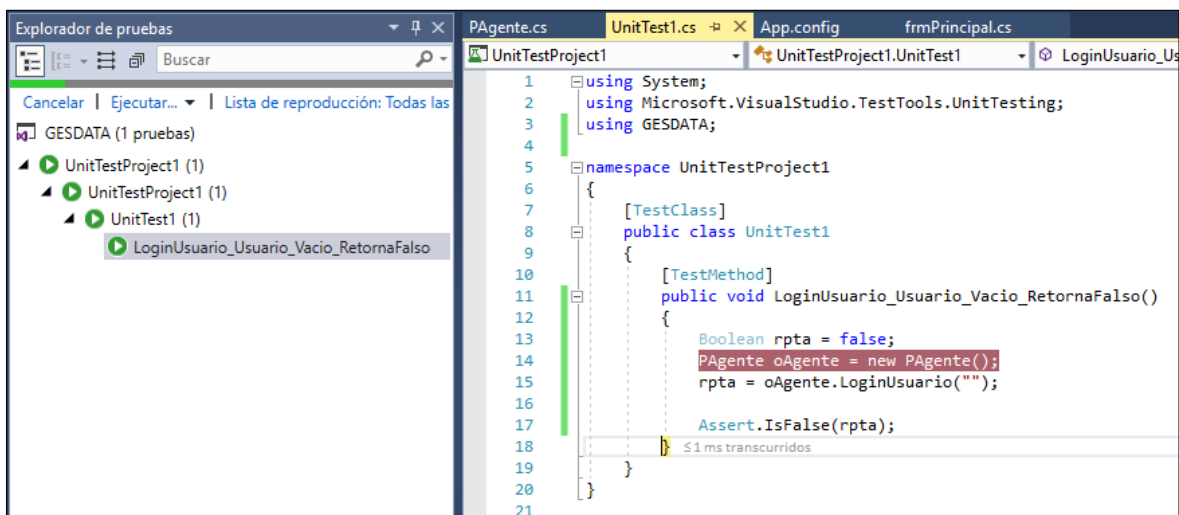
Fuente: Elaboración propia

c. **Pruebas.** El equipo realiza las pruebas funcionales según las tareas del sprint.

Para las pruebas unitarias se utiliza Visual Studio en un proyecto de tipo Unit Test.

LoginUsuario_Usuario_Vacio_ReturnaFalso. Para esta prueba se verifica que la autenticación del usuario debe ser falso para el caso de que se reciba un usuario con dato vacío.

Figura 22. Pruebas Unitarias Sprint 1



Fuente: Elaboración propia

d. *Sprint Review Meeting.* El equipo se reúne para hacer una demostración de lo que se ha logrado realizar y sobre las nuevas características de la aplicación, los participantes son; el equipo de desarrollo, Scrum Master, Product Owner y usuarios del módulo en cuestión. Para las actas de reunión ver Anexo 4.

e. *Sprint Retrospective.* El equipo se reúne y presenta problemas o inconvenientes ocurridos en el desarrollo del Sprint y se propone opciones de soluciones o mejoras. Para las actas de reunión ver Anexo 4.

3.3.3. Segundo Sprint. A continuación se describe a detalle el desarrollo de las fases del segundo sprint.

a. Planificación. En esta parte se define las tareas que son parte del segundo sprint y se realiza una estimación.

Tabla 9. Sprint 2.

Sprint Backlog			
Código	Tareas	Estimación	Sprint
HU06-01	Crear la lógica para contabilizar el total de tareas registradas.	20 días	2
HU07-01	Crear la interfaz para mostrar las notificaciones de tareas terminadas.		
HU07-02	Crear la lógica para mostrar notificaciones de tareas terminadas.		
HU08-01	Crear la lógica para mostrar como notificación mensaje de no se está siguiendo con el patrón.		
HU09-01	Crear lógica para mostrar como notificación las sub-tareas que continúan.		
HU10-01	Crear lógica de creación de archivo de registro de errores.		

Fuente: Elaboración propia

- b. **Ejecución del Sprint.** En esta etapa se desarrollarán las tareas especificadas en el primer Sprint.

HU06-01 Crear la lógica para contabilizar el total de tareas registradas. Para esta parte se puede obtener la contabilización de manera óptima, debido a que la forma de cómo se está registrando la información lo permite, la estructura de almacenamiento es detallada tarea por tarea.

Figura 23. Lógica para contabilizar el total de tareas registradas.

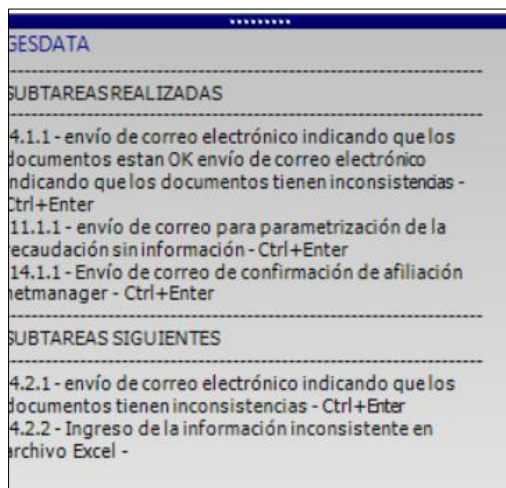
```
1  [{"  
2      "codigoTarea": "BPS-020-02-824-011011",  
3      "fechaOperacionHoraInicio": "20190418121127",  
4      "fechaOperacionHoraFin": "20190418121142",  
5      "fechaOperacion": "18/04/2019",  
6      "horaInicioOperacion": "12:11:27",  
7      "horaFinOperacion": "12:11:35",  
8      "operador": "Arnold",  
9      "equipo": "LAPTOP-F0GS3OCT",  
10     "ipAddress": "192.168.1.50",  
11     "OperacionPausas": [],  
12     "Stproduccion": 0  
13  },  
14  ],  
15  [{"  
16     "codigoTarea": "BPS-020-02-824-011011",  
17     "fechaOperacionHoraInicio": "20190418121155",  
18     "fechaOperacionHoraFin": "20190418121159",  
19     "fechaOperacion": "18/04/2019",  
20     "horaInicioOperacion": "12:11:55",  
21     "horaFinOperacion": "12:11:57",  
22     "operador": "Arnold",  
23     "equipo": "LAPTOP-F0GS3OCT",  
24     "ipAddress": "192.168.1.50",  
25     "OperacionPausas": [],  
26     "Stproduccion": 0  
27  },  
28  ],
```

Fuente: Elaboración propia

HU07-01 Crear la interfaz para mostrar las notificaciones de tareas terminadas.

Se utiliza la librería **Tulpep.NotificationWindow** para mostrar este tipo de notificaciones, se presenta la lógica que hace el llamado de la notificación al cual se le pasa como parámetro el formulario desde donde es invocado y el mensaje a mostrar.

Figura 24. Interfaz para mostrar las notificaciones de tareas terminadas.



Fuente: Elaboración propia

HU07-02 Crear la lógica para mostrar notificaciones de tareas terminadas. Para esta parte se crea la lógica necesaria para registrar las tareas terminadas y mostrar la notificación pertinente, de esta manera el operador está al tanto de su avance y puede continuar con la siguiente tarea. De manera técnica al momento de terminarse una tarea esta información es registrada como texto en formato JSON en archivos locales para luego ser enviados de manera automática a una base de datos.

Figura 25. Lógica para mostrar notificaciones de tareas terminadas.

```
public static void AbreNotificacion(Form obj, string Mensaje)
{
    obj.Invoke((MethodInvoker)delegate
    {
        Color tituloColor = Color.DarkBlue;
        Color textColor = Color.Black;
        Color textHoverColor = Color.DarkBlue;
        Color headerColor = Color.DarkBlue;
        Color fondoColor = Color.Silver;

        PopupNotifier popp = new PopupNotifier();
        popp.Size = new Size(300, 400);
        popp.TitleText = "GESDATA";
        popp.ContentText = Mensaje;
        popp.AnimationDuration = 1;
        popp.Popup();
        popp.ShowCloseButton = false;

        popp.TitleColor = tituloColor;
        popp.ContentHoverColor = textHoverColor;
        popp.ContentColor = textColor;
        popp.HeaderColor = headerColor;
        popp.BodyColor = fondoColor;

    });
}
```

Fuente: Elaboración propia

HU08-01 Crear la lógica para mostrar como notificación mensaje de no se está siguiendo con el patrón. El Agente debe poder detectar que no se está siguiendo con el patrón de la tarea a realizar, y debe poder emitir una notificación al operador y este poder retomar la tarea a realizar.

La lógica que se requiere se muestra en la siguiente figura.

Figura 26. Lógica para mostrar como notificación mensaje de no se está siguiendo con el patrón.

```

if (lista01.Count() > 0)
{
    rpta = true;
    bolDeshacer = false;

    foreach (var item in Program.listatareas)
        item.Patron.Posibles = 0;

    //Marca las subtareas como posibles
    PreparaMarcarSubTarea(lista01);

    //Obtiene las subtareas siguientes a partir de las posibles
    ObtenerSubtareasSiguientes();

    //
    List<ETarea> lstTarTermi = RetornaTareasTerminadas();

    ETarea oTarTermEleg = ElegirTareaTerminada(lstTarTermi);
    //muestra las notificaciones de posibles, posibles siguientes y tareas terminadas
    ShowNotificaciones(oTarTermEleg, "", true);

    MuestraEventoUsuario(oTarTermEleg);

    TerminarTarea(oTarTermEleg);

    tiempoIniAlerta = DateTime.Now;
}
else //No esta siguiendo el patron
{
    ShowNotificaciones(null, "NO SE ESTÁ SIGUIENDO CON EL PATRÓN.");
}

```

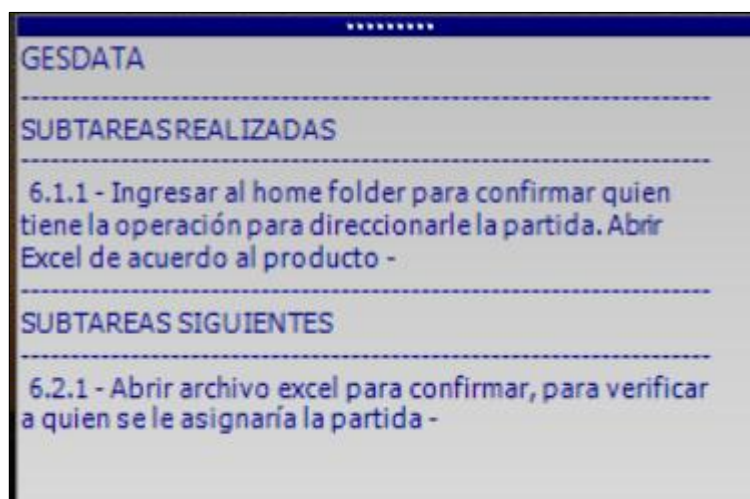
Fuente: Elaboración propia

HU09-01 Crear lógica para mostrar como notificación las sub-tareas que continúan. Se requiere que el Agente a la vez que identifica la sub-tarea que se está realizando pueda también identificar la siguiente sub-tarea a realizar y mostrarlo como notificación para que el operador pueda tomarlo en cuenta.

Para ello la estructura de la tabla que almacena las sub-tareas está diseñada de tal manera que se puede ordenar en forma secuencial y poder saber a partir de una determinada sub-tarea cuál es su sub-tarea siguiente y antecesora.

La lógica que se requiere se muestra en la siguiente figura.

Figura 27. Lógica para mostrar como notificación las sub-tareas que continúan.



Fuente: Elaboración propia

HU10-01 Crear lógica de creación de archivo de registro de errores. El Agente debe poder identificar los sucesos de errores de la aplicación ya sea generado inesperadamente por un factor externo o provocado por una falla no contemplada del sistema.

Debe poder también registrarlo localmente en un archivo de log de errores para su posterior lectura y análisis de la situación ocurrida.

La lógica que se requiere se muestra en la siguiente figura.

Figura 28. Lógica de creación de archivo de registro de errores.

```

public void LogFileWrite(string message)
{
    FileStream fileStream = null;
    StreamWriter streamWriter = null;

    for (int i = 1; i <= NumberOfRetries; ++i)
    {
        try
        {
            //string Rutax = Path.Combine(Application.StartupPath, FileName);
            string logFilePath = ""; // "c:\\GesdataError\\";

            logFilePath = Path.Combine(Application.StartupPath, "LogErrorGesdata" + "-" + i);

            if (logFilePath.Equals("")) return;
            Create the Log file directory if it does not exists

            if (!logFileInfo.Exists)
            {
                fileStream = logFileInfo.Create();
            }
            else
            {
                fileStream = new FileStream(logFilePath, FileMode.Append);
            }
            streamWriter = new StreamWriter(fileStream);
            streamWriter.WriteLine(message);
        }
        catch { }
    }
}

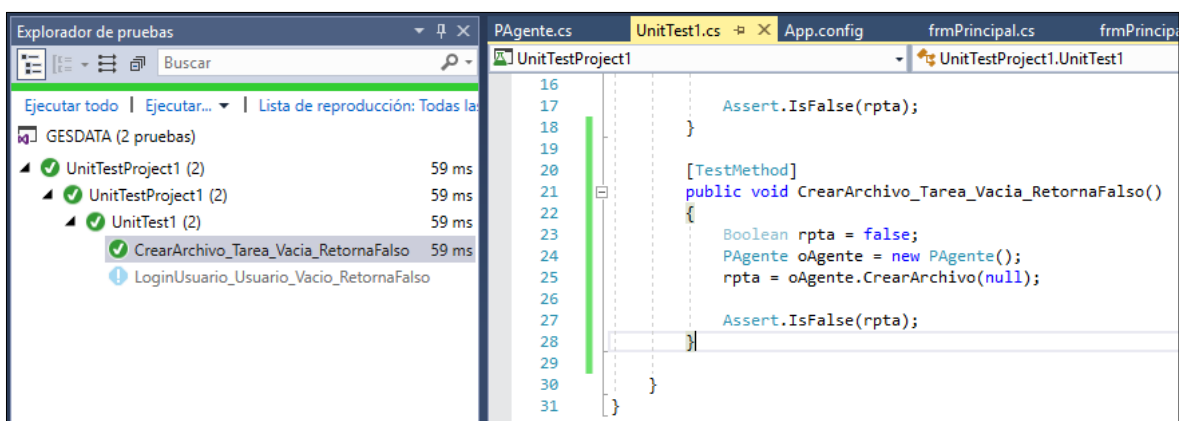
```

Fuente: Elaboración propia

c. **Pruebas.** El equipo realiza las pruebas unitarias según las historias de usuarios.

CrearArchivo_Tarea_Vacia_RetornaFalso. Esta prueba es para verificar si la lógica que crea los archivos de tareas completadas devuelve falso si no se ha pasado una tarea como parámetro, y como se muestra en la imagen, al ejecutar la prueba está devolviendo falso cuando no se especifica un parámetro, lo cual es correcto.

Figura 29. Pruebas Unitarias Sprint 2.



Fuente: Elaboración propia

d. *Sprint Review Meeting.* El equipo se reúne para hacer una demostración de lo que se ha logrado realizar y sobre las nuevas características de la aplicación, los participantes son; el equipo de desarrollo, Scrum Master, Product Owner y usuarios del módulo en cuestión. Para las actas de reunión ver Anexo 4.

e. *Sprint Retrospective.* El equipo se reúne y presenta problemas o inconvenientes ocurridos en el desarrollo del Sprint y se propone opciones de soluciones o mejoras. Para las actas de reunión ver Anexo 4.

3.3.4. Tercer Sprint. En esta parte se desarrolla a detalle el tercer Sprint.

a. *Planificación.* En esta fase se realiza el Sprint Backlog del tercer Sprint, se especifican las tareas a realizar y la estimación en días.

Tabla 10. Sprint 3.

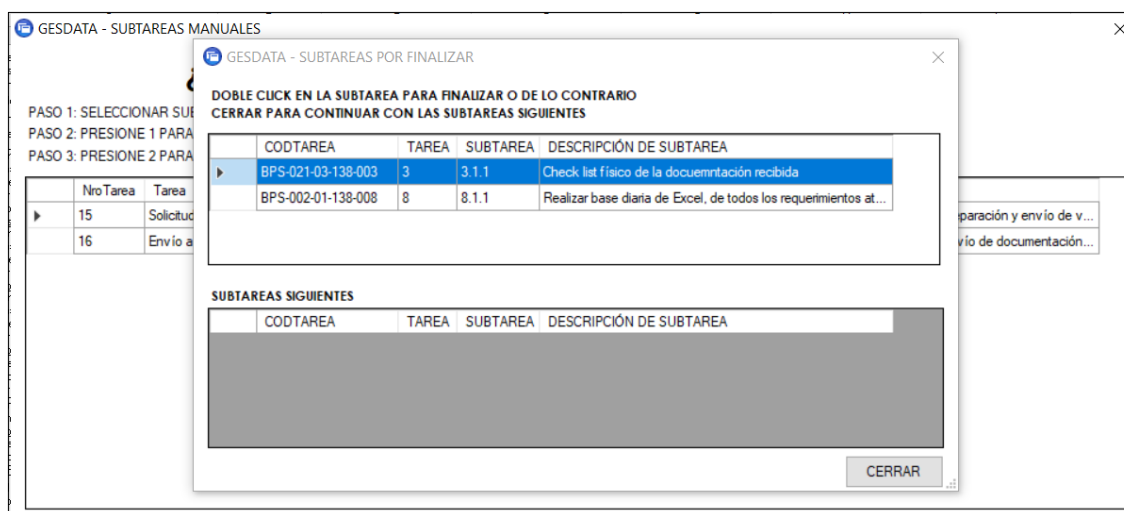
Sprint Backlog			
Código	Tareas	Estimación	Sprint
HU11-01	Crear interfaz para el registro tareas manuales.	20 días	3
HU11-02	Crear lógica para el registro de tareas manuales.		
HU12-01	Crear lógica para el monitoreo de los estados de agentes		
HU13-01	Crear interfaz para mostrar lista de tareas		
HU13-02	Crear lógica para mostrar lista de tareas seleccionadas		
HU14-01	Crear lógica de intercambio de flujos de manera automática.		
HU15-01	Crear lógica para poder deshacer acción.		

Fuente: Elaboración propia

b. Ejecución del Sprint. En esta etapa se desarrollarán las tareas especificadas en el primer Sprint.

HU11-01 Crear interfaz para el registro tareas manuales. Hasta ahora se tratado el registro de las tareas de manera automática, el Agente reconoce las actividades que el operador realiza en su computadora y lo va registrando de forma automática, pero también se debe ver el caso cuando son actividades físicas que el operador hace fuera del computador, para ello se debe crear una interfaz que le permita registrar esa actividad física y pueda ser tomado en consideración también.

Figura 30. Interfaz para el registro tareas manuales.



Fuente: Elaboración propia

HU11-02 Crear lógica para el registro de tareas manuales. Una vez creado la interfaz se crea la lógica que permita crear la tarea manual el cual deberá también tomar en cuenta el tiempo de inicio y fin de la tarea.

La lógica que se requiere se muestra en la siguiente figura.

Figura 31. Lógica para el registro de tareas manuales.

```

public void BuscaTareaManual(string NroTarea, int NroSubTareaN, string sNroSubTarea)
{
    ///quita las tareas que no se repiten y se han realizado
    //lista01 = BuscarRepiteTarea(lista01);
    Program.bolPrimeraBusq = false;
    foreach (var item in Program.listatareas)
        item.Patron.Posibles = 0;
    //marca subtarea manual
    MarcarSubTarea(NroTarea, NroSubTareaN, sNroSubTarea);
    //Obtiene las subtareas siguientes a partir de las posibles
    ObtenerSubtareasSiguienes();

    //MuestraEventoUsuario();

    //muestra las notificaciones de posibles, posibles siguientes y tareas terminadas
    //MuestraNotificaciones();
    List<ETarea> lstTarTermi = RetornaTareasTerminadas();

    ETarea oTarTermEleg = ElegirTareaTerminada(lstTarTermi);
    ShowNotificaciones(oTarTermEleg, "", true);

    MuestraEventoUsuario(oTarTermEleg);

    TerminarTarea(oTarTermEleg);

    tiempoIniAlerta = DateTime.Now;
}

```

Fuente: Elaboración propia

HUI2-01 Crear lógica para el monitoreo de los estados de agentes. El Agente debe poder comunicarse con una plataforma web y debe enviarle información acerca del estado actual del Agente en términos de si está activo o no.

Para ello enviará el archivo cada cierto tiempo a través de un servicio web, notificaciones a dicha plataforma web y esta mostrará en una pantalla el estado de los Agentes que están instalados en las computadoras de los operadores.

La lógica que se requiere se muestra en la siguiente figura.

Figura 32. Lógica para el monitoreo de los estados de agentes.

```

try
{
    ENotificarEstadoAgente Eobj = new ENotificarEstadoAgente();
    Eobj.equipo = Program.strNombreEquipo;
    Eobj.ipAddress = Program.strIpAddress;
    Eobj.operador = Program.strUsuario;
    Eobj.fechaHora = DateTime.Now.ToString("yyyyMMddHHmmss");
    Eobj.fecha = DateTime.Now.ToString("dd/MM/yyyy");
    Eobj.hora = DateTime.Now.ToString("HH:mm:ss");
    Eobj.encendido = strEncendidoEstado;

    //System.Runtime.CompilerServices.RuntimeHelpers.RunClassConstructor(HookManager HookManagers);

    HookManager.KeyDown += new KeyEventHandler(HookManager_KeyDown);
    HookManager.KeyDown += HookManager_KeyDown;

    string strJSON = JsonConvert.Serialize<ENotificarEstadoAgente>(Eobj);

    string strVar = cUtilitario.HttpPostJson(Program.strServerWeb + "/api/NotificarEstadoAgente", strJSON, "");
    Response_SW Eresp = JsonConvert.Deserialize<Response_SW>(strVar);
}

```

Fuente: Elaboración propia

HUI3-01 Crear interfaz para mostrar lista de tareas. Se requiere a través de un atajo de teclas poder mostrar un listado de las tareas seleccionas por el operador al inicio del flujo.

Figura 33. Interfaz para mostrar lista de tareas.

CodBPS	No.Tarea	Tarea	ms.SubTarea	SubTarea	TipoAplica	FocoSubtarea	FocoRequerimiento	EventoOperacion	EventoUsuario	AyudaContextual
BPS-021-01-138-001	1	Recepción del convenio físico, remitido por usuario Banco	1.1.1	Validar la información cargada en el HP, versus los docum...	Internet explorer %HP					
BPS-021-01-138-002	2	Identificación del tipo de requerimiento a través del análisis r...	2.1.1	Análisis del requerimiento ingresado	Internet explorer %HP					
BPS-021-03-138-003	3	Recepción del convenio físico, remitido por usuario Banco	3.1.1	Check list físico de la documentación recibida	Manual				1	Presione 1 si final...
BPS-021-01-138-004	4	Validación de fondo y forma. Fondo: Datos del convenio rec...	4.1.1	envío de correo electrónico indicando que los documentos ...	Correo Electrónico			Ctrl+Enter		
BPS-021-01-138-004	4	Validación de fondo y forma. Fondo: Datos del convenio rec...	4.2.1	envío de correo electrónico indicando que los documentos ...	Correo Electrónico			Ctrl+Enter		
BPS-021-01-138-004	4	Validación de fondo y forma. Fondo: Datos del convenio rec...	4.2.2	Ingreso de la información inconsistente en archivo Excel	Excel	Devueltos Conve...				
BPS-021-01-138-005	5	Envío de la confirmación de las instrucciones con visado es...	5.1.1	Ingreso a la herramienta definida para atención de requeri...	Internet explorer %HP					
BPS-002-01-138-006	6	Recepción de las instrucciones y documentos originales par...	6.1.1	Ingreso a la herramienta definida para afiliación de convenios	Internet explorer %easyafili...				2	Presione 2 si reali...
BPS-002-01-138-007	7	Creación de empresa (Datos generales de la empresa, cuen...	7.1.1	Asignación del ticket a administración de claves	Internet explorer %HP					
BPS-002-01-138-008	8	Asignación del ticket generado al área de administración de...	8.1.1	Realizar base diaria de Excel, de todos los requerimientos at...	Excel	Convenios				
BPS-002-05-138-009	9	Elaboración de reporte manual Base Excel con el detalle de...	9.1.1	Afiliación de la empresa en pagos pichincha o en motor de ...	Internet explorer %Motor de ...					
BPS-002-01-138-010	10	Parametrización de servicios recaudación para el canal WE...	10.1.1	Generación del ticket para el cobro de comisiones canal atm	Internet explorer %HP					
BPS-002-01-138-011	11	Generación de ticket del servicio de recaudaciones para el ...	11.1.1	envío de correo para parametrización de la recaudación sin...	Correo Electrónico			Ctrl+Enter		
BPS-002-01-138-012	12	Solicitud de la activación del plan novios por correo electró...	12.1.1	Afiliación de la empresa en pagos pichincha o en motor de ...	Internet explorer %Motor de ...					
BPS-002-01-138-013	13	Parametrización de servicios recaudación matrimonial para ...	13.1.1	Afiliación de la empresa para cobros de comisión canal CNB	Internet explorer %aystech					
BPS-002-01-138-014	14	Parametrización del canal CNB en el sistema %netmanager...	14.1.1	Envío de correo de confirmación de afiliación netmanager	Correo Electrónico			Ctrl+Enter		
BPS-002-01-138-015	15	Solicitud de activación del canal correspondiente por correo...	15.1.1	Envío de convenios físicos a custodia de valores	Manual				3	Presione 3 si final...
BPS-002-06-138-016	16	Envío a File	16.1.1	Envío de documentación física a File Storage	Manual				4	Presione 4 si final...

Fuente: Elaboración propia

HUI3-02 Crear lógica para mostrar lista de tareas seleccionadas. Una vez creada la interfaz se procede a crear la lógica necesaria para poder cargar la grilla de las tareas seleccionadas.

La lógica que se requiere se muestra en la siguiente figura.

Figura 34. Lógica para mostrar lista de tareas seleccionadas.

```

//LLamar al visor de patrones
if (teclaAnt == "LControlKey" && teclaAnt2 == "LMenu" && e.KeyCode.ToString() == "O")
{
    if (Program.ofrmVisorPatrones == null)
    {
        Program.ofrmVisorPatrones = new frmVisorPatrones();
        Program.ofrmVisorPatrones.Show();
    }
    else
    {
        Program.ofrmVisorPatrones.Activate();
    }
}
//frmVisorPatrones frmVP = new frmVisorPatrones();
//frmVP.Show();
HookManager.KeyDown += new KeyEventHandler(HookManager_KeyDown);
HookManager.KeyDown += HookManager_KeyDown;
}

```

Fuente: Elaboración propia

HUI4-01 Crear lógica de intercambio de flujos de manera automática. Se requiere poder elegir un flujo mediante un atajo de teclas y sin la necesidad de salir del sistema, para ello se debe validar que no se esté dentro de la ejecución de una tarea es decir que ya se debe haber finalizado una tarea o no se haya empezado alguna.

La lógica que se requiere se muestra en la siguiente figura.

Figura 35. Lógica de intercambio de flujos de manera automática.

```
//Cambio de flujo
if (teclaAnt == "LControlKey" && teclaAnt2 == "LMenu" && e.KeyCode.ToString() == "U")
{
    int intCtaPSigui = Program.listatareas.Where(x => x.Patron.PosiblesSiguietes == 1).Count();
    if (intCtaPSigui > 0)
    {
        MessageBox.Show("No se puede cambiar de flujo hasta completar la tarea.",
            "GESDATA",
            MessageBoxButtons.OK,
            MessageBoxIcon.Exclamation,
            MessageBoxDefaultButton.Button1);
    }
    else
    {
        bolMsjCamFlujo = true;
        if (MessageBox.Show("¿Desea seleccionar otro flujo?",
            "GESDATA",
            MessageBoxButtons.YesNo,
            MessageBoxIcon.Question,
            MessageBoxDefaultButton.Button1) == DialogResult.Yes)
        {
            Application.Restart();
        }else
        {
            bolMsjCamFlujo = false;
        }
    }
}
}
```

Fuente: Elaboración propia

HU15-01 Crear lógica para poder deshacer acción. Se requiere poder mediante un atajo de teclas deshacer una acción, es decir un reconocimiento de sub-tarea, por ejemplo si una tarea está compuesta de tres sub-tareas y el Agente reconoce que se realizó la primera, luego la segunda, pero el operador se cuenta que esa no es la acción que debió realizar, entonces debe tener la opción de regresar a la situación anterior cuando se reconoció la primera sub-tarea.

La lógica que se requiere se muestra en la siguiente figura.

Figura 36. Lógica para poder deshacer acción.

```

int ctaPosi = Program.listatareas.Where(x => x.Patron.Posibles == 1).Count();
if (ctaPosi > 0)
{
    bolDeshacer = true;
    //Obtiene las subtareas siguientes a partir de las posibles
    ObtenerSubtareasSiguietes();
    //muestra las notificaciones de posibles, posibles siguientes y tareas terminadas
    ShowNotificaciones(null, "ÚLTIMA ACCIÓN BORRADA");
}
else
{
    if (!bolUltAccTarTerm)
    {
        bolDeshacer = true;
        Program.bolPrimeraBusq = true;

        //borra los posibles siguientes
        foreach (var item in Program.listatareas)
            item.Patron.PosiblesSiguietes = 0;

        cUtilitario.AbreNotificacion(this, "ÚLTIMA ACCIÓN BORRADA");
    }
    else
    {
        bolDeshacer = true;
        cUtilitario.AbreNotificacion(this, "NO SE PUEDE DESHACER TAREA TERMINADA");
    }
}
}

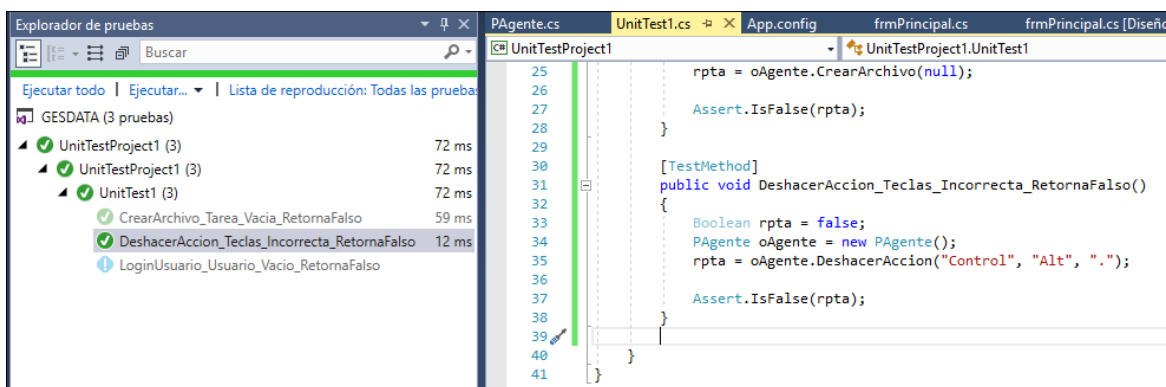
```

Fuente: Elaboración propia

- c. **Pruebas.** El equipo realiza las pruebas funcionales según las historias de usuarios.

DeshacerAccion_Teclas_Incorrecta_RetornaFalso. Esta prueba verifica el correcto funcionamiento de la función para deshacer una acción, lo que hace esta función es deshacer la última acción cuando se presione la tecla “Ctrl+Alt+Y”, en el caso que a la función se le envíe como parámetro una combinación de teclas que no es la indicada, la prueba dará como resultado falso.

Figura 37. Pruebas Unitarias Sprint 3.



Fuente: Elaboración propia

d. ***Sprint Review Meeting.*** El equipo se reúne para hacer una demostración de lo que se ha logrado realizar y sobre las nuevas características de la aplicación, los participantes son; el equipo de desarrollo, Scrum Master, Product Owner y usuarios del módulo en cuestión. Para las actas de reunión ver Anexo 4.

e. ***Sprint Retrospective.*** El equipo se reúne y presenta problemas o inconvenientes ocurridos en el desarrollo del Sprint y se propone opciones de soluciones o mejoras. Para las actas de reunión ver Anexo 4.

3.4. LANZAMIENTO

En esta fase se hace entrega del producto final y los entregables que corresponden.

3.4.1. Manual de Usuario. En este documento se explica paso a paso la manera de utilizar el sistema desde el punto de vista del usuario final.

Para revisar este documento ver el Anexo 5.

3.4.2. Acta de cierre del proyecto. Mediante este documento se especifica de manera formal que el proyecto está finalizado y el cliente debe revisarlo y aprobarlo. Para revisar este documento ver el Anexo 6.

CAPITULO 4

RESULTADOS

En esta parte del informe se muestra el resultado que tuvo el proyecto, la manera de medir el resultado es realizada mediante encuestas a las personas involucradas y usuarios del sistema, dichas encuestas son alineadas a cada uno de los objetivos específicos del proyecto, de esta manera se consigue medir el resultado del proyecto.

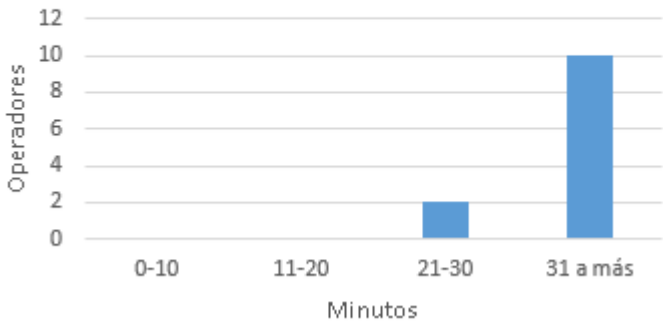
4.1.1. Implementar un sistema agente para reducir el tiempo de registro de las tareas realizadas en el área de operaciones.

Antes de la implementación del sistema.

El registro de las tareas realizadas era de manera manual y debido a eso los operadores debían de quedarse un tiempo más de su hora de trabajo para registrar los datos.

A continuación, se muestra una encuesta de cómo era el registro antes del sistema.

Tabla 11. Encuesta de tiempo de registro antes del sistema

Pregunta												
¿Cuánto es el tiempo en promedio que le tomaba realizar el registro de las tareas realizadas antes de implementarse el sistema?												
Encuestados												
Rol	Cantidad											
Operador	12											
Resultados												
Minutos	Cantidad Encuestados	Porcentaje										
0-10	0	0%										
11-20	0	0%										
21-30	2	16.66%										
31 a más	10	83.33%										
Gráfico y Análisis												
<p style="text-align: center;">Tiempo de registro de tareas antes del sistema</p>  <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>Minutos</th> <th>Operadores</th> </tr> </thead> <tbody> <tr> <td>0-10</td> <td>0</td> </tr> <tr> <td>11-20</td> <td>0</td> </tr> <tr> <td>21-30</td> <td>2</td> </tr> <tr> <td>31 a más</td> <td>10</td> </tr> </tbody> </table>			Minutos	Operadores	0-10	0	11-20	0	21-30	2	31 a más	10
Minutos	Operadores											
0-10	0											
11-20	0											
21-30	2											
31 a más	10											
<p>Se puede observar que dos de los encuestados tardaban entre 21 y 30 minutos y 10 de los encuestados tardaban de 31 a más, esto es debido a que el registro era manual.</p>												

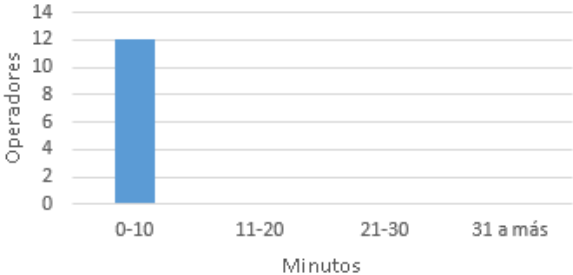
Fuente: Elaboración propia

Después de la implementación del sistema.

Una vez implementado el sistema los usuarios ya no tienen por qué registrar sus tareas, debido a que el sistema Agente lo realiza de manera automática.

A continuación, se muestra una encuesta tomada después de la implementación.

Tabla 12. Encuesta de tiempo de registro después del sistema

Pregunta		
¿Cuánto es el tiempo en promedio que le tomaba realizar el registro de las tareas realizadas después de implementarse el sistema?		
Encuestados		
Rol	Cantidad	
Operador	12	
Resultados		
Minutos	Cantidad Encuestados	Porcentaje
0-10	12	100%
11-20	0	0%
21-30	0	0%
31 a más	0	0%
Gráfico y Análisis		
<p style="text-align: center;">Tiempo de registro de tareas antes del sistema</p>  <p style="text-align: center;">Se puede observar que el 100% de los encuestados selecciono la primera opción es decir que ya no existe la demora en el tiempo de registro de las tareas.</p>		

Fuente: Elaboración propia

4.1.2. Implementar un sistema agente para salvaguardar la autenticación en el proceso de registro de las tareas realizadas en el área de operaciones.

Antes de la implementación del sistema.

El registro de las tareas realizadas por los operadores era de forma manual en archivos de tipo Excel, por tal motivo no existía una fiabilidad de la información contenida en tales archivos por la razón de que podían ser modificados y manipulados en cualquier momento y cualquier persona.

Después de la implementación del sistema.

Para solventar el asunto antes mencionado, se adicionó al sistema Agente una funcionalidad de autenticación cuando inicia el sistema. Esta característica permite tener identificado a la persona operador que está utilizando el sistema. El funcionamiento técnico es de la siguiente manera; El usuario operador tiene asignado una computadora de trabajo, dicha computadora esta registrado en el directorio activo de la organización, entonces al iniciar el sistema agente obtiene las credenciales de la computadora de trabajo del operador y verifica que exista en el directorio activo de la organización, si existe le permite el ingreso, de lo contrario se lo impide. Y de esta forma se tiene identificado al operador desde el inicio y todo registro de datos que se realice tiene asociado el usuario responsable sin posibilidad de modificación.

Como se puede observar en la figura se obtiene el usuario del sistema operativo y luego ese dato se pasa como parámetro a un servicio que es el que tiene la lógica de verificación si está en la base de datos.

Figura 38. Lógica autenticación de usuario.

```

WindowsIdentity currentUser = WindowsIdentity.GetCurrent();
Program.strUsuario = currentUser.Name.Substring(currentUser.Name.IndexOf(@"\") + 1);

Program.strNombreEquipo = System.Environment.MachineName;
Program.strIpAddress = Utilitario.GetIpAddress();

string mdoc = Environment.GetFolderPath(Environment.SpecialFolder.UserProfile);
Ruta_CARPETA = Path.Combine(mdoc, "GESDATA");
Utilitario.CrearCarpeta(Ruta_CARPETA);

string strTiempo = DateTime.Now.Date.ToString("yyyyMMdd");
string FileName = @"JSON_" + Program.strUsuario + "_" + strTiempo + ".txt";
Ruta = Path.Combine(Ruta_CARPETA, FileName);

try
{
    var rpt = Utilitario.HttpGetJson(Program.strServerWeb
    + "/api/LoginUsuarioOperador?usuarioRedOperador=" + Program.strUsuario, "");

    WsResponse oRpta = JsonConvert.DeserializeObject<WsResponse>(rpta);
    if (oRpta.codigoAccion > 0)
    {

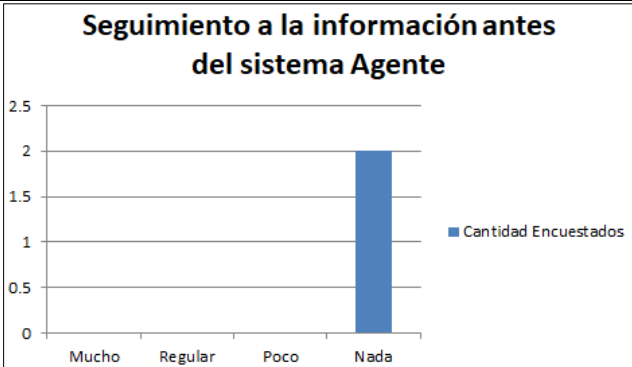
```

Fuente: Elaboración propia

4.1.3. Implementar un sistema Agente para mantener una trazabilidad de las actividades realizadas por un trabajador determinado en el área de operaciones.

Antes de la implementación del sistema. No existía una trazabilidad la información era registrada, pero no con los datos de fecha y hora de registro, usuario o computadora en donde se registró. Aquí se muestra una encuesta tomada antes de la implementación del sistema.

Tabla 13. Encuesta de seguimiento de la información antes del sistema.

Pregunta		
¿El sistema actual manual le permite hacer seguimiento de la información registrada?		
Encuestados		
Rol	Cantidad	
Encargado de TI	2	
Resultados		
Minutos	Cantidad Encuestados	Porcentaje
Mucho	0	0%
Regular	0	0%
Poco	0	0%
Nada	2	100%
Gráfico y Análisis		
<p style="text-align: center;">Seguimiento a la información antes del sistema Agente</p>  <p style="text-align: center;">Se puede observar que el 100% de los encuestados selecciono la opción de “Nada” es decir que antes no había manera de hacerle seguimiento a la información generada.</p>		

Fuente: Elaboración propia

Después de la implementación del sistema. El sistema permite registrar datos de trazabilidad y poder tener una fuente de datos más confiable.

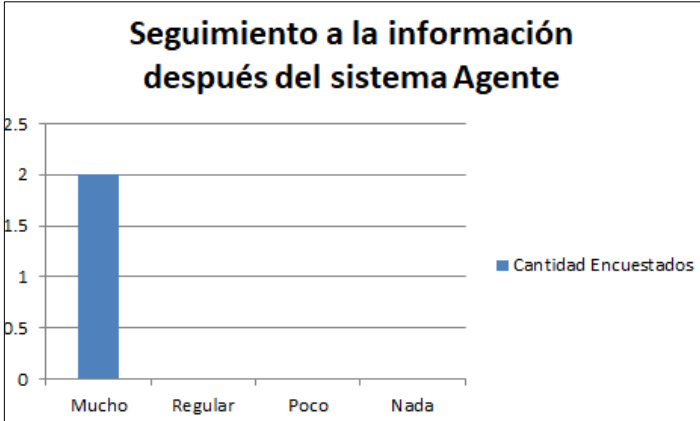
En la siguiente imagen se puede apreciar los datos generados que serán enviados a la base de datos. Y luego una encuesta tomada después de la implementación del sistema.

Figura 39. Datos de trazabilidad.

```
1  [{"  
2    "codigoTarea": "BPS-020-02-824-011011",  
3    "fechaOperacionHoraInicio": "20190418121127",  
4    "fechaOperacionHoraFin": "20190418121142",  
5    "fechaOperacion": "18/04/2019",  
6    "horaInicioOperacion": "12:11:27",  
7    "horaFinOperacion": "12:11:35",  
8    "operador": "Arnold",  
9    "equipo": "LAPTOP-F0GS3OCT",  
10   "ipAddress": "192.168.1.50",  
11   "OperacionPausas": [],  
12   "Stproduccion": 0  
13 },  
14 [{"  
15   "codigoTarea": "BPS-020-02-824-011011",  
16   "fechaOperacionHoraInicio": "20190418121155",  
17   "fechaOperacionHoraFin": "20190418121159",  
18   "fechaOperacion": "18/04/2019",  
19   "horaInicioOperacion": "12:11:55",  
20   "horaFinOperacion": "12:11:57",  
21   "operador": "Arnold",  
22   "equipo": "LAPTOP-F0GS3OCT",  
23   "ipAddress": "192.168.1.50",  
24   "OperacionPausas": [],  
25   "Stproduccion": 0  
26 }],  
27 ]
```

Fuente: Elaboración propia

Tabla 14. Encuesta de seguimiento de la información después del sistema.

Pregunta		
¿El sistema actual implementado ha mejorado la manera de hacer seguimiento de la información registrada?		
Encuestados		
Rol	Cantidad	
Encargado de TI	2	
Resultados		
Minutos	Cantidad Encuestados	Porcentaje
Mucho	2	100%
Regular	0	0%
Poco	0	0%
Nada	0	0%
Gráfico y Análisis		
<p style="text-align: center;">Seguimiento a la información después del sistema Agente</p>  <p style="text-align: center;">Se puede observar que el 100% de los encuestados selecciono la opción de “Mucho” es decir que ahora el sistema tiene la manera de hacerle seguimiento a la información generada.</p>		

Fuente: Elaboración propia

4.2. Análisis de costos

Para el desarrollo del presente proyecto se realizaron una serie de inversiones en temas de personal, hardware, software, etc. Esta parte del informe se trata de mostrar a detalle los cuadros que representan estas inversiones y un análisis de costo que nos va a permitir verificar si el proyecto es viable o no.

4.2.1. Costos de Personal

En el siguiente cuadro se muestra el personal técnico que participo en el desarrollo del proyecto y sus costos en el proyecto.

Figura 40. Costos de personal.

Cargo	Cantidad	Meses	Costo por Mes (S/.)	Costo Total (S/.)
Gerente del Proyecto	1	5	8,000.00	40,000.00
SCRUM Master	1	5	6,000.00	30,000.00
Analista Programador 1	1	5	3,500.00	17,500.00
Analista Programador 2	1	5	3,000.00	15,000.00
Analista de calidad	1	5	2,500.00	12,500.00
Total				115,000.00

Fuente: Elaboración propia

4.2.2. Costos de Hardware

En el siguiente cuadro se muestran el hardware utilizado en el proyecto y sus respectivos costos.

Figura 41. Costos de Hardware.

Descripcion	Unidades	Costo por Unidad (S/.)	Costo Total (S/.)
Servidor de Base de datos	1	5,000.00	5,000.00
Servidor de Aplicaciones	1	5,000.00	5,000.00
Total			10,000.00

Fuente: Elaboración propia

4.2.3. Costos de Software

En el siguiente cuadro se muestran los paquetes de software utilizados en el proyecto y sus respectivos costos.

Figura 42. Costos de Software.

Descripcion	Unidades	Meses	Costo Total (S/.)
Windows Server 2016	1	12	300.00
Windows 10	1	12	250.00
Visual studio 2017	1	12	750.00
SQL Server 2016	1	12	800.00
Total			2,100.00

Fuente: Elaboración propia

4.2.4. Costos Totales

En el siguiente se muestra un resumen de los costos totales de personal, hardware y software.

Figura 43. Costos Totales.

Tipo de Costo	Costo Total (S/.)
Personal	115,000.00
Hardware	10,000.00
Software	2,100.00
Total	127,100.00

Fuente: Elaboración propia

4.2.5. Costos variables

En el siguiente cuadro se muestran los costos variables utilizados en el proyecto.

Figura 44. Costos variables.

Descripción	Costo Total (S/.)
Electricidad	200.00
Agua	100.00
Internet y telefono	150.00
Total	450.00

Fuente: Elaboración propia

4.3. Análisis de Beneficios

4.3.1. Beneficios Tangibles

Es el aprovechamiento del resultado del proyecto y que se puede medir en términos monetarios, en la siguiente tabla se muestra que el beneficio se enfoca en el registro de las operaciones diarias que es realizado por los operadores, y que anteriormente era de manera manual y luego de la implementación del proyecto se hace de forma automática.

Figura 45. Beneficios Tangibles.

Beneficios Tangibles	Sin sistema			Con sistema			Total Beneficio (S/.)
	Horas de trabajo	RR.HH	Costo (S/.)	Horas de trabajo	RR.HH	Costo (S/.)	
Registro de operaciones diarias	4	300	24,000.00	0	0	0.00	24,000.00
Total							24,000.00

Fuente: Elaboración propia

4.4. Análisis de Flujo de Caja

4.4.1. Flujo de caja

En el siguiente grafico se muestra el análisis de flujo de caja del proyecto a doce meses, se puede observar en el mes cero se realiza los gastos que tienen que ver con el desarrollo del sistema, los costos variables se efectúan mes a mes, se visualiza los beneficios tangibles que se obtienen por mes también, se realiza el flujo de caja, y se puede observar también que a partir del sexto mes se empieza a recuperar la inversión.

4.4.2. Cálculo del VAN

El valor actual neto (VAN) es un indicador financiero que sirve para determinar si un proyecto es viable.

Figura 47. Fórmula del VAN.

$$VAN = B.N_0 + \frac{B.N_1}{(1+i)} + \frac{B.N_2}{(1+i)^2} + \dots + \frac{B.N_n}{(1+i)^n}$$

Fuente: Elaboración propia

$VAN < 0$ el proyecto no es rentable.

$VAN = 0$ el proyecto es rentable.

$VAN > 0$ el proyecto es rentable.

Para el presente proyecto el valor actual del VAN es mayor a cero, por lo tanto es rentable.

4.4.3. Cálculo del TIR

TIR es la tasa de descuento de un proyecto de inversión que permite que el Beneficio Neto Actualizado (BNA) sea igual a la inversión ($VAN = 0$).

Figura 48. Fórmula del TIR.

$$TIR = \frac{-I + \sum_{i=1}^n F_i}{\sum_{i=1}^n (i.F_i)}$$

Fuente: Elaboración propia

- Cuando $TIR >$ tasa de interés, se debe invertir en el proyecto.
- Cuando $TIR <$ tasa de interés, no se debe invertir en el proyecto.

Para el presente proyecto el valor calculado del TIR es mayor a la tasa de interés (10%), por lo tanto se debe invertir en el proyecto.

CONCLUSIONES

Se logró desarrollar un sistema agente para Automatizar el registro de las operaciones, de esta manera se redujeron los problemas que sucedían por registrar manualmente las operaciones.

Se logró armar una adecuada estructura de base de datos, para solventar los problemas de demora en acceso a la información, por que anteriormente se almacenaban en archivos de Excel.

Se implementó un módulo de acceso de usuario que permita tener la información de manera confidencial, y poder hacer el rastreo de la información generada y saber a quién le corresponde.

RECOMENDACIONES

Se recomienda que cuando exista una nueva aplicación que los usuarios deban utilizar y trabajar con ella, se deba registrar previamente dicha aplicación en el mantenimiento de aplicaciones para que el sistema agente pueda contemplarlo en su lectura.

Se recomienda que cuando ocurra algún inconveniente con el sistema agente ya sea que no está leyendo alguna aplicación o algún problema inesperado, revisar el registro “LOG” del agente ya que en dicho archivo se registra los problemas ocurridos.

Se recomienda estar al tanto de las notificaciones de alerta del agente ya que en ella muestra los problemas o inconvenientes que pudieran estar ocurriendo.

REFERENCIA BIBLIOGRÁFICA

Wooldridge. M, Jennings. R, (1994). *Intelligent Agents: Theory and Practice*, Knowledge Engineering Review.

Flanders. J, (2009). *RESTful.Net*, United States of America, O'Reilly.

(SBOK, 2017) *Guía SBOK*, SCRUMstudy.

(PMBOK, 2017) *Guía de proyectos PMBOK sexta Edición*.

Garro, H (2016) *Desarrollo de un sistema informático aplicando SCRUM para mejorar la gestión académica del Instituto Manuel Banda de Guadalupe* (Tesis de pregrado), Universidad Nacional de Trujillo, Guadalupe- Perú.

Godoy, E (2017) *Agente virtual inteligente para la orientación vocacional en el hogar Virgen de Fátima de la ciudad de Puno*, Tesis presentada para obtener el Título de Ingeniero Estadístico e Informático. Universidad Nacional del Altiplano - Puno.

Solano, J (2015) *Desarrollo de un Agente Inteligente para dispositivos móviles Android que permita llevar los ingresos y egresos económicos de una persona*, Tesis de Grado previo a la obtención del título de Ingeniería en Sistemas. Universidad Nacional de Loja - ECUADOR.

ANEXOS

ANEXO 1. ACTA DE CONSTITUCIÓN DEL PROYECTO	91
ANEXO 2. EDT	93
ANEXO 3. DICCIONARIO DE DATOS	94
ANEXO 4. ACTA DE REUNIÓN.....	96
ANEXO 5. MANUAL DE USUARIO	97
ANEXO 6. ACTA DE CIERRE DEL PROYECTO	111
ANEXO 7. INFORME DE TURNITIN.....	113

ANEXO 1. ACTA DE CONSTITUCIÓN DEL PROYECTO

ACTA DE CONSTITUCIÓN DEL PROYECTO	
Nombre de proyecto	Sistema agente para entidad financiera
Tiempo estimado	5 meses
Descripción de la necesidad del producto	Este proyecto tiene como objetivo implementar un sistema agente que le permita hacer el registro de las tareas realizadas por el personal en el área de operaciones de forma automatizada
Objetivo general del proyecto	Implementar un sistema agente para la gestión en el área de operaciones de una entidad financiera.
Alcance del proyecto	<p>La solución debe realizar el conteo de transacciones y registro de los tiempos de las tareas ejecutadas por el usuario en su PC.</p> <p>Para el proyecto se utilizará una metodología ágil, con una duración aproximada de 5 meses.</p> <p>La solución será utilizado por los usuarios del área de operaciones de la entidad financiera.</p> <p>La solución debe poder leer una matriz de patrones el cual estarán las tareas de manera secuencial de lo que debe realizar en el día el trabajador.</p> <p>La solución será construida en Visual Studio y con el lenguaje C#, será una aplicación de tipo winforms.</p>
Roles y responsabilidades	SCRUM MASTER PRODUCT OWNER Analista Programador 1 Analista programador 2 Analista de calidad
Resultado esperado	
Beneficio del proyecto	

Plan de trabajo

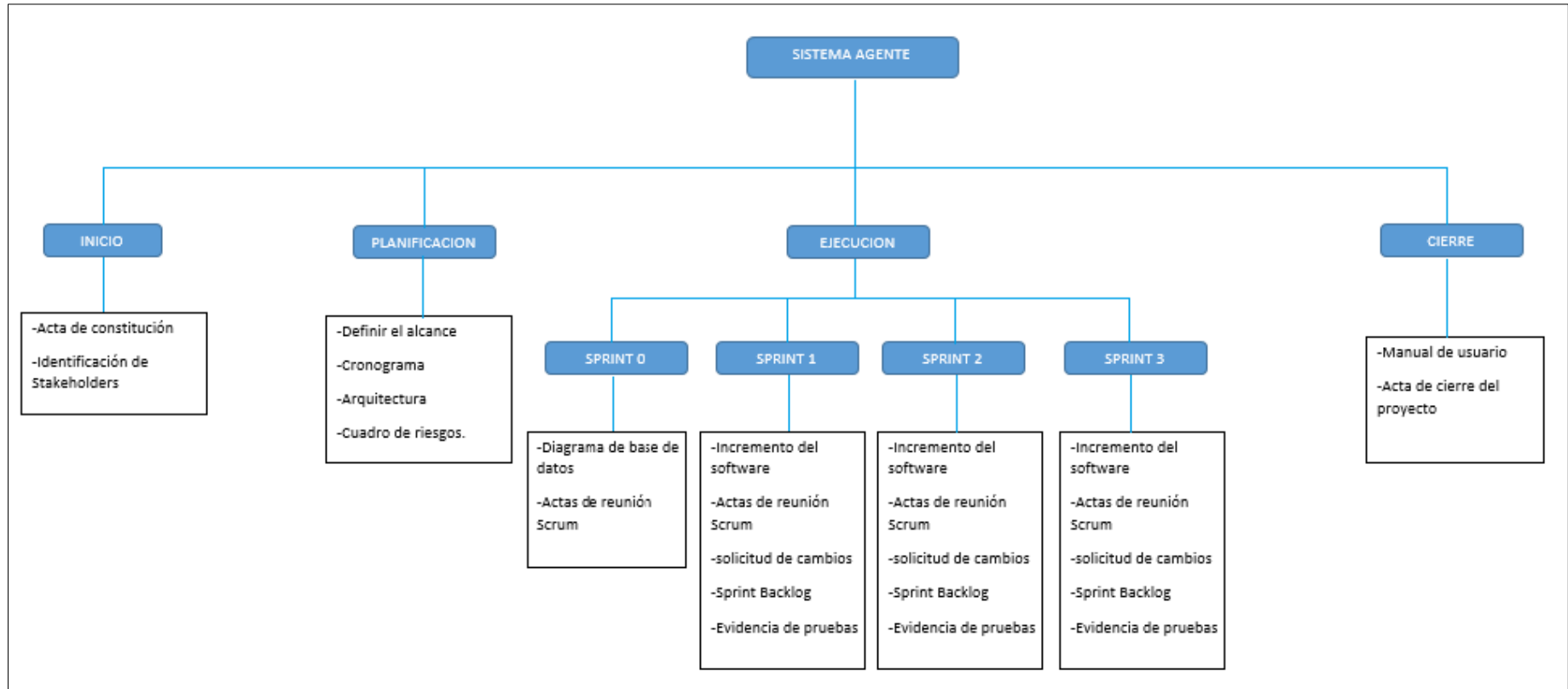
Fecha de Inicio : lunes 07/01/2019

Fecha de Fin : miércoles 27/05/2019

Jefe del área de sistemas
Empresa proveedora

Jefe del área de operaciones
Entidad Financiera

ANEXO 2. EDT



ANEXO 3. DICCIONARIO DE DATOS

Nombre de Tabla:	TipoAplicacion		
Descripción:	Guarda las aplicaciones que serán utilizadas por el sistema		
Nombre de campo	Tipo de dato	Tamaño	Descripción
Id	Int		Identificador
TipoAplicaPatron	Varchar	50	Nombre de la aplicación
CriterioProceso1	Varchar	200	Nombre de proceso
CriterioBuscaTitulo1	Varchar	200	Nombre del título de la ventana
CriterioExcluyeAplica1	Varchar	200	Nombre de proceso de exclusión
CriterioQuitaTitulo1	Varchar	200	Título de ventana de exclusión

Nombre de Tabla:	Pe_Frmmtzservicios		
Descripción:	Guarda la información de la matriz de servicios		
Nombre de campo	Tipo de dato	Tamaño	Descripción
Id	Int		Identificador
Tarea	Varchar	200	Nombre de la tarea
CodTarea	Varchar	200	Código de la tarea
FlujoProceso	Varchar	50	Código de flujo de proceso
NombreFlujo	Varchar	200	Nombre del flujo
Habilitado	Varchar	200	Indica si está habilitado

Nombre de Tabla:	Pe_mtzpatrones		
Descripción:	Guarda la información de las subtareas		
Nombre de campo	Tipo de dato	Tamaño	Descripción
Id	Int		Identificador
Nombre	Varchar	200	Nombre de la subtask
CodTarea	Varchar	200	Código de la tarea a la que pertenece
Habilitado	Varchar	200	Indica si está habilitado

Nombre de Tabla:	Operaciones		
Descripción:	Guarda la información de las tareas realizadas		
Nombre de campo	Tipo de dato	Tamaño	Descripción
Id	Int		Identificador
CodTarea	Varchar	200	Código de la tarea
FechaOperacionHoraInicio	Datetime		Fecha y hora de inicio de tarea
FechaOperacionHoraFin	Datetime		Fecha y hora de fin de tarea
Operador	Varchar	50	Usuario que hizo la tarea
Equipo	Varchar	50	Nombre de la pc

IpAddress	Varchar	50	Ip de la computadora
FechaRegistro	Datetime		Fecha y hora de registro de la tarea
Nombre de Tabla:	Usuario		
Descripción:	Guarda la información de los usuarios		
Nombre de campo	Tipo de dato	Tamaño	Descripción
Dni	Int		Identificador
Alias	Varchar	200	Nombre de usuario
Password	Varchar	50	contraseña
Habilitado	int		Indica si está habilitado

Nombre de Tabla:	Pe_mtzusuarioflujo		
Descripción:	Guarda la relación entre usuario y flujo		
Nombre de campo	Tipo de dato	Tamaño	Descripción
ID	Int		Identificador
Usuario	Varchar	200	Nombre de usuario
FlujoProceso	Varchar	50	Código de flujo

Nombre de Tabla:	Sw_Plan_integracion		
Descripción:	Guarda lo registros del resultado de los servicios		
Nombre de campo	Tipo de dato	Tamaño	Descripción
ID	Int		Identificador
Usuario	Varchar	200	Nombre de usuario
FechaHora	Datetime		Fecha y hora de registro
CodigoAccion	Int		Indica si ocurrió un error
CodigoMensaje	Int		Indica el código de mensaje de error
Descripcion	Varchar	200	Indica la descripción del error

ANEXO 4. ACTA DE REUNIÓN

Implementación de sistema Agente - Acta de la reunión <i>fecha</i> -	
Fecha, hora	fecha, 08:00
Lugar	Sala de reuniones
Asistentes	
Puntos a tratar	Punto 01: Tema A Punto 02: Tema B Punto 03: Tema C Otros asuntos
Detalle	
Punto 01: Tema A	
Observaciones	
Punto 02: Tema B	
Observaciones	
Punto 03: Tema C	
Observaciones	
Otros asuntos	

ANEXO 5. MANUAL DE USUARIO

MANUAL DE USUARIO

GESDATA

HISTORIAL DE VERSIONES	
VERSION	FECHA
Versión 1.0	06/03/2019
Versión 2.0	07/03/2019

La presente guía aplica al uso del agente GESDATA

Tabla de contenido

1. Introducción
2. Consideraciones Generales
3. Generalidades
4. AGENTE GESDATA
 - 4.1. INICIAR AGENTE
 - 4.2. COMBINACIÓN DE TECLAS
 - 4.3. SUBTAREAS MANUALES
 - 4.3.1. PASOS PARA EJECUTAR TAREAS MANUALES
 - 4.4. VISOR DE PATRONES (Lista de tareas y subtareas)
 - 4.5. ELECCION UN NUEVO FLUJO
 - 4.6. EJECUCIÓN DE TAREAS CON INTERNET EXPLORER

Introducción

Actualmente el 80% de las transacciones que produce la Fábrica de Operaciones es registrado de forma manual en bitácoras en Excel, información que posteriormente es ingresada manualmente en la herramienta Infotran. El 20% de las transacciones son contadas y registradas a través de la carga de tablas temporales con información de los diferentes sistemas (GAF, Bancs, Tablero Comex); este registro y conteo de transacciones presenta limitantes por cuanto no existe detalle ni información histórica.

Consideraciones Generales

OBJETIVO

El objetivo de la presente guía es brindar a los usuarios finales las orientaciones generales para el adecuado manejo del aplicativo GESDATA

ALCANCE

La presente guía está dirigida a los usuarios finales de banco Pichincha para la utilización del agente con el objetivo de realizar la contabilización de las tareas y el tiempo de ejecución real y esta información sea enviada a la plataforma web GESPLAN a través de un web Service.

FUNCIONALIDAD DE LA PLATAFORMA INFORMÁTICA

El agente instalado en cada una de las PCs de fábrica de operaciones y Adm. De claves, permite la autenticación del agente con el S.O, Permite que el operador seleccione el flujo a ejecutar y el agente empieza a contabilizar las tareas y tiempos, esta información capturada por el agente se registra de forma local (archivo encriptado) Cada cierto tiempo (parametizable) se envía la información a la base de datos mediante un web Service y toda esta información es gestionada por la aplicación web GESPLAN.

1.Generalidades

Para la utilización del agente GESDATA, se deben considerar los siguientes requerimientos básicos:

Requerimientos Básicos de hardware

- Procesador de datos Pentium 4
- 128 MB de memoria RAM
(recomendable)

Requerimientos Básicos para el Uso

- Sistema operativo: Windows 7 o superior
- Contar con conexión a Internet
- Resolución mínima de 800 x 600
- Tener instalado un navegador de Internet, tales como:
 - Internet Explorer versión 9 o

1. AGENTE GESDATA

Para acceder al agente GESDATA este tiene que estar previamente instalado y configurado en las PC o equipo del cual se quiere utilizar.

1.1. INICIAR AGENTE

Luego de haber iniciado la sesión de Windows, dar doble clic en el acceso directo del agente ubicado en el escritorio.



Figura 1. Acceso directo GESDATA

La aplicación valida si el usuario está registrado. Si el usuario no está registrado en la aplicación te mostrara una pantalla como se muestra en la figura a continuación.

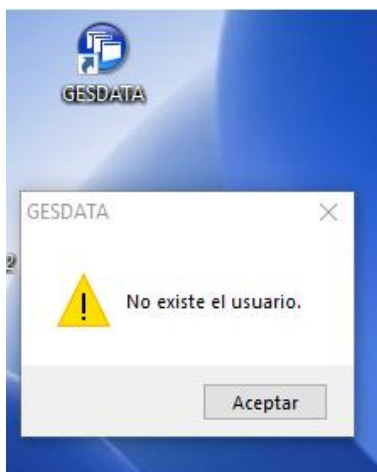


Figura 2. Imagen advertencia usuario no registrado

Si el usuario se encuentra registrado en aplicación. El sistema mostrará la lista de flujos como se muestra en la siguiente figura.

GESDATA - AGENTE

SELECCIONE EL FLUJO CERRAR AGENTE

Código	Descripción
824	Requerimiento Tarjetas - Aumento de Cupo / Levante de Cancelación

Figura 3. Inicio correcto agente – lista de patrones

En la aplicación se puede hacer búsquedas por la descripción

GESDATA - AGENTE

SELECCIONE EL FLUJO CERRAR AGENTE

Código	Descripción
824	Requerimiento Tarjetas - Aumento de Cupo / Levante de Cancelación

Figura 4. Búsqueda de patrones

Luego se deberá de seleccionar el flujo a validar, para ello hacer doble clic sobre el flujo, a partir de este momento el agente empezará a ejecutarse y validar las actividades del usuario. Se puede verificar en la barra de tareas como se muestra a continuación.

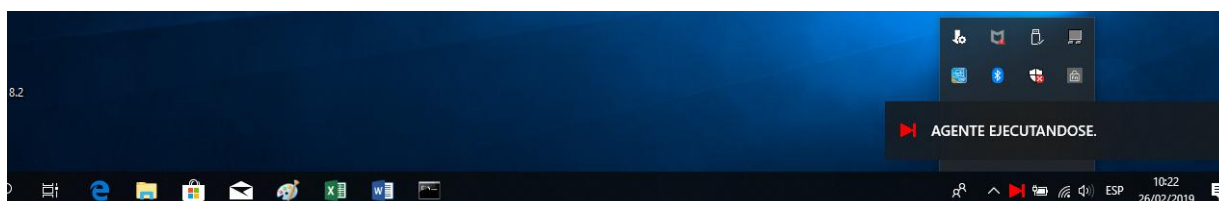


Figura 5. Agente iniciado (verificación en la barra de tareas)

Para finalizar el agente hacer clic derecho encima del ícono mostrado en la barra de tareas, luego dar clic en "CERRAR AGENTE"

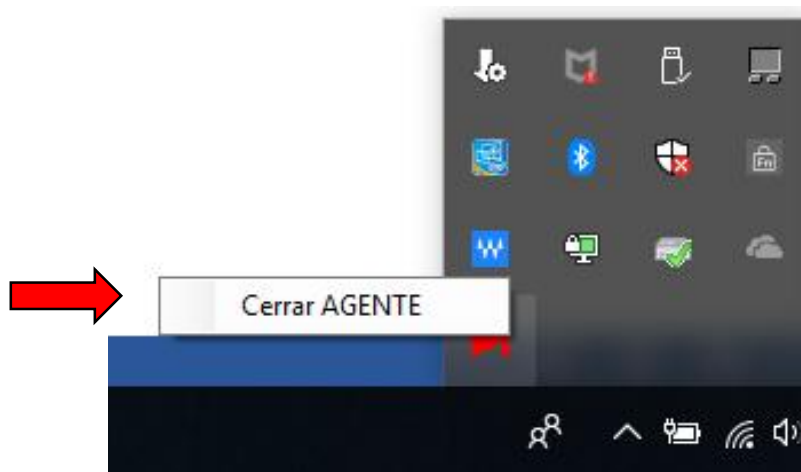


Figura 6. Cerrar agente

1.2. COMBINACIÓN DE TECLAS

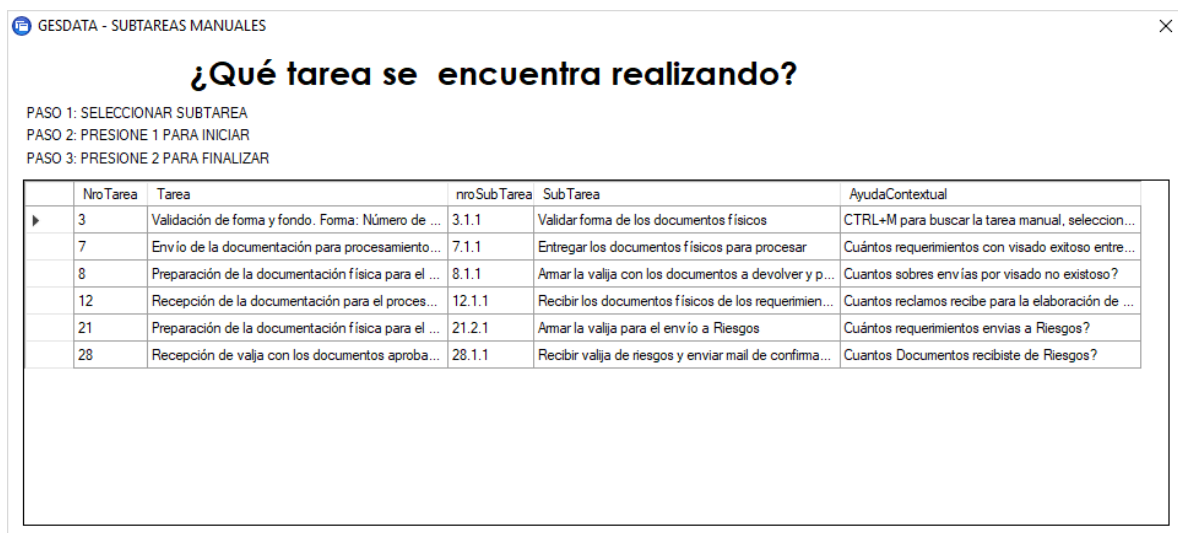
Para un fácil uso se han activado las siguientes combinaciones de teclas.

COMBINACIÓN DE TECLAS	EVENTO	DESCRIPCION
CTRL + ALT + T	Cierra la aplicación	Termina la ejecución del agente.
CTRL + ALT + O	Lista patrones	Muestra pantalla con la lista de patrones.
CTRL + ALT + U	Seleccionar nuevo Flujo	Muestra la pantalla de inicio de la aplicación.
CTRL + ALT + M	Lista tareas Manuales	Muestra la lista de tareas manuales a realizar.
CTRL + ALT + Y	Deshacer última acción	Permite deshacer última acción.

Figura 7. Tabla de combinación de teclas

1.3. SUBTAREAS MANUALES

Al usar la combinación de teclas CTRL + ALT + M, se muestra la lista de tareas manuales a realizar.



GESDATA - SUBTAREAS MANUALES

¿Qué tarea se encuentra realizando?

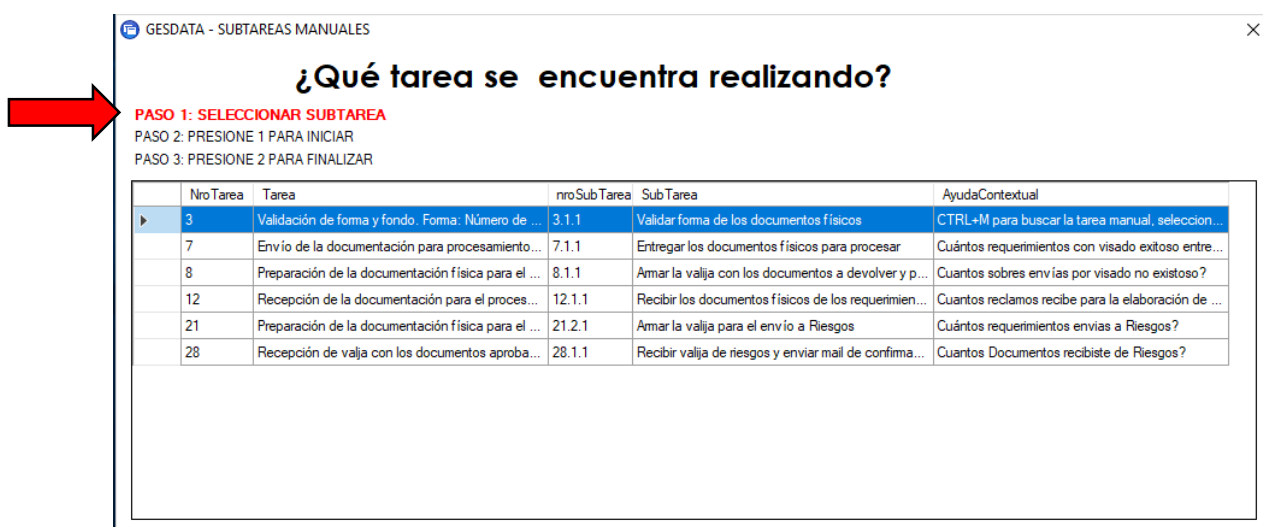
PASO 1: SELECCIONAR SUBTAREA
 PASO 2: PRESIONE 1 PARA INICIAR
 PASO 3: PRESIONE 2 PARA FINALIZAR

NroTarea	Tarea	nroSubTarea	SubTarea	AyudaContextual
3	Validación de forma y fondo. Forma: Número de ...	3.1.1	Validar forma de los documentos físicos	CTRL+M para buscar la tarea manual, seleccion...
7	Envío de la documentación para procesamiento...	7.1.1	Entregar los documentos físicos para procesar	Cuántos requerimientos con visado exitoso entre...
8	Preparación de la documentación física para el ...	8.1.1	Amar la valija con los documentos a devolver y p...	Cuántos sobres envías por visado no existoso?
12	Recepción de la documentación para el proces...	12.1.1	Recibir los documentos físicos de los requerimien...	Cuántos reclamos recibe para la elaboración de ...
21	Preparación de la documentación física para el ...	21.2.1	Amar la valija para el envío a Riesgos	Cuántos requerimientos envías a Riesgos?
28	Recepción de valija con los documentos aproba...	28.1.1	Recibir valija de riesgos y enviar mail de confirma...	Cuántos Documentos recibiste de Riesgos?

Figura 8. Ejecución tareas manuales

1.3.1. PASOS PARA EJECUTAR TAREAS MANUALES

Paso 1: Seleccionamos una tarea requerida, la aplicación remarcará como paso 1 como se muestra en la figura.



GESDATA - SUBTAREAS MANUALES

¿Qué tarea se encuentra realizando?

PASO 1: SELECCIONAR SUBTAREA
 PASO 2: PRESIONE 1 PARA INICIAR
 PASO 3: PRESIONE 2 PARA FINALIZAR

NroTarea	Tarea	nroSubTarea	SubTarea	AyudaContextual
3	Validación de forma y fondo. Forma: Número de ...	3.1.1	Validar forma de los documentos físicos	CTRL+M para buscar la tarea manual, seleccion...
7	Envío de la documentación para procesamiento...	7.1.1	Entregar los documentos físicos para procesar	Cuántos requerimientos con visado exitoso entre...
8	Preparación de la documentación física para el ...	8.1.1	Amar la valija con los documentos a devolver y p...	Cuántos sobres envías por visado no existoso?
12	Recepción de la documentación para el proces...	12.1.1	Recibir los documentos físicos de los requerimien...	Cuántos reclamos recibe para la elaboración de ...
21	Preparación de la documentación física para el ...	21.2.1	Amar la valija para el envío a Riesgos	Cuántos requerimientos envías a Riesgos?
28	Recepción de valija con los documentos aproba...	28.1.1	Recibir valija de riesgos y enviar mail de confirma...	Cuántos Documentos recibiste de Riesgos?

Figura 9. Selección de subtarea

Paso 2: si presionamos el número 1 de la aplicación, la aplicación remarcará el paso 2 como se muestra en la figura. También se iniciará el tiempo de realización de la tarea manual requerida.

Observación. Si se presiona el número 1 en una tarea, la aplicación iniciará el tiempo de la tarea, pero en caso se selecciona otra tarea el tiempo se volverá a iniciar con la nueva tarea elegida.

GESDATA - SUBTAREAS MANUALES

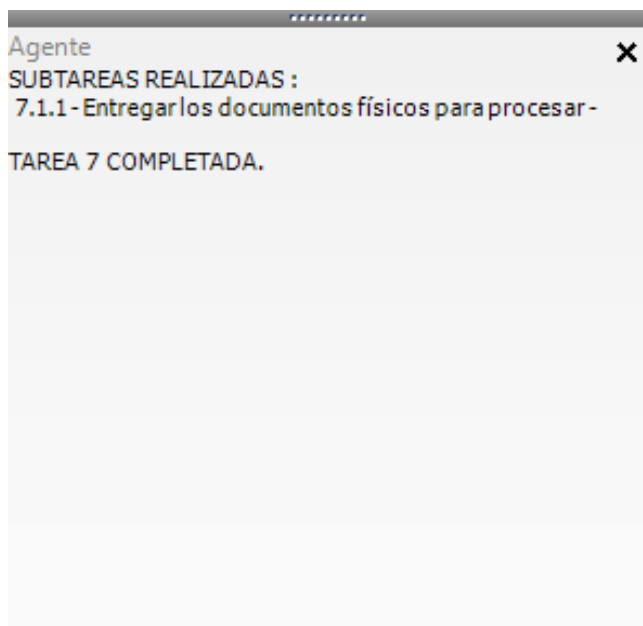
¿Qué tarea se encuentra realizando?

PASO 1: SELECCIONAR SUBTAREA
PASO 2: PRESIONE 1 PARA INICIAR
 PASO 3: PRESIONE 2 PARA FINALIZAR

Nro Tarea	Tarea	nroSubTarea	SubTarea	AyudaContextual
3	Validación de forma y fondo. Forma: Número de ...	3.1.1	Validar forma de los documentos físicos	CTRL+M para buscar la tarea manual, seleccion...
7	Envío de la documentación para procesamiento...	7.1.1	Entregar los documentos físicos para procesar	Cuántos requerimientos con visado exitoso entre...
8	Preparación de la documentación física para el ...	8.1.1	Amar la valija con los documentos a devolver y p...	Cuántos sobres envías por visado no existoso?
12	Recepción de la documentación para el proces...	12.1.1	Recibir los documentos físicos de los requerimien...	Cuántos reclamos recibe para la elaboración de ...
21	Preparación de la documentación física para el ...	21.2.1	Amar la valija para el envío a Riesgos	Cuántos requerimientos envías a Riesgos?
28	Recepción de valija con los documentos aproba...	28.1.1	Recibir valija de nesgos y enviar mail de confirma...	Cuántos Documentos recibiste de Riesgos?

Figura 10. Iniciar subtarea.

Paso 3: Si presionamos el número 2, la aplicación cerrará la pantalla del listado de tareas manuales y se mostrará una ventana con el mensaje de tarea completada como se muestra en la figura.



1.4. VISOR DE PATRONES (Lista de tareas y subtareas)

Al usar la combinación de teclas CTRL + ALT + O, se muestra la lista de tareas con sus respectivas subtareas como se muestra en la figura a continuación.

CodBPS	Nro Tarea	Tarea	nro Sub Tarea	Sub Tarea	Tipo Aplica	Foco Subtarea	Foco Requerimiento	Evento Operacion
BPS-021-01-824-001	1	Recepción del correo electrónico con la notificación de los	1.1.1	Recibir los datos físicos y validar vs el detalle del mail	Correo electrónico	OUTLOOK EXE		
BPS-021-01-824-002	2	Recepción del requerimiento por reclamos y requerimientos ...	2.1.1	Ingresar número de reclamo, seleccionar el registro, en la pe...	Internet Explorer%Requerimi...		http://ecbppnw3...	
BPS-021-01-824-003	3	Validación de forma y fondo. Forma: Número de documento...	3.1.1	Validar forma de los documentos físicos	Manual			
BPS-021-01-824-004	4	Validación de forma y fondo. Fondo: Para el levantamiento ...	4.1.1	En req y reclamos ir a pantalla "Datos Cliente", marcar el nú...	Internet Explorer%Requerimi...		http://ecbppnw3...	Ctrl+C
BPS-021-01-824-004	4	Validación de forma y fondo. Fondo: Para el levantamiento ...	4.2.1	Ingresar a AS400, pantalla "Datos del socio" Verificar visua...	AS400		Nexus Mainframe...	Ctrl+V
BPS-021-01-824-005	5	Envío de la confirmación del requerimiento con visado exito...	5.1.1	Ingresar a Requerimientos y Reclamos, copia datos del cle...	Internet Explorer%Requerimi...		http://ecbppnw3...	Ctrl+C
BPS-021-01-824-005	5	Envío de la confirmación del requerimiento con visado exito...	5.2.1	Pegar en Excel el número de requerimiento	Excel		BITACORA R&R ...	Ctrl+V
BPS-021-01-824-005	5	Envío de la confirmación del requerimiento con visado exito...	5.3.1	Ingresar a Requerimientos y Reclamos, copia datos del cle...	Internet Explorer%Requerimi...		http://ecbppnw3...	Ctrl+C
BPS-021-01-824-005	5	Envío de la confirmación del requerimiento con visado exito...	5.4.1	Pegar en Excel identificación	Excel		BITACORA R&R ...	Ctrl+V
BPS-021-01-824-005	5	Envío de la confirmación del requerimiento con visado exito...	5.5.1	Ingresar a Requerimientos y Reclamos, copia datos del cle...	Internet Explorer%Requerimi...		http://ecbppnw3...	Ctrl+C
BPS-021-01-824-005	5	Envío de la confirmación del requerimiento con visado exito...	5.6.1	Pegar en Excel número de tarjeta	Excel		BITACORA R&R ...	Ctrl+V
BPS-021-01-824-005	5	Envío de la confirmación del requerimiento con visado exito...	5.7.1	Ingresar a Requerimientos y Reclamos, copia datos del cle...	Internet Explorer%Requerimi...		http://ecbppnw3...	Ctrl+C
BPS-021-01-824-005	5	Envío de la confirmación del requerimiento con visado exito...	5.8.1	Pegar en Excel nombres	Excel		BITACORA R&R ...	Ctrl+V
BPS-021-01-824-005	5	Envío de la confirmación del requerimiento con visado exito...	5.9.1	Ingresar a R y R para buscar el siguiente requerimiento	Internet Explorer%Requerimi...		http://ecbppnw3...	
BPS-021-01-824-005	5	Envío de la confirmación del requerimiento con visado exito...	5.9.2	En devolución por visado inconsistente, copia el comentario...	Excel		BITACORA R&R ...	Ctrl+C
BPS-021-01-824-005	5	Envío de la confirmación del requerimiento con visado exito...	5.9.2.1	Pegar en Comentarios el motivo de devolución	Internet Explorer%Requerimi...		http://ecbppnw3...	Ctrl+V
BPS-021-01-824-005	5	Envío de la confirmación del requerimiento con visado exito...	5.9.2.2	Grabar el comentario y enviar para pasar el servicio	Internet Explorer%Requerimi...		ie explore.exe pop up	
BPS-021-01-824-006	6	Notificación de los requerimientos que continúan para verifi...	6.1.1	Desde Excel copia los requerimientos que pasan a VT	Excel		BITACORA R&R ...	
BPS-021-01-824-006	6	Notificación de los requerimientos que continúan para verifi...	6.2.1	Redactar correo electrónico para notificar el visado exitoso...	Correo electrónico	OUTLOOK Exe		Ctrl+Enter
BPS-021-03-824-007	7	Envío de la documentación para procesamiento de aument...	7.1.1	Entregar los documentos físicos para procesar	Manual			
BPS-021-03-824-008	8	Preparación de la documentación física para el envío a reg...	8.1.1	Amar la valija con los documentos a devolver y pegar sticker	Manual			
BPS-021-03-824-009	9	Ingreso en la herramienta MCS (Selección del destinatario, l...	9.1.1	Ingresar al MCS y registrar información de los datos a devol...	Internet Explorer%MCS		http://10.0.0.236...	
BPS-021-03-824-010	10	Entrega de la documentación física para regularización al p...	10.1.1	Entregar al pool de mensajería los documentos físicos	Manual	No considerar		
BPS-020-02-824-011	11	Recepción del requerimiento por reclamos y requerimientos ...	11.1.1	Recibir los requerimientos por correo electrónico.	Correo Electrónico	OUTLOOK EXE		
BPS-020-05-824-012	12	Recepción de la documentación para el procesamiento de ...	12.1.1	Recibir los documentos físicos de los requerimientos y verifi...	Manual			
BPS-020-02-824-013	13	Generación de base excel de verificación telefónica con da...	13.1.1	Renombrar en el Explorar de Windows el archivo personaliz...	Excel		VT_	
BPS-020-02-824-014	14	Envío de base excel para verificación telefónica con inform...	14.1.1	Redactar correo electrónico para enviar las fichas de VT. A...	Correo electrónico	OUTLOOK EXE...		Ctrl+Enter
BPS-002-02-824-015	15	Recepción de base excel de verificación telefónica con inform...	15.1.1	Envío de base excel para verificación telefónica con información de los requerimientos exitosos por correo electrónico.	Correo Electrónico	OUTLOOK EXE		
BPS-002-02-824-016	16	Consulta telefónica de datos (personales) a los clientes.	16.1.1	Abir Excel, verificar teléfonos y llamar cliente por cliente	Excel	VT_		
BPS-002-02-824-017	17	Envío de base excel de verificación telefónica actualizada ...	17.1.1	Redactar correo electrónico para notificar las VT realizadas ...	Correo Electrónico	OUTLOOK EXE		Ctrl+Enter
BPS-002-02-824-018	18	Actualización de datos básicos del cliente en el sistema AS ...	18.1.1	Abir Excel y copiar de VT el número de cédula del cliente	Excel		Excel	Ctrl+C
BPS-002-02-824-018	18	Actualización de datos básicos del cliente en el sistema AS ...	18.2.1	Ingresar a As400, pegar el número de cédula del cliente, En...	AS400		Nexus Mainframe...	Ctrl+V

Figura 12. Visor de patrones

1.5. ELECCION UN NUEVO FLUJO

Al usar la combinación de teclas CTRL + ALT + U, la aplicación mostrará un mensaje de confirmación.

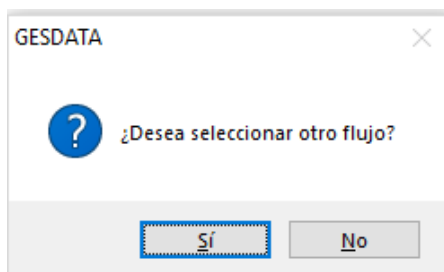


Figura 13. Elección de un nuevo flujo

Si le damos en "SI" la aplicación mostrará la pantalla de flujos en la cual se puede seleccionar una tarea, en caso de que seleccionemos "NO" se cerrará mensaje de confirmación.

1.6. EJECUCIÓN DE TAREAS CON INTERNET EXPLORER

Ingresamos en la url de la web <http://ecbpprw30.uio.com/Reclamos/> la cual mostrará una pantalla en la cual aparecen las tareas a finalizar como se muestra en la figura.

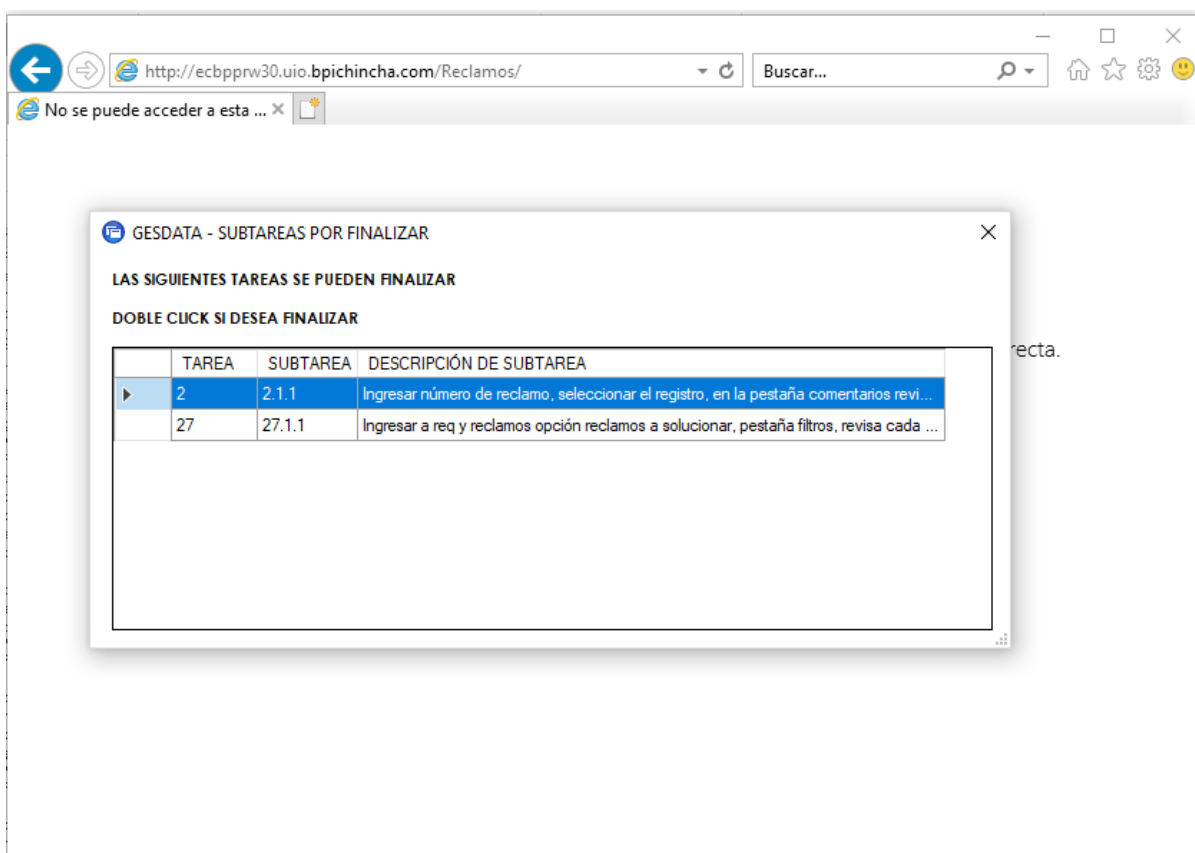


Figura 14. Ejecución de tareas con Internet Explorer

Al realizar doble clic a la tarea seleccionada, el sistema muestra un mensaje de que la tarea ha sido completada como se muestra en la figura a continuación.

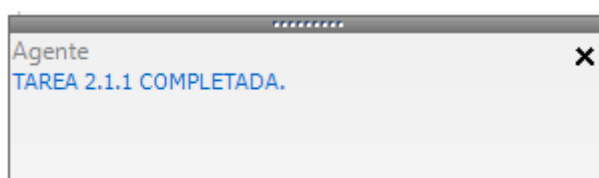


Figura 15. Tarea completada

Observación: en caso de no seguir el patrón de tareas el agente mostrará una lista con las tareas pendientes. La notificación de que "NO SE ESTA SIGUIENDO CON EL PATRON" hace referencia a que se está presionando otras combinaciones de tecla en las tareas incorrectas.

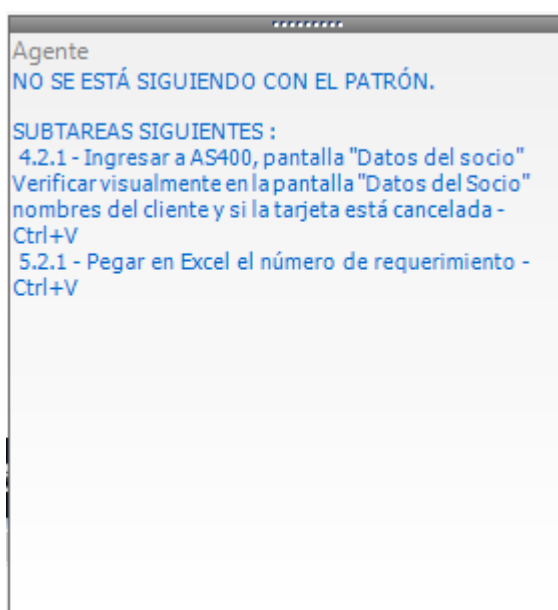
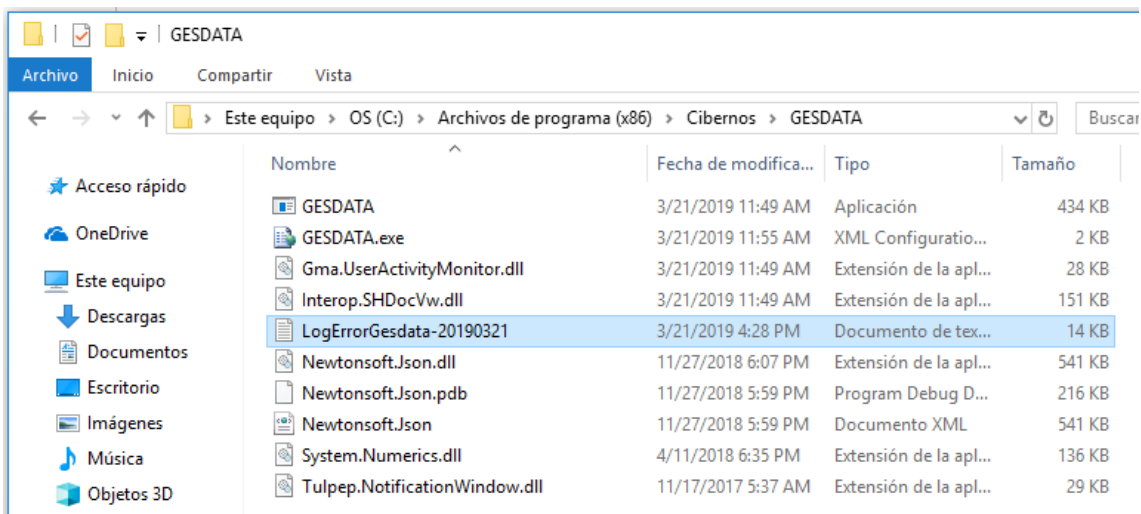


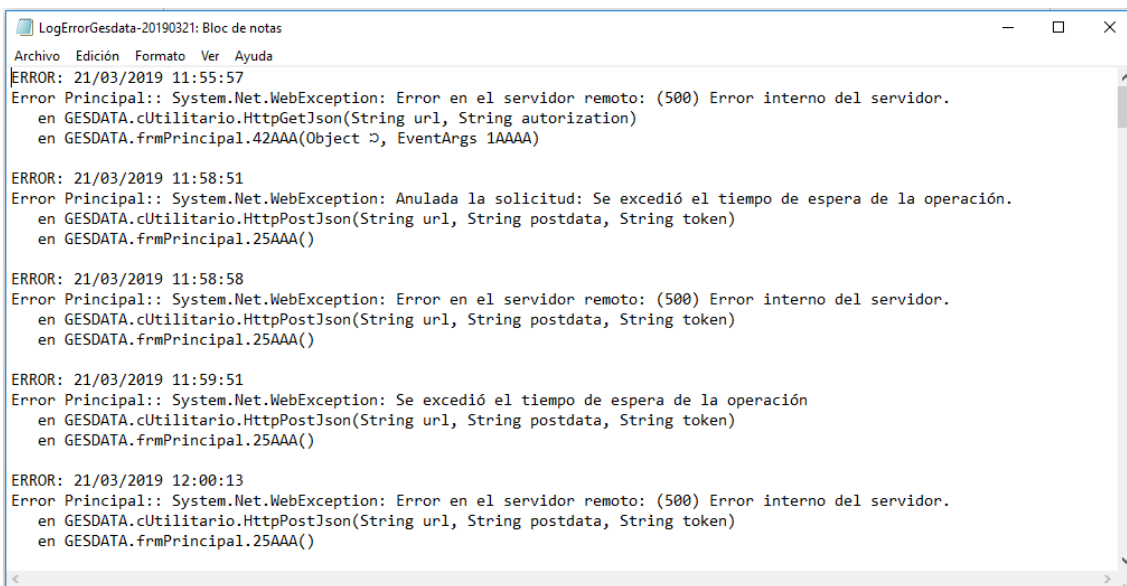
Figura 16. Notificación: No se esta siguiendo con el patrón.

LOG DE ERRORES

La aplicación genera archivos de registro de errores de manera diaria, su ubicación se encuentra en la carpeta de instalación del Agente con el siguiente nombre LogErrorGesdata-aaaaMMdd.



Este es un ejemplo del archivo de registro de errores.



ANEXO 6. ACTA DE CIERRE DEL PROYECTO

	GESTION DE PROYECTOS	CODIGO	
	ACTA DE CIERRE DE PROYECTO	VERSION	1
		FECHA	30/11/2019

Por medio de la presente, se da cierre formal al proyecto Análisis y Desarrollo de Sistema Agente.

1. INFORMACION DEL PROYECTO

Proyecto	Análisis y Desarrollo de Sistema Agente		
Responsable del proyecto			
Fecha de Inicio	07/01/2019	Fecha de Fin	08/05/2019

2. PROPOSITOS Y OBJETIVOS

Implementar un Sistema Agente que permita el registro automático de las tareas realizadas por los operadores.

3. ALCANCES

- La solución debe contar con un front-end para realizar la implementación de uno o varios patrones.
- La solución debe contar con un front-end desde donde se pueda parametrizar un nuevo flujo o realizar cambios en la parametrización de un flujo implementado
- La solución debe realizar el conteo de transacciones y registro de los tiempos de las tareas ejecutadas por el usuario en su PC con el apoyo con una aplicación, utilitario, correo

electrónico y explorador de Windows. La captura de la información debe ser registrada en tiempo real.

- La herramienta deberá contar con un módulo de reportería y consulta para su administración.
- Se requiere que la solución tecnológica cuente con niveles de acceso (perfilamiento de usuarios) que permita administrar los módulos y opciones de la herramienta.
- Se requiere que la solución tecnológica permita el monitoreo de los agentes instalados en los terminales de los usuarios

4. PRINCIPALES ENTREGABLES

N°	Descripción
1	Acta de constitución de servicios
2	Product Backlog
3	Cronograma
4	Manual de Instalación
5	Acta de cierre del Proyecto

ANEXO 7. INFORME DE TURNITIN

ANÁLISIS, DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA AGENTE PARA LA TOMA DE DATOS EN EL ÁREA DE OPERACIONES DE UNA ENTIDAD FINANCIERA.

INFORME DE ORIGINALIDAD

19%	12%	1%	15%
INDICE DE SIMILITUD	FUENTES DE INTERNET	PUBLICACIONES	TRABAJOS DEL ESTUDIANTE

FUENTES PRIMARIAS

1	Submitted to Universidad Tecnologica del Peru Trabajo del estudiante	3%
2	Submitted to Universidad Cesar Vallejo Trabajo del estudiante	2%
3	repositorio.unap.edu.pe Fuente de Internet	1%
4	dspace.unitru.edu.pe Fuente de Internet	1%
5	docplayer.es Fuente de Internet	1%
6	Submitted to Universidad Estatal a Distancia Trabajo del estudiante	1%
7	Submitted to Universidad de Ciencias y Humanidades Trabajo del estudiante	1%
8	Submitted to Universidad Continental	

	Trabajo del estudiante	1%
9	repositorio.utp.edu.pe Fuente de Internet	<1%
10	www.dsic.upv.es Fuente de Internet	<1%
11	www.safaribooksonline.com Fuente de Internet	<1%
12	Submitted to Universidad de Jaén Trabajo del estudiante	<1%
13	Submitted to Universitat Politècnica de València Trabajo del estudiante	<1%
14	Submitted to INACAP Trabajo del estudiante	<1%
15	cursos.itgam.nibira.com Fuente de Internet	<1%
16	m.monografias.com Fuente de Internet	<1%
17	es.scribd.com Fuente de Internet	<1%
18	Submitted to Universidad Internacional Isabel I de Castilla Trabajo del estudiante	<1%

Submitted to Universidad Internacional de la

19	Rioja Trabajo del estudiante	<1%
20	repositorio.ug.edu.ec Fuente de Internet	<1%
21	Submitted to Universidad Autónoma de Ica Trabajo del estudiante	<1%
22	Submitted to Universidad de Nebrija Trabajo del estudiante	<1%
23	Submitted to Universidad San Ignacio de Loyola Trabajo del estudiante	<1%
24	Submitted to Pontificia Universidad Católica del Perú Trabajo del estudiante	<1%
25	Submitted to Universidad Católica de Santa María Trabajo del estudiante	<1%
26	platzi.com Fuente de Internet	<1%
27	Submitted to Universidad EAFIT Trabajo del estudiante	<1%
28	Submitted to Universidad Andina del Cusco Trabajo del estudiante	<1%
29	eprints.soton.ac.uk Fuente de Internet	<1%

30	Submitted to Universidad Tecnológica Israel Trabajo del estudiante	<1%
31	repositorio.ufrn.br:8080 Fuente de Internet	<1%
32	Submitted to BENEMERITA UNIVERSIDAD AUTONOMA DE PUEBLA BIBLIOTECA Trabajo del estudiante	<1%
33	Submitted to Universidad Privada de Tacna Trabajo del estudiante	<1%
34	Submitted to Jose Maria Vargas University Trabajo del estudiante	<1%
35	repositorio.upao.edu.pe Fuente de Internet	<1%
36	repositorio.continental.edu.pe Fuente de Internet	<1%
37	www.avermedia.es Fuente de Internet	<1%
38	www.miembrosprodigy.com.mx Fuente de Internet	<1%
39	www.planetacodigo.com Fuente de Internet	<1%
40	www.kyberne.cl Fuente de Internet	<1%

41	Submitted to Universidad Ricardo Palma Trabajo del estudiante	<1%
42	Submitted to Universidad Europea de Madrid Trabajo del estudiante	<1%
43	oa.upm.es Fuente de Internet	<1%
44	www.manualslib.com Fuente de Internet	<1%
45	www.portalfox.com Fuente de Internet	<1%
46	biblioteca.unmsm.edu.pe Fuente de Internet	<1%
47	Submitted to Universidad de Lima Trabajo del estudiante	<1%
48	www.scribd.com Fuente de Internet	<1%
49	elguille.info Fuente de Internet	<1%
50	repositorio.uncp.edu.pe Fuente de Internet	<1%
51	www.ascofame.org.co Fuente de Internet	<1%
52	dspace.ucuenca.edu.ec Fuente de Internet	

		<1%
53	www.intenso.co.uk Fuente de Internet	<1%
54	es.slideshare.net Fuente de Internet	<1%
55	J. Aguiar, Talal Almahayni, M. Anagnostakis, B. Angjusheva et al. "List of contributors", Elsevier BV, 2017 Publicación	<1%
56	www.diariolaprensa.cl Fuente de Internet	<1%
57	Submitted to Universidad Peruana de Ciencias Aplicadas Trabajo del estudiante	<1%
58	Submitted to Universidad Pedagogica y Tecnologica de Colombia Trabajo del estudiante	<1%
59	Submitted to Colegio Vista Hermosa Trabajo del estudiante	<1%
60	Submitted to Universidad ESAN -- Escuela de Administración de Negocios para Graduados Trabajo del estudiante	<1%
61	www.minem.gob.pe Fuente de Internet	<1%

62	contadoresbc.org Fuente de Internet	<1 %
63	80.38.172.227 Fuente de Internet	<1 %
64	Submitted to Universidad Peruana de Las Americas Trabajo del estudiante	<1 %
65	Submitted to Escuela Politecnica Nacional Trabajo del estudiante	<1 %
66	Submitted to Universidad Dr. José Matías Delgado Trabajo del estudiante	<1 %
67	Submitted to ECCI Trabajo del estudiante	<1 %

Excluir citas

Apagado

Excluir coincidencias

Apagado

Excluir bibliografía

Apagado