



**UNIVERSIDAD TÉCNICA DE AMBATO**  
**FACULTAD DE TECNOLOGÍAS DE LA INFORMACIÓN,  
TELECOMUNICACIONES E INDUSTRIAL**  
**CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES E  
INFORMATICOS**

TEMA:

---

**SISTEMA OPEN SOURCE CON CONEXIÓN A LA PLATAFORMA  
CLOUDIOT PARA EL MONITOREO DE PROGRAMAS  
INFORMÁTICOS EN LOS LABORATORIOS DE LA FACULTAD  
DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL.**

---

Trabajo de Graduación. Modalidad: Proyecto de Investigación, presentado previo la obtención del título de Ingeniero en Sistemas Computacionales e Informáticos.

**LINEA DE INVESTIGACION: Desarrollo de Software**

**AUTOR: Tandazo Quichimbo Paul Andres**

**TUTOR: Ing. Nogales Portero Rubén Eduardo Mg.**

Ambato - Ecuador

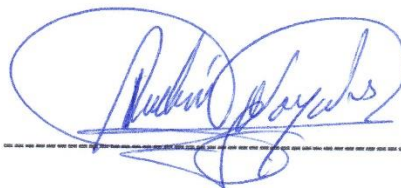
Junio 2019

## **APROBACIÓN DEL TUTOR**

En mi calidad de Tutor del Trabajo de Investigación sobre el tema: “SISTEMA OPEN SOURCE CON CONEXIÓN A LA PLATAFORMA CLOUDIOT PARA EL MONITOREO DE PROGRAMAS INFORMÁTICOS EN LOS LABORATORIOS DE LA FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL”, del señor Tandazo Quichimbo Paul Andres, estudiante de la Carrera de Ingeniería en Sistemas Computacionales e Informáticos, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, considero que el informe investigativo reúne los requisitos suficientes para que continúe con los trámites y consiguiente aprobación de conformidad con el numeral 7.2 de los Lineamientos Generales para la aplicación de Instructivos de las Modalidades de Titulación de las Facultades de la Universidad Técnica de Ambato.

Ambato, junio 2019

EL TUTOR




Ing. Mg. Rubén Eduardo Nogales Portero

## AUTORÍA

El presente Proyecto de Investigación titulado: "SISTEMA OPEN SOURCE CON CONEXIÓN A LA PLATAFORMA CLOUDIOT PARA EL MONITOREO DE PROGRAMAS INFORMÁTICOS EN LOS LABORATORIOS DE LA FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL", es absolutamente original, auténtico y personal, en tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, junio 2019



Paul Andres Tandazo Quichimbo

CC: 180539619-7

## **DERECHOS DE AUTOR**

Autorizo a la Universidad Técnica de Ambato, para que haga uso de este Trabajo de Titulación como un documento disponible para la lectura, consulta y procesos de investigación. Cedo los derechos de mi Trabajo de Titulación, con fines de difusión pública, además autorizo su reproducción dentro de las regulaciones de la Universidad.

Ambato, junio 2019



Paul Andres Tandazo Quichimbo

CC: 180539619-7

## APROBACIÓN DE LA COMISIÓN CALIFICADORA

La Comisión Calificadora del presente trabajo conformada por los señores docentes Ing. Carlos Núñez e Ing. Hernán Naranjo, revisó y aprobó el Informe Final del Proyecto de Investigación titulado “SISTEMA OPEN SOURCE CON CONEXIÓN A LA PLATAFORMA CLOUDIOT PARA EL MONITOREO DE PROGRAMAS INFORMÁTICOS EN LOS LABORATORIOS DE LA FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL”, presentado por el señor Paul Andres Tandazo Quichimbo de acuerdo al numeral 9.1 de los Lineamientos Generales para la aplicación de Instructivos de las Modalidades de Titulación de las Facultades de la Universidad Técnica de Ambato.




---

Ing. Mg. Elsa Pilar Urrutia Urrutia  
PRESIDENTA DEL TRIBUNAL



---

Ing. Mg. Carlos Núñez  
DOCENTE CALIFICADOR



---

Ing. Mg. Hernán Naranjo  
DOCENTE CALIFICADOR

## **DEDICATORIA**

Dedico el presente trabajo a mi madre Maritza, mi padre Ángel y a mi hermano Francisco porque con su permanente apoyo me impulsaron a finalizar este proceso fundamental para mi vida.

Paul Andres Tandazo Quichimbo

## **AGRADECIMIENTO**

Agradezco eternamente a Dios por permitirme iniciar y terminar un camino lleno de satisfacciones, a todos los docentes de la FISEI por sus valiosas enseñanzas durante más de 9 semestres.

Y de manera especial, a mi tutor Ing. Rubén Nogales, quien con su paciencia y amplio conocimiento supo guiar la correcta culminación del presente trabajo de investigación.

Paul Andres Tandazo Quichimbo

## Índice

<b>APROBACIÓN DEL TUTOR</b> .....	ii
<b>AUTORÍA</b> .....	iii
<b>DERECHOS DE AUTOR</b> .....	iv
<b>APROBACIÓN DE LA COMISIÓN CALIFICADORA</b> .....	v
<b>DEDICATORIA</b> .....	vi
<b>AGRADECIMIENTO</b> .....	vii
<b>CAPITULO 1 EL PROBLEMA</b> .....	1
<b>1.1 Tema de Investigación</b> .....	1
<b>1.2 Planteamiento del Problema</b> .....	1
<b>1.3 Delimitación</b> .....	2
<b>1.4 Justificación</b> .....	2
<b>1.5 Objetivos</b> .....	3
<b>1.5.1 General</b> .....	3
<b>1.5.2 Objetivos Específicos</b> .....	3
<b>CAPITULO 2 MARCO TEORICO</b> .....	5
<b>2.1 Antecedentes investigativos</b> .....	5
<b>2.2 Fundamentación Teórica</b> .....	8
<b>2.2.1 Aplicación Informática</b> .....	8
<b>2.2.2 Actividades Informáticas</b> .....	9
<b>2.2.3 Proceso</b> .....	9
<b>2.2.3.1 Estados de un proceso</b> .....	9
<b>2.2.3.2 Bloque de control de proceso</b> .....	10
<b>2.2.4 Software de monitoreo</b> .....	11
<b>2.2.5 Tráfico de red</b> .....	11
<b>2.2.6 Servidor</b> .....	12
<b>2.2.7 Base de datos</b> .....	13
<b>2.2.8 Cloud</b> .....	13



2.2.9	Almacenamiento en la nube (DBaaS).....	13
2.2.10	Reportes .....	13
2.2.11	Informes detallados .....	14
2.2.12	Sub-informes.....	14
2.2.13	Modelos de datos .....	14
2.2.14	Análisis de datos .....	14
2.2.15	Metodología.....	15
2.2.16	Conectividad .....	15
2.2.17	Metodologías de desarrollo de software.....	15
2.2.18	Metodologías tradicionales .....	15
2.2.18.1	Metodología en Cascada.....	16
2.2.18.2	Modelo Prototipo .....	16
2.2.18.3	Modelo Incremental .....	17
2.2.18.4	Modelo Espiral.....	17
2.2.19	Metodologías ágiles.....	18
2.2.19.1	Metodología Scrum.....	18
2.2.19.2	Metodología XP .....	19
2.3	Propuesta de solución.....	20
<b>CAPITULO 3 METODOLOGÍA .....</b>		<b>21</b>
3.1	Modalidad de la Investigación.....	21
3.2	Población y Muestra.....	21
3.3	Recolección de Información.....	21
3.4	Procesamiento y Análisis de Datos.....	22
3.5	Desarrollo del Proyecto.....	23
<b>CAPÍTULO 4 DESARROLLO DE LA PROPUESTA .....</b>		<b>24</b>
4.1	Análisis de factibilidad.....	24
4.1.1	Factibilidad operativa .....	24
4.1.2	Factibilidad económica.....	24
4.1.3	Factibilidad técnica .....	25
4.1.3.1	Factibilidad en software .....	25
4.1.3.2	Factibilidad en hardware .....	26
4.2	Metodología por aplicar.....	26
4.2.1	Metodología Extreme Programming.....	28
4.2.1.1	Valores de Extreme Programming.....	28
4.2.1.2	Fases de Extreme Programming .....	29

<b>4.3</b>	<b>Ejecución del proyecto</b> .....	<b>31</b>
<b>4.3.1</b>	<b>Planificación del proyecto</b> .....	<b>31</b>
<b>4.3.1.1</b>	<b>Roles del proyecto</b> .....	<b>31</b>
<b>4.3.1.2</b>	<b>Historias de usuario</b> .....	<b>32</b>
<b>4.3.1.3</b>	<b>Iteraciones</b> .....	<b>52</b>
<b>4.3.1.4</b>	<b>Estimación de Historias de Usuario</b> .....	<b>53</b>
<b>4.3.1.4.1</b>	<b>Módulo de Monitoreo de Escritorio</b> .....	<b>53</b>
<b>4.3.1.4.2</b>	<b>Módulo de Monitoreo para la Web</b> .....	<b>54</b>
<b>4.3.1.4.3</b>	<b>Modulo Web (Reportes)</b> .....	<b>54</b>
<b>4.3.2</b>	<b>Diseño</b> .....	<b>55</b>
<b>4.3.2.1</b>	<b>Tarjetas CRC</b> .....	<b>55</b>
<b>4.3.3</b>	<b>Desarrollo</b> .....	<b>59</b>
<b>4.3.3.1</b>	<b>Base de Datos</b> .....	<b>59</b>
<b>4.3.3.2</b>	<b>Generación de Actividades Informáticas</b> .....	<b>63</b>
<b>4.3.3.3</b>	<b>Almacenamiento de respaldo</b> .....	<b>63</b>
<b>4.3.3.4</b>	<b>Almacenamiento de datos</b> .....	<b>65</b>
<b>4.3.3.5</b>	<b>Codificación</b> .....	<b>66</b>
<b>4.3.3.5.1</b>	<b>Primera Iteración</b> .....	<b>66</b>
<b>4.3.3.5.2</b>	<b>Segunda Iteración</b> .....	<b>69</b>
<b>4.3.3.5.3</b>	<b>Tercera Iteración</b> .....	<b>71</b>
<b>4.3.3.5.4</b>	<b>Cuarta Iteración</b> .....	<b>73</b>
<b>4.3.4</b>	<b>Pruebas</b> .....	<b>74</b>
	<b>CAPITULO 5 CONCLUSIONES Y RECOMENDACIONES</b> .....	<b>83</b>
<b>5.1</b>	<b>Conclusiones</b> .....	<b>83</b>
<b>5.2</b>	<b>Recomendaciones</b> .....	<b>84</b>
	<b>Bibliografía</b> .....	<b>85</b>
	<b>Anexos y Apéndices</b> .....	<b>89</b>

## Índice de Tablas

Tabla 1: Tipos de Servidores.....	12
Tabla 2: Principales características de Visual Studio y NetBeans .....	25
Tabla 3: Comparación de Metodologías Tradicionales y Agiles .....	26
Tabla 4: Ranking de Agilidad .....	28
Tabla 5: Roles definidos dentro del proyecto .....	31
Tabla 6: Historias de Usuario definidas por el cliente .....	32
Tabla 7: Actividades por Historia de Usuario.....	33
Tabla 8: Historia de Usuario – Capturar actividades informáticas en tiempo real ....	34
Tabla 9: Actividad 1 – Historia de Usuario 1 – Capturar de actividades informáticas .....	34
Tabla 10: Actividad 2 – Historia de Usuario 1 – Capturar de actividades informáticas .....	35
Tabla 11: Actividad 3 – Historia de Usuario 1 – Capturar de actividades informáticas .....	35
Tabla 12: Actividad 4 – Historia de Usuario 1 – Capturar de actividades informáticas .....	36
Tabla 13: Actividad 5 – Historia de Usuario 1 – Capturar de actividades informáticas .....	36
Tabla 14: Historia de Usuario – Almacenamiento de datos en el Cloud .....	37
Tabla 15: Actividad 1 – Historia de Usuario 2 – Almacenamiento de datos en Cloud .....	38
Tabla 16: Actividad 2 – Historia de Usuario 2 – Almacenamiento de datos en Cloud .....	38
Tabla 17: Actividad 3 – Historia de Usuario 2 – Almacenamiento de datos en Cloud .....	39
Tabla 18: Historia de Usuario – Implementación de respaldos .....	39
Tabla 19: Actividad 1 – Historia de Usuario 3 – Implementación de respaldos.....	40
Tabla 20: Actividad 2 – Historia de Usuario 3 – Implementación de respaldos.....	40
Tabla 21: Actividad 3 – Historia de Usuario 3 – Implementación de respaldos.....	41
Tabla 22: Historia de Usuario – Transferencia de los datos respaldados al servidor	41
Tabla 23: Actividad 1 – Historia de Usuario 4 – Transferencia de los datos respaldados al servidor.....	42

Tabla 24: Actividad 2 – Historia de Usuario 4 – Transferencia de los datos respaldados al servidor.....	42
Tabla 25: Actividad 3 – Historia de Usuario 4 – Transferencia de los datos respaldados al servidor.....	43
Tabla 26: Actividad 4 – Historia de Usuario 4 – Transferencia de los datos respaldados al servidor.....	43
Tabla 27: Historia de Usuario – Reconocimiento e identificación de páginas web visitadas.....	44
Tabla 28: Actividad 1 – Historia de Usuario 5 – Reconocimiento e identificación de páginas web visitadas.....	45
Tabla 29: Actividad 2 – Historia de Usuario 5 – Reconocimiento e identificación de páginas web visitadas.....	45
Tabla 30: Historia de Usuario – Almacenamiento de datos web en el Cloud.....	46
Tabla 31: Actividad 1 – Historia de Usuario 6 – Almacenamiento de datos web en el Cloud.....	46
Tabla 32: Actividad 2 – Historia de Usuario 6 – Almacenamiento de datos web en el Cloud.....	47
Tabla 33: Historia de Usuario – Generación de Reportes.....	47
Tabla 34: Actividad 1 – Historia de Usuario 7 – Generación de Reportes.....	48
Tabla 35: Actividad 2 – Historia de Usuario 7 – Generación de Reportes.....	49
Tabla 36: Actividad 3 – Historia de Usuario 7 – Generación de Reportes.....	49
Tabla 37: Actividad 4 – Historia de Usuario 7 – Generación de Reportes.....	50
Tabla 38: Historia de Usuario – Integración con la Plataforma CloudIoT .....	50
Tabla 39: Actividad 1 – Historia de Usuario 8 – Integración con la Plataforma CloudIoT .....	51
Tabla 40: Actividad 2 – Historia de Usuario 8 – Integración con la Plataforma CloudIoT .....	51
Tabla 41: Historias de Usuario por Iteración .....	52
Tabla 42: Estimación del módulo de Monitoreo de Escritorio .....	53
Tabla 43: Estimación del módulo de Monitoreo para la Web .....	54
Tabla 44: Estimación del módulo Web.....	54
Tabla 45: Tarjeta CRC – Capturar actividades informáticas en tiempo real .....	55
Tabla 46: Tarjeta CRC – Almacenamiento de datos en el Cloud .....	55

Tabla 47: Tarjeta CRC – Implementación de Respaldos .....	56
Tabla 48: Tarjeta CRC – Transferencia de los datos respaldados al servidor .....	56
Tabla 49: Tarjeta CRC – Reconocimiento e de Páginas Web visitadas .....	57
Tabla 50: Tarjeta CRC – Almacenamiento de datos en el Cloud .....	58
Tabla 51: Tarjeta CRC – Generación de Reportes.....	58
Tabla 52: Tarjeta CRC – Integración con la Plataforma CloudIoT .....	58
Tabla 53: Programas solicitados para horas clase.....	61
Tabla 54: Comparación entre SQLite – CSV.....	64
Tabla 55: Servidor de Base de datos Plataforma CloudIoT .....	65
Tabla 56: Propiedades de Win32_ProcessStartTrace y Win32_ProcessStopTrace...	66
Tabla 57: Servidor Web Plataforma CloudIoT .....	73
Tabla 58: Prueba de aceptación 1 – Ejecución en segundo plano.....	74
Tabla 59: Prueba de aceptación 2 – Obtener inicio de las aplicaciones ejecutadas...	75
Tabla 60: Prueba de aceptación 3 – Obtener cierre de aplicaciones ejecutadas .....	75
Tabla 61: Prueba de aceptación 4 – Ingreso de datos con conexión.....	76
Tabla 62: Prueba de aceptación 5 - Seleccionar alternativa para respaldos.....	77
Tabla 63: Prueba de aceptación 6 – Ingreso de datos sin conexión.....	78
Tabla 64: Prueba de aceptación 7 - Almacenamiento de datos guardados fuera de línea.....	79
Tabla 65: Prueba de aceptación - Obtener datos generados desde navegadores web	79
Tabla 66: Prueba de aceptación 9 - Ingresar de datos web con conexión.....	80
Tabla 67: Prueba de aceptación 10 - Mostrar gráficos de los datos obtenidos .....	81
Tabla 68: Prueba de aceptación 11 - Paginas disponibles en el sitio de la Plataforma .....	82
Tabla 69: Comparativa de herramientas de monitoreo comerciales .....	92

## Índice de Figuras

Figura 1: Secciones de la Memoria de un Proceso .....	9
Figura 2: Diferentes estados para los procesos .....	10
Figura 3: Bloque de control de proceso .....	11
Figura 4: Fases de la Metodología Tradicional.....	16
Figura 5: Pasos del Modelo Incremental.....	17
Figura 6: Pasos del modelo espiral.....	18
Figura 7: Proceso de la Metodología XP .....	19
Figura 8: Flujograma del proceso general para la conseguir los datos .....	22
Figura 9: Cuadro de presupuesto.....	24
Figura 10: Tablas para almacenar los datos monitoreados .....	59
Figura 11: Tabla para el Monitoreo de sitios .....	60
Figura 12: Tabla para el Software Autorizado.....	60
Figura 13: Diagrama completo de la Plataforma CloudIoT.....	62
Figura 14: Trama de datos a capturar en el monitoreo de escritorio.....	63
Figura 15: Trama de datos a capturar en el monitoreo de web .....	63
Figura 16: Imagen de Rangos de crecimiento en Memoria de Proceso para SQLite	65
Figura 17: Imagen de Rangos de crecimiento en Memoria de Proceso para CSV ....	65
Figura 18: Prototipo de interfaz de usuario general del sistema web .....	72
Figura 19: Árbol de Problemas .....	90
Figura 20: Esquema general de monitoreo.....	91
Figura 21: Acta de conformidad - Laboratorio Redes 1 .....	95
Figura 22: Acta de conformidad - Laboratorio 7 .....	95
Figura 23: Acta de conformidad - Laboratorio 6 .....	96
Figura 24: Acta de conformidad - Laboratorio 5 .....	96
Figura 25: Acta de conformidad - Laboratorio 4 .....	97
Figura 26: Acta de conformidad - Laboratorio 3 .....	97
Figura 27: Acta de conformidad - Laboratorio 1 .....	98
Figura 28: Acta de conformidad - Laboratorio 1 .....	98
Figura 29: Acta de conformidad - Laboratorio 1 .....	99
Figura 30: Acta de conformidad - Laboratorio 1 .....	99
Figura 31: Acta de conformidad - Laboratorio 2 .....	100
Figura 32: Requerimientos de software - Carrera Electrónica.....	100

Figura 33: Requerimientos de software - Telecomunicaciones Electrónica.....	101
Figura 34: Requerimientos de software – Electrónica .....	101
Figura 35: Requerimientos de software – Electrónica .....	102
Figura 36: Requerimientos de software - Electrónica.....	102
Figura 37: Diagrama de flujo - Módulos de Monitoreo.....	103
Figura 38: Asistente de Instalación 1 .....	105
Figura 39: Asistente de Instalación 2 .....	106
Figura 40: Asistente de Instalación 3 .....	106
Figura 41: Asistente de Instalación 4 .....	107
Figura 42: Asistente de Instalación 5 .....	107
Figura 43: Lista de procesos en segundo plano .....	108
Figura 44: Uso de software por carrera.....	108
Figura 45: Uso de software por laboratorio .....	109
Figura 46: Uso de software por docente .....	109
Figura 47: Uso de software autorizado y no autorizado .....	110
Figura 48: Tiempo de uso utilizado por programa no autorizado .....	110
Figura 49: Análisis de datos.....	111

## **Resumen Ejecutivo**

Dentro del proyecto de Investigación macro denominado “**PLATAFORMA CLOUDIOT DE CONTROL Y MONITOREO DEL USO DE EQUIPAMIENTO Y PROGRAMAS INFORMÁTICOS EN AULAS Y LABORATORIOS**” nace como problemática el inadecuado uso que tienen los programas informáticos instalados en los equipos de la facultad.

Una vez reconocida la dimensión real del problema, se ha optado por implementar un sistema de monitoreo como una herramienta de solución efectiva y que no represente mayores inconvenientes en la carga presupuestaria de la universidad.

El sistema desarrollado se debe adaptar como la fase 3 del proyecto macro y netamente estará enfocado a capturar los datos que se generan a partir de la ejecución de cualquier aplicación informática. A su vez, los módulos enviarán los datos al servidor en el que reside el proyecto o en su defecto almacenarán las actividades en archivos fuera de línea.

El sistema de monitoreo a más de capturar datos está orientado para adicionar otros campos que son contantes por equipo como nombre, usuario, etc. La información que se va generando en un equipo se filtra y organiza en filas de tal manera que bajo todo concepto se garantiza la integridad y validez de los datos monitoreados.

Al implementar la propuesta de la solución, es factible determinar que programas se ejecutan en una determinada hora clase, y en base a esa información se puede establecer el nivel de uso que tienen las aplicaciones solicitadas por los docentes al inicio de cada semestre.



## **Abstract**

Inside of the macro research project called “CLOUDIOT PLATFORM FOR CONTROL AND MONITORING THE USE OF EQUIPMENT AND COMPUTER PROGRAMS IN CLASSROOMS AND LABORATORIES” was detected as problematic the inadequate use of computer programs in the teams of the faculty.

Once the real dimension of the problem is recognized, it has been decided to implement a monitoring system as an effective solution tool that does not represent major drawbacks in the university's budget burden.

The developed system must be adapted as phase 3 of the macro project and will be clearly focused on capturing the data generated from the execution of any computer application. The modules send the data to the server where the project resides or otherwise, they store the activities in offline files.

The monitoring system is oriented to add other fields that are constant by computer such as name, user, etc. The information that is generated in a team is filtered and organized in rows in such a way that the integrity and validity of the monitored data is always guaranteed.

After implementing the solution proposal, it is feasible to determine which programs are executed in a certain class hour, and based on this information, it is possible to establish the level of use of applications requested by teachers at the beginning of each semester.

## Glosario de términos y acrónimos

- CRC: Clase - Responsabilidad – Colaboración.
- XP: Programación Extrema.
- IDE: Entorno de desarrollo integrado.
- Visual Studio: IDE para desarrollar software.
- SQL: Lenguaje de programación estándar, utilizado por los motores de base de datos para recuperar, guardar, modificar información.
- CSV: Archivo delimitado por comas.
- DataTable: Representa una tabla de datos en memoria.
- MySQL: Base de datos.
- MySqlConnection: Librería ADO.NET completamente diseñada para MySQL.
- SQLite: Pequeña base de datos, usada por dispositivos móviles y ciertas aplicaciones para mantener datos fuera de línea.
- Apache: Servidor web.
- Centos: Sistema operativo, distribución de Linux
- Windows: Sistema operativo.
- .exe: Ejecutable en Windows.

## INTRODUCCIÓN

El desarrollo del presente trabajo de investigación denominado **SISTEMA OPEN SOURCE CON CONEXIÓN A LA PLATAFORMA CLOUDIOT PARA EL MONITOREO DE PROGRAMAS INFORMÁTICOS EN LOS LABORATORIOS DE LA FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL**, se encuentra dividido en capítulos para facilitar la comprensión de todo el contenido que fue generado durante el desarrollo de la propuesta de solución:

**Capítulo I:** Como punto de partida en el primer capítulo, denominado “EL PROBLEMA” se plantea como problemática a investigar el inadecuado uso de los programas informáticos utilizados para horas clase en los laboratorios de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, a la vez se realiza la justificación de la investigación y los objetivos que se conseguirán durante el desarrollo de esta.

**Capítulo II:** denominado “MARCO TEORICO”, contiene los antecedentes investigativos necesarios para sustentar toda la investigación y plantear la propuesta de solución.

**Capítulo III:** durante el tercer capítulo, denominado “METODOLOGIA” se establece la modalidad básica que se utilizara en la investigación, como se recolectara la información de utilidad, la manera para abarcar el procesamiento y análisis de datos, y finalmente se definen las etapas o pasos a cumplir para cumplir los objetivos.

**Capítulo IV:** el cuarto capítulo, denominado “DESARROLLO DE LA PROPUESTA” esta totalmente direccionado al desarrollo del software en base a una determinada metodología.

**Capítulo V:** finalmente en el último capítulo se definen las “CONCLUSIONES Y RECOMENDACIONES” obtenidas tras desarrollar, probar y entregar la propuesta de solución.

# **CAPITULO 1**

## **EL PROBLEMA**

### **1.1 Tema de Investigación**

Sistema Open Source con conexión a la Plataforma CloudIoT para el monitoreo de programas informáticos en los laboratorios de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial.

### **1.2 Planteamiento del Problema**

El uso de la informática en el mundo educativo ha evolucionado, a medida que los países en vías de desarrollo han puesto su enfoque en mejorar la calidad en los sistemas de educación [1]. En este contexto, en el Ecuador la mayoría de centros de educación han optado por el empleo o adquisición de aplicaciones de software (TI), para contribuir como herramientas de apoyo en la metodología de enseñanza utilizada por cada docente.

A pesar de que en nuestro país rige el decreto 1425 que recomienda el uso de estándares abiertos y software libre [2], varias instituciones de educación superior, han optado por destinar una parte de su presupuesto en la adquisición de licencias de programas informáticos, para determinadas actividades en el campo educativo, este ambicioso esfuerzo económico se ve limitado principalmente por la información generada sobre el impacto que tiene el uso del software en los resultados educativos y de productividad. La escasa información no denota claramente cómo está siendo utilizada por los estudiantes la inversión realizada por las universidades, un ejemplo palpable de esta situación son las bibliotecas virtuales que tienen un gran costo, pero poca concurrencia por los estudiantes.

El mal uso del software dentro de una entidad educativa desencadena en una desmedida inversión en herramientas que a fin de cuentas no son utilizadas por los estudiantes o que su uso no representa la inversión realizada por el centro de educación superior, adicionalmente se producen problemas de aprendizaje de los

estudiantes generados por la desatención y que en muchos casos provocan la pérdida del semestre. También es relevante tomar en cuenta que para determinadas situaciones el uso de las TIC en la educación no es una garantía clara de que se vaya a mejorar el rendimiento académico [3].

Al igual que en todo el país, la provincia de Tungurahua no es ajena a esta realidad, es así como, de las 5 universidades establecidas en la provincia, aún es muy poca la información generada por estos centros educativos acerca de cómo emplean el software o un tiempo estimado del uso de las diferentes herramientas adquiridas para el periodo académico.

En la Facultad de Ingeniería en Sistemas, Electrónica e Industrial de la Universidad Técnica de Ambato, tampoco existe algún método para conocer en detalle el uso que tienen los productos software instalados en cada uno de los laboratorios de la facultad, que a fin de cuentas son bienes financiados por el estado, e incluso menos se sabe sobre la incidencia que provoca el mal uso de las herramientas instaladas en el rendimiento estudiantil, el único dato conocido es que en el periodo ya culminado (marzo – septiembre 2017) en la carrera de Ingeniería en Sistemas Computacionales e Informáticos se registraron 250 estudiantes reprobados en diferentes módulos, es decir el 14,53% del total de estudiantes matriculados, pero este dato suelto no da a conocer que causas lo llegaron a provocar.

### **1.3 Delimitación**

#### **1.3.1 De contenidos**

**Área Académica:** Software

**Línea de Investigación:** Desarrollo de Software

**Sublínea de Investigación:** Intercambio de Información

#### **1.3.2 Espacial**

Laboratorios de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial.

#### **1.3.3 Temporal**

La presente investigación se desarrollará en los periodos académicos Marzo - Agosto 2018, Septiembre 2018 – Febrero 2019.

### **1.4 Justificación**

Es necesario realizar la presente investigación ya que el problema afecta directamente a los estudiantes que utilizan los equipos informáticos de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, el único fin que se persigue con la elaboración del trabajo de titulación es desarrollar una aplicación informática que permita alimentar a la Plataforma CloudIoT con información sobre el uso que tiene los programas informáticos adquiridos y de software libre instalados en los laboratorios de la facultad.

La información generada por el sistema a desarrollar le permitirá a la plataforma de la Universidad Técnica de Ambato determinar si el inadecuado uso del software en los laboratorios en primera instancia tiene incidencia en los resultados finales de cada estudiante, y también si afecta de alguna manera, a la tasa de retención contemplada en el Modelo de Evaluación Institucional de Universidades y Escuelas Politécnicas 2018 propuesto por el CEAACES. Recordando que de acuerdo con el Modelo se espera que dicha tasa sea del 80%, es decir que al menos ese porcentaje de estudiantes siga cursando sus estudios en la IES.

Finalmente, este trabajo intenta ser de utilidad teórica en el área de Ingeniería en Sistemas Computacionales e Informáticos, buscando promover el uso y construcción de nuevas herramientas de software Open Source, con el único propósito de aportar en la optimización de recursos dentro de la universidad y a la vez seguir el decreto 1425 que actualmente rige en nuestro país y que sugiere el uso de software libre en las entidades públicas.

## **1.5 Objetivos**

### **1.5.1 General**

- Implementar una herramienta Open Source que reporte el uso del software instalado en los laboratorios de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial.

### **1.5.2 Objetivos Específicos**

- Identificar las actividades informáticas que no son afines al horario establecido en los laboratorios de la facultad.

- Desarrollar una solución informática Open Source que registre el uso de las diferentes aplicaciones instaladas en los equipos computacionales de la facultad.
- Representar gráficamente mediante un análisis de datos el uso que tienen las aplicaciones instaladas en los diferentes laboratorios.
- Desarrollar una herramienta que permita ofrecer una mayor disponibilidad en los informes generados a toda la Facultad de Ingeniería en Sistemas, Electrónica e Industrial.

## **CAPITULO 2**

### **MARCO TEORICO**

#### **2.1 Antecedentes investigativos**

Luego de revisar los trabajos de investigación de la biblioteca de la Facultad de Ingeniería en Sistemas Electrónica e Industrial se ha encontrado la investigación realizada por Gonzalo Daniel Ruiz Mesías que están relacionado con los Sistemas de monitoreo y tiene como título “Sistema de control y monitoreo para mejorar los procesos de administración de los laboratorios de las carreras de Sistemas, Electrónica e Industrial en la FISEI - UTA” cuyo propósito se orientaba al planteamiento de una propuesta que permita la implementación de un Sistema de Control y Monitoreo para mejorar los procesos administrativos de los laboratorios. Esta solución buscaba en términos generales resolver el problema que existe para controlar y monitorear los laboratorios de las diferentes carreras de la FISEI, al finalizar el estudio [4] se obtuvo como resultado un mejoramiento en los procesos de administración de los laboratorios, ya que se pasó de un control y monitoreo manual por otro más automatizado que incluía desde comunicación con cámaras IP hasta el uso tarjetas Arduino [4].

Fuera de los repositorios de la Universidad Técnica de Ambato, para el presente trabajo se puede recopilar como antecedente investigativo, el Sistema para monitoreo y control de uso de software en la empresa (SIMCUS), que propone un programa mediante el cual se hace posible monitorear programas asignados que debe utilizar un empleado informático para luego presentar información que refleje su utilización respectiva y posteriormente de ese modo permita llevar un determinado control.

SIMCUS está desarrollado en JAVA por Fernando Galarza como trabajo de titulación para la Universidad Politécnica Salesiana, este software se diseñó para funcionar en Windows XP, hace uso de la base de datos Microsoft Access a la hora de almacenar datos. Además, el sistema implementa una solución ideada en



recolectar datos de los programas que tiene que utilizar un empleado, para luego con esos datos recogidos, calcular y almacenar en la base de datos del sistema los tiempos de ocupación y desocupación cuando se abren dichos programas. Los datos recolectados son los porcentajes de uso de CPU que los programas ocupan cuando son abiertos y utilizados [5].

Referente a investigaciones sobre el monitoreo de equipos informáticos, podemos tomar como un antecedente el trabajo [6], en donde se propuso el monitoreo de normas para sistemas de múltiples agentes, lo que se buscaba con estas normas es que el monitoreo en sí, sea realizado por los agentes involucrados en el sistema, para que de esa forma el monitoreo resuelva la problemática, en la que los agentes pueden publicar contenido que vaya en contra de los intereses del sistema principal. Esta investigación presenta varios resultados: primero se tiene el caso en que se logra un monitoreo perfecto que impide que se violen las normas establecidas, pero también se presenta el otro caso, en donde la aplicación no es perfecta y la probabilidad de violación de norma es relativamente mínima [6].

El mismo autor de la investigación [6] para solucionar el problema del monitoreo en tiempo de ejecución de las normas con monitores imperfectos, es decir cuando un monitor tiene capacidades de observación insuficientes, para determinar si una traza de ejecución determinada de un sistema de agente múltiple cumple o infringe la norma. Como primer paso se procedió a dar un enfoque en el cual se mejoró las capacidades de observación de la organización normativa, sin embargo, esto resulta ser costoso o en algunos casos imposible. Es por ello que en la investigación [7] se propone sintetizar una aproximación de una norma ideal que puede monitorearse perfectamente con un monitor, y que es óptima en el sentido de que cualquier otra aproximación no detectaría al menos tantas violaciones de la norma ideal [7].

En un estudio realizado por Kutare et al., se propone hacer uso de una solución que combine tanto el monitoreo como el análisis de datos para sistemas virtualizados e infraestructuras en la nube (Monalytics). Esta solución a través de una integración de funcionalidad con Xen, busca gestionar los centros de datos a gran escala y, las nubes de servicios de manera eficiente, para lo cual necesariamente se incluyó alguna consideración de escalabilidad en tiempo y espacio. Según Kutare et al., para obtener

el monitoreo y el análisis de datos en múltiples niveles de abstracción se debe buscar la capacidad de ajustar y desplegar dinámicamente la funcionalidad.

Mediante evaluaciones experimentales, se demostró que las acciones pueden ser simples si están integradas con el monitoreo, particularmente cuando se trata de cuestiones de escalabilidad. También los resultados del monitoreo y análisis en línea demostraron su utilidad a través de sus características que iban desde la implementación de bucles de control local como acciones correctivas, instalaciones en diferentes niveles de abstracción o en diferentes subsistemas hasta lograr el principal propósito, una escalabilidad a través del análisis local de datos y conseguir un grado de proporción de fallas para sistemas o tareas mayores [8].

Un antecedente bastante interesante es la investigación realizada por Taylor y He, en donde se propone el monitoreo de recursos de red basado en push, esta investigación se apoya en la evolución de los switch que hoy en día pasaron de admitir solamente el uso del protocolo SNMP, por emplear un sistema operativo extensible que permite implementar un enfoque basado en push, para monitorear una red, esta solución busca resolver los problemas de uso intensivo de datos, que implican la transmisión y el análisis de volúmenes masivos de datos de grandes redes de sensores u otros dispositivos de adquisición, simulaciones o redes sociales. A pesar de que el monitoreo propuesto por [9] resulta innovador, los resultados obtenidos evidencian que la estrategia utilizada es insuficiente en términos de escalabilidad y rendimiento [9].

Dentro de la investigación realizada por Reyson, se propone la construcción de una herramienta para el monitoreo de la administración pública, esta solución informática busca supervisar el avance de cada institución y de los indicadores definidos previamente por el Ministerio de Administraciones Públicas. El trabajo elaborado por [10] en parte está fundamentado por el Banco Interamericano de Desarrollo, ya que el método para distribuir indicadores en la función pública de manera eficiente se inspira en una herramienta desarrollada por este organismo.

Desde que el sistema de monitoreo fue implementado en República Dominicana, se han publicado dos clasificaciones del nivel de cumplimiento de la Ley de Servicio Civil, y a la vez también se ha generado opinión pública sobre la administración

pública y en parte se ha contribuido en que los gerentes públicos cumplan con la ley. La solución desarrollada tuvo estos resultados debido a que el software es fácil de usar y orientado a gráficos, por lo que cualquiera puede usarlo y comprenderlo [10].

Un estudio un tanto más centralizado y direccionado a un tipo de monitoreo en específico lo realiza Cook et al., en la investigación de este autor se propone netamente tratar con aplicaciones científicas o utilizadas para experimentaciones, en donde se busca hacer un monitoreo del lado de la producción de las aplicaciones científicas de manera que sea eficiente y que produzca información utilizable sobre el cálculo, tanto para los científicos como para los administradores del sistema, para de ese modo solucionar el inconveniente en el que se envía una solicitud de ejecución a un recurso computacional, y esta solicitud eventualmente termina, posiblemente falla o se detiene y necesita reiniciarse. Mientras se está ejecutando, no se sabe nada de lo que dicha solicitud está haciendo, aparte de lo que podría aparecer en los archivos de salida que podría estar creando.

Por ello [11] para evitar que los administradores de sistemas de estos grandes recursos encuentren frustrante ni siquiera saber si una aplicación está progresando o no, propone un monitoreo que va del ambiente de control al de producción intentando obtener más información sobre el progreso de una aplicación [11].

## **2.2 Fundamentación Teórica**

### **2.2.1 Aplicación Informática**

“Las aplicaciones son programas diseñados para ejecutar trabajos o procesos de cálculo específicos que precisa el usuario o la unidad empresarial”, más allá de la plataforma en que funcionen las aplicaciones (escritorio, móvil, web, etc.) hay dos tipos para clasificarlas: [12]

- Estándar: “aplicación ya existente en el mercado que responde a las necesidades genéricas del proceso a mecanizar”.
- A medida: “responden a necesidades específicas”.

### 2.2.2 Actividades Informáticas

Se definiría actividad: “Conjunto de operaciones o tareas propias de una persona o entidad” [13]

Informática: “Conjunto de conocimientos científicos y técnicas que hacen posible el tratamiento automático de la información por medio de computadoras.” [14]

Luego de enlazar ambos conceptos se puede asegurar que, una actividad informática es el conjunto de operaciones realizadas por una persona en donde se hace uso de técnicas que hacen factible emplear una computadora para dar tratamiento a cierta información.

### 2.2.3 Proceso

Un proceso es un programa en ejecución, que también puede ser considerado como una entidad activa [15], a la vez los procesos pueden mantener atributos que van desde el estado del hardware, la memoria, CPU, etc. Cada uno de los procesos tiene asignado un ID único hasta que el proceso finaliza su ejecución [16].

Para garantizar un trabajo eficiente un proceso se divide en cuatro secciones, como se muestra en la figura 1.

Seccion de Texto	Seccion de datos	Heap	Stack
<ul style="list-style-type: none"><li>•Codigo del programa compilado que se lee desde el almacenamiento no volatil cuando se inicia el programa.</li></ul>	<ul style="list-style-type: none"><li>•Se compone de las variables globales y estaticas, asignadas e inicializadas antes de ejecutar el principal.</li></ul>	<ul style="list-style-type: none"><li>•Se utiliza para la asignacion de memoria dinamica y es administrado via llamados.</li></ul>	<ul style="list-style-type: none"><li>•Es usado para variables locales. El espacio de la pila se reserva para las variables locales cuando se declaran.</li></ul>

Figura 1: Secciones de la Memoria de un Proceso

Elaborado por: Paul Tandazo

#### 2.2.3.1 Estados de un proceso

A su vez los procesos pueden estar en un estado determinado, tal como se describe en la figura 2:

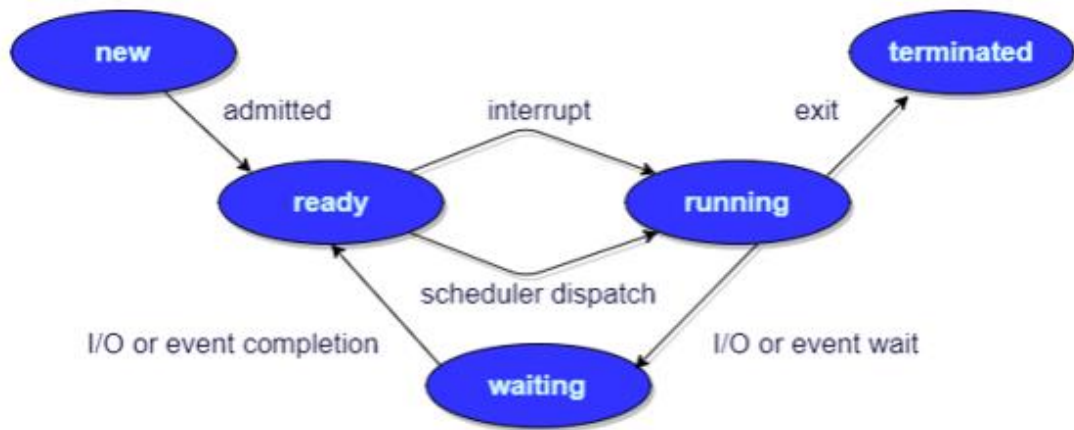


Figura 2: Diferentes estados para los procesos

Elaborado por: [16]

Nuevo (**New**): Cuando el proceso ha sido creado.

Listo (**Ready**): El proceso está a la espera de ser asignado a un procesador.

Ejecutando (**Running**): Las instrucciones están siendo ejecutadas.

Esperando (**Waiting**): El proceso esta esperando por algún evento que debe ocurrir en E/S o la recepción de una señal.

Terminado (**Terminated**): El proceso ha finalizado su ejecución.

### 2.2.3.2 Bloque de control de proceso

Para cada proceso hay un Bloque de Control que se encarga de adjuntar toda la información sobre el proceso [16]. Por cada proceso el bloque contiene un identificador único, estado, prioridad, contador de programa, registros de CPU, información de entrada/salida, información de cuenta y otras propiedades regidas por el sistema operativo. Un PCB tiene una estructura como se describe en la figura 3:



Figura 3: Bloque de control de proceso

Elaborado por: [16]

#### 2.2.4 Software de monitoreo

Un programa de monitoreo es aquel que tiene como único objetivo llevar un registro de cada una de “... las actividades de la computadora y el internet, rastree las horas de trabajo, análisis y estadísticas del uso equipos, detecta el nivel de uso ...” [17].

#### 2.2.5 Tráfico de red

“Comunicaciones de red informática que se transmiten a través de redes cableadas o inalámbricas entre hosts” [18].

En el ámbito de redes el tráfico de redes puede ser: [19]

- Elástico: “Se ajusta a variaciones de retardo y rendimiento, es un tipo de tráfico propio de las redes IP”.
- No elástico: “No se adapta a las variaciones de retardo y rendimiento de la red y necesita optimizar: rendimiento, retardo, jitter (fluctuación del retardo), pérdida de paquetes”.

## 2.2.6 Servidor

“Un servidor es un ordenador u otro tipo de equipo informático encargado de suministrar información a una serie de clientes, que pueden ser tanto personas como otros dispositivos conectados a él. La información que puede transmitir es múltiple y variada: desde archivos de texto, imagen o vídeo y hasta programas informáticos, bases de datos, etc.” [20]

### 2.2.6.1 Tipos de servidor

Dependiendo las necesidades del negocio, en una misma red pueden existir varios tipos de servidores, los más comunes se describen en la siguiente tabla [21]:

Tabla 1: Tipos de Servidores

Elaborado por: Paul Tandazo

Tipo de servidor	Función
<b>Servidor de correo</b>	Se encarga de gestionar el flujo de correo electrónico de los usuarios, envía, recibe y almacena los correos de una organización.
<b>Servidor proxy</b>	Generalmente guarda en memoria caché las páginas web a las que acceden los usuarios de la red durante un cierto tiempo, de esta forma las siguientes veces que estos acceden al mismo contenido, la respuesta es más rápida.
<b>Servidor de Base de Datos</b>	Provee servicios de base de datos a otros programas o equipos cliente.
<b>Servidores DNS</b>	Permite establecer la relación entre los nombres de dominio y las direcciones IP de los equipos de una red.
<b>Servidor de archivos</b>	Almacena y sirve ficheros a equipos de una red.
<b>Servidor Web</b>	Almacena contenido web y lo pone al servicio de aquellos usuarios que lo solicitan.

### **2.2.7 Base de datos**

“Una base de datos es una colección interrelacionada de datos, almacenados en un conjunto sin redundancias innecesarias cuya finalidad es la de servir a una o más aplicaciones de la manera más eficiente.” [22]

### **2.2.8 Cloud**

“Es un concepto abstracto que resume la posibilidad de utilizar herramientas, aplicaciones o sistemas de gestión de manera remota desde la propia internet. Es decir, no se utilizaría un sistema de gestión remoto apuntando a una maquina concreta ni tampoco software local.” [23]

### **2.2.9 Almacenamiento en la nube (DBaaS)**

“Es un modelo de cloud que permite a usuarios realizar requerimientos “On-demand” de ambientes de bases de datos a través de la elección de un catálogo de servicio predefinido utilizando un marco de trabajo de aprovisionamiento de autoservicio.

Los puntos claves que benefician este tipo de modelo son agilidad y más rápido despliegue de servicios de base de datos. Como las bases de datos y los ambientes son aprovisionados y des aprovisionados entonces los recursos informáticos asociados son consumidos y luego puestos en libertad.

Los recursos de bases de datos pueden ser consumidos durante un proyecto y posteriormente des aprovisionados a la agrupación de recursos. Los costos computacionales consumidos son sometidos a un proceso de seguimiento y monitoreo para posteriormente ser cobrados al consumidor.” [24]

### **2.2.10 Reportes**

Los reportes permiten crear informes que contienen datos tabulares, agregados y multidimensionales. A pesar de haber varias herramientas para elaborar reportes en si todos los IDEs de programación manejan el mismo concepto, solamente se varia en ciertas funciones que admite la herramienta de informes dependiendo de la plataforma en que se desarrolle, por ejemplo, los controles ReportViewer de Visual



Studio permiten procesar y mostrar reportes en una aplicación de escritorio o web. [25]

### **2.2.11 Informes detallados**

“Es un informe que los usuarios abren al hacer clic en un vínculo de otro informe. Este tipo de informes suele incluir información detallada acerca de los elementos del informe de resumen original.” [26]

### **2.2.12 Sub-informes**

“Es un elemento de informe que muestra otro informe dentro del cuerpo del informe principal. Cualquier informe puede utilizarse como sub-informe. El informe que se muestra como el sub-informe se almacena en un servidor de informes, normalmente en la misma carpeta que el informe primario.

Es posible diseñar el informe primario para que pase sus parámetros al sub-informe. Este tipo de informe puede repetirse dentro de las regiones de datos mediante un parámetro que filtre los datos de cada instancia del sub-informe.” [27]

### **2.2.13 Modelos de datos**

“El modelo de datos representa la información que se encuentra en una base de datos y cómo se relacionan entre sí los elementos de la base de datos. A través del modelo de datos, puede controlar el aspecto y comportamiento de los campos de datos y realizar la validación automática con poco o nada de código.

Además, se puede personalizar el modelo de datos para configurar cómo se validan los campos de datos y cómo se representa la interfaz de usuario que muestra y modifica los campos de datos.” [28]

### **2.2.14 Análisis de datos**

“Analizar un conjunto de datos quiere decir “verlos”, manipularlos, “volverlos a ver”, y en cierto sentido “concluir” a partir de lo que en ellos “se ve” ... generalmente se da en el contexto de una investigación o estudio que pretende “demostrar” o “evidenciar” algo “sospechoso”.” [29]

### **2.2.15 Metodología**

Un conjunto de procedimientos, técnicas, herramientas, y un soporte documental que ayuda a los desarrolladores a realizar nuevo software, persiguiendo con ello cubrir necesidades que empiezan por mejorar aplicaciones, tendientes a una mejor calidad, procurando un proceso de desarrollo controlado que asegure uso de recursos apropiados y costo adecuado, empleando un proceso estándar en la organización, que no sienta los cambios del personal. [30]

### **2.2.16 Conectividad**

La conectividad es la existencia simultánea de alternativas y vínculos directos entre los distintos puntos de una red. En esta visión, los usuarios, las computadoras y los servidores, entre otros, hacen parte de las redes tecnológicas [31].

### **2.2.17 Metodologías de desarrollo de software**

Una metodología para el desarrollo de software es aquel marco de trabajo que es usado para estructurar, planificar y controlar el proceso que se ocupa para construir un programa en sistemas de información. La metodología es un modo sistemático de realizar, gestionar y administrar un proyecto para llevarlo a cabo con altas posibilidades de éxito minimizando al máximo los riesgos de fracaso.

También es útil resaltar que la metodología comprende actividades a seguir para idear, implementar y mantener un producto de software desde que surge la necesidad del producto hasta que se cumple el objetivo por el cual fue creado [32].

### **2.2.18 Metodologías tradicionales**

Las principales metodologías tradicionales se basan en normas provenientes de estándares seguidos por el entorno de desarrollo [33], están centradas en llevar una documentación exhaustiva de todo el proyecto, cumplir con un plan de proyecto que previamente ha sido definido en la fase inicial y se focalizan en la documentación, planificación y procesos [34].

Desafortunadamente, al ser metodologías que se llevaron de otras ramas al software se detectaron problemas en los altos costos al implementar un cambio y al no ofrecer

una buena solución para proyectos donde el entorno es volátil o muy cambiante [34], problemas que en algunos casos se mantienen hasta hoy.

Con la evolución e impacto del software, las metodologías tradicionales, pasaron a denominarse metodologías no ágiles.

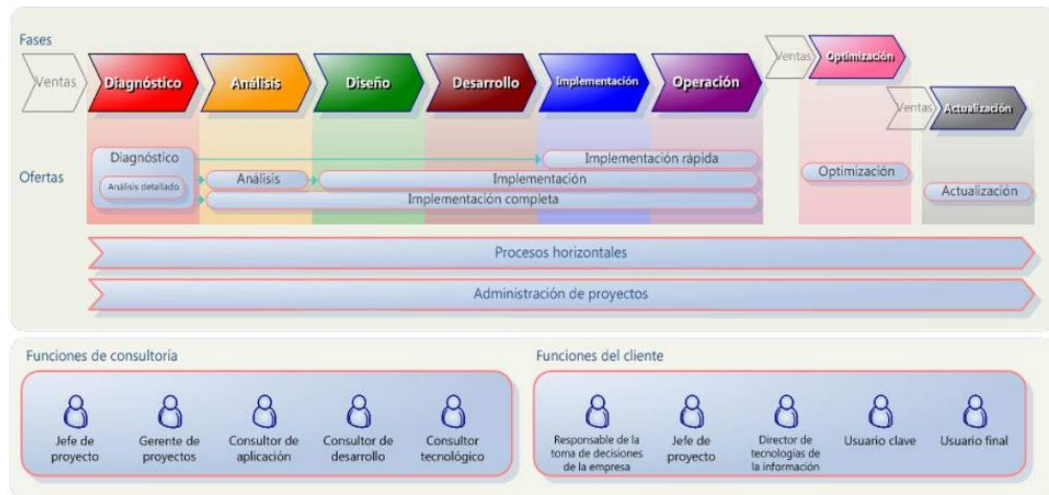


Figura 4: Fases de la Metodología Tradicional

Elaborado por: [35]

### 2.2.18.1 Metodología en Cascada

Es una metodología con un enfoque secuencial que considera etapas esenciales como: planteamiento de requisitos, análisis y diseño del software, implementación, verificación y mantenimiento, como se describe en la siguiente figura:

La gran desventaja al usar cascada es que, ante cualquier fallo en algún paso, necesariamente se deberá solucionar el mismo en la fase actual de desarrollo sin tener que regresar a la etapa anterior, este mecanismo de solución a largo tiempo representa demoras en la ejecución del paso en el que se busca solucionar problemas.

### 2.2.18.2 Modelo Prototipo

El modelo de prototipos tiene un enfoque iterativo y está basado la elaboración de prototipos finales de la aplicación que se van desarrollando encima de una versión anterior hasta poder llegar a la aplicación final, en este modelo los refinamientos son

esenciales ya que gracias a ellos se podrá ir agregando nuevas especificaciones del sistema a las versiones tempranas de lo que llegara a ser el software [36].

Debido a que el prototipo es una versión temprana del software, su uso se ha visto direccionado a escenarios en donde el cliente no tiene bien definidos los requerimientos del software.

### 2.2.18.3 Modelo Incremental

El modelo incremental aplica secuencias lineales en forma escalonada a medida que avanza el calendario de actividades. Cada secuencia lineal produce "incrementos" de software susceptibles de entregarse de manera parecida a los incrementos producidos en un flujo de proceso evolutivo [37].

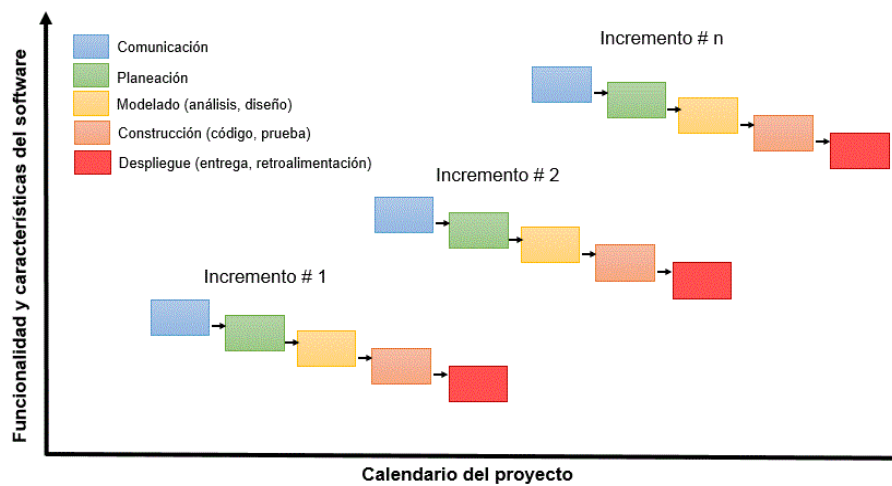


Figura 5: Pasos del Modelo Incremental

Elaborado por: [37]

De acuerdo con la figura 7 se puede notar que el modelo incremental tiene un enfoque combinado que al término de cada etapa entrega un producto que ya puede operar, como una versión del sistema [37].

### 2.2.18.4 Modelo Espiral

El modelo espiral es un proceso de software evolutivo que conjuga la naturaleza iterativa de construcción de prototipos con los aspectos controlados y sistemáticos del modelo lineal secuencial [38].

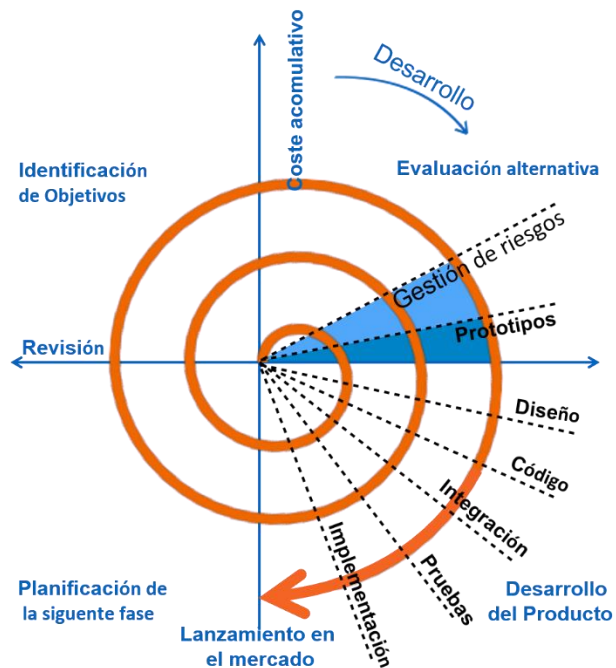


Figura 6: Pasos del modelo espiral

Elaborado por: [39]

En el modelo espiral cada rotación representa una cascada en pequeña escala y la cantidad de vueltas la complejidad del proyecto es por ello que entre más avanzado sea un sistema, obviamente tendrá mayor volumen. El uso de este tipo de modelo se hizo muy común para proyectos críticos en donde el mínimo fallo podría causar daños irreversibles.

### 2.2.19 Metodologías ágiles

Las metodologías ágiles nacieron alrededor del año 2001 en Estados Unidos, este nuevo tipo de técnicas para desarrollar software se basan principalmente en permitir a los equipos de desarrollo actuar de forma rápida y respondiendo a los cambios que puedan surgir a lo largo del proyecto.

Ligado a la nueva forma para hacer programas informáticos se creó el Manifiesto Ágil que es el punto de partida para resumir la filosofía ágil y contiene los conceptos relacionados con el desarrollo ágil de software [33].

#### 2.2.19.1 Metodología Scrum

La metodología Scrum define un marco para la gestión de proyectos con un rápido cambio de requisitos. En términos generales Scrum se caracteriza en que el desarrollo de software se realiza mediante iteraciones, denominadas sprints, con una duración de 30 días. El resultado de cada sprint es un incremento ejecutable que se muestra al cliente.

Otra característica importante son las reuniones a lo largo proyecto, estas son las verdaderas protagonistas, especialmente la reunión diaria de 15 minutos del equipo de desarrollo para coordinación e integración, Scrum ha sido utilizado con éxito durante varios años [33].

### 2.2.19.2 Metodología XP

XP es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo.

XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico [33].

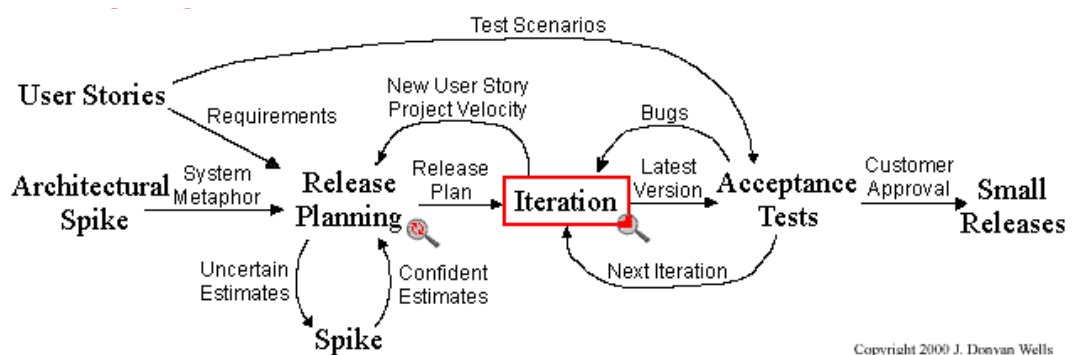


Figura 7: Proceso de la Metodología XP

Elaborado por: Don Wells [40]

### **2.3 Propuesta de solución**

El presente trabajo presenta como propuesta de solución, el desarrollo de un sistema de monitoreo que obtenga datos específicos, que han sido generados a partir de las aplicaciones informáticas que se ejecuten en un equipo de los laboratorios de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial. El software por desarrollarse constituye la tercera fase del proyecto Plataforma CloudIoT de Control y Monitoreo del Uso de Equipamiento y Programas Informáticos en Aulas y Laboratorios, por lo tanto, una parte del mismo deberá integrarse al servidor ya utilizado por el proyecto.

## **CAPITULO 3 METODOLOGÍA**

### **3.1 Modalidad de la Investigación**

La investigación será de campo porque obligatoriamente se requerirá trabajar con frecuencia en los laboratorios de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial para recolectar la información que será necesaria en el desarrollo de las aplicaciones de monitoreo y también en algún grado determinar la validez de las mismas, también es importante señalar que la presente tendrá un enfoque bibliográfico porque se emplearán conceptos y técnicas de libros, documentos científicos, etc.

El trabajo a su vez contará con investigación aplicada ya que se utilizarán varios conceptos, técnicas, herramientas aprendidas y utilizadas a lo largo de la carrera.

### **3.2 Población y Muestra**

De acuerdo con la propuesta establecida no se requiere población y muestra.

### **3.3 Recolección de Información**

La información de utilidad para el presente trabajo de investigación será recolectada por medio de la observación del entorno en los laboratorios de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, en un tiempo determinado y con los permisos necesarios de los mismos para poder monitorear actividades informáticas, con lo que se constituye totalmente un trabajo de campo.



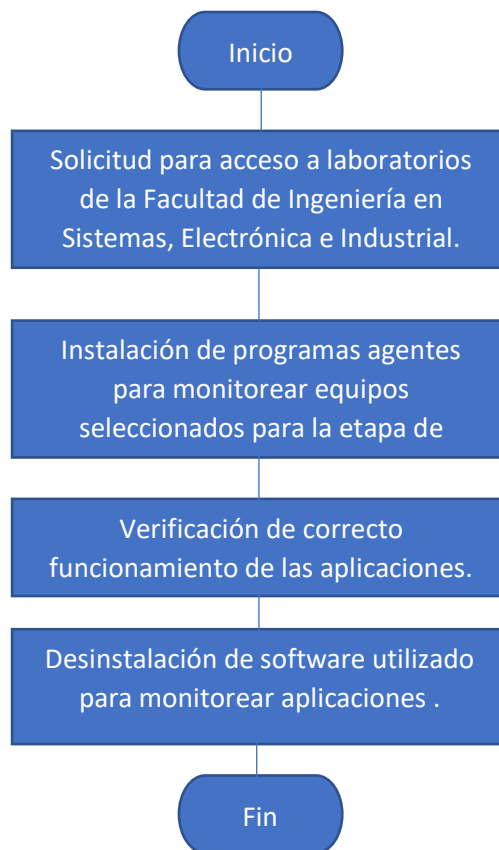


Figura 8: Flujograma del proceso general para la conseguir los datos

Elaborado por: Paul Tandazo

Durante la investigación las actividades informáticas serán generadas por los estudiantes que utilizan los laboratorios de la FISEI, ya que son estas personas quienes al utilizar las aplicaciones desktop y navegadores generan una gran cantidad de datos útiles para el presente trabajo. El proceso de recolección netamente se realizará mediante aplicativos que serán instalados en cada una de las computadoras de los laboratorios y tendrán como propósito capturar y guardar en un servidor de base de datos, la información de las aplicaciones utilizadas durante ciertos lapsos de tiempo.

### 3.4 Procesamiento y Análisis de Datos

Los datos serán recolectados, seleccionados y clasificados de acuerdo a su origen (desktop o desde navegador) en los laboratorios de la FISEI, para luego mediante una herramienta informática proceder a organizarlos mediante gráficos estadísticos en

cuanto a criterios de los inconvenientes presentados por el mal uso de la red o de las aplicaciones con fines no educativos instaladas en los laboratorios de la facultad.

### **3.5 Desarrollo del Proyecto**

Para el presente proyecto se considera las siguientes fases:

- Almacenamiento de datos
  - Modelado del almacenamiento de datos.
  - Integración del modelo de datos a la Plataforma CloudIoT de control y monitoreo de la Universidad Técnica de Ambato.
- Monitoreo de aplicaciones de escritorio.
  - Estudio y análisis de herramientas comerciales para monitorear actividad de aplicaciones Desktop.
  - Construcción de una herramienta Open Source para monitorear la actividad de aplicaciones únicamente desktop.
  - Almacenamiento masivo de las actividades informáticas de las aplicaciones desktop en un sistema de Cloud.
- Monitoreo de actividades generadas por navegadores Web.
  - Desarrollar un módulo que permita capturar directamente las páginas web accedidas desde cada aplicación utilizada para navegar en las computadoras de los laboratorios de la facultad.
  - Depuración del código desarrollado.
  - Implementación y pruebas del software desarrollado en base a disponibilidad y autorizaciones.
  - Almacenar en el Cloud los datos de navegación generados por los laboratorios.
  - Revisión de conexión, escritura de datos en el Cloud.
  - Corrección de errores de programación.
- Análisis y representación de los datos monitoreados y almacenados en el Cloud.
  - Desarrollo de un sistema web para la generación de reportes sobre el uso que tienen los programas informáticos en un horario determinado.
- Integración de las páginas web con la plataforma CloudIoT.

## CAPÍTULO 4 DESARROLLO DE LA PROPUESTA

### 4.1 Análisis de factibilidad

#### 4.1.1 Factibilidad operativa

El sistema de monitoreo permitirá la obtención y representación de las actividades informáticas que se puedan generar en una computadora de la Facultad de Ingeniería de Sistemas, Electrónica e Industrial. La recolección de datos está basada en aplicativos agente que únicamente capturaran datos necesarios y establecidos por los requerimientos del software.

#### 4.1.2 Factibilidad económica

Debido a que el presente proyecto se realizará mediante la modalidad de Proyecto de Investigación y estará destinado a integrar un Proyecto de Investigación de la Universidad será financiado en su totalidad por el investigador, el uso de otros servicios como el motor de base de datos están contemplados en el proyecto macro como herramientas libres, sin costo.

N°	Detalle	Unidad	Cantidad	Valor Unitario	Valor Total
1	Internet 2 Mbps	h	360	0,8	288
2	Resma de papel bond A4	c/u	1	5	5
3	Transporte	c/u	250	0,3	75
4	Carpetas	c/u	2	1	2
5	Impresiones	c/u	150	0,05	7,5
6	Copias	c/u	400	0,03	12
7	Ordenador i3 HP	c/u	1	1200	1200
8	Esferos	c/u	3	0,5	1,5
9	Lapiz	c/u	3	0,6	1,8
10	Disco duro 2 TB	c/u	1	150	150
Subtotal					1742,8
Imprevistos (10%)					174,28
<b>TOTAL</b>					<b>1917,08</b>

Figura 9: Cuadro de presupuesto

Elaborado por Paul Tandazo

### 4.1.3 Factibilidad técnica

#### 4.1.3.1 Factibilidad en software

El sistema de monitoreo deberá instalarse en los equipos de la facultad y obtener datos de estos, por lo cual se procedió a seleccionar un IDE de programación que ofrezca mejores características a la hora de desarrollar el sistema [41], [42], [43]:

Tabla 2: Principales características de Visual Studio y NetBeans

Elaborado por: Paul Tandazo

Visual Studio	NetBeans
<ul style="list-style-type: none"><li>• Herramientas de desarrollo.</li><li>• Servidores Azure para desarrolladores.</li><li>• Servicios de Visual Studio Team.</li><li>• Soporte para Windows y Windows Server.</li><li>• SQL Server.</li><li>• Soporte para varios motores de Base de Datos.</li><li>• Visual Studio App Center.</li><li>• Administrador de paquetes nuget.</li><li>• Versiones de pago y libres.</li><li>• Plataforma de desarrollo con múltiples funciones para varios SO, Web y Cloud.</li><li>• IntelliSense para sugerencias y descripciones de API, autocompleta palabras para mayor velocidad.</li><li>• Rapida localización y diagnosticos de errores.</li><li>• Soporte de depuración para todos los idiomas incluidos.</li><li>• Implementación de aplicaciones en el escritorio o emuladores en dispositivos móviles y otros métodos de depuración.</li><li>• Power BI Pro.</li></ul>	<ul style="list-style-type: none"><li>• Editor de código.</li><li>• Gestión de Proyectos.</li><li>• Herramientas de análisis.</li><li>• Depurador visual.</li><li>• Soporte de múltiples idiomas.</li><li>• Soporte multiplataforma.</li><li>• Convertidores de código.</li><li>• Analizadores de código.</li><li>• IDE de código abierto, basado en Java.</li><li>• Versión libre.</li><li>• Arquitectura modular.</li><li>• Generadores de código.</li><li>• Plantillas de código.</li><li>• Soporte para múltiples lenguajes.</li><li>• Plugging para extensión de idiomas.</li><li>• Constructor de GUI intuitivo.</li><li>• FindBugs para identificar y solucionar problemas comunes.</li><li>• Asistencia para crear aplicaciones JavaFX, Java</li></ul>

<ul style="list-style-type: none"> <li>• Dependencia del Net Framework.</li> <li>• Producido por Microsoft Corporation.</li> <li>• 96% en experiencia del cliente.</li> </ul>	<p>SE, and Java EE.</p> <ul style="list-style-type: none"> <li>• Dependencia de la máquina virtual de Java.</li> <li>• Producido por NetBeans.</li> <li>• 94% en experiencia del cliente.</li> </ul>
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Tomando en referencia las características analizadas y que el sistema de monitoreo debe adaptarse al resto de la Plataforma CloudIoT como fase 3 de la misma. Durante el desarrollo de software necesario para el proyecto se utilizará el IDE Visual Studio que ofrece ventajas importantes para este caso en específico como soporte nativo para sistemas operativos Windows y adaptadores para conexiones estables con MySQL que actualmente está siendo usado como base de datos principal del proyecto.

#### 4.1.3.2 Factibilidad en hardware

La Facultad de Ingeniería en Sistemas, Electrónica e Industrial cuenta con los equipos y plataformas necesarias para la ejecución adecuada del sistema de monitoreo a desarrollarse.

#### 4.2 Metodología por aplicar

Como se investigó en el capítulo 2, existen dos tipos de metodologías muy utilizadas para el desarrollo de software, primero partiremos por seleccionar el tipo de metodología que más se ajuste al proyecto en ejecución:

Tabla 3: Comparación de Metodologías Tradicionales y Agiles

Elaborado por: Patricio Letelier y Carmen Penades [33].

<b>Metodologías Tradicionales</b>	<b>Metodologías Agiles</b>
Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo	Basadas en heurísticas provenientes de prácticas de producción de código
Cierta resistencia a los cambios	Especialmente preparados para

	cambios durante el proyecto
Impuestas externamente	Impuestas internamente (por el equipo de desarrollo)
Proceso mucho más controlado, con numerosas políticas/normas	Proceso menos controlado, con pocos principios
Existe un contrato prefijado	No existe contrato tradicional o al menos es bastante flexible
El cliente interactúa con el equipo de desarrollo mediante reuniones	El cliente es parte del equipo de desarrollo
Grupos grandes y posiblemente distribuidos	Grupos pequeños (<10 integrantes) y trabajando en el mismo sitio
Más artefactos	Pocos artefactos
Más roles	Pocos roles
La arquitectura del software es esencial y se expresa mediante modelos	Menos énfasis en la arquitectura del software

Por las características del proyecto resulta más oportuno utilizar una metodología ágil ya que conceptualmente están mejor adaptadas para afrontar posibles cambios que se presenten a lo largo del proyecto, además no requieren grandes grupos de desarrolladores y aportan una cierta garantía de calidad al producto software.

Una vez establecido el tipo de metodología, a continuación, se presentará un análisis de comparación que establece diferentes aproximaciones entre metodologías ágiles en base a ciertos parámetros como la vista del sistema como algo cambiante, colaboración entre miembros del equipo, y características un tanto más específicas de la propia metodología que van desde la simplicidad, excelencia académica, resultados, adaptabilidad, como se describe en la siguiente tabla 4:

Tabla 4: Ranking de Agilidad

Elaborado por Patricio Letelier y Carmen Penades [33].

Características	Crystal	DSDM	FDD	LD	Scrum	XP
Sistema como algo cambiante	4	3	3	4	5	5
Colaboración	5	4	4	4	5	5
Resultados	5	4	4	4	5	5
Simplicidad	4	3	5	3	5	5
Adaptabilidad	5	3	3	4	4	3
Excelencia técnica	3	4	4	4	3	4
Prácticas de colaboración	5	4	3	3	4	5

En base a resultados de las pruebas obtenidos en la Tabla 4, se puede notar que la metodología que más sobresale como ágil es XP porque supera a Scrum en excelencia técnica y prácticas de colaboración. Netamente para el desarrollo del sistema de monitoreo se utilizará como metodología Extreme Programming porque a más de ser una técnica probada por varios años.

#### 4.2.1 Metodología Extreme Programming

La programación extrema o simplemente XP es una metodología alineada en potenciar las relaciones interpersonales como un punto clave en el éxito del desarrollo de un producto software [33], permitiendo a los desarrolladores responder con confianza ante los cambios en los requisitos del cliente, incluso al final del ciclo de vida. XP hace énfasis en el trabajo de equipo integrando como socios iguales tanto a clientes como al equipo de desarrollo [44].

##### 4.2.1.1 Valores de Extreme Programming

La Programación Extrema busca mejorar un proyecto de software apoyándose en 5 puntos esenciales [44] o también llamados valores de la metodología.

### **Comunicación**

XP requiere que los programadores se comuniquen constantemente con los clientes y con el resto del equipo de desarrollo, una comunicación constante evita y corrige los problemas que se puedan generar con los requerimientos de actualizaciones constantes y diseños técnicos [45].

### **Simplicidad**

XP debe conseguir mantener un diseño simple y limpio, por lo cual se recomienda empezar con lo más sencillo de todo el sistema entendiendo que los cambios pueden presentarse en todo momento [45].

### **Retroalimentación**

La retroalimentación o Feedback se produce desde la primera vez que se prueba el software, este valor está ligado permanente a los cambios, una vez que se entrega el sistema a los clientes se debe implementar los cambios que se puedan sugerir. En muchos casos una retroalimentación más importante es la que puede surgir de colegas programadores, miembros del equipo en las etapas de desarrollo.

### **Respeto**

En XP cada pequeño éxito profundiza el respeto por las contribuciones únicas que realizan cada uno de los miembros del equipo de desarrollo [44].

### **Coraje**

Los programadores en XP pueden responder con coraje a los cambios en los requisitos y la tecnología [44]. Este valor tiene estrecha relación con el nivel de confianza que debe existir en el equipo de desarrollo para responder de manera satisfactoria a retroalimentaciones puntuales.

#### **4.2.1.2 Fases de Extreme Programming**

Al igual que en otras metodologías, XP tiene un ciclo de vida para cumplir con el desarrollo total de un proyecto de software. Las fases que edifican la programación



extrema son esenciales para desarrollar software a medida y que sobre todo cumpla con metas, objetivos y plazos establecidos en el inicio del proyecto.

### **Planificación del proyecto**

Al igual que en otras metodologías ya sean tradicionales o ágiles, en XP el primer paso es la recolección de requerimientos. En esta etapa se emplean pequeñas tarjetas, también conocidas como Historias de Usuario en las que el cliente define sus necesidades y que acciones realizara el sistema a desarrollarse [46].

Una vez que las historias de usuario han sido definidas se procede a realizar un plan de entregas en donde se especifica qué historias de usuario se implementarán para cada lanzamiento del sistema y las fechas de esos lanzamientos [46].

### **Diseño**

Debido a que XP persigue la simplicidad como un valor fundamental de la metodología, los diseños están encaminados a ser simples y sencillos tomando en cuenta que lo simple siempre lleva menos tiempo para ser realizado que lo complejo [47].

En esta etapa entran en acción las tarjetas CRC (Clase, Responsabilidad, Colaboración) que permiten contribuir en el diseño al equipo completo de desarrollo, CRC tiene como finalidad terminar con el modo de pensamiento procedimental y apreciar más plenamente la tecnología de objetos [48].

### **Codificación**

A pesar de que XP implementa sus propias etapas, a la hora de escribir el código se debe respetar el formato de los estándares de codificación ya establecidos. Los estándares de codificación mantienen el código consistente, fácil de leer y refactorizar por todo el equipo. Es importante resaltar que en la metodología XP uno de los pocos requisitos es tener al cliente disponible con la intención de aprovechar una retroalimentación oportuna y eficiente [40].

### **Pruebas**

Durante la etapa de pruebas, todos los resultados conseguidos con el código deben ser sometidos a pruebas unitarias antes de ser liberados, cuando se detecta un error el

equipo de desarrollo debe ser capaz de encontrar una solución que no altere los resultados buscados con el software.

Las pruebas de aceptación son creadas a partir de las historias de usuario y por ende deben garantizar que las funcionalidades definidas respondan acorde las necesidades del cliente [49].

### 4.3 Ejecución del proyecto

#### 4.3.1 Planificación del proyecto

Debido a que en el desarrollo del presente proyecto se utilizara la metodología XP, el primer paso que se llevó a cabo fue la reunión con el equipo encargado del proyecto Plataforma Cloud, es así, como se pudo llegar a definir los roles que intervendrán directamente en el proyecto y las historias de usuario mediante las cuales se puede determinar las necesidades que deberá cumplir el software a desarrollar.

##### 4.3.1.1 Roles del proyecto

Para desarrollar el proyecto “Sistema Open Source con conexión a la Plataforma CloudIoT para el monitoreo de programas informáticos en los laboratorios de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial” se dispondrá de los siguientes roles:

Tabla 5: Roles definidos dentro del proyecto

Elaborado por: Paul Tandazo

Rol	Responsable	Función
<b>Usuario</b>	FISEI	Establecer las funciones que debe cumplir el software de monitoreo.
<b>Desarrollador</b>	Paul Andres Tandazo	Programar el código fuente de los módulos necesarios.
<b>Tester</b>	Ing. Rubén Nogales	Ejecutar pruebas a fin de conocer el correcto funcionamiento de la aplicación de monitoreo.

#### 4.3.1.2 Historias de usuario

Las historias de usuario son la base sobre la cual se fundamenta un sistema que se desarrolle con XP o cualquier otra metodología ágil. Para el proyecto en desarrollo se definieron historias de usuario en donde se detalla los requerimientos que serán necesarios para que el software de monitoreo cumpla con los objetivos planificados inicialmente:

Tabla 6: Historias de Usuario definidas por el cliente

Elaborado por: Paul Tandazo

<b>Numero</b>	<b>Identificador</b>	<b>Nombre</b>	<b>Prioridad</b>
<b>1</b>	H1	Capturar actividades informáticas de aplicaciones de escritorio en tiempo real	<b>Alta</b>
<b>2</b>	H2	Almacenamiento de datos de escritorio en el Cloud	<b>Media</b>
<b>3</b>	H3	Implementación de respaldos para datos fuera de línea.	<b>Alta</b>
<b>4</b>	H4	Transferencia de los datos respaldados al servidor	<b>Alta</b>
<b>5</b>	H5	Reconocimiento e identificación de páginas web visitadas	<b>Alta</b>
<b>6</b>	H6	Almacenamiento de datos web en el Cloud	<b>Media</b>
<b>7</b>	H7	Generación de Reportes	<b>Alta</b>
<b>8</b>	H8	Integración con la Plataforma CloudIoT	<b>Alta</b>

Para el cumplimiento de cada una de las Historias de Usuario, se definieron las siguientes actividades:

Tabla 7: Actividades por Historia de Usuario

Elaborado por: Paul Tandazo

H1	<ul style="list-style-type: none"> <li>• Desarrollar la Capa de acceso a datos</li> <li>• Identificar procesos de inicio y cierre de las aplicaciones</li> <li>• Identificar el usuario que ejecuta la actividad</li> <li>• Filtrar aplicaciones en segundo plano</li> <li>• Ejecutar aplicación en segundo plano.</li> </ul>
H2	<ul style="list-style-type: none"> <li>• Conectar con el servidor</li> <li>• Guardar actividades en la base de datos</li> <li>• Eliminar conexiones abiertas</li> </ul>
H3	<ul style="list-style-type: none"> <li>• Seleccionar alternativa para respaldos</li> <li>• Generar archivos para respaldar datos</li> <li>• Guardar datos fuera de línea</li> </ul>
H4	<ul style="list-style-type: none"> <li>• Conectar con el servidor</li> <li>• Leer el respaldo generado por datos fuera de línea</li> <li>• Guardar datos respaldados en el servidor</li> <li>• Eliminar archivo de respaldo</li> </ul>
H5	<ul style="list-style-type: none"> <li>• Obtener sitios visitados</li> <li>• Generar clase de sitios visitados.</li> </ul>
H6	<ul style="list-style-type: none"> <li>• Establecer conexión al Cloud</li> <li>• Guardar datos en el servidor</li> </ul>
H7	<ul style="list-style-type: none"> <li>• Diseñar estructura del sitio web de monitoreo</li> <li>• Diseño de interfaces</li> <li>• Obtener datos desde el servidor</li> <li>• Representar datos en reportes</li> </ul>
H8	<ul style="list-style-type: none"> <li>• Subir paginas al sitio de la Plataforma CloudIoT</li> <li>• Agregar funcionalidades de Monitoreo al Menú de la Plataforma.</li> </ul>

Historia 1: Capturar actividades informáticas en tiempo real

Tabla 8: Historia de Usuario – Capturar actividades informáticas en tiempo real

Elaborado por: Paul Tandazo

<b>Historia de Usuario</b>	
<b>Numero: 1</b>	<b>Usuario: Todos</b>
<b>Nombre de historia:</b> Capturar actividades informáticas de aplicaciones de escritorio en tiempo real	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 6	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Paul Andres Tandazo	
<p><b>Descripción:</b> De cada aplicación ejecutada: el software capturara el nombre del ejecutable, fecha y hora de ejecución, nombre de la ventana (si se despliega ventana alguna), nombre del equipo, usuario y laboratorio.</p> <p>La recolección de información será en tiempo real en los equipos de escritorio que pertenezcan a los laboratorios de la facultad y que previamente tengan instalado el software.</p>	
<p><b>Observaciones:</b> Por necesidades de proyecto, se requiere que la captura de actividades informáticas este totalmente garantizada, motivo por el cual la prioridad de la historia es alta.</p>	

Actividades de la Historia

Tabla 9: Actividad 1 – Historia de Usuario 1 – Capturar de actividades informáticas

Elaborado por: Paul Tandazo

<b>Tarea</b>	
<b>Numero: 1</b>	<b>Numero de la Historia: 1</b>

<b>Nombre de la tarea:</b> Desarrollar la Capa de Acceso a Datos	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Fecha inicio:</b> 2018	<b>Fecha fin:</b> 2018
<b>Programador responsable:</b> Paul Tandazo	
<b>Descripción:</b> Diseñar y desarrollar clases que implementen métodos, propiedades, variables y cualquier otro objeto necesario para acceder a los datos de la plataforma CloudIoT.	

Tabla 10: Actividad 2 – Historia de Usuario 1 – Capturar de actividades informáticas

Elaborado por: Paul Tandazo

<b>Tarea</b>	
<b>Numero:</b> 2	<b>Numero de la Historia:</b> 1
<b>Nombre de la tarea:</b> Identificar procesos de inicio y cierre de las aplicaciones	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Fecha inicio:</b> 2018	<b>Fecha fin:</b> 2018
<b>Programador responsable:</b> Paul Tandazo	
<b>Descripción:</b> La aplicación debe poseer la capacidad de guardar temporalmente el id, fecha y hora de ejecución del proceso bajo el cual se lanza una aplicación, además deberá identificar la fecha y hora de cierre del proceso iniciado.	

Tabla 11: Actividad 3 – Historia de Usuario 1 – Capturar de actividades informáticas

Elaborado por: Paul Tandazo

<b>Tarea</b>	
<b>Numero:</b> 3	<b>Numero de la Historia:</b> 1

<b>Nombre de la tarea:</b> Identificar el usuario que ejecuta la actividad	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.5
<b>Fecha inicio:</b> 2018	<b>Fecha fin:</b> 2018
<b>Programador responsable:</b> Paul Tandazo	
<b>Descripción:</b> En todos los datos capturados, es necesario identificar el usuario asociado a la aplicación que se esté ejecutando en ese momento.	

Tabla 12: Actividad 4 – Historia de Usuario 1 – Capturar de actividades informáticas

Elaborado por: Paul Tandazo

<b>Tarea</b>	
<b>Numero: 4</b>	<b>Numero de la Historia: 1</b>
<b>Nombre de la tarea:</b> Filtrar aplicaciones en segundo plano	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 2018	<b>Fecha fin:</b> 2018
<b>Programador responsable:</b> Paul Tandazo	
<b>Descripción:</b> En el sentido de evitar datos innecesarios o redundancia de datos, el software a desarrollarse deberá eliminar aplicaciones en segundo plano, principalmente procesos Windows que se ejecutan automáticamente.	

Tabla 13: Actividad 5 – Historia de Usuario 1 – Capturar de actividades informáticas

Elaborado por: Paul Tandazo

<b>Tarea</b>	
<b>Numero: 5</b>	<b>Numero de la Historia: 1</b>
<b>Nombre de la tarea:</b> Ejecutar en segundo plano	

<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.5
<b>Fecha inicio:</b> 2018	<b>Fecha fin:</b> 2018
<b>Programador responsable:</b> Paul Tandazo	
<p><b>Descripción:</b> Al ser un software agente, se deberá garantizar que la aplicación se pueda ejecutar en segundo plano y no despliegue ventana alguna.</p> <p>El principal propósito de esta actividad es reducir al máximo, el impacto que el software de monitoreo pueda tener en el equipo, se debe garantizar un consumo reducido de recursos como CPU, memoria, disco y red.</p>	

Historia 2: Almacenamiento de datos de escritorio en el Cloud

Tabla 14: Historia de Usuario – Almacenamiento de datos en el Cloud

Elaborado por: Paul Tandazo

<b>Historia de Usuario</b>	
<b>Numero:</b> 2	<b>Usuario:</b> Todos
<b>Nombre de historia:</b> Almacenamiento de datos de escritorio en el Cloud	
<b>Prioridad en negocio:</b> Media	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 2	<b>Iteración asignada:</b>
<b>Programador responsable:</b> Paul Andres Tandazo	
<p><b>Descripción:</b> Con el fin de garantizar que los datos estén asegurados, la aplicación de monitoreo almacenara los datos previamente capturados en el servidor donde reside el resto de la Proyecto Cloud IOT.</p>	
<p><b>Observaciones:</b> En vista que el almacenamiento de datos depende de una conexión al servidor y esta conexión puede estar o no disponible, la prioridad de la historia se fijara en media.</p>	



Actividades de la Historia

Tabla 15: Actividad 1 – Historia de Usuario 2 – Almacenamiento de datos en Cloud

Elaborado por: Paul Tandazo

<b>Tarea</b>	
<b>Numero: 1</b>	<b>Numero de la Historia: 2</b>
<b>Nombre de la tarea:</b> Conectar con el servidor	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.5
<b>Fecha inicio:</b> 2018	<b>Fecha fin:</b> 2018
<b>Programador responsable:</b> Paul Tandazo	
<b>Descripción:</b> Al tratarse de un sistema que tiene por objetivo capturar datos desde un equipo de escritorio, la aplicación, dentro de las posibilidades del entorno deberá garantizar una conexión estable con el servidor.	

Tabla 16: Actividad 2 – Historia de Usuario 2 – Almacenamiento de datos en Cloud

Elaborado por: Paul Tandazo

<b>Tarea</b>	
<b>Numero: 2</b>	<b>Numero de la Historia: 2</b>
<b>Nombre de la tarea:</b> Guardar actividades en la base de datos	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 2018	<b>Fecha fin:</b> 2018
<b>Programador responsable:</b> Paul Tandazo	
<b>Descripción:</b> Implementar clases, métodos, propiedades, variables necesarias para insertar las actividades informáticas en el servidor.	

Tabla 17: Actividad 3 – Historia de Usuario 2 – Almacenamiento de datos en Cloud

Elaborado por: Paul Tandazo

<b>Tarea</b>	
<b>Numero: 3</b>	<b>Numero de la Historia: 2</b>
<b>Nombre de la tarea:</b> Eliminar conexiones abiertas	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.5
<b>Fecha inicio:</b> 2018	<b>Fecha fin:</b> 2018
<b>Programador responsable:</b> Paul Tandazo	
<b>Descripción:</b> En el sentido de reducir el impacto de la aplicación sobre el equipo se deberá eliminar los objetos que establecen conexión con el servidor y que ocupen espacio en memoria.	

Historia 3: Implementación de respaldos para datos fuera de línea

Tabla 18: Historia de Usuario – Implementación de respaldos

Elaborado por: Paul Tandazo

<b>Historia de Usuario</b>	
<b>Numero: 3</b>	<b>Usuario:</b> Todos
<b>Nombre de historia:</b> Implementación de respaldos para datos fuera de línea.	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b>
<b>Programador responsable:</b> Paul Andres Tandazo	
<b>Descripción:</b> Para prevenir perdida de datos por falta de conexión al servidor, la información que se vaya generando fuera de línea se deberá respaldar en algún tipo de archivo.	

**Observaciones:** La prioridad de la historia se ha fijado en alta, debido a que los datos tienen vital importancia y en si la aplicación se enfoca netamente a capturar y guardar datos, la información se debe garantizar bajo todo sentido.

Se debe tomar en cuenta que los respaldos se utilizan para guardar datos cuando el equipo está fuera de línea.

#### Actividades de la Historia

Tabla 19: Actividad 1 – Historia de Usuario 3 – Implementación de respaldos

Elaborado por: Paul Tandazo

<b>Tarea</b>	
<b>Numero: 1</b>	<b>Numero de la Historia: 3</b>
<b>Nombre de la tarea:</b> Seleccionar alternativa para respaldos	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados: 2</b>
<b>Fecha inicio:</b> 2018	<b>Fecha fin:</b> 2018
<b>Programador responsable:</b> Paul Tandazo	
<b>Descripción:</b> Determinar mediante pruebas una manera eficiente, liviana y limpia para respaldar los datos de monitoreo que se producen cuando la aplicación no ha encontrado una conexión al servidor.	

Tabla 20: Actividad 2 – Historia de Usuario 3 – Implementación de respaldos

Elaborado por: Paul Tandazo

<b>Tarea</b>	
<b>Numero: 2</b>	<b>Numero de la Historia: 3</b>
<b>Nombre de la tarea:</b> Generar archivos para respaldar datos	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados: 0.5</b>

<b>Fecha inicio:</b> 2018	<b>Fecha fin:</b> 2018
<b>Programador responsable:</b> Paul Tandazo	
<b>Descripción:</b> Para prevenir pérdida de datos por falta de conexión al servidor, una vez que se detecte que no exista conexión, se debe crear un archivo en el equipo local para respaldar la información que se vaya generando fuera de línea.	

Tabla 21: Actividad 3 – Historia de Usuario 3 – Implementación de respaldos

Elaborado por: Paul Tandazo

<b>Tarea</b>	
<b>Numero:</b> 3	<b>Numero de la Historia:</b> 3
<b>Nombre de la tarea:</b> Guardar datos fuera de línea	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.5
<b>Fecha inicio:</b> 2018	<b>Fecha fin:</b> 2018
<b>Programador responsable:</b> Paul Tandazo	
<b>Descripción:</b> Crear métodos para gestionar la escritura de datos fuera de línea de monitoreo en el archivo de respaldo creado con anterioridad.	

Historia 4: Transferencia de los datos respaldados al servidor

Tabla 22: Historia de Usuario – Transferencia de los datos respaldados al servidor

Elaborado por: Paul Tandazo

<b>Historia de Usuario</b>	
<b>Numero:</b> 4	<b>Usuario:</b> Todos
<b>Nombre de historia:</b> Transferencia de los datos respaldados al servidor.	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alto

<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b>
<b>Programador responsable:</b> Paul Andres Tandazo	
<p><b>Descripción:</b> El sistema de monitoreo permitirá recuperar los datos guardados fuera de línea para que puedan almacenados en el servidor del proyecto.</p> <p>Una vez reestablecida la conexión, la aplicación deberá escribir todos los datos generados fuera de línea en el servidor.</p>	
<p><b>Observaciones:</b> Por la necesidad de garantizar todos los datos generados por el monitoreo en el Cloud, la inserción de los respaldos generados fuera de línea tendrá una prioridad alta, al igual que la recuperación de información en tiempo real.</p>	

#### Actividades de la Historia

Tabla 23: Actividad 1 – Historia de Usuario 4 – Transferencia de los datos respaldados al servidor

Elaborado por: Paul Tandazo

<b>Tarea</b>	
<b>Numero:</b> 1	<b>Numero de la Historia:</b> 4
<b>Nombre de la tarea:</b> Conectar con el servidor	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.5
<b>Fecha inicio:</b> 2018	<b>Fecha fin:</b> 2018
<b>Programador responsable:</b> Paul Tandazo	
<p><b>Descripción:</b> Implementar métodos para establecer una conexión con el servidor y acceso a las entidades de la base de datos.</p>	

Tabla 24: Actividad 2 – Historia de Usuario 4 – Transferencia de los datos respaldados al servidor

Elaborado por: Paul Tandazo

<b>Tarea</b>	
<b>Numero: 2</b>	<b>Numero de la Historia: 4</b>
<b>Nombre de la tarea:</b> Lectura del respaldo	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados: 1</b>
<b>Fecha inicio:</b> 2018	<b>Fecha fin:</b> 2018
<b>Programador responsable:</b> Paul Tandazo	
<b>Descripción:</b> En caso de existir respaldos en el equipo local, el sistema de monitoreo permitirá leer los archivos con respaldos fuera de línea y mantener en memoria los datos recuperados.	

Tabla 25: Actividad 3 – Historia de Usuario 4 – Transferencia de los datos respaldados al servidor

Elaborado por: Paul Tandazo

<b>Tarea</b>	
<b>Numero: 3</b>	<b>Numero de la Historia: 4</b>
<b>Nombre de la tarea:</b> Guardar datos respaldados en el servidor	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados: 1</b>
<b>Fecha inicio:</b> 2018	<b>Fecha fin:</b> 2018
<b>Programador responsable:</b> Paul Tandazo	
<b>Descripción:</b> Crear métodos, variables y demás objetos que permitan el almacenamiento de datos en base a entidades establecidas y creadas en el servidor.	

Tabla 26: Actividad 4 – Historia de Usuario 4 – Transferencia de los datos respaldados al servidor

Elaborado por: Paul Tandazo

<b>Tarea</b>	
<b>Numero: 4</b>	<b>Numero de la Historia: 4</b>
<b>Nombre de la tarea:</b> Eliminar archivo de respaldo	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.5
<b>Fecha inicio:</b> 2018	<b>Fecha fin:</b> 2018
<b>Programador responsable:</b> Paul Tandazo	
<b>Descripción:</b> Los archivos de respaldo generados fuera de línea se eliminarán una vez que los datos hayan sido leídos e ingresados en el servidor.	

Historia 5: Reconocimiento de páginas web visitadas

Tabla 27: Historia de Usuario – Reconocimiento e identificación de páginas web visitadas

Elaborado por: Paul Tandazo

<b>Historia de Usuario</b>	
<b>Numero: 5</b>	<b>Usuario:</b> Todos
<b>Nombre de historia:</b> Reconocimiento e identificación de páginas web visitadas.	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 4	<b>Iteración asignada:</b>
<b>Programador responsable:</b> Paul Andres Tandazo	
<b>Descripción:</b> El sistema de monitoreo identificara y guardara temporalmente las direcciones, fecha, hora y títulos de páginas web visitadas desde los diferentes navegadores instalados en los equipos.	
<b>Observaciones:</b> Debido a que los datos generados por el uso de navegadores web constituyen un gran volumen de información, importante y útil para el propósito de	

la Plataforma, se considera de alta prioridad reconocer e identificar las páginas visitadas.

Actividades de la Historia

Tabla 28: Actividad 1 – Historia de Usuario 5 – Reconocimiento e identificación de páginas web visitadas

Elaborado por: Paul Tandazo

<b>Tarea</b>	
<b>Numero: 1</b>	<b>Numero de la Historia: 5</b>
<b>Nombre de la tarea:</b> Obtener sitios visitados	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados: 2</b>
<b>Fecha inicio:</b> 2018	<b>Fecha fin:</b> 2018
<b>Programador responsable:</b> Paul Tandazo	
<b>Descripción:</b> Se debe desarrollar clases, métodos para obtener datos de las direcciones web disponibles en los historiales y que se van generando en tiempo real.	

Tabla 29: Actividad 2 – Historia de Usuario 5 – Reconocimiento e identificación de páginas web visitadas

Elaborado por: Paul Tandazo

<b>Tarea</b>	
<b>Numero: 2</b>	<b>Numero de la Historia: 5</b>
<b>Nombre de la tarea:</b> Generar clase de sitios visitados	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados: 2</b>
<b>Fecha inicio:</b> 2018	<b>Fecha fin:</b> 2018



<b>Programador responsable:</b> Paul Tandazo
<b>Descripción:</b> Se debe adjuntar a cada dirección web visitada campos adicionales como: fecha y hora de la visita, nombre del equipo desde donde se accedió a la dirección y el laboratorio al que pertenece el equipo desde donde se visitó la página.

Historia 6: Almacenamiento de datos web en el Cloud

Tabla 30: Historia de Usuario – Almacenamiento de datos web en el Cloud

Elaborado por: Paul Tandazo

<b>Historia de Usuario</b>	
<b>Numero: 6</b>	<b>Usuario:</b> Todos
<b>Nombre de historia:</b> Almacenamiento de datos web en el Cloud.	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 1.5	<b>Iteración asignada:</b>
<b>Programador responsable:</b> Paul Andres Tandazo	
<b>Descripción:</b> La información generada y recuperada desde los navegadores se debe almacenar en el Cloud, con el propósito de salvaguardar los datos que se vayan obteniendo con las herramientas de monitoreo desarrolladas.	
<b>Observaciones:</b> Debido a que los datos deberán almacenarse y conservarse en el Cloud de la plataforma, se debe dar una alta prioridad al momento de almacenar los datos web producidos con el monitoreo.	

Actividades de la Historia

Tabla 31: Actividad 1 – Historia de Usuario 6 – Almacenamiento de datos web en el Cloud

Elaborado por: Paul Tandazo

<b>Tarea</b>
--------------

<b>Numero: 1</b>	<b>Numero de la Historia: 6</b>
<b>Nombre de la tarea:</b> Establecer conexión al Cloud	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 0.5
<b>Fecha inicio:</b> 2018	<b>Fecha fin:</b> 2018
<b>Programador responsable:</b> Paul Tandazo	
<b>Descripción:</b> Se debe garantizar una conexión eficiente, que permita establecer un medio para realizar operaciones de inserción de datos.	

Tabla 32: Actividad 2 – Historia de Usuario 6 – Almacenamiento de datos web en el Cloud

Elaborado por: Paul Tandazo

<b>Tarea</b>	
<b>Numero: 2</b>	<b>Numero de la Historia: 6</b>
<b>Nombre de la tarea:</b> Guardar datos en el servidor	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 2018	<b>Fecha fin:</b> 2018
<b>Programador responsable:</b> Paul Tandazo	
<b>Descripción:</b> Se debe desarrollar métodos para registrar los datos producidos con el monitoreo web en la base de datos que reside en el Cloud.	

Historia 7: Generación de Reportes

Tabla 33: Historia de Usuario – Generación de Reportes

Elaborado por: Paul Tandazo

<b>Historia de Usuario</b>
----------------------------

<b>Numero: 7</b>	<b>Usuario: Todos</b>
<b>Nombre de historia:</b> Generación de Reportes.	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 7	<b>Iteración asignada:</b>
<b>Programador responsable:</b> Paul Andres Tandazo	
<b>Descripción:</b> El sistema de monitoreo permitirá visualizar reportes en base a los datos generados por el monitoreo y que han sido almacenados en el Cloud desde los programas agentes.	
<b>Observaciones:</b> Debido a que los reportes representaran los datos generados con el monitoreo tanto web como de escritorio, es de alta prioridad generar reportes que se acoplen con el resto de la plataforma.	

Actividades de la Historia

Tabla 34: Actividad 1 – Historia de Usuario 7 – Generación de Reportes

Elaborado por: Paul Tandazo

<b>Tarea</b>	
<b>Numero: 1</b>	<b>Numero de la Historia: 7</b>
<b>Nombre de la tarea:</b> Diseñar estructura del sitio web de monitoreo	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Fecha inicio:</b> 2018	<b>Fecha fin:</b> 2018
<b>Programador responsable:</b> Paul Tandazo	
<b>Descripción:</b> En base a las necesidades del proyecto, se debe estructurar un sitio que garantice el fácil acceso a los datos generados con el monitoreo.	

Tabla 35: Actividad 2 – Historia de Usuario 7 – Generación de Reportes

Elaborado por: Paul Tandazo

<b>Tarea</b>	
<b>Numero: 2</b>	<b>Numero de la Historia: 7</b>
<b>Nombre de la tarea:</b> Diseño de interfaces	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados: 2</b>
<b>Fecha inicio:</b> 2018	<b>Fecha fin:</b> 2018
<b>Programador responsable:</b> Paul Tandazo	
<b>Descripción:</b> Se debe desarrollar interfaces que contengan gráficos que permitan presentar reportes específicos en base a los datos que han sido recolectados por los aplicativos y residen en la Plataforma.	

Tabla 36: Actividad 3 – Historia de Usuario 7 – Generación de Reportes

Elaborado por: Paul Tandazo

<b>Tarea</b>	
<b>Numero: 3</b>	<b>Numero de la Historia: 7</b>
<b>Nombre de la tarea:</b> Obtener datos desde el servidor	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados: 1</b>
<b>Fecha inicio:</b> 2018	<b>Fecha fin:</b> 2018
<b>Programador responsable:</b> Paul Tandazo	
<b>Descripción:</b> Se debe crear métodos que permitan recuperar todos los datos pertenecientes a las actividades informáticas.	

Tabla 37: Actividad 4 – Historia de Usuario 7 – Generación de Reportes

Elaborado por: Paul Tandazo

<b>Tarea</b>	
<b>Numero: 4</b>	<b>Numero de la Historia: 7</b>
<b>Nombre de la tarea:</b> Representar datos en reportes	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados: 2</b>
<b>Fecha inicio:</b> 2018	<b>Fecha fin:</b> 2018
<b>Programador responsable:</b> Paul Tandazo	
<b>Descripción:</b> Implementar librerías graficas utilizadas por la plataforma para representar datos recuperados desde la Plataforma Cloud.	

Historia 8: Integración con la Plataforma CloudIoT

Tabla 38: Historia de Usuario – Integración con la Plataforma CloudIoT

Elaborado por: Paul Tandazo

<b>Historia de Usuario</b>	
<b>Numero: 8</b>	<b>Usuario:</b> Todos
<b>Nombre de historia:</b> Integración con la Plataforma CloudIoT.	
<b>Prioridad en negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b>
<b>Programador responsable:</b> Paul Andres Tandazo	
<b>Descripción:</b> Las páginas web a desarrollarse deberán integrarse con el sitio web que contiene el resto del Proyecto Plataforma CloudIoT.	
Debido a que se trata de una incorporación de más código al ya existente, se deberá	

utilizar las librerías, complementos y demás objetos con los que se desarrolló el resto del proyecto.

**Observaciones:** Debido a que se debe garantizar un fácil acceso a las páginas con los datos del monitoreo, es de alta prioridad unir las páginas a desarrollarse con la Plataforma Cloud.

Actividades de la Historia:

Tabla 39: Actividad 1 – Historia de Usuario 8 – Integración con la Plataforma CloudIoT

Elaborado por: Paul Tandazo

<b>Tarea</b>	
<b>Numero: 1</b>	<b>Numero de la Historia: 8</b>
<b>Nombre de la tarea:</b> Subir paginas al sitio de la Plataforma CloudIoT	
<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 1
<b>Fecha inicio:</b> 2018	<b>Fecha fin:</b> 2018
<b>Programador responsable:</b> Paul Tandazo	
<b>Descripción:</b> Integrar las páginas que mostraran los reportes al resto de la Plataforma CloudIoT.	

Tabla 40: Actividad 2 – Historia de Usuario 8 – Integración con la Plataforma CloudIoT

Elaborado por: Paul Tandazo

<b>Tarea</b>	
<b>Numero: 2</b>	<b>Numero de la Historia: 8</b>
<b>Nombre de la tarea:</b> Agregar funcionalidades de Monitoreo al Menú de la Plataforma	

<b>Tipo de tarea:</b> Desarrollo	<b>Puntos estimados:</b> 2
<b>Fecha inicio:</b> 2018	<b>Fecha fin:</b> 2018
<b>Programador responsable:</b> Paul Tandazo	
<b>Descripción:</b> Se debe incorporar las referencias necesarias al código existente del sitio para permitir un acceso a las páginas que presentaran los reportes.	

#### 4.3.1.3 Iteraciones

Una vez establecidas las historias de usuario, se definieron 4 iteraciones para desarrollar todo el sistema:

Tabla 41: Historias de Usuario por Iteración

Elaborado por: Paul Tandazo

<b>Iteración</b>	<b>Historia de Usuario</b>	<b>Fecha (Desde - Hasta)</b>
<b>1</b>	H1: Capturar actividades informáticas de aplicaciones de escritorio en tiempo real	Desde el 08/02/2018 al 24/08/2018
	H2: Almacenamiento de datos de escritorio en el Cloud	
	H3: Implementación de respaldos para datos fuera de línea.	
	H4: Transferencia de los datos respaldados al servidor	
<b>2</b>	H5: Reconocimiento e identificación de páginas web visitadas	Desde el 04/09/2018 al 31/11/2018
	H6: Almacenamiento de datos web en el Cloud	
	H3: Implementación de respaldos para datos fuera de línea.	

	H4: Transferencia de los datos respaldados al servidor	
<b>3</b>	H7: Generación de Reportes	Desde el 04/12/2018 al 30/12/2018
<b>4</b>	H8: Integración con la Plataforma CloudIoT	Desde el 04/01/2019 al 12/02/2019

#### 4.3.1.4 Estimación de Historias de Usuario

En vista que el sistema de monitoreo necesitara obtener datos y también representarlos, la presente solución se desarrollara por módulos:

- Los primeros módulos constarán de programas agentes que se ejecutarán en segundo plano en sistemas operativos Windows y serán los encargados de alimentar el Cloud con las actividades informáticas que se generen en los equipos que cuenten con el software.
- El segundo módulo únicamente se enfocará en representar los datos recolectados por los agentes en reportes sencillos y que empleen funcionalidades del resto del sitio de la Plataforma CloudIoT.

##### 4.3.1.4.1 Módulo de Monitoreo de Escritorio

Tabla 42: Estimación del módulo de Monitoreo de Escritorio

Elaborado por: Paul Tandazo

No.	Historia de Usuario	Tiempo Estimado		
		Semanas	Días	Horas
<b>1</b>	Capturar actividades informáticas en tiempo real	<b>2</b>	<b>10</b>	<b>40</b>
<b>2</b>	Almacenamiento de datos en el Cloud	<b>1</b>	<b>3</b>	<b>12</b>
<b>3</b>	Implementación de respaldos	<b>1</b>	<b>2</b>	<b>8</b>



4	Transferencia de los datos respaldados al servidor	1	1	4
<b>Tiempo estimado total</b>		<b>2.1</b>	<b>16</b>	<b>64</b>

#### 4.3.1.4.2 Módulo de Monitoreo para la Web

Tabla 43: Estimación del módulo de Monitoreo para la Web

Elaborado por: Paul Tandazo

No.	Historia de Usuario	Tiempo Estimado		
		Semanas	Días	Horas
5	Reconocimiento de páginas web visitadas	2	10	40
6	Ingreso de datos	1	5	20
3	<i>Implementación de respaldos (<u>reutilización de código</u>)</i>	-	-	-
4	<i>Transferencia de los datos respaldados al servidor (<u>reutilización de código</u>)</i>	-	-	-
<b>Tiempo estimado total</b>		<b>3</b>	<b>15</b>	<b>60</b>

#### 4.3.1.4.3 Modulo Web (Reportes)

Tabla 44: Estimación del módulo Web

Elaborado por: Paul Tandazo

No.	Historia de Usuario	Tiempo Estimado		
		Semanas	Días	Horas
7	H7: Generación de Reportes	3	15	60
8	H8: Integración con la Plataforma CloudIoT	1	3	12
<b>Tiempo estimado total</b>		<b>3.1</b>	<b>18</b>	<b>72</b>

## 4.3.2 Diseño

### 4.3.2.1 Tarjetas CRC

Las tarjetas CRC (Clase - Responsabilidad - Colaboración) permite que durante el desarrollo del proyecto se pueda apreciar a plenitud la tecnología de objetos que se pueden generar en base a las historias de usuario. CRC contribuye y se convierte en pilar fundamental del diseño ya que integra en esta etapa a la mayor cantidad de participantes del proyecto.

Tabla 45: Tarjeta CRC – Capturar actividades informáticas en tiempo real

Elaborado por: Paul Tandazo

<b>Capturar actividades informáticas de aplicaciones de escritorio en tiempo real</b>	
<b>Responsabilidades</b>	<b>Colaboradores</b>
Detectar procesos de inicio	Capa de Acceso a Datos
Obtener usuario y dominio del proceso	Clase Win32_ProcessStartTrace
Eliminar procesos de sistema	Clase Win32_ProcessStopTrace
Obtener propiedades del proceso	
Detectar cierre del proceso	
Completar trama de datos a guardar	
<b>Observaciones:</b> La captura de actividades es un proceso automático, que inicia a la par del arranque de la máquina.	

Tabla 46: Tarjeta CRC – Almacenamiento de datos en el Cloud

Elaborado por: Paul Tandazo

**Almacenamiento de datos de escritorio en el Cloud**

<b>Responsabilidades</b>	<b>Colaboradores</b>
Instanciar clase de almacenamiento de datos. Verificar conexión disponible Insertar datos en el servidor	Clase ConfigurationManager Clase MySQLClient Capa de Acceso a Datos Clase Monitoreo_Software_Inicio (creada) Clase Monitoreo_Software_Cierre (creada)
<b>Observaciones:</b> Para asegurar el almacenamiento de datos, es necesario que existan conexiones disponibles.	

Tabla 47: Tarjeta CRC – Implementación de Respaldos

Elaborado por: Paul Tandazo

<b>Implementación de Respaldos para datos fuera de línea</b>	
<b>Responsabilidades</b>	<b>Colaboradores</b>
Crear directorio para archivos temporales Crear archivo de respaldo Adjuntar trama de monitoreo al archivo	Capa de Acceso a Datos Clase System.IO.Directory Clase System.IO.File Clase System.Environment
<b>Observaciones:</b>	

Tabla 48: Tarjeta CRC – Transferencia de los datos respaldados al servidor

Elaborado por: Paul Tandazo

<b>Transferencia de los datos respaldados al servidor</b>
-----------------------------------------------------------

<b>Responsabilidades</b>	<b>Colaboradores</b>
Comprobar existencia de respaldos	Capa de Acceso a Datos
Leer archivo plano de respaldo (traer datos a un objeto del lenguaje de programación)	Clase MySqlConnection
Iniciar transacción para la escritura de datos	Clase ConfigurationManager
Eliminar archivo de respaldo	
<b>Observaciones:</b>	

Tabla 49: Tarjeta CRC – Reconocimiento e de Páginas Web visitadas

Elaborado por: Paul Tandazo

<b>Reconocimiento e identificación de Páginas Web visitadas</b>	
<b>Responsabilidades</b>	<b>Colaboradores</b>
Obtener páginas web visitadas	Clase EquipoLaboratorio (creada)
Recuperar referencias de ultimo sitio visitado (identificador, fecha y hora)	Clase DatosNavegador (creada)
Extraer datos web (guardar temporalmente datos nuevos)	Capa de Acceso a Datos
Recuperar estructura completa (adicionar a cada registro datos como: usuario, equipo, facultad, laboratorio)	Clase System.IO Clase System.Data
Creación de archivos temporales para referencias	
Almacenar referencia del ultimo sitio visitado	
<b>Observaciones:</b>	

Tabla 50: Tarjeta CRC – Almacenamiento de datos en el Cloud

Elaborado por: Paul Tandazo

<b>Almacenamiento de datos en el Cloud</b>	
<b>Responsabilidades</b>	<b>Colaboradores</b>
Comprobar conexión	Clase MySQLClient
Insertar datos en el servidor	Capa de Acceso a Datos
<b>Observaciones:</b>	

Tabla 51: Tarjeta CRC – Generación de Reportes

Elaborado por: Paul Tandazo

<b>Generación de Reportes</b>	
<b>Responsabilidades</b>	<b>Colaboradores</b>
Crear y abrir conexión	Librería java.sql
Preparar sentencia SQL	Librería javax
Ejecutar SQL contra la base de datos	Librería FusionCharts JavaScript
Crear grafico con atributos e información.	
<b>Observaciones:</b>	

Tabla 52: Tarjeta CRC – Integración con la Plataforma CloudIoT

Elaborado por: Paul Tandazo

<b>Integración con la Plataforma CloudIoT</b>	
<b>Responsabilidades</b>	<b>Colaboradores</b>
Comprobar conexión	R_ReporteUsoSoftware (creada)

Acceder a datos	
Ejecutar sentencias SQL	
Presentar reportes en el servidor	
<b>Observaciones:</b>	

### 4.3.3 Desarrollo

#### 4.3.3.1 Base de Datos

Debido a que el sistema de monitoreo debe integrarse al proyecto “Plataforma CloudIoT de control y monitoreo del uso de equipamiento y programas informáticos en aulas y laboratorios”, será necesario crear algunas entidades y tomar referencias de otras ya existentes.

Tomando en referencia las definiciones realizadas en las historias de usuario, para almacenar los datos de las aplicaciones ejecutadas en las computadoras de escritorio se adicione las tablas “Monitoreo Software Inicio – Monitoreo Software Cierre”. En la intención de medir el uso del software instalado en los equipos de los laboratorios, es fundamental considerar un ambiente: en el cual las aplicaciones ejecutadas tienen un inicio y un cierre.

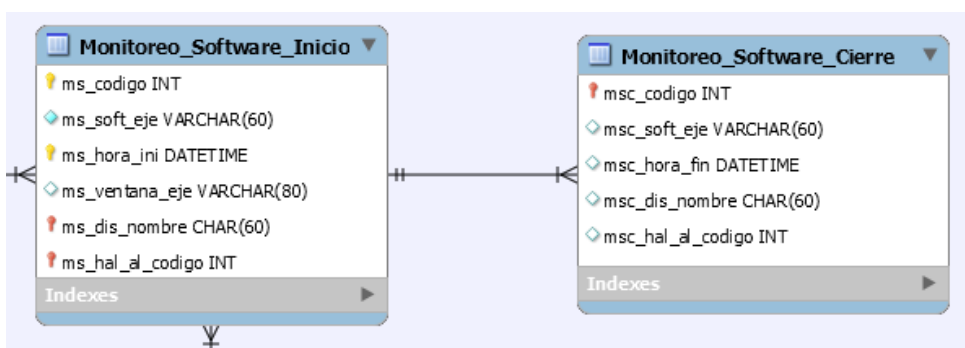


Figura 10: Tablas para almacenar los datos monitoreados

Elaborado por: Paul Tandazo

La tabla “Monitoreo Sitios” se creó con el fin de almacenar las direcciones a las que se vayan accediendo desde los diferentes computadores en donde se instale los programas agentes:

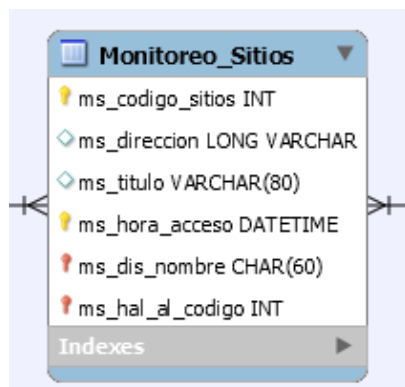


Figura 11: Tabla para el Monitoreo de sitios

Elaborado por: Paul Tandazo

Adicionalmente, se creó la entidad “Software Autorizado” para almacenar los aplicativos que fueron instalados con autorización de los docentes que dictan clases en un determinado laboratorio:

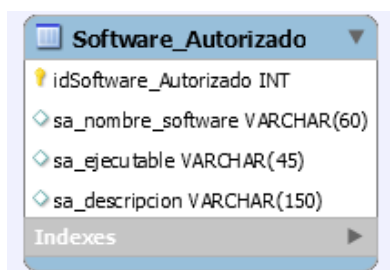


Figura 12: Tabla para el Software Autorizado

Elaborado por: Paul Tandazo

Para determinar que programas están autorizados por los docentes para una determina hora clase, el departamento de laboratoristas implementa las “Actas de Conformidad de Instalación de Software” o “Requerimientos de Software” (Anexos) mediante las cuales los docentes solicitan el software a utilizar durante todo el semestre en sus horas clase. Mediante la entidad “Software Autorizado” y ciertas comparaciones con las entidades “Monitoreo\_Software\_Inicio”, “Monitoreo\_Software\_Cierre” en base de datos es posible identificar que aplicaciones no tienen relación con las horas clases dictadas en un determinado laboratorio.

De acuerdo con las políticas de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, la manera para solicitar un software que será utilizado durante un ciclo académico se realiza a través de las actas de conformidad de software o requerimientos de software. Por ejemplo, durante el periodo académico marzo – agosto (2018) los programas solicitados por docentes fueron:

Tabla 53: Programas solicitados para horas clase

Elaborado por: Paul Tandazo

Software	Docentes	Laboratorios
Packet Tracer	Ing. David Guevara Ing. Pilar Urrutia,	Redes 1
Virtual Box	Ing. David Guevara Ing. Frankil Mayorga Ing. Julio Balarezo Ing. Hernán Naranjo	Redes 1 Lab 07 Lab 05 Lab 04
Oracle Apex	Ing. Hernando Buenaño	Lab 07
Oracle 11g	Ing. Hernando Buenaño Ing. Carlos Nuñez	Lab 07 Lab05
Office	Ing. Victor Guachibomsa	Lab 07
Project Profesional	Ing. Victor Guachibomsa Ing. Carlos Nuñez	Lab 07 Lab 06 Lab 04
Octave	Ing. Alexis Sánchez Ing. Ruben Nogales	Lab 07 Lab 05
Netbeans	Ing. Carlos Nuñez Ing. Ruben Nogales	Lab 06
Xampp	Ing. Carlos Nuñez	Lab 06
Visual Studio	Ing. Hernán Naranjo Ing. Clay Aldas	Lab 06
SQL Server	Ing. Hernán Naranjo Ing. Clay Aldas	Lab 06
PostgreSQL	Ing. Hernán Naranjo	Lab 06
Google Chrome	Ing. Hernán Naranjo	Lab 06
Visual Studio Code	Ing. Clay Aldas	Lab 06
Note Pad ++	Ing. Clay Aldas	Lab 06
Oracle Enterprise Manager	Ing. Carlos Nuñez	Lab 05
Star UML	Ing. Dennis Chicaiza	Lab 05
Visual Prolog	Ing. Hernando Buenaño	Lab 05

En base a las necesidades expuestas en el desarrollo de la investigación, los programas solicitados se consideran como “Software Autorizado” debido a que son un método ya comprobado por empleados de la facultad para conocer que software instalado en los diferentes equipos tiene un fin educativo.



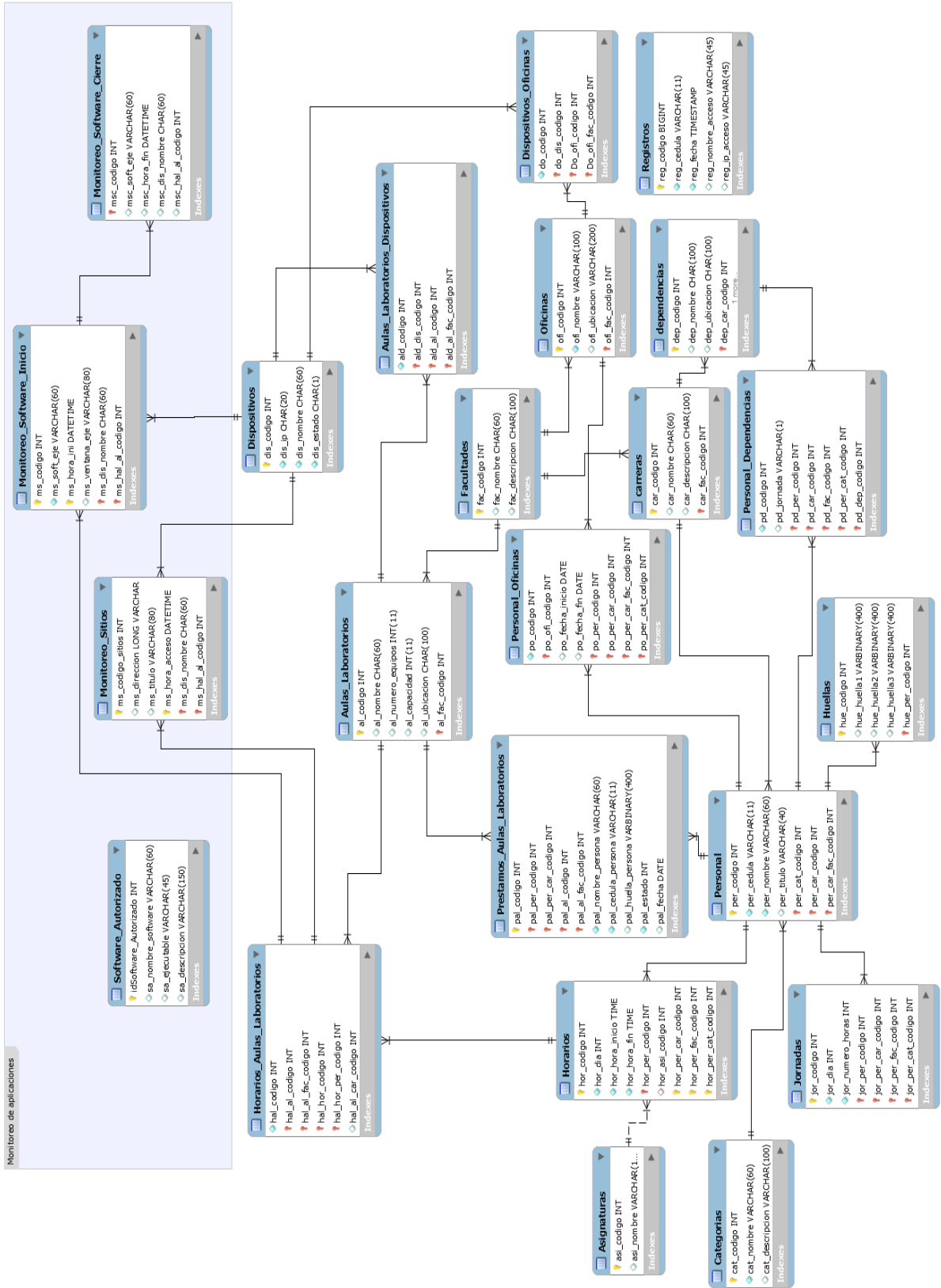


Figura 13: Diagrama completo de la Plataforma CloudIoT

Elaborado por: Paul Tandazo

#### 4.3.3.2 Generación de Actividades Informáticas

Para el desarrollo del proyecto se debe tomar muy en cuenta que las actividades informáticas constituyen información específica que se genera cuando en un equipo de un laboratorio se inicia un programa o se accede a cualquier página web. En base a los requerimientos descritos en las historias de usuario el sistema deberá capturar y almacenar las siguientes tramas de datos:

	Código del programa ejecutado	Nombre del software ejecutado	Hora de inicio y cierre del programa	Nombre de la ventana	Nombre del equipo	Laboratorio
<i>Longitud máxima</i>	11	216	19	256	60	11

Figura 14: Trama de datos a capturar en el monitoreo de escritorio

Elaborado por: Paul Tandazo

	Id de la página web	Dirección Web	Título de la página web	Fecha y hora de acceso	Nombre del equipo	Laboratorio
<i>Longitud máxima</i>	11	2.083	80	19	60	11

Figura 15: Trama de datos a capturar en el monitoreo de web

Elaborado por: Paul Tandazo

#### 4.3.3.3 Almacenamiento de respaldo

El almacenamiento de datos de la aplicación en tiempo real depende de la conexión a la red, existe la posibilidad que dicha conexión no se encuentre disponible en un momento determinado, por lo que se necesitara respaldar los datos en el disco local del equipo en el cual se instaló la aplicación de monitoreo. Para determinar la mejor alternativa se optó por comparar el rendimiento de la aplicación en ambientes

similares, guardar 5513 tuplas entre una base de datos SQLite (misma que es usada por navegadores para historiales o dispositivos móviles) y un archivo plano:

Tabla 54: Comparación entre SQLite – CSV

Elaborado por Paul Tandazo

Respaldo	SQLite	Archivo Plano CSV	Mejor elección
Numero de datos	5513		
Creación del Archivo	766 ms	11 ms	CSV
Tiempo de escritura	1.234 ms	757 ms	CSV
Memoria de proceso	6,893 s 41,2 MB	3,863 s 38,7 MB	CSV
CPU (% procesadores)	5,435 s 4 veces en 50%	3,143 s 3 veces en 50%	CSV
Tamaño del archivo en disco	1.264 KB	1.191 KB	CSV
Rangos de crecimiento en Memoria de Proceso	15,9 - 18,8 – 19,2 – 30,3 – 30,4 – 34,9 – 37,1 – 40,8 – 41,2 – 31,6 – 30,9 MB	18,8 – 19,0 – 30,1 – 30,4 – 32,5 – 38,7 – 28,6 MB	CSV
Manejo de datos	Propiedades de Base de datos	Sin propiedades de base de datos.	SQLite

Al realizarse las pruebas, se obtuvieron datos que guardan mucha concordancia con un escenario pronosticado, SQLite requiere de 766 ms para crear el archivo ya que adicionalmente se realiza la creación de la base de datos y la tabla que servirá para almacenar la información, mientras que el CSV solamente tarda 11 ms ya que únicamente crea un archivo plano.

Para escribir los 5513 datos en la base SQLite, el software tarda mucho más tiempo que en el archivo CSV, debido a que se debe abrir una conexión permanente hasta completar la escritura total de los datos en el destino. Otro efecto que provoca mantener una conexión abierta es el consumo de memoria y el consumo en porcentaje de los procesadores, que es más alto para el caso de SQLite.

Al requerir una estructura de datos más compleja, el archivo SQLite tiene un tamaño mayor del 5.77% respecto al documento CSV.



Figura 16: Imagen de Rangos de crecimiento en Memoria de Proceso para SQLite

Elaborado por: Paul Tandazo



Figura 17: Imagen de Rangos de crecimiento en Memoria de Proceso para CSV

Elaborado por: Paul Tandazo

Debido a que el sistema de monitoreo debe ser imperceptible y causar el menor efecto posible, se debe tomar en cuenta el impacto que la aplicación tiene en el rendimiento del equipo. Luego de las pruebas realizadas en tiempo de desarrollo, se determinó que la mejor alternativa por tiempos de respuesta es respaldar la información en un CSV (archivo delimitado por comas).

#### 4.3.3.4 Almacenamiento de datos

El almacenamiento de los datos que son de importancia y relevancia para el desarrollo del presente proyecto dependerán del motor de base de datos con el que cuenta y en el que está alojado el resto de la Plataforma CloudIoT.

Tabla 55: Servidor de Base de datos Plataforma CloudIoT

Elaborado por: Paul Tandazo

Servidor	Localhost a través de socket UNIX
Tipo de servidor	MariaDB
Versión del servidor	5.5.56-MariaDB - MariaDB Server
Versión del protocolo	10

Debido a que MariaDB fue elaborado por los desarrolladores de MySQL. Las librerías, clases y controladores utilizados en la presente investigación conservan la denominación de MySQL pero su funcionamiento no se modifica con MariaDB.

### 4.3.3.5 Codificación

#### 4.3.3.5.1 Primera Iteración

En este punto se desarrollará el software que se encargará de monitorear las aplicaciones de escritorio. De acuerdo con las historias de usuario, el programa tendrá como propósito identificar las aplicaciones que se ejecuten en un momento determinado y almacenar dicha información en una base de datos. Para la identificación de aplicaciones iniciadas, C# dentro de Visual Studio tiene librerías integradas que permiten determinar la información de las aplicaciones se han iniciado y han finalizado en el equipo, a través de las clases *Win32\_ProcessStartTrace* y *Win32\_ProcessStopTrace* respectivamente que tienen la siguiente estructura:

Tabla 56: Propiedades de Win32\_ProcessStartTrace y Win32\_ProcessStopTrace

Elaborado por: Paul Tandazo

Propiedad	Tipo de dato	Win32_ProcessStartTrace	Win32_ProcessStopTrace
SECURITY_DESCRIPTOR	uint8	X	X
TIME_CREATED	uint64	X	X
ProcessID	uint32	X	X
ParentProcessID	uint32	X	X
Sid[]	uint8	X	X
ExitStatus	uint32	-	X
ProcessName	string	X	X

SessionID	uint32	X	X
-----------	--------	---	---

Para poder acceder y recuperar la definición de las clases Win32\_ProcessStartTrace y Win32\_ProcessStopTrace que son eventos de la clase WMI se puede utilizar la clase ManagementEventWatcher que permite recibir notificaciones en base a como ocurran los eventos consultados con el método EventArrived, tal como si se tratara de una consulta SQL como se muestra en el siguiente código:

```
ManagementEventWatcher startWatch = new ManagementEventWatcher(
    new WqlEventQuery("SELECT * FROM
Win32_ProcessStartTrace"));
startWatch.EventArrived += new
EventArrivedEventHandler(startWatch_EventArrived);
startWatch.Start();

ManagementEventWatcher stopWatch = new ManagementEventWatcher(
    new WqlEventQuery("SELECT * FROM Win32_ProcessStopTrace"));
stopWatch.EventArrived += new
EventArrivedEventHandler(stopWatch_EventArrived);
stopWatch.Start();
startWatch.Stop();
stopWatch.Stop();
```

Los datos requeridos en las historias de usuario se pueden obtener creando métodos startWatch\_EventArrived - stopWatch\_EventArrived en donde se obtiene el identificador único de los procesos iniciados y finalizados respectivamente:

```
static void startWatch_EventArrived(object sender, EventArrivedEventArgs e)
{
    int idProceso = Convert.ToInt32(e.NewEvent.Properties["ProcessID"].Value);
    ...

static void stopWatch_EventArrived(object sender, EventArrivedEventArgs e)
{
    int idProceso = Convert.ToInt32(e.NewEvent.Properties["ProcessID"].Value);
    ...
```

Una vez que se tiene el identificador (idProceso) se puede usar el método GetProcessById() de la clase Process para recuperar el resto de los datos necesarios, como el nombre del software ejecutado (MsSoftEje), ventana (MsVentanaEje) y hora (MsHoraIni) en que se inicia el proceso:

```

using (Process localById = Process.GetProcessById(idProceso))
{
    using (DatosMS procesoEjecutado = new DatosMS())
    {
        //Ingresa al IF: Si el proceso a ejecutarse no se encuentra en la
        lista
        if (datosFiltrar == null)
        {
            procesoEjecutado.MsCodigoSoft = localById.Id;
            procesoEjecutado.MsSoftEje = localById.ProcessName;
            procesoEjecutado.MsHoraIni = localById.StartTime;
            procesoEjecutado.MsVentanaEje =
(localById.MainWindowTitle.Replace(',', ' ').Replace('\', ' '));
            procesoEjecutado.MsDisNombre = DominioWin + "-" + UsuarioWin;
            procesoEjecutado.MsHalAlCodigo = ObtenerLaboratorio();
        }
    }
}
...

```

Para obtener los datos de los procesos finalizados no se puede utilizar el método `GetProcessById` ya que el proceso deja de existir para la clase `Process`, por tal motivo se obtiene datos en base a como ocurren los eventos dentro del método `stopWatch_EventArrived`:

```

using (DatosMSCierre informacionCierre = new DatosMSCierre())
{
    //Console.WriteLine("nombre del proceso de cierre a guardar..." +
    nombreProceso);
    informacionCierre.MsCodigoSoft = idProceso;
    informacionCierre.MsSoftEje =
e.NewEvent.Properties["ProcessName"].Value.ToString();
    informacionCierre.MsHoraFin = DateTime.Now;
    informacionCierre.MsDisNombre = DominioWin + "-" + UsuarioWin;
    informacionCierre.MsHalAlCodigo = ObtenerLaboratorio();
}
...

```

Para almacenar los datos en el servidor de la plataforma, tanto para la información de los procesos capturados iniciados como finalizados se emplea la clase `DatosDAL` que cuenta con el método `InsertarDatosServidor` el cual recibe como parámetro una instancia de las clases `DatosMS` - `DatosMSCierre` (instanciadas y completadas anteriormente). En caso de que la aplicación no encuentre conexión con el servidor, inmediatamente se almacenara los datos en un archivo plano con extensión `.CSV`, como se presenta en el siguiente código:

```

using (DatosDAL datosDAL = new DatosDAL())
{
    //Insertar el dato en el servidor
    bool bandera = datosDAL.InsertarDatosServidor(procesoEjecutado);

    //Respalda información en un CSV
    if (!bandera)
    {
        List<string> temporal = new List<string>();
        temporal.Add(procesoEjecutado.getData());
        datosDAL.AlmacenarRespaldo(temporal);
        temporal = null;
    }
}

```

#### 4.3.3.5.2 Segunda Iteración

El desarrollo del módulo que estará encargado de recolectar los datos que se generan a partir de las aplicaciones utilizadas para navegar por internet, se realizara mediante análisis y extracción directa de información desde este tipo de programas.

El historial generado por la aplicación navegador se extrae en un objeto DataTable mediante el método ExtraerDesdeTabla(), en dicha extracción de información primero se envía parámetros a fin de recuperar únicamente las páginas web visitadas a partir de la última fecha en que se ejecutó el software. Como DataTable internamente contiene filas, si durante la recuperación arroja un valor mayor a 0 en el conteo, se entiende que se han generado nuevos datos y solo entonces será necesario entrar a insertar los datos en el servidor:

```

DataTable datosNuevos = navegador.ExtraerDesdeTabla(idTemp, fecTemp);

if (datosNuevos.Rows.Count > 0)
{
    bool insercion =
navegador.InsertarDatosServidor(navegador.RecuperarEstructuraCompleta(datosNuevos));
    //Llamar al metodo para insertar datos directamente en el servidor.
    if (insercion)
    {
        navegador.UltimoSitio();
    }
    else
    {
        //Si falla la transaccion, los datos se respaldan en un CSV.
        navegador.AlmacenarRespaldo(navegador.ObtenerHistorial(datosNuevos));
        navegador.UltimoSitio();
    }
}
}

```



En la extracción del historial se consulta directamente con la base de datos SQLite que utiliza el programa navegador para mantener los datos. Al realizar la consulta se extrae en un DataTable información como id, dirección (url), título y fecha de cada página visitada, siempre obteniendo datos en donde la fecha de la última visita sea mayor que la fecha en que corre el programa, evitando así datos redundantes:

```
public DataTable ExtreerDesdeTabla(int identificador, string fecha)
{
    try
    {
        conexion.Open();
        SQLiteDataAdapter sd = new SQLiteDataAdapter("select visits.id as id,
urls.url as link, urls.title as titulo, " +
            " datetime(visits.visit_time / 1000000 + (strftime('%s',
'1601-01-01')), 'unixepoch', 'localtime') as fecha " +
            " from urls, visits" +
            " where urls.id = visits.url " +
            " and visits.id != " + identificador +
            " and datetime(visits.visit_time / 1000000 + (strftime('%s',
'1601-01-01')), 'unixepoch', 'localtime') > '" + fecha + "'"; ", conexion);

        DataSet datos = new DataSet();
        sd.Fill(datos);
        return datos.Tables[0];
    }
    catch (SQLiteException excepcion)
    {
        throw excepcion;
    }
    finally
    {
        conexion.Close();
    }
}
```

Una vez que se ha recuperado los datos desde el navegador, es necesario adicionar información extra requerida en las historias de usuario que son propias por cada equipo, como el nombre de la máquina y el laboratorio en donde se ubica la misma, para ello, el método RecuperarEstructuraCompleta recibe como parámetro un DataTable y se lo completa con las columnas requeridas para finalmente poder retornar un DataTable con los datos completos:

```

public DataTable RecuperarEstructuraCompleta(DataTable infoInicial)
{
    string equipo = equipoLaboratorio.Equipo;
    string lab;
    int labFinal;
    try
    {
        lab = equipoLaboratorio.Laboratorio;
        labFinal = Convert.ToInt32(lab);
    }
    catch (Exception ex)
    {
        lab = "1234";
        labFinal = Convert.ToInt32(lab);
    }
    DataTable dt = infoInicial;

    dt.Columns.Add("equipo", typeof(System.String));
    dt.Columns.Add("laboratorio", typeof(System.Int32));
    dt.Columns.Add("navegador", typeof(System.String));

    foreach (DataRow row in dt.Rows)
    {
        row["equipo"] = equipo; // or set it to some other value
        row["laboratorio"] = labFinal;
        row["navegador"] = "Google Chrome";
    }
    return dt;
}

```

#### 4.3.3.5.3 Tercera Iteración

El desarrollo de la tercera iteración se enfoca netamente, a los reportes que se deben generar a partir de los datos capturados por los módulos de monitoreo para aplicaciones informáticas, debido a que el sistema se debe acoplar o integrar a un sistema ya existe, es necesario adaptarse al diseño de interfaces ya establecido en la creación de la plataforma, tal como se muestra en la siguiente figura:

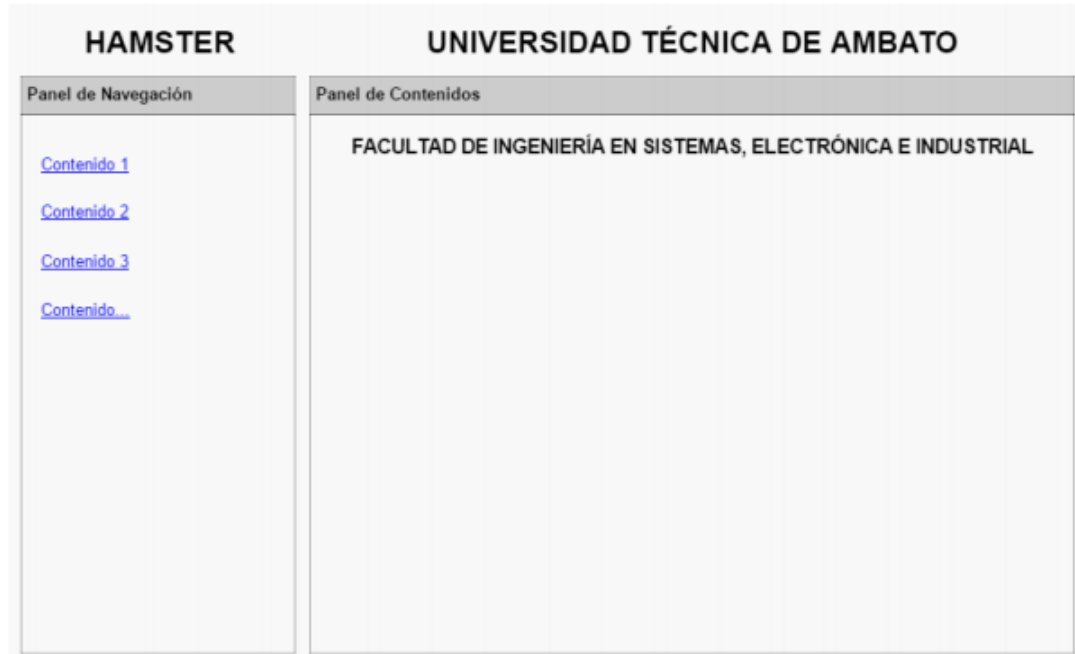


Figura 18: Prototipo de interfaz de usuario general del sistema web

Elaborado por: Andres Tisalema [50]

Debido a que la plataforma está desarrollada en JAVA, para la tercera iteración que solamente constituye los reportes, se empleará Netbeans como IDE de programación. El código generado permite conectarse con el servidor utilizando clases ya existentes, para luego únicamente mediante sentencias SQL acceder a los datos almacenados en las tablas descritas anteriormente, realizar comparaciones y representar los resultados de las consultas mediante la librería grafica FusionCharts Suite XT, dichos resultados que previamente se almacenan en ArrayList y mediante métodos se enlazan al objeto de la librería gráfica.

```
// Sentencia SQL a ejecutarse contra la base de datos.
String sql = "SELECT pe.Carrera as cod_car, ca.car_nombre as car_nombre,
sum(pe.Tiempo_Ejecucion) as Tie_Eje "
+ "FROM programas_escritorio pe, carreras ca "
+ "where pe.Facultad = 1 "
+ "and pe.Carrera = ca.car_codigo GROUP BY pe.Carrera";

// Ejecutar la consulta.
java.sql.Statement psd = cn.createStatement();
ResultSet rs= psd.executeQuery(sql);

// El objeto 'chartobj' permite mantener los atributos y datos del grafico.
Map<String, String> chartobj = new HashMap<String, String>();

chartobj.put("caption", "Uso de Software");
chartobj.put("subCaption" , "");
```

```

chartobj.put("xAxisName", "Carreras");
chartobj.put("yAxisName", "Tiempo de uso (minutos)");
// Crear los ArrayList para mantener la informacion.
ArrayList arrData = new ArrayList();
ArrayList linkeddata = new ArrayList();
while(rs.next())
{
    // Crear el map para extraer los datos consultados.
    Map<String, String> lv = new HashMap<String, String>();
    lv.put("label", rs.getString("car_nombre"));
    lv.put("value", rs.getString("Tie_Eje"));
    lv.put("link", "newchart-json-" + rs.getString("cod_car"));
    arrData.add(lv);

    // Crear el objeto linkedDataObj for subinformes
    Map<String, String> linkedDataObj = new HashMap<String, String>();

    // Pasar el código necesario para el los subinformes
    linkedDataObj.put("id", rs.getString("cod_car"));
    ...
}
// Cerrar la conexion.
rs.close();

// Crear el objeto 'dataMap' para completar el reporte.
Map<String, String> dataMap = new LinkedHashMap<String, String>();

// Asociar gráfico, información con los objetos que mantienen la información
dataMap.put("chart", gson.toJson(chartobj));
dataMap.put("data", gson.toJson(arrData));
dataMap.put("linkeddata", gson.toJson(linkeddata));

```

#### 4.3.3.5.4 Cuarta Iteración

En la cuarta iteración se incorpora las páginas de reportes al resto de la plataforma. Para llevar a cabo este proceso únicamente es necesario acceder al servidor con dirección 172.21.123.50 y subir a la carpeta que aloja al resto de los recursos web de la plataforma las nuevas páginas generadas en la iteración 3, al estar desarrolladas en el mismo proyecto la integración no representa mayores inconvenientes.

Tabla 57: Servidor Web Plataforma CloudIoT

Elaborado por: Paul Tandazo

Tipo	Apache / 2.4.6 (CentOS) PHP / 5.4.16
Versión del cliente de la base de datos	libmysql - 5.5.56-MariaDB
Extensión de PHP	mysqli
Versión de PHP	5.4.16

#### 4.3.4 Pruebas

En el cumplimiento de XP es fundamental como paso final realizar las pruebas a todo el código desarrollado a fin de conocer si el software realiza lo que se planifico y solicitó en las historias de usuario.

Tabla 58: Prueba de aceptación 1 – Ejecución en segundo plano

Elaborado por: Paul Tandazo

<b>Prueba de Aceptación</b>	
<b>Numero:</b> 1	<b>Historia de Usuario #1:</b> Capturar actividades informáticas de aplicaciones de escritorio en tiempo real
<b>Nombre:</b> Ejecución en Segundo Plano	
<b>Descripción:</b>  Una vez que se encienda el equipo y el usuario inicie sesión, el programa se ejecutara automáticamente con las credenciales del usuario específico.	
<b>Condiciones de ejecución:</b> <ul style="list-style-type: none"><li>• Inicio del equipo</li><li>• Conexión a la Plataforma CloudIoT</li></ul>	
<b>Entrada:</b>  No se ingresa ningún tipo de información, la aplicación en sí misma no requiere intervención directa del usuario.	
<b>Resultado esperado:</b>  La aplicación aparece listada en los procesos de segundo plano del Administrador de Tareas.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria	

Tabla 59: Prueba de aceptación 2 – Obtener inicio de las aplicaciones ejecutadas

Elaborado por: Paul Tandazo

<b>Prueba de Aceptación</b>	
<b>Numero:</b> 2	<b>Historia de Usuario #1:</b> Capturar actividades informáticas de aplicaciones de escritorio en tiempo real
<b>Nombre:</b> Obtener inicio de aplicaciones ejecutadas	
<b>Descripción:</b>  La aplicación empieza a obtener datos conforme el usuario va ejecutando aplicaciones instaladas o ejecutables.	
<b>Condiciones de ejecución:</b>  <ul style="list-style-type: none"> <li>• La aplicación de monitoreo ya se debe estar ejecutando.</li> </ul>	
<b>Entrada:</b>  No se requiere intervención del usuario.	
<b>Resultado esperado:</b>  Se obtiene el identificador, nombre, titulo (en caso de que se genere una ventana), fecha y hora de la aplicación iniciada.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria	

Tabla 60: Prueba de aceptación 3 – Obtener cierre de aplicaciones ejecutadas

Elaborado por: Paul Tandazo

<b>Prueba de Aceptación</b>	
<b>Numero:</b> 3	<b>Historia de Usuario #1:</b> Capturar actividades informáticas de aplicaciones de escritorio en tiempo real

<b>Nombre:</b> Obtener cierre de aplicaciones ejecutadas
<b>Descripción:</b>  La aplicación obtiene datos conforme el usuario va cerrando aplicaciones instaladas o ejecutables.
<b>Condiciones de ejecución:</b>  <ul style="list-style-type: none"> <li>• La aplicación de monitoreo ya se debe estar ejecutando.</li> </ul>
<b>Entrada:</b>  No se requiere intervención del usuario.
<b>Resultado esperado:</b>  Se obtiene el identificador, nombre, fecha y hora del cierre de la aplicación.
<b>Evaluación de la prueba:</b> Prueba satisfactoria

Tabla 61: Prueba de aceptación 4 – Ingreso de datos con conexión

Elaborado por: Paul Tandazo

<b>Prueba de Aceptación</b>	
<b>Numero:</b> 4	<b>Historia de Usuario #2:</b> Almacenamiento de datos de escritorio en el Cloud
<b>Nombre:</b> Ingresar de datos con conexión	
<b>Descripción:</b>  Una vez que las entidades de inicio y cierre de las aplicaciones han sido completadas, las tuplas con los datos se ingresaran en la base de datos.	
<b>Condiciones de ejecución:</b>  <ul style="list-style-type: none"> <li>• Aplicación de monitoreo en ejecución</li> </ul>	

<ul style="list-style-type: none"> <li>• Tener conexión a internet</li> </ul>
<p><b>Entrada:</b></p> <p>El método de ingreso recibe información de las aplicaciones una vez que han iniciado y también cuando han finalizado.</p>
<p><b>Resultado esperado:</b></p> <p>Los datos se escriben en las tablas creadas en la base de datos de la plataforma.</p>
<p><b>Evaluación de la prueba:</b> Prueba satisfactoria</p>

Tabla 62: Prueba de aceptación 5 - Seleccionar alternativa para respaldos

Elaborado por: Paul Tandazo

<b>Prueba de Aceptación</b>	
<b>Numero:</b> 5	<b>Historia de Usuario #3:</b> Implementación de respaldos para datos fuera de línea
<b>Nombre:</b> Seleccionar alternativa para respaldos	
<p><b>Descripción:</b></p> <p>Para elegir una manera adecuada de almacenar datos fuera de línea se debe guardar copias de los datos en el equipo local que va generando la información.</p>	
<p><b>Condiciones de ejecución:</b></p> <ul style="list-style-type: none"> <li>• Datos generados en la máquina de pruebas.</li> <li>• Conexión al servidor de base de datos.</li> </ul>	
<p><b>Entrada:</b></p> <p>Los datos generados por la máquina de prueba se almacenan en archivos fuera de línea.</p>	
<b>Resultado esperado:</b>	



Escritura correcta en archivos de respaldo.

Medir tiempos de escritura en todos los casos considerados.

**Evaluación de la prueba:** Prueba satisfactoria

Tabla 63: Prueba de aceptación 6 – Ingreso de datos sin conexión

Elaborado por: Paul Tandazo

Prueba de Aceptación	
<b>Numero: 6</b>	<b>Historia de Usuario #3:</b> Implementación de respaldos para datos fuera de línea
<b>Nombre:</b> Ingresar de datos sin conexión	
<b>Descripción:</b>  Una vez que las entidades de inicio y cierre de las aplicaciones han sido completadas y se intente guardar las tuplas con los datos, se puede dar el caso en el que la conexión no se encuentra disponible, por lo que se necesita mantener los datos en archivos locales.	
<b>Condiciones de ejecución:</b> <ul style="list-style-type: none"><li>• Aplicación de monitoreo en ejecución</li></ul>	
<b>Entrada:</b>  El método de ingreso recibe información de las aplicaciones una vez que han iniciado y también cuando han finalizado.	
<b>Resultado esperado:</b>  Los datos se escriben en archivos planos con extensión CSV.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria	

Tabla 64: Prueba de aceptación 7 - Almacenamiento de datos guardados fuera de línea

Elaborado por: Paul Tandazo

<b>Prueba de Aceptación</b>	
<b>Numero: 7</b>	<b>Historia de Usuario #4:</b> Transferencia de los datos respaldados al servidor
<b>Nombre:</b> Almacenamiento de datos guardados fuera de línea	
<b>Descripción:</b>  Una vez que las aplicaciones comienzan a ejecutarse, comprueban conexión con el servidor e inmediatamente transfieren los generados fuera de línea a la base de datos en el Cloud.	
<b>Condiciones de ejecución:</b>  <ul style="list-style-type: none"> <li>• Archivos generados fuera de línea (CSV en el equipo local)</li> </ul>	
<b>Entrada:</b>  No se requiere intervención del usuario, aun así el método internamente recibe los archivos CSV y automáticamente guarda los datos en la Plataforma.	
<b>Resultado esperado:</b>  Los datos se escriben en la Plataforma y los archivos planos de respaldo con extensión CSV se eliminan del equipo.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria	

Tabla 65: Prueba de aceptación - Obtener datos generados desde navegadores web

Elaborado por: Paul Tandazo

<b>Prueba de Aceptación</b>
-----------------------------

<b>Numero: 8</b>	<b>Historia de Usuario #5:</b> Reconocimiento e identificación de páginas web visitadas
<b>Nombre:</b> Obtener datos generados desde navegadores web	
<b>Descripción:</b>  La aplicación empieza a obtener datos conforme el usuario vaya accediendo a páginas web desde los navegadores previamente instalados.	
<b>Condiciones de ejecución:</b>  <ul style="list-style-type: none"> <li>• La aplicación de monitoreo ya se debe estar ejecutando.</li> </ul>	
<b>Entrada:</b>  No se requiere intervención del usuario.	
<b>Resultado esperado:</b>  La trama de datos se completa con campos adicionales por cada dirección web y queda lista para ser almacenada en la Plataforma	
<b>Evaluación de la prueba:</b> Prueba satisfactoria	

Tabla 66: Prueba de aceptación 9 - Ingresar de datos web con conexión

Elaborado por: Paul Tandazo

<b>Prueba de Aceptación</b>	
<b>Numero: 9</b>	<b>Historia de Usuario #6:</b> Almacenamiento de datos web en el Cloud
<b>Nombre:</b> Ingresar de datos web con conexión	
<b>Descripción:</b>  Una vez que la trama con los datos web se ha completado las tuplas generadas se	

ingresaran en la base de datos.
<p><b>Condiciones de ejecución:</b></p> <ul style="list-style-type: none"> <li>• Aplicación de monitoreo en ejecución</li> <li>• Tener conexión a internet</li> </ul>
<p><b>Entrada:</b></p> <p>El método de ingreso recibe información de las paginas accedidas conforme el usuario vaya utilizando un determinado navegador.</p>
<p><b>Resultado esperado:</b></p> <p>Los datos se escriben en las tablas creadas en la base de datos de la plataforma, caso contrario entra en acción la implementación de respaldos.</p>
<p><b>Evaluación de la prueba:</b> Prueba satisfactoria</p>

Tabla 67: Prueba de aceptación 10 - Mostrar gráficos de los datos obtenidos

Elaborado por: Paul Tandazo

<b>Prueba de Aceptación</b>	
<b>Numero: 10</b>	<b>Historia de Usuario #7:</b> Generación de Reportes
<b>Nombre:</b> Mostrar gráficos de los datos obtenidos	
<p><b>Descripción:</b></p> <p>El sistema debe permitir generar reportes en los cuales se muestre el uso del software en determinadas hora clase.</p>	
<p><b>Condiciones de ejecución:</b></p> <ul style="list-style-type: none"> <li>• Datos almacenados en el servidor de base de datos</li> <li>• Tener conexión a internet</li> </ul>	
<b>Entrada:</b>	

El método accede a la información almacenada y en base a parámetros previamente definidos abstrae información que se considera relevante.
<p><b>Resultado esperado:</b></p> <p>Los gráficos generados se rigen a los datos presentes en el servidor de la Plataforma.</p>
<p><b>Evaluación de la prueba:</b> Prueba satisfactoria</p>

Tabla 68: Prueba de aceptación 11 - Paginas disponibles en el sitio de la Plataforma

Elaborado por: Paul Tandazo

<b>Prueba de Aceptación</b>	
<b>Numero: 11</b>	<b>Historia de Usuario #8:</b> Integración con la Plataforma CloudIoT
<b>Nombre:</b> Paginas disponibles en el sitio de la Plataforma	
<p><b>Descripción:</b></p> <p>El sistema debe integrarse a la Plataforma CloudIoT por tal motivo las páginas web desarrolladas deberán estar en el sitio del proyecto.</p>	
<p><b>Condiciones de ejecución:</b></p> <ul style="list-style-type: none"> <li>• Archivos que generan los reportes</li> <li>• Tener conexión a internet</li> </ul>	
<b>Entrada:</b> El usuario accede a los reportes ingresando la dirección del servidor de la Plataforma.	
<p><b>Resultado esperado:</b></p> <p>Se muestran los archivos y reportes con la información almacenada en la base de datos del proyecto.</p>	
<p><b>Evaluación de la prueba:</b> Prueba satisfactoria</p>	

## **CAPITULO 5**

### **CONCLUSIONES Y RECOMENDACIONES**

#### **5.1 Conclusiones**

- El sistema de monitoreo permitió capturar de manera imperceptible y causando el menor efecto posible sobre el rendimiento del equipo datos que se generaron a partir del inicio y cierre de las aplicaciones que fueron utilizadas por los estudiantes en sistemas operativos Windows dentro de los laboratorios de la facultad.
- Con la recolección de datos en un determinado periodo de tiempo, se comprobó que los módulos tanto de escritorio como web pudieron establecer conexión con el servidor de la Plataforma y almacenaron datos de manera normal. datos recolectados permitieron representar gráficamente el uso que tiene el software en una determinada hora clase de la facultad.
- Durante la etapa de desarrollo, los módulos que componen al sistema de monitoreo Open Source consideran el uso de una metodología ágil como una medida comprobada para ejecutar una correcta interpretación de los requerimientos solicitados por el cliente en la fase de planificación.
- La información generada a partir del monitoreo pudo ser representada gráficamente mediante reportes que contribuyen en gran medida a determinar el uso que tiene el software en una determinada hora clase en los laboratorios de la facultad, fundamentados en el funcionamiento total de la Plataforma CloudIoT.
- Con la implementación de páginas web y su integración con el resto de la Plataforma CloudIoT, en el servidor 172.21.123.50 los datos almacenados están disponibles en base a ciertos parámetros útiles que miden determinados aspectos fundamentales y de importancia para el trabajo de investigación.
- Mediante las pruebas realizadas bajo autorización de la facultad, se capturaron actividades informáticas durante 13 horas clase en 5 equipos

diferentes, de los datos recolectados se determinó que 4,37 horas son utilizadas en la ejecución de software no autorizado, es decir programas informáticos fuera de las actas de conformidad de instalación de software. Al aplicar una línea de tendencia basada en una función polinómica grado 5 sobre los datos obtenidos se determinó que el tiempo de uso podrían aumentar a medida que se dicten más horas clase, también es importante recalcar que la función utilizada no se ajusta de manera adecuada a los datos por la irregularidad y la poca representatividad de estos en torno al global de horas clase dictadas durante un semestre.

- Por otro lado, los datos generados a partir de software autorizado representan 34,28 horas de ejecución, es decir se comprobó que el tipo de software que más se utilizó en el laboratorio de pruebas fue el solicitado por los docentes.

## **5.2 Recomendaciones**

- Debido a que el sistema de monitoreo es Open Source, a futuro es posible adicionar más datos a la trama que se captura en cada actividad informática, sin embargo, se deberá medir que el impacto de la escalabilidad no sea negativo o afecte el rendimiento normal de la aplicación.
- Los requerimientos para generar otros reportes en el sitio de la Plataforma CloudIoT podrían variar o incrementar en el tiempo, por ello la incorporación de nuevos gráficos se deberá hacer respetando la estructura y diseño del proyecto en general.
- El despliegue de las aplicaciones de monitoreo por un tiempo mucho más prolongado permitirá obtener un mayor volumen de datos y de ese modo generar un análisis de datos más exhaustivo a raíz de la implementación de herramientas como Big Data.
- Para poder establecer algún tipo de relación, en el uso que tiene el software y su incidencia en la tasa de retención, es necesario capturar datos por un tiempo más prolongado, de modo que se permita encontrar un polinomio que se ajuste de mejor manera a la función dada en base a la información capturada con el monitoreo y así se pueda encontrar una tendencia acorde a los datos almacenados en el servidor del proyecto.

## Bibliografía

- [1] F. Barrera-Osorio y L. Leigh, «ABDUL LATIF JAMEEL POVERTY ACTION LAB,» 2009. [En línea]. Available: <https://www.povertyactionlab.org/es/evaluation/el-uso-y-mal-uso-de-computadores-en-educaci%C3%B3n-evidencia-de-una-prueba-de-evaluaci%C3%B3n-aleat>. [Último acceso: 27 Septiembre 2017].
- [2] R. Correa, «El Comercio,» 22 Mayo 2017. [En línea]. Available: [http://www.elcomercio.com/uploads/files/2017/05/24/Decreto-1425-2017\\_mayo\\_prelacion.pdf](http://www.elcomercio.com/uploads/files/2017/05/24/Decreto-1425-2017_mayo_prelacion.pdf). [Último acceso: 27 Septiembre 2017].
- [3] Altablero, «Ministerio de Educación Superior,» Abril 2004. [En línea]. Available: <http://www.mineducacion.gov.co/1621/article-87403.html>. [Último acceso: 12 Octubre 2017].
- [4] G. D. Ruiz Mesias, *Sistema de control y monitoreo para mejorar los procesos de administración de los laboratorios de las carreras de Sistemas, Electrónica e Industrial en la FISEI-UTA*, Ambato, 2014.
- [5] F. Galarza Molina, «Universidad Politécnica Salesiana,» 2007. [En línea]. Available: <http://dspace.ups.edu.ec/handle/123456789/737>. [Último acceso: 22 Mayo 2017].
- [6] N. Alechina, J. Halpern, I. Kash y L. Brian, «Decentralised Norm Monitoring in Open Multi-Agent Systems,» *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*, pp. 1399-1400, 2016.
- [7] N. Alechina, M. Dastani y B. Logan, «Norm approximation for imperfect monitors,» *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*, n° 14, pp. 117-124, 2014.
- [8] M. Kutare, C. W. Eisenhauer, K. Schwan, V. Talwar y M. Wolf, «Monalytics: online monitoring and analytics for managing large scale data centers,» *Proceedings of the 7th international conference on Autonomic computing*, pp. 141-150, 2010.
- [9] G. Taylor y Y. He, «In-network, push-based network resource monitoring: scalable, responsive network management,» *Proceedings of the Third International Workshop on Network-Aware Data Management*, pp. 8-16, 2013.
- [10] R. L. Galva, «A tool for monitoring the public administration in Dominican Republic: SISMAP,» *Proceedings of the 5th International Conference on Theory and Practice of Electronic Governance*, n° 11, pp. 371-372, 2011.
- [11] J. Cook, H. Sharifi y A. Farrahi, «Towards production monitoring of application progress,» *Proceedings of the 4th International Workshop on Software Engineering for Computational Science and Engineering*, n° 11, pp. 56-57, 2011.
- [12] d. C. Pablos, J. López, S. Martín y S. Medina, *Informática y comunicaciones en la empresa*, Madrid: ESIC, 2004.



- [13] Real Academia Española, «Diccionario de la lengua española,» 2017. [En línea]. Available: <http://dle.rae.es/?id=0chgoNb>. [Último acceso: 22 Mayo 2017].
- [14] Diccionario de la lengua española, «Real Academia Española,» 2017. [En línea]. Available: <http://dle.rae.es/?id=LY8zQy3>. [Último acceso: 22 Mayo 2017].
- [15] A. Srinivasan, «Study.com,» [En línea]. Available: <https://study.com/academy/lesson/process-in-operating-systems-definition-scheduling-states.html>. [Último acceso: 20 Febrero 2018].
- [16] Studytonight, «Studytonight Library,» 2018. [En línea]. Available: <https://www.studytonight.com/operating-system/operating-system-processes>. [Último acceso: 20 Febrero 2018].
- [17] iMonitor Software, «iMonitorSoft TM,» 2016. [En línea]. Available: <http://es.imonitorsoft.com/employee-monitoring-software.html>. [Último acceso: 22 Mayo 2017].
- [18] NIST, «National Institute of Standards and Technology,» Agosto 2006. [En línea]. Available: <https://csrc.nist.gov/Glossary/?term=269>. [Último acceso: 20 Octubre 2017].
- [19] M. d. C. Romero, «Departamento de Tecnología Electrónica,» 2009. [En línea]. Available: <http://www.dte.us.es/personal/mcromero/masredes/docs/SMARD.0910.qos.pdf>. [Último acceso: 20 Octubre 2017].
- [20] Infortelecom, «Infortelecom.es,» 29 Septiembre 2016. [En línea]. Available: <https://infortelecom.es/blog/que-es-un-servidor-y-para-que-sirve/>. [Último acceso: 21 Febrero 2018].
- [21] Grupo Aner, «Onyx Systems,» [En línea]. Available: <http://www.onyxsystems.es/que-es-un-servidor.html>. [Último acceso: 20 Febrero 2018].
- [22] M. V. Nevado Cabello, Introducción a Las Bases de Datos Relacionales, Madrid: Vision Libros, 2006.
- [23] J. Niño, Sistemas operativos en red, Editex, 2011.
- [24] J. Pérez, «ORACLE,» Enero 2016. [En línea]. Available: <http://www.oracle.com/technetwork/es/articles/cloudcomp/oracle-cloud-dbaas-2877305-esa.html>. [Último acceso: 23 Mayo 2017].
- [25] Developer Network, «Microsoft,» 2013. [En línea]. Available: <https://msdn.microsoft.com/es-es/library/bb885185.aspx>. [Último acceso: 22 Mayo 2017].
- [26] D. Network, «Microsoft,» 2016. [En línea]. Available: <https://msdn.microsoft.com/es-es/library/ff519554.aspx>. [Último acceso: 22 Mayo 2017].
- [27] D. Network, «Microsoft,» 2016. [En línea]. Available: <https://msdn.microsoft.com/es-es/library/dd239314.aspx>. [Último acceso: 22 Mayo 2017].

- [28] D. Network, «Microsoft,» 2010. [En línea]. Available: [https://msdn.microsoft.com/es-es/library/cc488539\(v=vs.100\).aspx](https://msdn.microsoft.com/es-es/library/cc488539(v=vs.100).aspx). [Último acceso: 22 Mayo 2017].
- [29] M. Ojeda, «DSpace JSPUI,» [En línea]. Available: <http://cdigital.uv.mx/bitstream/123456789/5009/2/198801P121.pdf>. [Último acceso: 19 Julio 2017].
- [30] Z. Cataldi, «Repositorio Institucional de la UNLP,» 5 Diciembre 2003. [En línea]. Available: [http://sedici.unlp.edu.ar/bitstream/handle/10915/4055/3\\_-\\_La\\_ingenier%C3%ADa\\_de\\_software.pdf?sequence=7](http://sedici.unlp.edu.ar/bitstream/handle/10915/4055/3/_La_ingenier%C3%ADa_de_software.pdf?sequence=7). [Último acceso: 19 Julio 2017].
- [31] F. Duarte y F. Pires, «Universitat de Barcelona,» 1 Agosto 2011. [En línea]. Available: <http://www.ub.edu/geocrit/aracne/aracne-150.htm>. [Último acceso: 19 Julio 2017].
- [32] D. d. Sistemas, «Universidad Católica los Ángeles de Chimbote,» 2017. [En línea]. Available: <https://www.uladech.edu.pe/images/stories/universidad/documentos/2018/metodologia-desarrollo-software-v001.pdf>. [Último acceso: 3 Marzo 2018].
- [33] P. Letelier y C. Penadés, «Repositorio institucional de la Universidad de Las Tunas,» [En línea]. Available: <http://roa.ult.edu.cu/bitstream/123456789/477/1/masyxp.pdf>. [Último acceso: 8 Marzo 2018].
- [34] R. Figueroa, C. Solis y A. Cabrera, «ResearchGate,» [En línea]. Available: [https://www.researchgate.net/publication/299506242\\_METODOLOGIAS\\_TRADICIONALES\\_VS\\_METODOLOGIAS\\_AGILES](https://www.researchgate.net/publication/299506242_METODOLOGIAS_TRADICIONALES_VS_METODOLOGIAS_AGILES). [Último acceso: 6 Marzo 2018].
- [35] «Conecta Software,» 2018. [En línea]. Available: <https://conectasoftware.com/sobrenosotros/metodologia-implantacion-agil-vs-metodologia-tradicional>. [Último acceso: 5 Marzo 2018].
- [36] J. García, «SOMECE Tecnología en Educación,» [En línea]. Available: [http://simposio.somece.org.mx/2006/contenido/grupo3/pdf/6\\_GarciaAlvarezJoseLuis.pdf](http://simposio.somece.org.mx/2006/contenido/grupo3/pdf/6_GarciaAlvarezJoseLuis.pdf). [Último acceso: 20 Marzo 2018].
- [37] CIDECAME, «Universidad Autónoma del Estado de Hidalgo,» [En línea]. Available: [http://cidecame.uaeh.edu.mx/lcc/mapa/PROYECTO/libro17/12\\_etapas.html](http://cidecame.uaeh.edu.mx/lcc/mapa/PROYECTO/libro17/12_etapas.html). [Último acceso: 8 Marzo 2018].
- [38] EcuRed, «EcuRed,» [En línea]. Available: [https://www.ecured.cu/Modelo\\_espiral](https://www.ecured.cu/Modelo_espiral). [Último acceso: 7 Marzo 2018].
- [39] H. H. O. Castillo, «Piktochart,» [En línea]. Available: <https://create.piktochart.com/output/23824799-grupo-02>. [Último acceso: 8 Marzo 2018].
- [40] Wells, Ron, [En línea]. Available: <http://www.extremeprogramming.org/rules/customer.html>. [Último acceso: 25 Junio 2018].

- [41] «Finances Online,» 11 Marzo 2018. [En línea]. Available: <https://comparisons.financesonline.com/netbeans-vs-visual-studio-ide>. [Último acceso: 6 Mayo 2018].
- [42] «Finances Online,» 19 Diciembre 2018. [En línea]. Available: <https://reviews.financesonline.com/p/visual-studio-ide/>. [Último acceso: 7 Mayo 2018].
- [43] «Finances Online,» 19 Diciembre 2018. [En línea]. Available: <https://reviews.financesonline.com/p/netbeans/>. [Último acceso: 4 Mayo 2018].
- [44] D. Wells, «Extreme Programming,» 8 Octubre 2013. [En línea]. Available: <http://www.extremeprogramming.org/>. [Último acceso: 4 Junio 2018].
- [45] K. Kendall y J. Kendall, Analisis y Diseño de Sistemas, Ciudad de Mexico: Pearson Education, 2005.
- [46] Don Wells, «Extreme Programming,» [En línea]. Available: <http://www.extremeprogramming.org/rules/userstories.html>. [Último acceso: 8 Junio 2018].
- [47] Wells, Don, «Extreme Programming,» [En línea]. Available: <http://www.extremeprogramming.org/rules/simple.html>. [Último acceso: 20 Junio 2018].
- [48] D. Wells, «Extreme Programming,» [En línea]. Available: <http://www.extremeprogramming.org/rules/crccards.html>. [Último acceso: 21 Junio 2018].
- [49] Wells, Don, «Extreme Programming,» [En línea]. Available: <http://www.extremeprogramming.org/rules/functionaltests.html>. [Último acceso: 28 Junio 2018].
- [50] A. Tisalema, «Repositorio Universidad Tecnica de Ambato,» Agosto 2017. [En línea]. Available: <http://repo.uta.edu.ec/handle/123456789/26222>. [Último acceso: 10 Enero 2019].
- [51] Mell, Peter; Grance, Timothy, «National Institute of Standars and Technology,» Septiembre 2011. [En línea]. Available: <http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf>. [Último acceso: 20 Octubre 2017].
- [52] Velázquez, Jesus, «Northware,» 29 Noviembre 2012. [En línea]. Available: <https://www.northware.mx/desarrollo-en-cascada-waterfall-vs-desarrollo-agile-scrum/>. [Último acceso: 7 Marzo 2018].
- [53] «Blogdiario,» Marzo 2016. [En línea]. Available: <http://marich.blogspot.es/1459397526/metodologia-de-prototipos/>. [Último acceso: 8 Marzo 2018].

# Anexos y Apéndices

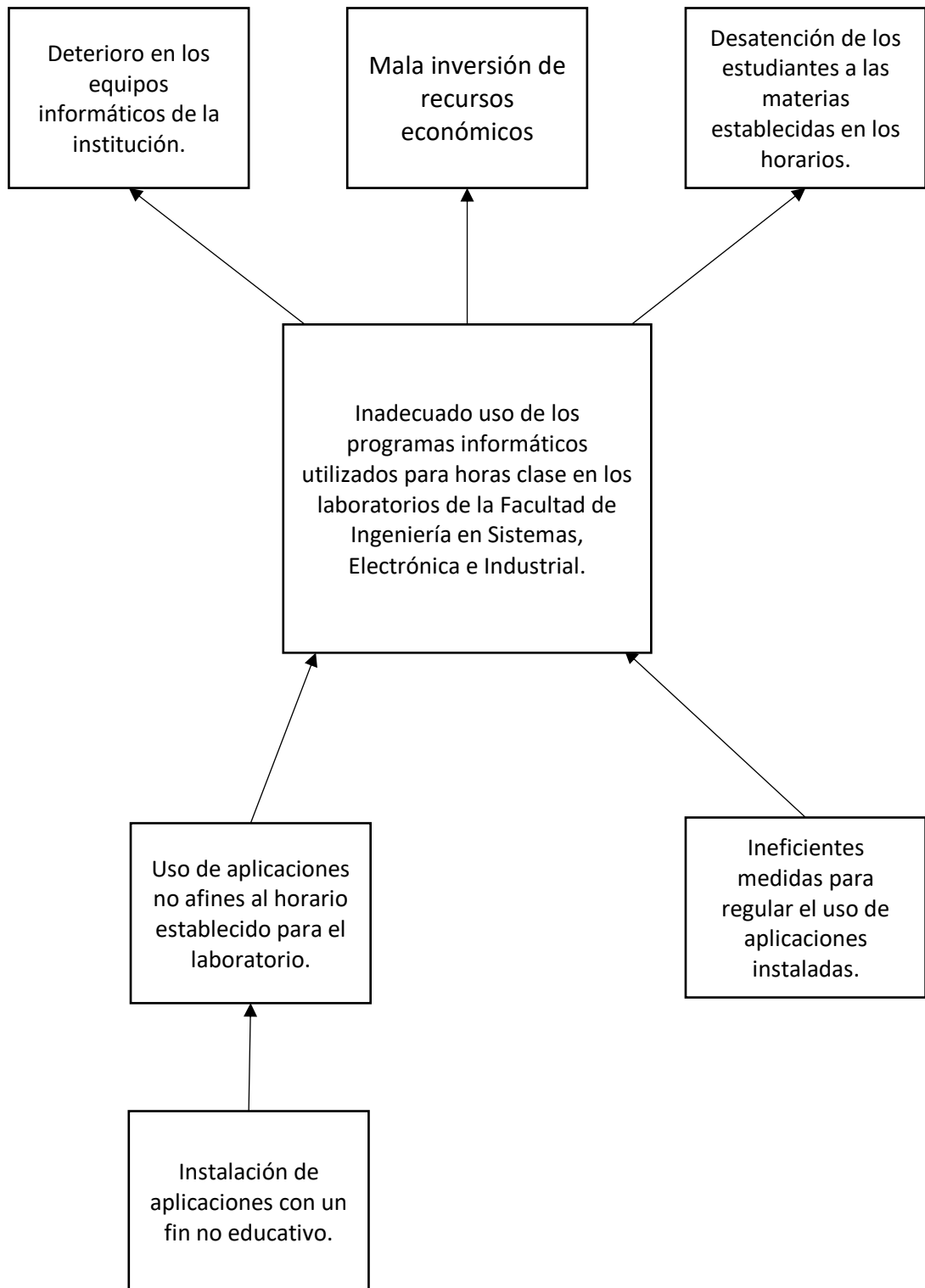


Figura 19: Árbol de Problemas

Elaborado por: Paul Tandazo

## Estudio y análisis de herramientas comerciales para monitorear actividades en aplicaciones Desktop

Actualmente en el mercado existe varias aplicaciones para monitorear las actividades que se generan en un equipo informático, en términos generales las aplicaciones listas para descargar convergen en el mismo procedimiento base: un sistema madre que controla que monitorear y los agentes, pequeños programas que registran la información que se produce en el equipo, como se describe en la figura 22:

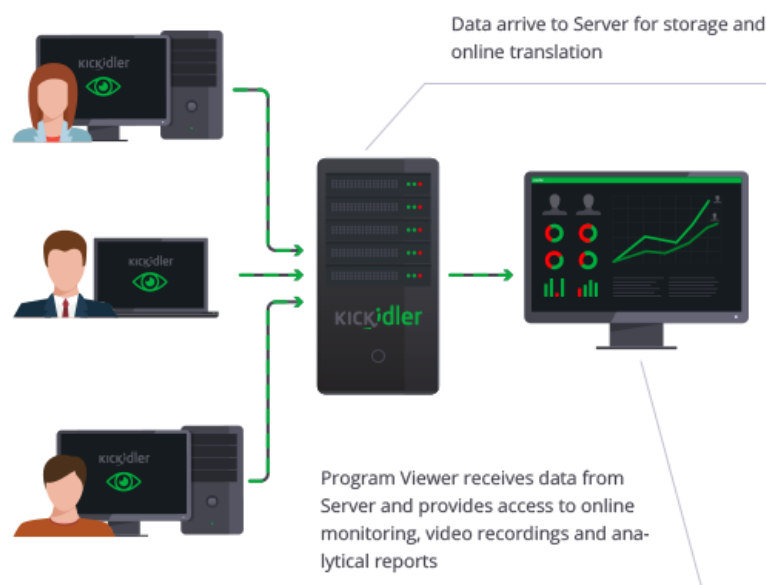


Figura 20: Esquema general de monitoreo

Elaborado por: Kickidler.

Tabla 69: Comparativa de herramientas de monitoreo comerciales

Elaborado por: Paul Tandazo

Kickidler	iMonitor EAM	ActivTrak
<ul style="list-style-type: none"> <li>• Ver y capturar la pantalla del equipo en cualquier momento.</li> <li>• Mostrar todos los equipos en los que está instalado el software de monitoreo.</li> <li>• Agregar equipos de manera ilimitada.</li> <li>• Informes de tiempo de trabajo.</li> <li>• Monitoreo en línea: Obtener información en tiempo real sobre las aplicaciones que se ejecutan en un equipo.</li> <li>• Grabación de video de actividades.</li> <li>• Seguimiento y registro del tiempo de trabajo.</li> <li>• Keylogger.</li> <li>• DLP: Prevención de fugas de datos / Protección de pérdida de datos: Mecanismo contra la explotación de datos</li> </ul>	<ul style="list-style-type: none"> <li>• Monitoreo de 360 grados: Permite grabar cada actividad, incluyendo keylogger, actividades en el portapapeles, capturas de pantalla, correos electrónicos, uso de almacenamiento, FTP, búsqueda de sitios web, descarga de archivos y etc.</li> <li>• Solución de Monitoreo Centralizado: permite monitorear más de 1000 ordenadores con un solo servidor.</li> <li>• Prevención de potenciales amenazas internas: Identificar el origen desde donde se copian datos a dispositivos, se permite monitorear cada correo electrónico, correo web.</li> <li>• Alertas en tiempo real y Alertas de Palabras Clave: Detección de palabras preestablecidas en las</li> </ul>	<ul style="list-style-type: none"> <li>• Fácil de instalar y usar: Solo se necesita instalar el agente ya sea de forma manual o remota en el cliente.</li> <li>• Solución de Monitoreo en tiempo real: Sin lagunas de cobertura permitiendo ver un flujo de datos en tiempo real.</li> <li>• Capturas: Se puede conocer exactamente lo que se hace en la maquina donde esté instalado el agente mediante la recolección de imágenes en un intervalo de 10 segundos.</li> <li>• Reportes de uso: Informes para contribuir a la productividad.</li> <li>• Bloqueo: Rechazar o permitir el acceso a cualquier página web.</li> <li>• Alarmas: Definición de condiciones para desencadenar acciones (capturas, notificaciones, finalización del proceso).</li> <li>• Seguimiento de USB: Proteger la información</li> </ul>



<p>confidenciales al mundo exterior.</p> <ul style="list-style-type: none"> <li>• Control remoto de equipos.</li> <li>• Análisis de productividad.</li> <li>• Basado en SQL para almacenamiento de datos.</li> </ul>	<p>pulsaciones de teclado, nombres de archivo, URL de sitios web, títulos de sitios web, y captura de pantalla del contenido.</p> <ul style="list-style-type: none"> <li>• Registra actividades de internet.</li> <li>• Visualiza y Controla Remotamente.</li> <li>• Características de Reportes Extensos.</li> <li>• Características de Filtros Flexibles.</li> </ul>	<p>del equipo de amenazas internas.</p> <ul style="list-style-type: none"> <li>• Puntuación de riesgo: Determinar la cantidad de riesgo que está relacionada con un determinado comportamiento.</li> <li>• Bloqueo de sitios web.</li> <li>• Monitoreo Programado.</li> <li>• Acceso de datos sin procesar.</li> <li>• Seguimiento personalizado: establecer un monitoreo de acuerdo con horarios útiles de la organización.</li> <li>• Exportar registro de actividad en CSV.</li> <li>• Acceso de datos SQL.</li> </ul>
<p>Viewer:</p> <ul style="list-style-type: none"> <li>• Procesador de 1 GHz, 1GB de RAM, 1 Mbit/s tráfico de red con el servidor.</li> </ul> <p>Server:</p> <ul style="list-style-type: none"> <li>• Procesador de 1 GHz + 100 MHz por cada viewer, RAM de 1GB + 50 Mb por cada viewer, 500 Kbits/s en tráfico de red</li> </ul>	<p>Computadora cliente:</p> <ul style="list-style-type: none"> <li>• Procesador de 500 MHz, 265 MB de RAM, 1GB en disco</li> </ul> <p>Computadora servidor:</p> <ul style="list-style-type: none"> <li>• Procesador de 2 GHz, 2048 MB de RAM, 10 GB en disco duro</li> </ul>	<p>Computador cliente:</p> <ul style="list-style-type: none"> <li>• Procesador de 1GHz, 1GB de RAM, 3GB en disco.</li> </ul> <p>Servidor:</p> <ul style="list-style-type: none"> <li>• Navegador actualizado, equipo con capacidades altas para navegar en Internet.</li> </ul>



por cada viewer.		
Trial: 14 días en un equipo, adicionalmente obtener información en 6 equipos.	15 días para pruebas, en dos equipos.	Instalación en 3 equipos – almacenamiento de 3 GB.
Licencia por año 19 agentes: \$ 855.	Licencia 19 agentes: \$ 1,306.44.	Licencia por año 19 agentes: \$ 1,641.40.

El mayor inconveniente para usar cualquiera de las herramientas disponibles en el mercado radica en el precio, los valores aumentan dependiendo de los agentes que se desea implementar, en promedio un agente en el mercado tiene un valor de 67 dólares. Otra consideración importante al momento de usar cualquiera de las aplicaciones comerciales indicadas en la tabla es que el sistema no es escalable, tampoco se puede obtener más información que la considerada por el desarrollador inicial del producto software, la limitación para crecer en base a las futuras necesidades del proyecto está marcada por la falta de independencia en el código.

## Actas de conformidad de instalación de software

 <b>UNIVERSIDAD TÉCNICA DE AMBATO</b> FACULTAD DE...Sistemas... <b>ACTA DE CONFORMIDAD DE INSTALACIÓN DE SOFTWARE</b> 							
GENERALIDADES							
En la ciudad de Ambato, a los ..... días del mes de <del>Febrero</del> del 2018, siendo las ....., en presencia del <del>José...Clay...Aldas...</del> , Coordinador de la Carrera de <del>Sistemas</del> ....., se procede a verificar por parte del mismo que se encuentra instalado y funcionando correctamente el siguiente software de acuerdo a las características solicitadas.							
N	N° Documento Solicitante	Docente Solicitante	Software Instalado	Versión	Lab. N°	Fecha Instalación	Observación
1	//	Ing. David Guevara	Packet tracer	7.0	Redes 1	12/03/18	
2	//	Ing. David Guevara	Virtual Box		Redes 1	12/03/18	
3							
4							
5							
6							

Para constancia firman:

  
 Coordinador

Administrador de Redes

Figura 21: Acta de conformidad - Laboratorio Redes 1

Elaborado por: Administrador de redes FISEI

 <b>UNIVERSIDAD TÉCNICA DE AMBATO</b> FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL <b>ACTA DE CONFORMIDAD DE INSTALACIÓN DE SOFTWARE</b> 							
GENERALIDADES							
En la ciudad de Ambato, a los 16 días del mes de Marzo del 2018, siendo las ....., en presencia del ....., Coordinador de la Carrera de ....., se procede a verificar por parte del mismo que se encuentra instalado y funcionando correctamente el siguiente software de acuerdo a las características solicitadas.							
N	N° Documento Solicitante	Docente Solicitante	Software Instalado	Versión	Lab. N°	Fecha Instalación	Observación
1		Ing. Franklin Mayorga	Virtual Box Putty	5.2 0.7	7	13/03/2018	Instalado Centos 6.5 y Windows Server 2008 en Virtual Box
2		Ing. Pilar Urrutia	Packet Tracer	7.1	7	13/03/2018	
3		Ing. Carlos Núñez	Netbeans xampp	7.0.1 1.6.2	7	13/03/2018	
4		Ing. Hernando Buenaño	Oracle Apex Oracle 11g	4.0 11g Enterprise	7	13/03/2018	
5		Ing. Víctor Guachimbosa	Office Project Profesional	2016 2013	7	13/03/2018	
6		Ing. Alexis Sánchez	Octave	4.2.1	7	13/03/2018	

Para constancia firman:

  
 Coordinador

Administrador de Redes

Figura 22: Acta de conformidad - Laboratorio 7

Elaborado por: Administrador de redes FISEI

 <b>UNIVERSIDAD TÉCNICA DE AMBATO</b> FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL <b>ACTA DE CONFORMIDAD DE INSTALACIÓN DE SOFTWARE</b> 							
GENERALIDADES							
En la ciudad de Ambato, a los 16 días del mes de Marzo del 2018, siendo las ....., en presencia del ....., Coordinador de la Carrera de ....., se procede a verificar por parte del mismo que se encuentra instalado y funcionando correctamente el siguiente software de acuerdo a las características solicitadas.							
N	N° Documento Solicitante	Docente Solicitante	Software Instalado	Versión	Lab. N°	Fecha Instalación	Observación
1		Ing. Carlos Núñez	Netbeans Xampp Office-Project	7.0.1 1.6.2 2016	6	12/03/2018	
2		Ing. Hernán Naranjo	Visual Studio SQL Server Pstgres Google Chrome	2017 2016 10.2	6	12/03/2018	
3		Ing. Clay Aldás	Visual Studio SQL Server Visual Studio Code Note Pad ++	2017 2016 3.0.0	6	12/03/2018	

Para constancia firman:

  
 \_\_\_\_\_  
 Coordinador

\_\_\_\_\_  
 Administrador de Redes

Figura 23: Acta de conformidad - Laboratorio 6

Elaborado por: Administrador de redes FISEI

 <b>UNIVERSIDAD TÉCNICA DE AMBATO</b> FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL <b>ACTA DE CONFORMIDAD DE INSTALACIÓN DE SOFTWARE</b> 							
GENERALIDADES							
En la ciudad de Ambato, a los 16 días del mes de Marzo del 2018, siendo las ....., en presencia del ....., Coordinador de la Carrera de ....., se procede a verificar por parte del mismo que se encuentra instalado y funcionando correctamente el siguiente software de acuerdo a las características solicitadas.							
N	N° Documento Solicitante	Docente Solicitante	Software Instalado	Versión	Lab. N°	Fecha Instalación	Observación
1		Ing. Rubén Nogales	Octave	4.2.1	5	08/03/2018	
2		Ing. Julio Balarezo	Virtual Box	5.2	5	08/03/2018	
3		Ing. Carlos Núñez	Oracle 11g Oracle Enterprise Manager Oracle Developer	11g Web	5	08/03/2018	
4		Ing. Dennis Chicaiza	Star UML	2.8.1	5	08/03/2018	
5		Ing. Hernando Buenaño	Visual Prolog	5.2	5	08/03/2018	



Para constancia firman:

  
 \_\_\_\_\_  
 Coordinador

\_\_\_\_\_  
 Administrador de Redes

Figura 24: Acta de conformidad - Laboratorio 5

Elaborado por: Administrador de redes FISEI

 <b>UNIVERSIDAD TÉCNICA DE AMBATO</b> FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL <b>ACTA DE CONFORMIDAD DE INSTALACIÓN DE SOFTWARE</b> 							
GENERALIDADES							
En la ciudad de Ambato, a los 16 días del mes de Marzo del 2018, siendo las ....., en presencia del ....., Coordinador de la Carrera de ....., se procede a verificar por parte del mismo que se encuentra instalado y funcionando correctamente el siguiente software de acuerdo a las características solicitadas.							
N	N° Documento Solicitante	Docente Solicitante	Software Instalado	Versión	Lab. N°	Fecha Instalación	Observación
1		Ing. Rubén Nogales	C++(Api Code Block) Netbeans Bambit	17.12 7.0.1-8.0.1 15.1.1	4	06/03/2018	
2		Ing. Hernán Naranjo	Virtual Box	5.2	4	06/03/2018	
3		Ing. Carlos Núñez	Office-Project	2016	4	06/03/2018	
4		Ing. Sandra Carrillo	MySQL Workbench	6.3	4	06/03/2018	
5		Ing. Franklin Mayorga	Mendeley Desktop JabRef Zotero Netbeans Pencil Project Brackets Simple CSS	1.17 2.10 5.0 8.0.1 1.3 1.6 2.1	4	06/03/2018	
6		Ing. Julio Balarezo	Virtual Box	5.2	4	06/03/2018	
7		Ing. Félix Fernández	Kali Linux (Virtual Box) Wamp(Virtual Box)	2018 3.0.6	4	06/03/2018	

Para constancia firman:

  
 \_\_\_\_\_  
 Coordinador

\_\_\_\_\_  
 Administrador de Redes

Figura 25: Acta de conformidad - Laboratorio 4

Elaborado por: Administrador de redes FISEI

 <b>UNIVERSIDAD TÉCNICA DE AMBATO</b> FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL <b>ACTA DE CONFORMIDAD DE INSTALACIÓN DE SOFTWARE</b> 							
GENERALIDADES							
En la ciudad de Ambato, a los 16 días del mes de Marzo del 2018, siendo las ....., en presencia del ....., Coordinador de la Carrera de ....., se procede a verificar por parte del mismo que se encuentra instalado y funcionando correctamente el siguiente software de acuerdo a las características solicitadas.							
N	N° Documento Solicitante	Docente Solicitante	Software Instalado	Versión	Lab. N°	Fecha Instalación	Observación
1		Ing. Edison Álvarez	SQL Server 2014	Enterprise	3	05/03/2018	
2		Ing. Julio Balarezo	SQL Server 2014 Virtual Box	Enterprise 5.2.6	3	05/03/2018	
3		Ing. Sandra Carrillo	MySQL Workbench	6.3 CE	3	05/03/2018	
4		Ing. Víctor Guachimposa	Proyect Profesional	2013 español	3	05/03/2018	

Para constancia firman:

  
 \_\_\_\_\_  
 Coordinador

\_\_\_\_\_  
 Administrador de Redes

Figura 26: Acta de conformidad - Laboratorio 3

Elaborado por: Administrador de redes FISEI

 <b>UNIVERSIDAD TÉCNICA DE AMBATO</b> FACULTAD DE... <i>Sistemas</i> ..... <b>ACTA DE CONFORMIDAD DE INSTALACIÓN DE SOFTWARE</b> 							
GENERALIDADES							
En la ciudad de Ambato, a los ..... días del mes de <i>Noviembre</i> del 201 <i>8</i> , siendo las ....., en presencia del Ing. <i>Clay Aldas</i> ....., Coordinador de la Carrera de <i>Tecnologías de la I.</i> , se procede a verificar por parte del mismo que se encuentra instalado y funcionando correctamente el siguiente software de acuerdo a las características solicitadas.							
N	N° Documento Solicitante	Docente Solicitante	Software Instalado	Versión	Lab. N°	Fecha Instalación	Observación
1	<i>15</i>	<i>Ing. Osvaldo Paredes</i>	<i>OpenShot</i>	<i>24.1</i>	<i>1</i>	<i>05/03/18</i>	
2							
3							
4							
5							
6							



Para constancia firman:

  
 \_\_\_\_\_  
 Coordinador

\_\_\_\_\_  
 Administrador de Redes

Figura 27: Acta de conformidad - Laboratorio 1

Elaborado por: Administrador de redes FISEI

 <b>UNIVERSIDAD TÉCNICA DE AMBATO</b> FACULTAD DE... <i>Sistemas</i> ..... <b>ACTA DE CONFORMIDAD DE INSTALACIÓN DE SOFTWARE</b> 							
GENERALIDADES							
En la ciudad de Ambato, a los ..... días del mes de <i>Noviembre</i> del 201 <i>8</i> , siendo las ....., en presencia del Ing. <i>Clay Aldas</i> ....., Coordinador de la Carrera de <i>Sistemas</i> ....., se procede a verificar por parte del mismo que se encuentra instalado y funcionando correctamente el siguiente software de acuerdo a las características solicitadas.							
N	N° Documento Solicitante	Docente Solicitante	Software Instalado	Versión	Lab. N°	Fecha Instalación	Observación
1	<i>03</i>	<i>Ing. Dennis Chicaiza</i>	<i>G'UI Turbo Assembler</i>	<i>3.0.1</i>	<i>1</i>	<i>05/03/18</i>	
2	<i>03</i>	<i>Ing. Dennis Chicaiza</i>	<i>Emu 8086</i>		<i>1</i>	<i>05/03/18</i>	
3	<i>09</i>	<i>PhD. Felix Fernández</i>	<i>Netbeans</i>	<i>8.2</i>	<i>1</i>	<i>05/03/18</i>	
4	<i>02</i>	<i>Ing. Patricio Gonzalez</i>	<i>Alice</i>	<i>33.12</i>	<i>1</i>	<i>05/03/18</i>	
5	<i>02</i>	<i>Ing. Patricio Gonzalez</i>	<i>Greenfoot</i>	<i>3.1.0</i>	<i>1</i>	<i>05/03/18</i>	
6	<i>02</i>	<i>Ing. Patricio Gonzalez</i>	<i>Netbeans</i>	<i>8.2</i>	<i>1</i>	<i>05/03/18</i>	



Para constancia firman:

  
 \_\_\_\_\_  
 Coordinador

\_\_\_\_\_  
 Administrador de Redes

Figura 28: Acta de conformidad - Laboratorio 1

Elaborado por: Administrador de redes FISEI

 <b>UNIVERSIDAD TÉCNICA DE AMBATO</b> FACULTAD DE... <i>Sistemas</i> ..... <b>ACTA DE CONFORMIDAD DE INSTALACIÓN DE SOFTWARE</b> 							
GENERALIDADES							
En la ciudad de Ambato, a los ..... días del mes de <i>Mayo</i> del 201 <i>8</i> , siendo las ....., en presencia del <i>Ing. Clay Aldas</i> ..., Coordinador de la Carrera de <i>Tecnologías de la I+D+i</i> , se procede a verificar por parte del mismo que se encuentra instalado y funcionando correctamente el siguiente software de acuerdo a las características solicitadas.							
N	N° Documento Solicitante	Docente Solicitante	Software Instalado	Versión	Lab. N°	Fecha Instalación	Observación
1	<i>/5</i>	<i>Ing. Oswaldo Paredes</i>	<i>Bibtex 4 word</i>		<i>1</i>	<i>05/03/18</i>	
2	<i>/5</i>	<i>Ing. Oswaldo Paredes</i>	<i>Jabref</i>	<i>3.8.2</i>	<i>1</i>	<i>05/03/18</i>	
3	<i>/5</i>	<i>Ing. Oswaldo Paredes</i>	<i>PseInt</i>		<i>1</i>	<i>05/03/18</i>	
4	<i>/5</i>	<i>Ing. Oswaldo Paredes</i>	<i>Netbeans</i>	<i>8.2</i>	<i>1</i>	<i>05/03/18</i>	
5	<i>/5</i>	<i>Ing. Oswaldo Paredes</i>	<i>Alice</i>	<i>3.31.2</i>	<i>1</i>	<i>05/03/18</i>	
6	<i>/5</i>	<i>Ing. Oswaldo Paredes</i>	<i>Greenfoot</i>	<i>3.1.0</i>	<i>1</i>	<i>05/03/18</i>	



Para constancia firman:

  
 \_\_\_\_\_  
 Coordinador

\_\_\_\_\_  
 Administrador de Redes

Figura 29: Acta de conformidad - Laboratorio 1

Elaborado por: Administrador de redes FISEI

 <b>UNIVERSIDAD TÉCNICA DE AMBATO</b> FACULTAD DE... <i>Sistemas</i> ..... <b>ACTA DE CONFORMIDAD DE INSTALACIÓN DE SOFTWARE</b> 							
GENERALIDADES							
En la ciudad de Ambato, a los ..... días del mes de <i>Mayo</i> del 201 <i>8</i> , siendo las ....., en presencia del <i>Ing. Clay Aldas</i> ..., Coordinador de la Carrera de <i>Tecnologías de la I+D+i</i> , se procede a verificar por parte del mismo que se encuentra instalado y funcionando correctamente el siguiente software de acuerdo a las características solicitadas.							
N	N° Documento Solicitante	Docente Solicitante	Software Instalado	Versión	Lab. N°	Fecha Instalación	Observación
1	<i>02</i>	<i>Ing. Patricia Gonzalez</i>	<i>Office</i>	<i>2016</i>	<i>1</i>	<i>05/03/18</i>	
2	<i>02</i>	<i>Ing. Patricia Gonzalez</i>	<i>Menuley Desktop Plus in Word</i>	<i>1.17.13</i>	<i>1</i>	<i>05/03/18</i>	
3	<i>02</i>	<i>Ing. Patricia Gonzalez</i>	<i>Lyx</i>	<i>2.2.3</i>	<i>1</i>	<i>05/03/18</i>	
4	<i>02</i>	<i>Ing. Patricia Gonzalez</i>	<i>Bibtex 4 word</i>		<i>1</i>	<i>05/03/18</i>	
5	<i>15</i>	<i>Ing. Oswaldo Paredes</i>	<i>Office</i>	<i>2016</i>	<i>1</i>	<i>05/03/18</i>	
6	<i>15</i>	<i>Ing. Oswaldo Paredes</i>	<i>Lyx</i>	<i>2.2.3</i>	<i>1</i>	<i>05/03/18</i>	


Para constancia firman:

  
 \_\_\_\_\_  
 Coordinador

\_\_\_\_\_  
 Administrador de Redes

Figura 30: Acta de conformidad - Laboratorio 1

Elaborado por: Administrador de redes FISEI

 <b>UNIVERSIDAD TÉCNICA DE AMBATO</b> FACULTAD DE <i>Sistemas</i> <b>ACTA DE CONFORMIDAD DE INSTALACIÓN DE SOFTWARE</b> 							
GENERALIDADES							
En la ciudad de Ambato, a los ..... días del mes de <i>Abril</i> del 201 <i>9</i> siendo las ....., en presencia del <i>Ing. Alex Alós</i> , Coordinador de la Carrera de <i>Sistemas</i> , se procede a verificar por parte del mismo que se encuentra instalado y funcionando correctamente el siguiente software de acuerdo a las características solicitadas.							
N	N° Documento Solicitante	Docente Solicitante	Software Instalado	Versión	Lab. N°	Fecha Instalación	Observación
1	<i>05</i>	<i>Ing. Alexis Sánchez</i>	<i>R</i>	<i>3.4.3</i>	<i>2</i>	<i>04/03/2019</i>	
2							
3							
4							
5							
6							

Para constancia firman:



  
 Coordinador

Administrador de Redes

Figura 31: Acta de conformidad - Laboratorio 2

Elaborado por: Administrador de redes FISEI

### Requerimientos de software

 <b>UNIVERSIDAD TÉCNICA DE AMBATO</b> FACULTAD DE ..... <b>REQUERIMIENTOS DE SOFTWARE</b> 						
DATOS GENERALES						
Docente Solicitante:		Período:			Fecha:	
N	Carrera	Nivel	Módulo	Software Requerido	Versión	Observación
1	<i>Electrónica</i>	<i>Sexto</i>	<i>Com Analógica</i>	<i>Mat Lab</i>	<i>la más actualizada</i>	
2	<i>Electrónica</i>	<i>Séptimo</i>	<i>Com Digital</i>	<i>Mat Lab</i>		
3	<i>Electrónica</i>	<i>Séptimo</i>	<i>Propagación</i>	<i>Radio Mobile</i>		
4						
5						
6						
7						
8						
9						
10						

Coordinador

  
 Docente  
*Ing MSc Carlos Alberto Serra Jiménez*

Figura 32: Requerimientos de software - Carrera Electrónica

Elaborado por: Ing. Carlos Serra

UNIVERSIDAD TÉCNICA DE AMBATO FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL REQUERIMIENTOS DE SOFTWARE						
						N. Documento: <u>20</u>
DATOS GENERALES						
Docente Solicitante: Ing. César A. Granizo L.			Período: Marzo 18 - Septiembre 18		Fecha: 23/02/18	
N	Carrera	Nivel	Módulo	Software Requerido	Versión	Observación
1	TELECOMUNICACIONES	SEGUNDO A Y B	PROGRAMACIÓN	DFD, Netbeans,		
2	ELECTRÓNICA	TERCERO A Y B	PROGRAMACIÓN III	Netbeans, Arduino, Proteus		
3	ELECTRÓNICA	TERCERO B	REDES DE COMPUTADORAS	Packet Tracer		
4						
5						
6						
7						
8						
9						
10						



 **Coordinador**
 **Docente**

Figura 33: Requerimientos de software - Telecomunicaciones Electrónica

Elaborado por: Ing. Cesar Granizo

UNIVERSIDAD TÉCNICA DE AMBATO FACULTAD DE <u>F.I.S.E.I.</u> REQUERIMIENTOS DE SOFTWARE						
						N. Documento: <u>19</u>
DATOS GENERALES						
Docente Solicitante: <u>Ing. Juan P. Pallo</u>			Período: <u>Marzo - Agosto 2018</u>		Fecha: <u>22/02/2018</u>	
N	Carrera	Nivel	Módulo	Software Requerido	Versión	Observación
1	<u>Electrónica y Comunicaciones</u>	<u>Noveno "E"</u>	<u>Comunicaciones Móviles</u>	<u>SCILAB</u>		
2	<u>//</u>	<u>Noveno "E"</u>	<u>Comunicaciones Móviles</u>	<u>Radio Works Satmaster</u>		
3						
4						
5						
6						
7						
8						
9						
10						



 **Coordinador**
 **Docente**

Figura 34: Requerimientos de software – Electrónica

Elaborado por: Ing. Juan Pallo



UNIVERSIDAD TÉCNICA DE AMBATO FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICOS INDUSTRIALES REQUERIMIENTOS DE SOFTWARE						
						N. Documento: 18
DATOS GENERALES						
Docente Solicitante:		Período:			Fecha:	
N	Carrera	Nivel	Módulo	Software Requerido	Versión	Observación
1	ELECTRÓNICA	SEXTO	INTERFAZ DE PC	LABVIEW	V13	
2	ELECTRÓNICA	SEXTO	SISTEMAS DE CONTROL	LABVIEW	V13	
3	ELECTRÓNICA	NOVENO	OPTATIVA 3	LABVIEW	V13	
4	ELECTRÓNICA	NOVENO	OPTATIVA 3	TRA PORTAL	V13	
5						
6						
7						
8						
9						
10						

Coordinador Docente  
Ing. Patricia Encalada

Figura 35: Requerimientos de software – Electrónica

Elaborado por: Ing. Patricio Encalada

UNIVERSIDAD TÉCNICA DE AMBATO FACULTAD DE SISTEMAS ELECTRÓNICOS E INDUSTRIALES FISEI REQUERIMIENTOS DE SOFTWARE						
						N. Documento: 17
DATOS GENERALES						
Docente Solicitante: Ing. Alexis Sánchez Mirón		Período: Marzo 2018/Agosto 2018			Fecha: 26/02/2018	
N	Carrera	Nivel	Módulo	Software Requerido	Versión	Observación
1	Electrónica	3 <sup>er</sup>	E y P	R	3.3.3 o 3.4.3	Software Libre
2		4 <sup>to</sup>	Met. Numéricas	Matlab u Octave	9.0 superior	Software Proprietario
3				Octave	3.8.1	Software Libre
4						
5						
6						
7						
8						
9						
10						

Coordinador Docente

Figura 36: Requerimientos de software - Electrónica

Elaborado por: Ing. Alexis Sánchez

Para poder realizar un monitoreo eficiente, los módulos que permitirán capturar datos de escritorio y web se basan en el proceso descrito en figura 23:

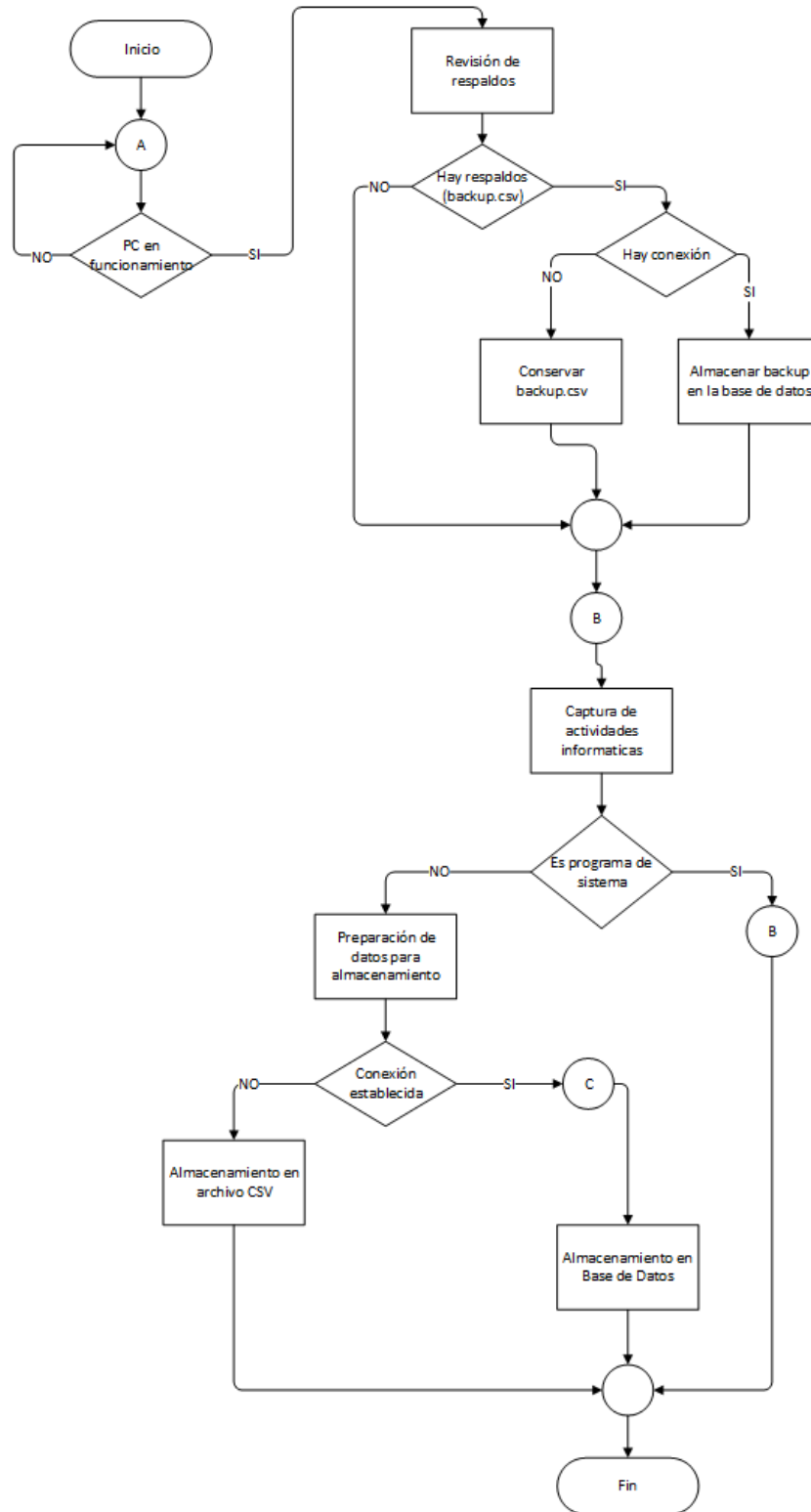


Figura 37: Diagrama de flujo - Módulos de Monitoreo

Elaborado por: Paul Tandazo

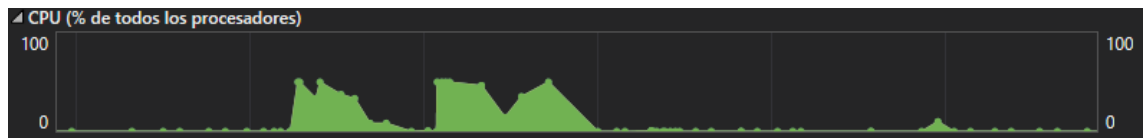
## Pruebas de almacenamiento en CSV:



En 3 momentos el uso de la CPU llega al 50 %: 3,143 s; 3,358 s; 3,897 s. Además, el uso llega a 42% en los 3,64 s.

Evento	Hora	Duraci...	Subp...
◆ Gesto: Clic en "History" (botón)	3,06s		[1280] <S
Punto de interrupción: button1_Click, Form1.cs línea 803	3,06s	3.066 ms	[1280] <S
Paso: button1_Click, Form1.cs línea 804	3,06s	3 ms	[1280] <S
Paso: button1_Click, Form1.cs línea 805	3,07s	11 ms	[1280] <S
Paso: button1_Click, Form1.cs línea 806	3,10s	27 ms	[1280] <S
◆ Excepción: Se detectó una excepción: "System.FormatException" en mscorlib.dll ("La cadena de entrada no tie...	3,14s		[1280] <S
◆ Excepción: Se detectó una excepción: "System.FormatException" en mscorlib.dll ("La cadena de entrada no tie...	3,14s		[1280] <S
◆ Resultados del programa: Native library pre-loader is trying to load native SQLite library "C:\Users\Andres...	3,29s		
➔ Paso: button1_Click, Form1.cs línea 868	3,86s	757 ms	[1280] <S
Un usuario llevó a cabo un paso en el depurador.			
<a href="#">Activar depuración histórica</a>			

## Pruebas de almacenamiento en SQLite:



En 4 momentos el uso de la CPU llega a 50 %: 5,286 s; 5,418 s; 6,081 s; 6,671 s.

Evento	Hora	Duraci...	Subp...
◆ Resultados del programa: El subprocesso 0xca2c se ha cerrado con el código 0 (0x0).	3,49s		
◆ Gesto: Clic en "History" (botón)	5,18s		[50416] <
Punto de interrupción: button1_Click, Form1.cs línea 817	5,18s	5.182 ms	[50416] <
Paso: button1_Click, Form1.cs línea 818	5,26s	82 ms	[50416] <
◆ Resultados del programa: Native library pre-loader is trying to load native SQLite library "C:\Users\Andres...	5,42s		
Paso: button1_Click, Form1.cs línea 820	6,02s	766 ms	[50416] <
Paso: button1_Click, Form1.cs línea 825	6,07s	52 ms	[50416] <
◆ Excepción: Se detectó una excepción: "System.FormatException" en mscorlib.dll ("La cadena de entrada no tie...	6,09s		[50416] <
◆ Excepción: Se detectó una excepción: "System.FormatException" en mscorlib.dll ("La cadena de entrada no tie...	6,09s		[50416] <
➔ Paso: button1_Click, Form1.cs línea 868	7,31s	1.234 ms	[50416] <
Un usuario llevó a cabo un paso en el depurador.			
<a href="#">Activar depuración histórica</a>			



## Manual de Usuario

### Requerimientos para instalación de módulos de monitoreo:

Es necesario que el equipo cuente con:

- Sistema Operativo Windows 7 Service Pack 1 o superior.
- Microsoft .NET Framework 4.5 (Windows 8 o superiores ya incluyen este framework).
- Conexión al servidor (172.123.21.50), software funcional en la facultad.

Ejecutamos el instalador “setup.exe” que está disponible en la carpeta MonitoreoUTACloudIoT/Debug:

Nombre	Fecha de modifica...	Tipo	Tamaño
 MonitoreoUTACloudIoT.msi	30/11/2018 19:19	Paquete de Windo...	734 KB
 setup.exe	30/11/2018 19:19	Aplicación	531 KB

Luego únicamente bastara con seguir el Asistente para la instalación de programas de Windows:

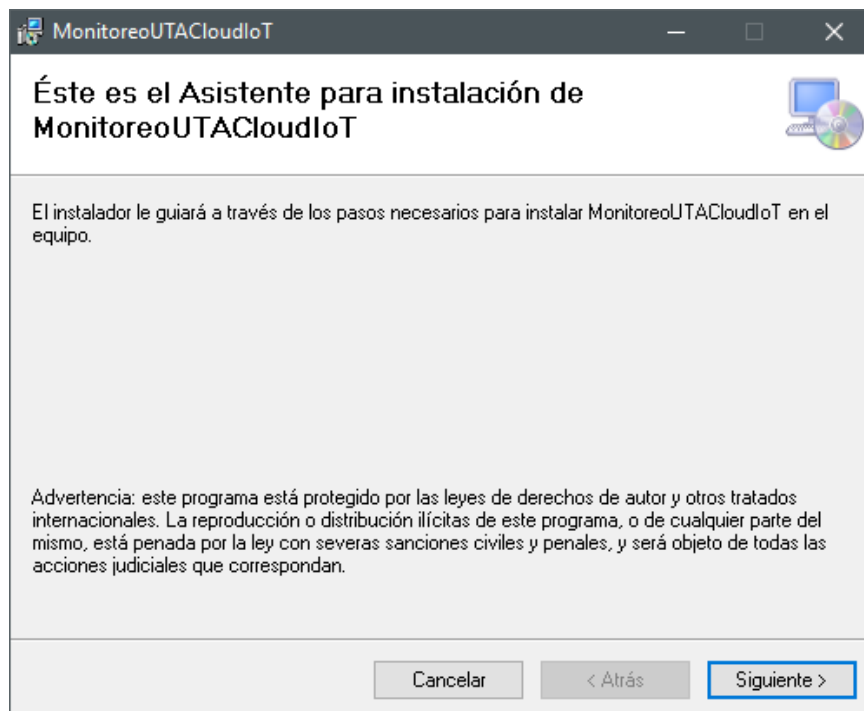


Figura 38: Asistente de Instalación 1  
Elaborado por: Paul Tandazo

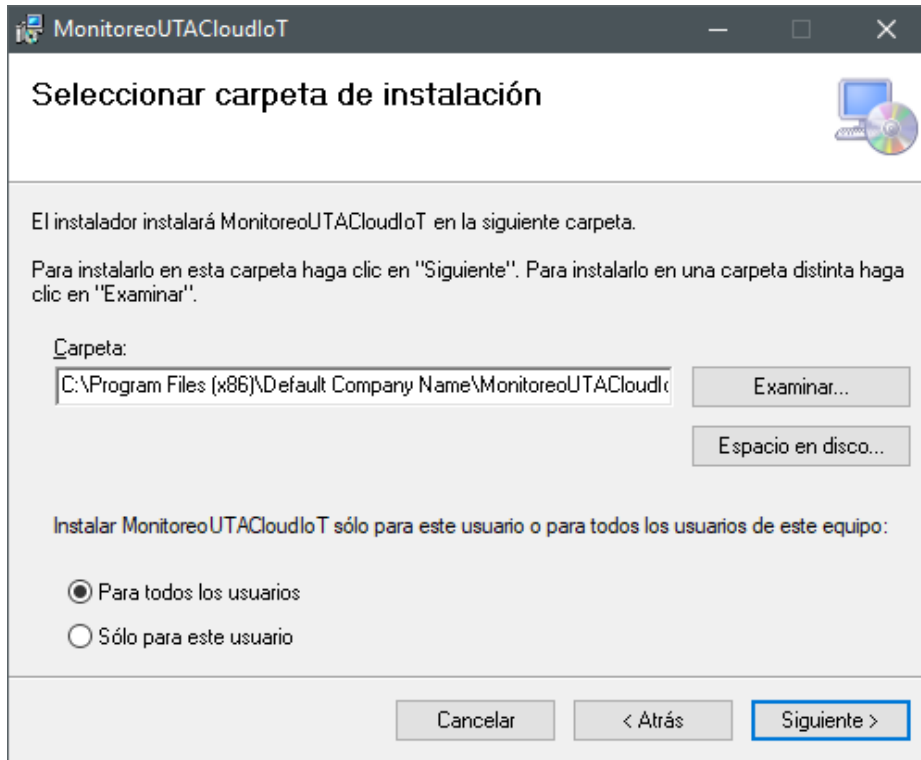


Figura 39: Asistente de Instalación 2  
Elaborado por: Paul Tandazo

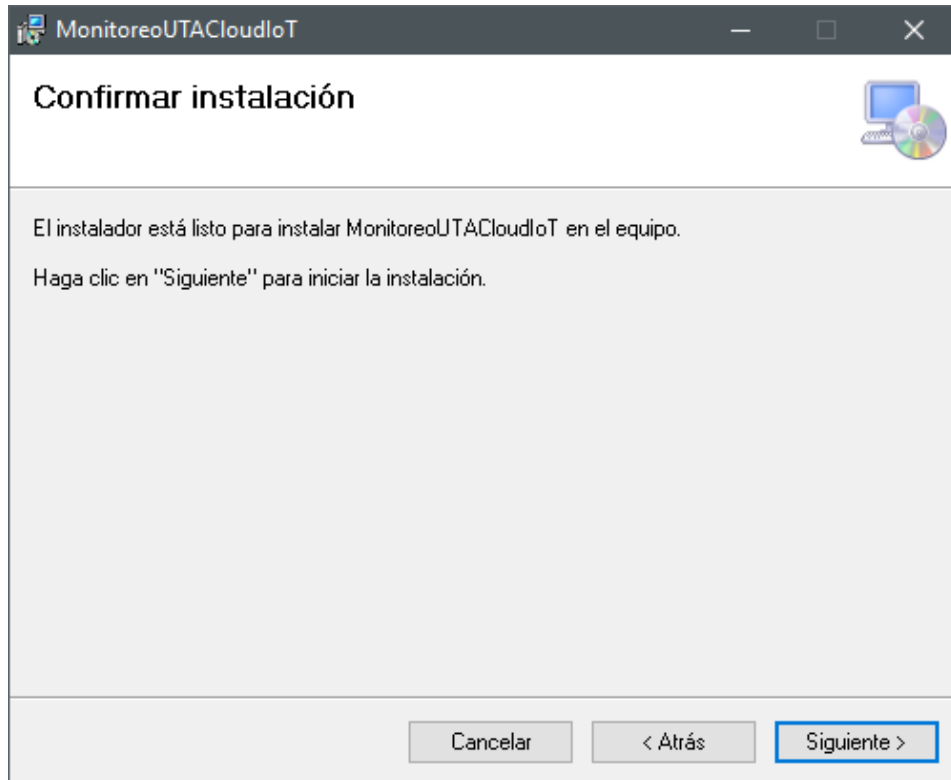


Figura 40: Asistente de Instalación 3  
Elaborado por: Paul Tandazo

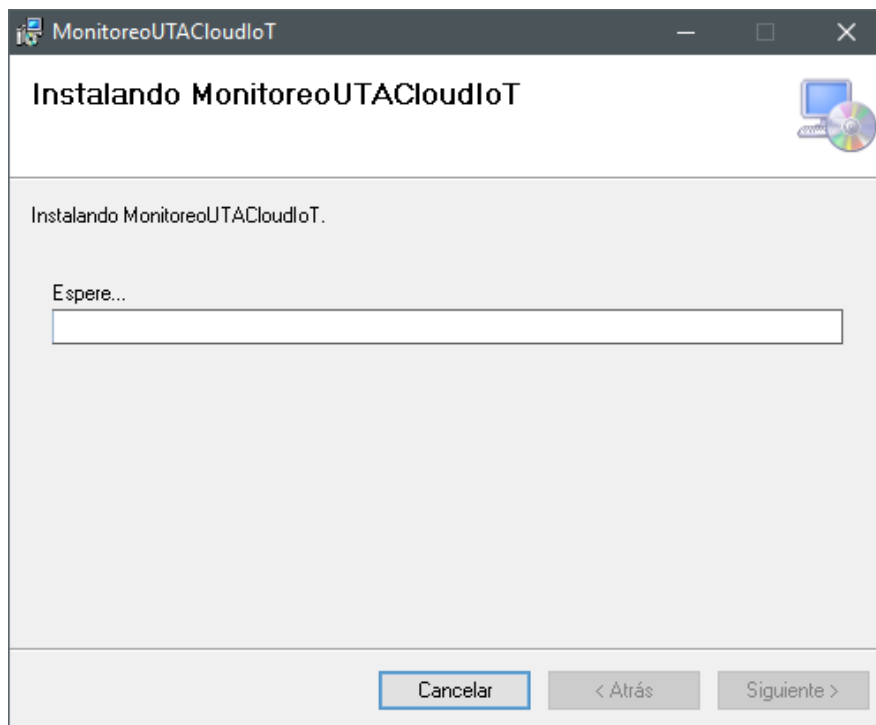


Figura 41: Asistente de Instalación 4  
Elaborado por: Paul Tandazo

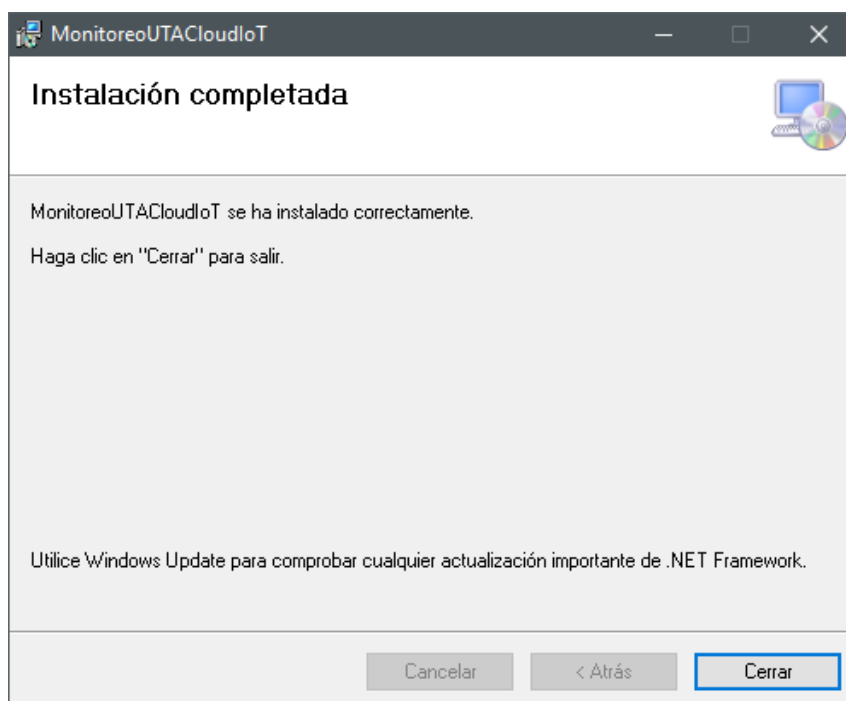


Figura 42: Asistente de Instalación 5  
Elaborado por: Paul Tandazo

Debido a que los módulos son programas que se ejecutaran en segundo plano, para comprobar que la instalación fue exitosa únicamente bastara con revisar el Administrador de Tareas y comprobar que el módulo instalado se encuentre la lista de Procesos en Segundo Plano:

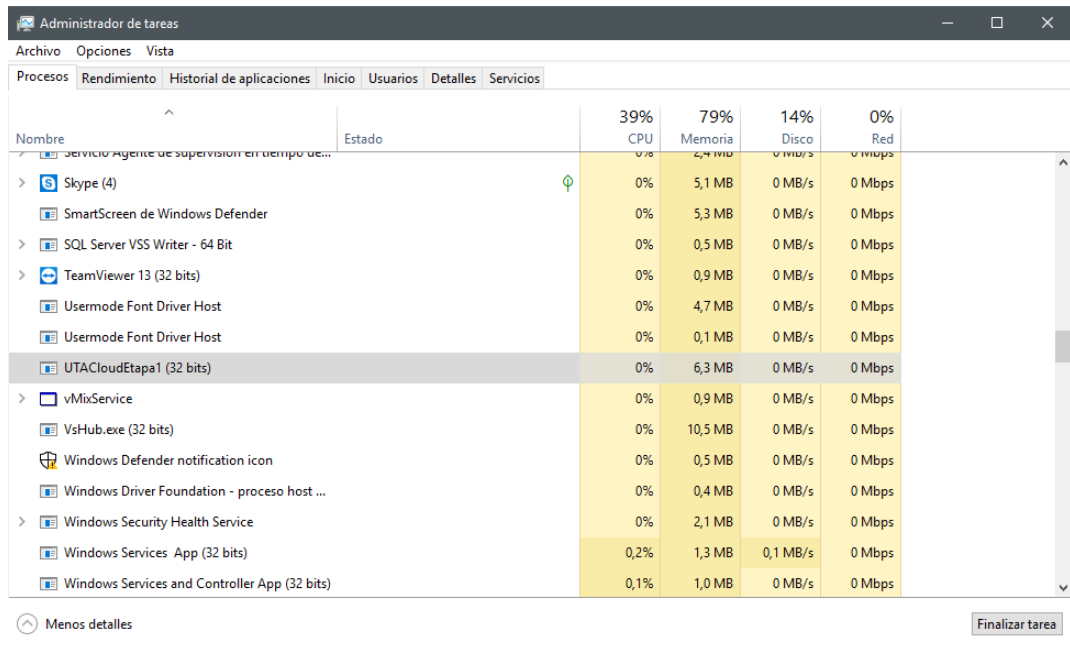


Figura 43: Lista de procesos en segundo plano  
Elaborado por: Paul Tandazo

## Acceso a reportes en la Plataforma

Uso del Software

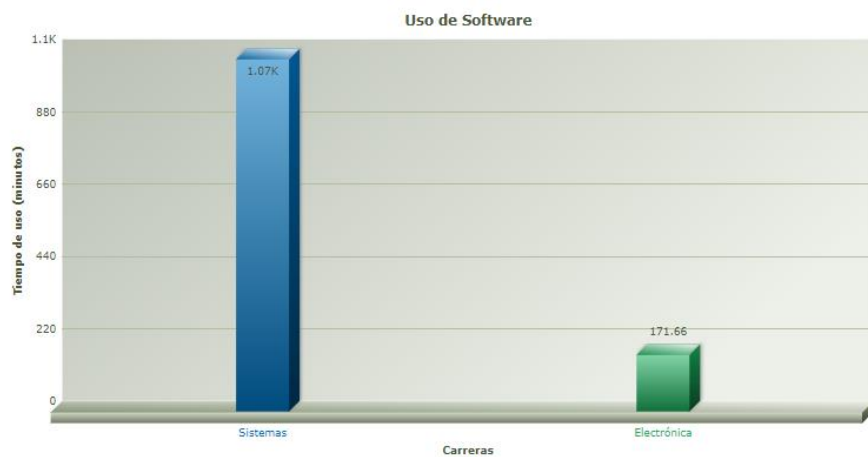


Figura 44: Uso de software por carrera  
Elaborado por: Paul Tandazo

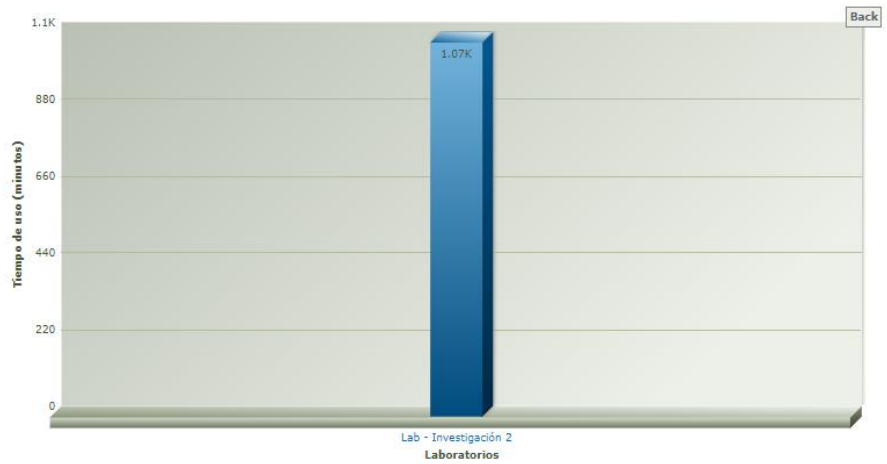


Figura 45: Uso de software por laboratorio  
Elaborado por: Paul Tandazo

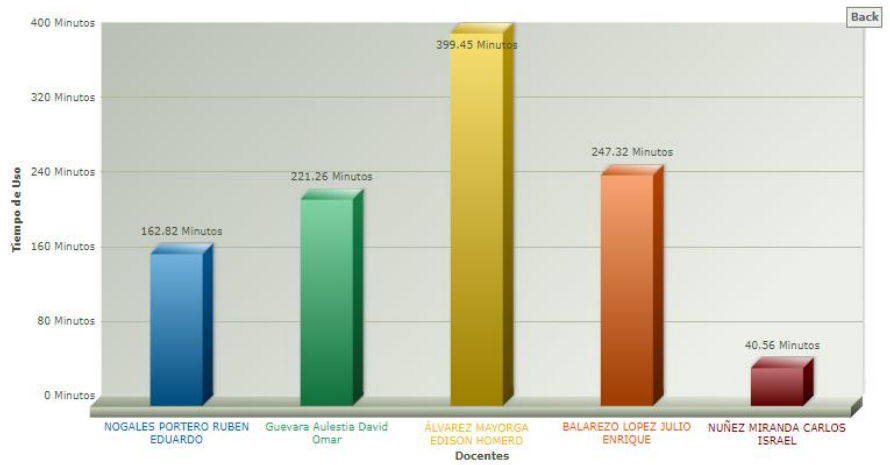


Figura 46: Uso de software por docente  
Elaborado por: Paul Tandazo



## Uso del Software

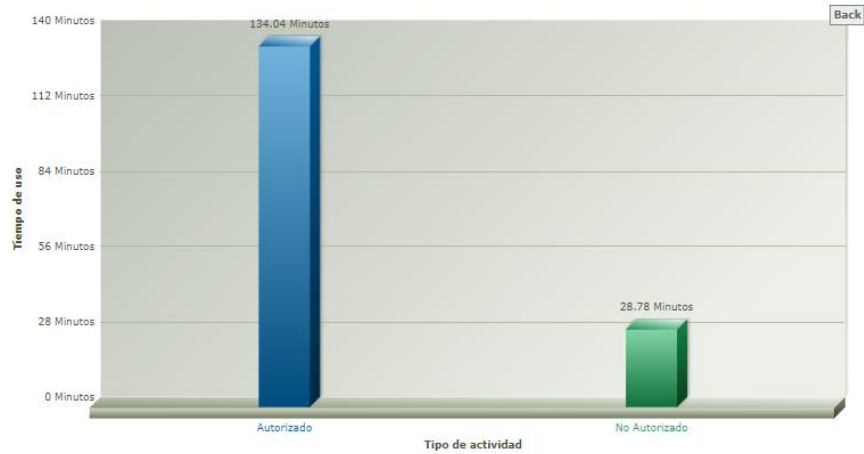


Figura 47: Uso de software autorizado y no autorizado  
Elaborado por: Paul Tandazo

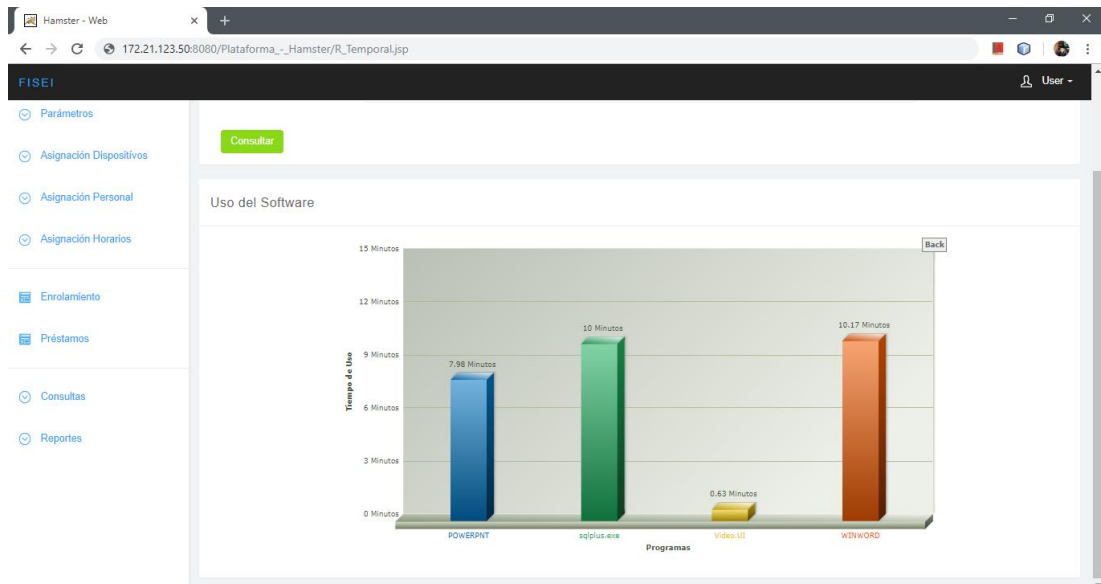


Figura 48: Tiempo de uso utilizado por programa no autorizado  
Elaborado por: Paul Tandazo

### Análisis de datos capturados durante 13 horas clase:

Mediante las pruebas de funcionamiento del sistema de monitoreo se capturó actividades informáticas entre los días 4 y 5 de diciembre del 2018 en el laboratorio 5 de la FISEI, durante ese lapso se pudo determinar que la aplicación opera sin inconvenientes y no reporto problemas de ejecución.

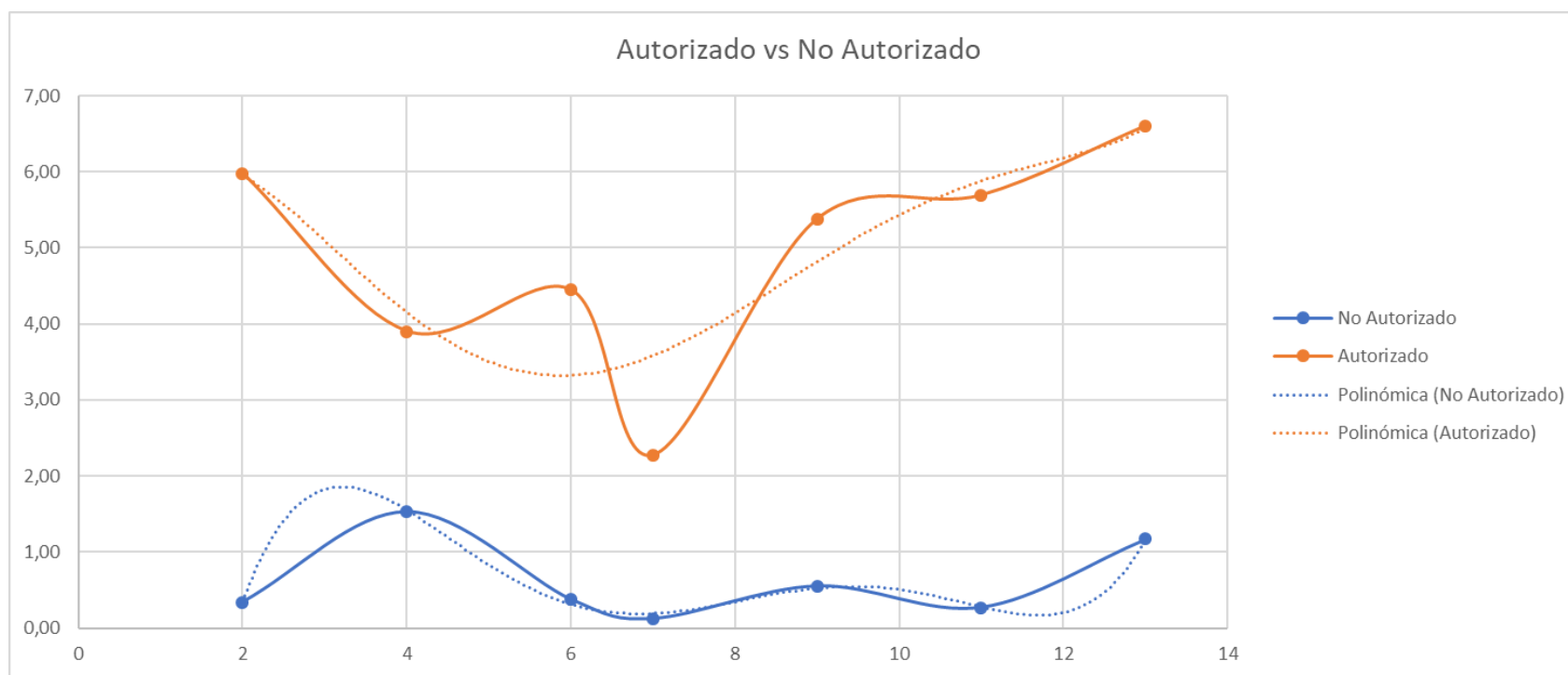


Figura 49: Análisis de datos  
Elaborado por Paul Tandazo

