

*Universidad de las Ciencias Informáticas*

*Facultad 3*

*Centro de Informatización de Entidades*



*Título: Sistema de reportes basado en indicadores para contribuir a la toma de decisiones en la actividad productiva de la UCI.*

*Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas.*

*Autora: Irina Elsa Mayedo Tamayo.*

*Tutores: Ing. Virtudes Milagro Figueredo Lara.*

*Ing. Feili Ibarra Monteagudo.*

*Junio 29, 2017*

*"Año 58 de la Revolución"*

# *Declaración de autoría*

---

Declaro que soy el único autor de este trabajo y autorizo a la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmo la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año\_\_\_\_\_.

Irina Elsa Mayedo Tamayo

**Autor**

---

Virtudes Milagros Figueredo Lara

**Tutor**

---

Yeili Ibarra Monteagudo

**Tutor**

---

# Agradecimientos

---

*A mi madre porque eres la que posibilitó este momento de mi vida, porque solo nosotras sabemos todo lo que nos ha costado, pero a ti te ha costado más, te ha costado la vida entera dedicándote a tus hijos, yo soy la última en llegar y lo hice sólo porque tú estabas conmigo. Gracias mamá.*

*Le agradezco a mis hermanos, Eblis y Adanay, por ayudar a mami y papi a educarme en los mejores principios éticos, morales y revolucionarios. A mi hermana Ismaray, por todo lo que has hecho por mí, que no tengo palabras para expresar todo mi agradecimiento.*

*A mi padre porque la vocación al estudio e inculcarme a terminar lo que comencé, a ser una buena persona y honrada, me lo enseñaste tú, eso te lo voy a agradecer toda la vida, porque esa forma de ser es lo que me abrió muchas puertas cuando otras se cerraban. Gracias*

*A mis amigos de toda la vida y a mis compañeros de la universidad, con los que comencé en Manzanillo, y con los que terminé aquí en la Habana, gracias a todos por los mejores años de mi vida, por las maldades, por las historias que vivimos juntos que son mis mejores recuerdos y llevaré conmigo siempre.*

*A Ivys y su familia, por la amistad a pesar de la distancia y los años; Yeili y Ricardo por lo especiales e incomparables que son; Arlene, Yanser, Rafael, Camilo, Julio, Alejandro, Claudia, Doris, Félix, por acogerme en su grupo cuando llegué de Granma, gracias a ustedes, y al resto de mi grupo. A Sucef, Adrianna, Maite, Lizandra, Yuliet, Rosalia, y a todas las niñas y niños de mi querida FRG Dayana Díaz, y Jenniffer.*

*A la familia de Noel: a Yayi, Luis, Odalis, Armando, Nailín, Pedrito, Tere, y a los de Pílon, a todos por acogerme, apoyarme y darme la oportunidad de ser parte de su familia, muchas gracias. Agradecer a mi persona muy especial y que le ha tocado vivir conmigo la peor parte de este proceso, en el cual has tenido que soportar mi estrés, los cambios de humor, las lágrimas, el dolor, y aun así te quedaste a mi lado, por eso te agradezco infinitamente y te amo Noel.*

# Dedicatoria

---

*A mi madre por ser inigualable, incansable, por el esfuerzo infinito, por apoyarme todo el camino hasta aquí. Por ser mi madre, mi amiga y mi motor para llegar a ser lo que soy hoy, sin ti nada podría haber sido posible, porque cuando caí me hiciste levantarme porque incluso cuando yo te fallé, madre tú a mí, nunca.*

*A mi padre por ser quien fijara la meta y el camino a seguir para ser lo que hoy soy. A mis sobrinos que son mi vida para que vean que los caminos hacia nuestros sueños están llenos de irregularidades, de tropiezos, de caídas, pero también en ellos encuentras siempre la forma de seguir adelante, de levantarse tantas veces como se ha caído y llegar hasta el final; lograr sus sueños mis amores, solo depende de ustedes.*

*A mi hermana por el carácter, por el impulso, por educarme y creer ser mi mamá y yo tu ensayo de hija, por ser mi guía, mi espejo del alma y por ser quién puso la altura a la cual debía llegar, aunque para ser como tú aún me falta un poco más.*

*A mi hermano por su nobleza y humildad, y por el amor que yo sé que me tienes detrás de toda esa seriedad. A toda mi familia y a mis amigos. A Noel por quererme y hacerme feliz.*

*A Fidel y la Revolución por permitirme estudiar lo que quería y no lo que podría.*

# *Resumen*

---

Los sistemas de reportes se han convertido en una importante herramienta para el proceso de toma de decisiones, analizando los datos de una forma rápida y automatizada. Esto permite establecer comparaciones entre la documentación histórica y actual, ofreciendo una visión general de la situación de la empresa.

La presente investigación propone el desarrollo de un sistema de reportes basado en los indicadores de gestión de proyecto que utiliza la herramienta Gespro, con el objetivo de favorecer las decisiones tomadas sobre la planificación, asignación de recursos y aprovechamiento del fondo de tiempo en la actividad productiva de la Universidad de las Ciencias Informáticas.

Para realizar la validación del sistema se utilizan técnicas de caja blanca, caja negra, pruebas internas y pruebas de aceptación con el cliente. Y para validar la investigación, fue aplicado el Cuasiexperimento preprueba/posprueba, lo que evidenció que la propuesta de solución ofrece ventajas que la herramienta Gespro y el Servidor de Reportes SDR no brindan, y que facilitará al usuario final realizar el análisis e interpretación de los datos de forma interactiva para tomar decisiones desde etapas tempranas del desarrollo de los proyectos.

## **PALABRAS CLAVE**

Comparaciones, decisiones, indicadores, proyectos, reportes, sistema.

# Tabla de contenido

---

<b>INTRODUCCIÓN</b> .....	<b>1</b>
<b>CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA</b> .....	<b>7</b>
<b>1.1. INTRODUCCIÓN</b> .....	<b>7</b>
<b>1.2. CONCEPTOS FUNDAMENTALES DE LA INVESTIGACIÓN</b> .....	<b>7</b>
Proyecto .....	7
Factores críticos de éxito de los proyectos .....	7
Indicadores.....	8
<b>1.3. INDICADORES PARA LA PROPUESTA DE SOLUCIÓN</b> .....	<b>9</b>
<b>1.4. SISTEMAS DE REPORTES</b> .....	<b>10</b>
Servidor de Reportes (SDR) .....	11
Herramienta de gestión de proyectos Gespro.....	11
<b>1.5. ANÁLISIS COMPARATIVO DE LOS SISTEMAS DE REPORTES</b> .....	<b>12</b>
<b>1.6. TOMA DE DECISIONES</b> .....	<b>13</b>
Modelos y métodos para la toma de decisiones .....	14
<b>1.7. ESTADO DEL PROCESO DE TOMA DE DECISIONES EN LA UCI</b> .....	<b>15</b>
<b>1.8. COMPONENTE HUMANÍSTICO</b> .....	<b>17</b>
<b>1.9. METODOLOGÍAS PARA EL DESARROLLO DE SOFTWARE</b> .....	<b>19</b>
AUP-UCI.....	19
Disciplinas de AUP-UCI.....	20
Escenarios de la metodología.....	20
<b>1.10. HERRAMIENTAS Y LENGUAJES PARA EL DESARROLLO DE SOFTWARE</b> .....	<b>20</b>
Lenguaje de Modelado UML 2.0.....	20
Lenguajes de programación utilizados.....	21
Marco de trabajo para desarrollar la propuesta de solución.....	22
Gestión y administración de bases de datos.....	23
Entorno de Desarrollo Integrado (IDE).....	23
Servidor de Aplicaciones .....	24
<b>1.11. CONCLUSIONES PARCIALES</b> .....	<b>24</b>
<b>CAPÍTULO 2: DESARROLLO DE LA SOLUCIÓN</b> .....	<b>25</b>

# Tabla de contenido

---

<b>2.1. INTRODUCCIÓN .....</b>	<b>25</b>
<b>2.2. NECESIDADES DEL NEGOCIO .....</b>	<b>25</b>
<b>2.3. REQUISITOS .....</b>	<b>25</b>
Técnicas para la obtención de requisitos de software .....	25
Especificación de los requisitos .....	26
Requisitos Funcionales .....	26
Requisitos No Funcionales .....	28
<b>2.4. VALIDACIÓN DE LOS REQUISITOS .....</b>	<b>30</b>
Técnicas de validación de requisitos .....	30
<b>2.5. DISEÑO DEL SISTEMA .....</b>	<b>30</b>
Arquitectura del sistema .....	31
Diagrama de clases del diseño .....	32
Patrones de diseño de software .....	34
Modelo de datos .....	36
Métricas para validación del diseño .....	36
<b>2.6. IMPLEMENTACIÓN DEL SISTEMA .....</b>	<b>43</b>
Diagrama de componentes .....	43
Diagrama de despliegue .....	44
Estándares de codificación .....	44
Descripción de la implementación .....	46
<b>2.7. CONCLUSIONES PARCIALES .....</b>	<b>48</b>
<b>CAPÍTULO 3: VALIDACIÓN DE LA SOLUCIÓN .....</b>	<b>49</b>
<b>3.1. INTRODUCCIÓN .....</b>	<b>49</b>
<b>3.2. VALIDACIÓN DEL SOFTWARE .....</b>	<b>49</b>
Pruebas de caja blanca .....	49
Pruebas de caja negra .....	52
Descripción del caso de prueba <i>Graficar</i> .....	53
Pruebas internas .....	54
Pruebas de aceptación .....	54

# Tabla de contenido

---

<b>3.3. VALIDACIÓN CIENTÍFICA APLICANDO EL CUASIEXPERIMENTO DE TIPO PREPRUEBA/POSPRUEBA</b>	
<b>55</b>	
Resultados obtenidos a partir de aplicar el sistema SRBin al grupo de control.....	57
<b>3.4. CONCLUSIONES PARCIALES .....</b>	<b>59</b>
<b>CONCLUSIONES GENERALES .....</b>	<b>60</b>
<b>RECOMENDACIONES .....</b>	<b>61</b>
<b>REFERENCIAS BIBLIOGRÁFICAS .....</b>	<b>62</b>
<b>ANEXOS.....</b>	<b>69</b>
<b>ANEXO 1: CUESTIONARIO: “PROCESO DE TOMA DE DECISIONES EN LA UCI Y CREACIÓN DE UNA HERRAMIENTA PARA FAVORECER LA TOMA DE DECISIONES.” .....</b>	<b>69</b>
<b>ANEXO 2: RESULTADO DE LA ENCUESTA EN EL CENTRO CEIGE. ....</b>	<b>71</b>
<b>ANEXO 3: RESULTADO DE LA ENCUESTA EN EL CENTRO CEGEL. ....</b>	<b>72</b>
<b>ANEXO 4: RESULTADO DE LA ENCUESTA EN EL CENTRO DATEC. ....</b>	<b>73</b>
<b>ANEXO 5: RESULTADO DE LA ENCUESTA EN EL CENTRO VERTEX. ....</b>	<b>74</b>
<b>ANEXO 6: RESULTADO DE LA ENCUESTA PREGUNTA 4 SOBRE HERRAMIENTAS. ....</b>	<b>75</b>
<b>ANEXO 7 HISTORIA DE USUARIO AUTENTICAR USUARIO. ....</b>	<b>75</b>
<b>ANEXO 8: HISTORIA DE USUARIO GESTIONAR ROLES DE LOS USUARIOS DEL SISTEMA. ....</b>	<b>76</b>
<b>ANEXO 9: HISTORIA DE USUARIO DE LA FUNCIONALIDAD LISTAR LOS USUARIOS REGISTRADOS EN EL SISTEMA.....</b>	<b>77</b>
<b>ANEXO 10: HISTORIA DE USUARIO VISUALIZAR LOS DATOS GENERALES DEL PROYECTO.....</b>	<b>77</b>
<b>ANEXO 11: HISTORIA DE USUARIO GUARDAR BÚSQUEDAS.....</b>	<b>78</b>
<b>ANEXO 12: DIAGRAMA DE CLASES DEL GESTIONAR EVALUACIONES DEL PROYECTO.....</b>	<b>79</b>
<b>ANEXO 13: DISEÑO DE CASO DE PRUEBA PARA EL CAMINO 2. ....</b>	<b>79</b>
<b>ANEXO 14: DISEÑO DE CASO DE PRUEBA PARA EL CAMINO 3. ....</b>	<b>80</b>
<b>ANEXO 15: DISEÑO DE CASO DE PRUEBA PARA EL CAMINO 4. ....</b>	<b>81</b>
<b>ANEXO 16: DISEÑO DE CASO DE PRUEBA PARA EL CAMINO 5. ....</b>	<b>81</b>
<b>ANEXO 17: DISEÑO DE CASO DE PRUEBA PARA EL CAMINO 6. .... ¡ERROR! MARCADOR NO DEFINIDO.</b>	
<b>ANEXO 18: DESCRIPCIÓN DE CASO DE PRUEBA GUARDAR BÚSQUEDA.....</b>	<b>82</b>
<b>ANEXO 19: DESCRIPCIÓN DE CASO DE PRUEBA EXPORTAR.....</b>	<b>82</b>



# *Tabla de contenido*

---

<b>ANEXO 20: DESCRIPCIÓN DE CASO DE PRUEBA COMPARAR PROYECTOS.....</b>	<b>83</b>
<b>ANEXO 21: DESCRIPCIÓN DE CASO DE PRUEBA GRAFICAR.....</b>	<b>83</b>
<b>ANEXO 22: ACTA DE LIBERACIÓN DE PRUEBAS INTERNAS .....</b>	<b>85</b>
<b>ANEXO 23: ACTA DE ACEPTACIÓN CON EL CLIENTE.....</b>	<b>90</b>

# Índice de Figuras

---

Figura 1: Resultados generales sobre el proceso de toma de decisiones en la UCI.....	16
Figura 2: Historia de usuario de la funcionalidad Exportar reportes.....	28
Figura 3: Patrón MVC en el sistema SRBin.....	31
Figura 4: Aplicación del patrón MVC en el sistema SRBin. Fuente: elaboración propia.....	32
Figura 5: Diagrama de clases de diseño de la funcionalidad Gestionar los proyectos.....	33
Figura 6: Patrón GRASP aplicado a la clase de diseño.....	35
Figura 7: Modelo de datos de la solución propuesta. Fuente: elaboración propia.....	36
Figura 8: Resultado del atributo Responsabilidad. ....	38
Figura 9: Resultado del atributo Complejidad de implementación. ....	39
Figura 10: Resultado del atributo Reutilización. ....	39
Figura 11: Porciento de los resultados obtenidos.....	41
Figura 12: Resultado del atributo Acoplamiento. ....	41
Figura 13: Resultado del atributo Complejidad de mantenimiento.....	41
Figura 14: Resultado del atributo Cantidad de pruebas.....	42
Figura 15: Resultado del atributo Reutilización. ....	42
Figura 16: Diagrama de componentes. ....	43
Figura 17: Diagrama de Despliegue del sistema SRBin.....	44
Figura 18: Ejemplo del uso de los estándares de codificación. Fuente: elaboración propia.....	45
Figura 19. Interfaz del sistema SRBin sección Mis Proyectos. ....	46
Figura 20: Fragmento de código del método reporteEvaluaciónProyecto(). ....	48
Figura 21: Representa el grafo de flujo obtenido del código representado anteriormente. ....	50
Figura 22: Resultados de la prueba interna por iteraciones realizadas.....	54
Figura 23: Resultados de la prueba de aceptación.....	55

# *Índice de Figuras*

---

Figura 24: Observación 1 sobre el grupo experimental. ....	56
Figura 25: Observación 2 sobre el grupo experimental: Gráfica de comparación entre proyectos..	57
Figura 26: Observación 2 sobre el grupo experimental: Gráfica de diferencias entre los indicadores. .....	58
Figura 27: Indicadores del Gespro luego de aplicado el estímulo.....	59

# *Índice de Tablas*

---

Tabla 1. Comparación entre los sistemas SDR y Gespro.....	13
Tabla 2: Porcentajes obtenidos correspondientes a la Figura1. ....	16
Tabla 3:Tamaño operacional de clase (TOC).....	37
Tabla 4: Rango de valores para evaluar la técnica de atributos de calidad de la métrica TOC.....	37
Tabla 5: Tamaño de las clases del sistema.....	38
Tabla 6 Atributos de calidad evaluados por la métrica RC. ....	40
Tabla 7: Criterios de evaluación para la métrica RC.....	40
Tabla 8: Diseño de caso de prueba para el camino 1.....	52
Tabla 9: Diseño de caso de prueba de la funcionalidad Graficar.....	53
Tabla 10: Casos Válidos y No Válidos del EC 1.1 Graficar.....	54

# *Introducción*

---

## **INTRODUCCIÓN**

El tránsito de la sociedad industrial a la sociedad de la información y el conocimiento, constituye una de las principales transformaciones que ocurren en el mundo contemporáneo de hoy, sitúa en nuevas metas a las ciencias modernas y de forma particular a las de la información (1). Las empresas y organizaciones mediante el uso de la información y las tecnologías, ambicionan el poder resolver los problemas y satisfacer las necesidades del ámbito empresarial, organizacional y de la sociedad que se encuentran sumergidas en constantes cambios aparejados a los avances tecnológicos e informacionales.

Para garantizar la viabilidad y gestión de los procesos empresariales en la solución de sus problemas, es necesario planificar, monitorear y controlar las actividades y recursos con la finalidad de orientarlos hacia la consecución de los mismos, lo que a su vez deriva la necesidad de adoptar herramientas y metodologías que permitan a las organizaciones mejorar su sistema de gestión (2).

Dentro de las herramientas informáticas guiadas por metodologías de desarrollo de software se encuentran los Sistemas de Reportes, estos se basan en las estadísticas generadas a partir de los indicadores específicos definidos para cada organización. Estos han sido utilizados en gran medida para el apoyo a la toma de decisiones en las empresas y con ello la obtención de mejores resultados en la ejecución de los procesos.

El análisis basado en la toma de decisiones puede ser utilizado en todas las fases del ciclo de vida de un proyecto, con el objetivo de establecer criterios que garanticen la calidad de los productos o servicios, puesto que de ello depende la satisfacción del cliente para el cual se obtiene el producto final.

Cuba avanza en la informatización segura de la sociedad, consciente de que la era de Internet y las nuevas tecnologías debe ser un espacio de aprendizaje, desarrollo, inclusivo y también seguro, unido a la utilización de las Tecnologías de la Informática y las Comunicaciones (TIC) (3).

# *Introducción*

---

La Universidad de la Ciencias Informáticas (UCI) posee un programa de estudios, que tiene como objetivo principal: formar ingenieros en ciencias informáticas vinculados al desarrollo de software, con el fin de crear sistemas para la informatización de la sociedad cubana, así como la incursión de esta rama en el ámbito internacional (4).

Desde los primeros años de la UCI se han adquirido variadas experiencias y mejoras continuas en cuanto a su enfoque y perspectivas para la producción de software, lo que le ha permitido en el transcurso de los años la obtención de varios galardones, uno de estos fue alcanzado el pasado 7 de octubre del 2015, con la certificación de la actividad productiva de la UCI en el Nivel 2 de CMMI-DEV v1.3<sup>1</sup> (5), otorgado por la prestigiosa empresa consultora SIE Center<sup>2</sup>, la cual posee entre sus principales actividades las certificaciones en modelos de calidad para la industria de Tecnologías de la Información (6).

Los proyectos de desarrollados en la UCI utilizan para la planificación, monitoreo y control de sus procesos la herramienta de gestión de proyectos Gespro. Realiza además la asignación de recursos materiales y humanos, chequeos de calidad, costos y tiempo, así como el análisis de los datos y la gestión de los riesgos; todas estas acciones son ejecutadas a través del cálculo automático de indicadores para cada proyecto, permitiendo analizar la ejecución de los hitos principales en el desarrollo y obtención del producto final (7).

Gespro muestra un reporte diario y semanal del estado del proyecto por indicadores, pero no permite visualizar los datos históricos de estos reportes, además no permite exportar el estado de los indicadores en ningún formato, teniendo como consecuencia que no puedan ser utilizados en el análisis de los resultados en chequeos de proyectos internos o gerencial, ni puedan guardarse bajo gestión documental.

Actualmente al cerrar los proyectos en la herramienta Gespro se trasladan todos sus datos hacia una base de datos (BD) que realiza la función de histórico, nombrándose *Proyectos terminados*,

---

<sup>1</sup> CMMI (Capability Maturity Model Integration): Modelo Integrado de Capacidad y Madurez

<sup>2</sup> SIE Center: Centro de excelencia de la industria del software

# *Introducción*

---

esta operación se realiza cada vez que culmina un proyecto. Éste histórico tiene como desventaja el no poseer una interfaz gráfica que permita analizar los reportes de los proyectos finalizados. Cuando se necesita obtener un análisis de estos proyectos se realiza únicamente a través de consultas directas a la base de datos (8).

Los proyectos terminados son una importante fuente de información para la Universidad, a los que no se puede acceder a través de una interfaz, no pudiéndose realizar un grupo de análisis estadísticos necesarios para el proceso productivo y la toma de decisiones. La responsabilidad de realizar las consultas a la BD y elaborar un informe para la gerencia, recae en una sola persona del Departamento de Gestión Producción (DGP), existiendo como riesgos que tanto la veracidad de los datos tratados manualmente no sea correcta, así como el cumplimiento de la tarea no se vea afectada por la fluctuación del personal continua que hoy sufre la Universidad.

La tendencia en la Universidad ha demostrado que en el desarrollado de los proyectos existen varios casos que culminan correctamente, pero que no cumplen en tiempo con los hitos planificados desde el inicio, lo que reporta valores negativos para los proyectos venideros. Este problema no ha podido ser solucionado por factores esenciales como la inexperiencia del personal y deficiente conocimiento en la planificación, monitoreo y control de los proyectos.

Ante la situación problemática descrita anteriormente surge el siguiente **problema a resolver**: El análisis deficiente de los indicadores de gestión de proyecto afecta la toma de decisiones sobre la planificación, asignación de recursos y el aprovechamiento del fondo de tiempo de los proyectos de desarrollo de software de la UCI.

Los indicadores de gestión de proyecto son:

- Plan.
- Índice de Calidad del Dato (ICD).
- Índice de Rendimiento de la Planificación (IRP).
- Índice de Rendimiento de la Ejecución (IRE).
- Índice de Ejecución del Proyecto (IE) o Real.

# *Introducción*

---

Las planificaciones de los proyectos de desarrollo de software de la UCI, se planifican a través de un cronograma donde se definen los hitos principales del proyecto, los recursos asignados y el fondo de tiempo para realizar cada tarea.

Actualmente las decisiones tomadas se ven afectadas porque, en ocasiones se planifica un periodo de tiempo para la ejecución de un proyecto, donde los hitos principales rompen la ruta crítica de su desarrollo incide directamente en su planificación inicial, incumpliendo de esta forma con el tiempo real asignado para realizar todas las tareas. Lo mencionado anteriormente, conlleva al mal aprovechamiento del fondo de tiempo, reduciendo la calidad, eficacia y eficiencia en la realización de las tareas y la incorrecta asignación de recursos al proyecto. Todo esto trae consigo el aumento del costo y tiempo; y la disminución de la calidad.

A partir del problema de investigación planteado se define como **objeto de estudio**: Proceso de toma de decisiones. Enmarcado en el **campo de acción**: Proceso de toma de decisiones en la actividad productiva de la UCI.

Teniendo como **objetivo general de la investigación**: Desarrollar un sistema de reportes para apoyar la toma de decisiones sobre la planificación, asignación de recursos y el aprovechamiento del fondo de tiempo de los proyectos de desarrollo de software de la UCI.

Para dar cumplimiento al objetivo general de la investigación se definen los siguientes **objetivos específicos**:

1. Definir el marco teórico de la investigación referente al análisis de indicadores para evaluar la toma de decisiones en la actividad productiva de la UCI.
2. Realizar el análisis y diseño del sistema de reportes.
3. Realizar la implementación del sistema de reportes.
4. Validar la propuesta de solución mediante las pruebas de caja blanca, caja negra, pruebas internas, pruebas de aceptación y un cuasiexperimento de tipo preprueba/posprueba.

Se tiene como **Idea a defender**: El desarrollo de un sistema de reportes apoyará la toma de decisiones sobre la planificación, asignación de recursos y el aprovechamiento del fondo de tiempo de los proyectos de desarrollo de software de la UCI.



# *Introducción*

---

Como **resultado esperado** se obtendrá un sistema de reportes para apoyar la toma de decisiones sobre la planificación, asignación de recursos y el aprovechamiento del fondo de tiempo de los proyectos de desarrollo de software de la UCI.

## ***Métodos Científicos de Investigación***

Para el desarrollo de la investigación es necesario la fundamentación a través de los métodos científicos, siendo utilizados en la presente los siguientes:

### **Métodos Teóricos:**

***Histórico-Lógico:*** Se enfocó el estudio de los sistemas de reportes desde un enfoque histórico lógico de forma general. Permite determinar las principales causas del desenlace del problema, así como las posibles soluciones a poner en práctica, a partir del análisis de estudios anteriormente realizados.

***Análisis y síntesis:*** Permite desarrollar un estudio sobre los sistemas de reportes basados en indicadores en el mundo y las experiencias adquiridas durante la actividad productiva en la organización, además de enfatizar en las buenas prácticas asociadas a este proceso. Se realiza una síntesis de los conceptos y definiciones sobre el tema luego del análisis de la bibliografía.

***Sistémico:*** En función de organizar la información encontrada de manera dispersa y obtener una conceptualización más clara sobre sistemas de reportes basado en indicadores y su relación con la toma de decisiones que determinen el éxito o fracaso en los proyectos.

### **Métodos Empíricos:**

***Encuesta:*** Para conocer el estado del proceso de toma de decisiones en la UCI, y la opinión de los expertos acerca de la aplicación desarrollada.

***Entrevista:*** Se conocerá las especificaciones y funcionalidades de los sistemas similares existentes, además de las deficiencias que tiene para ser solucionada en el sistema de reportes a desarrollar, a partir de las necesidades del cliente. Además, para conocer acerca de los sistemas similares encontrados en la investigación tales como Gespro y Servidor de Reportes (SDR).

# *Introducción*

---

**Experimentación (Cuasiexperimento):** Con la experimentación se puede demostrar si el sistema desarrollado logra de manera práctica el objetivo general definido en la investigación, para alcanzar mayor calidad, disminuir el tiempo y el costo de ejecución de los proyectos de desarrollo de software de la UCI.

Para facilitar su comprensión, el documento está estructurado en tres capítulos, a continuación, se realiza una breve descripción:

**Capítulo 1:** Se analizan e investigan los sistemas de reportes y modelos existentes. Se referencian conceptos básicos, metodología y herramientas. Se estudia el proceso de toma de decisiones en la actividad productiva de la UCI.

**Capítulo 2:** Se describen las necesidades del negocio, así como la especificación de requisitos de información, funcionales y no funcionales. Se realiza el análisis, diseño e implementación del sistema de reportes.

**Capítulo 3:** Se aplican las pruebas de caja blanca, caja negra para validar la correcta implementación de las funcionalidades y las respuestas del sistema al ejecutar las mismas. Realizar las pruebas internas y de aceptación para validar el correcto funcionamiento de la solución desarrollada.

# *Capítulo 1: Fundamentación Teórica*

---

## **CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA**

### **1.1. Introducción**

En el presente capítulo se realiza una investigación de los aspectos relacionados con el desarrollo del sistema, sus características, y una descripción de los principales conceptos asociados al problema de la investigación para entender mejor la propuesta de solución. Se estudia el proceso de toma de decisiones en la UCI. Además, se define la metodología, herramientas y lenguaje para el desarrollo del sistema a proponer.

### **1.2. Conceptos fundamentales de la investigación**

#### **Proyecto**

Un proyecto tiene recursos materiales, humanos y financieros, objetivos, resultados, acción, gestión y tiempo. Son desarrollados por personas y, se encuentran condicionados por recursos limitados, donde además son planificados, ejecutados y controlados. Los proyectos tienen un tiempo limitado para todo su ciclo de vida ya que cada uno tiene un inicio y un fin. La característica fundamental de los proyectos es que son únicos (9).

Es importante conocer estos factores que influyen en el correcto funcionamiento de la planificación del proyecto y además de su finalización con excelentes resultados.

#### **Factores críticos de éxito de los proyectos**

Los proyectos poseen elementos claves o básicos que determinan su éxito o fracaso. A continuación, algunos de los factores críticos o fundamentales para cualquier proyecto. Las técnicas participativas y de capacitación deben ser las apropiadas para la entidad que labora en conjunto con la entidad desarrolladora.

- En los proyectos productivos, se debe realizar una correcta selección de actividades a ser implementadas, que tiendan a presentar reales perspectivas en cuanto a la finalidad del proyecto.
- Detallar el funcionamiento de las actividades o metodología del proyecto. Es definir 'qué hacer' y también en "cómo hacerlo".

# *Capítulo 1: Fundamentación Teórica*

---

- Brindar asistencia técnica, formación y capacitación a la entidad beneficiada. Estos servicios deben ser esencialmente actividades de tipo participativo. Deben ser entendidos como una acción educativa y su principal propósito debe ser que la entidad beneficiada se apropie de nuevos conocimientos (tecnología) y experiencias en correspondencia con el producto que se le entregue (10).

Resulta importante para cada empresa que emprende el camino de desarrollar proyectos informáticos para comercializar que los mismos culminen exitosamente, tal es el caso de la UCI. La UCI cuenta con proyectos que han culminado de manera exitosa y otros que no. Por lo que, sería ventajoso aplicar las técnicas antes mencionadas para que los factores críticos sean positivos y de esta forma conseguir el éxito deseado.

Si la Universidad pudiera contar con un sistema donde el usuario pueda ingresar datos del proyecto en el cual está trabajando y permitiese compararlo con proyectos almacenados en la base de datos de proyectos terminados. Para brindar opciones de análisis de los datos históricos y estado por proyectos. Además, de obtener estadísticas generales dado los posibles estados, naturaleza y clasificación de los proyectos almacenados, posibilitaría a los usuarios (ingenieros en la gestión de proyectos) identificar los factores críticos a partir de los patrones de comportamiento de los indicadores en determinadas circunstancias que el sistema posibilitaría obtener. Otra opción es que podría utilizarse para la formación de los ingenieros tanto en pre-grado como en post-grado como casos de estudio.

## **Indicadores**

Los indicadores permiten a la organización medir sus resultados, obtener reportes de sus actividades y son los elementos que determinan sustancialmente las decisiones que se toman eventualmente por parte de la entidad que desarrolla el proyecto. El uso de indicadores es una herramienta para conocer donde estamos, hacia donde nos dirigimos y obtener una evaluación de resultados obtenidos mediante el uso de las métricas (11).

Los indicadores son elementos medibles, y que definen el estado de determinados objetivos con respecto a valores específicos definidos por quienes los utilizan en los diferentes ámbitos de

# Capítulo 1: Fundamentación Teórica

---

trabajo o desarrollo. Ayudan a determinar qué rumbo y qué acciones tomar para mejorar los aspectos que se encuentren por debajo de lo planificado y esperado.

## 1.3. Indicadores para la propuesta de solución

En la investigación se emplean los datos de los indicadores obtenidos de la BD de proyectos terminados de la herramienta de gestión de proyectos Gespro. De los nueve indicadores que controla la herramienta se utilizarán solamente los cinco explicados a continuación porque son el pilar fundamental para dar solución al problema planteado en la investigación:

- **Plan:** Muestra el porcentaje de realización de la ejecución del proyecto (según los hitos de ejecución cerrados) previsto hasta la fecha en que se realiza el análisis en relación a todo el ciclo de vida del proyecto. Adquiere valores entre 0 y 100. Interpretación: a mayor por ciento, mayor nivel de ejecución (12).

Fórmula para calcular este indicador:

$$Plan = 100 * \Sigma \text{impacto\_hitos\_previstos\_hasta\_la\_fecha} \div \Sigma \text{impacto\_hitos\_del\_proyecto}$$

- **Índice de Ejecución del Proyecto (IE) o Real:** Muestra el porcentaje de realización de la ejecución del proyecto (según los hitos de ejecución cerrados) hasta la fecha en que se realiza el análisis en relación a todo el ciclo de vida del proyecto. Adquiere valores entre 0 y 100. Interpretación: a mayor por ciento, mayor nivel de ejecución. Cuando el por ciento que representa el indicador de ejecución (real) es menor al plan está atrasado y por encima del plan está adelantado. El escenario ideal es cuando el por ciento del IE es igual al Plan. Objetivo de IE: Analizar las tareas e hitos ejecutados en el proyecto a partir del esfuerzo y progreso evidenciado en la ejecución de las mismas, con el fin de controlarlas (12).

Fórmula para calcular este indicador:

$$IE = 100 * \Sigma \text{impacto\_hitos\_cerrados\_hasta\_la\_fecha} \div \Sigma \text{impacto\_hitos\_del\_proyecto}$$

- **Índice de Rendimiento de la Ejecución (IRE):** Expresa la relación entre la sumatoria del impacto de los hitos cerrados y el sumatorio total del impacto de los hitos del proyecto planificados hasta la fecha de corte dd/mm/aaaa. Interpretación: IRE < 1, baja; IRE = 0, atrasada; IRE = 1, buena; IRE > 1, adelantada. Muestra el estado de realización de la ejecución del proyecto (según el porcentaje de ejecución de los hitos) hasta la fecha en que se realiza el análisis. Objetivo de IRE:

# Capítulo 1: Fundamentación Teórica

---

Analizar las tareas e hitos ejecutados en el proyecto a partir del esfuerzo y progreso evidenciado en la ejecución de las mismas, con el fin de controlarlas (12).

- **Índice de Rendimiento de la Planificación (IRP):** Muestra el estado de avance o progreso del proyecto. Relación entre la sumatoria de los porcentos de ejecución real y la sumatoria de los porcentos de ejecución planificado para cada una de sus tareas según la fecha de corte dd/mm/aaaa. Interpretación:  $IRP < 1$ , atraso;  $IRP = 0$ , atraso por no haber comenzado;  $IRP = 1$ , en tiempo;  $IRP > 1$ , adelantado. **Objetivo de IRP:** Analizar las tareas e hitos ejecutados en el proyecto a partir del esfuerzo y progreso evidenciado en la ejecución de las mismas, con el fin de controlarlas.
- **Índice de Calidad del Dato (ICD):** Propone el análisis de la completitud y correcta entrada de los datos a la aplicación. Se construye a partir de cuatro sub-indicadores, ejemplo el *Índice de Calidad del Dato de la Planificación (ICDP)* y *Calidad del Dato de la Ejecución (ICDE)*, *Índice de Calidad del Dato de la Planificación General (ICDPG)*, *Índice de Calidad del Dato de la Evaluación (ICDV)* donde su interpretación es entre más cercana a 1, mejor (12). **Objetivo de ICD:** Analizar el índice de calidad de los datos con el fin de verificar su integridad y completitud.

Cada variable es sometida a un conjunto de reglas para la clasificación del resultado lingüístico correspondiente a la evaluación (Bien, Regular o Mal) alcanzada por el indicador. Si el resultado del indicador es positivo, entonces se puede realizar el análisis de los indicadores de planificación, de lo contrario deberá verificarse en cuáles de las medidas derivadas especificadas anteriormente existe algún problema para su inmediata resolución (12).

## 1.4. Sistemas de Reportes

Se puede definir de forma general como sistema al conjunto de partes que están integradas con el propósito de lograr un objetivo. Por otra parte, un sistema de información es un conjunto de elementos que interactúan entre sí con el fin de apoyar las actividades de una empresa o negocio. También, total de procedimientos, operaciones, funciones y difusión de datos o información en una organización. Un sistema de información realiza actividades básicas: almacenamiento, procesamiento y salida de información (13). Se puede considerar a partir de las definiciones

# Capítulo 1: Fundamentación Teórica

---

anteriores que el objetivo de un proyecto informático es la producción de un producto, en este caso, un software y que, el mismo constituye un sistema de información.

Los reportes o informes, son una manera bien formateada de presentar los datos que se hayan introducido. Los reportes se relacionan directa o indirectamente, con consultas de bases de datos y muestran los resultados en un formato específico. Los reportes se usan para captar información fuera de un montón de datos (14).

Los sistemas de reportes representan la forma más acertada a la hora de brindar informaciones claras y precisas, permitiendo establecer comparaciones, debido a que los reportes realizan gráficas a partir de los datos que contiene el sistema. Los datos expresados en forma de gráficas se convierten en información oportuna acerca de determinada empresa, negocio u organización en cuanto a indicadores específicos definidos inicialmente donde se obtienen determinados resultados. En la actualidad son muy usados por su utilidad en cuanto a los análisis de diferentes fenómenos o comportamientos dentro de empresas u organizaciones (14).

La generación de reportes a partir de la información almacenada en una fuente de datos representa un apoyo significativo en la toma de decisiones de las organizaciones. Con estos fines fueron desarrolladas la herramienta Gespro y el Servidor de Reportes (SDR) en la UCI. A continuación, se desglosan estos sistemas:

## **Servidor de Reportes (SDR)**

El Servidor de Reportes (SDR) versión 1.0 se basa en servicios web que son consumidos por las aplicaciones clientes sin necesidad de una interfaz de usuario que medie entre los dos sistemas. El mismo utiliza como motor para generar reportes el JasperReport. SDR se encuentra vinculado al Generador de Reportes (GDR). SDR tiene la limitación que la entrada de sus datos debe ser en jrxml (15).

## **Herramienta de gestión de proyectos Gespro**

Según el PMBOK, un software de gestión de proyectos para programación permite hacer un seguimiento de las fechas planificadas en comparación con las fechas reales, informar sobre las desviaciones en el avance con respecto a la línea base y pronosticar los efectos de los cambios en el cronograma del proyecto (16).

# Capítulo 1: Fundamentación Teórica

---

Garantiza una vista completa al estado del proyecto respecto a la calidad, costos, tiempo, recursos humanos y contratos, a través del cálculo automático de indicadores y visualización de reportes de estado. Basa su funcionamiento en la implementación de buenas prácticas sugeridas por los estándares de mejora de procesos PMBOK y CMMI, permitiendo la gestión de las áreas de conocimiento y procesos asociados con dichos estándares (7). El sistema para la dirección integrada de proyectos Gespro 16.05 incluye módulos para la planificación, alcance y tiempo, la gestión de riesgos, la calidad, los costos, las adquisiciones, los interesados, las comunicaciones, los recursos humanos y la gestión de integración (17).

El estudio del avance de los proyectos, el estado y el éxito de los proyectos se realiza y se lleva un control mensual. Sacan todos los indicadores de ellos y determinan un estado a partir la interpretación de los indicadores. Los datos de los proyectos se resumen en un Excel, se guardan los datos de los proyectos, indicadores, recursos humanos que tiene, las facturaciones y características del proyecto, el Gespro guarda los datos de manera histórica de los últimos 5 cortes (cada corte es por semana). Los resultados del reporte se almacenan en el Excriba, se guardan los datos de cada corte y en el Excel. Con los indicadores Plan, Real (estado del progreso), IRE (rendimiento del progreso ejecutado), IRP (rendimiento del progreso planificado), se puede determinar el estado de un proyecto (8).

## 1.5. Análisis comparativo de los sistemas de reportes

Los sistemas de reportes antes mencionados, poseen características a fines en cuanto al objetivo general que se persigue con el desarrollo del sistema de reportes a implementar. A continuación, se establece una comparación de los mismos de acuerdo a indicadores definidos para darle solución a la problemática planteada:

**Generar reportes:** Se establece este indicador dada la importancia que tiene la representación de los datos para el usuario, para llevar a cabo comparaciones. Identifica la posibilidad o no de un sistema de generar información representada en gráficos, a partir de determinados datos.

**Comparar proyectos:** El presente indicador se establece para demostrar la necesidad de crear un sistema que permita a partir de los reportes que el mismo emite, establecer de una forma



# Capítulo 1: Fundamentación Teórica

---

automatizada comparaciones que a su vez faciliten el trabajo del usuario. Identifica si el sistema posee la capacidad de permitir comparar las evaluaciones o indicadores de un proyecto con otros.

**Brindar información de datos históricos:** Este indicador demuestra cuán importante son los datos antiguos para realizar análisis más profundos y tomar acciones según lo mostrado. El indicador ofrece información sobre la funcionalidad de un sistema para crear estadísticas que analicen datos históricos.

Indicadores	Generar	Comparar	Brindar información
Sistemas	Reportes	proyectos	de datos históricos
SDR	Sí	No	No
Gespro	Sí	No	Solo semanal

Tabla 1. Comparación entre los sistemas SDR y Gespro.

El sistema SDR genera reportes, pero no brinda información de datos históricos limitándole la comparación de proyectos mediante el análisis del estado real de los proyectos en ejecución y los terminados. Por tanto, este sistema no aporta elementos que puedan guiar la investigación.

La herramienta Gespro genera reportes a través del sistema SDR, pero no muestra información de datos históricos de un proyecto en ejecución, solo muestra el estado de avance del mismo, diario y semanal, de igual manera limita realizar la comparación de proyectos mediante el análisis del estado real de los proyectos en ejecución y los terminados. Aunque de este sistema si se guardan los datos históricos de proyectos terminados, pero no contenidos dentro del propio sistema, se almacenan en una base de datos independiente. A pesar de estas contradicciones de la herramienta se utilizarán los datos de cinco indicadores de los proyectos terminados, para poder realizar la comparación entre proyectos y de esta forma dar solución al problema planteado en la investigación.

## 1.6. Toma de decisiones

La toma de decisiones se puede ver como un proceso continuo que va desde el estudio del entorno, el estudio de las técnicas para la recolección de los datos, hasta obtener la información que se necesita para poner en práctica una acción. Es un proceso en el cual comienza por una

# *Capítulo 1: Fundamentación Teórica*

---

necesidad de información, se recopilan los datos, se analizan y se selecciona un curso de acción entre varias alternativas. Su efectividad radica en el análisis de los datos y la información (18).

Conseguir que el éxito sea un referente en la gestión empresarial depende de la habilidad de tomar buenas decisiones. Las decisiones inteligentes hacen que los proyectos prosperen, en cambio las decisiones tomadas a la ligera bloquearán y perjudicarán el rendimiento del equipo de trabajo del proyecto (18). Por tanto, la prosperidad de una empresa radica básicamente en las decisiones que tome.

## **Modelos y métodos para la toma de decisiones**

### **Modelos de toma de decisiones**

Para la toma de decisiones en grupo existen diferentes modelos que se pueden aplicar de acuerdo con el problema y objetivos específicos dados en el ámbito de una organización o empresa. A continuación, los tipos de ejemplos de modelos que existen:

- **Modelo racional:** El modelo racional implica el proceso de elegir entre varias alternativas a efecto de maximizar los beneficios para la organización. Las preferencias y las opciones de individuos y organizaciones están en función de la mejor alternativa para la organización entera.
- **Modelo de la racionalidad limitada:** Describe las limitaciones de la racionalidad y hace hincapié en los procesos de la toma de decisiones que las personas o los equipos utilizan con frecuencia.
- **Modelo de la administración basada en evidencias:** El modelo parte de la premisa de usar el diagnóstico mejor, más profundo y de hechos, en la medida de lo posible, abarca elementos centrales del modelo racional, pero en una versión ampliada y aplicada en la toma de decisiones gerenciales.
- **Modelo político:** Se enfoca en los empleados y estimula la creatividad para descubrir problemas, oportunidades y hacer elecciones novedosas para resolver dichos problemas (18).

### **Métodos para la toma de decisiones**

# Capítulo 1: Fundamentación Teórica

---

Existen 2 tipos de decisiones: decisiones de certidumbre y decisiones de incertidumbre. Las decisiones de certidumbre son cuando el decisor cuenta con toda la información sobre la decisión, conoce su resultado final y cómo se va a comportar. Las decisiones de incertidumbre son aquellas que el decisor toma sin un conocimiento completo de la situación y cada estrategia corresponde a diversos resultados (18). A continuación:

- Método de Expertos: En este método se hace necesario en el análisis de sistemas complejos el concurso de expertos, con su poder de análisis, conocimiento, razonamiento, experiencia e intuición, pueden disminuir la complejidad de la problemática a resolver al cuantificar sus criterios en un rango de valores.
- Método Delphi: Este método tiene como objetivo alcanzar un consenso sobre una estimación, a través de reuniones, cuestionarios y encuestas.
- Método de Jerarquías Analíticas (AHP): Es un método que se basa en descomponer jerárquicamente el problema a tratar en sus elementos componentes (18).

## 1.7. Estado del proceso de toma de decisiones en la UCI

Los modelos y métodos analizados en el epígrafe anterior, fueron utilizados en la encuesta como indicadores para valorar el estado del proceso de toma de decisiones en la Universidad.

Para conocer el estado del proceso de toma de decisiones en la UCI, se aplicó una encuesta (**Anexo 1**) con el objetivo de determinar los problemas existentes y fundamentar la necesidad de realizar un sistema de reportes que lo favorezca. A continuación, se muestran los resultados obtenidos:

La actividad desarrollo-producción de la UCI es soportada por una Red de Centros, integrada por 15 centros de desarrollo. En los que se desarrollan proyectos productivos y de servicios para la informatización de la sociedad cubana, la exportación y la informatización de los procesos de la propia Universidad (20). Para la selección de los elementos de la muestra se toma el muestreo aleatorio simple donde se seleccionan al azar los elementos. Es el ideal en poblaciones pequeñas (21).

La población definida es la cantidad de centros de desarrollo de la UCI: 15 centros por lo que se tomará como muestra el 30%, lo que equivale a 4,5; se redondea por defecto a 4. Los centros para

# Capítulo 1: Fundamentación Teórica

la muestra fueron: CEGEL, CEIGE, VERTEX, y DATEC. Los resultados arrojados por la encuesta aplicada, clasificados por Centros se encuentran en los Anexo 2: Resultado de la encuesta en el Centro CEIGE. A continuación la Figura 1 muestra de manera general los resultados de la misma:

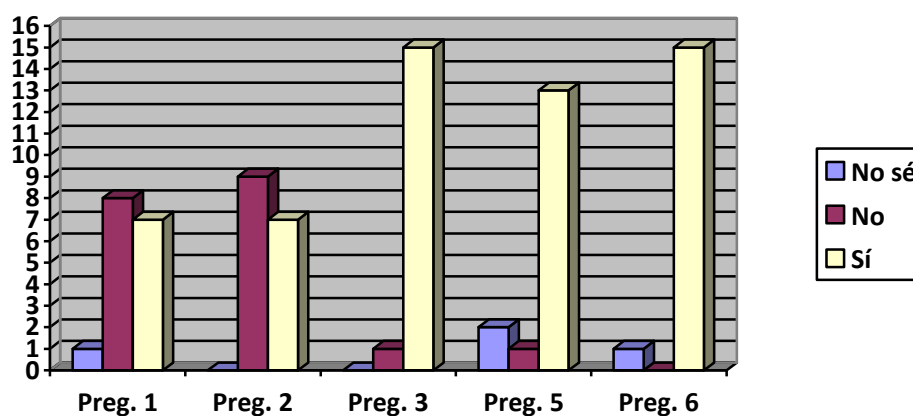


Figura 1: Resultados generales sobre el proceso de toma de decisiones en la UCI.

Preguntas	Sí	No	No sé
Preg. 1	43,75%	50%	6,25%
Preg. 2	43,75%	56,25%	-
Preg. 3	93,75%	6,25%	-
Preg. 5	81,25%	6,25%	12,50%
Preg. 6	93,75%	-	6,25%

Tabla 2: Porcentajes obtenidos correspondientes a la Figura 1.

A partir de los resultados obtenidos en la pregunta #1 muestra que: un 43,75% sí conoce los Modelos de toma de decisiones, mientras que el 50% los desconoce. Los resultados de la pregunta #2, expresan que el 43,75% reconoce haber utilizado algunos de estos modelos, mientras que el resto no ha utilizado ninguno de ellos. El modelo más seleccionado fue el Modelo Relacional.

Los resultados arrojados por la pregunta #3 fueron: sólo el 93,75% respondió haber utilizado algún método, el más seleccionado fue el Método de Expertos. La pregunta #4 enuncia que la herramienta Gespro ha sido la más usada. Esta pregunta se graficó individualmente por las características de la misma, los resultados se pueden apreciar en la figura del **Anexo 2**: Resultado

# *Capítulo 1: Fundamentación Teórica*

---

de la encuesta en el Centro CEIGE.. La pregunta #5 arrojó que sólo el 81,25% encuentra importante la medición o el uso de indicadores que evalúen el trabajo en la gestión de proyectos.

La creación de una herramienta para contribuir a la toma de decisiones que, a partir de indicadores específicos (obtenidos de Gespro) muestre reportes estadísticos de los proyectos realizados en la UCI; y permita comparar resultados de proyectos anteriores con los proyectos actuales, es apoyada por el 93,75% de los encuestados. Estos fueron los resultados de la pregunta #6 que evidencia la necesidad de desarrollar un sistema de reportes basados en indicadores que favorezca la toma de decisiones.

Los resultados de la encuesta muestran el grado de conocimiento con el que cuenta nuestra Universidad sobre el proceso de toma de decisiones y la importancia que le es otorgada a la medición en la gestión de proyectos. Este proceso tiene una incidencia importante en los resultados productivos de cualquier entidad. El escenario ideal sería que, a partir de las decisiones tomadas, el cliente obtenga lo que necesita en el tiempo acordado.

Aumentar la preparación de los integrantes de los equipos de proyectos y estandarizar el proceso de toma de decisiones en la Universidad, podría ser la forma de lograr mejores resultados en la actividad productiva. Mejores resultados aumentarían la competitividad de la Universidad, tanto en el mercado nacional como el internacional.

## **1.8. Componente Humanístico.**

En la UCI actualmente se encuentra la herramienta de gestión de proyectos Gespro, la misma provee a los equipos de desarrollo de proyectos de cada Centro de la Universidad y al Departamento de Gestión de Producción (DGP) una serie de opciones para planificar, controlar y gestionar los proyectos de la Universidad.

Gespro es una herramienta que sirve para el apoyo a la toma de decisiones y utiliza para ello un conjunto de indicadores específicos para la organización, los cuales permiten determinar el estado de los proyectos. Realiza en tiempo real cortes a los proyectos de una semana de producción y muestra los resultados de esa etapa.

# Capítulo 1: Fundamentación Teórica

---

El sistema de reportes basados en indicadores que se desea desarrollar, forma parte de una pirámide de investigación en la que el cliente es la Ingeniera Virtudes Milagro Figueredo Lara<sup>3</sup>, con el fin de que pueda ser implementado en la actividad productiva de los Centros de desarrollo de la UCI. El cliente tiene la necesidad de la creación del sistema conociendo que en la Universidad existe una BD de proyectos terminados, pero no posee una interfaz gráfica impidiendo realizar un grupo de análisis estadísticos de los proyectos terminados. Además, que no permite comparar los datos de proyectos terminados con la información actual que genera Gespro.

El sistema de reportes basados en indicadores SRBin como se propone nombrar, permitirá que el usuario pueda introducir datos del proyecto que se desea analizar y gestionarlo. Luego podrá establecer una búsqueda a partir de que seleccione de que *Naturaleza, Clasificación, Estado* de los proyectos que desea listar para comparar con el suyo, ver las diferencias positivas y negativas de su(s) proyecto(s) con respecto al que eligió. Podrá obtener reportes de los datos de los indicadores almacenados de ambos proyectos.

SRBin permitirá obtener reportes de las estadísticas generales de los proyectos almacenados en el sistema, permitiéndole escoger el tipo de gráfico en el que desea visualizar sus reportes, así como exportar todos los reportes que sean realizados por el sistema en 8 formatos distintos, que incluyen imagen, documentos PDF y Excel, entre otros.

La importancia de realizar el SRBin, es que, al reutilizar los datos de los proyectos finalizados de la Universidad, permita a los equipos de desarrollo que inicien un nuevo proyecto revisar los ya culminados, establecer patrones de comportamiento, comparaciones y apreciar semejanzas y diferencias. Se espera que con el sistema SRBin que se propone en esta tesis, se pueda identificar posibles riesgos, ventajas, limitaciones, y similitudes, lo cual tendrá una incidencia positiva en la toma de decisiones del nuevo equipo de desarrollo, luego de ser identificadas tanto las debilidades como las fortalezas.

El sistema tiene impacto económico en cuanto a la eficiencia, ya que puede contribuir a una mayor productividad del trabajo al poder revisar proyectos en estado “Terminado” y obtener su

---

<sup>3</sup> Ingeniera en Ciencias Informáticas. Graduada de la UCI, 7 años de experiencia en el área de gestión de proyectos.

# Capítulo 1: Fundamentación Teórica

---

información básica, partiendo del nombre del proyecto el usuario se puede remitir a la herramienta de gestión documental Excriba para obtener toda la documentación asociada al mismo y utilizarla para beneficiar al proyecto actual. Se ganará en organización del trabajo en el área de proyectos, ya que existirá un ahorro de tiempo al poder reutilizar información existente y previamente puesta en práctica siempre que se hayan tenido buenos resultados o modificando acciones.

Asimismo, el sistema debe facilitar el trabajo del personal partiendo del hecho que funciona sobre software libre, código reutilizable y con estándares de codificación para el desarrollo de futuras versiones. Desde el punto de vista ético, el sistema propuesto al ser usado por los integrantes de los diferentes proyectos desarrollará en mayor grado la responsabilidad, confidencialidad de los involucrados, así como su profesionalidad.

## 1.9. Metodologías para el desarrollo de software

Las metodologías para el desarrollo de software se basan en la utilización de modelos, técnicas y herramientas utilizados durante el ciclo de vida de un software. Las metodologías ágiles, están más orientadas a la generación de código con ciclos muy cortos de desarrollo, se dirigen a equipos de desarrollo pequeños, hacen especial hincapié en aspectos humanos asociados al trabajo en equipo e involucran activamente al cliente en el proceso (22).

La metodología para el desarrollo del software es una metodología ágil, la AUP (Proceso Unificado Ágil), pero en su variación o adaptación a los procesos de desarrollo de software de la UCI. Se escoge por encontrarse definida dentro de los procesos institucionalizados y certificados de la actividad productiva de la UCI en el Nivel 2 de CMMI-DEV v1.3. A continuación, se detallan aspectos importantes de la metodología.

### AUP-UCI

El ciclo de vida de los proyectos se recrea en tres fases: fase de Inicio, pero modificando el objetivo de la misma, se unifican las restantes tres fases de AUP en una sola, a la que llamaremos Ejecución y se agrega la fase de Cierre (23). A continuación, se describe la fase de Ejecución dónde se desarrollará el proyecto:

*Fase de Ejecución:* En esta fase de la variación AUP-UCI se reúnen: elaboración, construcción y transición. Se ejecutan las actividades requeridas para desarrollar el software, incluyendo el ajuste

# *Capítulo 1: Fundamentación Teórica*

---

de los planes del proyecto considerando los requisitos y la arquitectura. Durante el desarrollo se modela el negocio, obtienen los requisitos, se elaboran la arquitectura y el diseño, se implementa y se libera el producto (23). En esta fase estará comprendido el desarrollo del sistema propuesto como solución al problema planteado.

## **Disciplinas de AUP-UCI**

Las disciplinas definidas que utiliza esta metodología se desarrollan en la etapa de Ejecución de la metodología. Donde se realizan iteraciones y se obtienen resultados incrementales. Las disciplinas a implementar serán: Requisitos, Análisis y diseño, Implementación, Pruebas internas y Pruebas de aceptación.

## **Escenarios de la metodología**

En la metodología AUP-UCI están definidos 4 escenarios para modelar el negocio, donde cada escenario está diseñado para los 4 casos básicos en los que puede encontrarse cualquier proyecto que se desee desarrollar. El escenario donde se desarrollará el proyecto es el número 4, el mismo aplica para los proyectos que hayan evaluado el negocio a informatizar y como resultado obtengan un negocio bien definido. El cliente estará acompañando el desarrollo del proyecto para convenir los detalles de los requisitos, así como poder probarlos y validarlos. Este escenario es utilizado en proyectos que no modelen negocio donde solo se pueden modelar los requisitos con Historias de Usuarios (23).

## **1.10. Herramientas y lenguajes para el desarrollo de software**

### **Lenguaje de Modelado UML 2.0**

UML (Unified Modeling Language) o Lenguaje de Modelación Unificado es un lenguaje gráfico para detallar, construir, visualizar y documentar las partes o artefactos en el proceso de desarrollo de software (24). UML es un lenguaje visual de modelado para visualizar, especificar, construir, y documentar los artefactos en un sistema de software, es decir, UML es el lenguaje que estandariza la forma de crear diagramas, el significado preciso de los mismos, y las relaciones existentes entre ellos (25). Los diagramas de diseño de clases, componente y despliegue serán realizados utilizando este lenguaje.



# Capítulo 1: Fundamentación Teórica

---

## **Visual Paradigm for UML 8.2**

Visual Paradigm es una herramienta CASE para modelamiento UML muy potente, gratuita, fácil de instalar, utilizar y actualizar. Te permite dibujar todo tipo de diagramas UML, revertir código fuente a modelos UML, generar código fuente desde los diagramas UML, y mucho más. Incluye los objetos de UML además de diagramas de casos de uso, diagramas de clase, diagramas de componentes, reversa instantánea para Java, C++, entre otros. Es una herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue (26). Es Visual Paradigm la herramienta con la que se realizará el modelamiento de los diagramas para el desarrollo del software.

## **Lenguajes de programación utilizados**

### **HTML 5**

Se utiliza el lenguaje de programación web HTML5, el cual propone estándares para cada elemento de la web y también un propósito claro para cada una de las tecnologías involucradas. Tiene los elementos necesarios para ubicar contenido estático o dinámico, además, provee los elementos estructurales, CSS se encuentra involucrado en cómo volver toda esa estructura utilizable y atractiva a la vista, y JavaScript tiene todo el poder necesario para proveer dinamismo, seguridad y construir aplicaciones web completamente funcionales (27).

### **CSS 3**

Para crear los estilos web se utiliza CSS que es un lenguaje que trabaja con HTML para proveer estilos visuales a los elementos del documento como el tamaño, color, borde, fondo, entre otros. Es la forma de separar la estructura de la presentación. CSS ha crecido y ganado importancia, pero siempre desarrollado en paralelo, enfocado en las necesidades de los diseñadores y apartado del proceso de evolución de HTML (27).

### **JavaScript 1.6**

JavaScript es un lenguaje interpretado usado para múltiples propósitos. Posee motores de interpretación, creados para acelerar el procesamiento del código (27). Es un lenguaje compacto, y basado en objetos, diseñado para el desarrollo de aplicaciones cliente-servidor a través de internet. Tiene la particularidad que se encuentra insertado dentro del mismo documento HTML

# Capítulo 1: Fundamentación Teórica

---

(28). Se utilizará la librería **amChart.js** de JavaScript para la generación de las gráficas y reportes que debe mostrar el sistema.

## **PHP 7**

PHP es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas. Entre sus características están la inclusión de la visibilidad, las clases abstractas, clases y métodos finales, manejo de excepciones, interfaces, clonación y tipos sugeridos. Forma parte de las tecnologías LAMP (Linux, Apache, MySQL, PHP/Python/Perl), siendo esta plataforma una de las más extendidas en cuanto software open source para desarrollo web se refiere (29). Se utilizó este lenguaje en conjunto con los antes mencionados e integrándose con el marco de trabajo Laravel para crear la propuesta de solución.

## **Marco de trabajo para desarrollar la propuesta de solución**

### **Laravel 5.4**

Laravel es un framework de desarrollo web escrito en PHP. Se ha diseñado para mejorar la calidad de su software, reduciendo los costos del mantenimiento en curso de desarrollo inicial, proporcionado sintaxis expresiva, clara y una funcionalidad de conjunto que ahorrará horas de tiempo de implementación. Laravel permite cambiar fácilmente y extiende el almacenamiento en caché, las sesiones, la base de datos y la funcionalidad de autenticación. Es un recurso valioso de bibliotecas y subsistemas de terceros (30).

Posee migraciones de bases de datos habilitadas para diseñar y modificar fácilmente bases de datos de forma independiente de la plataforma. Las migraciones se pueden ejecutar contra cualquiera de los tipos de base de datos que soporta Laravel (MySQL, PostgreSQL, MSSQL y SQLite) (30). El framework permite la arquitectura de tipo Modelo-Vista-Controlador para el desarrollo de aplicaciones web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja.

### **Eloquent ORM**

Laravel incluye un sistema de mapeo de datos relacional llamado Eloquent ORM (Mapeador Relacional de Objetos) que facilita la creación de modelos (31). Con Eloquent, se puede crear, recuperar, actualizar y borrar el registro de la base de datos. Además, ofrece la gestión de relaciones e incluso puede manejar la paginación de forma automática (30). Es utilizado en el

# Capítulo 1: Fundamentación Teórica

---

desarrollo de la aplicación a partir de su integración con el marco de trabajo Laravel para el mapeado de las bases de datos relacional.

## **Gestión y administración de bases de datos**

### **PostgreSQL 9.4**

PostgreSQL ofrece un poderoso sistema de roles para controlar la forma en que los usuarios pueden acceder a los objetos en la base de datos como son las tablas, vistas, secuencias y funciones, es decir las partes con las que interactúa el usuario. PostgreSQL es un gestor de bases de datos relacionales organizado en diferentes niveles. El puerto por defecto es 5432 y es el mismo que se asume en la presente tesis. Una base de datos (database) es una colección de esquemas (schema), quienes, a su vez, finalmente contienen los objetos que se conocen habitualmente en el contexto de técnicas de almacenamiento de datos, tales como tablas (table), vistas (view), secuencias (sequence) (32).

PostgreSQL utiliza un modelo cliente/servidor y usa multiprocesos en vez de multihilos para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afectará el resto y el sistema continuará funcionando (33).

### **PgAdmin III**

PgAdmin III es una plataforma de administración y desarrollo Open Source. La aplicación puede utilizarse en plataformas Linux, FreeBSD, Solaris, MacOS y Windows. PgAdmin está diseñado para responder a las necesidades de los usuarios, desde escribir consultas SQL sencillas hasta desarrollar bases de datos complejas. La interfaz gráfica puede ejecutarse en el escritorio o en un servidor web. La aplicación incluye una sintaxis que resalta el editor de SQL (34). PgAdmin es la interfaz gráfica que permite la administración de la base de datos en este caso vinculada con PostgreSQL9.4 que es el gestor de bases de datos que se utiliza para desarrollar el sistema de reportes.

## **Entorno de Desarrollo Integrado (IDE)**

### **PhpStorm 2017.1.2**

Se utiliza el PhpStorm como un Entorno de Desarrollo Integrado (IDE) inteligente para desarrollar aplicaciones en Pre-procesador de hipertextos (PHP), proporcionando herramientas esenciales

# Capítulo 1: Fundamentación Teórica

---

como refactorización, análisis de código y comprobación de errores. Las principales novedades en PhpStorm 2017.1.2 incluyen:

- Soporte para PHP 5.3 en adelante.
- Depuración con cero configuraciones con todos los navegadores.
- Compatibilidad con el marco de trabajo Laravel.
- Editores de Lenguaje de Consulta Estructurado (SQL) con resultados editables.
- Soporte para Lenguaje Marcado de Hipertextos (HTML5) (35).

## **Servidor de Aplicaciones**

### **Apache 2.4.18**

El servidor web Apache se basa en un concepto muy simple de división del sistema en módulos, representando estos la división del trabajo dentro del sistema, valga la redundancia. Este servidor tiene un Núcleo (core) que lleva a cabo todas las funcionalidades importantes del sistema, y los módulos, externos a este núcleo, añaden funcionalidades al servidor web, configurándose cada uno de ellos por separado. Es por ello que se obtiene globalmente un sistema muy flexible, que permite ser configurado y optimizado acorde a unas necesidades mediante la implementación de módulos externos (36).

### **1.11. Conclusiones parciales**

El desarrollo del marco teórico de la investigación permitió caracterizar los sistemas de reportes existentes, posibilitó conocer las deficiencias de estos con el objetivo de fundamentar la necesidad de desarrollar un sistema que compare proyectos y realice reportes estadísticos, utilizando indicadores basados en datos históricos. Se analizaron los datos de la encuesta aplicada para determinar el estado actual de la toma de decisiones en la UCI. Se seleccionaron los lenguajes, herramientas y metodología que permitirán dar solución el problema planteado en la presente investigación.

# *Capítulo 2: Desarrollo de la solución*

---

## **CAPÍTULO 2: DESARROLLO DE LA SOLUCIÓN**

### **2.1. Introducción**

En el presente capítulo se exponen los elementos fundamentales del sistema, que permitirán tener una visión objetiva acerca del producto final. Se describe el proceso realizado durante las disciplinas de Requisitos, Análisis y diseño e Implementación definidas en la metodología.

### **2.2. Necesidades del negocio**

La necesidad de negocio de una organización puede deberse a una necesidad de formación, a una demanda del mercado, a un avance tecnológico, a un requisito legal o a una norma gubernamental (37). Las funcionalidades del negocio deben estar en correspondencia con las necesidades del cliente y en este caso, corresponde a un avance tecnológico.

Para concretar las necesidades del negocio se procedió a realizar una encuesta para determinar cómo se encuentra el proceso de toma de decisiones en los centros de desarrollo de la Universidad. Los resultados arrojaron que: el 81,25% encuentra importante la medición o el uso de indicadores en la producción de software, de igual manera un 93,75% apoya la creación de una herramienta que permita comparar proyectos terminados con proyectos en ejecución para favorecer la toma de decisiones.

### **2.3. Requisitos**

Un requisito es un atributo necesario para el sistema a desarrollar, en el cual se puede describir una funcionalidad o característica que tenga valor para los Stakeholder dentro del mismo (38). Los requisitos del sistema, son los que representan una necesidad del sistema (39). Este tipo de requisitos son el resultado de acoplar los requisitos de usuario al modelo de la solución. Estos ya tienen cierto nivel de detalle avanzado y son la base para empezar la fase de diseño del sistema. Hacen referencia hacia los requisitos funcionales y no funcionales (38).

#### **Técnicas para la obtención de requisitos de software**

Para realizar la captura de los requisitos fue necesario la utilización de técnicas para la obtención o levantamiento de requisitos. Las técnicas utilizadas son:

## Capítulo 2: Desarrollo de la solución

---

**Entrevista:** Se realizaron talleres, reuniones y entrevistas con el cliente, en este caso, la Ing. Virtudes Milagro Figueredo Lara, perteneciente al centro de Informatización de Entidades (CEIGE), quien posee varios años de experiencia en la gestión de proyectos de desarrollo de software.

**Tormenta de ideas:** En cada encuentro realizado fueron debatidos los requisitos identificados, arrojando como resultado que se obtuviera una visión más amplia de las funcionalidades a implementar, favoreciéndose de esta forma el avance en el desarrollo del sistema.

### **Especificación de los requisitos**

La especificación de requisitos es el producto del trabajo final que genera la ingeniería de requisitos. Sirve como base para la ingeniería de software subsecuentes. Describen la función y desempeño de un sistema basado en computadoras y las restricciones que regirán su desarrollo (40).

### **Requisitos Funcionales**

Los requisitos funcionales en un sistema describen lo que este debe realizar. Estas dependen del tipo de software que se esté desarrollando, de los usuarios, el software y el enfoque general adoptado por la organización al escribir requisitos. Los requisitos funcionales del sistema describen detalladamente las funciones del sistema, sus entradas y salidas, excepciones, etc. (41).

Los requisitos funcionales del sistema SRBin se muestran a continuación:

**RF1:** Registrar usuario.

**RF2:** Autenticar usuario.

**RF3:** Listar los usuarios registrados en el sistema.

**RF4:** Gestionar los usuarios del sistema.

**RF5:** Listar roles de los usuarios del sistema.

**RF6:** Gestionar roles de los usuarios del sistema.

**RF7:** Gestionar los nomencladores del sistema: *Centro, Estado, Naturaleza y Clasificación.*

**RF8:** Mostrar el listado de proyectos almacenados en el sistema y los creados por usuarios.

**RF9:** Gestionar los datos de todos los proyectos almacenados en el sistema.

**RF10:** Mostrar listado de proyectos creados por el usuario.

**RF11:** Gestionar los proyectos.

## Capítulo 2: Desarrollo de la solución

---

**RF12:** Visualizar las evaluaciones de los proyectos por semana.

**RF13:** Gestionar evaluaciones por semana de los proyectos creado por el usuario.

**RF14:** Visualizar los datos generales del proyecto.

**RF15:** Visualizar tabla de las evaluaciones por indicadores y gráficas de los proyectos del usuario.

**RF16:** Exportar los reportes en los formatos: **.jpg, .png, .pdf, .svg, .csv, .json y .xlsx.**

**RF17:** Búsqueda por nombre, estado, naturaleza y clasificación del proyecto.

**RF18:** Visualizar los datos de un proyecto seleccionado del resultado de la búsqueda.

**RF19:** Graficar el proyecto seleccionado por el usuario con sus proyectos mostrando una comparación entre los indicadores de los proyectos por semana.

**RF20:** Graficar las diferencias de los proyectos.

**RF21:** Eliminar o reincorporar proyectos en las gráficas de comparación y diferencia.

**RF22:** Guardar búsquedas.

**Descripción del requisito funcional: Exportar reportes en los formatos .jpg, .png, .pdf, .svg, .csv, .json y .xlsx**

A continuación, se describe mediante una historia de usuario el requisito Exportar reportes en 8 formatos: **.jpg, .png, .pdf, .svg, .csv, .json y .xlsx.** El resto de las Historias de usuarios se pueden observar en los **Anexo 7** Historia de usuario Autenticar usuario.- **Anexo 11:** Historia de usuario Guardar búsquedas.

<b>Número: 16</b>	<b>Nombre del requisito:</b> Exportar reportes en los formatos: <b>.jpg, .png, .pdf, .svg, .csv, .json y .xlsx.</b>
<b>Programador:</b> Irina E. Mayedo Tamayo	<b>Iteración Asignada:</b> 4
<b>Prioridad:</b> Media	<b>Tiempo Estimado:</b> 3 días
<b>Riesgo en Desarrollo:</b>	<b>Tiempo Real:</b> 60 horas
<b>Descripción:</b> La funcionalidad debe permitir exportar los reportes generados del proyecto seleccionado.	
<b>Prototipo de interfaz:</b>	

## Capítulo 2: Desarrollo de la solución

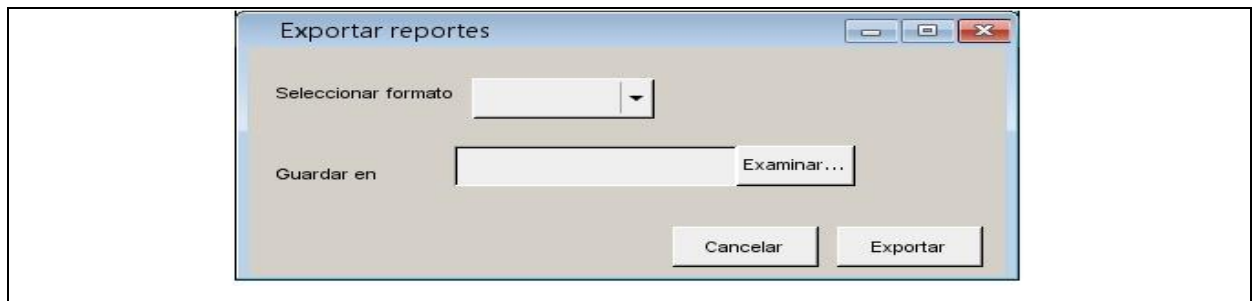


Figura 2: Historia de usuario de la funcionalidad Exportar reportes.

### Requisitos No Funcionales

Los requisitos no funcionales, como su nombre indica, son requisitos que no son directamente relacionados con los servicios específicos prestados por el sistema a sus usuarios. Pueden relacionarse con propiedades emergentes del sistema tales como fiabilidad, tiempo de respuesta, el rendimiento, la seguridad, y el especificar o restringir las características del sistema en su conjunto (41).

### RNF1 Usabilidad

**RNF1.1 Inteligibilidad:** El idioma de todas las interfaces de la aplicación será español.

**RNF1.2 Operatividad:**

- Los errores cometidos por el usuario les serán notificados.
- El sistema expondrá el menú general desde cualquiera de sus páginas.

**RNF1.3 Facilidad de aprendizaje:**

- Los botones siempre aparecen del lado derecho de la interfaz.
- La forma de llenar los formularios es de arriba hacia abajo.
- Los menús aparecen en el orden en que se lleva a cabo el negocio.

**RNF1.4 Capacidad de atracción:**

- El sistema ofrece una interfaz fácil de operar para el cliente.
- Diseño sencillo, con pocas entradas, permitiendo que no sea necesario mucho entrenamiento para que los usuarios puedan utilizar el módulo (42).



# Capítulo 2: Desarrollo de la solución

---

## RNF2 Fiabilidad:

- Ante el fallo de una funcionalidad del sistema, el resto de las funcionalidades que no dependen de esta deberán seguir funcionando.
- El sistema impondrá campos obligatorios para garantizar la integridad de la información que se introduce por el usuario y no permitirá la entrada de datos incorrectos (42).

## RNF3 Mantenibilidad

La codificación del sistema será estándar y las funcionalidades serán comentadas. Además, el sistema debe permitir adicionar o modificar funcionalidades (42).

## RNF4 Portabilidad

**RNF4.1 Adaptabilidad:** El sistema deberá adaptarse al entorno operativo siempre y cuando este cumpla con las características de la aplicación y debe coexistir con otros sistemas sin dificultad.

## RNF5 Rendimiento

**RNF5.1: Comportamiento en el tiempo:** El sistema en las operaciones de modificación o búsqueda de datos, deberá tener respuestas a peticiones del cliente en un tiempo máximo de 5 segundos (esta cifra no incluye los retardos por concepto de tráfico de red) (42).

## RNF6 Hardware

**RNF6.1: Estaciones de trabajo:** Las estaciones de trabajo deberán poseer como mínimo un procesador Pentium IV a 2.20 GHZ, una RAM de 1GB y una tarjeta de red.

**RNF6.2: Servidor de aplicaciones:** El servidor de aplicaciones deberá poseer como mínimo un Core 2 Duo a 2.33 GHz, una RAM de 4 GB, disco duro 250 GB, una tarjeta de red y una fuente de alimentación de corriente.

## RNF7 Software

**RNF7.1: Navegador web:** Las estaciones de trabajo deberán poseer el navegador web Mozilla Firefox 34.0 o una versión superior y el sistema operativo Linux/Windows.

**RNF7.2: Servidor de aplicaciones web:** El servidor de aplicaciones web deberá ser Apache 2.0 y el sistema operativo Linux.

## *Capítulo 2: Desarrollo de la solución*

---

### **2.4. Validación de los requisitos**

La validación de los requisitos examina que estos se hayan establecido de manera precisa; que se han detectado las inconsistencias, omisiones y errores, han sido corregidos y que los productos de trabajo cumplen con los estándares establecidos para el proceso, proyecto y producto (40).

#### **Técnicas de validación de requisitos**

Las revisiones del software son un “filtro” para el proceso del software. Es decir, se aplican en varios puntos durante la ingeniería de software y sirven para descubrir errores y defectos a fin de poder eliminarlos. Las revisiones del software “purifican” los productos del trabajo de la ingeniería de software, incluso los modelos de requisitos y diseño, código y datos de prueba (43).

Existen varias técnicas para la validación de requisitos de software como: Técnica de prototipado y Diseño de casos de prueba.

**Prototipado:** son la versión inicial de un sistema de software que se utiliza para demostrar los conceptos, probar las opciones de diseño y entender mejor el problema y su solución, permiten revelar errores u omisiones en los requisitos propuestos (43).

**Diseño de casos de prueba (DCP):** el objetivo principal del diseño de casos de prueba es obtener un conjunto de pruebas que tengan la mayor probabilidad de descubrir los defectos del software (40).

Se validaron con el cliente los prototipos de interfaz de usuario y diseños de casos de pruebas definidos para los requisitos funcionales, comprobándose la coherencia con las necesidades expuestas. Las validaciones de los requisitos utilizando las técnicas mencionadas permitieron que el cliente tuviera una visión real de la propuesta de solución, lo cual permitió realizar las correcciones necesarias durante el proceso de análisis, diseño e implementación de las funcionalidades del sistema.

### **2.5. Diseño del sistema**

Una vez que se analizan y especifican los requisitos, se realiza el diseño del software. Es un proceso iterativo mediante el cual los requisitos se traducen en un plano para construir el software (40). El diseño del sistema debe estar conforme a los requisitos funcionales y no funcionales definidos del sistema.

# Capítulo 2: Desarrollo de la solución

## Arquitectura del sistema

La arquitectura es la estructura u organización de los componentes del programa (módulos), la manera en que estos componentes interactúan, y la estructura de datos que utilizan los componentes (40). EL modelo de arquitectura de sistema a utilizar en el desarrollo del sistema de reportes es el modelo-vista-controlador (MVC).

## Patrón MVC

**Vista:** Presenta el modelo en un formato adecuado para interactuar (usualmente la interfaz de usuario) por tanto dicho modelo la información que debe representar como salida (44).

**Modelo:** Envía a la vista aquella parte de la información que en cada momento se le solicita para que sea mostrada. Las peticiones de acceso o manipulación de información llegan al modelo a través del controlador (44).

**Controlador:** Responde a eventos e invoca peticiones al modelo cuando se hace alguna solicitud sobre la información. El controlador hace de intermediario entre la vista y el modelo (44).

EL patrón Modelo-Vista-Controlador fue aplicado de la siguiente manera como se expresa en la Figura 3 mostrada a continuación:

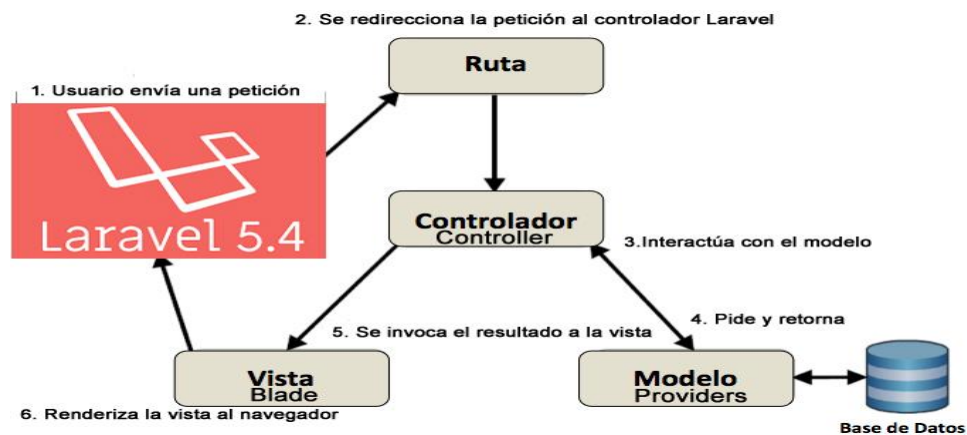


Figura 3: Patrón MVC en el sistema SRBin.

En la Figura 4 se representa la separación de las clases de la aplicación según el marco de trabajo Laravel 5.4. Se observa la utilización del patrón MVC, en la carpeta **Controller** se encuentran las

## Capítulo 2: Desarrollo de la solución

---

clases controladoras, en **Providers** las clases del modelo y en la **views** las clases que forman parte de la vista del sistema.

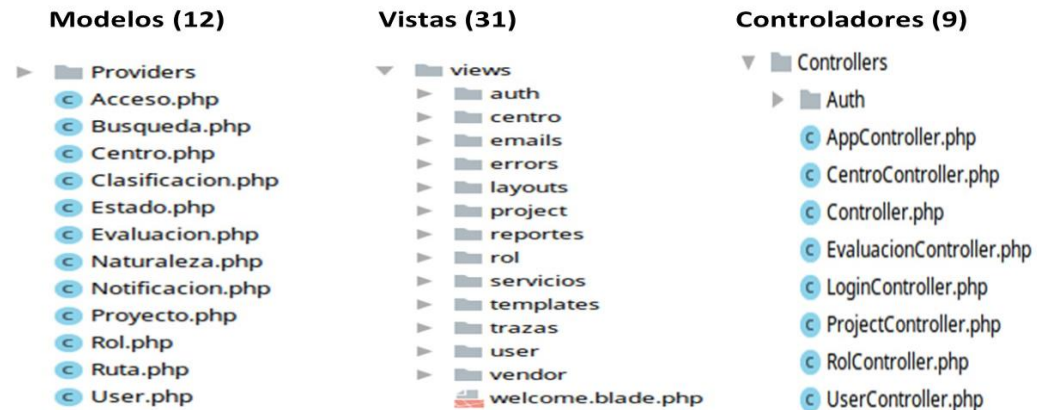


Figura 4: Aplicación del patrón MVC en el sistema SRBin. Fuente: elaboración propia.

### Diagrama de clases del diseño

El diseño es la forma exacta en la que un requisito se puede convertir en un sistema o producto de software terminado. Crea un modelo del software de forma detallada sobre la estructura de datos, la arquitectura, las interfaces y los componentes del software que son necesarios para implementar el sistema e influye directamente en la calidad del producto informático (42).

## Capítulo 2: Desarrollo de la solución

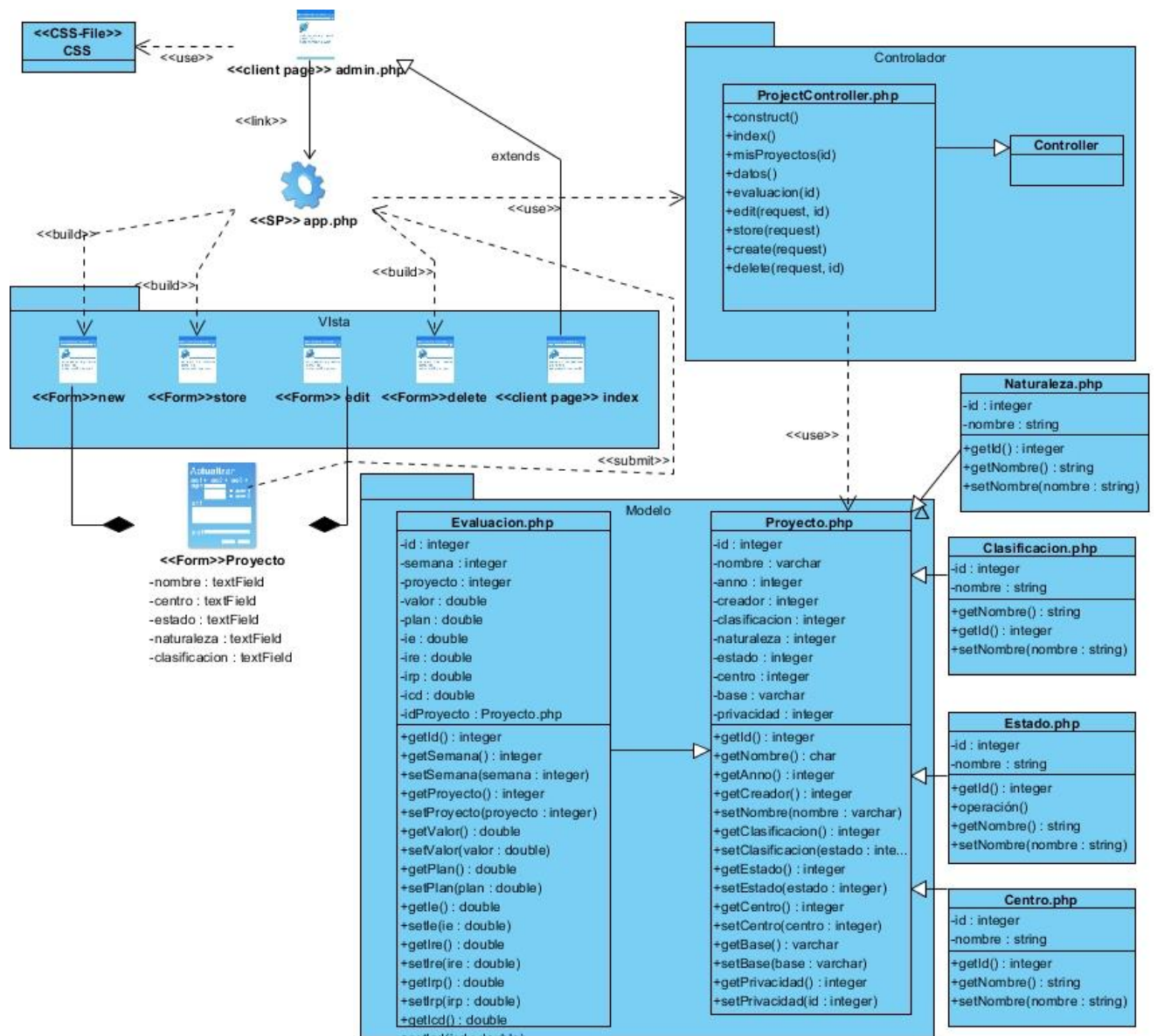


Figura 5: Diagrama de clases de diseño de la funcionalidad Gestionar los proyectos.

El diagrama de clases del diseño que muestra la Figura 5, contiene las distintas clases del sistema y las relaciones existentes entre ellas, para el escenario *Gestionar los proyectos*. Para un mejor entendimiento de las clases ver el **Modelo de datos**, además el **Anexo 12: Diagrama de clases**

## Capítulo 2: Desarrollo de la solución

---

del gestionar evaluaciones del proyecto. muestra otra de las funcionalidades del sistema desarrollado a un mayor nivel de detalle.

- *Clases de interfaz*: definen todas las abstracciones necesarias para la interacción entre los usuarios y la aplicación. Están formadas por las páginas cliente: *create.php*, *store.php*, *edit.php*, *delete.php*, *index.php* y *admin.php*.
- *Clase controladora*: permiten encapsular las funcionalidades necesarias para interactuar con las clases interfaces, las clases entidades y de acceso a los datos. Agrupa la lógica del negocio ya que responde a las peticiones HTTP provenientes del cliente e invoca peticiones a la base de datos. En este caso es la clase *ProjectController.php*
- *Clase persistente*: representa el almacenamiento de datos que persistirá más allá de la ejecución del sistema. Está representada por la clase del modelo: *Proyecto.php* y *Evaluacion.php*.

### Patrones de diseño de software

Un conjunto de patrones arquitectónicos permite que un ingeniero de software reutilice conceptos en el nivel de diseño. Un patrón de diseño describe una estructura de diseño que resuelve un problema de diseño particular dentro de un contexto específico y en medio de fuerzas que pueden tener un impacto en la manera en que se aplica y utiliza el patrón (40). En el diseño del sistema se utilizaron los patrones GRASP.

- **Experto**: plantea que las responsabilidades se deben asignar al objeto experto en información.
- **Creador**: consiste en determinar el objeto que debe ser responsable de crear una nueva instancia de la clase.
- **Controlador**: asigna la responsabilidad de gestionar mensajes en el sistema a una clase que represente al sistema global, a un subsistema, manejador de sesión o de eventos de un escenario de caso de uso.
- **Alta cohesión**: asegura la necesidad de mantener el sistema con un bajo nivel de complejidad, es decir, una clase tiene una responsabilidad moderada en un área funcional y colabora con otras clases para llevar a cabo las tareas.

## Capítulo 2: Desarrollo de la solución

- **Bajo acoplamiento:** permite mantener bajas dependencias, bajo impacto al cambio e incrementar la reutilización (42).

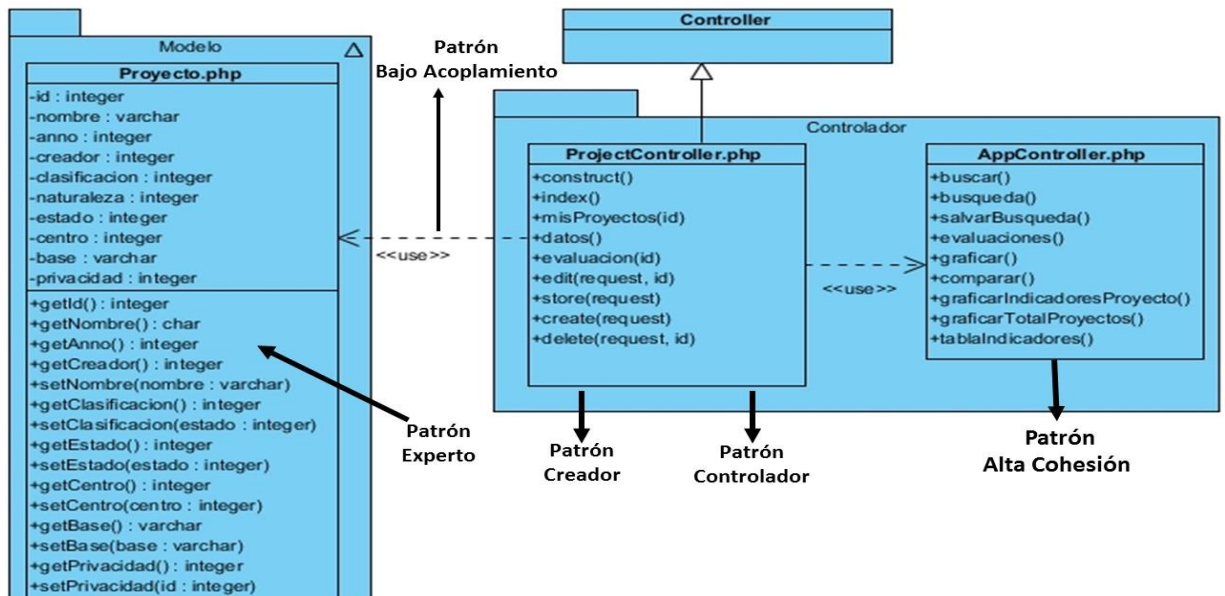


Figura 6: Patrón GRASP aplicado a la clase de diseño.

**Experto:** se evidencia en la clase **Proyecto.php**, este contiene las funcionalidades necesarias para acceder a la información de un servicio.

**Creador:** se utiliza en la clase **ProjectController.php**, es la clase que posee las funcionalidades necesarias para la creación de los servicios.

**Controlador:** se ve reflejado en la clase **ProjectController.php**, controla el flujo de información de todo el componente.

**Alta cohesión:** se refleja en todo el diseño, cada clase tiene la responsabilidad de un área funcional y colabora con otras clases para realizar las tareas asociadas. Ejemplo del patrón es la clase **AppController.php**.

**Bajo acoplamiento:** se evidencia en el diseño de las clases de la propuesta de solución, manteniendo bajas relaciones entre las clases, permitiendo bajo impacto al cambio y un incremento de la reutilización.

# Capítulo 2: Desarrollo de la solución

## Modelo de datos

El modelo de datos permite estructurar la base de datos, en cuanto a los tipos de datos presentes y la forma en que se relacionan entre sí (45). Representa la información persistente en el sistema, es decir, las 11 entidades con que cuenta la solución propuesta para el almacenamiento de la información.

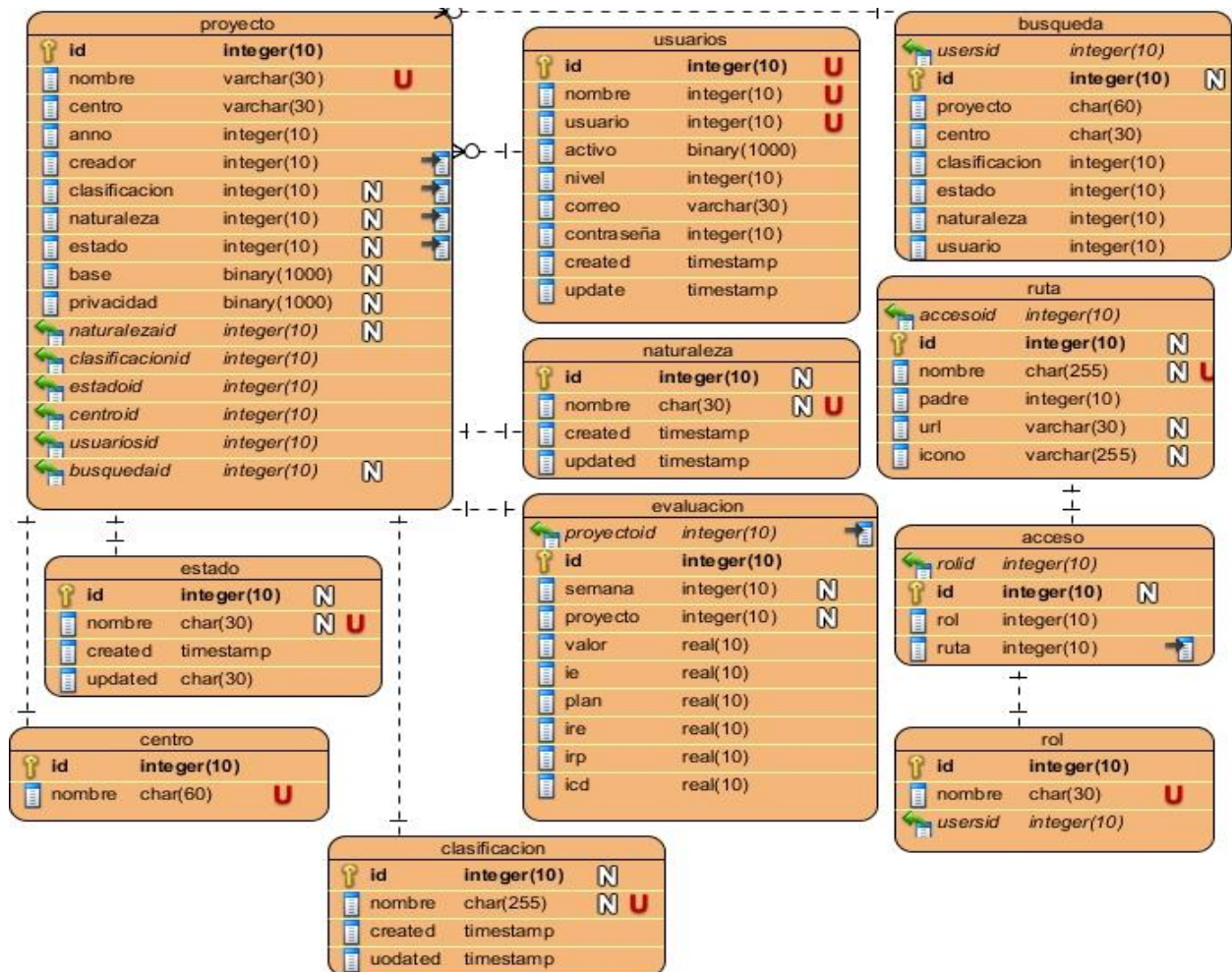


Figura 7: Modelo de datos de la solución propuesta. Fuente: elaboración propia.

## Métricas para validación del diseño

Para la validación del diseño se aplicaron las métricas **Tamaño operacional de clase** y **Relaciones entre clases** debido al conjunto de atributos de calidad de diseño que ambos miden.



## Capítulo 2: Desarrollo de la solución

### Tamaño operacional de clase (TOC)

La métrica del *Tamaño Operacional* se refiere al número de métodos pertenecientes a una clase. Está determinada por los atributos: *Responsabilidad*, *Complejidad de implementación* y *Reutilización*, existiendo una relación directa con los dos primeros e inversa con el último antes mencionado.

Atributo de calidad	Modo en que lo afecta
<b>Responsabilidad</b>	Un aumento del TOC implica un aumento de la responsabilidad asignada a la clase.
<b>Complejidad de implementación</b>	Un aumento del TOC implica un aumento de la complejidad de implementación de la clase.
<b>Reutilización</b>	Un aumento del TOC implica una disminución en el grado de reutilización de la clase.

Tabla 3: Tamaño operacional de clase (TOC).

Atributos	Categoría	Criterio
<b>Responsabilidad</b>	Baja	$\leq$ Prom. (PO)
	Media	Entre Prom. Y 2* Prom
	Alta	$>$ 2* Prom
<b>Complejidad de implementación</b>	Baja	$\leq$ Prom
	Media	Entre Prom. y 2* Prom
	Alta	$>$ 2* Prom
<b>Reutilización</b>	Baja	$>$ 2*Prom
	Media	Entre Prom. y 2* Prom
	Alta	$\leq$ Prom

Tabla 4: Rango de valores para evaluar la técnica de atributos de calidad de la métrica TOC.

No	Subsistema	Clase	Cantidad de
1	SRBin	AppController	12
2	SRBin	Busqueda	3
3	SRBin	Controller	1
4	SRBin	Centro	1
5	SRBin	EvaluacionController	7
6	SRBin	Evaluacion	1

## Capítulo 2: Desarrollo de la solución

7	SRBin	LoginController	1
8	SRBin	Acceso	1
9	SRBin	Ruta	0
10	SRBin	Rol	1
11	SRBin	ProjectController	10
12	SRBin	Proyecto	5
13	SRBin	Naturaleza	1
14	SRBin	Estado	1
15	SRBin	Clasificacion	1
16	SRBin	RolController	7
17	SRBin	Rol	1
18	SRBin	UserController	10
19	SRBin	User	6

Tabla 5: Tamaño de las clases del sistema.

En la Figura 8 se aprecia la incidencia de los resultados para el atributo *Responsabilidad* según la evaluación de la métrica TOC.

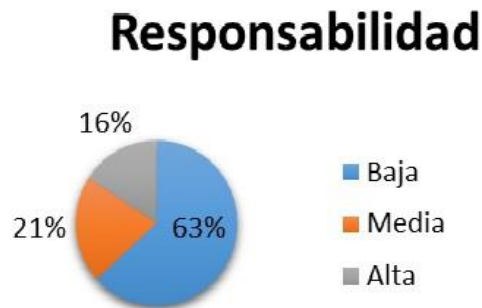


Figura 8: Resultado del atributo Responsabilidad.

En la Figura 9 se observa la incidencia de los resultados para el atributo *Complejidad de implementación* según la evaluación de la métrica TOC.

## Capítulo 2: Desarrollo de la solución

---

### Complejidad

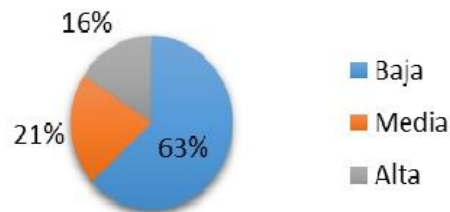


Figura 9: Resultado del atributo Complejidad de implementación.

En la Figura 10 se muestra la incidencia de los resultados para el atributo Reutilización según la evaluación de la métrica TOC.

### Reutilización

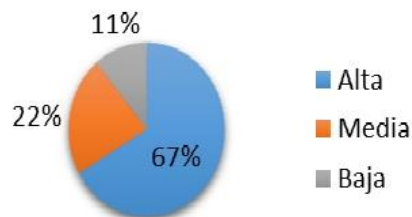


Figura 10: Resultado del atributo Reutilización.

Como resultado de la aplicación de la métrica TOC se puede apreciar de manera general que el 63% de las clases del diseño tienen baja responsabilidad. Esto significa que cuentan con menor grado de complejidad de implementación, lo que favorece notablemente la reutilización de las clases. Mientras que el 37% restante, presenta una responsabilidad y complejidad algorítmica alta, disminuyendo la posibilidad de ser reutilizables.

## Capítulo 2: Desarrollo de la solución

### Relaciones entre clases (RC)

Está determinada por el número de relaciones de uso de una clase y por los atributos: *Acoplamiento*, *Complejidad de mantenimiento*, *Cantidad de pruebas* y *Reutilización*, existiendo una relación directa con los tres primeros e inversa con el último antes mencionado.

Atributo	Modo en que lo afecta
<b>Acoplamiento</b>	Un aumento del RC implica un aumento del Acoplamiento de la clase.
<b>Complejidad de mantenimiento</b>	Un aumento del RC implica un aumento de la complejidad del mantenimiento de la clase.
<b>Reutilización</b>	Un aumento del RC implica una disminución en el grado de reutilización de la clase.
<b>Cantidad de pruebas</b>	Un aumento del RC implica un aumento de la Cantidad de pruebas de unidad necesarias para probar una clase.

Tabla 6 Atributos de calidad evaluados por la métrica RC.

Atributo	Categoría	Criterio
<b>Acoplamiento</b>	Ninguno	0
	Bajo	1
	Medio	2
	Alto	>2
<b>Complejidad de mantenimiento</b>	Baja	$\leq$ Prom.
	Media	Entre Prom y $2^*$ Prom.
	Alta	$> 2^*$ Prom.
<b>Reutilización</b>	Baja	$> 2^*$ Prom.
	Media	Entre Prom y $2^*$ Prom
	Alta	$\leq$ Prom.
<b>Cantidad de pruebas</b>	Baja	$\leq$ Prom.
	Media	Entre Prom y $2^*$ Prom.
	Alta	$> 2^*$ Prom.

Tabla 7: Criterios de evaluación para la métrica RC.

## Capítulo 2: Desarrollo de la solución

---

### Resultados del instrumento de evaluación de la métrica Relaciones entre Clases (RC).

Representación en por ciento de los resultados obtenidos en el instrumento agrupados en los intervalos definidos, ver Figura 11:

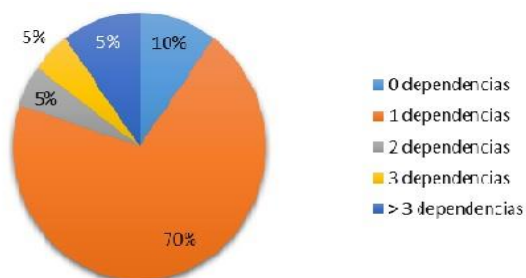


Figura 11: Por ciento de los resultados obtenidos.

En la Figura 12 se puede apreciar la incidencia de los resultados para el atributo Acoplamiento según la evaluación de la métrica TOC.



Figura 12: Resultado del atributo Acoplamiento.

En la Figura 13 se observa la incidencia de los resultados para el atributo Complejidad de mantenimiento según la evaluación de la métrica TOC.



Figura 13: Resultado del atributo Complejidad de mantenimiento.

## Capítulo 2: Desarrollo de la solución

---

En la Figura 14 se muestra la incidencia de los resultados para el atributo Cantidad de pruebas según la evaluación de la métrica TOC.

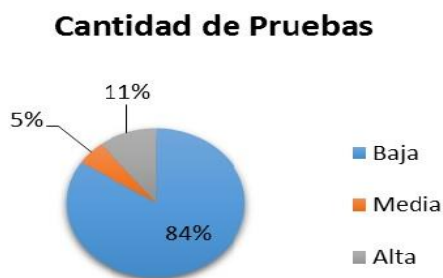


Figura 14: Resultado del atributo Cantidad de pruebas.

En la Figura 15 se evidencia la incidencia de los resultados para el atributo Reutilización según la evaluación de la métrica TOC.



Figura 15: Resultado del atributo Reutilización.

Al aplicar la métrica RC se obtuvo como resultado un 74% de bajo acoplamiento, un 84% de baja complejidad de mantenimiento, un 84% de baja cantidad de pruebas y un 84% de alta reutilización que poseen las clases diseñadas para el sistema. Sin embargo, el 16% restante presenta un alto acoplamiento, lo que provoca que la complejidad de mantenimiento y la cantidad de pruebas unitarias sea mayor, provocando que la reutilización sea menor con respecto a las clases.

La aplicación de las métricas TOC y RC demuestra que las clases del sistema poseen una baja carga de responsabilidades, un bajo acoplamiento y una alta reutilización, permitiendo una fácil implementación, una baja complejidad de mantenimiento y una baja cantidad de pruebas de

## Capítulo 2: Desarrollo de la solución

unidad necesarias para probar las clases. Los resultados obtenidos por la aplicación y evaluación de las métricas fueron positivos, quedando validado el diseño del sistema.

### 2.6. Implementación del sistema

#### Diagrama de componentes

Los diagramas de componentes se encuentran dentro de los diagramas definidos en UML. Un diagrama de componentes muestra un conjunto de componentes y sus relaciones de manera gráfica a través del uso de nodos y arcos entre estos. Visto de otro modo un diagrama de componentes puede ser un tipo especial de diagrama de clases que se centra en los componentes físicos del sistema (46). A continuación, una representación del diagrama de componentes del sistema SRBin.

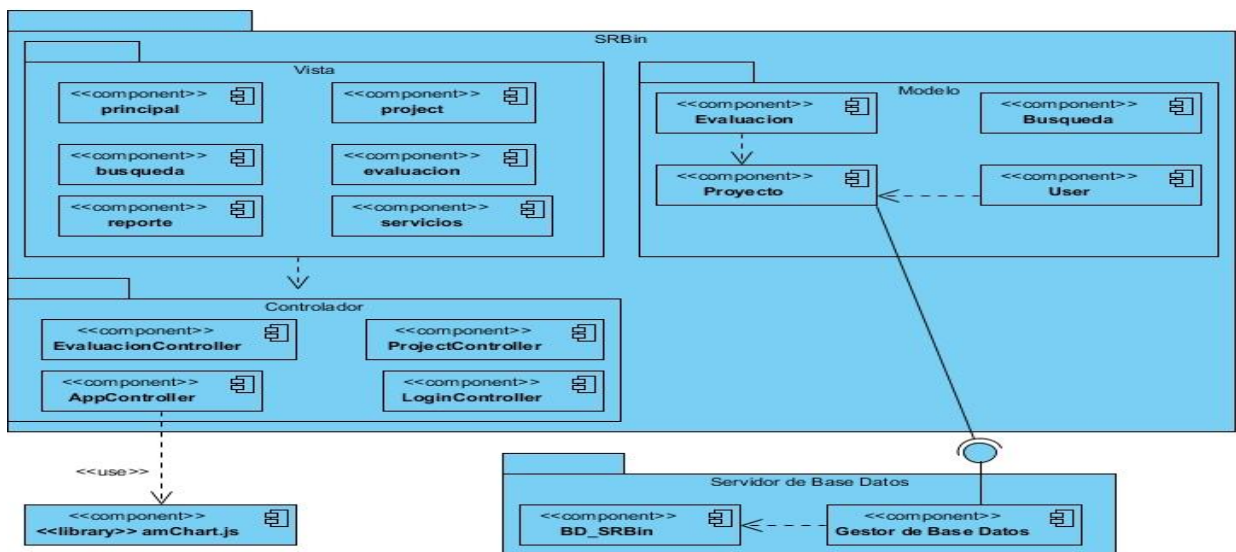


Figura 16: Diagrama de componentes.

Los componentes contenidos en el diagrama anterior son:

- En el paquete Vista contiene las páginas clientes que permiten la comunicación de los usuarios con el sistema.
- En el paquete Modelo contiene la información persistente de la aplicación.
- El paquete Controlador representa la clase controladora, que gestiona el flujo de información en el sistema.

## Capítulo 2: Desarrollo de la solución

- El componente *amChart* es una librería de JavaScript 1.6 para la generación de gráficos.
- El paquete Servidos de Base Datos contiene los componentes Gestor de Base Datos y el componente *BD\_SRBIn* que representa la base de datos del sistema.

El diagrama de componentes proporciona una visión física de la construcción del sistema. Muestra los componentes y sus relaciones. En la Figura 16 se representan los componentes del sistema siguiendo la arquitectura Modelo-Vista-Controlador y su relación con el resto de los componentes utilizados. Cada capa es representada como un paquete, agrupando la parte representativa de sus componentes internos y mostrando su relación con los servicios externos.

### Diagrama de despliegue

Los diagramas de despliegue son útiles para facilitar la comunicación entre los ingenieros de hardware y los de software. La división entre cliente y servidor en un sistema es complicada ya que implica tomar alguna decisión sobre dónde colocar físicamente sus componentes software, qué cantidad de software debe residir en el cliente, etc. Para modelar un sistema cliente/servidor se requiere:

- Identificar los nodos que representan los procesadores cliente y servidor del sistema.
- Destacar los dispositivos relacionados con el comportamiento del sistema (46).

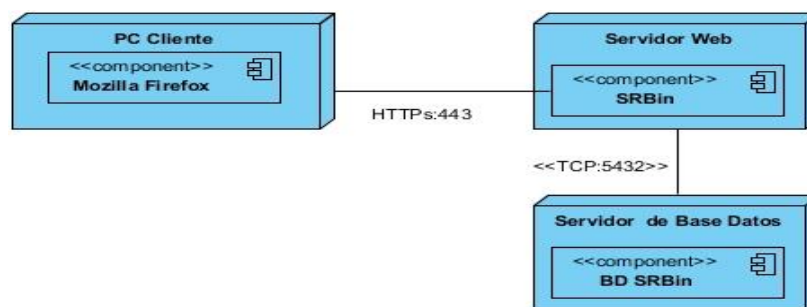


Figura 17: Diagrama de Despliegue del sistema SRBin.

### Estándares de codificación

Los estándares de codificación se pueden definir como “un conjunto de reglas, normas o patrones destinados a establecer uniformidad en el proceso de generación de un código. Son pautas de



## Capítulo 2: Desarrollo de la solución

programación enfocadas a la estructura y apariencia física del código para facilitar su lectura, comprensión y mantenimiento” (47).

**Notación de Pascal (del inglés *Pascal Case*):** este estilo de escritura se utiliza para escribir el nombre de los identificadores, variables o componente de un proyecto colocando en mayúscula la primera letra de cada palabra que forme el nombre del elemento (48). Esta notación fue empleada en la implementación de la aplicación para declarar el nombre de las clases.

**Notación de camello (del inglés *Camel Case*):** este estilo de notación se emplea para escribir la primera letra de la identificación con minúsculas, y la inicial de cada una de las palabras concatenadas se escribe con mayúscula. Existen dos tipos de *Camel Case*, el estilo *Upper Camel Case*, define que la primera letra de cada una de las palabras que identifican a un elemento de un proyecto es mayúscula y el *Lower Camel Case* cuando la primera letra es minúscula (48). El estilo de notación utilizado en la declaración de variables y métodos es ***Lower Camel Case***.

- Los nombres de las clases controladoras terminan con la palabra Controller.
- Las variables que poseen nombres de palabras compuestas se escriben juntas.
- Se hizo uso de los comentarios para la explicación del código siempre que fue necesario.

En la Figura 18, se presenta un fragmento de código de la clase ***AppController.php***. En ella se ejemplifica el uso de los estándares de codificación utilizados en la implementación de la solución.

```
class AppController extends Controller
{
    public function __construct(){...}
    /** muestra la lista de recursos ...*/
    public function index(){...}
    public function getData($id)
    {
        $result = array();
        $evaluaciones = Evaluacion::all()->groupBy('groupBy: 'week');
        $projects = Proyecto::find($id)->get();
        for ($i = 1; $i <= count($evaluaciones); $i++) {...}
        foreach ($projects as $p) {
            $result['graficos'][] = array(...);
        }
    }
    return $result;
}
```

Diagrama de anotaciones:

- Una flecha apunta desde el texto "Pascal Case" a la línea `class AppController extends Controller`.
- Una flecha apunta desde el texto "Comentarios para la explicación del código" a la línea de comentario `/** muestra la lista de recursos ...*/`.
- Una flecha apunta desde el texto "Lower Camel Case" a la línea `$result = array();`.

Figura 18: Ejemplo del uso de los estándares de codificación. Fuente: elaboración propia.

## Capítulo 2: Desarrollo de la solución

Definir los estándares de codificación a utilizar en la investigación permite tener un estilo único en la implementación de la solución, lo cual facilitará el estudio y comprensión del código haciendo que sea más fácil a la hora de realizar cambios o dar mantenimiento.

### Descripción de la implementación

En este epígrafe se explica el flujo de trabajo en el sistema SRBin a partir de la descripción de las funcionalidades implementadas que muestra la Figura 19:

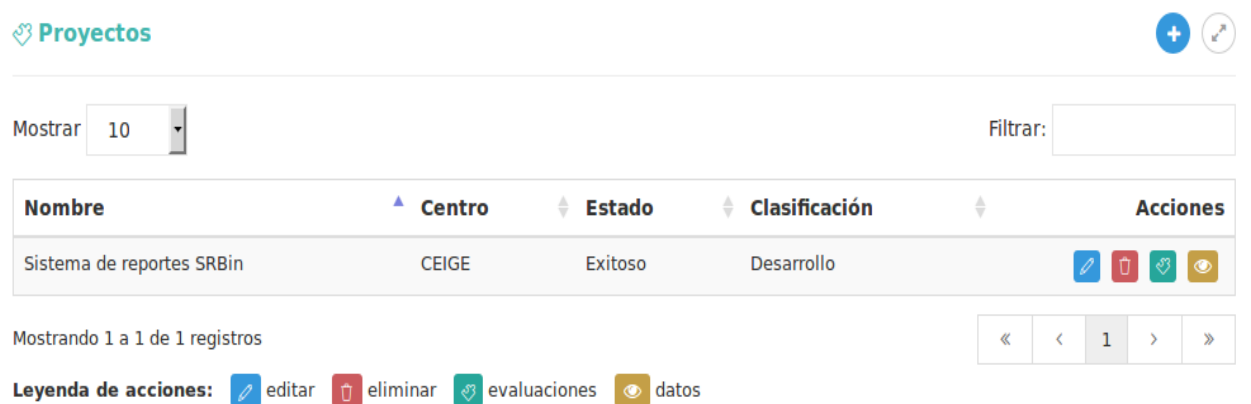


Figura 19. Interfaz del sistema SRBin sección Mis Proyectos.

- En esta área se pueden realizar 2 funcionalidades, de izquierda a derecha, el número 1 es Adicionar un proyecto, que consiste en crear un proyecto.
- El número 2 es un ícono para ampliar la vista en la pantalla conocido como fullscreen por su término en inglés.
- El número 3, a partir de un parámetro introducido se realiza una búsqueda dentro de los proyectos del usuario, es un filtro.
- En el área de las acciones a realizar sobre el proyecto, de izquierda a derecha el usuario tiene varias opciones. El número 4 representa la acción de editar los datos del proyecto (estado, naturaleza, etc.). El número 5, es la acción de eliminar un proyecto. El número 6 es ver las evaluaciones del proyecto seleccionado y una vez en esa vista permite que sean gestionadas las evaluaciones. El número 7 brinda la posibilidad de ver todos los datos del proyecto, lo cual incluye sus evaluaciones con sus respectivas semanas y los valores de

## Capítulo 2: Desarrollo de la solución

---

sus indicadores en esas semanas. Permite ver una gráfica con el estado del proyecto, y los valores de los indicadores del proyecto por semana en una gráfica.

- El número 8 en caso de existir más elementos que los listados por el usuario permite avanzar o retroceder páginas. Se aplica en todas las vistas del sistema con esas características.
- El número 9 permite seleccionar que cantidad de elementos desea listar en la página. Se aplica en todas las vistas del sistema con esas características.

En otras vistas del sistema, ejemplo en las evaluaciones de los proyectos contenidos en el sistema permite:

- Visualizar los datos del proyecto donde se obtienen todos los datos del proyecto, valores de los indicadores del proyecto por semana en una gráfica, y otra que muestra la evaluación por semana, además muestra una tabla con la evaluación semanal por los indicadores, señalando en colores el estado del indicador: Bien, Regular y Mal, corresponden a Verde, Amarillo y Rojo respectivamente.
- Realiza un reporte de las evaluaciones registradas del proyecto por cada semana, o sea permite ver los estados del proyecto por semana.
- Exporta todos los reportes en los formatos **.jpg, .png, .pdf, .svg, .csv, .json y .xlsx**.

A continuación, fragmento de código del método **reporteEvaluaciónProyecto()** que permite visualizar el gráfico con las evaluaciones de un proyecto por semana:

```
public function reporteEvaluacionProyecto($id){
    $result = array();
    $projects = Proyecto::find($id);
    $evaluaciones = $projects->Evaluacion()->groupBy('semana');
    for ($i = 1; $i <= count($evaluaciones); $i++) {
        $data = array();
        $data['semana'] = $i;
        foreach ($evaluaciones[$i] as $project) {
            $data[$project->getEval()->nombre] = $project->valor;
        }
        $result['datos'][] = $data;
    }
    $result['graficos'][] = array("balloonText" => "<span style='font-size:13px;'>[[title]]
    <b>en la semana</b> [[category]]:<b>[[value]]</b> [[additional]]</span>",
    "bullet" => "round", "dashLengthField" => "dashLengthLine", "lineThickness"
    => 3, "bulletSize" => 7, "bulletBorderAlpha" => 1, "bulletColor" => "#FFFFFF",
    "useLineColorForBulletBorder" => true, "bulletBorderThickness" => 3,
```

## Capítulo 2: Desarrollo de la solución

---

```
"fillAlphas" => 0, "lineAlpha" => 1, title" => $projects->nombre,  
"valueField" => $projects->nombre);  
return $result;  
}
```

Figura 20: Fragmento de código del método `reporteEvaluaciónProyecto()`.

### 2.7. Conclusiones parciales

En este capítulo se presentan las necesidades del negocio en correspondencia con los requisitos de software que deberá cumplir el sistema, obteniéndose 25 requisitos funcionales y 18 no funcionales. La utilización de patrones proporcionó una mayor calidad al diseño, este fue validado aplicando las métricas TOC y RC para comprobar de forma cuantitativa el grado en que el sistema cumple con los indicadores de calidad anteriormente descritos, como resultado se obtuvo que el diseño es sencillo y posee una calidad aceptable. Fueron implementados todos los requisitos planteados por el cliente, cumpliendo con los estándares de codificación establecidos.

# Capítulo 3: Validación de la solución

---

## CAPÍTULO 3: VALIDACIÓN DE LA SOLUCIÓN

### 3.1. Introducción

En el presente capítulo se realizará la validación para evaluar la propuesta de solución. Se verifica la calidad del resultado de la implementación, mediante los productos de trabajos generados durante las disciplinas de pruebas, exponiendo los resultados obtenidos.

### 3.2. Validación del software

La verificación y la validación del software (VyV) incluyen un conjunto de procedimientos, actividades, técnicas y herramientas que se utilizan paralelamente al desarrollo del software, para asegurar que el producto resuelve correctamente el problema para el que fuera diseñado. El objetivo es prevenir las fallas desde los requisitos hasta su implementación. Respecto de las pruebas a realizar en el software, ellas pueden ser dinámicas o estáticas, de acuerdo a si se realizan o no sobre el código. Entre las pruebas dinámicas, están las llamadas de caja blanca y las de caja negra, probando los procedimientos estructurales o las salidas (49).

#### Pruebas de caja blanca

La prueba de caja blanca, en ocasiones llamada prueba de caja de vidrio, es una filosofía de diseño de casos de prueba que usa la estructura de control descrita como parte del diseño a nivel de componentes para derivar casos de prueba. Al usar los métodos de prueba de caja blanca se revisan estructuras de datos internas para garantizar su validez (40).

#### Técnica camino básico

Dentro de las pruebas de caja blanca se encuentran la técnica de pruebas de *ruta básica* o *camino básico*, que es la seleccionada para verificar el código, o sea las estructuras y sus relaciones. Esta técnica fue aplicada a los métodos reporte Indicadores Proyecto, Obtener comparación de proyecto, Exportar reportes y Buscar. A continuación, se relacionan los pasos realizados durante la aplicación de la técnica del camino básico al método **reporteIndicadoresProyecto**.

1. Dibujar el grafo de flujo de la funcionalidad reporte Indicadores Proyecto.

```
public function reporteIndicadoresProyecto($id){  
    $result = array();  
    $projects = Proyecto::find($id);  
1 $evaluaciones = $projects->Evaluacion()->groupBy('semana');
```

## Capítulo 3: Validación de la solución

```
$indicadores = ['PLAN', 'IE', 'ICD', 'IRE', 'IRP'];
/*recorre el arreglo de evaluaciones del proyecto agrupadas por semana*/
2 for ($i = 1; $i <= count($evaluaciones); $i++) {
  /*se crea una variable para almacenar los datos de una semana*/
3  $data = array();
  $data['semana'] = $i;
  /*para cada una de la evaluaciones de una semana dada*/
4  foreach ($evaluaciones[$i] as $project) {
5    if ($project->proyecto == $id) {
      /*almacena los valores de cada uno de los indicadores del proyecto*/
6      $data[$indicadores[0]] = $project->indicador_PLAN;
      $data[$indicadores[1]] = $project->indicador_IE;
      $data[$indicadores[2]] = $project->indicador_ICD;
      $data[$indicadores[3]] = $project->indicador_IRE;
      $data[$indicadores[4]] = $project->indicador_IRP;
7,8    }
9  } /*almacena la evaluación de una semana al resto*/
10 $result['datos'][] = $data;
11 } /*para cada uno de los indicadores analizados*/
12 foreach ($indicadores as $p) {
13 /*crea un gráfico para representar dicha información*/
    $result['graficos'][] = array("balloonText" => "<span style='font-size:13px;'><b>[[title]]</b> en la
    <b>semana [[category]]:</b><b style='color:#5d2d77'>[[value]]</b></span>", "title" => $p, "type" =>
    "column",
    "fillAlphas" => 0.8, "valueField" => $p);
14 }
15 return $result;
}
```

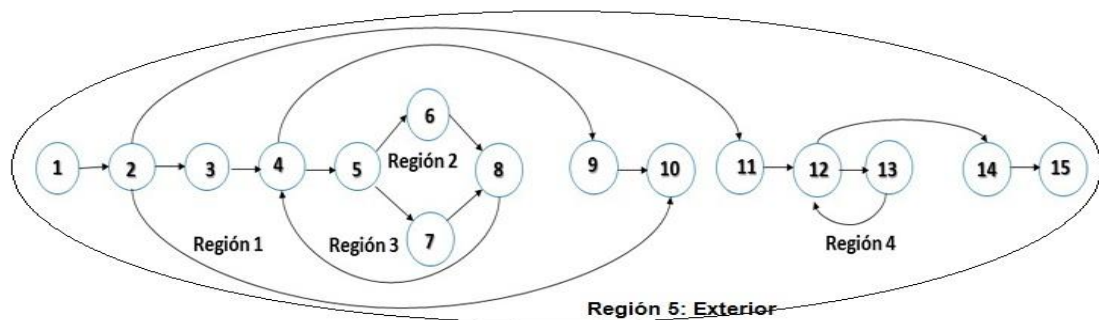


Figura 21: Representa el grafo de flujo obtenido del código representado anteriormente.

2. Determinar la complejidad ciclomática del grafo de flujo resultante.

La complejidad ciclomática del grafo  $V(G)$  se puede calcular de las tres formas siguientes:

$$V(G) = A - N + 2$$

donde: A es el número de aristas del grafo de flujo

$$V(G) = 18 - 15 + 2 = 5$$

N es el número de nodos del grafo

## *Capítulo 3: Validación de la solución*

---

$$V(G) = P + 1$$

donde: P es el número de nodos predicados (nodos con más de una arista de salida) contenidos en el grafo.

$$V(G) = 4 + 1 = 5$$

$$\text{Regiones} = V(G)$$

donde: Regiones son las áreas delimitadas por nodos y aristas en el grafo.

$$5 = 5$$

3. Determinar el conjunto básico de caminos linealmente independientes.

- Camino básico 1: 1-2-3-4-5-6-8-4-9-10-2-11-12-13-12-14-15
- Camino básico 2: 1-2-3-4-5-7-8-4-9-10-11-12-13-12-14-15
- Camino básico 3: 1-2-3-4-5-6-8-4-9-10-2-11-12-14-15
- Camino básico 4: 1-2-3-4-5-7-8-4-9-10-11-12-14-15
- Camino básico 5: 1-2-11-12-13-12-14-15

4. Definir los casos de prueba para comprobar la ejecución de cada camino del conjunto básico.

En el diseño de los casos de prueba se debe especificar los siguientes elementos:

- Descripción: contiene una descripción sobre las restricciones de los datos de entrada que debe tener el caso de prueba.
- Condición de ejecución: se especifican los parámetros que debe poseer el caso de prueba para que se cumpla una condición deseada como respuesta del funcionamiento del procedimiento.
- Entrada: se muestran los parámetros de entrada al procedimiento.
- Resultados esperados: se explica el resultado esperado de la ejecución del procedimiento.

<b>Diseño de caso de prueba para el camino 1</b>	
<b>Descripción</b>	El dato se obtiene a través del parámetro <i>id</i> .
<b>Condición de ejecución</b>	Campos válidos: El campo <b>Gráfico</b> permite visualizar la gráfica correspondiente al reporte de los indicadores del proyecto seleccionado. Para acceder al reporte el método realiza una búsqueda a través del <i>id</i> del proyecto seleccionado.  Si el <i>id</i> del proyecto seleccionado corresponde con el <i>id</i> de ese mismo proyecto almacenado y tienen datos los indicadores, entonces el sistema muestra el reporte.
<b>Entrada</b>	Selección de la opción para ver el <b>Gráfico</b> de las evaluaciones de un proyecto.

## Capítulo 3: Validación de la solución

<b>Resultados esperados</b>	Obtener un reporte de las evaluaciones por cada semana de un proyecto seleccionado.
<b>Respuesta del sistema</b>	Respuesta satisfactoria.

Tabla 8: Diseño de caso de prueba para el camino 1.

Con la realización de las pruebas de caja blanca al método **reporteIndicadoresProyecto ()**, se obtuvieron los resultados esperados, ya que se comprobó que la sentencia correspondiente al camino básico 1 es acertado, cumpliéndose así las condiciones de la prueba. De igual forma se probó el cumplimiento de los caminos restantes de acuerdo a los casos de pruebas reflejados en los **Anexo 13 – Anexo 16**.

### Pruebas de caja negra

Las pruebas de caja negra, también llamadas pruebas de comportamiento, se enfocan en los requisitos funcionales del software; es decir, las técnicas de prueba de caja negra le permiten derivar conjuntos de condiciones de entrada que revisarán por completo todos los requisitos funcionales para un programa. Las pruebas de caja negra intentan encontrar errores en las categorías siguientes: 1) funciones incorrectas o faltantes, 2) errores de interfaz, 3) errores en las estructuras de datos o en el acceso a bases de datos externas, 4) errores de comportamiento o rendimiento y 5) errores de inicialización y terminación (40).

### Método de partición de equivalencia

Dentro de las pruebas de caja negra se utiliza el método de partición de equivalencia es un método que divide el dominio de entrada de un programa en clases de datos de los que pueden derivarse casos de prueba. Se basa en una evaluación de las clases de equivalencia para una condición de entrada. Una clase de equivalencia representa un conjunto de estados válidos o inválidos para condiciones de entrada (40).

A continuación, el diseño del caso de prueba realizado para la validación del requisito funcional **Graficar**, el resto de estos serán mostrados en los **Anexo 17 – Anexo 20**.



## Capítulo 3: Validación de la solución

### Descripción del caso de prueba *Graficar*.

El requisito funcional permite graficar los datos históricos según el estado, clasificación y naturaleza de los proyectos que determine el usuario para visualizar.

### Condiciones de ejecución

1. El usuario debe estar autenticado en el sistema.
2. El usuario debe seleccionar la opción "Reportes".
3. El usuario selecciona la opción "Graficar".

Escenario	Descripción	Agrupar por	Buscar por	Respuesta del sistema	Flujo central
<b>EC 1.1 Graficar</b>	Permite graficar por elementos de proyectos.	V(Estado)	V(Gráfico de barras, clasificación, Naturaleza del proyecto)	Muestra el gráfico correspondiente a lo seleccionado para agrupar y buscar.	<ol style="list-style-type: none"> <li>1. Seleccionar la opción "<b>Reportes</b>".</li> <li>2. Se especifica en los campos de selección correspondientes a las gráficas que el usuario que se desea buscar.</li> <li>3. Selecciona la opción de menú "<b>Graficar</b>".</li> <li>4. El sistema realiza la gráfica del reporte como lo seleccionó el usuario en el criterio de búsqueda especificado.</li> </ol>

Tabla 9: Diseño de caso de prueba de la funcionalidad *Graficar*.

Diseño de caso de prueba	Caso Válido	Caso No Válido
<b>EC 1.1 Graficar</b>	El sistema responde realizando	El sistema no muestra la

## Capítulo 3: Validación de la solución

	la gráfica	gráfica
--	------------	---------

Tabla 10: Casos Válidos y No Válidos del EC 1.1 Graficar

Al poner en práctica el método de Partición de equivalencia perteneciente a las Pruebas de Caja Negra, los resultados obtenidos luego de aplicar los DCP, fueron satisfactorios para los casos válidos y no válidos especificados. Demostrando la consistencia del sistema desarrollado a partir de los requisitos descritos.

### Pruebas internas

Se ejecutaron las Pruebas internas, definidas como disciplina en la metodología que guía la investigación. Fueron ejecutadas por el grupo de calidad del Centro CEIGE, apoyándose en los Diseños de casos de prueba definidos para cada requisito funcional del sistema. Se realizaron 2 iteraciones para detectar las no conformidades, resolviéndose en su totalidad. Al finalizar las pruebas fue emitida un Acta de liberación del sistema (**Anexo 21**). A continuación, en la Figura 21 se muestran los resultados de las pruebas por las iteraciones realizadas y la cantidad de *No conformidades* encontradas, por el tipo y las resueltas.

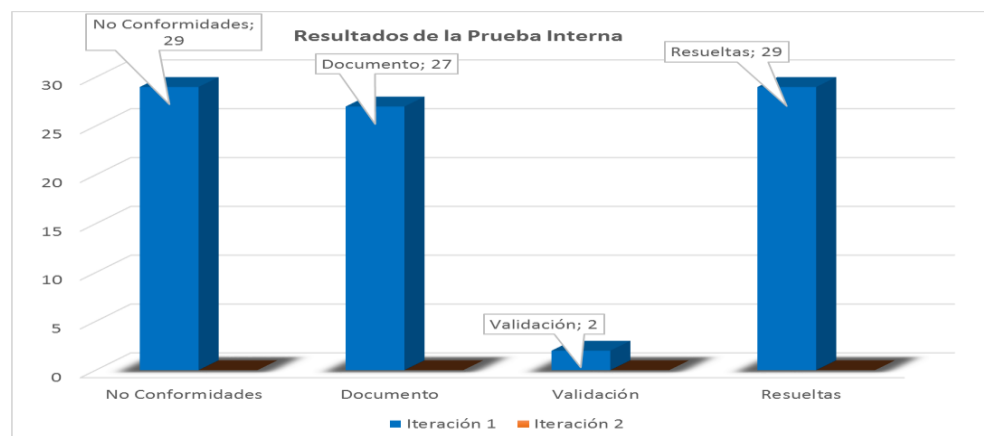


Figura 22: Resultados de la prueba interna por iteraciones realizadas.

### Pruebas de aceptación

Las pruebas de aceptación se realizan para que el cliente certifique que el sistema es válido para él. La planificación detallada de estas pruebas debe haberse realizado en etapas tempranas del

## Capítulo 3: Validación de la solución

desarrollo del proyecto, con el objetivo de utilizar los resultados como indicador de su validez: si se ejecutan las pruebas documentadas a satisfacción del cliente, el producto se considera correcto, y por tanto, adecuado para su puesta en producción (50).

El cliente, luego de haber revisado los productos de trabajo entregados, así como la explotación del Sistema de Reportes SRBin, determina que está de acuerdo con el producto final mostrado, el cual fue desarrollado bajo los requisitos previamente definidos.

La aceptación del producto arrojó un total de 12 solicitudes de cambios en 3 Iteraciones, las cuales fueron resueltas. Estas solicitudes fueron solucionadas para el correcto funcionamiento del sistema. Una vez comprobada la implementación de los requisitos, el cliente emitió el Acta de aceptación con el cliente, la cual puede ser consultada en el **Anexo 22**.

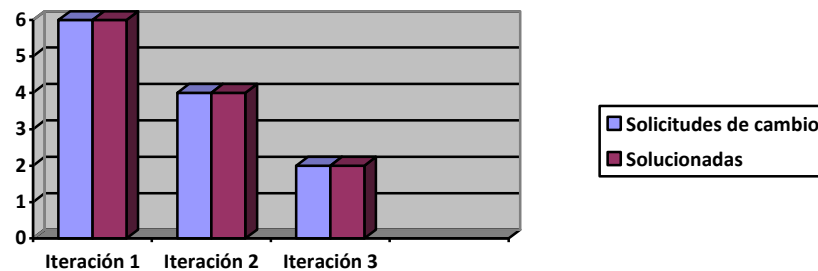


Figura 23: Resultados de la prueba de aceptación.

### 3.3. Validación científica aplicando el Cuasiexperimento de tipo preprueba/posprueba

De los centros encuestados, expuesto en el capítulo 1, se decide seleccionar del Centro VERTEX, el proyecto “Realidad Aumentada”, por tener establecido un alto compromiso de prioridad e importancia con un cliente nacional y además encontrarse actualmente en estado de “Ejecución”. Este será el grupo de observación para ser comparado con proyectos en estado “Terminado” y que hayan sido *Exitosos*. Se escoge la comparación con un proyecto finalizado exitosamente, por ser el caso ideal recomendado que permitirá la obtención de mejores resultados para el apoyo a la toma de decisiones.

# Capítulo 3: Validación de la solución

Dado el escenario que se plantea, se utiliza el método científico Cuasiexperimento de tipo preprueba/posprueba, ya que se tomará un solo grupo como muestra donde será aplicada la solución. Este método se diagramará de la siguiente manera:

G    O<sub>1</sub>    X    O<sub>2</sub> (51)

**Donde:**

**G:** se refiere al grupo experimental o de control seleccionado y que estará constituido por el proyecto “Realidad Aumentada”.

**X:** corresponde a la aplicación de la propuesta como tratamiento o estímulo al grupo experimental.

**O<sub>1</sub>:** corresponde a la observación sobre el grupo experimental antes de la aplicación de la solución propuesta.

**O<sub>2</sub>:** corresponde a la observación sobre el grupo de control luego de aplicado el sistema desarrollado.

## Estado del proyecto Realidad Aumentada antes de aplicar la solución desarrollada

En la Figura 24 se observa el estado actual del proyecto Realidad Aumentada, el cual se toma como grupo experimental, evidenciando a partir de los indicadores mostrados que existe una baja calidad al ejecutar el proceso de planificación, lo que repercute directamente en el aumento del costo y el tiempo para realizar las tareas.

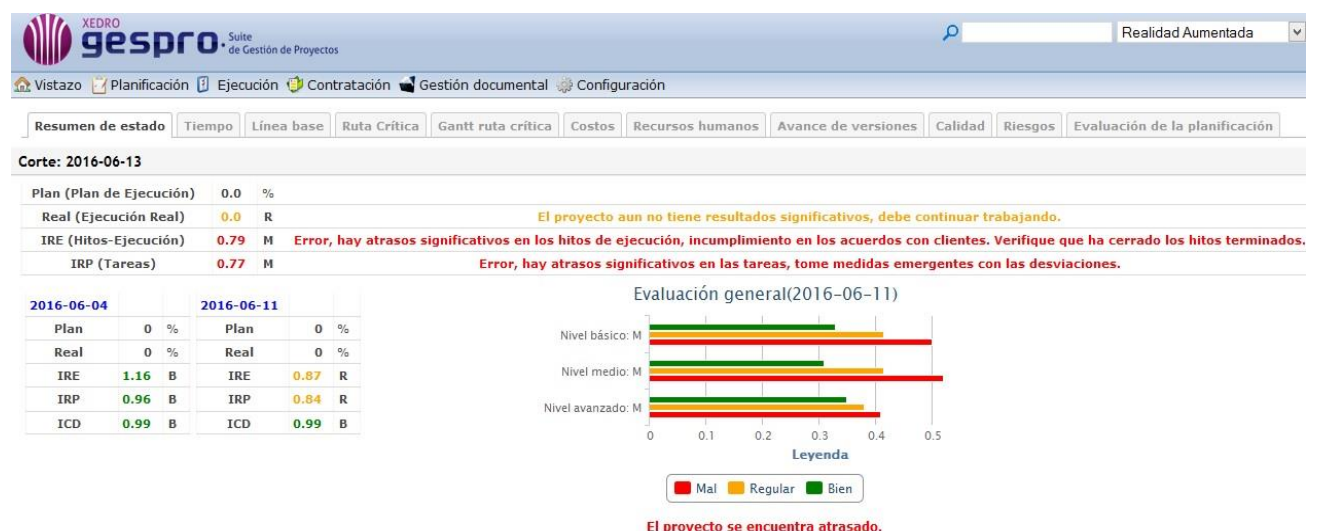


Figura 24: Observación 1 sobre el grupo experimental.

## Capítulo 3: Validación de la solución

### Resultados obtenidos a partir de aplicar el sistema SRBin al grupo de control

Una vez creado el proyecto Realidad Aumentada en el sistema SRBin con los datos correspondientes a los indicadores de la semana que se desea analizar, se compara con un proyecto *Exitoso* almacenado en la base de datos y que posea las mismas características en cuanto a: *Clasificación, Estado y Naturaleza*, en este caso el Proyecto Meteorix. Al realizar la comparación el sistema muestra una gráfica, donde la línea Azul representa el proyecto con el que se compara (Meteorix) y la Amarilla el proyecto que se examina (Realidad Aumentada). Cada punto contenido en las líneas, corresponde a los indicadores que se mencionan en el capítulo 1, para verificar cómo se comportaron en el tiempo. La Figura 25 muestra el análisis realizado por el sistema, permitiendo al usuario identificar los indicadores que inciden negativamente, para que de esta forma pueda tomar las decisiones necesarias y mejorar la evaluación del proyecto.

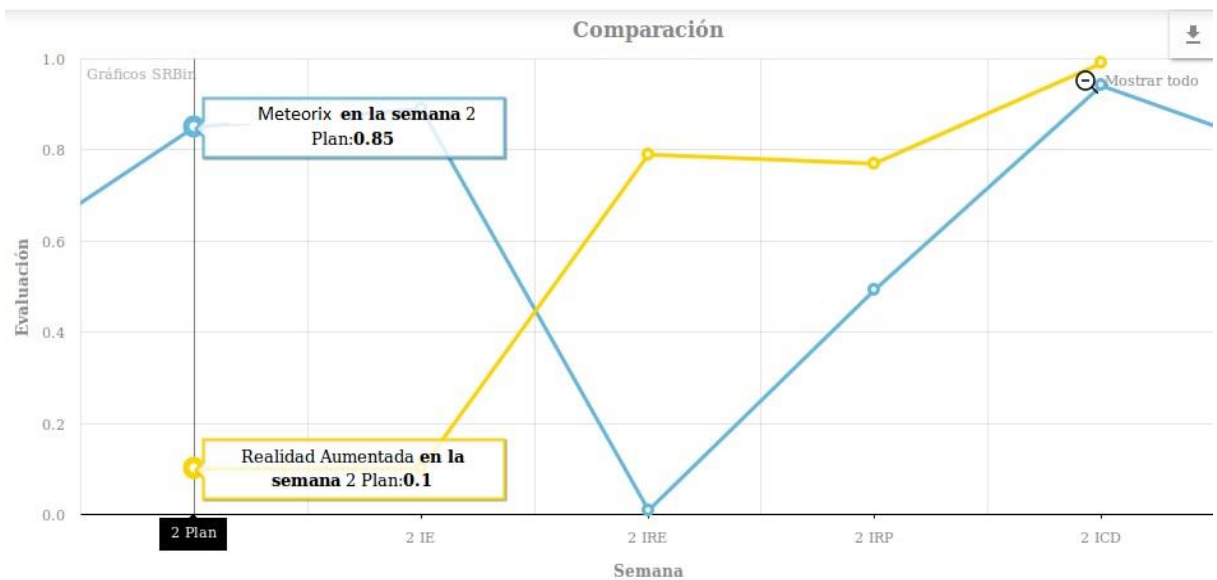


Figura 25: Observación 2 sobre el grupo experimental: Gráfica de comparación entre proyectos.

El sistema también permite graficar los proyectos, estableciendo las diferencias entre los indicadores de cada uno, para saber cuán lejos o cerca está de ser exitoso el proyecto que analiza el usuario. Si el valor mostrado es negativo, significa que este se encuentra “-X<sup>4</sup>” por debajo del

<sup>4</sup> Valor que representa que tan alejado se encuentra el proyecto de ser exitoso respecto a ese indicador.

## Capítulo 3: Validación de la solución

proyecto exitoso en el indicador “Y” de la semana que se evalúa. En caso de ser positivo el valor del indicador, es un resultado favorable para el proyecto. Para este análisis el proyecto Realidad Aumentada, el indicador **Plan** se encuentra con un valor negativo de **-0.75**, lo que indica que está incidiendo negativamente sobre la evaluación, las decisiones a tomar para este caso deben estar centradas en mejorar la planificación del proyecto para evitar desviaciones de la ruta crítica del cronograma, esto favorecerá la calidad del proceso y la disminución del costo en cuanto al tiempo de ejecución de los hitos de desarrollo del proyecto.

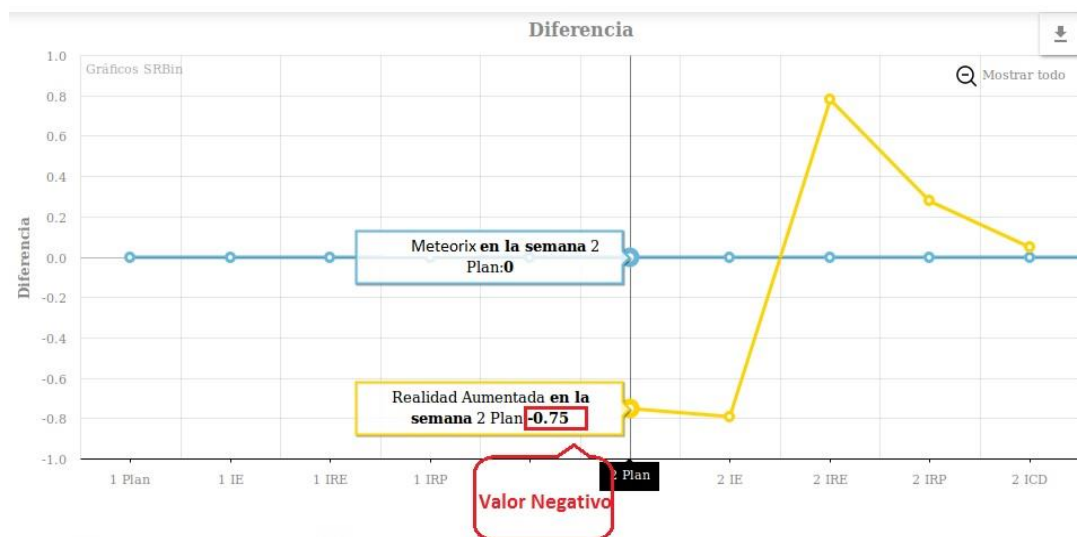


Figura 26: Observación 2 sobre el grupo experimental: Gráfica de diferencias entre los indicadores.

Al poner en práctica la experimentación preprueba/posprueba se realizó:

1. La captura manual de los indicadores del proyecto a analizar que proporciona la herramienta Gespro, para una primera observación (Figura 24).
2. Estos datos fueron analizados a través del sistema SRBin, lo cual arrojó los indicadores sobre los que se debe mejorar y en qué medida, para alcanzar el estado que se espera, obteniéndose una segunda observación (Figuras 25 y 26).

Una vez tomadas las acciones pertinentes sobre los indicadores que incidían de forma negativa en el estado del proyecto, se puede apreciar en la Figura 27 el cambio favorable en el que se encuentra el proyecto en cuestión.

# Capítulo 3: Validación de la solución

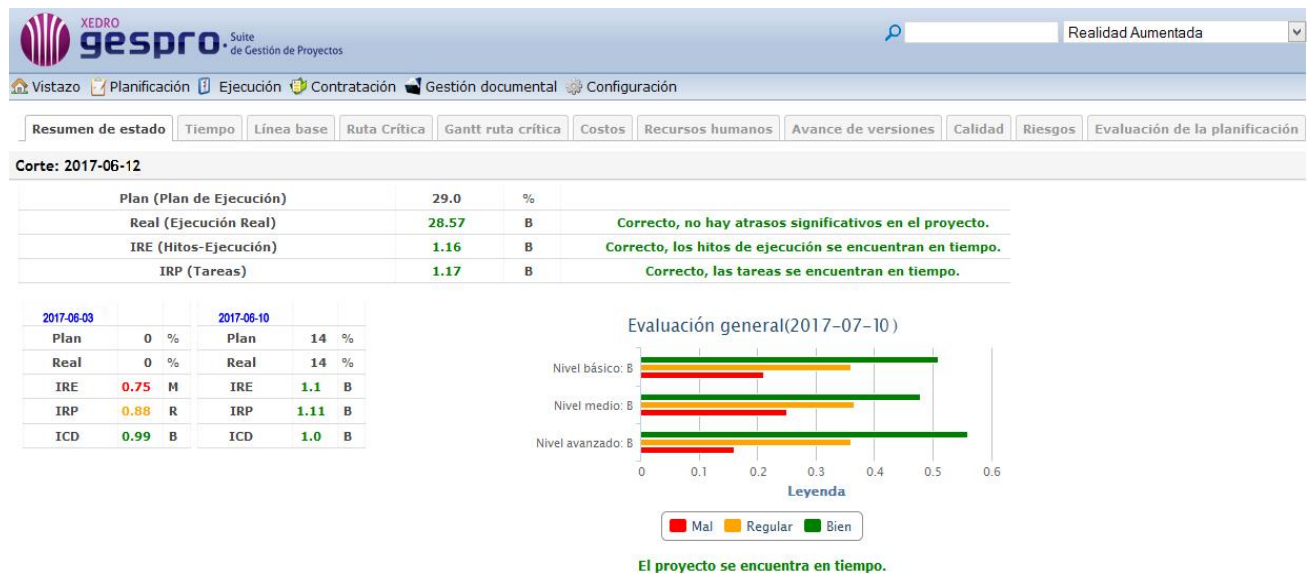


Figura 27: Indicadores del Gespro luego de aplicado el estímulo.

Finalmente, el método demostró la ventaja del uso del sistema SRBin para los proyectos en “Ejecución” de la UCI. A partir de los datos mostrados por la solución desarrollada, se pueden identificar los indicadores donde se debe centrar la atención para mitigar el riesgo de tener un proyecto fracasado. Se apoya la toma de decisiones sobre la planificación, asignación de recursos y el aprovechamiento del fondo de tiempo desde etapas más tempranas. Todo lo antes expuesto permitirá tomar acciones para contrarrestar la disminución de calidad al ejecutar el proceso, y el aumento del costo en cuanto a tiempo de los proyectos productivos de la UCI.

### 3.4. Conclusiones Parciales

En el presente capítulo se realizaron las pruebas de validación del Sistema de Reportes Basado en Indicadores (SRBin). La validación del sistema se realizó mediante pruebas internas, aplicándose la técnica del Camino básico para comprobar que el flujo de trabajo de las funcionalidades fuera el correcto, mientras que la técnica de Partición equivalente verificó en 2 iteraciones realizadas a partir de los casos de pruebas aplicados, que el sistema cumple con los requisitos definidos y aprobados. Se evidenció la satisfacción del cliente a través de los resultados arrojados durante las pruebas de aceptación. Con la puesta en práctica de la experimentación con un grupo de control y las observaciones realizadas preprueba/posprueba, se demuestra el resultado esperado.

# *Conclusiones Generales*

---

## **CONCLUSIONES GENERALES**

El análisis de los referentes bibliográficos sobre los sistemas de reportes, evidenció la necesidad de desarrollar un sistema informático para apoyar el proceso de toma de decisiones sobre la planificación, asignación de recursos y el aprovechamiento del fondo de tiempo de los proyectos de desarrollo de software de la UCI.

El diseño de la solución permitió modelar como debía ser implementado el sistema de reportes SRBin cumpliendo con los requisitos funcionales y no funcionales definidos.

Se obtuvo el sistema de reportes SRBin que permite la comparación de proyectos a partir de los indicadores definidos en la herramienta Gespro y realizar reportes estadísticos de datos históricos de los proyectos de la UCI, con lo que se cumple el objetivo general de la investigación planteado.

La validación del sistema se realizó a partir de la aplicación de técnicas, métricas y pruebas que garantizaron el correcto funcionamiento del sistema y demostraron la satisfacción del cliente con respecto a lo desarrollado.



# *Recomendaciones*

---

## **RECOMENDACIONES**

1. Implementar un servicio web que le permita al sistema SRBin, la obtención de los datos de los indicadores de gestión de proyecto, almacenados en la base de datos de proyectos terminados.
2. Implementar una funcionalidad aplicando técnicas de Inteligencia Artificial, para que el sistema SRBin muestre al usuario un conjunto alternativo de decisiones.

# Referencias Bibliográficas

---

## REFERENCIAS BIBLIOGRÁFICAS

1. FERREIRA, Sueli Mara Soares Pinto, DUDZIAK, Elisabeth Adriana y OTHERS. La alfabetización informacional para la ciudadanía en América Latina: el punto de vista del usuario final de programas nacionales de información y/o inclusión digital. In: *World library y information congress: 70th IFLA General Conference y Council*. IFLA, 2004.
2. BELTRÁN SANZ, Jaime, CARMONA CALVO, Miguel Angel, CARRASCO PÉREZ, Remigio, RIVAS ZAPATA, Miguel Angel y TEJEDOR PANCHÓN, Fernando. *Guía para una Gestión basada en procesos* [online]. 2009. Instituto Yaluz de Tecnología. [Accessed 3 November 2016]. Available from: <http://excelencia.iat.es/files/2012/08/2009.Gestión-basada-procesos-completa.pdf>
3. La informatización de la sociedad, una prioridad para Cuba › Cuba › Granma - Órgano oficial del PCC. [online]. [Accessed 3 November 2016]. Available from: <http://www.granma.cu/cuba/2014-12-12/la-informatización-de-la-sociedad-una-prioridad-para-cuba>
4. La introducción de resultados investigativos, un problema de actualidad en la formación del Ingeniero en Ciencias Informáticas. [online]. [Accessed 4 November 2016]. Available from: [http://scielo.sld.cu/scielo.php?pid=S2227-18992015000400011&script=sci\\_arttext&lng=pt](http://scielo.sld.cu/scielo.php?pid=S2227-18992015000400011&script=sci_arttext&lng=pt)
5. SOFTWARE ENGINEERING PROCESS MANAGEMENT PROGRAM. CMMI\_for\_Development\_v1.3\_Spanish. Mejora de los procesos para el desarrollo de mejores productos y servicios. [online]. November 2010. [Accessed 14 December 2016]. Available from: <https://excriba.prod.uci.cu/page/context/shared/document-details?nodeRef=workspace://SpacesStore/6f9b9296-e2a3-4722-8e0f-b747a34484bd>
6. SIECenter de México. [online]. [Accessed 14 December 2016]. Available from: <http://www.siecenter.com.mx/>
7. MARÍN-SÁNCHEZ\*, Jacqueline, LUGO-GARCÍA, José Alejandro, PIÑERO-PÉREZ, Pedro Yobanis, SANTIESTEBAN-GARCÍA, Alena María, ABELARDO-SANTANA, Félix Noel y MENÉNDEZ-RIZO, Javier. Proceso para la planificación y control de proyectos de software utilizando Xedro-GESPRO. *Revista Cubana de Ciencias Informáticas* [online]. June 2014. Vol. 8,

# Referencias Bibliográficas

---

- no.2, p.144–161. [Accessed 17 February 2017]. Available from: [http://scielo.sld.cu/scielo.php?script=sci\\_abstract&pid=S222718992014000200010&lng=es&nrm=iso&tlng=en](http://scielo.sld.cu/scielo.php?script=sci_abstract&pid=S222718992014000200010&lng=es&nrm=iso&tlng=en)
8. García Martín, Ing. Jorge. *Sobre la herramienta Gespro, sus funcionalidades*. 22 de marzo de 2017. Entrevista acerca de las funcionalidades de la herramienta Gespro.
9. Polaino de los Santos, Dra. Cecilia. *Gestión de Proyectos.pdf* [online]. [Accessed 22 November 2016].
10. TAPPELLA, ESTEBAN. *Por qué fracasan los proyectos. La Importancia de la Evaluación Ex ante en el Ciclo de Vida de los Proyectos.pdf*. Instituto de Investigaciones Socioeconómicas Programa de Estudios del Trabajo, el Ambiente y la Sociedad, 2007.
11. Ginarte Durán, Ing. Maidel Beatriz. *Proceso diagnóstico a la actividad productiva en la Universidad de las Ciencias Informáticas.pdf*. November 2012.
12. Velázquez Álvarez, Ing. Raúl. *Objetivos de medición-Objetivos\_ Preguntas\_e\_Indicadores.pdf* [online]. 18 December 2014. [Accessed 14 December 2016]. Available from: [http://mejoras.prod.uci.cu/proceso\\_desarrollo\\_produccion/guidances/whitepapers/resources/Objetivos\\_%20Preguntas\\_e\\_Indicadores.pdf](http://mejoras.prod.uci.cu/proceso_desarrollo_produccion/guidances/whitepapers/resources/Objetivos_%20Preguntas_e_Indicadores.pdf)
13. UNIVERSIDAD DEL CAUCA. *CONCEPTOS BÁSICOS DE LOS SISTEMAS DE INFORMACIÓN*. [online]. [Accessed 8 November 2016]. Available from: <http://fcea.unicauca.edu.co/old/siconceptosbasicos.htm>
14. FRANKLIN RIVERO DUHARTE, JUAN CARLOS GÓMEZ CORREA. y YADIRA BENAVIDES ZAILA. *Reportes con JasperReport, IReport y JFreeChart en Aplicaciones Empresariales en Java.pdf*.2006.
15. CENTRO DE TECNOLOGÍAS DE GESTIÓN DE DATOS (DATEC). *Propuesta de proyecto SDRv2.0: Servidor Dinámico de Reportes*. Ficha técnica para propuesta de desarrollo de RED-SDRv2.0. Universidad de las Ciencias Informáticas, 2016.

# Referencias Bibliográficas

---

16. PROJECT MANAGEMENT INSTITUTE. *Guía de los Fundamentos para la Dirección de Proyectos (Guía del PMBOK)* [online]. Quinta edición. 2013. [Accessed 17 February 2017]. ISBN 978-1-62825-009-1. Available from: [libros\\_pmbok\\_guide5th\\_spanish.pdf](#)
17. GONZÁLEZ, Rosel Sosa, PUPO, Iliana Pérez, GARCÍA, Roberto, PEÑA HERRERA, Esteban y PÉREZ, Pedro Yobanis Pinero. Ecosistema de Software GESPRO-16.05 para la Gestión de Proyectos. *Revista Cubana de Ciencias Informáticas* [online]. 24 November 2016. Vol. 10, no. 0, p. 239–251. [Accessed 17 February 2017]. Available from: <http://rcci.uci.cu/?journal=rcci&page=article&op=view&path%5B%5D=1475>
18. ING. FILIBERTO LÓPEZ PALENZUELA. *Modelo para la toma de decisiones en los proyectos de software basado en los criterios de expertos.pdf*. June 2013.
19. *Modelo para la toma de decisiones en los proyectos de software basado en los criterios de expertos.pdf* [online]. [Accessed 14 December 2016].
20. Entorno Productivo | Portal de la Universidad de las Ciencias Informáticas. [online]. [Accessed 2 February 2017]. Available from: <http://www.uci.cu/?q=entorno-productivo>
21. POBLACIÓN MUESTRA Y MUESTREO. [online]. [Accessed 2 February 2017]. Available from: [http://www.scielo.org.bo/scielo.php?script=sci\\_arttext&pid=S1815-02762004000100012](http://www.scielo.org.bo/scielo.php?script=sci_arttext&pid=S1815-02762004000100012)
22. HERNÁNDES CHENPNE, MARCELO. Diseño de una Metodología Ágil de Desarrollo de Software.pdf. [online]. 2004. [Accessed 16 November 2016]. Available from: <ftp://ucistore.uci.cu/documentacion/Ingenieria%20Software/eXtreme%20Programming/books/Dise%C3%B1o%20de%20una%20Metodolog%C3%ADa%20%C3%81gil%20de%20Desarrollo%20de%20Software.pdf> Facultad de Ingeniería. Universidad de Buenos Aires.
23. Rodríguez Sánchez, Ing. Tamara. Metodología de desarrollo para la Actividad productiva de la UCI.pdf. [online]. [Accessed 16 November 2016]. Available from: <https://excriba.prod.uci.cu/proxy/alfresco/api/node/content/workspace/SpacesStore/a622adab-eac5-4fb3-ba08-a266767fff5f/Metodologia%20UCI.pdf?a=true>

# Referencias Bibliográficas

---

24. ING. YALICE GÁMEZ BATISTA, DR. VALERY MORENO VEGA y MSC. YOAN MARTÍNEZ MÁRQUEZ. Consideraciones generales para el diseño de una herramienta interactiva de simulación de procesos.pdf. [online]. 2008. [Accessed 21 February 2017]. Available from: [http://sedici.unlp.edu.ar/bitstream/hyle/10915/21987/Documento\\_completo.pdf?sequence=1](http://sedici.unlp.edu.ar/bitstream/hyle/10915/21987/Documento_completo.pdf?sequence=1)
25. GONZALO GÉNOVA, JOSÉ M. FUENTES y MARIA C. VALIENTE. Evaluación comparativa de herramientas CASE para UML.pdf. [online]. [Accessed 21 February 2017]. Available from: [http://s3.amazonaws.com/academia.edu.documents/45998726/Evaluacion\\_comparativa\\_de\\_herramientas\\_CASE\\_para\\_UML.pdf?AWSAccessKeyId=AKIAIWOWYYGZ2Y53UL3A&Expires=1487694932&Signature=PeJdsuX9RTPiI0PKwh58Z%2B2wIbM%3D&response-content-disposition=inline%3B%20filename%3DEvaluacion\\_de\\_herramientas\\_CASE\\_para\\_UML.pdf](http://s3.amazonaws.com/academia.edu.documents/45998726/Evaluacion_comparativa_de_herramientas_CASE_para_UML.pdf?AWSAccessKeyId=AKIAIWOWYYGZ2Y53UL3A&Expires=1487694932&Signature=PeJdsuX9RTPiI0PKwh58Z%2B2wIbM%3D&response-content-disposition=inline%3B%20filename%3DEvaluacion_de_herramientas_CASE_para_UML.pdf) Departamento de Informática, Universidad Carlos III de Madrid
26. DENYS PÉREZ-BORROTO LORENZO. *Subsistema de registro de equipos médicos para el sistema informatizado de salud.pdf* [online]. La Habana: UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS, 2011. [Accessed 21 February 2017]. Available from: [https://repositorio\\_institucional.uci.cu/jspui/bitstream/ident/TD\\_03983\\_11/1/TD\\_03983\\_11.pdf](https://repositorio_institucional.uci.cu/jspui/bitstream/ident/TD_03983_11/1/TD_03983_11.pdf)
27. JUAN DIEGO GAUCHAT. *El gran libro de HTML5, CSS3 y Javascript* [online]. primera. marcombo ediciones técnicas, 2012. [Accessed 21 February 2017]. ISBN 978-84-267-1770-2. Available from: [https://books.google.com.cu/books/about/El\\_gran\\_libro\\_de\\_HTML5\\_CSS3\\_y\\_Javascript.html?hl=es&id=szDMIRzwzuUC](https://books.google.com.cu/books/about/El_gran_libro_de_HTML5_CSS3_y_Javascript.html?hl=es&id=szDMIRzwzuUC)
28. MIQUEL ÁNGEL SÁNCHEZ MAZA. *Javascript* [online]. INNOVA 2001. [no date]. [Accessed 21 February 2017]. ISBN 84-95733-18-8. Available from: <https://books.google.com.cu/books/about/Javascript.html?hl=es&id=3x09sewjaHIC>
29. LANZA, Michele y MARINESCU, Radu. *Object-oriented metrics in practice: using software metrics to characterize, evaluate, y improve the design of object-oriented systems; with 8 tables*. Berlin: Springer, 2006. ISBN 978-3-540-24429-5.

# Referencias Bibliográficas

---

30. SHAWN MCCOOL. *Laravel Starter* [online]. Packt Publishing Ltd., 2012. [Accessed 25 April 2017]. ISBN 978-1-78216-090-8. Available from: [https://books.google.com/books/about/Laravel\\_Starter.html?hl=es&id=plBevFHNX3oC](https://books.google.com/books/about/Laravel_Starter.html?hl=es&id=plBevFHNX3oC)
31. REES, Dayle. *Code Happy. Application Development with the Laravel PHP framework for beginners*. Lulu.com, 2012. ISBN 978-1-4717-7749-3. Google-Books-ID: t9LXAwAAQBAJ
32. CLAVADETSCHER, Charles. Autorización en PostgreSQL. [online]. 2015. [Accessed 19 May 2017]. Available from: [http://www.schmiedewerkstatt.ch/documents/04-publications/autorizacion\\_en\\_postgresql\\_script\\_pdfa.pdf](http://www.schmiedewerkstatt.ch/documents/04-publications/autorizacion_en_postgresql_script_pdfa.pdf)
33. Sobre PostgreSQL | [www.postgresql.org.es](http://www.postgresql.org.es). [online]. [Accessed 15 December 2016]. Available from: [http://www.postgresql.org.es/sobre\\_postgresql](http://www.postgresql.org.es/sobre_postgresql)
34. PgAdmin: PostgreSQL administration y management tools. [online]. [Accessed 15 December 2016]. Available from: <https://www.pgadmin.org/>
35. Themes | PhpStorm Themes & Color Styles. [online]. [Accessed 5 May 2017]. Available from: <http://www.phpstorm-themes.com/>
36. Castillo Dayer, Juan Antonio. Diseño e implementación de una aplicación web para la administración de recursos de investigación del área de ingeniería telemática. [online]. 2008. [Accessed 22 February 2017]. Available from: <http://repositorio.upct.es/hyle/10317/257>
37. PROJECT MANAGEMENT INSTITUTE. *Guía' de los fundamentos de la dirección de proyectos (Guía del PMBOK) 3era*. 3era Edición. Newton Square, PA: Project Management Institute, 2004. ISBN 978-1-930699-73-1.
38. GUERRERO, José Miguel Martínez y DELGADO, Camilo Yrés Silva. ÍNDICE DE TABLAS. [online]. [Accessed 6 April 2017]. Available from: <http://pegasus.javeriana.edu.co/~CIS1010IS06/descargas/Memoria%20de%20Trabajo%20de%20Grado%20Versi%C3%B3n%20Completa.docx>

# Referencias Bibliográficas

---

39. VICENTE CHICOTE, Cristina, MOROS, Begoña, TOVAL, Ambrosio. REMM-Studio: an integrated model-driven environment for requirements specification, validation y formatting. [online]. 2007. [Accessed 6 April 2017]. Available from: <http://repositorio.upct.es/hyle/10317/362>
40. ROGER PRESSMAN. *Ingeniería del Software. Un enfoque práctico.pdf* [online]. 6th. Ed. McGraw-Hill, [no date]. [Accessed 22 February 2017]. Available from: <ftp://ucistore.uci.cu/documentacion/Ingenieria%20Software/Ingenieria.del.Software.-.Roger.Pressman.6th.Ed.McGraw-Hill.pdf>
41. SOMMERVILLE, Ian. *Software engineering*. Boston: Addison-Wesley, 2011. ISBN 978-0-13-703515-1.
42. LARMAN, Craig, HERNÁNDEZ RODRÍGUEZ, Luz María y CÁRDENAS ANAYA, Humberto. *UML y patrones: introducción al análisis y diseño orientado a objetos*. Naucalpan de Juárez: Prentice Hall, 1999. ISBN 978-970-17-0261-1.
43. PRESSMAN, Roger S. *Software engineering: a practitioner's approach*. 7. ed. Boston, Mass.: McGraw-Hill/Higher Education, 2010. ISBN 978-0-07-337597-7.
44. MUÑOZ, Vicente Javier Eslava. *El nuevo PHP. Conceptos avanzados*. Vicente Javier Eslava Muñoz, 2013. ISBN 978-84-686-4434-9. Google-Books-ID: NSj3AQAAQBAJ
45. ORTEGA PAREDES, J. G. CONTABILIDAD. Paradigma de reconstrucción a través del giro informático. *JG Ortega Paredes, Contabilidad Practica. Madrid: EAE*. 2012.
46. CIANCIARDO, Juan, Xavier Ferré Grau, y MARÍA ISABEL SÁNCHEZ SEGURA. Los límites del sistema normativo. *Revista de Derecho*. 2004. Vol. 5, no. 5, p. 421–451.
47. OSORIA, D. *Módulo para la extracción y representación de los metadatos en el Sistema de Gestión Documental de audio y video digitales TeVeo Plus V1.0*. 2013. 2013.

# Referencias Bibliográficas

---

48. TAKEYAS, B LÓPEZ. Nomenclatura sugerida para identificar los componentes de un proyecto. [online]. 2015. [Accessed 5 May 2017]. Available from: [http://www.itnuevolaredo.edu.mx/maestros/sis\\_com/takeyas/Apuntes/POO/Apunte](http://www.itnuevolaredo.edu.mx/maestros/sis_com/takeyas/Apuntes/POO/Apunte).
49. CATALDI, Zulma. *Una metodología para el diseño, desarrollo y evaluación de software educativo* [online]. Facultad de Informática, 2000. [Accessed 11 April 2017]. Available from: <http://sedici.unlp.edu.ar/hyle/10915/4055>
50. TUYA, Javier, ROMÁN, Isabel Ramos y COSÍN, José Javier Dolado. *Técnicas cuantitativas para la gestión en la ingeniería del software*. Netbiblo, 2007. ISBN 978-84-9745-204-5. Google-Books-ID: PZQoZ9KTNaEC
51. DR. ROBERTO HERNÁNDEZ SAMPIERI, DR. CARLOS FERNÁNDEZ COLLADO y DRA. MARÍA DEL PILAR BAPTISTA LUCIO. *Metodología de la investigación - Metodología de la Investigación 5ta edición 2010.pdf*. 5ta. México: McGRAW-HILL / INTERAMERICANA EDITORES, S.A. DE C.V, 2010. ISBN 978-607-15-0291-9.



# Anexos

---

## ANEXOS

**Anexo 1: Cuestionario:** “Proceso de toma de decisiones en la UCI y creación de una herramienta para favorecer la toma de decisiones.”

Al llenar este cuestionario usted estará dando su consentimiento de que su nombre estará en la recogida de datos que sirvan como evidencia para el desarrollo de la investigación para el tema de tesis de grado para optar por el título de Ingeniero en Ciencias Informáticas: “Sistema de reportes basado en indicadores para contribuir a la toma de decisiones en la actividad productiva de la UCI”. Su respuesta será valiosa para medir el estado actual del Proceso de toma de decisiones en la UCI y determinar cómo se están tomando las decisiones en la Universidad. Marque con una cruz la evaluación que considere correcta para cada parámetro y exprese su opinión en el caso necesario.

**Nombre y Apellidos del encuestado:**

**Qué rol (es) desempeña actualmente en el proyecto:**

\_\_\_ Arquitecto \_\_\_ Jefe de proyecto \_\_\_ Desarrollador \_\_\_ Analista \_\_\_ Diseñador

\_\_\_ Probador \_\_\_ Administrador de Calidad \_\_\_ Otro

**Experiencia en gestión de proyectos:** \_\_\_ años.

**Preguntas:**

1. ¿Conoce usted alguno de los Modelos de Toma de Decisión en Grupo que existen?

\_\_\_ Sí      \_\_\_ No      \_\_\_ No sé

2. De los Modelos de Toma de Decisión en Grupo que marque con una X en caso de que lo haya utilizado en algún momento durante un proyecto.

\_\_\_ **Modelo racional:** Elige entre varias alternativas a efecto de maximizar los beneficios para la organización.

# Anexos

---

\_\_\_ **Modelo de la racionalidad limitada:** El modelo refleja la tendencia del individuo y el equipo a: elegir lo menor en lugar de lo mejor en cuanto a metas o soluciones alternativas (mínimo satisfactorio).

\_\_\_ **Modelo de la administración basada en evidencias:** Abarca elementos centrales del modelo racional, pero en una versión ampliada y aplicada en la toma de decisiones gerenciales.

\_\_\_ **Modelo político:** La información es ambigua e incompleta. Se enfoca en estimular la creatividad para descubrir problemas, oportunidades y hacer elecciones novedosas para resolver dichos problemas (1).

3. ¿Conoce usted alguno de los Métodos para la toma de decisiones que existen? Marque cuál de los siguientes métodos ha utilizado en los últimos proyectos en los que ha participado.

\_\_\_ **Método de Expertos:** Consiste en expertos que pueden llegar a un mejor pronóstico que una sola persona, consenso logrado respecto a las alternativas sobre la base de criterios.

\_\_\_ **Método Delphi:** Tiene como objetivo alcanzar un consenso sobre una estimación, a través de reuniones, cuestionarios y encuestas.

\_\_\_ **Método de Jerarquías Analíticas (AHP):** Es un método que se basa en descomponer jerárquicamente el problema a tratar en sus elementos componentes.

4. Diga qué herramientas para el apoyo a la toma de decisiones ha estado utilizando en los proyectos en los cuáles está trabajando.

\_\_\_ DotProject      \_\_\_ Microsoft Project 2010      \_\_\_ Gespro

5. ¿Considera usted importante la medición o el uso de indicadores que evalúen el trabajo en la gestión de proyectos?

\_\_\_ Sí      \_\_\_ No      \_\_\_ No sé

Argumente su respuesta.

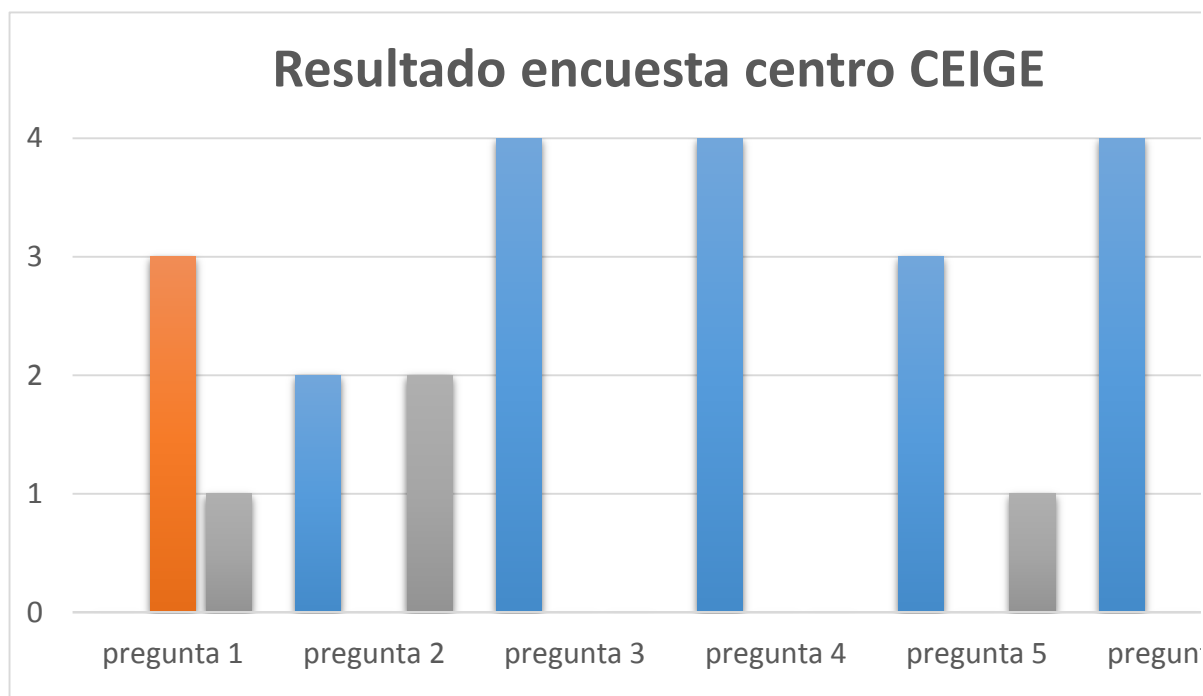
# Anexos

6. La creación de una herramienta para contribuir a la toma de decisiones que, a partir de indicadores específicos (obtenidos de Gespro) almacene datos históricos de los proyectos realizados en la UCI; donde la misma emita reportes y permita comparar resultados de proyectos anteriores con los proyectos actuales, y así obtener los riesgos potenciales en los proyectos actuales. ¿Cree usted que le beneficiaría contar con una herramienta así?

Sí       No       No sé

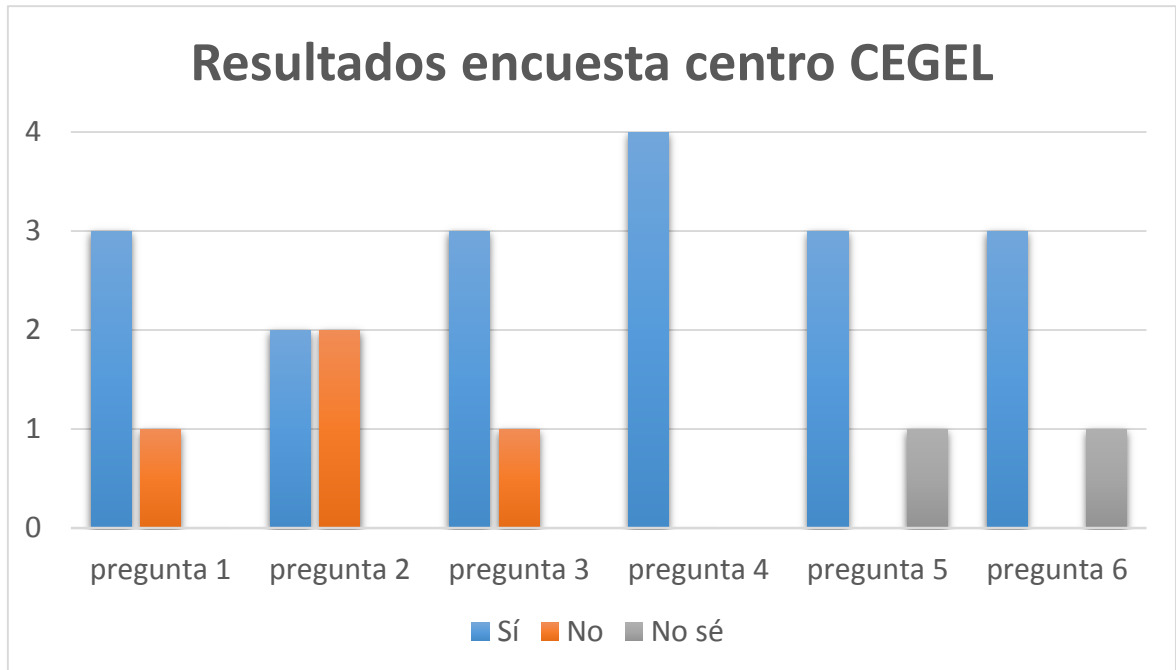
a. Argumente su respuesta

## Anexo 2: Resultado de la encuesta en el Centro CEIGE.



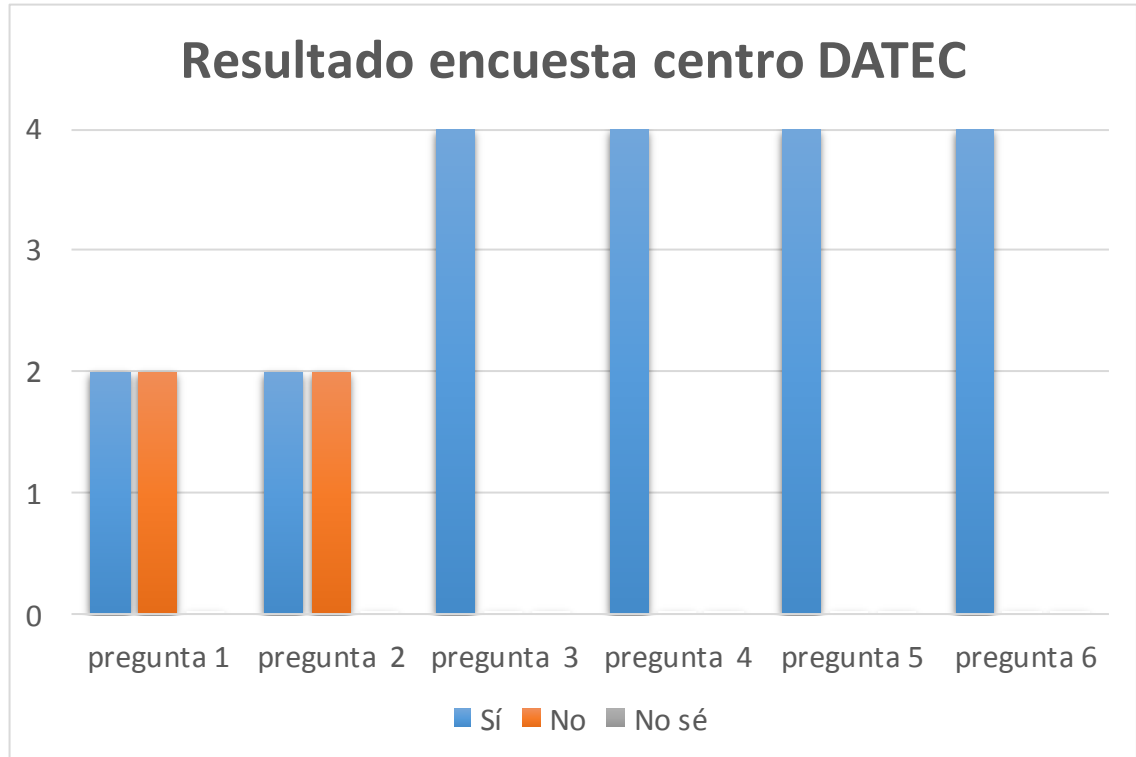
# Anexos

## Anexo 3: Resultado de la encuesta en el Centro CEGEL.



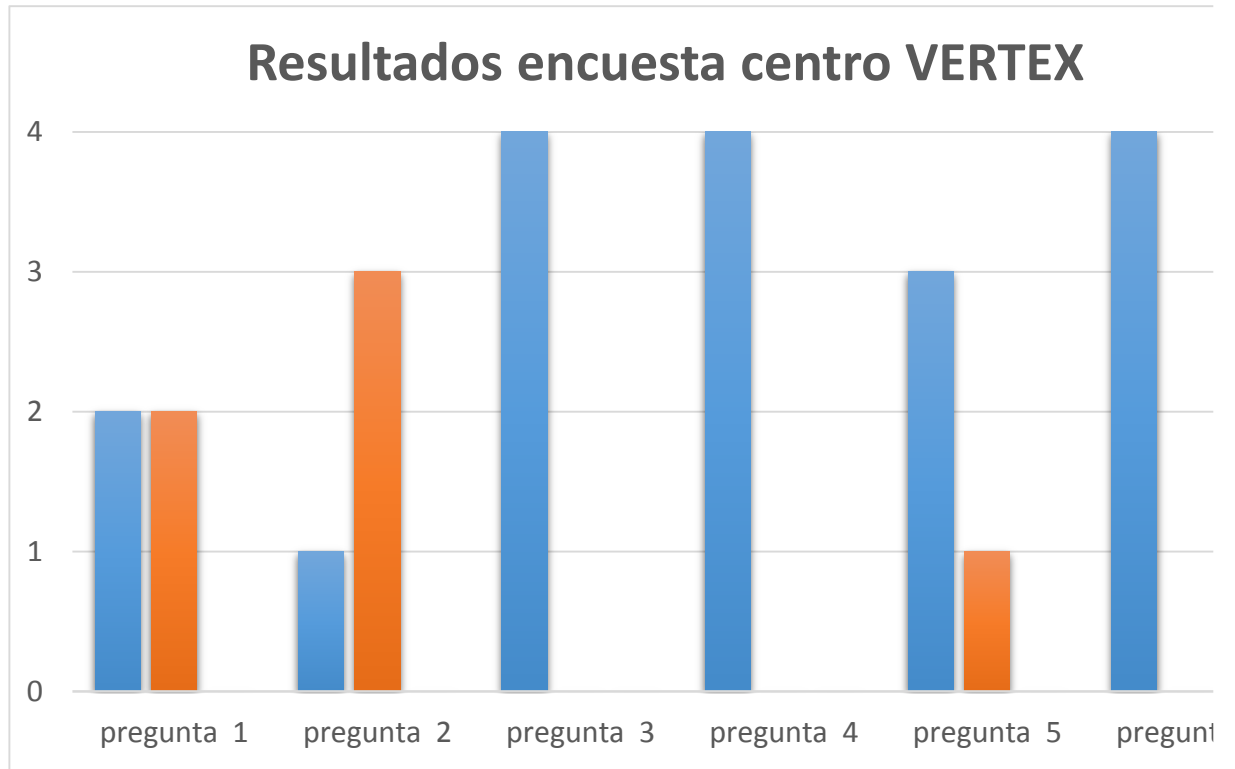
# Anexos

## Anexo 4: Resultado de la encuesta en el Centro DATEC.



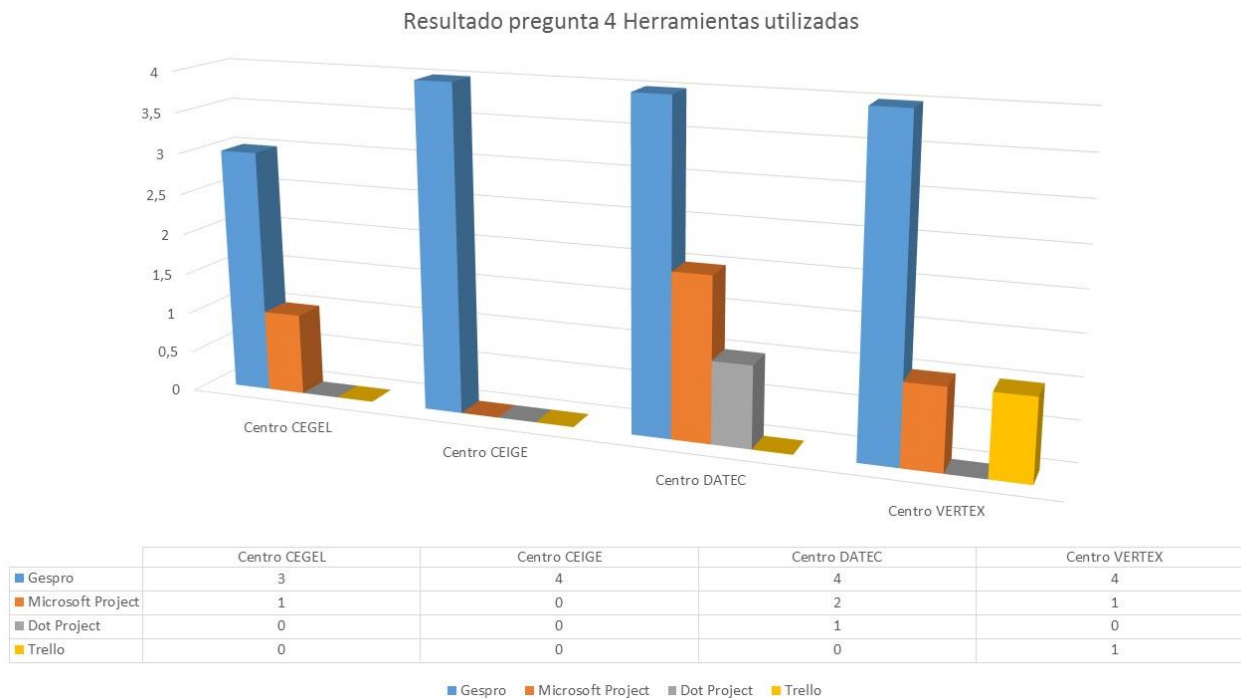
# Anexos

## Anexo 5: Resultado de la encuesta en el Centro VERTEX.



# Anexos

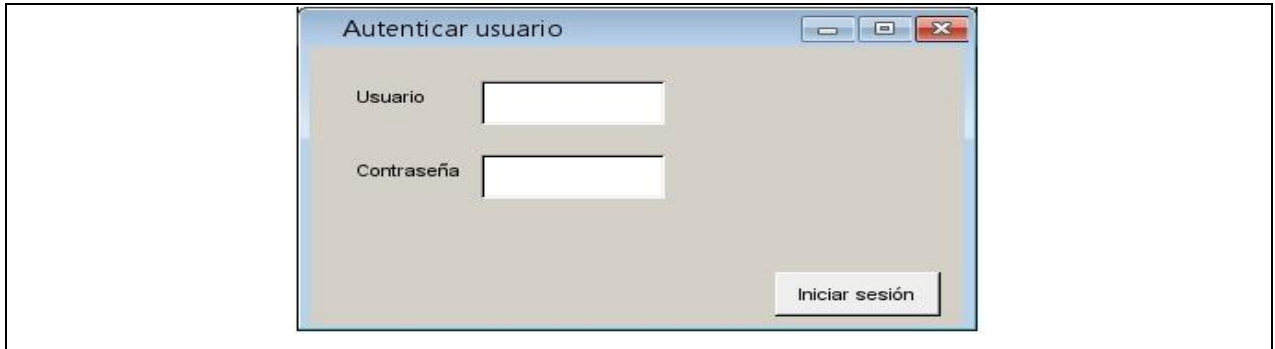
## Anexo 6: Resultado de la encuesta pregunta 4 sobre Herramientas.



## Anexo 7 Historia de usuario Autenticar usuario.

<b>Número: 2</b>	<b>Nombre del requisito:</b> Autenticar usuario.	
<b>Programador:</b> Irina E. Mayedo Tamayo	<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> <i>Media</i>	<b>Tiempo Estimado:</b> 1 día	
<b>Riesgo en Desarrollo:</b>	<b>Tiempo Real:</b> 8 horas	
<b>Descripción:</b> Esta funcionalidad debe permitir al usuario que se encuentre registrado previamente, acceder al sistema.		
<b>Observaciones:</b>		
<b>Prototipo elemental de interfaz gráfica de usuario:</b>		

# Anexos



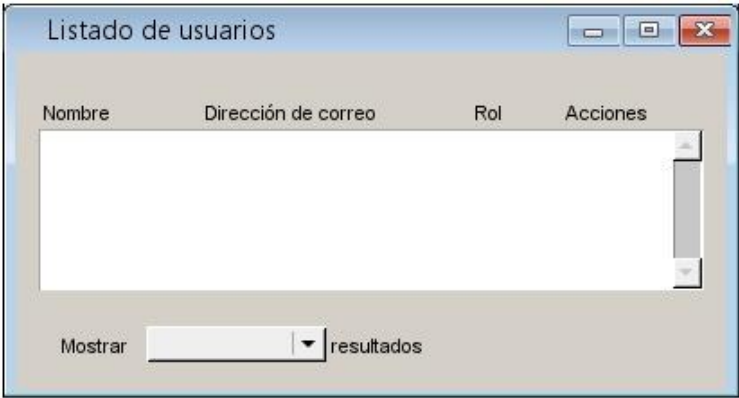
## Anexo 8: Historia de usuario Gestionar roles de los usuarios del sistema.

<b>Número: 6</b>	<b>Nombre del requisito:</b> Gestionar roles de los usuarios del sistema.	
<b>Programador:</b> Irina E. Mayedo Tamayo	<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> <i>Media</i>	<b>Tiempo Estimado:</b> 1 día	
<b>Riesgo en Desarrollo:</b>	<b>Tiempo Real:</b> 12 horas	
<b>Descripción:</b> La funcionalidad debe permitir otorgar o derogar permisos a los usuarios según el rol que posean.		
<b>Observaciones:</b>		
<b>Prototipo elemental de interfaz gráfica de usuario:</b>		
<p>Un prototipo de interfaz de usuario para administrar roles. El diálogo tiene un título "Administrar roles" y contiene dos campos de selección: "Rol" y "Permisos". En la parte inferior hay tres botones: "Cancelar", "Modificar" y "Aceptar".</p>		



# Anexos

## Anexo 9: Historia de usuario de la funcionalidad Listar los usuarios registrados en el sistema.

<b>Número:</b> 13	<b>Nombre del requisito:</b> Listar los usuarios registrados en el sistema.		
<b>Programador:</b> Irina E. Mayedo Tamayo	<b>Iteración Asignada:</b> 4		
<b>Prioridad:</b> <i>Media</i>	<b>Tiempo Estimado:</b> 2 días		
<b>Riesgo en Desarrollo:</b>	<b>Tiempo Real:</b> 30 horas		
<b>Descripción:</b> La funcionalidad debe permitir mostrar la lista de usuarios que tiene el sistema registrados.			
<b>Observaciones:</b>			
<b>Prototipo elemental de interfaz gráfica de usuario:</b>			
			

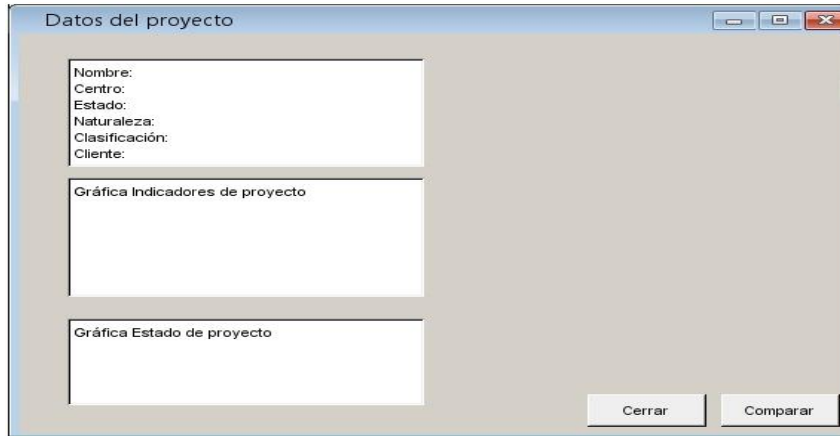
## Anexo 10: Historia de usuario Visualizar los datos generales del proyecto.

<b>Número:</b> 14	<b>Nombre del requisito:</b> Visualizar los datos generales del proyecto.		
<b>Programador:</b> Irina E. Mayedo Tamayo	<b>Iteración Asignada:</b> 2		
<b>Prioridad:</b> Media	<b>Tiempo Estimado:</b> 5 días		
<b>Riesgo en Desarrollo:</b>	<b>Tiempo Real:</b> 100 horas		
<b>Descripción:</b> La funcionalidad debe permitir mostrar datos e indicadores de cada proyecto seleccionado.			

# Anexos

**Observaciones:**

**Prototipo elemental de interfaz gráfica de usuario:**

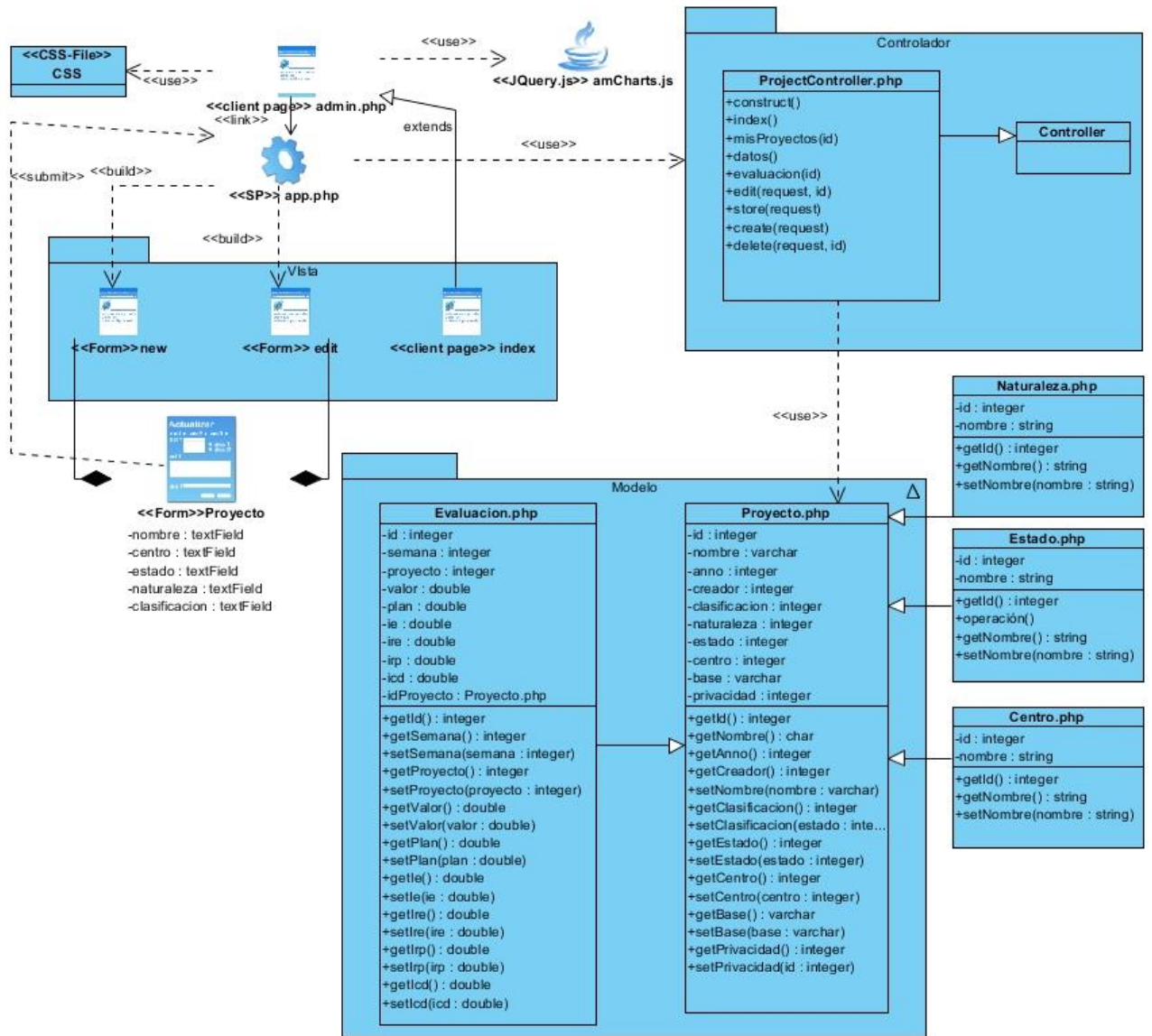


## Anexo 11: Historia de usuario Guardar búsquedas.

<b>Número:</b> 22	<b>Nombre del requisito:</b> Guardar búsquedas.	
<b>Programador:</b> Irina E. Mayedo Tamayo	<b>Iteración Asignada:</b> 4	
<b>Prioridad:</b> <i>Alta</i>	<b>Tiempo Estimado:</b> 3 días	
<b>Riesgo en Desarrollo:</b>	<b>Tiempo Real:</b> 24 horas	
<b>Descripción:</b> La funcionalidad debe permitir guardar las búsquedas realizadas por el usuario.		
<b>Observaciones:</b>		
<b>Prototipo elemental de interfaz gráfica de usuario:</b>		
<p>El prototipo muestra una ventana de diálogo titulada "Guardar búsquedas". En la parte superior izquierda, hay un campo de texto etiquetado "Resultados búsqueda". A la derecha de este campo, hay un botón "Guardar". En la parte inferior de la ventana, hay dos botones: "Cancelar" y "Aceptar".</p>		

# Anexos

Anexo 12: Diagrama de clases del gestionar evaluaciones del proyecto.



Anexo 13: Diseño de caso de prueba para el camino 2.

Diseño de caso de prueba para el camino 2	
Descripción	El dato se obtiene a través del parámetro <i>id</i> .

## *Anexos*

<b>Condición de ejecución</b>	Campos válidos: El campo <b>Gráfico</b> permite visualizar la gráfica correspondiente al reporte de los indicadores del proyecto seleccionado. Para acceder al reporte el método realiza una búsqueda a través del <b>id</b> del proyecto seleccionado. Si el <b>id</b> del proyecto seleccionado corresponde con el <b>id</b> de ese mismo proyecto almacenado y tienen datos los indicadores, entonces el sistema muestra el reporte.
<b>Entrada</b>	Selección de la opción para ver el <b>Gráfico</b> de las evaluaciones de un proyecto.
<b>Resultados esperados</b>	No se debe obtener ningún dato
<b>Respuesta del sistema</b>	Respuesta satisfactoria. No devuelve nada.

### Anexo 14: Diseño de caso de prueba para el camino 3.

<b>Diseño de caso de prueba para el camino 3</b>	
<b>Descripción</b>	El dato se obtiene a través del parámetro <b>id</b> .
<b>Condición de ejecución</b>	Campos válidos: El campo <b>Gráfico</b> permite visualizar la gráfica correspondiente al reporte de los indicadores del proyecto seleccionado. Para acceder al reporte el método realiza una búsqueda a través del <b>id</b> del proyecto seleccionado. Si el <b>id</b> del proyecto seleccionado corresponde con el <b>id</b> de ese mismo proyecto almacenado y tienen datos los indicadores, entonces el sistema muestra el reporte.
<b>Entrada</b>	Selección de la opción para ver el <b>Gráfico</b> de las evaluaciones de un proyecto.
<b>Resultados</b>	No se debe obtener ningún dato.

## Anexos

---

esperados	
Respuesta del sistema	Respuesta insatisfactoria. No devuelve nada.

### Anexo 15: Diseño de caso de prueba para el camino 4.

Diseño de caso de prueba para el camino 4	
Descripción	El dato se obtiene a través del parámetro <i>id</i> .
Condición de ejecución	Campos válidos: El campo <b>Gráfico</b> permite visualizar la gráfica correspondiente al reporte de los indicadores del proyecto seleccionado. Para acceder al reporte el método realiza una búsqueda a través del <i>id</i> del proyecto seleccionado. Si el <i>id</i> del proyecto seleccionado corresponde con el <i>id</i> de ese mismo proyecto almacenado y tienen datos los indicadores, entonces el sistema muestra el reporte.
Entrada	Selección de la opción para ver el <b>Gráfico</b> de las evaluaciones de un proyecto.
Resultados esperados	No se debe obtener ningún dato.
Respuesta del sistema	Respuesta satisfactoria. No devuelve nada.

### Anexo 16: Diseño de caso de prueba para el camino 5.

Diseño de caso de prueba para el camino 5	
Descripción	El dato se obtiene a través del parámetro <i>id</i> .

## Anexos

<b>Condición de ejecución</b>	<p>Campos válidos: El campo <b>Gráfico</b> permite visualizar la gráfica correspondiente al reporte de los indicadores del proyecto seleccionado.</p> <p>Para acceder al reporte el método realiza una búsqueda a través del <b>id</b> del proyecto seleccionado.</p> <p>Si el <b>id</b> del proyecto seleccionado corresponde con el <b>id</b> de ese mismo proyecto almacenado y tienen datos los indicadores, entonces el sistema muestra el reporte.</p>
<b>Entrada</b>	Selección de la opción para ver el <b>Gráfico</b> de las evaluaciones de un proyecto.
<b>Resultados esperados</b>	No se debe obtener ningún dato.
<b>Respuesta del sistema</b>	Respuesta satisfactoria, no obtiene datos para graficar.

### Anexo 17: Descripción de caso de prueba Guardar búsqueda.

Escenario	Descripción	Respuesta del sistema	Flujo central
<b>EC 1.1 Guardar búsqueda.</b>	Permite guardar la búsqueda realizada.	Guarda la búsqueda realizada y muestra un mensaje de búsqueda guardada correctamente.	<ol style="list-style-type: none"> <li>1. Seleccionar la opción "<b>Reportes/Graficar</b>".</li> <li>2. El sistema realiza el reporte como lo seleccionó el usuario en su criterio de búsqueda especificado.</li> <li>3. Se selecciona el botón "<b>Guardar búsqueda</b>".</li> <li>4. El sistema guarda la búsqueda en "<b>Mis búsquedas</b>" y muestra un mensaje de "<b>Búsqueda guardada correctamente</b>".</li> </ol>

### Anexo 18: Descripción de caso de prueba Exportar.

Escenario	Descripción	Respuesta del sistema	Flujo central
<b>EC 1.1 Exportar</b>	Permite exportar los reportes en	Exporta en el formato seleccionado.	1. Seleccionar las opciones " <b>Administración/Servicios/Reportes</b> " + <b>Proyecto+datos+Búsqueda</b>

# Anexos

	diferentes formatos.		<b>Avanzada+Graficar.</b> 2. Se muestran las gráficas y se presiona el botón a la derecha arriba de <b>Exportar como</b> . 3. Se selecciona el formato en el que se desea exportar. 4. Se guarda en el formato seleccionado. 5. El sistema lo mantiene en la misma vista.
--	----------------------	--	---

## Anexo 19: Descripción de caso de prueba Comparar proyectos.

Escenario	Descripción	Respuesta del sistema	Flujo central
<b>EC 1.1</b> <b>Comparar proyectos.</b>	Se debe acceder a la opción de comparar y el sistema muestra la comparación de un proyecto seleccionado con los proyectos del usuario.	Se muestra la comparación de un proyecto seleccionado con los proyectos del usuario en una gráfica.	1. Se debe seleccionar <b>"Servicios/Búsqueda Avanzada/Buscar"</b> + <b>"Realizar búsqueda"</b> . Se presiona el botón <b>"comparar"</b> . 3. Se visualiza debajo del resultado de la búsqueda, una comparación del proyecto seleccionado con todos los proyectos del usuario en una gráfica de líneas. 4. El usuario presionando en la leyenda de la gráfica puede <b>eliminar/incorporar</b> sus proyectos en la misma. 5. Si presiona el botón <b>Diferencias</b> se grafican las diferencias de los proyectos del usuario con respecto al proyecto seleccionado.

## Anexo 20: Descripción de caso de prueba Graficar.

Escenario	Descripción	Agrupar por	Buscar por	Respuesta del sistema	Flujo central

# Anexos

---

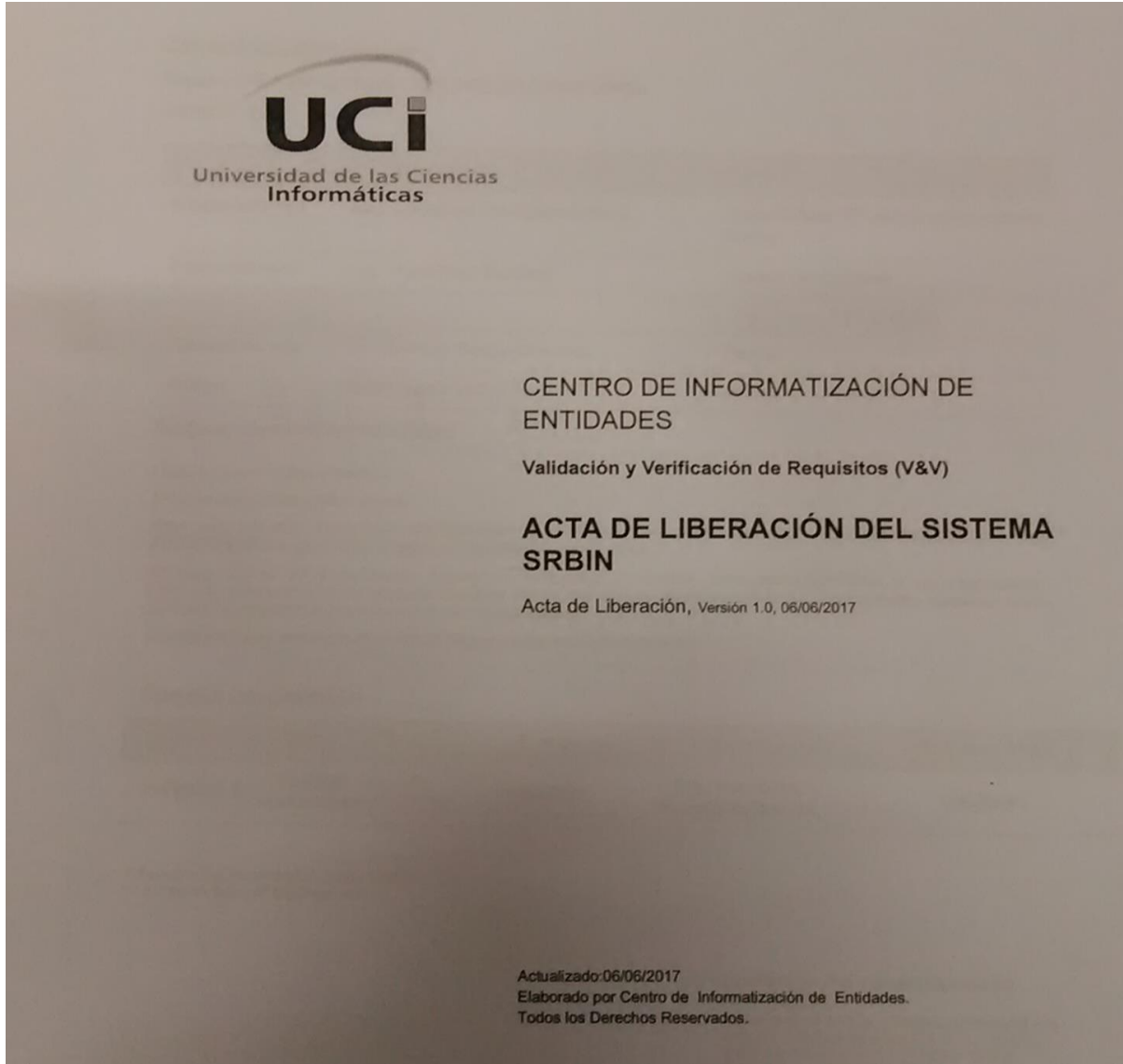
<b>EC</b>	<b>1.1</b>	Permite graficar por elementos de proyectos.	V(Estado)	V(Gráfico de barras, clasificación, Naturaleza del proyecto)	Muestra el gráfico correspondiente a lo seleccionado para agrupar y buscar.	<ol style="list-style-type: none"><li>1. Seleccionar la opción "<b>Reportes</b>".</li><li>2. Se especifica en los campos de selección correspondientes a las gráficas que el usuario que se desea buscar.</li><li>3. Selecciona la opción de menú "<b>Graficar</b>".</li><li>4. El sistema gráfica el reporte como lo seleccionó el usuario en su criterio de búsqueda especificado.</li></ol>
-----------	------------	--	-----------	--	---	--



# Anexos

---

## Anexo 21: Acta de liberación de Pruebas Internas



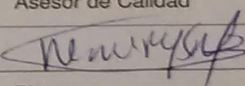
# Anexos

UCI | Acta de liberación

## Control del documento

Título: ACTA DE LIBERACIÓN DEL SISTEMA SRBin

Versión: 1.0

	Nombre	Cargo
Elaborado por	Ing. Yordanka Parellada Batista	Especialista "B" en Ciencias Informáticas
Revisado por	Ing. Yisel Niño Benitez	Asesor de Calidad
Aprobado por	Dr. Nemury Silega Martínez	Firma 
Cargo	Subdirector I+D+i	Fecha 06/06/2017

## Reglas de confidencialidad

Clasificación: Uso Interno

Forma de distribución: Word

Este documento contiene información propietaria del **CENTRO DE INFORMATIZACIÓN DE ENTIDADES**, y es emitido para un propósito específico.

El que recibe el documento asume la custodia y control, comprometiéndose a no reproducir, divulgar, difundir o de cualquier manera hacer de conocimientos público su contenido, excepto para cumplir el propósito para el cual se ha generado.

Las reglas son aplicables a las 8 páginas de este documento.

## Control de cambios

Versión	Lugar*	Tipo**	Fecha	Autor	Descripción
Versión 1.0	Todo el documento	A	05/06/2017	Ing. Yordanka Parellada Batista	Creación

\* Sección del documento, Tabla, Figura.

\*\* A Alta; B Baja; M Modificación

**CENTRO DE INFORMATIZACIÓN DE ENTIDADES**  
Universidad de las Ciencias Informáticas  
Carretera a San Antonio Km 2 ½. Torrens.  
Boyeros. Ciudad de La Habana. Cuba  
Teléfono: + 53 (7) 837 3680  
E-mail: software.gestion@uci.cu

# Anexos

UCI | Acta de liberación

## Índice de contenidos

<b>1</b>	<b>INTRODUCCIÓN</b>	<b>4</b>
1.1	OBJETIVO	4
1.2	ALCANCE	4
1.3	DEFINICIONES Y ACRÓNIMOS	4
1.4	REFERENCIAS	4
<b>2</b>	<b>DATOS DEL PRODUCTO</b>	<b>5</b>
2.1	CLASIFICADO COMO:	5
2.2	DETALLE DE LOS ELEMENTOS PROBADOS Y SU ESTADO FINAL:	5
2.3	CANTIDAD DE ITERACIONES:	5
<b>3</b>	<b>ELEMENTOS REVISADOS O PROBADOS Y HERRAMIENTAS UTILIZADAS</b>	<b>5</b>
3.1	CANTIDAD TOTAL DE HORAS EMPLEADAS Y RANGO DE FECHAS:	6
3.2	ESTRUCTURA DEL EQUIPO DE PRUEBA EMPLEADO Y TURNOS DE TRABAJO:	6
<b>4</b>	<b>EVALUADO POR:</b>	<b>6</b>
4.1	ESPECIALISTA PRINCIPAL ASIGNADO:	6
4.2	OTRO PERSONAL ESPECIALIZADO PARTICIPANTE:	6

**CENTRO DE INFORMATIZACIÓN DE ENTIDADES**  
**Universidad de las Ciencias Informáticas**  
Carretera a San Antonio Km 2 ½. Torrens.  
Boyeros. Ciudad de La Habana. Cuba  
Teléfono: + 53 (7) 837 3680  
E-mail: software.gestion@uci.cu

# Anexos

---

UCI | Acta de liberación

## 1 Introducción

### 1.1 Objetivo

A través de este documento se emite el acta de liberación del sistema SRBin.

### 1.2 Alcance

Está destinado a todos los miembros de la entidad.

### 1.3 Definiciones y acrónimos

DGP: Diseño de casos de prueba.

### 1.4 Referencias

IEEE. 1991. *IEEE Standard Glossary of Software Engineering Terminology*. Spring 1991 Edition. 1991. IEEE Standard 610.12-1990.

**CENTRO DE INFORMATIZACIÓN DE ENTIDADES**

Universidad de las Ciencias Informáticas  
Carretera a San Antonio Km 2 ½. Torrens.  
Boyeros. Ciudad de La Habana. Cuba  
Teléfono: + 53 (7) 837 3680  
E-mail: software.gestion@ucl.cu

# Anexos

UCI | Acta de liberación

## 2 Datos del producto

Emitida a favor de: Desarrollador del sistema SRBin

Fecha de emisión del acta: 06/06/2017

Responsable: Ing. Virtudes Milagro Figueredo Lara

Cargo: Especialista "A" en Ciencias Informáticas

### 2.1 Clasificado como:

- Aplicación y Diseños de caso de prueba(DCP).

### 2.2 Detalle de los elementos probados y su estado final:

Artefacto	Estado Final
Aplicación	0 No Conformidades
DCP	0 No Conformidades

### 2.3 Cantidad de iteraciones:

Para la revisión se emplearon un total de 2 iteraciones de trabajo.

El desglose de liberación de los componentes es el siguiente; se revisaron la aplicación y los DCP, en 2 iteraciones en total para lograr el resultado de 0 No Conformidades al cierre de la segunda iteración.

Artefacto	Versión	Estado final	Cantidad Iteraciones	Tipos de pruebas realizadas	Fecha de liberación
Aplicación y DCP	Versión 1.0	0 No Conformidades	2 iteraciones total	Pruebas Funcionales	06/06/2017

## 3 Elementos revisados o probados y herramientas utilizadas

Elemento	Herramienta
Aplicación	Diseño de Caso de Pruebas

CENTRO DE INFORMATIZACIÓN DE ENTIDADES  
Universidad de las Ciencias Informáticas  
Carretera a San Antonio Km 2 1/2, Torrens.  
Boyeros, Ciudad de La Habana, Cuba  
Teléfono: + 53 (7) 837 3680  
E-mail: software.gestion@uci.cu

15

# Anexos

UCI | Acta de liberación

### 3.1 Cantidad total de horas empleadas y rango de fechas:

Se emplearon un total de 8 horas efectivas de trabajo desde el 05/06/2017 hasta el 06/06/2017.

### 3.2 Estructura del equipo de prueba empleado y turnos de trabajo:

Las pruebas se realizaron en un total de 2 turnos de trabajo, con 1 probador, toda la actividad estuvo dirigida por el Asesor de Calidad.

### 4 Evaluado por:

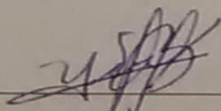
Ing. Yordanka Parellada Batista

#### 4.1 Especialista principal Asignado:

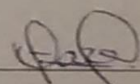
Ing. Yisel Niño Benítez

#### 4.2 Otro personal especializado participante:

Nombres y Apellidos	Cargo
Ing. Yordanka Parellada Batista	Especialista "B" en Ciencias Informáticas
Ing. Virtudes Milagro Figueredo Lara	Especialista "A" en Ciencias Informáticas
Irina Elsa Mayedo Tamayo	Desarrollador



Ing. Yisel Niño Benítez  
Asesor de Calidad

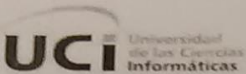


Ing. Virtudes Milagro Figueredo Lara  
Especialista "A" en Ciencias Informáticas

**CENTRO DE INFORMATIZACIÓN DE ENTIDADES**  
Universidad de las Ciencias Informáticas  
Carretera a San Antonio Km 2 ½. Torrens.  
Boyeros. Ciudad de La Habana. Cuba  
Teléfono: + 53 (7) 837 3680  
E-mail: software.gestion@uci.cu

# Anexos

## Anexo 22: Acta de Aceptación con el cliente.



### Acta de aceptación

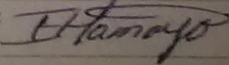
**ACTA DE ACEPTACIÓN CON EL CLIENTE**

En cumplimiento del *Trabajo de diploma* orientado por la Facultad 3, para ser desarrollado en el Centro de Informatización de Entidades (CEIGE) perteneciente a la Universidad de las Ciencias Informáticas (UCI), titulado: "*Sistema de reportes basado en indicadores para contribuir a la toma de decisiones en la actividad productiva de la UCI*". Se realiza la entrega de los productos de trabajo de la **Parte Desarrolladora**: Irina Elsa Mayedo Tamayo a la **Parte Cliente**: Ing. Virtudes Milagro Figueredo Lara, como se detalla a continuación:

- Historias de usuario.
- Diseños de casos de pruebas.
- Diagramas de clases del diseño.
- Modelo de datos.
- Arquitectura del sistema.
- Diagrama de componente.
- Diagrama de despliegue.
- Sistema de Reportes SRBin.

La **Parte Cliente**, luego de haber revisado los productos de trabajo entregados, así como la explotación del Sistema de Reportes SRBin, determina que está de acuerdo con el producto final mostrado, el cual fue desarrollado bajo los requisitos previamente definidos.

En la aceptación del producto la **Parte Cliente** realizó un total de 12 solicitudes de cambios en 3 iteraciones, las cuales fueron resueltas por la **Parte Desarrolladora**. Estas solicitudes fueron solucionadas para el correcto funcionamiento del sistema.

Entrega: Parte Desarrolladora	Recibe: Parte Cliente
<b>Nombre y apellidos:</b> Irina Elsa Mayedo Tamayo Estudiante, Facultad 3	<b>Nombre y apellidos:</b> Ing. Virtudes Milagro Figueredo Lara Jefe de Proyecto, Centro CEIGE
<b>Fecha:</b> 15/6/17	<b>Fecha:</b> 15/6/17
<b>Firma:</b> 	<b>Firma:</b> 