



**Universidad de las Ciencias Informáticas**

**Facultad 4**

**Trabajo de Diploma para optar por el título de**

**Ingeniero en Ciencias Informáticas.**

**Título:**

Módulo para el análisis de datos basado en CRISP-DM para la plataforma XEDRO-GESPRO.

**Autores:**

Yem Liyanis Tamayo Cárdenas

Pedro Alejandro Villavicencio Martínez

**Tutores:**

MSc. Rosel Sosa González

Ing. Claudia Marrero Águila

Junio, 2018



“Estudiar no es un acto de consumir ideas,  
sino de crearlas y recrearlas.”

*Paulo Freire*

**Declaración de autoría**

Declaramos ser los autores de la presente tesis que tiene por título: Módulo para el análisis de datos basado en CRISP-DM para la plataforma XEDRO-GESPRO y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo. Para que así conste firmamos la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

\_\_\_\_\_

Firma del autor

Yem Liyanis Tamayo Cárdenas

\_\_\_\_\_

Firma del tutor

Rosel Sosa González

\_\_\_\_\_

Firma del autor

Pedro Alejandro Villavicencio Martínez

\_\_\_\_\_

Firma del tutor

Claudia Marrero Águila

**Agradecimientos**

*Yo Yem Liyanis Tamayo Cárdenas le agradezco a mis padres Emilio y Sandra por haberme dado todo el apoyo para que pudiera lograr mis sueños de ser ingeniera, a mis tutores Rosel por haberme guiado en todo este tiempo, por haberme ayudado incondicionalmente y por convertirse en mi amigo, a Claudia por además de ser mi tutora ser mi hermanita y aguantarme estos años que no han sido fáciles, a Sony por esperarme todo este tiempo y soportar la distancia por tal que yo cumpliera mi propósito, a la Nisse por estar ahí siempre, ser mi amiga y apoyarme en todo este recorrido, a la loca de Sol a Roberto y Alexander por soportarme a pesar de no ser una tarea fácil, a la amiga Jessi, a mis profesores, a todas esas amistades que no se olvidan y a todas esas personas que a pesar de no nombrar formaron parte de mi vida universitaria y que nunca voy a olvidar y por último a mi compañero de tesis que casi me deja sin vida pero que también me soporto muchas veces y lo solucionaba todo con un simple "pinky tienes poderes".*

**Agradecimientos**

*Yo Pedro Alejandro Villavicencio Martínez le agradezco a todas las personas que participaron e hicieron posible este triunfo, muchas gracias por su apoyo y enseñanza en especial a mis padres, a mis abuelos, a mi familia en general ya que gracias a ellos sigo siendo una persona de bien pese a cualquier tipo de situación, gracias por su eterno apoyo, así como por ayudarme a cumplir este gran paso en mi vida y lo más importante, ser partícipes de este, también le agradezco profundamente a nuestro tutor Rosel Sosa por su inmensa ayuda, su infinita paciencia, así como su inagotable apoyo, a mi otra tutora Claudia Marrero, más que tutora eres parte de mi colectivo de amigos, gracias a Leo mi hermano de otra madre y padre que siempre pude contar con él para todo, Osley, Brian, Jonathan, Yesenia, Greter, Álvaro, Amalia, Antonio, Josué, Sol a todos mis compañeros del eterno grupo(5101-hasta-4501), en fin a todas esas amistades que reuní en estos largos 5 años, también a Seidy y a su familia por soportarme y por ultimo a mi compañera de tesis Yem, por dios que puedo decir de ti, no tengo palabras para agradecerte, sabes que más que una amiga eres una hermana para mí, pero si te voy a decir una cosa, voy a ir a un relojero porque me dejaste sin pines, siempre te diré 'Tienes poderes'. Los quiero a todos, sinceramente han dejado una huella en mí.*

**Dedicatoria**

*Yo Yem Lyanis le dedico esta tesis a mis padres, a mi pareja, a mi bebe que está por nacer y a toda mi familia completa que son tantos que no puedo mencionar.*

**Dedicatoria**

*Yo Pedro Alejandro Villavicencio Martínez dedico este logro en especial a mi abuela Mimi, a mis padres, a mi familia en general por haberme apoyado a lo largo de todo este camino que ya estoy culminando, a todos los que me han apoyado durante todo este periodo, a los que estudiaron conmigo en esos momentos de alegría o tristeza y que nunca me dejaron solo en ningún momento de este arduo camino.*

## **Resumen**

La gestión de proyectos actualmente es una disciplina que se generaliza en el entorno empresarial y consiste en la aplicación de conocimientos, metodologías y técnicas. El análisis de datos es una técnica que se utiliza para procesar grandes volúmenes de datos, y permite además de la inspección de estos, su limpieza y recuperación. La metodología CRISP-DM se rige por seis fases que garantizan un análisis efectivo de los datos. La investigación proyecta como objetivo implementar un módulo informático a partir de la metodología, realizando un análisis detallado a un conjunto de datos almacenados por la plataforma XEDRO-GESPRO, lo que facilita la planificación de futuros proyectos a los usuarios a partir de experiencias anteriores, además de predecir aspectos alarmantes sobre los datos con el uso de algoritmos, planes de ejecución y experimentos, y permite que se pueda llevar a cabo una correcta interpretación de estos para la adecuada toma de decisiones.

**Palabras clave:** Análisis de datos, CRISP-DM, fases, gestión de proyectos, toma de decisiones.

**Índice General**

<b>Introducción.....</b>	<b>1</b>
<b>Capítulo 1. Marco Teórico Referencial .....</b>	<b>5</b>
1.1 Gestión de proyectos.....	5
1.2 Análisis de datos y su importancia.....	8
1.3 Metodología para el desarrollo de proyectos en minería de datos basados en CRISP-DM .....	9
1.4 Plataformas para el análisis de datos .....	12
<b>1.4.1 ¿Qué es XEDRO GESPRO?.....</b>	<b>14</b>
<b>1.4.2 Surgimiento.....</b>	<b>15</b>
<b>1.4.3 Evolución.....</b>	<b>15</b>
<b>1.4.4 Características generales del XEDRO GESPRO .....</b>	<b>15</b>
1.5 Herramientas y metodologías de desarrollo .....	16
<b>1.5.1 Metodologías de desarrollo.....</b>	<b>16</b>
<b>1.5.2 Herramienta de modelado .....</b>	<b>19</b>
<b>1.5.3 Gestor de base de datos .....</b>	<b>20</b>
<b>1.5.4 Entorno de desarrollo integrado.....</b>	<b>21</b>
<b>1.5.5 Lenguaje de programación .....</b>	<b>21</b>
<b>1.5.6 Framework.....</b>	<b>22</b>
<b>1.5.7 Estilo CSS.....</b>	<b>23</b>
1.6 Conclusiones Parciales .....	23
<b>Capítulo 2. Características y diseño del sistema .....</b>	<b>24</b>
2.1 Descripción de la propuesta de solución .....	24
2.2 Modelo de Dominio .....	25
2.3 Especificación de los requisitos del sistema .....	27
<b>2.3.1 Requisitos funcionales (RF).....</b>	<b>28</b>
<b>2.3.2 Requisitos no funcionales (RNF).....</b>	<b>32</b>

2.4 Historias de Usuario .....	34
2.5 Patrón arquitectónico.....	38
2.6 Patrones de diseño .....	39
<b>2.6.1 Patrones de Diseño GRASP .....</b>	<b>39</b>
<b>2.6.2 Patrones de Diseño GOF .....</b>	<b>40</b>
2.7 Diagrama entidad-relación .....	41
2.8 Conclusiones Parciales .....	42
<b>Capítulo 3. Implementación y Pruebas.....</b>	<b>43</b>
3.1. Estilo de escritura.....	43
3.2. Estándares de Codificación .....	43
3.3. Diagrama de despliegue.....	45
3.4. Pruebas de software.....	46
3.5. Ambientes de pruebas.....	47
3.6. Diseño de casos de pruebas .....	47
3.7. Ejecución de los casos de pruebas de aceptación .....	50
3.8. Conclusiones parciales.....	50
<b>Conclusiones Generales .....</b>	<b>52</b>
<b>Recomendaciones.....</b>	<b>59</b>
<b>Anexos .....</b>	<b>60</b>

**Índice de Figuras**

Ilustración 1 Encuesta de metodologías(Fuente: VÁSQUEZ VALLES, 2014). ..... 9

Ilustración 2 Esquema de los cuatro niveles de CRIPS-DM (Fuente: CRIPS-DM, 2000). 10

Ilustración 3 Modelo de proceso CRISP–DM (Fuente: Elaboración propia). ..... 10

Ilustración 4 Vista Arquitectónica del Módulo. .... 25

Ilustración 5 Modelo de dominio (Fuente: elaboración propia). ..... 26

Ilustración 6 Arquitectura Modelo-Vista-Controlador (Fuente: Elaboración propia). ..... 39

Ilustración 7 Modelo entidad relación (Fuente: Elaboración propia) ..... 41

Ilustración 8 Identación(Elaboración propia) ..... 44

Ilustración 9 Declaraciones(Elaboración propia) ..... 44

Ilustración 10 Clases(Elaboración propia)..... 45

Ilustración 11 Diagrama de despliegue (Fuente: Elaboración propia). ..... 45

Ilustración 12 Encuesta de satisfacción (Fuente: Elaboración propia)..... 49

Ilustración 13 Resultados de la encuesta (Fuente: Elaboración propia). ..... 49

Ilustración 14 Resumen de no conformidades (Fuente: Elaboración propia)..... 50

**Índice de Tablas**

Tabla 1.1 Fases de SCRUM (Fuente: Satpathy, 2016). .....	17
Tabla 2.2.1 Descripción de entidades (Fuente: elaboración propia).....	26
Tabla 3.2.2 Descripción de los requisitos funcionales (Fuente: Elaboración propia) .....	28
Tabla 4.2.3 HU Registrar Experimento .....	35
Tabla 5.2.4 HU Seleccionar los datos que se van a utilizar en el experimento.....	36
Tabla 6.2.5 HU Registrar plan de ejecución.....	37
Tabla 7.3.1 Caso de prueba 1.....	47
Tabla 8.3.2 Caso de prueba 2.....	47
Tabla 9.3.3 Caso de prueba 3.....	48
Tabla 10.2.6 HU Modificar Experimento.....	60
Tabla 11.2.7 HU Eliminar Experimento .....	61
Tabla 12.2.8 HU Mostrar datos de consultas select. ....	62
Tabla 13.2.9 HU Mostrar resultados del plan de ejecución. ....	63
Tabla 14.2.10 HU Exportar ficheros CSV.....	64
Tabla 15.2.11 HU Cargar ficheros CSV. ....	65
Tabla 16.2.12 HU Modificar plan de ejecución.....	66
Tabla 17.2.13 HU Eliminar plan de ejecución. ....	68
Tabla 18.2.14 HU Búsqueda avanzada .....	69
Tabla 19.3.1 Caso de prueba 4.....	71
Tabla 20.3.2 Caso de prueba 5.....	72
Tabla 21.3.3 Caso de prueba 6.....	72
Tabla 22.3.4 Caso de prueba 7.....	72
Tabla 23.3.5 Caso de prueba 8.....	73
Tabla 24.3.6 Caso de prueba 9.....	74
Tabla 25.3.7 Caso de prueba 10.....	74
Tabla 26.3.8 Caso de prueba 11.....	74
Tabla 27.3.9 Caso de prueba 12.....	75
Tabla 28.3.10 Caso de prueba 13.....	75
Tabla 29.3.11 Caso de prueba 14.....	76
Tabla 30.3.12 Caso de prueba 15.....	76
Tabla 31.3.13 Caso de prueba 16.....	77
Tabla 32.3.14 Caso de prueba 17.....	77
Tabla 33.3.15 Caso de prueba 18.....	78

Tabla 34.3.16 Caso de prueba 19..... 78

### **Introducción**

El notable desarrollo de la tecnología de la información en el mercado mundial, unido al desarrollo de grandes organizaciones que consumen y producen grandes cantidades de datos, ha implicado la disponibilidad de una gran cantidad de información, en general de fuentes como bases de datos. La interpretación correcta de estos datos es una limitación aún latente en muchas empresas y organizaciones que afecta al proceso de toma de decisiones. Un escenario particular de esta situación lo constituyen las organizaciones que trabajan en la gestión de proyectos.

La gestión de proyectos es importante dentro del desarrollo sostenible y constante de la sociedad, ayuda a visualizar un horizonte de posibilidades en un escenario determinado, para tomar la mejor decisión posible. En el país actualmente hay varios sectores de desarrollo como empresas, organizaciones, entidades, compañías, centros de estudios que realizan programas para la gestión de diferentes proyectos, necesarios para la comunidad. Estos sectores desarrollan proyectos para la salud, la tecnología, la ciencia, la informática, la medicina, entre otros y en general para todos los procesos donde existan actividades que requieran ser automatizadas.

Uno de los centros de estudio basado en el concepto de universidad productiva que tiene como misión fundamental producir aplicaciones y servicios informáticos basados en la gestión de proyectos es la Universidad de las Ciencias Informáticas (UCI), en la cual funcionan 15 centros que desarrollan actividades de investigación, desarrollo e innovación (I+D+I), encargados de las aplicaciones informáticas, el desarrollo tecnológico y las investigaciones asociadas. De conjunto conforman una red de trabajo colaborativo que opera bajo normas y procedimientos comunes, posibilitando la reutilización de componentes y eficiencia industrial, además disponen de laboratorios, a los cuales se vinculan los estudiantes como parte del modelo de formación - producción - investigación puesto en práctica en la Universidad.

Uno de los centros de desarrollo es el Centro de Consultoría y Desarrollo de Arquitecturas Empresariales (CDAE) el cual ofrece servicios de consultoría a organizaciones que ejecutan iniciativas de interoperabilidad e integración de sistemas empleando Arquitecturas Orientadas a Servicios. De esta forma contribuyen a optimizar procesos de negocios y elevar la eficiencia operacional.

Entre sus funciones se encuentra:

- Desarrollar investigaciones asociadas a las Arquitecturas Empresariales, Interoperabilidad, Arquitecturas Orientadas a Servicios y Gestión de Procesos de Negocio.
- Desarrollar soluciones basadas en Arquitecturas Orientadas a Servicios favoreciendo el uso de tecnologías libres.
- Desarrollar un sistema de formación y certificación que satisfaga las áreas de conocimiento que aborda el centro.

Se hizo necesidad optimizar las diferentes áreas del conocimiento por lo que se crearon las herramientas para la gestión de proyectos y ya en el 2010 se crea la plataforma de gestión de proyectos XEDRO GESPRO. Esta plataforma se presenta como un modelo de negocios basado en servicios donde se combina el uso de una solución informática y un sistema de formación especializada en gestión de proyectos. Esta combinación posibilita no sólo la informatización de la gestión de proyectos en las organizaciones, sino también la mejora continua de sus procesos de planificación, seguimiento y control, el mismo es desarrollado por el Laboratorio de Investigaciones en gestión de proyectos de la Universidad de las Ciencias Informáticas de Cuba.

Actualmente la herramienta XEDRO-GESPRO se encuentra integrada con herramientas comercializadas bajo licencia de software libre, que incluyen módulos para la gestión documental, de alcance, de tiempo, de riesgos, de la calidad, de recursos humanos, el control de versiones, entre otras, pero no cuenta con un módulo que haga análisis a partir de datos almacenados en la base de datos de proyectos terminados por la plataforma XEDRO-GESPRO, por lo que empiezan a surgir algunos problemas que hacen que el centro pierda eficiencia en el desempeño laboral. Algunos de ellos son:

- Aunque se cuenta con un gran grupo de datos e información almacenada de los proyectos que se han terminado y de los que aún están en ejecución, no se realiza un análisis detallado de los mismos, afectando el uso de experiencias anteriores para tener éxito en el desarrollo de nuevos proyectos.

- No se mantiene un registro de fácil acceso y estructurado de las evaluaciones realizadas a los proyectos terminados, afectando el estudio de tendencias para predecir futuros problemas en determinadas áreas del proyecto.
- Los datos de los proyectos terminados se guardan en sus bases de datos, pero no se explota la utilización de los mismos afectando que no se pueda realizar una planificación adecuada de los recursos necesarios para iniciar un nuevo proyecto e impide poder predecir una estimación de tiempo y de costo real.
- En general la interpretación correcta de los datos por parte de algunos trabajadores del centro es una limitación aún latente que afecta al proceso de toma de decisiones y selección de estrategias que faciliten el trabajo en nuevos proyectos.

Por todo lo anteriormente expuesto se propone como **problema de investigación**:

¿Cómo realizar el análisis de datos a la información generada por la plataforma XEDRO-GESPRO aplicando la metodología CRISP-DM?

Por lo que se identifica como **objeto de estudio**: Plataformas para el análisis de datos basados en CRISP-DM. Para dar solución al problema de la investigación se establece como **objetivo general**: Desarrollar un módulo para el análisis de datos basado en CRISP-DM que garantice un análisis efectivo para los usuarios, enmarcado en el **campo de acción**: El análisis de datos para la ayuda del desarrollo de nuevos proyectos en la plataforma XEDRO-GESPRO.

Para el cumplimiento de los objetivos trazados, se propone las siguientes tareas de investigación:

1. Diseño del marco teórico referencial asociado a las técnicas y herramientas para el análisis de datos y las facilidades para la integración de estas con la plataforma XEDRO-GESPRO.
2. Definición de los requisitos funcionales y no funcionales para poder identificar las necesidades que debe cubrir el módulo.
3. Descripción de la arquitectura por la cual estará soportada la implementación de las funcionalidades.
4. Desarrollo de un módulo para el análisis de datos basado en CRISP-DM para la plataforma XEDRO-GESPRO.

5. Realización de pruebas al módulo y comprobar la integración del mismo a la plataforma XEDRO-GESPRO.

### **Métodos Teóricos**

**Análisis - Síntesis:** permite analizar las teorías y documentos, permitiendo la extracción de los elementos más importantes que se relacionan con el objeto de estudio, además de la información y documentación estudiada sobre el tema en diferentes fuentes y la obtención de un conocimiento general durante la investigación.

**Histórico - Lógico:** permite estudiar de forma analítica la trayectoria histórica real del análisis de datos, evolución y desarrollo, con el objetivo de investigar teóricamente cómo ha evolucionado en un área determinada, en un período de tiempo, en toda su trayectoria o en un fragmento temporal de la lógica de su desarrollo.

## Capítulo 1. Marco Teórico Referencial

Para el desarrollo de la investigación es necesario hacer un estudio de diferentes conceptos relacionados con la gestión de proyectos, se realiza una revisión y un análisis de trabajos con soluciones semejantes, además de consultar y obtener la bibliografía y otros materiales que sean útiles para los propósitos de la investigación. Se justifican las herramientas, tecnologías y metodologías, así como el framework, lenguaje de programación para dar solución al problema de la investigación. Además, se realiza un estudio de las herramientas que se van a utilizar ya que las mismas están predefinidas por el laboratorio de investigación.

### 1.1 Gestión de proyectos

Es el método para la planificación, organización, colaboración, evaluación y control de recursos necesarios para lograr uno o varios objetivos. Son todas aquellas acciones que debes realizar para cumplir con una necesidad definida dentro de un periodo de tiempo, durante el cual se utilizan recursos, herramientas y personas, que tienen un coste que se ha de tener en cuenta cuando se realiza el presupuesto. La gestión de proyecto se puede aplicar a casi cualquier tipo de proyecto y es ampliamente utilizado para controlar los complejos procesos de los proyectos de desarrollo de software (Rouse, 2005).

Existen cinco grupos de procesos que comprenden el ciclo de vida de la gestión de proyectos (Rouse, 2005):

- ✓ **Iniciación:** Se identifica el objetivo del proyecto, necesidad o problema.
- ✓ **Planificación:** Se planifican todos los pasos necesarios para llegar a una conclusión exitosa del proyecto. Los procesos de planificación de proyectos son de naturaleza iterativa y se espera que la planificación suceda con frecuencia durante todo el proyecto.
- ✓ **Ejecución:** Una vez que se ha creado el plan del proyecto, el equipo busca ejecutar el plan de proyecto para crear los entregables. El proyecto puede pasar por la fase de planificación según sea necesario a lo largo de la ejecución del mismo.

## Capítulo 1. Marco Teórico Referencial

- ✓ **Seguimiento y control:** Conforme el proyecto es ejecutado por el equipo, se monitorea y controla el trabajo para evaluar los factores de tiempo, costo, alcance, calidad, riesgo y otros relacionados.
- ✓ **Cierre:** El cierre del proyecto garantiza que todo el trabajo se ha completado, es aprobado y finalmente se transfiere la propiedad del equipo del proyecto al de operaciones.

Una vez definidos los grupos que comprenden el ciclo de vida de la gestión de proyectos, se pasa a explicar cada una de las áreas de conocimiento al que pertenecen. Estas diez áreas de conocimiento segmentan diferentes acciones realizadas por el director del proyecto a lo largo del proyecto. Las diez áreas de conocimiento de gestión de proyectos son (Rouse, 2005):

**Gestión del alcance del proyecto:** El alcance del proyecto está definido, documentado y aprobado. Este está protegido contra cambios no autorizados, editado con los cambios aprobados, y validado por los interesados en el proyecto para que sea aceptado.

**Gestión del cronograma del proyecto:** El cronograma del proyecto se define en primer lugar por las horas de trabajo, los hitos y en última instancia por la fecha límite del proyecto. La disponibilidad del equipo durante todo el proyecto está documentado y planeado en consecuencia. El director del proyecto trabajará con el equipo para identificar las tareas del proyecto y las estimaciones de duración, con el fin de crear un cronograma o línea de tiempo para el proyecto.

**Gestión de los costos del proyecto:** Se estiman los costos del proyecto de tal modo que se pueda asignar un presupuesto para el mismo. Los costos del proyecto incluyen materiales, servicios, instalaciones, licencias de software y otros gastos imputados directamente al proyecto.

**Gestión de la calidad del proyecto:** El concepto de calidad en el proyecto se define en métricas específicas y es acordado entre las partes interesadas lo antes posible. Los programas y políticas de garantía de calidad dirigen el trabajo del proyecto, mientras que el

## Capítulo 1. Marco Teórico Referencial

control de calidad inspecciona el trabajo del proyecto para confirmar que la calidad ha sido comprobada en el trabajo.

**Gestión de los recursos humanos del proyecto:** El director del proyecto trabaja con el equipo para verificar que cada miembro está completando sus asignaciones, trabajando bien con los demás y que su participación y desempeño es informado a sus respectivos gerentes.

**Gestión de las comunicaciones del proyecto:** Los interesados deberán proporcionar información al director a lo largo del ciclo de vida del proyecto. Esta área de conocimiento crea un plan de gestión de las comunicaciones que indica quién y qué tipo de información, cuándo será necesaria la información y la mejor modalidad para las comunicaciones.

**Gestión de riesgos del proyecto:** Los riesgos son situaciones, eventos, condiciones que pueden amenazar y a veces beneficiar los objetivos del proyecto. Los riesgos deben ser identificados, analizados y se debe crear una respuesta para el evento de riesgo. Se evalúa la probabilidad y el impacto de cada evento de riesgo para crear una puntuación de riesgo para justificar los costos necesarios para gestionar el evento de riesgo en cuestión.

**Gestión de las adquisiciones del proyecto:** Si se necesita adquirir bienes o servicios para el proyecto, será necesario crear un proceso formal de contratación. El plan debe abordar la selección del tipo de contrato, la administración del contrato, las auditorías de compra y la liquidación del contrato del proyecto.

**Gestión de las partes interesadas en el proyecto:** Los accionistas o interesados son cualquier persona que tiene un interés personal en el proyecto. Gestionar los grupos de interés implica la identificación, la inclusión y la comunicación con los grupos de interesados en el proyecto. Aquí se administran las preocupaciones que los grupos de interés puedan tener sobre el trabajo del proyecto.

**Gestión de la integración del proyecto:** Esta área de conocimiento especial es la coordinación de eventos en todas las otras áreas de conocimiento. La gestión de la integración del proyecto examina las interacciones y contingencias entre las áreas de

conocimiento, para asegurar que el proyecto sea adecuadamente planeado, ejecutado, controlado y cerrado.

Es válido destacar que en casi todas las áreas de proyectos se puede realizar análisis de datos ya que todas presentan gran volumen de información.

### **1.2 Análisis de datos y su importancia**

El análisis de datos es un proceso de inspeccionar, limpiar y transformar datos con el objetivo de resaltar información útil, es la ciencia que se encarga de examinar un conjunto de datos con el propósito de sacar conclusiones sobre la información para poder tomar decisiones o simplemente ampliar los conocimientos sobre diversos temas. Este sólo es posible gracias a una buena recolección y organización de datos. Cuanto más fiables sean los datos y mejor organizados estén, mayor es la capacidad de analizarlos y generar información con el potencial de ayudar a una organización. No es una tarea simple y no debe ser realizada solo una vez. Es necesario tener periodicidad en los estudios. Estos datos deben ser monitoreados semanal o mensualmente, lo que constituya la mejor opción para la organización (Ferreira, 2017).

Sin embargo, el análisis de informes e indicadores desarrollados deben ser controlados estrechamente de forma rutinaria. Con esto, se puede comprobar de antemano ciertas situaciones factibles para la creación de nuevos proyectos. Pero esto sólo es posible si estos informes de la empresa se recogieron muy bien o se guardaron en una base de datos especial para eso, si son bien organizados, bien analizados y monitoreados muy de cerca.

Para el análisis de datos se utilizó el proceso de minería de datos: La minería de datos no es más que el proceso de computación para descubrir patrones en grandes conjuntos de datos que involucran métodos en la intersección de aprendizaje automático, estadísticas y sistemas de bases de datos. Es un proceso esencial en el que se aplican métodos inteligentes para extraer patrones de datos. El objetivo general del proceso de minería de datos es extraer información de un conjunto de datos y transformarla en una estructura comprensible para su posterior uso (Arancibia,2016).

En el mundo actual hay otros centros de desarrollo que han utilizado soluciones similares para desarrollar trabajos con requerimientos parecidos al que se quiere desarrollar en estos momentos. Es necesario realizar un análisis de otras fuentes de información para saber técnicas que se podían emplear para el análisis de datos y en cuanto a lenguaje de programación que se utilizó para el desarrollo de la aplicación.

### 1.3 Metodología para el desarrollo de proyectos en minería de datos basados en CRISP-DM

Para la selección de la metodología se buscaron datos relevantes de por qué la seleccionada además de ser predefinida por el centro y tener una gran tendencia en el mundo. A continuación, se muestra una encuesta que se realizó evaluando los porcentos de utilización para cada metodología:

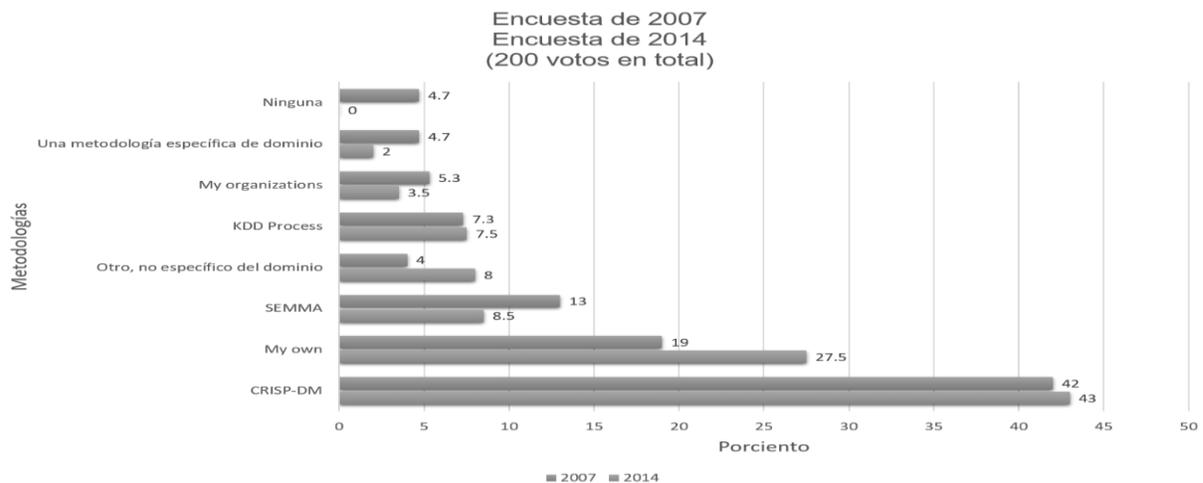


Ilustración 1 Encuesta de metodologías (Fuente: VÁSQUEZ VALLES, 2014).

RISP-DM (*Cross Industry Standard Process for Data Mining*) es la guía de referencia más ampliamente utilizada en el desarrollo de proyectos de *Data Mining*. CRISP-DM, está dividida en 4 niveles de abstracción organizados de forma jerárquica en tareas que van desde el nivel más general hasta los casos más específicos y organiza el desarrollo de un proyecto de *Data Mining*, en una serie de seis fases (Arancibia, 2016).

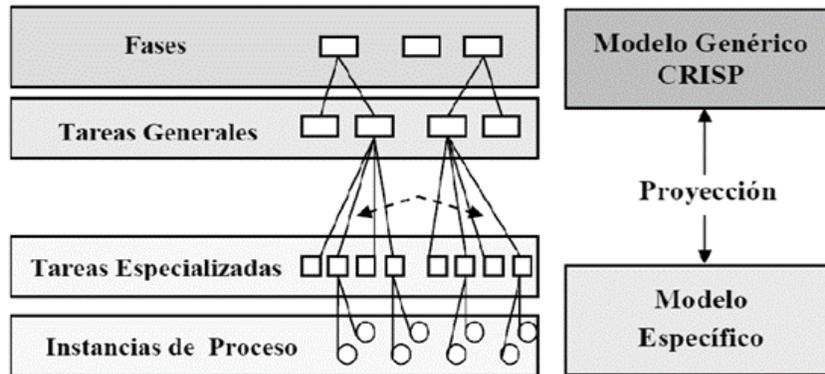


Ilustración 2 Esquema de los cuatro niveles de CRIPS-DM (Fuente: CRIPS-DM, 2000).

La sucesión de fases requiere seguir un orden lógico ya que se necesita que exista dependencia entre las mismas, pero no necesariamente entre todas. Cada fase es estructurada en varias tareas generales de segundo nivel. Las tareas generales se proyectan a tareas específicas, donde finalmente se describen las acciones que deben ser desarrolladas para situaciones específicas.

A continuación, se describen cada una de las fases en que se divide CRISP-DM.

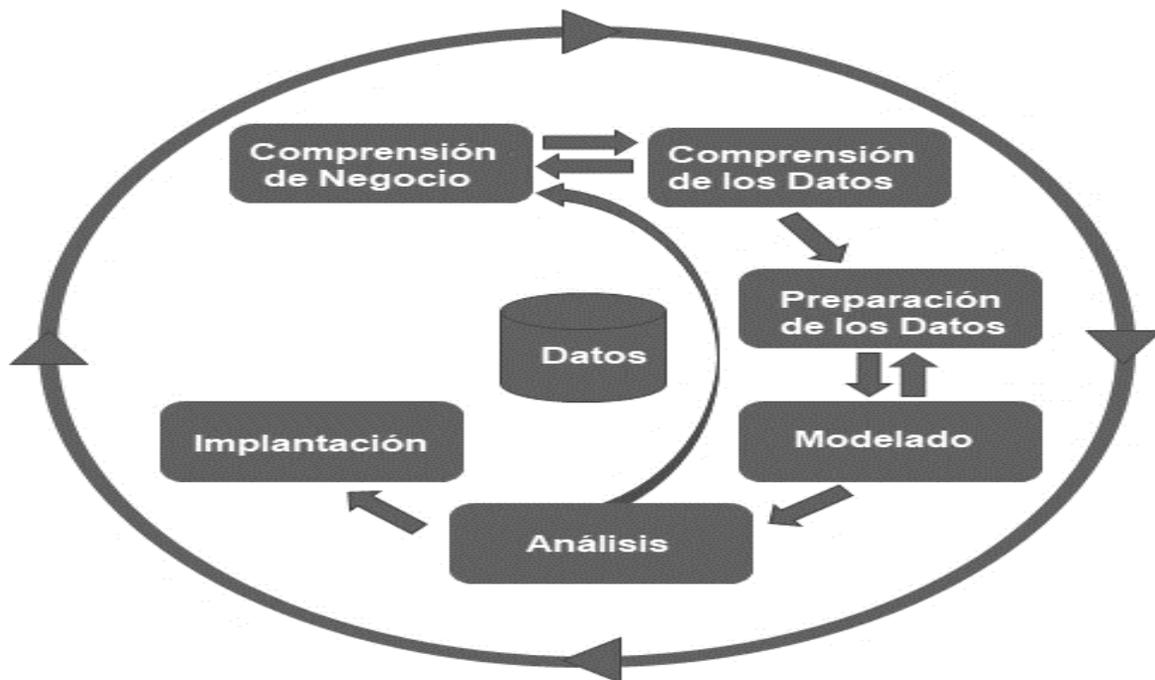


Ilustración 3 Modelo de proceso CRISP-DM (Fuente: Elaboración propia).

**Fase de comprensión del negocio o problema:** la primera fase de la guía de referencia CRISP-DM, contiene las tareas de comprensión de los objetivos y requisitos del proyecto, con el fin de convertirlos en objetivos técnicos y en un plan de proyecto. Sin lograr comprender dichos objetivos, ningún algoritmo por muy sofisticado que sea permitirá obtener resultados fiables. En esta fase, es muy importante la capacidad de poder convertir el conocimiento adquirido del negocio en un problema de *Data Mining* y en un plan preliminar cuya meta sea el alcanzar los objetivos del negocio (Arancibia,2016).

Las principales tareas son:

- ✓ Determinar los objetivos del negocio
- ✓ Evaluación de la situación
- ✓ Determinación de los objetivos de DM

**Fase de comprensión de los datos:** comprende la recolección inicial de datos, con el objetivo de establecer un primer contacto con el problema, familiarizándose con ellos, identificar su calidad y establecer las relaciones más evidentes que permitan definir las primeras hipótesis.

Las principales tareas son:

- ✓ Recolección de datos iniciales
- ✓ Descripción de los datos
- ✓ Exploración de datos
- ✓ Verificación de la calidad de los datos

**Fase de preparación de los datos:** en esta fase y una vez efectuada la recolección inicial de datos, se procede a su preparación para adaptarlos a las técnicas de *Data Mining* que se utilicen posteriormente, tales como: técnicas de visualización de datos, de búsqueda de relaciones entre variables u otras medidas para exploración de los datos.

Las principales tareas son:

- ✓ Selección de datos
- ✓ Limpieza de los datos
- ✓ Estructuración de los datos
- ✓ Producción de un plan del proyecto

**Fase de modelado:** en esta fase de CRISP-DM, se seleccionan las técnicas de modelado más apropiadas para el proyecto de Minería de datos específico. Las técnicas a utilizar en esta fase se eligen en función de los siguientes criterios: ser apropiada al problema, disponer de datos adecuados y cumplir los requisitos del problema.

Las principales tareas son:

- ✓ Selección de la técnica de modelado
- ✓ Generación del plan de prueba
- ✓ Construcción del Modelo
- ✓ Evaluación del modelo

**Fase de evaluación:** en esta fase se evalúa el modelo, teniendo en cuenta el cumplimiento de los criterios de éxito del problema. Debe considerarse, además, que la fiabilidad calculada para el modelo se aplica solamente para los datos sobre los que se realizó el análisis. Considerar que se pueden emplear múltiples herramientas para la interpretación de los resultados (Arancibia,2016).

Las principales tareas son:

- ✓ Evaluación de los resultados
- ✓ Proceso de revisión
- ✓ Determinación de futuras fases

**Fase de implementación:** en esta fase una vez que el modelo ha sido construido y validado, se transforma el conocimiento obtenido en acciones dentro del proceso de negocio. Por otra parte, en la fase de explotación se debe asegurar el mantenimiento de la aplicación y la posible difusión de los resultados (Arancibia,2016).

Las principales tareas son:

- ✓ Plan de implementación
- ✓ Monitorización y Mantenimiento
- ✓ Revisión del proyecto

### 1.4 Plataformas para el análisis de datos

Algunas de las plataformas que trabajan para el análisis de datos son:

**Los sistemas ERP:** contienen grandes cantidades de datos relacionados con la ejecución real de procesos comerciales. Estos sistemas tienen una forma particular de registrar las actividades, lo que da como resultado una visualización poco clara de los procesos de negocios en los registros de eventos. Se han llevado a cabo varios trabajos en sistemas ERP, la mayoría de ellos enfocados en el desarrollo de nuevos algoritmos para el descubrimiento automático de procesos comerciales. Se ha enfocado en abordar problemas como, cómo pueden las organizaciones con sistemas ERP aplicar la minería de procesos para analizar sus procesos comerciales con el fin de mejorarlos. El aspecto de manejo de datos de los sistemas ERP contrasta con los de BPMS o sistemas basados en flujo de trabajo, cuyo almacenamiento sistemático de eventos facilita la aplicación de técnicas de minería de procesos. CRISP-DM se ha convertido en el estándar de facto para desarrollar proyectos de minería de datos y descubrimiento de conocimiento. El éxito de la extracción de datos requiere tres familias de capacidades analíticas, a saber, informes, clasificación y previsión. Un minero de datos utiliza más de un método analítico para obtener los mejores resultados. El objetivo de este documento es mejorar la usabilidad y comprensibilidad de las técnicas de minería de procesos, mediante la implementación de la metodología CRISP-DM para su aplicación en contextos de ERP, detallada en términos de herramientas de implementación específicas y coordinación paso a paso. El estudio confirma que el descubrimiento de datos del sistema ERP mejora la toma de decisiones estratégicas y operativas.

**La CMIN:** es una herramienta CASE (*Computer Aided Software Engineering*) integrada (que soporta todas las fases de un proceso) basada en CRISP-DM 1.0 (*Cross Industry Standard Process for Data Mining*) para soportar el desarrollo de proyectos de minería de datos. Primero se expone la funcionalidad general de CMIN, lo que incluye la gestión de procesos, plantillas y proyectos, y se destaca la capacidad de CMIN para realizar el seguimiento de los proyectos de una forma fácil e intuitiva y la manera como CMIN posibilita que el usuario incremente su conocimiento en el uso de CRISP-DM o de cualquier otro proceso que se defina en la herramienta a través de las ayudas e información que se ofrece en cada paso del proceso. Después, se detalla cómo CMIN permite enlazar en tiempo de ejecución (sin necesidad de volver a compilar la herramienta) nuevos algoritmos de minería de datos que apoyen la labor de modelado (basada en un flujo de trabajo o *workflow*) en un

proyecto de minería de datos. Finalmente, se ofrecen los resultados de dos evaluaciones de la herramienta, las conclusiones y el trabajo futuro.

### **1.4.1 ¿Qué es XEDRO GESPRO?**

Esta Suite de Gestión de Proyectos se presenta como un modelo de negocios basado en servicios donde se combina el uso de una solución informática para la gestión de proyectos y un sistema de formación especializada en gestión de proyectos. Esta combinación posibilita no sólo la informatización de la gestión de proyectos en las organizaciones, sino también la mejora continua de sus procesos de planificación, seguimiento y control (Pestano Pino, 2011).

La solución informática desarrollada bajo un ecosistema de software basado en tecnologías libres, permite:

- ✓ La planificación, el control y seguimiento de los proyectos y de los recursos asociados a los mismos, alineadas con la proyección estratégica de las organizaciones.
- ✓ La planificación del alcance y el tiempo, la gestión de recursos humanos y sus competencias, la gestión de riesgos, así como el financiamiento de los proyectos. Gestión logística y gestión de recursos compartidos.
- ✓ El control y seguimiento de proyectos a través de la combinación de un cuadro de mando integral y un sistema para el diseño dinámico de reportes que permiten el acceso a la información del estado de los proyectos con diferentes niveles de detalles de la información.
- ✓ Gestión documental con facilidades para la gestión del expediente de los proyectos.
- ✓ Gestión de contratos y de interesados en los proyectos, que permite tanto la gestión de acuerdos con clientes como con los proveedores y garantiza integrar el control y el seguimiento de los compromisos alineados completamente con la información del estado de los proyectos.

Los servicios de formación especializada en gestión de proyectos se presentan como un paquete de hasta 20 cursos y sus materiales cubren todas las áreas de conocimiento de la gestión de proyectos y su comercialización; garantizan diferentes niveles de formación personalizables en función de las necesidades en diferentes escenarios. Las actividades

de formación se presentan tanto como cursos independientes como en forma de programas de postgrado que pueden tributar a la obtención de grados científicos de los beneficiados (Pestano Pino, 2011).

### **1.4.2 Surgimiento**

El surgimiento de XEDRO-GESPRO (conocido en sus inicios solamente como GESPRO), tiene lugar en la Universidad de las Ciencias Informáticas. Hacia el año 2008 se planteaba el reto de institucionalizar una única herramienta que permitiera el almacenamiento de los datos correspondientes a la gestión de los proyectos de software que se desarrollaban y la ayuda a la toma de decisiones. En el mes de abril del año 2010, el Laboratorio de Investigaciones en Gestión de Proyectos de la UCI lanzó la primera versión de GESPRO. Desde entonces, se utilizaría la herramienta de gestión de proyectos Redmine (Lang, 2006) como núcleo para su desarrollo de conjunto con otras tecnologías libres y propias de la institución. Gracias a la gran capacidad de extensión de su núcleo, poco a poco se le han continuado añadiendo y/o perfeccionando nuevas funcionalidades mediante el trabajo con plugins (Pestano Pino, 2011).

### **1.4.3 Evolución**

La plataforma GESPRO tiene como su base los estándares PMBoK (PMI, 2013) y la ISO 21500. A partir de la segunda versión (lanzada en mayo del 2011), el equipo de desarrollo acordó realizar los lanzamientos de GESPRO con una frecuencia anual y un mes posterior a la entrega de las distribuciones del sistema operativo Ubuntu, promoviendo de esta manera la reutilización de actualizaciones de seguridad y parches en los servidores de GESPRO que utilizaran esta distribución como anfitrión. Siguiendo el estilo de Canonical, el equipo de desarrollo acordó además etiquetar el número de versiones del producto según el año y mes de su lanzamiento. Así en el mes de mayo del año 2012 salió a la luz GESPRO 12.05 (Pestano Pino, 2011).

### **1.4.4 Características generales del XEDRO GESPRO**

Comprende dos niveles de gestión: a nivel organizacional y a nivel de proyecto.

- En el primero se realiza el control y seguimiento de las sucursales u organizaciones y de los proyectos, pero a nivel gerencial. En el 2do se controla y evalúan los proyectos y los recursos humanos.
- Está compuesto por varios módulos, donde cada uno responde a las diferentes áreas de conocimiento de la gestión de proyectos, tanto a nivel organizacional como a nivel operacional.
- En cada uno de los módulos se recoge información que tributan a los diferentes indicadores que procesa la plataforma para la evaluación de los proyectos. Estos indicadores permiten medir el nivel de rendimiento y aprovechamiento de los recursos humanos, evaluar la logística, la calidad de los datos, la eficiencia y la eficacia de los proyectos, cubriendo las áreas de conocimiento de la gestión de proyectos.
- Un análisis exhaustivo de estos datos podría facilitar la identificación de posibles errores que se cometen durante el proceso de gestión y sus causas, incluso, facilitaría también la planificación de futuros proyectos. Sin embargo, la plataforma carece de este análisis, lo cual le limita al usuario tomar decisiones preventivas y de carácter reactivo.

### **1.5 Herramientas y metodologías de desarrollo**

Es de vital importancia para el correcto desarrollo de un software realizar la selección de cada una de las herramientas a utilizar. Existen diferentes metodologías que han sido en los últimos años herramientas de apoyo para el desarrollo del software, además de lenguajes de programación y gestores de bases de datos que responden a las necesidades de la aplicación a implementar. Se realiza un estudio de las herramientas que directamente se utilizaron ya que las mismas estaban predefinidas por el laboratorio de investigaciones en gestión de proyectos.

#### **1.5.1 Metodologías de desarrollo**

Para la selección de la metodología fue necesario analizar los aspectos con los que cuenta la metodología predeterminada SCRUM y cuáles son las necesidades que cubre de lo que se necesita desarrollar, se analizó que la forma de desarrollo es muy simple y por tal motivo

## Capítulo 1. Marco Teórico Referencial

está dirigida a proyectos con requisitos inestables y que requieren rapidez y flexibilidad. También se tuvo en cuenta que el tipo de producto ya es conocido o es un proyecto que se entiende su naturaleza por el centro.

La misma no cuenta con suficiente documentación desde el punto de vista ingenieril ni con técnicas que sustenten una mejor visibilidad a temas ingenieriles por lo que se hará necesario el uso de las técnicas de otra metodología que cubra esta necesidad y que respalde bibliográficamente las tareas que de ella se realicen.

SCRUM es un modelo de desarrollo ágil caracterizado por adoptar una estrategia de desarrollo incremental, en lugar de la planificación y ejecución completa del producto, basar la calidad del resultado más en el conocimiento tácito de las personas en equipos auto-organizados, que, en la calidad de los procesos empleados, solapamiento de las diferentes fases del desarrollo, en lugar de realizar una tras otra en un ciclo secuencial o de cascada (Palacio, 2014). Los procesos de Scrum abordan las actividades y el flujo específico de un proyecto Scrum. En total hay diecinueve procesos que se agrupan en cinco fases (Satpathy, 2016).

Tabla 1.1 Fases de SCRUM (Fuente: Satpathy, 2016).

Fase	Procesos
Inicio	Creación de la visión del proyecto Identificación del Scrum Master y el(los) socio(s) Formación de equipos Scrum Desarrollo de épica(s) Creación de la lista priorizada de pendientes del producto Realizar la planificación de lanzamiento
Planificación y estimación	Creación de historias de usuario Aprobación, estimación y asignación de historias de usuario Creación de tareas Estimación de tareas

## Capítulo 1. Marco Teórico Referencial

	Creación de la lista de pendientes del sprint
Implementación	Creación de entregables Llevar a cabo la reunión diaria Mantenimiento de la lista priorizada de pendientes del producto
Revisión y retrospectiva	Convocar el Scrum de Scrums Demostración y validación del sprint Retrospectiva del sprint
Lanzamiento	Envío de entregables Retrospectiva del proyecto

Esta metodología se utilizara esencialmente en la etapa de planificación y estimación donde se comprenden las tareas como (Satpathy, 2016):

- Creación de historias de usuario—En este proceso, se crean las historias de usuario y los criterios de aceptación de las historias de usuario. Las historias de usuario son generalmente escritas por el propietario del producto, y están diseñadas para asegurar que los requisitos del cliente estén claramente representados y puedan ser plenamente comprendidos por todos los socios. Se pudieran llevar a cabo ejercicios de escritura de historias de usuario, lo cual involucra a los miembros del equipo Scrum, resultando en la creación de dichas historias. Estas se incorporan a la lista priorizada de pendientes del producto.
- Creación de la lista de pendientes del sprint—En este proceso, el equipo principal de Scrum lleva a cabo una reunión de planificación del sprint donde el grupo crea una lista priorizada de pendientes del Sprint, que contiene todas las tareas que deben completarse en el sprint. Esta tarea se hizo necesaria ya que cada semana

se requiere entregar al cliente y visualizar por parte del cliente un sprint que sería una versión de la funcionalidad que se está desarrollando o de lo que se está implementando.

- Creación de entregables—En este proceso, el equipo Scrum trabaja en las tareas de la lista priorizada de pendientes del sprint para crear los entregables del sprint. Se utiliza a menudo un tablero de Scrum para realizar el seguimiento del trabajo y las actividades que se realizan.

Se decide utilizar la metodología XP ya que al igual que SCRUM es una metodología ágil que también cuenta con el escenario de historias de usuarios y que además presenta técnicas que sustentan la visibilidad ingenieril que hace falta para el desarrollo de la solución con las que no cuenta SCRUM.

XP (Extreme Programming) es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos muy cambiantes (JOSKOWICZ, 2008).

### 1.5.2 Herramienta de modelado

La herramienta para el modelado de la propuesta de solución es Visual Paradigm, debido fundamentalmente a que es una herramienta multiplataforma que ayuda a una rápida construcción de aplicaciones de calidad. A continuación, se brindan detalles de esta herramienta.

Visual Paradigm (VP): La misma propicia un conjunto de ayudas para el desarrollo de programas informáticos, desde la planificación, pasando por el análisis y el diseño, hasta la generación del código fuente de los programas y la documentación. Es decir, Visual Paradigm ha sido concebida para soportar el ciclo de vida completo del proceso de desarrollo del software a través de la representación de todo tipo de diagramas. Los autores han seleccionado esta herramienta debido a que se caracteriza por (Nazar, 2014):

- ✓ Disponibilidad en múltiples plataformas (Windows, Linux).

- ✓ Diseño centrado en casos de uso y enfocado al negocio que generan un software de mayor calidad.
- ✓ Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación.
- ✓ Capacidades de ingeniería directa e inversa.
- ✓ Licencia gratuita y comercial.
- ✓ Varios idiomas.
- ✓ Fácil de instalar y actualizar.
- ✓ Compatibilidad entre ediciones.
- ✓ Diagramas de flujo de datos.
- ✓ Generación de bases de datos, transformación de diagramas de Entidad-Relación en tablas de base de datos.

### **1.5.3 Gestor de base de datos**

PostgreSQL en su versión 9.5, es un sistema de gestión de bases de datos relacionales de objetos basados en POSTGRES, desarrollado en el Departamento de Ciencias de la Computación de la Universidad de California en Berkeley. PostgreSQL es un descendiente de fuente abierta de este código original de Berkeley. Es compatible con una gran parte del estándar SQL y ofrece muchas características modernas (Escuela politécnica Internacional, 2012):

- ✓ consultas complejas
- ✓ llaves extranjeras
- ✓ desencadenantes
- ✓ vistas actualizables
- ✓ integridad transaccional
- ✓ control de concurrencia multiversión

Además, PostgreSQL puede ser extendido por el usuario de muchas maneras, por ejemplo, agregando nuevos tipos de datos, función, operadores, funciones agregadas, métodos de índice, lenguajes de procedimiento.

### 1.5.4 Entorno de desarrollo integrado

RubyMine en su versión 2017.3 es una herramienta para la gestión de proyectos, que con sus diversas funcionalidades permite a los usuarios de diferentes proyectos realizar el seguimiento y organización de los mismos. Además, es posible optimizar su funcionamiento agregando funcionalidades. Incluye un sistema de seguimiento de incidentes con seguimiento de errores. Otras herramientas que incluye son calendario de actividades, diagramas de Gantt para la representación visual de la línea del tiempo de los proyectos, foro, visor del repositorio de control de versiones, control de flujo de trabajo basado en roles, integración con correo electrónico, entre otras opciones. Está escrito usando el framework Ruby on Rails. Es software libre y de código abierto, disponible bajo la Licencia Pública General de GNU v2 (Jetbrains, 2017).

#### Características

- ✓ Soporta múltiples proyectos.
- ✓ Soporta múltiples usuarios.
- ✓ Roles flexibles basados en control de acceso.
- ✓ Sistema de seguimiento de errores flexible.
- ✓ Diagramas de Gantt y calendario.
- ✓ Administración de noticias, documentos y archivos.
- ✓ Fuentes web y notificaciones por correo electrónico.
- ✓ Creación de peticiones por correo.
- ✓ Soporte para autenticación con el protocolo ligero de acceso a directorios LDAP.
- ✓ Soporta múltiples plataformas (GNU/Linux, Microsoft Linux, Mac OSX).
- ✓ Soporta diferentes bases de datos (MySQL, PostgreSQL y SQLite).

### 1.5.5 Lenguaje de programación

Ruby en su versión 2.3.1 es un lenguaje de programación interpretado, reflexivo y orientado a objetos, creado por el programador japonés Yukihiro "Matz" Matsumoto, es un lenguaje de programación interpretado en una sola pasada y su implementación oficial es distribuida bajo una licencia de software libre. Está orientado a objetos:

Todos los tipos de datos son un objeto, incluidas las clases y tipos que otros lenguajes definen como primitivas, (como enteros, booleanos, y null). Toda función es un método. Las variables siempre son referencias a objetos, no los objetos mismos. Ruby soporta herencia con enlace dinámico, mixins y métodos singleton (pertenecientes y definidos por una sola instancia más que definidos por la clase). A pesar de que Ruby no soporta herencia múltiple, las clases pueden importar módulos como mixins (BYKBAEV, Ingenius, 2008).

### **Características:**

- ✓ Orientado a objetos.
- ✓ Cuatro niveles de ámbito de variable: global, clase, instancia y local.
- ✓ Manejo de excepciones.
- ✓ Iteradores o clausuras (pasando bloques de código).
- ✓ Posibilidad de redefinir los operadores (sobrecarga de operadores).
- ✓ Recolección de basura automática.
- ✓ Altamente portable.
- ✓ Hilos de ejecución simultáneos en todas las plataformas usando hilos verdes, o no gestionados por el sistema operativo.
- ✓ Carga dinámica de DLL/bibliotecas compartidas en la mayoría de las plataformas.
- ✓ Introspección, reflexión y meta programación.
- ✓ Amplia librería estándar.
- ✓ Soporta inyección de dependencias.
- ✓ Soporta alteración de objetos en tiempo de ejecución.
- ✓ continuaciones y generadores.

### **1.5.6 Framework**

Ruby on Rails es un framework de aplicaciones web de código abierto el cual estamos utilizando en su versión 4.2.7 escrito en el lenguaje de programación Ruby, siguiendo el paradigma del patrón Modelo Vista Controlador (MVC). Trata de combinar la simplicidad con la posibilidad de desarrollar aplicaciones del mundo real escribiendo menos código que con otros frameworks y con un mínimo de configuración. El lenguaje de programación Ruby permite la meta programación, de la cual Rails hace uso, lo que resulta en una sintaxis que

muchos de sus usuarios encuentran muy legible. Rails se distribuye a través de RubyGems, que es el formato oficial de paquete y canal de distribución de bibliotecas y aplicaciones Ruby (PERALES, 2014).

### **1.5.7 Estilo CSS**

Bootstrap 3, es un framework originalmente creado por Twitter, que permite crear interfaces web con CSS y JavaScript, cuya particularidad es la de adaptar la interfaz del sitio web al tamaño del dispositivo en que se visualice. Es decir, el sitio web se adapta automáticamente al tamaño de una PC, una Tablet u otro dispositivo.

Aun ofreciendo todas las posibilidades que ofrece Bootstrap a la hora de crear interfaces web, los diseños creados con Bootstrap son simples, limpios e intuitivos, esto le da agilidad a la hora de cargar y al adaptarse a otros dispositivos. El Framework trae varios elementos con estilos predefinidos fáciles de configurar: Botones, Menús desplegables, Formularios incluyendo todos sus elementos de integración jQuery para ofrecer ventanas y tooltips dinámicos (SCHULZ, 2008).

### **1.6 Conclusiones Parciales**

- La gestión de proyectos sirve de gran ayuda para el desarrollo de aplicaciones que se desarrollen para optimizar varios procesos dentro y fuera de un centro o área de producción siendo una vía cómoda para planificar el trabajo dentro de las organizaciones que lo utilicen.
- El análisis de datos es una forma muy recomendable para tomar las mejores decisiones en un centro de desarrollo ya que mediante sus prácticas se puede ver el estado de los proyectos vigentes y no vigentes, permite de aquellos proyectos que sean vigentes saber si su desarrollo va por buen camino o está atrasado ya que se puede medir a través de indicadores y estados que indican su progreso.
- Los principales conceptos relacionados a la investigación, brinda un mayor conocimiento de la solución. Las herramientas y tecnologías a utilizar para la realización de la aplicación, serán aquellas que se tienen planificadas por políticas de restricción del centro de investigación de manera que se logre insertar el módulo sin problemas y que faciliten una mejor planificación de futuros proyectos.

## **Capítulo 2. Características y diseño del sistema**

En el presente capítulo se realiza la descripción de la propuesta de solución, se capturan los requisitos funcionales que detallan las funcionalidades del módulo y los no funcionales que son restricciones que debe cumplir el sistema, además se explica el uso de los patrones de diseño y el patrón arquitectónico que se utilizan para la implementación.

### **2.1 Descripción de la propuesta de solución**

La propuesta de solución consiste en crear un módulo para la plataforma XEDRO-GESPRO que permita realizar el análisis de datos basado en la metodología CRIPS-DM tanto para proyectos que estén en ejecución como para proyectos que ya hayan sido terminados, básicamente lograr la informatización de la metodología. El módulo informático permitirá crear experimentos a los que se le puede realizar análisis y seleccionar solo aquellos datos que le sean necesarios al usuario de proyectos ya estén terminados o aun en ejecución almacenados en bases de datos, estas solo pueden modificarse por administradores, también permite almacenar consultas en tablas que fueron marcadas, además de ver la estructura real de cada una y los valores de los datos que se almacenan en ellas, otra de las funciones del módulo es que se pueden utilizar los experimentos que se crearon, construir consultas más complejas y guardarlas, aunque estas consultas solo pueden recuperar datos y exportar a ficheros csv, este a su vez permite realizar planes de ejecución de algoritmos a dicho fichero csv para un mejor entendimiento de la información seleccionada, quedando demostrado de forma visual en los resultados de dicho plan de ejecución y en un final genera soluciones q pueden ser utilizadas por los usuarios para desarrollar nuevos proyectos.

Se diseñó una arquitectura para facilitarle al usuario un mejor entendimiento del funcionamiento del módulo.

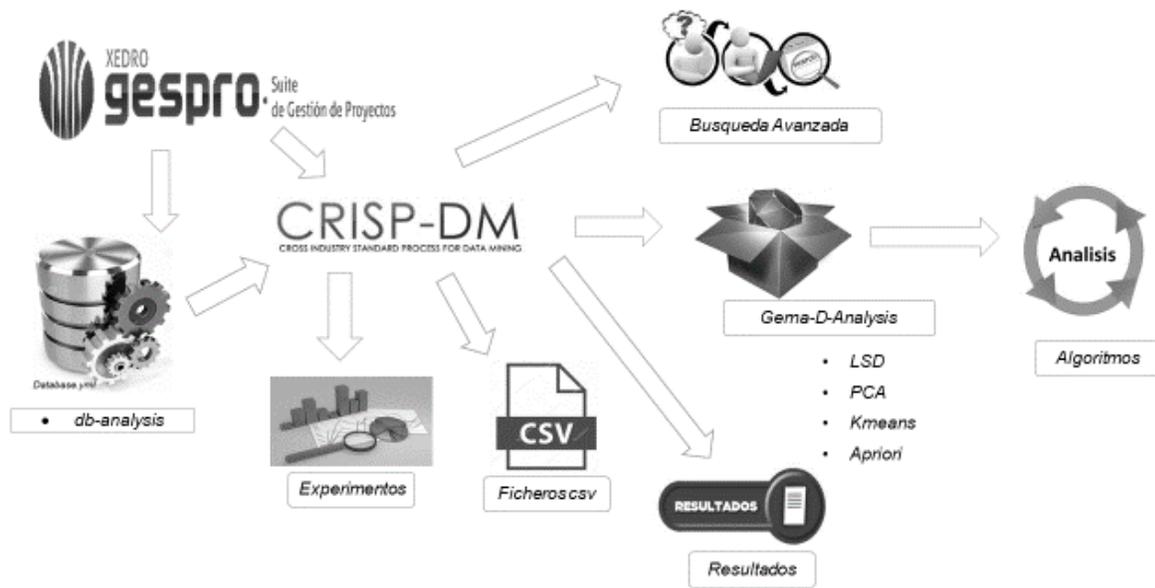


Ilustración 4 Vista Arquitectónica del Módulo.

## 2.2 Modelo de Dominio

El modelo de dominio se crea con el fin de representar el vocabulario y los conceptos claves del dominio del problema. Identifica las relaciones y atributos entre todas las entidades comprendidas en el dominio del problema. Este a su vez proporciona una visión estructural del dominio que puede ser complementado con otros puntos de vista dinámicos (Thames, 2011).

A la hora de obtener una mejor idea del análisis del problema de investigación se confecciona un modelo de dominio con el fin de mostrar a los usuarios de manera visual los principales conceptos que intervienen en el negocio y establece un lenguaje natural y común entre desarrolladores, clientes e interesados.

## Capítulo 2. Características y diseño del sistema.

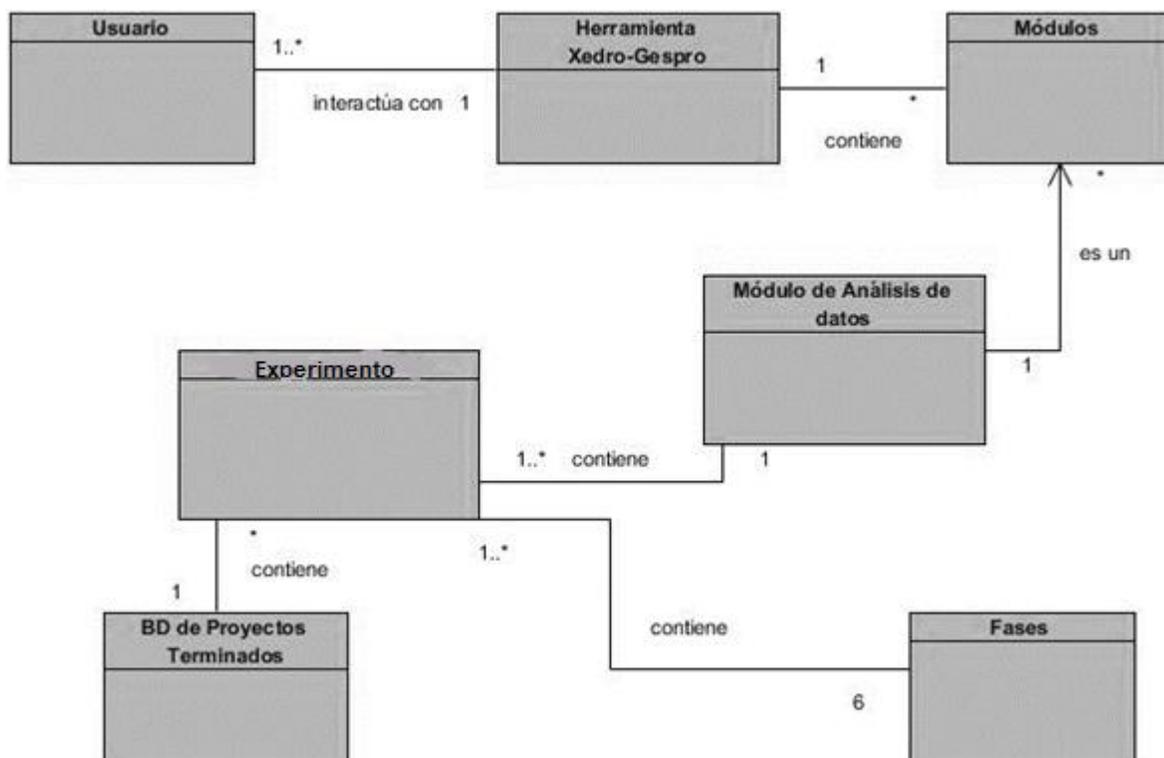


Ilustración 5 Modelo de dominio (Fuente: elaboración propia).

Tabla 2.2.1 Descripción de entidades (Fuente: elaboración propia).

Entidad	Descripción
Usuario	Miembro de un proyecto determinado que en dependencia del rol que desempeñe y/o los permisos asociados al mismo en el módulo, puede utilizar las funcionalidades del mismo.
Herramienta XEDRO-GESPRO	Herramienta para la gestión de proyecto en la que están incluidos varios módulos.
Módulos	Módulo del software que agrupa el conjunto de subprogramas y estructuras de datos.

## Capítulo 2. Características y diseño del sistema.

Módulo de análisis de datos	Permite analizar los datos de cada proyecto terminado o no terminado que sean de interés para un usuario determinado
Proyectos de Análisis	Conjunto de proyectos terminados.
BD de Proyectos Terminados	Contiene los proyectos que ya no están en ejecución y sus datos fueron guardados en una base datos.
Fases:	Conjunto de seis pasos lógicos que deben realizarse en todos los proyectos que pertenezcan al módulo de análisis de datos.

### 2.3 Especificación de los requisitos del sistema

Los requisitos para un sistema son la descripción de los servicios proporcionados por el sistema y sus restricciones operativas. Estos requisitos reflejan la necesidad de los clientes, para que un sistema ayude a resolver un determinado problema (Henao, 2011).

Los mismos se refieren a la tarea de reunir una descripción del producto desde el punto de vista de negocio y es un paso importante para el proceso de desarrollo de un buen producto de software que debe tener en cuenta las exigencias por parte del cliente ya que estos suelen tener una noción de lo que quieren, una idea del resultado final; pero no de las funciones que deben llevarse a cabo para que su software sea como lo pensaron. Realizar con éxito este proceso de descripción de requerimientos ayuda a los ingenieros a comprender el problema y así desarrollar una solución que satisfaga las necesidades de los interesados. Los requisitos se pueden clasificar en dos grupos:

## Capítulo 2. Características y diseño del sistema.

### 2.3.1 Requisitos funcionales (RF)

Estos describen las capacidades o funciones que el sistema debe cumplir y los servicios que de él se esperan. Luego de un estudio de las necesidades que presenta el sistema en cuanto al análisis de datos, se identificaron los siguientes RF:

Tabla 3.2.2 Descripción de los requisitos funcionales (Fuente: Elaboración propia)

Nº	Nombre	Descripción	Complejidad	Prioridad
RF1	Registrar Experimento.	Cuando el usuario selecciona la opción nuevo experimento se llenan los campos nombre y descripción y se presiona crear.	Alta	Alta
RF2	Modificar Experimento	El usuario selecciona el experimento a modificar, se modifica el campo que se desee y se presiona modificar.	Alta	Alta

## Capítulo 2. Características y diseño del sistema.

RF3	Eliminar Experimento.	El usuario selecciona el experimento a eliminar y luego se muestra un mensaje preguntando si está seguro de borrar el experimento y se presiona aceptar.	Alta	Media
RF4	Seleccionar los datos que se van a utilizar en el experimento.	El usuario selecciona los datos con los que quiere crear el experimento.	Baja	Alta
RF5	Mostrar datos de consultas select.	El usuario debe acceder a la consola para ejecutar la consulta select y luego se muestran los resultados de la ejecución de la consulta la cual se exporta en fichero csv.	Baja	Media

## Capítulo 2. Características y diseño del sistema.

RF6	Registrar plan de ejecución.	El usuario selecciona el botón crear plan de ejecución, se llenan los campos obligatorios y se ejecuta el algoritmo seleccionado.	Alta	Media
RF7	Mostrar resultados del plan de ejecución.	Luego de haberse creado el plan de ejecución se muestran los resultados del mismo.	Baja	Baja
RF8	Exportar ficheros CSV.	Luego de haber seleccionado los datos y ejecutar la consulta, se exporta el fichero csv.	Media	Alta
RF9	Cargar ficheros CSV.	Después de crear el plan de ejecución se carga el fichero csv guardado	Media	Alta

## Capítulo 2. Características y diseño del sistema.

		en una dirección específica sin tener que seleccionar los datos ni ejecutar la consulta select.		
RF10	Modificar plan de ejecución	El usuario selecciona el botón modificar plan de ejecución, se modifican los campos que se deseen y se selecciona el botón modificar.	Alta	Media
RF11	Eliminar plan de ejecución	El usuario selecciona el botón eliminar plan de ejecución y se muestra un mensaje indicando que si está seguro de borrar y se	Alta	Media

## Capítulo 2. Características y diseño del sistema.

		presiona aceptar.		
RF12	Buscar soluciones	El usuario escribe una palabra sobre lo que desea buscar, selecciona el botón aceptar y se muestra un listado de los experimentos y/o planes de ejecución relacionados.	Media	Media

### 2.3.2 Requisitos no funcionales (RNF)

Los RNF surgen de la necesidad del cliente debido a las restricciones en el presupuesto, las políticas de la organización, la necesidad de interoperabilidad con otros sistemas de software o hardware o a factores externos como los reglamentos de seguridad, las políticas de privacidad, entre otros. Los mismos forman una parte significativa de la especificación. Son importantes para que clientes y usuarios puedan valorar las características no funcionales del producto.

Son aquellos requerimientos que se refieren a las propiedades del sistema, como el tiempo de respuesta, la capacidad de almacenamiento y otros aspectos como el diseño, la seguridad entre otros. De forma alternativa, definen las restricciones del sistema como la capacidad entrada/salida y las representaciones de datos que se utilizan en las interfaces del sistema (Henao, 2011).

#### RnF1. Requisito de Usabilidad

## Capítulo 2. Características y diseño del sistema.

**RnF1.2.** Facilidad de uso por parte de los usuarios: el módulo debe presentar una interfaz amigable que permita la fácil interacción con el mismo y llegar de manera rápida y efectiva a la información buscada. Debe, además, ser una interfaz de manejo cómodo que posibilite a los usuarios sin experiencia una rápida adaptación.

### **RnF2. Requisito de Interfaz y Apariencia externa**

**RnF2.1** La interfaz debe ser sencilla con colores suaves a la vista y sin muchas imágenes u objetos que distraigan al cliente del objetivo.

**RnF2 2** Debe utilizar como idioma principal el español, aunque debe existir traducción al inglés.

**RnF2.3** Los botones deben expresar su función ya sea mediante su texto o la imagen que lo acompañe.

### **RnF3. Requisito de Seguridad**

**RnF3.1** El sistema debe estar disponible las 24 horas, los 365 días del año, para garantizar la ejecución de las tareas en el momento requerido.

**RnF3.2** Permitir el acceso al sistema a través de una cuenta de usuario única que puede ser local.

**RnF3.3** Los usuarios creados en la Suite GESPRO tienen acceso (o permiso) limitado, pueden acceder a un conjunto de funcionalidades según el rol o perfil donde fue clasificado el usuario.

### **RnF4. Requisito de software**

**RnF4.1** PC cliente: debe tener un navegador web, preferentemente Mozilla Firefox o Google Chrome.

## Capítulo 2. Características y diseño del sistema.

**RnF4.2** PC servidor de base de datos: El servidor debe contar con sistema operativo GNU/Linux Ubuntu 16.04 y el SGDB PostgreSQL 9.5.

### **RnF5 Hardware**

**RnF5.1** Servidor Aplicaciones Web: 2GB RAM, 250 GB disco duro.

**RnF5.2** PC cliente: prestaciones de hardware que permitan la instalación de cualquier Sistema Operativo, conexión de red.

### **RnF6. Restricción del diseño y la implementación**

**RnF6.1** IDE de desarrollo: RubyMine.

**RnF6.2** Paradigma de la arquitectura Modelo-Vista-Controlador (MVC).

**RnF6.3** Lenguaje de programación: Ruby.

**RnF6.4.** Visual Paradigm como herramienta para el modelado.

**RnF6.5** Rails 4.2.7 como framework para la implementación de la propuesta de solución.

**RnF6.6** Metodologías SCRUM y XP.

## **2.4 Historias de Usuario**

Una historia de usuario es una representación de un requisito escrito en una o dos frases utilizando el lenguaje común del usuario. Las historias de usuario son utilizadas en las metodologías de desarrollo ágiles para la especificación de requisitos (acompañadas de las discusiones con los usuarios y las pruebas de validación). Cada historia de usuario debe ser limitada. (Wesley, 2004).

Tabla 4.2.3 HU Registrar Experimento

HU Registrar Experimento.																					
<b>Número:</b> 1	<b>Nombre del requisito:</b> Registrar Experimento.																				
<b>Programador:</b> Yem Liyanis Tamayo Cárdenas  Pedro Alejandro Villavicencio Martínez	<b>Iteración Asignada:</b> 1																				
<b>Prioridad:</b> Alta	<b>Tiempo Estimado:</b> 8 días.																				
<b>Riesgo en Desarrollo:</b> N/A	<b>Tiempo Real:</b> 7 días.																				
<p><b>Descripción</b></p> <p>Cuando el usuario selecciona la opción nuevo experimento se llenan los campos nombre y descripción y se presiona crear.</p> <p>Para ello el módulo debe contar con la opción de crear experimento y debe existir al menos un proyecto.</p>																					
<p><b>Observaciones:</b> Debe existir al menos un proyecto en el módulo.</p>																					
<p><b>Prototipo de interfaz:</b></p>  <p>The screenshot shows a web application interface. At the top, there is a navigation menu with several tabs: 'Comprensión del negocio', 'Comprensión de datos', 'Preprocesamiento de datos', 'Modelación', 'Análisis', and 'Toma decisiones'. The 'Comprensión del negocio' tab is active. Below the navigation menu, there is a sub-menu for 'Comprensión del negocio' with two options: 'Nuevo experimento' and 'Ayuda'. The main content area displays a table with the following columns: 'NOMBRE', 'DESCRIPCION', and 'SQL PARA DATOS'. The table contains three rows of data:</p> <table border="1"> <thead> <tr> <th>NOMBRE</th> <th>DESCRIPCION</th> <th>SQL PARA DATOS</th> <th></th> <th></th> </tr> </thead> <tbody> <tr> <td>prueba1</td> <td>1</td> <td>SELECT "projects"."id" FROM "projects"</td> <td>Modificar</td> <td>Borrar</td> </tr> <tr> <td>prueba3</td> <td>asd</td> <td>SELECT attachments.id, attachments.author_id FROM attachments</td> <td>Modificar</td> <td>Borrar</td> </tr> <tr> <td>prueba2</td> <td>Prueba 2</td> <td>SELECT attachments.id, attachments.author_id FROM attachments</td> <td>Modificar</td> <td>Borrar</td> </tr> </tbody> </table> <p>At the bottom of the page, there is a footer with the following text: 'Project Management Suite (GESPRO 18.01). Research Laboratory in Project Management, UCI. Powered by GESPRO, PostgreSQL, D3, Redmine © 2006-2017 maestrnago@uci.cu, gestion.proyectos@uci.cu, ppp@uci.cu'.</p>		NOMBRE	DESCRIPCION	SQL PARA DATOS			prueba1	1	SELECT "projects"."id" FROM "projects"	Modificar	Borrar	prueba3	asd	SELECT attachments.id, attachments.author_id FROM attachments	Modificar	Borrar	prueba2	Prueba 2	SELECT attachments.id, attachments.author_id FROM attachments	Modificar	Borrar
NOMBRE	DESCRIPCION	SQL PARA DATOS																			
prueba1	1	SELECT "projects"."id" FROM "projects"	Modificar	Borrar																	
prueba3	asd	SELECT attachments.id, attachments.author_id FROM attachments	Modificar	Borrar																	
prueba2	Prueba 2	SELECT attachments.id, attachments.author_id FROM attachments	Modificar	Borrar																	

Tabla 5.2.4 HU Seleccionar los datos que se van a utilizar en el experimento.

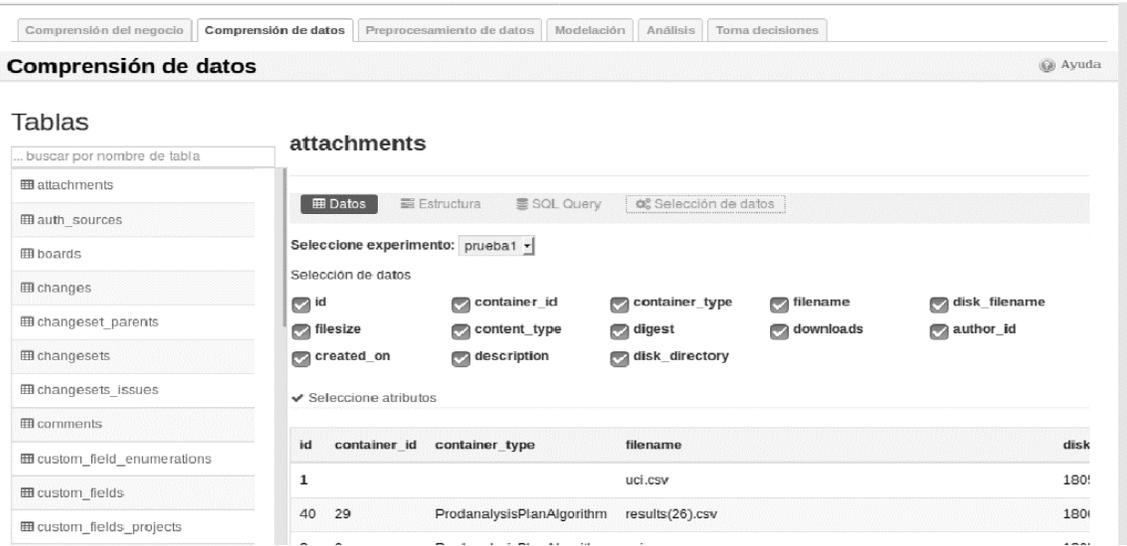
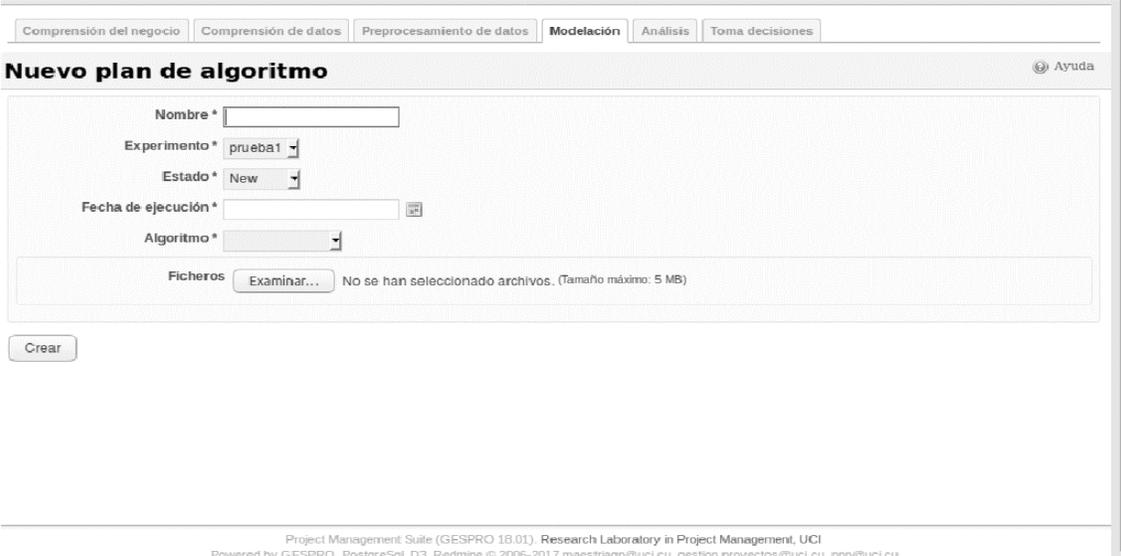
HU Seleccionar los datos que se van a utilizar en el experimento.																
<b>Número:</b> 4	<b>Nombre del requisito:</b> Seleccionar los datos que se van a utilizar en el experimento.															
<b>Programador:</b> Yem Liyanis Tamayo Cárdenas Pedro Alejandro Villavicencio Martínez	<b>Iteración Asignada:</b> 1															
<b>Prioridad:</b> Alta	<b>Tiempo Estimado:</b> 4 días.															
<b>Riesgo en Desarrollo:</b> N/A	<b>Tiempo Real:</b> 4 días.															
<p><b>Descripción</b></p> <p>El usuario selecciona los datos con los que quiere crear el experimento.</p> <p>Para ello debe existir en el módulo al menos un proyecto en el que se guarden las tablas que contienen la información de los datos en la base de datos.</p>																
<p><b>Observaciones:</b> Debe existir en el módulo al menos un proyecto y debe existir una base de datos que guarde la información de los datos en las tablas.</p>																
<p><b>Prototipo de interfaz:</b></p>  <table border="1"> <thead> <tr> <th>id</th> <th>container_id</th> <th>container_type</th> <th>filename</th> <th>disk</th> </tr> </thead> <tbody> <tr> <td>1</td> <td></td> <td></td> <td>uci.csv</td> <td>180!</td> </tr> <tr> <td>40</td> <td>29</td> <td>ProdanalysisPlanAlgorithm</td> <td>results(26).csv</td> <td>180!</td> </tr> </tbody> </table>		id	container_id	container_type	filename	disk	1			uci.csv	180!	40	29	ProdanalysisPlanAlgorithm	results(26).csv	180!
id	container_id	container_type	filename	disk												
1			uci.csv	180!												
40	29	ProdanalysisPlanAlgorithm	results(26).csv	180!												

Tabla 6.2.5 HU Registrar plan de ejecución.

HU Registrar plan de ejecución.	
<b>Número:</b> 6	<b>Nombre del requisito:</b> Registrar plan de ejecución.
<b>Programador:</b> Yem Liyanis Tamayo Cárdenas Pedro Alejandro Villavicencio Martínez	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Media	<b>Tiempo Estimado:</b> 16 días.
<b>Riesgo en Desarrollo:</b> N/A	<b>Tiempo Real:</b> 15 días.
<p><b>Descripción</b></p> <p>El usuario selecciona el botón crear plan de ejecución, se llenan los campos obligatorios y se ejecuta el algoritmo escogido.</p> <p>Para ello debe existir en el módulo al menos un proyecto, el usuario debe ser miembro de algún proyecto y se deben llenar los campos obligatorios al crear el plan.</p>	
<p><b>Observaciones:</b> Debe existir en el módulo al menos un proyecto y el usuario debe ser miembro de algún proyecto.</p>	
<p><b>Prototipo de interfaz:</b></p> 	

## 2.5 Patrón arquitectónico

Los patrones de arquitectura, o también llamados arquetipos ofrecen soluciones a problemas de arquitectura de software. Dan una descripción de los elementos y el tipo de relación que tienen junto con un conjunto de restricciones sobre cómo pueden ser usados (Pérez, 2016).

Para la construcción del sistema propuesto se define el patrón Modelo-Vista-Controlador (MVC), es un patrón de diseño, que se ubica en la categoría de patrones arquitectónicos; este patrón se especifica bajo la proposición de dividir la aplicación en tres tipos de elementos, el modelo, las vistas y controladores. La manera en que los elementos dentro de MVC se comunican difieren y no sólo lo diferencia el tipo de aplicación que se está describiendo (Desktop, WEB). Este patrón de arquitectura de software se basa en las ideas de reutilización de código y la separación de conceptos, características que buscan facilitar la tarea de desarrollo de aplicaciones y su posterior mantenimiento (Pérez, 2016).

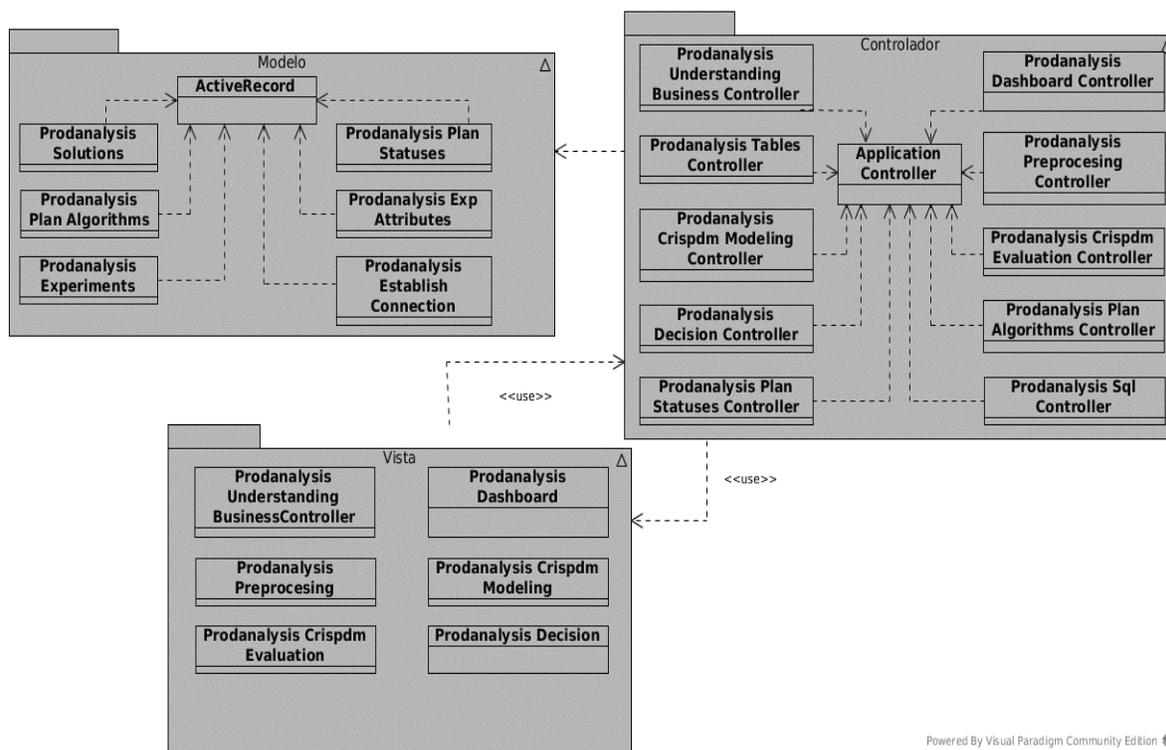


Figura.2.3 Arquitectura Modelo-Vista-Controlador (Fuente: Elaboración propia).

## Capítulo 2. Características y diseño del sistema.

En el modelo están todas las clases relacionadas con el modelo de la aplicación y los elementos (objetos) que contienen los datos y definen la lógica para manipular dichos datos. En la vista se hace referencia a los elementos que representan algo visible en la interfaz de usuario, por ejemplo, un panel, botones y el controlador actúa como un mediador entre los objetos del modelo y la vista, además realiza todas las tareas específicas de la aplicación.

### **2.6 Patrones de diseño**

A un nivel menor de abstracción de los patrones arquitectónicos se encuentran los patrones de diseño. Un patrón de diseño provee un esquema para refinar los subsistemas o componentes de un sistema de software, o las relaciones entre ellos. Describe la estructura comúnmente recurrente de los componentes en comunicación, que resuelve un problema general de diseño en un contexto particular. Tienden a ser independientes de los lenguajes y paradigmas de programación y su aplicación no afecta necesariamente al sistema completo pero si a un subsistema o parte del mismo (Pérez, 2016).

Para la definición de las clases del sistema, es importante la revisión de algunos patrones que permiten realizar un diseño adecuado y consistente.

#### **2.6.1 Patrones de Diseño GRASP**

Los patrones GRASP (Patrones Generales de Asignación de Responsabilidades) son patrones de diseño que se usan para asignar responsabilidades a una clase. A continuación Ilustración 6 Arquitectura Modelo-Vista-Controlador (Fuente: Elaboración propia).

se muestran los más utilizados en la solución propuesta, con una breve descripción y ejemplos donde son aplicados (Garzás, 2014).

Experto: Asignar una responsabilidad a la clase más competente en información, la clase cuenta con la información necesaria para cumplir la responsabilidad. Es el principio básico de asignación de responsabilidades que suele utilizarse en el diseño orientado a objetos (Garzás, 2014), en el módulo se pone en práctica en la clase del módulo `prodanalysis_plan_algorithm`.

## Capítulo 2. Características y diseño del sistema.

Controlador: Se aplica para realizar las asignaciones en cuanto al manejo de los eventos del sistema y definir sus operaciones (Garzías, 2014), en el módulo se pone en práctico en la clase `prodanalysis_plan_algorithms_controller`.

Bajo acoplamiento: Soporta el diseño de clases más independientes. Asigna las responsabilidades de forma tal que las clases se comuniquen con el menor número de clases que sea posible (Garzías, 2014), en el módulo se pone en práctica en la clase del módulo `prodanalysis_plan_algorithm`.

Las clases de las vistas y los modelos del dominio, son totalmente independientes unas de otras, o sea, no existe una relación directa entre estas clases debido a la separación que hace el patrón arquitectónico Modelo-Vista-Controlador (MVC).

Alta cohesión: Asignar una responsabilidad de modo que la unión se mantenga a gran escala. Asignar a las clases responsabilidades que trabajen sobre una misma área de aplicación y que no tengan mucha complejidad. Mejoran la claridad y facilidad con que se entiende el diseño (Garzías, 2014), en el módulo se pone en práctica en la clase del módulo `prodanalysis_plan_algorithms_controller`.

Este patrón arquitectónico permite agrupar de forma clara las vistas, la clase controladora y las clases modelos, estas últimas donde se implementa la lógica de negocio.

### 2.6.2 Patrones de Diseño GOF

Los patrones GOF (*Gand of Four*, Banda de los Cuatro) son patrones de diseño que definen una descripción de clases y objetos comunicándose entre sí, adaptada para resolver un problema de diseño general en un contexto particular. Estos patrones se dividen en tres categorías: los creacionales, los estructurales y los de comportamiento. Seguidamente se expondrán los patrones empleados (Pérez, 2016)

#### Creacionales

- Singleton: Este patrón consiste en garantizar que una clase solo tenga una instancia y proporcionar un punto de acceso global a ella.

## Estructurales

- Fachada: Este patrón se emplea para brindar una interfaz que abstraee completamente al usuario de la complejidad de los procesos.

## 2.7 Diagrama entidad-relación

Un diagrama entidad-relación, conocido como modelo entidad relación es un tipo de diagrama de flujo que ilustra cómo las "entidades", como personas, objetos o conceptos, se relacionan entre sí dentro de un sistema. Los diagramas ER se usan a menudo para diseñar o depurar bases de datos relacionales en los campos de ingeniería de software, sistemas de información empresarial, educación e investigación. También conocidos como los ERD o modelos ER, emplean un conjunto definido de símbolos, tales como rectángulos, diamantes, óvalos y líneas de conexión para representar la interconexión de entidades, relaciones y sus atributos. Son un reflejo de la estructura gramatical y emplean entidades como sustantivos y relaciones como verbos (Chen., 2018).

Con este modelo se logra representar de manera gráfica la estructura lógica de la base de datos.

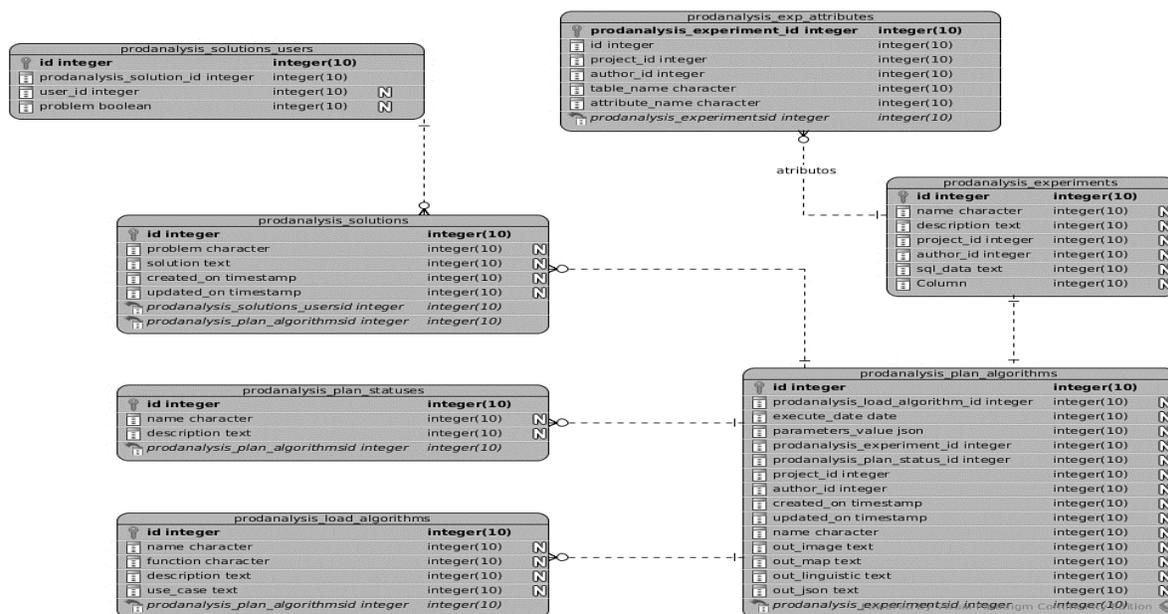


Ilustración 7 Modelo entidad relación (Fuente: Elaboración propia)

## 2.8 Conclusiones Parciales

- Mediante la representación del modelo de dominio, se identificaron las entidades relacionados con el módulo y las relaciones entre ellas posibilitando un mejor entendimiento para el usuario cliente de cuáles serían las funciones específicas de cada una de ellas en el módulo.
- El levantamiento de los requisitos funcionales de la aplicación, permitió dar respuesta a las necesidades del problema, describiendo a detalle lo que el módulo debe hacer en cada momento y dejando claro la prioridad de cada una de las funcionalidades que deben existir en él para dar solución al problema planteado.
- A través de la identificación de los requisitos no funcionales se definieron las restricciones del módulo posibilitando a los autores enmarcar el área de trabajo y uso de tecnologías específicas que cumplieran con las restricciones establecidas por el cliente.
- La definición de la propuesta de solución del problema, detallándola con la ayuda de los artefactos propuestos por la metodología SCRUM y XP, posibilitó estructurar todo el proceso de desarrollo ya que se tomó de cada una los elementos que hacían falta para darle solución al problema y que el cliente pudiera tener mayor claridad de todo el desarrollo realizado.
- La selección del patrón arquitectónico, los patrones de diseño, así como el modelo de datos, facilitó garantizar el correcto funcionamiento y organización del módulo.

## Capítulo 3. Implementación y Pruebas

Este capítulo consiste en la implementación y ejecución de casos de prueba que evalúen las funcionalidades del módulo. Se definen los estándares de codificación que posibilitan la organización en la escritura del código en el proceso de desarrollo del mismo, buenas prácticas de programación que se tuvieron en cuenta en la implementación siguiendo el principio de estandarización. Además, se abordan los tipos de pruebas realizadas y los resultados obtenidos en las mismas.

### 3.1. Estilo de escritura

El estilo de escritura que se aplica a frases o palabras compuestas es el *CamelCase*. Deben de empezar un una letra mayúscula y la separación de múltiples palabras debe de ser concatenadas, pero con la primera letra de la siguiente palabra en mayúscula y se utiliza en declaración de nombres de clases y módulos. También se utiliza una de las variantes del CamelCase q se llama `SCREAMING_SNAKE_CASE` lo que consiste en escribir nombres en mayúsculas y la separación de múltiples palabras debe de ser concatenadas con un guion bajo (`_`) y se utiliza para crear variables.

### 3.2. Estándares de Codificación

Un estándar de codificación agrupa los aspectos relacionados con la generación del código. Los programadores lo realizan para trabajar de forma coordinada dentro del proyecto, de acuerdo con el tipo de lenguaje que utilizan y las normas que propone el mismo, tratando de lograr un código claro y legible tanto para él como para otro programador (Network, 2015).

A continuación, se muestran algunos estándares de codificación que se tuvieron en cuenta para la implementación:

#### **Indentación:**

Todos los métodos, bloques, condiciones, deben de ir correctamente indentados y dicha tabulación debe de ser de 2 espacios en vez de tabs.

```
def delete
  unless @execution_plan.nil?
    @execution_plan.destroy
    flash[:notice] = l(:notice_successful_delet
  end
  redirect_to action: 'index'
end
```

Ilustración 8 Identación(Elaboración propia)

### **Declaraciones:**

Los métodos y variables se declararon con nombres asociados a la función por la cual fueron creados.

```
def params_by_algorithm(algorithm_id, objective_id)
  result = []
  algorithm_params =
  ProdanalysisAlgorithmsCustomParams.where(prodanalysis_algorithm_id: algorithm_id).order('param_order')
  str_params = ''
  position = true
  algorithm_params.each do |algorithm_param|
    str_format =
    param_format_by_id(algorithm_param.prodanalysis_custom_param_id)
  end
```

Ilustración 9 Declaraciones(Elaboración propia)

### **Clases:**

Las clases deben comenzar con mayúscula y en caso de estar conformada por palabras compuestas, la definición debe ser continua y cada palabra debe iniciar con mayúscula siguiendo el estilo determinado.

```

class ProdanalysisPlanStatusesController < BaseProdanalysisDataController
  before_action :require_login
  before_action :find_prodanalysis_plan_status, only: [:update, :delete, :retrieve]
  def index
    @statuses = RiskStatus.all
  end
  def create
    @status = RiskStatus.new
    edit(@status, params[:status], l(:notice_successful_create), nil)
  end
  def update
    edit(@status, params[:status], l(:notice_successful_update), nil)
  end
  def retrieve
  end

  def delete
    flash[:notice] = l(:notice_successful_delete) if @status.destroy
    redirect_to action: 'index'
  end
  protected
  def find_prodanalysis_plan_status
    @status = ProdanalysisPlanStatus.find(params[:id])
    rescue ActiveRecord::RecordNotFound
      flash[:error] = l(:plan_status_does_not_exists_message)
      redirect_to action: :index, project_id: @project
    end
  end
end

```

Ilustración 10 Clases(Elaboración propia)

### 3.3. Diagrama de despliegue

El Diagrama de Despliegue es un tipo de diagrama del Lenguaje Unificado de Modelado que se utiliza para modelar el hardware utilizado en las implementaciones de sistemas y las relaciones entre sus componentes. Un diagrama de despliegue muestra las relaciones físicas entre los componentes hardware y software en el sistema final, es decir, la configuración de los elementos de procesamiento en tiempo de ejecución y los componentes software (procesos y objetos que se ejecutan en ellos) (Martínez, 2013).

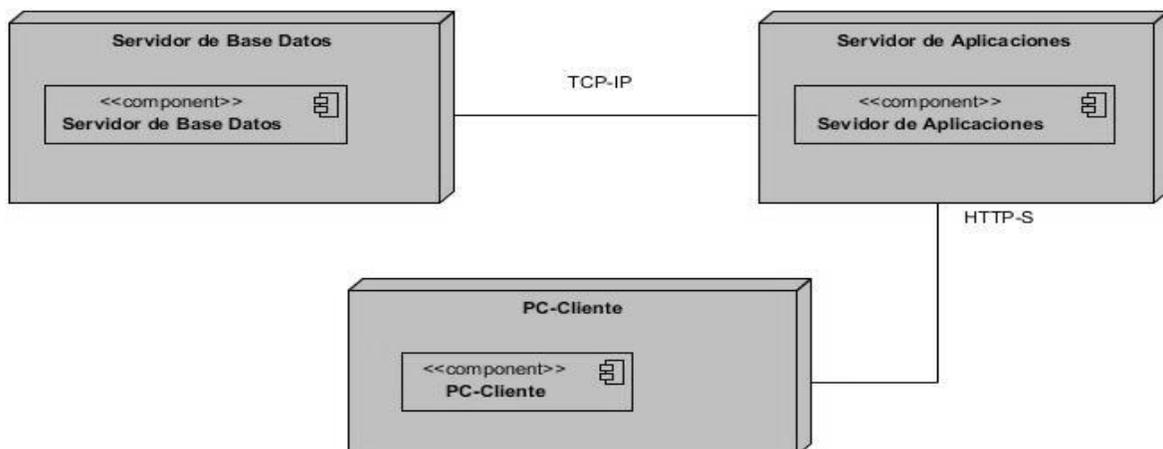


Ilustración 11 Diagrama de despliegue (Fuente: Elaboración propia).

### 3.4. Pruebas de software

Las aplicaciones (en general cualquier mecanismo diseñado e implementado por un humano) son propensas a tener fallos. A veces, pueden contribuir al fracaso de cualquier proyecto de software, e impactar de forma negativa en toda una empresa. Los tiempos de desarrollo, los entornos de programación, las diferencias entre versiones, todo influye para que, incluso con la máxima dedicación, puedan darse fallos que empañen la imagen y a veces la reputación, de una organización. Surge por tanto la necesidad de asegurar en lo posible, la calidad del producto. Las pruebas de software son las investigaciones empíricas y técnicas cuyo fin es proporcionar información objetiva e independiente sobre la calidad del producto. Esta actividad forma parte del proceso de control de calidad global. Las pruebas son básicamente un conjunto de actividades dentro del desarrollo de software y dependiendo del tipo de pruebas, estas actividades podrán ser implementadas en cualquier momento del proceso de desarrollo (Londoño, 2017)

Por tanto, las pruebas de software son un conjunto de actividades para la verificación del comportamiento de un programa y encontrar los posibles fallos de implementación o usabilidad dentro del software. El método de prueba que se aplica a la herramienta de configuración es el de caja negra y consiste en el elemento estudiando desde el punto de vista de las entradas que recibe y las salidas o respuestas que produce, sin tener en cuenta su funcionamiento interno; permitiendo examinar los aspectos funcionales del sistema haciendo mínimo enfoque en la estructura lógica interna del software. Esta técnica aplica pruebas de aceptación que son creadas en base a las historias de usuarios, en cada ciclo de la iteración del desarrollo.

El cliente debe especificar uno o diversos escenarios para comprobar que una historia de usuario ha sido correctamente implementada. Los clientes son responsables de verificar que los resultados de estas pruebas sean correctos. Asimismo, en caso de que fallen varias pruebas, deben indicar el orden de prioridad de resolución. Una historia de usuario no se puede considerar terminada hasta tanto pase correctamente todas las pruebas de aceptación. Dado que la responsabilidad es grupal, es recomendable publicar los resultados de las pruebas de aceptación, de manera que todo el equipo esté al tanto de esta información (JOSKOWICZ, 2008).

### 3.5. Ambientes de pruebas

Las pruebas fueron realizadas en una PC con un microprocesador Core I3 a una frecuencia de 3.3 GHz, una memoria RAM de 4 Gb, sobre la distribución 16.04 de Ubuntu como sistema operativo y un disco duro de 1TB.

### 3.6. Diseño de casos de pruebas

Los casos se diseñaron a partir de las historias de usuario. Una historia de usuario puede tener tantos casos de prueba como sean necesarios para evaluar su funcionamiento. A continuación se muestran los casos de prueba de aceptación elaborados.

Tabla 7.3.1 Caso de prueba 1

<b>Caso de prueba de aceptación</b>
<b>Historia de usuario:</b> Cargar ficheros CSV.
<b>Nombre:</b> Cargar ficheros CSV.
<b>Responsable:</b> Pedro Alejandro Villavicencio Martínez.
<b>Descripción:</b> Esta prueba se realiza con el objetivo de verificar si la funcionalidad de cargar ficheros en formato CSV funciona conforme a lo esperado.
<b>Condiciones de ejecución:</b> Debe existir algún archivo en alguna ubicación del directorio que contenga el formato CSV.
<b>Entradas/Pasos de ejecución:</b> Seleccionar del menú la opción Examinar y localizar el archivo deseado.
<b>Resultado esperado:</b> Se carga el archivo del directorio seleccionado.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

Tabla 8.3.2 Caso de prueba 2

<b>Caso de prueba de aceptación</b>
<b>Historia de usuario:</b> Cargar ficheros CSV.
<b>Nombre:</b> Cargar ficheros CSV.
<b>Responsable:</b> Pedro Alejandro Villavicencio Martínez.
<b>Descripción:</b> Esta prueba se realiza con el objetivo de ver si el usuario puede cargar ficheros que no estén en formato CSV.

### Capítulo 3. Implementación y Pruebas.

<b>Condiciones de ejecución:</b> Debe existir algún archivo en alguna ubicación del directorio que contenga el formato CSV.
<b>Entradas/Pasos de ejecución:</b> Seleccionar del menú la opción Examinar y localizar el archivo que no esté en formato CSV.
<b>Resultado esperado:</b> No se carga el archivo del directorio seleccionado y muestra el mensaje "Parámetros es requerido".
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

Tabla 9.3.3 Caso de prueba 3

<b>Caso de prueba de aceptación</b>
<b>Historia de usuario:</b> Salvar ficheros CSV.
<b>Nombre:</b> Salvar ficheros CSV.
<b>Responsable:</b> Pedro Alejandro Villavicencio Martínez.
<b>Descripción:</b> Esta prueba se realiza con el objetivo de verificar si la funcionalidad de salvar ficheros en formato CSV funciona conforme a lo esperado.
<b>Condiciones de ejecución:</b> Debe existir espacio suficiente en el directorio.
<b>Entradas/Pasos de ejecución:</b> Seleccionar del menú la opción Exportar a CSV y ubicar en el directorio seleccionado.
<b>Resultado esperado:</b> Se guarda el archivo en el directorio seleccionado.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

Otras de las pruebas utilizadas fue la de encuestas de satisfacción, la cual se les realizó a algunos expertos del centro CDAE, realizando la siguiente pregunta:

¿Qué importancia tiene el análisis de datos en el departamento de GESPRO atendiendo a los indicadores?:

- Muy Baja
- Baja
- Media
- Alta
- Muy Alta

Encuesta Importancia Analisis de Datos de Xedro Gespro. Centro CDA					
Nombres y Apellidos	Muy baja	Baja	Media	Alta	Muy Alta
Suraine Torres Lopez					X
Javier Menendez Rizos				X	
Arisney Figueredo Ramos				X	
Eliubis Matos Matos				X	
Ariel Tamayo Heredia				X	
Lainer Piedra Dieguez				X	
Ever Leandro Santiesteban				X	
Henry Dexter Acuña Perez				X	
Arturo Aguilera Jardines				X	
Iliana Perez Pupo					X
Pedro Piñeiro Perez					X
Yairile Cruz Castillo					X
Daysi Oropeza Mendez				X	
Pascual Verdecia Vicet					X
Anie Bermudez Peña					X
Oswaldo Santos Acosta					X
Rosel Sosa Gonzalez					X

Ilustración 12 Encuesta de satisfacción (Fuente: Elaboración propia).

A través de la encuesta se realizó un gráfico donde se muestran los resultados concretos sobre las opiniones de los expertos del departamento GESPRO el cual se muestra a continuación.

## Encuestas

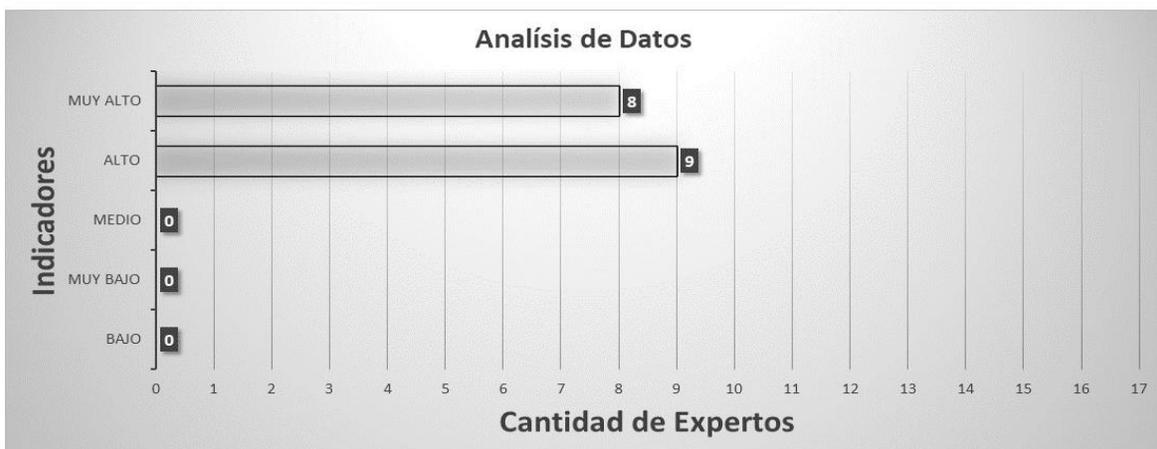


Ilustración 13 Resultados de la encuesta (Fuente: Elaboración propia).

### 3.7. Ejecución de los casos de pruebas de aceptación

El proceso de pruebas a cualquier software se realiza a través de iteraciones, donde, a medida que se procede con una nueva iteración deben haberse corregido los defectos encontrados en la anterior, para garantizar que al final del proceso el producto quede libre de la mayor cantidad de errores posible y listo para entregar al cliente.

Durante la ejecución de los casos de prueba de aceptación quedó en evidencia algunas no conformidades por la cuales no se obtuvieron los resultados esperados. En la primera iteración se obtuvieron catorce no conformidades, en la segunda iteración se obtuvo cuatro no conformidades y en la tercera iteración se resolvieron las no conformidades detectadas donde los resultados de los casos de prueba correspondían con los resultados esperados.



Ilustración 14 Resumen de no conformidades (Fuente: Elaboración propia).

### 3.8. Conclusiones parciales

- En este capítulo se registraron las bases necesarias para la implementación de la aplicación lo que facilitó a los programadores poder predecir un mejor camino de desarrollo y comodidad a la hora de implementar cada una de las funcionalidades del módulo ya que las bases sobre las cuales se implementaría ya estaban registradas.

### Capítulo 3. Implementación y Pruebas.

- Con la especificación de los estándares de codificación se abarco todo lo referente a la estructura que contendrá el desarrollo del sistema posibilitando que la misma sea homogénea.
- El diagrama de despliegue posibilita un mejor entendimiento de cómo quedará estructurado el módulo y de cuáles serán las conexiones específicas entre cada componente del mismo.
- Por otra parte, se realizaron las pruebas de aceptación a cada historia de usuario dependiendo de la cantidad de flujos que se encuentran en cada una de ellas las cuales permitieron a los autores saber cuántas no conformidades podían encontrarse en el módulo en general para así resolverlas y saber cuántas pruebas fueron satisfactorias las que proporcionaran al cliente conformidad y seguridad.

## **Conclusiones Generales**

Al terminar la investigación y desarrollo del módulo de análisis se arribaron a las siguientes conclusiones:

- El mismo garantiza obtener experiencia de errores de proyectos anteriores y permite una mejor visibilidad en cuanto a costo.
- Satisface las necesidades del cliente, es factible para aquellos usuarios que necesiten el servicio del mismo y que estén autorizados a su fácil acceso.
- Le facilita la planificación de futuros proyectos a los usuarios que lo utilicen ya que permite ver experiencias de proyectos anteriores.
- La visibilidad que brinda de proyectos terminados y de aquellos que están en ejecución permite que se pueda implementar un proceso de mejora que ayude al perfeccionamiento a partir de experiencias adquiridas llevando a cabo el análisis de datos.
- Permite predecir aspectos alarmantes sobre los datos teniendo en cuenta el uso de algoritmos, planes de ejecución y experimentos.
- Permite realizar una planificación adecuada de los recursos necesarios, estimación de tiempo para proyectos futuros.
- Permite al usuario que pueda llevar a cabo una correcta interpretación de los datos lo que facilita la toma de decisiones y selección de estrategias para nuevos proyectos.

**Bibliografía**

1. Arancibia, J. A. (2016). Metodología para el Desarrollo de Proyectos en Minería de Datos CRISP-DM.
2. Ferreira, T. (2017). Análisis de datos ¿por qué es tan importante para mi empresa ?  
Obtenido de Análisis de datos ¿por qué es tan importante para mi empresa ?
3. Garzás, J. (5 de Agosto de 2014). Los patrones GRASP. Obtenido de Los patrones GRASP: <http://www.javiergarzas.com>.
4. goicochea, a. (2009). CRISP-DM, Una metodología para proyectos de Minería de Datos. Obtenido de CRISP-DM, Una metodología para proyectos de Minería de Datos.
5. gonzalez., M. (2012). Técnicas, métodos empíricos y tipos de investigación .  
Obtenido de Técnicas, métodos empíricos y tipos de investigación :  
<http://es.slideshare.net/MAGDAGONZALEZP/técnicas-métodos-empíricos-y-tipos-de-investigación-y-de-observación-por-magda-gonzalez>.
6. Henao, J. C. (25 de Enero de 2011). Tipos de requisistos. Obtenido de Tipos de requisitos : <https://es.slidereshare.net/juanchenao/tipos-de-requisitos>.
7. MODELADO, ANÁLISIS, DISEÑO Y DOCUMENTACIÓN. (5 de Marzo de 2018).  
Obtenido de MODELADO, ANÁLISIS, DISEÑO Y DOCUMENTACIÓN:  
<https://prezi.com>.
8. Nazar. (2014). Herramienta de modelado.
9. Nazar, L. A. (2014). Uml tutorial-visual-paradigm. Obtenido de Uml tutorial-visual-paradigm: <http://es.slideshare.net/fabiannazar1/uml-tutorialvisualparadigm>.
10. Palacio. (2014). Metodología de desarrollo.
11. Pérez, J. S. (2016). Patrones diseño y arquitectura. Obtenido de Patrones diseño y arquitectura: <https://es.slideshare.net/SebastianRamrez2/patrones-diseño-1>.
12. Pérez., J. S. (2016).Patrones GOF. . Obtenido de Patrones GOF. :  
<https://es.slideshare.net/SebastianRamrez2/patrones-gof>.

13. Pestano Pino, H. (2011). Propuesta de modelo de desarrollo para líneas de productos de software en centros de producción”, Tesis de Maestría en Gestión de Proyectos.
14. Peter Chen. (2018). Qué-es-un-diagrama-entidad-relación. Obtenido de qué-es-un-diagrama-entidad-relación: <https://www.lucidchart.com>.
15. Rouse, M. (2005). ¿Qué es Gestión de proyectos de TI? Obtenido de ¿Qué es Gestión de proyectos de TI?: <http://¿Qué es Gestión de proyectos de TI ?>.
16. Satpathy, T. (2016). A Guide to the SCRUM BODY OF KNOWLEDGE (SBOK™GUIDE) 2016 Edition.
17. Wesley, M. C. (2004). "User Stories Applied. Obtenido de Extreme Software Engineering, Pearson Education.
18. CHAPMAN, Pete, et al. CRISP-DM 1.0 Step-by-step data mining guide. 2000.
19. AZEVEDO, Ana Isabel Rojão Lourenço; SANTOS, Manuel Filipe. KDD, SEMMA and CRISP-DM: a parallel overview. IADS-DM, 2008.
20. LUQUE MARTÍNEZ, Teodoro; BARRIO GARCÍA, Salvador del. Técnicas de análisis de datos en investigación de mercados. 2000.
21. MIRANDA, Juan José Miranda. Gestión de proyectos. MMEditores, 2010.
22. LLEDÓ, Pablo; RIVAROLA, Gustavo. Gestión de proyectos. Pearson Educación, 2007.
23. EDITORIAL VÉRTICE. Gestión de proyectos. Editorial Vértice, 2008.
24. WALLACE, William. Gestión de proyectos. Edimburgo: Edinburgh Business School– Heriot Watt University, 2014.
25. GÓMEZ GONZÁLEZ, Fernando. Gestión de Proyectos. 2017.
26. RAMIREZ-ANORMALIZA, Richard Ramirez, et al. Gestión de Proyectos Informáticos. 2017.
27. CURTO GONZÁLEZ, Tomás, et al. Análisis de Datos. 2018.

28. RODRÍGUEZ, José Ramón. Gestión de proyectos informáticos: métodos, herramientas y casos. Editorial Uoc, 2005.
29. GARRETA, José Salvador Sánchez. Ingeniería de proyectos informáticos: actividades y procedimientos. Publicacions de la Universitat Jaume I, 2003.
30. GARRE, Miguel, et al. Comparación de diferentes algoritmos de clustering en la estimación de coste en el desarrollo de software. REICIS. Revista Española de Innovación, Calidad e Ingeniería del Software, 2007, vol. 3, no 1.
31. KUNA, Horacio Daniel, et al. Avances en procedimientos de la explotación de información con algoritmos basados en la densidad para la identificación de outliers en bases de datos. En XIII Workshop de Investigadores en Ciencias de la Computación. 2011.
32. Snyder C. A project manager's book of forms: A companion to the PMBOK guide. John Wiley & Sons; 2017.
33. SCHWABER, Ken. Scrum development process. En Business object design and implementation. Springer, London, 1997. p. 117-134.
34. CARDOZO, Elisa SF, et al. SCRUM and Productivity in Software Projects: A Systematic Literature Review. En EASE. 2010.
35. ŁUKASIEWICZ, K.; MILER, J. Improving agility and discipline of software development with the Scrum and CMMI. IET software, 2012, vol. 6, no 5, p. 416-422.
36. SCHWABER, Ken; BEEDLE, Mike. Agile software development with Scrum. Upper Saddle River: Prentice Hall, 2002.13:41.
37. COHN, Mike. Succeeding with agile: software development using Scrum. Pearson Education, 2010.
38. ŁUKASIEWICZ, K.; MILER, J. Improving agility and discipline of software development with the Scrum and CMMI. IET software, 2012, vol. 6, no 5, p. 416-422.13:42.

39. NOORDELOOS, Ramon; MANTELI, Christina; VAN VLIET, Hans. From RUP to Scrum in global software development: A case study. En Global Software Engineering (ICGSE), 2012 IEEE Seventh International Conference on. IEEE, 2012. p. 31-40.13:43.
40. MAHALAKSHMI, M.; SUNDARARAJAN, M. Traditional SDLC Vs Scrum Methodology–A Comparative Study. International Journal of Emerging Technology and Advanced Engineering, 2013, vol. 3, no 6, p. 192-196.
41. MATHARU, Gurpreet Singh, et al. Empirical study of agile software development methodologies: A comparative analysis. ACM SIGSOFT Software Engineering Notes, 2015, vol. 40, no 1, p. 1-6.
42. PERMANA, Putu Adi Guna. Scrum method implementation in a software development project management. International Journal of Advanced Computer Science and Applications, 2015, vol. 6, no 9, p. 198-204.
43. MACHADO, Thais Cristina Sampaio; PINHEIRO, Plácido Rogério; TAMANINI, Isabelle. Project management aided by verbal decision analysis approaches: a case study for the selection of the best SCRUM practices. International Transactions in Operational Research, 2015, vol. 22, no 2, p. 287-312.13:44.
44. LEI, Howard, et al. A statistical analysis of the effects of Scrum and Kanban on software development projects. Robotics and Computer-Integrated Manufacturing, 2017, vol. 43, p. 59-67.
45. LEI, Howard, et al. A statistical analysis of the effects of Scrum and Kanban on software development projects. Robotics and Computer-Integrated Manufacturing, 2017, vol. 43, p. 59-67.
46. HANSSON, David Heinemeier, et al. Ruby on rails. Website. Projektseite: <http://www.rubyonrails.org>, 2009.13:45.
47. WAGSTROM, Patrick; JERGENSEN, Corey; SARMA, Anita. A network of rails: a graph dataset of ruby on rails and associated projects. En Proceedings of the 10th

- Working Conference on Mining Software Repositories. IEEE Press, 2013. p. 229-232.
48. HANSSON, David Heinemeier. Ruby on Rails (2013). URL <http://rubyonrails.org>.
49. HARTL, Michael. Ruby on Rails Tutorial: Learn Web Development with Rails. Addison-Wesley Professional, 2016.
50. CHAO, Kevin Zi Jun, et al. Surgical device and methods. U.S. Patent No 9,308,011, 12 Abr. 2016.
51. SOSA GONZÁLEZ, Rosel, et al. Ecosistema de Software GESPRO-16.05 para la Gestión de Proyectos. Revista Cubana de Ciencias Informáticas, 2016, vol. 10, p. 239-251.
52. CURTO GONZÁLEZ, Tomás, et al. Análisis de Datos. 2018.
53. VÁSQUEZ VALLES, Edson. Descriptive-analytics-machine-learning-deep-learning-crisp-dm.2018/05/. <https://www.kdnuggets.com>
54. VÁSQUEZ VALLES, Edson. Crisp-dm-top-methodology-analytics-data-mining-data-science-projects.2014/10. <https://www.kdnuggets.com>
55. Londoño, J. H. (octubre de 2017). Tipos de Pruebas de Software. Obtenido de Tipos de Pruebas de Software.
56. JOSKOWICZ, José. Reglas y prácticas en eXtreme Programming. Universidad de Vigo, 2008, vol. 22.
57. BYKBAEV, Vladímir Robles. Lenguajes de Scripting: ¿ una nueva forma de programar?. Ingenius, 2008, no 2, p. 27-30.
58. Escuela politécnica Internacional, A. V. (17 de Noviembre de 2012). GUÍA PARA LA INSTALACIÓN Y UTILIZACIÓN DEL SISTEMA GESTIONADOR DE BASES DE DATOS (DBMS) "POSTGRESQL" . Obtenido de GUÍA PARA LA INSTALACIÓN Y UTILIZACIÓN DEL SISTEMA GESTIONADOR DE BASES DE DATOS (DBMS) "POSTGRESQL" : <http://postgresql-dbms.blogspot.com>

59. PERALES, Ismael Bautista. Aplicación Web de bases de datos usando el Framework Ruby on Rails. 2014. Tesis Doctoral.
60. SCHULZ, Ralph G. Diseño web con CSS. Marcombo, 2008.
61. JetBrains. Entorno de desarrollo integrado, rubymine.marzo, 2017.
62. CORO, Untuña, et al. Multimedia Educativa para la Asignatura Fundamentos de la Informática, de la Carrera Ingeniería Informática en la Universidad de Granma. 2012.
63. ROMERO REYES, Ronald Jared, et al. Impact of an ERP system on the productivity of SME. *Tecnura*, 2012, vol. 16, no 34, p. 94-102.
64. DE PABLOS HEREDERO, Carmen, et al. *Organización y transformación de los sistemas de información en la empresa*. Esic Editorial, 2012.
65. ARCOS RODRÍGUEZ, Uzai. *Implantación de sistemas ERP en las pymes*. 2010. Tesis Doctoral.
66. COBOS, Carlos, et al. CMIN-herramienta case basada en CRISP-DM para el soporte de proyectos de minería de datos. *Ingeniería e Investigación*, 2010, vol. 30, no 3.
67. MARTHA, Mendoza, et al. CMIN-herramienta case basada en CRISP-DM para el soporte de proyectos de minería de datos CMIN—a CRISP-DM-based case tool for supporting data mining projects. *Ingeniería e Investigación*, 2010, vol. 30, no 3, p. 45-56.

### **Recomendaciones**

- Para proyecciones futuras la aplicación pueda ejecutar el plan de ejecución de forma asíncrona.
- Al realizar el plan de ejecución, dependiendo de las especificaciones del algoritmo seleccionado permitir cargar varios ficheros.
- Permitir en la fase de preprocesamiento de datos integrar los conjuntos de datos considerados útil para el modelado, así como la normalización de los mismos, tratamiento de valores ausentes y reducción de su volumen.

## Anexos

Tabla 10.2.6 HU Modificar Experimento

<b>HU Modificar Experimento</b>		
<b>Número:</b> 2	<b>Nombre del requisito:</b> Modificar Experimento	
<b>Programador:</b> Yem Liyanis Tamayo Cárdenas Pedro Alejandro Villavicencio Martínez	<b>Iteración Asignada:</b> 1	
<b>Prioridad:</b> Alta	<b>Tiempo Estimado:</b> 7 días.	
<b>Riesgo en Desarrollo:</b> N/A	<b>Tiempo Real:</b> 6 días.	
<b>Descripción</b>  El usuario selecciona el experimento a modificar, se modifica el campo que se desee y se presiona modificar.  Para ello debe existir al menos un proyecto donde haya creado como mínimo un experimento y el usuario debe ser miembro de algún proyecto.		
<b>Observaciones:</b> Debe existir al menos un experimento, debe existir en el módulo al menos un proyecto y el usuario debe ser miembro de algún proyecto.		
<b>Prototipo</b>	<b>de</b>	<b>interfaz:</b>

Tabla 11.2.7 HU Eliminar Experimento

HU Eliminar Experimento	
<b>Número:</b> 3	<b>Nombre del requisito:</b> Eliminar Experimento.
<b>Programador:</b> Yem Liyanis Tamayo Cárdenas Pedro Alejandro Villavicencio Martínez	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Media	<b>Tiempo Estimado:</b> 5 días.
<b>Riesgo en Desarrollo:</b> N/A	<b>Tiempo Real:</b> 4 días.

**Descripción**

El usuario selecciona el experimento a eliminar y luego se muestra un mensaje indicando si está seguro de borrar el experimento y se presiona aceptar.

Para ello debe existir al menos un experimento, en el módulo al menos un proyecto y el usuario debe ser miembro de algún proyecto.

**Observaciones:** Debe existir al menos un experimento, en el módulo al menos un proyecto y el usuario debe ser miembro de algún proyecto.

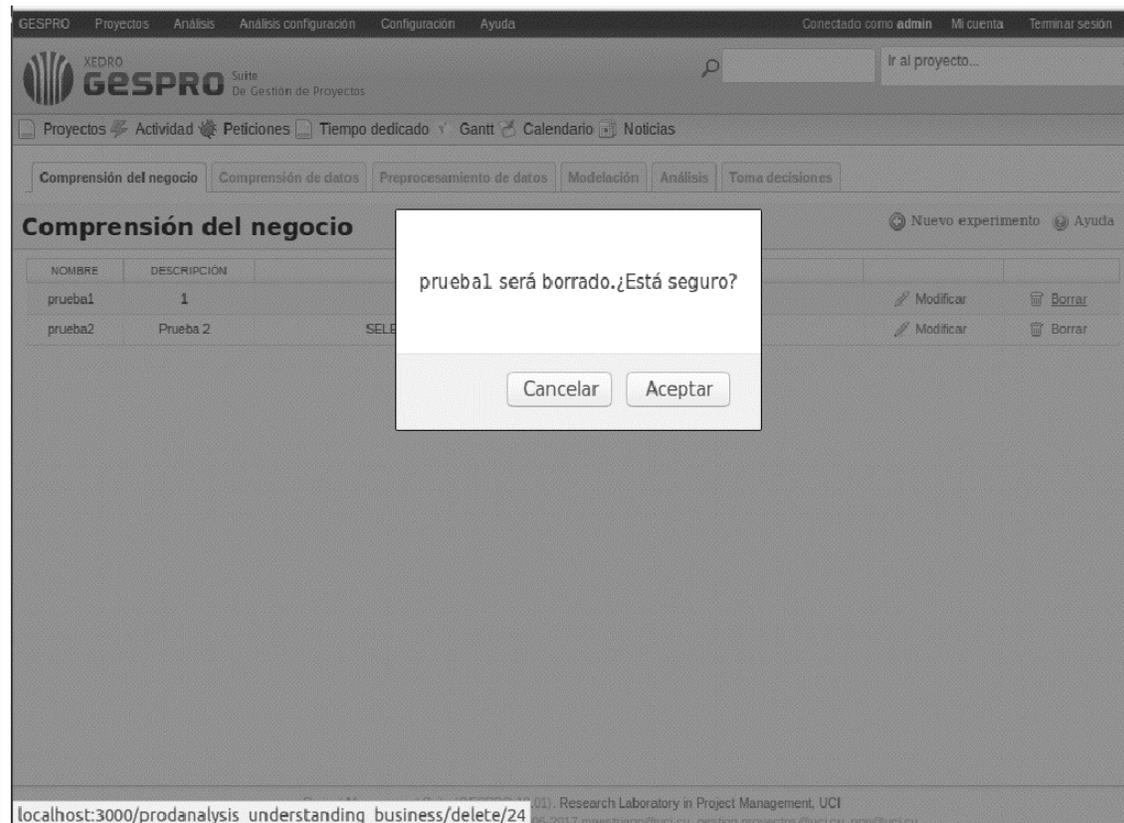
**Prototipo de interfaz:**

Tabla 12.2.8 HU Mostrar datos de consultas select.

<b>HU Mostrar datos de consultas select.</b>	
<b>Número:</b> 5	<b>Nombre del requisito:</b> Mostrar datos de consultas select.
<b>Programador:</b> Yem Liyanis Tamayo Cárdenas Pedro Alejandro Villavicencio Martínez	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Media	<b>Tiempo Estimado:</b> 10 días.
<b>Riesgo en Desarrollo:</b> N/A	<b>Tiempo Real:</b> 9 días.
<b>Descripción</b>  El usuario debe acceder a la consola para ejecutar la consulta select y luego se muestran los resultados de la ejecución de la consulta la cual se exporta en fichero csv  Para ello debe existir una consola donde se puedan ejecutar las consultas select, existir en el módulo al menos un proyecto y el usuario debe ser miembro de algún proyecto.	
<b>Observaciones:</b> Debe existir una consola donde se puedan ejecutar las consultas select, existir en el módulo al menos un proyecto y el usuario debe ser miembro de algún proyecto.	

Tabla 13.2.9 HU Mostrar resultados del plan de ejecución.

<b>HU Mostrar resultados del plan de ejecución.</b>	
<b>Número:</b> 7	<b>Nombre del requisito:</b> Mostrar resultados del plan de ejecución.
<b>Programador:</b> Yem Liyanis Tamayo Cárdenas Pedro Alejandro Villavicencio Martínez	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Baja	<b>Tiempo Estimado:</b> 6 días.

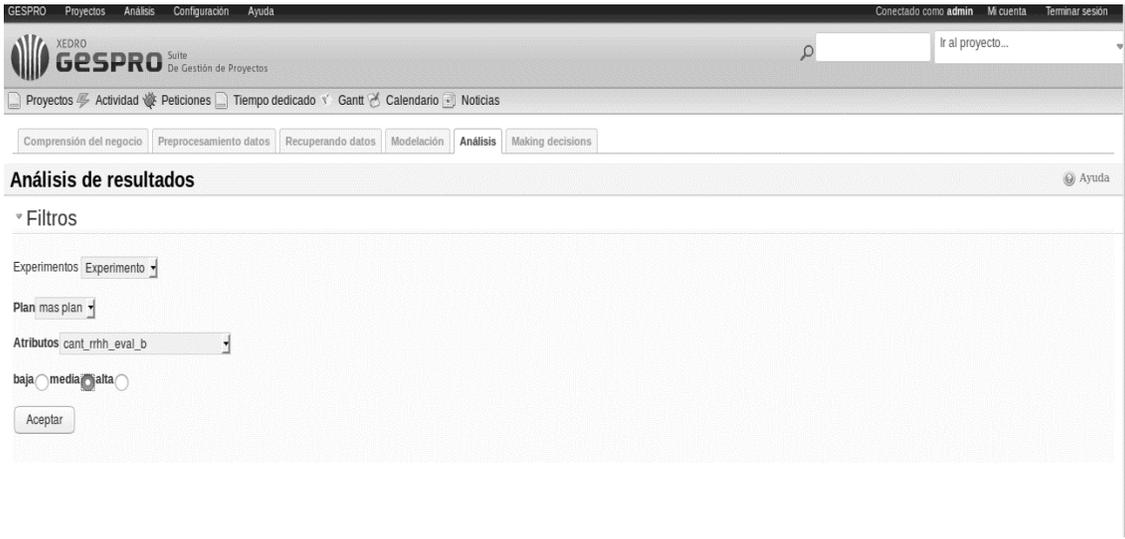
<b>Riesgo en Desarrollo:</b> N/A	<b>Tiempo Real:</b> 6 días.
<p><b>Descripción</b></p> <p>Luego de haberse creado el plan de ejecución se muestran los resultados del mismo.</p> <p>Para ello debe haberse creado al menos un plan de ejecución, existir en el módulo al menos un proyecto y el usuario debe ser miembro de algún proyecto.</p>	
<p><b>Observaciones:</b> Debe haberse creado al menos un plan de ejecución, existir en el módulo al menos un proyecto y el usuario debe ser miembro de algún proyecto.</p>	
<p><b>Prototipo de interfaz:</b></p> 	

Tabla 14.2.10 HU Exportar ficheros CSV.

<b>HU Exportar ficheros CSV.</b>	
<b>Número:</b> 8	<b>Nombre del requisito:</b> Exportar ficheros CSV.
<p><b>Programador:</b></p> <p>Yem Liyanis Tamayo Cárdenas</p> <p>Pedro Alejandro Villavicencio Martínez</p>	<p><b>Iteración Asignada:</b> 1</p>

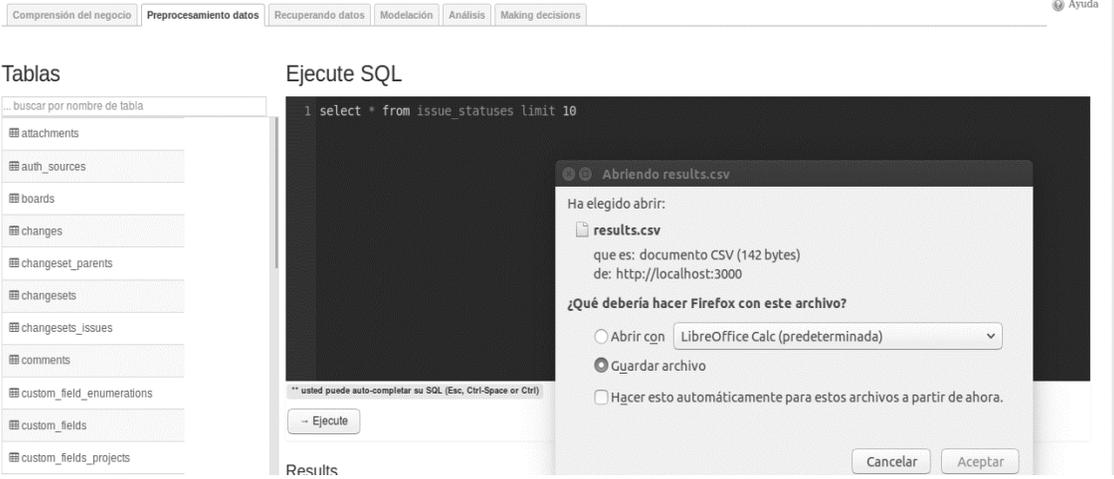
<b>Prioridad:</b> Baja	<b>Tiempo Estimado:</b> 5 días.
<b>Riesgo en Desarrollo:</b> N/A	<b>Tiempo Real:</b> 3 días.
<p><b>Descripción</b></p> <p>Luego de haber seleccionado los datos y ejecutar la consulta select se exporta el fichero CSV.</p> <p>Para ello se deben haber seleccionado los datos previamente y haber ejecutado la consulta select.</p>	
<p><b>Observaciones:</b> Deben haber seleccionado los datos previamente y haber ejecutado la consulta select.</p>	
<p><b>Prototipo de interfaz:</b></p> 	

Tabla 15.2.11 HU Cargar ficheros CSV.

<b>HU Cargar ficheros CSV.</b>	
<b>Número:</b> 9	<b>Nombre del requisito:</b> Cargar ficheros CSV.
<b>Programador:</b> Yem Lyanis Tamayo Cárdenas	<b>Iteración Asignada:</b> 1

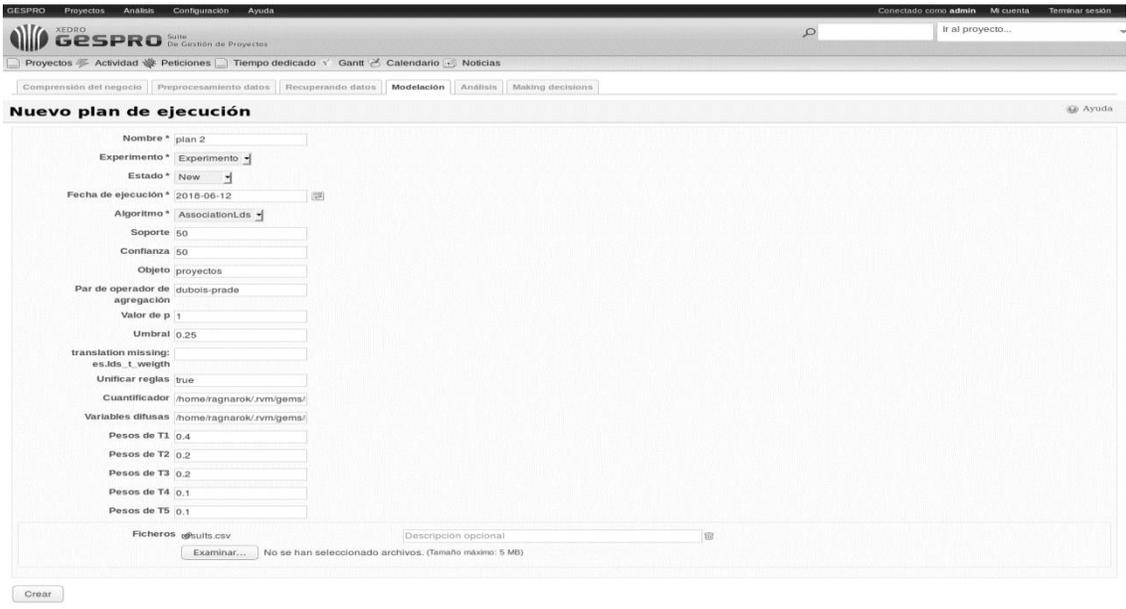
Pedro Alejandro Villavicencio Martínez	
<b>Prioridad:</b> Baja	<b>Tiempo Estimado:</b> 5 días.
<b>Riesgo en Desarrollo:</b> N/A	<b>Tiempo Real:</b> 3 días.
<p><b>Descripción</b></p> <p>Después de crear el plan de ejecución se carga el fichero csv guardado en una dirección específica sin tener que seleccionar los datos ni ejecutar la consulta select.</p> <p>Para ello debe existir un plan de ejecución y un fichero guardado en una dirección específica</p> <p><b>Observaciones:</b> Debe existir un plan de ejecución y un fichero guardado en una dirección específica.</p> <p><b>Prototipo de interfaz:</b></p> 	

Tabla 16.2.12 HU Modificar plan de ejecución.

**HU Modificar plan de ejecución.**

<b>Número:</b> 10	<b>Nombre del requisito:</b> Modificar plan de ejecución
<b>Programador:</b> Yem Liyanis Tamayo Cárdenas Pedro Alejandro Villavicencio Martínez	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Media	<b>Tiempo Estimado:</b> 5 días.
<b>Riesgo en Desarrollo:</b> N/A	<b>Tiempo Real:</b> 3 días.
<b>Descripción</b>	
<p>El usuario selecciona el botón modificar plan de ejecución, se llenan los campos obligatorios y se ejecuta el algoritmo escogido.</p> <p>Para ello debe existir en el módulo al menos un proyecto, el usuario debe ser miembro de algún proyecto y se deben llenar los campos obligatorios al modificar el plan.</p>	
<b>Observaciones:</b> Debe existir en el módulo al menos un proyecto, el usuario debe ser miembro de algún proyecto y se deben llenar los campos obligatorios al modificar el plan.	
<b>Prototipo de interfaz:</b>	

Comprensión del negocio | Comprensión de datos | Preprocesamiento de datos | **Modelación** | Análisis | Toma decisiones

### Editar plan de algoritmo Ayuda

Nombre \*

Experimento \*

Estado \*

Fecha de ejecución \*

Algoritmo \*

number\_components

Ficheros  No se han seleccionado archivos. (Tamaño máximo: 5 MB)

Project Management Suite (GESPRO 18.01). Research Laboratory in Project Management, UCI  
Powered by GESPRO, PostgreSql, D3, Redmine © 2006-2017 maestriaap@uci.cu, gestion.proyectos@uci.cu, app@uci.cu

Tabla 17.2.13 HU Eliminar plan de ejecución.

HU Eliminar plan de ejecución.	
<b>Número:</b> 11	<b>Nombre del requisito:</b> Eliminar plan de ejecución.
<b>Programador:</b> Yem Liyanis Tamayo Cárdenas Pedro Alejandro Villavicencio Martínez	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Media	<b>Tiempo Estimado:</b> 5 días.
<b>Riesgo en Desarrollo:</b> N/A	<b>Tiempo Real:</b> 3 días.

**Descripción**

El usuario selecciona el botón eliminar plan de ejecución y se muestra un mensaje indicando si está seguro que desea eliminar el plan y se selecciona la opción aceptar.

Para ello debe existir en el módulo al menos un proyecto, el usuario debe ser miembro de algún proyecto.

**Observaciones:** Debe existir en el módulo al menos un proyecto, el usuario debe ser miembro de algún proyecto.

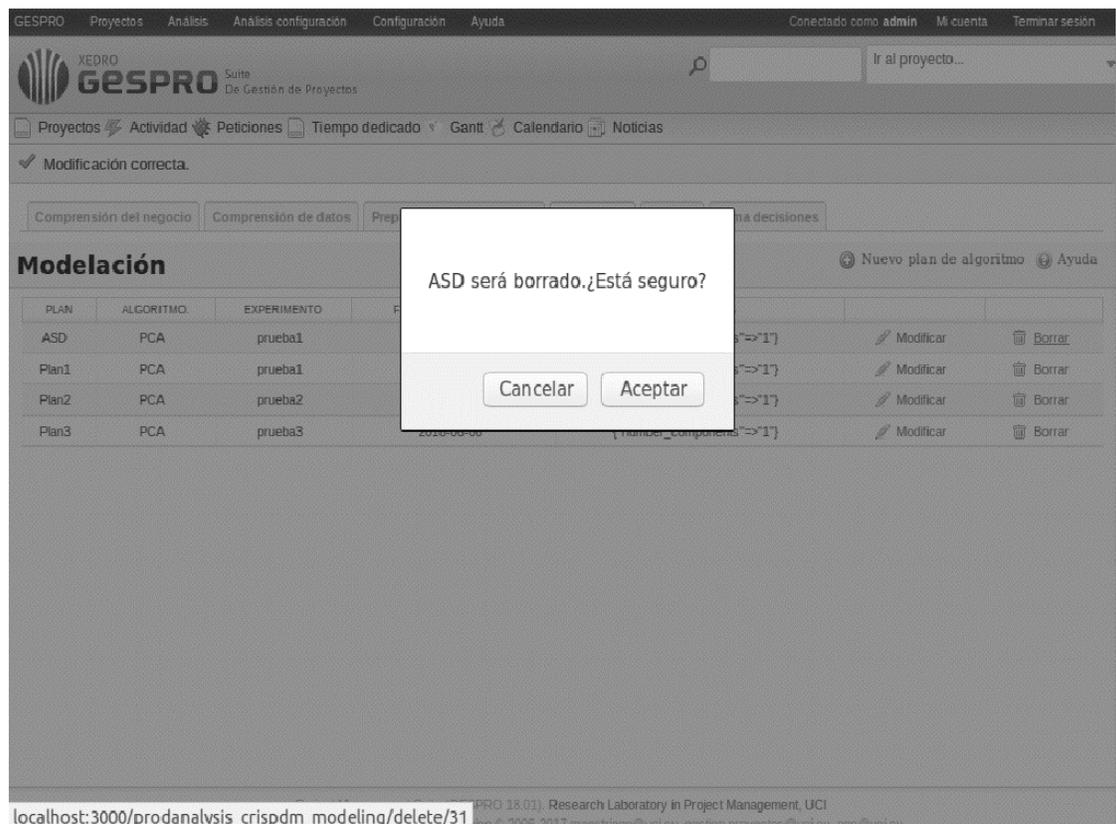
**Prototipo de interfaz:**

Tabla 18.2.14 HU Búsqueda avanzada

**HU Búsqueda avanzada.**

<b>Número:</b> 12	<b>Nombre del requisito:</b> Búsqueda avanzada.
<b>Programador:</b> Yem Liyanis Tamayo Cárdenas Pedro Alejandro Villavicencio Martínez	<b>Iteración Asignada:</b> 1
<b>Prioridad:</b> Media	<b>Tiempo Estimado:</b> 8 días.
<b>Riesgo en Desarrollo:</b> N/A	<b>Tiempo Real:</b> 5 días.
<p><b>Descripción</b></p> <p>El usuario escribe una palabra sobre lo que desea buscar, selecciona el botón aceptar y se muestra un listado de los experimentos y/o planes de ejecución relacionados.</p> <p>Para ello debe existir en el módulo al menos un proyecto, el usuario debe ser miembro de algún proyecto.</p>	
<p><b>Observaciones:</b> Debe existir en el módulo al menos un proyecto, el usuario debe ser miembro de algún proyecto.</p>	

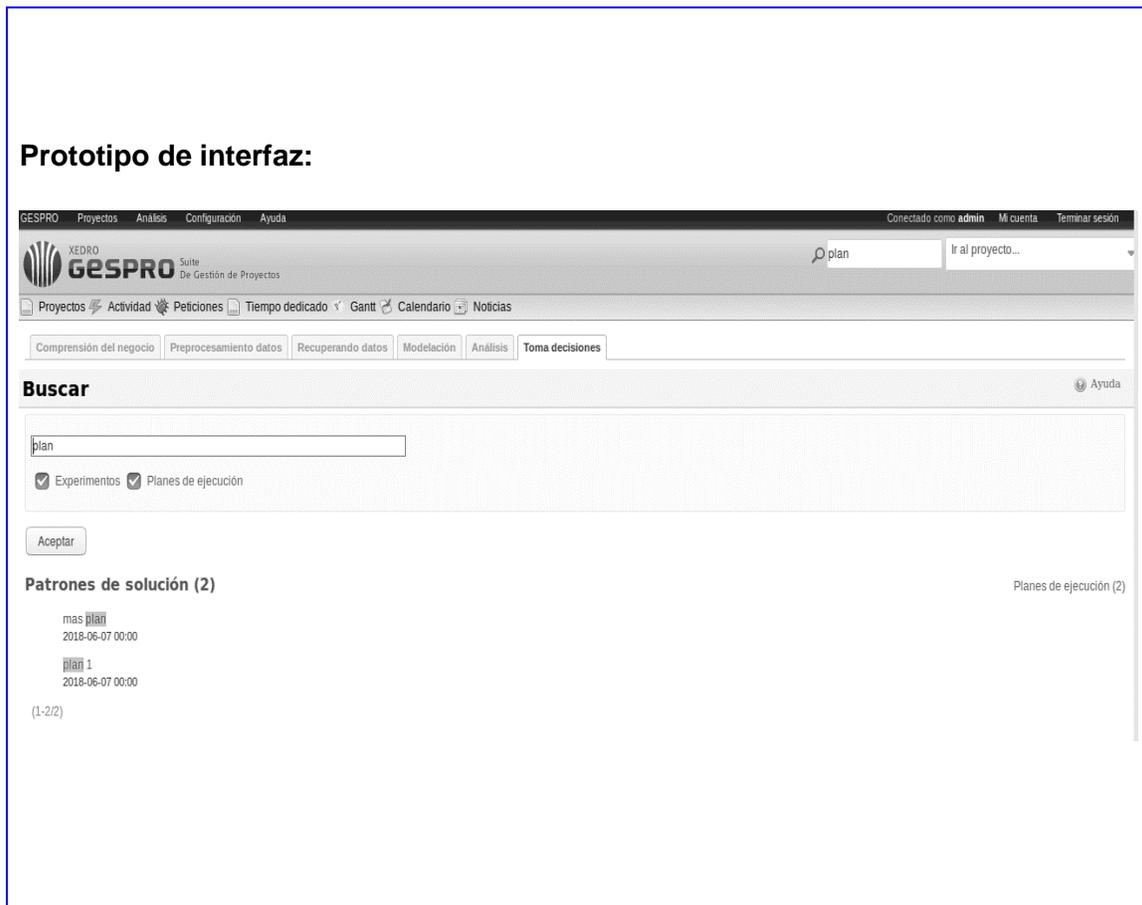


Tabla 19.3.1 Caso de prueba 4

<b>Caso de prueba de aceptación</b>
<b>Historia de usuario:</b> Registrar experimento.
<b>Nombre:</b> Registrar un nuevo experimento.
<b>Responsable:</b> Yem Liyanis Tamayo Cárdenas.
<b>Descripción:</b> Esta prueba se realiza con el objetivo de verificar si la funcionalidad de crear un nuevo experimento funciona conforme a lo esperado.
<b>Condiciones de ejecución:</b> Establecer la conexión a las bases de datos predeterminadas.
<b>Entradas/Pasos de ejecución:</b> Seleccionar del menú la opción Nuevo experimento.
<b>Resultado esperado:</b> Se muestran los experimentos ya creados así como el experimento recién creado.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

Tabla 20.3.2 Caso de prueba 5

<b>Caso de prueba de aceptación</b>
<b>Historia de usuario:</b> Registrar experimento.
<b>Nombre:</b> Registrar un nuevo experimento.
<b>Responsable:</b> Yem Liyanis Tamayo Cárdenas.
<b>Descripción:</b> Esta prueba se realiza con el objetivo de verificar si la funcionalidad de crear un nuevo experimento funciona relleno con datos incompletos.
<b>Condiciones de ejecución:</b> Establecer la conexión a las bases de datos predeterminadas.
<b>Entradas/Pasos de ejecución:</b> Seleccionar del menú la opción Nuevo experimento.
<b>Resultado esperado:</b> Se muestra el mensaje "Campos incompletos"
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

Tabla 21.3.3 Caso de prueba 6

<b>Caso de prueba de aceptación</b>
<b>Historia de usuario:</b> Modificar experimento.
<b>Nombre:</b> Modificar un experimento.
<b>Responsable:</b> Yem Liyanis Tamayo Cárdenas.
<b>Descripción:</b> Esta prueba se realiza con el objetivo de verificar si la funcionalidad de modificar un experimento funciona conforme a lo esperado.
<b>Condiciones de ejecución:</b> Establecer la conexión a las bases de datos predeterminadas.
<b>Entradas/Pasos de ejecución:</b> Seleccionar del menú Comprensión del negocio la opción Modificar.
<b>Resultado esperado:</b> Se muestran los campos a modificar del experimento.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

Tabla 22.3.4 Caso de prueba 7

<b>Caso de prueba de aceptación</b>
<b>Historia de usuario:</b> Modificar experimento.
<b>Nombre:</b> Modificar un experimento.
<b>Responsable:</b> Yem Liyanis Tamayo Cárdenas.
<b>Descripción:</b> Esta prueba se realiza con el objetivo de verificar si la funcionalidad de modificar un experimento funciona con datos incompletos.
<b>Condiciones de ejecución:</b> Establecer la conexión a las bases de datos predeterminadas.
<b>Entradas/Pasos de ejecución:</b> Seleccionar del menú Comprensión del negocio la opción Modificar.
<b>Resultado esperado:</b> Se muestra el mensaje "Datos incompletos".
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

Tabla 23.3.5 Caso de prueba 8

<b>Caso de prueba de aceptación</b>
<b>Historia de usuario:</b> Eliminar experimento.
<b>Nombre:</b> Eliminar un experimento.
<b>Responsable:</b> Yem Liyanis Tamayo Cárdenas.
<b>Descripción:</b> Esta prueba se realiza con el objetivo de verificar si la funcionalidad de eliminar un experimento funciona conforme a lo esperado.
<b>Condiciones de ejecución:</b> Establecer la conexión a las bases de datos predeterminadas.
<b>Entradas/Pasos de ejecución:</b> Seleccionar del menú Comprensión del negocio la opción Eliminar.
<b>Resultado esperado:</b> Se muestra el mensaje "Nombre del experimento' será borrado. ¿Está seguro?", y es eliminado el experimento.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

Tabla 24.3.6 Caso de prueba 9

<b>Caso de prueba de aceptación</b>
<b>Historia de usuario:</b> Seleccionar los datos a utilizar en el experimento.
<b>Nombre:</b> Seleccionar los datos a utilizar en el experimento.
<b>Responsable:</b> Pedro Alejandro Villavicencio Martínez.
<b>Descripción:</b> Esta prueba se realiza con el objetivo de verificar si la funcionalidad de seleccionar los datos a utilizar en un experimento funciona conforme a lo esperado.
<b>Condiciones de ejecución:</b> Establecer la conexión a las bases de datos predeterminadas.
<b>Entradas/Pasos de ejecución:</b> Seleccionar del menú Preprocesamiento de los datos y seleccionar las tablas así como los datos de la misma.
<b>Resultado esperado:</b> Se muestran los datos de las tablas escogidas.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

Tabla 25.3.7 Caso de prueba 10

<b>Caso de prueba de aceptación</b>
<b>Historia de usuario:</b> Mostrar datos de consultas Select.
<b>Nombre:</b> Mostrar las consultas a utilizar en el experimento.
<b>Responsable:</b> Pedro Alejandro Villavicencio Martínez.
<b>Descripción:</b> Esta prueba se realiza con el objetivo de verificar que se puedan hacer consultas para seleccionar los datos a utilizar en un experimento funciona conforme a lo esperado.
<b>Condiciones de ejecución:</b> Establecer la conexión a las bases de datos predeterminadas.
<b>Entradas/Pasos de ejecución:</b> Seleccionar del menú Recuperando datos y realizar las consultas select.
<b>Resultado esperado:</b> Se muestran los resultados de las consultas select escogidas.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

Tabla 26.3.8 Caso de prueba 11

<b>Caso de prueba de aceptación</b>
-------------------------------------

<b>Historia de usuario:</b> Mostrar datos de consulas Select.
<b>Nombre:</b> Mostrar las consultas a utilizar en el experimento.
<b>Responsable:</b> Pedro Alejandro Villavicencio Martínez.
<b>Descripción:</b> Esta prueba se realiza con el objetivo de verificar que solo se puedan hacer consultas select.
<b>Condiciones de ejecución:</b> Establecer la conexión a las bases de datos predeterminadas.
<b>Entradas/Pasos de ejecución:</b> Seleccionar del menú Recuperando datos y realizar las consultas que no sean select.
<b>Resultado esperado:</b> Se muestra el mensaje “Existen restricciones para realizar SQL, no puede usar UPDATE, DELETE, DROP o INSERT”.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

Tabla 27.3.9 Caso de prueba 12

<b>Caso de prueba de aceptación</b>
<b>Historia de usuario:</b> Registrar plan de ejecución.
<b>Nombre:</b> Registrar plan de ejecución.
<b>Responsable:</b> Yem Liyanis Tamayo Cárdenas.
<b>Descripción:</b> Esta prueba se realiza con el objetivo de verificar que se puede registrar un nuevo plan de ejecución.
<b>Condiciones de ejecución:</b> Establecer la conexión a las bases de datos predeterminadas.
<b>Entradas/Pasos de ejecución:</b> Seleccionar del menú Nuevo plan de ejecución y llenar los campos predeterminados.
<b>Resultado esperado:</b> Se muestra el nuevo plan de ejecución creado.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

Tabla 28.3.10 Caso de prueba 13

<b>Caso de prueba de aceptación</b>
<b>Historia de usuario:</b> Registrar plan de ejecución.
<b>Nombre:</b> Registrar plan de ejecución.

<b>Responsable:</b> Pedro Alejandro Villavicencio Martínez.
<b>Descripción:</b> Esta prueba se realiza con el objetivo de verificar que no se pueda registrar un plan de ejecución con campos vacíos.
<b>Condiciones de ejecución:</b> Establecer la conexión a las bases de datos predeterminadas.
<b>Entradas/Pasos de ejecución:</b> Seleccionar del menú Modificar y llenar los campos predeterminados.
<b>Resultado esperado:</b> Se muestra el mensaje “Campos vacíos”.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

Tabla 29.3.11 Caso de prueba 14

<b>Caso de prueba de aceptación</b>
<b>Historia de usuario:</b> Mostrar resultados del plan de ejecución.
<b>Nombre:</b> Mostrar los resultados del plan de ejecución.
<b>Responsable:</b> Yem Liyanis Tamayo Cárdenas.
<b>Descripción:</b> Esta prueba se realiza con el objetivo de verificar que se muestran los resultados del plan de ejecución.
<b>Condiciones de ejecución:</b> Establecer la conexión a las bases de datos predeterminadas.
<b>Entradas/Pasos de ejecución:</b> Seleccionar del menú Análisis.
<b>Resultado esperado:</b> Se muestran los resultados del plan de ejecución.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

Tabla 30.3.12 Caso de prueba 15

<b>Caso de prueba de aceptación</b>
<b>Historia de usuario:</b> Mostrar resultados del plan de ejecución.
<b>Nombre:</b> Mostrar los resultados del plan de ejecución.
<b>Responsable:</b> Pedro Alejandro Villavicencio Martínez.
<b>Descripción:</b> Esta prueba se realiza con el objetivo de verificar que no se muestran los resultados del plan de ejecución utilizando datos incompatibles.

<b>Condiciones de ejecución:</b> Establecer la conexión a las bases de datos predeterminadas.
<b>Entradas/Pasos de ejecución:</b> Seleccionar del menú Análisis.
<b>Resultado esperado:</b> Se muestran los resultados del plan de ejecución.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

Tabla 31.3.13 Caso de prueba 16

<b>Caso de prueba de aceptación</b>
<b>Historia de usuario:</b> Modificar plan de ejecución.
<b>Nombre:</b> Modificar plan de ejecución.
<b>Responsable:</b> Yem Liyanis Tamayo Cárdenas.
<b>Descripción:</b> Esta prueba se realiza con el objetivo de verificar que se puede modificar un plan de ejecución.
<b>Condiciones de ejecución:</b> Establecer la conexión a las bases de datos predeterminadas.
<b>Entradas/Pasos de ejecución:</b> Seleccionar del menú Modificar y llenar los campos predeterminados.
<b>Resultado esperado:</b> Se muestra la modificación del plan de ejecución.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

Tabla 32.3.14 Caso de prueba 17

<b>Caso de prueba de aceptación</b>
<b>Historia de usuario:</b> Modificar plan de ejecución.
<b>Nombre:</b> Modificar plan de ejecución.
<b>Responsable:</b> Yem Liyanis Tamayo Cárdenas.
<b>Descripción:</b> Esta prueba se realiza con el objetivo de verificar que no se pueda modificar un plan de ejecución rellenando con datos incompletos.
<b>Condiciones de ejecución:</b> Establecer la conexión a las bases de datos predeterminadas.
<b>Entradas/Pasos de ejecución:</b> Seleccionar del menú Modificar y llenar los campos predeterminados.

<b>Resultado esperado:</b> Se muestra el mensaje “Nombre es requerido”.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

Tabla 33.3.15 Caso de prueba 18

<b>Caso de prueba de aceptación</b>
<b>Historia de usuario:</b> Eliminar plan de ejecución.
<b>Nombre:</b> Eliminar plan de ejecución.
<b>Responsable:</b> Pedro Alejandro Villavicencio Martínez.
<b>Descripción:</b> Esta prueba se realiza con el objetivo de verificar que se puede registrar un nuevo plan de ejecución.
<b>Condiciones de ejecución:</b> Establecer la conexión a las bases de datos predeterminadas.
<b>Entradas/Pasos de ejecución:</b> Seleccionar del menú Nuevo plan de ejecución y llenar los campos predeterminados.
<b>Resultado esperado:</b> Se muestra el nuevo plan de ejecución creado.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

Tabla 34.3.16 Caso de prueba 19

<b>Caso de prueba de aceptación</b>
<b>Historia de usuario:</b> Buscar soluciones.
<b>Nombre:</b> Buscar soluciones.
<b>Responsable:</b> Pedro Alejandro Villavicencio Martínez.
<b>Descripción:</b> Esta prueba se realiza con el objetivo de verificar que se muestran resultados dada una búsqueda.
<b>Condiciones de ejecución:</b> Establecer la conexión a las bases de datos predeterminadas.
<b>Entradas/Pasos de ejecución:</b> Seleccionar del menú Toma de decisiones y llenar el campo predeterminado.
<b>Resultado esperado:</b> Se muestra el resultado de la búsqueda.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.