**Universidad de las Ciencias Informáticas**


**Facultad 3**

# Trabajo de Diploma para optar por el título de Ingeniero en Ciencias Informáticas.

Desarrollo del Sistema de Recomendación de equipos de investigación para tesis de grado.

**Autor:** Alejandro Iglesias Mizrahi

**Tutor:** Ing. Yadira Beatriz Reyes García

***La Habana, julio de 2016***

***"Año 58 de la Revolución"***

## Declaración de Autoría

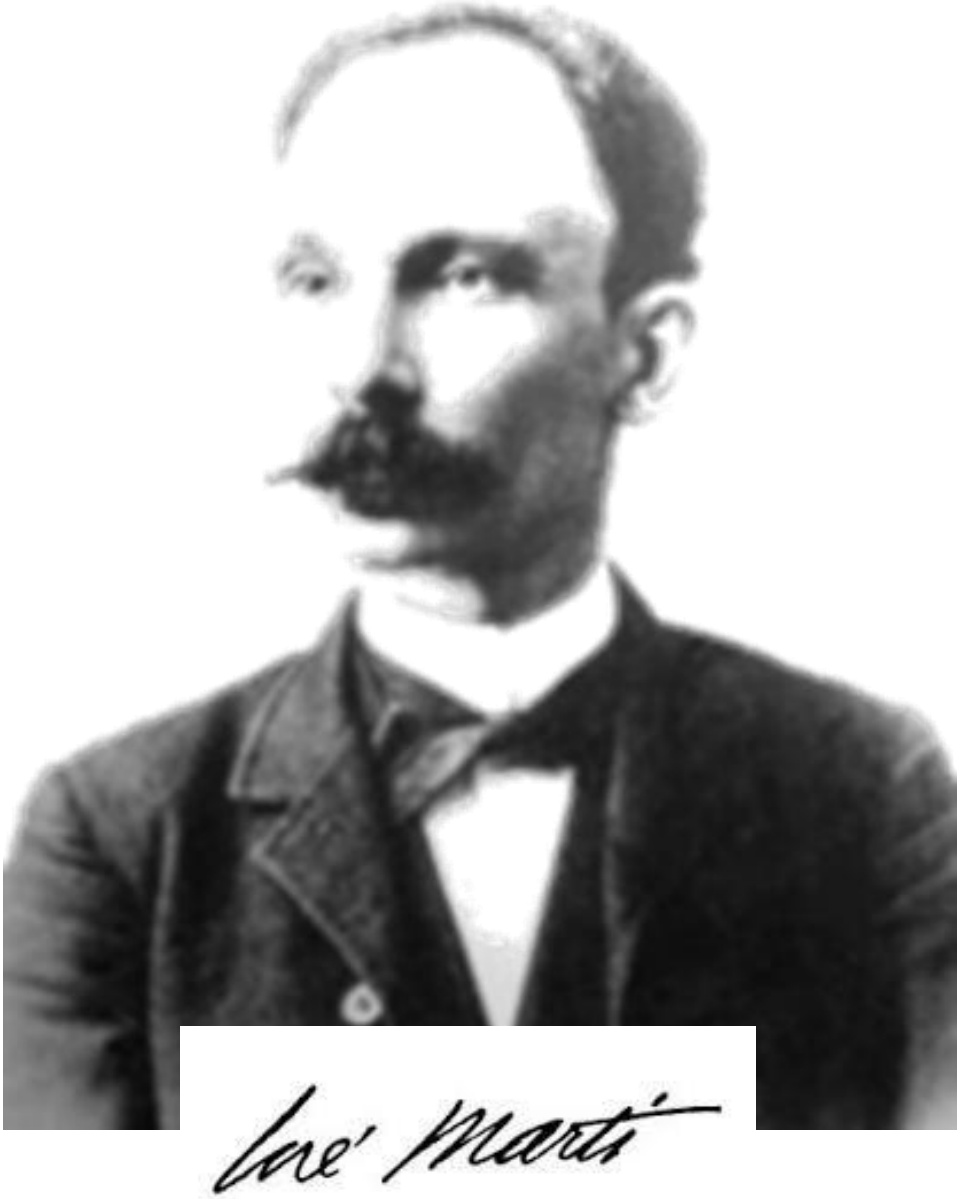
Declaro ser el único autor del presente trabajo de diploma y autorizo a la Facultad 3 de la Universidad de las Ciencias Informáticas a hacer uso con carácter exclusivo delos derechos patrimoniales del mismo en su beneficio.

Para que así conste firmo el presente a los ___ días del mes de _____ del año 2016.

_____ _____

Firma del autor Firma del tutor

Alejandro Iglesias MizrahiIng. Yadira Beatriz Reyes García




*"El único autógrafo digno de un hombre es aquel que deja escrito con sus obras."*

*José Martí (1853-1895)*

## Datos de contacto

**Ing. Yadira Beatriz Reyes García**: Graduada de Ingeniería de Ciencias Informáticas de la UCI en la Primera Graduación del Curso 2006-2007. Trabaja en la facultad 3 del propio centro de enseñanza superior. Se ha desempeñado como profesora del departamento de la Ingeniería de Software impartiendo asignaturas vinculadas al mismo, así como las asignaturas de Proyecto de Investigación y Desarrollo e Introducción a las Ciencias Informáticas. Ha ejercido como tutor, oponente y miembro de tribunal en varias tesis de grado. Posee varias publicaciones científicas de carácter nacional e internacional.

Correo electrónico: yreyesg@uci.cu

## Agradecimientos

**A todas aquellas personas que me ayudaron a llegar hasta aquí, a los que compartieron conmigo todo este tiempo en esta gran universidad. A mi familia, mis amigos, mi tutora, a la Revolución Cubana. Gracias por todo, esto es para ustedes…**

*Ale*

## Dedicatoria

**A mi madre y a mi abuela, las dos protagonistas más grandes de la historia que ha sido mi vida, gracias por estar siempre ahí, ahora me toca a mi estar ahí para ustedes…**

## Resumen

Los sistemas de recomendaciónson herramientas de software y técnicas que tienen el objetivo de brindar determinadas sugerencias, para ayudara obtener la información que necesita un usuario sobre un determinado grupo de elementos para que sirvan de ayuda al proceso natural de toma de decisiones. El objetivo del presentetrabajo de diploma es desarrollar un sistema de recomendacióncomo apoyo al proceso de conformación deequipos de investigación,y que favorezca un mejor desarrollo del trabajo de diploma de los estudiantes de quinto año.Cabe destacar que el sistema no pretende conformar los equipos de investigación sino ser una herramienta útil en la toma de decisiones basada en el análisis de datos y en técnicas de inteligencia artificial empleadas a la hora de realizar las recomendaciones. Para la concepción e implementación de la herramienta se realizó un estudio de los sistemas de recomendación, así como de técnicas para el análisis de datos que posibilitaron darle solución al problema en cuestión y generar las recomendaciones.Asimismo, fueron valoradas las herramientas, tecnologías y metodología para su desarrollo y finalmente fueron utilizadas métricas para evaluar los requisitos y el diseño de la solución, así como la validación de su correcto funcionamiento mediante las clásicas pruebas de análisis de calidad de software.


**Palabras clave:** *sistema recomendación, toma de decisiones, equipo de investigación, inteligencia artificial*


## Abstract

Recommender systems are software tools and techniques which are used to provide some suggestions to help people to get the information needed, according to a certain group of items used as a help to the natural process of decision making. The goalin this diploma thesis is to develop a recommender system in a way of support to the process of creation of research teams, in order to facilitate favor a better development of research and accomplishment of diploma thesis for students of fifth year. Is important to realizethat the system is not intended to form research teams themselves but being a useful tool to support decision making based on data analysis and artificial intelligence techniques employed when making recommendations. For the realization of the tool itself was achieved a study of the recommender systems and techniques for analyzing data that permitted giving solution to the problem in question and generate the proper recommendations requested. Likewise, tools, technologies and methodologies were valued for its development and were eventually used metrics to evaluate requirements and design of the released software, as well as validation of its correct performance by conventional software analysis quality tests.


**Keywords:***recommender system, decision making, research team, artificial intelligence*

# Índice de contenidos

## Índice de figuras

## Índice de tablas

# Introducción

En el mundo y como parte de la interacción social las personas se comunican entre sí para intercambiar información, por lo tanto, casi siempre hay una posibilidad potencial de pedir sugerencias u opiniones, ya sea para tomar decisiones o valorar posibilidades. En el entorno informático este proceso es realizado por los sistemas de recomendación. Estos tratan de ser una alternativa al proceso social de recomendación: el acto habitual de pedir a conocidos o expertos su opinión para tomar una decisión sin necesidad de tener la suficiente información para ello. Generalmente son decisiones sencillas como qué libro leer, qué música escuchar, qué contenido multimedia ver o qué lugares visitar. En general, el uso de los sistemas de recomendación se ha enfocado, dado el amplio uso que tienen las redes sociales a recomendarles a sus usuarios contenido específico que pueden encontrar en la red. Esto es posible mediante valoraciones que haga el propio usuario sobre lo que prefiere o recomendaciones del sistema sobre lo que prefieren usuarios con características similares.

Otro factor importante es que el uso de los sistemas de recomendación como herramientas que apoyan el proceso de toma de decisiones, favorece la valoración de propuestas y obtención de información relevante cuando se cuenta con un gran cúmulo de datos, dado que estos siguen patrones de evaluación de la información que a las personas les pudieran resultar difícil de encontrar por sí mismos. Sin embargo, en la educación y los efectos que puedenteneren la mejora de los resultados de los educandos, tras la investigación llevada a cabo para la realización de este trabajo, que se ha profundizado muy poco, utilizándose principalmente estos en el área del aprendizaje electrónico o E-learning.

Otra aplicación de los sistemas de recomendación puede darse en la gestión del capital humano. Dado que puede utilizarse en la conformación de equipos de desarrollo, investigación, analistas, etc. para logar un balance cualitativo en cualquier área o departamento y así lograr un mejor resultado en la productividad, la eficiencia y la calidad en cada una de estas, ya que muchas veces la gestión de los recursos humanos en estos casos se realiza de forma empírica y con poca o ninguna base científica. Por lo tanto, puesto que nuestro país comienza a abrirse un camino para lograr la informatización de la sociedad, el uso de sistemas de recomendación puede ser una herramienta para mejorar los resultados en cualquier esfera productiva del país, elevando así la calidad de las prestaciones utilizando menos recursos.

La Universidad de las Ciencias Informáticas (UCI), para cumplir con su misión estratégica de formar profesionales comprometidos con su Patria, altamente calificados en la rama de la informática y de producir software y servicios informáticos, a partir de la vinculación estudio-trabajo como modelo de formación, requiere de cambios profundos en su modelo actual, transitando hacia la integración de los

procesos de formación, producción e investigación. En este último aspecto la dirección de investigaciones de la UCI ha estado trabajando en perfeccionar sus procesos, dígase la publicación de la revista científica, la reorganización de las líneas de investigación, los grupos asociados a ellas, la gestión de las publicaciones mediante el Sistema de Gestión Universitaria; dirigiendo sus esfuerzos a organizar las investigaciones en la universidad estimulando la participación de profesores y estudiantes.

Los profesionales de la UCI publican sus artículos en diferentes revistas, en memorias de eventos, asesoran las tesis de grado, tutoran estudiantes en su Práctica Profesional, y sin embargo cuando se va a caracterizar al Profesional es difícil decidir en qué Línea de Investigación se encuentra trabajando o en qué áreas del conocimiento ha estudiado. Esta situación dificulta el proceso de asignación de temas de investigación a estudiantes que estén en el ciclo profesional, tanto para su Proyecto de Investigación y Desarrollo (PID) como para su trabajo de culminación de estudios. En este sentido la selección de un tema de investigación para el estudiante conociendo su recorrido también es engorroso, pues hay que consultar su recorrido estudiantil para saber a qué tipo de investigación se debe asignar. Pero no siempre es así, y el proceso de asignación en las facultades puede ser de las siguientes formas:

1. Los estudiantes son seleccionados por los tutores por sus resultados docentes en asignaturas claves para la investigación,de forma empírica, por afinidad o porque conocen su línea de investigación y están asociados a ella.

2. Los estudiantes son asignados a temas de tesis según la ubicación física donde se encuentren, atendiendo a la distribución de proyectos en la asignatura PID.

Además, para la asignación de tesis de pregrado no existe un correcto desglose de las pirámides de investigación. Los temas de tesis son publicados en quinto año, ya cuando el estudiante debe desarrollar y no antes para que pueda ampliar sus conocimientos y su interacción con la esfera investigativa sea más profunda. Por otra parte, en el proceso de asignación de temas se pudieran considerar a los estudiantes todos por igual obviando el principio psicológico de que los todos individuos son diferentes y la manera de tratar con estos debe ser diferenciada.

Esta situación produce demora en la investigación, poca motivación, falta de comunicación entre estudiantes y tutores, entre otros factores que influyen la calidad de del desarrollo del proceso de desarrollo de software y puede atentar contra el éxito del resultado científico.

Por estas razones surge la necesidad de agrupar a los profesionales por sus líneas de investigación para así saber las temáticas que están trabajando y poder reorientarlo a los grupos de investigación ya existentes y se obtendría además, con la correcta conformación de equipos de investigación provenientes de pirámides de maestrías y doctorados donde se lleva un estudio de los miembros del

equipo y se trabajaría con estudiantes de los diferentes años de la carrera, de tal forma que al llegar al 5to año el estudiante presente características bien definidas para que sea recomendado a un tutor con ciertas similitudes.

En correspondencia con el contexto descrito, se identificó el siguiente **problema a resolver:** ¿Cómo contribuir a la toma de decisiones en el proceso de asignación de tesis de pregrado teniendo en cuenta el desempeño científico y académico de tutores y estudiantes?

La parte de la ciencia que será el**objeto de estudio**es elproceso de desarrollo de los sistemas de recomendación, enmarcándose en el**campo de acción:** sistemas de recomendación de apoyo al proceso de asignación de tesis de pregrado.

Para resolver el problema identificado se definió el siguiente **objetivo general:** desarrollar un Sistema de Recomendación de equipos de investigación para tesis de grado que contribuya a la toma de decisiones en el proceso de asignación de tesis de pregrado teniendo en cuenta el desempeño académico y científico de tutores y estudiantes.

Para darle solución al objetivo general se definen los siguientes **objetivos específicos**:

1. Establecer el marco teórico-conceptual para fundamentar la investigación a partir del estudio de los sistemas de recomendación y de técnicas de inteligencia artificial para descubrimiento de conocimiento.
2. Realizar el análisis y diseño de la estructura de la solución de software para facilitar la implementación del sistema.
3. Implementar las funcionalidades de la herramienta utilizando la información obtenida en el proceso de descubrimiento de conocimiento para realizar las recomendaciones.
4. Validar la solución de software propuesta mediante la aplicación de técnicas de evaluación dinámicas y prueba de aceptación del cliente.

Para darle cumplimiento a los **objetivos específicos** se plantean las siguientes **tareas de la investigación:**

1. Estudio del estado del arte sobre los sistemas de recomendación en el área educacional.
2. Estudio del proceso de descubrimiento de conocimiento a través de técnicas y herramientas de minería de datos.
3. Definición de las tecnologías y herramientas a utilizar en el desarrollo de la solución de software.
4. Especificación y validación de los requisitos del sistema mediante atributos de calidad.

5. Confección de los diagramas de clases del diseño para definir las clases y sus relaciones. Evaluación del diseño mediante métricas.
6. Diseño de la interfaz de usuario. Realización del diseño de la base de datos.
7. Implementación de la solución de software.
8. Validación de la calidad de la solución propuesta mediante la aplicación de las técnicas de evaluación de Caja Negra y Caja Blanca.
9. Realización de pruebas de aceptación al cliente relativas a su conformidad con la herramienta desarrollada.

Para guiar la investigación se plantea la siguiente **idea a defender:** Con el desarrollo de la herramienta Sistema de Recomendación de equipos de investigación para tesis de grado, teniendo en cuenta el desempeño académico y científico de tutores y estudiantes, se contribuirá a mejorar la toma de decisiones en el proceso de asignación de tesis de pregrado.

Para el desarrollo satisfactorio del presente trabajo de diploma se emplearon métodos científicos tanto teóricos como empíricos buscando que se mantuviera un balance entre lo cuantitativo y lo cualitativo de la información obtenida a través de la investigación.

## Métodos Teóricos

### Método Histórico-Lógico

Enestainvestigaciónserealizaunestudiodelestadodel arte,conelobjetivodeanalizaryprofundizarenlaevoluciónquehantenidolosdiferentestemasasociadosalos sistemas de recomendación y la utilización de técnicas de inteligencia artificial para su confección. Así como de la información asociada a las metodologías, herramientas y validaciones a usar en la solución.

### Método Analítico-Sintético

Se aplicó para todas las fuentes de información: documentos, artículos, trabajos científicos e investigaciones y literatura científica en general. Posteriormente se realizó un profundo análisis de los sistemas de recomendación, sus características, funcionamiento, uso, y ventajas, además del estudio de los algoritmos de clasificación bajo el enfoque supervisado y su funcionamiento. De esta forma se logró entender el tema, facilitando su estudio y definiendo una estrategia para llegar al resultado final.

### Método de Modelación

La utilización de este método posibilitó la obtención de los principales artefactos de cada una de lasfases de desarrollo del sistema.

## Métodos Empíricos

### Entrevistas

Se utilizó a la hora de realizar entrevistas y para obtener de personal calificado, la información necesaria para que deberá cumplir la aplicación para garantizar que la solución propuesta pueda satisfacer las necesidades existentes(Hernández, Coello 2005).

### Medición

Con el objetivo de obtener información numérica acerca del proceso de asignación de tesis en las facultades 1 y 3, y otros datos que se precisaron a la hora de realizar el trabajo de diploma. Además, se utiliza para evaluar el comportamiento del software diseñado y su pertinencia con los requisitos establecidos.

**Estructuración por capítulos**

La tesis consta de introducción, trescapítulos, conclusiones, recomendaciones, referencias bibliográficas, glosario de términos y anexos.El trabajo de diploma queda estructurado de la siguiente forma:

**Capítulo 1. Fundamentación Teórica:** En este capítulo se realiza un estudio sobre los sistemas de recomendación,el proceso de descubrimiento de conocimiento y las técnicas de inteligencia artificial a emplear en la implementación, así como se detallan y seleccionan herramientas, metodologías y tecnologías que se utilizarán en la solución final.

**Capítulo 2. Propuesta de solución:** En este capítulo se exponen las características del sistema, incluidos los requisitos funcionales y no funcionales, se realiza la descripción de la solución, además de los patrones de diseño que se utilizaron y artefactos que plantea la metodología escogida para el desarrollo.

**Capítulo 3. Validación de la solución:** En este capítulo se validan los requisitos y el diseño del sistema, se especifica el estándar de codificación empleado en la implementación, se obtienen los casos de prueba que se aplican a este, buscando validar la solución dada y además se realiza un análisis de los resultados obtenidos para determinar la implicación de los mismos en el posterior uso del sistema.

# Capítulo 1. Fundamentación Teórica

El objetivo de este capítulo es identificar los conceptos y elementos teóricos que permitan entender lascaracterísticas principales de un sistema de recomendación,siendo necesarios además para la realización y descripción del mismo. También se realiza una descripción de las técnicas de minería de datos e inteligencia artificial a utilizar tanto en la extracción de conocimiento como en la implementación de la solución. Por último, se describen las principales metodologías y herramientas utilizadas en la solución propuesta.

## 1.1 Sistema de Recomendación

Un primer concepto enunciado por Paul Resnick y Hal Varian en los tiempos del surgimiento de los sistemas de recomendación (SR) enuncia que: es aquel sistema que utiliza las opiniones de los usuarios de una comunidad para ayudar a usuarios de esa comunidad a encontrar contenidos de su gusto entre un conjunto sobrecargado de posibles elecciones(Martínez 2014).

Otro concepto extraído de un trabajo realizado por Ricardo Moya en el 2014refleja que: un SR es un sistema inteligente que proporciona a los usuarios una serie de sugerencias personalizadas (recomendaciones) sobre un determinado tipo de elementos (ítems). Los sistemas de recomendación estudian las características de cada usuario y mediante un procesamiento de los datos, encuentra un subconjunto de ítems que puedan resultar de interés para el usuario(Moya 2014). Véase **Figura 1**.

El autor de la presente investigación considera que es necesario resaltar que en el caso de la investigación no se tratará el método tradicional de estudio del comportamiento de usuarios en las redes sociales típicas como Facebook, Twitter, etc. sino de redes sociales institucionales como por ejemplo el caso del Sistema de Gestión Universitaria en la UCI. Por lo cual, se asume como concepto más apropiado entre todos los estudiados para definir a los SR el enunciado por Moya.

*“Uno de los principales retos que tienen que afrontar los Sistemas de Recomendación es la gestión eficaz del enorme volumen de datos que almacenan, con el objetivo de facilitar a los consumidores la información que satisfaga sus necesidades de una manera rápida y sencilla. Esta necesidad obtiene una mayor importancia en una sociedad como la actual, donde el nivel de exigencia de los usuarios es cada vez mayor*(Herlocker, Konstan, Terveen, Riedl 2004)*.”*

### 1.1.1Origen de los sistemas de recomendación

El origen de este tipo de sistema se remonta a principios de la década de los 90, servicios de filtrado de noticias que permitían a su comunidad de usuarios acceder exclusivamente a aquellas noticias que potencialmente podían ser de su interés. El primer sistema que surgió en este ámbito fue el llamado

Tapestry, sistema que permite almacenar la retroalimentación de los usuarios sobre artículos o noticias interesantes para ellos, donde estos artículos o noticias pueden ser utilizados por otros usuarios para determinar si la información del documento es relevante o no(Font 2009).

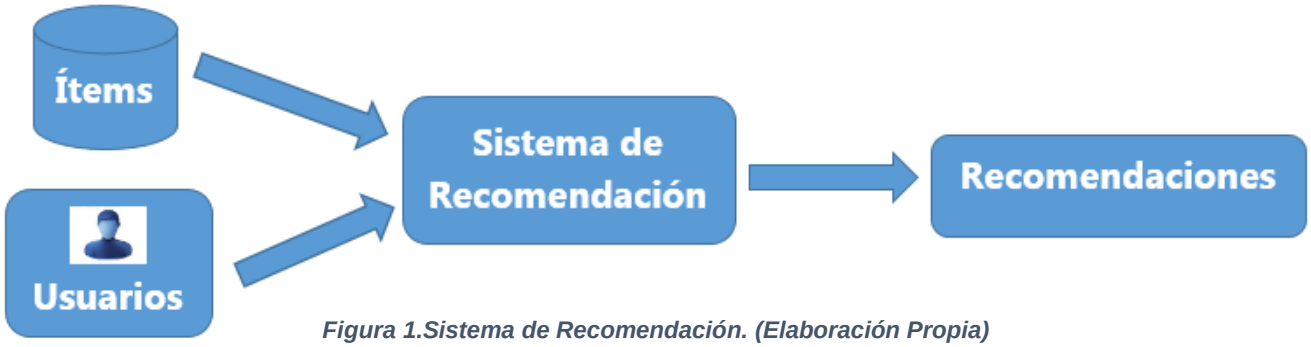



*Figura 1.Sistema de Recomendación. (Elaboración Propia)*

Otra de las primeras formas de filtrado de información electrónica apareció con el trabajo de Edward M. Housman y Elaine D. Kaskela, en el que se diseñó un método que de forma automática se pudiera mantener a los científicos informados sobre nuevos documentos publicados en sus áreas de trabajo o especialización. Este método se basaba en la creación de un perfil de usuario que contenía ciertas palabras clave relevantes para el usuario, que son utilizadas para buscar coincidencias entre estas y los nuevos documentos o artículos, con el fin de intentar predecir cuáles de las informaciones encontradas serían del interés para los científicos(Font 2009).

Ya en 1997, Paul Resnick y Hal R. Varian (pioneros en la implementación y estudio de los primeros sistemas de recomendación) proponen llamar a estas técnicas con el nombre de Sistema de recomendación por dos razones: en primer lugar, porque puede ocurrir que los usuarios no colaboren explícitamente entre ellos y, ensegundo lugar, porque el sistema puede sugerir elementos no conocidos hasta el momento por el usuario y en ese caso se estaría realizando una recomendación(Bertate, Machado, Molina 2006).

### 1.1.2Tipos de sistemas de recomendación

A partir de la bibliografía consultada y considerando los aspectos establecidos anteriormente, los sistemas de recomendación pueden recibir diferentes clasificaciones dependiendo del tipo de información que utilizan para realizar sus recomendaciones. De todas las formas de clasificación revisadas la más común es la que se expone en la

**Figura** *2*:

**Sistemas basados en contenido**

Los sistemas de recomendación basados en el contenido intentan recomendar elementos que son similares a aquellos que el usuario previamente valoró en el pasado (o está examinando en la actualidad). Se enfocan en algoritmos de aprendizaje de preferencias de usuarios y filtran nuevos

elementos, teniendo en cuenta los que más satisfacen las preferencias de estos. Entre las ventajas que posee este tipo de SR se puede destacar que el sistema puede generar recomendaciones sin la necesidad de contar con un historial previo, por lo que permite realizar predicciones independientemente del historial del usuario. Como principal desventaja se tiene la sobre-especialización, donde el usuario está limitado a que le recomienden ítems similares a los que recomendó. Por tanto, las recomendaciones suelen ser siempre muy similares dado que se basan en la misma información(Peinado, Coavas, Coronado, Núñez 2011).

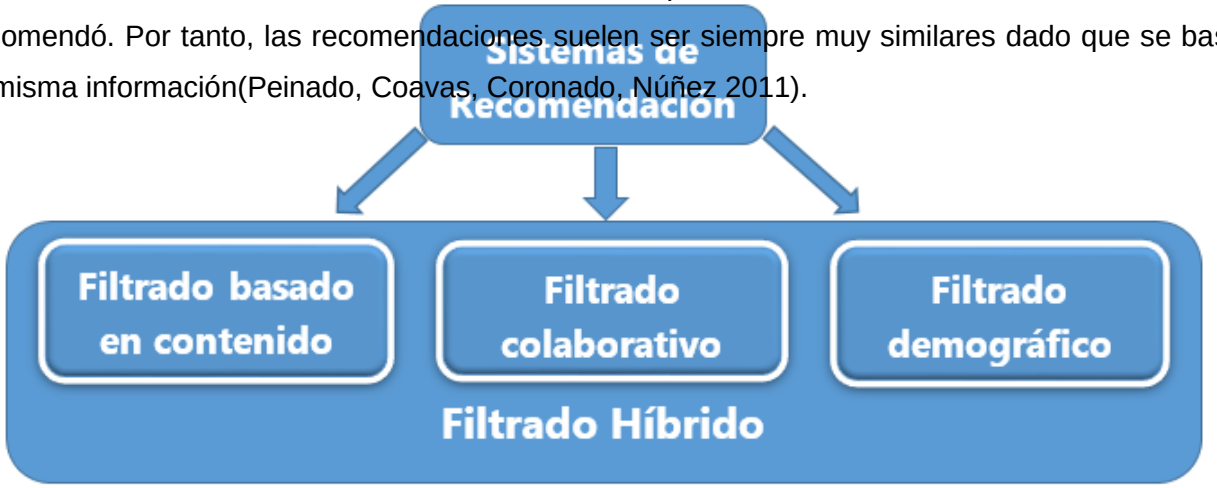



*Figura 2. Clasificación de los Sistemas de recomendación.Elaboración propia*

**Sistemas basados en filtrado colaborativo**

El filtrado colaborativo es otra de las formas de filtrado de información que permite recomendar los elementos que son preferidos por usuarios similares o predecir la utilidad de ciertos elementos para un usuario particular. Las preferencias de los usuarios utilizadas en el sistemason representadas por una evaluación numérica, las cuales pueden obtenerse a partir de diferentes acciones que realice el usuario(Sinbad 2012).

**Sistemas basados en datos demográficos**

Los sistemas basados en datos demográficos son similares a los colaborativos, solo que en el cálculo de los usuarios más cercanos usan información demográfica, es decir, elementos basados en el perfil demográfico de los usuarios como la edad, el sexo, etc. Por ejemplo, se les recomienda información a los usuarios de determinados sitios web basados en su idioma o país. Dichas sugerencias pueden ser personalizadas de acuerdo a la edad del usuario(Sinbad 2012).

Los beneficios que posee este sistema frente a los sistemas basados en filtrado colaborativo es que no requieren un largo historial de valoraciones para hacer recomendaciones a un usuario. Además, los sistemas basados en datos demográficos clasifican a los usuarios en grupos y hace recomendaciones de acuerdo con el grupo(Sinbad 2012).

**Sistemas híbridos**

Los sistemas híbridos utilizan las ventajas de los sistemas anteriores, además de aminorar las desventajas de los mismos con el objetivo de sobrellevar los inconvenientes de estos sistemas para lograr una mayor eficiencia en las recomendaciones y hacer llegar al usuario las mejores de ellas.

Como los Híbridos están basados en los tres tipos de Sistemasde recomendación ya mencionados, se hace una mezcla entre los funcionamientos de estos, por lo cual se crea el perfil de cada usuario a partir de sus predilecciones anteriores y después se comparan estos perfiles para encontrar las similitudes entre usuarios.

Es decir, los resultados se obtienen a partir de las preferencias del usuario y a su vez de las de los usuarios con preferencias parecidas a las de él(Burke 2002).

Para la implementación de la solución y tras analizar las características de cada tipo de recomendador se pretende realizar un SR híbrido con características principales de los Sistemas de Filtrado Colaborativo. Para mayor entendimiento de la estructura de un SR, en el próximo subepígrafe se detallan sus características.

### 1.1.3 Estructura de un Sistema de Recomendación

El usuario solicita la recomendación y el sistema, a partir de la información que tiene almacenada, realiza el proceso de recomendación y brinda los resultado, ver en la **Figura 3**. Es por esto que la estructura de estos sistemas está dada por:

1. Las entradas y salidas del proceso de generación de la recomendación: La entrada es la información que el sistema ingresa del usuario y las salidas son las recomendaciones finales(Peña, Riffo 2008).

2. El método usado para generar las recomendaciones (Proceso de Recomendación): Es la manera en que el proceso de recomendación se va a desenvolver. Existen tres métodos generales que pueden ser utilizados ya sea uno o varios de ellos en un mismo SR(Peña, Riffo 2008).

    I. Recuperación pura o recomendación nula: Es, de forma general, un sistema de búsqueda que permite al usuario examinar ítems en una Base de Datos.
    II. Recomendaciones seleccionadas manualmente por expertos: Las recomendaciones son seleccionadas por expertos en el tema a partir de sus propias preferencias, hacen comentarios para una mayor información y son puestas en una lista disponible para todos los usuarios del sistema.
    III. Resúmenes estadísticos calculados en función de las opiniones del conjunto de usuarios: Se hacen cálculos concretos sobre los usuarios que prefieren a un determinado ítem y a partir de esto es que se hacen las recomendaciones. Estos métodos son muy generales y simples por lo que no se consideran totalmente métodos para generar recomendaciones. Existen otras

posibilidades más específicas, las cuales se evidencian en los dos grandes grupos de Sistemas de Recomendación: los basados en contenido y los de filtrado colaborativo.

El sistema utilizará la recomendación de los resúmenes estadísticos, pero utilizando el filtrado basado en contenido fundamentalmente, aunque se trata de un SR híbrido. En este se empleará inteligencia artificial para realizar el filtrado y realizar las recomendaciones.

3. El grado de personalización:

Según el grado de personalización los Sistemas de Recomendación también se clasifican en(Peña, Riffo 2008):

I. No personalizados: Brindan las mismas recomendaciones a todos los usuarios.
II. Personalización efímera: Las recomendaciones son dadas tomando en cuenta el comportamiento y acciones del usuario en su sesión actual de navegación.
III. Personalización persistente: Estos sistemas están basados en el perfil de los usuarios, por lo que hacen uso de métodos de filtrado colaborativo, filtrado basado en contenidos o correlaciones entre ítems. Brindan recomendaciones con mayor grado de personalización.

La herramienta a desarrollar utilizará la no personalización dado que, aunque las recomendaciones utilizan filtrado híbrido con características del basado en contenido, no están basados en perfil de usuarios ni tomará en cuenta comportamiento y acciones del usuario ya que a cada usuario se le mostrarán las mismas recomendaciones.


*Figura 3. Estructura de un Sistema de Recomendación. Elaboración propia*

Con la información obtenida para realizar el proceso de recomendación, se hace necesario utilizar técnicas de inteligencia artificial para descubrir patrones a partir de los datos, con el fin de utilizar este conocimiento en la implementación de la herramienta. En el próximo epígrafe se describe la utilización de la Minería de Datos para este objetivo.

## 1.2 Utilización del Descubrimiento de Conocimiento y la Minería de Datos

### 1.2.1 Descubrimiento de conocimiento y Minería de Datos

Hoy en día, la cantidad de información almacenada en las bases de datos excede nuestra habilidad para reducir y analizarlos sin el uso de técnicas de análisis automatizadas. En el tema de

descubrimiento de conocimiento se encuentra el proceso completo de extracción de información en bases de datos o KDD (siglas en inglés de Knowledge Database Discovery), que se encarga además de la preparación de los datos y de la interpretación de los resultados obtenidos partiendo de los datos de una base de datos(Valentín, Rodríguez, Piñero, Martínez 2012).Es definido como “el proceso no trivial de identificación en los datos de patrones válidos, nuevos, potencialmente útiles, y finalmente comprensibles”(García, Molina 2006). Las tareas comunes en KDD son la inducción de reglas, los problemas de clasificación y agrupamiento, el reconocimiento de patrones, el modelado predictivo, la detección de dependencias, entre otras(García, Molina 2006).

Este proceso consta de una secuencia iterativa de etapas o fases que se relacionan y describen a continuación y se encuentran ilustradas en la **Figura 4**:

Integración y recopilación: esta fase tiene como objetivo la preparación de las fuentes de datos iniciales y la selección de las mismas.

Selección, limpieza y transformación: en ella se aplican técnicas como limpieza de datos, la integración y transformación de los mismos, la reducción de ellos y la selección de atributos.

Minería de Datos: es donde se perfeccionan constantemente las técnicas y algoritmos que se encargan de extraer y representar el conocimiento de forma adecuada para la toma de decisiones. Se combinan técnicas potenciando las ventajas de cada una y atenuando sus debilidades.

Interpretación y evaluación: aquí se procede al análisis de los resultados descubiertos. Incluye a su vez la resolución de posibles inconsistencias con otros conocimientos anteriores a la investigación(Vercellis 2009).

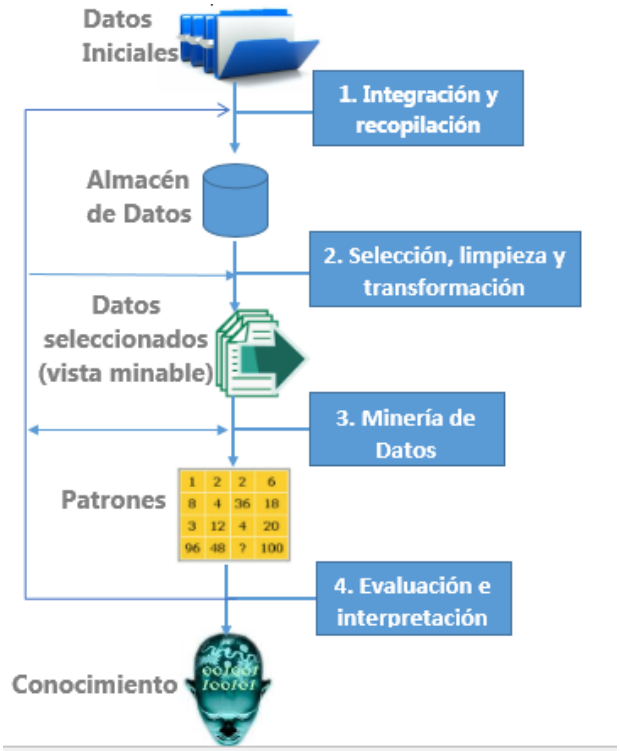


*Figura 4. Etapas para el proceso de extracción del conocimiento. Elaboración propia*

Entre las fases de KDD mostradas anteriormente la fase de Minería de datos es en la que se realiza el proceso de descubrimiento del conocimiento por lo que seguidamente se explica con más detalle esta etapa.

La Minería de Datos (MD) es el análisis, extracción de información y búsqueda de patrones de comportamiento que permanecen ocultos entre grandes cantidades de información. Esta técnica se utiliza para evitar que los investigadores tengan que explorar de forma manual toda la información que está almacenada en las bases de datos(Montes, Gelbukh, López 2005)(Sánchez 2001).

Actualmente la MD puede ser útil en la gestión de capital humano, por ejemplo, en la identificación de las características de sus empleados de mayor rendimiento. La información obtenida puede ayudar a la contratación del personal y la toma de decisiones, centrándose en los esfuerzos de sus empleados y los resultados obtenidos por estos. La MD abarca todo un conjunto de técnicas que abordan la solución a problemas de predicción, clasificación y segmentación. Las funciones de estas técnicas son: explorar datos, darles sentido, convertir un volumen de datos que poco o nada aportan a la descripción en información para interpretar un fenómeno para adoptar decisiones de acuerdo con las necesidades(Valentín, Rodríguez, Piñero, Martínez 2012).

**Técnicas de Minería de Datos**

Las técnicas de la MD (una etapa dentro del proceso completo de KDD) provienen de la Inteligencia Artificial (IA) y de la estadística, dichas técnicas no son más que algoritmos, más o menos sofisticados que se aplican sobre un conjunto de datos para obtener resultados. En la fase de MD, se decide cuál es la tarea a realizar (clasificar, agrupar, etc.) y se elige la técnica descriptiva o predictiva que se va a utilizar(García, Molina 2006).Véase **Figura 5**.

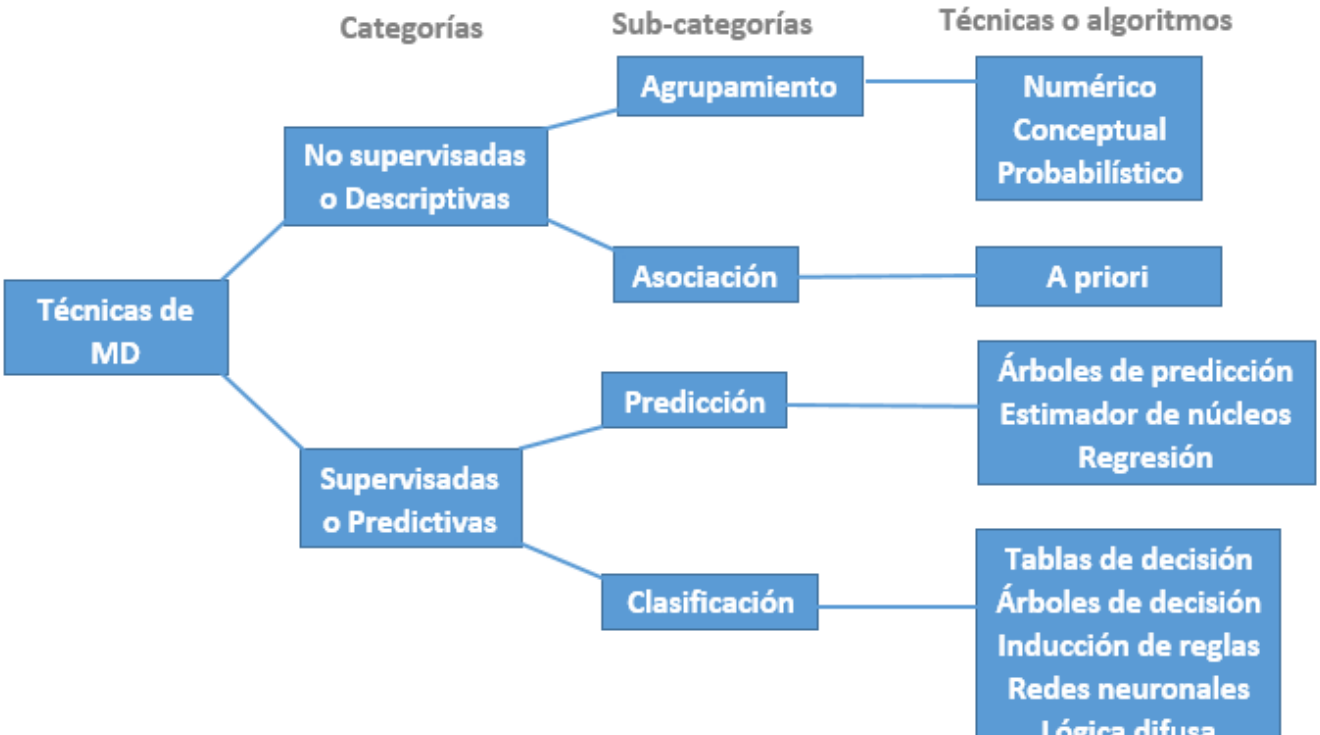


*Figura 5. Técnicas de la Minería de Datos. Elaboración Propia*

Una técnica constituye el enfoque conceptual para extraer la información de los datos y es implementada por varios algoritmos. Cada algoritmo representa, en la práctica, la manera de desarrollar

una determinada técnica paso a paso, de forma que es preciso un entendimiento de alto nivel de los algoritmos para saber cuál es la técnica más apropiada para cada problema. Igualmente es preciso entender los parámetros y las características de los algoritmos para preparar los datos a analizar(Lozano 2013).

### 1.2.3 Clasificación como técnica de Minería de Datos

La clasificación es el proceso de dividir un conjunto de datos en grupos mutuamente excluyentes, de tal forma que cada miembro de un grupo esté lo más cerca posible de otros y grupos diferentes estén lo más lejos posible de otros, donde la distancia se mide con respecto a las variables especificadas, que se quieren predecir. Esta técnica analiza un conjunto de datos cuya clasificación de clase se conoce y construye un modelo para cada clase que suele representarse con un árbol de decisión o reglas de clasificación que muestran las características de los datos(García, Molina 2006).

**Técnicas de clasificación**

1. Tabla de decisión

   Estos algoritmos consisten en seleccionar subconjuntos de atributos y calcular su precisión para predecir o clasificar los ejemplos. Una vez seleccionado el mejor de los subconjuntos, la tabla de decisión estará formada por los atributos seleccionados (más la clase), en la que se insertarán todos los ejemplos de entrenamiento únicamente con el subconjunto de atributos elegido(Martineaux 2008).
2. Árboles de decisión

   La técnica de clasificación árbol de decisión es un modelo de predicción utilizado en el ámbito de la IA. Son muy similares a los sistemas de predicción basados en reglas, que sirven para representar ycategorizar una serie de condiciones que suceden de forma sucesiva, para la resolución de un problema. El aprendizaje de árboles de decisión suele ser más robusto frente al ruido y conceptualmente sencillo, pudiéndose interpretar esencialmente como una serie de reglas compactadas para su representación en forma de árbol(García, Molina 2006). Una de las ventajas

de los árboles de decisión es que representan las relaciones mediante reglas que son fáciles de interpretar permitiendo representar los resultados en lenguajes naturales(Marante 2008).

3. Inducción de reglas

   Estas técnicas permiten la generación y entendimiento de árboles de decisión, o reglas y patrones a partir de los datos de entrada. La información de entrada será un conjunto de casos donde se ha asociado una clasificación o evaluación a un conjunto de variables o atributos. Con esa información estas técnicas obtienen el árbol de decisión o conjunto de reglas que soportan la evaluación o clasificación(García, Molina 2006).

A partir de un análisis de las técnicas de clasificación de acuerdo a sus características antes abordadas, por el tipo de problema y los datos analizados, se decide utilizar los árboles de decisión ya que sirven de ayuda para la elección entre varios cursos de acción, representando la dependencia lógica entre la decisión a tomar y los atributos considerados. Los resultados a través de la representación del conocimiento en árboles de decisión son fáciles de interpretar para un posterior análisis, ya que esta representación es intuitiva y generalmente de fácil entendimiento para las personas. La relación que tienen los árboles con las reglas de clasificación es que se pueden elaborar reglas fácilmente interpretables que definen las características que más diferencian a las distintas clases establecidas inicialmente.

### 1.2.4 Algoritmos de clasificación de árboles de decisión

Para comprender el ámbito de los árboles de decisión a partir de lo antes descripto se analizaron los algoritmos: ID3, C4.5, CHAID, CART y J4.8. Algoritmos que de forma automática bajo el enfoque de aprendizaje supervisado identifican patrones en datos para crear un modelo, en este caso un árbol. Losárboles de decisión predicen el valor de la variable llamada clase, en función de las variables de entradas diferentes y así ayudan en la toma de decisiones.

**ID3**

Es un algoritmo simple pero potente, cuya misión es la elaboración de un árbol de decisión como un método para aproximar una función objetivo de valores discretos. El procedimiento para generar un árbol de decisión consiste en seleccionar un atributo como raíz del árbol y crear una rama con cada uno de los posibles valores de dicho atributo. El algoritmo es resistente al ruido en los datos eligiendo el mejor atributo a través del concepto de entropía, cuanto menor sea el valor de la entropía, menor será la incertidumbre y más útil será el atributo para la clasificación(Quinlan 1993).

**C4.5**

Es uno de los algoritmos más utilizados en el ámbito de los árboles de clasificación de acuerdo a que en cada nodo del árbol, elige el atributo que más eficazmente lo divide(Lozano 2013). Este algoritmo es una extensión del algoritmo ID3 mejorado, ya que trabajan con datos numéricos y simbólicos y

manipulan los casos con valores de atributo que faltan. El C4.5 construye árboles de clasificación de un conjunto de entrenamiento de la misma manera que el algoritmo ID3, usando el concepto de entropía. Su criterio es el de ganancia de información (diferencia de entropía) que dependiendo de los resultados tomará la elección de un atributo para dividir los datos. El atributo con la mayor ganancia será el que se elija para tomar la decisión, luego seguirá recursivamente en los siguientes subnodos(Quinlan 1993).

**J4.8**

J4.8 es la implementación libre del C4.5 y está integrada en la plataforma WEKA. Es uno de los algoritmos de MD más utilizados en multitud de aplicaciones(García, Molina 2006). El parámetro más importante a tener en cuenta es el factor de confianza para la poda que influye en el tamaño y capacidad de predicción del árbol construido. Para cada operación de poda, define la probabilidad de error que permite a la hipótesis de que el empeoramiento debido a esta operación es significativo. A probabilidad menor, se exigirá que la diferencia en los errores de predicción antes y después de podar sea más significativa para no podar. El valor por defecto del factor de confianza es de 25%, según baje este valor se permiten más operaciones de poda(Quinlan 1993).

Teniendo en cuenta las características principales de los algoritmos, se realizó un estudio más profundopara intentar diferenciarlos y compararlos teniendo en cuenta aspectos tales como: la variable predictora, el tipo y criterio de división, método de poda y código de implementación. Véase **Tabla 1**.

Analizando ventajas y deficiencias de cada uno respecto al tipo y criterio de división, casos perdidos y disponibilidad del código, se destaca como el más factible para la presente investigación el algoritmo J4.8. Este permite generar árboles de decisión, facilita la obtención de reglas para la toma de decisiones y además permite el tratamiento de casos con información incompleta o perdida. Además, se aplica en este estudio, trata de enfocarse explícitamente hacia los atributos relevantes e ignora los irrelevantes, es relativamente rápido y muestra fiabilidad en los resultados.

| ALGORITMO | VARIABLES PREDICTORAS | TIPO DE DIVISIÓN | CRITERIO DE DIVISIÓN | DATOS AUSEN-TES | MÉTODO DE PODA | IMPLEMENTA-CIÓN |
|---|---|---|---|---|---|---|
| CART (1984) | Continuas/Discretas | Binaria | Impureza | Sí | Post- | Libre Comercial |
| ID3 (1979) | Discretas | N-aria | Entropía | No | No | Comercial |
| C4.5 (1993) | Continuas/Discretas | N-aria | Entropía | Sí | Pre-/Post- | Libre |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | | Comercial |
| J4.8 | Continuas/Discretas | Binaria/ N-aria | Entropía | Sí | Pre-/Post- | Libre (WEKA) |
| C5.0 | Continuas/Discretas | Binaria/ N-aria | Entropía | Sí | Pre-/Post- | Comercial |
| CHAID (1975) | Discretas | Binaria/ N-aria | x^2 | Sí | Post- | Comercial |

***Tabla 1. Características de los principales algoritmos de árboles de decisión. Datos Extraídos de****(Lozano 2013)*

### 1.2.5 Herramientas empleadas en el proceso de Minería de Datos

Para comprender el ámbito de la solución propuesta se estudiaron varios softwares comerciales y académicos que permitieran realizar tareas de MD. Dentro de las herramientas analizadas se encuentran: KNIME, RapidMiner, WEKA y SAS Enterprise Miner, de las cuales se brinda una caracterización.

**RapidMiner**

Es una herramienta de código abierto implementada en Java por la Universidad de Dortmund, basada en el aprendizaje automático para el descubrimiento del conocimiento y la MD. La aplicación de RapidMinercubre un amplio rango dentro de la MD y es utilizado tanto en investigación como en tareas de día a día por diferentes empresas. Además de ser una herramienta flexible para aprender y explorar la MD, la interfaz gráfica de usuario tiene como objetivo simplificar el uso para las tareas complejas de estaárea. Es un software de tipo Open-Source con licencia GNU 16, trabaja bajo las plataformas Windows y Linux, y usa el lenguaje de scripting XML para describir los operadores y su configuración(RapidMiner 2014).

**WEKA**

WEKA (siglas en inglés de Waikato Environment for Knowledge Analysis) es una plataforma de software libre desarrollado en Java la cual se centra en el aprendizaje automático y en la MD. Proporciona a los usuarios la capacidad de los diversos componentes del software para comunicarse entre sí. El software contiene la capacidad de realizar más de 100 tipos de métodos de MD, incluyendo los métodos de clustering o generación de grupos de datos, clasificación, regresiones, visualización, selección de propiedades y análisis estadísticos. Incluye una interfaz gráfica de usuario para acceder y configurar las diferentes herramientas integradas. Uno de los puntos más fuertes a favor que tiene WEKA es su licencia, ya que es GNU por lo que cualquiera puede modificar o acceder el código del mismo. Uno de los puntos débiles del software es su apoyo a la visualización, ya que, aunque proporciona la visualización de los datos, resultados y procesos, es algo limitada. El uso de WEKA a nivel de usuario no es muy complicado, pero se requiere un conocimiento básico de aprendizaje automático y MD. Además, es independiente de la arquitectura, ya que funciona en cualquier plataforma sobre la que haya una máquina virtual Java disponible(García 2013).

*Tabla 2. Análisis comparativo de sistemas de minería de datos.*

| SISTEMAS | LICENCIA LIBRE | CONOCIMIENTOS SOBRE MD | MULTIPLATAFORMA | MODELO DE CLASIFICACIÓN | J4.8 | CÓDIGO ABIERTO |
|---|---|---|---|---|---|---|
| **KNIME** | Sí | Sí | Sí | Sí | Sí | Sí |
| **RapidMiner** | Sí | Sí | Sí | Sí | Sí | Sí |
| **SAS** | No | Sí | - | Sí | No | No |
| **WEKA** | Sí | Sí | Sí | Sí | Sí | Sí |

Después de haberrealizado una comparación en la

| SISTEMAS | LICENCIA LIBRE | CONOCIMIENTOS SOBRE MD | MULTIPLATAFORMA | MODELO DE CLASIFICACIÓN | J4.8 | CÓDIGO ABIERTO |
|---|---|---|---|---|---|---|
| **KNIME** | Sí | Sí | Sí | Sí | Sí | Sí |
| **RapidMiner** | Sí | Sí | Sí | Sí | Sí | Sí |
| **SAS** | No | Sí | - | Sí | No | No |
| **WEKA** | Sí | Sí | Sí | Sí | Sí | Sí |

, se puede concluir que las herramientasRapidMiner, Wekay KNIME cumplen con las características que se necesitan para el desarrollo de la aplicación.Perode estas solo se utilizará la herramienta WEKA por adecuarse al problema y fácil de usar. WEKA está constituido por una serie de paquetes de código abierto con diferentes técnicas de preprocesado, clasificación, agrupamiento, asociación, y visualización, así como facilidades para su aplicación y análisis de prestaciones cuando son aplicadas a los datos de entrada seleccionados.

### 1.2.6 Sistemas Expertos Basados en Casos

Los SBC, son una delas tecnologías actuales para construir Sistemas Basados en el Conocimiento. En ellos,los nuevos problemas se resuelven considerando la solución dada a problemas similares resueltos en el pasado. La arquitectura básica de un SBC consiste enuna base de casos, un procedimiento para buscar casos similares y un procedimiento de adaptación para ajustar las soluciones de los problemas similares a los requerimientos del nuevo problema(Gutiérrez, Bello, Tellería 2002).Estos representan y almacenan el conocimiento a través de casos y utilizan el Razonamiento Basado en Casos (RBC) como método de solución de problemas para resolver nuevas situaciones. Este modelo de razonamiento permite resolver problemas, entender situaciones y

aprender, utilizando mecanismos de memorización, problemas superpuestos y criterios de optimalidad(Cortez, Navarro, Pariona 2010).

### 1.2.7 Sistemas Expertos Basados en Reglas

Un Sistema Basado en Reglas (SBR) es un sistema basado en el conocimiento,en el cual se realiza una representación simbólica declarativa de un dominio mediante reglas de producción o reglas condicionales. En la **Figura6**se muestra la estructura básica y la estructura detallada de un SBR(Gil 2015).

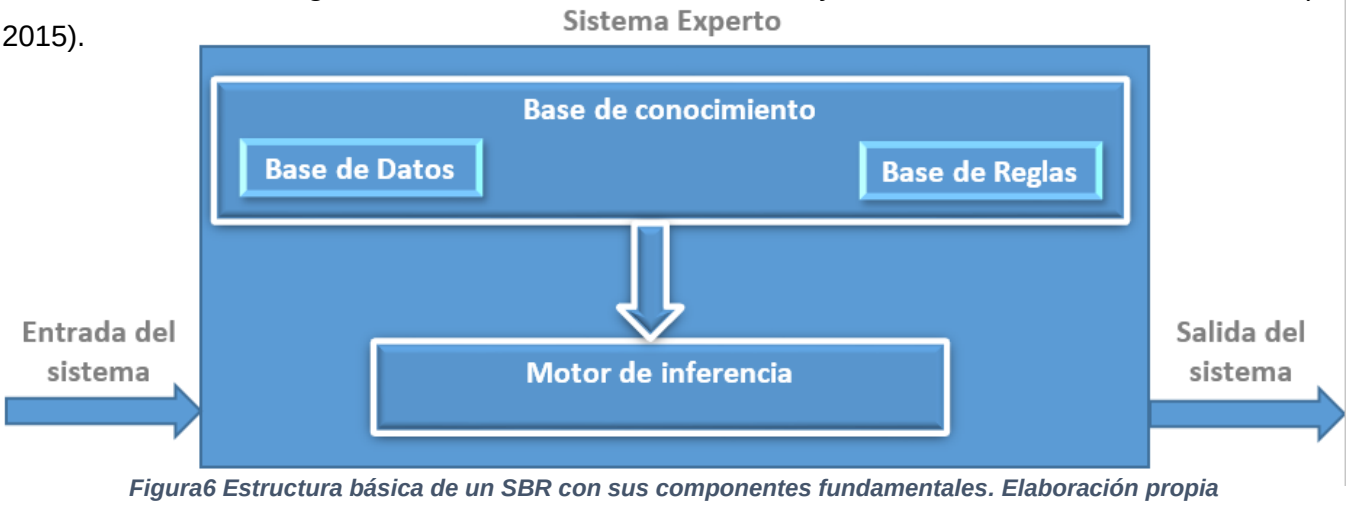



*Figura6 Estructura básica de un SBR con sus componentes fundamentales. Elaboración propia*

**Componentes fundamentales de un SBR**

1. Base de conocimiento:

   Conformada por la base de datos y la base de reglas, que permiten la representación del conocimiento sobre el dominio de aplicación del sistema. En la base de hechos se representa el conocimiento en las variables de entrada y salida del sistema, que forman parte de las reglas semánticas almacenadas en la base de reglas(César 2012).

2. Base de hechos

   Representa el conocimiento del estado del sistema en un cierto instante. Usualmente se representa en una base de datos, y su información está directamente enlazada con la base de conocimientos. Un hecho puede provocar la conformación de una regla(César 2012).

3. Motor de inferencia:

   Por cada entrada al sistema, el motor de inferencia utiliza el conocimiento almacenado en forma de hechos y reglas para generar una salida. Para la obtención de conclusiones, se utilizan diferentes mecanismos y estrategias de inferencia y control. Estas pueden ser simples o compuestas, estas últimas permiten estructurar el conocimiento en diferentes niveles y en este caso se da un encadenamiento de reglas que produce finalmente una conclusión(César 2012).

4. Interfaz de usuario:

Permite la comunicación entre el usuario y el Sistema Experto. El usuario puede consultar con el sistema a través de menús y gráficos, y este le responde con resultados. También es interesante mostrar la forma en que extraen las conclusiones a partir de los hechos(César 2012).

5. Módulo de adquisición del conocimiento

A través de este componente, se construye el sistema o se actualiza el conocimiento de la base de conocimientos en general. Adicionalmente, por medio de este módulo se pueden realizar actividades relacionadas con la configuración del sistema, específicamente del motor de inferencia, de acuerdo con las necesidades del usuario(César 2012).

**Mecanismo de inferencia en SBR**

Encadenamiento de reglas: esconsiderada una de las estrategias más usadas en la obtención de conclusiones compuestas que son las que resultan de la aplicación de más de una regla(César 2012).Este tipo de inferencia parte de la observación de hechos en las variables de entrada para, mediante el encadenamiento de reglas, alcanzar un hecho de salida deseado. Atendiendo al sentido del encadenamiento, se diferencian dos tipos de inferencia: hacia delante y hacia atrás(César 2012).

1. Encadenamiento hacia delante

Este tipo de inferencia parte de la observación de hechos en las variables de entrada mediante el encadenamiento de reglas(Font 2008).

2. Encadenamiento hacia atrás:

El encadenamiento hacia atrás no tiene como objetivo inferir el valor de una variable, sino la demostración de una hipótesis(Font 2008).

## 1.3 Metodología de desarrollo adoptada

Las metodologías son utilizadas para solucionar problemas existentes en la producción de software. Mediante la utilización de las mismas se realizan procedimientos, técnicas, se genera documentación y se utilizan herramientas con el objetivo de lograr un mejor producto software. Esta no es más que la definición de un conjunto de actividades que guían los esfuerzos de las personas implicadas en el proyecto a modo de plantilla que explica los pasos necesarios para la finalización del mismo(Carrillo, Pérez, Rodríguez 2008).

Mediante estudio de diversas metodologías de desarrollo de software (RUP, MSF, XP, SCRUM) resulta necesario utilizar una metodología ágil, dado que la solución se implementará en un proyecto pequeño y cuyo desarrollo está más orientado a un desarrollo rápido y no a los pasos y artefactos de metodologías más robustas como RUP. Sin embargo, a la hora de establecer una metodología para el desarrollo del software, se escogió Scrum dadas las características que posee la misma, principalmente por ser unmodo de desarrollo de carácter adaptable, orientado a las personas antes que a los procesos y en el que se emplea un desarrollo iterativo y escalable. Otro punto positivo de SCRUM es que para mayores especificidades puede combinarse con otras metodologías según los

artefactos que se necesiten. Evita la burocracia y la generación documental excesiva. El cliente final se convierte en uno más del equipo y se es totalmente transparente en cuanto a la información en el desarrollo de la misma para que a medida que se van cumplimentando los requisitos el cliente vaya quedando satisfecho con el resultado y sugiera los cambios que pudiera tener el producto entregado hasta el momento.

### 1.3.1 Scrum

Scrum es un modelo de referencia que define un conjunto de prácticas y roles, y que puede tomarse como punto de partida para definir el proceso de desarrollo que se ejecutará durante un proyecto(SCRUM 2011).

Durante cada *sprint*, un periodo entre una y cuatro semanas (la magnitud es definida por el equipo y debe ser lo más corta posible), el equipo crea un incremento de software potencialmente entregable (utilizable)(SCRUM 2011). Véase **Figura 7**.

Un principio clave de Scrum es reconocer que durante un proyecto los clientes pueden cambiar de idea sobre lo que quieren y necesitan, y los desafíos impredecibles no pueden ser fácilmente enfrentados de una forma predictiva y planificada. Scrum adopta una aproximación pragmática, aceptando que el problema no puede ser completamente entendido o definido, y centrándose en maximizar la capacidad del equipo de entregar rápidamente y responder a requisitos emergentes(SCRUM 2011).Una de las mayores ventajas de Scrum es que es muy fácil de aprender, y requiere muy poco esfuerzo para comenzarse a utilizar(SCRUM 2011).

**Beneficios de Scrum**

1. **Flexibilidad a cambios.** Gran capacidad de reacción ante los cambiantes requerimientos generados por las necesidades del cliente o la evolución del mercado. El marco de trabajo está diseñado para adecuarse a las nuevas exigencias que implican proyectos complejos.

2. **Reducción del Time to Market(Tiempo de Entrega al mercado).** El cliente puede empezar a utilizar las características más importantes del proyecto antes de que esté completamente terminado.

3. **Mayor calidad del software.** El trabajo metódico y la necesidad de obtener una versión de trabajo funcional después de cada iteración, ayuda a la obtención de un software de alta calidad.

4. **Mayor productividad.** Se logra, entre otras razones, debido a la eliminación de la burocracia y la motivación del equipo proporcionado por el hecho de que pueden estructurarse de manera autónoma.

5. **Predicciones de tiempos.** A través de este marco de trabajo se conoce la velocidad media del equipo por sprint, con lo que es posible estimar de manera fácil cuando se podrá hacer uso de una determinada funcionalidad que todavía está en el Backlog.

6. **Reducción de riesgos** El hecho de llevar a cabo las funcionalidades de mayor valor en primer lugar y de saber la velocidad a la que el equipo avanza en el proyecto, permite despejar riesgos efectivamente de manera anticipada(SCRUM 2011).

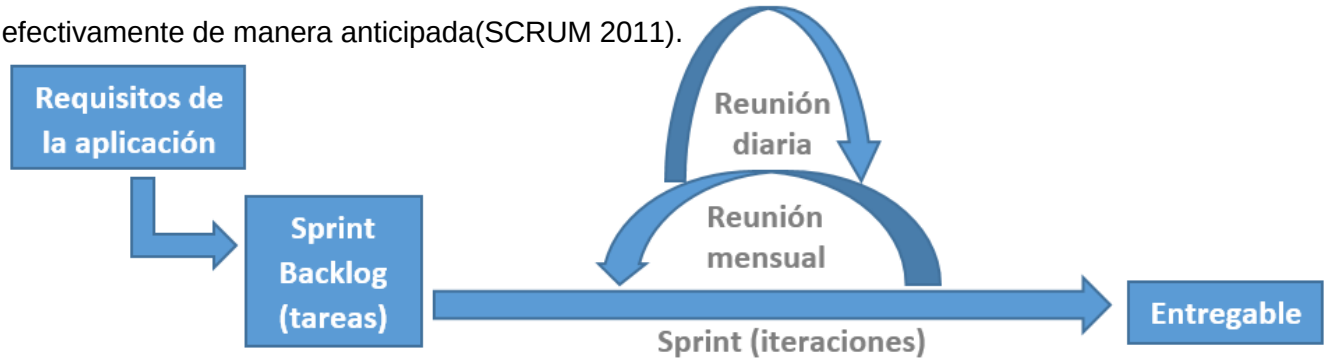



*Figura 7. Marco de desarrollo de SCRUM. Elaboración propia*

## 1.4 Herramientas y tecnologías

Para el desarrollo de la aplicación se hace necesario el uso de herramientas y tecnologías. Se realiza un estudio para valorar cuáles serían las adecuadas teniendo en cuenta la naturaleza del problema que se mueve dentro del contexto académico y científico.

### 1.4.1 Lenguaje de modelado

Se estudiaron dos lenguajes, UML 2.0 y SysML. Finalmente se decidió no utilizar el segundo ya que está destinado a proporcionar técnicas de modelado a una gran variedad de sistemas sin tener en cuenta la dirección de proyectos de ingeniería de sistemas como aspectos de la planificación, entre otras de vital importancia en el desarrollo del software.

En cambio, UML es de fácil comprensión, resulta menos engorroso realizar cambios una vez comenzado a desarrollarse el mismo y con el resulta más sencillo encontrar dificultades y dependencia en los sistemas.

**UML**

Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema de software. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocios y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables(UML 2011).

UML implementa un lenguaje de modelado común para todos los desarrollos por lo que crea una documentación común que cualquier desarrollador con conocimientos de UML será capaz de entender, independientemente del lenguaje utilizado en el desarrollo(Larman 1999).

### 1.4.2 Lenguaje de programación

**Python 2.7**

Python es un lenguaje de programación interpretado cuya filosofía hace hincapié en una sintaxis muy limpia y que favorezca un código legible. Se trata de un lenguaje de programación multiparadigma, ya que soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional. Es un lenguaje multiplataforma(Python Software 2012).

Usa tipado dinámico y conteo de referencias para la administración de memoria. Una característicaimportante de Python es la resolución dinámica de nombres; es decir, lo que enlaza un método y un nombre de variable durante la ejecución del programa (también llamado enlace dinámico de métodos)
(Python Software 2012). Este lenguaje se utiliza para la solución propuesta en conjunto con el framework Django ya que el mismo permite la implementación de algunas funcionalidades del sistema.

Después de revisar diversas alternativas, como el uso de Java, PHP y Python en la implementación del sistema, se decide utilizar este último por adecuarse al problema académico de la investigación.

**HTML5**

HTML5 es el último de marcado de hipertexto para los sitios web desarrollado por World Wide Web Consortium (W3C por sus siglas en inglés). Se utilizará el lenguaje HTML 5 para la creación de las interfaces del sistema, tomando como provecho las peculiaridades del soporte CSS3 y el manejo de formularios en el navegador.

**CSS3**

Las hojas de estilo en cascada (Cascading Style Sheets o CSS por sus siglas en inglés) son las que ofrecen la posibilidad de definir las reglas y estilos de representación en diferentes dispositivos, ya sean pantallas de equipos de escritorio, portátiles, móviles u otros dispositivos capaces de mostrar contenidos web. Permiten definir de manera eficiente la representación de nuestras páginas y es uno de los conocimientos fundamentales que todo diseñador web debe manejar a la perfección para realizar su trabajo (De Luca 2010). CSS3 se utilizará para dar estética siguiendo las pautas de diseño de la interfazweb del sistema.

**JavaScript**

JavaScript es un lenguaje de programación que se utiliza principalmente para crear páginas web

dinámicas, incorporando efectos como texto que aparece y desaparece, animaciones, acciones que se activan al pulsar botones y ventanas con mensajes de aviso al usuario (Pérez 2007). Se utilizará este lenguaje en el desarrollo del sistema para incrementar la interacción en la aplicación con respecto a la visualización de los efectos asociados a los eventos que se manejarán de manera más rápida y dinámica.

### 1.4.3 Sistema Gestor de Base de Datos

Aunque fueron valoradas otras alternativas como Oracle y MySQL, el gestor de base de datos escogido finalmente fue PostgreSQL. La alternativa Oracle fue desechada pese a ser uno de los sistemas de base de datos más complejos dado que este es un producto muy costoso, por lo que es utilizado en grandes empresas y multinacionales de desarrollo de software dado el alto costo que supondría su soporte técnico y mantenimiento. PostgreSQLes la alternativa idóneapor ser un SGBD de código abierto y orientado a objetos, cuenta con un entorno amigable y fácil de utilizar, es un sistema que soporta la gran mayoría de las transacciones SQL y trabaja con gran cantidad de lenguajes como Java, Python, PHP, entre otros.

**PostgreSQL 9.0**

Para el buen funcionamiento de la aplicación se necesita contar con un Sistema Gestor de Base de Datos que posibilite la obtención de la información de la forma más sencilla y rápida posible. En la actualidad PostgreSQL es considerado uno de los gestores de BD de código abierto más avanzado del mundo, debido a que soporta casi toda la sintaxis SQL. Tomando como referencia las características antes expuestas y por la condición de ser una herramienta libre y multiplataforma se decide utilizar en el desarrollo del sistema.

**PgAdmin 1.14.1**

Es una aplicación gráfica para administrar PostgreSQLcon licencia de código abierto. Es capaz de gestionar versiones a partir de la PostgreSQL 7.3 y superiores ejecutándose en cualquier plataforma. Este está diseñado para responder a las necesidades de todos los usuarios, desde escribir consultas SQL simples hasta desarrollar bases de datos complejas. La interfaz gráfica soporta todas las características de PostgreSQL y facilita enormemente la administración. Una característica interesante es que, cada vez que se realiza alguna modificación en un objeto, escribe lassentencias SQL correspondientes, lo que hace que, además de una herramienta muy útil, sea a la vez didáctica. También incorpora funcionalidades para diseñar consultas(Moquillaza, Vega, Guerra 2010).

### 1.4.4 Herramienta CASE (Computer Aided Software Engineering)

Tras estudiar y valorar un trío de las herramientas CASE más importantes (dígase Visual Paradigm, Rational Rose y MagicDraw) se resolvió que Rational Rose no es aplicable en este caso dado que

obliga a que el usuario desarrolle sobre sistema operativo Windows por lo que no es multiplataforma, MagicDrawpor su parte solo permite el desarrollo de código para un reducido grupo de lenguajes como C#, C++ y Java,que es el lenguaje en el que está desarrollada la aplicación. Sin embargo, y aunque en un principio no se descartó usar MagicDraw, la tercera opción, Visual Paradigm es la escogida finalmente por ser sobre la que más información se puede encontrar y en la que más experiencia tiene el desarrollador, dado que es objeto de estudio por parte de la universidad, es más usable que otras herramientas, y resulta más cómodo ya que al utilizar las funcionalidades de este se puede generar el prototipo de base de datos a utilizarse en una aplicación y así ahorrar tiempo. Por tanto, la opción aceptada sería Visual Paradigm en su versión 8.0 ya queademás permite crear diagramas durante todo el ciclo de vida de desarrollo de un proyecto de manera fácil, entendible y rápida y está disponible en varias plataformas y versiones.

**Visual Paradigm 8.0**

Visual Paradigm for UML(VP) es una herramienta CASE que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, implementación y pruebas. Ayuda a una rápida construcción de aplicaciones de calidad, mejores y a un menor costo. Permite construir diagramas de diversos tipos, código inverso, generar código desde diagramas y generar documentación(Larman 2003).

Entre sus características fundamentales se tienen:

1. Multiplataforma: soportada en plataforma Java para Sistemas Operativos Windows, Linux, Mac OS.
2. Interoperabilidad: intercambia diagramas y modelos con otras herramientas. Soporta la importación y exportación a formatos XMI y XML y archivos Excel. Permite importar proyectos de Rational Rose y la integración con Microsoft Office Vision.
3. Ingeniería de código: permite la generación de código e ingeniería inversa para los lenguajes: Java, C, C++, PHP, XML, Python, C#, VB .Net, Flash, ActionScript, Delphi y Perl.
4. Integración con entornos de desarrollo: apoyo al ciclo de vida completo de desarrollo de software en IDEs como: Eclipse, Microsoft VisualStudio, NetBeans, Sun ONE, Oracle JDeveloper, Jbuilder y otros.
5. Brinda la posibilidad de documentar todo el trabajo sin necesidad de utilizar herramientas externas.

### 1.4.5 Arquitectura de software

Una arquitectura de software define la estructura del sistema, la cual adquiere gran importancia debido a que las representaciones de arquitecturas de software permiten la comunicación entre todas las partes interesadas en el desarrollo de un sistema de cómputo. Dicha arquitectura constituye un modelo

comprensible de cómo está estructurado el sistema y cómo trabajan juntos sus componentes (Cervantes 2010).

**Arquitectura Cliente-Servidor**

En esta arquitectura la computadora de cada uno de los usuarios, llamada cliente, produce una demanda de información a cualquiera de las computadoras que proporcionan información, conocidas como servidores estos últimos responden a la demanda del cliente que la produjo. Mediante esta arquitectura el usuario puede acceder a la información sin tener en cuenta su ubicación física y donde pueda estar alojada la misma (Pressman 2007).

### 1.4.6 Patrón arquitectónico

Los patrones arquitectónicos, o patrones de arquitectura, son patrones de diseño de software que ofrecen soluciones a problemas de arquitectura de software en ingeniería 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. Un patrón arquitectónico expresa un esquema de organización estructural esencial para un sistema, que consta de subsistemas, sus responsabilidades e interrelaciones (Jacobson 2009). Para el desarrollo del sistema, se utiliza como patrón arquitectónico a seguir Modelo Plantilla Vista (MTV según sus siglas en inglés).

**Modelo Plantilla Vista**

Dentro de las tecnologías a utilizar en el desarrollo del sistema se encuentra el framework Django el cual emplea una modificación del patrón arquitectónico Modelo-Vista-Controlador (MVC), la cual es llamada Modelo-Plantilla-Vista (MTV). Véase **Figura 8**.

1. **Modelo:** Se refiere a la capa de acceso a datos. Esta capa contiene todo lo referido a los datos: cómo acceder a ellos, cómo validarlos, qué comportamiento tienen y las relaciones entre ellos.
2. **Plantilla:** Se refiere a la capa de presentación. Esta capa contiene las decisiones relacionadas con la presentación: cómo debería mostrarse el contenido en una página web u otro tipo de documento.
3. **Vista:** Se refiere a la capa de lógica del negocio. Esta capa contiene el acceso al modelo y delega en las plantillas apropiadas.

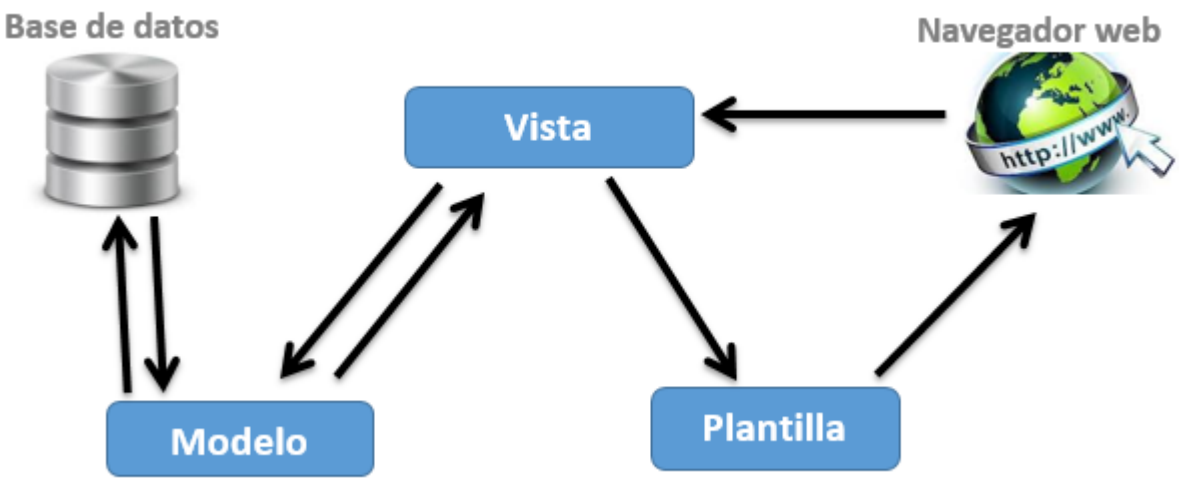


*Figura 8. Patrón Modelo-Plantilla-Vista. Elaboración propia.*

### 1.4.6 Entorno de Desarrollo Integrado (IDE)

Un IDE es un programa informático compuesto por un conjunto de herramientas de programación. Puede dedicarse en exclusiva a un solo lenguaje de programación o bien poder utilizarse para varios. Puede denominarse como un entorno de programación que ha sido tratadocomo un programa aplicación. Esto significa que consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica(Tendenzias 2015). En el caso de la solución por utilizar como lenguaje de programación Python en el IDE escogido para el desarrollo es PyCharm Professional.

**PyCharm Professional 3.4.1**

PyCharm presenta un editor de códigos inteligente, que entiende los detalles específicos de Python y ofrece extraordinarios mejoradores de la productividad: formateo automático de código, finalización de código, refactorizaciones, importación automática, navegación de código con un solo clic. Respaldadas por avanzadas rutinas de análisis de código, estas características hacen de PyCharm una poderosa herramienta tanto en las manos de los desarrolladores Python profesionales como en las de quienes recién comienzan a usar la tecnología. Se escogió ya que el lenguaje principal es Python y este como se menciona anteriormente entiende los detalles específicos de Python, completa código, tiene importación automática y otras características que ayudan mucho porque los programadores están familiarizados con la herramienta (JetBrains 2010).

### 1.4.7 Framework de desarrollo

Con el objetivo de usar un framework de desarrollo acorde a la arquitectura empleada y al tipo de herramienta a desarrollar se escoge Django dado que el trabajo con este produce un diseño efectivo e interfaz amigable en las herramientas desarrolladas (Holovaty, Kaplan-Moss 2004).

**Django 1.7.11**

Django es un framework web de código abierto escrito en Python que permite construir aplicaciones web de forma rápida y con menos código (Holovaty, Kaplan-Moss 2004). El mismo emplea una modificación del patrón arquitectónico Modelo-Vista-Controlador (MVC), la cual es llamada Modelo-Plantilla-Vista
(MTV).

La versión 1.7.11 aparte de presentar las propiedades de las versiones anteriores, incluye nuevas características que permiten mejorar el desarrollo en aplicaciones mediante su uso, aplica migraciones al esquema de base de datos, permitiendo hacer modificaciones sin perder la información almacenada. Además, este framework es compatible con diversos gestores de datos.

**JQuery 1.9.1**

jQuery es un framework para el lenguaje Javascript que implementa una serie de clases (de programación orientada a objetos) que permite programar sin tener en cuenta el navegador con el que está visitando el usuario. Ofrece una infraestructura con la que se tendrá mucha mayor facilidad para la creación de aplicaciones complejas del lado del cliente(Álvarez 2016). Se usará para la implementación del sistema debido a que proporciona el manejo rápido de propiedades y CSS que se emplearán en la solución propuesta.

**Bootstrap 3.0**

Bootstrap es un excelente framework que permite crear interfaces de usuario limpias y totalmente adaptables a todo tipo de dispositivos y pantallas basado en HTML, CSS y JavaScript (Fontela 2016). Esta herramienta ayudará a agilizar la creación de la interfaz de nuestra página web y permitirá crear un diseño limpio, intuitivo, usable y de poco peso, por lo que la carga de nuestra web será muy rápida(Robledano 2015).

## Conclusiones Parciales

El estudio de los sistemas de recomendación permitió entender el funcionamiento, las características y los antecedentes de los mismos. Se determinaron las técnicas de recomendación a utilizar llegando a la conclusión de que el sistema a desarrollar para la asignación de equipos de investigación de tesis de grado, es un SR híbrido enfocado en la técnica basada en el filtro colaborativo, ya que el mismo permite mitigar las limitaciones que poseen estas técnicas y generar recomendaciones con mejor calidad.

Con el estudio de las técnicas de inteligencia artificialadecuadas al problema se decidió realizar un sistema basado en reglas que tenga como motor de inferencia la evaluación de reglas aportadas a partir del descubrimiento de conocimiento.

La selección de las herramientas, tecnologías y metodología a utilizar permitió establecer el entorno de desarrollo en el que se implementará la propuesta de solución.

# Capítulo 2. Propuesta de solución

En el capítulo presente será abordado el modelo conceptual delSR, así como el funcionamiento de los requisitos funcionales y no funcionales tenidos en cuenta para la realización de la solución propuesta, se incluirán algunos patrones de diseño utilizados para lograr buenas prácticas de diseño y programación, será explicada la estructura de la base de datos desarrollada, así como las principales opciones de configuración del SRde grupos de investigación.

## 2.1 Modelo conceptual

*La propuesta de solución de la presente investigación comprende el desempeño de procesos relacionados con la asignación relacionados con la asignación de tesis a estudiantes de pregrado y la vinculación de estudiantes y profesores a las líneas profesores a las líneas temáticas sobre las que se desarrolla la investigación en la Universidad de las Ciencias Informáticas Ciencias Informáticas (UCI). Además de valorar a través del resultado académico de los estudiantes y el criterio científico de el criterio científico de los profesionales para propiciar que con los datos extraídos en cada caso pueda conformarse un mejor conformarse un mejor equipo de investigación para desarrollar el trabajo de diploma. Para una mejor comprensión se muestra comprensión se muestra un modelo conceptual sobre el negocio a desarrollar, y posteriormente se mostrará una descripción mostrará una descripción de los conceptos mostrados previamente en la*

**Figura 9**.

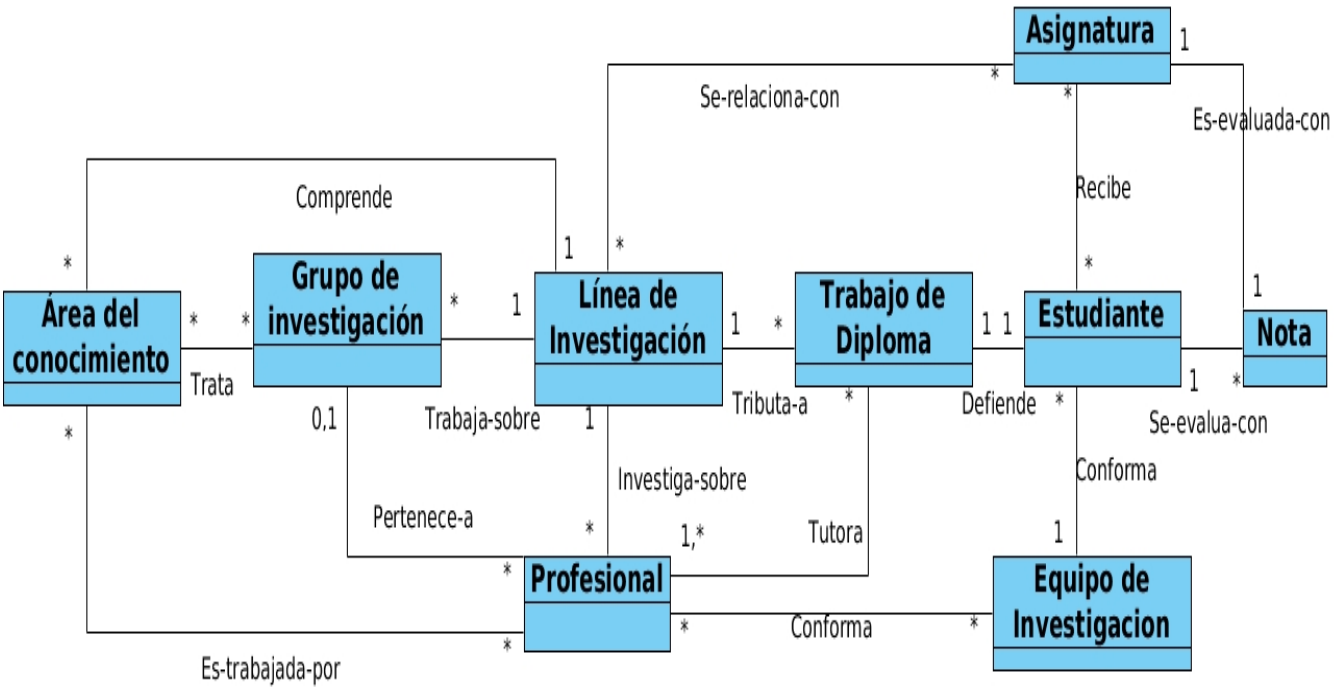


*Figura 9. Modelo conceptual*

**Estudiante**

Según el Diccionario de la Real Academia Española (RAE) en su segunda acepción, es la persona que cursa estudios en un establecimiento de enseñanza(RAE 2014). En el caso de la investigación es aquella que lo hace en el curso regular diurno de la UCI. Para el manejo de la información se tomó en cuenta como estado docente solo a los estudiantes matriculadosy se decidió no utilizar datos de estudiantes de reingreso, baja, repitencia o provenientes de otro Centro de Enseñanza Superior, para asegurar mayor homogeneidad en los datos. En el caso del atributo de situación escolar se consideraron a los estudiantes promovidos y promovidos arrastrando una asignatura del año previo de la carrera.

**Asignatura**

Según del Diccionario de la RAE es cada una de las materias que se enseñan en un centro docente o forman parte de un plan de estudio(RAE,2014). Para la investigación fueron utilizadas las materias que se imparten en el plan de estudio del curso regular diurno. En el estudio y extracción de datos se utilizaron asignaturas de tercero, cuarto y quinto años de la carrera, preferentemente aquellas que tuvieran vínculos con las líneas de investigación de la universidad. Las asignaturas se clasifican en lectivas, optativas y electivas, de las cuales las últimas no fueron utilizadas en el desarrollo de la soluciónde bido a que presentan una baja relación con las líneas de investigación definidas en la UCI y no ser asignaturas clave en el estudio de la carrera.

**Nota**

En el Diccionario de la RAE y asociado a su sinónimo “calificación” en su segunda acepción, es la puntuación obtenida en un examen o cualquier tipo de prueba(RAE 2014). En este caso se asocia solo a la nota final que es la forma de evaluación de la asimilación del contenido de una asignatura por parte de un estudiante. La calidad de la evaluación estaráfundamentadaen un conjunto de objetivos definidos por las asignaturas,que el estudiante debe vencer para demostrar el conocimiento adquirido en estas. Esta evaluación varía entre 2 y 5 puntos. Siendo dos puntos la nota más deficiente y que demuestra que el estudiante no venció el mínimo de objetivos para esa asignatura y cinco puntos la nota que refleja que el estudiante pudo vencer con calidad todos los objetivos definidos en esta.

**Profesional**

Según el Diccionario de la RAE en su tercera acepción: “dicho de una persona: que practica habitualmente una actividad de la cual vive”(RAE 2014). En los marcos de la investigación es aquella persona graduada con categoría científica o no (MSc. ó DrC.) que trabaja en la universidad y está vinculada a las esferas de Docencia, Investigación y/oProducción. En este caso cualquier profesional con título universitario y con conocimientos en el campo sobre el cual va a asesorar en la investigación puede ser tutor en el trabajo de diploma de un estudiante de quinto año.

**Trabajo de Diploma**

Un Trabajo de Diploma o tesis de investigaciónes un informe que concierne a un problema o conjunto de problemas en un área definida de la ciencia y explica lo que se sabe de él previamente, lo que se haría para resolverlo, lo que sus resultados significan, y dónde o cómo se pueden proponer progresos, más allá del campo delimitado por el trabajo(UAP 2007).En el contexto de la investigación es la forma de evaluación final de cada estudiante en la UCI, mediante la cual puede optar por el título de Ingeniero en Ciencias Informáticas. Cada tema del trabajo de diploma está relacionado a su vez a una línea de investigación definida por la universidad.

**Equipo de Investigación**

Para el contexto de la presente investigación, se denomina a la asociación formada por uno o más tutores junto a uno o dos estudiantes (en dependencia de si el alcance y complejidad de un trabajo de diploma así lo requieren) para investigar y desarrollar el trabajo de diploma sobre el cual los estudiantes van a defender para obtener el título de Ingeniero en Ciencias Informáticas. La conformación de este equipo de investigación es la base principal del proceso de asignación de tesis en la UCI e influye indirectamente en el éxito del desarrollo y posterior defensa del trabajo de diploma. El tutor funge como asesor en todo el proceso de desarrollo y muchas veces puede ser el mismo cliente al cual entregarle la aplicación final o ser parte del proyecto que lo integra a través de varios módulos.

**Línea de Investigación**

Es un enfoque intradisciplinario que permite englobar procesos, prácticas y perspectivas de análisis y definición disciplinaria con énfasis en los aportes de experimentalidad simbólica y creatividad expansiva e inclusiva del campo de la comunicación en sus más amplias acepciones y potencialidades. Sus alcances y desarrollos materiales de las prácticas y saberes involucrados son transversales a los Proyectos(UNLP 2011).Referido a la investigación, las líneas son definidas por la universidad y se encuentran ampliamente vinculadas al estudio de las ciencias informáticas. A estas se encuentran asociadas varias áreas del conocimiento o temáticas que a su vez son estudiadas por grupos de investigación. Cada profesional está asociado a una línea de investigación lacualse vincula a aquella sobre la que trabajó su tesis de pregrado y en la que investiga a través de los trabajos de diploma que tutora y los artículos científicos que publica. Los estudiantes también pueden estar asociados a una línea de investigación a través de un grupo de investigación o bien como parte de su trabajo en asignaturas como Proyecto de Investigación y Desarrollo (PID) (ACM 2014).

**Área del conocimiento**

Es un elemento estructural de alto nivel usado para organizar, clasificar y describir los conocimientos de una disciplina determinada (ACM 2014). En el contexto de desarrollo se trabajan como las

temáticas de una línea investigación. Un grupo de investigación se enfoca en una o varias áreas del conocimiento de la línea de investigación sobre la que se desarrolla.

**Grupo de Investigación**

Equipo creado con carácter oficial en la universidad para realizar investigaciones en un determinado campo de las ciencias informáticas y así aportar los conocimientos adquiridos en el área donde se investiga para impulsar el desarrollo y mayor aprovechamiento de las tecnologías de la información y las comunicaciones (TIC) así como de la información vinculada a estas y con el que pueden beneficiar tanto al centro como a la sociedad en general. Está compuesto por un grupo de profesionales que además asesoran a estudiantes, introduciéndoles en la esfera de la investigación para que además de aportar al grupo con los conocimientos adquiridos, también les sirva como base para especializarse en determinado campo de la informática y mejorar las habilidades investigativas del estudiante.

En la **Tabla 3** se refleja, la relación existente entre las líneas de investigación, las áreas del conocimiento y los grupos de investigación existentes actualmente en la universidad(ACM 2014).

*Tabla 3. Relación entre líneas de investigación, áreas del conocimiento y grupos de investigación*

| NO. | LÍNEAS DE INVESTIGACIÓN | ÁREAS DEL CONOCIMIENTO | GRUPOS DE INVESTIGACIÓN |
|---|---|---|---|
| **1** | Visión por computadora (señales, realidad virtual) | -Realidad virtual y simulaciones (RVS)<br>-Procesamiento de imágenes y señales (PDIS) | -VIVIRG<br>-PDSG<br>-Informática Médica |
| **2** | Computación científica (computación de alto rendimiento, matemática física, bioinformática) | -Bioinformática (BIO)<br>-Matemática y Física aplicada a la computación (MF)<br>-Programación de avanzada (CAR) | -Bioinformática<br>-Matemática<br>-GCAR |
| **3** | Inteligencia artificial y Reconocimiento de patrones | -Procesamiento de imágenes y señales (PDIS)<br>-Inteligencia Artificial (IA)<br>-Gestión de Proyectos (GPI) | -Minería de proceso<br>-Web semántica<br>-GRISOFT<br>-SIPII<br>-AIRI<br>-Ingeniería y gestión de proyectos<br>-Inteligencia Artificial<br>-Informática Médica |
| **4** | Redes y seguridad | -Procesamiento de imágenes y señales (PDIS)<br>-Redes de telecomunicaciones (RT)<br>-Seguridad informática (SI) | -Redes |
| **5** | Ingeniería de sistemas y gestión | -Inteligencia organizacional (ITO)<br>-Tecnologías de base de datos (BD)<br>-Ingeniería de Software (ISW)<br>-Gestión de Proyectos (GPI) | -Bases de datos<br>-GRISOFT<br>-Informática Jurídica<br>-Ingeniería y gestión de proyectos |
| **6** | Software libre y Código abierto (tecnologías abiertas) | -Software libre, sistemas de código abierto (SL) | -Software libre |
| **7** | Informática aplicada en la sociedad | -Informática educativa (IED)<br>-Formación del ingeniero informático (FII)<br>-Impacto Social de las TIC (TIC) | -Ingeniería y gestión de proyectos<br>-GIDIPRED<br>-Informática Jurídica<br>-SIPII |

| | | | |
|---|---|---|---|
| | | -Informática Jurídica (IJ)<br>-Ciencias del deporte (CD)<br>-Gestión de Proyectos (GPI)<br>-Informática Médica (IM) | -GRISOFT<br>-GITAE<br>-Impacto en las TICs<br>-Informática Médica |

## 2.2 Requisitos del sistema

Los requisitos del sistema son características o funcionalidades que deberá cumplir el sistema una vez haya concluido el proceso de desarrollo de la solución propuesta. Scrum, la metodología escogida asume como punto de partida un par de documentos conocidos como Product Backlog (cuya traducción sería Cartera del Producto) y Sprint Backlog (Cartera del Sprint) en los cuales se identifican los requisitos funcionales generales del producto y los requisitos que serán tenidos en cuenta para el sprint respectivamente, además de un conjunto de tareas que también son tenidas en cuenta para el primer Sprint o parte entregable del sistema a desarrollar(SCRUM 2011).

### 2.2.1 Requisitos funcionales

Los requisitos funcionales de un sistema describen lo que el sistema debe hacer. Estos requisitos dependen del tipo de software que se desarrolle, de los posibles usuarios y del enfoque general tomado por la organización al redactar el requisito. Los requisitos funcionales del sistema describen con detalle lafunción de este, sus entradas, salidas y excepciones(Sommerville 2005). Para el desarrollo del sistema se identificaron los siguientes requisitos funcionales.

**RF1** Autenticar usuario.

**RF2** Gestionar usuario.

**RF2.1** Adicionar usuario.

**RF2.2** Modificar usuario.

**RF2.3** Eliminar usuario.

**RF3** Asignar rol a usuario.

**RF4** Eliminar rol a usuario.

**RF5** Gestionar estudiante.

**RF5.1** Adicionar estudiante.

**RF5.2** Modificar estudiante.

**RF5.3** Eliminar estudiante.

**RF6** Gestionar profesional.

**RF6.1** Adicionar profesional.

**RF6.2** Modificar profesional.

**RF6.3** Eliminar profesional.

**RF7** Gestionar asignatura.

**RF7.1** Adicionar asignatura.

**RF7.2** Modificar asignatura.

**RF7.3** Eliminar asignatura.

**RF8**Insertar nota.

**RF9**Modificar nota.

**RF10**Gestionar línea de investigación.

**RF10.1** Adicionar línea de investigación.

**RF10.2** Modificar línea de investigación.

**RF10.3** Eliminar línea de investigación.

**RF11** Gestionar área del conocimiento.

**RF11.1** Adicionar área del conocimiento.

**RF11.2** Modificar área del conocimiento.

**RF11.3** Eliminar área del conocimiento.

**RF12** Gestionar grupo de investigación.

**RF12.1** Adicionar grupo de investigación.

**RF12.2** Modificar grupo de investigación.

**RF12.3** Eliminar grupo de investigación.

**RF13** Buscar estudiante.

**RF14**Buscar profesional.

**RF15** Realizar recomendación.

**RF15.1** Mostrar reporte de recomendaciones de estudiantes.

**RF15.2** Mostrar reporte de recomendaciones de profesionales.

### 2.2.2 Requisitos no funcionales

Los requisitos no funcionales son aquellos que no se refieren directamente a las funciones específicas que proporciona el sistema, sino a las propiedades emergentes de este como la fiabilidad, el tiempo de respuesta y la capacidad de almacenamiento. Define las restricciones del sistema como la capacidad de los dispositivos de entrada/salida y las representaciones de datos que se utilizan en las interfaces del sistema(Sommerville 2005).El sistema se realizará bajo los siguientes requisitos no funcionales.

**Requisitos de apariencia o interfaz externa**

**RNF1:**la resolución mínima recomendada es de 1024x768px.

**RNF2:** el sistema mostrará el nombre del producto.

**RNF3:** el sistema mostrará el logo del producto.

**RNF4:** el idioma que se utilizará será el español.

**RNF5:** el texto será de color negro.

**Requisitos de usabilidad**

**RNF6:** el sistema debe poseer una distribución de la información simple y que posibilite a los usuarios llegar al contenido que desea en un corto tiempo, siempre que no fuerce la estructura del sistema.

**RNF7:** debe poseer una interfaz amigabley que muestrede forma clara todas las funcionalidades que brinda la aplicación.

**Requisitos de portabilidad**

**RNF8:** la herramienta debe ser ejecutada sobre los sistemas operativos Linux y Windows, por su característica de ser multiplataforma.

**Requisitos de eficiencia**

**RNF9:** tiempo de respuesta: la mayoría de los procesos que se implementan contransacciones donde se modifica la base de datos deben tener tiempos de respuesta nomayores a los 700 milisegundos. En el caso de la obtención de información a través de las recomendaciones, se busca que el tiempo de respuesta se simplifique al máximo y no exceda los 3 segundos.

**Requisitos de software**

**RNF10:** para que el sistema funcione deberá ser instalada la máquina virtual de java 7, para garantizar la gestión de los datos el SGBD PostgreSQL 9.4.

**RNF11:**sistema operativo Microsoft Windows 7 o superior o cualquier distribución de Linux.

**Requisitos del diseño y de implementación**

**RNF12:** el sistema implementado será una aplicación web.

**RNF13:** el sistema será diseñado siguiendo los principios de programación orientada aobjeto.

**RNF14:** el sistema se implementará usando PyCharm 3.4.1.

**RNF15:** el lenguaje de programación a utilizar será Python.

**Requisitos de funcionalidad**

**RNF16:** el sistema cumplirá con los requisitos funcionales previamente descritos,incluyendo la entrada de datos, procesamiento y obtención de resultados.

**RNF17:** el sistema mostrará los errores en forma de mensajes. Todos los mensajes de errordel sistema deberán incluir una descripción textual del mismo.

## 2.3 Patrones de diseño

Un patrón de diseño es una descripción de clases y objetos comunicándose entre sí adaptada para resolver un problema de diseño general en un contexto particular(Gamma, Helm, Johnson, Vlissides 1995). Son soluciones simples, elegantes, probadas y documentadas, a problemas específicos y comunes del diseño orientado a objetos. Están basados en la recopilación del conocimiento de los expertos en desarrollo de software. Clasifican y describen formas de solucionar problemas que ocurren de forma frecuente en el desarrollo. Elconocimiento de los patrones de diseño es vital para entender el funcionamiento del mismo y por ende lograr los objetivos trazados. Por esta razón uno de los pasos a tener en cuenta es identificar qué patrones pueden ser utilizados. Los patrones aplicados en la confección de la herramienta fueron los siguientes:

### 2.3.1 Patrones para Asignar Responsabilidades (GRASP)

Durante el diseño de la herramienta se emplearon patrones de software para asignación general de responsabilidades o GRASP (siglas en inglés de General Responsibility Assignment Software Patterns), específicamente:

**Experto**

El patrón experto es el principio básico de asignación de responsabilidades. Indica que la responsabilidad de la creación de un objeto o la implementación de un método, debe recaer sobre la clase que conoce toda la información necesaria para crearlo. Se evidencia este patrón en la clase “Profesional”, pues cuenta con la información necesaria para cumplir responsabilidades específicas.

**Creador**

El patrón creador ayuda a identificar quién debe ser el responsable de la creación (o instanciación) de nuevos objetos o clases(Larman 1999). Una de las consecuencias de usar este patrón es la visibilidad entre la clase creada y la clase creador, este patrón se pone de manifiesto en las clases: Estudiante, Profesional y Sistema. A la hora de crear objetos se deben tener en cuenta las características de la clase.

**Controlador**

El patrón controlador es un patrón que sirve como intermediario entre una interfaz y el algoritmo que la implementa, de tal forma que es la que recibe los datos del usuario y la que los envía a las distintas clases según corresponda(Larman 1999). La clase “Sistema” es la responsable de atender los eventos del sistema y, además, encargada de capturar las llamadas que se efectúan en la aplicación en tiempo real.

**Alta Cohesión**

La cohesión es una medida de la fuerza con la que se relacionan las clases y el grado de focalización de las responsabilidades de un elemento. Se evidencia en cada clase del diseño propuesto que realiza una labor única dentro del sistema y colabore con las otras para llevar a cabo una tarea, como son las clases que forman parte de las capas negocio y acceso a datos(Larman 1999).

**Bajo Acoplamiento**

Este patrón GRASP se pone en práctica en las clases que poseen pocas relaciones entre sí, para que, en caso de producirse una modificación en alguna de ellas, se tenga la mínima repercusión en las demás. Esta característica permitió potenciar la reutilización y disminuyó la dependencia entrelas clases(Larman 1999).

### 2.3.2 Patrones GoF (Gang of Four)

En el diseño fueron tomados en cuenta los siguientes patrones de la “banda de los cuatro” o Gang of four (GoF) denominados así por ser creado por los 4 autores del libro Design Patterns: Elements of Reusable Object-Oriented Software, Erich Gamma, Richard Helm, Ralph Johnson y John Vlissides:

**Creacionales**

Fábrica Abstracta (Abstract Factory): el problema a solucionar por este patrón es el de crear diferentes familiasde objetos, como por ejemplo la creación de interfaces gráficas de distintos tipos (ventana,menú, botón)(Larman 1999). Se pone de manifiesto en cada una de las clases que conforman la programación de las interfaces de la herramienta.

Instancia Única (Singleton): garantiza la existencia de una única instancia para una clase y la creación deun mecanismo de acceso global a dicha instancia. Se muestra en todas las clases controladoras que soninstancias únicas.

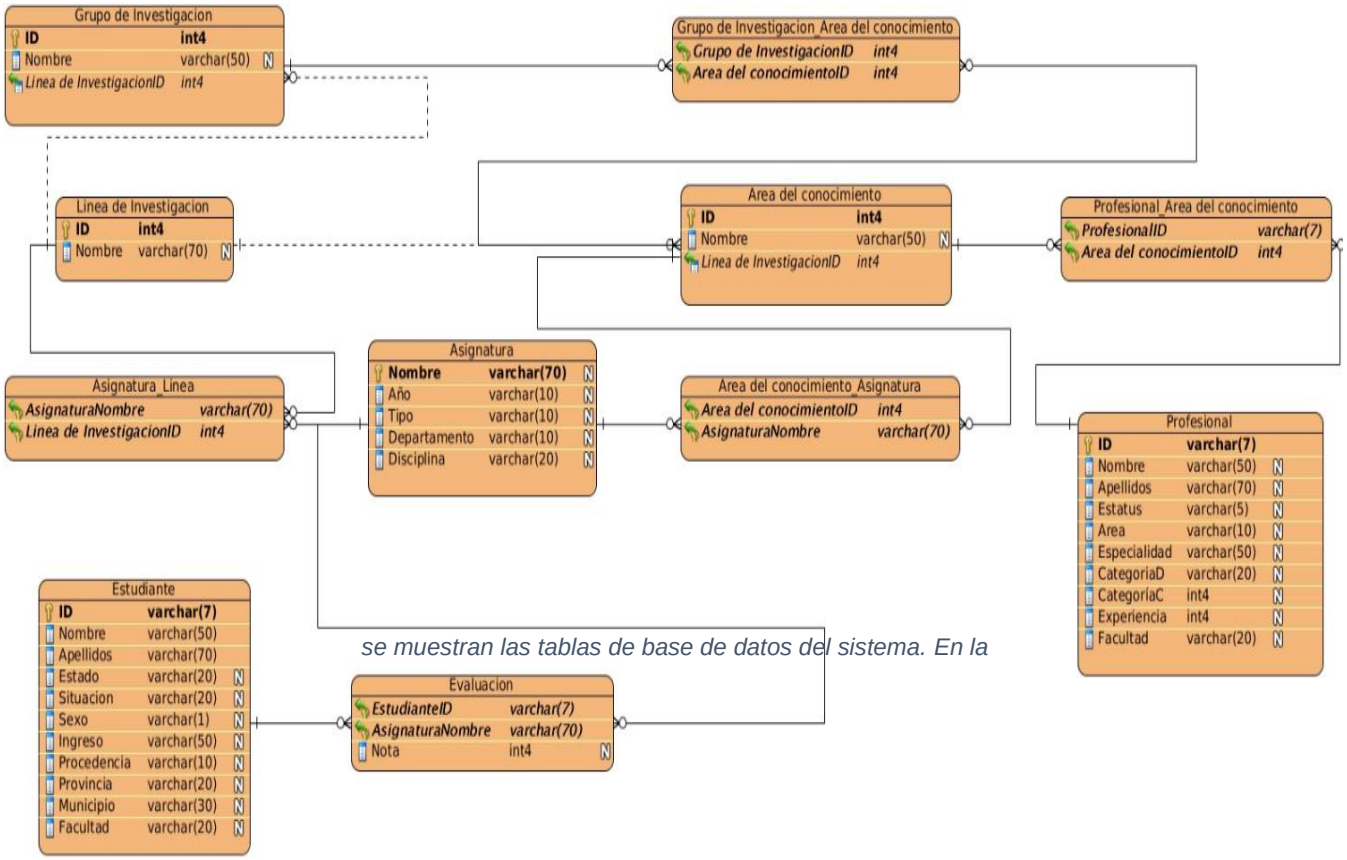
**Comportamiento**

Mediador (Mediator): define un objeto que coordine la comunicación entre objetos de distintas clases, peroque funcionan como un conjunto. Se evidencia en las librerías que funcionan como mediadoras entrelasclases controladoras y las modelos o acceso a datos.

## 2.4 Diseño de la Base de Datos

*La tarea de diseño de la base de datos es una de las tareas más importantes que se realiza en el desarrollo de un sistema desarrollo de un sistema que necesite trabajar con información persistente. En general, el objetivo de diseñar una base de diseñar una base de datos relacional es generar un conjunto de esquemas de relaciones que permitan almacenar la almacenar la información con un mínimo de redundancia, pero que a su vez faciliten la recuperación de información.En la*

*de información.En la*




*se muestran las tablas de base de datos del sistema. En la*

**Figura 11** se observan con más detalle las entidades con sus atributos.

*Figura 10. Diagrama Entidad-Relación de la base de datos*

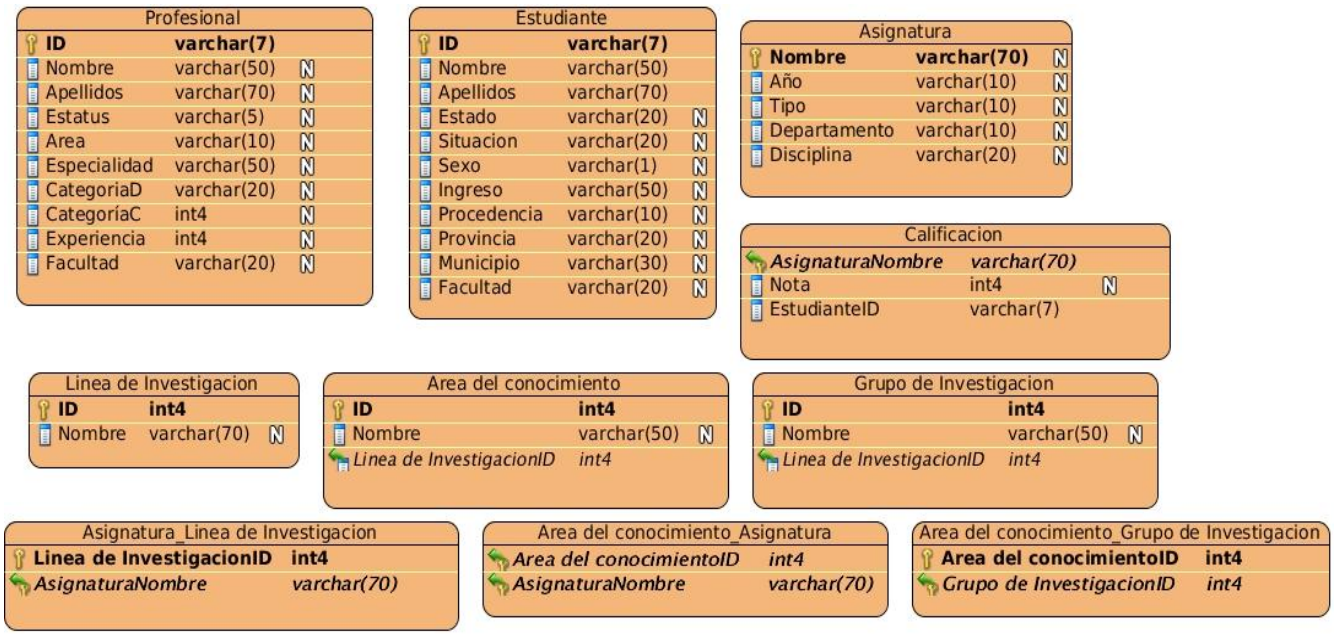



*Figura 11. Entidades de la Base de Datos sin relaciones*

## 2.5 Descripción delasolución

Para el desarrollo de la herramienta se requiere aplicar técnicas que procesen la información para luego realizar la recomendación de las líneas de investigación de los profesionales y estudiantes. Para apoyar a la toma de decisiones de la persona que realiza la asignación en estos procesos es de gran utilidad emplear técnicas y herramientas de inteligencia artificial (IA) tales como los sistemas basados en reglas y sistemas basados en casos. Los pasos definidos son los siguientes:

Previo a la implementación:

Realizar entrenamiento de los datos de estudiantes y profesionales.

Extraer las reglas como resultado de la aplicación del algoritmo J4.8

Después de realizado el entrenamiento:

Implementar el sistema de reglas como base para las recomendaciones.

Establecer una medida para ordenar a profesionales y estudiantes en una escala de mayor a menor criterio de recomendación.

Lo primero que debe tenerse en cuenta es que los procesos de recomendación de estudiantes y profesionales son diferentes. En el caso de los estudiantes se utilizó el método de la entrevista como instrumento para recolectar opiniones sobre la relación de las asignaturas y las áreas del conocimiento identificadas en la universidad, un total de 20 entrevistas con las siguientes características: Los profesionales entrevistados fueron profesores de más de cinco años de experiencia en las asignaturas seleccionadas para el análisis, y especialistas de la producción de CEIGE, CEGEL, CESOL y Especialistas de Seguridad Informática que trabajan las áreas temáticas de las facultades 1 y 3. Se realizó la triangulación de los datos y la triangulación de participantes a partir de los resultados de la entrevista con el objetivo de confrontar sus opiniones para ver las diferencias y semejanzas.

En la realización del análisis de los resultados de la entrevista se pudo constatar que cada entrevistado les daba una prioridad a las asignaturas relacionándola con más del 50 % de las áreas del conocimiento, siempre enfatizando que asignatura era la más importante en la relación.Se confirmó que las asignaturas de idioma, y metodología delainvestigación son importantes para todas las áreas del conocimiento. La línea de investigación de Software Libre tiene relación directa con todas las asignaturas analizadas, por tanto, solo van a estar en ella directamente en el análisis aquellas personas que estén involucrados en la creación de sistema operativos. El resultado de esta entrevista fue establecer la relación existente entre las asignaturas del ciclo profesional y las áreas del conocimiento previamente definidas. Véase ejemplo en**Tabla 5**, en los anexos podrá encontrar la relación completa.

La muestra utilizada para trabajar el sistema fueron los estudiantes de las cohortes 2011-2016 (actual quinto año) y 2012-2017 (estudiantes que actualmente cursan el cuarto año de la carrera) de las facultades 1 y 3. Por lo que se procedió a utilizar a los estudiantes de quinto año en el entrenamiento de datos y según las reglas obtenidas realizar las recomendaciones a los estudiantes de quinto año.

Se realizó un profundo análisis con los estudiantes de quinto año según los resultados obtenidos en las asignaturas a mostrarse en la **Tabla 4**y se evaluó cuan favorable es su desempeño académico frente a las áreas del conocimiento a las que se vinculan las asignaturas escogidas(Uci 2015). Su objetivo fue mostrar si los estudiantes alcanzan mayores o menores resultados en las líneas de investigación en dependencia de sus resultados docentes por lo que en el análisis se valorócon un 0 si las evaluaciones fueron insuficientes (obtuvo notas iguales o menores a tres puntos en la mayoría de las asignaturas) y 1 en caso contrario. Se podrá consultar en los anexos una muestra de esta evaluación efectuada a los estudiantes.

*Tabla 4. Asignaturas utilizadas para conformar los datos de entrenamiento*

| MUESTRA DE ASIGNATURAS PARA LA INVESTIGACIÓN | | | |
|---|---|---|---|
| **NO** | **LECTIVAS** | **NO** | **OPTATIVAS** |
| 1 | Idioma Extranjero III | 20 | Comercio Electrónico |
| 2 | Idioma Extranjero IV | 21 | Informática Forense y Hacking Ético |
| 3 | Ingeniería de Software I | 22 | Optimización de Bases de Datos |
| 4 | Ingeniería de Software II | 23 | Protocolo TCP/IP y sus Aplicaciones |
| 5 | Investigación de Operaciones | 24 | Reconocimiento de Patrones |
| 6 | Metodología de la Investigación Científica | 25 | Almacenes de Datos |
| 7 | Probabilidades y Estadística | 26 | Hardware de Computadoras |
| 8 | Programación IV | 27 | Inteligencia Organizacional |
| 9 | Programación V | 28 | Lenguajes de Programación |
| 10 | Sistemas Operativos | 29 | Minería de Procesos |
| 11 | Sistemas de Bases de Datos II | 30 | Optimización |
| 12 | Subsistemas de Organizaciones | 31 | Redes Neuronales Artificiales |
| 13 | Teleinformática | 32 | Web Semántica |
| 14 | Componente Profesional de Ingeniería y Gestión de Software | 33 | Análisis y Diseño de Algoritmos |
| 15 | Gestión de Software | 34 | Arquitectura Orientada a Servicios |
| 16 | Inteligencia Artificial I | 35 | Configuración de Equipos de Interconexión de Redes |
| 17 | Redes y Seguridad Informática | 36 | Gráficos por Computadora |
| 18 | Simulación | 37 | Factibilidad de Proyectos de Inversión |
| 19 | Inteligencia Artificial II | | |

Una vez realizado este análisis se le asignó a cada estudiante en la línea de investigación donde obtuvo mejores resultados como variable objetivo en el proceso de clasificación.

En el entrenamiento a los datos de los profesionalesse utilizancomo variables predictoras sus datos principales (dígase experiencia, categoría docente, científica, etc.), además de un conjunto de hasta

tres temáticas o áreas del conocimiento en las que ha estado trabajando, a través de su autoría y tutoría en tesis de pre y postgrado, publicaciones científicas y trabajos presentados en eventos de esa índole. Posteriormente, se define en cuál área temática ha investigado más definiéndola como su primera temática y más importante. Con esta información se le asigna a una línea de investigación que será la variable objetivo a emplear en la clasificación.

*Tabla 5. Vinculación entre el contenido de las asignaturas y el área del conocimiento*

| ASIGNATURA/ ÁREAS DEL CONOCIMIENTO | RVS | PDIS | BIO | MF | CAR | IA | GPI | RT | SI | ITO | BD | ISW | SL | IAS |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Ingeniería de Software I | | | | | | | x | | | | x | x | | x |
| Ingeniería de Software II | | | | | | | x | | | | x | x | | x |
| Investigación de Operaciones | | | | | | | x | | | x | | | | |
| Programación IV | | | | x | | | | | | | | | | |
| Programación V | x | | | | | | | | | | | | x | x |
| Inteligencia Artificial I | x | x | x | x | | x | | | | | | | | |
| Redes y Seguridad Informática | | x | | | | | | X | x | | | | | |
| Minería de Procesos | | | | | | x | x | | | x | x | | | |
| Redes Neuronales Artificiales | x | x | x | x | x | x | x | | | x | | | | |

### 2.5.1 Entrenamiento de los datos

Para obtener las reglas con que se recomendará a los estudiantes se utiliza la técnica de clasificación, que es la más frecuente en la práctica.

**Preparación de los datos**

Los datos de entrada a la herramienta, sobre los que operarán las técnicas implementadas, deben estar codificados en un formato específico, denominado Attribute-Relation File Format (extensión "arff"). La herramienta permite cargar los datos en tres soportes: fichero de texto, acceso a una base de datos y acceso a través de internet sobre una dirección URL de un servidor web. En este caso se trabajó con ficheros de texto. Los datos deben estar dispuestos en el fichero de la forma siguiente:

cada instancia en una fila, y con los atributos separados por comas. Un ejemplo de la estructura que sigue un ficheroarff con los datos de los estudiantes es la siguiente:

% comentarios

@relation Estudiantes_Asignaturas

@attribute 'Gestión de Software' {2,3,4,5}

@attribute 'Inteligencia Artificial I' {2,3,4,5}

@attribute 'Redes y Seguridad Informática' {2,3,4,5}

@attribute Simulación {2,3,4,5}

@attribute LI {1,2,3,4,5,6,7}

…

@data

5,4,4,5,5,4,5,5,5,5,4,5,5,5,5,4,5,5,5,?,?,5,?,5,5,5,?,?,?,?,5,5,?,5,?,?,?,3

5,5,3,5,4,4,5,3,5,3,3,3,3,4,4,3,3,5,3,?,5,5,?,?,?,4,?,3,5,?,?,3,?,5,?,?,?,2

…

DATOS(contendrán separadas por coma las notasde las asignaturas como variables predictoras y al final la línea en la que se clasificó como variable objetivo)

Por tanto, los atributos pueden ser principalmente de dos tipos: numéricos de tipo real o entero (indicado con las palabras real o integer tras el nombre del atributo), y simbólicos, en cuyo caso se especifican los valores posibles que puede tomar entre llaves. La relación se llamará Estudiante_Asignaturas, como atributo simbólico tendrá las líneas de investigación y los nombres, y como atributos enteros las notas de las asignaturas. Para mayor especificación puede encontrarse en los anexos un ejemplo completo de este fichero arff empleado en la clasificación de los datos.

**Clasificación en WEKA**

En ocasiones, el problema de clasificación se formula como un refinamiento en el análisis, una vez que se han aplicado algoritmos no supervisados de agrupamiento y asociación para describir relaciones de interés en los datos. Se pretende construir un modelo que permita predecir la categoría de las instancias en función de una serie de atributos de entrada. En el caso de WEKA, la clase es simplemente uno de los atributos simbólicos disponibles, que se convierte en la variable objetivo a predecir, que en este caso serán las líneas de investigación. Por defecto, es el último atributo (última columna) a no ser que se indique otro explícitamente(Gil 2015). La configuración de la clasificación se efectúa con la ventana mostrada en el anexo 1.

**Modos de evaluación del clasificador**

El resultado de aplicar el algoritmo de clasificación se efectúa comparando la clase predicha con la clase real de las instancias. Esta evaluación puede realizarse de diferentes modos, según la selección en el cuadro Test options y en el que se utilizó el modo Cross Validation (usar datos de entrenamiento)(Gil 2015).

Validación cruzada (Cross-validation): Esta opción es la más elaborada y costosa. Se realizan tantas evaluaciones como se indica en el parámetro Folds. Se dividen las instancias en tantas carpetas como indica este parámetro y en cada evaluación se toman las instancias de cada carpeta como datos de test, y el resto como datos de entrenamiento para construir el modelo. Los errores calculados son el promedio de todas las ejecuciones (Gil 2015).

**Clasificador como árbol de decisión: J4.8**

El algoritmo J4.8 de WEKA es una implementación del algoritmo C4.5, uno de los algoritmos de MD que más se ha utilizado para el desarrollo de aplicaciones. De los múltiples parámetros de configuración únicamente se resaltará uno de los más importantes, el factor de confianza para la poda, (confidence level), puesto que influye notoriamente en el tamaño y capacidad de predicción del árbol construido(Gil 2015). Una explicación simplificada de este parámetro de construcción del árbol es la siguiente: para cada operación de poda, define la probabilidad de error máxima para la cual se considerará que el empeoramiento de los resultados luego de la realización de la poda es significativo. Cuanto más baja se haga esa probabilidad, se exigirá que la diferencia en las probabilidades de error antes y después de podar sea más significativa para saber si realizar la poda ya no resulta eficiente. El valor por defecto de este factor es del 25%, y conforme va bajando se permiten más operaciones de poda y por tanto llegar a árboles cada vez más pequeños. Otra forma de variar el tamaño del árbol es a través de un parámetro que especifica el mínimo número de instancias por nodo, si bien es menos elegante puesto que depende del número absoluto de instancias en el conjunto de partida.

Al terminar la clasificación se devolverá un árbol con las reglas que se emplearán para realizar la implementación del proceso de recomendación de estudiantes como se muestra en la **¡Error! No se encuentra el origen de la referencia.**(Gil 2015).

### 2.5.2 Sistema Basado en Reglas para recomendación de estudiantes y profesionales

Una vez obtenidas las reglas del árbol de decisión se procede a la implementación de los sistemas basados en reglas (SBR) para los profesionales y estudiantes. Los SBR,son uno de los modelos de representación del conocimiento más ampliamente utilizados,debido a que resultan muy apropiados en situaciones en las que el conocimiento que se desea representar, surge de forma natural conestructura

de reglas. Las reglas son una afirmación lógica que relaciona dos o más objetos e incluye dos partes, la

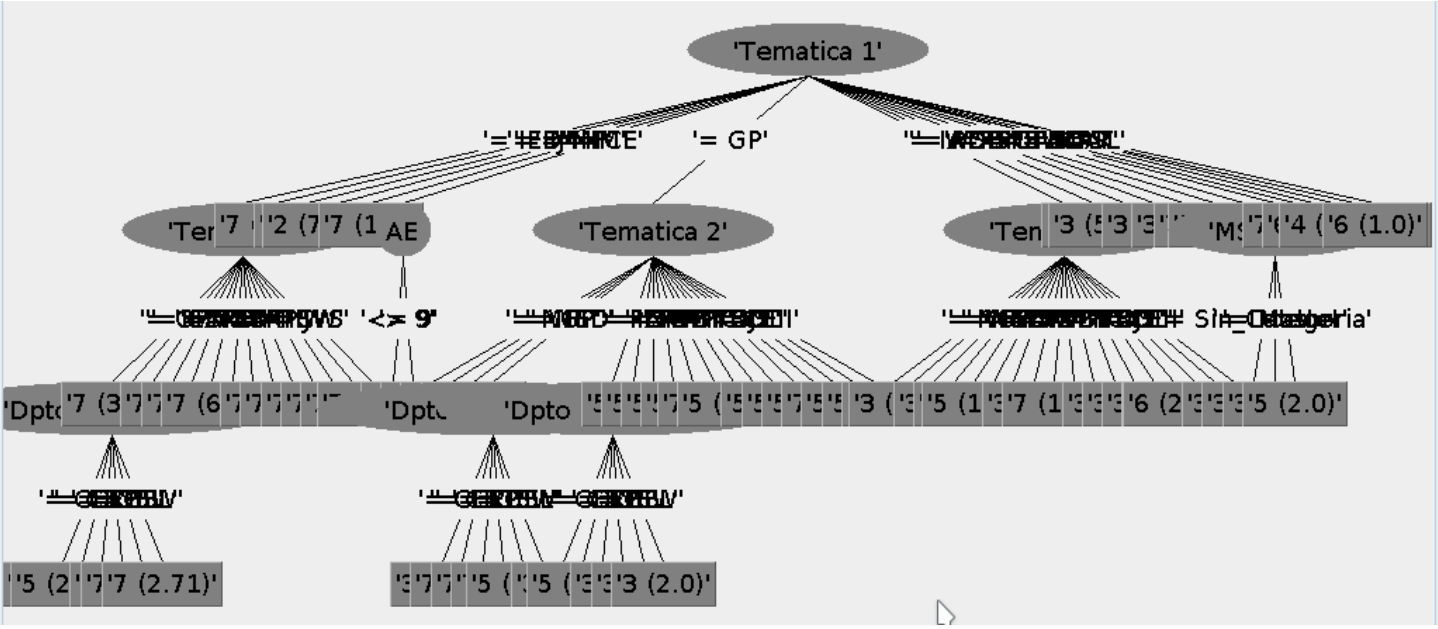



*Figura 12. Ejemplo de árbol de decisión mostrado por el WEKA al realizar la clasificación con J4.8*

premisa (antecedente) y la conclusión (consecuente). Cada una de estas partes consiste en una expresión lógica con una o más afirmaciones objeto-valor conectadas mediante los operadores lógicos y(and), o(or),ono(not)(César i 2012).Las reglas siguen el formato SI antecedenteENTONCES consecuente donde:

Antecedente:conjuncionesde atributos de un mismo dominio.

Consecuente: atributos que pasarán a ser conocidos por el Sistema Experto.

### 2.5.3Criterio de recomendación de estudiantes

Para establecer un ordenamiento de cuales profesionales y estudiantes son los mejores recomendados se estableció un criterio de recomendación. En el caso de los estudiantes se realizó mediante el promedio de las notas alcanzadas hasta el momento al que se le añadió un valor de ponderación entre 0 y 1 definido de la siguiente forma:

Según la línea de investigación a la cual se está clasificando se analizan sus resultados en las asignaturas que pertenecen a las áreas del conocimiento de la línea.

Se le suma 1 por cada asignatura en la que obtiene 4 o 5 puntos

Se le resta 1 por cada asignatura en la que obtiene 3 puntos.

Se le resta 2 puntos por cada asignatura en la que obtiene 2 puntos.

Si el número obtenido es negativo, no se recomienda al estudiante para esa línea

Si el número es positivo se divide entre la cantidad de notas y ese resultado es el valor de ponderación que se le suma a su promedio.

El porciento de recomendación se calcula dividiendo la suma entre el promedio de las notas y la variable de ponderación con el mejor de los casos que sería 6 (un estudiante con promedio de 5 y valor de ponderación igual a uno).

Para el caso de los profesionales el criterio de recomendación amerita la puesta en práctica de un sistema basado en casos.

### 2.5.4Criterio de recomendación de profesionales. Sistema basado en casos

Un Sistema Basado en Casos (SBC), es una de las técnicas de Inteligencia Artificial (IA) más empleadas en la actualidad en los procesos de toma de decisiones, es un tipo de sistema experto que utiliza los casos como forma de representación del conocimiento y emplean el Razonamiento Basado en Casos (RBC) como mecanismo de inferencia(Gutiérrez, Pérez 2012).

A partir del estudio de los SBC se enuncian un conjunto de pasos para el desarrollo del mismo con el objetivo de recomendar los profesionales basados en las líneas de investigación en la cual trabajan.

Secuencia de pasos para el desarrollo del SBC:

1.Definir los rasgos predictores y los rasgos objetivos.

2. Determinar el dominio de definición de cada rasgo.

3. Determinar la prioridad y peso informacional de cada rasgo.

4. Definir la función de comparación de casos.

**Definición de los rasgos predictores y el rasgo objetivo**

Para definir los rasgos predictores, se basóen la caracterización de recursos humanos, donde se evalúan los siguientes rasgos:

-Estatus (ej. Si es plantilla, se encuentra prestando servicios, o trabaja a tiempo parcial.)

-Especialidad (si su especialidad está vinculada directamente o no con las ciencias informáticas)

-Categoría Docente (si es Asistente, Especialista, Instructor, Profesor Auxiliar o Profesor Titular)

-Categoría Científica (si es MSc.,DrC. o no tiene categoría)

-Años de experiencia (Cantidad de años trabajando como profesional)

Rasgo objetivo: Línea de Investigación.

**Dominio de definición de cada rasgo**

*En este paso se procede a determinar el dominio de definición de cada rasgo. Ver*

**Tabla 6**:

*Tabla 6.Dominio de definición de cada rasgo seleccionado.*

| RASGO | TIPO DE VALOR | DOMINIO |
|---|---|---|
| **Estatus** | Cadena | [P,PS,TP] |
| **Especialidad** | Cadena | [sí, no] |
| **Categoría Docente** | Cadena | [A,E,I,PA,PT] |
| **Categoría Científica** | Cadena | [Master, Doctor, Ninguna] |
| **Años de experiencia** | Entero | [0,…,60]* |

Para esta última categoría se definieron rangos siendo estos:

[0-5] años: Experiencia baja

[6-15] años: Experiencia media

[16-60] años: Experiencia alta

Esta evaluación se tuvo en cuenta dado el promedio de años de experiencia calculado entre los profesionales de las facultades 1 y 3.

**Prioridad y Peso informacional de cada rasgo**

Después de tener los rasgos y su dominio de definición se hace importante definir el peso informacional
que posee cada uno de estos rasgos para el problema así como la prioridad que van a tener estos, siendo algunos más significativos que otros.

*Dado que cada uno está confeccionado según el estudio de comportamiento y el criterio de especialistas en recursos especialistas en recursos humanos quedan de la siguiente forma evidenciados en la* ***Tabla 7*** *la prioridad de los rasgos y en la*

**Tabla 8** el peso informacional por cada uno de ellos:

*Tabla 7. Prioridad de cada rasgo seleccionado*

| RASGO | PRIORIDAD (ENTRE 1 Y 5) |
|---|---|
| **Estatus** | 1 |
| **Especialidad** | 2 |
| **Categoría Docente** | 2 |
| **Categoría Científica** | 5 |
| **Años de experiencia** | 4 |

*Tabla 8. Peso informacional de rasgos*

| RASGO ESTATUS | PESO INFORMACIONAL (ENTRE 1 Y 3) |
|---|---|
| **P** | 2 |
| **PS** | 1 |
| **TP** | 1 |
| **RASGO ESPECIALIDAD** | **PESO INFORMACIONAL (ENTRE 1 Y 2)** |
| **Sí** | 2 |
| **No** | 1 |
| **RASGO CATEGORÍA DOCENTE** | **PESO INFORMACIONAL (ENTRE 1 Y 5)** |
| **Asistente** | 2 |
| **Especialista** | 3 |
| **Instructor** | 2 |
| **Profesor Auxiliar** | 4 |
| **Profesor Titular** | 5 |
| **RASGO CATEGORÍA CIENTÍFICA** | **PESO INFORMACIONAL (ENTRE 1 Y 3)** |
| **Máster** | 3 |
| **Doctor** | 2 |
| **Ninguna** | 1 |
| **RASGO EXPERIENCIA** | **PESO INFORMACIONAL (ENTRE 1 Y 3)** |
| **Baja** | 1 |
| **Media** | 2 |

**Selección de la función de comparación de casos**

La función para comparar los casos retornará un número que será la evaluación del profesional que reflejará la posición que ocupará respecto a los demás profesionales vinculados a su misma línea de investigación y que demostrará cuál es el porciento de recomendación que se le otorga en cada caso. Por tanto,esta función devolverá un valor entre 0 y 100, que comprueba el desempeño académico y profesional del mismo respecto al mejor caso posible. La fórmula de comparación por tanto queda enunciada de la siguiente manera:

$$Vc = \frac{\sum_{1}^{n}(Pr_i \times PI_{ij})}{Mcp}$$

Donde:

Vc: valoración de un caso.

n: cantidad de casos.

$Pr_i$: prioridad del rasgo i.

$PI_{ij}$: peso informacional del rasgo i en el rango j.

Mcp: mejor caso posible (es una constante que reflejará el valor del mejor caso posible que en este caso es 43)

El resultado obtenido es el promedio de recomendación del profesional en esa línea.

## 2.6 Empleo de la metodología Scrum

Scrum es un marco de trabajo por el cual las personas pueden acometer problemas complejos adaptativos, a la vez que entregar productos del máximo valor posible productiva y creativamente. Scrum no es un proceso o una técnica para construir productos; en lugar de eso, es un marco de trabajo dentro del cual se pueden emplear varias técnicas y procesos. Scrum muestra la eficacia relativa de las prácticas de gestión de producto y las prácticas de desarrollo, de modo que se pueda mejorar. El marco de trabajo Scrum consiste en los Equipos Scrum, roles, eventos, artefactos y reglas asociadas. Cada componente dentro del marco de trabajo sirve a un propósito específico y es esencial para el éxito de Scrum y para su uso. Las reglas de Scrum relacionan los eventos, roles y artefactos, gobernando las relaciones e interacciones entre ellos(Schwaber, Sutherland 2014).

### 2.6.1 Scrum Team

El Equipo Scrum consiste en un Dueño de Producto (Product Owner), el Equipo de Desarrollo (Development Team) y un Scrum Master, estos a su vez son los roles de la metodología El modelo de equipo en Scrum está diseñado para optimizar la flexibilidad, la creatividad y la productividad(Schwaber, Sutherland 2014).

**El Dueño de Producto (Product Owner)**

Es el responsable de maximizar el valor del producto y del trabajo del Equipo de Desarrollo. El Dueño de Producto es la única persona responsable de gestionar la Lista del Producto (Product Backlog). La gestión de la Lista del Producto incluye:

1. Expresar claramente los elementos de la Lista del Producto.
2. Ordenar los elementos en la Lista del Producto para alcanzar los objetivos y misiones de la mejor manera posible.
3. Optimizar el valor del trabajo desempeñado por el Equipo de Desarrollo.
4. Asegurar que la Lista del Producto es visible, transparente y clara para todos, y que muestra aquello en lo que el equipo trabajará a continuación.
5. Asegurar que el Equipo de Desarrollo entiende los elementos de la Lista del Producto al nivel necesario(Schwaber, Sutherland 2014).

El Dueño de Producto podría hacer el trabajo anterior, o delegarlo en el Equipo de Desarrollo. Sin embargo, en ambos casos el Dueño de Producto sigue siendo el responsable de dicho trabajo. Se definieron como dueños del producto al cliente de la herramienta, o sea, al tutor de la presente investigación, en este caso a la Ing. Yadira B. Reyes García.

**El Equipo de Desarrollo (Development Team)**

El Equipo de Desarrollo consiste en los profesionales que desempeñan el trabajo de entregar un Incremento de producto "terminado", que potencialmente se pueda poner en producción, al final de cada Sprint. Solo los miembros del Equipo de Desarrollo participan en la creación del Incremento.

Los Equipos de Desarrollo son estructurados y empoderados por la organización para organizar y gestionar su propio trabajo. La sinergia resultante optimiza la eficiencia y efectividad del Equipo de Desarrollo(Schwaber, Sutherland 2014).

**El Scrum Master**

El Scrum Master es el responsable de asegurar que Scrum es entendido y adoptado. Los Scrum Masters hacen esto asegurándose de que el Equipo Scrum trabaja ajustándose a la teoría, prácticas y reglas de Scrum.

El Scrum Master es un líder que está al servicio del Equipo Scrum. El Scrum Master ayuda a las personas externas al Equipo Scrum a entender qué interacciones con el Equipo Scrum pueden ser de ayuda y cuáles no. El Scrum Master ayuda a todos a modificar estas interacciones para maximizar el valor creado por el Equipo Scrum(Schwaber, Sutherland 2014).

En ambos casos se definió como Equipo de Desarrollo y Scrum Master al autor del presente trabajo.

### 2.6.2 Sprints

Los Sprints contienen y consisten de la Reunión de Planificación del Sprint (*Sprint PlanningMeeting*), los Scrums Diarios (*Daily Scrums*), el trabajo de desarrollo, la Revisión del Sprint (*Sprint Review*), y la Retrospectiva del Sprint (*Sprint Retrospective*).

Durante el Sprint:

1. No se realizan cambios que puedan afectar al Objetivo del Sprint (*Sprint Goal*).
2. Los objetivos de calidad no disminuyen.
3. El alcance puede ser clarificado y renegociado entre el Dueño de Producto y el Equipo de Desarrollo a medida que se va aprendiendo más(Schwaber, Sutherland 2014).

Cada Sprint puede considerarse un proyecto con un horizonte no mayor de un mes.Estos tienen una definición de qué se va a construir, un diseño y un plan flexible que guiará la construcción y el trabajo y el producto resultante(Schwaber, Sutherland 2014).

Fueron definidos los siguientes Sprints, todos con una duración de 30 días. Véase **Tabla 9**.

*Tabla 9. Cronología de los Sprints definidos en la implementación*

| SPRINT | COMIENZO | ENTREGA |
|---|---|---|
| **Primer sprint** | Lunes 8 de febrero de 2016 | Miércoles 9 de marzo de 2016 |

| | | |
|---|---|---|
| **Segundo sprint** | Jueves 10 de marzo de 2016 | Sábado 9 de abril de 2016 |
| **Tercer sprint** | Domingo 10 de abril de 2016 | Martes 10 de mayo de 2016 |
| **Cuarto sprint** | Miércoles 11 de mayo de 2016 | Sábado 11 de junio de 2016 |

En los días de comienzo de cada Sprint se realizó la Reunión de Planificación de Sprint (*Sprint Planning Meeting*) en la misma se analizó lo ya hecho y lo que falta por hacer, se definieron los Objetivos del Sprint (Sprint goals) que son una meta establecida para el Sprint que puede ser alcanzada mediante la implementación de la Cartera de Producto (Product Backlog).Además, en estas reuniones se dio respuesta a dos preguntas:

¿Qué puede entregarse en el Incremento resultante del Sprint que comienza?

¿Cómo se conseguirá hacer el trabajo necesario para entregar el Incremento?

**Revisión de Sprint (Sprint Review)**

Al final del Sprint se lleva a cabo una Revisión de Sprint para inspeccionar el Incremento y adaptar la Lista de Producto si fuese necesario. Durante la Revisión de Sprint, el Equipo Scrum y los interesados colaboran acerca de lo que se hizo durante el Sprint. Basándose en esto, y en cualquier cambio a la Lista de Producto durante el Sprint, los asistentes colaboran para determinar las siguientes cosas que podrían hacerse para optimizar el valor. Se trata de una reunión informal, no una reunión de seguimiento, y la presentación del Incremento tiene como objetivo facilitar la retroalimentación de información y fomentar la colaboración. El Scrum Masterse asegura de que el evento se lleve a cabo y que los asistentes entiendan su propósito. El ScrumMaster enseña a todos a mantener el evento dentro del bloque de tiempo fijado(Schwaber, Sutherland 2014). Las revisiones se llevaron a cabo los días de entrega del Sprint.

### 2.6.3 Artefactos de Scrum

Los artefactos de Scrum representan trabajo o valor en diversas formas que son útiles para proporcionar transparencia y oportunidades para la inspección y adaptación. Los artefactos definidos por Scrum están diseñados específicamente para maximizar la transparencia de la información clave, que es necesaria para asegurar que todos tengan el mismo entendimiento del artefacto(Schwaber, Sutherland 2014).

**Product Backlog**

La Cartera de Producto (Product Backlog) es una lista ordenada de todo lo que podría ser necesario en el producto, y es la única fuente de requisitos para cualquier cambio a realizarse en el producto. El Dueño de Producto (Product Owner) es el responsable de la Lista de Producto, incluyendo su contenido, disponibilidad y ordenación.

El desarrollo más temprano de la Cartera de Producto solo refleja los requisitos conocidos y mejor entendidos al principio. LaCartera de Producto evoluciona a medida que el producto y el entorno en el que se usará también lo hacen. La Lista de Producto es dinámica; cambia constantemente para identificar lo que el producto necesita para ser adecuado, competitivo y útil.

La Cartera de Producto enumera todas las características, funcionalidades, requisitos, mejoras y correcciones que constituyen cambios a ser hechos sobre el producto para entregas futuras. Los elementos de la Cartera de Producto tienen como atributos la descripción, la ordenación, la estimación y el valor(Schwaber, Sutherland 2014).

En la **Tabla 10** se muestraun ejemplo de la Cartera de producto luego de terminado el primer Sprint, para mayor información consultar los anexos:

| ID DE LA HISTORIA | ENUNCIADO DE LA HISTORIA | ALIAS | ESTADO | ESFUERZO | SPRINT | PRIORIDAD |
|---|---|---|---|---|---|---|
| **R10-SREI-GELI** | Como un Product Owner, necesito Gestionar líneas de investigación con la finalidad de adicionar, modificar o eliminar líneas de investigación. | Gestionar LI | Planificada | 6 | 2 | 2 |
| **R11-SREI-GEAC** | Como un Product Owner, necesito Gestionar áreas del conocimiento con la finalidad de adicionar, modificar o eliminar áreas del conocimiento. | Gestionar AC | Planificada | 6 | 2 | 2 |
| **R12-SREI-GEGI** | Como un Product Owner, necesito Gestionar grupos de investigación con la finalidad de adicionar, modificar o eliminar grupos de investigación. | Gestionar GI | Planificada | 6 | 2 | 2 |
| **R13-SREI-BUES** | Como un Product Owner, necesito Buscar estudiantes para mediante parámetros de búsqueda encontrar estudiantes. | Buscar estudiante | | 4 | 3 | 1 |
| **R14-SREI-BUPR** | Como un Product Owner, necesito Buscar profesionales para mediante parámetros de búsqueda encontrar profesionales. | Buscar profesional | | 4 | 3 | 1 |
| **R15-SREI-RERE** | Como un Product Owner, necesito Realizar recomendaciones para mostrar reportes de estudiantes y profesionales recomendados. | Realizar recomendación | | 36 | 3, 4 | 9 |

*Tabla 10. Ejemplo de Cartera de Producto actualizada terminado el primer Sprint*

**Sprint Backlog**

La Cartera del Sprint (Sprint Backlog) es el conjunto de elementos de la Cartera de Producto seleccionados para el Sprint, más un plan para entregar el Incremento de producto y conseguir el

Objetivo del Sprint. La Lista de Pendientes del Sprint es una predicción hecha por el Equipo de Desarrollo acerca de qué funcionalidad formará parte del próximo Incremento y del trabajo necesario para entregar esa funcionalidad en un Incremento "terminado". La Cartera del Sprint hace visible todo el trabajo que el Equipo de Desarrollo identifica como necesario para alcanzar el Objetivo del Sprint.

Se muestra en la **Tabla 11** un ejemplo de Sprint Backlog utilizado en el primer Sprint, para mayor información se puede encontrar el resto de los Sprint Backlog en los anexos:

*Tabla 11. Ejemplo de Sprint Backlog a los 23 días del primer Sprint*

| HISTORIA DEL PRODUCT BACKLOG | TAREAS | HECHAS | EN PROCESO | COMPLETADO | PRIORIDAD |
|---|---|---|---|---|---|
| Como un Product Owner, necesito Autenticar usuarios con la finalidad de que accedan a la aplicación las personas pertinentes. | 4 | 4 | 0 | 100% | 5 |
| Como un Product Owner, necesito Gestionar usuarios con la finalidad de adicionar, modificar o eliminar a personas con acceso a la aplicación | 8 | 6 | 2 | 75% | 5 |
| Como un Product Owner, necesito Gestionar estudiantes con la finalidad de adicionar, modificar o eliminar a profesionales | 12 | 7 | 5 | 58% | 2 |
| Como un Product Owner, necesito Gestionar profesionales con la finalidad de adicionar, modificar o eliminar a profesionales | 12 | 7 | 5 | 58% | 2 |
| Como un Product Owner, necesito Gestionar asignaturas con la finalidad de adicionar, modificar o eliminar asignaturas | 12 | 3 | 9 | 25% | 2 |
| Total | 54 | 33 | 21 | 61% | 3 |

**Burndown Chart**

Un gráfico de trabajo (Burndown Chart) pendiente a lo largo del tiempo muestra la velocidad a la que se están completando los requisitos funcionales. Permite extrapolar si el equipode desarrollo podrá completar el trabajo en el tiempo estimado.

Se pueden utilizan los siguientes gráficos de esfuerzo pendiente:

1. Días pendientes para completar los requisitos del producto o proyecto (Product burndown chart), realizado a partir de la lista de requisitos priorizada (Product Backlog).
2. Días pendientes para completar las tareas de la iteración (Sprint burndown chart), realizado a partir de la lista de tareas de la iteración (Sprint Backlog).

Se muestran en la

el burndown chart para el ejemplo anterior de Sprint Backlog y otro en la

**Figura 13. Burndown chart del primer Sprint Backlog**

**Figura 14** para mostrar el avance en el Product Backlog hasta el tercer Sprint.

El gráfico de trabajo de la Cartera de Sprint en la **¡Error! No se encuentra el origen de la referencia.** consiste en mostrar la cantidad de tareas a realizarse durante este y el tiempo destinado al cumplimiento de las mismas. Gráficamente se puede apreciar la cantidad de tareas por realizar de acuerdo al estimado para la realización de las actividades conforme pasan los días y el tiempo de ejecución real de las mismas. En este ejemplo para el final del primer sprint hubo un ligero retraso en el cumplimiento de las actividades, lo cual provocó que el próximo sprint se re-estructurara adicionándose las cuatro actividades que no pudieron concluirse a tiempo.

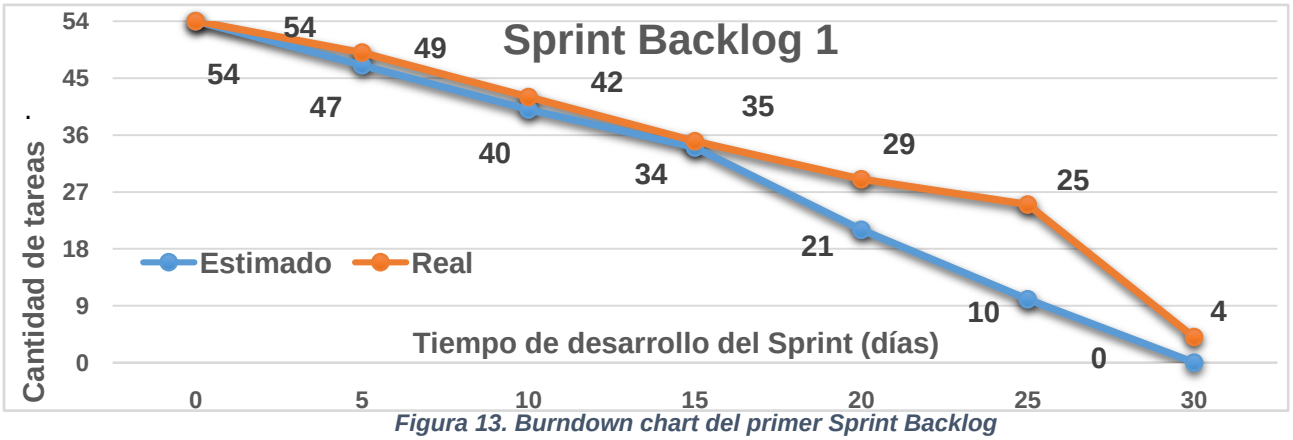



*Figura 13. Burndown chart del primer Sprint Backlog*

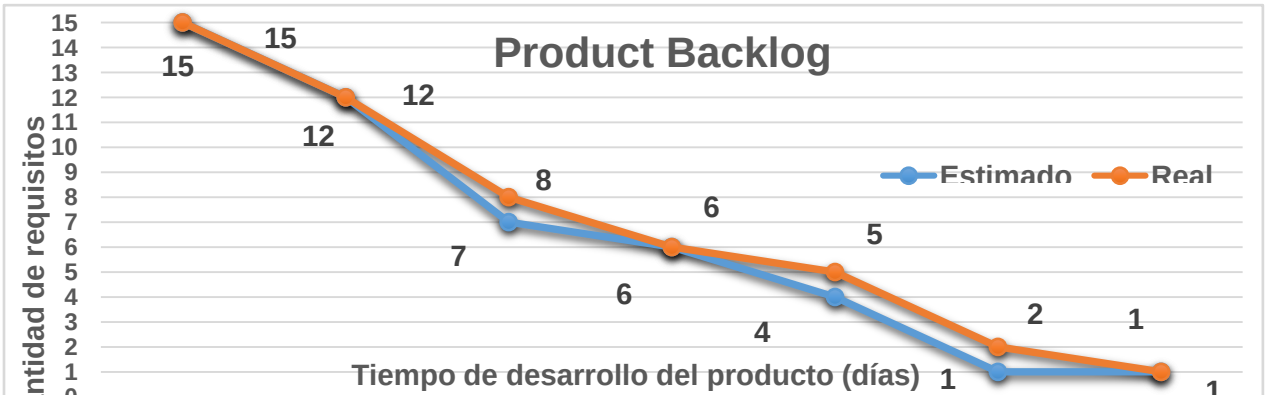


*Figura 14. Burndown chart del Product Backlog hasta el tercer Sprint.*

En la

**Figura 13. Burndown chart del primer Sprint Backlog**

**Figura 14** se muestra un ejemplo del gráfico de trabajo de la Cartera de Producto, que consiste en mostrar la cantidad de requisitos funcionales a implementar durante la etapa de desarrollo y el tiempo destinado al cumplimiento de las mismas. Gráficamente se puede apreciar la cantidad de requisitos por realizar de acuerdo al estimado para la conclusión de la implementación de estos por cada Sprint días además del tiempo de ejecución real de las mismas. La cantidad de días es 90 ya que constituye la duración de los primeros tres Sprints.

## Conclusiones Parciales

Como resultado del capítulo se conceptualizó el negocio, lo que permitió identificar los requisitos del sistema y un listado de requisitos no funcionales a tener en cuenta para el desarrollo de la herramienta. La aplicación de los patrones de diseño ofreció soluciones simples y elegantes a problemas específicos y comunes del diseño Orientado a Objetos. El diseño de la base de datos donde permitió mostrar las entidades que componen la misma y sus relaciones para asegurar una buena funcionalidad del sistema. La descripción de la solución propuesta planteó los aspectos necesarios para responder a los requisitos funcionales. El uso de la metodología de desarrollo Scrum y sus artefactos evidenció la utilización de una guía viable para realizar el proceso de desarrollo de software.

# Capítulo 3. Validación del sistema

El presente capítulo está orientado a las pruebas y validación de las variables de la investigación que se llevan a cabo para dar total cumplimiento a los requisitos establecidos por el cliente. Se evidencian las pruebas aplicadas a la solución, con el objetivo de comprobar las funcionalidades en los diferentes escenarios y así verificar en todos los casos que los resultados sean los esperados. Terminada las mismas se exponen los resultados obtenidos mediante la prueba piloto ejecutada por los especialistas demostrando el cumplimiento del problema planteado mediante el uso de la herramienta del SR.

## 3.1 Validación de requisitos

Esta actividad se realizó con el objetivo de garantizar que los requisitos fueran correctos y cumplieran con las necesidades del cliente. Se tuvieron en cuenta las métricas para medir la estabilidad y especificidad.

### 3.1.1 Métrica de estabilidad de requisitos

Es necesario lograr una estabilidad en los requisitos para el correcto funcionamiento de los demásflujos de trabajo. Se considera que los requisitos son estables cuando no existen adiciones o supresiones en ellos que impliquen modificaciones en las funcionalidades principales de la aplicación (Sánchez 2010).

$$EstR = \left[\frac{TR - RM}{TR}\right] \times 100$$

Dónde:

EstR: valor de la estabilidad de los requisitos.

TR: total de requisitos definidos.

RM: número de requisitos modificados, que se obtienen como la sumatoria de los requisitos insertados, modificados y eliminados.

Esta métrica ofrece valores entre 0 y 100. El mejor valor de EstR es el más cercano a 100. Teniendo en cuenta que se identificaron un total de 15 requisitos funcionales, de los cuales 2 resultaron modificados para el desarrollo del cuarto Sprint de un total de 5, se calcula:

$$EstR = [(15 - 2)/15] \times 100 = 86{,}67$$

Como resultado se obtuvo un valor de 86,67. Dicha cifra demuestra que no se han realizado cambios significativos sobre los requisitos, son estables y, por tanto, es confiable trabajar el diseño sobre ellos.

### 3.1.2 Métrica de especificidad de los requisitos

El objetivo de esta métrica es cuantificar la especificidad o falta de ambigüedad en la definición de los requisitos. Para calcular esta métrica se contaron los requisitos que tuvieron igual interpretación por los revisores y se compararon con el total de requisitos definidos (Sánchez 2010). Para ello se contó con un número de especialistas quienes revisaron los requisitos para saber si su interpretación coincidía o no con la definición de cada uno de estos. En la se muestran la relación de revisores consultados.

*Tabla 12. Revisores consultados para evaluar la especificidad de requisitos.*

| NOMBRE DEL REVISOR | CARGO QUE OCUPA | ÁREA | TOTAL DE COINCIDENCIAS |
|---|---|---|---|
| Yadira Beatriz Reyes García | Profesor | Departamento de Ingeniería de Software Facultad 3 | 15/15 |
| Yarilaisi Blanco Sánchez | Especialista de Calidad de Software | Centro de Informatización y Seguridad Digital | 15/15 |
| Arlety Sánchez Santos | Profesor | Departamento de Ingeniería de Software Facultad 2 | 15/15 |
| Yor Alex Remond Recio | RGA | Centro de Telemática | 14/15 |

La especificidad de los requisitos se calcula como:

$$EspR = {}^{n_{ri}}/_{n_r}$$

Dónde:

EspR: grado de especificidad de los requisitos.

$n_{ri}$: número de requisitos para los que todos los revisores tuvieron interpretaciones idénticas.

$n_r$:cantidad de requisitos en una especificación.

El valor de esta métrica debe estar siempre entre 0 y 1. Mientras más cerca de 1 esté el valor de EspR mayor será la consistencia de la interpretación de los revisores para cada requisito y menor será la ambigüedad en la especificación de los requisitos.

Luego:

$$EspR = {}^{n_{ri}}/_{n_r} = {}^{14}/_{15} = 0{,}93$$

Como resultado se obtuvo que la especificación de los requisitos presenta muy poca ambigüedad, asegurándose un alto nivel de calidad en el proceso de especificación.

## 3.2 Validación del diseño mediante métricas

Un aspecto importante a tener en cuenta en la evaluación del diseño, es la creación de métricas básicas inspiradas en el estudio de la calidad del diseño orientado a objetos. Para medir la calidad del diseño realizado se tuvieron en cuenta distintos atributos como son la responsabilidad de las clases, la reutilización y la complejidad de implementación y mantenimiento. Las métricas Tamaño Operacional de Clases (TOC) y Relaciones entre Clases (RC) permiten evaluar estos atributos y adicionalmente el bajo acoplamiento y la cantidad de pruebas unitarias que deben realizarse (Camacho, Cardeso, Núñez 2004). Véase **Tabla 13**.

| MÉTRICA TOC | MÉTRICA RC |
|---|---|
| Responsabilidad | Acoplamiento |
| Complejidad | Complejidad de Mantenimiento |
| Reutilización | Cantidad de Pruebas |

*Tabla 13. Atributos de las métricas de validación*

### 3.2.1 Métrica tamaño operacional de clase (TOC)

Seguidamente se muestra un ejemplo del resultado de la aplicación de esta métrica al diseño en la **Tabla 14**. En la **Figura15** se muestra la representación gráfica del resultado.

*Tabla 14. Análisis de los atributos de la métrica TOC*

| RESPONSABILIDAD | CANTIDAD DE CLASES | PROMEDIO SOBRE EL TOTAL |
|---|---|---|
| Alta | 2 | 15,38 |
| Media | 4 | 30,76 |
| Baja | 7 | 53,84 |
| **COMPLEJIDAD** | **CANTIDAD DE CLASES** | **PROMEDIO SOBRE EL TOTAL** |
| Alta | 2 | 69,23 |
| Media | 2 | 15,38 |
| Baja | 9 | 15,38 |
| **REUTILIZACIÓN** | **CANTIDAD DE CLASES** | **PROMEDIO SOBRE EL TOTAL** |
| Alta | 7 | 15,38 |
| Media | 4 | 30,76 |
| Baja | 2 | 53,84 |

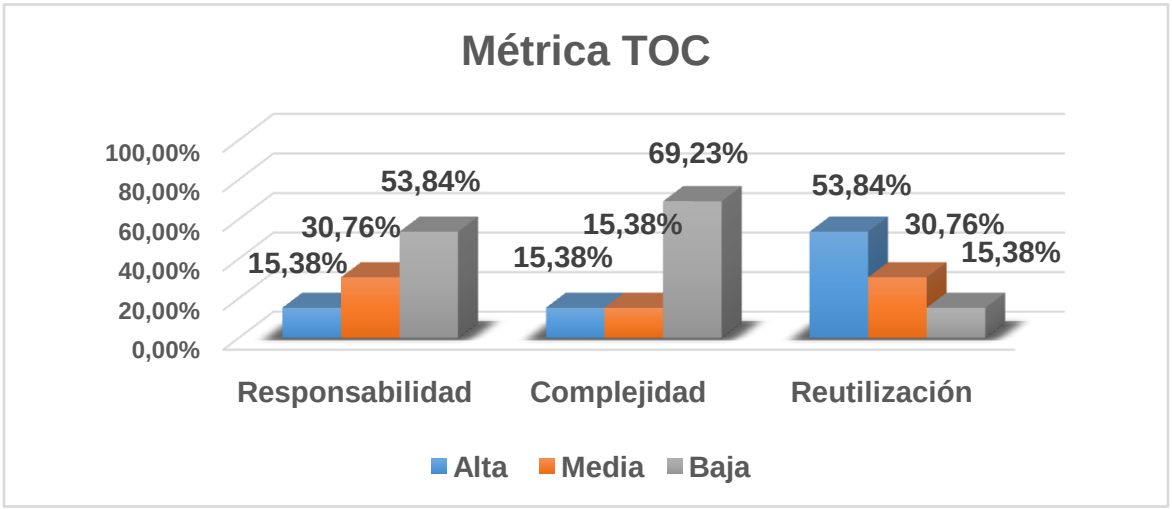


*Figura15. Gráfica de aplicación de la métrica TOC*

Después de aplicar la métrica TOC a un total de 13 clases, se llega a la conclusión de que el diseño propuesto cumple con lo requerido inicialmente ya que el mismo arrojó como resultado una alta reutilización, baja responsabilidad y complejidad, permitiendo la eliminación de clases repetidas, teniendo en cuenta que más de la mitad de las clases poseen baja dependencia respecto a otras, evidenciando que el sistema no presenta una compleja implementación en cuanto a lo que diseño se refiere.

### 3.2.2 Métrica de relaciones entre clases (RC)

Se muestra un ejemplo del resultado de la aplicación de esta métrica al diseño en la **Tabla 15**. En la **Figura 16** se muestra la representación gráfica del resultado.

*Tabla 15. Análisis de los atributos de la métrica RC*

| **CANTIDAD DE PRUEBAS** | **CANTIDAD DE CLASES** | **PROMEDIO SOBRE EL TOTAL** |
|---|---|---|
| Alta | 1 | 7,69 |
| Media | 5 | 38,46 |
| Baja | 7 | 53,84 |
| **COMPLEJIDAD DE MANTENIMIENTO** | **CANTIDAD DE CLASES** | **PROMEDIO SOBRE EL TOTAL** |
| Alta | 2 | 15,38 |
| Media | 4 | 30,77 |
| Baja | 7 | 53,84 |
| **ACOPLAMIENTO** | **CANTIDAD DE CLASES** | **PROMEDIO SOBRE EL TOTAL** |
| Alta | 3 | 23,07 |
| Media | 0 | 0,00 |
| Baja | 9 | 69,23 |
| Ninguna | 1 | 7,69 |

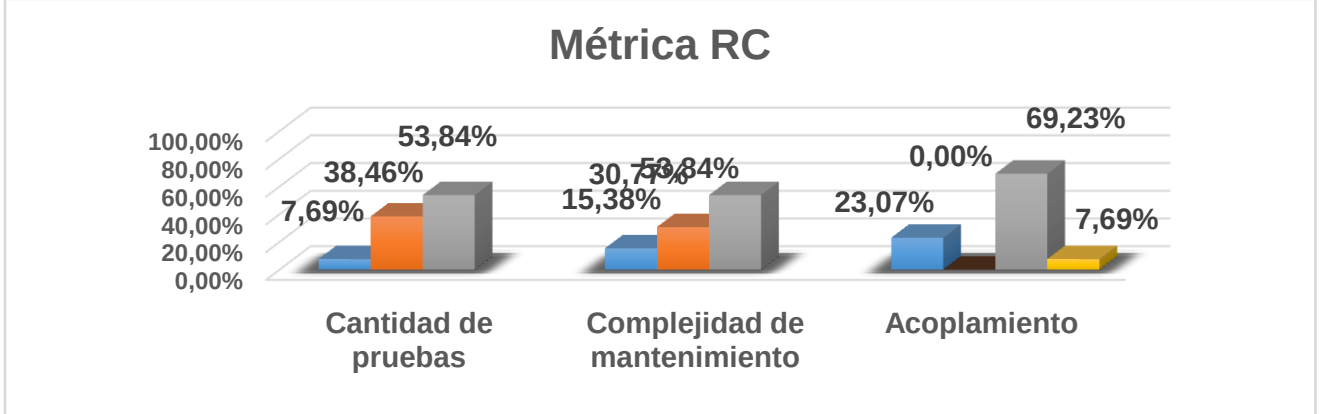


*Figura 16. Gráfica de aplicación de la métrica RC*

Los resultados obtenidos durante la evaluación del instrumento de medición de la métrica RC demuestran que el diseño propuesto se encuentra dentro de los niveles de calidad requeridos. Los atributos de calidad fueron evaluados satisfactoriamente confirmando que las clases están diseñadas correctamente ya que presentan un bajo acoplamiento, poca complejidad de mantenimiento y poca cantidad de pruebas que implica que se necesita menos esfuerzo a la hora de realizar pruebas unitarias a estas clases.

## 3.3 Estándares de codificación

Los estándares de código son una forma de "normalizar" la programación de forma tal que, al trabajar en un proyecto, cualquier persona involucrada en el mismo tenga acceso y comprenda el código (Microsoft 2016). A continuación, se presentan diferentes ejemplos de estilos de codificación definidos para la implementación de la aplicación:

**Nombre de los archivos**

informes: report_módulo_nombre

Ej. `<template id= "report_estudiante_listarrecomendacion">`

vista: módulo_nombre

Ej. `estudiante_add.xml`

wizard: módulo_nombre_wizard

Ej. `estudiante_add_wizard.xml`

**Clases de un modelo**

Herencia: class herencia_nombre

Ej. `class herencia_persona (models.Model)`

Normal: class módulo_nombre

Ej. `class persona_estudiante (models.Model)`

**Vistas**

vista Formulario:id = ¨módulo_modelo_form ¨

Ej. `<record id= "view_estudiante_form" model="ir.ui.view">`

vista Arbol: id= ¨módulo_modelo_tree ¨

Ej. `<record id= "view_estudiante_listar_tree" model="ir.ui.view">`

**Menú**

id = ¨menú_módulo_nombre¨

Ej. `<menuitem id= "menu_estudiante_recomendacion" model="ir.ui.view">`

## 3.4 Pruebas realizadas al sistema

Una vez finalizada la implementación del producto que se solicita, es necesario realizarle pruebas, con el objetivo de detectar errores en la aplicación y la documentación e identificar posibles fallos de implementación, calidad, o usabilidad del software. Además, medir el grado en que el software cumple con los requerimientos. Este proceso resulta de gran importancia porque proporciona una medida de la calidad de dicho sistema, siempre que se ejecute de manera correcta(Pressman 2005).

### 3.4.1 Prueba de caja negra

Según (Pressman, 2005), *Las pruebas de* caja negra, también denominada prueba de comportamiento, se centran en los requisitos funcionales del software.

Los casos de prueba de la caja negra pretenden demostrar que:

1. Las funciones del software son operativas.
2. La entrada se acepta de forma correcta.
3. Se produce una salida correcta.
4. La integridad de la información externa se mantiene.

**Resultados de la prueba de caja negra.**

La siguiente grafica muestra los resultados obtenidos en las pruebas realizadas al sistema durante cada sprint; en búsqueda de no conformidades de funcionalidad, redacción e interfaz. Obteniendo un resultado satisfactorio durante la última iteración.

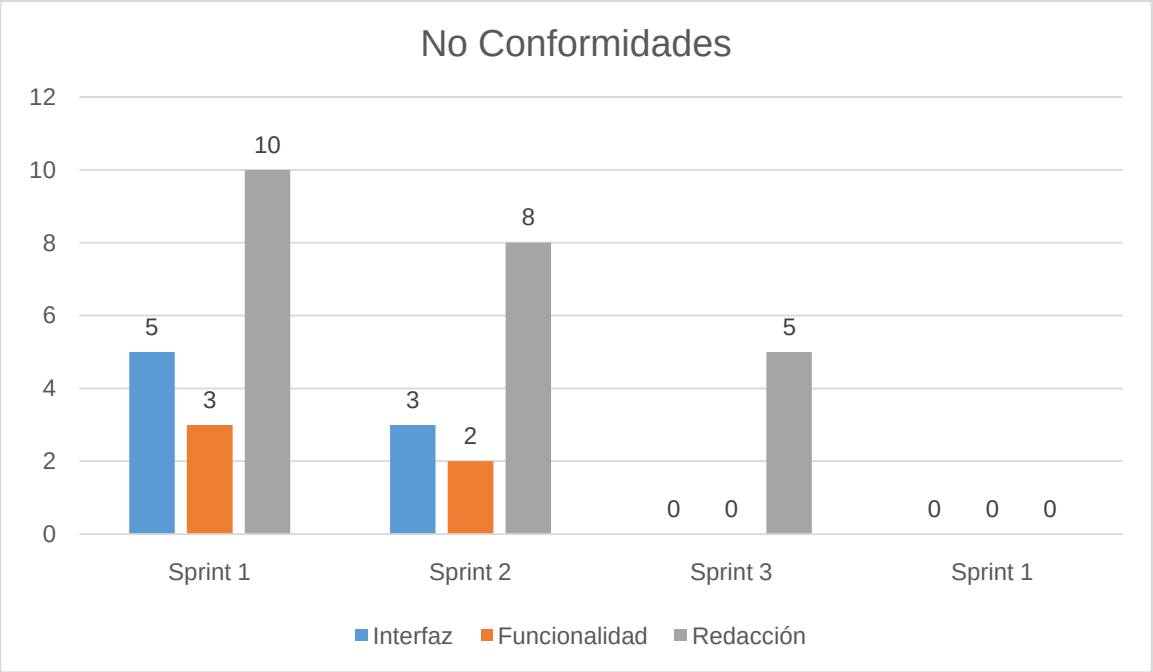



*Figura 17. Gráfica de No Conformidades*

### 3.4.2. Prueba de caja blanca

La prueba de caja blanca, denominada a veces prueba de caja de cristal es un método de diseño de casos de prueba que usa la estructura de control del diseño procedimental para obtener los casos de prueba. Mediante los métodos de prueba de caja blanca, el ingeniero del software puede obtener casos de prueba que garanticen que se ejercita por lo menos una vez todos los caminos independientes de cada módulo (Pressman 2005).

**Técnica del camino básico**

Es una técnica de prueba de caja blanca propuesta inicialmente por Tom McCabe. El método del camino básico permite al diseñador de casos de prueba obtener una medida de la complejidad lógica de un diseño procedimental y usar esa medida como guía para la definición de un conjunto básico de caminos de ejecución(Pressman 2005).

**Complejidad ciclomática**

La complejidad ciclomática es una métrica del software que proporciona una medición cuantitativa de la complejidad lógica de un programa. Cuando se usa en el contexto del método de prueba del camino básico, el valor calculado como complejidad ciclomática define el número de caminos independientes del conjunto básico de un programa y nos da un límite superior para el número de pruebas que se deben realizar para asegurar que se ejecuta cada sentencia al menos una vez(Pressman 2005).

A continuación, se muestra el código de la funcionalidad get_recomendacion_estudiante (obtener recomendación de estudiante) de la clase recomendacion, la cual constituye uno de los requisitos de

mayor importancia para la construcción del sistema, debido a que realiza las recomendaciones para los estudiantes, además constituye uno de los requisitos de mayor complejidad a implementar.

```
1def get_ recomendacion_estudiante(self, estudiante, lista_asignatura):
2sql = "SELECT l1.linea,l1.area_del_conocimiento FROM estudiantel1 WHERE l1.estudiante_id = "
+ str(
self.estudiante_id.id) + " and l1.ponderacion>= '" + nombre + "' and l1.valor<= '" +
apellidos + "'"
self.env.cr.execute(sql)
    lista_asignatura = [t for t in self.env.cr.fetchall()]
    total_lista_asignatura= []
3if not lista_lista_asignatura:
4return []
9else:
5ponderacion = 0.00
6valor = 0
7for asignaturain lista_asignatra:
8total_estudiante.append(ponderacion + (move_line[0] - move_line[1]))
            aux = total_estudiante[contador]
            contador += 1
9return total_estudiante
```

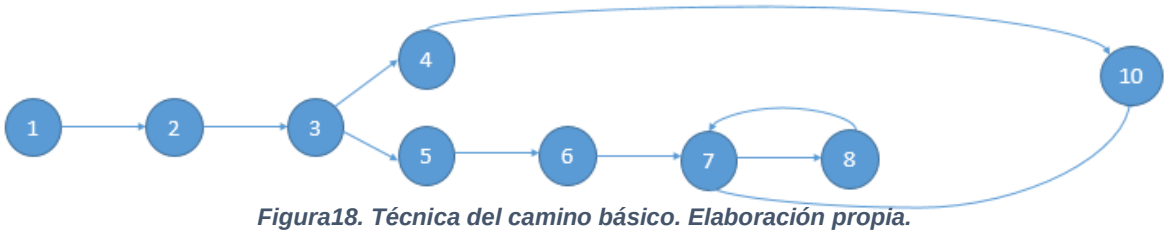



*Figura18. Técnica del camino básico. Elaboración propia.*

V (G) = A – N + 2
= 11 – 10 +2
= 3

V (G) = Complejidad ciclomática
A = Cantidad de aristas del grafo
N = Cantidad de nodos del grafo

V (G) = P +1
= 2+1
= 3

P= Nodos predicados. Nodos que tienen más de una arista de salida

V (G) = Regiones
3 = 3

Regiones: Áreas delimitadas por nodos y aristas del grafo

Cantidad de caminos mínimos = 3

Camino 1: 1, 2, 3, 4,9

Camino 2: 1, 2, 3, 5, 6, 7, 8, 7,9

Camino 3: 1, 2, 3, 5, 6, 7, 9

**Caso de prueba para el camino 2**

*Tabla 16. Diseño de caso de prueba para el camino 2.*

| | |
|---|---|
| Descripción | A partir de seleccionar datos, se comprueba si estos datos son válidos para adicionarlos posteriormente |

| | |
|---|---|
| Condición de ejecución | Se selecciona los campos estudiante, área del conocimiento y línea temática |
| Entrada | Estudiante, lista de asignaturas , ponderación y valor |
| Resultado | Realiza las recomendaciones y muestra el valor de ponderación |

### 3.4.3 Pruebas de aceptación

Es la prueba final antes del despliegue del sistema. Su objetivo es verificar que el software está listo y que puede ser usado por usuarios finales para ejecutar aquellas funciones y tareas para las cuales el software fue construido.En esta prueba se evalúa el grado de calidad del software con relación a todos los aspectos relevantes para que el uso del producto se justifique(Software, 2016). En la **Tabla 17** se muestra la cantidad de no conformidades encontradas en las dos iteraciones, las cuales fueron resueltas, obteniéndose así la carta de aceptación del cliente.

*Tabla 17. Sprint1-4. Pruebas de aceptación.*

| Criterio | Sprint 1 | Sprint 2 | Sprint 3 | Sprint 4 |
|---|---|---|---|---|
| No Conformidades(NC) | 23 | 18 | 7 | 0 |
| Significativas | 16 | 12 | 3 | 0 |
| No significativas | 0 | 0 | 0 | 0 |
| Recomendaciones | 7 | 6 | 2 | 0 |
| Ortografía | 10 | 8 | 5 | 0 |
| Excepciones | 3 | 2 | 0 | 0 |
| Interfaz | 5 | 3 | 0 | 0 |

## Conclusiones Parciales

Al concluir el desarrollo de este capítulo se obtuvo un sistema que cumple satisfactoriamente con los requerimientos definidos, para arribar a dicho resultado se tuvieron en cuenta los siguientes elementos:

1. Se validó el diseño y los requisitos para obtener una especificación del producto más robusta.
2. Se realizaron las pruebas necesarias para el control de la calidad del producto que permitieron la corrección de las no conformidades detectadas.
3. Se detalló el uso de un estándar de codificación, lo que proporcionó mayor organización y claridad, contribuyendo así a realizar una programación explícita.
4. Se obtuvo la implementación de todas las funcionalidades, garantizando el correcto funcionamiento mediante la prueba de rendimiento, además que esta tributó a la prueba de aceptación donde el cliente avaló la solución.

## Conclusiones Generales

Con la investigación realizada, el diseño y la implementación de sistema de recomendación para equipos de investigación se obtuvieron resultados que permiten arribar a las siguientes conclusiones:

1. El estudio de los sistemas de recomendación y de las técnicas de inteligencia artificial para descubrimiento de conocimiento permitió entender el funcionamiento, las características y los antecedentes de los mismos.
2. El análisis y diseño de la estructura de solución de la herramienta permitió facilitar la implementación del sistema.
3. La fase de implementación permitió desarrollar el sistema de recomendación para equipos de investigación acorde a los requisitos funcionales previamente establecidos.
4. Las fases de pruebas de software permitieron garantizar la verificación y validación del sistema, garantizando así la obtención de un sistema con mayor calidad.

## Recomendaciones

El autor de la presente investigación recomienda para posteriores versiones de la herramienta, profundizar en la investigación, para encontrar maneras más eficaces de realizar los procesos de recomendación e incluir el estudio del factor motivacional para lograr una herramienta más completa y aplicable en cualquier centro de educación superior del país para contribuir a mejorar los resultados docentes de los estudiantes

## Referencias Bibliográficas

ACM, 2014. Information Technology Competency Model of Core Learning Outcomes and Assessment for Associate-Degree Curriculum. S.l.: s.n. ISBN ISBN: 978-1-4503-3918-6.

ÁLVAREZ, Miguel A., 2016. Manual de jQuery. En: . 2016.

BERTATE, Alicia, MACHADO, Rodrigo y MOLINA, Valeria, 2006. PGMúsica Sistema de Recomendación de música [en línea]. Uruguay: Universidad de la Republica. [Accedido 15 noviembre 2015]. Disponible desde: www.fing.edu.uy/inco/grupos/pln/prygrado/InformePGMusica.pdf.

BURKE, Robin, 2002. Hybrid recommender systems: Survey and experiments. En: California State University. 2002. Vol. 12.

CAMACHO, Erika, CARDESO, Fabio y NÚÑEZ, Gabriel, 2004. Arquitecturas de software. Guía de estudio [en línea]. S.l. Universidad Simón Bolívar. [Accedido 15 febrero 2016]. Disponible desde: http://prof.usb.ve/lmendoza/Documentos/PS-6116/Guia%20Arquitectura%20v.2.pdf.

CARRILLO, Isaías, PÉREZ, Rodrigo y RODRÍGUEZ, Aureliano David, 2008. Metodología de Desarrollo de Software. S.l.: s.n.

CERVANTES, Humberto, 2010. Arquitectura de Software. En: Arquitectura [en línea]. 2010. Disponible desde: http://sg.com.mx/content/view/922.

CÉSARI, Matilde, 2012. Sistema Basado en Reglas. En: Scribd [en línea]. 2012. [Accedido 16 marzo 2016]. Disponible desde: http://es.scribd.com/doc/109492619/SISTEMAS-BASADOS-REGLAS.

CORTEZ, Augusto, NAVARRO, Carlos y PARIONA, Jaime, 2010. Sistemas de razonamiento basado en casos aplicado a sistemas de líneas de productos software. En: RISI [en línea]. 2010. Vol. 7. [Accedido 15 mayo 2016]. Disponible desde: http://revistasinvestigacion.unmsm.edu.pe/index.php/sistem/article/view/3278.

DE LUCA, Damian, 2010. CSS3 HTML5 [en línea]. S.l. [Accedido 21 febrero 2016]. Disponible desde: http://html5.dwebapps.com/que-es-css3/.

FONTELA, Álvaro, 2016. ¿Qué es Bootstrap? En: [en línea]. 2016. [Accedido 23 febrero 2016]. Disponible desde: https://raiolanetworks.es/blog/que-es-bootstrap/.

FONT, José María, 2008. Generación de Sistemas Basados en Reglas mediante Programación Genética. S.l.: Universidad Politécnica de Madrid.

FONT, Miguel, 2009. Sistemas de recomendación para webs de información sobre la salud. En: Universidad Politecnica de Cataluña [en línea]. 2009. [Accedido 22 noviembre 2015]. Disponible desde: http://upcommons.upc.edu/pfc/handle/2099.1/7193.

GAMMA, Erich, HELM, Richard, JOHNSON, Ralph y VLISSIDES, John, 1995. Design Patterns: Elements of Reusable Object-Oriented Software. S.l.: Addison-Wesley. ISBN 978-3-8266-9904-7.

GARCÍA, Diego, 2013. Manual de WEKA [en línea]. Universidad de Granada. [Accedido 15 marzo 2016]. Disponible desde: sci2s.ugr.es/sites/default/files/files/Teaching/GraduatesCourses/.../weka.pdf.

GARCÍA, José Manuel y MOLINA, Jesús, 2006. Técnicas de análisis de datos. En: Universidad Carlos III de Madrid [en línea]. 2006. [Accedido 22 noviembre 2015]. Disponible desde: https://senaintro.blackboard.com/bbcswebdav/institution/semillas/621121_1_VIRTUAL/Contenido/Documentos/Otros%20documentos/Material_apoyo_Gu%C3%ADa%206/TECNICA%20DE%20ANALISIS%20DE%20DATOS.pdf.

GIL, Arturo, 2015. Técnicas de Análisis de Datos en WEKA – ingeniería [en línea]. Universidad Miguel Hernández. [Accedido 18 marzo 2016]. Disponible desde: isa.umh.es/asignaturas/crss/turorialWEKA.pdf.

GUTIÉRREZ, Ileana, BELLO, Rafael E. y TELLERÍA, Andrés, 2002. Un Sistema Basado en Casos para la toma de decisiones en condiciones de incertidumbre. En: Investigación Operacional [en línea]. 2002. [Accedido 15 marzo 2016]. Disponible desde: http://www.invoperacional.uh.cu/index.php/IO/article/view/77.

GUTIÉRREZ, Ileana y PÉREZ, Rafael, 2012. Un SBC para la toma de decisiones en condiciones de incertidumbre. En: Investigación Operacional. 2012. Vol. 7.

HERLOCKER, J.L., KONSTAN, J.A., TERVEEN, L.G. y RIEDL, J.T., 2004. Evaluating Collaborative Filtering Recommender Systems. En: ACM Transactions on Information Systems [en línea]. 2004. Vol. 22. [Accedido 16 noviembre 2015]. Disponible desde: http://grouplens.org/site-content/uploads/evaluating-TOIS-20041.pdf.

HERNÁNDEZ, Rolando Alfredo y COELLO, Sayda, 2005. El paradigma cuantitativo de la investigación científica. S.l.: Editorial Universitaria. ISBN 978-959-16-0343-2.

HOLOVATY, A. y KAPLAN-MOSS, J., 2004. El libro de Django [en línea]. S.l.: IBM. [Accedido 6 julio 2016]. Disponible desde: http://www.ibm.com/developerworks/rational/library/content/04August/3153/3153_Rumbaugh_ch01.pdf.

JACOBSON, Ivar, 2009. El Proceso Unificado de Desarrollo de Software. 2009. S.l.: s.n.

JETBRAINS, 2010. PR Newswire. En: JetBrains [en línea]. 2010. [Accedido 6 junio 2016]. Disponible desde: http://www.prnewswire.co.uk/news-releases/con-pycharm-losdesarrolladores-de-python-obtienen-finalmente-una-ide-potente-155001805.html.

LARMAN, Craig, 1999. UML y Patrones. Introducción al análisis y diseño orientado a objetos. S.l.: Prentice Hall. ISBN 0-13-748880-7.

LARMAN, Craig, 2003. Introducción al análisis y diseño orientado a objetos. S.l.: s.n.

LOZANO, Fernando, 2013. Minería de datos para empresas. En: [en línea]. 2013. [Accedido 24 febrero 2016]. Disponible desde: http://www.fernandolozanopajaron.com/documentos/FernandoLozano-Memoria.pdf.

MARANTE, Danier, 2008. Aplicación de la minería de datos para la exploración y detección de patrones delictivos. Cuba: Universidad de las Ciencias Informáticas.

MARTINEAUX, Karina, 2008. Comparación de algoritmos de clasificación y agrupamiento aplicando técnicas de Minería de datos. En: . 2008.

MARTÍNEZ, Carlos Javier, 2014. Implantación y evaluación de un recomendador social de canciones. En: Universidad Politecnica de Valencia [en línea]. 2014. [Accedido 11 agosto 2015]. Disponible desde: https://riunet.upv.es/bitstream/handle/10251/53954/MART%C3%8DNEZ%20-

%20Implantaci%C3%B3n%20y%20evaluaci%C3%B3n%20de%20un%20recomendador%20social%20de%20canciones.pdf?sequence=3.

MICROSOFT, 2016. Estándares de Codificación [en línea]. 2016. S.l.: s.n. [Accedido 5 octubre 2016]. Disponible desde: https://msdn.microsoft.com/es-es/library/aa291591%28v=vs.71%29.aspx.

MONTES, Manuel, GELBUKH, Alexander y LÓPEZ, Aurelio, 2005. Minería de texto empleando la Semejanza entre Estructuras Semánticas. En: Computación y sistemas [en línea]. 2005. Vol. 9. [Accedido 2 junio 2015]. Disponible desde: http://www.scielo.org.mx/scielo.php?pid=S1405-55462005000300008&script=sci_abstract.

MOQUILLAZA, Santiago Domingo, VEGA, Hugo y GUERRA, Luis, 2010. Programación en N capas. En: RISI. 2010. Vol. 7.

MOYA, Ricardo, 2014. ¿Que son los Sistemas de Recomendación? En: Jarroba [en línea]. 2014. [Accedido 11 agosto 2015]. Disponible desde: http://jarroba.com/que-son-los-sistemas-de-recomendacion/.

PEINADO, Silvia Andrea, COAVAS, Elisaina Lorena, CORONADO, Eduar Jesús y NÚÑEZ, Isabel, 2011. Diseño E Implementación De Un Software Recomendador Y Adaptativo De Educación Básica Secundaria En Las Instituciones Educativas Del Municipio De Lorica-Cordoba. En: Buenas Tareas [en línea]. 2011. [Accedido 14 octubre 2015]. Disponible desde: http://www.buenastareas.com/ensayos/Dise%C3%B1o-e-Implementaci%C3%B3n-De-Un-Software/2714517.html.

PEÑA, Fernando Andrés y RIFFO, Ricardo Elías, 2008. Revisión, selección e implementación de un algoritmo de recomendación de material bibliográfico utilizando tecnología j2ee. Chile: Universidad del Bío-Bío.

PÉREZ, Damián, 2007. Maestros del Web. En: [en línea]. 2007. [Accedido 21 febrero 2016]. Disponible desde: http://www.maestrosdelweb.com/que-es-javascript/.

PRESSMAN, Roger S., 2005. Ingeniería del software. Un enfoque práctico. 6ta. S.l.: Prentice Hall.

PRESSMAN, Roger S., 2007. Ingeniería de software. Un enfoque práctico. 6. S.l.: Prentice Hall.

PYTHON SOFTWARE, Foundartion, 2012. Python Software [en línea]. S.l. Disponible desde: http://www.python.org/.

QUINLAN, J. Ross, 1993. C4.5 Programs for Machine Learning. S.l.: Morgan Kauffman. ISBN 978-1-55860-238-0.

RAE, 2014. Diccionario de la Lengua Española [en línea]. S.l. Real Academia de la Lengua española. [Accedido 6 mayo 2016]. Disponible desde: https://dle.rae.es.

RAPIDMINER, 2014. RapidMiner [en línea]. S.l.: s.n. [Accedido 22 febrero 2016]. Disponible desde: http://RapidMiner.com/.

ROBLEDANO, Ángel M., 2015. Tutorial Bootstrap 3: Introducción e instalación. En: [en línea]. 2015. [Accedido 24 febrero 2016]. Disponible desde: https://openwebinars.net/tutorial-bootstrap-3-introduccion-e-instalacion/.

SÁNCHEZ, José M., 2001. Arquitectura distribuida de control para sistemas con capacidades de data mining. España: Universidad Politécnica de Madrid.

SÁNCHEZ, M., 2010. Propuesta de una guía de métricas para evaluar el desarrollo de los Sistemas de Información Geográfica. En: Vinculando [en línea]. 2010. [Accedido 13 abril 2016]. Disponible desde: http://vinculando.org/articulos/sociedad_america_latina/propuesta_guia_de_medidas_para_evaluacion_sistemas_informacion.html.

SCHWABER, Ken y SUTHERLAND, 2014. La Guía Definitiva de Scrum: Las Reglas del Juego [en línea]. S.l. Scrum Org. [Accedido 28 noviembre 2015]. Disponible desde: www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-ES.pdf.

SCRUM, 2011. SCRUM [en línea]. S.l.: s.n. [Accedido 10 diciembre 2015]. Disponible desde: http://www.scrum.org.

SINBAD, Grupo de Investigación, 2012. Personalización y los Sistemas de Recomendación. En: Universidad de Jaen [en línea]. 2012. [Accedido 23 octubre 2015]. Disponible desde: http://sinbad2.ujaen.es/cod/archivosPublicos/presentaciones/sistemasDeRecomendacion.pdf.

SOMMERVILLE, 2005. Ingeniería de Software. 7ma. S.l.: Pearson. ISBN 84-7829-074-5.

TENDENZIAS, 2015. Qué es un entorno de desarrollo integrado, IDE. En: Tendenzias [en línea]. 2015. [Accedido 10 mayo 2015]. Disponible desde: http://programaciondesarrollo.es/que-es-un-entorno-de-desarrollo-integrado-ide.

UAP, 2007. ¿Qué es una tesis? S.l.

UCI, 2015. Documento Metodológico para la Carrera de Ingeniería en Ciencias Informáticas. Curso - 2016 2015. S.l.: s.n.

UML, 2011. UML [en línea]. S.l.: s.n. [Accedido 15 octubre 2016]. Disponible desde: http://www.uml.org.

UNLP, 2011. Líneas de Investigación. En: Universidad Nacional de La Plata [en línea]. 2011. [Accedido 6 enero 2016]. Disponible desde: http://perio.unlp.edu.ar/iicom/content/l%C3%ADneas-de-investigaci%C3%B3n.

VALENTÍN, Eliana Bárbara, RODRÍGUEZ, Rafael, PIÑERO, Pedro Y. y MARTÍNEZ, Hugo Arnaldo, 2012. Descubrimiento de conocimiento a partir de lecciones aprendidas documentadas en los procesos de cierre de proyectos informáticos. En: RCCI [en línea]. 2012. Vol. 7. [Accedido 21 enero 2016]. Disponible desde: rcci.uci.cu/?journal=rcci&page=article&op=view&path[]=494&path[]=229.

VERCELLIS, Carlo, 2009. Business-intelligence-data-mining-and-optimization-for-decision-making. Politecnico di Milano, Italy.: John Wiley & Sons. ISBN 978-0-470-51138-1.