



Universidad de las Ciencias Informáticas
Facultad 3

Título: Aplicación de un algoritmo de clasificación supervisada mediante una herramienta para el análisis de la influencia de los rasgos de personalidad en el rendimiento laboral.

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

Autor(es): Yoendry Betancourt Peña.

Tutora: Ing. Eilys Pacheco Rodríguez.

La Habana, Cuba

Junio, 2014



“La responsabilidad nuestra es luchar porque la calidad del producto que aquí se haga sea de los mejores y la mejor posible.”

ERNESTO CHE GUEVARA

DECLARACIÓN DE AUTORÍA

DECLARACIÓN DE AUTORÍA

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

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

Yoendry Betancourt Peña

(Autor)

Ing. Eilys Pacheco Rodríguez

(Tutora)

DATOS DE CONTACTO

Ing. Eilys Pacheco Rodríguez. (epacheco@uci.cu) Graduada de Ingeniería en Ciencias Informáticas en el año 2009. Profesor Instructor. Se desempeñó como Líder de Gestión en el área de producción del proyecto de desarrollo ERP-Cuba en el primer semestre del curso 2009 a 2010. Durante el segundo semestre del curso 2009-2010 ocupó el rol de Líder de Gestión del proyecto Aduana. Actualmente es profesora en el Departamento de Ingeniería y Gestión de Software de la Facultad3 y sus investigaciones científicas de la maestría Gestión de Proyectos se dirigen hacia el campo de la evaluación de los Recursos Humanos.

AGRADECIMIENTOS

A toda mi familia quien de una forma u otra se preocuparon por mi progreso en la universidad.

A mi padrastro Rafa que ha sido un padre más para mí, gracias por enseñarme a ser el hombre que soy hoy, aconsejarme a lo largo de estos años que hemos compartido juntos y por estar siempre pendiente de mí.

A mis abuelos Gladys, Mariliza, Lilo y Santiago quienes se han preocupado por mi bienestar en el transcurso de estos años y siempre se han preocupado por mi vida.

A mi mamá y papá por su constante apoyo y dedicación sin los cuales no hubiera tenido las fuerzas necesarias para sobreponerme a todos los problemas y obstáculos que encontré en mi camino hasta aquí, por ser los mejores padres del mundo, por darme la educación y el mejor ejemplo que pudiera seguir, que sepan que mejor no lo pudieron hacer, que me siento muy orgulloso de ellos y los quiero mucho.

A mi tutora la Ing. Eilys Pacheco Rodríguez que ha sido una madre para mí en la universidad por su ayuda incondicional y sus orientaciones en la realización de este trabajo; por su constante apoyo y aliento para seguir adelante en los momentos más difíciles; por guiarme a ser un buen profesional; por su sabiduría y su tiempo, sin los cuales no hubiese sido posible la realización de esta tesis. A ella mis sinceros agradecimientos por ayudarme a lograr este sueño.

A Leydis y Nay por ser mis amigas y hermanas durante todo este tiempo, por su preocupación constante y su eterno apoyo; por los momentos lindos que nunca vamos a olvidar y por estar siempre a mi lado en las buenas y en las malas.

A Leo por ser un hermano durante este tiempo y darme su apoyo de amigo incondicional, a Etna, Susel y Carlos por compartir buenos momentos con ustedes que no voy a olvidar.

A mis compañeros de aula por los buenos momentos que pasamos y los recuerdos que hoy llevamos con nosotros.

A mis amistades de Las Tunas por los momentos que hemos compartido.

Al tribunal, y en especial a la profesora Danaysa por brindarme su ayuda incondicional en mis estudios cada vez que lo necesitaba y ayudarme a ser el profesional que soy hoy.

A las amistades de Ventanilla Única profesionales y estudiantes.

Agradecerles a todas esas personas que de una forma u otra hicieron realidad este sueño.

DEDICATORIA

A mis padres: Adelaida y Alexander a los que debo lo que soy, que me han apoyado todos estos años, han tenido confianza en mí y sobre todo me han brindado su amor, cariño y comprensión.

A mis abuelitos Gladys, Mariliza, Lilo y Santiago, los quiero mucho.

RESUMEN

La competitividad entre las empresas en el transcurso de los años, se ha disparado considerablemente en la búsqueda de aportar innumerables innovaciones y aumentar su nivel de productividad. Por esta razón, en la actualidad las organizaciones están más conscientes de la necesidad de contar con óptimos Recursos Humanos para la obtención de buenos resultados y altos niveles de rendimiento en la industria. Muchas veces estos resultados y niveles no son los esperados, ya que pueden estar condicionados por el comportamiento, las características o rasgos de cada persona que labora en la institución. El propósito de esta investigación es mostrar la relación existente entre las características que hacen única a una persona y el nivel de rendimiento que pueda alcanzar ante las tareas asignadas en proyectos informáticos. Se presenta una propuesta de solución para analizar la relación entre estos dos factores, se propone la utilización de un algoritmo de Minería de Datos de clasificación supervisado para analizar los resultados e inferir las reglas del árbol generado, así como la identificación de los principales indicadores de rendimiento que facilitan el análisis. Además, para la aplicación del algoritmo se obtuvo una base de casos con datos adquiridos del Sistema de Gestión de Proyectos (GESPRO¹), con el propósito de contar con evidencias reales. La propuesta realizada es extensible, lo cual significa que se le pueden adicionar otros indicadores que sean relevantes para la investigación.

PALABRAS CLAVE

Rasgos, Personalidad, Rendimiento, Minería de Datos, Clasificación, Algoritmo, Recursos Humanos.

¹ GESPRO: Paquete de Herramientas de Gestión de Proyectos.

TABLA DE CONTENIDOS

INTRODUCCIÓN.....	9
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	16
1.1. Descubrimiento de conocimiento.....	16
1.2. Minería de Datos	17
1.2.1. Técnicas de Minería de Datos	18
1.3. Clasificación.....	19
1.3.1. Técnicas de clasificación.....	19
1.3.2. Selección de las variables	21
1.3.3. Algoritmos de clasificación de árboles de decisión	22
1.4. Herramientas usadas en el proceso de Minería de Datos.....	24
1.5. Rendimiento laboral	27
1.5.1. Evaluación del rendimiento laboral	28
1.5.2. Fuente de datos: GESPRO.....	28
1.5.3. Indicadores específicos del rendimiento laboral	29
1.6. Estudio de la personalidad en los RRHH	31
1.6.1. Modelos de Personalidad	31
1.6.2. Rasgos de Personalidad	35
1.7. Herramientas y tecnologías de desarrollo	36
1.7.1. Metodología de desarrollo Extreme Programming (XP)	37
1.7.2. Herramienta de modelado Visual Paradigm for UML 8.0 Enterprise Edition	37
1.7.3. Lenguaje de programación Java.....	38
1.7.4. Entorno de desarrollo integrado	39
1.8. Conclusiones Parciales	39
CAPÍTULO 2: PROPUESTA DE SOLUCIÓN	41
2.1. Descripción del algoritmo.....	41
2.2. Levantamiento y especificación de requisitos	43
2.2.1. Requisitos funcionales	44
2.2.2. Requisitos no funcionales	44
2.2.3. Historias de Usuario	45
2.2.4. Plan de iteraciones	46

2.2.5.	Plan de entrega	47
2.3.	Diseño de la herramienta	47
2.3.1.	Tarjetas CRC	47
2.3.2.	Diagrama de clase del diseño	48
2.3.3.	Patrones de diseño utilizados	48
2.3.4.	Métricas de diseño	51
2.4.	IMPLEMENTACIÓN	58
2.4.1.	Tareas de ingeniería	58
2.4.2.	Diagrama de componentes.....	58
2.4.3.	Estándares de codificación	59
2.5.	Pruebas.....	60
2.6.	Conclusiones del capítulo	60
CAPÍTULO 3: RESULTADOS Y VALIDACIÓN DE LA PROPUESTA.....		62
3.1.	DESCRIPCIÓN DE LA HERRAMIENTA	62
3.2.	VALIDACIÓN DE LA HERRAMIENTA.....	66
3.2.1.	Pruebas unitarias.....	66
3.2.2.	Pruebas de aceptación.....	68
3.3.	VALIDACIÓN DE LA PROPUESTA	69
3.5.	Conclusiones del capítulo	75
CONCLUSIONES GENERALES		76
RECOMENDACIONES		77
BIBLIOGRAFÍA.....		78
ÍNDICE DE TABLAS		
Tabla 1: Análisis comparativo de sistemas de minería de datos. Fuente: Elaboración Propia.		27
Tabla 3: Análisis comparativo de modelos/test psicológicos. Elaboración Propia.		34
Tabla 4: HU Mostrar el resultado de la salida del clasificador.		46
Tabla 6: Plan de entrega de las iteraciones.		47
Tabla 7: Descripción de la Tarjeta CRC PersonalidadModel.		48
Tabla 8: Clases fundamentales de la herramienta.		52
Tabla 9: Umbrales para la responsabilidad, complejidad y reutilización de la métrica TOC.		52
Tabla 10: Clases analizadas con resultados obtenidos con la métrica TOC.		53
Tabla 11: Umbrales para la reutilización, complejidad del mantenimiento, acoplamiento y cantidad de pruebas de la métrica RC.		54
Tabla 12: Clases analizadas y resultados obtenidos de la métrica RC.		55
Tabla 13: Atributos identificados de la clase Personalidad.....		56

Tabla 14: Métodos de la clase PersonalidadModel y atributos correspondientes.	57
Tabla 15: Total de métodos que acceden a un atributo de la clase PersonalidadModel.	57
Tabla 16: Caminos básicos del flujo.	67
Tabla 17 : Descripción de Caso de prueba de aceptación Mostrar resultado de la clasificación.	68
Tabla 18: Resultado de la técnica de selección aplicada.	69
Tabla 19: Resultados de la evaluación del conjunto de datos de entrenamiento para cada clase.	71
Tabla 20: Descripción de las precisiones por clase.	71
Tabla 21: Análisis de las reglas definidas a partir de la aplicación del algoritmo para el mejor criterio de cada clase.	72
Tabla 22: Resultados del análisis de la BC de 50 personas.	73
Tabla 23: Resultados arrojados por ambas técnicas.	74

ÍNDICE DE ILUSTRACIONES

Ilustración 1: Etapas para el proceso de extracción de conocimiento. Tomado de (Valentín, 2012).	17
Ilustración 2 Técnicas de la Minería de Datos. Adaptado de (García, 2006).	18
Ilustración 3: Características de los principales algoritmos de árboles de decisión. Tomado de (Lozano, 2013).	24
Ilustración 4: La definición del comportamiento en niveles. Adaptado de (Feist, 2009).	36
Ilustración 5: Rasgos de personalidad definidos por el FFM. Elaboración Propia.	36
Ilustración 6 : Algoritmo C4.5. Tomado de (Takeyas, 2005).	43
Ilustración 7: Diagrama de clases del diseño.	48
Ilustración 8: Aplicación del patrón MVC en el diagrama de clases de la solución.	49
Ilustración 9: Atributos de calidad evaluados en la métrica TOC.	53
Ilustración 10: Atributos de calidad evaluados en la métrica RC.	55
Ilustración 11: Diagrama de componente de la solución.	59
Ilustración 12: Estructura del fichero de datos .arff.	63
Ilustración 13: Interfaz para cargar fichero de datos.	64
Ilustración 14: Interfaz para realizar la selección de variables.	64
Ilustración 15: Interfaz para realizar la clasificación.	65
Ilustración 16: Segmento de código y grafo del flujo del método registrarVista ().	67
Ilustración 17: Resultados por cada una de las iteraciones de pruebas funcionales.	69
Ilustración 18: Rasgos relevantes a partir de la selección de atributos. Fuente: Elaboración propia.	70
Ilustración 19: Árbol generado para el índice de rendimiento IRHE.	72

INTRODUCCIÓN

Hasta hace poco las grandes empresas, condicionadas por el desarrollo acelerado de la ciencia y la tecnología, la organización por proyectos y la necesidad del perfeccionamiento en sus procesos; se preocupaban mucho más por el capital y la tecnología, al punto de considerarla como la combinación perfecta para alcanzar el éxito en el mercado. Muy poco importaba su mayor activo: los Recursos Humanos (RRHH), esa fuerza capaz de transformar los insumos en productos o servicios altamente competitivos que marcan la diferencia en un mercado cada vez más globalizado (Newstrom, 2000) (Salazar, 2010).

En la actualidad la mayoría de las organizaciones, centran su esfuerzo en la búsqueda del reconocimiento en la industria. Este reto ha llevado a focalizar el esfuerzo en contratar óptimos RRHH para incrementar la eficiencia y la productividad en sus procesos (Cardenas, 2012). En este sentido, la realidad actual impone la gestión adecuada del talento humano, considerando los procesos a que ellos son sometidos dentro de estas instituciones. Como es de esperar en este aspecto los directivos juegan un papel fundamental y en consecuencia, se les debe exigir una actuación permanentemente orientada a alcanzar y mantener la motivación, el interés, el involucramiento, la participación y el desempeño de los empleados.

El desempeño laboral constituye un elemento fundamental para el funcionamiento de cualquier organización, por lo que debe prestársele especial atención dentro del proceso de Administración de Recursos Humanos. La evaluación de dicho desempeño debe proporcionar beneficios a la organización y a las personas, en virtud de contribuir a la satisfacción de los empleados para garantizar el alcance de los objetivos institucionales (Pedraza, 2010).

Sin embargo, la mayoría de estas instituciones desconoce que un gran número de factores pueden influir en esta última exigencia. Los empleados pueden sentirse influenciados negativamente por la microgestión por parte de sus supervisores, por el cambio de labor o simplemente por otros factores tan graves y que pueden necesitar atención inmediata: los factores emocionales u otros específicos del aprendizaje como parte de su labor (Pedraza, 2010). Por el contrario, los empleados probablemente se sentirán inspirados e influenciados positivamente con equipos de alta calidad y un estilo de gestión del supervisor que sea accesible, sin importar el campo o la industria en la que trabajen (Pascarella, 1983).

Varias son las investigaciones centradas en descubrir cuáles factores inciden negativa o positivamente en el rendimiento de un individuo y en qué medida lo hace cada uno de ellos, a fin de implementar las

INTRODUCCIÓN

estrategias que incidan en la mejora del rendimiento. Chiavenato en su investigación “Administración de Recursos Humanos” plantea que el rendimiento está referido “...por una parte, a las capacidades y habilidades del individuo para realizar una tarea y por la otra, a las percepciones que él tiene del papel que debe desempeñar en la organización, indicando un enfoque sistémico” (Soto, 2006) (Chiavenato, 2000).

Autores como Tinto, Landeta, Ynzunza y López Gama señalan que el rendimiento laboral se condiciona por algunos aspectos psicológicos entorno a los antecedentes familiares y educativos; las características personales; los conocimientos y las funciones psicológicas básicas, como son la percepción, la memoria y la conceptualización; además del compromiso por alcanzar las metas (Tinto, 1989) (Landeta, 2011).

En España, Jano y Ortiz en el trabajo titulado “Determinación de los factores que afectan al rendimiento en las organizaciones”, han investigado los factores, llegando a la conclusión de que uno de ellos es el esfuerzo efectivo realizado por el individuo, así como las habilidades y conocimientos previos a su labor (Jano, 2005).

Algunos individuos pueden presentar problemas intelectuales, es decir, su capacidad intelectual es inferior a la media, lo que hace que el aprendizaje sea una labor aún más difícil, aunque paradójicamente también existe la posibilidad de que su capacidad intelectual sea superior a la media, es decir, que sea un individuo superdotado, en cuyo caso también se presentará el problema de rendimiento laboral (Psicológica, 2014). A pesar de que el resto de los factores propuestos UNIVERSIA² afectan directamente en el rendimiento laboral, existe una estrecha relación con el factor emocional condicionado por las habilidades, pensamientos y comportamientos de un individuo (Universia, 2014).

Cuando se hable de altos índices de rendimiento en el trabajo se alude al estado afectivo de agrado y altos conocimientos que experimenta un individuo al afrontar su realidad laboral, es decir, el conjunto de condiciones presentes en el ambiente de trabajo. Igualmente constituye la expresión emocional de la percepción y posee tanto elementos cognitivos como conductuales. Cabe destacar, que son variados los factores que determinan el rendimiento en el trabajo, destacándose entre otros, el trabajo mentalmente desafiante (que permitan el uso de las propias habilidades cognoscitivas y emocionales, apoyo de colegas y compañeros amigables, evaluaciones justas y un supervisor comprensivo, variedad de tareas e idoneidad en la selección, libertad y retroalimentación) (Bozal, 2000) (Chiavenato, 2000).

² Revista española que promueve el desarrollo social y empresarial a través del talento, el conocimiento, la investigación aplicada y la colaboración institucional entre universidades y empresas.

En el Centro de Informatización de la Gestión de Entidades (CEIGE) en la Universidad de las Ciencias Informáticas (UCI), un estudio realizado a 524 individuos seleccionados aleatoriamente en la herramienta GESPRO, agrupados en: estudiantes, profesores y especialistas de la rama informática, arrojó como resultados que el 83 % había sido evaluado con un índice de rendimiento en el trabajo bajo o medio en la labor realizada durante el período 2011-2013.

A raíz de los resultados obtenidos, se decidió aplicar una entrevista a 50 directivos del mismo centro: jefes de equipo, jefes de proyecto y jefes de departamento. El 21% de los entrevistados afirman que el rendimiento de las personas, entre otros factores, está condicionado mayormente por la experiencia, las competencias específicas en el rol que desempeñan, la motivación y los comportamientos específicos del individuo. Sin embargo, el 79% afirma que el rendimiento se basa solamente en los índices de productividad individual de los miembros de un equipo ante las tareas asignadas, sin importar los factores mencionados. Esta entrevista posibilitó, a través de sus resultados, determinar que existe diversidad de opiniones a la hora de evaluar el rendimiento de los subordinados y por ende, en el desconocimiento de la existencia de otros factores como los rasgos o atributos de personalidad, que inciden en la obtención de altos niveles en el rendimiento laboral.

A partir de lo anteriormente planteado se presenta el siguiente **problema científico**: ¿Cómo elevar el rendimiento de los RRHH de CEIGE a partir del análisis de la influencia que tienen los rasgos de personalidad en el desempeño laboral?

Se plantea como **objeto de estudio**: la Gestión de los RRHH y como **campo de acción**: el Proceso de evaluación del rendimiento laboral de los RRHH en los proyectos de desarrollo de software.

Para dar solución al problema planteado se define como **objetivo general**: Desarrollar una herramienta basada en algoritmos de clasificación supervisada para evaluar la influencia de los rasgos de personalidad en el desempeño laboral que contribuya a elevar el rendimiento de los RRHH del CEIGE.

Al logro del objetivo general tributan los siguientes **objetivos específicos**:

- Establecer el marco teórico referencial de la investigación a partir del estudio de herramientas y algoritmos de minería de datos bajo el enfoque de aprendizaje supervisado.
- Describir las funcionalidades de la herramienta basada en algoritmos de clasificación supervisada mediante las Historias de Usuarios.

- Realizar el análisis y diseño de la estructura de la herramienta para la aplicación de algoritmos de clasificación supervisada.
- Implementar las funcionalidades de la herramienta basada en algoritmos de clasificación supervisada.
- Validar la herramienta propuesta a partir de pruebas internas y métricas aplicables que permitan evaluar los resultados.

Para cumplir con los objetivos específicos se planificaron las siguientes **tareas de la investigación:**

- Estudio de fases del proceso de descubrimiento de conocimiento, principalmente la tarea de Minería de Datos: Clasificación.
- Estudio de técnicas de clasificación, particularmente algoritmos de árboles de decisión.
- Definición de los requisitos funcionales y no funcionales de la herramienta a implementar.
- Elaboración de los Diagramas de Clases del Diseño.
- Elaboración de los Diagramas de Componente.
- Implementación de las Historias de Usuario de la herramienta.
- Desarrollo de una estrategia de prueba sobre la herramienta.
- Análisis de las variables de la investigación.
- Análisis comparativo de los resultados obtenidos.

La investigación se sustenta en la siguiente **idea a defender:** Con el desarrollo de una herramienta basada en algoritmos de clasificación supervisada para evaluar la influencia de los rasgos de personalidad en el desempeño laboral, se logrará elevar el rendimiento de los RRHH del CEIGE.

Fueron identificadas como **variables de la investigación** las siguientes:

- **Variable independiente:** herramienta basada en algoritmos de clasificación supervisada.
- **Variables dependientes:** elevar el rendimiento de los RRHH del CEIGE.

Se espera como **posibles resultados:**

1. Análisis de la influencia de los rasgos de personalidad de los RRHH en el rendimiento de las personas en proyectos informáticos.
2. Herramienta basada en algoritmo de clasificación supervisado.

3. Validación de la herramienta y las variables de la investigación.

Para el cumplimiento de los objetivos se llevan a cabo los siguientes **métodos y técnicas** en la búsqueda y el procesamiento de la información:

Métodos Teóricos

Histórico-Lógico: El uso de este método permitió conocer y comprender el estado del arte de las técnicas de Minería de Datos y del uso de los modelos de personalidad en el mundo, cómo han evolucionado y desarrollo hasta la actualidad los métodos, modelos y herramientas relacionados con estos temas. Este estudio permitió determinar las tendencias actuales de las técnicas de Minería de Datos para la toma de decisiones como forma de evaluación del personal basándose en los rasgos de personalidad dentro del contexto organizacional de proyectos informáticos.

Analítico-sintético: Se aplicó para todas las fuentes de información: documentos, artículos, documentos normativos y metodológicos, trabajos científicos e investigaciones y literatura científica en general. Posteriormente se realizó un profundo análisis de los procesos relacionados con la evaluación del rendimiento del personal de manera individual en proyectos de software, y todas las partes que lo conforman, 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.

Métodos Empíricos

Entrevista: Con el fin de determinar la dinámica de asignación de tareas a los equipos y así poder conocer el tipo de personalidad dentro del personal de los proyectos, se realizó este método científico de investigación a líderes y personal en CEIGE y así obtener información acerca de la situación problemática que afecta el rendimiento competente de los Recursos Humanos de este centro.

Encuesta: Se utiliza la encuesta para conocer las opiniones y valoraciones de los expertos sobre los rasgos de personalidad relevantes en el rendimiento laboral. Se utiliza también el instrumento NEO-PI del modelo Cinco Factores de Personalidad, instrumento aplicado para evaluar los rasgos de personalidad de los individuos del CEIGE. Además se emplea el método de evaluación por criterio de expertos para validar teóricamente el resultado alcanzado en la investigación.

Experimento: Se utiliza el pre-experimento pre y post prueba con datos provenientes de situaciones reales obtenidos en el GESPRO y el instrumento NEO-PI. Se parte de una primera identificación de los rasgos de personalidad relevantes a partir de la clasificación del conjunto de datos con el algoritmo utilizado en la investigación y luego se obtiene una identificación usando el criterio de expertos. Estas observaciones son analizadas en función de determinar la influencia de los rasgos de personalidad en el rendimiento laboral y determinar los rasgos que permiten elevar el rendimiento de los Recursos Humanos del CEIGE.

Muestreo

La población está constituida por 525 individuos que pertenecieron al CEIGE durante el período 2011-2013. Se utilizará una muestra probabilística con una técnica de muestreo aleatorio simple. Los individuos que conformarán la muestra serán seleccionados al azar ya que todos los individuos tienen la misma probabilidad de ser elegidos. La muestra está conformada por 150 personas lo que representa el 29% de los individuos de la población.

Se define la siguiente estructura del documento, quedando conformado en tres capítulos:

✓ **Capítulo 1: Fundamentación teórica.**

En el primer capítulo se hace el análisis del estado del arte y se contextualizan conceptos sobre los algoritmos de clasificación de Minería de Datos, el aprendizaje automatizado y el rendimiento laboral de los estudiantes en proyectos de desarrollo de software. Se evalúan las ventajas y desventajas que presentan los diferentes algoritmos de árboles de decisión estudiados a partir de datos numéricos y simbólicos, detallando algunos aspectos que permitirán su elección para la investigación. Se realiza una síntesis en relación a los indicadores de rendimiento, el algoritmo a utilizar así como los rasgos de personalidad. Se estudiarán los modelos de personalidad aplicados a los RRHH, así como los instrumentos utilizados en cada uno de ellos para definir el modelo a utilizar en la investigación.

✓ **Capítulo 2: Propuesta de solución.**

En este capítulo se realizará la propuesta de aplicación del algoritmo en el análisis de los rasgos de personalidad. Se presenta una breve descripción del algoritmo implementado en la herramienta, haciendo mención de sus principales características y el pseudocódigo. Se reflejan las características y funcionalidades de la herramienta mediante las Historias de Usuario a partir de los requisitos funcionales y no funcionales identificados. Se presentan los diferentes artefactos que se obtienen del diseño realizado; también se detallará la implementación desarrollada a partir del diagrama de clases, diagrama de componentes como parte de la descripción del diseño e implementación realizado.

✓ **Capítulo 3: Resultados y validación de la propuesta.**

En este capítulo se validarán las variables del problema de la investigación, aplicando un conjunto de métricas, atendiendo a dimensiones, indicadores y unidad de medida. Se realizan distintos niveles y tipos de pruebas a la herramienta para comprobar las funcionalidades de la misma así como la calidad de su implementación. Por último se estudian los resultados obtenidos realizando la corrida del algoritmo con la base de casos confeccionada.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Introducción

En el presente capítulo se realizará un análisis de las principales fuentes bibliográficas para obtener la información actualizada acerca de los componentes utilizados en la realización de la investigación. Primeramente se abordan los conceptos relacionados con el descubrimiento del conocimiento y la Minería de Datos para enmarcar la temática en estudio y se discuten varios enfoques, ventajas y desventajas de las técnicas de clasificación. Se estudian además los algoritmos de clasificación como paso necesario para la resolución de complejos problemas de aprendizaje y clasificación. Se describe el marco teórico de las principales tendencias modernas tanto en el rendimiento laboral, como de su evaluación a partir de las definiciones del término y finalmente se realiza un análisis de los modelos más utilizados para evaluar los rasgos de personalidad.

1.1. Descubrimiento de conocimiento

Hoy en día, la cantidad de información que han sido almacenados 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 KDD³, 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, 2012). Según García es definido como “el proceso no trivial de identificación en los datos de patrones válidos, nuevos, potencialmente útiles, y finalmente comprensibles”. Las tareas comunes en KDD son la inducción de reglas, los problemas de clasificación y clustering, el reconocimiento de patrones, el modelado predictivo, la detección de dependencias, entre otras (García, 2006).

Este proceso consta de una secuencia iterativa de etapas o fases que se relacionan y describen a continuación (Ver Ilustración 1):

³ Knowledge Discovery in Database (descubrimiento de conocimiento en bases de datos, KDD por sus siglas en inglés)

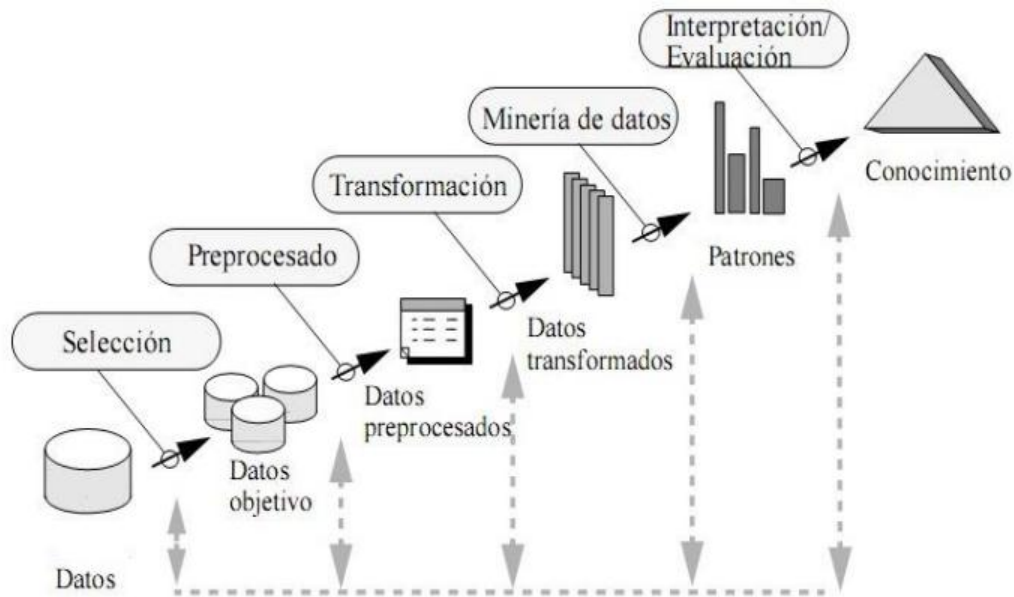


Ilustración 1: Etapas para el proceso de extracción de conocimiento. Tomado de (Valentín, 2012).

Selección: tiene como objetivo la preparación de las fuentes de datos y la selección de las mismas.

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

Técnicas de Minería de Datos: 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: se procede al análisis de los resultados descubiertos. Incluye a su vez la resolución de posibles inconsistencias con otros conocimientos anteriores a la investigación.

1.2. Minería de Datos

La Minería de Datos (DM) es considerada como 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 Datos (Gómez, 2003). La DM constituye un campo multidisciplinario al tener involucrado otras disciplinas como: la recuperación de información, el análisis de los datos, la extracción de información, el clustering, la categorización, la visualización, el almacenaje de datos y los sistemas de aprendizaje (Díaz, 2007).

Actualmente la DM puede ser útil para los departamentos de RRHH 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 éstos. El impacto de KDD y DM para las empresas puede ser muy amplio, ya que su utilidad potencial depende de los resultados de dicho análisis (Saldaña, 2005).

La DM 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, 2012).

1.2.1. Técnicas de Minería de Datos

Las técnicas de la DM (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 DM, 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, 2006).

A continuación en la Ilustración 2 se muestra un gráfico con las principales técnicas de DM que se clasifican en dos grandes categorías: supervisadas o predictivas y no supervisadas o descriptivas:



Ilustración 2 Técnicas de la Minería de Datos. Adaptado de (García, 2006).

Una técnica constituye el enfoque conceptual para extraer la información de los datos y en general 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.

1.3. Clasificación

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, 2006).

1.3.1. Técnicas de clasificación

A continuación se detalla y explican algunas de las técnicas de clasificación que permiten analizar un conjunto de datos y ayudar a la toma de decisiones.

A. 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. Si hay dos ejemplos con exactamente los mismos pares *atributo-valor* para todos los atributos del subconjunto, la clase que se elija será la media de los ejemplos (en el caso de una clase numérica) o la que mayor probabilidad de aparición tenga (en el caso de una clase simbólica) (Martineaux, 2008).

B. 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 y categorizar 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, 2006). Dado un conjunto de ejemplos, estructurados como vectores de pares ordenados atributo-valor, de acuerdo con el formato general en el aprendizaje inductivo a partir de ejemplos, el concepto que estos sistemas adquieren durante el proceso de aprendizaje consiste en un árbol. 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 (Jacas, 2009).

C. 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. En los casos en que la información de entrada posee algún tipo de "ruido" o defecto (insuficientes atributos o datos, atributos irrelevantes o errores u omisiones en los datos) estas técnicas pueden habilitar métodos estadísticos de tipo probabilístico para generar árboles de decisión recortados o podados. También en estos casos pueden identificar los atributos irrelevantes, la falta de atributos discriminantes o detectar "gaps" o huecos de conocimiento. Esta técnica suele llevar asociada una alta interacción con el analista de forma que éste pueda intervenir en cada paso de la construcción de las reglas, bien para aceptarlas o bien para modificarlas (García, 2006).

A partir de un análisis de las técnicas de clasificación de acuerdo a sus características antes abordadas, se decide utilizar los árboles de decisión ya que es una excelente ayuda para la elección entre varios cursos de acción, representando la dependencia lógica entre la decisión a tomar y los atributos considerados (Ponte, 2013). 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 fácil de interpretar por los seres humanos. Al compararse con las tablas de decisión los árboles se entienden con más facilidad en una organización y son apropiadas como un método de comunicación. Las condiciones y las acciones del árbol de decisión se encuentran en ciertas ramas pero no en otras, a diferencia de las tablas de decisión, donde todas forman parte de la misma tabla (Descuadrando, 2013). 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.

En general, los clasificadores de árbol de decisión tienen buena precisión y han sido utilizados para la clasificación en muchas áreas de aplicación, tales como la medicina, producción, análisis financiero, la astronomía, análisis de los RRHH y la ayuda en la toma de decisiones (Magazine, 2014). Además, los árboles de decisión son un método efectivo para la toma de decisiones por varias razones: en el problema que plantea todas las posibilidades son analizadas, permite analizar totalmente el resultado para cada una de las posibles opciones y ayuda a realizar las mejores decisiones con base a la información existente y a las mejores suposiciones.

1.3.2. Selección de las variables

La selección de variables (SV) engloba a todo un conjunto de técnicas de reducción de la dimensionalidad⁴ de los datos a procesar. Los problemas de SV han adquirido especial interés en diversas áreas de aplicación donde el número de variables para un problema de clasificación es elevado y muchas de ellas son redundantes o irrelevantes. La meta de la SV es encontrar un subconjunto "óptimo" de variables que minimice la pérdida de información, maximice la reducción de ruido y que debe ser más significativa para la clasificación, para dar así una mejor interpretación a los resultados. La estrategia para la SV más común consiste en evaluar cada variable individualmente y seleccionar aquellas variables que aportan información de mayor calidad. El paradigma de la selección de variables trata de cubrir tres aspectos: mejorar el error de predicción de los clasificadores, proporcionar clasificadores más rápidos y eficientes y facilitar la comprensión del proceso subyacente Generador de los Datos (Luján, 2009).

Cualquier procedimiento para la SV basa su funcionamiento en dos aspectos fundamentales: un criterio de selección y un procedimiento de búsqueda que proporcione buenos resultados. Existen diversos criterios de selección y estos se clasifican en: Filtros, que utilizan características generales de los datos para evaluar los atributos y funcionan de forma independiente de cualquier algoritmo de aprendizaje, dentro de estos se encuentran la selección de atributos teniendo en cuenta el ranking de importancia de estos; y Wrapper los cuales usan el desempeño de algún clasificador (algoritmo de aprendizaje) para determinar lo esperado de un subconjunto (Hernández, 2007). Ejemplos de estos criterios de selección son:

- **Selección de atributos basada en la correlación (CFS⁵):** evalúa un subconjunto de atributos considerando la habilidad predictiva individual de cada variable, así como el grado de redundancia

⁴ Tamaño del conjunto de datos.

⁵ Correlation-based Feature Selection,

entre ellas. Se prefieren los subconjuntos de atributos que estén altamente correlacionados con la clase y tengan baja intercorrelación (Hall, 1998).

- **Evaluación de Conjunto basada en la Consistencia (CSE⁶):** estos métodos buscan combinaciones de atributos cuyos valores dividen los datos en subconjuntos que contienen una gran mayoría de clase única. Por lo general, la búsqueda está sesgada hacia los pequeños subconjuntos de características de clase alta consistencia (Hall, 1998).
- Evaluación de Conjunto en Envoltura (WSE⁷):** evalúa los subconjuntos de atributos utilizando un clasificador. Emplea validación cruzada para estimar la exactitud del esquema de aprendizaje en cada conjunto (Kohavi R, 1997).

El criterio de selección utilizado en la investigación es CFS, de tal forma que se obtenga un subconjunto de rasgos de personalidad altamente correlacionadas con los índices de rendimiento que en este caso son los atributos clases, intentando mantener un bajo nivel de correlación lineal entre ellos. Desde el punto de vista del diseño del clasificador, un algoritmo de selección de variables forma parte del proceso de clasificación, donde una vez aplicado la selección de variables continúa la aplicación de un clasificador estándar sobre el subconjunto escogido (Luján, 2009).

1.3.3. Algoritmos de clasificación de árboles de decisión

Para comprender el ámbito de los árboles de decisión a partir de lo antes descrito 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. Los árboles 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. Seguidamente se realiza la descripción de algunos de ellos:

A. 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

⁶ Consistency-based Subset Evaluation.

⁷ Wrapper Subset Evaluation.

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 (García, 2006).

B. 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 entropía⁸. 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 (García, 2006).

C. J4.8:

Es una implementación libre del C4.5 y está integrada en la plataforma WEKA⁹. García plantea en su investigación que es uno de los algoritmos de Minería de Datos más utilizados en multitud de aplicaciones. El parámetro más importante a tener en cuenta es el factor de confianza para la poda (confidence level), 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 es de 25%, según baje este valor se permiten más operaciones de poda (García, 2006).

Teniendo en cuenta las características principales de los algoritmos, se realizó un estudio más profundo para 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 (Ver Ilustración 3).

⁸ Se trata de una medida que permite calcular el grado de incertidumbre de una muestra. Una muestra completamente homogénea tiene entropía 0.

⁹ Herramienta de minería de datos de libre distribución.

Algoritmo	Variables predictoras	Tipo de división	Criterio de División	Casos <i>missing</i>	Método de poda	Implementación
CART (1984)	Continuas/ Discretas	Binaria	Impureza (<i>Gini index</i>)	SI	Post-	Libre Comercial
ID3 (1979)	Discretas	<i>n</i> -aria	Ganancia de información (Entropía)	NO	NO	Comercial
C4.5 (1993)	Continuas/ Discretas	Binaria/ <i>n</i> -aria	<i>Gain ratio</i> (Entropía)	SI	Pre-/ Post-	Libre Comercial
J4.8	Continuas/ Discretas	Binaria/ <i>n</i> -aria	<i>Gain ratio</i> (Entropía)	SI	Pre-/ Post-	Libre (Weka)
C5.0	Continuas/ Discretas	Binaria/ <i>n</i> -aria	<i>Gain ratio</i> (Entropía)	SI	Pre-/ Post-	Comercial
CHAID (1975)	Discretas	<i>n</i> -aria	χ^2	SI	Pre- (nivel de significancia)	Comercial

Ilustración 3: Características de los principales algoritmos de árboles de decisión. Tomado de (Lozano, 2013).

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. Pues es un algoritmo como los demás estudiados que 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, porque trata de enfocarse explícitamente hacia los atributos relevantes y de ignorar los irrelevantes, además es relativamente rápido y muestra fiabilidad en los resultados. Representa uno de los algoritmos de aprendizaje más comúnmente usados en aplicaciones de Minería de Datos (García, 2006) (Jacas, 2009).

1.4. Herramientas usadas en el proceso de Minería de Datos

El desarrollo de sistemas de DM ha evolucionado notablemente en los últimos años. Un sistema de DM es definido como una aplicación integrada diseñada para ejecutar varios algoritmos de DM y otras operaciones del proceso de KDD sobre un conjunto de datos bien gestionados por el propio sistema o accesibles de diversas fuentes de datos externas (Sánchez, 2001).

Para comprender el ámbito de la solución propuesta se estudiaron varios software comerciales y académicos que permitieran realizar tareas de DM. Dentro de las herramientas analizadas se encuentran:

KNIME¹⁰, RapidMiner¹¹, WEKA¹² y SAS Enterprise Miner. Seguidamente se realiza la caracterización de cada uno de ellas.

KNIME

Es una plataforma de fácil uso y comprensible para integración de datos, procesamiento, análisis y exploración. Es un sistema totalmente gratuito para el desarrollo y ejecución de técnicas de DM. KNIME fue desarrollado originalmente en el departamento de bioinformática y Minería de Datos de la Universidad de Constanza, Alemania. Permite la creación de modelos estadísticos y de DM, como árboles de decisión y regresiones. La API¹³ del software permite el montaje fácil y rápido de nodos para procesamiento de datos (ETL¹⁴), para el análisis de datos, modelado y visualización. La visualización de los datos, los resultados y los procesos en KNIME pretende ser simple para los usuarios, por ejemplo, mediante la integración con R¹⁵ es posible mejorar la visualización de funciones estadísticas y resultados. Esta herramienta está implementada en Java, basada en Eclipse y hace uso de sus métodos de extensión para soportar plugins proporcionando así una funcionalidad adicional. A través de plugins, los usuarios pueden añadir módulos de texto, imágenes, procesamiento de series de tiempo y la integración de varios proyectos de código abierto, tales como el lenguaje de programación R, WEKA, el kit de desarrollo de Química y LIBSVM (KNIME, 2013)(Berzal, 2013).

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 DM. La aplicación de RapidMiner cubre un amplio rango dentro de la DM y es utilizado tanto en investigación como en tareas de día a día por diferentes empresas. Tiene soporte API completo, lo que hace que sea posible acceder a una amplia variedad de funcionalidades y soporte. Además de ser una herramienta flexible para aprender y explorar la DM, la interfaz gráfica de usuario tiene como objetivo simplificar el uso para las tareas complejas de esta área. Es un software de tipo Open-Source con licencia GNU¹⁶, trabaja bajo las plataformas Windows y

¹⁰ Konstanz Information Miner.

¹¹ Anteriormente conocido como YALE, Yet Another Learning Environment.

¹² Waikato Environment for Knowledge Analysis.

¹³ Interfaz de programación para añadir nuevos algoritmos al sistema.

¹⁴ extracción, transformación, carga.

¹⁵ Paquete de Software Estadístico.

¹⁶ General Public License.

Linux, y usa el lenguaje de scripting XML para describir los operadores y su configuración (RapidMiner, 2014).

SAS Enterprise Miner

Es un sistema que proporciona el SAS Institute para tareas de DM que permite transformar todos los datos de la organización en conocimiento, reduciendo la incertidumbre, realizando predicciones fiables y optimizando el desempeño. Posee una arquitectura distribuida con una interfaz gráfica de usuario potente, eficaz y fácil de usar. Entre las tareas que realiza la herramienta están: pre-procesamiento de los datos, modelos matemáticos, evaluación para comprobar la eficiencia y eficacia de la herramienta, y la visualización de resultados. Ofrece un conjunto de algoritmos de modelado predictivo y descriptivo, incluyendo árboles de decisión, incremento de gradientes, redes neuronales, regresión lineal y logística y regresión de mínimos cuadrados parciales, entre muchos otros. Además presenta un proceso de calificación automatizado, preparación, indexación y exploración de datos y un método rápido, fácil y autosuficiente de generación de modelos para usuarios empresariales (García, 2006).

WEKA

Es una plataforma de software libre desarrollado en Java la cual se centra en el aprendizaje automático y en la DM. La funcionalidad de su API 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 DM, 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 DM. Además, es independiente de la arquitectura, ya que funciona en cualquier plataforma sobre la que haya una máquina virtual Java disponible (WEKA, 2013) (Morate, 2013).

A partir del estudio de las herramientas anteriores DM se realizó un análisis comparativo teniendo en cuenta aspectos como: si la licencia es libre o no, si requiere de conocimientos para la interpretación de los resultados, si es multiplataforma, si contiene modelo de clasificación, si implementa árboles de

decisión, si implementa el algoritmo J4.8 (implementación del C4.5), si presenta una interfaz amigable y si dispone o no de un API (Ver Tabla 1).

Tabla 1: Análisis comparativo de sistemas de minería de datos. Fuente: Elaboración Propia.

Sistemas	Licencia libre	Conocimientos sobre DM	Multiplataforma	Modelo de Clasificación	Implementa árboles de decisión	J4.8	Interfaz amigable	API
KNIME	Sí	Sí	Sí	Sí	Sí	Sí	Sí	Sí
RapidMiner	Sí	Sí	Sí	Sí	Sí	Sí	Sí	Sí
SAS	No	Sí	-	Sí	Sí	No	Sí	Sí
WEKA	Sí	Sí	Sí	Sí	Sí	Si	Sí	Sí

En esta tabla comparativa se puede observar que aunque SAS Enterprise Miner constituye una aplicación líder en el mercado, presentan la desventaja de ser herramientas comerciales, y su adquisición puede ser altamente costosa, además no implementan el algoritmo J4.8 de utilidad en la investigación. Por otro lado, tanto RapidMiner, KNIME como WEKA son aplicaciones de código abierto y de libre distribución, implementan el algoritmo a utilizar y son multiplataforma. La mayoría de estas herramientas son de dominio público, un tanto populares por su entorno gráfico integrado y otra serie de características significativas. Uno de los problemas que los usuarios podrían encontrar en estas herramientas es que las funciones de visualización no son muy avanzadas, a pesar de que presentan una interfaz amigable y entendible. Además, los gráficos que se producen en dichas herramientas probablemente necesitarán ser cambiados y hechos más visualmente atractivos, sobre todo si el público objetivo son las personas que no presentan conocimientos sobre la DM o el análisis estadístico.

El estudio de estas herramientas sirvió de base para identificar las principales funcionalidades con la que pudiera contar la herramienta a desarrollar, mostrando resultados entendibles por usuarios con conocimientos básicos de DM y una interfaz amigable. Teniendo en cuenta las potencialidades de la herramienta WEKA respecto a las demás se utilizará como base para el desarrollo de la investigación, ya que a pesar de los inconvenientes anteriormente descritos, presenta a su favor muchas ventajas.

1.5. Rendimiento laboral

El rendimiento de los empleados es la piedra angular para desarrollar la efectividad y el éxito de una compañía, por esta razón hay un constante interés de las empresas por mejorar el rendimiento de los empleados a través de continuos programas de capacitación y desarrollo.

En la UCI, el rendimiento laboral de los proyectos productivos se define como el resultado de las actividades planificadas por este, en cuanto al estado de conclusión de los productos entregables, y sobre aquello que se ha realizado. Algunos autores como Cordero y Chiavenato en este entorno coinciden en que el rendimiento laboral es definido como los comportamientos de los empleados frente al trabajo, siendo imprescindible dentro de la gestión del personal por lo que es necesario medirlo, analizarlo, predecirlo y modificarlo; contribuyendo además a la estrategia individual para lograr los objetivos de la organización. Es por esto que la manera más usada hoy día para estimar o apreciar el comportamiento de una persona en un centro laboral, es la evaluación del rendimiento (Cordero, 2008) (Chiavenato, 2000).

1.5.1. Evaluación del rendimiento laboral

La evaluación del rendimiento, es un proceso destinado a determinar y comunicar a los empleados, la forma en que están desempeñando su trabajo y a elaborar planes de mejora, siendo importante que cuando se evalúa al personal no solo se haga saber su nivel de cumplimiento, sino que debe influir en su nivel futuro de esfuerzo y en el rendimiento correcto de sus tareas (Byars, 1996).

Según Mondy y Noe en su investigación sobre la administración de recursos humanos, la evaluación del rendimiento “es un sistema formal de revisión y evaluación del rendimiento laboral individual o de equipos”. Aunque la evaluación del rendimiento de equipos es fundamental cuando estos existen en una organización, el enfoque de la evaluación en la mayoría de las empresas se centra en el empleado individual. Sin importar el énfasis, un sistema de evaluación eficaz evalúa los logros e inicia planes de desarrollo, metas y objetivos (Mondy, 2005).

Dentro de los objetivos de la evaluación del rendimiento en la UCI está la toma de decisiones de RRHH con fundamento, la mejora del rendimiento de los empleados, la comprobación de la validez de otras prácticas de RRHH y ser instrumento para comunicar normas y valores de la organización. Por lo que la evaluación del rendimiento sería un proceso para obtener un índice cuantitativo que muestre el nivel de desarrollo que tiene un individuo en uno o varios factores de rendimiento. Esta evaluación se realiza teniendo en cuenta un conjunto de indicadores de rendimiento, los cuales son medidas que describen cuán bien se están desarrollando los objetivos de un programa, un proyecto y/o la gestión de una institución, a qué costo y con qué nivel de calidad (Rodríguez, 2011).

1.5.2. Fuente de datos: GESPRO

Como parte de la decisión de estandarizar los sistemas de control y seguimiento de la Red de Centros de desarrollo de software de la UCI, se decide desarrollar el GESPRO como solución única e integral para la gestión de sus proyectos (Piñero Pérez, 2010). Basado en Redmine¹⁷ y otras herramientas libres construidas en la propia organización, el sistema está alineado con los estándares propuestos por el Instituto de Gestión de Proyectos (PMI¹⁸) y el Instituto de Ingeniería de Software (SEI¹⁹), mostrando las interfaces adecuadas a los usuarios para que introduzcan al sistema los datos (PMI, 2009) (SEI, 2010).

GESPRO en su versión 12.05 posibilita la automatización del cálculo de indicadores para realizar la evaluación del rendimiento laboral, dentro de estos se encuentra los indicadores propuestos para el área de RRHH, que van a ser de gran importancia para la presente investigación (Lugo, 2012). Debido a lo anteriormente mencionado para la confección de la base de caso a utilizar, se concebirá necesario utilizar esta herramienta como fuente de datos porque proporciona la información necesaria para el cálculo de los indicadores de rendimiento como: las tareas a ejecutar, sus porcentajes de ejecución, los recursos humanos y materiales asociados, así como otros datos relacionados.

1.5.3. Indicadores específicos del rendimiento laboral

El rendimiento refleja principalmente los objetivos de la eficiencia: alcanzar las metas al tiempo que se utilizan los recursos eficientemente (Vega, 2010). Para medir si se alcanzaron o no los objetivos propuestos es necesario evaluar el rendimiento mediante indicadores cuantitativos y cualitativos que reflejen el estado o progreso de la empresa, unidad o individuo.

Lugo en su estudio aborda un conjunto de indicadores para la evaluación de los centros productivos a través de la herramienta GESPRO. En la presente investigación se utilizarán estos indicadores para calcular el rendimiento de los RRHH del CEIGE en cuanto a las tareas asignadas y su desarrollo. El cálculo de los cuatro sub-indicadores que permiten obtener el Índice de Rendimiento de los Recursos Humanos (IRRH) se realiza como se muestra a continuación (Lugo, 2012):

- **Índice de Correlación del Recursos Humanos con respecto al Trabajo (IRHT):** correlación entre la estandarización de los tiempos estimados (TE_{Tarea}) y los tiempos dedicados (TD_{Tarea}) de las tareas involucradas con el Recursos Humanos. Adquiere valores entre -1 y 1.

¹⁷ Aplicación web flexible de gestión de proyectos, utiliza el framework Ruby on Rails y es multiplataforma.

¹⁸ Project Management Institute, por sus siglas en inglés.

¹⁹ Software Engineering Institute, por sus siglas en inglés.

Interpretación: cercano a 1, mejor. Se calcula según la **Ecuación 1**:

$$IRHT = CORR \frac{TE_Tarea - AVG(TE_Tarea)}{STDV(TE_Tarea)}, \frac{TD_Tarea - AVG(TD_Tarea)}{STDV(TD_Tarea)}$$

- **Índice de Rendimiento del Recursos Humanos con respecto a la Eficacia (IRHE)**: relación entre la cantidad de tareas cerradas evaluadas de Bien o Excelente (*CTBERH*) y el total de tareas cerradas del proyecto (*CTC*) involucradas con el Recursos Humanos hasta la fecha de corte dd/mm/aaaa. Adquiere valores entre 0 y 1.

Interpretación: cercano a 1, mejor. Se calcula según la **Ecuación 2**:

$$IRHE(dd/mm/aaaa) = \frac{CTBERH}{CTC}$$

Donde:

$$CTBERH = 3 * CTCPA + 2 * CTCPN + CTCPB$$

$$CTC = 3 * CTCPA + 2 * CTCPN + CTCPB$$

CTCPABE: cantidad de Tareas Cerradas con Prioridad Alta evaluadas de Bien o Excelente.

CTCPNBE: cantidad de Tareas Cerradas con Prioridad Normal evaluadas de Bien o Excelente.

CTCPBBE: cantidad de Tareas Cerradas con Prioridad Baja evaluadas de Bien o Excelente.

CTCPA: cantidad de Tareas Cerradas con Prioridad Alta.

CTCPN: cantidad de Tareas Cerradas con Prioridad Normal.

CTCPB: cantidad de Tareas Cerradas con Prioridad Baja.

- **Índice de Rendimiento del Recursos Humanos con respecto a su Aprovechamiento (IRHA)**: relación entre los totales de tiempo planificado (*TTP*) y disponible (*TTD*) asignados al Recursos Humanos hasta la fecha de corte dd/mm/aaaa.

Interpretación: IRHA < 0.75, sub-utilizado; 0.75 <= IRHA <=1, aprovechado; IRHA > 1, sobrecargado. Se calcula según la **Ecuación 3**:

$$IRHA(dd/mm/aaaa) = \frac{TTP}{TTD}$$

- **Índice de Rendimiento de los Recursos Humanos con respecto a la Eficiencia (IRHF):** muestra el estado de avance o progreso del Recursos Humanos en la realización de las tareas que tiene asignadas. Relación entre la sumatoria de los por cientos de ejecución real y la sumatoria de los por cientos de ejecución planificado para cada una de sus tareas según la fecha de corte dd/mm/aaaa.

Interpretación: IRHF < 1, atraso; IRHF = 0, atraso por no haber comenzado; IRHF =1, en tiempo; IRHF > 1, adelantado. Se calcula según la **Ecuación 4**:

$$IRHF(dd/mm/aaaa) = \frac{\sum_{i=1}^n Ejec_Real_Tarea_RH(i)}{\sum_{i=1}^n Ejec_Planif_Tarea_RH(i)}$$

Donde: n es la cantidad de tareas asignadas al Recursos Humanos, desde el comienzo del proyecto hasta la fecha de corte.

1.6. Estudio de la personalidad en los RRHH

El estudio de la personalidad generalmente se ha centrado en el análisis de la personalidad de los individuos, pero las consideraciones evolutivas y los cambios o estabilidades que se producen a lo largo del tiempo o desarrollo del individuo ha sido un tema pendiente en esta área. El estudio de la personalidad humana Allport define que "la personalidad es la organización dinámica, interna al individuo, de aquellos sistemas psicofísicos que determinan su ajuste único a la situación" (Espinoza, 2014).

Existen hoy en día un conjunto de modelos de personalidad que sirven para descubrir los tipos de personalidad y las diferencias individuales, así como los que tratan de agrupar las características de la personalidad en rasgos o dimensiones para definir y diferenciar personalidades distintas. Cada uno de estos modelos utiliza un instrumento para evaluar y describir la personalidad de los individuos.

1.6.1. Modelos de Personalidad

Actualmente existen varias investigaciones relacionadas con los modelos de personalidad de los RRHH, en particular cada una de las diferentes teorías, modelos de la personalidad y la motivación humana ofrecen una perspectiva diferente de un área de gran complejidad que abarca la personalidad, la motivación y el comportamiento de los individuos (Stiven, 2012). Entre los modelos más utilizados en las investigaciones sobre evaluación del tipo de personalidad se identifican el Modelo de Myers-Briggs y el Modelo Cinco Factores de la Personalidad descritos a continuación:

A. Modelo de Cinco Factores de la Personalidad

Los defensores de este modelo más conocido por sus siglas en inglés Five-Factor Model (FFM), postulan que cualquier tipo de personalidad, normal o anormal, puede ser conceptualizada a partir de cinco amplias dimensiones ortogonales con que se podría describir cómo es una persona: Neurotismo, Extraversión, Apertura a la Experiencia, Amabilidad y Sentido de Responsabilidad (Lingjaerde, 2001) (Pérez, 2004). Cada una de estas cinco dimensiones se categoriza a partir de la evaluación de un grupo de rasgos que explican la mayor parte de las diferencias entre las personalidades individuales, a pesar de las barreras del lenguaje o la cultura (Ver Anexo 1) (Feldt, 2010).

Cada modelo de personalidad debe utilizar test o instrumentos para medir el estudio de la personalidad, en el caso del modelo FFM no es propio de un instrumento en particular, en realidad es una descripción y explicación de la personalidad que posee diversos inventarios contruidos para su evaluación. Dentro de los instrumentos o test que existen para medir el modelo FFM se encuentran: Cuestionario Big Five de Caprara, Babaranelli y Borgogni, el Big Five Questionnaire (BFQ), el Inventario de Personalidad NEO-PI (The Personality Inventory NEO) y el Sixteen Personality Factor Inventory (16PF). Dentro de este grupo el más reconocido e investigado es el NEO-PI, ya que además de medir esas cinco dimensiones, analiza también seis de los rasgos específicos de cada factor (Herreras, 2005).

Este test desarrollado por McRae y Costa, presenta adecuados niveles de confiabilidad y posee también estudios que avalan su utilidad práctica y sus cimientos teóricos. El NEO-PI entrega una descripción simple de la personalidad, destacando aspectos sobresalientes; y una descripción detallada y compleja que refiere a procesos cognitivos, bienestar psicológico, motivos, etc (Criado, 2012).

El modelo se aplica, principalmente, en tareas de selección de personal y en otros casos se utiliza para evaluar las competencias genéricas de los empleados (Aluja, 2002). Es un modelo relativamente realizable en las instituciones debido a que muchas personas están calificadas para administrarlo y es fácil

de entender. Ha ayudado a millones de personas en todo el mundo a obtener conocimientos acerca de sí mismos y cómo interactuar con los demás y mejorar la forma de comunicarse, aprender y trabajar. Es un modelo libre y hay muchos libros de apoyo, sitios web y otras fuentes útiles que están disponibles para el público en general. En círculos científicos, el FFM es por ahora el modelo de personalidad más aceptado y más usado (aunque aún existen otros sistemas psicológicos también muy utilizados tales como el “MBTI”) (Wordpress, 2012).

B. Modelo de Myers-Briggs

El modelo de Myers-Briggs (MBTI) es utilizado en el estudio de la estructuración cognitiva y las características de la personalidad en el ámbito profesional. Este modelo presenta cuatro dimensiones diferentes de las preferencias humanas que se miden en factores bipolares (Seddigi, 2009) (Stiven, 2012):

- **La orientación de la persona:** se refiere a cómo las personas se relacionan con el mundo. Es Extrovertido (E) quien tiende a preferir el intercambio con otras. En el extremo opuesto Introverso (I) señala a quien se siente más cómodo con el mundo de las ideas.
- **La percepción:** se refiere a cómo las personas capturan la información. El factor Intuición (N) caracteriza a personas que buscan por relaciones y patrones en la información; mientras que el extremo opuesto de los Sentidos (S) se prefieren hechos y experiencias para la búsqueda de información.
- **Cómo las personas toman decisiones:** el factor Pensamiento (T) se refiere a cuando la persona basa sus decisiones en el análisis y la lógica. En el otro extremo de esta dimensión se encuentra el factor Sentimiento (F), que caracteriza a la persona que se basa en las emociones y valores personales.
- **Estilo de vida:** se refiere a cómo la persona trabaja o se organiza. El Juicio (J) define a aquellos que prefieren la planeación y el orden; mientras el factor Percepción (P) define quién prefiere un estilo de vida más espontáneo y flexible.

El instrumento de medición o test empleado en este modelo es el definido por las mismas autoras: el Indicador de Tipo de Myers-Briggs (MBTI), siendo el más empleado en las dimensiones especificadas ya que analiza la composición de la personalidad de equipos de trabajo. Este indicador es un inventario fiable y validado de evaluación de la personalidad que identifica la orientación de un individuo de la energía y sus funciones mentales básicas (Adele, 2013).

En investigaciones el modelo ha sido utilizado fundamentalmente para identificar: el tipo o el patrón que predomina en el personal del software, los tipos más adecuados para desempeñar tareas y roles en el proceso de desarrollo, para evaluar el impacto de la diversidad de tipos en el equipo y de las distintas dimensiones en los resultados del proyecto. Es un modelo relativamente fácil de aplicar en las instituciones, pero está sujeto a derechos de autor (Stiven, 2012).

En la Tabla 3 se realiza un análisis comparativo de los modelos y test psicológicos estudiados teniendo en cuenta aspectos tales como: si son modelos libres (LIB); facilidad de aplicación teniendo en cuenta si requiere o no personal especializado para el análisis de los datos (ESPEC) y si está disponible en línea (ACCESO); la existencia de instrumento de aplicación reconocido del modelo (INSTRUMENTO); el tiempo de ejecución del instrumento y la edad permitida para aplicar el instrumento; se consideró además el número de trabajos analizados que emplean alguno de estos modelos (NUM TRABAJOS) y área de aplicación (APLICACIÓN INFORMÁTICA).

Tabla 2: Análisis comparativo de modelos/test psicológicos. Elaboración Propia.

Modelo/Test	Modelo Libre	Fácil Aplicación		Instrumento	Tiempo de Ejecución	Edad Aplicación	Número Trabajos	Aplicación Informática			
		ESPEC	Acceso					Definición de patrones de formación de equipos	Evaluación por Competencias	Análisis diversidad/ efectividad	Evaluación de la personalidad
MB	No	Sí	Sí	MBTI	30-40 min	14	5	X		X	X
FFM	Sí	No	Sí	NEO-PI	45 min	16	3	X	X		X

Valorando las ventajas y desventajas de los modelos analizados se evidenció que el modelo FFM es el que presenta mayor ventaja para utilizar en la investigación pues es un modelo fácil de utilizar, no necesita de personal especializado para el análisis de sus datos y ha sido ampliamente utilizado en el ámbito del software. La información que proporciona su aplicación resulta valiosa para la evaluación de la personalidad y del rendimiento individual y por equipos. Además, resulta muy favorable para ayudar a los

a Jefes de Proyecto saber el tipo de personalidad de sus empleados y así poder asignar las tareas correctamente para alcanzar un nivel alto de productividad.

Es válido destacar que los cuestionarios psicológicos por sí solos no identifican a los buenos ingenieros, ni sus resultados pueden predecir de manera absoluta la calidad de la ejecución de un trabajo. Es preciso lograr un balance entre las características o rasgos de la personalidad y el rendimiento.

Cada modelo de personalidad propone por su parte la existencia de diferentes tipos o dimensiones de personalidad con que se podría describir cómo es una persona. Para la categorización del personal en el tipo según el modelo, es necesario el análisis de los rasgos de personalidad que tributan a su clasificación. En función de los rasgos de un individuo se pertenece a un tipo u otro de personalidad. Estos rasgos serán un punto importante en la presente investigación.

1.6.2. Rasgos de Personalidad

En la presente investigación se asume como rasgo las disposiciones que permiten describir a las personas y predecir su comportamiento expresadas por Eysenck en su estudio de la personalidad. Este autor plantea que los rasgos no son observables, sino que se infieren de observar ciertos hechos. Así, rasgo es una disposición de comportamiento expresada en patrones consistentes de funcionamiento en un rango amplio de situaciones.

En los estudios de personalidad es común encontrar la caracterización del individuo a partir de la agrupación de los comportamientos afines en rasgos; y estos rasgos se agrupan en otros rasgos superiores y eventualmente en un número determinado de dimensiones (Revelle, 1995). En este enfoque los comportamientos de los rasgos se extraen a partir del análisis de los hábitos, y a su vez, estas respuestas habituales se extraen de respuestas específicas del individuo en diversas situaciones (Feist, 2009). Los rasgos obtenidos de este análisis tributan a la categorización del individuo en el tipo, factor, o dimensión de personalidad según el modelo. La jerarquía descrita anteriormente se representa en la Ilustración 4.

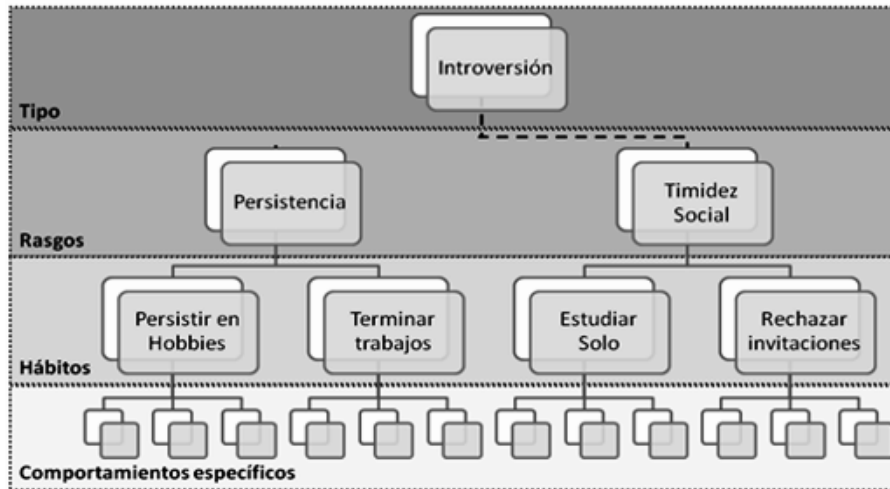


Ilustración 4: La definición del comportamiento en niveles. Adaptado de (Feist, 2009).

De acuerdo al análisis de las fuentes de datos utilizadas y a la definición abordada según el modelo FFM, se definen como rasgos de la personalidad a utilizar en la presente investigación los mostrados en la Ilustración 5:

<p>Neuroticismo</p> <p>Ansiedad</p> <p>Ira</p> <p>Depresión</p> <p>Timidez</p> <p>Impulsividad</p> <p>Vulnerabilidad</p>	<p>Extraversión</p> <p>Amigabilidad</p> <p>Gregarismo</p> <p>Asertividad</p> <p>Actividad</p> <p>Búsqueda de emociones</p> <p>Emoción positiva</p>	<p>Apertura a la experiencia</p> <p>Imaginación</p> <p>Intereses artísticos</p> <p>Sensibilidad emocional</p> <p>Aventurero</p> <p>Curiosidad intelectual</p> <p>Liberalismo</p>
<p>Amigabilidad</p> <p>Confianza</p> <p>Valores éticos</p> <p>Altruismo</p> <p>Cooperación</p> <p>Modestia</p> <p>Simpatía</p>		<p>Responsabilidad</p> <p>Autoeficacia</p> <p>Orden</p> <p>Obediencia</p> <p>Capacidad de logro</p> <p>Autodisciplina</p> <p>Prudencia</p>

Ilustración 5: Rasgos de personalidad definidos por el FFM. Elaboración Propia.

1.7. Herramientas y tecnologías de desarrollo

Uno de los aspectos fundamentales en la elaboración de un software es seleccionar las tecnologías y

herramientas que mayores beneficios aporten. Para realizar la implementación y documentación de este trabajo se estudiaron las tecnologías y herramientas más utilizadas en el desarrollo de sistemas similares al propuesto en esta investigación.

1.7.1. Metodología de desarrollo Extreme Programming (XP)

Para transitar por las fases del proceso de desarrollo del software es necesario aplicar una de las metodologías existentes, las cuales proponen las pautas para arribar a un software con calidad. Teniendo en cuenta los procesos a modelar y el corto plazo de producción se decide utilizar la metodología ágil de desarrollo XP (Ecured, 2014). Ya que es una de las metodologías ágiles más conocidas y utilizadas en la actualidad, desarrollada por Kent Beck y se utiliza para proyectos de corto y mediano alcance, en pequeños equipos y plazos apretados de entrega, además enfatiza las comunicaciones cara a cara con el cliente, en vez de la documentación. El desarrollo con XP se hace de forma iterativa e incremental y dentro de sus características se encuentran (Cockburn, 2005)(Informatizate, 2011):

- **Pruebas unitarias continuas:** son realizadas frecuentemente, repetidas y automatizadas.
- **Corrección:** antes de añadir una nueva funcionalidad se corrigen todos los errores encontrados.
- **Simplicidad en el código:** es más sencillo cambiar algo pequeño que cambiar algo grande.
- **Frecuente interacción:** el cliente siempre es tenido en cuenta. Se recomienda que un representante del cliente trabaje junto con el equipo de desarrollo.
- **Refactorización:** algunas partes del código son reescritas para aumentar su legibilidad y mantención.

El ciclo de vida ideal de XP consiste en cuatro fases fundamentales: Planificación, Diseño, Desarrollo y Prueba. Se trata de realizar ciclos de desarrollo cortos (llamados iteraciones), con entregables funcionales al finalizar cada ciclo. Esta metodología divide las pruebas del sistema en dos grupos: pruebas unitarias, encargadas de verificar el código y diseñada por los programadores, y pruebas de aceptación o pruebas funcionales destinadas a evaluar si al final de una iteración se consiguió la funcionalidad requerida, diseñadas por el cliente final (Chavez, 2010).

1.7.2. Herramienta de modelado Visual Paradigm for UML 8.0 Enterprise Edition

Visual Paradigm es una herramienta CASE²⁰ de software libre que propicia un conjunto de servicios para el desarrollo de programas informáticos y fue concebida para soportar el ciclo de vida completo del proceso de desarrollo del software a través de la representación de un conjunto amplio de diagramas (Pressman, 2001). Es una herramienta colaborativa ya que permite a múltiples usuarios trabajar sobre el mismo proyecto, así como obtener una mejor calidad en el producto. Es fácil de instalar y actualizar y es compatible entre ediciones. Entre sus ventajas se encuentran (Visual Paradigm, 2014):

- Automatiza la realización de diagramas.
- Ayuda a la creación de relaciones en la base de datos.
- Genera estructura de código.
- Ayuda a la documentación del sistema.

Esta herramienta es empleada para visualizar y diseñar elementos de software, para ello utiliza el lenguaje UML. Además, proporciona a los desarrolladores una plataforma que les permite diseñar un producto con calidad de forma rápida (Visual Paradigm, 2014).

1.7.3. Lenguaje de programación Java

En la construcción de la herramienta propuesta se utilizará el lenguaje de programación Java, ya que es un lenguaje avanzado con el que se puede realizar cualquier tipo de programa, desde una aplicación web o desktop hasta aplicaciones independientes sin ninguna conexión a Internet. Al ser un lenguaje libre dispone de una gran cantidad de características descritas a continuación que lo convierte en la herramienta ideal para la creación de aplicaciones distribuidas (Alvarez, 2001):

- Simple.
- Indiferente a la arquitectura.
- Robusto.
- Orientado a objetos.
- Portable.
- Distribuido.
- Dinámico.

Una de las principales características por las que Java se ha hecho muy famoso es por ser un lenguaje independiente de la plataforma (Alvarez, 2001). Wilson en su investigación plantea que actualmente Java se utiliza en un amplio abanico de posibilidades y casi cualquier cosa que se puede hacer en cualquier lenguaje se puede hacer también en Java y muchas veces con grandes ventajas. La versatilidad y eficiencia de la tecnología Java, la portabilidad de su plataforma y la seguridad que aporta, la han convertido en la tecnología ideal para el desarrollo de todo tipo de aplicaciones informáticas (Wilson,

²⁰ Ingeniería de Software Asistida por Computación.

1993).

1.7.4. Entorno de desarrollo integrado

Un entorno de desarrollo integrado (IDE²¹) es un entorno de programación que ha sido empaquetado como un programa de aplicación. Consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica. Los IDEs pueden ser aplicaciones por sí solas o pueden ser parte de aplicaciones existentes. Estos proveen un marco de trabajo amigable para la mayoría de los lenguajes de programación tales como C++, Java y C#. Existen varios IDEs en la actualidad entre los cuales se pueden mencionar C++ Builder, Eclipse, Netbeans y Visual Studio. A continuación se hará referencia al IDE que se utilizará para el desarrollo del sistema:

NetBeans 7.3: es un IDE de código abierto, libre y gratuito pensado para escribir, compilar, depurar y ejecutar programas. Está escrito en Java pero puede servir para cualquier otro lenguaje de programación. Además puede ser usado para desarrollar cualquier tipo de aplicación y tiene la facilidad de poder reutilizar módulos (Netbeans, 2013). Este IDE de desarrollo permite el uso de un amplio rango de tecnologías de desarrollo tanto para escritorio, como aplicaciones Web, o para dispositivos móviles. Puede instalarse en varios sistemas operativos como: Windows, Linux y Mac OS dando soporte a las tecnologías Java, PHP, Groovy, C/C++ y HTML5 (Genbetadev, 2013)

Por las características antes mencionadas, esta plataforma de desarrollo se empleará para la programación de la herramienta, debido a que soporta Java que es el lenguaje escogido para el desarrollo, además de ser un producto libre y gratuito sin restricciones de uso y por la popularidad con que cuenta entre la comunidad de desarrolladores de la UCI.

1.8. Conclusiones Parciales

En este capítulo fueron identificados los conceptos necesarios para ayudar a la comprensión de la investigación tales como: Minería de Datos, algoritmos de clasificación, rendimiento laboral y estudio de la personalidad. Se realizó un análisis y estudio del estado del arte relacionado con diferentes algoritmos de generación de árboles de decisión tales como: C4.5, CHAID, CART y J4.8; analizando ventajas y deficiencias de cada uno respecto a varios aspectos. Además de realizar un análisis bibliográfico sobre la personalidad en los RRHH, los principales modelos utilizados en esta rama y cómo se realiza la evaluación de su rendimiento laboral. Como resultado de este estudio se alcanzaron las siguientes

²¹ Integrated Development Environment,

conclusiones:

- Como base para el entendimiento del tema en que se desenvolverá esta investigación, se describió brevemente el Proceso de Descubrimiento de Conocimiento y dentro de este la fase de Minería de Datos.
- Se enunciaron diferentes tareas de la Minería de Datos y específicamente características de la tarea de clasificación.
- Se realizará la aplicación de algoritmos para el aprendizaje y la clasificación en la generación de árboles de decisión, debido a que el uso de estos algoritmos en problemas similares al de la investigación muestra buenos resultados.
- Se definió evaluar el rendimiento laboral mediante indicadores que permitirán medir la eficiencia, eficacia, el trabajo y aprovechamiento de los individuos.
- El estudio comparativo realizado entre los modelos de la personalidad estudiados posibilitó la selección del modelo MBTI como el más apropiado para aplicar en la solución.

CAPÍTULO 2: PROPUESTA DE SOLUCIÓN

Introducción

En este capítulo se presenta el algoritmo J4.8 para la clasificación basada en árboles de decisión, propuesto a utilizar en la implementación de la herramienta. En el capítulo se describen las características y funcionalidades de la herramienta a partir de los requisitos funcionales y no funcionales, el diagrama de clase de diseño y el diagrama de componentes como parte del modelo de implementación elaborado. Por otro lado se refleja de manera clara y explícita la estrategia de prueba trazada, así como los diferentes niveles y tipos de pruebas que se ejecutaron a la herramienta.

2.1. Descripción del algoritmo

El algoritmo J4.8 es una implementación libre en lenguaje de programación Java del algoritmo C4.5, que utiliza el concepto de entropía de la información para la construcción de árboles de clasificación de un conjunto de entrenamiento y la selección de variables que mejor clasifiquen a la variable (clase) estudiada (García, 2006). Entre sus principales características se encuentra (Takeyas, 2005):

- Permite trabajar con valores continuos para los atributos, separando los posibles resultados en dos ramas $A_i \leq Z$ y $A_i > Z$; siendo Z un umbral escogido anteriormente.
- Los árboles son menos frondosos, ya que cada hoja cubre una distribución de clases, no una clase en particular.
- Utiliza el método "divide y vencerás" para generar el árbol de decisión inicial a partir de un conjunto de datos de entrenamiento.
- Se basa en la utilización del criterio de proporción de ganancia. De esta manera consigue evitar que las variables con mayor número de categorías salgan beneficiadas en la selección.
- Es recursivo.
- Los datos de entrenamiento son un conjunto $S = s_1, s_2...$ de las muestras que ya han sido clasificadas.

Cada caso $S_i = x_1, x_2...$ es un vector, donde $x_1, x_2...$ representan los atributos o características de la muestra. Los datos de entrenamiento se ven aumentadas con un vector $C = c_1, c_2...$ donde $c_1, c_2...$ representan la clase a que pertenece. En cada nodo del árbol, C4.5 elige el atributo que más eficazmente lo divide. Su criterio es el de proporción o ratio de ganancia (ratio gain) que dependiendo de los resultados

tomará la elección de un atributo para dividir los datos (Ver ecuación 5). En el caso de la ganancia como en el de la proporción de ganancia, el mejor atributo para la división es aquel que la maximiza. El algoritmo C4.5 luego seguirá recursivamente en los siguientes subnodos (Lozano, 2013).

Ecuación 5:

$$\text{proporción_de_ganancia}(S) = \frac{\text{Ganancia}(S)}{I_división(S)}$$

La ganancia se calcula según la **Ecuación 6:**

$$\text{Gan Inf}(S, A) = \text{Entropia}(S) - \sum_{v \in V(A)} \frac{|Sv|}{|S|} \text{Entropia}(Sv)$$

Donde:

S: colección de objetos o casos.

A: son los atributos de los objetos o casos.

V(A): conjunto de valores que A puede tomar.

La entropía se calcula según la **Ecuación 7:**

$$\text{Entropia}(S) = \sum_{i=1}^n -p_i \log_2 p_i$$

Siendo:

S: colección de objetos o casos.

Pi: es la probabilidad de los posibles valores.

i: las posibles respuestas de los objetos.

La información de división se calcula según la **Ecuación 8:**

$$I_división(X) = - \sum_{i=1}^n \frac{|T_i|}{|T|} \times \log_2 \left(\frac{|T_i|}{|T|} \right)$$

Donde:

T: total de atributos del objeto o caso **X**.

Ti: cantidad de atributos i del objeto o caso **X**.

Este algoritmo tiene unos pocos casos base (Quinlan, 1993):

- Todas las muestras en la lista pertenecen a la misma clase. Cuando esto sucede, simplemente crea un nodo de hoja para el árbol de decisión diciendo que elija esa clase.
- Ninguna de las características proporciona ninguna ganancia de información. En este caso, C4.5 crea un nodo de decisión más arriba del árbol utilizando el valor esperado de la clase.
- Instancia de la clase previamente no vista encontrada. Una vez más, C4.5 crea un nodo de decisión más arriba en el árbol con el valor esperado.

El pseudocódigo del algoritmo se detalla en la Ilustración 6:

Función C4.5

(R: conjunto de atributos no clasificadores,
C: atributo clasificador,
S: conjunto de entrenamiento) devuelve un árbol de decisión;

Comienzo

Si S está vacío,

Devolver un único nodo con Valor Falla;

Si todos los registros de S tienen el mismo valor para el atributo clasificador,

Devolver un único nodo con dicho valor;

Si R está vacío, entonces

Devolver un único nodo con el valor más frecuente del atributo clasificador en los registros de S [Nota: habrá errores, es decir, registros que no estarán bien clasificados en este caso];

Si R no está vacío, entonces

$D \leftarrow$ atributo con mayor Proporción de Ganancia(D,S) entre los atributos de R;

Sean $\{d_j | j=1,2, \dots, m\}$ los valores del atributo D;

Sean $\{S_j | j=1,2, \dots, m\}$ los subconjuntos de S correspondientes a los valores de d_j respectivamente;

Devolver un árbol con la raíz nombrada como D y con los arcos nombrados d_1, d_2, \dots, d_m que van respectivamente a los árboles

$C4.5(R-\{D\}, C, S_1), C4.5(R-\{D\}, C, S_2), \dots, C4.5(R-\{D\}, C, S_m)$;

Fin

Ilustración 6 : Algoritmo C4.5. Tomado de (Takeyas, 2005).

2.2. Levantamiento y especificación de requisitos

Todas las ideas que los clientes, usuarios y miembros del equipo de proyecto tengan acerca de lo que debe hacer la herramienta, deben ser analizadas como candidatas a requisito. Un requisito de software es una propiedad que debe ser exhibida por el software desarrollado o adaptado para resolver un problema particular. Los mismos se pueden clasificar en: Funcionales y No funcionales (Pressman, 2001). Para el levantamiento de estos requerimientos se aplicaron las técnicas de captura de requisitos Sistemas Existentes y Observación, específicamente la Etnografía, obteniéndose un total de 21 requisitos, de ellos

13 funcionales y 8 no funcionales. Posteriormente estos requisitos fueron validados mediante las técnicas de validación de requerimientos Revisión Técnica Formal y Prototipo de Interfaz de Usuario.

2.2.1. Requisitos funcionales

Los requisitos funcionales son características del sistema que expresan funcionalidades, es decir, describen lo que el sistema debe hacer (Pressman, 2001). A continuación se listan los requisitos funcionales de la aplicación a desarrollar, los cuales serán descritos con mayor detalle en las Historias de Usuario:

RF1: Gestionar fichero.

- R.1.1: Cargar fichero desde una ubicación especificada.
- R.1.2: Salvar en fichero el resultado de la clasificación.
- R.1.3: Salvar en fichero el resultado de la selección de atributos.

RF2: Realizar Clasificación.

- R.2.1: Configurar parámetros de ejecución del algoritmo de clasificación.
- R.2.2: Seleccionar atributo clase para clasificar.
- R.2.3: Visualizar árbol generado por el algoritmo.
- R.2.4: Mostrar resultado de la clasificación.
- R.2.5: Mostrar la lista de corridas de clasificación.

RF3: Realizar Selección de atributos.

- R.3.1: Seleccionar modo de selección de atributo.
- R.3.2: Seleccionar atributo clase para la selección de atributo.
- R.3.3: Mostrar el resultado de la salida de selección de atributo.
- R.3.4: Mostrar la lista de corridas de selección.

RF4: Graficar Resultados.

- R.4.1: Mostrar gráfica con los resultados obtenidos.

2.2.2. Requisitos no funcionales

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Representan las características del producto, así como las características para que este sea atractivo, confiable, usable y seguro (Pressman, 2001). Los Requisitos No Funcionales identificados en la solución son los siguientes:

Software:

Se debe disponer en las computadoras de Sistema Operativo Linux, Windows XP o superior para la

instalación de la aplicación, garantizando una interoperabilidad de forma similar en diferentes sistemas operativos o plataformas.

Se necesita tener instalada una Máquina Virtual de Java versión 1.5 o superior.

Hardware:

Para el desarrollo y utilización del proyecto se requieren computadoras con Procesador Pentium 3 o superior, 512 MB de RAM o más y 30 GB de capacidad de disco duro.

Usabilidad:

La herramienta debe ser utilizada por cualquier tipo de persona que posea conocimientos básicos en el manejo de la computadora.

Soporte:

Mantenimiento: La herramienta debe estar bien documentada de forma tal que el tiempo de mantenimiento sea mínimo en caso de necesitarse.

Instalación: La instalación de la herramienta debe caracterizarse por su facilidad, claridad y sencillez.

Apariencia o interfaz externa:

La herramienta debe estar diseñada con una interfaz amigable, de forma tal que el usuario navegue sin dificultad alguna.

Portabilidad:

La herramienta debe ser ejecutada sobre los sistemas operativos Linux y Windows, por su característica de ser multiplataforma.

2.2.3. Historias de Usuario

Las historias de usuario son la técnica utilizada en la metodología XP para especificar los requisitos del software en la fase de planificación, proporcionan ventajas tales como (Jeffries, 2001):

- Están escritas en lenguaje del cliente, por lo que es muy fácil su comprensión.
- Especifican cada uno de los requisitos de la herramienta, sin necesidad de documentaciones extensas.
- Reflejan todas las características de la herramienta.

- Si se definen correctamente, guían el proceso de implementación.

Como resultado del trabajo realizado durante esta fase se muestra en la Tabla 4 la descripción textual de la historia de usuario *Mostrar el resultado de la salida del clasificador*. El resto de las Historias de Usuarios se pueden consultar en el Anexo 2:

Tabla 3: HU Mostrar el resultado de la salida del clasificador.

Historia de Usuario	
Número: 7	Nombre: Mostrar el resultado de la clasificación.
Usuario: Todos	
Prioridad en Negocio: Alta	Riesgo en Desarrollo: Bajo
Programador responsable: Yoendry Betancourt Peña.	
Iteración Asignada: 1	
Descripción: Permite al usuario visualizar el resultado de la aplicación del algoritmo de selección de atributos por ejemplo la cantidad de atributos seleccionados, el método de búsqueda utilizado, el evaluador de atributos utilizado, la cantidad de instancias correctamente e incorrectamente clasificadas, el error mínimo, el valor verdadero positivo, el valor falso positivo y la predicción.	
Observaciones: Da cumplimiento al requisito funcional 7.	

2.2.4. Plan de iteraciones

Luego de ser confeccionadas las HU se procede a realizar la planificación de la etapa de implementación de la herramienta, elaborando el plan de iteraciones. En este plan se especifica la prioridad con que se irán desarrollando las HU organizadas por iteraciones (Ver Anexo 3). Una iteración no es más que un mini-proyecto donde se obtiene un resultado del software con valor para el cliente. Es preciso aclarar que no se obtiene la versión completa del producto, ya que este concluye al finalizar la última iteración (Rodríguez, 2007).

Iteración 1:

Esta iteración tiene como objetivo la implementación de las HU de prioridad alta con una duración de seis semanas. Propiciando al finalizar tener implementadas las peticiones del cliente descritas en las historias, contando con las funcionalidades principales para el funcionamiento de la herramienta.

Iteración 2:

En esta iteración serán implementadas las HU de prioridad media con una duración de tres semanas, las mismas tienen como propósito brindarle comodidad al cliente en la gestión de otras tareas asociadas a las

historias de prioridad alta. Al terminar esta iteración se obtendrá el producto final, listo para poner en funcionamiento y ser evaluado.

2.2.5. Plan de entrega

Es un artefacto generado como parte de la planificación que permite realizar una organización del tiempo de las entregas al final de cada una de las iteraciones, permitiéndoles así a los programadores una guía para el desarrollo en tiempo de la solución.

Tabla 4: Plan de entrega de las iteraciones.

Entregable	Fin Iteración 1	Fin Iteración 2
Aplicación de un algoritmo de clasificación mediante una herramienta para el análisis de la influencia de los rasgos de personalidad en el rendimiento laboral.	Tercera semana Mayo/2014	Primera semana de Junio/2014

2.3. Diseño de la herramienta

Una vez planificada la realización de la herramienta se procede al diseño, definiendo los patrones a utilizar y realizando los diagramas de clase de diseño donde se registran las relaciones más frecuentes entre las clases de la herramienta. También XP permite obtener las tarjetas CRC²² para ilustrar las clases y la información necesaria a almacenar en cada una de ellas. Con el desarrollo de esta fase se pretende conseguir diseños simples y sencillos para procurar hacerlo todo lo menos complicado posible para el usuario o cliente y un diseño fácilmente entendible e implementable que a la larga costará menos tiempo y esfuerzo.

2.3.1. Tarjetas CRC

Las tarjetas CRC son una técnica utilizada en XP, su uso permite un diseño de software orientado a objetos. Estas tarjetas proponen una forma de trabajo para encontrar los objetos del dominio de la herramienta, donde la clase a la que pertenecen se puede escribir arriba mientras que en una columna a la izquierda se pueden reflejar las responsabilidades u objetivos que debe cumplir el objeto y a la derecha, las clases que colaboran con cada responsabilidad. De esta manera el algoritmo para utilizar las tarjetas CRC es: primero la identificación de clases y asociaciones que participan en el diseño de la herramienta, luego la obtención de las responsabilidades que debe cumplir cada clase y por último el establecimiento de cómo una clase colabora con otras para cumplir con sus responsabilidades (XP, 2006). A continuación

²² Clase, Responsabilidad, Colaboración.

se expone una representación de una de las tarjetas obtenidas durante el proceso de desarrollo de la herramienta, las demás se pueden consultar en el Anexo 4:

Tabla 5: Descripción de la Tarjeta CRC PersonalidadModel.

Clase PersonalidadController	
Responsabilidades	Colaboradores
Realizar la clasificación de los atributos. Realizar la selección de atributos. Permitir visualizar el árbol generado por el algoritmo. Permitir guardar los resultados obtenidos. Graficar los resultados.	PersonalidadModel

2.3.2. Diagrama de clase del diseño

A partir de la definición de las tarjetas CRC, donde se identificaron las clases de la herramienta y sus principales responsabilidades, se está en condiciones de confeccionar el diagrama de clases del diseño. Este diagrama aunque no es un artefacto propio de XP, permitirá representar gráficamente las clases (atributos y métodos), las relaciones entre clases (dependencias y asociaciones) y la navegabilidad. Los diagramas de clase son la primicia para entender lo que posteriormente se va a implementar. A continuación en la Ilustración 8 se muestra el diagrama de clase del diseño de la solución (Ver detalles en el Anexo 5):

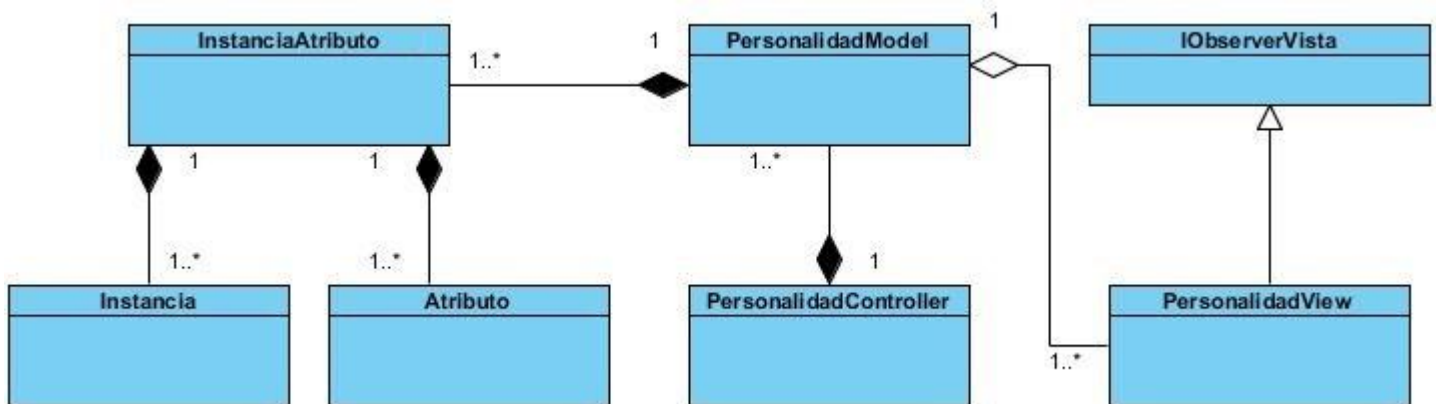


Ilustración 7: Diagrama de clases del diseño.

2.3.3. Patrones de diseño utilizados

Los patrones de diseño son un conjunto de estrategias, o buenas prácticas, que pueden facilitar el trabajo en muchas situaciones en el momento de realizar una aplicación orientada a objetos (Larman, 1999). Seguidamente se describirán los patrones utilizados en el diseño de la investigación:

Patrón arquitectónico Modelo-Vista-Controlador:

Se encarga de separar interfaces, fue creado con Smalltalk-809²³. El modelo es la capa del dominio, la vista es la capa de presentación y el controlador son los objetos de flujo de trabajo. Es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. Donde el modelo representa la información con la que trabaja la aplicación y se encarga de acceder a los datos, la vista transforma esta información obtenida por el modelo en las interfaces a las que acceden los usuarios y el controlador es el encargado de coordinar todos los demás elementos y transformar las peticiones del usuario en operaciones sobre el modelo y la vista (Larman, 1999). En la investigación la vista está representada por la interfaz principal de la herramienta manejando la visualización de la información, el modelo por las clases donde se manejan los datos del fichero cargado y el controlador por la clase controladora donde se analiza los mensajes de eventos que recibe la herramienta y modifica u obtiene datos del modelo en respuesta a las peticiones del usuario.

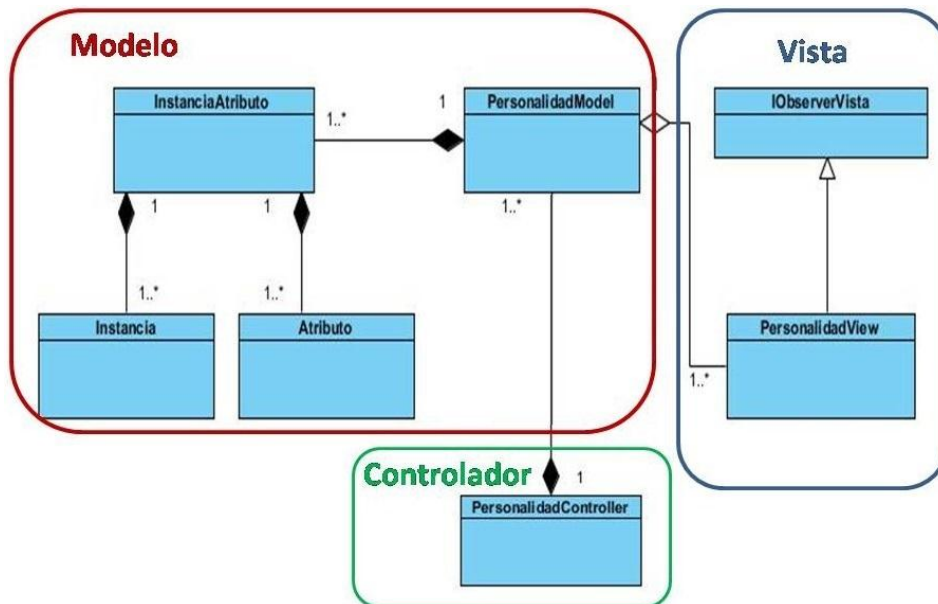


Ilustración 8: Aplicación del patrón MVC en el diagrama de clases de la solución.

Patrones GRASP²⁴:

Experto: propone asignar la responsabilidad a la clase que cuenta con la información necesaria para cumplir la responsabilidad. Permitiendo que se conserve el encapsulamiento, soportando un bajo

²³ Lenguaje de programación reflexivo orientado a objetos, de tipo dinámico.

²⁴ Patrones de asignación de responsabilidades.

acoplamiento y una alta cohesión (Larman, 1999). Es evidenciado en la clase *PersonalidadModel*, la cual cuenta con la información necesaria para acceder a los atributos y datos del fichero a clasificar.

Creador: en el patrón creador se asigna la responsabilidad de que una clase B cree un objeto de la clase A si se cumple uno o más de los casos siguientes (respondiendo al problema de quién debería ser el responsable de una nueva instancia de una clase) (Larman, 1999):

- B agrega objetos de A.
- B contiene objetos de A.
- B registra instancias de objetos de A.
- B utiliza más estrechamente objetos de A.
- B tiene los datos de inicialización que se pasarán a un objeto A cuando sea creado (por tanto B es un Experto con respecto a la creación de A).

En este caso la clase *PersonalidadModel* tiene la responsabilidad de crear un objeto de la clase *InstanciaAtributo* para acceder a los datos de la misma.

Controlador: esta clase sirve como intermediaria entre la interfaz del formulario principal y las clases subyacentes, de tal forma que es la que recibe los datos del usuario y la que los envía a las distintas clases según el método llamado (Larman, 1999). En esta ocasión la clase *PersonalidadControler* es la que maneja todos los eventos y funcionalidades relacionadas con la clasificación y selección de los datos de entrenamiento por el algoritmo propuesto.

Alta Cohesión: este patrón propone asignar la responsabilidad de manera que la complejidad se mantenga dentro de límites manejables asumiendo solamente las responsabilidades que deben manejar, evadiendo un trabajo excesivo. Su utilización mejora la claridad y facilidad con que se entiende el diseño, simplifica el mantenimiento y las mejoras de funcionalidad, generan un bajo acoplamiento y soporta mayor capacidad de reutilización (Gamma, 1995). En todas las clases se implementa la funcionalidad de este patrón, ejemplo la clase *InstanciaAtributo* donde se encapsula solo aquella información que requiere.

Bajo Acoplamiento: propone asignar una responsabilidad para mantener bajo acoplamiento. Permite mantener las relaciones mínimas entre las clases, de forma que en caso de realizar modificaciones no implique grandes cambios al resto de las clases (Larman, 1999). En este caso las clases no tienen relaciones innecesarias con otras clases.

Patrones GOF²⁵:

Observer (Observador): Define una dependencia de uno a muchos entre objetos de forma que, cuando un objeto cambia de estado, se notifica a los objetos dependientes para que se actualicen automáticamente (Larman, 1999). Este patrón se utilizó cuando se necesita añadir alguna modificación en las clases del modelo se notifique esos cambios en las clases de la vista.

Patrón Solitario (Singleton): Este es un patrón cuyo propósito es la creación de instancias únicas de objetos y precisamente por esto actúa en un ámbito de objetos. Su utilidad radica en garantizar que una clase tenga una única instancia y de esta forma proporcionar un único punto de acceso a los miembros de ella (Larman, 1999). Se utilizó en la clase *PersonalidadController* en la cual se crea una instancia de la misma que puede ser utilizada desde cualquier otra clase.

2.3.4. Métricas de diseño

El IEEE “Standard Glossary of Software Engineering Terms” define como métrica: “Una medida cuantitativa del grado en que un sistema, componente o proceso posee un atributo dado” (IEEE, 2006). Alguno de los elementos que permiten evaluar la calidad del diseño son las métricas orientadas a objetos como son:

- Métricas orientadas a las operaciones.
- Métricas sobre el tamaño del software.
- Métricas para pruebas orientadas a objeto.
- Métricas orientadas a clases.

En este último caso existen varios tipos de series aplicadas, se pueden citar algunas como las de Chidamber y Kemerer (CK) y Lorentz y Kidd (LK). En el diseño desarrollado se aplicaron algunas de las métricas propuestas por las series CK y LK, por ser las más reconocidas y las que más se ajustan al diseño desarrollado.

En su libro sobre métricas Lorenz y Kidd separan las métricas basadas en clases en cuatro amplias categorías: tamaño, herencia, valores internos y valores externos. Las métricas orientadas al tamaño para las clases se centran en el recuento de atributos y operaciones para cada clase individual, y los valores promedio para el sistema como un todo (Pressman, 2001). Por lo que para la validación del diseño de la herramienta se escogió de la serie LK la métrica:

Tamaño Operacional de Clases (TOC): donde las clases pueden medirse determinando el total de operaciones, tanto heredadas como privadas de la instancia que se encapsulan dentro de una clase, más

²⁵ Patrones Gang of Four o “pandilla de los cuatro”.

el total de atributos, atributos tanto heredados como privados de la instancia encapsulados por la clases. Los valores grandes para esta métrica, indican que la clase posee bastante responsabilidad y complejidad, esto reduce la reutilización de esta clase y dificulta las pruebas sobre la misma (Pressman, 2001). La aplicación de la métrica TOC define los siguientes atributos de calidad:

- **Responsabilidad:** consiste en la responsabilidad asignada a una clase en un marco de modelado de un dominio o concepto, de la problemática propuesta.
- **Complejidad de implementación:** consiste en el grado de dificultad que tiene implementar un diseño de clases determinado.
- **Reutilización:** consiste en el grado de reutilización presente en una clase o estructura de clase, dentro de un diseño de software.

A continuación se muestran las clases que juegan un papel fundamental en los procesos principales de la herramienta:

Tabla 6: Clases fundamentales de la herramienta.

Clase	Cantidad de procedimientos
Atributo	6
Instancia	2
InstanciaAtributo	4
PersonalidadModel	29
PersonalidadController	14
PersonalidadView	3
IObserverVista	1

Resultados:

Los umbrales que referencian a los atributos de calidad: responsabilidad de las clases, complejidad al implementar las mismas, así como sus niveles de reutilización, los cuales se le aplican a las 7 clases de la herramienta con un promedio de 8.43 operaciones, quedan especificados en la siguiente tabla:

Tabla 7: Umbrales para la responsabilidad, complejidad y reutilización de la métrica TOC.

Atributos de calidad	Categoría	Criterio
Responsabilidad	Baja	< =8.43
	Media	Entre 8.43 y 16.86
	Alta	> 16.86
Complejidad Implementación	Baja	< =8.43
	Media	Entre 8.43 y 16.86
	Alta	> 16.86

CAPÍTULO 2: PROPUESTA DE SOLUCIÓN

Reutilización	Baja	> 16.86
	Media	Entre 8.43 y 16.86
	Alta	<= 8.43

En la Tabla y las figuras siguientes se muestran cómo quedan reflejados los atributos de calidad antes mencionados en las clases sometidas a la métrica de diseño propuesta:

Tabla 8: Clases analizadas con resultados obtenidos con la métrica TOC.

Clase	Cantidad de procedimientos	Responsabilidad	Complejidad	Reutilización
Atributo	6	Baja	Baja	Alta
Instancia	2	Baja	Baja	Alta
InstanciaAtributo	4	Baja	Baja	Alta
PersonalidadModel	29	Alta	Alta	Baja
PersonalidadController	14	Media	Media	Media
PersonalidadView	3	Baja	Baja	Alta
IObserverVista	1	Baja	Baja	Alta

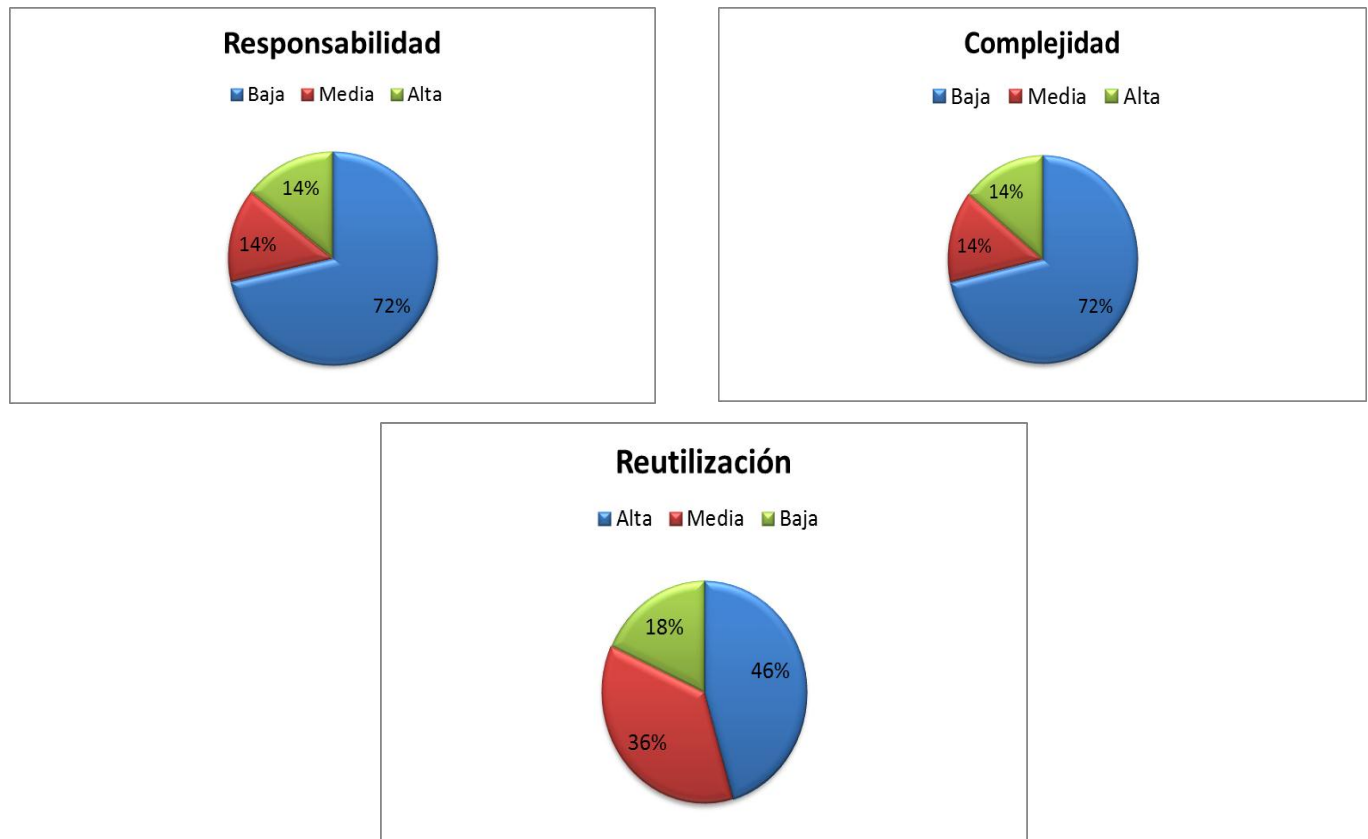


Ilustración 9: Atributos de calidad evaluados en la métrica TOC.

Luego de analizados los resultados obtenidos de la aplicación de la métrica TC se puede afirmar que los mismos son satisfactorios y se incluyen en los rangos de calidad aceptables dentro del desarrollo de software. La aplicación de esta métrica arrojó como resultado una baja responsabilidad y complejidad y una alta reutilización, esto favorece al buen uso del diseño empleado y ayuda en la eliminación de clases repetidas, teniendo en cuenta que más de la mitad de las clases poseen baja dependencia respecto a otras. Esto evidencia que el sistema no tendrá una compleja implementación.

Relación entre clases (RC): La métrica RC está definida por el número de relaciones de uso de una clase con otra, definiendo los siguientes criterios y categorías de evaluación para los atributos de calidad (Ecured, 2014):

- **Reutilización:** Consiste en el grado de reutilización presente en una clase o estructura de clase, dentro de un diseño de software.
- **Acoplamiento:** Consiste en el grado de dependencia o interconexión de una clase o estructura de clase, con otras, está muy ligada a la característica de Reutilización.
- **Complejidad del mantenimiento:** Consiste en el grado de esfuerzo necesario a realizar para desarrollar un arreglo, una mejora o una rectificación de algún error de un diseño de software. Puede influir indirecta, pero fuertemente en los costos y la planificación del proyecto.
- **Cantidad de pruebas:** Consiste en el número o el grado de esfuerzo para realizar las pruebas de calidad (Unidad) del producto (componente, módulo, clase, conjunto de clases, etc.) diseñado.

Resultados:

Tabla 9: Umbrales para la reutilización, complejidad del mantenimiento, acoplamiento y cantidad de pruebas de la métrica RC.

Atributos de calidad	Categoría	Criterio
Acoplamiento	Ninguno	0
	Bajo	1
	Medio	2
	Alto	>2
Responsabilidad	Baja	≤ 0.86
	Media	Entre 0.86 y 1.71
	Alta	> 1.71
Complejidad Implementación	Baja	> 1.71
	Media	Entre 0.86 y 1.71
	Alta	≤ 0.86
Reutilización	Baja	≤ 0.86
	Media	Entre 0.86 y 1.71
	Alta	> 1.71

CAPÍTULO 2: PROPUESTA DE SOLUCIÓN

En la Tabla y las figuras que se muestran a continuación quedan reflejados los atributos de calidad antes mencionados en las clases sometidas a la métrica de diseño propuesta:

Tabla 10: Clases analizadas y resultados obtenidos de la métrica RC.

Clase	Cantidad de Relaciones de uso	Acoplamiento	Complejidad Mantenimiento	Reutilización	Cantidad de Pruebas
Atributo	1	Bajo	Media	Media	Media
Instancia	1	Bajo	Media	Media	Media
InstanciaAtributo	1	Bajo	Media	Media	Media
PersonalidadModel	2	Medio	Alta	Baja	Alta
PersonalidadController	1	Bajo	Media	Media	Media
PersonalidadView	0	Ninguno	Baja	Alta	Baja
IObserverVista	0	Ninguno	Baja	Alta	Baja

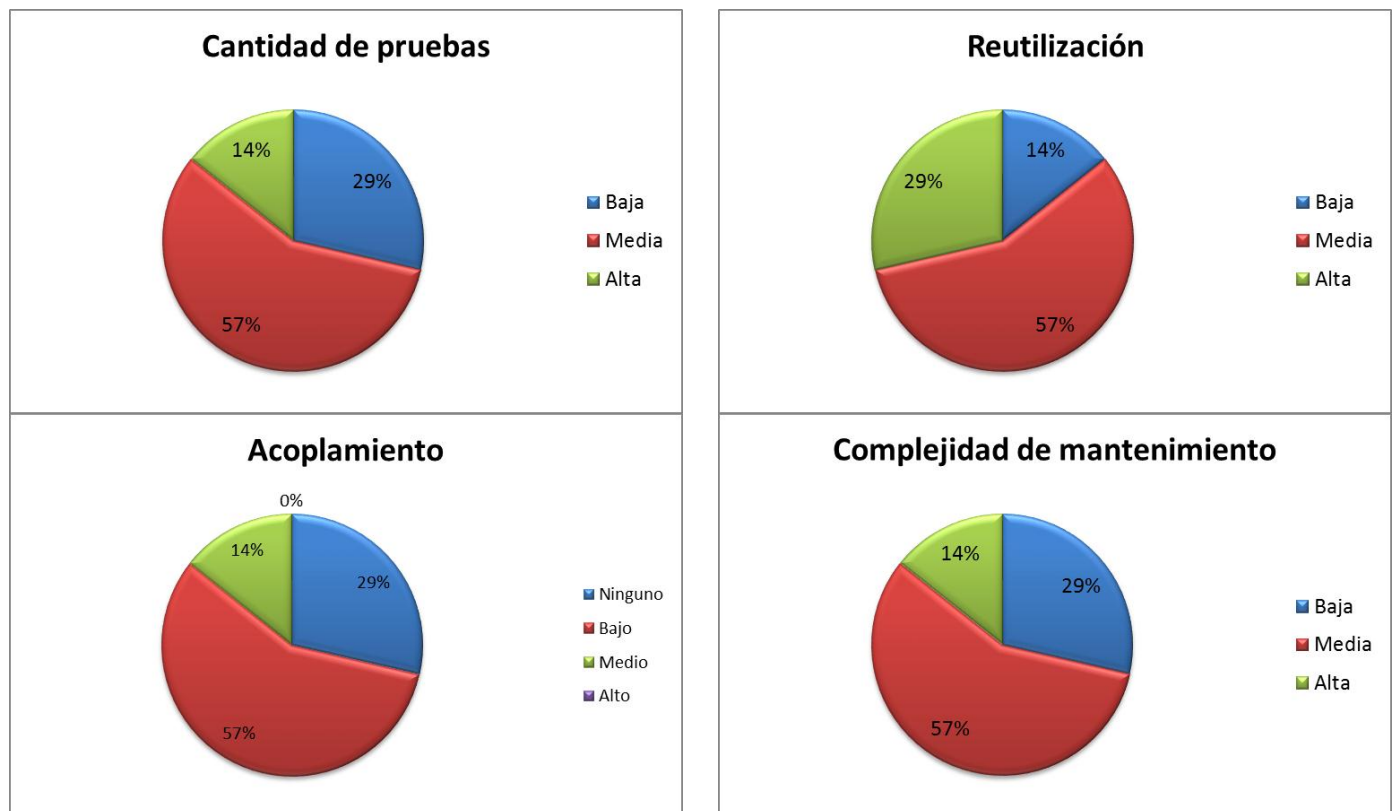


Ilustración 10: Atributos de calidad evaluados en la métrica RC.

Como resultado de la aplicación de la métrica RC se evidencia que las clases del sistema poseen una reutilización, complejidad de mantenimiento y cantidad de pruebas media, mientras que el acoplamiento es bajo, por lo que se puede concluir que el diseño de las clases en cuanto a cantidad de relaciones de cada una en la herramienta es bueno.

De la serie CK se emplearon las siguientes métricas:

Carencia de cohesión en los métodos (CCM): cada método dentro de una clase, C, accede a uno o más atributos (también llamados variables de instancia). CCM es el número de métodos que accede a uno o más de los mismos atributos. Si no existen métodos que accedan a los mismos atributos, entonces $CCM = 0$. Un valor más alto de CCM indica la disminución de la encapsulación y el aumento de la complejidad, lo que aumenta la probabilidad de errores. CCM varía en el intervalo $[0,2]$. Si $CCM \geq 1$ indica una clase muy problemática. Lo deseable es mantener alta la cohesión, es decir conservar baja la CCM (Pressman, 2010). La métrica se aplicará a la clase *PersonalidadModel*, posteriormente a cada uno de los atributos de la clase se le fijará una identificación:

Tabla 11: Atributos identificados de la clase Personalidad.

PersonalidadModel	
Atributos	Identificador
datosBase	a
datosEntrenamiento	b
modeloConstruido	c
estadisticasClase	d
medicionesEntropia	e
matrizConfusion	f
atributoClase	g
numeroParticiones	h
semilla	i
ficheroBase	j
ficheroEntrenamiento	k
listCorridasClasificacion	l
listCorridasSeleccion	m
listaObserver	n

Después de identificar cada atributo, se le establece a los métodos de la clase los atributos correspondientes:

Tabla 12: Métodos de la clase PersonalidadModel y atributos correspondientes.

PersonalidadModel		PersonalidadModel	
Métodos	Atributos	Métodos	Atibutos
registrarVista	n	getNumeroParticiones	h
getDatosBase	a	setNumeroParticiones	h
setDatosBase	a	getSemilla	i
getDatosEntrenamiento	b	setSemilla	i
setDatosEntrenamiento	b	getFicheroBase	j
getModeloConstruido	c	setFicheroBase	j
setModeloConstruido	c	getFicheroEntrenamiento	k
getEstadisticaClase	d	setFicheroEntrenamiento	k
setestadisticaClase	d	getListaCorridasClasificacion	l
getMedicionesEntropia	e	setListaCorridasClasificacion	l
setMedicionesEntropia	e	getListaCorridasSeleccion	m
getMatrizConfusion	f	setListaCorridasSeleccion	m
setMatrizConfusion	f	getListaObserver	n
getAtributoClase	g	setListaObserver	n
setAtributoClase	g		

Una vez obtenidos los atributos con acceso a cada método, se calcula la cantidad de métodos que acceden a uno o más de los mismos atributos:

Tabla 13: Total de métodos que acceden a un atributo de la clase PersonalidadModel.

PersonalidadModel		
Método	Identificación	mA
datosBase	a	2
datosEntrenamiento	b	2
modeloConstruido	c	2
estadisticasClase	d	2
medicionesEntropia	e	2
matrizConfusion	f	2
atributoClase	g	2
numeroParticiones	h	2
semilla	i	2
ficheroBase	j	2
ficheroEntrenamiento	k	2
listCorridasClasificacion	l	2
listCorridasSeleccion	m	2
listaObserver	n	3
$\Sigma(mA)$		29

Finalmente, luego de obtener los datos se calcula la CCM mediante la siguiente fórmula:

$$FCM = 1 - \frac{\sum(mA)}{(m \times a)}$$

Donde:

m : Número de métodos en la clase. ($m=29$).

a : Número de atributos en la clase. ($a=14$).

mA : Número de métodos que acceden a un atributo. ($\sum(mA)=29$).

$$FCM = 1 - 29 / (29 \times 14) = 1 - 29 / 406 = 1 - 0.071 = \mathbf{0.929}$$

Para un valor de $FCM \geq 1$ se alcanzaría una baja cohesión, lo que indica que el resultado obtenido existe una alta cohesión, por lo cual se puede afirmar que *PersonalidadModel* es una clase bien diseñada.

2.4. IMPLEMENTACIÓN

La metodología XP plantea que la implementación de un software debe realizarse de forma iterativa, obteniendo al finalizar cada iteración un producto funcional que debe ser probado y mostrado al cliente (Pressman, 2001). En esta fase se realiza la implementación de cada una de las HU durante el transcurso de las iteraciones, creándose las tareas de la ingeniería para ayudar a organizar la implementación exitosa de las HU, estas tareas son asignadas a los programadores que son los responsables de la implementación.

2.4.1. Tareas de ingeniería

Para llevar a cabo las tareas de la ingeniería se cuenta con una tabla la cual permite definir cada una de las actividades en las que estarán asociadas cada HU. Estas brindan un escaso nivel de detalle para afrontar la implementación por lo que las tareas de ingeniería juegan un papel fundamental al indicar a los programadores las acciones a realizar por cada una de ellas (XP, 2006). Para consultar las tareas de ingeniería pertenecientes al desarrollo de la herramienta Ver Anexo 6.

2.4.2. Diagrama de componentes

El diagrama de componentes representa, como indica su nombre, la separación de un sistema de software en componentes físicos más pequeños (por ejemplo archivos, módulos, librerías, paquetes, etc.) y muestra las dependencias y organización entre estos componentes. Las dependencias en este diagrama indican que un componente utiliza los servicios ofrecidos por otro componente (Larman, 1999). A

continuación en la Ilustración 11 se describe el diagrama de componentes de la herramienta.

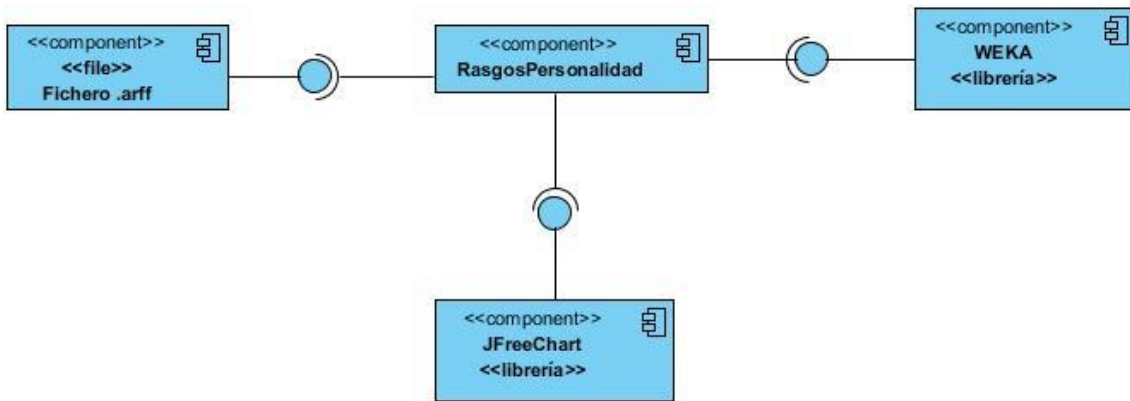


Ilustración 11: Diagrama de componente de la solución.

RasgosPersonalidad: es el encargado de realizar todo el proceso de clasificación del conjunto de datos de entrenamiento a través del algoritmo J4.8.

JFreeChart: contiene los paquetes y clases que se utilizan para la graficación de los resultados.

WEKA: contiene los paquetes y clases necesarias para que las selecciones de atributos funcionen de forma correcta, brindando los métodos útiles para el desarrollo de ese servicio.

Fichero .arff: contiene el conjunto de datos de entrenamiento a utilizar en la investigación.

2.4.3. Estándares de codificación

Establecer un estándar de codificación que sea aceptado por todo el equipo de desarrollo es muy importante en una metodología como XP que promueve la propiedad colectiva del código y la constante refactorización. Además, influye directamente en la facilidad con que pueda mantenerse el código si fuese necesario añadir nuevas funcionalidades al software, modificar las ya existentes, depurar errores o mejorar el rendimiento. Para la implementación de la herramienta se utilizó el estándar de codificación que se describe a continuación.

Para la capitalización de los identificadores se utilizaron los convenios:

Pascal: La primera letra en el identificador y la primera letra de cada subsiguiente palabra concatenada se capitalizan. Por ejemplo: *PersonalidadModel*.

Camel: La primera letra en el identificador se pone en minúscula y la primera letra de cada subsiguiente palabra concatenada en mayúscula. Por ejemplo: *visualizarArbol()*.

La convención Pascal se empleó en los identificadores de las clases. Mientras que la convención Camel es el estilo de los identificadores de las variables, métodos, atributos y parámetros.

2.5. Pruebas

Las pruebas del software son un elemento crítico para la garantía de calidad del software y representa una revisión final de las especificaciones, del diseño y de la codificación (Pressman, 2001). Una de las características principales de XP, es su fuerte énfasis en las pruebas, comprobando el funcionamiento de la herramienta. En ellas los desarrolladores realizan pruebas constantemente tanto como sea posible; de esta manera se reduce el número de errores no detectados, así como el tiempo entre la introducción de éste en la herramienta y su detección. Todo esto contribuye a elevar la calidad del producto desarrollado. XP divide las pruebas en dos grupos Pruebas unitarias y de aceptación las cuales fueron seleccionadas para la validación de la herramienta (Pressman, 2005). Para conocer los detalles de las pruebas puede consultar el Capítulo 3: Resultados y validación de la propuesta.

2.6. Conclusiones del capítulo

Con la realización de este capítulo siguiendo la metodología de software seleccionada y la utilización de las herramientas propuestas se concluyó que:

- Se realizó la definición y descripción de los requisitos de la herramienta, que permitió cumplir con todas las funcionalidades requeridas para el proceso de clasificación basada en árboles de decisión. Se describieron las HU, artefacto generado en la fase de planificación de la metodología XP, realizándose una estimación de esfuerzo por cada una de las historias, y quedando definido el plan de duración de entregas.
- Se definieron los componentes del sistema en cuestión, así como las tarjetas CRC que son los artefactos generados en la fase de diseño de la metodología XP. Se realizó además la descripción del diagrama de clases de diseño para conocer la estructura y las relaciones entre las clases que se manejan en la herramienta.
- El uso del modelo arquitectónico definido posibilitó visualizar la herramienta desde diferentes puntos de vistas y mediante la correcta aplicación de los patrones de diseño, se facilitó la reutilización de código y una mejora considerable del nivel de complejidad de las clases.

CAPÍTULO 2: PROPUESTA DE SOLUCIÓN

- Se aplicaron métricas de diseño que proporcionaron porcentos satisfactorios en la validación del diseño de la herramienta.
- La utilización de los estándares de codificación a utilizar, facilitaron el entendimiento del código por los programadores y el futuro mantenimiento de la herramienta.
- La correcta aplicación de la metodología seleccionada acortó el tiempo de desarrollo estimado.

CAPÍTULO 3: RESULTADOS Y VALIDACIÓN DE LA PROPUESTA

Introducción

En este capítulo se realizará un análisis de los resultados obtenidos en la investigación y se expondrán las características y funcionalidades de la herramienta implementada. Se presenta un análisis comparativo basado en pruebas experimentales realizadas al fichero de muestra, así como un conjunto de Pruebas de aceptación y unitarias definidas con anterioridad para validar la solución desarrollada.

3.1. DESCRIPCIÓN DE LA HERRAMIENTA

La herramienta desarrollada permite mediante la aplicación del algoritmo J4.8 facilitar el análisis de la influencia de los rasgos de personalidad en el rendimiento laboral. Para obtener estos resultados es necesario seguir un conjunto de pasos especificados a continuación:

1. Confeccionar la Base de Casos (BC) a procesar

Inicialmente para la confección de la BC se seleccionan los individuos, indistintamente al sexo que presentan y al rol que desempeñan. Utilizando el GESPRO como fuente de datos en la investigación se obtiene la información referente a los datos²⁶ de las personas seleccionadas, con el fin de calcular los valores que presentan estos individuos en los índices de rendimiento respecto al trabajo (IRHT), la eficacia (IRHE), eficiencia (IRHF) y el aprovechamiento (IRHA). Este cálculo se realiza mediante las fórmulas correspondientes y planteadas en el Capítulo 1 de la presente investigación.

Por otro lado a los individuos seleccionados en la BC se les debe aplicar el instrumento NEO-PI²⁷ definido en el modelo FFM para evaluar el tipo de personalidad que presentan. Este instrumento contiene 300 afirmaciones y cada una de ellas debe responderse en una escala de cinco opciones²⁸ en función de la personalidad del individuo y donde solo debe seleccionarse una. Al procesar los resultados, el instrumento permite evaluar los rasgos de la personalidad que caracterizan a cada individuo en una escala de: Bajo, Normal y Alto.

²⁶ Datos obtenidos del GESPRO: cantidad de tareas cerradas con prioridad Alta, Normal y Baja, porcentaje de las tareas realizadas, tiempo estimado, tiempo dedicado, entre otros.

²⁷ Disponible en el sitio de Psicología Online (http://www.psicologia-online.com/test/inventario_personalidad_5_factores/)

²⁸ Opciones del instrumento: Totalmente de acuerdo, De acuerdo, Ni a favor ni en contra, En desacuerdo, Totalmente en desacuerdo.



Ilustración 13: Interfaz para cargar fichero de datos.

3. Seleccionar las variables

Una vez que el fichero a procesar sea cargado en la herramienta, es necesario realizar una selección y limpieza de los datos a través de la selección de variables para disminuir la dimensionalidad de los datos a procesar. Para llevar a cabo este proceso la herramienta utiliza la técnica CFS, la cual permite reducir el número de rasgos dentro de un conjunto inicial de 30 rasgos de personalidad, seleccionando aquellos que son “relevantes” para cada índice de rendimiento laboral.

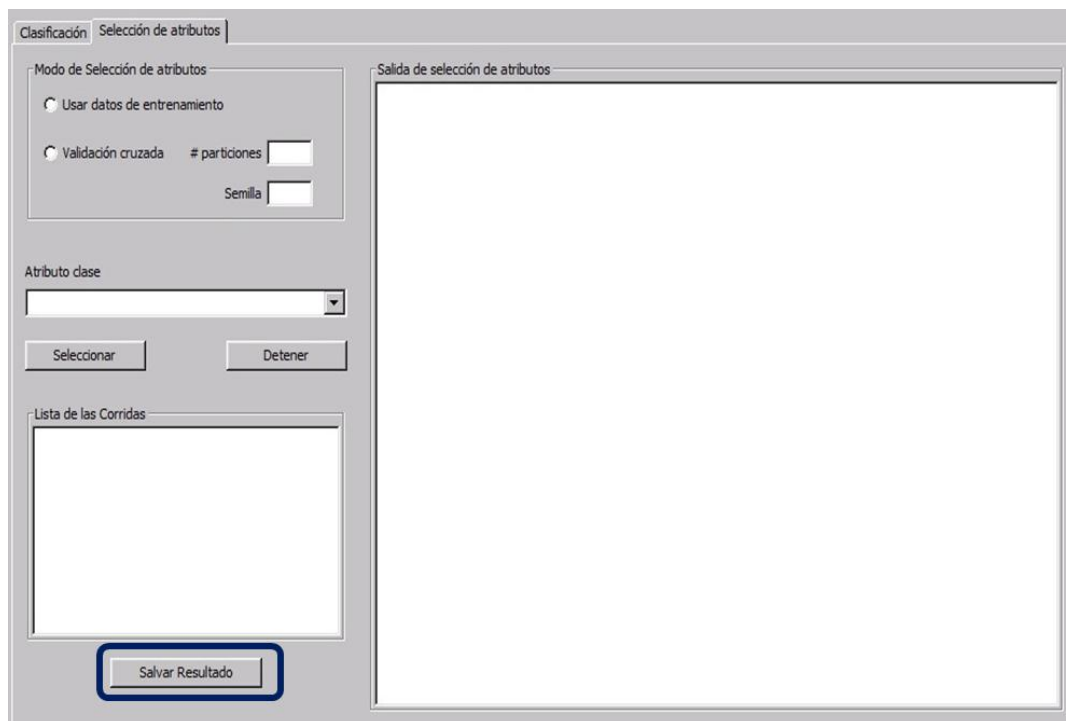


Ilustración 14: Interfaz para realizar la selección de variables.

La herramienta posibilita salvar el resultado de la selección de variables en un fichero .arff para utilizar estos rasgos relevantes posteriormente en la clasificación.

4. Clasificar el conjunto de datos

Para clasificar los datos debe cargarse el fichero .arff con los rasgos relevantes que fueron seleccionados para cada clase (entiéndase por clase los índices de rendimiento utilizados). Después de cargar el fichero la herramienta permite escoger el atributo clase para la clasificación dentro del conjunto de datos. Este sería el último paso para obtener el modelo generado por el algoritmo.

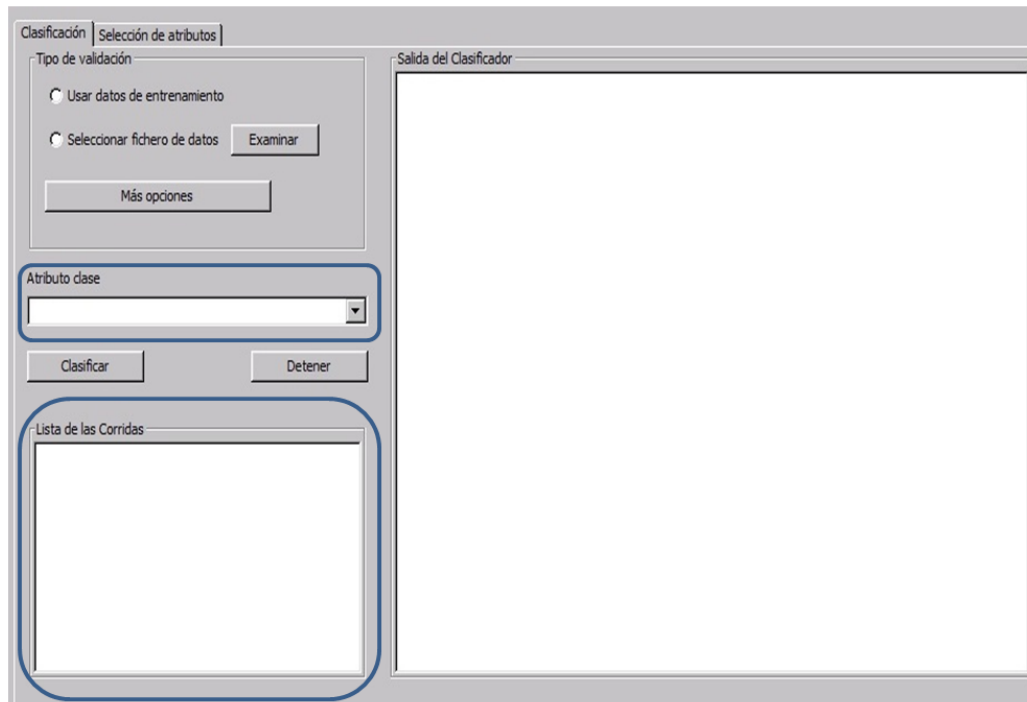


Ilustración 15: Interfaz para realizar la clasificación.

El modelo contiene el árbol de decisión acompañado de diferentes valores estadísticos que se calculan en el proceso de clasificación como son: la cantidad de Instancias correctamente e incorrectamente clasificadas, el error absoluto, el estadístico de Kappa, la tasa de verdaderos positivos, la tasa de falsos positivos y precisión.

La herramienta permite como otra forma de evaluación de un modelo de clasificación generar la matriz de confusión. Además brinda la posibilidad de realizar varias corridas, es decir, almacena los diferentes modelos que puedan ser generados, para que el usuario pueda saber en todo momento los resultados de las corridas que haya realizado. Los resultados de la aplicación del algoritmo pueden ser guardados por el usuario en caso que desee visualizarlo en otro momento.

3.2. VALIDACIÓN DE LA HERRAMIENTA

Para la validación de la herramienta se realizaron pruebas unitarias y de aceptación, facilitando detectar la mayor cantidad de defectos posibles para su debida corrección. Permitiendo de igual manera aumentar la calidad y cumplimiento de los requisitos implementados en la presente investigación. A continuación se detalla las pruebas aplicadas a la herramienta.

3.2.1. Pruebas unitarias

Las pruebas unitarias son desarrolladas por los programadores, dirigidas a probar o verificar la funcionalidad del código, probar las diferentes sentencias y validar que cumplan la función para las que fueron programadas proporcionando una medida de la complejidad de la implementación (Letelier, 2006).

➤ Prueba de Camino Básico

La prueba del camino básico es una técnica de prueba de caja blanca que permite 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. Para realizar esta prueba se utiliza la complejidad ciclomática de McCabe, que consiste en la ejecución de un conjunto de caminos independientes proporcionando una medición cuantitativa de la complejidad lógica de un programa, así como determinar el número de casos de prueba que se deben realizar para asegurar que se ejecuta cada sentencia al menos una vez (Pressman, 2001).

Los pasos que se siguieron para aplicar esta técnica son:

Paso 1: Obtener el grafo de flujo, a partir del diseño o del código del módulo.

Paso 2: Obtener la complejidad ciclomática del grafo de flujo.

Paso 3: Definir el conjunto básico de caminos independientes.

Paso 4: Preparar un caso de prueba por camino hallado en el paso anterior.

A continuación se ejemplificará la aplicación la prueba de camino básico de la funcionalidad *registrarVista* de la clase *PersonalidadModel*. Dado el segmento de código, se construye el grafo de flujo (Ver Ilustración 16) y posteriormente se realiza el cálculo de la complejidad ciclomática a partir de tres variantes representadas a continuación, las cuales arrojan el mismo resultado, condición necesaria para asegurar que el cálculo de la complejidad es correcto.

```

815 public boolean registrarVista(IObserverVista vista) {
816     boolean vistaActual=false;
817     for(int i=0;i<listaObserver.length();i++){
818         if(listaObserver.get(i).equals(vista)){
819             vistaActual=true;
820         }
821     }
822     if(vistaActual){
823         return false;
824     }
825     listaObserver.add(vista);
826     return true;
827 }

```

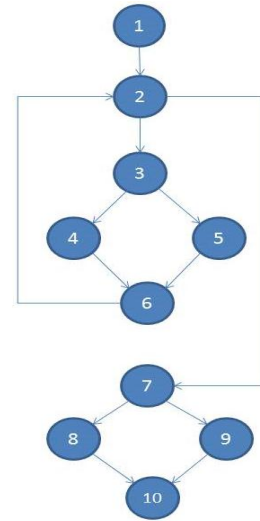


Ilustración 16: Segmento de código y grafo del flujo del método registrarVista ().

Para obtener los caminos independientes se calcula la complejidad ciclomática, la cual se puede obtener de tres formas (Pressman, 2001):

- El número de regiones del grafo de flujo coincide con la complejidad ciclomática.

Cantidad de regiones=4

- La complejidad ciclomática, $V(G)$, de un grafo de flujo G se define como $V(G)=A- N+2$ siendo A el número de aristas del grafo de flujo y N el número de nodos del mismo.

$$V(G) = 12 \text{ aristas} - 10 \text{ nodos} + 2 = 4$$

- La complejidad ciclomática, $V(G)$, de un grafo de flujo G se define como $V(G)=P+1$ donde P es el número de nodos predicado contenidos en el grafo de flujo G .

$$V(G) = 3 \text{ nodos predicados} + 1 = 4$$

Como complejidad ciclomática se obtiene el mismo valor (4) para cualquiera de las tres formas de calcularlo, por tanto la cantidad de caminos básicos que puede tomar el algoritmo en su ejecución es 4 (Ver Tabla 16).

Tabla 14: Caminos básicos del flujo.

No.	Camino
Camino básico # 1	1-2-3-4-6-2-7-8-10
Camino básico # 2	1-2-4-5-6-2-7-9-10

Camino básico # 3	1-2-7-8-10
Camino básico # 4	1-2-7-9-10

3.2.2. Pruebas de aceptación

El objetivo de las pruebas de aceptación es la evaluación del producto y la realización de una revisión de la documentación final. Se enfocan en las características generales y la funcionalidad de la herramienta, elementos visibles y revisables por el cliente (Pressman, 2001). Las pruebas de aceptación comprueban que las historias de usuario han sido implementadas correctamente al final de cada iteración. Se realizan los casos de prueba de aceptación como artefacto en la fase de pruebas, en el que se especifican los puntos a probar para garantizar la calidad del producto final con el cumplimiento de los requisitos especificados. A continuación se muestra el caso de prueba diseñado a la *HU Mostrar resultado de la clasificación*. El resto de los casos de pruebas asociados a las restantes HU pueden consultarse en el Anexo 7.

Tabla 15 : Descripción de Caso de prueba de aceptación *Mostrar resultado de la clasificación*.

Caso de prueba de aceptación	
Código de caso de prueba: CP7_ HU2.4	Nombre de la historia de usuario: Mostrar resultado de la clasificación.
Responsable de la prueba: Yoendry Betancourt Peña.	
Descripción de la prueba: Prueba de funcionalidad para permitirle al usuario visualizar el resultado de la aplicación del algoritmo al conjunto de datos de entrenamiento.	
Condiciones de ejecución: Debe haberse cargado el fichero con los datos de entrenamiento. Debe haberse configurado los parámetros de ejecución y escogido el atributo clase a clasificar. Debe haberse realizado la clasificación.	
Entrada/Pasos de ejecución: El usuario carga un fichero con los datos de entrenamiento en la herramienta desde un directorio local de la PC. Configura los parámetros de ejecución del algoritmo y escoge el atributo clase para clasificar. Realizar la clasificación.	
Resultado esperado: El sistema muestra el resultado de la aplicación del algoritmo al conjunto de datos de entrenamiento.	
Evaluación de la prueba: Prueba satisfactoria.	

Para validar que la salida emitida por la herramienta coincidiera con el resultado esperado se realizaron 10 pruebas funcionales para validar que el sistema funcionara correctamente, mostrando las salidas

correspondientes a cada escenario. De estas pruebas realizadas, en una primera iteración resultaron satisfactorias 7 de ellas, representando un 70% del total y 3 de ellas resultaron insatisfactorias, representando un 30%. En una segunda iteración de un total de 8 pruebas realizadas, resultaron todas satisfactorias, constituyendo un 100% de pruebas funcionales exitosas (Ver Ilustración 17).

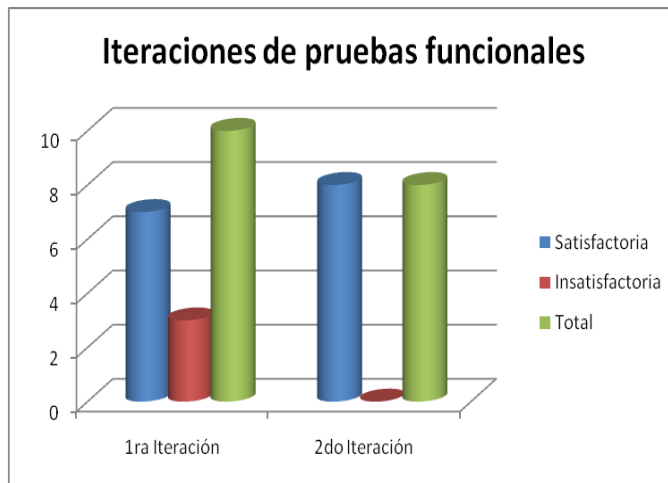


Ilustración 17: Resultados por cada una de las iteraciones de pruebas funcionales.

3.3. VALIDACIÓN DE LA PROPUESTA

Para conocer los resultados de la investigación y realizar el análisis de la influencia de los rasgos de personalidad en el rendimiento laboral, se decide validar la propuesta, utilizando el método de expertos.

Para validar la solución se realizó un caso de estudio con 150 individuos seleccionados aleatoriamente entre los proyectos del CEIGE. Se realizó el cálculo de los índices de rendimiento IRHT, IRHE, IRHF e IRHA y se evaluaron los rasgos que presentaban estos individuos para confeccionar la BC.

Se seleccionaron los atributos para reducir el número de rasgos y seleccionar aquellos relevantes para cada índice de rendimiento laboral. A continuación se muestra el resultado de la técnica de selección utilizada para cada índice de rendimiento y una gráfica con los resultados:

Tabla 16: Resultado de la técnica de selección aplicada.

Rasgos de personalidad más relevantes	IRHA	IRHE	IRHF	IRHT
Escala_de_Ansiedad	X	X	X	
Escala_de_Ira	X		X	
Escala_de_amigabilidad	X		X	X

Escala_de_gregarismo	X	X	X	X
Escala de nivel de actividad	X	X		
Escala_de_aventuras	X	X		X
Escala_de_liberalismo	X			
Escala_de_confianza	X			
Escala_de_prudencia	X	X	X	X
Escala_de_impulsividad			X	
Escala_de_vulnerabilidad		X	X	X
Escala_de_emoción_positiva			X	
Escala_de_sensibilidad_emocional			X	
Escala_de_capacidad_logro			X	
Escala_de_autodisciplina			X	X
Escala_de_búsqueda_emociones				X
Escala_de_autoeficacia				X
Escala_de_imaginación		X		
Total	9	7	11	8

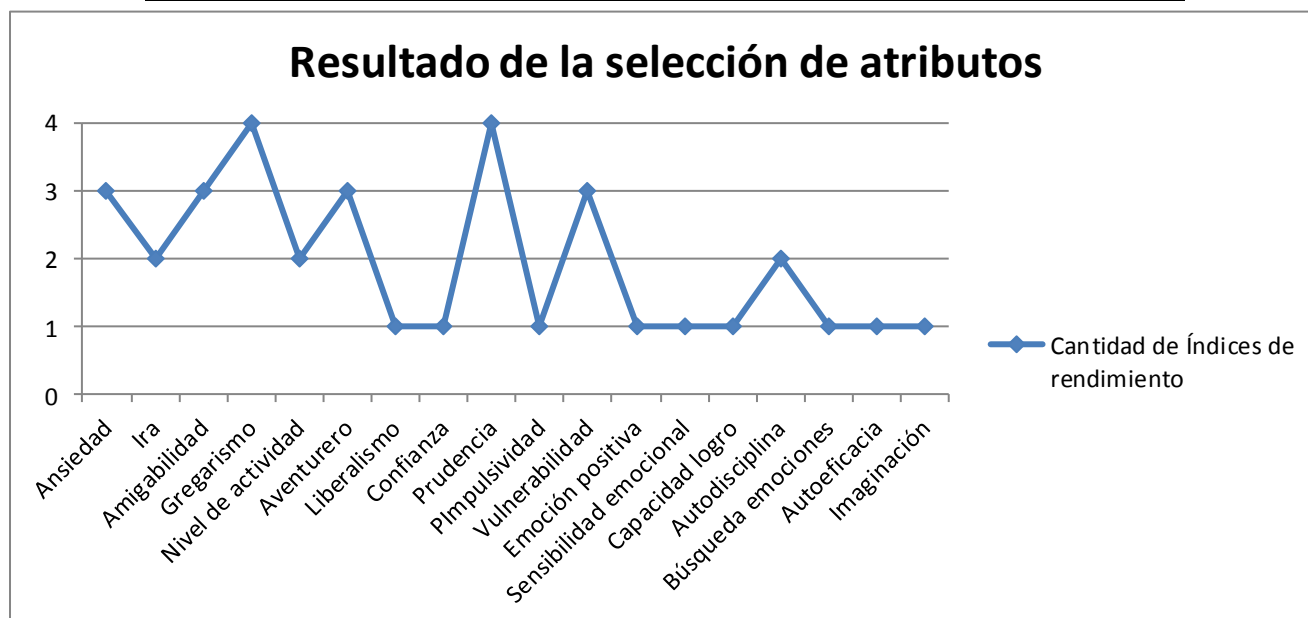


Ilustración 18: Rasgos relevantes a partir de la selección de atributos. Fuente: Elaboración propia.

Los resultados muestran que los rasgos relevantes para los índices de rendimiento son 18, representando un 60 % del total (30). Los rasgos que más prevalecen entre al menos tres de los cuatro índices de rendimiento son: ansiedad, amigabilidad, gregarismo, aventurero, prudencia y vulnerabilidad.

CAPÍTULO 3: RESULTADOS Y VALIDACIÓN DE LA PROPUESTA

Posteriormente se realizó la clasificación del conjunto de datos la cual arrojó los siguientes resultados:

Tabla 17: Resultados de la evaluación del conjunto de datos de entrenamiento para cada clase.

Clases	Instancias Correctamente	Clasificadas	Instancias Incorrectamente	Clasificadas	Estadístico Kappa	de	Error absoluto
IRHA	72	48%	78	52%	0.0611		0.4142
IRHE	76	50.7 %	74	49.3%	0.2276		0.3975
IRHF	102	68%	48	32 %	0.3286		0.219
IRHT	110	73.3 %	40	26.7 %	0.4003		0.2558

Tabla 18: Descripción de las precisiones por clase.

Clases	Criterios	Tasa de Verdaderos Positivos	Tasa de Falsos Positivos	Precisión
IRHA	Sub_utilizado	1	0.927	0.472
	Aprovechado	0.049	0.018	0.5
	Sobrecargado	0.049	0	1
IRHE	Bajo	0.273	0.075	0.6
	Medio	0.383	0.165	0.514
	Alto	0.78	0.538	0.484
IRHF	Atraso	0.333	0	1
	Atraso_no_comenzado	0.931	0.635	0.669
	En_tiempo	0.383	0.078	0.692
	Adelantado	0	0	0
IRHT	Bajo	0.4	0	1
	Medio	0.415	0.072	0.759
	Alto	0.935	0.569	0.569

Analizando los resultados de las tablas se puede observar que los índices de rendimiento que mejor clasificaron fueron el IRHF y IRHT, con el 68% y 73.3% de las instancias respectivamente. Además se muestra la tasa de verdaderos positivos, que detalla por clase el porcentaje de instancias con ese valor que son correctamente predichas y la tasa de falsos positivos que representa el porcentaje de instancias con otros valores que son incorrectamente predichas a ese valor aunque tenían otro. Igualmente con las medidas anteriores, se obtiene la precisión del modelo para cada criterio de las clases. Este porcentaje indica la proporción de aciertos del modelo obtenido.

Como resultado se genera un árbol de decisión por cada clase definida, mostrándose a continuación un ejemplo de árbol generado (clase IRHE). Los demás pueden ser visualizados en el Anexo 8.

```
J48 pruned tree
-----

Escala_de_aventuras = Bajo: Medio (1.0)
Escala_de_aventuras = Normal
| Escala_de_gregarismo = Bajo: Bajo (2.0)
| Escala_de_gregarismo = Normal
| | Escala_de_imaginación = Bajo: Alto (0.0)
| | Escala_de_imaginación = Normal
| | | Escala_de_vulnerabilidad = Bajo: Medio (27.0/14.0)
| | | Escala_de_vulnerabilidad = Normal: Alto (58.0/35.0)
| | | Escala_de_vulnerabilidad = Alto: Medio (0.0)
| | | Escala_de_vulnerabilidad = Ninguno: Medio (0.0)
| | Escala_de_imaginación = Alto: Alto (26.0/11.0)
| | Escala_de_imaginación = Ninguno: Alto (0.0)
| Escala_de_gregarismo = Alto: Medio (7.0/3.0)
| Escala_de_gregarismo = Ninguno: Alto (0.0)
Escala_de_aventuras = Alto
| Escala_de_vulnerabilidad = Bajo: Bajo (18.0/8.0)
| Escala_de_vulnerabilidad = Normal: Alto (10.0/3.0)
| Escala_de_vulnerabilidad = Alto: Alto (0.0)
| Escala_de_vulnerabilidad = Ninguno: Alto (1.0)
Escala_de_aventuras = Ninguno: Alto (0.0)
```

Ilustración 19: Árbol generado para el índice de rendimiento IRHE.

Tras un análisis de los árboles generados para una mejor interpretación de los resultados, se define un conjunto de reglas para cada clase, que dan respuesta a los comportamientos que debe tener cada individuo para cumplir con los diferentes criterios establecidos por clase. Las reglas definidas para las clases clasificadas a partir del resultado del algoritmo de clasificación se pueden visualizar en el Anexo 9.

Como parte de la validación se utilizó además una BC confeccionada por 50 personas totalmente distinta a las 150 seleccionadas anteriormente en la investigación. El objetivo es obtener cuántos individuos en esta BC cumplieran con las reglas generadas para cada clase en su mejor criterio (Ver Tabla 21).

Tabla 19: Análisis de las reglas definidas a partir de la aplicación del algoritmo para el mejor criterio de cada clase.

Reglas generadas para IRHA en su mejor criterio	Reglas generadas para IRHE en su mejor criterio
Ansiedad=Alto.	Aventuras = Normal, Alto, Ninguno. Gregarismo=Normal, Ninguno.

	Imaginación=Bajo, Normal, Alto. Vulnerabilidad=Normal, Alto, Ninguno.
Reglas generadas para IRHF en su mejor criterio	Reglas generadas para IRHT en su mejor criterio
Ira = Bajo, Normal, Alto. Autodisciplina = Normal. Gregarismo=Normal, Alto. Amigabilidad=Normal, Alto. Emoción positiva= Alto. Vulnerabilidad=Bajo. Impulsividad = Bajo, Normal. Capacidad de logro=Bajo, Normal, Ninguno.	Búsqueda emociones = Normal, Alto, Ninguno. Gregarismo = Normal, Alto, Ninguno. Autoeficacia = Bajo, Normal, Alto, Ninguno. Aventuras=Bajo, Alto, Ninguno. Amigabilidad= Alto. Prudencia=Bajo, Ninguno.

Para evaluar los resultados se seleccionó en esta BC los individuos que presentaban los mejores criterios de rendimiento en cada una de las cuatros clases. Con este resultado se seleccionaron los individuos que además de tener los mejores criterios de rendimiento por cada clase, cumplieran con las reglas generadas en el algoritmo. Este estudio permitió obtener los siguientes resultados:

Tabla 20: Resultados del análisis de la BC de 50 personas.

Índices	Cantidad de Individuos que cumplieron con los mejores criterios por cada clase		Cantidad de individuos que cumplieron con las reglas condicionantes para los mejores criterios por cada clase	
	Cantidad	Por ciento que representa del total	Cantidad	Por ciento que representa del total
IRHT	29 personas	58 %	20	69%
IRHE	36 personas	72%	28	78%
IRHA	4 personas	8%	1	25%
IRHF	6 personas	12%	6	100%

Por otro lado se utilizó el método de expertos con el fin de obtener criterios un poco más condicionados por la experiencia y no desde el punto de vista estadístico. Para ello se seleccionaron seis expertos, considerando en el caso de los psicólogos al menos cinco años de experiencia, y en los directivos y subordinados más de cinco años de experiencia. La encuesta realizada puede ser consultada en el Anexo 10, y los expertos encuestados fueron:

- 1- Irene Alejandra Cordero Leyva, Graduada en Psicología en la Universidad de Oriente.
- 2- Oralys Estrada García, Graduada en Psicología en la Universidad de Oriente.

- 3- Ailia Parra Fernández, Directora del CEIGE, Facultad 3.
- 4- Osnier Ramírez Alea, Desarrollador Departamento Banco, CEIGE, Facultad 3.
- 5- Yadira Calimano Meneses, Profesora, Facultad 3.
- 6- Ailenis Macías, Profesora, Facultad 3.

Este método permitió identificar los rasgos relevantes de acuerdo al criterio de los expertos, basados en las experiencias con los RRHH en el área en la que se desempeñan. Una comparación entre los datos arrojados por este método y los resultantes tras aplicar el algoritmo demostró que coincidió en un 59 % los rasgos más relevantes seleccionados por ambas partes.

Tabla 21: Resultados arrojados por ambas técnicas.

Expertos		Algoritmo	
Nivel de actividad	Gregarismo	Ansiedad	Amigabilidad
Curiosidad intelectual	Aventura	Aventuras	Emoción positiva
Cooperación	Confianza	Gregarismo	Impulsividad
Valores éticos	Capacidad de logro	Imaginación	Capacidad de logro
Ansiedad	Búsqueda de Emociones	Vulnerabilidad	Búsqueda
Amigabilidad	Autodisciplina	Ira	emociones
Liberalismo	Imaginación	Autodisciplina	Autoeficacia
Autoeficacia	Emoción positiva		Prudencia
Obediencia			

Los resultados anteriormente expuestos son alentadores, ya que permiten identificar un patrón ideal de la persona con un mayor nivel de precisión (Ver Anexo 11). Analizando las BC utilizadas en su conjunto se identifica que solo ocho personas del CEIGE cumplieron con este patrón. Sin embargo estos resultados no son absolutos, ya que pueden estar condicionados por el ruido en la información e inconsistencias en el proceso de recopilación de los datos. Además, desde el punto de vista psicológico, el ser humano es impredecible y su comportamiento puede verse afectado por otros factores externos al proceso.

3.4. Análisis de los resultados

Como resultado de la presente investigación se identificaron los rasgos de la personalidad que caracterizan a los individuos del CEIGE seleccionados en la muestra, información relevante que permite a los directivos conocer más a fondo las características de sus subordinados. A partir de la confección de los árboles de decisión una persona sin conocimientos de DM podrá identificar en la herramienta,

fácilmente los valores de los rasgos (reglas) que las personas deben cumplir para obtener índice de rendimiento bajos, medios y altos.

Las reglas generadas en la clasificación servirán de apoyo a los directivos para la toma de decisiones en la selección de individuos, ya que se podrá predecir los diferentes índices de rendimiento que tiene cada individuo a partir de los rasgos y en qué medida influye cada uno de ellos. La obtención de las reglas permitió definir un patrón de persona ideal que cumple con los mejores criterios para los cuatro índices de rendimiento. Este patrón sirve para determinar qué características tiene que tener un individuo para ser totalmente eficiente y obtener altos índices de rendimiento. Permitted además aportar nuevos elementos en el proceso de adquisición de individuos para integrar los proyectos y conformar equipos eficientes.

3.5. Conclusiones del capítulo

Luego de finalizado este capítulo se concluyó que:

- Con las pruebas unitarias se corrigieron los posibles errores en la codificación y la obtención de un prototipo funcional acorde a los requisitos definidos, dando así un valor de calidad añadido a la herramienta. Se realizó una validación satisfactoria de la herramienta aplicando pruebas de aceptación al final de cada iteración y las revisiones durante todo el desarrollo.
- A partir de las pruebas realizadas, se pudo evidenciar que todas las funcionalidades quedaron implementadas, independientemente de las inconformidades encontradas.
- El análisis de los resultados arrojados por la aplicación del algoritmo al conjunto de datos de entrenamiento y el criterio de experto, permitieron obtener los rasgos que más influyen en el rendimiento laboral.

CONCLUSIONES GENERALES

Para la realización del presente trabajo de diploma se plantearon un conjunto de tareas que posibilitaron el desarrollo y cumplimiento de los objetivos propuestos, así como la definición de las siguientes conclusiones:

- Se elaboró el marco teórico de la investigación permitiendo obtener conocimiento referente a los árboles de decisión, herramientas de Minería de Datos y algoritmos de clasificación para el descubrimiento del conocimiento.
- Se identificaron y describieron los requisitos funcionales de la herramienta a través de las historias de usuarios.
- La aplicación de los patrones en el diseño de las clases garantizó una comprensión exacta sobre las funcionalidades a implementar.
- La implementación de la herramienta basada en algoritmos de árboles de decisión permitió evaluar la influencia de los rasgos de la personalidad en el rendimiento laboral.
- La aplicación de métricas del diseño y pruebas de calidad aplicadas, tanto a la herramienta como a los artefactos generados, permitieron validar la solución propuesta.

RECOMENDACIONES

Se recomienda para futuros trabajos:

- Aplicar el conjunto de datos a otros algoritmos de clasificación de árboles de decisión para comparar resultados.
- Continuar con la implementación de otras gráficas que aporten valor agregado a la solución.

BIBLIOGRAFÍA

- Adele H. T., Kam H. P. 2013.** *Personality Type, Learning Modalities and Academic Performance in Undergraduate Engineering* . 2013.
- Aluja, A., & Blanch, A. 2002.** *Análisis descriptivo y comparativo del 16PF-5 en muestras americanas y española*. 2002.
- Alvarez, Miguel Angel. 2001.** *¿Qué es Java?* 2001.
- André, M.A. 2009.** *Un modelo para la asignación de recursos humanos a equipos de proyectos de software, in Centro de Estudios de Ingeniería y Sistemas*. 2009.
- Belbin Associates . 2012.** BELBIN Associates. *BELBIN Associates*. [Online] 2012. [Cited: mayo 10, 2014.] <http://www.Belbin.es/rte.asp?id=466>.
- Belbin Associates . 2010.** Belbin.com. *Belbin.com*. [Online] 2010. [Cited: 02 27, 2014.] <http://www.Belbin.com/rte.asp?id=7>.
- Berzal, Juan Carlos Cubero & Fernando. 2013.** *Herramientas de Minería de Datos*. Granadas : s.n., 2013.
- Bozal, Guillén G. Carlos y Rosio Guil. 2000.** *Psicología del Trabajo para las Relaciones Laborales*. Madrid, España. : :McGraw Hill/Interamericana de España, S.A.U., 2000.
- Byars, Lioyd L. y Rue, W. L. 1996.** *Gestión de Recursos Humanos*. España : Mosby, 1996. p. 583.
- Cano de Amo, J.R. 2004.** In Departamento de Ciencias de la Computación e Inteligencia Artificial. . Granada: Universidad de Granada. : s.n., 2004.
- Cardenas, Hector. 2012.** *DESARROLLO DE LOS RECURSOS HUMANOS EN LAS ORGANIZACIONES ECONOMICAS INTERNACIONALES DE LA CUENCA DEL PACIFICO*. 2012.
- Chapman, A. 2011.** businessballs.com. *businessballs.com*. [Online] 2011. [Cited: 02 26, 2014.] <http://www.businessballs.com/personalitystylesmodels.htm>.
- Chavez, Joseph Gonza. 2010.** jgcprogramacion. *jgcprogramacion*. [Online] agosto 01, 2010. [Cited: mayo 10, 2014.] <http://jgcprogramacion.blogspot.com/2010/08/test.html>.
- Chiavenato, I. 2000.** *Administración de Recursos Humanos*. . 2000.
- Cockburn, Alistair. 2005.** *Agile Software Development*. 2005.
- Cordero, F.C. 2008.** *Medición del Rendimiento Laboral Individual*. 2008.
- Criado, Laura HernanGómez. 2012.** *PSICOLOGÍA DE LA PERSONALIDAD Y DIFERENCIAL*. 2012.
- Descuadrando. 2013.** descuadrando. *descuadrando*. [Online] 2013. [Cited: mayo 15, 2014.] http://descuadrando.com/%C3%81rbol_de_decisi%C3%B3n.
- Díaz, Darel Camps. 2007.** *Implementación de algoritmos*. 2007.
- Ecured. 2014.** Ecured. *Ecured*. [Online] 2014. [Cited: mayo 10, 2014.] http://www.ecured.cu/index.php/Programaci%C3%B3n_Extrema_o_XP.
- EcuRed. 2014.** EcuRed. *EcuRed*. [Online] 2014. [Cited: mayo 15, 2014.] http://www.ecured.cu/index.php/M%C3%A9trica_de_dise%C3%B1o.
- Espinoza, Fátima. 2014.** slideshare.net. *slideshare.net*. [Online] 2014. [Cited: 02 08, 2014.] <http://www.slideshare.net/fatimaespinoza/rasgos-y-tipos-de-personalidad>.

- Feist, Feist and. 2009.** *Theories of Personality*. 2009.
- Feldt, R., Angelis, L., Torkar, R., & Samuelsson, M. 2010.** "Links Between the Personalities, Views and Attitudes of Software Engineers". s.l. : Information & Software Technology,, 2010. pp. 611-624. Vol. 56.
- Gamma, Erich. 1995.** *Design Patterns: Elements of Reusable Object-Oriented Software (APC)*. 1995.
- García, José A. Lugo. 2012.** *Modelo para el control de la ejecución de proyectos basado en indicadores y lógica borrosa*. La Habana : s.n., 2012.
- García, José Manuel Molina López y Jesús. 2006.** *TÉCNICAS DE ANÁLISIS DE DATOS* . 2006.
- Genbetadev. 2013.** genbetadev. *genbetadev*. [Online] 2013. [Cited: mayo 02, 2014.] <http://www.genbetadev.com/herramientas/netbeans-1>.
- Gómez, Monte Manuel. 2003.** *Minería de texto empleando la Semejanza entre Estructuras Semánticas*. 2003.
- Hall, Mark A. 1998.** *Benchmarking Attribute Selection Techniques for Data Mining*. 1998.
- . **1998.** *Correlation-based Feature Selection for Machine Learning*. New Zealand : s.n., 1998.
- Hernández, Faustino Pérez. 2007.** *Aplicación de métodos de selección de atributos para determinar factores relevantes en la evaluación nutricional de los niños*. 2007.
- Herreras, Esperanza Bausela. 2005.** *MODELO ALTERNATIVOS DE EVALUACIÓN DE LA PERSONALIDAD: MODELO DE LOS CINCO FACTORES, MODELO 16 PF Y OTROS*. 2005.
- Informatizate. 2011.** Informatizate. *Informatizate*. [Online] 2011. [Cited: mayo 1, 2014.] http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html
..
- IEEE. 2006.** "IEEE Standard Glossary of Software Engineering Terminology". [Online] mayo 24, 2006. [Cited: enero 13, 2014.] http://standards.ieee.org/reading/ieee/std_public/description/se/610.12-1990_desc.html. Std 610.12-1990.
- Jacas, Danier Marante. 2009.** *Aplicación de la minería de datos para la exploración y detección de patrones delictivos*. La Habana : s.n., 2009.
- Jano, D. & Ortiz, S. 2005.** *Determinación de los factores que afectan al rendimiento académico en la educación superior*. XII Jornadas de la Asociación de Economía de la Educación, Oviedo. 2005.
- Jung, C. G. 1968.** *Analytical Psychology: Its Theory and Practise*. s.l.: The Tavistock Lectures, 1968.
- Jeffries, R., Anderson, A., Hendrickson, C. 2001.** "Extreme Programming Installed". Addison-Wesley.
- Keirsey, D. 1996.** Keirsey.com. *Keirsey.com*. [Online] 1996. [Cited: mayo 15, 2014.] <http://www.keirsey.com/sorter/register.aspx>.
- KNIME. 2013.** KNIME. *KNIME*. [Online] 2013. [Cited: mayo 05, 2014.] <http://www.knime.org/>.
- Kohavi R, John GH. 1997.** *Wrappers for feature subset selection*. . 1997.

- Landeta, Juan Manuel Izar. 2011.** *Factores que afectan el desempeño académico de los estudiantes de nivel superior en Rioverde.* Instituto de Investigaciones en Educación, Universidad Veracruzana. San Luis Potosí, México : s.n., 2011.
- Larman, Craig. 1999.** *UML y Patrones. Introducción al análisis y diseño orientado a objetos.* [ed.] Pablo Eduardo Roig Vlkquez. Mexico : Dawn Speth White, 1999.
- Letelier, Patricio y Penadés, M^a Carmen. 2006.** *Métodologías ágiles para el desarrollo de software: eXtreme Programming (XP).* Valencia: Universidad Politécnica de Valencia.
- Lingjaerde, Regine Foreland, A. y O.Engvik, H. 2001.** *Personality structure in patients with winter depression, assessed in a depression – free state according to the five – factor model of personality.* 2001.
- Llopis, Vicente Mars. 2000.** *psicologia-online.com/. psicologia-online.com/.* [Online] 2000. [Cited: 03 3, 2014.] <http://www.psicologia-online.com/pir/definicion-de-rasgo.html>.
- Lozano, Fernando. 2013.** [Online] 2013. [Cited: mayo 05, 2014.] <http://www.fernandolozanopajaron.com/documentos/FernandoLozano-Memoria.pdf>.
- Lugo, García José A. 2012.** *Modelo para el control de la ejecución de proyectos basado en indicadores y lógica borrosa.* La Habana : s.n., 2012.
- Luján, Irene Rodríguez. 2009.** *Selección de variables mediante programación cuadrática.* Madrid : s.n., 2009.
- Magazine, Estrategia. 2014.** *estrategiamagazine. estrategiamagazine.* [Online] 2014. [Cited: mayo 15, 2014.] <http://www.estrategiamagazine.com/administracion/la-tecnica-del-arbol-para-la-toma-de-decisiones-efectivas-herramientas/>.
- Martineaux, Karina Silveira. 2008.** *Comparación de algoritmos de clasificación y agrupamiento aplicando técnicas de Minería de datos.* La Habana : s.n., 2008.
- McCrae, R. R., & Costa, P. T., Jr. 1997.** *Personality trait structure as a human universal.* s.l. : American Psychologist, 1997. pp. 509-516.
- . 1996. *Toward a new generation of personality theories: Theoretical contexts for the five-factor model.* New York: Guilford. : s.n., 1996. pp. 51-87.
- Mondy, R.W. y Noe, F. 2005.** *Administración de los recursos humanos.* México : s.n., 2005.
- Morate, Diego García. 2013.** *MANUAL DE WEKA.* 2013.
- Muñoz, Ana. 2014.** *Las cinco grandes dimensiones de la personalidad.* 2014.
- Myers, Isabel Briggs, et al. 1998.** *MBTI Manual (A guide to the development and use of the Myers Briggs type indicator).* s.l. : Consulting Psychologist Press., 1998.
- Netbeans. 2013.** *netbeans. netbeans.* [Online] 2013. [Cited: mayo 02, 2014.] <https://netbeans.org/>.
- Newstrom, Davis K. y Jhon W. 2000.** *Comportamiento Humano en el Trabajo.* México:D.F McGraw Hill/Interamerica : S.A de C.V., 2000.
- Pascarella, E. & Terenzini P. 1983.** *Predicting voluntary freshman year persistence/withdrawal behavior in a residential university: a path analytic validation of Tinto's model.* s.l. : Journal of Educational Psychology, 1983. pp. 215-226.
- Pedraza, Esperanza., Amaya Glenys. y Conde Mayrene. 2010.** *scielo. scielo.* [Online] septiembre 2010. http://www.scielo.org.ve/scielo.php?pid=S1315-95182010000300010&script=sci_arttext.

Pérez, E. M. C. C. B. 2004. *Adaptación del Inventario de Personalidad 16PF-IPIP a un Contexto de Orientación. Estudio Preliminar.* Córdoba: Universidad Nacional de Córdoba : s.n., 2004.

Piñero Pérez, P. & et.al. 2010. *Paquete de Herramientas para la Gestión de Proyectos GESPRO.* Cuba : s.n., 2010.

PMI. 2009. *Guía de los Fundamentos para la Dirección de Proyectos.* Pennsylvania : Project Management Institute, 2009. ISBN:978-1-933890-72-2..

Ponte, Enrique Bonsón. 2013. ciberconta. *ciberconta.* [Online] 2013. [Cited: mayo 15, 2014.] <http://www.ciberconta.unizar.es/biblioteca/0007/arboles.html>.

Pressman, R. S. 2005. *Ingeniería del software.* Ciudad Habana : s.n., 2005.

Pressman, Roger S. 2001. *Ingeniería de Software, un enfoque práctico.* s.l. s.l. : McGraw-Hill, 2001.

—. **2010.** *Software Engineering a Practitioner´s Approach, Seventh Edition.* United States : Mc Graw Hill, Higher Education, 2010.

Prieto, M. (2010). Método Cuantitativo para Integración y Comparación de Grupos de Trabajo de Instalación y Desarrollo de Software 1-29. <http://www.manuelprieto.com/publicaciones/gruposDeTrabajoDeInstalacionYDesarrolloDeSoftware.pdf>.

Psicologica, Terapia. 2014. *Terapia Psicologica.com.mx. Terapia Psicologica.com.mx.* [Online] Centro de Desarrollo de Habilidades Psicolingüísticas (CEDHAP), 2014. <http://TerapiaPsicologica.com.mx>.

Quinlan, J. R. 1993. *C4.5: Programs for Machine Learning.* s.l. : Morgan Kaufmann Publishers,, 1993.

Rajendran, M. 2005. *Analysis of team effectiveness in software development teams working on hardware and software environments using Belbin Self-perception Inventory.* 2005.

RapidMiner. 2014. RapidMiner. *RapidMiner.* [Online] 2014. [Cited: Mayo 05, 2014.] <http://rapidminer.com/>.

Revelle, W. 1995. "Personality processes. Annual Review of Psychology.". <https://personality-project.org/revelle/publications/revelle.scherer.08.pdf>.

Rodríguez, Meylin. 2011. *Modelo de evaluación del rendimiento de los estudiantes en los proyectos de desarrollo de software del CEIGE.* LaHabana : s.n., 2011.

Rodríguez, Corbea Maite. 2007. LA METODOLOGÍA XP APLICABLE AL DESARROLLO DEL SOFTWARE EDUCATIVO EN CUBA. La Habana: s.n., 2007.

Salazar, Carlos T. 2010. *FACTORES CRÍTICOS QUE AFECTAN EL RENDIMIENTO LABORAL.* Venezuela : s.n., 2010.

Saldaña, J.F.REYES AND FLORES, R.GARCÍA. 2005. *El proceso de descubrimiento de conocimiento en base de datos.* s.l. : Ingenierías, 2005.

Sánchez, José M. Peña. 2001. *ARQUITECTURA DISTRIBUIDA DE CONTROL PARA SISTEMAS CON CAPACIDADES DE DATA MINING.* Madrid : s.n., 2001.

Seddigi, Z.S., L.F. Capretz and D. House. 2009. *A multicultural comparison of engineering students: Implications to teaching and learning.* 2009.

SEI. 2010. *CMMI for Dev v1.3.* . Pittsburgh: Carnegie Mellon University : s.n., 2010.

Silvente, Vanesa Berlanga. 2013. *Cómo aplicar árboles de decisión en SPSS*. Barcelona : s.n., 2013.

Soto, Francia Celis de. 2006. *La gestión de recursos humanos en las organizaciones de servicio*. Venezuela : s.n., 2006.

Stevens, K.T. and S.M. Henry. 2002. radford.edu. *radford.edu*. [Online] 2002. [Cited: 02 27, 2014.] <http://www.radford.edu/~kstevens2/>.

Stiven, Elizabeth Rodríguez. 2012. *MODELO PARA LA EVALUACIÓN DELA COMPOSICIÓN DE EQUIPO DE PROYECTOS INFORMÁTICOS*. 2012.

Takeyas, Bruno López. 2005. itnuevolaredo.edu.mx. *itnuevolaredo.edu.mx*. [Online] 2005. [Cited: mayo 10, 2014.] http://www.itnuevolaredo.edu.mx/takeyas/Apuntes/Inteligencia%20Artificial/Apuntes/tareas_alumnos/C4.5/C4.5%282005-II-B%29.pdf.

Tinto, V. 1989. *Definir la deserción: Una cuestión de perspectiva*. 1989. pp. 33-51.

Universia. 2014. Universia España. *Universia España*. [Online] 2014. <http://www.universia.es>.

Valentín, Ing. Eliana Bárbara Ril. 2012. *Descubrimiento de conocimiento a partir de lecciones aprendidas documentadas en los procesos de cierre de proyectos informáticos*. La Habana : s.n., 2012.

Vega, Margarita Chiang. 2010. *Como influye la satisfacción laboral sobre el desempeño*. 2010.

Visual Paradigm. 2014. Visual Paradigm. *Visual Paradigm*. [Online] 2014. [Cited: mayo 02, 2014.] <http://www.visual-paradigm.com/>.

WEKA. 2013. WEKA. *WEKA*. [Online] 2013. [Cited: mayo 05, 2014.] <http://www.cs.waikato.ac.nz/ml/WEKA/>.

Wilson, Leslie B. 1993. *Comparative Programming Languages*. Addison-Wesley . 1993.

Wordpress. 2012. articulosbm.files.wordpress.com. *articulosbm.files.wordpress.com*. [Online] 2012. [Cited: mayo 15, 2014.] <http://articulosbm.files.wordpress.com/2012/03/modelos-de-personalidad-the-big-five.pdf>.

XP, Extreme Programming. 2006. Extreme Programming: A gentle introduction. [Online] Febrero 17, 2006. [Cited: Mayo 17, 2014.] <http://www.extremeprogramming.org/index.html>.

Young, C. G., Quilty, L. C., & Peterson, J. B. 2007. *Between facets and domains: 10 aspects of the big five*. s.l. : Journal of Personality and Social Psychology,, 2007. pp. 880–896.