



Universidad de las Ciencias Informáticas

Facultad 2

Título: Herramienta de apoyo en la Evaluación de Impacto Ambiental.

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

Autoras:

Katia Ramírez Bruzón.

Lien Noa Jiménez.

Tutores:

Ing. David Rodríguez Rodríguez.

Ing. Yoanny Torres Rubio.

La Habana, 2013

“Año 55 del Triunfo de la Revolución”

Pensamiento

A portrait of Simón Bolívar, a Venezuelan military leader and statesman. He is depicted from the chest up, wearing a red military jacket with gold embroidery and a blue cape. He has a serious expression and is looking slightly to the left. The background is a plain, light color.

“Para el logro del triunfo siempre ha sido indispensable pasar por la senda de los sacrificios”.

Simón Bolívar

Declaración de Autoría

Declaramos ser las autoras del presente trabajo de diploma, reconociendo al Centro de Telemática de la Universidad de las Ciencias Informáticas el derecho de hacer uso del mismo en su beneficio.

Para que así conste firman la presente a los ____ días del mes de _____ del año _____.

Lien Noa Jiménez

Firma Autora

Katia Ramírez Bruzón

Firma Autora

Ing. David Rodríguez Rodríguez

Firma Tutor

Ing. Yoanny Torres Rubio

Firma Tutor

Dedicatoria

Katia

Dedico este trabajo a mi madre, la persona que más quiero en el mundo y solo espero que algún día pueda estar tan orgullosa de mí como yo lo estoy de ella.

Lien

Dedico esta tesis a mis abuelos, a mis hermanos y en especial a mi madre, porque siempre me apoyaron y depositaron toda su confianza en mí.

Agradecimientos

Lien

A mi madre, que es mi ejemplo a seguir, por confiar siempre en mí y en las cosas que puedo hacer y lograr, por brindarme todo su apoyo, por enseñarme que los objetivos para lograrlos requieren de sacrificios y por darme todo su amor y entrega.

A mi abuela, por todos los consejos que me ha brindado, porque me ha enseñado que siempre hay que pensar en el mañana, por todos sus mimos y por confiar siempre en mí.

A mi abuelo, porque supo ser padre y abuelo al mismo tiempo, por darme siempre todo su apoyo y por sus consejos.

A mis hermanos Liván y Luis Manuel por cuidarme siempre, por formar parte de mi vida y por ser los mejores hermanos del mundo.

A mi tío Sandor porque sé que podré contar con él para lo que necesite.

A Pedrín, el padre de mi hermano menor, porque a pesar de no tener la misma sangre me acogió como su hija y se comportó como el padre que no tuve, por cumplir todos mis caprichos y por su cariño.

A mi amiga y compañera de tesis Katia, porque fue una suerte haberla conocido, porque a pesar de todos mis errores ella siempre estuvo presente cada vez que la necesité. Le agradezco por todos los momentos que pasamos juntas en la universidad, porque a pesar de que a veces se comporta como una niña mimada, aunque no lo crea, me ha servido de ejemplo en muchas ocasiones.

A mi amiga Janet de Camagüey, que estuvo conmigo los dos primeros años de la universidad, por apoyarme siempre, porque a pesar de que éramos muy diferentes nos hicimos muy buenas amigas.

A todas mis amistades de la universidad y compañeros de aula, que de una forma u otra me apoyaron siempre y con las cuales compartí momentos inolvidables.

A todos los profesores que me impartieron clase desde mi primer año de la carrera, pero en especial a Baby que más que una profesora ha sido como una amiga.

A mis Tutores por el apoyo brindado durante el desarrollo del trabajo.

¡A todos muchas gracias!

Katia

Agradezco primeramente a mis padres por ser los mayores responsables de la persona en que me he convertido, por darme siempre su apoyo y su amor, por ser mi inspiración y los mejores padres que se puedan pedir.

A mis hermanitos, a mis tías Damaris y Dalquis, a mis abuelos, a mis primos y a toda mi familia en general por su cariño, por enseñarme a ser mejor persona y por su apoyo que ha sido tan importante durante toda mi vida.

A mi novio Daniel por haber estado siempre conmigo ayudándome y dándome fuerzas para seguir en los momentos más difíciles.

A mis tutores por su tiempo y por su dedicación, a Yoanny por su paciencia y por hacer suyo este trabajo y ayudarnos siempre que lo necesitamos.

A los profesores que me han ayudado durante toda la carrera, teniendo una parte importante en mi formación, en especial a los ingenieros Deivis Ricardo Alvarez, Erick Perez y Ernesto Melian por su ayuda con este trabajo.

A mis amigos por hacer mi estancia aquí más fácil, por hacerme sentir en familia, por todo su apoyo, en especial a Dailenka, Elizabeth y Nayilet, por estar siempre ahí cuando las necesito.

A Lien mi compañera de tesis, por hacer más fácil este trabajo por su ayuda inigualable, y a Lien mi amiga, mi hermana, por sus consejos, por su apoyo, por todo lo que me ha enseñado en todos estos años.

¡A todos simplemente muchas gracias!

Resumen

La Evaluación de Impacto Ambiental es un procedimiento jurídico-administrativo. La misma tiene por objetivo la identificación, predicción e interpretación de los impactos ambientales que un proyecto o actividad produciría en caso de ser ejecutado, así como la prevención, corrección y valoración de los mismos.

El Centro de Estudios de Química Aplicada de la Universidad Central de las Villas, es una entidad facultada para la realización de la Evaluación de Impacto Ambiental. Los especialistas del centro brindan este servicio a las diferentes empresas e instituciones que lo requieran, haciendo uso de la metodología propuesta por Vicente Conesa en la que se realizan evaluaciones simplificadas y detalladas. Dentro de la evaluación simplificada se encuentra la valoración cualitativa de impactos, que tiene como salida principal la Matriz de Importancia de Impacto. Dicha metodología parte de la evaluación de 10 símbolos que poseen naturaleza lingüística y numérica. En el caso de los símbolos con naturaleza lingüística pueden presentar cierta dificultad para ser representados con valores numéricos exactos por parte de los Expertos que los evalúan, introduciendo cierto grado de incertidumbre basado en sus propias percepciones de la realidad.

En la siguiente memoria se describió el desarrollo de una aplicación informática que emplea un nuevo modelo sobre valoración heterogénea (tanto lingüística como numérica) para la conformación de la Matriz de Importancia de Impacto, haciendo uso del Enfoque Lingüístico Difuso basado en el Modelo de Representación Lingüística de 2-tuplas.

Palabras claves: Evaluación de Impacto Ambiental, símbolos, heterogénea, incertidumbre, Modelo de Representación Lingüística.

Índice de contenido

INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	5
1.1 Introducción	5
1.2 Evaluación de Impacto Ambiental.....	5
1.3 Métodos para la EIA	6
1.4 Metodología para la EIA	8
1.4.1 Análisis crítico de la metodología.....	13
1.5 Modelado Lingüístico de Preferencias	14
1.6 Modelos Computacionales.....	15
1.6.1 Modelo de Representación Lingüística con 2-tuplas	16
1.7 Análisis de las soluciones internacionales	23
1.8 Análisis de soluciones existentes en Cuba	24
1.9 Resumen del análisis de las soluciones existentes.....	25
1.10 Metodología, Lenguajes y Herramientas de desarrollo	25
1.10.1 Metodología de desarrollo	25
1.10.2 Herramienta de modelado	26
1.10.3 Lenguaje de programación	26
1.10.4 Frameworks utilizados	27
1.10.5 Entorno integrado de desarrollo.....	28
1.10.6 Sistema de gestor de base de datos.....	28
1.10.7 Servidor web.....	28
1.11 Conclusiones del capítulo	29
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA.....	30
2.1 Introducción	30
2.2 Modelación del proceso del negocio.....	30
2.3 Propuesta del sistema	31
2.4 Funcionalidades del sistema.....	32

2.5	Personas relacionadas con el sistema	33
2.6	Lista de reserva del producto	33
2.7	Historias de Usuario.....	34
2.8	Planificación	37
2.8.1	Estimación de esfuerzo por Historias de Usuario.....	38
2.8.2	Plan de iteraciones	38
2.8.3	Plan de duración de las iteraciones	39
2.8.4	Plan de entregas.....	40
2.9	Conclusiones del capítulo	40
CAPÍTULO 3: DISEÑO DEL SISTEMA		41
3.1	Introducción	41
3.2	Patrón de arquitectura	41
3.3	Patrones de diseño.....	43
3.4	Tarjetas Clase-Responsabilidad-Colaborador (CRC).....	46
3.5	Diseño de la Base de Datos.....	49
3.6	Conclusiones	50
CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBA		51
4.1	Introducción	51
4.2	Tareas de la ingeniería	51
4.3	Validación del diseño.....	52
4.4	Pruebas	58
4.5	Conclusiones	60
CONCLUSIONES GENERALES		62
RECOMENDACIONES.....		63
REFERENCIAS BIBLIOGRÁFICAS.....		64
ANEXOS		66
	Anexo I: Importancia de Impacto	66
	Anexo II: Ejemplos de aplicaciones del Modelado Lingüístico de Preferencias	67

Anexo III: Historias de Usuario	68
Anexo IV: Tarjetas Clase-responsabilidad-Colaborador	79
Anexo V: Descripción de las tablas del modelo de datos del sistema.....	83
Anexo VI: Tareas de la ingeniería.....	90
Anexo VII: Pruebas de aceptación.....	97
Anexo VIII: Aceptación del sistema por parte del cliente.	111

Índice de tablas

Tabla 1: Nivel de 2 Jerarquías Lingüísticas.....	19
Tabla 2 HU Nro. 11 Valoración de importancia de impacto.....	36
Tabla 3: Estimación de esfuerzo por HU.....	38
Tabla 4: Plan de duración de iteraciones.....	39
Tabla 5: Plan de entregas.....	40
Tabla 6: Clase Usuario.....	47
Tabla 7: Clase Proyecto.....	47
Tabla 8: Tarea Nro. 18 Valoración de importancia de impacto.....	51
Tabla 9: Atributos de calidad evaluados por la métrica TOC.....	53
Tabla 10: Criterios de evaluación para la métrica TOC.....	53
Tabla 11: Atributos de calidad evaluados por la métrica RC.....	54
Tabla 12: Criterios de evaluación para la métrica RC.....	54
Tabla 13: Resultado de las pruebas de aceptación.....	60

Índice de ilustraciones

Ilustración 1 : Matriz de Impacto	9
Ilustración 2: Jerarquía Lingüística de 3, 5 y 9 etiquetas	20
Ilustración 3: Proceso de negocio en el CEQA	31
Ilustración 4: Propuesta del sistema SEVIA	32
Ilustración 5: Arquitectura N-Capas basada en el Framework Spring	42
Ilustración 6: Tratamiento de errores	43
Ilustración 7: Estructura de la Tarjeta CRC	47
Ilustración 8: Modelo de datos del sistema	49
Ilustración 9: Resultados de la métrica TOC para el atributo Responsabilidad	56
Ilustración 10: Resultados de la métrica TOC para el atributo Complejidad	56
Ilustración 11: Resultados de la métrica TOC para el atributo Reutilización	56
Ilustración 12: Resultados de la métrica RC para el atributo Acoplamiento	57
Ilustración 13: Resultados de la métrica RC para el atributo Complejidad de Mantenimiento	57
Ilustración 14: Resultados de la métrica RC para el atributo Cantidad de Pruebas	58
Ilustración 15: Resultados de la evaluación de la métrica RC para el atributo Reutilización	58
Ilustración 16: Resultado de las pruebas unitarias con junit	59

Introducción

El medio ambiente es el entorno vital, o sea el conjunto de factores físicos-naturales, estéticos, culturales, sociales y económicos, que interactúan con el individuo y con las comunidades en donde viven, capaces de, en un plazo corto o largo, causar efectos adversos directos o indirectos sobre los seres vivos y las actividades humanas (1).

En Cuba en 1997 se crea la Ley 81 del Medio Ambiente en la cual se define la estrategia a seguir para el cuidado y protección de los diferentes ecosistemas. Al emprender determinados proyectos (nuevas obras, instalaciones o programas) se hace necesario conseguir un desarrollo sostenido y equilibrado. Con este objetivo se emiten Licencias Medioambientales por entidades certificadas, que validan la no incidencia negativa de estos proyectos en cuestión sobre el medioambiente (2). Para ello es necesario realizar la Evaluación de Impacto Ambiental (EIA).

La EIA es un procedimiento jurídico-administrativo que tiene por objetivo la identificación, predicción e interpretación de los impactos ambientales que un proyecto o actividad produciría en caso de ser ejecutado, así como la prevención, corrección y valoración de los mismos (1). El Centro de Estudios de Química Aplicada (CEQA) de la Universidad Central de las Villas, es una entidad facultada para la realización de la EIA. Sus especialistas brindan este servicio a las diferentes empresas e instituciones que lo requieran, haciendo uso de la metodología para la realización de la EIA propuesta por Conesa (1).

En la metodología propuesta por Conesa se realizan evaluaciones simplificadas (evaluar solo una parte del proceso) y detalladas (evaluación completa). Dentro de las evaluaciones simplificadas se encuentra la valoración cualitativa de impactos, que tiene como salida principal la Matriz de Importancia de Impacto¹. El modelo propuesto para el cálculo de la Importancia del Impacto por Conesa, parte de la evaluación de 10 símbolos² que poseen naturaleza lingüística y numérica. En el caso de los símbolos con naturaleza lingüística pueden presentar cierta dificultad para ser representados con valores numéricos exactos por parte de los Expertos que los evalúan, introduciendo cierto grado de incertidumbre basado en sus propias percepciones de la realidad.

En el proceso de valoración, los Expertos se reúnen y llegan a un consenso de forma verbal, sobre la evaluación de cada uno de los símbolos, registrándolo en documentos Excel, lo que puede provocar

¹ La Matriz de Importancia de Impacto, que es de tipo causa-efecto, consiste en un cuadro de doble entrada, donde las columnas figuran las acciones impactantes y las filas los factores medioambientales susceptibles a recibir impactos.

² Indicadores cualitativos y cuantitativos para medir el impacto que puede causar una acción sobre un factor del medio.

dispersión de la información y dificultad en el momento de realizar estudios comparativos. La valoración de los símbolos es realizada con valores numéricos, disminuyendo la precisión del resultado final y sin tener en cuenta la naturaleza cualitativa de los mismos. La agregación de las preferencias se realiza de forma verbal sin que medie un método científico y no disponen de ninguna herramienta que centralice y guíe el proceso.

Se presenta entonces esta situación como un problema de Toma de Decisiones Multiexpertos/Multicriterios, donde los aspectos a valorar cuentan con una naturaleza cualitativa, bajo incertidumbre y la información numérica no se presenta como una buena opción para su modelado por la dificultad de su valoración mediante números exactos.

De lo planteado se deriva la necesidad e importancia para el CEQA, de implementar un modelo que permita la valoración de preferencias heterogéneas (tanto lingüísticas como numéricas) de los símbolos en el proceso de EIA por parte de los Expertos.

Problema a resolver:

La evaluación numérica de los símbolos con naturaleza cualitativa, dificulta su valoración por parte de los Expertos para la conformación de la Matriz de Importancia de Impacto en el CEQA durante la EIA.

Objeto de estudio:

Proceso de Toma de Decisiones bajo incertidumbre en la evaluación de impactos medio ambientales.

Objetivo general:

Desarrollar una aplicación informática que permita modelar preferencias heterogéneas (lingüísticas y numéricas) durante la Evaluación de Impacto Ambiental en el Centro de Estudios de Química Aplicada.

Objetivos específicos:

- Elaborar el marco teórico de la investigación.
- Aplicar un modelo computacional para operar sobre las etiquetas lingüísticas.
- Aplicar una técnica para integrar vectores de preferencias con diferentes granularidades.
- Aplicar una técnica para la unificación de información heterogénea en el proceso de EIA.
- Lograr la aceptación del software por parte del cliente.

Campo de acción:

Proceso de conformación de la Matriz de Importancia de Impacto Ambiental en el CEQA durante la EIA.

Tareas de investigación:

- Elaboración de la fundamentación teórica del trabajo.
- Estudio de la Metodología para la Evaluación de Impacto Ambiental propuesta por Conesa.
- Estudio del Enfoque Lingüístico Difuso.
- Estudio del Modelo de Representación de Información Lingüística de las 2-tuplas.
- Aplicación del Modelo de Representación de Información Lingüística de las 2-tuplas para valoraciones heterogéneas Multiexpertos para la realización de la EIA.
- Análisis de las diferentes herramientas y metodología de desarrollo de software que faciliten el desarrollo de la aplicación.
- Identificación y descripción de los procesos del negocio.
- Elaboración del modelo de negocio de los procesos que tienen lugar en la entidad en cuestión, para la comprensión del flujo normal de funcionamiento para la EIA.
- Definición de requisitos funcionales y no funcionales de la solución de software para controlar la EIA.
- Elaboración del modelo de diseño del software a desarrollar.
- Desarrollo de un software para la EIA que implemente el modelo computacional para el trabajo con etiquetas lingüísticas seleccionado.
- Realización de pruebas de funcionalidad a la solución de software.

El presente documento tiene la siguiente estructura:

Capítulo 1. Fundamentación Teórica: se realiza un estudio de los sistemas de EIA y las soluciones existentes. Se define la metodología de desarrollo, herramientas y tecnologías a utilizar en el desarrollo de la aplicación.

Capítulo 2. Características del sistema: contiene la propuesta del sistema, además de los requisitos no funcionales. Se plantean los requisitos funcionales a través de las Historias de Usuario (HU).

Capítulo 3. Diseño del sistema: se especifica el patrón arquitectónico y los patrones de diseño, se representan las clases del sistema mediante las Tarjetas Clase–Responsabilidad–Colaborador (CRC). Además se detalla el modelo físico de la base de datos.

Capítulo 4. Implementación y prueba: se exponen las tareas de la ingeniería generadas por cada Historia de Usuario (HU) y se realizan las pruebas al software con el propósito de garantizar una adecuada implementación del sistema informático.

Capítulo 1: Fundamentación Teórica

1.1 Introducción

En el presente capítulo se realiza un estudio sobre algunas de las metodologías, métodos y sistemas de EIA usados internacionalmente. Se abordan diversas definiciones como el Modelado de Representación Lingüística de las 2-tuplas y su inclusión dentro de los procedimientos a realizar en la EIA. También se describen las herramientas, lenguajes de programación y metodología de desarrollo que serán empleadas en la aplicación.

1.2 Evaluación de Impacto Ambiental

El medio ambiente es una fuente de recursos que abastece al ser humano de las materias primas que necesita para su desarrollo en el planeta. Solo una parte de estos recursos son renovables, por lo que requieren un inmenso cuidado para evitar su agotamiento.

Las acciones humanas afectan a diversos ecosistemas, modificando con ello la evolución natural del planeta. Cada vez son mayores los efectos negativos que ocurren en el medio ambiente a causa de las actividades irresponsables que comete la humanidad. Algunos de estos efectos son: los cambios climáticos, el ascenso del nivel del mar, las lluvias ácidas, el deshielo de los glaciares, la extinción de plantas y animales, entre otros.

Con el fin de prevenir o mitigar los efectos perjudiciales causados en proyectos constructivos, inversionistas o de diferente procedencia, es importante y necesario realizar la EIA. Estas evaluaciones pretenden como principio establecer un equilibrio entre el desarrollo de la humanidad y el medio ambiente, sin pretender llegar a ser una figura negativa, ni un freno al desarrollo, sino un instrumento operativo para impedir sobreexplotaciones del medio natural (1).

En Cuba la Ley 81 del Medio Ambiente, tiene como objetivo establecer los principios que rigen la política ambiental, las normas básicas para regular la gestión ambiental del Estado, las acciones de los ciudadanos y la sociedad en general. En su capítulo IV la misma establece el procedimiento para la realización de la Evaluación de Impacto Ambiental y comprende los siguientes pasos (2):

- a. La solicitud de licencia ambiental.
- b. El estudio de impacto ambiental, en los casos en que proceda.

- c. La evaluación propiamente dicha, a cargo del Ministerio de Ciencia, Tecnología y Medio Ambiente.
- d. El otorgamiento o no de la licencia ambiental.

La EIA se realiza a un proyecto específico, con particularidades tales como: tipo de obra, materiales a ser usados, procedimientos constructivos, trabajos de mantenimiento en la fase operativa, tecnologías utilizadas, insumos, entre otros. Dado el carácter de instrumento predictivo de las EIA, estas no se aplicarán a obras o planes ya realizados, solo puede aplicarse a estos para prevenir sus efectos hacia el futuro. Las consecuencias de una evaluación negativa pueden ser diversas según la legislación y según el rigor con que ésta se aplique, que va desde la paralización definitiva del proyecto hasta su ignorancia completa.

Para realizar la EIA se pueden utilizar diferentes metodologías. Algunos métodos son generales y otros muy específicos, pero de todos ellos pueden obtenerse técnicas que con variaciones pueden ser útiles para la evaluación. La mayor parte de estos métodos se elaboran para determinados trabajos, por lo que en ocasiones no es sencillo su uso tal y como fueron creados, pero adaptándolos a cada caso específico pueden llegar a ser muy útiles. Los métodos de EIA pueden no tener aplicabilidad uniforme en todos los países debido a diferencias en su legislación, marco de procedimientos, datos de referencia, estándares ambientales y programas de administración ambiental.

1.3 Métodos para la EIA

Entre los métodos para la EIA, se encuentran los siguientes (3):

- ✓ Listas de chequeo.
- ✓ Matriz de Leopold.
- ✓ Sistema de Evaluación Ambiental Batelle-Columbus.
- ✓ Método de transparencias (Mc Harg).
- ✓ Análisis costes-beneficios.
- ✓ Modelos de simulación.
- ✓ Sistemas basados en un soporte informatizado del territorio.

Listas de chequeo

Su funcionalidad principal es la de servir en las primeras etapas para identificar los impactos ambientales, su contenido cambia según el tipo de proyecto y el medio de actuación, por lo que no son inmutables (3).

Matriz de Leopold

La Matriz de Leopold es un método cuantitativo de EIA creado en 1971 (4).

Méritos del método (3):

- ✓ Fuerza al considerar los posibles impactos de acciones proyectuales sobre diferentes factores ambientales.
- ✓ Incorpora la consideración de magnitud e importancia de un impacto ambiental.
- ✓ Permite la comparación de alternativas desarrollando una matriz para cada opción.
- ✓ Sirve como resumen de la información contenida en el informe de impacto ambiental.

Sistema de Evaluación Ambiental Batelle-Columbus

El Sistema de Evaluación Ambiental de Battelle (SEA) es un método para el análisis del impacto ambiental desarrollado en los laboratorios Battelle-Columbus por un equipo de investigación interdisciplinario bajo contrato con el U.S. Bureau of Reclamation (Buró de Reclamaciones de Estados Unidos) (5) (6).

Méritos del método (3):

- ✓ Se trata del primer esfuerzo serio de valoración de impactos que ha servido de base a métodos posteriores.
- ✓ Los parámetros o factores ambientales se transformaron a unidades comparables, representativas de la calidad del medio ambiente, lo que permite la adición de las magnitudes de impacto para cada acción y para cada factor ambiental.
- ✓ Permite el cálculo del impacto ambiental global del proyecto y la comparación de alternativas al proyecto.

Método de transparencias (Mc Harg)

Este método tiene presente las características del territorio, sin llegar a una evaluación profunda de los impactos, pero haciendo una identificación e inventario de los recursos para la integración de los proyectos al entorno, de la forma más armoniosa posible, dejando íntegras las zonas de gran valor social, con el costo mínimo y la obtención de plusvalía (3).

Análisis costes-beneficios

Un análisis costes-beneficios puede valorar un problema ambiental mediante una comparación de los costes por daños frente a los costes para evitarlos y estos se sitúan en oposición a los beneficios. Cuando existen datos, este sistema analítico de tipo económico puede ser usado para comparar opciones alternativas (3).

Modelos de simulación

Están basados en modelos de transporte y transformación de contaminantes en la atmósfera o el agua superficial o subterránea. Si existen datos básicos suficientes y correctos de las zonas de afectación por las emisiones o vertidos de uno o varios focos, estos métodos efectúan un análisis mediante la modelización de las características básicas de los medios emisores, difusor y receptor, considerando las interrelaciones temporales y espaciales (3).

Sistemas basados en un soporte informatizado del territorio

Surgen como una herramienta para el manejo de los datos espaciales, aportando soluciones a problemas geográficos complejos, lo cual permite mejorar la habilidad del usuario en la toma de decisiones de investigación, planificación y desarrollo (3).

1.4 Metodología para la EIA

Durante la investigación se estudió la metodología propuesta por Conesa, la cual propone un modelo de Estudio del Impacto Ambiental (EsIA) basado en el método de las matrices causa-efecto, derivadas de la Matriz de Leopold (4) con resultados cualitativos y del método del Instituto Batelle-Columbus (5) (6) con resultados cuantitativos. Dicho modelo consiste en un cuadro de doble entrada cuyas columnas figuran las acciones impactantes y las filas los factores ambientales susceptibles de recibir impactos (1).

El EsIA es el estudio técnico, de carácter interdisciplinar que incorporado en el procedimiento de la EIA, está destinado a predecir, identificar, valorar y corregir, las consecuencias o efectos ambientales que determinadas acciones pueden causar sobre la calidad de vida del hombre y su entorno (1). Este EsIA consta de las siguientes fases:

1. Análisis del proyecto y sus alternativas.
2. Definición del entorno del proyecto y posterior descripción y estudio del mismo.
3. Previsiones que los efectos generarán sobre el medio.

4. Identificación de las acciones del proyecto potencialmente impactantes.
5. Identificación de los factores del medio potencialmente impactados.
6. Identificación de relaciones causa-efecto entre acciones del proyecto y factores del medio.
Elaboración de la Matriz de Importancia y valoración cualitativa del impacto.
7. Predicción de la magnitud de impacto sobre cada factor.
8. Valoración cuantitativa del impacto ambiental.
9. Definición de medidas correctoras.
10. Proceso de participación pública.
11. Emisión del informe.
12. Decisión del órgano competente.

Las seis primeras fases corresponden a la valoración cualitativa. Las fases siete, ocho y nueve corresponden a la valoración cuantitativa. Las nueve primeras fases corresponden al EsIA. Las fases diez, once y doce no corresponden propiamente al EsIA, sino que forman parte del proceso de la EIA. Obviando las fases siete, ocho y nueve, se encuentran ante una Evaluación Simplificada. El conjunto de las doce fases conducen a la Evaluación Detallada.

La investigación se centrará en el análisis de la fase seis, específicamente en la determinación de la Importancia del Impacto, o sea la importancia del efecto de una acción sobre el factor ambiental.

Matriz de Impacto. Valoración cualitativa del impacto ambiental

La Matriz de Impacto que es de tipo causa-efecto, consiste en un cuadro de doble entrada, donde las columnas figuran las acciones impactantes y las filas los factores medioambientales susceptibles a recibir impactos (1).

	Acciones				
Factores					

Ilustración 1 : Matriz de Impacto

Para su ejecución es necesario determinar las acciones que puedan ocasionar impactos sobre una serie de factores del medio.

En la valoración cualitativa se busca obtener una estimación de los posibles efectos que recibirá el medio ambiente, mediante una descripción lingüística de las propiedades de tales efectos.

Identificación de acciones que puedan causar impacto

Existen varios medios para identificar acciones, entre los que se pueden destacar los cuestionarios específicos para cada tipo de proyecto, las consultas a paneles de expertos, escenarios comparados, consultas a los propios proyectos, grafos de interacción causa-efecto, entre otros (1).

Identificación de factores ambientales del entorno susceptibles a recibir impactos

En esta fase se lleva a cabo la identificación de factores ambientales con el fin de detectar aquellos aspectos del medioambiente cuyos cambios, motivados por las distintas acciones del proyecto en sus sucesivas fases (construcción, explotación o funcionamiento, ampliación o reforma y abandono o derribo), supongan modificaciones positivas o negativas en la calidad ambiental del mismo. Para la identificación de estos factores se usarán los mismos instrumentos citados en la detección de las acciones, (cuestionarios especializados, consultas a paneles de expertos, entre otros.).

A cada factor medioambiental se le asigna una medida de su importancia relativa al entorno (un número de 1 a 1000), medida en Unidades de Importancia (UIP) y que servirá posteriormente para efectuar ponderaciones en las estimaciones globales de los efectos.

Matriz de Importancia

Una vez identificadas las acciones y los factores del medio, la Matriz de Importancia brindará una valoración cualitativa al nivel requerido por una EIA Simplificada. La valoración cualitativa se efectuará a partir de la Matriz de Impacto.

Para obtener la Importancia del Impacto se valoran diez símbolos, a los que se añade uno que sintetiza en una cifra el valor final de dicha importancia. Estos símbolos debido a la naturaleza que presentan son analizados de forma cualitativa, ellos son (1):

Signo: Hace referencia al carácter beneficioso (+) o perjudicial (-) de las acciones que van a actuar sobre los factores considerados. Existe la posibilidad de incluir un nuevo valor (x) cuando se refiere a efectos externos al proyecto, previsibles pero difíciles de cualificar.

Intensidad (I): Se refiere al grado de incidencia de la acción sobre el factor. El rango de valoración estará comprendido entre 1-12, donde el 1 representa una afección mínima y el 12 expresará una destrucción total.

Extensión (EX): Se refiere al área de influencia teórica del impacto en relación con el entorno del proyecto. Para una acción que cause un efecto localizado se considera de carácter Puntual 1. Si el efecto tiene una influencia generalizada dentro del entorno del proyecto adquirirá valor de 8, en situaciones intermedias se tendrán valores de Parcial 2 y Extenso 4. En caso de que el efecto sea puntual pero se produzca en un lugar crítico (por ejemplo, vertimiento cercano a una toma de agua), se le otorgará un valor de 4 unidades por encima del que le correspondería en función de la extensión en la que se manifiesta. En el caso de considerar que sea peligroso y no exista posibilidad de tomar medidas correctoras se deberá cambiar a otra alternativa al proyecto, anulando la causa que produce el efecto.

Recuperabilidad (MC): Se refiere a la posibilidad de reconstrucción, total o parcial, del factor afectado como consecuencia del proyecto, es decir, la posibilidad de retornar a las condiciones iniciales previas a la actuación, por medio de la intervención humana (medidas correctoras). Si el efecto es totalmente Recuperable, se le asigna un valor de 1 o 2 en caso de que sea de manera inmediata o a medio plazo, si lo es parcialmente, el efecto es Mitigable y toma un valor de 4. Cuando el efecto es Irrecuperable (imposible de reparar, tanto por la acción natural, como por la humana) se le asigna un valor de 8. En caso de ser Irrecuperable pero que exista la posibilidad de introducir medidas compensatorias el valor adoptado será 4.

Reversibilidad (RV): Se refiere a la posibilidad de reconstruir el factor afectado por el proyecto, es decir la posibilidad de retornar a las condiciones iniciales previas a la acción, por medios naturales, una vez que deja de actuar sobre el medio. Si es a Corto Plazo se le asigna valor de 1, si es a Medio Plazo 2 y si el efecto es irreversible se le asigna valor de 4. Los intervalos de tiempo comprendidos en este período se corresponden con el parámetro anterior.

Persistencia (PE): Se refiere al tiempo que, supuestamente, permanecería el efecto desde su aparición y a partir del cual el factor afectado retornaría a las condiciones iniciales previas a la acción, por medios naturales o por acciones correctoras introducidas. Si tienen lugar durante menos de un año produce un efecto Fugaz y tomaría un valor de 1. Si el efecto permanece por un período de 1 a 10 años sería Temporal 2; si tienen una duración superior a los diez años, se considera un efecto Permanente 4.

Momento (MO): Hace referencia al tiempo que transcurre entre la aparición de la acción y el comienzo del efecto sobre el factor del medio considerado. Cuando el tiempo transcurrido sea nulo el momento será Inmediato y si es inferior a un año tendrá Corto Plazo, ambos casos con un valor de 4. Si es un período de tiempo que va de 1-5 años, Medio Plazo 2 y si tarda de manifestarse el efecto más de 5 años, Largo Plazo con el valor asignado 1. Si ocurre alguna circunstancia que hiciera crítico el momento de impacto se le atribuiría un valor de 1 a 4 unidades por encima de las especificadas (por ejemplo, el ruido en la noche cercano a un centro hospitalario).

Sinergia (SI): Este atributo contempla el reforzamiento de dos o más efectos simples. La componente total de la manifestación de los componentes simples, provocados por acciones que actúan simultáneamente, es superior a la que se esperaría de la manifestación de los efectos cuando las acciones que la provocan actúan de manera independiente no simultánea. Cuando una acción sobre un factor, no es sinérgica con otras acciones que actúan sobre el mismo factor, el atributo toma valor de 1, si presenta un sinergismo moderado 2 y si es altamente sinérgico 4.

Acumulación (AC): Representa el incremento progresivo de la manifestación de efecto cuando persiste de forma continua o reiterada la acción que lo genera. Cuando una acción no produce efectos acumulativos, se valora como 1. Si el efecto es acumulativo se incrementa a 4.

Efecto (EF): Se refiere a la relación causa-efecto, es decir la forma de manifestación del efecto sobre el factor, como consecuencia de una acción. El efecto puede ser directo o primario cuando la repercusión de la acción es consecuencia directa de esta. En caso de que el efecto sea indirecto o secundario, su manifestación no es consecuencia directa de la acción sino que tiene lugar a partir de un efecto primario, actuando este como una acción de segundo orden. Toma el valor de 1 en caso de que sea secundario y el valor de 4 en caso de que sea primario.

Periodicidad (PR): Se refiere a la regularidad de manifestación del efecto, ya sea de forma cíclica o recurrente (efecto periódico), de forma impredecible en el tiempo (efecto irregular), o constante en el tiempo (efecto continuo). A los efectos continuos se les asigna un valor de 4, a los periódicos un valor de 2 y a los de aparición irregular 1.

La Importancia del impacto (I) viene representada por un número que se obtiene a través de la siguiente fórmula y en función de los valores asignados a los símbolos correspondientes:

$$I = +[3I + 2EX + MO + PE + RV + SI + AC + EF + PR + MC]$$

Esta Importancia toma valores entre 13 y 100. Los factores de importancia por debajo de 25, son irrelevantes, o sea compatibles. Los impactos moderados presentan una importancia entre 25 y 50. Serán severos cuando la importancia se encuentre entre 50 y 75 y críticos cuando el valor sea superior a 75.

La determinación de la Importancia del Impacto de una acción sobre los factores del medio, es uno de los pasos fundamentales para la confección de la Matriz de Importancia perteneciente al EsIA propuesto en la metodología. Para ello se evalúan de forma cualitativa 10 símbolos, como se ilustró anteriormente, cada uno con diferencias en los intervalos de evaluación, caracterizándose estos valores por un amplio grado de incertidumbre. El resultado obtenido tiene una naturaleza numérica.

En el Anexo I se puede consultar la tabla de Importancia del Impacto.

En el caso de algunos de estos símbolos a pesar de presentar naturaleza cualitativa, su valoración por parte de los Expertos, es realizada con valores numéricos, disminuyendo la precisión del resultado final y sin tener presente las preferencias individuales de los Expertos. Los principios de valoración de los Expertos están definidos bajo cierto grado de incertidumbre basado en sus propias percepciones de la realidad. Debido a que generalmente no cuentan con toda la información necesaria para tener un alto grado de precisión durante las evaluaciones.

1.4.1 Análisis crítico de la metodología

La metodología propuesta por Conesa es muy útil para evaluar impactos ambientales y logra en alguna medida satisfacer los requisitos de las EIA, sin embargo, al analizarla se evidencian algunos problemas, como por ejemplo (7):

La “valoración cualitativa” utiliza variables cuantitativas:

En el cómputo de la Importancia se emplean etiquetas para caracterizar variables que son claramente cuantificables. Es de anotar que estas variables numéricas primero se convierten en variables no-numéricas (por ejemplo el Momento se evalúa como inmediato, a medio plazo o a largo plazo), para luego volver a convertirse en variable numérica mediante la asignación de un número asociado a cada etiqueta.

La “valoración cualitativa” es realmente cuantitativa:

La valoración cualitativa es en realidad una serie de operaciones sobre variables definidas en la recta entera, ya que en el fondo lo único que se hace es representar los posibles valores de las

variables como un conjunto de valores numéricos discretos. Puede decirse, por tanto, que el modelo lingüístico de la valoración cualitativa se define sobre números enteros.

No se modela la incertidumbre:

Pese a que es de esperar que algunas variables no se puedan determinar con absoluta precisión, la metodología no establece ningún procedimiento para tratar variables con incertidumbre.

Cuando se intenta modelar aspectos cuantitativos (longitud, superficie, entre otros) el uso de información numérica suele ser habitual en el momento de realizar la modelación de las preferencias, mientras que cuando intentamos modelar aspectos cualitativos (confort, diseño, entre otros), la información numérica no se presenta como una buena opción ya que son aspectos que pueden ser difíciles de valorar mediante un valor exacto (8).

Para el estudio que se realiza se considera necesario profundizar en las escalas lingüísticas, por brindar las herramientas necesarias para la valoración de los símbolos relacionados con la Importancia del Impacto, pertenecientes a la EIA definida por Conesa.

1.5 Modelado Lingüístico de Preferencias

El modelado lingüístico difuso permite representar aspectos cualitativos y está basado en el concepto de variables lingüísticas, es decir, variables cuyos valores no son números, sino palabras o sentencias expresadas en lenguaje natural o artificial (9).

El Enfoque Lingüístico Difuso tiene como base teórica los conjuntos difusos y se ha mostrado como una técnica eficaz para valorar aspectos de naturaleza cualitativa (8). Es ideal para aplicarlo en situaciones en las que la información con que trabajan los Expertos es demasiado imprecisa para ser analizada utilizando valores numéricos o en las que la información que se maneja no es fácil de cuantificar y requiere de términos lingüísticos para que sea valorada.

El Enfoque Lingüístico Difuso ha sido utilizado ampliamente en gran cantidad de aplicaciones, sin embargo, atendiendo a la forma en que se utiliza la información lingüística, se pueden distinguir dos líneas principales de aplicación (8):

1. *Sistemas basados en reglas difusas con variables lingüísticas:* Son una extensión de los sistemas basados en reglas de proposiciones. Utilizan la lógica difusa tanto para representar distintas formas de conocimiento sobre el problema como para modelar las interacciones y

relaciones existentes entre las variables del mismo. La principal aplicación de estos sistemas inteligentes es el modelado difuso de sistemas (8).

2. *Modelado Lingüístico de Preferencias*: Consiste en representar las preferencias de un problema mediante etiquetas lingüísticas, para a partir de ellas resolver el problema. Este modelo se ha utilizado con éxito sobre un gran número de problemas, tales como, "Toma de Decisiones" y "Decisiones Multicriterios \Multiexpertos" (10) (11) (12) (13) (14) (15) (16) (8).

El Modelado Lingüístico de Preferencias implica la necesidad de realizar operaciones con etiquetas lingüísticas, es decir, el uso de técnicas computacionales que tienen definidos operadores de agregación, comparación, negación, entre otros, sobre información lingüística.

En el Anexo II se muestra una tabla en la que se presenta un resumen de los distintos problemas agrupados por las áreas mencionadas en las que se ha utilizado el Modelado Lingüístico de Preferencias.

1.6 Modelos Computacionales

El uso de variables lingüísticas implica la necesidad de realizar procesos de computación con palabras, tales como comparación, negación y agregación de variables lingüística. En la literatura consultada se encontraron tres tipos de modelos computacionales:

1. Modelos basados en el Principio de Extensión.
2. Modelos Simbólicos.
3. Modelo de Representación Lingüística con 2-tuplas.

Modelo basado en el Principio de Extensión: Cuando la semántica de las etiquetas está definida por conjuntos difusos (22) se puede utilizar el principio de extensión para operar sobre éstos. El uso de la aritmética difusa desarrollada a partir del Principio de Extensión (23), puede incrementar la imprecisión de los resultados (números difusos) y hacer que no coincidan con ninguno de los números difusos que definen la semántica de las etiquetas lingüísticas iniciales. En ese caso, se lleva a cabo un proceso de aproximación lingüística (24) (10) que identifica el número difuso obtenido en las operaciones con el número difuso más cercano asociado a una etiqueta inicial.

Modelo Simbólico: Los operadores actúan, en este caso, sobre una valoración del orden que ocupa la etiqueta en el conjunto de términos lingüísticos (valor discreto numérico) (25), obteniendo resultados en un dominio continuo (intervalo de valores de R), por lo que éstos pueden no coincidir con ningún

valor asociado al orden que ocupa una etiqueta. En tal caso, hay que hacer una aproximación del resultado al valor discreto más cercano que represente a una etiqueta.

En los dos primeros modelos se observó que al operar sobre información lingüística se produce una pérdida de información debido a las operaciones de aproximación, que afectan la precisión de los resultados finales (8).

Esta pérdida de información se debe al modelo de representación de información del Enfoque Lingüístico Difuso, ya que opera con valores discretos sobre un universo del discurso continuo. La pérdida de información causa una falta de precisión en los resultados, es decir, puede ocasionar que al operar con distintos valores de entrada obtengamos iguales resultados, aunque antes de utilizar los procesos de aproximación dichos valores fueran distintos (8).

Para solucionar estos problemas de pérdida de información, se desarrolló en 1999 por Luis Martínez López (8), un nuevo Modelo de Representación de Información Lingüística basado en 2-tuplas para la agregación de preferencias lingüísticas, dicho modelo es el que se empleará en la investigación.

1.6.1 Modelo de Representación Lingüística con 2-tuplas

Sea $S = \{S_0, \dots, S_g\}$ un conjunto de términos lingüísticos y $\beta \in [0; g]$ un valor obtenido por un método simbólico operando con información lingüística.

La Traslación Simbólica de un término lingüístico S_i es un número valorado en el intervalo $[-0.5, 5)$ que expresa la “diferencia de información” entre una cantidad de información expresada por el valor $\beta \in [0; g]$ obtenido en una operación simbólica y el valor entero más próximo, $i \in [0, \dots, g]$, que indica el índice de la etiqueta lingüística (S_i) más cercana en S .

A partir de este concepto se desarrolló un nuevo modelo de representación para la información lingüística, el cual usa como base de representación 2-tuplas, (r_i, α_i) , $r_i \in S$ y $\alpha_i \in [-0.5, .5)$ (26).

- r_i representa la etiqueta lingüística.
- α_i es un número que expresa el valor de la distancia desde el resultado original β al índice de la etiqueta lingüística más cercana (r_i) en el conjunto de términos lingüísticos S , es decir, su traslación simbólica.

Sea $S_i \in S$ un término lingüístico, su representación mediante una 2-tupla equivalente se obtiene mediante la función Θ (8):

$$\Theta: S \rightarrow (S \times [-0.5, .5])$$

$$\Theta (S_i) = (S_i, 0) / S_i \in S$$

Sea $S = \{S_0, \dots, S_g\}$ un conjunto de términos lingüísticos y $\beta \in [0; g]$ un valor que representa el resultado de una operación simbólica, entonces la 2-tupla lingüística que expresa la información equivalente β se obtiene usando la siguiente función (8):

$$F: [0; g] \rightarrow S \times [-0.5, .5)$$

$$F(\beta) = (S_i, \alpha), \text{ con } \begin{cases} S_i, i = \text{round}(\beta) \\ \alpha = \beta - i, \alpha \in [-0.5, .5) \end{cases}$$

Donde *round* es el operador usual de redondeo, S_i es la etiqueta con índice más cercano a β y α el valor de la traslación simbólica.

En “Un Modelo de Representación de Información Lingüística basado en 2-tuplas para la agregación de información lingüística” (8), se define el modelo computacional asociado que permite operar sobre dicha representación, así como el conjunto de operadores sobre este modelo para las operaciones: agregación, composición y negación.

Operadores de agregación de 2-tuplas: La agregación de información consiste en obtener un valor que resuma un conjunto de valores. En “Un Modelo de Representación de Información Lingüística basado en 2-tuplas para la agregación de información lingüística” (8), se pueden encontrar numerosos operadores de agregación que permiten combinar la información de acuerdo a distintos criterios. Cualquiera de estos operadores puede ser fácilmente extendido para trabajar con 2-tuplas usando funciones Δ y Δ^{-1} , que transforman valores numéricos en 2-tuplas y viceversa sin pérdida de información.

Algunos ejemplos de estos operadores son los siguientes (8):

Definición 1. Siendo $X = \{(s_1, \alpha_1), \dots, (s_n, \alpha_n)\}$ un conjunto de varias 2-tuplas lingüísticas, la 2-tupla que simboliza la media aritmética extendida, x^{-e} , se calcula de la siguiente forma:

$$x^{-e}[(r_1, \alpha_1), \dots, (r_n, \alpha_n)] = \Delta \left(\sum_{i=1}^n \frac{1}{n} \Delta^{-1}(r_i, \alpha_i) \right) = \Delta \left(\sum_{i=1}^n \frac{1}{n} \Delta^{-1}\beta \right)$$

Definición 2. Siendo $X = \{(r_1, \alpha_1), \dots, (r_n, \alpha_n)\}$ un conjunto de varias 2-tuplas lingüísticas y $W = (w_1, \dots, w_n)$ un vector numérico con los pesos asociados a cada 2-tupla. La media ponderada extendida X_w^{-e} se define como:

$$X_w^{-e} = \Delta \left(\frac{\sum_{i=1}^n (r_i, \alpha_i) \cdot w_i}{\sum_{i=1}^n w_i} \right) = \Delta \left(\frac{\sum_{i=1}^n \beta \cdot w_i}{\sum_{i=1}^n w_i} \right)$$

1.6.1.1 Enfoque para manejar Información Lingüística Multigranular

En el proceso de Evaluación de Impacto Ambiental se pretende predecir los efectos que causarán sobre el medio ambiente las diferentes acciones llevadas a cabo por el hombre en su proceso de desarrollo, valorando diferentes símbolos que en su mayoría presentan una naturaleza cualitativa. Debido a esto, normalmente los Expertos que participan en la valoración no tienen la información precisa, predominando sus propias percepciones de la realidad en una ambiente vago e impreciso, adaptándose mejor a un modelado cualitativo mediante el uso de etiquetas lingüísticas y no de valores numéricos precisos.

Por otro lado, el hecho de que para determinar la Importancia del Impacto de una acción sobre cada factor susceptible a ser impactado sean valorados diferentes símbolos, cada uno con sus particularidades y las mismas necesidades de precisión para varios Expertos, hace evidente la necesidad de someter, en algún momento al proceso de agregación, vectores de preferencias representados por escalas lingüísticas con diferentes granularidades.

En “Nuevo modelo de información sensorial con información lingüística” (9), se realiza una revisión de los diferentes enfoques para manejar múltiples escalas lingüísticas. Teniendo en cuenta que han sido identificadas las escalas lingüísticas necesarias para la representación de cada uno de los símbolos y que la granularidad varía ente 3 y 5, con funciones de pertenencia triangulares, simétricas y uniformemente distribuidas en el intervalo [0; 1] el enfoque de Jerarquías Lingüísticas se presenta adecuado para el cumplimiento de los objetivos de la presente investigación.

Jerarquías Lingüísticas

Este enfoque está basado en el Modelo de Representación Lingüística basado en 2-tuplas. Introduce un nuevo enfoque para manejar la información lingüística multigranular de un modo simbólico y preciso.

A) Marco de Trabajo

Los conjuntos de términos lingüísticos, S_i que componen el marco de trabajo, F_{MS} deben cumplir ciertas reglas asociadas al modo de construcción de una Jerarquía Lingüística (LH).

Construcción de una Jerarquía Lingüística

Una LH es un conjunto de niveles, donde cada nivel es un conjunto de términos lingüísticos con una granularidad diferente del resto de niveles de las jerarquías lingüísticas (27).

Cada nivel de una LH se denota como $l(t; n(t))$, siendo t un número que indica el nivel de la jerarquía y $n(t)$ la granularidad del conjunto de términos lingüísticos del nivel t (9).

Los niveles de una LH están ordenados en función de su granularidad, es decir, que para dos niveles consecutivos t y $t + 1$, $n(t + 1) > n(t)$. Por lo tanto, cada nivel $t + 1$ proporciona un refinamiento lingüístico con respecto al nivel anterior t . Las LH trabajan con términos lingüísticos cuyas funciones de pertenencia son triangulares, simétricas y uniformemente distribuidas en el intervalo $[0; 1]$. Además, los conjuntos de términos lingüísticos tienen una granularidad impar, indicando la etiqueta central, un valor de indiferencia (9).

Formalmente, una LH se define como la unión de todos los niveles t que la conforman (9):

$$LH = \bigcup_t l(t, n(t))$$

De forma genérica, se puede establecer que la granularidad del conjunto de términos lingüísticos del nivel $t + 1$, $S^{n(t+1)}$ puede obtenerse a partir del nivel anterior t , $S^{n(t)}$ de la siguiente manera (9):

$$l(t; n(t)) \rightarrow l(t + 1; 2 \cdot n(t) - 1)$$

Tabla 1: Nivel de 2 Jerarquías Lingüísticas

	$l(t, n(t))$	$l(t, n(t))$
$t=1$	$l(1, 3)$	$l(1, 7)$
$t=2$	$l(2, 5)$	$l(2, 13)$
$t=3$	$l(3, 9)$	

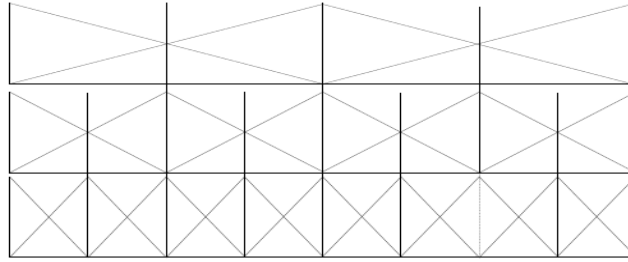


Ilustración 2: Jerarquía Lingüística de 3, 5 y 9 etiquetas

La granularidad de una LH viene determinada por el primer nivel. Si se define el primer nivel $l(1; 3)$, los siguientes niveles son $l(2; 5)$, $l(3; 9)$, entre otros. Por tanto, el uso de jerarquías lingüísticas no permite considerar conjuntos de etiquetas lingüísticas de cualquier granularidad como, por ejemplo, 5, 7 y 11. Este hecho hace que se restrinjan los conjuntos de términos lingüísticos que pueden pertenecer al marco de trabajo F_{MS} , siendo únicamente posible utilizar conjuntos de etiquetas que estén definidos sobre una misma LH (9).

B) Proceso de Unificación

El uso de las jerarquías lingüísticas permite realizar el proceso de unificación de información lingüística multigranular sin pérdida de información y obteniendo las valoraciones lingüísticas en 2-tuplas sin necesidad de utilizar conjuntos difusos.

En “A model based on linguistic 2-tuples for dealing with multigranularity hierarchical linguistic contexts in multiexpert decision-making” (28) fue definida una función de transformación entre etiquetas de diferentes niveles, esta función permite unificar la información en cualquier nivel de la LH.

Definición 3. Sea $LH = \cup_t l(t, n(t))$ una LH cuyos conjuntos de términos se denotará como $S^{n(t)} = \{S_0^{n(t)}, \dots, S_{n(t)-1}^{n(t)}\}$. La función de transformación de un término del nivel t a uno del nivel t' se define como:

$$TF_{t'}^t(s_i^{n(t)}, \alpha^{n(t)}) = \Delta \left(\frac{\Delta^{-1}(s_i^{n(t)}, \alpha^{n(t)}) \cdot (n(t') - 1)}{n(t) - 1} \right)$$

Esta función de transformación entre etiquetas lingüísticas de distintos niveles de una LH es biyectiva (28):

$$TF_t^{t'} \left(TF_{t'}^t(s_i^{n(t)}, \alpha^{n(t)}) \right) = s_i^{n(t)}, \alpha^{n(t)}$$

Este resultado garantiza que la transformación entre niveles de una LH se realice sin pérdida de información.

C) Proceso Computacional

La utilización de las LH permite realizar los procesos de computación con palabras sin pérdida de información, usando el modelo computacional para 2-tuplas (29).

Es importante notar que los resultados obtenidos en el proceso computacional pueden ser expresados en cualquier conjunto de términos lingüísticos del marco de evaluación, F_{MS} de un modo preciso por medio de la función de transformación TF_V^t (9).

1.6.2 Integración de información heterogénea basado en el modelo de representación lingüística con 2-tuplas

El proceso de agregación basado en el Modelo de Representación Lingüística con 2-tuplas facilita la combinación de información numérica y lingüística solucionando los problemas que aparecen en los enfoques anteriores, como los que se muestran a continuación (8):

1. Pérdida de precisión en el momento de combinar la información.
2. Las funciones de transformación entre elementos de un dominio y otro no son biyectivas, por lo tanto, pierden información.

Función de Transformación de un Valor en [0,1] a una 2-tupla en S

Sea $\vartheta \in [0; 1]$ un valor numérico y $S = \{s_0, \dots, s_g\}$ un conjunto de términos lingüísticos. El objetivo es obtener una 2-tupla lingüística basada en la traslación simbólica valorada en S y que represente la misma información que ϑ . A continuación se relaciona el proceso de transformación (8):

1. Convertir ϑ en un conjunto difuso sobre S.

Sea $F(S)$ el conjunto de conjuntos difusos en S, para transformar un valor numérico $\vartheta \in [0; 1]$ en un conjunto difuso en $F(S)$ se calcula el grado de emparejamiento de ϑ con las funciones de pertenencia de los términos lingüísticos de S.

Definición 4. Sea $\vartheta \in [0; 1]$ un número y $S = \{s_0, \dots, s_g\}$ un conjunto de términos lingüísticos. Se transformará ϑ en un conjunto difuso en S utilizando la función definida como sigue (8):

$$\tau: [0,1] \rightarrow F(S)$$

$$\tau(\vartheta) = \{(s_0, a_0), \dots, (s_g, a_g)\}, s_i \in S \text{ y } a_i \in [0,1], \text{ tal que,}$$

$$a_i = \mu_{s_i}(\vartheta) = \begin{cases} 0, & \text{si } \vartheta \notin \text{Support}(\mu_{s_i}(x)) \\ \frac{\vartheta - a_i}{b_i - a_i} & \text{si } a_i \leq \vartheta \leq b_i \\ 1, & \text{si } b_i \leq \vartheta \leq d_i \\ \frac{c_i - \vartheta}{c_i - d_i}, & \text{si } d_i \leq \vartheta \leq c_i \end{cases}$$

2. Transformar el conjunto difuso anterior en 2-tuplas lingüísticas valoradas en S.

Definición 5. Sea $T_{\text{Sist}}(li) = \{(c_0, \alpha_i^0), \dots, (c_g, \alpha_i^g)\}$ un conjunto difuso que representa un término lingüístico $li \in S_i$ sobre el conjunto básico de términos lingüísticos S_t . Se va a obtener un valor numérico que soporta la información del conjunto difuso y está valorado en el intervalo $[0; g]$ usando la siguiente función (8):

$$x: F(S_T) \rightarrow [0, g]$$

$$x(\{(c_k, a_k), k = 0, \dots, g, c_k \in S_T, a_k \in [-5,5]\}) = \frac{\sum_{j=0}^g j a_j^i}{\sum_{j=0}^g a_j^i} = \beta$$

Este valor se puede transformar fácilmente en 2-tuplas lingüísticas utilizando la función Δ . En este momento toda la información de entrada queda representada de forma uniforme en un elemento parte de un único conjunto de términos lingüísticos, S_t utilizando 2-tuplas.

La ventaja más importante de este modelo es la de ser continua en su dominio, por lo que puede representar cualquier cantidad de información del universo del discurso sin necesidad de realizar ningún proceso de aproximación (8).

Las 2-tuplas, se presenta como un Modelo de Representación Lingüística preciso, posibilita el trabajo en contextos donde exista información numérica y lingüística, sin ocasionar pérdida de información. El Modelo de Representación Lingüística 2-tuplas incorporado al EsIA puede facilitar el proceso de valoración de signos por los Expertos, aumentar la precisión de los resultados y con esto contribuir a la toma de una decisión más certera en la Evaluación de Impacto Ambiental.

1.7 Análisis de las soluciones internacionales

Con el desarrollo de las tecnologías cada día son más variadas las aplicaciones, software, entre otros, que realizan o sirven de apoyo en las Evaluaciones de Impacto Ambiental. Durante la investigación se encontraron las siguientes:

Sistema de Evaluación de Impacto Ambiental

El Sistema de Evaluación de Impacto Ambiental es una aplicación de escritorio que se desarrolló en la Universidad Complutense de Madrid en España por Vicente Cruz, Enrique Gallego y Luis González, es gratuito y permite el uso en el ámbito docente (30).

Dicho programa incluye elementos de inferencia mediante lógica difusa, lo cual permite tratar con problemas de incertidumbre y de conocimiento incompleto, que resulta de gran ayuda para la estimación de ciertas características que presentan gran subjetividad y suponen complicaciones para los no expertos en la materia (30). Sin embargo a pesar de que este sistema soluciona dichos problemas, todavía queda por solucionar la pérdida de información.

Aplicación del Análisis Multicriterio en la EIA

Este sistema es una aplicación de escritorio que se realizó en Barcelona en el 2004 por Luis A. García Leyton y se emplea en las Evaluaciones de Impacto Ambiental. Integra en su metodología la teoría de conjuntos difusos que permite utilizar directamente tanto información numérica como lingüística de los proyectos, sin necesidad de realizar ninguna transposición de variables a una escala numérica (3).

Dicho sistema utiliza las herramientas del software desarrolladas por Duarte (2000) y Martín-Ramos (2003) para abordar problemas en los que la imprecisión y la vaguedad están presentes, permitiendo además el manejo simultáneo de variables numéricas y lingüísticas. Para esto utilizan un sistema de computación con palabras basado en aritmética difusa (3).

La utilización de aritmética difusa hace que sea necesaria en la agregación la aproximación lingüística, por lo tanto existe pérdida de información.

Técnicas difusas en la Evaluación de Impacto Ambiental

Este programa se desarrolló en la Universidad de Granada en el 2000 por Oscar Germán Duarte Velasco y se emplea en las Evaluaciones de Impacto Ambiental. Es una aplicación de escritorio que utiliza las técnicas difusas, con el propósito de incorporar en los Estudios de Impacto Ambiental la

posibilidad de definir variables con incertidumbres. Manipula en un marco unificado las variables de tipo numérico y lingüísticos y obtiene una metodología que permite caracterizar las medidas correctoras que deben tomarse para lograr que el impacto total tenga un valor “permitido” (7).

La caracterización se realizan en términos de la Valoración Aproximada, para esto hacen uso de un sistema de computación con palabras basado en aritmética difusa (7). Sin embargo la utilización de la aritmética difusa hace que sea necesaria en la agregación la aproximación lingüística, por lo tanto existe pérdida de información.

Modelo Jerárquico de Evaluación de Impacto Ambiental empleando técnicas difusas

El sistema es una aplicación de escritorio desarrollada en la Universidad de Granada en febrero de 2008 por Darlines Sánchez y se emplea en las Evaluaciones de Impacto Ambiental. Como característica del modelo se encuentra la posibilidad de realizar evaluaciones parciales. Estas evaluaciones se efectúan de forma temporal (con el fin de realizar comparaciones en el tiempo teniendo en cuenta los cambios que puedan haber ocurrido en las zonas explotadas) y de forma espacial (permite obtener evaluaciones conociendo a priori las valoraciones desarrolladas en otras zonas del proyecto). Por otro lado y atendiendo a las necesidades planteadas por los expertos, el modelo da la posibilidad de diseñar proyectos ambientales “a medida”, es decir, dado un valor deseable de Efecto Ambiental Global, el método muestra cuál debe ser el comportamiento de las variables ambientales que intervienen en la evaluación (32).

Este modelo jerárquico incorpora principalmente técnicas difusas porque la mayoría de las variables involucradas están definidas de forma vaga, debido a que se trata de predicciones sobre los valores que podrán tomar (32), pero este modelo no resuelve problemas como la pérdida de información.

1.8 Análisis de soluciones existentes en Cuba

Evaluación en el tiempo del Impacto Ambiental con Técnicas Difusas. Aplicación en la Minería de Moa

SIAM (Sistema para la EIA) facilita el manejo de información difusa. El software tiene como objetivo aplicar técnicas difusas en la EIA para los proyectos de minería en Moa (Cuba). Además se describe una metodología difusa de EIA que permite realizar el seguimiento en el tiempo del impacto ambiental y su aplicación en el Proyecto Minero Yacimiento Punta Gorda (Moa, Cuba). Este sistema informático sugiere de manera automática los factores ambientales afectados y los impactos que provocarán las acciones que se realizarán en el proyecto. Facilita la obtención de diferentes índices de la EIA y

permite la visualización de los resultados en diferentes formatos y niveles de detalles. Incorpora métodos para la caracterización de las medidas correctoras y recomienda cuáles aplicar a partir de los impactos ambientales identificados (33).

Este sistema utiliza técnicas difusas pero no resuelve problemas como la pérdida de información, además este proyecto es exclusivamente para los procesos de minería.

1.9 Resumen del análisis de las soluciones existentes

Los sistemas mencionados fueron desarrollados mediante la aplicación de la teoría de conjuntos difusos para solucionar los problemas de incertidumbre que surgen durante las Evaluaciones de Impacto Ambiental. Algunas de estas aplicaciones emplean sistemas de computación con palabras basados en aritmética difusa para realizar las evaluaciones, permitiendo el manejo de variables numéricas y lingüísticas. Sin embargo, en todos estos sistemas para la EIA, está presente la pérdida de información en el proceso de agregación de preferencias, por lo que no se adaptan a las necesidades de la investigación que se está realizando.

Durante la investigación no se encontraron aplicaciones que permitan el intercambio de los Expertos, que se encuentren en diferentes lugares geográficos, para realizar las Evaluaciones de Impacto Ambiental.

1.10 Metodología, Lenguajes y Herramientas de desarrollo

1.10.1 Metodología de desarrollo

Programación Extrema (XP)

La metodología XP perteneciente al grupo de metodologías ágiles, se basa en la retroalimentación continua entre el cliente y el equipo de desarrollo y la comunicación fluida entre todos los participantes, utiliza además las Historias de Usuario (HU) para especificar los requisitos del software (34). Esta metodología fue seleccionada para guiar todo el proceso de desarrollo del software, ya que está pensada para equipos de desarrollo pequeños y un ciclo de desarrollo corto. Estas características son fundamentales para la investigación, ya que la ejecución del sistema de EIA está integrado por dos personas, además del cliente que es parte del equipo de desarrollo y consta de un corto plazo de tiempo para la entrega del producto terminado.

1.10.2 Herramienta de modelado

Se utiliza la herramienta profesional Visual Paradigm en su versión 8.0 (Enterprise Edition), para representar el modelamiento de los procesos del negocio del presente trabajo. Es una herramienta fácil de instalar y actualizar, multiplataforma y presenta capacidades de ingeniería directa e inversa. Potencia la reutilización del software y estandarización de la documentación. Permite modelar en BPMN (Notación para el Modelado de Procesos de Negocio) (35).

1.10.3 Lenguaje de programación

Java 1.7.3

Java es un lenguaje de programación del lado del servidor, de propósito general, basado en clases y orientado a objetos, se usa mayormente en aplicaciones web cliente-servidor. Java es multiplataforma, posee amplia compatibilidad con diferentes motores de base de datos y es de código libre, es distribuido lo que permite trabajar en ambiente de redes y contiene una gran biblioteca de clases para la utilización del protocolo TCP/IP, incluyendo HTTP, además permite el trabajo con diferentes framework como: Spring e Hibernate (36).

HTML 5

Se utiliza como lenguaje de programación en el lado del cliente HTML (HyperText Markup Language). Es un lenguaje para la definición de estilos lógicos en documentos de hipertexto, siendo el medio principal para la creación de información en internet. HTML se limita a describir la estructura y el contenido de un documento, nunca el formato de la página y su apariencia, ya que éstos son muy dependientes del navegador utilizado. Además de texto normal incluye también, elementos multimedia (gráficos, vídeo, audio) y enlaces (links) que permiten acceder a otras partes del documento o a otro sitio cualquiera de Internet. Este lenguaje es portable, es decir, se pueden visualizar las páginas con cualquier sistema operativo y, por supuesto también crearlas (37).

CSS 3

CSS (las siglas de "Cascade StyleSheet") especifica la forma del diseño de los documentos (tanto XHTML como HTML) y ayuda a mejorar el posicionamiento web. CSS brinda varias ventajas como hacer que el código de las páginas Web sea más limpio y claro. El trabajo con CSS resulta muy eficiente para separar el contenido del diseño, siendo esto muy útil cuando se quiere cambiar un aspecto del diseño de un sitio web, ya que sin hojas de estilo se tendría que cambiar página a página dicho aspecto y sin embargo, cuando se han definido hojas de estilo, se puede cambiar dicho aspecto

modificando únicamente la hoja de estilo, con lo que logramos además de un ahorro de tiempo una mayor uniformidad en el diseño (38).

1.10.4 Frameworks utilizados

Un framework es una estructura de archivos y utilidades que aceleran la programación de una aplicación informática y provee una metodología de trabajo que sistematiza y facilita la generación de formularios, funciones y módulos de uso común, permitiendo al desarrollador dedicar su atención hacia los aspectos específicos de cada aplicación (39).

JQuery

jQuery es un framework de JavaScript. Es compatible con CSS3 y con diferentes navegadores como Firefox. Su objetivo principal es hacer la programación mucho más fácil y rápida del lado del cliente. Con jQuery se pueden producir páginas dinámicas en relativamente corto tiempo. Es gratis y el código fuente puede ser modificado y adaptado a nuestras necesidades siguiendo las políticas de las licencias (MIT y GPL2) (40).

Bootstrap 2.0

Bootstrap es un framework diseñado para simplificar el proceso de creación de diseños web. Para ello ofrece una serie de plantillas CSS y de ficheros JavaScript, los cuales permiten conseguir interfaces que funcionen de manera brillante en los navegadores actuales y un diseño que pueda ser visualizado de forma correcta en distintos dispositivos y a distintas escalas y resoluciones. Además admite una mejor integración con otros frameworks que se usan habitualmente, como por ejemplo jQuery y un diseño sólido basado en herramientas actuales y potentes como CSS3/HTML (41).

Spring Framework 3.0

Es un framework de código abierto de desarrollo de aplicaciones para la plataforma Java. Por su diseño el framework ofrece mucha libertad a los desarrolladores (42).

Spring es flexible ya que está diseñado como una serie de módulos que pueden trabajar independientemente uno de otro. Además mantiene un mínimo acoplamiento entre la aplicación y el propio framework de forma que podría ser desvinculada de él sin demasiada dificultad. Es un framework ligero en términos de tamaño y costes. La mayor parte de Spring framework se puede distribuir en un único archivo .jar que pesa aproximadamente 3 MB. Permite configurar y componer complejas solicitudes de los componentes más simples. Los objetos de aplicación son compuestos de forma declarativa, por lo general en un archivo XML (42).

Hibernate Framework 3.0

Hibernate es un framework de Mapeo Objeto-Relacional (ORM) para Java que facilita el mapeo de atributos entre una base de datos relacional tradicional y el modelo de objetos de una aplicación, mediante archivos declarativos (XML) o anotaciones en los beans³ de las entidades que permiten establecer estas relaciones (43). Tiene por objetivo facilitar la persistencia de objetos Java en bases de datos relacionales y al mismo tiempo la consulta de estas bases de datos para obtener objetos.

1.10.5 Entorno integrado de desarrollo

Eclipse-índigo

El entorno integrado de desarrollo (IDE) Eclipse facilita enormemente las tareas de edición, compilación y ejecución de programas durante su fase de desarrollo. Es una plataforma ligera para componentes de software. Eclipse proporciona para el entorno de desarrollo, en el caso del lenguaje Java, disponer solamente del Java Development Kit (JDK). Se decidió utilizar Eclipse ya que es multiplataforma, permite trabajar con distintos tipos de framework como: Spring e Hibernate, además es libre y se presenta como una alternativa de fácil acceso para todos (44).

1.10.6 Sistema de gestor de base de datos

PostgreSQL 8.4.2

PostgreSQL es un sistema que brinda la posibilidad de contar con una alta calidad en la seguridad de la información manejada, posee una arquitectura cliente-servidor y posibilita la optimización de consultas. Este gestor de base de datos tiene transacciones, integridad referencial, vistas y multitud de funcionalidades, además de ser una aplicación de código abierto. Además permite el trabajo con el lenguaje de programación Java, así como la posibilidad de utilizar un framework como Hibernate (45).

1.10.7 Servidor web

Apache Tomcat 7.0.23

El servidor web Apache es una tecnología rápida y estable. Se caracteriza por presentar una arquitectura modular que permite construir un servidor que responda a los requisitos del usuario. Sirve para atender y responder a las diferentes peticiones, proporcionando los recursos que soliciten usando el protocolo HTTP o el protocolo HTTPS (la versión cifrada y autenticada). El servidor web se encarga

³ Un beans es un componente software que tiene la particularidad de ser reutilizable y así evitar la tediosa tarea de programar los distintos componentes uno a uno.

de contestar a estas peticiones de forma adecuada, entregando como resultado una página web o información de todo tipo de acuerdo a los comandos solicitados. Su rendimiento consume menos recursos del sistema en comparación con otros servidores (46).

1.11 Conclusiones del capítulo

En este capítulo se conformó el marco teórico conceptual que muestra la situación existente en el mundo sobre los distintos métodos para la EIA, así como el estudio de la metodología propuesta por Conesa y el Modelado Lingüístico Difuso.

A través de este estudio se evidenció:

- ✓ Las técnicas difusas son una herramienta útil para abordar problemas en los que la imprecisión y la vaguedad están presentes y también brindan un marco adecuado para tratar simultáneamente variables numéricas y lingüísticas.
- ✓ Al operar sobre información lingüística con el Modelado Lingüístico de Preferencias, se produce pérdida de información por concepto de aproximación que afecta la precisión de los resultados finales.
- ✓ El Modelo de Representación de Información Lingüística basado en 2-tuplas para la agregación de preferencias lingüísticas posee como ventaja ser continua en su dominio, por lo que puede representar cualquier cantidad de información del universo del discurso sin necesidad de realizar ningún proceso de aproximación, evitando la pérdida de información.
- ✓ No se encontraron en la bibliografía estudiada sistemas para realizar la EIA que permitan el intercambio de los Expertos y que eviten la pérdida de información al realizar las evaluaciones.

A partir de estos resultados se desarrollará una aplicación web que implemente el Modelo de Representación de Información Lingüística basado en 2-tuplas. Dicho sistema permitirá el intercambio con Expertos que se encuentren en diferentes lugares geográficos. Para la ejecución de este software se empleará como lenguaje de programación Java, se utilizarán los frameworks Spring e Hibernate y como metodología de desarrollo de software XP.

Capítulo 2: Características del sistema

2.1 Introducción

En el presente capítulo se aborda la propuesta de la aplicación web a implementar, detallando las características que debe tener el sistema para un mejor entendimiento. Se describe y modela el proceso del negocio y los principales artefactos generados, teniendo presente la planificación del tiempo y el esfuerzo en las fases posteriores. Se confeccionan las HU para cada iteración definida con vista a documentar los procedimientos y técnicas empleados.

2.2 Modelación del proceso del negocio

El proceso de negocio inicia con la solicitud del servicio de EIA que determinada empresa o institución realiza al CEQA, esta solicitud es aceptada o rechazada. En caso de que el Especialista acepte, se procede a la búsqueda y recogida de toda la información necesaria para realizar la EIA. Luego el Especialista reúne a todos los Expertos que estarán vinculados con esta evaluación, ellos analizan y procesan toda la información. A partir de este momento los Expertos determinan las acciones que puedan causar impactos y los factores ambientales susceptibles a ser impactados, con esta información se determinan la Matriz de Impacto y la UIP por cada factor medioambiental. Los Expertos después de obtener la matriz realizan las valoraciones de Importancia del impacto, esto se basa en la selección de preferencias de valoración por experto, en la valoración de impactos con preferencias lingüísticas y en la agregación de preferencias con igual o diferentes granularidades, entre otros. Después de realizada todas estas tareas por los Expertos, estos confeccionan la Matriz de Importancia de Impacto, con la cual el Especialista realiza un informe final y lo entrega a la empresa o institución que haya realizado la solicitud.

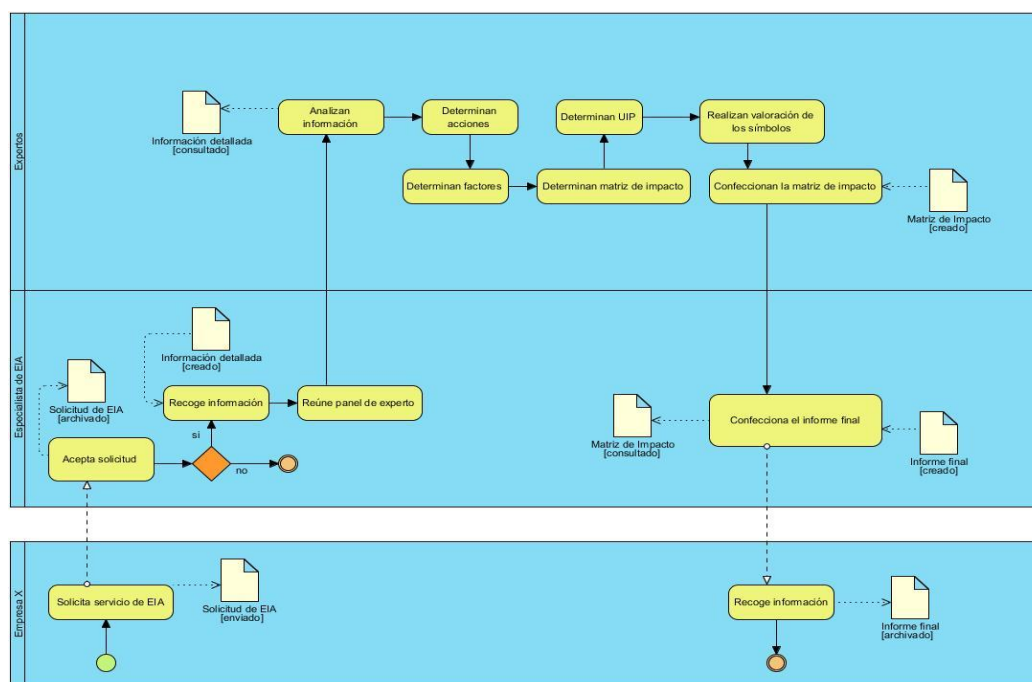


Ilustración 3: Proceso de negocio en el CEQA

2.3 Propuesta del sistema

La solución propuesta está diseñada para los especialistas encargados de realizar las Evaluaciones de Impacto Ambiental.

El sistema a desarrollar es una aplicación web el cual constará de dos módulos:

El primero tiene como objetivo la Administración de la EIA, el cual permitirá al Administrador: crear los proyectos para la EIA que le hayan sido solicitados, añadir las acciones y los factores con los cuales se realiza la evaluación, mostrar el estado de los proyectos que se estén ejecutando, archivar los proyectos que estén terminados, exportar a un documento Excel la información final de las evaluaciones realizadas a los proyectos, gestionar los conjuntos de etiquetas (tres y cinco etiquetas), además de gestionar los usuarios que realizarán las Evaluaciones de Impacto Ambiental, ya sea adicionarlos, modificarlos o eliminarlos.

El segundo permitirá a los Expertos: efectuar las Evaluaciones de Impacto Ambiental a los proyectos que le fueron asignados utilizando para esto la Matriz de Importancia de Impacto y realizando las valoraciones de los símbolos. En este módulo es donde se implementa todo el proceso del Modelado Lingüístico de Preferencia de las 2-tuplas para la unificación de información heterogénea durante las EIA. Este proceso consiste en aplicar operadores de agregación con 2-tuplas (media aritmética y

media ponderada) para obtener un único criterio de las valoraciones de los símbolos emitidas por los expertos. Además se emplea la técnica de Jerarquía Lingüística para operar sobre las etiquetas lingüísticas e integrar vectores de preferencias con diferentes granularidades.

En la aplicación la seguridad está respaldada por Spring Security, permitiendo que el acceso a la aplicación sea solo para las personas autorizadas y a las funcionalidades correspondientes según su rol dentro del sistema. Con el empleo de Spring Security se controla el manejo de sesiones (Administrador y Experto), además evita acceder al sistema a un usuario que ya esté autenticado y establece un tiempo de expiración de las sesiones. Así se garantiza la integridad de la información dentro del sistema.

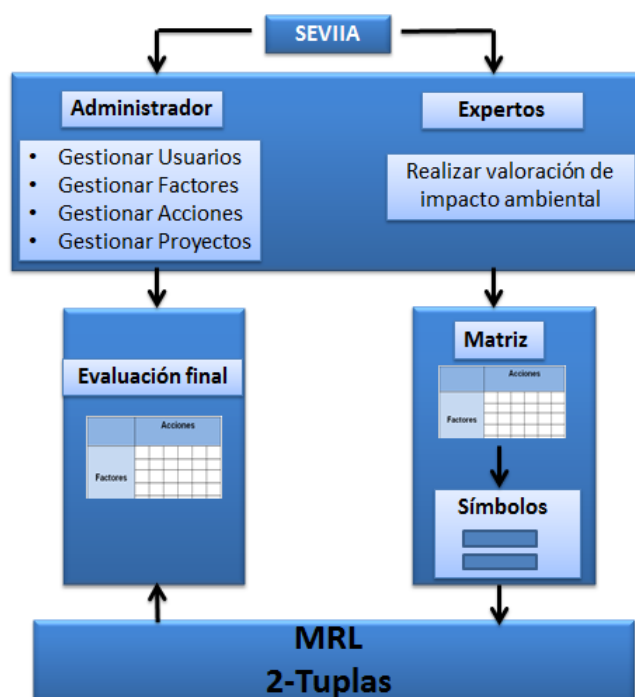


Ilustración 4: Propuesta del sistema SEVIA

2.4 Funcionalidades del sistema

A continuación se muestran las funcionalidades que desarrolla el presente sistema:

Subsistema Administración:

- Gestionar usuario.
- Crear proyecto para realizar la EIA.
- Gestionar factores.

- Gestionar acciones.
- Relacionar factores-acciones.
- Mostrar estado de los proyectos.
- Archivar proyectos terminados.
- Mostrar proyectos archivados.
- Gestionar Conjuntos.
- Exportar a Excel.

Subsistema EIA

- Mostrar proyectos asignados.
- Valoración de importancia de impactos.
 - ✓ Seleccionar preferencias de valoración por experto.
 - ✓ Valorar impactos con preferencias lingüísticas.
 - ✓ Agregar preferencias con igual o diferentes granularidades.
 - ✓ Integrar información heterogénea.
 - ✓ Calcular importancia absoluta y relativa.

2.5 Personas relacionadas con el sistema

Se definen como personas relacionadas con el sistema:

Administrador: administra y realiza todas las configuraciones necesarias para las Evaluaciones de Impacto Ambiental.

Experto: encargado de realizar las valoraciones de las Evaluaciones de Impacto Ambiental.

2.6 Lista de reserva del producto

Las listas de reserva del producto en una aplicación son muy importantes ya que son las cualidades que todo sistema debe poseer para su correcto funcionamiento. Para ello se deben tener presente los siguientes requisitos:

Usabilidad

Para el manejo de la aplicación se requiere un nivel básico de conocimiento de computación.

Seguridad

La seguridad del sistema está basada en niveles de acceso sobre las funcionalidades y la información. Los principios básicos que determinan la seguridad del sistema son los siguientes:

- La seguridad se establecerá por roles que se le asignarán a los usuarios que interactúen con el sistema, para garantizar que la información almacenada solo sea modificada y/o visualizada por los usuarios autorizados.
- Seguridad a nivel del gestor de datos, estableciendo privilegios de acceso sobre los datos almacenados y garantizando el acceso a los datos solo a aquellos usuarios con los permisos necesarios para hacerlo y solo a los datos que le esté permitido acceder de acuerdo a sus privilegios en el sistema.

Hardware

Para la instalación de la aplicación se debe disponer de:

- Una computadora servidor de 2 GB de RAM o superior, 320 GB de disco duro o superior y un procesador Intel 2.5 GHz.
- Una computadora cliente de 512 MB de RAM o superior, 80 GB de disco duro o superior y un procesador Intel 2.5 GHz.

Software

Se requiere la instalación de la máquina virtual de Java en su versión 1.7.3 y el navegador Mozilla Firefox 19.0.

2.7 Historias de Usuario

Historias de Usuario (HU)

Las HU son definidas por el cliente, en su propio lenguaje describen lo que el sistema debe realizar. Las mismas presentan detalles sobre la estimación del riesgo y cuánto tiempo será empleado en la implementación. El cliente es el encargado de asignarle una prioridad a cada HU y el equipo de desarrollo de asignarle un costo. Durante todo el progreso del proyecto existe una estrecha comunicación entre los desarrolladores y el cliente para obtener todos los detalles necesarios. Si las

HU, según lo planificado, demoran en desarrollarse se sugiere dividirlos en historias más pequeñas. También es importante destacar que las HU nuevas pueden describirse en cualquier momento, con esto se comprueba la flexibilidad de la metodología. Las HU se clasifican según:

- La prioridad en el negocio:

Alta: se le otorga a las HU que resultan funcionalidades fundamentales en el desarrollo del sistema, a las que el cliente define como principales para el control integral del sistema.

Media: se le otorga a las HU que resultan para el cliente funcionalidades a tener presente, sin que estas tengan una afectación sobre el sistema que se esté desarrollando.

Baja: se le otorga a las HU que constituyen funcionalidades que sirven de ayuda al control de elementos asociados al equipo de desarrollo y a la estructura. Estas no tienen nada que ver con el sistema que se está desarrollando.

- El riesgo en su desarrollo:

Alto: cuando en la implementación de las HU se consideran la posible existencia de errores que conlleven a la inoperatividad del código.

Medio: cuando pueden aparecer errores en la implementación de la HU que puedan retrasar la entrega de la versión.

Bajo: cuando pueden aparecer errores que serán tratados con relativa facilidad sin que traigan perjuicios para el desarrollo del proyecto.

Las HU son representadas mediante tablas divididas por las siguientes secciones:

- Número: esta sección representa el número, incremental en el tiempo, de la HU que se describe.
- Nombre de Historia de Usuario: identifica la HU que se describe entre los desarrolladores y el cliente.
- Modificación de Historia de Usuario Número: sección que representa si a la HU se le realizó alguna modificación con respecto al estado anterior.
- Usuario: se citan los desarrolladores responsables de la implementación de la HU.

- Iteración Asignada: número de la iteración donde va a desarrollarse la HU.
- Prioridad en Negocio: se le otorga una prioridad (Alta, Media, Baja) a las HU de acuerdo a la necesidad de desarrollo.
- Riesgo en Desarrollo: se le otorga una medida, de Alto, Medio, Bajo, a la ocurrencia de errores en el proceso de desarrollo de la HU.
- Puntos Estimados: es el tiempo estimado en semanas que se demorará el desarrollo de la HU.
- Puntos Reales: representa el tiempo que se demoró en realidad el desarrollo de la HU.
- Descripción: breve descripción de la HU.
- Observaciones: señalamientos o advertencias del sistema.
- Prototipo de Interface: prototipo de interfaz si aplica.

A continuación se expone una muestra de una las HU más relevante definida por el equipo de desarrollo en conjunto con el cliente.

Subsistema EIA

Tabla 2 HU Nro. 11 Valoración de importancia de impacto

Historia de Usuario	
Número: 11	Nombre de Historia de Usuario: Valoración de importancia de impacto
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Lien Noa Jiménez Katia Ramírez Bruzón	Iteración Asignada: 3
Prioridad en negocio: Alto	Puntos estimados: 3
Riesgo en Desarrollo: Alto	Puntos Reales: 3
Descripción: El Experto escoge uno de los proyectos asignados y el sistema muestra en una interfaz la matriz correspondiente a ese proyecto para que el Experto realice la valoración de los símbolos para Evaluación de Impacto Ambiental.	

Observaciones: El Experto del sistema debe estar previamente autenticado.

Prototipo de interface:

Introduzca la relación Factor-Acción :

Acciones / Factores	factor a	factor b
accion a	Evaluado	Evaluado
accion b	Evaluado	Sin relación

Finalizar

Las demás HU definidas pueden consultarse en el Anexo III.

2.8 Planificación

El proceso de planificación en la metodología XP inicia con la creación de una serie de HU que se utilizan para especificar los requisitos del software a construir. Se describen brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales.

En esta fase de planificación el cliente y desarrolladores acuerdan el orden en que deberán implementarse las Historias de Usuario, se realiza la estimación del esfuerzo que cuesta la implementación de cada HU y asociadas a estas las entregas. El criterio utilizado para medir el desempeño del proyecto en cuestión es la medida de puntos. Un punto se considera como una semana ideal de trabajo donde los miembros de los equipos de desarrollo trabajan el tiempo planeado sin ningún tipo de interrupción. Esta estimación incluye todo el esfuerzo asociado a la implementación de la HU.

En esta fase se realiza una o varias reuniones grupales de planificación y el resultado de esta fase es un Plan de Entregas.

2.8.1 Estimación de esfuerzo por Historias de Usuario

Para el desarrollo de la aplicación propuesta se aplicó la estimación de esfuerzo para cada HU identificada, llegando a los resultados que se muestran a continuación:

Tabla 3: Estimación de esfuerzo por HU

Historia de Usuario	Puntos de estimación
Gestionar usuario	2
Crear proyecto para realizar las EIA	1
Gestionar factores	1
Gestionar acciones	1
Relacionar factores-acciones	1
Mostar estado de los proyectos	1
Archivar proyectos terminados	1
Mostar proyectos archivados	1
Gestionar conjuntos	1
Exportar a Excel.	1
Valoración de importancia de impactos	3
Mostrar proyectos asignados	1

2.8.2 Plan de iteraciones

Una vez identificadas las HU y estimado el esfuerzo dedicado a la realización de cada una de estas se realiza la planificación de la etapa de implementación de la aplicación. Para una mejor organización en el desarrollo del trabajo el equipo de desarrollo dividió la implementación en 3 iteraciones:

Iteración 1:

En la primera iteración se implementarán las HU 1, 2, 3, 4, 5 dando al sistema las primeras funcionalidades que responden al subsistema Administración.

Iteración 2:

En la segunda iteración se implementarán las HU 6, 7, 8, 9 que responden al subsistema Administración.

Iteración 3:

En la tercera iteración se implementarán las HU 10, 11, 12 que responden al subsistema de EIA y de Administración.

2.8.3 Plan de duración de las iteraciones

El plan de duración de las iteraciones se encarga de mostrar las HU en el orden en que se implementarán en cada iteración, así como la duración estimada de las mismas. Las iteraciones, duraron un total de 15 semanas, divididas como sigue:

- Iteración # 1: 6 semanas.
- Iteración # 2: 4 semanas.
- Iteración # 3: 5 semanas.

El tiempo de duración de las HU en cada iteración puede verse en la tabla siguiente:

Tabla 4: Plan de duración de iteraciones

Iteración	Orden de las HU a implementar.	Duración total
1	HU-1, HU-2, HU-3, HU-4, HU-5	6 semanas
2	HU-6, HU-7, HU-8, HU-9,	4 semanas
3	HU-10, HU-11, HU-12,	5 semanas

2.8.4 Plan de entregas

Se presenta el Plan de Entrega estimado para la fase de implementación, detallando la fecha fin para cada una de las iteraciones definidas.

Tabla 5: Plan de entregas

Iteración	Fecha de Entrega
1	1 de abril del 2013
2	29 de abril del 2013
3	3 de junio del 2013

2.9 Conclusiones del capítulo

En este capítulo se realizó la descripción del proceso de negocio del CEQA, se obtuvo una mejor visión sobre lo que el cliente desea y con estos resultados:

- ✓ Se logró un mejor entendimiento de la aplicación a través de la propuesta del sistema.
- ✓ Se puntualizaron las necesidades de los Expertos y del Administrador, definiendo las funcionalidades del sistema.
- ✓ Se describió las cualidades que el sistema poseerá para su correcto funcionamiento.
- ✓ Se definió lo que la aplicación realizará, para esto se conformaron las HU relacionadas con el sistema que fueron creadas para cada versión del programa.

Finalmente se obtuvo una planificación del tiempo de desarrollo de las iteraciones y el orden en que se implementarán las HU.

Capítulo 3: Diseño del sistema

3.1 Introducción

En este capítulo se generan los artefactos correspondientes a la fase de diseño. En cada una de las iteraciones se definen las tarjetas CRC (Contenido, Responsabilidad, Colaboración) como herramienta de reflexión en el diseño de software orientado a objetos. Además se aborda acerca de la arquitectura y patrones de diseño empleados en la aplicación.

3.2 Patrón de arquitectura

N-Capas

La arquitectura N-Capas divide el procesamiento en niveles independientes que se distribuyen entre el cliente y el servidor. Las aplicaciones con N-Capas se dividen en varios niveles, uno de presentación, uno intermedio y una capa de datos (47).

Entre las ventajas de la arquitectura en capas se encuentran las siguientes:

- Soporta un diseño basado en niveles de abstracción crecientes, lo cual a su vez permite a los implementadores la partición de un problema complejo en una secuencia de pasos incrementales.
- El desarrollo se lleva a cabo en varios niveles y en el caso de que exista algún error o la necesidad de algún cambio obligatorio, solo es necesario cambiar el nivel en cuestión, sin afectar el correcto funcionamiento del resto del sistema.
- Posibilita la reutilización de código debido a las características propias de su implementación en capas.

El framework Spring permite una separación entre el código de modelo de dominio y las formas web. A continuación se presenta una representación de la arquitectura N-Capas basada en el Framework Spring que se utilizó en el Sistema de EIA:

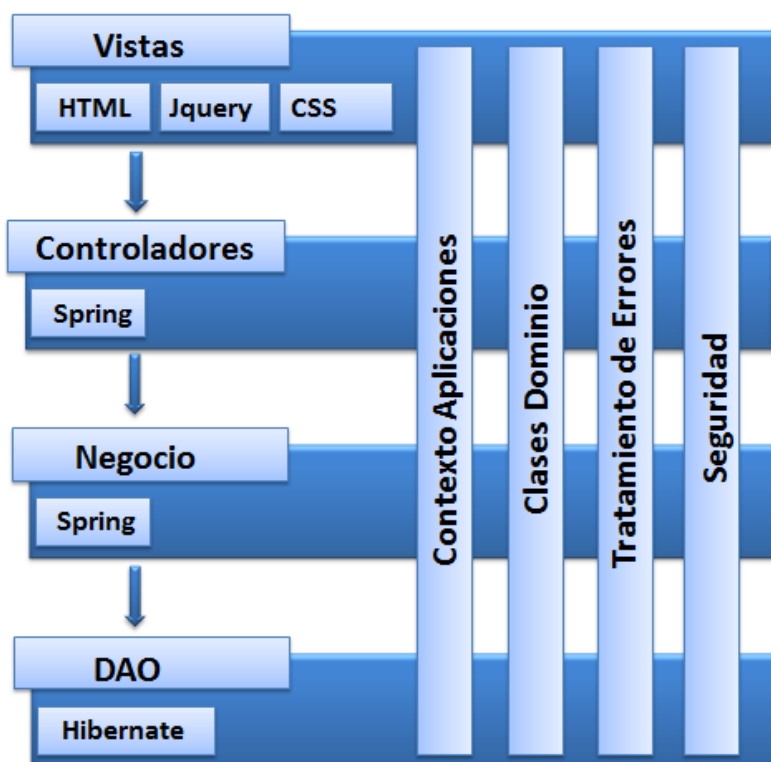


Ilustración 5: Arquitectura N-Capas basada en el Framework Spring

El sistema en desarrollo se divide en cuatro capas distintas:

Capa Vistas: contiene las clases encargadas de garantizar la interacción del usuario con el sistema. Esta capa se comunica únicamente con la capa controladora a la cual envía los datos a través de los formularios.

Capa Controladores: es la encargada de recibir los eventos de entrada desde las vistas. Envía las peticiones a las clases del negocio que correspondan y devuelve los datos solicitados por el usuario a la capa vista.

Capa Negocio: se implementan todas las funcionalidades y métodos del sistema. Recibe las peticiones de los controladores y les da respuesta accediendo, en el caso de que sea necesario, a la capa DAO y devuelve los datos al controlador.

Capa DAO: es la encargada de realizar las gestiones a la base de datos, su función es almacenar y devolver datos a la capa de negocio.

Spring permite que estén presentes las excepciones y la seguridad en las cuatro capas. Las excepciones se pueden originar en cualquier parte de la aplicación por lo que hay que darle un adecuado tratamiento, las mismas son manejadas por los frameworks Hibernate y Spring y la seguridad por Spring Security.

A continuación se muestra un ejemplo de cómo se realiza el tratamiento de errores en la aplicación:

```
} catch (Exception e) {  
    // TODO Auto-generated catch block  
    model.addAttribute("error", e.getMessage());  
}
```

Ilustración 6: Tratamiento de errores

3.3 Patrones de diseño

Los patrones de diseño son la base para la búsqueda de soluciones a problemas comunes en el desarrollo de software y otros ámbitos referentes al diseño de interacción o interfaces. Un patrón es una solución a un problema de diseño no trivial. Para que esta solución sea considerada un patrón se debe comprobar la efectividad, lo que significa que la solución ha valido para resolver el problema en diseños pasados y además debe ser reusable, es decir, aplicable a diferentes problemas de diseño en distintas circunstancias (48).

Patrones de diseño del Framework Spring utilizados en el Sistema de EIA

Singleton (Instancia única)

Está diseñado para restringir la creación de objetos pertenecientes a una clase o el valor de un tipo a un único objeto. Su intención consiste en garantizar que una clase solo tenga una instancia y proporcionar un punto de acceso global a ella (42).

Este patrón está presente, dentro de la aplicación, en el uso que le da Hibernate a los beans para instanciar los objetos de la base de datos. Además las instancias únicas se pueden encontrar en las clases del negocio y en los controladores del sistema.

IOC (Inversión del control)

El flujo de ejecución de un programa se invierte respecto a los métodos de programación tradicionales, en los que la interacción se expresa de forma imperativa haciendo llamadas a procedimientos o

funciones. Permite invertir el camino en el que un objeto obtiene las referencias a otros objetos de los cuales depende (42).

Este patrón está presente las capas DAO, Negocio y Controladores, donde a través de la anotación @Autowire se realizan las inyecciones de dependencia.

En el Anexo IV se puede encontrar un ejemplo de este patrón a través de la anotación @Autowire.

DAO (Data Access Object)

Es un componente de software que suministra una interfaz común entre la aplicación y uno o más dispositivos de almacenamiento de datos, tales como una base de datos o un archivo. Su uso se extiende al problema de encapsular no solo la fuente de datos, sino además ocultar la forma de acceder a los datos. Se trata de que el software cliente se centre en los datos que necesita y se olvide de cómo se realiza el acceso a los datos o cuál es la fuente de almacenamiento (42).

Este patrón está presente en las clases que permiten el acceso a los objetos de la base de datos, por ejemplo en las clases del paquete DAO.

En el Anexo IV se puede encontrar un ejemplo de este patrón a través del paquete dao.

Front Controller (Controlador Frontal)

Es un patrón de diseño que se basa en usar un controlador como punto inicial para la gestión de las peticiones. El controlador gestiona estas peticiones y realiza algunas funciones como: comprobación de restricciones de seguridad, manejo de errores, mapeo y delegación de las peticiones a otros componentes de la aplicación que se encargarán de generar la vista adecuada para el usuario (42).

Este patrón se evidencia en el dispatcher servlet⁴ de Spring, es decir en el fichero sevia-servlet.xml de la aplicación.

En el Anexo IV se puede encontrar un ejemplo del fichero sevia-servlet.xml.

Patrones para Asignar Responsabilidades (GRASP)

Los patrones GRASP describen principios fundamentales de la asignación de responsabilidades dentro de la programación orientada a objetos mediante la aplicación de "buenas prácticas" recomendables en el diseño de software. Para el desarrollo del sistema se usaron los siguientes patrones GRASP (48):

⁴ El dispatcher servlet se encarga de redireccionar las peticiones a los controladores responsables de manejarlas.

Experto

Es el principio básico de asignación de responsabilidades que suele utilizarse en el diseño orientado a objetos; con él que se pretende designar una idea clara de la forma más eficiente posible. El patrón experto indica que la responsabilidad de la creación de un objeto o la implementación de un método debe recaer sobre la clase que conoce toda la información necesaria para crearlo, es decir, una clase contiene toda la información necesaria para realizar la labor que tiene encomendada (48).

Este patrón se ve reflejado en las clases del negocio ya que cada una tiene su función específica. Un ejemplo de esto se evidencia en la clase ProyectoManagerImpl la cual se encarga de realizar todas las funcionalidades relacionadas con los proyectos.

En el Anexo IV se evidencia un ejemplo de este patrón.

Creador

Este patrón ayuda a identificar quién debe ser el responsable de la creación (o instanciación) de nuevos objetos o clases (48).

Este patrón se evidencia en la clase IntervaloManagerImpl que es la encargada de crear las instancias de la clase Intervalo.

En el Anexo IV se puede encontrar un ejemplo de este patrón.

Controlador

Es un patrón que sirve como intermediario entre la interfaz y el algoritmo que la implementa, de tal forma que es el que recibe los datos del usuario y el que los envía a las distintas clases según el método llamado. Sugiere que la lógica de negocios debe estar separada de la capa de presentación, esto posibilita la reutilización de código y a la vez tener un mayor control (48).

En la presente aplicación cada una de las vistas cuenta con un controlador para manejar sus eventos. Un ejemplo de esto se evidencia en el controlador SalvarMatrizProyectoController el cual se encarga de atender los eventos de la vista matrizImportancia.jsp.

En el Anexo IV se puede encontrar un ejemplo de este patrón.

Bajo Acoplamiento

El bajo acoplamiento fomenta el aumento de la reutilización y la eliminación de las redundancias, creando clases más independientes y con mayor resistencia al impacto de los cambios, que aumentan la productividad y la posibilidad de reutilización.

Este patrón se basa en la idea de tener las clases lo menos entrelazadas posible. De tal forma que en caso de producirse una modificación en alguna de ellas, se tenga la mínima repercusión posible en el resto de clases, potenciando la reutilización y disminuyendo la dependencia entre las clases (48).

Spring plantea el uso de interfaces para evitar un alto acoplamiento, por lo que en la aplicación este patrón se ve claramente reflejado, ya que tanto la capa de negocio como la capa de acceso a datos tienen interfaces que son implementadas por otras clases. Por ejemplo la clase Interface IFactorManager que es implementada por la clase FactorManagerImpl.

En el Anexo IV se evidencia un ejemplo de este patrón a través de los paquetes business y dao.

Alta Cohesión

Al igual que el patrón Bajo Acoplamiento, Alta Cohesión es un principio que se debe tener presente en todas las decisiones de diseño, es la meta principal que ha de buscarse en todo momento.

La cohesión es una medida de cuán relacionadas y enfocadas están las responsabilidades de una clase, además de que una alta cohesión garantiza que clases con responsabilidades estrechamente relacionadas no realicen un trabajo enorme y que cada elemento del diseño realice una labor única dentro del sistema (48).

Anteriormente se representó el uso del patrón experto, el cual distribuye el comportamiento entre las clases que cuentan con la información, alentando con ello definiciones de clases sencillas y más cohesivas que son más fáciles de comprender y mantener. Así se brinda soporte a la Alta Cohesión.

3.4 Tarjetas Clase-Responsabilidad-Colaborador (CRC)

Para el diseño de la aplicación la metodología XP propone el uso de las tarjetas Clase-Responsabilidad-Colaborador (CRC) con el objetivo de desarrollar una representación organizada de las clases.

Un modelo CRC es en realidad una colección de tarjetas índices estándar que representan clases (35). Las tarjetas se dividen en tres secciones:

Nombre de la clase: se coloca en el borde superior en forma de título. La clase es cualquier evento, individuo, objeto, concepto, pantalla o reporte.

Colaboradores: se coloca en la parte derecha. Los colaboradores de una clase son aquellas clases con las que trabaja en conjunto para llevar a cabo sus funcionalidades.

Responsabilidades: se coloca en el extremo izquierdo. Son las obligaciones que tiene un objeto con respecto a su comportamiento. Se asignan a los objetos en el diseño y son acciones que realizan atributos y métodos.

Nombre de la Clase	
Responsabilidades	Colaboradores

Ilustración 7: Estructura de la Tarjeta CRC

A continuación se definen las tarjetas CRC más relevantes correspondientes a las clases del negocio.

Tabla 6: Clase Usuario

Clase UsuarioManagerImpl	
Responsabilidades	Colaboradores
Adiciona los usuarios del proyecto	IUsuarioManager
Modifica los usuarios del proyecto	Usuario
Edita los usuarios del proyecto	CrearUsuarioController
Elimina los usuarios del proyecto	EditarUsuarioController
	EliminarUsuarioController

Tabla 7: Clase Proyecto

Clase ProyectoManagerImpl	
Responsabilidades	Colaboradores

Crea un proyecto	IProyectoManager
Cambia el estado de un proyecto	Proyecto
Edita un proyecto	CrearProyectoController
Calcula porciento	
Lista los proyectos	
Buscar proyectos por nombre	

Las demás Tarjetas CRC definidas pueden consultarse en el Anexo V.

3.5 Diseño de la Base de Datos

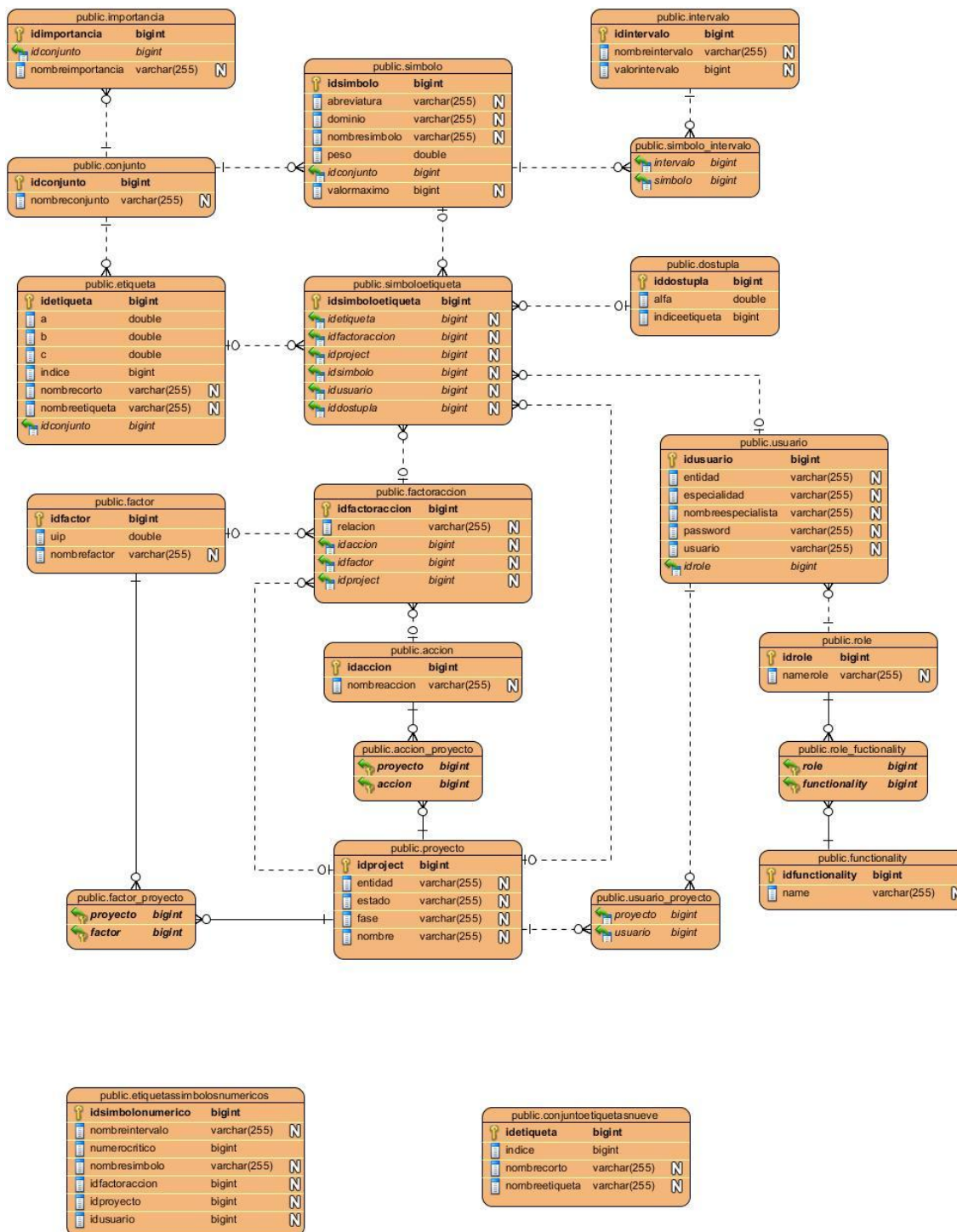


Ilustración 8: Modelo de datos del sistema

En el Anexo VI se pueden consultar las descripciones de las tablas del modelo de datos del sistema.

3.6 Conclusiones

La realización de este capítulo posibilitó:

- ✓ Un mejor entendimiento de la aplicación a través de la arquitectura empleada y los patrones de diseño.
- ✓ Representar organizadamente las clases más relevantes del sistema a través de las tarjetas CRC.
- ✓ Obtener una mejor visión de la organización de los datos dentro de las tablas en la base de datos.

Finalmente a partir de estos resultados se obtuvo el diseño del sistema.

Capítulo 4: Implementación y Prueba

4.1 Introducción

En este capítulo se describe la fase de implementación y prueba y se detallan las tareas de la ingeniería generadas por las correspondientes HU definidas en la metodología XP. Además se realizan las pruebas al software con el propósito de garantizar una adecuada implementación del sistema informático.

4.2 Tareas de la ingeniería

La metodología XP propone que la implementación debe ejecutarse de forma iterativa e incremental, alcanzando al final de cada iteración un producto funcional que debe ser examinado y mostrado al cliente, de esta forma se garantiza una constante retroalimentación entre los desarrolladores y clientes. Como parte de este plan se exponen las tareas de la ingeniería generadas por cada HU las cuales son asignadas a los programadores para ser implementadas durante la iteración correspondiente.

Las tareas de la ingeniería serán representadas mediante tablas divididas por las siguientes secciones:

- Número de Tarea: sección que representa el número de la tarea a desarrollar, los mismos deben ser consecutivos.
- Número Historia de Usuario: número de la HU a la que pertenece la tarea.
- Nombre Tarea: nombre que identifica a la tarea.
- Tipo de Tarea: las tareas pueden ser de Desarrollo, Corrección, Mejora y Otra, en este último caso se debe especificar.
- Puntos Estimados: tiempo (estimado) en días que se le asignará al desarrollo de la tarea.
- Programador Responsable: nombre y apellidos del programador encargado de desarrollar la tarea.
- Descripción: breve descripción de la tarea.

A continuación se muestra una de las tareas de la ingeniería más relevante correspondiente a la HU:

Tabla 8: Tarea Nro. 18 Valoración de importancia de impacto

Tarea de Ingeniería

Número Tarea: 18	Número Historia de Usuario: 11
Nombre Tarea: Valoración de importancia de impacto	
Tipo de Tarea: Desarrollo	Puntos Estimados: 15
Programador Responsable: Lien Noa Jiménez Katia Ramírez Bruzón	
Descripción: El Experto escoge uno de los proyectos asignados y el sistema muestra en una interfaz la matriz correspondiente a ese proyecto para que el Experto realice la valoración de los símbolos para la Evaluación de Impacto Ambiental.	

Las demás tareas de la ingeniería definidas pueden consultarse en el Anexo VII.

4.3 Validación del diseño

El IEEE⁵ define métrica como una medida cuantitativa del grado en que un sistema, componente o proceso posee un atributo dado (43).

Las métricas del software permiten medir de forma cuantitativa la calidad de los atributos internos del producto, esto permite al ingeniero evaluar la calidad durante el desarrollo del sistema. Son varios los puntos de vista relacionados con la calidad del software. Las métricas de diseño se concentran en las características internas de los componentes del software con medidas que pueden ayudar al desarrollador a juzgar la calidad de un diseño a nivel de componente (43).

Un aspecto importante a tener en cuenta en la evaluación del diseño, es la creación de métricas básicas inspiradas en el estudio de la calidad del diseño orientado a objetos. Los atributos de calidad que se tienen en cuenta son (43):

Responsabilidad: es la responsabilidad asignada a una clase en un marco de modelado de un dominio o concepto de la problemática propuesta.

Complejidad de implementación: es la complejidad de implementación que posee una estructura de diseño de clases.

⁵ Instituto de Ingenieros Eléctricos y Electrónicos, del inglés Standard Glossary of Software Engineering Terms.

Reutilización: es el grado de reutilización, presente en una clase o estructura de clase, dentro de un diseño de software.

Acoplamiento: es 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: es 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 costes y la planificación del proyecto.

Cantidad de pruebas: es el número o el grado de esfuerzo para realizar las pruebas de calidad (Unidad) del producto (componente, módulo, clase, conjunto de clases, etc.) diseñado.

Las métricas seleccionadas como instrumento para evaluar la calidad del diseño del módulo son las siguientes:

Tamaño Operacional de Clase (TOC): está dado por el número de métodos asignados a una clase y evalúa los siguientes atributos de calidad:

Tabla 9: Atributos de calidad evaluados por la métrica TOC

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

Se definen los siguientes criterios y categorías de evaluación para los atributos de calidad anteriores:

Tabla 10: Criterios de evaluación para la métrica TOC

Atributos	Categoría	Criterio
Responsabilidad	Baja	< =Prom.

	Media	Entre Prom. Y 2* Prom
	Alta	> 2* Prom
Complejidad de implementación	Baja	< =Prom
	Media	Entre Prom. y 2* Prom
	Alta	> 2* Prom
Reutilización	Baja	> 2*Prom
	Media	Entre Prom. y 2* Prom
	Alta	<= Prom

Relaciones entre Clases (RC): está dado por el número de relaciones de uso de una clase con otra y evalúa los siguientes atributos de calidad:

Tabla 11: Atributos de calidad evaluados por la métrica RC

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

Se definen los siguientes criterios y categorías de evaluación para los atributos de calidad anteriores:

Tabla 12: Criterios de evaluación para la métrica RC

Atributos	Categoría	Criterio
-----------	-----------	----------

Acoplamiento	Ninguno	0
	Bajo	1
	Medio	2
	Alto	>2
Complejidad de mantenimiento	Baja	< =Prom
	Media	Entre Prom. y 2* Prom
	Alta	> 2* Prom
Reutilización	Baja	> 2*Prom
	Media	Entre Prom. y 2* Prom
	Alta	<= Prom
Cantidad de Pruebas	Baja	< =Prom
	Media	Entre Prom. y 2* Prom
	Alta	> 2* Prom

Resultados obtenidos de la aplicación de la métrica TOC

Luego de aplicarse la métrica de diseño TOC se obtuvieron resultados que permiten evaluar el diseño propuesto de calidad aceptable, teniendo en cuenta que el 79% de las clases empleadas en el sistema poseen 5 operaciones o menos, lo que conlleva a evaluaciones positivas de los atributos de calidad involucrados (responsabilidad, complejidad de implementación y reutilización).

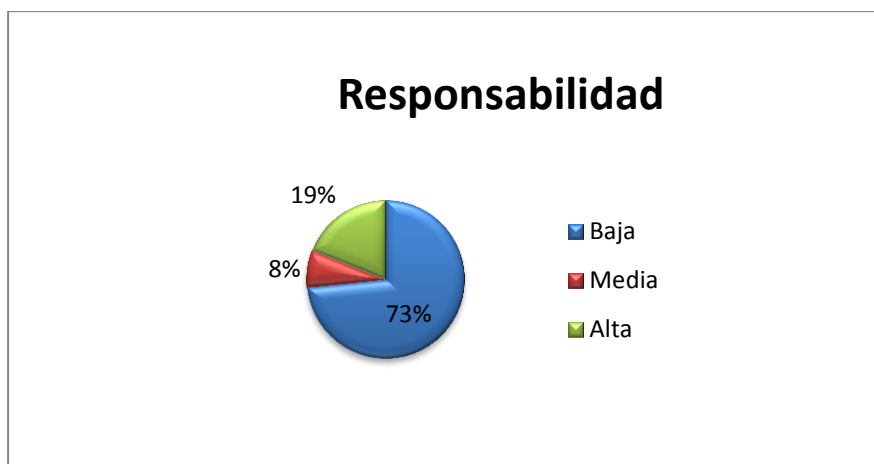


Ilustración 9: Resultados de la métrica TOC para el atributo Responsabilidad

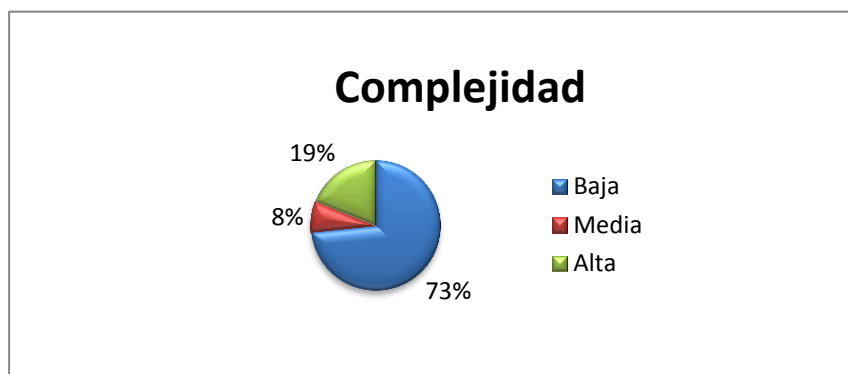


Ilustración 10: Resultados de la métrica TOC para el atributo Complejidad

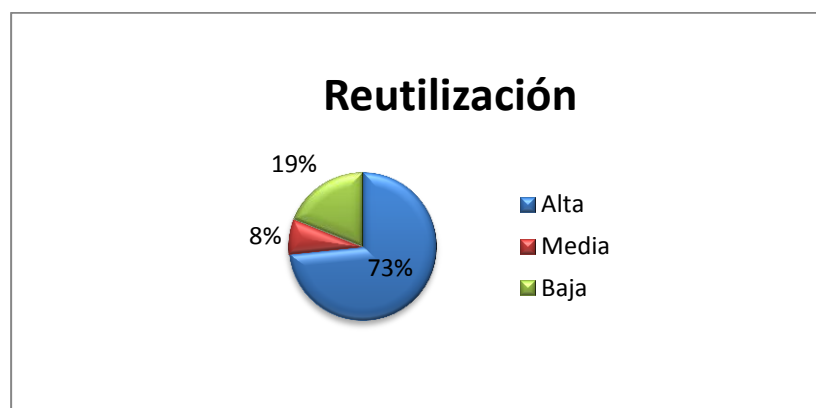


Ilustración 11: Resultados de la métrica TOC para el atributo Reutilización

Resultados obtenidos de la aplicación de la métrica RC

Luego de aplicarse la métrica de diseño RC se determinó que la aplicación cuenta con un total de 79 clases y 160 dependencias entre ellas, con un promedio de 2,02 relaciones por clases.

Se puede concluir que el diseño propuesto se puede evaluar de calidad aceptable ya que el 74% de las clases empleadas cuentan con dos o menos relaciones de uso con otras clases, lo que conlleva a una evaluación positiva del atributo de calidad acoplamiento. Los atributos de calidad complejidad de mantenimiento y cantidad de pruebas demostraron resultados positivos, con un valor de 75% del total. Para el atributo de calidad reutilización se concluye que de manera general el diseño cuenta con un 75% de alta reutilización. Estos resultados favorecen la reutilización de las clases así como la modificación e implantación del diseño.

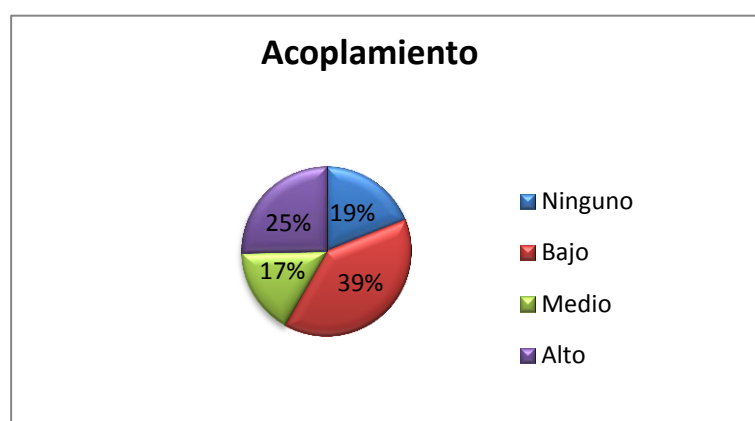


Ilustración 12: Resultados de la métrica RC para el atributo Acoplamiento

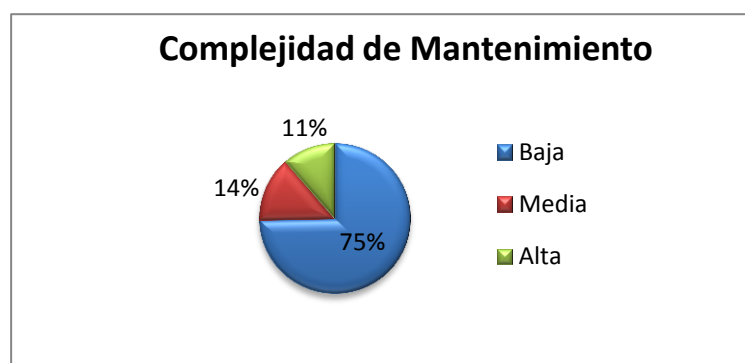


Ilustración 13: Resultados de la métrica RC para el atributo Complejidad de Mantenimiento

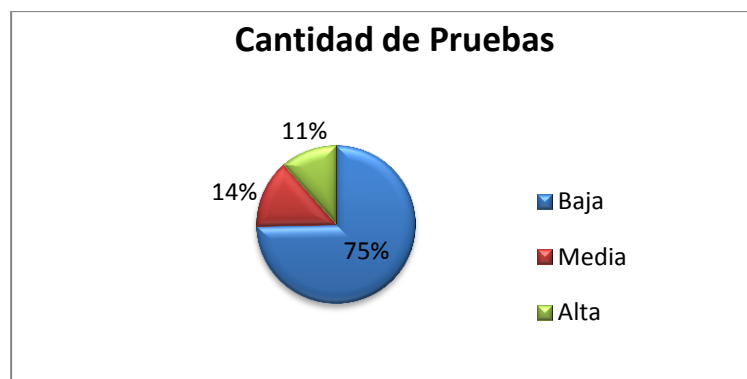


Ilustración 14: Resultados de la métrica RC para el atributo Cantidad de Pruebas

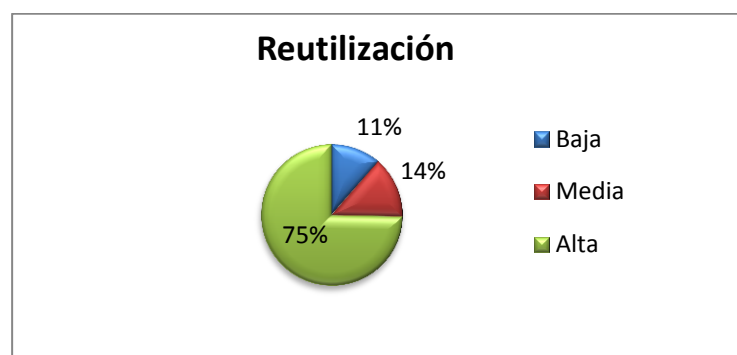


Ilustración 15: Resultados de la evaluación de la métrica RC para el atributo Reutilización

4.4 Pruebas

Como parte de la metodología de desarrollo XP se realizan pruebas constantemente, tanto como sea posible, a lo que se va construyendo. Esto permite aumentar la calidad de los sistemas reduciendo el número de errores no detectados y disminuyendo el tiempo transcurrido entre la aparición de un error y su detección. XP divide las pruebas del sistema en dos grupos: pruebas unitarias y pruebas de aceptación, destinadas a evaluar si al final de una iteración se consiguió la funcionalidad requerida diseñada para el cliente final.

Pruebas unitarias

La base de este método es el hacer pruebas en pequeños fragmentos del código de la aplicación. Estos fragmentos deben ser unidades estructurales del programa encargados de una tarea específica, en programación orientada a objetos se puede afirmar que estas unidades son los métodos o las funciones que se tienen definidos. El objetivo de estas pruebas es el aislamiento de partes del código y la demostración de que no contienen errores. Estas no son directamente palpables para el cliente.

Las pruebas unitarias fueron desarrolladas constantemente cada vez que se terminaba de implementar alguna funcionalidad probándola directamente en el entorno real. Dichas pruebas fueron realizadas con junit⁶ a las clases del negocio de la aplicación.

A continuación se muestra el resultado obtenido al realizar las pruebas unitarias con junit:

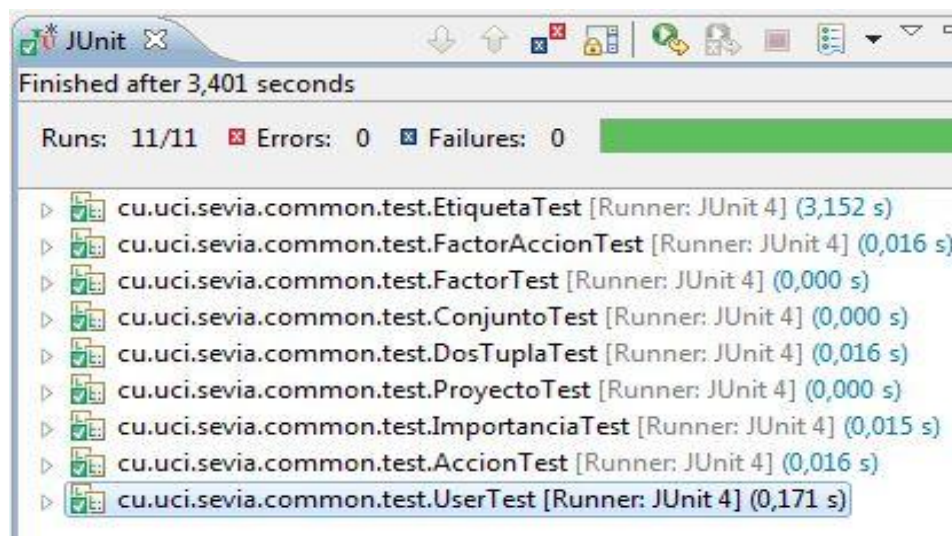


Ilustración 16: Resultado de las pruebas unitarias con junit

Prueba de aceptación

En este tipo de prueba se evalúa el grado de calidad del software con relación a todos los aspectos relevantes para que el uso del producto se justifique. Las pruebas de aceptación son la mejor forma de probar la aplicación de extremo a extremo, ya que prueban todas las capas de la aplicación: el sistema de enrutamiento, el modelo, las acciones y las plantillas, en otras palabras, lo que se hace es probar un escenario correspondiente a la HU que se acaba de implementar en la aplicación.

Las pruebas de aceptación correspondientes a cada una de las funcionalidades serán representadas mediante tablas divididas por las siguientes secciones:

- **Clases Válidas:** se realiza la descripción de cada uno de los pasos seguidos durante el desarrollo de la prueba y se tiene presente cada una de las entradas válidas que hace el usuario con el objetivo de ver si se obtiene el resultado esperado.
- **Clases Inválidas:** se realiza la descripción de cada uno de los pasos seguidos durante el desarrollo de la prueba y se tiene presente cada una de las posibles entradas inválidas que

⁶ JUnit es un conjunto de bibliotecas que son utilizadas en programación para hacer pruebas unitarias de aplicaciones Java.

hace el usuario con el objetivo de ver si se obtiene el resultado esperado y cómo responde el sistema.

- **Resultado Esperado:** se realiza una breve descripción del resultado que se espera ya sea para entradas válidas o entradas inválidas.
- **Resultado de la Prueba:** se realiza una breve descripción del resultado que se obtiene.
- **Observaciones:** algún señalamiento o advertencia que sea necesario hacerle a la sección que se está probando.

Las pruebas de aceptación se llevan a cabo redactando los casos de prueba, teniendo en cuenta el orden de las HU y la prioridad que ha sido asignada a las funcionalidades. Luego se realiza la planificación con el cliente de cuándo y cuáles pruebas serán llevadas a cabo. Finalmente, se completa cada uno de los campos de las tablas de las pruebas de aceptación con el resultado de la prueba.

Las pruebas de aceptación definidas pueden consultarse en el Anexo VIII.

En el Anexo IX se evidencia el aval que testifica la aceptación del sistema por parte del cliente.

A continuación se exponen los resultados alcanzados de realizar las pruebas de aceptación.

Tabla 13: Resultado de las pruebas de aceptación

Iteraciones	Total NC	NC aplicación	NC documento
1	21	14	7
2	11	9	2
3	0	0	0

En la tabla anterior se muestran los resultados obtenidos de las pruebas de aceptación. En la primera iteración se detectaron 21 No Conformidades (NC), 14 de ellas de aplicación y 7 de documentos. En la segunda iteración se obtuvieron 11 NC, 9 de aplicación y 2 de documento. Al realizar la última iteración no se detectaron NC puesto que las mismas fueron resueltas en las anteriores iteraciones.

4.5 Conclusiones

En el presente capítulo se presentaron los resultados obtenidos de la validación del diseño y de las pruebas realizadas a la aplicación, tanto unitarias como de aceptación. Estos mostraron que la

aplicación presenta un diseño favorable y que cumple con las funcionalidades requeridas ya que los resultados son los esperados, lo que permitió:

- ✓ La aplicación de las métricas para validar el diseño, las cuales arrojaron resultados satisfactorios sobre el diseño realizado.
- ✓ La realización al sistema de las pruebas unitarias para demostrar que el código no contiene errores y de las pruebas de aceptación para comprobar el correcto funcionamiento de la aplicación, lográndose con ello la aceptación del cliente.

Conclusiones generales

Con el desarrollo del presente trabajo se obtuvieron varios resultados que se muestran a continuación:

- ✓ Las soluciones estudiadas, para realizar la EIA, arrojaron que estas no son eficientes en el manejo de la incertidumbre que se presenta en las valoraciones emitidas por los Expertos.
- ✓ Se diseñó e implementó una aplicación que emplea el Modelo de Representación de Información Lingüística de las 2-tuplas, para evitar la pérdida de información en la agregación de preferencias por concepto de aproximación.
- ✓ Se aplicaron las pruebas unitarias y de aceptación al software. Estas pruebas arrojaron resultados positivos con lo que se verificó la correcta implementación de las funcionalidades del sistema.

Finalmente se concluye que los objetivos propuestos para el presente trabajo se han cumplido satisfactoriamente, poniendo en práctica cada una de las funcionalidades de los módulos de la aplicación para la realización de las Evaluaciones de Impacto Ambiental y logrando la aceptación del sistema por parte del cliente.

Recomendaciones

Una vez terminado el desarrollo de la aplicación se recomienda:

- ✓ Añadir una funcionalidad que permita establecer orden de prioridad entre los proyectos.
- ✓ Incluir una funcionalidad que permita notificar a los Especialistas, mediante el envío de correos electrónicos, los proyectos que le han sido asignados y detalles de los mismos.

Referencias bibliográficas

1. **Fernández, Vicente Conesa.** *Guía Metodológica para la Evaluación de Impacto Ambiental.* Madrid : Mundi-prensa, 2000.
2. **Cuba, Asamblea Nacional del Poder Popular de.** *Ley 81 "Medio Ambiente".* La Habana : Gaceta Oficial de la República de Cuba, 1997.
3. **Leyton, Luis Alberto García.** *Aplicación del Análisis Multicriterio en las Evaluaciones de Impacto Ambiental.* Barcelona : s.n., 2004.
4. **Leopold, LB, y otros, y otros.** *A Procedure for Evaluating Environmental Impact.* Washintong : Geologica Survey, Circular 645, 1971.
5. **J, Baker N, Drobny, N K y Fahringer, D.** *Environmental evaluation system for water resource planning (to Bureau of Reclamation, U.S. Department of Interior).* Battelle Columbus Laboratory, Columbus, Ohio : s.n., 1972.
6. **J, Baker N, y otros, y otros.** *An environmental evaluation system for water resource planning.* Water Resources Research : s.n., 1973. Vol 9, No.3.
7. **G, O Duarte.** *Técnicas Difusas en la Evaluación de Impacto Ambiental.* Universidad de Granada : s.n., 2000.
8. **López, Luis Martínez.** *UN NUEVO MODELO DE REPRESENTACION DE INFORMACION LINGÜÍSTICA BASADO EN 2-TUPLAS PARA LA AGREGACION DE PREFERENCIAS LINGÜÍSTICAS.* Universidad de Granada : s.n., 1999.
9. **Estévez, Macarena Espinilla.** *Nuevos Modelos de Evaluación Sensorial con Información Lingüística.* España : s.n., 2009.
10. **Bordogna, G y Passi, G.** *A Linguistic Modeling of Consensus in Group Decision Making Based on OWA operators, IEEE Trans. on Systems,.* 1992.
11. **Delgado, M, Verdegay, J L y Vila, M A.** *Linguistic Decision Making Models, Int. J. of Intelligent Systems .* 1992.
12. **Herrera, F, Herrera-Vierdma, E y Verdegay, J L.** *Direct Approach Processes in Group Decision Making Using Linguistic OWA Operators, Fuzzy Sets and Systems .* 1996.
13. **M, Lui C, Wang, M J y Pang, Y S.** *A Multiple Criteria Linguistic Decision Model (MCLDM) for Human Decision Making, European Journal of Operational Research.* 1994.
14. **Marimin, M, y otros, y otros.** *Linguistic Labels for Expressing Fuzzy Preference Relations in Fuzzy Group Decision Making, IEEE Trans. on Systems, Man and Cybernetics.* 1998.
15. **Tong, M y Bonissone, P P.** *A Linguistic Approach to Decision Making with Fuzzy Sets, IEEE Transactions on Systems, Man and Cybernetics 10 .* 1980.
16. **R, R Yager.** *An Approach to Ordinal Decision Making, Int. J. of Approximate Reasoning.* 1995.
17. **A, L Zadeh.** *The Concept of a Linguistic Variable and Its Applications to Approximate Reasoning.* 1975.
18. **Dubois, D y Prade, H.** *Fuzzy Sets and Systems: Theory and Applications,.* New York : s.n., 1980.
19. **P, P Bonissone y Decker, K S.** *Selecting Uncertainty Calculi and Granularity: An Experiment in Trading-precision and Complexity, en: L.H. Kanal and J.F. Lemmer, Eds. North-Hollanda : s.n., 1986.*
20. **Delgado, M, Verdegay, J L y Vila, M A.** *On Aggregation Operations of Linguistic Labels, Int. J. of Intelligent Systems.* 1993.
21. **Canter, L y Sandle, B.** *A tool kit for effective EIA practice review of methods and perspectives on their application A Supplementary Report of the International Study of the Effectiveness of Environmental Assessment. International Association for Impact Assessment.* 1997.
22. **Cordón, O, Herrera, F y Zwir, I.** *Linguistic modeling by hierarchical systems of linguistic rules. IEEE Transactions on Fuzzy Systems.* 2001.
23. **Herrera, F y Martinez, L.** *A model based on linguistic 2-tuples for dealing with multigranularity hierarchical linguistic contexts in multiexpert decision-making. IEEE Transactions on Systems, Man, and Cybernetics. Part B: Cybernetics.* 2001.

24. **Herrera, F y Martínez, L.** *A 2-tuple fuzzy linguistic representation model for computing with words.* *IEEE Transactions on Fuzzy Systems.* 2000.
25. **Cruz, V, Gallego, E y González, L.** *Sistema de Evaluación de Impacto Ambiental.* Madrid : s.n., 2008-2009.
26. **Muñoz, DY Sánchez.** . *Modelo Jerárquico de Evaluación de Impacto Ambiental empleando técnicas difusas.* Granada : s.n., 2008.
27. **Ferrer, Yiezenia Rosario.** *Evaluación en el tiempo del Impacto Ambiental con Técnicas Difusas. Aplicación en la Minería de Moa.* Granada : s.n., 2009.
28. **Erljman Piwen, Ariel y Goyén Fros, Alejandro.** *Problemas y soluciones en la Implementación de Extreme Programming.* Universidad Católica de Uruguay : s.n., abril, 2011.
29. **Pressman, Roger.** *Ingeniería de Software, Un Enfoque Práctico.* [En línea] 10 de diciembre de 2011. <http://es.scribd.com/doc/7978336/Ingenieria-de-Software-Un-Enfoque-Practico-Pressman-5th-Ed..>
30. **Fernández., Oscar Belmonte.** *Introducción al lenguaje de programación Java.* . [En línea] 2005. <http://www3.uji.es/~belfern/pdidoc/IX26/Documentos/introJava.pdf..>
31. **Castro, Luis.** *¿Qué es HTML?* [En línea] <http://aprenderinternet.about.com/od/Glosario/g/Que-Es-Html.htm>.
32. *Introducción - Tutorial CSS .* [En línea] <http://es.html.net/tutorials/css/introduction.php>.
33. **Gutiérrez, Javier J.** *Que es un framework web.* [En línea] http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf.
34. *¿Qué es y Para qué Sirve jQuery?* [En línea] <http://programacionya.com/%C2%BFque-es-y-para-que-sirve-jquery/>.
35. *Diseñando tu nuevo proyecto web con Bootstrap 2.0.* [En línea] 23 de febrero de 2012. <http://www.genbetadev.com/desarrollo-web/disenando-tu-nuevo-proyecto-web-con-bootstrap-2-0>.
36. *Spring Framework.* [En línea] <http://www.springsource.org/spring-framework>.
37. *Hibernate.* [En línea] <http://www.hibernate.org/>.
38. **Martínez, Jairo Chapela.** *Introducción al entorno de desarrollo Eclipse.* [En línea] 27 de septiembre de 2007. http://www.gris.det.uvigo.es/wiki/pub/Main/MiscResources/Manual_Eclipse.pdf.
39. **Martínez, Rafael.** *Sobre PostgreSQL.* [En línea] 02 de Octubre de 2010. http://www.postgresql.org.es/sobre_postgresql.
40. **Lomeli, Walter Ivan Reyes.** *Optimización de la Web del I.T.M, Aplicación Servidor Web.* [En línea] 12 de Diciembre de 2011. <http://es.scribd.com/doc/52208534/29/CARACTERISTICAS-Y-VENTAJAS-DEL-APACHE..>
41. *Información general sobre aplicaciones de datos con n capas.* [En línea] <http://msdn.microsoft.com/es-es/library/bb384398.aspx>.
42. **Larman, Craig.** *UML y Patrones. Introducción al análisis y diseño orientado a objetos.* Mexico : Pearson Education, 1999.
43. **Pressman, Roger S.** *Ingeniería de Software, un enfoque práctico. Quinta edición.* [En línea] [Citado el: 10 de diciembre de 2011.] Disponible en: <http://es.scribd.com/doc/7978336/Ingenieria-de-Software-Un-Enfoque-Practico-Pressman-5th-Ed>.
44. **Delgado, M, Verdegay, J L y Vila, M A.** *A Model for Linguistic Partial Information in Decision Making Problems, Int. J. of Intelligent Systems.* 1994.
45. **Ida, M.** *Possibility Valuation based on Ordinal Utility, Japanese Journal of Fuzzy Theory and Systems.* 1995.
46. **F, L Sugianto y Kacprzyk, J.** *Linguistic Expressions in Decision Making with Uncertain Data.* 1998.
47. **J, J Buckley.** *The Multiple Judge, Multiple Criteria Ranking Problem: A Fuzzy Set Approach, Fuzzy Sets and Systems .* 1984.
48. **R, R Yager.** *Non-Numeric Multi-Criteria Multi-Person Decision Making, Group Decision and Negotiation .* 1993.

Anexos

Anexo I: Importancia de Impacto

Tabla 10: Importancia del Impacto (1)

<p style="text-align: center;">Naturaleza</p> <ul style="list-style-type: none"> • Impacto beneficioso + • Impacto perjudicial - 	<p style="text-align: center;">Intensidad(I) (grado de destrucción)</p> <ul style="list-style-type: none"> • Baja 1 • Media 2 • Alta 4 • Muy alta 8 • Total 12
<p style="text-align: center;">Extensión(EX) (área de influencia)</p> <ul style="list-style-type: none"> • Puntual 1 • Parcial 2 • Extenso 4 • Total 8 • Crítica (+4) 	<p style="text-align: center;">Momento(MO) (plazo de manifestación)</p> <ul style="list-style-type: none"> • Largo plazo 1 • Medio plazo 2 • Inmediato 4 • Crítico (+4)
<p style="text-align: center;">Persistencia(PE) (permanencia del efecto)</p> <ul style="list-style-type: none"> • Fugaz 1 • Temporal 2 • Permanente 4 	<p style="text-align: center;">Reversibilidad(RV)</p> <ul style="list-style-type: none"> • Corto plazo 1 • Medio plazo 2 • Irreversible 4
<p style="text-align: center;">Sinergia(SI) (regularidad de la manifestación)</p> <ul style="list-style-type: none"> • Sin sinergismo(simple) 1 • Sinérgico 2 • Muy sinérgico 4 	<p style="text-align: center;">Acumulación(AC) (Incremento progresivo)</p> <ul style="list-style-type: none"> • Simple 1 • Acumulativo 4

Efecto(EF) (relación causa-efecto)	Periodicidad(PR) (regularidad de la manifestación)
<ul style="list-style-type: none"> • Indirecto(secundario) 1 • Directo 4 	<ul style="list-style-type: none"> • Irregular o aperiódico o discontinuo 1 • Periódico 2 • Continuo 4
Recuperabilidad(MC) (reconstrucción por medios humanos)	Importancia(I)
<ul style="list-style-type: none"> • Recuperable de manera inmediata 1 • Recuperable a medio plazo 2 • Mitigable 4 • Irrecuperable 8 	$I = \frac{1}{3}[3I + 2EX + MO + PE + RV + SI + AC + EF + PR + MC]$

Anexo II: Ejemplos de aplicaciones del Modelado Lingüístico de Preferencias

Tabla 11 Ejemplos de aplicaciones del Modelado Lingüístico de Preferencias

Aplicaciones	Referencias	Autores	Título
Toma de Decisión	(11)	M. Delgado, J.L. Verdegay, M.A. Vila	Linguistic Decision-Making Models.
	(17)	M. Delgado, J.L. Verdegay, M.A. Vila	A Model for Linguistic Partial Information in Decision-Making Problems.
	(18)	M. Ida	Possibility Valuation based on Ordinal Utility.
	(19)	L.F. Sugianto, J. Kacprzyk	Linguistic Expression in Decision Making with Uncertain Data.
Decisión Multicriterio y Multiexperto (DMM)	(20)	J.J. Buckley	The Multiple Judge, Multiple Criteria Ranking Problem: A Fuzzy Set Approach.
	(21)	R.R. Yager	Non-Numeric Multicriteria Multi-Person Decision Making.

Anexo III: Historias de Usuario

Tabla 12: HU Nro. 1: Gestionar Usuarios

Historia de Usuario	
Número: 1	Nombre de Historia de Usuario: Gestionar Usuarios
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Lien Noa Jiménez Katia Ramírez Bruzón	Iteración Asignada: 1
Prioridad en negocio: Alto	Puntos estimados: 2
Riesgo en Desarrollo: Bajo	Puntos Reales: 2
Descripción: El Administrador del sistema insertará los datos de los usuarios que realizarán las EIA (nombre, especialidad, entidad, usuario, contraseña y rol). El Administrador podrá también modificar o eliminar dichos usuarios en caso de ser necesario.	
Observaciones: El Administrador del sistema debe estar previamente autenticado.	
Prototipo de interface:	

Tabla 13: HU Nro. 2: Crear proyecto para realizar la EIA

Historia de Usuario	
Número: 2	Nombre de Historia de Usuario: Crear proyecto para realizar la EIA
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Lien Noa Jiménez Katia Ramírez Bruzón	Iteración Asignada: 1
Prioridad en negocio: Alto	Puntos estimados: 1
Riesgo en Desarrollo: Alto	Puntos Reales: 1

Descripción: El Administrador del sistema creará los proyectos a los cuales se le realizará la EIA, para ello llenará los siguientes campos: proyecto a evaluar, entidad a la que pertenece, expertos que participan, factores, acciones, relación factor-acción y fase del proyecto.

Observaciones: El Administrador del sistema debe estar previamente autenticado.

Prototipo de interface:

Tabla 14: HU Nro. 3: Gestionar factores

Historia de Usuario	
Número: 3	Nombre de Historia de Usuario: Gestionar factores
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Lien Noa Jiménez	Iteración Asignada: 1


Katia Ramírez Bruzón	
Prioridad en negocio: Alto	Puntos estimados: 1
Riesgo en Desarrollo: Alto	Puntos Reales: 1
Descripción: El Administrador añadirá los factores medio ambientales que estarán relacionados con los proyecto. El Administrador podrá eliminar o editar dichos factores.	
Observaciones: El Administrador del sistema debe estar previamente autenticado.	
Prototipo de interface:	
	

Tabla 15: HU Nro. 4: Gestionar acciones

Historia de Usuario	
Número: 4	Nombre de Historia de Usuario: Gestionar acciones
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Lien Noa Jiménez Katia Ramírez Bruzón	Iteración Asignada: 1


Prioridad en negocio: Alta	Puntos estimados: 1
Riesgo en Desarrollo: Alto	Puntos Reales: 1
Descripción: El Administrador añadirá las acciones impactantes que estarán relacionadas con los proyecto. El Administrador podrá editar o eliminar dichas acciones.	
Observaciones: El Administrador del sistema debe estar previamente autenticado.	
Prototipo de interface:	
	

Tabla 16: HU Nro. 5: Relacionar factores-acciones

Historia de Usuario	
Número: 5	Nombre de Historia de Usuario: Relacionar factores-acciones
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Lien Noa Jiménez Katia Ramírez Bruzón	Iteración Asignada: 1
Prioridad en negocio: Alta	Puntos estimados: 1
Riesgo en Desarrollo: Alto	Puntos Reales: 1

Descripción: El sistema muestra una matriz para que el Administrador determine la relación que existe entre los factores y las acciones.

Observaciones: El Administrador del sistema debe estar previamente autenticado.

Prototipo de interface:

Introduzca la relación Factor-Acción :

Acciones / Factores	factor a	factor b
accion a
accion b
hgjrrr

Atrás Crear

Tabla 17: HU Nro. 6: Mostrar estado de los proyectos

Historia de Usuario	
Número: 6	Nombre de Historia de Usuario: Mostrar estado de los proyectos
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Lien Noa Jiménez Katia Ramírez Bruzón	Iteración Asignada: 2
Prioridad en negocio: Medio	Puntos estimados: 1

Riesgo en Desarrollo: Medio	Puntos Reales: 1																												
Descripción: El sistema muestra al Administrador el estado de cada proyecto.																													
Observaciones: El Administrador del sistema debe estar previamente autenticado.																													
Prototipo de interface:																													
<ul style="list-style-type: none"> 👤 Gestionar Usuarios ▾ 🕒 Gestionar Factores ▾ 🔥 Gestionar Acciones ▾ 📁 Gestionar Proyectos ▾ 🍃 Gestionar Conjuntos ▾ ⚙️ Administrar ▾ 	<table border="1"> <thead> <tr> <th>Proyectos</th> <th>Estado</th> </tr> </thead> <tbody> <tr><td>quizas</td><td>0%</td></tr> <tr><td>c</td><td>0%</td></tr> <tr><td>guijkoiopokpkop</td><td>0%</td></tr> <tr><td>asdas</td><td>0%</td></tr> <tr><td>sadadasd</td><td>0%</td></tr> <tr><td>klfklñlkñ</td><td>0%</td></tr> <tr><td>wqeqe</td><td>0%</td></tr> <tr><td>uuuuuuulll</td><td>0%</td></tr> <tr><td>pruebaEditarSeis</td><td>0%</td></tr> <tr><td>pruebaEditarSiete</td><td>0%</td></tr> <tr><td>pruebaEditarOcho</td><td>0%</td></tr> <tr><td>dgf</td><td>0%</td></tr> <tr><td>sdfsdf</td><td>0%</td></tr> </tbody> </table>	Proyectos	Estado	quizas	0%	c	0%	guijkoiopokpkop	0%	asdas	0%	sadadasd	0%	klfklñlkñ	0%	wqeqe	0%	uuuuuuulll	0%	pruebaEditarSeis	0%	pruebaEditarSiete	0%	pruebaEditarOcho	0%	dgf	0%	sdfsdf	0%
Proyectos	Estado																												
quizas	0%																												
c	0%																												
guijkoiopokpkop	0%																												
asdas	0%																												
sadadasd	0%																												
klfklñlkñ	0%																												
wqeqe	0%																												
uuuuuuulll	0%																												
pruebaEditarSeis	0%																												
pruebaEditarSiete	0%																												
pruebaEditarOcho	0%																												
dgf	0%																												
sdfsdf	0%																												

Tabla 18: HU Nro. 7: Archivar proyectos terminados

Historia de Usuario	
Número: 7	Nombre de Historia de Usuario: Archivar proyectos terminados
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Lien Noa Jiménez Katia Ramírez Bruzón	Iteración Asignada: 2

Prioridad en negocio: Media	Puntos estimados: 1
Riesgo en Desarrollo: Medio	Puntos Reales: 1
Descripción: El Administrador archivará en la base de datos aquellos proyectos que ya se le realizó la evaluación.	
Observaciones: El Administrador del sistema debe estar previamente autenticado.	
Prototipo de interface:	

Tabla 19: HU Nro. 8: Mostrar proyectos archivados.

Historia de Usuario	
Número: 8	Nombre de Historia de Usuario: Mostrar proyectos archivados.
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Lien Noa Jiménez	Iteración Asignada: 2

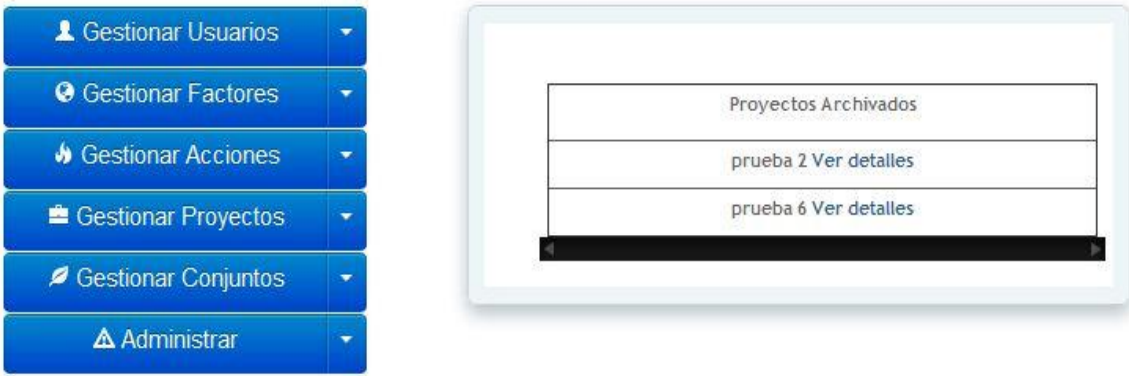
Katia Ramírez Bruzón	
Prioridad en negocio: Baja	Puntos estimados: 1
Riesgo en Desarrollo: Bajo	Puntos Reales: 1
Descripción: El Administrador podrá ver los proyectos que ya fueron archivados.	
Observaciones: El Administrador del sistema debe estar previamente autenticado.	
Prototipo de interface:	
	

Tabla 20: HU Nro. 9: Gestionar Conjuntos

Historia de Usuario	
Número: 9	Nombre de Historia de Usuario: Gestionar Conjuntos
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Lien Noa Jiménez Katia Ramírez Bruzón	Iteración Asignada: 2
Prioridad en negocio: alto	Puntos estimados: 1
Riesgo en Desarrollo: Medio	Puntos Reales: 1

Descripción: El Administrador podrá crear los conjuntos de cinco o tres etiquetas, además de asignar un conjunto a un símbolo.

Observaciones: El Administrador del sistema debe estar previamente autenticado.

Prototipo de interface:



Tabla 21: HU Nro. 10: Mostrar proyectos asignados

Historia de Usuario	
Número: 10	Nombre de Historia de Usuario: Mostrar proyectos asignados
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Lien Noa Jiménez Katia Ramírez Bruzón	Iteración Asignada: 3
Prioridad en negocio: Alto	Puntos estimados: 1

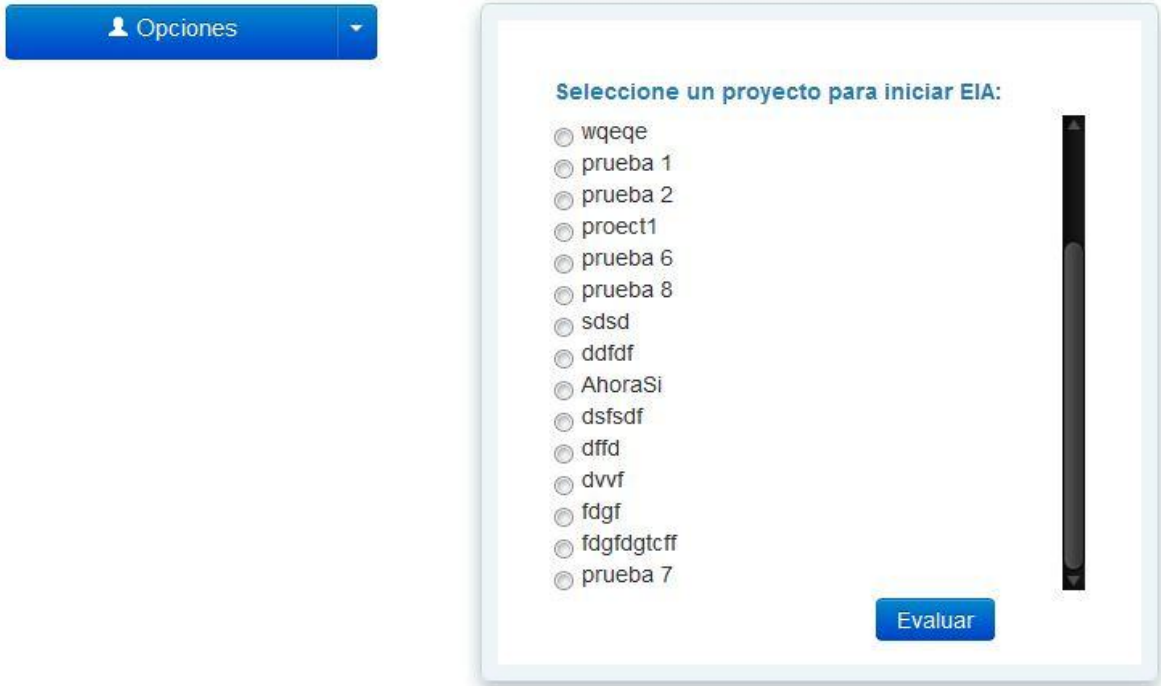
Riesgo en Desarrollo: Alto	Puntos Reales: 1
Descripción: El usuario Experto podrá ver los proyectos que el Administrador le asignó.	
Observaciones: El Experto del sistema debe estar previamente autenticado.	
Prototipo de interface:	
	

Tabla 22: HU Nro. 12: Exportar a Excel

Historia de Usuario	
Número: 12	Nombre de Historia de Usuario: Exportar a Excel
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Lien Noa Jiménez Katia Ramírez Bruzón	Iteración Asignada: 3

Prioridad en negocio: Baja	Puntos estimados: 1																														
Riesgo en Desarrollo: Bajo	Puntos Reales: 1																														
Descripción: El sistema permitirá al Administrador exportar a un documento Excel la información final de los proyectos a los que se le haya realizado la evaluación.																															
Observaciones: El Administrador del sistema debe estar previamente autenticado.																															
Prototipo de interface:																															
<p>Matriz de importancia :</p> <table border="1"> <thead> <tr> <th>Factores /Acciones</th> <th>factor a</th> <th>factor b</th> <th>Imp. Absoluta</th> <th>Imp. Relativa</th> </tr> </thead> <tbody> <tr> <td>accion a</td> <td>(b,0.2)</td> <td>- (b,0.42)</td> <td>(T,-0.11)</td> <td>(T,-0.49)</td> </tr> <tr> <td>accion b</td> <td>- (m,0.0)</td> <td>Sin relación</td> <td>(MP,0.0)</td> <td>(MP,0.0)</td> </tr> <tr> <td>UIP</td> <td>0.031</td> <td>0.056</td> <td></td> <td></td> </tr> <tr> <td>Imp. Absoluta</td> <td>(T,-0.4)</td> <td>(IP,-0.42)</td> <td>(IP,-0.05)</td> <td></td> </tr> <tr> <td>Imp. Relativa</td> <td>(T,-0.4)</td> <td>(IP,-0.42)</td> <td></td> <td>(IP,-0.25)</td> </tr> </tbody> </table> <p style="text-align: right;"> <input type="button" value="Archivar Proyecto"/> <input type="button" value="Exportar a Excel"/> </p>		Factores /Acciones	factor a	factor b	Imp. Absoluta	Imp. Relativa	accion a	(b,0.2)	- (b,0.42)	(T,-0.11)	(T,-0.49)	accion b	- (m,0.0)	Sin relación	(MP,0.0)	(MP,0.0)	UIP	0.031	0.056			Imp. Absoluta	(T,-0.4)	(IP,-0.42)	(IP,-0.05)		Imp. Relativa	(T,-0.4)	(IP,-0.42)		(IP,-0.25)
Factores /Acciones	factor a	factor b	Imp. Absoluta	Imp. Relativa																											
accion a	(b,0.2)	- (b,0.42)	(T,-0.11)	(T,-0.49)																											
accion b	- (m,0.0)	Sin relación	(MP,0.0)	(MP,0.0)																											
UIP	0.031	0.056																													
Imp. Absoluta	(T,-0.4)	(IP,-0.42)	(IP,-0.05)																												
Imp. Relativa	(T,-0.4)	(IP,-0.42)		(IP,-0.25)																											

Anexo IV: Ejemplos del empleo de los patrones de diseño en la aplicación

Patrones de diseño del Framework Spring

```
package cu.uci.sevia.managment.web.especialistas;  
  
import java.util.List;  
  
@Controller  
public class MatrizProyectoController {  
  
    @Autowired  
    private IProyectoManager proyectoManager;  
  
    public IProyectoManager getProyectoManager() {  
        return proyectoManager;  
    }  
}
```

Ilustración 24: Representación de la anotación @Autowire

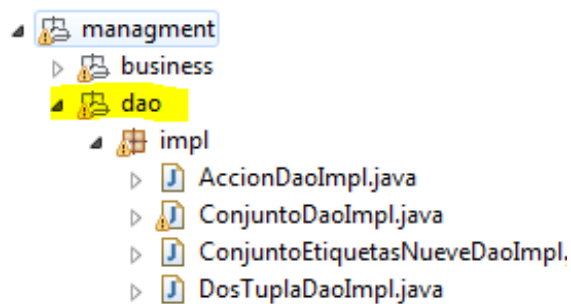


Ilustración 25: Representación del patrón DAO

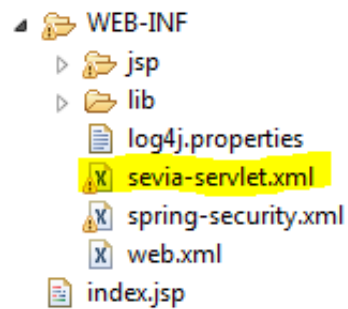


Ilustración 26: Representación del fichero sevia-servlet.xml

Patrones para Asignar Responsabilidades (GRASP)

```

package cu.uci.sevia.managment.business.impl;

import java.util.ArrayList;

@Service
public class ProyectoManagerImpl implements IProyectoManager {

    @Autowired
    private IProyectoDao proyectoDao;

    public IProyectoDao getProyectoDao() {
        return proyectoDao;
    }
}
    
```

Ilustración 27: Representación del patrón Experto en la clase ProyectoManagerImpl

```

public class IntervaloManagerImpl implements IIntervaloManager{

    @Autowired
    private IIntervaloDao intervaloDao;

    public IIntervaloDao getIntervaloDao() {
        return intervaloDao;
    }
}
    
```

Ilustración 28: Representación del patrón Creador en la clase IntervaloManagerImp

```

package cu.uci.sevia.managment.web.especialistas;

import org.springframework.beans.factory.annotation.Autowired;

@Controller
public class SalvarMatrizProyecto {

    @Autowired
    private IProyectoManager proyectoManager;
}
    
```

Ilustración 29: Representación del patrón Controlador en la clase SalvarMatrizProyectoController

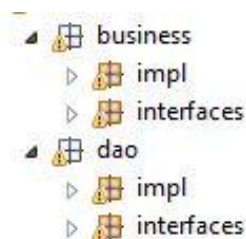


Ilustración 30: Representación del patrón Bajo Acoplamiento en los paquetes business y dao

Anexo V: Tarjetas Clase-responsabilidad-Colaborador

Tabla 23: Clase Factor

Clase FactorManagerImpl

Responsabilidades	Colaboradores
Crea los factores del proyecto.	IFactorManager
Modifica los factores del proyecto.	Factor
Edita los factores del proyecto.	CrearFactorController
Elimina los factores del proyecto.	EditarFactorController
	EliminarFactorController

Tabla 24: Clase Acción

Clase AccionManagerImpl	
Responsabilidades	Colaboradores
Crea las acciones del proyecto.	IAccionManager
Modifica las acciones del proyecto.	Accion
Edita las acciones del proyecto.	CrearAccionController
Elimina las acciones del proyecto.	EditarAccionController
	EliminarAccionController

Tabla 25: Clase Símbolo Etiqueta

Clase SimboloEtiquetaManagerImpl	
Responsabilidades	Colaboradores
Adiciona etiquetas a los símbolos.	SimboloEtiqueta
Edita las etiquetas de los símbolos.	ISimboloEtiquetaManager
Elimina etiquetas de los símbolos.	MatrizImportanciaController
	GuardarSimboloController

--	--

Tabla 26: Clase Dos Tuplas

Clase DosTuplasManagerImpl	
Responsabilidades	Colaboradores
Adiciona 2-tuplas.	IDosTuplasManager
Implementa el operador media aritmética.	Proyecto
Implementa el operador media ponderada.	MatrizImportanciaController
Invierte conjuntos.	
Realiza función delta.	
Realiza función delta Inversa.	

Anexo VI: Descripción de las tablas del modelo de datos del sistema

Tabla 27: Proyecto

Nombre: proyecto		
Tipo: Entidad		
Descripción: Almacena los datos de los proyectos.		
Atributo	Tipo	Descripción
idproject	bigint	Identificador de la tabla.
entidad	varchar	Entidad a la que pertenece el proyecto.
estado	varchar	Estado en el que está el proyecto.
fase	varchar	Fase en la que se encuentra el proyecto.
nombre	varchar	Nombre del proyecto.

Tabla 28: Usuario

Nombre: usuario
Tipo: Entidad

Descripción: Almacena los datos de los usuarios.		
Atributo	Tipo	Descripción
idusuario	bigint	Identificador de la tabla.
entidad	varchar	Entidad a la que pertenece el usuario.
especialidad	varchar	Especialidad que tiene el usuario.
nombreespecialista	varchar	Nombre del usuario.
password	varchar	Contraseña del usuario.
usuario	varchar	Usuario.
idrole	bigint	Identificador de la tabla Role.

Tabla 29: Etiqueta

Nombre: etiqueta		
Tipo: Entidad		
Descripción: Almacena los datos de las etiquetas.		
Atributo	Tipo	Descripción
idetiqueta	bigint	Identificador de la tabla.
a	double	Número difuso triangular de la etiqueta.
b	double	Número difuso triangular de la etiqueta.
c	double	Número difuso triangular de la etiqueta.
indice	bigint	Índice de la etiqueta.
nombrecorto	varchar	Abreviatura de la etiqueta.
nombreetiqueta	varchar	Nombre de la etiqueta.
idconjunto	bigint	Identificador de la tabla conjunto.

Tabla 30: Símbolo

Nombre: simbolo		
Tipo: Entidad		
Descripción: Almacena los datos de los símbolos.		
Atributo	Tipo	Descripción
idsimbolo	bigint	Identificador de la tabla.

abreviatura	varchar	Abreviatura del símbolo.
dominio	varchar	Dominio del símbolo.
nombresimbolo	varchar	Nombre del símbolo.
peso	double	Peso del símbolo.
idconjunto	bigint	Identificador de la tabla conjunto.
valormaximo	bigint	Valor máximo del símbolo.

Tabla 31: Dos tuplas

Nombre: dostuplas		
Tipo: Entidad		
Descripción: Almacena los datos de las 2-tuplas.		
Atributo	Tipo	Descripción
iddostuplas	bigint	Identificador de la tabla.
alfa	double	Traslación simbólica.
indiceetiqueta	bigint	Índice de la etiqueta.

Tabla 32. Factor

Nombre: factor		
Tipo: Entidad		
Descripción: Almacena los datos de los factores.		
Atributo	Tipo	Descripción
idfactor	bigint	Identificador de la tabla.
nombrefactor	varchar	Nombre del factor.
uip	double	Unidad de Importancia del factor.

Tabla 33. Factor proyecto

Nombre: factor_proyecto		
Tipo: Entidad		
Descripción: Relación entre la tabla factor y la tabla proyecto.		
Atributo	Tipo	Descripción
proyecto	bigint	Identificador de la tabla proyecto.

factor	bigint	Identificador de la tabla factor.
--------	--------	-----------------------------------

Tabla 34. Acción

Nombre: accion		
Tipo: Entidad		
Descripción: Almacena los datos de las acciones.		
Atributo	Tipo	Descripción
idaccion	bigint	Identificador de la tabla.
nombreaccion	varchar	Nombre de la acción.

Tabla 35. Acción proyecto

Nombre: accion_proyecto		
Tipo: Entidad		
Descripción: Relación entre la tabla accion y la tabla proyecto.		
Atributo	Tipo	Descripción
proyecto	bigint	Identificador de la tabla proyecto.
accion	bigint	Identificador de la tabla accion.

Tabla 36. Factor acción

Nombre: factoraccion		
Tipo: Entidad		
Descripción: Relación entre la tabla factor y la tabla accion.		
Atributo	Tipo	Descripción
idfactoraccion	bigint	Identificador de la tabla.
idaccion	bigint	Identificador de la tabla accion
idfactor	bigint	Identificador de la tabla factor.
idproject	bigint	Identificador de la tabla proyecto.
relacion	varchar	Relación que existe entre los factores y las acciones.

Tabla 37. Usuario proyecto

Nombre: usuario_proyecto		
---------------------------------	--	--

Tipo: Entidad		
Descripción: Relación entre la tabla usuario y la tabla proyecto.		
Atributo	Tipo	Descripción
proyecto	bigint	Identificador de la tabla proyecto.
usuario	varchar	Identificador de la tabla usuario.

Tabla 38.role

Nombre: role		
Tipo: Entidad		
Descripción: Almacena los datos de los roles		
Atributo	Tipo	Descripción
idrole	bigint	Identificador de la tabla role.
namerole	varchar	Nombre del rol.

Tabla 39. Role fuctionality

Nombre: role_fuctionality		
Tipo: Entidad		
Descripción: Relación entre la tabla role y la tabla fuctionality.		
Atributo	Tipo	Descripción
role	bigint	Identificador de la tabla role.
fuctionality	bigint	Identificador de la tabla fuctionality.

Tabla 40. Fuctionality

Nombre: fuctionality		
Tipo: Entidad		
Descripción: Almacena los datos de las funcionalidades.		
Atributo	Tipo	Descripción
idfuctionality	bigint	Identificador de la tabla fuctionality.
name	varchar	Nombre de la funcionalidad.

Tabla 41. Símbolo etiqueta

Nombre: simboloetiqueta		
Tipo: Entidad		
Descripción: Relación entre la tabla simbolo y la tabla etiqueta.		
Atributo	Tipo	Descripción
idsimboloetiqueta	bigint	Identificador de la tabla.
idetiqueta	bigint	Identificador de la tabla etiqueta.
idfactoraccion	bigint	Identificador de la tabla factoraccion.
idproject	bigint	Identificador de la tabla proyecto.
idsimbolo	bigint	Identificador de la tabla simbolo.
idusuario	bigint	Identificador de la tabla conjunto.
iddostuplas	bigint	Identificador de la tabla 2-tuplas.

Tabla 42. Símbolo intervalo

Nombre: simbolo_intervalo		
Tipo: Entidad		
Descripción: Relación entre la tabla simbolo y la tabla intervalo.		
Atributo	Tipo	Descripción
intervalo	bigint	Identificador de la tabla intervalo.
simbolo	bigint	Identificador de la tabla simbolo.

Tabla 43. Intervalo

Nombre: intervalo		
Tipo: Entidad		
Descripción: Almacena los datos de los intervalo		
Atributo	Tipo	Descripción
intervalo	bigint	Identificador de la tabla.
nombreintervalo	varchar	Nombre del intervalo.
valorintervalo	bigint	Valor del intervalo.

Tabla 44. Etiqueta símbolos numéricos

Nombre: etiquetasimbolosnumericos		
Tipo: Entidad		
Descripción: Almacena los datos de la tabla etiquetasimbolosnumericos.		
Atributo	Tipo	Descripción
idsimbolonumerico	bigint	Identificador de la tabla.
nombreako	varchar	Nombre del intervalo.
numerocritico	bigint	Número crítico.
nombresimbolo	varchar	Nombre del símbolo.
idfactoracion	bigint	Identificador de la tabla factoracion.
idproyecto	bigint	Identificador de la tabla proyecto.
idusuario	bigint	Identificador de la tabla usuario.

Tabla 45. Conjunto etiquetas nuevo

Nombre: conjuntoetiquetasnuevo		
Tipo: Entidad		
Descripción: Almacena los datos de la tabla conjuntoetiquetasnuevo.		
Atributo	Tipo	Descripción
idetiqueta	bigint	Identificador de la tabla.
indice	bigint	Índice de la etiqueta.
nombreako	varchar	Abreviatura de la etiqueta.
nombreetiqueta	varchar	Nombre de la etiqueta.

Tabla 46. Importancia

Nombre: importancia		
Tipo: Entidad		
Descripción: Almacena los datos de la importancia.		
Atributo	Tipo	Descripción
idimportancia	bigint	Identificador de la tabla.
idconjunto	bigint	Identificador de la tabla conjunto.
nombreako	varchar	Nombre de la importancia.

Tabla 47. Conjunto

Nombre: conjunto		
Tipo: Entidad		
Descripción: Almacena los datos de los conjuntos.		
Atributo	Tipo	Descripción
idconjunto	bigint	Identificador de la tabla.
nombreconjunto	varchar	Nombre del conjunto.

Anexo VII: Tareas de la ingeniería

Tabla 48: Tarea Nro. 1 Adicionar Usuario

Tarea de Ingeniería	
Número Tarea: 1	Número Historia de Usuario: 1
Nombre Tarea: Adicionar Usuario	
Tipo de Tarea: Desarrollo	Puntos Estimados: 10
Programador Responsable: Lien Noa Jiménez Katia Ramírez Bruzón	
Descripción: El Administrador del sistema insertará los datos de los usuarios que realizarán las EIA (nombre, especialidad, entidad, usuario, contraseña y rol).	

Tabla 49: Tarea Nro. 2 Editar Usuario

Tarea de Ingeniería	
Número Tarea: 2	Número Historia de Usuario: 1
Nombre Tarea: Editar Usuario	
Tipo de Tarea: Desarrollo	Puntos Estimados: 10

Programador Responsable: Lien Noa Jiménez Katia Ramírez Bruzón
Descripción: El Administrador del sistema modificará los datos de los usuarios en caso de ser necesario.

Tabla 50: Tarea Nro. 3 Eliminar Usuario

Tarea de Ingeniería	
Número Tarea: 3	Número Historia de Usuario: 1
Nombre Tarea: Eliminar Usuario	
Tipo de Tarea: Desarrollo	Puntos Estimados: 10
Programador Responsable: Lien Noa Jiménez Katia Ramírez Bruzón	
Descripción: El Administrador del sistema eliminará los usuarios que ya no trabajen en el proyecto.	

Tabla 51: Tarea Nro. 4 Crear proyecto para realizar la EIA

Tarea de Ingeniería	
Número Tarea: 4	Número Historia de Usuario: 2
Nombre Tarea: Crear proyecto para realizar la EIA	
Tipo de Tarea: Desarrollo	Puntos Estimados: 5
Programador Responsable: Lien Noa Jiménez Katia Ramírez Bruzón	
Descripción: El Administrador del sistema creará los proyectos a los cuales se les realizará la EIA, para ello llenará los siguientes campos: proyecto a evaluar, entidad a la que pertenece,	

expertos que participan, factores, acciones, relación factor-acción y etapa del proyecto.

Tabla 52: Tarea Nro. 5 Añadir Factores

Tarea de Ingeniería	
Número Tarea: 5	Número Historia de Usuario: 3
Nombre Tarea: Añadir Factores	
Tipo de Tarea: Desarrollo	Puntos Estimados: 5
Programador Responsable: Lien Noa Jiménez Katia Ramírez Bruzón	
Descripción: El Administrador añadirá los factores medio ambientales que estarán relacionados con los proyecto.	

Tabla 53: Tarea Nro. 6 Editar Factores

Tarea de Ingeniería	
Número Tarea: 6	Número Historia de Usuario: 3
Nombre Tarea: Editar Factores	
Tipo de Tarea: Desarrollo	Puntos Estimados: 5
Programador Responsable: Lien Noa Jiménez Katia Ramírez Bruzón	
Descripción: El Administrador editará los factores medio ambientales que estarán relacionados con los proyecto.	

Tabla 54: Tarea Nro. 7 Eliminar Factores

Tarea de Ingeniería

Número Tarea: 7	Número Historia de Usuario: 3
Nombre Tarea: Eliminar Factores	
Tipo de Tarea: Desarrollo	Puntos Estimados: 5
Programador Responsable: Lien Noa Jiménez Katia Ramírez Bruzón	
Descripción: El Administrador eliminará los factores medio ambientales que ya no estarán relacionados con los proyecto.	

Tabla 55: Tarea Nro. 8 Añadir Acciones

Tarea de Ingeniería	
Número Tarea: 8	Número Historia de Usuario: 4
Nombre Tarea: Añadir Acciones	
Tipo de Tarea: Desarrollo	Puntos Estimados: 5
Programador Responsable: Lien Noa Jiménez Katia Ramírez Bruzón	
Descripción: El Administrador añadirá las acciones impactantes que estarán relacionadas con los proyecto.	

Tabla 56: Tarea Nro. 9 Editar Acciones

Tarea de Ingeniería	
Número Tarea: 9	Número Historia de Usuario: 4
Nombre Tarea: Editar Acciones	
Tipo de Tarea: Desarrollo	Puntos Estimados: 5
Programador Responsable: Lien Noa Jiménez	

Katia Ramírez Bruzón
Descripción: El Administrador editará las acciones impactantes que estarán relacionadas con los proyectos.

Tabla 57: Tarea Nro. 10 Eliminar Acciones

Tarea de Ingeniería	
Número Tarea: 10	Número Historia de Usuario: 4
Nombre Tarea: Eliminar Acciones	
Tipo de Tarea: Desarrollo	Puntos Estimados: 5
Programador Responsable: Lien Noa Jiménez Katia Ramírez Bruzón	
Descripción: El Administrador eliminará las acciones impactantes que ya no estarán relacionadas con los proyecto.	

Tabla 58: Tarea Nro. 11 Relacionar factores-acciones

Tarea de Ingeniería	
Número Tarea: 11	Número Historia de Usuario: 5
Nombre Tarea: Relacionar factores-acciones	
Tipo de Tarea: Desarrollo	Puntos Estimados: 5
Programador Responsable: Lien Noa Jiménez Katia Ramírez Bruzón	
Descripción: El sistema muestra una matriz para que el Administrador determine la relación que existe entre los factores y las acciones.	

Tabla 59: Tarea Nro. 12 Mostrar estado de los proyectos

Tarea de Ingeniería	
Número Tarea: 12	Número Historia de Usuario: 6
Nombre Tarea: Mostrar estado de los proyectos	
Tipo de Tarea: Desarrollo	Puntos Estimados: 5
Programador Responsable: Lien Noa Jiménez Katia Ramírez Bruzón	
Descripción: El sistema muestra al Administrador el estado de cada proyecto.	

Tabla 60: Tarea Nro. 13 Archivar proyectos terminados

Tarea de Ingeniería	
Número Tarea: 13	Número Historia de Usuario: 7
Nombre Tarea: Archivar proyectos terminados	
Tipo de Tarea: Desarrollo	Puntos Estimados: 5
Programador Responsable: Lien Noa Jiménez Katia Ramírez Bruzón	
Descripción: El Administrador archivará en la base de datos aquellos proyectos a los que ya se le realizó la evaluación.	

Tabla 61: Tarea Nro. 14 Mostrar proyectos archivados

Tarea de Ingeniería	
Número Tarea: 14	Número Historia de Usuario: 8
Nombre Tarea: Mostrar proyectos archivados	
Tipo de Tarea: Desarrollo	Puntos Estimados: 5

Programador Responsable: Lien Noa Jiménez Katia Ramírez Bruzón
Descripción: El Administrador podrá ver los proyectos que ya fueron archivados.

Tabla 62: Tarea Nro. 15 Crear etiquetas

Tarea de Ingeniería	
Número Tarea: 15	Número Historia de Usuario: 9
Nombre Tarea: Crear etiquetas	
Tipo de Tarea: Desarrollo	Puntos Estimados: 5
Programador Responsable: Lien Noa Jiménez Katia Ramírez Bruzón	
Descripción: El Administrador podrá crear los conjuntos de cinco o tres etiquetas, introduciendo el nombre de la etiqueta y su abreviatura.	

Tabla 63: Tarea Nro. 16 Asignar Conjuntos

Tarea de Ingeniería	
Número Tarea: 16	Número Historia de Usuario: 9
Nombre Tarea: Asignar Conjuntos	
Tipo de Tarea: Desarrollo	Puntos Estimados: 5
Programador Responsable: Lien Noa Jiménez Katia Ramírez Bruzón	
Descripción: El Administrador podrá asignar un conjunto a un símbolo.	

Tabla 64: Tarea Nro. 17 Mostrar proyectos asignados

Tarea de Ingeniería	
Número Tarea: 17	Número Historia de Usuario: 11
Nombre Tarea: Mostrar proyectos asignados	
Tipo de Tarea: Desarrollo	Puntos Estimados: 5
Programador Responsable: Lien Noa Jiménez Katia Ramírez Bruzón	
Descripción: El usuario Experto podrá ver los proyectos que el Administrador le asignó.	

Tabla 65: Tarea Nro. 19 Exportar a Excel

Tarea de Ingeniería	
Número Tarea: 19	Número Historia de Usuario: 10
Nombre Tarea: Exportar a Excel	
Tipo de Tarea: Desarrollo	Puntos Estimados: 5
Programador Responsable: Lien Noa Jiménez Katia Ramírez Bruzón	
Descripción: El sistema permitirá al Administrador exportar a un documento Excel la información final de los proyectos que ya se le haya realizado la evaluación.	

Anexo VIII: Pruebas de aceptación

Tabla 66: Prueba de aceptación 1: Adicionar Usuario

Clases Válidas	Clases Inválidas	Resultado Esperado	Resultado de la Prueba	Observaciones
El Administrador accede al sistema y selecciona Gestionar		El sistema inserta los datos en la base de datos y muestra una	Satisfactorio.	El Administrador debe estar

Usuarios, escoge la opción Adicionar e introduce los siguientes datos: nombre, especialidad, entidad, usuario, contraseña y rol.		notificación informando que el usuario se añadió correctamente.		previamente autenticado.
	El Administrador accede al sistema y selecciona Gestionar Usuarios, escoge la opción Adicionar y de los campos nombre, especialidad, entidad, usuario, contraseña y rol, deja alguno vacío.	El sistema muestra una notificación informando que se dejaron campos vacíos y ofrece la opción de volver a introducir los datos.	Satisfactorio.	El Administrador debe estar previamente autenticado.
	El Administrador accede al sistema y selecciona Gestionar Usuarios, escoge la opción Adicionar y en los campos nombre, especialidad, entidad, usuario, escribe números o caracteres.	El sistema no permite que el usuario introduzca números o caracteres.	Satisfactorio.	El Administrador debe estar previamente autenticado.

Tabla 67: Prueba de aceptación 4: Crear proyecto para realizar la EIA

Clases Válidas	Clases Inválidas	Resultado Esperado	Resultado de la Prueba	Observaciones
El Administrador		El sistema inserta los	Satisfactorio.	El

<p>accede al sistema y selecciona Gestionar Proyectos, escoge la opción Crear e introduce los siguientes datos: proyecto a evaluar, entidad a la que pertenece, cantidad de expertos que participan, factores, acciones, relación. factor-acción y fase del proyecto.</p>		<p>datos en la base de datos y muestra una notificación informando que el proyecto ya ha sido creado.</p>		<p>Administrador debe estar previamente autenticado.</p>
	<p>El Administrador accede al sistema y selecciona Gestionar Proyectos, escoge la opción Crear y deja vacío algunos de los siguientes campos: proyecto a evaluar, entidad a la que pertenece, cantidad de expertos que participan, factores, acciones, relación factor-acción y fase del proyecto.</p>	<p>El sistema muestra una notificación informando que se dejaron campos vacíos y ofrece la opción de volver a introducir los datos.</p>	<p>Satisfactorio.</p>	<p>El Administrador debe estar previamente autenticado.</p>

	El Administrador accede al sistema y selecciona Gestionar Proyectos, escoge la opción Crear y en los campos proyecto a evaluar, entidad y fase, escribe números o caracteres.	El sistema no permite que el usuario introduzca números o caracteres.	Satisfactorio.	El Administrador debe estar previamente autenticado.
--	---	---	----------------	--

Tabla 68: Prueba de aceptación #2: Editar Usuario

Clases Válidas	Clases Inválidas	Resultado Esperado	Resultado de la Prueba	Observaciones
El Administrador accede al sistema y selecciona Gestionar Usuarios, escoge la opción Editar, elige el usuario y cambia algunos de los siguientes datos: nombre, especialidad, entidad, usuario, contraseña y rol.		El sistema cambia los datos en la base de datos y muestra una notificación informando que el usuario se modificó correctamente.	Satisfactorio.	El Administrador debe estar previamente autenticado. Debe existir al menos un usuario en la base de datos.
	El Administrador accede al sistema y selecciona Gestionar Usuarios, escoge la opción Editar, elige el	El sistema muestra una notificación informando que se dejaron campos vacíos y ofrece la posibilidad de volver a	Satisfactorio.	El Administrador debe estar previamente autenticado.

	usuario y de los campos nombre, especialidad, entidad, usuario, contraseña y rol, deja alguno vacío.	introducir los datos.		Debe existir al menos un usuario en la base de datos.
	El Administrador accede al sistema y selecciona Gestionar Usuarios, escoge la opción Editar, elige el usuario y en los campos nombre, especialidad, entidad, usuario, escribe números o caracteres.	El sistema no permite que el usuario introduzca números o caracteres.	Satisfactorio.	El Administrador debe estar previamente autenticado. Debe existir al menos un usuario en la base de datos.

Tabla 69: Prueba de aceptación #3: Eliminar Usuario

Clases Válidas	Clases Inválidas	Resultado Esperado	Resultado de la Prueba	Observaciones
El Administrador accede al sistema y selecciona Gestionar Usuarios, escoge Eliminar, elige el usuario que desee y selecciona Eliminar usuario.		El sistema elimina al usuario de la base de datos y muestra una notificación informando que el usuario se eliminó correctamente.	Satisfactorio.	El Administrador debe estar previamente autenticado. Debe existir al menos un usuario en la base de datos. El usuario no

				puede estar en ningún proyecto activo.
--	--	--	--	--

Tabla 70: Prueba de aceptación #5: Añadir Factores

Clases Válidas	Clases Inválidas	Resultado Esperado	Resultado de la Prueba	Observaciones
El Administrador accede al sistema, selecciona Gestionar Factores, escoge la opción Adicionar y llena los campos nombre del factor y UIP.		El sistema inserta en la base de datos el factor con la UIP correspondiente.	Satisfactorio.	El Administrador debe estar previamente autenticado.
	El Administrador accede al sistema, selecciona Gestionar Factores, escoge la opción Adicionar y de los campos nombre del factor y UIP deja alguno vacío.	El sistema muestra una notificación informando que se dejaron campos vacíos y ofrece la posibilidad de volver a introducir los datos.	Satisfactorio.	El Administrador debe estar previamente autenticado.
	El Administrador accede al sistema, selecciona Gestionar Factores, escoge la opción Adicionar y en el campo nombre del factor escribe números o	El sistema no permite que el usuario introduzca números o caracteres.	Satisfactorio.	El Administrador debe estar previamente autenticado.

	caracteres.			
	El Administrador accede al sistema, selecciona la opción Gestionar Factores y en el campo UIP escribe letras o caracteres.	El sistema no permite que el usuario introduzca números o caracteres.	Satisfactorio.	El Administrador debe estar previamente autenticado.

Tabla 71: Prueba de aceptación #6: Editar Factores

Clases Válidas	Clases Inválidas	Resultado Esperado	Resultado de la Prueba	Observaciones
El Administrador accede al sistema, selecciona Gestionar Factores, escoge la opción Editar, elige el factor y modifica algunos de los campos nombre del factor o UIP.		El sistema cambia los datos en la base de datos y muestra una notificación indicando que la información fue modificada correctamente.	Satisfactorio.	El Administrador debe estar previamente autenticado. Debe existir al menos un factor en base de datos.
	El Administrador accede al sistema, selecciona Gestionar Factores, escoge la opción Editar, elige el factor y de los campos nombre del factor y UIP y deja alguno vacío.	El sistema muestra una notificación informando que se dejaron campos vacíos y ofrece la posibilidad de volver a introducir los datos.	Satisfactorio.	El Administrador debe estar previamente autenticado. Debe existir al menos un factor en base de datos.

	El Administrador accede al sistema, selecciona Gestionar Factores, escoge la opción Editar, elige el factor y en el campo nombre del factor escribe números o caracteres.	El sistema no permite que el usuario introduzca números o caracteres.	Satisfactorio.	El Administrador debe estar previamente autenticado. Debe existir al menos un factor en base de datos.
	El Administrador accede al sistema, selecciona Gestionar Factores, escoge la opción Editar, elige el factor y en el campo UIP escribe letras o caracteres.	El sistema no permite que el usuario introduzca números o caracteres.	Satisfactorio.	El Administrador debe estar previamente autenticado. Debe existir al menos un factor en base de datos.

Tabla 72: Prueba de aceptación #7: Eliminar Factores

Clases Válidas	Clases Inválidas	Resultado Esperado	Resultado de la Prueba	Observaciones
El Administrador accede al sistema y selecciona Gestionar Factores, escoge la opción Eliminar, elige el factor y selecciona Eliminar factor.		El sistema elimina el factor de la base de datos y muestra una notificación informando que el factor se eliminó correctamente.	Satisfactorio.	El Administrador debe estar previamente autenticado. Debe existir al menos un factor en

				base de datos.
--	--	--	--	----------------

Tabla 73: Prueba de aceptación #8: Añadir Acciones

Clases Válidas	Clases Inválidas	Resultado Esperado	Resultado de la Prueba	Observaciones
El Administrador accede al sistema, selecciona Gestionar Acciones, escoge la opción Adicionar y llena el campo nombre de la acción.		El sistema inserta en la base de datos la acción y muestra una notificación indicando que la información fue añadida correctamente.	Satisfactorio.	El Administrador debe estar previamente autenticado.
	El Administrador accede al sistema, selecciona Gestionar Acciones, escoge la opción Adicionar y deja vacío el campo nombre de la acción.	El sistema muestra una notificación informando que se dejaron campos vacíos y ofrece la posibilidad de volver a introducir los datos.	Satisfactorio.	El Administrador debe estar previamente autenticado.

Tabla 74: Prueba de aceptación #9: Editar Acciones

Clases Válidas	Clases Inválidas	Resultado Esperado	Resultado de la Prueba	Observaciones
El Administrador accede al sistema, selecciona Gestionar Acciones, escoge la opción Editar, elige la acción y modifica el campo nombre de la acción.		El sistema cambia los datos en la base de datos y muestra una notificación indicando que la información fue modificada correctamente.	Satisfactorio.	El Administrador debe estar previamente autenticado. Debe existir al menos una acción en la

				base de datos.
	El Administrador accede al sistema, selecciona Gestionar Acciones, escoge la opción Editar, elige la acción y deja vacío el campo nombre de la acción.	El sistema muestra una notificación informando que se dejaron campos vacíos y ofrece la posibilidad de volver a introducir los datos.	Satisfactorio.	El Administrador debe estar previamente autenticado. Debe existir al menos una acción en la base de datos.

Tabla 75: Prueba de aceptación #10: Eliminar Acciones

Clases Válidas	Clases Inválidas	Resultado Esperado	Resultado de la Prueba	Observaciones
El Administrador accede al sistema y selecciona Gestionar Acciones, escoge la opción Eliminar, elige la acción y luego selecciona Eliminar acción..		El sistema elimina la acción de la base de datos y muestra una notificación informando que la acción se eliminó correctamente.	Satisfactorio.	El Administrador debe estar previamente autenticado. Debe existir al menos una acción en la base de datos.

Tabla 76: Prueba de aceptación #11: Mostrar estado de los proyectos

Clases Válidas	Clases Inválidas	Resultado Esperado	Resultado de la Prueba	Observaciones
El Administrador accede al sistema,		El sistema muestra el estado en el que se	Satisfactorio.	El Administrador

selecciona Gestionar Proyectos y escoge la opción Pendientes.		encuentran los proyectos.		debe estar previamente autenticado.
---	--	---------------------------	--	-------------------------------------

Tabla 77: Prueba de aceptación #12: Archivar proyectos terminados

Clases Válidas	Clases Inválidas	Resultado Esperado	Resultado de la Prueba	Observaciones
El Administrador accede al sistema y selecciona Gestionar Proyectos, escoge la opción Pendientes y selecciona Evaluar, al proyecto, luego elige la opción Archivar proyecto.		El sistema guarda en la base de datos el proyecto seleccionado.	Satisfactorio.	El Administrador debe estar previamente autenticado. El proyecto debe estar en 100%.

Tabla 78: Prueba de aceptación #13: Mostrar proyectos archivados

Clases Válidas	Clases Inválidas	Resultado Esperado	Resultado de la Prueba	Observaciones
El Administrador accede al sistema, selecciona Gestionar Proyectos y escoge la opción Archivados.		El sistema muestra los proyectos que ya están archivados en la base de datos.	Satisfactorio.	El Administrador debe estar previamente autenticado.

Tabla 79: Prueba de aceptación #14: Conjuntos de cinco etiquetas

Clases Válidas	Clases Inválidas	Resultado Esperado	Resultado de la Prueba	Observaciones
El Administrador accede al sistema y		El sistema inserta en la base de datos el	Satisfactorio.	El Administrador

selecciona Gestionar Conjuntos, escoge la opción cinco etiquetas y llenas los campos con los nombres de las etiquetas y sus abreviaturas.		conjunto.		debe estar previamente autenticado.
	El Administrador accede al sistema y selecciona Gestionar Conjuntos, escoge la opción cinco etiquetas y deja algunos de los campos vacíos.	El sistema muestra una notificación informando que se dejaron campos vacíos y ofrece la opción de volver a introducir los datos.	Satisfactorio.	El Administrador debe estar previamente autenticado.

Tabla 80: Prueba de aceptación #15: Conjuntos de tres etiquetas

Clases Válidas	Clases Inválidas	Resultado Esperado	Resultado de la Prueba	Observaciones
El Administrador accede al sistema y selecciona Gestionar Conjuntos, escoge la opción tres etiquetas y llenas los campos con los nombres de las etiquetas y sus abreviaturas.		El sistema inserta en la base de datos el conjunto.	Satisfactorio.	El Administrador debe estar previamente autenticado.
	El Administrador accede al sistema y selecciona Gestionar Conjuntos, escoge la	El sistema muestra una notificación informando que se dejaron campos vacíos y ofrece la	Satisfactorio.	El Administrador debe estar previamente

	opción tres etiquetas y deja algunos de los campos vacíos.	opción de volver a introducir los datos.		autenticado.
--	--	--	--	--------------

Tabla 81: Prueba de aceptación #16: Asignar un conjunto a un símbolo

Clases Válidas	Clases Inválidas	Resultado Esperado	Resultado de la Prueba	Observaciones
El Administrador accede al sistema y selecciona Gestionar Conjuntos, escoge la opción Asignar y selecciona el conjunto y el símbolo al que se lo desea asignar.		El sistema inserta en la base de datos la selección realizada.	Satisfactorio.	El Administrador debe estar previamente autenticado.
	El Administrador accede al sistema y selecciona Gestionar Conjuntos, escoge la opción Asignar y no selecciona el conjunto ni el símbolo al que se lo desea asignar.	El sistema muestra una notificación informando que debe seleccionar un símbolo y un conjunto.	Satisfactorio.	El Administrador debe estar previamente autenticado.
	El Administrador accede al sistema y selecciona Gestionar Conjuntos, escoge la opción Asignar y selecciona el conjunto pero no el símbolo al que se lo	El sistema muestra una notificación informando que debe seleccionar un símbolo.	Satisfactorio.	El Administrador debe estar previamente autenticado.

	desea asignar.			
	El Administrador accede al sistema y selecciona Gestionar Conjuntos, escoge la opción Asignar y no selecciona el conjunto.	El sistema muestra una notificación informando que debe seleccionar un conjunto.	Satisfactorio.	El Administrador debe estar previamente autenticado.

Tabla 82: Prueba de aceptación #17: Valoración de importancia de impacto

Clases Válidas	Clases Inválidas	Resultado Esperado	Resultado de la Prueba	Observaciones
El Experto accede al sistema, selecciona Opciones, escoge Proyectos Asignados y selecciona el proyecto al cual le va a realizar la evaluación.		El sistema muestra la matriz correspondiente a este proyecto.	Satisfactorio.	El Administrador debe estar previamente autenticado.
	El Experto accede al sistema, selecciona Opciones, escoge Proyectos Asignados y no escoge ningún proyecto.	El sistema muestra una notificación informando que debe seleccionara un proyecto.	Satisfactorio.	El Administrador debe estar previamente autenticado.

Tabla 83: Prueba de aceptación #18: Exportar a Excel

Clases Válidas	Clases Inválidas	Resultado Esperado	Resultado de la Prueba	Observaciones
El Administrador accede al sistema y selecciona Gestionar		El sistema muestra un documento Excel con la matriz correspondiente.	Satisfactorio.	El Administrador debe estar

<p>Proyectos, escoge la opción Pendientes y selecciona Evaluar, al proyecto, luego elige la opción Exportar a Excel.</p>			<p>previamente autenticado. El proyecto debe estar terminado.</p>
--	--	--	---

Anexo IX: Aceptación del sistema por parte del cliente

UCI Universidad de las Ciencias Informáticas

Acta de aceptación

ACTA DE ACEPTACIÓN

Producto: SEVIA V 1.0.

Involucrados en el proceso:

- **Estudiantes:** Katia Ramírez Bruzón,
Lien Noa Jiménez
- **Tutores:** Ing. David Rodríguez Rodríguez
Ing. Yoanny Torres Rubio.

Observaciones del proceso:

Las no conformidades detectadas en el proceso de revisión fueron resueltas. Se comprobó la correcta implementación del nuevo Modelo Representación de Preferencias Homogéneas. Por tanto se acepta con fecha de 9 de junio del 2013 la aplicación SEVIA en su V 1.0.

Lista de productos que son aceptados y que deben ser entregados:

- Código fuente de la aplicación.
- Fichero sevía.war.

Entrega: _____ Recibe: _____

Nombre y Apellidos: Katia Ramírez Bruzón, Lien Noa Jiménez Nombre y Apellidos: David Rodríguez Rodríguez

Cargo: Estudiante Cargo: Jefe de Departamento

Firma: *[Firma]* Firma: *[Firma]*

Comentarios: Los productos aceptados deben ser liberados por Calidad Centro y entregados al cliente previa a la defensa.

UCI Universidad de las Ciencias Informáticas
Centro Telemática CTLM

1

Ilustración 31: Carta de Aceptación por parte del cliente