

INDICE

1. INTRODUCCION Y OBJETIVOS DEL PROYECTO.....	5
1.1. AMBITO	5
1.2. ALCANCE	6
1.3. OBJETIVOS.....	6
1.4. CONTENIDO DEL CD-ROM	6
2. INFORMATICA PARA DISCAPACITADOS.....	9
2.1. SISTEMAS DE COMUNICACIÓN	9
2.1.1. INTRODUCCION.....	9
2.1.2. CLASIFICACION.....	10
2.1.2.1 PICTURE COMMUNICATIONS SYMBOLS (SPC)	11
2.1.2.2 SISTEMA BLISS	12
2.1.2.3 SISTEMA BRAILLE	12
2.2. INFORMATICA Y DISCAPACITADOS	13
2.2.1 ENSEÑANZA ASISTIDA POR ORDENADOR.....	14
2.2.2. COMUNICACIÓN.....	14
3. INTRODUCCION A LA REUTILIZACION Y REQUISITOS	15
3.1. DESARROLLO DEL SOFTWARE.....	15
3.2 REUTILIZACION DEL SOFTWARE.....	16
3.3. REQUISITOS DEL SOFTWARE.....	16
3.4. REUTILIZACION DE REQUISITOS	17
3.5. EL ENTORNO DE REUTILIZACION	19
3.6. EVALUACION DE LA REUTILIZACION	23
3.6.1. ESTUDIO DE RECURSOS REUTILIZABLES	23
3.6.2. IMPACTO SOBRE LA CALIDAD, PRODUCTIVIDAD Y COSTE	24
4. ANALISIS DE REQUISITOS EN EL DOMINIO DE INFORMATICA PARA DISCAPACITADOS	27
4.1. INTRODUCCION	27
4.1.1. NFR FRAMEWORK.....	28
4.1.2. I* FRAMEWORK	29
4.2. OBJETIVOS.....	30
4.2.1. REQUISITOS FUNCIONALES	31
4.2.2. REQUISITOS NO FUNCIONALES.....	35

4.3. CASOS DE USO.....	36
4.3.1. DEFINICION DE ACTORES.....	36
4.3.2. DIAGRAMAS DE LOS CASOS DE USO	37
4.3.3. DESCRIPCION DE LOS CASOS DE USO	50
4.4. RELACION CASOS DE USO – OBJETIVOS	114
4.5. APLICACIÓN A UN CASO PRÁCTICO	116
4.5.1. INTRODUCCION.....	116
4.5.2. ENTORNO UTILIZADO	117
4.5.3. ESTUDIO DEL CASO PRÁCTICO.....	121
4.5.3.1. PROBLEMAS ENCONTRADOS	127
4.5.3.2. CONCLUSIONES.....	128
5. DESARROLLO RAPIDO DE UNA HERRAMIENTA: “REPERTORY GRID”	131
5.1. FUNDAMENTO TEORICO.....	131
5.2. ANALISIS.....	133
5.2.1. DIAGRAMA DE OBJETIVOS	134
5.2.2. DEFINICION DE ACTORES.....	134
5.2.3. DIAGRAMA DE CASOS DE USO.....	134
5.2.4. DESCRIPCION DE LOS CASOS DE USO	135
5.2.5. DIAGRAMA INICIAL DE CLASES	141
5.3. DISEÑO	142
5.3.1. CASOS DE USO.....	143
5.3.2. IDENTIFICACIÓN DE LAS CLASES	146
5.3.3. DESCRIPCION DE LAS CLASES	147
5.3.4. DIAGRAMAS DE SECUENCIA	151
5.3.5. DIAGRAMAS DE COLABORACION	154
5.4. IMPLEMENTACION	158
5.4.1. LENGUAJE DE PROGRAMACIÓN SELECCIONADO.....	158
5.4.2. SOFTWARE EMPLEADO.....	159
5.4.3. HARDWARE EMPLEADO	159
5.5. PRUEBAS.....	160
5.6. EJEMPLO DE USO DE LA APLICACION	164
5.6.1. DESCRIPCIÓN.....	164
5.6.2. PANTALLA PRINCIPAL	164
5.6.3. REJILLA	165

5.6.4. CREACIÓN DE ALPHA-PLANOS.....	166
5.6.5. ANÁLISIS DE LAS RELACIONES.....	168
5.6.6. RESULTADOS	169
5.6.7. REEMPLAZO	170
6. CONCLUSIONES Y FUTURAS ACTUACIONES	173
6.1. OBJETIVOS ALCANZADOS	173
6.2. FUTURAS ACUTACIONES	174
7. APENDICE	175
7.1. DESCRIPCIÓN DE LA APLICACIÓN	175
7.1.1. PANTALLA PRINCIPAL.....	175
7.1.2. REJILLA	176
7.1.3. FORMULARIO DE CREACIÓN DE ALPHA-PLANOS	177
7.1.4. FORMULARIO DE ANÁLISIS DE LAS RELACIONES	178
7.1.5. HOJA DE RESULTADOS	179
7.1.6. FORMULARIO DE REEMPLAZO.....	180
8. BIBLIOGRAFIA	183

1. INTRODUCCION Y OBJETIVOS DEL PROYECTO

1.1. AMBITO

Este proyecto forma parte de una serie de proyectos propuestos por el Departamento de Informática orientados al desarrollo de componentes software reutilizables en distintos dominios. El dominio que aquí se ha tratado ha sido “Informática para Discapacitados”.

Estos componentes software se desarrollan con la finalidad de elaborar un conjunto de elementos reutilizables que permitan el desarrollo de aplicaciones futuras a partir de la reutilización de dichos componentes, para ello son clasificados y almacenados en un repositorio al cual se podrá acceder posteriormente para su utilización.

En los últimos años, el dominio de Informática para discapacitados ha tenido un importante auge gracias a los avances técnicos de la informática, que permiten desarrollar aplicaciones para suplir cada vez mayor cantidad de discapacidades, facilitando así la comunicación de dichas personas con su entorno. Esto también se ha visto reflejado en el desarrollo de numerosos proyectos de fin de carrera sobre este tema.

De entre los proyectos citados, se ha trabajado con proyectos dirigidos por D. Fernando de Prada Moraga y D. Miguel Ángel Laguna Serrano, sobre los que se ha hecho un estudio del dominio para la extracción de requisitos.

El concepto de reutilización proporciona unos beneficios inherentes en lo tocante a la calidad del software, productividad del desarrollador y coste general del sistema. Sin embargo, es preciso

vencer muchas dificultades antes de que el modelo del proceso de reutilización se utilice ampliamente en la industria.

1.2. ALCANCE

El proyecto intenta la incorporación en la medida de lo posible de las personas discapacitadas en la sociedad, esta destinado tanto a aquellos que deseen obtener información sobre el tema de reutilización, requisitos y todo el proceso que conlleva, como para aquellos que quieran interesarse por componentes software reutilizables en el dominio de Informática para Discapacitados, sirviéndoles como apoyo y documentación.

Propone a empresas y particulares una manera de realizar aplicaciones en el dominio de la informática para discapacitados siguiendo un modelo determinado usando unos componentes reutilizables ya creados.

1.3. OBJETIVOS

El objetivo principal de este proyecto es el análisis del dominio de software para discapacitados, lo que implica revisar la documentación existente de requisitos. Se pueden catalogar los objetivos en varias etapas:

- Explicación acerca de la Informática para Discapacitados y una introducción a la reutilización y los requisitos.
- Utilización de un proceso de desarrollo específico y una herramienta de organización la información de diagramas de requisitos en una estructura jerárquica.
- Obtención de una colección una colección de requisitos y modelos análisis-diseño y su utilización en un ejemplo.
- Desarrollo de una herramienta llamada *Repertory Grid*, partiendo de la teoría de Constructos Personales (PCT) como ampliación de los objetivos del proyecto.

1.4. CONTENIDO DEL CD-ROM

Se enumera a continuación los contenidos del CD-ROM que acompaña a la memoria:

- Carpeta **Herramienta Requirements Reuse**, que contiene:

Manual: Archivo pdf con las instrucciones para la instalación de la aplicación.

FuentesR2: Archivo zip que contiene las fuentes necesarias de la herramienta.

Repositorio: Archivo zip con la base de datos genérica con la que se trabaja.

Discapacitados: Archivo zip con la base de datos del dominio de discapacitados.

- Carpeta **Herramienta Variabilidad**, que contiene:

WizardModels: Plantilla Excel para introducir los diagramas de Requisitos.

WizardPBVC: Plantilla Excel para realizar el análisis de Variabilidad.

RF DISCAPACITADOS: Archivo Excel con el diagrama de Requisitos Funcionales del dominio de la Informática para Discapacitados.

RNF DISCAPACITADOS: Archivo Excel con el diagrama de Requisitos No Funcionales del dominio de la Informática para Discapacitados.

Herramienta variabilidad_comunicador: Archivo Excel con el resultado del ejemplo realizado del comunicador.

- Carpeta **Herramienta Repertory Grid**, que contiene:

Repertory Grid: Archivo de Rational Rose con diagramas de casos de uso, secuencia, colaboración y clases.

Plantilla_Repertory_Grid_version_español: Versión en Español de la herramienta Repertory Grid.

Plantilla_Repertory_Grid_English_Version: Versión en Inglés de la herramienta Repertory Grid.

- Carpeta **Documentación**, que contiene:

Memoria: Archivo PDF que contiene la memoria del Proyecto de Fin de Carrera.

Presentación: Archivo PowerPoint que contiene la presentación del proyecto.

2. INFORMATICA PARA DISCAPACITADOS

2.1. SISTEMAS DE COMUNICACIÓN

2.1.1. INTRODUCCION

Dentro del amplio campo de la Informática para discapacitados, este proyecto se ha centrado en las diferentes posibilidades de comunicación para personas con imposibilidad de hacerlo por medios normales.

Las aplicaciones sobre las que se ha llevado a cabo el proyecto se basan en el aprendizaje de los distintos sistemas de comunicación existentes, dependiendo de la minusvalía que afecte al alumno.

De unos años a esta parte, las posibilidades de adquirir las habilidades comunicativas y lingüísticas para personas con grandes trastornos, se ha visto incrementada con la aparición de técnicas “asistidas”. Por medio de estas técnicas, las personas con estas deficiencias pueden lograr interactuar con su entorno físico y social.

Algunas de estas aportaciones son:

- Mecanismos físicos para poder transmitir mensajes comunicativos.
- Conjuntos de símbolos para representar el mundo de las palabras.
- Un conjunto de reglas para combinar los símbolos.
- Funciones comunicativas.

Por medio de estas aportaciones, con ayuda especializada y voluntad por parte del comunicador, se puede conseguir encontrar un sistema de comunicación adaptado a las habilidades motrices y cognitivas de cada persona.

Lo que define estos sistemas de comunicación es que la transmisión de información significativa se realiza mediante un sistema lingüístico de canales senso-motores en lugar del canal audio-vocal característico del lenguaje hablado.

Estos sistemas, en tanto que sistemas lingüísticos se componen de un conjunto de símbolos y un conjunto de reglas, mas simples y generales que las del lenguaje verbal; llegando a ser lenguajes mas limitados, menos flexibles, con menos vocabulario, con menos riqueza expresiva, con menos modelos, etc.. que el lenguaje verbal. Sin embargo, las limitaciones suponen, al mismo tiempo, ventajas para personas que los utilizan, dadas sus peculiares características personales (afectación motórica, deficiencias cognitivas, de aprendizaje,...).

Los conceptos se representan a través de símbolos, los cuales pueden ser de distinta índole: objetos, fotografías, dibujos, palabras escritas, etc. Estos símbolos le sirven a la persona discapacitada para representar la realidad, para interaccionar y controlar el entorno, para controlar su propia conducta. En el caso de niños, estos sistemas son muy adecuados, ya que facilitan enormemente su aprendizaje.

2.1.2. CLASIFICACION

Los sistemas de comunicación no vocal, se pueden dividir en dos grandes grupos:

A. Sistemas sin ayuda

Son aquellos que no requieren ningún instrumento o ayuda técnica para que tenga lugar la comunicación. Estos sistemas tienen una restringida aplicación en el campo de las deficiencias físicas. Tipos:

- * Gestos de uso común
- * Códigos gestuales no lingüísticos
- * Sistemas de signos manuales de no oyentes
- * Sistemas de signos manuales pedagógicos
- * Lenguajes codificados gestuales

B. Sistemas con ayuda.

Requieren algún tipo de asistencia externa, instrumento o ayuda técnica para que pueda tener lugar la comunicación. Tipos:

- * Sistemas basados en elementos muy representativos, tales como objetos, miniaturas, fotografías o dibujos fotográficos.
- * Sistemas basados en dibujos lineales (pictogramas), algunos de esos sistemas son:
 - El Picsyms
 - El Pictogram Ideogram Communication (PIC)
 - El Picture Communication Symbols (SPC)

* Sistemas que combinan símbolos pictográficos, ideográficos y arbitrarios, los principales son:

- El sistema Rebus
- El sistema Bliss

* Lenguajes codificados “con ayuda”. En este grupo pueden incluirse los siguientes:

- El sistema Braille
- El sistema Morse
- El sistema Taquigráfico

Si una persona necesita una ayuda de comunicación no vocal, habrá que valorar qué sistema de comunicación es adecuado para ese individuo concreto. Como ayuda a esta valoración, se pueden tener en cuenta las siguientes consideraciones:

- *Niveles de sofisticación del lenguaje.* Los SPC son apropiados para personas a quienes basta un nivel de lenguaje expresivo (vocabulario limitado y estructuras cortas).
- *Agudeza y percepción visuales.* En ocasiones aparecen problemas visuales en personas con deficiencias de origen neurológico, que dificulta la visión y diferenciación de los símbolos.
- *Habilidades cognitivas.* La edad mental de la personas debe ser adecuada para el reconocimiento de objetos dibujados, y tener la suficiente capacidad de almacenamiento mnésico y estrategias para recordar.
- *Actitud del usuario de la ayuda.* Como cualquier forma de comunicación no vocal, la persona debe tener motivación para comunicarse si ha de aprender ha usar el sistema.
- *Actitud de los oyentes.* Las figuras clave en la vida de la persona deben estar dispuestas a tomarse tiempo, la molestia y el interés de entender el nuevo modo de comunicación.
- *Alguien que colabore.* Debe haber una persona para facilitar el aprendizaje del sistema, para crear la ayuda apropiada para el sujeto y para modificarla según sea necesario.

2.1.2.1 PICTURE COMMUNICATIONS SYMBOLS (SPC)

Los símbolos pictográficos son dibujos que mantienen una estrecha relación de forma (bidimensional) con su correspondiente referente. Su nivel de abstracción y arbitrariedad es superior al mismo objeto, pero menor que otros sistemas como pueden ser: el sistema Bliss, o la ortografía tradicional. Su principal característica es el grado icónico de las imágenes.

El SPC se compone principalmente de dibujos simples, con la palabra que simboliza cada dibujo impreso encima del mismo. Si la palabra tiene un significado abstracto y, por lo tanto, no puede ser dibujada estará simplemente impresa. También se incluyen los caracteres del alfabeto, los números y los colores.

Para que el SPC sea una herramienta útil, los símbolos se diseñan tal que:

- Simbolicen palabras de uso común en la comunicación diaria.
- Sean apropiados para el uso en todos los grupos de edad.
- Sean reproducidos rápida y fácilmente por fotocopiadoras.

- Puedan ser separados para la adaptación a individuos concretos con necesidades específicas.
- Se distingan unos de otros de manera fácil y rápida.

El SPC es un sistema de comunicación apropiado para personas con distintas discapacidades y con propósitos muy distintos. Puede usarse con personas con deficiencias físicas graves, retraso mental, autismo, lesión cerebral, etc.

2.1.2.2 SISTEMA BLISS

El Bliss es un lenguaje simbólico cuyos símbolos son abstractos en la mayor parte, no teniendo nada que ver su forma con su significado. Está formado por símbolos gráfico-visuales con significado acordado y registrado, siendo capaz de proporcionar una comunicación comprensiva total.

Cada símbolo Bliss podría ser interpretado por un hablante de cualquier idioma, porque los símbolos están referidos a un significado, y este es el mismo en todos los idiomas.

Este sistema cuenta con una gran cantidad de símbolos pictográficos e ideográficos, favoreciendo la generación de otros nuevos a partir de una serie de reglas simples.

Pese a estas ventajas, el alumno tiene pocos modelos de personas que se comunican con símbolos Bliss, esto influye considerablemente en su aprendizaje.

Este sistema da la posibilidad a los minusválidos físicos de aprovechar el contenido semántico de los símbolos para crear otros, al combinarlos. Sin embargo para los deficientes mentales, las diferencias de tamaño, de angulación y de dirección, en un número limitado de formas, son difíciles de percibir. Estas personas requieren la simplicidad de una correspondencia única entre símbolo y el significado. Además habrá que tener en cuenta la edad y las capacidades físicas e intelectuales de cada alumno para valorar la ventaja o desventaja de utilizar este sistema.

2.1.2.3 SISTEMA BRAILLE

El sistema Braille es un alfabeto táctil que utilizan los ciegos para poder leer y escribir. Se ha extendido bastante y con el los métodos para crear e imprimir textos en dicho alfabeto y hacer un uso cómodo del mismo.

Tras investigaciones se determino que seis puntos era la cantidad máxima que podía percibir simultáneamente la yema de un dedo. Se creo una célula o signo generados, consistente en una estructura rectangular formada por dos filas paralelas de tres puntos cada una, a cada uno de los cuales corresponde un número, Fue así como Louis Braille creo un código táctil en relieve, que permitía leer el alfabeto, los números, los signos musicales y de puntuación.

Existe una asociación entre el conocimiento y uso del Braille y el grado de deficiencia visual y la existencia o no de otra deficiencia o limitación personal unida a la visual. Quienes mas conocen y usan el Braille son los totalmente ciegos o que solo perciben luz, mas que los que distinguen formas de objetos u objetos a una cierta distancia.

El principal problema es que los ciegos usan Braille para comunicarse pero las personas que no tienen problemas de visión no suelen conocerlo, lo cual dificulta la comunicación entre ambos. Una ventaja de un editor Braille es que sirve para que la persona con vista que no sabe Braille pueda aprenderlo ya que lo que escriba en el editor aparecerá escrito en la pantalla de dos maneras: la normal, con letras, y la Braille, con una representación de los caracteres escritos en Braille.

Todos los signos y símbolos del lenguaje escrito pueden expresarse en Braille.

El proceso de lectura es inmediato una vez que se conoce el código Braille: se pasan los dedos por las diferentes líneas, distinguiendo unos caracteres de otros y sabiendo a que carácter corresponde cada combinación de puntos.

Las personas con carencias visuales encuentran uno de sus mayores apoyos a nivel de comunicación y acceso a la información en la informática. Son muchas las posibilidades que esta puede aportar en el citado campo y han aparecido ya muchas utilidades que lo corroboran como los citados anteriormente editores o las impresoras en Braille.

2.2. INFORMATICA Y DISCAPACITADOS

El aumento de los ordenadores personales y los productos de software ha dado como resultado que los programadores se dediquen a realizar programas educacionales para su uso experimental en escuelas y colegios. Esto ha dado lugar a diversos tipos de programas que realizan muy distintas funciones (comunicación, aprendizaje, entretenimiento, etc...) por ello nos podemos centrar en algunos tipos de software.

La informática posee la capacidad de brindar soluciones totales o parciales a las personas que tienen algún tipo de discapacidad, sin embargo, muchas veces surgen una serie de barreras de acceso, pues para hacer uso de la computadora existe un conjunto de requerimientos que la persona no puede llevar a cabo de forma autónoma.

Afortunadamente, la computadora es una maquina con gran versatilidad y ofrece la posibilidad de adaptarla a las necesidades de cada usuario.

Alrededor de la informática convencional ha ido surgiendo todo un mundo de adaptaciones y ayudas técnicas, algunas veces no desarrolladas expresamente para ser utilizadas por personas con discapacidad, que facilitan el acceso al ordenador, eliminando las barreras que pudiera existir.

Detectar cuales son las barreras de acceso existentes en cada situación, la elección de la ayuda técnica mas adecuada en función de las capacidades y necesidades de este usuario y el entrenamiento necesario, son algunas de las etapas a tener en cuenta en el momento de dotar a una persona con aquella adaptación que favorecerá o posibilitará el uso de la computadora.

Dentro del campo de la informática, una de las aplicaciones más nombradas y con más expectativas es la dirigida a la Educación Asistida por Ordenador, conocida como E.A.O. Ya que en los comienzos de la informática, se hablo de la gran utilidad que los ordenadores tendrían en el campo de la educación.

En la actualidad, la mayoría de los educadores ya utilizan ordenadores no solo para la preparación de sus clases, sino también a la hora de impartir determinadas unidades didácticas a sus alumnos. De hecho, gracias al avance de Internet y a la aparición de los lenguajes visuales, se están

diseñando aplicaciones de fácil manejo para los niños que les permite aprender determinados temas de forma grafica.

En casos muy específicos y para materias muy concretas, la utilización del ordenador en los colegios resulta bastante beneficiosa para los alumnos. Además, si hay un campo en el que la informática esta resultado especialmente beneficiosa, es en la educación de alumnos disminuidos física o psíquicamente.

2.2.1 ENSEÑANZA ASISTIDA POR ORDENADOR

La educación asistida por ordenador debe ser desarrollada por psicólogos y profesores que estén en contacto con los alumnos a los que va a ir enfocada la aplicación para que esta no quede abstraída de su verdadero uso.

La Escuela Informática de Valladolid, en colaboración con diversas entidades, dedicadas a la atención e integración de personas con minusvalías, ha realizado varios proyectos sobre temas de Informática para Discapacitados. Tan solo algunos de ellos pueden considerarse como E.A.O.

Debemos de tener en cuenta una serie de consideraciones de la enseñanza asistida por ordenador (E.A.O.)

- Se deben aportar directrices claras y consistentes para el estudiante y su tutor, se debe conocer en todo momento que es lo que se debe hacer a continuación, si no, el estudiante puede perder mucho tiempo e interés por el programa.
- El tamaño y formato de las letras es muy importante sobretodo cuando se trabaja con niños, pues estos se fijaran mucho mas en los textos que puedan leer con facilidad y no en aquellos que les suponga un esfuerzo poder leerlos.
- Cada lección debe ser programada por el educador. El alumno no tiene nivel cognitivo suficiente para hacerlo por si mismo. Además la programación debe ser lo mas agradable posible.
- Las lecciones deben ser lo mas abiertas posibles para que la aplicación E.A.O. sea utilizable por el mayor numero de alumnos posibles, con sus diferentes niveles cognitivos.
- Los refuerzos, tanto positivos como negativos, deben ser suficientemente atractivos para los alumnos.
- Se tiene que buscar una manera cómoda de interactuar del alumno discapacitado tonel ordenador.

2.2.2. COMUNICACIÓN

Para no aislar a las personas, la informática facilita a las personas con algún tipo de discapacidad su integración en la sociedad. Para conseguirlo se adapta tanto el software como el hardware.

Los comunicadores pretenden lograr la interacción con el entorno a personas con discapacidades, en la medida de lo posible dotando así al usuario de independencia.

A conseguir este fin, ayudan los periféricos que están experimentando un gran aumento que beneficia a los usuarios con alguna discapacidad pues se encuentran en el mercado una amplia gama: pantallas táctiles, diferentes tipos de ratón y teclados, reconocimiento de voz, emuladores de teclado y ratón, etc..

3. INTRODUCCION A LA REUTILIZACION Y REQUISITOS

3.1. DESARROLLO DEL SOFTWARE

La actividad de desarrollo del software, desde el punto de vista económico y de negocios, esta enfocada a satisfacer las necesidades de usuario. Por lo tanto, podría decirse que el problema fundamental del desarrollo de software consiste en la satisfacción del usuario a través de las prestaciones de productos.

El intento por satisfacer las necesidades de usuario forma una espiral, ya que al satisfacer unas surgen otras nuevas. Esto lleva a la creación de un software cada vez más complejo. Se hace necesario disponer de un proceso racional para el desarrollo de aplicaciones, dando lugar a la *ingeniería del software*.

La producción del software esta limitada por tres restricciones: calidad, tiempo y requisitos. El software debe cumplir unos mínimos de calidad, producirse en el menor tiempo posible para ahorrar costes, y además necesariamente debe satisfacer los requisitos del sistema y del cliente. La reutilización del software ha sido propuesta para cumplir con las dos primeras restricciones: tiempo y calidad. Este tipo de reutilización ha demostrado su utilidad al aplicarse a dominios limitados. Sin embargo, no tiene en cuenta los requisitos porque se tiende a pensar, erróneamente, que es un proceso que lleva demasiado tiempo y retrasa la producción de código.

A pesar de esto, una buena documentación de los requisitos es fundamental para minimizar los fallos del software y garantizar su calidad. Actualmente se buscan métodos y herramientas que ayuden a producir las especificaciones de requisitos. Si los requisitos se pudieran reutilizar para

familias de productos, su desarrollo podría llevarse a cabo en un tiempo menor, a la vez que se disminuiría la probabilidad de errores.

3.2 REUTILIZACION DEL SOFTWARE

La mayoría de componentes se construye a medida, en lugar de ensamblar componentes existentes (al contrario que en el caso del hardware).

La reutilización es una característica importante para un componente software de alta calidad. El componente debería diseñarse e implementarse para que pueda volver a ser reutilizado en muchos programas diferentes. La reutilización debería ser parte integrante de cualquier proceso software.

Existen dos factores que dan lugar a la reutilización del software. De un lado las necesidades de obtener software de calidad en el menor tiempo posible, y de otro la similitud existente entre productos de la misma línea. La obtención de software a partir de elementos previamente probados, conduce al incremento en la calidad y al ahorro de tiempo.

La reutilización de software consiste en tomar sistemáticamente la experiencia previa y los elementos reutilizables para incorporarlos en desarrollos futuros. Las investigaciones han demostrado que se requiere un enfoque particular, propio y adaptado a la naturaleza del software, para reutilizar artefactos de desarrollos previos. Este enfoque se basa en la selección, especialización e integración de elementos del software que hayan sido intencionalmente diseñados, desarrollados y documentados para servir como materia prima para nuevos desarrollos.

Pero para que la reutilización este presente a lo largo de todo el ciclo de vida del software, se requieren herramientas que faciliten y fomenten tanto la reutilización como el diseño para reutilización en las distintas etapas que componen el desarrollo de un producto.

Las ventajas potenciales de la reutilización estimulan nuevos planteamientos dentro de la Ingeniería del Software. La reutilización sistemática requiere una organización apropiada y una cultura idónea. Se deben revisar las prácticas de administración, las estructuras de organización y las tecnologías utilizadas para explotar eficientemente los elementos reutilizables. Se debe contar con herramientas de apoyo, desarrolladores entrenados y con una visión de largo plazo para reutilizar el software como producto lógico o simbólico, que no se manufactura ni se deteriora, más bien se desarrolla y se adapta a los cambios.

Un aspecto esencial de la reutilización del software es la delimitación de los artefactos reutilizables. Intuitivamente, la reutilización se asocia de manera directa con la reutilización de código fuente para construir un nuevo producto software. Sin embargo en el sentido más amplio, un elemento reutilizable es cualquier producto del ciclo de vida del software con potencial de reutilización: modelos y arquitecturas de dominio, requisitos, diseño, código, componentes de base de datos, documentación y pruebas.

3.3. REQUISITOS DEL SOFTWARE

La definición de requisito que da la IEEE es:

1. Condición o capacidad necesaria para que un usuario resuelva un problema o alcance un objetivo.

2. Condición o capacidad que debe reunir un sistema o un componente para satisfacer un contexto, un estándar, o una especificación, u otro documento formal.
3. Representación documentada de una condición o capacidad como 1 o 2.

Dicho de otro modo, un requisito es una expresión de una necesidad de un usuario. Para satisfacer las necesidades del usuario, se deben representar mediante una especificación de requisitos.

3.4. REUTILIZACION DE REQUISITOS

La Reutilización del software es una alternativa tecnológica para lograr el incremento de la productividad. La reutilización de los productos iniciales del proceso de desarrollo del software estimula la reutilización a lo largo del resto del ciclo de vida y permite aprovechar mejor el esfuerzo de desarrollo.

La Ingeniería de Dominios permite capturar, organizar y representar la información útil para el desarrollo de sistemas software de manera que esta pueda ser reutilizada para crear nuevos sistemas software. Para que la reutilización esté presente a lo largo de todo el ciclo de vida del software, se requieren herramientas que faciliten y fomenten tanto la reutilización, como el diseño para la reutilización en las distintas etapas que componen el desarrollo de un producto.

El completar la especificación de requisitos de una nueva aplicación con elementos reutilizables calificados, mejora la calidad y agiliza el proceso de Ingeniería de Requisitos (IR), a la vez que propicia el ahorro de recursos en el contexto de la producción empresarial del software. Dado que la ingeniería de requisitos desencadena el proceso del ciclo de vida, su reutilización puede provocar un impacto positivo en este proceso.

La reutilización desde las etapas de captura y análisis de requisitos de la aplicación, requiere de un marco adecuado con el soporte de estructuras y herramientas. El carecer de este marco equivale a poca efectividad en la estrategia de reutilización.

Para reutilizar requisitos, al igual que cualquier otro producto del ciclo de vida, es necesario representar, clasificar, almacenar, seleccionar y adaptar los assets. Estas labores requieren el soporte de estructuras para interrelacionar los requisitos de manera coherente y así reflejar la realidad del dominio en el repositorio. Las estructuras de reutilización, por ejemplo un mecano, pueden adoptar tal nivel de complejidad que se hace necesario el empleo de herramientas automatizadas para las labores de gestión de los elementos reutilizables, tanto en la fase de *Ingeniería del Dominio* como en la de *Ingeniería de la Aplicación*.

Diversos autores recomiendan el abordar la reutilización lo más temprano posible dentro del ciclo de vida del software. Surge así el campo de estudio de reutilización de requisitos como una aproximación para aprovechar sistemáticamente los documentos existentes de requisitos con el propósito de reducir el esfuerzo general dentro del ciclo de vida del software. Aunque ha recibido poca atención, la reutilización de requisitos es un área de investigación que ofrece beneficios potenciales en el proceso de reutilización de software dado que los requisitos dentro de dominios o tareas similares tienen mayor probabilidad de compartir similitudes que los componentes de menor nivel de abstracción que los implementan. Los requisitos que se refieren a restricciones dentro de un dominio, o se refieren a estilos de presentación de la información o se relacionaba con políticas de la organización, son potencialmente reutilizables y pueden representar mas del 50% de los requisitos, lo que hace su reutilización muy atractiva desde el punto de vista de reducir el coste del desarrollo.

Los **beneficios** fundamentales de reutilizar los productos y procesos de etapas tempranas del desarrollo de software se resumen en un mejor aprovechamiento del esfuerzo de desarrollo y el estímulo a la reutilización a lo largo del resto del ciclo de vida de un proyecto. Estos beneficios se hacen patentes en el proceso de Ingeniería de Requisitos, tanto con independencia del paradigma de desarrollo, como dentro del paradigma de orientación al objeto. La reutilización de requisitos es una alternativa para completar la especificación de nuevas aplicaciones mejorando la calidad y la productividad del proceso de Ingeniería de Requisitos en el contexto de la producción empresarial del software.

Podemos distinguir entre dos tipos de requisitos:

- **Requisitos funcionales:** describen los casos de uso en los que los diferentes actores utilizan los servicios proporcionados por el sistema. Cada requisito funcional identifica el evento de activación, las pre y postcondiciones y los pasos que componen el caso de uso, así como las posibles excepciones.
- **Requisitos no funcionales:** describen aquellas características no funcionales que los clientes y usuario desean que tenga el sistema a desarrollar.

Aunque la reutilización de requisitos es una alternativa para obtener especificaciones software de mejor calidad y en menor tiempo que si se desarrollan desde cero, existen algunos **obstáculos** que superar para abordar la reutilización de requisitos:

- *Distintos niveles de descripción:* Aparecen distintos niveles de descripción en el nivel de abstracción de los requisitos.
- *Diversidad de técnicas de modelado:* Los requisitos actúan como medio de comunicación y negociación entre las diferentes personas involucradas en el proceso de desarrollo del software. Por esta razón se utilizan diversas técnicas y formatos para mostrar distintas vistas de los sistemas.
- *Bajo nivel de formalidad:* Las necesidades de comunicación y de acuerdo entre las partes involucradas impiden por lo general utilizar elevados niveles de formalidades las etapas iniciales de la ingeniería de requisitos.
- *Especificidad de los requisitos:* Los diagramas de requisitos responden a las necesidades específicas de una aplicación software. Esto plantea la necesidad de acciones especiales a fin de reutilizar requisitos en la especificación de nuevas aplicaciones.

Además, se requieren herramientas para automatizar el propio proceso de reutilización de artefactos software, pero si no se solventasen los obstáculos previamente expuestos, la complejidad de esa herramienta podría ser excesiva.

Para afrontar esos obstáculos se pueden realizar dos acciones generales. La primera acción consiste en restringir el ámbito de estudio de los requisitos que se representan mediante determinadas técnicas de modelado con un nivel de formalidad dado. Por ejemplo, se pueden escoger un nivel de formalidad medio como el que se emplea en los diagramas de escenarios, casos de uso, diagramas de actividades, flujos de datos, documentos-tareas y “workflows”. La segunda acción consiste en establecer un marco de trabajo para la reutilización de requisitos. Este marco ha de estar conformado por etapas claramente definidas, como por ejemplo:

- describir los diagramas de requisitos mediante un modelo común.
- Realizar la proyección de los diagramas integrados por ese modelo común hacia un lenguaje formal para realizar un análisis de validez de los diagramas.
- Organizar los requisitos dentro de un repositorio.

En este proyecto, ese marco de trabajo será la herramienta R^2 .

3.5. EL ENTORNO DE REUTILIZACION R²

Es un entorno de reutilización de requisitos cuyo desarrollo se realiza mediante seis proyectos de fin de carrera en el Departamento de Informática de la Universidad de Valladolid:

- Sonsoles Muñoz y Sheila Vicente. “Un entorno para la reutilización sistemática de requisitos: El gestor del Léxico”, 2003.
- J. Rodríguez y D. Jiménez. “Un entorno para la reutilización sistemática de requisitos: Cualificador de Diagramas de Requisitos”, 2003.
- E. Ecube. “Un entorno para la reutilización sistemática de requisitos: El traductor de Diagramas de Requisitos”, 2003.
- J. Saldaña. “Verificador Sintáctico de Diagramas de Requisitos”, 2003.
- D. Cabezas y J. Vicente. “Un entorno para reutilización sistemática de requisitos: El Gestor de Dato”s, 2002.
- J. Arranz y M. Rico. “Un entorno para la reutilización sistemática de requisitos: El editor de Diagramas”, 2002.

El entorno permite integrar diferentes técnicas semiformales de modelado de requisitos en el contexto de la reutilización del software.

Los diversos diagramas de requisitos pueden ser ingresados directamente por parte del reutilizador, o bien ser importados desde otros entornos a través de un interfaz con el lenguaje XML. Los diagramas se almacenan en una base de datos ORACLE. Dado el bajo nivel de formalidad de los diagramas de requisitos, el entorno comprende una interfaz con una herramienta con redes de Petri coloreadas. Mediante este interfaz el entorno permite exportar a redes de Petri la información de requisitos y recibir desde ahí la información de verificación y validación de los requisitos reutilizables.

En la figura 1 se muestra la arquitectura del entorno de reutilización de requisitos. Este, esta formado por ocho elementos principales con sus correspondientes subelementos:

1. Interfaz de Usuario
2. Traductor de diagramas
3. Gestor de léxico
4. Editor de requisitos
5. Gestor de datos
6. Verificador sintáctico
7. Gestor de repositorio
8. Base de datos

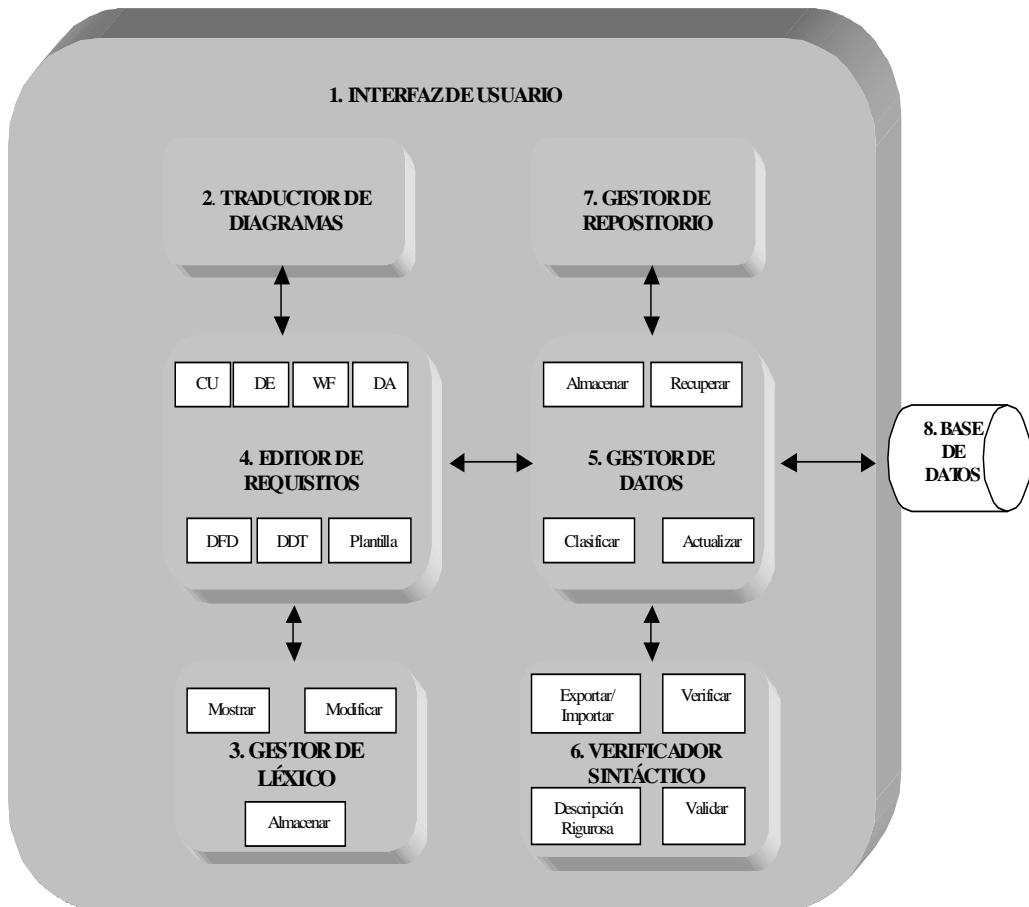


Figura 1: Arquitectura del entorno de reutilización de requisitos

1. Interfaz de usuario

La interfaz de usuario fue propuesta incluyendo las opciones de menú y vistas necesarias para que el reutilizador interactúe con un entorno de ventanas, incluyendo opciones de edición, archivo, vistas, herramientas de reutilización y ayuda.

La estructura básica de la interfaz de usuario que finalmente se desarrollo dividiéndose entre una vista de edición de Diagramas (o vista de proyectos), y una vista de edición de objetivos (o vista de objetivos).

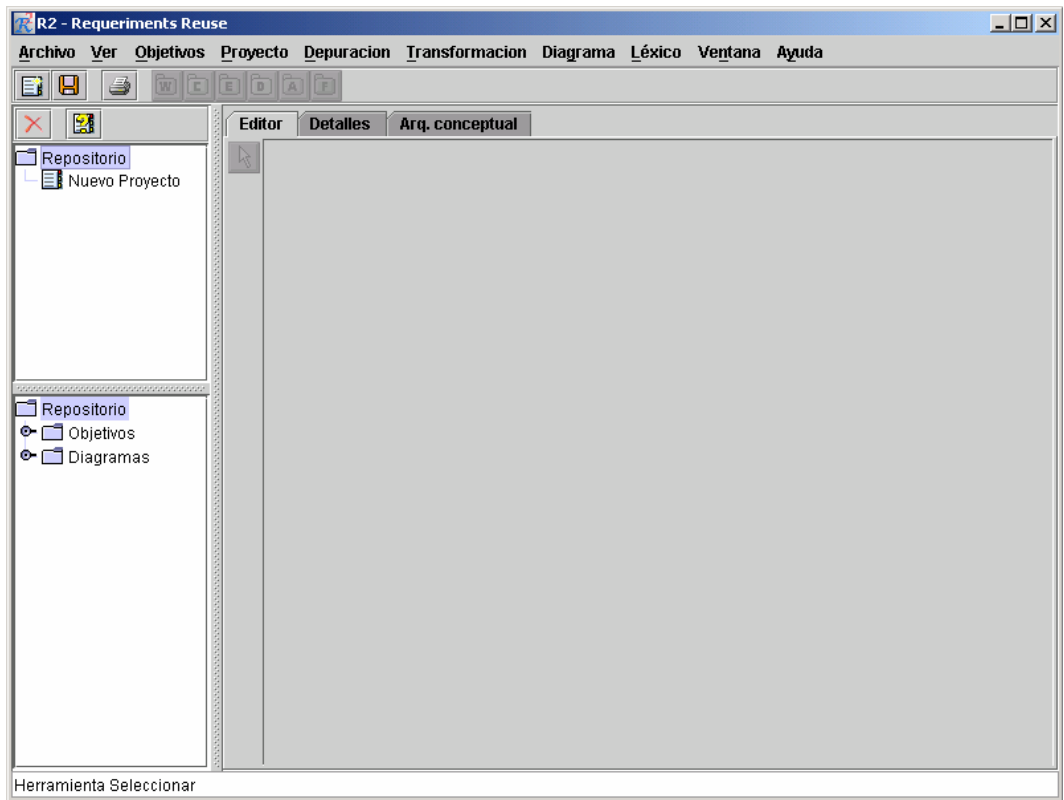


Figura 2: Vista de la interfaz de usuario del entorno R²

2. Traductor de diagramas

Su función consiste en apoyar la reescritura de diagramas de requisitos dentro de un conjunto de técnicas de modelado de requisitos.

3. Gestor del léxico

Apoya las labores de definición y control del lenguaje del dominio en el que se modelan los diagramas de requisitos.

Incluye la funcionalidad para la definición, almacenamiento y mantenimiento de conceptos y los relaciona con elementos del metamodelo (campos de base de datos). Las operaciones básicas que realiza son:

- **Mostrar:** Realiza un listado de los nombres de los símbolos que componen el diccionario léxico, dando opción de ver los nombres, nociones e impactos de uno determinado.
- **Modificar:** Permite variar nombre, noción y/o impacto de uno o mas símbolos del diccionario.
- **Almacenar:** Si un símbolo no se encuentra almacenado en el repositorio, añade este símbolo, que pasa a formar parte del diccionario léxico.

4. Editor de requisitos

Este elemento incluye la funcionalidad necesaria para crear y modificar diagramas de requisitos que se modelan mediante diversas técnicas de modelado. Esta diversidad se describe con apoyo de un esquema conceptual. Este esquema conceptual constituye un metamodelo de representación de requisitos.

5. Gestor de Datos

Es el elemento que controla, de manera centralizada, la información de los requisitos. Realiza cuatro operaciones relacionadas con los demás elementos del entorno:

- Almacenar: Ubicar la información proveniente de los restantes elementos en la base de datos.
- Recuperar: Atender a las solicitudes de información de los elementos restantes del entorno.
- Clasificar: Mantener índices especializados para la reutilización de requisitos.
- Actualizar: Mantener la consistencia de la información ante las modificaciones que indiquen los restantes elementos del entorno.

6. Intercambio de datos

Apoya la detección de inconsistencias lógicas en los diagramas de requisitos con base en la descripción rigurosa de diagramas. En el Analizador de Consistencia se concentran las funciones de intercambio de información entre el entorno y otras aplicaciones. Las funciones son:

- Exportar/importar: Interfaz para intercambiar datos con otras aplicaciones, como Rational Rose y Together.
- Descripción rigurosa: Interfaz para llevar la información de requisitos a redes de Petri coloreadas. Para ello se utiliza XML como formato intermedio entre R^2 y la herramienta Design/CPN.
- Verificación: Recibir información de valoración de la calidad sintáctica de los requisitos reutilizables. Esta información se recibe desde la herramienta externa Design/CPN.

El entorno soporta la verificación de diagramas de requisitos mediante la proyección de los diagramas a una representación rigurosa en CPN. Esta verificación se realiza de manera automática para determinar la ausencia de bloqueos, finitud del conjunto de estados, y ausencia de conflictos en modelos de redes de Petri.

7. Gestor de repositorio

Apoya la determinación cuantitativa de propiedades de los diagramas de requisitos. Se han propuesto diferentes métricas para los requisitos, de las que en R^2 se miden:

- Tamaño de los diagramas:
 - i. Numero de acciones atómicas en el flujo principal.
 - ii. Numero de acciones atómicas en cada flujo alternativo.
 - iii. Camino mas largo, desde la primera acción atómica hasta la acción atómica final.
 - iv. Numero de flujos alternativos (las ramas del flujo principal).

- Factores del “entorno o dominio”:
 - i. Numero de actores.
 - ii. Numero total de “metas y objetivos”.
 - iii. Numero de Stakeholders.
- métricas compuestas:
 - i. Numero total de acciones atómicas en los flujos alternativos.
 - ii. Numero total de acciones atómicas en todos los flujos.
 - iii. Numero de acciones atómicas por actor.
 - iv. Numero de acciones atómicas por objetivo del dominio.
 - v. Numero de metas por cada stakeholder.

8. Base de datos

Es el elemento que contiene físicamente la información de los diagramas de requisitos. La base de datos responde al esquema conceptual del metamodelo de requisitos. Los elementos principales que debe contener la base de datos son:

- Documentos de requisitos.
- Representaciones de requisitos y relaciones entre ellas.
- Unidades de Modelado y relaciones entre ellas.
- Relaciones entre unidades de modelado y representaciones de requisitos.

La versión de R² utilizada para este proyecto es una versión para PC cuya base de datos es Access y en la que alguna de las funcionalidades de la aplicación no están incluidas o están inhabilitadas.

3.6. EVALUACION DE LA REUTILIZACION

3.6.1. ESTUDIO DE RECURSOS REUTILIZABLES

Cualquier estudio sobre recursos de software estaría incompleto sin estudiar la reutilización. Los bloques de software contruidos, habrán sido catalogados para una consulta fácil, estandarizados para una fácil aplicación y validados para una sencilla aplicación.

Bennatan sugiere cuatro categorías de componentes software a tener en cuenta a la hora de realizar una evaluación de recursos:

- *componentes ya desarrollados*: El software existente se puede adquirir de una tercera parte o provenir de un desarrollo internamente para un desarrollo anterior. Estos componentes están listos para ser utilizados en el proyecto actual y se han validado totalmente.
- *componentes ya experimentados*: Las especificaciones, diseño, código o datos de prueba existentes y desarrollados para proyectos anteriores que son similares al software que se va a construir para el proyecto actual. Los miembros del equipo software actual ya han tenido la experiencia completa en el área de la aplicación representada para estos componentes. Las modificaciones, por tanto, requeridas para componentes de total experiencia, tendrán un riesgo relativamente bajo.
- *componentes con experiencia parcial*: Las especificaciones, los diseños, código o los datos de prueba existentes desarrollados para proyectos anteriores que se relacionaba con el software que se va a construir para el proyecto actual, pero que requerirán una modificación sustancial. Los miembros del equipo software actual ha limitado su

experiencia solo al área de aplicación representada por estos componentes. Las modificaciones, por tanto, requeridas para componentes de experiencia parcial tendrán bastante grado de riesgo.

- *componentes nuevos*: Los componentes de software que el equipo de software debe construir específicamente para las necesidades del proyecto actual.

Deberían considerarse las directrices siguientes por el planificador de software cuando los componentes reutilizables se especifiquen como recurso:

1. Si los componentes ya desarrollados cumplen los requisitos del proyecto, se deben adquirir. El coste de la adquisición y de la integración de los componentes ya desarrollados serán casi siempre menores que el coste para desarrollar el software equivalente. Además, el riesgo es relativamente bajo.
2. Si se dispone de componentes ya experimentados, los riesgos asociados a la modificación y a la integración generalmente se aceptan. El plan del proyecto debería reflejar la utilización de estos componentes.
3. Si se dispone de componentes de experiencia parcial para el proyecto actual, su uso se debe analizar con detalle. Si antes de que se integren adecuadamente los componentes con otros elementos del software se requiere una gran modificación, se debe proceder cuidadosamente. El coste de modificar los componentes de experiencia parcial algunas veces puede ser mayor que el coste de desarrollar componentes nuevos.

La economía de la reutilización del software en teoría debería proporcionar a las organizaciones de software unas ventajas notables en lo tocante a la calidad y a tiempos de realización. Estas, a su vez, deberían de traducirse en unos ahorros de costes. Para apoyar esta intuición se necesitan datos reales, es decir, desarrollar un análisis de costes y beneficios para la reutilización.

3.6.2. IMPACTO SOBRE LA CALIDAD, PRODUCTIVIDAD Y COSTE.

A lo largo de los últimos años, existen evidencias procedentes de casos prácticos en la industria que indican que es posible obtener notables beneficio de negocios mediante una reutilización del software. Mejoran la calidad del producto, la productividad del desarrollador y los costes en general.

Calidad. Un componente del software que se desarrolle para su reutilización estará verificado en lo tocante a su corrección y no contendrá defectos. En realidad, la verificación no se lleva a cabo de forma rutinaria, y los defectos pueden aparecer. Sin embargo, con cada reutilización, los defectos se van hallando y van siendo eliminados y como consecuencia mejora la calidad del componente. La reutilización proporciona un beneficio no despreciable en términos de calidad y fiabilidad del software proporcionado.

Productividad. Cuando se aplican elementos reutilizables a lo largo y ancho del proceso de software, se invierte menos tiempo creando los planes, modelos, documentos, código y datos necesarios para crear un sistema fiable. Se proporciona un mismo nivel de funcionalidad al cliente con menos esfuerzo.

Coste. El ahorro neto de costes para la reutilización se estima proyectando el coste del proyecto si se hubiera desarrollado este desde cero, y restando la suma de los costes asociados para la reutilización, y el coste actual del software cuando este finalmente se implanta.

Los costes asociados a la reutilización incluyen:

- Modelado y análisis del dominio.
- Desarrollo de la arquitectura del dominio.
- Incremento de la documentación para facilitar la reutilización.
- Mantenimiento y mejora de elementos de la reutilización.
- Licencias para componentes adquiridos externamente.
- Creación o adquisición y funcionamiento de un depósito para la reutilización.
- Formación del personal en el diseño y construcción para la reutilización.

Aun cuando los costes asociados al análisis del dominio y la operación de un depósito para la reutilización pueden resultar notables, se pueden amortizar en muchos proyectos. Muchos de los demás costes indicados anteriormente afectan a temas que forman parte de las buenas maneras de la ingeniería del software, tanto si se considera prioritaria la reutilización como si no.

4. ANALISIS DE REQUISITOS EN EL DOMINIO DE INFORMATICA PARA DISCAPACITADOS

4.1. INTRODUCCION

La **Ingeniería de Requisitos** (IR) es uno de los principales campos de trabajo para la Ingeniería del Software. Esto se debe a que los requisitos representan uno de los principales problemas durante el desarrollo del software y, su incumplimiento, una de las principales quejas de los clientes.

Dentro de esta rama de la ingeniería del software, los objetivos aparecen como una de las propuestas de modelado para fijar el proceso de definición de requisitos. Los objetivos proporcionan varias ventajas respecto a otros mecanismos como su mayor estabilidad (es más difícil que cambien las razones por las que se necesita un sistema, que la forma en que se realizan sus funciones) o su posibilidad de analizar distintas alternativas y seleccionar la mejor a partir de los requisitos no funcionales. Además, este análisis puede ser realizado adaptando técnicas de Inteligencia Artificial (IA), mediante un razonamiento formal sobre los modelos de objetivos.

Uno de los principales desafíos de la IR es la manera de elicitar estos requisitos. Aunque los objetivos proporcionan una manera más natural de expresarlos, hay poco trabajo realizado en su elicitación. Los requisitos de un sistema se obtienen básicamente del Universo del Discurso (UdeD). En el UdeD están todas las fuentes de información, incluso los interesados. Obtener o elicitar estos requisitos de los interesados es una tarea difícil, tanto por el conocimiento tácito (oculto o profundo) como por las diversas opiniones, probablemente contrapuestas, que se pueden obtener.

La **Ingeniería de Requisitos Orientada al Objetivo** (Goal-oriented requirement engineering), promueve el enfoque de la utilización de objetivos como base para obtener los requisitos, incorporando de esta forma un punto de vista intencional, es decir, el propósito del sistema.

Introducir un punto de vista intencional permite que los interesados expresen sus necesidades de una manera mas natural, centrándose en lo que quieren (sus objetivos) frente a la manera de alcanzarlos (los requisitos convencionales). A partir de los objetivos, los requisitos se pueden derivar como maneras de alcanzar esos objetivos.

Los objetivos se suelen estructurar en árboles descomponiéndose en otros obligatorios u optativos. Así, es posible estudiar alternativas en los requisitos y verificar la completitud de un conjunto de requisitos con respecto a los objetivos planteados. Además este estudio puede hacerse de manera formal si los objetivos están suficientemente formalizados. Por tanto, los objetivos ayudan a identificar, justificar y organizar los requisitos. Otras ventajas de los requisitos son que los requisitos son más independientes de la tecnología, que el tratamiento de requisitos no funcionales (RNF) y conflictos es más natural y que guían el desarrollo software permitiendo trazar los artefactos software hasta sus objetivos finales.

Existen diversos enfoques dentro de la IR Orientada a Objetivos, diferentes entre si, si bien los conceptos básicos que utilizan son similares. Los enfoques se diferencian básicamente en dos factores:

- La actividad de la IR al que se enfocan (elicitación, modelado o análisis de requisitos).
- El grado de formalismo que soportan.

Por ejemplo, la propuesta *KAOS* (Knowledge Acquisition in Automated Specification of Software) esta enfocada al modelado de requisitos con un alto grado de formalismo, lo que permite realizar un proceso de análisis formal de los requisitos. En cambio, el *NFR Framework* aunque también esta enfocado al modelado de requisitos, se centra en RNF (Requisitos no Funcionales), y utiliza un enfoque cualitativo mucho menos formal. Un enfoque basado en *NFR Framework* es *i**, centrado en la elicitación de requisitos en las fases iniciales de la IR, es decir en el modelado de negocio. Otras propuestas buscan integrar otras técnicas como escenarios en el modelado de los objetivos como el caso de *GBRAM* (Goal Based Requirements Analysis Method). Su principal interés es que define un proceso en el cual los escenarios se crean a partir de los objetivos y, a su vez ayudan a descubrir objetivos ocultos.

Un problema de estas propuestas es que no proporcionaban un enfoque sistemático para la elicitación de los elementos, ni de las relaciones entre ellos. Por lo general no se dan técnicas orientadas específicamente a la obtención de los objetivos de los interesados. Si existen propuestas que utilizan casos de uso o escenarios con objetivos, pero en la mayoría de los casos los objetivos no tienen la importancia que se les da en la IR Orientada a Objetivos. Seguidamente se van a explicar dos de los enfoques orientados a objetivos más importantes.

4.1.1. NFR FRAMEWORK

El propósito fundamental del NRF Framework es tratar con los requisitos no funcionales. Los RNF son considerados como una parte fundamental de la Ingeniería del Software desde hace tiempo, pero solo han sido considerados desde la fase de diseño, por eso existen pocas propuestas para introducirlos de una manera sistemática en el desarrollo a partir de requisitos.

El NFR Framework aboga por analizarlos en la fase de requisitos por medio de objetivos. El primer problema que aparece es que los objetivos tradicionales se deben poder verificar si se han alcanzado, pero los RNF son mucho más difusos, ya que no suelen tener una definición completa. Por tanto, no siempre se puede afirmar si se ha logrado el objetivo al depender de muchos factores. Para solucionar esto Cheng Define un nuevo concepto: el *Softgoal*, es decir, un objetivo para el cual su satisfacción no siempre se puede establecer de manera clara. Así, un Softgoal puede considerarse alcanzado o no, pero también parcialmente satisfecho y parcialmente insatisfecho. Por tanto, no se habla de satisfacción sino de grado de satisfacción.

A partir de este concepto se define un modelo, el Softgoal Interdependence Graph (SIG). Este modelo permite relacionar Softgoals por medio de relaciones típicas dentro de los modelos de objetivos (descomposición en árboles And-Or) y por medio de un conjunto de nuevas relaciones denominadas enlaces de correlación. Estos enlaces son la base para calcular el grado de satisfacción.

Otros elementos que se utilizan en los SIG son las operacionalizaciones (operationalization), que permiten enlazar los requisitos no funcionales con los funcionales y las afirmaciones (claims) que modifican las correlaciones según sea cierta o no la afirmación. Además, es posible indicar la prioridad de los Softgoals por medio de signos de exclamación.

Gracias a las relaciones es posible especificar explícitamente como unos Softgoals afectan a otros. De esta forma, dando valores a algunos Softgoals y propagándolos según las correlaciones, se puede obtener el grado de satisfacción del resto.

4.1.2. I* FRAMEWORK



I* es un enfoque, propuesto por Yu, que utiliza la idea de Softgoal. La principal peculiaridad del modelado de negocio sobre otros campos de la IR es la importancia de los Agentes. Un Agente se define como una entidad que existe en la organización, que tiene unos objetivos y que puede realizar tareas o utilizar recursos para alcanzar dichos objetivos o ayudar a otros agentes a alcanzar sus objetivos. En esta definición se incluyen los elementos de modelado de esta propuesta: Objetivos (Goals y Softgoals), agentes, tareas (manera de alcanzar un objetivo) y recursos (objeto que es necesario para realizar una tarea o para alcanzar un objetivo).

A partir de estos elementos se definen dos modelos en los que los agentes son la pieza fundamental: En el primero, el Strategic Diagram (SD) se modelan las dependencias entre actores. Estas dependencias pueden deberse a objetivos (Goals o Softgoals), a tareas o a recursos y dan una visión global del sistema. En cuanto al segundo, denominado Strategic Rationale (SR) lo que se modela es el razonamiento interno del agente mediante objetivos y tareas.

A continuación, en nuestro caso de estudio, utilizando la herramienta de desarrollo R^2 se analizó cuales eran los requisitos tanto funcionales como no funcionales del Dominio de la Informática para Discapacitados, buscando el mayor número de variantes posibles para poder abarcar todo el dominio de la informática para discapacitados. Se introdujeron esos requisitos funcionales y no funcionales en la herramienta R^2 e independientemente, se estudiaron los casos de uso asociados también a ese dominio de Informática para Discapacitados, mediante la herramienta R^2 .

Seguidamente se compararon el diagrama de objetivos con el de casos de uso que se había generado y a cada caso de uso se le asignó un objetivo de manera que la secuencia de cada objetivo este explicada en los casos de uso y cuando se tenga que cumplir un determinado objetivo, se deberán seguir los pasos del caso de uso asociado a el. A continuación se realizó una búsqueda en la base de datos de la aplicación R^2 para encontrar la relación entre los casos de uso y los objetivos.

Para concluir este capítulo se verá un ejemplo que se realizó partiendo del estudio previo del Dominio de la Informática para discapacitados realizado anteriormente.

4.2. OBJETIVOS

El Dominio de la informática para discapacitados es bastante complejo debido al gran número de variantes que puede haber a la hora de tener distintas discapacidades y distintas maneras de tratar esas discapacidades, el objetivo fue conseguir el diagrama de requisitos mas completo posible para que tenga cabida todo lo relacionado con la Informática para discapacitados, para ello hubo que buscar tanto los requisitos funcionales como los no funcionales e ir obteniendo todos los puntos de variabilidad posibles. El uso de la herramienta R^2 para poder expresar estos requisitos fue muy útil.

En la herramienta R^2 el grado de variabilidad viene dado por varias opciones:

- **OB** (Obligatoria): Se deben de realizar todos los subobjetivos.
- **OP** (Optativa): Se debe de realizar uno de los subobjetivos.
- **MU** (Múltiple): Se debe de realizar al menos uno de los subobjetivos.

Algunos de los **Requisitos Funcionales** que se identificaron son:

- Transportabilidad del programa a diferentes entornos, como pueden ser un ordenador un PDA, o un portátil.
- Permitir poder configurar distintos mecanismos de entrada para distintos tipos de discapacidades.
- Configurar la manera de mostrar los datos en la pantalla para poder abarcar un amplio número de alumnos con distintas deficiencias.
- Seleccionar un conjunto de símbolos adecuados al tipo de usuario al que va destinado el proyecto.
- Posibilidad de ampliación de librerías.
- Deberá ser posible realizar algunas de las siguientes operaciones para controlar la organización de la información de los alumnos:
 1. Visualizar un listado con los usuarios registrados.
 2. Posibilidad de dar de alta un nuevo alumno.
 3. Posibilidad de dar de baja un alumno registrado.
 4. Opción de modificar la ficha de un alumno.
- Permitir multiusuarios en la aplicación.
- Cada alumno debe ser diferenciado en la aplicación por el educador, adaptándose así sus características concretas.
- Cada lección de E.A.O. debe prepararse para calificar o juzgar las respuestas dadas por el alumno y compararlas con las respuestas aportadas por el autor de la aplicación o el educador.
- La lección de cada día debe ser programada por el educador.
- Permitir la realización de animaciones así como su preparación por parte del educador.
- Necesidad de una evaluación inmediata del alumno en las lecciones.
- Permitir al alumno la comunicación con el entorno.
- Realizar un seguimiento a los alumnos para ver su progresión.
- Permitir al educador realizar una configuración manual (el educador introduce todos los parámetros de la aplicación) o automática (con unos parámetros ya establecidos dependiendo de la discapacidad del alumno).
- Poder realizar las actividades mediante una comunicación remota o local.

Algunos de los **Requisitos No Funcionales** fueron:

- Facilidad de uso, hay que permitir que la aplicación resulte fácil para el alumno, así como adaptarse a distintos alumnos con distintas discapacidades.
- Fomentar la autoestima, la independencia y la comunicación.
- Ayudar a la educación del usuario, fomentando el interés y mejorando el aprendizaje.
- Permitir añadir más tipos de discapacidades y más mecanismos para tratar dichas discapacidades.
- Facilitar la instalación, gestión y modificación de contenidos.

4.2.1. REQUISITOS FUNCIONALES

El diagrama 1 es el diagrama principal de requisitos funcionales, los diagramas 2, 3 y 4 son los subobjetivos asociados a este diagrama principal.

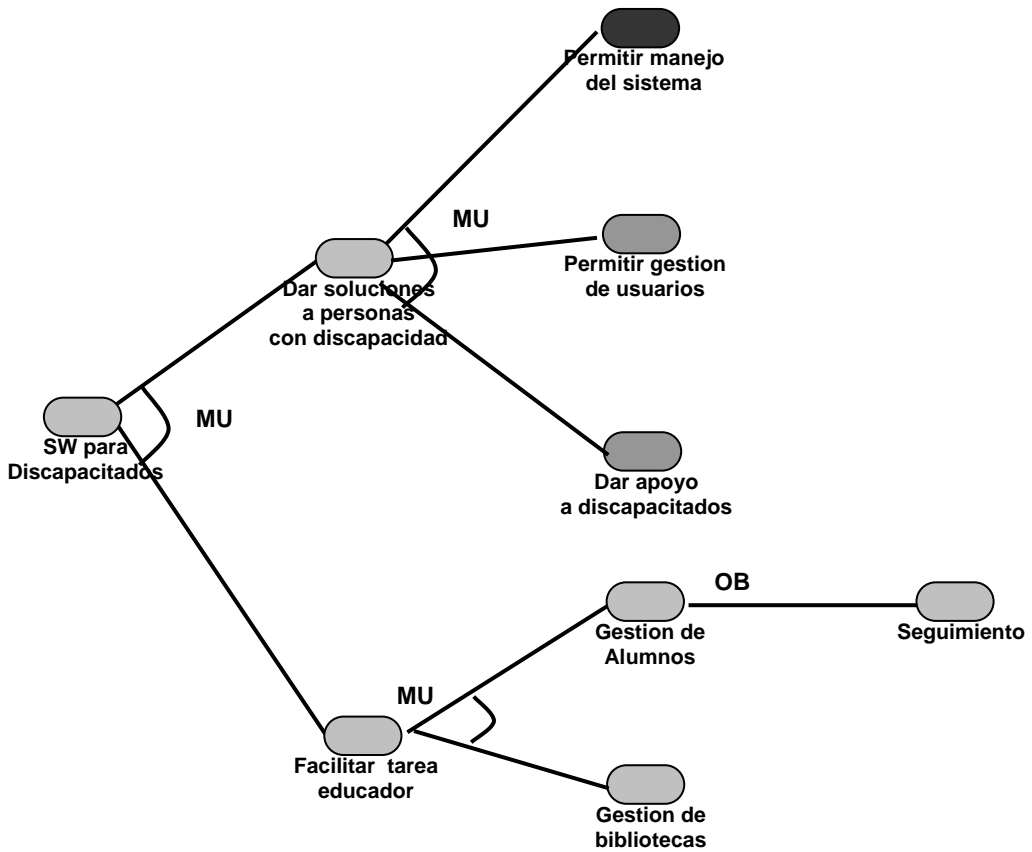


Diagrama1: Diagrama principal de requisitos funcionales en el dominio de Informática para Discapacitados.

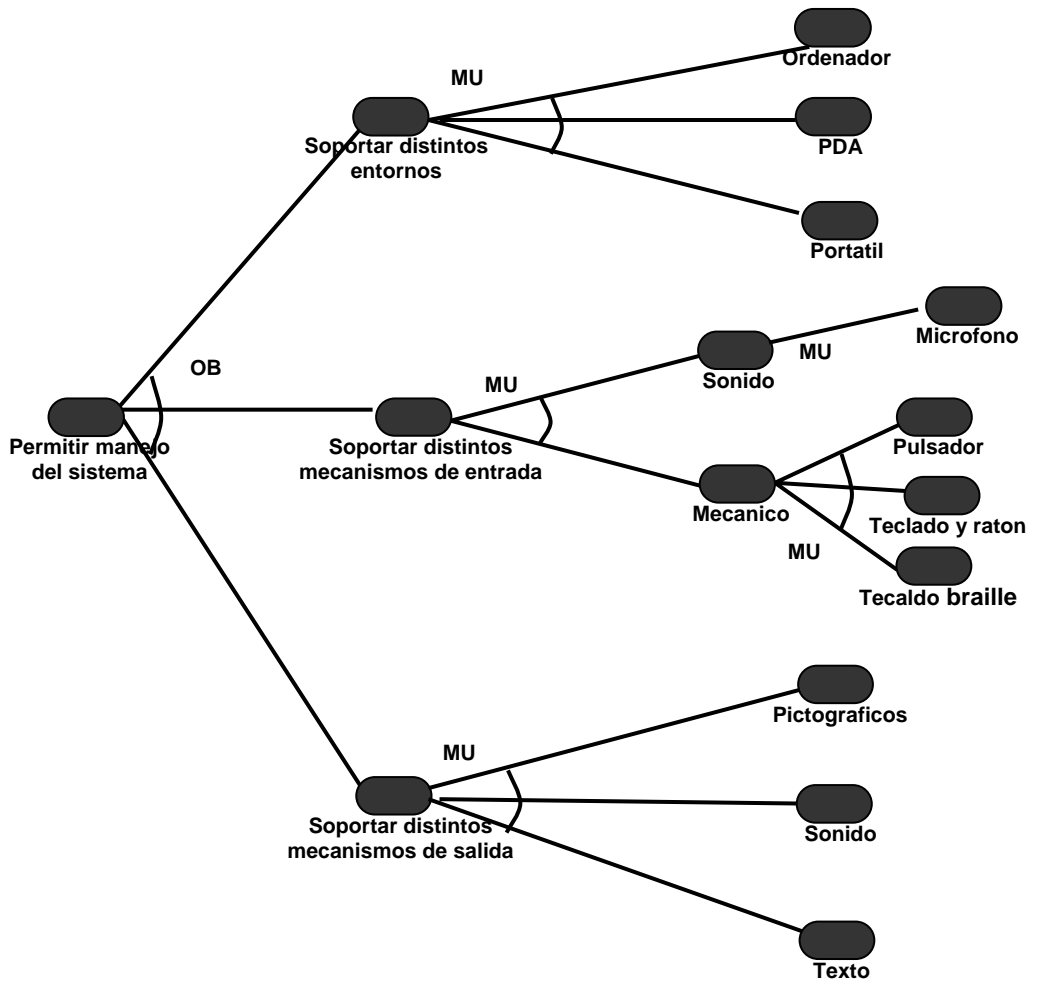


Diagrama 2: Diagrama de Subobjetivos asociados al objetivo "Permitir manejo del sistema".

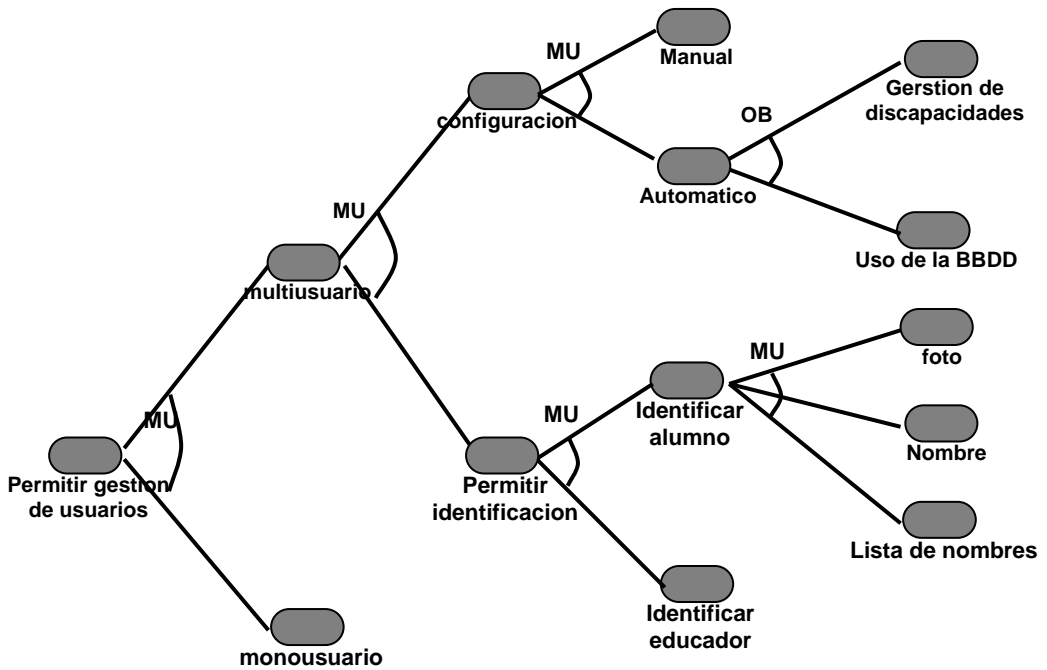


Diagrama 3: Diagrama de Subobjetivos asociados al objetivo "Permitir gestión de usuarios".

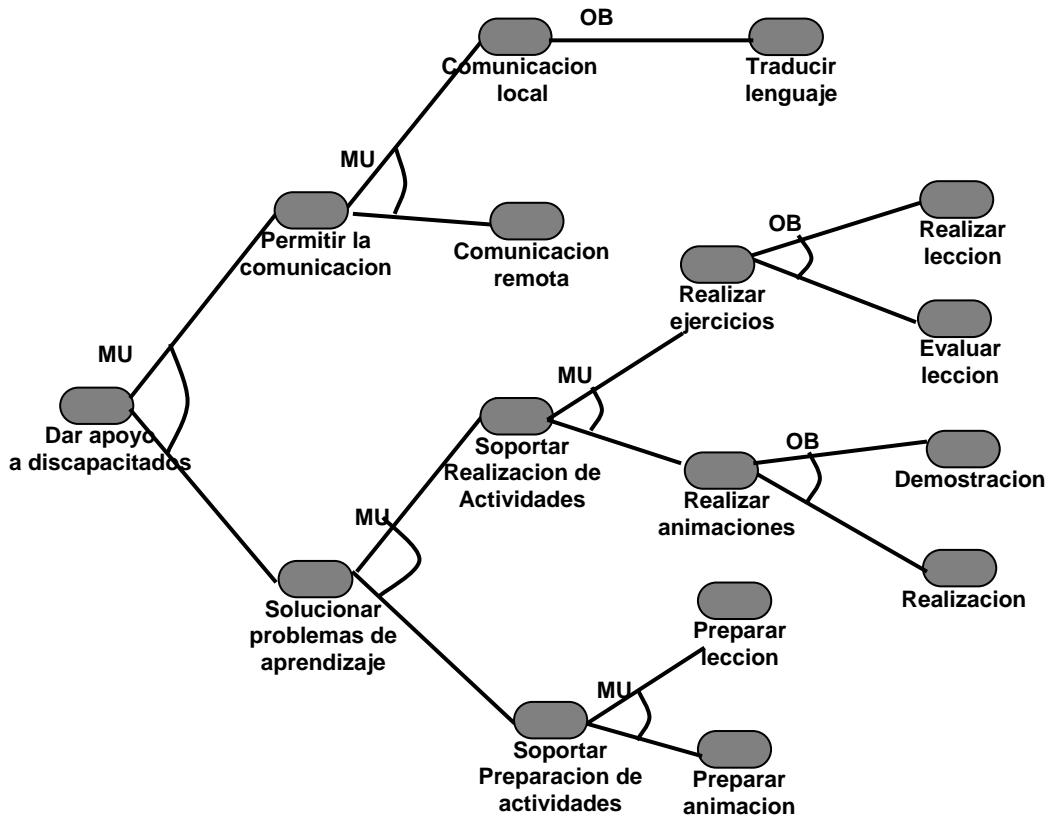
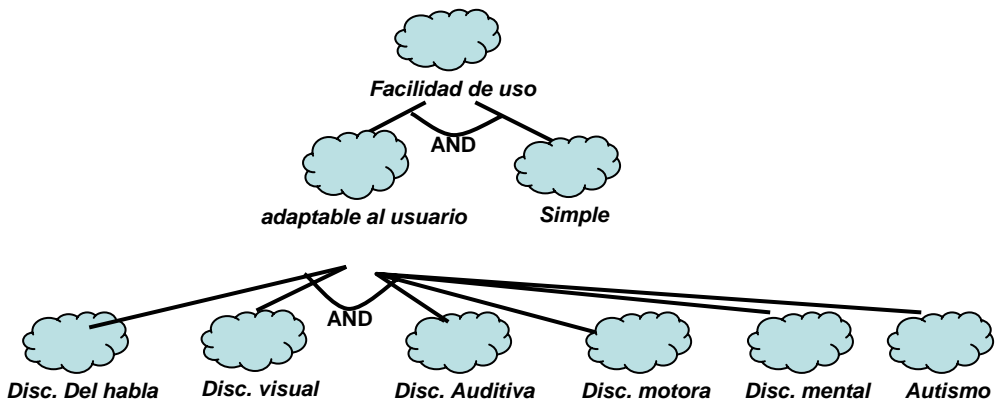


Diagrama 4: Diagrama de Subobjetivos asociados al objetivo “Dar apoyo a discapacitados”.

4.2.2. REQUISITOS NO FUNCIONALES



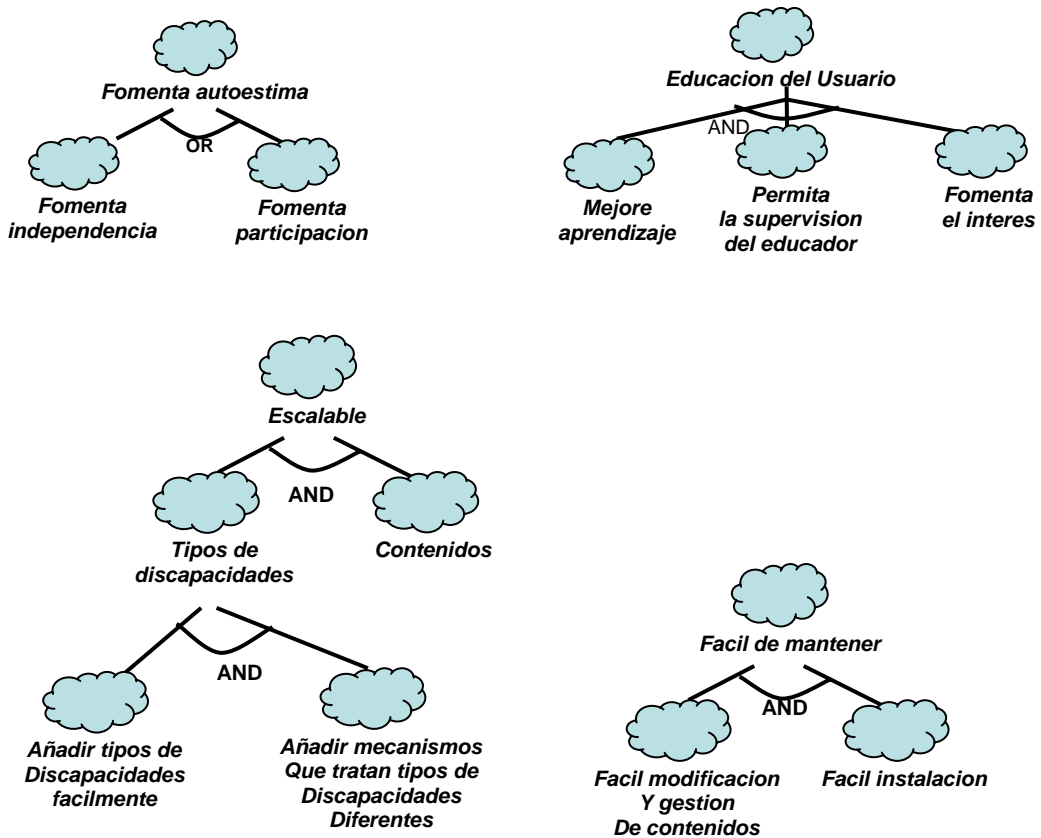


Diagrama 5: Diagramas de Requisitos no Funcionales.

4.3. CASOS DE USO

Por otro lado independientemente de los objetivos identificados anteriormente, se crearon el conjunto de casos de uso que se aplicarían al Domino de la Informática para Discapacitados y como actuarían los actores al querer usar una aplicación de informática para discapacitados.

También se obtuvieron el mayor número de casos de uso posibles para abarcar todas las posibilidades del dominio tratado. La manera de organizar los casos de uso en la herramienta R² fue mediante el uso de paquetes, dentro de cada paquete esta descrito un conjunto de casos de uso.

La principal idea ha sido conseguir cuantos más puntos de extensión mejor debido a que eso implicaba diferentes opciones a la hora de escoger entre casos de uso.

4.3.1. DEFINICION DE ACTORES

Los actores son personas o entidades externas a la aplicación que interactúan con ella, realizando intercambio de información. Este podrá ser tanto de entrada como de salida.

En este caso se han identificado tres:

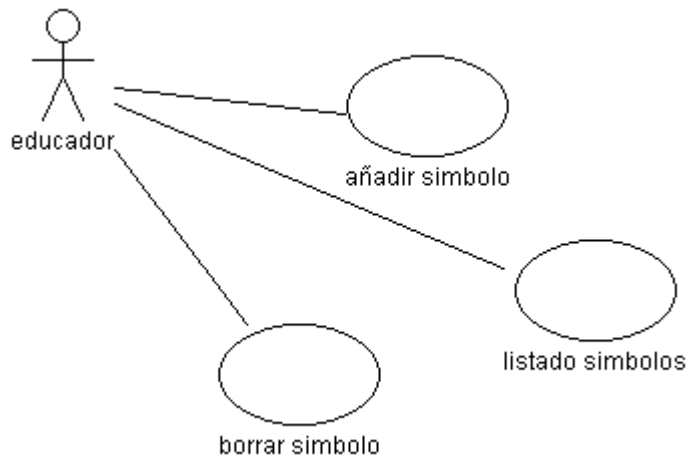
- **Alumno:** Es la persona a la que va dirigida la aplicación.

- **Educador:** Es el encargado de preparar las actividades y gestionar la aplicación para que sea utilizada por el alumno.
- **Usuario:** Es una generalización de alumno y educador, significa que cualquiera de los dos anteriores puede desempeñar ese intercambio de información.

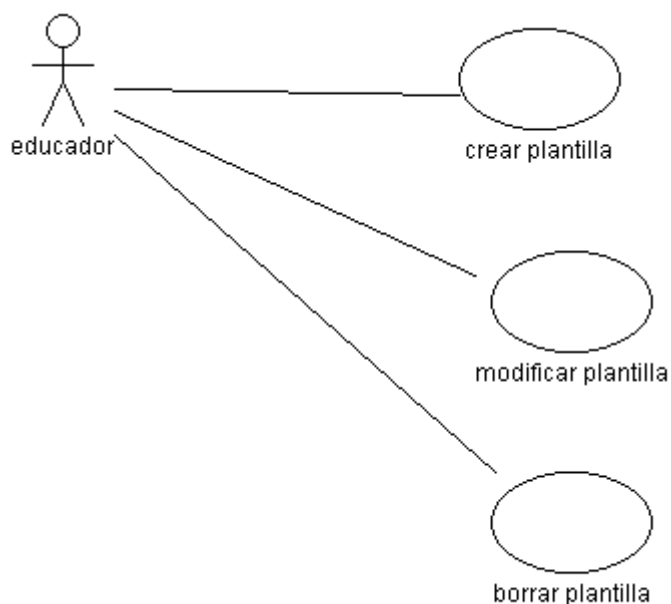
4.3.2. DIAGRAMAS DE LOS CASOS DE USO

El diagrama de uso es muy útil para definir como debería ser el comportamiento de una parte del sistema, ya que solo especifica como deben comportarse y no como están implementadas las partes que define.

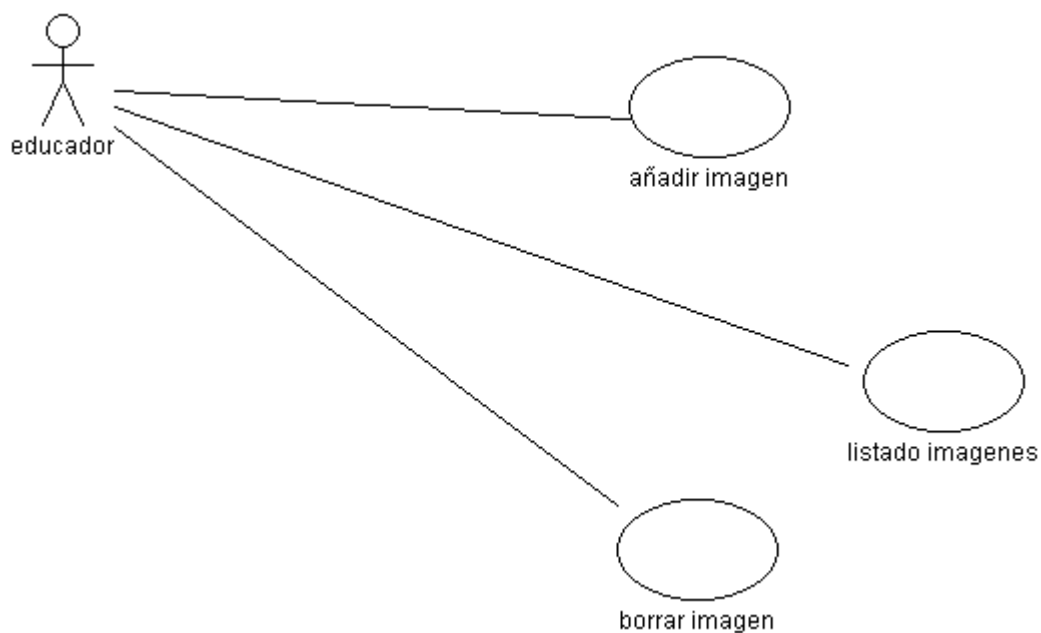
Se organizan los diagramas en paquetes debido a elevado número de casos de uso y para seguir con la estructura de la aplicación R².



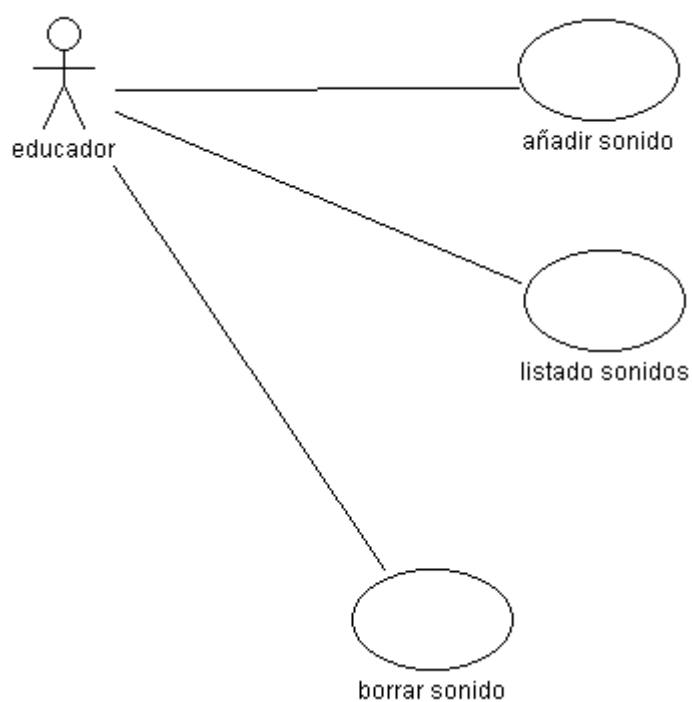
Paquete de Casos de Uso 1: Soporte gestión de símbolos.



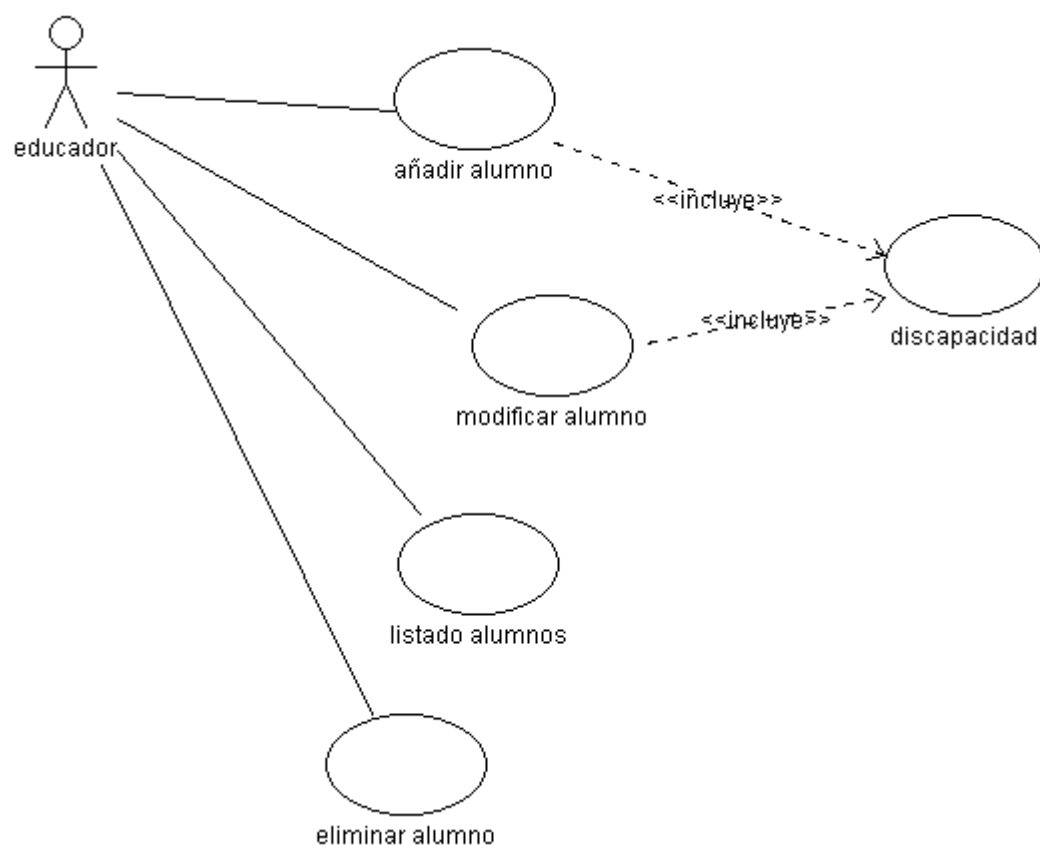
Paquete de Casos de Uso 2: Soporte gestión plantillas.



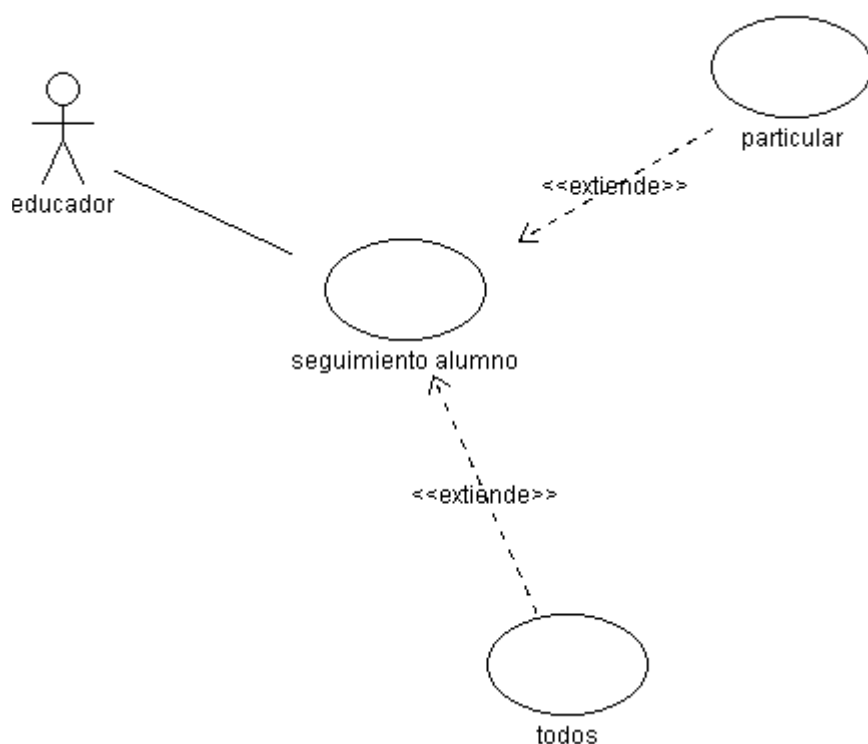
Paquete de Casos de Uso 3: Soporte gestión imágenes.



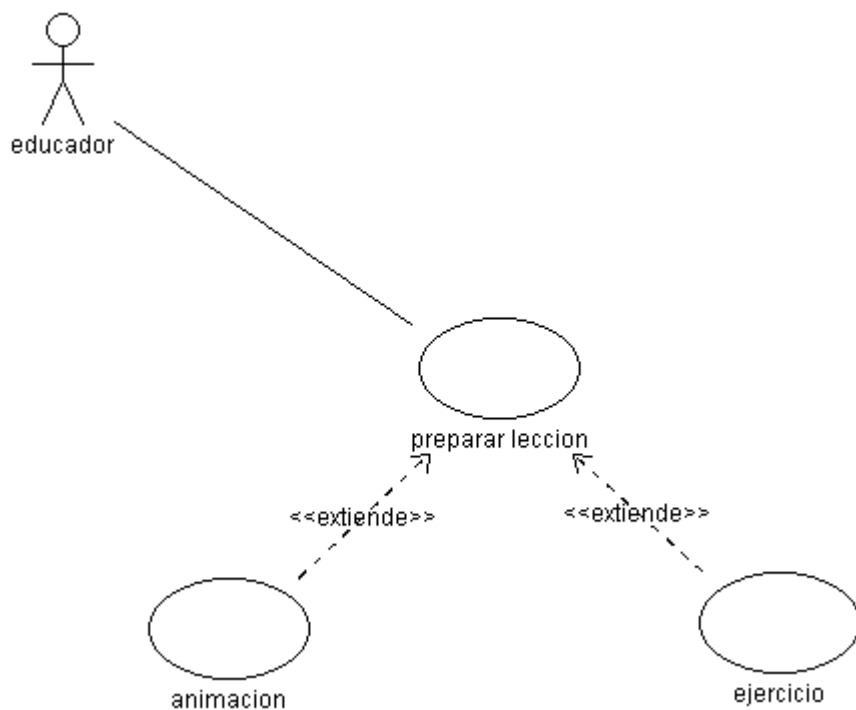
Paquete de Casos de Uso 4: Soporte gestión sonidos.



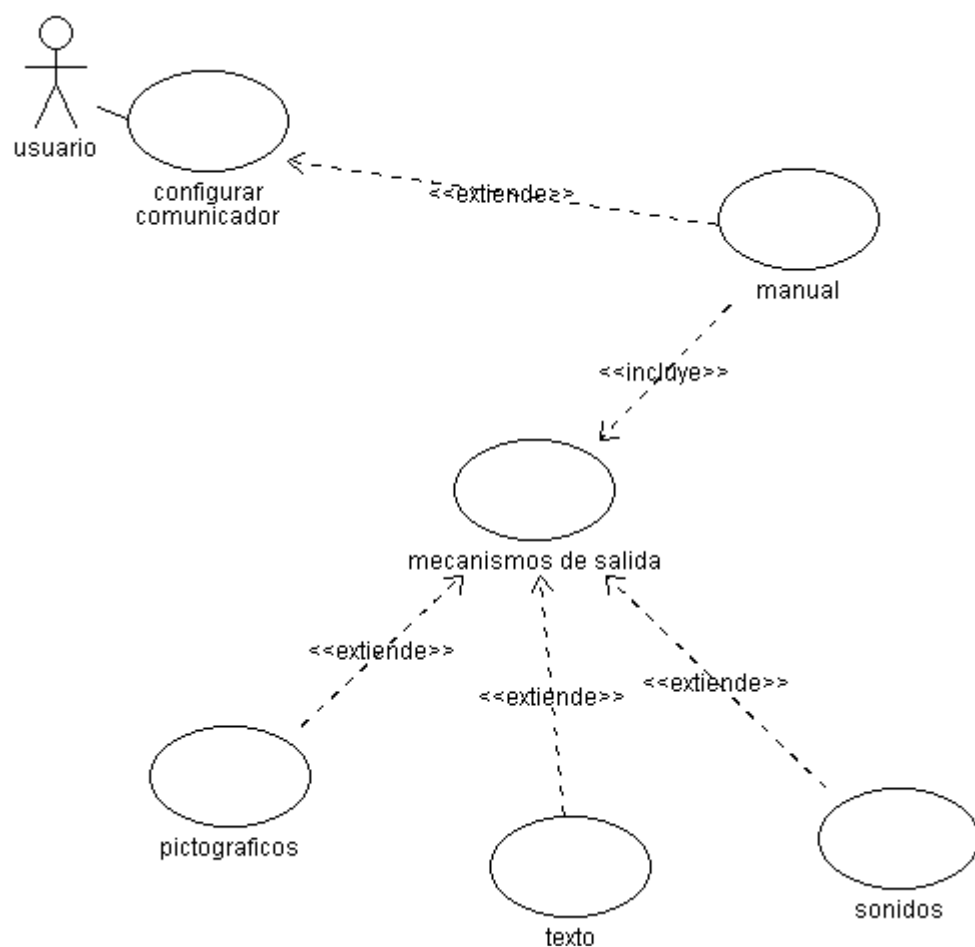
Paquete de Casos de Uso 5: Soportar la biblioteca de alumnos.



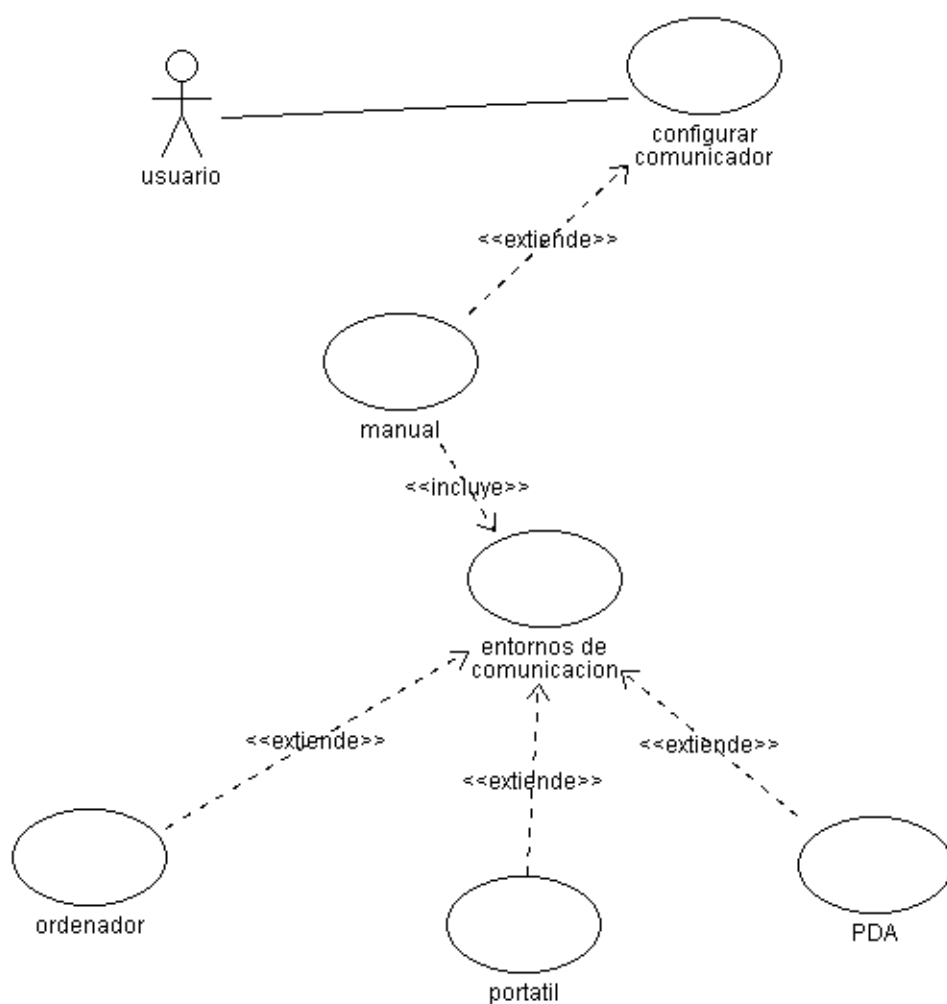
Paquete de Casos de Uso 6: Soportar el seguimiento de alumnos.



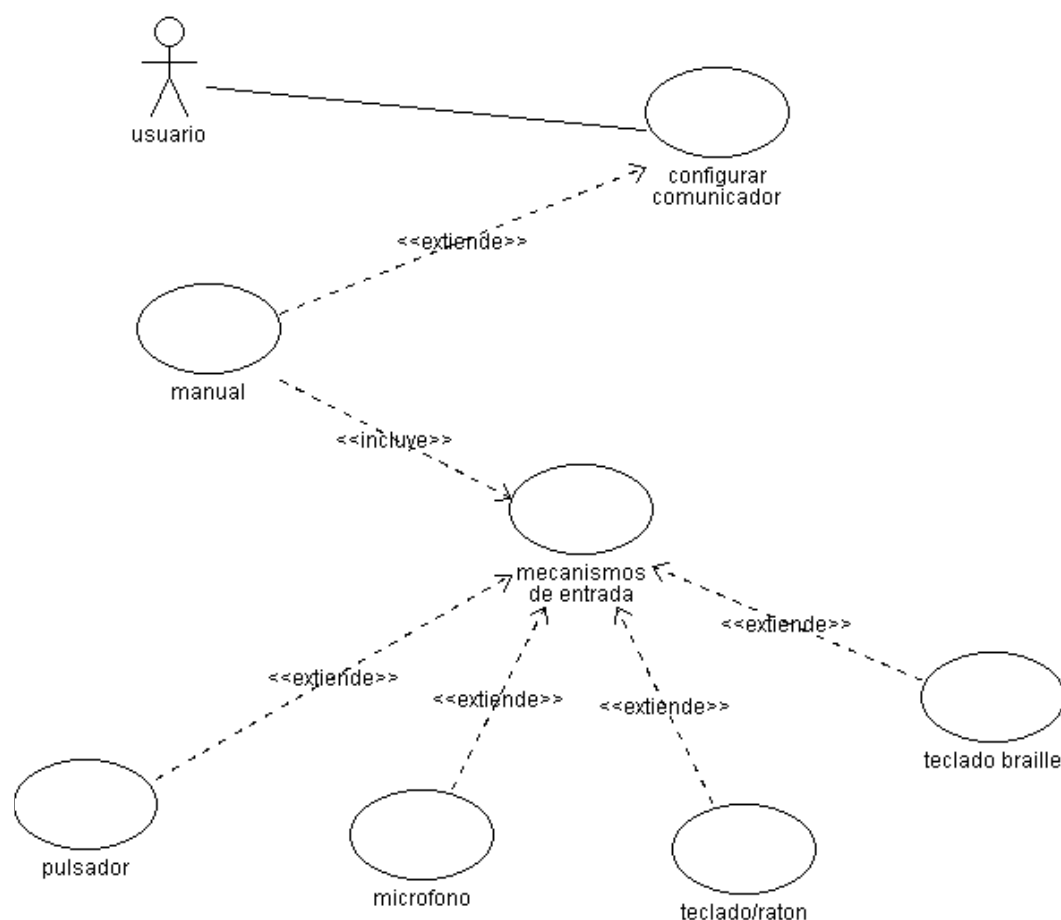
Paquete de Casos de Uso 7: Soporte de preparación de lecciones.



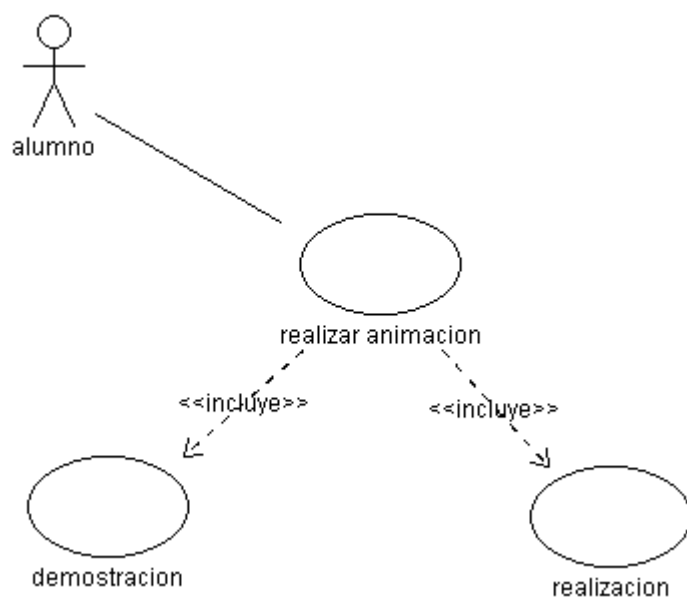
Paquete de Casos de Uso 8: Soportar distintos mecanismos de salida.



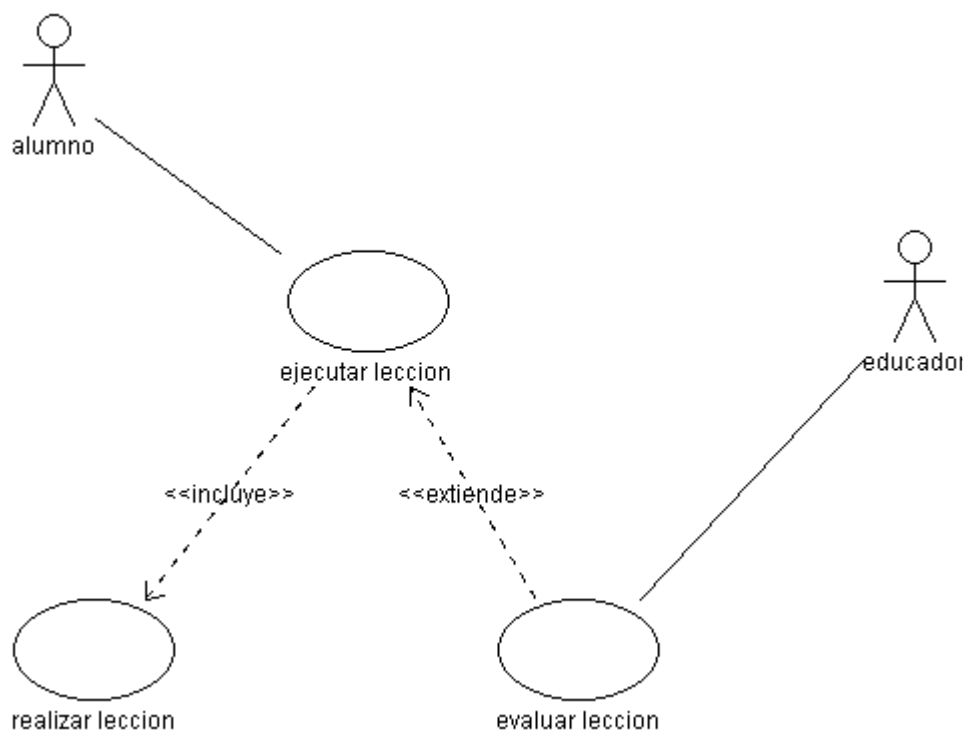
Paquete de Casos de Uso 9: Soportar distintos entornos de trabajo.



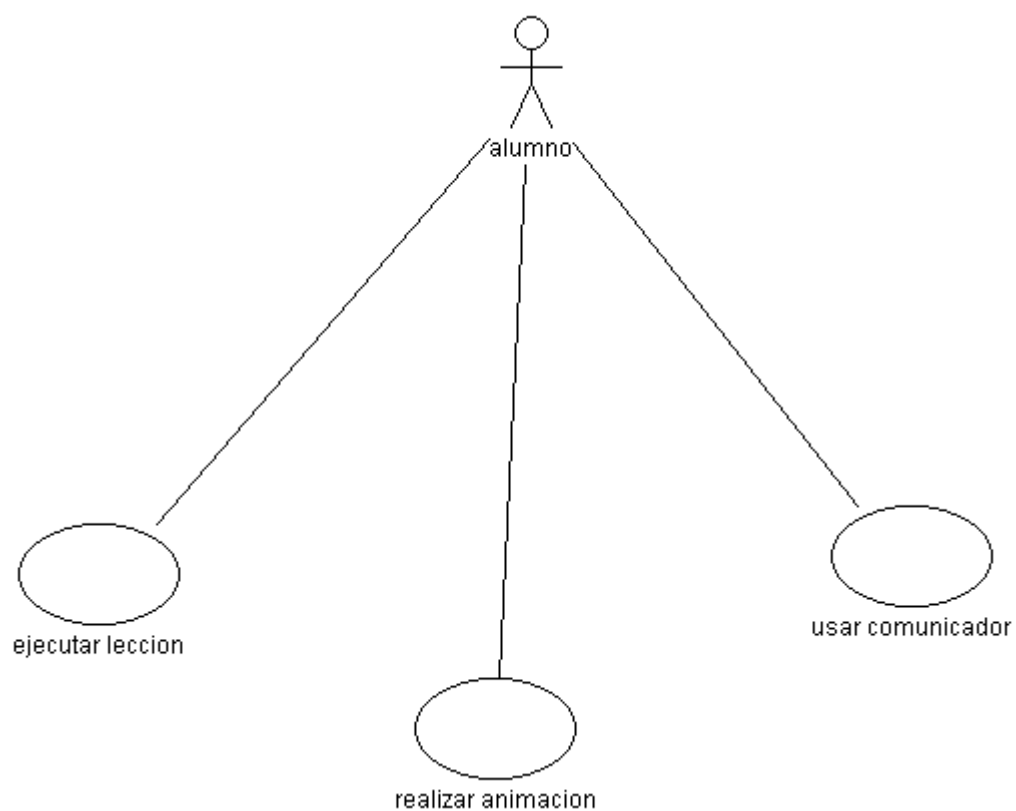
Paquete de Casos de Uso 10: Soportar distintos mecanismos de entrada.



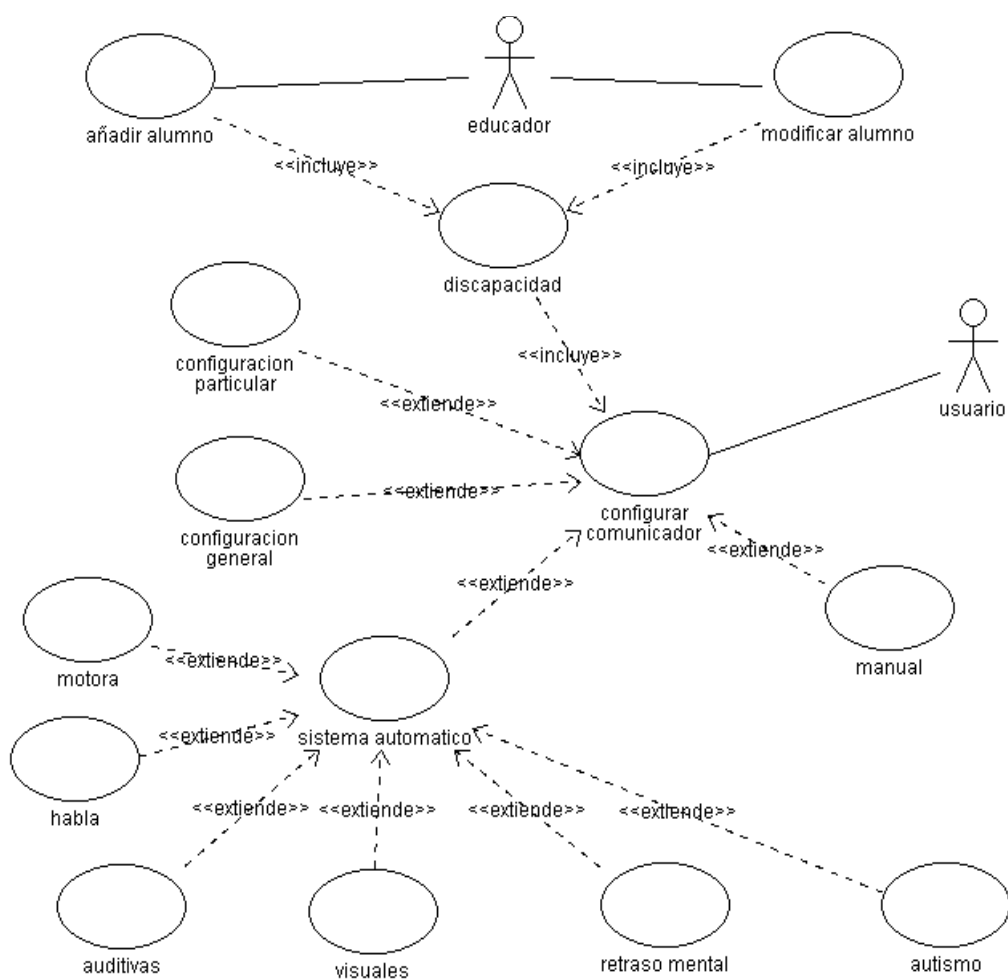
Paquete de Casos de Uso 11: Soporte realización de animaciones.



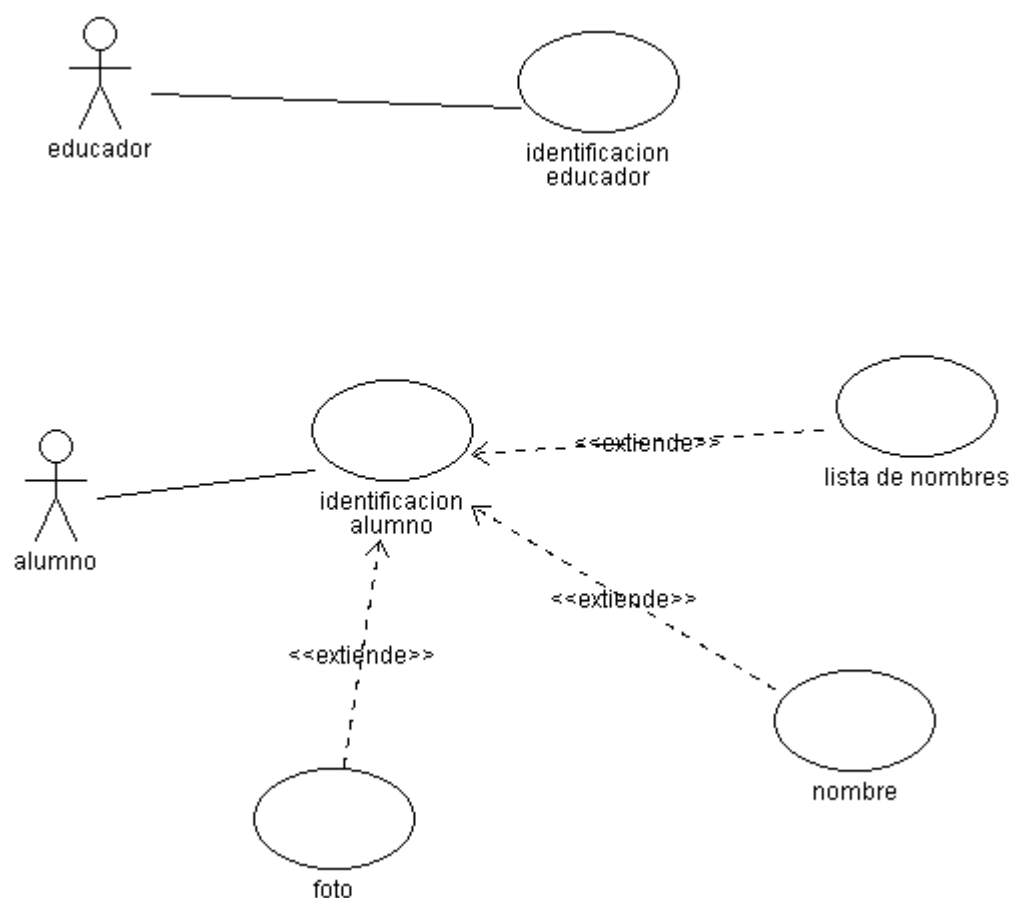
Paquete de Casos de Uso 12: Soporte realización de ejercicios.



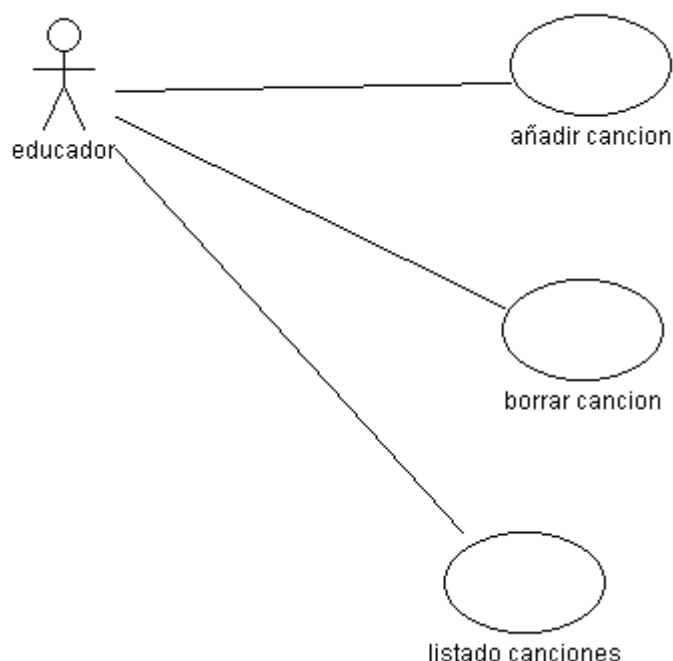
Paquete de Casos de Uso 13: Soporte de soluciones para el aprendizaje



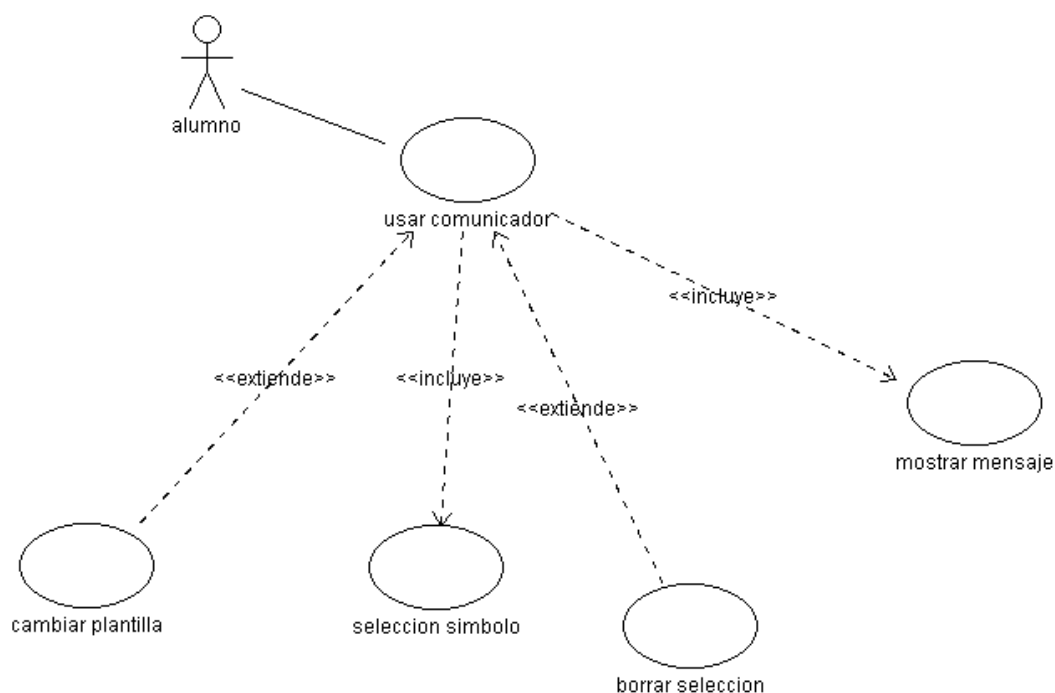
Paquete de Casos de Uso 14: Soportar distintas discapacidades.



Paquete de Casos de Uso 15: Soportar multiusuarios.



Paquete de Casos de Uso 16: Soporte la gestión de canciones.



Paquete de Casos de Uso 17: Soportar la comunicación.

4.3.3. DESCRIPCION DE LOS CASOS DE USO

A continuación se muestran la descripción de los casos de uso vistos anteriormente.

RF- 1	IDENTIFICACION ALUMNO	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	permitir identificación	
Descripción	El sistema deberá permitir al alumno identificarse mediante la forma que pueda debido a su discapacidad.	
Precondición		
Secuencia Normal	Paso	Acción
	1	El alumno pide identificarse
	2	El sistema muestra al alumno las distintas opciones para identificarse.
	3	El alumno introduce la opción elegida.
	4	Se realiza el Punto de Extensión: tipo de identificación
Postcondición	El alumno se ha identificado	
Excepciones	Paso	Acción
	2	Si el sistema no puede mostrar las opciones para identificarse, muestra un mensaje de error y el caso de uso finaliza
Comentarios		

RF- 2	NOMBRE <u><< extend RF-1>></u>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	Nombre	
Descripción	El sistema deberá permitir al alumno identificarse introduciendo su nombre	
Precondición		
Secuencia Normal	Paso	Acción
	1	El alumno pide identificarse escribiendo su nombre
	2	El sistema pide al alumno que introduzca su nombre y apellidos
	3	El alumno introduce sus datos
	4	El sistema confirma la identificación
	5	El sistema adapta la configuración del comunicador al alumno identificado
Postcondición	El alumno se ha identificado por su nombre	
Excepciones	Paso	Acción
	4.a	Si los datos introducidos no están en la base de datos, el sistema muestra un mensaje de error, y vuelve a realizarse el caso de uso
	4.b	Si ya hay un alumno que esta identificado en el sistema con ese nombre y apellidos en ese momento, muestra un mensaje de error y el caso de uso finaliza
	5.a	Si el sistema no puede acceder a la base de datos para configurar el comunicador al alumno introducido, entonces se muestra un mensaje de error y el caso de uso finaliza.
	5.b	Si el sistema no puede adaptar el comunicador al alumno identificado, el sistema muestra un mensaje de error y el caso de uso finaliza
Comentarios		

RF- 3	FOTO <u><< extend RF-1>></u>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	Foto	
Descripción	El sistema deberá permitir al alumno identificarse eligiendo su foto	
Precondición		
Secuencia Normal	Paso	Acción
	1	El alumno pide identificarse mediante su foto
	2	El sistema muestra las distintas fotos que hay en la base de datos
	3	El alumno elige su foto
	4	El sistema confirma la identificación
	5	El sistema adapta la configuración del comunicador al alumno identificado
Postcondición	El alumno se ha identificado mediante su foto	
Excepciones	Paso	Acción
	4	Si ya hay un alumno que esta identificado en el sistema con esa foto en ese momento, muestra un mensaje de error y el caso de uso finaliza
	5.a	Si el sistema no puede acceder a la base de datos para configurar el comunicador al alumno introducido, entonces se muestra un mensaje de error y el caso de uso finaliza.
	5.b	Si el sistema no puede adaptar el comunicador al alumno identificado, el sistema muestra un mensaje de error y el caso de uso finaliza
Comentarios		

RF- 4	LISTA DE NOMBRES << <u>extend RF-1</u> >>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	Lista de nombres	
Descripción	El sistema deberá permitir al alumno identificarse eligiendo su nombre de la lista	
Precondición		
Secuencia Normal	Paso	Acción
	1	El alumno pide identificarse seleccionando su nombre de la lista
	2	El sistema muestra la lista con los nombres de los alumnos
	3	El usuario selecciona el nombre
	4	El sistema confirma la identificación
	5	El sistema adapta la configuración del comunicador al alumno identificado
Postcondición	El alumno se ha identificado mediante una lista de nombres	
Excepciones	Paso	Acción
	2	Si la lista de nombres no se puede cargar, el sistema muestra un mensaje informando de ello, y el caso de uso finaliza
	4	Si ya hay un alumno que esta identificado en el sistema con ese nombre y apellidos en ese momento, muestra un mensaje de error y el caso de uso finaliza
	5.a	Si el sistema no puede acceder a la base de datos para configurar el comunicador al alumno introducido, entonces se muestra un mensaje de error y el caso de uso finaliza.
	5.b	Si el sistema no puede adaptar el comunicador al alumno identificado, el sistema muestra un mensaje de error y el caso de uso finaliza
Comentarios		

RF- 5	IDENTIFICAR EDUCADOR	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	permitir identificación	
Descripción	El sistema debe permitir al educador identificarse	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador solicita identificarse en el sistema
	2	El sistema pide el nombre y el password del educador
	3	El educador introduce el nombre y el password
	4	El sistema confirma que se ha introducido los datos correctamente
Postcondición	El educador se ha identificado	
Excepciones	Paso	Acción
	4	si el educador introduce los datos incorrectamente muestra un mensaje de error y vuelve a pedir los datos
Comentarios		

RF- 6	SEGUIMIENTO ALUMNO	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	Seguimiento	
Descripción	El educador desea conocer la evolución de los alumnos	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador pide al sistema información del seguimiento de los alumnos
	2	Pt. Extensión: Tipo de información
Postcondición	El educador ha comprobado la evolución	
Excepciones	Paso	Acción
Comentarios		

RF- 7	TODOS <u><<extend RF-6>></u>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	Seguimiento	
Descripción	El educador desea conocer la evolución de todos los alumnos	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador pide información de todos los alumnos al sistema
	2	El sistema muestra una lista con todos los alumnos y su evolución, los ejercicios realizados, las animaciones hechas, etc..
Postcondición	El educador ha comprobado la evolución de todos los alumnos	
Excepciones	Paso	Acción
	2	Si el sistema no puede acceder a la base de datos de los alumnos, se muestra un mensaje de error y a continuación el caso de uso finaliza
Comentarios		

RF- 8	PARTICULAR <u><<extend RF-6>></u>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	Seguimiento	
Descripción	El educador desea conocer la evolución de un alumno en particular	
Precondición	El educador quiere seguir la evolución de un alumno	
Secuencia Normal	Paso	Acción
	1	El educador introduce el nombre y apellidos del alumno
	2	El sistema muestra una lista del alumno con toda su información y su evolución, así como los ejercicios realizados, las animaciones hechas, etc..
Postcondición	El educador ha comprobado la evolución de todos los alumnos	
Excepciones	Paso	Acción
	2.a	Si los datos del alumno no están en el sistema, se muestra un mensaje advirtiéndolo de esta irregularidad y a continuación el caso de uso vuelve a empezar
	2.b	Si el sistema no puede acceder a la base de datos de los alumnos, se muestra un mensaje de error y a continuación el caso de uso finaliza
Comentarios		

RF- 9	PREPARAR LECCION	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	solucionar problemas de aprendizaje	
Descripción	El sistema debe permitir al educador preparar una lección para un alumno determinado	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador pide al sistema preparar una lección
	2	El sistema pide los datos del alumno
	3	El educador introduce los datos del alumno
	4	El sistema pide el tipo de lección que se desea realizar
	5	Se realiza el punto de extensión: tipo lección
Postcondición		
Excepciones	Paso	Acción
	2	Si el educador introduce los datos del alumno incorrectamente, se muestra un mensaje de error y a continuación el sistema vuelve a pedir los datos
Comentarios		

RF- 10	ANIMACION <u><< extend RF-9>></u>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	preparar animación	
Descripción	El sistema debe permitir al educador preparar una animación	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador debe introducir la cantidad de veces que se desea que se repita el ejercicio
	2	El educador introduce el tiempo de latencia
	3	El educador introduce las palabras clave de la lista, la imagen y la canción
	4	El sistema comprueba que el numero de veces es correcto
	5	El sistema comprueba que el tiempo de latencia es correcto
	6	El sistema comprueba que se ha seleccionado al menos una palabra clave, una imagen y una canción
Postcondición	La programación del ejercicio se ha realizado con satisfacción	
Excepciones	Paso	Acción
	1	Si el numero de veces es incorrecto, se indicara que mostrara un mensaje de error y se pedirá otra vez el numero de veces
	2	Si el tiempo de latencia es incorrecto, se indicara que mostrara un mensaje de error y se pedirá otra vez el tiempo de latencia
	3	Si el numero de veces es incorrecto, se indicara que mostrara un mensaje de error y se pedirá otra vez el numero de veces
	4	Si no se selecciona una palabra clave al menos, se indicara que mostrara un mensaje de error y se pedirá introducir la palabra clave
	5	En el caso de que la palabra clave seleccionada no tenga ninguna imagen asociada, se indicara que debe ser añadida en la biblioteca.
Comentarios		

RF- 11	EJERCICIOS <u><<extend RF-9>></u>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	preparar lección	
Descripción	El sistema debe permitir al educador preparar un ejercicio	
Precondición	El educador desea preparar el ejercicio	
Secuencia Normal	Paso	Acción
	1	El educador debe introducir la cantidad de veces que se desea que se repita el ejercicio
	2	El educador introduce el ejercicio con el que desea trabajar
	3	El sistema comprueba que el numero de veces introducido es correcto
	4	El sistema comprueba que el ejercicio introducido es correcto
Postcondición	La programación del ejercicio se ha realizado con satisfacción	
Excepciones	Paso	Acción
	3	Si el numero de veces es incorrecto, se indicara que mostrara un mensaje de error y se pedirá otra vez el numero de veces
	4	Si el ejercicio introducido es incorrecto, se indicara que mostrara un mensaje de error y se pedirá otra vez la introducción de otro ejercicio
Comentarios		

RF- 12	DISCAPACIDAD	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	Permitir manejo del sistema	
Descripción	El educador introduce la discapacidad	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador introduce en el sistema el tipo de discapacidad del alumno especificado
	2	El sistema actualiza el tipo de discapacidad del alumno especificado
	3	Se realiza el caso de uso <<RF-13>> Configurar comunicador
Postcondición	El educador ha introducido la discapacidad del alumno	
Excepciones	Paso	Acción
	2	Si el sistema no puede actualizar el tipo de discapacidad, muestra un mensaje de error y a continuación el caso de uso finaliza
Comentarios		

RF- 13	CONFIGURAR COMUNICADOR	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	configuración	
Descripción	El sistema debe permitir al educador configurar el comunicador a las necesidades del alumno, o configurar el comunicador para un uso genérico	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador pide configurar el comunicador
	2	El sistema le pregunta al educador por el tipo de configuración y el modo de configuración.
	3	Pt. De extensión: Tipo de configuración
	4	Pt. De extensión: Modo de configuración
Postcondición	Se ha configurado el comunicador	
Excepciones	Paso	Acción
	1	Si el sistema en ese momento no puede configurar el comunicador, muestra un mensaje de error y a continuación el caso de uso finaliza
Comentarios		

RF- 14	SISTEMA AUTOMATICO <<extend RF-13>>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	automático	
Descripción	El sistema permite al educador configurar el comunicador de forma automática eligiendo el tipo de discapacidad y configurando el comunicador con unos mecanismos estándar para esa discapacidad	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador pide configurar el comunicador de forma automática
	2	El sistema muestra un mensaje al educador advirtiéndole de que al elegir un tipo de configuración automático, esta se hará de acuerdo a unos parámetros estándar
	3	Pt. De extensión: Configuración por discapacidad
Postcondición	Se ha configurado el comunicador automáticamente	
Excepciones	Paso	Acción
	2	Si el educador no quiere en este momento seguir con la configuración automática, el sistema muestra un mensaje y a continuación el caso de uso finaliza
Comentarios		

RF- 15	MANUAL <u><<extend RF-13>></u>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	manual	
Descripción	El sistema permite al educador configurar el comunicador de forma manual eligiendo el mismo todos los mecanismos disponibles	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador pide configurar el comunicador de forma manual
	2	Se realiza el caso de uso <u><<RF-16>></u> Entornos de Comunicación
	3	Se realiza el caso de uso <u><<RF-17>></u> Mecanismos de entrada
	4	Se realiza el caso de uso <u><<RF-18>></u> Mecanismos de salida
Postcondición	Se ha configurado el comunicador manualmente	
Excepciones	Paso	Acción
	1	Si el sistema no puede configurar el comunicador de forma manual en ese instante, muestra un mensaje de error y el caso de uso finaliza
Comentarios		

RF- 16	CONFIGURACION GENERAL <u><<extend RF-13>></u>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	configuración	
Descripción	El sistema permite al educador configurar el comunicador de forma general para cualquier usuario	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador pide configurar el comunicador de forma genérica
	2	El sistema informa al educador de que la configuración será usada como configuración por defecto para cualquier alumno no identificado en el sistema
	3	El educador esta de acuerdo con guardar dicha configuración para uso genérico
	4	El sistema guarda la configuración seleccionada para el uso genérico del comunicador
Postcondición		
Excepciones	Paso	Acción
	1	Si el sistema no puede configurar el comunicador de forma genérica en ese instante, muestra un mensaje de error y el caso de uso finaliza
Comentarios		

RF- 17	CONFIGURACION PARTICULAR <u><<extend RF-13>></u>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	configuración	
Descripción	El sistema permite al educador configurar el comunicador de forma particular para un determinado usuario	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador pide configurar el comunicador de forma particular para el alumno elegido anteriormente
	2	El sistema informa al educador de que la configuración será usada como configuración para ese alumno
	3	El educador esta de acuerdo con guardar dicha configuración para ese alumno
	4	El sistema guarda la configuración seleccionada en la base de datos para ese alumno
Postcondición		
Excepciones	Paso	Acción
	4	Si el sistema no puede configurar el comunicador de forma particular en ese instante, muestra un mensaje de error y el caso de uso finaliza
Comentarios		

RF- 18	ENTORNOS DE COMUNICACION	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	soportar distintos entornos	
Descripción	El sistema deberá permitir al educador configurar los entornos de trabajo que desee para utilizar el comunicador	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador pide al sistema configurar un entorno
	2	El sistema pregunta al educador que tipo de dispositivo quiere configurar
	3	El educador introduce el tipo de dispositivo que desea configurar
	4	Pt. De Extensión: Dispositivo
Postcondición		
Excepciones	Paso	Acción
	1	Si el sistema no puede configurar el dispositivo en ese instante, muestra un mensaje de error y el caso de uso finaliza
Comentarios		

RF- 19	ORDENADOR <u><<extend RF-18>></u>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	ordenador	
Descripción	El sistema debe permitir al alumno seleccionar como dispositivo un ordenador	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador pide al sistema configurar un ordenador
	2	El sistema pide los datos que configuran el ordenador
	3	El educador introduce los datos para configurar el ordenador
	4	El sistema confirma la configuración del ordenador
Postcondición	El dispositivo ordenador ha sido seleccionado y configurado	
Excepciones	Paso	Acción
	1	Si el sistema no puede configurar el ordenador en ese instante, muestra un mensaje de error y el caso de uso finaliza
	4	Si alguno de los datos que configuran el ordenador es incorrecto, el sistema muestra un mensaje advirtiendo de esta circunstancia y el caso de uso vuelve a iniciarse
Comentarios		

RF- 20	PORTATIL <u><<extend RF-18>></u>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	portátil	
Descripción	El sistema debe permitir al alumno seleccionar como dispositivo un Portátil	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador pide al sistema configurar un Portátil
	2	El sistema pide los datos que configuran el Portátil
	3	El educador introduce los datos para configurar el Portátil
	4	El sistema confirma la configuración del Portátil
Postcondición	El dispositivo Portátil ha sido seleccionado y configurado	
Excepciones	Paso	Acción
	1	Si el sistema no puede configurar el Portátil en ese instante, muestra un mensaje de error y el caso de uso finaliza
	4	Si alguno de los datos que configuran el portátil es incorrecto, el sistema muestra un mensaje advirtiéndolo de esta circunstancia y el caso de uso vuelve a iniciarse
Comentarios		

RF- 21	PDA <u><<extend RF-18>></u>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	PDA	
Descripción	El sistema debe permitir al alumno seleccionar como dispositivo un PDA	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador pide al sistema configurar un PDA
	2	El sistema pide los datos que configuran el PDA
	3	El educador introduce los datos para configurar el PDA
	4	El sistema confirma la configuración del PDA
Postcondición	El dispositivo PDA ha sido seleccionado y configurado	
Excepciones	Paso	Acción
	1	Si el sistema no puede configurar el PDA en ese instante, muestra un mensaje de error y el caso de uso finaliza
	4	Si alguno de los datos que configuran el PDA es incorrecto, el sistema muestra un mensaje advirtiéndolo de esta circunstancia y el caso de uso vuelve a iniciarse
Comentarios		

RF- 22	MECANISMOS DE ENTRADA	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	soportar distintos mecanismos de entrada	
Descripción	El sistema deberá permitir al educador configurar los mecanismos de entrada que desee	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador pide al sistema configurar un mecanismo de entrada
	2	El sistema pregunta al educador que tipo de dispositivo de entrada quiere configurar
	3	El educador introduce el tipo de dispositivo de entrada que desea configurar
	4	Pt. De Extensión: Dispositivo de entrada
Postcondición	El sistema ha configurado un mecanismo de entrada	
Excepciones	Paso	Acción
	1	Si el sistema no puede configurar el dispositivo de entrada en ese instante, muestra un mensaje de error y el caso de uso finaliza
Comentarios		

RF- 23	MICROFONO <u><<extend RF-22>></u>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	micrófono	
Descripción	El sistema debe permitir al alumno seleccionar como dispositivo de entrada un micrófono	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador pide al sistema configurar un micrófono
	2	El sistema pide los datos que configuran el micrófono
	3	El educador introduce los datos para configurar el micrófono
	4	El sistema confirma la configuración del micrófono como mecanismo de entrada
Postcondición	El dispositivo de entrada micrófono ha sido seleccionado y configurado	
Excepciones	Paso	Acción
	1	Si el sistema no puede configurar el micrófono en ese instante, muestra un mensaje de error y el caso de uso finaliza
	4	Si alguno de los datos que configuran el micrófono es incorrecto, el sistema muestra un mensaje advirtiendo de esta circunstancia y el caso de uso vuelve a iniciarse
Comentarios		

RF- 24	PULSADOR <u><<extend RF-22>></u>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	pulsador	
Descripción	El sistema debe permitir al alumno seleccionar como dispositivo de entrada un pulsador	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador pide al sistema configurar un pulsador
	2	El sistema pide los datos que configuran el pulsador
	3	El educador introduce los datos para configurar el pulsador
	4	El sistema confirma la configuración del pulsador como mecanismo de entrada
Postcondición	El dispositivo de entrada pulsador ha sido seleccionado y configurado	
Excepciones	Paso	Acción
	1	Si el sistema no puede configurar el pulsador en ese instante, muestra un mensaje de error y el caso de uso finaliza
	4	Si alguno de los datos que configuran el pulsador es incorrecto, el sistema muestra un mensaje advirtiendo de esta circunstancia y el caso de uso vuelve a iniciarse
Comentarios		

RF- 25	TECLADO Y RATON <u><<extend RF-22>></u>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	Teclado y ratón	
Descripción	El sistema debe permitir al alumno seleccionar como dispositivo de entrada un teclado y ratón	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador pide al sistema configurar un teclado y ratón
	2	El sistema pide los datos que configuran el teclado y ratón
	3	El educador introduce los datos para configurar el teclado y ratón
	4	El sistema confirma la configuración del teclado y ratón como mecanismo de entrada
Postcondición	El dispositivo de entrada teclado y ratón ha sido seleccionado y configurado	
Excepciones	Paso	Acción
	1	Si el sistema no puede configurar el teclado y ratón en ese instante, muestra un mensaje de error y el caso de uso finaliza
	4	Si alguno de los datos que configuran el teclado y ratón es incorrecto, el sistema muestra un mensaje advirtiéndolo de esta circunstancia y el caso de uso vuelve a iniciarse
Comentarios		

RF- 26	TECLADO BRAILLE <u><<extend RF-22>></u>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	Teclado braille	
Descripción	El sistema debe permitir al alumno seleccionar como dispositivo de entrada un teclado braille	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador pide al sistema configurar un teclado braille
	2	El sistema pide los datos que configuran el teclado braille
	3	El educador introduce los datos para configurar el teclado braille
	4	El sistema confirma la configuración del teclado braille como mecanismo de entrada
Postcondición	El dispositivo de entrada teclado braille ha sido seleccionado y configurado	
Excepciones	Paso	Acción
	1	Si el sistema no puede configurar el teclado braille en ese instante, muestra un mensaje de error y el caso de uso finaliza
	4	Si alguno de los datos que configuran el teclado braille es incorrecto, el sistema muestra un mensaje advirtiendo de esta circunstancia y el caso de uso vuelve a iniciarse
Comentarios		

RF- 27	MECANISMOS DE SALIDA	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	soportar distintos mecanismos de salida	
Descripción	El sistema deberá permitir al educador configurar los mecanismos de salida que desee para utilizar el comunicador	
Precondición		
Secuencia Normal	Paso	Acción
	1	El sistema pregunta al educador que tipo de mecanismo de salida desea configurar
	2	El educador introduce el tipo de mecanismo de salida que desea configurar
	3	Se realiza el punto de extensión: mecanismo salida
Postcondición	El educador ha configurado un mecanismo de salida	
Excepciones	Paso	Acción
	1	Si el sistema no puede configurar el mecanismo de salida en ese instante, muestra un mensaje de error y el caso de uso finaliza
Comentarios		

RF- 28	PICTOGRAFICOS <u><<extend RF-27>></u>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	pictográficos	
Descripción	El sistema debe permitir al educador seleccionar el mecanismo de salida por pictogramas	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador pide al sistema configurar la salida por pictogramas
	2	El sistema pide los datos que configuran los pictogramas
	3	El educador introduce los datos para configurar los pictogramas
	4	El sistema confirma la configuración de los pictogramas como mecanismo de salida
Postcondición	El dispositivo salida por pictogramas ha sido seleccionado y configurado	
Excepciones	Paso	Acción
	1	Si el sistema no puede configurar los pictogramas en ese instante, muestra un mensaje de error y el caso de uso finaliza
	4	Si alguno de los datos que configuran los pictogramas es incorrecto, el sistema muestra un mensaje advirtiendo de esta circunstancia y el caso de uso vuelve a iniciarse
Comentarios		

RF- 29	SONIDOS <u><<extend RF-27>></u>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	sonidos	
Descripción	El sistema debe permitir al educador seleccionar el mecanismo de salida por sonidos	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador pide al sistema configurar la salida por sonidos
	2	El sistema pide los datos que configuran los sonidos
	3	El educador introduce los datos para configurar los sonidos
	4	El sistema confirma la configuración de los sonidos como mecanismo de salida
Postcondición	El dispositivo salida por sonidos ha sido seleccionado y configurado	
Excepciones	Paso	Acción
	1	Si el sistema no puede configurar los sonidos en ese instante, muestra un mensaje de error y el caso de uso finaliza
	4	Si alguno de los datos que configuran los sonidos es incorrecto, el sistema muestra un mensaje advirtiéndolo de esta circunstancia y el caso de uso vuelve a iniciarse
Comentarios		

RF- 30	TEXTO <u><<extend RF-27>></u>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	texto	
Descripción	El sistema debe permitir al educador seleccionar el mecanismo de salida por texto	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador pide al sistema configurar la salida por texto
	2	El sistema pide los datos que configuran los texto
	3	El educador introduce los datos para configurar los texto
	4	El sistema confirma la configuración de texto como mecanismo de salida
Postcondición	El dispositivo salida por texto ha sido seleccionado y configurado	
Excepciones	Paso	Acción
	1	Si el sistema no puede configurar texto en ese instante, muestra un mensaje de error y el caso de uso finaliza
	4	Si alguno de los datos que configuran el mecanismo de salida por texto es incorrecto, el sistema muestra un mensaje advirtiendo de esta circunstancia y el caso de uso vuelve a iniciarse
Comentarios		

RF- 31	REALIZAR ANIMACION	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	Realizar animaciones	
Descripción	El sistema debe permitir al alumno realizar la animación que tenia programada el educador para el	
Precondición		
Secuencia Normal	Paso	Acción
	1	El alumno selecciona realizar la animación
	2	El sistema comprueba la configuración del comunicador
	3	El sistema recoge los datos para la animación.
	4	Se realiza el caso de uso << RF-32>> Demostración
	5	Se realiza el caso de uso << RF-33>> Realización
Postcondición	La animación se ha realizado	
Excepciones	Paso	Acción
	1	Si el sistema no puede realizar la animación en ese instante, se muestra un mensaje de error y a continuación el caso de uso finaliza
	2	Si el sistema no puede cargar la configuración del comunicador para ese determinado alumno, se muestra un mensaje advirtiéndolo y el caso de uso finaliza.
Comentarios		

RF- 32	DEMOSTRACION	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	demostración	
Descripción	El sistema debe permitir animar la imagen y reproducir la canción que el educador ha preparado para el alumno identificado	
Precondición		
Secuencia Normal	Paso	Acción
	1	El sistema reproduce la canción seleccionada
	2	El sistema reproduce el texto seleccionado
	3	El sistema reproduce la imagen mientras dura la canción
Postcondición	La demostración se ha realizado con éxito	
Excepciones	Paso	Acción
	1	Si el sistema no puede ejecutar la canción seleccionada, se muestra un mensaje de error y el caso de uso finaliza
	2	Si el sistema no puede ejecutar el texto seleccionado, se muestra un mensaje de error y el caso de uso finaliza
	3	Si el sistema no puede ejecutar la imagen seleccionada, se muestra un mensaje de error y el caso de uso finaliza
Comentarios		

RF- 33	REALIZACION	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	realización	
Descripción	El sistema debe permitir animar la imagen y reproducir la canción que el educador ha preparado para el alumno identificado, mediante el mecanismo de entrada que se haya configurado previamente y el entorno de trabajo.	
Precondición		
Secuencia Normal	Paso	Acción
	1	El sistema configura la animación de acuerdo a la configuración guardada para ese alumno
	2	El sistema recoge la emisión del mecanismo de entrada del alumno
	3	El sistema reproduce la canción seleccionada
	4	El sistema reproduce el texto seleccionado
	5	El sistema reproduce la imagen mientras dura la canción
	6	El sistema guarda la animación como realizada para ese alumno
Postcondición	La realización de la animación por el alumno se ha realizado correctamente	
Excepciones	Paso	Acción
	3	Si el sistema no puede ejecutar la canción seleccionada, se muestra un mensaje de error y el caso de uso finaliza
	4	Si el sistema no puede ejecutar el texto seleccionado, se muestra un mensaje de error y el caso de uso finaliza
	5	Si el sistema no puede ejecutar la imagen seleccionada, se muestra un mensaje de error y el caso de uso finaliza
	5	Si el alumno deja de utilizar el mecanismo de entrada, el caso de uso finaliza

RF- 34	REALIZAR LECCION	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	Realizar lección	
Descripción	El sistema debe permitir al alumno realizar la lección que tenia programada el educador para el	
Precondición		
Secuencia Normal	Paso	Acción
	1	El alumno selecciona realizar la lección
	2	El sistema comprueba la configuración del comunicador
	3	El sistema recoge los datos para la lección.
	4	Se realiza el caso de uso << RF-35>> Ejecutar Ejercicio
	5	Se realiza el punto de extensión: Evaluación
Postcondición	La animación se ha realizado	
Excepciones	Paso	Acción
	1	Si el sistema no puede realizar la lección en ese instante, se muestra un mensaje de error y a continuación el caso de uso finaliza
	2	Si el sistema no puede cargar la configuración del comunicador para ese determinado alumno, se muestra un mensaje advirtiéndolo y el caso de uso finaliza.
Comentarios		

RF- 35	EJECUTAR EJERCICIO	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	Realizar ejercicios	
Descripción	El sistema debe permitir realizar el ejercicio que el educador ha preparado para el alumno identificado, mediante el mecanismo de entrada que se haya configurado previamente y el entorno de trabajo.	
Precondición		
Secuencia Normal	Paso	Acción
	1	El sistema configura el ejercicio de acuerdo a la configuración guardada para ese alumno
	2	El sistema recoge la emisión del mecanismo de entrada del alumno
	3	El sistema muestra el ejercicio seleccionado
	4	El alumno introduce su resolución del ejercicio
	5	El sistema guarda la resolución del ejercicio del alumno
Postcondición		
Excepciones	Paso	Acción
	3	Si el sistema no puede configurar el ejercicio de acuerdo a la configuración guardada, se muestra un mensaje advirtiendo de esta situación y el caso de uso finaliza
	5	Si el sistema no puede guardar la respuesta del alumno, se muestra un mensaje de error y a continuación el caso de uso finaliza
Comentarios		

RF- 36	MOTORA <u><<extend RF-14>></u>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	motora	
Descripción	El sistema permite al educador configurar el comunicador de forma automática eligiendo que el tipo de discapacidad sea motora y configurando el comunicador con unos mecanismos estándar para la discapacidad motora	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador introduce que le tipo de discapacidad es motora
	2	El sistema configura como entorno de comunicación un ordenador
	3	El sistema configura como mecanismo de entrada un micrófono
	4	El sistema configura como mecanismo de salida un sistema pictográfico
	5	El sistema confirma la configuración
Postcondición	Se ha configurado el comunicador automáticamente para una discapacidad motora	
Excepciones	Paso	Acción
	2	Si el sistema no puede configurar el entorno de comunicación en ese momento, se muestra un mensaje y a continuación el caso de uso finaliza
	3	Si el sistema no puede configurar el mecanismo de entrada en ese momento, se muestra un mensaje y a continuación el caso de uso finaliza
	4	Si el sistema no puede configurar el mecanismo de salida en ese momento, se muestra un mensaje y a continuación el caso de uso finaliza
	5	Si el sistema no puede guardar al configuración en ese instante, muestra un mensaje de error y a continuación el caso de uso finaliza
Comentarios		

RF- 37	HABLA <u><<extend RF-14>></u>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	habla	
Descripción	El sistema permite al educador configurar el comunicador de forma automática eligiendo que el tipo de discapacidad sea del habla y configurando el comunicador con unos mecanismos estándar para la discapacidad de habla	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador introduce que le tipo de discapacidad es el habla
	2	El sistema configura como entorno de comunicación un ordenador
	3	El sistema configura como mecanismo de entrada un teclado y ratón
	4	El sistema configura como mecanismo de salida un sistema pictográfico
	5	El sistema confirma la configuración
Postcondición	Se ha configurado el comunicador automáticamente para una discapacidad de habla	
Excepciones	Paso	Acción
	2	Si el sistema no puede configurar el entorno de comunicación en ese momento, se muestra un mensaje y a continuación el caso de uso finaliza
	3	Si el sistema no puede configurar el mecanismo de entrada en ese momento, se muestra un mensaje y a continuación el caso de uso finaliza
	4	Si el sistema no puede configurar el mecanismo de salida en ese momento, se muestra un mensaje y a continuación el caso de uso finaliza
	5	Si el sistema no puede guardar al configuración en ese instante, muestra un mensaje de error y a continuación el caso de uso finaliza
Comentarios		

RF- 38	AUDITIVA <u><<extend RF-14>></u>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	auditiva	
Descripción	El sistema permite al educador configurar el comunicador de forma automática eligiendo que el tipo de discapacidad sea auditiva y configurando el comunicador con unos mecanismos estándar para la discapacidad auditiva	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador introduce que le tipo de discapacidad es auditiva
	2	El sistema configura como entorno de comunicación un ordenador
	3	El sistema configura como mecanismo de entrada un teclado y ratón
	4	El sistema configura como mecanismo de salida un sistema pictográfico
	5	El sistema confirma la configuración
Postcondición	Se ha configurado el comunicador automáticamente para una discapacidad auditiva	
Excepciones	Paso	Acción
	2	Si el sistema no puede configurar el entorno de comunicación en ese momento, se muestra un mensaje y a continuación el caso de uso finaliza
	3	Si el sistema no puede configurar el mecanismo de entrada en ese momento, se muestra un mensaje y a continuación el caso de uso finaliza
	4	Si el sistema no puede configurar el mecanismo de salida en ese momento, se muestra un mensaje y a continuación el caso de uso finaliza
	5	Si el sistema no puede guardar al configuración en ese instante, muestra un mensaje de error y a continuación el caso de uso finaliza
Comentarios		

RF- 39	VISUAL <u><<extend RF-14>></u>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	visual	
Descripción	El sistema permite al educador configurar el comunicador de forma automática eligiendo que el tipo de discapacidad sea visual y configurando el comunicador con unos mecanismos estándar para la discapacidad visual	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador introduce que le tipo de discapacidad es visual
	2	El sistema configura como entorno de comunicación un ordenador
	3	El sistema configura como mecanismo de entrada un teclado braille
	4	El sistema configura como mecanismo de salida un sistema de sonidos
	5	El sistema confirma la configuración
Postcondición	Se ha configurado el comunicador automáticamente para una discapacidad visual	
Excepciones	Paso	Acción
	2	Si el sistema no puede configurar el entorno de comunicación en ese momento, se muestra un mensaje y a continuación el caso de uso finaliza
	3	Si el sistema no puede configurar el mecanismo de entrada en ese momento, se muestra un mensaje y a continuación el caso de uso finaliza
	4	Si el sistema no puede configurar el mecanismo de salida en ese momento, se muestra un mensaje y a continuación el caso de uso finaliza
	5	Si el sistema no puede guardar al configuración en ese instante, muestra un mensaje de error y a continuación el caso de uso finaliza
Comentarios		

RF- 40	RETRASO MENTAL <u><<extend RF-14>></u>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	Retraso mental	
Descripción	El sistema permite al educador configurar el comunicador de forma automática eligiendo que el tipo de discapacidad sea retraso mental y configurando el comunicador con unos mecanismos estándar para la discapacidad retraso mental	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador introduce que le tipo de discapacidad es retraso mental
	2	El sistema configura como entorno de comunicación un ordenador
	3	El sistema configura como mecanismo de entrada un pulsador
	4	El sistema configura como mecanismo de salida un sistema pictográfico
	5	El sistema confirma la configuración
Postcondición	Se ha configurado el comunicador automáticamente para una discapacidad de retraso mental	
Excepciones	Paso	Acción
	2	Si el sistema no puede configurar el entorno de comunicación en ese momento, se muestra un mensaje y a continuación el caso de uso finaliza
	3	Si el sistema no puede configurar el mecanismo de entrada en ese momento, se muestra un mensaje y a continuación el caso de uso finaliza
	4	Si el sistema no puede configurar el mecanismo de salida en ese momento, se muestra un mensaje y a continuación el caso de uso finaliza
	5	Si el sistema no puede guardar al configuración en ese instante, muestra un mensaje de error y a continuación el caso de uso finaliza
Comentarios		

RF- 41	AUTISMO <u><<extend RF-14>></u>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	autismo	
Descripción	El sistema permite al educador configurar el comunicador de forma automática eligiendo que el tipo de discapacidad sea autismo y configurando el comunicador con unos mecanismos estándar para la discapacidad de autismo	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador introduce que le tipo de discapacidad es autismo
	2	El sistema configura como entorno de comunicación un ordenador
	3	El sistema configura como mecanismo de entrada un teclado y ratón
	4	El sistema configura como mecanismo de salida un sistema pictográfico
	5	El sistema confirma la configuración
Postcondición	Se ha configurado el comunicador automáticamente para una discapacidad de retraso mental	
Excepciones	Paso	Acción
	2	Si el sistema no puede configurar el entorno de comunicación en ese momento, se muestra un mensaje y a continuación el caso de uso finaliza
	3	Si el sistema no puede configurar el mecanismo de entrada en ese momento, se muestra un mensaje y a continuación el caso de uso finaliza
	4	Si el sistema no puede configurar el mecanismo de salida en ese momento, se muestra un mensaje y a continuación el caso de uso finaliza
	5	Si el sistema no puede guardar al configuración en ese instante, muestra un mensaje de error y a continuación el caso de uso finaliza
Comentarios		

RF- 42	AÑADIR CANCION	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	gestión de BBDD	
Descripción	El sistema permite al educador añadir nuevas canciones a la aplicación.	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador introduce el titulo de la canción que desea añadir
	2	El educador introduce el nombre del fichero asociado a la canción
	3	El sistema añade la canción a la aplicación
Postcondición	El educador ha añadido una nueva canción a la base de datos	
Excepciones	Paso	Acción
	1	Si el educador no completa el campo del titulo, se avisara de que no puede ser vacío y el caso de uso se volverá a realizar
	2	Si el educador no introduce el nombre del fichero asociado, se avisara de que no puede ser vacío y el caso de uso se vuelve a realizar
	3	Si el fichero ya esta asociado a otra canción en al base de datos, se avisara y se volverá a realizar el caso de uso
Comentarios		

RF- 43	BORRAR CANCION	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	gestión de BBDD	
Descripción	El sistema permite al educador borrar canciones de la aplicación.	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador introduce el titulo de la canción que desea borrar
	2	El sistema elimina la canción seleccionada de la base de datos
Postcondición	El educador ha eliminado una canción de la base de datos	
Excepciones	Paso	Acción
	1	Si el educador no completa el campo del titulo, se avisara de que no puede ser vacío y el caso de uso se volverá a realizar
	2	Si el sistema no puede borrar la canción por que ha sido programada para realizar una animación, se avisa de esta circunstancia y a continuación el caso de uso finaliza
Comentarios		

RF- 44	LISTADO CANCIONES	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	gestión de BBDD	
Descripción	El sistema debe permitir al educador obtener un listado con las canciones.	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador desea un listado de canciones
	2	El sistema muestra un listado con todas las canciones
Postcondición		
Excepciones	Paso	Acción
	1	Si no existen canciones en la base de datos, se informa de ello y el caso de uso finaliza
	2	Si el sistema no puede mostrar el listado en ese momento, se informa de ello y el caso de uso finaliza
Comentarios		

RF- 45	AÑADIR ALUMNO	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	gestión de BBDD	
Descripción	El sistema debe permitir al educador añadir un alumno a la base de datos.	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador desea añadir un nuevo alumno en la base de datos
	2	El sistema pide los datos de nuevo alumno que se desea introducir
	3	El educador introduce los datos del alumno
	4	Se realiza el caso de uso <<RF-12>> Discapacidad
	5	El sistema guarda los datos del alumno en la base de datos
Postcondición		
Excepciones	Paso	Acción
	5.a	Si el educador ha dejado algún campo vacío, el sistema muestra un mensaje advirtiéndolo y a continuación el caso de uso finaliza
	5.b	Si los datos introducidos por el educador ya figuran en la base de datos, se muestra un mensaje de error y el caso de uso finaliza
	5.c	Si el sistema no puede guardar los datos en ese momento, se muestra un mensaje informando al educador y a continuación el caso de uso finaliza
Comentarios		

RF- 46	BORRAR ALUMNO	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	gestión de BBDD	
Descripción	El sistema debe permitir al educador borrar un alumno a la base de datos.	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador desea borrar un nuevo alumno en la base de datos
	2	El sistema pide los datos del alumno que se desea borrar
	3	El educador introduce los datos del alumno
	4	El sistema pregunta si los datos introducidos son correctos
	5	El sistema elimina de la base de datos la información de ese alumno
Postcondición		
Excepciones	Paso	Acción
	5.a	Si el educador ha dejado algún campo vacío, el sistema muestra un mensaje advirtiéndolo y a continuación el caso de uso finaliza
	5.b	Si los datos introducidos son incorrectos, se muestra un mensaje con esa información y el caso de uso finaliza.
	5.c	Si el sistema no puede eliminar los datos en ese momento, se muestra un mensaje informando al educador y a continuación el caso de uso finaliza
Comentarios		

RF- 47	MODIFICAR ALUMNO	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	gestión de BBDD	
Descripción	El sistema debe permitir al educador modificar un alumno a la base de datos.	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador desea borrar un nuevo alumno en la base de datos
	2	El sistema pide los datos del alumno que se desea modificar
	3	El educador introduce los datos del alumno
	4	El sistema muestra los datos del alumno
	5	El educador introduce los nuevos datos del alumno
	6	Se realiza el caso de uso <<RF-12>> Discapacidad
	7	El sistema modifica en la base de datos la información de ese alumno
Postcondición		
Excepciones	Paso	Acción
	3	Si los datos introducidos no son correctos, el sistema muestra un mensaje de error y a continuación el caso de uso finaliza
	7.a	Si el educador ha dejado algún campo vacío, el sistema muestra un mensaje advirtiéndolo y a continuación el sistema vuelve a pedir los datos
	7.b	Si el sistema no puede modificar los datos en ese momento, se muestra un mensaje informando al educador y a continuación el caso de uso finaliza
Comentarios		

RF- 48	LISTADO ALUMNOS	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	gestión de BBDD	
Descripción	El sistema debe permitir al educador listar los alumnos de la base de datos.	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador desea un listado de los alumnos
	2	El sistema muestra un listado con toda la información de todos los alumnos
Postcondición		
Excepciones	Paso	Acción
	2	Si el sistema no puede mostrar los datos en ese momento, se muestra un mensaje informando al educador y a continuación el caso de uso finaliza
Comentarios		

RF- 49	AÑADIR IMAGEN	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	gestión de BBDD	
Descripción	El sistema debe permitir al educador añadir una imagen a la base de datos	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador desea añadir una nueva imagen al sistema
	2	El sistema pide los datos de la imagen que desea introducir
	3	El educador introduce la ruta donde se encuentra la imagen
	4	El sistema pregunta por el sonido que desea añadir a la imagen
	5	El educador introduce el sonido que desea introducir con la imagen, seleccionando de los disponibles en la base de datos
	6	El sistema confirma que se ha añadido una imagen
Postcondición		
Excepciones	Paso	Acción
	5	Si el educador no introduce ningún sonido, el sistema no asocia ninguno a la imagen que se va a añadir
	6.a	Si el sistema no puede guardar la imagen en ese momento, se muestra un mensaje de error y el caso de uso finaliza
	6.b	Si la imagen ya esta en la base de datos, el sistema avisa al educador y el caso de uso finaliza
Comentarios		

RF- 50	BORRAR IMAGEN	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	gestión de BBDD	
Descripción	El sistema debe permitir al educador borrar una imagen a la base de datos	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador desea borrar una imagen del sistema
	2	El sistema pide los datos de la imagen que desea borrar
	3	El educador introduce de entre las imágenes del sistema, el nombre de la imagen que desea borrar
	4	El sistema pregunta si desea eliminar también el sonido de la base de datos
	4.a	Si el educador responde afirmativamente, entonces se borra también el sonido asociado de la base de datos
	4.b	Si el educador responde negativamente, entonces borra solo la imagen que se ha seleccionado
	5	El sistema confirma que se ha eliminado la imagen del sistema.
Postcondición		
Excepciones	Paso	Acción
	3	Si el educador no introduce el nombre de la imagen que desea borrar, se muestra un mensaje advirtiéndolo y a continuación el caso de uso se vuelve a ejecutar
	5.a	Si el sistema no puede borrar la imagen en ese momento, se muestra un mensaje de error y el caso de uso finaliza
	5.b	Si el sistema no puede borrar el sonido en ese momento, se muestra un mensaje de error y el caso de uso finaliza
Comentarios		

RF- 51	LISTADO IMAGENES	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	gestión de BBDD	
Descripción	El sistema debe permitir al educador obtener un listado con las imágenes de la base de datos	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador desea un listado de las imágenes
	2	El sistema muestra todas las imágenes en un listado
Postcondición		
Excepciones	Paso	Acción
	2	Si el sistema no puede mostrar el listado en ese momento, muestra un mensaje de error y a continuación el caso de uso finaliza
Comentarios		

RF- 52	AÑADIR SIMBOLO	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	gestión de BBDD	
Descripción	El sistema debe permitir al educador añadir un símbolo a la base de datos	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador desea añadir un nuevo símbolo al sistema
	2	El sistema pide los datos del símbolo que desea introducir
	3	El educador introduce la ruta donde se encuentra el símbolo
	4	El sistema pregunta por el sonido que desea añadir al símbolo
	5	El educador introduce el sonido que desea introducir con el símbolo, seleccionando de los disponibles en la base de datos
	6	El sistema confirma que se ha añadido un símbolo
Postcondición		
Excepciones	Paso	Acción
	5	Si el educador no introduce ningún sonido, el sistema no asocia ninguno al símbolo que se va a añadir
	6.a	Si el sistema no puede guardar el símbolo en ese momento, se muestra un mensaje de error y el caso de uso finaliza
	6.b	Si el símbolo ya esta en la base de datos, el sistema avisa al educador y el caso de uso finaliza
Comentarios		

RF- 53	BORRAR SIMBOLO	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	gestión de BBDD	
Descripción	El sistema debe permitir al educador borrar un símbolo a la base de datos	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador desea borrar un símbolo del sistema
	2	El sistema pide los datos del símbolo que desea borrar
	3	El educador introduce de entre los símbolos del sistema, el nombre del símbolo que desea borrar
	4	El sistema pregunta si desea eliminar también el sonido de la base de datos
	4.a	Si el educador responde afirmativamente, entonces se borra también el sonido asociado de la base de datos
	4.b	Si el educador responde negativamente, entonces borra solo el símbolo que se ha seleccionado
	5	El sistema confirma que se ha eliminado el símbolo del sistema.
Postcondición		
Excepciones	Paso	Acción
	3	Si el educador no introduce el nombre del símbolo que desea borrar, se muestra un mensaje advirtiéndolo y a continuación el caso de uso se vuelve a ejecutar
	5.a	Si el sistema no puede borrar el símbolo en ese momento, se muestra un mensaje de error y el caso de uso finaliza
	5.b	Si el sistema no puede borrar el sonido en ese momento, se muestra un mensaje de error y el caso de uso finaliza
Comentarios		

RF- 54	LISTADO SIMBOLOS	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	gestión de BBDD	
Descripción	El sistema debe permitir al educador obtener un listado con los símbolos de la base de datos	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador desea un listado de los símbolos
	2	El sistema muestra todos los símbolos en un listado
Postcondición		
Excepciones	Paso	Acción
	2	Si el sistema no puede mostrar el listado en ese momento, muestra un mensaje de error y a continuación el caso de uso finaliza
Comentarios		

RF- 55	AÑADIR SONIDO	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	gestión de BBDD	
Descripción	El sistema debe permitir al educador añadir un sonido a la base de datos	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador desea añadir un nuevo sonido al sistema
	2	El sistema pide los datos del sonido que desea introducir
	3	El educador introduce la ruta donde se encuentra el sonido
	4	El sistema pregunta por la imagen o el símbolo que desea añadir al sonido
	5	El educador introduce la imagen o el símbolo que desea introducir con el sonido, seleccionando de los disponibles en la base de datos
	6	El sistema confirma que se ha añadido un sonido
Postcondición		
Excepciones	Paso	Acción
	5	Si el educador no introduce ningún símbolo o imagen, el sistema no asocia ninguno al sonido que se va a añadir
	6.a	Si el sistema no puede guardar el sonido en ese momento, se muestra un mensaje de error y el caso de uso finaliza
	6.b	Si el sonido ya esta en la base de datos, el sistema avisa al educador y el caso de uso finaliza
Comentarios		

RF- 56	BORRAR SONIDO	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	gestión de BBDD	
Descripción	El sistema debe permitir al educador borrar un sonido a la base de datos	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador desea borrar un sonido del sistema
	2	El sistema pide los datos del sonido que desea borrar
	3	El educador introduce de entre los sonidos del sistema, el nombre del sonido que desea borrar
	4	El sistema pregunta si desea eliminar también el símbolo o la imagen asociada a ese sonido de la base de datos
	4.a	Si el educador responde afirmativamente, entonces se borra también el símbolo o la imagen asociado de la base de datos
	4.b	Si el educador responde negativamente, entonces borra solo el sonido que se ha seleccionado
	5	El sistema confirma que se ha eliminado el sonido del sistema.
Postcondición		
Excepciones	Paso	Acción
	3	Si el educador no introduce el nombre del sonido que desea borrar, se muestra un mensaje advirtiéndolo y a continuación el caso de uso se vuelve a ejecutar
	5.a	Si el sistema no puede borrar el sonido en ese momento, se muestra un mensaje de error y el caso de uso finaliza
	5.b	Si el sistema no puede borrar el símbolo o la imagen en ese momento, se muestra un mensaje de error y el caso de uso finaliza
Comentarios		

RF- 57	LISTADO SONIDOS	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	gestión de BBDD	
Descripción	El sistema debe permitir al educador obtener un listado con los sonidos de la base de datos	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador desea un listado de los sonidos
	2	El sistema muestra todos los sonidos en un listado
Postcondición		
Excepciones	Paso	Acción
	2	Si el sistema no puede mostrar el listado en ese momento, muestra un mensaje de error y a continuación el caso de uso finaliza
Comentarios		

RF- 58	CREAR PLANTILLA		
Versión	1.0		
Autores	Jose Carlos Jiménez Sánchez		
Fuentes			
Objetivos asociados	gestión de BBDD		
Descripción	El sistema debe permitir al educador añadir una plantilla a la base de datos		
Precondición			
Secuencia Normal	Paso	Acción	
	1	El educador desea crear una nueva plantilla	
	2	El sistema pide el nombre de la plantilla	
	3	El educador introduce el nombre de la plantilla que desea añadir	
	4	El educador añade un símbolo, imagen o sonido a la plantilla	
	5	El sistema pregunta si desea seguir introduciendo objetos en la plantilla	
		5.a	Si el educador responde afirmativamente, se realiza otra vez el paso 4
		5.b	Si el educador responde negativamente, el sistema confirma los datos se han guardado en la base de datos
Postcondición			
Excepciones	Paso	Acción	
	5.b.1	Si el educador no introduce ningún nombre de la plantilla, el sistema muestra un mensaje y el caso de uso vuelve a ejecutarse	
	5.b.2	Si el sistema no puede guardar la plantilla en ese momento, se muestra un mensaje de error y el caso de uso finaliza	
Comentarios			

RF- 59	MODIFICAR PLANTILLA	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	gestión de BBDD	
Descripción	El sistema debe permitir al educador modificar una plantilla a la base de datos	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador desea modificar una plantilla
	2	El sistema muestra los nombres de las plantillas de la base de datos
	3	El educador elige el nombre de la plantilla a modificar
	4	El educador modifica los sonidos, imágenes u objetos de la plantilla
	5	El sistema guarda los cambios realizados en la plantilla
Postcondición		
Excepciones	Paso	Acción
	5	Si el sistema no puede guardar la plantilla en ese momento, se muestra un mensaje de error y el caso de uso finaliza
Comentarios		

RF- 60	BORRAR PLANTILLA	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	gestión de BBDD	
Descripción	El sistema debe permitir al educador borrar una plantilla a la base de datos	
Precondición		
Secuencia Normal	Paso	Acción
	1	El educador desea borrar una plantilla
	2	El sistema muestra los nombres de las plantillas de la base de datos
	3	El educador elige el nombre de la plantilla a eliminar
	4	El sistema confirma la eliminación de la plantilla
Postcondición		
Excepciones	Paso	Acción
	4	Si el sistema no puede borrar la plantilla en ese momento, se muestra un mensaje de error y el caso de uso finaliza
Comentarios		

RF- 61	USAR COMUNICADOR	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	Permitir la comunicación	
Descripción	El sistema deberá permitir al alumno utilizar el comunicador para comunicarse con los demás	
Precondición		
Secuencia Normal	Paso	Acción
	1	El alumno desea utilizar el comunicador
	1.a	Si el alumno esta identificado, el comunicador se adapta a su configuración particular
	1.b	Si el alumno no esta identificado, el comunicador se configura con la configuración para uso general.
	2	El sistema busca la plantilla que esta seleccionada en ese momento y la muestra
	3	Pt. Extensión: plantilla
	4	Se realiza el caso de uso <<RF-63>> Selección símbolo
	5	Pt. Extensión: borrar
	6	Se realiza el caso de uso <<RF-65>> Mostrar mensaje
	7	El sistema espera un tiempo y a continuación borra el mensaje
Postcondición		
Excepciones	Paso	Acción
	1	Si el sistema no puede cargar la configuración del comunicador, el sistema muestra un mensaje advirtiéndolo de ello y el caso de uso finaliza
Comentarios		

RF- 62	CAMBIAR PLANTILLA <<extend RF-61>>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	Permitir la comunicación	
Descripción	El sistema deberá permitir al alumno cambiar la plantilla activa por la que se seleccione	
Precondición		
Secuencia Normal	Paso	Acción
	1	El alumno desea cambiar la plantilla activa
	2	el sistema muestra las plantillas de la base de datos
	3	El alumno selecciona la plantilla que desea
	4	El sistema actualiza la ventana con la nueva plantilla
Postcondición		
Excepciones	Paso	Acción
	2	Si el sistema no puede cargar las plantillas de la base de datos, el sistema muestra un mensaje advirtiéndolo de ello y el caso de uso finaliza
	4	Si el sistema no puede actualizar la plantilla activa, muestra un mensaje de error y el caso de uso finaliza
Comentarios		

RF- 63	SELECCIÓN SIMBOLO	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	Permitir la comunicación	
Descripción	El sistema deberá permitir al alumno seleccionar un símbolo o imagen de la plantilla mediante el mecanismo de entrada seleccionado	
Precondición		
Secuencia Normal	Paso	Acción
	1	El alumno desea seleccionar un símbolo
	2	El sistema hace sonar el fichero de sonido asociado al símbolo sobre el que se esta seleccionado
	2.a	Si el alumno utiliza el mecanismo de entrada, este símbolo se añadirá al mensaje del comunicador
	2.b	Si no utiliza el mecanismo de entrada, el sistema pasara al siguiente símbolo de la plantilla y se volverá a realizar la secuencia numero 2
Postcondición		
Excepciones	Paso	Acción
	2	Si el sistema no puede reproducir el archivo de sonido asociado a esa imagen o símbolo, siempre que este tenga un sonido asociado, el sistema muestra un mensaje de error y el caso de uso finaliza
Comentarios		

RF- 64	BORRAR SELECCIÓN <<extend RF-61>>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	Permitir la comunicación	
Descripción	El sistema deberá permitir al alumno borrar el ultimo elemento del mensaje	
Precondición		
Secuencia Normal	Paso	Acción
	1	El alumno desea borrar el ultimo elemento del mensaje
	2	el sistema elimina el ultimo símbolo o imagen elegida
	3	El sistema actualiza la ventana del mensaje
Postcondición		
Excepciones	Paso	Acción
	2	Si el sistema no puede borrar el ultimo elemento del mensaje, muestra un mensaje de error y el caso de uso finaliza
Comentarios		

RF- 65	MOSTRAR MENSAJE	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	Permitir la comunicación	
Descripción	El sistema deberá permitir al alumno mostrar el mensaje que ha realizado	
Precondición		
Secuencia Normal	Paso	Acción
	1	El alumno desea mostrar su mensaje
	2	el sistema para cada símbolo o imagen reproduce su archivo de sonido asociado
Postcondición		
Excepciones	Paso	Acción
	2.a	Si el mensaje esta vacío, se muestra un mensaje advirtiéndolo y a continuación el caso de uso finaliza
	2.b	Si el sistema no puede mostrar el mensaje, se mostrara un mensaje de error y el caso de uso finaliza
Comentarios		

RF- 66	EVALUACION <<extend RF-34>>		
Versión	1.0		
Autores	Jose Carlos Jiménez Sánchez		
Fuentes			
Objetivos asociados	evaluación		
Descripción	El sistema o el educador deberá evaluar si el ejercicio es correcto		
Precondición			
Secuencia Normal	Paso	Acción	
	1	El educador desea comprobar la corrección del ejercicio	
	2	El sistema evalúa el ejercicio	
		2.a	SI el ejercicio es correcto, se lo comunica al educador y el caso de uso continua
		2.b	Si es incorrecto, se lo comunica al educador y se vuelve a ejecutar el ejercicio
Postcondición			
Excepciones	Paso	Acción	
	2	Si el sistema no puede evaluar el ejercicio en ese momento, se mostrara un mensaje de error y el caso de uso finaliza	
Comentarios			

4.4. RELACION CASOS DE USO – OBJETIVOS

Después de introducir en la herramienta R^2 tanto los objetivos como los casos de uso, a continuación hay que introducir la relación entre cada caso de uso y el objetivo al que está asociado, normalmente se asocia un caso de uso por objetivo aunque también puede haber varios casos de uso por objetivo. Esas relaciones quedan guardadas en la base de datos de la aplicación por lo que luego, empleando consultas SQL se pueden obtener las relaciones entre casos de uso y objetivos, que son bastante importante debido a que pueden ser utilizadas en otras aplicaciones.

En el Dominio de la Informática para Discapacitados y usando los casos de uso y requisitos que se introdujeron en la herramienta, la consulta devolvió estos resultados:

Relación Casos de uso – Objetivos	
CASO DE USO	OBJETIVO
animación	preparar animación
añadir alumno	biblioteca alumnos
añadir canción	gestión de canciones
añadir imagen	gestión imágenes
añadir símbolo	gestión símbolos

Relación Casos de uso – Objetivos	
CASO DE USO	OBJETIVO
añadir sonido	gestión sonidos
auditivas	Auditivas
autismo	Autismo
borrar canción	gestión de canciones
borrar imagen	gestión imágenes
borrar plantilla	gestión plantillas
borrar selección	Permitir la comunicación
borrar símbolo	gestión símbolos
borrar sonido	gestión sonidos
cambiar plantilla	Permitir la comunicación
configuración general	Configuración
configuración particular	Configuración
configurar comunicador	Configuración
crear plantilla	gestión plantillas
demostración	Demostración
discapacidad	Permitir distintas discapacidades
ejecutar lección	realizar ejercicios
ejercicio	preparar lección
eliminar alumno	biblioteca alumnos
entornos de comunicación	soportar distintos entornos
evaluar lección	evaluar lección
foto	Foto
habla	Habla
identificación alumno	permitir identificación
identificación educador	permitir identificación
lista de nombres	lista de nombres
listado alumnos	biblioteca alumnos
listado canciones	gestión de canciones
listado imágenes	gestión imágenes
listado símbolos	gestión símbolos
listado sonidos	gestión sonidos
manual	Manual
mecanismos de entrada	soportar distintos mecanismos de entrada
mecanismos de salida	soportar distintos mecanismos de salida
micrófono	Micrófono
modificar alumno	biblioteca alumnos
modificar plantilla	gestión plantillas
mostrar mensaje	Permitir la comunicación
motora	Motora
nombre	Nombre
ordenador	Ordenador
particular	Seguimiento

Relación Casos de uso – Objetivos	
CASO DE USO	OBJETIVO
PDA	PDA
pictográficos	Pictográficos
portátil	Portátil
preparar lección	solucionar problemas de aprendizaje
pulsador	Pulsador
realización	Realización
Realizar animación	realizar animaciones
realizar lección	realizar lección
retraso mental	retraso mental
seguimiento alumno	Seguimiento
selección símbolo	Permitir la comunicación
Sistema automático	Automático
sonidos	Sonidos
teclado braille	teclado braille
teclado/ratón	teclado/ratón
texto	Texto
todos	Seguimiento
usar comunicador	Permitir la comunicación
visuales	Visuales

Como se puede ver, esta consulta muestra rápidamente los casos de uso y sus objetivos relacionados.

4.5. APLICACIÓN A UN CASO PRÁCTICO

4.5.1. INTRODUCCION

Para ver la practicidad de todo lo anteriormente expuesto, en este apartado se emplea una herramienta ya existente de técnicas de análisis de variabilidad, creada por:

Bruno González Baixauli (Universidad de Valladolid).

Julio Cesar Sampaio do Prado Leite (Universidad Politécnica de Sao Paulo).

John Mylopoulos (Universidad de Toronto).

La idea era, ver si utilizando esta metodología de reutilización, se podía reducir el tiempo de realización de aplicaciones, así como qué porcentaje de casos de uso se podrían haber reutilizado utilizando esta herramienta de análisis.

Para ello se tomo como ejemplo una aplicación ya realizada de un comunicador para personas con parálisis cerebral, se estudiaron como están realizados los casos de uso y como se hubiera realizado usando esta herramienta de análisis, al final se compararon los resultados.

El proyecto empleado para comparar los resultados fue:

Mª Belén Arenal González y Cristina Mª López Aguado. "Comunicador", Junio 2002.

De este proyecto se sacaron los casos de uso que se compararon con los resultados de la herramienta de análisis que se empleó.

El siguiente apartado expone una pequeña parte de los fundamentos de dicha aplicación para después mostrar el ejemplo realizado con los requisitos expresados anteriormente en la herramienta R² y los casos de uso, para comparar con la aplicación ya realizada en el proyecto indicado.

4.5.2. ENTORNO UTILIZADO

El modelo que se va a emplear aquí, utiliza una parte del *NFR Framework* explicado en el apartado 4.1. Se emplea la combinación de dos sub-modelos: Uno es para los requisitos funcionales (**Goals**) y tareas, mientras que el otro es para los requisitos no funcionales (**Softgoals**). Ambos sub-modelos son diagramas de árboles AND/OR. La figura 3 muestra un ejemplo de los sub-modelos.

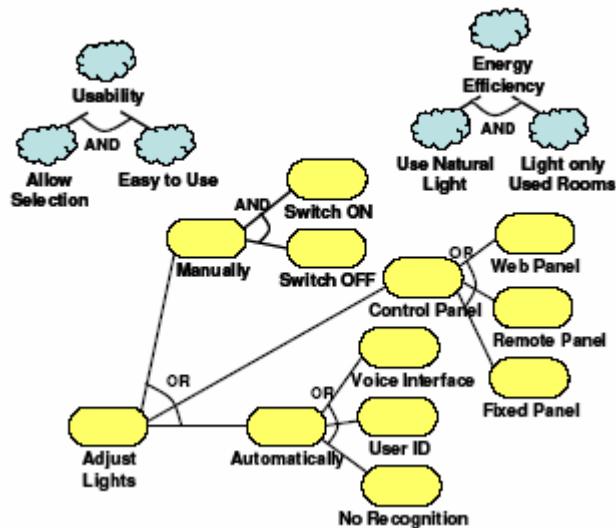


Figura 3:Goals (amarillo) y Softgoals (azul) para un sistema de control de luz (parcial).

Una vez que los dos sub-modelos están disponibles, es posible relacionarlos a través de correlaciones, en la figura 4 se muestra un ejemplo de las posibles correlaciones entre los dos sub-modelos.

Es importante señalar que cada rama OR en el sub-modelo de requisitos funcionales (Goals) representa una posible variante para el software.

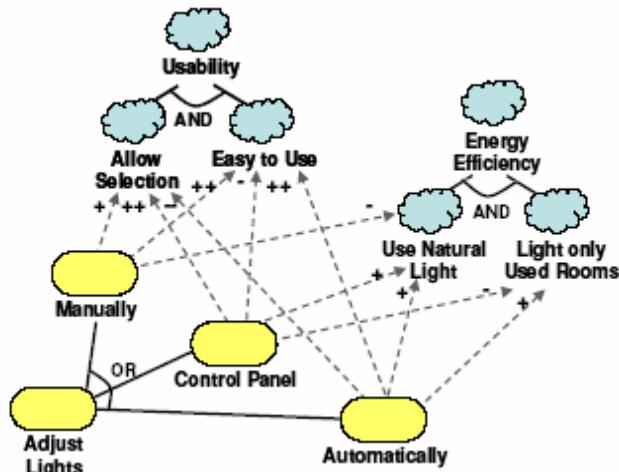


Figura 4: Correlaciones entre Goals y Softgoals.

Terminadas de establecer las correlaciones de los dos modelos, se elige una variante particular seleccionando un conjunto de criterios (Softgoals) .

Introduciendo los 2 sub-modelos y empleando una tabla de decisiones, se estructura el problema de tal forma que la herramienta de análisis es capaz de procesarlo. Específicamente, el modelo de requisitos funcionales expresa las posibles variantes, y el modelo de requisitos no funcionales describe los criterios por los cuales se elige entre una variante u otra.

En un contexto más general, las tablas de decisiones tienen *condiciones* que implican algunas *acciones*. Esas condiciones están en columnas y para cada una se marcan las acciones requeridas. Si se representan los Softgoals como condiciones (los factores de calidad que deseamos) y los requisitos funcionales como acciones (la funcionalidad a implementar), se podrán comparar las variantes y escoger la mejor entre los Softgoals que se escojan.

En la figura 5 se muestra como seria una tabla de decisiones, se observa que una tabla es una manera natural de implementar este análisis, perfecta para poder trabajar con la herramienta de análisis.

		SG Combinations			
Actions	Conditions				
	Functional Goals Selection				
	Analysis				
	Soft Goals				

Figura 5: Tabla de decisiones con los Softgoals como condiciones y los requisitos funcionales como acciones.

Una vez que los sub-modelos están representados en la tabla, se procede a correlacionar los Softgoals con el modelo de requisitos funcionales (solo están permitidas las correlaciones entre los

modelos). Para llevarlo a cabo, se colocan los requisitos funcionales en filas y los requisitos no funcionales en columnas.

El esquema de conversión de la tabla 1 se usa para otorgar la propagación cuantitativa de todas las posibles variantes. Esta conversión es necesaria para obtener una puntuación total con el que comparar las distintas variantes. La propagación cuantitativa utiliza el mismo esquema que el modelo probabilístico propuesto por Giorgini, pero simplificado.

Este algoritmo usa dos valores para cada objetivo: uno para el grado de satisfacción ($Sat(G)$) y otro para el grado de denegación ($Den(G)$) que se representan como la probabilidad de que G es satisfecho o no. Para realizar esta propagación, se utilizan los operadores probabilísticos de la tabla 2. Este esquema muestra los valores de satisfacción que dan los Softgoals a una variante.

Las prioridades de los requisitos no funcionales se usan con el fin de calcular valores de variantes, para propósitos de comparación. Las prioridades son dadas en una escala de tantos por ciento por requisitos no funcionales.

<i>Satisfaction Value</i>	<i>S/++</i>	<i>PS/+</i>	<i>N</i>	<i>PD/-</i>	<i>D/--</i>
<i>Quantitative Value</i>	(1, 0)	(0.5, 0)	(0, 0)	(0, 0.5)	(0, 1)

Tabla 1: Esquema de conversión de las etiquetas a valores cuantitativos de satisfacción y denegación

	$(SG_2, SG_3) \xrightarrow{sat} SG_1$	$(SG_2, SG_3) \xrightarrow{x} SG_1$	$G_2 \xrightarrow{sat} SG_1$
$Sat(SG_1)$	$Sat(SG_2) \otimes Sat(SG_3)$	$Sat(SG_2) \otimes Sat(SG_3)$	$Sat(G_2) \otimes Sat(SG_1)$
$Den(SG_1)$	$Den(SG_2) \otimes Den(SG_3)$	$Den(SG_2) \otimes Den(SG_3)$	$Den(G_2) \otimes Den(SG_1)$

Tabla 2: Valores del algoritmo de propagación. SGx son los Softgoals, Gx son los Goals.

El esquema del proceso de análisis es el siguiente: primero el experto del dominio crea los dos sub-modelos de las fuentes del dominio. Los datos de los modelos se linearizan y el análisis de variantes los correlaciona, establece prioridades entre los Softgoals y analiza las tablas comparando las variantes, esto puede ser echo iterativamente, cambiando correlaciones hasta que se encuentra el mejor resultado. Los resultados son devueltos al experto del dominio. En la figura 6 podemos ver todo el proceso completo.

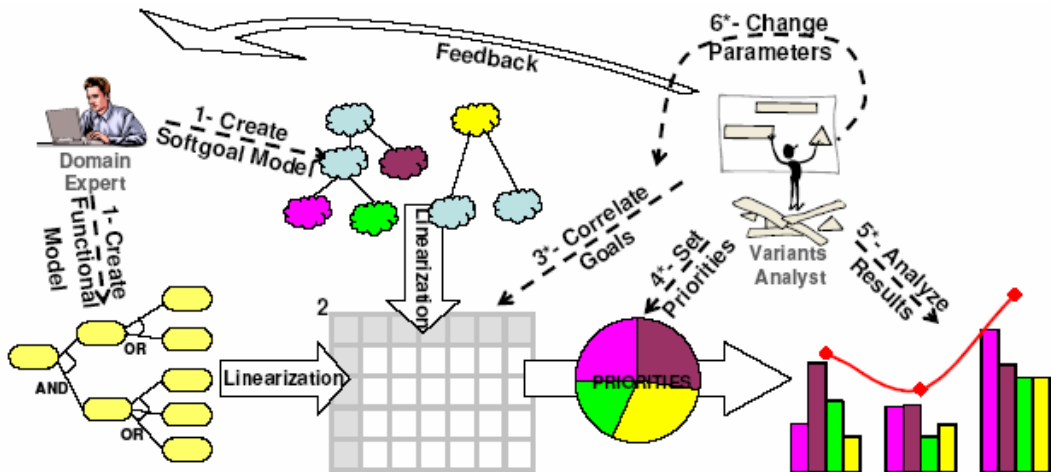


Figura 6: El proceso de análisis de objetivos

La figura 7 muestra la manera de introducir los datos en la herramienta manualmente por el experto del dominio, cabe destacar que todas las posibles relaciones entre Goals y Softgoals se muestran en una tabla (tabla 3), las celdas grises indican la imposibilidad de introducir en ellas correlaciones, por ejemplo si se correlaciona un Goal con un Softgoal, los posibles sub-Goals de ese objetivo no pueden ser correlacionados.

El formulario 'Children Selection' permite introducir datos manualmente. Incluye un campo 'Adjust lights' con un botón 'Leaf Child'. Hay una sección 'Relation Type' con un menú desplegable que muestra 'Xor'. A la derecha, hay un campo 'Children Number' con el valor '3' y un botón '+'. En la parte superior derecha, hay un campo 'Manually' con un botón 'Control Panel' y un botón 'Automatically'. En la parte inferior, hay botones '<< Previous' y 'Next >>'. Hay una columna de casillas de verificación a la derecha de cada fila de datos.

Figura 7: Formulario de introducción de datos.

	En Eff	Li on Us Ro	Us Na Li Us	Al Se	Ea to Us
Ad Li					
Ma				+	++
Sw ON					
Sw OF					
Co Pa	-		+	++	-
We Pa					
Re Pa					
Fi Pa					
Av	+		+	--	++
Vo In					
Us ID					
Na Re					

Done

Tabla 3: Tabla de relaciones.

La figura 8 muestra como se establecen las prioridades, al igual que en las correlaciones, si se da un valor al objetivo, sus sub-objetivos no pueden ser seleccionados.

	En Eff	Li on Us Ro	Us Na Li Us	Al Se	Ea to Us
Priorities	50			50	

Done

Total Priority
100 %

Figura 8: Formulario de Prioridades.

El análisis ofrece dos ventajas: Por un lado, es fácil cambiar el nivel de prioridad que se desea, por el otro, los valores de correlación pueden ser cambiados por el usuario, pero estos cambios requieren más esfuerzo.

4.5.3. ESTUDIO DEL CASO PRÁCTICO

Después de ver los fundamentos de la herramienta, procederemos a explicar cómo fue empleada para realizar el ejemplo en el dominio que se deseaba tratar: *La Informática para Discapacitados*. Lo primero que se hizo fue introducir los Diagramas de Requisitos que se crearon con la aplicación R^2 . Para ello se empleó la plantilla “**Wizard Models**” en la cual se podían introducir los dos diagramas de requisitos. La manera de introducir los datos en dicha herramienta es en forma de grafo, de izquierda a derecha pasando por objetivos y subobjetivos. En estos diagramas se deben cambiar las siglas *OB*, *OP* y *MU* de la herramienta R^2 por **AND**, **OR**, **XOR** para que la aplicación se ejecute, estas siglas nos ofrecen el nivel de variabilidad como en la herramienta R^2 .

Tanto el Modelo de Requisitos Funcionales como el Modelo de Requisitos no Funcionales se pueden ver en el CD-ROM que acompaña a esta memoria.

El modelo de requisitos funcionales esta en la carpeta **herramienta variabilidad** en el archivo **RF DISCAPACITADOS**.

El modelo de requisitos no funcionales esta en la misma carpeta en el archivo **RNF DISCAPACITADOS**.

Se deben guardar estos diagramas de requisitos para, a continuación, utilizarlos con la herramienta **“WizardPBVC”** que es la que realmente realizó el análisis.

Al iniciar esta aplicación, pide al experto introducir los dos diagramas realizados anteriormente de requisitos funcionales así como de requisitos no funcionales.

Seguidamente la tabla de decisiones se muestra con el fin de poder introducir los valores de satisfacción que se crean convenientes. La tabla se ordenó como se ha visto anteriormente, los Requisitos Funcionales son acciones que hay que realizar estando colocados en filas, y los requisitos no funcionales son decisiones sobre las acciones que se quieren llevar a cabo.

Goals Correlations Input

	Di vi	Di au	Di mo	Re me	au	Si	Fa de ma	Fa in	Fa mo
PD	-	-	-	-	+	-	-		
Po	-	-	-	+	++	+	+		
So di									
So									
Mi	+	+	++	++	+	+		+	+
Me									
Pu	--	+	++	++	+	+		-	+
Te	--	+	+	-	++	++		++	+
Te	++	--	--	--	--	-		-	+
So di									
Pi	--	++	++	++	++	+	+		
So	++	--	++	++	++	+	+		
te	-	+	-	--	++	+		-	+
Pe ge									
Mo									
Mu									+
Co									

Pulsador x Discapacidad motora

Insert for each functional goal its influences to the soft goal. Only the next values will be considered:

S (or ++) --> Implies satisfaction D (or --) --> Implies denial
 + --> Contributes positively - --> Contributes negatively
 U (or any other value) --> Doesn't influence

Done

Tabla 4: Tabla de decisiones para el ejemplo.

Se puede comprobar como al introducir un grado resatisfacción entre objetivos, las casillas correspondientes a sus subobjetivos quedan marcadas en gris para imposibilitar la introducción de nuevos grados de satisfacción.

Con posterioridad se introdujeron las prioridades de nuestro análisis. El comunicador realizado en el proyecto de fin de carrera que se seleccionó como ejemplo tenía como objetivos prioritarios:

- Lograr una mejor integración de las personas con problemas de comunicación.
- Manipulación sencilla de la herramienta para una asimilación rápida e incorporación a la vida cotidiana.
- Englobar todas las capacidades tanto motrices como cognitivas para no limitar al usuario.
- Poder expresar tanto los deseos como las necesidades de una persona de manera sencilla para el usuario.
- Adaptar la configuración del comunicador dependiendo del usuario para su mejor uso.
- No cerrarnos a las opciones escogidas, sino dejar una puerta abierta para posibles modificaciones y ampliaciones.
- Dotar de la máxima independencia al usuario.
- El comunicador estaba enfocado a personas con parálisis cerebral que es un tipo de discapacidad motora.
- Aumento de la capacidad de comunicarse de las personas con discapacidades.

Al introducir las prioridades se debían de tener en cuenta estos objetivos, dotando de mayor prioridad a los aquí descritos, se tuvo que dividir el 100% entre los diferentes objetivos que se querían cumplir, así nuestras prioridades fueron:

1. *Discapacidad del Habla* **10%**
2. *Discapacidad motora* **30%**
3. *Retraso mental* **10%**
4. *fácil de mantener* **10%**
5. *Tipos de discapacidades* (referido a introducir nuevas discapacidades y nuevos mecanismo para tratarlas) **10%**
6. *Fomenta la independencia* **10%**
7. *Fomenta la comunicación* **20%**

The screenshot shows a window titled "Priorities Selection" with a table of priorities and a total priority display.

Priorities	Di de Ha	Di vi	Di au	Di mo	Re me	au	Si	Fa de ma	Fa
	10			30	10			10	

Below the table, there is a "Soft Goal:" section with the text "Discapacidad motora".

At the bottom right, a green box displays "Total Priority" as "100 %".

A "Done" button is located at the bottom center.

Tabla 5: Selección de prioridades para el comunicador.

Una vez introducidos las prioridades, el análisis estaba realizado, los resultados se muestran en varias pantallas, en el grafico 1 se puede ver la selección de prioridades y la leyenda del Requisito no Funcional (Softgoal) al que pertenece. Se puede ver que la mayor prioridad se le ha

dado a la discapacidad motora debido a que el comunicador se empleaba por personas con este tipo de discapacidades, y la segunda mayor prioridad corresponde a el fomento de la comunicación, ya que el objetivo de cualquier comunicador es comunicarse eficazmente con los demás.

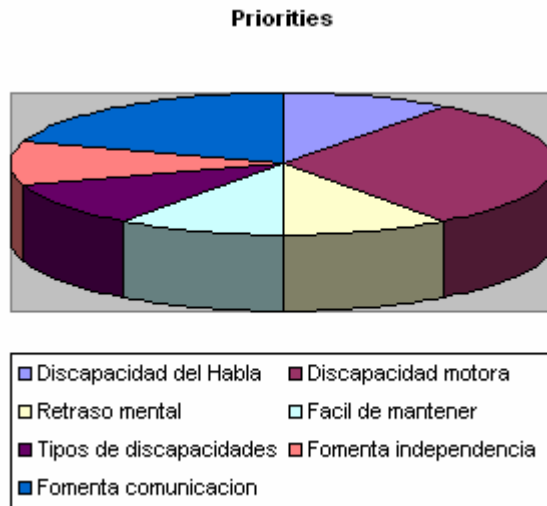


Gráfico 1: Selección de prioridades realizada.

El resto de gráficos ofrecen una muestra de comparaciones entre Softgoals mostrados en su totalidad o independientemente, conflictos entre distintas variantes, etc.. el principal problema es que debido al elevado número de variantes que ofrece nuestro dominio estudiado, los gráficos son poco claros ya que en nuestro caso estamos trabajando con mas de veintitrés mil variantes posibles, complicando mucho el análisis y los gráficos de resultados.

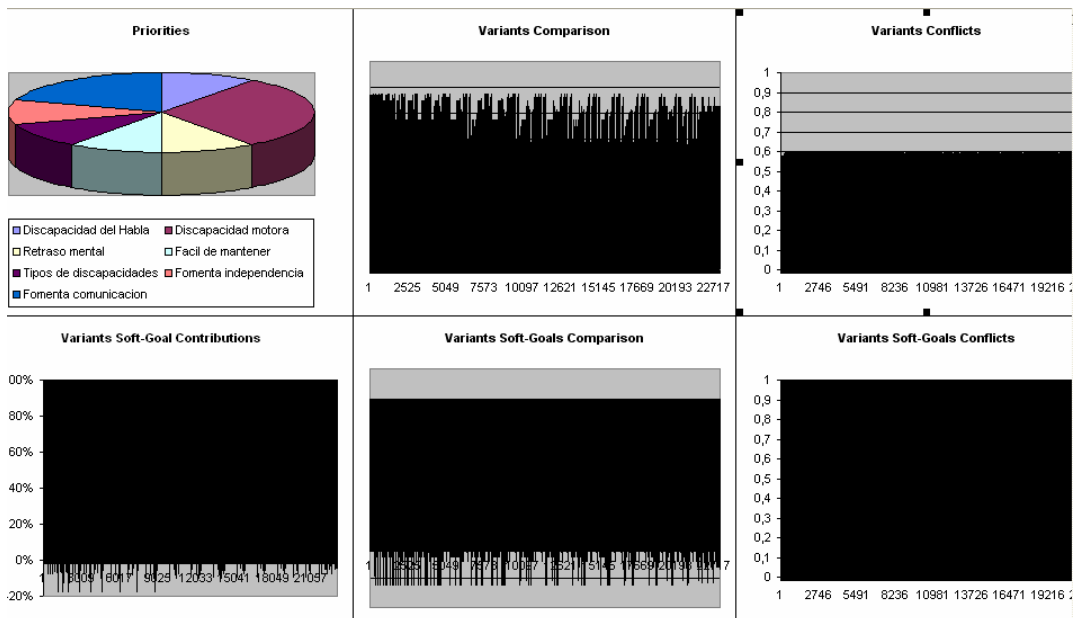


Gráfico 2: Distintas graficas para comparaciones, conflictos, etc. nótese la dificultad de ver los valores debido al elevado número de variantes del dominio.

El resultado del análisis nos da también un grafico en forma de diagrama de árbol de objetivos, tanto funcionales (que componen el diagrama de árbol en si mismo) como los no funcionales (que se encuentran en la parte superior del grafico).

Hasta ahora, los resultados de las variantes que se ven de la aplicación son numéricos, para ver los objetivos asociados a ellos, se puede usar una nueva barra de comandos de la herramienta que incluye todas las variantes y sus nombres. Esta barra de comandos permite modificar las prioridades y las correlaciones.

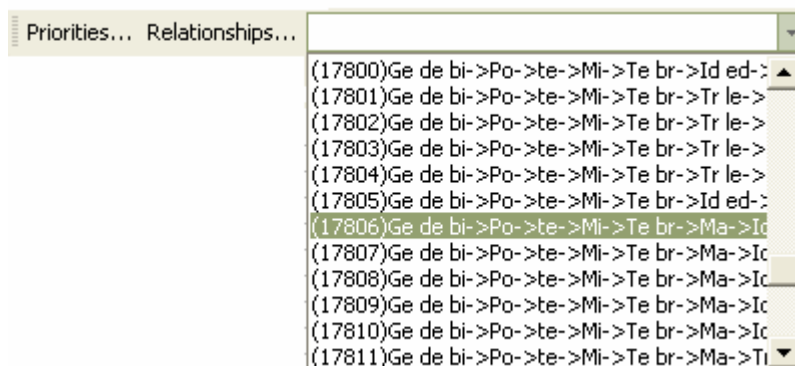


Figura 9: Barra de comandos: Permite modificar las prioridades y las correlaciones.

Cuando una variante es seleccionada, el árbol muestra los valores de satisfacción para cada objetivo dentro de esa variante. El grado de satisfacción mostrado en el árbol representa los conflictos usando un código de colores, desde el verde, si no hay conflicto, hasta el rojo, si el Softgoal tiene el mismo número de contribuciones positivas y negativas.

Los objetivos seleccionados en el análisis se muestran en color amarillo, dichos objetivos son el resultado de nuestro análisis. Con estos objetivos y los casos de uso que implementan dichos objetivos se pueden comparar los resultados con el comunicador que fue elegido, para ver los porcentajes de casos de uso que se podían haber reutilizado y el tiempo y coste ahorrado en desarrollarlos.

El diagrama de árbol resultante en este ejemplo se puede ver en el contenido del CD-ROM que acompaña a la memoria en la carpeta **Herramienta Variabilidad** en el archivo **Herramienta variabilidad_comunicador**.

Seguidamente, para comparar los casos de uso realizados con este análisis y los llevados a cabo en el proyecto tomado como referencia, se construyó una tabla de manera que cada caso de uso que se seleccionó del análisis, correspondía con algún caso de uso del proyecto del comunicador, si era así, ese caso de uso se podría haber reutilizado para la realización del comunicador evitando realizar el análisis desde el principio. Esta técnica da un porcentaje de casos de uso reutilizables en este proyecto si hubiéramos seguido esta técnica de análisis.

Cabe destacar que lo importante de la reutilización de los casos de uso no es el nombre del caso de uso sino el interior, es decir, la secuencia, las excepciones, etc... de dichos casos de uso.

Para ver la secuencia de casos de uso del proyecto del comunicador léase el proyecto de fin de carrera;

M^a Belén Arenal González y Cristina M^a López Aguado. Comunicador, Junio 2002.

Casos de Uso empleando la herramienta de análisis	Casos de Uso del proyecto de comunicador
Añadir alumno	Añadir alumno
Borrar alumno	Borrar alumno
Modificar Alumno	Modificar Alumno
Listado Alumno	Listado Alumno
Añadir imagen	Añadir imagen
Borrar imagen	Borrar imagen
Listado imagen	Listado imagen
Añadir símbolo	Añadir símbolo
Listado símbolo	Listado símbolo
Borrar símbolo	Borrar símbolo
Crear plantilla	Crear plantilla
Modificar plantilla	Modificar plantilla
Borrar plantilla	Borrar plantilla
	Ayuda
Identificación Educador	Identificación programador
	Cambiar contraseña
	Imprimir símbolo/imagen perteneciente a una plantilla
	Imprimir símbolo/imagen
lista de nombres	Identificación de alumno
Usar comunicador	Inicio
	Escribir
Selección de símbolo	Selección de símbolo
Mostrar mensaje	Leer
Borrar selección	Borrar
Cambiar plantilla	Cambiar plantilla
	Avanzar símbolos
	Retroceder símbolos
	Barrido
	Imprimir comunicador
	Imprimir plantilla
	Añadir programador

Además de estos casos de uso, la herramienta de análisis seleccionó más casos de uso acerca de configuración manual de entrada/salida, entornos, etc...que también se podían reutilizar para realizar el comunicador. El resto de casos de uso que no eran reutilizables, correspondían a casos de uso que el programador había realizado como propios para este comunicador, por ejemplo, impresiones, añadir programador, etc....

4.5.3.1. PROBLEMAS ENCONTRADOS

En la aplicación de las técnicas y herramientas explicadas anteriormente se han encontrado ciertas dificultades que se enumeran a continuación:

Herramienta Requirements Reuse (R²)

El primer problema que se ha encontrado y quizás el mas importante es que a partir de un determinado volumen de datos, o cuando se lleva trabajando con la aplicación durante un tiempo largo, esta no rinde al mismo nivel que al principio, incluso puede dejar de grabar algunos datos o quedarse bloqueada en el proceso de grabación de los datos.

Como digo es el principal problema ya que se pierde información importante con cierta facilidad, ocurre cuando se lleva trabajando con la herramienta unas horas y al guardar los datos se observa que no los ha grabado.

Las siguientes dificultades son más de manejo de la aplicación que de problemas que de la misma. Con las dificultades que me he encontrado son las siguientes:

La dificultad de guardar la información al realizar un diagrama de objetivos, ya que no se utiliza el mismo botón que para guardar el resto de diagramas sino otro situado al lado izquierdo de la pantalla del diagrama de objetivos como una “G” negra, es evidente que a fuerza de utilizar la aplicación un usuario se acaba acostumbrando a esto, pero en la primera ocasión que se realiza el diagrama de objetivos resulta un tanto complicado saber que hay que seleccionar ese botón y no el que se emplea normalmente para guardar los diagramas en las aplicaciones.

También relacionado con esto esta la imposibilidad de poder diferenciar los requisitos funcionales de los no funcionales, existe una única ventana para ambos tipos de requisitos lo que hace pensar que únicamente se deben introducir los requisitos funcionales, aunque la experiencia en este proyecto indica que son importantes los dos tipos de requisitos.

Otra dificultad es que al realizar varios paquetes para casos de uso, aunque se ponga el mismo actor en dos paquetes diferentes, no son detectados como el mismo actor sino que son diferentes actores cosa que en el fondo no es verdad ya que normalmente es el mismo actor en un paquete y en otro, al guardar la información de un actor en un paquete, esta no aparece en el resto de paquetes de casos de uso, esto mismo pasa con los casos de uso, que al guardar la secuencia de uno en un paquete, esta, no queda reflejada en el mismo caso de uso en otros paquetes.

La descripción de los casos de uso se debería poder mejorar y exponer de un modo grafico y claro mostrando cuales son los puntos de extensión en los diagramas, así como un lugar en la descripción en la que poder ponerlos. También seria recomendable que la relación entre un caso de uso o un paquete de casos de uso y un objetivo fuese mas clara.

La generalización de los actores, o que haya un actor abstracto es otra de las cuestiones que se debería poder poner gráficamente.

Otra de las dificultades es la impresión de un proyecto, esta se debe realizar diagrama a diagrama, no del proyecto entero, y para un proyecto de una practica o un PFC se debería de tener la opción de imprimir todos los diagramas así como el diagrama de objetivos.

Seria recomendable que apareciera un resumen del proyecto con toda la información del mismo, no solo numérica sino en texto, de los objetivos asociados a cada caso de uso, el número de casos de uso que hay, etc...

Estas son las principales dificultades que he encontrado en la aplicación R² y son las principales actuaciones futuras que se pueden llevar a cabo para mejorar su rendimiento, con vistas a poder ser empleada como herramienta didáctica en algunas asignaturas debido a su enorme potencial.

Herramienta de Análisis de Variabilidad

Hay que tener en cuenta que la herramienta empleada de Análisis de Variabilidad es el primer prototipo de esta herramienta.

Partiendo de este condicionante, se puede observar que la aplicación no permite ciclos y restringe las contribuciones entre objetivos funcionales, no funcionales y tareas. Estas contribuciones están limitadas a OR/XOR/AND.

Los enlaces cualitativos solo se permiten como correlaciones entre el modelo de objetivos funcionales y el de objetivos no funcionales.

Estas limitaciones son impuestas para facilitar la implementación del prototipo.

En términos de escalabilidad, esta solución basada en Excel tiene limitaciones debido a la propia herramienta (la longitud de la matriz), pero también impuestas por la complejidad del algoritmo. Por ejemplo, en este caso de análisis realizado en el dominio de la informática para discapacitados, con 55 requisitos funcionales y 24 requisitos no funcionales, relaciones XOR, AND, OR, se obtienen más de veintitrés mil variantes. Realizado el análisis en un AMD XP 1.7 Ghz y 512 Mb RAM ha requerido unos 45 minutos para computar todas las variantes. Hay que tener en cuenta que este cálculo hay que realizarlo cada vez que se cambian las correlaciones.

El cambio de prioridades no requiere tanto tiempo, pero debido a las limitaciones de Excel con rangos complejos, los datos son reordenados y para 7 requisitos no funcionales que se han cambiado las prioridades, la herramienta ha tardado más de 2 minutos en mostrar todas las tablas. Posiblemente estos problemas se puedan mejorar empleando otras estructuras diferentes a las que ofrece Excel.

Otra limitación y posible mejora es la visualización de las graficas y el diagrama de árbol resultante, a medida que las variantes aumentan. Aquí, se pueden emplear las funcionalidades de Excel como el zoom, pero aun así, en modelos muy complejos es muy difícil de poder verse con claridad como en este caso en el dominio tratado.

Todas estas limitaciones se deberían estudiar y ver la manera de poder solucionarlas, esto haría que la herramienta fuese aun mas interesante de lo que ya es de por si.

4.5.3.2. CONCLUSIONES

En nuestro ejemplo, utilizando la herramienta de análisis, se ha conseguido reutilizar alrededor del **70%** de casos de uso, lo que supone:

1. **Incremento de la calidad.** Al usar ya casos de uso bien definidos, en cada reutilización se pueden ir hallando defectos y depurándolos, esto hace que también aumente la fiabilidad.
2. **Ahorro importante de costes.** Al tener ya un modelado y análisis del dominio y un conjunto de casos de uso reutilizables para ese dominio, se pueden reducir parte de los costes asociados al estudio y análisis.
3. **Aumento de productividad.** Se invierte menos tiempo en planes, modelos, documentos, etc... para crear un sistema fiable. Se proporciona un mismo nivel o superior de funcionalidad al cliente con menos esfuerzo.

Si se hubiera realizado el comunicador empleando la herramienta de análisis, podríamos haber reutilizado el **100%** de los casos de uso, ya que todos estarían realizados y documentados para su posterior reutilización, en este caso se hubiera reducido aun mas los costes, aumentado la productividad y la velocidad de realización de la aplicación.

Puede ser que no se hayan encontrado todos los requisitos, ya que hay algunos requisitos de los interesados que son difíciles de identificar, tanto por el conocimiento tácito u oculto como por las diversas opiniones que se pueden obtener.

Por ello se necesita otra herramienta que sirva para identificar esos requisitos tácitos, esa herramienta es la que se desarrolla a continuación como una ampliación al Proyecto de Fin de Carrera, partiendo de la *Teoría de Constructos Personales*.

5. DESARROLLO RAPIDO DE UNA HERRAMIENTA: “REPERTORY GRID”

5.1. FUNDAMENTO TEORICO

El fundamento de esta herramienta se basa en la **Teoría de Constructos Personales** (TCP), que es una teoría psicológica constructivista propuesta por Nelly en 1995 que trata de explicar la visión del mundo por una persona por medio de Constructos. Un constructo es un elemento de conocimiento con dos polos opuestos. Según Nelly, el conjunto de Constructos de una persona y las relaciones entre ellos da la base de los juicios y comportamientos de dicha persona, puesto que le permiten hacer hipótesis y llegar a conclusiones.

Además Nelly desarrollo una técnica basada en esta teoría, el *Repertory Grid* o rejilla, para elicitar el conocimiento oculto de una persona. Esta técnica, que ha sido automatizada en el campo de la gestión del conocimiento, presenta un gran potencial para la elicitación de objetivos por permitir elicitar el conocimiento difícil de expresar y porque saca a la luz relaciones entre los distintos Constructos.

La técnica del *Repertory Grid* es básicamente una entrevista en la que el sujeto de la entrevista debe evaluar un conjunto de elementos sobre la base del conjunto de Constructos. Por, tanto, los elementos representan elementos de interés, que pueden ser tanto concretos como abstractos, y los constructos representan las maneras por las cuales los elementos son juzgados. Los resultados de la evaluación se muestran en una matriz de constructos (filas) y elementos (columnas).

En la tabla 6 hay un ejemplo de rejilla resumida a partir de un ejemplo.

		Elementos					
		Jose	Victor	Ana	Merche	Alberto	
Constructos	Honesto	2	4	2	1	1	Deshonesto
	Generoso	2	4	3	2	1	Avaro
	Simple	4	3	2	3	4	Complejo
	Atrevido	1	2	4	3	4	Prudente
	Motivado	1	2	4	3	4	Desmotivado

Existen numerosas técnicas para extraer un nuevo conocimiento a partir de una rejilla, por ejemplo, usando distancias entre vectores o aplicando lógica difusa (fuzzy logic). Sin embargo, la técnica que aplicamos será la propuesta por Ford, que permite obtener implicaciones parciales (y por tanto relaciones asimétricas) entre constructos de manera fácil de automatizar.

[illegible]

A partir de estas matrices binarias denominadas **alpha-planos** (tabla 7), se comparan los constructos para ciertos valores con la siguiente formula, donde X es el vector de elementos, y además ψ y π son valores de un constructo.

De tal forma, con este análisis, se obtienen reglas del tipo:

(Valor_constructo_1) Constructo1 **implica** (Valor_Constructo_2) Constructo2
[Grado de confirmación]

Que si damos un nombre a los valores de los constructos, por ejemplo extremadamente para el mayor valor quedaría:

Extremadamente Constructo_1 **implica** Extremadamente Constructo_2 []

Otra opción de análisis es agrupar valores de constructos, en ese caso se crearía una nueva matriz a comparar. En este caso, las reglas quedarían:

Al menos Algo Constructo_1 **implica** Al menos Algo Constructo_2 []

Donde se pueden distinguir dos características de las reglas, por un lado la precisión (utilizar un valor del constructo frente a un conjunto de valores) y el grado de verdad (el grado de confirmación) que nos permite hacernos una idea de la validez de la regla.

La aplicación de la TCP a la IR Orientada a Objetivos es la elicitación de conocimiento que es difícil expresar. Aplicando la técnica del *Repertory Grid* se obtienen reglas que relacionan constructos a partir de elementos. Por tanto, lo primero que se debe hacer es determinar que conceptos se van a emplear como constructos y cuales como elementos. Lo que se quiera relacionar debe ser un constructo, puesto que las reglas que se obtienen son relaciones entre constructos. Para más información acerca de la TCP y el Repertory Grid, consultar la bibliografía.

5.2. ANALISIS

A continuación se va a mostrar el desarrollo de la herramienta “Repertory Grid”, en sus fases de análisis, diseño e implementación.

El análisis se dedica a la comprensión y modelado de la aplicación y del dominio en el cual funciona. La entrada inicial de la fase de análisis es una descripción del problema que hay que resolver y proporciona una visión general conceptual del sistema propuesto. La salida del análisis es un modelo formal que captura los tres aspectos esenciales del sistema: los objetivos y sus relaciones, el flujo dinámico de control y la transformación funcional de datos sometidos a restricciones.

Esta aplicación esta basada en la teoría anteriormente expuesta, se quiere realizar un prototipo que implemente la técnica del *Repertory Grid*. Los requisitos son capturados gracias a entrevista con Bruno González Baixauli que es el experto del dominio que trabajara con dicha herramienta.

Los requisitos que se pidieron fueron:

- Realizar una aplicación que genere la rejilla con los constructos y los elementos, para introducir sus valores.
- A continuación se creasen los alpha-planos, tanto de los constructos solos como agrupaciones de constructos.
- Seguidamente que pidiese el grado de confirmación y mostrase por pantalla solo las relaciones que fuesen mayor que ese grado de confirmación.
- También se pidió que se pudieran cambiar los valores de los constructos.
- Por ultimo se pensó en la posibilidad de realizar la herramienta en otro lenguaje, preferiblemente el inglés.

5.2.1. DIAGRAMA DE OBJETIVOS

El diagrama de objetivos se obtiene de los requisitos anteriormente explicados.

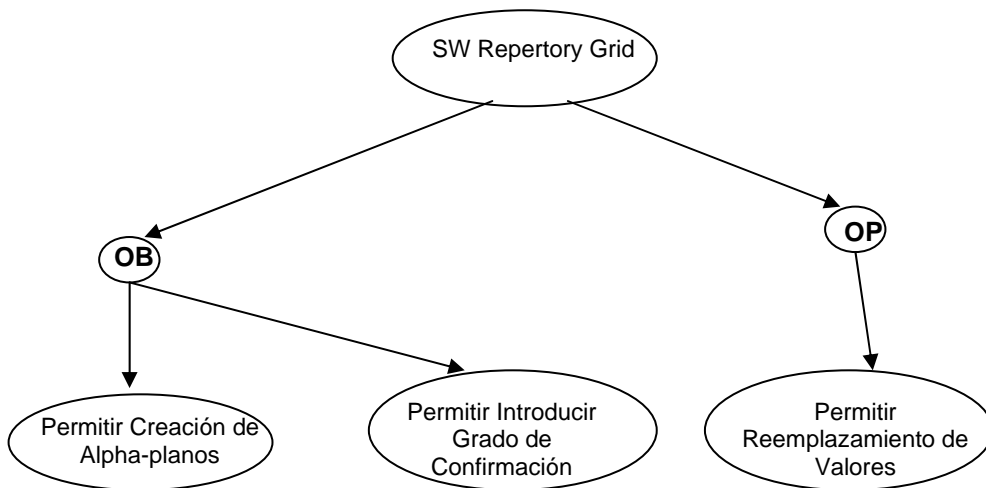


Diagrama 6: Diagrama de Objetivos de la herramienta.

5.2.2. DEFINICION DE ACTORES

En sistemas más complejos, un actor puede representar varios papeles. En el caso de esta aplicación el único actor considerado es el *experto del dominio* que es el que desea trabajar con la herramienta.

5.2.3. DIAGRAMA DE CASOS DE USO

Los casos de uso empleados en la herramienta han sido los siguientes:

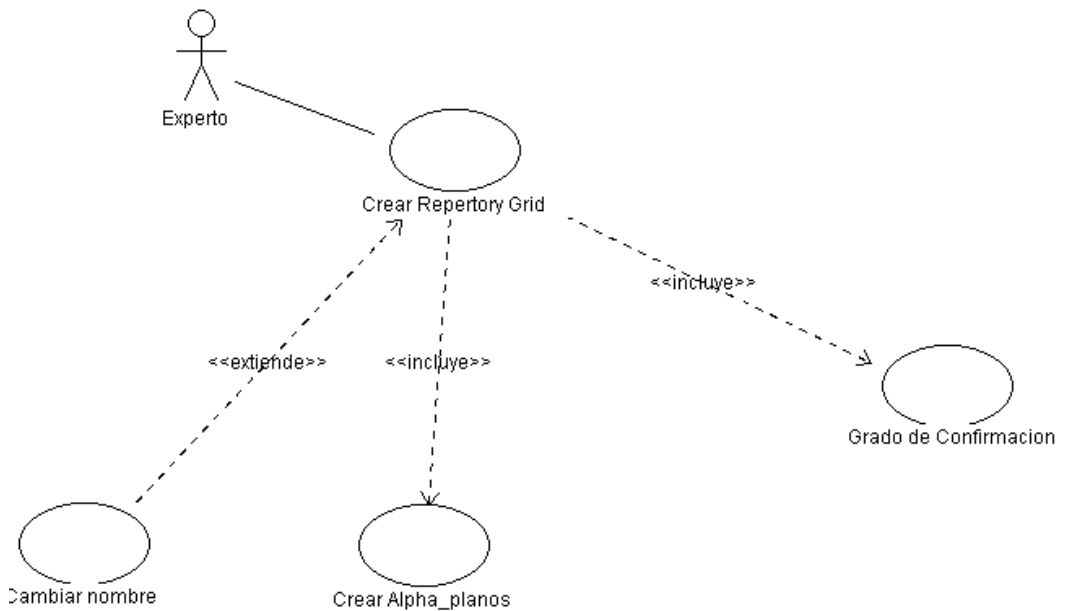


Figura 10: Diagrama de Casos de Uso de la herramienta Repertory Grid.

5.2.4. DESCRIPCION DE LOS CASOS DE USO

El comportamiento de un caso de uso se puede especificar describiendo un flujo de eventos de forma textual, lo suficientemente claro para que una persona ajena al sistema lo entienda fácilmente. A la hora de describir este flujo de eventos, se debe incluir cómo y cuando empieza y termina el caso de uso en cuestión, cuando interactúa con los actores y que objetivos intercambian, el flujo básico y los alternativos de comportamiento.

A medida que se obtiene una mejora en la comprensión de los requisitos del sistema, estos flujos de eventos se especifican gráficamente mediante los diagramas de interacción. Normalmente, se utiliza un diagrama de secuencia para especificar el flujo principal de un caso de uso.

RF- 1	CREAR REPERTORY GRID	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	SW Repertory Grid	
Descripción	El sistema deberá permitir utilizar la herramienta <i>Repertory Grid</i> para crear las relaciones	
Precondición		
Secuencia Normal	Paso	Acción
	1	El experto desea utilizar la herramienta
	2	El sistema le pide al experto, que introduzca el numero de constructos, elementos, y el valor máximo
	3	El alumno introduce el numero de constructos, elementos y el valor máximo
	4	El sistema muestra la rejilla para que se puedan introducir los datos
	5	El experto introduce los datos en la rejilla
	6	El sistema crea los alpha-planos
	7	El experto introduce un grado de confirmación
	8	El experto desea cambiar los nombres
Postcondición	El experto ha utilizado la herramienta para conseguir los resultados	
Excepciones	Paso	Acción
	3.a	Si el experto no ha introducido todos los valores, el sistema muestra un mensaje de error y los vuelve a pedir
	3.b	Si el numero máximo introducido es par, el sistema muestra un mensaje de error y vuelve a pedir los valores
	5.a	Si el experto ha dejado un numero en blanco, el sistema muestra un mensaje de error y a continuación vuelve a pedir los datos de la rejilla
	5.b	Si el experto ha introducido un numero mayor que el máximo, el sistema muestra un mensaje de error y a continuación vuelve a pedir los datos
Comentarios		

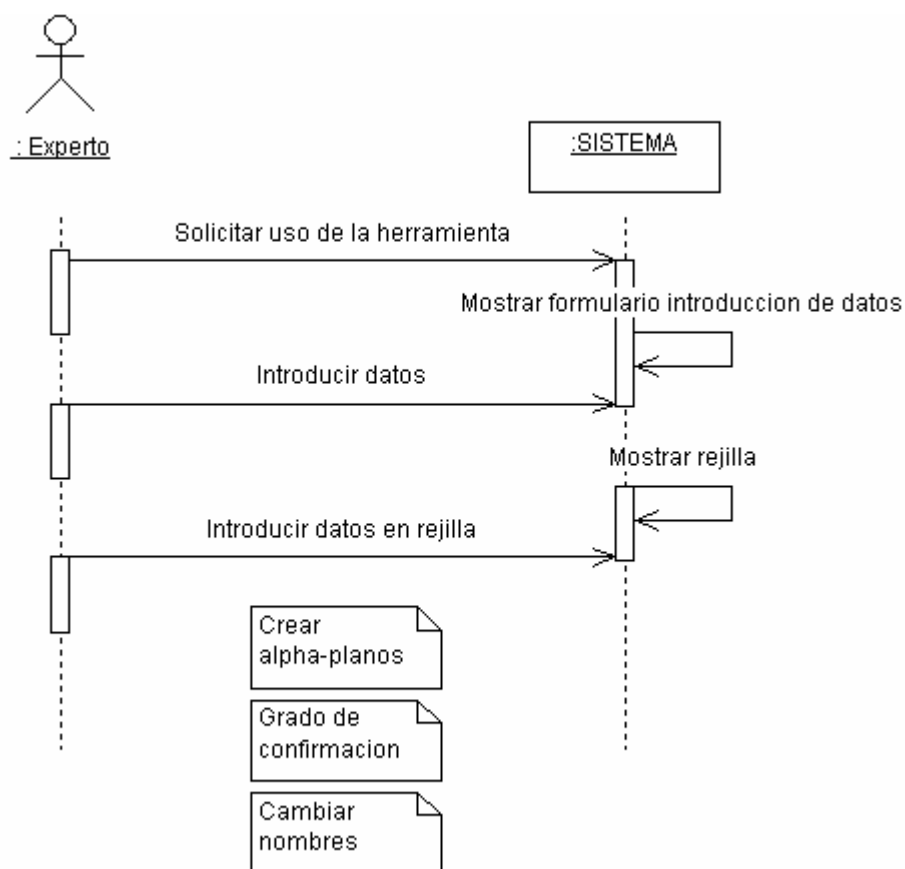


Figura 11: Diagrama del caso de uso Crear Repertory Grid.

RF- 2	CREAR ALPHA-PLANOS	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	Permitir Creación alpha-planos	
Descripción	El sistema deberá crear los alpha-planos correspondientes a los valores introducidos	
Precondición		
Secuencia Normal	Paso	Acción
	1	El experto desea crear los alpha-planos
	2	El sistema crea cada alpha-plano con los valores correspondientes
Postcondición	El experto ha introducido el grado de confirmación	
Excepciones	Paso	Acción
Comentarios		

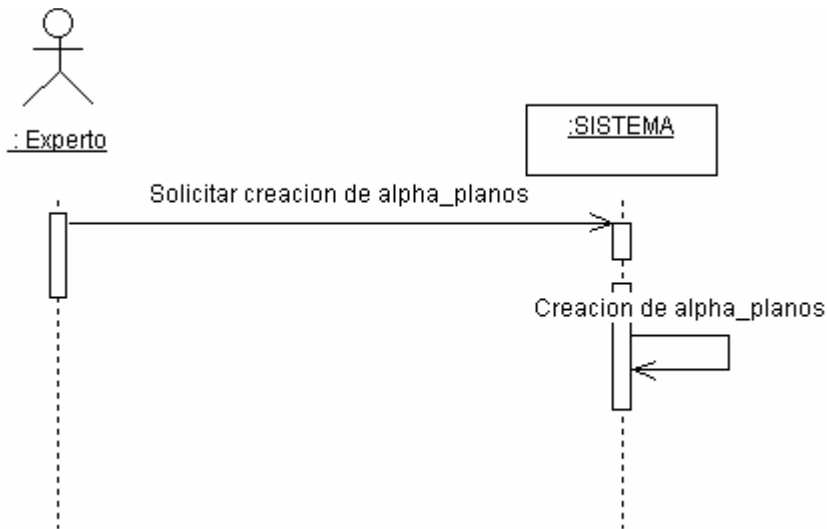


Figura 12: Diagrama del Caso de Uso Crear alpha-planos.

<i>RF- 2</i>	GRADO DE CONFIRMACION	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	Permitir introducir Grado de Confirmación	
Descripción	El sistema deberá permitir introducir el grado de confirmación para calcular las relaciones	
Precondición		
Secuencia Normal	Paso	Acción
	1	El experto desea introducir el grado de confirmación
	2	El sistema pide al experto el grado de confirmación
	3	El experto introduce el grado de confirmación
	4	El sistema muestra por pantalla los resultados
Postcondición	El experto ha introducido el grado de confirmación	
Excepciones	Paso	Acción
	3	Si el experto no ha introducido el grado de confirmación, el sistema muestra un mensaje de error y vuelve a pedir el valor
Comentarios		

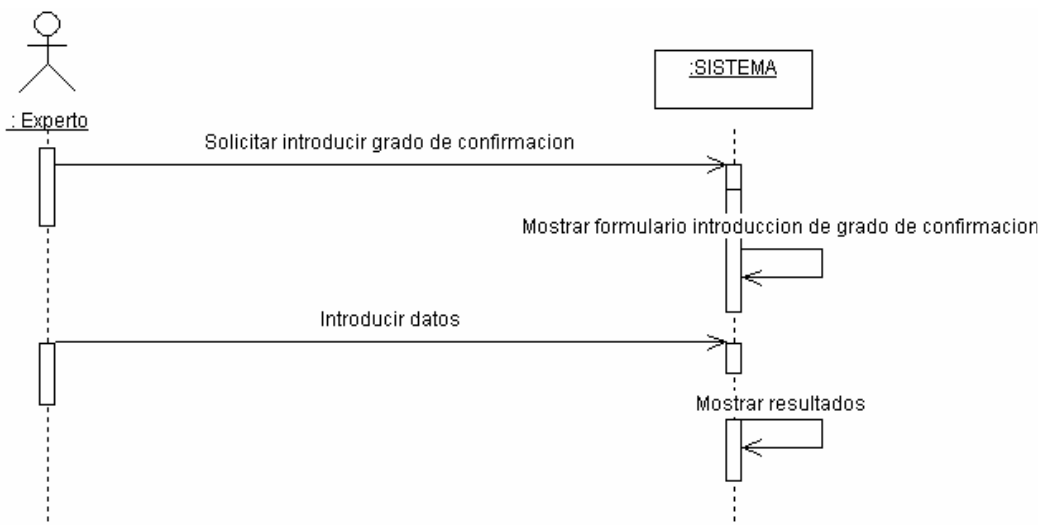


Figura 13: Descripción del caso de uso Grado de confirmación.

RF- 3	CAMBIAR NOMBRE << <u>extend RF-1</u>>>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	Permitir reemplazamiento de valores	
Descripción	El sistema deberá permitir cambiar el nombre de los valores de los constructos en las relaciones.	
Precondición		
Secuencia Normal	Paso	Acción
	1	El experto desea cambiar los nombres de los constructos
	2	El sistema muestra el nombre del valor de los constructor asociados a la herramienta
	3	El experto va introduciendo los nombres nuevos por los que desea cambiar los valores
	4	El sistema va cambiando los valores introducidos en el resultado
Postcondición	El experto ha cambiado el nombre	
Excepciones	Paso	Acción
	3	Si el experto no introduce el valor por el que desea cambiar los nombres, el sistema muestra un mensaje de error y vuelve a pedir el valor
Comentarios		

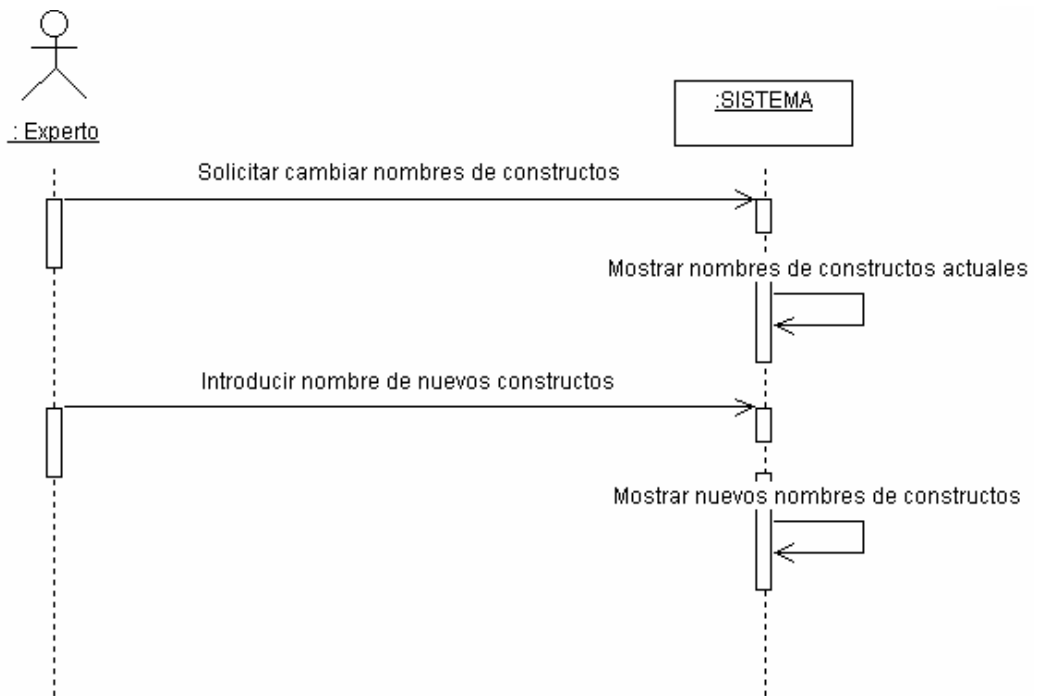


Figura 14: Diagrama de caso de uso Cambiar nombre.

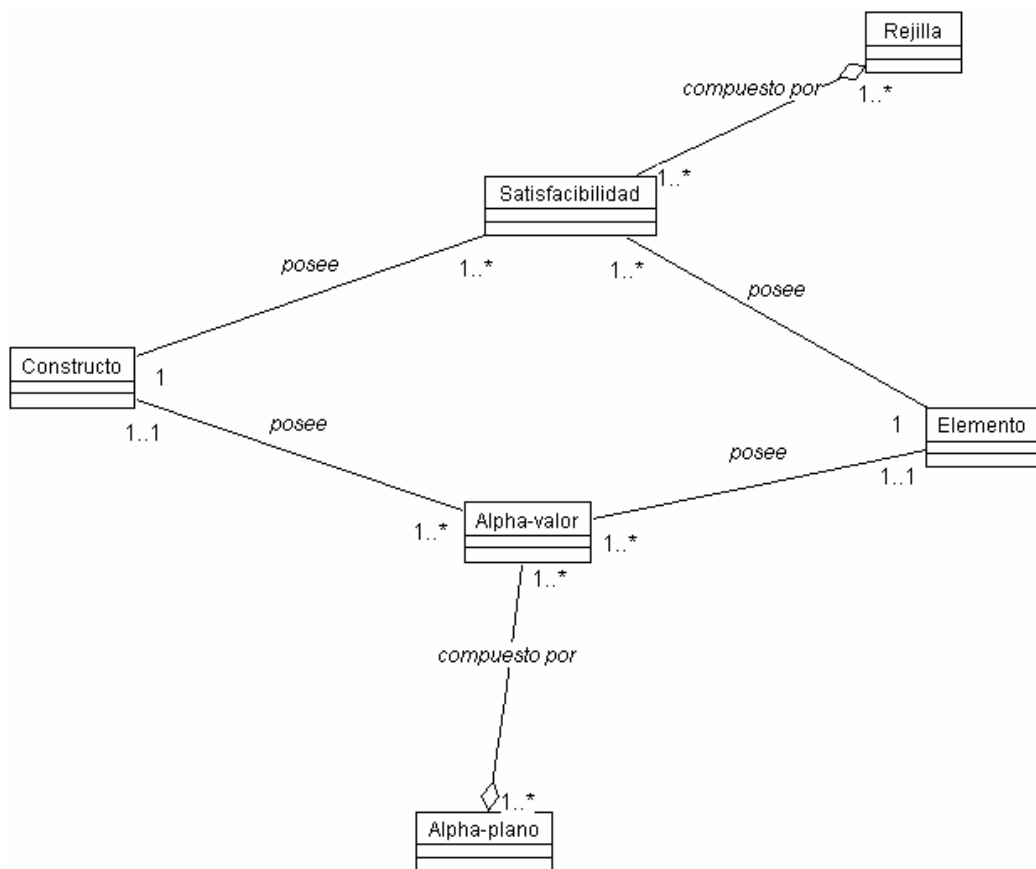
5.2.5. DIAGRAMA INICIAL DE CLASES

El modelo de objetos muestra la estructura estática de datos correspondientes al sistema del mundo real, y la organiza en segmentos manejables describiendo clases de objetos del mundo real, y sus relación entre si. Lo más importante es la organización a más alto nivel del sistema, en clases conectadas mediante asociaciones.

Atendiendo a los fundamentos teóricos vistos anteriormente, se pueden observar las siguientes clases:

La **Rejilla**, de entrada de datos que esta formada por un conjunto de **Constructos**, **Elementos** y las relaciones entre ambos (**Satisfacibilidad**).

Los **Alpha-planos**, creados también por una serie de **Constructos**, **Elementos**, y relaciones entre ambos (**alpha-valores**), en este caso esos valores serán 1 o 0.



5.3. DISEÑO

La fase de diseño es una fase intermedia entre la vista abstracta de un sistema y la implementación real de este. Los productos de esta fase pueden ser más cercanos a una visión abstracta o una visión implementable.

5.3.1. CASOS DE USO

RF- 1	CREAR REPERTORY GRID	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	SW Repertory Grid	
Descripción	El sistema deberá permitir utilizar la herramienta <i>Repertory Grid</i> para crear las relaciones.	
Precondición		
Secuencia Normal	Paso	Acción
	1	El experto abre la herramienta.
	2	El sistema abre el formulario principal.
	3	El experto introduce el número de constructos, elementos y el valor máximo.
	4	El sistema guarda los valores introducidos en la hojal.
	5	El sistema muestra la rejilla para que se puedan introducir los datos
	6	El sistema cambia el color de la rejilla según el número de valores introducidos.
	7	El sistema cierra el formulario principal.
	8	El experto introduce los datos en la rejilla.
	9	Se realiza el caso de uso: Crear alpha-planos <<RF-2>>
	10	Se realiza el caso de uso: Grado de Confirmación <<RF-3>>
	11	Pt. Extensión: cambio de nombre.
Postcondición	El experto ha utilizado la herramienta para conseguir los resultados.	
Excepciones	Paso	Acción
	3.a	Si el experto no ha introducido todos los valores, el sistema muestra un mensaje de error y los vuelve a pedir.
	3.b	Si el número máximo introducido es par, el sistema muestra un mensaje de error y vuelve a pedir los valores.
	8.a	Si el experto ha dejado un número en blanco, el sistema muestra un mensaje de error y a continuación vuelve a pedir los datos de la rejilla.

	8.b	Si el experto ha introducido un número mayor que el máximo, el sistema muestra un mensaje de error y a continuación vuelve a pedir los datos.
Comentarios		

RF- 2	CREAR ALPHA-PLANOS	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	Permitir Creación alpha-planos	
Descripción	El sistema deberá permitir crear los alpha-planos como se describe a continuación	
Precondición		
Secuencia Normal	Paso	Acción
	1	El experto abre el formulario de fuerza
	2	El sistema crea las tablas
	3	El sistema configura las tablas
	4	El sistema va introduciendo los valores correspondientes
	5	El sistema cierra el formulario
Postcondición	El experto ha creado los alpha-planos	
Excepciones	Paso	Acción
Comentarios	Este caso de uso creará todos los alpha-planos, en el diagrama de secuencia se muestra solo un ejemplo, ya que el numero de alpha-planos dependerá del numero máximo	

RF- 3	GRADO DE CONFIRMACION	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados		
Descripción	El sistema deberá permitir introducir el grado de confirmación para calcular las relaciones	
Precondición	Permitir introducir Grado de Confirmación	
Secuencia Normal	Paso	Acción
	1	El experto abre el formulario analizar
	2	El experto introduce el valor de la fuerza
	3	El sistema crea la hoja de potencia
	4	El sistema va completando los arrays con los valores de las filas de las celdas de cada constructo de cada alpha-plano
	5	El sistema calcula el array de la intersección de los arrays de datos
	6	El sistema calcula la ocurrencia del primer array de la comparación
	7	El sistema calcula la ocurrencia del array resultado de la intersección de los arrays del paso 5
	8	El sistema muestra el valor en la hoja de potencia
	9	El sistema cierra el formulario analizar
	10	El sistema ordena los resultados
Postcondición		
Excepciones	Paso	Acción
	3	Si el experto no ha introducido el grado de confirmación, el sistema muestra un mensaje de error y vuelve a pedir el valor
Comentarios	Este caso de uso, va completando los arrays que luego se utilizaran para la comparación y para ver el grado de confirmación entre constructos	

RF- 4	CAMBIAR NOMBRE << <u>extend RF-1</u> >>	
Versión	1.0	
Autores	Jose Carlos Jiménez Sánchez	
Fuentes		
Objetivos asociados	Permitir reemplazamiento de valores	
Descripción	El sistema deberá permitir cambiar el nombre de los valores de los constructos en las relaciones.	
Precondición		
Secuencia Normal	Paso	Acción
	1	El experto abre el formulario reemplazar
	2	El sistema busca el nombre de todas las hojas y las muestra
	3	El sistema selecciona todas las celdas de la hoja "potencia" donde se llevara a cabo el reemplazamiento
	4	El experto introduce los valores por los que desea cambiar
	5	El sistema reemplaza el valor anterior por el nuevo valor que hemos escrito
	6	El sistema cierra el formulario reemplazar
	7	El sistema ordena los resultados
Postcondición	El experto ha cambiado el valor de los constructos	
Excepciones	Paso	Acción
	3	Si el experto no introduce el valor por el que desea cambiar los nombres, el sistema muestra un mensaje de error y vuelve a pedir el valor
Comentarios		

5.3.2. IDENTIFICACIÓN DE LAS CLASES

Las clases suelen corresponderse con sustantivos o entidades físicas. Aunque hay que evitar las estructuras de implementación propias de la computadora.

Una vez obtenidas las clases, hay que descartar todas aquellas no necesarias e incorrectas, eliminando:

- Clases redundantes: Cuando haya dos clases que expresen la misma información, hay que mantener solamente la que tenga el nombre más descriptivo.
- Clases irrelevantes: Si una clase tiene muy poco o nada que ver con el problema, debe ser eliminada. Así, habrá que emplear nuestro propio criterio, porque esa clase en otro contexto podría resultar importante.
- Clases vagas: Una clase debe ser algo específico, por lo que habrá que evitar clases con límites mal definidos o con un ámbito excesivo.

- Atributos: los nombres que describen objetos individuales suelen recalificarse como atributos.
- Operaciones: Cuando un nombre describe una operación que se aplica a objetos y que no es manipulada en si, entonces no debe ser una clase. Sin embargo, operaciones con características propias deben ser modeladas como clases.
- Rol: El nombre de una clase no debe reflejar el papel que desempeña en una asociación.
- Estructuras de implementación: Deben ser eliminadas del modelo de análisis las estructuras extrañas al mundo real, porque quizás se necesiten en una fase posterior del diseño.

A continuación se muestra el diagrama de clases junto con la descripción de los atributos y métodos de cada clase.

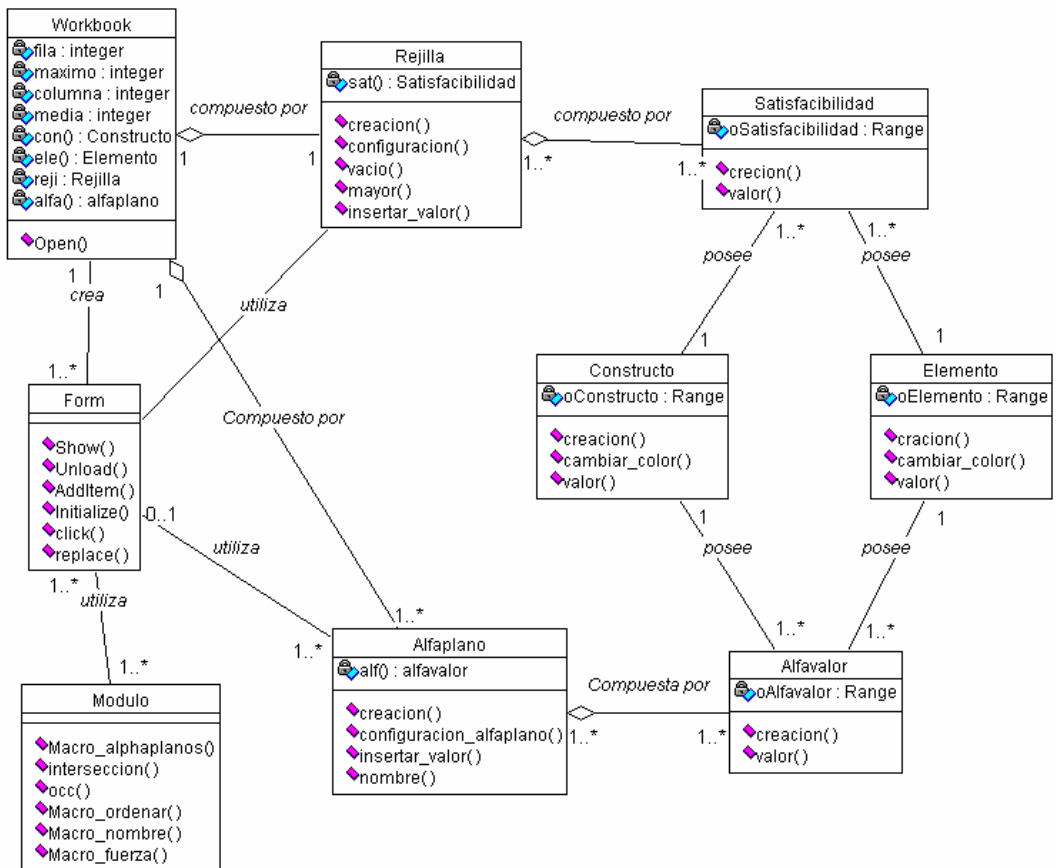


Figura 15: Diagrama de Clases.

5.3.3. DESCRIPCION DE LAS CLASES

En este diccionario de datos aparecen únicamente los atributos, propiedades y métodos de las clases que se han creado, las clases propias de Excel no se han descrito debido a la enorme cantidad

de atributos que poseen, si se desea obtener mas información, Excel muestra todos los métodos y atributos de todas las clases, en este apartado solo se muestran los atributos y métodos que se van a utilizar en la herramienta.

- **Clase Workbook:**

Es la clase principal de Excel, encargada de arrancar el formulario principal de la aplicación.

- Atributos:

Fila: Guarda el valor de los constructos introducidos por el experto.

Máximo: Guarda el valor máximo que se puede introducir en la rejilla.

Columna: Guarda el valor de los elementos introducidos.

Media: Guarda el valor medio del máximo introducido anteriormente.

Con (): Array de objetos de tipo Constructo.

Ele (): Array de objetos de tipo Elemento.

Reji: Objeto de tipo Rejilla.

Alfa (): Array de objetos de tipo Alfaplano.

- Métodos:

Open(): Método que abre el formulario principal.

- **Clase Rejilla:**

Es la clase que contiene la tabla donde se introducen los datos.

- Atributos:

Sat (): Array de objetos de tipo Satisfacibilidad.

- Métodos

creacion (nombre : string, fila : integer, columna : integer, aux : boolean) : Método de creación de la rejilla, se le pasa como argumentos el nombre de la rejilla, el numero de filas, columnas, así como un valor para saber si la rejilla ya existe.

configuracion (fila : integer, columna : integer) : Método que configura la rejilla (color, ...).

vacío () : boolean: Función que comprueba que no se han introducido valores nulos en la rejilla.

mayor (numero : integer) : boolean: Función que comprueba que no se han introducido valores mayores que el máximo.

insertar_valor (a : integer, b : integer) : integer: Función que nos devuelve el valor del objeto Satisfacibilidad que deseamos conocer.

- **Clase Alfaplano:**

Es la clase que contiene su tabla correspondiente con sus alfavalores.

- Atributos:

Alf (): Array de objetos de tipo Alfavalor.

- Propiedades:

insertar_valor (a As Integer, b As Integer) : Propiedad que devuelve o inserta un valor en la posición del alfaplano especificada.

nombre (): Propiedad que devuelve el nombre del alfaplano.

- Métodos

creacion (nombre : string, fila : integer, columna : integer,

aux : boolean): Método de creación del alfaplano, pasa como argumentos el nombre del alfaplano, el numero de filas, columnas, así como un valor para saber si el alfaplano ya existe.
configuracion_alfaplano (nombre : string, fila : integer, columna : integer) : .Método que configura el alfaplano (color, ...).

- **Clase Constructo:**

Es la clase que contiene la información de un constructo.

- Atributos:

oConstructo : Range: Objeto de tipo Rango.

- Propiedades:

valor (): Propiedad que devuelve o inserta el valor del constructo.

- Métodos:

creacion (i : integer): Método que crea el objeto de tipo Constructo.

cambiar_color (): Método que cambia el color del constructo en la rejilla.

- **Clase Elemento:**

Es la clase que contiene la información de un elemento.

- Atributos:

oElemento : Range: Objeto de tipo Rango.

- Propiedades:

valor (): Propiedad que devuelve o inserta el valor del elemento.

- Métodos:

creacion (j : integer): Método que crea el objeto de tipo Elemento.

cambiar_color (): Método que cambia el color del elemento en la rejilla.

- **Clase Satisfacibilidad:**

Es la clase que contiene la información de un valor de satisfacibilidad.

- Atributos:

oSatisfacibilidad : Range: Objeto de tipo Rango.

- Propiedades:

valor (): Propiedad que devuelve o inserta el valor del Satisfacibilidad.

- Métodos:

creacion (i : integer, j : integer): Método de creación del objeto Satisfacibilidad.

- **Clase Alfavalor:**

Es la clase que contiene la información de un alfavalor.

- Atributos:

oAlfavalor : Range: Objeto de tipo Rango.

- Propiedades:

valor (): Propiedad que devuelve o inserta el valor del alfavalor.

- Métodos:

creacion (nombre : string, i : integer, j : integer): Método de creación de un objeto alfavalor.

- **Clase Form:**

Es la clase formulario.

- Atributos:

Sin atributos.

➤ Métodos:

Show(): Muestra el formulario indicado.

Unload(): Cierra el formulario indicado.

Add Item(string a): añade uno de los valores de los constructos al formulario.

Initialize(): Método que realiza las operaciones indicadas en el antes de cargar el formulario.

Click(): Método que realiza las acciones descritas en el cuando se pulsa el botón del formulario.

Replace(): Método que realiza el reemplazo de los valores de las tablas por los valores seleccionados.

• **Clase Modulo:**

Es la clase que contiene las funciones de llamadas a los formularios y cálculos entre arrays.

➤ Atributos:

Sin atributos

➤ Métodos:

Interseccion(array a, array b):array: Función que nos calcula el array de la intersección entre 2 arrays..

Occ(array a):Integer: Función que nos devuelve el valor de la suma de los 1's del array que le pasamos.

Macro_fuerza(): Función que abre el formulario analizar.

Macro_ordenar(): Función que ordena los valores obtenidos como resultado.

Macro_Alphaplanos(): Función que abre el formulario de Fuerza.

Macro_nombre(): Función que abre el formulario de Reemplazar valores.

5.3.4. DIAGRAMAS DE SECUENCIA

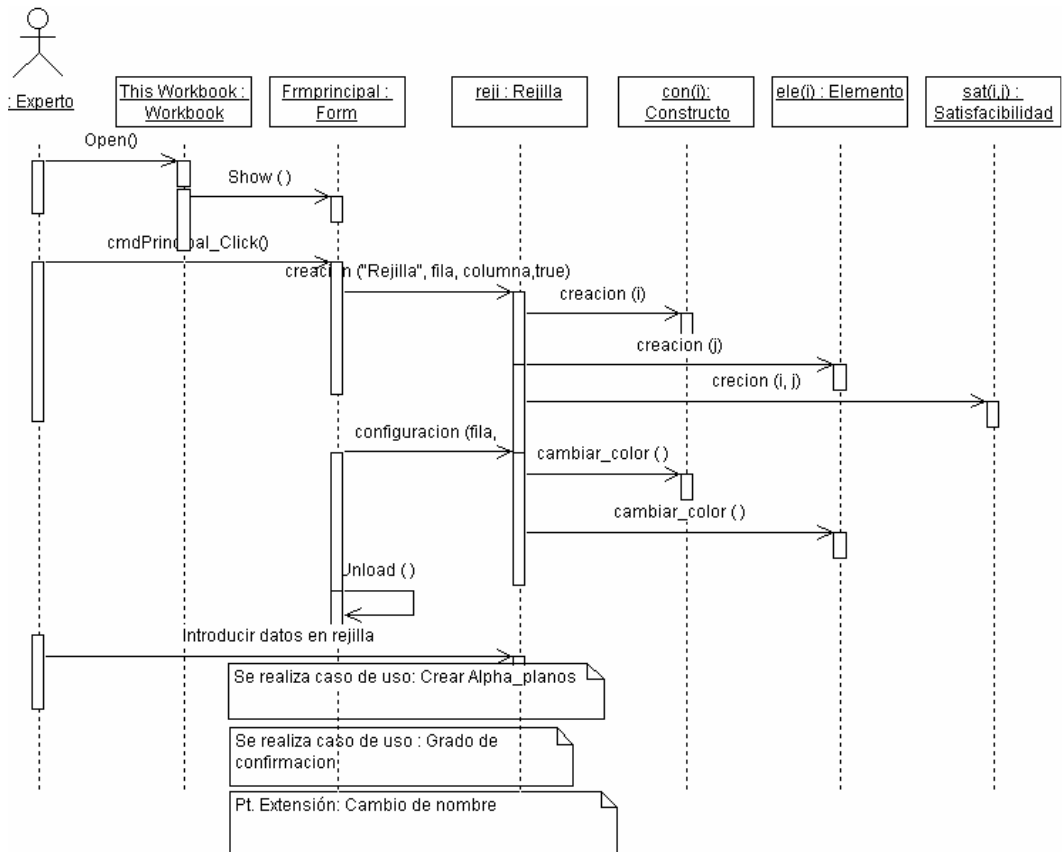


Figura 16: Diagrama de secuencia Crear Repertory Grid

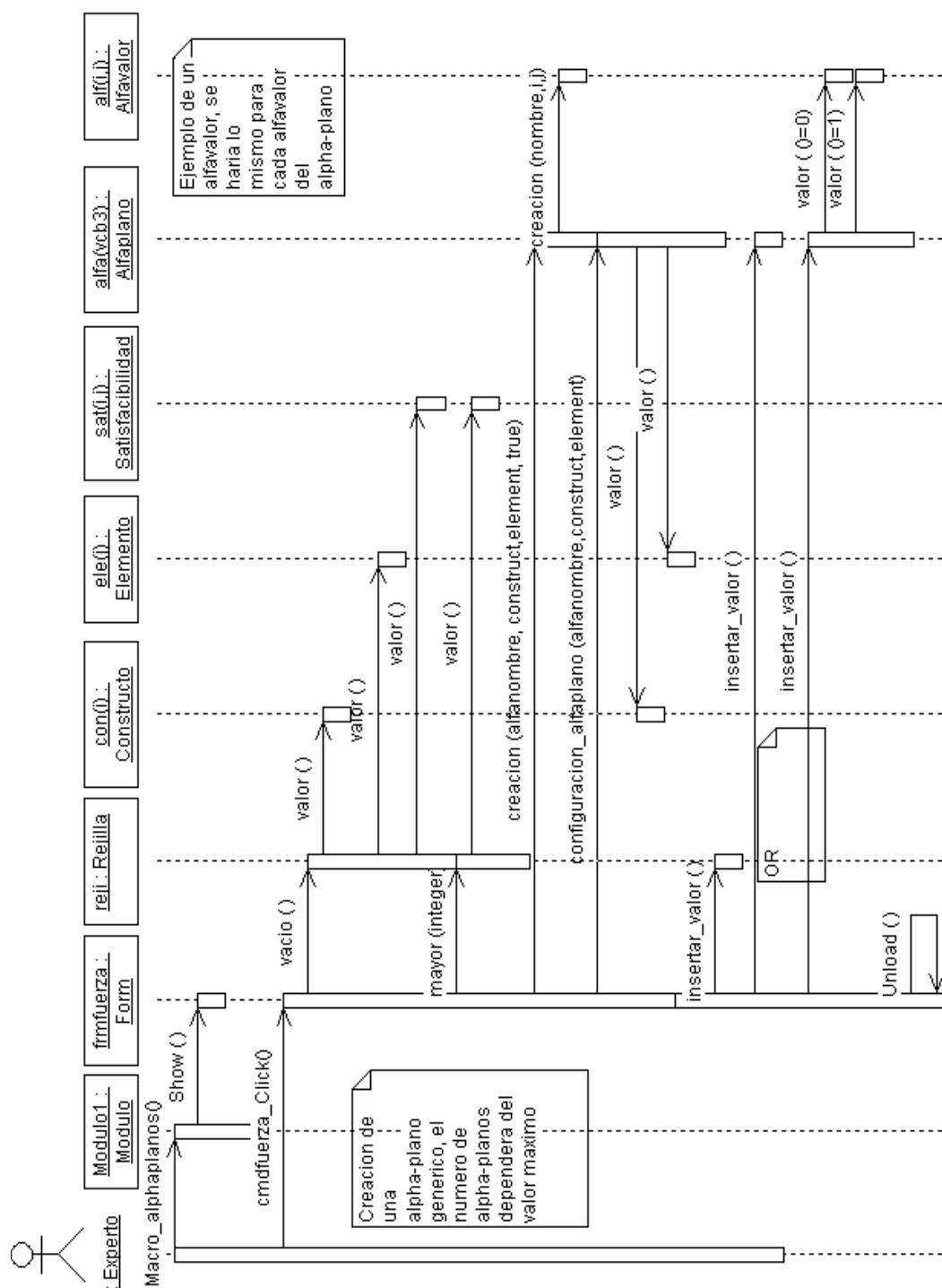


Figura 17: Diagrama de secuencia Crear Alpha-planos

[illegible]

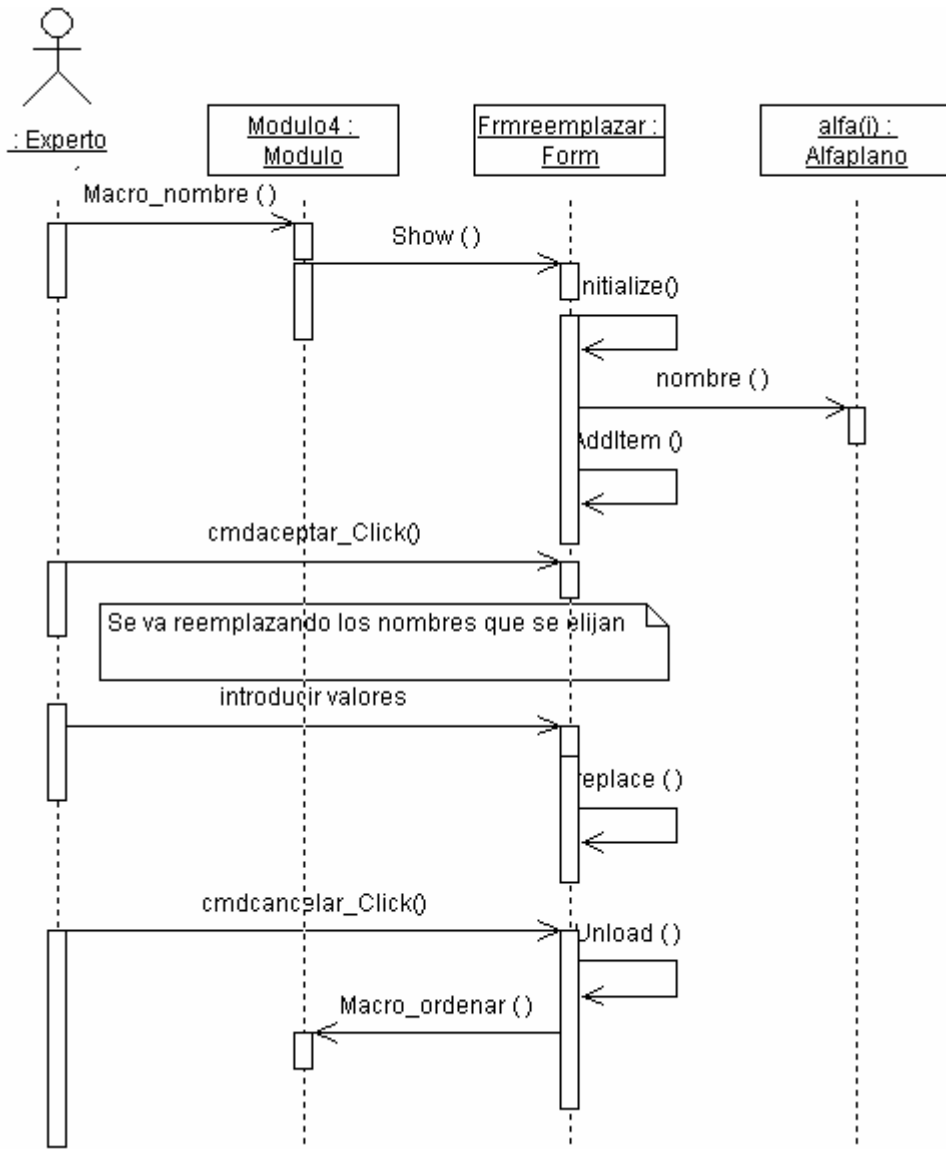


Figura 19: Diagrama de secuencia Cambiar nombre

5.3.5. DIAGRAMAS DE COLABORACION

En el diagrama de colaboración se muestra el orden de las llamadas en el sistema. Se utiliza un diagrama para cada caso de uso a representar. Se genera a partir del Diagrama de Secuencia correspondiente, los diagramas de secuencia de la aplicación se encuentran en el CD del proyecto, en el archivo de Rational Rose.

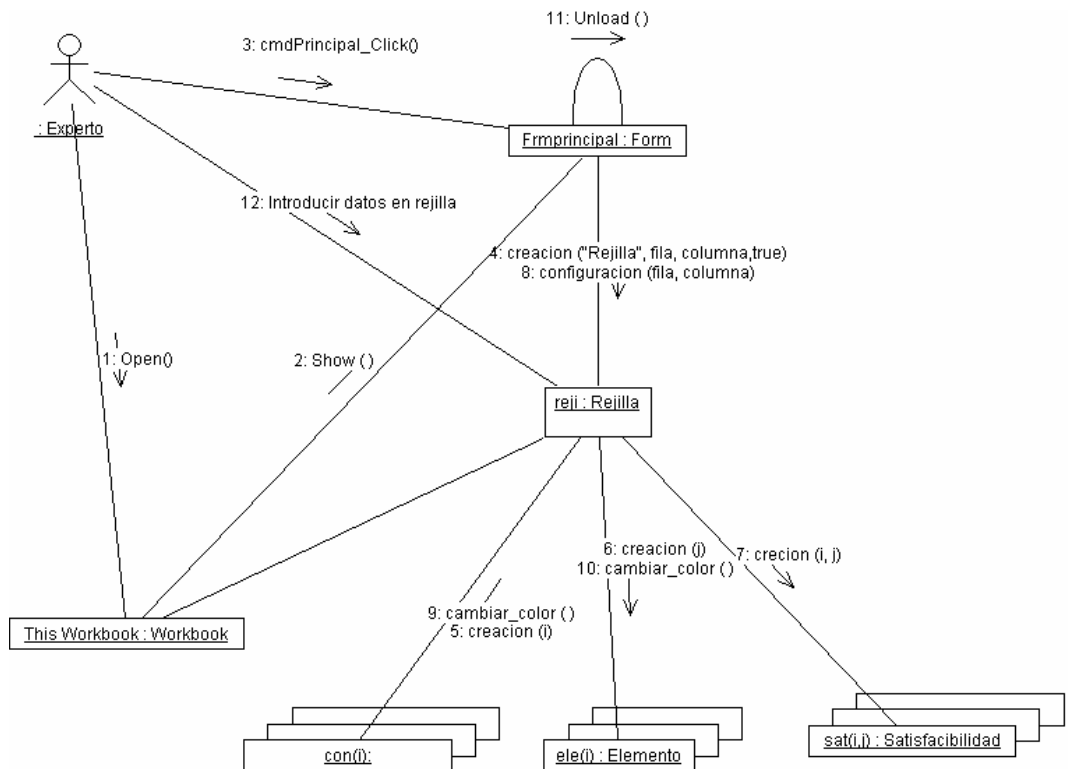


Figura 20: Diagrama de colaboración Crear Repertory Grid

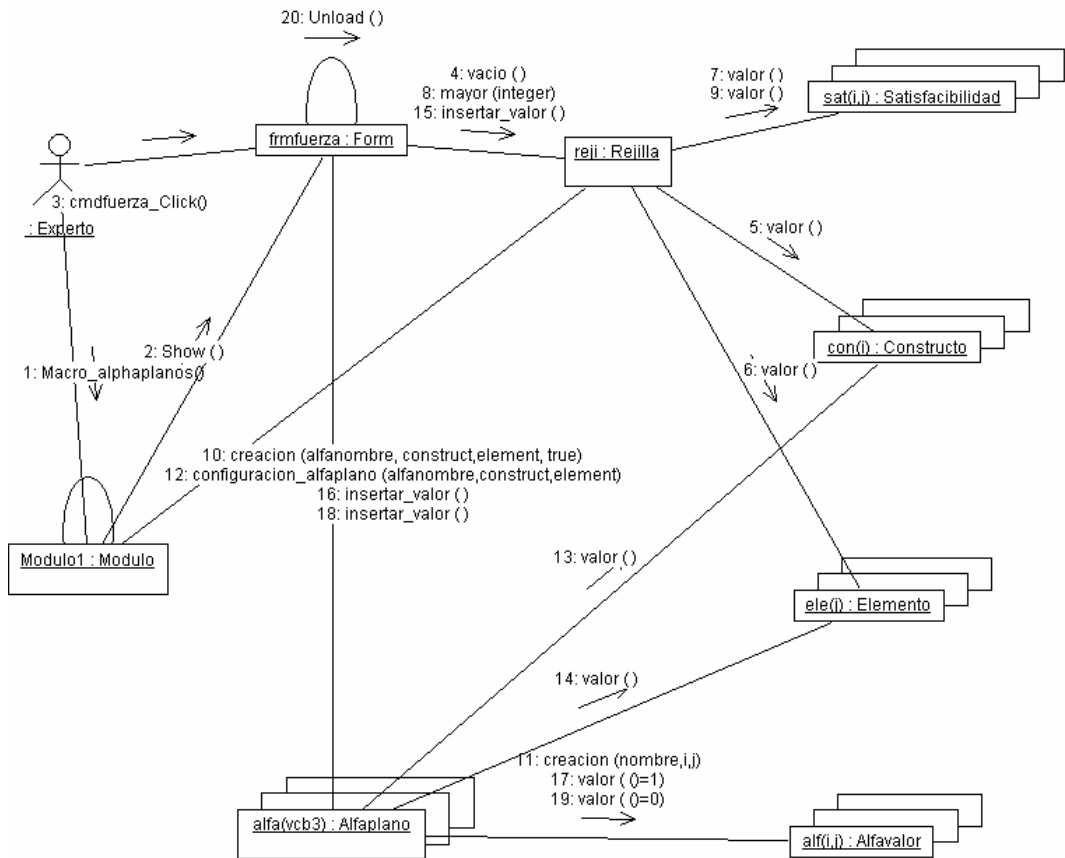


Figura 20: Diagrama de colaboración Crear Alpha-planos

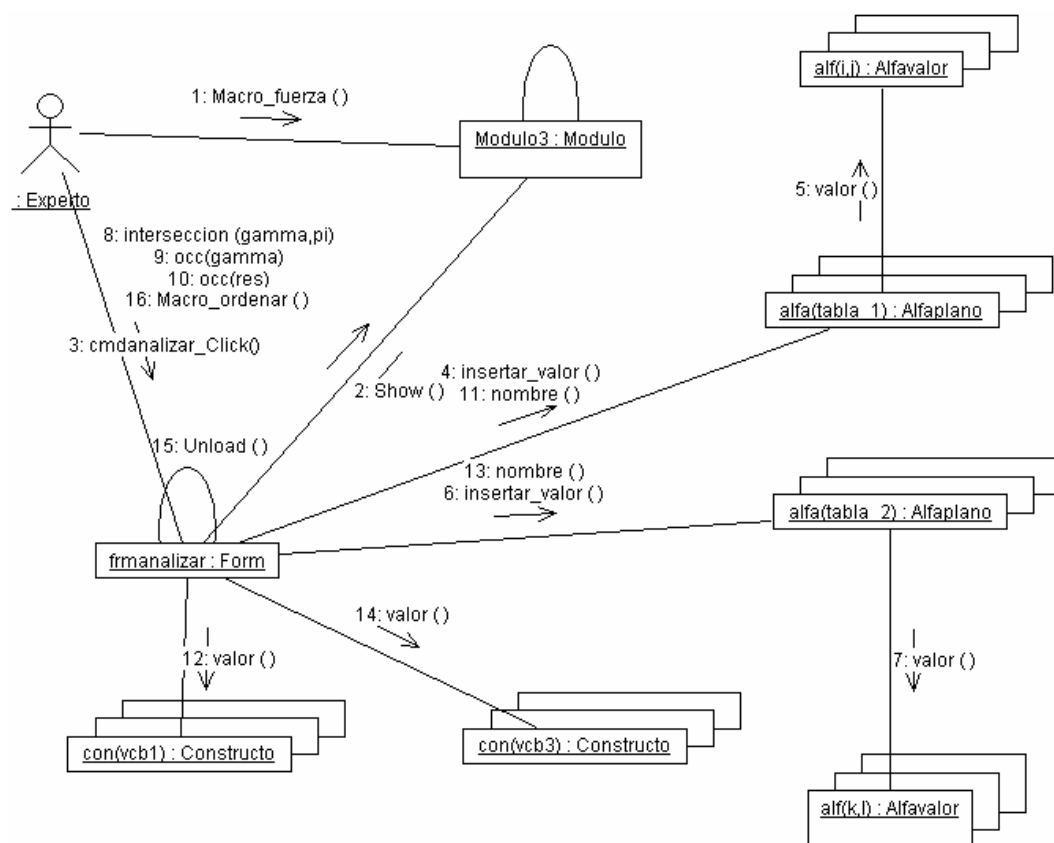


Figura 21: Diagrama de colaboración Grado de Confirmación

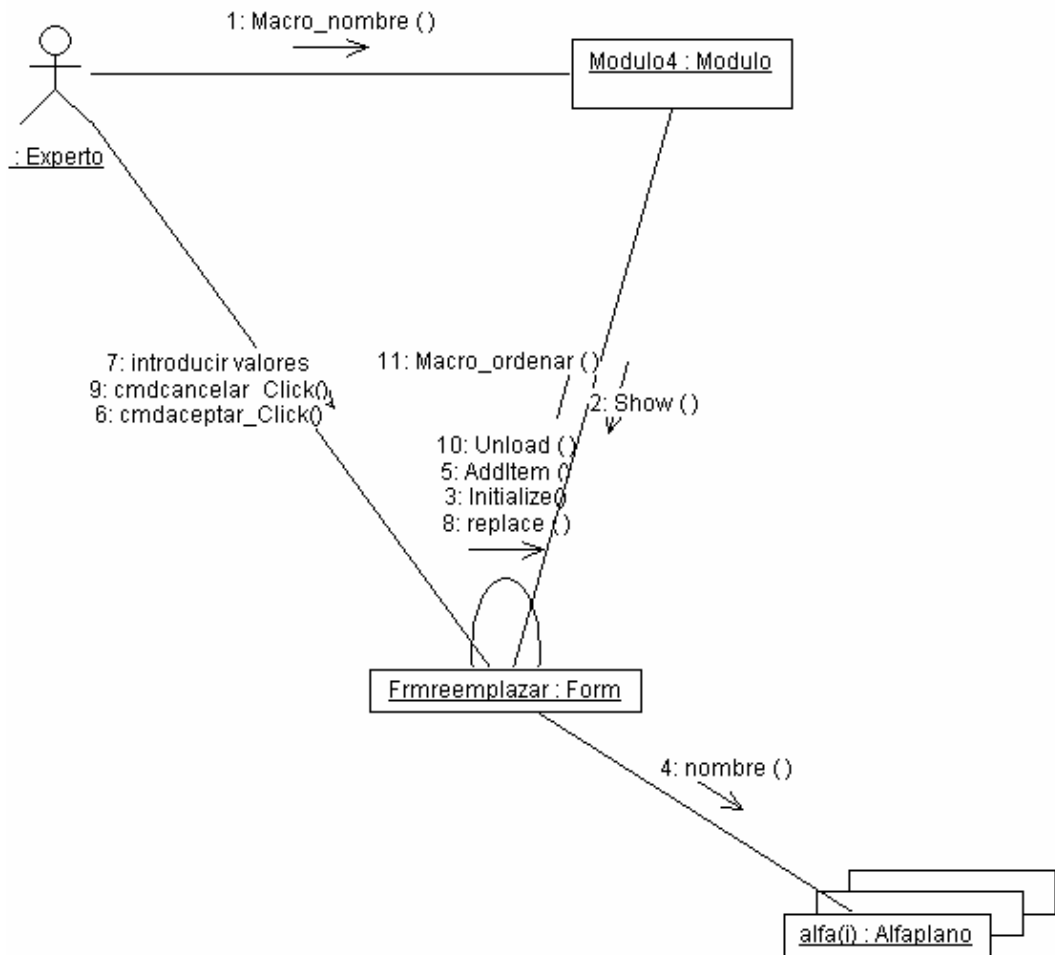


Figura 22: Diagrama de colaboración Cambiar nombre

5.4. IMPLEMENTACION

Durante la fase de Implementación del sistema, las clases de objetos y las relaciones desarrolladas durante el diseño se traducen a un lenguaje de programación concreto.

5.4.1. LENGUAJE DE PROGRAMACIÓN SELECCIONADO

Desde el principio del desarrollo de la aplicación se ha tenido claro que la mejor manera de programar esta herramienta era utilizando Excel y Visual Basic for Applications. Debido al trabajo que se debía realizar con tablas, la estructura de Excel en libro hojas y celdas se adaptaba perfectamente a lo que se pretendía construir.

Aunque, en un principio, mi nivel de utilización de Excel no sobrepasaba el nivel usuario, gracias a la formación que obtuve en la asignatura practicas en empresa realizadas en la empresa

Michelín España Portugal S.A. en la que trabaje con VBA y Access 2000, un curso anterior de Visual Basic 6.0 realizado por el G.U.I. y la inestimable colaboración de Bruno González Baixauli unido a algunas herramientas didácticas como manuales, ejemplos etc.. me sirvieron para adquirir una base con la que poder desarrollar esta herramienta.

Además la potencia que ofrece Excel y la rapidez de desarrollo es muy alto comparado con otras alternativas que se estudiaron en un principio como podía ser el caso de Java para esta aplicación.

La posibilidad, además, de poder añadir otras aplicaciones como pueden ser bases de datos como Access o editores de texto como Word hace tremendamente completa la utilización de Visual Basic para aplicaciones como lenguaje de programación, además de la facilidad para crear formularios de inserción de datos para alimentar la herramienta.

Otra de las cosas por las que se decidió realizar la herramienta en Excel y VBA fue la existencia, como se ha podido ver anteriormente en este proyecto, de herramientas ya creadas con estos lenguajes en el mismo ámbito de aplicación.

El principal problema que se puede ver al utilizar estas herramientas es la utilización de licencias, y que las herramientas de software libre en las que se podían pensar para realizar esta herramienta no mejoraban de ninguna manera la utilización de Excel.

5.4.2. SOFTWARE EMPLEADO

- Microsoft Windows XP Professional Service Pack 1
- Microsoft Windows 2000 Professional Service Pack 4
- Microsoft Excel 2000 y Excel XP
- Adobe Acrobat 6.0 Professional
- Microsoft Word XP
- Rational Rose Demo Ver. 4
- Adobe Photoshop CS

5.4.3. HARDWARE EMPLEADO

Equipo principal del desarrollo

- S.O. Windows 2000 Professional Service Pack 4
- Pentium III
- 192 Mb RAM

Equipo secundario de desarrollo

- S.O. Windows XP Professional Service Pack 1
- AMD Athlon XP, 1666 Mhz
- 512 Mb RAM
- HD 60 Gb

Otros

- USB 2.0 JetFlash 64 Mb

5.5. PRUEBAS

<i>DESCRIPCION</i>	<i>COMPROBACION DEL BOTON ACEPTAR DEL FORMULARIO PRINCIPAL SIN DATOS INTRODUCIDOS</i>
Acción Realizada	Pulsación del botón ACEPTAR sin haber introducido datos
Resultado Esperado	Mensaje de error de no haber introducido ningún constructo
Resultado Obtenido	Mensaje "El numero de constructos es nulo"
Observaciones	Correcto

Figura 23: Comprobación del botón ACEPTAR del formulario principal sin datos introducidos.

<i>DESCRIPCION</i>	<i>COMPROBACION DEL BOTON ACEPTAR DEL FORMULARIO PRINCIPAL SIN ELEMENTOS INTRODUCIDOS</i>
Acción Realizada	Pulsación del botón ACEPTAR habiendo introducido solo el numero de constructos
Resultado Esperado	Mensaje de error de no haber introducido ningún elemento
Resultado Obtenido	Mensaje "El numero de elementos es nulo"
Observaciones	Correcto

Figura 24: Comprobación del botón ACEPTAR del formulario principal sin elementos introducidos.

<i>DESCRIPCION</i>	<i>COMPROBACION DEL BOTON ACEPTAR DEL FORMULARIO PRINCIPAL SIN VALOR MAXIMO INTRODUCIDO</i>
Acción Realizada	Pulsación del botón ACEPTAR sin haber introducido el valor máximo
Resultado Esperado	Mensaje de error de no haber introducido ningún valor máximo
Resultado Obtenido	Mensaje "El valor máximo es nulo"
Observaciones	Correcto

Figura 25: Comprobación del botón ACEPTAR del formulario principal sin valor máximo introducido.

<i>DESCRIPCION</i>	<i>COMPROBACION DEL BOTON ACEPTAR DEL FORMULARIO PRINCIPAL CON VALOR MAXIMO PAR</i>
Acción Realizada	Pulsación del botón ACEPTAR habiendo introducido un numero máximo par
Resultado Esperado	Mensaje de error de no haber introducido un numero impar
Resultado Obtenido	Mensaje "El valor máximo es un numero par"
Observaciones	Correcto

Figura 26: Comprobación del botón ACEPTAR del formulario principal con valor máximo par.

<i>DESCRIPCION</i>	<i>COMPROBACION DEL BOTON ACEPTAR DEL FORMULARIO PRINCIPAL</i>
Acción Realizada	Pulsación del botón ACEPTAR habiendo introducido todos los datos
Resultado Esperado	Mostrar la rejilla de introducción de datos
Resultado Obtenido	Rejilla mostrada
Observaciones	Correcto

Figura 27: Comprobación del botón ACEPTAR del formulario principal.

<i>DESCRIPCION</i>	<i>COMPROBACION DE LA REJILLA AL LLAMAR AL FORMULARIO FUERZA</i>
Acción Realizada	Pulsar CTRL+a
Resultado Esperado	Aparición del formulario Fuerza
Resultado Obtenido	Formulario Fuerza mostrado
Observaciones	Correcto

Figura 28: Comprobación de la rejilla al llamar al formulario Fuerza.

<i>DESCRIPCION</i>	<i>COMPROBACION DEL BOTON CREAR ALPHA-PLANOS SIN INTRODUCIR TODOS LOS DATOS</i>
Acción Realizada	Pulsar el botón sin haber introducido todos los datos en la rejilla
Resultado Esperado	Mensaje de error de no haber introducido todos los datos en la rejilla
Resultado Obtenido	Mensaje "Alguna celda esta vacía, por favor introduzca los datos"
Observaciones	Correcto

Figura 29: Comprobación del botón CREAR ALPHA-PLANOS sin introducir todos los datos.

<i>DESCRIPCION</i>	<i>COMPROBACION DEL BOTON CREAR ALPHA-PLANOS CON NUMEROS MAYORES QUE EL MAXIMO</i>
Acción Realizada	Pulsar el botón habiendo introducido todos los datos en la rejilla pero con algún valor mayor que el máximo
Resultado Esperado	Mensaje de error de haber introducido valores mayores que el máximo
Resultado Obtenido	Mensaje "Ha introducido algún numero mayor que el máximo"
Observaciones	Correcto

Figura 30: Comprobación del botón CREAR ALPHA-PLANOS con números mayores que el máximo.

<i>DESCRIPCION</i>	<i>COMPROBACION DEL BOTON CREAR ALPHA-PLANOS CON TODOS LOS DATOS CORRECTOS</i>
Acción Realizada	Pulsar el botón habiendo introducido los datos correctamente
Resultado Esperado	Creación de alpha-planos y aparición del menú Analizar
Resultado Obtenido	Alpha-planos creados y mostrado el formulario Analizar
Observaciones	Correcto

Figura 31: Comprobación del botón CREAR ALPHA-PLANOS con todos los datos correctos.

<i>DESCRIPCION</i>	<i>COMPROBACION DEL BOTON ANALIZAR SIN INTRODUCIR DATOS</i>
Acción Realizada	Pulsar el botón sin haber introducido el grado de confirmación
Resultado Esperado	Mensaje de error de no haber introducido el grado de confirmación
Resultado Obtenido	Mensaje "El grado de confirmación es nulo"
Observaciones	Correcto

Figura 32: Comprobación del botón ANALIZAR sin introducir datos.

<i>DESCRIPCION</i>	<i>COMPROBACION DEL BOTON ANALIZAR CON LOS DATOS INTRODUCIDOS</i>
Acción Realizada	Pulsar el botón habiendo introducido el grado de confirmación
Resultado Esperado	Aparición de la hoja de resultados ordenada
Resultado Obtenido	Se muestra la hoja con los resultados ordenada
Observaciones	Correcto

Figura 33: Comprobación del botón ANLIZAR con los datos introducidos.

<i>DESCRIPCION</i>	<i>COMPROBACION DE LA APARICION DEL FORMULARIO REEMPLAZAR</i>
Acción Realizada	Pulsar ctrl+d para que aparezca el formulario Reemplazar
Resultado Esperado	Aparición del formulario reemplazar
Resultado Obtenido	El formulario Reemplazar es mostrado
Observaciones	Correcto

Figura 34: Comprobación de la aparición del formulario reemplazar.

<i>DESCRIPCION</i>	<i>COMPROBACION DEL BOTON REEMPLAZAR DEL FORMULARIO SIN INTRODUCIR VALORES</i>
Acción Realizada	Pulsar Reemplazar sin haber introducido ningún valor
Resultado Esperado	Mensaje de error advirtiéndolo de introducir los datos
Resultado Obtenido	Mensaje "No se ha introducido ningún valor por el que reemplazar el texto"
Observaciones	Correcto

Figura 35: Comprobación del botón REEMPLAZAR del formulario sin introducir valores.

<i>DESCRIPCION</i>	<i>COMPROBACION DEL BOTON REEMPLAZAR DEL FORMULARIO INTRODUCIENDO VALORES</i>
Acción Realizada	Pulsar Reemplazar habiendo introducido algún valor
Resultado Esperado	Reemplazamiento de ese valor en la hoja de resultados
Resultado Obtenido	El nombre de seleccionado ha sido reemplazado
Observaciones	Correcto

Figura 36: Comprobación del botón REEMPLAZAR del formulario introduciendo valores.

<i>DESCRIPCION</i>	<i>COMPROBACION DEL BOTON CANCELAR DEL FORMULARIO</i>
Acción Realizada	Pulsar Cancelar en el formulario Reemplazar
Resultado Esperado	Cierre de el formulario y aparición de la hoja de resultados
Resultado Obtenido	Formulario cerrado y hoja de resultados mostrada
Observaciones	Correcto

Figura 37: Comprobación del botón CANCELAR del formulario reemplazar.

5.6. EJEMPLO DE USO DE LA APLICACION

5.6.1. DESCRIPCIÓN

Para mostrar los resultados de la herramienta de Repertory Grid se expone a continuación un breve ejemplo de su funcionamiento, en el cual, a partir de una tabla con constructos y elementos, se obtienen las relaciones entre constructos.

Para más información sobre la fuente de este ejemplo, consultar bibliografía.

5.6.2. PANTALLA PRINCIPAL

Esta es la pantalla que se muestra nada mas arrancar la herramienta.

En ella se introducen los valores del número de constructos, elementos y el valor máximo de las relaciones, para este ejemplo los valores son:

- N° Constructos = 13
- N° Elementos = 27
- Valor máximo = 5

PERSONAL CONSTRUCT

introduzca el numero de CONSTRUCTOS, ELEMENTS y VALOR MAXIMO correspondientes y pulse el boton "ACEPTAR", a continuacion rellene los nombres y los valores de la tabla con la que desee trabajar.
Cuando tenga los valores introducidos pulse CTRL+a para seguir con la ejecucion de la aplicacion.

	A	B	C	D	E	F	G
1	P.Constructi	z	x	w	q	r	l
2	a	1	3	2	3	1	3
3	h	3	6	5	4	3	6
4	c	5	7	6	7	5	7
5	d	7	7	1	1	7	7
6	e	3	9	5	3	8	9
7	f	12	6	6	12	12	6
8	g	3	0	5	4	3	0
9	h	4	5	3	3	4	5
10	i	3	6	8	7	8	6
11	j	5	5	9	4	5	5
12	k	3	4	9	6	8	4
13	l	3	1	6	5	9	7
14	m	13	2	4	4	10	2
15	n	2	3	3	2	2	3
16	e	4	5	1	1	4	5

Nº CONSTRUCTOS Nº ELEMENTS

VALOR MÁXIMO

ACEPTAR

Ilustración 1: Formulario Principal de la herramienta "Repertory Grid"-

5.6.3. REJILLA

Es la parte principal de la herramienta, en ella se introducen todos los valores de la tabla que se ha empleado como entrada de datos.

Para seguir con la herramienta, después de introducir todos los datos en la rejilla, se pulsa CTRL+a.

	A	B	C	D	E	F	G	H	I	J	K	L	
		CapMed	PatientSite	Your Health	IQHealth	ChannelHeal th	YourHealthC hart.com	ePatient Health Record	Bank of Health	HealthComp ass	i-Beacon	PHDtogo.co m	LifeC eve
1	P.Constructi												
2	Cost of PHR	1	5	5	3	3	5	5	3	5	3	5	3
3	Public Access	5	5	5	3	1	5	5	5	5	5	5	1
4	Health Record	5	5	4	5	3	3	4	5	5	4	4	4
5	Demo/ Tour	5	5	1	5	5	5	5	5	1	5	1	1
6	Access to other Health Info	1	5	5	5	5	3	5	3	5	5	1	5
7	Health care Practitioner On-line	1	1	1	1	1	3	1	1	1	1	1	1
8	Health Risk Appraisal	1	1	5	5	1	3	5	5	5	5	1	3
9	Data Exchange Ability	1	1	1	1	1	3	5	1	1	1	1	5
10	Privacy, Security, Confidentia lity Tech / Info	1	2	4	4	1	1	1	4	5	5	2	1
	Cust. Services for consumers, clinicians,	1	1	1	5	5	5	5	1	1	1	1	5

Ilustración 2: Rejilla de introducción de datos.

5.6.4. CREACIÓN DE ALPHA-PLANOS

Permite crear los alpha-planos aunque no son visibles al usuario.

Al pulsar el botón “CREAR ALPHA-PLANOS” la herramienta crea todos los alpha-planos con los datos recogidos en la rejilla.



FORMULARIO CREACION DE ALPHA-PLANOS

A continuacion se crearan las N-tablas binarias (alpha-planos), nos aseguraremos que ningun valor es nulo, esta sin introducir o es mayor que el maximo introducido anteriormente.

Pulse el boton "CREAR ALPHA-PLANOS" para crear las tablas

CREAR ALPHA-PLANOS

Ilustración 3: Formulario de creación de Alpha-planos.

	A	B	C	D	E	F	G	H	I	J	K	L
1	P. Constructi	CapMed	PatientSite	Your Health	IQHealth	ChannelHealt	YourHealthCie	Patient Heal	Bank of Healt	HealthCompai	-Beacon	PHDtogo.com Lif
2	Cost of PHR	1	0	0	0	0	0	0	0	0	0	0
3	Public Acces	0	0	0	0	0	1	0	0	0	0	0
4	Health Recon	0	0	0	0	0	0	0	0	0	0	0
5	Demo/ Tour	0	0	1	0	0	0	0	0	1	0	1
6	Access to oth	1	0	0	0	0	0	0	0	0	0	1
7	Health care F	1	1	1	1	1	0	1	1	1	1	1
8	Health Risk A	1	1	0	0	1	0	0	0	0	0	1
9	Data Exchang	1	1	1	1	1	0	0	1	1	1	1
10	Privacy, Sect	1	0	0	0	1	1	1	0	0	0	0
11	Cust. Service	1	1	1	0	0	0	0	1	1	1	1
12	Ease of Navig	0	0	0	0	0	0	0	0	0	0	0
13	Quality/ Quar	0	0	0	0	0	0	0	0	0	0	0
14	Site Presenta	0	0	0	0	0	0	0	0	0	0	0
15												
16												
17												
18												
19												
20												
21												
22												
23												
24												
25												
26												
27												
28												
29												
30												
31												
32												
33												

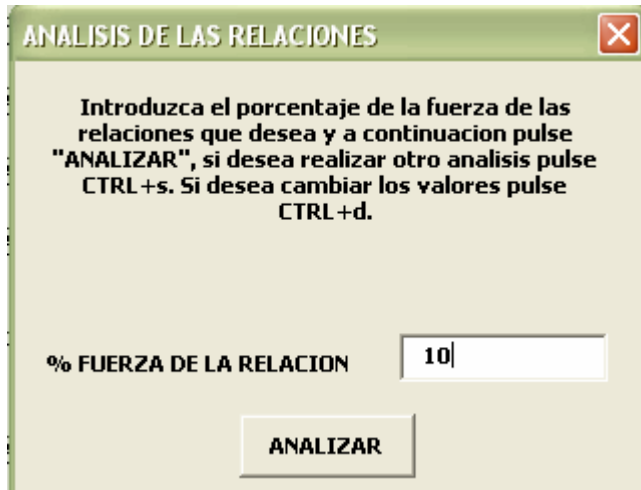
Ilustración 4: Ejemplo de alfaplano.

5.6.5. ANÁLISIS DE LAS RELACIONES

A continuación de crear los alpha-planos, la herramienta muestra el formulario de Análisis de las relaciones en el que se puede introducir el Grado de Confirmación deseado para los resultados.

Para este ejemplo se ha optado por elegir un grado de confirmación del 10%.

Al pulsar el botón “ANALIZAR”, la herramienta realiza las operaciones necesarias para crear las relaciones que a continuación e muestran en la hoja de resultados.



ANÁLISIS DE LAS RELACIONES

Introduzca el porcentaje de la fuerza de las relaciones que desea y a continuación pulse "ANALIZAR", si desea realizar otro analisis pulse CTRL+s. Si desea cambiar los valores pulse CTRL+d.

% FUERZA DE LA RELACION

ANALIZAR

Ilustración 5: Formulario de Análisis de las Relaciones.

5.6.6. RESULTADOS

A continuación se muestran los resultados de las relaciones entre los constructos que se han introducido como datos de entrada, la hoja esta ordenada por constructos.

Si se pulsan las teclas CTRL+d, aparecerá el formulario para reemplazar los valores de los constructor por otros.

	A	B	C	D	E	F	G	H	I	J	K	L
43	Valor 5		Access to other Health Info				Valor 1		Cust. Services for consume		61	
44	Valor 5		Access to other Health Info				Valor 3		Cust. Services for consume		11	
45	Valor 5		Access to other Health Info				Valor 5		Cust. Services for consume		28	
46	Valor 5		Access to other Health Info				Valor 1 2		Cust. Services for consume		61	
47	Valor 5		Access to other Health Info				Valor 5 4		Cust. Services for consume		28	
48	Valor 5 4		Access to other Health Info				Valor 1		Cust. Services for consume		61	
49	Valor 5 4		Access to other Health Info				Valor 3		Cust. Services for consume		11	
50	Valor 5 4		Access to other Health Info				Valor 5		Cust. Services for consume		28	
51	Valor 5 4		Access to other Health Info				Valor 1 2		Cust. Services for consume		61	
52	Valor 5 4		Access to other Health Info				Valor 5 4		Cust. Services for consume		28	
53	Valor 1		Access to other Health Info				Valor 1		Data Exchange Ability		100	
54	Valor 1		Access to other Health Info				Valor 1 2		Data Exchange Ability		100	
55	Valor 1 2		Access to other Health Info				Valor 1		Data Exchange Ability		100	
56	Valor 1 2		Access to other Health Info				Valor 1 2		Data Exchange Ability		100	
57	Valor 3		Access to other Health Info				Valor 1		Data Exchange Ability		50	
58	Valor 3		Access to other Health Info				Valor 3		Data Exchange Ability		50	
59	Valor 3		Access to other Health Info				Valor 1 2		Data Exchange Ability		50	
60	Valor 5		Access to other Health Info				Valor 1		Data Exchange Ability		83	
61	Valor 5		Access to other Health Info				Valor 5		Data Exchange Ability		17	
62	Valor 5		Access to other Health Info				Valor 1 2		Data Exchange Ability		83	
63	Valor 5		Access to other Health Info				Valor 5 4		Data Exchange Ability		17	
64	Valor 5 4		Access to other Health Info				Valor 1		Data Exchange Ability		83	
65	Valor 5 4		Access to other Health Info				Valor 5		Data Exchange Ability		17	
66	Valor 5 4		Access to other Health Info				Valor 1 2		Data Exchange Ability		83	
67	Valor 5 4		Access to other Health Info				Valor 5 4		Data Exchange Ability		17	
68	Valor 1		Access to other Health Info				Valor 1		Demo/ Tour		40	
69	Valor 1		Access to other Health Info				Valor 5		Demo/ Tour		60	
70	Valor 1		Access to other Health Info				Valor 1 2		Demo/ Tour		40	
71	Valor 1		Access to other Health Info				Valor 5 4		Demo/ Tour		60	
72	Valor 1 2		Access to other Health Info				Valor 1		Demo/ Tour		40	
73	Valor 1 2		Access to other Health Info				Valor 5		Demo/ Tour		60	
74	Valor 1 2		Access to other Health Info				Valor 1 2		Demo/ Tour		40	
75	Valor 1 2		Access to other Health Info				Valor 5 4		Demo/ Tour		60	

Ilustración 6: Resultados de la herramienta.

5.6.7. REEMPLAZO

Si se desea cambiar los valores de los constructos por otros, seleccionamos el valor a reemplazar y el valor por el cual se reemplaza, los resultados se mostraran en la ventana de resultados.

En este ejemplo se han sustituido los valores “Valor1” de los constructor por el texto “++”.

Reemplazar valores

Buscar:

- Valor 1
- Valor 2
- Valor 3
- Valor 4
- Valor 5
- Valor 1 2
- Valor 5 4

Reemplazar por : ++

REEMPLAZAR

CANCELAR

Ilustración 7: Formulario de Reemplazo.

Microsoft Excel - Repertory_Grid_version_español1

Archivo Edición Ver Insertar Formato Herramientas Datos Ventana ? Ayuda PDF

Escriba una pregunta

Responder con cambios... Terminar revisión...

A1 VALOR

	A	B	C	D	E	F	G	H	I	J	K	L
	VALOR		CONSTRUCT	IMPLICA			VALOR		CONSTRUCT		GRADO DE CONFIRMACION	
1	++		Access to other Health Info				++		Cost of PHR		20	
2	++		Access to other Health Info				Valor 2		Cost of PHR		20	
3	++		Access to other Health Info				Valor 3		Cost of PHR		20	
4	++		Access to other Health Info				Valor 5		Cost of PHR		40	
5	++		Access to other Health Info				Valor 1 2		Cost of PHR		40	
6	++		Access to other Health Info				Valor 5 4		Cost of PHR		40	
7	++		Access to other Health Info				++		Cost of PHR		20	
8	Valor 1 2		Access to other Health Info				Valor 2		Cost of PHR		20	
9	Valor 1 2		Access to other Health Info				Valor 3		Cost of PHR		20	
10	Valor 1 2		Access to other Health Info				Valor 5		Cost of PHR		40	
11	Valor 1 2		Access to other Health Info				Valor 1 2		Cost of PHR		40	
12	Valor 1 2		Access to other Health Info				Valor 5 4		Cost of PHR		40	
13	Valor 1 2		Access to other Health Info				++		Cost of PHR		25	
14	Valor 3		Access to other Health Info				Valor 2		Cost of PHR		25	
15	Valor 3		Access to other Health Info				Valor 3		Cost of PHR		25	
16	Valor 3		Access to other Health Info				Valor 5		Cost of PHR		25	
17	Valor 3		Access to other Health Info				Valor 1 2		Cost of PHR		50	
18	Valor 3		Access to other Health Info				Valor 5 4		Cost of PHR		25	
19	Valor 5		Access to other Health Info				++		Cost of PHR		11	
20	Valor 5		Access to other Health Info				Valor 3		Cost of PHR		28	
21	Valor 5		Access to other Health Info				Valor 5		Cost of PHR		61	
22	Valor 5		Access to other Health Info				Valor 1 2		Cost of PHR		11	
23	Valor 5		Access to other Health Info				Valor 5 4		Cost of PHR		61	
24	Valor 5 4		Access to other Health Info				++		Cost of PHR		11	
25	Valor 5 4		Access to other Health Info				Valor 3		Cost of PHR		28	
26	Valor 5 4		Access to other Health Info				Valor 5		Cost of PHR		61	
27	Valor 5 4		Access to other Health Info				Valor 1 2		Cost of PHR		11	
28	Valor 5 4		Access to other Health Info				Valor 5 4		Cost of PHR		61	
29	++		Access to other Health Info				++		Cust. Services for consume		80	
30	++		Access to other Health Info				Valor 5		Cust. Services for consume		20	
31	++		Access to other Health Info				Valor 1 2		Cust. Services for consume		80	
32	++		Access to other Health Info				Valor 5 4		Cust. Services for consume		20	
33	++		Access to other Health Info									

Lunes, 21 de febrero de 2005

Ilustración 8: Resultado del reemplazo de los valores.

6. CONCLUSIONES Y FUTURAS ACTUACIONES

6.1. OBJETIVOS ALCANZADOS

El objetivo principal de este proyecto consistía en el análisis de componentes software reutilizables en el dominio de la informática para discapacitados, para ello se emplearon, entrevistas, revisión de documentaciones existentes, etc..

Se utilizó un proceso de desarrollo específico y una herramienta que permite organizar la información de diagramas de requisitos en una estructura jerárquica.(\mathbf{R}^2). Esta herramienta complementa el marco de trabajo para la reutilización de requisitos y constituye la parte práctica de la investigación. La herramienta permite ver que los conceptos vistos en este proyecto son representables en herramientas automáticas, necesarias para afrontar la producción industrial del software. Contar con un entorno ha sido un factor relevante para facilitar la creación de los diagramas de requisitos tanto funcionales como no funcionales.

El resultado de emplear esta herramienta ha sido una colección de requisitos y modelos de análisis para su posterior utilización.

Se ha empleado posteriormente una técnica visual para entender la variabilidad de requisitos, empleando una **herramienta visual interactiva de análisis de variantes** que ha servido para realizar un ejemplo de cómo utilizar dicha herramienta con la colección de requisitos anteriormente estudiados, comparándolo con un ejemplo real.

Finalmente, y como extensión del proyecto se ha desarrollado una herramienta llamada Repertory Grid, partiendo de la Teoría de Constructos Personales (PCT) para la elicitation de requisitos.

La utilización de estas herramientas debe ser por parte de un Experto en el dominio que pueda entenderlas perfectamente y cuya utilización aumente las posibilidades que el experto tendría sin ellas.

El mayor trabajo que se ha llevado a cabo en el proyecto es definir claramente los objetivos de este dominio, cosa bastante compleja debido a la enorme dificultad del dominio de la informática para discapacitados, que abarca un muy amplio abanico de discapacidades, aplicaciones, necesidades, etc..

Con este proyecto, se pretende tener al alcance de los desarrolladores una forma de elicitar objetivos y utilizarlos posteriormente que en este caso se ha acotado a la informática para discapacitados, pero que se puede emplear para cualquier dominio que se desee.

Se han empleado distintas herramientas en distintos lenguajes de programación para poder realizarlo, y se ha terminado el proyecto con la incursión en una manera diferente de elicitar los objetivos como es la Teoría de Constructos Personales y la técnica del *Repertory Grid*, una técnica muy potente y con distintas aplicaciones.

La idea principal del desarrollo de este proyecto es estudiar los requisitos del dominio de la informática para Discapacitados y la potencia que puede tener la reutilización de dichos requisitos si están adecuadamente explicados y definidos, así como la utilización de mecanismos que favorezcan su búsqueda, clasificación y reutilización.

La potencia de la reutilización de los requisitos del dominio de la informática para discapacitados se ha visto en el ejemplo realizado del capítulo 4, que pone de manifiesto que esta reutilización reduce costes, trabajo, tiempo, etc.

6.2. FUTURAS ACUTACIONES

Herramienta Repertory Grid

La herramienta realizada en este proyecto es el primer prototipo de una herramienta de este estilo, este prototipo a partir de una tabla donde esta almacenado el Repertory Grid, devuelve las reglas del análisis propuesto por Ford que sugieren un cierto grado de confirmación.

Un posible futuro enfoque es realizar otros tipos de análisis sobre el Repertory Grid, como puede ser el estudio de distancias, etc.

7. APENDICE

7.1. DESCRIPCIÓN DE LA APLICACIÓN

Esta herramienta esta basada en la Teoría de Constructos Personales empleando la técnica del *Repertory Grid*.

Permitirá al experto llevar a cabo las siguientes opciones:

- Introducir los datos para crear la rejilla del Repertory Grid.
- Crear los Alpha-planos
- Introducir el Grado de Confirmación deseado.
- Cambiar el valor de los constructos.

A continuación se detalla el uso de la aplicación, para un manejo correcto.

7.1.1. PANTALLA PRINCIPAL

Esta es la pantalla que se muestra nada mas arrancar la herramienta.

En ella se muestra una pequeña información sobre lo que se debe de realizar y los pasos a seguir, así como cajas de texto para la introducción de los datos y un botón, el cual vera si los datos son correctos y mostrara la rejilla para introducir los datos.

PERSONAL CONSTRUCT

introduzca el numero de CONSTRUCTOS, ELEMENTS y VALOR MAXIMO correspondientes y pulse el boton "ACEPTAR", a continuacion rellene los nombres y los valores de la tabla con la que desee trabajar.
Cuando tenga los valores introducidos pulse CTRL+a para seguir con la ejecucion de la aplicacion.

	A	B	C	D	E	F	G
1	P.Constru	1	3	2	3	1	3
2	a	1	3	2	3	1	3
3	h	3	6	5	4	3	6
4	c	5	7	6	7	5	7
5	d	7	7	1	1	7	7
6	e	3	9	5	3	8	9
7	f	12	6	6	12	12	6
8	g	3	0	5	4	3	0
9	h	4	5	3	3	4	5
10	i	3	6	8	7	8	6
11	j	5	5	9	4	5	5
12	k	3	4	9	6	8	4
13	l	3	1	6	5	9	1
14	m	13	2	4	4	10	2
15	n	2	3	3	2	2	3
16	e	4	5	1	1	4	5

N° CONSTRUCTOS

N° ELEMENTOS

VALOR MÁXIMO

ACEPTAR

Ilustración 9: Formulario Principal de la herramienta "Repertory Grid"-

7.1.2. REJILLA

Es la parte principal de la herramienta, en ella se introducen los constructos, los elementos así como el numero indicado por el experto para relacionarlos. A partir de ella se obtendrán todos los resultados que se desean.

Para seguir con la herramienta, después de introducir todos los datos en la rejilla, se pulsa CTRL+a.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
		CapMed	PatientSite	Your Health	IQHealth	ChannelHealth	YourHealthC	ePatient	Bank of	HealthComp	i-Beacon	PHDrogo.co	LifeConnect	Worldwide	98point6	healthAtoZ	Med
1	P. Construc																
2	Cost of	1	5	5	3	3	5	5	3	5	3	5	3	1	5	5	3
3	Public	5	5	5	3	1	5	5	5	5	5	5	1	5	5	5	1
4	Access	5	5	4	5	3	3	4	5	5	4	4	4	5	5	5	5
5	Health	5	5	1	5	5	5	5	5	1	5	1	1	1	1	1	1
6	Record	1	5	5	5	5	3	5	3	5	5	1	5	3	5	5	5
7	Demo/	1	1	1	1	1	3	1	1	1	1	1	1	3	1	1	1
8	Tour	1	1	1	1	1	3	1	1	1	1	1	1	3	1	1	1
9	Access to	1	1	1	1	1	3	1	1	1	1	1	1	3	1	1	1
10	Health	1	1	1	1	1	3	1	1	1	1	1	1	3	1	1	1
11	Practitioner	1	1	5	5	1	3	5	5	5	5	1	3	3	1	1	3
12	Health	1	1	1	1	1	3	5	1	1	1	1	5	1	1	1	1
13	Risk	1	1	1	1	1	3	5	1	1	1	1	5	1	1	1	1
14	Data	1	2	4	4	1	1	1	4	5	5	2	1	2	2	5	1
15	Exchange	1	1	1	1	1	3	5	1	1	1	1	5	1	1	1	1
16	Ability	1	1	1	1	1	3	5	1	1	1	1	5	1	1	1	1
17	Privacy, Security, Confidentiality Tech	1	1	1	1	1	3	5	1	1	1	1	5	1	1	1	1
18	Info	1	1	1	1	1	3	5	1	1	1	1	5	1	1	1	1
19	Cust. Services for consumer s, clinicians.	3	4	4	3	3	3	4	3	5	3	3	3	3	4	4	3
20	Ease of Navigation	3	5	4	3	3	3	4	3	4	3	3	3	3	4	4	3
21	Quality/ Quantity of Info Available	3	4	4	3	3	3	4	3	5	3	3	3	3	4	2	3
22	Site Presentation	3	4	4	3	3	3	4	3	5	3	3	3	3	4	2	3
23																	
24																	
25																	
26																	
27																	

Ilustración 10: Ejemplo de rejilla de introducción de datos.

7.1.3. FORMULARIO DE CREACIÓN DE ALPHA-PLANOS

A este formulario se llega después de haber introducido todos los datos en la rejilla y haber pulsado CTRL+a.

Permite crear los alpha-planos empleando la formula de creación de alpha-planos vista anteriormente en la técnica del *Repertory Grid*.

Al pulsar el botón “CREAR ALPHA-PLANOS” la herramienta crea todos los alpha-planos con los datos recogidos en la rejilla.

Los Alpha-planos resultantes no se muestran al experto por que son medios para alcanzar el resultado final.



FORMULARIO CREACION DE ALPHA-PLANOS

A continuacion se crearan las N-tablas binarias (alpha-planos), nos aseguraremos que ningun valor es nulo, esta sin introducir o es mayor que el maximo introducido anteriormente. Pulse el boton "CREAR ALPHA-PLANOS" para crear las tablas

CREAR ALPHA-PLANOS


Ilustración 11: Formulario de creación de Alpha-planos.

7.1.4. FORMULARIO DE ANÁLISIS DE LAS RELACIONES

A continuación de crear los alpha-planos, la herramienta muestra el formulario de Análisis de las relaciones en el que se puede introducir el Grado de Confirmación deseado para los resultados.

Al pulsar el botón “ANALIZAR”, la herramienta realiza las operaciones necesarias para crear las relaciones que a continuación e mostraran en la hoja de resultados.

A este formulario se llega también a través de la pulsación de las teclas CTRL+s.



ANÁLISIS DE LAS RELACIONES

Introduzca el porcentaje de la fuerza de las relaciones que desea y a continuación pulse "ANALIZAR", si desea realizar otro análisis pulse CTRL+s.

% FUERZA DE LA RELACION

ANALIZAR

Ilustración 12: Formulario de Análisis de las Relaciones.

7.1.5. HOJA DE RESULTADOS

En esta hoja se muestran los resultados obtenidos por la herramienta con los datos que se han introducido.

La hoja esta ordenada por constructos, además, dependiendo de la franja de grado de confirmación el resultado aparecerá en diferentes colores.

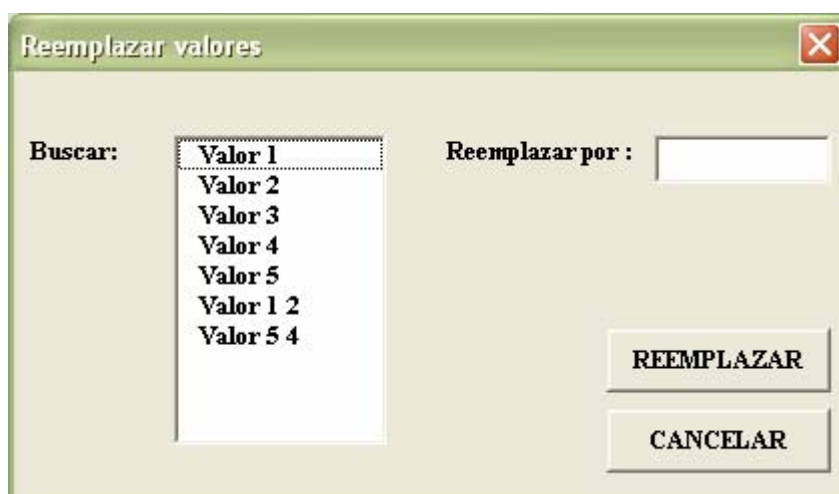
Si se pulsan las teclas CTRL+d, aparecerá el formulario para reemplazar los valores de los constructor por otros.

	A	B	C	D	E	F	G	H	I	J	K	L
	VALOR		CONSTRUCT		IMPLICA		VALOR		CONSTRUCT		GRADO DE CONFIRMACION	
1	Valor 1		Cost of PHR				Valor 1		Demo/ Tour		50	
2												
3	Valor 1		Cost of PHR				Valor 1		Access to other Health Info		25	
4												
5	Valor 1		Cost of PHR				Valor 1		Health care Practioner On-l		50	
6												
7	Valor 1		Cost of PHR				Valor 1		Health Risk Appraisal		50	
8												
9	Valor 1		Cost of PHR				Valor 1		Data Exchange Ability		100	
10												
11	Valor 1		Cost of PHR				Valor 1		Privacy, Security, Confiden		25	
12												
13	Valor 1		Cost of PHR				Valor 1		Cust. Services for consume		100	
14												
15	Valor 1		Cost of PHR				Valor 2		Privacy, Security, Confiden		25	
16												
17	Valor 1		Cost of PHR				Valor 2		Quality/ Quantity of Info A		25	
18												
19	Valor 1		Cost of PHR				Valor 2		Site Presentation		25	
20												
21	Valor 1		Cost of PHR				Valor 3		Health Record		25	
22												
23	Valor 1		Cost of PHR				Valor 3		Access to other Health Info		25	
24												
25	Valor 1		Cost of PHR				Valor 3		Health care Practioner On-l		25	
26												
27	Valor 1		Cost of PHR				Valor 3		Health Risk Appraisal		50	
28												
29	Valor 1		Cost of PHR				Valor 3		Ease of Navigation		50	
30												
31	Valor 1		Cost of PHR				Valor 3		Quality/ Quantity of Info A		50	
32												
33	Valor 1		Cost of PHR				Valor 3		Site Presentation		50	
34												
35	Valor 1		Cost of PHR				Valor 3					

Ilustración 13: Hoja de resultados de la herramienta.

7.1.6. FORMULARIO DE REEMPLAZO

A este formulario se accede pulsando las teclas CTRL+d, permite cambiar los valores de los constructos por otros que deseemos, los resultados se mostraran en la ventana de resultados.



Reemplazar valores

Buscar:

- Valor 1
- Valor 2
- Valor 3
- Valor 4
- Valor 5
- Valor 1 2
- Valor 5 4

Reemplazar por :

REEMPLAZAR

CANCELAR

Ilustración 14: Ejemplo de Formulario de Reemplazo.

El botón “REEMPLAZAR” sustituye los valores seleccionados por los que el Experto desea, mientras que el botón “CANCELAR” cierra el formulario.

8. BIBLIOGRAFIA

Informática para Discapacitados

A. Bork, **“El ordenador en la Enseñanza”**, Gustavo Gili, S.A., 1986.

Robert L. Burke, **“Enseñanza Asistida por Ordenador”**, Paraninfo, S.A., 1986.

M. Fernandez González, **“Enseñanza Asistida por Ordenador”**, Anaya, 1983.

M. Del. P. Lecuona, **“Lenguajes Alternativos para personas con dificultades en la comunicación”**, CEPE S.A., 1994.

Reutilización y Requisitos

G. Arango y R. Prieto-Díaz, **“Domain Analysis: Concepts and Research Directions”**, IEEE Computer Society Press, 1989.

Maria Belén Arenal González, Cristina Maria López Aguado **“Comunicador”**, Proyecto de Fin de Carrera, junio 2002.

M. Bellinzoni, M.G. Gugini y B. Pernici, **“Reusing Specifications in OO Applications”**, IEEE Software, Marzo 1994.

D. Cabezas y J. Vicente, **“Un entorno para reutilización sistemática de requisitos: El Gestor de Datos”**, Proyecto de Fin de Carrera, 2002.

Eva Maria Curero Gomez, Raquel Garcia Rocha **“Herramienta Didáctica para la Estimulación en la Comunicación”** Proyecto de Fin de Carrera, Junio 2002.

E. Ecube, **“Un entorno para la reutilización sistemática de requisitos: El traductor de Diagramas de Requisitos”**, Proyecto de Fin de Carrera, 2003.

Stuart R. Faulk, **“Product-line Requirements Specifications: An Approach and Case Study”**, IEEE Computer Society, pg. 48-55, Agosto 2001.

Kenneth M. Ford, Frederick E. Petry, Jack R. Adams-Webber y Paul J. Chang, **“An Approach to knowledge Acquisition Based on Structure of Personal Construct Systems”**, IEEE Transactions on Knowledge and Data Engineering, Vol. 3, nº 1, Marzo 1991.

Bruno González Baixauli, Miguel Ángel Laguna Serrano, Julio Cesar Sampaio Do Prado Leite, **“Aplicaciones de la Teoría de Constructos Personales a la Elicitación de Requisitos”**, 2004.

Bruno González Baixauli, Julio Cesar Sampaio Do Prado Leite, John Mylopoulos, **“Visual Variability Analysis for Goal Models”**, 2004.

I. Jacobson, M. Griss, P. Jonsson **“Software Reuse”**

Axel Van Lamsweerde, **“Requirements Engineering in the year 00: A research perspective”**, ACM Press, Junio 2000.

Fermín Juan Martinez y Virginia Vitoria Mozo, **“Teclado sobre una PDA para personas con Parálisis Cerebral”**, Proyecto de Fin de Carrera, Septiembre 2004.

B. Meyer, **“Reusable Software: The Base Object-Oriented Component Libraries”**, Prentice-Hall, 1995.

Carma McClure, **“Software Reuse Techniques: Adding Reuse to the System Development Process”**, Prentice-Hall, 1997.

Sonsoles Muñoz y Sheila Vicente, **“Un entorno para la reutilización sistemática de requisitos: El gestor del Léxico”**, Proyecto de Fin de Carrera, 2003.

K. Pohl, **“Requirements Engineering, an overview”**, Marcel Decker Inc., 1996.

Natalia Sedano Alonso, Beatriz Garcia Tapia **“Desarrollo de Componentes Software Reutilizables en el Dominio de informática para Discapacitados”** Proyecto de Fin de Carrera, Septiembre 1998.

A. Sutcliffe y M. Maiden, **“The Domain Theory for Requirements Engineering”**, IEEE Transactions on Software Engineering, pg. 174-196, Marzo 1998.

Edward N. Yourdon, **“Software Reuse”**, Application Development Strategies, Cutter Information Corp., vol. VI, nº 12, Diciembre 1994.

Meter G. Aitken, “ **Visual Basic 6: Manual complete de programación**”, Ed. Paraninfo, 1999.

Benedikt Van Almsick, Andreas Dahms, Wolfgang Marzian, "**El Gran Libro de la Programación de Office**".

Grady Booch, James Rumbaugh, Ivan Jacobson, “**El Lenguaje Unificado de Desarrollo Software**”, Ed. Addison Wesley, 2000.

OTROS RECURSOS

Paginas Web

Grupo de investigación y desarrollo sobre reutilización sistemática en el desarrollo de sistemas software. (Posee artículos, libros, herramientas, etc...)
<http://giro.infor.uva.es/>

“Personal Health Record Sites (PHR) Comparison”. The Informatics Review.
http://www.informatics-review.com/thoughts/PHR_Site_Comparison.htm

Cursos, foros de consulta para desarrollo de aplicaciones en diversos lenguajes.
<http://www.lawebdelprogramador.com>

Pagina oficial de Microsoft de la que se puede obtener información acerca del entorno de desarrollo, herramientas, etc..
<http://www.microsoft.com>

Manual de Visual Basic para Excel
<http://tft.upv.es/~JPGARCIA/documentos/practicaseexcel.pdf>