



**UNIVERSIDAD DE CASTILLA-LA MANCHA
ESCUELA SUPERIOR DE INFORMÁTICA**

**INGENIERÍA
EN INFORMÁTICA**

PROYECTO FIN DE CARRERA

**Sistema de Información para la Gestión
Automática de los Intercambios entre Estudiantes
Sócrates/Erasmus**

Miguel Ángel Camarena Nieto-Márquez

Septiembre, 2011



UNIVERSIDAD DE CASTILLA-LA MANCHA
ESCUELA SUPERIOR DE INFORMÁTICA

TECNOLOGÍAS Y SISTEMAS DE LA INFORMACIÓN

PROYECTO FIN DE CARRERA

**Sistema de Información para la Gestión
Automática de los Intercambios entre Estudiantes
Sócrates/Erasmus**

Autor: Miguel Ángel Camarena Nieto-Márquez

Director: Javier Alonso Albusac Jiménez

Tutor Académico: Luis Rodríguez Benítez

Septiembre, 2011

En virtud del Artículo 7 del Real Decreto Legislativo 1/1996, de 12 de abril por el que se aprueba el Texto Refundido de la Ley de Propiedad Intelectual, modificado por la Ley 23/2006 de 7 de julio, este PFC se considera una obra en colaboración entre las diferentes partes. Por tanto la propiedad intelectual de este PFC, productos y de su difusión será compartida con iguales porcentajes entre el alumno, director y coordinador.

TRIBUNAL:

Presidente: _____

Vocal 1: _____

Vocal 2: _____

Secretario: _____

FECHA DE DEFENSA: _____

CALIFICACIÓN: _____

PRESIDENTE

VOCAL 1

VOCAL 2

SECRETARIO

Fdo.:

Fdo.:

Fdo.:

Fdo.:

*Este proyecto está dedicado a mis padres y mi hermano,
por su constante apoyo y confianza durante mis años de estudio.*

Resumen

La proliferación y el avance de Internet y las posibilidades de comunicación que nos ofrece, proporciona una amplia mejora en el tratamiento y gestión de la información de determinados procesos.

La realidad de un estudiante que quiere cursar una beca Erasmus requiere de una serie de pasos, que aún estando bien definidos, en la actualidad no son tratados ni gestionados mediante las Tecnologías de la Información disponibles. No se da la facilidad de mantener una evolución participativa, controlada, segura y sencilla del proceso.

Lo que se pretende conseguir es un Sistema de Información que sirva de apoyo tanto al alumno como a los diferentes roles implicados. Se buscará tener una visión clara de los distintos pasos que hay que ejecutar, el estado actual en el que se encuentra y los pasos que restan hasta finalizar la beca.

Agradecimientos

A Javier Alonso Albusac Jiménez y Luis Rodríguez Benítez,
por su gran idea, su continuo apoyo y dedicación.

Índice general

1	Introducción	1
1.1	Enfoque de la problemática existente	2
1.2	Estructura del documento	4
2	Objetivos del Proyecto	5
3	Antecedentes, Estado del Conocimiento	9
3.1	Desarrollo Web Estático y Dinámico	9
3.1.1	HTML Estático	9
3.1.2	HTML Dinámico	11
3.1.2.1	PHP	13
3.1.2.2	JSP	15
3.1.2.3	ASP	15
3.1.2.4	Javascript	16
3.1.3	CSS	18
3.1.3.1	Ventajas	18
3.1.3.2	¿Cómo se aplica?	18
3.1.4	XML	20
3.2	Usabilidad Web	23
3.3	Almacenamiento de la Información	25
3.3.1	Ficheros	25
3.3.2	Bases de datos	26
3.3.2.1	Access	28
3.3.2.2	SQL Server	28
3.3.2.3	MySQL	29
3.3.2.4	ORACLE	30
3.4	Gestores de Contenidos	31
3.4.1	Usuarios	32
3.4.2	Roles y Permisos	33
3.4.3	Flujos de trabajo. Estados.	33
3.4.4	Ejemplos de CMS.	34

3.4.4.1	Joomla	34
3.4.4.2	Drupal	36
3.5	Firmas digitales	38
3.5.1	Criptografía	38
3.5.1.1	Clave Simétrica	40
3.5.1.2	Clave Asimétrica	40
3.5.1.3	Clave Asimétrica y Cifrado de Clave Pública	41
3.5.1.4	Firma Digital	42
3.5.1.5	Certificado Digital	42
3.5.1.6	Algoritmos de Cifrado	43
3.6	Máquinas de Estados	46
3.7	Aprendizaje Automático y Supervisado	48
3.7.1	Aprendizaje Basado en Casos	50
3.7.2	Resolución de Conflictos	51
3.8	Agentes Inteligentes	52
3.8.1	Agentes Recomendadores	53
4	Gestión Automática de los Intercambios entre Estudiantes Sócrates/Erasmus	57
4.1	Proceso de Gestión en Becas Erasmus	57
4.2	Problemática actual	60
4.3	Propuesta de Soluciones a la Problemática Actual	63
4.4	Diseño Multicapa	66
4.5	Tipos de Usuarios y Roles	68
4.5.1	Estructura de la Base de Datos para la configuración de roles y permisos	69
4.6	Máquina de Estados	72
4.6.1	Proceso de Elaboración del Contrato de Estudios	72
4.6.2	Diagrama de Estados	73
4.6.3	Estructura de la Base de Datos para el Almacenamiento de la Máquina de Estados	74
4.6.4	Diagrama de clases relacionadas con el contrato	75
4.7	Sistema de Control de Eventos	77
4.7.1	Creación de un nuevo evento	80
4.7.2	Estructura de la Base de Datos para el diseño del sistema de eventos .	81
4.7.3	Diagrama de clases relacionadas con el control de eventos	83
4.8	Cálculo de Notas y Equivalencias	85
4.8.1	Generación del Acta Final	86
4.8.2	Estructura de la Base de Datos para diseñar el modo de conversión de calificaciones europeas	90
4.8.3	Diagrama de clases relacionadas con el cálculo de notas	91

4.9	Autenticación de Documentos en el Sistema Desarrollado	93
4.9.1	JSigndf	93
4.10	Aprendizaje de Conflictos	102
4.10.1	Tipos de conflictos	103
4.10.2	Agente Recomendador	107
4.10.3	Estructura de la Base de Datos para el diseño del agente recomendador	110
4.10.4	Diagrama de clases relacionadas con el aprendizaje de conflictos . . .	111
5	Resultados	113
5.1	Resultados Obtenidos	113
5.2	Posibles Costes de Implantación del Sistema	118
5.2.1	Costes Anuales de Hosting	118
5.2.2	Costes de Alquiler de un Servidor	119
5.2.3	Costes de Adquisición de un Servidor	120
5.2.4	Decisiones de las tecnologías utilizadas	120
5.2.5	Mecanismos de financiación del sistema	122
6	Conclusiones y Propuestas	125
6.1	Conclusiones	125
6.2	Propuestas y Líneas Futuras	128
A	Manual de Usuario.-	131
B	Normativa Erasmus.-	157
C	Contrato de Estudios.-	163
D	Ficha de Reconocimiento de Estudios.-	167
E	Tabla de Conversión de Calificaciones Europeas.-	169
	Bibliografía	171

Índice de figuras

3.1	Esquema de como se produce la comunicación entre cliente y servidor	10
3.2	Esquema del funcionamiento de las páginas con scripts del servidor	13
3.3	Esquema del proceso de una petición web	14
3.4	Comparación entre SGML, HTML y XML	20
3.5	Ejemplo de código XML	21
3.6	Relación entre Eficacia, Eficiencia y Satisfacción en términos de usabilidad web	23
3.7	Esquema que muestra el SGBD junto a las aplicaciones y BBDD	27
3.8	Panel de administración de Joomla	35
3.9	Gráfica comparativa entre varios CMS	37
3.10	Ejemplo básico de criptografía	39
3.11	Ejemplo de codificación de la información mediante el uso de una clave asimétrica	40
3.12	Ejemplo de codificación de la información mediante el uso de una clave asimétrica y cifrado de clave pública	41
3.13	Máquina de estados finita	46
3.14	Máquina de estados para un ascensor	47
3.15	Esquema del proceso de aprendizaje automático	50
3.16	Agente inteligente de búsqueda	53
3.17	Asistente de Microsoft Office	54
3.18	Asistentes de Microsoft Office	55
4.1	Proceso a seguir para cursar una beca Erasmus	58
4.2	Proceso final de una estancia Erasmus	59
4.3	Esquema del diseño multicapa	67
4.4	Tabla roles de la BBDD	70
4.5	Tabla módulos de la BBDD	70
4.6	Tabla permisos de la BBDD	70
4.7	Diagrama de estados del contrato de estudios	74
4.8	Almacenamiento de la máquina de estados en la BBDD	75
4.9	Diagrama de clases relacionadas con el contrato	76

4.10	Notificación de un nuevo evento	77
4.11	Caso de uso: Firmar contrato por el usuario	78
4.12	Caso de uso: Calificar alumno por el coordinador de centro	78
4.13	Caso de uso: Firmar o rechazar contrato de estudios por el profesor de movilidad	78
4.14	Caso de uso: Introducir usuario	79
4.15	Mensaje opcional que acompaña a la generación de eventos	80
4.16	Primera tabla del sistema de gestión de eventos	82
4.17	Segunda tabla del sistema de gestión de eventos	82
4.18	Diagrama de clases relacionadas con sistema de control de eventos	84
4.19	Tabla de conversión de calificaciones europea	90
4.20	Diagrama de clases relacionadas con el cálculo de notas	92
4.21	Interfaz gráfica de JSigPdf. (versión 0.7)	94
4.22	Formulario para firmar contratos utilizando un certificado digital	96
4.23	Firma digital producida con la herramienta	99
4.24	Algoritmo (Curso correcto)	103
4.25	Algoritmo (Cuatrimestre correcto)	104
4.26	Algoritmo (Número de créditos correcto)	106
4.27	Fase de aprendizaje del algoritmo	108
4.28	Fase de recomendación del algoritmo	109
4.29	Tabla donde se reflejan los conflictos del sistema	110
4.30	Conflicto en la asignatura Information Business	111
4.31	Agente recomendador mostrando los conflictos de la asignatura seleccionada	111
4.32	Diagrama de clases relacionadas con el aprendizaje de conflictos	112
A.1	Modulos de un estudiante en la pantalla de inicio	132
A.2	Guía visual de pasos para realizar el contrato de estudios	132
A.3	Información de la asignatura seleccionada de la lista de asignaturas	134
A.4	Formulario para elegir asignaturas en la UCLM	135
A.5	Tabla que muestra las asignaturas que hay en el contrato de estudios	135
A.6	Asociación de asignaturas de las universidades origen y destino	136
A.7	Formulario para que el alumno firme el contrato de estudios	137
A.8	Guía visual de pasos para realizar modificaciones en el contrato de estudios	137
A.9	Modificaciones de las asignaturas de la UCLM en el contrato de estudios	139
A.10	Modificaciones de las asignaturas de la universidad destino en el contrato de estudios	140
A.11	Agente recomendador mostrando un conflicto entre asignaturas	140
A.12	Formulario para que el alumno firme las modificaciones del contrato de estudios	141
A.13	Modulos del profesor de movilidad en la pantalla de inicio	142
A.14	Listado de los alumnos erasmus del profesor de movilidad	142

A.15 Contratos de los alumnos del profesor de movilidad	142
A.16 Mensaje de error producido por no tener permisos de lectura del contrato de estudios	143
A.17 Listado de contratos disponibles para ser firmados	143
A.18 Formulario para firmar el contrato de estudios	144
A.19 Mensaje enviado al alumno por rechazar su contrato de estudios	145
A.20 Contratos firmados por el profesor de movilidad	145
A.21 Listado de alumnos que ya han obtenido las notas finales	145
A.22 Modulos del coordinador de centro en la pantalla de inicio	146
A.23 Contratos de los alumnos del coordinador de centro	146
A.24 Listado de contratos disponibles para ser firmados	147
A.25 Formulario para firmar el contrato de estudios	148
A.26 Mensaje enviado al alumno por rechazar su contrato de estudios	149
A.27 Contratos firmados por el coordinador de centro	149
A.28 Listado de alumnos que ya han obtenido las notas finales	149
A.29 Tabla de conversión de calificaciones europea	150
A.30 Tabla para editar los intervalos de calificación de un país específico	150
A.31 Listado de alumnos erasmus para calificar	150
A.32 Listado de alumnos erasmus para calificar	151
A.33 Formulario para firmar la ficha de reconocimiento de estudios	151
A.34 Contratos de los alumnos del coordinador institucional	152
A.35 Listado de contratos disponibles para ser firmados	153
A.36 Formulario para firmar la ficha de reconocimiento de estudios	153
A.37 Contratos firmados por el coordinador institucional	154
A.38 Modulos del administrador en la pantalla de inicio	154
A.39 Listado de algunos módulos del sistema	154
A.40 Configuración de permisos de un módulo del sistema	155
A.41 Formulario de registro de los usuarios	156

Índice de cuadros

3.1	Ejemplo de una tabla de una base de datos	26
3.2	Comparativa entre distintos SGBD	30
3.3	Métodos de Cifrado	43
4.1	Módulos y permisos de acceso al sistema de los mismos	69
4.2	Permisos de lectura y escritura para los distintos estados del contrato de estudios.	73
4.3	Ejemplo de la tabla de conversión de calificaciones europeas	79
4.4	Ejemplo de la tabla de conversión de calificaciones europeas	86
5.1	Costes de Hosting en plataforma GNU/Linux	118
5.2	Costes de Hosting en plataforma Windows	118
5.3	Comparativa de alquiler de servidores en las plataformas GNU/Linux y Windows	119
5.4	Comparativa de alquiler de servidores en las plataformas GNU/Linux y Windows	120

Capítulo 1

Introducción

El avance imparable de las tecnologías, en concreto de Internet, hace posible la llegada de información a cualquier parte del mundo de manera instantánea. Ya no es necesario que varias personas estén físicamente presentes en tiempo ni en lugar para trabajar juntas, ya que utilizando las herramientas que nos proporcionan las Tecnologías de la Información (TICs) podemos hacer de este inconveniente una ventaja.

Con este sistema lo que se pretende es precisamente aprovechar esas ventajas para poder gestionar de la manera más eficaz el proceso que se tiene que llevar a cabo cuando un estudiante decide cursar una beca Erasmus.

Según la oficina de relaciones internacionales de la Universidad de Castilla-La Mancha, Erasmus forma parte del programa Sócrates de la Comisión Europea, cuya misión es promover la cooperación europea en el campo de la enseñanza, desde la educación primaria a la enseñanza superior y desde las nuevas tecnologías al aprendizaje de adultos. Específicamente, el programa Erasmus, establecido en 1987, aborda el área de la enseñanza superior, es decir, la movilidad de los estudiantes universitarios. En términos sencillos, Erasmus te ofrece la posibilidad única de estudiar en otro país europeo durante un periodo de entre 3 a 9 meses. Esta estancia le aportará al estudiante no sólo enriquecimiento cultural y lingüístico sino también académico a través de la convalidación de estudios y créditos.

Los países participantes son los veinticinco estados miembros de la Unión Europea, los tres países del Espacio Económico Europeo (Islandia, Liechtenstein y Noruega) y Chipre. Orientado a la enseñanza superior, tiene como objetivo *«mejorar la calidad y fortalecer la dimensión europea de la enseñanza superior fomentando la cooperación transnacional entre universidades, estimulando la movilidad en Europa y mejorando la transparencia y el pleno reconocimiento académico de los estudios y cualificaciones en toda la Unión»*.

Cuando un estudiante obtiene una beca Erasmus comienza un proceso laborioso en el que tendrá que gestionar mucho trámite burocrático por parte de la universidad origen y la universidad destino. Actualmente en la Universidad de Castilla-La Mancha este proceso

no está automatizado. A continuación se muestran los inconvenientes que supone esto y las soluciones que se pueden llevar a cabo aplicando TICs a la gestión de la beca Erasmus. Para saber más sobre el programa Erasmus y todos los aspectos que contiene se puede revisar la normativa. (Véase Anexo B)

1.1 Enfoque de la problemática existente

El camino que debe seguir un alumno cuando se le ha concedido una beca Erasmus es realmente tedioso, implica un largo proceso y actualmente no existen herramientas que faciliten su gestión. El sistema propuesto en este PFC permitirá a todos los organismos implicados, alumnos, coordinadores de programa, coordinadores de centro y coordinador institucional, realizar todas las gestiones de manera electrónica.

En primer lugar cuando un alumno recibe la beca Erasmus debe preparar su contrato de estudios. Esto no es un proceso sencillo ya que requiere de múltiples pasos:

- Recoger todos los formularios a rellenar en la ORI (*Oficina de Relaciones Internacionales*).
- Informarse de los coordinadores que le han sido asignados. (Profesor de movilidad, coordinador de centro y coordinador institucional).
- Organizar una o varias reuniones con el profesor de movilidad hasta que se termine de realizar el contrato de estudios. La realización de este documento no es nada trivial ya que coger el plan de estudios de la universidad destino e intentar buscar asignaturas similares en créditos y temática a las que desea coger el alumno. Aparte de esto hay que tener en cuenta el periodo en el que se imparte la asignatura, ya que puede darse el caso que la asignatura se curse en el primer cuatrimestre y el alumno tenga una beca Erasmus para el segundo cuatrimestre.
- Organizar una reunión con el coordinador de centro que comprobará que está todo correcto en el contrato de estudios y no se incumple ningún requisito especificado en la normativa (Véase Anexo B).
- Acudir de nuevo a la ORI donde dejará el contrato de estudios para que sea firmado por el coordinador institucional.
- Cuando el alumno sea informado, recogerá de la ORI el contrato de estudios ya finalizado.

Después de que el estudiante llega al país destino y se dispone a elaborar su horario se da cuenta de que en la mayoría de las ocasiones tiene que modificar su contrato de estudios debido a sucesos imprevistos. La asignatura se imparte en el idioma nativo del país, la asignatura se suspende por falta de alumnos, el horario es incompatible con otra asignatura matriculada, etc.

El proceso de modificaciones del contrato de estudios es complicado ya que todas las comunicaciones hay que hacerlas mediante email y se tiene que mandar el contrato de estudios mediante fax. Puede ocurrir que algunos coordinadores no tengan una versión definitiva del contrato de estudios y se produzcan problemas de coherencia del mismo.

Para realizar las modificaciones del contrato de estudios hay que realizar otra serie de pasos. Primeramente informar por email al profesor de movilidad en España y que de su visto bueno a todos los cambios que el alumno desea realizar. Seguidamente, el estudiante debe enviar por fax su contrato de estudios con las modificaciones pertinentes para que el profesor de movilidad lo firme. Este después tendría que pasar el contrato modificado al coordinador de centro para que firmase las modificaciones y este a su vez al coordinador institucional. Después de este tedioso proceso se enviaría un fax a la universidad destino con el contrato de estudios modificado final.

Una vez finalizada la estancia del alumno Erasmus obtendrá un acta de la universidad de destino con las calificaciones obtenidas en las asignaturas cursadas allí. El alumno al regresar a España deberá entregar ese acta al coordinador de centro para que sea revisada y ponga las calificaciones adecuadas en el expediente del alumno. Este paso no es automático, ya que hay que realizar una conversión de notas antes de poder rellenar el expediente. En todos los países que participan en el convenio Erasmus no se utiliza el mismo sistema de calificación. Existe una tabla de conversión europea donde el coordinador de centro tiene que mirar e ir calculando la equivalencia de todas las notas según el país destino del estudiante.

Hay que tener en cuenta la problemática de todos los pasos expuestos anteriormente. Ya que no siempre es fácil preparar reuniones presenciales con los coordinadores por falta de disponibilidad o tiempo. A parte de lo cansado que puede resultar tener que organizar tantas reuniones para revisar el contrato de estudios y llevarlo a tantos sitios distintos.

1.2 Estructura del documento

El documento está compuesto de seis capítulos y cinco anexos:

- **Capítulo 1 - Introducción.** Consta de la introducción, enfoque del proyecto y estructura del documento.
- **Capítulo 2 - Objetivos del proyecto.** Enumeración y descripción de cada uno de los objetivos propuestos.
- **Capítulo 3 - Antecedentes, Estado del Conocimiento.** Para este proyecto se ha realizado estudios sobre diferentes tecnologías y herramientas. En este capítulo se comenta cual es la situación actual de todas estas herramientas y tecnologías.
- **Capítulo 4 - Gestión Automática de los Intercambios entre Estudiantes Sócrates/Erasmus.** En este capítulo se describen los diferentes elementos que componen el sistema, métodos empleados, diagramas que muestran la estructura de la base de datos, diagramas de clase, diagramas de casos de uso, algoritmos, etc. que se han llevado a cabo para la realización del proyecto.
- **Capítulo 5 - Resultados.** Resultados obtenidos tras la realización del proyecto y un análisis de los posibles costes que supondrían la futura implantación del sistema.
- **Capítulo 6 - Conclusiones y Propuestas.** Cuales han sido las conclusiones obtenidas, y algunas propuestas que pueden mejorar el sistema actual.
- **Bibliografía.** Bibliografía y referencias web utilizadas para la realización del proyecto.
- **Anexo A - Manual de Usuario.** Existe un manual de usuario para cada uno de los roles del sistema. Se muestra el funcionamiento de cada uno de los módulos disponibles para cada rol.
- **Anexo B - Normativa Erasmus.** Normativa Erasmus vigente en la Escuela Superior de Informática de Ciudad Real.
- **Anexo C - Contrato de Estudios.** Formulario tradicional del contrato de estudios.
- **Anexo D - Ficha de Reconocimiento de Estudios.** Formulario tradicional de la ficha de reconocimiento de estudios.
- **Anexo E - Tabla de Conversión de Calificaciones Europea.** Tabla donde se especifican los baremos de calificación de notas de algunos países que participan en el convenio Erasmus.

Objetivos del Proyecto

Dada la problemática expuesta en el capítulo anterior, lo que se pretende con la realización de este proyecto es crear un sistema de información que sea capaz de gestionar los intercambios para los estudiantes Erasmus, de manera que se facilite la labor del alumno y la cooperación entre las distintas personas implicadas (profesor de movilidad, coordinador de centro, coordinador institucional y administrador).

Actualmente existe un proceso claro y diferenciado de las etapas a seguir por un estudiante cuando quiere cursar una beca Erasmus. El problema es que este procedimiento no está informatizado y es largo y tedioso. La normativa es extensa y al no ser un proceso guiado puede resultar pesado para el alumno. El desarrollo de una herramienta informática guiará al alumno en todo el proceso facilitando su labor. El sistema ayudará a aclarar todos los pasos a seguir tanto por el estudiante como por el resto de usuarios, coordinador de programa, coordinador de centro, coordinador institucional, etc.

En definitiva el objetivo principal es **diseñar y desarrollar un Sistema de Información para la gestión del intercambio entre estudiantes Erasmus.**

Este objetivo principal se divide en una serie de sub-objetivos relevantes:

- **Automatización de tareas en el proceso global.-** Se pretende realizar una herramienta de control que facilite la interacción entre todas las personas que deben intervenir en el proceso del alumno Erasmus. Se intentará migrar todo el proceso tradicional que conlleva todo el procedimiento Erasmus a un procedimiento controlado digitalmente. Se seguirá una evolución del transcurso de la beca desde que el alumno comienza a realizar el plan de estudios hasta que finalmente termina el periodo de beca y obtiene las calificaciones pertinentes. Todos los organismos implicados podrán realizar las gestiones de forma rápida y eficaz.
- **Reglas de identificación mediante sistemas de firmas digitales.-** Todos los pasos de este proceso requieren de una serie de firmas por parte de los diferentes responsables. El sistema permitirá que los documentos generados se puedan firmar de manera electrónica,

para lo cual la herramienta tendrá implementado un sistema de firmas digitales para la gestión de dichos documentos. De esta manera se asegura la identidad de la persona que firma y se asegura la integridad del documento.

- **Sistema flexible y escalable constituido por componentes independientes.-** Cada uno de los componentes que forman parte del sistema se construirá de forma independiente de tal forma que cambios en la normativa no afectan directamente al funcionamiento global del sistema. Un sistema flexible y escalable facilitará su mantenimiento futuro.
- **Adaptación de la herramienta a cada tipo de usuario y sus privilegios.-** En este tipo de sistemas, la identificación personal de cada usuario y las transacciones seguras son fundamentales. Por esta razón, deberá existir una identificación por parte de los usuarios antes de utilizar la herramienta. Ésta tendrá implementados un sistema de permisos y privilegios, ya que habrá distintos roles o usuarios que podrán hacer uso del sistema. Además la herramienta será flexible y escalable a la hora de poder añadir nuevos roles. La apariencia de esta y las actividades que pueda desempeñar cada usuario dependerán del rol que tenga asignado, así como los privilegios asociados a su rol y perfil.
- **Dotar al sistema con capacidad de aprendizaje para la resolución de conflictos previos.-** En algunas ocasiones existen incompatibilidades de unas asignaturas de la universidad origen con otras de la universidad destino que tanto el alumno como los coordinadores implicados no pueden prever de ante mano y por lo tanto siempre había que recurrir a modificaciones de matrícula. El sistema será capaz de anticiparse a este tipo de situaciones basándose en estadísticas y resultados de alumnos de años anteriores para decidir que es lo más adecuado para el futuro alumno.
- **Agente inteligente recomendador.-** El sistema contará con un agente que estará alerta para ayudar al estudiante siempre que se haya producido algún conflicto. Este agente mostrará posibles consejos para guiar al estudiante a tomar la decisión adecuada.
- **Sistema de control de eventos.-** Con este método los roles implicados podrán darse cuenta de cualquier cambio producido una vez entren en la herramienta. El sistema avisará con cualquier evento producido, el autor del mismo y en la fecha y hora que fue producido. Proporcionará una mayor facilidad a la hora de seguir el control de los cambios que se van produciendo en el contrato Erasmus de cualquier alumno; o de los coordinadores para llevar el control de todos sus alumnos.
- **Cálculo automático de calificaciones.-** Existe una tabla de conversión de calificaciones europeas en la que se reflejan los baremos de calificación de cada país. La

herramienta será capaz de realizar la conversión de notas aplicando los baremos del país de origen y transformarlos en válidos para la universidad de Castilla-La Mancha.

- **Diseño multicapa.-** El diseño del sistema debe estar dividido en tres capas: Presentación, Dominio, y Persistencia de tal forma que facilite el desarrollo y mantenimiento futuro.
- **Generación digital de contratos de estudios y actas de notas.-** La herramienta será capaz de generar contratos de estudios y actas de notas en formato PDF. Estos documentos estarán firmados digitalmente por lo que serán igualmente válidos que los contratos tradicionales en papel.
- **Accesible vía web.-** La herramienta será utilizada por diferentes roles y en diferentes lugares. Dada la naturaleza del problema los estudiantes tendrán que acceder al sistema desde muy diferentes ubicaciones en sus universidades de destino. Por este motivo se ha pensado hacer un sistema de gestión de contenidos web y no una aplicación de otro tipo.
- **Control de flujo con máquinas de estados.-** El contrato de estudios es un elemento que puede ser accedido por cualquiera de los roles implicados. Por este motivo es imprescindible llevar el control de flujo del mismo con una máquina de estados. De esta forma se evitará que cualquier agente pueda ver o modificar el contrato cuando no le corresponda.

Antecedentes, Estado del Conocimiento

En este capítulo se describe el estudio realizado de diversas tecnologías, métodos y algoritmos que son necesarios para la consecución de los objetivos planteados en el capítulo 2. Es necesario revisar previamente las diversas tecnologías que se tienen disponibles. Una vez hecho esto, se analizará el problema planteado y se optará por elegir los métodos más adecuados para el desarrollo de la herramienta.

3.1 Desarrollo Web Estático y Dinámico

3.1.1 HTML Estático

HTML (*HyperText Markup Language*) es un lenguaje utilizado para escribir documentos para la WWW (*World Wide Web*). Este lenguaje guarda una cierta estructura que permite que se muestre correctamente en los navegadores web independientemente del que se utilice. Básicamente consiste en escribir el texto que queremos que se muestre en la página web y posteriormente añadir una serie de marcas que controlarán los diferentes aspectos de la presentación y comportamiento de los elementos. Es un lenguaje muy simple y no es necesario utilizar ningún editor especial para escribirlo.

Una de las claves del éxito del HTML es la coherencia que se puede establecer con un conjunto de documentos. Normalmente se suele partir de un documento inicial el cuál tiene enlaces a otros documentos a los que podemos acceder con un simple clic de ratón.

Se necesitan dos partes para que el servicio web se pueda establecer.

- Servidor: Es la entidad que se encarga de poner a disposición del público general el contenido web.
- Cliente: Es el navegador web por el cuál un usuario accede al contenido web expuesto por el servidor. Existen múltiples navegadores (Mozilla Firefox, Internet Explorer,

Google Chrome . . .) para diversos sistemas operativos en los cuáles la apariencia de la web no debería verse afectada por utilizar un navegador u otro. Aunque esto no es siempre cierto.

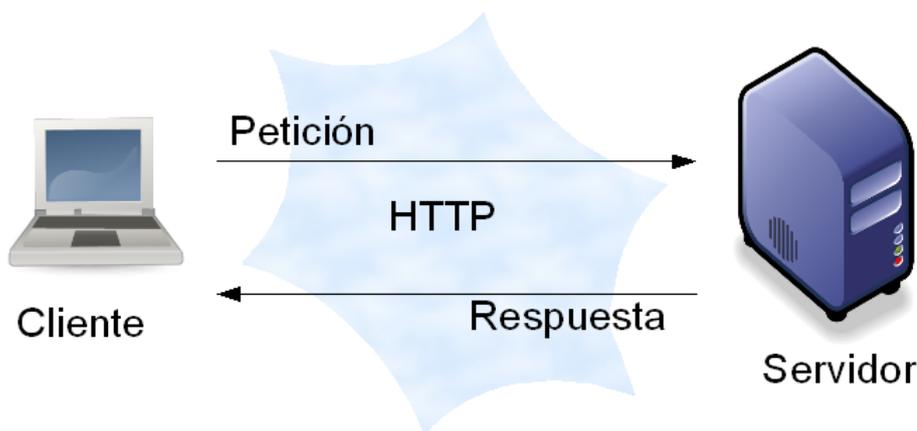


Figura 3.1: Esquema de como se produce la comunicación entre cliente y servidor

En el lenguaje HTML se pueden utilizar multitud de opciones para dar una apariencia vistosa y ordenada al documento.

- **Estilos de Caracteres:** Estos estilos son etiquetas que afectan a palabras o caracteres dentro de otras entidades de HTML modificando el aspecto de ese texto para que sea diferente del texto que lo rodea.
- **Colores:** Se indican en valores *RGB*, es decir, que para conseguir un color cualquiera se mezclan cantidades de rojo, verde y azul.
- **Hiperenlaces:** Permite indicar zonas de texto o imágenes que si son seleccionados por el lector del documento nos traslada a otros documentos HTML o otras zonas del documento actual.
- **Listas:** Con frecuencia se usan listas en los documentos. El lenguaje HTML incorpora unas listas con viñetas sencillas o también letras o números.
- **Tablas:** El modelo de tablas de HTML permite a los autores organizar datos (textos, texto preformateado, imágenes, vínculos, formularios, campos de formularios, otras tablas, etc.) en filas y en columnas de celdas.
- **Formularios:** Los formularios son una característica del estándar HTML que permite a los autores coleccionar información provista por los visitantes. Estos formularios pueden

ser útiles para recolectar información personal, información de contacto, preferencias u opiniones, o cualquier tipo de entrada por parte del visitante que el autor pueda necesitar.

Esta es la estructura básica de un documento HTML.

```
1 <html>
2 <head>
3 <title> Título del documento HTML.</title>
4 </head>
5 <body>
6 Cuerpo del documento.
7 </body>
8 </html>
```

3.1.2 HTML Dinámico

Con el paso del tiempo la programación web ha ido avanzando adquiriendo cada vez nuevos objetivos. Llegó el momento en el que el HTML estático se quedó corto y hubo que utilizar una tecnología que permitiese realizar desarrollos más complejos y dinámicos. Se puede imaginar lo costoso que sería realizar una página web que mantuviese un periódico digital desarrollado solo con HTML estático. Necesitaría de actualizaciones diarias para alojar todas las noticias y sería un trabajo demasiado laborioso ya que habría que crear nuevas páginas web a diario. Sin embargo, este mismo desafío realizado con HTML dinámico simplifica de manera muy considerable el esfuerzo a realizar.

También se puede hacer que los elementos HTML interactúen con el usuario agregando *controladores de eventos* que se ejecutan al producirse algunos eventos. A continuación se muestran unos cuantos y un ejemplo concreto del evento *onclick*.

- **onabort:** Se activa cuando el usuario para la descarga.
- **onblur:** Se activa cuando el usuario pierde el foco de entrada.
- **onclick:** Se activa cuando el usuario hace clic con el ratón en un sitio determinado.
- **ontimeout:** Se activa cuando hay un error que impide la realización de la petición.

```
1 <html>
2 <body>
3
4 <form>
```

```
5 <input type="Button" value="Hola_a_todos" onclick="cambiaTitulo  
  (' Adios_a_todos ')">  
6 </form>  
7  
8 </body>  
9 </html>
```

El HTML Dinámico o *DTHML* es el lenguaje que se compone de HTML estático y la combinación de otra serie de lenguajes para dar dinamismo a la página web. Ahora, en vez de tener hipertexto estático, se obtendrá una web con la que el usuario podrá interactuar en tiempo real. Imágenes que van cambiando con el tiempo o al pasar el ratón por encima de ellas, menús desplegables, formularios . . .

Los lenguajes con los que se combina el html estático para producir el dinamismo pueden estar del lado del servidor o del lado del cliente.

Lado del servidor: Lenguajes que se basan en el servidor para ejecutar sus scripts. Cuando una página es solicitada por parte de un cliente, el servidor ejecuta los scripts y genera una página resultado, que envía al cliente. La página resultado contiene únicamente código HTML.

- PHP (*PHP Hypertext Pre-processor*)
- JSP (*JavaServer Pages*)
- ASP (*Active Server Pages*)

Lado del cliente: El DHTML se desarrolla en el ámbito de una página web, cuando la página se está viendo en la pantalla de los usuarios , es decir, en los navegadores. En estos casos, para realizar cualquier tipo de efecto o interactividad en la página se tiene como recurso al navegador, por eso se llama de cliente.

- Javascript (Véase sección 3.1.2.4)

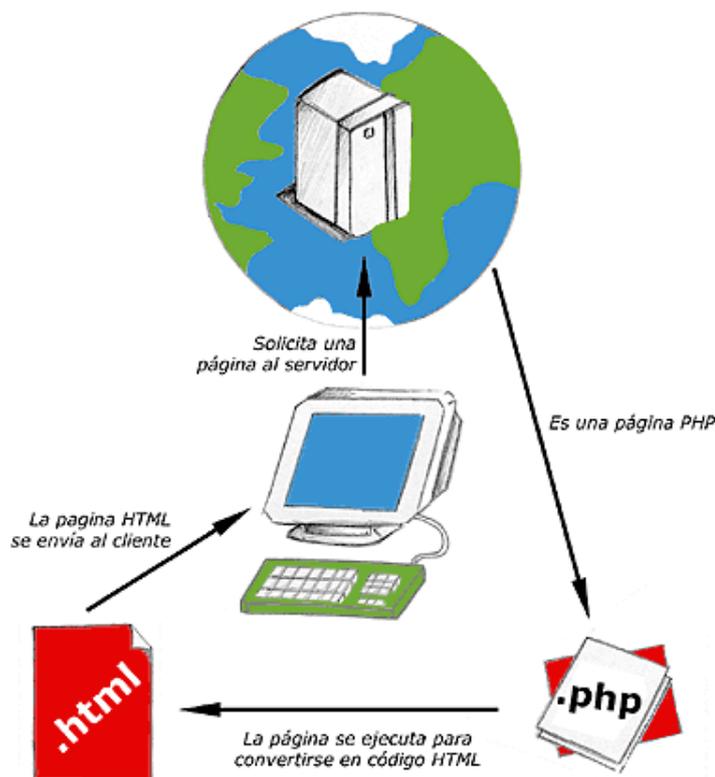


Figura 3.2: Esquema del funcionamiento de las páginas con scripts del servidor

3.1.2.1 PHP

PHP[10] es un lenguaje de programación interpretado que sirve principalmente para realizar páginas web dinámicas. Funciona principalmente del lado del servidor, lo que hace que el servidor si que tenga que soportar esta tecnología, pero los navegadores cliente que hagan uso de estas páginas web son independientes y no tienen por qué soportar PHP.

PHP puede procesar información de formularios, generar páginas con contenidos dinámicos, mandar y recibir cookies, etc.

Cuando el cliente hace una petición web, se interpreta el código PHP en el servidor y este genera de forma dinámica un documento HTML estático. Este código es enviado al navegador web del cliente que sabe interpretarlo y mostrarlo correctamente.

En PHP se permite la conexión a múltiples servidores de bases de datos y es compatible con muchos sistemas operativos.

En la actualidad PHP ha ido mejorando y hoy en día se utilizan otras versiones como PHP 5. En esta versión la principal novedad es la orientación a objetos, lo que le proporciona todas las ventajas de la misma.



Figura 3.3: Esquema del proceso de una petición web

- **Constructores y destructores con nombres fijos:** Son los que se encargan de resumir las tareas de inicialización y destrucción de los objetos. Ahora se han de llamar `__construct()` y `__destruct()`
- **Interfaces:** Las interfaces se utilizan en la POO (*programación orientada a objetos*) para definir un conjunto de métodos que implementa una clase. Una clase puede implementar varias interfaces o conjuntos de métodos. En la práctica, el uso de interfaces es utilizado muy a menudo para suplir la falta de herencia múltiple de lenguajes como PHP o Java.
- **Sistema de protección:** Se pueden utilizar los modificadores de acceso habituales de la POO. Estos modificadores sirven para definir qué métodos y propiedades de las clases son accesibles desde cada entorno.
- **Métodos, atributos estáticos:** Son las propiedades y funcionalidades a las que se puede acceder a partir del nombre de clase, sin necesidad de haber instanciado un objeto de dicha clase.
- **Clases y métodos abstractos:** Las clases abstractas no se pueden instanciar, se suelen utilizar para heredarlas desde otras clases que no tienen porque ser abstractas. Los métodos abstractos no se pueden llamar, se utilizan más bien para ser heredados por otras clases, donde no tienen porque ser declarados abstractos.
- **Constantes de clase:** Se pueden definir constantes dentro de la clase. Luego se pueden acceder dichas constantes a través de la propia clase.

- **Clonación de objetos:** Si se desea, se puede realizar un objeto a partir de la copia exacta de otro objeto. Para ello se utiliza la instrucción *clone*. También se puede definir el método `__clone()` para realizar tareas asociadas con la clonación de un objeto.

3.1.2.2 JSP

JSP [27] (*Java Server Pages*) es un lenguaje que utiliza la tecnología Java para crear páginas web dinámicas. Ha sido desarrollado por Sun Microsystems. Para crear dinamismo los JSP pueden incluir código Java [5] o un lenguaje propio de etiquetas denominado *Jstl*.

La portabilidad de Java se transmite a JSP haciendo de él un lenguaje igual de portable y capaz de ser ejecutado en múltiples plataformas sin necesidad de cambios. Al poder utilizar Java como lenguaje es posible crear una buena estructura del programa dividiéndolo en clases que luego serán utilizadas por los JSP.

Las páginas JSP están compuestas de código HTML mezclado con unas marcas especiales para poder incluir código Java. Ambos lenguajes podrían estar combinados aunque se pueden crear páginas únicamente con código JSP.

Esto sería un ejemplo de una página sencilla escrita en JSP.

```
1 <html>
2 <body>
3
4 <% int hasta = 5;
5 <% for (int i = 0; i < hasta; i++)
6     out.println("Hola_mundo!<br/>"); %>
7
8 </body>
9 </html>
```

Uno de los servidores de aplicaciones escritas en lenguaje JSP más utilizado es *Tomcat* [1]. Está escrito en Java, por lo que funcionará en cualquier ordenador que tenga instalada la máquina virtual de Java. Está en constante evolución ya que sus últimas versiones son las 7.x y están disponibles desde Enero del 2011.

3.1.2.3 ASP

ASP [2] (*Active Server Pages*) es una tecnología propietaria de Microsoft. Consiste en un lenguaje que trata textos basados en Basic, y que se denominan VBScript (*Visual Basic Script*). Se utiliza en los servidores de Microsoft (*Internet Information Server* y *Personal*

Web Server). Al igual que otros lenguajes de este tipo como PHP y JSP, ASP se ejecuta en el servidor y puede combinarse con HTML y Javascript para realizar tareas interactivas. Igualmente permite acceder a bases de datos de una forma sencilla y rápida.

Esto sería un ejemplo de una página sencilla escrita en ASP.

```
1 <html>
2 <head>
3 </head>
4 <body>
5 <p> <% Response . Write " hola_mundo" %> </p>
6 </body>
7 </html>
```

Con el tiempo surgió una nueva versión llamada ASP.net [18] la cuál permitía escribir en cualquier lenguaje soportado por el .net Framework (*VB.net, C# o JScript.net*). Otra de las principales mejoras fue la introducción de la orientación a objetos. Aparte de las mejoras mencionadas anteriormente se produjeron otras como:

- Mayor rendimiento.
- Rapidez en la programación.
- Servicios web.
- Seguridad

3.1.2.4 Javascript

Javascript [15] es un lenguaje de sintaxis similar a Java o C que se utiliza en aplicaciones web. Se puede decir que este lenguaje nació de la necesidad de crear interacción entre el usuario y la web. Permite realizar páginas web dinámicas y mejora la interfaz de usuario a la vez que interactúa con él.

Son los navegadores web los encargados de interpretar estos códigos. Aunque puedan parecer iguales, Java y Javascript son dos lenguajes distintos con fines distintos. Javascript es un lenguaje interpretado, basado en prototipos, mientras que Java es un lenguaje más orientado a objetos.

Javascript forma parte del cliente, a diferencia de los lenguajes explicados anteriormente como JSP, PHP o ASP. Las sentencias JavaScript colocadas en una página web pueden

dar respuesta a eventos de usuario, tales como la pulsación de un botón del ratón, la entrada de datos en un formulario y la navegación por una página. Se podría crear una función JavaScript que permitiera verificar que la información introducida por el usuario en un campo de entrada de datos de un formulario (número de teléfono, código postal, número de tarjeta de crédito, etc.) tiene el formato correcto. En este caso, lo importante es que, sin necesidad de realizar ninguna transmisión de datos por la red, se puede validar dicha información, mostrando al usuario un cuadro de diálogo en caso de que ésta sea incorrecta.

3.1.3 CSS

CSS[13] (*Cascading Style Sheets*) es un lenguaje que se utiliza para definir el estilo en documentos HTML y XML. Sirve principalmente para separar el contenido de la presentación. Se puede controlar el estilo asociado a múltiples páginas web simplemente asociando un documento CSS a ellas.

3.1.3.1 Ventajas

- Se acelera considerablemente la actualización de la presentación de la web.
- Las hojas de estilo ayudan a adaptar la web al dispositivo externo desde donde se están mostrando (monitor, pda, móvil, etc).
- El documento HTML es más breve y conciso al no tener que escribir el código para darle formato.
- Aumenta la accesibilidad, ya que los usuarios pueden especificar la hoja de estilo con la que quieren mostrar el documento.
- Cualquier cambio que se haga en el estilo no afectan al contenido ya que son totalmente independientes.

3.1.3.2 ¿Cómo se aplica?

CSS funciona mediante reglas que se aplican a los elementos que existen en el documento. Por ejemplo:

```
h2 {background-color : red;}
```

`h2` es el selector e indica que todos los elementos `h2` se verán afectados por la propiedad `background-color` y tomarán el valor de `red`.

Existen varias formas de aplicar CSS a los documentos:

- Utilizando el elemento `<link>`, que deberá estar situado dentro del `<head>` del documento web. Es la forma elegante, ya que se separa totalmente el código de presentación del resto.

```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD_XHTML_1.1//EN">
2 <html>
3   <head>
4     <title>Título</title>
```

```
5     <link rel="stylesheet" type="text/css" href="css/miCss.
      css" />
6 </head>
7 <body>
8
9 </body>
10 </html>
```

- Utilizando el elemento `<style>`, que generalmente está situado dentro del `<head>`. Con esto se consigue que el estilo se aplique antes de que el documento se cargue.

```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD_XHTML_1.1//EN">
2 <html>
3   <head>
4     <title>hoja de estilo interna</title>
5     <style type="text/css">
6
7       body {
8         color: purple;
9         background-color: #d8da3d }
10
11      h1 {
12        font-family: Helvetica, Geneva, Arial, sans-serif;
13      }
14
15     </style>
16   </head>
17   <body>
18
19   </body>
20 </html>
```

- También se puede utilizar el atributo `<style>` dentro de `<body>`, mezclado con el documento. Esta es la forma menos elegante y recomendada ya que se pierde la ventaja de tener separado el contenido de la presentación.

3.1.4 XML

XML (*Extensible Markup Language*) es un metalenguaje que permite definir otros lenguajes de marcas. La versión 1.0 de XML está basado en SGML. Se trata de un subconjunto de este último en el que se han eliminado las partes menos útiles y más complejas. XML al igual que su padre SGML es un sublenguaje y a diferencia de HTML los elementos que lo componen, pueden dar información sobre lo que contienen y no necesariamente de su estructura física o presentación.

Algunos de los objetivos [13] de XML son:

- XML debe ser utilizable directamente sobre internet.
- XML debe soportar una amplia variedad de aplicaciones.
- XML debe ser compatible con SGML.
- Debe ser fácil escribir programas que procesen documentos SGML.
- Los documentos XML deben ser legibles por un humano y razonablemente claros.
- El diseño de XML debe ser preparado rápidamente.
- El diseño XML debe ser formal y conciso.

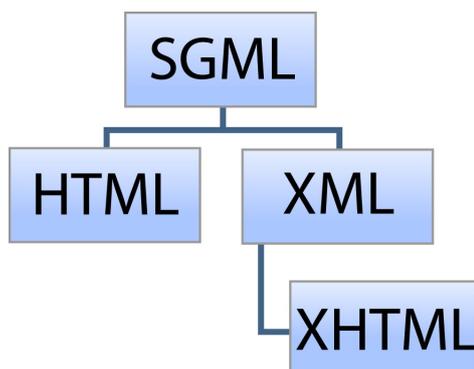


Figura 3.4: Comparación entre SGML, HTML y XML

Un documento XML está formado por una estructura lógica y otra física. La primera de ellas engloba elementos como declaraciones y comentarios. La segunda se refiere a las entidades que componen el documento.

- **DTD:** (*Document Type Definition*): Archivo que contiene la especificación de la estructura lógica de cada documento y define los elementos y atributos que contiene. El DTD es opcional en el XML.

- **XSL:** (*eXtensible Stylesheet Language*): Define el lenguaje de estilo de los documentos escritos para XML.
- **XLL:** (*eXtensible Linking Language*): Define el modo de enlace establecido entre los diferentes enlaces.
- **XUA:** (*XML User Agent*): Es usado para la normalización de navegadores XML.

```
<CATALOG>
  <PLANT>
    <COMMON>Bloodroot</COMMON>
    <BOTANICAL>Sanguinaria canadensis</BOTANICAL>
    <ZONE>4</ZONE>
    <LIGHT>Mostly Shady</LIGHT>
    <PRICE>$2.44</PRICE>
    <AVAILABILITY>031599</AVAILABILITY>
  </PLANT>
  <PLANT>
    <COMMON>Columbine</COMMON>
    <BOTANICAL>Aquilegia canadensis</BOTANICAL>
    <ZONE>3</ZONE>
    <LIGHT>Mostly Shady</LIGHT>
    <PRICE>$9.37</PRICE>
    <AVAILABILITY>030699</AVAILABILITY>
  </PLANT>
  <PLANT>
    <COMMON>Marsh Marigold</COMMON>
    <BOTANICAL>Caltha palustris</BOTANICAL>
    <ZONE>4</ZONE>
    <LIGHT>Mostly Sunny</LIGHT>
    <PRICE>$6.81</PRICE>
    <AVAILABILITY>051799</AVAILABILITY>
  </PLANT>
</CATALOG>
```

Figura 3.5: *Ejemplo de código XML*

Los documentos XML se pueden dividir en dos grupos:

- **XML bien formado:** Un documento XML está bien formado cuando puede ser analizado por cualquier analizador sintáctico de forma correcta. Para ello tiene que cumplir una serie de reglas:
 - La estructura de todo documento XML ha de ser jerárquica.
 - Un documento XML puede contener etiquetas vacías.
 - El elemento raíz es único.
 - Los valores de los atributos deben estar encerrados entre comillas.
 - XML es sensible a mayúsculas.
 - Las estructuras y tipos de elementos deben tener asignados un nombre.

- No debe haber etiquetas aisladas.
- **XML valido:** Un documento XML es válido cuando además de cumplir con las reglas necesarias para que sea un documento bien formado, sigue una estructura y semántica definida por un DTD.

3.2 Usabilidad Web

Antes de dar unas definiciones más técnicas y concretas se intentará deducir que significado tiene la *Usabilidad Web*. Hoy en día es muy habitual que cualquier persona use la web para múltiples fines. Cuando un usuario entra en una web por primera vez puede identificar cuando un sitio está bien diseñado o estructurado, es decir, si el sitio resulta *fácil de usar*.

Se puede decir entonces que la *usabilidad web* nos ayuda a identificar cuando una web es fácil de usar, los métodos para hacer que una web sea usable y como construirla así desde el principio. A parte de todo esto, no se puede olvidar que la web además de ser fácil de usar tiene que cumplir con el objetivo para el que fue diseñada.

A continuación se exponen unas definiciones más formales:

- La usabilidad se refiere a la capacidad de un software de ser comprendido, aprendido, usado y ser atractivo para el usuario, en condiciones específicas de uso [ISO/IEC 9126].
- Usabilidad es la eficiencia y satisfacción con la que un producto permite alcanzar objetivos específicos a usuarios específicos en un contexto de uso específico [ISO/IEC 9241].
- Un atributo de calidad que mide lo fáciles de usar que son las interfaces web [26].
- ¡No me hagas pensar! Primera norma de usabilidad. Se trata de que nada importante esté a más de dos clics de distancia. [23]

Se puede decir que la usabilidad se encarga de analizar lo fácil e intuitiva que resulta ser una página web. Estas son las características que se miden:

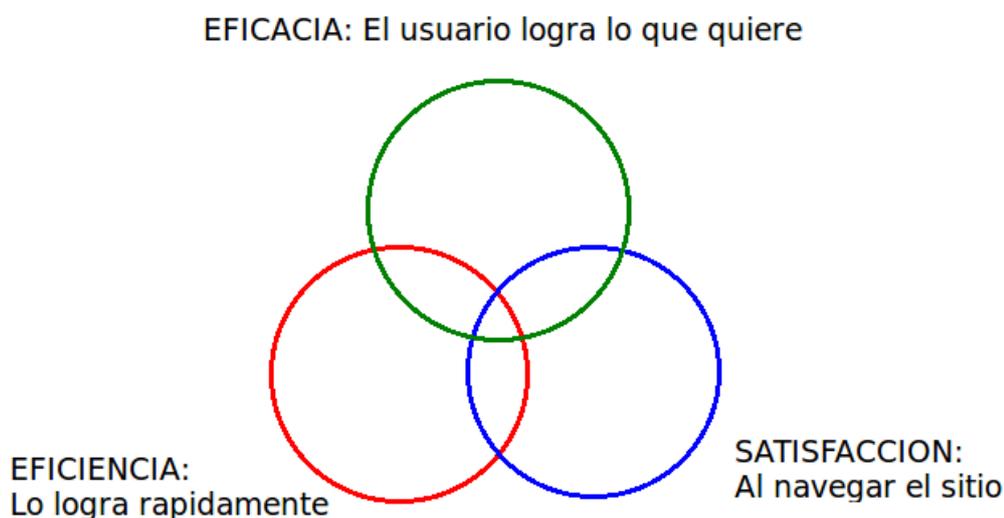


Figura 3.6: Relación entre Eficacia, Eficiencia y Satisfacción en términos de usabilidad web

Es necesario encontrar un equilibrio en estas tres características para poder alcanzar la máxima usabilidad web. Cuando un usuario visita una web intenta lograr un objetivo, ya sea comprar un artículo en una página de subastas u obtener información sobre la segunda guerra mundial para su tesis. Si el usuario alcanza su objetivo entonces podremos hablar de que existe una eficacia alta. Si además lo ha conseguido sin mucha dificultad a través de la navegación por el sitio web el usuario estará satisfecho. Y por último, si no ha tardado mucho tiempo en conseguir su objetivo existe una alta eficiencia. En resumen se podría decir que el usuario ha estado navegando a través de una página web con buena usabilidad.

Según Steve Crug [23], si se quiere diseñar páginas webs eficientes, se tiene que aprender a convivir con tres factores sobre cómo se utiliza realmente la Web.

- **No leemos las páginas Web, las hojearmos.** Es un hecho que normalmente las páginas web que visitan los usuarios lo hacen para echarles un vistazo rápido y no se paran a leer todo el contenido con detenimiento. Normalmente se suele buscar alguna frase atrayente que capte la atención.
- **No tomamos decisiones óptimas. Nos es suficiente.** Se refiere al hecho de que el usuario no se para a pensar muy detenidamente en lo que está buscando, simplemente cuando ve algún enlace en la web que cree que le puede llevar a su objetivo hace clic en él. Este comportamiento viene motivado por la prisa del usuario en alcanzar su objetivo, porque las consecuencias de equivocarse no son importantes o porque adivinar puede ser más divertido.
- **No averiguamos el funcionamiento de las cosas. Nos las arreglamos.** No es nueva la idea de que cuando se prueba algo sin tener mucho conocimiento al respecto y funciona, nadie se molesta en saber con certeza y fundamento porqué ha sido así. Muchos de los diseños web son mal utilizados por los usuarios con fines distintos para los que fueron creados. Un caso real es el de introducir en la barra de búsqueda de un buscador la url completa del sitio de destino. No es el uso correcto, pero realmente se llega a donde se quiere. Este comportamiento se debe a que realmente al usuario le importa poco el diseño de la web si realmente está alcanzando su objetivo.

3.3 Almacenamiento de la Información

El almacenamiento de información es una de las principales funciones que tiene una computadora. De esta manera se podrá recordar la información procesada en cualquier momento que se necesite. La información suele estructurarse en unidades lógicas llamadas archivos. Los archivos suelen guardarse en unidades físicas como son el disco duro, discos compactos, diskettes, etc.

Aunque una parte principal es la manera en la que se va a organizar dicha información a la hora de ser guardada. Es una decisión importante a la hora de realizar el diseño de un sistema de información ya que esto puede determinar el funcionamiento y el rendimiento futuro de todo el sistema.

3.3.1 Ficheros

Los ficheros permiten guardar información de forma permanente en una memoria física una vez se ha terminado de ejecutar la aplicación con la que se estaba trabajando. De esta forma se evita la pérdida de información como sucede con los arrays o variables de programa que se pierden una vez cerrada la aplicación.

El modo de trabajar con un fichero tiene una serie de pasos:

- **Abrir fichero:** Se crea un objeto de una determinada clase de acceso al fichero.
- **Realizar operaciones:** Se realizan operaciones de lectura, escritura, posicionamiento, etc.
- **Cerrar fichero:** Para que se guarden todos los datos correctamente.

Los ficheros también cuentan con unos determinados modos de acceso según el uso que se les quiera dar:

- **Modo lectura:** Se puede acceder al fichero para visualizar su contenido, pero no podrá ser modificado.
- **Modo escritura:** Se puede añadir contenido al fichero.
- **Modo lectura/escritura:** Se puede visualizar el contenido y modificarlo.

Dentro del sistema de información de una empresa se puede utilizar ficheros para gestionar dicho sistema. A medida que las necesidades de información de la empresa van creciendo, se irán creando nuevos programas que accedan a estos ficheros y vayan añadiendo la información que sea necesaria junto con la ya existente. Puede ser que los distintos

departamentos de la empresa hagan uso de los mismos ficheros con distintas aplicaciones informáticas. Todo esto puede suponer un problema ya que los nuevos programas tienen que ser creados teniendo en cuenta la sintaxis con la que se manipulan los ficheros del sistema. Muchas veces esto no es posible de realizar ya que la propia empresa compra programas externos que no guardan relación con el tipo de sistema de ficheros utilizados. Esto puede conllevar problemas de aislamiento y redundancia de datos.

Por todo lo expuesto anteriormente se suelen utilizar más frecuentemente sistemas de bases de datos.

3.3.2 Bases de datos

Una base de datos (*BD*) es una colección de datos sobre el mismo contexto, que está organizada y estructurada para su uso posterior de una forma eficaz. Normalmente los datos almacenados en una base de datos son persistentes, lo que quiere decir que perduran en el tiempo y pueden ser recuperados en cualquier momento. A diferencia de los datos temporales que genera el computador durante la ejecución de los programas, los cuales circulan por las líneas de comunicación (buses) y se almacenan en la memoria cache o memoria secundaria.

En una base de datos, la información se suele organizar en tablas donde cada fila es un registro y cada columna un atributo.

Código	Título	Director	Año
1	El Padrino	Francis Ford Coppola	1972
2	Pulp Fiction	Quentin Tarantino	1994
3	Infiltrados	Martin Scorsese	2006

Tabla 3.1: Ejemplo de una tabla de una base de datos

Las bases de datos suelen utilizarse en multitud de aplicaciones para el almacenamiento masivo de datos. Trae consigo muchas ventajas respecto a otros sistemas de almacenamiento de la información como son los ficheros:

- Eliminación de información redundante.
- Eliminación de información inconsistente.
- Permite compartir información. Varios sistemas o usuarios pueden utilizar la misma entidad.
- Permite que la información sea íntegra.

- Independencia entre datos y programas. Se podrían sustituir los programas que utilizan los datos sin tener que realizar ningún tipo de cambios en estos.

Un sistema gestor de bases de datos (*SGBD*) es un software que hace de intermediario entre el usuario y la BD, o bien, entre aplicaciones (*software*) que pretendan acceder o realizar operaciones sobre la BD.

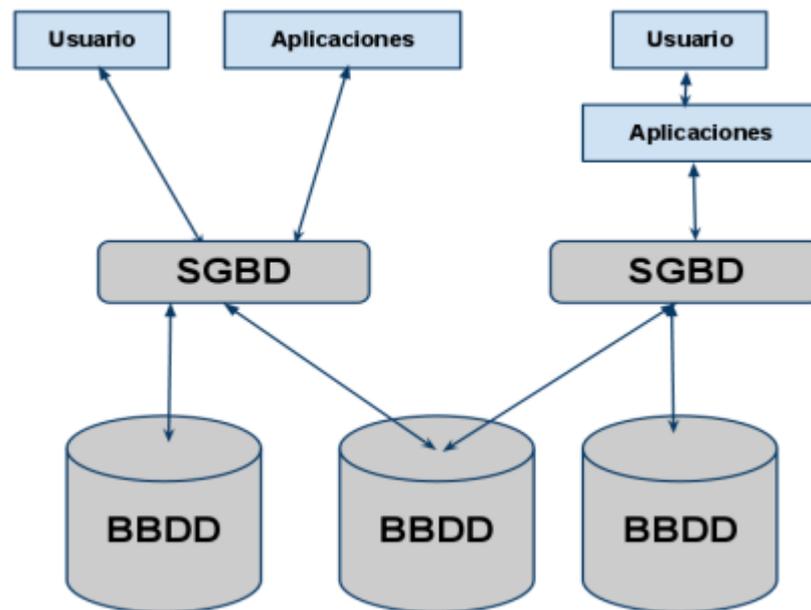


Figura 3.7: Esquema que muestra el SGBD junto a las aplicaciones y BBDD

El SGBD, la BD, los usuarios y las aplicaciones no tienen por qué encontrarse físicamente localizados en el mismo lugar o la misma máquina. Un usuario localizado en Madrid podría solicitar a un SGBD localizado en París, insertar un nuevo registro en una BD que se encuentra almacenada en un ordenador localizado en Berlín.

Algunos de los objetivos más importantes de un SGBD son los siguientes:

- **Abstracción de la información:** La forma en la que se almacena la información físicamente es transparente al usuario. Se omiten detalles técnicos y simplemente se muestra la estructura lógica en la que se almacena la información. De esta forma, un mayor rango de posibles usuarios, puede hacer uso de la BD, independientemente de los conocimientos técnicos que posean.
- **Independencia:** Cambios en la estructura lógica o física de la BD no repercuten en las aplicaciones que hacen uso de ella.
- **Consistencia:** La información de una BD no se actualiza de cualquier forma si no que se hace de acuerdo a una serie de reglas o normas. Este tipo de normas/reglas se pueden

definir o establecer desde el SGBD. Por ejemplo, se podría crear una tabla de usuarios en una BD en la que no se pudieran insertar usuarios con una edad inferior a 18.

- **Seguridad:** La información almacenada en una BD puede llegar a ser realmente valiosa y por ello los SGBD permiten la creación de roles de usuario, grupos de usuarios y permisos asociados a éstos. No todos los usuarios pueden realizar todas las operaciones sobre las tablas de una BBDD.
- **Manejo de transacciones:** Cuando se realiza una operación sobre una base de datos, ésta podría ser interrumpida por una determinada causa. El SGBD debe garantizar la integridad de los datos, de tal forma que si la operación fue interrumpida, los valores puedan restablecerse al estado en el que se encontraban antes de la operación.
- **Tiempo de respuesta:** El SGBD debe realizar las operaciones en el menor tiempo posible.

Existen muchos tipos de sistemas gestores de bases de datos distintas aunque algunas de las más utilizadas son *Access*, *SQL Server*, *MySQL* y *ORACLE*.

3.3.2.1 Access

Es un programa [16] que permite crear ficheros de bases de datos relacionales que pueden ser fácilmente gestionadas por una interfaz gráfica sencilla. Se utiliza en los sistemas operativos Microsoft Windows, para la gestión de bases de datos y está orientado a ser usado en un entorno personal o en pequeñas organizaciones. Es un componente de la suite Microsoft Office. Además, estas bases de datos pueden ser consultadas por otros programas. Dentro de un sistema de información, entraría dentro de la categoría de gestión, y no en la de ofimática, como podría pensarse. Este programa permite manipular los datos en forma de tablas, crear relaciones entre tablas, consultas, formularios para introducir datos e informes para presentar la información.

3.3.2.2 SQL Server

Es un sistema de bases de datos [19] creado por Microsoft basado en el modelo relacional. SQL Server es la solución de gestión de la información de Microsoft para la próxima generación, que procurará mayor escalabilidad, disponibilidad y seguridad a la información empresarial y las aplicaciones de análisis al tiempo que simplificará su creación, implantación y gestión.

Algunas de las principales características son las siguientes:

- Desarrollar e implantar aplicaciones empresariales más escalables, fiables y seguras.

- Optimizar la productividad del sector TI reduciendo la complejidad en la creación, implantación y administración de las aplicaciones de bases de datos.
- Aumentar las capacidades de los desarrolladores con un entorno de desarrollo valioso, flexible y actual para que creen bases de datos más seguras.
- Compartir datos a través de múltiples plataformas, aplicaciones y dispositivos para facilitar la interconexión entre sistemas internos y externos.
- Ofrecer soluciones de inteligencia empresarial que ayuden a tomar decisiones con fundamento y aumentar la productividad por toda la empresa.
- Controlar los costes sin sacrificar el rendimiento, la disponibilidad ni la fiabilidad.

3.3.2.3 MySQL

Es uno de los motores de bases de datos [21] más usados en la actualidad. Una de las principales razones es que se puede usar de manera gratuita. Es un gestor de bases de datos rápido y sencillo de usar.

Algunas de las principales características son las siguientes:

- **Es un SGBD.** (*Sistema gestor de base de datos*) Una base de datos es un conjunto de datos y un gestor de base de datos es una aplicación capaz de manejar este conjunto de datos de manera eficiente y cómoda.
- **Permite diseñar y crear bases de datos relacionales.** Una base de datos relacional es un conjunto de datos que están almacenados en tablas entre las cuales se establecen unas relaciones para manejar los datos de una forma eficiente y segura. Para usar y gestionar una base de datos relacional se usa el lenguaje estándar de programación SQL.
- **Es Open Source.** El código fuente de MySQL se puede descargar y está accesible a cualquiera, por otra parte, usa la licencia GPL para aplicaciones no comerciales.
- **Es un SGBD muy rápido, seguro y fácil de usar.** Gracias a la colaboración de muchos usuarios, MySQL ha ido mejorando optimizándose en velocidad. Por eso es uno de los SGBD más usados en Internet.
- **Está ampliamente extendido** Es utilizado por una comunidad extensa de usuarios y un elevado número de programas informáticos.

3.3.2.4 ORACLE

Oracle [14] es un sistema de gestión de base de datos objeto-relacional. Está formado de una potente herramienta cliente/servidor para la gestión de bases de datos. Es un producto extendido y conocido en todo el mundo. El principal inconveniente es que su elevado precio no hace que esté disponible para cualquier usuario, sino más bien para grandes empresas.

Como se trata de una herramienta cliente/servidor, primeramente sería necesaria la instalación de la herramienta servidor, para posteriormente instalar la parte cliente en los equipos que vayan a hacer uso de Oracle.

Algunas de las principales características son las siguientes:

- Soporte de transacciones
- Estabilidad
- Escalabilidad
- Soporte multiplataforma

	Creador	Licencia	Windows	Linux	Mac OS X
Microsoft SQL Server	Microsoft	Propietario	Si	No	No
MySQL	MySQL AB	GPL/propietario	Si	Si	Si
Oracle	Oracle Corporation	propietario	Si	Si	Si

Tabla 3.2: Comparativa entre distintos SGBD

3.4 Gestores de Contenidos

Los sistemas de gestión de contenidos (*Content Management Systems o CMS*) se usan principalmente para la administración y manipulación de sitios web de manera eficaz. Los CMS se utilizan tanto en internet como en una intranet, por eso son también conocidos como gestores de contenido web (*Web Content Management o WCM*). Se debe tener en cuenta que los CMS no se utilizan solo para las webs.

Los CMS incluyen herramientas para poder editar el contenido de la web sin tener conocimientos avanzados en informática. Suele ser común insertar una especie de procesador de textos *WYSIWYG What you see is what you get* (donde lo que se escribe es lo que se ve). De esta forma el autor no tiene que preocuparse de los aspectos técnicos y puede centrarse mejor en el contenido. Los CMS suelen aportar también algunas herramientas para definir el aspecto visual, el formato, sistemas para incluir nuevos módulos, etc.

Todos los datos y documentos creados en la web se almacenan en una base de datos. La estructura de la web presenta una visión jerárquica del sitio y permite modificaciones. Mediante esta estructura se puede asignar un grupo a cada área, con responsables, editores, autores y usuarios con diferentes permisos. Se crea un circuito de edición que va desde el autor hasta el responsable final de la publicación. El CMS permite la comunicación entre los miembros del grupo y hace un seguimiento del estado de cada paso del ciclo de trabajo.

Cuando la web está lista para ser publicada se aplica un patrón común que define el estilo de todo el sitio. La separación entre contenido y forma facilita las futuras modificaciones de estilo del sitio sin afectar para nada al contenido. También liberan al autor de preocuparse por el diseño final. Los CMS se adaptan a todo tipo de plataformas, sistemas operativos y al idioma del visitante. También se encarga de gestionar los menús de navegación y los módulos externos e internos del sistema.

Existen diferentes tipos de gestores de contenidos. Estos pueden ser algunos ejemplos:

- **Blogs:** Pensados para páginas personales.
- **Foros:** Pensados para compartir opiniones.
- **Wikis:** Pensados para el desarrollo colaborativo.
- **Enseñanza:** Plataformas para el aprendizaje on-line.
- **Comercio electrónico:** Plataforma de gestión de usuarios, catálogo, compras y pagos.
- **Publicaciones digitales:** Software de publicación digital como periódicos y revistas.

Las ventajas de los CMS pueden resumirse en:

- Gestión dinámica de usuarios y permisos
- Se reducen los costes de gestión de la información.
- Permite la publicación de contenidos a usuarios inexpertos.
- El mantenimiento del portal es mucho más sencillo, ya que todo el contenido de este está alojado en una base de datos.

3.4.1 Usuarios

Pueden existir diferentes tipos de niveles de acceso para los usuarios dependiendo del tipo de plataforma que se utilice. También se pueden crear usuarios con permisos para hacer casi cualquier cosa, como los creadores del portal y otros que apenas podrán interactuar con la web, como usuarios sin registrar. Se podrán crear tantos perfiles de usuarios como sea necesario para el tipo de plataforma utilizada.

Otra alternativa sería la de aplicar la lógica contraria. En vez de asignar permisos al usuario para acceder a las secciones, se podría asignar usuarios o grupos de usuarios a las secciones. Estas secciones tendrían cierto nivel de permisos. El nombre usado para este tipo de técnicas es ACL (*Access Control List, listas de control de acceso*).

Se entiende por usuario a cualquier persona que accede al sitio web. Los usuarios se pueden agrupar en roles en función de las operaciones que se les permite realizar. Es decir, que cada rol se caracteriza por disponer de una serie concreta de permisos. Se entiende por permiso la posibilidad de ejecutar determinadas operaciones o no sobre los diferentes elementos del CMS. Se pueden crear los roles que se crean necesarios para adecuarse al tipo de CMS utilizado.

Los usuarios más comúnmente utilizados en los CMS son los siguientes:

- **Administrador:** El administrador se encarga de gestionar el sitio web mostrando a los internautas los contenidos que él desea.
- **Editor:** El editor se encarga de la creación de contenidos para el CMS (noticias, imágenes, artículos, etc).
- **Usuario autenticado:** Este tipo de usuario podrán publicar comentarios sobre noticias, mandar mensajes a otros usuarios registrados, editar su perfil, etc.
- **Visitante:** Estos usuarios solo podrán visualizar generalmente pocas secciones, incluso a veces ninguna.

3.4.2 Roles y Permisos

Un punto importante a considerar en los CMS es el de la seguridad. Es necesario evitar que, por accidente o de forma intencionada, los usuarios puedan tener acceso a ciertos módulos u operaciones que pertenecen exclusivamente a otros tipos de usuarios.

A continuación se muestran los permisos que puede tener los distintos tipos de usuarios explicados anteriormente.

- **Administrador:** El administrador puede crear, bloquear o eliminar usuarios. Añadir y publicar nuevas secciones, etc.
- **Editor:** El editor tendrá acceso total a los contenidos del CMS para leerlos y modificarlos.
- **Usuario autenticado:** Este tipo de usuario tendrá privilegios para realizar ciertas operaciones que no le estarán permitidas a los usuarios sin registrar. Permisos de lectura sobre los contenidos del CMS, permisos de escritura para comentar ciertos contenidos pero no para editar los mismos, interactuar con otros usuarios registrados, etc.
- **Visitante:** Estos usuarios solo tendrán permisos de lectura para alguna sección o para ninguna. Normalmente no tienen permisos de escritura en ninguna sección. Este tipo de usuario necesita registrarse en el sistema para tener más permisos de lectura y escritura en las secciones al igual que los tiene el usuario autenticado.

3.4.3 Flujos de trabajo. Estados.

Los documentos que forma el CMS pueden pasar por una serie de estados dinámicos, los cuales son modificados por los usuarios que interactúan con la web. Se llama *flujo de trabajo* a la secuencia de acciones que los usuarios realizan sobre los documentos. Estas acciones hacen que los documentos pasen de unos *estados* a otros dinámicamente. Esto permite una visión más clara del proceso global del documento. También impide el acceso incontrolado de usuarios a los documentos cuando no se encuentran disponibles para ese tipo de usuario concreto.

Los diferentes estados y flujos de trabajo pueden variar según el tipo de CMS utilizado. Aunque estos son algunos de los más comúnmente utilizados:

- **Estado del documento previo a su revisión:** En este estado el documento ha sido creado, aunque todavía está pendiente de ser revisado.
- **Documento revisado:** El documento ha sido revisado y permanecerá en este estado hasta que se decida si está aprobado y listo para ser publicado.

- **Documento aprobado y listo para publicar:** En este estado se ha aprobado el documento y se verifica que está listo para ser publicado.
- **Documento publicado:** En este momento el documento está publicado y es accesible por todos los usuarios con los permisos adecuados.
- **Documento expirado:** Una vez haya terminado el tiempo de publicación, el documento pasa a no estar disponible para los usuarios y se cambia su estado a expirado.

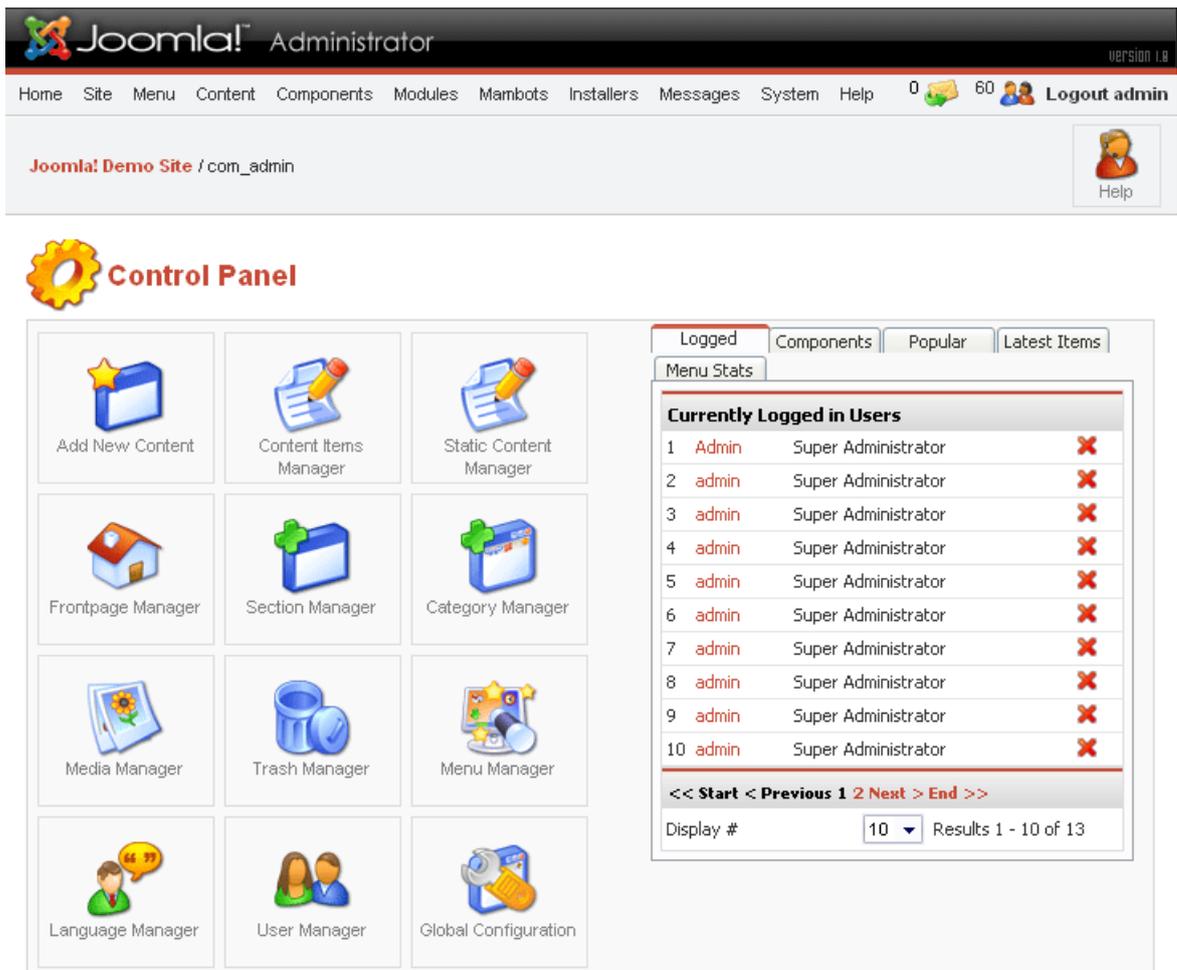
3.4.4 Ejemplos de CMS.

3.4.4.1 Joomla

Es un CMS que sirve para gestionar todo el contenido de un sitio web ofreciendo muy diversas funcionalidades. Ofrece muy diversas y prácticas características [6]:

- **Organización del sitio web:** Joomla está preparado para organizar eficientemente los contenidos de su sitio en secciones y categorías, lo que facilita la navegabilidad para los usuarios y permite crear una estructura sólida, ordenada y sencilla para los administradores. Desde el panel administrador de Joomla usted podrá crear, editar y borrar las secciones y categorías de su sitio de la manera en que más le convenga.
- **Publicación de Contenidos:** Con Joomla CMS podrá crear páginas ilimitadas y editarlas desde un sencillo editor que permite formatear los textos con los estilos e imágenes deseados. Los contenidos son totalmente editables y modificables.
- **Escalabilidad e implementación de nuevas funcionalidades:** Joomla ofrece la posibilidad de instalar, desinstalar y administrar componentes y módulos, que agregarán servicios de valor a los visitantes de su sitio web, por ejemplo: galerías de imágenes, foros, newsletters, clasificados, etc.
- **Administración de usuarios:** Joomla le permite almacenar datos de usuarios registrados y también la posibilidad de enviar E-mails masivos a todos los usuarios. La administración de usuarios es jerárquica, y los distintos grupos de usuarios poseen diferentes niveles de facultades/permisos dentro de la gestión y administración del sitio.
- **Diseño y aspecto estético del sitio:** Es posible cambiar todo el aspecto del sitio web tan solo con un par de clics, gracias al sistema de templates que utiliza Joomla.
- **Navegación y menú:** Totalmente editables desde el panel administrador de Joomla.
- **Administrador de Imágenes:** Joomla posee una utilidad para subir imágenes al servidor y usarlas en todo el sitio.

- **Disposición de módulos modificable:** En un sitio creado con Joomla, la posición de módulos puede acomodarse como se prefiera.
- **Encuestas:** Joomla posee un sistema de votaciones y encuestas dinámicas con resultados en barras porcentuales.
- **Feed de Noticias:** Joomla trae incorporado un sistema de sindicación de noticias por RSS/XMS de generación automática.
- **Publicidad:** Es posible hacer publicidad en el sitio usando el Administrador de Banners.
- **Estadísticas de visitas:** Información de navegador, OS, y detalles de los documentos (páginas) más vistos.



The screenshot displays the Joomla! Administrator interface. At the top, there is a navigation menu with options: Home, Site, Menu, Content, Components, Modules, Mambots, Installers, Messages, System, Help. The user is logged in as 'admin' and is viewing the 'com_admin' component. The main area is titled 'Control Panel' and contains a grid of icons for various administrative tasks: Add New Content, Content Items Manager, Static Content Manager, Frontpage Manager, Section Manager, Category Manager, Media Manager, Trash Manager, Menu Manager, Language Manager, User Manager, and Global Configuration.

On the right side, there is a 'Logged' tab and a 'Menu Stats' section. The 'Currently Logged in Users' table shows the following data:

ID	Username	Role	Action
1	Admin	Super Administrator	✗
2	admin	Super Administrator	✗
3	admin	Super Administrator	✗
4	admin	Super Administrator	✗
5	admin	Super Administrator	✗
6	admin	Super Administrator	✗
7	admin	Super Administrator	✗
8	admin	Super Administrator	✗
9	admin	Super Administrator	✗
10	admin	Super Administrator	✗

Below the table, there are navigation controls: '<< Start < Previous 1 2 Next > End >>' and a 'Display #' dropdown set to '10' with 'Results 1 - 10 of 13'.

Joomla! is Free Software released under the GNU/GPL License.
 Joomla! 1.0.2 Stable [Sunset] 02-Oct-2005 16:00 UTC
[Check for latest Version](#)

Figura 3.8: Panel de administración de Joomla

3.4.4.2 Drupal

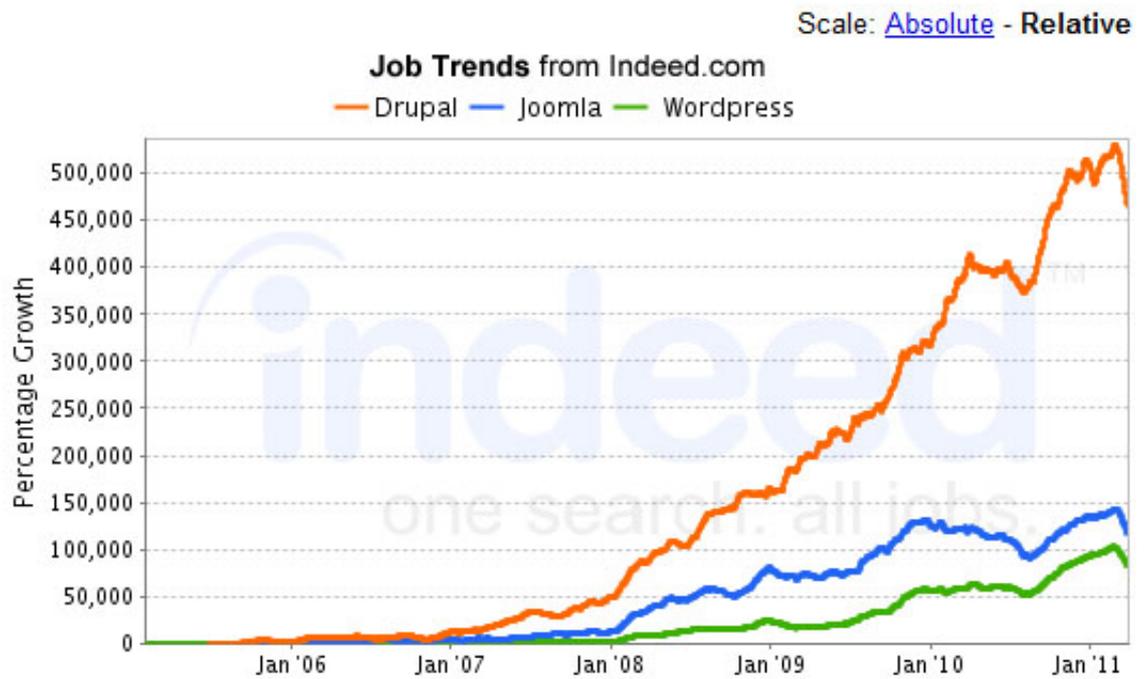
Drupal es un CMS basado en un sistema de módulos configurables con múltiples propósitos y que permite la publicación de contenidos como imágenes, artículos, foros, administrar usuarios, permisos, etc. Es construido, usado y soportado por una activa y diversa comunidad de usuarios en todo el mundo.

Con Drupal se pueden construir sitios muy diversos, desde blogs personales hasta aplicaciones empresariales. Cuenta con múltiples módulos para poder realizar el diseño que se quiera imaginar.

Este CMS cuenta con algunas características como estas [3]:

- **Ayuda on-line:** Un robusto sistema de ayuda online y páginas de ayuda para los módulos del *núcleo*, tanto para usuarios como para administradores.
- **Búsqueda:** Todo el contenido en Drupal es totalmente indexado en tiempo real y se puede consultar en cualquier momento.
- **Código abierto:** El código fuente de Drupal está libremente disponible bajo los términos de la licencia GNU/GPL. Al contrario que otros sistemas de *blogs* o de gestión de contenido propietarios, es posible extender o adaptar Drupal según las necesidades.
- **Módulos:** La comunidad de Drupal ha contribuido muchos módulos que proporcionan funcionalidades como 'página de categorías', autenticación mediante jabber, mensajes privados, bookmarks, etc.
- **Personalización:** Un robusto entorno de personalización está implementado en el núcleo de Drupal. Tanto el contenido como la presentación pueden ser individualizados de acuerdo las preferencias definidas por el usuario.
- **URLs amigables:** Drupal usa el `mod_rewrite` de Apache para crear URLs que son manejables por los usuarios y los motores de búsqueda.

Drupal, Joomla, Wordpress Job Trends



Indeed.com searches millions of jobs from thousands of job sites.
This job trends graph shows relative growth for jobs we find matching your search terms.

Figura 3.9: *Gráfica comparativa entre varios CMS*

3.5 Firmas digitales

Los riesgos en Internet en cuanto a seguridad son cada día mayores. Debido a esto se necesita un mecanismo que pueda dar autenticidad a la información que circula por internet. Sobre todo cuando se trata de transacciones comerciales, contratos de cualquier tipo, etc. Por esto motivo nace la *firma digital*.

En los últimos años ha crecido el número de usuarios que utilizan internet para un amplio abanico de transacciones distintas. Sin embargo, a pesar de que es un medio rápido y cómodo, actualmente no es el más seguro. Es necesario utilizar mecanismos y técnicas que permitan hacer cualquier operación a través de la red de manera segura y sabiendo con certeza con quien se está haciendo.

La firma digital certificada es un mecanismo electrónico, equivalente a un pasaporte o un documento de identidad en Internet, que permite certificar la autoría e integridad de los documentos que viajan por la red. Este medio es imprescindible cuando se utilizan sistemas de información que necesitan realizar operaciones de la misma forma que si los miembros implicados estuvieran presentes físicamente.

Algunos ejemplos de trámites que se podrían realizar de forma no presencial gracias a la firma digital serían los siguientes:

- Consulta e inscripción en el padrón municipal.
- Presentación y liquidación de impuestos.
- Consulta de multas de circulación.
- Consulta de asignación de colegios electorales.
- Firma electrónica de documentos oficiales y expedición de copias compulsadas.
- Consulta y trámites para solicitud de subvenciones.

3.5.1 Criptografía

Desde hace mucho tiempo para el hombre ha sido necesario conocer la identidad de los participantes en un proceso de comunicación de modo que no pueda existir ningún fraude ni se altere el contenido del mensaje. Algunos métodos utilizados son los siguientes:

- **DNI:** El documento nacional de identidad es el documento oficial exigido en España a los ciudadanos para poder acreditar su identidad antes cualquier administración, empresas, entidades u otros ciudadanos.

- **Pasaporte:** Es un documento público, personal e intransferible que acredita, dentro y fuera del Estado del que se es nacional, la identidad y la nacionalidad de su titular. En España este documento se solicitan en las Jefaturas Superiores y Comisarías de Policía; en el extranjero habrán de solicitarse en el Consulado (o en su defecto, Embajada) del Estado del que se sea nacional.
- **Firma:** Es el trazo, dibujo o título que una persona escribe con su propia mano en un documento, con la intención de dar su conformidad con lo expresado o para garantizar su autenticidad.
- **Notario que de fe de lo ocurrido:** Funcionario público autorizado para dar fe de los contratos, testamentos y otros actos extrajudiciales.
- **Correo Certificado :** Este tipo de servicio se caracteriza por que el correo queda registrado desde el momento de ser depositado en el sistema postal hasta su recepción por parte del destinatario. Durante todo el trayecto de la correspondencia, la agencia de correos efectúa un seguimiento del mismo, permitiendo al remitente que verifique la recepción por parte del destinatario.

El avance imparable de las tecnologías esta haciendo que cada vez más transacciones típicas del ser humano, como coger cita para el médico o trámites con la administración pública, se pueda hacer ahora a través de internet. Al no existir presencia física entre las partes implicadas, surge el gran problema de la seguridad. Para solventar tal problema se aplica la criptografía [17] ¹.

Para entender en que se basa la criptografía se va a exponer un ejemplo sencillo que muestre su funcionamiento.

Supongamos que se tienen dos personas, Miguel y Ángel. Miguel quiere enviar un mensaje a Ángel y no quiere que nadie más sepa interpretar su contenido. Lo que podría hacer Miguel sería introducir una clave criptográfica. La clave podría consistir simplemente en desplazar cada letra del mensaje original 5 posiciones a la derecha del abecedario, de forma que el mensaje *HOLA* quedaría transformado en *MTPF*.



Figura 3.10: Ejemplo básico de criptografía

¹La criptografía (escritura oculta) es la técnica, bien sea aplicada al arte o la ciencia, que altera las representaciones lingüísticas de un mensaje.

Es de suponer que las claves criptográficas empleadas en la actualidad son muchísimo más complicadas que las de este ejemplo. Existen dos técnicas que se basan en la criptografía para obtener el cifrado digital. Criptografía de *clave simétrica* y criptografía de *clave asimétrica*.

3.5.1.1 Clave Simétrica

Este sistema se basa en el ejemplo visto anteriormente. Para realizar la comunicación es necesario un mensaje y una clave [4]. El principal problema de este método es que la clave debe ser conocida no solo por el emisor sino también por el receptor. Esto repercute en problemas de seguridad, ya que no se asegura que algún otro sujeto haya podido dar con la clave.

La principal ventaja de este método es su velocidad, por eso su uso más destacado es en la transmisión de grandes volúmenes de datos.

3.5.1.2 Clave Asimétrica

En este sistema cada sujeto tiene un par de claves [4]. Una *clave pública* y una *clave privada*. La clave privada será guardada por el propietario y no deberá conocerla nadie que no sea él. La clave pública estará a disposición de todo el mundo.

El proceso de comunicación se realizaría de la siguiente forma. El sujeto Miguel quiere mandar un mensaje al sujeto Ángel. Miguel cifra el mensaje utilizando la clave pública de Ángel, que es accesible para todo el mundo. Se envía el mensaje, sin tener que enviar ninguna clave con él. Ángel al recibir el mensaje utilizaría su clave privada, que solo conoce él, para descifrar el mensaje.



Figura 3.11: Ejemplo de codificación de la información mediante el uso de una clave asimétrica

La ventaja de este método es que ya no es necesario enviar la clave con el mensaje por lo que se gana en seguridad. El inconveniente que se tendría ahora es la lentitud de este método. Para ganar eficiencia se utiliza un método combinado de clave asimétrica en el que se encripta la clave pública.

3.5.1.3 Clave Asimétrica y Cifrado de Clave Pública

El uso de claves asimétricas puede llegar a tener problemas de eficiencia, por lo que se utiliza este método utilizando también el encriptado de la clave pública [4]. El proceso de comunicación se produce de la siguiente forma. El sujeto Miguel quiere enviar un mensaje al sujeto Ángel. Miguel cifra el mensaje utilizando el método de clave simétrica, que se genera aleatoriamente. Se obtiene la *clave de sesión*. A continuación Miguel quiere enviar esta clave de sesión pero antes la encripta con la clave pública de Ángel utilizando el método de clave asimétrica. Ángel cuando reciba el mensaje utilizaría su clave privada para obtener la clave de sesión y esta ya podría descifrar el mensaje original.

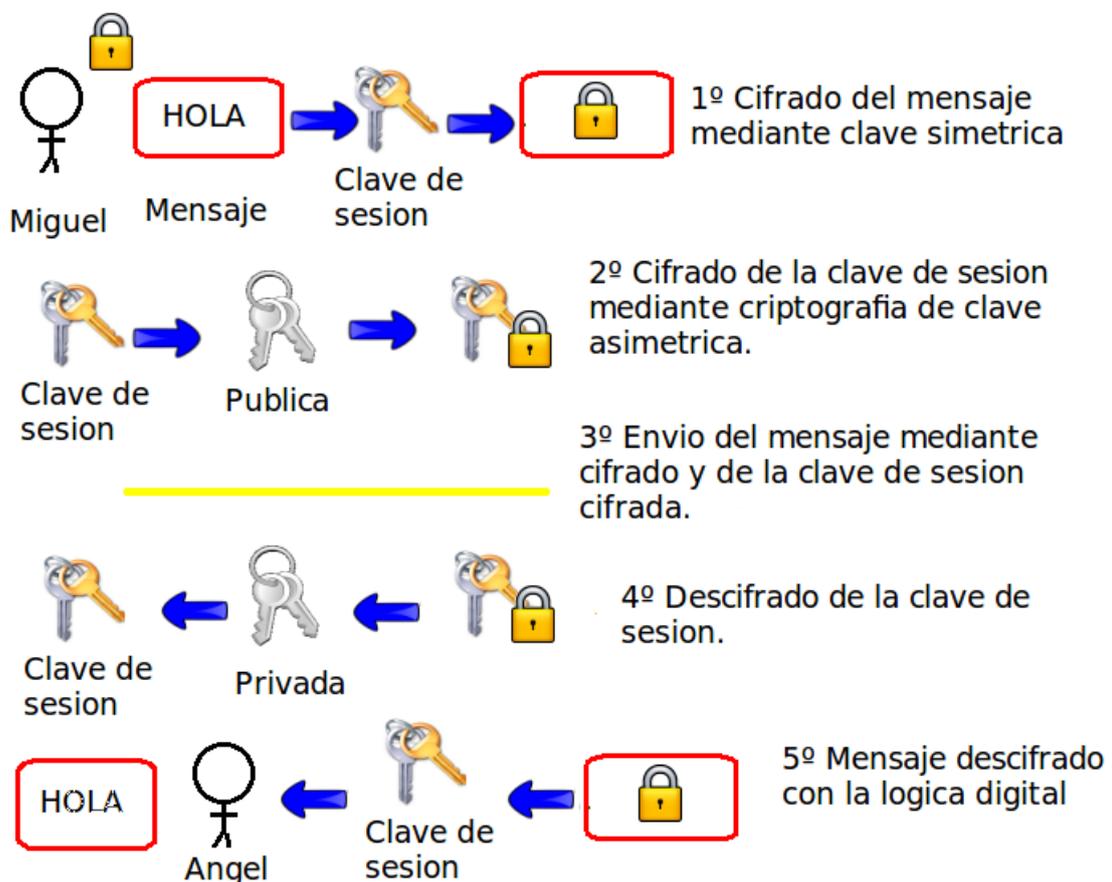


Figura 3.12: Ejemplo de codificación de la información mediante el uso de una clave asimétrica y cifrado de clave pública

Hasta ahora se ha conseguido un intercambio de información *confidencial e integral*. Es confidencial porque el mensaje solo lo puede leer su destinatario. La integridad se consigue porque se puede estar seguro que el contenido del mensaje no ha sido alterado ni modificado.

3.5.1.4 Firma Digital

La firma digital [25] permite que el receptor de un mensaje no pueda negar que el emisor es quien dice ser. Se consigue entonces otra propiedad añadida a la confidencialidad e integridad, el *no repudio*. La firma digital tiene la misma finalidad que una firma tradicional y es más difícil de falsificar. Solo se podría hacer si se conociera la clave privada del sujeto en cuestión.

El encriptado mediante firma digital se basa en lo explicado anteriormente, pero utilizando la clave privada en lugar de la pública para cifrar el mensaje. Otro de los inconvenientes que soluciona la firma digital es la lentitud que tenía el método de clave asimétrica. Ahora se utilizan las funciones hash ² para solventar este problema de eficiencia. Se asocia una función hash a un conjunto de los datos de tal forma que se obtiene otro conjunto de datos de tamaño fijo e independiente denominado resumen de los datos originales. El proceso de encriptado mediante firma digital sería de la siguiente forma.

El sujeto Miguel quiere enviar un mensaje al sujeto Ángel. Tanto Miguel como Ángel tienen en su poder sus respectivas claves pública y privada. Para que Ángel tenga total seguridad de que el mensaje fue enviado por Miguel, este deberá enviar el mensaje firmado. Miguel obtiene el resumen de los datos aplicando la función hash. Este resumen lo encripta utilizando su clave privada obteniendo la firma digital. Miguel envía a Ángel el resumen junto con la firma digital. Ángel utiliza ahora la clave pública de Miguel para descifrar el resumen del mensaje. Aplica la función hash para obtener el resumen. Si el resumen obtenido mediante la función hash es el mismo que el recibido, Ángel podrá estar seguro que el mensaje fue enviado por Miguel.

3.5.1.5 Certificado Digital

Como se ha explicado en los apartados anteriores, para realizar determinados procesos de encriptado, cada sujeto posee una clave privada y clave pública. La clave privada solo debe ser conocida por el propietario y se utilizan diversos métodos para intentar que no pueda ser obtenida por agentes externos, como tarjetas PCMCIA o tarjetas inteligentes que impiden su duplicado. Por otra parte la clave pública se puede dar a conocer a otros usuarios sin problemas. Pero para estar seguros de que una clave pública pertenece a un usuario concreto se utilizan los *certificados digitales*. Un certificado digital [20] es un documento electrónico que asocia la clave pública con la identidad de su propietario. Suele contener algunos datos como fechas de inicio y validez del certificado, ámbito de utilización, etc.

Ahora el problema que podría existir sería no tener la seguridad de que un certificado

²Es un algoritmo matemático que da como resultado Y sobre un resultado inicial X

digital sea válido o no. La única forma de asegurar que el certificado digital es válido sería que una tercera parte ajena a la comunicación entre los dos sujetos que pudiera dar fe de ello. Sería como en la forma tradicional de hacer un contrato, en el que se utiliza a un notario que da fe y testimonio del contrato que se está firmando por ambas partes. Esta tercera parte tiene que gozar de la confianza de las dos partes implicadas en la comunicación. Esta tercera parte que goza de plena confianza es la encargada de emitir los certificados digitales de los usuarios y se conoce con el nombre de *Autoridad de Certificación*.

En España existen varias autoridades certificadoras. La primera fue la FNMT (*Fábrica Nacional de Moneda y Timbre*) y luego se han ido sumando otras muchas, como ACE (*Agencia de Certificación Electrónica*), que está formada fundamentalmente por la banca, y FESTE (*Fundación para el estudio de la Seguridad de las Telecomunicaciones*), que está constituida por notarios, registradores, etc. Todas ellas emplean unos medios de identificación reconocidos jurídicamente y muy seguros.

	Eficiente	Seguro	Confidencial	Integridad	No repudio	Autenticación
Clave Simétrica	X					
Clave Asimétrica		X				
Cifrado Clave Pública	X	X	X	X		
Firma Digital	X	X	X	X	X	
Certificado Digital	X	X	X	X	X	X

Tabla 3.3: Métodos de Cifrado

3.5.1.6 Algoritmos de Cifrado

MD5

MD5 (*Message-Digest Algorithm 5, algoritmo de resumen del mensaje 5*) es un algoritmo de reducción criptográfico de 128 bits ampliamente usado en criptografía. Fue diseñado por el profesor Ronald Rives del MIT (*Massachusetts Institute of Technology, Instituto Tecnológico de Massachusetts*).

En 1996 se detectó una colisión de Hash de la función de comprensión del MD5. Posteriormente en 2004, otros investigadores chinos, encontraron también colisiones. Esto que los criptográficos empezasen a recomendar el reemplazo de la codificación MD5 a otras como SHA-1. Actualmente el uso de MD5 es muy amplio y se desconoce cómo afectarán estos problemas a su uso y a su futuro.

La codificación del MD5 de 128 bits es representada típicamente como un número de 32 dígitos hexadecimal. El siguiente código ASCII será tratado con MD5 y veremos su correspondiente hash de salida:

```
MD5("proyecto fin de carrera") = e9697882d3731198aac422090ae45cde
```

MD5 se utiliza ampliamente en internet para comprobar que un archivo descargado no ha sido alterado por el camino por algún intermediario malintencionado. Se compara una suma de comprobación con la suma MD5, y si el resultado es el mismo el usuario puede tener la certeza de que el archivo es igual al que comenzó a descargar. Esta suma también comprueba las descargas corruptas e incompletas.

Para comprobar la integridad de los archivos descargados de la red se pueden utilizar algunas herramientas. En los sistemas UNIX existe el comando *md5sum*. En PHP también está implementado esta función, la cuál suele utilizarse para codificar contraseñas de acceso entre otros campos secretos.

Para comprobar la integridad de un archivo descargado de Internet se puede utilizar una herramienta MD5 para comparar la suma MD5 de dicho archivo con un archivo MD5SUM con el resumen MD5 del primer archivo. En los sistemas UNIX, el comando de *md5sum* es un ejemplo de tal herramienta. Además, también está implementado en el lenguaje de scripting PHP como *MD5()* entre otros.

El MD5 también se puede usar para comprobar que los correos electrónicos no han sido alterados usando claves públicas y privadas.

SHA-1

La familia SHA (*Secure Hash Algorithm, Algoritmo de Hash Seguro*) es un sistema de funciones hash criptográficas relacionadas de la Agencia de Seguridad Nacional de los Estados Unidos y publicadas por el NIST (*National Institute of Standards and Technology*).

Existen varios tipos de SHA han ido surgiendo desde su comienzo en 1993:

- **SHA:** También conocido como SHA-0 fue el primer algoritmo Hash de la familia SHA.
- **SHA-1:** La resistencia del algoritmo SHA-1 se ha visto comprometida a lo largo del año 2005. Un grupo de investigadores chinos fue capaz de romper el SHA-1 en al menos 269 operaciones, unas 2000 veces más rápido que un ataque de fuerza bruta.
- **SHA-2:** También conocidos como SHA-224, SHA-256, SHA-384, y SHA-512. Se basan en un diseño algo modificado y rangos de salida incrementados.

Este algoritmo es usado entre otras aplicaciones en los sistemas distribuidos de control de versiones como *Git* y *Mercurial*. También se utiliza ampliamente para el encriptado de contraseñas en diversas aplicaciones. Como curiosidad fue utilizado en la consola *Nintendo Wii* para la verificación de firmas durante el arranque. Aunque más tarde fue descubierto un error que permitía al atacante eludir la seguridad. Este algoritmo también se utiliza en aplicaciones de seguridad de uso común y protocolos como TLS y SSL, PGP, SSH, etc.

Cabe destacar que en el desarrollo de este PFC se ha utilizado el algoritmo SHA-1 para el encriptado de las contraseñas de todos los usuarios que utilizan la herramienta.

3.6 Máquinas de Estados

Una máquina de estados es un sistema que modela una lógica de comportamiento y tiene señales de entradas y salidas. Estas salidas no dependen solo del comportamiento de las señales de entrada, sino también de señales anteriores. El comportamiento de la máquina en un estado determinado dependerá única y exclusivamente del estado y las entradas actuales.

Las máquinas de estados pueden ser:

- **Máquinas de estados infinitos:** Si el conjunto de estados de la máquina es infinito. Máquina universal de Turing[11]³ la cuál puede verse teóricamente con memoria infinita.
- **Máquinas de estados finitos o FSM:** Si el conjunto de estados de la máquina es finito. Este tipo es el utilizado para modelar la máquina de estados de un computador actual.

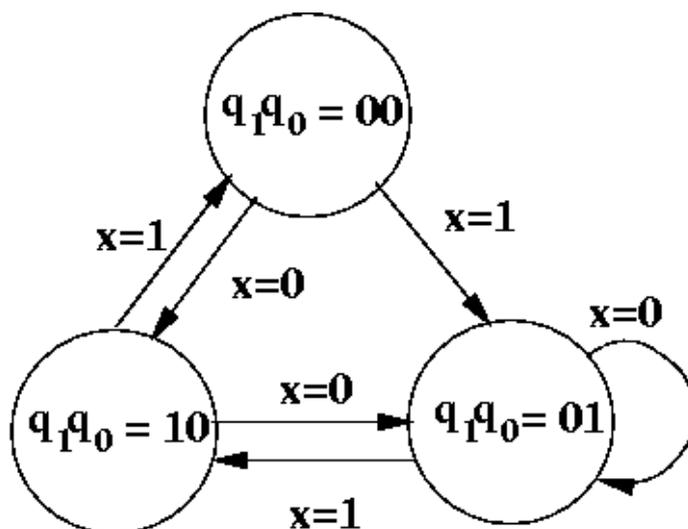


Figura 3.13: Máquina de estados finita

Las maquinas de estado finito se usan de manera muy amplia. Muchas veces pasan desapercibidas pero están presente en multitud de situaciones en la vida cotidiana.

- Cajeros automáticos
- Semáforos
- Ascensores
- Máquinas expendedoras

³Modelo computacional que realiza una lectura/escritura de manera automática sobre una entrada llamada cinta, generando una salida en esta misma.

- Wii Remote ⁴



Figura 3.14: Máquina de estados para un ascensor

Las máquinas de estados se utilizan en el modelado de sistemas dinámicos y en los sistemas de gestión de contenidos. Los contenidos de estos sistemas son elementos modificables por todos los roles que participan en el mismo. Estos roles realizan ciertas acciones que producen transacciones entre los diferentes estados de los contenidos y determinan de forma dinámica el comportamiento y la forma de operar del sistema con respecto a estos.

Por este motivo se utilizan máquinas de estado. Se busca que el sistema sea lo más flexible posible y que determine los diferentes flujos de trabajo. La máquina suele estar formada por los siguientes elementos:

- **Roles de Usuario:** Cada uno de los agentes que pueden modificar con su comportamiento los contenidos.
- **Acciones:** Actos que realizan los agentes por medio de los cuales modifican los contenidos.
- **Permisos:** Derechos de acceso que poseen los diferentes roles para poder actuar de diferente forma con los contenidos.
- **Estados:** Situación en la que se encuentran los contenidos dentro de la máquina de estados. El estado de un contenido dentro del sistema determinará todo el funcionamiento del mismo, las acciones que pueden realizar los distintos roles, los permisos de que disponen, etc.

⁴Es el mando principal de la consola Wii de Nintendo. Sus características más destacables son la capacidad de detección de movimiento en el espacio y la habilidad de apuntar hacia objetos en la pantalla.

3.7 Aprendizaje Automático y Supervisado

Hace unos años la idea de que un ordenador pudiera aprender podía parecer algo absurda. La capacidad de aprendizaje de los seres humanos es grandiosa. Desde que un niño aprende a decir sus primeras palabras hasta algo tan básico como reconocer un paisaje en una foto de un lugar que se visitó años atrás.

Se entiende como aprendizaje al conjunto de conductas que se pueden observar en la naturaleza y pueden ser observadas, procesadas y clasificadas. Es importante no solo responder, el qué y el cómo sino también identificar ese algo que evite obtener información sesgada y que no permita una comprensión total del proceso de aprendizaje.

El aprendizaje automático[24] se encarga de que ciertas entidades, que suelen ser robots o máquinas, tengan un comportamiento similar al del ser humano. Suele existir un agente racional encargado de buscar la mejor solución y en caso de conflicto intentar seleccionar la más adecuada.

Actualmente existen mecanismos, técnicas y reglas mediante las cuáles un computador es capaz de realizar labores de aprendizaje y a desarrollar algunas tareas similares a las de los seres humanos. Algunas de estas tareas podrían ser:

- Tomar decisiones.
- Escanear imágenes o vídeo.
- Entender ordenes dadas en lenguaje natural.

El aprendizaje automático, es un área de la inteligencia artificial, encargada del desarrollo de entidades, generalmente robots o máquinas, que tengan un comportamiento similar al del género humano o animal.

El aprendizaje se podría definir con la siguiente formula:

$$\text{Aprendizaje} = \text{Selección} + \text{Adaptación}$$

Según Herbert Simon⁵ el aprendizaje es:

⁵Economista, politólogo y teórico de las ciencias sociales estadounidense. En 1978 le fue concedido el Premio Nobel de Economía por ser uno de los investigadores más importantes en el terreno interdisciplinario y porque su trabajo ha contribuido a racionalizar el proceso de toma de decisiones.

Cualquier cambio en un sistema que le permite desempeñarse mejor la próxima vez, sobre la misma tarea u otra tomada de la misma población.

El aprendizaje automático se centra en desarrollar algoritmos y técnicas que permiten a las máquinas *aprender* a partir de unos datos dados. Se extraen patrones, reglas o características relevantes que ayudan a cumplir ciertos objetivos. Pueden ser usados por personas o también llamado *Minería de Datos* aplicada en Inteligencia de negocio. También puede ser usado por sistemas de predicción automática, una vez que el sistema ha aprendido puede calcular valores a partir de ciertos datos, tomar decisiones basadas en reglas, asesorar en la toma de decisiones, etc.

Se pueden distinguir dos tipos de aprendizaje:

- **Supervisado:** Se trata de aprender de ejemplos de una función o conjunto de entrenamiento, es decir, a partir de un conjunto de pares entrada-salida, con el objetivo de predecir la salida en problemas futuros.
- **No supervisado:** Trata de averiguar cómo se estructuran los datos en sí. Al contrario que el aprendizaje supervisado no existe una función con una salida, sino únicamente datos de entrada a estudiar.

Uno de los objetivos principales del aprendizaje automático es producir programas capaces de mejorar su rendimiento a través de la experiencia:

- Mejorar al realizar una tarea T
- Respecto a una medida de rendimiento P
- Gracias a la utilización de la experiencia E

Se construyen sistemas capaces de adquirir el conocimiento necesario para realizar tareas, usando la experiencia acumulada.

A continuación se muestran algunas de las aplicaciones típicas donde se utiliza el aprendizaje supervisado.

- Tareas que los humanos hacemos pero no podemos describir como las hacemos. Ejemplo: Reconocimiento de caracteres.
- Problemas donde la función objetivo cambia frecuentemente. Ejemplo: Predicciones bursátiles.

- Situaciones donde cada usuario tiene su propia función objetivo. Ejemplo: Adaptación de interfaces, spam, gestión de correo electrónico.
- Problemas donde el volumen de datos hace imposible a los humanos realizar cualquier análisis. Ejemplo: Técnicas de Data-Mining capaces de encontrar relaciones dentro de grandes volúmenes de datos.

Estas son las etapas que se suelen seguir en el diseño de sistemas de aprendizaje automático:

- Elegir la experiencia que usaremos para entrenar el sistema de aprendizaje.
- Definir el tipo de función objetivo que pretendemos aprender.
- Determinar la representación que utilizaremos para la función objetivo.
- Seleccionar el algoritmo utilizar y aplicarlo para aprender la función objetivo.
- Evaluar el sistema.

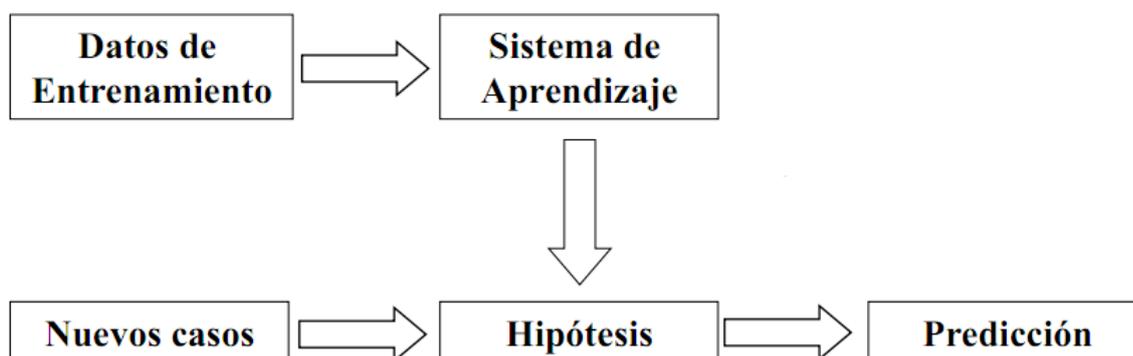


Figura 3.15: Esquema del proceso de aprendizaje automático

3.7.1 Aprendizaje Basado en Casos

El aprendizaje basado en casos[28] es una estrategia constituida por un conjunto de experiencias, situaciones o problemas de la vida real, que tienen la finalidad de reflexionar sobre el curso de acción elegido y proponer acciones alternativas ante tales situaciones. Es el proceso de solucionar nuevos problemas basándose en las soluciones de problemas anteriores.

Para la resolución de problemas es necesaria cierta experiencia que se va adquiriendo poco a poco y ayuda a crear un modelo inteligente de comportamiento. Si se toman como ejemplo dos sujetos, uno novato y otro experto, se podrá observar la diferencia entre ellos a la hora de resolver un problema cualquiera.

Por un lado el usuario experto recogerá los datos más útiles y relevantes. A partir de estos podrá generar hipótesis muy verosímiles. Por otro lado el usuario novato necesitará tomar muchos datos, la mayoría de los cuales no serán nada relevantes, y a partir de estos generará algunas soluciones poco probables.

Cabe destacar que en algún momento los usuario que ahora son expertos algún día fueron usuarios novatos. Por lo que debe existir algún método por el cuál llegaron a adquirir esa experiencia.

El aprendizaje basado en casos ayuda a resolver problemas a partir de experiencias pasadas o casos, adaptando soluciones antiguas para resolver problemas nuevos, o recuperando casos anteriores para iluminar aspectos de la situación actual.

El razonamiento basado en casos es el que más se aproxima al pensamiento del ser humano. Véase el ejemplo de un niño pequeño que toca una olla de agua hirviendo y se quema. Para la próxima vez ya habrá aprendido que no debe tocar la olla si no quiere quemarse. Habrá aprendido de su propia experiencia. Aunque también puede tener la experiencia de ver a su madre cogiendo la olla con un trapo para no quemarse. El niño volverá a aprender de otra experiencia y sabrá como tiene que tocar la olla si no quiere quemarse.

3.7.2 Resolución de Conflictos

Para llevar a cabo la resolución de conflictos es necesario utilizar una serie de pasos bien definidos. Una vez identificado el problema se pasará a actuar de la siguiente forma:

- **Identificar el problema:** Descripción del problema y características que lo manifiestan.
- **Analizar la causa:** Averiguar cuál fue el origen del problema para poder tratarlo en un futuro.
- **Buscar soluciones alternativas:** Analizar las razones que producen el problema.
- **Medidas de evaluación y resultados:** Tomar algún objetivo del cual se puedan utilizar métodos contables para determinar el flujo y mejorarlo.

3.8 Agentes Inteligentes

Se puede definir al *agente inteligente* [29] como una entidad software que, basándose en su propio conocimiento, realiza un conjunto de operaciones destinadas a satisfacer las necesidades de un usuario o de otro programa, bien por iniciativa propia o porque alguno de éstos se lo requiere. [22]

Se debe tener en cuenta que todos los agentes inteligentes son programas, pero no todos los programas son agentes inteligentes. Un agente inteligente se diferencia de un programa común y puede tener varias de estas características:

- **Autonomía:** Actuar sin ningún tipo de intervención humana directa y tener control sobre sus propios actos.
- **Sociabilidad:** Comunicarse por medio de un lenguaje común con otros agentes, e incluso con los humanos.
- **Reactividad:** Percibir su entorno y reaccionar para adaptarse a él.
- **Racionalidad:** El agente siempre realiza *lo correcto* a partir de los datos que recibe del entorno.
- **Pro-actividad:** El agente intenta cumplir sus propios planes u objetivos.
- **Adaptabilidad:** Relacionado con el aprendizaje que un agente es capaz de realizar y si puede cambiar su comportamiento basándose en ese aprendizaje.
- **Movilidad:** Capacidad de un agente de trasladarse a través de una red telemática.

No es necesario que un agente inteligente cumpla todas las propiedades citadas anteriormente, aunque sería deseable que al menos cumpliera con autonomía, sociabilidad y capacidad de reacción.

Los agentes inteligentes se basan en la percepción de acciones y en la actuación en consecuencia según sea el tipo de aplicación.

El conocimiento que posee un agente inteligente lo va adquiriendo de otros agentes, del propio usuario que lo utiliza y del trabajo que desempeña. Existen distintos tipos de agentes de acuerdo a la función que desempeñan.

- **Agentes Cooperativos:** Se trata de agentes autosuficientes y conocedores del medio en el que se encuentran, pero que son capaces de interactuar con otros agentes para alcanzar su objetivo.



Figura 3.16: Agente inteligente de búsqueda

- **Agentes Móviles:** Estos agentes son capaces de suspender un proceso en ejecución, transportarse a través de la red y reanudar la ejecución del proceso que estaban realizando. Se evita transportar grandes cantidades de información a través de la red, ya que el agente puede seleccionar cierta información antes de enviarla.
- **Interfaces Inteligentes de usuarios:** Estos agentes se utilizan para hacer búsquedas expandidas de información. Devuelven a parte de la información pedida expresamente por el usuario, información relacionada con la materia que podría resultarle de interés.
- **Agentes de Búsqueda Inteligentes:** Estos agentes son capaces de realizar búsquedas de información de manera muy descentralizada, recuperarla y almacenar las preguntas en un resultado para un usuario en concreto.
- **Agentes de Consulta:** Estos tipos de agentes generan más subagentes dividiendo la consulta en subconsultas hasta que la demanda de información quede resuelta.
- **Agentes Recomendadores:** Estos agentes básicamente ayudan a la gente a encontrar lo que necesitan en la web. Ayudan a los usuarios haciendo recomendaciones sobre que opciones tomar atendiendo al comportamiento de los mismos. Este tipo de agente se explicará mas detenidamente en el siguiente apartado.

3.8.1 Agentes Recomendadores

En general, un agente recomendador se refiere a cualquier sistema que hace recomendaciones teniendo en cuenta la características del usuario. Básicamente se dedican a captar

información de los usuarios acerca de preferencias de sitios web, productos, personas, etc para luego procesarla y almacenarla. Después se puede usar con diferentes fines como aconsejar a futuros usuarios sobre ciertos temas de interés en los que los criterios de partida coincidan con usuarios que hicieron búsquedas similares. Estos agentes utilizan la experiencia acumulada que han ido obteniendo de almacenar múltiples casos para asesorar correctamente a futuros usuarios. La experiencia de estos agentes va creciendo con el tiempo.

Algunos ejemplos de sistemas recomendadores son los siguientes:

- **Clipo:** Era el sistema de ayuda que se incluía en el paquete ofimático Microsoft Office hasta su versión Office 2007. En esta última fue sustituido por un nuevo sistema de ayuda. Tenía forma de clip animado. Esta herramienta presentaba funciones de búsqueda de ayuda y ofrecía consejos al usuario sobre como redactar correctamente el documento.
- **Asistente de OpenOffice:** Este paquete ofimático también cuenta con un asistente de ayuda. Cuando la herramienta detecta que puede sugerir algún tipo de ayuda aparece el asistente. Al hacer clic en él se muestra un menú con la ayuda disponible. La forma de este asistente es de una bombilla con un flotador salvavidas.
- **Ono:** Dispone de un agente recomendador [8] que ayuda al cliente a elegir entre combinados de televisión, teléfono e internet.
- **Vodafone:** Tiene un agente recomendador [12] para ayudar al cliente a elegir la tarifa que más se adapta a sus necesidades.



Figura 3.17: *Asistente de Microsoft Office*

La apariencia de estos agentes siempre ofrece una interfaz amigable con el usuario y fácil de usar. Esto ayuda a que el usuario se familiarice pronto con el asistente y tenga en cuenta sus opiniones. En Microsoft Office existían otros asistentes aparte del mencionado Clipo.

- **Ridondo:** Una pelota roja con cara sonriente.
- **F1:** Un robot.
- **Logotipo de Office:**
- **Natura:** Una versión animada de la Tierra.
- **Rocky:** Un perro.
- **Mínimo:** Un gato animado.
- **Robby:** Una nave espacial.
- **Super can:** Un perro superhéroe.

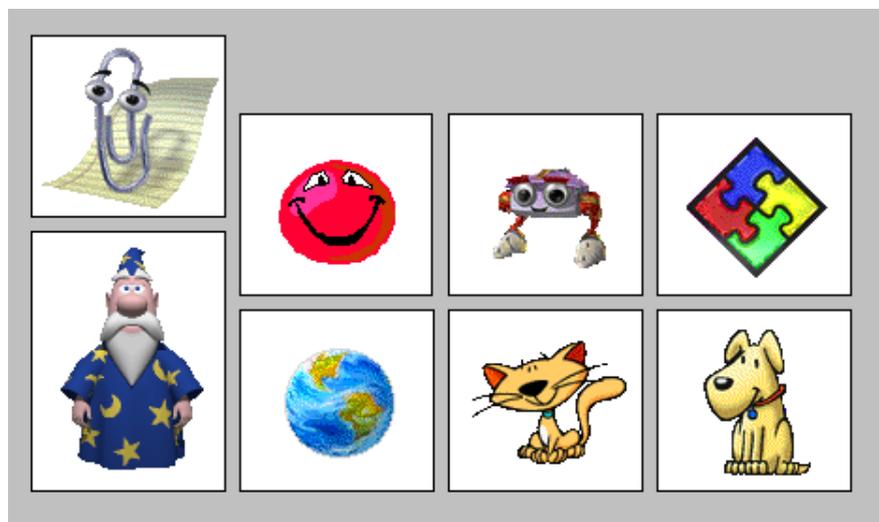


Figura 3.18: *Asistentes de Microsoft Office*

Generalmente la presencia y el uso de agentes inteligentes en sistemas de información para la gestión de contenidos ha sido bastante amplia. Esto se hace con el fin de mejorar la asistencia al usuario mediante la ayuda que se le proporciona. Los agentes propuestos se desempeñan como asistentes personales que ayudan al usuario a llevar a cabo las actividades que desea realizar.

En este aspecto, los agentes inteligentes pueden asistir al usuario en la búsqueda de información útil y relevante, a la vez que recuperar, analizar, manipular e integrar informaciones y datos de carácter heterogéneo.

En base a la información explicada anteriormente, es factible la creación y uso de agentes inteligentes en gestores de contenidos, ya que son capaces de suministrar información confiable, actualizada y sola aquella que tenga que ver sobre el tema que se trata.

Gestión Automática de los Intercambios entre Estudiantes Sócrates/Erasmus

En este capítulo se describe de forma detallada todo el trabajo desarrollado en el PFC. Después del trabajo de análisis realizado en el capítulo anterior, se ofrecen una serie de soluciones a la problemática planteada para la consecución de los objetivos del capítulo 2.

4.1 Proceso de Gestión en Becas Erasmus

En este apartado se van a comentar todos los pasos que debe seguir un estudiante con beca Erasmus desde el principio hasta el final. De esta manera se entenderá mejor la necesidad de aplicar TICs para facilitar todo el proceso a los roles implicados.

El primer paso que debe seguir un alumno que acaba de obtener la beca Erasmus es preparar el *contrato de estudios*. Se trata de un documento en el que constarán datos personales del alumno y coordinadores implicados, datos relativos a la universidad de Castilla-La Mancha, datos relativos a la universidad de destino, información sobre el periodo de estancia en el extranjero, etc. Aunque la parte más importante de este contrato de estudios es en la que aparecen las asignaturas que el alumno elige cursar como estudiante Erasmus. Aparecerá en una parte del documento todas las asignaturas de la UCLM que el alumno decide coger y a continuación las asignaturas de la universidad destino por las que las quiere convalidar.

Este documento será rellenado por el alumno junto con la ayuda y el asesoramiento del *profesor de movilidad*. Entre los dos realizarán un contrato de estudios totalmente válido que será después entregado al *coordinador de centro* para que lo revise y de su posterior aprobación. Finalmente una vez aprobado el contrato de estudios por el profesor de movilidad y el coordinador de centro el documento deberá ser entregado al *coordinador institucional*. Este deberá revisar el contrato ver que se están cumpliendo todos los requisitos que manda la

normativa y firmar el documento. Cuando el coordinador institucional firma el contrato de estudios ya está listo y preparado para poder ser entregado en la universidad de destino.

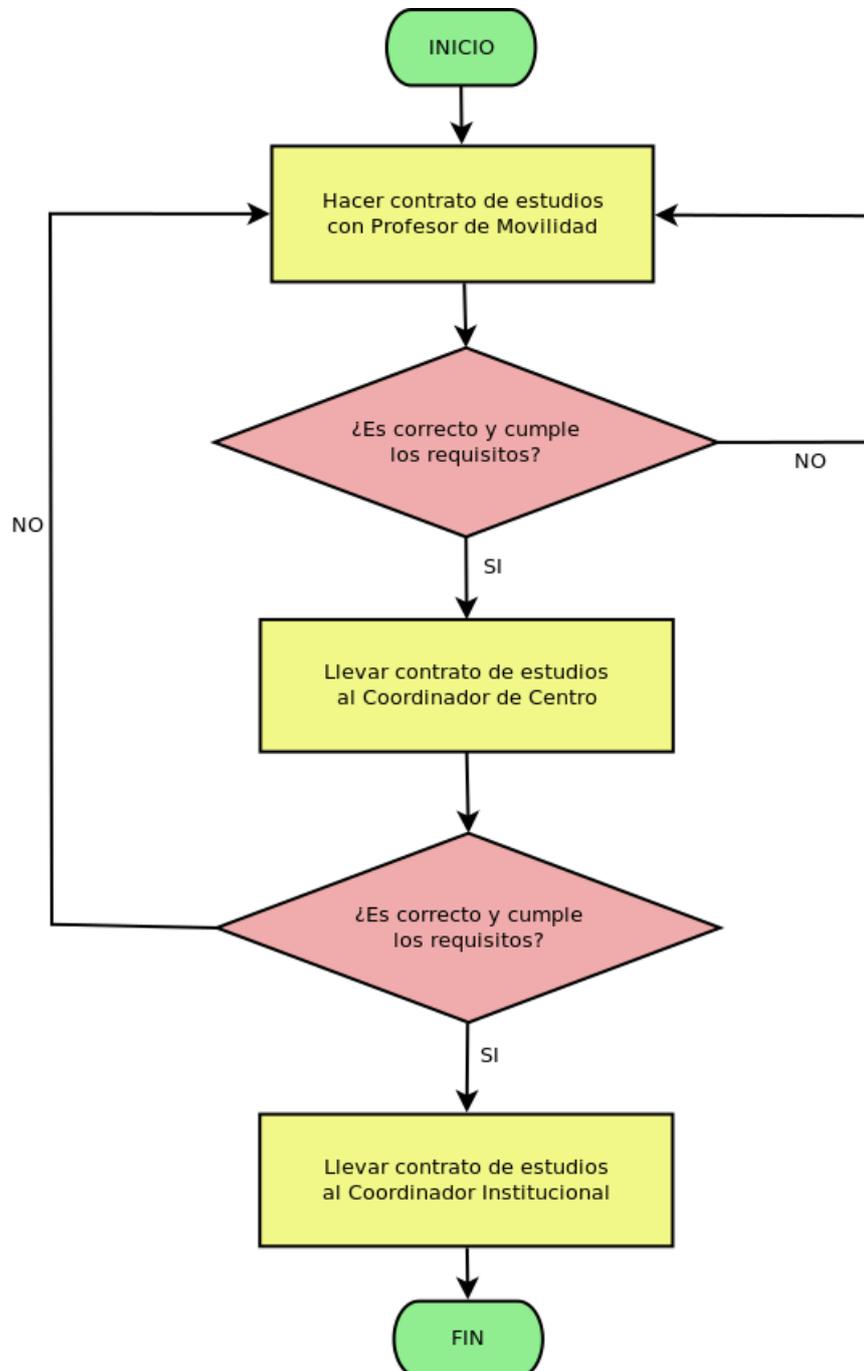


Figura 4.1: *Proceso a seguir para cursar una beca Erasmus*

Una vez comenzada la estancia del alumno Erasmus en el país extranjero pueden surgir algunos inconvenientes que hagan que necesite modificar el contrato de estudios. Acompaña también al contrato de estudios un formulario para realizar dichas modificaciones. El alumno deberá realizar los cambios que considere oportunos y entregarlo de nuevo al profesor de movilidad para que revise que los cambios realizados son correctos. Al igual que se hizo

anteriormente cuando se preparo el contrato de estudios por primera vez, este deberá entregar el contrato de estudios al coordinador de centro y este a su vez al coordinador institucional. Cuando todos ellos hayan firmado las modificaciones del contrato de estudios el contrato original quedará modificado de acuerdo a los cambios realizados.

Una vez finalizada la estancia Erasmus el estudiante deberá regresar con un documento donde se reflejen las calificaciones obtenidas en las asignaturas cursadas en la universidad de destino. Este documento será entregado al coordinador de centro para ser revisado. El coordinador de centro deberá realizar la conversión de notas de acuerdo al sistema de calificación español. El coordinador de centro utilizará una tabla de conversión de calificaciones europea. En esta tabla aparecen los sistemas de calificaciones de notas de los países en los que existe Erasmus. Las calificaciones definitivas del alumno serán reflejadas en un documento llamado *ficha de reconocimiento de estudios*.

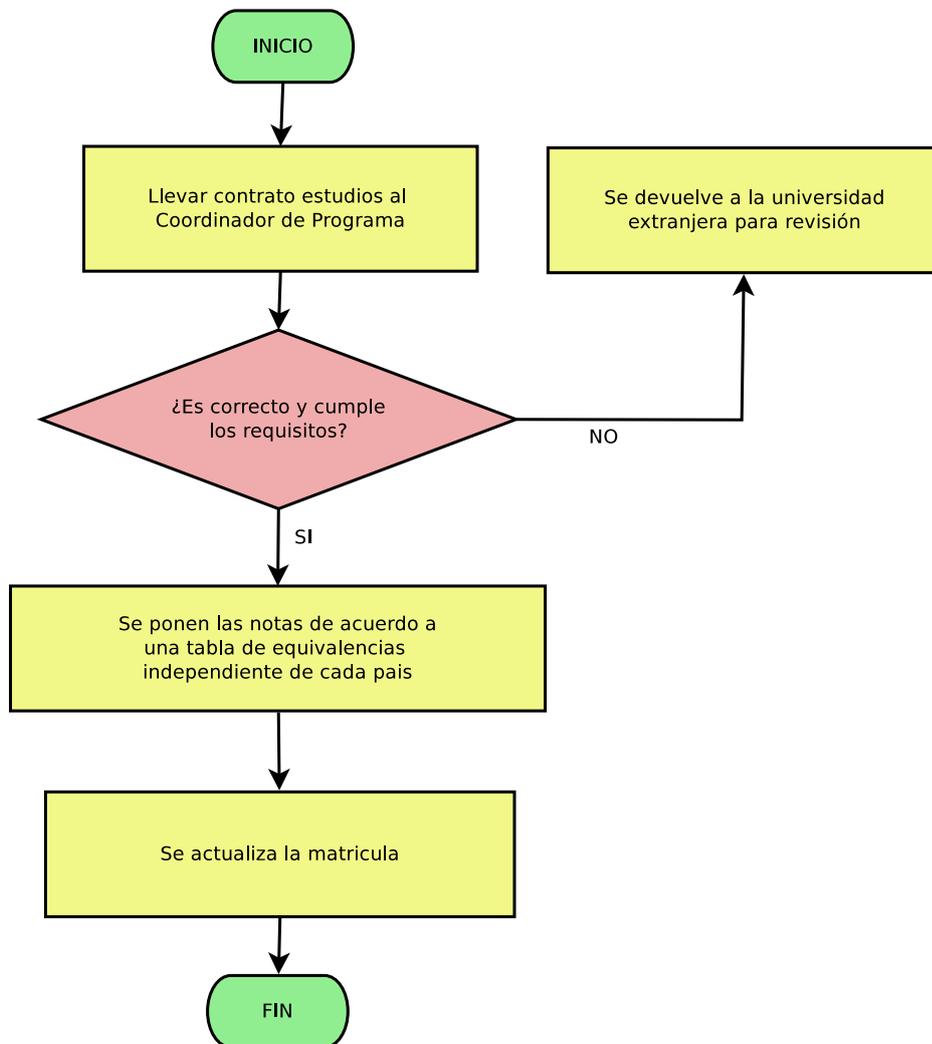


Figura 4.2: *Proceso final de una estancia Erasmus*

4.2 Problemática actual

El camino que debe seguir un alumno cuando se le ha concedido una beca Erasmus es realmente tedioso, implica un largo proceso y actualmente no existen herramientas que faciliten su gestión. El sistema propuesto en este PFC permitirá a todos los organismos implicados, alumnos, coordinadores de programa, coordinadores de centro y coordinador institucional, realizar todas las gestiones de manera electrónica.

En primer lugar cuando un alumno recibe la beca Erasmus debe preparar su contrato de estudios. Para ello debe acudir a la ORI (*Oficina de Relaciones Internacionales*) a coger todos los formularios que debe rellenar. Seguidamente tiene que mirar que profesor de movilidad tiene asignado y organizar una reunión con él para preparar el contrato de estudios. El contrato de estudios es un documento en el cual se acuerdan que asignaturas de la Universidad de Castilla-La Mancha va a cursar el alumno y por que asignaturas de la universidad destino van a ser convalidadas. La realización de este documento no es nada trivial ya que coger el plan de estudios de la universidad destino e intentar buscar asignaturas similares en créditos y temática a las que desea coger el alumno. Aparte de esto hay que tener en cuenta el periodo en el que se imparte la asignatura, ya que puede darse el caso que la asignatura se curse en el primer cuatrimestre y el alumno tenga una beca Erasmus para el segundo cuatrimestre.

Una vez se ha terminado el contrato de estudios, paso que suele requerir de varias reuniones entre el profesor de movilidad y el estudiante, hay que organizar una nueva reunión con el coordinador de centro. Este revisará el contrato de estudios y comprobará que está todo correcto y no se incumple ningún requisito especificado en la normativa (Véase Anexo B).

Algunos de estos requisitos son los siguientes:

- No se admitirá la inclusión en el contrato de estudios de asignaturas que el alumno curse en cuarta o posterior convocatoria dentro del plan de estudios de la E.S.I.
- El contrato de estudios debe ser equilibrado. Para estancias de un semestre la carga lectiva estará comprendida entre 15 ECTS y 33 ECTS y para estancias anuales la carga lectiva estará comprendida entre 30 ECTS y 66 ECTS.
- El Proyecto Fin de Carrera (PFC) se puede realizar durante una estancia Erasmus si el profesor responsable del programa lo estima oportuno.

Una vez aprobado el contrato de estudios por parte del coordinador de centro, el alumno deberá entregarlo de nuevo en la ORI para que sea firmado por el coordinador institucional. Una vez firmado la ORI avisará al estudiante para que recoja su contrato de estudios

terminado y válido para ser entregado en el país de destino.

Después de que el estudiante llega al país destino y se dispone a elaborar su horario se da cuenta de que en la mayoría de las ocasiones tiene que modificar su contrato de estudios debido a sucesos imprevistos. La asignatura se imparte en el idioma nativo del país, la asignatura se suspende por falta de alumnos, el horario es incompatible con otra asignatura matriculada, etc.

Para realizar las modificaciones del contrato de estudios hay que realizar otra serie de pasos. Primeramente informar por email al profesor de movilidad en España y que de su visto bueno a todos los cambios que el alumno desea realizar. Seguidamente, el estudiante debe enviar por fax su contrato de estudios con las modificaciones pertinentes para que el profesor de movilidad lo firme. Este después tendría que pasar el contrato modificado al coordinador de centro para que firmase las modificaciones y este a su vez al coordinador institucional. Después de este tedioso proceso se enviaría un fax a la universidad destino con el contrato de estudios modificado final.

Una vez finalizada la estancia del alumno Erasmus obtendrá un acta de la universidad de destino con las calificaciones obtenidas en las asignaturas cursadas allí. El alumno al regresar a España deberá entregar ese acta al coordinador de centro para que sea revisada y ponga las calificaciones adecuadas en el expediente del alumno. Este paso no es automático, ya que hay que realizar una conversión de notas antes de poder rellenar el expediente. En todos los países que participan en el convenio Erasmus no se utiliza el mismo sistema de calificación. Existe una tabla de conversión europea donde el coordinador de centro tiene que mirar e ir calculando la equivalencia de todas las notas según el país destino del estudiante.

Hay que tener en cuenta la problemática de todos los pasos expuestos anteriormente. Ya que no siempre es fácil preparar reuniones presenciales con los coordinadores por falta de disponibilidad o tiempo. A parte de lo cansado que puede resultar tener que organizar tantas reuniones para revisar el contrato de estudios y llevarlo a tantos sitios distintos.

Cuando el alumno llega al país extranjero el proceso de modificaciones del contrato de estudios se complica ya que todas las comunicaciones hay que hacerlas mediante email y se tiene que mandar el contrato de estudios mediante fax. Puede ocurrir que algunos coordinadores no tengan una versión definitiva del contrato de estudios.

Con el desarrollo de esta herramienta se propone facilitar en gran medida todo lo explicado anteriormente. El alumno después de obtener la beca Erasmus solo tendría que identificarse en la herramienta para empezar a hacer el contrato de estudios. Tendrá que navegar a

través de una serie de páginas rellenando una serie de formularios. Puede ver información de todas las asignaturas que tiene disponibles. La herramienta asesorará al alumno con posibles conflictos que puedan existir con algunas asignaturas. Tampoco tendrá que preocuparse por no cumplir la normativa, ya que el programa en todo momento avisará de cualquier punto que se esté incumpliendo. Una vez finalizado el contrato de estudios, el alumno haciendo uso de su certificado digital podrá generar una copia en formato PDF y firmarla digitalmente.

Después de realizar el paso anterior, automáticamente el contrato de estudios será enviado al profesor de movilidad. Un sistema de gestión de eventos avisará al coordinador de que se han realizado cambios desde la última vez que utilizó la herramienta. Todos los roles que utilicen la herramienta serán avisados mediante el sistema de eventos de cualquier suceso que ocurra. Siempre aparecerá el tipo de evento, el creador del mismo y la fecha y hora en la que se produjo. El coordinador podrá revisar el contrato de estudios realizado por el alumno y si está de acuerdo firmarlo digitalmente al igual que hizo el alumno. También puede no estar de acuerdo y rechazar el contrato. Si ocurre esto, automáticamente el contrato será devuelto al alumno junto con el motivo por el cual fue rechazado.

De esta forma el contrato irá de un coordinador a otro automáticamente sin que haya que organizar ninguna reunión física. De tal modo que cuando el contrato sea firmado por el coordinador institucional, automáticamente todos los roles implicados tendrán una copia del contrato de estudios final válido y firmado digitalmente.

De igual forma, el alumno, al final de su estancia Erasmus, entregará el acta de calificaciones al coordinador de centro. Este solo tendrá que introducirlas en la herramienta, y se traducirán automáticamente al sistema de calificación español teniendo en cuenta el país destino del estudiante. Se podrá generar entonces la *ficha de reconocimiento de estudios*. Es un documento donde aparecen las asignaturas cursadas, tanto en la universidad origen como en la universidad destino, y las calificaciones de las mismas. Este documento irá acompañado de la firma del coordinador de centro para darle validez.

4.3 Propuesta de Soluciones a la Problemática Actual

En esta sección se van a resumir los puntos a desarrollar en el PFC. Estos ofrecen una solución a toda la problemática explicada anteriormente.

- **Tipos de usuarios y roles:** A lo largo del proceso de una beca Erasmus intervienen, a parte del alumno en cuestión, otras personas que hay que tener en cuenta. Estos usuarios tienen unos roles dependiendo de las tareas que tienen que realizar. Los roles utilizados son los mismos que en el proceso tradicional, solo que ahora no existe el problema de la distancia ni el tiempo entre ellos. Cada rol podrá comunicarse con otro desarrollando sus labores a través de la herramienta lo que facilitará enormemente la gestión de la beca Erasmus.

- **Máquina de estados:** En esta herramienta el contrato de estudios es un elemento principal para todos los roles. Todos ellos necesitan acceder a él a lo largo del proceso de una beca Erasmus. Está claro que no sería coherente que cada rol pudiera acceder al contrato y modificarlo siempre que quisiera. Si esto ocurriese podrían ocurrir muchas incoherencias.

Es necesario llevar un control de flujo del contrato para ver que roles y con que permisos pueden acceder al contrato dependiendo en el estado en el que este se encuentre. Para ello se ha pensado en utilizar una máquina de estados que modele el comportamiento que ha de seguir el contrato de estudios a lo largo de todo el proceso.

- **Sistema de control de eventos:** Hay cinco tipos de roles distintos diseñados para el acceso a esta herramienta. Cada uno de ellos tiene disponibles un menú de opciones distintos dependiendo de la función que tenga que realizar en cada momento. Se ha diseñado un sistema de eventos con el que los roles podrán ser conscientes de los cambios que se han producido desde la última vez que utilizaron el software.

Varios de los roles pueden tener multitud de alumnos Erasmus a su cargo. Llevar el control del estado en que se encuentra el contrato de estudios de cada alumno en cada momento puede ser una tarea complicada. Por eso el sistema de eventos avisará de cualquier cambio producido para no tener que estar revisando cada contrato de forma particular cada vez que se entre en la herramienta. El sistema generará un evento cada vez que se produzca un acontecimiento importante en el transcurso del contrato de estudios y algún otro hito a lo largo de la beca Erasmus del estudiante como puede ser la realización de la ficha de reconocimiento de estudios. En el evento producido siempre se mostrará el emisor del evento, el tipo de evento, la fecha y hora en que se produjo y un mensaje opcional que sería escrito por el emisor del evento especificando alguna anotación o aclaración.

- **Cálculo de notas y equivalencias:** Son muchos los países participantes en la beca Erasmus. Hay que tener en cuenta que el sistema de calificación de notas que se tiene en estos países no tiene porqué ser el mismo al sistema de calificación español. El coordinador de centro es el encargado de hacer estas conversiones. Pero no es nada trivial tener que estar transformando cada una de las notas que trae cada uno de los alumnos Erasmus. Por este motivo es necesario implementar de alguna forma alguna función que transforme las notas de sistema de calificación extranjero al español.

Existe una tabla de conversión de calificaciones europea donde se especifican las reglas de conversión a utilizar en cada país. Se introducirán estos datos en la herramienta de modo que el coordinador de centro pueda cambiarlos en un futuro si la normativa actual cambiase. Utilizando esta tabla, el software tendrá en cuenta el país extranjero donde el alumno cursó la beca Erasmus y dará como resultado las notas convertidas al sistema de calificación español.

- **Autenticación de documentos:** En este PFC se pretende ayudar tanto al alumno y a los distintos coordinadores a realizar el proceso Erasmus de una manera más fácil, controlada y ordenada aplicando TICs. Durante el proceso Erasmus se generan dos documentos importantes para el alumno. El contrato de estudios y la ficha de reconocimiento de estudios. Esta herramienta tiene que seguir generando estos documentos y que tengan la misma validez que si los documentos fueran generados en papel de la manera tradicional.

Estos documentos tienen que ser firmados por el estudiante y los coordinadores para que tengan validez oficial. La herramienta pondrá a disposición de los distintos roles un mecanismo de firmas digitales mediante certificados para firmar estos documentos.

- **Aprendizaje de conflictos y recomendaciones:** Lo que se pretende con esta función es ayudar al alumno en la creación del contrato de estudios. Es posible que el alumno en un principio elija asignaturas del país extranjero que después no le sea posible cursar. Esto puede deberse a diferentes motivos, ya sea porque la asignatura sea dada en el idioma oficial del país y este sea desconocido para el alumno, porque coincida el horario con alguna otra asignatura matriculada, etc.

La herramienta debería ir aprendiendo de la experiencia pasada de otros alumnos y aconsejar a los futuros teniendo en cuenta toda esa información. De esta manera cada vez se podrán hacer contratos de estudios más rápidos y libres de errores evitando en su mayor medida las posteriores modificaciones de matrícula.

Cuando un alumno esté realizando el contrato de estudios y elija una asignatura en la que algún alumno anterior haya tenido algún problema, un agente recomendador estará al tanto de la elección de alumno y aparecerá para hacerle las recomendaciones oportunas y para mostrarle los motivos por los cuáles no sería recomendable elegir esa asignatura.

4.4 Diseño Multicapa

Desde el comienzo del desarrollo de la herramienta se ha seguido un diseño multicapa. Este diseño divide y estructura el código en tres capas que son las siguientes:

- **Presentación:** La presentación trata de todos aquellos elementos que permiten la interacción entre el usuario final y la aplicación. Se refiere además a las tecnologías de lado del cliente como CSS ó JavaScript, los navegadores, los lenguajes de marcas y de transformación.
- **Lógica de dominio:** La lógica de dominio ó de negocio es quien dicta las reglas sobre como debe trabajar la aplicación en si. Constituye todo lo que tiene que ver con cálculos de entrada de datos, validaciones, procesos y como se presentaran los datos en la capa de presentación.
- **Persistencia:** La persistencia de datos trata sobre como la lógica de dominio requiere de los datos que le proporcionan servicios de bases de datos, sistemas de transacciones, mensajes, sistemas de archivos, etc.

Cada capa mencionada anteriormente ofrece unos servicios y tiene unas responsabilidades diferentes. La separación de las capas es fundamental para un desarrollo satisfactorio. Algunas de las ventajas de utilizar el diseño multicapa son las siguientes:

- Clara separación entre interfaz, lógica de negocio y de presentación, que además provoca parte de las ventajas siguientes.
- Sencillez para crear distintas representaciones de los mismos datos.
- Reutilización de los componentes.
- Simplicidad en el mantenimiento de los sistemas.
- Facilidad para desarrollar prototipos rápidos.
- Los desarrollos suelen ser más escalables.
- Cambios en una de las capas no implica cambios en el resto.

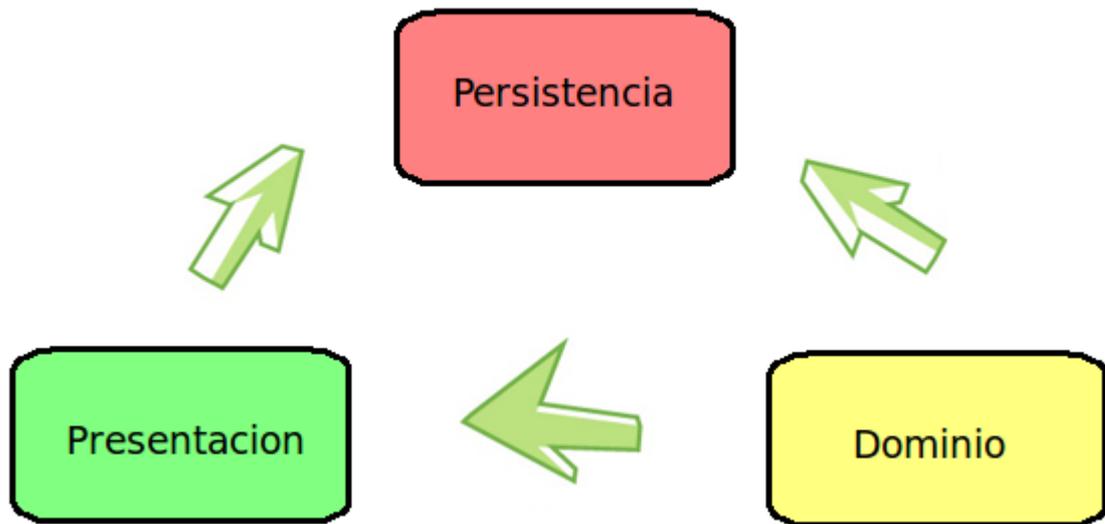


Figura 4.3: *Esquema del diseño multicapa*

La herramienta es accedida por distintos roles. Cada rol tiene distintas funciones y no está permitido que pueda acceder a los distintos módulos del sistema en los que no tiene permisos. El sistema se encuentra dividido en módulos teniendo en cuenta la función que se realiza en los mismos. El acceso a cada módulo está controlado teniendo en cuenta el rol que accede en cada momento. Si algún rol intenta acceder a algún módulo y no tiene permisos se le deniega el acceso. A continuación se explica con más detalle los distintos tipos de usuarios y roles que existen en el sistema.

4.5 Tipos de Usuarios y Roles

A lo largo de todo el proceso que implica una beca Erasmus intervienen una serie de personas que van a ser catalogadas en roles dependiendo del papel que desempeñen.

Los tipos de roles pueden ser:

- **Estudiante:** Dentro de este rol se encuadran todos los alumnos que han obtenido la beca Erasmus.
Este tipo de rol tendrá permisos para crear un contrato de estudios, hacer modificaciones sobre él y poder visualizar el contrato en cualquier momento.
- **Profesor de Movilidad:** En esta categoría está el contacto más directo con el estudiante. La función principal de este rol será la de elaborar un contrato de estudios adecuado para el estudiante. Un profesor de movilidad tendrá varios estudiantes Erasmus bajo su tutela.
El profesor de movilidad tendrá permisos para visualizar una lista de todos los alumnos Erasmus que tiene, ver sus contratos y editarlos siempre que el contrato se encuentre en el estado adecuado. Tendrá un módulo donde aparecerán todos los contratos Erasmus que tiene pendientes de revisar y firmar y otro con los que ya han sido revisados. Finalmente también podrá ver un listado con las calificaciones que han obtenido sus alumnos Erasmus.
- **Coordinador de Centro:** En este rol entran los responsables de las becas Erasmus en cada facultad. Una de las funciones de este rol será la de revisar que los contratos de estudios son adecuados y se adaptan a la normativa vigente (Véase Anexo B). También se encargará de revisar las calificaciones del alumno Erasmus en el extranjero y generar un acta oficial de notas convalidadas para su expediente en la UCLM.
Este tipo de rol tendrá permisos para ver y editar el contrato de estudios de sus alumnos Erasmus al igual que el profesor de movilidad. También tendrá un apartado donde aparecerán los contratos pendientes de revisar y los que ya han sido firmados. El coordinador de centro tiene permisos para realizar la conversión de notas de acuerdo al sistema de calificación español. La conversión se realizará teniendo en cuenta la tabla de conversión de calificaciones europea. Este coordinador también tiene permisos de modificar dicha tabla si en un futuro se modificase dicho plan de calificaciones.
- **Coordinador Institucional:** Existirá un coordinador institucional por cada campus. La función de este es la de dar el visto bueno para que el contrato de estudios quede listo para ser enviado al país destino.
El coordinador institucional podrá ver un listado de todos los alumnos Erasmus que

tiene y sus contratos. A parte de esto contará con un módulo donde aparecerán todos los contratos pendientes de ser firmados y otro con los que ya han sido revisados.

- **Administrador:** La función del administrador consistirá en añadir a la base de datos a los alumnos que hayan obtenido una beca Erasmus. A partir de este momento los demás roles podrán empezar a hacer las gestiones correspondientes. También se encargará de administrar los permisos de acceso de los distintos roles a todos los módulos que componen la herramienta.

A continuación se puede ver una tabla donde aparecen distintos módulos de la herramienta y si tienen permiso de acceso o no los roles del sistema.

	Prof. Movilidad	Coord. Centro	Coord. Institucional	Estudiante	Administrador
Realizar Contrato de Estudios	X	X	X	X	
Lista de Alumnos Erasmus	X				
Modificar Permisos					X
Creación de Usuarios					X
Ver y modificar contratos	X	X	X		
Firmar contratos de estudios (PM)	X				
Contratos firmados (PM)	X				
Visualizar contrato				X	
Firmar contratos de estudios (CC)		X			
Contratos firmados (CC)		X			
Firmar contratos de estudios (CI)			X		
Contratos firmados (CI)			X		
Modificar contrato de estudios				X	
Calificar a los alumnos		X			
Visualizar notas				X	
Ver notas	X	X			

Tabla 4.1: Módulos y permisos de acceso al sistema de los mismos

4.5.1 Estructura de la Base de Datos para la configuración de roles y permisos

En la imagen se puede apreciar como se ha realizado la configuración de los roles y permisos en la base de datos del sistema. Se ha diseñado tres tablas llamadas *roles*, *módulos* y *permisos*. La primera cuenta con dos columnas que se explican a continuación:

- **id_rol:** Es el identificador que diferencia de manera única a un rol del resto.
- **rol:** Es el nombre identificativo del rol.

id_rol	rol
A	Estudiante
B	Coordinador Centro
C	Profesor Movilidad
D	Coordinador Institucional
E	Administrador

Figura 4.4: Tabla roles de la BBDD

La segunda tabla cuenta con tres columnas:

- **id_modulo:** Es el identificador que diferencia de manera única a un módulo del resto.
- **modulo:** Es el nombre identificativo del módulo.
- **urlmodulo:** Contiene el enlace por el cuál se accede a dicho módulo.

id_modulo	modulo	urlModulo
1	Realizar Contrato de Estudios	../presentacion/present_paso1.jsp?numero=1
2	Lista de Alumnos Erasmus	../presentacion/present_listaAlumnos.jsp
3	Modificar Permisos	../presentacion/present_admin.jsp
4	Introducir Usuario	../presentacion/present_introducir_usuario.jsp
5	Ver y Modificar Contratos	../presentacion/present_contratos.jsp

Figura 4.5: Tabla módulos de la BBDD

La tercera tabla cuenta con cuatro columnas:

- **id_rol:** Es el identificador que diferencia de manera única a un rol del resto.
- **id_modulo:** Es el identificador que diferencia de manera única a un módulo del resto.
- **lectura:** Este campo puede tomar dos valores. Tomará 0 si el rol no tiene permisos de lectura y tomará 1 cuando tenga permisos de lectura.
- **escritura:** Este campo puede tomar dos valores. Tomará 0 si el rol no tiene permisos de escritura y tomará 1 cuando tenga permisos de escritura.

id_rol	id_modulo	lectura	escritura
A	1	1	1
C	1	1	1
B	1	1	1
D	1	1	1
E	1	0	0

Figura 4.6: Tabla permisos de la BBDD

Cada vez que un usuario intenta acceder a uno de los módulos del sistema se hacen una serie de comprobaciones:

- El tipo de rol al que pertenece el usuario.
- El tipo de módulo al que quiere acceder el usuario.
- Si ese tipo de rol tiene permisos para acceder a ese módulo en concreto.

Si las anteriores comprobaciones tuvieron éxito el usuario podrá acceder al módulo. De esta forma se evitan accesos indeseados a módulos restringidos para ciertos usuarios.

La creación de un nuevo rol debido a cambios en la normativa no supondría grandes modificaciones en el sistema. Solamente habría que crear el nuevo registro en la tabla *roles* de la base de datos indicando el *id_rol* y el *rol* correspondiente. La instrucción que habría que añadir en la base de datos sería la siguiente.

```
1 insert into roles values ('codigo','nombre_rol');
```

También puede ocurrir que sea necesario la actualización de los permisos de lectura/escritura de los roles existentes. El administrador del sistema puede cambiarlos fácilmente a través del formulario que tiene disponible en el sistema (Véase imagen A.40).

4.6 Máquina de Estados

En el desarrollo de esta herramienta hay un elemento principal que se usa en todo momento, desde que el alumno recibe su beca Erasmus, hasta que la finaliza. Este elemento es el *contrato de estudios* del estudiante. Este documento desde el momento en que se crea pasa a tener un estado. En función de las acciones que realicen los agentes este estado puede cambiar. Por este motivo es necesario definir una máquina de estados que añada dinamismo al sistema. La máquina de estados se modelará de tal manera que cualquier cambio en la normativa vigente no afecte al código de la herramienta. Solo habría que configurar la máquina de estados y seguiría funcionando correctamente. De esta forma también se evitarían accesos inadecuados al contrato de estudios, ya que dependiendo del estado en el que se encuentre estará disponible para ser visualizado o modificado solo para algunos roles.

4.6.1 Proceso de Elaboración del Contrato de Estudios

El contrato de estudios va evolucionando según va avanzando el estudiante en su beca Erasmus. Dividiendo todo el proceso que debe seguir un contrato de estudios Erasmus desde su creación hasta su finalización se pueden llegar a tener varias fases.

- **En creación:** Este primer paso se da cuando el alumno ha recibido la beca Erasmus y está preparando el contrato de estudios. En este momento solo el alumno tendrá permisos para ver el contrato y poder modificarlo. Se pasará al siguiente estado cuando el alumno haya terminado el contrato y lo pase al profesor de movilidad para su supervisión.
- **Supervisión del profesor de movilidad:** En este estado el profesor de movilidad revisará el contrato de estudios confeccionado por el estudiante. El contrato de estudios solo podrá ser visto por el estudiante y el profesor de movilidad, aunque solo podrá ser editado por este último. Ya que no sería coherente, que el alumno modificase el contrato cuando ya está siendo revisado. Si la revisión ha finalizado satisfactoriamente el contrato pasará al siguiente estado.
- **Supervisión del coordinador de centro:** En este estado el coordinador de centro revisará el contrato para ver que se está cumpliendo con la normativa Erasmus. El contrato podrá ser visualizado, además de por el coordinador de centro, por el estudiante y por el profesor de movilidad. Aunque solo podrá ser modificado por el coordinador de centro.
- **Supervisión del coordinador institucional:** En este paso el coordinador institucional revisará el contrato y lo firmará. A partir de este momento el contrato está terminado y es totalmente válido para ser entregado en la universidad destino. En este estado

todos los roles implicados pueden visualizar el contrato, aunque solo el coordinador institucional puede modificarlo.

- **Finalizado:** El contrato se encontrará en este estado cuando haya sido revisado y firmado por todos los roles participantes. Este contrato es el definitivo y el que deberá ser entregado en la universidad destino cuando el alumno comience su beca Erasmus.
- **En modificación:** A este estado se pueden llegar desde varias situaciones. Existe un periodo de modificaciones en el que el alumno puede modificar el contrato final para hacer algún cambio en las asignaturas que había elegido previamente. También puede ser que en el proceso de creación del contrato, alguno de los coordinadores o el profesor de movilidad no estuviera de acuerdo con el contrato que se estaba generando y rechazaran firmarlo. En este caso el contrato pasaría a estar en este estado. Cuando ocurre esto solo el alumno puede visualizar y modificar el contrato de estudios.

Para dejar más claro los pasos explicados anteriormente se muestra la siguiente tabla. En ella se puede apreciar los permisos de lectura y escritura que poseen los roles en cada estado.

A = Estudiante

C = Profesor de Movilidad

B = Coordinador de Centro

D = Coordinador Institucional

E = Administrador del Sistema

Estado	Permisos de Escritura	Permisos de Lectura
En creación	A	A
Supervisión del profesor de movilidad	C	A,C
Supervisión del coordinador de centro	B	A,B,C
Supervisión del coordinador institucional	D	A,B,C,D
Finalizado	-	A,B,C,D
En modificación	A	A

Tabla 4.2: Permisos de lectura y escritura para los distintos estados del contrato de estudios.

4.6.2 Diagrama de Estados

A continuación se muestra un diagrama de estados para identificar cada una de las rutas o caminos que puede tomar el flujo de información después de ejecutarse cada proceso.

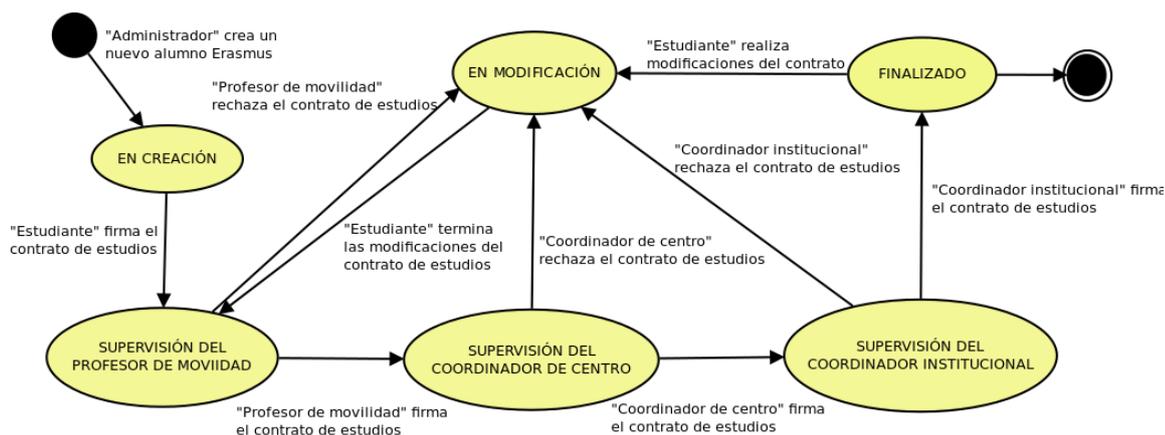


Figura 4.7: Diagrama de estados del contrato de estudios

Este diagrama de la figura 4.7 representa la situación actual de la máquina de estados. En un futuro puede cambiar la actual normativa Erasmus, pero esto no supondría un gran perjuicio. No habría que modificar el código fuente de la herramienta, sino que bastaría con volver a configurar la máquina de estados.

Un posible cambio en la normativa podría ser que se eliminase el rol *Profesor de movilidad*. Las tareas asignadas a este rol pasarían a ser responsabilidad del coordinador de centro. Simplemente habría que eliminar este estado de la máquina de estados y reconducir las acciones que apuntaban al rol *Profesor de movilidad* hacía el *Coordinador de centro*.

El tener una máquina de estados que gestione este proceso facilita mucho la labor de mantenimiento de la herramienta y la fácil actualización de esta en caso de cualquier cambio.

4.6.3 Estructura de la Base de Datos para el Almacenamiento de la Máquina de Estados

En la imagen se puede apreciar como se ha realizado la configuración de la máquina de estados en la base de datos del sistema. Se ha diseñado una tabla llamada *estados_contrato* con cinco columnas que se explican a continuación:

- **idAlumno:** Es el identificador que diferencia de manera única a un alumno Erasmus del resto.
- **idEstado:** Este campo posee dos posibles valores. (1 o 0) Obtendrá el valor de 1 cuando la máquina de estados se encuentre en ese estado. En cualquier otro caso este campo permanecerá a 0. La transición de un estado a otro se produce por algún evento ocurrido en el sistema. Tales transiciones se pueden ver en la imagen 4.7.
- **estado:** Es el nombre del estado.

- **lectura:** En este campo se reflejan los identificadores de los roles que tienen permisos de lectura en el estado actual.
- **escritura:** En este campo se reflejan los identificadores de los roles que tienen permisos de escritura en el estado actual.

idAlumno	idEstado	estado	escritura	lectura
1	1	En creación	A	A
1	0	En modificación	A	A
1	0	Supervisión del Profesor de Movilidad	C	A,C
1	0	Supervisión del Coordinador de Centro	B	A,C,B
1	0	Supervisión del Coordinador Institucional	D	A,C,B,D
1	0	Finalizado		A,C,B,D
2	1	En creación	A	A
2	0	En modificación	A	A
2	0	Supervisión del Profesor de Movilidad	C	A,C
2	0	Supervisión del Coordinador de Centro	B	A,C,B
2	0	Supervisión del Coordinador Institucional	D	A,C,B,D
2	0	Finalizado		A,C,B,D

Figura 4.8: Almacenamiento de la máquina de estados en la BBDD

4.6.4 Diagrama de clases relacionadas con el contrato

A continuación se muestra un diagrama de clases donde aparecen las clases que tienen alguna relación con el contrato de estudios. Se pueden apreciar tres clases.

- **Usuario:** Clase que se compone de todos los atributos del usuario. Dispone de métodos para firmar el contrato de estudios y para realizar modificaciones en él.
- **Contrato:** Esta clase contiene métodos para añadir al contrato la firma digital del usuario y las modificaciones que haya realizado en él.
- **Modificaciones:** Clase que permite anotar para cada usuario las modificaciones que se producen.

Se puede ver que un usuario puede realizar varias firmas y que cada una de estas firmas pertenecerá a un solo contrato. Cada contrato puede ser firmado cuatro veces, una por cada coordinador y la última por el propio estudiante.

También se puede apreciar que el usuario puede realizar ninguna o varias modificaciones en el contrato. Y que cada contrato puede tener o no modificaciones.

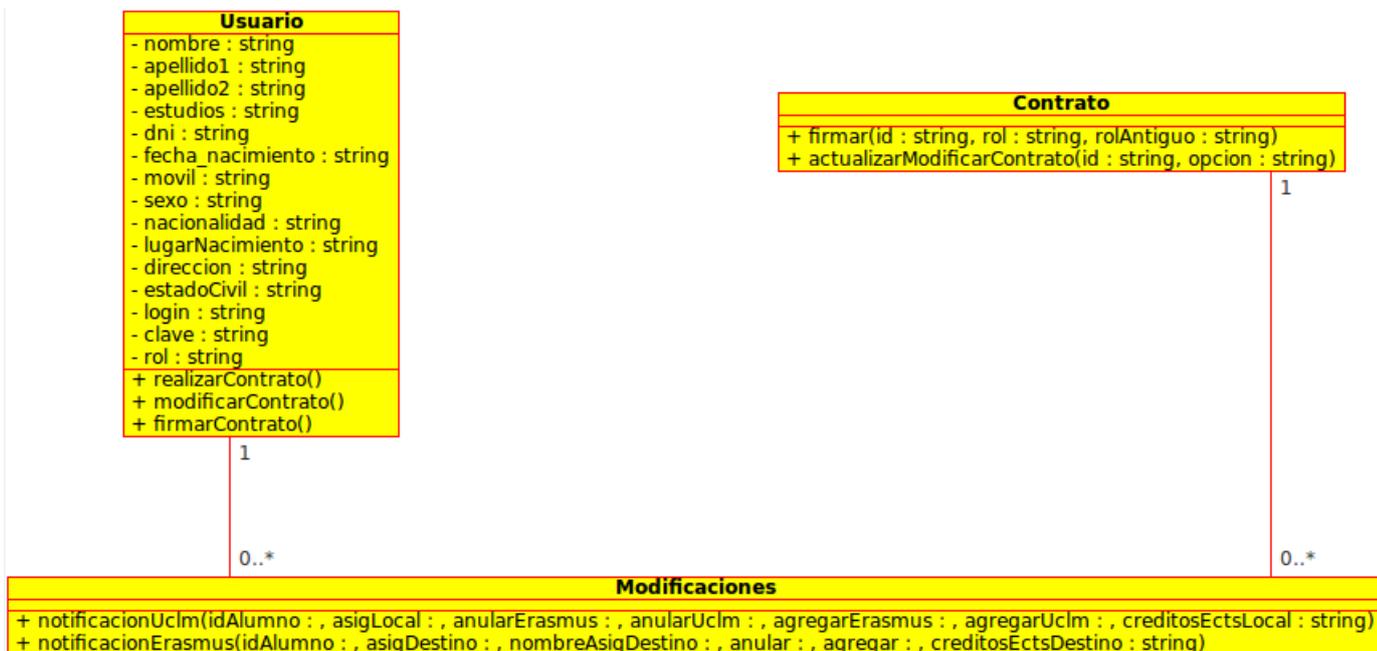


Figura 4.9: Diagrama de clases relacionadas con el contrato

1

¹Los tipos de los atributos no se incluyen con el objetivo de simplificar el diagrama.

4.7 Sistema de Control de Eventos

Debido a la naturaleza de la herramienta es necesario implementar un sistema de eventos para facilitar la labor de control y el seguimiento correcto de todos los alumnos Erasmus por parte de los coordinadores. También será muy útil para los estudiantes, que podrán ver en todo momento la evolución de su contrato de estudios cuando este va pasando por los distintos coordinadores. Se producirá un evento en el sistema siempre que se produzca un suceso importante. Se puede apreciar que estos sucesos coinciden con un cambio de estado en la máquina de estados 4.6. Por lo que se puede decir que se produce un evento en el sistema siempre que se haya producido un cambio de estado en el contrato de estudios.

La herramienta es manejada por varios roles distintos como son los estudiantes, profesores de movilidad, coordinadores de centro, coordinador institucional y administrador. Cada uno de estos roles cuando utilice la herramienta verá en su pantalla principal los módulos que tiene disponibles y puede utilizar. Al interactuar con ellos se producirán una serie de eventos que afectarán a otros usuarios del sistema.

Un ejemplo podría ser el de un estudiante que acaba de rellenar los formularios del contrato de estudios y lo firma. Automáticamente el contrato es enviado al profesor de movilidad que corresponda y lo tendrá disponible para su lectura. A partir de este momento si el profesor de movilidad utiliza la herramienta, podrá ver en la pantalla principal un nuevo evento indicándole que un nuevo contrato de estudios ha sido generado y está pendiente de revisión.

Nuevos Eventos.-

EVENTO	CREADOR DEL EVENTO	FECHA Y HORA		
 Existe un nuevo contrato de estudios	Miguel Angel Camarena Nieto-Márquez	04-06-2011 -- 18:12:57		
Comentario: He terminado el contrato de estudios. ¿Está todo correcto?				

Figura 4.10: Notificación de un nuevo evento

Como se puede ver en la imagen en el evento se podrá observar el tipo de evento generado, el emisor del mismo y la fecha y la hora a la que fue producido. Aparecen también dos botones al lado del evento. Con el primero se puede visualizar el contrato de estudios del estudiante que creó el evento. Con el segundo se puede borrar el evento de la lista de eventos que pueda tener el usuario que esté utilizando la herramienta. También se puede apreciar debajo del evento un espacio donde se indica un mensaje procedente del emisor. Este mensaje puede ayudar a comprender o dar más información al receptor del evento del motivo de la creación del mismo.

A continuación aparecen unos diagramas de casos de uso donde se muestran algunos ejemplos que hacen que la herramienta genere eventos.

Evento que se produce cuando el estudiante termina el contrato de estudios y lo firma.

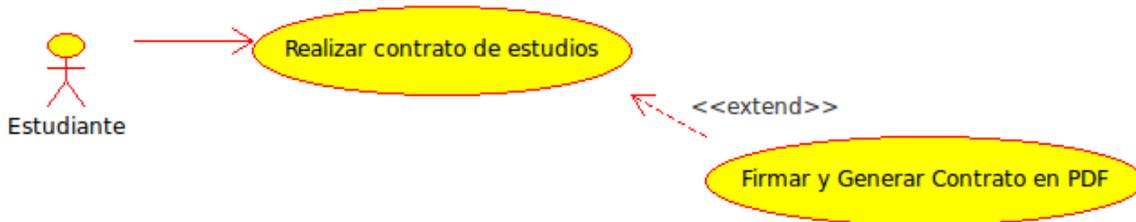


Figura 4.11: Caso de uso: Firmar contrato por el estudiante

Evento que se produce cuando el coordinador de centro califica a un alumno.

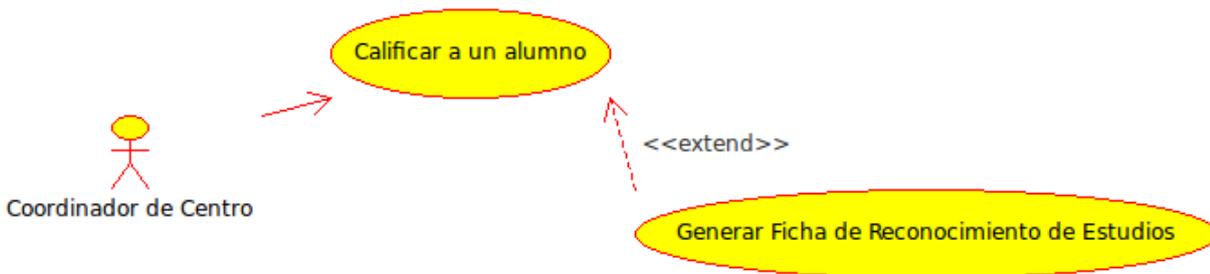


Figura 4.12: Calificar alumno por el coordinador de centro

Eventos que se producen cuando el profesor de movilidad firma o rechaza un contrato de estudios.

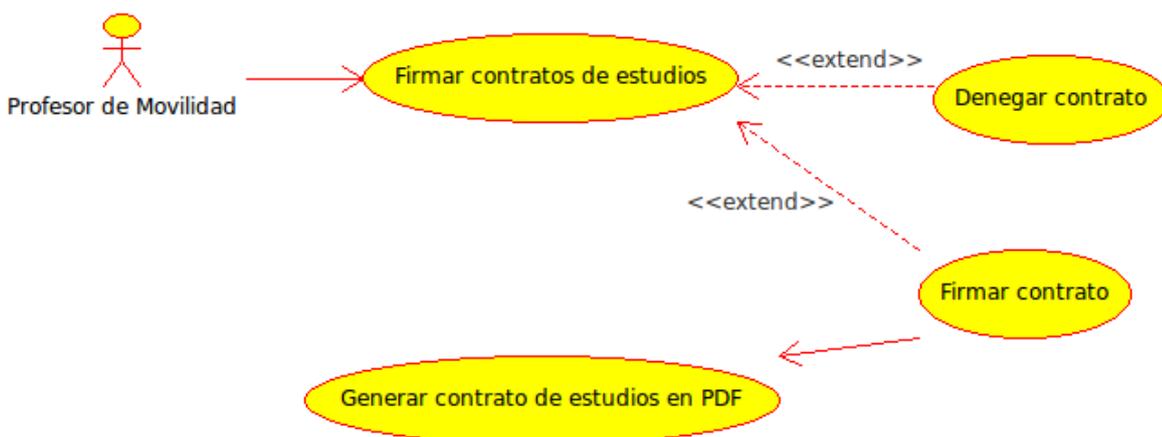


Figura 4.13: Caso de uso: Firmar o rechazar contrato de estudios por el profesor de movilidad

Eventos que se producen cuando el administrador del sistema registra un nuevo estudiante.

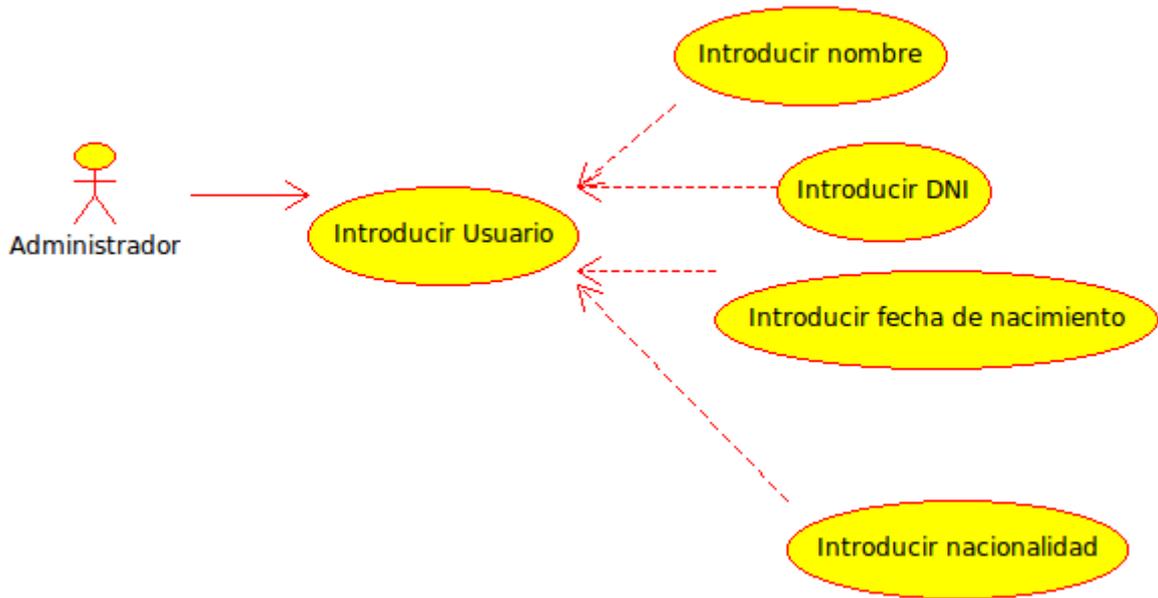


Figura 4.14: Caso de uso: *Introducir usuario*

Los eventos son creados automáticamente por la herramienta cuando ocurren ciertos sucesos clave. Esta misma se encarga de enviar el evento a los destinatarios correspondientes dependiendo del tipo de evento creado. A continuación se muestran los tipos de eventos que pueden producirse en el sistema debido a las acciones de los usuarios:

Acción	Origen	Destino/s
Se crea un nuevo estudiante Erasmus	E	A,B,C,D
Se crea un nuevo contrato de estudios	A	C
El estudiante firma el contrato de estudios	A	C
El profesor de movilidad firma el contrato de estudios	C	A,B
El profesor de movilidad rechaza el contrato de estudios	C	A
El coordinador de centro firma el contrato de estudios	B	A,C,D
El coordinador de centro rechaza el contrato de estudios	B	A,C
El coordinador institucional firma el contrato de estudios	D	A,B,C
El coordinador institucional rechaza el contrato de estudios	D	A,B,C
El estudiante va a modificar el contrato de estudios	A	C,D
El coordinador de centro ha puesto las notas	B	C,A
El administrador ha creado un nuevo estudiante Erasmus	E	C

Tabla 4.3: Ejemplo de la tabla de conversión de calificaciones europeas

- A = Estudiante
- C = Profesor de Movilidad
- B = Coordinador de Centro
- D = Coordinador Institucional
- E = Administrador del Sistema

4.7.1 Creación de un nuevo evento

Los eventos son producidos automáticamente por el sistema cuando se produce algún hito importante y alguno de los demás roles debe ser informado. El usuario no tiene que hacer nada porque la generación de eventos es casi transparente. Esto es así porque cada vez que se produzca un evento se mostrará una ventana donde el usuario podrá escribir un mensaje opcional que aparecerá cuando los destinatarios del evento vean el mismo.

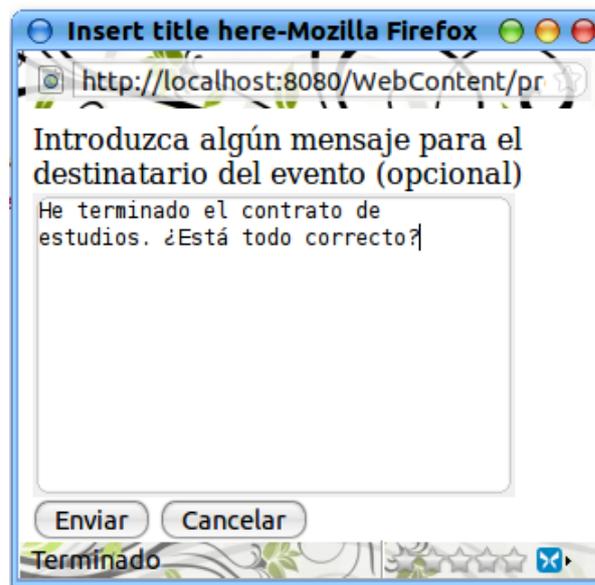


Figura 4.15: Mensaje opcional que acompaña a la generación de eventos

Cada vez que se produce un evento se ejecuta el siguiente código fuente.

```

1 //se genera el evento de contrato nuevo
2 Evento eve = new Evento();
3
4 // se anotan los destinatarios del evento
5 String destinos [] = {"B", "D", contrato};
6
7 //se anota la fecha del evento
8 Fecha fec = new Fecha();
9
10 // se registra el evento
11 eve.anotarEvento("El_Profesor_de_Movilidad_ha_firmado_un_
    contrato_de_estudios", dir+"/"+contrato+".pdf", "C", "1",
    destinos, id, contrato, fec.getFecha());

```

A continuación se explica el código detalladamente.

```
1 // se genera el evento de contrato nuevo
2 Evento eve = new Evento();
```

En la primera línea de programa se ve como se crea el objeto *eve* de la clase *Evento*.

```
1 // se anotan los destinatarios del evento
2 String destinos [] = {"B", "D", contrato};
```

A continuación se introducen en el array *destinos* todos los destinatarios del evento.

```
1 // se anota la fecha del evento
2 Fecha fec = new Fecha();
```

Después de esto se crea el objeto *fec* de la clase *Fecha*. Con ese objeto se determina la fecha y la hora a la que se produjo el evento. Así cuando los destinatarios vean el evento podrán saber en el momento en el que se produjo.

```
1 // se registra el evento
2 eve.anotarEvento("El_Profesor_de_Movilidad_ha_firmado_un_
    contrato_de_estudios", dir+"/"+contrato+".pdf", "C", "1",
    destinos, id, contrato, fec.getFecha());
```

Por último se registra el evento en la base de datos. Se le pasan al método algunos parámetros como pueden ser el motivo del evento, la dirección donde se encuentra el contrato de estudios del estudiante en cuestión, el rol emisor del evento, los destinos, etc.

4.7.2 Estructura de la Base de Datos para el diseño del sistema de eventos

Se han utilizado dos tablas en la base de datos para realizar el diseño del sistema de eventos de la herramienta. La primera tabla llamada *eventos* cuenta con seis columnas que se explican a continuación:

- **idEvento:** Este valor identifica de manera única al evento producido.
- **textoAviso:** Es el texto que se mostrará al receptor del evento.
- **enlace:** Es la url en la que se guarda el contrato de estudios del alumno con el que se está tratando en el evento.
- **usuarioEvento:** Se guarda el identificador del rol que produce el evento.

- **activo:** Esta celda puede tomar dos valores, 1 o 0. Si el evento ya ha sido visto y borrado por el receptor tendrá valor 0. En caso contrario tendrá valor 1. Si el evento tiene múltiples destinos, este campo no tomará valor 0 hasta que todos los destinatarios hayan borrado el evento.
- **mensaje:** Este campo es opcional. Es un mensaje que puede mandar el emisor al receptor cuando un evento es producido. Puedo utilizarlo para hacer cualquier aclaración, pregunta o confirmación.

idEvento	textoAviso	enlace	usuarioEvento	activo	mensaje
1	Ya está disponible la Ficha de Reconocimiento de Estudios	/home/porrito/Escritorio/Informe/1.pdf	B	1	Ya está todo. Un saludo.
2	Existe un nuevo contrato de estudios	/home/porrito/Escritorio/Informe/1.pdf	1	1	firmado por el alumno
3	El Profesor de Movilidad ha firmado un contrato de estudios	/home/porrito/Escritorio/Informe/1.pdf	C	1	contrato firmado para ma
4	Se ha modificado un contrato de estudios	/home/porrito/Escritorio/Informe/1.pdf	1	1	

Figura 4.16: Primera tabla del sistema de gestión de eventos

En la segunda tabla, tres valores conjuntos actuarán de clave primaria de la tabla. Esto es debido a que un mismo evento puede tener varios destinatarios. Esta tabla es llamada *eventos_destino* y cuenta con siete columnas que se explican a continuación:

- **idEvento:** Este valor identifica un evento dentro de esta tabla. Sirve de clave primaria junto con los dos próximos valores.
- **origen:** En este campo se guarda el rol que ha producido el evento.
- **destino:** En este campo se guarda el destinatario que recibirá el evento.
- **activo:** Esta celda puede tomar dos valores, 1 o 0. Si el evento ya ha sido visto y borrado por el receptor tendrá valor 0. En caso contrario tendrá valor 1.
- **idUsuario:** Aquí se guarda el id del usuario que ha producido el evento.
- **contrato:** En esta celda se almacena el identificador del contrato de estudios del alumno.
- **fecha:** En este campo se almacena la fecha y hora en la que se ha producido el evento.

idEvento	origen	destino	activo	idUsuario	contrato	fecha
1	B	1	0	5	1	2011-06-07 18:55:36
2	1	C	0	1	1	2011-06-07 18:57:58
3	C	B	1	3	1	2011-06-07 18:59:07
3	C	1	0	3	1	2011-06-07 18:59:07
4	1	C	0	1	1	2011-06-09 11:57:32

Figura 4.17: Segunda tabla del sistema de gestión de eventos

Esta herramienta puede ser manejada por muchos usuarios. Los cambios que hagan algunos influirán directamente en otros por lo que era totalmente necesario desarrollar este sistema de control de eventos. Sobre todo para los coordinadores, ya que estos podrían tener por ejemplo a una veintena de alumnos haciendo cambios en sus contratos y sería muy complicado seguir la pista de los cambios que realiza cada uno sin que la herramienta fuese avisando sobre esto.

En el futuro es posible que la normativa actual pudiera cambiar y se necesite añadir algún tipo de evento nuevo. No habría que hacer ningún cambio radical en la herramienta ni modificar la base de datos. Simplemente habría que añadir en el código la instrucción que genera el evento indicando los valores que son necesarios para añadir a la base de datos.

4.7.3 Diagrama de clases relacionadas con el control de eventos

A continuación se muestra un diagrama de clases donde aparecen las clases que tienen alguna relación con el sistema de control de eventos. Se pueden apreciar cuatro clases.

- **Usuario:** Clase que se compone de todos los atributos del usuario. El usuario de rol administrador puede registrar nuevos usuarios generándose nuevos eventos.
- **Contrato:** Cuando se genere un nuevo contrato de estudios también se creará un nuevo evento.
- **Evento:** Esta clase contiene todas las operaciones para gestionar el control de eventos del sistema.
- **Fecha:** La clase Fecha se utilizará para obtener la fecha y hora actual en la que se ha producido un evento.

En el diagrama se puede apreciar como un administrador al registrar nuevos usuarios puede hacer que se genere uno o más eventos. De igual forma, cuando se genere un nuevo contrato de estudios o sea firmado por parte de los coordinadores pueden ser generados uno o varios eventos que serán enviados a los destinatarios correspondientes.

La clase Fecha proporcionará una fecha y hora por cada evento que se produce.

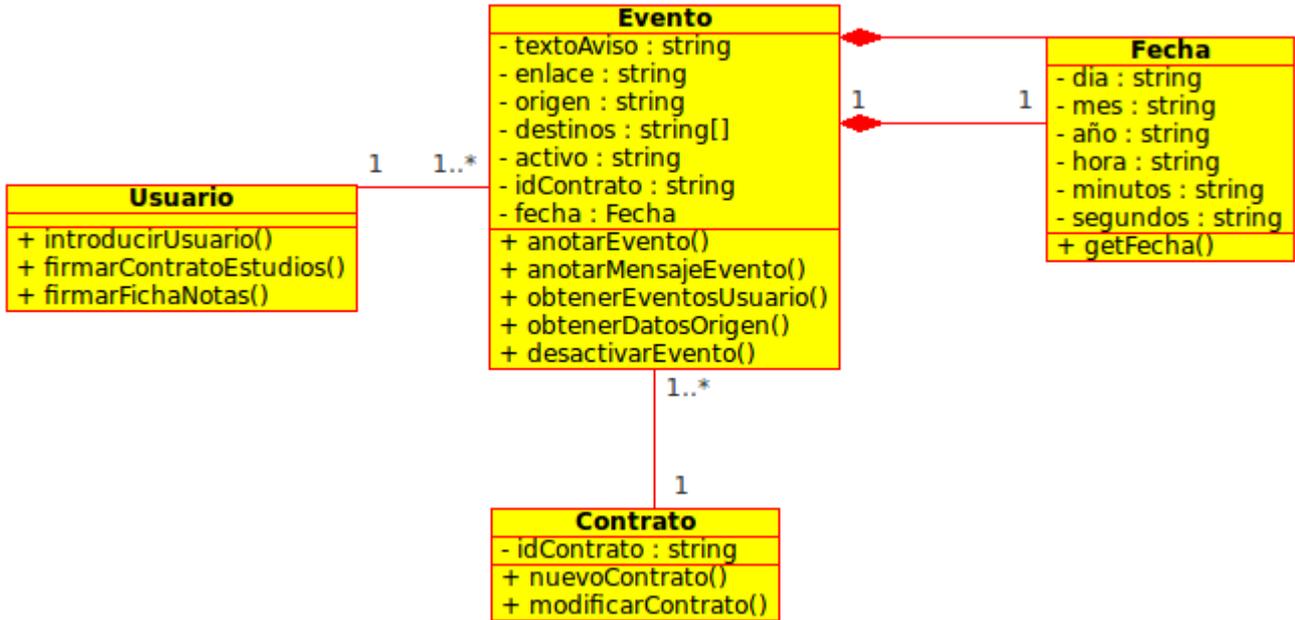


Figura 4.18: Diagrama de clases relacionadas con sistema de control de eventos

4.8 Cálculo de Notas y Equivalencias

Cuando un estudiante finaliza su beca Erasmus y regresa a España debe traer consigo un documento donde se reflejen las calificaciones obtenidas en las asignaturas del país extranjero. Este documento debe entregarlo al coordinador de centro que es el encargado de revisar dichas calificaciones y poner las notas definitivas en el expediente del estudiante.

Hay que tener en cuenta que la forma de evaluar a los estudiantes en cada país es distinta. Por lo que existe una *tabla de conversión de calificaciones europeas* (Véase Anexo E). En ella aparecen los países involucrados con el programa Erasmus y la forma de calificar que existe en el país.

Era realmente tedioso para el coordinador de centro tener que hacer las conversiones de las calificaciones de todos los estudiantes que estaban cursando la beca Erasmus. Esto es debido a que tenía que estar mirando el país donde el estudiante había cursado la beca, y calcular la equivalencia entre la nota del país destino y el sistema de calificación en España.

Véase el ejemplo de un alumno que ha cursado su beca Erasmus en Bélgica. Este alumno trae el siguiente acta de notas de Bélgica:

- Artificial Intelligence - 11.5
- Software Engineering - 15
- Human Computer Interaction - 17.2
- Digital Image Computer- 8.5

El coordinador de centro debe consultar la tabla de conversión de calificaciones europea para ver cuáles son las notas que corresponden al alumno (Véase Anexo E). Tiene que encuadrar cada nota dentro de los intervalos correspondientes en la tabla. Por ejemplo, para la asignatura *Artificial Intelligence* se debe ir al país de *Bélgica* y al intervalo *13/12/11*. Esto es porque 10.5 está comprendido entre 13 y 11. Ahora se debe mirar con que nota del sistema de calificación español se corresponde dicho intervalo. En este caso sería con un 6, por lo que el alumno obtendría un *Aprobado* en esta asignatura.

La herramienta actúa de un modo similar para realizar el cálculo automático de las calificaciones. Se ha introducido la tabla de conversión de calificaciones europea en la base de datos tal y como se puede ver en la imagen 4.19. El coordinador de centro tiene que introducir en el sistema el acta de calificaciones que el alumno ha traído del país extranjero. El sistema recogerá estos valores encuadrándolos dentro de los intervalos correspondientes según la

tabla de la base de datos y el país donde se cursó la beca Erasmus. Seguidamente el programa asociará esos intervalos con la nota que le corresponde en el sistema de calificación español y se obtendrán las notas definitivas.

El ahorro de trabajo para el coordinador de centro es bastante considerable. Por ejemplo, si se tiene en cuenta que un alumno Erasmus puede tener cursar una media de 10 asignaturas y que un coordinador de centro puede tener a su cargo unos ocho alumnos al año, tendría que realizar 80 cálculos de notas. Teniendo en cuenta que los cálculos serán diferentes para cada alumno debido al país destino del mismo.

A continuación se puede ver un ejemplo comparando el sistema de calificación Español con el de dos países europeos que participan en el programa Erasmus.

País	A	B	C	D	E	F/FX
España (numérica)	10	9	8-7	6	5	4-1
España (cualitativa)	Matricula de Honor	Sobresaliente	Notable	Aprobado	Aprobado	Suspense
Alemania	1.0	1-	2	3	4	5/6
Bélgica	20	19/18/17	16/15/14	13/12/11	10	9/8/7

Tabla 4.4: Ejemplo de la tabla de conversión de calificaciones europeas

4.8.1 Generación del Acta Final

Una vez las calificaciones finales hayan sido obtenidas por completo el coordinador de centro podrá generar un acta final de notas en un documento en formato pdf. Este documento debe ser firmado por el coordinador de centro, por lo que antes de ser generada la ficha de reconocimiento de estudios se procederá al proceso de firma del mismo.

A continuación se muestran los pasos que se dan en la herramienta para finalmente obtener la ficha de reconocimiento de estudios.

```

1 <html>
2 </ head>
3 </ head>
4 <body>
5
6 <%
7 // se obtiene el usuario
8 String id = (String) session.getAttribute("id");
9
10 // se recogen los parámetros

```

```
11 String contrato = request.getParameter("contrato");
12 String fichero = request.getParameter("fichero");
13 String clave = request.getParameter("clave");
14
15 //se crean las variables y objetos necesarios
16 Firmar fir = new Firmar();
17 AsignaturasAlumno alum = new AsignaturasAlumno();
18
19 //se actualiza la tabla de contratos
20 Contrato cont = new Contrato();
21 cont.actualizarContratosPdf(id);
22
23 // se genera el evento de contrato nuevo
24 Evento eve = new Evento();
25 // se anotan los destinatarios del evento
26 String destinos[] = {contrato};
27 // se anota la fecha del evento
28 Fecha fec = new Fecha();
29 // se registra el evento
30 eve.anotarEvento("Ya está disponible la Ficha de Reconocimiento de Estudios", dir+"/"+contrato+".pdf", "B", "1", destinos, id,
    contrato, fec.getFecha());
31
32 // se general el informe
33 NotasPdf not = new NotasPdf(contrato, dir);
34
35 //se firma el contrato
36 fir.firmarPdf(contrato, fichero, clave, "Z");
37
38 //se borran los ficheros auxiliares
39 File origen = new File(dir+"/notas"+contrato+"_signed.pdf");
40 File destino = new File(dir+"/notas"+contrato+".pdf");
41 origen.renameTo(destino);
42
43 // se habilita el modulo ver notas
44 alum.cambiarFichaNotas(contrato);
45 %>
46
47 </body>
48 </html>
```

Después de que el anterior código sea ejecutado se genera el acta de notas del estudiante. A este documento se le denomina *Ficha de Reconocimiento de Estudios*. Se puede ver el documento original aquí. (Véase Anexo D)

A continuación se muestra una ficha de reconocimiento de estudios generada con la herra-

mienta. Este documento tiene la misma validez que el documento original en papel ya que está firmado digitalmente por el coordinador de centro.



FICHA DE RECONOCIMIENTO DE ESTUDIOS

Datos Personales:

Nombre y Miguel Angel Camarena Nieto-Márquez DNI: 70580107F

Facultad: E. S. Informatica

Duración de la estancia: 9 meses

Fechas de inicio y 2011-09-15 2012-06-15

Materias cursadas en la Vilnius University Pais: Lituania

Código Ects	Asignatura	Créditos Ects	Calificación
FS5324	Databases	5	5
NV8872	Computer Architecture	5	7
UR9865	Software Engineering	6	4
KKKK	Information Business	6	8.9

Propuesta de conversión de las calificaciones extranjeras al sistema

Código Ects	Asignatura	Créditos Ects	Calificación
42409	Bases de Datos	6	5 - Aprobado
42503	Arquitectura de Computadores	6	7.5 - Notable
42504	Ingeniería del Software II	6	4 - Suspenso
42505	Aspectos Profesionales de la Informática	6	7.5 - Notable

NOTA: Este documento deberá ir acompañado del formulario de candidatura original y de las notas originales obtenidas en la universidad extranjera.

Firma del Coordinador
de

Nombre: Camarena

Fecha: Camarena

4.8.2 Estructura de la Base de Datos para diseñar el modo de conversión de calificaciones europeas

Se ha utilizado una tabla para realizar el diseño de la conversión de calificaciones europeas en la base de datos. Esta tabla consta de trece columnas ya que se han utilizado intervalos para acotar la conversión de notas.

📍 pais	A1	A2	B1	B2	C1	C2	D1	D2	E1	E2	F1	F2
Espania	10	10	9.9	9	8.9	7	6.9	6	5.9	5	4.9	0
Alemania	1	1	1	1.9	2	2.9	3	3.9	4	4.9	5	6
Austria	1	1	1	1.9	2	2.9	3	3.9	4	4.9	5	5
Bélgica	20	20	19.9	17	16.9	14	13.9	11	10.9	10	9.9	7
Dinamarca	13	13	12.9	10	9.9	9	8.9	8	7.9	7	6.9	6
Finlandia	3	3	2.9	2.5	2.4	2	1.9	1.5	1.4	1	1	0
Francia	20	20	19.9	16	15.9	14	13.9	12	11.9	10	9.9	7
Grecia	10	10	9.9	8	7.9	7	6.9	6	5.9	5	4.9	1

Figura 4.19: *Tabla de conversión de calificaciones europea*

En la primera columna se especifica el país al que se está haciendo referencia. Hay que tener en cuenta que no todos los países tienen un sistema de calificación definido. En este caso, estos países tomarán por defecto el mismo sistema de calificaciones que en España.

En el resto de columnas se especifican intervalos. Se ha dividido el sistema de calificación en seis intervalos distintos. Tomando como modelo el sistema de calificación español estos serían los distintos intervalos:

- **A1-A2:** Matricula de honor. Nota = 10
- **B1-B2:** Sobresaliente. Nota: $9,9 \leq x \leq 9$
- **C1-C2:** Notable. Nota: $8,9 \leq x \leq 7$
- **D1-D2:** Aprobado. Nota: $6,9 \leq x \leq 6$
- **E1-E2:** Aprobado. Nota: $5,9 \leq x \leq 5$
- **F1-F2:** Suspenso. Nota: $4,9 \leq x \leq 1$

El sistema detectará qué país debe escoger para tener en cuenta el sistema de calificación vigente en el mismo. Después de esto cogerá la nota que tiene que convertir y teniendo en cuenta la *tabla de conversión de notas europea* la encuadrará en el intervalo adecuado dando como resultado la calificación correcta de acuerdo al sistema español.

Cambios en la normativa actual de calificaciones europeo no afectarán al funcionamiento de la herramienta. El coordinador de centro, que es el rol encargado de gestionar y utilizar este módulo, podrá cambiar desde el propio menú de la herramienta la configuración de las calificaciones de los distintos países si esta cambiase en un futuro.

4.8.3 Diagrama de clases relacionadas con el cálculo de notas

A continuación se muestra un diagrama de clases donde aparecen las clases que tienen alguna relación con el cálculo de notas. Se pueden apreciar tres clases.

- **Usuario:** Clase que se compone de todos los atributos del usuario. El usuario de rol coordinador de centro puede modificar la tabla de calificaciones europeo de acuerdo a la normativa de cada país. También se encarga de poner las notas definitivas a los estudiantes basándose en el sistema de calificación español. Después de esto firmará la ficha de reconocimiento de estudios.
- **Asignatura:** Clase que contiene todos los atributos que puede tener una asignatura. Cada asignatura tendrá asociada una calificación.
- **Calificaciones:** Esta clase contiene todas las operaciones para editar la tabla de conversión de calificaciones europea. También contiene métodos para calcular las notas de los estudiantes convertidas al sistema de calificación español.

En el diagrama se puede ver como el coordinador de centro puede modificar una o varias calificaciones. También puede poner las notas a uno o varios estudiantes. El coordinador de centro puede también firmar una o varias fichas de reconocimiento de estudios. Cada estudiante tendrá una o varias asignaturas y cada una de ellas tendrá una calificación.

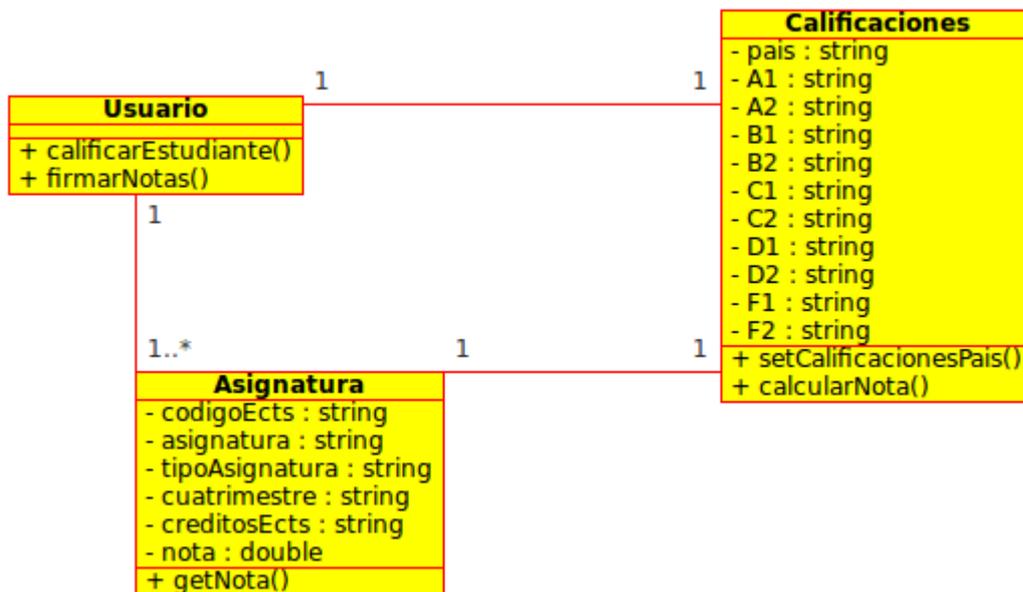


Figura 4.20: Diagrama de clases relacionadas con el cálculo de notas

4.9 Autenticación de Documentos en el Sistema Desarrollado

Con el desarrollo de esta herramienta se pretende mejorar el proceso tradicional en el que un alumno puede cursar una beca Erasmus. Pero al intentar hacer mejor el proceso tradicional no se pueden perder ciertas propiedades que se tenían como la garantía de unos contratos legales en papel firmados por el alumno y los coordinadores pertinentes. Se debe buscar el camino para generar un contrato de estudios similar al tradicional que tenga la misma validez. El sistema tendrá implementado un sistema de autenticación con el que los usuarios y roles podrán firmar los documentos que sean necesarios. Los documentos resultantes de aplicar el sistema de firmas de la herramienta serán igualmente válidos y tendrán la misma integridad que si el documento hubiera sido firmado en papel.

4.9.1 JSigPdf

Después de barajar entre varias opciones se optó por generar el contrato de estudios en formato PDF². Aunque esta decisión por si sola no era válida, ya que se pretende convertir cualquier método antiguo a un proceso totalmente digitalizado. Entonces generar un contrato en pdf para que el alumno posteriormente tuviera que imprimirlo y llevarlo personalmente a los coordinadores para que lo firmasen no era una buena opción. Se buscó entonces una forma de poder firmar los contratos digitalmente para que tuvieran la misma validez que un contrato firmado en papel.

Se optó por usar la herramienta *JSigPdf*[7]. Es una aplicación Java la cual permite añadir firmas digitales a documentos PDF. Esta puede ser usada por si sola o añadirla para que trabaje junto a *OpenOffice.org*[9]. JSigPdf es software de código abierto y puede ser libremente usado en sectores privados y de negocio.

Estas son algunas de las características de la herramienta:

- Firmas visibles en el documento. Una imagen y/o descripción se puede agregar en un documento PDF. La página o posición puede ser especificada.
- Varios tipos de claves distintas pueden ser utilizadas.
- Multifirma en el mismo documento

²(acrónimo del inglés portable document format, formato de documento portátil) es un formato de almacenamiento de documentos, desarrollado por la empresa Adobe Systems. Este formato es de tipo compuesto (imagen vectorial, mapa de bits y texto)

- Marcas de tiempo³ con la autoridad de la firma TSA. Es una autoridad de sellado de tiempo que tiene la funcionalidad de agregar la fecha y la hora a una determinada acción. La firma electrónica junto con el sello de tiempo forman una prueba irrefutable que responde al quién y al cuándo se desarrolló tal acción electrónica dentro de cualquier entorno de operación.
- Certificado de validación a través de CRL⁴ y/o OCSP⁵.
- Ajuste del nivel de certificación.
- Ajuste del algoritmo Hash.
- Encriptado del Pdf
- Interfaz en Swing intuitiva
- Soporte para procesamiento batch (controlado por línea de comandos, sin interfaz gráfica)

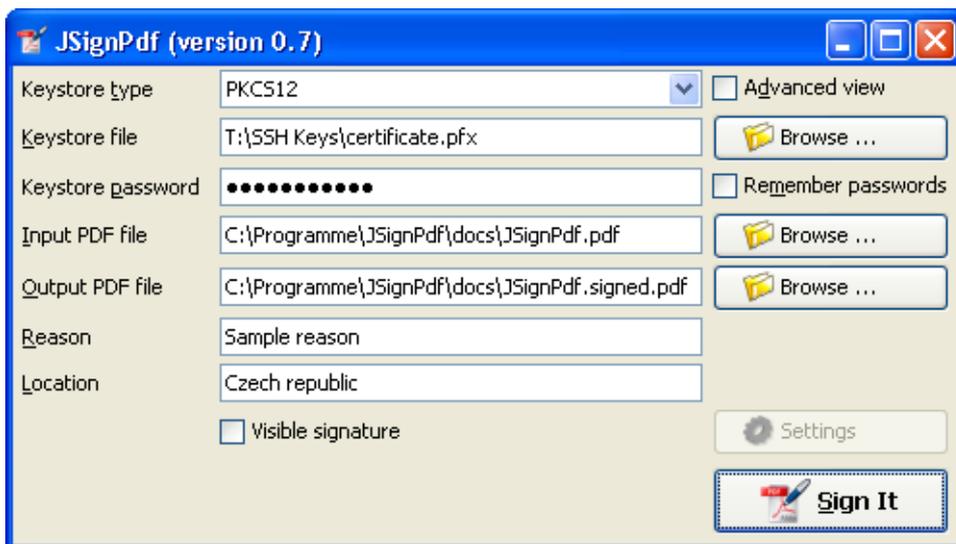


Figura 4.21: Interfaz gráfica de JSigntool. (versión 0.7)

Esta herramienta permitía firmar documentos en pdf, justo lo que se necesitaba para tener un contrato de estudios digital similar al contrato en papel. La herramienta utiliza certificados digitales para firmar los documentos. Cuando un usuario desee firmar un documento

³El sellado de tiempo o timestamping es un mecanismo on-line que permite demostrar que una serie de datos han existido y no han sido alterados desde un instante específico en el tiempo.

⁴La *Lista de Certificados Revocados* contiene el número de serie de todos los certificados emitidos por una Autoridad de Certificación y que, por algún motivo han dejado de ser válidos de manera previa a la expiración de su periodo de validez original.

⁵El servicio OCSP (Online Certificate Status Protocol), definido en el estándar RFC-2560, proporciona a los usuarios y las aplicaciones un método ágil y rápido de obtener el estado de un certificado, evitando tener que descargar la CRL

debe estar en posesión de un certificado digital soportado por este software.

Algunos de los tipos de certificados soportados son:

- **PKCS#12:** Claves almacenadas en archivos .p12 y .pfx
- **JKS:** Claves del almacén de Java
- **WINDOWS-MY:** Soportado solo en Ms Windows con Java 6 o superior. Se pueden usar directamente los certificados importados desde el sistema operativo.

Para este proyecto se decidió utilizar las claves del tipo PKCS#12 aprovechando las posibilidades de obtener este tipo de certificados que brinda la Administración Pública. En España, las distintas Administraciones están apostando decididamente por Internet como vía de comunicación, creando webs con información de interés público a disposición de la ciudadanía. Estas iniciativas están teniendo una gran aceptación y repercusión positiva en la opinión pública, que está demandando una utilización más generalizada de la red.

La más ambiciosa de estas iniciativas, puestas en marcha por la Administración, es el denominado proyecto *CERES* (CERTificación ESpañola) que lidera la *Fábrica Nacional de Moneda y Timbre*⁶, y que en líneas generales, consiste en establecer una Entidad Pública de Certificación, que permita autenticar y garantizar la confidencialidad de las comunicaciones entre ciudadanos, empresas u otras instituciones y administraciones públicas a través de las redes abiertas de comunicación.

El objetivo principal de CERES es la securización de las comunicaciones electrónicas con la Administración, siendo un intermediario transparente al usuario que garantizará a ciudadanos y Administraciones la identidad de ambos partícipes en una comunicación, así como la confidencialidad e integridad del mensaje enviado.

Para ello, CERES utiliza técnicas y sistemas criptográficos basados en lo que se conoce como sistema de clave pública, con dos características básicas:

La identidad del usuario, al igual que su capacidad de firma, se encuentra, en el caso de máxima seguridad, almacenada en una tarjeta inteligente, que no puede ser accesible salvo por su propietario cuando introduzca el número de identificación personal, similar a la clave de una tarjeta de crédito. En caso de no utilizar tarjeta, el perfil criptográfico queda almacenado en un fichero, siendo necesario también un PIN de acceso.

⁶Es una Entidad Pública Empresarial dependiente de la Subsecretaría de Economía y Hacienda del Ministerio de Economía y Hacienda

El sistema es completamente transparente al usuario, es decir, no es necesario conocer ninguna técnica criptográfica para realizar o verificar una firma electrónica o cifrar o descifrar un mensaje.

Una vez obtenido el correspondiente certificado digital emitido por CERES, se integró el software JSignPdf en la herramienta para que permitiese firmar el contrato de estudios de los alumnos. Cuando algún agente desea firmar el contrato de estudios le aparecerá en pantalla un formulario como este:

Firmar Contrato.-

Tipo de Certificado: PKCS12 ▼

Certificado: Examinar...

Clave:

Firmar

Figura 4.22: *Formulario para firmar contratos utilizando un certificado digital*

Se puede elegir el tipo de certificado con el que firmar el contrato de estudios. Actualmente la herramienta solo soporta la firma con certificados PKCS#12. También hay que elegir la contraseña del certificado y la ruta del mismo en el disco duro.

Como se puede ver en la figura 4.21, esta herramienta cuenta con interfaz gráfica. Está desarrollada en Java utilizando AWT⁷. Se trataba de integrar dicha interfaz gráfica en la web para que los usuarios pudieran firmar digitalmente los documentos. Debido a que JSignPdf utilizaba AWT, la integración no era posible de realizar en JSP. Así que se optó por utilizar el módulo de JSignPdf que permite utilizar esta herramienta por línea de comandos. Por lo que el usuario solo necesita introducir por medio de un formulario algunos campos necesarios. Los demás están configurados en la herramienta por defecto, por lo que todo el proceso de firma es totalmente transparente al usuario.

Internamente el proceso de firma ocurre de la siguiente forma. Primeramente se rellena el formulario que contendrá los datos necesarios para hacer la firma. Estos campos son los siguientes.

⁷La Abstract Window Toolkit (AWT, en español Kit de Herramientas de Ventana Abstracta) es un kit de herramientas de gráficos, interfaz de usuario, y sistema de ventanas independiente de la plataforma original de Java.

- **Tipo de certificado:** Aquí se elegirá el tipo de certificado digital para firmar. Se elegirá uno de la lista desplegable.
- **Certificado:** Aquí se seleccionará del directorio local el certificado con el cuál se firmará.
- **Clave:** En este campo se anotará la clave asociada al certificado.

```
1 <html>
2 <head>
3 </head>
4 <body>
5
6 <form action=" ../ dominio / firmar_pm . jsp " method=" post ">
7   <table>
8     <tr>
9       <td>
10        Tipo de Certificado :
11      </td>
12      <td>
13        <select name=" tipo ">
14          <option value="PKCS12">PKCS12</ option>
15        </select>
16      </td>
17    </tr>
18
19    <tr>
20      <td>
21        Certificado :
22      </td>
23      <td>
24        <input type=" file " value="" name=" fichero ">
25      </td>
26    </tr>
27
28    <tr>
29      <td>
30        Clave :
31      </td>
32      <td>
33        <input type=" password " value="" name=" clave ">
34      </td>
35    </tr>
36
37    <tr>
38      <td>
```

```

39     <input type="submit" value="Firmar" onclick="javascript:
        abrirVentana()">
40     </td>
41     </tr>
42 </table>
43 </form>
44 <input type="button" value="Volver" onclick="window.location.
        href='present_firmar_pm.jsp'">
45
46 </body>
47 </html>

```

Si los campos del formulario han sido rellenados correctamente se procederá al proceso de firma. Internamente se creará un objeto de la clase *Firma* y se llamará al método *firmarPdf* para ejecutar la firma.

```

1
2 // se crean el objeto de la clase Firma
3 Firmar firma = new Firmar();
4
5 // se firma el contrato
6 firma.firmarPdf(idContrato, fichero, clave, rol);

```

En el método *firmarPdf* estará el código fuente que servirá para firmar digitalmente el documento en formato PDF. A continuación se muestra un ejemplo del código que se aplica cuando el profesor de movilidad realiza una firma.

```

1 // firma el profesor de movilidad
2 if(rol.equals("C")){
3     String comando = "java -jar /home/user/JSignPdf/jsignpdf
        -1.1.1/JSignPdf.jar -kst_PKCS12 -ksf_/home/user/
        Certificado/" + fichero + "_ksp_" + clave + "_pr_
        DISALLOW_PRINTING_/home/user/Contrato/" + contrato + ".pdf -d_
        /home/porrito/Esitorio/Contrato_V_pg_2_urx_169.79167_
        -ury_59.696697_ -llx_55.416668_ -lly_10.214884_ -fs_6";
4 }

```

Después de ejecutarse estas líneas se habrá generado un documento firmado digitalmente en formato PDF.

En la imagen 4.23 se puede ver en detalle como sería la firma que se genera en los documentos firmados con la herramienta.

Digitally signed by NOMBRE
CAMARENA NIETO MÁRQUEZ
MIGUEL ÁNGEL - NIF 70580107F
Date: 2011.06.09 18:31:05 CEST

Figura 4.23: *Firma digital producida con la herramienta*

Esta firma está formada por los siguientes datos:

- **Autor:** Nombre y apellidos de la persona que firma el documento.
- **NIF:** Número del documento nacional de identidad.
- **Fecha y hora:** Fecha y hora en que se produjo la firma del documento.

Existen dos documentos que serán firmados digitalmente. El contrato de estudios y la ficha de reconocimiento de estudios.

El contrato de estudios está compuesto de dos hojas si no tiene ninguna modificación de asignaturas. En este caso las firmas aparecerán en la parte inferior de la segunda hoja. Si el contrato tiene modificaciones de asignaturas las firmas aparecerán también en la parte inferior de la tercera hoja. El contrato de estudios será firmado por cuatro roles distintos. El estudiante, el profesor de movilidad, el coordinador de centro y el coordinador institucional.

La ficha de reconocimiento de estudios está compuesta de una sola hoja. Este documento será firmado por el coordinador de centro. La firma aparecerá en la parte inferior de la hoja.

Se puede ver el documento original del contrato de estudios aquí. (Véase Anexo C)

A continuación se adjunta un contrato de estudios generado por la herramienta y firmado digitalmente por el alumno propietario del contrato.

FORMULARIO DE CANDIDATURA, ESTUDIANTES UCLM



DATOS PERSONALES DEL ESTUDIANTE

Primer Apellido: Camarena **Segundo Apellido:** Nieto-Márquez **Nombre:** Miguel Angel
Fecha de Nacimiento: 1984-01-04 **Sexo:** Hombre **Nacionalidad:** España
Dirección: Francisco Elipe 32 A **Email:** peporron17@hotmail.com **D.N.I.:** 70580107 **Teléfono:** 680826262

UNIVERSIDAD ASOCIADA

Nombre legal de la Institución: Vilnius University **Dirección:** Naugarduko 24
Coordinador Institucional.-
Nombre: Remijius leipus
Teléfono: 0024867956876 **Fax:** 0024867956866 **Email:** remijius@coord.lt

Profesor responsable de la movilidad:

Nombre: Kristina Lapin
Teléfono: 0024867944662 **Fax:** 0024867956897 **Email:** kristina@prof.lt

UNIVERSIDAD DE CASTILLA-LA MANCHA

Coordinador Institucional.-

Nombre: Andres Perez Fernandez **Dirección:** Lirio 4
Teléfono: 675884345 **Fax:** 926564543 **Email:** andres.perez@uclm.es

Coordinador de Centro.-

Nombre: Pedro Fernandez Caba **Dirección:** Toledo 35 3A
Teléfono: 654545778 **Fax:** 926558873 **Email:** pedro.

Centro: E. S. Informatica

Profesor responsable de la movilidad:

Nombre: Manuel Galan Nieto **Dirección:** Virgen del Espino 23
Teléfono: 655887633 **Fax:** 926929287 **Email:** manuel.galan@uclm.es

APTITUDES LINGÜÍSTICAS

Lengua materna: Castellano **Lengua en universidad destino:** Ingles **Nivel de conocimiento:** Avanzado

ESTUDIOS REALIZADOS Y EN CURSO

Licenciatura-Diplomatura- I.T. Gestión **Duración (años):** 5 **Curso que realiza:** 2010-2011

PERIODO DE ESTUDIOS SOLICITADO

Duración del período de estudios (meses): 9 **Fecha de inicio:** 2011-09-15 **Fecha de fin:** 2012-06-15

FORMULARIO DE CANDIDATURA, ESTUDIANTES UCLM

ASIGNATURAS DE LA UCLM

Código Ects	Código	Asignatura	Tipo	Créditos Ects
42403	1	Sistemas Operativos I	B	6
42404	2	Ingeniería del Software I	B	6
42503	3	Arquitectura de Computadores	B	6

ASIGNATURAS DE LA UNIVERSIDAD

Código Ects	Código	Asignatura	Créditos Ects
LKI4321	1	Operating Systems	6
YUI8823	3	Computer Architecture	5

Firma del Estudiante:

Firma del Profesor de
Movilidad:

Firma del Coordinador
de Centro:

Firma del Coordinador
Institucional:

4.10 Aprendizaje de Conflictos

Durante el proceso de elaboración de los contratos de estudios normalmente se producen una serie de conflictos que provocan modificaciones y demora en el tiempo del proceso global. Un tipo de conflicto podría ser la incompatibilidad de alguna asignatura del país extranjero en la que se haya matriculado el estudiante. Puede ser debido a que esta asignatura sea impartida en el idioma nativo del país y no en inglés. Esto probablemente podría ser un gran problema para el alumno. Si algún alumno anterior a este se matriculó de esta asignatura seguramente se realizó una modificación del contrato para cursar otra que se impartiera en inglés. Al hacer este cambio la herramienta registra el tipo de conflicto y los motivos que han llevado al alumno a realizarlo. Después de esto, cuando futuros alumnos vayan a elegir esta asignatura, la herramienta advertirá de este tipo de conflictos para asesorar al alumno. La herramienta irá aprendiendo con el paso del tiempo y de su propia experiencia podrá asesorar a los alumnos a la hora de realizar el contrato de estudios. El algoritmo irá almacenando la experiencia de todos los alumnos que realizan el contrato de estudios. Cuando algún alumno realiza cualquier modificación en su contrato de estudios tiene que indicar el motivo por el cuál realiza el cambio solicitado. Todos estos motivos son recogidos y almacenados para un futuro uso. Posteriormente cuando algún alumno vaya a realizar su contrato de estudios la herramienta podrá aconsejarle mostrando los motivos que llevaron a pasados alumnos a realizar modificaciones con la asignatura seleccionada. De esta forma el alumno tendrá un valor añadido de información para poder sopesar si sigue queriendo cursar tal asignatura.

Existen también otra serie de conflictos que se producen a la hora de realizar el contrato de estudios y que la herramienta debe informar al usuario para que este pueda solucionarlos y generar un contrato de estudios sin errores. Algunos de estos conflictos pueden ser los siguientes:

- **Asignaturas de primer curso:** Un estudiante no puede cursar en Erasmus asignaturas de primer curso.
- **Número de créditos correcto:** Existen un mínimo y un máximo de créditos que el alumno puede coger en su estancia Erasmus. El número de créditos varía dependiendo del tiempo que dure la beca Erasmus.
- **Cuatrimestre correcto:** El estudiante debe elegir asignaturas que se impartan en el mismo cuatrimestre en el que curse su beca Erasmus.

4.10.1 Tipos de conflictos

Uno de los posibles conflictos que pueden ocurrir es que algún alumno intente cursar durante la beca Erasmus alguna asignatura de primer curso. Según la normativa (Véase Anexo B) esto no es posible y solo se podrán cursar asignaturas de segundo curso o posteriores.

Algoritmo 1: Asignaturas de primer curso

```
Alumno selecciona asignatura;  
/* La asignatura elegida pertenece a un curso */  
if cursoAsignatura es distinto de primer curso then  
    Se comprueba si el cuatrimestre es correcto;  
    /* Ver algoritmo en la página siguiente */  
else  
    No se introduce asignatura;  
    Se muestra mensaje de error explicando el motivo;  
    FIN;
```

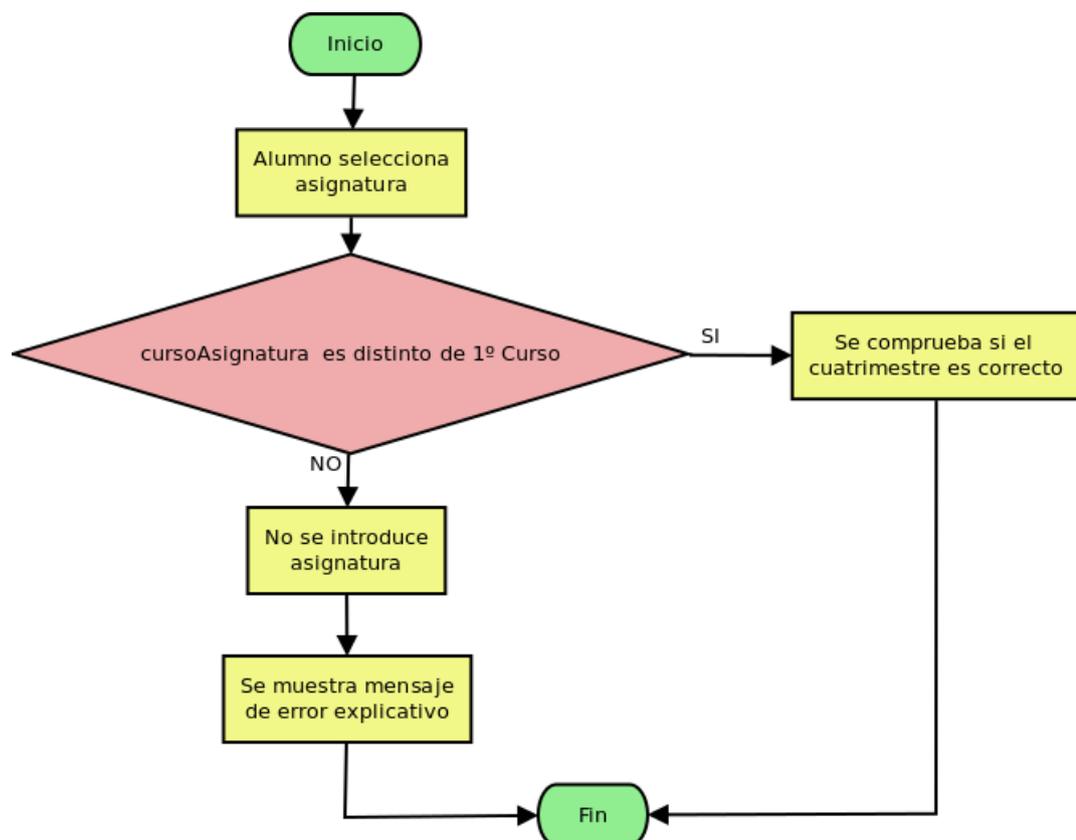


Figura 4.24: Algoritmo (Curso correcto)

La duración de la beca de un alumno Erasmus puede ser diferente dependiendo de las necesidades de este o del país que le haya sido asignada. Puede tener una duración anual o de un cuatrimestre, pudiendo ser el primero o segundo si la universidad destino lo permite.

Algoritmo 2: Cuatrimestre correcto

Se obtiene el periodo del alumno;

/ periodoAlumno = Tiempo de estancia del alumno (anual, primer cuatrimestre o segundo cuatrimestre) */*

Alumno selecciona asignatura;

/ La asignatura también tiene un periodo en el que es impartida */*

if *periodoAlumno = Anual o periodoAlumno = periodoAsignatura* **then**

 Se comprueba si el número de créditos es correcto es correcto;

/ Ver algoritmo en la página siguiente */*

else

 No se introduce asignatura;

 Se muestra mensaje de error explicando el motivo;

FIN;

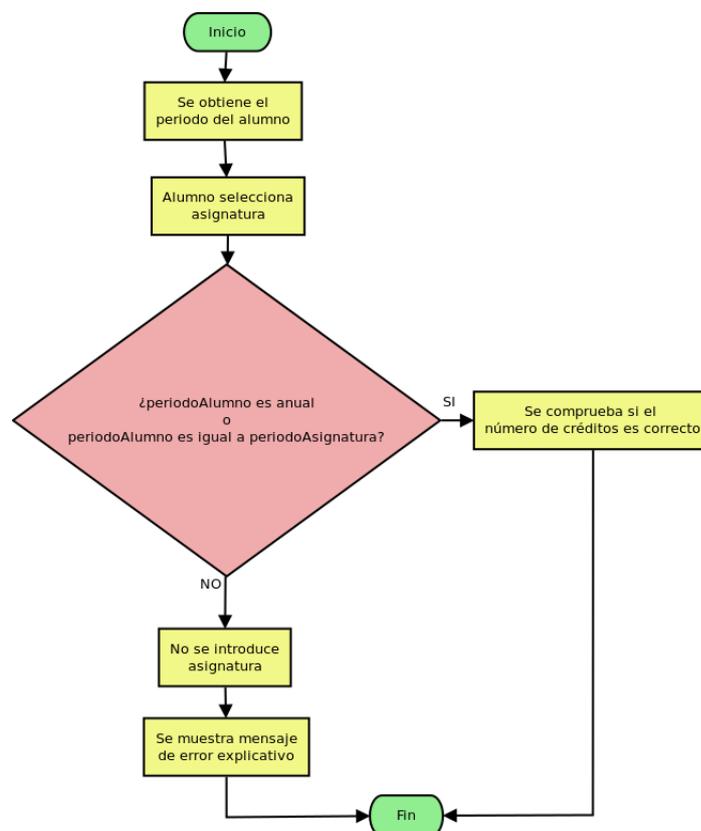


Figura 4.25: Algoritmo (Cuatrimestre correcto)

Cada periodo Erasmus tiene asignado un intervalo de créditos mínimo y máximo al que el alumno se tendrá que adaptar cuando quiere cursar su beca Erasmus. Si el alumno va a cursar una beca anual tendrá que elegir un mínimo de 30 créditos ECTS y un máximo de 66. Si va a cursar su beca Erasmus en un cuatrimestre el alumno deberá escoger entre 15 y 33 créditos ECTS.

Algoritmo 3: Número de créditos correcto

Se obtienen el máximo y el mínimo número de créditos ECTS que puede tener el alumno;

/ Beca anual: Mínimo créditos: 30 Máximo créditos: 66 */*

/ Beca primer o segundo cuatrimestre: Mínimo créditos: 15 Máximo créditos: 33 */*

Alumno selecciona asignatura;

/ Cada asignatura tiene unos créditos ECTS distintos */* **if** *creditosAlumno + creditosAsignatura* \geq *minCreditos* **then**

if *creditosAlumno + creditosAsignatura* \leq *maxCreditos* **then**

 Se introduce asignatura;

 Se muestra mensaje de confirmación;

 FIN;

else

 No se introduce asignatura;

 Se muestra mensaje de error explicando el motivo;

 FIN;

else

 No se introduce asignatura;

 Se muestra mensaje de error explicando el motivo;

 FIN;

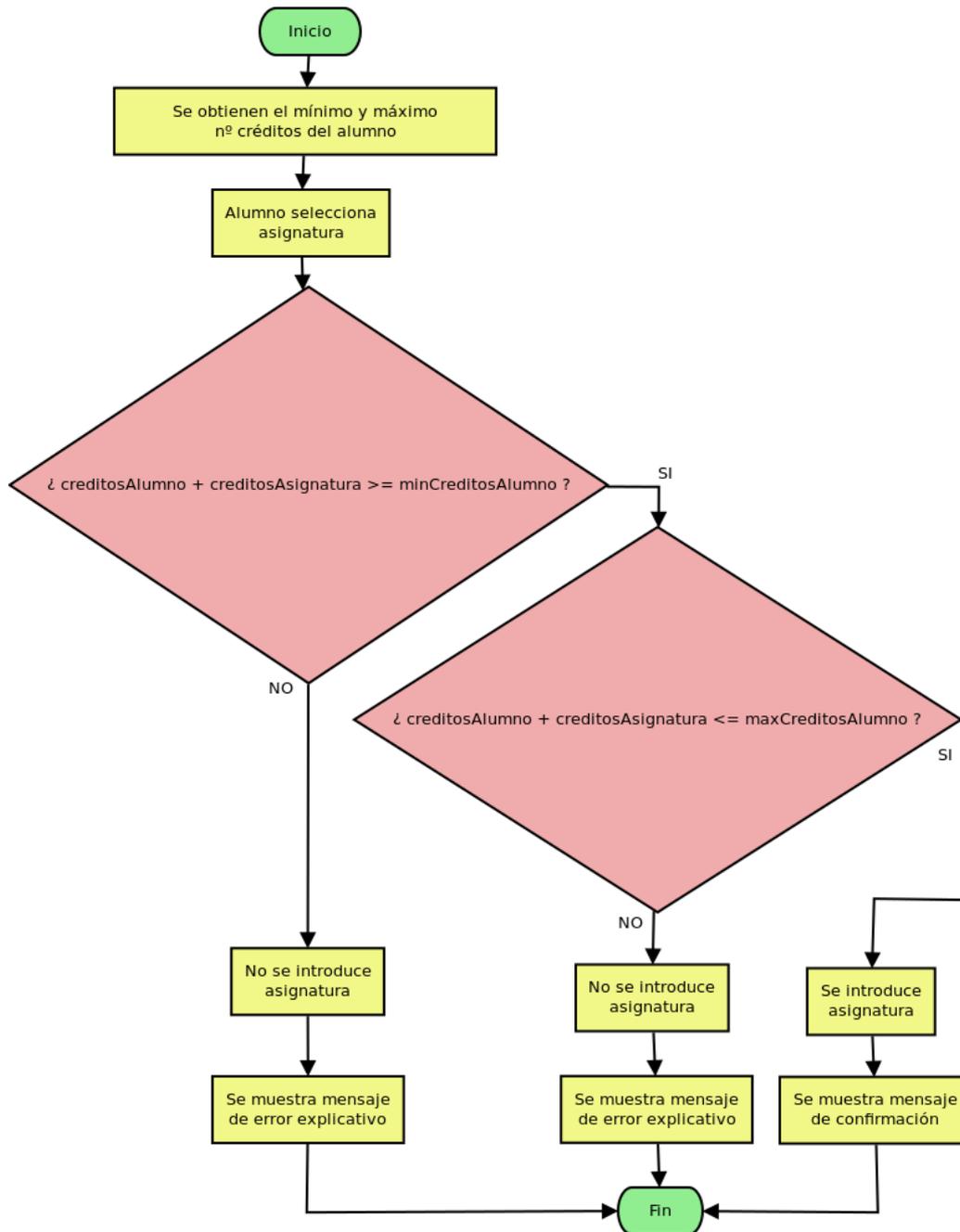


Figura 4.26: Algoritmo (Número de créditos correcto)

Un buen complemento para la herramienta sería un agente recomendador que aprendiera de todos estos conflictos que se pueden producir para en un futuro poder dar consejos o asesorar para que el proceso se pueda desarrollar con el menor número de errores.

4.10.2 Agente Recomendador

Un agente recomendador es una entidad de software persistente con un propósito específico, capaz de percibir el entorno y de actuar sobre él para alcanzar una serie de objetivos para los que fue diseñado.

Los agentes recomendadores tienen una serie de características comunes:

- Punto de vista incompleto, aunque muchas veces se busca una solución aproximada.
- Los agentes se mueven dentro de un entorno *virtual* operando a través de un sistema.
- Controlan el entorno. Suelen tener control parcial del entorno.
- Tienen capacidad de aprendizaje.
- Debe actuar autónomamente sin la intervención de seres humanos u otros sistemas y debe tener control sobre su estado interno y sobre su propio comportamiento.

El algoritmo utilizado por el agente recomendador está siempre a la escucha. Es un agente reactivo y consciente de lo que ocurre en el ambiente. En el momento que el estudiante esté realizando su contrato de estudios y se produzca un conflicto, este revisa las reglas disponibles y activa la que sea más adecuada para el problema en cuestión.

El algoritmo del agente recomendador tiene dos fases. Una primera en la que se produce el conflicto. Este conflicto es añadido a una base de conocimiento que es posteriormente utilizada por el agente recomendador para dar recomendaciones.

Esta primera fase se produce cuando un alumno no está conforme con alguna asignatura matriculada y se dispone a hacer una modificación del contrato. Deberá especificar el motivo de la modificación de contrato.

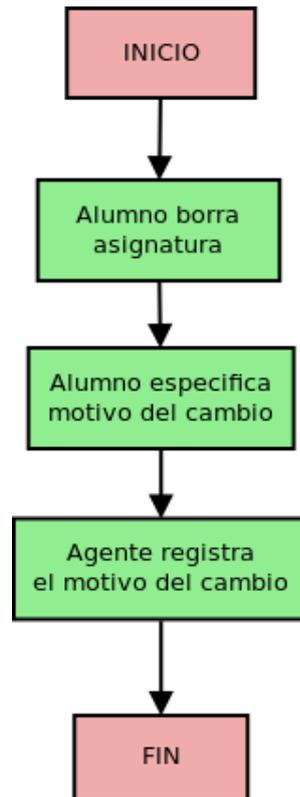


Figura 4.27: Fase de aprendizaje del algoritmo

La segunda fase comienza cuando un alumno se dispone a hacer su contrato de estudios. Este irá cogiendo asignaturas y si en algún momento se produce algún conflicto con alguna de las asignaturas elegidas el agente recomendador se dará cuenta y reaccionará haciendo una recomendación al alumno mostrándole por qué no debería coger esa asignatura.

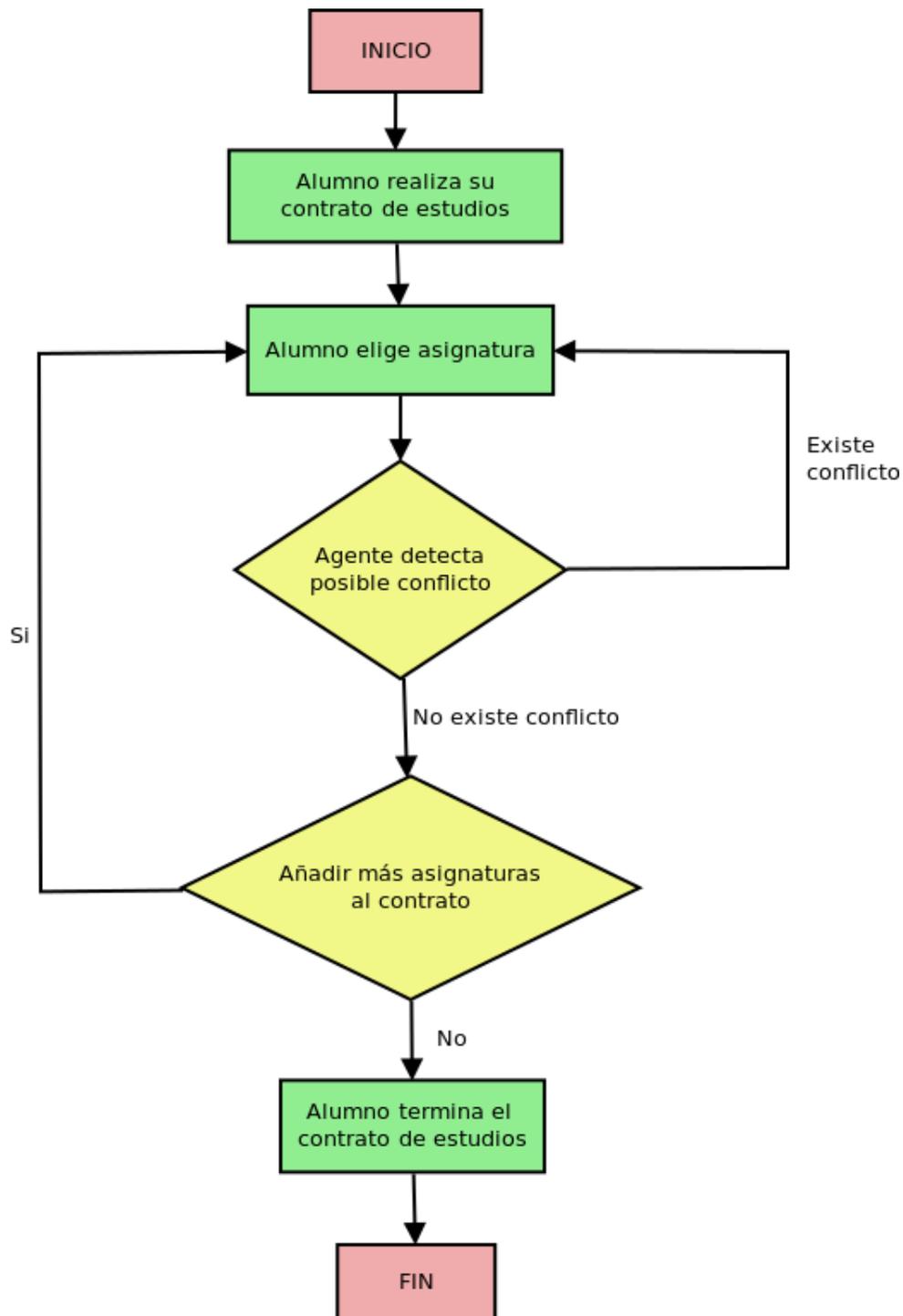


Figura 4.28: Fase de recomendación del algoritmo

El algoritmo utilizado está *basado en casos*. Los sistemas basados en casos basan su funcionamiento en experiencias pasadas. Basándose en la experiencia adquirida pueden alcanzar la solución. Un ejemplo de esto sería el de un niño que intenta coger una olla y se quema. Después de esto el niño habrá adquirido una nueva experiencia y no volverá a tocar la olla. Es posible que el niño vea posteriormente a su madre coger la olla con un paño para no quemarse. El niño estará entonces adquiriendo otra experiencia que añadirá a su base de conocimiento. Con toda esta experiencia analizada, la próxima vez que quiera coger una olla,

optará por la solución correcta y cogerá un paño antes de tocar la olla.

4.10.3 Estructura de la Base de Datos para el diseño del agente recomendador

Se ha utilizado una tabla en la base de datos para realizar el diseño del agente recomendador y su base de conocimiento. La tabla está formada por siete columnas que se explican a continuación:

- **idAlumno:** Este campo identifica al alumno por el cuál se está registrando el conflicto.
- **asigLocal:** Este valor indica la posición de la asignatura en la tabla de asociación de asignaturas de la uclm que se muestra en la web.
- **asigDestino:** Este valor indica la posición de la asignatura en la tabla de asociación de asignaturas de la universidad destino que se muestra en la web.
- **agregar:** Este campo puede tomar dos valores, 0 o 1. Tomará 1 cuando el cambio haya sido para agregar una asignatura. Tomará 0 cuando el cambio haya sido para borrar una asignatura.
- **motivo:** Se indica el motivo por el cuál ha surgido el cambio.
- **asignaturaLocal:** En este campo se refleja el código ects de la asignatura local.
- **asignaturaDestino:** Esta columna contiene el código ects de la asignatura destino.

idAlumno	asigLocal	asigDestino	agregar	motivo	asignaturaLocal	asignaturaDestino
1	1	1	1	El coord erasmus no acepta el cambio	42503	MMNM
2	1	1	1	Horario no compatible	42503	MMNM
2	1	1	0	El idioma era lituano	42505	KKKK

Figura 4.29: Tabla donde se reflejan los conflictos del sistema

Cuando un alumno está haciendo su contrato de estudios y añade alguna asignatura en la cuál podría existir algún conflicto se muestra una advertencia.

Asignaturas de la universidad de destino elegidas por el alumno.-

Número	Código ECTS	Asignatura	Créditos ECTS	Tipo	Cuatrimestre	Eliminar	Sugerencia
1	FS5324	Databases	5	FB	Primero	✗	
2	NV8872	Computer Architecture	5	FB	Primero	✗	
3	UR9865	Software Engineering	6	FB	Primero	✗	
4	KKKK	Information Business	6	FB	Primero	✗	

Figura 4.30: *Conflicto en la asignatura Information Business*

De esta forma el alumno puede darse cuenta de que a otro alumno le ocurrió algo con esta asignatura en el pasado. Bastaría con hacer clic en la imagen de advertencia para que apareciera un agente recomendador mostrando los conflictos de la asignatura.

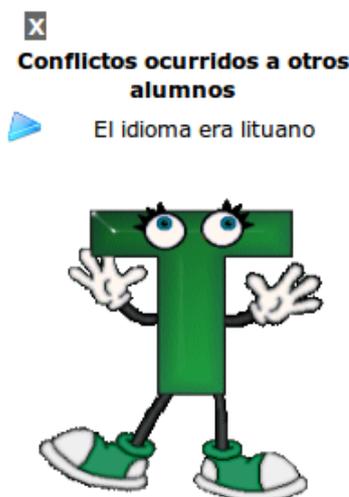


Figura 4.31: *Agente recomendador mostrando los conflictos de la asignatura seleccionada*

4.10.4 Diagrama de clases relacionadas con el aprendizaje de conflictos

A continuación se muestra un diagrama de clases donde aparecen las clases que tienen alguna relación con el aprendizaje de conflictos. Se pueden apreciar cuatro clases.

- **Usuario:** Clase que se compone de todos los atributos del usuario. El estudiante puede elegir asignaturas durante el periodo de realización del contrato de estudios y durante las modificaciones del mismo. La elección de ciertas asignaturas, tanto de la universidad de origen como de la universidad destino, pueden generar algún tipo de conflicto.
- **AsignaturasAlumno:** Esta clase contiene la información de la asociación entre asignaturas de la universidad origen y destino.

- **AsignaturaAsociada:** En esta clase se tienen los atributos que contienen la información de las asignaturas de la universidad destino.
- **AsignaturaLocal:** En esta clase se tienen los atributos que contienen la información de las asignaturas de la universidad origen.
- **Conflicto:** En esta clase se registran los cambios producidos en asignaturas y los motivos de dichos cambios. Esto ayudará posteriormente para que el sistema pueda mostrar recomendaciones y ayudas a la hora de realizar el contrato de estudios.

En el diagrama se puede ver como un alumno tiene una o varias asignaturas. Estas asignaturas pueden ser de la universidad origen y de la universidad destino. Cada asignatura de la universidad origen puede estar asociada a una o varias asignatura de la universidad destino y viceversa.

También se puede apreciar como cada asignatura del alumno puede tener uno o ningún conflicto.

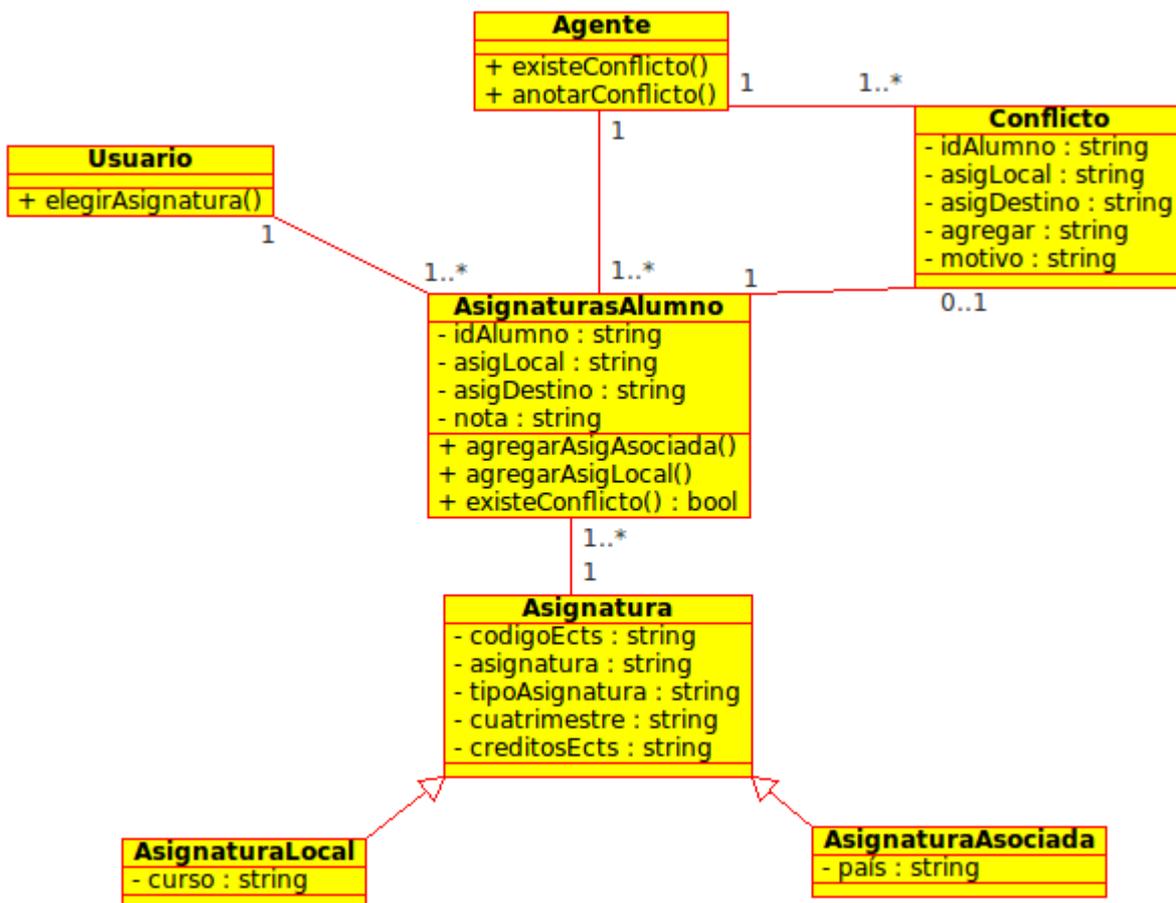


Figura 4.32: Diagrama de clases relacionadas con el aprendizaje de conflictos

Capítulo 5

Resultados

5.1 Resultados Obtenidos

El resultado es el diseño y desarrollo de una herramienta de gestión automática de los intercambios entre estudiantes Sócrates/Erasmus.

Con el objetivo de cumplir los objetivos descritos en el capítulo 2, se han desarrollado las siguientes funciones/herramientas:

- Se ha conseguido automatizar todas las tareas del proceso global de tal forma que todos los roles implicados pueden interaccionar fácilmente entre ellos agilizando todos los trámites que se tienen que llevar a cabo.

En todo el proceso Erasmus intervienen varios roles distintos y necesitan interactuar entre unos y otros para conseguir llevar a cabo con éxito la beca del estudiante. Existía un problema grande a la hora de que todos pudiesen comunicarse para elaborar sus tareas, sobre todo por problemas de localidad geográfica y/o temporal.

Teniendo en cuenta la naturaleza del problema se pensó que la mejor solución para abordar este problema sería realizar una aplicación web. De esta forma cualquiera de los roles puede utilizar la herramienta desde cualquier ordenador que contenga un navegador web y una conexión a internet. Así mismo no tiene importancia el sistema operativo que se utilice ya que la herramienta funcionará correctamente.

El sistema irá automatizando todas las tareas según avancen en el transcurso de la beca e irá informando a los roles pertinentes del estado y acontecimientos que van sucediendo según sea necesario. De esta manera cada rol no tiene que estar pendiente de la tarea que debe realizar. En el menú de la herramienta se mostrarán las distintas opciones que tiene disponible cada rol dependiendo del transcurso de la beca.

- La herramienta es capaz de generar contratos de estudios y actas de notas en formato PDF. Estos documentos pueden ser firmados digitalmente siendo igualmente válidos que los tradicionales en papel.

En una beca Erasmus, los dos documentos más importantes son el contrato de estudios y la ficha de reconocimiento de estudios. En el primero de ellos se especifican todos los datos referentes al alumno, datos personales, datos de los coordinadores en las universidades de origen y de destino, todas las asignaturas matriculadas en la universidad origen y sus convalidaciones en la universidad destino, todas las modificaciones de matrícula, etc. En el segundo documento se reflejan las notas que ha obtenido el alumno en el país extranjero y sus calificaciones definitivas en el sistema de calificación español.

La herramienta es capaz de generar estos documentos en formato PDF. De esta manera el alumno o coordinadores pueden obtenerlos utilizando una impresora. Para conseguir generar estos documentos en formato PDF se ha utilizado *iReport*. Es una herramienta que permite generar documentos en PDF a partir de formularios web, ya que estos suelen dar pobres resultados cuando son impresos. Java está provisto de una extensa API para imprimir, pero es un trabajo muy laborioso escribir código específico para el formato de cada documento. La respuesta a ese problema fue usar la librería de informes llamada *JasperReports*. Es la librería de código abierto más popular de Java en el mundo para generar informes. *iReport* es una herramienta con entorno visual que hace uso de esta librería. De esta manera se consiguieron obtener en formato PDF los documentos necesarios para la beca Erasmus.

Estos documentos necesitaban ser firmados por algunos roles implicados en el sistema. De poco valía tener los documentos en formato PDF, si posteriormente había que imprimirlos para llevarlos personalmente a que los firmasen los distintos coordinadores. Por lo que se buscó un método que permitiera firmar digitalmente estos documentos sin tener que imprimirlos. Se utilizaron para ello certificados digitales obtenidos por la FNMT-RCM (*Fábrica Nacional de Moneda y Timbre*), a través de su departamento CERES (*CERTificación ESpañola*). Hubo que integrar en el sistema la herramienta *JSigntool*, que permitía la firma de documentos digitales en formato PDF utilizando los certificados digitales citados anteriormente. Esta herramienta permite la múltiple firma de un mismo documento, cosa que era necesaria ya que el contrato de estudios debe ser firmado por varios roles. De esta manera se ha conseguido crear documentos digitales con igual validez que documentos originales en papel.

- La herramienta es capaz de adaptarse a cada tipo de usuario y sus privilegios mostrando las opciones que cada perfil necesite. Existe una identificación segura para entrar a la herramienta, la cuál se mostrará distinta de acuerdo al sistema de módulos y privilegios

asociados al rol que se ha diseñado.

Para tener acceso a la herramienta cada usuario ha tenido que ser dado de alta por el administrador. Este les proporciona unos datos de acceso a la herramienta. La contraseña de cada usuario quedará almacenada de manera encriptada en la base de datos. La encriptación se ha realizado utilizando un algoritmo SHA3.5.1.6.

Una vez el usuario entre en la herramienta se adaptará a cada tipo de usuario y a los privilegios que tenga en ese mismo momento. Esta misma interfaz podrá ir cambiando a medida que avanza el transcurso de la beca Erasmus mostrando unas opciones u otras. Por ejemplo, en un principio un alumno podrá ver en el menú la opción de *Realizar el contrato de estudios*. Sin embargo una vez el contrato de estudios ha sido finalizado por completo, esta opción desaparecerá mostrándose la opción de *Modificar el contrato de estudios*.

El sistema está dividido en módulos que se encargan de diferentes funcionalidades en el proceso de la beca Erasmus. Por cuestiones de seguridad no todos los módulos son accesibles por todos los roles del sistema. Por esta razón se ha implementado un sistema de permisos que permite el acceso de algunos usuarios a ciertos módulos del sistema. Estos permisos pueden ser modificados fácilmente por el administrador si fuese necesario por cambios en la normativa.

- Se ha dotado a la herramienta de capacidad de aprendizaje que le permite informar sobre criterios de la normativa Erasmus que no se están cumpliendo o incompatibilidades entre asignaturas de la universidad origen y destino. Se ha desarrollado un agente recomendador que se mantiene alerta para ayudar al estudiante a realizar correctamente su contrato de estudios asesorando sobre cualquier conflicto producido.

Uno de los problemas más comunes a la hora de realizar el contrato de estudios son los múltiples errores que se cometen en la realización del mismo. Algunas veces por desconocimiento de la normativa por parte del alumno o coordinadores, o simplemente por errores que no pueden ser conocidos a priori por estos.

Solucionar este problema era de vital importancia para ayudar a realizar contratos de estudios más eficientes y sin errores. También se pensó en aprovechar todos los errores y problemas que iban surgiendo en la elaboración de contratos y crear una herramienta con capacidad para aprender de estos errores. Esta experiencia adquirida por la herramienta a través del tiempo es ahora aprovechada por los nuevos usuarios en

la realización de sus contratos de estudios.

Se ha desarrollado un agente recomendador que se encuentra alerta cuando un estudiante está realizando su contrato de estudios. En el momento que este comete algún error o se produce algún conflicto aparece el agente recomendador y hace las sugerencias pertinentes para que el alumno pueda tomar la mejor decisión.

- Se ha desarrollado un sistema de control de eventos que permite a cualquier usuario darse cuenta de los cambios que se han producido en la herramienta desde la última vez que accedió a ella.

Este requisito era muy necesario para facilitar la tarea de los coordinadores. Ya que era una tarea difícil estar al tanto de la evolución de las becas Erasmus de cada uno de los estudiantes. El proceso de beca de cada estudiante irá avanzando a un ritmo distinto y era necesaria una manera que ayudase a los coordinadores a seguir este control.

Se pensó en crear un sistema de control de eventos donde se registrasen en la pantalla de inicio todos los acontecimientos ocurridos desde la última vez que se utilizó la herramienta. De esta forma los coordinadores pueden ser conscientes de los alumnos que han terminado su contrato de estudios y están pendientes de ser revisados, los alumnos pueden darse cuenta de si su contrato de estudios ha sido firmado o si tienen disponible la ficha de reconocimiento de estudios, etc.

En cada evento mostrado en la pantalla de inicio se mostrará la suficiente información para que el receptor pueda saber quien provocó tal evento, el motivo del evento y posibles acciones a realizar con él.

- Se permite el cálculo automático de calificaciones procedentes del país extranjero. Igualmente es posible la modificación de los baremos de calificación de cada país.

El sistema de calificación de las asignaturas en España no es igual al del resto de países. Existe una tabla de conversión de calificaciones europea donde se muestran los baremos de calificación para cada país que pertenece al convenio Erasmus. El problema que existía anteriormente era que el coordinador de centro tenía que convertir todas las actas de notas procedentes del país extranjero de todos los alumnos Erasmus que tuviera ese año.

Realizar todos esos cálculos para tantos alumnos no era una tarea trivial así que se ha implementado en la herramienta un módulo que convierte automáticamente las notas procedentes del país extranjero al sistema de calificación español.

Es posible que los baremos de calificación de algún país cambien con el tiempo, por esta razón se ha implementado también un módulo que permite modificar los sistemas de calificación de cada país. De esta manera la herramienta siempre estará actualizada y nunca quedará obsoleta en este sentido.

Una vez se han convertido las notas al sistema de calificación español se puede generar la ficha de reconocimiento de estudios donde quedarán reflejadas las calificaciones definitivas del alumno en su estancia Erasmus.

- El contrato de estudios en esta herramienta es un elemento muy importante. Desde el inicio de la beca Erasmus hasta casi su finalización el contrato de estudios es accedido y modificado por casi todos los roles del sistema. Debido a las características del mismo, este debe seguir un determinado proceso marcado por la normativa Erasmus. Por este motivo no es posible que cualquier rol pueda acceder a él en cualquier momento, ya que se podrían producir problemas de seguridad e incoherencias.

Se ha diseñado una máquina de estados para llevar el control de flujo del contrato de estudios y evitar de esta forma un control de acceso indeseado al mismo por parte de los roles. De esta manera cada contrato de estudios se encontrará en un determinado estado en cada momento y cambiará de un estado a otro cuando exista una transacción debida a un evento producido en el sistema. Estos eventos son generados por el sistema como reacción a acciones producidas por los usuarios al utilizar la herramienta. Los estados de la máquina se han diseñado de tal forma que el contrato de estudios seguirá el proceso lógico que debe seguir de acuerdo a la normativa Erasmus.

- Ha sido desarrollado con el lenguaje JSP y MySQL como sistema gestor de bases de datos. Se encuentra alojado en un servidor Apache Tomcat. El sistema puede ser utilizado en diferentes máquinas con diferentes sistemas operativos. Solo es necesario que dispongan de un navegador web y conexión a Internet.

5.2 Posibles Costes de Implantación del Sistema

El siguiente estudio servirá para analizar los costes que supondría poner el sistema en explotación teniendo en cuenta las tecnologías que se han utilizado para su desarrollo.

5.2.1 Costes Anuales de Hosting

En las siguientes tablas se muestra coste anual del hosting en diferentes proveedores para las plataformas GNU/Linux y Windows. Se especifica junto al nombre de cada proveedor, las tecnologías que proporcionan para el desarrollo de web dinámicas, el espacio que ofrece y el coste anual del servicio.

Fuente	Apache	Tomcat	PHP	JSP	MySQL	PostgreSQL	Espacio	Precio/año
arsys.es	X		X		X		400 MB	192.00 €
abserver.es	X		X		X		300 MB	200.00 €
ferca.com	X	X	X	X	X		100 MB	325.00 €
claranet.es	X		X		X		200 MB	447.75 €
piensasolutions.com	X		X		X		300 MB	95.40 €
acens.com	X		X		X		225 MB	480.00 €
oxxus.net	X	X	X	X	X	X	500 MB	360.00 €
javaservlethosting.com	X	X	X	X	X		150 MB	320.00 €
visionwebhosting.com	X	X	X	X	X		1.2 GB	156.00 €
discountasp.net	X		X		X			Necesario Consultar
evidalia.com	X		X		X		400 MB	168.00 €
hosting.com	X		X		X		1 GB	420.00 €
interadventure.com	X	X	X	X	X	X	1.5 GB	180.00 €
mochahost.com	X	X	X	X	X		1 GB	120.00 €
weblinet.com	X	X	X	X	X	X	500 MB	420.00 €
winwebhosting.com	X		X		X		500 MB	95.00 €
mypagehost.com	X	X	X	X	X	X	2 GB	79.00 €
javapipe.com	X	X	X	X	X	X	500 MB	360.00 €
dhapcenter.es	X		X	X	X		300 MB	44.00 €

Tabla 5.1: Costes de Hosting en plataforma GNU/Linux

Fuente	Win 2003 Server	ODBC	ISS	MySQL	PHP	JSP	ASP.NET	ASP	Access	Postgre	Espacio	Precio/año
arsys.es	X										600 MB	348.00 €
ferca.com	X	X	X	X	X	X	X	X			450 MB	395.00 €
acens.com	X	X		X			X	X		X	225 MB	480.00 €
discountasp.net		X	X	X	X		X	X	X			
generalhosting	X				X		X	X	X			200.00 €
bizhostnet.com	X				X		X	X	X		20 GB	100.00 €
activehost.com	X				X		X	X	X		200 MB	300.00 €
hosting.com	X			X	X		X	X	X		1 GB	420.00 €
interadventure.com	X	X	X	X	X	X	X	X			1GB	480.00 €
winwebhosting.com	X		X	X	X	X	X	X			500 MB	130.00 €
dhapcenter.es	X				X		X	X	X		600 MB	44.00 €

Tabla 5.2: Costes de Hosting en plataforma Windows

De estas dos tablas es posible extraer las siguientes conclusiones:

- Los lugares de hosting donde es posible elegir entre servidores GNU/Linux y Windows, los precios suelen ser más bajos en el caso de servidores GNU/Linux, no pudiendo alojar aplicaciones desarrolladas en ASP o ASP.NET.
- Media aritmética de los costes en servidores GNU/Linux: 247,89 €
- Media aritmética de los costes en servidores Windows: 289,7 €
- Existe más hosting que soporta JSP en plataforma GNU/Linux que en plataforma Windows.
- No todos los servidores permiten alojar aplicaciones realizadas en Java y JSP, por lo que sería una opción interesante revisar cuanto costaría el alquiler o la compra de un servidor para poder utilizar dicha tecnología.

5.2.2 Costes de Alquiler de un Servidor

Según las conclusiones extraídas de la sección anterior no existían muchos proveedores de hosting que proporcionaran alquiler servidores para poder alojar aplicaciones realizadas con JSP. Una opción interesante sería la de alquilar un servidor, ya que esta opción te da más libertad a la hora de utilizar otras tecnologías no disponibles con el alquiler de hosting.

A continuación se muestra una tabla donde se indican algunos sitios de la red en los que es posible el alquiler de servidores, las características del mismo, y su coste si se trata de una plataforma GNU/Linux o Windows.

Fuente	Servidor	GNU/Linux	Windows
centrodedatos.com	P4 3 Ghz, 1 GB RAM, 160 GB HD	1428 €/año	1428 €/año
serveisweb.com	P4 2.8 Ghz, 2 GB RAM, 80 GB HD	653 €/año	763 €/año
ram.es	P4 2.8 Ghz, 512 MB RAM, 120 GB HD	588 €/año	1188 €/año
ferca.com	Intel Celeron 2.4 Ghz, 512 RAM, 80 GB HD	1296 €/año	1621 €/año
arsys.es	P4 2.8 Ghz, 512 RAM, 80 GB HD	1188 €/año	1188 €/año
evidaliahost.com	Pentium Celeron 2 Ghz, 1 GB RAM, 160 GB HD	2340 €/año	No posee máquinas con Windows
simbionet.com	P3 733 Mhz, 256 RAM, 10 GB HD	420 €/año	No posee máquinas con Windows
acens.com	P4 3.2 Ghz, 512 RAM, 80 GB HD	3000 €/año	3588 €/año
syncs.es	P4 2.8 Ghz, 512 RAM, 80 GB HD	2148 €/año	2688 €/año

Tabla 5.3: Comparativa de alquiler de servidores en las plataformas GNU/Linux y Windows

5.2.3 Costes de Adquisición de un Servidor

Si la herramienta fuera a estar en explotación durante mucho tiempo se podría estudiar la posibilidad de adquirir un servidor en vez de alquilarlo como se mostraba en la sección anterior.

A continuación se muestra una de servidores especificando las características y precios de los mismos. A parte del precio del servidor habría que tener en cuenta el coste que supondría la conexión a internet y el de la energía eléctrica.

Modelo	Procesador	Memoria RAM	Espacio	Precio
Power Edge SC430	P4 2.8 Ghz	512 MB RAM	80 GB	359 €
HP Proliant ML350	P4 2.8 Ghz	512 MB RAM	80 GB	789 €
Xseries 206	P4 3.2 Ghz	256 MB RAM	60 GB	600 €
Tyan Computer. Copr	AMD Opt. 2.4 Ghz	512 RAM	160 GB	879.55 €
HP Proliant	P4 2.8 Ghz	256 MB RAM	80 GB	496.44 €
HP Compact Proliant DL 380	P3 1 Ghz	128 MB RAM	60 GB	224 €
IBM NetFinity 5500	P3 500 Mhz	256 MB RAM	60 GB	602,95 €
Dell PowerEdge 4400	P3 953 Mhz	512 MB RAM	80 GB	467,50 €
IBM X Series X235 Xeon	P4 2.4 Ghz	256 MB RAM	80 GB	767,00 €
Monarch Computer	AMD Opt. 1.4 Ghz	512 RAM	80 GB	1153,00 €

Tabla 5.4: Comparativa de alquiler de servidores en las plataformas GNU/Linux y Windows

5.2.4 Decisiones de las tecnologías utilizadas

Una de las decisiones que era necesario tomar al comenzar a desarrollar la herramienta era la de escoger el lenguaje de programación adecuado. Teniendo en cuenta la naturaleza del problema era necesario elegir un lenguaje de programación que permitiera la construcción de sitios web dinámicos. Para un mejor diseño y estructuración del problema era necesario que permitiese una forma fácil de realizar un diseño multicapa, pudiendo separar las capas de presentación, dominio y persistencia.

Las opciones posibles eran las siguientes:

- PHP4
- PHP5

- ASP
- ASP.NET
- JSP

Teniendo en cuenta el análisis de costes realizado anteriormente ASP.NET quedó descartado debido al alto coste del alquiler de hosting. ASP tenía el inconveniente que la separación entre capas no era muy trivial.

La opción de hacerlo en PHP estuvo presente al comienzo del proyecto, sobre todo la de utilizar PHP5 que ofrece grandes ventajas sobre PHP4. Está altamente orientado a objetos y permite una clara separación entre capas.

JSP era la otra opción que faltaba por analizar. La principal ventaja de JSP frente a otros lenguajes es que permite integrarse con clases Java, lo que permite separar en niveles las aplicaciones web, almacenando en clases java las partes que consumen más recursos así como las que requieren más seguridad, y dejando la parte encargada de formatear el documento html en el archivo JSP. Este lenguaje es muy simple de usar si el desarrollador posee experiencia en Java, pues JSP utiliza la misma sintaxis de Java y da la facilidad de utilizar las mismas librerías y extensiones que utilizarían para crear aplicaciones de escritorio en el Java tradicional. Otra de las ventajas más importantes es que también es libre. Cualquiera puede desarrollar en este lenguaje sin necesidad de pagar un software adicional para desarrollar aplicaciones JSP.

Aunque la razón con más peso que llevó a elegir JSP como lenguaje para desarrollar este proyecto fue el alto conocimiento que ya poseía del mismo, ya que he participado en otros proyectos laborales utilizando esta tecnología.

Finalmente teniendo en cuenta lo anteriormente explicado se optó por elegir el lenguaje JSP para desarrollar el proyecto.

Estas eran las opciones disponibles como servidor web para JSP:

- **Apache Tomcat:** Es un buen servidor JSP y uno de los más populares. Tomcat es un software de código libre y los usuarios disponen de acceso libre a su código fuente y a su forma binaria en los términos establecido en la *Apache Software License*.
- **JSWDK:** Es un servidor sencillo y básico para probar aplicaciones JSP.
- **Sun Java System Web Server:** Es un servidor web de Sun Microsystems, por supuesto, pensado para ejecutar JSP y Servlets, pero también con soporte para ASP, PHP o CGI. Es comercial, pero tiene una opción de descarga para probarlo.

- **Macromedia JRun:** Es un servidor de Java desarrollado por Macromedia (ahora Adobe). Es comercial, aunque se puede obtener una versión de prueba.
- **ServletExec:** Aplicación comercial que sirve como módulo para ejecutar páginas JSP en servidores web populares, como Apache o IIS. Es un producto también comercial.

Al final opté por utilizar *Apache Tomcat* porque era un servidor que podía dar un buen rendimiento con esta herramienta y con las futuras actualizaciones que pudiera tener la misma. Otro de los motivos ha sido el conocimiento previo que existía con anterioridad, ya que ha permitido la configuración del mismo de manera trivial.

El sistema gestor de base de datos utilizado ha sido MySQL. Aparte de las múltiples ventajas de este SGBD 3.3.2.3, se añade que la mayoría de los servidores proporcionan soporte para MySQL. Además JSP proporciona mecanismos de comunicación sencilla y eficiente entre las aplicaciones y bases de datos MySQL.

Teniendo en cuenta todo lo explicado anteriormente, la posibilidad de poner este proyecto en explotación no supondría un alto coste, ya que tan sólo sería necesario el alquiler del hosting.

5.2.5 Mecanismos de financiación del sistema

Para costear la posible inversión de alquilar o comprar un servidor, comprar hosting, etc. se podría pensar en poner publicidad en la herramienta. Aunque esta opción no es viable al ser una aplicación para la UCLM.

Teniendo en cuenta el tiempo de desarrollo de la herramienta que han sido unos 10 meses, se puede establecer el coste aproximado de la misma si hubiera sido desarrollada por diferentes categorías de programadores.

Esto es una estimación media de los sueldos de un informático de diferentes categorías dentro de una empresa. La estimación ha sido hecha por InfoJobs ¹.

- Programador júnior en multinacional ->13.000 - 17.000 €/año
- Programador senior ->23.000 - 29.000 €/año
- Analista con programadores a cargo ->36.000 €/año

¹InfoJobs.net es una bolsa de empleo privada online, especializada en el mercado español.

Teniendo en cuenta que un trabajador emplease la jornada completa de su trabajo en el desarrollo de la herramienta se estima que la duración en terminarla hubiera sido entre 3 y 4 meses, dependiendo de la experiencia del programador. Por este motivo los costes estimados de la herramienta serían los siguientes:

- Programador júnior en multinacional ->4.300 - 5.600 €
- Programador senior ->5.900 - 6.300 €
- Analista con programadores a cargo ->6.100 - 7800 €

En este caso particular el desarrollo de esta herramienta es para la UCLM, por lo que se encargaría de su desarrollo final el centro de cálculo. Este departamento ya cuenta con su propio grupo de programadores y diseñadores software. Este proyecto fin de carrera serviría como una gran base y supondría un gran ahorro para la universidad a la hora de poner esta herramienta en explotación ya que no tendría que partir de cero a la hora de desarrollarla.

Otra de las opciones para rentabilizar el coste del desarrollo de la herramienta sería estudiar la posibilidad de vender el código fuente a otras universidades que al igual que la UCLM no tuvieran una herramienta para la gestión automática de sus alumnos Erasmus.

También se podría desarrollar nuevas funcionalidades para la misma a petición de las entidades que hayan adquirido el paquete de código fuente.

Conclusiones y Propuestas

6.1 Conclusiones

El camino que debe seguir un alumno cuando se le ha concedido una beca Erasmus es realmente tedioso, implica un largo proceso y actualmente no existen herramientas que faciliten su gestión. Debido a esto se plantearon una serie de objetivos con la finalidad de mejorar el proceso actual de realización de una beca Erasmus y ayudar a todos los roles implicados en ella.

Finalmente se ha conseguido el desarrollo de un sistema de información que gestiona de forma automática los intercambios entre estudiantes Erasmus.

Este tipo de sistemas requiere la participación de usuarios con diferentes responsabilidades por lo que se pensó en desarrollar una herramienta capaz de adaptarse a cada usuario y los privilegios que estos pudiesen tener. Esto se ha conseguido haciendo que la herramienta fuera diferente en función del rol que accediera a ella. De esta forma cada rol puede trabajar con las opciones que tiene disponibles sin interferir en las tareas de otros roles.

Dada la naturaleza de la aplicación era una condición imprescindible poder desarrollar una herramienta que fuera de fácil acceso desde múltiples lugares. Por eso la solución más adecuada era el desarrollo de un sistema web que es accesible desde cualquier parte del mundo donde exista un ordenador con navegador web y una conexión a internet.

El desarrollo de la herramienta se ha estructurado en tres capas permitiendo un desarrollo más satisfactorio entre otras múltiples ventajas. También se ha pensado en el mantenimiento futuro de la herramienta ya que los componentes han sido desarrollados de manera independiente de tal forma que cambios futuros en la normativa no afecten al funcionamiento global del sistema.

Se ha conseguido automatizar las tareas del proceso global facilitando la interacción

entre todas las personas que intervienen en el proceso del alumno Erasmus. Para ello se ha desarrollado una herramienta capaz de generar digitalmente en formato PDF el contrato de estudios del estudiante y la ficha de reconocimiento de estudios. Estos son los documentos más importantes que existen en una beca Erasmus. Estos elementos serán revisados por los coordinadores y el estudiante durante todo el periodo que dure la beca. Se debe brindar la posibilidad de que esta interacción se produzca de la manera más ágil y sencilla posible. Debido a esto se creó un sistema de control de eventos que permite a todos los usuarios tener conocimiento de los cambios que se han producido en la herramienta desde la última vez que accedieron a ella. De esta forma cualquier coordinador puede saber los cambios que han sufrido los contratos de estudios de sus alumnos con tan solo acceder a la herramienta.

Tanto el contrato de estudios como la ficha de reconocimiento de estudios requieren ser firmados por los distintos coordinadores y el estudiante para que puedan tener validez. Se pretendía aplicar TICs a todo el proceso de la beca Erasmus de un alumno, por lo que se pensó en una manera de realizar firmas digitales para firmar estos documentos. Se han utilizado certificados digitales que permiten identificar de manera íntegra y segura la persona que está firmando el documento. El resultado son unos documentos electrónicos firmados digitalmente por los coordinadores pertinentes evitando así los problemas que pueden surgir para que estos sean firmados. Al estar firmados digitalmente estos documentos tienen la misma validez que un contrato en papel.

El contrato de estudios del estudiante puede ser modificado y cambiado muchas veces por los coordinadores o el alumno durante el transcurso de la beca Erasmus. Por este motivo se ha desarrollado una máquina de estados para llevar el control de flujo del contrato de estudios. Es un elemento muy importante y no se puede permitir el acceso incontrolado al mismo. De esta manera se evitan incoherencias al ser accedido y modificado por varios usuarios a la vez y no existen problemas de seguridad.

Uno de los problemas que ralentizaban más el desarrollo de un contrato de estudios eran los múltiples conflictos y problemas que podían surgir debido a varios motivos como el desconocimiento de la normativa por parte del alumno o por problemas externos con las asignaturas de la universidad de destino. A causa de estos problemas se ha desarrollado una herramienta con capacidad de aprendizaje para resolver conflictos que se hayan producido con anterioridad. El sistema irá aprendiendo conflictos que le hayan ido ocurriendo a los alumnos en el transcurso de sus becas. De esta forma la herramienta puede asesorar a alumnos futuros para que puedan desarrollar su contrato de estudios de manera más eficiente. El sistema cuenta con un agente recomendador que se encuentra alerta para ayudar al estudiante a resolver un conflicto que se haya producido en la realización de su contrato de estudios. Cuando se produce algún conflicto el agente recomendador aparece mostrando los motivos por los cuáles

otros alumnos tuvieron problemas.

Para generar la ficha de reconocimiento de estudios el coordinador de centro debía realizar múltiples cálculos para transformar las calificaciones traídas de la universidad de destino por el alumno al sistema de calificación español. Se ha desarrollado un módulo que transforma automáticamente las notas teniendo en cuenta el país de procedencia del alumno. Después de esto ya es posible generar la ficha de reconocimiento de estudios en formato PDF donde aparecen las notas definitivas que ha obtenido el alumno en su beca Erasmus.

6.2 Propuestas y Líneas Futuras

Son varias las mejoras que se podrían realizar en el sistema. Con el diseño realizado, estas mejoras no supondrían grandes cambios en el código fuente. A continuación pasaremos a analizar algunas de ellas.

Las principales mejoras que se podrían aplicar estarían enfocadas a mejorar la interacción con la universidades de los países de destino de la siguiente forma:

- Disponer de los planes de estudios de las universidades asociadas. De esta forma se podrían realizar contratos de estudios más fiables y de esta forma evitar la mayoría de conflictos que se producen.
- Añadir varios roles al sistema como por ejemplo el de *Coordinador Extranjero*.
- Permitir la firma electrónica de los documentos a los coordinadores de las universidades extranjeras.
- Permitir la conversión de la interfaz de la herramienta en múltiples idiomas para que pudiera ser utilizada sin problema en cualquier otro país.
- Apartado de estadísticas personalizadas para cada rol. Los coordinadores podrían ver el porcentaje de alumnos aprobados cada año y las notas que obtienen dependiendo del país donde han cursado la beca Erasmus. Los alumnos podrían ver estadísticas donde se muestren los porcentajes de estudiantes de años anteriores que cursaron ciertas asignaturas y las notas que obtuvieron. También podrían ver los países donde se obtienen mejores calificaciones, lugares donde es más frecuente encontrar más estudiantes españoles, etc.
- Foro donde los alumnos que vayan a cursar la beca Erasmus puedan compartir opiniones, dudas y cualquier otra cosa que puedan necesitar.
- En algunos países las universidades asociadas tienen convenios con residencias de estudiantes donde se alojan los estudiantes Erasmus cuando van a estudiar allí. Se podría añadir otra nueva sección donde aparecieran los alojamientos disponibles para cada universidad, mostrando una descripción del lugar, fotos del mismo, etc. También podrían aparecer comentarios de alumnos anteriores que hayan utilizado ese alojamiento y puedan valorar que tal fue su estancia allí. Finalmente sería posible hacer una petición de reserva y poder gestionar el alquiler final.
- Se podría añadir al sistema el rol de *Administrador de Residencia*. Se permitiría de este modo gestionar las reservas de los alumnos Erasmus que desean obtener plaza en estos alojamientos.

- También sería útil añadir un chat que permitiera la comunicación sincrónica entre los distintos roles que participen en el proceso Erasmus del alumno. Cualquier rol al acceder a la herramienta podría ver los roles que están disponibles en el chat y poder discutir de manera sincrónica cualquier problema de manera más ágil.
- Ultimamente es cada vez más frecuente el acceso a Internet utilizando un teléfono móvil así que se podría mejorar la usabilidad de la herramienta haciendo que fuera accesible mediante estos dispositivos.

Analizando las opciones de mejora expuestas anteriormente, cabe destacar que existen muchas posibilidades de expansión y mejora pudiendo llegar a ser una herramienta muy útil, completa y que ayudaría enormemente a todas las personas que se implican en el proceso de una beca Erasmus

Apéndice A

Manual de Usuario.-

En este manual de usuario se va a explicar el funcionamiento de la herramienta mostrando las funciones más comunes que pueden desarrollar cada uno de los roles. Como la herramienta puede cambiar dependiendo del rol que la utilice, se va a dividir el manual en varias partes. Cada parte atenderá a uno de los roles existentes.

Una parte común a cualquier rol es la pantalla de autenticación. Cualquier usuario que desee utilizar la herramienta debe usar sus datos de acceso. Cada usuario que haya sido registrado por el rol administrador, deberá tener un nombre de usuario y una contraseña. Una vez se introduzcan los datos de acceso el usuario accederá a su pantalla de inicio.

La pantalla de inicio ofrecerá distintas opciones dependiendo del tipo de usuario que haya accedido a ella.

- **Rol Estudiante.-** Cuando los estudiantes que van a cursar una beca Erasmus accedan a la herramienta podrán encontrar las siguientes opciones:

Módulos disponibles.-



Figura A.1: *Módulos de un estudiante en la pantalla de inicio*

- Realizar contrato de estudios. Este es el módulo donde los alumnos podrán realizar su contrato de estudios. A continuación se explicará cada uno de los pasos a realizar para confeccionar el contrato de estudios. Para poder avanzar a cada uno de los pasos aparecerán en la parte superior unos círculos de color azul que indican los pasos existentes y el paso actual del proceso donde nos encontramos. Para ir de un paso a otro bastará con hacer clic en cada uno de los círculos.



Figura A.2: *Guía visual de pasos para realizar el contrato de estudios*

- * Bienvenida. En esta pantalla se muestra el mensaje de bienvenida al usuario. Aparece un pequeño manual que orienta al alumno sobre el funcionamiento del proceso de realización del contrato de estudios.

- * Datos personales del estudiante. En esta sección se encuentran los datos personales del alumno. No hay que rellenar nada. Los datos que aparecen automáticamente fueron dados al administrador cuando se realizó el registro del usuario. Si hay que modificar algún dato porque exista algún error solo hay que escribirlo y pulsar sobre el botón *Introducir datos*.

- * Universidad asociada. En la siguiente pantalla aparece un formulario para rellenar datos sobre la universidad de destino. También hay que rellenar datos sobre el coordinador institucional y el profesor de movilidad asignados en la universidad destino.

- * Universidad de Castilla-La Mancha. Este apartado es similar al anterior pero con datos de la Universidad de Castilla-La Mancha. También hay que rellenar datos sobre el profesor de movilidad, coordinador de centro y coordinador institucional.

- * Aptitudes Lingüísticas, Estudios y Periodo. En este apartado habrá que rellenar datos sobre las aptitudes y conocimientos del alumno. Conocimiento del idioma que se exigirá en la universidad de destino, estudios realizados hasta el momento, duración de la beca Erasmus, etc.

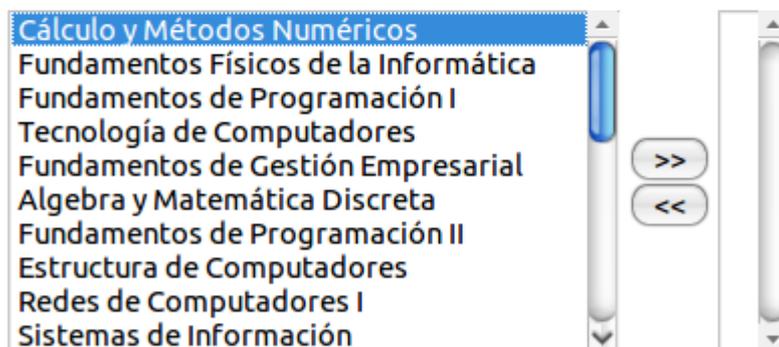
- * Asignaturas Universidad Origen. En esta pantalla se podrán elegir las asignaturas de la UCLM que van a ser cambiadas por las asignaturas de la universidad del país extranjero. Aparece una lista en la que se muestran los nombres de todas las asignaturas disponibles para el alumno. Al hacer clic en alguna de ellas se mostrará información detallada de la asignatura.

Información de la asignatura seleccionada.-

Asignatura:	Cálculo y Métodos Numéricos
Código Ects:	42300
Tipo de asignatura:	FB
Cuatrimestre:	Primero
Créditos Ects:	6
Curso:	1

Asignaturas Universidad de Castilla-La Mancha.-

Seleccione las asignaturas de la UCLM que desee añadir al contrato de estudios.



Introducir Asignaturas

Figura A.3: Información de la asignatura seleccionada de la lista de asignaturas

Para elegir una o más asignaturas bastará con seleccionarlas en la lista de la izquierda y pulsar el botón >> para pasar las asignaturas a la lista de la derecha. Una vez tenemos en la lista de la derecha todas las asignaturas que queremos cursar se pulsa el botón *Introducir asignaturas* para añadirlas al contrato de estudios.

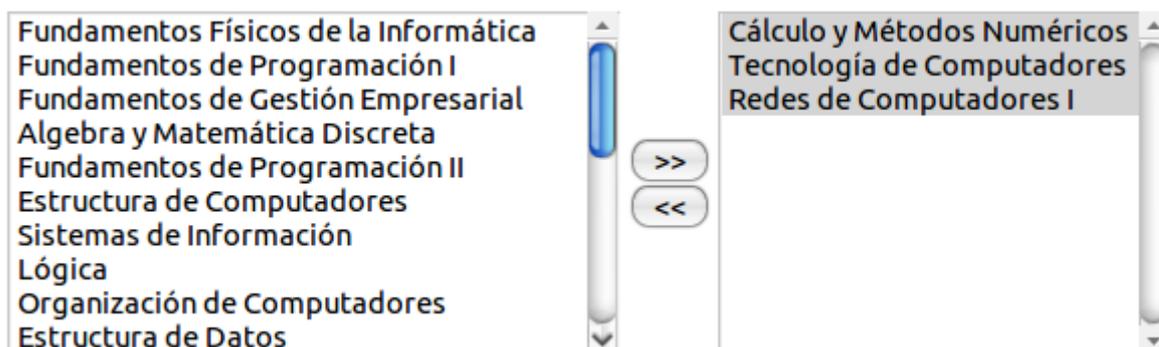


Figura A.4: Formulario para elegir asignaturas en la UCLM

Más abajo se mostrará una tabla con las asignaturas que se tienen en el contrato de estudios. El alumno deberá introducir la convocatoria correcta de la asignatura y pulsar el icono de actualizar para que esta información quede guardada. Si el alumno desea borrar alguna de las asignaturas introducidas solo deberá pulsar el icono de eliminar y la asignatura quedará borrada del contrato de estudios.

Código ECTS	Asignatura	Créditos ECTS	Tipo	Cuatrimestre	Curso	Convocatoria	Eliminar
42405	Estadística	6	FB	Segundo	2	<input type="checkbox"/> 	
42406	Metodología de la Programación	6	B	Segundo	2	<input type="checkbox"/> 	

Figura A.5: Tabla que muestra las asignaturas que hay en el contrato de estudios

- * Asignaturas Universidad Destino. Esta sección es para añadir al contrato de estudios las asignaturas de la universidad destino. En primer lugar aparecerá una tabla resumen con las asignaturas de la UCLM que el alumno ha elegido con anterioridad. Esta tabla tiene un campo llamado *numero* que sirve para realizar la asociación entre asignaturas de la UCLM y asignaturas extranjeras.

A continuación se encuentra el formulario para añadir asignaturas. Primeramente aparece una lista desplegable en la que se debe elegir un número haciendolo corresponder con alguna de las asignaturas mostradas arriba. Después solo queda introducir los datos de la asignatura destino y hacer clic en el botón *Introducir asignatura*.

Asignaturas de la UCLM elegidas por el alumno.-

Código ECTS	Número	Asignatura	Créditos ECTS	Tipo	Cuatrimestre	Curso	Convocatoria
42405	1	Estadística	6	FB	Segundo	2	
42406	2	Metodología de la Programación	6	B	Segundo	2	

Introduzca la asignatura destino.-

Número: Asocia la asignatura elegida con la de la Uclm

Código ECTS:

Tipo de Asignatura:

Cuatrimestre:

Créditos ECTS:

Asignatura:

Figura A.6: Asociación de asignaturas de las universidades origen y destino

Después de esto se puede observar como aparece una tabla donde se muestran las asignaturas elegidas de la universidad destino. Si se quisiera eliminar alguna bastaría con pulsar el icono de eliminar y quedaría eliminada del contrato.

* Firmar y Generar Contrato en Pdf. Esta opción en un principio está deshabilitada y no aparece en la barra de pasos del proceso, ya que no es posible generar ningún contrato de estudios si previamente no se ha realizado alguna asociación de asignaturas. En esta pantalla hay que hacer clic en el botón *Generar contrato*. Entonces aparecerá el formulario para que el alumno firme el contrato de estudios. Tendrá que rellenar los siguientes datos:

- Tipo de certificado: Se debe seleccionar de una lista desplegable el tipo de certificado que se va a usar para firmar. Actualmente solo está disponible el certificado de tipo *PKCS12*.
- Certificado: Aquí se tiene que seleccionar el archivo del certificado en el directorio local donde se encuentre.
- Clave: Se especifica la clave que se nos proporcione con el certificado.

Firmar Contrato.-

Tipo de Certificado:

Certificado:

Clave:

Figura A.7: *Formulario para que el alumno firme el contrato de estudios*

Una vez completado el formulario bastará con pulsar el boton *Firmar* y el contrato de estudios estará firmado.

Se habrá generado entonces el contrato de estudios en formato pdf y estará disponible para su visualización por el alumno y el profesor de movilidad.

- Visualizar contrato. Al hacer clic en este enlace el alumno podrá visualizar su contrato en formato pdf. Para que esto sea posible antes debe haber sido generado previamente utilizando el enlace *Realizar contrato de estudios*.
- Visualizar notas. Mediante este enlace los alumnos podrán acceder a su ficha de reconocimiento de estudios para ver sus notas definitivas. Para poder visualizar el documento antes debe haber sido generado por el coordinador de centro.
- Modificar contrato de estudios. Este modulo no estará disponible la primera vez que el alumno entre en la herramienta. Solo podrá visualizarlo una vez ha generado el contrato de estudios por completo la primera vez. En ese caso el menu *Realizar contrato de estudios* desaparecerá y en su lugar estará disponible este. Cualquier modificación que haya que realizar al contrato habrá que hacerla a traves de este enlace.

A continuación se explicará cada uno de los pasos a realizar para realizar una modificación en el contrato de estudios. Para poder avanzar a cada uno de los pasos aparecerán en la parte superior unos círculos de color azul que indican los pasos existentes y el paso actual del proceso donde nos encontramos. Para ir de un paso a otro bastará con hacer clic en cada uno de los círculos.



Figura A.8: *Guía visual de pasos para realizar modificaciones en el contrato de estudios*

- * **Bienvenida.** En esta pantalla se muestra el mensaje de bienvenida al usuario. Aparece un pequeño manual que orienta al alumno sobre el funcionamiento del proceso de modificación del contrato de estudios.
- * **Modificar asignaturas UCLM.** En esta pantalla se podrán elegir las asignaturas de la UCLM que van a ser cambiadas por las asignaturas de la universidad del país extranjero. Aparece una lista en la que se muestran los nombres de todas las asignaturas disponibles para el alumno. Al hacer clic en alguna de ellas se mostrará información detallada de la asignatura. Para elegir una o más asignaturas bastará con seleccionárselas en la lista de la izquierda y pulsar el botón >> para pasar las asignaturas a la lista de la derecha. Una vez tenemos en la lista de la derecha todas las asignaturas que queremos cursar se pulsa el botón *Introducir asignaturas* para añadirlas al contrato de estudios.

Más abajo se mostrará una tabla con las asignaturas que se tienen en el contrato de estudios. El alumno deberá introducir la convocatoria correcta de la asignatura y pulsar el icono de actualizar para que esta información quede guardada. Si el alumno desea borrar alguna de las asignaturas introducidas solo deberá pulsar el icono de eliminar y la asignatura quedará borrada del contrato de estudios.

- * **Modificar asignaturas Universidad Destino.** Esta sección es para añadir al contrato de estudios las asignaturas de la universidad destino. En primer lugar aparecerá una tabla resumen con las asignaturas de la UCLM que el alumno ha elegido con anterioridad. Esta tabla tiene un campo llamado *numero* que sirve para realizar la asociación entre asignaturas de la UCLM y asignaturas extranjeras.

A continuación se encuentra el formulario para añadir asignaturas. Primeramente aparece una lista desplegable en la que se debe elegir un número haciéndolo corresponder con alguna de las asignaturas mostradas arriba. Después solo queda introducir los datos de la asignatura destino y hacer clic en el botón *Introducir asignatura*.

Después de esto se puede observar como aparece una tabla donde se muestran las asignaturas elegidas de la universidad destino. Si se quisiera eliminar alguna bastaría con pulsar el icono de eliminar y quedaría eliminada del contrato.

- * **Asignar modificaciones.** Todas las modificaciones realizadas en el contrato de estudios implican cambios tanto en las asignaturas de la universidad origen como en las asignaturas de la universidad destino. Por eso cualquier cambio que se haga, ya sea para añadir o anular alguna asignatura de la universidad

destino, tendrá que ser asignado a alguna asignatura de la universidad destino.

En esta pagina se pueden apreciar dos tablas. La primera corresponde a los cambios que se acaben de hacer en las asignaturas de la UCLM. Si alguna asignatura ha sido añadida se podrá ver un 1 en los campos *Añadir E** y *Añadir M** de la tabla. Por otra parte si alguna asignatura ha sido añadida se podrá ver un 1 en los campos *Anular E** y *Anular M** de la tabla.

Esta tabla también tiene el campo código con el que más adelante podremos asignar los cambios entre asignaturas.

Cambios realizados en las asignaturas de la UCLM.-

Código ECTS	Código	Asignatura	Anular E*	Anular M*	Añadir E*	Añadir M*	Créditos ECTS
42503	1	Arquitectura de Computadores	0	0	1	1	6
42403	2	Sistemas Operativos I	0	0	1	1	6
42404	3	Ingeniería del Software I	0	0	1	1	6
42503	4	Arquitectura de Computadores	0	0	1	1	6

Figura A.9: *Modificaciones de las asignaturas de la UCLM en el contrato de estudios*

Si por algún motivo se desea deshacer algún cambio bastará con pinchar en el botón de eliminar para quitar cualquier cambio que se haya seleccionado anteriormente.

En la segunda tabla aparecen las modificaciones que se han hecho recientemente en las asignaturas de la universidad origen. Los asignaturas que hayan sido añadidas tendrán un 1 en el campo *Añadir* y un 0 en el campo *Anular*. Por otro lado las asignaturas que hayan sido borradas tendrán un 1 en el campo *Anular* y un 0 en el campo *Añadir*.

También existe un campo llamado código. Tiene una lista desplegable donde aparecen unos números que se corresponden con los números que aparecen en la primera tabla en cada una de las asignaturas. Este campo sirve para asociar un cambio entre asignaturas.

En la tabla también aparece un campo llamado *Motivo del cambio*. Como su propio nombre indica en este campo de texto el alumno tiene que anotar el motivo por el cuál realiza el cambio de asignatura.

Cambios realizados en las asignaturas de la universidad destino.-

Debe asociar cada cambio de asignatura de la UCLM con algun cambio de la universidad destino.

Código ECTS	Código	Asignatura	Anular	Añadir	Créditos ECTS	Motivo del cambio
MMNM	1	arco1	0	1	7	El coord erasmus no acepta el cambio

Figura A.10: Modificaciones de las asignaturas de la universidad destino en el contrato de estudios

Si por algún motivo se desea deshacer algún cambio bastará con pinchar en el botón de eliminar para quitar cualquier cambio que se haya seleccionado anteriormente.

Para que las asignaciones se hayan realizado correctamente se tendrán que tener asociadas todas las asignaturas de la universidad origen con todas las asignaturas de la universidad destino mediante el campo *Código*. Una vez hecho esto y para que las asignaciones se hagan efectivas solo restará hacer clic en el botón *Asociar asignaturas*. Es posible que después de asociar las asignaturas aparezca un simbolo mostrando algún posible conflicto producido entre algunas asignaturas de la universidad origen y la universidad destino. Bastará con hacer clic en el simbolo de aviso para que aparezca el agente recomendador mostrando los motivos del conflicto.

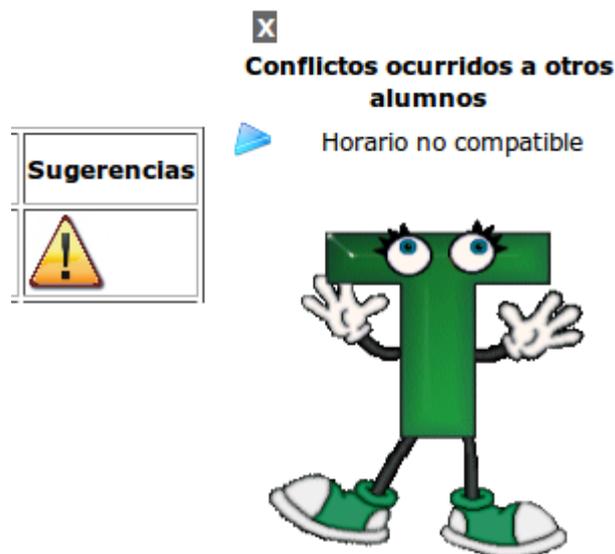


Figura A.11: Agente recomendador mostrando un conflicto entre asignaturas

- * Generar Modificaciones en Pdf. Esta opción en un principio está deshabilitada y no aparece en la barra de pasos del proceso, ya que no es posible modificar el contrato de estudios si previamente no se ha realizado alguna cambio en

las asignaturas del contrato original. En esta pantalla hay que hacer clic en el botón *Generar modificaciones*. Entonces aparecerá el formulario para que el alumno firme el contrato de estudios. Tendrá que rellenar los siguientes datos:

- Tipo de certificado: Se debe seleccionar de una lista desplegable el tipo de certificado que se va a usar para firmar. Actualmente solo está disponible el certificado de tipo *PKCS12*.
- Certificado: Aquí se tiene que seleccionar el archivo del certificado en el directorio local donde se encuentre.
- Clave: Se especifica la clave que se nos proporcione con el certificado.

Firmar Contrato.-



Tipo de Certificado: PKCS12 ▼

Certificado: Examinar...

Clave:

Firmar

Figura A.12: Formulario para que el alumno firme las modificaciones del contrato de estudios

Una vez completado el formulario bastará con pulsar el botón *Firmar* y el contrato de estudios estará firmado.

Se habrá generado entonces el contrato de estudios con las modificaciones pertinentes en formato pdf y estará disponible para su visualización por el alumno y el profesor de movilidad.

- **Rol Profesor de Movilidad.-** Están son las opciones que tiene disponible el profesor de movilidad en la pantalla de inicio.

Módulos disponibles.-



Figura A.13: *Modulos del profesor de movilidad en la pantalla de inicio*

- Lista de alumnos Erasmus. En esta sección el profesor de movilidad podrá ver un listado de todos los alumnos erasmus que tiene a su cargo.

Lista de alumnos con beca Erasmus.-

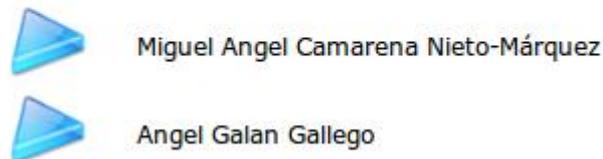


Figura A.14: *Listado de los alumnos erasmus del profesor de movilidad*

- Ver y modificar contratos. En este modulo se muestra un listado de todos los alumnos Erasmus que tiene el profesor de movilidad junto con sus contratos de estudios. Al lado del nombre de cada estudiante Erasmus aparecen dos iconos.

Contratos de Estudios de los Alumnos.-

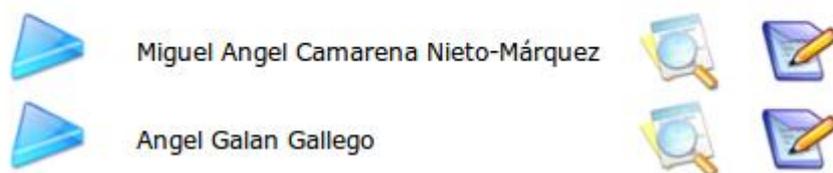


Figura A.15: *Contratos de los alumnos del profesor de movilidad*

El primero de ellos sirve para visualizar el contrato de estudios. Al hacer clic sobre la imagen se abrirá una ventana mostrando el contrato de estudios del estudiante

en formato pdf. Puede ocurrir que el contrato de estudios no pueda ser accedido por el profesor de movilidad en todo momento. Esto es debido a que el contrato puede estar siendo revisado por otros coordinadores o por el propio estudiante. En este caso se mostrará un mensaje avisando que el contrato no puede ser visualizado.

El contrato se encuentra en modo "En modificación". Acceso denegado.

Figura A.16: *Mensaje de error producido por no tener permisos de lectura del contrato de estudios*

El segundo icono permite que el profesor de movilidad realice personalmente alguna modificación al contrato de estudios del estudiante. Puede ocurrir que el contrato de estudios no pueda ser editado por el profesor de movilidad en todo momento. Esto es debido a que el profesor de movilidad no tiene permisos de escritura sobre el contrato en ese momento porque está siendo modificado por otros coordinadores o por el propio estudiante. En este caso se mostrará un mensaje avisando que el contrato no puede ser editado.

- Firmar contratos de estudios. En este apartado se mostrará un listado con los contratos de los alumnos que están pendientes de que se firme su contrato. Aparecerá un listado con los nombres de los alumno junto a tres iconos.

Firmar Contratos de Estudios.-



Figura A.17: *Listado de contratos disponibles para ser firmados*

El primero de ellos servirá para visualizar el contrato de estudios. El profesor de movilidad podrá ver el contrato antes de decidirse o no a firmarlo.

Haciendo clic en la segunda imagen se podrá proceder a la firma del contrato de estudios. Aparecerá entonces un formulario donde el profesor de movilidad tendrá que rellenar ciertos campos para proceder a la firma.

Firmar Contrato.-

Tipo de Certificado: ▼

Certificado:

Clave:

Figura A.18: *Formulario para firmar el contrato de estudios*

- * Tipo de certificado: Se debe seleccionar de una lista desplegable el tipo de certificado que se va a usar para firmar. Actualmente solo está disponible el certificado de tipo *PKCS12*.
- * Certificado: Aquí se tiene que seleccionar el archivo del certificado en el directorio local donde se encuentre.
- * Clave: Se especifica la clave que se nos proporcione con el certificado.

Una vez completado el formulario bastará con pulsar el botón *Firmar* y el contrato de estudios estará firmado.

El tercer icono sirve para rechazar el contrato de estudios del estudiante. Después de haber revisado el contrato de estudios el profesor de movilidad no está de acuerdo con alguna cláusula y lo rechaza. Aparecerá una ventana emergente para que el profesor de movilidad pueda especificar el motivo del rechazo. Inmediatamente será comunicado al alumno en cuestión con el mensaje que explica el motivo del rechazo por parte del profesor de movilidad.

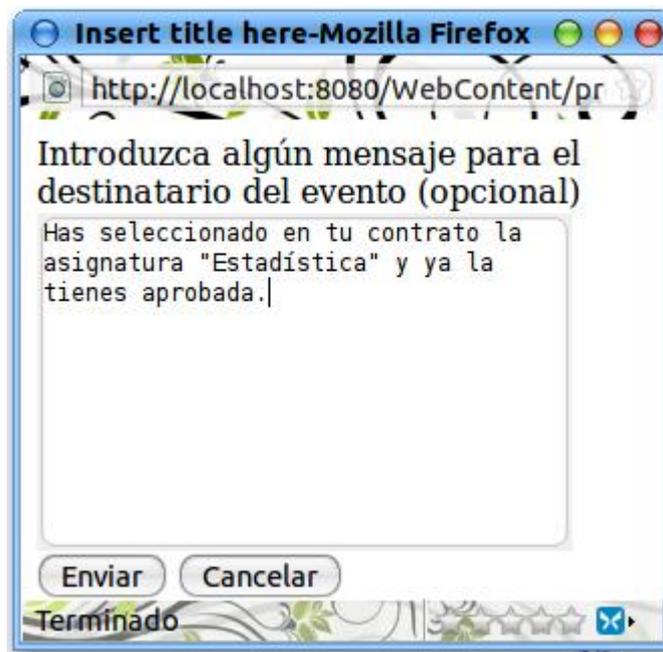


Figura A.19: Mensaje enviado al alumno por rechazar su contrato de estudios

- Contratos firmados. En esta sección el profesor de movilidad puede ver un listado de todos los contratos Erasmus que ha firmado. Al lado del nombre de cada estudiante aparecerá un botón para que el profesor de movilidad pueda visualizar el contrato de estudios que firmó.

Contratos Firmados.-



Figura A.20: Contratos firmados por el profesor de movilidad

- Ver notas. Este modulo sirve para que el profesor de movilidad pueda ver las notas que han sacado los alumnos una vez finalizada su estancia Erasmus en el país extranjero. Cuando los alumnos tengan disponible la ficha de reconocimiento de estudios aparecerá su nombre junto con dos iconos.

Notas de los Alumnos.-

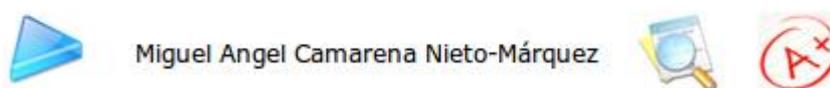


Figura A.21: Listado de alumnos que ya han obtenido las notas finales

El primero de ellos es para visualizar el contrato de estudios que tiene el estudiante Erasmus.

Haciendo clic en el segundo botón aparecerá la ficha de reconocimiento de estudios del alumno en formato pdf donde se podrán apreciar las calificaciones obtenidas en el país extranjero junto con las notas finales traducidas al sistema de calificación español.

- **Rol Coordinador de Centro.-** Estan son las opciones que tiene disponible el coordinador de centro en la pantalla de inicio.

Módulos disponibles.-



Figura A.22: Módulos del coordinador de centro en la pantalla de inicio

- Ver y modificar contratos. En este modulo se muestra un listado de todos los alumnos Erasmus que tiene el coordinador de centro junto con sus contratos de estudios. Al lado del nombre de cada estudiante Erasmus aparecen dos iconos.

Contratos de Estudios de los Alumnos.-

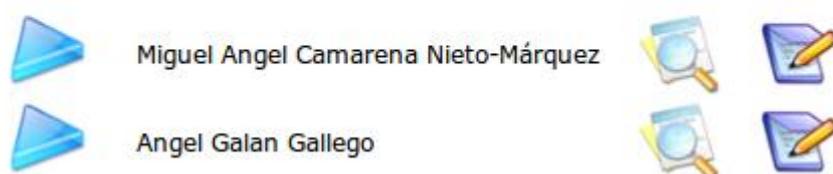


Figura A.23: Contratos de los alumnos del coordinador de centro

El primero de ellos sirve para visualizar el contrato de estudios. Al hacer clic sobre la imagen se abrirá una ventana mostrando el contrato de estudios del estudiante en formato pdf. Puede ocurrir que el contrato de estudios no pueda ser accedido por el coordinador de centro en todo momento. Esto es debido a que el contrato puede estar siendo revisado por otros coordinadores o por el propio estudiante. En este caso se mostrará un mensaje avisando que el contrato no puede ser visualizado.

El segundo icono permite que el coordinador de centro realice personalmente alguna modificación al contrato de estudios del estudiante. Puede ocurrir que el contrato de estudios no pueda ser editado por el coordinador de centro en todo momento. Esto es debido a que el profesor de de centro no tiene permisos de escritura sobre el contrato en ese momento porque está siendo modificado por otros coordinadores o por el propio estudiante. En este caso se mostrará un mensaje avisando que el contrato no puede ser editado.

- Firmar contratos de estudios. En este apartado se mostrará un listado con los contrato de los alumnos que están pendientes de que se firme su contrato. Aparecerá un listado con los nombres de los alumno junto a tres iconos.

Firmar Contratos de Estudios.-



Figura A.24: *Listado de contratos disponibles para ser firmados*

El primero de ellos servirá para visualizar el contrato de estudios. El coordinador de centro podrá ver el contrato antes de decidirse o no a firmarlo.

Haciendo clic en la segunda imagen se podrá proceder a la firma del contrato de estudios. Aparecerá entonces un formulario donde el coordinador de centro tendrá que rellenar ciertos campos para proceder a la firma.

Firmar Contrato.-

Tipo de Certificado: ▼

Certificado:

Clave:

Figura A.25: *Formulario para firmar el contrato de estudios*

- * Tipo de certificado: Se debe seleccionar de una lista desplegable el tipo de certificado que se va a usar para firmar. Actualmente solo está disponible el certificado de tipo *PKCS12*.

- * Certificado: Aquí se tiene que seleccionar el archivo del certificado en el directorio local donde se encuentre.

- * Clave: Se especifica la clave que se nos proporcione con el certificado.

Una vez completado el formulario bastará con pulsar el botón *Firmar* y el contrato de estudios estará firmado.

El tercer icono sirve para rechazar el contrato de estudios del estudiante. Después de haber revisado el contrato de estudios el coordinador de centro no está de acuerdo con alguna cláusula y lo rechaza. Aparecerá una ventana emergente para que el coordinador de centro pueda especificar el motivo del rechazo. Inmediatamente será comunicado al alumno en cuestión con el mensaje que explica el motivo del rechazo por parte del coordinador de centro.

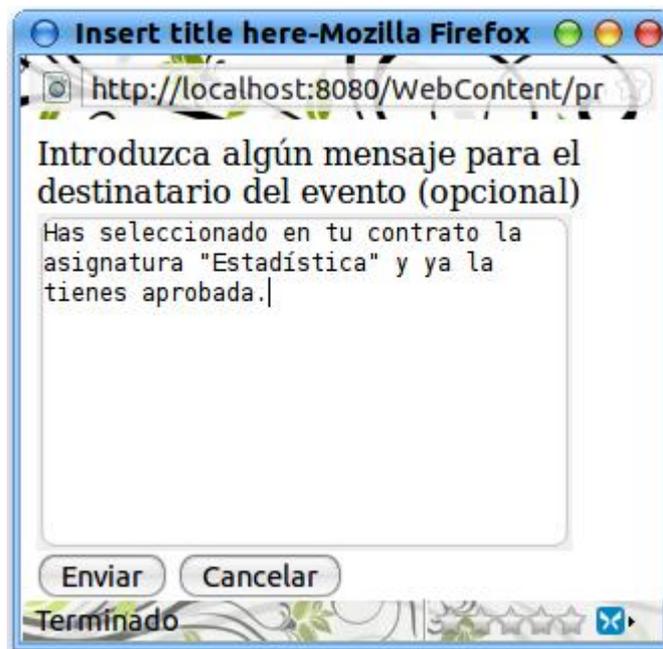


Figura A.26: Mensaje enviado al alumno por rechazar su contrato de estudios

- Contratos firmados. En esta sección el coordinador de centro puede ver un listado de todos los contratos Erasmus que ha firmado. Al lado del nombre de cada estudiante aparecerá un botón para que el coordinador de centro pueda visualizar el contrato de estudios que firmó.

Contratos Firmados.-



Figura A.27: Contratos firmados por el coordinador de centro

- Ver notas. Este modulo sirve para que el coordinador de centro pueda ver las notas que han sacado los alumnos una vez finalizada su estancia Erasmus en el país extranjero. Cuando los alumnos tengan disponible la ficha de reconocimiento de estudios aparecerá su nombre junto con dos iconos.

Notas de los Alumnos.-

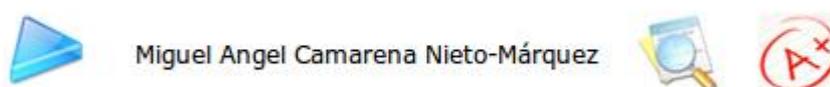


Figura A.28: Listado de alumnos que ya han obtenido las notas finales

- Modificar tabla de calificaciones europea. Esta sección es necesaria porque en

el futuro puede cambiar el sistema de calificación de algún país y es necesario adaptar la herramienta a estos cambios. Se muestra una tabla con todos los países que tienen un sistema de calificación distinto del español. Lo único que hay que hacer para realizar algún cambio es pulsar en el icono de *Editar* que hay a la derecha de cada país.

País	A	B	C	D	E	F/FX	Editar
España (cualitativa)	Matricula de H	Sobresaliente	Notable	Aprobado	Aprobado	Suspense	
España	10 - 10	9.9 - 9	8.9 - 7	6.9 - 6	5.9 - 5	4.9 - 0	
Alemania	1 - 1	1 - 1.9	2 - 2.9	3 - 3.9	4 - 4.9	5 - 6	
Austria	1 - 1	1 - 1.9	2 - 2.9	3 - 3.9	4 - 4.9	5 - 5	
Bélgica	20 - 20	19.9 - 17	16.9 - 14	13.9 - 11	10.9 - 10	9.9 - 7	
Dinamarca	13 - 13	12.9 - 10	9.9 - 9	8.9 - 8	7.9 - 7	6.9 - 6	

Figura A.29: *Tabla de conversión de calificaciones europea*

Después de esto aparecerá una tabla mostrando el país especificado junto con los intervalos. Solo hay que realizar las modificaciones que se estimen oportunas y pulsar el botón *Actualizar* y los cambios habrán quedado registrados.

Editar la tabla de conversión del país.-

País	A	B	C	D	E	F/FX
España (cualitativa)	Matricula de H	Sobresaliente	Notable	Aprobado	Aprobado	Suspense
Bélgica	20 - 20	19.9 - 17	16.9 - 14	13.9 - 11	10.9 - 10	9.9 - 7

[Volver](#) [Actualizar](#)

Figura A.30: *Tabla para editar los intervalos de calificación de un país específico*

- Calificar a los alumnos. Aquí es donde el coordinador de centro tiene que calificar al estudiante Erasmus. Tendrá que coger el listado de notas que traiga el alumno de su país extranjero y pasarlas a la herramienta. En este modulo aparece un listado con los alumnos Erasmus y dos iconos al lado de cada nombre.

Lista de alumnos con beca Erasmus.-

	Miguel Angel Camarena Nieto-Márquez		
	Angel Galan Gallego		

Figura A.31: *Listado de alumnos erasmus para calificar*

El primero de ellos es para visualizar el contrato de estudios antes de pasar a realizar las calificaciones.

Haciendo clic en el segundo icono aparecerá una tabla donde se muestran las asignaturas que se han cursado en la universidad destino. Solo hay que rellenar el campo *Nota* y pulsar en actualizar para que automáticamente la herramienta convierta esa nota al valor correcto de acuerdo al sistema de calificación español.

Asignaturas de la Universidad Destino.-

Asignatura extranjera	Nota	Asignatura UCLM	Calificación final
Operating Systems	1.5 	Sistemas Operativos I	6 - Aprobado
Computer Architecture	0.8 	Arquitectura de Computadores	4 - Suspenso

Generar Ficha de Reconocimiento de Estudios

Figura A.32: Listado de alumnos erasmus para calificar

Cuando todas las notas se hayan metido en la tabla hay que generar la ficha de reconocimiento de estudios. Es el documento oficial donde se especifican las asignaturas cursadas en la universidad origen y destino, junto con las notas definitivas que tiene el alumno. Al hacer clic en el botón *Generar ficha de reconocimiento de estudios* se mostrará el formulario para proceder a la firma del documento. Estos son los campos que habrá de rellenar el coordinador de centro:

Firmar Contrato.-

Tipo de Certificado:

Certificado:

Clave:

Figura A.33: Formulario para firmar la ficha de reconocimiento de estudios

* Tipo de certificado: Se debe seleccionar de una lista desplegable el tipo de

certificado que se va a usar para firmar. Actualmente solo está disponible el certificado de tipo *PKCS12*.

- * Certificado: Aquí se tiene que seleccionar el archivo del certificado en el directorio local donde se encuentre.
- * Clave: Se especifica la clave que se nos proporcione con el certificado.

Una vez completado el formulario bastará con pulsar el botón *Firmar* y la ficha de reconocimiento de estudios estará firmada.

- **Rol Coordinador Institucional.-** Están son las opciones que tiene disponible el coordinador institucional en la pantalla de inicio.
 - Ver y modificar contratos. En este módulo se muestra un listado de todos los alumnos Erasmus que tiene el coordinador institucional junto con sus contratos de estudios. Al lado del nombre de cada estudiante Erasmus aparecen dos iconos.

Contratos de Estudios de los Alumnos.-

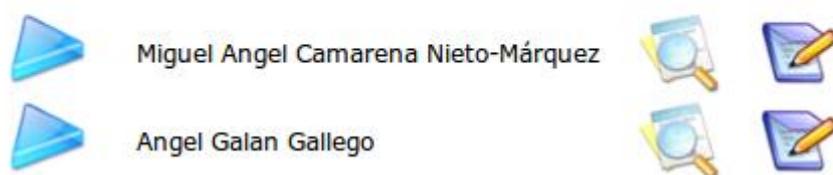


Figura A.34: *Contratos de los alumnos del coordinador institucional*

El primero de ellos sirve para visualizar el contrato de estudios. Al hacer clic sobre la imagen se abrirá una ventana mostrando el contrato de estudios del estudiante en formato pdf. Puede ocurrir que el contrato de estudios no pueda ser accedido por el coordinador institucional en todo momento. Esto es debido a que el contrato puede estar siendo revisado por otros coordinadores o por el propio estudiante. En este caso se mostrará un mensaje avisando que el contrato no puede ser visualizado.

El segundo icono permite que el coordinador institucional realice personalmente alguna modificación al contrato de estudios del estudiante. Puede ocurrir que el contrato de estudios no pueda ser editado por el coordinador institucional en todo momento. Esto es debido a que el coordinador institucional no tiene permisos de escritura sobre el contrato en ese momento porque está siendo modificado por otros coordinadores o por el propio estudiante. En este caso se mostrará un mensaje avisando que el contrato no puede ser editado.

- Firmar contratos de estudios. En este apartado se mostrará un listado con los contratos de los alumnos que están pendientes de que se firme su contrato. Aparecerá un listado con los nombres de los alumno junto a tres iconos.

Firmar Contratos de Estudios.-



Figura A.35: Listado de contratos disponibles para ser firmados

El primero de ellos servirá para visualizar el contrato de estudios. El coordinador institucional podrá ver el contrato antes de decidirse o no a firmarlo.

Haciendo clic en la segunda imagen se podrá proceder a la firma del contrato de estudios. Aparecerá entonces un formulario donde el coordinador institucional tendrá que rellenar ciertos campos para proceder a la firma.

Firmar Contrato.-

Tipo de Certificado:

Certificado:

Clave:

Figura A.36: Formulario para firmar la ficha de reconocimiento de estudios

- * Tipo de certificado: Se debe seleccionar de una lista desplegable el tipo de certificado que se va a usar para firmar. Actualmente solo está disponible el certificado de tipo *PKCS12*.
- * Certificado: Aquí se tiene que seleccionar el archivo del certificado en el directorio local donde se encuentre.
- * Clave: Se especifica la clave que se nos proporcione con el certificado.

Una vez completado el formulario bastará con pulsar el botón *Firmar* y el contrato de estudios estará firmado.

- Contratos firmados. En esta sección el coordinador institucional puede ver un listado de todos los contratos Erasmus que ha firmado. Al lado del nombre de cada

estudiante aparecerá un botón para que el coordinador de centro pueda visualizar el contrato de estudios que firmó.

Contratos Firmados.-



Figura A.37: Contratos firmados por el coordinador institucional

- **Rol Administrador.-** Estan son las opciones que tiene disponible el administrador en la pantalla de inicio.

Módulos disponibles.-



Figura A.38: Módulos del administrador en el pantalla de inicio

- Modificar permisos. La herramienta tiene disponibles muchos modulos. Por cuestiones de seguridad no todos ellos pueden ser accedidos por cualquier usuario de la herramienta. No sería una buena opción que los estudiante pudieran acceder al modulo de calificaciones del coordinador de centro. Por este motivo cada modulo tiene permisos de lectura y escritura. Esto determina que usuarios pueden acceder y modificar cada módulo. De adjudicar que permisos tiene cada modulo se encarga el administrador.

En esta pantalla se puede ver un listado de todos los módulos disponibles en la aplicación. Después de cada módulo aparecen dos imagenes.

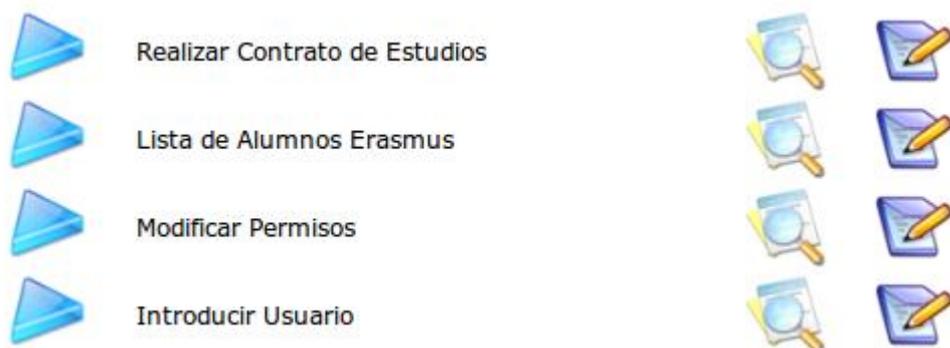


Figura A.39: Listado de algunos módulos del sistema

La primera sirve para previsualizar el módulo. Al hacer clic en la imagen el administrador accederá al módulo. De esta manera podrá visualizar el módulo al que está adjudicando los permisos.

La segunda sirve para adjudicar los permisos de ese módulo en particular. Se muestra una lista de los roles disponibles y al lado unos checkbox para elegir los permisos de lectura y escritura.

Rol	Lectura Escritura	
Estudiante	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Coordinador Centro	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Profesor Movilidad	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Coordinador Institucional	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Administrador	<input type="checkbox"/>	<input type="checkbox"/>

Figura A.40: *Configuración de permisos de un módulo del sistema*

- Introducir usuario. Cada usuario antes de poder utilizar la herramienta debe realizar el registro con el administrador. Este tomará nota de todos los datos necesarios para crear el registro. El administrador suministrará unos datos de acceso (nombre de usuario y contraseña) al estudiante o coordinadores. A partir de este momento ya podrán hacer uso de la herramienta.

Nombre:	<input type="text"/>
Primer Apellido:	<input type="text"/>
Segundo Apellido:	<input type="text"/>
D.N.I.:	<input type="text"/>
Estudios:	<input type="text" value="I.T. Sistemas"/>
Fecha de Nacimiento:	<input type="text"/> / <input type="text"/> / <input type="text"/> (dd/mm/aaaa)
E-mail:	<input type="text"/>
Repita E-mail:	<input type="text"/>
Movil:	<input type="text"/>
Sexo:	<input type="text" value="Hombre"/>
Nacionalidad:	<input type="text" value="Alemania"/>
Lugar de Nacimiento:	<input type="text"/>
Dirección Permanente:	<input type="text"/>
Estado Civil:	<input type="text" value="Soltero"/>
Login:	<input type="text"/>
Contraseña:	<input type="text"/>
Repita contraseña:	<input type="text"/>
Elija su rol:	<input type="text" value="Estudiante"/>

Figura A.41: *Formulario de registro de los usuarios*

Apéndice B

Normativa Erasmus.-

PROCEDIMIENTO PARA LA ELABORACIÓN DEL CONTRATO DE ESTUDIOS DENTRO DEL PROGRAMA ERASMUS

La Oficina de Relaciones Internacionales requiere de los distintos centros la homogeneización de los criterios o pautas a seguir en el proceso de elaboración de los contratos de estudio así como en el papel que deben asumir los distintos actores del proceso (estudiantes, responsables de programa, coordinador del centro, etc.). Por todo esto, la Dirección de la Escuela Superior de Informática basándose *en la Guía del Coordinador de Relaciones Internacionales* editada en el año 2007 por la Oficina de Relaciones Internacionales establece las siguientes líneas generales de actuación en el proceso de elaboración del contrato de estudios:

- 1) El alumno de la E.S.I. debe escoger programas Erasmus del **área de Informática**.
- 2) Una vez seleccionado, el alumno debe contactar con el **profesor responsable del programa**. Este le informará sobre el plan de estudios de la universidad receptora y del cuadro de convalidaciones más idóneo en función del plan de estudios de la E.S.I. y de la situación particular del estudiante. El profesor responsable del programa elaborará una **propuesta de contrato de estudios** junto con el estudiante.
- 3) El alumno presentará en la Secretaría de la E.S.I. la propuesta de contrato de estudios firmada por el profesor responsable del programa. El **Coordinador de Relaciones Internacionales del Centro** comprobará que la propuesta de contrato cumpla los criterios específicos de la E.S.I. (adjuntos en este documento) y en dicho caso procederá a su firma. Las propuestas de contratos de estudios que no se ajusten a estos criterios y que tenga un carácter singular serán analizadas por la **Comisión Académica** de la E.S.I.
- 4) Posteriormente, la E.S.I. remitirá el anterior documento a la Oficina de Relaciones Internacionales para su firma por el **Coordinador de Relaciones Internacionales del Campus**. Tras la realización de este trámite el estudiante tendrá el **contrato de estudios**.
- 5) El contrato de estudios tiene que estar debidamente formalizado en el momento que el alumno deba realizar la matrícula en la UCLM, ya que el alumno se tiene

que matricular de todas las asignaturas especificadas en el mismo.

- 6) El contrato de estudios original podrá ser modificado una vez que el alumno se haya incorporado en la universidad de destino. Cualquier **modificación del contrato de estudios** deberá ser evaluada por el profesor responsable del programa, quien firmará dicha modificación junto con el Coordinador de Relaciones Internacionales del Centro. Posteriormente, se remitirán las modificaciones a la Oficina de Relaciones Internacionales. No se podrán realizar modificaciones del contrato de estudios después del **28 de febrero**.

CRITERIOS ESPECÍFICOS DE LA ESCUELA SUPERIOR DE
INFORMÁTICA PARA LA ELECCIÓN DE LAS ASIGNATURAS DEL
CONTRATO DE ESTUDIOS

- 1) Debido a la heterogeneidad entre los diversos Planes de Estudio es recomendable que las asignaturas elegidas correspondan a créditos de **libre configuración u optativas**.
- 2) No se admitirá la inclusión en el contrato de estudios de asignaturas que el alumno curse en **cuarta o posterior convocatoria** dentro del plan de estudios de la E.S.I.
- 3) El contrato de estudios debe ser equilibrado. Para estancias de un semestre la carga lectiva estará comprendida entre **15 ECTS** y **33 ECTS** y para estancias anuales la carga lectiva estará comprendida entre **30 ECTS** y **66 ECTS**.
- 4) El **Proyecto Fin de Carrera** (PFC) se puede realizar durante una estancia Erasmus si el profesor responsable del programa lo estima oportuno. Las cláusulas que regirán la realización de un PFC en un programa Erasmus son:
 - 4.1) La normativa propia de la E.S.I. para la realización del PFC establece la figura de un **tutor académico** en el caso que el director del PFC sea de otra universidad. El profesor responsable del programa hará la función de tutor académico.
 - 4.2) Para el reconocimiento de la calificación obtenida en la universidad de destino por la E.S.I. el alumno deberá entregar una memoria del PFC, ajustado a la normativa, con el visto bueno del tutor académico y que será depositado en la biblioteca.

**SELECCIÓN DE CANDIDATOS POR PARTE DEL RESPONSABLE DEL
PROGRAMA EN LA E.S.I.**

La E.S.I. seleccionará a los estudiantes que concurren a la Beca Erasmus entre los alumnos de Ingeniería Técnica de Sistemas y Gestión que tengan aprobadas **todas las asignaturas de primer curso** y entre los alumnos de Ingeniería Informática que tenga aprobadas **todas las asignaturas de primer y segundo curso**.

La selección de los candidatos se basa en los siguientes criterios establecidos por la Oficina de Relaciones Internacionales:

- Expediente académico. La puntuación obtenida en este apartado se calculará multiplicando la nota del expediente académico por 0.5 y por la proporción de créditos superados respecto al total de su Plan de Estudios.
- Conocimiento de la lengua en la que se impartan las clases en la universidad de destino. Este apartado se valorará hasta 3 puntos. Esta nota se desagrega en: i) nivel de conocimiento del idioma del lugar de destino, acreditado a través del CEVI-Erasmus y valorado hasta 1.8 puntos y ii) las estancias efectuadas en países de dicha lengua valoradas hasta 1.2 puntos.
- *Currículum Vitae*. Este apartado se valorará hasta 0.5 puntos.
- Adecuación del perfil curricular del candidato a las características específicas del programa. Este apartado se valorará hasta 2 puntos.
- Haber concurrido como alumno Cicerone. Este apartado se valorará hasta 0.5 puntos.
- En los programas que así lo requieran, se mantendrá una entrevista personal o por videoconferencia en el idioma del país de la plaza entre el estudiante y el responsable del programa.

Apéndice C

Contrato de Estudios.-

FORMULARIO DE CANDIDATURA, ESTUDIANTES UCLM
STUDENT APPLICATION FORM, UCLM STUDENTS



PROGRAMA DE MOVILIDAD
MOBILITY PROGRAMME

- ERASMUS
 PROGRAMA DE BECAS AECI/ICI
 CONVENIO DE COOPERACIÓN / Cooperation agreement
 OTROS/ Others

Año Académico / Academic year _____
Área de estudios / Field of Studies _____

2. DATOS PERSONALES DEL ESTUDIANTE/ STUDENT PERSONAL DATA
(Para rellenar por el estudiante / To be completed by the student)

Apellidos / Family name _____ Nombre / First name _____
Fecha de nacimiento / Date of birth _____ Sexo / Sex _____ Nacionalidad / Nationality _____
Lugar de nacimiento / Place of birth _____ Estado civil / Marital status _____
Dirección permanente / Permanent address _____
E-Mail _____
Núm. de D.N.I. o pasaporte / ID or passport _____ Teléfono / Telephone _____

3. UNIVERSIDAD ASOCIADA / PARTNER UNIVERSITY

Nombre legal de la Institución y domicilio completo / Institution name and full address _____
Coordinador institucional: nombre, teléfono, fax y correo electrónico / Institutional Coordinator: name, telephone, fax and e-mail _____
Profesor responsable de la movilidad: nombre, teléfono, fax y correo electrónico / Professor responsible for the mobility: name, tel., fax and e-mail _____

4. UNIVERSIDAD DE CASTILLA-LA MANCHA

Coordinador institucional: nombre, dirección, teléfono, fax y correo electrónico / Institutional coordinator: name, address, telephone, fax and e-mail _____
Coordinador de Centro: nombre, teléfono, fax y correo electrónico / Local coordinator: name, telephone, fax and e-mail _____
Centro / Faculty _____
Profesor responsable de la movilidad: nombre, tel., fax y correo electrónico / Professor responsible for the mobility: name, telephone, fax and e-mail _____

5. APTITUDES LINGÜÍSTICAS/LANGUAGE COMPETENCES

Lengua materna / Mother tongue _____
Lengua en que se imparte la enseñanza en la universidad de destino / Language of instruction at home institution _____
Nivel de conocimiento / Level
 Avanzado / Advanced Intermedio / Intermediate Necesita preparación / I need preparation

6. ESTUDIOS REALIZADOS Y EN CURSO / PREVIOUS AND CURRENT STUDIES

Licenciatura-Diplomatura- Grado / Degree for which you are currently studying _____
Área de estudios / Field of studies _____ Código Erasmus / Erasmus code _____ Duración / Duration _____ años/years
Curso que realiza / Academic year you are currently studying _____

7. PERÍODO DE ESTUDIOS SOLICITADO / PERIOD OF STUDY ABROAD

Duración del periodo de estudios / Length of study period _____ meses / months
Fecha de inicio / Beginning date _____
Fecha de fin / Ending date _____

Apéndice D

**Ficha de Reconocimiento de
Estudios.-**

Apéndice E

**Tabla de Conversión de
Calificaciones Europeas.-**

**TABLA DE CONVERSIÓN DE CALIFICACIONES EUROPEAS /
EUROPEAN MARKINGS CONVERSION TABLE**

NOTA ECTS % alumnos aprobados que normalmente alcanzan la calificación % passed students with the given marking	A 10%	B 25%	C 30%	D 25%	E 10%	F/FX
España (numérica) / Spain (numeric)	10	9	8-7	6	5	4-1
España (cualitativa) / Spain (qual.)	Matricula de H	Sobresaliente	Notable	Aprobado	Aprobado	Suspense
Cualitativa int. / Int. Qualitative scale	Excellent	Very good	Good	Satisfactory	Sufficient	Fail
Alemania / Germany	1.0	1-	2	3	4	5/6
Austria	1	1	2	3	4	5
Bélgica / Belgium	20	19/18/17	16/15/14	13/12/11	10	9/8/7
Dinamarca / Denmark	13	11/10	9	8	7	6
Finlandia / Finland	3	2 ½	2	1 ½	1	-
Francia France	20 - excellent	18/16 - tres bien	15/14 - bien=B	13/12 assez-bien=AB	11/10 passable =P	9/8/7 - echec = E
Grecia / Greece	10	9/8	7	6	5	4-1
Holanda / Holland	10	9/8 ½	8/7 ½	7/6 ½	6	5-1
Irlanda / Ireland	1st Class Hon. A = 70%	2nd Class Hon I A/B+ = 60%	2nd Class Hon II B = 50%	3rd Class Hon. C = 45%	Pass D = 44-40%	Fail E/F = 0-39%
Islandia / Island	10	9 - 8	7	6	5	fail
Italia / Italy	30 lode	30/29	28/27	26/25	24-18	17-1
Portugal	20	19/18/17	16/15/14	13/12	11/10	9-1
Reino Unido / U.K.	A	A-	B+/B	C/D	E	Fail
Suecia / Sweden	VG (Val Godkand - Pass with distinction)	VG	G (Godkand)	G	G	U (Underkant)
Suiza / Switzerland	6	5 ½	5	4 ½	4	-

Fuente / Source: Vicerectorado de Relaciones Internacionales



Bibliografía

- [1] *Apache tomcat - the apache software foundation*. <http://tomcat.apache.org/>.
- [2] *Asp official website*. <http://www.asp.net/>.
- [3] *Drupal official website*. <http://drupal.org/>.
- [4] *Gnu privacy guard*. <http://www.gnupg.org/gph/es/manual.html>.
- [5] *Java official website*. <http://www.java.com/es/>.
- [6] *Joomla official website*. <http://www.joomla.org/>.
- [7] *Jsignpdf official website*. <http://jsignpdf.sourceforge.net/>.
- [8] *Ono official website*. <http://www.ono.es/ofertas/recomendador/>.
- [9] *Openoffice official website*. <http://www.openoffice.org/>.
- [10] *Php official website*. <http://www.php.net/>.
- [11] *A turing machine in the classic style*. <http://aturingmachine.com/>.
- [12] *Vodafone official website*. <http://www.vodafone.es/particulares/es/moviles-y-fijo/tarifas/recomendador-de-tarifas/>.
- [13] *W3 official website*. <http://www.w3.org/>.
- [14] Abbey, Michael, Michael Corey, and Ian Abramson: *Oracle 9i : a beginner's guide*. Mac Graw Hill, 2002.
- [15] Aguíñ Alarcón, José Manuel: *Programación en JavaScript*. Anaya Multimedia, 2000.
- [16] Bluttman, Ken and Wayne S. Freeze: *Análisis de datos con Access*. Anaya Multimedia, 2007.
- [17] Buchmann, Johannes A.: *Introduction to cryptography*. Springer-Verlag, 2001.
- [18] Chiaretta, Simone: *Beginning ASP.NET MVC 1.0*. Wiley Publishing, 2009.

-
- [19] Dalton, Patrick and Paul Whitehead: *SQL Server 2000*. Anaya Multimedia, 2001.
- [20] Fegghi, Jalal and Peter Williams: *Digital Certificates: Applied Internet Security*. Addison-Wesley Professional, 1998.
- [21] Gilfillan, Ian: *La biblia de MySQL*. Anaya Multimedia, 2003.
- [22] Jennings, N. R. and M. Wooldridge: *Intelligent agents and multi-agent systems*. Applied Artificial Intelligence special issue, 1995.
- [23] Krug, Steve: *No me hagas pensar. Una aproximación a la usabilidad en la Web*. Prentice Hall, 2006.
- [24] Mitchel, T: *Machine Learning*. McGraw Hill, 1997.
- [25] Nadal Martinez, Apolonia: *Comercio electrónico, firma digital y autoridades de certificación*. S.L. Civitas ediciones, 2001.
- [26] Nielsen, Jakob: *Usabilidad. Jacob Nielsen 50 sitios web*. Prentice Hall, 2002.
- [27] Patzer, Andrew: *JSP: Ejemplos prácticos*. Anaya Multimedia, 2002.
- [28] Watson, Ian: *Applying Case-Based Reasoning: Techniques for Enterprise Systems*. Morgan Kaufmann, 1997.
- [29] Weiss, Gerhard: *Multiagent systems : a modern approach to distributed artificial intelligence*. Cambridge (Massachusetts), 2000.