



**Universidad
Zaragoza**

Proyecto Fin de Carrera

"SOFTWARE DE GESTIÓN PARA LA DOCUMENTACIÓN ACADÉMICA EN CENTROS DE SECUNDARIA".

Autor

José Rafael Isarre Arnal

Director

Dr. José Javier Merseguer

Departamento de Informática e Ingeniería de Sistemas

Escuela de Ingeniería y Arquitectura

Ingeniería en Informática

Junio 2012

Tomo: ½

“SOFTWARE DE GESTIÓN PARA LA DOCUMENTACIÓN ACADÉMICA EN CENTROS DE SECUNDARIA”

RESUMEN

Este PFC propone un “Software de gestión para la documentación académica en Centros de Secundaria”, englobando dos aplicaciones complementarias para centros de Educación Secundaria y el Servicio Provincial de Educación.

Es un modelo de carácter estructural que afecta a la gestión, organización y comunicación de los procesos administrativos y documentales de los centros. Esta propuesta es una herramienta formal y legal, que ofrece diferentes aplicaciones de tramitación, transmisión y agilización de la documentación educativa, y que ha sido construida para ponerla en práctica con posterioridad.

Esta aplicación de gestión hace una propuesta concreta, útil y viable en torno a tres ámbitos: gestión administrativa, documentación académica y gestión de recursos. Permite opciones para compartir información (uso, selección, gestión, organización), buscando la mayor eficiencia en términos de costes y tiempos.

A partir de la consideración de aspectos previos (estudio de viabilidad, DAR), se continuó con el proceso de diseño del sistema y se tomaron decisiones sobre las herramientas a utilizar (Java, NSIS, Jasperreport, MySQL, Java Web Start, etc.), y se implementó el proyecto realizando los diagramas entidad-relación, diseñando las interfaces de usuario, elaborando los ficheros que genera la aplicación, creando la firma digital para validar los archivos jar, etc.

Finalmente, se realizaron dos tipos de instaladores para adaptar el sistema a las necesidades de los centros educativos y simplificar así el despliegue de la aplicación.

El programa ha requerido un seguimiento exhaustivo que se ha ajustado a la planificación inicial, a pruebas del sistema realizadas por los usuarios finales, reuniones para tomar las decisiones sobre el diseño de interfaz de usuario, etc.

La mejora de procesos y métodos en la gestión de los centros, justifica su implantación para dar respuesta a una necesidad coyuntural ya que desarrolla una aplicación completa: desde un análisis de requisitos hasta su utilización por el usuario.

0. Índice

1. Introducción.	4
2. Fase de Análisis en el Proceso de Construcción del Software.	7
2.1 Análisis de Requisitos (DAR)	8
2.2 Diseño del Sistema	9
2.2.1 Entidad – Relación.	9
2.2.2 Diagrama de Estados	12
2.3 Implementación de la aplicación	13
2.4 Herramientas utilizadas	14
2.5 Secuenciación temporal del proyecto	18
3. Plan de aseguramiento de la calidad del Software.	19
3.1 Pruebas del sistema	20
3.2 Diseño de la interfaz	20
4. Instalación.	21
5. Resultados obtenidos.	26
6. Conclusiones.	30
7. Referencias.	31

1. Introducción

Este PFC (Proyecto Fin de Carrera) “SOFTWARE DE GESTIÓN PARA LA DOCUMENTACIÓN ACADÉMICA EN CENTROS DE SECUNDARIA”, engloba dos aplicaciones independientes pero complementarias en el ámbito en el cual se van a instalar: Centros Educativos de Infantil, Primaria y Secundaria y Servicio Provincial de Educación. Abarca desde componentes curriculares a procesos relacionados con la gestión y organización.

- La primera aplicación está concebida como una guía práctica para la gestión de centros docentes de Educación Secundaria. Los aspectos tratados han sido seleccionados entre aquellos que con mayor frecuencia se presentan en la gestión diaria del centro educativo: faltas de asistencia, evaluación de alumnos, partes de disciplina, generación de horarios, boletines y actas de evaluación, etc.
- El segundo programa agilizará todos los procesos que se están desarrollando en el Servicio Provincial de Educación y que engloban la comunicación y coordinación con los distintos centros educativos. El objetivo fundamental es automatizar las tareas que se vienen realizando en la actualidad de forma manual (rutas de transporte, mobiliario escolar, etc.).

La legislación educativa regula la vida institucional y administrativa de los centros y es necesario disponer de aplicaciones y procedimientos que faciliten gestionar el acceso rápido a modelos documentales y administrativos ligados a la gestión del centro.

Desde esta perspectiva, este trabajo debe entenderse como una herramienta formal y legal para la tramitación y transmisión de la documentación académica y administrativa para uso de los centros, es pues, un instrumento para el tratamiento de la información educativa de diversa índole, agilizar tareas y diseñar otras nuevas.

Teniendo en cuenta esta orientación, la selección, organización y tratamiento de los contenidos se concreta en tres ámbitos principales:

- Gestión administrativa:
 - Documentación académica.
 - Administración y Servicios.

- Registros y comunicaciones.
- Solicitudes, matrículas.
- Permisos y calificaciones.
- Documentación académica:
 - Datos personales.
 - Gestión de expedientes.
 - Actas de evaluación.
 - Acreditaciones de la evaluación.
 - A.C.N.E.E. (Alumnos con necesidades educativas especiales)
 - Observaciones.
- Gestión de recursos:
 - Mobiliario.
 - Rutas de transporte.
 - Gestión de material (stock y solicitud).

De manera esquemática las principales aplicaciones de gestión las represento en el siguiente mapa conceptual:

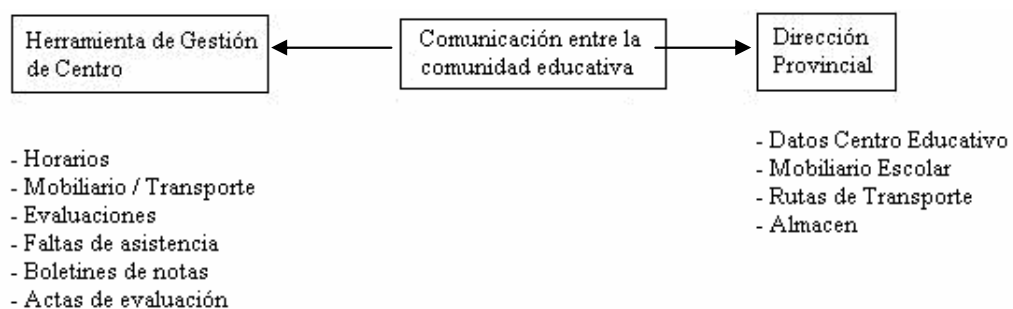


Figura 1: Relación entre las aplicaciones desarrolladas
Página 5 de 31

El desarrollo de este PFC surge tras la observación y análisis de las distintas herramientas utilizadas, tanto en centros educativos de secundaria como en el Servicio Provincial de Educación:

- En el caso de los centros de Educación Secundaria no existe una herramienta de gestión desarrollada desde el Departamento de Educación. Esto supone recurrir a aplicaciones implementadas hace varios años que no han sido adaptadas a las nuevas realidades y necesidades del centro/enseñanzas, y carecen de mantenimiento por parte de la empresa contratada inicialmente, al haber finalizado el contrato en vigor.
- Respecto al Servicio Provincial, se ha comprobado que cada Departamento funciona como un ente independiente, con sus propias herramientas, y esto origina problemas de redundancia de la información, falta de coherencia en los datos manejados por los diversos departamentos, etc. No existe una única aplicación que integre las necesidades de todas las secciones que lo conforman. Las herramientas utilizadas y el método de trabajo resultan poco operativas, puesto que la comunicación con los centros educativos se hace en formato papel, transcribiendo los datos que son comunicados por los centros a ficheros de datos en Access, con los problemas que esto conlleva: pérdida de tiempo, consumo de recursos económicos y humanos, errores, etc.

Teniendo en cuenta los aspectos comentados anteriormente, y una vez detectadas las necesidades y marcados los objetivos, he desarrollado un conjunto de aplicaciones destinadas a agilizar todos estos procesos, y subsanar las carencias identificadas en la gestión, tanto en los centros como en el Servicio Provincial.

Se trata de contribuir en la medida de lo posible a:

- Agilizar procesos de gestión mediante nuevas herramientas.
- Lograr una información permanente, actualizada e inmediata a disposición de todos.
- Posibilitar estudios comparativos, evolutivos, históricos, etc.
- Aportar instrumentos para el tratamiento de la información: gestión y control.

En todo caso, se trata de disponer de opciones que permitan compartir información (uso, selección, gestión, organización), utilizando una Bases de Datos, buscando la mayor explotación de los recursos y una mayor eficiencia en términos de costes y tiempos.

Para el desarrollo de este conjunto de herramientas mantuve varias reuniones con los usuarios finales. Se realizaron en el entorno de trabajo (docente/administrativo), para conocer los métodos de trabajo utilizados, recabar mayor información, y así mismo realicé pequeños muestreos con un grupo reducido de usuarios con distinto nivel de destreza en el manejo de herramientas informáticas.

Una vez finalizadas estas reuniones, que fueron desarrollándose a lo largo de varias semanas (el tiempo necesario hasta formarme una idea clara de lo realmente necesario a implementar), extraje conclusiones que fueron el punto de partida del documento de Análisis de Requerimientos (DAR) para su aprobación. Una vez aprobado el DAR se comenzó a elaborar la documentación del proyecto adjunto en los anexos.

2. Fase de Análisis en el Proceso de Construcción del Software

Antes de realizar la propuesta de este PFC efectué un estudio de viabilidad para asegurar que el desarrollo de este conjunto de aplicaciones estuviera ajustado en plazo y entregado en la fecha inicialmente prevista.

El proceso de análisis lo concreto en cuatro aspectos:

- Identificar los elementos exigidos por la administración educativa.
- Determinar las demandas de los usuarios.
- Incorporar las nuevas propuestas dentro de un contexto concreto.
- Diseñar soluciones adecuadas a las distintas necesidades de los centros y servicios educativos.

Creado el marco de referencia y una vez recogidas las indicaciones de los usuarios consideré los siguientes aspectos: ¿Qué es lo que debe hacer la aplicación?, ¿Para qué va a utilizarse?, ¿Cómo debe funcionar?, ¿Qué ventajas va a reportar esta herramienta como instrumento de trabajo?.

Las respuestas a estas preguntas las encontré tras mantener distintas reuniones con profesores, equipos directivos de los centros educativos y trabajadores de la administración educativa.

Durante el desarrollo de estas reuniones celebradas en el entorno del trabajo del usuario final, analicé la metodología de trabajo real. Esto me permitió detectar tareas repetitivas y susceptibles de ser mejoradas que no habían sido inicialmente planteadas por el usuario. En este punto resaltaría la

dificultad, en algunas ocasiones, para obtener información acerca del procedimiento seguido en temas administrativos y organizativos, ya que no los describían con suficiente claridad, y en otros casos me resultó complejo recabar toda la información, puesto que mientras revisaba el resultado de las reuniones me surgían nuevas dudas a las que había que dar respuesta. Una vez analizada la realidad y con todos los datos, extraje las conclusiones que son el punto de partida de la planificación.

2.1 Análisis de Requisitos (DAR)

La conclusión final, tras las reuniones comentadas en el apartado anterior, puede verse en el Anexo II (Documento de Análisis de Requisitos). Este contiene todos los requerimientos (clasificados en funcionales y no funcionales) que deben cumplir ambas aplicaciones:

- **Requerimientos Funcionales:** Definen el comportamiento interno de la aplicación. Detallan lo que se debe realizar.
- **Requerimientos No Funcionales:** Especifican cómo debe comportarse la aplicación.

El DAR ha sido revisado en varias ocasiones a lo largo del ciclo de vida del producto software. Se inició elaborando y comentando con el cliente el diagrama de Casos de Uso y planificando el escenario en cada caso¹:

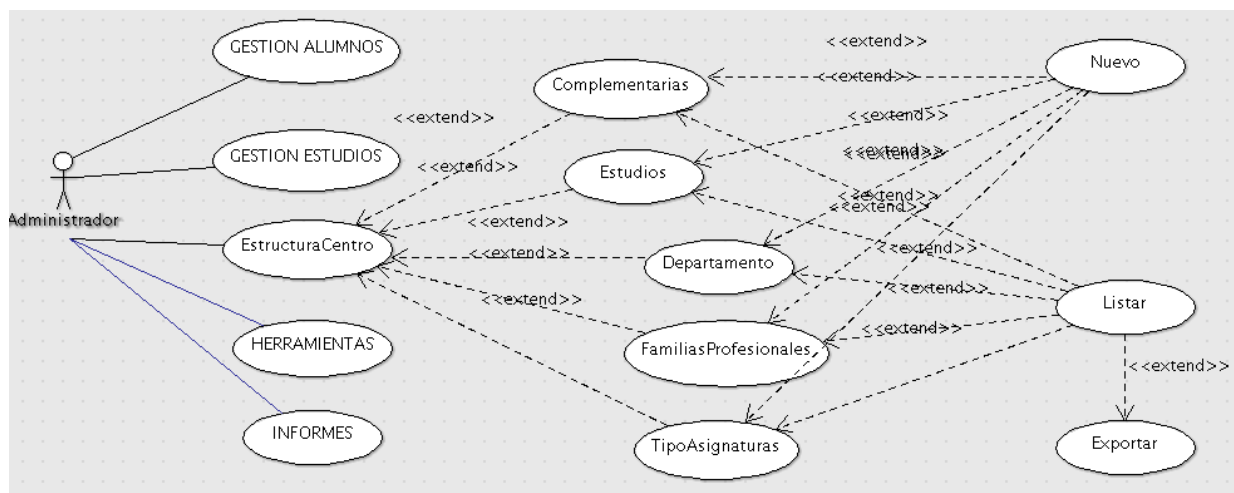


Figura 2: Casos de Uso del Administrador

El objetivo es mostrar al usuario final para que pueda identificar, en la fase de análisis, todas las funcionalidades que debe cumplir la aplicación y asegurarse que desarrollamos una aplicación

¹ Se pueden ver el análisis detallado en el Anexo II

que se ajusta a las necesidades realmente demandadas. En estas reuniones se incidió en la fase de detección de errores: Un profesor no puede dar dos clases a la misma hora, faltas de asistencia en días no lectivos, comprobar que la hora de la falta de asistencia del alumno se corresponda con la hora en la que el profesor pone la falta, etc. Estas decisiones fueron incluidas en el DAR.

2.2 Diseño del Sistema

El ciclo de vida utilizado para la implementación final de ambas aplicaciones corresponde a un ciclo de vida en cascada mejorado. Esto significa que las etapas por las que ha pasado la aplicación han sido: análisis de requerimientos, especificación del producto, planificación, diseño, implementación, integración y finalmente la instalación en real de la aplicación. La diferencia entre el ciclo de vida en cascada tradicional y el utilizado se basa en que en cada etapa se verificaban con el cliente los resultados y en caso de encontrarse alguna no conformidad se retrocedía a la fase anterior.

Una vez determinado el ciclo de vida, se abordó el problema en cada etapa siguiendo un método TOP-DOWN o análisis descendente. Esto implica que inicialmente abordamos el sistema planteado como único, y lo vamos dividiendo en subsistemas de forma sucesiva, hasta que alcanzamos un nivel en el cual podemos resolverlo de forma inmediata, por lo que al desarrollar satisfactoriamente cada uno de estos subsistemas e integrarlos a la aplicación, conseguiremos desarrollarla de modo adecuado.

2.2.1 Diagrama Entidad – Relación.

Se escogió MySQL como Sistema Gestor de Base de Datos, por ser gratuito y permitirnos automatizar distintas tareas de auditoría mediante triggers y procedimientos almacenados. Estas funcionalidades podrían ser utilizadas si nos exigieran satisfacer los requerimientos de la Ley Orgánica de Protección de Datos. El lenguaje de programación elegido fue JAVA [1] por ser multiplataforma y estar familiarizado con las herramientas de desarrollo. La herramienta utilizada para generar los informes ha sido Jasperreport [2], puesto que posibilita introducir parámetros de forma dinámica a las consultas y además permite incluir funciones aritméticas que serán interpretadas en tiempo de ejecución.

Las tablas utilizadas en el diseño de la Base de datos, se muestran en el siguiente diagrama:

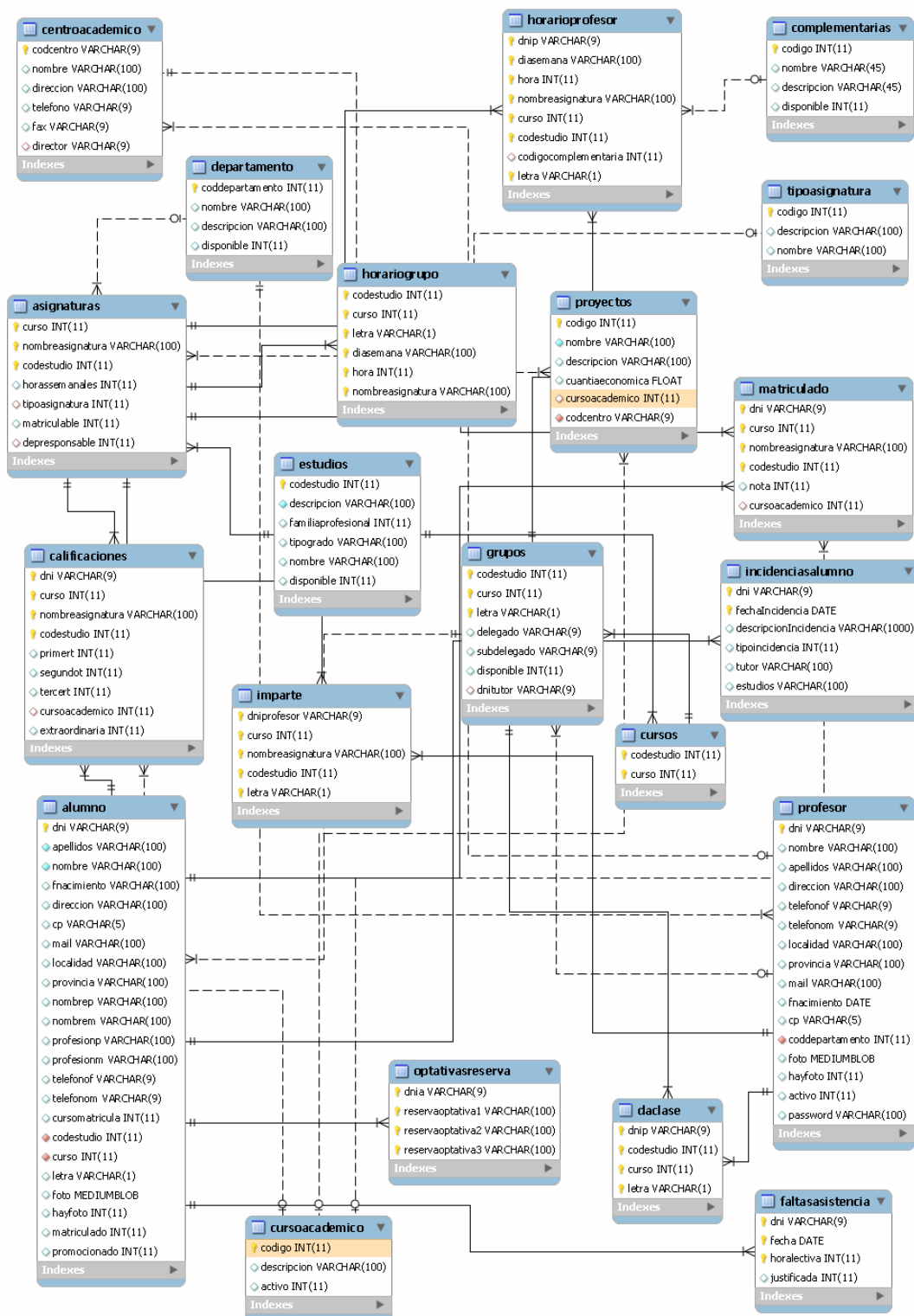


Figura 3: Diagrama Entidad-Relación

En el diagrama se muestran las tablas que intervienen en la aplicación. Entre las decisiones sobre el diseño destacaría las siguientes:

- Para introducir las fotos de los alumnos se ha utilizado un campo (foto) de tipo `mediumblob`.
- En las consultas de inserción o actualización de los datos de los alumnos, se utilizó un tipo de dato `PreparedStatement`, de este modo reducimos el tiempo de ejecución de la consulta.
- Con el objetivo de facilitar el almacenamiento de las calificaciones históricas, la tabla “matriculado” contiene todas las notas finales que ha obtenido un alumno en las distintas materias cursadas en el centro académico. Por el contrario, la tabla “calificaciones” incluye las notas, por trimestres, que ha obtenido el alumno.

Cuando se realiza la promoción de curso, la nota final obtenida (la nota de la tercera evaluación) se copia a la tabla “matriculado”. En caso de consultar los datos de los alumnos con materias pendientes de otros cursos, se efectuará la consulta sobre esta tabla, buscando los alumnos cuya calificación es inferior a cinco.

- Mediante los comandos `mysqldump` y `mysql` se puede efectuar un volcado o restauración de la Base de Datos. Estos comandos han sido automatizados mediante ficheros `sh` o `bat` dependiendo del Sistema Operativo a utilizar. Estos ficheros son invocados desde la aplicación o pueden ser ejecutados de forma automática utilizando las tareas programadas de Windows o el fichero `crontab` en Linux. El fichero resultante, de extensión `SQL`, tiene como nombre la fecha y hora de la creación de la copia de seguridad.
- Los ficheros de incidencias de los alumnos se generan en tiempo de ejecución al estar almacenados los datos en la Base de Datos. Se planteó la opción de crear una tabla con un campo de tipo `mediumblob` en la cual se guardara el fichero PDF que se crea. Esta opción se desechó al consumir mucho espacio en la Base de Datos.
- La conexión a la Base de Datos se realiza mediante `ODBC`[4], y para evitar problemas se ha incluido la opción `autoReconnect=true` que permite recuperar la conexión en caso de pérdida.

- El fichero que gestiona la Base de Datos se ha incluido en un paquete diferente, puesto que, en caso de migrar la aplicación a entorno Web, únicamente hay que importar su contenido.

2.2.2. Diagrama de Estados

Concluido el diagrama Entidad-Relación se elaboraron los diagramas de estado. Su finalidad fue determinar las ventanas a través de las cuales deben navegar los usuarios para llegar a la opción deseada. Estos diagramas fueron de gran utilidad, ya que permitieron determinar la organización de algunas ventanas de la interfaz de Usuario, evitando la creación de otras nuevas para acceder a una determinada información, (por ejemplo, el histórico de calificaciones o incidencias). El proceso de diseño de la interfaz de usuario viene explicado en el apartado técnicas utilizadas para diseñar la interfaz. En el documento de análisis del sistema se reseñan los diagramas elaborados y una explicación precisa de las distintas ventanas por las que debe navegar el usuario, así como el evento que debe dispararse para que esta transición se lleve a cabo. Cada una de estas ventanas puede agrupar varias pestañas que contienen información relacionada, con la finalidad de agrupar la mayor cantidad de funciones y de este modo minimizar el número de ventanas a utilizar.

Las ventanas que se abren pueden ser de dos tipos diferentes:

- JInternalFrame: Este tipo de ventanas nos permite mantener abiertas varias pantallas de forma simultánea. De esta forma facilitamos el manejo de la aplicación por parte del usuario (por ejemplo, puede comprobar las asignaturas optativas de reserva de un alumno en una ventana, mientras en otra puede modificar la matrícula del alumno). Al cambiar los datos de una pestaña se actualizan automáticamente todas las pestañas de esa ventana con el fin de mantener la coherencia de los datos.
- JDialog: Son ventanas que se abren de forma modal, esto significa que no se puede acceder a otra ventana hasta haber cerrado esta. Son utilizadas para mensajes de confirmación.

En el análisis de otras aplicaciones existentes en el mercado, he comprobado que todas las ventanas utilizadas son JDialog, por lo que su manejo no es sencillo al tener que anotar los datos obtenidos en una ventana para utilizarlos en otra parte de la aplicación. Además, en nuestro caso, siempre se mantiene la coherencia entre los datos mostrados en las tablas con las modificaciones realizadas, aspecto que los programas actualmente utilizados no lo hacen.

Los diagramas de clases se detallan en el DAS, en éstos se verifica la relación entre las clases implementadas en el proyecto, los métodos y atributos que las componen y las relaciones existentes entre ellas. Hay que resaltar que las clases implementadas se agrupan en dos tipos claramente diferenciados:

- Clases de diseño de interfaz: Son aquellos ficheros de tipo JDialog o JInternalFrame que componen la Interfaz de Usuario.
- Clases de procesamiento de información: Se utilizan para realizar el tratamiento de los datos, o bien para mostrar en las pantallas la información de una forma estructurada, (por ejemplo, en el caso de incluir la información en los jTable o JComboBox fue necesario implementar ficheros para dar formato a las tablas o sobrescribir el método toString).

2.3 Implementación de la aplicación

Una vez elaborados todos los diagramas² se comenzó con la implementación del proyecto. Para su desarrollo se han seguido las especificaciones indicadas en el Java Code Conventions [3] (<http://www.oracle.com/technetwork/java/codeconv-138413.html>).

La estructura del proyecto se ha dividido en paquetes que contienen: imágenes de la interfaz de usuario, ficheros jasper y jrxml (generados mediante la herramienta Jasperreport que corresponden al fichero de código y compilado) encargados de generar los archivos indicados por el usuario, el paquete de la Base de Datos, etc.

Esta organización de ficheros tiene como objetivo permitir una migración rápida de un entorno Swing [5] a otro Web. De esta forma, al copiar la estructura del proyecto, obtendremos las clases principales de la aplicación desarrollada, por lo que “únicamente” sería necesario adaptar la interfaz.

El periodo de tiempo de implementación de la aplicación se alargó más de un mes sobre la fecha de finalización prevista. Este retraso se debió a la exhaustiva fase de pruebas que se detalla en el siguiente apartado y en el documento de Plan de Aseguramiento de la Calidad del Software.

Cada vez que finalizaba un módulo de la aplicación se mostraba el resultado obtenido a los usuarios finales, quienes sugerían la información a mostrar en las tablas, modificaban los iconos de los botones o la posición de las opciones de la barra de herramientas.

² Pueden ser consultados en el DAS

Esta fase se consideró prioritaria, pues hubo que conjugar las propuestas de los usuarios que solicitaban soluciones adecuadas a sus necesidades, con las características que debía incluir la aplicación.

Igualmente se diseñó una página Web para el despliegue de la aplicación y se contrató un alojamiento Web que permitiera el acceso mediante ODBC, puesto que el inicialmente ofrecido por el Departamento no ofrecía los requerimientos que solicitábamos.

2.4 Herramientas Utilizadas

Siguiendo el esquema propuesto para la elaboración del proyecto, valoré las posibilidades que ofrecen distintas herramientas y seleccioné la más adecuada en función de los objetivos. Las consideraciones y motivos para optar por cada una, fundamentalmente son:

- En primer lugar, la interfaz de usuario fue desarrollada en SWING, ya que permitía integrar en una misma ventana más funcionalidades sin sobrecargar las pantallas. Si la aplicación se hubiera desarrollado mediante ficheros JSP, el usuario navegaría por un número mayor de páginas, lo cuál empeora la usabilidad de la aplicación.

El entorno seleccionado asegura que la aplicación se presente siempre de la misma forma, independientemente del Sistema Operativo utilizado por el usuario. Por el contrario, como existen una multitud de navegadores en el mercado, y cada uno de ellos posee sus características, resultaría prácticamente imposible mostrar las mismas pantallas en todos los navegadores³.

- Para elaborar los informes se utilizó Jasperreport a través de su entorno de desarrollo iReport. Esta herramienta ofrece una multitud de opciones; Entre ellas destacaría la posibilidad de pasar parámetros en tiempo de ejecución a la consulta que será interpretada. Esto permite utilizar un mismo fichero para diversos fines, por lo que el mantenimiento de la aplicación se reduce sustancialmente.

En comparación con otras clases que permiten la creación de ficheros en PDF (como FPDF en PHP), aporta ventajas tanto en sencillez de manejo como en rapidez en el desarrollo del informe e integración en la aplicación.

³ Se refiere a la correcta visualización de imágenes, colores y posiciones de los distintos elementos que conformarán la pantalla

- En cuanto al manejo de la Base de Datos existían dos alternativas: MySQL y Oracle. Se escogió MySQL por ser gratuita. Para la gestión de la Base de Datos se valoraron distintas herramientas de Administración: MySQL Administrator y HeidiSQL.

Entre ambas aplicaciones opté por MySQL Administrator, porque ya había trabajado con él y no existe prácticamente diferencia entre usar una herramienta u otra. Con esta aplicación planifiqué las copias de seguridad durante el desarrollo del programa, creé los distintos usuarios que accederán a la base de datos y asigné los permisos que tendrá cada uno de ellos sobre las tablas.

Como herramienta de trabajo para el diseño del diagrama entidad-relación, utilicé el Workbench de MySQL. Esta herramienta permite conectar la base de datos con el entorno gráfico, por lo que cualquier modificación que realizaba se veía reflejada automáticamente en las tablas. Esto permite una mayor rapidez en el desarrollo de aplicaciones, aunque en ocasiones tuve que conectarme directamente a la base de datos para realizar operaciones que no posibilitaba el entorno gráfico.

- Por otro lado, para que el usuario pudiera realizar copias de seguridad cuando quisiera se habilitó una opción desde la que se invoca a un fichero sh. Este fichero tiene el siguiente contenido:

```
fecha=`date +%d%m%y%H%M%S`.sql

/usr/local/mysql/bin/mysqldump --host=192.168.140.128 -uroot -prafa pfc > $fecha

mv $fecha CopiasSeguridadBD/
```

La primera línea deja en la variable fecha la cadena de caracteres compuesta por: día, mes, año, hora, minuto y segundo del momento en el cual se ha ejecutado el fichero y añade la extensión sql. La siguiente línea ejecuta el comando mysqldump (para crear el volcado de la base de datos) en el que se especifican por línea de comandos, la dirección IP del equipo donde esta alojada la base de datos, el usuario y la contraseña de acceso y finalmente el nombre de la base de datos. Este comando retorna las sentencias SQL (tanto de la creación de las tablas como del volcado de datos) y las deja en el fichero cuyo nombre contenga la variable fecha.

Un funcionamiento análogo tiene el fichero sh para realizar la restauración de la base de datos:

```
/usr/local/mysql/bin/mysql --host=192.168.140.128 -uroot -prafa pfc < $1
```

Donde \$1 es la ruta del fichero sql que contiene la copia de la base de datos, y el comando mysql permite realizar el volcado de las tablas (los argumentos utilizados en este comando son los mismos que en el mysqldump).

- El asistente para realizar la instalación de la aplicación ha sido generado mediante NSIS. Esta herramienta compila un fichero de extensión .nsi que se compone de varias secciones: definición de variables (nombre del ejecutable final, título de las ventanas durante el proceso de instalación, directorio de instalación, etc), especificar el número de ventanas que compondrán la instalación (en nuestro caso: acuerdo de licencia, directorio de instalación, selección de las componentes a instalar y finalización de la instalación), selección del lenguaje, etc.

Cada una de las páginas de instalación contiene el código que define el comportamiento de esa sección. Por ejemplo, para copiar las librerías y el fichero principal de la aplicación, se ejecutan las siguientes líneas:

```
Section "- Librerias"

CreateDirectory $INSTDIR\lib          --Crea el directorio de librerías
file "appframework-1.0.3.jar"        -- Copia los ficheros
...

CopyFiles $INSTDIR\a*.jar $INSTDIR\lib --Mueve los ficheros
...

Delete "appframework-1.0.3.jar"      -- Borra los ficheros copiados
...

SectionEnd;
```

- Finalmente, el fichero XML launch.jnlp, es el encargado de ejecutar la aplicación. Cada vez que se arranca la aplicación se llama a este fichero que se encarga de comprobar si existe alguna actualización disponible.

Este fichero se divide en las siguientes secciones:

- Una primera parte que contiene la información que se descargará durante el proceso de instalación de la aplicación

- Otra encargada de crear los accesos directos para inicializar la aplicación, sin necesidad de acceder a la Web desde la que se instaló la aplicación por primera vez.
- La sección de actualizaciones comprueba al inicializar la aplicación la existencia de nuevas actualizaciones. Si hay nuevas actualizaciones las descarga al estar los ficheros jar firmados digitalmente. Esto garantiza utilizar siempre la última versión.
- Definición de las librerías a descargar para ejecutar la aplicación y definir el fichero principal de la aplicación.

A continuación se muestra una sección de código del fichero para comprobar las distintas secciones en las que se divide:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>

<jnlp codebase="http://www.catedu.es/rychuesca" href="launch.jnlp" spec="1.0+">

  <information>

    <title>Serv. Provincial</title>

    <vendor>Unidad de Programas Educativos (Huesca)</vendor>

    <homepage href="http://www.catedu.es/rychuesca"/>

    <description>Incidencias o sugerencias: girhuesca@aragon.es</description>

    <description kind="short">Serv. Provincial</description>

    <shortcut>

      <desktop/>

      <menu submenu="Servicio Provincial"/>

    </shortcut>

    <icon href="logo.png" kind="splash"/>

    <icon href="logo.png" kind="default"/>

  </information>

  <update check="always"/>

  <security>

    <all-permissions/>

  </security>
```

```

<resources>

    <j2se version="1.5+" />

    <jar href="Educacion.jar" main="true" />

    <jar href="lib/appframework-1.0.3.jar" />

    ...

</resources>

    <application-desc main-class="educacion.VentanaMenu">

    </application-desc>

</jnlp>

```

2.5 Distribución temporal del proyecto

Antes de iniciar el proyecto realicé una planificación de tiempos para adecuarme al plazo de entrega señalado. En el siguiente diagrama GANTT⁴ se puede ver la estimación:

Nombre	Fecha de inicio	Fecha de fin
Proyecto GIE	4/07/11	9/06/12
Especificación del Producto	4/07/11	13/07/11
Revisión Acuerdo de Proyecto	13/07/11	15/07/11
Entrega del documento Acuerdo de Proyecto	18/07/11	19/07/11
Análisis	20/07/11	31/12/11
Requisitos de Usuario	20/07/11	30/07/11
Documento de Análisis y Revisiones	1/08/11	20/08/11
Documento Análisis de Sistema	1/09/11	31/12/11
Diagrama Entidad Relación	1/09/11	8/10/11
Diagrama de Clases	14/10/11	7/12/11
Diagrama de Estados	12/12/11	31/12/11
Prototipo Interfaz	12/12/11	31/12/11
Implementación	16/01/12	25/05/12
Base de Datos	16/01/12	28/01/12
Interfaz	25/01/12	12/05/12
Sistema	25/01/12	25/05/12
Pruebas	16/02/12	25/05/12
Unitarias	16/02/12	25/05/12
Integración	16/02/12	25/05/12
Documento Pruebas	16/02/12	25/05/12
Documentación	18/05/12	9/06/12
Manual de Usuario	18/05/12	26/05/12
Manual de Test	18/05/12	30/05/12
Documento de Clases	1/06/12	9/06/12

Figura 4: Diagrama de GANTT

Durante el análisis, diseño e implementación de la aplicación no se ha producido un retraso significativo en ninguna de las tareas. Esto es debido a que todas fueron planificadas con un amplio margen de tiempo, y al hecho de empezar el proyecto en los meses de verano del curso pasado.

El motivo principal de comenzar tan pronto con la fase de especificación del producto fue porque durante el mes de julio los equipos directivos se encuentran todavía en los centros por lo

⁴ Se ha omitido el diagrama de barras por no visualizarse completo en una imagen

que me podían atender sin ningún problema. Si esta fase no la hubiera iniciado en el mes de julio tendría que haber esperado hasta mediados del mes de octubre, puesto que durante el mes de septiembre ya me advirtieron que no me podrían atender, y en este caso se habrían presentado demoras.

Igualmente planifiqué un periodo de tiempo, en los meses de diciembre y enero, para preparar los exámenes en los que no avanzaría en el proyecto.

Se puede observar en el anterior diagrama, que existen periodos de tiempo muy largos para cada una de las tareas, esto es debido a que el desarrollo del proyecto ha sido realizado mientras cursaba las asignaturas pendientes para finalizar la carrera, y además, lo he simultaneado con mis tareas profesionales.

Concretando cada una de las tareas en las que dividí el proyecto, destacaría que el diseño del diagrama entidad-relación fue uno de los que más tiempo me costó. El problema fundamental era asegurar que el diseño elegido me permitiera mantener la coherencia de todos los datos (profesores activos, alumnos promocionados, recuperar alumnos dados de baja en cursos anteriores, rematricular alumnos, etc). Esto en la mayor parte de los casos lo conseguí, pero en algunos me vi forzado a modificar, durante la implementación del sistema, el diseño de la base de datos para adaptarla a las nuevas necesidades que aparecían.

3. Plan de Aseguramiento de la calidad del Software

Durante todo el proceso se ha tenido presente la calidad⁵ del software desarrollado, como una actividad de protección y se ha comprobando la concordancia entre los requisitos y la aplicación implementada. Fundamentalmente ha englobado los siguientes aspectos:

- Control de la documentación (como refleja la primera página de los distintos documentos generados).
- Estrategia de pruebas (descrita en el siguiente apartado).
- Revisiones periódicas.

⁵ Podríamos definir calidad como una propiedad o conjunto de propiedades inherentes a una cosa, que permiten apreciarla como igual, mejor o peor que las restantes de su especie.

En cada paso del proceso podían surgir errores que inicialmente pasaran inadvertidos, y también podrían no encontrarse y verse amplificados en sucesivas fases, por este motivo las pruebas se realizaron de forma exhaustiva como se indica en el siguiente apartado.

3.1 Pruebas del Sistema

La fase de pruebas se ha desarrollado a lo largo de todo el ciclo de vida del producto software: desde la captura de requisitos, especificando claramente cada uno de ellos, hasta la instalación de la aplicación en el entorno del cliente. Las pruebas realizadas se dividen en tres clases:

- Pruebas Unitarias: Comprobando el correcto funcionamiento de todos los componentes desarrollados de forma individual.
- Pruebas de Integración: Probando la integración de los componentes del sistema para demostrar que se pueden encajar correctamente.
- Pruebas de sistema: Experimentando el sistema globalmente.

En lo referente al diseño de los juegos de prueba, en todos los casos se planificaron las de tipo caja negra, esto significa que fueron diseñadas considerando exclusivamente las entradas y las salidas del sistema.

En algunos casos en los que podría existir ambigüedad (como por ejemplo en los casos de las calificaciones o los horarios de los grupos), se optó por realizar el conjunto de pruebas de caja blanca.

Estos conjuntos de pruebas fueron diseñados y testeados por mí, pero la persona que implementa una aplicación nunca debe ser la que realice el juego de pruebas. Es por este motivo que conforme se finalizaban los módulos, éstos se distribuían a los usuarios finales que testeaban la aplicación.

3.2 Diseño de la Interfaz

Un buen diseño de la interfaz no asegura el éxito de la aplicación, pero una interfaz de usuario mal diseñada conlleva el fracaso del sistema realizado, por lo que considero imprescindible el análisis de las interfaces de usuario. Esto requiere el análisis de la Interfaz de Usuario como un elemento más dentro de la calidad del software.

Como se comentó en la introducción, durante aproximadamente dos semanas se observó la metodología de trabajo de los futuros usuarios del sistema. Esto se realizó con un doble propósito:

Comprender el objeto/misión de la aplicación a desarrollar y observar al usuario final de forma indirecta para obtener información acerca de la destreza adquirida en el manejo de las Tecnologías de la Información y la Comunicación.

El grupo de usuarios con el que tuve contacto era heterogéneo tanto en el rango de edad como en la habilidad de manejar los equipos informáticos. Este grupo de usuarios me sirvió para realizar un análisis etnográfico del cual obtuve las siguientes conclusiones:

- Usuarios experimentados:
 - No solicitan Manual de Usuario
 - Indiferencia en cuanto al idioma de la aplicación
 - Solicitan que todos las opciones incluyan imágenes descriptivas
 - Priorizan la rapidez de la aplicación.
- Usuarios con nivel básicos:
 - Solicitan manual de usuario explicativo con imágenes.
 - Los textos deben ser en castellano.
 - Priorizan que el manejo de la aplicación sea sencillo e intuitivo por encima de la rapidez.

En vista de los resultados decidí realizar una interfaz de usuario que combinara imágenes y texto en todas las opciones, una barra de herramientas a modo de acceso rápido para las opciones más comunes y proporcionar atajos de teclado a los usuarios con más destreza. El tipo y tamaño de letra son de tamaño medio para permitir su lectura de forma sencilla.

3. Instalación

El desarrollo de ambas aplicaciones ha sido desarrollado en Java fundamentalmente por los siguientes motivos:

- Conocimiento del lenguaje de programación.
- Multiplataforma

- Rápida adaptación a posibles modificaciones/ampliaciones (Por ejemplo, permitir la matricula de los alumnos vía Web mediante el uso de ficheros jsp[6]).

Puesto que la tecnología Java permite realizar diferentes tipos de instalaciones, se ha utilizado el proceso de instalación más adecuado en cada caso.

En primer lugar, las aplicaciones que van a tener instaladas los usuarios administradores utilizan un proceso de instalación guiado mediante un asistente. Este asistente lo he generado mediante la aplicación NSIS[7]. Una vez que el usuario hace doble clic sobre el ejecutable se inicializa un asistente que les orientará durante todo el proceso. A continuación se muestra un ejemplo del proceso de instalación.

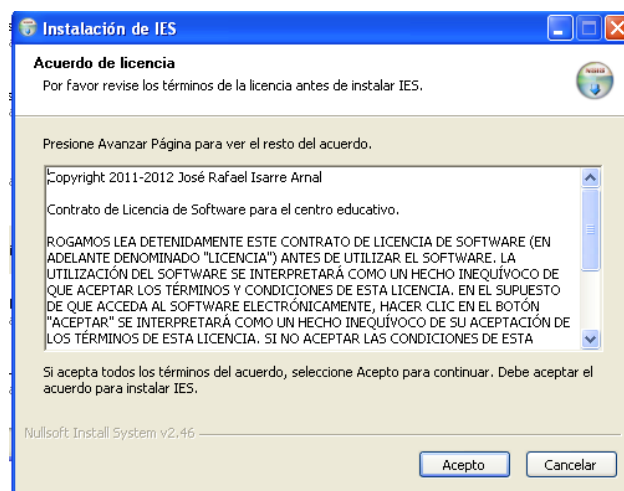


Figura 5 : Primera pantalla de instalación

En la imagen anterior se comprueba que al arrancar el programa se muestra el acuerdo de licencia para que sea leído por el usuario. Si no se acepta esta licencia no se completa la instalación de la aplicación. Una vez aceptado el acuerdo de licencia escogeremos el directorio de destino de la aplicación. Esta ventana nos mostrará el tamaño de la aplicación a instalar y el espacio disponible en la unidad de disco en la que instalaremos la aplicación.

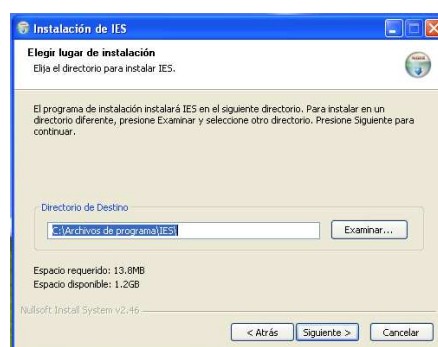


Figura 6: Directorio de Instalación

Una vez efectuados estos pasos solo resta finalizar la instalación de la aplicación, para ello seleccionaremos los componentes a instalar (en este caso escogeremos todos los indicados), escogeremos el directorio dentro del menú inicio donde figurará la aplicación y esperaremos a que finalice el proceso. Estos tres últimos pasos aparecen en las siguientes figuras:

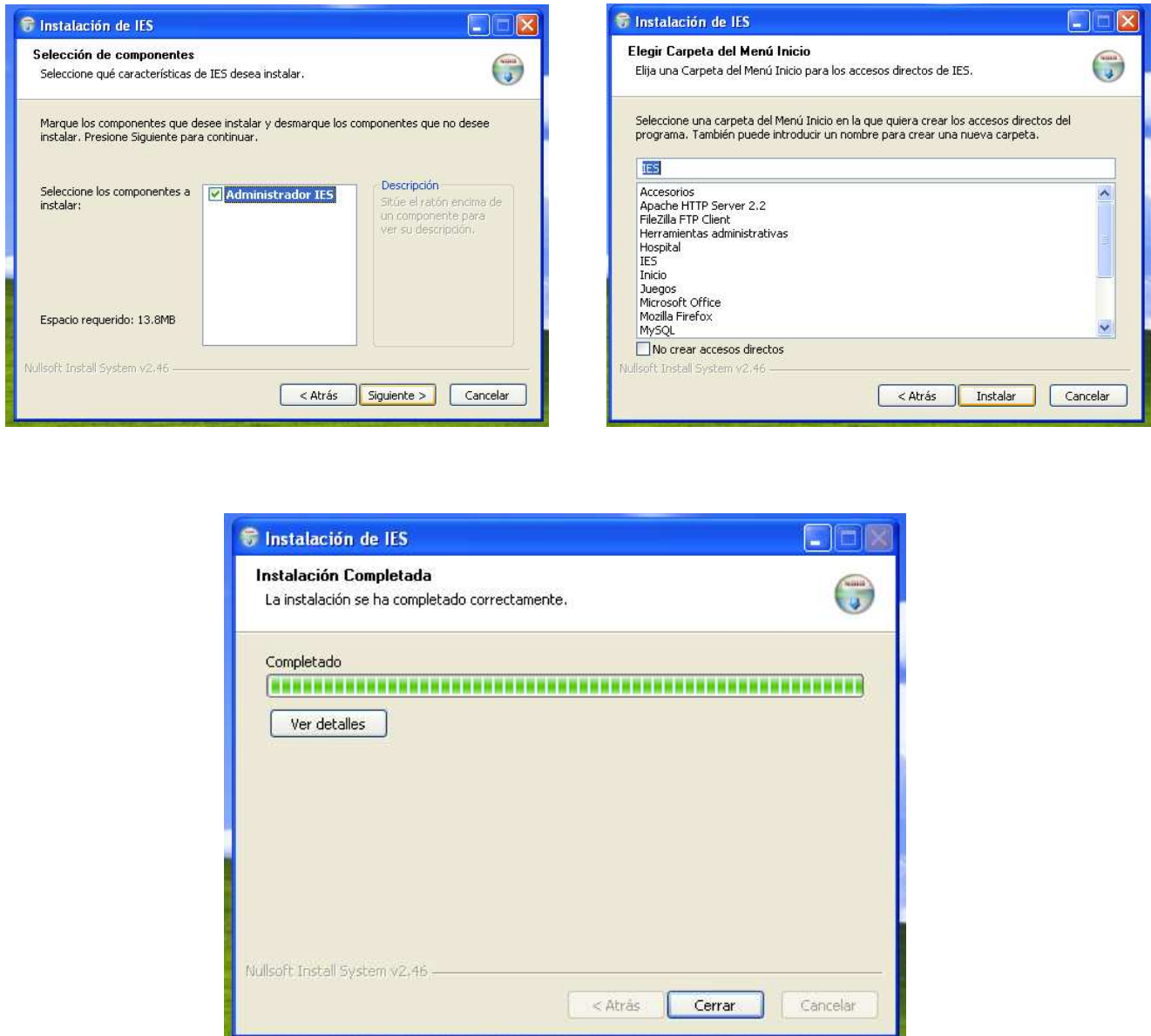


Figura 7: Instalación de la aplicación

El proceso de instalación de las aplicaciones de los terminales difiere del habitual. Mientras que las aplicaciones tradicionales se instalan mediante un asistente o wizard como acabamos de ver, la instalación de las aplicaciones de los clientes no se hace de esta forma.

Esto se debe a que en la instalación de la herramienta de gestión de un centro educativo, la aplicación estará instalada en varios ordenadores y en el caso de incluir alguna mejora se deberían revisar todos los equipos para actualizar la aplicación. Este hecho cobra más importancia en la

instalación del programa del Servicio Provincial. Esta aplicación está instalada en todos los ordenadores de los equipos directivos de la provincia, por lo que se hace inviable distribuir un instalador y controlar que todos los clientes utilicen la misma versión.

Por esta razón se ha optado para la instalación de los clientes la tecnología Java Web Start [8]. Esta permite que la aplicación se instale en el ordenador deseado mediante un fichero de extensión *.jnlp alojado en Internet. El uso de esta tecnología nos permite:

- Activar las aplicaciones con un solo clic.
- Garantizar que siempre estamos ejecutando la última versión de la aplicación.
- Eliminar complejos procedimientos de instalación o actualización.

Finalmente, en caso de realizar alguna modificación basta con subir al alojamiento Web que tenemos disponible el nuevo fichero jar que genera el Netbeans. Esto permite que cada vez que ejecutemos la aplicación se comprobará la existencia de una nueva versión como antes hemos mencionado. Esta comprobación se puede realizar, ya que el fichero va firmado electrónicamente mediante una firma digital.

En primer lugar accedemos a la página Web en la que está alojada la aplicación y nos validaremos con nuestro usuario y contraseña. Esta página es un fichero de extensión html que nos redirige a una página php en la cual se comprueba que los datos introducidos son correctos. En caso de no ser así volvemos a la página principal, pero si la comprobación de datos es satisfactoria accederemos a la página de descarga.

GOBIERNO DE ARAGON

Introduzca su usuario y contraseña de acceso:

Usuario:

Contraseña:

© Gobierno de Aragón, 2012. Todos los derechos reservados.

Figura 8: Validación del centro

GOBIERNO DE ARAGÓN
 Gestión del Servicio Provincial de Educación, Universidad, Cultura y Deporte
 UNIDAD DE PROGRAMAS EDUCATIVOS (HUESCA)
 Departamento: Educación, Universidad, Cultura y Deporte

PROCESO DE INSTALACIÓN

0.- Versión de Java Instalada
 Para saber que versión de Java tenemos instalada en el ordenador pulsa **AQUI**


1.- Descargar JRE7.
 Para saber que JRE hay que instalar puedes pulsar **AQUI**

Pulsar sobre uno de los siguientes enlaces para descargar el JRE que se corresponde con el ordenador

- jre-7u2-windows-x64.exe (Windows x64)
- jre-7u2-windows-i586.exe (Windows x86)

2.- Instalar JRE7.

3.- Instalar el programa pulsando sobre la imagen de debajo



El programa se instalará en el ordenador creando un acceso directo en el escritorio y un nuevo acceso en el menú inicio.
 La primera ejecución de la instalación puede costar unos minutos puesto que debe descargar la aplicación de Internet. Las actualizaciones del programa se realizarán automáticamente.

Figura 9: Página de descarga

Por otro lado, una vez accedamos a la página de descarga de la aplicación obtendremos una secuenciación de pasos que nos guiarán en el proceso de instalación: Identificar la versión de Java instalada, descarga de la máquina virtual, instalación de la aplicación y manuales de usuario.

Siguiendo los pasos descritos en el manual de usuario (puntos 1 y 2) tendremos actualizados todos los requisitos exigidos por la aplicación, por lo que para instalarla deberemos clicar únicamente sobre el icono del CD azul.

Al pulsar sobre la imagen se iniciará la descarga de la aplicación como se representa en la siguiente figura:

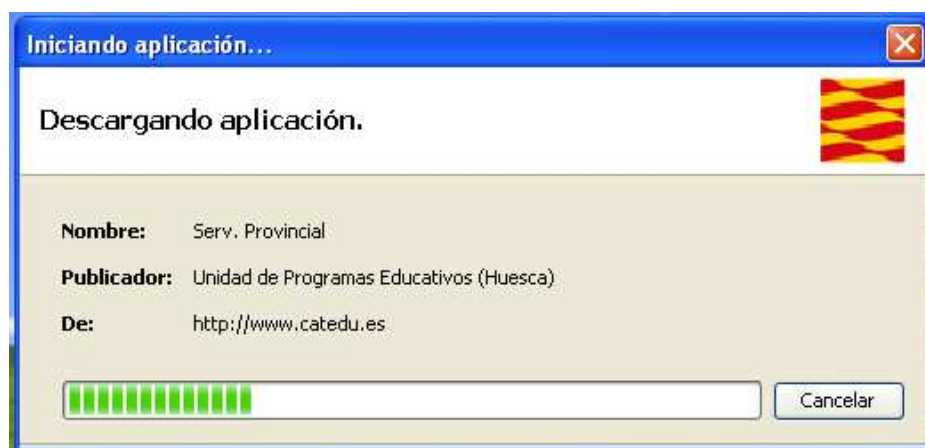


Figura 10: Descarga e Instalación

Por último, descargada la aplicación se creará un acceso directo en el escritorio y un nuevo elemento dentro del menú inicio, tal como se muestra en la siguiente imagen. De esta forma no será necesario volver a acceder a la Web.



Figura 11: Resultado del proceso de instalación

4. Resultados Obtenidos

En las siguientes páginas se muestran diversas capturas de pantalla que describen el funcionamiento real de la aplicación. Estas imágenes son solo una mínima muestra de la amplia funcionalidad de la aplicación, ya que resulta imposible mostrar todo el funcionamiento del programa en este resumen. Si se desea conocer con más detalle la aplicación (capturas, de pantalla, instalación de las aplicaciones, pruebas, etc.) se puede consultar el manual de test y el manual de usuario disponibles en los anexos.

APLICACIÓN CENTRO EDUCATIVO

Usuario Administrador:

Se muestran solo algunas de las pantallas más utilizadas por los usuarios.

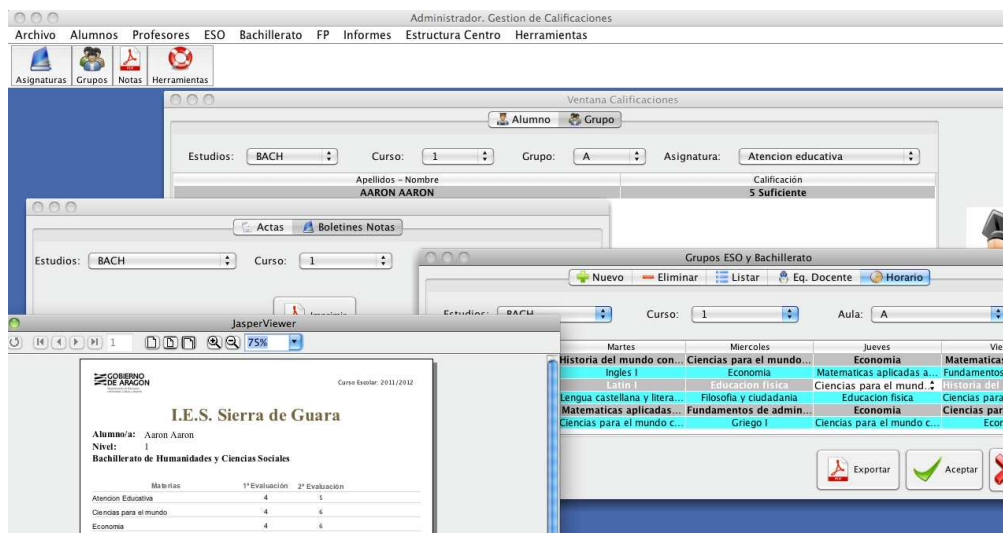


Figura 12: Boletines, Grupos y Horarios

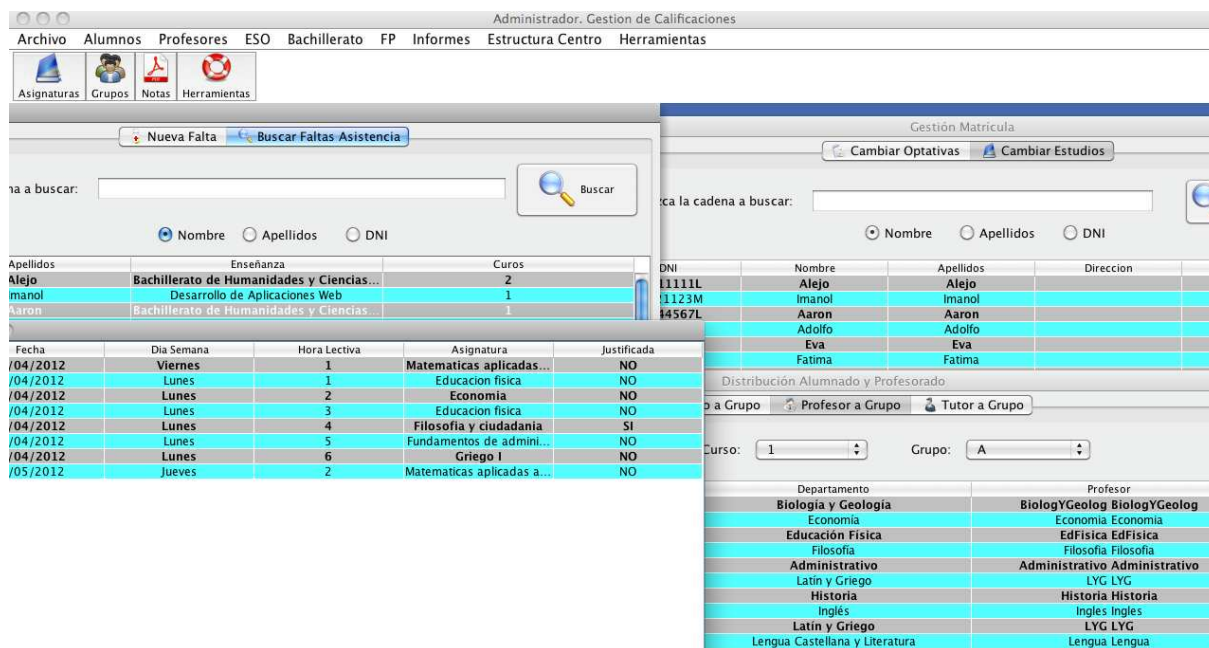


Figura 13: Matricula y faltas de asistencia de los alumnos

La figura 5 muestra un ejemplo del boletín de notas generado, la calificación de un alumno mediante su selección individual o grupo y la generación del horario de un grupo. En la Figura 6 se observa como se modifica la matrícula de un alumno o la consulta de sus faltas de asistencia.

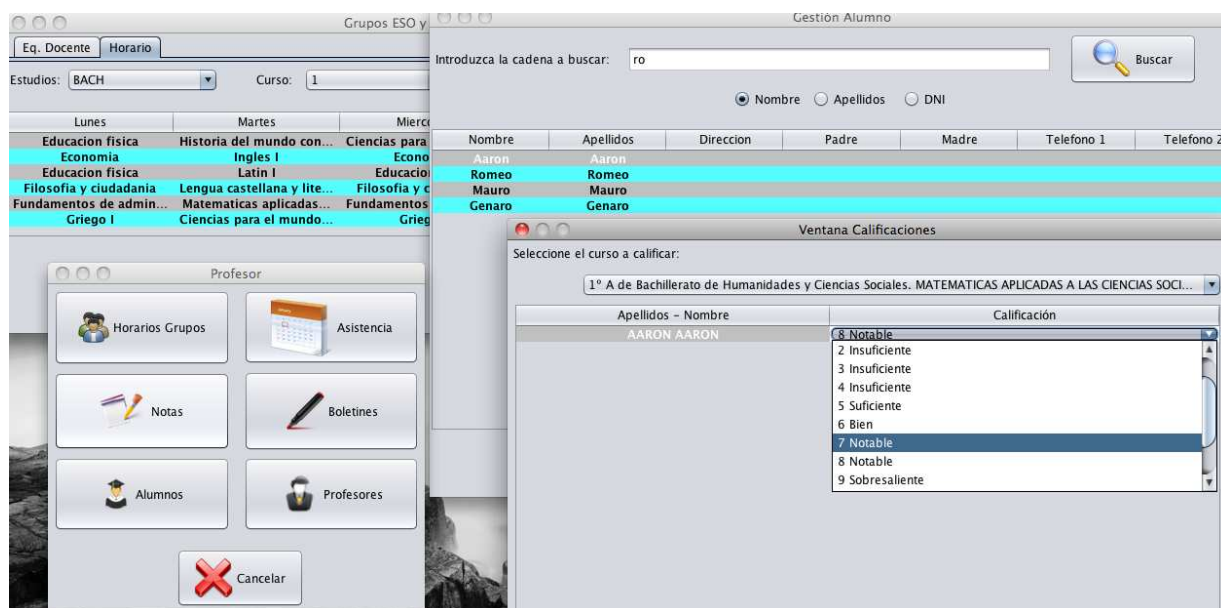
Usuario Profesor:

Figura 14: Horarios y calificaciones

En la imagen anterior se observa la ventana de calificación de alumnos. Cada profesor solo podrá valorar a los alumnos a los que imparte docencia.

APLICACIÓN DEL SERVICIO PROVINCIAL

Usuario Administrador: Se muestran solo algunas de las pantallas más relevantes

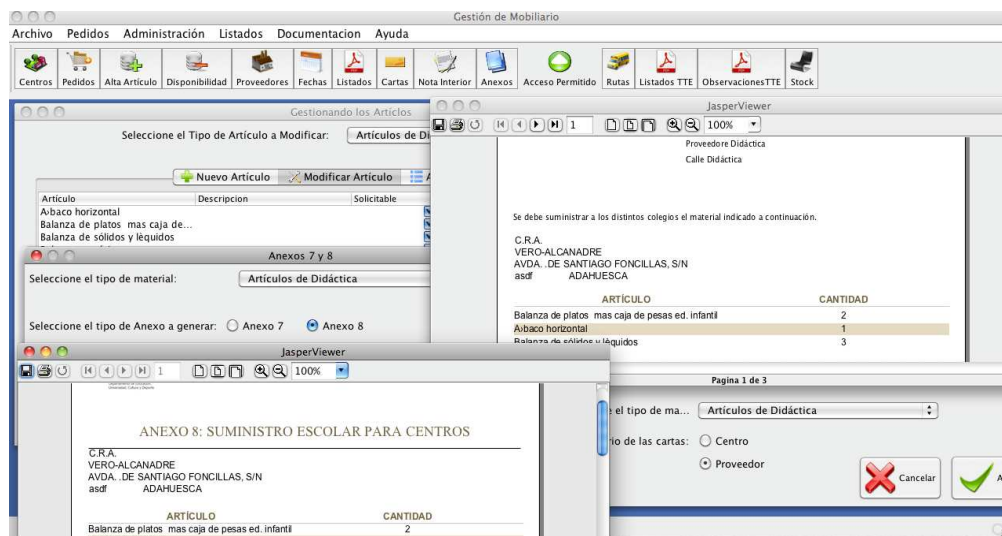


Figura 15: Ejemplo de ficheros a enviar a BOA con la relación de material solicitado

En la imagen anterior se muestra la pantalla principal de administrador de la aplicación del Servicio Provincial. Las opciones de visibilidad de un artículo para que sea solicitable y las cartas a enviar al centro/proveedor.

Usuario Centro:

ADAHUESCA (UNIDAD DE BIERGE)

Localidad o Parada	1º ESO	2º ESO	3º ESO	4º ESO	PCPI	1º BA
RODELLAR	0	2	3	4	5	6
LAS ALUMNIAS DE RODELLAR	0	2	0	0	0	0

MOBILIARIO ESCOLAR SOLICITADO POR EL CENTRO EI

ARTÍCULO	UNIDADES
A-baco horizontal	1
Armario de puertas batientes	10

Figura 16: Pantallas de introducción de datos de transporte y solicitud de mobiliario educativo

En la imagen anterior se puede comprobar el formato de las ventanas de introducción de datos para el transporte escolar o solicitud de mobiliario.

5. Conclusiones

Este PFC constituye una herramienta para mejorar los procesos de gestión escolar e impulsar un plan para reforzar la apuesta por la administración electrónica, avanzando en servicios que creen valor y en la mejora de su eficiencia. Por esta razón he insistido en la necesidad de mejorar la conectividad en los entornos de gestión mencionados: centros educativos y Servicio Provincial.

La experiencia me confirma que para gestionar un centro educativo se requieren cambios estructurales, organizativos, tecnológicos y resulta imprescindible planificar, diseñar y prever necesidades; así como valorar la relación coste-beneficio.

Para realizar esta aplicación software que considero útil y viable y lograr el mayor nivel de concreción he tenido como referente los objetivos antes enumerados. He pasado por las distintas etapas de planificación, implantación y evaluación del proceso, incorporando criterios variables en función de cada apartado.

No obstante, dada la estrecha relación que mantienen unos aspectos con otros y para evitar el riesgo de descoordinación de los usuarios se evidencia: la necesidad de formar usuarios avanzados, acometer cambios organizativos en los elementos funcionales y materiales, dotar de fórmulas para compartir las herramientas disponibles y prever la digitalización de la información.

El proceso seguido ha sido enriquecedor, los resultados me han confirmado que era necesario integrar los procesos y métodos en la gestión administrativa de manera global y justifican la incorporación de la aplicación en los centros para dar respuesta a una necesidad coyuntural.

La valoración que hago de este TFC es positiva, pues me ha permitido desarrollar una aplicación completa: desde un simple análisis de requisitos a entregar un proyecto a un cliente.

En esta aplicación he seguido el ciclo de vida del producto software: manteniendo reuniones, realizando la documentación, asegurando la calidad, etc. Esto me ha permitido conjugar teoría y práctica para comprobar que las estimaciones realizadas inicialmente pueden verse alteradas en cualquier momento y debemos adaptarnos a las nuevas necesidades.

Este trabajo ha sido un medio de aprendizaje para afrontar proyectos desde otros contextos y propongo su retroalimentación a lo largo de la propia implantación, pues en cualquier proyecto es indispensable evaluar las posibles mejoras y conseguir no solo enriquecer la aplicación sino que sirva como medio de aprendizaje.

6. Referencias

[1] The Java Language Specification. Java S.E. 7 Edition

James Gosling (<http://docs.oracle.com/javase/specs/>)

[2] Herramienta de creación de informes disponible en: <http://jasperforge.org/>

[3] Guía de estilo que realiza una serie de recomendaciones para hacer notaciones sobre clases, métodos y variables. Puede ser consultada en www.oracle.com/technetwork/java/codeconv-138413.html

[4] Estandar de conexión a Bases de Datos. Se ha utilizado para la conexión el driver disponible en: <http://dev.mysql.com/downloads/connector/odbc/>

[5] Herramienta gráfica para Java. Incluye elementos para el desarrollo de la interfaz gráfica como botones, tablas, listas desplegables, etc.

[6] Tecnología Java que permite generar contenido dinámico para Webs en forma de documentos HTML.

[7] NSIS: Manejador de scripts para realizar instaladores

[8] Permite descargar e instalar aplicaciones Java desde la Web.