



UNIVERSIDAD CARLOS III COLMENAREJO

INGENIERÍA TÉCNICA EN INFORMÁTICA DE GESTIÓN

PROYECTO FIN DE CARRERA

GESTIÓN DE LA FORMACIÓN APLICANDO MÉTRICA V3

Autor: Enrique Jiménez Montes

Director: Eduardo Muñoz Muñoz

Tutor: Luis García Sanchez

Colmenarejo, Diciembre de 2009



AGRADECIMIENTOS

Quiero agradecer a todas aquellas personas, que con las que sin su ayuda, y su apoyo, en los difíciles momentos, no podría haber llevado a cabo la realización de este Proyecto Fin de Carrera:

- En primer lugar al director del proyecto, Eduardo, por haberme introducido en el mundo laboral, y al tutor del proyecto, Luis, sin él que no hubiera sido posible la ejecución de este proyecto.
- A mi familia, ya que sin su comprensión en los malos momentos, esto hubiera sido impensable.
- A Diana por haber creído en mí, y animarme, siempre que las cosas no salían como yo esperaba.
- A mis amigos, por haber sabido entender mis ausencias cuando me ha tocado estar estudiando.
- Y por último y no menos importante, a mis compañeros de clase, algunos amigos, que gracias a su ayuda he podido acabar bien mis estudios.

ÍNDICE

1.	INTRODUCCIÓN	8
2.	ESTADO DE LA CUESTIÓN	10
2.1.	Herramientas de gestión de RR.HH.	10
2.2.	Soluciones personalizadas y el por qué se va a utilizar vs. Soluciones integrales.....	11
3.	OBJETIVOS DEL PROYECTO	12
4.	HIPÓTESIS DE TRABAJO	13
4.1.	Entorno de trabajo.	13
4.2.	Equipo y condiciones de trabajo.....	13
4.3.	Entorno de implantación.....	13
5.	MÉTODO DE RESOLUCIÓN	14
5.1.	Plan de proyecto.	14
5.1.1.	Tarea GPI 2.2: Selecciones de tareas, planificación detallada, y asignación de recursos.	14
5.1.1.1.	Selección de tareas.....	15
5.1.1.1.1.	Fases.....	15
5.1.2.	Equipo de trabajo.....	16
5.2.	Análisis del sistema.	16
5.2.1.	Actividad ASI 1: Definición del sistema.....	16
5.2.1.1.	Tarea ASI 1.1: Determinación alcance del sistema.	16
5.2.1.2.	Tarea ASI 1.4: Identificación de los usuarios Participantes y finales.	32
5.2.2.	Actividad ASI 2: Establecimiento de requisitos.....	33
5.2.2.1.	Tarea ASI 2.1: Obtención de requisitos.....	33
5.2.2.2.	Tarea ASI 2.2: Especificación de Casos de Uso.....	34
5.2.2.3.	Tarea ASI 2.4 (ASI-SEG 1.1): Estudio de la seguridad requerida en el proceso de análisis del sistema de información.....	49
5.2.3.	Actividad ASI 6: Elaboración del modelo de datos.	49
5.2.3.1.	Tarea ASI 6.1: Elaboración del modelo conceptual de datos.	50
5.2.3.2.	Tarea ASI 6.2: Elaboración del modelo lógico de datos.	51
5.2.3.3.	Tarea ASI 6.4: Especificación de necesidades de migración de datos y carga inicial.	53
5.2.4.	Actividad ASI 7: Elaboración del modelo de procesos.....	54
5.2.4.1.	Tarea ASI 7.1: Obtención del modelo de procesos del sistema.....	55
5.2.4.2.	Tarea ASI 7.2: Especificación de interfaces con otros sistemas.....	56
5.2.5.	Actividad ASI 8: Definición de interfaces de usuario.....	56



5.2.5.1.	Tarea ASI 8.3: Especificación de formatos individuales de la interfaz de pantalla.....	56
5.3.	Documento de Diseño del sistema.....	69
5.3.1.	Actividad DSI 1: Definición de la arquitectura del sistema.....	69
5.3.1.1.	Tarea DSI 1.2: Identificación de requisitos de diseño y construcción.....	69
5.3.1.2.	Requisitos funcionales.....	70
5.3.1.2.1.	Requisitos de rendimiento.....	70
5.3.1.2.2.	Requisitos de interfaz.....	70
5.3.1.2.3.	Requisitos de verificación.....	70
5.3.1.2.4.	Requisitos de aceptación.....	70
5.3.1.2.5.	Requisitos de documentación.....	71
5.3.1.2.6.	Requisitos de portabilidad.....	71
5.3.1.3.	Tarea DSI 1.6: Especificación del entorno tecnológico.....	71
5.3.1.3.1.	Hardware.....	72
5.3.1.3.2.	Software.....	72
5.3.1.4.	Tarea DSI 1.7: Especificación de requisitos de operación y seguridad. 73	
5.3.1.4.1.	Requisitos de operación.....	73
5.3.1.5.	Tarea DSI 1.9 (DSI-SEG 2.1): Análisis de los riesgos del entorno tecnológico.....	74
5.3.2.	Actividad DSI 6: Diseño físico de datos.....	75
5.3.2.1.	Tarea DSI 6.1: Diseño del modelo físico de datos.....	75
5.3.2.2.	Tarea DSI 6.2: Especificación de los caminos de acceso a los datos y optimización.....	81
5.3.2.3.	Tarea DSI 6.3: Especificación de la distribución de datos.....	82
5.3.3.	Actividad DSI 8: Generación de especificaciones de construcción (Diseño detallado).....	83
5.3.3.1.	Tarea DSI8.1: Especificación del entorno de construcción.....	83
5.3.3.1.1.	Hardware.....	85
5.3.3.1.2.	Software.....	85
5.4.	Documento de construcción del sistema.....	85
5.4.1.	Actividad CSI 1: Preparación del entorno de generación y construcción.....	85
5.4.1.1.	Tarea CSI 1.1: Implantación de la base de datos física o fichero.....	85
5.4.1.2.	Tarea CSI 1.2: Preparación del entorno de construcción.....	86
5.4.2.	Actividad CSI 2: Generación del código de los componentes y de los procedimientos.....	86
5.4.2.1.	Tarea CSI 2.1: Generación del código de los componentes.....	86
5.5.	Manual de usuario.....	87
5.5.1.	Alta de sesiones:.....	89
5.5.1.1.	Crear nueva sesión.....	90
5.5.1.2.	Guardar cambio de sesión.....	97



5.5.1.3.	Eliminar sesión.	97
5.5.1.4.	Buscar curso / sesión.....	97
5.5.1.5.	Imprimir diplomas preimpresos.....	98
5.5.1.6.	Cerrar formulario.	98
5.5.2.	Inscripción de alumnos.....	99
5.5.3.	Alertas.....	102
5.5.4.	Calendario Intranet.	104
6.	Documento de implantación del sistema.	107
6.1.1.	Actividad IAS 1: Establecimiento del plan de implantación.....	107
6.1.1.1.	Tarea IAS 1.1: Definición del plan de implantación.	107
6.1.1.2.	Tarea IAS 1.2: Especificación del equipo de implantación.....	108
7.	CONCLUSIONES Y EXTENSIONES AL SISTEMA	110
8.	ANEXOS.....	111

**ÍNDICE DE FIGURAS**

WBS	15
D.F.D. Gestión de las sesiones	18
Ejemplo Diploma.....	23
D.E.C. Gestión de las sesiones	24
D.F.D. Gestión de los asistentes	25
D.E.C. Gestión de los asistentes	27
D.F.D. Gestión de las convocatorias	27
D.E.C. Gestión de las convocatorias	29
D.F.D. Gestión de las alarmas	30
D.E.C. Gestión de las alarmas	31
Caso de uso Gestión de las asistentes.....	34
Caso de uso Gestión de las convocatorias	40
Caso de uso Gestión de los informes de planificación	45
Caso de uso Gestión de las alarmas.....	47
Modelo conceptual de datos	50
Modelo lógico de datos	52
Modelo de procesos del sistema	55
Interfaz de pantalla menú principal	57
Interfaz de pantalla Gestión de convocatorias	58
Interfaz de pantalla Gestión de convocatorias (Otros Datos)	62
Interfaz de pantalla Gestión de convocatorias (Detalles Fechas)	64
Interfaz de pantalla Gestión de candidatos	65
Interfaz de pantalla Envío de convocatoria	67
Relaciones entre tablas	75
Figura 1: Menú principal	87
Figura 2: Gestión de Convocatorias	89
Figura 3: Envío de Convocatoria.....	91
Figura 4: Otros datos	94
Figura 5: Detalles Fechas	96
Figura 6: Búsqueda de Sesiones de Cursos	98
Figura 7: Gestión de los candidatos.....	99



Figura 8: Lista de alarmas	102
Figura 9: Calendario para Intranet.....	104
Figura 10: Calendario Intranet.....	106



ÍNDICE DE TABLAS

Gestión de sesiones de los cursos:.....	35
Buscar curso:	35
Gestión de recursos:	36
Imprimir diplomas:.....	37
Gestión de los asistentes:.....	38
Gestión de candidatos:.....	40
Selección de cursos:	41
Gestión de plantillas:	42
Gestión de anexos:.....	43
Gestión de buzones:.....	43
Selección criterios de impresión:.....	45
Generación de informes en Excel:.....	46
Selección parámetros alarmas:	47
Selección alertas disponibles:.....	48
Alarmas:	75
AlarmasDefinidas:.....	75
Aula_Interna:.....	75
Buzón:.....	76
Centro_Interno:.....	76
Convocado:.....	76
Curso:	76
Envío:	77
EstadosPeticones:	77
Formador:	77
OtrasTareasSesion:	77
PlanFormación:.....	77
Responsable:.....	78
Sesion:	78
TB_PERSONAS:	79

1. INTRODUCCIÓN

La formación en las empresas, es una de las principales herramientas para conseguir una alta productividad, e imprescindible para el buen desarrollo de cualquier empresa moderna, esencialmente para su innovación y su competitividad dentro del mercado actual.

Cada vez la Formación es una materia más exigida por las distintas áreas de una empresa como medida de mejora productiva y de eficiencia en el desarrollo de las funciones de los empleados. Por eso mismo la actividad de los departamentos de formación se ha visto incrementada considerablemente. A todo esto se une la multitud de tareas que se realizan alrededor de la organización de un curso, por lo que la eficiencia de los propios departamentos de formación es un punto de preocupación.

Por ello, el control rutinario de los cursos de formación, comprendiendo desde las tareas más simples como las reservas de aulas o el material del cual disponen, hasta otras más complejas, como requerir autobuses para el transporte o reservar determinados restaurantes para sus dietas, pueden llevarse a cabo mediante este proyecto fin de carrera, por medio de alarmas.

Las alarmas nos avisarán dependiendo de las fechas en las que nos encontremos, con niveles de prioridad. Cuanto menos tiempo quede para la celebración de un curso las alarmas se tornaran en su nivel más alto, para que los cursos que se lleven a cabo tengan todas sus tareas correctamente planificadas, desde el aula donde se impartirán, hasta los recursos de transporte si llegan a ser necesarios.

Todas estas tareas se realizan conjuntamente con la gestión de los empleados que asisten a los cursos, de los que dispondremos a parte de su información personal, toda aquella información relativa a las convocatorias de los cursos, como son:

- los empleados candidatos a realizar un curso, de los cuales veremos las cancelaciones que han hecho de un curso y si ha sido por causas justificadas o no.
- los empleados convocados a realizar un curso, serán los usuarios finales que llegarán a realizar un curso, que previamente han confirmado la futura asistencia al curso.

Por tanto con la gestión de estas tareas agilizamos aún más la labor del departamento de recursos humanos, en cuanto a sus planes de formación se refiere. El desarrollo del proyecto se ha realizado mediante la metodología Métrica 3.

La metodología Métrica, creada en el año 1989 por el Ministerio de Administraciones Públicas, ofrece un instrumento para la sistematización de las actividades, que dan soporte al ciclo de vida del software, que define Sistemas de Información para satisfacer las necesidades de los usuarios, facilitando la operación de mantenimiento y uso de los productos software obtenidos.



Las versiones (2 en el año 1993, 2.1 en el año 1995, 3 en el año 1998) no han conseguido ser un estándar para la industria del software español, porque es una metodología que incorpora demasiadas tareas a realizar, todas ellas enfocadas desde un punto de vista muy teórico, y de la que no encontramos ningún caso práctico de cómo aplicarla.

Como incorporaciones ventajosas podemos distinguir tres apartados que nos han hecho decantarnos por esta metodología y no otras:

- la integración del diseño de las pruebas, en todas las fases del proyecto: de manera que cuando se realiza la planificación, análisis, diseño y programación, se establecen los ciclos de pruebas que se necesitan realizar en cada fase del proyecto.
- la integración de la gestión de la calidad de los productos obtenidos: comprobando la calidad del producto al finalizar cada fase del proyecto, teniendo que aprobarse, para que siga el buen funcionamiento del proyecto
- el mantenimiento de las aplicaciones: la tarea más descuidada por otras metodologías, que marca la gestión de cambios en el producto a lo largo de su ciclo de vida.

2. ESTADO DE LA CUESTIÓN

Dentro de las posibilidades que podemos encontrar para poder realizar nuestra aplicación tenemos dos diferentes caminos a seguir:

- Utilizar una solución integral como PeopleSoft, HR Access o SAP R/3.
- O utilizar una solución a medida, creada desde cero por nosotros mismos.

Analizaremos cual de las dos soluciones es mejor, si desde un punto de vista de solución integral, o desde un punto de vista de solución a medida.

2.1. Herramientas de gestión de RR.HH.

En primer lugar tomaremos como referencia algunas de las aplicaciones con las que podíamos haber intentado hacer el proyecto fin de carrera:

- PeopleSoft:

PeopleSoft, Inc. es una compañía que suministra software de Planificación de Recursos Empresariales (E.R.P. - Enterprise Resource Planning), gestión de Recursos Humanos, gestión de las Relaciones con los Clientes (CRM, Customer Relationship Management) y Gestión de Nómina a grandes empresas.

Fundada en 1987 por Dave Duffield y Ken Morris, y con sede en Pleasanton, California, los inicios de PeopleSoft comienzan con una idea de Duffield sobre la versión Cliente-Servidor (en aquel entonces un concepto nuevo) de Integral, un paquete de gestión de recursos humanos popular para grandes arquitecturas. Cuando Integral declinó seguir desarrollando la aplicación, permitió a Duffield lograr su sueño, PeopleSoft había nacido. En enero de 2005, PeopleSoft fue adquirida por Oracle Corporation y dejó de ser una compañía independiente, aunque su línea de productos sigue.

- HR Access:

HR Access fue fundada en 1973. Se basan en investigación, desarrollo e implementación en la gestión de los recursos humanos y de la nómina.

Diseña, desarrolla y comercializa productos y servicios previstos, ya sea para su procesamiento interno o para la externalización. Fidelity Investments ha adquirido las actividades del paquete de programas de HR Access a IBM en octubre de 2003.

HR Access ofrece un conjunto completo de soluciones adaptadas a las particularidades de un mercado global. Gestionan todos los aspectos de la administración del personal, de la gestión de tiempos y actividades, de la nómina

y de la gestión de talentos mediante una solución completamente integrada basada en una única base de datos.

▪ SAP R/3:

SAP R/3 es un ERP (Enterprise Resource Planning) de origen alemán, creado por SAP. Es un sistema integrado de gestión que permite controlar todos los procesos que se llevan a cabo en una empresa, a través de módulos:

- Finanzas.
- Ventas y Distribución.
- Almacenes e Inventarios.
- Producción.
- Recursos Humanos.
- Tecnología.
- Soluciones específicas por industria.

Además de estas soluciones estándares, el ambiente de desarrollo de SAP y su sistema de información, proveen a los clientes con poderosas herramientas para desarrollo y adaptación del sistema a los requerimientos individuales (personalización). El ambiente de desarrollo del sistema R/3 ofrece a los usuarios su propio lenguaje de programación de cuarta generación (ABAP/4), creado especialmente para las necesidades comerciales.

El amplio rango de servicios que ofrece el sistema, sin embargo, es solamente una de las causas del éxito del sistema R/3. SAP soporta el concepto de sistema abierto, construcción de interfaces (GUIs), servicios, sobre los actuales estándares.

El sistema SAP R/3 es un sistema integrado. Esto significa que una vez que la información es almacenada, está disponible a través de todo el sistema, facilitando el proceso de transacciones y el manejo de información.

2.2. Soluciones personalizadas y el por qué se va a utilizar vs. Soluciones integrales.

Los grandes sistemas de gestión de RR.HH. no solucionan las necesidades específicas que surgen en nuestro proyecto fin de carrera, porque no consiguen alcanzar el nivel de detalle que queremos abarcar.

Con estas otras soluciones como PeopleSoft o SAP solo nos quedaríamos en el detalle más general, sin poder conseguir especificaciones más concretas por falta de tiempo y de coste.

Por ello, nuestro proyecto fin de carrera abarca el nivel de detalle más bajo que queremos conseguir de una manera fácil y directa, consiguiendo solucionar todas las pequeñas funcionalidades que los otros sistemas solo pueden atacarlas de manera más general.

3. OBJETIVOS DEL PROYECTO

El presente proyecto fin de carrera surge como una necesidad dentro de la gestión de los cursos formativos, ya que con otros sistemas de gestión de RR.HH. no hubiera sido posible llegar al nivel de detalle más bajo que queremos abarcar, en lo que se refiere a la gestión de los empleados de un determinado curso, consiguiendo como principales objetivos:

- Reducir el tiempo de gestión dentro del departamento de Formación y así agilizar sus tareas dedicadas a la formación:
 - Logística de los cursos
 - Planificación
 - Gestión de la convocatoria
 - Control del gasto

- Disminuir por medio de alarmas el tiempo empleado en la gestión de la preparación de los cursos que impartirá el departamento de RR.HH. Estas alarmas nos avisarán por medio de prioridades, de todos los aspectos necesarios para la celebración de un curso. Cuanto menos tiempo quede para la celebración de un curso, más alarmas se activaran en el caso de que no estén evaluadas.

- Elaborar informes mensuales que sirvan para el control de los distintos cursos que se celebrarán: desde el departamento de R.R.H.H, se podrá gestionar la planificación de los siguientes cursos que se van a celebrar, informándonos de todos los datos de los cursos que se van a desarrollar. También se realizara un control de los costes asociados a los cursos que se van a celebrar, como por ejemplo de transporte si lo hubiese, material o diversos recursos.

- La utilización de la metodología Métrica versión 3, que es muy robusta, pero poco utilizada, la cual nos lleva a crear de manera más sintetizada y centrándonos solo en las fases principales de esta metodología, esta aplicación de gestión de la formación.

4. HIPÓTESIS DE TRABAJO

4.1. Entorno de trabajo.

Para el desarrollo de esta aplicación se ha utilizado un entorno de trabajo visual y más concretamente Visual Basic para aplicaciones, dentro de Microsoft Access 2003.

La parte referente a la programación ha sido orientada a eventos, implementada desde Microsoft Visual Basic para aplicaciones Access, con la que se han podido desarrollar todas las tareas necesarias.

Aunque el entorno de trabajo ha sido realizado en Microsoft Access 2003, internamente se ejecutan Microsoft Outlook 2003, en el apartado de gestión de las convocatorias de los cursos (“Enviados_F”) y Microsoft Excel 2003 para crear los informes de planificación de los cursos (“Planificación”). Por tanto se ha interactuado con los diferentes aplicativos que contiene el paquete Microsoft Office 2003, todos ellos configurados y ejecutados directamente desde Microsoft Access 2003.

Se han elegido estas aplicaciones ofimáticas para trabajar sobre ellas por su elevada portabilidad, todos los ordenadores del departamento disponen de Microsoft Office 2003, y en el caso de que apareciesen nuevas máquinas no habría ninguna complicación en instalarlo en ellas. También porque ofrecen una gran facilidad para su configuración, no teniendo que modificar ningún parámetro para su buen funcionamiento. Otra de las razones por la cual se ha escogido como entorno de trabajo Microsoft Access, es porque la aplicación va a ser utilizada por un número reducido de usuarios y la cantidad de información que va a contener, no es muy importante.

4.2. Equipo y condiciones de trabajo.

El equipo de trabajo lo he formado junto con Eduardo Muñoz Muñoz (Gerente de Formación de Áreas de Apoyo) y Juan Carlos Urbina (Escuela de Tecnología).

Las diferentes pruebas para comprobar que la aplicación se iba desarrollando correctamente y funcionando de manera óptima, se han realizado en las oficinas centrales de un departamento de RR.HH., en Madrid.

4.3. Entorno de implantación.

Los pc's en los que se ha implantado esta solución de la gestión de la formación han sido sistemas con Windows Xp Professional, y con la suite ofimática Microsoft Office 2003. Al ya estar en estos sistemas previamente instalados, no ha hecho falta alterar la configuración, ejecutándose en la red corporativa correctamente y sin ningún problema.

5. MÉTODO DE RESOLUCIÓN

5.1. Plan de proyecto.

5.1.1. Tarea GPI 2.2: Selecciones de tareas, planificación detallada, y asignación de recursos.

Productos

De entrada

- Planificación General del Proyecto (GPI 2.1):

Ya que nuestro proyecto es orientado a eventos, se ha decidido seguir la estrategia por prototipo. El prototipo se va completando en sucesivas evaluaciones y revisiones, añadiendo nuevas funcionalidades y mejoras, hasta cubrir los requisitos completamente.

De salida

- Planificación General del Proyecto:

- Estructura de actividades:

Nuestro proyecto de gestión de los cursos de formación consta principalmente de cinco grandes bloques:

- Gestión de las sesiones de los cursos.
- Gestión de los asistentes.
- Envío de convocatorias.
- Planificación de los cursos.
- Alarmas.

- Catálogo de productos a generar:

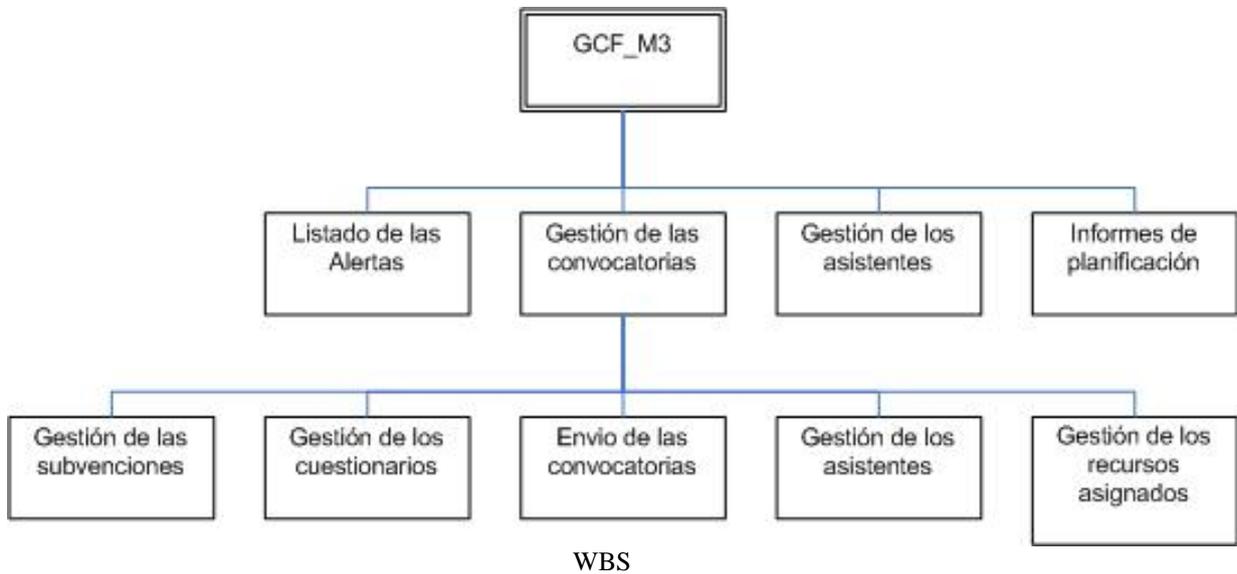
Para cada una de las tres principales actividades se ha generado un formulario de aplicación:

- Gestión de las sesiones de los cursos: Formulario Sesiones.
- Gestión de los asistentes: Formulario Asistentes
- Envío de convocatorias: Formulario Enviados_F.
- Planificación de los cursos: Formulario Planificación.
- Alarmas: Formulario ListaDeAlarmas

Técnicas

· Planificación:

- Estructura de Descomposición de Trabajo (WBS):



Prácticas

· Catalogación

Participantes

· Jefe de Proyecto

5.1.1.1. Selección de tareas.

Las tareas principales que hemos tenido que desarrollar, han sido, la creación de los distintos formularios tanto el principal de “Sesiones” como los subformularios que dependen de él, uniéndolos conjuntamente como una sola parte.

5.1.1.1.1. Fases

En una primera fase se ha desarrollado las sesiones de los cursos, y a partir de ahí, se han ido integrando el resto de las tareas: gestión de los asistentes, envío de convocatorias, planificación de los cursos y alarmas, comprobando cada tarea individualmente, junto con el resto del proyecto.

En una segunda fase se han completado de integrar las distintas funcionalidades, trabajando con datos de prueba, y viendo la coherencia y la unicidad de los mismos.

5.1.2. Equipo de trabajo.

El equipo de trabajo ha estado formado por el Jefe de Proyecto (Eduardo Muñoz Muñoz) y los Analistas y Desarrolladores del Proyecto.

5.2. Análisis del sistema.

5.2.1. Actividad ASI 1: Definición del sistema.

Nuestro sistema deberá proporcionar toda aquella información relativa a la sesión de un curso, desde los asistentes que tendrá esa sesión, como los requisitos logísticos que han de darse, para la celebración del curso. También nos deberá mantener informados, acerca de todas aquellas posibles alarmas que se produzcan cuando se acerque la celebración de un curso.

Será ejecutado únicamente en la red local de nuestra empresa, en modo monousuario, de manera que si alguien más ejecutase la aplicación se bloquearían todas las tablas utilizadas, hasta el momento en el que deje de usarlas.

Podrán utilizarlo todos los usuarios del departamento de RR.HH., que se encarguen de la formación de los empleados en la empresa.

5.2.1.1. Tarea ASI 1.1: Determinación alcance del sistema.

Productos

De entrada

- Descripción de la Solución (EVS 6.2)

El sistema será capaz de dar de alta, baja o modificar cualquier tipo de sesión perteneciente a un curso, así como de gestionar sus asistentes, enviar convocatorias automáticamente y planificar los informes de seguimiento de los cursos que se celebraran en un mes concreto.

- Catálogo de requisitos (EVS 6.2)

La solución deberá funcionar para un grupo cerrado de usuarios del departamento de RR.HH., para ello estará siempre disponible para su uso.

Dentro de sus opciones se podrá ejecutar las siguientes características:

- Gestión de las sesiones celebradas de un curso.
- Gestión de los asistentes.



- Gestión de las convocatorias enviadas al personal que asistirá a los cursos.
- Gestión de los informes de planificación de los cursos.
- Por último y como más importante, gestión de las alarmas para la correcta planificación y ejecución de los cursos disponibles en nuestro catalogo.

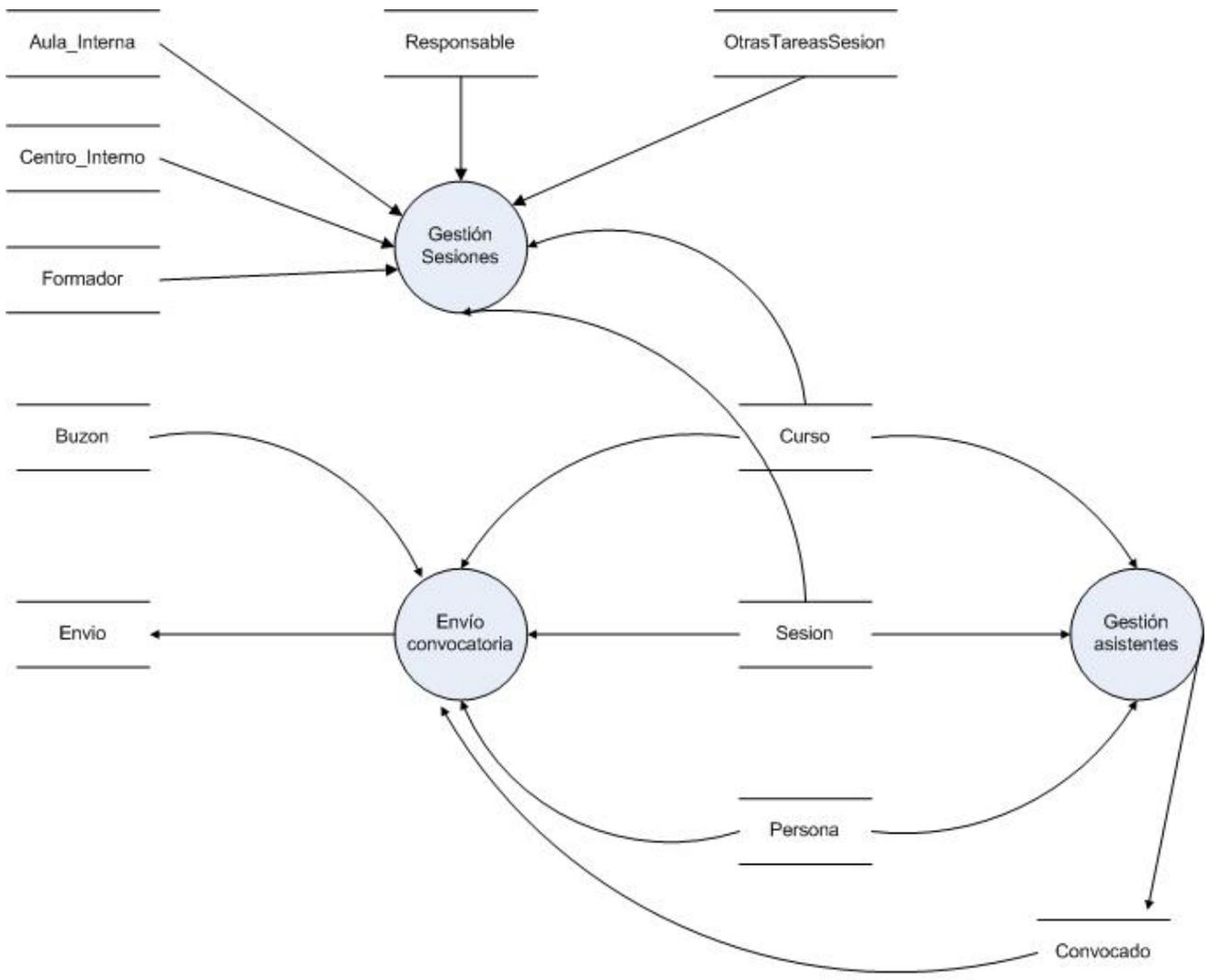
De salida

- Catálogo de Requisitos
- Glosario

Técnicas

- Diagrama de Flujo de Datos

1º- Gestión de las sesiones:



D.F.D. Gestión de las sesiones

Para cada una de las sesiones de un curso que tengamos podemos realizar las siguientes acciones:

- Crear una nueva sesión (Alta de sesiones).
- Guardar una sesión (Modificación de sesiones).
- Eliminar una sesión (Baja de sesiones).
- Buscar un curso / sesión.
- Imprimir diplomas preimpresos.
- Salir del formulario (Salir al menú principal).

Alta / Modificación de sesiones:

Si queremos dar de alta o modificar una sesión, dentro del formulario de Gestión de las Convocatorias podemos encontrar 3 funcionalidades principales que deberemos rellenar / modificar:

- Datos identificativos de la sesión.
- Datos generales.
- Otros datos.
- Detalles fechas.

Datos identificativos de la sesión:

El formulario de datos identificativos de la sesión contiene un campo nombre del curso, el cual desplegará una lista de los cursos disponibles al pulsar sobre él. A su lado mostrará un botón con la lista de cursos disponibles y el código asociado a cada curso.

Tanto la fecha de inicio como la fecha de finalización del curso desplegarán un control de tipo calendario, en el que quedará reflejado visualmente el día en el mes en el que nos encontramos pudiendo avanzar o retroceder sobre los meses. La fecha de finalización del curso no podrá ser nunca anterior a la fecha de inicio del curso.

El campo clave estará formado por una máscara de 6 dígitos correspondientes al código del curso y otros 4 correspondientes al código de la sesión. Por otra parte el número de horas que tiene el curso también tendrá una máscara de entrada de 3 dígitos.

Los campos asistentes mínimos y asistentes máximos tendrán una máscara de entrada de 3 dígitos, y el número de asistentes mínimos no podrá ser superior al número de asistentes máximos y viceversa, siempre hablando de la misma sesión de un curso.

Referencias cruzadas:

Podremos acceder a la funcionalidad gestión de asistentes dentro de este apartado, controlando desde ella las personas que formarán finalmente el listado de personas que acudirán a la siguiente celebración del curso. Una vez seleccionemos los candidatos convocados, como los candidatos confirmados, se actualizarán automáticamente los campos acuden y sin conf., del formulario gestión de las convocatorias.

Datos generales:

Podemos acceder a la funcionalidad gestión del envío de convocatorias. Para ello pulsaremos sobre el botón convocar al curso, seleccionando las personas a las que se las mandará el correo con los datos de la convocatoria.

Dentro de este formulario podremos seleccionar si la convocatoria ha sido enviada y cuando se ha realizado el envío, para ello se mostrará un control de calendario, mostrando el día del mes en el que estamos, por el que nos podemos mover, tanto para meses posteriores como anteriores.

Encontraremos un botón que permite abrir un fichero de texto en el que se han guardado con antelación los datos referentes a la convocatoria del curso que se envió.

En esta funcionalidad seleccionaremos los recursos materiales y humanos que necesitamos para la celebración del curso:

- **Material:** contiene una casilla de verificación para ver si se requiere o no material. Si se requiere material, se habilitará la opción de la reserva de material, y por el importe que se reserva. Podremos abrir también la factura del importe de ese material solicitado en concreto.
- **Formador:** contiene una casilla de verificación para ver qué tipo de formador se requiere, tanto externo o interno. Una vez activados se habilitarán los datos pertenecientes a cada uno de ellos. Dentro del externo podremos seleccionar mediante un control de lista desplegable el proveedor que nos da servicio, e incluso mediante el botón modificar proveedor, podremos cambiar los datos acerca de un proveedor. Podremos abrir también la factura del importe de ese formador solicitado en concreto.
- **Aula:** contiene una casilla de verificación, para ver si se impartirá el curso en un aula interna de la entidad o en un aula externa a la entidad, habilitándose en cada caso los campos correspondientes. Si el aula es reservada por parte del proveedor, se introducirá el importe numérico del coste, pudiendo ver después la factura del importe de esa aula solicitada en concreto.
- **Restaurante:** contiene una casilla de verificación para ver si se requiere restaurante para el curso, habilitándose en cada caso los campos correspondientes. Si el restaurante tiene que ser reservado, se introducirá el importe numérico del coste, pudiendo ver después la factura del importe de ese restaurante solicitada en concreto.
- **Transporte:** contiene una casilla de verificación, para ver si se requiere algún tipo de transporte para el curso, habilitándose en cada caso los campos correspondientes. Si el transporte tiene que ser reservado, se introducirá el importe numérico del coste, pudiendo ver después la factura del importe del transporte solicitado en concreto.

Otros datos:

- **Subvención y archivo:** contiene varias casillas de verificación para ver si el curso está subvencionado y los archivos que tenemos de él. En el caso de que este subvencionado, se habilitara la casilla de

verificación control de gestión, que mostrará una lista desplegable en formato calendario para poder seleccionar el día exacto, pudiendo avanzar meses anteriores y posteriores.

Podremos activar también los archivos disponibles que tenemos del curso, como son el listado de firmas, los cuestionarios y los diplomas.

- Cuestionarios: contiene una casilla de verificación para ver si se envía el cuestionario a grabar, mostrando una lista desplegable en formato calendario para poder seleccionar el día exacto, pudiendo avanzar meses anteriores y posteriores. Contiene un botón que podrá abrir un archivo, para poder ver la hoja de valoración del curso.

La otra casilla de verificación controla si se reciben los resultados del curso, mostrando una lista desplegable en formato calendario para poder seleccionar el día exacto, pudiendo avanzar meses anteriores y posteriores.

- Otras tareas: contiene una tabla anexada en vista de datos, mostrando otras tareas excepcionales, que no han podido ser recogidas en los apartados anteriores. En cada descripción tendremos una casilla de verificación controlando si esta tarea se ha realizado o no para llevar su control.
- Software: contiene una casilla de verificación para ver si se requiere software para el curso o no, habilitándose si se requiere la casilla de comprobado, controlando si se han hecho comprobaciones del software necesario.

Detalles fechas:

Dentro del apartado fechas de periodos, podremos ir incluyendo en aquellos cursos que se celebren de manera discontinua, las fechas de inicio y finalización de cada periodo, por ejemplo si un curso se imparte 2 semanas en un mes, y otras 2 semanas en el mes siguiente, en este caso para ver que la duración del curso no es de 2 meses, se separan los periodos que conforman el curso.

Baja de sesiones:

Para eliminar una sesión primero deberemos acceder a la funcionalidad buscar curso / sesión, para después proceder a darla de baja.

Búsqueda de curso / sesión:

Dentro de la funcionalidad buscar curso / sesión, se nos presentará un nuevo formulario en el que podremos realizar diferentes tipos de búsqueda por los siguientes criterios:

- Curso
- Centro de formación
- Fecha
- Subvención
- Plantilla

Una vez seleccionados los diferentes tipos de criterios que queramos usar, seleccionaremos la sesión del curso con la que queramos trabajar, y pulsando en el botón aceptar sesión, que nos llevará al formulario de gestión de las convocatorias.

Imprimir diplomas preimpresos:

Podremos imprimir diplomas de asistencia por cada alumno que ha confirmado su asistencia a un curso, certificando que ese alumno ha asistido al curso en unas fechas concretas.

Se abrirá un informe por cada alumno que haya asistido al curso, pudiendo imprimir cada diploma.



MARÍA PÉREZ MARTÍN

CERTIFICA QUE

GARCÍA RODRÍGUEZ, JESÚS

HA PARTICIPADO EN LA ACCION FORMATIVA

000159 - GESTIÓN DE PROYECTOS

Que tuvo lugar en: Madrid

Del 04/02/2009 al 05/02/2009 con un total de 16 horas presenciales.

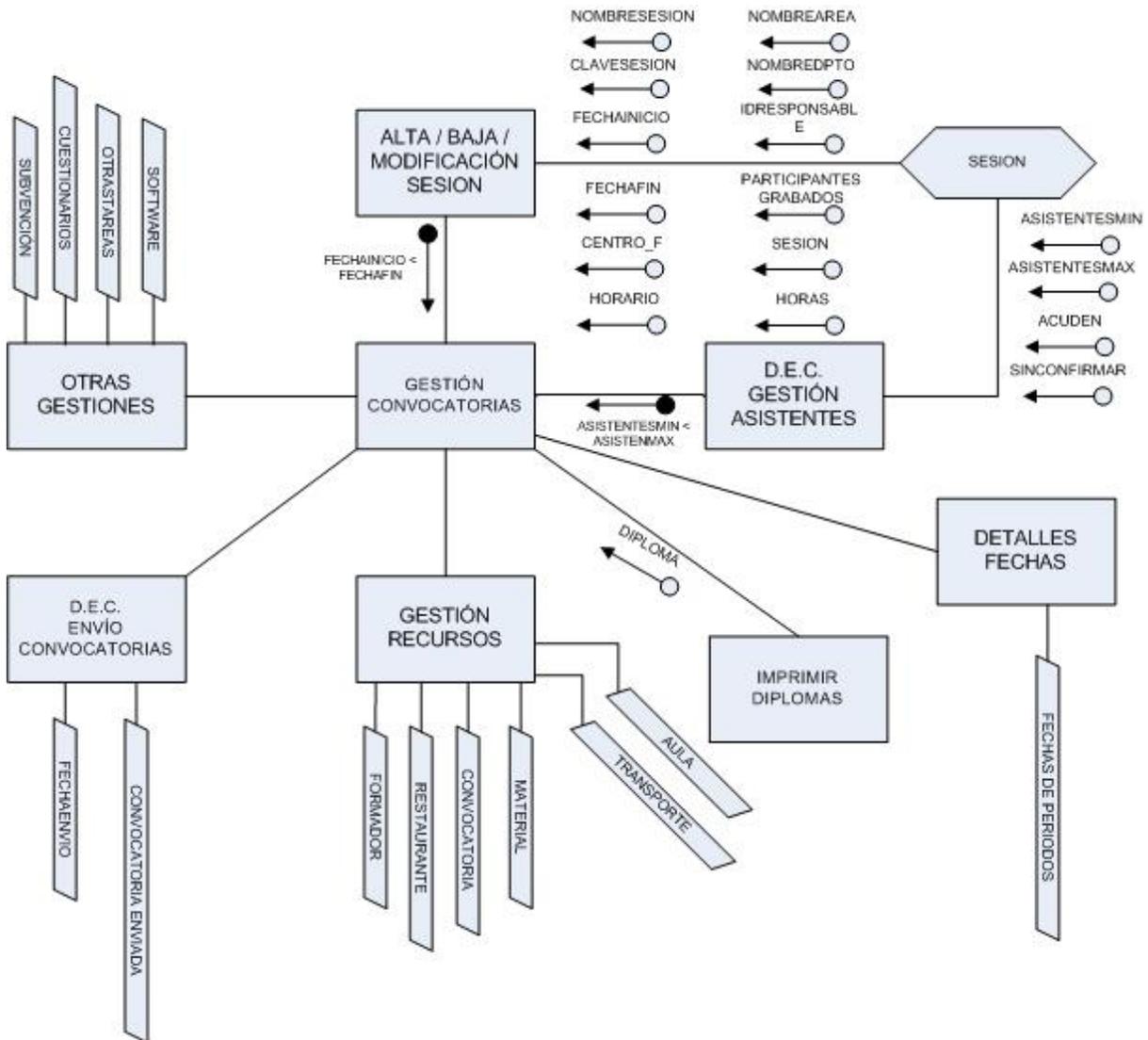
Y para que conste y surta los efectos oportunos donde proceda se expide el presente
Certificado de Asistencia

MARÍA PÉREZ MARTÍN
Directora de Formación

GARCÍA RODRÍGUEZ, JESÚS

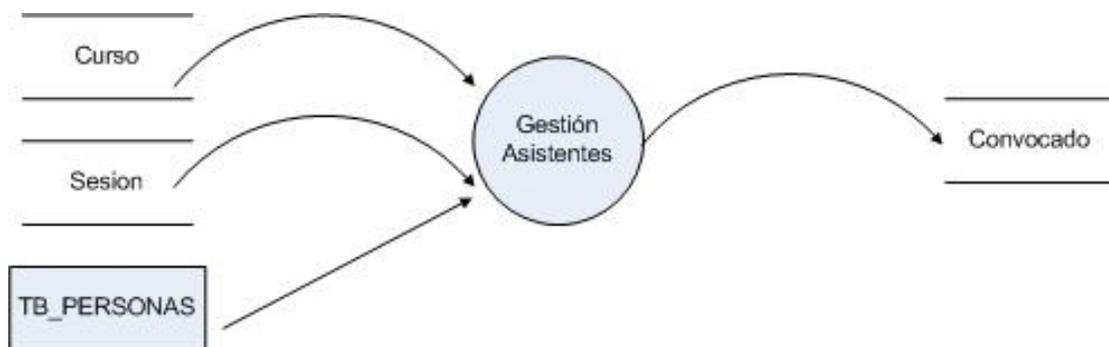
Ejemplo Diploma

D.E.C.:



D.E.C. Gestión de las sesiones

2º- Gestión de los asistentes:



D.F.D. Gestión de los asistentes

Dentro de la gestión de los candidatos podemos encontrar 3 apartados principales:

- Datos del curso
- Incluir asistente
- Candidatos y convocados

Datos del curso:

En este apartado encontraremos 2 controles de lista desplegable. El primero de ellos, curso, contendrá el listado de todos los cursos disponibles en la base de datos.

Una vez seleccionado el curso, se habilitará un desplegable con las sesiones disponibles para ese curso en concreto.

Incluir asistente:

Podremos buscar cualquier empleado para que acuda como asistente al curso seleccionado. Para ello dispondremos de un botón, el cual nos permitirá hacer búsquedas tanto por el nombre de usuario o sus apellidos, por matrícula, o por su código de usuario, mostrando en la parte de la derecha una lista desplegable con las posibles coincidencias que haya obtenido la consulta. Una vez que seleccionemos el usuario, podremos añadirlo a la lista de espera de posibles asistentes que asistan al curso mediante un botón, que hará que el empleado seleccionado entre en la lista de candidatos, antes de incluirlo se comprobará que el asistente no esté en la lista para no duplicarlo.

Candidatos y convocados:

Contendrá principalmente 2 sub-formularios: el de candidatos y el de convocados. El de candidatos tendrá que estar relleno desde el apartado de incluir asistente, estos asistentes podrán pasar mediante diferentes botones (<<), (<), (>), (>>)) de una lista a otra, pudiendo seleccionar un candidato o todos y



pasarlos a la lista de convocados, y viceversa, quitar empleados convocados y pasarlos a candidatos. Para ello se deberá realizar comprobaciones previas para que no contengan el mismo empleado en las 2 listas a la vez, ni en la misma lista 2 veces.

Al intentar pasar un empleado a la lista de convocados, si el empleado está inscrito en algún curso en las mismas fechas en las que se celebra el curso actual, el sistema le mostrará un mensaje de advertencia diciéndole que el empleado está inscrito en otro curso, y si realmente deseamos incluirle en la lista de convocados.

En todo momento estaremos informados mediante cuadros de texto de el número de candidatos y convocados que hay en cada lista, así como de el numero de cancelados y confirmados que hay en las listas respectivamente.

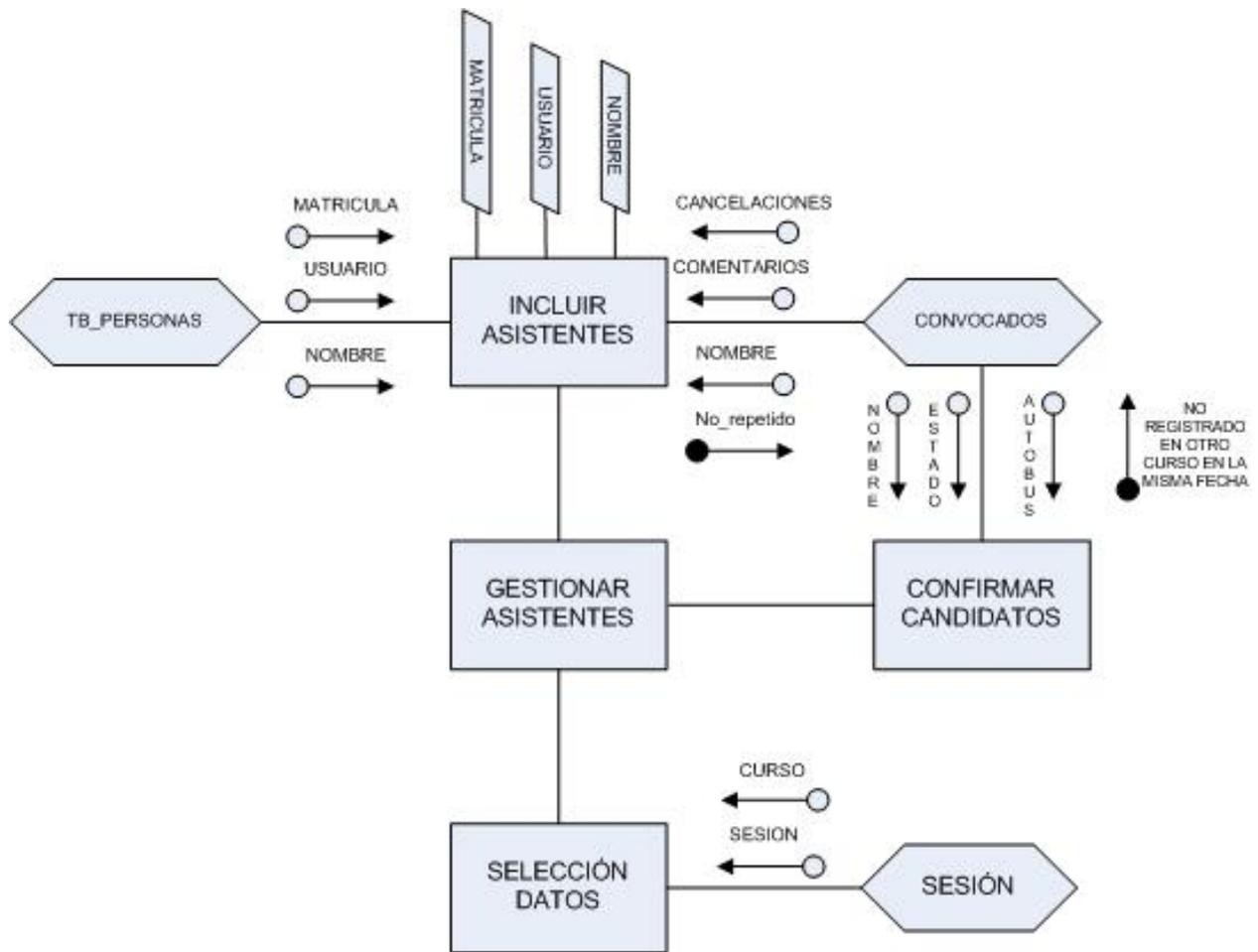
En la lista de candidatos podremos suprimir directamente a un empleado mediante el botón suprimir candidato, que sacara al empleado de la lista definitivamente.

Y en la lista de convocados podremos confirmar la asistencia de un empleado al curso mediante el botón confirmar asistente, y podremos cancelar la asistencia de un empleado mediante el botón cancelar asistente, el cual será sacado de la lista, para ir a la de candidatos, dejando previamente un comentario de porque se le ha tenido que sacar de la lista de convocados.

Este formulario difiere dependiendo de donde es llamado, si lo hacemos desde el menú principal, en inscripción de alumnos, nos saldrán los campos de nombre de curso y de sesión vacíos, en los que tendremos que seleccionar en la lista los cursos y sesiones disponibles.

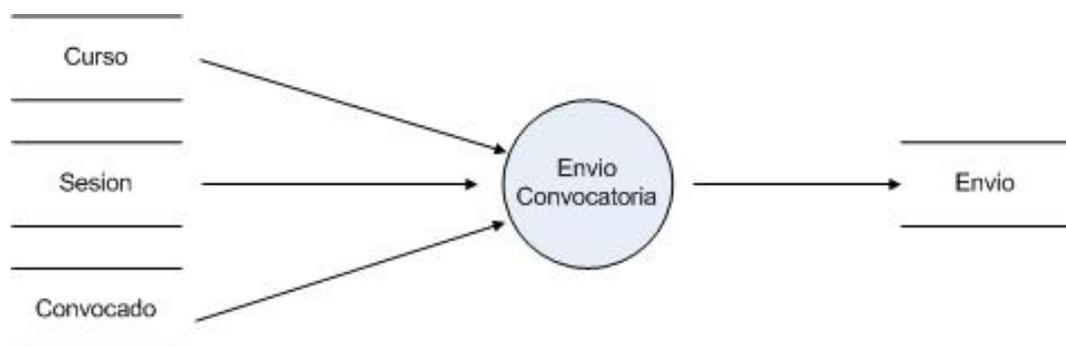
Si hacemos la llamada a la funcionalidad desde el menú principal, en alta de sesiones, podremos ir a gestionar asistentes, desde el formulario gestión de convocatorias. En este caso los campos de curso y sesión vendrán vinculados desde la gestión de las convocatorias, y solo se podrá trabajar en la sesión que estemos usando actualmente.

D.E.C.:



D.E.C. Gestión de los asistentes

3º- Gestión de las convocatorias:



D.F.D. Gestión de las convocatorias

En el envío de las convocatorias podemos distinguir como apartados principales:

- Datos del curso

- Candidatos
- Mensaje
- Anexos
- Buzón de envío

Datos del curso:

En este apartado encontraremos 2 controles de lista desplegable. El primero de ellos, curso, contendrá el listado de todos los cursos disponibles en la base de datos.

Una vez seleccionado el curso, se habilitará un desplegable con las sesiones disponibles para ese curso en concreto.

Candidatos:

Esta parte contendrá la lista de empleados convocados definitivos que asistirá en un futuro al curso. Dentro de este sub-formulario estarán habilitadas varias casillas de verificación, que nos permitirán elegir varios parámetros antes de enviar la convocatoria del curso en el que nos encontremos, como son:

- Copia a jefe: si activamos esta casilla de verificación, el envío de la convocatoria se realizará también al jefe del empleado.
- Enviar: con esta casilla de verificación podremos determinar si enviamos finalmente al empleado convocado el correo electrónico con los datos de la convocatoria.

Mensaje:

En este apartado podremos seleccionar de entre las plantillas (*.oft) que hemos diseñado previamente, cualquiera para componer el cuerpo que tendrá el mensaje.

También una vez seleccionada la plantilla, podremos visualizarla o modificarla en ese mismo instante.

Para que el mensaje sea correcto deberemos de rellenar los campos de:

- Título del curso
- Horario del curso
- Confirmar fecha antes de.

Anexos:

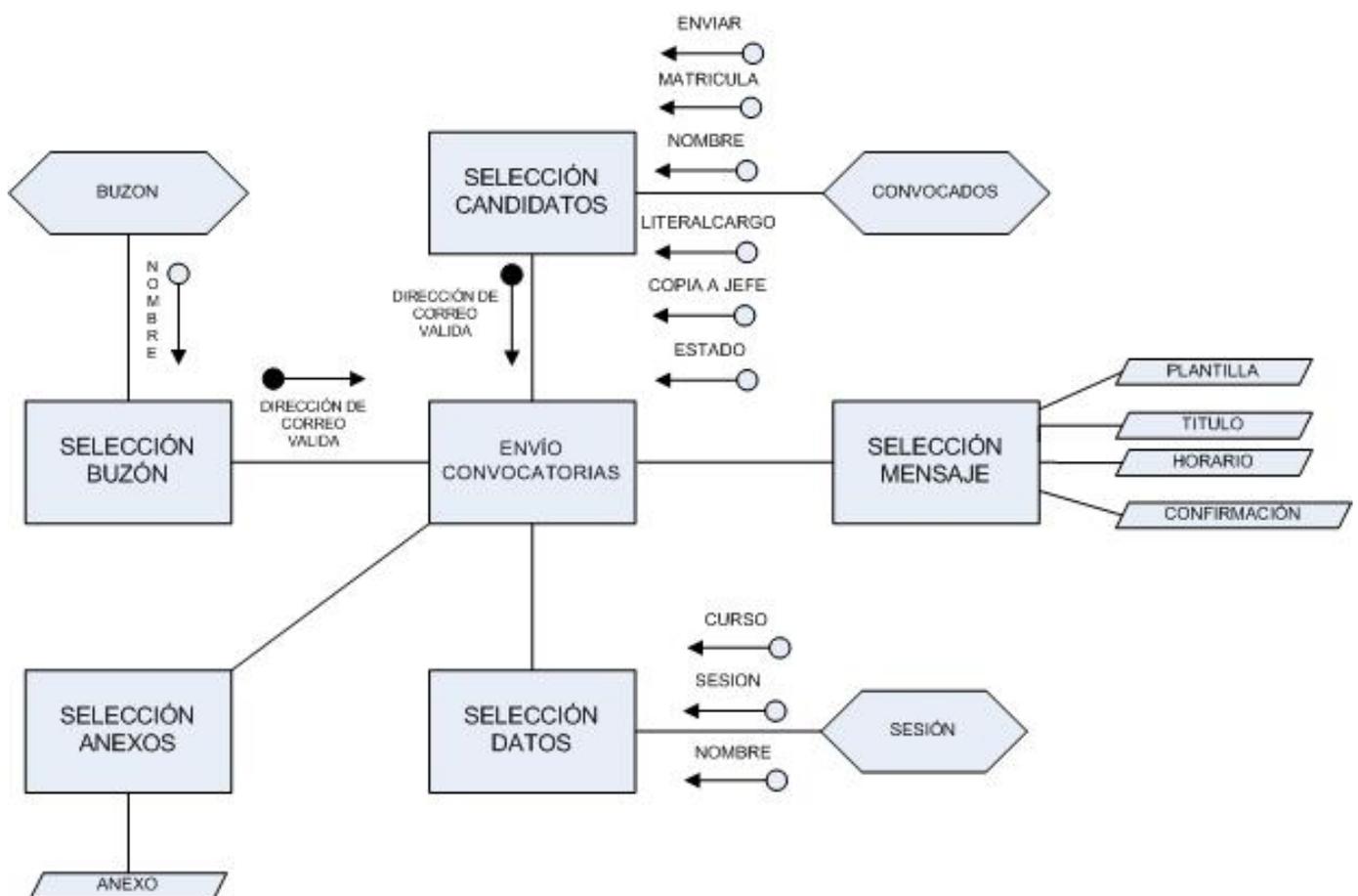
Mediante esta característica podremos controlar el envío de ficheros anexados a nuestro mensaje de confirmación de convocatoria, pudiendo incluir cualquier archivo que queramos como anexo, pulsando en el botón incluir anexo en la convocatoria.

Igualmente podremos eliminar de la lista de anexos, cualquier fichero que no consideremos que se debe enviar pulsando en el botón eliminar anexo de la convocatoria.

Buzón de envío:

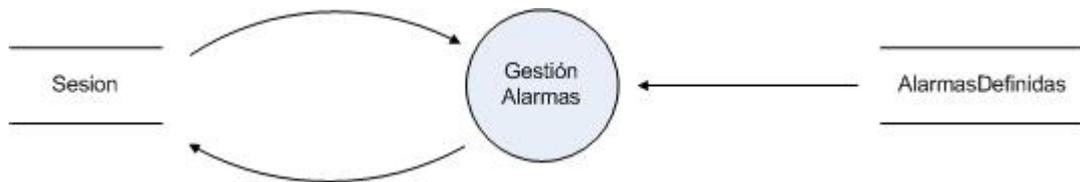
Por último podremos seleccionar desde este apartado el buzón desde donde queremos enviar el mensaje, de los que aparecen en la lista desplegable disponible, en definitiva rellenar el campo De, del mensaje que queremos enviar.

D.E.C.:



D.E.C. Gestión de las convocatorias

4º-Gestión de las alarmas:



D.F.D. Gestión de las alarmas

Dentro de esta funcionalidad podemos seleccionar los datos de los cursos y sus alarmas, para visualizar posteriormente la criticidad de las mismas:

- Escuela
- Orden (por alarmas o por fechas)
- Mostrar subvencionados
- Alertas
- Expandir todas

Escuela:

En este desplegable seleccionaremos las distintas escuelas por las que se ha querido filtrar: “Administración Central” o “Escuela de Tecnología” y también podremos seleccionar “Todas” a la vez.

Orden:

Aquí podemos elegir la ordenación de las diferentes consultas aplicadas, o por el tipo de alarmas, o por fecha.

Mostrar subvencionados:

Con esta casilla de verificación podemos distinguir aquellos cursos, que han sido subvencionados o los que por el contrario no.

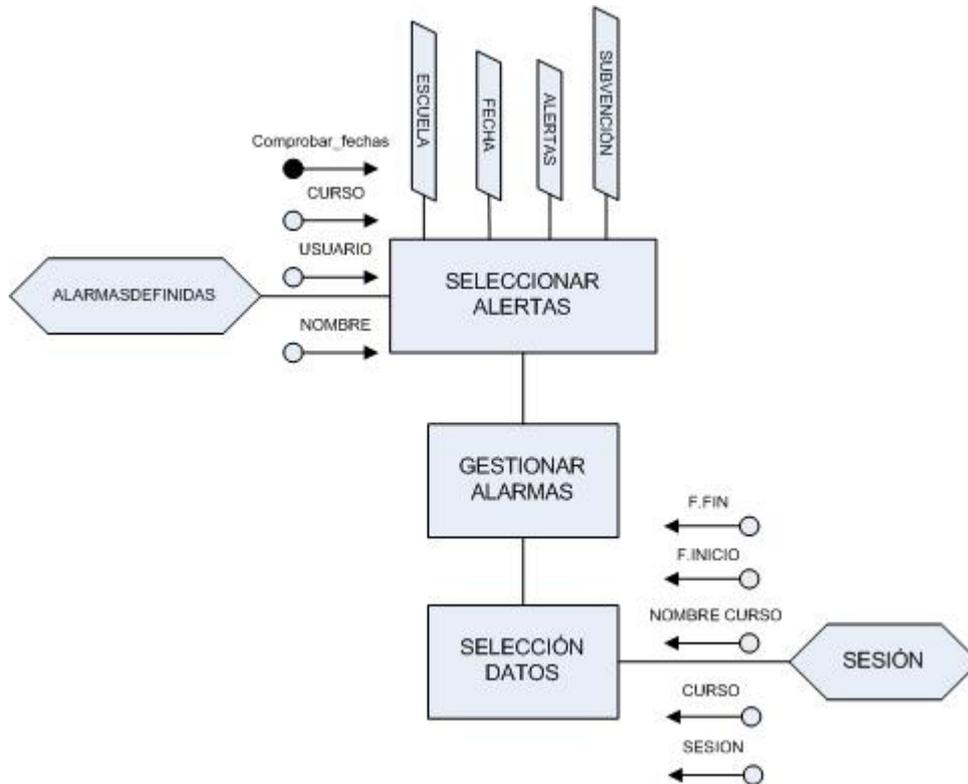
Alertas:

La lista desplegable que aquí se muestra, contendrá el listado de todas las posibles alarmas, que se hayan programado para un curso, pudiendo filtrar por cualquier alerta disponible.

Expandir todas:

Con este botón, podremos desplegar según los criterios de selección propuestos por los anteriores desplegables, todas las alarmas necesarias para los cursos que se muestran.

D.E.C:



D.E.C. Gestión de las alarmas

- Modelo Entidad / Relación extendido
- Diagrama de Clases
- Casos de Uso

Prácticas

- Sesiones de Trabajo
- Catalogación

Participantes

- Jefe de Proyecto
- Analistas
- Directores de los Usuarios

5.2.1.2. Tarea ASI 1.4: Identificación de los usuarios Participantes y finales.

En esta tarea podemos identificar como usuarios participantes para la toma de requisitos a Eduardo Muñoz Muñoz, y a Juan Carlos Urbina con los cuales hemos obtenido todas las especificaciones para la resolución del problema. Podemos contar como usuarios finales, a todas las personas que componen el departamento de RR.HH, más concretamente los encargados de gestionar los cursos de formación.

Productos

De entrada

- Catálogo de Usuarios (EVS 1.3 y EVS 2.2)
 - Eduardo Muñoz Muñoz.
 - Juan Carlos Urbina.
- Catálogo de Requisitos (ASI 1.2)

El entorno tecnológico usado para esta solución es un entorno básico, utilizando únicamente Microsoft Access como gestor de la información.

En Análisis Estructurado:

- Contexto del Sistema (ASI 1.1)

En Análisis Orientado a Objetos:

- Modelo de Negocio (ASI 1.1)
- Modelo de Dominio (ASI 1.1)

De salida

- Catálogo de Usuarios
- Plan de Trabajo

Prácticas

- Catalogación
- Sesiones de Trabajo

Participantes

- Jefe de Proyecto
- Analistas
- Directores de los Usuarios

5.2.2. *Actividad ASI 2: Establecimiento de requisitos.*

5.2.2.1. *Tarea ASI 2.1: Obtención de requisitos.*

Determinando los siguientes niveles tenemos:

- Funcionales: dentro de este apartado tenemos la gestión de las sesiones, gestión de los asistentes, envío de convocatorias, elaboración de informes de planificación y gestión de alarmas asociadas a los cursos.
- Rendimiento: la solución tiene que poder ejecutarse de manera óptima en cualquier sistema basado en tecnología Microsoft Windows, con Microsoft Access 2003.
- Seguridad: básica, protección de los datos contenidos en las tablas, para que sean accesibles únicamente por el personal de recursos humanos.
- Implantación: la solución debe ejecutarse junto con otras distintas aplicaciones, pero solo comparte información de la tabla con los datos de las personas de la compañía.
- Disponibilidad del sistema: la solución ha de estar disponible en cualquier momento para los usuarios del departamento de gestión de la formación.

Productos

De entrada

- Catálogo de Requisitos (ASI 1.4)
- Descripción General del Entorno Tecnológico del Sistema (ASI 1.4)
- Catálogo de Usuarios (ASI 1.4)
- Plan de Trabajo (ASI 1.4)

En Análisis Estructurado:

- Contexto del Sistema (ASI 1.1)

En Análisis Orientado a Objetos:

- Modelo de Negocio (ASI 1.1)
- Modelo de Dominio (ASI 1.1)

De salida

- Catálogo de Requisitos
- Modelo de Casos de Uso

Técnicas

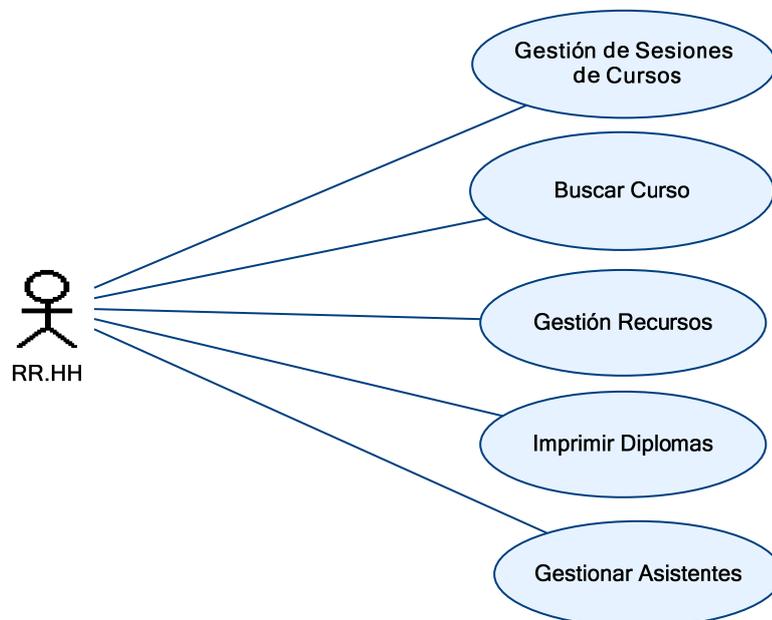
- Casos de Uso

Prácticas

- Sesiones de Trabajo
- Catalogación

5.2.2.2. Tarea ASI 2.2: Especificación de Casos de Uso.

1º- Gestión de los asistentes:



Caso de uso Gestión de las asistentes



Gestión de sesiones de los cursos:

Nombre:	Gestión de Sesiones de cursos.
Autor:	Enrique Jiménez.
Fecha:	15/02/2009.
Descripción:	Permite dar de alta, modificar y eliminar las sesiones de los cursos.
Actores:	RR.HH.
Precondiciones:	El usuario debe haberse logeado en el sistema.
Flujo Normal:	<ol style="list-style-type: none">1. El actor pulsa sobre el botón alta de sesiones.2. El sistema muestra un formulario con las opciones de alta, baja o modificación de sesión.3. El actor introduce los datos referentes a la sesión en caso de alta o modificación, o selecciona la sesión para proceder a su eliminación.4. El sistema comprueba la validez de los datos y los introduce, modifica o los borra.
Flujo Alternativo:	<ol style="list-style-type: none">5. El sistema comprueba la validez de los datos, si los datos no son correctos, se avisa al actor de ello permitiéndole que los corrija.
Postcondiciones:	La sesión queda introducida, modificada, o eliminada en el sistema.

Buscar curso:

Nombre:	Buscar cursos,
Autor:	Enrique Jiménez.
Fecha:	15/02/2009.
Descripción:	



Permite buscar las sesiones dadas de alta de los cursos.
Actores: RR.HH.
Precondiciones: El usuario debe haberse logeado en el sistema.
Flujo Normal: <ol style="list-style-type: none">1. El actor pulsa sobre el botón alta de sesiones.2. El sistema muestra un formulario con la opción de búsqueda de la sesión.3. El actor introducirá los datos en el botón de búsqueda de la sesión.4. El sistema comprueba la validez de los datos y los muestra.
Flujo Alternativo: <ol style="list-style-type: none">5. El sistema comprueba la validez de los datos, si los datos no son correctos, se avisa al actor de ello permitiéndole que los corrija.
Postcondiciones: Se mostrará la sesión en el sistema.

Gestión de recursos:

Nombre:	Gestión Recursos.
Autor:	Enrique Jiménez.
Fecha:	15/02/2009.
Descripción:	Permite especificar las necesidades de material, aulas, transporte, restaurante y formador de las sesiones de los cursos.
Actores: RR.HH.	
Precondiciones: El usuario debe haberse logeado en el sistema.	

**Flujo Normal:**

1. El actor pulsa sobre el botón alta de sesiones.
2. El sistema muestra un formulario con las opciones de material, aulas, transporte, restaurante y formador de las sesiones de los cursos de la sesión.
3. El actor introduce los datos referentes a los recursos necesarios para la celebración de los cursos.
4. El sistema comprueba la validez de los datos y los introduce, modifica o los borra.

Flujo Alternativo:

5. El sistema comprueba la validez de los datos, si los datos no son correctos, se avisa al actor de ello permitiéndole que los corrija.

Postcondiciones:

Los recursos de la sesión del curso actual quedan introducidos en el sistema.

Imprimir diplomas:

Nombre:	Imprimir Diplomas.
Autor:	Enrique Jiménez.
Fecha:	15/02/2009.
Descripción:	Permite imprimir diplomas preimpresos de las sesiones de los cursos.
Actores:	RR.HH.
Precondiciones:	El usuario debe haberse logeado en el sistema.
Flujo Normal:	<ol style="list-style-type: none">1. El actor pulsa sobre el botón alta de sesiones.2. El sistema muestra un formulario con el botón de imprimir diplomas preimpresos.3. El actor introduce los datos referentes a los diplomas de los cursos.4. El sistema comprueba la validez de los datos y los imprime.

**Flujo Alternativo:**

5. El sistema comprueba la validez de los datos, si los datos no son correctos, se avisa al actor de ello permitiéndole que los corrija.

Postcondiciones:

El sistema imprime los diplomas de los cursos.

Gestión de los asistentes:

Nombre:	Gestión Asistentes.
Autor:	Enrique Jiménez.
Fecha:	15/02/2009.
Descripción:	Permite incluir, modificar o eliminar los asistentes de las sesiones de los cursos que se celebrarán.
Actores:	RR.HH.
Precondiciones:	El usuario debe haberse logeado en el sistema.
Flujo Normal:	<ol style="list-style-type: none">1. El actor pulsa sobre el botón alta de sesiones.2. El actor pulsa sobre el botón Gestionar Asistentes.3. El sistema muestra un formulario con los datos referentes a los empleados que son candidatos a asistir al curso y los empleados que están convocados.4. El actor selecciona el curso y la sesión del curso.5. El actor seleccionará el empleado que quiere incluir.6. El actor introduce los datos referentes a los candidatos que desea incorporar tanto a la lista de candidatos como a la lista de convocados de los cursos disponibles.7. El sistema comprueba la validez de los datos y los introduce, modifica o borra de las correspondientes listas de candidatos y convocados.
Flujo Alternativo:	<ol style="list-style-type: none">8. El sistema comprueba la validez de los datos, si los datos no son correctos,

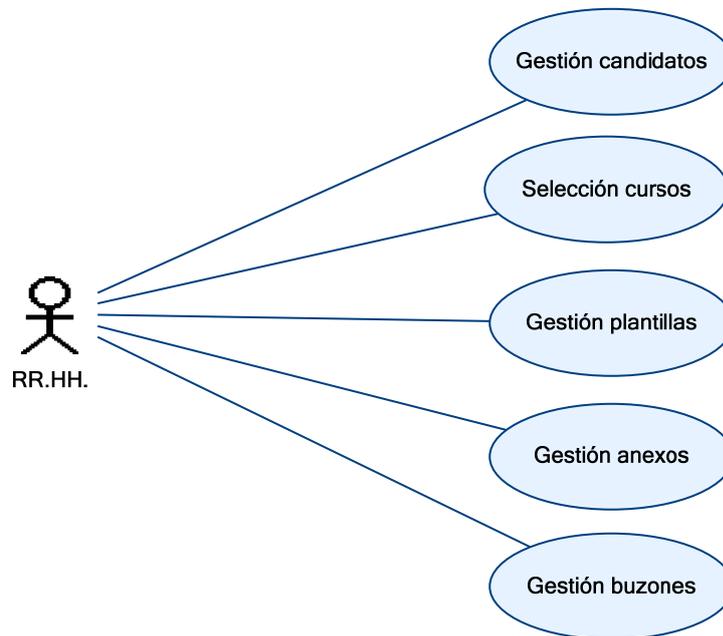


se avisa al actor de ello permitiéndole que los corrija.

Postcondiciones:

El sistema introduce, modifica o borra los empleados candidatos o convocados del curso seleccionado.

2º- Gestión de las convocatorias:



Caso de uso Gestión de las convocatorias

Gestión de candidatos:

Nombre:	Gestión candidatos.
Autor:	Enrique Jiménez.
Fecha:	15/02/2009.
Descripción:	Permite gestionar la lista de distribución de la sesión del curso que se celebrará.
Actores:	RR.HH.
Precondiciones:	El usuario debe haberse logeado en el sistema.
Flujo Normal:	<ol style="list-style-type: none"> 1. El actor pulsa sobre el botón alta de sesiones. 2. El actor pulsa sobre el botón Gestionar convocatoria. 3. El sistema muestra un formulario con los datos referentes a la lista de distribución. 4. El actor seleccionará candidatos a los que se les mandará la convocatoria. 5. El sistema comprueba la validez de los datos y los incluye dentro de la lista



de distribución.
Flujo Alternativo: 6. El sistema comprueba la validez de los datos, si los datos no son correctos, se avisa al actor de ello permitiéndole que los corrija.
Postcondiciones: El sistema mostrará la selección de los candidatos que se incluyen en la lista de distribución del curso seleccionado.

Selección de cursos:

Nombre:	Selección cursos.
Autor:	Enrique Jiménez.
Fecha:	15/02/2009.
Descripción:	Permite establecer la sesión del curso que se enviará.
Actores:	RR.HH.
Precondiciones:	El usuario debe haberse logeado en el sistema.
Flujo Normal:	<ol style="list-style-type: none">1. El actor pulsa sobre el botón alta de sesiones.2. El actor pulsa sobre el botón Gestionar convocatoria.3. El sistema muestra un formulario con los datos referentes a la lista de distribución.4. El actor seleccionará curso y la sesión.5. El sistema comprueba la validez de los datos y los incluye o borra de las correspondientes listas de distribución.
Flujo Alternativo:	<ol style="list-style-type: none">6. El sistema comprueba la validez de los datos, si los datos no son correctos, se avisa al actor de ello permitiéndole que los corrija

**Postcondiciones:**

El sistema mostrará la lista de distribución correspondiente al curso seleccionado.

Gestión de plantillas:

Nombre:	Gestión plantillas.
Autor:	Enrique Jiménez.
Fecha:	15/02/2009.
Descripción:	Permite seleccionar el modelo de plantilla el cual formará el cuerpo del mensaje del envío.
Actores:	RR.HH.
Precondiciones:	El usuario debe haberse logeado en el sistema.
Flujo Normal:	<ol style="list-style-type: none">1. El actor pulsa sobre el botón alta de sesiones.2. El actor pulsa sobre el botón Gestionar convocatoria.3. El sistema muestra un formulario con los datos referentes a la lista de distribución.4. El actor seleccionará la plantilla.5. El sistema comprueba la validez de los datos y conforma el mensaje de la lista de distribución.
Flujo Alternativo:	<ol style="list-style-type: none">6. El sistema comprueba la validez de los datos, si los datos no son correctos, se avisa al actor de ello permitiéndole que los corrija.
Postcondiciones:	El sistema generará el cuerpo del mensaje correspondiente al curso seleccionado.



Gestión de anexos:

Nombre:	Gestión anexos.
Autor:	Enrique Jiménez.
Fecha:	15/02/2009.
Descripción:	<p>Permite incluir los diferentes anexos opcionales que podrá tener el mensaje del envío.</p>
Actores:	<p>RR.HH.</p>
Precondiciones:	<p>El usuario debe haberse logeado en el sistema.</p>
Flujo Normal:	<ol style="list-style-type: none">1. El actor pulsa sobre el botón alta de sesiones.2. El actor pulsa sobre el botón Gestionar convocatoria.3. El sistema muestra un formulario con los datos referentes a la lista de distribución.4. El actor añadirá los anexos correspondientes.5. El sistema comprueba la validez de los datos y añade los anexos de la lista de distribución.
Flujo Alternativo:	<ol style="list-style-type: none">6. El sistema comprueba la validez de los datos, si los datos no son correctos, se avisa al actor de ello permitiéndole que los corrija.
Postcondiciones:	<p>El sistema incluirá los anexos en el mensaje correspondiente al curso seleccionado.</p>

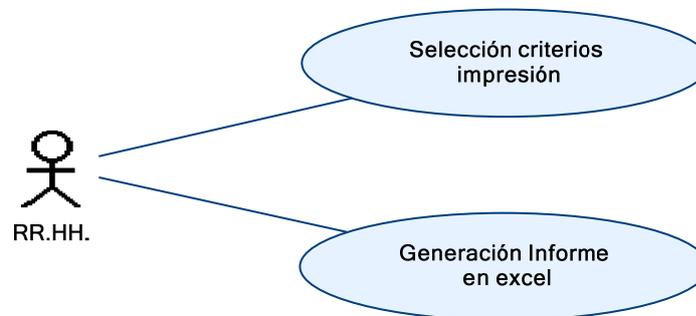
Gestión de buzones:

Nombre:	Gestión buzones.
Autor:	Enrique Jiménez.
Fecha:	15/02/2009.



Descripción: Permite seleccionar los buzones desde los que se enviará mensaje del envío.
Actores: RR.HH.
Precondiciones: El usuario debe haberse logeado en el sistema.
Flujo Normal: <ol style="list-style-type: none">1. El actor pulsa sobre el botón alta de sesiones.2. El actor pulsa sobre el botón Gestionar convocatoria.3. El sistema muestra un formulario con los datos referentes a la lista de distribución.4. El actor seleccionará los buzones disponibles.5. El sistema comprueba la validez de los datos e incluirá los buzones para el mensaje correspondiente.
Flujo Alternativo: <ol style="list-style-type: none">6. El sistema comprueba la validez de los datos, si los datos no son correctos, se avisa al actor de ello permitiéndole que los corrija.
Postcondiciones: El sistema incluirá los buzones seleccionados en el mensaje correspondiente al curso seleccionado.

3º - Gestión de los informes planificación de los cursos:



Caso de uso Gestión de los informes de planificación

Selección criterios de impresión:

Nombre:	Selección criterios impresión.
Autor:	Enrique Jiménez.
Fecha:	15/02/2009.
Descripción:	Permite seleccionar los criterios de impresión del informe que se generará.
Actores:	RR.HH.
Precondiciones:	El usuario debe haberse logeado en el sistema.
Flujo Normal:	<ol style="list-style-type: none"> 1. El actor pulsa sobre el botón Seguimiento. 2. El sistema muestra un formulario con los datos referentes al informe a generar. 3. El actor seleccionará el mes, el año y la escuela de tecnología dentro de las opciones. 4. El sistema comprueba la validez de los datos y mostrará la planificación de los cursos según los criterios.
Flujo Alternativo:	<ol style="list-style-type: none"> 5. El sistema comprueba la validez de los datos, si los datos no son correctos,



se avisa al actor de ello permitiéndole que los corrija.

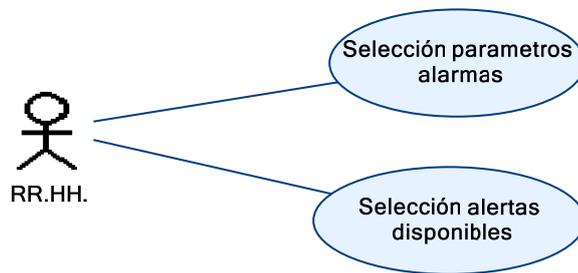
Postcondiciones:

El sistema mostrará la planificación de los cursos de las opciones seleccionadas.

Generación de informes en Excel:

Nombre:	Generación Informes en excel.
Autor:	Enrique Jiménez.
Fecha:	15/02/2009.
Descripción:	Permite generar un informe en formato Excel.
Actores:	RR.HH.
Precondiciones:	El usuario debe haberse logeado en el sistema.
Flujo Normal:	<ol style="list-style-type: none">1. El actor pulsa sobre el botón Seguimiento.2. El sistema muestra un formulario con los datos referentes a los informes de planificación.3. El actor pulsa sobre el botón Generar Excel.4. El sistema comprueba la validez de los datos y mostrará la planificación correspondiente.
Flujo Alternativo:	<ol style="list-style-type: none">5. El sistema comprueba la validez de los datos, si los datos no son correctos, se avisa al actor de ello permitiéndole que los corrija.
Postcondiciones:	El sistema mostrará la planificación de los cursos seleccionados.

4º-Gestión de las alarmas:



Caso de uso Gestión de las alarmas

Selección parámetros alarmas:

Nombre:	Selección parámetros alarmas.
Autor:	Enrique Jiménez.
Fecha:	15/02/2009.
Descripción:	Permite seleccionar las diferentes opciones para la generación de alarmas.
Actores:	RR.HH.
Precondiciones:	El usuario debe haberse logeado en el sistema.
Flujo Normal:	<ol style="list-style-type: none"> 1. El actor pulsa sobre el botón Alertas. 2. El sistema muestra un formulario con los datos referentes a las opciones de las alarmas disponibles. 3. El actor selecciona la escuela de tecnología, el orden de datos y si decide mostrar las subvenciones. 4. El sistema comprueba la validez de los datos y mostrará las alarmas presentadas.
Flujo Alternativo:	<ol style="list-style-type: none"> 5. El sistema comprueba la validez de los datos, si los datos no son correctos, se avisa al actor de ello permitiéndole que los corrija.



Postcondiciones:

El sistema mostrará las opciones disponibles para gestionar las alarmas.

Selección alertas disponibles:

Nombre:	Selección alertas disponibles.
Autor:	Enrique Jiménez.
Fecha:	15/02/2009.
Descripción:	Permite seleccionar las alertas disponibles para los cursos planificados.
Actores:	RR.HH.
Precondiciones:	El usuario debe haberse logeado en el sistema.
Flujo Normal:	<ol style="list-style-type: none">1. El actor pulsa sobre el botón Alertas.2. El sistema muestra un formulario con los datos referentes a las alarmas disponibles.3. El actor selecciona las alertas disponibles.4. El sistema comprueba la validez de los datos y mostrará las alarmas que se encuentran activadas.
Flujo Alternativo:	<ol style="list-style-type: none">5. El sistema comprueba la validez de los datos, si los datos no son correctos, se avisa al actor de ello permitiéndole que los corrija.
Postcondiciones:	El sistema mostrará las alarmas que se encuentran activadas.

5.2.2.3. Tarea ASI 2.4 (ASI-SEG 1.1): Estudio de la seguridad requerida en el proceso de análisis del sistema de información.

La autenticación del usuario será la proporcionada por Windows, sin tener ningún otro tipo de validación en nuestra solución. La disponibilidad será total una vez el usuario acceda a la aplicación sin poder ser usada por otro usuario hasta el término de su sesión en el sistema. Se podrán abrir varias instancias de la aplicación pero no se podrán grabar datos hasta que el primer usuario termine de usar la suya, para garantizar la consistencia e integridad de los datos.

Productos

De entrada

- Catálogo de Requisitos (ASI 1.1)
- Planificación detallada (GPI 2)

De salida

- Seguridad Requerida en el Proceso Análisis del Sistema de Información
- Seguridad para Ejecución de Actividades
- Seguridad para Clasificación y Catalogación de los Productos Intermedios

Prácticas

- Sesiones de Trabajo

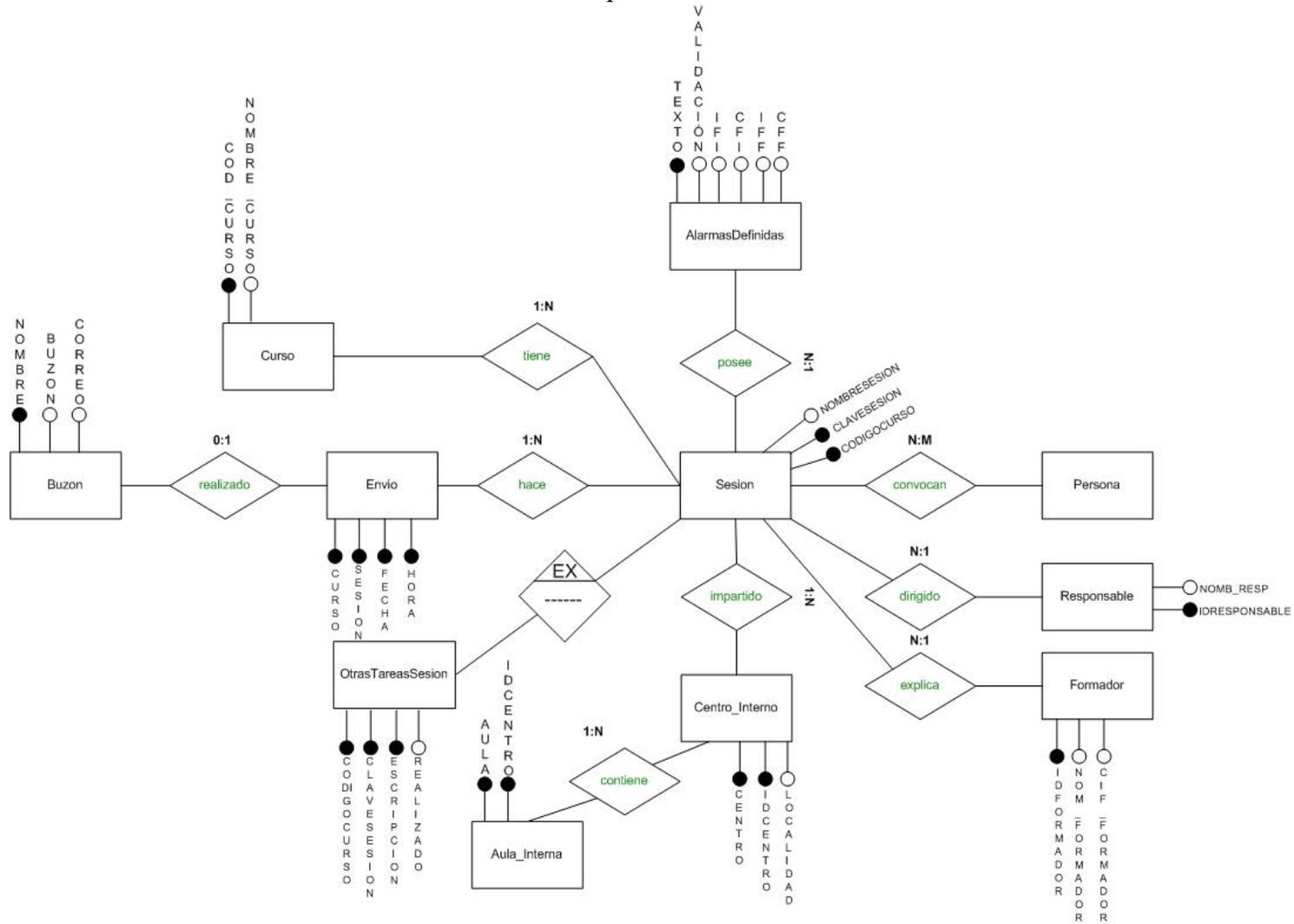
Participantes

- Equipo de Seguridad
- Responsable de Seguridad
- Jefe de Proyecto

5.2.3. Actividad ASI 6: Elaboración del modelo de datos.



5.2.3.1. Tarea ASI 6.1: Elaboración del modelo conceptual de datos.



Modelo conceptual de datos



Productos

De entrada

- Contexto del Sistema (ASI 1.1)
- Modelo Conceptual de Datos (ASI 1.1)

De salida

- Modelo Conceptual de Datos

Técnicas

- Modelo Entidad / Relación Extendido

Participantes

- Analistas

5.2.3.2. Tarea ASI 6.2: Elaboración del modelo lógico de datos.



Modelo lógico de datos

Productos

De entrada

- Modelo Conceptual de Datos (ASI 6.1)

De salida

- Modelo Lógico de Datos

Técnicas

- Modelo Entidad / Relación Extendido

Participantes

- Analistas

5.2.3.3. Tarea ASI 6.4: Especificación de necesidades de migración de datos y carga inicial.

Toda la carga de datos, en lo que se refiere a valores fijos, se ha introducido en la solución de forma manual, como por ejemplo cursos, aulas, etc.

Las tablas creadas a partir de estos valores, como son la de sesiones, convocatorias y demás, han sido rellenadas a través del uso de la aplicación, mediante la incorporación de sesiones completas de los cursos a nuestra solución.

Por último la tabla “TB_PERSONAS”, ya estaba creada, siendo una tabla que se vincula directamente con las demás, para poder utilizar los datos del personal de la empresa, que ya teníamos previamente.

Productos

De entrada

- Modelo Lógico de Datos Normalizado (ASI 6.3)
- Estructuras de Datos del Sistema Origen (externo)

De salida

- Plan de Migración y Carga Inicial de Datos



Prácticas

- Sesiones de Trabajo

Participantes

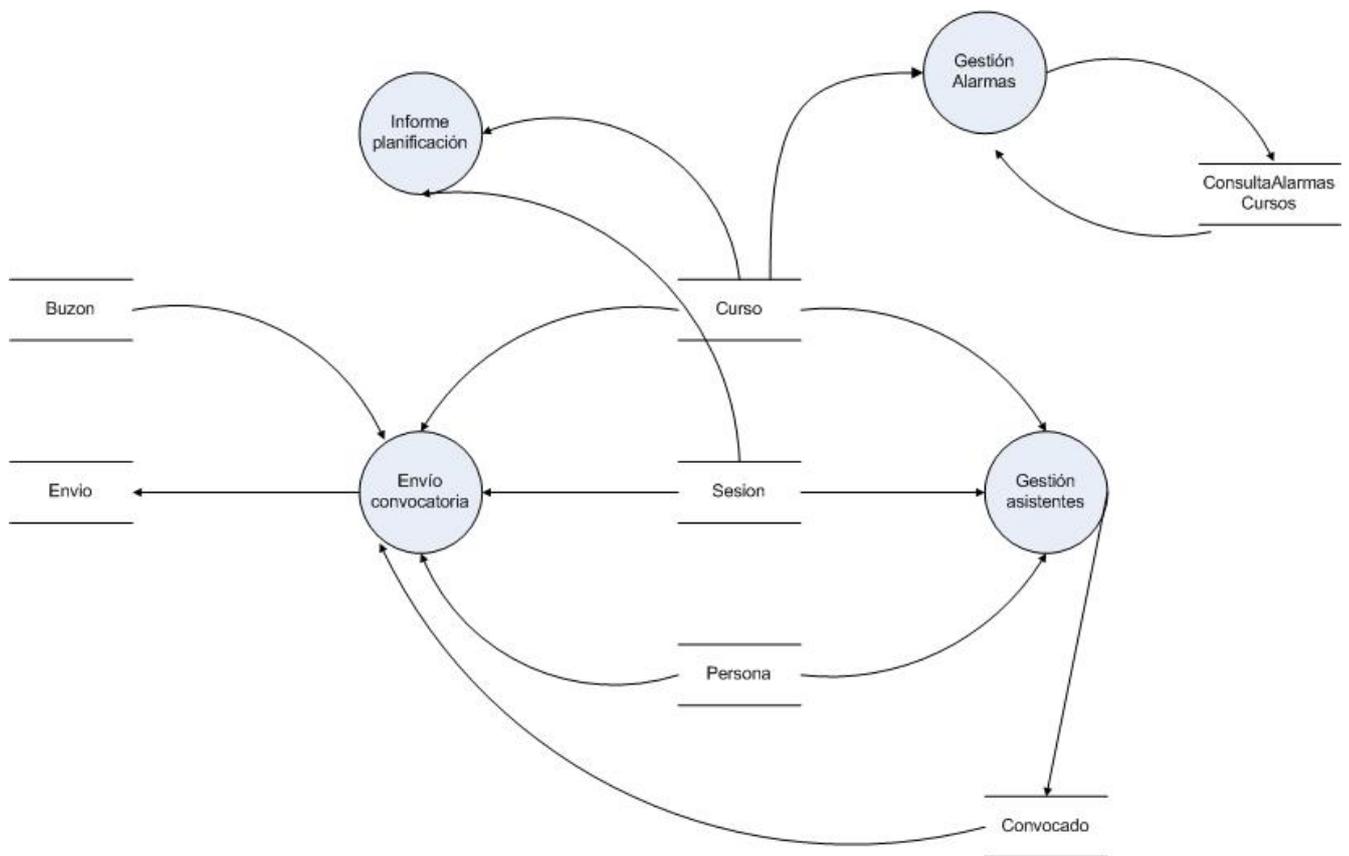
- Usuarios Expertos
- Analistas
- Equipo de Soporte Técnico

5.2.4. Actividad ASI 7: Elaboración del modelo de procesos.

Dentro de nuestra solución podemos encontrar como principales procesos o funcionalidades las siguientes:

- Gestión de los asistentes
- Envío de convocatorias.
- Creación de los informes de planificación.
- Gestión de las alarmas.

5.2.4.1. Tarea ASI 7.1: Obtención del modelo de procesos del sistema.



Modelo de procesos del sistema

Productos

De entrada

- Modelo de procesos (ASI 3.2)

De salida

- Modelo de Procesos
- Matriz de Procesos / Localización Geográfica (ampliada)

Técnicas

- Diagrama de Flujo de Datos
- Matricial

Participantes



- Analistas

5.2.4.2. Tarea ASI 7.2: Especificación de interfaces con otros sistemas.

Identificamos en esta tarea la tabla “TB_PERSONAS” como interfaz externa que actúa con nuestro sistema. Se trata de una tabla con información personal de los empleados de la compañía.

Este intercambio se producirá de manera mensual con las incorporaciones o las bajas que se puedan dar en el departamento de RR.HH.

En nuestro sistema es utilizada por los procesos de gestión de los asistentes y de envío de la convocatoria. También es utilizada de manera indirecta en la creación de los informes de planificación por medio de la tabla convocados que hace uso de esta tabla conteniendo solamente dentro de ella parte de los datos personales de las personas de la compañía.

Productos

De entrada

- Modelo de Procesos (ASI 7.1)

De salida

- Descripción de Interfaz con otros Sistemas

Participantes

- Analistas

5.2.5. Actividad ASI 8: Definición de interfaces de usuario.

5.2.5.1. Tarea ASI 8.3: Especificación de formatos individuales de la interfaz de pantalla.

1º- Menú principal



Interfaz de pantalla menú principal

En esta primera interfaz podemos encontrar el menú principal sobre el que se basa la solución. Dentro del menú, podemos ejecutar los siguientes formularios:

- Alta de sesiones
- Inscripción de Alumnos
- Alertas
- Seguimiento
- Calendario Intranet.

En esta interfaz el cambio de tamaño no está permitido, sin embargo el cambio de posición a lo largo de la pantalla sí, al no ejecutarse en pantalla completa podremos desplazarla hacia cualquier lugar de la pantalla.

Dispositivos de entrada:

Utilizaremos el ratón o teclado como único dispositivo de entrada para movernos por los diferentes botones que nos darán acceso a las distintas funcionalidades, o al botón salir para terminar la ejecución de la solución.

Datos y controles utilizados:

Para este menú los únicos datos que manejamos son la apertura de los distintos formularios y no hay ningún control adicional.

Elementos habilitados:

Como solo se trata de un menú de selección de opciones, estarán habilitados todos los botones al iniciar este formulario.

2º- Gestión de Convocatorias:

Interfaz de pantalla Gestión de convocatorias

Con esta interfaz podremos llevar el control de la gestión de las convocatorias, desde los datos identificativos a la sesión del curso, como los recursos necesarios y otras tareas para que se produzca la celebración del curso correctamente.

En esta interfaz el cambio de tamaño no está permitido, sin embargo el cambio de posición a lo largo de la pantalla sí, al no ejecutarse en pantalla completa podremos desplazarla hacia cualquier lugar de la pantalla.

Dispositivos de entrada:

Utilizaremos el ratón o teclado como único dispositivo de entrada para movernos por los diferentes apartados y características, que nos darán acceso a las distintas funcionalidades, o al botón salir para terminar la ejecución de la solución.

Datos y controles utilizados:

Para esta funcionalidad tendremos acceso principalmente a la tabla “Sesión”, y “Curso”.

Posteriormente cuando se haya completado los datos identificativos de la sesión se accederán a las tablas “Aula_Interna”, “Centro_Interno”, “Formador”, “Responsable” y “OtrasTareasSesion”.

Como desde este formulario se accede también a las funcionalidades de gestión de los asistentes y envío de las convocatorias también accederemos a los datos de “Convocados”, “TB_PERSONAS”, “Buzón”, “Envío”

Elementos habilitados:

En principio estarán habilitados los campos pertenecientes al subapartado “Datos identificativos de la sesión”. Una vez que queden guardados estos datos y la sesión haya sido introducida en la base de datos, se habilitarán los “Datos Generales de la sesión”, “Otros Datos” y “Detalles Fechas”, dependiendo de si tienen que ser rellenados sus campos correspondientes, no siendo siempre necesario que estén rellenos todos ellos.

Una vez este dada de alta la sesión es cuando se habilitara el botón “Eliminar Sesión”, que un principio permanecerá deshabilitado.

El botón imprimir diplomas preimpresos se habilitara en el caso de que generemos los diplomas para un curso ya finalizado.

Por ejemplo cuando hayamos hecho el envío de una convocatoria, se accederá al subapartado “Convocatoria”, tildaremos “Convocatoria Enviada” y se habilitará el campo de “fecha de envío” correspondiente a este apartado.

Al igual se procederá con los diferentes subapartados de los “Datos Generales de la sesión”:

- **Material**

En este apartado distinguiremos de si se requiere o no material de apoyo para la celebración del curso, en cuanto que activemos la casilla de verificación de “se requiere”, automáticamente se habilitarán las siguientes opciones:

- Material reservado.
- Importe del material.
- Factura del material reservado.

▪ Formador

Con esta opción podremos introducir el tipo de formador que queremos para nuestro curso.

Si seleccionamos formador interno se activará el nombre de nuestro formador interno, mientras que si seleccionamos formador externo, se habilitarán las siguientes opciones:

- Proveedor.
- CIF.
- Importe del formador.
- Factura del formador.
- Pagado.
- Botón de búsqueda del proveedor.

Una vez quede seleccionado cual va a ser finalmente nuestro formador, podremos activar la casilla de verificación de “formador cerrado”.

▪ Aula

En esta opción informaremos de cuál es el aula donde se impartirá el curso.

Si seleccionamos aula interna se activarán las listas desplegables donde se encuentra el centro y el aula donde se impartirá, mientras que si seleccionamos aula en proveedor, se habilitarán las siguientes opciones:

- Dirección.
- Coste.
- Importe del aula.
- Factura del aula.

Una vez quede seleccionado cual va a ser finalmente nuestra aula, podremos activar la casilla de verificación de “aula reservada”.

▪ Restaurante

En este apartado distinguiremos de si se requiere o no restaurante para la celebración del curso, en cuanto que activemos la casilla de verificación de “se requiere”, automáticamente se habilitarán las siguientes opciones:

- Lugar.
- Importe del restaurante.
- Factura del restaurante.



- Restaurante reservado.

- Transporte

Con esta última opción, distinguiremos de si se requiere o no, algún medio de transporte para la celebración del curso, en cuanto que activemos la casilla de verificación de “se requiere”, automáticamente se habilitarán las siguientes opciones:

- Transporte reservado.
- Importe del transporte.
- Factura del transporte reservado.

Alta de Sesiones - Alertas Formación

Departamento de Formación **Gestión de Convocatorias**

DATOS IDENTIFICATIVOS SESIÓN

Nombre: ... Clave: Fecha Ini: 29/11/2009 Fecha Fin: 29/11/2009

Centro F: Det. horario:

Área: Dpto: Responsable: Participantes Grabados

Sesión: Horas: Asistentes: Min: Max: Acuden: 0 Sin Conf.: 0

Datos Generales Otros Datos Detalles Fechas

SUBVENCIÓN Y ARCHIVO

NO Subvencionado FORCEM Programa

Listado de Firmas Cuestionarios Control de Gestión Diplomas Fecha Envío CG: 29/11/2009

CUESTIONARIOS

Enviado a Grabar Fecha Envío: 29/11/2009

Resultados Recibidos Fecha de Recibo: 29/11/2009 Valoración:

OTRAS TAREAS

Descripcion	Realizado
	<input type="checkbox"/>

SOFTWARE

No se Requiere Se Requiere Comprobado



Interfaz de pantalla Gestión de convocatorias (Otros Datos)

Y también en el subapartado “Otros Datos” con:

- Subvención y Archivo

Dentro de este apartado se habilitará la casilla de verificación “control de gestión”, en cuanto pasemos el curso de “NO Subvencionado” a “FORCEM”, o “Programa” mediante sus casillas de verificación correspondientes.

Contaremos con 3 casillas de verificación habilitadas desde un principio aparte:

- Listado de firmas.
- Cuestionarios.
- Diplomas.

- Cuestionarios

Una vez que el cuestionario haya sido enviado a grabar junto con la fecha de su envío, se habilitarán los siguientes campos:

- Resultados recibidos.
- Fecha de recibo.
- Valoración.



- Hoja de valoración.

- Otras Tareas

En este apartado, permanecerán habilitados siempre los campos descripción y realizado.

- Software

Cuando la opción “no se requiere” esta activada, no se pueden ver la casilla de verificación de “comprobado”, mientras que cuando si se requiere software, si se habilita esta opción.

Alta de Sesiones - Alertas Formación

Departamento de Formación **Gestión de Convocatorias**

DATOS IDENTIFICATIVOS SESIÓN

Nombre: Clave: Fecha Ini: 29/11/2009 Fecha Fin: 29/11/2009

Centro F: Det. horario:

Área: Dpto: Responsable: Participantes Grabados

Sesión: Horas: Asistentes: Min: Max: Acuden: 0 Sin Conf.: 0

Datos Generales | Otros Datos | Detalles Fechas

FECHAS DE PERIODOS

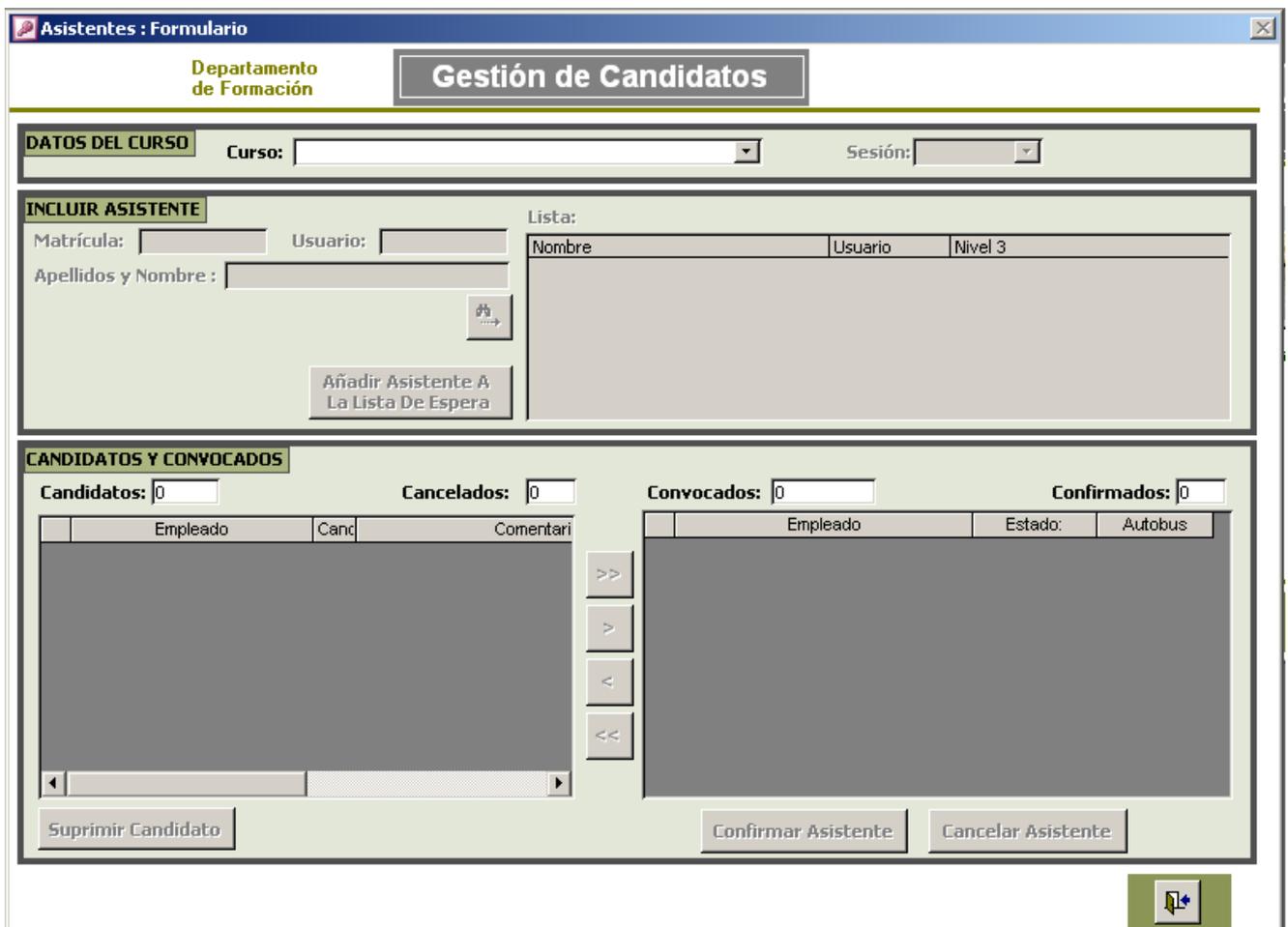
	Finicial:	FFinal:	Espaciado
▶			<input type="checkbox"/>

Interfaz de pantalla Gestión de convocatorias (Detalles Fechas)

Y en el apartado Detalles Fechas:

- Fechas de periodos: seleccionaremos la fecha inicial y la fecha final del curso, en el caso de que el curso se celebre en un periodo de tiempo continuo, e iremos añadiendo todas las filas necesarias en el caso de que el curso se imparta en periodos de tiempo discontinuo.

3º- Gestión de candidatos:



Interfaz de pantalla Gestión de candidatos

Con esta interfaz podremos llevar el control de la gestión de los candidatos, desde los datos del curso, como la gestión de los asistentes, controlando las listas de candidatos y convocados del curso seleccionado.

En esta interfaz el cambio de tamaño no está permitido, sin embargo el cambio de posición a lo largo de la pantalla sí, al no ejecutarse en pantalla completa podremos desplazarla hacia cualquier lugar de la pantalla.

Dispositivos de entrada:

Utilizaremos el ratón o teclado como único dispositivo de entrada para movernos por los diferentes apartados y características, que nos darán acceso a las distintas funcionalidades, o al botón salir para terminar la ejecución de la solución.

Datos y controles utilizados:

En esta interfaz accederemos únicamente a las tablas “Curso”, “Sesión”, “Convocados” y “TB_PERSONAS”.

Elementos habilitados:

A esta interfaz podemos llegar de 2 maneras:

- A través del formulario “Menú principal”: en este caso los datos identificativos a la sesión del curso, estarán vacíos. Pudiendo seleccionar la sesión de un curso de entre la lista de cursos disponibles.
- A través del formulario “Gestión de las convocatorias” donde los datos identificativos del curso, si estarán rellenos, debido a que se accede desde un curso seleccionado ya sea por la búsqueda, o la creación de uno nuevo.

En este caso no podremos seleccionar en la lista desplegable cualquier otro curso o sesión, ya que está vinculado con la sesión del curso correspondiente seleccionado en “Gestión de convocatorias”.

Todo lo demás correspondiente a esta interfaz será igual, aunque hagamos el acceso desde un u otro formulario.

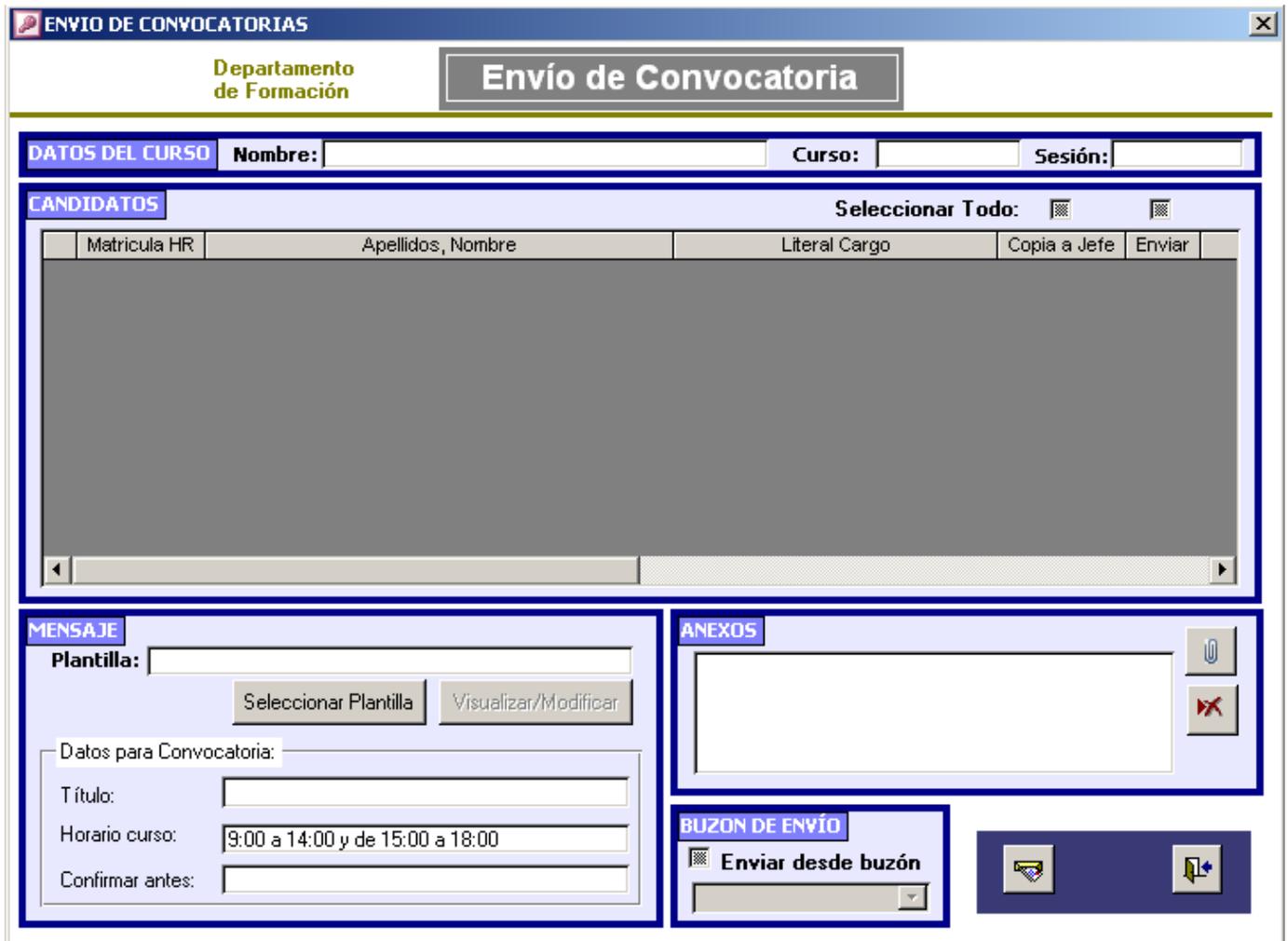
Aparte de los datos identificativos de la sesión de un curso, encontraremos los siguientes subapartados:

- Incluir asistente: esta funcionalidad se habilitará en el momento que quede una sesión seleccionada, y podremos utilizar todos los cuadros de texto para hacer búsquedas de empleados y poder agregarlos a la lista de candidatos.
- Candidatos y convocados: si la sesión del curso tenía candidatos o convocados quedarán habilitadas las listas, sus botones y sus cuadros de texto inmediatamente, una vez seleccionemos la sesión de los cursos.

Sino una vez que tengamos un asistente seleccionado en “Incluir asistente” y le demos al botón “Añadir asistente a la lista de espera”, se habilitara la lista de candidatos.

Una vez esa lista este llena de candidatos mediante los botones “(>)”, “(>>)”, se activara la lista de convocados, al igual que sus botones y sus cuadros de texto correspondientes.

4º- Envío de convocatoria



Interfaz de pantalla Envío de convocatoria

Con esta interfaz podremos llevar el control de los envíos de las convocatorias, desde los datos del curso, como la gestión de los candidatos, así como la composición del mensaje: configuración de las plantillas predefinidas, los anexos y los buzones de envío.

En esta interfaz el cambio de tamaño no está permitido, sin embargo el cambio de posición a lo largo de la pantalla sí, al no ejecutarse en pantalla completa podremos desplazarla hacia cualquier lugar de la pantalla.

Dispositivos de entrada:

Utilizaremos el ratón o teclado como único dispositivo de entrada para movernos por los diferentes apartados y características, que nos darán acceso a las distintas funcionalidades, o al botón salir para terminar la ejecución de la solución.

Datos y controles utilizados:

En esta interfaz accederemos únicamente a las tablas “Curso”, “Sesión”, “Convocados” y “Buzón”.

Elementos habilitados:

A esta interfaz llegamos a través del formulario “Gestión de convocatorias”, donde los datos identificativos del curso, si estarán rellenos debido a que se accede desde un curso seleccionado ya sea por la búsqueda o la creación de uno nuevo.

Aparte de los datos identificativos de la sesión de un curso, encontraremos los siguientes subapartados:

- Candidatos
- Mensaje
- Anexos
- Buzón de envío

Todos ellos habilitados una vez hayamos entrado en el formulario, excepto la funcionalidad buzón de envío, que permanecerá deshabilitada menos cuando tengamos que mandar el mensaje desde algún buzón de correo específico.

Productos

De entrada

- Especificación de Interfaz de Usuario (ASI 8.2)

En Análisis Orientado a Objetos:

- Especificación de Casos de Uso (ASI 2.4)
- Modelo de Casos de Uso (ASI 2.4)

De salida

- Especificación de Interfaz de Usuario:
 - Formatos Individuales de Interfaz de Pantalla
 - Catálogo de Controles y Elementos de Diseño de Interfaz de Pantalla

Técnicas

- Casos de Uso

Prácticas

- Prototipado
- Catalogación
- Sesiones de Trabajo

Participantes

- Usuarios Expertos
- Analistas

5.3. Documento de Diseño del sistema.**5.3.1. Actividad DSI 1: Definición de la arquitectura del sistema.****5.3.1.1. Tarea DSI 1.2: Identificación de requisitos de diseño y construcción.**

En nuestra solución, podemos encontrar de manera más general una plataforma basada en Microsoft Access, y como entorno de programación se ha diseñado entorno a objetos, con alguna información contenida en unas tablas ya predefinidas de las cuales hemos partido.

ProductosDe entrada

- Catálogo de Requisitos (ASI 2.4)
- Diseño de la Arquitectura del Sistema (DSI 1.1)

De salida

- Catálogo de Requisitos

Prácticas

- Sesiones de Trabajo
- Catalogación

Participantes

- Equipo de Arquitectura
- Equipo de Soporte Técnico

5.3.1.2. *Requisitos funcionales.*

- Gestión de las sesiones de los cursos.
- Gestión de los asistentes a los cursos.
- Envío de las convocatorias de los cursos.
- Planificación de los informes de seguimiento de los cursos.
- Gestión de las alarmas asociadas a los cursos.

5.3.1.2.1. *Requisitos de rendimiento.*

La solución deberá poder ejecutar todas sus funcionalidades con un tiempo de respuesta aceptable, no siendo este superior nunca a los 4 segundos para las consultas más complejas.

Los formularios diseñados se ejecutarán de manera instantánea, y al estar desarrollado en entorno Access podrán visualizarse sin necesidad alguna de instalar software complementario.

5.3.1.2.2. *Requisitos de interfaz.*

Se dispondrá de una interfaz gráfica fácil de interactuar con ella, para ello se ha requerido del diseño de todos los formularios con descripciones y cabeceras de las funcionalidades a ejecutar.

Tanto los botones de comandos, como los cuadros de texto tendrán un índice de tabulación consecutivo en el diseño para facilitar también su seguimiento.

Los recursos necesarios para el correcto funcionamiento de nuestra solución son básicos, ya que esta corre bajo cualquier ordenador que disponga de Microsoft Access instalado, con un hardware básico disponible.

5.3.1.2.3. *Requisitos de verificación.*

Para que la solución sea válida, se debe comprobar la integridad de los datos obtenidos en cada una de las funcionalidades, para ello se realizaran pruebas unitarias en cada uno de los módulos disponibles.

5.3.1.2.4. *Requisitos de aceptación.*

Una vez que se hayan realizado las pruebas de verificación, dejaremos al personal de RR.HH. que interactúe con la solución, para que vean que funcione correctamente y que de esta manera sea aceptada nuestra solución.

5.3.1.2.5. Requisitos de documentación.

Deberemos de especificar en todo momento, comentarios en el código acerca de los pasos que hemos realizado por si hay futuras modificaciones.

También deberemos mantener actualizado el manual del usuario sobre todo cuando se hagan nuevos procesos en nuestra solución.

5.3.1.2.6. Requisitos de portabilidad.

Ya que nuestra solución está desarrollada para Microsoft Access, deberá ser portable a cualquier ordenador que cumpla unas mínimas características hardware. Esto significa que podremos ejecutar la aplicación en cualquier ordenador de la empresa independientemente del software que tenga instalado previamente.

5.3.1.3. Tarea DSI 1.6: Especificación del entorno tecnológico.

Nuestro entorno de trabajo debe tener unas características mínimas, que no son demasiado inusuales, para poder utilizar nuestra solución, como en cualquier ordenador de sobremesa. No hace falta mucho espacio en disco, ni tampoco una gran capacidad de memoria RAM, es decir en un disco duro de 80 Gb tendríamos de sobra, y con 1 Gb de memoria RAM también.

En cuanto al tener cargada la aplicación en un servidor de archivos, deberíamos contar por lo menos con una velocidad para la red Ethernet mínimo de 100 Mb/s, y recomendable 1 Gb/s, ya que se podrá conectar remotamente a la aplicación desde otros lugares fuera de lo que es el propio servidor de ficheros donde estará alojada.

Productos

De entrada

- Descripción General del Entorno Tecnológico del Sistema (ASI 1.2)
- Catálogo de Requisitos (DSI 1.2)
- Diseño de la arquitectura del sistema (DSI 1.5)

En Diseño Estructurado:

- Matriz de Procesos / Localización Geográfica (ASI 7.1)
- Plan de Migración y Carga Inicial de Datos (ASI 6.4)

En Diseño Orientado a Objetos:

- Plan de Migración (DSI 4.7)

De salida

- Entorno Tecnológico del Sistema:
 - Especificación del Entorno Tecnológico
 - Restricciones Técnicas
 - Estimación de Planificación de Capacidades

Prácticas

- Sesiones de Trabajo
- Diagrama de Representación

Participantes

- Equipo de Arquitectura
- Equipo de Soporte Técnico

5.3.1.3.1. Hardware.

Dentro de la empresa se dispone:

- Servidor de ficheros.
- Servidor de correo
- Intranet corporativa

A nivel local cada usuario dispone de un ordenador personal y/u ordenador portátil con unas características internas de gama media.

5.3.1.3.2. Software.

Los servidores de la empresa están configurados en Windows 2003 Server, con un servidor de correo de Microsoft Exchange.

Los ordenadores personales disponen de sistemas operativos Windows XP / Windows Vista. Dentro de estos, podemos encontrar instalado Microsoft Office Professional 2003.

El sistema gestor de las copias de seguridad será VERITAS Backup Exec 10.0 for Windows Servers.

5.3.1.4. Tarea DSI 1.7: Especificación de requisitos de operación y seguridad.

5.3.1.4.1. Requisitos de operación.

La solución deberá estar disponible en horario de trabajo, es decir de 8:30 a 18:00 de la tarde, aumentando siempre esta franja unas 2 horas, por si alguien la utiliza antes o después, teniendo siempre un periodo crítico entre las 9:00 y 12:00 que es cuando están todos los usuarios, y empiezan a usar la base de datos.

Las copias de seguridad se harán diariamente, mediante VERITAS Backup Exec 10.0 for Windows Servers, al estar nuestra solución guardada en el servidor de RR.HH, contando con la salvaguarda de la información de manera correcta. Si sucediese algún problema por pérdida de datos o algún problema relacionado con las bases de datos, siempre podíamos restaurar la copia de seguridad del día anterior.

Contaremos también dentro de la solución con la generación de algunos logs, sobre todo en los momentos más críticos, como son el envío de las convocatorias de los cursos a los asistentes, para controlar que se han enviado a todos.

Productos

De entrada

- Catálogo de Requisitos (DSI 1.2)
- Diseño de la Arquitectura del Sistema (DSI 1.5)
- Entorno Tecnológico del Sistema (DSI 1.6)

De salida

- Procedimientos de Seguridad y Control de Acceso
- Procedimientos de Operación y Administración del Sistema

Prácticas

- Sesiones de Trabajo
- Catalogación

Participantes

- Equipo de Seguridad

- Equipo de Arquitectura
- Equipo de Soporte Técnico

5.3.1.5. Tarea DSI 1.9 (DSI-SEG 2.1): Análisis de los riesgos del entorno tecnológico.

Como principales riesgos podemos encontrar:

- Identificación de usuario: en principio cualquier persona que pueda acceder a la ubicación de nuestra solución podría ejecutarla sin ningún tipo de restricción, pudiendo encontrar problemas de acceso no deseado.
- Mantenimiento: como es una aplicación departamental, encontraremos dificultades a la hora de mantener la solución, ya que podrá sufrir muchos cambios inesperados dependiendo de las necesidades que se tengan.
- Integridad de los datos: al tratarse de una aplicación monousuario, dependiendo de las ejecuciones que tengamos en el mismo instante, podríamos sufrir inconsistencia de los datos por posibles modificaciones que se produzcan a la vez.

Productos

De entrada

- Diseño de la Arquitectura del Sistema (DSI 1.5)
- Entorno Tecnológico del Sistema (DSI 1.6)

De salida

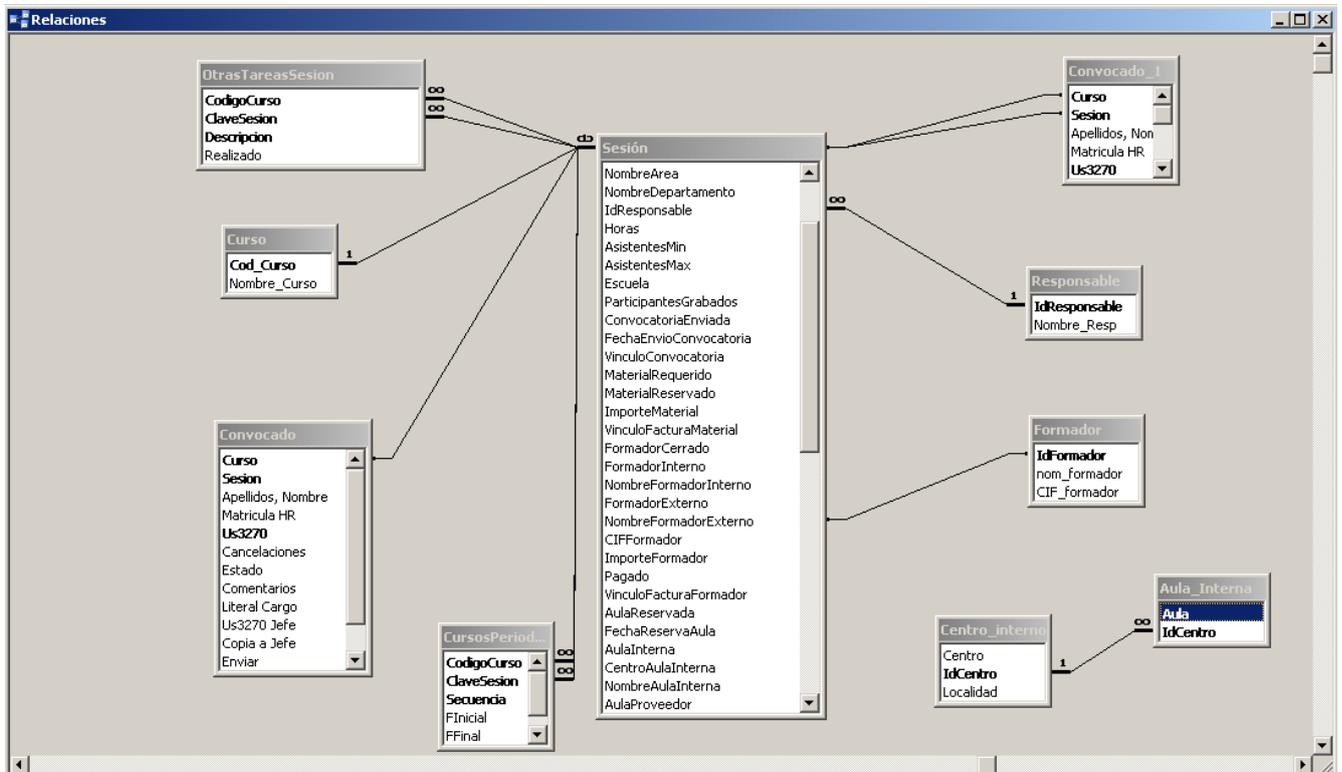
- Requisitos de Seguridad del Entorno Tecnológico

Participantes

- Equipo de Seguridad
- Responsable de Seguridad

5.3.2. Actividad DSI 6: Diseño físico de datos.

5.3.2.1. Tarea DSI 6.1: Diseño del modelo físico de datos.



Relaciones entre tablas

Tipos de datos usados:

Alarmas:

Nombre del campo	Tipo de datos
CodigoCurso	Texto
ClaveSesion	Texto
Texto	Texto
TipoAlarma	Número

AlarmasDefinidas:

Nombre del campo	Tipo de datos
Texto	Texto
Validación	Texto
IFI	Numero
CFI	Número
IFF	Número
CFF	Número

Aula_Interna:



Nombre del campo	Tipo de datos
Aula	Texto
IdCentro	Número

Buzón:

Nombre del campo	Tipo de datos
Nombre	Texto
Buzón	Texto
Correo	Texto

Centro_Interno:

Nombre del campo	Tipo de datos
IdCentro	Autonumérico
Centro	Texto
Localidad	Texto

Convocado:

Nombre del campo	Tipo de datos
Curso	Texto
Sesion	Texto
US3270	Texto
Apellidos, Nombre	Texto
Matricula HR	Texto
Cancelaciones	Número
Estado	Texto
Comentarios	Texto
Literal Cargo	Texto
US3270 Jefe	Texto
Copia a Jefe	Sí/No
Enviar	Sí/No
SEXO	Texto
Autobus	Sí/No

Curso:

Nombre del campo	Tipo de datos
Cod_Curso	Texto
Nombre_Curso	Texto



Envío:

Nombre del campo	Tipo de datos
Curso	Texto
Sesion	Texto
Fecha	Fecha/Hora
Hora	Fecha/Hora

EstadosPeticones:

Nombre del campo	Tipo de datos
Estado	Número
Descripción	Texto

Formador:

Nombre del campo	Tipo de datos
Informador	Autonumérico
nom_Formador	Texto
CIF_Formador	Texto

OtrasTareasSesion:

Nombre del campo	Tipo de datos
CodigoCurso	Texto
ClaveSesion	Texto
Descripción	Texto
Realizado	Sí/No

PlanFormación:

Nombre del campo	Tipo de datos
IdEmpleado	Texto
Curso	Texto
Estado	Número
Itinerario	Texto
Fecha Est	Fecha/Hora
Clasificación	Texto
US3270	Texto
SEXO	Texto
PDF	Sí/No
Email	Sí/No
Jefe	Texto



Responsable:

Nombre del campo	Tipo de datos
IdResponsable	Texto
Nombre_Resp	Texto

Sesion:

Nombre del campo	Tipo de datos
CodigoCurso	Texto
ClaveSesion	Texto
NombreSesion	Texto
FechaInicio	Fecha/Hora
FechaFin	Fecha/Hora
NombreArea	Texto
NombreDepartamento	Texto
Irresponsable	Número
US3270 Jefe	Texto
Horas	Número
AsistentesMin	Número
AsistentesMax	Número
Escuela	Texto
ParticipantesGrabados	Sí/No
ConvocatoriaEnviada	Sí/No
FechaEnvioConvocatoria	Fecha/Hora
VinculoConvocatoria	Texto
MaterialRequerido	Sí/No
MaterialReservado	Sí/No
ImporteMaterial	Número
VinculoFacturaMaterial	Texto
FormadorCerrado	Sí/No
FormadorInterno	Sí/No
NombreFormadorInterno	Texto
FormadorExterno	Sí/No
NombreFormadorExterno	Número
CIFFormador	Texto
ImporteFormador	Número
Pagado	Sí/No
VinculoFacturaFormador	Texto
AulaReservada	Sí/No
FechaReservaAula	Fecha/Hora
AulaInterna	Sí/No
CentroAulaInterna	Número
NombreAulaInterna	Texto
AulaProveedor	Sí/No
DirecciónAulaProveedor	Texto
CosteProveedor	Sí/No



ImporteProveedor	Número
VinculoFacturaProveedor	Texto
RestauranteRequerido	Sí/No
RestauranteReservado	Sí/No
NombreRestaurante	Texto
ImporteRestaurante	Número
VinculoFacturaRestaurante	Texto
TransporteRequerido	Sí/No
TransporteReservado	Sí/No
ImporteTransporte	Número
VinculoFacturaTransporte	Texto
Subvencionado	Sí/No
Forcem	Sí/No
Programa	Sí/No
ListadoFirmas	Sí/No
Cuestionarios	Sí/No
Diplomas	Sí/No
ControlGestion	Sí/No
FechaEnvioControlGestion	Fecha/Hora
CuestionariosEnviados	Sí/No
FechaEnvioCuestionarios	Fecha/Hora
VinculoValoraciones	Texto
CuestionariosRecibidos	Sí/No
FechaReciboCuestionarios	Fecha/Hora
ValoracionCurso	Número
SoftwareRequerido	Sí/No
SoftwareComprobado	Sí/No
DetHorario	Texto
IndCalendario	Sí/No

TB_PERSONAS:

Nombre del campo	Tipo de datos
fechadatos	Texto
ID PEOPLE	Texto
NOMRE	Texto
SEXO	Texto
EDAD_07	Número
MATRICULA HR	Texto
US3270	Texto
NIVEL_FORM	Texto
FORMACION	Texto
CATEG_OFIC	Texto
NOMBRE CATEGORIA	Texto
CARGO	Texto
LITCARGO	Texto
Fecha Cargo	Fecha/Hora



MAT_JEFE	Texto
NOMBRE_JEFE	Texto
BANCO	Texto
LIT_BANCO	Texto
OFICINA	Texto
LBco-Ofic	Texto
CTR_TRABAJ	Texto
Centro	Texto
Zona Gestion	Texto
Literal Zona Gestion	Texto
NIVEL 1	Texto
Literal Nivel 1	Texto
NIVEL 2	Texto
Literal Nivel 2	Texto
NIVEL 3	Texto
Literal Nivel 3	Texto
NIVEL 4	Texto
Literal Nivel 4	Texto
NIVEL 5	Texto
Literal Nivel 5	Texto
NIVEL 6	Texto
Literal Nivel 6	Texto
TIPO_CONTR	Texto
TIP_POS	Texto
FINGRESO	Fecha/Hora
DIRECTIVO	Texto
DNI	Texto
LETRADNI	Texto
FECHANACIM	Texto
MUNIC_NACIM	Texto
PAIS_NACIM	Texto
POSICION	Texto
FECHA_POSICION	Fecha/Hora
CARGO_2006	Texto
LITCORCG	Texto
POSICION_2006	Texto
F_POSICION_2006	Fecha/Hora
UO_2006	Texto
PROV SS EMP	Texto
NUM SS EMP	Texto
NUM COTIZA	Texto

Productos

De entrada

- Características Específicas del SGBD o Sistemas de Ficheros a Utilizar (externo)

En Análisis Estructurado:

- Modelo Lógico de Datos Normalizado (ASI 9.3)
- Plan de Migración y Carga Inicial de Datos (ASI 6.4)

En Análisis Orientado a Objetos:

- Modelo de Clases de Diseño (DSI 4.6)
- Plan de Migración y Carga Inicial de Datos (DSI 4.7)

De salida

- Modelo Físico de Datos

Técnicas

- Reglas de Obtención del Modelo Físico a partir del Lógico
- Reglas de Transformación

Participantes

- Equipo de Arquitectura
- Equipo del Proyecto
- Administradores de Bases de Datos

5.3.2.2. Tarea DSI 6.2: Especificación de los caminos de acceso a los datos y optimización.

Todas nuestras tablas estarán disponibles a través de nuestra solución en el servidor de RR.HH, por tanto el acceso se hará por medio de la red de área local.

Al no tener demasiados accesos simultáneos a la base de datos, se podrá acceder a ella por medio de bloqueos, para asegurar la integridad y coherencia de los datos.

Tenemos que cuidar sobre todo el acceso a la tabla “TB_PERSONAS”, ya que es una tabla que contiene información muy sensible.

Productos

De entrada

- Modelo Físico de Datos (DSI 6.1)
- Diseño Detallado de Subsistemas de Soporte (DSI 2.1)

En Diseño Estructurado:

- Diseño de la Arquitectura Modular del Sistema (DSI 5.2)

En Diseño Orientado a Objetos:

- Modelo de Clases de Diseño (DSI 4.6)

De salida

- Especificación de los Caminos de Acceso a los Datos

Prácticas

- Cálculo de Accesos Físicos
- Caminos de Acceso

Participantes

- Equipo del Proyecto

5.3.2.3. Tarea DSI 6.3: Especificación de la distribución de datos.

En esta tarea cabe destacar, el alto crecimiento de la tabla “TB_PERSONAS”, la cual podría reducirse solo componiéndose por los campos que usamos actualmente y los que podríamos llegar a necesitar, desechando campos que a lo mejor nunca usaríamos.

Alguna otra tabla como las creadas temporalmente para la gestión de las alarmas, contiene campos redundantes que podrían llegar a sacarse a partir de las tablas “Convocados” y “Sesiones” y “AlarmasDefinidas”, pero que por motivos de rapidez se ha decidido crearlas.

Productos

De entrada

- Catálogo de Requisitos (DSI 1.2)
- Modelo Físico de Datos (DSI 6.1)

- Especificación de los Caminos de Acceso a los Datos (DSI 6.2)

De salida

- Modelo Físico de Datos Optimizado

Técnicas

- Optimización

Participantes

- Equipo de Arquitectura
- Equipo del Proyecto
- Administradores de Bases de Datos
- Equipo de Seguridad

5.3.3. Actividad DSI 8: Generación de especificaciones de construcción (Diseño detallado)

5.3.3.1. Tarea DSI8.1: Especificación del entorno de construcción.

La construcción de la solución ha sido llevada a cabo en Visual Basic para aplicaciones. Dentro de ella se ha programado en un entorno orientado a eventos, basado en formularios, los cuales irán formando los distintos módulos de los que consta nuestra solución.

Productos

De entrada

- Catálogo de Requisitos (DSI 1.2)
- Diseño de la Arquitectura del Sistema (DSI 7.2)
- Entorno Tecnológico del Sistema (DSI 7.2)

De salida

- Especificaciones de Construcción del Sistema de Información
- Especificación del Entorno de Construcción



Participantes

- Equipo de Arquitectura
- Equipo del Proyecto
- Equipo de Soporte Técnico
- Equipo de Sistemas
- Equipo de Seguridad

5.3.3.1.1. Hardware.

La solución estará alojada en el servidor IBM de RR.HH, del cual contamos con un sistema de backups en cinta.

Los usuarios necesitarán un ordenador de sobremesa o portátil básico, con tarjeta de red Ethernet, para acceder a la solución mediante la red de área local.

5.3.3.1.2. Software.

En principio deberemos tener como sistema operativo Microsoft Windows XP / Vista, y como suite ofimática Microsoft Office 2003 / 2007. Con todo esto más algún paquete opcional para cargar un control específico sería suficiente.

5.4. Documento de construcción del sistema.

5.4.1. Actividad CSI 1: Preparación del entorno de generación y construcción.

5.4.1.1. Tarea CSI 1.1: Implantación de la base de datos física o fichero.

Partimos de que nuestra solución va a ser implantada en un servidor Windows 2003 Server, en el cual ya están predefinidos los datos relativos al tamaño de asignación de los archivos, el espacio disponible para el sistema gestor de la base de datos y las zonas de desbordamiento disponibles.

La carga de datos en las distintas tablas se hará de manera incremental, metiendo primero datos de cursos relativamente pequeños, para así ir aumentando la carga de cursos con más contenido.

Productos

De entrada

- Entorno Tecnológico del Sistema (DSI 7.2)
- Diseño de la Arquitectura del Sistema (DSI 7.2)
- Especificaciones de Construcción del Sistema de Información (DSI 8.4)

De salida

- Base de Datos Física o Sistema de Ficheros

Participantes

- Equipo del Proyecto

- Administradores de Bases de Datos

5.4.1.2. Tarea CSI 1.2: Preparación del entorno de construcción.

Todos los controles añadidos mediante ActiveX serán incluidos en esta parte, pudiendo instalar aparte de Microsoft Visual Basic para aplicaciones Office, la interfaz de programación Microsoft Visual Basic 2005.

En los puestos de trabajo únicamente debemos contar con Microsoft Windows instalado y Microsoft Office en cualquiera de sus distintas versiones, teniendo como mínimo Windows XP y Microsoft Office 2003.

Como los puestos de trabajo trabajaran contra el servidor, se procederá a la implantación en el servidor del sistema gestor de los backups que se realizarán diariamente.

Productos

De entrada

- Especificaciones de Construcción del Sistema de Información (DSI 8.1)

De salida

- Entorno de Construcción

Participantes

- Equipo del Proyecto
- Técnicos de Sistemas
- Técnicos de Operación
- Administradores de Bases de Datos

5.4.2. Actividad CSI 2: Generación del código de los componentes y de los procedimientos.

5.4.2.1. Tarea CSI 2.1: Generación del código de los componentes.

El código fuente desarrollado para nuestra solución se encuentra situado en el apartado 7 del índice, ANEXOS.

Productos

De entrada

- Catálogo de Normas (DSI 1.4)
- Especificaciones de Construcción del Sistema de Información (DSI 8.2)

De salida

- Producto Software
 - Código Fuente de los Componentes

Participantes

- Programadores

5.5. Manual de usuario.

Trataremos de explicar en este breve manual el funcionamiento de la solución desarrollada, partiendo del menú principal, encontraremos las siguientes opciones: (Figura 1)



Figura 1: Menú principal



1. Alta de sesiones.
2. Inscripción de alumnos.
3. Alertas.
4. Seguimiento.
5. Calendario Intranet.



5.5.1. Alta de sesiones:

Alta de Sesiones - Alertas Formación

Departamento de Formación **Gestión de Convocatorias**

DATOS IDENTIFICATIVOS SESIÓN

Nombre: ... Clave: Fecha Ini: 29/11/2009 Fecha Fin: 29/11/2009

Centro F: Det. horario:

Área: Dpto: Responsable: Participantes Grabados

Sesión: Horas: Asistentes: Min: Max: Acuden: 0 Sin Conf.: 0

Datos Generales | Otros Datos | Detalles Fechas

CONVOCATORIA Convocatoria Enviada Fecha Envío: 29/11/2009

MATERIAL No se Requiere Material Reservado Importe:

Se Requiere

FORMADOR Formador Cerrado

Interno Nombre:

Externo Proveedor: ...

CIF: Importe: Pagado

AULA Aula Reservada Fecha Reserva: 29/11/2009

Interna Centro: Aula:

Proveedor Dirección:

Coste Importe:

RESTAURANTE Restaurante Reservado

No se Requiere Lugar: Importe:

Se Requiere

TRANSPORTE No se Requiere Tpte. Reservado Importe:

Se Requiere

Figura 2: Gestión de Convocatorias

5.5.1.1. *Crear nueva sesión.*

Para crear una sesión de un curso nueva, introduciremos los datos relativos a la sesión en los siguientes apartados según muestra la Figura 2:

- Datos identificativos a la sesión: comprende desde el nombre del curso y la sesión en la que nos encontramos, hasta las fechas de inicio y finalización, horarios, responsable, horas de duración del curso, centro de formación, área a la que pertenece el curso, departamento y asistentes que acudirán a dicho curso.

La gestión de los asistentes del curso se verá más detalladamente en el Apartado 2 de este manual: Inscripción de alumnos. A esta funcionalidad podremos acceder de 2 maneras:

- o Desde el menú principal, teniendo que seleccionar el curso y la sesión con la que queramos trabajar, y a partir de ahí gestionando la convocatoria de los asistentes.
- o Desde el botón Gestionar Convocatorias: , dentro de Datos identificativos de la sesión y Asistentes. Aquí el curso y la sesión vendrán determinados según el curso con el que estemos trabajando en la Gestión de la convocatoria, y no podrán ser alterados en ningún momento, ya que sino no perderíamos la coherencia de los datos.

Para trabajar con las distintas sesiones y los distintos cursos, deberemos acceder desde el menú principal. (Figura 1).

- Convocatoria: accederemos mediante el botón Convocar al curso: , seleccionaremos aquellos empleados que queramos que asistan al curso o los preseleccionaremos en la lista de candidatos para ver aquellos que pueden optar a la realización del curso. (Figura 3):

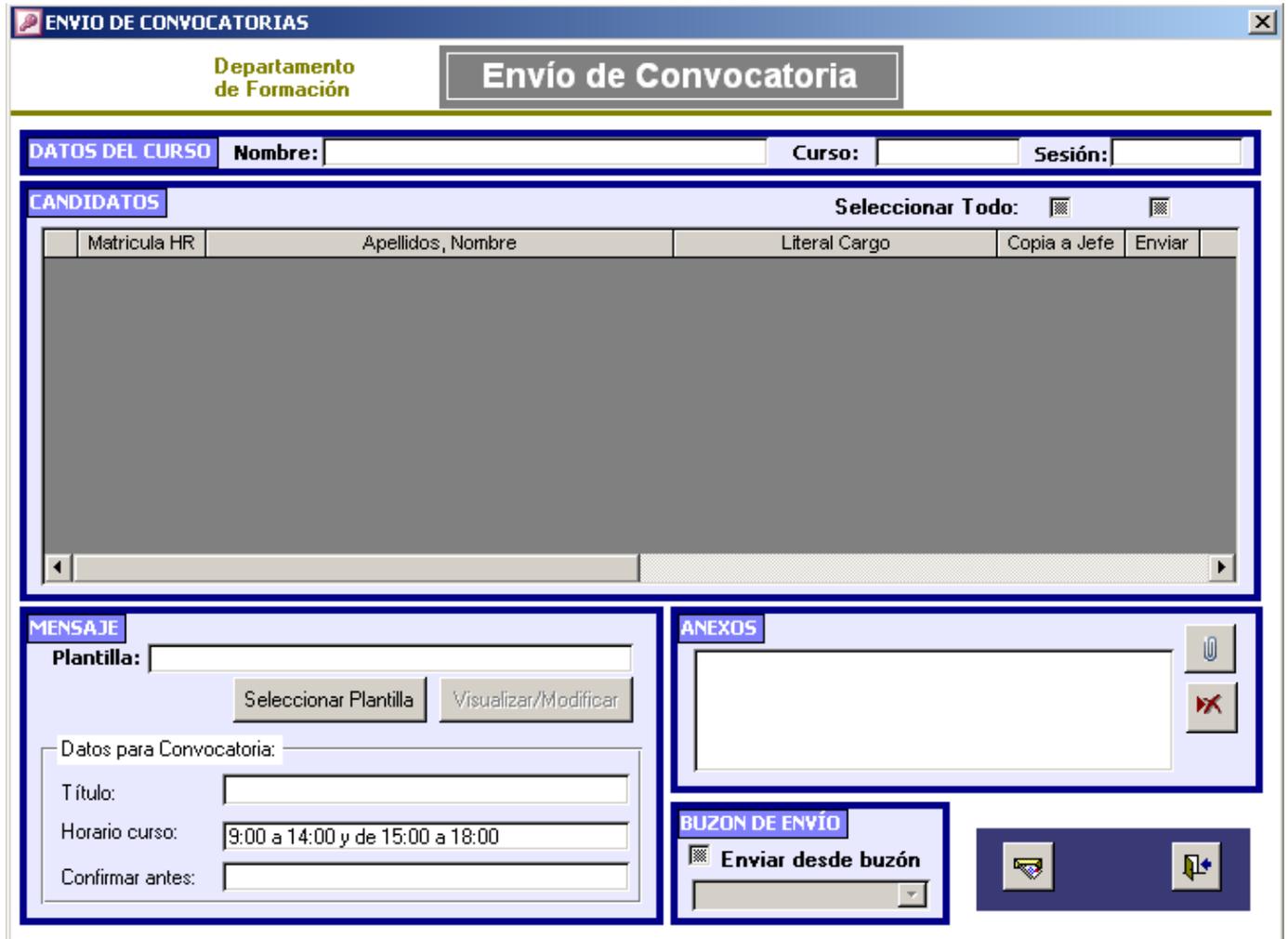


Figura 3: Envío de Convocatoria

- Podremos seleccionar aquellos candidatos a los que queremos que se le envíe la convocatoria por correo electrónico, por defecto se mandará la convocatoria a todos los empleados que hayan sido convocados al curso.

- Seleccionaremos el estilo de plantilla con el que queremos que se envíe nuestro mensaje, dentro de las plantillas disponibles, para ello en el apartado mensaje podremos seleccionar la plantilla o visualizarla para ver cómo queda, incluso modificar la plantilla, pulsando en los botones Seleccionar Plantilla o Visualizar / Modificar respectivamente.

En este apartado también podremos hacer modificaciones en el Título, Horario del curso y Fecha de confirmación del curso.

- Como anexos añadiremos los archivos que necesitamos que se adjunten al mensaje o quitarlos del mensaje desde:
 - el botón incluir anexo a la convocatoria:



- o el botón eliminar anexo:



- En buzón de envío, podemos seleccionar un buzón de correo distinto del que tenemos actualmente.

Por último procederemos a enviar la convocatoria pulsando en el botón:

Enviar correo a los seleccionados:



Y cuando hayamos terminado pulsando el botón Cerrar ventana:



- Material: Introduciremos el importe del material si se requiere para el curso en el que nos encontremos.

Podemos ver la factura en el caso de que haya sido introducida desde el botón: 

- Formador: Introduciremos el tipo de formador que impartirá el curso, poniendo el nombre en el caso de que sea un formador interno, y el proveedor, CIF, importe, si nos encontramos con un formador externo.

Podemos seleccionar mediante el botón: , algunos de los proveedores que tenemos disponibles.

Podemos ver la factura en el caso de que haya sido introducida desde el botón: 

- Aula: Introduciremos la fecha en la cual el aula ha quedado reservada, poniendo el nombre del centro y el aula en el caso de que sea un aula interna, y en el caso de que este fuera, el proveedor, y si tiene coste, su importe.

Podemos ver la factura en el caso de que haya sido introducida desde el botón: 

- Restaurante: Introduciremos el lugar y el importe del restaurante, en el caso de que haya sido reservado.

Podemos ver la factura en el caso de que haya sido introducida desde el botón: 

- Transporte: Introduciremos el importe del transporte en el caso de que haya sido reservado.

Podemos ver la factura en el caso de que haya sido introducida desde el botón: 



Alta de Sesiones - Alertas Formación

Departamento de Formación **Gestión de Convocatorias**

DATOS IDENTIFICATIVOS SESIÓN

Nombre: ... Clave: Fecha Ini: 29/11/2009 Fecha Fin: 29/11/2009

Centro F: Det. horario:

Área: Dpto: Responsable: Participantes Grabados

Sesión: Horas: Asistentes: Min: Max: Acuden: 0 Sin Conf.: 0

Datos Generales Otros Datos Detalles Fechas

SUBVENCIÓN Y ARCHIVO

NO Subvencionado Listado de Firmas

FORCEM Cuestionarios Control de Gestión

Programa Diplomas Fecha Envío CG: 29/11/2009

CUESTIONARIOS

Enviado a Grabar Fecha Envío: 29/11/2009

Resultados Recibidos Fecha de Recibo: 29/11/2009 Valoración:

OTRAS TAREAS

	Descripcion	Realizado
<input type="checkbox"/>		<input checked="" type="checkbox"/>

SOFTWARE

No se Requiere

Se Requiere Comprobado

Figura 4: Otros datos

- En la siguiente pestaña Otros Datos
 - o Subvención y archivo: Aquí podremos informar de si el curso en cuestión es o no subvencionado, si tiene listado de firmas, programa, cuestionarios, diplomas y control de la gestión y la fecha de envío del control de la gestión

Cuestionarios: informaremos de si el curso a enviado a grabar, los resultados y la fecha de envío. También de si los resultados han sido recibidos, junto con su fecha y la valoración. Podremos consultar la factura en el caso de que haya sido introducida desde el botón: 

- o Software: pondremos si se requiere o no software para la realización del curso, y si se ha comprobado la necesidad de utilizar software en ese curso.
- o Otras Tareas: aquí se definirá cualquier otra tarea que no se haya podido recoger en los apartados anteriores, y que requiera de un control para la realización del curso.



Alta de Sesiones - Alertas Formación

Departamento de Formación **Gestión de Convocatorias**

DATOS IDENTIFICATIVOS SESIÓN

Nombre: ... Clave: Fecha Ini: 29/11/2009 Fecha Fin: 29/11/2009

Centro F: Det. horario:

Área: Dpto: Responsable: Participantes Grabados

Sesión: Horas: Asistentes: Min: Max: Acuden: 0 Sin Conf.: 0

Datos Generales Otros Datos Detalles Fechas

FECHAS DE PERIODOS

	Finicial:	FFinal:	Espaciado
▶			<input type="checkbox"/>

Figura 5: Detalles Fechas

- En la pestaña de Detalles fechas:
 - o Fechas de periodos: Introduciremos una línea por cada periodo de discontinuidad del curso en el tiempo, es decir cuando haya paradas en el curso de uno o varios días, aunque luego posteriormente se prosiga el curso, aquí recogeremos todos estos intervalos.

5.5.1.2. *Guardar cambio de sesión.*

Una vez que hayamos introducido todos los datos referentes a la sesión de un curso nuevo, o hayamos acabado de modificar algún curso creado con anterioridad, podremos grabar los cambios mediante el botón:



5.5.1.3. *Eliminar sesión.*

Cuando queramos borrar una sesión ya creada, primero buscaremos la sesión, y después la borraremos mediante el botón:



5.5.1.4. *Buscar curso / sesión.*

Desde este apartado podremos buscar un curso seleccionándolo de la lista desplegable de los cursos disponibles. Luego podremos hacer una ordenación por los siguientes criterios: (Figura 6)

- Centro de formación
- Fecha
- Subvencionados
- Plantillas de los cursos

Una vez seleccionemos la sesión del curso buscado pulsaremos en el botón Aceptar sesión, para que nos cargue sus datos en el formulario gestión de convocatorias:



Si por el contrario deseamos volver a la pantalla anterior, pulsaremos en el botón Cancelar y volver a sesiones:



Búsqueda de Sesiones de Cursos

SELECCIÓN DE CURSO

Curso:

Centro Formación: Fecha desde: **Mostrar subvencionados**

Mostrar plantillas

SELECCIÓN DE SESIÓN

F. Inicio	F. Fin	Curso:	Sesión	Nombre Sesión	Lugar	Pro

Figura 6: Búsqueda de Sesiones de Cursos

5.5.1.5. *Imprimir diplomas preimpresos.*

Con esta opción podremos imprimir aquellos diplomas de las personas que hayan asistido a los cursos ya finalizados, mediante el botón:



La solución procesará el listado de alumnos y generará mediante una plantilla un diploma en el que consta que el alumno ha asistido al curso, la duración del mismo, y la persona que lo certifica.

5.5.1.6. *Cerrar formulario.*

Para poder regresar al menú principal deberemos pulsar el botón de Cerrar Formulario:





5.5.2. Inscripción de alumnos.

Asistentes : Formulario

Departamento de Formación **Gestión de Candidatos**

DATOS DEL CURSO Curso: Sesión:

INCLUIR ASISTENTE

Matrícula: Usuario:
Apellidos y Nombre :

Lista:

Nombre	Usuario	Nivel 3
--------	---------	---------

CANDIDATOS Y CONVOCADOS

Candidatos: Cancelados: Convocados: Confirmados:

Empleado	Canc	Comentari
----------	------	-----------

Empleado	Estado:	Autobus
----------	---------	---------

Figura 7: Gestión de los candidatos

Para poder seleccionar aquellos empleados que creamos que son los idóneos para la realización del curso, usaremos esta opción. Podremos llegar desde el menú principal (Figura 1) o desde la gestión de las sesiones de los cursos (Figura 2).

Dentro de esta opción distinguiremos las siguientes secciones:

- Datos del curso: esta opción estará deshabilitada si llegamos a ella desde el formulario gestión de las sesiones de los cursos. Si accedemos a ella desde el menú principal, podremos seleccionar el curso del que queramos modificar o incluir los asistentes y sus sesiones correspondientes.

Una vez seleccionado el curso y la sesión se habilitarán los demás apartados, apareciendo los datos que conforman las listas de candidatos y convocados.

- Incluir asistente: utilizaremos esta opción para buscar un empleado que queramos incorporar a la lista de candidatos a realizar el curso, el candidato podrá ser buscado por los siguientes criterios:
 - o Matricula
 - o Usuario
 - o Apellidos y Nombre

Una vez realizada la búsqueda, mediante el botón:



Se mostrarán los resultados en la lista desplegable situada en este apartado. (Figura 7).

Cuando pulsemos sobre un candidato de la lista, se cargarán inmediatamente sus datos, si es el empleado que estábamos buscando como candidato deberemos pulsar el botón añadir asistente a la lista de espera. En este momento el empleado entrará a formar parte de la lista de candidatos que barajamos para su posible incorporación a la lista definitiva de empleados convocado finalmente a la realización del curso.

- Candidatos y convocados: contendrá las listas de candidatos y convocados a realizar el curso.

Estas listas nos informarán en todo momento de algunos datos importantes como son:

- o En la lista de candidatos, se nos informara del número de cancelaciones que ha ejercido un empleado para realizar el curso, y un comentario de porque ha cancelado su asistencia al curso.
- o En la lista de convocados saldrá en todo momento si el empleado ha sido convocado y también si ha confirmado su asistencia. En el caso

de que no confirme su asistencia según los plazos establecidos, se le podrá volver a pasar a la lista de candidatos, para incorporar algún otro empleado. También dispondremos de si el empleado necesita autobús o no para su desplazamiento al curso.

Los candidatos y los convocados podrán pasarse de una lista a otra mediante los botones:

-  Pasará todos los empleados candidatos a la lista de convocados
-  Pasará el empleado candidato seleccionado a la lista de convocados.
-  Pasará el empleado convocado seleccionado a la lista de candidatos.
-  Pasará todos los empleados convocados a la lista de candidatos.

En la lista de candidatos, dispondremos de un botón para borrar al candidato: Suprimir candidato.

A su vez en la lista de convocados disponemos de 2 botones:

- Confirmar asistente: confirmaremos mediante este botón la asistencia de este empleado al curso solicitado.
- Cancelar asistente: cancelará la asistencia de este empleado al curso, poniendo un breve comentario, de porque ha solicitado la cancelación a dicho curso.

Por último podremos regresar al menú principal, cerrando este formulario mediante el botón:





5.5.3. Alertas.

Alarmas

Departamento de Formación

Lista de Alarmas

DATOS DE SELECCIÓN Selecciona Escuela: Administración Central Orden de datos: Fecha Mostrar subvencionados

Alerta:

ID	Curso	Sesión	Nombre Sesión	F. Inicio	F. Fin
1	<input type="checkbox"/> 000237	0308	Técnicas de Aprendizaje Virtual Los Cuestionarios aún no han sido recibidos de EPISE Los cuestionarios aún no han sido enviados a grabar a EPISE Falta la valoración global del curso	16/12/2008	17/12/2008
2	<input type="checkbox"/> 000237	0312	Liderazgo de Gestión del Cambio Todavía no han confirmado todos los convocados al curso Sesiones de seguimiento los días 6, 20 y 27 de Julio para grupo 1. Grupo 2 Días 7, 21 y 28 de 9.00 a 12:00h Sesión coaching martes 29 de septiembre de 09:00 a 12:00h Sesión coaching martes 15 de septiembre de 09:00 a 12:00h	28/05/2009	27/07/2009
3	<input type="checkbox"/> 000822	0087	PRL - EPISE Práctico Los Cuestionarios aún no han sido recibidos de EPISE Los cuestionarios aún no han sido enviados a grabar a EPISE Falta la valoración global del curso	28/05/2009	28/05/2009
4	<input type="checkbox"/> 000237	0314	Jornada de Atención Telefónica Los Cuestionarios aún no han sido recibidos de EPISE Los cuestionarios aún no han sido enviados a grabar a EPISE Falta la valoración global del curso	16/06/2009	16/06/2009
5	<input type="checkbox"/> 000237	0313	Adquira (CIS España) Falta el importe de la factura del restaurante Falta archivo de la factura del restaurante	24/06/2009	24/06/2009
6	<input type="checkbox"/> 000776	0087	Negociación Viernes, 11 de Septiembre en Vaguada 90 minutos Jueves, 10 de Septiembre en Vaguada 90 minutos Enviar recordatorio a los asistentes (Patri). Comprobar in situ que la reserva de Vaguada está efectuada en los carteles de la puerta (Txamo o Edu).	01/07/2009	03/07/2009

Figura 8: Lista de alarmas

Para determinar las alarmas de los próximos cursos que se van a celebrar, seleccionaremos la opción de Alertas (Figura 8).

Dentro de los criterios por los cuales podemos filtrar estas alarmas, encontramos los siguientes:

- Selección de escuela: mostrara las alarmas de los cursos que pertenezcan a esa determinada escuela.
- Orden de datos: podremos hacer una ordenación de manera ascendente o descendente
- Mostrar subvencionados: solo se mostrarán las alarmas de los cursos que tengan algún tipo de subvención.
- .Alerta: seleccionaremos de las alarmas disponibles, aquella por la cual queremos filtrar nuestra selección.

Una vez tengamos nuestros criterios de visualización seleccionados, se irán mostrando en la parte posterior las alarmas y los cursos que cumplen las especificaciones filtradas anteriormente.

Dispondremos de un botón (Expandir todas), que visualizará todas las alertas que pertenecen a los criterios de selección elegidos.

Para salir del listado de alertas pulsaremos el botón:



5.5.4. Calendario Intranet.

CALENDARIO PARA LA INTRANET

Departamento de Formación

Calendario para Intranet

DATOS

Cursos | Previsualización

	F. Inicio	F. Fin	Curso:	Sesión	Nombre Sesión	Lugar	Proveedor	Calendario
▶	03/04/2009	19/12/2009	001597	0001	Master en Sistemas de Información Banca	Aula Proveedor	La Salle - Funitec	<input type="checkbox"/>
	05/10/2009	02/12/2009	001023	0019	ITIL Service Manager (Plan de Desarrollo)	CF. Madrid - Alcobendas	NetMind	<input checked="" type="checkbox"/>

Escuela de tecnología

Previsualizar >>



Figura 9: Calendario para Intranet

Dentro de este apartado podemos generar el calendario de los cursos, que aparecerá en la página de la intranet corporativa.

Podremos filtrar por los cursos que pertenecen a la Escuela de tecnología, pulsando en la parte inferior de la pantalla (Figura 9) sobre el check de Escuela de tecnología.

Una vez seleccionado el criterio o no de si son únicamente los cursos pertenecientes a Escuela de tecnología, podremos visualizar el calendario, pulsando sobre el botón Previsualizar, el cual nos mostrara en la intranet corporativa el calendario de los cursos que se van a celebrar. (Figura 10).

Para salir del calendario para la intranet pulsaremos el botón:





Gestión de la formación: MV3

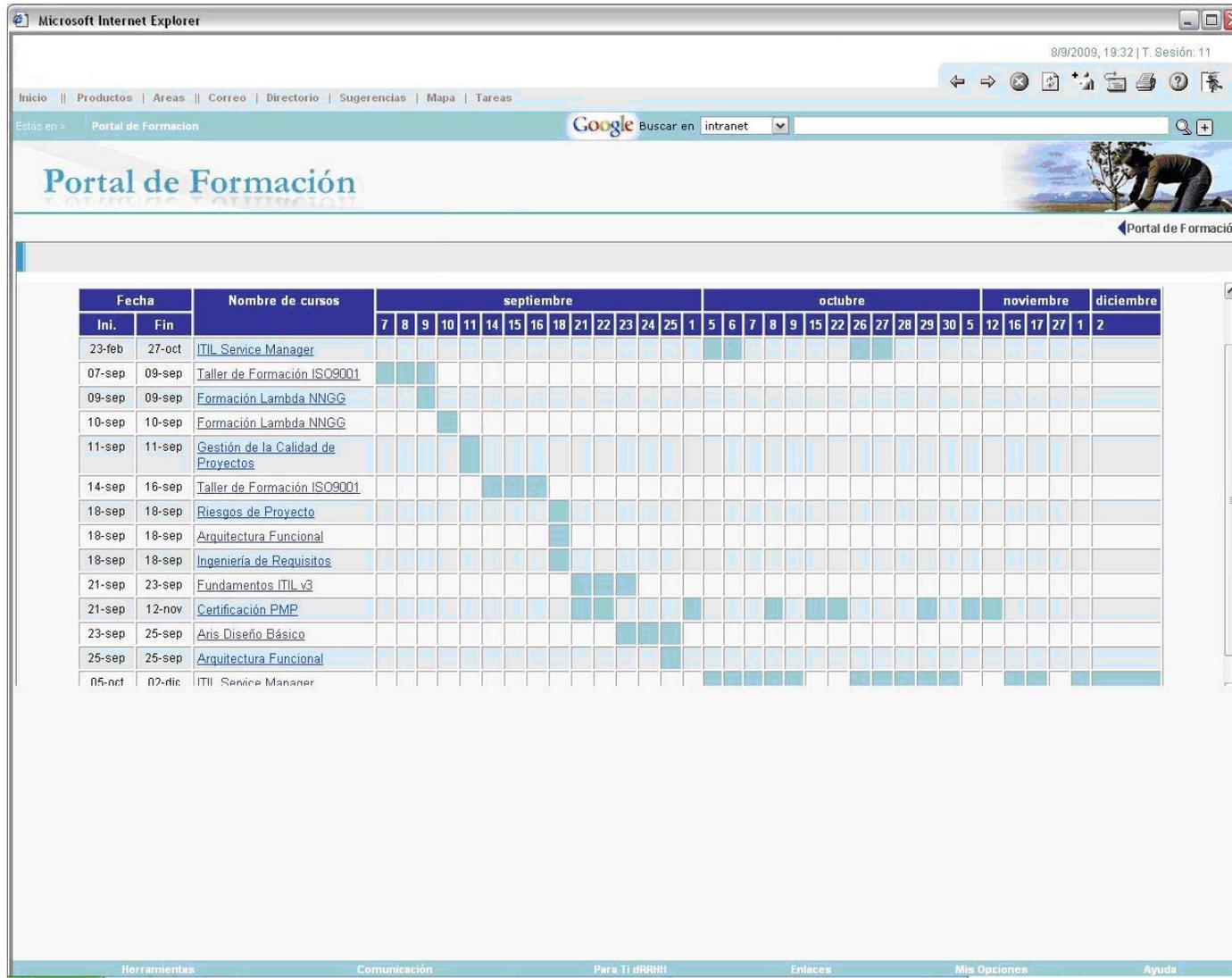


Figura 10: Calendario Intranet

6. Documento de implantación del sistema.

6.1.1. Actividad IAS 1: Establecimiento del plan de implantación.

6.1.1.1. Tarea IAS 1.1: Definición del plan de implantación.

En una primera fase, se procederá a la verificación de los requisitos software que necesitamos para que pueda ejecutarse nuestra solución, comprobaremos que los ordenadores están cargados con sistemas Windows, y que contienen el paquete ofimático Microsoft Office 2003 o superior. De los requisitos hardware como ya hablamos en puntos anteriores, no hace falta que sean grandes ordenadores, con unos componentes básicos podrá funcionar sin problemas.

Como nuestra solución es instalada dentro de las oficinas en nuestra red de área local, no es necesario en un primer momento adaptarla a accesos desde el exterior o desde cualquier otra oficina, contando en todo momento con los datos dentro de nuestros servidores locales.

Una vez haya sido comprobado todo esto en una segunda fase se realizará una a la carga inicial de algunas tablas y procederemos a la vinculación de las tablas externas que utilizamos para que haya una correspondencia real de los datos que usaremos más tarde.

Después realizaremos la carga de una sesión de un curso completo como prueba, para ver si existe algún tipo de problema y dejar la solución preparada para su uso posterior.

Por último realizaremos una sesión de formación con los empleados que van a usar nuestra solución, para ello en una mañana procederemos a la explicación de cómo funciona, los posible problemas que podemos encontrarnos y como se realizan las cosas para que no sucedan errores.

Productos

De entrada

- Solución Propuesta (EVS 6.2)
- Diseño de la Arquitectura del Sistema (DSI 7.2)
- Entorno Tecnológico del Sistema (DSI 7.2)
- Procedimientos de Operación y Administración del Sistema (DSI 1.7)
- Procedimientos de Seguridad y Control de Acceso (DSI 1.7)
- Plan de Migración y Carga Inicial de Datos (DSI 9.4)
- Plan de Pruebas (DSI 10.3)
- Catálogo de Requisitos (DSI 11.2)

De salida

- Plan de Implantación

Prácticas

- Sesiones de trabajo

Participantes

- Jefe de Proyecto
- Responsable de Implantación
- Responsable de Operación
- Responsable de Sistemas
- Directores de los Usuarios

6.1.1.2. Tarea IAS 1.2: Especificación del equipo de implantación.

Para la comprobación o instalación de los requisitos software, necesitaremos un Ingeniero o Técnico superior de Administración de Sistemas Informáticos, el cual se encargara de revisar que los ordenadores contienen los paquetes necesarios instalados para usar nuestra solución.

La carga inicial de datos y la incorporación de la sesión de un curso de prueba al sistema, así como la comprobación del correcto funcionamiento inicial de la solución la llevara a cabo un Ingeniero superior o Ingeniero Técnico en informática de Gestión. También será el encargado del mantenimiento de la misma, solucionando los posibles problemas que surjan cuando evolucione la solución.

En el apartado de formación, podrá realizarlo tanto el Gerente de Formación de Áreas de Apoyo como el encargado del mantenimiento y comprobación del sistema, o ambos, ya que son las personas que poseen más conocimientos acerca de cómo funciona la solución.

Productos

De entrada

- Recursos Humanos y Técnicos Disponibles (externo)
- Catálogo de Usuarios (ASI 1.4)
- Plan de Implantación (IAS 1.1)



De salida

- Equipo de Implantación

Participantes

- Jefe de Proyecto
- Responsable de Implantación
- Responsable de Operación
- Responsable de Sistemas
- Directores de los Usuarios

7. CONCLUSIONES Y EXTENSIONES AL SISTEMA

Con el presente proyecto de fin de carrera se permite la gestión logística de los cursos de formación dentro de cualquier departamento de RR.HH.

Todos los cursos quedarán perfectamente controlados por medio de las alarmas. Éstas nos avisarán de todos los elementos necesarios para que el curso se pueda celebrar sin ninguna complicación. Si se necesita una reserva de aula, restaurante o cualquier otra cosa, se nos avisará dependiendo del tiempo que nos quede para la realización del curso.

Podemos llevar el control de asistencia de los distintos empleados, para obtener información de quien rechaza y quien confirma su presencia, usándola para la convocatoria de las próximas celebraciones de los cursos. De esta manera aquellas personas que rechazan un curso o no confirman la asistencia a un curso de manera justificada, no podrán ser incluidas en la siguiente celebración de un curso, en el caso de que estén todas las plazas cubiertas.

De forma automática cuando se haya decidido qué personas serán las que asisten al curso, se enviará un correo electrónico con la información del curso, que el empleado tendrá que confirmar para su posible incorporación al curso.

Se podrá llevar un control, de los próximos cursos que se van a realizar en un mes determinado. De esta manera tendremos información útil de los siguientes cursos, pudiendo ver el catalogo de cursos que hay disponible.

Como extensiones o futuras mejoras a este proyecto fin de carrera, podemos incluir un histórico de los cursos que ha realizado cada persona, para poder seguir su trayectoria y evolución dentro de la empresa, comprobando si los cursos realizados han sido productivos.

También podemos realizar automáticamente informes semanales con los recursos que todavía no tenemos disponibles de los cursos que se van a celebrar, para poder preparar aquellas tareas que aún quedan pendientes por realizar.

En el apartado de alarmas se pueden llegar a diseñar alarmas de todo tipo, mucho más específicas, controlando cualquier parámetro del curso, como por ejemplo las necesidades de conectividad que se pudiesen tener.



8. ANEXOS

Form Asistentes

Option Compare Database

Public vNombre As String

Public vUsuario As String

Sub ActualizaListas()

Dim vCurso As String

If Curso.ListIndex <> -1 Then vCurso = IIf(IsNull(Curso.Column(1, Curso.ListIndex)), 0, Curso.Column(1, Curso.ListIndex))

'Actualizamos las entradas de las listas de candidatos y convocados.

SUBF_Candidatos.Form.RecordSource = "SELECT * FROM convocado WHERE ((curso="" & vCurso & "") AND (([convocado].[estado]='Seleccionado') OR ([convocado].[estado]='Cancelado')) order by [Apellidos, Nombre];"

SUBF_Candidatos.Requery

SUBF_Convocados.Form.RecordSource = "SELECT * FROM convocado WHERE ((curso="" & vCurso & "") AND (sesion="" & Sesion.Value & "") AND NOT ([convocado].[estado]='Seleccionado')) order by Estado DESC, [Apellidos, Nombre];"

SUBF_Convocados.Requery

'Blanquea variables de selección de empleado

vNombre = ""

vUsuario = ""

If SUBF_Convocados.Form.Recordset.EOF Then

If SUBF_Candidatos.Enabled = True Then SUBF_Candidatos.SetFocus

SUBF_Convocados.Enabled = False

Else

SUBF_Convocados.Enabled = True

End If

If SUBF_Candidatos.Form.Recordset.EOF Then

If matricula.Enabled = True Then matricula.SetFocus

SUBF_Candidatos.Enabled = False

suprimir.Enabled = False

Else

SUBF_Candidatos.Enabled = True

End If

'Inhabilita botones

moverconvcand.Enabled = False

movertodosconvcand.Enabled = False

movertodoscandconv.Enabled = False

movercandconv.Enabled = False

'Actualizamos el campo de Confirmados

Consulta = "SELECT count(Us3270) as TotalAcuden FROM CONVOCADO WHERE ((Curso= "" & vCurso & "") AND (Sesion= "" & Sesion.Value & "") AND (Estado='Confirmado'));"

Set Registro = CurrentDb.OpenRecordset(Consulta, dbOpenDynaset)



```
If Registro.RecordCount = 0 Then
    txtConfirmados = 0
Else
    txtConfirmados = Registro!totalacuden
End If
Registro.Close

' Actualizamos el campo de Convocados
Consulta = "SELECT count(Us3270) as TotalAcuden FROM CONVOCADO WHERE ((Curso= '' &
vCurso & '') AND (Sesion= '' & Sesion.Value & '') AND ((Estado=Convocado) OR
(Estado=Confirmado)));";
Set Registro = CurrentDb.OpenRecordset(Consulta, dbOpenDynaset)
If Registro.RecordCount = 0 Then
    txtConvocados = 0
Else
    txtConvocados = Registro!totalacuden
End If
Registro.Close

' Actualizamos el campo de Candidatos
Consulta = "SELECT count(Us3270) as TotalAcuden FROM CONVOCADO WHERE ((Curso= '' &
vCurso & '') AND (Sesion= '0000') AND ((Estado=Seleccionado) OR (Estado=Cancelado)));";
Set Registro = CurrentDb.OpenRecordset(Consulta, dbOpenDynaset)
If Registro.RecordCount = 0 Then
    txtNumConvocados = 0
Else
    txtNumConvocados = Registro!totalacuden
End If
Registro.Close

' Actualizamos el campo de Cancelados
Consulta = "SELECT count(Us3270) as TotalAcuden FROM CONVOCADO WHERE ((Curso= '' &
vCurso & '') AND (Sesion= '0000') AND Cancelaciones <> 0);";
Set Registro = CurrentDb.OpenRecordset(Consulta, dbOpenDynaset)
If Registro.RecordCount = 0 Then
    txtNumCancelados = 0
Else
    txtNumCancelados = Registro!totalacuden
End If
Registro.Close

End Sub

Sub AsignaEmpleadoASesion(FechaIni As Date, FechaFin As Date, sUsuario As String, sSesion As String,
sCurso As String)
    ' Comprobar si ya está inscrito en otro curso para las mismas fechas
    Consulta = "SELECT * FROM Convocado INNER JOIN Sesión ON (Convocado.Curso =
Sesión.CodigoCurso) AND (Convocado.Curso = Sesión.CodigoCurso) AND (Convocado.Sesion =
Sesión.ClaveSesion) WHERE ClaveSesion <> '0000' and (((# & Format(FechaIni, "yyyy/mm/dd") & "#
Between [FechaInicio] And [FechaFin]) AND (Convocado.Us3270="" & sUsuario & "")) OR
((Convocado.Us3270="" & sUsuario & "") AND (# & Format(FechaFin, "yyyy/mm/dd") & "# Between
[FechaInicio] And [FechaFin]))) ORDER BY Sesión.FechaInicio DESC;";
    Set Registro = CurrentDb.OpenRecordset(Consulta, dbOpenDynaset)
    If Registro.RecordCount <> 0 Then
        If MsgBox("El empleado " & Registro![Apellidos, Nombre] & " ya está inscrito en el curso: " & vbCr
& """" & Registro!NombreSesion & """" (" & Registro!Curso & "-" & Registro!Sesion & ")." & vbCr & vbCr
& vbCr & vbCr & "¿Desea realmente inscribirlo/a en el curso?.", vbQuestion + vbYesNo, "Error al agregar")
= vbNo Then
```



```
        Registro.Close
        suprimir.Enabled = False
        SUBF_Candidatos.SetFocus
        Exit Sub
    End If
End If
Registro.Close

' Asigna a una sesión y cambia el estado
CurrentDb.Execute "update Convocado set Estado = 'Convocado', Sesion = '" & sSesion & "' WHERE
Sesion='0000' and Curso = '" & sCurso & "' AND estado in ('Seleccionado', 'Cancelado') and Us3270 = '" &
sUsuario & "'"
End Sub

'Cancelamos el empleado seleccionado modificando su estado en el curso como "Cancelado".
Private Sub cancelar_Click()
    Dim index As Integer

    'Comprobamos que hay al menos 1 candidato en la lista y que alguno este seleccionado.

    ' If convocados.ListCount > 1 And convocados.ListIndex <> -1 Then
    If vUsuario <> "" Then
        If MsgBox("¿Cancelar al candidato: " & vNombre & "?", vbYesNo + vbQuestion, "Cancelar
Candidato") = vbNo Then Exit Sub
        comentario = InputBox("Introduzca los motivos de cancelación para: " & vNombre, "Comentario
cancelación candidato", "Sin Comentarios")
        If comentario <> "" Then
            ' Asigna a una sesión y cambia el estado
            CurrentDb.Execute "update Convocado set Sesion = '0000', Estado = 'Cancelado', Comentarios = '" &
comentario & "', Cancelaciones = Cancelaciones + 1 WHERE Sesion = '" & Sesion.Value & "' and Curso = '"
& Curso.Column(1, Curso.ListIndex) & "' AND estado in ('Convocado', 'Confirmado') and Us3270 = '" &
vUsuario & "'"

            SUBF_Candidatos.SetFocus
            ActualizaListas
        End If
    Else
        MsgBox "Debes seleccionar primero un convocado.", vbCritical, "Sin selección"
    End If
End Sub

Private Sub candidatos_Click()
    If SUBF_Candidatos.Form.Recordset.EOF Then Exit Sub
    moverconvcand.Enabled = False
    movertodosconvcand.Enabled = False
    confirmar.Enabled = False
    cancelar.Enabled = False
    movertodoscandconv.Enabled = True
    movercandconv.Enabled = True
    suprimir.Enabled = True
End Sub

Private Sub CMB_Buscar_Click()

    Dim Consulta As String
    Dim BD As DAO.Database
    Dim Registro As DAO.Recordset
```



```
Screen.MousePointer = 11
If matricula.Value <> "" Then
    Consulta = " [ID People] Like '*' & matricula.Value & '*'"
End If
If nombre.Value <> "" Then
    If matricula.Value <> "" Then
        Consulta = Consulta & " AND"
    End If
    Consulta = Consulta & " [Nombre] Like '*' & nombre.Value & '*'"
End If
If usuario.Value <> "" Then
    If (matricula.Value <> "") Or (nombre.Value <> "") Then
        Consulta = Consulta & " AND"
    End If
    Consulta = Consulta & " [US3270] Like '*' & usuario.Value & '*'"
End If
If Consulta <> "" Then
    Consulta = "SELECT TB_PERSONAS.NOMBRE as Nombre, TB_PERSONAS.US3270 as Usuario,
TB_PERSONAS.[NNIVEL 3] as [Nivel 3] FROM TB_PERSONAS WHERE " & Consulta & " ORDER BY
[Nombre]"
    lista.RowSource = Consulta
    If lista.ListCount = 0 Then
        Screen.MousePointer = 1
        MsgBox "No se ha encontrado a nadie que cumpla con los requisitos de búsqueda", 0 + 16, "Búsqueda
vacía"
    End If
End If
Screen.MousePointer = 1

End Sub

'Confirmamos el empleado seleccionado modificando su estado en el curso como "Confirmado".
Private Sub confirmar_Click()

    'Comprobamos que hay al menos 1 candidato en la lista y que alguno este seleccionado.
    ' If convocados.ListCount > 1 And convocados.ListIndex <> -1 Then
    If vUsuario <> "" Then
        If MsgBox("¿Confirmar al candidato: " & vNombre & "?", vbYesNo + vbQuestion, "Confirmar
Candidato") = vbNo Then Exit Sub
        ' Asigna a una sesión y cambia el estado
        CurrentDb.Execute "update Convocado set Estado = 'Confirmado' WHERE Sesion = " & Sesion.Value
& " and Curso = " & Curso.Column(1, Curso.ListIndex) & " AND estado = 'Convocado' and Us3270 = " &
vUsuario & ""

        Form_Asistentes.SetFocus
        ActualizaListas
    Else
        MsgBox "Debes seleccionar primero un convocado.", vbCritical, "Sin selección"
    End If
End Sub

'Inhabilitamos todo el formulario, limpiando su contenido previamente, excepto la lista desplegable de la
sesión del curso,
'que contendrá las sesiones del nuevo curso que hayamos seleccionado.
Private Sub curso_Click()
    Dim clave As String

    If Curso.ListIndex = -1 Then
```



```
matricula.Enabled = False
usuario.Enabled = False
nombre.Enabled = False
Sesion = ""
CMB_Buscar.Enabled = False
listaEspera.Enabled = False
lista.RowSource = "SELECT " AS Nombre, " AS Usuario, " AS [Nivel 3] from curso where Cod_Curso
= '-1';"
Sesion.RowSource = "SELECT [ClaveSesion] FROM Sesión WHERE ([CodigoCurso] = '9');"
lista.Enabled = False
Sesion = ""
Sesion.Enabled = False
vCurso = ""
ActualizaListas
Exit Sub
End If
clave = Curso.Column(1, Curso.ListIndex)
Sesion.Value = ""
'Cargamos todas las sesiones que pertenecen al curso seleccionado.
Sesion.RowSource = "SELECT [ClaveSesion] FROM Sesión WHERE ([CodigoCurso] = " & clave &
"";)"
matricula.Value = ""
usuario.Value = ""
nombre.Value = ""
lista.RowSource = "SELECT " AS Nombre, " AS Usuario, " AS [Nivel 3] from curso where Cod_Curso =
'-1';"
SUBF_Candidatos.RowSource = ""
Sesion.Enabled = True
matricula.Enabled = True
usuario.Enabled = True
nombre.Enabled = True
CMB_Buscar.Enabled = True
lista.Enabled = True
listaEspera.Enabled = True
ActualizaListas
End Sub

Private Sub Form_Close()

Dim Consulta As String
Dim BD As DAO.Database
Dim Registro As DAO.Recordset
Dim totalacuden As Integer

If LBL_Origen.Caption = "De Sesiones" Then
'Cargamos la lista de asistentes Confirmados a ese curso.
Set BD = CurrentDb
Consulta = "SELECT count(Us3270) as TotalAcuden FROM CONVOCADO WHERE ((Curso= " &
Curso.Column(1, Curso.ListIndex) & " ) AND (Sesion= " & Sesion.Value & " ) AND
(Estado='Confirmado'));"
Set Registro = BD.OpenRecordset(Consulta, dbOpenDynaset)
If Registro.RecordCount = 0 Then
Forms![Sesiones]!TXT_asistentesAcuden.Value = 0
Else
Forms![Sesiones]!TXT_asistentesAcuden.Value = Registro!totalacuden
End If
Registro.Close
```



```
Consulta = "SELECT count(Us3270) as TotalAcuden FROM CONVOCADO WHERE ((Curso= "" &  
Curso.Column(1, Curso.ListIndex) & "")) AND (Sesion= "" & Sesion.Value & "")) AND  
(Estado='Convocado');"
```

```
Set Registro = BD.OpenRecordset(Consulta, dbOpenDynaset)  
If Registro.RecordCount = 0 Then  
    Forms![Sesiones]!txtAsistentesConvocados.Value = 0  
Else  
    Forms![Sesiones]!txtAsistentesConvocados.Value = Registro!totalacuden  
End If  
Registro.Close
```

```
BD.Close  
End If
```

```
End Sub
```

'Inhabilitamos todo el formulario al abrir, excepto la lista desplegable curso, para que se cargue los datos en el formulario

'una vez que hayamos seleccionado un curso.

```
Private Sub Form_Open(Cancel As Integer)
```

```
    Sesion.Enabled = False  
    matricula.Enabled = False  
    usuario.Enabled = False  
    nombre.Enabled = False  
    lista.Enabled = False  
    listaEspera.Enabled = False  
    Me.SUBF_Candidatos.Enabled = False  
    Me.SUBF_Convocados.Enabled = False  
    suprimir.Enabled = False  
    CMB_Buscar.Enabled = False  
    confirmar.Enabled = False  
    cancelar.Enabled = False  
    movercandconv.Enabled = False  
    moverconvcand.Enabled = False  
    movertodoscandconv.Enabled = False  
    movertodosconvcand.Enabled = False  
    vNombre = ""  
    vUsuario = ""  
    ActualizaListas
```

```
End Sub
```

'Cargamos en los cuadros de texto usuario, matricula y apellidos y nombre, el empleado que hayamos seleccionado del cuadro

'de lista.

```
Private Sub lista_Click()
```

```
    Dim Consulta As String  
    Dim BD As DAO.Database  
    Dim Registro As DAO.Recordset
```

' Creamos el contenedor de la base de datos.

```
Set BD = CurrentDb
```

'Nos posicionamos en el registro del empleado seleccionado de la tabla "TB_PERSONAS".

```
Consulta = "SELECT TB_PERSONAS.NOMBRE, TB_PERSONAS.[MATRICULA HR],  
TB_PERSONAS.US3270 FROM TB_PERSONAS WHERE [US3270] = "" & lista.Value & """"
```

'Lo cargamos en un tipo de datos registro.

```
Set Registro = BD.OpenRecordset(Consulta, dbOpenDynaset)
```

```
If Registro.RecordCount = 1 Then
```

'Rellenamos los cuadros de texto usuario, matricula y apellidos y nombre.

```
nombre.Value = Registro.Fields(0).Value
```



```
matricula.Value = Registro.Fields(1).Value
usuario.Value = Registro.Fields(2).Value
End If
'Cerramos el contenedor del registro.
Registro.Close
End Sub

'Añadimos al empleado seleccionado a la lista de convocados de un curso.
Private Sub listaEspera_Click()
    Dim Consulta, LiteralCargo, UsuarioJefe As String
    Dim Registro As DAO.Recordset
    Dim Comprueba As DAO.Recordset
    Dim BD As DAO.Database
    Dim activado As Integer
    Dim SEXO As String

    'Comprobamos que el campo apellidos y nombre no este vacio.
    If (nombre.Value <> "") And (usuario.Value <> "") And (matricula.Value <> "") Then
        Set BD = CurrentDb

        ' Verificamos si ya ha sido inscrito a este curso
        Consulta = "SELECT * FROM Convocado INNER JOIN Sesión ON (Convocado.Curso =
Sesión.CodigoCurso) AND (Convocado.Curso = Sesión.CodigoCurso) AND (Convocado.Sesion =
Sesión.ClaveSesion) WHERE (Us3270="" & [usuario].[Value] & "" and Curso = "" & Curso.Column(1,
Curso.ListIndex) & "") ORDER BY Sesión.FechaInicio DESC;"
        Set Registro = CurrentDb.OpenRecordset(Consulta, dbOpenDynaset)
        If Registro.RecordCount <> 0 Then
            If MsgBox("El empleado " & Registro![Apellidos, Nombre] & " ya ha realizado este curso el " &
Format(Registro!FechaInicio, "dd/mm/yyyy") & " (" & Registro!Curso & "-" & Registro!Sesion & ")." &
vbCr & "¿Desea agregarlo a la lista de candidatos?", vbQuestion + vbYesNo, "Agregar a candidatos") =
vbNo Then
                Registro.Close
                Exit Sub
            End If
        End If
        Registro.Close

        'Nos posicionamos en el registro del empleado seleccionado en el cuadro lista, de la tabla
"TB_PERSONAS".
        Consulta = "SELECT * FROM TB_PERSONAS WHERE (([Nombre]= """" & [nombre].[Value] & """"))
AND ([Matricula HR]= """" & [matricula].[Value] & """")) AND ([Us3270]="" & [usuario].[Value] & ""));"

        'Lo cargamos en un tipo de datos registro.
        Set Registro = BD.OpenRecordset(Consulta, dbOpenDynaset)

        SEXO = Registro.Fields("SEXO").Value
        'Comprobamos que solo hay una coincidencia.
        If Registro.RecordCount = 1 Then

            'Obtenemos el cargo que tiene.
            LiteralCargo = Registro.Fields(12).Value
            activado = -1
            'Comprobamos que depende de algún jefe.
            If IsNull(Registro.Fields(15).Value) Then
                UsuarioJefe = ""
                activado = 0
            Else
                Consulta = "SELECT US3270 FROM TB_PERSONAS WHERE [MATRICULA HR] LIKE '*' &
Registro.Fields(14).Value & '*'"
            End If
        End If
    End Sub
```



```
Set Registro = BD.OpenRecordset(Consulta, dbOpenDynaset)
If Registro.RecordCount <> 0 Then UsuarioJefe = Registro.Fields(0).Value
End If
'Agregamos el empleado seleccionado a la tabla "CONVOCADO"
Consulta = "INSERT INTO Convocado (Curso, Sesion, [Apellidos, Nombre], [Matricula HR],
Us3270, Cancelaciones, Comentarios, Estado,[Literal Cargo],[Us3270 jefe],[Copia a Jefe], Enviar, SEXO) "
& _
"VALUES (" & Curso.Column(1, Curso.ListIndex) & ", '0000', " & nombre.Value & ", " &
matricula.Value & ", " & usuario.Value & ", " & 0 & ", 'Sin comentarios', 'Seleccionado', " & LiteralCargo
& ", " & UsuarioJefe & ", " & activado & ", " & -1 & ", " & SEXO & ");"
Call InsertarFila(convocados, Consulta)
End If
'Limpiamos los campos usuario, matricula, apellidos y nombre y lista para una posible nueva busqueda.
matricula.Value = ""
usuario.Value = ""
nombre.Value = ""
lista.RowSource = "SELECT " AS Nombre, " AS Usuario, " AS [Nivel 3] from curso where Cod_Curso
= '-1';"
Else
MsgBox "Por favor, seleccione una persona de la lista para poder añadirlo como candidato", 0 + 16,
"Falta candidato"
End If

End Sub

Private Sub matricula_KeyDown(KeyCode As Integer, Shift As Integer)
'Sólo ejecuta el código si se presiona la tecla ENTER
If KeyCode = 13 And matricula.Value <> "" Then
CMB_Buscar_Click
End If
End Sub

'Moveremos el candidato seleccionado a la lista de convocados.
Private Sub movercandconv_Click()
Dim FechaIni As Date
Dim FechaFin As Date

If Trim(Sesion.Value & " ") = "" Then Exit Sub

'Comprobamos que hay al menos 1 candidato en la lista y que alguno este seleccionado.
If vUsuario <> "" Then

'Obtener las fechas de inicio y fin del curso a donde se pretende apuntar al empleado
Consulta = "SELECT * FROM Sesión WHERE CodigoCurso=" & Curso.Column(1, Curso.ListIndex)
& " and ClaveSesion=" & Sesion.Value & ";"
Set Registro = CurrentDb.OpenRecordset(Consulta, dbOpenDynaset)
FechaIni = Registro!FechaInicio
FechaFin = Registro!FechaFin
Registro.Close

AsignaEmpleadoASesion FechaIni, FechaFin, vUsuario, Sesion.Value, Curso.Column(1,
Curso.ListIndex)

suprimir.Enabled = False
SUBF_Convocados.Enabled = True
SUBF_Convocados.SetFocus
ActualizaListas
End If
End Sub
```



```
'Moveremos el empleado seleccionado a la lista de candidatos.
Private Sub MoverConvCand_Click()
  'Comprobamos que hay al menos 1 candidato en la lista y que alguno este seleccionado.
  If vUsuario <> "" Then
    ' Asigna a una sesión y cambia el estado
    CurrentDb.Execute "update Convocado set Estado = 'Seleccionado', Sesion = '0000' WHERE Sesion = '"
& Sesion.Value & "' and Curso = '" & Curso.Column(1, Curso.ListIndex) & "' AND estado = 'Convocado'
and Us3270 = '" & vUsuario & "'"

    SUBF_Candidatos.Enabled = True
    SUBF_Candidatos.SetFocus
    confirmar.Enabled = False
    cancelar.Enabled = False
    ActualizaListas
  End If
End Sub

Private Sub nombre_KeyDown(KeyCode As Integer, Shift As Integer)
  ' Sólo ejecuta el código si se presiona la techa ENTER
  If KeyCode = 13 And nombre.Value <> "" Then
    CMB_Buscar_Click
  End If
End Sub

'Moveremos todos los empleados de la lista de candidatos a la lista de convocados.
Private Sub movertodoscandconv_Click()
  Dim FechaIni As Date
  Dim FechaFin As Date

  If Trim(Sesion.Value & " ") = "" Then Exit Sub

  'Comprobamos que hay al menos 1 candidato en la lista y que alguno este seleccionado.
  If Not SUBF_Candidatos.Form.Recordset.EOF Then
    Screen.MousePointer = 11

    ' Obtener las fecjas de inicio y fin del curso a donde se pretende apuntar al empleado
    Consulta = "SELECT * FROM Sesión WHERE CodigoCurso='" & Curso.Column(1, Curso.ListIndex)
& "' and ClaveSesion='" & Sesion.Value & "';"
    Set Registro = CurrentDb.OpenRecordset(Consulta, dbOpenDynaset)
    FechaIni = Registro!FechaInicio
    FechaFin = Registro!FechaFin
    Registro.Close

    ' Procesa cada elemento de la lista de candidatos
    SUBF_Candidatos.Form.Recordset.MoveFirst
    With SUBF_Candidatos.Form.Recordset
      Do While Not SUBF_Candidatos.Form.Recordset.EOF
        AsignaEmpleadoASesion FechaIni, FechaFin, !Us3270, Sesion.Value, Curso.Column(1,
Curso.ListIndex)
        .MoveNext
      Loop
    End With

    suprimir.Enabled = False
    SUBF_Convocados.Enabled = True
    SUBF_Convocados.SetFocus
    ActualizaListas
```



```
Screen.MousePointer = 1
End If
End Sub
```

'Moveremos todos los empleados de la lista de convocados a la lista de candidatos.

```
Private Sub movertodosconvcand_Click()
```

```
'Comprobamos que hay al menos 1 candidato en la lista y que alguno este seleccionado.
```

```
If vUsuario <> "" Then
```

```
' Asigna a una sesión y cambia el estado
```

```
CurrentDb.Execute "update Convocado set Estado = 'Seleccionado', Sesion = '0000' WHERE Sesion = "" & Sesion.Value & "" and Curso = "" & Curso.Column(1, Curso.ListIndex) & "" AND estado = 'Convocado'"
```

```
SUBF_Candidatos.Enabled = True
```

```
SUBF_Candidatos.SetFocus
```

```
confirmar.Enabled = False
```

```
cancelar.Enabled = False
```

```
ActualizaListas
```

```
End If
```

```
End Sub
```

'Cargamos todos los datos en el formulario correspondientes al curso y sesión seleccionado, habilitando para ello todos

los campos del formulario.

```
Private Sub sesion_Click()
```

```
matricula.Value = ""
```

```
usuario.Value = ""
```

```
nombre.Value = ""
```

```
lista.RowSource = "SELECT " AS Nombre, " AS Usuario, " AS [Nivel 3] from curso where Cod_Curso = '-1';"
```

```
SUBF_Candidatos.Form.RecordSource = "SELECT * FROM convocado WHERE ((curso='000000') And (sesion='9999')) ORDER BY Estado DESC , [Apellidos, Nombre]; "
```

```
matricula.Enabled = True
```

```
usuario.Enabled = True
```

```
nombre.Enabled = True
```

```
lista.Enabled = True
```

```
listaEspera.Enabled = True
```

```
CMB_Buscar.Enabled = True
```

```
ActualizaListas
```

```
End Sub
```

'Eliminaremos el candidato seleccionado de la lista de convocados.

```
Private Sub suprimir_Click()
```

```
Dim index As Integer
```

```
'Comprobaremos que exista un candidato seleccionado.
```

```
If Not SUBF_Candidatos.Form.Recordset.EOF Then
```

```
If MsgBox("¿Está seguro de eliminar al candidato: " & vNombre & "?", vbYesNo + vbQuestion, "Eliminar Candidato") = vbYes Then
```

```
'Borramos el convocado.
```

```
CurrentDb.Execute "delete * from Convocado WHERE Sesion='0000' and Curso = "" & Curso.Column(1, Curso.ListIndex) & "" AND estado in ('Seleccionado', 'Cancelado') and Us3270 = "" & vUsuario & """
```

```
End If
```

```
End If
```

```
SUBF_Candidatos.Requery
```

```
If SUBF_Candidatos.Form.Recordset.EOF Then
```

```
matricula.SetFocus
```

```
Else
```



```
        SUBF_Candidatos.SetFocus
    End If
    ActualizaListas
    suprimir.Enabled = False
End Sub
```

```
'Añadiremos una fila a un cuadro de lista.
Private Sub InsertarFila(ByVal convocados, ByVal Consulta As String)
    CurrentDb.Execute (Consulta)
    ActualizaListas
End Sub
```

```
'Procedimiento para salir del formulario cerrándolo y guardar el total de asistentes que acuden a ese curso.
Private Sub salir_Click()
```

```
    DoCmd.Close
```

```
End Sub
```

```
Private Sub usuario_KeyDown(KeyCode As Integer, Shift As Integer)
    ' Sólo ejecuta el código si se presiona la tecla ENTER
    If KeyCode = 13 And usuario.Value <> "" Then
        CMB_Buscar_Click
    End If
End Sub
```

Form BuscarCurso

```
Option Compare Database
Public gblCurso As String
```

```
Private Sub CBX_SeleccionCurso_Click()
    BuscarCursos
End Sub
```

```
Private Sub chkMostrarPlantillas_Click()
    If chkMostrarPlantillas.Value = True Then
        CBX_SeleccionCurso.Enabled = False
        cmbEscuela.Enabled = False
        dtFechaDesde.Enabled = False
        chkSubvencionados.Enabled = False
        txtFecha.ForeColor = 12632256
    Else
        CBX_SeleccionCurso.Enabled = True
        cmbEscuela.Enabled = True
        dtFechaDesde.Enabled = True
        dtFechaDesde.Left = dtFechaDesde.Left
        dtFechaDesde.Top = dtFechaDesde.Top
        chkSubvencionados.Enabled = True
        txtFecha.ForeColor = 0
    End If
    BuscarCursos
End Sub
```

```
Private Sub chkSubvencionados_Click()
    BuscarCursos
End Sub
```



```
Private Sub CMB_Aceptar_Click()
    Form_Sesiones.vCurso = gblCurso
    Form_Sesiones.vSesion = Forms![BuscarCurso]![SUBF_Sesiones].Form![Sesión]
    Form_Sesiones.SeleccionarCurso
    DoCmd.OpenForm ("Sesiones")
    CMB_Salir_Click
End Sub

Public Sub CMB_Salir_Click()
    ' Vacía la tabla temporal de sesiones
    ' CurrentDb.Execute "DELETE * FROM Temp_Sesion"

    ' SUBF_Sesiones.Requery

    DoCmd.Close
End Sub

Private Sub BuscarCursos()
    Dim SQL As String
    Dim Where As String

    CMB_Aceptar.Enabled = False

    If chkMostrarPlantillas.Value = True Then
        SQL = "SELECT CodigoCurso, ClaveSesion, FechaInicio, FechaFin, Lugar, Proveedor, NombreSesion,
        IdFormador FROM SesionesDeCursos WHERE "
        Where = " ClaveSesion = '0000' ORDER BY NombreSesion"
    Else
        SQL = "SELECT CodigoCurso, ClaveSesion, FechaInicio, FechaFin, Lugar, Proveedor, NombreSesion,
        IdFormador FROM SesionesDeCursos WHERE "
        'Where = " ClaveSesion <> '0000' AND FechaInicio >= #" & Format(dtFechaDesde, "yyyy/mm/dd") &
        "##"
        Where = " ClaveSesion <> '0000' AND FechaFin >= #" & Format(dtFechaDesde, "yyyy/mm/dd") & "##"
        If CBX_SeleccionCurso.Value <> "" Then
            If CBX_SeleccionCurso.ListIndex <> -1 Then
                Where = Where & " and CodigoCurso ='" & CBX_SeleccionCurso.Column(1,
                CBX_SeleccionCurso.ListIndex) & "'"
            Else
                Where = Where & " and UCase(NombreSesion) like '*' &
                Trim(UCase(CBX_SeleccionCurso.Value)) & '*"'
            End If
        End If
    End If

    If cmbEscuela.Value <> "" Then
        Where = Where & " and Escuela ='" & cmbEscuela & "'"
    End If

    If chkSubvencionados.Value = True Then
        Where = Where & " and Subvencionado = True"
    End If
    Where = Where & " ORDER BY FechaInicio"
End Sub

' Refresca la consulta
SUBF_Sesiones.Form.RecordSource = SQL & Where
SUBF_Sesiones.Requery
End Sub
```



```
Private Sub cmbEscuela_Click()  
    BuscarCursos  
End Sub
```

```
Private Sub dtFechaDesde_Exit(Cancel As Integer)  
    BuscarCursos  
End Sub
```

```
Private Sub Form_Load()  
    chkSubvencionados.Value = False  
End Sub
```

Form CalendarioIntranet

```
' Constantes de colores para los FlexGrid  
Const conNavy As Long = 8404992  
Const conRed As Long = 14013951  
Const conYellow As Long = 13434879  
Const conGreen As Long = 14548957
```

```
Private txtNombrePC As String
```

```
Sub GeneraHTML()  
    Dim dias As Integer  
    Dim Mes As String  
    Dim CambiaColor As Boolean
```

```
Screen.MousePointer = 11
```

```
If opTecnologia = True Then  
    Open "d:\8_Esc_Tecno_Planificacion_texto.html" For Output As #1  
Else  
    Open "d:\8_Esc_SC_Planificacion_texto.html" For Output As #1  
End If
```

```
Print #1, "<html>"  
Print #1, "<head>"  
Print #1, "<title>Portal Formacion - Escuela de Tecnologia - Calendario Formativo</title>"  
Print #1, "<META NAME='\"" Author '\" CONTENT='\"" Juan Carlos Urbina Camacho '\">"  
Print #1, "<script language='\"" JavaScript '\" type='\"" text/javascript '\"  
src='\"" /comun/IIRGDetecNavegadorGen.js '\"></script>"  
' *****  
' Poner esta línea sólo para hacer la prueba en local  
Print #1, "<link rel='stylesheet' href='IIRGEspacio2.css'>"  
' *****  
Print #1, "<script language='\"" JavaScript '\">"  
Print #1, "function abreVentana(URL)"  
Print #1, " {"  
Print #1, "  
window.open(URL, \"\"Ventana_Flotante\"\", \"\"width=919,height=699,top=0,left=0,menubar=no,scrollbars=no,t  
oolbar=no,status=no,location=no,resizable=no\"");"  
Print #1, " }"  
Print #1, "</script>"  
Print #1, "</head>"  
Print #1, ""  
Print #1, "<body>"
```



Print #1, "<p class="">Durante los próximos días se estarán realizando los cursos mostrados a continuación. Para más detalles hacer clic sobre el nombre del curso:</p>"

Print #1, "<table width="" border="" align="" cellpadding="" cellspacing="">"

Print #1, "<tr class="">"

Print #1, "<td colspan="">Fecha</td>"

Print #1, "<td rowspan="">Nombre de cursos</td>"

' Escribe el mes

Mes = flxCalendario.TextMatrix(0, 4)

dias = 1

For i = 4 To flxCalendario.Cols - 1

If Mes <> flxCalendario.TextMatrix(0, i) And flxCalendario.ColWidth(i) <> 0 Then ' And Trim(" " & flxCalendario.TextMatrix(0, i) <> ""

Print #1, "<td colspan="" & dias - 1 & "">" & Format(Mes, "mmmm") & "</td>"

Mes = flxCalendario.TextMatrix(0, i)

dias = 2

Else

If flxCalendario.ColWidth(i) <> 0 Then dias = dias + 1

End If

Next i

Print #1, "<td colspan="" & dias & "">" & Format(Mes, "mmmm") & "</td>"

Print #1, "</tr>"

Print #1, "<tr class="">"

Print #1, "<td width="">Ini.</td>"

Print #1, "<td width="">Fin.</td>"

' Escribe el días

For i = 4 To flxCalendario.Cols - 1

If flxCalendario.ColWidth(i) <> 0 Then Print #1, "<td width="">" & flxCalendario.TextMatrix(1, i) & "</td>"

Next i

Print #1, "</tr>"

CambiaColor = False

For Fila = 2 To flxCalendario.Rows - 1

CambiaColor = Not CambiaColor

If CambiaColor Then

Print #1, "<tr class="">"

Else

Print #1, "<tr>"

End If

' Ancho de las dos primeras columnas

Print #1, "<td class="" width="">" & Format(flxCalendario.TextMatrix(Fila, 1), "dd-mmm") & "</td>"

Print #1, "<td class="" width="">" & Format(flxCalendario.TextMatrix(Fila, 2), "dd-mmm") & "</td>"

If CambiaColor Then

Print #1, "<td class="">" & flxCalendario.TextMatrix(Fila, 3) & "</td>"



```
Else
  Print #1, " <td class=""TextoTabla8""><a href="" & flxCalendario.TextMatrix(Fila, 0) & """">" &
  flxCalendario.TextMatrix(Fila, 3) & "</a></td>"
End If

' Pone el color a la planificación dentro del calendario
For Columna = 4 To flxCalendario.Cols - 1
  If flxCalendario.ColWidth(Columna) <> 0 Then
    SeleccionaRango flxCalendario, Fila, Columna, Fila, Columna
    If Str(flxCalendario.CellBackColor) = Str(conRed) Then
      Print #1, " <td class=""FondoGris3"" width=""12"">&nbsp;</td>"
    Else
      If CambiaColor Then
        Print #1, " <td class=""FondoCrema2"" width=""12"">&nbsp;</td>"
      Else
        Print #1, " <td class=""TextoTabla8Cen"" width=""12"">&nbsp;</td>"
      End If
    End If
  End If
Next Columna
Print #1, " </tr>"
Next Fila

Print #1, " </table>"
Print #1, "<p>&nbsp;</p>"
Print #1, "</body>"
Print #1, "</html>"
Close #1

Screen.MousePointer = 1

If opTecnologia = True Then
  MsgBox "Se ha creado el fichero d:\8_Esc_Tecno_Planificacion_texto.html", vbInformation
Else
  MsgBox "Se ha creado el fichero d:\8_Esc_SC_Planificacion_texto.html", vbInformation
End If
End Sub

Sub Formatea()
  Dim rst As Recordset
  Dim strLink As String
  Dim sCurso As String
  Dim sSesion As String
  Dim RegPeriodos As DAO.Recordset

  Dim Fila As Integer
  Dim Columna As Integer
  Dim ColIni As Integer
  Dim ColFin As Integer
  Dim ColMes As Integer
  Dim UltimaFila As Integer
  Dim Colorear As Boolean
  Dim FechaIni As Date
  Dim FechaFin As Date
  Dim Fecha As Date
  Dim MesActual As String
  Dim PlanificacionET As Boolean

  ' Inicializa el FlexGrid
```



```
flxCalendarario.Rows = 2
```

```
flxCalendarario.Cols = 4
```

```
flxCalendarario.ColWidth(0) = 0 ' Para hiperlink
```

```
flxCalendarario.TextMatrix(0, 1) = "Fechas"
```

```
flxCalendarario.TextMatrix(0, 2) = "Fechas"
```

```
flxCalendarario.MergeRow(0) = True
```

```
flxCalendarario.TextMatrix(1, 1) = "Ini."
```

```
flxCalendarario.TextMatrix(1, 2) = "Fin"
```

```
flxCalendarario.ColWidth(3) = 3500
```

```
flxCalendarario.TextMatrix(0, 3) = "Nombre de cursos"
```

```
flxCalendarario.TextMatrix(1, 3) = "Nombre de cursos"
```

```
flxCalendarario.MergeCol(3) = False
```

```
' Formato de la cabecera
```

```
SeleccionaRango flxCalendarario, 0, 0, 1, 3
```

```
flxCalendarario.CellAlignment = 4
```

```
'then align Col 1 as Left-Centre
```

```
flxCalendarario.CellBackColor = conNavy
```

```
'Pinta el texto en blanco
```

```
flxCalendarario.CellForeColor = vbWhite
```

```
'show OrderID number in Bold
```

```
flxCalendarario.CellFontBold = True
```

```
If opTecnologia = True Then
```

```
Set rst = CurrentDb.OpenRecordset("SELECT * FROM " & txtNombrePC & " Where ((FechaFin) >= Now() And ((ClaveSesion) <> '9999' And (ClaveSesion) <> '0000') and IndCalendarario = true And Escuela = 'Escuela de Tecnología' ) ORDER BY FechaInicio, FechaFin;")
```

```
Else
```

```
Set rst = CurrentDb.OpenRecordset("SELECT * FROM " & txtNombrePC & " Where ((FechaFin) >= Now() And ((ClaveSesion) <> '9999' And (ClaveSesion) <> '0000') and IndCalendarario = true And Escuela <> 'Escuela de Tecnología') ORDER BY FechaInicio, FechaFin;")
```

```
End If
```

```
Do Until rst.EOF
```

```
strLink = "javascript:abreVentana('Convocatoria/" & rst!CodigoCurso & "-" & rst!ClaveSesion & "/Convocatoria.htm");"
```

```
flxCalendarario.AddItem strLink & vbTab & rst!FechaInicio & vbTab & rst!FechaFin & vbTab & rst!NombreSesion & vbTab
```

```
rst.MoveNext
```

```
Loop
```

```
rst.Close
```

```
' Obtiene rango de fechas para el calendario
```

```
If opTecnologia = True Then
```

```
Set rst = CurrentDb.OpenRecordset("SELECT MIN(FechaInicio) as MinIni, MAX(FechaFin) as MaxFin FROM " & txtNombrePC & " Where ((FechaFin) >= Now() And ((ClaveSesion) <> '9999' And (ClaveSesion) <> '0000') and IndCalendarario = true And Escuela = 'Escuela de Tecnología' );")
```

```
Else
```

```
Set rst = CurrentDb.OpenRecordset("SELECT MIN(FechaInicio) as MinIni, MAX(FechaFin) as MaxFin FROM " & txtNombrePC & " Where ((FechaFin) >= Now() And ((ClaveSesion) <> '9999' And (ClaveSesion) <> '0000') and IndCalendarario = true And Escuela <> 'Escuela de Tecnología' );")
```

```
End If
```

```
If rst.RecordCount > 0 Then
```

```
FechaIni = IIf(IsNull(rst!MinIni), Now, rst!MinIni)
```



```
FechaFin = IIf(IsNull(rst!MaxFin), Now, rst!MaxFin)
Else
  rst.Close
  Exit Sub
End If
rst.Close

' Coloca los nombres de los meses y los días
Fecha = FechaIni
MesActual = Format(Fecha, "mmmm")
Columna = 4
ColMes = 4
While Fecha <= FechaFin
  If Weekday(Fecha) <> vbSunday And Weekday(Fecha) <> vbSaturday Then
    flxCalendario.Cols = flxCalendario.Cols + 1
    flxCalendario.ColWidth(Columna) = 350
    ' Coloca el nombre del mes
    If MesActual <> Format(Fecha, "mmmm") Then
      MesActual = Format(Fecha, "mmmm")
    End If

    flxCalendario.TextMatrix(1, Columna) = Day(Fecha)
    flxCalendario.TextMatrix(0, Columna) = MesActual

    Columna = Columna + 1
  End If
  Fecha = Fecha + 1
Wend

flxCalendario.TextMatrix(0, Columna - 1) = MesActual

' Formato de la cabecera
SeleccionaRango flxCalendario, 0, 0, 1, Columna - 1

flxCalendario.CellAlignment = 4
flxCalendario.CellBackColor = conNavy
flxCalendario.CellForeColor = vbWhite
flxCalendario.CellFontBold = True

' then align Col 1 as Left-Centre
' Pinta el texto en blanco
' show OrderID number in Bold

' Pintar rangos de los cursos en azul
ColMes = 4
For Fila = 2 To flxCalendario.Rows - 1

  ' Verifica si tiene períodos especiales
  sCurso = Mid(flxCalendar.TextMatrix(Fila, 0), 38, 6)
  sSesion = Mid(flxCalendar.TextMatrix(Fila, 0), 45, 4)

  Consulta = "SELECT * FROM CursosPeriodos WHERE CodigoCurso ='" & sCurso & "' AND
ClaveSesion in ('" & sSesion & "') order by Secuencia"
  Set RegPeriodos = CurrentDb.OpenRecordset(Consulta, dbOpenDynaset)

  If RegPeriodos.RecordCount <> 0 Then
    RegPeriodos.MoveFirst
    While Not RegPeriodos.EOF
      ' Obtiene los valores de fechas a pintar
      FechaIni = RegPeriodos!FInicial
      FechaFin = RegPeriodos!FFinal
```



```
        If RegPeriodos!FFinal >= Now Then PintaDiasCurso FechaIni, FechaFin, Fila,
RegPeriodos!Espaciado
        RegPeriodos.MoveNext
    Wend
Else
    ' Obtiene los valores de fechas a pintar
    FechaIni = flxCalendario.TextMatrix(Fila, 1)
    FechaFin = flxCalendario.TextMatrix(Fila, 2)
    PintaDiasCurso FechaIni, FechaFin, Fila, False
End If

    RegPeriodos.Close

Next Fila

' Elimina las columnas vacias
For Columna = 4 To flxCalendario.Cols - 1
    For Fila = 2 To flxCalendario.Rows - 1
        Encontrado = False ' Utilizado para indicar que hay alguna celda con color
        SeleccionaRango flxCalendario, Fila, Columna, Fila, Columna
        If Str(flxCaleandario.CellBackColor) = Str(conRed) Or flxCaleandario.TextMatrix(Fila, Columna) =
"_" Then
            Encontrado = True
            Exit For
        End If
    Next Fila
    If Encontrado = False Then
        flxCaleandario.ColWidth(Columna) = 0
    End If
Next Columna

End Sub

Sub PintaDiasCurso(FechaIni As Date, FechaFin As Date, Fila As Integer, Separador As Boolean)
    Dim Columna As Integer
    Dim ColIni As Integer
    Dim ColFin As Integer
    Dim ColMes As Integer
    Dim MesActual As String

    Fecha = FechaIni
    ' buscar inicio del curso
    Encontrado = False
    Columna = 4
    ColMes = 4
    MesActual = flxCaleandario.TextMatrix(0, ColMes)

    ' busca le celda que coincide con el inicio del curso
    While Encontrado = False
        If Format(FechaIni, "mmmm") = MesActual And Int(flxCaleandario.TextMatrix(1, Columna)) =
Day(FechaIni) Then
            Encontrado = True
        Else
            If MesActual <> flxCaleandario.TextMatrix(0, Columna) And Trim(flxCaleandario.TextMatrix(0,
Columna) & " ") <> "" Then
                MesActual = flxCaleandario.TextMatrix(0, Columna)
                ColMes = Columna
            End If
        End If
    End While
End Sub
```



```
End If
' Verificar a ver si es primero de siguiente mes
If Format(FechaIni, "mddd") = MesActual And Int(flxCalendarario.TextMatrix(1, Columna)) =
Day(FechaIni) Then
    Encontrado = True
End If
End If
Columna = Columna + 1
If Columna >= flxCalendarario.Cols Then Encontrado = True
Wend
Columna = Columna - 1

' Colorea celdas
ColIni = Columna
For i = 1 To (DateDiff("d", FechaIni, FechaFin) + 1)
    If Weekday(Fecha) <> vbSunday And Weekday(Fecha) <> vbSaturday Then
        SeleccionaRango flxCalendarario, Fila, Columna, Fila, Columna

        If Separador Then
            flxCalendarario.TextMatrix(Fila, Columna) = "_"
            flxCalendarario.CellForeColor = vbWhite
        Else
            flxCalendarario.CellBackColor = conRed
        End If

        Columna = Columna + 1
    End If
    Fecha = Fecha + 1
Next i
Columna = Columna - 1
End Sub

Public Sub SeleccionaRango(ByVal vGrid As Control, ByVal vRow1 As Long, ByVal vCol1 As Long,
ByVal vRow2 As Long, ByVal vCol2 As Long)
    vGrid.Row = vRow1
    vGrid.Col = vCol1
    vGrid.RowSel = vRow2
    vGrid.ColSel = vCol2

End Sub

Private Sub cmdHTML_Click()
    GeneraHTML
End Sub

Private Sub cmdPrevisualizar_Click()
    Screen.MousePointer = 11
    TabCtl.Value = 1
    Formatea
    Screen.MousePointer = 1
End Sub

Private Sub flxCalendarario_DblClick()
    If flxCalendarario.MouseRow > 1 And flxCalendarario.MouseCol > 3 Then
        SeleccionaRango flxCalendarario, flxCalendarario.MouseRow, flxCalendarario.MouseCol,
flxCalendarario.MouseRow, flxCalendarario.MouseCol
        If Str(flxCalendarario.CellBackColor) = Str(conRed) Then
            flxCalendarario.CellBackColor = vbWhite
        Else

```



```
        flxCalendarario.CellBackColor = conRed
    End If
Else
    If flxCalendarario.MouseRow = 1 Then
        flxCalendarario.ColWidth(flxCalendarario.MouseCol) = 0 ' Para ocultarla
    End If
End If
End Sub

Private Sub Form_Load()
    Screen.MousePointer = 11
    ' Crea nombre único por PC para la tabla de calendario para evitar accesos simultáneos
    txtNombrePC = NombrePC & "Calendario"

    ' Crea la tabla temporal
    On Error Resume Next
    CurrentDb.Execute "SELECT FechaInicio, FechaFin, CodigoCurso, ClaveSesion, NombreSesion, Lugar,
Proveedor, IndCalendario, Escuela INTO " & txtNombrePC & " FROM SesionesDeCursos Where
((FechaFin) >= Now() And ((ClaveSesion) <> '9999' And (ClaveSesion) <> '0000')) ORDER BY
FechaInicio, FechaFin;"

    If Err.Number = 3010 Then
        CurrentDb.Execute "DELETE * FROM " & txtNombrePC
        CurrentDb.Execute "INSERT INTO " & txtNombrePC & " SELECT FechaInicio, FechaFin,
CodigoCurso, ClaveSesion, NombreSesion, Lugar, Proveedor, IndCalendario, Escuela FROM
SesionesDeCursos Where ((FechaFin) >= Now() And ((ClaveSesion) <> '9999' And (ClaveSesion) <>
'0000')) ORDER BY FechaInicio, FechaFin;"
    Else
        If Err.Number <> 0 Then GoTo MensajeError
    End If
    On Error GoTo 0

    DoEvents
    Me.SUBF_Datos.Form.RecordSource = "SELECT * FROM " & txtNombrePC
    Me.SUBF_Datos.Requery

    Screen.MousePointer = 1
    Exit Sub
MensajeError:
    MsgBox Err.Description, vbCritical, "Error"

End Sub

Private Sub salir_Click()
    Screen.MousePointer = 11
    On Error Resume Next
    Me.SUBF_Datos.Form.RecordSource = ""
    Me.SUBF_Datos.Requery
    CurrentDb.Execute "UPDATE Sesión INNER JOIN " & txtNombrePC & " ON (Sesión.ClaveSesion = " &
txtNombrePC & ".ClaveSesion) AND (Sesión.CodigoCurso = " & txtNombrePC & ".CodigoCurso) SET
Sesión.IndCalendario = [" & txtNombrePC & "].[IndCalendario];"
    CurrentDb.Execute "DROP TABLE " & txtNombrePC
    If Err.Number <> 0 Then MsgBox "No se ha podido eliminar la tabla temporal.", vbCritical, "Error"
    On Error GoTo 0

    Screen.MousePointer = 1
    DoCmd.Close
End Sub
```



Form Enviados F

'Procedimiento que adjunta un archivo que se encuentre en el sistema.

```
Private Sub adjuntar_Click()
```

```
' This requires a reference to the Microsoft Office 11.0 Object Library.
```

```
Dim fDialog As Office.FileDialog
```

```
Dim varFile As Variant
```

```
' Set up the File dialog box.
```

```
Set fDialog = Application.FileDialog(msoFileDialogFilePicker)
```

```
With fDialog
```

```
' Allow the user to make multiple selections in the dialog box.
```

```
.AllowMultiSelect = False
```

```
' Set the title of the dialog box.
```

```
.Title = "Selección de anexos"
```

```
.InitialFileName = Left$(CurrentDb.Name, Len(CurrentDb.Name) - InStr(StrReverse(CurrentDb.Name),
```

```
"\") + 1) & ".."
```

```
' Clear out the current filters, and then add your own.
```

```
.Filters.Clear
```

```
.Filters.Add "Todos los archivos", "*.*"
```

```
' Show the dialog box. If the .Show method returns True, the
```

```
' user picked at least one file. If the .Show method returns
```

```
' False, the user clicked Cancel.
```

```
If .Show = True Then
```

```
' Loop through each file that is selected and then add it to the list box.
```

```
For Each varFile In .SelectedItems
```

```
For i = 1 To anexos.ListCount
```

```
If UCase(anexos.ItemData(i - 1)) = UCase(varFile) Then Exit Sub
```

```
Next i
```

```
anexos.AddItem (varFile)
```

```
Next
```

```
cmdModificarPlantilla.Enabled = True
```

```
Else
```

```
If txtPlantilla = "" Then
```

```
cmdModificarPlantilla.Enabled = False
```

```
End If
```

```
End If
```

```
End With
```

```
End Sub
```

```
Private Sub buzon_Click()
```

```
If buzon.Value = True Then
```

```
buzones.Enabled = True
```

```
Else
```

```
buzones.Value = ""
```

```
buzones.Enabled = False
```

```
End If
```

```
End Sub
```

'Desplegable que nos muestra la lista de buzones disponibles de las diferentes escuelas.

```
Private Sub buzones_Change()
```

```
'Habilitamos el checkbox para que se envíe a ese buzón también.
```

```
buzon.Enabled = True
```

```
buzon.Value = 1
```



End Sub

'Inhabilitamos todo el formulario, limpiando su contenido previamente, excepto la lista desplegable de la sesión del curso,

'que contendrá las sesiones del nuevo curso que hayamos seleccionado.

Private Sub curso_Change()

buzones.Value = ""

plantilla.Value = ""

título.Value = ""

cuerpo.Value = ""

anexos.RowSource = ""

Sesion.Value = ""

Enviar.Enabled = False

buzon.Enabled = False

buzones.Enabled = False

plantilla.Enabled = False

título.Enabled = False

cuerpo.Enabled = False

anexos.Enabled = False

adjuntar.Enabled = False

eliminar.Enabled = False

FormConvocados.Enabled = False

Sesion.Enabled = True

Sesion.Value = ""

'Cargamos todas las sesiones que pertenecen al curso seleccionado.

Sesion.RowSource = "SELECT DISTINCT Convocado.Sesion FROM Convocado WHERE ((Convocado.curso=curso.value)); "

End Sub

Private Sub CHK_SeleccionarEnviar_Click()

Dim BD As DAO.Database

Dim Registro As DAO.Recordset

Dim Consulta As String

Set BD = CurrentDb

Consulta = "SELECT Convocado.[Matricula HR], Convocado.[Apellidos, Nombre], Convocado.[Literal Cargo], Convocado.[Copia a jefe], Convocado.Enviar FROM Convocado WHERE ((convocado.curso="" & TXT_Curso.Value & "") And (convocado.sesion="" & TXT_Sesion.Value & "") And (convocado.estado<>'Seleccionado'));"

Set Registro = BD.OpenRecordset(Consulta, dbOpenDynaset)

If Registro.RecordCount > 0 Then

Registro.MoveFirst

While Not Registro.EOF

Registro.Edit

Registro("[Enviar]") = CHK_SeleccionarEnviar.Value

Registro.Update

Registro.MoveNext

Wend

End If

Registro.Close

BD.Close

FormConvocados.Requery

End Sub

Private Sub CHK_SeleccionarJefe_Click()



```
Dim BD As DAO.Database
Dim Registro As DAO.Recordset
Dim Consulta As String
```

```
Set BD = CurrentDb
```

```
Consulta = "SELECT Convocado.[Matricula HR], Convocado.[Apellidos, Nombre], Convocado.[Literal Cargo], Convocado.[Copia a jefe], Convocado.Enviar FROM Convocado WHERE ((convocado.curso="" & TXT_Curso.Value & "") And (convocado.sesion="" & TXT_Sesion.Value & "") And (convocado.estado<>'Seleccionado'));"
```

```
Set Registro = BD.OpenRecordset(Consulta, dbOpenDynaset)
```

```
If Registro.RecordCount > 0 Then
```

```
    Registro.MoveFirst
```

```
    While Not Registro.EOF
```

```
        Registro.Edit
```

```
        Registro("[Copia a jefe]") = CHK_SeleccionarJefe.Value
```

```
        Registro.Update
```

```
        Registro.MoveNext
```

```
    Wend
```

```
End If
```

```
Registro.Close
```

```
BD.Close
```

```
FormConvocados.Requery
```

```
End Sub
```

```
Private Sub cmdModificarPlantilla_Click()
```

```
    ' Iniciar Outlook. ' Si ya se está ejecutando, utilizará la misma instancia...
```

```
    Dim olApp As Outlook.Application
```

```
    Dim Body As String
```

```
    If IsNull(txtPlantilla) = True Then Exit Sub
```

```
    DoEvents
```

```
    Set olApp = CreateObject("Outlook.Application")
```

```
    DoEvents
```

```
    'Inicio de sesión. No importa si ya lo está ejecutando y tiene iniciada una sesión...
```

```
    Dim olNs As Outlook.Namespace
```

```
    Set olNs = olApp.GetNamespace("MAPI")
```

```
    olNs.Logon
```

```
    ' Enviar un mensaje al nuevo contacto.
```

```
    Dim olMail As Outlook.MailItem
```

```
    Dim txtNombre As String
```

```
    ' Crea correo desde la Plantilla
```

```
    Set olMail = olApp.CreateItemFromTemplate(txtPlantilla)
```

```
    ' Viasaliza para modificar
```

```
    olMail.Display True
```

```
    ' Almacena la plantilla, utilizado para crear los primeros ficheros de plantillas cuando no había uno para copiar
```

```
    If MsgBox("¿Quieres guardar los cambios en la plantilla?", vbQuestion + vbYesNo, "Guardar Plantilla") = vbYes Then olMail.SaveAs txtPlantilla, olTemplate
```

```
    Set olMail = Nothing
```



```
olNs.Logoff

Set olNs = Nothing
Set olItem = Nothing
Set olApp = Nothing
End Sub

Private Sub cmdSeleccionaPlantilla_Click()
' This requires a reference to the Microsoft Office 11.0 Object Library.
Dim fDialog As Office.FileDialog
Dim varFile As Variant

' Set up the File dialog box.
Set fDialog = Application.FileDialog(msoFileDialogFilePicker)
With fDialog
' Allow the user to make multiple selections in the dialog box.
.AllowMultiSelect = False

' Set the title of the dialog box.
.Title = "Selección de plantilla de correo"
.InitialFileName = Left$(CurrentDb.Name, Len(CurrentDb.Name) - InStr(StrReverse(CurrentDb.Name),
"\") + 1) & "Plantillas Correo"

' Clear out the current filters, and then add your own.
.Filters.Clear
.Filters.Add "Plantillas Outlook", "*.oft"

' Show the dialog box. If the .Show method returns True, the
' user picked at least one file. If the .Show method returns
' False, the user clicked Cancel.
If .Show = True Then
' Loop through each file that is selected and then add it to the list box.
For Each varFile In .SelectedItems
txtPlantilla = varFile
Next
cmdModificarPlantilla.Enabled = True
Else
If txtPlantilla = "" Then
cmdModificarPlantilla.Enabled = False
End If
End If
End With

End Sub

'Procedimiento que elimina un archivo adjunto de la lista de anexos disponibles.
Private Sub eliminar_Click()
If (anexos.ListIndex <> -1) Then
anexos.RemoveItem (anexos.ListIndex)
End If
End Sub

'Envío del correo electrónico compuesto por el Título y el Cuerpo del mensaje, la lista de destinatarios
seleccionados,
'asi como la lista de los buzones si es que se envía desde ellos y por último la lista de los jefes si hemos
seleccionado
'que recibirán una copia.
Private Sub enviar_Click()
```



```
Dim Consulta As String
Dim BD As DAO.Database
Dim Registro As DAO.Recordset
Dim objOutlook As Outlook.Application
Dim objOutlookMsg As Outlook.MailItem
Dim objOutlookRecip As Outlook.Recipient
Dim objOutlookAttach As Outlook.Attachment
Dim Direccion As String
Dim i As Integer

Dim txtClientesSinCorreo As String
Dim rstSolicitudes As Recordset
Dim rstSesion As Recordset

' Comprobar que se ha seleccionado una plantilla
If Trim(" " & txtPlantilla) = "" Then
    MsgBox "Debes seleccionar primero una plantilla de correo.", vbCritical, "Error"
    txtPlantilla.SetFocus
    Exit Sub
End If

' Iniciar Outlook. ' Si ya se está ejecutando, utilizará la misma instancia...
Dim olApp As Outlook.Application
Dim Body As String

DoEvents

Set olApp = CreateObject("Outlook.Application")

DoEvents

' Inicio de sesión. No importa si ya lo está ejecutando y tiene iniciada una sesión...
Dim olNs As Outlook.Namespace
Set olNs = olApp.GetNamespace("MAPI")
olNs.Logon

' Enviar un mensaje al nuevo contacto.
Dim olMail As Outlook.MailItem
Dim txtNombre As String

' Selecciona los distintos asistentes para buscarles sus respectivos itinerarios
Set rstSolicitudes = CurrentDb.OpenRecordset("SELECT * FROM CONVOCADO WHERE ((CURSO=" & TXT_Curso & ") AND ([CONVOCADO].[SESION]= " & TXT_Sesion.Value & ") AND ([CONVOCADO].[ESTADO] in ('Confirmado', 'Convocado')) AND (Enviar = True));", , adOpenKeyset)

' Selecciona la sesión para obtener los datos
Set rstSesion = CurrentDb.OpenRecordset("SELECT * FROM SesionesDeCursos WHERE ((CodigoCurso=" & TXT_Curso & ") AND (ClaveSesion=" & TXT_Sesion.Value & "));", , adOpenKeyset)

Screen.MousePointer = vbHourglass

With rstSolicitudes
    If Not (.BOF And .EOF) Then
        .MoveLast
        .MoveFirst
        DoCmd.OpenForm "ProgresoEnvioDeCorreo"
        Forms("ProgresoEnvioDeCorreo").ProgressBar.Min = 0
    End If
End With
```



```
Forms("ProgresoEnvioDeCorreo").ProgressBar.Max = .RecordCount
DoEvents
If .RecordCount > 0 Then
  Do While Not .EOF
    Forms("ProgresoEnvioDeCorreo").ProgressBar =
Forms("ProgresoEnvioDeCorreo").ProgressBar + 1
    DoEvents
    *****
    'olApp.CreateItemFromTemplate
    If !Enviar = True Then
      Set olMail = olApp.CreateItemFromTemplate(txtPlantilla)
      olMail.To = Trim(" " & !Us3270)
      'Añadimos los jefes que hemos seleccionado que reciban una copia.
      If (![Copia a Jefe] = True) Then
        olMail.CC = Trim(" " & ![Us3270 Jefe])
      End If

      'Añadimos el campo "de"
      If (buzones.Value <> "") And (buzon.Value) Then
        olMail.SentOnBehalfOfName = buzones.Column(1, buzones.ListIndex)
      End If

      'Añadimos la lista de archivos adjuntos al mensaje.
      If anexos.ListCount > 0 Then
        For i = 1 To anexos.ListCount
          olMail.Attachments.Add anexos.ItemData(i - 1)
        Next i
      End If

      ' Comprobamos el nombre de las direcciones de correo electrónico.
      For Each objOutlookRecip In olMail.Recipients
        objOutlookRecip.Resolve
        If Not objOutlookRecip.Resolve Then
          olMail.Display True
        End If
      Next

      'Importancia del mensaje
      olMail.Importance = olImportanceHigh

      txtNombre = ![Apellidos, Nombre]
      txtNombre = Right(txtNombre, Len(txtNombre) - InStr(1, txtNombre, ","))

      ' Título del mensaje
      olMail.Subject = "Convocatoria curso "" & txtTitulo & """"

      ' Cuerpo del mensaje
      If !SEXO = "M" Then
        olMail.HTMLBody = Replace(olMail.HTMLBody, "[ASISTENTE]", "Estimado " &
txtNombre & ", " & vbCr & vbCr & vbCr & vbCr)
      Else
        olMail.HTMLBody = Replace(olMail.HTMLBody, "[ASISTENTE]", "Estimada " &
txtNombre & ", " & vbCr & vbCr & vbCr & vbCr)
      End If
    End While
  End If
End Sub
```



```
End If

olMail.HTMLBody = Replace(olMail.HTMLBody, "[NOMBRE_CURSO]", txtTitulo)

If rstSesion!FechaInicio <> rstSesion!FechaFin Then
    olMail.HTMLBody = Replace(olMail.HTMLBody, "[FECHA]", "Del " &
Day(rstSesion!FechaInicio) & " de " & Format(rstSesion!FechaInicio, "mmm") & " al " &
Day(rstSesion!FechaFin) & " de " & Format(rstSesion!FechaFin, "mmm"))
Else
    olMail.HTMLBody = Replace(olMail.HTMLBody, "[FECHA]", ("El " &
Day(rstSesion!FechaInicio) & " de " & Format(rstSesion!FechaInicio, "mmm"))
End If

olMail.HTMLBody = Replace(olMail.HTMLBody, "[CENTRO_DE_FORMACION]",
rstSesion!Lugar)
olMail.HTMLBody = Replace(olMail.HTMLBody, "[FECHA_CONFIRMACION]", Trim(" "
& txtFechaConfirmacion))
olMail.HTMLBody = Replace(olMail.HTMLBody, "[HORARIO]", Trim(" " &
txtHorarioCurso))

'Enviamos el mensaje.
olMail.Save

Set olMail = Nothing
*****
'Creamos el log con la fecha y la hora del envío
Consulta = "INSERT INTO Envio (Curso, Sesion, Fecha, Hora) VALUES (" & TXT_Curso
& ", " & TXT_Sesion & ", " & Date & ", " & Time & ");"
CurrentDb.Execute (Consulta)

End If
.MoveNext
Loop
End If
DoCmd.Close
End If
End With

olNs.Logoff
Set olNs = Nothing
Set olItem = Nothing
Set olApp = Nothing

rstSolicitudes.Close
rstSesion.Close
CurrentDb.Close
Screen.MousePointer = 0

End Sub

'Una vez seleccionada la sesión del curso, habilitaremos el resto del formulario.
Private Sub sesion_Change()
    buzones.Value = ""
    plantilla.Value = ""
    titulo.Value = ""
    cuerpo.Value = ""
```



```
anexos.RowSource = ""
Enviar.Enabled = True
buzones.Enabled = True
plantilla.Enabled = True
titulo.Enabled = True
cuerpo.Enabled = True
anexos.Enabled = True
adjuntar.Enabled = True
eliminar.Enabled = True
FormConvocados.Enabled = True
'Caregaremos los datos de los empleados convocados a la celebración de esa sesión en concreto del curso.
FormConvocados.Form.RecordSource = "SELECT Convocado.[Matricula HR], Convocado.[Apellidos,
Nombre], Convocado.[Literal Cargo], Convocado.[Copia a jefe], Convocado.[Enviar] FROM Convocado
WHERE ((convocado.curso=curso.value) And (convocado.sesion=sesion.value) And
(Convocado.Estado='Seleccionado')) order by Convocado.[Apellidos, Nombre];"
End Sub
```

'Funcion que devuelve solo el nombre completo del campo que contiene nombre + apellidos.

```
Private Function ConseguirNombre(ByVal nombre As String) As String
```

```
    Dim coma As Integer
```

```
    Dim aux As String
```

'Posición donde se encuentra el separador de nombre y apellidos.

```
    coma = InStr(1, nombre, ",")
```

'Copiamos el nombre del empleado.

```
    nombre = Right(nombre, Len(nombre) - coma)
```

'Ponemos su primer carácter a mayuscula.

```
    ConseguirNombre = StrConv(nombre, vbProperCase)
```

```
End Function
```

```
Private Sub Form_Load()
```

```
    txtHorarioCurso.Value = "9:00 a 14:00 y de 15:00 a 18:00"
```

```
End Sub
```

```
Private Sub salir_Click()
```

```
    DoCmd.Close
```

```
End Sub
```

Form Convocados

'Procedimiento que activa o desactiva la copia del envío al jefe de un empleado, dependiendo de como estuviese

'su estado en la tabla de "CONVOCADOS".

```
Private Sub Copia_a_Jefe_Click()
```

```
    Dim Consulta As String
```

```
    Dim BD As DAO.Database
```

```
    Dim Registro As DAO.Recordset
```

'Seleccionamos los futuros asistentes al curso.

```
Set BD = CurrentDb
```

```
Consulta = "SELECT * FROM CONVOCADO WHERE (([CONVOCADO].[CURSO]= "" &
Form_Enviados_F.TXT_Curso.Value & """) AND ([CONVOCADO].[SESION]= "" &
Form_Enviados_F.TXT_Sesion.Value & """) AND ([CONVOCADO].[ESTADO]='Confirmado') AND
([CONVOCADO].[MATRICULA HR]= "" & [matricula].Value & "");"
```

```
Set Registro = BD.OpenRecordset(Consulta, dbOpenDynaset)
```

'Comprobamos que existe al menos un empleado en el curso

```
If Registro.RecordCount > 0 Then
```



'Si el campo copia a jefe esta vacío, no lo activaremos en el formulario, en caso contrario se dejara activado.

```
If (Registro.Fields("Us3270 Jefe") = "") Then
    [Copia a Jefe].Value = 0
End If
End If
Registro.Close
End Sub
```

Form ListaDeAlarmas

Option Compare Database

```
Dim mExpand As String
Const conNavy As Long = 8404992
Const conRed As Long = 14013951
Const conYellow As Long = 13434879
Const conGreen As Long = 14548957
Dim mMouseX As Long
Public txtNombrePC As String
```

Option Explicit

```
Private Sub btnExpandAll_Click()
    Dim rst As Recordset
```

```
    If cmbEscuela.ListIndex = -1 Then Exit Sub
```

```
    mExpand = ""
```

```
    ' Set rst = CurrentDb.OpenRecordset("SELECT DISTINCT CodigoCurso, ClaveSesion, NombreSesion,
FechaInicio, FechaFin FROM ConsultaAlarmasCursos WHERE Escuela = ""
```

```
    If cmbEscuela = "Todo" Then ' Muestra todas las alarmas
```

```
    ' Set rst = CurrentDb.OpenRecordset("SELECT DISTINCT CodigoCurso, ClaveSesion, NombreSesion,
FechaInicio, FechaFin FROM (SELECT " & txtNombrePC & ".Texto, " & txtNombrePC & ".TipoAlarma,
Sesión.* FROM " & txtNombrePC & " INNER JOIN Sesión ON (" & txtNombrePC &
".CodigoCurso=Sesión.CodigoCurso) AND (" & txtNombrePC & ".ClaveSesion=Sesión.ClaveSesion))
ORDER BY FechaInicio")
```

```
    Set rst = CurrentDb.OpenRecordset("SELECT DISTINCT CodigoCurso, ClaveSesion, NombreSesion,
FechaInicio, FechaFin FROM (SELECT " & txtNombrePC & ".Texto, " & txtNombrePC & ".TipoAlarma,
Sesión.* FROM " & txtNombrePC & " INNER JOIN Sesión ON (" & txtNombrePC &
".CodigoCurso=Sesión.CodigoCurso) AND (" & txtNombrePC & ".ClaveSesion=Sesión.ClaveSesion))
ORDER BY FechaInicio")
```

```
    Else
```

```
    Set rst = CurrentDb.OpenRecordset("SELECT DISTINCT CodigoCurso, ClaveSesion, NombreSesion,
FechaInicio, FechaFin FROM (SELECT " & txtNombrePC & ".Texto, " & txtNombrePC & ".TipoAlarma,
Sesión.* FROM " & txtNombrePC & " INNER JOIN Sesión ON (" & txtNombrePC &
".CodigoCurso=Sesión.CodigoCurso) AND (" & txtNombrePC & ".ClaveSesion=Sesión.ClaveSesion))
WHERE Escuela = "" _
```

```
    & cmbEscuela & "" ORDER BY FechaInicio")
```

```
    End If
```

```
    Do Until rst.EOF
```

```
        mExpand = mExpand & "." & rst!CodigoCurso & "-" & rst!ClaveSesion & "." 'add this Project
number to list as .123.
```

```
        rst.MoveNext
```

```
    Loop
```

```
    rst.Close
```

```
    RellenaGrid
```

```
    're-display grid with all projects expanded
```



End Sub

```
Private Sub chkSubvencionados_AfterUpdate()
```

```
    If cmbEscuela.ListIndex <> -1 Then
```

```
        RellenaGrid
```

```
    Else
```

```
        MsgBox "Debe seleccionar la escuela primero.", vbCritical, "Error en selección"
```

```
        chkSubvencionados = False
```

```
        cmbEscuela.SetFocus
```

```
    End If
```

```
End Sub
```

```
Private Sub cmbEscuela_AfterUpdate()
```

```
    RellenaGrid
```

```
End Sub
```

```
Public Sub RellenaGrid()
```

```
'Fill flex grid with data from qryFlexGrid
```

```
'Entry (vBeginDate) holds date for first column of dates
```

```
Dim rst As Recordset
```

```
Dim vCount As Long
```

```
Dim vPath As String
```

```
Dim vDate As Date
```

```
Dim mRowCount As Integer
```

```
Dim strSQL As String
```

```
    On Error GoTo ErrorCode
```

```
    vPath = Left$(CurrentDb.Name, Len(CurrentDb.Name) - InStr(StrReverse(CurrentDb.Name), "\") + 1) &  
"Line Images\"
```

```
'calc 'Line Images' folder location
```

```
    vPath = "g:\Alerta\planificacion\Line Images\"
```

```
'calc 'Line Images'  
folder location
```

```
    Screen.MousePointer = 11
```

```
    flxAlertas.Redraw = False  
operation)
```

```
'hide drawing action (speeds up
```

```
    flxAlertas.ColWidth(-1) = 300  
Twips)
```

```
'first set width of all cols (300
```

```
    flxAlertas.ColWidth(0) = 700
```

```
'set width of col 0 (1000 Twips)
```

```
    flxAlertas.ColWidth(1) = 1000
```

```
'set width of col 0 (1000 Twips)
```

```
    flxAlertas.ColWidth(2) = 1000
```

```
'set width of col 1 (3000 Twips)
```

```
    flxAlertas.ColWidth(3) = 8000
```

```
'set width of col 2 (1200 Twips)
```

```
    flxAlertas.ColWidth(4) = 1200
```

```
'set width of col 3 (1200 Twips)
```

```
    flxAlertas.ColWidth(5) = 1200
```

```
'set width of col 4 (1200 Twips)
```

```
    flxAlertas.Rows = 1  
rows)
```

```
'empty FlexGrid first (except for fixed
```

```
    flxAlertas.TextMatrix(0, 0) = "ID"
```

```
'insert sub-headings
```

```
    flxAlertas.TextMatrix(0, 1) = "Curso"
```

```
    flxAlertas.TextMatrix(0, 2) = "Sesión"
```

```
    flxAlertas.TextMatrix(0, 3) = "Nombre Sesión"
```

```
    flxAlertas.TextMatrix(0, 4) = "F. Inicio"
```



```
flxAlertas.TextMatrix(0, 5) = "F. Fin"
```

```
SeleccionaRango flxAlertas, 0, 1, 0, 5  
flxAlertas.CellBackColor = conNavy  
flxAlertas.CellForeColor = vbWhite  
flxAlertas.CellFontBold = True
```

```
'Pinta la fila en azul marino
```

```
'Pinta el texto en blanco  
'show OrderID number in Bold
```

```
strSQL = "SELECT CodigoCurso, ClaveSesion, NombreSesion, FechaInicio, FechaFin FROM (SELECT  
" & txtNombrePC & ".Texto, " & txtNombrePC & ".TipoAlarma, Sesión.* FROM " & txtNombrePC & "  
INNER JOIN Sesión ON (" & txtNombrePC & ".CodigoCurso=Sesión.CodigoCurso) AND (" &  
txtNombrePC & ".ClaveSesion=Sesión.ClaveSesion)) WHERE 1=1"
```

```
' Si se ha seleccionado alguna Escuela  
If cmbEscuela <> "Todo" Then ' Muestra todas las alarmas  
    strSQL = strSQL & " and Escuela = '" & cmbEscuela & "'" & ""  
End If
```

```
' Mostrar sólo los subvencionados  
If chkSubvencionados.Value = True Then  
    strSQL = strSQL & " and Subvencionado = True"  
End If
```

```
If cmbTipoAlerta.ListIndex <> -1 Then  
    If Trim(cmbTipoAlerta.Value & " ") <> "" Then  
        strSQL = strSQL & " and Texto = '" & Trim(cmbTipoAlerta) & "'" & ""  
    End If  
End If
```

```
' Selección de orden  
If cmbOrden = "Fecha" Then  
    strSQL = strSQL & " GROUP BY CodigoCurso, ClaveSesion, NombreSesion, FechaInicio, FechaFin  
ORDER BY FechaInicio"  
Else  
    strSQL = strSQL & " GROUP BY CodigoCurso, ClaveSesion, NombreSesion, FechaInicio, FechaFin  
ORDER BY MAX(TipoAlarma) DESC, FechaInicio"  
End If
```

```
Set rst = CurrentDb.OpenRecordset(strSQL)
```

```
mRowCount = 1  
Do Until rst.EOF  
    'set initial (non-fixed) row counter  
    'step thru recordset
```

```
    flxAlertas.AddItem mRowCount & vbTab _  
    & rst!CodigoCurso & vbTab _  
    & rst!ClaveSesion & vbTab _  
    & rst!NombreSesion & vbTab _  
    & rst!FechaInicio & vbTab _  
    & rst!FechaFin 'copy Orders fields to Cols 0 and 8
```

```
    SeleccionaRango flxAlertas, flxAlertas.Rows - 1, 1, flxAlertas.Rows - 1, 5  
'show Project rows in Bold (Cols 1-4 only)  
    flxAlertas.CellFontBold = True
```

```
    flxAlertas.CellBackColor = ColorAlarma(rst!CodigoCurso, rst!ClaveSesion)
```

```
    SeleccionaRango flxAlertas, flxAlertas.Rows - 1, 1, flxAlertas.Rows - 1, 1  
'align col 1 as LeftCenter
```



```
flxAlertas.CellAlignment = flexAlignRightCenter

mRowCount = mRowCount + 1                                'inc Row counter

flxAlertas.Col = 1                                       'set Col No for +/-x sign
If InStr(1, mExpand, "." & rst!CodigoCurso & "-" & rst!ClaveSesion & ".") > 0 Then
'if this .ProjectID. is in expand string then
    Set flxAlertas.CellPicture = LoadPicture(vPath & "Minus Sign.bmp")    'copy - image to
Project rows (Col 1)
    MuestraTareas flxAlertas, rst!CodigoCurso, rst!ClaveSesion
Else
    flxAlertas.Row = flxAlertas.Rows - 1
    Set flxAlertas.CellPicture = LoadPicture(vPath & "Plus Sign.bmp")    'copy + image to
Project rows (Col 1)
End If
rst.MoveNext                                            'move to next Order record
Loop
rst.Close
Set rst = Nothing

'now do all this after rows have been added to grid

If flxAlertas.Rows > 1 Then                            'align col 1 as LeftCenter
    SeleccionaRango flxAlertas, 1, 3, flxAlertas.Rows - 1, 3
Else
    SeleccionaRango flxAlertas, 0, 3, 0, 3
End If

flxAlertas.CellAlignment = flexAlignLeftCenter

flxAlertas.Redraw = True                                're-display grid
Screen.MousePointer = 1

Exit Sub

ErrorCode:
flxAlertas.Redraw = True                                're-display grid if error
Screen.MousePointer = 1
If Not rst Is Nothing Then rst.Close
Set rst = Nothing
Beep
MsgBox Err.Description                                  'and display error

End Sub

Public Sub MuestraTareas(flxgrd As Control, vCurso As String, vSesion As String)
    Dim rst As Recordset
    Dim strSQL As String

    If cmbTipoAlerta.ListIndex <> -1 Then
        If Trim(cmbTipoAlerta.Value & " ") <> "" Then
            strSQL = "SELECT * FROM (SELECT " & txtNombrePC & ".Texto, " & txtNombrePC &
".TipoAlarma, Sesión.* FROM " & txtNombrePC & " INNER JOIN Sesión ON (" & txtNombrePC &
".CodigoCurso=Sesión.CodigoCurso) AND (" & txtNombrePC & ".ClaveSesion=Sesión.ClaveSesion))
WHERE CodigoCurso = " & vCurso & " and ClaveSesion = " & vSesion & " and Texto = " &
Trim(cmbTipoAlerta) & " ORDER BY TipoAlarma DESC"
        Else
```



```
strSQL = "SELECT * FROM (SELECT " & txtNombrePC & ".Texto, " & txtNombrePC &
".TipoAlarma, Sesión.* FROM " & txtNombrePC & " INNER JOIN Sesión ON (" & txtNombrePC &
".CodigoCurso=Sesión.CodigoCurso) AND (" & txtNombrePC & ".ClaveSesion=Sesión.ClaveSesion))
WHERE CodigoCurso = " & vCurso & " and ClaveSesion = " & vSesion & " ORDER BY TipoAlarma
DESC"
End If
Else
strSQL = "SELECT * FROM (SELECT " & txtNombrePC & ".Texto, " & txtNombrePC &
".TipoAlarma, Sesión.* FROM " & txtNombrePC & " INNER JOIN Sesión ON (" & txtNombrePC &
".CodigoCurso=Sesión.CodigoCurso) AND (" & txtNombrePC & ".ClaveSesion=Sesión.ClaveSesion))
WHERE CodigoCurso = " & vCurso & " and ClaveSesion = " & vSesion & " ORDER BY TipoAlarma
DESC"
End If

Set rst = CurrentDb.OpenRecordset(strSQL)

Do Until rst.EOF
flxgrd.AddItem "" & vbTab & "" & vbTab & rst!Texto & vbTab & rst!Texto & vbTab & rst!Texto &
vbTab & rst!Texto

flxgrd.MergeRow(flxgrd.Rows - 1) = True 'merge Row 0
flxgrd.MergeCol(2) = True 'and Column 5
flxgrd.MergeCol(3) = True 'and Column 5
flxgrd.MergeCol(4) = True 'and Column 5
flxgrd.MergeCol(5) = True 'and Column 5

SeleccionaRango flxAlertas, flxAlertas.Rows - 1, 2, flxAlertas.Rows - 1, 2

flxgrd.CellAlignment = flexAlignLeftCenter 'then align Col 1 as Left-
Centre

Select Case rst!TipoAlarma
Case 0
flxgrd.CellBackColor = conGreen 'colour cell yellow
Case 1
flxgrd.CellBackColor = conYellow 'colour cell Red
Case 2
flxgrd.CellBackColor = conRed 'colour cell Red
End Select

rst.MoveNext
Loop
rst.Close
End Sub

Public Function ColorAlarma(vCurso As String, vSesion As String) As Long
Dim rst As Recordset

Set rst = CurrentDb.OpenRecordset("SELECT max(TipoAlarma) as MaxAlarma FROM " &
txtNombrePC & " WHERE CodigoCurso = " & vCurso & " and ClaveSesion = " & vSesion & """)

Select Case rst!MaxAlarma
Case 0
ColorAlarma = conGreen 'colour cell yellow
Case 1
ColorAlarma = conYellow 'colour cell Red
```



```
Case 2
    ColorAlarma = conRed                'colour cell Red
End Select

rst.Close
End Function

Public Sub AgregaCabecera()

' Crea la cabecera de las alarmas

    flxAlertas.AddItem ("Curso" & vbTab _
    & "Sesión" & vbTab _
    & "Nombre Sesión" & vbTab _
    & "F. Inicio" & vbTab _
    & "Alarma")
    SeleccionaRango flxAlertas, flxAlertas.Rows - 1, 0, flxAlertas.Rows - 1, 4          'Pinta la fila en
    azul marino
    flxAlertas.CellBackColor = conNavy
    flxAlertas.CellForeColor = vbWhite                'Pinta el texto en blanco
    flxAlertas.CellFontBold = True                    'show OrderID number in Bold
End Sub

Public Sub SeleccionaRango(ByVal vGrid As Control, ByVal vRow1 As Long, ByVal vCol1 As Long,
ByVal vRow2 As Long, ByVal vCol2 As Long)

' Set up cell range
' Entry (vGrid) = name of FlexGrid control
' (vRow1) = first row No
' (vCol1) = first column No
' (vRow2) = last row No
' (vCol2) = last column No

    vGrid.Row = vRow1
    vGrid.Col = vCol1
    vGrid.RowSel = vRow2
    vGrid.ColSel = vCol2

End Sub

Private Sub cmbOrden_AfterUpdate()
    If cmbEscuela.ListIndex <> -1 Then
        RellenaGrid
    Else
        MsgBox "Debe seleccionar la escuela primero.", vbCritical, "Error en selección"
        cmbOrden = "Fecha"
        cmbEscuela.SetFocus
    End If
End Sub

Private Sub cmbTipoAlerta_AfterUpdate()
    If cmbEscuela.ListIndex <> -1 Then
        RellenaGrid
    Else
        MsgBox "Debe seleccionar la escuela primero.", vbCritical, "Error en selección"
        cmbTipoAlerta = ""
    End If
End Sub
```



```
cmbEscuela.SetFocus
End If
End Sub
```

```
Private Sub flxAlertas_Click()
'User clicks on flex grid
```

```
Dim vPosn As Long
Dim vProjectID As String
```

```
' FlexGridClick flxAlertas, vRow, vCol 'return relevant data about selected cell

If flxAlertas.Col <> 1 Then Exit Sub
If mMouseX > 735 And mMouseX < 900 Then 'Sólo entra si se hace clic en el simbolo de
+ o -
    vProjectID = "." & flxAlertas.TextMatrix(flxAlertas.Row, 1) & "-" &
flxAlertas.TextMatrix(flxAlertas.Row, 2) & "." 'fetch Project ID number and enclose in . chrs
    If Trim(flxAlertas.TextMatrix(flxAlertas.Row, 0) & " ") <> "" Then 'if user clicked
on a Task row (..) then take no action
        vPosn = InStr(1, mExpand, vProjectID) 'search Expand string for .ProjectID.
        If vPosn = 0 Then 'if Project NOT already expanded then
            mExpand = mExpand & vProjectID 'add .ProjectID. to Expand string (Project now
expanded)
        Else
            mExpand = Replace(mExpand, vProjectID, "") 'remove .ProjectID. from Expand string
(Project now contracted)
        End If
        RellenaGrid
    End If
End If
End Sub
```

```
Private Sub flxAlertas_DbIClick()
If Trim(flxAlertas.TextMatrix(flxAlertas.Row, 0) & " ") <> "" And
Trim(flxAlertas.TextMatrix(flxAlertas.Row, 0) & " ") <> "ID" Then
    DoCmd.OpenForm ("Sesiones")
    Form_Sesiones.vCurso = flxAlertas.TextMatrix(flxAlertas.Row, 1)
    Form_Sesiones.vSesion = flxAlertas.TextMatrix(flxAlertas.Row, 2)
    Form_Sesiones.SeleccionarCurso
    Form_Sesiones.vActualizaAlertas = True
    Form_Sesiones.CMB_Buscar.Enabled = False
    Form_Sesiones.CMB_Eliminar.Enabled = False
    Form_Sesiones.cmdImprimir.Enabled = False
    Form_Sesiones.cmdNuevaSesion.Enabled = False
End If
End Sub
```

```
Private Sub flxAlertas_MouseDown(ByVal Button As Integer, ByVal Shift As Integer, ByVal x As Long,
ByVal y As Long)
    mMouseX = x
End Sub
```



Private Sub Form_Open(Cancel As Integer)

```
Screen.MousePointer = 11
flxAlertas.ColWidth(-1) = 300           'first set width of all cols (300
Twips)

flxAlertas.ColWidth(0) = 700           'set width of col 0 (1000 Twips)
flxAlertas.ColWidth(1) = 1000        'set width of col 0 (1000 Twips)
flxAlertas.ColWidth(2) = 1000        'set width of col 1 (3000 Twips)
flxAlertas.ColWidth(3) = 8000        'set width of col 2 (1200 Twips)
flxAlertas.ColWidth(4) = 1200        'set width of col 3 (1200 Twips)
flxAlertas.ColWidth(5) = 1200        'set width of col 4 (1200 Twips)

flxAlertas.Rows = 1                   'empty FlexGrid first (except for fixed
rows)

flxAlertas.TextMatrix(0, 0) = "ID"     'insert sub-headings
flxAlertas.TextMatrix(0, 1) = "Curso"
flxAlertas.TextMatrix(0, 2) = "Sesión"
flxAlertas.TextMatrix(0, 3) = "Nombre Sesión"
flxAlertas.TextMatrix(0, 4) = "F. Inicio"
flxAlertas.TextMatrix(0, 5) = "F. Fin"

SeleccionaRango flxAlertas, 0, 1, 0, 5 'Pinta la fila en azul marino
flxAlertas.CellBackColor = conNavy
flxAlertas.CellForeColor = vbWhite    'Pinta el texto en blanco
flxAlertas.CellFontBold = True        'show OrderID number in Bold

flxAlertas.ColAlignment(-1) = flexAlignCenterCenter 'align all Cols as
Centre-Centre first

' Crea nombre único por PC para la tabla de alarman para evitar accesos simultáneos
txtNombrePC = NombrePC & "Alarmas"

' Crea la tabla temporal
On Error Resume Next
CurrentDb.Execute "CREATE TABLE " & txtNombrePC & "(CodigoCurso VARCHAR(6),ClaveSesion
VARCHAR(4),Texto VARCHAR(150), TipoAlarma integer, PRIMARY KEY (CodigoCurso, ClaveSesion,
Texto));"
If Err.Number = 3010 Then
    CurrentDb.Execute "DELETE * FROM " & txtNombrePC
Else
    If Err.Number <> 0 Then GoTo MensajeError
End If
On Error GoTo 0

' Genera alarmas sobre la tabla temporal
GeneraAlarmas txtNombrePC, "*", "*"

' Consulta listado de escuelas sobre la selección
cmbEscuela.RowSource = "SELECT DISTINCT Escuela FROM " & txtNombrePC & " INNER JOIN
Sesión ON (" & txtNombrePC & ".ClaveSesion = Sesión.ClaveSesion) AND (" & txtNombrePC &
".CodigoCurso = Sesión.CodigoCurso) union Select Distinct 'Todo' FROM " & txtNombrePC & ";"

' Consulta listado de escuelas sobre la selección
```



```
cmbTipoAlerta.RowSource = "SELECT DISTINCT Texto FROM " & txtNombrePC & " where Texto in  
(Select Texto from AlarmasDefinidas) union Select Distinct ' ' as Texto FROM " & txtNombrePC & " order  
by Texto;"
```

```
' Blanquea el contenido de los combos para que no muestren #Error  
cmbEscuela = ""  
cmbTipoAlerta = ""
```

```
' Inicializado de opciones de selección  
cmbOrden.Value = "Fecha"  
chkSubvencionados.Value = False
```

```
Screen.MousePointer = 1  
Exit Sub
```

MensajeError:

```
MsgBox Err.Description, vbCritical, "Error"  
End Sub
```

```
Private Sub salir_Click()  
cmbEscuela.RowSource = ""  
cmbTipoAlerta.RowSource = ""  
On Error Resume Next  
CurrentDb.Execute "DROP TABLE " & txtNombrePC  
If Err.Number <> 0 Then MsgBox "No se ha podido eliminar la tabla temporal de alarmas.", vbCritical,  
"Error"  
On Error GoTo 0  
DoCmd.Close  
End Sub
```

Form Menu principal

Option Compare Database

```
Private Sub CMB_Alertas_Click()  
DoCmd.OpenForm ("ListaDeAlarmas")  
End Sub
```

```
Private Sub CMB_Asistentes_Click()  
  
DoCmd.OpenForm ("Asistentes")  
Forms![Asistentes]!LBL_Origen.Caption = "De Menú Principal"  
Forms![Asistentes]!Curso.Locked = False  
Forms![Asistentes]!Curso.Locked = False  
Forms![Asistentes]!Sesion.Locked = False  
Forms![Asistentes]!Curso.Enabled = True  
Forms![Asistentes]!Curso.SetFocus  
End Sub
```

```
Private Sub CMB_Salir_Click()  
  
DoCmd.Quit
```

```
End Sub
```

```
Private Sub CMB_Sesiones_Click()  
  
DoCmd.OpenForm ("Sesiones")
```



End Sub

```
Private Sub cmdCalendarioIntranet_Click()  
    DoCmd.OpenForm ("CalendarioIntranet")  
End Sub
```

```
Private Sub Form_Load()
```

```
    On Error Resume Next
```

```
    DoCmd.Close acForm, "Asistentes"
```

```
End Sub
```

Form Planificación

Option Compare Database

```
Private Sub Aceptar_Click()
```

```
    Dim Consulta As String
```

```
    Dim BD As DAO.Database
```

```
    Dim Registro As DAO.Recordset
```

```
    ' Creamos el contenedor de la base de datos.
```

```
    Set BD = CurrentDb
```

```
    ' Nos posicionamos en el registro del empleado seleccionado de la tabla "TB_PERSONAS".
```

```
    Consulta = "SELECT Sesión.CodigoCurso, Sesión.ClaveSesion, Sesión.FechaInicio, Sesión.FechaFin,  
Sesión.NombreCurso, Sesión.EscuelaTecnologia, Sesión.ParticipantesGrabados, Sesión.AulaInterna,  
Sesión.CentroAulaInterna, Sesión.AulaProveedor, Sesión.DireccionAulaProveedor, Sesión.FormadorInterno,  
Sesión.NombreFormadorInterno, Sesión.FormadorExterno, Sesión.NombreFormadorExterno,  
Sesión.FechaEnvioConvocatoria, Sesión.AulaReservada, Sesión.RestauranteReservado,  
Sesión.TransporteReservado, Sesión.MaterialReservado, Sesión.Forcem, Sesión.ListadoFirmas,  
Sesión.Diplomas, Sesión.Cuestionarios, Sesión.CuestionariosEnviados, Sesión.ControlGestion,  
Sesión.ValoracionCurso FROM Sesión;"
```

```
    ' Lo cargamos en un tipo de datos registro.
```

```
    Set Registro = BD.OpenRecordset(Consulta, dbOpenDynaset)
```

```
    If Registro.RecordCount > 0 Then
```

```
        Registro.MoveFirst
```

```
        ' Insertamos en la tabla temporal.
```

```
        Consulta = "INSERT INTO Temp_Planificacion (Curso, Sesion, [Apellidos, Nombre], [Matricula HR],  
Us3270, Cancelaciones, Comentarios, Estado,[Literal Cargo],[Us3270 jefe],[Copia a Jefe],Enviar) VALUES  
('" & Curso.Value & "'," & Sesion.Value & "'," & nombre.Value & "'," & matricula.Value & "'," &  
usuario.Value & "', " & 0 & " , 'Sin comentarios' , 'Seleccionado'," & LiteralCargo & "'," & UsuarioJefe &  
"', " & activado & " , " & -1 & " );"
```

```
        CurrentDb.Execute
```

```
        nombre.Value = Registro.Fields(0).Value
```

```
        matricula.Value = Registro.Fields(1).Value
```

```
        usuario.Value = Registro.Fields(2).Value
```

```
    End If
```

```
    ' Cerramos el contenedor del registro.
```

```
    Registro.Close
```

```
End Sub
```



Form Sesiones

' Declaración de variables cuya utilización se repite a lo largo del programa.

```
Public BD As DAO.Database
Public Registro As DAO.Recordset
Public Consulta As String
Public vCurso As String
Public vSesion As String
Public vActualizaAlertas As Boolean
```

```
Public Sub SeleccionarCurso()
```

```
Dim BD As DAO.Database
Dim Registro As DAO.Recordset
Dim RegSesion As DAO.Recordset
Dim RegTareas As DAO.Recordset
Dim RegAsistentes As DAO.Recordset
Dim Consulta As String
```

```
Set BD = CurrentDb
```

```
Me.[TXT_ClaveCurso] = vCurso & vSesion
```

```
Consulta = "SELECT * FROM DetalleSesion WHERE (CodigoCurso="" & vCurso & "" AND  
ClaveSesion="" & vSesion & "")"
```

```
Set RegSesion = BD.OpenRecordset(Consulta, dbOpenDynaset)  
RegSesion.MoveFirst
```

```
Consulta = "SELECT * FROM CONVOCADO WHERE ((Curso= "" & vCurso & "") AND (Sesion= "" &  
vSesion & "") AND ([CONVOCADO].[Estado]='Confirmado'));"
```

```
Set RegAsistentes = BD.OpenRecordset(Consulta, dbOpenDynaset)  
Me.TXT_asistentesAcuden.Enabled = True  
If RegAsistentes.RecordCount > 0 Then  
RegAsistentes.MoveLast  
Me.TXT_asistentesAcuden.Value = RegAsistentes.RecordCount  
cmdImprimir.Enabled = True  
Else  
Me.TXT_asistentesAcuden.Value = 0  
cmdImprimir.Enabled = False  
End If  
RegAsistentes.Close
```

```
Consulta = "SELECT * FROM CONVOCADO WHERE ((Curso= "" & vCurso & "") AND (Sesion= "" &  
vSesion & "") AND ([CONVOCADO].[Estado]='Convocado'));"
```

```
Set RegAsistentes = BD.OpenRecordset(Consulta, dbOpenDynaset)  
Me.TXT_asistentesAcuden.Enabled = True  
If RegAsistentes.RecordCount > 0 Then  
RegAsistentes.MoveLast  
Me.txtAsistentesConvocados.Value = RegAsistentes.RecordCount  
Else  
Me.txtAsistentesConvocados.Value = 0#  
End If  
RegAsistentes.Close
```

```
Me.SUBF_OtrasTareasSesion.Enabled = True  
Me.SUBF_FPeriodos.Enabled = True
```



```
Me.CMB_asistentes.Enabled = True
Me.CMB_enviarConvocatoria.Enabled = True
Me.CBX_NombreCurso.Value = RegSesion("Nombre_Curso")
Me.TXT_NombreSesion = RegSesion("NombreSesion")
Me.dtFechaInicio.Value = RegSesion("FechaInicio")
Me.dtFechaFin.Value = RegSesion("FechaFin")
Me.CBX_Area.Value = RegSesion("NombreArea")
Me.CBX_Departamento.Value = RegSesion("NombreDepartamento")
Me.CBX_responsable.Value = RegSesion("NombreResp")
Me.txtDetHorario = RegSesion("DetHorario")
Me.TXT_Horas.Value = RegSesion("Horas")
Me.TXT_asistentesMin.Value = RegSesion("AsistentesMin")
Me.TXT_asistentesMax.Value = RegSesion("AsistentesMax")
Me.cmbEscuela.Value = RegSesion("Escuela")
Me.CHK_ParticipantesGrabados.Value = RegSesion("ParticipantesGrabados")
Me.CHK_ConvocatoriaEnviada.Value = RegSesion("ConvocatoriaEnviada")
If RegSesion("ConvocatoriaEnviada") = True Then
    Me.LBL_FechaEnvioConvocatoria.ForeColor = 0
    Me.dtFechaEnvioConvocatoria.Enabled = True
    Me.dtFechaEnvioConvocatoria.Top = Me.dtFechaEnvioConvocatoria.Top
    Me.dtFechaEnvioConvocatoria.Left = Me.dtFechaEnvioConvocatoria.Left
    Me.TabCtl.Value = 0
    Me.dtFechaEnvioConvocatoria.Value = RegSesion("FechaEnvioConvocatoria")
    Me.CMD_VerConvocatoria.Enabled = True
Else
    Me.LBL_FechaEnvioConvocatoria.ForeColor = 12632256
    Me.dtFechaEnvioConvocatoria.Enabled = False
    Me.CMD_VerConvocatoria.Enabled = False
End If
If RegSesion("VinculoConvocatoria") <> "" Then
    Me.LBL_verConvocatoria.Caption = IIf(Trim(RegSesion("VinculoConvocatoria") & " ") = "", "", "Haz clic aquí para ver la convocatoria.") 'RegSesion("VinculoConvocatoria")
    Me.LBL_verConvocatoria.Hyperlink.Address = RegSesion("VinculoConvocatoria")
    Me.LBL_verConvocatoria.ControlTipText = RegSesion("VinculoConvocatoria")
Else
    Me.LBL_verConvocatoria.Caption = ""
    Me.LBL_verConvocatoria.Hyperlink.Address = ""
    Me.LBL_verConvocatoria.ControlTipText = ""
End If
If RegSesion("MaterialRequerido") = False Then
    Me.CHK_MaterialNoRequerido.Value = True
    Me.CHK_MaterialRequerido.Value = False
    Me.CHK_materialReservado.Enabled = False
    Me.LBL_ImporteMaterial.ForeColor = 12632256
    Me.TXT_importeMaterial.Enabled = False
    Me.CMD_verFacturaMaterial.Enabled = False
    Me.LBL_verFacturaMaterial.Caption = ""
    Me.LBL_verFacturaMaterial.Hyperlink.Address = ""
    Me.LBL_verFacturaMaterial.ControlTipText = ""
Else
    Me.CHK_MaterialRequerido.Value = True
    Me.CHK_MaterialNoRequerido.Value = False
    Me.CHK_materialReservado.Enabled = True
    Me.CHK_materialReservado.Value = RegSesion("MaterialReservado")
    Me.LBL_ImporteMaterial.ForeColor = 0
    Me.TXT_importeMaterial.Enabled = True
    Me.TXT_importeMaterial.Value = RegSesion("ImporteMaterial")
    Me.CMD_verFacturaMaterial.Enabled = True
```



```
Me.LBL_verFacturaMaterial.Caption = Iif(Trim(RegSesion("VinculoFacturaMaterial") & " ") = "", "",
"Haz clic para ver la factura.") 'RegSesion("VinculoFacturaMaterial")
Me.LBL_verFacturaMaterial.Visible = True
Me.LBL_verFacturaMaterial.Hyperlink.Address = Trim(RegSesion("VinculoFacturaMaterial") & " ")
Me.LBL_verFacturaMaterial.ControlTipText = Trim(RegSesion("VinculoFacturaMaterial") & " ")
End If
Me.CHK_formadorCerrado.Value = RegSesion("FormadorCerrado")
If RegSesion("FormadorInterno") = True Then
Me.OPB_Interno.Value = True
Me.LBL_nombreFormadorInterno.ForeColor = 0
Me.TXT_nombreFormadorInterno.Enabled = True
Me.TXT_nombreFormadorInterno.Value = RegSesion("NombreFormadorInterno")
Else
Me.OPB_Interno.Value = False
Me.LBL_nombreFormadorInterno.ForeColor = 12632256
Me.TXT_nombreFormadorInterno.Enabled = False
Me.TXT_nombreFormadorInterno.Value = ""
End If
If RegSesion("FormadorExterno") = True Then
Me.cmdModificaProveedor.Enabled = True
Me.OPB_Externo.Value = True
Me.LBL_nombreFormadorExterno.ForeColor = 0
Me.CBX_nombreFormadorExterno.Enabled = True
Me.CBX_nombreFormadorExterno.Value = RegSesion("IdFormador")
Left(RegSesion("nom_formador") & Space(90), 90) & RegSesion("IdFormador")
Me.LBL_CIFformadorExterno.ForeColor = 0
Me.TXT_CIFformadorExterno.Enabled = True
Me.TXT_CIFformadorExterno.Value = RegSesion("CIFFormador")
Me.chkPagado.Value = RegSesion("Pagado")
Me.chkPagado.Enabled = True
Me.LBL_importeFormador.ForeColor = 0
Me.TXT_importeFormadorExterno.Enabled = True
Me.TXT_importeFormadorExterno.Value = RegSesion("ImporteFormador")
Me.CMD_verFacturaFormador.Enabled = True
If Trim(RegSesion("VinculoFacturaFormador") & " ") <> "" Then
Me.LBL_verFacturaFormador.Caption = Iif(Trim(RegSesion("VinculoFacturaFormador") & " ") = "",
"", "Haz clic aquí para ver la factura.") 'RegSesion("VinculoFacturaFormador")
Me.LBL_verFacturaFormador.Visible = True
Me.LBL_verFacturaFormador.ControlTipText = RegSesion("VinculoFacturaFormador")
Me.LBL_verFacturaFormador.Hyperlink.Address = RegSesion("VinculoFacturaFormador")
Else
Me.LBL_verFacturaFormador.Caption = ""
Me.LBL_verFacturaFormador.ControlTipText = ""
Me.LBL_verFacturaFormador.Hyperlink.Address = ""
End If
Else
Me.cmdModificaProveedor.Enabled = False
Me.OPB_Externo.Value = False
Me.LBL_nombreFormadorExterno.ForeColor = 12632256
Me.CBX_nombreFormadorExterno.Enabled = False
Me.CBX_nombreFormadorExterno.Value = ""
Me.LBL_CIFformadorExterno.ForeColor = 12632256
Me.TXT_CIFformadorExterno.Enabled = False
Me.TXT_CIFformadorExterno.Value = ""
Me.LBL_importeFormador.ForeColor = 12632256
Me.TXT_importeFormadorExterno.Enabled = False
Me.TXT_importeFormadorExterno.Value = ""
Me.CMD_verFacturaFormador.Enabled = False
Me.LBL_verFacturaFormador.Caption = ""
```



```
Me.LBL_verFacturaFormador.ControlTipText = ""
Me.LBL_verFacturaFormador.Hyperlink.Address = ""
End If
Me.CHK_aulaReservada.Value = RegSesion("AulaReservada")
If RegSesion("AulaReservada") = True Then
Me.LBL_FechaReservaAulaFormador.ForeColor = 0
Me.dtFechaReservaAula.Enabled = True
Me.TabCtl.Value = 0
Me.dtFechaReservaAula.Left = Me.dtFechaReservaAula.Left
Me.dtFechaReservaAula.Top = Me.dtFechaReservaAula.Top
Me.dtFechaReservaAula.Value = IIf(IsNull(RegSesion("FechaReservaAula")), Now(),
RegSesion("FechaReservaAula"))
If IsNull(RegSesion("FechaReservaAula")) Then Me.CHK_aulaReservada.Value = False
Else
Me.LBL_FechaReservaAulaFormador.ForeColor = 12632256
Me.dtFechaReservaAula.Enabled = False
End If
If RegSesion("AulaInterna") = True Then
Me.OPB_AulaInterna.Value = True
Me.LBL_centroAulaInterna.ForeColor = 0
Me.CBX_centroAulaInterna.Enabled = True
Me.CBX_centroAulaInterna.Value = RegSesion!CentroAulaInterna Left(RegSesion!Centro &
Space(90), 90) & RegSesion!CentroAulaInterna
Me.LBL_NombreAula.ForeColor = 0
Me.CBX_AulaInterna.Enabled = True
Me.CBX_AulaInterna.RowSource = "SELECT Aula FROM Aula_Interna WHERE IdCentro = " &
CBX_centroAulaInterna.Value ' Right(CBX_centroAulaInterna.Value, Len(CBX_centroAulaInterna.Value)
- 90)
Me.CBX_AulaInterna.Requery
Me.CBX_AulaInterna.Value = RegSesion("NombreAulaInterna")
Else
Me.OPB_AulaInterna.Value = False
Me.LBL_centroAulaInterna.ForeColor = 12632256
Me.CBX_centroAulaInterna.Enabled = False
Me.CBX_centroAulaInterna.Value = ""
Me.LBL_NombreAula.ForeColor = 12632256
Me.CBX_AulaInterna.Enabled = False
Me.CBX_AulaInterna.Value = ""
End If
If RegSesion("AulaProveedor") = True Then
Me.OPB_AulaProveedor.Value = True
Me.LBL_direccionAulaFormador.ForeColor = 0
Me.TXT_DireccionAulaFormador.Enabled = True
Me.TXT_DireccionAulaFormador.Value = RegSesion("DireccionAulaProveedor")
Me.OPB_CosteAula.Enabled = True
Me.OPB_CosteAula.Value = RegSesion("CosteProveedor")
Me.LBL_ImporteAulaExterna.ForeColor = 0
Me.TXT_importeAulaExterna.Enabled = True
Me.TXT_importeAulaExterna.Value = RegSesion("ImporteProveedor")
Me.CMD_verFacturaAula.Enabled = True
Me.LBL_verFacturaAula.Caption = IIf(Trim(RegSesion("VinculoFacturaProveedor") & " ") = "", "",
"Haz clic para ver la factura.") 'RegSesion("VinculoFacturaProveedor")
Me.LBL_verFacturaAula.Visible = True
Me.CMD_verFacturaAula.Visible = True
Me.LBL_verFacturaAula.ControlTipText = RegSesion("VinculoFacturaProveedor")
Me.LBL_verFacturaAula.Hyperlink.Address = RegSesion("VinculoFacturaProveedor")
Else
Me.LBL_verFacturaAula.Visible = False
Me.OPB_AulaProveedor.Value = False
```



```
Me.LBL_direccionAulaFormador.ForeColor = 12632256
Me.TXT_DireccionAulaFormador.Enabled = False
Me.TXT_DireccionAulaFormador.Value = ""
Me.OPB_CosteAula.Enabled = False
Me.OPB_CosteAula.Value = False
Me.LBL_ImporteAulaExterna.ForeColor = 12632256
Me.TXT_importeAulaExterna.Enabled = False
Me.TXT_importeAulaExterna.Value = ""
Me.CMD_verFacturaAula.Enabled = False
Me.LBL_verFacturaAula.Caption = ""
Me.LBL_verFacturaAula.ControlTipText = ""
Me.LBL_verFacturaAula.Hyperlink.Address = ""
End If
If RegSesion("RestauranteRequerido") = True Then
Me.OPB_RestauranteRequerido.Value = True
Me.OPB_RestauranteNoRequerido.Value = False
Me.CHK_restauranteReservado.Enabled = True
Me.CHK_restauranteReservado.Value = RegSesion("RestauranteReservado")
Me.LBL_nombreRestaurante.ForeColor = 0
Me.TXT_Restaurante.Enabled = True
Me.TXT_Restaurante.Value = RegSesion("NombreRestaurante")
Me.LBL_importeRestaurante.ForeColor = 0
Me.TXT_importeRestaurante.Enabled = True
Me.TXT_importeRestaurante.Value = RegSesion("ImporteRestaurante")
Me.CMD_verFacturaRestaurante.Enabled = True
Me.CMD_verFacturaRestaurante.Visible = True
Me.LBL_verFacturaRestaurante.Caption = IIf(Trim(RegSesion("VinculoFacturaRestaurante") & " ") =
"", "", "Haz clic para ver la factura.") 'RegSesion("VinculoFacturaRestaurante")
Me.LBL_verFacturaRestaurante.ControlTipText = RegSesion("VinculoFacturaRestaurante")
Me.LBL_verFacturaRestaurante.Hyperlink.Address = RegSesion("VinculoFacturaRestaurante")
Else
Me.OPB_RestauranteNoRequerido.Value = True
Me.OPB_RestauranteRequerido.Value = False
Me.CHK_restauranteReservado.Enabled = False
Me.CHK_restauranteReservado.Value = False
Me.LBL_nombreRestaurante.ForeColor = 12632256
Me.TXT_Restaurante.Enabled = False
Me.TXT_Restaurante.Value = ""
Me.LBL_importeRestaurante.ForeColor = 12632256
Me.TXT_importeRestaurante.Enabled = False
Me.TXT_importeRestaurante.Value = ""
Me.CMD_verFacturaRestaurante.Enabled = False
Me.LBL_verFacturaRestaurante.Caption = ""
Me.LBL_verFacturaRestaurante.ControlTipText = ""
Me.LBL_verFacturaRestaurante.Hyperlink.Address = ""
End If
If RegSesion("TransporteRequerido") = True Then
Me.OPB_TransporteRequerido.Value = True
Me.OPB_TransporteNoRequerido.Value = False
Me.CHK_transporteReservado.Enabled = True
Me.CHK_transporteReservado.Value = RegSesion("TransporteReservado")
Me.LBL_ImporteTransporte.ForeColor = 0
Me.TXT_importeTransporte.Enabled = True
Me.TXT_importeTransporte.Value = Trim(RegSesion("ImporteTransporte") & " ")
Me.CMD_verFacturaTransporte.Enabled = True
Me.LBL_verFacturaTransporte.Caption = IIf(Trim(RegSesion("VinculoFacturaTransporte") & " ") = "",
"", "Haz clic para ver la factura.") 'RegSesion("VinculoFacturaTransporte")
Me.LBL_verFacturaTransporte.ControlTipText = Trim(RegSesion("VinculoFacturaTransporte") & " ")
```



```
Me.LBL_verFacturaTransporte.Hyperlink.Address = Trim(RegSesion("VinculoFacturaTransporte") & "
")
Else
Me.OPB_TransporteRequerido.Value = False
Me.OPB_TransporteNoRequerido.Value = True
Me.CHK_transporteReservado.Enabled = False
Me.CHK_transporteReservado.Value = False
Me.LBL_ImporteTransporte.ForeColor = 12632256
Me.TXT_importeTransporte.Enabled = False
Me.TXT_importeTransporte.Value = ""
Me.CMD_verFacturaTransporte.Enabled = False
Me.LBL_verFacturaTransporte.Caption = ""
Me.LBL_verFacturaTransporte.ControlTipText = ""
Me.LBL_verFacturaTransporte.Hyperlink.Address = ""
End If
If RegSesion("Subvencionado") = False Then
Me.OPB_NoSubvencionado.Value = True
Me.OPB_FORCEM.Value = False
Me.OPB_Programa.Value = False
Me.CHK_ControlDeGestion.Enabled = False
Me.CHK_ControlDeGestion.Value = False
Me.LBL_FechaEnvioCG.ForeColor = 12632256
Me.dtFechaEnvioCG.Enabled = False
End If
If RegSesion("Forcem") = True Then
Me.OPB_NoSubvencionado.Value = False
Me.OPB_FORCEM.Value = True
Me.OPB_Programa.Value = False
Me.CHK_ControlDeGestion.Enabled = True
Me.CHK_ControlDeGestion.Value = RegSesion("ControlGestion")
Me.dtFechaEnvioCG.Enabled = True
Me.TabCtl.Value = 1
Me.dtFechaEnvioCG.Value = Format(IIf(IsNull(RegSesion("FechaEnvioControlGestion")), Now,
RegSesion("FechaEnvioControlGestion")), "yyyy/mm/dd")
If RegSesion("ControlGestion") = True Then
Me.LBL_FechaEnvioCG.ForeColor = 0
Me.dtFechaEnvioCG.Enabled = True
Else
Me.LBL_FechaEnvioCG.ForeColor = 12632256
Me.dtFechaEnvioCG.Enabled = False
End If
Me.dtFechaEnvioCG.Left = Me.dtFechaEnvioCG.Left
Me.dtFechaEnvioCG.Top = Me.dtFechaEnvioCG.Top
End If
If RegSesion("Programa") = True Then
Me.OPB_NoSubvencionado.Value = False
Me.OPB_FORCEM.Value = False
Me.OPB_Programa.Value = True
Me.CHK_ControlDeGestion.Enabled = True
Me.CHK_ControlDeGestion.Value = RegSesion("ControlGestion")
Me.LBL_FechaEnvioCG.ForeColor = 0
Me.dtFechaEnvioCG.Enabled = True
Me.TabCtl.Value = 0
Me.dtFechaEnvioCG.Enabled = True
Me.dtFechaEnvioCG.Value = IIf(Trim(RegSesion("FechaEnvioControlGestion") & " ") = "", Now,
RegSesion("FechaEnvioControlGestion"))
Me.dtFechaEnvioCG.Left = Me.dtFechaEnvioCG.Left
Me.dtFechaEnvioCG.Top = Me.dtFechaEnvioCG.Top
End If
```



```
Me.CHK_listadoDeFirmas.Value = RegSesion("ListadoFirmas")
Me.CHK_cuestionarios.Value = RegSesion("Cuestionarios")
Me.CHK_diplomas.Value = RegSesion("Diplomas")
Me.CHK_enviadoAGrabar.Value = RegSesion("CuestionariosEnviados")
If Me.CHK_enviadoAGrabar.Value = True Then
    Me.LBL_fechaEnvioCuestionarios.ForeColor = 0
    Me.dtFechaEnvioCuestionarios.Enabled = True
    Me.TabCtl.Value = 1
    Me.dtFechaEnvioCuestionarios.Value = Iif(Trim(RegSesion("FechaEnvioCuestionarios") & " ") = "",
Now, RegSesion("FechaEnvioCuestionarios"))
    Me.dtFechaEnvioCuestionarios.Left = Me.dtFechaEnvioCuestionarios.Left
    Me.dtFechaEnvioCuestionarios.Top = Me.dtFechaEnvioCuestionarios.Top
    Me.CMB_Evaluaciones.Enabled = True
    Me.LBL_VerValoraciones.ForeColor = 16711680
    Me.LBL_VerValoraciones.Caption = Iif(Trim(RegSesion("VinculoValoraciones") & " ") = "", "", "Ver
Valoraciones")
    Me.LBL_VerValoraciones.ControlTipText = RegSesion("VinculoValoraciones")
    Me.LBL_VerValoraciones.Hyperlink.Address = RegSesion("VinculoValoraciones")
Else
    Me.LBL_fechaEnvioCuestionarios.ForeColor = 12632256
    Me.dtFechaEnvioCuestionarios.Enabled = False
    Me.CMB_Evaluaciones.Enabled = False
    Me.LBL_VerValoraciones.ForeColor = 12632256
    Me.LBL_VerValoraciones.Caption = ""
    Me.LBL_VerValoraciones.ControlTipText = ""
    Me.LBL_VerValoraciones.Hyperlink.Address = ""
End If
Me.CHK_resultadosRecibidos.Value = RegSesion("CuestionariosRecibidos")
If Me.CHK_resultadosRecibidos.Value = True Then
    Me.LBL_fechaDeReciboCuestionarios.ForeColor = 0
    Me.dtFechaDeReciboCuestionarios.Enabled = True
    Me.dtFechaDeReciboCuestionarios.Left = Me.dtFechaDeReciboCuestionarios.Left
    Me.dtFechaDeReciboCuestionarios.Top = Me.dtFechaDeReciboCuestionarios.Top
    Me.TabCtl.Value = 1
    Me.dtFechaDeReciboCuestionarios.Value = RegSesion("FechaReciboCuestionarios")
    Me.LBL_valoracionCuestionarios.ForeColor = 0
    Me.TXT_valoracionCuestionarios.Enabled = True
    Me.TXT_valoracionCuestionarios.Value = RegSesion("ValoracionCurso")
Else
    Me.LBL_fechaDeReciboCuestionarios.ForeColor = 12632256
    Me.dtFechaDeReciboCuestionarios.Enabled = False
    Me.LBL_valoracionCuestionarios.ForeColor = 12632256
    Me.TXT_valoracionCuestionarios.Enabled = False
    Me.TXT_valoracionCuestionarios.Value = ""
End If
If RegSesion("SoftwareRequerido") = False Then
    Me.CHK_SoftwareNoRequerido.Value = True
    Me.CHK_SoftwareComprobado.Enabled = False
Else
    Me.CHK_SoftwareRequerido.Value = True
    Me.CHK_SoftwareComprobado.Enabled = True
End If
If RegSesion("SoftwareComprobado").Value = True Then
    Me.CHK_SoftwareComprobado.Value = True
Else
    Me.CHK_SoftwareComprobado.Value = False
End If

RegSesion.Close
```



```
' Muestra las otras tareas del curso
Me.SUBF_OtrasTareasSesion.Form.RecordSource = "SELECT CodigoCurso, ClaveSesion, Descripcion,
Realizado FROM OtrasTareasSesion WHERE CodigoCurso ='" & vCurso & "' AND ClaveSesion in ('" &
vSesion & "')"
Me.SUBF_OtrasTareasSesion.Requery
Me.SUBF_OtrasTareasSesion.Enabled = True

' Muestra las fechas de períodos del curso
Me.SUBF_FPeriodos.Form.RecordSource = "SELECT * FROM CursosPeriodos WHERE CodigoCurso
=''" & vCurso & "' AND ClaveSesion in ('" & vSesion & "')"
Me.SUBF_FPeriodos.Form.Requery
' Me.SUBF_FPeriodos.Form.Enabled = True

Me.CMB_Eliminar.Enabled = True

Me.TXT_NombreSesion.SetFocus
Me.CBX_NombreCurso.Enabled = False
Me.TXT_ClaveCurso.Enabled = False
Me.cmdMantenimientoCursos.Enabled = False
BD.Close

Me.TabCtl.Value = 0

' Activa el botón "Nuevo"
Me.cmdNuevaSesion.Enabled = True
End Sub

Private Sub CBX_centroAulaInterna_Click()
    CBX_AulaInterna.Value = ""
    CBX_AulaInterna.RowSource = "SELECT Aula FROM Aula_Interna WHERE IdCentro = " &
CBX_centroAulaInterna.Value ' Right(CBX_centroAulaInterna.Value, Len(CBX_centroAulaInterna.Value)
- 90)
    CBX_AulaInterna.Requery
End Sub

' Cuando selecciona un curso, escribe su código en el campo Clave.
Private Sub CBX_NombreCurso_Click()

    TXT_ClaveCurso.Value = CBX_NombreCurso.Column(1, CBX_NombreCurso.ListIndex)
    TXT_NombreSesion.Value = Trim(CBX_NombreCurso.Text)
    Registro.Close
    TXT_ClaveCurso.SetFocus
    TXT_ClaveCurso.SelStart = 7

' Verificar si existe plantilla para copiar los datos
Set BD = CurrentDb
Consulta = "SELECT * FROM DetalleSesion WHERE (CodigoCurso='" &
CBX_NombreCurso.Column(1, CBX_NombreCurso.ListIndex) & "' AND ClaveSesion='0000')"
Set Registro = BD.OpenRecordset(Consulta, dbOpenDynaset)
If Registro.RecordCount <> 0 Then
    ' Pregunta si se quiere utilizar la plantilla
    If MsgBox("Existe una plantilla para el curso seleccionado." & Chr(13) & Chr(10) & "¿Desea
utilizarla?", vbQuestion + vbYesNo, "Usar plantilla") = vbYes Then
        Me.vCurso = CBX_NombreCurso.Column(1, CBX_NombreCurso.ListIndex)
repite:
        Me.vSesion = InputBox("Introduce un código de sesión:", "Nuevo código de sesión")
        If vSesion <> "" Then
            ' Verifica el formato de la nueva sesión
```



```
If IsNumeric(vSesion) = False Or Len(vSesion) <> 4 Then
    MsgBox "Formato incorrecto.", vbCritical, "Error"
    GoTo repite
End If

' Copia los datos de la sesión al curso nuevo
BD.Execute "INSERT INTO Sesión SELECT CodigoCurso, " & vSesion & " as ClaveSesion,
NombreSesion, Format(Now(), 'yyyy/mm/dd') AS FechaInicio, Format(Now(), 'yyyy/mm/dd') AS FechaFin,
NombreArea, NombreDepartamento, IdResponsable, Horas, AsistentesMin, AsistentesMax, Escuela,
ParticipantesGrabados, ConvocatoriaEnviada, FechaEnvioConvocatoria, VinculoConvocatoria,
MaterialRequerido, MaterialReservado, ImporteMaterial, VinculoFacturaMaterial, FormadorCerrado,
FormadorInterno, NombreFormadorInterno, FormadorExterno, NombreFormadorExterno, CIFFormador,
ImporteFormador, Pagado, VinculoFacturaFormador, AulaReservada, FechaReservaAula, AulaInterna,
CentroAulaInterna, NombreAulaInterna, AulaProveedor, DireccionAulaProveedor, CosteProveedor,
ImporteProveedor, VinculoFacturaProveedor, RestauranteRequerido, RestauranteReservado,
ImporteRestaurante, VinculoFacturaRestaurante, TransporteRequerido, TransporteReservado,
ImporteTransporte, VinculoFacturaTransporte, Subvencionado, Forcem, " _
    & "Programa, ListadoFirmas, Cuestionarios, Diplomas, ControlGestion,
FechaEnvioControlGestion, CuestionariosEnviados, FechaEnvioCuestionarios, VinculoValoraciones,
CuestionariosRecibidos, FechaReciboCuestionarios, ValoracionCurso, SoftwareRequerido,
SoftwareComprobado FROM Sesión WHERE (CodigoCurso = " & CBX_NombreCurso.Column(1,
CBX_NombreCurso.ListIndex) & " AND ClaveSesion = '0000')"
```

```
' Copia las tareas del curso
BD.Execute "INSERT INTO OtrasTareasSesion SELECT CodigoCurso, " & vSesion & " as
ClaveSesion, Descripcion, Realizado FROM OtrasTareasSesion WHERE (((CodigoCurso)=" &
CBX_NombreCurso.Column(1, CBX_NombreCurso.ListIndex) & ") AND ((ClaveSesion)='0000'));"

' Muestra el curso nuevo
Me.SeleccionarCurso

MsgBox "Nuevo curso almacenado.", vbInformation, "Nuevo Curso"
Else
    MsgBox "No se ha introducido un código de sesión válido", vbCritical, "Error"
End If
End If
End If
Registro.Close
cmdNuevaSesion.Enabled = True
End Sub

Private Sub CBX_nombreFormadorExterno_Click()

Set BD = CurrentDb
Consulta = "SELECT * FROM Formador WHERE IdFormador = " &
CBX_nombreFormadorExterno.Value
Set Registro = BD.OpenRecordset(Consulta, dbOpenDynaset)
Registro.MoveFirst
TXT_CIFformadorExterno.Value = Registro("[cif_formador]")
Registro.Close
TXT_importeFormadorExterno.SetFocus

End Sub

Private Sub CHK_aulaReservada_Click()

If (OPB_AulaInterna.Value = 0) And (OPB_AulaProveedor.Value = 0) Then
    MsgBox "Para cerrar el aula hay que rellenar si es interna o de un proveedor", vbCritical, "Error aula"
    CHK_aulaReservada.Value = 0
```



```
ElseIf (OPB_AulaInterna.Value = -1) And (CBX_centroAulaInterna.ListIndex = -1) Then
    MsgBox "Para cerrar el aula hay que asignar un centro.", vbCritical, "Error aula"
    CHK_aulaReservada.Value = 0
    CBX_centroAulaInterna.SetFocus
ElseIf (OPB_AulaProveedor.Value = -1) And (TXT_DireccionAulaFormador.Value = "") Then
    MsgBox "Para cerrar el aula del proveedor hay que asignar una dirección.", vbCritical, "Error aula"
    CHK_aulaReservada.Value = 0
Else
    LBL_FechaReservaAulaFormador.ForeColor = 0
    dtFechaReservaAula.Enabled = True
    dtFechaReservaAula.Left = dtFechaReservaAula.Left
    dtFechaReservaAula.Top = dtFechaReservaAula.Top
    dtFechaReservaAula.SetFocus
End If

If CHK_aulaReservada.Value = 0 Then
    LBL_FechaReservaAulaFormador.ForeColor = 12632256
    CHK_aulaReservada.SetFocus
    dtFechaReservaAula.Enabled = False
End If

End Sub

Private Sub CHK_ControlDeGestion_Click()

    If CHK_ControlDeGestion.Value = True Then
        LBL_FechaEnvioCG.ForeColor = 0
        dtFechaEnvioCG.Enabled = True
        dtFechaEnvioCG.Left = dtFechaEnvioCG.Left
        dtFechaEnvioCG.Top = dtFechaEnvioCG.Top
        dtFechaEnvioCG.SetFocus
    Else
        LBL_FechaEnvioCG.ForeColor = 12632256
        dtFechaEnvioCG.Enabled = False
    End If

End Sub

Private Sub CHK_ConvocatoriaEnviada_Click()

    If CHK_ConvocatoriaEnviada = True Then
        LBL_FechaEnvioConvocatoria.ForeColor = 0
        dtFechaEnvioConvocatoria.Enabled = True
        dtFechaEnvioConvocatoria.Top = dtFechaEnvioConvocatoria.Top
        dtFechaEnvioConvocatoria.Left = dtFechaEnvioConvocatoria.Left
        CMD_VerConvocatoria.Enabled = True
        dtFechaEnvioConvocatoria.SetFocus
    Else
        LBL_FechaEnvioConvocatoria.ForeColor = 12632256
        dtFechaEnvioConvocatoria.Value = ""
        dtFechaEnvioConvocatoria.Enabled = False
        LBL_verConvocatoria.Caption = ""
        LBL_verConvocatoria.ControlTipText = ""
        LBL_verConvocatoria.Hyperlink.Address = ""
        CMD_VerConvocatoria.Enabled = False
    End If

End Sub
```



```
Private Sub CHK_enviadoAGrabar_Click()

    If CHK_resultadosRecibidos.Value = -1 Then
        MsgBox "No se puede borrar la opción de resultados enviados a grabar si éstos están recibidos",
vbCritical, "Error Encuestas"
        CHK_enviadoAGrabar.Value = -1
    Else
        If CHK_enviadoAGrabar = True Then
            LBL_fechaEnvioCuestionarios.ForeColor = 0
            dtFechaEnvioCuestionarios.Enabled = True
            dtFechaEnvioCuestionarios.Left = dtFechaEnvioCuestionarios.Left
            dtFechaEnvioCuestionarios.Top = dtFechaEnvioCuestionarios.Top
            dtFechaEnvioCuestionarios.SetFocus
        Else
            LBL_fechaEnvioCuestionarios.ForeColor = 12632256
            dtFechaEnvioCuestionarios.Enabled = False
        End If
    End If

End Sub

Private Sub CHK_formadorCerrado_Click()

    If (OPB_Interno.Value = 0) And (OPB_Externo.Value = 0) Then
        MsgBox "Para cerrar el proveedor hay que rellenar si es interno o externo", vbCritical, "Error proveedor"
        CHK_formadorCerrado.Value = 0
    ElseIf (OPB_Externo.Value = -1) And (CBX_nombreFormadorExterno.ListIndex = -1) Then
        MsgBox "Para cerrar el proveedor hay que asignar un nombre al proveedor externo", vbCritical, "Error
proveedor"
        CHK_formadorCerrado.Value = 0
        CBX_nombreFormadorExterno.SetFocus
    ElseIf (OPB_Interno.Value = -1) Then
        TXT_nombreFormadorInterno.SetFocus
        If TXT_nombreFormadorInterno.Text = "" Then
            MsgBox "Para cerrar el proveedor hay que asignar un nombre al proveedor interno", vbCritical, "Error
proveedor"
            CHK_formadorCerrado.Value = 0
        End If
    End If

End Sub

Private Sub CHK_MaterialNoRequerido_Click()

    CHK_MaterialRequerido.Value = False
    CHK_materialReservado.Value = 0
    CHK_materialReservado.Enabled = False
    LBL_materialReservado.ForeColor = 12632256
    LBL_ImporteMaterial.ForeColor = 12632256
    TXT_importeMaterial.Value = ""
    TXT_importeMaterial.Enabled = False
    CMD_verFacturaMaterial.Enabled = False
    LBL_verFacturaMaterial.Caption = ""
    LBL_verFacturaMaterial.ControlTipText = ""
    LBL_verFacturaMaterial.Hyperlink.Address = ""
    CHK_MaterialNoRequerido.Value = -1

End Sub
```



Private Sub CHK_MaterialRequerido_Click()

```
CHK_MaterialNoRequerido.Value = 0
CHK_materialReservado.Enabled = True
LBL_materialReservado.ForeColor = 0
LBL_ImporteMaterial.ForeColor = 0
TXT_importeMaterial.Enabled = True
TXT_importeMaterial.Value = 0
CMD_verFacturaMaterial.Enabled = True
CHK_MaterialRequerido.Value = -1
```

End Sub

Private Sub CHK_OtrasTareas_Click()

```
If CHK_OtrasTareas.Value = True Then
    TXT_OtrasTareasObservaciones.Enabled = True
    TXT_OtrasTareasObservaciones.SetFocus
Else
    TXT_OtrasTareasObservaciones.Value = ""
    TXT_OtrasTareasObservaciones.Enabled = False
End If
```

End Sub

Private Sub CHK_restauranteReservado_Click()

```
If (TXT_Restaurante.Value = "") Then
    MsgBox "Para cerrar el restaurante hay que indicar el lugar o el nombre de restaurante", vbCritical, "Error
    restaurante"
    CHK_restauranteReservado.Value = False
    TXT_Restaurante.SetFocus
End If
```

End Sub

Private Sub CHK_ResultadosRecibidos_Click()

```
If CHK_enviadoAGrabar.Value = -1 Then
    If CHK_resultadosRecibidos = True Then
        LBL_fechaDeReciboCuestionarios.ForeColor = 0
        dtFechaDeReciboCuestionarios.Enabled = True
        dtFechaDeReciboCuestionarios.Left = dtFechaDeReciboCuestionarios.Left
        dtFechaDeReciboCuestionarios.Top = dtFechaDeReciboCuestionarios.Top
        dtFechaDeReciboCuestionarios.SetFocus
        LBL_valoracionCuestionarios.ForeColor = 0
        TXT_valoracionCuestionarios.Enabled = True
        TXT_valoracionCuestionarios.Value = 0
        CMB_Evaluaciones.Enabled = True
        LBL_VerValoraciones.ForeColor = 0
    Else
        LBL_fechaDeReciboCuestionarios.ForeColor = 12632256
        dtFechaDeReciboCuestionarios.Enabled = False
        LBL_valoracionCuestionarios.ForeColor = 12632256
        TXT_valoracionCuestionarios.Value = ""
        TXT_valoracionCuestionarios.Enabled = False
        CMB_Evaluaciones.Enabled = False
        LBL_VerValoraciones.ForeColor = 12632256
        LBL_VerValoraciones.ControlTipText = ""
    End If
End If
```



```
    LBL_VerValoraciones.Hyperlink.Address = ""
    End If
Else
    MsgBox "No se pueden haber recibido las encuestas si no se han enviado a grabar", vbCritical, "Error Encuestas"
    CHK_resultadosRecibidos.Value = 0
    LBL_fechaDeREciboCuestionarios.ForeColor = 12632256
    dtFechaDeReciboCuestionarios.Enabled = False
    LBL_ValoracionCuestionarios.ForeColor = 12632256
    TXT_valoracionCuestionarios.Value = ""
    TXT_valoracionCuestionarios.Enabled = False
    CMB_Evaluaciones.Enabled = False
    LBL_VerValoraciones.ForeColor = 12632256
    LBL_VerValoraciones.ControlTipText = ""
    LBL_VerValoraciones.Hyperlink.Address = ""
    End If

End Sub

Private Sub CHK_SoftwareNoRequerido_Click()

    CHK_SoftwareRequerido.Value = False
    CHK_SoftwareComprobado.Value = False
    CHK_SoftwareComprobado.Enabled = False

End Sub

Private Sub CHK_SoftwareRequerido_Click()

    CHK_SoftwareNoRequerido.Value = False
    CHK_SoftwareComprobado.Enabled = True

End Sub

Private Sub CMB_Asistentes_Click()

    DoCmd.OpenForm "Asistentes"
    Forms![Asistentes]!LBL_Origen.Caption = "De Sesiones"
    Forms![Asistentes]!Curso.Value = CBX_NombreCurso.Value
    Forms![Asistentes]!Curso.Locked = True
    Forms![Asistentes]!Sesion.Enabled = True
    Forms![Asistentes]!Sesion.Value = Right(TXT_ClaveCurso.Value, 4)
    Forms![Asistentes]!Sesion.Locked = True
    Forms![Asistentes]!matricula.Value = ""
    Forms![Asistentes]!usuario.Value = ""
    Forms![Asistentes]!nombre.Value = ""
    Forms![Asistentes]!lista.RowSource = ""
    Forms![Asistentes]!Sesion.Enabled = True
    Forms![Asistentes]!matricula.Enabled = True
    Forms![Asistentes]!usuario.Enabled = True
    Forms![Asistentes]!nombre.Enabled = True
    Forms![Asistentes]!lista.Enabled = True
    Forms![Asistentes]!listaEspera.Enabled = True
    Forms![Asistentes]!CMB_Buscar.Enabled = True
    Form_Asistentes.ActualizaListas
    Forms![Asistentes]!matricula.SetFocus

End Sub
```



```
Private Sub CMB_Buscar_Click()

    DoCmd.OpenForm "BuscarCurso", , , , , acDialog

End Sub

Private Sub CMB_Eliminar_Click()

    Dim Curso As String
    Dim Sesion As String
    Dim BD As DAO.Database
    Dim Registro As DAO.Recordset
    Dim Consulta As String
    Dim Respuesta As Variant

    If TXT_ClaveCurso.Value <> "" Then
        Curso = Left(TXT_ClaveCurso.Value, 6)
        Sesion = Right(TXT_ClaveCurso.Value, 4)
    End If

    If Curso <> "" Then
        Set BD = CurrentDb
        Consulta = "SELECT * FROM Sesión WHERE CodigoCurso = '" & Curso & "'" AND ClaveSesion = '" & Sesion & "'"
        Set Registro = BD.OpenRecordset(Consulta, dbOpenDynaset)
        ' Si existe el curso, lo elimina
        If Registro.RecordCount > 0 Then
            If MsgBox("¿Está seguro que desea eliminar la sesión " & Curso & "-" & Sesion & "?", vbQuestion + vbYesNo, "Eliminar Sesión") = vbYes Then
                CurrentDb.Execute "update Convocado set Estado = 'Seleccionado', Sesion = '0000' WHERE Sesion = '" & Sesion & "' and Curso = '" & Curso & "' AND estado <> 'Seleccionado'"
                BD.Execute "DELETE * FROM Sesión WHERE CodigoCurso = '" & Curso & "' AND ClaveSesion = '" & Sesion & "'"
                ' Vacía la ventana
                cmdNuevaSesion_Click
            End If
        Else
            MsgBox "No se puede eliminar una sesión que todavía no ha sido guardada: " & Curso & "-" & Sesion & " ", vbCritical, "Error Eliminación"
        End If
        Registro.Close
        BD.Close
    End If

End Sub

Private Sub CMB_EnviarConvocatoria_Click()
    DoCmd.OpenForm "Enviados_F"
    Forms![Enviados_F]!TXT_Curso = CBX_NombreCurso.Column(1, CBX_NombreCurso.ListIndex)
    ' Forms![Enviados_F]!txtNombreCurso = CBX_NombreCurso.Column(0, CBX_NombreCurso.ListIndex)
    Forms![Enviados_F]!txtNombreCurso = TXT_NombreSesion
    Forms![Enviados_F]!TXT_Sesion.Value = Right(TXT_ClaveCurso.Value, 4)
    Forms![Enviados_F]!FormConvocados.Form.RecordSource = "SELECT Convocado.[Matricula HR], Convocado.[Apellidos, Nombre], Convocado.[Literal Cargo], Convocado.[Copia a jefe], Convocado.[Enviar], Convocado.[Estado] FROM Convocado WHERE ((convocado.curso='" & CBX_NombreCurso.Column(1, CBX_NombreCurso.ListIndex) & "') AND (convocado.sesion= '" & Right(TXT_ClaveCurso.Value, 4) & "') AND (convocado.estado <>'Seleccionado')) ORDER BY Convocado.[Apellidos, Nombre];"

```



```
Forms![Enviados_F]!txtTitulo = TXT_NombreSesion
' Consulta = "SELECT [Matricula HR], [Apellidos, Nombre], [Literal Cargo], [Copia a jefe], Enviar FROM
Convocado WHERE ((convocado.curso="" & CBX_NombreCurso.Column(1,
CBX_NombreCurso.ListIndex) & "") AND (convocado.sesion="" & Right(TXT_ClaveCurso.Value, 4) &
""));";
Forms![Enviados_F].Modal = True
End Sub

Private Sub CMB_Evaluaciones_Click()

' Pregunta el nombre de archivo de la hoja de valoración
With CDL_Abrir
.DialogTitle = "Seleccionar archivo de valoración" ' Título de la caja de diálogo.
.initdir = RutaBD ' Directorio por defecto.
.DefaultExt = "PDF" ' Extensión por defecto.
' Extensiones permitidas.
.Filter = "Archivos PDF de Adobe (*.pdf)|*.PDF|Todos los archivos (*.*)|*.*"
.FileName = ""
.ShowOpen
If .FileName <> "" Then ' Llamo a la ventana abrir archivo.
LBL_VerValoraciones.Visible = True
LBL_VerValoraciones.Caption = "Ver Valoraciones"
LBL_VerValoraciones.ControlTipText = .FileName
LBL_VerValoraciones.ForeColor = 16711680
LBL_VerValoraciones.Hyperlink.Address = .FileName
End If
End With
End Sub

Private Sub CMB_Guardar_Click()
Dim Curso As String
Dim Sesion As String
Dim BD As DAO.Database
Dim Registro As DAO.Recordset
Dim RegTemp As DAO.Recordset
Dim RegTareas As DAO.Recordset
Dim Consulta As String
Dim Respuesta As Variant
Dim GuardarCambios As Boolean

If ValidaCamposSesion Then
Curso = Left(TXT_ClaveCurso.Value, 6)
Sesion = Right(TXT_ClaveCurso.Value, 4)

If Sesion = "0000" Then
If MsgBox("Está intentando guardar una plantilla." & Chr(13) & Chr(10) & "¿Desea continuar?",
vbYesNo + vbQuestion, "Guardar Plantilla") = vbNo Then Exit Sub
End If

GuardarCambios = False
Set BD = CurrentDb
Consulta = "SELECT * FROM Sesión WHERE CodigoCurso = "" & Curso & "" AND ClaveSesion =
"" & Sesion & """"
Set Registro = BD.OpenRecordset(Consulta, dbOpenDynaset)
' Si existe el curso, actualiza la información
If Registro.RecordCount > 0 Then
Registro.MoveFirst
If Sesion = "0000" Then
```



```
If MsgBox("Ya existe la plantilla para el curso " & Curso & ". " & Chr(13) & Chr(10) & "¿Desea
sobreescribir los datos de la plantilla?", vbQuestion + vbYesNo, "Modificar Plantilla") = vbYes Then
    Registro.Edit
    GuardarCambios = True
End If
Else
    If MsgBox("Ya existe la sesión " & Sesion & " para este curso " & Curso & ". " & Chr(13) & Chr(10)
& "¿Desea sobreescribir los datos de la sesión?", vbQuestion + vbYesNo, "Modificar Curso") = vbYes Then
        Registro.Edit
        GuardarCambios = True
    End If
End If
' Si el curso es nuevo, da el alta
Else
    If Sesion <> "0000" Then
        If MsgBox("¿Desea guardar los datos del curso " & Curso & "-" & Sesion & "?", vbQuestion +
vbYesNo, "Nueva sesión de curso") = vbYes Then
            Registro.AddNew
            GuardarCambios = True
        End If
    Else
        If MsgBox("¿Desea guardar la plantilla del curso " & Curso & "?", vbQuestion + vbYesNo, "Nueva
plantilla de curso") = vbYes Then
            Registro.AddNew
            GuardarCambios = True
        End If
    End If
End If

If GuardarCambios Then
    ' Guardar los datos en sus correspondientes campos.
    Registro("CodigoCurso") = Curso
    Registro("ClaveSesion") = Sesion
    Registro("NombreSesion") = TXT_NombreSesion.Value
    Registro("FechaInicio") = Format(dtFechaInicio.Value, "yyyy/mm/dd")
    Registro("DetHorario") = txtDetHorario
    Registro("FechaFin") = Format(dtFechaFin.Value, "yyyy/mm/dd")
    Registro("NombreArea") = CBX_Area.Value
    Registro("NombreDepartamento") = CBX_Departamento.Value
    Registro("IdResponsable") = CBX_responsable.Value 'Right(CBX_responsable.Value,
Len(CBX_responsable.Value) - 40)
    Registro("Horas") = TXT_Horas.Value
    Registro("AsistentesMin") = TXT_asistentesMin.Value
    Registro("AsistentesMax") = TXT_asistentesMax.Value
    Registro("Escuela") = cmbEscuela.Value
    Registro("ParticipantesGrabados") = CHK_ParticipantesGrabados.Value
    Registro("ConvocatoriaEnviada") = CHK_ConvocatoriaEnviada.Value
    If dtFechaEnvioConvocatoria.Enabled = True Then
        Registro("FechaEnvioConvocatoria") = Format(dtFechaEnvioConvocatoria.Value, "yyyy/mm/dd")
    Else
        Registro("FechaEnvioConvocatoria") = Null
    End If

    If LBL_verConvocatoria.Caption <> "" Then
        Registro("VinculoConvocatoria") = LBL_verConvocatoria.Hyperlink.Address
    LBL_verConvocatoria.Caption
    Else
        Registro("VinculoConvocatoria") = ""
    End If
End If
```



```
If CHK_MaterialNoRequerido.Value = True Then
  Registro("MaterialRequerido") = False
End If
If CHK_MaterialRequerido.Value = True Then
  Registro("MaterialRequerido") = True
End If
Registro("MaterialReservado") = CHK_materialReservado.Value
If TXT_importeMaterial.Enabled = True Then
  Registro("ImporteMaterial") = TXT_importeMaterial.Value
Else
  Registro("ImporteMaterial") = 0
End If
If LBL_verFacturaMaterial.Caption <> "" Then
  Registro("VinculoFacturaMaterial") = LBL_verFacturaMaterial.Hyperlink.Address
Else
  Registro("VinculoFacturaMaterial") = ""
End If
Registro("FormadorCerrado") = CHK_formadorCerrado.Value
Registro("FormadorInterno") = OPB_Interno.Value
If TXT_nombreFormadorInterno.Enabled = True Then
  Registro("NombreFormadorInterno") = TXT_nombreFormadorInterno.Value
Else
  Registro("NombreFormadorInterno") = ""
End If
Registro("FormadorExterno") = OPB_Externo.Value
If CBX_nombreFormadorExterno.Enabled = True Then
  Registro("NombreFormadorExterno") = CBX_nombreFormadorExterno.Value
Right(CBX_nombreFormadorExterno.Value, Len(CBX_nombreFormadorExterno.Value) - 90)
Else
  Registro("NombreFormadorExterno") = -1
End If
If TXT_CIFformadorExterno.Enabled = True Then
  Registro("CIFFormador") = TXT_CIFformadorExterno.Value
Else
  Registro("CIFFormador") = ""
End If
If TXT_importeFormadorExterno.Enabled = True Then
  Registro("ImporteFormador") = TXT_importeFormadorExterno.Value
Else
  Registro("ImporteFormador") = 0
End If

If LBL_verFacturaFormador.Caption <> "" Then
  Registro("VinculoFacturaFormador") = LBL_verFacturaFormador.Hyperlink.Address
Else
  Registro("VinculoFacturaFormador") = ""
End If

Registro("AulaReservada") = CHK_aulaReservada.Value
If dtFechaReservaAula.Enabled = True Then
  Registro("FechaReservaAula") = Format(dtFechaReservaAula.Value, "yyyy/mm/dd")
Else
  Registro("FechaReservaAula") = Null
End If
Registro("AulaInterna") = OPB_AulaInterna.Value
If CBX_centroAulaInterna.Enabled = True Then
  If Trim(" " & CBX_centroAulaInterna.Value) = "" Then
    MsgBox "Debes seleccionar un aula interna.", vbCritical, "Faltan datos"
```



```
    CBX_centroAulaInterna.SetFocus
    Exit Sub
End If
    Registro("CentroAulaInterna") = CBX_centroAulaInterna.Value 'Right(CBX_centroAulaInterna.Value,
Len(CBX_centroAulaInterna.Value) - 90)
Else
    Registro("CentroAulaInterna") = -1
End If
If CBX_AulaInterna.Enabled = True Then
    Registro("NombreAulaInterna") = CBX_AulaInterna.Value
Else
    Registro("NombreAulaInterna") = ""
End If
Registro("AulaProveedor") = OPB_AulaProveedor.Value
If TXT_DireccionAulaFormador.Enabled = True Then
    Registro("DireccionAulaProveedor") = TXT_DireccionAulaFormador.Value
Else
    Registro("DireccionAulaProveedor") = ""
End If
Registro("CosteProveedor") = OPB_CosteAula.Value
Registro("Pagado") = chkPagado.Value
If TXT_importeAulaExterna.Enabled = True Then
    Registro("ImporteProveedor") = TXT_importeAulaExterna.Value
Else
    Registro("ImporteProveedor") = 0
End If
If LBL_verFacturaAula.Caption <> "" Then
    Registro("VinculoFacturaProveedor") = LBL_verFacturaAula.Hyperlink.Address
Else
    Registro("VinculoFacturaProveedor") = ""
End If
If OPB_RestauranteNoRequerido.Value = True Then
    Registro("RestauranteRequerido") = False
End If
If OPB_RestauranteRequerido.Value = True Then
    Registro("RestauranteRequerido") = True
End If
Registro("RestauranteReservado") = CHK_restauranteReservado.Value
If TXT_Restaurante.Enabled = True Then
    Registro("NombreRestaurante") = TXT_Restaurante.Value
Else
    Registro("NombreRestaurante") = ""
End If
If TXT_importeRestaurante.Enabled = True Then
    Registro("ImporteRestaurante") = TXT_importeRestaurante.Value
Else
    Registro("ImporteRestaurante") = 0
End If
If LBL_verFacturaRestaurante.Caption <> "" Then
    Registro("VinculoFacturaRestaurante") = LBL_verFacturaRestaurante.Hyperlink.Address
Else
    Registro("VinculoFacturaRestaurante") = ""
End If
If OPB_TransporteNoRequerido.Value = True Then
    Registro("TransporteRequerido") = False
End If
If OPB_TransporteRequerido.Value = True Then
    Registro("TransporteRequerido") = True
End If
```



```
Registro("TransporteReservado") = CHK_transporteReservado.Value
If TXT_importeTransporte.Enabled = True Then
    Registro("ImporteTransporte") = Trim("0" & TXT_importeTransporte.Value)
Else
    Registro("ImporteTransporte") = 0
End If
If LBL_verFacturaTransporte.Caption <> "" Then
    Registro("VinculoFacturaTransporte") = LBL_verFacturaTransporte.Hyperlink.Address
Else
    Registro("VinculoFacturaTransporte") = ""
End If
If OPB_NoSubvencionado.Value = True Then
    Registro("Subvencionado") = False
End If
If OPB_FORCEM.Value = True Then
    Registro("Forcem") = True
    Registro("Subvencionado") = True
Else
    Registro("Forcem") = False
End If
If OPB_Programa.Value = True Then
    Registro("Programa") = True
    Registro("Subvencionado") = True
Else
    Registro("Programa") = False
End If
Registro("ListadoFirmas") = CHK_listadoDeFirmas.Value
Registro("Cuestionarios") = CHK_cuestionarios.Value
Registro("Diplomas") = CHK_diplomas.Value
Registro("ControlGestion") = CHK_ControlDeGestion.Value
If dtFechaEnvioCG.Enabled = True Then
    Registro("FechaEnvioControlGestion") = Format(dtFechaEnvioCG.Value, "yyyy/mm/dd")
Else
    Registro("FechaEnvioControlGestion") = Null
End If
Registro("CuestionariosEnviados") = CHK_enviadoAGrabar.Value
If dtFechaEnvioCuestionarios.Enabled = True Then
    Registro("FechaEnvioCuestionarios") = Format(dtFechaEnvioCuestionarios.Value, "yyyy/mm/dd")
Else
    Registro("FechaEnvioCuestionarios") = Null
End If
If LBL_VerValoraciones.ControlTipText <> "" Then
    Registro("VinculoValoraciones") = LBL_VerValoraciones.ControlTipText
Else
    Registro("VinculoValoraciones") = ""
End If
Registro("CuestionariosRecibidos") = CHK_resultadosRecibidos.Value
If dtFechaDeReciboCuestionarios.Enabled = True Then
    Registro("FechaReciboCuestionarios") = Format(dtFechaDeReciboCuestionarios.Value,
"yyyy/mm/dd")
Else
    Registro("FechaReciboCuestionarios") = Null
End If
If TXT_valoracionCuestionarios.Enabled = True Then
    Registro("ValoracionCurso") = TXT_valoracionCuestionarios.Value
Else
    Registro("ValoracionCurso") = 0
End If
If CHK_SoftwareNoRequerido.Value = True Then
```



```
    Registro("SoftwareRequerido") = False
End If
If CHK_SoftwareRequerido.Value = True Then
    Registro("SoftwareRequerido") = True
End If
If CHK_SoftwareComprobado.Value = True Then
    Registro("SoftwareComprobado") = True
Else
    Registro("SoftwareComprobado") = False
End If
' Actualiza el registro
Registro.Update

' Habilita botones una vez que se ha almacenado la sesión
CMB_asistentes.Enabled = True
SUBF_OtrasTareasSesion.Enabled = True
Me.SUBF_FPeriodos.Enabled = True
CMB_enviarConvocatoria.Enabled = True

MsgBox "Se han guardado los datos", vbInformation, "Guardar"

' Si se han cambiado los datos se actualiza la vista
If Form_Sesiones.vActualizaAlertas = True Then
    GeneraAlarmas Form_ListaDeAlarmas.txtNombrePC, Curso, Sesion
    Form_ListaDeAlarmas.RellenaGrid
    DoCmd.Close
Else
    cmdNuevaSesion.Enabled = True
    DoCmd.OpenForm "Sesiones"
End If
' DoCmd.Close
End If

' cierra variables
Registro.Close
BD.Close

End If

End Sub

' Cierra el formulario.
Private Sub CMB_Salir_Click()

' Indica que se no deben actualizar los datos de la consulta si esta ventana es llamada desde otra
vActualizaVista = True
DoCmd.Close

End Sub

' Cuando se selecciona un área, se vuelcan los departamentos de nivel 5 de la misma.
Private Sub CBX_Area_Click()

CBX_Departamento.Value = ""
CBX_Departamento.RowSource = "SELECT DISTINCT [NNivel 5] FROM [TB_PERSONAS] WHERE ([NNivel 2]=''" & CBX_Area.Text & "'"') ORDER BY [NNivel 5]"
CBX_Departamento.Requery
```



End Sub

```
Private Sub CMD_VerConvocatoria_Click()
```

```
' Pregunta el nombre de archivo de convocatoria
With CDL_Abrir
.DialogTitle = "Seleccionar archivo de convocatoria" ' Título de la caja de diálogo.
.initdir = RutaBD ' Directorio por defecto.
.DefaultExt = "HTM" ' Extensión por defecto.
' Extensiones permitidas.
.Filter = "Archivos HTML (*.htm)*.HTM|Todos los archivos (*.*)*.*"
.FileName = ""
.ShowOpen
If .FileName <> "" Then ' Llamo a la ventana abrir archivo.
    LBL_verConvocatoria.Caption = "Haz clic aquí para ver la convocatoria" '.FileName
    LBL_verConvocatoria.ControlTipText = .FileName
    LBL_verConvocatoria.Hyperlink.Address = .FileName
End If
End With
End Sub
```

```
Private Sub CMD_VerFacturaAula_Click()
```

```
' Pregunta el nombre de archivo de factura de aula externa
With CDL_Abrir
.DialogTitle = "Seleccionar archivo de factura de aula externa" ' Título de la caja de diálogo.
.initdir = RutaBD ' Directorio por defecto.
.DefaultExt = "PDF" ' Extensión por defecto.
' Extensiones permitidas.
.Filter = "Archivos PDF de Adobe (*.pdf)*.PDF|Todos los archivos (*.*)*.*"
.FileName = ""
.ShowOpen
If .FileName <> "" Then ' Llamo a la ventana abrir archivo.
    LBL_verFacturaAula.Caption = "Haz clic para ver la factura" '.FileName
    LBL_verFacturaAula.ControlTipText = .FileName
    LBL_verFacturaAula.Visible = True
    LBL_verFacturaAula.Hyperlink.Address = .FileName
End If
End With
End Sub
```

```
Private Sub CMD_verFacturaFormador_Click()
```

```
' Pregunta el nombre de archivo de factura de formador
With CDL_Abrir
.DialogTitle = "Seleccionar archivo de factura de formador" ' Título de la caja de diálogo.
.initdir = RutaBD ' Directorio por defecto.
.DefaultExt = "PDF" ' Extensión por defecto.
' Extensiones permitidas.
.Filter = "Archivos PDF de Adobe (*.pdf)*.PDF|Todos los archivos (*.*)*.*"
.FileName = ""
.ShowOpen
If .FileName <> "" Then ' Llamo a la ventana abrir archivo.
    LBL_verFacturaFormador.Caption = "Haz clic aquí para ver la factura." '.FileName
    LBL_verFacturaFormador.Visible = True
    LBL_verFacturaFormador.ControlTipText = .FileName
    LBL_verFacturaFormador.Hyperlink.Address = .FileName
End If
```



End With
End Sub

Private Sub CMD_verFacturaMaterial_Click()

```
' Pregunta el nombre de archivo de factura de material
With CDL_Abrir
.DialogTitle = "Seleccionar archivo de factura de material" ' Título de la caja de diálogo.
.initdir = RutaBD ' Directorio por defecto.
.DefaultExt = "PDF" ' Extensión por defecto.
' Extensiones permitidas.
.Filter = "Archivos PDF (*.pdf)*.PDF|Todos los archivos (*.*)*.*"
.FileName = ""
.ShowOpen
If .FileName <> "" Then ' Llamo a la ventana abrir archivo.
    LBL_verFacturaMaterial.Caption = "Haz clic aquí para ver la factura." '.FileName
    LBL_verFacturaMaterial.ControlTipText = .FileName
    LBL_verFacturaMaterial.Hyperlink.Address = .FileName
End If
End With
```

End Sub

Private Sub CMD_VerFacturaRestaurante_Click()

```
' Pregunta el nombre de archivo de factura de restaurante
With CDL_Abrir
.DialogTitle = "Seleccionar archivo de factura restaurante" ' Título de la caja de diálogo.
.initdir = RutaBD ' Directorio por defecto.
.DefaultExt = "PDF" ' Extensión por defecto.
' Extensiones permitidas.
.Filter = "Archivos PDF (*.pdf)*.PDF|Todos los archivos (*.*)*.*"
.FileName = ""
.ShowOpen
If .FileName <> "" Then ' Llamo a la ventana abrir archivo.
    LBL_verFacturaRestaurante.Caption = "Haz clic aquí para ver la factura." '.FileName
    LBL_verFacturaRestaurante.ControlTipText = .FileName
    LBL_verFacturaRestaurante.Hyperlink.Address = .FileName
End If
End With
End Sub
```

Private Sub CMD_VerFacturaTransporte_Click()

```
' Pregunta el nombre de archivo de factura de transporte.
With CDL_Abrir
.DialogTitle = "Seleccionar archivo de factura de transporte" ' Título de la caja de diálogo.
.initdir = RutaBD ' Directorio por defecto.
.DefaultExt = "PDF" ' Extensión por defecto.
' Extensiones permitidas.
.Filter = "Archivos PDF (*.pdf)*.PDF|Todos los archivos (*.*)*.*"
.FileName = ""
.ShowOpen
If .FileName <> "" Then ' Llamo a la ventana abrir archivo.
    LBL_verFacturaTransporte.Caption = "Haz clic aquí para ver la factura." '.FileName
    LBL_verFacturaTransporte.ControlTipText = .FileName
    LBL_verFacturaTransporte.Hyperlink.Address = .FileName
End If
```



```
End With
End Sub

Private Sub cmdImprimir_Click()
    Dim mirpt As Report

    'abres el informe aplicando filtros
    DoCmd.OpenReport "Diplomas", acViewPreview, , "((CodigoCurso="" & vCurso & "") AND
(ClaveSesion="" & vSesion & "") AND (Estado='Confirmado'))", acWindowNormal
End Sub

Private Sub cmdMantenimientoCursos_Click()
    DoCmd.OpenForm "MantenimientoTabla"
    Form_MantenimientoTabla.txtTitulo.Caption = "Mantenimiento Cursos"
    Form_MantenimientoTabla.SUBF_Datos.SourceObject = "SUBF_DatosCursos"
    CBX_NombreCurso.Requery
    CBX_NombreCurso = ""
    TXT_ClaveCurso = ""
    TXT_NombreSesion = ""
End Sub

Private Sub cmdNuevaSesion_Click()
    Dim Ctrl As Control

    Me.CBX_NombreCurso.Enabled = True
    Me.CBX_NombreCurso = ""
    Me.CBX_NombreCurso.SetFocus

    For Each Ctrl In Me
        If TypeOf Ctrl Is CheckBox Then
            Ctrl.Value = False
            End If

        If TypeOf Ctrl Is DTPicker Then
            Ctrl.Value = Now
            Ctrl.Enabled = False
            End If

        If TypeOf Ctrl Is ComboBox Or TypeOf Ctrl Is TextBox Then
            Ctrl = ""
            If Ctrl.Tag <> 1 Then Ctrl.Enabled = False
            End If

        If TypeOf Ctrl Is CommandButton Then
            If Ctrl.Tag <> 1 Then Ctrl.Enabled = False
            End If
    Next

    Me.TXT_ClaveCurso.Enabled = True
    Me.TXT_ClaveCurso.Value = ""

    Me.dtFechaInicio.Value = Now
    Me.dtFechaFin.Value = Now

    Me.CBX_responsable = ""
```



Me.TXT_Horas = ""
Me.TXT_asistentesMin = ""
Me.CMB_asistentes.Enabled = False
Me.CBX_Area.Value = ""
Me.CHK_ParticipantesGrabados.Value = False
Me.CHK_ConvocatoriaEnviada.Value = False
Me.dtFechaEnvioConvocatoria.Enabled = True
Me.dtFechaEnvioConvocatoria = Now
Me.dtFechaEnvioConvocatoria.Enabled = False
Me.dtFechaEnvioConvocatoria.Enabled = False
Me.CMD_VerConvocatoria.Enabled = False
Me.CHK_enviadoAGrabar.Value = False
Me.CHK_resultadosRecibidos.Value = False
Me.LBL_fechaEnvioCuestionarios.ForeColor = 12632256
Me.TabCtl.Value = 1
Me.dtFechaEnvioCuestionarios = Now
Me.dtFechaEnvioCuestionarios.Enabled = False
Me.LBL_fechaDeREciboCuestionarios.ForeColor = 12632256
Me.dtFechaDeReciboCuestionarios = Now
Me.dtFechaDeReciboCuestionarios.Enabled = False
Me.LBL_ValoracionCuestionarios.ForeColor = 12632256
Me.TXT_valoracionCuestionarios.Enabled = False
Me.CMD_verFacturaFormador.Enabled = False
Me.CHK_ControlDeGestion.Value = False
Me.CHK_cuestionarios.Value = False
Me.CHK_listadoDeFirmas.Value = False
Me.CHK_diplomas.Value = False
Me.LBL_FechaEnvioCG.Visible = False
Me.dtFechaEnvioCG = Now
Me.dtFechaEnvioCG.Enabled = False



Me.CHK_materialReservado.Value = False
Me.LBL_materialReservado.ForeColor = 12632256
Me.CMD_verFacturaMaterial.Enabled = False
Me.LBL_ImporteMaterial.ForeColor = 12632256
Me.LBL_nombreFormadorExterno.ForeColor = 12632256
Me.LBL_CIFformadorExterno.ForeColor = 12632256
Me.TXT_CIFformadorExterno = ""
Me.TXT_CIFformadorExterno.Enabled = False
Me.LBL_importeFormador.ForeColor = 12632256
Me.LBL_nombreFormadorInterno.ForeColor = 12632256
Me.LBL_centroAulaInterna.ForeColor = 12632256
Me.LBL_FechaReservaAulaFormador.ForeColor = 12632256
Me.TabCtl.Value = 0
Me.dtFechaReservaAula = Now
Me.dtFechaReservaAula.Enabled = False
Me.LBL_direccionAulaFormador.ForeColor = 12632256
Me.LBL_ImporteAulaExterna.ForeColor = 12632256
Me.CMD_verFacturaAula.Enabled = False
Me.TXT_nombreFormadorInterno.Value = ""
Me.TXT_nombreFormadorInterno.Enabled = False
Me.TXT_importeFormadorExterno.Value = ""
Me.TXT_importeFormadorExterno.Enabled = False
Me.TXT_importeMaterial.Value = ""
Me.TXT_importeMaterial.Enabled = False
Me.TXT_DireccionAulaFormador.Value = ""
Me.TXT_DireccionAulaFormador.Enabled = False
Me.TXT_importeAulaExterna.Value = ""
Me.TXT_importeAulaExterna.Enabled = False
Me.LBL_verConvocatoria.Caption = ""



```
Me.LBL_verFacturaFormador.Caption = ""

' Blanquea enlaces
Me.LBL_verFacturaAula.Caption = ""

Me.LBL_verFacturaTransporte.Caption = ""

Me.LBL_verFacturaRestaurante.Caption = ""

Me.LBL_VerValoraciones.Caption = ""

' No muestra ninguna de las otras tareas del curso
Me.SUBF_OtrasTareasSesion.Form.RecordSource = "SELECT CodigoCurso, ClaveSesion, Descripcion,
Realizado FROM OtrasTareasSesion WHERE CodigoCurso ='11' "

Me.SUBF_OtrasTareasSesion.Requery

Me.SUBF_OtrasTareasSesion.Enabled = True

' No muestra ninguna fecha de período
Me.SUBF_FPeriodos.Form.RecordSource = "SELECT * FROM CursosPeriodos WHERE CodigoCurso
='11' "

Me.SUBF_FPeriodos.Form.Requery

Me.LBL_verFacturaMaterial.Caption = ""

Me.cmdMantenimientoCursos.Enabled = True

Exit Sub

MsgBox Me.LBL_verFacturaFormador

MsgBox Me.CMB_enviarConvocatoria

MsgBox Me.LBL_enviarConvocatoria

MsgBox Me.LBL_NombreAula

MsgBox Me.TXT_asistentesMax

MsgBox Me.LBL_asistentesMax

MsgBox Me.LBL_AsistentesAcuden

MsgBox Me.LBL_asistentesMin

MsgBox Me.TXT_asistentesAcuden

MsgBox Me.CHK_aulaReservada

MsgBox Me.LBL_aulaReservada

MsgBox Me.CHK_formadorCerrado
```



MsgBox Me.LBL_FormadorCerrado
MsgBox Me.CBX_centroAulaInterna
MsgBox Me.CBX_AulaInterna
MsgBox Me.CBX_Departamento
MsgBox Me.CBX_nombreFormadorExterno
MsgBox Me.LNE_Encabezado
MsgBox Me.CMB_Guardar
MsgBox Me.CMB_Eliminar
MsgBox Me.LBL_DepartamentoFormacion
MsgBox Me.IMG_ET
MsgBox Me.LBL_Convocatorias
MsgBox Me.CMB_Salir
MsgBox Me.cmdImprimir
MsgBox Me.CMB_Buscar
MsgBox Me.CDL_Abrir
MsgBox Me.CMB_Evaluaciones
MsgBox Me.LBL_formador
MsgBox Me.OPB_Interno
MsgBox Me.LBL_FormadorInterno
MsgBox Me.CUA_FormadorInterno
MsgBox Me.OPB_Externo
MsgBox Me.LBL_FormadorExterno
MsgBox Me.Cuadro46
MsgBox Me.LBL_material
MsgBox Me.CHK_MaterialNoRequerido
MsgBox Me.LBL_MaterialNoRequerido
MsgBox Me.CUA_MaterialNoRequerido
MsgBox Me.CHK_MaterialRequerido
MsgBox Me.LBL_MaterialRequerido
MsgBox Me.LNE_MaterialRequerido3



MsgBox Me.LNE_MaterialRequerido4
MsgBox Me.LNE_MaterialRequerido5
MsgBox Me.LNE_MaterialRequerido2
MsgBox Me.LNE_MaterialRequerido1
MsgBox Me.LNE_MaterialRequerido6
MsgBox Me.LBL_SubvencionArchivo
MsgBox Me.CUA_SubcuadroConvocatorias
MsgBox Me.LBL_aula
MsgBox Me.OPB_AulaInterna
MsgBox Me.LBL_AulaInterna
MsgBox Me.CUA_AulaInterna
MsgBox Me.OPB_AulaProveedor
MsgBox Me.LBL_AulaProveedor
MsgBox Me.OPB_CosteAula
MsgBox Me.LBL_CosteAula
MsgBox Me.CUA_AulaExterna
MsgBox Me.OPB_NoSubvencionado
MsgBox Me.LBL_NoSubvencionado
MsgBox Me.CUA_Subvenciones
MsgBox Me.OPB_Programa
MsgBox Me.LBL_Programa
MsgBox Me.OPB_FORCEM
MsgBox Me.LBL_FORCEM
MsgBox Me.CUA_transporte
MsgBox Me.CHK_transporteReservado
MsgBox Me.LBL_TransporteReservado
MsgBox Me.CMD_verFacturaTransporte
MsgBox Me.LBL_ImporteTransporte
MsgBox Me.TXT_importeTransporte
MsgBox Me.CUA_restaurante



MsgBox Me.CHK_restauranteReservado
MsgBox Me.LBL_RestauranteReservado
MsgBox Me.CMD_verFacturaRestaurante
MsgBox Me.LBL_importeRestaurante
MsgBox Me.LBL_nombreRestaurante
MsgBox Me.TXT_importeRestaurante
MsgBox Me.LBL_verFacturaRestaurante
MsgBox Me.LBL_restaurante
MsgBox Me.LBL_transporte
MsgBox Me.OPB_RestauranteNoRequerido
MsgBox Me.LBL_RestauranteNoRequerido
MsgBox Me.CUA_RestauranteNoRequerido
MsgBox Me.OPB_RestauranteRequerido
MsgBox Me.LBL_RestauranteRequerido
MsgBox Me.LNE_RestauranteRequerido3
MsgBox Me.LNE_RestauranteRequerido4
MsgBox Me.LNE_RestauranteRequerido5
MsgBox Me.LNE_RestauranteRequerido2
MsgBox Me.LNE_RestauranteRequerido1
MsgBox Me.LNE_RestauranteRequerido6
MsgBox Me.TXT_Restaurante
MsgBox Me.OPB_TransporteNoRequerido
MsgBox Me.LBL_TransporteNoRequerido
MsgBox Me.OPB_TransporteRequerido
MsgBox Me.LBL_TransporteRequerido
MsgBox Me.CUA_TransporteNoRequerido
MsgBox Me.LNE_TransporteRequerido3
MsgBox Me.LNE_TransporteRequerido4
MsgBox Me.LNE_TransporteRequerido5
MsgBox Me.LNE_TransporteRequerido2



MsgBox Me.LNE_TransporteRequerido6

MsgBox Me.LNE_TransporteRequerido1

MsgBox Me.CUA_OtrasTareas

MsgBox Me.LBL_OtrasTareas

MsgBox Me.SUBF_OtrasTareasSesion

MsgBox Me.CHK_SoftwareNoRequerido

MsgBox Me.Etiqueta127

MsgBox Me.CUA_SoftwareNoRequerido

MsgBox Me.CHK_SoftwareRequerido

MsgBox Me.Etiqueta130

MsgBox Me.CUA_SoftwareRequerido

MsgBox Me.CHK_SoftwareComprobado

MsgBox Me.Etiqueta133

MsgBox Me.TXT_NombreSesion

MsgBox Me.Etiqueta137

MsgBox Me.cmbEscuela

MsgBox Me.Etiqueta139

MsgBox Me.chkPagado

MsgBox Me.texto1111

MsgBox Me.Etiqueta142

MsgBox Me.txtAsistentesConvocados

MsgBox Me.cmdNuevaSesion

End Sub

```
Private Sub cmdModificaProveedor_Click()  
    DoCmd.OpenForm "MantenimientoTabla"  
    Form_MantenimientoTabla.txtTitulo.Caption = "Mantenimiento Proveedor"  
    CBX_nombreFormadorExterno.Requery  
End Sub
```

```
Private Sub Form_Close()
```

```
    ' Dim BD As DAO.Database  
    ' Dim Registro As DAO.Recordset  
    ' Dim Consulta As String  
    '  
    ' If TXT_ClaveCurso.Value <> "" Then
```



```
' Set BD = CurrentDb
' Consulta = "SELECT * FROM OtrasTareasSesionTemp WHERECodigoCurso = "" &
Left(TXT_ClaveCurso.Value, 6) & "" AND ClaveSesion = "" & Right(TXT_ClaveCurso, 4) & ""
' Set Registro = BD.OpenRecordset(Consulta, dbOpenDynaset)
' If Registro.RecordCount > 0 Then
'   While Not Registro.EOF
'     Registro.Delete
'     Registro.MoveNext
'   Wend
' End If
' Registro.Close
' BD.Close
' End If
```

End Sub

```
Private Sub Form_Load()
  vCurso = ""
  vSeccion = ""
  vActualizaAlertas = False
  dtFechaInicio.Value = Format(Now, "dd/mm/yyyy")
  dtFechaFin = Format(Now, "dd/mm/yyyy")
  dtFechaReservaAula.Enabled = True
  dtFechaReservaAula.Value = Format(Now, "dd/mm/yyyy")
  dtFechaReservaAula.Enabled = False
  dtFechaEnvioConvocatoria.Enabled = True
  dtFechaEnvioConvocatoria.Value = Format(Now, "dd/mm/yyyy")
  dtFechaEnvioConvocatoria.Enabled = False
  dtFechaEnvioCG.Enabled = True
  dtFechaEnvioCG.Value = Format(Now, "dd/mm/yyyy")
  dtFechaEnvioCG.Enabled = False
  dtFechaEnvioCuestionarios.Enabled = True
  dtFechaEnvioCuestionarios.Value = Format(Now, "dd/mm/yyyy")
  dtFechaEnvioCuestionarios.Enabled = False
  dtFechaDeReciboCuestionarios.Enabled = True
  dtFechaDeReciboCuestionarios.Value = Format(Now, "dd/mm/yyyy")
  dtFechaDeReciboCuestionarios.Enabled = False
```

End Sub

' Inicialización de variables y parámetros a la apertura del formulario.

```
Private Sub Form_Open(Cancel As Integer)
```

```
  Set BD = CurrentDb
```

```
End Sub
```

```
Private Sub OPB_AulaInterna_Click()
```

```
If OPB_AulaInterna.Value = True Then
  LBL_centroAulaInterna.ForeColor = 0
  CBX_centroAulaInterna.Enabled = True
  CBX_centroAulaInterna.SetFocus
  LBL_NombreAula.ForeColor = 0
  CBX_AulaInterna.Enabled = True
  CHK_aulaReservada.Value = False
Else
  LBL_centroAulaInterna.ForeColor = 12632256
  CBX_centroAulaInterna.Value = ""
  CBX_centroAulaInterna.Enabled = False
```



```
LBL_NombreAula.ForeColor = 12632256
CBX_AulaInterna.Value = ""
CBX_AulaInterna.Enabled = False
If OPB_AulaProveedor.Value = False Then
    LBL_FechaReservaAulaFormador.ForeColor = 12632256
    CHK_aulaReservada.Value = False
    CHK_aulaReservada.SetFocus
    dtFechaReservaAula.Enabled = False
End If
End If
```

End Sub

Private Sub OPB_AulaProveedor_Click()

```
If OPB_AulaProveedor.Value = True Then
    LBL_direccionAulaFormador.ForeColor = 0
    OPB_CosteAula.Enabled = True
    TXT_DireccionAulaFormador.Enabled = True
    TXT_DireccionAulaFormador.SetFocus
    CHK_aulaReservada.Value = False
Else
    LBL_direccionAulaFormador.ForeColor = 12632256
    OPB_CosteAula.Value = 0
    OPB_CosteAula.Enabled = False
    TXT_DireccionAulaFormador.Value = ""
    TXT_DireccionAulaFormador.Enabled = False
    LBL_ImporteAulaExterna.ForeColor = 12632256
    TXT_importeAulaExterna.Enabled = False
    TXT_importeAulaExterna.Value = ""
    CMD_verFacturaAula.Enabled = False
    LBL_verFacturaAula.Caption = ""
    LBL_verFacturaAula.Hyperlink.Address = ""
    LBL_verFacturaAula.ControlTipText = ""
    If OPB_AulaInterna.Value = False Then
        LBL_FechaReservaAulaFormador.ForeColor = 12632256
        CHK_aulaReservada.Value = False
        CHK_aulaReservada.SetFocus
        dtFechaReservaAula.Enabled = False
    End If
End If
```

End Sub

Private Sub OPB_CosteAula_Click()

```
If OPB_CosteAula.Value = True Then
    LBL_ImporteAulaExterna.ForeColor = 0
    TXT_importeAulaExterna.Enabled = True
    TXT_importeAulaExterna.Value = 0
    TXT_importeAulaExterna.SetFocus
    CMD_verFacturaAula.Enabled = True
Else
    LBL_ImporteAulaExterna.ForeColor = 12632256
    TXT_importeAulaExterna.Enabled = False
    TXT_importeAulaExterna.Value = ""
    CMD_verFacturaAula.Enabled = False
    LBL_verFacturaAula.Caption = ""
    LBL_verFacturaAula.Hyperlink.Address = ""
```



```
    LBL_verFacturaAula.ControlTipText = ""
End If

End Sub

Private Sub OPB_Externo_Click()

If OPB_Externo.Value = True Then
    LBL_nombreFormadorExterno.ForeColor = 0
    CBX_nombreFormadorExterno.Enabled = True
    LBL_CIFformadorExterno.ForeColor = 0
    TXT_CIFformadorExterno.Enabled = True
    TXT_CIFformadorExterno.Locked = True
    LBL_importeFormador.ForeColor = 0
    TXT_importeFormadorExterno.Enabled = True
    TXT_importeFormadorExterno.Value = 0
    CMD_verFacturaFormador.Enabled = True
    CBX_nombreFormadorExterno.SetFocus
    CHK_formadorCerrado.Value = False
    chkPagado.Enabled = True
    cmdModificaProveedor.Enabled = True
Else
    LBL_nombreFormadorExterno.ForeColor = 12632256
    CBX_nombreFormadorExterno.Enabled = False
    LBL_CIFformadorExterno.ForeColor = 12632256
    TXT_CIFformadorExterno.Enabled = False
    TXT_CIFformadorExterno.Locked = False
    LBL_importeFormador.ForeColor = 12632256
    TXT_importeFormadorExterno.Enabled = False
    CMD_verFacturaFormador.Enabled = False
    CBX_nombreFormadorExterno.Value = ""
    TXT_CIFformadorExterno.Value = ""
    TXT_importeFormadorExterno.Value = ""
    LBL_verFacturaFormador.Caption = ""
    LBL_verFacturaFormador.ControlTipText = ""
    LBL_verFacturaFormador.Hyperlink.Address = ""
    cmdModificaProveedor.Enabled = False
    chkPagado.Enabled = False
    If OPB_Interno.Value = False Then
        CHK_formadorCerrado.Value = False
        CHK_formadorCerrado.SetFocus
    End If
End If

End Sub

Private Sub OPB_FORCEM_Click()

If OPB_FORCEM.Value = True Then
    OPB_NoSubvencionado.Value = False
    OPB_Programa.Value = False
    CHK_ControlDeGestion.Enabled = True
Else
    CHK_ControlDeGestion.Value = False
    CHK_ControlDeGestion.Enabled = False
    LBL_FechaEnvioCG.ForeColor = 12632256
    dtFechaEnvioCG.Value = ""
    dtFechaEnvioCG.Enabled = False
End If
```



End Sub

Private Sub OPB_Interno_Click()

```
If OPB_Interno.Value = True Then
    LBL_nombreFormadorInterno.ForeColor = 0
    TXT_nombreFormadorInterno.Enabled = True
    TXT_nombreFormadorInterno.SetFocus
    CHK_formadorCerrado.Value = False
Else
    LBL_nombreFormadorInterno.ForeColor = 12632256
    TXT_nombreFormadorInterno.Enabled = False
    TXT_nombreFormadorInterno.Value = ""
    If OPB_Externo.Value = False Then
        CHK_formadorCerrado.Value = False
        CHK_formadorCerrado.SetFocus
    End If
End If
```

End Sub

Private Sub OPB_NoSubvencionado_Click()

```
If OPB_NoSubvencionado.Value = True Then
    OPB_FORCEM.Value = False
    OPB_Programa.Value = False
    CHK_ControlDeGestion.Value = False
    CHK_ControlDeGestion.Enabled = False
    LBL_fechaEnvioCG.ForeColor = 12632256
    dtFechaEnvioCG.Value = ""
    dtFechaEnvioCG.Enabled = False
End If
```

End Sub

Private Sub OPB_Programa_Click()

```
If OPB_Programa.Value = True Then
    OPB_NoSubvencionado.Value = False
    OPB_FORCEM.Value = False
    CHK_ControlDeGestion.Enabled = True
Else
    CHK_ControlDeGestion.Value = False
    CHK_ControlDeGestion.Enabled = False
    LBL_fechaEnvioCG.ForeColor = 12632256
    dtFechaEnvioCG.Value = ""
    dtFechaEnvioCG.Enabled = False
End If
```

End Sub

Private Sub OPB_RestauranteNoRequerido_Click()

```
OPB_RestauranteRequerido.Value = False
CHK_restauranteReservado.Value = 0
CHK_restauranteReservado.Enabled = False
LBL_RestauranteReservado.ForeColor = 12632256
LBL_nombreRestaurante.ForeColor = 12632256
```



```
TXT_Restaurante.Value = ""  
TXT_Restaurante.Enabled = False  
LBL_importeRestaurante.ForeColor = 12632256  
TXT_importeRestaurante.Value = ""  
TXT_importeRestaurante.Enabled = False  
CMD_verFacturaRestaurante.Enabled = False  
LBL_verFacturaRestaurante.Caption = ""  
LBL_verFacturaRestaurante.ControlTipText = ""  
LBL_verFacturaRestaurante.Hyperlink.Address = ""  
OPB_RestauranteNoRequerido.Value = -1
```

End Sub

Private Sub OPB_RestauranteRequerido_Click()

```
OPB_RestauranteNoRequerido.Value = False  
CHK_restauranteReservado.Value = 0  
CHK_restauranteReservado.Enabled = True  
LBL_RestauranteReservado.ForeColor = 0  
LBL_importeRestaurante.ForeColor = 0  
TXT_importeRestaurante.Enabled = True  
TXT_importeRestaurante.Value = 0  
LBL_nombreRestaurante.ForeColor = 0  
TXT_Restaurante.Enabled = True  
TXT_Restaurante.SetFocus  
CMD_verFacturaRestaurante.Enabled = True  
OPB_RestauranteRequerido.Value = -1
```

End Sub

Private Sub OPB_TransporteNoRequerido_Click()

```
OPB_TransporteRequerido.Value = 0  
CHK_transporteReservado.Value = 0  
CHK_transporteReservado.Enabled = False  
LBL_ImporteTransporte.ForeColor = 12632256  
TXT_importeTransporte.Value = ""  
TXT_importeTransporte.Enabled = False  
CMD_verFacturaTransporte.Enabled = False  
LBL_verFacturaTransporte.Caption = ""  
LBL_verFacturaTransporte.ControlTipText = ""  
LBL_verFacturaTransporte.Hyperlink.Address = ""  
OPB_TransporteNoRequerido.Value = -1
```

End Sub

Private Sub OPB_TransporteRequerido_Click()

```
OPB_TransporteNoRequerido.Value = False  
CHK_transporteReservado.Enabled = True  
LBL_ImporteTransporte.ForeColor = 0  
TXT_importeTransporte.Enabled = True  
TXT_importeTransporte.Value = 0  
TXT_importeTransporte.SetFocus  
CMD_verFacturaTransporte.Enabled = True  
OPB_TransporteRequerido.Value = -1
```

End Sub



Private Function ValidaCamposSesion() As Boolean

ValidaCamposSesion = False

' Validación del contenido de combo de nombre de curso

If CBX_NombreCurso.Value = "" Then

MsgBox "Para poder grabar el curso es necesario introducir el Nombre", vbCritical, "Falta Nombre Curso"

CBX_NombreCurso.SetFocus

Exit Function

End If

' Validación de contenido del código de curso

If TXT_ClaveCurso.Value = "" Or Len(TXT_ClaveCurso.Value) < 10 Then

MsgBox "Código de curso erróneo o incompleto.", vbCritical, "Falta Código Curso"

TXT_ClaveCurso.SetFocus

Exit Function

End If

' Validación de fechas

If CDate(dtFechaInicio.Value) > CDate(dtFechaFin.Value) Then

MsgBox "La Fecha de Inicio no puede ser mayor que la Fecha de Fin", vbCritical, "Error Fechas"

dtFechaInicio.SetFocus

Exit Function

End If

' Validación de combo de área

If Trim(CBX_Area.Value & " ") = "" Then

MsgBox "Para poder grabar el curso es necesario introducir el Área solicitante", vbCritical, "Falta Área Solicitante"

CBX_Area.SetFocus

Exit Function

End If

' Validación de combo de responsable

If Trim(CBX_responsable.Value & " ") = "" Then

MsgBox "Para poder grabar el curso es necesario introducir el Responsable", vbCritical, "Falta Responsable Curso"

CBX_responsable.SetFocus

Exit Function

End If

' Validación de contenido horas

If TXT_Horas.Value = "" Then

MsgBox "Para poder grabar el curso es necesario introducir las Horas", vbCritical, "Faltan Horas Curso"

TXT_Horas.SetFocus

Exit Function

End If

' Validación de contenido de asistentes mínimo

If Trim(TXT_asistentesMin.Value & " ") = "" Then

MsgBox "Para poder grabar el curso es necesario introducir el Número Mínimo de Asistentes", vbCritical, "Faltan Asistentes Mínimos"

TXT_asistentesMin.SetFocus

Exit Function

End If

' Validación de contenido de asistente máximo

If Trim(TXT_asistentesMax.Value = "" & " ") Then



```
MsgBox "Para poder grabar el curso es necesario introducir el Número Máximo de Asistentes.",
vbCritical, "Faltan Asistentes Máximos"
    TXT_asistentesMax.SetFocus
Exit Function
End If

' Validación de asistentes mínimos
If CInt(TXT_asistentesMin.Value) > CInt(TXT_asistentesMax.Value) Then
    MsgBox "El número mínimo de asistentes no puede ser mayor que el número máximo.", vbCritical,
"Error Asistentes"
    TXT_asistentesMin.SetFocus
Exit Function
End If

' Validación del nombre del instructor cuando se indica que es interno
If OPB_Interno.Value = True And Trim(TXT_nombreFormadorInterno & " ") = "" Then
    MsgBox "No se ha indicado el nombre del instructor.", vbCritical, "Falta instructor"
    TXT_nombreFormadorInterno.SetFocus
Exit Function
End If

' Validación de selección de proveedor cuando se ha indicado que es externo
If OPB_Externo.Value = True And Trim(CBX_nombreFormadorExterno & " ") = "" Then
    MsgBox "No se ha seleccionado proveedor.", vbCritical, "Falta Proveedor"
    CBX_nombreFormadorExterno.SetFocus
Exit Function
End If

ValidaCamposSesion = True

End Function
```

SubF Candidatos

Option Compare Database

```
Private Sub Apellidos__Nombre_Click()
    Form_Asisntentes.moverconvcand.Enabled = False
    Form_Asisntentes.movertodosconvcand.Enabled = False
    Form_Asisntentes.confirmar.Enabled = False
    Form_Asisntentes.cancelar.Enabled = False
    Form_Asisntentes.movertodoscandconv.Enabled = False
    Form_Asisntentes.movercandconv.Enabled = False
    Form_Asisntentes.suprimir.Enabled = False
End Sub

Private Sub Apellidos__Nombre_KeyDown(KeyCode As Integer, Shift As Integer)
    Form_Asisntentes.moverconvcand.Enabled = False
    Form_Asisntentes.movertodosconvcand.Enabled = False
    Form_Asisntentes.confirmar.Enabled = False
    Form_Asisntentes.cancelar.Enabled = False
    Form_Asisntentes.movertodoscandconv.Enabled = False
    Form_Asisntentes.movercandconv.Enabled = False
    Form_Asisntentes.suprimir.Enabled = False
End Sub

Private Sub Cancelaciones_Click()
```



```
Apellidos__Nombre_Click  
End Sub
```

```
Private Sub Comentarios_Click()  
Apellidos__Nombre_Click  
End Sub
```

```
Private Sub Form_Click()  
If Me.Form.Recordset.RecordCount = 0 Then Exit Sub  
Form_Asistentes.vNombre = Trim(" " & Me.Form![Apellidos, Nombre])  
Form_Asistentes.vUsuario = Trim(" " & Me.Form![Us3270])  
Form_Asistentes.moverconvcand.Enabled = False  
Form_Asistentes.movertodosconvcand.Enabled = False  
Form_Asistentes.confirmar.Enabled = False  
Form_Asistentes.cancelar.Enabled = False  
Form_Asistentes.movertodoscandconv.Enabled = True  
Form_Asistentes.movercandconv.Enabled = True  
Form_Asistentes.suprimir.Enabled = True  
  
End Sub
```

```
Private Sub Form_Current()  
Form_Asistentes.vNombre = ""  
Form_Asistentes.vUsuario = ""  
End Sub
```

SubF Convocados

Option Compare Database

```
Private Sub Apellidos__Nombre_Click()  
Form_Asistentes.moverconvcand.Enabled = False  
Form_Asistentes.movertodosconvcand.Enabled = False  
Form_Asistentes.confirmar.Enabled = False  
Form_Asistentes.cancelar.Enabled = False  
Form_Asistentes.movertodoscandconv.Enabled = False  
Form_Asistentes.movercandconv.Enabled = False  
Form_Asistentes.suprimir.Enabled = False  
End Sub
```

```
Private Sub Apellidos__Nombre_KeyDown(KeyCode As Integer, Shift As Integer)  
Form_Asistentes.moverconvcand.Enabled = False  
Form_Asistentes.movertodosconvcand.Enabled = False  
Form_Asistentes.confirmar.Enabled = False  
Form_Asistentes.cancelar.Enabled = False  
Form_Asistentes.movertodoscandconv.Enabled = False  
Form_Asistentes.movercandconv.Enabled = False  
Form_Asistentes.suprimir.Enabled = False  
End Sub
```

```
Private Sub Autobus_AfterUpdate()  
If Me.Estado <> "Confirmado" And Me.Autobus = True Then  
MsgBox "Se debe confirmar primero al asistente antes de indicar el uso del autobus.", vbCritical,  
"Error"  
Me.Autobus = False  
End If  
End Sub
```



```
Private Sub Form_Click()  
    If Me.Form.Recordset.RecordCount = 0 Then Exit Sub  
    Form_Asisntentes.vNombre = Trim(" " & Me.Form![Apellidos, Nombre])  
    Form_Asisntentes.vUsuario = Trim(" " & Me.Form![Us3270])  
    If Me.Form![Estado] = "Convocado" Then  
        Form_Asisntentes.moverconvcand.Enabled = True  
        Form_Asisntentes.movertodosconvcand.Enabled = True  
        Form_Asisntentes.confirmar.Enabled = True  
    Else  
        Form_Asisntentes.moverconvcand.Enabled = False  
        Form_Asisntentes.movertodosconvcand.Enabled = False  
        Form_Asisntentes.confirmar.Enabled = False  
    End If  
    Form_Asisntentes.cancelar.Enabled = True  
    Form_Asisntentes.movertodoscandconv.Enabled = False  
    Form_Asisntentes.movercandconv.Enabled = False  
    Form_Asisntentes.suprimir.Enabled = False  
End Sub  
  
Private Sub Form_Current()  
    Form_Asisntentes.vNombre = ""  
    Form_Asisntentes.vUsuario = ""  
End Sub
```

SubF Periodos

Option Compare Database

```
Private Sub Form_BeforeInsert(Cancel As Integer)  
    CodigoCurso.Value = Left(Forms![Sesiones]!TXT_ClaveCurso.Value, 6)  
    ClaveSesion.Value = Right(Forms![Sesiones]!TXT_ClaveCurso.Value, 4)  
End Sub
```



SubF Otras tareas

Option Compare Database

```
Private Sub Form_BeforeInsert(Cancel As Integer)
    CodigoCurso.Value = Left(Forms![Sesiones]!TXT_ClaveCurso.Value, 6)
    ClaveSesion.Value = Right(Forms![Sesiones]!TXT_ClaveCurso.Value, 4)
End Sub
```

SubF Sesiones

Option Compare Database

```
Private Sub CodigoCurso_Click()
    Forms![BuscarCurso]!CMB_Aceptar.Enabled = False
End Sub
```

```
Private Sub F_Fin_Click()
    Forms![BuscarCurso]!CMB_Aceptar.Enabled = False
End Sub
```

```
Private Sub F_Inicio_Click()
    Forms![BuscarCurso]!CMB_Aceptar.Enabled = False
End Sub
```

```
Private Sub Form_Click()
    On Error GoTo salida
    Form_BuscarCurso.gblCurso = Trim(" " & Me.Form![CodigoCurso])
    Forms![BuscarCurso]!CMB_Aceptar.Enabled = True
    Exit Sub
salida:
End Sub
```

```
Private Sub Form_DblClick(Cancel As Integer)
    On Error GoTo salir
    Form_Sesiones.vCurso = Form_BuscarCurso.gblCurso
    Form_Sesiones.vSesion = Forms![BuscarCurso]![SUBF_Sesiones].Form![Sesión]
    On Error GoTo 0
    Form_Sesiones.SeleccionarCurso
    Form_BuscarCurso.CMB_Salir_Click
    Exit Sub
salir:
End Sub
```

```
Private Sub Lugar_Click()
    Forms![BuscarCurso]!CMB_Aceptar.Enabled = False
End Sub
```

```
Private Sub NombreSesion_Click()
    Forms![BuscarCurso]!CMB_Aceptar.Enabled = False
End Sub
```

```
Private Sub Proveedor_Click()
    Forms![BuscarCurso]!CMB_Aceptar.Enabled = False
End Sub
```



```
Private Sub Sesión_Click()  
    Forms![BuscarCurso]!CMB_Aceptar.Enabled = False  
End Sub
```

1. BIBLIOGRAFÍA

- **MÉTRICA. VERSIÓN 3: Metodología de Planificación, Desarrollo y Mantenimiento de sistemas de información.**
Ministerio de Administraciones Públicas.
<http://www.csi.map.es/csi/metrica3/index.html>
- **Medidas en el marco del modelo CMM.** Medición para la gestión en la Ingeniería del Software.
J.J. Cuadrado, A. de Amescua, A. de Miguel, J. García. (ISBN 84-7897-403-2). *Chapter*. RA-MA. Spain. 2000.
- **Construcción lógica de programas.**
Luis García, Juan José Cuadrado, Antonio de Amescua, Manuel Velasco. (ISBN 84-7897-492-X). *Full Book*. RA-MA. Spain. 2001.
- **Gestión del Proceso Software.**
Cuevas, G.; Amescua, A.; Cerrada, J.A.; San Feliu, T.; Calvo-Manzano, J.; Arcilla, M.; García, M. (ISBN 84-8004-546-9). *Full Book*. Centro de Estudios Ramón Areces. Spain. 2002.
- **Análisis y diseño estructurado y orientado a objetos de sistemas informáticos.**
Antonio de Amescua, Juan J. Cuadrado, Emilio Ernica, Javier García, Luis García, Paloma Martínez, María Isabel Sánchez. (ISBN 84-481-3924-0). *Full Book*. McGraw- Hill. Spain. 2003.