



ugr

Universidad
de Granada

TRABAJO FIN DE GRADO
GRADO EN INGENIERÍA INFORMÁTICA

Sistema de Información Web para la Gestión Administrativa de Cursos de Formación

Autor

Francisco Serrano Carmona

Director

Daniel Sánchez Fernández



Escuela Técnica Superior de Ingenierías Informática y de
Telecomunicación

—
Granada, Julio de 2014

Sistema de Información Web para la Gestión Administrativa de Cursos de Formación

Francisco Serrano Carmona

Palabras Clave: Sistema de Información, Gestión de Cursos, Modelo-Vista-Controlador, Proyecto Web, Java Server Faces, Desarrollo, Fundación Tripartita de Formación en el Empleo, Diseño, Análisis.

Resumen

En este documento se expone el análisis, desarrollo e implementación de un sistema de información basado en tecnologías web para la gestión administrativa de cursos de formación continua: qué metodología se ha seguido para desarrollar el sistema, qué herramientas y tecnologías se han utilizado para afrontar los requisitos del sistema, cómo se han diseñado e implementado los diferentes componentes del sistema, cómo se usa el sistema implementado y qué beneficios ha supuesto el uso del sistema para sus usuarios.

En este trabajo se han explorado diferentes técnicas de análisis, desarrollo e ingeniería de software, que uniéndolas han dado como resultado un sistema robusto y adaptado a las necesidades de quienes van a usarlo. Se han utilizado tecnologías y metodologías eficaces que han ayudado al análisis, desarrollo e implementación del sistema de información, tales como: uso del patrón de arquitectura de software de Modelo-Vista-Controlador[23]; uso de modelos entidad-relación [7] y modelo relacional [6] para el diseño de la base de datos; uso de técnicas de ingeniería del software como desarrollo ágil [3] y programación extrema [4]; uso de Java Server Faces [9] como tecnología para el desarrollo; uso de virtualización para la ejecución del sistema.

También se explica la implementación del proyecto: qué jerarquía de páginas tiene el sistema; qué agrupación de funcionalidades tiene el código del sistema; cómo puede instalarse el sistema en un entorno de producción; cómo se interactúa con el sistema.

Como resultado, se obtiene un sistema que facilita la administración de cursos de formación continua por parte de una empresa que los ofrezca: academias, asesorías o entidades públicas. El sistema facilita la comunicación entre sus actores (tutores, alumnos, empresas que reciben la formación, entidades financieras), con especial atención a la Fundación Tripartita de Formación en el Empleo [24], organismo encargado de validar y financiar los cursos de formación continua para las empresas que necesiten formar a sus trabajadores.

Sistema de Información Web para la Gestión Administrativa de Cursos de Formación

Francisco Serrano Carmona

Keywords: information system, training courses administration, Model-View-Controller, web porject, Java Server Faces, develop, Fundación Tripartita de Formación en el Empleo, design, analysis.

Abstract

In this document it is presented the analysis, development and implementation of an information system which is based on new web technologies to the *continuing professional development* administrative process: what methodology was made use for carrying out the system, what tools and technologies have been used in order to face the system requirements, how the different parts of the system have been designed and implemented, how the implemented system is used and what benefits from the use of the system has been led for its users.

In this report many analysis techniques have been used, development and software engineering, that combined together turned out to be a robust and compliant system to the users needs.

Effective technologies and methodologies have helped with the analysis, development and implementation of the information system, such as: the use of software architecture pattern *Model-View-Controller*; the use of *entity-relationship* models and relational model for the design of the database; the use of software engineering techniques such as agile development and extreme programming; using *Java Server Faces* and technology for development; the use of virtualization to implement the system.

There is also explained the project implementation: the hierarchy of pages in the system, what group of functions have the system's code, how can the system be installed in an environment of production, how it interacts with the system.

As a result, there is a system which provides with ease the administration of training courses by different companies: academies, consulting businesses or public entities. The system make easier the communication between actors (tutors, students, businesses receiving training, financial institutions), with special attention to the *Fundación Tripartita de Formación en el Empleo* (Spanish public entity) which is responsible for validating and finance training courses for businesses which need to train their employees.

Índice de Contenido

Resumen.....	2
Abstract.....	3
Índice de Contenido.....	4
Introducción.....	5
Ámbito del sistema.....	5
Actores implicados en el sistema.....	5
Empresa principal de formación continua.....	5
Fundación Tripartita de Formación en el Empleo.....	6
Empresas proveedoras de formación continua.....	6
Empresas asesorías.....	6
Empresas clientes.....	6
Entidad financiera.....	6
Relación entre actores y funcionamiento del sistema.....	7
Objetivos y metodología.....	9
Metodología de obtención de requerimientos.....	9
Metodología de Ingeniería del Software.....	9
Metodología de Diseño de la Base de Datos.....	10
Objetivos y Requisitos Funcionales.....	11
Tecnologías a utilizar.....	14
Java.....	14
Bibliotecas y frameworks usados en Java.....	15
Software del servidor.....	16
Modelo-vista-controlador.....	17
Diseño del sistema.....	18
Diseño de la base de datos.....	18
Relación de Requisitos Funcionales con Entidades.....	18
Modelo Entidad/Relación.....	20
Modelo Relacional.....	21
Vistas de la Base de Datos.....	28
Diseño del proyecto.....	29
Paquetes del proyecto.....	29
Jerarquía de archivos.....	30
Mapa del sitio.....	33
Conclusión.....	34
Guía de Instalación.....	34
Usando Máquina Virtual.....	34
Configuración de GassFish y PostgreSQL en un servidor.....	35
Manual de Usuario.....	38
Ejemplo de configuración inicial.....	38
Ejemplo de uso paso a paso.....	41
Ayuda integrada.....	49
Uso del sistema en un entorno real.....	50
Bibliografía.....	51

Introducción

En esta sección se expondrá un análisis de los elementos más relevantes del sistema: el ámbito del sistema (a qué está destinado el sistema), actores del sistema (qué entes interaccionarán con el sistema) y las relaciones de los actores en el sistema (cómo se comunica cada actor entre los otros actores para que queden estas comunicaciones reflejadas en el sistema).

Ámbito del sistema

Se pretende diseñar un sistema de información basado en tecnologías web para la gestión administrativa de cursos de Formación Profesional Continua. La Formación Profesional Continua está destinada al colectivo de trabajadores en activo, cuyo objetivo es la adquisición de mayores competencias que permitan una actualización permanente del trabajador al puesto de trabajo que desempeña. Actualmente las empresas pueden disponer de financiación para los cursos de formación continua para sus empleados a través de una fundación estatal española: la Fundación Tripartita para la Formación en el Empleo [24].

Para asesorar y facilitar el acceso a dicha Formación Profesional Continua, existen empresas dedicadas a gestionar los diversos Cursos de Formación Continua a los que pueden acceder los empleados de las empresas (según el puesto que desempeñen), gestionar la comunicación con la Fundación Tripartita para la Formación en el Empleo, la financiación de los cursos y el seguimiento tutorial. Además, estas empresas certifican ante la Fundación Tripartita para la Formación en el Empleo que los empleados de dichas empresas han superado con éxito los Cursos de Formación Continua impartidos. A estas empresas encargadas de gestionar la Formación Profesional Continua está dirigido este proyecto.

Con este proyecto se persigue crear un sistema de información que facilite la gestión y comunicación de la Fundación Tripartita de Formación en el Empleo con la Empresa de Formación Continua; de la Empresa de Formación Continua con sus proveedores; de la Empresa de Formación Continua con sus clientes y asesorías, y con la entidad financiera encargada de gestionar los pagos de los clientes.

Actores implicados en el sistema

A continuación se exponen los actores presentes en el sistema de información. Estos actores han sido identificados tras diversas entrevistas con los representantes de una Empresa de Formación Continua que va a utilizar el sistema de información que se desarrolla en este proyecto (explicado más en profundidad en el punto “Metodología de obtención de requisitos”).

Empresa principal de formación continua

Es el actor principal del sistema. Es la empresa que posee el sistema y la que le da el servicio de asesoramiento y gestión de los Cursos de Formación Continua a las empresas clientes. Dentro de este actor podemos encontrar diferentes perfiles de usuario:

Usuario Administrativo: este usuario es el encargado de administrar el sistema de información en su mayor grado. Habitualmente será el usuario que mayores privilegios tenga dentro del sistema. Se identifica con el personal de administración dentro de la empresa, encargado de dar de alta en el sistema las matriculaciones de nuevas empresas y alumnos, tramitar la facturación, etcétera.

Usuario Comercial: La empresa principal de formación continua también puede tener una plantilla de comerciales. Dichos comerciales se encargarán de contactar con nuevas empresas y ofrecerles a las empresas nuevas ofertas de formación: Cursos de Formación Continua que puedan ser atractivos para sus empleados. Este usuario dentro del sistema habitualmente sólo tendrá acceso a la consulta de empresas y matriculaciones de su propia estructura comercial (las empresas clientes que pertenezcan a su cartera de clientes).

Fundación Tripartita de Formación en el Empleo

La Fundación Tripartita de Formación en el Empleo [24] es también un actor importante, ya que es el organismo encargado de validar que las empresas (clientes de la empresa principal de formación continua) registradas en el sistema existen y tienen derecho a obtener financiación para los Cursos de Formación Continua, debe de validar los cursos que se imparten (Acciones Formativas), debe validar los alumnos a los que se les imparte dicha formación y debe subvencionar el precio del Curso de Formación Continua, si procede según la ley vigente en dicha materia. La comunicación con este actor se establece desde la empresa principal de formación continua hacia dicho actor, y viceversa (no se comunica con otros actores del sistema). La comunicación se hace principalmente a través de la carga de archivos XML (por lo que el sistema de información deberá de gestionar la carga de dichos archivos XML). La especificación de los archivos XML de la Fundación Tripartita de Formación en el Empleo viene descrita en el *Anexo 1*.

Empresas proveedoras de formación continua

Puede haber varios actores de este tipo, o ninguno. La empresa principal de formación continua también puede asumir las propiedades de este actor. Estos actores son los encargados de proporcionarle a la empresa principal de formación continua el material de formación, tanto tangible (libros, plataforma de seguimiento tutorial, plataforma de educación a distancia...) como humano (tutores que se pongan en contacto con los alumnos para realizar tareas de seguimiento tutorial: evaluación, resolución de dudas, etc.). Por lo tanto, obtenemos de este actor un usuario para el sistema:

Usuario Tutor: es la persona encargada de tutorizar a los alumnos designados de cierta Acción Formativa. Deberá de tener contacto con los alumnos y registrar las incidencias que se produzcan en el curso al que los alumnos estén matriculados. Este usuario debe tener acceso al sistema para consultar los datos de sus alumnos y sus matriculaciones, y anotar los datos e incidencias que se produzcan en el curso.

Empresas asesorías

Estos actores son empresas que tienen contacto con un grupo de empresas, pues las empresas de dicho grupo son clientes de las asesorías. Las asesorías les ofrecen diversos servicios a las empresas, y estas asesorías contactan con la empresa principal de formación continua para, a través de la asesoría, ofrecerles los servicios de gestión de Cursos de Formación Continua a las empresas finales. Este actor interactúa en el sistema puesto que es necesario registrar qué empresas tienen relación con qué asesorías, para poder ejercer una labor comercial eficaz con dichos actores desde la empresa principal de formación continua.

Empresas clientes

Las empresas clientes son uno de los actores principales para el sistema. Son las empresas que van a recibir el servicio de Formación Continua para los empleados de la empresa. Estas empresas rellenarán la matrícula para dar de alta a los trabajadores en la acción formativa a realizar (los datos a rellenar de una matrícula pueden encontrarse en el *Anexo 2*). De este actor dependen los alumnos:

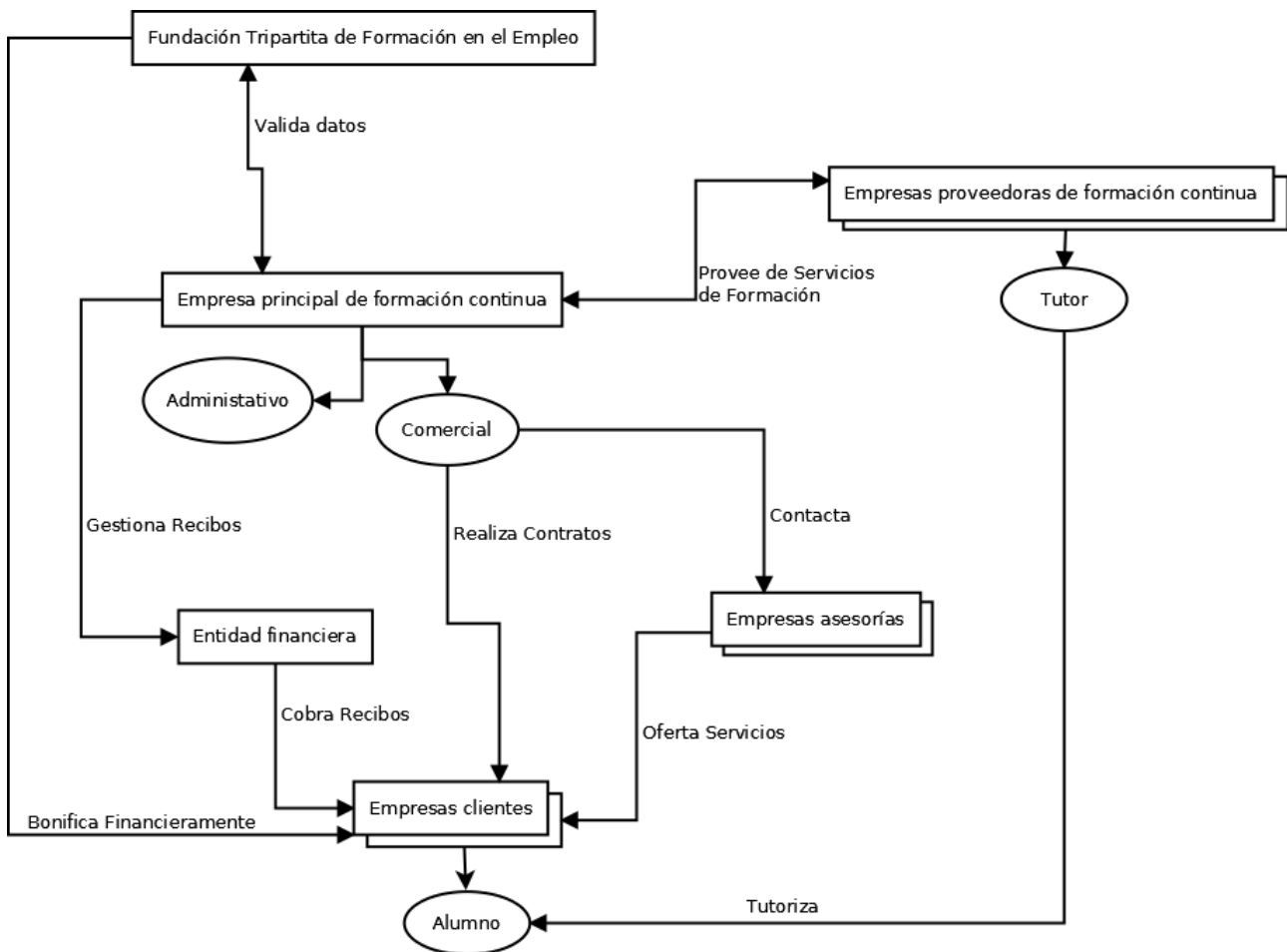
Usuario Alumno: es la persona que recibirá el Curso de Formación Continua: trabajador de la empresa que ha contratado los servicios con la empresa principal de formación continua. Será quien reciba el seguimiento tutorial, y quien reciba una acreditación si supera el curso en el que se ha matriculado. También estará en contacto con los tutores del curso en el que se matricule.

Entidad financiera

La entidad financiera es un actor con el cual se relaciona la empresa principal de formación continua para poder cobrar los recibos generados por contratar sus servicios una empresa cliente. Recibirá los recibos generados para cobrarlos a las cuentas bancarias de las empresas clientes utilizando los archivos con norma 19 (especificado en el *Anexo 3*).

Relación entre actores y funcionamiento del sistema

A continuación se presenta un diagrama de la relación simplificada entre los actores que intervienen en el sistema:



Del diagrama presentado se detallan las siguientes propiedades:

- Los datos a validar en la Fundación Tripartita de Formación en el Empleo son:
 - Acciones Formativas: Informa de las acciones formativas que oferta la empresa principal de formación continua.
 - Inicio de Grupos Formativos: Informa a la Fundación Tripartita de Formación en el Empleo qué Grupos Formativos van a iniciarse. Los Grupos Formativos son agrupaciones de Alumnos que van a cursar una Acción Formativa en un tiempo determinado, con determinados tutores.
 - Finalización de Grupos Formativos: Informa a la Fundación Tripartita de Formación en el Empleo qué Grupos Formativos van a finalizar. Los Grupos Formativos son agrupaciones de Alumnos que van a cursar una Acción Formativa en un tiempo determinado, con determinados tutores. En la finalización van incluidos más datos que en el inicio, como el coste del curso.
 - Datos de Empresas en la Fundación Tripartita: Informa a la empresa principal de formación continua de ciertos datos de las empresas clientes: plantilla de trabajadores, crédito asignado para la empresa y crédito disponible. Para obtener estos datos, las empresas deben de estar dadas de alta en el sistema de la Fundación Tripartita de Formación en el Empleo.
- Una vez validados los datos de los cursos, y según el crédito disponible de las empresas clientes en

la Fundación Tripartita, la Fundación Tripartita bonifica financieramente a las empresas clientes con los gastos del Curso de Formación Profesional cursado por sus trabajadores.

- Los usuarios comerciales (o administrativos) son encargados de matricular a los alumnos, contactando con las empresas y asesorías.
- La comunicación entre la empresa cliente o alumno, y la empresa principal de formación continua se realiza principalmente mediante e-mail, fax o correo ordinario, generando la documentación necesaria para que pueda iniciarse y finalizarse el Curso de Formación Continua. La documentación a generar es la siguiente:
 - Documentación a rellenar por la empresa cliente para la empresa principal de formación continua, antes de que se inicie el curso:
 - Matrículas de los alumnos que harán el Curso de Formación Continua (documento incluido en el **Anexo 2**).
 - Documentación a entregar al cliente desde la empresa principal de formación continua en el inicio del Curso de Formación Continua (documentos incluidos en el **Anexo 4**):
 - Carta al representante legal de la empresa cliente, explicando las obligaciones de ambas partes en el marco legislativo actual.
 - Ficha con los datos del curso en el que se ha matriculado el alumno.
 - Ficha de registro de alumno, con los datos del alumno.
 - Factura o facturas de la matrícula, con los datos de facturación.
 - Certificado de recibí del material, el cual tiene que devolver la empresa cliente para indicar que recibió el material correctamente.
 - Material del Curso de Formación Continua (libro didáctico, material multimedia, formulario de evaluación).
 - Documentación a entregar al cliente desde la empresa principal de formación continua en la finalización del Curso de Formación Continua (documentos incluidos en el **Anexo 5**):
 - Informe de finalización de curso, el cual informa a la empresa cliente de las cantidades a bonificar por parte de la Fundación Tripartita de Formación en el Empleo.
 - Diploma para el alumno, indicando el Curso de Formación Continua que ha finalizado.
 - Justificante de haber recibido el diploma de finalización del Curso de Formación Continua.
 - Además de la documentación indicada, puede generarse más documentación a comunicar a la empresa cliente desde la empresa principal de formación continua, si se dan ciertos requisitos (es un Curso de Formación Continua presencial; existe en la empresa Representante Legal de los Trabajadores; etc.).
- Para que la empresa principal de formación continua pueda tener empresas proveedoras de formación continua, estas empresas deben de facilitarle a la empresa principal:
 - Los datos de las acciones formativas que oferta la empresa proveedora a la empresa principal de formación continua.
 - Los datos de los tutores que pueden tutorizar ciertas acciones formativas que oferta la empresa proveedora.
- Los comerciales y empresas asesorías podrán tener una bonificación por parte de la empresa principal de formación continua, si superan un límite de ventas acordados.

Objetivos y metodología

En este apartado se desarrolla la metodología que se ha seguido para desarrollar el sistema. Se detallará la metodología según 3 perspectivas: metodología de obtención de requisitos, para explicar qué técnicas se han usado para la obtención de requisitos; metodología de Ingeniería del Software, donde se expondrá qué técnicas de Ingeniería del Software se han seguido para desarrollar el proyecto; y metodología de diseño de Base de Datos, donde se expondrá la metodología seguida para diseñar la estructura de la base de datos.

Una vez explicada la metodología, se expondrán los objetivos que tiene que cumplir el sistema, obteniendo de esta manera los requisitos funcionales. Por último en este apartado, se expondrán las tecnologías y técnicas que se utilizarán para la implementación del proyecto.

Metodología de obtención de requerimientos

La obtención de requerimientos consiste en recoger la información de cómo se desea que funcione el sistema en función de los objetivos que debe de cumplir [5]. Para el desarrollo de este proyecto se han usado diversas técnicas de obtención de requerimientos como metodología:

- Entrevistas. Para desarrollar este proyecto, se ha contactado con responsables de una empresa de formación continua, la cual ha ido especificando los requisitos que debía cumplir el sistema para adaptarse a las necesidades de una empresa de este tipo, como actor principal de las interacciones en el sistema. Los requerimientos se han ido obteniendo a través de las entrevistas realizadas, donde se ha ido preguntando: el entorno del sistema, los actores principales, las interacciones entre dichos actores, los procedimientos administrativos que se hacen paso a paso en la empresa, etcétera. Con esta información ha podido hacerse un análisis exhaustivo de los requerimientos que necesita un sistema de estas características. Durante el desarrollo del proyecto ha habido más reuniones con entrevistas: 2 reuniones al mes, de unas 3 horas de duración, para mejorar las características del sistema, usando también las otras técnicas de obtención de requerimientos.
- Desarrollo de Prototipos. La obtención de requerimientos se ha desarrollado con ayuda del desarrollo de prototipos: modelos de la interfaz, más o menos operativos (comenzando con un boceto en papel y terminando con la aplicación desarrollada). Las ventajas de esta metodología de obtención de requerimientos son:
 - Al demostrar las funciones del sistema, se identifican las discrepancias entre desarrolladores y usuarios.
 - Durante el desarrollo del prototipo, se puede contrastar si los requerimientos son inconsistentes y/o están incompletos.
 - Se dispone rápidamente de un sistema que demuestra usabilidad de la aplicación.
 - El prototipo se utiliza como base para escribir la especificación para el desarrollo.
- Estudio de documentos. Analizando los documentos de la empresa de formación (*Anexo 2, Anexo 4, Anexo 5*) se puede conseguir identificar de forma precisa los requisitos de información que necesita el sistema. Por ejemplo: analizando la matrícula del *Anexo 2* se pueden identificar los datos a guardar de una empresa cliente y de un alumno. El uso de estos documentos junto con las entrevistas consigue tener una visión de los datos a guardar por el sistema, qué datos han de ser los más importantes, qué conjuntos de datos forman una unidad y qué unidades dependen unas de otras, etcétera. Además, sabiendo cuándo y cómo se utilizan dichos documentos, se consigue obtener el esquema de comunicación entre los actores, el cual es útil a la hora de desarrollar la funcionalidad del sistema.

Metodología de Ingeniería del Software

Metodología de ingeniería de software es un marco de trabajo usado para estructurar, planificar y controlar el

proceso de desarrollo en sistemas de información. Una metodología de desarrollo de software es usada para estructurar, planear y controlar el proceso de desarrollo en sistemas de información.

En este proyecto se han usado diversas técnicas de Ingeniería del Software como metodología:

- Prototipado: El prototipado permite desarrollar modelos de aplicaciones de software que permiten ver la funcionalidad básica de la misma, sin necesariamente incluir toda la lógica o características del modelo terminado. El prototipado permite al cliente evaluar en forma temprana el producto, e interactuar con los diseñadores y desarrolladores para saber si se está cumpliendo con las expectativas y las funcionalidades acordadas.
- Diseño orientado a objetos [1]: enfoque de la ingeniería de software que modela un sistema como un grupo de objetos que interactúan entre sí. Se aplican técnicas de modelado de objetos para analizar los requerimientos para un contexto - por ejemplo, un sistema de negocio, un conjunto de módulos de software - y para diseñar una solución para mejorar los procesos involucrados [2].
- Desarrollo ágil: El desarrollo ágil de software refiere a métodos de ingeniería del software basados en el desarrollo iterativo e incremental [3]. Los métodos ágiles enfatizan las comunicaciones cara a cara en vez de la documentación técnica. Los métodos ágiles también enfatizan que el software funcional es la primera medida del progreso del desarrollo. Para el desarrollo de este proyecto se han utilizado diversas técnicas del desarrollo ágil, como la filosofía del la programación extrema.
 - La programación extrema es una metodología de desarrollo de la ingeniería de software [4]. Es el más destacado de los procesos ágiles de desarrollo de software. Al igual que éstos, la programación extrema se diferencia de las metodologías tradicionales principalmente en que pone más énfasis en la adaptabilidad que en la previsibilidad. Los defensores de la programación extrema consideran que los cambios de requisitos sobre la marcha son un aspecto natural, inevitable e incluso deseable del desarrollo de proyectos. Creen que ser capaz de adaptarse a los cambios de requisitos en cualquier punto de la vida del proyecto es una aproximación mejor y más realista que intentar definir todos los requisitos al comienzo del proyecto e invertir esfuerzos después en controlar los cambios en los requisitos. Los valores de la programación extrema son: simplicidad, comunicación, retroalimentación y coraje.
 - Simplicidad: Se simplifica el diseño para agilizar el desarrollo y facilitar el mantenimiento.
 - Comunicación: En el desarrollo, el código comunica mejor cuanto más simple sea. Si el código es complejo hay que esforzarse para hacerlo inteligible. El cliente (la empresa de formación en el caso de este proyecto) decide qué características tienen prioridad y debe estar disponible para solucionar dudas.
 - Realimentación: Al realizarse ciclos muy cortos tras los cuales se muestran resultados, se minimiza el tener que rehacer partes que no cumplen con los requisitos y ayuda en el desarrollo a centrarse en lo que es más importante.
 - Coraje: Esto significa revisar el sistema existente y modificarlo, o si con ello los cambios futuros se implementarían más fácilmente. Otro ejemplo de coraje es saber cuando desechar una parte del sistema: quitar código fuente obsoleto, sin importar cuanto esfuerzo y tiempo se invirtió en crear ese código.
 - Aunque no se ha podido aplicar todo el sistema de la programación extrema en su conjunto (roles, desarrollo en parejas, pruebas unitarias...), sí se ha utilizado la filosofía ágil para el desarrollo del proyecto, consiguiendo unos resultados exitosos para los implicados (requisitos ajustados a la realidad de las necesidades, en el tiempo estimado).

Metodología de Diseño de la Base de Datos

Para el diseño de la base de datos se ha utilizado, tras el análisis de los requisitos (explicados en el punto “Objetivos y Requisitos Funcionales”), la creación del modelo entidad-relación extendido que refleje los

requisitos de información del proyecto.

- Un modelo entidad-relación [7] es una herramienta para el modelado de datos que permite representar las entidades relevantes de un sistema de información así como sus interrelaciones y propiedades. El modelo de datos entidad-relación está basado en una percepción del mundo real que consta de una colección de objetos básicos, llamados entidades, y de relaciones entre esos objetos. Una entidad representa una “cosa” u “objeto” del mundo real con existencia independiente, es decir, se diferencia unívocamente de otro objeto o cosa. Los atributos son las propiedades que describen a cada entidad en un conjunto de entidades. Las relaciones son enlaces entre las distintas entidades del modelo, con una cardinalidad establecida.

Una vez conseguido el modelo entidad-relación, se producirá un modelo relacional [6] para generar las relaciones a través del Sistema Gestor de Base de Datos. En este modelo todos los datos son almacenados en relaciones. La información puede ser recuperada o almacenada por medio de consultas que ofrecen una amplia flexibilidad y poder para administrar la información. El modelo relacional considera la base de datos como una colección de relaciones. De manera simple, una relación representa una tabla que no es más que un conjunto de filas, cada fila es un conjunto de campos y cada campo representa un valor que interpretado describe el mundo real. Cada fila también se puede denominar *tupla* o registro y a cada columna también se le puede llamar campo o atributo.

Objetivos y Requisitos Funcionales

Analizando (con las metodologías vistas en “Metodología de obtención de requerimientos”) los requerimientos del sistema, se ha llegado a una serie de objetivos generales que debe de cumplir el sistema, que dan lugar a varios sub-objetivos:

- OBJ. 1. Guardar y administrar en el sistema los datos necesarios de los actores implicados para el uso del sistema. Poder tener un registro de los diferentes actores y datos que intervienen en el sistema.

Para ello se identifican las unidades de datos que entrarán en juego dentro del sistema. Analizando la documentación (**Anexo 2, Anexo 4, Anexo 5**) y las relaciones entre los actores, podemos encontrar los siguientes sub-objetivos:

- OBJ. 1.1. Guardar y gestionar los datos de las empresas implicadas en el sistema: tanto empresas proveedoras de Cursos de Formación Continua, como empresas clientes, como empresas asesorías; incluyendo la empresa de formación principal. Observando el **Anexo 2**, se pueden obtener los datos de las empresas clientes. Los datos de las otras empresas son subconjuntos de atributos de los datos de las empresas clientes.

- OBJ. 1.2. Guardar y gestionar los datos de las personas implicadas en el sistema: tanto contactos dentro de las empresas (Representante Legal de la empresa, comerciales de la empresa), como tutores de las empresas de formación; alumnos de las empresas clientes; comerciales de la empresa de formación principal; etcétera. Observando el **Anexo 2**, se pueden obtener los datos de los alumnos.

- OBJ. 1.3. Guardar y gestionar los datos comerciales de las asesorías y qué contactos tienen con los comerciales: hace falta mantener un registro de qué comerciales tienen contacto con qué empresas. Para ello hace falta mantener una “estructura comercial” que indique de qué comerciales depende cada empresa y asesoría.

- OBJ. 1.4. Guardar y gestionar en el sistema los datos de las acciones formativas que oferta la empresa de formación principal. En el **Anexo 1** vienen registrados los datos a guardar para poder dar de alta acciones formativas en la Fundación Tripartita de Formación en el Empleo. Una acción formativa es un tipo de Curso de Formación Continua, el cual está dirigido a cumplir unos objetivos de formación en una cantidad de horas determinadas.

- OBJ. 1.5. Guardar y gestionar en el sistema los datos de los grupos de formación que se creen

en la empresa de formación principal. En el **Anexo 1** vienen registrados los datos a guardar para poder dar de alta grupos de formación en la Fundación Tripartita de Formación en el Empleo. Un grupo de formación es un conjunto de alumnos matriculados en una acción formativa determinada, con unos tutores determinados que guiarán la acción formativa, en un espacio de tiempo determinado, con un horario de tutorías determinado.

OBJ. 1.6. Guardar y gestionar en el sistema los datos de las matriculaciones que hagan los alumnos en los grupos de los Cursos de Formación Continua. Guardar y gestionar los datos de la documentación remitida por parte de estas matriculaciones. Observando el **Anexo 2**, se pueden obtener los datos de las matrículas a guardar en el sistema. Los datos de la documentación se pueden guardar en forma de eventos asociados a la matrícula, guardando en dichos eventos cuándo se produjo el evento y qué detalles tuvo dicho evento.

OBJ. 1.6.1. Gestionar y visualizar los eventos generados por las matriculaciones, en forma de calendario, para poder visualizar cuánta carga de trabajo se genera y queda pendiente cada día.

OBJ. 2. Generar la documentación necesaria para los actores en el sistema. Generar los flujos de comunicación necesarios para que los actores puedan comunicarse. Para poder cumplir este objetivo, podemos dividirlo en varios sub-objetivos:

OBJ. 2.1. Posibilidad en el sistema de poder enviar correos electrónicos de forma masiva, para poder enviar la documentación necesaria a los actores implicados.

OBJ. 2.2. Posibilidad en el sistema de poder rellenar archivos en formato PDF para enviar la documentación con los datos necesarios.

OBJ. 2.3. Posibilidad en el sistema de poder generar hojas de cálculo para poder analizar diferentes aspectos de los datos del sistema: listado de empresas, listado de personas, listado de matrículas, etcétera.

OBJ. 3. Validar los datos guardados en el sistema frente a la Fundación Tripartita de Formación en el Empleo, para poder asegurar que se está haciendo una gestión correcta de los datos frente al organismo competente. Esto se puede conseguir generando los archivos XML que gestiona la Fundación Tripartita de Formación en el Empleo, por lo tanto se deducen los siguientes sub-objetivos:

OBJ. 3.1. Generar el fichero XML de Acciones Formativas, para poder cargar las acciones formativas en la plataforma de la Fundación Tripartita de Formación en el Empleo, y estas acciones puedan ser validadas.

OBJ. 3.2. Generar el fichero XML de Inicio de Grupos Formativos, para poder cargar los grupos de Cursos de Formación Continua en la plataforma de la Fundación Tripartita de Formación en el Empleo, y estos grupos puedan ser validados.

OBJ. 3.3. Generar el fichero XML de Finalización de Grupos Formativos, para poder cargar los alumnos de los grupos de Cursos de Formación Continua en la plataforma de la Fundación Tripartita de Formación en el Empleo, y estos alumnos puedan ser validados.

OBJ. 4. Ofrecer la posibilidad de que puedan entrar varios usuarios en el sistema, con diferentes permisos. Para ello se contará con un sistema de usuarios y permisos, por lo tanto se deducen los siguientes sub-objetivos:

OBJ. 4.1. Generar usuarios que puedan acceder al sistema, que dependan de personas que estén registradas al sistema. Los usuarios podrán entrar en el sistema por medio de un usuario y una contraseña. Si olvidan la contraseña, la podrán recuperar recibiendo un mensaje de correo electrónico desde el sistema.

OBJ. 4.2. Los usuarios dispondrán de una serie de permisos que les permitirán realizar las

acciones posibles dentro del sistema. Cada permiso debe corresponder a una funcionalidad dentro del sistema. Con los permisos se controlará qué puede hacer cada usuario, dándole seguridad al sistema.

OBJ. 5. Generar la facturación de la empresa de formación en el sistema, facilitando las labores contables.

Para este objetivo se gestionará los siguientes sub-objetivos:

OBJ. 5.1. Guardar y gestionar en el sistema los datos de las facturas y remesas creadas a partir de las matriculaciones para las empresas clientes. En el **Anexo 4** pueden encontrarse los datos a tramitar de la factura, según la documentación estudiada. En el **Anexo 3** se pueden estudiar los datos para generar una remesa. Una factura es un recibo donde se detallan los servicios prestados a la empresa cliente y su precio, que se ofrece al cliente como justificante del pago realizado. Una remesa es un conjunto de facturas que serán cobradas a los clientes mediante transferencia bancaria, a través de una entidad financiera de forma telemática.

OBJ. 5.2. Analizar listados de facturas y remesas, para poder ajustar la contabilidad anual de la empresa de formación principal.

A partir de los objetivos anteriormente descritos, se extraen los **Requisitos Funcionales** del sistema:

R.F. 1. Gestionar Empresas.

R.F. 1.1. Dar de alta Empresas.

R.F. 1.2. Consultar las Empresas del Sistema.

R.F. 1.3. Borrar Empresas del Sistema.

R.F. 1.4. Consultar las Empresas del Sistema que dependan de la Estructura Comercial del usuario conectado.

R.F. 2. Gestionar Personas.

R.F. 2.1. Dar de alta Personas.

R.F. 2.2. Consultar las Personas del Sistema.

R.F. 2.3. Borrar Personas del Sistema.

R.F. 2.4. Consultar las Personas del Sistema que dependan de la Estructura Comercial del usuario conectado.

R.F. 3. Gestionar Acciones Formativas.

R.F. 3.1. Dar de alta Acciones Formativas.

R.F. 3.2. Consultar las Acciones Formativas del Sistema.

R.F. 3.3. Borrar Acciones Formativas del Sistema.

R.F. 4. Gestionar Grupos.

R.F. 4.1. Dar de alta Grupos.

R.F. 4.2. Consultar los Grupos del Sistema.

R.F. 4.3. Borrar Grupos del Sistema.

R.F. 5. Gestionar Matrículas.

R.F. 5.1. Dar de alta Matrículas.

R.F. 5.2. Consultar las Matrículas del Sistema.

R.F. 5.3. Borrar Matrículas del Sistema.

- R.F. 5.4. Consultar las Matrículas del Sistema que dependan de la Estructura Comercial del usuario conectado.
- R.F. 6. Gestionar Usuarios.
 - R.F. 6.1. Dar de alta Usuarios.
 - R.F. 6.2. Consultar los Usuarios del Sistema.
 - R.F. 6.3. Borrar Usuarios del Sistema.
 - R.F. 6.4. Gestionar los Permisos para los Usuarios del Sistema.
 - R.F. 6.5. Proporcionar a los acceso al sistema, según los permisos que tengan activados.
 - R.F. 6.6. Restablecer la contraseña de un usuario, enviando un correo electrónico al usuario.
- R.F. 7. Gestionar Eventos de Matrículas.
 - R.F. 7.1. Dar de alta Eventos.
 - R.F. 7.2. Consultar los Eventos del Sistema.
 - R.F. 7.2.1. Dentro de las Matrículas.
 - R.F. 7.2.2. En formato Calendario.
 - R.F. 7.3. Borrar Eventos del Sistema.
- R.F. 8. Gestionar Documentación.
 - R.F. 8.1. Generar la Documentación de Inicio de Curso.
 - R.F. 8.2. Enviar la Documentación de Inicio de Curso.
 - R.F. 8.3. Generar la Documentación de Final de Curso.
 - R.F. 8.4. Enviar la Documentación de Final de Curso.
 - R.F. 8.5. Enviar comunicaciones arbitrarias a empresas.
- R.F. 9. Gestionar la Contabilidad del Sistema.
 - R.F. 9.1. Gestionar las Facturas del Sistema.
 - R.F. 9.2. Gestionar las Remesas del Sistema.

Tecnologías a utilizar

A continuación se especifican qué tecnologías se usarán en la implementación del proyecto. El lenguaje de programación utilizado para implementar el proyecto ha sido Java, con diversas bibliotecas y recursos para conseguir un sistema de información web estable, robusto, escalable, con posibilidad de funcionar en “la nube” y fácilmente administrable. Para el Sistema Gestor de Base de Datos se ha elegido PostgreSQL, un SGBD libre y robusto. Se ha diseñado para funcionar en una máquina virtual ejecutada sobre VMWare, bajo la distribución de Linux Ubuntu Server. Se ha implementado también usando el patrón de arquitectura software Modelo-Vista-Controlador.

Java

El lenguaje Java se creó con cinco objetivos principales:

- Debería usar el paradigma de la programación orientada a objetos.
- Debería permitir la ejecución de un mismo programa en múltiples sistemas operativos.
- Debería incluir por defecto soporte para trabajo en red.

- Debería diseñarse para ejecutar código en sistemas remotos de forma segura.
- Debería ser fácil de usar y tomar lo mejor de otros lenguajes orientados a objetos.

Una máquina virtual Java es una máquina virtual de proceso nativo, es decir, ejecutable en una plataforma específica, capaz de interpretar y ejecutar instrucciones expresadas en un código binario especial (el bytecode Java), el cual es generado por el compilador del lenguaje Java.

Java Runtime Environment o JRE (Entorno de Ejecución de Java) es un conjunto de utilidades que permite la ejecución de programas Java. En su forma más simple, el entorno en tiempo de ejecución de Java está conformado por una Máquina Virtual de Java o JVM, un conjunto de bibliotecas Java y otros componentes necesarios para que una aplicación escrita en lenguaje Java pueda ser ejecutada. El JRE actúa como un "intermediario" entre el sistema operativo y Java.

Por estas propiedades del lenguaje (Programa Orientado a Objetos, Multiplataforma y altamente extendido entre la comunidad de desarrolladores), se ha decidido que el sistema funcione bajo **Java SE 7**, usando el Entorno de Ejecución de Java (Java Runtime Environment) **1.7.0_60**.

Bibliotecas y frameworks usados en Java

En el desarrollo de software, un framework, es una estructura conceptual y tecnológica de soporte definido, normalmente con módulos de software concretos, que puede servir de base para la organización y desarrollo de software.

Sobre las bibliotecas y frameworks usados en java, para este proyecto se han utilizado las que se detallan a continuación:

- **Java Server Faces (JSF) [8]:** Es un framework [10] para aplicaciones Java basadas en web que simplifica el desarrollo de interfaces de usuario en aplicaciones Java EE [9]. JSF usa JavaServer Pages (JSP) como la tecnología que permite hacer el despliegue de las páginas. Los siguientes objetivos de diseño representan el foco de desarrollo de JSF:
 - Definir un conjunto simple de clases base de Java para componentes de la interfaz de usuario, estado de los componentes y eventos de entrada.
 - Proporcionar un conjunto de componentes para la interfaz de usuario, incluyendo los elementos estándares de HTML para representar un formulario.
 - Definir APIs para la validación de entrada, incluyendo soporte para la validación en el lado del cliente.
 - Especificar un modelo para la internacionalización y localización de la interfaz de usuario.
 - Automatizar la generación de salidas apropiadas para el objetivo del cliente, teniendo en cuenta todos los datos de configuración disponibles del cliente, como versión del navegador.

En el proyecto se ha utilizado la versión de JSF 2.1, desarrollada en 2010.

- **PrimeFaces:** Es una biblioteca de componentes para Java Server Faces (JSF) de código abierto que cuenta con un conjunto de componentes enriquecidos que facilitan la creación de las aplicaciones web [11]. Es una biblioteca muy liviana, todas las decisiones hechas son basadas en mantener a PrimeFaces lo más liviano posible. El proyecto utiliza la versión de PrimeFaces 3.4.2.
- **Java Persistence API (JPA):** Es la API de persistencia desarrollada para la plataforma Java [12]. Sigue la filosofía de Mapeo objeto-relacional [14]. El mapeo objeto-relacional es una técnica de programación para convertir datos entre el sistema de tipos utilizado en un lenguaje de programación orientado a objetos y la utilización de una base de datos relacional, utilizando un motor de persistencia.
JPA posibilita usar Java Persistence Query Language [15], un lenguaje de consulta orientado a objetos. Es usado para hacer consultas contra las entidades almacenadas en una base de datos

relacional. Está inspirado en gran medida por SQL, y sus consultas se asemejan a las consultas SQL en la sintaxis, pero opera con objetos entidad de JPA en lugar de hacerlo directamente con las tablas de la base de datos. Además, posibilita usar los Enterprise JavaBeans (EJB). El objetivo de los EJB es dotar al programador de un modelo que le permita abstraerse de los problemas generales de una aplicación empresarial (conurrencia, transacciones, persistencia, seguridad, etc.) para centrarse en el desarrollo de la lógica de negocio en sí. El hecho de estar basado en componentes permite que éstos sean flexibles y sobre todo reutilizables. En este proyecto se ha usado la implementación EclipseLink de JPA 2.0 [13].

- **Apache POI:** Es una biblioteca [16] que provee a java de funcionalidad para leer y escribir archivos en los formatos de Microsoft Office: Word, PowerPoint y Excel. La biblioteca hace posible leer, modificar y crear archivos de este tipo, necesarios para importar y exportar hojas de cálculo en el proyecto. El proyecto utiliza la versión de Apache POI 3.9.
- **Itext:** Es una biblioteca Open Source para crear y manipular archivos PDF, RTF, y HTML en Java [17]. El soporte de PDF de iText es bastante extensivo, soportando numerosas funcionalidades de manipulación de archivos PDF. En el proyecto esta biblioteca será útil para rellenar los campos de la documentación de la empresa con los datos que se necesiten. El proyecto utiliza la versión de Itext 5.3.5.

Software del servidor

A continuación se detallan las herramientas que se utilizarán en el servidor del proyecto: PostgreSQL como Sistema Gestor de Base de Datos, GlassFish como servidor de aplicaciones web Java, etcétera.

- **GlassFish:** Es un servidor de aplicaciones de software libre que implementa las tecnologías definidas en la plataforma Java y permite ejecutar aplicaciones que siguen esta especificación [21]. Se trata de un dispositivo de software que proporciona servicios de aplicación a los host cliente. EL servidor de aplicaciones generalmente gestiona la mayor parte (o la totalidad) de las funciones de lógica de negocio y de acceso a los datos de la aplicación. El proyecto presentado utiliza la versión de GlassFish 3.1.2.
- **PostgreSQL [19]:** Es un sistema de gestión de bases de datos [18] relacional orientado a objetos basado de Software Libre. Un sistema de gestión de bases de datos (SGBD) es un conjunto de programas que permiten el almacenamiento, modificación y extracción de la información en una base de datos, además de proporcionar herramientas para añadir, borrar, modificar y analizar los datos.
PostgreSQL ha tenido una larga evolución, la cual se inicia en 1982 con el proyecto Ingres en la Universidad de Berkeley. Algunas de sus principales características son, entre otras:
 - Alta concurrencia: Cada usuario obtiene una visión consistente de lo último a lo que se envió al sistema. Esta estrategia es superior al uso de bloqueos por tabla o por filas común en otros SGBD, eliminando la necesidad del uso de bloqueos explícitos.
 - PostgreSQL provee nativamente soporte para una amplia variedad de tipos, como: texto ilimitado, números de precisión arbitraria, etc.

Por todo ello fue elegido como SGBD para este proyecto. La versión utilizada es: PostgreSQL 9.1.11.

- **Ubuntu Server:** Es el sistema operativo en el que está configurado el sistema. Es un sistema operativo basado en Linux y que se distribuye como software libre. Ubuntu Server está orientado a servidores, sin instalar una interfaz gráfica de usuario. En este proyecto se ha utilizado Ubuntu Server 12.04.4 (versión de soporte técnico extendido, la cual garantiza actualizaciones de seguridad hasta Abril de 2017). A éste sistema operativo se le han añadido las siguientes herramientas:
 - IPTables, como uso de cortafuegos y redireccionamiento de paquetes en la red.

- SSH, para la conexión remota al servidor.
- VSFTPD, como servidor FTP seguro para poder mover archivos en el servidor.
- **VMWare:** Es un sistema de virtualización por software. Un sistema virtual por software es un programa que simula un sistema físico con unas características de hardware determinadas. Permite ejecutar varios sistemas dentro de un mismo hardware de manera simultánea, permitiendo así el mayor aprovechamiento de recursos. Sin embargo al ser una capa intermedia entre el sistema físico y el sistema operativo que funciona en el hardware emulado, la velocidad de ejecución de este último es menor, pero en la mayoría de los casos suficiente para usarse en entornos de producción. Esto también permite transportar sistemas creados en VMWare a otras máquinas físicas manteniendo toda la configuración y con una muy sencilla instalación y puesta en marcha. Para este proyecto se ha utilizado VMWare Player 5.0.2.

Modelo–vista–controlador

El modelo–vista–controlador (MVC) es un patrón de arquitectura de software que separa los datos y la lógica de negocio de una aplicación de la interfaz de usuario y el módulo encargado de gestionar los eventos y las comunicaciones [23]. Para ello MVC propone la construcción de tres componentes distintos que son el modelo, la vista y el controlador, es decir, por un lado define componentes para la representación de la información, y por otro lado para la interacción del usuario.

- **Modelo:** Es la representación de la información con la cual el sistema opera, por lo tanto gestiona todos los accesos a dicha información, tanto consultas como actualizaciones, implementando también la lógica de negocio de la aplicación.
- **Vista:** Presenta el 'modelo' (información y lógica de negocio) en un formato adecuado para interactuar (usualmente la interfaz de usuario).
- **Controlador:** Responde a eventos (usualmente acciones del usuario) e invoca peticiones al 'modelo' cuando se hace alguna solicitud sobre la información (por ejemplo, editar un documento o un registro en una base de datos).

En el proyecto se aplica este patrón de arquitectura de software por medio del framework Java Server Faces, el cual posibilita la arquitectura MVC en el proyecto. Según el funcionamiento de JSF, la vista sería la página XHTML donde se muestra la información, la cual se enviará mediante el protocolo HTTP a la computadora cliente (tal y como funcionan las aplicaciones web); el controlador es la clase Java con la que se comunica la vista para interaccionar con los eventos de la vista; y el modelo son las clases EJB soportadas por JPA. El MVC supone una arquitectura de desarrollo de software limpia y sencilla, que hace más sencillo el desarrollo del sistema.

Diseño del sistema

En este apartado se expondrá cómo se ha llevado a cabo el diseño, desarrollo e implementación del sistema. Se expone el diseño seguido para crear la base de datos del sistema y el diseño del código fuente e implementación del sistema.

Diseño de la base de datos

Se procede al Diseño de la Base de Datos del proyecto. Como se ha explicado en el punto “Metodología de Diseño de la Base de Datos”, el diseño se desarrollará haciendo uso de un Modelo Entidad/Relación y un Modelo Relacional, que servirán para implementar mediante código SQL la base de datos en el sistema.

Relación de Requisitos Funcionales con Entidades

En primer lugar se deben de identificar las entidades y sus relaciones. Para ello se van a utilizar los Requisitos Funcionales del punto “Objetivos y Requisitos Funcionales”:

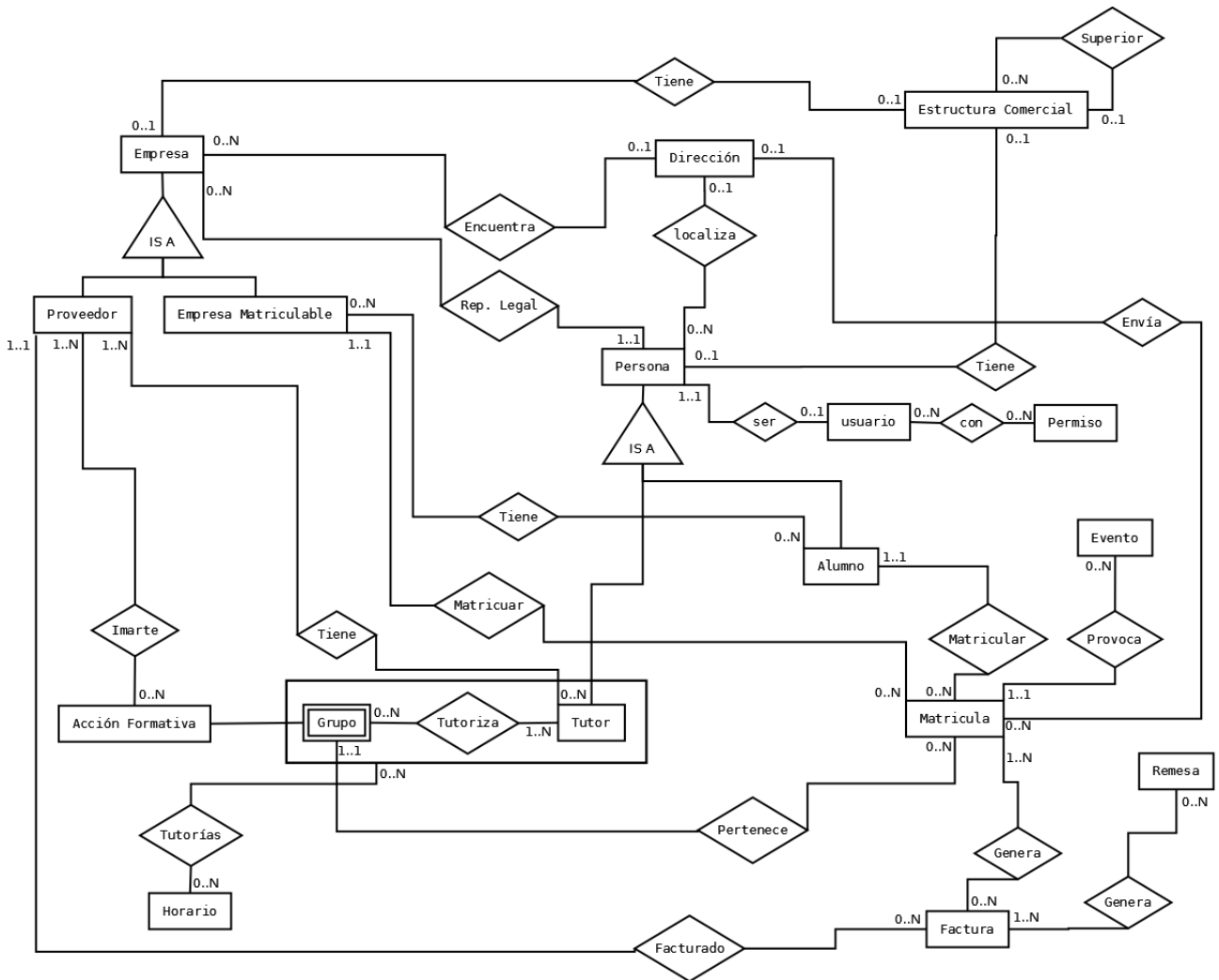
- Se necesita una entidad “Empresa” para preservar los Requisitos Funcionales RF1, RF1.1, RF1.2 y RF1.3.
 - De esta entidad deberán de heredar la entidad “Proveedor” y la entidad “Empresa Matriculable” (empresa cliente), puesto que son entidades que comparten funciones y atributos, pero están especializadas. Esta herencia es solapare (una Empresa puede ser Proveedor y Empresa Matriculable) y no exclusiva (una Empresa no tiene porqué ser Proveedor o Empresa Matriculable). Los atributos de estas entidades han de deducirse a partir de la documentación del *Anexo 2*.
- Se necesita una entidad “Persona” para preservar los Requisitos Funcionales RF2, RF2.1, RF2.2 y RF2.3.
 - De esta entidad deberán de heredar la entidad “Tutor” y la entidad “Alumno”, puesto que son entidades que comparten funciones y atributos, pero están especializadas. Esta herencia es solapare (una Persona puede ser Tutor y Alumno) y no exclusiva (una Empresa no tiene porqué ser Tutor o Alumno). Los atributos de estas entidades pueden de deducirse a partir de la documentación del *Anexo 2*.
- Para cumplir los Requisitos Funcionales RF1.4, RF2.4 y RF5.4, se necesita una entidad “Estructura Comercial”, la cual esté auto-referenciada para posibilitar crear estructuras comerciales jerárquicas (una Persona de la que dependa una Empresa, o viceversa, jerárquicamente).
- Se necesita una entidad “Acción Formativa” para preservar los Requisitos Funcionales RF3, RF3.1, RF3.2 y RF3.3. Esta entidad recogerá todos los atributos de la Acción Formativa, que pueden analizarse a partir del *Anexo 1*.
- Se necesita una entidad “Grupo” para preservar los Requisitos Funcionales RF4, RF4.1, RF4.2 y RF4.3. Esta entidad recogerá todos los atributos del Grupo, que pueden analizarse a partir del *Anexo 1*. Además, “Grupo” será una entidad débil de la Acción Formativa, puesto que es necesario que un Grupo pertenezca a una Acción Formativa, y no tiene sentido que exista un Grupo sin Acción Formativa.
- Para cumplir los Requisitos Funcionales RF5.1, RF5.2 y RF5.3, se necesita una entidad “Matrícula”. Esta entidad estará relacionada con las entidades “Alumno”, “Empresa Matriculable” y “Grupo”. Esta entidad recogerá todos los atributos de la Matrícula, que pueden analizarse a partir del *Anexo 2*.
- Los requisitos RF6, RF6.1, RF6.2, RF6.3, RF6.4, RF6.5 y RF6.6, se integrarán en la base de datos usando las entidades “Usuario” y “Permisos”. La entidad “Usuario” está relacionada con una persona, puesto que para que exista un Usuario debe de estar relacionado con una Persona en el

sistema. Además, entrará en relación con la entidad “Permiso”, la cual guardará los permisos que pueden darse en el sistema, y qué permisos tiene disponible cada usuario.

- Para integrar los requisitos RF7 (y sub-requisitos) y RF8 (y subrequisitos), existe la entidad “Evento”, la cual está relacionada con la entidad “Matrícula” (y con la entidad Persona, para saber quién ha creado el evento). Un evento dependerá de una Matrícula, y una Matrícula podrá tener varios Eventos. Con los eventos se controlará el historial de envío de documentación para las empresas clientes y alumnos, además de controlar otras comunicaciones del sistema con tutores y alumnos.
- Para integrar los requisitos RF9 (y sub-requisitos), se usarán las entidades “Factura” y “Remesa”. La Factura tendrá los atributos especificados en el documento de Factura, dentro del **Anexo 4**. La Remesa tendrá los atributos especificados en el documento de creación de Remesas según el Cuaderno 19.14, dentro del **Anexo 3**. Las facturas están asociadas a las Matrículas, puesto que de las Matrículas se generan las facturas para realizar la contabilidad de facturación de la empresa de formación principal.
- Existe la entidad “Dirección” para poder tener un objeto que guarde las direcciones en un formato determinado. Esta entidad Dirección está relacionada con las entidades Empresa (dirección fiscal de la empresa), Persona (domicilio de la persona) y Matrícula (dirección de envío).
- Cuando un Tutor tutoriza un Grupo, además de crear una relación entre ambas entidades, estas tutorías tendrán un Horario. Para identificar el horario de las tutorías se usará una agregación de la relación entre Tutor y Grupo, identificando qué horario de tutorías tiene dicho grupo para dicho tutor.

Modelo Entidad/Relación

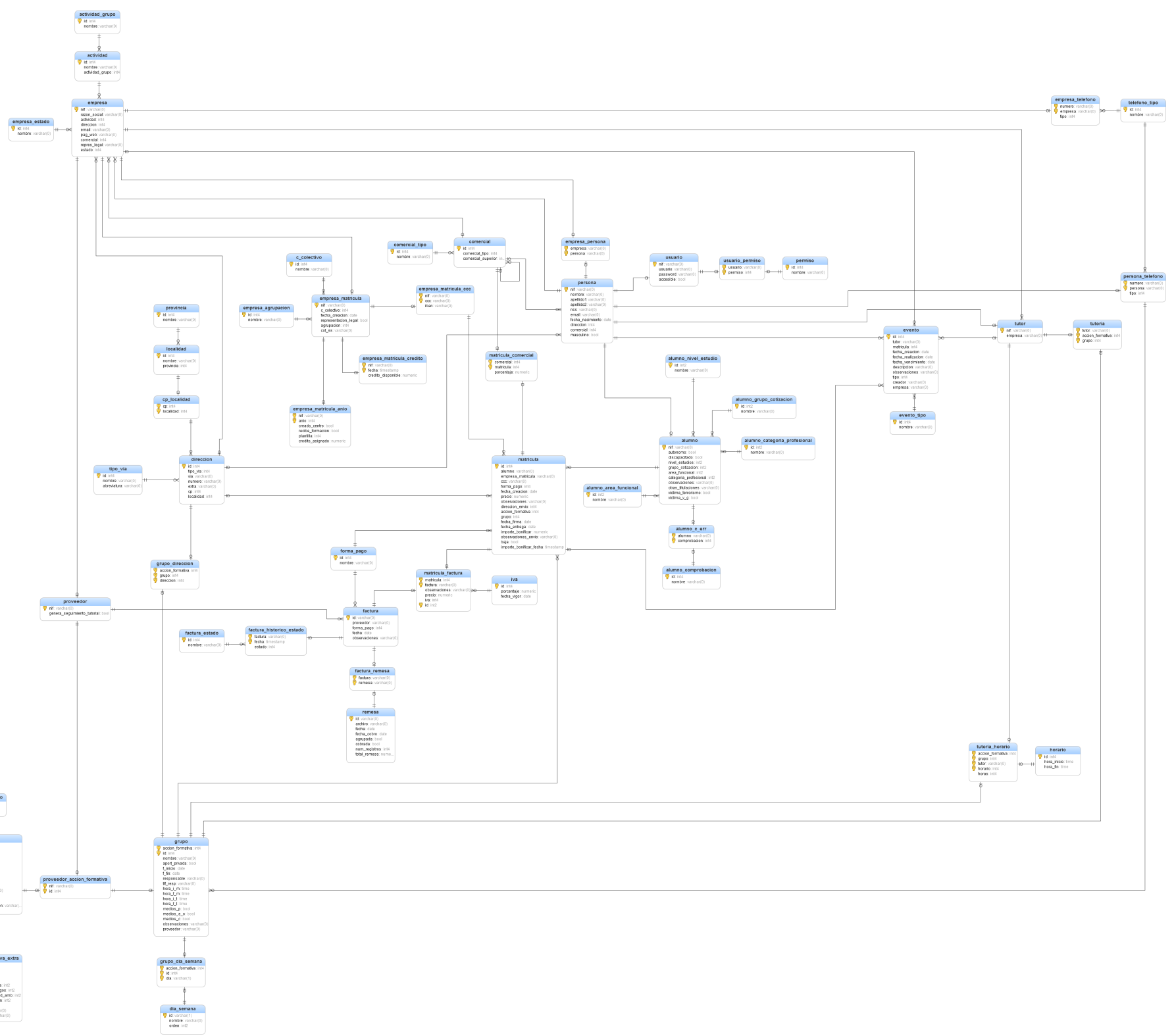
Una vez analizados los requisitos en el punto anterior, se crea el Modelo Entidad/Relación. A continuación se presenta el Modelo Entidad/Relación, simplificado para que sea más sencillo de interpretar conceptualmente (la simplificación ha consistido en no mostrar los atributos de las entidades y ocultar ciertas entidades que dependen de entidades de mayor importancia conceptual; por ejemplo: no se muestran las entidades que dependen de la dirección, como Provincia, Localidad o Código Postal).



Modelo Relacional

Tras obtener el modelo Entidad/Relación simplificado, se desarrolla dicho modelo hasta obtener el Modelo Relacional. El Modelo Relacional es un modelo que muestra las tablas (relaciones) y referencias (claves externas) que se implementará en la estructura de la base de datos, por lo que es una conceptualización del modelo físico de la base de datos. Se ha creado aplicando las Formas Normales [20] al Modelo Relacional extraído del modelo Entidad/Relación, hasta conseguir la Tercera Forma Normal en el diseño de la base de datos. Además, se le han añadido los atributos necesarios para cumplir todos los requisitos de información necesarios para el sistema. El Modelo Relacional resultante es el siguiente:

*El modelo se presenta en la página siguiente, de forma apaisada. Se dispone de una imagen del modelo de mayor calidad en la documentación adjunta del proyecto:
diagramas/modelo_relacional.png*



A continuación se presentan las relaciones más importantes que aparecen en el Modelo Relacional, comentando sus atributos:

- Empresa: Entidad Empresa del Modelo Entidad/Relación. Se compone de los siguientes atributos:
 - NIF: Número de Identificación Fiscal, es la clave primaria de la relación.
 - Razón Social de la empresa.
 - Actividad a la que se dedica la empresa. Esta actividad forma una entidad nueva, “Actividad”. Las actividades se agrupan dentro de “Grupos de Actividad”; por ello existe la entidad “Grupo Actividad”.
 - Dirección: Referencia a la entidad Dirección de la Empresa.
 - Correo Electrónico de la Empresa.
 - Página Web de la Empresa.
 - Comercial: Referencia a la Estructura Comercial que pertenece la Empresa.
 - Persona Representante Legal de la Empresa.
 - Estado de la Empresa: Es una entidad nueva: “Empresa Estado”, que identifica en qué estado se encuentra la empresa (propiedad para la lógica de negocio, arbitraria para el sistema).
- Proveedor: Entidad que hereda de Empresa. Representa a las empresas proveedoras de cursos de formación.
 - Genera Seguimiento Tutorial: indica si con éste proveedor se generan las comunicaciones de los tutores de Seguimiento Tutorial. Si este valor es “Verdadero”, al generar matrículas dependientes de este proveedor, se generarán eventos para comunicar al cliente cómo está siguiendo el curso.
- Empresa Matrícula: Entidad que hereda de Empresa. Representa a las empresas clientes.
 - Convenio Colectivo: Entidad dependiente de Empresa Matrícula, la cual guarda los diferentes Convenios Colectivos que existen y se pueden asociar a la Empresa.
 - Fecha de Creación de la Empresa: Fecha en la que la empresa fue creada.
 - Existencia de Representante Legal de los Trabajadores.
 - Agrupación a la que pertenece la Empresa Matrícula: las Agrupaciones son divisiones lógicas de empresas dentro de la Fundación Tripartita de Formación en el Empleo. Cada Agrupación tiene un máximo de empresas que puede guardar, por lo que este sistema ha de guardar en qué agrupación se ubica la actual empresa.
 - Número de la Seguridad Social de la Empresa.
 - Valores por Año: esta entidad, dependiente de Empresa Matrícula, guarda ciertos valores que van cambiando cada año en la empresa:
 - Año al que se refiere la entidad.
 - Se ha creado un nuevo centro este año en la empresa.
 - Ha recibido formación este año la empresa.
 - Número de trabajadores en la Empresa (plantilla).
 - Crédito asignado por la Fundación Tripartita de Formación en el Empleo para esta Empresa.
 - Crédito Disponible: el crédito disponible puede ir variando a lo largo del año según las empresas van consumiendo crédito en Cursos de Formación Continua. Esta entidad guarda el crédito

disponible en una fecha concreta, para disponer de un histórico de créditos disponibles para la empresa.

- Código de Cuenta Corriente: Guarda los datos bancarios de la Empresa. Una empresa puede tener varias cuentas bancarias, por eso es una relación de uno a muchos en la entidad Empresa Matrícula.
- Persona: Entidad Persona del Modelo Entidad/Relación. Se compone de los siguientes atributos:
 - NIF: Número de Identificación Fiscal de la persona, es la clave primaria de la relación.
 - Nombre y Apellidos de la persona.
 - Número de Identificación de la Seguridad Social de la persona.
 - Correo electrónico de la persona.
 - Fecha de nacimiento de la persona.
 - Entidad dirección asociada a la persona.
 - Comercial: Referencia a la Estructura Comercial que pertenece la persona.
 - Género de la persona (Masculino/Femenino).
 - La persona puede pertenecer a varias empresas, y una empresa puede disponer de varias personas.
- Tutor: Entidad que hereda de Persona. Representa a los tutores del sistema. Los tutores deben de estar asociados a una empresa proveedora de formación.
- Alumno: Entidad que hereda de Persona. Representa a los alumnos del sistema. Los alumnos deben de estar asociados a una empresa matriculable. Sus atributos son;
 - Es o no es autónomo.
 - Es o no es discapacitado.
 - Nivel de estudios del alumno: Entidad para poder elegir entre los diversos Niveles de Estudio que puede tener un alumno.
 - Grupo de Cotización: Entidad para elegir el Grupo de Cotización al que pertenece el alumno.
 - Área Funcional: Entidad para elegir el Área Funcional al que pertenece el alumno.
 - Categoría Profesional: Entidad para elegir la Categoría Profesional al que pertenece el alumno.
 - Observaciones: Observaciones sobre el Alumno.
 - Otras Titulaciones: texto para poder escribir si el alumno tiene otras titulaciones no descritas en el sistema.
 - Es o no es víctima de terrorismo.
 - Es o no es víctima de violencia de género.
 - Comprobación de errores: Es posible que ciertos valores introducidos en el Alumno no sean correctos, pero se hayan introducido por no saber en ese momento si esta información es correcta. Para esto existe esta relación: si se marcan valores como erróneos, quedarán marcados como aviso administrativo para anotar que dichos datos pueden no ser correctos y deben ser revisados.

- Comercial: Entidad que refleja la Estructura Comercial de una Empresa o Persona. Guarda los siguientes atributos:
 - Tipo de Comercial: Tipo de “ente” dentro de la estructura comercial: persona comercial, asesoría, distribuidor, etc.
 - Comercial Superior: Guarda el nodo superior dentro de la jerarquía de la Estructura Comercial.
- Acción Formativa: Relación de la entidad “Acción Formativa”. Está compuesta por los siguientes atributos:
 - ID: identificador de la Acción Formativa.
 - Nombre: nombre de la Acción Formativa.
 - Horas: número de horas que dura la Acción Formativa.
 - Tipo de Acción formativa: Específica o General.
 - Tipo de Nivel de Formación para cursarla: Superior o Básica.
 - Objetivos a completar por la Acción Formativa.
 - Contenidos de la Acción Formativa.
 - Observaciones de la Acción Formativa.
 - Grupo al que pertenece la Acción Formativa: las Acciones Formativas se dividen en Grupos, para identificar las distintas ramas de formación. Esta entidad se referencia en la Acción Formativa para identificar el Grupo al que pertenece dicha Acción Formativa.
 - Modalidad de la Acción Formativa: Existen varias modalidades de Acción Formativa: Presencial, A Distancia, Teleformación y Mixta. Según qué modalidad se elija, existen ciertos datos que se deben rellenar. Esta relación guarda la modalidad elegida para la Acción Formativa.
 - Proveedores de la Acción Formativa: La acción Formativa puede ser impartida por uno o varios proveedores. Esta relación guarda qué proveedores imparten la Acción Formativa.
 - Datos Extra de la acción formativa: esta es una entidad opcional, de 1 a 1, la cual guarda los datos necesarios según la modalidad de la Acción Formativa elegida.
- Grupo: Relación de la entidad débil “Grupo”, dependiente de la entidad fuerte “Acción Formativa”. Tiene los siguientes atributos:
 - ID: identificador del grupo (junto con el identificador de la Acción Formativa).
 - Nombre: Nombre del Grupo.
 - Recibe o no recibe aportación financiera privada.
 - Fecha en la que el Grupo comienza la docencia.
 - Fecha en la que el Grupo finaliza la docencia.
 - Persona de la empresa proveedora de formación principal responsable administrativamente del Grupo.
 - Teléfono de la persona de la empresa proveedora de formación principal responsable administrativamente del Grupo.
 - Hora de Inicio del curso en horario de Mañana.
 - Hora de Fin del curso en horario de Mañana.

- Hora de Inicio del curso en horario de Tarde.
- Hora de Fin del curso en horario de Tarde.
- Grupo financiado con Medios Propios.
- Grupo Financiado con Medios de la Empresa Organizadora.
- Grupo Financiado con Medios de Colaboradores.
- Observaciones del Grupo.
- Proveedor asociado al Grupo (las Acciones Formativas pueden tener más de un proveedor asociado; el proveedor del Grupo debe de ser único).
- Días de la semana en las que se imparte la docencia del Grupo.
- Matrícula: Relación de la entidad “Matrícula” del Esquema Entidad/Relación. Se compone de los siguientes atributos:
 - Alumno asociado a la Matrícula.
 - Empresa (matriculable) asociada a la Matrícula.
 - Cuenta bancaria donde cobrar la Matrícula.
 - Forma de pago de la Matrícula (relación a la entidad “Forma de Pago”, la cual guarda las Formas de Pago aceptadas en el sistema).
 - Fecha de Creación de la Matrícula.
 - Precio de la Matrícula.
 - Observaciones de la Matrícula.
 - Dirección de envío de la Matrícula.
 - Grupo asociado a la Matrícula.
 - Fecha de firma de la Matrícula.
 - Fecha de entrega de la Matrícula.
 - Importe a bonificar de la Matrícula: importe de la Empresa Matriculable que aún puede bonificar en la Fundación Tripartita de Formación en el Empleo.
 - Observaciones para el envío de la Matrícula.
 - Guarda si la matrícula está de baja.
 - Fecha en la que se ha generado el Importe a Bonificar.
 - “Ente” de la Estructura Comercial asociado a la Matrícula.
 - Factura o Facturas asociadas a la Matrícula.
- Usuario: Relación de la entidad “Usuario” del Esquema Entidad/Relación. Se compone de los siguientes atributos:
 - Persona asociada al Usuario.
 - Nombre de Usuario del Usuario.
 - Contraseña del Usuario (encriptada usando el algoritmo de encriptación MD5).
 - Guarda si el usuario es o no es accesible al sistema.

- Relación con la entidad Permiso, la cual guarda los Permisos a los que tiene acceso el usuario.
- Evento: Relación de la entidad “Evento” del Esquema Entidad/Relación. Se compone de los siguientes atributos:
 - Identificador del Evento
 - Tutor asociado al evento (si lo hubiera).
 - Matrícula asociada al evento.
 - Fecha de Creación del evento.
 - Fecha de Realización del evento.
 - Fecha de Vencimiento del evento.
 - Descripción del evento.
 - Observaciones del evento.
 - Tipo de evento (relación a la entidad Tipo de Evento, la cual guarda los tipos de eventos registrados en la plataforma).
 - Persona que ha creado el evento.
- Factura: Relación de la entidad “Factura” del Esquema Entidad/Relación. Se compone de los siguientes atributos:
 - Identificador de la Factura
 - Proveedor de la Factura (normalmente la empresa proveedora de formación principal).
 - Forma de Pago de la factura (relación a la entidad “Forma de Pago”, la cual guarda las Formas de Pago aceptadas en el sistema).
 - Fecha de creación de la factura.
 - Observaciones de la Factura.
 - Histórico de estados de la factura: Relación que guarda un histórico, con la fecha y hora del cambio, de los estados de la factura. Los estados de la factura es una entidad que guarda los estados aceptados por el sistema.
 - Remesa o Remesas asociadas a la factura.
- Remesa: Relación de la entidad “Remesa” del Esquema Entidad/Relación. Se compone de los siguientes atributos:
 - Identificador de la remesa.
 - Archivo asociado a la remesa.
 - Fecha de creación de la remesa.
 - Fecha de cobro de la remesa.
 - Es o no es una remesa agrupada de varias matrículas.
 - Número de registros (número de filas de la remesa).
 - Importe total de la remesa.
- Dirección: Relación de la entidad “Dirección” del Esquema Entidad/Relación. Se compone de los siguientes atributos:

- Identificador de la dirección.
- Tipo de vía de la dirección (relación a la entidad “Tipo de Vía”, la cual guarda los Tipos de Vía aceptados en el sistema).
- Vía de la dirección.
- Número de la dirección.
- Datos extra de la dirección (piso, escalera...).
- Código Postal de la Dirección y Localidad de la Dirección: Relación a la entidad “Código Postal y Localidad”, la cual guarda los Códigos Postales y Localidades existentes en el sistema. También guarda las Provincias a las que pertenecen las Localidades.

Vistas de la Base de Datos

Además de las relaciones vistas anteriormente, el sistema incluye vistas para facilitar la persistencia y optimizar la visualización de contenidos. A continuación se enumeran las vistas del sistema, y sus propiedades:

- Vista de Acción Formativa: Vista que muestra los atributos más relevantes de las Acciones Formativas, para poder hacer búsquedas rápidas filtrando dicha vista.
- Vista de Alumnos: Vista que muestra los alumnos del sistema con su nombre y apellidos para poder seleccionarlos en el sistema de forma ligera.
- Vista de Códigos Postales, Localidades y Provincias: Vista que muestra las tres entidades unidas, facilitando la búsqueda y el filtrado a la hora de buscar direcciones.
- Vista de Empresa Matriculable y Código de Cuenta Corriente: Vista que muestra los Códigos de Cuenta Corriente de todas las empresas para agilizar la búsqueda por estos criterios.
- Vista de empresas: Vista que muestra los atributos más relevantes de las Empresas, para poder hacer búsquedas rápidas filtrando dicha vista.
- Vista de facturas: Vista que muestra los atributos más relevantes de las Facturas, para poder hacer búsquedas rápidas filtrando dicha vista.
- Vista de grupos: Vista que muestra los atributos más relevantes de los Grupos, para poder hacer búsquedas rápidas filtrando dicha vista.
- Vista de matrículas y facturas: Vista que muestra las facturas existentes en el sistema y sus facturas asociadas para facilitar labores de facturación.
- Vista de matrículas: Vista que muestra los atributos más relevantes de las Matrículas, para poder hacer búsquedas rápidas filtrando dicha vista. Esta vista incluye gran cantidad de atributos compuestos para facilitar labores administrativas, como localizar tipos de eventos concretos o conjuntos de estados de diversos atributos de las matrículas y sus entidades relacionadas.
- Vista de Personas y Usuarios: vista que muestra los Usuarios y sus Personas asociadas, para hacer más eficiente la búsqueda y creación de usuarios.
- Vista de personas: Vista que muestra los atributos más relevantes de las Personas, para poder hacer búsquedas rápidas filtrando dicha vista.

Diseño del proyecto

En esta sección se detalla el diseño usado para desarrollar el código fuente e implementar el sistema. Está dividido en 3 partes: paquetes del proyecto, donde se indica qué paquetes de clases Java existen en el proyecto, sus contenidos, funcionalidades y agrupamientos; jerarquía de archivos, donde se indica cómo están organizadas las carpetas y archivos del sistema web; y mapa del sitio, donde se detalla a qué vistas se puede llegar una vez se entra al sistema.

Paquetes del proyecto

A continuación se detallan los paquetes de clases Java existentes en el proyecto. Un Paquete en Java es un contenedor de clases que permite agrupar las distintas partes de un programa cuya funcionalidad tienen elementos comunes. Se explicará el contenido del paquete, las funcionalidades de las clases del paquete y el por qué de dicho agrupamiento:

- **Paquete de Beans de Fachada de Modelos de Entidades:** Este paquete incluye las clases de control de acceso a la base de datos (consulta, edición, borrado) de los Modelos de las Entidades. En este paquete se encuentran las clases que dan la funcionalidad de recuperar, editar y borrar de la base de datos las distintas entidades que existen. Las clases son “Beans” de Fachada: Un Bean es un componente software que es reutilizable y evita reescribir el código para replicar la funcionalidad de un componente. “Fachada” hace referencia a que se ha usado el patrón de diseño “Fachada” para diseñar las clases de este paquete: el patrón de diseño “Fachada” significa que encapsula las llamadas a distintas clases (en este caso, llamadas a la biblioteca JPA) para que sea sencillo utilizar esta clase desde otras clases como interfaz.
- **Paquete de Beans de Fachada de Modelos de Vistas:** Igual que el Paquete de Beans de Fachada de Modelos de Entidades, pero referido a las vistas de la base de datos.
- **Paquete de Modelos de Entidades:** Este paquete guarda las clases EJB de los Modelos de las Entidades de la base de datos. Siguiendo el patrón de arquitectura software MVC, este paquete guarda las clases que están asociadas a las relaciones de la base de datos. Estas clases también incluyen el código de lógica de negocio para las entidades que lo necesitan. Es el paquete que incluye las “estructuras” de los objetos de la base de datos, para que puedan ser usadas en los controladores y las vistas.
- **Paquete de Modelos de Vistas:** Igual que el Paquete de Modelos de Entidades, pero referido a las vistas de la base de datos. Las vistas se guardan como objetos en el sistema, siendo afectadas así por el patrón MVC como si de entidades se trataran (pero sólo son entidades de consulta; no se permite editar ni borrar estos objetos).
- **Paquete de Filtros de Recepción HTTP:** Paquete que guarda las clases de tipo Filtro HTTP de Java. Un filtro HTTP es una clase que se ejecuta cada vez que llega una petición HTTP, y se procesa siempre que es pedida una página del sistema. Este paquete contiene la clase de procesamiento de permisos del sistema, evaluando si el usuario conectado tiene permisos para acceder a la página de la que hace la petición HTTP. Si el usuario no tiene permisos para ver dicha página se le muestra un mensaje de error. Esto hace la gestión de seguridad más sencilla y robusta para el sistema.
- **Paquete de Controladores de Gestión de Entidades:** En el patrón de arquitectura software MVC, este paquete guarda las clases de Controladores de las Vistas que interaccionan con los Modelos del sistema. Guarda las clases que tienen implementada la funcionalidad de la interfaz del sistema para las entidades que son accesibles para su gestión. Por ejemplo: permiten crear, editar y borrar (llamando a las clases del Paquete de Beans de Fachada de Modelos de Entidades) Personas, Empresas, Grupos, etcétera. En estas clases están implementadas las llamadas a las clases que cumplen los requisitos funcionales del sistema.

- Paquete de Controladores de Vistas: Igual que el Paquete de Controladores de Gestión de Entidades, pero referido a las vistas de la aplicación: no incorporan la gestión de entidades, si no muestran los listados de entidades y permiten acceder a funcionalidades de gestión masiva de entidades, como la generación de documentos o el envío de correos electrónicos a empresas.
- Paquete de Utilidades: Este paquete incluye clases de utilidades diversas para el funcionamiento del sistema. Incluye diversos tipos de clases:
 - Clases de estructuras de datos que ofrecen la funcionalidad de generar estructuras de datos complejas para ciertos requisitos de las vistas.
 - Clases de comunicación y generación de documentos:
 - Generación de documentación PDF.
 - Generación de hojas de cálculo de listados.
 - Importación y Exportación de archivos XML.
 - Envío de correos electrónicos.
 - Compresión de archivos.
 - Clases de validación de tipos de datos: Validación de DNI, Validación de correos electrónicos, Validación de Códigos de Cuenta Corriente, etcétera.
 - Clases que contienen variables para el funcionamiento del sistema, como la ruta absoluta en el sistema de ficheros del servidor o los datos de acceso al servidor de correo electrónico para hacer posible el envío de correos electrónicos.

Jerarquía de archivos

Los archivos del sistema están jerarquizados para un acceso sencillo y robusto que posibilite un acceso ordenado a los distintos recursos mediante peticiones HTTP. A continuación se indica la jerarquía usada en el sistema:

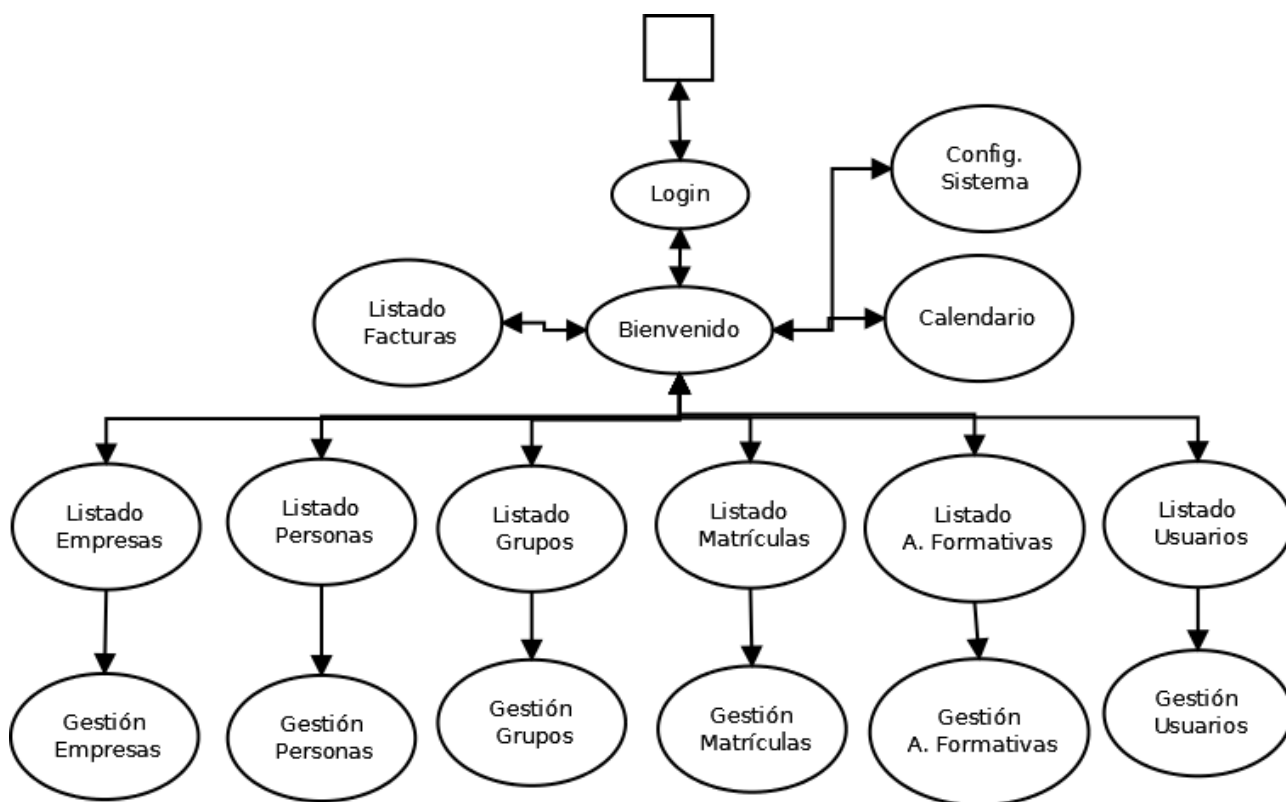
- **docs**: Carpeta de archivos que guarda documentos que no son vistas XHTML
 - **plantillas**: Carpeta que guarda las plantillas de los archivos PDF destinados a generar la documentación del sistema.
 - **temas**: Carpeta que guarda, para cada usuario, información sobre el tema visual utilizado (la biblioteca PrimeFaces soporta el uso de temas, ofreciendo diferentes estilos para que los usuarios puedan escoger).
 - **datoscal**: Carpeta que guarda, para cada usuario, información acerca de sus preferencias para el visionado de la vista de calendario: eventos a mostrar, forma de ordenarlos, etcétera.
 - **temp**: Carpeta que guarda los archivos generados de forma temporal, como la documentación que se genera para que los usuarios la descarguen. Esta carpeta debería de ser limpiada en el sistema cada cierto tiempo.
- **img**: Carpeta que guarda los recursos de imágenes usados en el sistema.
- **resources/css**: Carpeta que guarda las hojas de estilo en cascada (CSS) usadas en el sistema.
- **template.xhtml**: Vista XHTML que guarda la distribución de los elementos principales del sistema (menú principal, botón de desconexión del sistema, pie de página, etcétera).
- **JSF/vista**: Carpeta que guarda las vistas en XHTML para navegar por el sistema. En la arquitectura MVC, esta carpeta y subcarpetas guardan las Vistas de la arquitectura, es decir: los archivos de interfaz con los que interactúa el usuario.

- **login.xhtml**: Vista que ofrece la interfaz para poder conectarse, con usuario y contraseña, al sistema. También incluye la funcionalidad recuperar una nueva contraseña por correo electrónico.
- **bienvenido.xhtml**: Vista que ofrece la interfaz de bienvenida una vez que se ha entrado al sistema. Muestra las opciones disponibles para el usuario que ha accedido a través del menú de opciones.
- **sin_permiso.xhtml**: Vista utilizada cuando se intenta entrar en una página en la que no se tienen permisos para acceder.
- **configuracion_sistema.xhtml**: Vista utilizada cuando se ejecuta por primera vez el sistema y no está configurado. Ofrece la configuración avanzada del sistema: Datos para el envío de correos electrónicos desde el sistema, elección de empresa principal proveedora de formación, creación de usuario administrador etcétera. También permite restaurar el sistema a los valores de fábrica (perdiendo todos los cambios realizados en el sistema, si no se mantienen copias de seguridad de la base de datos).
- **accionFormativa**: Carpeta que contiene las vistas para la consulta, gestión y requisitos funcionales asociados a las acciones formativas.
 - **lista.xhtml**: Vista utilizada para mostrar el listado de Acciones Formativas del sistema. Además, incluye funcionalidad para crear filtros sobre el listado y exportar hojas de cálculo para su gestión administrativa.
 - **accionFormativa.xhtml**: Vista utilizada para la gestión (creación, edición y borrado) de Acciones Formativas en el sistema. Incluye toda la funcionalidad para crear una Acción Formativa de forma eficiente.
- **calendario**: Carpeta que contiene la vista para la consulta, gestión y requisitos funcionales asociados al calendario de eventos.
- **empresa**: Carpeta que contiene las vistas para la consulta, gestión y requisitos funcionales asociados a las empresas.
 - **lista.xhtml**: Vista utilizada para mostrar el listado de Empresas del sistema. Además, incluye funcionalidad para aplicar filtros sobre el listado y exportar hojas de cálculo para su gestión administrativa. También incluye la funcionalidad para importar datos administrativos de las empresas desde la Fundación Tripartita de Formación en el Empleo, la generación de documentos de Matrícula para las empresas y la posibilidad de envío masivo de correos electrónicos a las empresas.
 - **empresa.xhtml**: Vista utilizada para la gestión (creación, edición y borrado) de Empresas en el sistema. Incluye toda la funcionalidad para crear una Empresa de forma eficiente.
- **factura**: Carpeta que contiene las vistas para la consulta, gestión y requisitos funcionales asociados a las empresas.
 - **lista.xhtml**: Vista utilizada para mostrar el listado de Facturas del sistema. Además, incluye funcionalidad para aplicar filtros sobre el listado y exportar hojas de cálculo para su gestión administrativa. Incluye también la funcionalidad para generar y gestionar las Remesas del sistema.
- **grupo**: Carpeta que contiene las vistas para la consulta, gestión y requisitos funcionales asociados a los grupos.
 - **lista.xhtml**: Vista utilizada para mostrar el listado de Grupos del sistema. Además, incluye funcionalidad para aplicar filtros sobre el listado y exportar hojas de cálculo para su gestión administrativa.

- **grupo.xhtml**: Vista utilizada para la gestión (creación, edición y borrado) de Grupos en el sistema. Incluye toda la funcionalidad para crear un Grupo de forma eficiente.
- **matricula**: Carpeta que contiene las vistas para la consulta, gestión y requisitos funcionales asociados a las matrículas.
 - **lista.xhtml**: Vista utilizada para mostrar el listado de Matrículas del sistema. Además, incluye funcionalidad para aplicar filtros sobre el listado y exportar hojas de cálculo para su gestión administrativa. También incluye la gestión de documentación de las matrículas, la gestión de eventos de las matrículas y la gestión de envío de archivos XML a la Fundación Tripartita de Formación en el Empleo sobre inicio y fin de grupos. Incluye la gestión masiva de matrículas, como la creación masiva de Facturas para las matrículas seleccionadas que no dispongan de estas.
 - **matricula.xhtml**: Vista utilizada para la gestión (creación, edición y borrado) de Matrículas en el sistema. Incluye toda la funcionalidad para crear una Matrícula de forma eficiente.
- **persona**: Carpeta que contiene las vistas para la consulta, gestión y requisitos funcionales asociados a las personas.
 - **lista.xhtml**: Vista utilizada para mostrar el listado de Personas del sistema. Además, incluye funcionalidad para aplicar filtros sobre el listado y exportar hojas de cálculo para su gestión administrativa.
 - **persona.xhtml**: Vista utilizada para la gestión (creación, edición y borrado) de Personas en el sistema. Incluye toda la funcionalidad para crear una Persona de forma eficiente.
- **usuario**: Carpeta que contiene las vistas para la consulta, gestión y requisitos funcionales asociados a los usuarios.
 - **lista.xhtml**: Vista utilizada para mostrar el listado de Usuarios del sistema. Además, incluye funcionalidad para aplicar filtros sobre el listado y exportar hojas de cálculo para su gestión administrativa.
 - **usuario.xhtml**: Vista utilizada para la gestión (creación, edición y borrado) de Usuarios en el sistema. Incluye toda la funcionalidad para crear un Usuario de forma eficiente.

Mapa del sitio

A continuación se presenta un mapa de acceso al sitio, con las posibilidades de entrar y moverse por las diferentes vistas del sistema (si se tienen todos los privilegios concedidos):



El inicio de la navegación se empieza en el cuadrado. Los círculos son vistas a las que se puede acceder. Las flechas indican la dirección para poder acceder correctamente a las vistas.

Conclusión

En esta sección, a modo de conclusión de este documento, se indica el uso y ventajas del sistema: un apartado para explicar cómo instalarlo, el siguiente apartado donde se explica cómo usarlo y un apartado final donde se detallan las ventajas que ha supuesto usar este sistema en un entorno real de producción (en una empresa de formación).

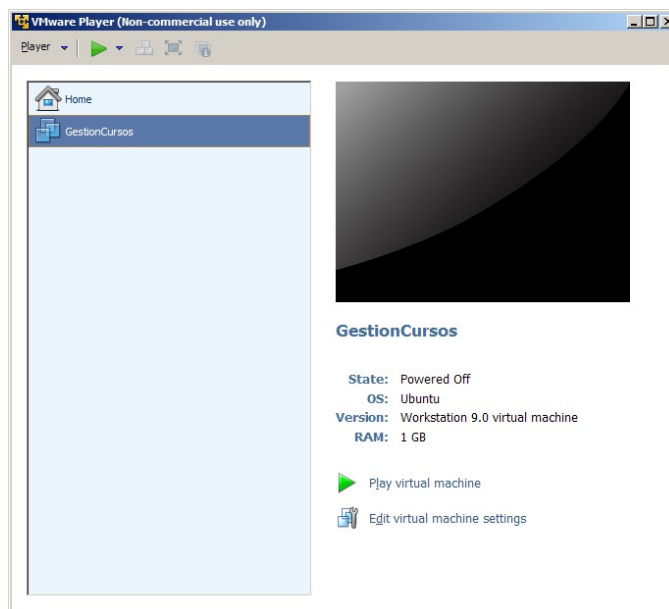
Guía de Instalación

A continuación se explican las formas de instalar el sistema en un entorno de producción. Pueden existir dos variables para instalar el sistema: utilizar la máquina virtual ya configurada, o instalar y configurar las aplicaciones que utiliza sistema en un sistema operativo. Se recomienda utilizar la instalación del sistema utilizando la máquina virtual ya que ahorra pasos a realizar en la instalación y configuración del sistema.

Usando Máquina Virtual

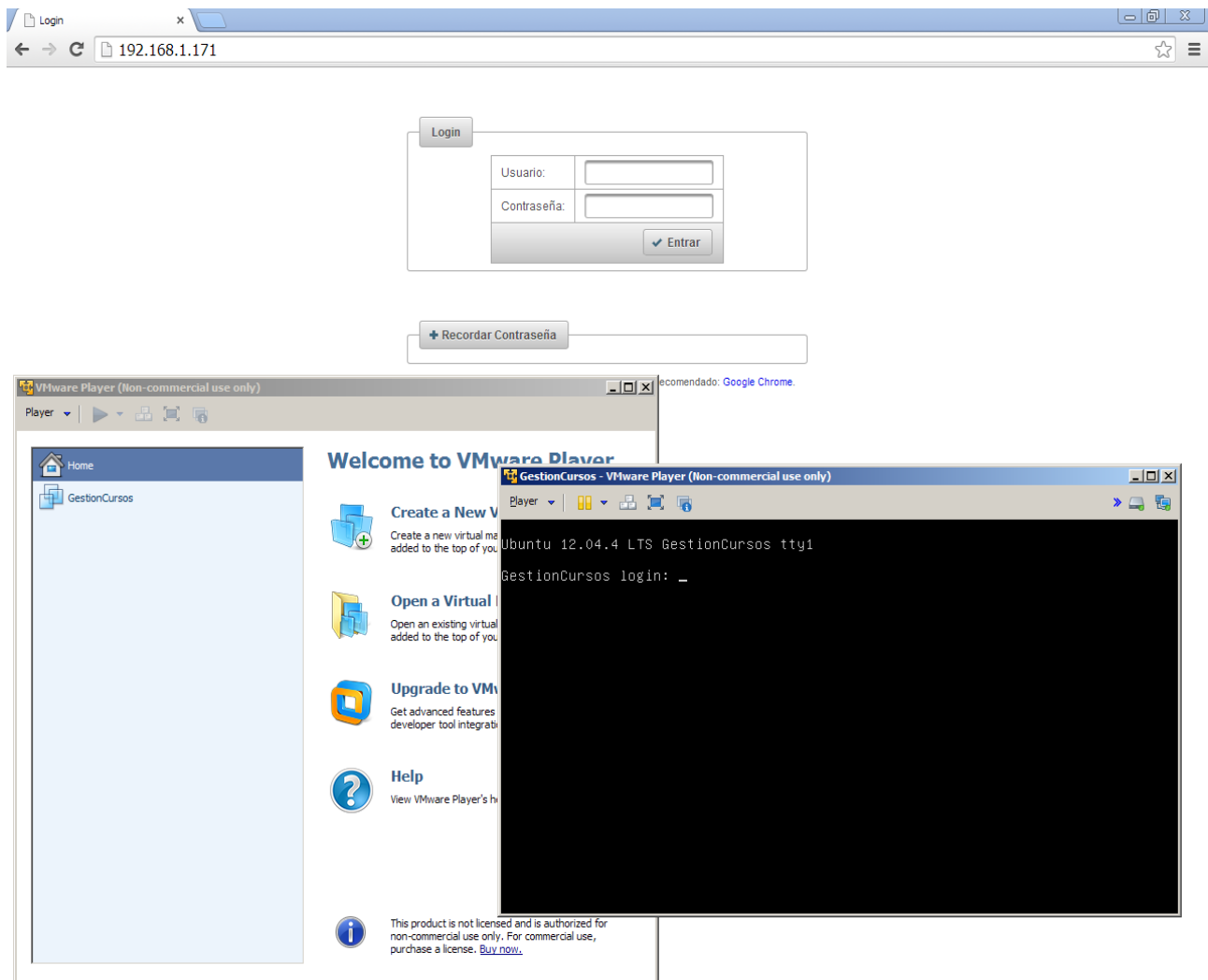
Como se indicó en el apartado “Tecnologías a utilizar”, es posible instalar el sistema de forma virtualizada usando el software de virtualización “VMWare Player 5.0.2”. Para instalar el sistema de forma virtualizada con este software seguiremos los siguientes pasos:

- Descargar e instalar el software de virtualización “VMWare Player 5.0.2” en el sistema operativo.
- Descargar y descomprimir la máquina virtual “GestionCursos.zip” (incluida en este proyecto).
- Abrir el software de virtualización “VMWare Player 5.0.2”.
- Iniciar la funcionalidad de abrir máquinas virtuales “Open a Virtual Machine”; buscar la carpeta descomprimida y seleccionar “GestionCursos.vmx”.
- La máquina virtual será añadida a la lista de máquinas virtuales del sistema.



- La máquina virtual está configurada para usar 1 GB de memoria RAM y 1 núcleo del procesador. Estas propiedades pueden cambiarse si pulsamos botón derecho en la máquina virtual → “Virtual Machine Settings...”.
- Iniciamos la máquina virtual seleccionándola y pulsando “Play Virtual Machine”. Con esto tendremos todo el sistema virtualizado listo para ser usado.

- Una vez iniciado el sistema, si la configuración de nuestra red tiene como puerta de enlace predeterminada “192.168.1.1”, podremos acceder al sistema a través de la dirección “192.168.1.171”, que es la dirección IP configurada por defecto en el sistema.

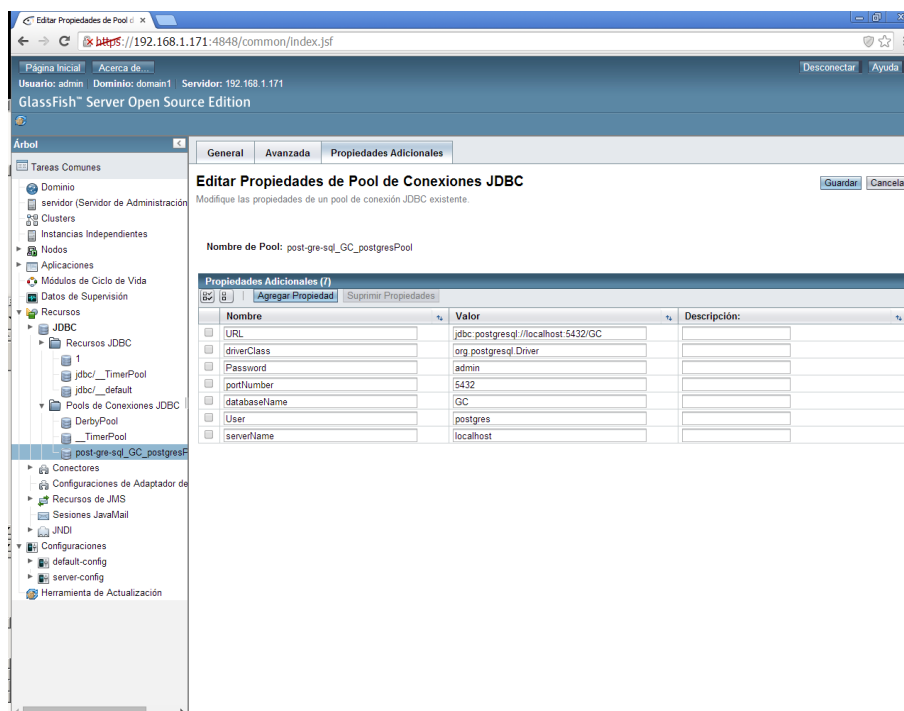


Configuración de GassFish y PostgreSQL en un servidor

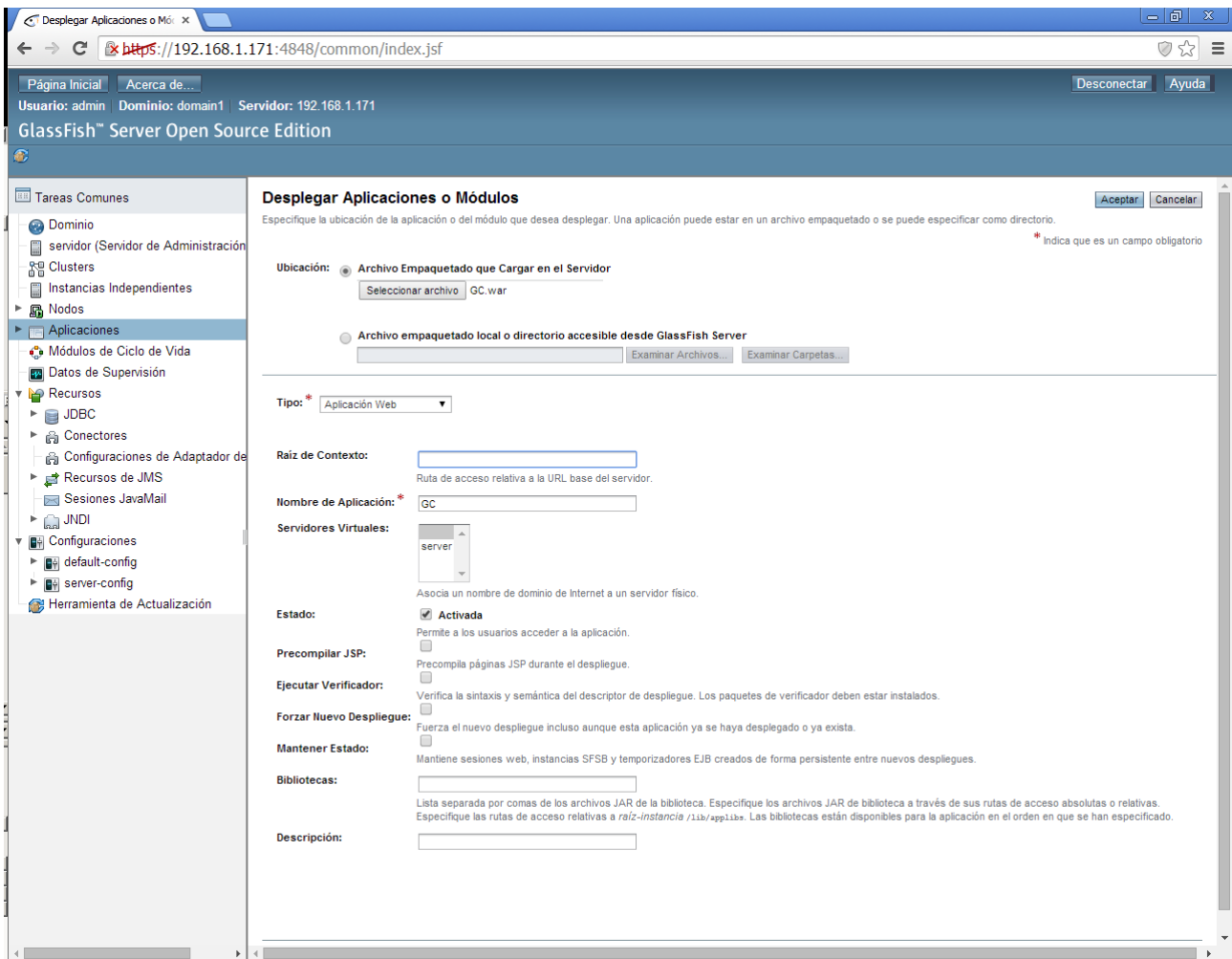
Si se desea instalar el sistema sin virtualizarlo, es necesario seguir los siguientes pasos en el sistema operativo en el que se desea instalar el sistema:

- Es necesario instalar el paquete de Java Development Kit versión 7 (JDK 7) de Oracle.
- Es necesario instalar el paquete de PostgreSQL versión 9.1.11.
- Es necesario instalar GlassFish 3.1.2. Al instalarlo crearemos el dominio “domain1” [22].
- Una vez instalado el software necesario, se configura para poder ejecutar el sistema:
 - Para instalar la base de datos, se debe crear una base de datos llamada “GC” en el Sistema Gestor de Base de Datos de PostgreSQL. En esta nueva base de datos se cargará el esquema relacional guardado en el archivo SQL “GC.dump” (incluido en este proyecto). Se puede conseguir ejecutando el comando “*psql -d GC < GC.dump*”.

- Para instalar la implementación Java del sistema, deberemos de entrar en la interfaz de administración de GlassFish. A esta interfaz se puede acceder desde el puerto 4848 del servidor (127.0.0.1:4848).
- Primero se debe de crear la conexión a la base de datos creada. Para ello debe de crearse un nuevo Pool de conexiones JDBC desde *Recursos* → *JDBC* → *Pools de Conexiones JDBC*, pulsando el botón *Nuevo...* (*Pool JDBC*).
- Nombre de Pool: **post-gre-sql_GC_postgresPool**
- Tipo de Recurso: **javax.sql.DataSource**
- Proveedor de Controladores de la Base de Datos: **org.postgresql.ds.PGSimpleDataSource**
 - Para que reconozca esta Clase de Origen de Datos, es posible que se deba de incluir la biblioteca “*postgresql-9.2-1002.jdbc4.jar*” dentro de la ruta de glassfish: */glassfish3/glassfish/domains/domain1/lib/*.
- Método de Validación: **auto-commit**
- Propiedades Adicionales:
 - URL: **jdbc:postgresql://localhost:5432/GC**
 - driverClass: **org.postgresql.Driver**
 - Password: **[password del usuario de la base de datos]**
 - portNumber: **5432**
 - databaseName: **GC**
 - User: **[usuario de la base de datos]**
 - serverName: **localhost**



- Tras esto, se añade la conexión de base de datos para que la aplicación pueda conectarse con la base de datos. Para ello debe de crearse un nuevo Recurso JDBC desde *Recursos* → *JDBC* → *Recursos JDBC*, pulsando el botón *Nuevo...* (*JDBC*).
- Debe de configurarse con las siguientes propiedades:
 - Nombre JNDI: **1**
 - Nombre de Pool: **post-gre-sql_GC_postgresPool**
- Una vez creada la conexión con la base de datos, se procede a desplegar la aplicación en el sistema. Para ello se siguen los siguientes pasos:
 - En la interfaz de administración de GlassFish, se accede a *Aplicaciones* → *Desplegar...* (*Nueva Aplicación*).
 - En *ubicación*, se selecciona la aplicación a desplegar “*GC.war*” (incluida en este proyecto).
 - El atributo *Raíz de Contexto* debemos dejarlo vacío.
 - Pulsamos “Aceptar”, y ya estará desplegada la Aplicación en el servidor GlassFish.



- Con esto se finaliza la instalación del sistema configurando GlassFish y PostgreSQL en un servidor.

Manual de Usuario

En esta sección se expone documentación de ayuda para la configuración y uso del sistema. Se divide en tres apartados: ejemplo de configuración real, para ilustrar cómo podría ser la configuración del sistema inicial para una empresa principal de formación; ejemplo de uso paso a paso, donde se explica cómo usar la aplicación en sus controles más básicos paso a paso; y un último apartado donde explica cómo usar la ayuda integrada en el sistema.

Ejemplo de configuración inicial

La empresa que desea usar este sistema va a ser la empresa “Gestión de Cursos”, con NIF 11111111H. Se enviarán los correos electrónicos del sistema desde la dirección de correo electrónico con los siguientes datos:

- Dirección: **prueba.gestion.cursos@gmail.com**
- Contraseña: **pruebagestioncursos2014**
- Servidor SMTP: **smtp.gmail.com**

[Estos datos pueden usarse como datos de prueba reales.]

Se creará una Firma para ilustrar el pie de página de los correos electrónicos que se envíen desde el sistema.

Configuración del Sistema

Ruta Absoluta del Proyecto: /opt/glassfish3/glassfish/domains/domain1/applications/GC

Dirección de envíos de E-mail: prueba.gestion.cursos@gmail.com

Contraseña de envíos de E-mail: pruebagestioncursos2014

Servidor SMTP de envíos de E-mail: smtp.gmail.com

Firma para envíos de E-mail:

Gracias por su confianza. Gestión de Cursos.

Crear Proveedor Principal Crear Persona Administradora Crear Usuario Administrador

Cargar Valores de Fábrica Guardar

Resolución Recomendada: 1440 x 900. Navegador Recomendado: Google Chrome.

Una vez rellenados estos datos, se procede a introducir en el sistema los datos de la empresa principal de formación, para ello se pulsa en el botón “Crear Proveedor Principal”. En esta vista se introducen los datos de la empresa principal de formación.

Nueva empresa

Configuración del Sistema

La empresa se ha guardado.

Datos de la Empresa

NIF: 11111111H (DNI)

Razón Social: Gestión de Cursos

Repres. Legal: Francisco Serrano Carmona (75165929C)

Email: prueba.gestion.cursos@gmail.com

Web: etsiit.ugr.es

Estado: Activa

Actividad: CENTRO DE FORMACIÓN

Dirección: C/ Saucedo Aranda s/n, 18007 Granada (Granada)

Volver a Configuración del Sistema Guardar

Resolución Recomendada: 1440 x 900. Navegador Recomendado: Google Chrome.

Tras guardar la empresa principal de formación, se procede a crear o seleccionar la persona administradora del sistema (que tendrá asociada el usuario administrador del sistema). Como en este ejemplo la persona asociada al sistema es el representante legal de la empresa principal de formación, no hará falta generar otra persona y entraremos en la funcionalidad de crear el usuario administrador. Este usuario tendrá los siguientes atributos:

- Nombre de usuario: **admin**
- Contraseña: **admin**

Nuevo usuario

Configuración del Sistema

El usuario se ha guardado.

Datos del Usuario

Persona asociada: Francisco Serrano Carmona (75165929C)

Nombre de Usuario: admin

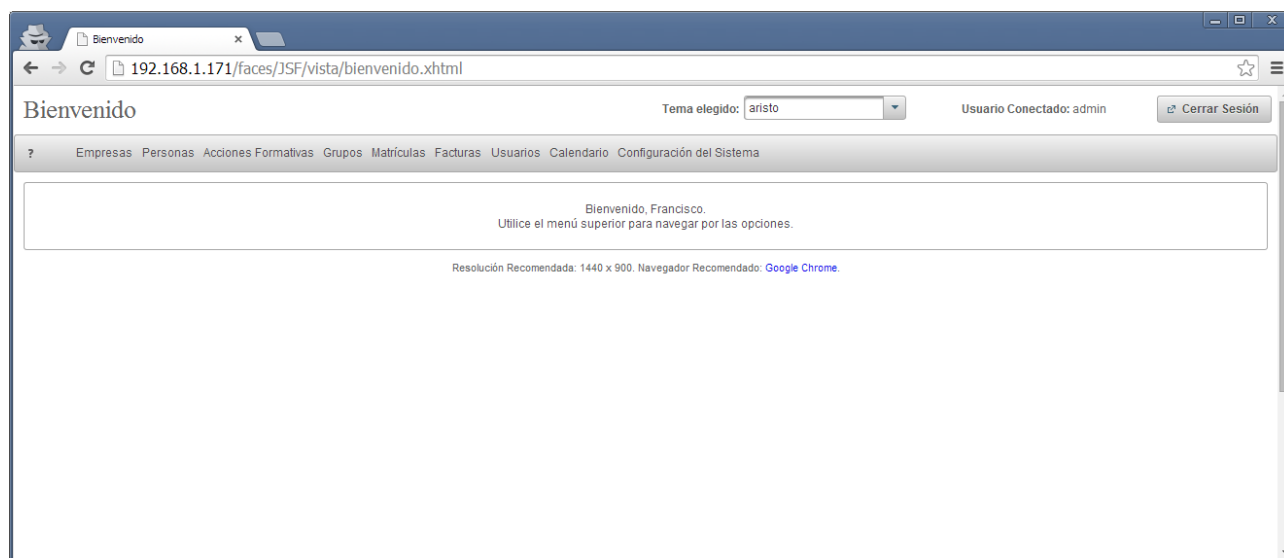
Email: francisco@correo.ugr.es

Nueva Contraseña:

Volver a Configuración del Sistema Guardar

Resolución Recomendada: 1440 x 900. Navegador Recomendado: Google Chrome.

Una vez introducidos todos los datos de configuración del sistema, se guarda en el sistema usando la opción “Guardar”. Una vez guardados los datos se pulsa “Conectar a la plataforma”, se introduce usuario y contraseña y ya se podrá acceder al sistema con la configuración inicial establecida.



Si el sistema ha sido instalado usando la máquina virtual incluida en este proyecto, los datos de usuario y contraseña de administración de las herramientas usadas en el sistema son los siguientes:

- Usuario/Contraseña en PostgreSQL: postgres/admin.
- Usuario/Contraseña en GlassFish: asdmin/admin.
- Usuario/Contraseña en el Sistema Operativo: administrador/admin.
- Usuario/Contraseña en el servidor FTP: administrador/admin.
- Si se desean cambiar las opciones de configuración de la red, se debe editar el archivo */etc/network/interfaces*:
 - address 192.168.1.171 *[dirección IP que se desea que tenga el servidor]*
 - netmask 255.255.255.0 *[máscara de subred de la red]*
 - network 192.168.1.0 *[dirección de la red]*
 - broadcast 192.168.1.255 *[dirección de difusión de la red]*
 - gateway 192.168.1.1 *[puerta de enlace de la red]*
 - dns-nameservers 192.168.1.1 *[servidor DNS de la red]*

Ejemplo de uso paso a paso

A continuación se expone un ejemplo de uso del sistema, para conocer cómo funciona la aplicación paso a paso. Partiendo del sistema usado en el apartado “Ejemplo de configuración inicial”, se mostrarán las acciones a realizar para conseguir dar de alta un alumno, una empresa cliente, una acción formativa, un grupo y una matrícula (proceso de matriculación). También se pondrá un ejemplo de generación de documentación para los actores implicados.

- Antes de poder comenzar el proceso de matriculación, se deben de rellenar los datos necesarios para la empresa principal de formación, ya que estos datos serán necesarios para la generación de documentación. Para ello se rellenan los datos de la empresa principal de formación desde *Empresas* → *Seleccionar empresa principal de formación (1111111H)* rellenos los siguientes datos:
 - La empresa principal de formación debe de ser empresa proveedora y empresa matriculable. Por lo tanto se seleccionan ambos tipos de empresa.
 - Deben de rellenarse los campos: NIF, Razón Social, Actividad, Dirección, Correo Electrónico, Web, Teléfono de la empresa.
 - De los datos para matricular, deben de rellenarse: Convenio Colectivo, Cuenta de Cotización a la Seguridad Social y un Código de Cuenta Corriente como dato bancario.
 - Se debe de añadir un numero de teléfono asociado a la persona que esté conectada a la plataforma. Una vez esté editada la persona, se añade a la empresa desde la sección “Personal de la Empresa”.
 - Se debe de añadir un tutor asociado a la empresa principal de formación, con un teléfono asociado al tutor. Para ello se introducen sus datos desde *Persona* → *Nueva persona*.
 - Una vez rellenos todos los campos, se guarda la empresa principal de formación.

Gestión empresa

La empresa se ha guardado.

Datos de la Empresa

NIF: 1111111H (DNI)

Razón Social: Gestión de Cursos

Repres. Legal: Francisco Serrano Carmona (75165929C)

Email: prueba.gestion.cursos@gmail.com

Web: etsit.egr.es

Estado: Activa

Matriculable

Es Proveedor

Actividad: CENTRO DE FORMACIÓN

Dirección: C/ Saucedo Aranda s/n , 18007 Granada (Granada)

Datos de Contacto

Teléfonos de la Empresa

Teléfono	Tipo
666000000	TELÉFONO

Personal de la Empresa

Tipo	Persona	Email	Tlf. Contacto
Contacto	Jaime Chamorro (41220246Z)		666000001 (TELÉFONO)

Datos para Matricular

Convenio Colectivo: Servicios a las empresas

Fecha de Creación:

Cuenta Cotización S.S.: 28123456742

Agrupación: Sin Agrupación

RLT:

Datos Bancarios

CCC	IBAN
41679179139107727085	

Datos por Año

Año	Plantilla Media	Crédito Asignado	Crédito Disponible	Ha creado Centro	Recibe Formación
La empresa no tiene Datos por Año asociados					

- Una vez guardados los datos de la empresa principal de formación, se procede a crear la matrícula, además de sus objetos relacionados (acción formativa, grupo, alumno, empresa cliente). Para crear la matrícula se procede desde *Matrículas* → *Nueva Matrícula*
 - Opcionalmente se puede generar una estructura comercial para que se registre qué ente de la estructura comercial tiene asociada esta matrícula. Para asociar estructuras comerciales a personas o empresas se procede desde la vista de personas o empresas, con el botón “*asociar estructura comercial*”, situado en la parte inferior de la vista. Para este ejemplo se ha guardado la empresa principal de formación como ente de la estructura comercial de tipo “distribuidor”, sin estructura comercial superior. Se ha asociado dicho ente de la estructura comercial a la matrícula a crear desde el apartado “*Estructura Comercial*” de la matrícula.
 - Para añadir la empresa cliente, desde la sección “*Empresa*” de la matrícula, se pulsa el botón “*Nueva Empresa*” y se rellenan los datos necesarios para crear una empresa matriculable. Una vez guardada la empresa, se pulsa “*Volver a Matrícula*” para seguir rellenando la matrícula.

Nueva empresa

Tema elegido: aristo

Empresas Personas Acciones Formativas Grupos Matrículas Facturas Usuarios Calendario Configuración del Sistema

La empresa se ha guardado.
La empresa se ha añadido a la matrícula.

La empresa se ha guardado.
La empresa se ha añadido a la matrícula.

Datos de la Empresa

NIF: L07629066 (CIF) Estado: Activa + Matriculable No es Proveedor

Razón Social: Empresa Matriculable 1 Actividad: PARAFARMACIA

Repres. Legal: Dueño Empresa Matriculable (24361897K) Dirección: C/ Mercedes 1, 18008 Granada (Granada)

Email: frandai2304@gmail.com

Web:

Datos de Contacto

Teléfonos de la Empresa

Teléfono	Tipo
866555555	TELÉFONO

Personal de la Empresa

Tipo	Persona	Email	Tlf. Contacto
La empresa no tiene Personal asociados			

Datos para Matricular

Convenio Colectivo: Sanidad Fecha de Creación:

Cuenta Cotización S.S.: 28123456742 Agrupación: Sin Agrupación RLT:

Datos Bancarios

CCC	IBAN
40204014247608970222	

- Una vez creada la empresa cliente, se procede a crear el alumno. Para añadir el alumno, desde la sección “*Alumno*” de la matrícula, se pulsa el botón “*Nuevo Alumno*” y se rellenan los datos necesarios para crear un alumno. Una vez guardado el alumno, se pulsa “*Volver a Matrícula*” para seguir rellorando la matrícula.

Nueva persona x

192.168.1.171/faces/JSF/vista/persona/persona.xhtml

La persona se ha guardado.
La persona se ha añadido a la matrícula.

La persona se ha guardado.

La persona se ha añadido a la matrícula. Alumno Masculino

Datos de la Persona

NIF: 77738201H (DNI)

Nombre: Alumno Apellido 1: 1 Apellido 2:

Número S.S.: 181039474231 Email: frandai2304@gmail.com Fecha de Nac.: 1/01/90 Fecha de Nac. Real

Dirección: Plz/ Gertrudis 2 , 18004 Granada (Granada)

Teléfonos de la Persona

Teléfono	Tipo
La persona no tiene Teléfonos asociados	

Empresas asociadas a Persona

Tipo	Estado	NIF	Razón Social
Matricular	Activa	L07629066	Empresa Matricutable 1

Datos de Alumno

No es Víctima de Violencia de Género No es Víctima de Terrorismo No es Discapacitado Asalariado

Nivel de Estudios: FP I o Enseñanza Técnico profesional equiv., Bachillerato Superior, BUP y equiv., FP II Grupo Cotización: (4) Ayudantes no titulados Área Funcional: Administración Categoría Profesional: Trabajador con Baja Cualificación Otras Titulaciones:

Nivel de Estudios Real Grupo Cotización Real Área Funcional Real Categoría Profesional Real

Observaciones:

B I U abc x' x' T t HI • T Y T

Undo Redo Bold Italic Underline Link Unlink Bulleted List Numbered List Decrease Indent Increase Indent Print

- Una vez creado del alumno, se procede a crear la acción formativa. Para añadir el alumno, desde la sección “Grupo” de la matrícula, se pulsa el botón “Nueva Acción Formativa” y se rellenan los datos necesarios para crear una acción formativa. Una vez guardada, se pulsa “Volver a Matrícula” para seguir rellenando la matrícula.

- En la matrícula se asocia el Grupo de tipo “Pendiente de Gestionar”. Este es un tipo de grupo especial que simboliza grupos de alumnos que están matriculados pero que aún no se sabe en qué fechas van a realizar la formación, o qué tutores van a tener asociados. En nuestro ejemplo se va a gestionar también el grupo, por lo tanto desde la sección “Grupo” de la matrícula, se pulsa el botón “Nuevo Grupo” y se rellenan los datos necesarios para crear el grupo. Una vez guardado, se pulsa “Volver a Matrícula” para seguir rellenando la matrícula.

Horario	NIF	Nombre	Email
	412202462	Jaime Chamorro	

- Para poder asociar tutores al grupo, el grupo pendiente de gestionar debe de tener tutores asociados. Esto se consigue asociando tutores al grupo pendiente de gestionar desde *Grupos* → *Seleccionar Grupo (10)*.
- En la vista de gestión de matrícula se rellenan los datos que faltan para poder crear la matrícula: fecha de firma de la matrícula, fecha de entrega de la matrícula, dirección de envío. También se puede elegir la forma de pago por defecto que se usará en la matrícula y si va a ser un pago de la empresa cliente en el mes de bonificación o va a pagar tras bonificar (si va a pagar lo más brevemente posible, se escoge la opción “Sin pago posterior”). Una vez introducidos los datos necesarios, se guarda la matrícula pulsando “*Guardar*”.

- El sistema nos avisa de que se han asociado las estructuras comerciales de la matrícula al alumno y a la empresa cliente, para asociar estos entes al ente de la matrícula.
- En la sección “*Eventos*” se puede observar que se ha creado un evento de “Confirmación de Matrícula”. Este evento simboliza la necesidad de llamar a la empresa cliente para confirmar los datos de la matrícula.

- Una vez creada la matrícula se procede a crear la factura de la matrícula. Esto se hace desde la misma interfaz de gestión de la matrícula, pulsando el botón “+”. Con esta función se creará una matrícula con el importe igual al precio de la matrícula (400 € en nuestro ejemplo), con la forma de pago elegida para la matrícula y en estado “Pendiente”. También se han generado los datos de costes para la Fundación Tripartita. Estos costes se generan automáticamente según requisito de la Fundación Tripartita de Formación en el Empleo con el siguiente criterio: del total del precio de la matrícula, 80% corresponde a “Costes de Impartición” y 20% corresponde a “Costes de Organización.”

The screenshot shows the 'Gestión matrícula' web interface. At the top, there are fields for 'Alumno: Alumno 1 (77738201H)', 'Grupo: 10-1 [F. Inicio: 03/07/2014 - F. Fin: 23/07/2014]', 'Acción Formativa: 10-Desarrollo Java (Distancia)', and 'Dirección Envío: Plz/ Gertrudis 2, 18004 Granada (Granada)'. Below these are buttons for 'Usar Dirección del Alumno' and 'Usar Dirección de la Asesoría'. A red box highlights the cost breakdown: 'Costes de Organización: 80.0 € - Costes de Impartición: 320.0 € - Costes Salariales: 0.0 €'. Below this is a text area for 'Observaciones' with a rich text editor toolbar. Further down, there is a section for 'Facturas' with a table showing a single invoice. A red box highlights this table. Below the 'Facturas' section is an 'Eventos' section with a 'Generar Eventos:' area containing buttons like 'Matrícula de Baja', 'Diploma Adicional', etc. At the bottom, there is a table with columns: Descripción, Estado, Fecha de Creación, Fecha de Realización, and Fecha de Vencimiento. The first row shows 'Confirmación de Matrícula' with a state of 'Pendiente' and a creation date of '03/07/2014'.

Historio	Número	Fecha	Importe	Forma de Pago	Estado
1	2014-07-0001	03/07/2014	400.0 €	RECIBO BANCARIO	Pendiente

Descripción	Estado	Fecha de Creación	Fecha de Realización	Fecha de Vencimiento
Confirmación de Matrícula	Pendiente	03/07/2014		18/07/2014

- Una vez creada la matrícula y la factura, se procede a generar la documentación para la empresa. Para generar la documentación, hay que ir a la vista de consulta de matrículas, en la sección *Matrículas* del menú. En esta sección podemos seleccionar la matrícula generada para generar documentación a imprimir pulsando el botón de impresora (situado a la izquierda de la fila de la matrícula a seleccionar). Si se quisiera enviar un correo electrónico de la documentación generada, se puede seleccionar el botón de correo electrónico que hay al lado de la impresora. En la parte inferior de la vista se encuentra toda la documentación que se puede generar a partir de la matrícula. Para este ejemplo se va a generar la documentación de inicio de curso, la cual está disponible seleccionando los botones de dicha sección y pulsando “*Generar documentación*”. Una vez generada la documentación se habrá generado un evento de “Envío de documentación”, para reflejar en la matrícula la acción realizada.

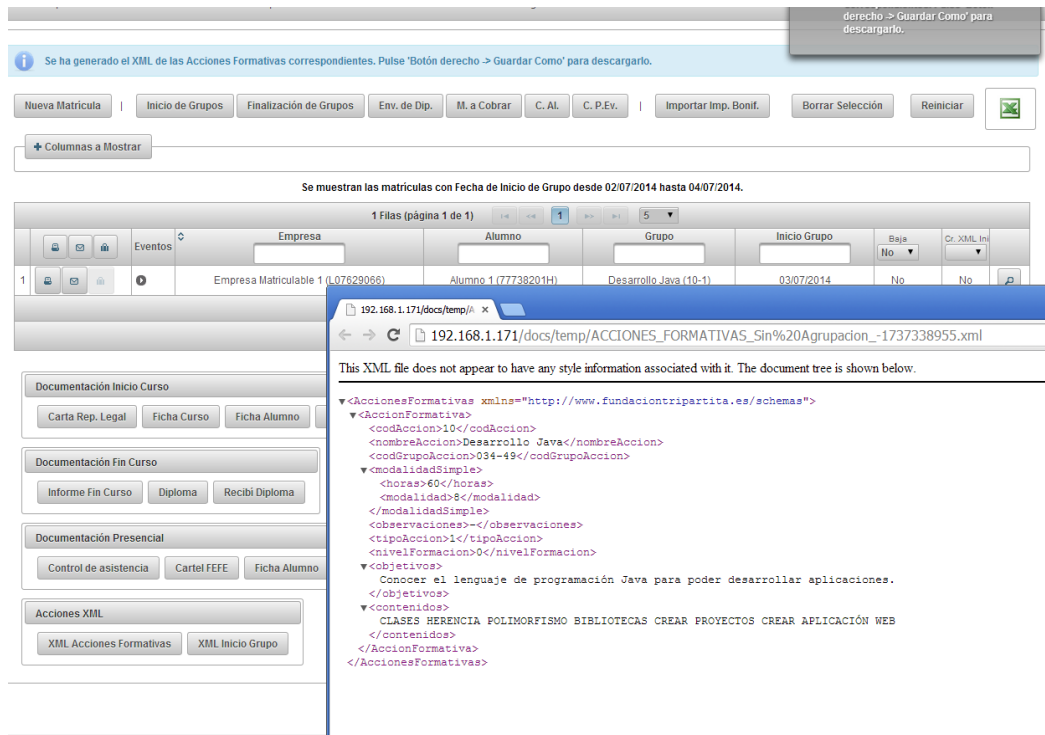
The screenshot displays the 'Matrículas' web application. At the top, there's a navigation bar with 'Matrículas' selected. Below it, a menu bar includes 'Empresas', 'Personas', 'Acciones Formativas', 'Grupos', 'Matrículas', 'Facturas', 'Usuarios', 'Calendario', and 'Configuración del Sistema'. A 'Tema elegido: aristo' dropdown is visible. A notification box states: 'Se ha generado el PDF correspondiente. Pulse "Descargar" para descargar los archivos.' Below this, a blue banner repeats the message. The 'Detalles de Evento' window is open, showing event details for 'Envío de Documentación'. It includes fields for 'Estado: Realizado', 'Tipo: Envío al Cliente', 'Empresa Asociada: Empresa Matriculable 1 (L07629066)', 'Alumno Asociado: Alumno 1 (77738201H)', 'Curso Asociado: Desarrollo Java (10-1)', and 'Matrícula: 12541 - 400,0 €'. The 'Observaciones' section contains a list of actions performed. At the bottom of the window are buttons for 'Eliminar', 'Control de asistencia', 'Cartel FEFE', and 'Ficha Alumno'. In the background, a PDF document is displayed with the following content:

Gestión de Cursos C/ Saucedo Aranda Nº s/n 18007 Granada CIF: 11111111H		Cliente Dirección	Empresa Matriculable 1 C/ Mercedes Nº 1 18008 Granada
		Cif/NIF	L07629066
Documento	Número	Fecha	Forma de pago
FACTURA	2014-07-0001	03/07/2014	RECIBO BANCARIO

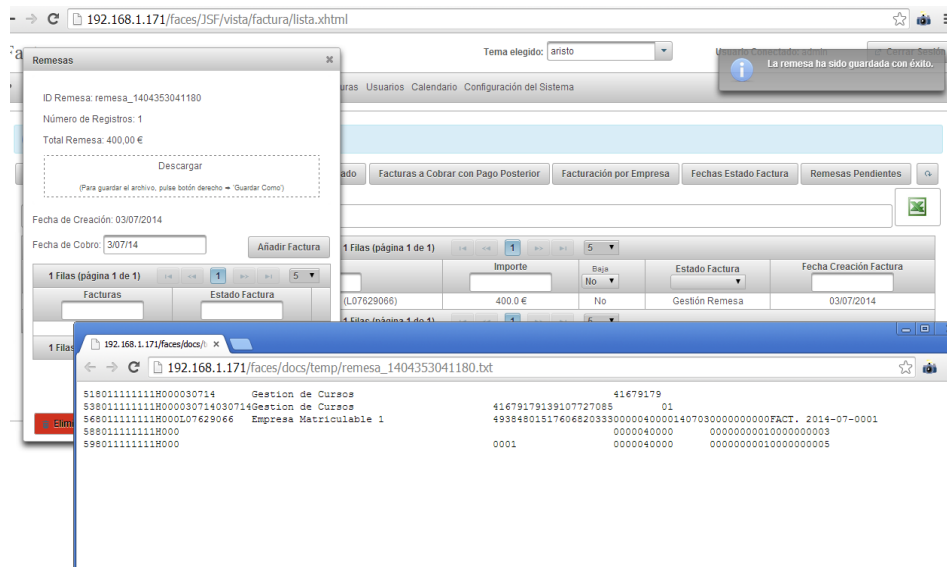
Below the table, the 'Descripción' section lists course details:

Cursos de Formación		
Acción Formativa	10/1	Desarrollo Java
Modalidad	Distancia	
Duración	60 horas	
Fecha inicio	03/07/2014	
Fecha Fin	23/07/2014	
Alumno	Alumno 1 - 77738201H	

- También se puede generar el XML a enviar a la Fundación Tripartita de Formación en el Empleo del inicio del grupo asociado a la matrícula. Esto puede conseguirse seleccionando el filtro “*Inicio de Grupos*”, situado en la parte de arriba de la vista. Se selecciona el rango de fechas de la fecha de inicio del grupo de la matrícula creada. Una vez seleccionada la matrícula, se genera el XML desde la sección *Acciones XML → XML Inicio Grupo*.



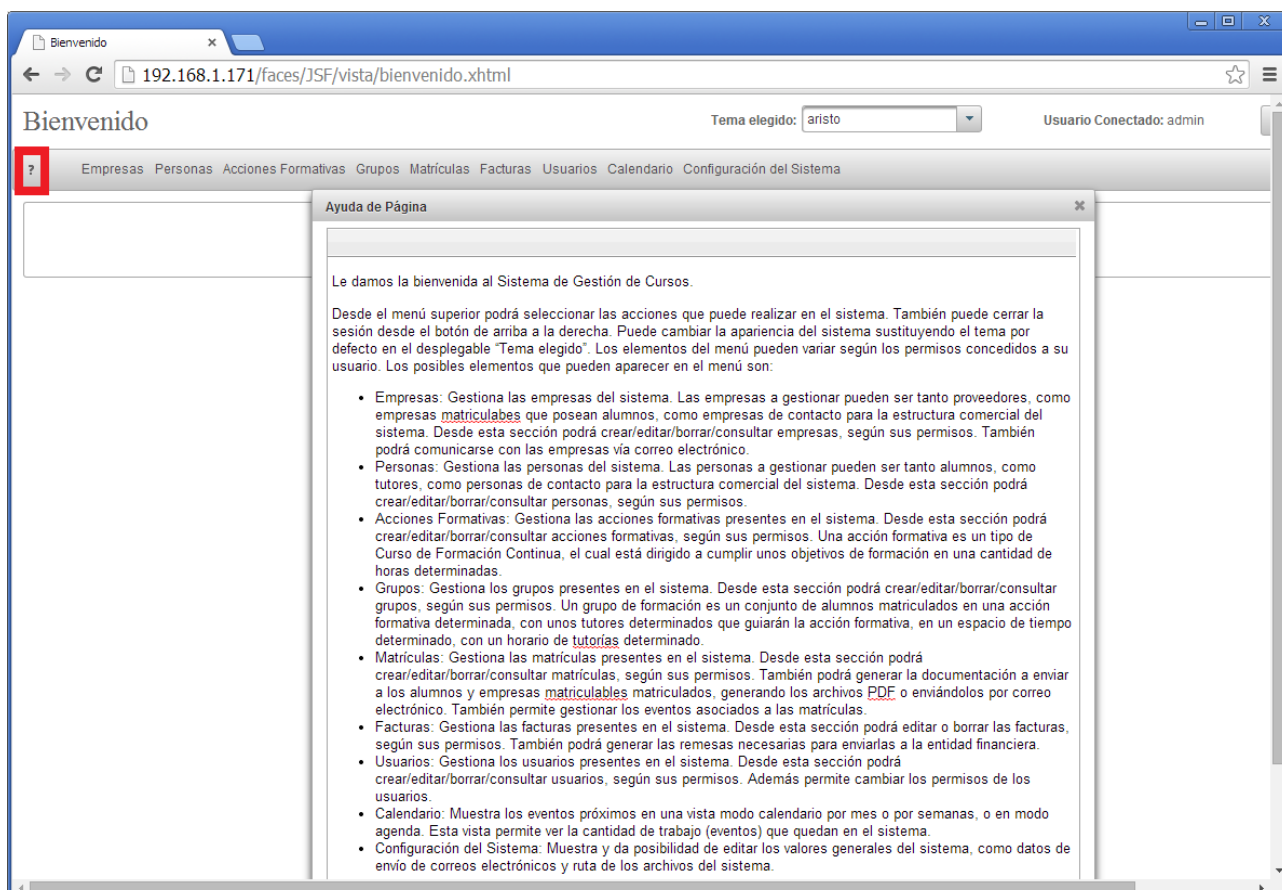
- Por último, como ejemplo de uso del sistema, se generará una remesa de la factura generada en la matrícula. Esto se consigue desde la sección *Facturas*, seleccionando la factura creada y pulsando en la acción “*Generar Remesa*”, en la parte inferior de la vista. Se debe rellenar una fecha de cobro. Una vez rellena, si se pulsa “*Guardar*”, se habrá generado una remesa siguiendo las especificaciones del **Anexo 3** de este proyecto. Este archivo puede enviarse a una entidad financiera para gestionar el cobro de los recibos de forma telemática.



Con estos pasos finaliza el ejemplo de uso. Existen más acciones posibles a realizar en el sistema (importación de datos de empresas de la Fundación Tripartita de Formación en el Empleo, gestión de eventos, etcétera). Lo mostrado en este punto es un ejemplo básico a modo ilustrativo.

Ayuda integrada

Para conocer todas las funcionalidades del sistema, existen páginas de ayuda integrada en las vistas de la plataforma. Se puede acceder a dichas ayudas desde el menú de opciones de las vistas (desde el icono de la interrogación, arriba a la izquierda).



Una vez se entra al sistema, todas las páginas disponen de esta ayuda integrada para poder conocer todas las acciones posibles en el sistema con facilidad, posibilitando que nuevos usuarios que utilicen el sistema puedan adaptarse a este con soltura, conociendo el funcionamiento de las vistas del sistema de las que tengan permisos.

Uso del sistema en un entorno real

Como conclusión final, se presentan las ventajas de implantar el sistema descrito en un entorno de producción en una empresa de formación real. El sistema descrito en el proyecto ha sido instalado en una empresa de formación real, aportando los siguientes beneficios para la empresa:

- Ha agilizado las tareas de administración de la empresa, puesto que anteriormente debían introducir los datos de empresas clientes, alumnos, acciones formativas, etcétera, de forma manual en el sistema interno de la Fundación Tripartita de Formación en el Empleo. Actualmente gracias al envío de archivos XML a la Fundación, el proceso administrativo se ha agilizado.
- Ha agilizado las búsquedas de documentos, empresas, matrículas, alumnos, etcétera, en la empresa: gracias a las vistas de consulta, estas búsquedas son rápidas y eficientes.
- Ha evitado la gestión manual de documentos impresos en la empresa, puesto que al estar todos los datos de la empresa y sus contactos dentro del sistema, y poder trabajar de forma digital, los documentos impresos no son necesarios. Esto ha reducido el gasto en papel y tinta en la empresa de una forma notable.
- Al ser un sistema web, los empleados, clientes y demás actores de la empresa pueden acceder al sistema desde cualquier punto con conexión a Internet. Esto posibilita que los trabajadores de la empresa puedan trabajar de forma telemática cómodamente sin tener que estar en la oficina.
- Se ha aumentado la calidad en los procesos con el cliente, puesto que al estar sistematizado en el sistema dichos procesos, los errores humanos administrativos se han reducido. Esto ha contribuido a una mayor satisfacción de los clientes de la empresa.
- Ha optimizado el cobro de facturas en la empresa, puesto que gracias al sistema implementado estas facturas se pueden cobrar de forma telemática por parte de la entidad financiera.
- Se tiene una mayor seguridad con los datos de la empresa, ya que el centralizar todos los datos en un mismo sistema posibilita una mayor facilidad de crear copias de seguridad de los datos de la empresa.

Aunque el proyecto actual es una versión estable y que cumple los requisitos previstos, en el futuro se desarrollarán e implementarán los siguientes requisitos funcionales en el sistema:

- Integración del sistema con plataformas de educación a distancia (tipo “moodle”) para que sean mas útiles para alumnos y tutores.
- Gestión de inventario de la empresa, para mantener un stock de material didáctico para enviar a los alumnos.
- Gestión de liquidaciones para la estructura comercial, para controlar un sistema de incentivos para comerciales y asesorías que consigan clientes para la empresa.

Si se desea probar el sistema desde Internet, existe una instancia destinada a pruebas en la dirección <http://gestion-cursos.noip.me>

Usuario: demo

Contraseña: demo

Bibliografía

1. Anónimo. Diseño orientado a objetos. Wikipedia. http://es.wikipedia.org/wiki/Dise%C3%B1o_orientado_a_objetos. Accedido el 7 de Julio del 2014.
2. Anónimo. Análisis y diseño orientado a objetos. Wikipedia. <http://es.wikipedia.org/wiki/ADOO>. Accedido el 7 de Julio del 2014.
3. Anónimo. Desarrollo ágil de software. Wikipedia. http://es.wikipedia.org/wiki/Desarrollo_%C3%A1gil_de_software. Accedido el 7 de Julio del 2014.
4. Anónimo. Programación extrema. Wikipedia. http://es.wikipedia.org/wiki/Extreme_Programming. Accedido el 7 de Julio del 2014.
5. César Arturo Guerra. Obtención de Requerimientos. Técnicas y Estrategia. SG. <http://sg.com.mx/revista/17/obtencion-requerimientos-tecnicas-y-estrategia>. Accedido el 7 de Julio del 2014.
6. Anónimo. Modelo relacional. Wikipedia. http://es.wikipedia.org/wiki/Modelo_relacional. Accedido el 7 de Julio del 2014.
7. Anónimo. Modelo entidad-relación. Wikipedia. http://es.wikipedia.org/wiki/Modelo_entidad-relaci%C3%B3n. Accedido el 7 de Julio del 2014.
8. Ricard Lou Torrijos. Introducción a la Tecnología JavaServer Faces. Programación en castellano. http://programacion.net/articulo/introduccion_a_la_tecnologia_javascript_faces_233. Accedido el 7 de Julio del 2014.
9. Anónimo. JavaServer Faces. Wikipedia. http://es.wikipedia.org/wiki/JavaServer_Faces. Accedido el 7 de Julio del 2014.
10. Riehle, Dirk (2000), Framework Design: A Role Modeling Approach. <http://www.riehle.org/computer-science/research/dissertation/diss-a4.pdf>. Accedido el 7 de Julio del 2014.
11. Anónimo. PrimeFaces. Wikipedia. <http://es.wikipedia.org/wiki/PrimeFaces>. Accedido el 7 de Julio del 2014.
12. Anónimo. Java Persistence API. Wikipedia. http://es.wikipedia.org/wiki/Java_Persistence_API. Accedido el 7 de Julio del 2014.
13. Doug Clarke. Introducing EclipseLink. ECLIPSE ZONE. <http://eclipse.dzone.com/articles/introducing-eclipselink>. Accedido el 7 de Julio del 2014.
14. Anónimo. Mapeo objeto-relacional. Wikipedia. http://es.wikipedia.org/wiki/Mapeo_objeto-relacional. Accedido el 7 de Julio del 2014.
15. Anónimo. JPA Queries. Objectdb. <http://www.objectdb.com/java/jpa/query>. Accedido el 7 de Julio del 2014.
16. Anónimo. Apache POI. Wikipedia. http://en.wikipedia.org/wiki/Apache_POI. Accedido el 7 de Julio del 2014.
17. Anónimo. Itext. Wikipedia. <http://es.wikipedia.org/wiki/IText>. Accedido el 7 de Julio del 2014.
18. Anónimo. Sistema de gestión de bases de datos. Wikipedia. <http://es.wikipedia.org/wiki/SGBD>. Accedido el 7 de Julio del 2014.
19. Anónimo. PostgreSQL. Wikipedia. <http://es.wikipedia.org/wiki/PostgreSQL>. Accedido el 7 de Julio del 2014.

20. Anónimo. Forma normal (base de datos). Wikipedia.
[http://es.wikipedia.org/wiki/Forma_normal_\(base_de_datos\)](http://es.wikipedia.org/wiki/Forma_normal_(base_de_datos)). Accedido el 7 de Julio del 2014.
21. Anónimo. GlassFish. Wikipedia. <http://es.wikipedia.org/wiki/GlassFish>. Accedido el 7 de Julio del 2014.
22. Marlon J. Manrique. Instalando GlassFish 3.1.2 en Ubuntu Server 12.04. Marlonj.
<http://www.marlonj.com/blog/2012/05/instalando-glassfish-3-1-2-en-ubuntu-server-12-04/comment-page-1/>. Accedido el 7 de Julio del 2014.
23. Anónimo. Modelo–vista–controlador. Wikipedia. <http://es.wikipedia.org/wiki/Modelo%E2%80%93vista%E2%80%93controlador>. Accedido el 7 de Julio del 2014.
24. Anónimo. ¿Quiénes somos?. Fundación Tripartita para la Formación en el Empleo.
<http://www.fundaciontripartita.org/Con%C3%B3cenos/Pages/Presentacion.aspx>. Accedido el 7 de Julio del 2014.