



Universidad
Carlos III de Madrid

Departamento de Ingeniería Telemática

PROYECTO FIN DE CARRERA
Ingeniería Técnica de Telecomunicación
Telemática

Aplicación web para la gestión
de un centro sanitario

Autora: Raquel Arenas Prieto
Tutor: Julio Villena Román

Leganés, octubre de 2015

Título:
Autor:
Director:

EL TRIBUNAL

Presidente: _____

Vocal: _____

Secretario: _____

Realizado el acto de defensa y lectura del Proyecto Fin de Carrera el día __ de _____
de 20__ en Leganés, en la Escuela Politécnica Superior de la Universidad Carlos III de
Madrid, acuerda otorgarle la CALIFICACIÓN de

VOCAL

SECRETARIO

PRESIDENTE

Agradecimientos

A todos los que en algún momento se han preocupado y/o alegrado por mí...

No pondré nombres propios porque todos sabéis quienes sois y siempre se quedaría alguno en el tintero.

Pero...

Al tutor de este proyecto, porque no debe ser una tarea nada fácil...

A mis compañeros-amigos de la universidad, porque esta etapa no hubiera sido igual...

A los trabajadores de la biblioteca de Leganés, porque como becaria tuve muchos de estos en mis manos, espero no daros mucho trabajo con el que ahora os mando yo...

A mis compañeros de trabajo, por la tabarra que os he dado con la entrega de esto.

Al grupo de 20 (hoy ya unos cuantos más), porque aunque desperdigados, un mensajito siempre alegra el día...

A mis padres, hermano y novio, porque siempre estáis ahí...

A la "pequeñaja" recién llegada al mundo, que quiso adelantarse por poquito a que su tía terminara con esto...

Y a toda mi familia (de nacimiento y la que se ha ido añadiendo), sois muchos y siempre he recibido palabras de ánimo de todos.

Resumen

El objetivo final de este Proyecto Fin de Carrera es el planteamiento, diseño y desarrollo de una aplicación web para la gestión de un centro sanitario con distintas especialidades.

Con esta aplicación web se llegará a gestionar el registro tanto de empleados como de pacientes en la base de datos. Se gestionará la agenda de los empleados sanitarios y las citas de los pacientes. Y se gestionará la historia clínica del paciente.

A esta aplicación tendrán acceso los empleados de la clínica, teniendo distintas funcionalidades según sean personal sanitario o no sanitario.

El personal no sanitario podrá gestionar los datos personales y de contacto de empleados y pacientes, y gestionar sus citas. El personal sanitario podrá hacerlo sobre su propia persona y sobre los pacientes, además podrá gestionar los datos clínicos de los pacientes.

Esta aplicación ha sido desarrollada en el lenguaje de programación Java, utilizando el framework JSF y la gestión de los datos sobre una base de datos MySQL. Y será soportada sobre un servidor web Tomcat.

Abstract

The final objective of this master Thesis is to develop a web application for the management of a sanitary center with different medical specialities.

This web application will allow to manage the record of both employees and patients in the database. Both sanitary employees agenda and patients appointments will be arranged and record with the applications. Clinical history of the patient will be managed too.

Application will be open for access to the employees of the clinic, having different functionalities depending on if they are sanitary or non sanitary personnel.

Non sanitary personnel will be able to manage the personal information and contact of employees and patients, and to manage his appointments. The sanitary personnel will be able to do this on his own person and on the patients, in addition it will be able to manage the clinical information of the patients.

This application has been developed in Java programming language using the framework JSF and the management of the information on a database MySQL. And it will be supported on a web server Tomcat.

Índice general

Agradecimientos

Resumen

Abstract

Índice general i

Índice de figuras iii

Índice de tablas v

Capítulo 1

Introducción y objetivos 1

1.1. Contexto 1

1.2. Motivación..... 4

1.3. Objetivos..... 4

1.4. Medios utilizados..... 5

1.5. Estructura de la memoria..... 6

Capítulo 2

Estado del arte 8

2.1. Introducción a las aplicaciones para gestión de los datos clínicos 8

2.2. Ejemplos de aplicaciones sanitarias 9

2.2.1. OMI 9

2.2.2. SALUS 12

2.2.3. Ofimedica 14

2.3. Protección de datos clínicos 15

Capítulo 3

Descripción del proyecto 16

3.1. Análisis del sistema 16

3.1.1. Perfiles de usuario 16

3.1.2. Especificación funcional..... 17

3.1.3. Modelo relacional de base de datos 30

3.1.4. Diagramas de flujo..... 47

3.2. Interfaz de usuario 49

3.2.1. Perfil no sanitario 51

3.2.2. Perfil sanitario 55

3.3. Pruebas 59

3.3.1. Pruebas perfil no sanitario 60

3.3.2. Pruebas perfil sanitario 63

Capítulo 4

Arquitectura tecnológica de la aplicación	68
4.1. Arquitectura del sistema	68
4.2. Medios utilizados y justificación	70
4.3. Instalación de las distintas herramientas para el desarrollo de la aplicación.....	73
4.4. Estructura de paquetes utilizada para el desarrollo de la aplicación	78

Capítulo 5

Planificación y presupuesto	81
5.1. Planificación	81
5.2. Presupuesto	82

Capítulo 6

Conclusiones y posibles trabajos futuros	84
6.1. Conclusiones.....	84
6.2. Posibles trabajos futuros.....	84
Glosario	87
Referencias	89

Índice de figuras

Figura 1.1. Plano planta principal del centro sanitario	2
Figura 1.2. Plano planta baja del centro sanitario.....	3
Figura 2.1. Pantalla de búsqueda de pacientes (OMI).....	10
Figura 2.2. Pantalla de gestión del paciente (OMI).....	10
Figura 2.3. Pantalla de gestión del historial clínico (OMI)	11
Figura 2.4. Pantalla de anamnesis del paciente	11
Figura 2.5. Funcionalidades (SALUS)	12
Figura 2.6. Pantalla de gestión agenda (SALUS).....	13
Figura 2.7. Pantalla de historia clínica (SALUS)	13
Figura 2.8. Pantalla Agenda (Ofimedic).....	14
Figura 3.1. Funcionalidad general	17
Figura 3.2. Funcionalidad “Gestión empleado”	18
Figura 3.3. Funcionalidad “Gestión paciente”	19
Figura 3.4. Funcionalidad “Gestión cita”	20
Figura 3.5. Funcionalidad general	21
Figura 3.6. Funcionalidad “Gestión paciente”	22
Figura 3.7. Funcionalidad “Gestión historia clínica”	23
Figura 3.8. Funcionalidad “Gestión episodio”	23
Figura 3.9. Funcionalidad “Gestión antecedente”	24
Figura 3.10. Funcionalidad “Gestión valor”	25
Figura 3.11. Funcionalidad “Gestión cita”	26
Figura 3.12. Esquema relacional de base de datos	30
Figura 3.13. Tabla EMPLEOS y script de creación	31
Figura 3.14. Tabla ESPECIALIDADES y script de creación	32
Figura 3.15. Tabla CODIGO_VALOR y script de creación	33
Figura 3.16. Tabla POBLACIONES y script de creación	33
Figura 3.17. Tabla PROVINCIAS y script de creación	34
Figura 3.18. Tabla COM_AUTONOMAS y script de creación.....	35
Figura 3.19. Tabla PAISES y script de creación	36
Figura 3.20. Tabla PACIENTES y script de creación.....	36
Figura 3.21. Tabla ANTECEDENTES y script de creación	37
Figura 3.22. Tabla EPISODIOS y script de creación	38
Figura 3.23. Tabla VALORES y script de creación	39
Figura 3.24. Tabla EMPLEADOS y script de creación	41
Figura 3.25. Tabla DATOS_PERSONALES y script de creación.....	43
Figura 3.26. Tabla USUARIOS y script de creación	45
Figura 3.27. Tabla CITAS y script de creación.....	46
Figura 3.28. Mapa web perfil no sanitario.....	47
Figura 3.29. Mapa web perfil sanitario.....	48
Figura 3.30. Diseño general de pantallas.....	49
Figura 3.31. Pantalla de validación “Login”	50
Figura 3.32. Pantalla de error de validación.....	50
Figura 3.33. Pantalla” PaginaPrincipal”	51
Figura 3.34. Pantalla “BusquedaPacientes”	51
Figura 3.35. Pantalla “DatosPersonalesPac”	52
Figura 3.36. Pantalla “Citas”	52

Figura 3.37. Pantalla “BusquedaEmpleados”	53
Figura 3.38. Pantalla “DatosPersonalesEmp”	53
Figura 3.39. Pantalla “Agenda”	54
Figura 3.40. Pantalla “PaginaPrincipal”	55
Figura 3.41. Pantalla “Agenda”	55
Figura 3.42. Pantalla “BusquedaPacientes”	56
Figura 3.43. Pantalla “DatosPersonalesPac”	57
Figura 3.44. Pantalla “DatosClinicos” (episodio)	57
Figura 3.45. Pantalla “DatosClinicos” (antecedente)	58
Figura 3.46. Pantalla “DatosClinicos” (valor)	58
Figura 3.47. Pantalla “Citas”	59
Figura 4.1. Arquitectura a nivel del sistema	68
Figura 4.2. Esquema MVC	69
Figura 4.3. Vista JSF	70
Figura 4.4. Vista String MVC	71
Figura 4.5. Conexión JDBC	72
Figura 4.6. Página de descarga de los JDK	73
Figura 4.7. Página de descarga de productos MySQL	74
Figura 4.8. Herramienta de diseño DBDesigner	74
Figura 4.9. Configuración tabla en DBDesigner	75
Figura 4.10. Página de descarga de servidor de aplicaciones	75
Figura 4.11. Librerías en el servidor de aplicaciones	76
Figura 4.12. Página de descarga del software y complementos de Eclipse	76
Figura 4.13. Entorno de desarrollo de la aplicación Eclipse	77
Figura 4.14. Fichero web.xml	77
Figura 4.15. Fichero faces-config.xml	78
Figura 4.16. Estructura del proyecto	78
Figura 4.17. Carpeta de ficheros WebContent	79
Figura 4.18. Carpeta de ficheros “Java Resources”	79
Figura 5.1. Ciclo de vida de software	81
Figura 5.2. Presupuesto de la fase de desarrollo	83
Figura 5.3. Presupuesto de implementación/puesta en marcha/mantenimiento	83

Índice de tablas

Tabla 3.1. Valores de la tabla EMPLEOS	32
Tabla 3.2. Valores de la tabla ESPECIALIDADES	32
Tabla 3.3. Valores de la tabla CODIGO_VALOR	33
Tabla 3.4. Pruebas pantalla “Login”	60
Tabla 3.5. Pruebas pantalla “PaginaPrincipal”	60
Tabla 3.6. Pruebas pantalla “BusquedaPacientes”	60
Tabla 3.7. Pruebas pantalla “DatosPersonalesPac”	61
Tabla 3.8. Pruebas pantalla de citas	61
Tabla 3.9. Pruebas pantalla “BusquedaEmpleados”	62
Tabla 3.10. Pruebas pantalla “DatosPersonalesEmp”	62
Tabla 3.11. Pruebas pantalla “Agenda”	63
Tabla 3.12. Pruebas pantalla “PaginaPrincipal”	63
Tabla 3.13. Pruebas pantalla “Agenda”	63
Tabla 3.14. Pruebas pantalla “BusquedaPacientes”	64
Tabla 3.15. Pruebas pantalla “DatosPersonalesPac”	64
Tabla 3.16. Pruebas pantalla “DatosClinicos” (episodio)	65
Tabla 3.17. Pruebas pantalla “DatosClinicos” (antecedente)	66
Tabla 3.18. Pruebas pantalla “DatosClinicos” (valor)	67
Tabla 3.19. Pruebas pantalla “Citas”	67

Capítulo 1

Introducción y objetivos

En este primer capítulo hacemos una introducción al proyecto, especificando el contexto en el que nos hemos basado para el análisis y desarrollo de la aplicación, describiendo el lugar, los equipos y los usuarios que tendrán acceso a esta.

También explicamos cuál es la motivación que lleva a la implementación de una aplicación como la que se desarrolla en este proyecto. Se listan los objetivos finales de éste y los medios que hemos empleado para ello.

Y por último hacemos un breve resumen de cada capítulo de esta memoria.

En esta introducción cabría aclarar que en la actualidad me encuentro trabajando en desarrollo de aplicaciones web, pero por confidencialidad no puedo poner algo real, por lo que aquí presento una aplicación sencilla basada en los proyectos en los que trabajo.

1.1. Contexto

El análisis y desarrollo de esta aplicación web, está dirigido a ser implantado en un centro sanitario para el registro y gestión de empleados, pacientes e historia clínica de éstos.

En este caso se trata de un centro sanitario simulado que podría ser ejemplo de uno real.

Aunque más adelante valoraremos las posibles adaptaciones y/o ampliaciones de esta aplicación, el entorno de utilización en el que nos hemos basado para desarrollarla es una clínica de atención y gestión privada, para la atención de pacientes en medicina familiar y distintas especialidades médicas. En esta clínica la atención es de consulta para los pacientes y se realizarán algunos tipos de pruebas como radiografías, ecografías... También contiene una consulta de fisioterapia con un gimnasio para atender la rehabilitación de los pacientes.

Esta clínica está en un solo edificio compuesto por:

- 12 consultas médicas (1 ordenador en cada consulta)
- 2 consultas de enfermería (1 ordenador en cada consulta)

- 1 gimnasio para fisioterapia (1 ordenador)
- 1 sala de curas (1 ordenador)
- 1 sala de descanso (1 ordenador)
- 1 despacho de dirección (1 ordenador)
- 1 mostrador central de recepción (3 ordenadores)
- 1 Archivo (1 ordenador y 2 servidores -uno de aplicación y otro de base de datos-)
- 2 salas de radiodiagnóstico (2 ordenadores)

En total: 24 ordenadores y 2 servidores de aplicación y base de datos.

En las dos figuras siguientes tenemos las dos plantas que componen la clínica en la que se querría implantar la aplicación.

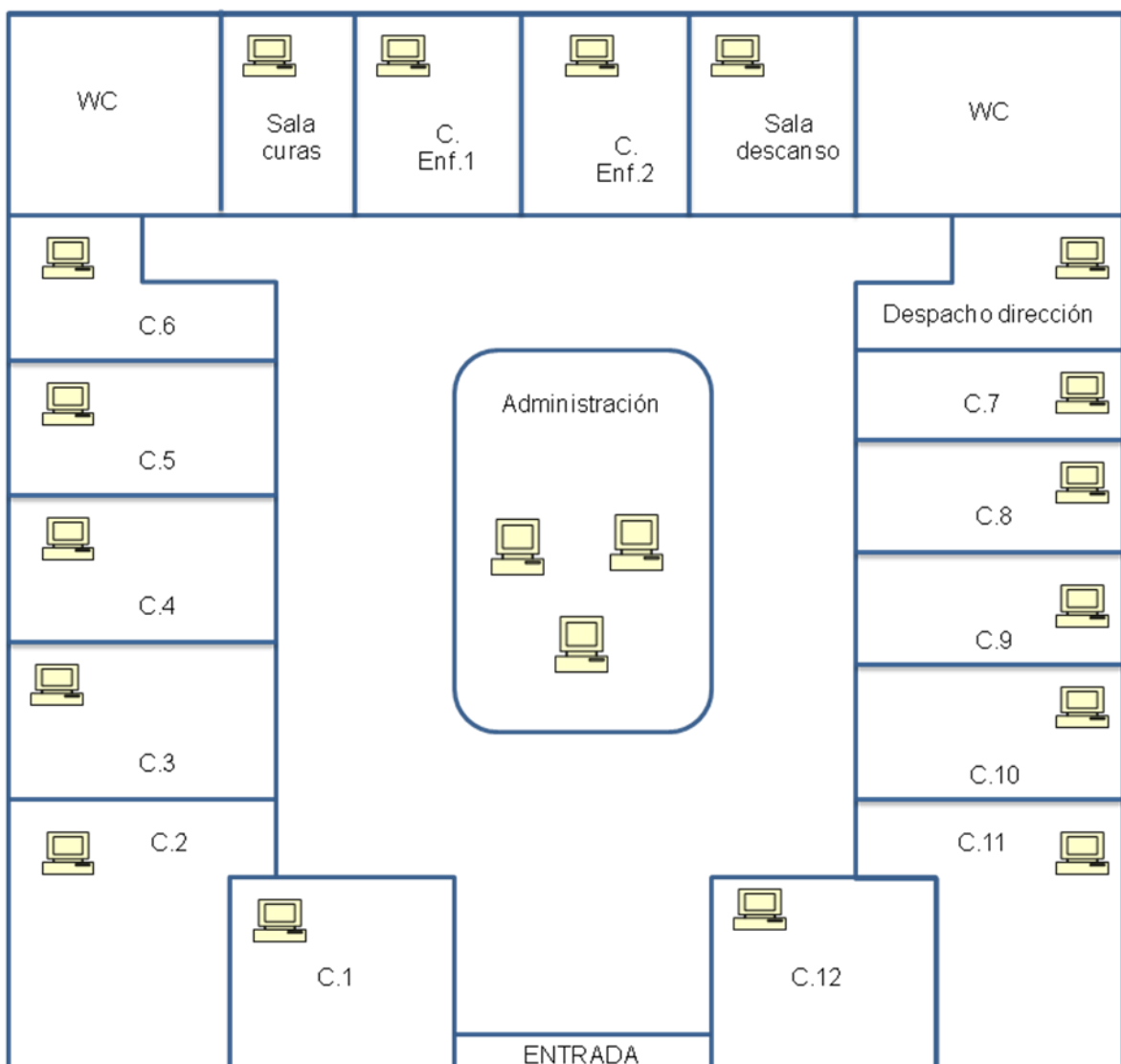


Figura 1.1. Plano planta principal del centro sanitario

En esta planta principal podemos observar:

- 12 consultas médicas con un ordenador cada una.
- 2 consultas de enfermería con un ordenador cada una.
- 1 mostrador de recepción donde se encontrarían 3 ordenadores.

- 1 sala de curas con un ordenador.
- 1 sala de descanso con un ordenador.
- 2 cuartos de baño.

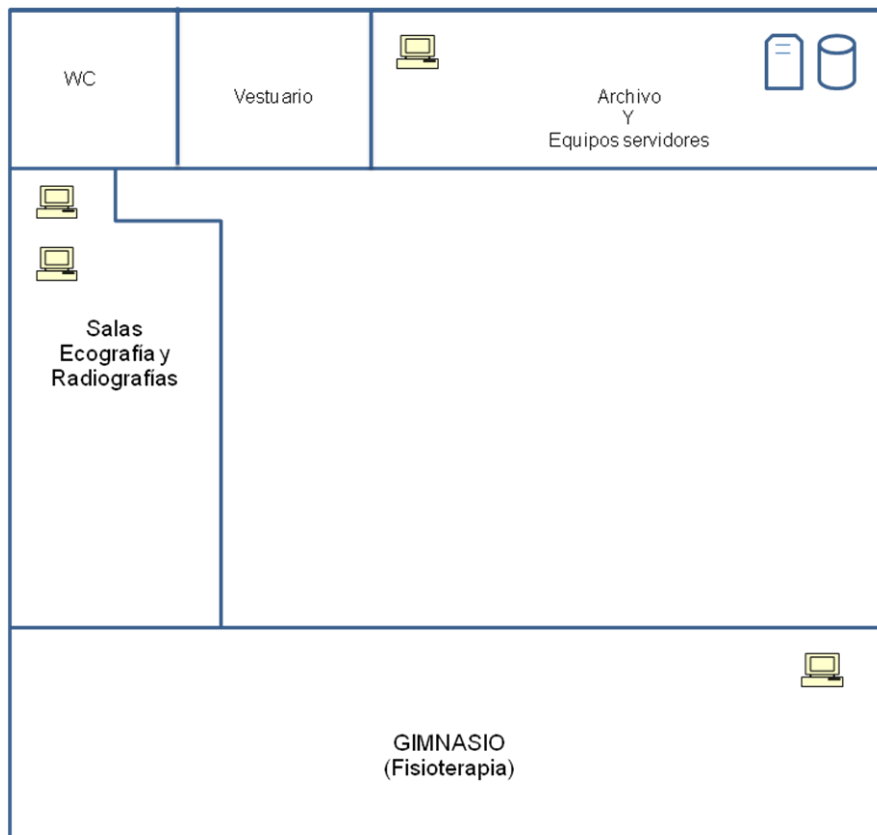


Figura 1.2. Plano planta baja del centro sanitario

En la planta baja, encontramos:

- Una sala de fisioterapia y gimnasio, con un ordenador.
- Varias salas de ecografías y radiografías que compartirán dos ordenadores.
- Un archivo en el que se encontrará un ordenador y el servidor de la aplicación con su respectiva base de datos.
- Un vestuario para los empleados.
- Un cuarto de baño.

Todos los ordenadores del centro se encuentran conectados a la red y por lo tanto tienen acceso a la aplicación.

Los empleados con acceso a la aplicación son:

- 4 administrativos
- 2 auxiliares de enfermería
- 4 enfermeros
- 2 fisioterapeutas
- 2 podólogos
- 17 médicos especialistas:
 - Dermatología (2 empleados)
 - Ginecología (2 empleados)

- Medicina familiar (5 empleados)
- Oftalmología (2 empleados)
- Otorrinolaringología (2 empleados)
- Pediatría (2 empleados)
- Traumatología (2 empleados)

En total 31 usuarios de la aplicación.

1.2. Motivación

La idea de realizar una aplicación web para la gestión de un centro sanitario surge de la necesidad que existe ante todo, de conservar la historia clínica de cada paciente, para la posibilidad de futuras consultas y así recordar lo acontecido, comprobar si se repiten los síntomas, si tienen relación unos con otros, comparar los valores tomados... y con ello poder diagnosticar de la manera más aproximada las dolencias del paciente, así como aplicar los mejores remedios.

La historia clínica comenzó a guardarse en papel, aunque poco a poco se está pasando al uso de la historia clínica electrónica.

Además de la gestión de la historia clínica del paciente, como ya hemos dicho, esta aplicación va dirigida a la gestión del centro sanitario, por lo tanto también se gestiona con ésta el registro de los empleados que trabajan en él y sus agendas.

1.3. Objetivos

Tal y como hemos comentado en las motivaciones, el objetivo fundamental de este proyecto es la construcción de una aplicación web para la gestión de un centro sanitario, con la que llegar a gestionar a los empleados, sus agendas, además de la gestión de los pacientes que asisten a él y la creación y mantenimiento de sus historias clínicas.

Los objetivos que nos marcamos desde el principio del proyecto para su desarrollo son:

- Estudio y decisión de una tecnología que se adapte a las necesidades.
- Estudio y modelado de las necesidades de la aplicación.
- Modelado e implementación de una base de datos para el contenido que deberá soportar.
- Desarrollo de la aplicación y simulación.

A continuación listamos las funcionalidades requeridas para esta aplicación, aunque entre los últimos capítulos de esta memoria contemplaremos un apartado de posibles mejoras y/o ampliaciones para ésta:

- Los usuarios entrarán al programa con su identificador y contraseña.
- Crear, buscar, modificar y borrar empleados.
- Crear, buscar, modificar y borrar pacientes.

- Crear, consultar y modificar datos en la historia clínica.
- Crear, consultar y eliminar citas.

Los empleados que serán registrados en la aplicación, serán también usuarios de ésta y serán los que estén directamente contratados por el centro sanitario. Dependiendo de si tienen una relación directa con el paciente a nivel administrativo o clínico, se determinará qué tipo de perfil se le aplica, como está especificado en el apartado 3.1.1. de esta memoria.

Los pacientes que se gestionarán desde la aplicación serán las personas que acudan al centro sanitario para ser atendidos por los empleados de este. Se gestionarán sus datos personales, de contacto y sus historias clínicas.

La historia clínica pertenecerá a cada paciente y estará compuesta por los episodios, antecedentes y valores.

Un episodio es la descripción de las causas y/o síntomas con los que llega el paciente a la consulta y las consideraciones que tome el empleado oportunas. Un ejemplo de episodio es: “A fecha 22-09-2015 el paciente asiste a la consulta con dolor de garganta, dolor de cabeza y fiebre de 37.8°. Se le receta amoxicilina 600mg cada 8 horas durante 7 días y paracetamol 500mg.”.

Un antecedente es cualquier episodio de relevancia que haya padecido el propio paciente o algún familiar consanguíneo de él. Por ejemplo: “Alergia a la penicilina” o “Abuelo con hernia de hiato a los 60 años”.

Un valor será cualquier medida que se tome al paciente. Por ejemplo: “22-09-2015 / Altura(metros) / 1.68”, “22-09-2015 / Peso(Kg) / 67”, “22-09-2015 / Tensión(alta,baja) / 13,7”, “22-09-2015 / Glucosa(mg/dl) / 80”.

Las citas que se guardarán en la aplicación serán los registros que forman la agenda de cada empleado, y relacionan al empleado con el paciente en un horario y fecha determinados.

1.4. Medios utilizados

Para el desarrollo de la aplicación hemos necesitado el uso de las siguientes herramientas:

Hardware (1 Equipo de desarrollo):

- Procesador: Intel Atom CPU Z520.
- Memoria RAM: 2GB.
- Sistema Operativo: Windows 7 32 bits.

Software:

- Lenguaje de programación: Java versión 7JRE (jdk1.7.0_51).

- Framework: JSF versión 2.2.8.
- Base de datos: MySQL versión 5.6.
- Servidor web: Apache Tomcat versión 8.0.3.
- Herramienta de desarrollo: Eclipse Java EE IDE for Web Developers (Version: Kepler Service Release 2).
- Explorador web: Internet Explorer version 8 y Google Chrome version 43.

1.5. Estructura de la memoria

A continuación incluimos un breve resumen de cada capítulo que compone esta memoria para facilitar su lectura.

Capítulo 1. Introducción y objetivos

Este capítulo en el que nos encontramos hace una introducción al proyecto como tal, situándonos en el contexto, y describiendo la motivación, los objetivos y los medios utilizados para su fin.

Capítulo 2. Estado del arte

El segundo capítulo nos explica el estado del arte del proyecto, hablando sobre las aplicaciones web que se utilizan actualmente para gestión clínica y las leyes sobre las que se apoya el criterio de acceso a cada funcionalidad de la aplicación.

Capítulo 3. Descripción del proyecto

El tercer capítulo nos muestra una descripción exhaustiva del proyecto a nivel de análisis del sistema, explicando los distintos perfiles de usuarios, la especificación funcional, la base de datos y los diagramas de flujo de la aplicación. También se muestra la interfaz de usuario de la misma, junto con un apartado de pruebas de ésta.

Capítulo 4. Arquitectura tecnológica de la aplicación

El capítulo 4 hace una descripción de la arquitectura del sistema en el que se implanta el proyecto. Explica las tecnologías que se utilizan y la instalación que se requiere para el entorno de desarrollo. Además se describe la estructura de paquetes que se ha utilizado.

Capítulo 5. Planificación y presupuesto

El capítulo 5 muestra la planificación que se debe llevar a cabo para el desarrollo e implantación del proyecto y su presupuesto con el desglose de los costes personales y materiales.

Capítulo 6. Conclusiones y posibles trabajos futuros

El capítulo 6 es el último de esta memoria y resume las conclusiones a las que nos lleva la finalización de este proyecto y nos describe posibles trabajos futuros que podrían ser añadidos a él.

Capítulo 2

Estado del arte

En este capítulo hacemos un estudio de cómo en la actualidad se gestionan los datos clínicos mediante aplicaciones en la sanidad pública en España y analizamos algunas de las aplicaciones web que se utilizan en el ámbito privado para la gestión de centros sanitarios.

También analizamos cuáles son las leyes de protección de datos, que hacen que no todos los empleados de un centro sanitario puedan acceder a los datos clínicos de un paciente y debido a esto, haya que tener distintos perfiles de usuario, para el acceso a las aplicaciones que gestionan este tipo de datos.

2.1. Introducción a las aplicaciones para gestión de los datos clínicos

Como viene siendo habitual en todos los ámbitos y a todos los niveles, en los últimos años se tiende a informatizar todo lo que se nos pueda pasar por la cabeza y mantener un registro de cada actividad que se realiza.

Y como no podía ser menos, en el campo de la medicina además de aplicable es muy beneficioso. Ya que la importancia de saber lo que le ha sucedido en otros momentos a un paciente, en ocasiones ayuda a poder dar un buen diagnóstico.

Actualmente en el Sistema Nacional de Salud ya se encuentran implantados los sistemas informáticos correspondientes para el registro de la historia clínica. Aunque en este momento la sanidad nacional está dividida por comunidades autónomas y cada comunidad tiene una aplicación distinta para su gestión. Por ejemplo para la Comunidad de Madrid para la atención primaria se comenzó utilizando el programa OMI-AP y ahora están utilizando AP-Madrid.

No obstante, existe un proyecto a nivel nacional llamado HCDSNS:

“La Historia Clínica Digital del Sistema Nacional de Salud (HCDSNS) tiene como finalidad garantizar a ciudadanos y profesionales sanitarios el acceso a la

documentación clínica más relevante para la atención sanitaria de cada paciente. Se incluye documentación que se encuentre disponible en soporte electrónico en cualquier lugar del SNS, asegurando a los ciudadanos que la consulta de sus datos queda restringida a quien esté autorizado para ello.” [1]

Se ha localizado un informe de situación de este proyecto a 20 de noviembre de 2014 [2]. El cual nos dice que el proyecto se encuentra implantado totalmente para la mayoría de las comunidades autónomas, con excepción de Asturias y Castilla y León que sólo emiten datos, Navarra que sólo los recibe, y Cataluña que todavía se encuentra desconectada por estar realizando tareas técnicas previas a las pruebas.

A nivel de sanidad privada la gestión ya es particular de cada clínica y nos encontramos con un gran abanico de posibilidades, pero al final todas las opciones llevan al mismo fin que es el de tener una base de datos que contenga la historia clínica del paciente.

2.2. Ejemplos de aplicaciones sanitarias

En este apartado vamos a explicar algunas aplicaciones sanitarias que se encuentran en el mercado en la actualidad.

2.2.1. OMI

En primer lugar tratamos la aplicación OMI que se utilizó en la Comunidad de Madrid. Es una aplicación de escritorio, aunque en la Comunidad de Madrid, la aplicación que se usa actualmente es una aplicación web y se llama AP-Madrid.

Su fabricante es la empresa STACKS, que se dedica en exclusiva a creación de aplicaciones sanitarias a nivel internacional. [3]

Esta aplicación tiene un abanico muy grande de funcionalidades, a continuación listamos algunas de ellas:

- Gestión de pacientes
- Gestión de historia clínica
- Gestión de episodios
- Gestión de vacunas
- Gestión de anamnesis
- Gestión de recetas

Sabemos que en el Gobierno del Principado de Asturias también fue utilizada y de su página oficial [4] hemos podido descargar una demo de esta aplicación y a continuación señalamos algunas de las opciones más significativas.

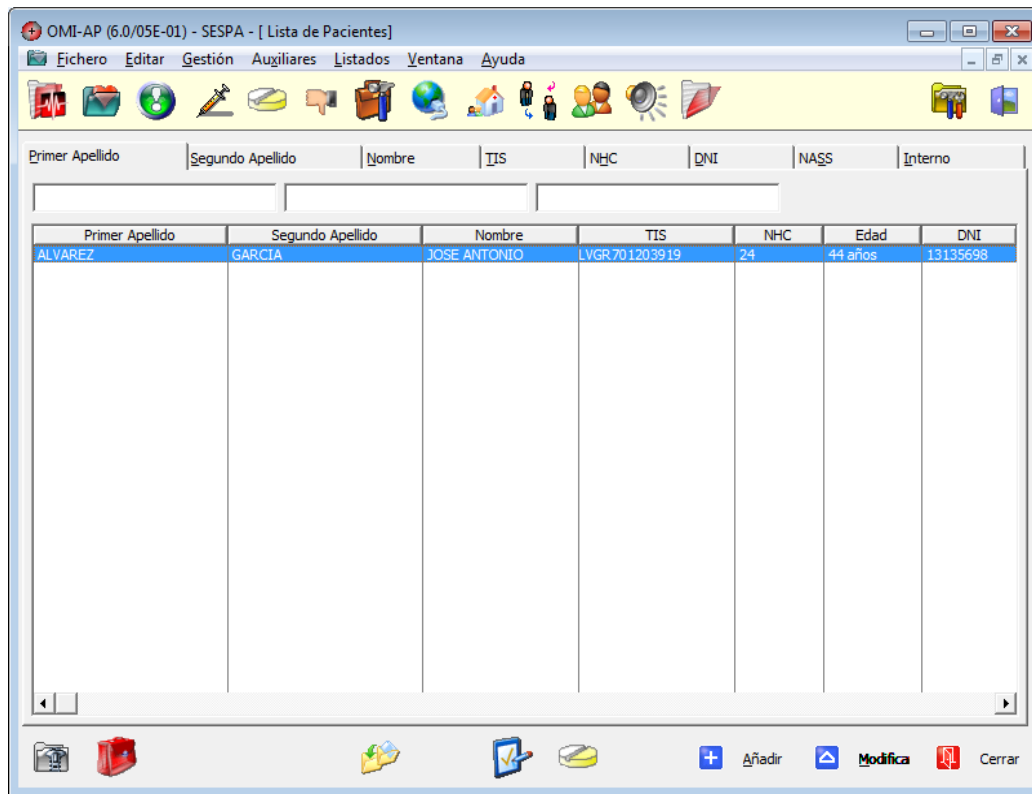


Figura 2.1. Pantalla de búsqueda de pacientes (OMI)

En esta pantalla vemos el listado de pacientes dentro de su pantalla de búsqueda de pacientes, que se podrá filtrar por las opciones que nos dejan las pestañas que aparecen.

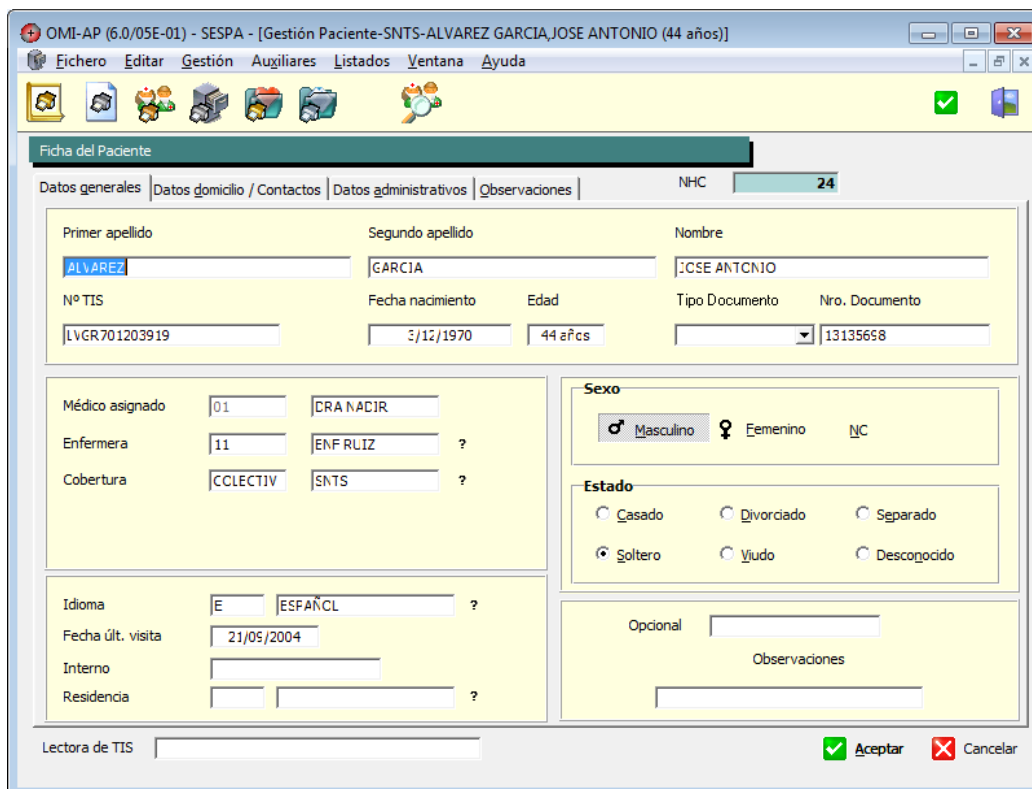


Figura 2.2. Pantalla de gestión del paciente (OMI)

En esta captura de la aplicación de escritorio OMI vemos la pantalla en la que se recogen todos los datos del paciente.

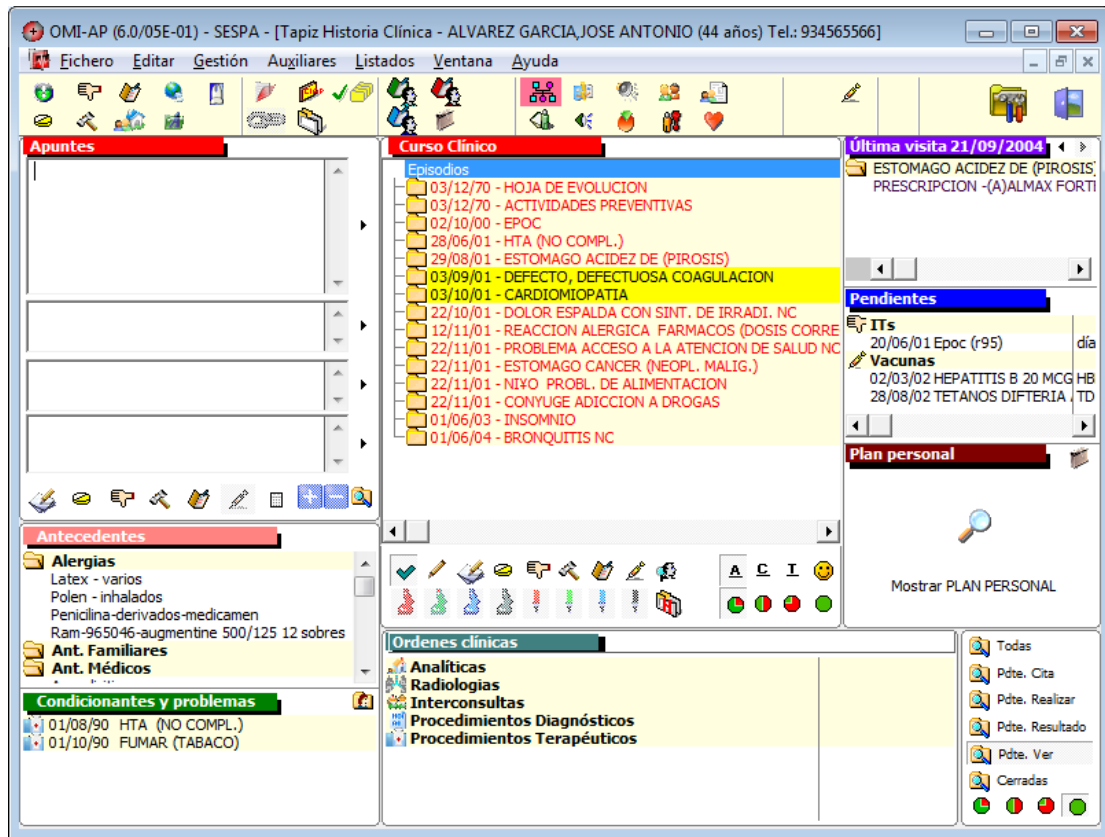


Figura 2.3. Pantalla de gestión del historial clínico (OMI)

En esta figura lo que podemos observar es el historial completo del paciente, dividido en varias partes entre las que cabe destacar por su mayor importancia los episodios y los antecedentes.

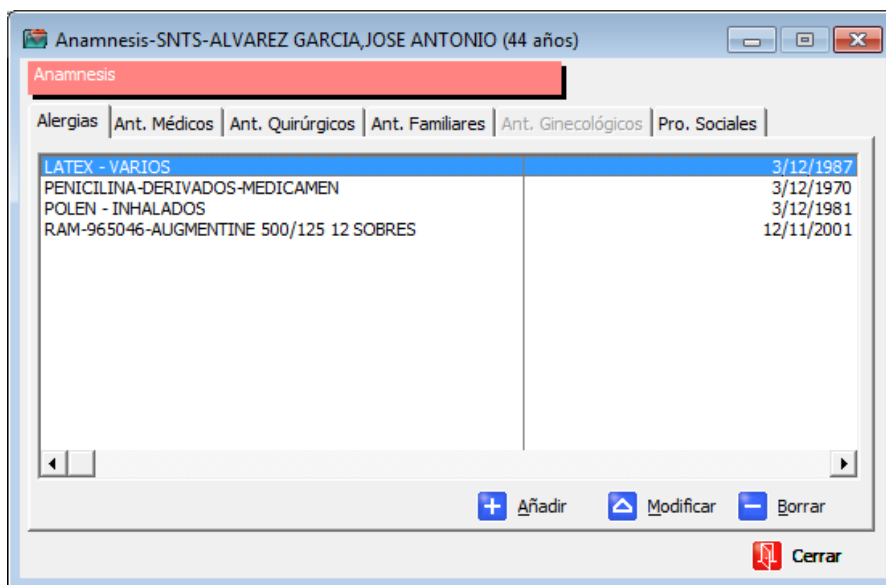


Figura 2.4. Pantalla de anamnesis del paciente

En esta última pantalla vemos la pantalla de gestión de la anamnesis, que serán los antecedentes y/o datos más relevantes del paciente.

2.2.2. SALUS

Es el software para hospitales y clínicas producido por la empresa española QSOFT, la cual lleva desde 1995 dedicándose al software especializado en la rama de la salud.[5]

Al igual que OMI las aplicaciones de esta empresa también son de escritorio.



Figura 2.5. Funcionalidades (SALUS)

Estas son las funcionalidades principales que cubre el software desarrollado por esta empresa. Agendas, historia clínica y económica.

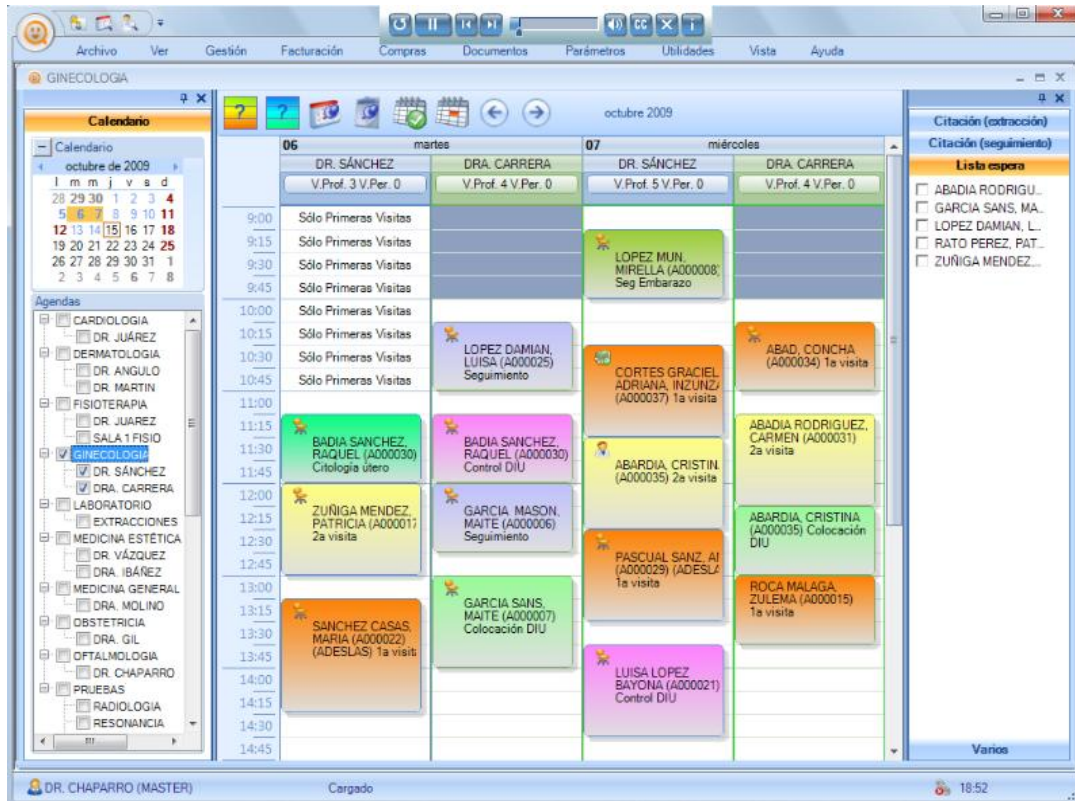


Figura 2.6. Pantalla de gestión agenda (SALUS)

Aquí podemos observar la pantalla que utilizan sus aplicaciones para la gestión de agenda de pacientes por especialidad y/o especialista.

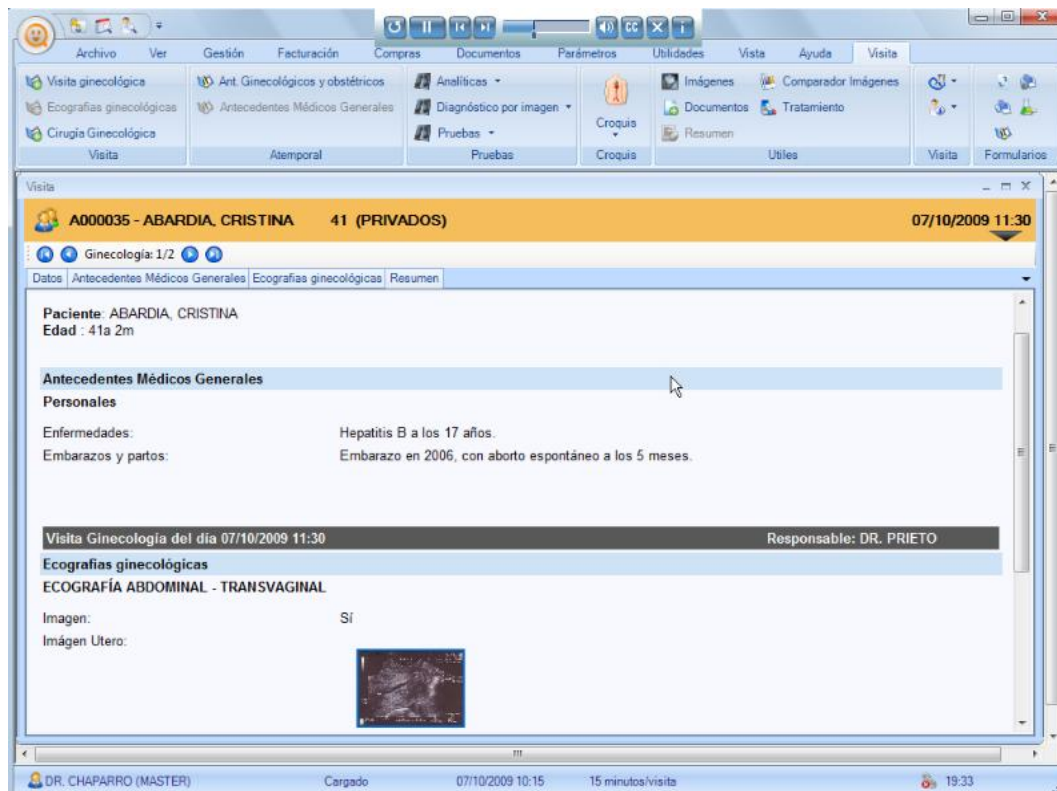


Figura 2.7. Pantalla de historia clínica (SALUS)

Aquí observamos el historial clínico de un paciente en una de sus aplicaciones y vemos que contemplan los datos que podrían ser nuestros episodios y los antecedentes del paciente como campos importantes.

2.2.3. Ofimedic

[6]Ofimedic es un software médico diseñado para la gestión global de consultas médicas y sus pacientes. En la actualidad mantiene dos versiones (ofimedic3 y ofimedic5), la primera está preparada para entornos muy pequeños de trabajo, en torno a cuatro o 5 consultas.

Ofimedic también es una aplicación de escritorio en lugar de web.

El fabricante de este software es la empresa Ofimedic Empresa.

Tiene muchas funcionalidades, pero como con las anteriores aplicaciones vemos las que más nos interesan, ya que son las más similares a nuestra aplicación.

La opción de “Recepción” nos permite gestionar las agendas tanto de los facultativos como del material que lo requiera.

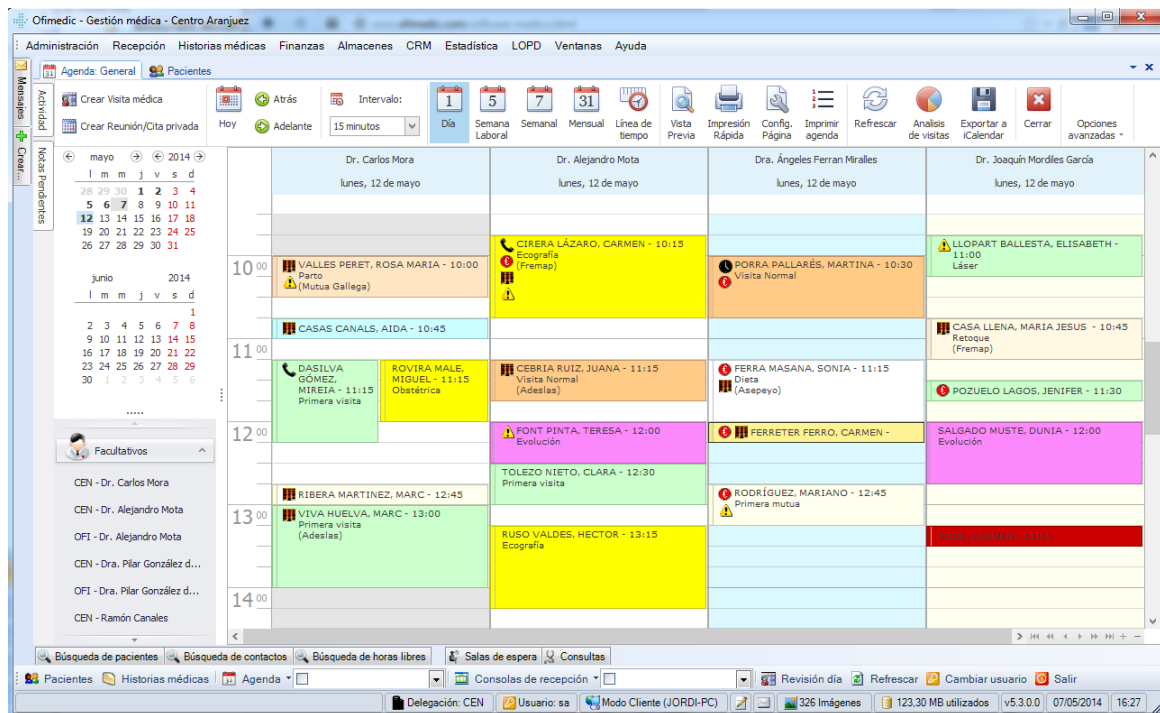


Figura 2.8. Pantalla Agenda (Ofimedic)

En esta pantalla se consultarán y gestionarán los valores de la agenda, seleccionando un periodo de tiempo, y pudiendo elegir también el elemento del que se quiere ver su agenda.

Y la opción de “historias médicas” que lo que permite es manejar todos los datos médicos del paciente.

2.3. Protección de datos clínicos

De las leyes para la protección de datos, la más conocida y que actualmente se encuentra en vigor es la LOPD (*Ley Orgánica 15/1999, de 13 de diciembre, de Protección de Datos de Carácter Personal*).

Pero, para el acceso a los datos clínicos, además de las leyes generales sobre la protección de datos, existen otras que tratan específicamente el tema de los datos clínicos.

Ley 41/2002, de 14 de noviembre, básica reguladora de la autonomía del paciente y de derechos y obligaciones en materia de información y documentación clínica [7]

En esta ley encontramos el artículo siguiente:

Artículo 16.

4. El personal de administración y gestión de los centros sanitarios sólo puede acceder a los datos de la historia clínica relacionados con sus propias funciones

Estas leyes nos llevan a tener una aplicación con un acceso restringido mediante una validación y contemplar una valoración de las funcionalidades a las que debe tener acceso cada empleo que existe en el centro sanitario, para asignarles distintos perfiles de acceso según las funciones que tengan que cumplir.

Debido a esto, en el apartado 3.1.1 de este documento especificamos dos tipos de perfiles y explicamos las funcionalidades a las que tiene acceso cada uno de éstos.

Estos perfiles definidos serán para los usuarios que lleguen a tener acceso a la aplicación, que serán los empleados que en sus funciones como empleado tengan un contacto directo con el paciente y necesiten dejar reflejados los datos, como por ejemplo un administrativo podrá dar de alta al paciente en el sistema y gestionarle una cita.

Con todo esto justificamos los dos perfiles de usuarios que encontramos en nuestra aplicación web.

Capítulo 3

Descripción del proyecto

En este capítulo vamos a describir minuciosamente la aplicación desde el punto de vista del diseño y desarrollo de ésta.

Primeramente haremos un análisis del sistema, estudiando los distintos perfiles de usuarios que la utilizarán, a partir de éstos, definimos las funcionalidades a las que tendrán acceso cada uno de ellos y describimos la base de datos que hemos diseñado para la gestión de los datos.

En este capítulo también queremos mostrar de forma visual cuál es el resultado del desarrollo de la aplicación, realizando una captura de cada pantalla que se puede ver desde cada perfil de usuario. Así como una batería de pruebas que hemos realizado a las funcionalidades que deben poderse llevar a cabo mediante ella.

3.1. Análisis del sistema

En este primer apartado de este capítulo, exponemos el análisis realizado para el desarrollo de la aplicación, a partir de los objetivos marcados por el proyecto que describimos ya en el apartado 1.3. de esta memoria y las leyes que existen sobre protección de datos que estudiamos en el apartado 2.2.

3.1.1. Perfiles de usuario

Nos encontramos ante un caso en el que podríamos hacer dos grandes grupos de usuarios.

- **No sanitarios:** que son empleados que no están relacionados directamente con la rama sanitaria, en nuestro caso, serían los administrativos.
- **Sanitarios:** que son empleados directamente relacionados con esta rama, en nuestro caso se encuentran los enfermeros, fisioterapeutas, podólogos y médicos.

Esta división es clara ya que como hemos comentado en el capítulo 2 sobre la protección de datos, los datos clínicos tienen un tratamiento especial, por lo que los empleados de la clínica no relacionados con la rama sanitaria, no podrán tener acceso a los datos clínicos de los pacientes, sino que tan solo tendrán acceso a los datos personales.

Aun así, a la hora de desarrollar la aplicación, sí que dejaremos abierta la posibilidad de poder definir más perfiles de usuarios.

Esto es porque en las posibles ampliaciones que comentamos en el capítulo 6, pueden necesitar de distintos perfiles de usuario entre el grupo de sanitarios, por ejemplo, a la hora de generar recetas, esto es competencia sólo para algunos de los sanitarios. O para el caso de que los pacientes llegaran a tener acceso a la aplicación, que tan solo podrían acceder a su propia información personal, historial clínico y/o sus propias citas.

3.1.2. Especificación funcional

Las funcionalidades de la aplicación serán distintas, según el usuario que la utilice, por lo que a continuación describimos la especificación funcional para cada perfil de usuario.

3.1.2.1. Funcionalidad de usuario no sanitario

A nivel general los usuarios no sanitarios tienen las siguientes funcionalidades globales:

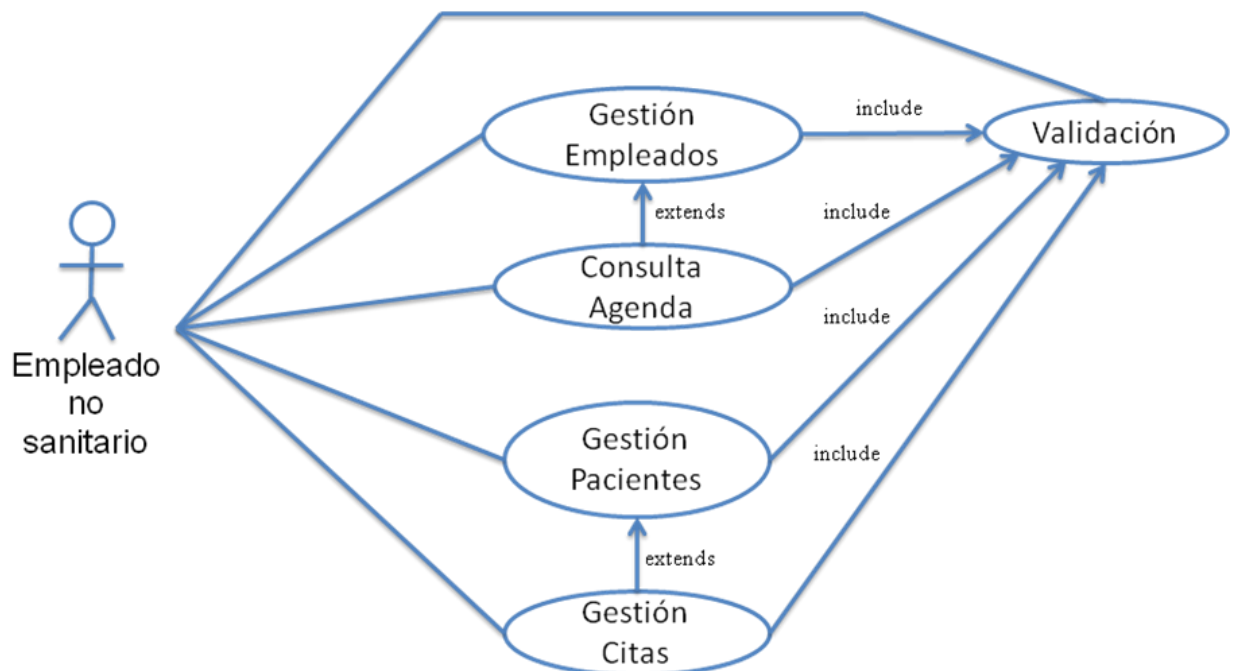


Figura 3.1. Funcionalidad general

Todos los usuarios no sanitarios, deberán validarse en la aplicación con su usuario y contraseña antes de acceder a cualquiera de las posibles funcionalidades a las que tienen acceso.

A partir de haber superado la “Validación”, este tipo de usuarios, tendrán acceso a “Gestión empleado”, “Consultar agenda”, “Gestión paciente” y “Gestión cita”.

La consulta de la agenda de los empleados es una extensión de la gestión de empleados, ya que se podrá consultar la agenda de cada empleado una vez estemos consultando cada uno de los empleados por individual.

Con la gestión de citas de los pacientes pasará lo mismo que con la consulta de agenda de los empleados, necesitaremos estar ya consultando un paciente en concreto para poder acceder a la gestión de sus citas.

• Gestión de los empleados:

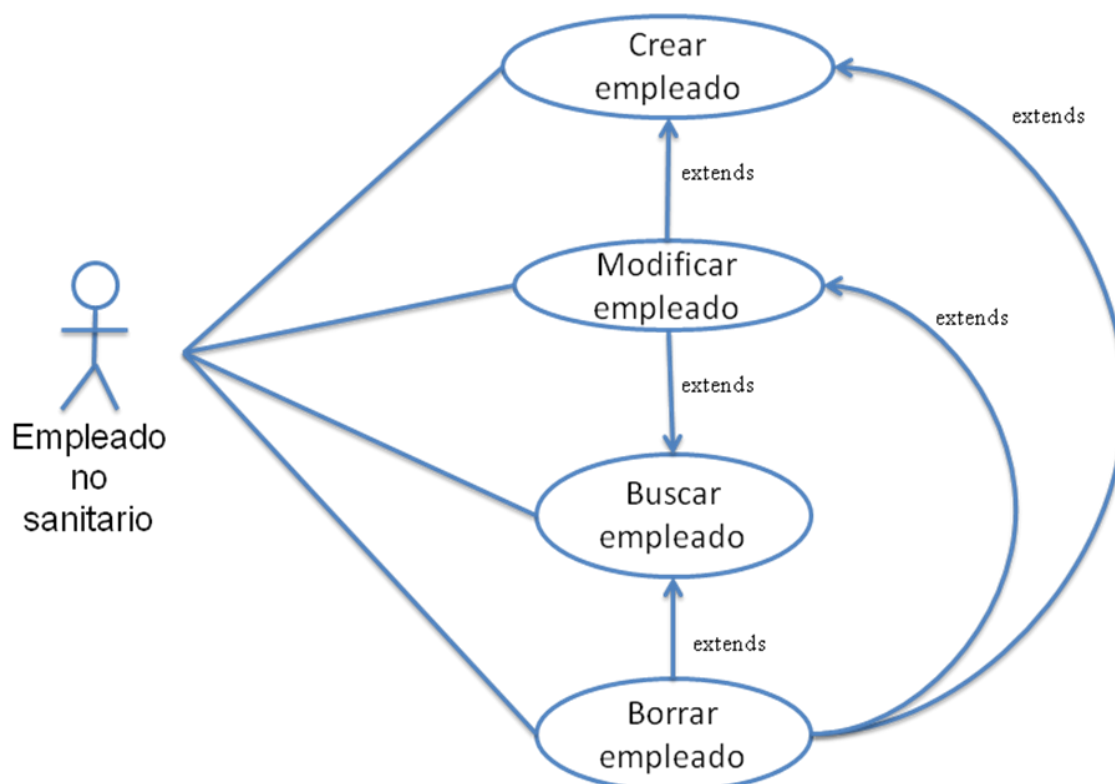
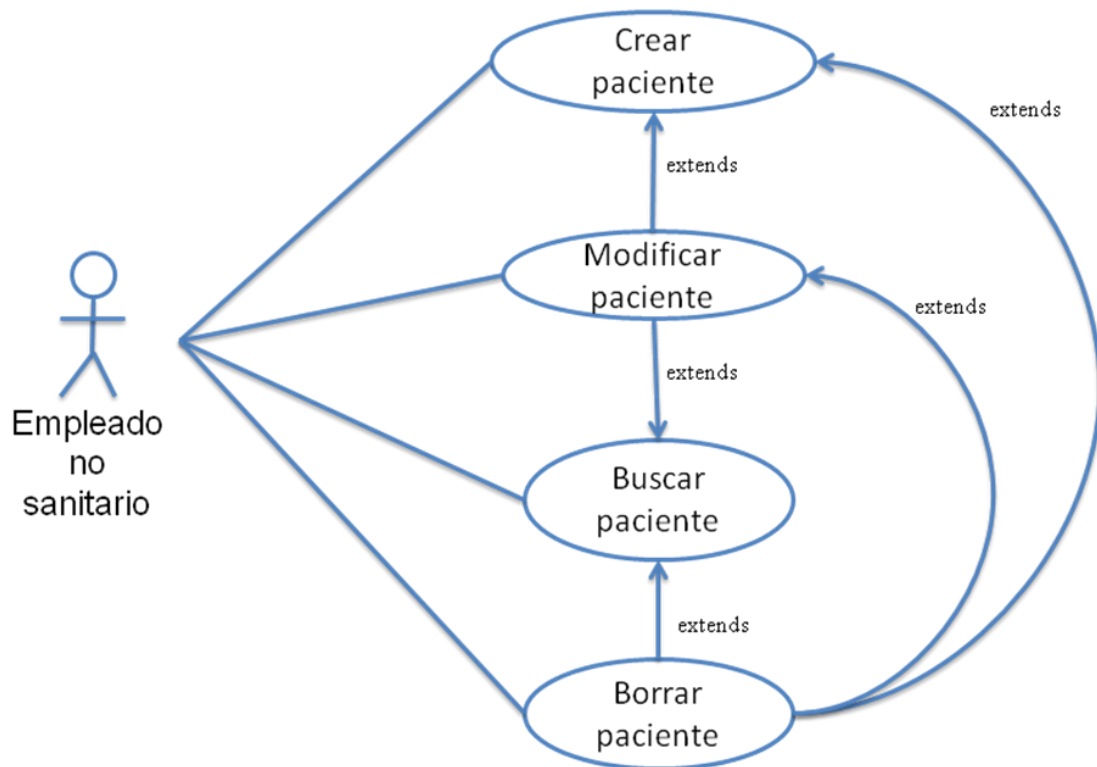


Figura 3.2. Funcionalidad “Gestión empleado”

En la gestión de empleados, los empleados de perfil no sanitario, tienen acceso a las funcionalidades “Crear empleado”, “Modificar empleado”, “Buscar empleado” y “Borrar empleado”.

La funcionalidad de “Crear empleado” da lugar a poder “Modificar empleado” y “Borrar empleado”. Y las funcionalidades de “Modificar empleado” y “Borrar empleado” serán accesibles también a partir de haber hecho la búsqueda de un empleado concreto.

• Gestión de los pacientes:

**Figura 3.3. Funcionalidad “Gestión paciente”**

Para la gestión de los pacientes por parte de los usuarios no sanitarios tendrán acceso a las funcionalidades “Crear paciente”, “Modificar paciente”, “Buscar paciente” y “Borrar paciente”.

La funcionalidad de “Crear paciente” da lugar a poder “Modificar paciente” y “Borrar paciente”. Y las funcionalidades de “Modificar paciente” y “Borrar paciente” serán accesibles también a partir de haber hecho la búsqueda de un paciente concreto.

• Gestión de citas:

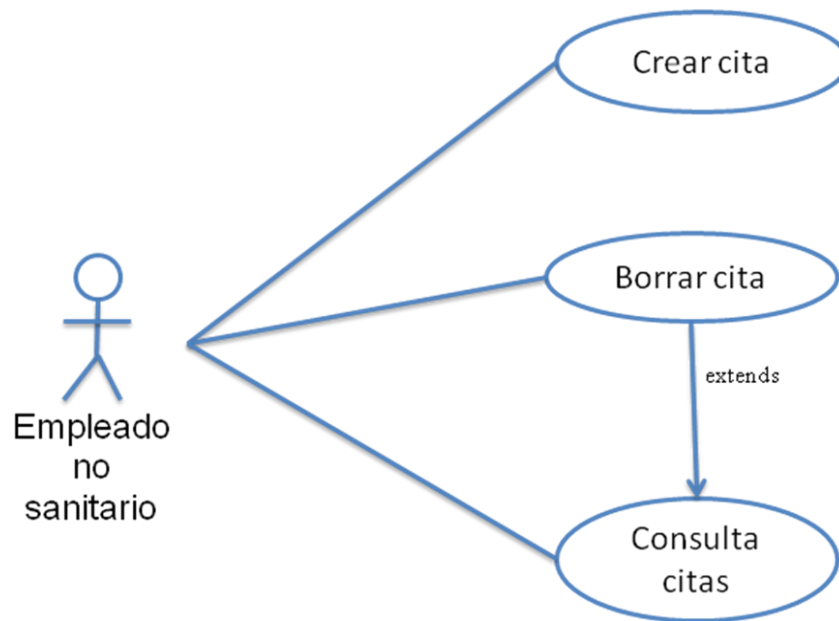


Figura 3.4. Funcionalidad “Gestión cita”

La gestión de las citas se realiza por los usuarios no sanitarios, a partir de la gestión de un paciente en concreto y tiene las funcionalidades “Crear cita”, “Borrar cita” y “Consultar citas”.

En la gestión de citas los usuarios no sanitarios acceden a la funcionalidad de “Consultar citas”, y a partir de esta tienen acceso a las funcionalidades de “Crear cita” y “Borrar cita”.

3.1.2.2. Funcionalidad de usuario sanitario

Las funcionalidades globales a las que tienen acceso los usuarios sanitarios, las mostramos en el siguiente diagrama.

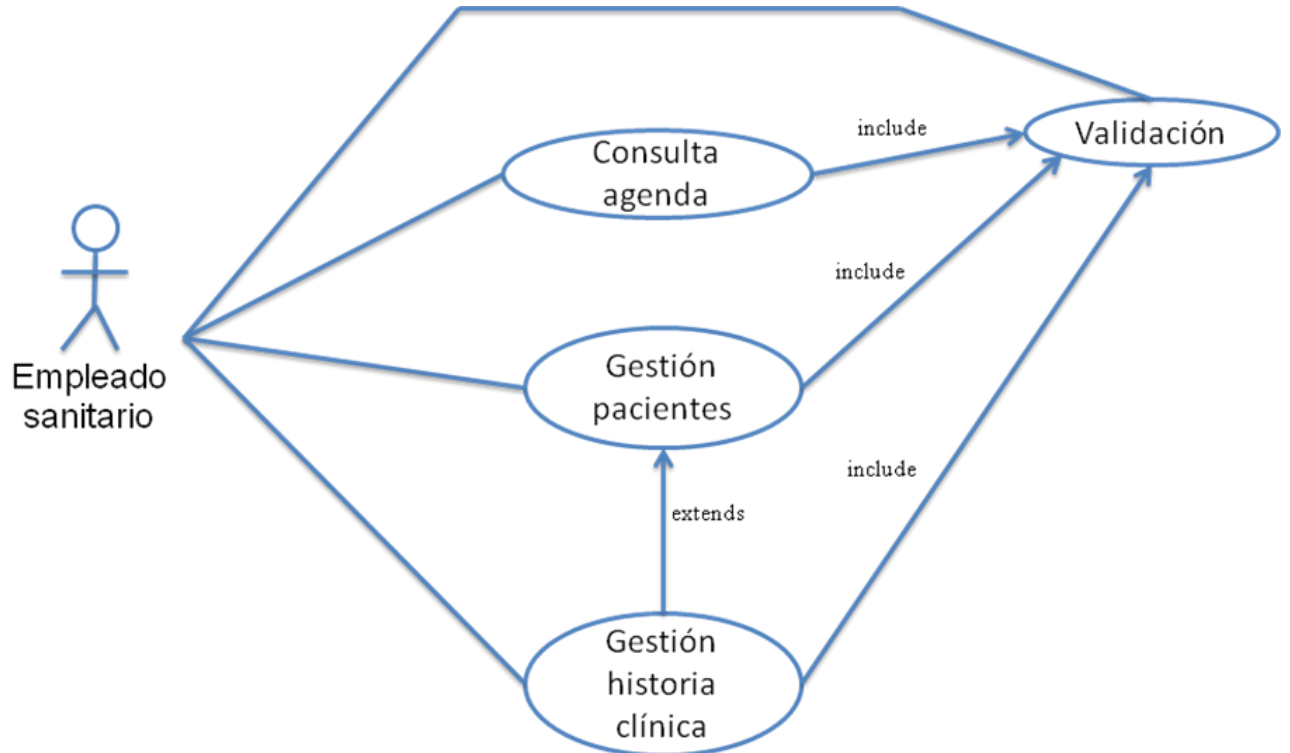


Figura 3.5. Funcionalidad general

Los usuarios sanitarios de esta aplicación disponen de acceso a las funcionalidades “Consultar agenda”, “Gestión paciente” y “Gestión historia clínica”, siempre que antes hayan superado la “Validación”.

Estos usuarios tendrán acceso directo entonces a la consulta de su propia agenda y a la gestión de los pacientes. Una vez se encuentren en la gestión de un paciente concreto, podrán tener acceso a la gestión de la historia clínica de éste, esto mismo sucede con la “Gestión cita”, que está directamente relacionada con estar en la gestión de un paciente concreto.

• Gestión del paciente:

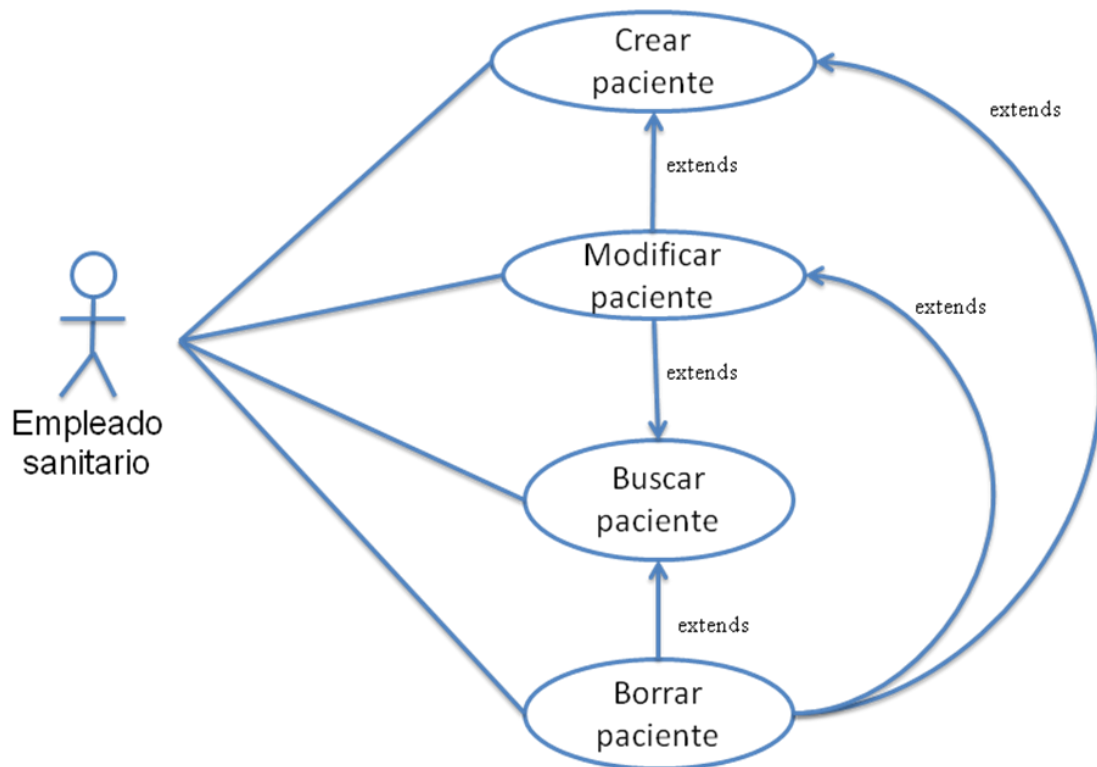


Figura 3.6. Funcionalidad “Gestión paciente”

Para la gestión de los pacientes por parte de los usuarios sanitarios tendrán acceso a las funcionalidades “Crear paciente”, “Modificar paciente”, “Buscar paciente” y “Borrar paciente”.

La funcionalidad de “Crear paciente” da lugar a poder “Modificar paciente” y “Borrar paciente”. Y las funcionalidades de “Modificar paciente” y “Borrar paciente” serán accesibles también a partir de encontrarse en la gestión de un paciente concreto.

• **Gestión de la historia clínica:**

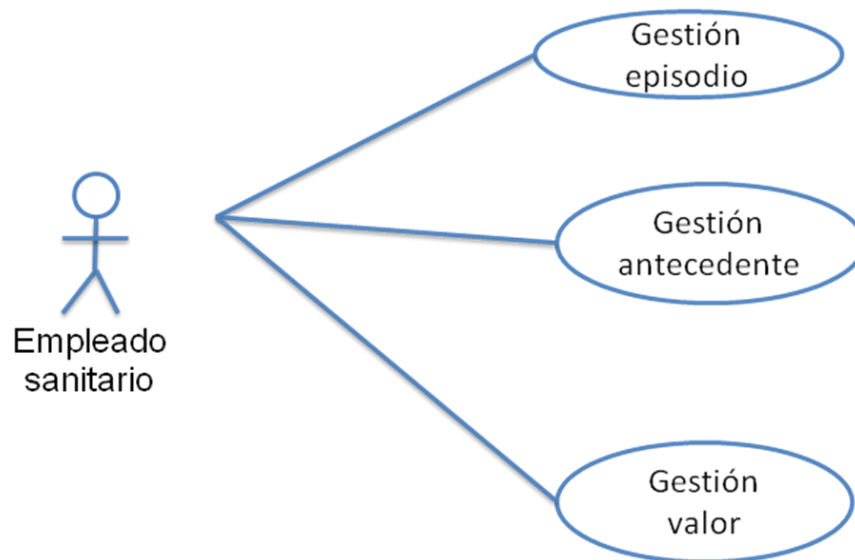


Figura 3.7. Funcionalidad “Gestión historia clínica”

La gestión de la historia clínica la realizan los usuarios sanitarios a partir de estar gestionando a un paciente.

La historia clínica la dividimos en 3 bloques: “Gestión episodio”, “Gestión antecedente” y “Gestión valor”.

• **Gestión de un episodio:**

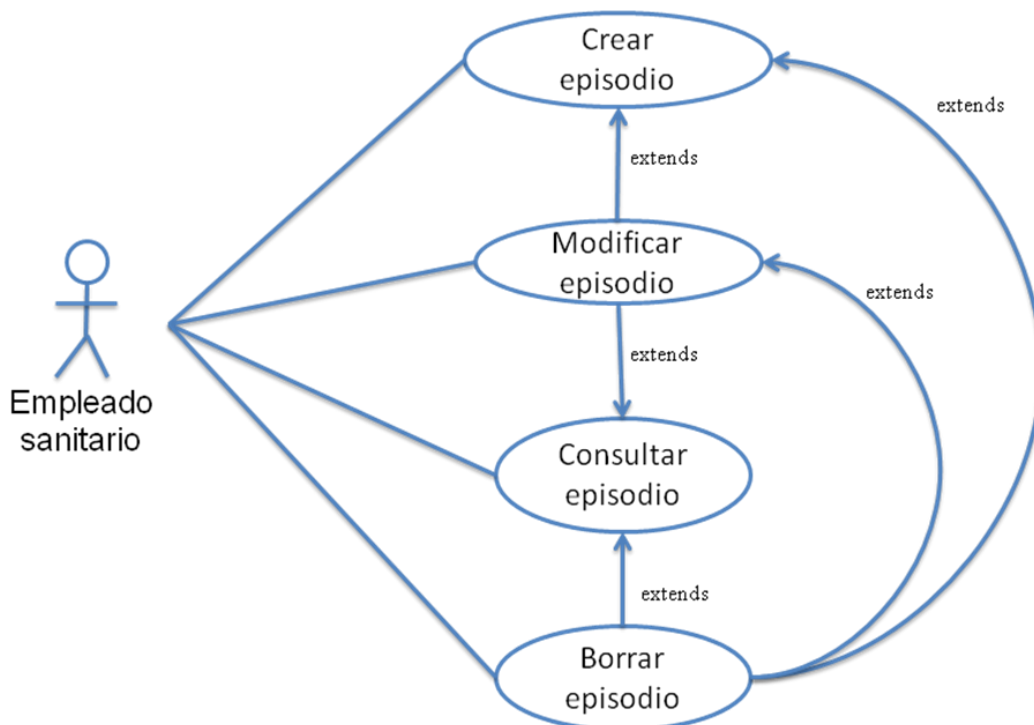


Figura 3.8. Funcionalidad “Gestión episodio”

La gestión de los episodios, tiene acceso a las funcionalidades “Crear episodio”, “Modificar episodio”, “Consultar episodio” y “Borrar episodio”.

La funcionalidad de “Crear episodio” da lugar a poder “Modificar episodio” y “Borrar episodio”. Y las funcionalidades de “Modificar episodio” y “Borrar episodio” serán accesibles también a partir de haber hecho la selección de un episodio concreto.

• **Gestión de antecedentes:**

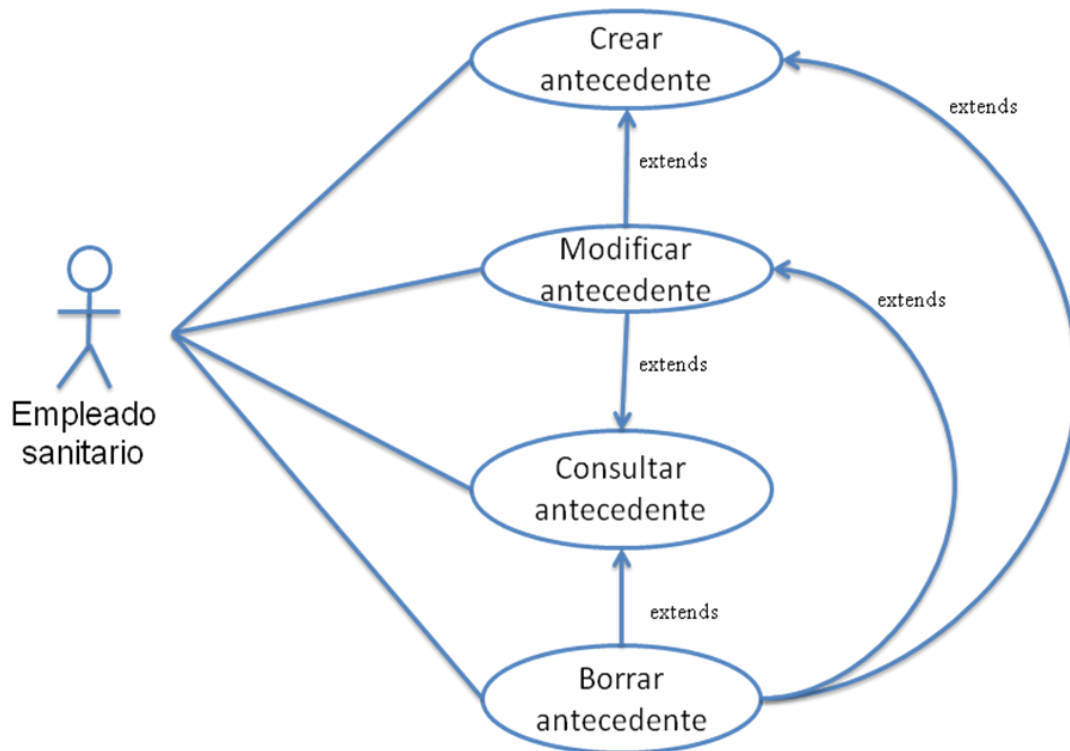
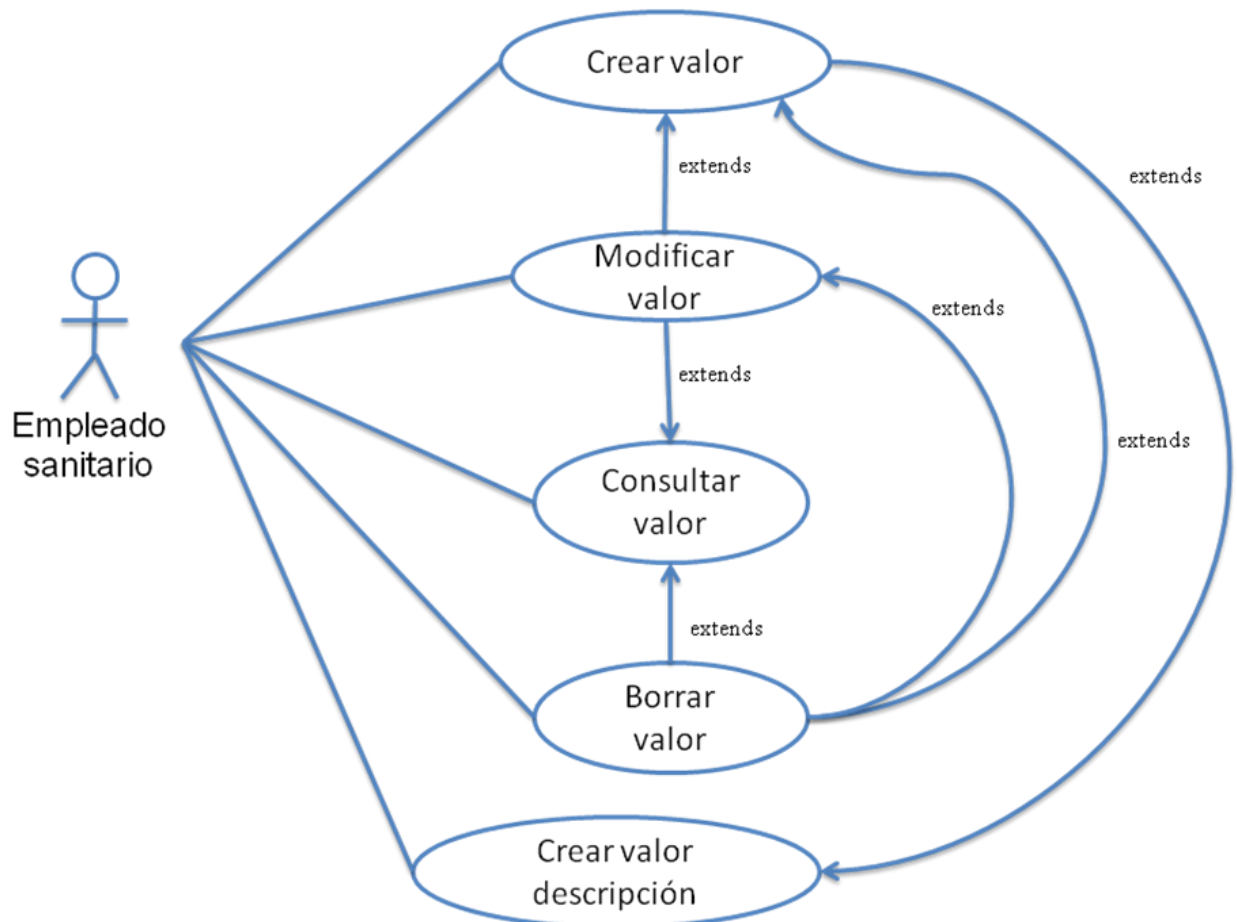


Figura 3.9. Funcionalidad “Gestión antecedente”

La gestión de los antecedentes, tiene acceso a las funcionalidades “Crear antecedente”, “Modificar antecedente”, “Consultar antecedente” y “Borrar antecedente”.

La funcionalidad de “Crear antecedente” da lugar a poder “Modificar antecedente” y “Borrar antecedente”. Y las funcionalidades de “Modificar antecedente” y “Borrar antecedente” serán accesibles también a partir de haber hecho la selección de un antecedente concreto.

• Gestión del valor:

**Figura 3.10. Funcionalidad “Gestión valor”**

La gestión de los valores, tiene acceso a las funcionalidades “Crear valor”, “Modificar valor”, “Consultar valor”, “Borrar valor” y “Crear valor descripción”.

La funcionalidad de “Crear valor” da lugar a poder “Modificar valor” y “Borrar valor”. Y las funcionalidades de “Modificar valor” y “Borrar valor” serán accesibles también a partir de haber hecho la selección de un valor concreto.

En los casos de “Crear valor” y “Modificar valor”, se tendrá acceso a “Crear valor descripción”.

- **Gestión de citas:**

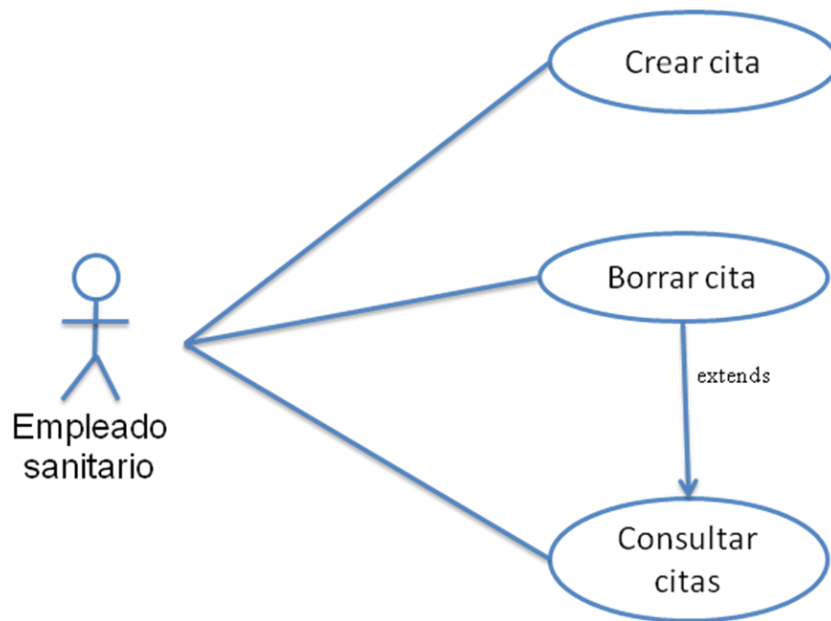


Figura 3.11. Funcionalidad “Gestión cita”

La gestión de las citas se realiza por los usuarios sanitarios, a partir de la gestión de un paciente en concreto y tiene las funcionalidades “Crear cita”, “Borrar cita” y “Consultar citas”.

En la gestión de citas los usuarios no sanitarios acceden a la funcionalidad de “Consultar citas”, y a partir de esta tienen acceso a las funcionalidades de “Crear cita” y “Borrar cita”.

3.1.2.3. Descripción de las funcionalidades

- **Validación**

El usuario tendrá que estar registrado en la base de datos y tendrá que introducir su usuario y contraseña, para comprobar que esté dado de alta en la base de datos y comprobar qué perfil tiene para que así la aplicación le redirija a su funcionalidad correspondiente.

- **Cerrar sesión**

El usuario se saldrá del contexto que creó al validarse en la aplicación y dejará limpia la sesión para próximas entradas desde esos mismo equipo.

- **Gestión empleado**

- * **Crear empleado**

Al crear un empleado, se generará un registro nuevo de empleado en la base de datos, junto con el que se guardarán los datos personales y de contacto de éste, a

su vez se guardará un nombre de usuario para sus posibles futuros accesos a la aplicación.

*** Buscar empleado**

Se realizará una búsqueda en la base de datos a partir de los criterios que se hayan introducido y se mostrarán los resultados por pantalla para la posible gestión de alguno de ellos.

*** Borrar empleado**

Se borrará el registro del empleado en la base de datos.

*** Consultar agenda**

Se realiza una consulta en la base de datos de las citas que tiene pendientes ese médico para ese día y se muestran por pantalla, también le permitirá consultar distintas fechas.

*** Modificar empleado**

Esta funcionalidad nos permite modificar todos los datos personales y de contacto del empleado, a excepción del usuario de acceso a la aplicación, que será único durante el tiempo que permanezca el empleado registrado en la aplicación.

• **Gestión paciente**

*** Crear paciente**

Se generará un registro nuevo de paciente en la base de datos, junto con el que se guardarán los datos personales y de contacto de éste.

*** Buscar paciente**

Se realizará una búsqueda en la base de datos a partir de los criterios que se hayan introducido y se mostrarán los resultados por pantalla para la posible gestión de alguno de ellos.

*** Borrar paciente**

Se borrará el registro del paciente en la base de datos.

*** Modificar paciente**

La modificación del paciente nos permite modificar cualquiera de los datos personales y de contacto del paciente. Y además nos permitirá el acceso a la gestión de su historia clínica y de sus citas.

• **Gestión historia clínica**

- **Gestión episodio**

*** Crear episodio**

Al crear un episodio en la base de datos se genera un registro relacionado con el paciente que estamos gestionando. Este episodio se forma con una fecha (normalmente la del día), un título y una descripción que es donde

se describirán las causas y síntomas por las que el paciente ha acudido a la consulta ese día.

*** Modificar episodio**

En la modificación del episodio se podrán modificar los datos del registro y que queden reflejados estos cambios en la base de datos.

*** Consultar episodio**

En la pantalla de “Gestión historia clínica” de continuo se mostrará un listado con todos los episodios del paciente, mostrando la fecha y el título de éste, y en este listado se puede seleccionar uno de ellos y consultarlo completo y a su vez tener la posibilidad de modificarlo.

*** Borrar episodio**

Cuando estamos consultando un episodio, tenemos la posibilidad de borrarlo, borrando así su registro en base de datos.

- **Gestión antecedente**

*** Crear antecedente**

Al crear un antecedente en la base de datos se genera un registro relacionado con el paciente que estamos gestionando. Este registro se compondrá de un título y una descripción del mismo.

*** Modificar antecedente**

En la modificación del antecedente se podrán modificar los datos del registro y que queden reflejados estos cambios en la base de datos.

*** Consultar antecedente**

En la pantalla de “Gestión historia clínica” de continuo se mostrará un listado con todos los antecedentes del paciente, mostrando el título de éste, y en este listado se puede seleccionar uno de ellos y consultarlo completo y a su vez tener la posibilidad de modificarlo.

*** Borrar antecedente**

Cuando estamos consultando un antecedente, tenemos la posibilidad de borrarlo, borrando así su registro en base de datos.

- **Gestión valor**

*** Crear valor**

Al crear un valor en la base de datos se genera un registro relacionado con el paciente que estamos gestionando. Este registro se compondrá de una descripción, un valor y la fecha en que fue tomado.

Crear valor descripción

Podremos crear una descripción de valor si no está ya registrada en la base de datos. Esta descripción lo que nos permite es registrar un nuevo tipo de valor.

*** Modificar valor**

En la modificación del valor se podrán modificar los datos del registro y que queden reflejados estos cambios en la base de datos.

*** Consultar valor**

En la pantalla de “Gestión historia clínica” de continuo se mostrará un listado con todos los valores del paciente, mostrando la descripción, el valor y la fecha en que se tomó, y en este listado se puede seleccionar uno de ellos y consultarlo completo y a su vez tener la posibilidad de modificarlo.

*** Borrar valor**

Cuando estamos consultando un antecedente, tenemos la posibilidad de borrarlo, borrando así su registro en base de datos.

• Gestión cita*** Crear cita**

Al crear una cita, se guardará en la base de datos un nuevo registro que lo que relaciona es al paciente con un sanitario en una fecha y hora concreta.

*** Borrar cita**

Al borrar una cita se eliminará este registro de la base de datos.

*** Consultar citas**

Se mostrarán por pantalla todas las citas de un mismo paciente.

3.1.3. Modelo relacional de base de datos

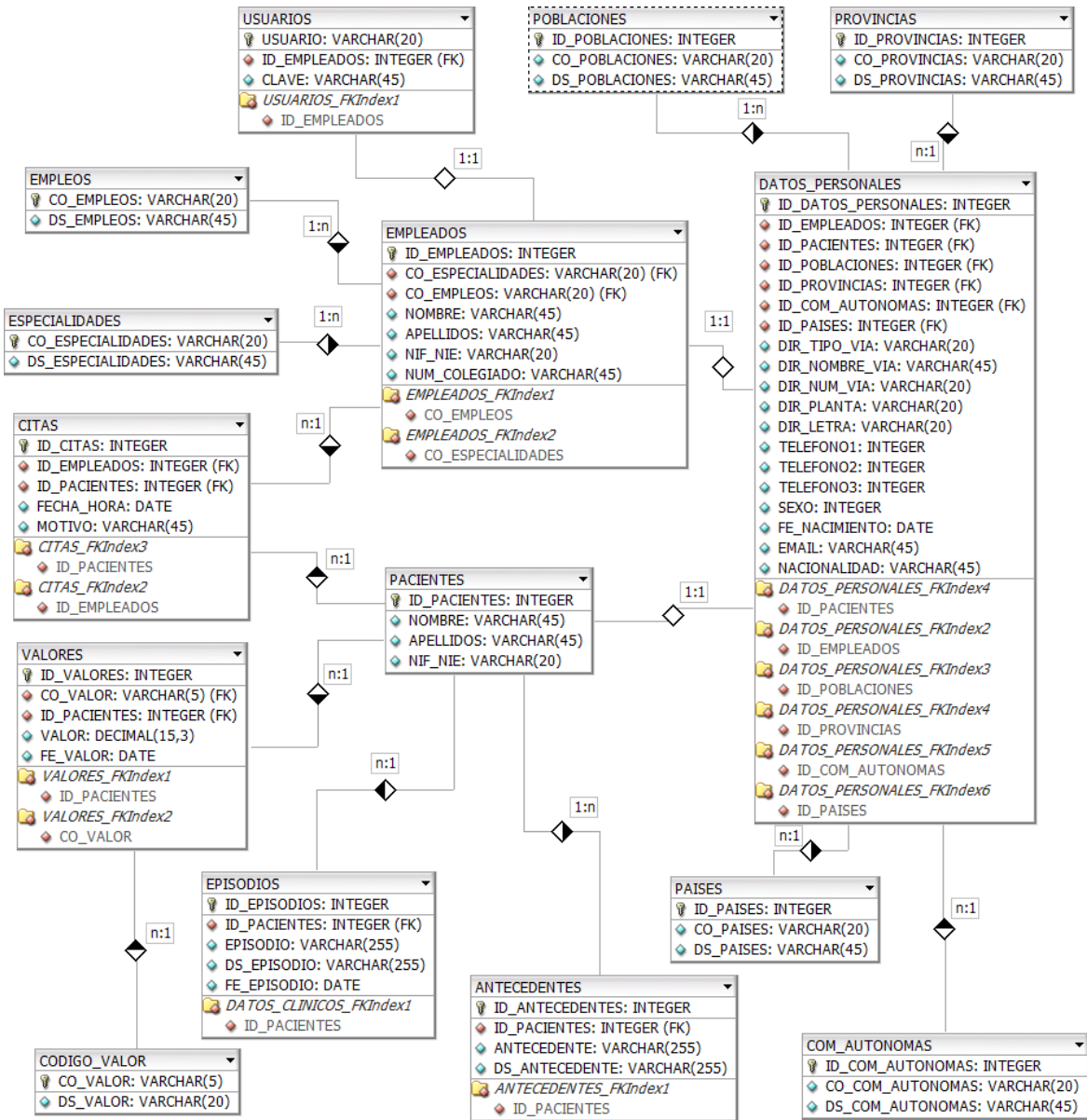


Figura 3.12. Esquema relacional de base de datos



Leyenda de la figura 3.12:

Cada rectángulo es una tabla

Clave primaria de la tabla

Campo que hace referencia a la clave primaria de otra tabla

Campo básico de una tabla

-  — Relación $1:n$ entre tablas, quiere decir que por cada registro de la tabla de la izquierda, puede haber n en la tabla de la derecha.
-  — Relación $1:1$ entre tablas, quiere decir, que por cada registro de una de las tablas, sólo existirá otro en la otra tabla.

Como vemos en la primera imagen de este apartado, nuestra base de datos está formada por 15 tablas relacionadas entre sí. Todas tienen una clave primaria, que utilizamos como nexo de relación entre ellas.

A continuación describimos las tablas que componen nuestra base de datos.

Tabla EMPLEOS

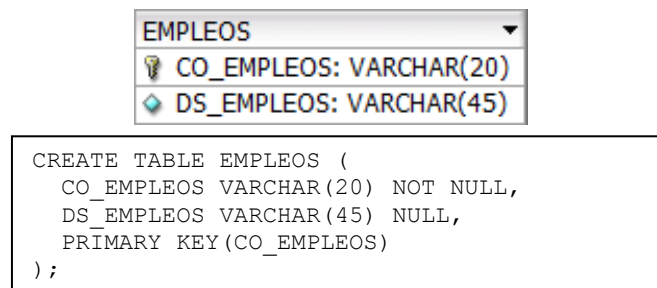


Figura 3.13. Tabla EMPLEOS y script de creación

La tabla EMPLEOS contiene todos los empleos que se pueden encontrar en el centro de salud. Se compone de los campos CO_EMPLEOS y DS_EMPLEOS.

El campo CO_EMPLEOS es la clave primaria de la tabla, es de tipo VARCHAR con capacidad de hasta 20 caracteres y contiene el código del campo al que harán referencia las tablas que estén enlazadas con ella. Este campo nunca podrá ser null.

El campo DS_EMPLEOS es de tipo VARCHAR con capacidad de hasta 45 caracteres y contiene la descripción del empleo, que será lo que se mostrará en las pantallas de la aplicación.

Los datos que se guardan aquí se guardarán directamente por base de datos, no son modificables por aplicación. Y los valores introducidos son los que mostramos en la siguiente tabla.

CO_EMPLEOS	DS_EMPLEOS
ADM	Administrativo
AUX	Auxiliar de enfermería
ENF	Enfermero
MED	Médico
FIS	Fisioterapeuta
POD	Podólogo

Tabla 3.1. Valores de la tabla EMPLEOS

Tabla ESPECIALIDADES

```
CREATE TABLE ESPECIALIDADES (
    CO_ESPECIALIDADES VARCHAR(20) NOT NULL,
    DS_ESPECIALIDADES VARCHAR(45) NULL,
    PRIMARY KEY(CO_ESPECIALIDADES)
);
```

Figura 3.14. Tabla ESPECIALIDADES y script de creación

La tabla ESPECIALIDADES contiene todas las especialidades de todos los empleos que se pueden encontrar en el centro de salud. Se compone de los campos CO_ESPECIALIDADES y DS_ESPECIALIDADES.

El campo CO_ESPECIALIDADES es la clave primaria de la tabla, es de tipo VARCHAR con capacidad de hasta 20 caracteres y contiene el código del campo al que harán referencia las tablas que estén enlazadas con ella. Este campo nunca podrá ser null.

El campo DS_ESPECIALIDADES es de tipo VARCHAR con capacidad de hasta 45 caracteres y contiene la descripción de la especialidad, que será lo que se mostrará en las pantallas de la aplicación.

Los datos que se guardan aquí se guardarán directamente por base de datos, no son modificables por aplicación. Y los valores introducidos son los que mostramos en la siguiente tabla.

CO_ESPECIALIDADES	DS_ESPECIALIDADES
DER	Dermatología
GIN	Ginecología
FAM	Medicina Familiar
OFT	Oftalmología
OTO	Otorrinolaringología
PED	Pediatría
TRA	Traumatología

Tabla 3.2. Valores de la tabla ESPECIALIDADES

Tabla CODIGO_VALOR

CODIGO_VALOR	
🔑	CO_VALOR: VARCHAR(5)
🔑	DS_VALOR: VARCHAR(20)

```
CREATE TABLE CODIGO_VALOR (
  CO_VALOR VARCHAR(5) NOT NULL,
  DS_VALOR VARCHAR(20) NULL,
  PRIMARY KEY (CO_VALOR)
);
```

Figura 3.15. Tabla CODIGO_VALOR y script de creación

La tabla CODIGO_VALOR contiene los posibles tipos de valores que se pueden guardar desde la aplicación en la pantalla de datos clínicos. Se compone de los campos CO_VALOR y DS_VALOR.

El campo CO_VALOR es la clave primaria de la tabla, es de tipo VARCHAR con capacidad de hasta 5 caracteres y contiene el código del campo al que harán referencia las tablas que estén enlazadas con ella. Este campo nunca podrá ser null.

El campo DS_VALOR es de tipo VARCHAR con capacidad de hasta 20 caracteres y contiene la descripción del valor.

En esta tabla podrán insertarse nuevos valores desde la pantalla de datos clínicos. Aunque en de momento tendremos cargados los que mostramos en la siguiente tabla.

CO_VALOR	DS_VALOR
ALT	Altura(m)
PESO	Peso(Kg)
TENS	Tensión(alta,baja)
GLUC	Glucosa(mg/dl)

Tabla 3.3. Valores de la tabla CODIGO_VALOR

Tabla POBLACIONES

POBLACIONES	
🔑	ID_POBLACIONES: INTEGER
🔑	CO_POBLACIONES: VARCHAR(20)
🔑	DS_POBLACIONES: VARCHAR(45)

```
CREATE TABLE POBLACIONES (
  ID_POBLACIONES INTEGER UNSIGNED NOT NULL AUTO_INCREMENT,
  CO_POBLACIONES VARCHAR(20) NULL,
  DS_POBLACIONES VARCHAR(45) NULL,
  PRIMARY KEY (ID_POBLACIONES)
);
```

Figura 3.16. Tabla POBLACIONES y script de creación

La tabla POBLACIONES contiene todas las poblaciones españolas. Se compone de los campos ID_POBLACIONES y DS_POBLACIONES.

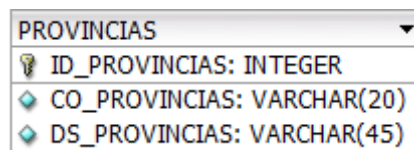
El campo ID_POBLACIONES es la clave primaria de la tabla, es de tipo INTEGER y contiene el identificador único de la población a la que harán referencia las tablas que estén enlazadas con ella. Este campo nunca podrá ser null y se autoincrementará al insertar un nuevo registro en la tabla.

El campo CO_POBLACIONES es de tipo VARCHAR con capacidad de hasta 20 caracteres y contiene el código de la población.

El campo DS_POBLACIONES es de tipo VARCHAR con capacidad de hasta 45 caracteres y contiene el nombre de la población, que será lo que se mostrará en las pantallas de la aplicación.

Los datos que se guardan aquí se guardarán directamente por base de datos, no son modificables por aplicación. Y los datos que se han insertado son los localizados en las tablas que facilita el INE (Instituto Nacional de Estadística) [8].

Tabla PROVINCIAS



PROVINCIAS
ID_PROVINCIAS: INTEGER
CO_PROVINCIAS: VARCHAR(20)
DS_PROVINCIAS: VARCHAR(45)

```
CREATE TABLE PROVINCIAS (  
  ID_PROVINCIAS INTEGER UNSIGNED NOT NULL AUTO_INCREMENT,  
  CO_PROVINCIAS VARCHAR(20) NULL,  
  DS_PROVINCIAS VARCHAR(45) NULL,  
  PRIMARY KEY (ID_PROVINCIAS)  
);
```

Figura 3.17. Tabla PROVINCIAS y script de creación

La tabla PROVINCIAS contiene todas las provincias españolas. Se compone de los campos ID_PROVINCIAS y DS_PROVINCIAS.

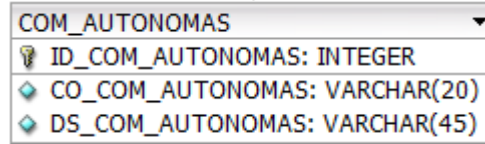
El campo ID_PROVINCIAS es la clave primaria de la tabla, es de tipo INTEGER y contiene el identificador único de la provincia a la que harán referencia las tablas que estén enlazadas con ella. Este campo nunca podrá ser null y se autoincrementará al insertar un nuevo registro en la tabla.

El campo CO_PROVINCIAS es de tipo VARCHAR con capacidad de hasta 20 caracteres y contiene el código de la provincia.

El campo DS_PROVINCIAS es de tipo VARCHAR con capacidad de hasta 45 caracteres y contiene el nombre de la provincia, que será lo que se mostrará en las pantallas de la aplicación.

Los datos que se guardan aquí se guardarán directamente por base de datos, no son modificables por aplicación. Y los datos que se han insertado son los localizados en las tablas que facilita el INE (Instituto Nacional de Estadística) [9].

Tabla COM_AUTONOMAS



Column Name	Data Type
ID_COM_AUTONOMAS	INTEGER
CO_COM_AUTONOMAS	VARCHAR(20)
DS_COM_AUTONOMAS	VARCHAR(45)

```
CREATE TABLE COM_AUTONOMAS (  
  ID_COM_AUTONOMAS INTEGER UNSIGNED NOT NULL AUTO_INCREMENT,  
  CO_COM_AUTONOMAS VARCHAR(20) NULL,  
  DS_COM_AUTONOMAS VARCHAR(45) NULL,  
  PRIMARY KEY (ID_COM_AUTONOMAS)  
);
```

Figura 3.18. Tabla COM_AUTONOMAS y script de creación

La tabla COM_AUTONOMAS contiene todas las comunidades autónomas españolas. Se compone de los campos ID_COM_AUTONOMAS y DS_COM_AUTONOMAS.

El campo ID_COM_AUTONOMAS es la clave primaria de la tabla, es de tipo INTEGER y contiene el identificador único de la comunidad autónoma a la que harán referencia las tablas que estén enlazadas con ella. Este campo nunca podrá ser null y se autoincrementará al insertar un nuevo registro en la tabla.

El campo CO_COM_AUTONOMAS es de tipo VARCHAR con capacidad de hasta 20 caracteres y contiene el código de la comunidad autónoma.

El campo DS_COM_AUTONOMAS es de tipo VARCHAR con capacidad de hasta 45 caracteres y contiene el nombre de la comunidad autónoma, que será lo que se mostrará en las pantallas de la aplicación.

Los datos que se guardan aquí se guardarán directamente por base de datos, no son modificables por aplicación. Y los datos que se han insertado son los localizados en las tablas que facilita el INE (Instituto Nacional de Estadística) [10].

Tabla PAISES

PAISES	
ID_PAISES	INTEGER
CO_PAISES	VARCHAR(20)
DS_PAISES	VARCHAR(45)

```
CREATE TABLE PAISES (
  ID_PAISES INTEGER UNSIGNED NOT NULL AUTO_INCREMENT,
  CO_PAISES VARCHAR(20) NULL,
  DS_PAISES VARCHAR(45) NULL,
  PRIMARY KEY (ID_PAISES)
);
```

Figura 3.19. Tabla PAISES y script de creación

La tabla PAISES contiene todos los países. Se compone de los campos ID_PAISES y DS_PAISES.

El campo ID_PAISES es la clave primaria de la tabla, es de tipo INTEGER y contiene el identificador único del país a la que harán referencia las tablas que estén enlazadas con ella. Este campo nunca podrá ser null y se autoincrementará al insertar un nuevo registro en la tabla.

El campo CO_PAISES es de tipo VARCHAR con capacidad de hasta 20 caracteres y contiene el código del país.

El campo DS_PAISES es de tipo VARCHAR con capacidad de hasta 45 caracteres y contiene el nombre del país, que será lo que se mostrará en las pantallas de la aplicación.

Los datos que se guardan aquí se guardarán directamente por base de datos, no son modificables por aplicación. Y los datos que se han insertado son los localizados en las tablas que facilita el INE (Instituto Nacional de Estadística) [11].

Tabla PACIENTES

PACIENTES	
ID_PACIENTES	INTEGER
NOMBRE	VARCHAR(45)
APELLIDOS	VARCHAR(45)
NIF_NIE	VARCHAR(20)

```
CREATE TABLE PACIENTES (
  ID_PACIENTES INTEGER UNSIGNED NOT NULL AUTO_INCREMENT,
  NOMBRE VARCHAR(45) NULL,
  APELLIDOS VARCHAR(45) NULL,
  NIF_NIE VARCHAR(20) NULL,
  PRIMARY KEY (ID_PACIENTES)
);
```

Figura 3.20. Tabla PACIENTES y script de creación

La tabla PACIENTES contiene los principales datos de identificación de un paciente. Se compone de los campos ID_PACIENTES, NOMBRE, APELLIDOS y NIF_NIE.

El campo ID_PACIENTES es la clave primaria de la tabla, es de tipo INTEGER y contiene el identificador único del paciente al que harán referencia las tablas que estén enlazadas con ella. Este campo nunca podrá ser null y se autoincrementará al insertar un nuevo registro en la tabla.

El campo NOMBRE es de tipo VARCHAR con capacidad de hasta 45 caracteres y contiene el nombre del paciente.

El campo APELLIDOS es de tipo VARCHAR con capacidad de hasta 45 caracteres y contiene los apellidos del paciente.

El campo NIF_NIE es de tipo VARCHAR con capacidad de hasta 20 caracteres y contiene el NIF o NIE del paciente.

En esta tabla se insertará un registro a la hora de crear un nuevo paciente a través de la aplicación en la pantalla de datos personales. Se podrán modificar todos los campos a excepción de la clave primaria que sólo se podrá eliminar junto con todo el registro, todas estas acciones serán posibles desde la pantalla de datos personales.

Tabla ANTECEDENTES

Field Name	Data Type	Constraints
ID_ANTECEDENTES	INTEGER	Primary Key
ID_PACIENTES	INTEGER	Foreign Key
ANTECEDENTE	VARCHAR(255)	
DS_ANTECEDENTE	VARCHAR(255)	
ANTECEDENTES_FKIndex1	INDEX	Index on ID_PACIENTES

```
CREATE TABLE ANTECEDENTES (
  ID_ANTECEDENTES INTEGER UNSIGNED NOT NULL AUTO_INCREMENT,
  ID_PACIENTES INTEGER UNSIGNED NOT NULL,
  ANTECEDENTE VARCHAR(255) NULL,
  DS_ANTECEDENTE VARCHAR(255) NULL,
  PRIMARY KEY(ID_ANTECEDENTES),
  INDEX ANTECEDENTES_FKIndex1(ID_PACIENTES),
  FOREIGN KEY(ID_PACIENTES)
    REFERENCES PACIENTES(ID_PACIENTES)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION
);
```

Figura 3.21. Tabla ANTECEDENTES y script de creación

La tabla ANTECEDENTES contiene los antecedentes de los pacientes. Se compone de los campos ID_ANTECEDENTES, ID_PACIENTES, ANTECEDENTE y DS_ANTECEDENTE.

El campo ID_ANTECEDENTES es la clave primaria de la tabla, es de tipo INTEGER y contiene el identificador único del antecedente al que harán referencia las tablas que

puedan estar enlazadas con ella. Este campo nunca podrá ser null y se autoincrementará al insertar un nuevo registro en la tabla.

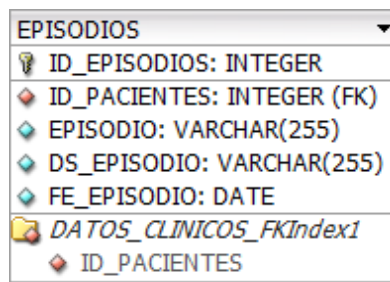
El campo ID_PACIENTES es una clave foránea, que relaciona esta tabla con PACIENTES. Un antecedente siempre pertenece a un paciente, por lo que este campo nunca podrá ser null.

El campo ANTECEDENTE es de tipo VARCHAR con capacidad de hasta 255 caracteres y contiene una breve descripción del antecedente, que podríamos llamar título.

El campo DS_ANTECEDENTE es de tipo VARCHAR con capacidad de hasta 255 caracteres y contiene la descripción del antecedente.

En esta tabla se inserta un registro nuevo al crear un antecedente nuevo en la pantalla de datos clínicos. Y sus campos serán modificables desde la misma pantalla, a excepción de la clave primaria que sólo podrá ser eliminada junto con el registro completo desde esta misma página.

Tabla EPISODIOS



Column Name	Data Type	Constraints
ID_EPISODIOS	INTEGER	Primary Key
ID_PACIENTES	INTEGER	Foreign Key
EPISODIO	VARCHAR(255)	
DS_EPISODIO	VARCHAR(255)	
FE_EPISODIO	DATE	
Index: DATOS_CLINICOS_FKIndex1 (on ID_PACIENTES)		

```
CREATE TABLE EPISODIOS (  
    ID_EPISODIOS INTEGER UNSIGNED NOT NULL AUTO_INCREMENT,  
    ID_PACIENTES INTEGER UNSIGNED NOT NULL,  
    EPISODIO VARCHAR(255) NULL,  
    DS_EPISODIO VARCHAR(255) NULL,  
    FE_EPISODIO DATE NULL,  
    PRIMARY KEY (ID_EPISODIOS),  
    INDEX DATOS_CLINICOS_FKIndex1 (ID_PACIENTES),  
    FOREIGN KEY (ID_PACIENTES)  
        REFERENCES PACIENTES (ID_PACIENTES)  
        ON DELETE NO ACTION  
        ON UPDATE NO ACTION  
);
```

Figura 3.22. Tabla EPISODIOS y script de creación

La tabla EPISODIOS contiene los episodios de los pacientes. Se compone de los campos ID_EPISODIOS, ID_PACIENTES, EPISODIO, DS_EPISODIO y FE_EPISODIO.

El campo ID_EPISODIOS es la clave primaria de la tabla, es de tipo INTEGER y contiene el identificador único del episodio al que harán referencia las tablas que

puedan estar enlazadas con ella. Este campo nunca podrá ser null y se autoincrementará al insertar un nuevo registro en la tabla.

El campo ID_PACIENTES es una clave foránea, que relaciona esta tabla con PACIENTES. Un episodio siempre pertenece a un paciente, por lo que este campo nunca podrá ser null.

El campo EPISODIO es de tipo VARCHAR con capacidad de hasta 255 caracteres y contiene una breve descripción del episodio, que podríamos llamar título.

El campo DS_EPISODIO es de tipo VARCHAR con capacidad de hasta 255 caracteres y contiene la descripción del episodio.

El campo FE_EPISODIO es de tipo Date y contendrá la fecha en la que se ha producido el episodio.

En esta tabla se inserta un registro nuevo al crear un episodio nuevo en la pantalla de datos clínicos. Y sus campos serán modificables desde la misma pantalla, a excepción de la clave primaria que sólo podrá ser eliminada junto con el registro completo desde esta misma página.

Tabla VALORES

Column Name	Data Type	Constraints
ID_VALORES	INTEGER	Primary Key, Auto-Increment
CO_VALOR	VARCHAR(5)	Foreign Key (FK)
ID_PACIENTES	INTEGER	Foreign Key (FK)
VALOR	DECIMAL(15,3)	
FE_VALOR	DATE	

Indexes:

- VALORES_FKIndex1: Foreign Key Index on ID_PACIENTES
- VALORES_FKIndex2: Foreign Key Index on CO_VALOR

```
CREATE TABLE VALORES (
  ID_VALORES INTEGER UNSIGNED NOT NULL AUTO_INCREMENT,
  CO_VALOR VARCHAR(5) NOT NULL,
  ID_PACIENTES INTEGER UNSIGNED NOT NULL,
  VALOR DECIMAL(15,3) NULL,
  FE_VALOR DATE NULL,
  PRIMARY KEY (ID_VALORES),
  INDEX VALORES_FKIndex1 (ID_PACIENTES),
  INDEX VALORES_FKIndex2 (CO_VALOR),
  FOREIGN KEY (ID_PACIENTES)
    REFERENCES PACIENTES (ID_PACIENTES)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  FOREIGN KEY (CO_VALOR)
    REFERENCES CÓDIGO_VALOR (CO_VALOR)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION
);
```

Figura 3.23. Tabla VALORES y script de creación

La tabla VALORES contiene los valores tomados a los pacientes. Se compone de los campos ID_VALORES, ID_PACIENTES, CO_VALOR, VALOR y FE_VALOR.

El campo ID_VALORES es la clave primaria de la tabla, es de tipo INTEGER y contiene el identificador único del valor al que harán referencia las tablas que puedan estar enlazadas con ella. Este campo nunca podrá ser null y se autoincrementará al insertar un nuevo registro en la tabla.

El campo ID_PACIENTES es una clave foránea, que relaciona esta tabla con PACIENTES. Un valor siempre pertenece a un paciente, por lo que este campo nunca podrá ser null.

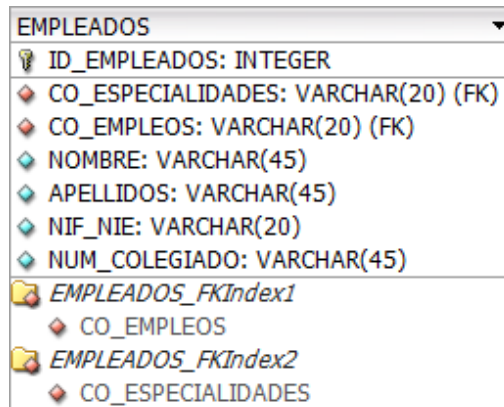
El campo CO_VALOR es una clave foránea, que relaciona esta tabla con CODIGO_VALOR. Un valor siempre estará relacionado con un tipo de valor, por lo que este campo nunca podrá ser null.

El campo VALOR es de tipo DECIMAL pudiendo tener hasta 15 cifras en la parte entera y 3 en la parte decimal. Contiene el valor tomado.

El campo FE_VALOR es de tipo Date y contendrá la fecha en la que se ha tomado el valor.

En esta tabla se inserta un registro nuevo al crear un valor nuevo en la pantalla de datos clínicos. Y sus campos serán modificables desde la misma pantalla, a excepción de la clave primaria que sólo podrá ser eliminada junto con el registro completo desde esta misma página.

Tabla EMPLEADOS



Field Name	Field Type	Field Properties
ID_EMPLEADOS	INTEGER	Primary Key
CO_ESPECIALIDADES	VARCHAR(20)	Foreign Key (FK)
CO_EMPLEOS	VARCHAR(20)	Foreign Key (FK)
NOMBRE	VARCHAR(45)	
APELLIDOS	VARCHAR(45)	
NIF_NIE	VARCHAR(20)	
NUM_COLEGIADO	VARCHAR(45)	
EMPLEADOS_FKIndex1		Foreign Key Index (FK)
CO_EMPLEOS		Foreign Key (FK)
EMPLEADOS_FKIndex2		Foreign Key Index (FK)
CO_ESPECIALIDADES		Foreign Key (FK)

```
CREATE TABLE EMPLEADOS (
  ID_EMPLEADOS INTEGER UNSIGNED NOT NULL AUTO_INCREMENT,
  CO_ESPECIALIDADES VARCHAR(20) NULL,
  CO_EMPLEOS VARCHAR(20) NOT NULL,
  NOMBRE VARCHAR(45) NULL,
  APELLIDOS VARCHAR(45) NULL,
  NIF_NIE VARCHAR(20) NULL,
  NUM_COLEGIADO VARCHAR(45) NULL,
  PRIMARY KEY(ID_EMPLEADOS),
  INDEX EMPLEADOS_FKIndex1(CO_EMPLEOS),
  INDEX EMPLEADOS_FKIndex2(CO_ESPECIALIDADES),
  FOREIGN KEY(CO_EMPLEOS)
    REFERENCES EMPLEOS(CO_EMPLEOS)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  FOREIGN KEY(CO_ESPECIALIDADES)
    REFERENCES ESPECIALIDADES(CO_ESPECIALIDADES)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION
);
```

Figura 3.24. Tabla EMPLEADOS y script de creación

La tabla EMPLEADOS contiene los principales datos de identificación de un empleado. Se compone de los campos ID_EMPLEADOS, CO_ESPECIALIDADES, CO_EMPLEOS, NOMBRE, APELLIDOS, NIF_NIE y NUM_COLEGIADO.

El campo ID_EMPLEADOS es la clave primaria de la tabla, es de tipo INTEGER y contiene el identificador único del empleado al que harán referencia las tablas que estén enlazadas con ella. Este campo nunca podrá ser null y se autoincrementará al insertar un nuevo registro en la tabla.

El campo CO_ESPECIALIDADES es una clave foránea, que relaciona esta tabla con ESPECIALIDADES. Un empleado puede estar relacionado con una especialidad, pero no es imprescindible.

El campo CO_EMPLEOS es una clave foránea, que relaciona esta tabla con EMPLEOS. Un empleado siempre estará relacionado con un tipo de empleo, por lo que este campo nunca podrá ser null.

El campo NOMBRE es de tipo VARCHAR con capacidad de hasta 45 caracteres y contiene el nombre del empleado.

El campo APELLIDOS es de tipo VARCHAR con capacidad de hasta 45 caracteres y contiene los apellidos del empleado.

El campo NIF_NIE es de tipo VARCHAR con capacidad de hasta 20 caracteres y contiene el NIF o NIE del empleado.

El campo NUM_COLEGIADO es de tipo VARCHAR con capacidad de hasta 45 caracteres y contiene el número de colegiado del empleado.

En esta tabla se insertará un registro a la hora de crear un nuevo empleado a través de la aplicación en la pantalla de datos personales. Se podrán modificar todos los campos a excepción de la clave primaria que sólo se podrá eliminar junto con todo el registro. Todas estas acciones serán posibles desde la pantalla de datos personales.

Tabla DATOS_PERSONALES

DATOS_PERSONALES	
ID_DATOS_PERSONALES	INTEGER
ID_EMPLEADOS	INTEGER (FK)
ID_PACIENTES	INTEGER (FK)
ID_POBLACIONES	INTEGER (FK)
ID_PROVINCIAS	INTEGER (FK)
ID_COM_AUTONOMAS	INTEGER (FK)
ID_PAISES	INTEGER (FK)
DIR_TIPO_VIA	VARCHAR(20)
DIR_NOMBRE_VIA	VARCHAR(45)
DIR_NUM_VIA	VARCHAR(20)
DIR_PLANTA	VARCHAR(20)
DIR_LETRA	VARCHAR(20)
TELEFONO1	INTEGER
TELEFONO2	INTEGER
TELEFONO3	INTEGER
SEXO	INTEGER
FE_NACIMIENTO	DATE
EMAIL	VARCHAR(45)
NACIONALIDAD	VARCHAR(45)
<i>DATOS_PERSONALES_FKIndex4</i>	
ID_PACIENTES	
<i>DATOS_PERSONALES_FKIndex2</i>	
ID_EMPLEADOS	
<i>DATOS_PERSONALES_FKIndex3</i>	
ID_POBLACIONES	
<i>DATOS_PERSONALES_FKIndex4</i>	
ID_PROVINCIAS	
<i>DATOS_PERSONALES_FKIndex5</i>	
ID_COM_AUTONOMAS	
<i>DATOS_PERSONALES_FKIndex6</i>	
ID_PAISES	

```

CREATE TABLE DATOS_PERSONALES (
  ID_DATOS_PERSONALES INTEGER UNSIGNED NOT NULL AUTO_INCREMENT,
  ID_EMPLEADOS INTEGER UNSIGNED NULL,
  ID_PACIENTES INTEGER UNSIGNED NULL,
  ID_POBLACIONES INTEGER UNSIGNED NULL,
  ID_PROVINCIAS INTEGER UNSIGNED NULL,
  ID_COM_AUTONOMAS INTEGER UNSIGNED NULL,
  ID_PAISES INTEGER UNSIGNED NULL,
  DIR_TIPO_VIA VARCHAR(20) NULL,
  DIR_NOMBRE_VIA VARCHAR(45) NULL,
  DIR_NUM_VIA VARCHAR(20) NULL,
  DIR_PLANTA VARCHAR(20) NULL,
  DIR_LETRA VARCHAR(20) NULL,
  TELEFONO1 INTEGER UNSIGNED NULL,
  TELEFONO2 INTEGER UNSIGNED NULL,
  TELEFONO3 INTEGER UNSIGNED NULL,
  SEXO INTEGER UNSIGNED NULL,
  FE_NACIMIENTO DATE NULL,
  EMAIL VARCHAR(45) NULL,
  NACIONALIDAD VARCHAR(45) NULL,
  PRIMARY KEY(ID_DATOS_PERSONALES),
  INDEX DATOS_PERSONALES_FKIndex4 (ID_PACIENTES),
  INDEX DATOS_PERSONALES_FKIndex2 (ID_EMPLEADOS),
  INDEX DATOS_PERSONALES_FKIndex3 (ID_POBLACIONES),
  INDEX DATOS_PERSONALES_FKIndex4 (ID_PROVINCIAS),
  INDEX DATOS_PERSONALES_FKIndex5 (ID_COM_AUTONOMAS),
  INDEX DATOS_PERSONALES_FKIndex6 (ID_PAISES),
  FOREIGN KEY (ID_PACIENTES)
    REFERENCES PACIENTES (ID_PACIENTES)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  FOREIGN KEY (ID_EMPLEADOS)
    REFERENCES EMPLEADOS (ID_EMPLEADOS)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  FOREIGN KEY (ID_POBLACIONES)
    REFERENCES POBLACIONES (ID_POBLACIONES)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  FOREIGN KEY (ID_PROVINCIAS)
    REFERENCES PROVINCIAS (ID_PROVINCIAS)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  FOREIGN KEY (ID_COM_AUTONOMAS)
    REFERENCES COM_AUTONOMAS (ID_COM_AUTONOMAS)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  FOREIGN KEY (ID_PAISES)
    REFERENCES PAISES (ID_PAISES)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION
);

```

Figura 3.25. Tabla **DATOS_PERSONALES** y script de creación

La tabla **DATOS_PERSONALES** contiene los datos personales tanto de los empleados como de los pacientes. Se compone de los campos **ID_DATOS_PERSONALES**, **ID_EMPLEADOS**, **ID_PACIENTES**, **ID_POBLACIONES**, **ID_PROVINCIAS**, **ID_COM_AUTONOMAS**, **ID_PAISES**, **DIR_TIPO_VIA**, **DIR_NOMBRE_VIA**, **DIR_NUM_VIA**, **DIR_PLANTA**, **DIR_LETRA**, **TELEFONO1**, **TELEFONO2**, **TELEFONO3**, **SEXO**, **FE_NACIMIENTO**, **EMAIL** y **NACIONALIDAD**.

El campo `ID_DATOS_PERSONALES` es la clave primaria de la tabla, es de tipo `INTEGER` y contiene el identificador único del valor al que harán referencia las tablas que puedan estar enlazadas con ella. Este campo nunca podrá ser `null` y se autoincrementará al insertar un nuevo registro en la tabla.

El campo `ID_EMPLEADOS` es una clave foránea, que relaciona esta tabla con `EMPLEADOS`. Los datos personales pueden pertenecer a empleados o pacientes, por lo que este campo puede ser `null`.

El campo `ID_PACIENTES` es una clave foránea, que relaciona esta tabla con `PACIENTES`. Los datos personales pueden pertenecer a empleados o pacientes, por lo que este campo puede ser `null`.

El campo `ID_POBLACIONES` es una clave foránea, que relaciona esta tabla con `POBLACIONES`.

El campo `ID_PROVINCIAS` es una clave foránea, que relaciona esta tabla con `PROVINCIAS`.

El campo `ID_COM_AUTONOMAS` es una clave foránea, que relaciona esta tabla con `COM_AUTONOMAS`.

El campo `ID_PAISES` es una clave foránea, que relaciona esta tabla con `PAISES`.

El campo `DIR_TIPO_VIA` es de tipo `VARCHAR` con capacidad de hasta 20 caracteres y contiene el tipo de vía del domicilio (por ejemplo: calle, avenida...).

El campo `DIR_NOMBRE_VIA` es de tipo `VARCHAR` con capacidad de hasta 45 caracteres y contiene el nombre de la vía.

El campo `DIR_NUM_VIA` es de tipo `VARCHAR` con capacidad de hasta 20 caracteres y contiene el número de la vía.

El campo `DIR_PLANTA` es de tipo `VARCHAR` con capacidad de hasta 20 caracteres y contiene el número de planta del edificio.

El campo `DIR_LETRA` es de tipo `VARCHAR` con capacidad de hasta 20 caracteres y contiene la letra de la puerta.

Los campos `TELEFONO1`, `TELEFONO2` Y `TELEFONO3` son de tipo `INTEGER` y contienen los teléfonos de contacto.

El campo `SEXO` es de tipo `INTEGER` y contiene 1 si es mujer y 0 si es hombre.

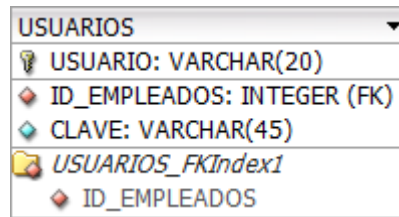
El campo `FE_NACIMIENTO` es de tipo `Date` y contendrá la fecha de nacimiento.

El campo `EMAIL` es de tipo `VARCHAR` con capacidad de hasta 45 caracteres y contiene la dirección de correo electrónico de contacto.

El campo NACIONALIDAD es de tipo VARCHAR con capacidad de hasta 45 caracteres y contiene la nacionalidad.

En esta tabla se inserta un registro nuevo al crear un registro nuevo tanto en la tabla PACIENTES como en la tabla EMPLEADOS desde la pantalla datos personales. Y sus campos serán modificables desde la misma pantalla, a excepción de la clave primaria que sólo podrá ser eliminada junto con el registro completo desde esta misma página.

Tabla USUARIOS



Column Name	Data Type	Constraints
USUARIO	VARCHAR(20)	Primary Key
ID_EMPLEADOS	INTEGER (FK)	Foreign Key to EMPLEADOS
CLAVE	VARCHAR(45)	

```
CREATE TABLE USUARIOS (  
    USUARIO VARCHAR(20) NOT NULL,  
    ID_EMPLEADOS INTEGER UNSIGNED NOT NULL,  
    CLAVE VARCHAR(45) NOT NULL,  
    PRIMARY KEY (USUARIO),  
    INDEX USUARIOS_FKIndex1 (ID_EMPLEADOS),  
    FOREIGN KEY (ID_EMPLEADOS)  
        REFERENCES EMPLEADOS (ID_EMPLEADOS)  
        ON DELETE NO ACTION  
        ON UPDATE NO ACTION  
);
```

Figura 3.26. Tabla USUARIOS y script de creación

La tabla USUARIOS contiene los identificadores y contraseñas de los usuarios de la aplicación. Se compone de los campos USUARIO, ID_EMPLEADOS y CLAVE.

El campo USUARIO es la clave primaria de la tabla, es de tipo VARCHAR con un máximo de 20 caracteres y contiene el identificador único del usuario. Este campo nunca podrá ser null.

El campo ID_EMPLEADOS es una clave foránea, que relaciona esta tabla con EMPLEADOS. Un usuario siempre está relacionado con un empleado, por lo que este campo nunca podrá ser null.

El campo CLAVE es de tipo VARCHAR con capacidad de hasta 45 caracteres y contiene la contraseña con la que deberá validarse el usuario para acceder a la aplicación. Esta contraseña estará guardada de forma cifrada, habiéndose creado a partir de pasar por una función MD5 la contraseña real, con esto conseguimos que no se pueda recuperar la contraseña original, aunque si un usuario la perdiera, deberá insertar una nueva por completo.

En esta tabla se inserta un registro nuevo al crear un empleado nuevo en la pantalla de datos personales.

Tabla CITAS

CITAS	
🔑	ID_CITAS: INTEGER
🔗	ID_EMPLEADOS: INTEGER (FK)
🔗	ID_PACIENTES: INTEGER (FK)
📅	FECHA_HORA: DATE
📄	MOTIVO: VARCHAR(45)
📁	CITAS_FKIndex3
🔗	ID_PACIENTES
📁	CITAS_FKIndex2
🔗	ID_EMPLEADOS

```

CREATE TABLE CITAS (
  ID_CITAS INTEGER UNSIGNED NOT NULL AUTO_INCREMENT,
  ID_EMPLEADOS INTEGER UNSIGNED NOT NULL,
  ID_PACIENTES INTEGER UNSIGNED NOT NULL,
  FECHA_HORA DATE NULL,
  MOTIVO VARCHAR(45) NULL,
  PRIMARY KEY (ID_CITAS),
  INDEX CITAS_FKIndex3 (ID_PACIENTES),
  INDEX CITAS_FKIndex2 (ID_EMPLEADOS),
  FOREIGN KEY (ID_PACIENTES)
    REFERENCES PACIENTES (ID_PACIENTES)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  FOREIGN KEY (ID_EMPLEADOS)
    REFERENCES EMPLEADOS (ID_EMPLEADOS)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION
);

```

Figura 3.27. Tabla CITAS y script de creación

La tabla CITAS contiene las citas de los empleados con los pacientes. Se compone de los campos ID_CITAS, ID_EMPLEADOS, ID_PACIENTES, FECHA_HORA y MOTIVO.

El campo ID_CITAS es la clave primaria de la tabla, es de tipo INTEGER y contiene el identificador único del valor al que harán referencia las tablas que puedan estar enlazadas con ella. Este campo nunca podrá ser null y se autoincrementará al insertar un nuevo registro en la tabla.

El campo ID_EMPLEADOS es una clave foránea, que relaciona esta tabla con EMPLEADOS. Las citas siempre estarán relacionando un empleado con un paciente, por lo que no puede ser null.

El campo ID_PACIENTES es una clave foránea, que relaciona esta tabla con PACIENTES. Las citas siempre estarán relacionando un empleado con un paciente, por lo que no puede ser null.

El campo FECHA_HORA es de tipo Date y contendrá la fecha de la cita.

El campo MOTIVO es de tipo VARCHAR con capacidad de hasta 45 caracteres y contiene una breve descripción del motivo por el que se solicita la cita. En esta tabla se inserta un registro nuevo al crear una cita desde la pantalla Citas.

3.1.4. Diagramas de flujo

En la siguiente figura contemplamos el mapa web para los usuarios de perfil no sanitario.

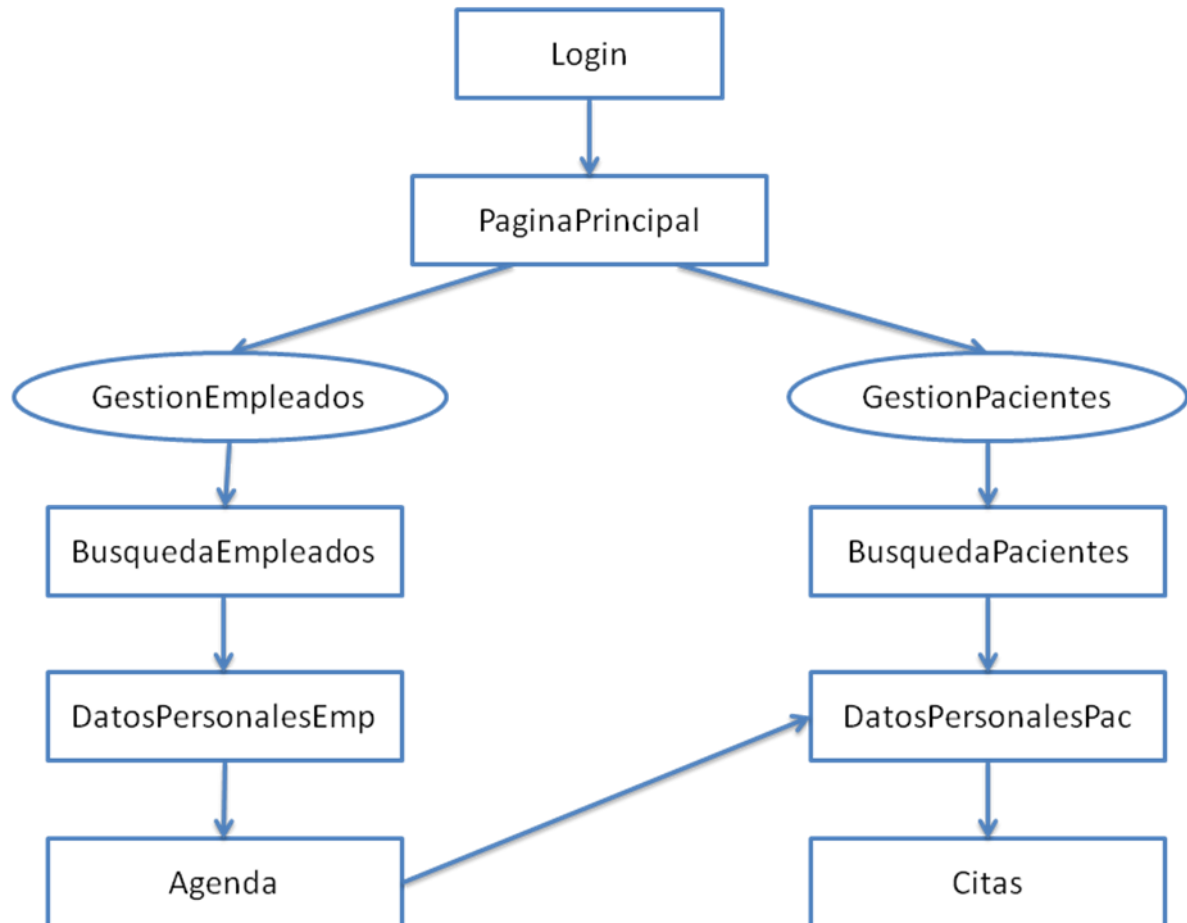


Figura 3.28. Mapa web perfil no sanitario

En esta figura se observa claramente cuáles son los posibles caminos que podrá seguir un usuario de perfil no sanitario a través de la aplicación, por sus distintas páginas.

Este usuario accederá a la página de “Login” de la aplicación, y si introduce sus datos correctos y pasa la validación llegará a la página principal donde tendrá acceso a la gestión de empleados y gestión de los pacientes, estas dos opciones, como veremos en el siguiente apartado, permanecerán accesibles desde cualquier pantalla, habiéndonos ya validado correctamente.

Si se elige la gestión de empleados, se podrá llegar a la pantalla de búsqueda de empleados, desde la que podremos acceder a la página de detalle de un empleado. Si

este empleado es del perfil sanitario, entonces podremos llegar también hasta la consulta de su agenda.

En la siguiente figura contemplamos el mapa web para los usuarios de perfil sanitario.

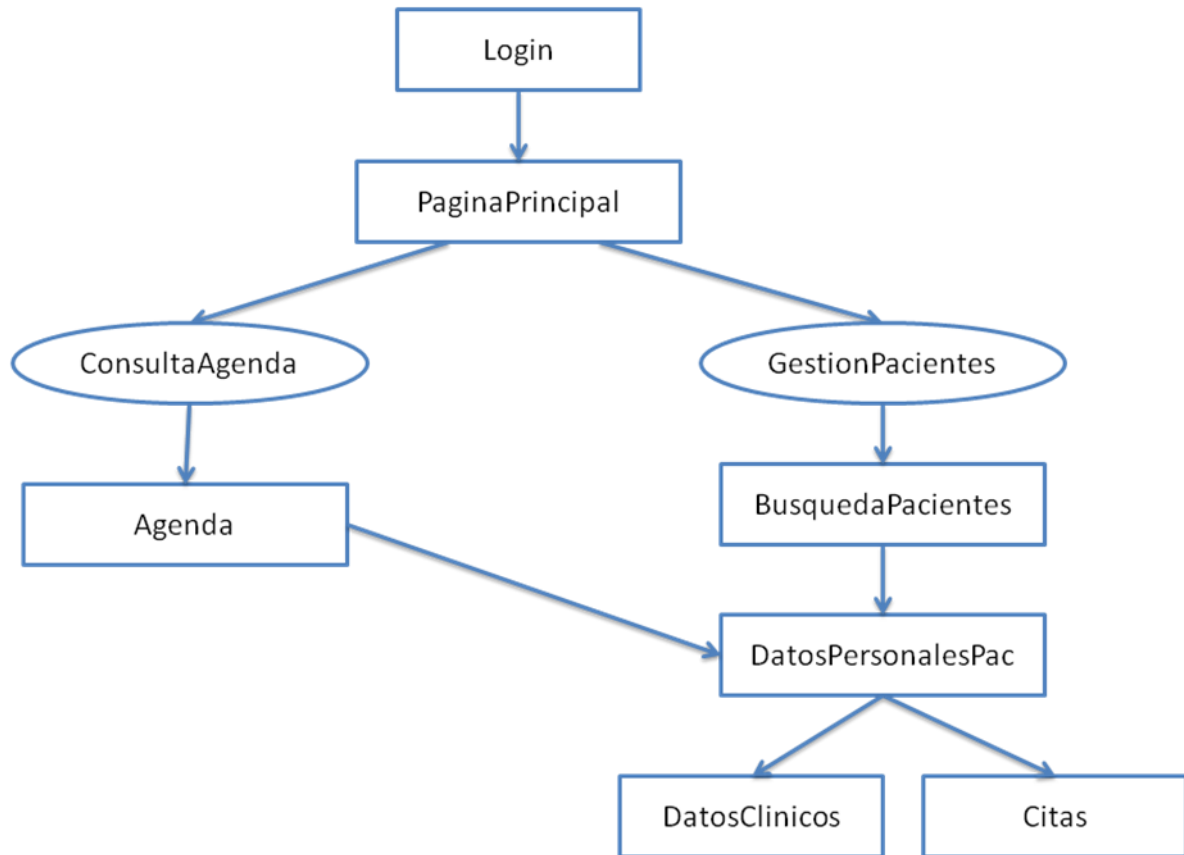


Figura 3.29. Mapa web perfil sanitario

Lo que se muestra en esta figura es que un usuario de perfil sanitario, tendrá que entrar antes de todo por la pantalla de “Login” para poder validarse y ya acceder a la página principal de este perfil de usuario.

Desde la página principal el empleado sanitario podrá acceder a la página donde consultar su agenda, que a su vez desde esta podrá acceder a la gestión de los pacientes que tenga citados, entrando directamente a la página de datos personales del paciente seleccionado.

Desde la página principal también podrá elegir gestionar a los pacientes directamente, por lo que pasará a la página de búsqueda de pacientes. Desde esta pasará a los datos personales de un paciente.

Desde la página de datos personales de un paciente, se tendrá acceso a sus datos clínicos y a sus citas.

3.2. Interfaz de usuario

En este apartado vamos a mostrar las pantallas a las que tiene acceso cada perfil de usuario.

Todas estas pantallas están desarrolladas en XHTML conjuntamente con CSS y utilizan las etiquetas que nos proporcionan las librerías de JSF y su componente Richfaces.

El diseño general de estas pantallas se basa en que todas estarán compuestas por un componente cabecera en el que se mostrará el logo de la aplicación, el nombre del usuario, su empleo y las opciones de cada perfil junto al botón de cierre de sesión común a todas las pantallas de un mismo perfil de usuario y el cuerpo como tal de la pantalla.



Figura 3.30. Diseño general de pantallas

Y una pantalla básica de acceso mediante usuario y contraseña común a todos los usuarios de la aplicación que mostramos a continuación.

Login



The screenshot shows a login interface with the following elements:

- Logo: **AGC** in a stylized pink font.
- Title: **APLICACIÓN GESTIÓN CLÍNICA** in pink.
- Form fields:
 - Label: **Usuario:**
 - Input:
 - Label: **Contraseña:**
 - Input:
- Button: **ACCEDER** in a pink box.

Figura 3.31. Pantalla de validación “Login”

En esta pantalla se validan los usuarios, si superan la validación pasarán a su pantalla de menú principal correspondiente a su tipo de perfil, si por lo contrario no han introducido bien el usuario y/o contraseña, se mostrará la siguiente pantalla y tendrán que volver a introducir los datos.



The screenshot shows an error message on the login interface with the following elements:

- Logo: **AGC** in a stylized pink font.
- Title: **APLICACIÓN GESTIÓN CLÍNICA** in pink.
- Form fields:
 - Label: **Usuario:**
 - Input:
 - Label: **Contraseña:**
 - Input:
- Button: **ACCEDER** in a pink box.
- Error message: **USUARIO Y/O CONTRASEÑA INCORRECTOS**

Figura 3.32. Pantalla de error de validación

3.2.1. Perfil no sanitario

Página principal



Figura 3.33. Pantalla "PaginaPrincipal"

Esta pantalla es la primera que le aparece al usuario de perfil no sanitario tras superar la validación. Y desde aquí tendrá la posibilidad de ir a la gestión de pacientes, gestión de empleados o cerrar sesión de usuario, las cuales se podrán realizar desde cualquiera de las pantallas siguientes.

Búsqueda de pacientes

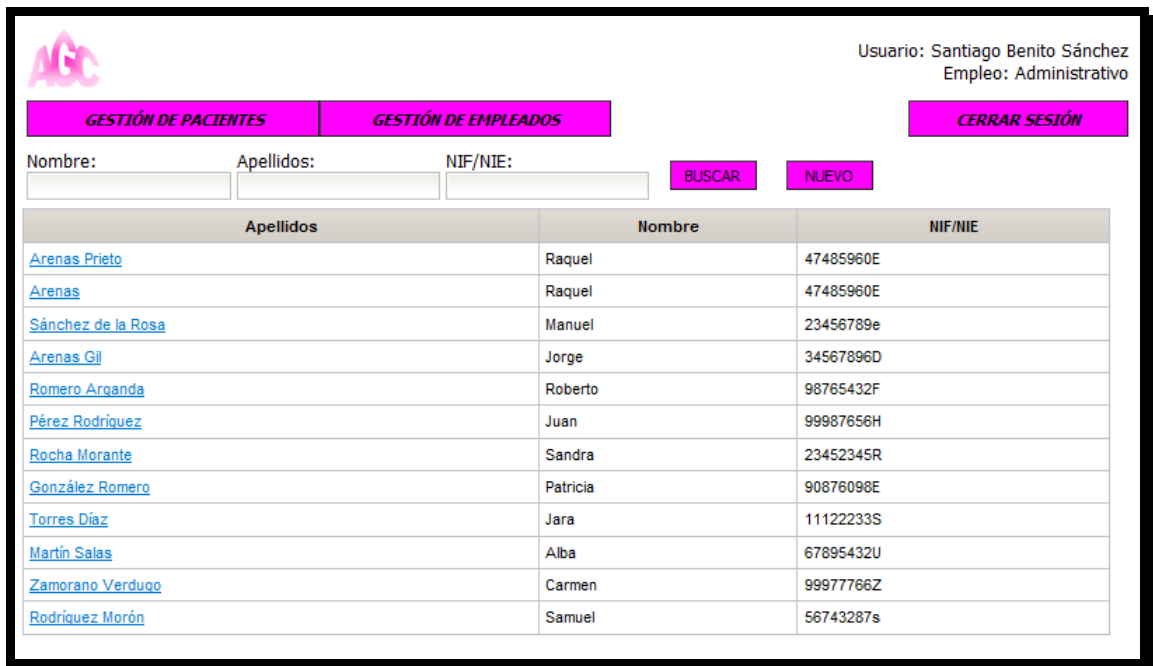


Figura 3.34. Pantalla "BusquedaPacientes"

Esta pantalla es la primera que aparece al pulsar “GESTIÓN DE PACIENTES”. Se utiliza para listar y filtrar los pacientes y desde ella podríamos ir a generar un nuevo paciente o consultar uno de la lista mediante su enlace.

Datos personales de paciente

AGC Usuario: Santiago Benito Sánchez
Empleo: Administrativo

GESTIÓN DE PACIENTES | **GESTIÓN DE EMPLEADOS** | **CERRAR SESIÓN**

Nombre: Apellidos: NIF/NIE:

Fecha Nacimiento: Sexo: Nacionalidad:

Tipo vía: Nombre vía: Núm.: Planta: Letra:

Población: Provincia:

Comunidad Autónoma: País:

Teléfono1: Teléfono2: Teléfono3:

Email:

GUARDAR | **CITAS** | **BORRAR**

Figura 3.35. Pantalla “DatosPersonalesPac”

Esta pantalla muestra los datos personales de un paciente. Desde ella se puede dar de alta uno nuevo, modificarlo, borrarlo y acceder a la gestión de sus citas.

Citas

AGC Usuario: Santiago Benito Sánchez
Empleo: Administrativo

GESTIÓN DE PACIENTES | **GESTIÓN DE EMPLEADOS** | **CERRAR SESIÓN**

Paciente: Patricia González Romero Edad: 4 Sexo: Mujer

Fecha/Hora	Empleo/Especialidad	Médico/Enfermero
oct 14, 2015 / 10:15	Médico / Medicina Familiar	Rubén soto Benito

Empleo: Especialidad: **BUSCAR** Fecha: **BUSCAR**

Nombre	Apellidos
No se han encontrado datos	

Horas	
<input type="radio"/>	10:00
<input type="radio"/>	10:15
<input type="radio"/>	10:30
<input type="radio"/>	10:45

GUARDAR

Figura 3.36. Pantalla “Citas”

En esta pantalla veremos arriba las citas que tenga pendientes el paciente. Y se puede buscar a un empleado para seleccionarlo junto con una hora y grabar una nueva cita.

Búsqueda de empleados

Usuario: Santiago Benito Sánchez
Empleo: Administrativo

GESTIÓN DE PACIENTES | **GESTIÓN DE EMPLEADOS** | **CERRAR SESIÓN**

Empleo: Todos | Nombre: m | Apellidos: | NIF/NIE: | **BUSCAR** | **NUEVO**

Apellidos	Nombre	NIF/NIE
Arenas Redondo	Mariano	46789034a
Cruz Vazquez	Remedios	
Aranda Gómez	Mara	
Prieto Elquido	Miguel	

Figura 3.37. Pantalla “BusquedaEmpleados”

Esta pantalla es la primera que aparece al pulsar “GESTIÓN DE EMPLEADOS”. Se utiliza para listar y filtrar los empleados y desde ella podríamos ir a generar un nuevo empleado o consultar uno de la lista mediante su enlace.

Datos personales de empleado

Usuario: Santiago Benito Sánchez
Empleo: Administrativo

GESTIÓN DE PACIENTES | **GESTIÓN DE EMPLEADOS** | **CERRAR SESIÓN**

Nombre: Santiago | Apellidos: Benito Sánchez | NIF/NIE: 02398722A
 Fecha Nacimiento: oct 7, 1981 | Sexo: Hombre | Nacionalidad: Español

Tipo vía: Calle | Nombre vía: Almendro | Núm.: 3 | Planta: 1 | Letra: D
 Población: Zamora | Provincia: Zamora | Comunidad Autónoma: Castilla y León | País: España
 Teléfono1: 980654789 | Teléfono2: 0 | Teléfono3: 0 | Email:

Empleo: Administrativo | Especialidad: Ninguna | Núm. Colegiado:
 Identificador Usuario: santiagoADM

GUARDAR | **AGENDA** | **BORRAR**

Figura 3.38. Pantalla “DatosPersonalesEmp”

Esta pantalla muestra los datos personales de un empleado. Desde ella se puede dar de alta uno nuevo, modificarlo, borrarlo y consultar su agenda.

Agenda

AGC

Usuario: Santiago Benito Sánchez
Empleo: Administrativo

GESTIÓN DE PACIENTES **GESTIÓN DE EMPLEADOS** **CERRAR SESIÓN**

Empleado: Rubén soto Benito Empleo/Esp: Médico/Medicina Familiar

Fecha:

Apellidos	Nombre	NIF/NIE
No se han encontrado datos		

Figura 3.39. Pantalla “Agenda”

En esta pantalla se mostrarán en un principio las citas que el empleado tiene para el día en curso, aunque cabe la posibilidad de buscar por otra fecha que se introduzca en el campo fecha. Y se podrá ir hasta la pantalla de datos personales de un paciente si pulsamos en su enlace.

3.2.2. Perfil sanitario

Página principal



Figura 3.40. Pantalla “PaginaPrincipal”

Esta pantalla es la primera que le aparece al usuario de perfil sanitario tras superar la validación. Y desde aquí tendrá la posibilidad de ir a la consulta de su agenda, la gestión de pacientes o cerrar sesión de usuario, las cuales se podrán realizar desde cualquiera de las pantallas siguientes.

Agenda

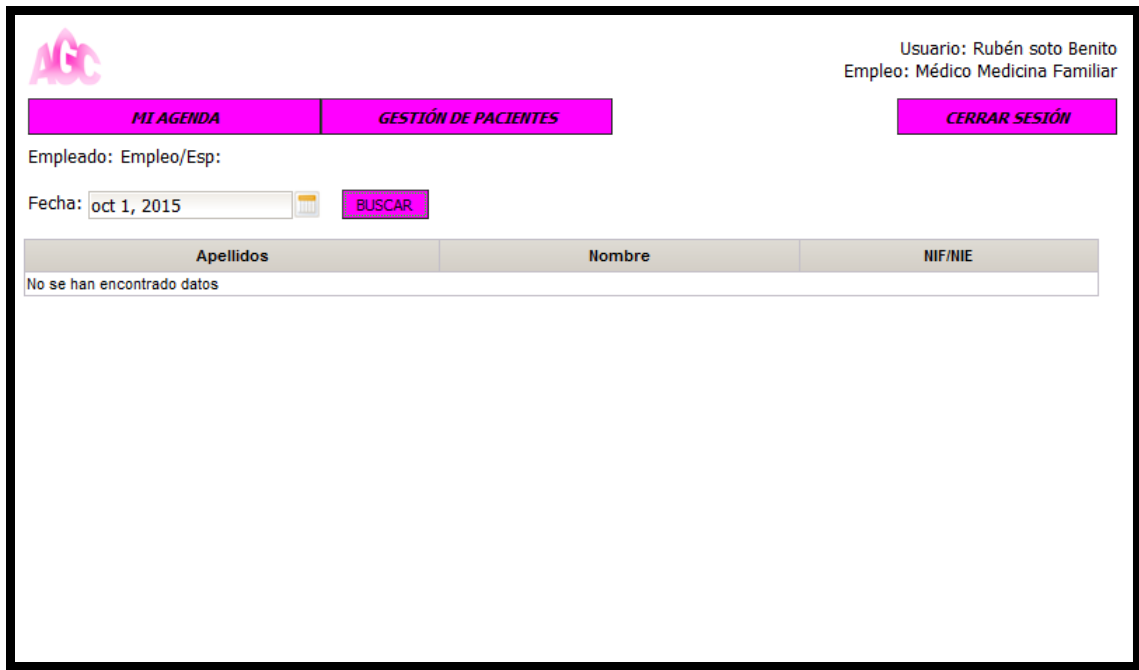


Figura 3.41. Pantalla “Agenda”

Esta pantalla aparecerá al pulsar el botón “MI AGENDA” y se mostrarán en un principio las citas que el usuario tiene para el día en curso, aunque cabe la posibilidad de buscar por otra fecha que se introduzca en el campo fecha. Y se podrá ir hasta la pantalla de datos personales de un paciente si pulsamos en su enlace.

Búsqueda de pacientes

Apellidos	Nombre	NIF/NIE
Arenas Prieto	Raquel	47485960E
Arenas	Raquel	47485960E
Sánchez de la Rosa	Manuel	23456789e
Arenas Gil	Jorge	34567896D
Romero Arganda	Roberto	98765432F
Pérez Rodríguez	Juan	99987656H
Rocha Morante	Sandra	23452345R
González Romero	Patricia	90876098E
Torres Díaz	Jara	11122233S
Martín Salas	Alba	67895432U
Zamorano Verdugo	Carmen	99977766Z
Rodríguez Morón	Samuel	56743287s

Figura 3.42. Pantalla “BusquedaPacientes”

Esta pantalla es la primera que aparece al pulsar “GESTIÓN DE PACIENTES”. Se utiliza para listar y filtrar los pacientes y desde ella podríamos ir a generar un nuevo paciente o consultar uno de la lista mediante su enlace.

Datos personales de paciente

Usuario: Rubén soto Benito
 Empleo: Médico Medicina Familiar

MI AGENDA **GESTIÓN DE PACIENTES** **CERRAR SESIÓN**

Nombre: Apellidos: NIF/NIE:
 Fecha Nacimiento: Sexo: Nacionalidad:

Tipo vía: Nombre vía: Núm.: Planta: Letra:
 Población: Provincia:
 Comunidad Autónoma: País:
 Teléfono1: Teléfono2: Teléfono3:
 Email:

GUARDAR **DATOS CLÍNICOS** **CITAS** **BORRAR**

Figura 3.43. Pantalla “DatosPersonalesPac”

Esta pantalla muestra los datos personales de un paciente. Desde ella se puede dar de alta uno nuevo, modificarlo, borrarlo y acceder a la gestión de sus datos clínicos y citas.

Datos clínicos

Usuario: Rubén soto Benito
 Empleo: Médico Medicina Familiar

MI AGENDA **GESTIÓN DE PACIENTES** **CERRAR SESIÓN**

Paciente: Patricia González Romero **Edad:** 4 **Sexo:** Mujer

EPISODIOS **NUEVO** **GUARDAR** **BORRAR**

Fecha	Episodio
01/10/2015	Dolor cervical

Fecha:
 Episodio:
 Descripción:

ANTECEDENTES **NUEVO**

- [Alergia fruta](#)
- [Operación quiste](#)

VALORES **NUEVO**

Descripción	Valor	Fecha
Altura(m)	1.580	15/09/2015
Peso(Kg)	70.000	01/10/2015

Figura 3.44. Pantalla “DatosClínicos” (episodio)

Esta pantalla muestra la historia clínica del paciente, mostrándonos todos sus datos clínicos. Vemos que está dividida en tres secciones, en la primera se mostrará un listado de todos los episodios que se tienen registrados del paciente, la parte central, será la que

en realidad nos dejará crear, modificar o borrar un episodio, y la parte derecha nos muestra los antecedentes y los valores que tenemos registrados de este paciente.

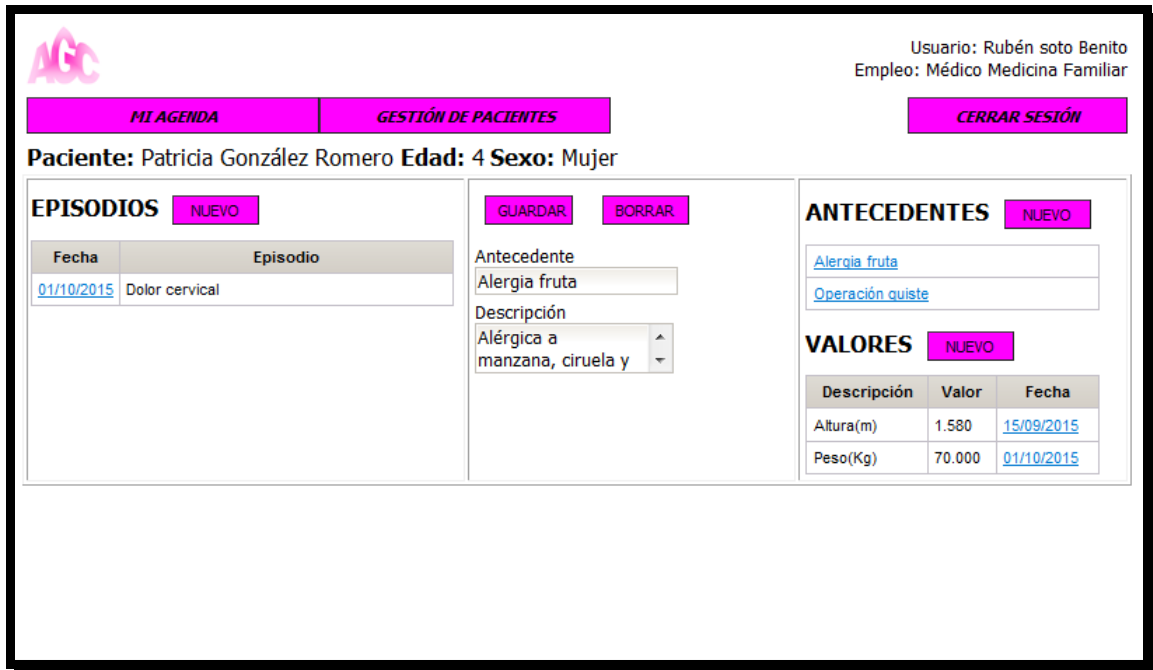


Figura 3.45. Pantalla “DatosClinicos” (antecedente)

Esta es la misma pantalla que la de episodio pero una vez se haya seleccionado un antecedente o pulsado el botón “NUEVO” en la sección de los antecedentes. Nos permitirá crear un nuevo antecedente, modificarlo o borrarlo.

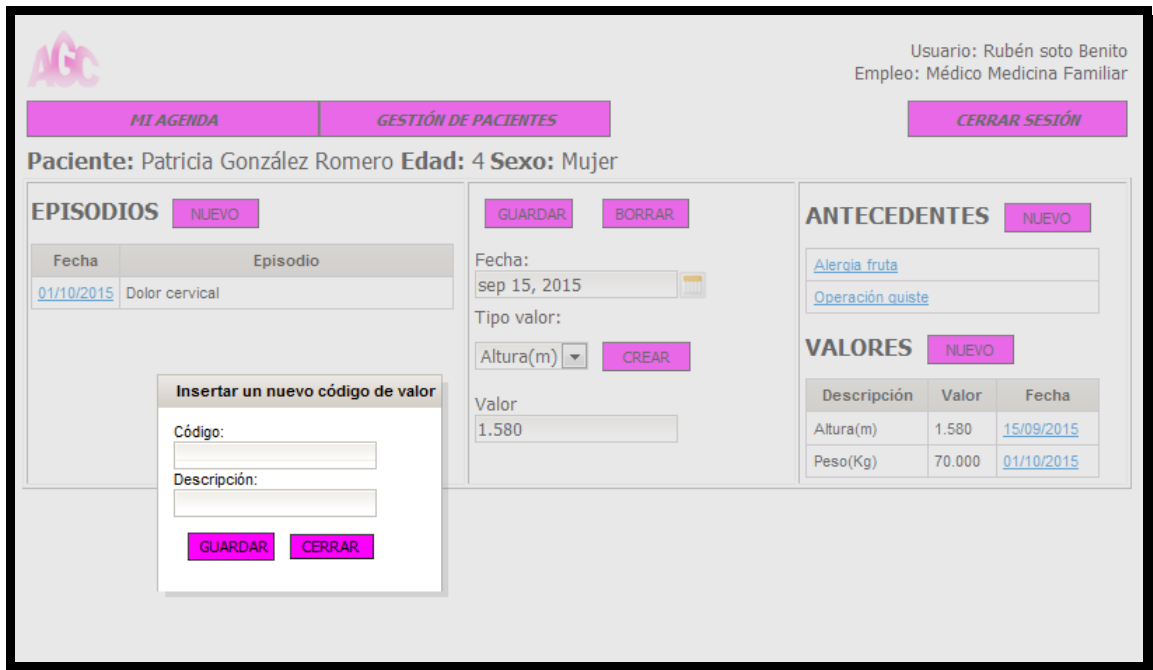


Figura 3.46. Pantalla “DatosClinicos” (valor)

Esta es la misma pantalla que la de episodio pero una vez se haya seleccionado un valor o pulsado el botón “NUEVO” en la sección de los valores. Nos permitirá crear un nuevo valor, modificarlo o borrarlo. Además en la captura se observa una pantalla emergente que sale al pulsar el botón “CREAR” y que nos permite registrar nuevos tipos de valores.

Citas

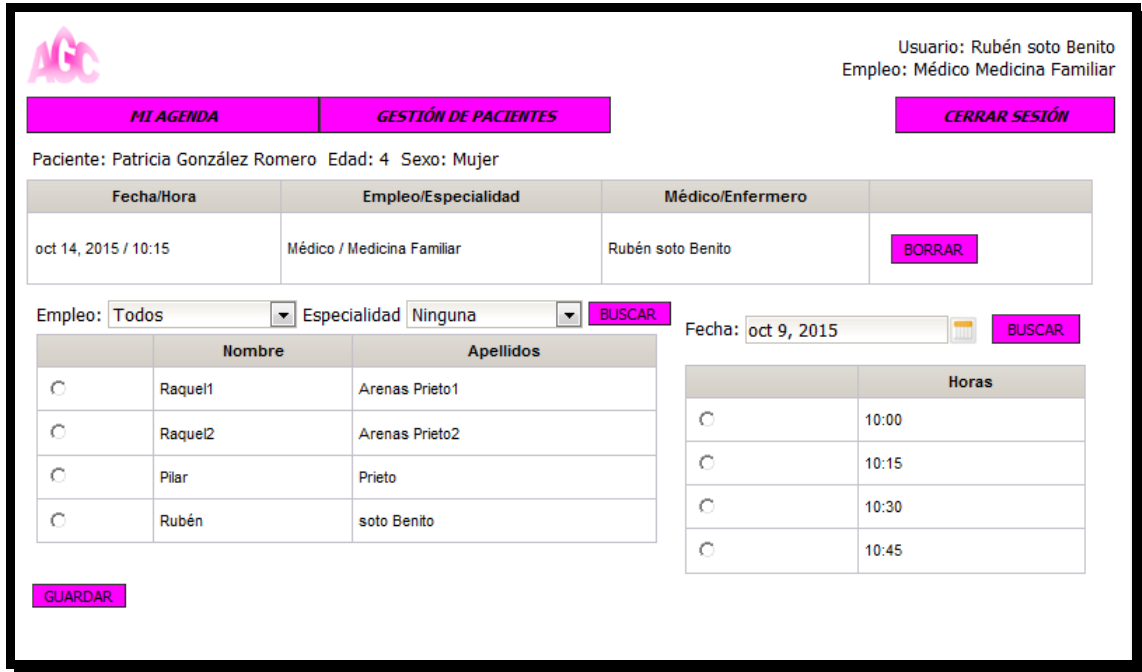


Figura 3.47. Pantalla “Citas”

En esta pantalla veremos arriba las citas que tenga pendientes el paciente. Y se puede buscar a un empleado para seleccionarlo junto con una hora y grabar una nueva cita.

3.3. Pruebas

En los anteriores apartados de este capítulo hemos explicado detalladamente qué es lo que tiene que permitir realizar la aplicación. Y en este apartado nos dedicaremos a probar todas estas funcionalidades.

Al igual que en los anteriores apartados, también realizaremos las pruebas desde los dos perfiles de usuario que se han diseñado. A excepción de la pantalla “Login” que como hemos dicho en el apartado anterior, es común a todos los usuarios de la aplicación.

Y a continuación mostramos la tabla de pruebas de la pantalla Login.


Pantalla “Login”		
FUNCIONALIDAD	RESULTADO	DESCRIPCIÓN
Validación		Se realiza al pulsar el botón “ACCEDER” y comprueba que el usuario y contraseña sean correctos con los que hay grabados en base de datos.

Tabla 3.4. Pruebas pantalla “Login”

3.3.1. Pruebas perfil no sanitario

En este apartado vamos a realizar las pruebas por pantalla para el perfil de usuarios no sanitarios para cada una de las pantallas a las que tiene acceso y las funcionalidades que puede realizar en ellas.




Pantalla “PaginaPrincipal” – Perfil no sanitario		
FUNCIONALIDAD	RESULTADO	DESCRIPCIÓN
Gestión empleados		Al pulsar el botón “GESTIÓN EMPLEADOS” nos debe redirigir a la pantalla de búsqueda de empleados.
Gestión pacientes		Al pulsar el botón “GESTIÓN PACIENTES” nos debe redirigir a la pantalla de búsqueda de pacientes.
Cerrar sesión		Al pulsar el botón “CERRAR SESIÓN” nos debe sacar del contexto de la página y llevarnos a la página de login.

Tabla 3.5. Pruebas pantalla “PaginaPrincipal”

Estas funcionalidades se pueden realizar desde cualquier pantalla de las siguientes.



Pantalla “BusquedaPacientes” – Perfil no sanitario		
FUNCIONALIDAD	RESULTADO	DESCRIPCIÓN
Buscar paciente		Pulsando el botón “BUSCAR” se realiza la búsqueda de los pacientes y se listan por pantalla. Si antes de pulsar el botón está algún campo relleno, la búsqueda filtra por ellos.
Gestión paciente		Tanto desde el botón “NUEVO” como pinchando en el link del nombre del paciente nos redirigirá a la pantalla de datos personales del paciente (en el primer caso vacía y en el segundo relleno con los suyos)

Tabla 3.6. Pruebas pantalla “BusquedaPacientes”





Pantalla “DatosPersonalesPac” – Perfil no sanitario		
FUNCIONALIDAD	RESULTADO	DESCRIPCIÓN
Crear paciente		Una vez rellenos los campos con los datos del paciente, se pulsa el botón “GUARDAR”, y todos los datos se quedan grabados en las tablas correspondientes en la base de datos y se muestra de nuevo la misma pantalla.
Modificar paciente		Viendo los datos de un paciente ya existente, modificamos alguno de los campos de sus datos y pulsamos el botón “GUARDAR”, se actualizan los registros en la base de datos por los actuales. Y a continuación seguimos viendo la misma pantalla.
Borrar paciente		Viendo los datos de un paciente ya existente, pulsamos el botón “BORRAR”, se borrarán todos los registros de la base de datos que están relacionados con este paciente.
Consultar citas		Pulsando el botón “CITAS” nos lleva a la pantalla de citas y nos muestra todas las citas que tiene pendientes.

Tabla 3.7. Pruebas pantalla “DatosPersonalesPac”




Pantalla “Citas” – Perfil no sanitario		
FUNCIONALIDAD	RESULTADO	DESCRIPCIÓN
Crear cita		Habiendo seleccionado un empleado de la lista y una fecha con una hora, al dar al botón “GUARDAR” se creará el registro correspondiente en la base de datos.
Borrar cita		En el listado de citas, si pulsamos al botón “BORRAR” se eliminará la cita de esa línea de la base de datos.
Consultar cita		Se ven las citas de un paciente al entrar en esta pantalla.

Tabla 3.8. Pruebas pantalla de citas



Pantalla “BusquedaEmpleados” – Perfil no sanitario		
FUNCIONALIDAD	RESULTADO	DESCRIPCIÓN
Buscar empleado		Pulsando el botón “BUSCAR” se realiza la búsqueda de los empleados y se listan por pantalla. Si antes de pulsar el botón está algún campo relleno, la búsqueda filtra por ellos.
Gestión empleado		Tanto desde el botón “NUEVO” como pinchando en el link del nombre del empleado nos redirige a la pantalla de datos personales del empleado (en el primer caso vacía y en el segundo relleno con los suyos)

Tabla 3.9. Pruebas pantalla “BusquedaEmpleados”





Pantalla “DatosPersonalesEmp” – Perfil no sanitario		
FUNCIONALIDAD	RESULTADO	DESCRIPCIÓN
Crear empleado		Una vez rellenos los campos con los datos del empleado, se pulsa el botón “GUARDAR”, y todos los datos se quedan grabados en las tablas correspondientes en la base de datos y se muestra de nuevo la misma pantalla.
Modificar empleado		Viendo los datos de un empleado ya existente, modificamos alguno de los campos de sus datos y pulsamos el botón “GUARDAR”, se actualizan los registros en la base de datos por los actuales. Y a continuación seguimos viendo la misma pantalla.
Borrar empleado		Viendo los datos de un empleado ya existente, pulsamos el botón “BORRAR”, se borrarán todos los registros de la base de datos que están relacionados con este empleado.
Consultar agenda		<i>(si el empleado es de perfil sanitario)</i> Pulsando el botón “AGENDA” se hace una búsqueda de las citas que tiene pendientes el empleado para el día actual y se redirige a la pantalla de agenda.

Tabla 3.10. Pruebas pantalla “DatosPersonalesEmp”


Pantalla “Agenda” – Perfil no sanitario		
FUNCIONALIDAD	RESULTADO	DESCRIPCIÓN
Consultar agenda		De momento aparecen en esta pantalla listadas las citas para ese mismo día, pero se puede elegir una fecha para ver las de ese día.

Tabla 3.11. Pruebas pantalla “Agenda”

3.3.2. Pruebas perfil sanitario

En este apartado vamos a realizar las pruebas por pantalla para el perfil de usuarios sanitarios para cada una de las pantallas a las que tiene acceso y las funcionalidades que puede realizar en ellas.




Pantalla “PaginaPrincipal” – Perfil sanitario		
FUNCIONALIDAD	RESULTADO	DESCRIPCIÓN
Gestión pacientes		Al pulsar el botón “GESTIÓN PACIENTES” nos debe redirigir a la pantalla de búsqueda de pacientes.
Consultar agenda		Al pulsar el botón “AGENDA” deberá llevarnos a la página de la agenda del empleado que está usando la aplicación.
Cerrar sesión		Al pulsar el botón “CERRAR SESIÓN” nos debe sacar del contexto de la página y llevarnos a la página de login.

Tabla 3.12. Pruebas pantalla “PaginaPrincipal”

Estas funcionalidades se pueden realizar desde cualquier pantalla de las siguientes.


Pantalla “Agenda” – Perfil sanitario		
FUNCIONALIDAD	RESULTADO	DESCRIPCIÓN
Consultar agenda		De momento aparecen en esta pantalla listadas las citas para ese mismo día, pero se puede elegir una fecha para ver las de ese día.

Tabla 3.13. Pruebas pantalla “Agenda”



Pantalla “BusquedaPacientes” – Perfil sanitario		
FUNCIONALIDAD	RESULTADO	DESCRIPCIÓN
Buscar paciente		Pulsando el botón “BUSCAR” se realiza la búsqueda de los pacientes y se listan por pantalla. Si antes de pulsar el botón está algún campo relleno, la búsqueda filtra por ellos.
Gestión paciente		Tanto desde el botón “NUEVO” como pinchando en el link del nombre del paciente nos redirigirá a la pantalla de datos personales del paciente (en el primer caso vacía y en el segundo relleno con los suyos)

Tabla 3.14. Pruebas pantalla “BusquedaPacientes”






Pantalla “DatosPersonalesPac” – Perfil sanitario		
FUNCIONALIDAD	RESULTADO	DESCRIPCIÓN
Crear paciente		Una vez rellenos los campos con los datos del paciente, se pulsa el botón “GUARDAR”, y todos los datos se quedan grabados en las tablas correspondientes en la base de datos y se muestra de nuevo la misma pantalla.
Modificar paciente		Viendo los datos de un paciente ya existente, modificamos alguno de los campos de sus datos y pulsamos el botón “GUARDAR”, se actualizan los registros en la base de datos por los actuales. Y a continuación seguimos viendo la misma pantalla.
Borrar paciente		Viendo los datos de un paciente ya existente, pulsamos el botón “BORRAR”, se borrarán todos los registros de la base de datos que están relacionados con este paciente.
Consultar citas		Pulsando el botón “CITAS” nos lleva a la pantalla de citas y nos muestra todas las citas que tiene pendientes.
Gestión historia clínica		Pulsando el botón “DATOS CLÍNICOS” nos lleva a la página de los datos clínicos mostrándolos.

Tabla 3.15. Pruebas pantalla “DatosPersonalesPac”







Pantalla “DatosClinicos” (episodio) – Perfil sanitario		
FUNCIONALIDAD	RESULTADO	DESCRIPCIÓN
Crear episodio		Pulsar en la sección de los episodios el botón “NUEVO”, y nos muestra en el centro de la pantalla los campos a rellenar de un episodio. Una vez rellenos se pulsa el botón “GUARDAR”. Y los datos se graban en la base de datos. Después seguimos viendo la misma pantalla.
Modificar episodio		Viendo los datos de un episodio ya existente, modificamos alguno de los campos de sus datos y pulsamos el botón “GUARDAR”, se actualizan los registros en la base de datos por los actuales. Y a continuación seguimos viendo la misma pantalla.
Borrar episodio		Viendo los datos de un episodio ya existente, pulsamos el botón “BORRAR”, se borra el episodio de la base de datos.
Consultar episodio		Pulsando el link del episodio que queremos ver que aparece en el listado de episodios.
Consultar antecedente		Pulsando el link del antecedente que queremos ver que aparece en el listado de antecedentes.
Consultar valor		Pulsando el link del valor que queremos ver que aparece en el listado de valores.

Tabla 3.16. Pruebas pantalla “DatosClinicos” (episodio)







Pantalla “DatosClinicos” (antecedente) – Perfil sanitario		
FUNCIONALIDAD	RESULTADO	DESCRIPCIÓN
Crear antecedente		Pulsar en la sección de los antecedentes el botón “NUEVO”, y nos muestra en el centro de la pantalla los campos a rellenar de un antecedente. Una vez rellenos se pulsa el botón “GUARDAR”. Y los datos se graban en la base de datos. Después seguimos viendo la misma pantalla
Modificar antecedente		Viendo los datos de un antecedente ya existente, modificamos alguno de los campos de sus datos y pulsamos el botón “GUARDAR”, se actualizan los registros en la base de datos por los actuales. Y a continuación seguimos viendo la misma pantalla.
Borrar antecedente		Viendo los datos de un antecedente ya existente, pulsamos el botón “BORRAR”, se borra el antecedente de la base de datos.
Consultar antecedente		Pulsando el link del antecedente que queremos ver que aparece en el listado de antecedentes.
Consultar episodio		Pulsando el link del episodio que queremos ver que aparece en el listado de episodios.
Consultar valor		Pulsando el link del valor que queremos ver que aparece en el listado de valores.

Tabla 3.17. Pruebas pantalla “DatosClinicos” (antecedente)








Pantalla “DatosClinicos” (valor) – Perfil sanitario		
FUNCIONALIDAD	RESULTADO	DESCRIPCIÓN
Crear valor		Pulsar en la sección de los valores el botón “NUEVO”, y nos muestra en el centro de la pantalla los campos a rellenar de un valor. Una vez rellenos se pulsa el botón “GUARDAR”. Y los datos se graban en la base de datos. Después seguimos viendo la misma pantalla
Modificar valor		Viendo los datos de un valor ya existente, modificamos alguno de los campos de sus datos y pulsamos el botón “GUARDAR”, se actualizan los registros en la base de datos por los actuales. Y a continuación seguimos viendo la misma pantalla.
Borrar valor		Viendo los datos de un valor ya existente, pulsamos el botón “BORRAR”, se borra el valor de la base de datos.
Crear valor descripción		Al pulsar el botón “CREAR” nos muestra una ventana emergente que nos permite grabar una descripción de valor nueva.
Consultar valor		Pulsando el link del valor que queremos ver que aparece en el listado de valores.
Consultar episodio		Pulsando el link del episodio que queremos ver que aparece en el listado de episodios.
Consultar antecedente		Pulsando el link del antecedente que queremos ver que aparece en el listado de antecedentes.

Tabla 3.18. Pruebas pantalla “DatosClinicos” (valor)




Pantalla “Citas” – Perfil sanitario		
FUNCIONALIDAD	RESULTADO	DESCRIPCIÓN
Crear cita		Habiendo seleccionado un empleado de la lista y una fecha con una hora, al dar al botón “GUARDAR” se creará el registro correspondiente en la base de datos.
Borrar cita		En el listado de citas, si pulsamos al botón “BORRAR” se eliminará la cita de esa línea de la base de datos.
Consultar cita		Se ven las citas de un paciente al entrar en esta pantalla.

Tabla 3.19. Pruebas pantalla “Citas”

Capítulo 4

Arquitectura tecnológica de la aplicación

En este cuarto capítulo exponemos cómo es la arquitectura a nivel de sistema que implementa el proyecto, describiendo los medios y herramientas tecnológicos utilizados para su desarrollo.

Exponemos los pasos seguidos para la instalación de las herramientas que han sido necesarias para la programación de la aplicación.

Y por último describimos cuál es la estructura de ficheros usada durante el desarrollo de ésta.

4.1. Arquitectura del sistema

La arquitectura de la aplicación a nivel de sistema, queda de la forma indicada en la siguiente figura.

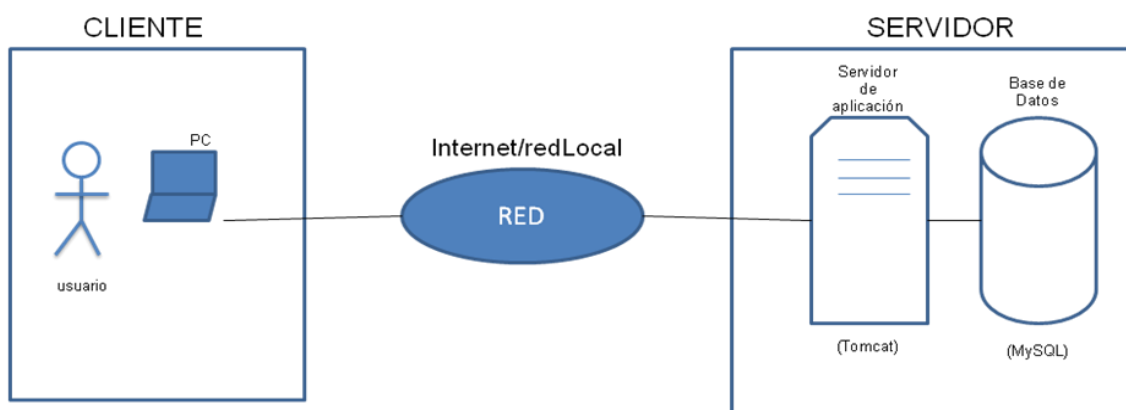


Figura 4.1. Arquitectura a nivel del sistema

En ella podemos diferenciar claramente las partes del cliente y servidor.

En la parte de cliente vemos cómo el usuario tendrá acceso a la aplicación desde su ordenador por mediación de un navegador web apuntando por URL al equipo que soporta el servidor de la aplicación.

La parte del servidor se compone del servidor web Tomcat versión 8 y la base de datos MySQL versión 5.6. En el servidor web se encontrará corriendo la aplicación y desde éste accederemos a la base de datos.

Nuestra aplicación, como la gran mayoría de aplicaciones web, es un claro caso del patrón Modelo-Vista-Controlador (MVC), ya que el cliente desde el navegador, va a hacer llamadas al servidor de aplicaciones, el cual gestionará y controlará el acceso al modelo de datos.

Desde un punto de vista general, el patrón MVC es un estilo de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos: modelo, vista y controlador. [12]

El modelo será la representación de los datos con los que el sistema va a trabajar. Las peticiones de acceso o manipulación de los datos llegan a este a través del controlador.

La vista presenta el modelo en un formato adecuado para interactuar. Será la interfaz de usuario.

Y por último el controlador responde a eventos y realiza peticiones al modelo si se hace alguna solicitud de datos. Por lo que podemos afirmar que el controlador es un intermediario entre la vista y el modelo.

[13]

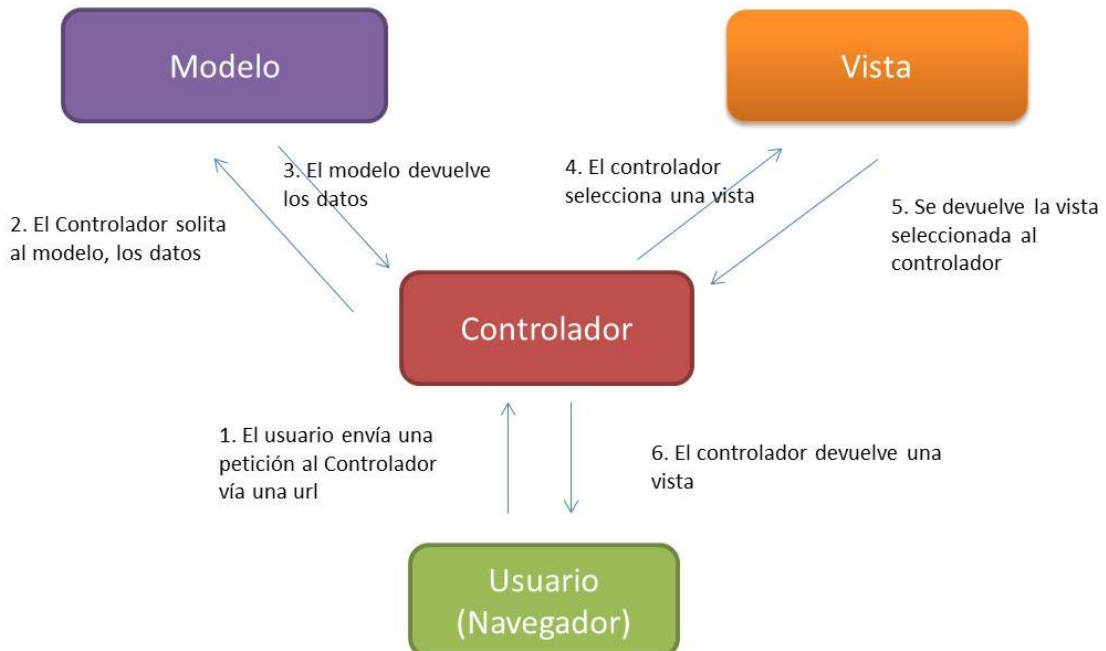


Figura 4.2. Esquema MVC

4.2. Medios utilizados y justificación

A continuación describimos los medios que hemos utilizado para el desarrollo de nuestra aplicación web.

Lenguaje de programación: *Java JDK1.7.0_51*

“Es un lenguaje de programación de propósito general, concurrente, orientado a objetos que fue diseñado específicamente para tener tan pocas dependencias de implementación como fuera posible.” [14]

Framework: *JSF 2.2.8*

Entre los dos framework más utilizados en la actualidad (**JSF** y **Spring**) nos quedamos con JSF, aunque pueden utilizarse a la vez, porque la aplicación desarrollada va dirigida a un entorno único donde el acceso se producirá siempre desde equipos del mismo tipo. Y no es una aplicación a gran escala, por lo que no hace falta liberar el servidor de carga de controles, ya que JSF centraliza todos los controles en el servidor y tiene mucho más tráfico de consulta entre la aplicación y el servidor. [15] [16] [17]

JSF [18] es un marco de trabajo para crear aplicaciones Java J2EE basadas en el patrón MVC.

Utiliza páginas JSP para generar las vistas, añadiendo una biblioteca de etiquetas propia para crear los elementos de los formularios HTML.

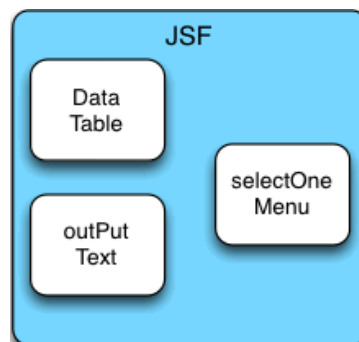


Figura 4.3. Vista JSF

Asocia a cada vista con formularios un conjunto de objetos java manejados por el controlador (managed beans) que facilitan la recogida, manipulación y visualización de los valores mostrados en los diferentes elementos de los formularios.

Utiliza un sencillo fichero de configuración para el controlador en formato xml, que se llama `faces-config.xml`.

Es extensible, pudiendo crearse nuevos elementos de la interfaz o modificar los ya existentes.

Spring MVC [19] también es un framework para el desarrollo de aplicaciones y contenedor de inversión de control, de código abierto para la plataforma Java.

El Contenedor se puede configurar mediante la carga de archivos XML o la detección de anotaciones Java específicas sobre la configuración de las clases.

Su vista se forma de ficheros HTML básicos y por lo tanto se pueden acoplar otras tecnologías sobre esta, como es el caso de poder utilizar JSF sobre Spring.

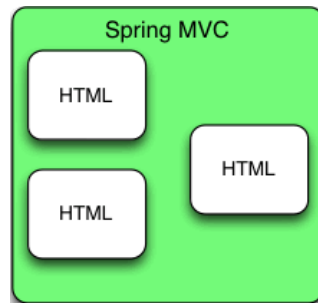


Figura 4.4. Vista Spring MVC

Componente de JSF: *RichFaces 4.5.3*

Es una biblioteca de código abierto basada en Java que construye sobre el framework de Java Server Faces. [20]

Lo utilizamos para el manejo de tablas, ya que ofrece varias opciones sobre ellas que nos parecen interesantes para ahora y en posibles trabajos futuros. [21]

Base de datos: *MySQL 5.6*

“MySQL es un sistema de gestión de bases de datos relacional, multihilo y multiusuario. MySQL AB –desde enero de 2008 una subsidiaria de Sun Microsystems y ésta a su vez de Oracle Corporation desde abril de 2009-.” [22]

Desde el código conectamos a la base de datos mediante la conexión JDBC.

```
4
5
6
7
8 public final class BaseDatos {
9     private static Connection conexion;
10    private static Statement statement;
11
12    public BaseDatos() {
13        try {
14            Class.forName("com.mysql.jdbc.Driver");
15            String servidor = "jdbc:mysql://localhost/prueba2";
16            String usuarioDB = "root";
17            String passwordDB = "PFCRAP";
18            conexion = DriverManager.getConnection(servidor, usuarioDB,
19                passwordDB);
20            conexion.setAutoCommit(true);
21            statement = conexion.createStatement();
22            System.out.println("conexion cerrada: " + conexion.isClosed());
23            System.out.println("conexion: " + conexion.getSchema());
24        } catch (ClassNotFoundException ex) {
25            // JOptionPane.showMessageDialog(null, ex,
```

Figura 4.5. Conexión JDBC

Servidor web: *Apache Tomcat 8.0.3*

“También llamado Jakarta Tomcat o simplemente Tomcat. Funciona como un contenedor de servlets desarrollado bajo el proyecto Jakarta en la Apache Software Foundation. Tomcat implementa las especificaciones de los servlets y de Java Server Pages (JSP) de Oracle Corporation (aunque creado por Sun Microsystems).” [23]

Herramienta de desarrollo web: *Eclipse Java EE IDE for Web Developers (Version: Kepler Service Release 2)*

“Es un programa informático compuesto por un conjunto de herramientas de programación de código abierto multiplataforma para desarrollar lo que el proyecto llama “Aplicaciones de Cliente Enriquecido”, opuesto a las aplicaciones “Clienteliviano” basadas en navegadores. Esta plataforma, típicamente ha sido usada para desarrollar entornos de desarrollo integrados (del inglés IDE), como el IDE de Java llamado Java Development Toolkit (JDT) y el compilador (ECJ) que se entrega como parte de Eclipse (y que son usados también para desarrollar el mismo Eclipse).” [24]

Herramienta de diseño de base de datos: *DBDesigner 4*

DBDesigner es una herramienta de diseño y creación de base de datos MySQL que proporciona de forma gratuita la plataforma MySQL, junto al instalador de la propia base de datos.

Lenguajes de programación para la interfaz: XHTML 1.0 y CSS

Tal y como se ha comentado en el apartado 3.2, la interfaz de la aplicación se ha generado a partir de ficheros XHTML y se ha pulido su estética con CSS.

XHTML es el lenguaje utilizado para la creación de páginas web a partir de los originales HTML y XML.

CSS es un lenguaje usado para definir el estilo de las páginas web. Pudiendo así tener el estilo centralizado en un fichero `.css` en el que se describen las propiedades para cada tipo de campo, y así en distintas páginas web llamar al mismo estilo para los campos sin tener que definirlo en cada una de ellas.

4.3. Instalación de las distintas herramientas para el desarrollo de la aplicación

Para el desarrollo de la aplicación diseñada, hemos tenido que preparar un equipo de desarrollo, el cual lo hemos configurado de la siguiente forma.

1º. Para la utilización de la tecnología Java, en la compilación y ejecución de nuestro código, hemos tenido que descargar e instalar el ejecutable “jdk1.7.0_51” de la página oficial de Oracle.

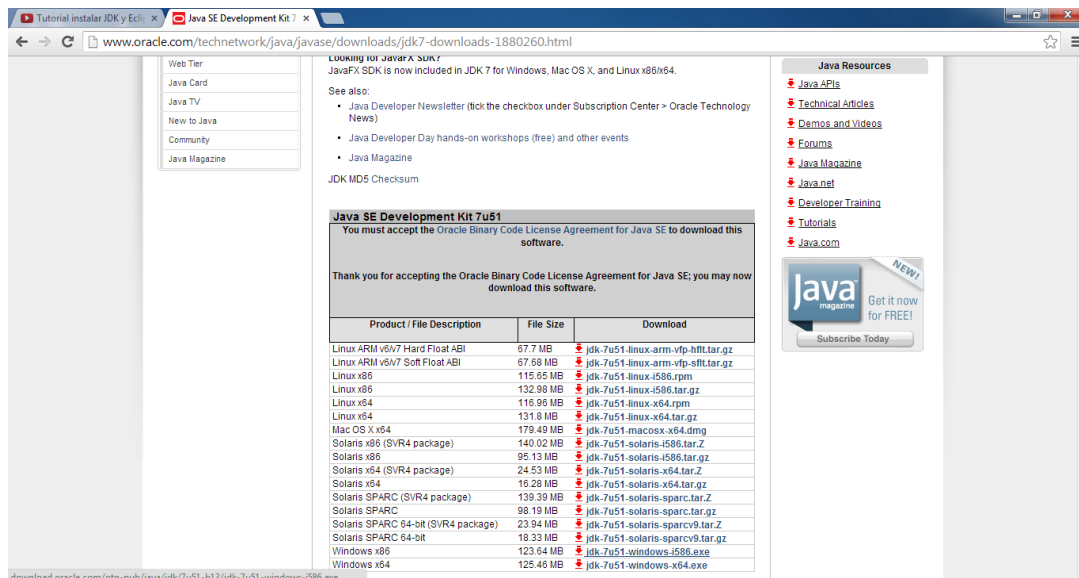


Figura 4.6. Página de descarga de los JDK

Lo que vemos en esta figura es la página de descarga de ejecutables JDK. Allí hay que elegir el ejecutable según el sistema operativo que se utilice en el equipo en el que se

quiera instalar, en nuestro caso es un Windows de 32 bits, por lo que nos descargamos el ejecutable “jdk-7u51-windows-i586.exe”. [25]

2º. Descargamos e instalamos las herramientas para el diseño, creación y mantenimiento de la base de datos MySQL.

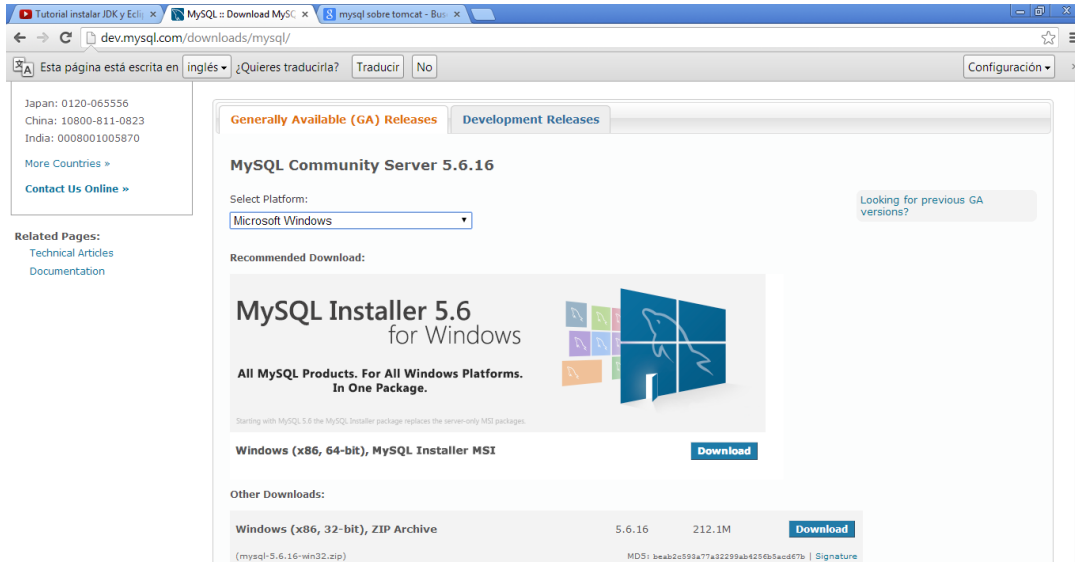


Figura 4.7. Página de descarga de productos MySQL

En esta figura observamos la página web desde donde descargamos el instalador de la base de datos y la herramienta para la creación y configuración de ésta que en nuestro caso se llama DBDesigner. [26]

3º. Para el diseño y configuración de la base de datos utilizamos la herramienta de diseño de base de datos DBDesigner que vemos en la siguiente figura.

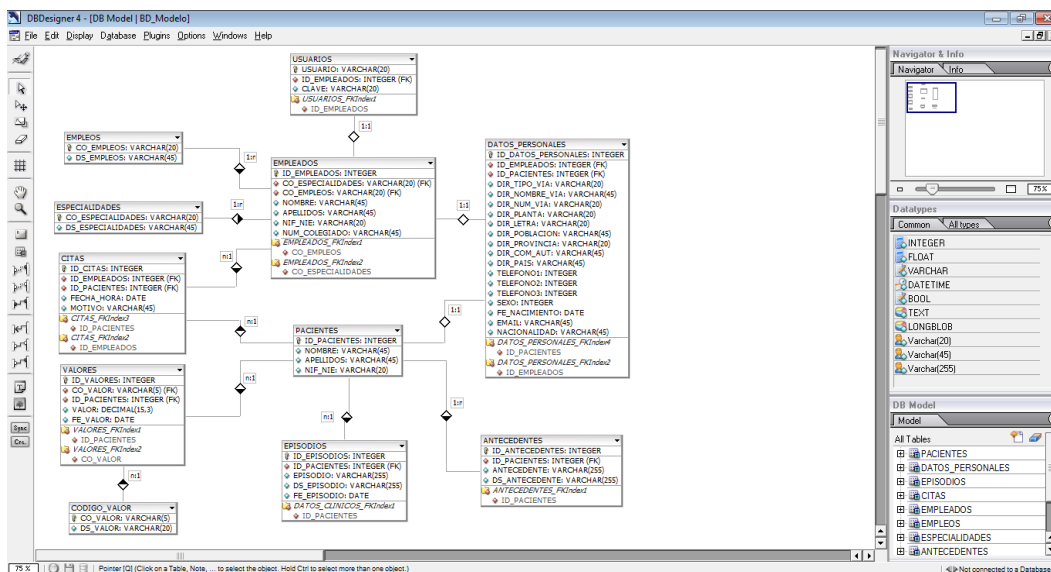


Figura 4.8. Herramienta de diseño DBDesigner

En esta figura, podemos observar cómo se está diseñando la base de datos, definiendo las tablas, las relaciones entre ellas y los tipos de datos que contienen. En la *Figura 4.8* se muestra el esquema entidad relación de nuestra base de datos al completo y en la *Figura 4.9*, el detalle de cada campo de una de las tablas contenidas en esta base de datos.

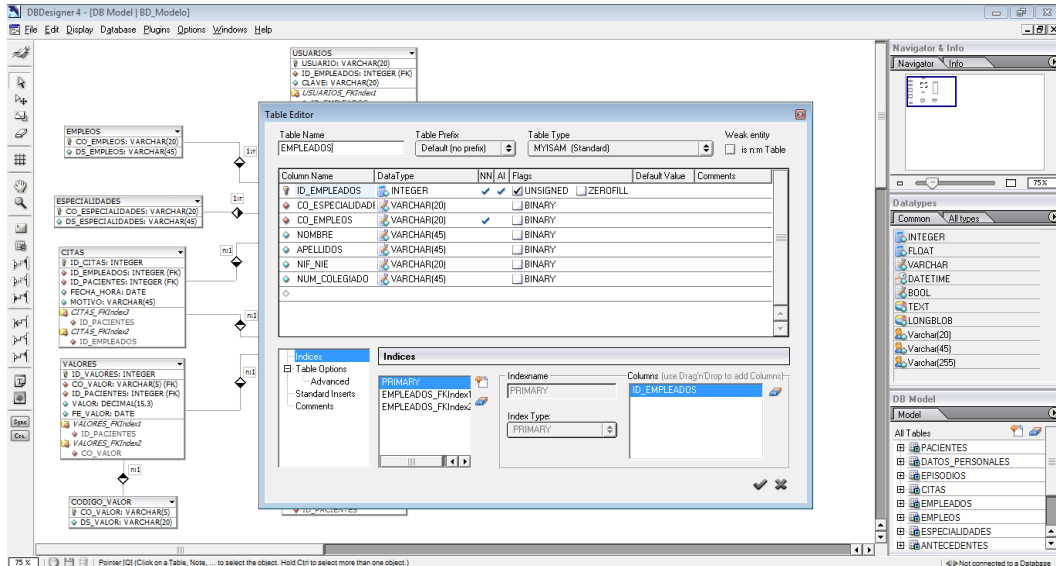


Figura 4.9. Configuración tabla en DBDesigner

En esta ventana se consulta y modifica la configuración de los campos de la tabla. Por ejemplo, en este caso vemos cómo el campo ID_EMPLEADOS es un campo Primary Key, es de tipo INTEGER y nunca podrá ser NULL.

4º. Para desplegar la aplicación y poder realizar pruebas utilizamos el servidor de aplicaciones Apache Tomcat.

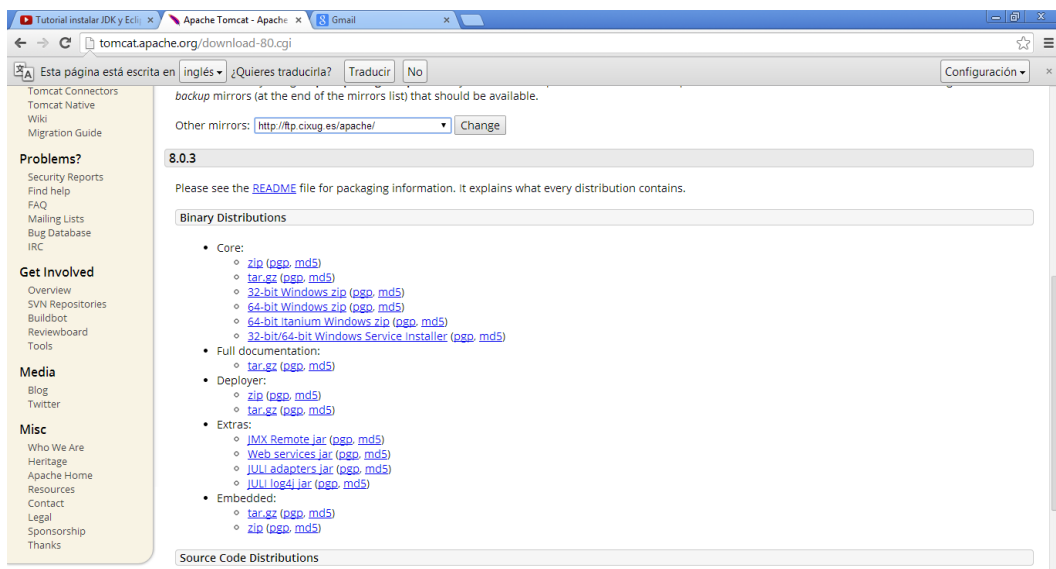


Figura 4.10. Página de descarga de servidor de aplicaciones

De esta página lo que se va a descargar es una carpeta que descomprimiremos y dejaremos en un directorio del equipo, para que al desplegar la aplicación, elijamos desplegarla en esta carpeta. [27]

En la carpeta “lib” meteremos las distintas librerías que necesitamos para ejecutar nuestra aplicación, como se muestra en la siguiente figura. Por ejemplo el conector a base de datos, las librerías de JSF (javax.faces-2.2.8-02 y jsf-impl-2.2.8-02)...

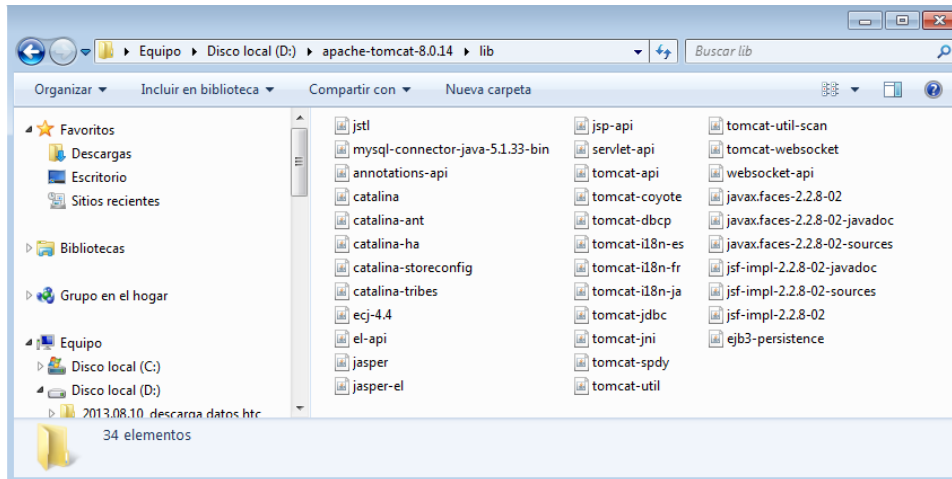


Figura 4.11. Librerías en el servidor de aplicaciones

5º. Para programar utilizamos el software libre Eclipse.

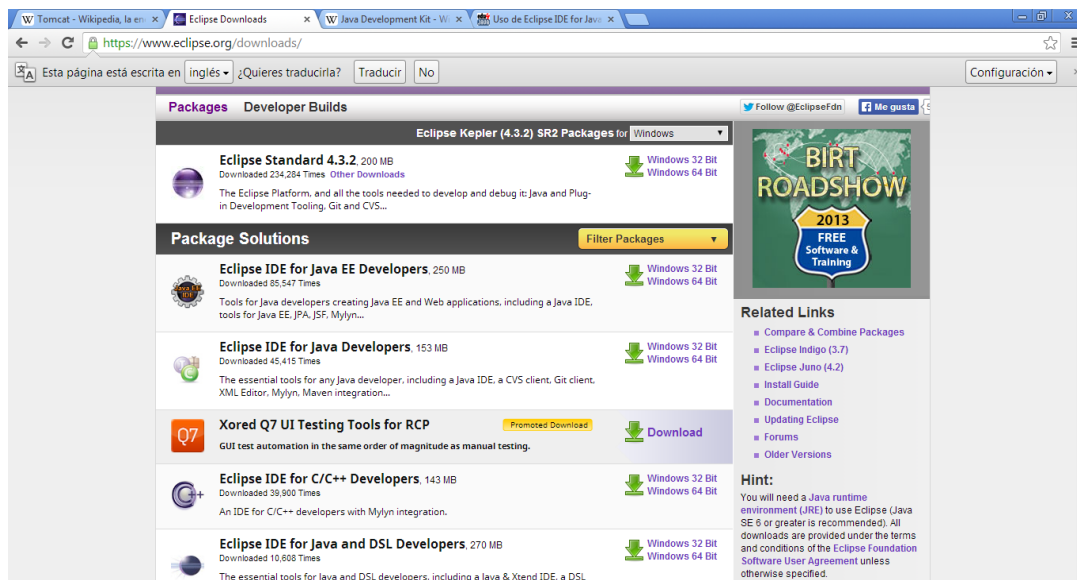


Figura 4.12. Página de descarga del software y complementos de Eclipse

En la figura 4.12 vemos la web de descargas de Eclipse, en la que se pueden descargar la herramienta de desarrollo en todas sus versiones y algunos complementos, nosotros descargaremos la primera opción que aparece “Eclipse IDE for Java EE Developers”. Esta herramienta de desarrollo no necesita de instalación, tan solo hay que descomprimir la carpeta que se nos descarga. [28]

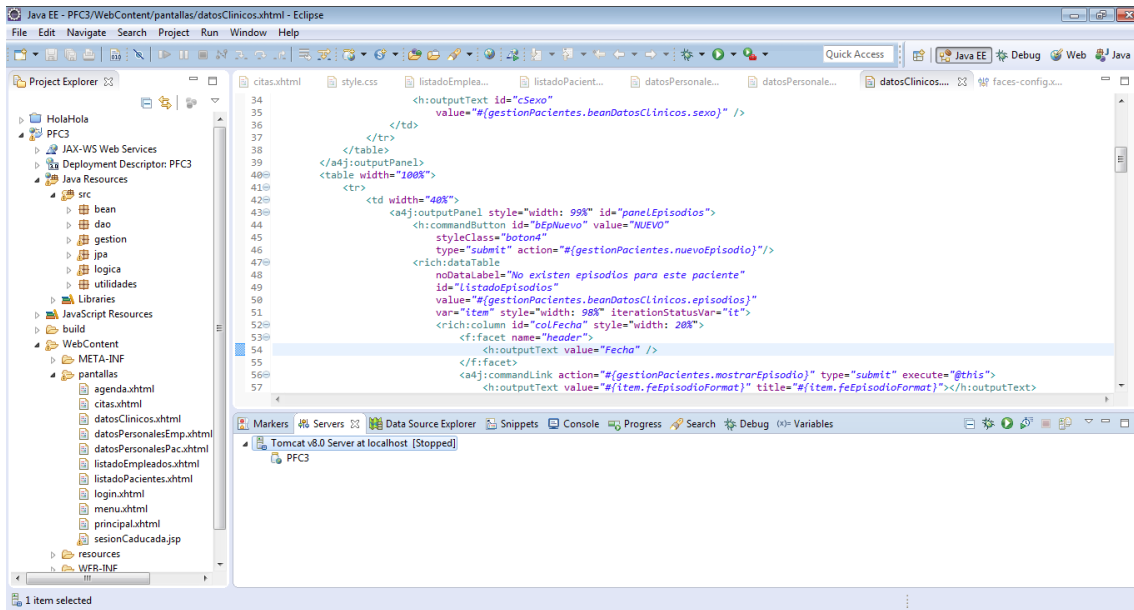


Figura 4.13. Entorno de desarrollo de la aplicación Eclipse

En esta figura lo que muestra es una captura de pantalla de la aplicación durante su utilización en el desarrollo de la aplicación.

A continuación mostramos los ficheros de configuración necesarios, para que nuestra aplicación utilice correctamente el framework JSF.

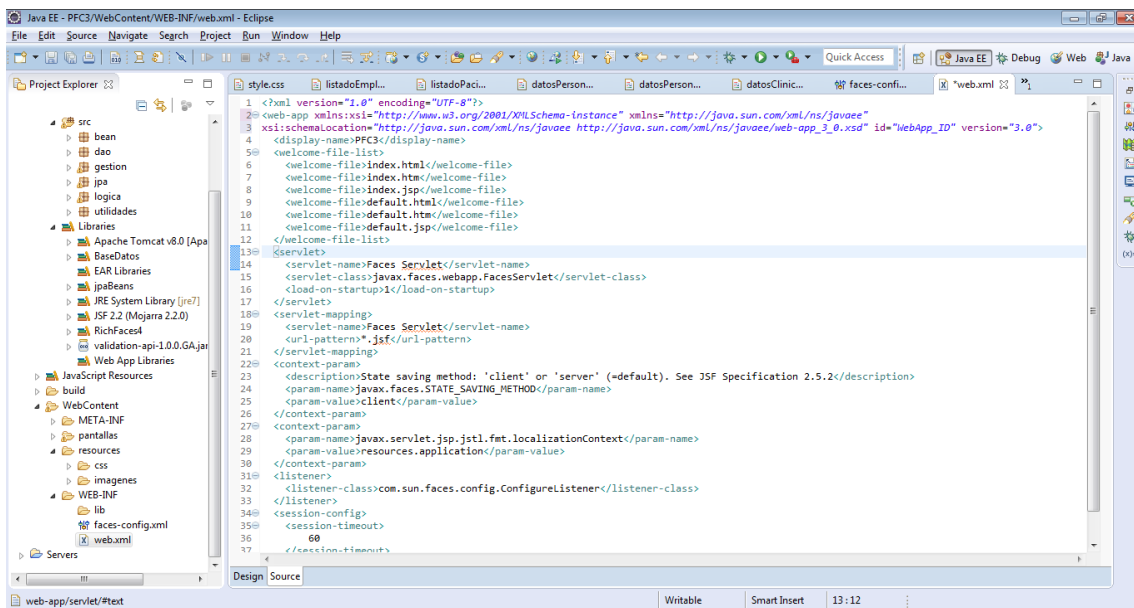


Figura 4.14. Fichero web.xml

Se trata del fichero web.xml en el que configuramos la versión de JSF a utilizar, la extensión por defecto de las páginas, duración máxima de la sesión, el localizador del servlet...

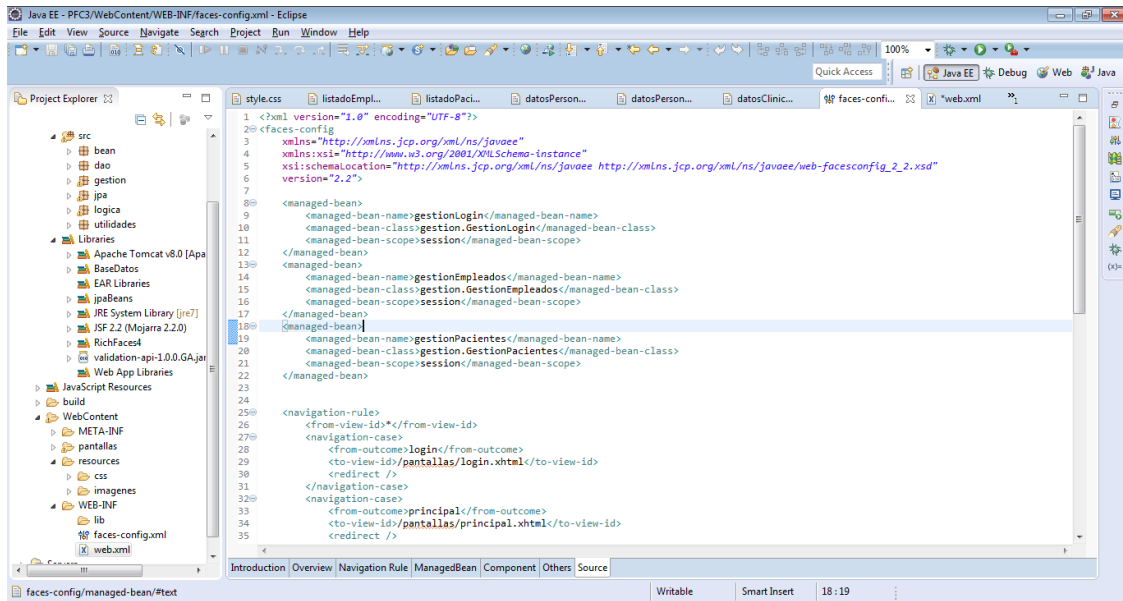


Figura 4.15. Fichero faces-config.xml

En este fichero `faces-config.xml` lo que se configura son los controles de cada gestión y en qué ámbito se tratarán. En este fichero también se encontrarán las reglas de navegación, correspondiendo los nombres cortos con sus directorios concretos al XHTML.

4.4. Estructura de paquetes utilizada para el desarrollo de la aplicación

A nivel de programación hemos utilizado la distribución de paquetes mostrada en la figura:

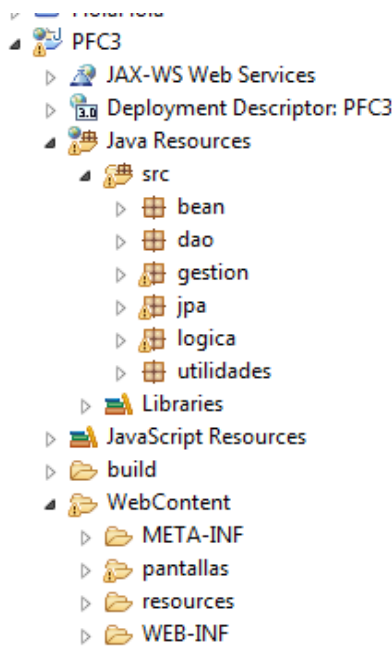


Figura 4.16. Estructura del proyecto

En la carpeta “Java Resources” se encuentran los ficheros fuente de la aplicación en lenguaje Java, y en la carpeta “WebContent” el contenido web.

Dentro de la carpeta WebContent están los siguientes contenidos:

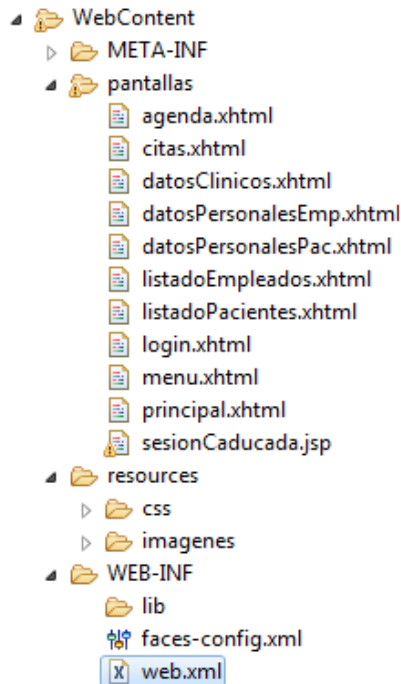


Figura 4.17. Carpeta de ficheros WebContent

En la carpeta “pantallas” tenemos todos los ficheros .xhtml que hemos creado para la vista de nuestra aplicación.

En la carpeta “resources” encontramos las posibles imágenes y ficheros .css para la estética de la aplicación.

En la carpeta WEB-INF encontramos el fichero web.xml en el que se encuentra la referencia al framework que se va a utilizar para resolver las acciones en los .xhtml. Y por otro lado el fichero faces-config.xml en el que se van insertando los nombres cortos de las gestiones y vistas que se utilizan en la aplicación.

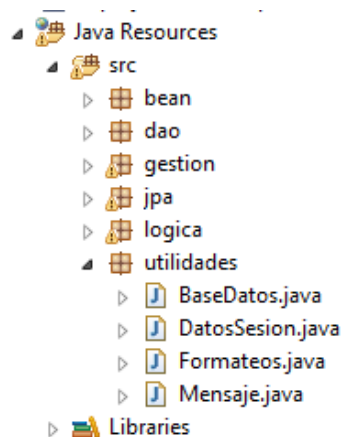


Figura 4.18. Carpeta de ficheros “Java Resources”

En la carpeta “Java Resources” tenemos el paquete “Libraries” que contiene las librerías que utilizamos para programar en nuestra aplicación, como lo son RichFaces, las librerías de JSF, el conector para la base de datos...

Y en la carpeta “src” tenemos separado en paquetes:

- “bean”: contiene los beans de pantalla que son las clases que contienen todos los datos de cada pantalla.
- “dao”: contiene los ficheros que son capaces de realizar acciones sobre la base de datos.
- “gestion”: aquí se encuentran los ficheros que gestionan la comunicación entre la vista y la base de datos, contienen los objetos de pantalla y los unen con la lógica.
- “jpa”: contiene los objetos del modelo de la base de datos, en los que se apoyarán los DAO para trabajar con los datos.
- “lógica”: contiene la lógica de la aplicación, desde donde se realizan las llamadas a los daos para que estos ejecuten las operaciones básicas hacia la base de datos.

“utilidades”: aquí encontramos distintos ficheros que sirven de apoyo para el resto de paquetes, como son la gestión de la base de datos, el formateo de los datos para guardarlos o mostrarlos en base de datos y/o pantalla, se guardan ciertos datos de sesión para poder utilizarlos desde cualquier parte de la aplicación.

Capítulo 5

Planificación y presupuesto

En este capítulo mostramos como ha sido la planificación en tiempos y fases del proyecto y el presupuesto que supone el diseño, implementación y puesta en marcha de la aplicación en el centro sanitario.

5.1. Planificación

La planificación de un desarrollo de software suele estar marcada por distintas partes, como se muestra en la siguiente figura.

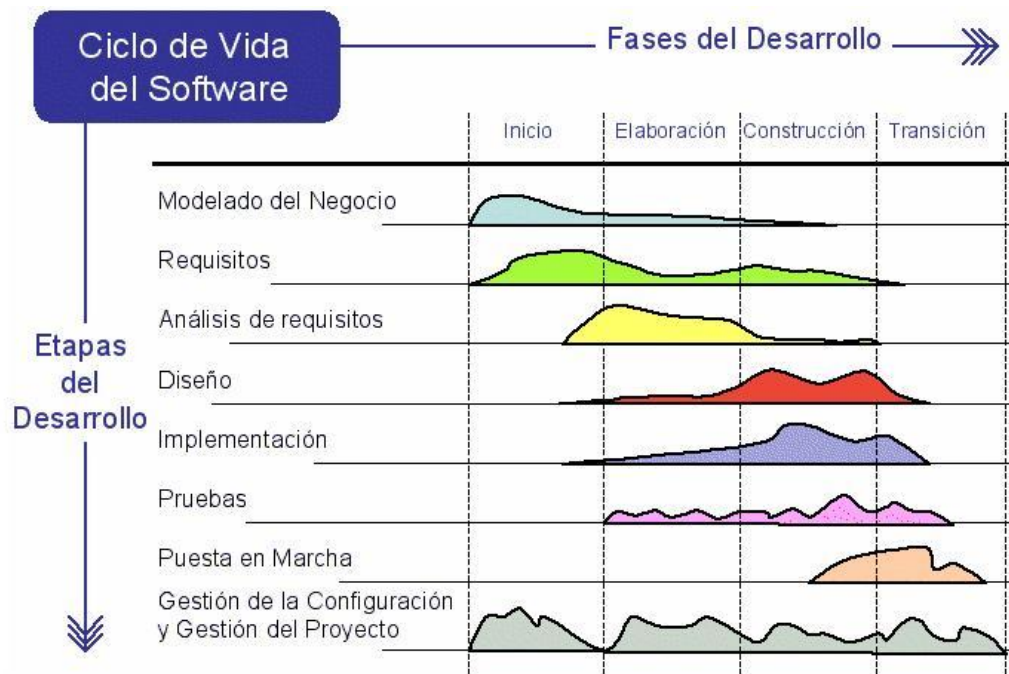


Figura 5.1. Ciclo de vida de software

En nuestro caso no ha existido la parte “puesta en marcha”, ya que es sólo una simulación y no se lleva a término en ningún escenario real.

El desarrollo de nuestra aplicación se ha realizado en el transcurso de 8 meses más o menos, y dedicando aproximadamente 2 horas al día de lunes a viernes.

Casi todas las fases que a continuación se señalan se han ido solapando en ocasiones unas con otras, pero mostramos una aproximación de duración de cada una.

1^{er} mes (*Modelado de negocio y requisitos*):

Búsqueda de información sobre tecnologías (~15h)

Búsqueda de otras aplicaciones similares (~25h)

2^o mes (*Modelado de negocio y requisitos*):

Instalación del equipo de desarrollo (~10h)

Iniciación a las tecnologías (~30h)

3^{er} mes (*Análisis de requisitos*):

Análisis de requisitos (~40h)

4^o mes (*Diseño*):

Diseño de la base de datos (~10h)

Diseño de la interfaz (~30h)

5^o, 6^o, 7^o y 8^o mes (*Implementación*):

Desarrollo de código (~160h)

9^o, 10^o, 11^o y 12^o mes:

Redacción de la memoria (~160h)

5.2. Presupuesto

En este apartado pasamos a desglosar el presupuesto para esta aplicación, contando con la fase de desarrollo y su posible futura implantación y puesta en marcha.

Para decidir qué honorarios para un Ingeniero Técnico de Telecomunicación en Telemática, hemos consultado en el COITT, pero nos hemos encontrado con que los honorarios son totalmente libres [29]. Por lo que tomaremos un valor que nos parece justo a nosotros, que sería 40€/hora.

Además, a este presupuesto añadiremos las horas invertidas por el tutor (Ingeniero de Telecomunicación) para reuniones, revisar memoria... Por lo que estimamos 10 horas invertidas por él, con unos honorarios de 45€/hora.

FASE DE DESARROLLO

<i>PERSONAL</i>	Cantidad	Nº horas	€/hora	Total
Ing.T.T.Telemática	1	480	40 €	19.200 €
Ing. Telec.	1	10	45 €	450 €
<i>MATERIAL</i>			€/equipo	
Ordenador	1		500 €	500 €
TOTAL				20.150 €

Figura 5.2. Presupuesto de la fase de desarrollo

IMPLEMENTACIÓN / PUESTA EN MARCHA / MANTENIMIENTO

80h instalación y puesta en marcha

60h (1h/día durante 3 meses) mantenimiento para solventar posibles errores

<i>PERSONAL</i>	Cantidad	Nº horas	€/hora	Total
Ing.T.T.Telemática	1	140	40 €	5.600 €
<i>MATERIAL</i>			€/equipo	
Servidores	2		700 €	1.400 €
Ordenadores	24		300 €	7.200 €
Monitores	25		150 €	3.750 €
TOTAL				17.950 €

Figura 5.3. Presupuesto de implementación/puesta en marcha/mantenimiento

Hemos separado el presupuesto en la fase de “desarrollo” y la de “implantación, puesta en marcha y mantenimiento”.

Obteniendo unos costes totales de 20150€ para la fase de desarrollo y 17950€ para la implantación, puesta en marcha y mantenimiento de la aplicación.

Capítulo 6

Conclusiones y posibles trabajos futuros

Finalizamos esta memoria con este capítulo, en el que exponemos las conclusiones a las que llegamos tras el trabajo realizado para este proyecto y proponemos posibles ampliaciones que se podrían añadir en un futuro a la aplicación.

6.1. Conclusiones

En el apartado 1.3. Objetivos se listaron una serie de funcionalidades requeridas para nuestra aplicación. Y una vez terminada la fase de Implementación, que como mostramos en el apartado 5.1. Planificación es hasta esa fase a la que hemos llegado con nuestro proyecto.

Podemos confirmar que nuestro proyecto cumple con todas estas funcionalidades listadas en los objetivos, ya que han pasado las pruebas realizadas en el apartado 3.3. por cada pantalla en la que se pueden realizar y por cada perfil que tiene acceso a ellas.

Además, como vemos en el apartado 3.1.1. Perfiles de usuario, se ha desarrollado la aplicación valorando los tipos de perfiles de usuario, ya que como hemos descrito en el apartado 2.2. Protección de datos clínicos existen leyes que restringen el acceso a los datos clínicos a según las funciones que tenga cada empleado, por lo que hemos considerado el dividir los empleos de un centro sanitario en estos dos perfiles: “sanitario” y “no sanitario”.

6.2. Posibles trabajos futuros

Como hemos podido observar, esta aplicación cumple con los objetivos que se han marcado en un primer momento. Pero no podemos dejar de pensar, en que es una aplicación básica, a la que se le podrían ir añadiendo muchas otras funcionalidades.

A continuación describimos algunas de las funcionalidades que se le podrían ir incluyendo en un futuro.

Impresión de historial clínico

Se daría la posibilidad de poder imprimir el historial clínico del paciente, tanto en un documento digital, como en papel. Esto sería bueno para poder enviar el historial clínico de un paciente a otro centro sanitario, o si el paciente quisiera tener una copia.

Generación e impresión de recetas

Generar las recetas directamente desde la aplicación e imprimirlas para entregárselas al paciente.

Traducción de la aplicación

Podría darse la posibilidad de traducir a otros idiomas, como por ejemplo las lenguas cooficiales de España (gallego, catalán, euskera...).

Mejorar la seguridad del tratamiento de datos

Se podrían cifrar determinados campos que pudieran ser más comprometidos, para que a la hora de que la aplicación fuera a mostrarlos, nos pidiera algún identificativo distinto del de la entrada a la interfaz.

Consultar y guardar resultados de datos clínicos externos

Se daría acceso a consultar los historiales clínicos registrados en otros centros sanitarios, así como resultados de pruebas que se realizaran en otros sistemas, y con opción de descargarlos y asociarlos al historial del paciente en nuestro sistema.

Por ejemplo comunicación desde el centro a un hospital, un laboratorio u otros centros sanitarios de características similares.

Para esto deberíamos estudiar los protocolos y estándares de comunicación médica.

Consultar diccionario

Mostrar el enlace a un posible diccionario de medicina, por ejemplo Vademécum.

Gestión de nóminas y vacaciones

En la pantalla de los datos personales de los empleados se añadirían dos botones. Uno sería para acceder a una pantalla con un listado de documentos, que serían las nóminas y con posibilidad de ir añadiendo cada mes la nómina correspondiente. Y el otro botón nos llevaría a la pantalla de gestión de las vacaciones en las que se podrían consultar los días de vacaciones solicitados por el empleado.

Mantenimiento de tablas

Crear un mantenimiento de tablas por aplicación, para poder crear, modificar o borrar registros en las tablas maestras.

Añadir nuevos perfiles de usuario

Hasta el momento se ha hecho la división de perfiles sólo en sanitario y no sanitario, pero ya dentro del sanitario encontramos distintos tipos de competencias, que de lo que la aplicación gestiona hasta el momento no encontramos ningún conflicto, pero pensando en algunas de las posibles mejoras que hemos propuesto en este mismo apartado sí podemos encontrarlas.

Por ejemplo nos encontramos con que un auxiliar de enfermería no tiene la competencia de recetar, sin embargo un médico sí pueda recetar. Otro caso sería si se incluyera en la plantilla del centro sanitario un empleado con el puesto, por ejemplo, de informático, éste no tendría por qué tener acceso a los datos de los pacientes (ni clínicos, ni personales), pero sí podría gestionar su nómina y vacaciones mediante la aplicación.

Y entre estos perfiles se podría añadir otro más para los pacientes del centro. Creando un usuario para cada uno de ellos, y así permitirles consultar sus propias citas y su propio historial médico.

Tablón de anuncios, avisos y noticias

Utilizando el espacio que queda libre en la pantalla de menú principal, se podría mostrar un tablón de anuncios, avisos y noticias en él. Se trataría de anuncios y noticias relacionadas con el centro sanitario, por ejemplo: “Reunión de personal del centro. El miércoles 14 de octubre a las 14:30h en el gimnasio”.

Glosario

Antecedente. *Dato o circunstancia personal o familiar en la historia clínica del paciente, previa a su situación actual. Historia.*

Resumen global de la salud general de la persona hasta la fecha, incluyendo las lesiones antiguas, alergias, intervenciones quirúrgicas, inmunizaciones, hospitalizaciones, así como la historia obstétrica y psiquiátrica. [30]

Centro sanitario. *Instalación médica dedicada a la asistencia y al cuidado de la salud de los enfermos. [31]*

Clave foránea. *En el contexto de bases de datos relacionales, una clave foránea o clave ajena (o Foreign Key FK) es una limitación referencial entre dos tablas. La clave foránea identifica una columna o grupo de columnas en una tabla (tabla hija o referendo) que se refiere a una columna o grupo de columnas en otra tabla (tabla maestra o referenciada). Las columnas en la tabla referendo deben ser la clave primaria u otra clave candidata en la tabla referenciada. [32]*

Clave primaria. *En el diseño de bases de datos relacionales, se llama clave primaria a un campo o a una combinación de campos que identifica de forma única a cada fila de una tabla. Una clave primaria comprende de esta manera una columna o conjunto de columnas. No puede haber dos filas en una tabla que tengan la misma clave primaria. [33]*

Episodio. *El episodio de atención es un problema de salud o enfermedad desde la primera consulta a un profesional sanitario hasta la última consulta con el mismo. Un episodio de atención es un conjunto de una o más consultas y su relación en el curso del tiempo (transición). Mientras que “episodio de enfermedad” (o simplemente “episodio”) se refiere a la atención prestada a un problema concreto en un determinado paciente. [34]*

Framework. *(marco de trabajo) define, en términos generales, un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular que sirve como referencia, para enfrentar y resolver nuevos problemas de índole similar.*

En el desarrollo de software, un framework o infraestructura digital, es una estructura conceptual y tecnológica de soporte definido, normalmente con artefactos o módulos de software concretos, que puede servir de base para la organización y desarrollo de software. Típicamente, puede incluir soporte de programas, bibliotecas, y un lenguaje interpretado, entre otras herramientas, para así ayudar a desarrollar y unir los diferentes componentes de un proyecto. [35]

Historia clínica. *Es un documento médico-legal que surge del contacto entre el profesional de la salud (médico, fisioterapeuta, odontólogo, psicólogo, asistente social, enfermero, kinesiólogo, podólogo) y el paciente donde se recoge la información necesaria para la correcta atención de los pacientes. La historia clínica es un documento válido desde el punto de vista clínico y legal, que recoge información de tipo asistencia, preventivo y social.* [36]

J2EE. *Es una plataforma de programación –parte de la Plataforma Java- para desarrollar y ejecutar software de aplicaciones en el lenguaje de programación Java.* [37]

MD5. *En criptografía, MD5 (abreviatura de Message-Digest Algorithm 5, Algoritmo de Resumen del Mensaje 5) es un algoritmo de reducción criptográfico de 128 bits ampliamente usado.* [38]

Referencias

Las referencias que aparecen en este apartado se encuentran ordenadas por el orden en que se citan en el documento.

- [1] Ministerio de Sanidad, Servicios Sociales e Igualdad
<http://www.msssi.gob.es/profesionales/hcdsns/home.htm>
[accedido el 23/09/2015]
- [2] Redacción Médica. Consejo interterritorial Sistema Nacional de Salud
http://www.redaccionmedica.com/contenido/images/NOTICIA_3.pdf
[accedido el 23/09/2015]
- [3] Soluciones OMI360 para el Sector Salud. STACKS
<http://www.stacks.es/software-atencion-primaria>
[accedido el 23/09/2015]
- [4] Gobierno del Principado de Asturias. Consejería de Sanidad
<http://gcsalud.asturias.org/demoomi.htm>
[accedido el 23/09/2015]
- [5] SALUS. Software para hospitales y clínicas. QSOFT
<http://www.softwaresalus.com/>
[accedido el 23/09/2015]
- [6] Ofimedic. Informática médica
<http://www.ofimedic.com/>
[accedido el 23/09/2015]
- [7] Boletín Oficial del Estado. Ministerio de la presidencia
<http://www.boe.es/buscar/act.php?id=BOE-A-2002-22188>
[accedido el 23/09/2015]
- [8] Instituto Nacional de Estadística
<http://www.ine.es/daco/daco42/codmun/codmunmapa.htm>
[accedido el 23/09/2015]
- [9] Instituto Nacional de Estadística
http://www.ine.es/daco/daco42/codmun/cod_provincia.htm
[accedido el 23/09/2015]
- [10] Instituto Nacional de Estadística
http://www.ine.es/daco/daco42/codmun/cod_ccaa.htm
[accedido el 23/09/2015]
- [11] Instituto Nacional de Estadística
http://www.ine.es/daco/daco42/clasificaciones/paises_estandar.xls
[accedido el 23/09/2015]

- [12] Wikipedia (Modelo–vista–controlador).
<https://es.wikipedia.org/wiki/Modelo%E2%80%93vista%E2%80%93controlador>
[accedido el 23/09/2015]
- [13] Mi blog técnico. Un blog dedicado a cuestiones TIC
<https://miblogtecnico.wordpress.com/2012/07/16/desarrollo-de-aplicaciones-con-el-framework-mvc-de-net-i/>
[accedido el 23/09/2015]
- [14] Wikipedia (Java (lenguaje de programación))
[https://es.wikipedia.org/wiki/Java_\(lenguaje_de_programaci%C3%B3n\)](https://es.wikipedia.org/wiki/Java_(lenguaje_de_programaci%C3%B3n))
[accedido el 23/09/2015]
- [15] Arquitecturajava. “JSF vs Spring MVC (Video)” por Cecilio Álvarez Caules
<http://www.arquitecturajava.com/jsf-vs-spring-mvc-video/>
[accedido el 23/09/2015]
- [16] Arquitecturajava. “¿Tiene futuro JSF?” por Cecilio Álvarez Caules
<http://www.arquitecturajava.com/tiene-futuro-jsf/>
[accedido el 23/09/2015]
- [17] Arquitecturajava. “JSF, Spring MVC y Java EE 8” por Cecilio Álvarez Caules
<http://www.arquitecturajava.com/jsf-spring-mvc-y-java-ee-8/>
[accedido el 23/09/2015]
- [18] AdictosAlTrabajo. “Introducción a JSF Java Server Faces” por Cristóbal González Almirón
<http://www.adictosaltrabajo.com/tutoriales/introduccion-jsf-java/>
[accedido el 23/09/2015]
- [19] Wikipedia (Spring Framework)
https://es.wikipedia.org/wiki/Spring_Framework
[accedido el 23/09/2015]
- [20] Wikipedia (RichFaces)
<https://es.wikipedia.org/wiki/RichFaces>
[accedido el 23/09/2015]
- [21] RichFaces
<http://showcase.richfaces.org/>
[accedido el 23/09/2015]
- [22] Wikipedia (MySQL)
<https://es.wikipedia.org/wiki/MySQL>
[accedido el 23/09/2015]
- [23] Wikipedia (Tomcat)
<https://es.wikipedia.org/wiki/Tomcat#Historia>
[accedido el 23/09/2015]

- [24] Wikipedia (Eclipse (software))
[https://es.wikipedia.org/wiki/Eclipse_\(software\)](https://es.wikipedia.org/wiki/Eclipse_(software))
[accedido el 23/09/2015]
- [25] Oracle
<http://www.oracle.com/technetwork/java/javase/downloads/jdk7-downloads-1880260.html>
[accedido el 23/09/2015]
- [26] MySQL
<http://dev.mysql.com/downloads/mysql/>
[accedido el 23/09/2015]
- [27] Apache Tomcat
<http://Tomcat.apache.org/download-80.cgi>
[accedido el 23/09/2015]
- [28] eclipse
<https://www.eclipse.org/>
[accedido el 23/09/2015]
- [29] COITT/AEGITT. “Honorarios profesionales”
<http://www.coitt.es/res/libredocs/Honorarios.pdf>
[accedido el 23/09/2015]
- [30] IQB Instituto Químico Biológico
<http://www.iqb.es/diccio/a/an2.htm>
[accedido el 23/09/2015]
- [31] Wikipedia (Centro sanitario)
https://es.wikipedia.org/wiki/Centro_sanitario
[accedido el 23/09/2015]
- [32] Wikipedia (Clave foránea)
https://es.wikipedia.org/wiki/Clave_for%C3%A1nea
[accedido el 23/09/2015]
- [33] Wikipedia (Clave primaria)
https://es.wikipedia.org/wiki/Clave_primaria
[accedido el 23/09/2015]
- [34] Wikipedia (Episodio de atención)
https://es.wikipedia.org/wiki/Episodio_de_atenci%C3%B3n
[accedido el 23/09/2015]
- [35] Wikipedia (Framework)
<https://es.wikipedia.org/wiki/Framework>
[accedido el 23/09/2015]

- [36] Wikipedia (Historia Clínica)
https://es.wikipedia.org/wiki/Historia_cl%C3%ADnica
[accedido el 23/09/2015]
- [37] Wikipedia (Java EE)
https://es.wikipedia.org/wiki/Java_EE
[accedido el 23/09/2015]
- [38] Wikipedia (MD5)
<https://es.wikipedia.org/wiki/MD5>
[accedido el 23/09/2015]