



ugr

Universidad de Granada
Departamento de Ciencias de la Computación
e Inteligencia Artificial



Asignatura: Tecnologías Web

Curso 2015/16

Curso: 3º Grado

Grupo de prácticas: 3

Práctica: ____ Ejercicio: ____

Datos de los Alumnos: (ordenar alfabéticamente por apellidos).

Apellidos: Alcalá Martínez

Nombre: Antonio

D.N.I.: 75152552-Y

e-mail ugr: totidos@correo.ugr.es



Apellidos: Sánchez Jiménez

Nombre: David

D.N.I.: 75570649-D

e-mail ugr: dauidsanchezjimenez@correo.ugr.es



Logo

Especificación de requisitos de software

Proyecto: Gestor de revisiones de exámenes
1.0

Abril

Ficha del documento

Fecha	Revisión	Autor	Verificado dep. calidad.
14/06/2016	1.0	Antonio Alcalá Martínez. David Sánchez Jiménez.	

Documento validado por las partes en fecha: 14/06/2016

Por el cliente	Por la empresa suministradora
Fdo. D./ Dña [Nombre]	Fdo. D./Dña [Nombre]

Contenido

FICHA DEL DOCUMENTO	3
CONTENIDO	4
1 INTRODUCCIÓN	7
1.1 Propósito	7
1.2 Alcance	7
1.3 Personal involucrado	7
1.4 Definiciones, acrónimos y abreviaturas	7
1.5 Referencias	8
1.6 Resumen	8
2 DESCRIPCIÓN GENERAL	8
2.1 Perspectiva del producto	8
2.2 Funcionalidad del producto	8
2.3 Restricciones	9
2.4 Suposiciones y dependencias	9
2.5 Evolución previsible del sistema	9
3 REQUISITOS ESPECÍFICOS	9
3.1 Requisitos comunes de los interfaces	9
3.1.1 Interfaces de usuario	9
3.1.1.1 IU-01: Pantalla de inicio	9
3.1.1.2 IU-03: Pantalla del administrador	10
3.1.1.3 IU-04: Pantalla del profesor	10
3.1.1.4 IU-05: Pantalla de usuario no registrado	11
3.1.1.5 IU-06: Pantalla de espera	11
3.1.2 Interfaces de hardware	11
3.1.3 Interfaces de software	11
3.1.4 Interfaces de comunicación	11
3.2 Requisitos funcionales	12
3.2.1 Requisito funcional 1: Pantalla inicio	12
3.2.1.1 Requisito funcional 1.1: Acceso usuario registrado	12
3.2.1.2 Requisito funcional 1.2: Acceso usuario no registrado	12
3.2.2 Requisito funcional 2: Pantalla Login usuario registrado	13
3.2.2.1 Requisito funcional 2.1: Login usuario registrado	13
3.2.2.1.1 Requisito funcional 2.1.1: Comprobación tipo de usuario	14
3.2.3 Requisito funcional 3: Pantalla usuario no registrado	14
3.2.3.1 Requisito funcional 3.1: Unirse a cola	14
3.2.3.1.1 Requisito funcional 3.1.1: Generar código usuario	15

3.2.4	Requisito funcional 4: Acceder usando código	15
3.2.4.1	Requisito funcional 4.1: Comprobar código	15
3.2.4.2	Requisito funcional 4.2: Consultar información de la cola	16
3.2.4.3	Requisito funcional 4.3: Darse de baja de la cola	16
3.2.5	Requisito funcional 5: Pantalla del Administrador	17
3.2.5.1	Requisito funcional 5.1: Añadir usuario	17
3.2.5.2	Requisito funcional 5.2: Borrar usuario	17
3.2.5.3	Requisito funcional 5.3: Modificar usuario	18
3.2.5.3.1	Requisito funcional 5.3.1: Modificar nombre	18
3.2.5.3.2	Requisito funcional 5.3.2: Modificar email	19
3.2.5.4	Requisito funcional 5.4: Crear cola	20
3.2.5.4.1	Requisito funcional 5.4.1: Profesor asociado a la cola	20
3.2.5.5	Requisito funcional 5.5: Borrar cola	21
3.2.5.6	Requisito funcional 5.6: Modificar cualquier cola	21
3.2.5.6.1	Requisito funcional 5.6.1: Modificar profesor asociado	22
3.2.6	Requisito funcional 6: Pantalla del profesor	22
3.2.6.1	Requisito funcional 6.1: Modificar mi usuario	22
3.2.6.1.1	Requisito funcional 6.1.1: Modificar mi nombre	23
3.2.6.1.2	Requisito funcional 6.1.2: Modificar mi email	23
3.2.6.2	Requisito funcional 6.2: Crear cola propia	24
3.2.6.3	Modificar mis colas	24
3.2.6.4	Requisito funcional 6.5: Siguiente turno	25
3.2.6.5	Requisito funcional 6.5: Turno en espera	25
3.2.7	Requisito funcional 7: Pantalla de espera	26
3.2.7.1	Requisito funcional 7.1: Lista de últimos llamados	26
3.2.8	Requisito funcional 8: Requisitos generales	26
3.2.8.1	Requisito funcional 8.1: Comprobación de DNI	26
3.2.8.2	Requisito funcional 8.2: Comprobación de email	27
3.2.8.3	Requisito funcional 8.3: Confirmación de borrado	27
3.2.8.3.1	Requisito funcional 8.3.1: Eliminar información de la cola	28
3.2.8.4	Requisito funcional 8.4: Modificar lugar de revisión	28
3.2.8.5	Requisito funcional 8.5: Modificar horario	29
3.2.8.6	Requisito funcional 8.6: Crear aviso	29
3.2.8.7	Requisito funcional 8.7: Lista colas	30
3.2.8.8	Requisito funcional 8.8: Modificar password	30
3.2.8.9	Requisito funcional 8.9: Buscar profesor	31
3.3	Requisitos no funcionales	31
3.3.1	Requisitos de rendimiento	31
3.3.2	Seguridad	31
3.3.3	Fiabilidad	32
3.3.4	Disponibilidad	32
3.3.5	Mantenibilidad	32
3.3.6	Portabilidad	32
3.4	Otros requisitos	32
4	DESCRIPCIÓN DE LA SOLUCIÓN	33
4.1.1	Interfaces de usuario	33
4.1.1.1	IU-01: Pantalla de inicio	33
4.1.1.2	IU-03: Pantalla del administrador	34
4.1.1.3	IU-04: Pantalla del profesor	35
4.1.1.4	IU-05: Pantalla de usuario no registrado	36
4.1.1.5	IU-06: Modulo visualización	37
4.2	Requisitos funcionales	38
4.2.1	Requisito funcional 1: Pantalla inicio	38
4.2.1.1	Requisito funcional 1.1: Pantalla inicio	38
4.2.2	Requisito funcional 2: Pantalla Login usuario registrado	39

4.2.3	Requisito funcional 3: Pantalla usuario no registrado	39
4.2.3.1	Requisito funcional 3.1: Unirse a cola	40
4.2.3.1.1	Requisito funcional 3.1.1: Generar código usuario	41
4.2.3.2	Requisito funcional 3.2: Consultar posición en la cola	41
4.2.3.2.1	Requisito funcional 3.2.1: Darse de baja de la cola	42
4.2.4	Requisito funcional 4: Pantalla del Administrador	42
4.2.4.1	Requisito funcional 4.1: Gestión de usuarios	42
4.2.4.1.1	Requisito funcional 4.1.1: Dar de alta usuario	42
4.2.4.1.2	Requisito funcional 4.1.2: Borrar usuario	43
4.2.4.1.3	Requisito funcional 4.1.3: Modificar usuario	44
4.2.4.2	Requisito funcional 4.2: Gestión de colas	45
4.2.4.2.1	Requisito funcional 4.2.1: Crear cola	45
4.2.4.2.1.1	Requisito funcional 4.2.1.1: Generar código cola	45
4.2.4.2.1.2	Requisito funcional 4.2.1.2: Profesor asociado a la cola	46
4.2.4.2.2	Requisito funcional 4.2.2: Borrar cola	46
4.2.4.2.3	Requisito funcional 4.2.3: Modificar cola	47
4.2.4.2.3.1	Requisito funcional 4.2.3.1: Modificar profesor asociado	48
4.2.5	Requisito funcional 5: Pantalla del profesor	48
4.2.5.1	Requisito funcional 5.1: Gestión de usuarios	48
4.2.5.1.1	Requisito funcional 5.1.1: Modificar mi usuario	48
4.2.5.2	Requisito funcional 5.2: Gestión de colas	49
4.2.5.2.1	Requisito funcional 5.2.1: Crear cola propia	49
4.2.5.2.2	Requisito funcional 5.2.2: Modificar mis colas	50
4.2.5.2.3	Requisito funcional 5.2.3: Gestión turnos	50
4.2.6	Requisito funcional 6: Módulo de visualización	51
4.2.6.1	Requisito funcional 6.1: Lista de últimos llamados	51
4.2.6.2	Requisito funcional 6.2: Lista de avisos	52
4.2.7	Requisito funcional 7: Requisitos generales	53
4.2.7.1	Requisito funcional 7.1: Validar campos de registro	53
4.2.7.2	Requisito funcional 7.2: Eliminar información de la cola	53
4.2.7.3	Requisito funcional 7.3: Confirmación de borrado	54
4.2.7.4	Requisito funcional 7.4: Comprobar fecha y hora	54
4.2.7.5	Requisito funcional 7.5: Crear aviso	55
4.2.7.6	Requisito funcional 7.6: Conectarse base de datos	55
4.2.7.7	Requisito funcional 7.7: Cerrar sesión	56
5	APÉNDICES	58
5.1	Estructura de la base de datos	58
5.2	Diseño interfaces de usuario	59
5.2.1	Página inicio	59
5.2.2	Página administrador	59
5.2.3	Página profesor	60
5.2.4	Modulo visualización.	60

1 Introducción

Esta memoria presentará la documentación de un sistema web de gestión de citas para la revisión de calificaciones de un departamento de la Universidad.

Este sistema consistirá en la asignación de turnos por orden de solicitud, indicando el lugar de la cita, la hora y el código de cita. Esta información será mostrada por un monitor en una sala de espera.

1.1 Propósito

- El principal propósito del documento, es dar a conocer al cliente o personal técnica, los requisitos, las funcionalidades y restricciones del producto a desarrollar.
- Cliente y personal técnico.

1.2 Alcance

- El alcance de este proyecto, es modernizar el sistema de citas de un departamento, de forma que el usuario usando las nuevas tecnologías, puede inscribirse a una revisión de examen.

1.3 Personal involucrado

Nombre	Antonio Alcalá Martínez
Rol	Desarrollador
Categoría profesional	Estudiante
Responsabilidades	Co-dirección de desarrollo y gestión del proyecto
Información de contacto	totidos@correo.ugr.es
Aprobación	NA

Nombre	David Sánchez Jiménez
Rol	Desarrollador
Categoría profesional	Estudiante
Responsabilidades	Co-dirección de desarrollo y gestión del proyecto
Información de contacto	davidsanchezjimenez@correo.ugr.es
Aprobación	NA

Nombre	Waldo Fajardo Contreras
Rol	Cliente
Categoría profesional	Profesor universitario
Responsabilidades	NA
Información de contacto	aragorn@decsai.es
Aprobación	NA

1.4 Definiciones, acrónimos y abreviaturas

- HTML: Lenguaje de marcado para la elaboración de páginas web
- CSS: Hoja de estilo en cascada es un lenguaje usado para definir y crear la presentación de un documento estructurado escrito en HTML o XML.
- PHP: Lenguaje de programación orientados a objetos, de uso general de código del lado del servidor
- JS: Es un lenguaje de programación interpretado, usado en el lado del cliente
- URL: Es un localizador de recursos uniforme
- AJAX: Es una técnica de desarrollo web para crear aplicaciones interactivas, que se ejecutan en el cliente.
- SQL: Lenguaje declarativo de acceso a base de datos relacionales
- BD: Banco de información que contiene datos.

- UTF-8: Formato de codificación de caracteres latinos.
- HTTP: Es el protocolo de comunicación que permite la transferencia de información en internet
- XML: Lenguaje de marcas utilizado para almacenar datos
- WEB: Conjunto de información que se encuentra en una dirección determinada de internet.
- SMS: Es un servicio disponible en los teléfonos móviles que permite el envío de mensajes cortos
- DNI: Documento nacional de identidad

1.5 Referencias

Referencia	Título	Ruta	Fecha	Autor
Internet	PHP	http://www.w3schools.com/	28/05/2016	W3schools
Internet	JAVA	http://www.w3schools.com/	28/05/2016	W3schools
Internet	HTML	http://www.w3schools.com/	28/05/2016	W3schools
Internet	SQL	http://www.w3schools.com/	28/05/2016	W3schools
Internet	JAVA	http://www.desarrolloweb.com/	28/05/2016	Desconocido
Internet	PHP	http://php.net/manual	28/05/2016	Desconocido
Internet	JAVA, PHP	http://www.forosdelweb.com	28/05/2016	Usuarios

1.6 Resumen

En las siguientes secciones de este documento vamos a hacer una introducción al problema, explicando los detalles que no tienen que ver directamente con la definición del problema.

También se va abordando el problema desde el punto de vista de las necesidades, los requisitos y los problemas que nos ha ido surgiendo mientras realizábamos el desarrollo de este proyecto, así como las soluciones que hemos implementado. Para ello el documento se va a estructurar siguiendo el estándar IEEE830.

2 Descripción general

2.1 Perspectiva del producto

El producto solicitado por el cliente, será una aplicación web, la cual será ejecutada en un servidor de forma independiente, sin necesidad de interactuar con otros sistemas. Será accesible desde la red.

2.2 Funcionalidad del producto

La funcionalidad principal de nuestro producto es gestionar un sistema de citas, el cual tendrá como mínimo dos módulos diferenciados:

1. El módulo de visualización situada en la sala de espera, mostrará el código de usuario, el despacho al que debe acudir, así como la hora y las últimas n personas llamadas.
2. En el módulo de control, diferenciaremos entre tres tipos de pantalla, una pantalla de administrador, una pantalla de profesor y una pantalla de usuario, cada pantalla según sea el usuario tendrá unas funciones habilitadas específicas.

Características de los usuarios

Tipo de usuario	Administrador
Formación	NA
Habilidades	Conocimientos de administración y desarrollo web
Actividades	Dar de alta, baja y modificar a los usuarios en el sistema Dar de alta, baja y modificar todos los recursos del sistema

Tipo de usuario	Profesor
Formación	NA
Habilidades	Conocimientos de informática básica
Actividades	Editar sus datos Dar de alta, baja o modificar un recurso

Tipo de usuario	Alumno
Formación	NA
Habilidades	Conocimientos de informática básica
Actividades	Inscribirse o borrarse de un recurso

2.3 Restricciones

Para esta aplicación web será necesario un servidor, el cual tenga instalado el servidor web apache, mysql y php. Además, como mínimo funcionará en el navegador Firefox.

2.4 Suposiciones y dependencias

Servidor deberá disponer de apache 2.14.18, de php 7.0.5 y de mysql 5.7.11. En el caso de que surjan versiones más recientes de este software descrito y deseen actualizarse, será necesario realizar un mantenimiento.

2.5 Evolución previsible del sistema

En próximas actualizaciones, se implementará un servicio de aviso mediante SMS, así como mejoras y ampliaciones de la interfaz de usuario.

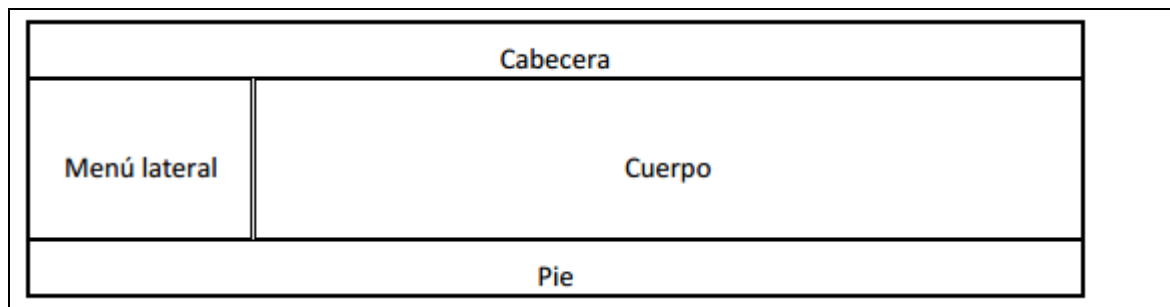
3 Requisitos específicos

3.1 Requisitos comunes de los interfaces

3.1.1 Interfaces de usuario

3.1.1.1 IU-01: Pantalla de inicio

Número de requisito	IU-01
Nombre de requisito	Pantalla de inicio
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito	La pantalla de inicio nos permitirá loguearnos como usuario registrado, ya sea profesor o administrador. Los usuarios no registrados podrán seleccionar la cola que desea y podrán inscribirse, consultar su posición o borrarse de dicha cola.

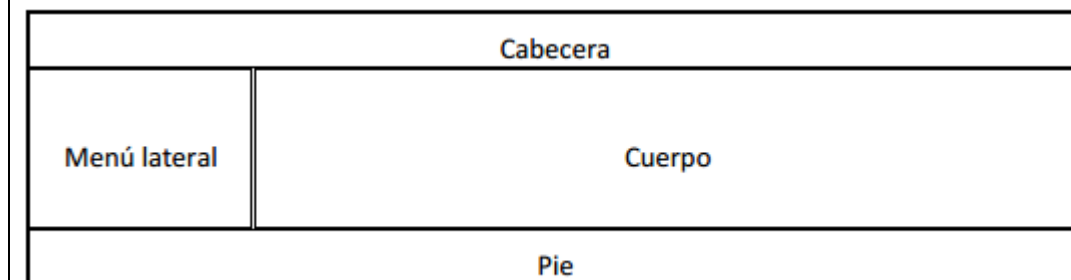


3.1.1.2 IU-03: Pantalla del administrador

Número de requisito	IU-03
Nombre de requisito	Pantalla del administrador
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Descripción del requisito

Pantalla de control del administrador del sistema, donde estarán reflejadas todas las funcionalidades del administrador. El administrador tendrá un submenú donde podrá gestionar todos los usuarios registrados, pudiendo borrar, modificar y dar de alta aun usuario (profesores y administradores). Tendrá otro submenú donde podrá gestionar las colas, pudiendo sólo borrar, modificar y crear una cola. Además de estos dos submenús, tendrá otro submenú donde podrá crear avisos.

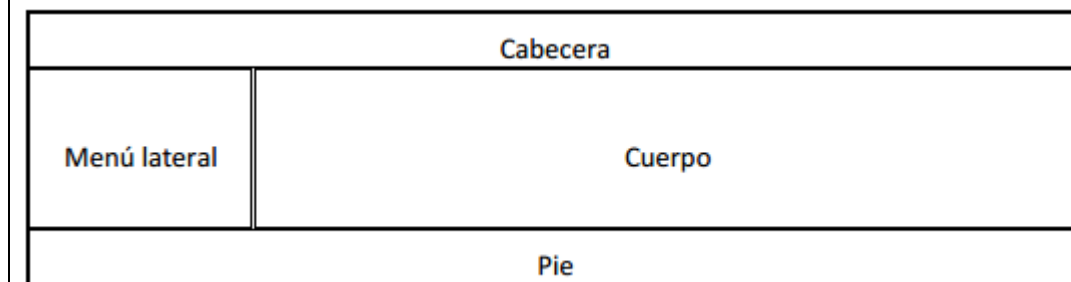


3.1.1.3 IU-04: Pantalla del profesor

Número de requisito	IU-04
Nombre de requisito	Pantalla del profesor
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Descripción del requisito

Pantalla de control del profesor, donde estarán reflejadas todas las funcionalidades del profesor. El profesor tendrá un submenú donde podrá gestionar su perfil, pudiendo borrar y modificar su perfil. Tendrá otro submenú donde podrá gestionar las colas, pudiendo borrar, modificar, activar y crear una cola. Además de estos dos submenús, tendrá otro submenú donde podrá crear avisos.



3.1.1.4 IU-05: Pantalla de usuario no registrado

Número de requisito	IU-05
Nombre de requisito	Pantalla de usuario no registrado
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito Pantalla de control del usuario no registrado, donde estarán reflejadas todas las funcionalidades. El usuario no registrado tendrá un submenú donde podrá apuntarse a la cola seleccionada, consultar su posición en la cola seleccionada o borrarse de dicha cola.	

3.1.1.5 IU-06: Pantalla de espera

Número de requisito	IU-06
Nombre de requisito	Pantalla del administrador
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito Pantalla de espera, en la cual se visualizará la información relativa a la gente de la cola que ha sido llamada y la persona siguiente en ser llamada. También dispondrá de una tabla donde mostrará los avisos.	

Cabecera

Título

Código 1	Lugar 1
Código 2	Lugar 2
...	...

Información auxiliar (mostrar sólo cuando sea necesario)

Pie

3.1.2 Interfaces de hardware

NA. No hay requisitos específicos, cualquier navegador que vaya fluido en un ordenador.

3.1.3 Interfaces de software

Necesitaremos un servicio de servidor que tendrá instalado una base de datos MySQL. También necesitaremos navegador que interprete JavaScript, css y html.

3.1.4 Interfaces de comunicación

Necesitaremos poder hacer comunicaciones usando el protocolo HTTP.

3.2 Requisitos funcionales

3.2.1 Requisito funcional 1: Pantalla inicio

3.2.1.1 Requisito funcional 1.1: Acceso usuario registrado

Número de requisito	RF-1.1
Nombre de requisito	Acceso usuario registrado
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: Opción en la pantalla de inicio que utilizarán el administrador y los profesores que se encuentran registrados en el sistema.	
Comprobación de validez de las entradas: Comprobación de email y de la contraseña, en la base de datos	
Secuencia exacta de operaciones: <ol style="list-style-type: none"> 1. El usuario introducirá su correo y su contraseña en el formulario de acceso. 2. Le dará a enviar 3. Comprobación en la base de datos, de que los datos introducidos son correctos. 4. Será enviado a la pantalla de acceso a usuarios registrados. 	
Respuesta a situaciones anormales: Si el email no es correcto	
Parámetros: Email y contraseña	
Generación de salidas: Se mostrará la pantalla de acceso a usuarios registrados.	
Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida	
Especificación de los requisitos lógicos: NA. No almacenamos información en la base de datos	

3.2.1.2 Requisito funcional 1.2: Acceso usuario no registrado

Número de requisito	RF-1.2
Nombre de requisito	Acceso usuario no registrado
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: Opción en la pantalla de inicio que utilizarán los usuarios no registrados en el sistema y que deseen apuntarse a una cola para revisión.	
Comprobación de validez de las entradas: NA. No hay ningún parámetro que comprobar	
Secuencia exacta de operaciones: <ol style="list-style-type: none"> 1. El usuario seleccionará en la pantalla de inicio la cola que desea ver. 2. Será redireccionado a una nueva ventana, donde se mostrará la información de la cola seleccionada 	
Respuesta a situaciones anormales: NA	
Parámetros: NA. No hay parámetros	
Generación de salidas: Se mostrará una pantalla con la información de la cola	

seleccionada.

Relaciones entre entradas y salidas: NA.

Especificación de los requisitos lógicos: NA

3.2.2 Requisito funcional 2: Pantalla Login usuario registrado

3.2.2.1 Requisito funcional 2.1: Login usuario registrado

Número de requisito	RF-2.1
Nombre de requisito	Login usuario registrado
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
<p>Descripción del requisito: Pantalla con un formulario de acceso al sistema para usuarios registrados, en el cual se le pedirá al usuario que ingrese el email con el que está dado de alta y su password.</p> <p>Comprobación de validez de las entradas: Se comprobará que el email y el password son correctos y buscando si se encuentran almacenados en la base de datos de usuarios registrados en el sistema.</p> <p>Secuencia exacta de operaciones:</p> <ol style="list-style-type: none"> 1. Introducir email. 2. Introducir password. 3. Pichar en el botón acceder. 4. Comprobar email y comprobar password. 5. Comprobación de que el usuario está en la base de datos(RF-2.1.1) 6. Redireccionamiento a la página correspondiente. <p>Respuesta a situaciones anormales: En caso de que el email, el password o ambos no sean correctos el sistema avisará al usuario de este error y le volverá a solicitar dichos parámetros.</p> <p>Parámetros: Email del usuario y password.</p> <p>Generación de salidas: El sistema generará una pantalla con las operaciones disponibles para los usuarios registrados, distinguiéndose entre la pantalla del administrador y la pantalla del profesor.</p> <p>Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida</p> <p>Especificación de los requisitos lógicos: NA se estudiará posteriormente</p>	

3.2.2.1.1 Requisito funcional 2.1.1: Comprobación tipo de usuario

Número de requisito	RF-2.1.1
Nombre de requisito	Comprobación tipo de usuario
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
<p>Descripción del requisito: Comprobación del sistema para saber si se trata de un profesor o un administrador.</p> <p>Comprobación de validez de las entradas: NA.</p> <p>Secuencia exacta de operaciones:</p> <ol style="list-style-type: none"> 1. El sistema busca el rol del usuario registrado en la base de datos de usuarios. 2. Si el usuario esta en el sistema se redireccionará a la página correspondiente. 3. Si eres administrador se enviará a la página del administrador. Si eres profesor se enviará a la página del profesor <p>Respuesta a situaciones anormales: Si el usuario no está en la base de datos, se enviará un mensaje avisando del error</p> <p>Parámetros: NA, no hace falta enviar parámetros</p> <p>Generación de salidas: El sistema mostrará al usuario la interfaz que le corresponda según el rol que tenga en el sistema.</p> <p>Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida</p> <p>Especificación de los requisitos lógicos: NA</p>	

3.2.3 Requisito funcional 3: Pantalla usuario no registrado

3.2.3.1 Requisito funcional 3.1: Unirse a cola

Número de requisito	RF-3.1
Nombre de requisito	Unirse a cola
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
<p>Descripción del requisito: El usuario podrá unirse a una cola para revisión.</p> <p>Comprobación de validez de las entradas: Se comprobará la validez de los</p> <p>Secuencia exacta de operaciones:</p> <ol style="list-style-type: none"> 1. Buscar cola (RF-8.7) 2. Seleccionar cola 3. Introducción de los datos del usuario que se va a registrar. Como mínimo se introducirá el email, el nombre y los apellidos 4. Unirse a cola <p>Respuesta a situaciones anormales: Se lanzará un mensaje de error en caso de que la cola no sea válida</p> <p>Parámetros: Nombre de la cola.</p>	

Generación de salidas: Se generará la pantalla de inscripción en la cola.

Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida

Especificación de los requisitos lógicos: Nombre de la cola almacenado como cadena.

3.2.3.1.1 Requisito funcional 3.1.1: Generar código usuario

Número de requisito	RF-3.1.1
Nombre de requisito	Generar código de usuario
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: Generación del código personal del usuario no registrado	
Comprobación de validez de las entradas: El sistema comprobará que el email y el DNI sean correctos.	
Secuencia exacta de operaciones: <ol style="list-style-type: none"> 1. Introducir DNI 2. Introducir email 3. Introducir nombre 4. Comprobar DNI 5. Comprobar email 6. Generar código 	
Respuesta a situaciones anormales: Si el código ya existe por que el usuario está apuntado en otra cola no lo volverá a generar y podrá utilizarlo en ambas colas	
Parámetros: DNI, nombre, email	
Generación de salidas: NA. No hay relación entre entrada y salida	
Relaciones entre entradas y salidas: Código compuesto por la concatenación de los tres primeros caracteres del nombre y los cuatro últimos del DNI del usuario, dando como salida el código	
Especificación de los requisitos lógicos: NA	

3.2.4 Requisito funcional 4: Acceder usando código

3.2.4.1 Requisito funcional 4.1: Comprobar código

Número de requisito	RF-4.1
Nombre de requisito	Comprobar código
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: Comprobación del código de usuario	
Comprobación de validez de las entradas: El sistema comprobará que el código de usuario introducido es valido	
Secuencia exacta de operaciones: <ol style="list-style-type: none"> 1. Introducido código 2. Se comprueba que el código este en la base de datos. 3. Redirecciona a la página en la que estaba el usuario 	
Respuesta a situaciones anormales: En caso de que el código introducido no sea	

correcto o tenga menos caracteres de los que le corresponden avisará al usuario

Parámetros: Código de usuario

Generación de salidas: NA. Se comprueba internamente sin necesidad de mostrar nada

Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida

Especificación de los requisitos lógicos: NA

3.2.4.2 Requisito funcional 4.2: Consultar información de la cola

Número de requisito	RF-4.2
Nombre de requisito	Consultar información de la cola
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: Mostrar información de la cola	
Comprobación de validez de las entradas: NA, no tiene parámetros que comprobar	
Secuencia exacta de operaciones: <ol style="list-style-type: none"> 1. Seleccionar una cola (RF-8.7) 2. El sistema mostrará nombre de la cola 3. El sistema mostrará lugar de revisión 4. El sistema mostrará posición relativa 	
Respuesta a situaciones anormales: NA, no es necesario ya que si no existiera la cola no podrías seleccionarla	
Parámetros: NA	
Generación de salidas: Generación de la información de la cola y el estado del usuario en esta en el momento que consulta dicha información.	
Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida	
Especificación de los requisitos lógicos: NA	

3.2.4.3 Requisito funcional 4.3: Darse de baja de la cola

Número de requisito	RF-4.3
Nombre de requisito	Darse de baja de la cola
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: El usuario se puede dar de baja en una cola	
Comprobación de validez de las entradas: Comprobará que el código y el email sean correctos.	
Secuencia exacta de operaciones: <ol style="list-style-type: none"> 1. El usuario selecciona una cola. 2. El usuario seleccionará darse de baja en una cola. 3. Introducirá su código de revisión y su email. 4. El sistema lo eliminará de la cola. 	

Respuesta a situaciones anormales: Si el código o el email no son correctos, se mandará un mensaje diciendo que el email o el código no existe

Parámetros: Código y email

Generación de salidas: El usuario ya no formará parte de la cola y se le mostrará un mensaje de que se ha dado de baja de esta

Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida

Especificación de los requisitos lógicos: NA

3.2.5 Requisito funcional 5: Pantalla del Administrador

3.2.5.1 Requisito funcional 5.1: Añadir usuario

Número de requisito	RF-5.1
Nombre de requisito	Añadir usuario
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: Añadir un usuario al sistema	
Comprobación de validez de las entradas: Se comprobará que el DNI y el email se ha correcto	
Secuencia exacta de operaciones: <ol style="list-style-type: none"> 1. Seleccionar añadir un nuevo usuario 2. Añadir el nombre 3. Añadir el email 4. Añadir password 5. Comprobar email (RF-8.2) 	
Respuesta a situaciones anormales: Si no se ha podido añadir al usuario correctamente el sistema mostrará un mensaje indicándolo y volverá a solicitar los datos del nuevo usuario.	
Parámetros: Nombre y email	
Generación de salidas: Si ha sido correcto el sistema incluirá al nuevo usuario en la base de datos de usuarios registrados	
Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida	
Especificación de los requisitos lógicos: email, nombre y DNI almacenados como cadena en la base de datos	

3.2.5.2 Requisito funcional 5.2: Borrar usuario

Número de requisito	RF-5.2
Nombre de requisito	Borrar usuario
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: Borrar un usuario del sistema	
Comprobación de validez de las entradas: NA. Como el profesor ya está en la base de datos no hace falta comprobar nada.	

Secuencia exacta de operaciones:

1. Buscar un profesor (RF-8.8)
2. Seleccionar eliminar profesor
3. Confirmar borrado(RF-8.3)

Respuesta a situaciones anormales: Se informará con un mensaje de error en caso de que el usuario no se pueda borrar.

Parámetros: Nombre profesor

Generación de salidas: El sistema generará una ventana de aviso de borrado de dicho usuario.

Relaciones entre entradas y salidas: NA. Si el profesor no existiera no podrías seleccionarlo.

Especificación de los requisitos lógicos: NA

3.2.5.3 Requisito funcional 5.3: Modificar usuario

Número de requisito	RF-5.3
Nombre de requisito	Modificar usuario
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: Modificación de los datos del usuario	
Comprobación de validez de las entradas: Comprobar que el email sea correcto	
Secuencia exacta de operaciones: <ol style="list-style-type: none">4. Seleccionar modificar profesor o seleccionar su usuario5. Buscar un profesor (RF-8.8) en el caso de desear modificar un profesor6. Modificar nombre del usuario (RF-5.3.1)7. Modificar email (RF-5.3)8. Modificar password (RF-8.8)9. Darle a modificar.	
Respuesta a situaciones anormales: Si el email no es correcto que envíe un aviso de que debe cambiar el email.	
Parámetros: Nombre, email, password	
Generación de salidas: NA. No es necesario generar salida	
Relaciones entre entradas y salidas: NA. Si el profesor no existiera no podrías seleccionarlo.	
Especificación de los requisitos lógicos: NA	

3.2.5.3.1 Requisito funcional 5.3.1: Modificar nombre

Número de requisito	RF-5.3.1
Nombre de requisito	Modificar nombre
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente

Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: El administrador podrá modificar el nombre asociado a un usuario	
Comprobación de validez de las entradas: NA. No tiene entradas	
Secuencia exacta de operaciones: <ol style="list-style-type: none"> 1. Se seleccionará el usuario que se desea modificar 2. Se introducirá un nuevo nombre 3. Se actualizará el nombre 	
Respuesta a situaciones anormales: NA. Todo está muy encasillado es complicado que se produzca situaciones anormales	
Parámetros: Nombre	
Generación de salidas: NA, no es necesario generar salida	
Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida	
Especificación de los requisitos lógicos: NA	

3.2.5.3.2 Requisito funcional 5.3.2: Modificar email

Número de requisito	RF-5.3.2
Nombre de requisito	Modificar email
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: El administrador podrá modificar el email asociado a un usuario	
Comprobación de validez de las entradas: Se comprobará que el email nuevo sea válido	
Secuencia exacta de operaciones: <ol style="list-style-type: none"> 1. Se introducirá un nuevo email 2. Se comprobará que dicho email es válido(RF-8.2) 3. Se actualizará el email 	
Respuesta a situaciones anormales: Si el email no es correcto se le mandará un mensaje avisándole del correo es erróneo y se le solicitará que ingrese de nuevo el correo	
Parámetros: Email	
Generación de salidas: NA. No produce salida	
Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida	
Especificación de los requisitos lógicos: NA	

3.2.5.4 Requisito funcional 5.4: Crear cola

Número de requisito	RF-5.4
Nombre de requisito	Crear cola
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
<p>Descripción del requisito: El administrador podrá crear una cola a petición de un profesor.</p> <p>Comprobación de validez de las entradas: Se comprobará que la fecha y la hora este en un formato correcto.</p> <p>Secuencia exacta de operaciones:</p> <ol style="list-style-type: none"> 1. Se seleccionará crear una cola 2. Ingresaremos un profesor asociado a dicha cola (RF-5.4.1), 3. Ingresaremos un lugar de revisión 4. Ingresaremos una fecha de revisión 5. Ingresaremos un horario de revisión <p>Respuesta a situaciones anormales: Si la cola a crear ya existe, se enviará un aviso informado de la existencia de la cola.</p> <p>Parámetros: Profesor, lugar de revisión, fecha de revisión, hora de revisión</p> <p>Generación de salidas: NA. No produce salida</p> <p>Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida</p> <p>Especificación de los requisitos lógicos: NA</p>	

3.2.5.4.1 Requisito funcional 5.4.1: Profesor asociado a la cola

Número de requisito	RF-5.4.1
Nombre de requisito	Profesor asociado a la cola
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
<p>Descripción del requisito: El administrador cuando creé la cola deberá asignar un profesor a dicha cola.</p> <p>Comprobación de validez de las entradas: NA. Como los profesores ya están en la base de datos no hace falta comprobar nada.</p> <p>Secuencia exacta de operaciones: Se busca un profesor determinado en la cola de profesor (RF-8.8) y luego se añade dicho profesor al campo profesor.</p> <p>Respuesta a situaciones anormales: NA. No se contempla situaciones anormales</p> <p>Parámetros: NA. No tiene parámetros</p> <p>Generación de salidas: NA. No genera salida</p> <p>Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida</p> <p>Especificación de los requisitos lógicos: NA</p>	

3.2.5.5 Requisito funcional 5.5: Borrar cola

Número de requisito	RF-5.5
Nombre de requisito	Borrar cola
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: El administrador podrá a petición del profesor, borrar una cola. Comprobación de validez de las entradas: NA, como seleccionamos una cola que está en la base de datos, no hace falta comprobar nada. Secuencia exacta de operaciones: <ol style="list-style-type: none"> 1. Se buscará la cola a borrar 2. Se seleccionará dicha cola 3. Se dará a borrar cola 4. El sistema lanzará una ventana para confirmar el borrado de la cola (RF-8.3). Respuesta a situaciones anormales: NA. No se contempla situaciones anormales Parámetros: NA. Cuando seleccionas la cola internamente estas trabajando con los parámetros. Generación de salidas: Generará una ventana para confirmar el borrado (RF-8.3) Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida Especificación de los requisitos lógicos: NA	

3.2.5.6 Requisito funcional 5.6: Modificar cualquier cola

Número de requisito	RF-5.6
Nombre de requisito	Modificar cola
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: El administrador podrá modificar los datos de una cola ya creada Comprobación de validez de las entradas: NA. Como la cola ya está en la base de datos no hace falta comprobar nada. Secuencia exacta de operaciones: <ol style="list-style-type: none"> 1. El administrador accederá a la pantalla de edición de la cola 2. Podrá editar el profesor asociado (RF-5.6.1) 3. Podrá editar el lugar de revisión (RF-8.4) 4. Podrá editar el horario (RF-8.5) Respuesta a situaciones anormales: NA. No se contempla situaciones anormales Parámetros: Nombre cola Generación de salidas: NA. No genera salida Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida Especificación de los requisitos lógicos: NA	

3.2.5.6.1 Requisito funcional 5.6.1: Modificar profesor asociado

Número de requisito	RF-5.6.1
Nombre de requisito	Modificar profesor asociado
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: El administrador podrá modificar el profesor asociado a una cola. Comprobación de validez de las entradas: NA. Como los profesores ya están en la base de datos no hace falta comprobar nada. Secuencia exacta de operaciones: <ol style="list-style-type: none"> 1. Se busca un profesor determinado en la cola de profesor (RF-8.7) 2. Se selecciona el profesor deseado. 3. Se modifica el profesor asociado a esta cola por el nuevo profesor. Respuesta a situaciones anormales: NA. No se contempla situaciones anormales. Parámetros: Profesor Generación de salidas: NA. No genera salida Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida Especificación de los requisitos lógicos: NA	

3.2.6 Requisito funcional 6: Pantalla del profesor

3.2.6.1 Requisito funcional 6.1: Modificar mi usuario

Número de requisito	RF-6.1
Nombre de requisito	Modificar mi usuario
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: El profesor podrá modificar los datos asociados a su usuario Comprobación de validez de las entradas: NA. Como el profesor ya está en la base de datos no hace falta comprobar nada. Secuencia exacta de operaciones: <ol style="list-style-type: none"> 1. El profesor accederá a la pantalla de edición de su perfil 2. Podrá editar su nombre(RF-6.1.1) 3. Podrá editar su email (RF-6.1.2). 4. Podrá editar su password (RF-8.8) Respuesta a situaciones anormales: NA. No se contempla situaciones anormales Parámetros: NA Generación de salidas: NA. No genera salida Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida Especificación de los requisitos lógicos: NA	

3.2.6.1.1 Requisito funcional 6.1.1: Modificar mi nombre

Número de requisito	RF-6.1.1
Nombre de requisito	Modificar mi nombre
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: El profesor podrá modificar el nombre asociado a su usuario Comprobación de validez de las entradas: NA. Secuencia exacta de operaciones: <ol style="list-style-type: none"> 1. Se introducirá un nuevo nombre 2. Se actualizará el nombre Respuesta a situaciones anormales: NA. No se contempla situaciones anormales Parámetros: Nombre Generación de salidas: NA. No genera salida Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida Especificación de los requisitos lógicos: NA	

3.2.6.1.2 Requisito funcional 6.1.2: Modificar mi email

Número de requisito	RF-6.1.2
Nombre de requisito	Modificar email
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: El profesor podrá modificar el email asociado a su usuario Comprobación de validez de las entradas: Se comprobará que el email nuevo sea correcto Secuencia exacta de operaciones: <ol style="list-style-type: none"> 1. Se ingresará un nuevo email 2. Se comprobará que dicho email es válido(RF-8.2) 3. Se actualizará el email Respuesta a situaciones anormales: Si el email no es correcto se le mandará un mensaje avisándole del correo es erróneo y se le solicitará que ingrese de nuevo el correo Parámetros: email Generación de salidas: NA. No genera salida Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida Especificación de los requisitos lógicos: NA	

3.2.6.2 Requisito funcional 6.2: Crear cola propia

Número de requisito	RF-6.2
Nombre de requisito	Crear cualquier cola
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: El profesor podrá crear una cola para la revisión de una asignatura. Comprobación de validez de las entradas: NA. No hay entradas Secuencia exacta de operaciones: <ol style="list-style-type: none"> 1. Se selecciona crear cola 2. Ingresaremos un lugar de revisión 3. Ingresaremos un horario de revisión 4. Le damos a crear revisión Respuesta a situaciones anormales: NA. No se contempla situaciones anormales Parámetros: Lugar de revisión, hora de revisión. Generación de salidas: Genera un aviso de que la cola se ha generado con éxito. Relaciones entre entradas y salidas: NA. No genera salida Especificación de los requisitos lógicos: NA	

3.2.6.3 Modificar mis colas

Número de requisito	RF-6.3
Nombre de requisito	Modificar cualquier cola
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: El profesor podrá modificar los datos de una cola ya creada Comprobación de validez de las entradas: NA, no hay parámetros Secuencia exacta de operaciones: <ol style="list-style-type: none"> 1. El profesor accederá a la pantalla de edición 2. Podrá editar el lugar de revisión (RF-8.4) 3. Podrá editar el horario (RF-8.5) Respuesta a situaciones anormales: NA. No se contempla situaciones anormales Parámetros: Cola Generación de salidas: NA. No genera salida Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida Especificación de los requisitos lógicos: NA	

3.2.6.4 Requisito funcional 6.5: Siguiente turno

Número de requisito	RF-6.4
Nombre de requisito	Siguiente turno
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
<p>Descripción del requisito: El profesor una vez terminada una revisión, llamará al siguiente en la lista.</p> <p>Comprobación de validez de las entradas: NA. Como los alumnos están en la base de datos, no hace falta comprobar nada.</p> <p>Secuencia exacta de operaciones:</p> <ol style="list-style-type: none"> 1. Cuando se pulsa el botón siguiente 2. El que estaba siendo atendido pasa a atendido. 3. El sistema procederá a llamar al siguiente en la cola. <p>Respuesta a situaciones anormales: Si no hay más alumnos no se muestra nada en la tabla.</p> <p>Parámetros: NA, no hay parámetros</p> <p>Generación de salidas: Mostrará por pantalla el siguiente usuario en ser atendido.</p> <p>Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida</p> <p>Especificación de los requisitos lógicos: NA</p>	

3.2.6.5 Requisito funcional 6.5: Turno en espera

Número de requisito	RF-6.5
Nombre de requisito	Turno en espera
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
<p>Descripción del requisito: El profesor podrá mantener a un usuario no registrado en espera, en el caso de que este, tenga que acudir a otra cola diferente y después deba volver a la cola origen, sin necesidad de esperar nuevamente turno.</p> <p>Comprobación de validez de las entradas: NA. Como los alumnos están en la base de datos, no hace falta comprobar nada.</p> <p>Secuencia exacta de operaciones:</p> <ol style="list-style-type: none"> 1. Se marcará al usuario llamado en estado de espera. 2. El usuario en espera volverá a la lista de gente no atendida 3. Pasado un turno se volverá a llamar <p>Respuesta a situaciones anormales: NA. No se contempla situaciones anormales</p> <p>Parámetros: NA, no hay parámetros</p> <p>Generación de salidas: Mostrará por pantalla el siguiente usuario en ser atendido.</p> <p>Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida</p> <p>Especificación de los requisitos lógicos: NA</p>	

3.2.7 Requisito funcional 7: Pantalla de espera

3.2.7.1 Requisito funcional 7.1: Lista de últimos llamados

Número de requisito	RF-7.1
Nombre de requisito	Lista de últimos llamados
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
<p>Descripción del requisito: El sistema mostrará por pantalla la lista de los últimos usuarios que han sido llamados</p> <p>Comprobación de validez de las entradas: NA. Como los alumnos están en la base de datos, no hace falta comprobar nada.</p> <p>Secuencia exacta de operaciones: Muestra por pantalla una lista con los usuarios que han sido llamados, por cada usuario se mostrará:</p> <ol style="list-style-type: none"> 1. Código usuario 2. Lugar de revisión 3. Hora de llamada <p>Respuesta a situaciones anormales: NA. No se contempla situaciones anormales</p> <p>Parámetros: NA. No hay parámetros</p> <p>Generación de salidas: NA. No genera salida</p> <p>Relaciones entre entradas y salidas: Genera una pantalla con esta información.</p> <p>Especificación de los requisitos lógicos: NA</p>	

3.2.8 Requisito funcional 8: Requisitos generales

3.2.8.1 Requisito funcional 8.1: Comprobación de DNI

Número de requisito	RF-8.1
Nombre de requisito	Comprobación del DNI
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
<p>Descripción del requisito: Comprobación de la estructura de un DNI</p> <p>Comprobación de validez de las entradas: El sistema comprobará que el DNI introducido por el usuario tiene 8 dígitos y un carácter alfabético.</p> <p>Secuencia exacta de operaciones:</p> <ol style="list-style-type: none"> 1. El sistema comprobará si la estructura del DNI es correcta. 2. El sistema comprobará que el DNI este en el sistema. <p>Respuesta a situaciones anormales: Si no se trata de un DNI válido se mostrará un mensaje al usuario indicándoselo</p> <p>Parámetros: NA. No hay parámetros</p>	

Generación de salidas: NA. No genera salida

Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida

Especificación de los requisitos lógicos: NA

3.2.8.2 Requisito funcional 8.2: Comprobación de email

Número de requisito	RF-8.2
Nombre de requisito	Comprobación del email
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: Comprobación de la estructura del email	
Comprobación de validez de las entradas: <ol style="list-style-type: none"> 1. El sistema comprobará que el email introducido por el usuario tiene n caracteres alfanuméricos @ n caracteres alfanuméricos y una extensión. 2. El sistema comprobará que el email este en el sistema. 	
Secuencia exacta de operaciones: El sistema comprobará si la estructura del email es correcta.	
Respuesta a situaciones anormales: Si no se trata de un email válido se mostrará un mensaje al usuario indicándoselo	
Parámetros: NA. No hay parámetros	
Generación de salidas: NA. No genera salida	
Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida	
Especificación de los requisitos lógicos: NA	

3.2.8.3 Requisito funcional 8.3: Confirmación de borrado

Número de requisito	RF-8.3
Nombre de requisito	Confirmación de borrado
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Desarrollador
Prioridad del requisito	<input type="checkbox"/> Alta/Esencial <input checked="" type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: El sistema mostrará un aviso, pidiendo confirmar el borrado de la cola	
Comprobación de validez de las entradas: NA, no hay entrada	
Secuencia exacta de operaciones: <ol style="list-style-type: none"> 1. El sistema lanzará una ventana de aviso de borrado 2. El usuario aceptará o rechazará borrar la cola 3. En caso de aceptar el sistema eliminará la información de la cola (RF-8.3.1) 4. En caso de rechazar se volverá al estado anterior 	
Respuesta a situaciones anormales: NA. No se contempla situaciones anormales	
Parámetros: Nombre de la cola	
Generación de salidas: El sistema lanzará un aviso de que el borrado se ha realizado	

con éxito

Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida

Especificación de los requisitos lógicos: NA

3.2.8.3.1 Requisito funcional 8.3.1: Eliminar información de la cola

Número de requisito	RF-8.3.1
Nombre de requisito	Eliminar información de la cola
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: El sistema borrará toda la información asociada a la cola.	
Comprobación de validez de las entradas: NA. Como las colas están en la base de datos, no hace falta comprobar nada.	
Secuencia exacta de operaciones: La información asociada a la cola será eliminada de forma automática. <ol style="list-style-type: none"> 1. Se buscará en la base de datos la cola 2. Una vez encontrada se procederá a su borrado 	
Respuesta a situaciones anormales: Si la información no se puede borrar mandará un aviso	
Parámetros: NA. No hay parámetros	
Generación de salidas: NA. No genera salida	
Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida	
Especificación de los requisitos lógicos: NA	

3.2.8.4 Requisito funcional 8.4: Modificar lugar de revisión

Número de requisito	RF-8.4
Nombre de requisito	Modificar lugar de revisión
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: Tanto el profesor como el administrador podrá modificar el lugar de revisión asociada a una cola.	
Comprobación de validez de las entradas: NA. No hay entradas	
Secuencia exacta de operaciones: <ol style="list-style-type: none"> 1. Se introducirá un lugar de revisión 2. Se actualizará el lugar de revisión 	
Respuesta a situaciones anormales: NA. No se contempla situaciones anormales	
Parámetros: Lugar de revisión	

Generación de salidas: NA. No genera salida

Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida

Especificación de los requisitos lógicos: NA

3.2.8.5 Requisito funcional 8.5: Modificar horario

Número de requisito	RF-8.5
Nombre de requisito	Modificar horario
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: Tanto el profesor como el administrador podrá modificar la hora de la revisión.	
Comprobación de validez de las entradas: NA. No hay parámetros	
Secuencia exacta de operaciones: <ol style="list-style-type: none"> 1. Se introducirá una hora de comienzo. 2. Se introducirá una hora de finalización. 3. Se actualizará la hora de comienzo. 4. Se actualizará la hora de finalización. 	
Respuesta a situaciones anormales: NA. No se contempla situaciones anormales	
Parámetros: Hora inicio, hora final	
Generación de salidas: NA. No genera salida	
Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida	
Especificación de los requisitos lógicos: NA	

3.2.8.6 Requisito funcional 8.6: Crear aviso

Número de requisito	RF-8.6
Nombre de requisito	Crear aviso
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: Creación de un aviso por parte de un administrador o un profesor	
Comprobación de validez de las entradas: NA. Como los profesores están en la base de datos, no hace falta comprobar nada.	
Secuencia exacta de operaciones: <ol style="list-style-type: none"> 1. El profesor o administrador seleccionarán crear aviso 2. Escribirán el aviso 3. Enviarán el aviso 4. Se rellenará el nombre con el nombre del profesor que ha creado el aviso 5. En caso de ser el administrador, el seleccionará el profesor que desea enviar el mensaje. 	

Respuesta a situaciones anormales: NA. No se contempla situaciones anormales

Parámetros: Texto del aviso, profesor

Generación de salidas: Se enviará el aviso que se mostrará en la pantalla de la sala de espera.

Relaciones entre entradas y salidas: Se enviará el aviso y se mostrará en la pantalla de la sala de espera.

Especificación de los requisitos lógicos: NA

3.2.8.7 Requisito funcional 8.7: Lista colas

Número de requisito	RF-8.7
Nombre de requisito	Listar colas
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: Lista de las colas que se encuentran dadas de alta en el sistema.	
Comprobación de validez de las entradas: NA	
Secuencia exacta de operaciones:	
<ol style="list-style-type: none"> 1. El sistema mostrará una lista con todas las colas disponibles 2. Las listas se ordenarán por fecha 	
Respuesta a situaciones anormales: Si no hay ninguna cola el sistema lo indicará mediante un aviso	
Parámetros: NA. No hay parámetros	
Generación de salidas: NA. No genera salida	
Relaciones entre entradas y salidas: Lista de colas	
Especificación de los requisitos lógicos: NA	

3.2.8.8 Requisito funcional 8.8: Modificar password

Número de requisito	RF-8.8
Nombre de requisito	Modificar password
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: Tanto el profesor como el administrador, podrán modificar el password asociado a su usuario	
Comprobación de validez de las entradas: NA. Como los profesores y los administradores están en la base de datos, no hace falta comprobar nada.	
Secuencia exacta de operaciones:	
<ol style="list-style-type: none"> 1. Se ingresará un nuevo password 	

2. Se actualizará el password

Respuesta a situaciones anormales: NA. No se contempla situaciones anormales

Parámetros: password

Generación de salidas: NA. No genera salida

Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida

Especificación de los requisitos lógicos: NA

3.2.8.9 Requisito funcional 8.9: Buscar profesor

Número de requisito	RF-8.9
Nombre de requisito	Buscar profesor
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Desarrollador
Prioridad del requisito	<input type="checkbox"/> Alta/Esencial <input checked="" type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción del requisito: El administrador buscará en la cola de profesores, el profesor deseado.	
Comprobación de validez de las entradas: NA. Como los profesores están en la base de datos, no hace falta comprobar nada.	
Secuencia exacta de operaciones: Se busca al profesor deseado y se devuelve el nombre del profesor	
Respuesta a situaciones anormales: NA. No se contempla situaciones anormales	
Parámetros: Lista de profesores	
Generación de salidas: Devuelve un string con el nombre del profesor	
Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida	
Especificación de los requisitos lógicos: NA	

3.3 Requisitos no funcionales

3.3.1 Requisitos de rendimiento

El sistema deberá soportar un número 70 de conexiones, así como una alta disponibilidad. El sistema deberá de ser lo suficientemente ágil y rápido para minimizar los tiempos de espera, que deberán soportar los distintos tipos de usuarios al realizar cualquier tipo de gestión. Cada transacción no debe superar los 5 segundos

3.3.2 Seguridad

Los datos sensibles como passwords serán enviados a la base de datos encriptados mediante algoritmos MD5. Se implementa un sistema de privilegios, para que los usuarios sólo puedan acceder a las partes que tengan privilegios.

3.3.3 Fiabilidad

Uno de los objetivos más importantes por parte de nuestro sistema es la fiabilidad. Nuestro sistema debe ser capaz de recuperarse frente a los distintos fallos que pueden surgir, ya sea fallo en la transferencia de datos, transferencias incompletas, etc.

Este objetivo es fundamental, ya que, si el sistema acumula muchos fallos, el usuario empezará a cansarse y terminará dejando de usar la aplicación.

3.3.4 Disponibilidad

Nuestro objetivo es conseguir una tasa de disponibilidad plena, salvo fallos inesperados en la red, ajenos a la aplicación. En caso de tener que hacer mantenimiento este durará lo mínimo posible.

3.3.5 Mantenibilidad

Cada mes se hará un mantenimiento preventivo, revisando bases de datos y funcionamiento general del sistema. Este mantenimiento será llevado a cabo por los administradores

3.3.6 Portabilidad

Nuestra aplicación funcionará de forma nativa en Firefox, pero además podrá utilizarse en el resto de los navegadores. Nuestra aplicación utiliza un servidor Apache con sus correspondientes versiones de mysql y php. En caso de que actualizaciones del php y de mysql, el sistema será adaptado a dichas versiones.

3.4 Otros requisitos

NA

4 Descripción de la solución

4.1.1 Interfaces de usuario

4.1.1.1 IU-01: Pantalla de inicio

Número de requisito	IU-01
Nombre de requisito	Pantalla de inicio
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Descripción del requisito

La pantalla de inicio nos permitirá loguearnos como usuario registrado, ya sea profesor o administrador. Los usuarios no registrados podrán seleccionar la cola que desea y podrán inscribirse, consultar su posición o borrarse de dicha cola.

Se han usado los estilos de la universidad de granada, se han usado el banner superior del departamento de ciencias de la computación e inteligencia artificial y el banner inferior de la universidad de granada.

Los archivos utilizados son index.php, los estilos de la carpeta de css y el archivo ejercicio.js de la carpeta js.



Universidad de Granada



4.1.1.2 IU-03: Pantalla del administrador

Número de requisito	IU-03
Nombre de requisito	Pantalla del administrador
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Descripción del requisito

Pantalla de control del administrador del sistema, donde estarán reflejadas todas las funcionalidades del administrador. El administrador tendrá un submenú donde podrá gestionar todos los usuarios registrados, pudiendo borrar, modificar y dar de alta a un usuario (profesores y administradores). Tendrá otro submenú donde podrá gestionar las colas, pudiendo sólo borrar, modificar y crear una cola. Además de estos dos submenús, tendrá otro submenú donde podrá crear avisos.

Se han usado los estilos de la universidad de granada, se han usado el banner superior del departamento de ciencias de la computación e inteligencia artificial y el banner inferior de la universidad de granada.

Los archivos utilizados son PaginaAdministrador.php, los estilos de la carpeta css y el script ejercicio.js de la carpeta js. Cada una de las funcionalidades del submenú se están en los archivos con DichaFuncionAdministrador.php



4.1.1.3 IU-04: Pantalla del profesor

Número de requisito	IU-04
Nombre de requisito	Pantalla del profesor
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Descripción del requisito

Pantalla de control del profesor, donde estarán reflejadas todas las funcionalidades del profesor. El profesor tendrá un submenú donde podrá gestionar su perfil, pudiendo borrar y modificar su perfil. Tendrá otro submenú donde podrá gestionar las colas, pudiendo borrar, modificar, activar y crear una cola. Además de estos dos submenús, tendrá otro submenú donde podrá crear avisos.

Se han usado los estilos de la universidad de granada, se han usado el banner superior del departamento de ciencias de la computación e inteligencia artificial y el banner inferior de la universidad de granada.

Los archivos utilizados son PaginaProfesor.php, los estilos de la carpeta css y el script ejercicio.js de la carpeta js. Cada una de las funcionalidades del submenú se están en los archivos con DichaFuncionProfesor.php



4.1.1.4 IU-05: Pantalla de usuario no registrado

Número de requisito	IU-05
Nombre de requisito	Pantalla de usuario no registrado
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Descripción del requisito

Pantalla de control del usuario no registrado, donde estarán reflejadas todas las funcionalidades. El usuario no registrado tendrá un submenú donde podrá apuntarse a la cola seleccionada, consultar su posición en la cola seleccionada o borrarse de dicha cola.

Se han usado los estilos de la universidad de granada, se han usado el banner superior del departamento de ciencias de la computación e inteligencia artificial y el banner inferior de la universidad de granada.

Los archivos utilizados son VerMiPosicion.php, los estilos de la carpeta css y el script ejercicio.js de la carpeta js.



4.1.1.5 IU-06: Modulo visualización

Número de requisito	IU-06
Nombre de requisito	Pantalla del administrador
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional

Descripción del requisito

Pantalla de espera, en la cual se visualizará la información relativa a la gente de la cola que ha sido llamada y la persona siguiente en ser llamada. También dispondrá de una tabla donde mostrará los avisos.

Se han usado los estilos de la universidad de granada, se han usado el banner superior del departamento de ciencias de la computación e inteligencia artificial y el banner inferior de la universidad de granada.

Los archivos utilizados son ModuloVisualizacion.php, los estilos de la carpeta css y el script ejercicio.js de la carpeta js.



ALUMNO	LUGAR	ASIGNATURA	ESTADO
ocohcbac	A2.5	saPlsaka	No atendido

PROFESOR	MENSAJE
adwer@zvcv	Hola que tal como estamos todos, espero que bien, pablo es un despistado
totidos@hotmail.com	Esto es una prueba para ver si esto funciona bastante bien y se amplia de narices el cuadro de texto y así poder ver todo el mensaje

DECSAI - Departamento de Ciencias de la Computación e Inteligencia Artificial - Acceso Identificado

4.2 Requisitos funcionales

4.2.1 Requisito funcional 1: Pantalla inicio

4.2.1.1 Requisito funcional 1.1: Pantalla inicio

Número de requisito	RF-1.2
Nombre de requisito	Pantalla inicio
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
<p>Descripción del requisito: En la pantalla de inicio encontraremos una tabla con las colas de revisiones que hay en el departamento. El usuario clicando sobre los datos de la cola, podrá acceder a la cola. Debajo del submenú estará el formulario de login.</p> <p>Comprobación de validez de las entradas: NA. No hay ningún parámetro que comprobar</p> <p>Secuencia exacta de operaciones:</p> <ol style="list-style-type: none"> 1. Usuario registrado(RF-2.1). 2. Usuarios no registrados <ol style="list-style-type: none"> 1. El usuario seleccionará en la pantalla de inicio la cola que desea ver. 2. Cuando seleccionamos la cola, mediante un enlace enviamos la url de la página a la que nos redirecciona y el código asociado a la cola. 3. Será redireccionado a una nueva ventana, donde se mostrará la información de la cola seleccionada. En dicha página tendrás un submenú compuesto por dos opciones, buscar la posición en la cola o unirte a la cola <p>Respuesta a situaciones anormales: NA.</p> <p>Parámetros: Código</p> <p>Generación de salidas: Se mostrará una pantalla con la información de la cola seleccionada.</p> <p>Relaciones entre entradas y salidas: NA. No hay relación entre entradas y salidas</p> <p>Especificación de los requisitos lógicos: NA. No almacenamos información en la base de datos</p>	

4.2.2 Requisito funcional 2: Pantalla Login usuario registrado

Número de requisito	RF-2
Nombre de requisito	Login usuario registrado
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
<p>Descripción del requisito: En la pantalla de inicio tendremos un formulario debajo del submenú, donde nos podremos loguear. Este formulario se compone de un campo para introducir el email y un campo para introducir el password, además dispone de un botón para enviar la información</p> <p>Comprobación de validez de las entradas: Mediante una función javascript se comprobará que el email y la contraseña no estén vacíos.</p> <p>Secuencia exacta de operaciones:</p> <ol style="list-style-type: none"> 1. El usuario introducirá su correo y su contraseña en el formulario de acceso. 2. Le dará a enviar 3. Una vez le dé a enviar, mediante un post enviaremos el email y la contraseña a la página login. php, para su procesamiento 4. Comprobación en la base de datos mediante un select, de que los datos introducidos existen en la base de datos. 5. Crearemos un \$_SESSION con los datos del usuario. 6. Comprobará que tipo de rol (Administrador/Profesor) tiene. (RF-2.1.1) 7. Si es profesor será enviada a la pantalla de inicio de profesor. 8. Si es administrador será enviado a la pantalla de inicio de administrador. <p>Respuesta a situaciones anormales: Si no está registrado en la base de datos devuelve un aviso.</p> <p>Parámetros: Email y contraseña</p> <p>Generación de salidas: Se mostrará la pantalla de inicio del profesor o del administrador. Se mostrará un alert en caso de que el usuario no esté en la base de datos.</p> <p>Relaciones entre entradas y salidas: Recibido el password hay que hacerle un md5 para comprobar que el password coincide con el password almacenado en la base de datos</p> <p>Especificación de los requisitos lógicos: NA. No almacenamos información en la base de datos</p>	

4.2.3 Requisito funcional 3: Pantalla usuario no registrado

Número de requisito	RF-3
Nombre de requisito	Pantalla usuario no registrado
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
<p>Descripción del requisito: Una vez seleccionada una cola, se enviará a esta página donde se mostrará mediante una tabla, la información de la cola seleccionada. Esta pantalla dispondrá de un submenú donde podremos darnos de alta en la cola o ver la posición que tenemos en dicha cola.</p> <p>Comprobación de validez de las entradas: Se comprobará que el código se encuentre en la base de datos.</p> <p>Secuencia exacta de operaciones:</p> <ol style="list-style-type: none"> 1. Extraemos con un GET el código enviado por url y lo grabamos es una variable 2. Haremos una consulta en la base de datos, con el código que hemos recibido. 	

3. Una vez que la consulta es exitosa, imprimimos en una tabla los valores.
4. Creamos un \$_SESSION con el código.

Respuesta a situaciones anormales: Se lanzará un mensaje de error en caso de que la cola no se encuentre en el servidor

Parámetros: Código

Generación de salidas: NA

Relaciones entre entradas y salidas: El código recibido por un GET se envía en un \$_SESSION.

Especificación de los requisitos lógicos: NA. No almacenamos información en la base de datos

4.2.3.1 Requisito funcional 3.1: Unirse a cola

Número de requisito	RF-3.1
Nombre de requisito	Unirse a cola
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
<p>Descripción del requisito: El usuario podrá unirse a una cola para revisión, seleccionado en la pantalla de usuario no registrado, el submenú unirse a cola. Esta página contiene un formulario de inscripción, este formulario está compuesto por: nombre, primer apellido, segundo apellido, dni, correo electrónico y código revisión.</p> <p>Comprobación de validez de las entradas: Se comprobará que todos los campos estén rellenos, que el email sea un email correcto y el dni sea un dni válido.</p> <p>Secuencia exacta de operaciones:</p> <ol style="list-style-type: none"> 1. Rellenamos los campos del formulario y le damos a registrar. 2. Una vez que le hemos dado a registrar, validamos los campos de registro(RF-7.1). 3. Una vez que comprobado que los datos son correctos, se procederá a realizar el registro por post. 4. Hacemos la consulta a la base de datos con un select y como parámetro el código de la revisión, esto se hace para saber el número de alumnos que hay en esa cola de revisión y así poder asignarle una posición en la cola 5. Comprobamos que el usuario no esté ya en esta revisión, si no está generamos el código de usuario(RF-3.1.1) y hacemos un insert con los datos del usuario. <p>Respuesta a situaciones anormales: Se lanzará un mensaje de error en caso de el usuario ya este registrado en la cola.</p> <p>Parámetros: Código.</p> <p>Generación de salidas: Se generará una pantalla de inscripción en la cola. Se generará una pantalla de error en el caso de que el usuario ya esté inscrito en esa cola. Se generará una pantalla que nos señala el código de usuario y que ha sido inscrito correctamente</p> <p>Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida</p> <p>Especificación de los requisitos lógicos: Se almacenará como clave primaria el código de usuario, el código de revisión estará asociado a una cola y no será un valor único. El resto de campos se pueden repetir.</p>	

4.2.3.1.1 Requisito funcional 3.1.1: Generar código usuario

Número de requisito	RF-3.1.1
Nombre de requisito	Generar código de usuario
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
<p>Descripción del requisito: Generación del código personal del usuario no registrado. Generará un código aleatorio</p> <p>Comprobación de validez de las entradas: NA. Ya están comprobadas las entradas.</p> <p>Secuencia exacta de operaciones:</p> <ol style="list-style-type: none"> 1. Introducir dni y el nombre en una variable 2. Mediante el comando substr cogemos las 3 primeras letras del nombre del usuario, los 4 últimos dígitos del dni y las primeras letras del código d revisión. 3. Lo guardamos en una variable <p>Respuesta a situaciones anormales: NA.</p> <p>Parámetros: Nombre, dni</p> <p>Generación de salidas: NA. No hay relación entre entrada y salida</p> <p>Relaciones entre entradas y salidas: Código compuesto por un conjunto de letras obtenidas de forma aleatoria, de entre el nombre y el email.</p> <p>Especificación de los requisitos lógicos: NA. No se accede a la base de datos</p>	

4.2.3.2 Requisito funcional 3.2: Consultar posición en la cola

Número de requisito	RF-3.2
Nombre de requisito	Consultar información de la cola
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
<p>Descripción del requisito: Cargará un formulario para que introduzcas el email y el código de usuario, si los datos son correctos mostrará en la página dos tablas, una tabla donde mostrará la información del usuario y cuantas posiciones le queda para ser atendido. La otra tabla mostrará la información de la cola a la que se ha unido.</p> <p>Comprobación de validez de las entradas: Comprobará que el email y la contraseña estén en la base de datos.</p> <p>Secuencia exacta de operaciones:</p> <ol style="list-style-type: none"> 1. Una vez introducidos los datos y dado a ver mi posición, se comprobarán que los campos no estén vacíos mediante la función loginUsuario() en javascript y en el caso de que no los estén se enviarán los datos por POST a ComprobacionUsuarios.php. 2. En ComprobacionUsuarios.php comprobamos que el usuario este registrado en la cola mediante un select y grabaremos las variables en unos \$_SESSION 3. El sistema mostrará posición relativa <p>Respuesta a situaciones anormales: Si el usuario no existiera mandaría un aviso de que el usuario no está unido a ninguna cola</p> <p>Parámetros: Email, código usuario</p>	

Generación de salidas: Generación de la información de la cola y la posición del usuario en esta en el momento que consulta dicha información.

Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida

Especificación de los requisitos lógicos: NA. No se graba nada en la base de datos

4.2.3.2.1 Requisito funcional 3.2.1: Darse de baja de la cola

Número de requisito	RF-3.2.1
Nombre de requisito	Darse de baja de la cola
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: El usuario se puede dar de baja en una cola, para ello una vez que ha accedido a la posición en la cola, podrá mediante una opción del submenú borrarse de dicha cola. Para ello cuando le dé a borrarse de una, aparecerá un formulario con los datos del usuario y el código de la cola	
Comprobación de validez de las entradas: Comprobará que el código y el email sean correctos.	
Secuencia exacta de operaciones: <ol style="list-style-type: none"> 1. Introducirá su código de revisión y su email en ver mi posición. 2. El usuario selecciona borrarse. 3. Se redireccionado a una nueva página, donde los datos serán cargados en un formulario. 4. Una vez que se le dé a borrar, se le solicitará confirmación del borrado. 5. Una vez confirmado el borrado de la información, se enviará los datos mediante un \$_SESSION a BorrarAlumno.php 6. Se borrará de la base de datos mediante un delete usando como parámetro de búsqueda en la base de datos el código del alumno. 	
Respuesta a situaciones anormales: Si el código o el email no son correctos, se mandará un mensaje diciendo que el email o el código no existe	
Parámetros: Código y email	
Generación de salidas: El usuario ya no formará parte de la cola y se le mostrará un mensaje de que se ha dado de baja de esta	
Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida	
Especificación de los requisitos lógicos: NA. No se graba nada en la base de datos.	

4.2.4 Requisito funcional 4: Pantalla del Administrador

4.2.4.1 Requisito funcional 4.1: Gestión de usuarios

4.2.4.1.1 Requisito funcional 4.1.1: Dar de alta usuario

Número de requisito	RF-4.1.1
Nombre de requisito	Dar de alta usuario
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: Colgando de la pantalla de gestión de usuarios hay un	

submenú de dar alta usuario. La pantalla de dar alta usuario se compone de un formulario, con los campos nombre, primer apellido, segundo apellido, dni, correo electrónico y tipo de cuenta

Comprobación de validez de las entradas: Se comprobará que el DNI y el email se ha correcto. También se comprueba que los campos no se queden vacíos.

Secuencia exacta de operaciones:

1. Rellenamos los campos del formulario y le damos a registrar. Una vez que le demos a registrar validaremos los campos del registro(RF-7.1)
2. Si los datos se han validado correctamente, se enviará la información mediante un post.
3. Se guardarán los datos en la base de datos mediante un insert.

Respuesta a situaciones anormales: Si no se ha podido añadir al usuario correctamente el sistema mostrará un mensaje indicándolo y volverá a solicitar los datos del nuevo usuario. En el caso de que el usuario exista, se enviará un aviso de que el usuario ya existe.

Parámetros: Nombre, primer apellido, segundo apellido, dni, correo electrónico y tipo de cuenta

Generación de salidas: Si ha sido correcto el sistema mandará un aviso de que el usuario se ha creado correctamente

Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida

Especificación de los requisitos lógicos: correo electrónico, nombre, apellidos, tipo de cuenta y dni almacenados como varchar(30) en la base de datos

4.2.4.1.2 Requisito funcional 4.1.2: Borrar usuario

Número de requisito	RF-4.1.2
Nombre de requisito	Borrar usuario
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: Colgando de la pantalla de gestión de usuarios hay un submenú de borrar datos del usuario. La pantalla de borrar datos de un usuario se compone de un formulario donde se cargan los datos del usuario.	
Comprobación de validez de las entradas: NA. Como el profesor ya está en la base de datos no hace falta comprobar nada.	
Secuencia exacta de operaciones:	
<ol style="list-style-type: none"> 1. En la pantalla de gestión de usuarios hay una tabla con todos los profesores y administradores, seleccionamos el usuario que deseamos borrar. 2. Una vez seleccionado el usuario, mediante la función obtenerID() obtenemos el email del usuario. 3. Cuando le damos al botón borrar, mediante la función cogerDatos(), enviamos por url el email. 4. Los datos se cargan en un formulario, cuando pinchamos en borrar se nos pedirá que confirmemos que deseamos borrar ese usuario(RF-8.3). 5. Confirmado el borrado de los datos, se envía la información mediante post a BorrarRegistroUsuario.php 6. Se busca por el email al usuario y se borra de la base de datos mediante un delete. 	
Respuesta a situaciones anormales: Se informará con un mensaje de error en caso de	

que el usuario no se pueda borrar.

Parámetros: Email

Generación de salidas: El sistema generará una ventana de aviso de borrado de dicho usuario. En el caso de que no se pueda borrar se generará un aviso del error.

Relaciones entre entradas y salidas: NA. No hay relación entre la entrada y la salida

Especificación de los requisitos lógicos: NA. No hay datos que almacenar en la base de datos

4.2.4.1.3 Requisito funcional 4.1.3: Modificar usuario

Número de requisito	RF-4.1.3
Nombre de requisito	Modificar usuario
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Eencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
<p>Descripción del requisito: Colgando de la pantalla de gestión de usuarios hay un submenú de modificar datos del usuario. La pantalla de modificar datos de un usuario se compone de un formulario donde se cargan los datos del usuario.</p> <p>Comprobación de validez de las entradas: NA. Se comprobará que los campos no estén vacíos</p> <p>Secuencia exacta de operaciones:</p> <ol style="list-style-type: none"> 1. En la pantalla de gestión de usuarios hay una tabla con todos los profesores y administradores, seleccionamos el usuario que deseamos modificar. 2. Una vez seleccionado el usuario, mediante la función obtenerID() obtenemos el email del usuario. 3. Cuando le damos al botón modificar, mediante la función cogerDatos(), enviamos por url el email. 4. Los datos se cargan en un formulario, en este formulario podremos modificar el nombre, los apellidos y el password. Se nos pedirá que confirmemos que deseamos modificar ese usuario 5. Confirmado el modificado de los datos, se envía la información mediante post 6. Se busca por el email al usuario y se modifican los datos en la base de datos mediante un update. <p>Respuesta a situaciones anormales: Se informará con un mensaje de error en caso de que el usuario no se pueda modificar.</p> <p>Parámetros: Nombre, apellidos y password</p> <p>Generación de salidas: El sistema generará una ventana de aviso de modificado de dicho usuario. En el caso de que no se pueda modificar el usuario se generará un aviso de error.</p> <p>Relaciones entre entradas y salidas: Si se introduce una contraseña, se le hará un md5</p> <p>Especificación de los requisitos lógicos: nombre, apellidos y password almacenados como varchar(30) en la base de datos.</p>	

4.2.4.2 Requisito funcional 4.2: Gestión de colas

4.2.4.2.1 Requisito funcional 4.2.1: Crear cola

Número de requisito	RF-4.2.1
Nombre de requisito	Crear cola
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
<p>Descripción del requisito: Colgando de la pantalla de gestión de colas hay un submenú de crear cola. La pantalla de crear cola se compone de un formulario, con los campos asignatura, lugar de revisión, fecha de revisión, hora de revisión y profesor</p> <p>Comprobación de validez de las entradas: Se comprobará que asignatura, lugar de revisión, fecha de revisión, hora de revisión y profesor no estén vacíos. Además, se comprobará que la fecha no sea una fecha pasada.</p> <p>Secuencia exacta de operaciones:</p> <ol style="list-style-type: none"> 1. Rellenamos los campos del formulario y le damos a registrar. Una vez que le demos a registrar validaremos los campos del registro(RF-7.1) 2. Si los datos se han validado correctamente, se enviará la información mediante un post. 3. Se generará el código (RF-4.2.1.1) 4. Se guardarán los datos en la base de datos mediante un insert. <p>Respuesta a situaciones anormales: Si no se ha podido añadir la cola correctamente el sistema mostrará un mensaje indicándolo y volverá a solicitar los datos. En el caso de que la cola ya exista, se enviará un aviso de que el usuario ya existe.</p> <p>Parámetros: Asignatura, lugar de revisión, fecha de revisión, hora de revisión y profesor</p> <p>Generación de salidas: Si ha sido correcto el sistema mandará un aviso de que la cola se ha creado correctamente</p> <p>Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida</p> <p>Especificación de los requisitos lógicos: asignatura, lugar de revisión, fecha de revisión, hora de revisión y profesor serán almacenados como varchar(30) en la base de datos</p>	

4.2.4.2.1.1 Requisito funcional 4.2.1.1: Generar código cola

Número de requisito	RF-4.2.1.1
Nombre de requisito	Generar código cola
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
<p>Descripción del requisito: Generación del código personal del usuario no registrado. Generará un código aleatorio</p> <p>Comprobación de validez de las entradas: NA. Ya están comprobadas las entradas.</p> <p>Secuencia exacta de operaciones:</p> <ol style="list-style-type: none"> 1. Introducir año y el nombre en una variable 2. Mediante el comando substr cogemos las 3 primeras letras del nombre del usuario, los 2 últimos dígitos del año, un guion y un numero aleatorio 3. Lo guardamos en una variable <p>Respuesta a situaciones anormales: NA.</p> <p>Parámetros: Nombre, año</p>	

Generación de salidas: NA. No hay relación entre entrada y salida

Relaciones entre entradas y salidas: Código compuesto por un conjunto de letras obtenidas de forma aleatoria, de entre el nombre y el email.

Especificación de los requisitos lógicos: NA. No se accede a la base de datos

4.2.4.2.1.2 Requisito funcional 4.2.1.2: Profesor asociado a la cola

Número de requisito	RF-4.2.1.2
Nombre de requisito	Profesor asociado a la cola
Tipo	<input type="checkbox"/> Requisito <input checked="" type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: El administrador cuando creé la cola deberá asignar un profesor a dicha cola.	
Comprobación de validez de las entradas: NA. Como los profesores ya están en la base de datos no hace falta comprobar nada.	
Secuencia exacta de operaciones: Mediante una búsqueda en la base de datos, sacaremos en un menú desplegable la lista de todos los profesores registrados en la base de datos. Para ello hacemos un select y como condición de búsqueda, ponemos que el rol sea profesor.	
Respuesta a situaciones anormales: NA. No se contempla situaciones anormales	
Parámetros: NA. No tiene parámetros	
Generación de salidas: Genera un menú desplegable con todos los profesores registrados en la base de datos.	
Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida	
Especificación de los requisitos lógicos: NA. No se accede a la base de datos	

4.2.4.2.2 Requisito funcional 4.2.2: Borrar cola

Número de requisito	RF-4.2.2
Nombre de requisito	Borrar cola
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: Colgando de la pantalla de gestión de colas hay un submenú de borrar cola. La pantalla de borrar cola se compone de un formulario donde se cargan los datos del usuario.	
Comprobación de validez de las entradas: NA. Como la cola ya está en la base de datos no hace falta comprobar nada.	
Secuencia exacta de operaciones: <ol style="list-style-type: none"> 1. En la pantalla de gestión de colas hay una tabla con todas las colas creadas, seleccionamos la cola que deseamos borrar. 2. Una vez seleccionada la cola, mediante la función obtenerID() obtenemos el código de la revisión. 	

3. Cuando le damos al botón borrar, mediante la función `cogerDatos()`, enviamos por url el código de la revisión.
4. Los datos se cargan en un formulario, cuando pinchamos en borrar se nos pedirá que confirmemos que deseamos borrar esa cola(RF-8.3).
5. Confirmado el borrado de los datos, se envía la información mediante post a `BorrarRegistroCola.php`
6. Se busca por el código de la revisión y se borra de la base de datos mediante un delete.

Respuesta a situaciones anormales: Se informará con un mensaje de error en caso de que la cola no se pueda borrar.

Parámetros: código de la revisión

Generación de salidas: El sistema generará una ventana de aviso de borrado de dicha cola. En el caso de que no se pueda borrar se generará un aviso de error.

Relaciones entre entradas y salidas: NA. No hay relación entre la entrada y la salida

Especificación de los requisitos lógicos: NA. No hay datos que almacenar en la base de datos

4.2.4.2.3 Requisito funcional 4.2.3: Modificar cola

Número de requisito	RF-4.2.3
Nombre de requisito	Modificar cualquier cola
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: Colgando de la pantalla de gestión de colas hay un submenú de modificar cola. La pantalla de modificar colas se compone de un formulario donde se cargan los datos de las colas.	
Comprobación de validez de las entradas: NA. Se comprobará que los campos no estén vacíos	
Secuencia exacta de operaciones: <ol style="list-style-type: none"> 1. En la pantalla de gestión de colas hay una tabla con todas las colas creadas, seleccionamos la cola que deseamos borrar. 2. Una vez seleccionada la cola, mediante la función <code>obtenerID()</code> obtenemos el código de la revisión. 3. Cuando le damos al botón modificar, mediante la función <code>cogerDatos()</code>, enviamos por url el código de la revisión. 4. Los datos se cargan en un formulario, en este formulario podremos modificar el lugar de revisión, la fecha de revisión, la hora de revisión y el profesor. Se nos pedirá que confirmemos que deseamos modificar ese usuario 5. Confirmado el modificado de los datos, se envía la información mediante post 6. Se busca por el código de la revisión y se modifican los datos en la base de datos mediante un update. 	
Respuesta a situaciones anormales: Se informará con un mensaje de error en caso de que la cola no se pueda modificar.	
Parámetros: Asignatura, lugar de revisión, fecha de revisión, hora de revisión y profesor	
Generación de salidas: El sistema generará una ventana de aviso de modificado de	

dicho usuario. En el caso de que no se pueda modificar el usuario se generará un aviso de error.

Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida

Especificación de los requisitos lógicos: Asignatura, lugar de revisión, fecha de revisión, hora de revisión y profesor serán almacenados como varchar(30) en la base de datos.

4.2.4.2.3.1 Requisito funcional 4.2.3.1: Modificar profesor asociado

Número de requisito	RF-4.2.3.1
Nombre de requisito	Modificar profesor asociado
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: El administrador podrá modificar el profesor asociado a una cola.	
Comprobación de validez de las entradas: NA. Como los profesores ya están en la base de datos no hace falta comprobar nada.	
Secuencia exacta de operaciones: Mediante una búsqueda en la base de datos, sacaremos en un menú desplegable la lista de todos los profesores registrados en la base de datos. Para ello hacemos un select y como condición de búsqueda, ponemos que el rol sea profesor.	
Respuesta a situaciones anormales: NA. No se contempla situaciones anormales	
Parámetros: NA. No tiene parámetros	
Generación de salidas: Genera un menú desplegable con todos los profesores registrados en la base de datos.	
Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida	
Especificación de los requisitos lógicos: NA. No hay datos que almacenar en la base de datos	

4.2.5 Requisito funcional 5: Pantalla del profesor

4.2.5.1 Requisito funcional 5.1: Gestión de usuarios

4.2.5.1.1 Requisito funcional 5.1.1: Modificar mi usuario

Número de requisito	RF-5.1.1
Nombre de requisito	Modificar mi usuario
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: Colgando de la pantalla de gestión de usuarios hay un submenú de modificar datos del usuario. La pantalla de modificar datos de un usuario se compone de un formulario donde se cargan los datos del usuario.	
Comprobación de validez de las entradas: NA. Se comprobará que los campos no	

estén vacíos

Secuencia exacta de operaciones:

1. En la pantalla de gestión de usuarios hay una tabla con los datos del profesor.
2. Una vez seleccionado el profesor, mediante la función obtenerID() obtenemos el email del profesor.
3. Cuando le damos al botón modificar, mediante la función cogerDatos(), enviamos por url el email.
4. Los datos se cargan en un formulario, en este formulario podremos modificar el nombre, los apellidos y el password. Se nos pedirá que confirmemos que deseamos modificar los datos.
5. Confirmado el modificado de los datos, se envía la información mediante post
6. Se busca por el email al usuario y se modifican los datos en la base de datos mediante un update.

Respuesta a situaciones anormales: Se informará con un mensaje de error en caso de que el usuario no se pueda modificar.

Parámetros: Nombre, apellidos y password

Generación de salidas: El sistema generará una ventana de aviso de modificado de dicho usuario. En el caso de que no se pueda modificar el usuario se generará un aviso de error.

Relaciones entre entradas y salidas: Si se introduce una contraseña, se le hará un md5

Especificación de los requisitos lógicos: nombre, apellidos y password almacenados como varchar(30) en la base de datos.

4.2.5.2 Requisito funcional 5.2: Gestión de colas

4.2.5.2.1 Requisito funcional 5.2.1: Crear cola propia

Número de requisito	RF-5.2.1
Nombre de requisito	Crear cualquier cola
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción del requisito: Colgando de la pantalla de gestión de colas hay un submenú de crear cola. La pantalla de crear cola se compone de un formulario, con los campos asignatura, lugar de revisión, fecha de revisión, hora de revisión y profesor	
Comprobación de validez de las entradas: Se comprobará que asignatura, lugar de revisión, fecha de revisión, hora de revisión. Además, se comprobará que la fecha no sea una fecha pasada.	
Secuencia exacta de operaciones:	
<ol style="list-style-type: none"> 1. Rellenamos los campos del formulario y le damos a registrar. Una vez que le demos a registrar validaremos los campos del registro(RF-7.1) 2. Si los datos se han validado correctamente, se enviará la información mediante un post. 3. Se generará el código (RF-4.2.1.1) 4. Se guardarán los datos en la base de datos mediante un insert. 	
Respuesta a situaciones anormales: Si no se ha podido añadir la cola correctamente el sistema mostrará un mensaje indicándolo y volverá a solicitar los datos. En el caso de que la cola ya exista, se enviará un aviso de que el usuario ya existe.	

Parámetros: Asignatura, lugar de revisión, fecha de revisión, hora de revisión y profesor
Generación de salidas: Si ha sido correcto el sistema mandará un aviso de que la cola se ha creado correctamente
Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida
Especificación de los requisitos lógicos: asignatura, lugar de revisión, fecha de revisión, hora de revisión y profesor serán almacenados como varchar(30) en la base de datos

4.2.5.2.2 Requisito funcional 5.2.2: Modificar mis colas

Número de requisito	RF-5.2.2
Nombre de requisito	Modificar cualquier cola
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: Colgando de la pantalla de gestión de colas hay un submenú de modificar cola. La pantalla de modificar colas se compone de un formulario donde se cargan los datos de las colas.	
Comprobación de validez de las entradas: NA. Se comprobará que los campos no estén vacíos	
Secuencia exacta de operaciones: <ol style="list-style-type: none">1. En la pantalla de gestión de colas hay una tabla con todas las colas creadas, seleccionamos la cola que deseamos borrar.2. Una vez seleccionada la cola, mediante la función obtenerID() obtenemos el código de la revisión.3. Cuando le damos al botón modificar, mediante la función cogerDatos(), enviamos por url el código de la revisión.4. Los datos se cargan en un formulario, en este formulario podremos modificar el lugar de revisión, la fecha de revisión, la hora de revisión y el profesor. Se nos pedirá que confirmemos que deseamos modificar ese usuario5. Confirmado el modificado de los datos, se envía la información mediante post6. Se busca por el código de la revisión y se modifican los datos en la base de datos mediante un update.	
Respuesta a situaciones anormales: Se informará con un mensaje de error en caso de que la cola no se pueda modificar.	
Parámetros: Asignatura, lugar de revisión, fecha de revisión, hora de revisión y profesor	
Generación de salidas: El sistema generará una ventana de aviso de modificado de dicho usuario. En el caso de que no se pueda modificar el usuario se generará un aviso de error.	
Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida	
Especificación de los requisitos lógicos: Asignatura, lugar de revisión, fecha de revisión, hora de revisión y profesor serán almacenados como varchar(30) en la base de datos.	

4.2.5.2.3 Requisito funcional 5.2.3: Gestión turnos

Número de requisito	RF-6.4
----------------------------	--------

Nombre de requisito	Siguiente turno
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: Colgando de la pantalla de gestión de colas hay un submenú de gestión turnos. La pantalla de gestión turnos se compone de dos tablas, una tabla de la gente que están siendo atendida y otra tabla de la gente que están en espera. Para poder pasar turno o poner a la gente en espera disponemos de dos botones	
Comprobación de validez de las entradas: NA. Como los alumnos están en la base de datos, no hace falta comprobar nada.	
Secuencia exacta de operaciones: <ol style="list-style-type: none"> Desde gestión de colas activamos la cola, cuando le demos a activar nos redireccionará a la página GestionTurnos.php En la tabla de atendidos mostrará a la gente según la posición de inscripción en la revisión. Para ello haremos una consulta a la base de datos, usando como parámetros el código de revisión y la posición actual. Cada vez que le des a pasar turno, el iterador posición aumentará en una unidad y el estado del alumno pasará a atendido, para ello haremos un update a la base de datos. Si seleccionas poner en espera, el alumno se introducirá en un array de alumnos en espera, el iterador posición se le aumentará en una unidad y el estado del alumno pasará a esperando. En el caso de que tengamos gente en espera, cuando pasemos turno se volverá a llamar a los usuarios que estén en espera para ello haremos un update a la base de datos. Cuando un alumno en espera sea atendido, su estado pasará a atendido. Cuando no queden más alumnos por atender, se borrará la cola y todos los datos asociados (RF-7.2). 	
Respuesta a situaciones anormales: Si no hay más alumnos no se muestra nada en la tabla.	
Parámetros: Código revisión, código alumno.	
Generación de salidas: Mostrará por pantalla el siguiente alumno en ser atendido. Mostrará los alumnos en espera.	
Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida	
Especificación de los requisitos lógicos: el estado será almacenado como varchar(30) en la base de datos	

4.2.6 Requisito funcional 6: Módulo de visualización

4.2.6.1 Requisito funcional 6.1: Lista de últimos llamados

Número de requisito	RF-6.1
Nombre de requisito	Lista de últimos llamados
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: El módulo de visualización estará compuesto por dos tablas, una para mostrar a la gente que están siendo llamada y otra tabla con los avisos generales.	

Comprobación de validez de las entradas: NA. Como los alumnos están en la base de datos, no hace falta comprobar nada.

Secuencia exacta de operaciones: Muestra por pantalla una lista con los usuarios que han sido llamados, por cada usuario se mostrará:

1. Código usuario
2. Lugar de revisión
3. Asignatura
4. Estado

Para ello haremos una consulta a la base de datos mediante el comando select, en la cual como parámetros de búsqueda podremos que el estado sea no atendido o esperando y además hemos puesto un límite de dos personas. La tabla se refrescará cada 10 segundos más o menos.

Respuesta a situaciones anormales: NA. No se contempla situaciones anormales

Parámetros: Código de revisión

Generación de salidas: NA. No genera salida

Relaciones entre entradas y salidas: Genera una pantalla con esta información.

Especificación de los requisitos lógicos: NA. No hay datos que almacenar en la base de datos

4.2.6.2 Requisito funcional 6.2: Lista de avisos

Número de requisito	RF-6.2
Nombre de requisito	Lista de avisos
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: El módulo de visualización estará compuesto por dos tablas, una para mostrar a la gente que están siendo llamada y otra tabla con los avisos generales.	
Comprobación de validez de las entradas: NA. Como los avisos están en la base de datos, no hace falta comprobar nada.	
Secuencia exacta de operaciones: Muestra por pantalla una lista con todos los avisos generados por los usuarios: <ol style="list-style-type: none"> 1. Profesor 2. Mensaje Para ello haremos una consulta a la base de datos mediante el comando select. La tabla se refrescará cada 10 segundos más o menos.	
Respuesta a situaciones anormales: NA. No se contempla situaciones anormales	
Parámetros: NA. No es necesario ya que se cogen todos los mensajes.	
Generación de salidas: NA. No genera salida	
Relaciones entre entradas y salidas: Genera una pantalla con esta información.	
Especificación de los requisitos lógicos: NA. No hay datos que almacenar en la base de datos	

4.2.7 Requisito funcional 7: Requisitos generales

4.2.7.1 Requisito funcional 7.1: Validar campos de registro

Número de requisito	RF-7.1
Nombre de requisito	Validar campos de registro
Tipo	<input type="checkbox"/> Requisito <input checked="" type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
<p>Descripción del requisito: Comprobación de los campos de los formularios</p> <p>Comprobación de validez de las entradas: El sistema comprobará que el DNI introducido por el usuario tiene 8 dígitos, un carácter alfabético y sea válido. Se comprobará que el email sea correcto y que los demás campos no estén vacíos</p> <p>Secuencia exacta de operaciones:</p> <ol style="list-style-type: none"> 1. Se utiliza un onsubmit que devuelve el resultado de la función validarDatosRellenos(). 2. Esta función comprobará que los campos del formulario no estén vacíos. Además, comprobará que el dni sea válido, para ello llamará a la función validarDNI(), donde comprobará que tenga el tamaño adecuado, además comprobará que, si es un dni para nacionales, la letra coincida con el valor de los números. En el caso de que sea un dni para residentes extranjeros, se comprobará que tenga la estructura adecuada y que la letra final coincida con el valor de los números del dni. 3. En el caso de que todo sea correcto devuelve true. <p>Respuesta a situaciones anormales: Si no se trata de un DNI válido se mostrará un mensaje al usuario indicándoselo</p> <p>Parámetros: NA. No hay parámetros</p> <p>Generación de salidas: NA. No genera salida</p> <p>Relaciones entre entradas y salidas: Según los datos sean correctos devuelve true en caso contrario devuelve false</p> <p>Especificación de los requisitos lógicos: NA. No hay datos que almacenar en la base de datos.</p>	

4.2.7.2 Requisito funcional 7.2: Eliminar información de la cola

Número de requisito	RF-7.2
Nombre de requisito	Eliminar información de la cola
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
<p>Descripción del requisito: El sistema borrará toda la información asociada a la cola.</p> <p>Comprobación de validez de las entradas: NA. Como las colas están en la base de datos, no hace falta comprobar nada.</p> <p>Secuencia exacta de operaciones: La información asociada a la cola será eliminada de forma automática. Se buscará en la base de datos el código de la revisión que se desea borrar. Una vez encontrada se procederá a su borrado mediante un delete</p> <p>Respuesta a situaciones anormales: Si la información no se puede borrar mandará un aviso</p> <p>Parámetros: Código revisión</p>	

Generación de salidas: Genera un aviso de que la revisión se ha borrado correctamente.

Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida

Especificación de los requisitos lógicos: NA. No hay datos que almacenar en la base de datos.

4.2.7.3 Requisito funcional 7.3: Confirmación de borrado

Número de requisito	RF-7.3
Nombre de requisito	Confirmación de borrado
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Desarrollador
Prioridad del requisito	<input type="checkbox"/> Alta/Esencial <input checked="" type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito:	El sistema mostrará un aviso, pidiendo confirmar el borrado de la cola
Comprobación de validez de las entradas:	NA, no hay entrada
Secuencia exacta de operaciones:	<ol style="list-style-type: none">1. El sistema lanzará una ventana de aviso de borrado2. El usuario aceptará o rechazará borrar la cola3. En caso de aceptar el sistema eliminará la información de la cola (RF-7.2)4. En caso de rechazar se volverá a la página anterior
Respuesta a situaciones anormales:	NA. No se contempla situaciones anormales
Parámetros:	Código revisión
Generación de salidas:	El sistema lanzará un aviso de que el borrado se ha realizado con éxito
Relaciones entre entradas y salidas:	NA. No hay relación entre entrada y salida
Especificación de los requisitos lógicos:	NA. No hay datos que almacenar en la base de datos.

4.2.7.4 Requisito funcional 7.4: Comprobar fecha y hora

Número de requisito	RF-7.4
Nombre de requisito	Modificar horario
Tipo	<input type="checkbox"/> Requisito <input checked="" type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito:	Cuando se vaya a alterar la fecha o la hora o cuando se vaya a añadir una fecha u hora, se procederá a comprobar que la fecha sea actual.
Comprobación de validez de las entradas:	NA. El formato es estricto, por lo cual el usuario deberá introducir ese tipo de formato de fecha y hora
Secuencia exacta de operaciones:	<ol style="list-style-type: none">1. Se utiliza un onsubmit que devuelve el resultado de la función comprobarFecha ().2. Se le pasará como parámetros una fecha y una hora.3. Se comprobará que el año sea el mismo o superior.4. Se comprobará que el mes, si estamos en el mismo año, sea mayor o igual al actual, en el caso de que sean años diferentes no hará falta que se compruebe.5. Se comprobará que el día sea igual o mayor al actual si el mes es el mismo y el

- año es el mismo. Sino no hará falta
6. Se devolverá true o false, dependiendo de la fecha introducida.

Respuesta a situaciones anormales: NA. No se contempla situaciones anormales

Parámetros: Fecha, hora

Generación de salidas: NA. No genera salida

Relaciones entre entradas y salidas: NA. No hay relación entre entrada y salida

Especificación de los requisitos lógicos: NA. No hay datos que almacenar en la base de datos.

4.2.7.5 Requisito funcional 7.5: Crear aviso

Número de requisito	RF-7.5
Nombre de requisito	Crear aviso
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
<p>Descripción del requisito: Creación de un aviso por parte de un administrador o un profesor. La pantalla de crear aviso está compuesta por un formulario con los campos nombre, fecha de expiración, hora de expiración y un campo de texto para escribir el aviso</p> <p>Comprobación de validez de las entradas: Se comprueba que la fecha y la hora sean actuales.</p> <p>Secuencia exacta de operaciones:</p> <ol style="list-style-type: none"> 1. El profesor o administrador seleccionarán crear aviso 2. Si el usuario es el administrador podrá elegir de la base de datos el nombre del usuario que quiere enviar el mensaje, para ello hacemos una consulta de los usuarios. Si el usuario es el profesor en nombre aparecerá su nombre, para ello cogemos el nombre de la sesión. 3. Rellenamos la fecha y la hora de expiración, se comprueba que la fecha sea actual (RF-7.4) 4. Se rellena el aviso y le damos a crear aviso, enviamos los datos por post y almacenamos el aviso con un insert. <p>Respuesta a situaciones anormales: NA. No se contempla situaciones anormales</p> <p>Parámetros: Texto del aviso, profesor, fecha y hora.</p> <p>Generación de salidas: Se enviará el aviso que se mostrará en la pantalla de la sala de espera.</p> <p>Relaciones entre entradas y salidas: Se enviará el aviso y se mostrará en la pantalla de la sala de espera.</p> <p>Especificación de los requisitos lógicos: El profesor, fecha y hora será almacenado como varchar(30) en la base de datos. El texto de aviso se almacenará en un varchar(250).</p>	

4.2.7.6 Requisito funcional 7.6: Conectarse base de datos

Número de requisito	RF-7.6
----------------------------	--------

Nombre de requisito	Conectarse base de datos
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: Usaremos una función php para conectarnos a la base de datos.	
Comprobación de validez de las entradas: NA	
Secuencia exacta de operaciones:	
<ol style="list-style-type: none"> 1. Grabamos en unas variables, el nombre del servidor, el usuario de la base de datos, la contraseña y el nombre de la base de datos. 2. Con un mysqli_connect(\$servername, \$username, \$password, \$dbname), nos conectamos a la base de datos. 	
Respuesta a situaciones anormales: Si no se puede conectar a la base de datos, enviara un mensaje de error.	
Parámetros: nombre del servidor, usuario, contraseña y nombre de la base de datos	
Generación de salidas: NA. No genera salida	
Relaciones entre entradas y salidas: Genera una conexión con la base de datos	
Especificación de los requisitos lógicos: NA. No hay datos que almacenar en la base de datos.	

4.2.7.7 Requisito funcional 7.7: Cerrar sesión

Número de requisito	RF-7.7
Nombre de requisito	Cerrar sesión
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente del requisito	Cliente
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/ Opcional
Descripción del requisito: El usuario registrado siempre podrá cerrar la sesión desde cualquier página, para ello tanto en el menú como en la esquina superior derecha, tendrá una opción de cerrar sesión	
Comprobación de validez de las entradas: NA	
Secuencia exacta de operaciones:	
<ol style="list-style-type: none"> 1. Se hace un session_start(); 2. Se procede destruir la sesión con un session_destroy(); 	
Respuesta a situaciones anormales: NA.	
Parámetros: NA. No hay parámetros	
Generación de salidas: Genera un mensaje informando que la sesión se ha cerrado correctamente.	
Relaciones entre entradas y salidas: NA.	
Especificación de los requisitos lógicos: NA. No hay datos que almacenar en la base de datos.	

5 Conclusiones

5.1 Problemas encontrados

- Uno de los principales problemas que hemos encontrado ha sido el desarrollo de la documentación, debido a que para nosotros todo eran requisitos funcionales y nos ha costado distinguir entre lo que verdaderamente era requisitos funcionales y lo que no.
- Otro de los problemas encontrados ha sido el cómo estructurar la página web, ya que no teníamos claro cuántos usuarios iba a tener nuestro proyecto, además de cuantos sub-apartados iba a tener cada usuario.
- Un gran problema fue el cómo estructurar la base de datos, nos encontramos con el problema de las tablas, ya que no sabíamos, el conjunto necesario de tablas y atributos.
- Luego hemos tenido problemas para integrar javascript y php con html. Sacar la información de tablas y formulario ha sido bastante complicado, además no solo eso sino el pasar la información de una variable javascript a una variable php.

5.2 Conclusiones

Ha sido un proyecto excitante y complejo, ha exigido de nosotros todos nuestros conocimiento y aptitudes. Con este proyecto hemos aprendido a desenvolvemos en un proyecto real. Hemos aprendido a conectarnos a una base de datos, mediante php y como php es tan flexible que permite hacer casi cualquier cosa.

De javascript hemos aprendido que es sumamente útil para comprobar la validez de los campos y nos permite de forma sencilla obtener la información de los formularios y tablas

De css y html hemos aprendido la versatilidad para crear la página web que deseamos.

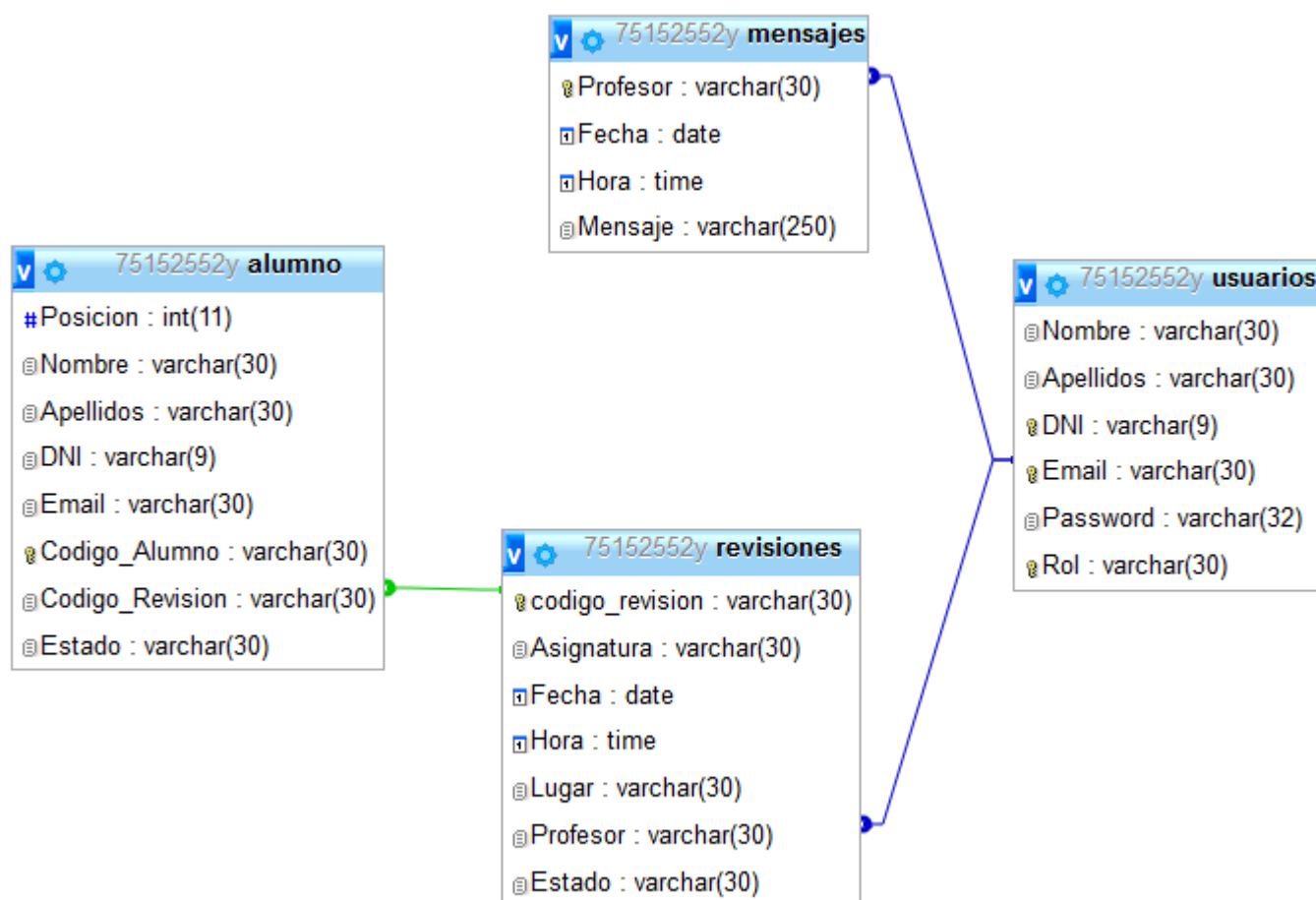
También hemos aprendido la importancia de hacer una buena documentación para que el cliente tenga toda la información necesaria del proyecto y pueda aportar sus ideas.

6 Bibliografía

[http : //www.w3schools.com/](http://www.w3schools.com/)
[http : //es.html.net/](http://es.html.net/)
<http://forums.mysql.com/>
<http://stackoverflow.com/>
<http://www.librosweb.es>
[http : //www.cssblog.es](http://www.cssblog.es)
<http://www.desarrolloweb.com/>
<http://php.net/manual>
<http://www.forosdelweb.com>
<http://www.clubdelphi.com/foros>
<http://www.desarrolloweb.com/>

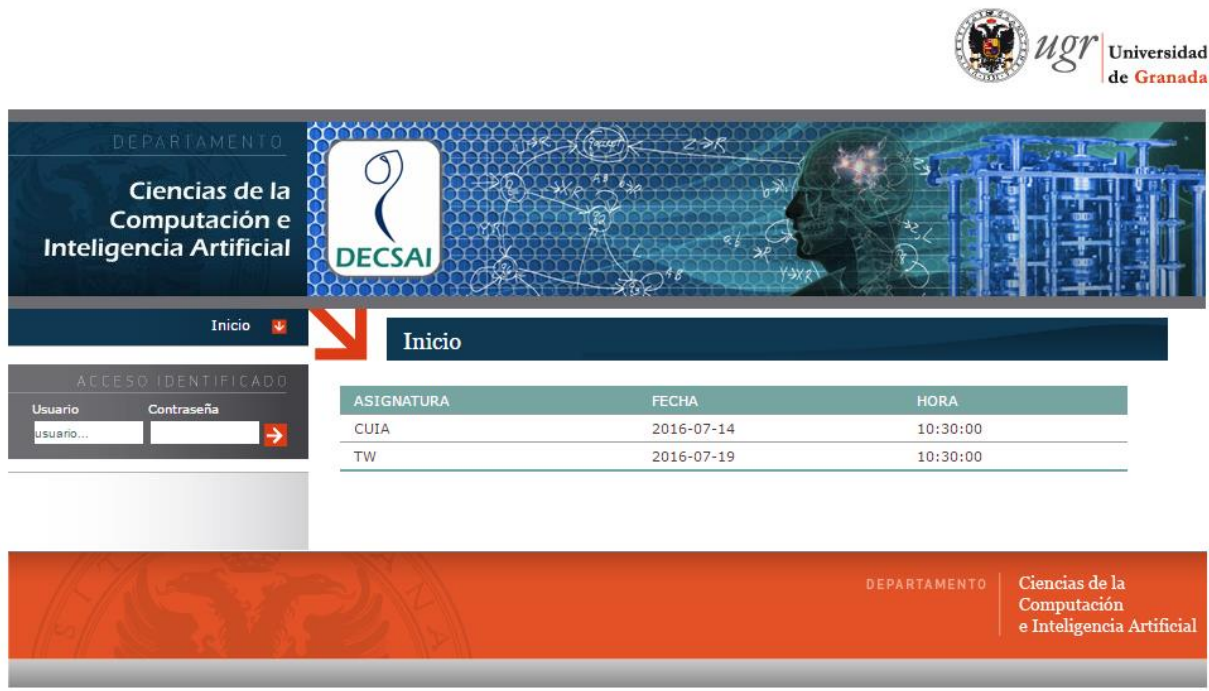
7 Apéndices

7.1 Estructura de la base de datos



7.2 Diseño interfaces de usuario

7.2.1 Página inicio



7.2.2 Página administrador



7.2.3 Página profesor



The screenshot shows the DECSAI Professor Page. At the top, there is a header with the DECSAI logo and the text "DEPARTAMENTO Ciencias de la Computación e Inteligencia Artificial". Below the header, there is a navigation menu with options: "Inicio", "Gestionar mi usuario", "Gestionar mis colas", "Crear aviso", and "Cerrar Sesión". The main content area displays the DECSAI logo and the text "Perfil de Profesor". On the right side, there is a user status bar showing "Usuario: Antonio" and "Cerrar Sesión".

7.2.4 Página usuario no registrado



The screenshot shows the DECSAI User Page. At the top, there is a header with the DECSAI logo and the text "DEPARTAMENTO Ciencias de la Computación e Inteligencia Artificial". Below the header, there is a navigation menu with options: "Mis datos en la cola", "Dar de Baja", and "Salir". The main content area displays the text "Gestión de usuarios" and a table with user information. The table has two sections: one for user details and one for exam details.

NOMBRE	APELLIDOS	DNI	CODIGO ALUMNO	EMAIL	POSICION RELATIVA
Javier	Sanchez Garcia	88888888Y	Jav888YTW1	javier@gmail.com	Quedan 3 Personas

CÓDIGO	ASIGNATURA	FECHA	HORA	LUGAR
TW16-1	TW	2016-06-15	20:30:00	Despacho 1

7.2.5 Modulo visualización.



Departamento de Ciencias de la Computación e I. A.
 Universidad de Granada



Modulo Visualización

ALUMNO	LUGAR	ASIGNATURA	ESTADO
ocohcbac	A2.5	saPksaks	No atendido

PROFESOR	MENSAJE
adwer@zvcv	Hola que tal como estamos todos, espero que bien, pablo es un despistado
totidos@hotmail.com	Esto es una prueba para ver si esto funciona bastante bien y se amplia de narices el cuadro de texto y asi poder ver todo el mensaje

DECSAI - Departamento de Ciencias de la Computación e Inteligencia Artificial - Acceso Identificado