

Aplicación web realizada con el Framework  
Symfony para gestionar las funcionalidades y  
base de datos de Adobe Connect por parte de  
usuarios de Campus do Mar

Eduardo Alonso Gil

18 de Marzo 2012

A mi familia

# Índice

<b>I</b>	<b>Introducción</b>	<b>5</b>
1.	FRAMEWORK	5
2.	SYMFONY	5
<b>II</b>	<b>Estado del Arte</b>	<b>5</b>
3.	Una aproximación al diseño y desarrollo de interfaces de usuario para aplicaciones colaborativas	6
<b>III</b>	<b>Finalidad del Proyecto</b>	<b>7</b>
4.	Especificación de la organización	7
4.1.	Definición de roles de acceso a la aplicacion eMeeting . . . . .	7
4.1.1.	Owner . . . . .	7
4.1.2.	Presenter . . . . .	9
4.1.3.	Participant . . . . .	9
5.	Definición de roles internos de eMeeting para DigiMar	9
5.1.	Administrador . . . . .	9
5.2.	Usuario VIP . . . . .	9
5.3.	Usuario estándar . . . . .	9
6.	Acceso anónimo	9
7.	Funciones incluidas en el Cron	10
7.1.	Sincronización de las grabaciones . . . . .	10
7.2.	Sincronización de los usuarios de Adobe Connect . . . . .	10
<b>IV</b>	<b>Primera Versión</b>	<b>10</b>
8.	El Middleware	11
9.	Definición de la Base de Datos de la aplicación	11
9.1.	Relaciones entre las diferentes tablas . . . . .	11
10.	Seguridad de la Aplicación	12

<b>11.Funcionalidades del interfaz de Usuario</b>	<b>12</b>
11.1. Botonera global . . . . .	12
11.1.1. + eMeetings . . . . .	12
11.1.2. Admin . . . . .	13
11.1.3. eMeeting Help . . . . .	13
11.1.4. Adobe Connect Test . . . . .	13
11.1.5. Achievement Files . . . . .	13
11.1.6. Change Password . . . . .	13
11.2. Cuadro de acceso al eMeeting . . . . .	13
11.2.1. Botonera del cuadro de acceso . . . . .	13
11.3. Funciones PHP asociadas a la botonera . . . . .	14
11.3.1. Join . . . . .	14
11.3.2. Enabled/Disabled . . . . .	14
11.3.3. Delete . . . . .	14
11.3.4. Recordings . . . . .	14
11.3.5. Change Nick Name . . . . .	15
 <b>V Segunda Versión</b>	 <b>15</b>
<b>12.Seguridad de la Aplicación</b>	<b>15</b>
<b>13.Concepto de Grupo</b>	<b>16</b>
<b>14.Página de usuario</b>	<b>17</b>
<b>15.Página Controladora</b>	<b>18</b>
<b>16.Página de Grupo</b>	<b>18</b>
<b>17.Concepto de sala</b>	<b>18</b>
<b>18.Elaboración del estilo de la aplicación a partir de plantillas     CSS3 y HTML 5</b>	<b>19</b>

## Parte I

# Introducción

### 1. FRAMEWORK

La palabra inglesa "framework" define, en términos generales, un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular, que sirve como referencia para enfrentar y resolver nuevos problemas de índole similar. En el desarrollo de software, un framework o infraestructura digital, es una estructura conceptual y tecnológica de soporte definido, normalmente con artefactos o módulos de software concretos, con base a la cual otro proyecto de software puede ser más fácilmente organizado y desarrollado. Típicamente, puede incluir soporte de programas, bibliotecas, y un lenguaje interpretado, entre otras herramientas, para así ayudar a desarrollar y unir los diferentes componentes de un proyecto. Representa una arquitectura de software que modela las relaciones generales de las entidades del dominio, y provee una estructura y una especial metodología de trabajo, la cual extiende o utiliza las aplicaciones del dominio.

En las siguientes líneas abordaremos la solución a un problema mediante un framework específico para PHP y destinado al desarrollo de aplicaciones web. Una gran lista de aplicaciones se han desarrollado bajo el framework Symfony.

### 2. SYMFONY

<http://www.symfony.es/que-es-symfony/>

Symfony es un framework PHP que facilita el desarrollo de las aplicaciones web. Symfony se encarga de todos los aspectos comunes y aburridos de las aplicaciones web, dejando que el programador se dedique a aportar valor desarrollando las características únicas de cada proyecto. Si todavía programas aplicaciones PHP a mano, estás perdiendo el tiempo y mucho dinero. Symfony aumenta exponencialmente tu productividad y te ayuda a mejorar la calidad de tus aplicaciones web aplicando todas las buenas prácticas y patrones de diseño que se han definido para la web. Otro framework para aplicaciones web es WordPress más extendido entre usuarios Windows.

## Parte II

# Estado del Arte

Qué hay antes de esta aplicación para gestionar videoconferencias. Hablamos un poco de Adobe Connect y su interfaz, y buscamos algún otro por ahí.

### 3. Una aproximación al diseño y desarrollo de interfaces de usuario para aplicaciones colaborativas

La aproximación que se propone, parte de la descripción de un sistema colaborativo que se realiza en la metodología AMENITIES. Dicha descripción consta de dos grupos de modelos:

1. Modelos de requisitos: El proceso de elicitación de requisitos se realiza principalmente con la técnica de etnografía aplicada y los diagramas de casos de uso, aunque pueden ser utilizadas otras técnicas complementarias, como escenarios y brainstorming.
  - Etnografía: Permite reunir y documentar informalmente los requisitos de la interfaz de usuario desde el punto de vista del usuario.
  - Diagramas de casos de uso: Estructuran y especifican los requisitos funcionales, roles y actores del sistema.
2. Modelo cooperativo: Es un modelo conceptual que describe y representa la estructura y funcionamiento de un sistema colaborativo. Para su construcción de forma jerárquica, en base a modelos específicos que proporcionan diferentes vistas para un mismo sistema cooperativo, se aplica un método que consta de cuatro etapas:
  - Especificación de la organización: Se definen las organizaciones que intervienen en el sistema en base a los roles de cada organización, leyes que especifican los cambios dinámicos entre roles y capacidades de los actores. También se puede especificar eventos y acciones asociadas a los roles.
  - Definición de roles: Se definen las tareas que están asociadas a cada rol, distinguiendo entre tarea individual y cooperativa. Además, para cada tarea se especifica: conjunto de tareas que pueden interrumpir su realización, eventos que la disparan, leyes que habilitan su realización por miembros del grupo, acciones en las transiciones de entrada y salida de la tarea, objetos de información requeridos y multiplicidad necesaria para desarrollar la tarea.
  - Definición de tareas: Se describen cada una de las tareas identificadas anteriormente, indicando: los roles/actores que colaboran en su realización, los objetos de información utilizados, los artefactos necesarios para realizar la tarea, los protocolos de interacción entre los participantes, número de veces que se puede repetir, restricciones temporales, su descomposición en subactividades, tareas mixtas o internas.
  - Especificación de protocolos de interacción: Se definen las características de los protocolos, las subactividades asociadas, los roles participantes y las leyes que tienen que cumplir.[5]

A partir de la información descrita en los dos modelos anteriores, se derivaran los modelos básicos sobre los que se fundamentará la construcción de la interfaz de usuario para las aplicaciones colaborativas a desarrollar. Las relaciones entre los distintos modelos facilitan el proceso de obtención del interfaz de usuario a través de un proceso de transformación de modelos. De este modo, se define un conjunto de modelos encargados de recoger las características intrínsecas de estos sistemas, es decir, tanto el aspecto del trabajo individual como en grupo. El proceso de trabajo en grupo debería ser tan natural como cuando se trabaja individualmente, aunque en la práctica los procesos de trabajo en grupo son más complejos que los individuales ya que se ha de considerar el comportamiento del grupo en base a las dinámicas presentes en las relaciones sociales, como entorno, implicación del participante, procedimiento de trabajo en grupo, conciencia de grupo y cognición distribuida.

## Parte III

# Finalidad del Proyecto

Reuniones de videoconferencia sobre ADOBE CONNECT, gestión de salas disponibles tanto para usuarios individuales como para grupos de usuarios, creados en la plataforma del Campus del Mar.

## 4. Especificación de la organización

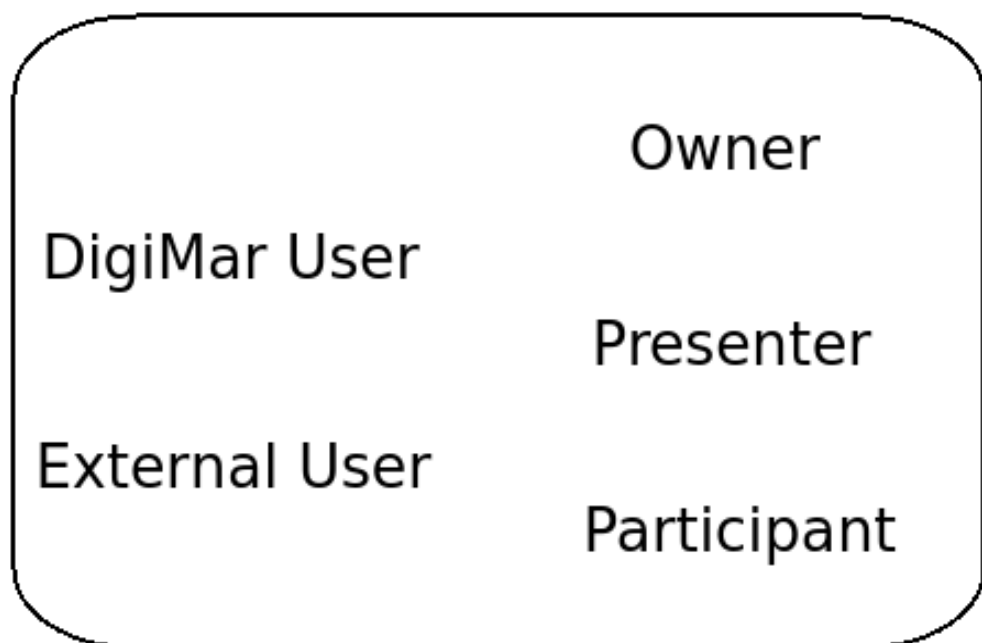
Tipo de Usuario - Role	Owner	Presenter	Participant
DigiMar Partner	X	X	X
External Partner		X	X

### 4.1. Definición de roles de acceso a la aplicación eMeeting

En esta subsección explicaremos los diferentes tipos de acceso que concede la aplicación eMeeting sobre las salas creadas en Adobe Connect.

#### 4.1.1. Owner

Este tipo de usuario sólo es posible si formamos parte de los usuarios de DigiMar. Es el creador del eMeeting. El único con el privilegio de borrar el eMeeting y poder grabar las reuniones.



Cuadro 1: Organización de la Estructura de Usuarios



#### **4.1.2. Presenter**

Este tipo de usuario es posible tanto si tienes acceso a DigiMar como si algún miembro de DigiMar te invita con la URL adecuada que describiremos más adelante. Este tipo de usuario puede interactuar en la sala pero no puede grabar reuniones.

#### **4.1.3. Participant**

Es el usuario con menos privilegios, no le está permitido grabar ni interactuar en la sala, únicamente puede ver a los diferentes usuarios que participen en el eMeeting.

## **5. Definición de roles internos de eMeeting para DigiMar**

Los roles de la aplicación serán administrador, usuario VIP y usuario estándar.

### **5.1. Administrador**

Este usuario tiene acceso a la base de datos de la aplicación y puede manipular los contenidos a su voluntad.

### **5.2. Usuario VIP**

Este usuario tiene acceso a las salas retringidas destinadas a eMeetings de alta relevancia.

### **5.3. Usuario estándar**

La mayoría de los usuarios poseeran este tipo de acceso que permite crear y borrar eMeetings dentro de la aplicación y la gestión de los mismos.

## **6. Acceso anónimo**

Cuando un usuario es invitado desde una URL generada para tal efecto, se le redirige a una pantalla de acceso, en la cual, se da la opción de ingresar en la aplicación como usuario un usuario de DigiMar o entrar como usuario anónimo.

La URL generada para este efecto tiene dos roles, presentador o participante. Si se entra como participante el código envía la opción `guest=true` lo cual hace que por defecto el panel de acceso anónimo marque la opción de entrar como

huesped. En cualquier otro caso la opción marcado por defecto es la de cubrir el formulario y entrar como usuario de DigiMar.

El código Javascript de este acceso alterna entre los dos tipos de entrada ocultado el formulario de cada entrada en función del estado del radio-button.

## **7. Funciones incluidas en el Cron**

Se han diseñado dos funciones que se ejecutan en el servidor periódicamente. Estas funciones realizan tareas de mantenimiento de la aplicación

### **7.1. Sincronización de las grabaciones**

Para poder visualizar las grabaciones en el botón Recordings, antes de eliminar el eMeeting se ejecuta este archivo que guarda en la base de datos la URL de acceso a las grabaciones una vez que ha terminado la grabación

### **7.2. Sincronización de los usuarios de Adobe Connect**

Cuando un usuario accede a un eMeeting y no tiene una cuenta de acceso en DigiMar, se crea un usuario temporal en Adobe Connect para poder darle los privilegios que se precisen a convenir en la URL desde la que accede a la aplicación.

Este usuario es borrado de Adobe Connect dos días después de su creación. Para ello en el campo descripción que ofrece Adobe Connect se incluye el texto: “” y en el email incluimos la fecha de creación ya que Adobe Connect no incluye este campo en los objetos que almacenan a los usuarios.

## **Parte IV**

# **Primera Versión**

En la primera versión no se dispone de grupos para alimentar a la aplicación, por tanto la aplicación meeting creará grupos de forma dinámica, independientes para cada usuario, y también realizará videoconferencias de forma similar a una llamada telefónica de forma que el usuario seleccionará de una lista a los usuarios con los que desea realizar la videoconferencia, y se creará la sala de forma dinámica. El número máximo de conferencias simultáneas sigue siendo una limitación para la aplicación, pero en este concepto se supone que nunca se llegará a tal número, para lo cual se realizará un interfaz de seguimiento para saber el número máximo de conferencias simultáneas y otros parámetros de interés. De esta forma la lógica de número de salas será transparente para el usuario.

## 8. El Middleware

En el fichero `middleware.wsdl` indicaremos el formato de información que directorio plus envía a nuestra aplicación en formato XML. Esta información se procesará en las distintas funciones implementadas en la aplicación Meeting para rellenar los campos necesarios en nuestra base de datos, para los usuarios y grupos pertenecientes a la plataforma Campus do Mar. Algunas de estas funciones son: `createUser`, `EnjoinGroup` o `addGroup`. Como sus nombres indican, estas funciones a partir de los flujos generados por directorio plus hacia nuestra aplicación, generan el contenido de la base de datos necesaria para la aplicación meeting de forma automática y en el momento que se hace uso de la aplicación, quedando almacenado para sucesivos accesos, todo ello de forma completamente transparente para el usuario.

## 9. Definición de la Base de Datos de la aplicación

Diferentes tablas con sus campos en la base de datos:

1. Meeting: Esta tabla describe las reuniones que los usuarios o grupos realizan para utilizar el Adobe Connect. Es la tabla más importante de la aplicación. Id Title State Salt SecretSalt Duration Title User Date Finish-Date
2. Users: Describe los usuarios que pertenecen a Campus del Mar y ya han accedido alguna vez a una reunión. Id Name Surname Type email
3. Users\_staffs: Relaciona los usuarios con los diferentes grupos creados en campus do mar. Group\_Id User\_Id
4. Users\_meetings: Relaciona las reuniones con el usuario propietario o creador de la reunión Meeting\_Id User\_Id
5. Staffs: Describe los grupos de Campus del Mar que han realizado alguna reunión Id Name Type Key
6. Recordings: Almacena las grabaciones de cada eMeeting. Campos: id, description, title, dateCreated, meeting, url.
7. NickName: Relaciona eMeetings con diferentes campos para un acceso relacional rápido. Campos: id, NickName, meeting, user, rank.

### 9.1. Relaciones entre las diferentes tablas

Gracias a la potencia de Symfony, podemos realizar con facilidad relaciones en las tablas de la base de datos, de forma que podemos acceder, por ejemplo, al nombre de un grupo a partir de un usuario que pertenezca a ese grupo sin más que realizar las instrucciones:

```
$Group_name = $user->getGroup()->getName()
```

Muchas de estas asociaciones son indispensables para cumplir con facilidad los permisos que damos a cada “role” de usuario sobre las reuniones. O para acceder a información dentro de los controladores de aplicación o los servicios, incluso para interactuar con el Adobe Connect.

En esta versión se realizan diálogos modales para la creación de reuniones. En estos diálogos podemos seleccionar el nombre de la reunión, una breve descripción, seleccionar si la reunión es pública o privada.

En otro diálogo modal que se generará por cada reunión creada podremos agregar más usuarios y visualizar los que ya pueden disfrutar del eMeeting.

En una tercera funcionalidad que se crea por cada reunión podremos ver las grabaciones de cada reunión.

Se dispone de una botonera que a mayores de la creación de reuniones nos permite acceder a nuestro histórico de reuniones y grabaciones, donde con un menú seleccionaremos el año y mes de la lista de reuniones que aparecerán en la pantalla.

Para los administradores se crea un acceso directo a la base de datos de la aplicación que nos permite gestionar reuniones y usuarios directamente sobre la base de datos, pero de una forma rápida y cómoda.

## 10. Seguridad de la Aplicación

En el fichero `security.yml` de Symfony tenemos configuradas tres opciones de logueo en nuestra aplicación, la ofrecida por `html` con una caja donde podemos introducir login y contraseña, la que ofrece Symfony mediante la redirección a un formulario que podemos maquetar a nuestro gusto como puerta de acceso a nuestra aplicación. Esta forma de acceso es la que utilizaremos en la primera versión ya que no se dispone de los usuarios y grupos que alimentarán al servidor CAS.

El fichero `routing.yml` también debe ser configurado para que el formulario nos redirija a la página adecuada. Symfony compara la información introducida con la almacenada en la base de datos de la aplicación configurando adecuadamente los parámetros.

De esta forma podemos alternar fácilmente entre 3 tipos de logueo sin más que comentar y descomentar unas cuantas líneas en nuestro fichero `security.yml`.

## 11. Funcionalidades del interfaz de Usuario

### 11.1. Botonera global

#### 11.1.1. + eMeetings

Con este interfaz podremos crear una reunión llamada eMeeting. En la ventana modal que se desplegará podremos seleccionar el título de nuestro eMeeting,

un NickName con el que nos conectaremos a la sala, usuarios de "DigiMar" a los que invitaremos a nuestro nuevo eMeeting y podremos elegir si la sala es pública o privada.

#### **11.1.2. Admin**

Con este botón se accede a la base de datos de la aplicación a través de la propia aplicación y operar sobre ella directamente con un interfaz gráfico sencillo para evitarnos así conocer los diferentes comandos de cada base de datos con la que podemos trabajar.

#### **11.1.3. eMeeting Help**

Este botón nos da acceso a la página de ayuda de eMeeting, donde los usuarios podrán resolver sus dudas y plantear nuevas dudas para que sean resueltas.

#### **11.1.4. Adobe Connect Test**

Este botón realiza un test de conexión que nos permite determinar posibles fallos de conexión sobre el servidor que aloja a Adobe Connect.

#### **11.1.5. Achievement Files**

En esta sección se almacenarán todos nuestros eMeetings pasados una vez los eliminemos y sus correspondientes grabaciones.

#### **11.1.6. Change Password**

Tenemos acceso a una ventana modal que nos permite cambiar el password de la aplicación.

### **11.2. Cuadro de acceso al eMeeting**

Una vez creada la sala, aparece en el centro de la página un recuadro azul con los detalles de nuestro eMeeting creado.

#### **11.2.1. Botonera del cuadro de acceso**

La visión de nuestro interfaz nos ofrece un botonera con las diferentes funcionalidades disponibles para actuar sobre nuestro eMeeting.

Join, para ingresar en la sala del eMeeting en el Adobe Connect, un botón para habilitar y deshabilitar la sala a nuestro gusto, un botón para eliminar la sala. Un párrafo con información sobre la sala:

1. Fecha de creación de la sala

2. Tipo de sala ya sea pública o privada
3. Número de usuarios DigiMar invitados al eMeeting

La siguiente parte de nuestro interfaz nos permite visualizar las grabaciones del eMeeting si las hubiera y cambiar nuestro NickName para cada eMeeting.

### **11.3. Funciones PHP asociadas a la botonera**

#### **11.3.1. Join**

Este botón realiza una serie de comprobaciones antes de llevarlos al eMeeting correspondiente en Adobe Connect. Comprueba el número de eMeetings simultáneos ya que el número está limitado por el número de licencias compradas por el administrador del servidor de conexión a Adobe Connect. Comprueba que la sala todavía existe y que se tienen los privilegios necesarios para disfrutar el eMeeting, ya que el usuario creador ha podido desahabilitar la sala o incluso eliminarla antes de que la página se recargue y por tanto todavía aparezca como activa en nuestro interfaz.

#### **11.3.2. Enabled/Disabled**

Con este botón el creador del eMeeting puede habilitar y deshabilitar la sala a su voluntad impidiendo la entrada a todos los usuarios, incluso aunque tengan acceso desde la puerta trasera creada en las salas privadas.

#### **11.3.3. Delete**

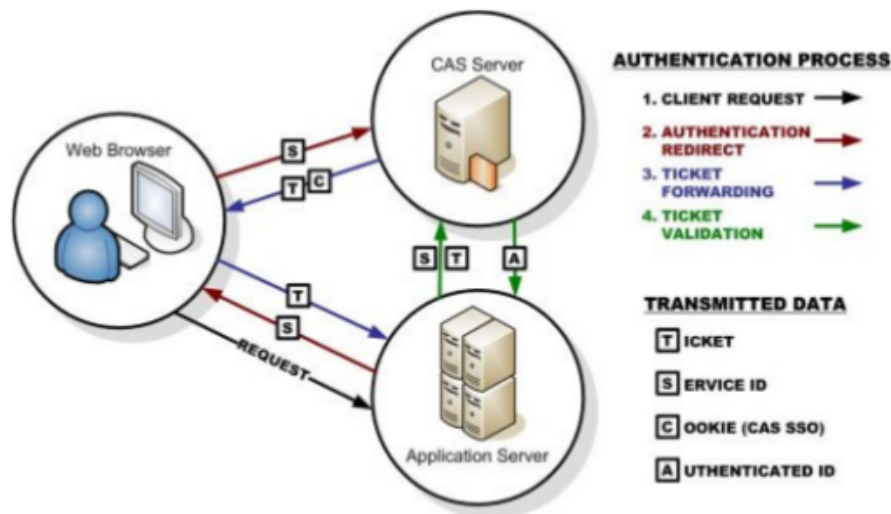
Con este botón eliminamos la sala de nuestro interfaz de usuario y también de todos los usuarios invitados al eMeeting. Únicamente el creador del eMeeting tiene el privilegio de eliminar su sala para evitar futuros conflictos y potenciar que los usuarios creen sus propias salas.

Se comprueba si el eMeeting posee grabaciones y se almacenan en una carpeta interna de Adobe Connect.

En el botón antes descrito como "Achievement Files" tendremos acceso a las grabaciones de los eMeetings eliminados.

#### **11.3.4. Recordings**

Con este botón podremos visualizar las grabaciones realizadas en el eMeeting hasta el momento. Se accede por AJAX al EndPoint correspondiente de Symfony para conseguir los valores actualizados de nuestra lista de grabaciones. Más adelante hablaremos de la función que se ejecuta en el servidor cada minuto para almacenar las nuevas grabaciones en nuestra base de datos al finalizar dicha grabación.



Cuadro 2: Servidor CAS

#### 11.3.5. Change Nick Name

En la ventana modal que despliega este botón se nos da la opción de cambiar nuestro "NickName" en cada reunión. Este código comprueba que no insertamos un "NickName" vacío, aunque sería posible en el Adobe Connect la posibilidad de no tener "NickName"

## Parte V

# Segunda Versión

## 12. Seguridad de la Aplicación

Para entrar a la página de reuniones debemos encontrarnos logueados mediante un servidor CAS (Servicio Centralizado de Autenticación), implementado para este servicio y otros de la plataforma de Campus del Mar.

CAS: Control de sesiones mediante tickets de uso único Single Sign-On. Permite autenticar contra múltiples backends, solución ventajosa en calidad/precio en comparativa con otras disponibles y testeadas.

CAS (Central Authentication Service) es una aplicación web que nos permite implementar el conocido SSO (Single Sign On) que es un procedimiento de autenticación que habilita a un usuario para acceder a distintas aplicaciones web (en distintos dominios y en distintos servidores) con hacer login una única vez. En general, cuando un usuario se conecta a una de estas aplicaciones el sistema

comprueba si está autenticado y si no lo está lo redirige a la pantalla del servidor de autenticación. Si la autenticación es correcta el sistema de autenticación, en este caso CAS, vuelve a redirigir al usuario a la página a la que quería acceder en un primer momento. El desarrollador no tiene que preocuparse por mantener la seguridad y tener un formulario de login en cada una de las aplicaciones web que desarrolla, sino que simplemente tiene que hacer la comprobación de si el usuario ya está registrado, que es tan sencillo como preguntar a la request de esta forma: `request.getRemoteUser()`. Si el método devuelve null es que no está registrado y será redireccionado a la página de login de CAS. Si ya está registrado por CAS, este método devolverá un valor con el que sabremos que es un usuario registrado. Algo que tiene que quedar muy claro es que CAS se encarga única y exclusivamente de la autenticación es decir, de comprobar contra una fuente de datos específica si el usuario y contraseña facilitados existen, NO se encarga de la autorización, que sería la gestión de lo que puede o no puede hacer ese usuario en función de sus roles. INCLUIR MÁS INFORMACIÓN SOBRE CAS

## 13. Concepto de Grupo

El concepto de grupo en esta aplicación es muy importante, ya que está destinado a que fácilmente, grupos de investigadores de las diferentes Universidades integradas en el Proyecto del Campus de Excelencia, Campus del Mar, puedan realizar una videoconferencia de forma rápida, sencilla y que englobe a componentes de un grupo que necesitarán este servicio de forma continuada a lo largo de su investigación.

Iniciamos el proceso realizando una lógica para saber si hay más salas concurrentes de las disponibles que en este caso 10. No debemos saber el número total de salas en el intervalo de duración de la reunión, si no que debemos saber el número máximo de salas en cada instante de tiempo para dicho intervalo, es decir, a lo largo de la duración de la reunión, debemos comprobar si en algún momento el número de reuniones es mayor que el número de salas disponibles, para poder así reservar una sala para la nueva reunión, o devolver un mensaje al usuario de que no hay salas disponibles. También se introduce la lógica de número máximo de reuniones simultáneas para un usuario, de forma que un usuario no puede realizar más de una reunión de forma simultánea. Se solucionó un BUG en el cuál un usuario que no crea la reunión podía cancelarla si tenía acceso al enlace que cancela dicha reunión, sin aparecer el botón de cancelar en su perfil.

En una primera versión, podemos crear una reunión personal de carácter público e invitar a los usuarios de la aplicación que queramos, de echo, el link de la reunión aparece en la lista de reuniones públicas de los demás usuarios para que puedan entrar a la reunión si lo desean. Sin embargo, las reuniones para los grupos son privadas y sólo se pueden acceder si se pertenece al grupo para el cuál se ha creado la reunión. Sólo se pueden crear reuniones de carácter inmediato y de una duración de 30, 60 o 90 minutos, para usuarios y grupos. Más adelante se implementará un “Wizard” que permitirá la reserva de salas de



reunión en el futuro, con la posibilidad de determinar: duración, fecha y control de acceso, tanto para usuarios como para grupos. La lógica de la aplicación para usuario y grupo es muy próxima excepto por algunas sutiles diferencias que explicaremos.

## 14. Página de usuario

Existe un histórico de reuniones, en el cuál podremos observar, todas las reuniones del usuario, tanto canceladas como terminadas, en una primera versión, esta lista era únicamente de reuniones terminada. En la siguiente versión si amplió la lista a reuniones canceladas, y en todas estas reuniones se podían visualizar las grabaciones, en caso de existir, de estas reuniones. Debido al alto número de reuniones pueden aparecer en esta lista, se decidió acotar el histórico, y así seleccionamos el mes al cual queremos acceder, y en esa propia caja de selección podemos visualizar para cada mes el número de reuniones y el total de grabaciones sin acceder al histórico de forma que el usuario tendrá mayores posibilidades de éxito en la búsqueda de su reunión o grabación pasada. En este punto se modifica la apariencia de la página para que la experiencia del usuario sea más intuitiva y satisfactoria, ofreciéndole más opciones a la hora de actuar sobre su reunión. El usuario puede incrementar en 30 minutos la duración de su reunión una vez iniciada sin más que presionar un botón. Se incluye un contador que nos dice el tiempo restante de reunión y el número de salas disponibles. INCLUIR MÁS INFORMACIÓN SOBRE COMO SE IMPLEMENTA ESTA FUNCIÓN

Existen diferentes roles de usuarios y grupos que implementaremos más adelante, para determinar los permisos de acceso de los usuarios a las reuniones. Dentro de los diferentes roles de usuario existe uno muy importante para nuestra aplicación como es el usuario anónimo que no pertenece a Campus del Mar. Para dar mayor flexibilidad a la aplicación es posible mantener una reunión con un usuario no perteneciente a Campus del Mar, para ello cuando se procesa la invitación y el usuario no se encuentra en la base de datos de Campus del Mar, en una plantilla creada para este fin, puede elegir su propio nombre de usuario y de forma rápida la aplicación le permite entrar en la reunión si ésta está configurada para tal fin. Esta plantilla también permite al usuario que pertenece a Campus del Mar introducir su usuario y contraseña para que a nivel estadístico aparezca reflejada la reunión en su perfil. En una nueva modificación un usuario puede crear varias reuniones simultáneas y sólo se comprueba si la sala está ocupada cuando se intenta crear la nueva reunión, en la página del usuario aparecerán todas las reuniones creadas para cada grupo, ya que un usuario puede crear o participar en reuniones de diferentes grupos y en caso de que un grupo tenga un secretario que cree las reuniones este puede visualizar las reuniones de todos los grupos para los cuales las haya creado.

## 15. Página Controladora

La página inicial es un enlace a las dos páginas que describiremos a continuación. En la primera versión esta página contiene un listado de las salas de usuario y de todos los grupos a los que pertenece. De esta forma puede visualizar si existe alguna reunión actual en ejecución a la que el usuario tiene acceso y si la sala está libre para reservarla para una reunión, todo ello mediante un sencillo código de colores. Se muestran enlaces a las salas para poder reservar las salas. Estos enlaces nos redirigen a las páginas que comentaremos a continuación.

## 16. Página de Grupo

La lógica de la página de grupo es muy similar a la de usuario salvo en pequeños detalles que hacen imposible a la hora de programar evitar crear el procesamiento de la información con los mismos conceptos. Por ejemplo, cuando creamos una reunión un usuario decide quienes participarán en la reunión, sin embargo el concepto de grupo del Campus del Mar obliga a programar una lógica en la cual los usuarios que formen parte del grupo reciban una invitación y se encuentren en disposición de participar en la reunión del grupo ya que la sala está reservada y preparada para la reunión. Del mismo modo el control de acceso para las salas de grupo es distinto que para las salas de usuario.

## 17. Concepto de sala

Como consecuencia de la existencia de grupos de investigación en la Plataforma de Campus del Mar nace el concepto de Room. Este concepto no se refleja en la lógica de la base de datos, pero se tiene muy presente a la hora de diseñar y mostrar la aplicación al usuario. De forma transparente al usuario, se crea una sala de videoconferencia para cada usuario y una sala para cada uno de los grupos que tienen cabida en Campus do Mar, de esta forma podemos mostrar al usuario los estados de sala(room) disponible u ocupada, tanto para usuario como para grupo, más adelante se intentará implementar una lógica que nos indique si ya hay alguien dentro de la sala de videoconferencia o no. De momento podemos mostrar si la sala está libre y, por tanto, podemos reservarla para una videoconferencia o por el contrario está ocupada. En nuestra lógica de base de datos, las salas libres son en realidad reuniones(meeting) almacenadas en la database “meeting” y que tienen estado NOW, NEW, SCHEDULE o CANCELLED. Con estos 4 estados sabemos si la sala está libre si el estado es distinto de NEW, o sala ocupada si el estado es NEW, para el “meeting” correspondiente. Debido a los diferentes tipos de salas y usuarios es necesario implementar un Control de Acceso para las salas. Para esta lógica definimos el concepto de Salt y SecretSalt, el cual nos ayudará a construir la url que dará acceso a las diferentes salas de la aplicación. Este campo que almacenamos asociado a cada uno de los “meetings” creados, tiene dos partes, “Salt”, que generamos con el login de

usuario o el nombre del grupo, según corresponda y “SecretSalt” que se genera con el algoritmo MD5 a partir del login de usuario o del nombre del grupo. “Salt” construye la url de acceso para las salas públicas y “SecretSalt” construye las url para las salas privadas. Esta decisión se toma a partir del momento en que se decide crear todas las salas públicas en el Adobe Connect y gestionar el control de acceso desde la propia aplicación meeting. De esta forma a partir de la url generada el usuario puede decidir invitar a cualquier persona, ya sea miembro de la Plataforma Campus del Mar o no. Este planteamiento pareció altamente útil para dar una mayor flexibilidad a la hora de trabajar con personal ajeno a Campus del Mar, pero que en un momento dado puede llegar a colaborar en un proyecto de investigación. Este control de acceso se implementa en la primera versión con la siguiente lógica: Room: Usuario: Pública: Entrada con Salt: Pueden entrar los usuarios que conozcan la sala del usuario ya que la url se construye con el login del usuario que la crea Entrada con SecretSalt: Pueden entrar todos los que conozcan la url generada con el SecretSalt de la sala. Privada: Entrada con Salt: Sólo puede entrar el usuario, no tiene mucho sentido ya que un meeting para un sólo usuario no necesita comunicación. Entrada con SecretSalt: Pueden entrar todos los que conozcan la url generada con el SecretSalt de la sala.

Grupo: Pública: Entrada con Salt: Pueden entrar los usuarios que conozcan la sala del grupo ya que la url se construye con el nombre del grupo al que pertenece la sala. Entrada con SecretSalt: Pueden entrar todos los que conozcan la url generada con el SecretSalt de la sala. Privada: Entrada con Salt: Pueden entrar sólo los usuarios que pertenezcan al grupo, y serán automáticamente invitados. Entrada con SecretSalt: Pueden entrar todos los que conozcan la url generada con el SecretSalt de la sala.

## 18. Elaboración del estilo de la aplicación a partir de plantillas CSS3 y HTML 5

En el archivo cmar.css y meeting.css tenemos nuestras hojas de estilo para la aplicación “meeting”, como es obvio en el archivo cmar.css tenemos el estilo común a toda la plataforma Campus del Mar y en meeting.css tenemos el estilo específico y necesario para cada uno de los conceptos relatados a lo largo de este proyecto.

## Referencias

- [1] Zaninotto F., Potencier F. Symfony 1.2, la guía definitiva
- [2] Potencier F. Jobeet
- [3] ICB Editores, Programación De Páginas Web Dinámicas Con Apache, Base De Datos Mysql Y Php

- [4] Domine Php Y Mysql. 2ª Edición, López Quijado, José
- [5] Rodríguez Mª L., Garrido J. L., Noguera M., Hurtado Mª V., Polo J. R.  
Diseño de interfaces de usuario para aplicaciones colaborativas a partir de  
modelos independientes de la computación

Gracias a todos los que han colaborado a la realización de este proyecto.