

UNIVERSIDADE DE VIGO

**DEPARTAMENTO DE  
ENXEÑERÍA TELEMÁTICA**

**ANTEPROYECTO**

**Diseño, desarrollo y puesta en servicio de una aplicación  
web para la gestión de salas virtuales de videoconferencia  
Adobe Connect®**

**AUTOR**

Eduardo Alonso Gil

**TUORES**

Yolanda Blanco Fernández

Vicente Goyanes de Miguel

## Objetivos

Los mecanismos de comunicación por videoconferencia siempre han sido una herramienta esencial para el correcto desarrollo empresarial, ya que el aprendizaje y comunicación entre sedes alejadas territorialmente, ya sea de la misma empresa o de empresas del mismo sector, son sin duda fundamentales para el desarrollo ágil y la búsqueda de soluciones. También en el ámbito universitario nos encontramos con colaboraciones entre universidades situadas en diferentes comunidades autónomas o países, con una gran demanda de mecanismos de comunicación, tanto para ampliar sus grupos de investigación como para poder compartir sus hallazgos con la creciente y globalizada comunidad científica. Estos escenarios nos ofrecen múltiples roles de usuarios altamente diferenciados, pero con una necesidad común.

El presente proyecto surge de la necesidad de abastecer de una plataforma de videoconferencia a usuarios de múltiples roles y diferentes ámbitos de conocimiento abarcando así un mayor mercado. Se busca una relación de compromiso entre manejo y complejidad para que cualquier tipo de usuario pueda disfrutar de la aplicación de una manera intuitiva. Se buscará un proyecto modular para facilitar futuras ampliaciones y adaptaciones a los perfiles de usuarios de organizaciones con funcionalidades específicas.

Para realizar esta aplicación haremos uso del *framework* específico para PHP Symfony en su segunda versión revisada, una potente herramienta que simplificará nuestro trabajo a la hora de diseñar la aplicación. Nos aportará un escenario excelente para poder ampliar el proyecto en caso de ser necesario y como toda plataforma de software libre, cuenta con una gran comunidad que lo respalda, lo cual nos ayudará a una rápida resolución de errores. Hemos elegido PHP 5 porque puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno.

Como motor de videoconferencia utilizaremos el Software propietario Adobe Connect<sup>®</sup> que posee un interfaz perfecto para multitud de comunicaciones, ya sea a nivel académico o profesional. Este software también posee una potente API que nos permitirá sacar el mayor partido a las funciones de Adobe Connect Profesional<sup>®</sup> y controlarlas desde nuestra aplicación. La experiencia nos hace saber que el uso de Adobe Connect Profesional<sup>®</sup> es complejo para cualquier usuario que no posea un perfil técnico y la multitud de opciones disponibles hace que su interfaz sea poco intuitivo.

Para almacenar toda la información de nuestra aplicación, tanto de los usuarios como de sus reuniones de videoconferencia, utilizaremos MySQL

5.5, ya que su velocidad, característica importante para aplicaciones web, es ideal. Siguiendo con la filosofía de software libre, se acotará enormemente el tiempo de búsqueda de soluciones ante errores que puedan surgir durante la fase de desarrollo.

Uno de los mayores objetivos de este proyecto es ofrecer un interfaz intuitivo que haga popular el servicio sin necesidad de conocimientos técnicos o lecturas extensas. Para ello usaremos la biblioteca de Javascript jQuery. Su gran capacidad para personalizar el diseño de nuestra página y la facilidad para crear funcionalidades y animaciones avanzadas nos proporcionará unos magníficos resultados. Las interacciones AJAX son otro punto a tener en cuenta a la hora de utilizar jQuery ya que simplifica enormemente su utilización y nos da la capacidad de disminuir el ancho de banda utilizado, así como de reducir la velocidad de acceso a la aplicación.

Hoy día es muy importante que las aplicaciones puedan estar en contacto con otras, para obtener datos o simplemente sincronizar información relevante. Un ejemplo sencillo sería comunicar una agenda con nuestra aplicación, de este modo si se agenda una reunión de videoconferencia nuestra aplicación reservará una sala que se encontrará disponible en nuestro perfil. Para ello debemos dotar a la aplicación de las funciones necesarias para comunicarse con otras, o con un servidor que centralice todas las peticiones de sincronización.

Otra funcionalidad será la visualización de un historial de utilización, de este modo obtendremos, en tiempo real, información para predecir la saturación de las salas disponibles o cuáles son los días de mayor carga de peticiones. Para ello usaremos los datos proporcionados por la API de Adobe Connect® y los visualizaremos haciendo uso de la librería karmic-graphs de jQuery.

## Plan de Desarrollo

El trabajo de este proyecto se organiza de acuerdo al siguiente plan de desarrollo:

- Consulta de bibliografía del *framework* Symfony/Symfony 2.
- Consulta de bibliografía del lenguaje PHP.
- Consulta de bibliografía del lenguaje Javascript.
- Consulta de bibliografía de la biblioteca jQuery de Javascript.
- Consulta de bibliografía de la base de datos MySQL.

- Definición de los roles de todos los posibles usuarios de la aplicación.
- Estudio de las bases de datos relacionales.
- Implementación de consultas a la base de datos.
- Implementación de las diferentes lógicas de registro de usuarios (CAS, formulario y HTML básico).
- Estudio de la API de Adobe Connect Profesional®.
- Despliegue del software necesario en el servidor de la aplicación y en el servidor de Adobe Connect Profesional®.
- Prueba del software desarrollado.

## Medios materiales

- Servidor para alojar a Adobe Connect Profesional®.
  - Placa base:
    - Intel® Xeon® processor 5600/5500 series, con QPI up to 6.4 GT/s.
    - Intel® 5520 (Tylersburg) Chipset.
    - Up to 192GB DDR3 1333/ 1066/ 800MHz ECC Registered DIMM / 48GB Unbuffered DIMM.
    - Dual Intel® 82574L Gigabit Ethernet Controller.
    - 6x SATA2 (3 Gbps) Ports via ICH10R Controller.
    - 3 (x8) PCI-E 2.0, 1 (x4) PCI-E 2.0 and 1 (x4) PCI-E slot.
    - Integrated Matrox G200eW Graphics.
    - LSI 2008 8-Port 6Gbps SAS Controller; RAID 0, 1, 10; RAID 5 optional.
    - Integrated IPMI 2.0 with Dedicated LAN.
  - Microprocesadores:
    - 2 x Intel Xeon E5645 Six core 2.4Ghz.
  - Memoria RAM:
    - 6 x DDR3 2GB PC1333.
  - Discos duros:
    - 2 x SAS 300GB 15000RPM Hitachi.
  - Sistema operativo Windows Server 2008 r2.
  - Base de datos SQL Server 2008 r2.

- Servidor LAMP que alojará la aplicación realizada.
  - Memoria RAM: 512MB.
  - Disco Duro: 7.3GB.
  - Microprocesador: Intel(R) Xeon(R) CPU E5420 @ 2.50GHz.
  - PHP 5.3.6.
    - Paquetes de PHP utilizados:
      - ◊ sqlite3.
      - ◊ curl.
      - ◊ pdo\_mysql.
      - ◊ ldap.
      - ◊ soap.
      - ◊ php53-mbstring.x86\_64.
  - Subversion
  - jQuery JavaScript Library v1.7.1.
  - jQuery UI 1.8.18.
  - jQuery karmicgraphs.
  - Base de datos MySQL 5.5 en utf8.
- PC con Ubuntu 12.04 para el desarrollo de la aplicación.
  - Memoria RAM: 4GB.
  - Disco duro: 455GB.
  - Microprocesadores: 4 x Intel(R) Core(TM) i3-2100 CPU @ 3.10GHz.
  - Symfony 2.
- Editor de texto emacs.

## Bibliografía

- Zaninotto F., Potencier F. “*Symfony 1.2, la guía definitiva*”
- Potencier F. “*Jobee*”
- <http://symfony.com/>
- ICB Editores, “*Programación De Páginas Web Dinámicas Con Apache, Base De Datos Mysql Y Php*”
- López Quijado, José, “*Domine Php Y Mysql 2ª Edición*”
- López Quijado, José (2010). “*Domine Javascript 3ª edición*”. RAMA EDITORIAL