



Máster en Software Libre

Trabajo Final de Máster

Administración Web y Comercio electrónico

INTEGRACIÓN DE MÓDULOS A LA RED kPAX

Institución: **Fundación para la Universitat Oberta de Catalunya - EIMT**

Alumna: **Cecilia Cámara López**

Consultor: **Francisco Javier Noguera Otero**

Profesor responsable: **Daniel Riera Terren**

Junio 2015

Licencia de publicación del documento

Esta obra esta sujeta a licencia libre CC-BY-SA cuyo contenido puede verse en:
<http://creativecommons.org/licenses/by-sa/3.0/es/legalcode.es>



Resumen

El presente proyecto donde se integran los módulos **Desarrolladores** y **Perfil completo del usuario**, tiene como objetivo lograr la compatibilidad en el funcionamiento conjunto de los distintos plugins programados para la plataforma kPAX.

En trabajos anteriores, se han incorporado en forma independientes los siguientes plugins:

“Desarrolladores”, que aporta una mejora visual en cuanto a la presentación de los juegos y otorga una rápida gestión por parte de sus desarrolladores y administradores. Éstos podrán, además de publicar su juego, añadir detalles como imágenes, vídeo, una breve descripción, etc. Lo cual ayudará a que los usuarios reconozcan el juego de una manera más atractiva para tener una mayor aceptación en la plataforma.

“Perfil completo del usuario” , no sólo muestra los datos personales, que el usuario quiera compartir con los demás usuarios, sino que despliega los logros alcanzados en la práctica de los diferentes juegos disponibles, cada uno de los cuales requiere una o diversas habilidades, que se acreditan mediante las puntuaciones y sus respectivos premios obtenidos. También, las condecoraciones (badges) o reconocimientos recibidos por su dedicación, su constancia, etc., es decir, por su participación en la red social.

kPAX es una plataforma en crecimiento que surge de una propuesta de los Estudios de Informática, Multimedia y Telecomunicación de la UOC. Este proyecto integrará estos dos módulos solucionando los inconvenientes de integridad que se pudieran producir para ampliar su funcionalidad.

Índice de contenido

| | |
|---|----|
| Licencia de publicación del documento..... | 2 |
| Resumen..... | 3 |
| Introducción..... | 6 |
| Objetivos..... | 7 |
| Estudio de viabilidad..... | 8 |
| Establecimiento del alcance del sistema..... | 8 |
| Estudio de la situación actual..... | 8 |
| Servicio web svrkpax..... | 9 |
| Entorno de producción..... | 9 |
| Entorno de desarrollo..... | 10 |
| Base de datos..... | 10 |
| Requisitos del sistema..... | 11 |
| Riesgos del proyecto..... | 11 |
| Análisis del sistema..... | 12 |
| Definición del sistema..... | 12 |
| Requisitos exactos..... | 12 |
| Identificación de usuarios..... | 16 |
| Especificación del plan de pruebas..... | 16 |
| Diseño del Sistema..... | 17 |
| Arquitectura..... | 17 |
| Integración de los módulos..... | 20 |
| Incorporación del módulo Desarrolladores..... | 20 |
| Incorporación del módulo Perfil completo del usuario..... | 22 |
| Implantación..... | 24 |
| Aplicaciones Auxiliares..... | 25 |
| Puesta en marca de la versión..... | 27 |
| Integración del módulo Desarrolladores..... | 29 |
| Actualización del core..... | 29 |
| Actualización de los módulos..... | 30 |
| Modificaciones importantes..... | 35 |
| Integración del módulo Perfil completo del usuario..... | 37 |
| Actualización del core..... | 37 |
| Actualización de los módulos..... | 38 |
| Conclusiones..... | 40 |
| Referencias bibliográficas..... | 42 |

Índice de ilustraciones

| | |
|---|----|
| Figura 1: Esquema de la plataforma kPAX..... | 9 |
| Figura 2: Esquema base de datos – Kpax..... | 19 |
| Figura 3: Modificación de las tablas por Melgar..... | 21 |
| Figura 4: Modificación de las Tablas en Farrerons..... | 23 |
| Figura 5: Máquina virtual de trabajo..... | 24 |
| Figura 6: Modificaciones sobre el archivo GameBO.java..... | 25 |
| Figura 7: Imágenes tomadas sobre la versión inicial..... | 27 |
| Figura 8: Imágenes tomadas sobre la versión inicial..... | 28 |
| Figura 9: Modificaciones sobre el archivo ConstantsKPAX.java..... | 29 |
| Figura 10: Modificaciones sobre el archivo srvKpax.xml..... | 30 |
| Figura 11: Modificaciones sobre el archivo kpaxSrv.php..... | 30 |
| Figura 12: Modificaciones sobre el archivo kpaxOauth.php..... | 31 |
| Figura 13: Incorporación de la pestaña Develop o Desarrolladores..... | 31 |
| Figura 14: Formulario modificado de agregar un nuevo juego..... | 32 |
| Figura 15: Pestaña Game..... | 33 |
| Figura 16: Pestaña Game – Show my games..... | 33 |
| Figura 17: Pestaña Game – Anauthorized games..... | 33 |
| Figura 18: Pestaña Game – Edit..... | 34 |
| Figura 19: save.php - Imagen obtenida con Melg.org..... | 35 |
| Figura 20: Modificación de UserBO- UserBOImp..... | 37 |
| Figura 21: Plugins activados..... | 38 |
| Figura 22: El módulo instalado y funcionando..... | 39 |
| Figura 23: Plugin theme activado..... | 39 |

Índice de tablas

| | |
|--|----|
| Tabla 1: Tablas de la base de datos kpax - módulo Developer..... | 13 |
| Tabla 2: Tablas de la base de datos kpax - módulo Perfil completo del usuario..... | 15 |
| Tabla 3: Modificación del módulo Developers Imagen obtenida con Melg.org..... | 20 |
| Tabla 4: Modificación del módulo Perfil completo del usuario Imagen obtenida con Melg.org..... | 22 |

Introducción

Las Tecnologías de la información y la comunicación han modificado la participación en nuestra vida cotidiana y la educación no ha permanecido ajena a estos avances, siendo uno de las más activas a la hora de adaptarlas e incorporarlas para una mejor respuesta didáctica.

Las redes sociales han dado origen a sistemas dinámicos en constante evolución, que permiten actividades abiertas e interdisciplinarias, focalizadas hacia un aprendizaje basado en la resolución de problemas de manera tal que el estudiante construya su propio conocimiento[1].

Los juegos tienen una presencia destacada en los contextos de aprendizaje informal, incluyen elementos de fantasía y diversión, además de promover la cualidad en los jugadores para comunicarse e interactuar entre sí durante las partidas. Los juegos de ordenador pueden ayudar a los participantes a pensar de manera crítica cuando tienen que establecer vínculos entre la vida virtual y la real.[2]

Los juegos serios favorecen la construcción del conocimiento y el desarrollo de capacidades en el jugador a partir de exhibir diferentes situaciones, casos o problemas de forma lúdica y atractiva.

kPAX es una plataforma de aprendizaje, implementada sobre el motor libre de Elgg. Buscando como finalidad desarrollar una red social sustentada en los juegos serios, los cuales proporcionan al participante de un conjunto de fortalezas físicas y mentales al requerir el seguimiento de un grupo de reglas en orden con la finalidad de un objetivo. Esta plataforma tecnológica abierta, permite ampliar sus funciones mediante módulos (conectores, juegos, simuladores,etc.)[3] Las ampliaciones agregadas a Elgg deben hacerse bajo la forma de conector o plugin y con una estructura concreta. El grupo de trabajo ha añadido un plugin Kpax con funcionalidades orientadas a compartir juegos como recursos de aprendizaje en línea o fuera de línea y que es accesible desde múltiples plataformas.

Objetivos

El objetivo del proyecto se basa en el análisis de los módulos “*Desarrolladores*” y “*Perfil completo del usuario*” e incorporarlos, solucionando los problemas de integración y compatibilidad que pudieran presentarse, generando una nueva versión de la plataforma. La integración de cada plugin a la plataforma implica que no debe interferir con su funcionamiento y que cada funcionalidad que se incorpora, no debe perjudicar a las que ya se encuentran habilitadas.

Para lograrlo, será necesario:

- Comprender la arquitectura de kPAX y los componentes que la forman.
- Definir la metodología de desarrollo del proyecto a utilizar.
- Recopilar y analizar los diferentes plugins ya implementados y la versión inicial de kPAX.
- Incorporar los plugins de manera incremental para obtener una nueva versión estable con las nuevas funcionalidades.
- Añadir la nueva versión pública estable GitHub

Estudio de viabilidad

Establecimiento del alcance del sistema

El proyecto kPAX tiene por objetivo el desarrollo de una red social para el aprendizaje basado en juegos, que tiene la posibilidad de ampliar sus funcionalidades mediante módulos, incorporándolos a la plataforma mediante plugins.

En el proceso de integración será necesario garantizar la funcionalidad integral de la plataforma, para ello deberemos estudiar en profundidad cada módulo, analizando el código de programación y la estructura planteada, con respecto a la base de datos y servicios a utilizar.

Este proyecto incluye el análisis en detalle de cada módulo pero no está previsto el desarrollo de nuevos plugins, como tampoco el análisis de costos y/o presupuesto asignado.

Estudio de la situación actual

La plataforma kPAX ha sido desarrollada a partir de Elgg, que es un motor de redes sociales de código abierto, el cual ofrece un marco sólido sobre el que construir todo tipo de entornos sociales.

Elgg funciona sobre un entorno LAMP,[4] que recordemos combina las tecnologías más utilizadas para definir la infraestructura de un servidor web este caso Apache, utilizado como lenguaje de programación PHP y como gestor de base de datos a MySQL.

La combinación Apache con PHP y servidor de base de datos MySQL se ha convertido casi en un estándar para la mayoría de los desarrolladores. Apache es el servidor más utilizado para mostrar páginas estáticas, y PHP es un lenguaje sencillo y robusto para la generación de páginas html dinámicas, es muy fácil embeberlo en html al ser de tipo script, interpretado y no compilado, y sus funciones para el acceso a bases de datos Postgres y MySQL son eficientes y rápidas.

kPAX tiene una arquitectura de núcleo o core, basada en servicios, separada en tres componentes:

Los clientes: usuarios, y su relación con las redes sociales de las que son miembro como por ejemplo: estadísticas, modificar perfiles de usuarios, conectarse a la red e incluir aspectos de seguridad. Y juegos: status y actividad de las partidas.

El core de la aplicación, desarrollada en lenguaje de programación Java: permite que la plataforma Elgg sea transparente, donde los usuarios se conectan y también gestiona la base de datos. En esta parte hay algunos servicios implementados, como por ejemplo registrar un juego, el control de las puntuaciones, etc.

Visualización: que se implementa mediante módulos o plugins de Elgg. Éstos se activan dentro de la plataforma, y su función principal es leer o enviar información, a la base de datos para presentarla al usuario y recoger sus acciones.

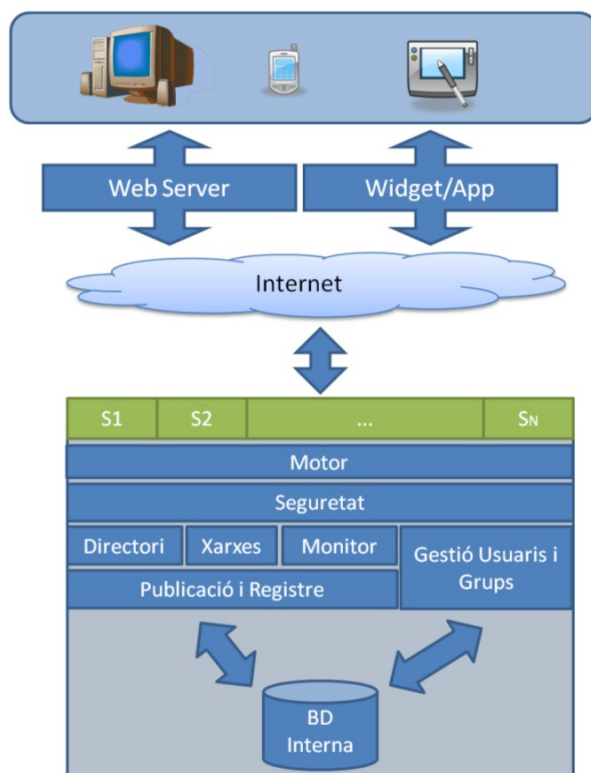


Figura 1: Esquema de la plataforma kPAX

Servicio web svrkpax

El servicio web Kpaxsrv de tipo REST, [5] es una aplicación JAVA que permite acceder a través de llamadas REST para permitir la ejecución remota de ciertos métodos y funciones publicados.

Esta aplicación web llamada svrKpax tiene una arquitectura capas:

- DAO (Data Access Object): capa de acceso a datos.
- VO (Virtual Object): capa de representación de tablas en objetos.
- BO (Bussiness Object): capa lógica de negocio.
- web Rest: capa de servicios.

Entorno de producción

JBoss ejecuta la aplicación Java, generando un fichero .war, que contiene todo el código Java y las librerías y dependencias principales, lo que ofrece son servicios REST a la aplicación kPAX que se ejecuta en el entorno Elgg.

Éste transforma el war en servicios web mediante una API tipo REST que recibe y emite las peticiones [6], y una herramienta de mapeo objeto llamada Hibernate.

Entorno de desarrollo

El entorno de desarrollo para la aplicación Java se compone:

- Entorno integrado, IDE Eclipse Juno (versión 4.2) , el cual emplea módulos para proporcionar toda su funcionalidad al frente de la plataforma de cliente. Es una plataforma ligera de código abierto, que permite a Eclipse extenderse usando otros lenguajes de programación como son C/C++ y Python.
- Gestor de proyectos Apache Maven [7] esta herramienta gestiona proyectos de software y su comprensión. Basado en el concepto de un modelo de objeto de proyecto (POM), Maven puede gestionar de un proyecto de construcción, generación de informes y documentación de una pieza central de la información.
- Repositorio en red servido a través de GitHub, mediante este servicio de alojamiento repositorio Git basado en la web, es posible el control de revisión y gestión de código fuente. Sistema cuyo objetivo es facilitar la administración de las distintas versiones de cada producto.

Algunas características que podemos rescatar:

- La posibilidad de decidir si el código alojado será público o limitado a unos usuarios determinados.
- Un visor de código mediante el cual podemos realizar consultas a través del navegador, resaltando la sintaxis correspondiente al lenguaje a consultar.
- Una wiki para el mantenimiento de las diferentes versiones de un proyecto
- Un visor de ramas para comparar los progresos realizados en las diferentes ramas de nuestro repositorio

Base de datos

La base de datos Kpax soportada en MySQL permite organizar y compartir juegos. Para cada una de las tablas relevantes se crean en el proyecto web al menos tres ficheros Java:

- GameVO es la definición de la tabla como un objeto.
- GameBO contiene la definición de funciones que tienen como objeto esa tabla.
- GameDAO es la clase que ejecutaría la consulta sql que requiere la función definida en lógica de negocios para obtener su resultado, , sino que se comunicará con la Base de Datos a través del componente Hibernate.

Requisitos del sistema

Para incorporar cada módulo debemos analizar los siguientes requisitos:

Requisitos técnicos:

Se incorporarán las funcionalidades de manera incremental, de esta manera se analizará que cada módulo:

- La integración de las bases de datos con otros plugins ya instalados en la plataforma.
- El nuevo código compilado no deben interferir en los servicios o funciones existentes.

Requisitos operativos:

- Establecer una metodología de desarrollo del proyecto.
- Añadir la nueva versión pública estable GitHub.

Requisitos legales:

- Mantener la compatibilidad con la licencia.

Requisitos económicos:

- No existen requisitos económicos.

Riesgos del proyecto

Se estima que existe una serie de riesgos que pueden atentar contra el desarrollo del proyecto:

- La falta de experiencia de los participantes en este tipo de actividad, podría dar lugar a estimaciones de tiempo, esfuerzos inadecuados y superposición de la tarea.
- Desconocimiento sobre el grado de calidad con que han sido desarrollados los plugins.
- Problemas de funcionalidad al integrar nuevos módulos.

Análisis del sistema

Definición del sistema

Como fue mencionado, el punto de partida para el proyecto será la rama estable del proyecto, para el caso la bifurcación del código original que realizara el tutor externo¹, tanto para el núcleo como para los módulos que permiten el funcionamiento de K-PAX sobre la plataforma Elgg.

Requisitos exactos

En la expectativa de logros a alcanzar hemos enfatizado en lograr que cada módulo incorporado debe ser compatible con el funcionamiento conjunto de la plataforma.

R1: Se incorporarán los plugins de manera incremental, de esta manera se analizará que cada módulo:

- La integración de las bases de datos con otros plugins ya instalados en la plataforma.
- El nuevo código compilado no deben interferir en los servicios o funciones existentes.

Para cumplir este requisito será necesario un análisis detallado de cada módulo.

Plugin Desarrolladores [8]

Funcionalidades que el módulo agregará a kPAX

- Una mejora visual de todo lo relacionado con la presentación de los juegos de la plataforma además de una rápida gestión de los mismos por parte de sus desarrolladores y administradores de la plataforma.
- Los desarrolladores podrán, además de publicar su juego, añadir detalles del mismo como son imágenes, un vídeo, descripción, etc. Dependerá de cada desarrollador el aspecto final que tenga la página principal de su juego de manera que atraiga al mayor número de jugadores posibles

En la base de datos Kpax:

¹<https://github.com/drierat>

| Tabla | Descripción |
|-------------------------|---|
| Game | Modificada – Información para describir los juegos. |
| Category | Representa la categoría del juego. |
| Comment | Almacena distintos los comentarios realizados sobre un juego. |
| Tag | Guarda la información de una etiqueta y el juego al que está asignada. |
| GameDetail | Almacena detalles de los juegos. |
| GameImage | Guarda las URL de las imágenes del juego. |
| GameView | Vista de la tabla Game con algunos datos sobre el uso y comentarios. |
| GameSimilitudeView | Vista Sobre Game mostrando categorías o etiquetas similares entre juegos. |
| TotalGameSimilitudeView | Vista que cuenta las similitudes con un juego. |

Tabla 1: Tablas de la base de datos kpax - módulo Desarrolladores

Requisitos del sistema

- 1: un sistema de identificación y autenticidad de usuarios que accedan a la plataforma. Esto permitirá al usuario acceder a distintas funcionalidades según su tipo de usuario.
- 2: gestionar los permisos de usuarios según sean desarrolladores, administradores o usuarios comunes.
- 3: un usuario administrador de la plataforma también puede ser desarrollador de juegos.
- 4: todos los usuarios han de poder ver el listado de juegos autorizados de la plataforma. Los desarrolladores han de poder ver además sus juegos que no hayan sido aún autorizados.
- 5: los usuarios administradores han de poder modificar los datos de los juegos si lo creen conveniente y autorizarlos.

Plugin Perfil completo del usuario[9]

Funcionalidades que el módulo agregará a kPAX

Además de los datos personales, que el usuario quiera compartir con los demás, muestra, por un lado, los logros alcanzados en la práctica de los diferentes juegos disponibles, cada uno de los cuales requiere una o diversas habilidades, que se acreditan mediante las puntuaciones y sus respectivos premios obtenidos.

Asimismo, el módulo deberá mostrar las condecoraciones (badges) o reconocimientos recibidos por su dedicación, su constancia, etc., es decir, por su participación en la red social.

Requisitos del sistema

- 1: un sistema de identificación de usuario
- 2: sistema de permisos que distingue administradores, usuarios registrados usuarios no registrados.
- 3: permiso de consulta de resultados para cualquier usuario registrado.

En cuanto a la base de datos de kPAX: se crean varias tablas que representarán las nuevas entidades y las nuevas interrelaciones:

| Tabla 1: Tabla | Descripción |
|-----------------------|--|
| Ability | Representara las habilidades genéricas (ciencias, humanísticas, idiomas, historia, etc). |
| Proficiency | Representa las habilidades más específicas (dentro de ciencias: matemáticas, biología ...; dentro de humanísticas: economía, derecho ..., dentro de idiomas: catalán, español, inglés). |
| AbilityProficiency | Representa la interrelación entre ambas entidades. |
| Award | Representa los diversos premios, cada uno dentro del ámbito de una habilidad específica y con un grado máximo de logro. |
| UserAware | Representa la interrelación entre ambas entidades. |
| AwardImage | Representa las diversas iconos que corresponden a cada premio (una por cada nivel posible de logro y de color diferente). |
| Usability | Representa la interrelación entre ambas entidades. |
| Badge | Representa las condecoraciones o los reconocimientos que se otorgan al usuario para la participación en los diferentes juegos. |
| UserBadge | Interrelación entre user y badge. |
| Platform | Representará las diferentes plataformas en las que se puede ejecutar un juego (en vistas a una futura ampliación de las funcionalidades de la plataforma). |
| PlatformGame | Interrelación entre platform y game. |
| Performance | Representará los diferentes "récord" obtenidos por el usuario, es decir las mejores puntuaciones obtenidas en los diferentes juegos; en cierta medida es redundante con relación a la entidad GameScore pero se implementa por facilitar la eficiencia del código y con vistas a futuras nuevas funcionalidades. |

Tabla 2: Tablas de la base de datos kpax - módulo Perfil completo del usuario

R2: Establecer una metodología de desarrollo del proyecto.

Se adopta una división del proyecto en tres fases:

Inicial

Se estudia cada módulo, iniciando por Desarrolladores, analizando que modificación se hace sobre la bases de datos kpax y el funcionamiento en general.

Desarrollo

Se procede a introducir los cambios en el código y base de datos.

Implementación

Se realizan las pruebas necesarias para verificar su funcionalidad.

R3: Añadir la nueva versión pública estable GitHub.

Siguiendo con los lineamientos del proyecto y los requerimientos establecidos en puntos anteriores, para cada módulo a incorporar, deberán ser tenidos en cuenta sus propios requisitos, de acuerdo a su documentación. Luego de la integración, éstos deben ser verificados.

Identificación de usuarios

Administrador: será quien tenga todos los privilegios dentro de la plataforma, será el rol de máximo mando dentro de la aplicación.

Desarrollador: pertenecerá a todo usuario común que tenga o suba algún juego a la plataforma

Usuario común: representa a todo usuario del sistema. Puede jugar juegos, realizar actividades de redes sociales (comentar, buscar amigos) y en este caso buscar juegos.

Especificación del plan de pruebas

En el proyecto se han implementado distintos niveles de pruebas del sistema, siempre partiendo de la misma configuración de la máquina virtual inicial, entre ellas son:

Pruebas unitarias: cada módulo será implementado, probado y analizado.

Pruebas de integración: en esta etapa del proyecto hemos tomado dos vías por las cuales cada uno en forma independiente integrará dos módulos, en mi caso he tomado la integración del módulo Desarrolladores y el módulo de Modificación de Usuario.

Diseño del Sistema

La especificación del diseño del sistema nos permitirá identificar los componentes del mismo y proporcionará especificaciones necesarias para el desarrollo y la implementación.

Arquitectura

La plataforma k-PAX está dividida en 3 capas: presentación, lógica de negocio y persistencia que permite que cada elemento trabaje a un único nivel de abstracción, donde la capa de presentación y lógica de negocio utilizan el patrón de diseño MVC (modelo vista- controlador), para organizar las responsabilidades de interacción con el usuario.

Capa de presentación

La capa de presentación se ocupará de dar una interfaz gráfica a nivel de usuario, implementada por módulos y es la conexión entre la capa de negocio y el usuario.

- Como base: la Red social Elgg.
- El conector kpax: con el servicio web de la capa de lógica de negocio, que sirven para aumentar las funcionalidades de Elgg.

Capa de lógica de negocio

La capa de lógica se encargará de validar los datos introducidos por el usuario, comprobar que el usuario tiene permiso, realizar las acciones solicitadas, cálculos, comunicarse con servicios externos y servirá para establecer la conexión entre la capa de presentación y de persistencia.

Su interfaz deberá indicar las acciones necesarias para la realización de las operaciones de lógica de negocio del sistema.

En esta capa se encuentran:

- Interfaces y clases BO del package uoc.edu.srv Kpax.business, contienen la lógica de negocio y es la comunicación entre la capa de persistencia (DAO) y la capa de presentación.
- Servicios web definido en las clases del package uoc.edu.srvKpax.rest, los que se utilizan para comunicarse con la capa de presentación.

Capa de persistencia

La capa de persistencia debe interactuar con la base de datos y ser capaz de añadir, modificar, borrar, y consultar los datos, en nuestro caso trabajamos con el motor MySQL y será necesario mantener la configuración deseada para la aplicación en cada ordenador en el que se ejecute.

Encontramos:

- Base de datos MySQL, que contiene toda la información de la plataforma kPAX.
- Clases VO del package uoc.edu.srv Kpax.vo que definen una entidad, así como el mapeo con la base de datos. Se utilizan para interactuar con la base de datos desde

las interfaces DAO.

- Interfaces y clases DAO del package uoc.edu.srv Kpax.dao, que se utilizan para obtener y modificar los datos guardados en la base de datos, a través de las clases VO. Es la conexión entre la base de datos y la capa de lógica de negocio.
- Archivo hibernate.cfg.xml del directorio main / resources, el cual es el archivo de configuración del marco de trabajo de Java Hibernate. Hay diversos parámetros de configuración de la base de datos.

Los módulos que forman parte de kPAX, son:

- apiadmin: este conector es el encargado de la gestión de los certificados de autenticación.
- kpax: añade el menú Games, este es el conector que contiene la biblioteca de peticiones de los servicios web que se necesitan para solicitar las consultas y las actualizaciones de la base de datos de kPAX.
- likekpax: es el conector encargado de gestionar las anotaciones "like this" correspondientes a los objetos propios de la plataforma kPAX.
- loginrequired: este conector que obliga a los usuarios a identificarse antes de poder acceder a la plataforma, es decir sólo permite el acceso a las páginas de entrada (login), de registro (en caso de que el usuario no esté dado de alta) y de petición de recordatorio (en caso de que el usuario la haya olvidado).

El módulo *kpax* es el encargado de facilitar que desde *Elgg* pueda realizarse la gestión de juegos, mediante llamadas para comunicarse con el núcleo de servicios de *k-PAX*.

La característica de la **base de datos Kpax** es la siguiente:

Usuarios:

El sistema dispone de un conjunto de usuarios, los cuales pueden acceder directamente a la plataforma, utilizando su nombre de usuario (login) y contraseña (password), o bien a través de alguna otra red social.

Grupos

Para gestionar el acceso a juegos, los usuarios se pueden organizar en grupos. Todos los grupos deben estar registrados en la tabla Group, y su estructura es jerárquica, pudiendo haber varios grupos raíz. Cada grupo puede formar parte sólo de otro grupo (se indicara en el atributo idGroupParent), aunque no tiene por qué estarlo. Los usuarios pueden pertenecer a tantos grupos como sea necesario, y su afiliación al grupo se realiza utilizando la tabla UserGroup.

Juegos

Para que un juego sea accesible a través de la plataforma kPAX, debe estar correctamente registrado en el sistema. Solo los usuarios autorizados para registrar juegos deben poder hacer el registro, que

consiste en crear una entrada en la tabla Game. Cada juego dispone de un idGame único dentro del sistema. Otras propiedades pueden ser si el juego permite jugar más de un jugador o no, si es colaborativo o no, etc.

Acceso a los juegos

Dado que es posible que se utilicen algunos juegos con finalidades evaluadoras, se puede restringir el acceso a los juegos a los usuarios que la han de poder utilizar. Por este motivo, se pueden asignar permisos de acceso a los juegos por parte de los diferentes grupos. La tabla GameAccess cuenta la información sobre acceso a los juegos por parte de los diferentes grupos.

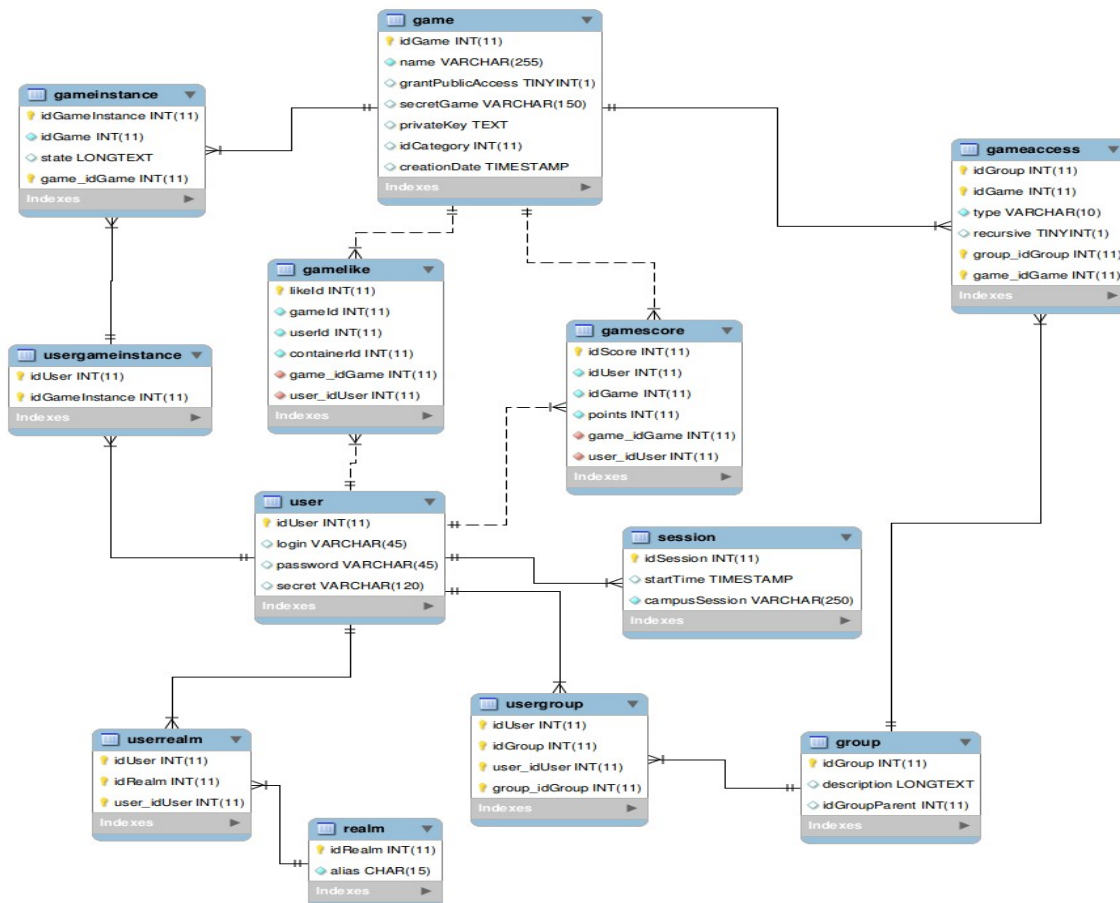


Figura 2: Esquema base de datos – Kpax

Integración de los módulos

La integración se produjo gradualmente, es decir, primero se integró el módulo Desarrolladores, trabajo que realizamos conjuntamente con mi compañero y luego, una vez obtenido una versión estable y funcional, integré el siguiente módulo en forma individual.

Incorporación del módulo Desarrolladores

Este módulo fue integrado en forma conjunta con mi compañero de trabajo y para dicha incorporación fue necesario realizar cambios en el plugin Kpax, en los siguientes archivos:

Modificaciones al servicio SvrKpax

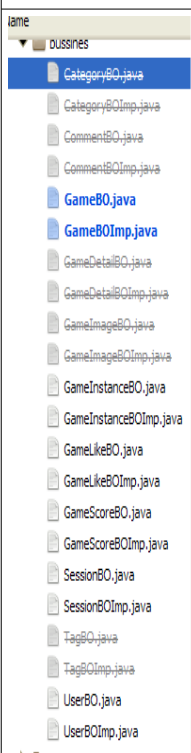
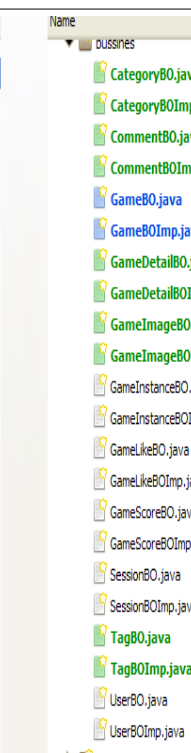
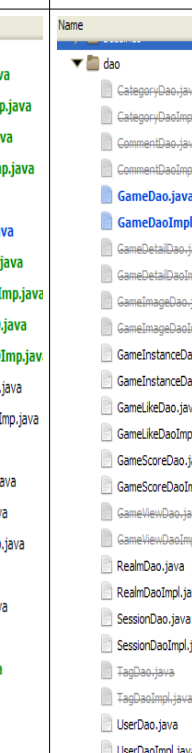
| business | dao | res – util - vo |
|--|--|--|
|  |  |  |

Tabla 3: Modificación del módulo Desarrolladores Imagen obtenida con Melg.org

Modificaciones en la base de datos Kpax

La modificación de la base de datos representa un punto decisivo en el desarrollo de la aplicación, porque a partir de ellas se diseñan los objetos que intervienen en el servicio. Para la incorporación del módulo fue necesario modificar las siguientes tablas, a las ya diseñadas anteriormente, estas son:

- La tabla Game seguirá teniendo como clave primaria el identificador idGame.
- La tabla User seguirá teniendo como clave primaria el identificador idUser.
- La tabla GameDetail tendrá una clave primaria idGameDetail.
- La tabla GameImage tendrá una clave primaria idGameImage.

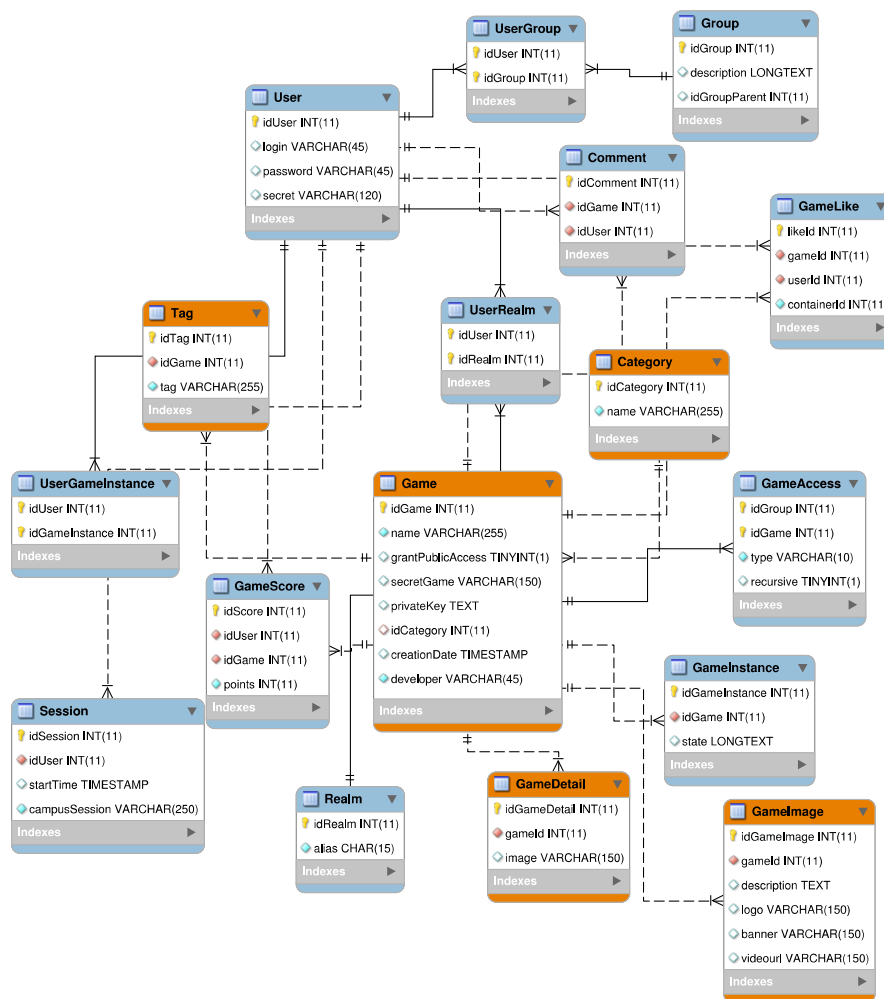


Figura 3: Modificación de las tablas por Melgar

Incorporación del módulo Perfil completo del usuario

Este módulo fue integrado en forma individual y para ello fue necesario incorporar y modificar los siguientes archivos:

Modificaciones al servicio SvrKpax

| business | dao | res – util - vo |
|---|---|--|
| <div> <div>Name</div> <div> <div>business</div> <div> AbilityBO.java AbilityBOImp.java AbilityProficiencyBO.java AbilityProficiencyBOImp.java AwardBO.java AwardBOImp.java AwardImageBO.java AwardImageBOImp.java BadgeBO.java BadgeBOImp.java GameBO.java GameBOImp.java GameInstanceBO.java GameInstanceBOImp.java GameLikeBO.java GameLikeBOImp.java GameScoreBO.java GameScoreBOImp.java ProficiencyBO.java ProficiencyBOImp.java SessionBO.java SessionBOImp.java UserAbilityBO.java UserAbilityBOImp.java UserAwardBO.java UserAwardBOImp.java UserBO.java UserBO.java.antic UserBOImp.java UserBOImp.java.antic UserBadgeBO.java UserBadgeBOImp.java </div> </div> </div> <div> <div>Name</div> <div> <div>business</div> <div> AbilityBO.java AbilityBOImp.java AbilityProficiencyBO.java AbilityProficiencyBOImp.java AwardBO.java AwardBOImp.java AwardImageBO.java AwardImageBOImp.java BadgeBO.java BadgeBOImp.java GameBO.java GameBOImp.java GameInstanceBO.java GameInstanceBOImp.java GameLikeBO.java GameLikeBOImp.java GameScoreBO.java GameScoreBOImp.java ProficiencyBO.java ProficiencyBOImp.java SessionBO.java SessionBOImp.java UserAbilityBO.java UserAbilityBOImp.java UserAwardBO.java UserAwardBOImp.java UserBO.java UserBO.java.antic UserBOImp.java UserBOImp.java.antic UserBadgeBO.java UserBadgeBOImp.java </div> </div> </div> | <div> <div>Name</div> <div> <div>dao</div> <div> AbilityDao.java AbilityDaoImp.java AbilityProficiencyDao.java AbilityProficiencyDaoImp.java AwardDao.java AwardDaoImp.java AwardImageDao.java AwardImageDaoImp.java BadgeDao.java BadgeDaoImp.java GameDao.java GameDaoImp.java GameInstanceDao.java GameInstanceDaoImp.java GameLikeDao.java GameLikeDaoImp.java GameScoreDao.java GameScoreDaoImp.java ProficiencyDao.java ProficiencyDaoImp.java RealmDao.java RealmDaoImp.java SessionDao.java SessionDaoImp.java UserAbilityDao.java UserAbilityDaoImp.java UserAwardDao.java UserAwardDaoImp.java UserBadgeDao.java UserBadgeDaoImp.java UserDao.java UserDao.java.antic </div> </div> </div> <div> <div>Name</div> <div> <div>dao</div> <div> AbilityDao.java AbilityDaoImp.java AbilityProficiencyDao.java AbilityProficiencyDaoImp.java AwardDao.java AwardDaoImp.java AwardImageDao.java AwardImageDaoImp.java BadgeDao.java BadgeDaoImp.java GameDao.java GameDaoImp.java GameInstanceDao.java GameInstanceDaoImp.java GameLikeDao.java GameLikeDaoImp.java GameScoreDao.java GameScoreDaoImp.java ProficiencyDao.java ProficiencyDaoImp.java RealmDao.java RealmDaoImp.java SessionDao.java SessionDaoImp.java UserAbilityDao.java UserAbilityDaoImp.java UserAwardDao.java UserAwardDaoImp.java UserBadgeDao.java UserBadgeDaoImp.java UserDao.java UserDao.java.antic </div> </div> </div> | <div> <div>Name</div> <div> <div>rest</div> <div> Games.java Jsonp.java User.java User.java.antic </div> </div> </div> <div> <div>Name</div> <div> <div>util</div> <div> AES.java ConstantsKPAX.java Oauth.java Security.java </div> </div> </div> <div> <div>Name</div> <div> <div>vo</div> <div> Ability.java AbilityProficiency.java Award.java AwardImage.java Badge.java Game.java GameInstance.java GameLike.java GameScore.java Proficiency.java Realm.java ResQueryAbilities.java ResQueryBadges.java ResQueryProficiencies Score.java Session.java User.java User.java.modern UserAbility.java UserAward.java UserBadge.java </div> </div> </div> |

Tabla 4: Modificación del módulo Perfil completo del usuario Imagen obtenida con Melg.org

Los archivos que están marcados en color verde, son incorporados al sistema y los de color azul, son los archivos que revistan modificación.

Modificaciones en la base de datos Kpax

En la base de datos de kPAX, será necesario realizar modificaciones para representar las entidades: Ability, Badge, Proficiency, Performance, Platform, PlatformGame, Award, AwardImagen, y sus interrelaciones.

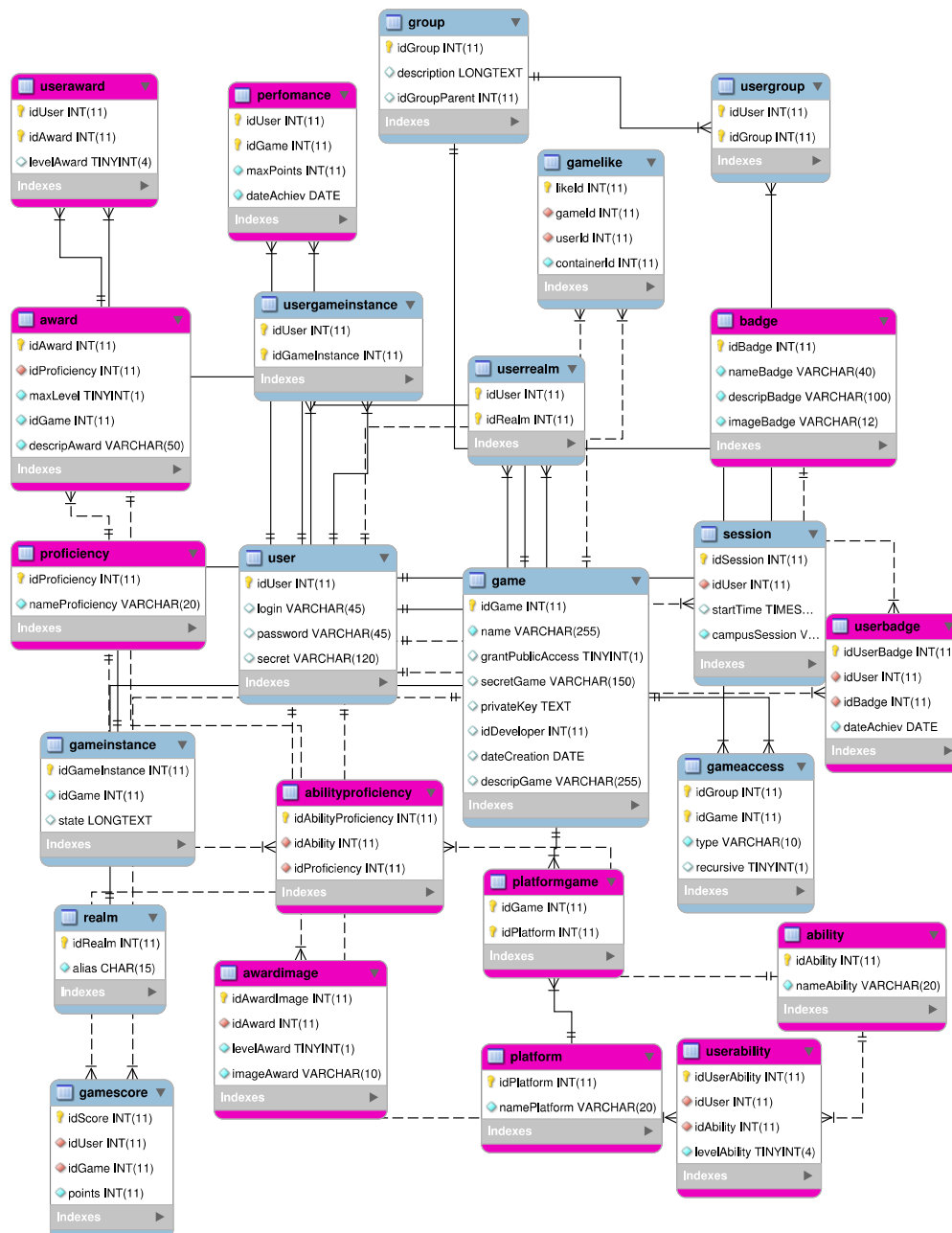


Figura 4: Modificación de las Tablas en Farrerons

Implantación

Cada módulo se ha integrado en forma individual, partiendo de:

1) GitHub: <https://github.com/jsanchezramos>:

En el repositorio encontramos dos carpetas:

- k-pax: svrKpax, servicios creados para la interacción de diferentes aplicaciones. Estos servicios interactúan mediante plugins de Elgg y las aplicaciones que sean necesarias para su funcionamiento.
- Mods-kpax: módulos necesarios para el funcionamiento de kpax en la plataforma Elgg, estos son:
 - ✓ loginrequired, oculta todas las páginas de Elgg, excepto las de inicio, registro y olvido del password, al usuario no autenticado.
 - ✓ kpax, contiene los webservices necesarios para interactuar desde fuera con el servidor Elgg interno.
 - ✓ apiadmin, genera y gestiona los certificados para la autenticación.
 - ✓ likeKpax, gestiona las anotaciones 'me gusta' en los objetos kPax.

2) Mvirtual otorgada por el equipo de tutores.

La misma, posee un disco E: con dos carpetas: (con los servicios kpax svrKpax) y wamp (con elgg, dentro de esta carpeta están incorporados los módulos).

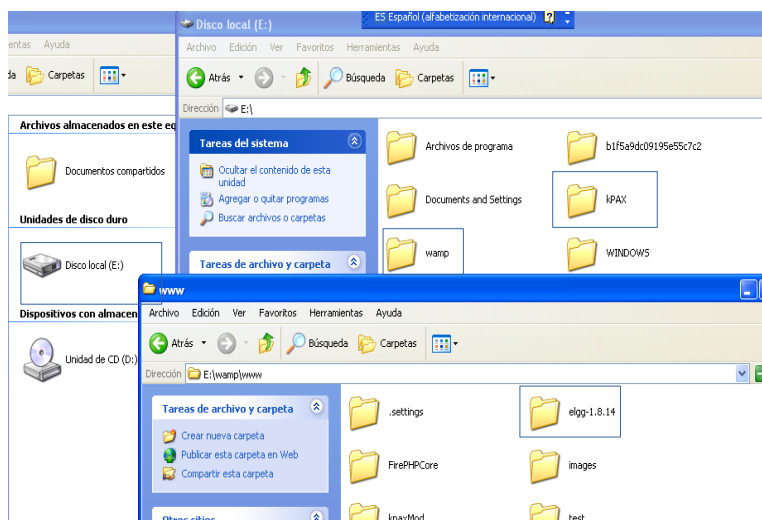


Figura 5: Máquina virtual de trabajo

Aplicaciones Auxiliares

Meld:

Se a incorporado la herramienta Meld,[10] dirigida a los desarrolladores para ayuda a comparar archivos, directorios y proyectos de control de versiones. Se proporciona la comparación de dos y de tres vías de los archivos y directorios, ayudando a revisar los cambios de código y entender parches.

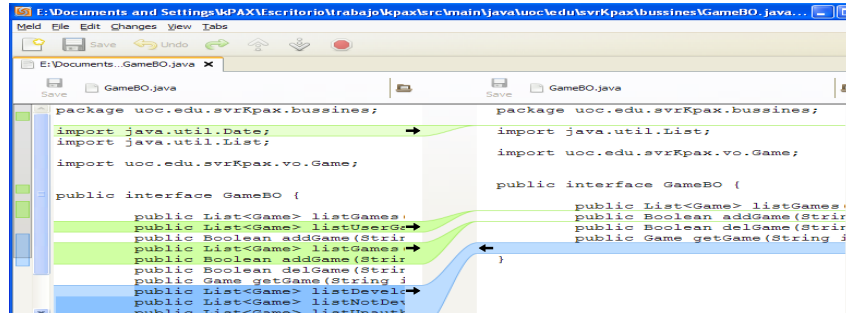


Figura 6: Modificaciones sobre el archivo GameBO.java

Eclipse:[11]

Aplicación instalada en la máquina virtual inicial. Esta herramientas de programación de código abierto y multiplataforma nos sirvió de apoyo para modificar código desarrollado en Java. Creado inicialmente por IBM, pero es mantenido por la Eclipse Foundation desde 2004 y respaldado por una comunidad de voluntarios. Consta como mínimo de editor de texto, compilador, intérprete, depurador y un cliente. Eclipse incluye todo lo anterior y muchas funcionalidades más. Las distribuciones o proyectos de Eclipse pueden ser de varios tipos, y estar organizados en conjuntos de herramientas elegidas para dar una funcionalidad más completa.

Maven [12]

Es una herramienta de construcción de proyectos que abarca los procesos de compilación, empaquetado, despliegue o test, mediante un proceso de aplicación de patrones o plantillas, predefinidas o creadas ad-hoc por el usuario. Además posee otras funciones como gestión de dependencias, gestión de versiones o distribución y despliegue del proyecto. Es un producto de Apache Software Foundation para gestión de proyectos de desarrollo de software en Java.

WampServer:[13]

PHP: lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas.

MySQL: sistema de gestión de bases de datos relacional, multihilo y multiusuario.

Apache: servidor web HTTP de código abierto, para plataformas Unix, Microsoft Windows, y otras que implementen el protocolo.

MySQL Workbench[14]

Es una herramienta visual muy útil para el diseño de las bases de datos, desarrolladores y administradores de bases.

Github[15] es uno de los principales servicios de alojamiento y publicación para proyectos de código empleados en la actualidad. Es una plataforma de código abierto. Está basado en Git, la herramienta creada por Linus Torvalds para gestionar el repositorio del núcleo de Linux.

Puesta en marcha de la versión

La puesta en marcha del sistema en su versión inicial, sobre la máquina virtual con la configuración antes mencionada, funcionó sin inconvenientes, para el sistema operativo Windows y no así para GNU/Linux. El problema que se planteaba era que al instalar las tablas, los nombres de las mismas eran automáticamente incorporados en minúsculas. Por lo tanto, al migrarlas luego a un sistema libre, éstas no podrán ser accedidas si no existe correspondencia entre los nombres de tablas y el código de acceso.

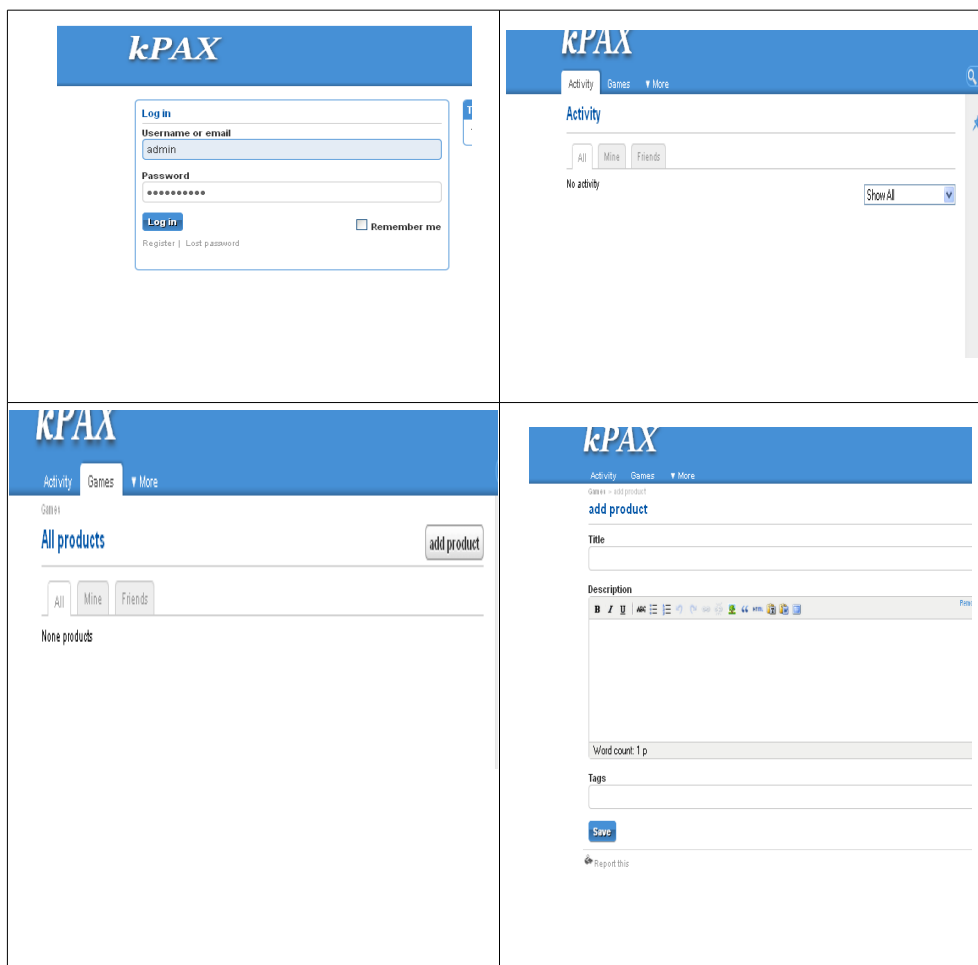
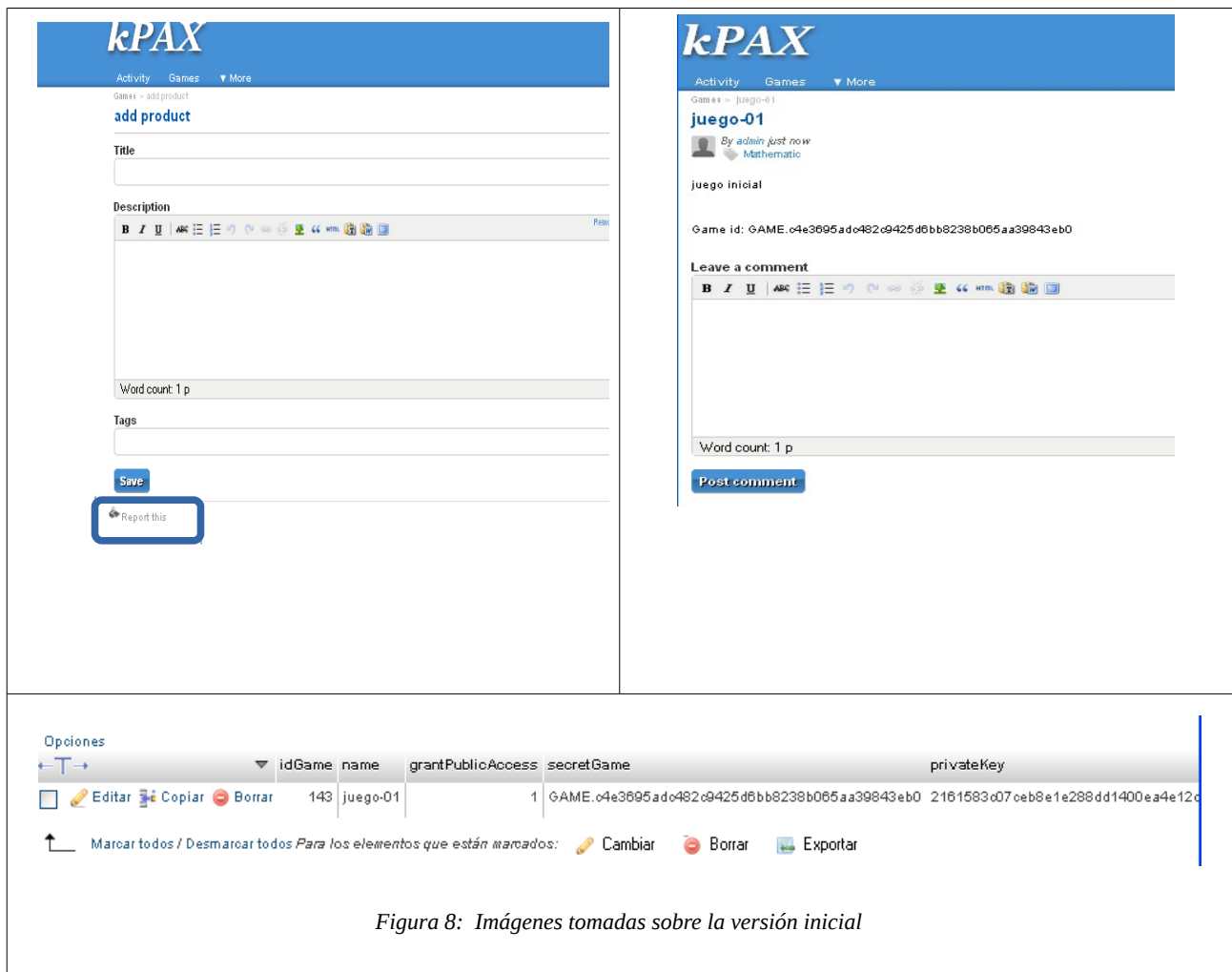


Figura 7: Imágenes tomadas sobre la versión inicial

Prueba Inicial: para comprobar el funcionamiento del sistema se ha ingresado un nuevo juego, tal como se muestra en la figura.



Se registra error en:

“Class 'ElggKpax' was not found, missing plugin?, referer: <http://localhost/elgg/kpax/add/54>”

El cual se ha solucionado creando una clase: ElggKpax.php, dentro del subdirectorio classes.

Y en:

"Undefined index: HTTP_X_ELGG_APIKEY" in fileengine/lib/web_services.php (line 556).

Tal error, correspondiente a la versión 1.8.x de Elgg que no se reparó, puesto que a partir de la versión 1.9 esa función se quita del núcleo y se implementa como un módulo.

Integración del módulo Desarrolladores

Luego de poner en marcha el sistema en versión inicial se procederá a instalar los diferentes módulos previstos en el proyecto. Comenzando por “Desarrolladores”.

El módulo fue entregado vía correo electrónico por el tutor, donde en el archivo comprimido se entrega los archivos correspondientes a las dos carpetas principales: kpax:/ y wamp:/.

La metodología de de trabajo propuesta será:

Actualización del core

A) Sustituir la carpeta KPAX del disco [E:kpax/](#)² con el contenido de la misma carpeta del proyecto.

Como primera medida se verifica el archivo:

```
E://kpax/src/main/java/uoc/edu/svrKpax/util
```

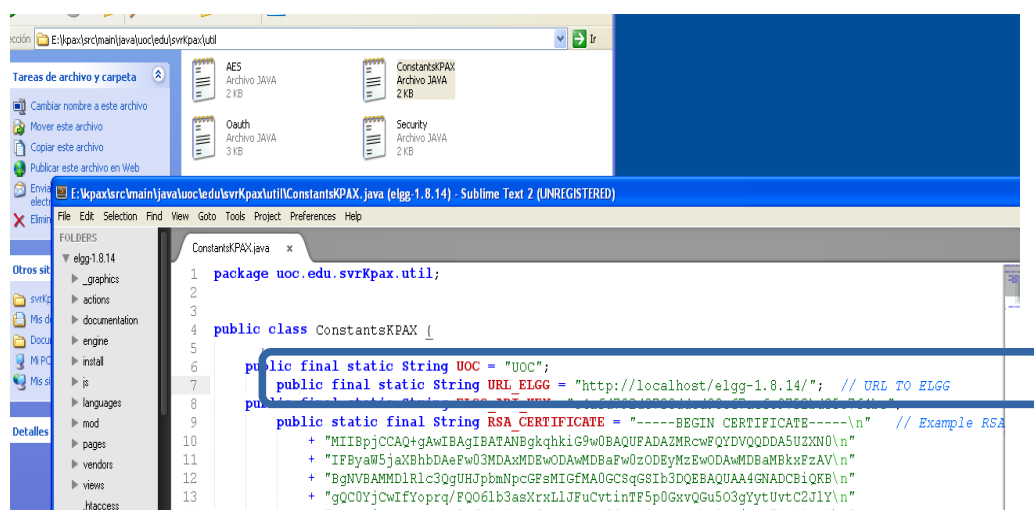


Figura 9: Modificaciones sobre el archivo ConstantsKPAX.java

B) Se creo una base de datos kpax, utilizando kpac.sql. Este archivo se encuentra en la carpeta kpax/doc/sql.

C) Se configura el archivo `srvKpax-ds.xml` con el fin de establecer la configuración de base de datos y almacenarlo donde se implementa el servidor Jboss.

2 Máquina virtual que se utilizará como plataforma de trabajo.

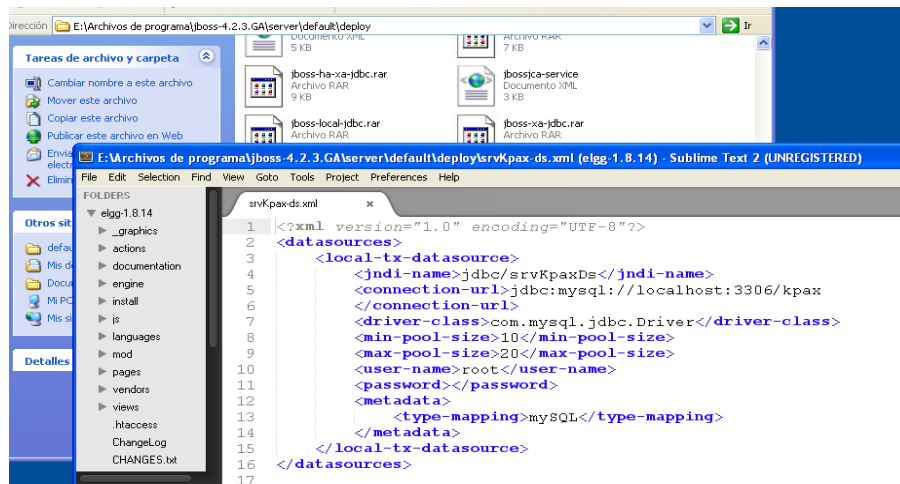


Figura 10: Modificaciones sobre el archivo srvKpax.xml

Este procedimiento de configuración de los archivos de la carpeta kpax, deberá repetirse cada vez que se intente instalar un nuevo módulo. Luego, al instalar el módulo “Perfil completo del usuario” se analizará que archivos deberán modificarse y cuales no es necesario el cambio.

Actualización de los módulos

Estas carpetas fueron copiadas, sustituyendo las existentes en E:/wamp/www/elgg-1.8.14/. Se han modificado los siguientes archivos:

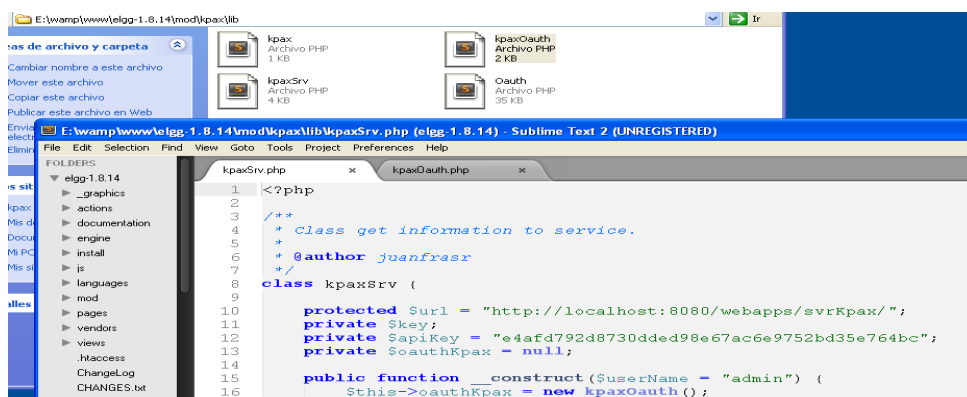


Figura 11: Modificaciones sobre el archivo kpaxSrv.php

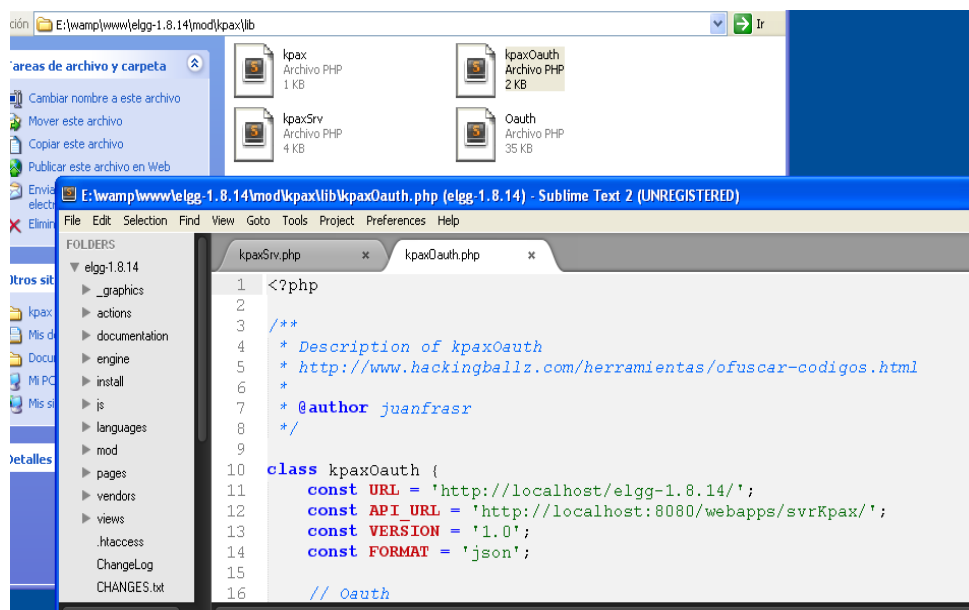


Figura 12: Modificaciones sobre el archivo kpaxOauth.php

Iniciando el sistema podemos observar la siguiente ventana:

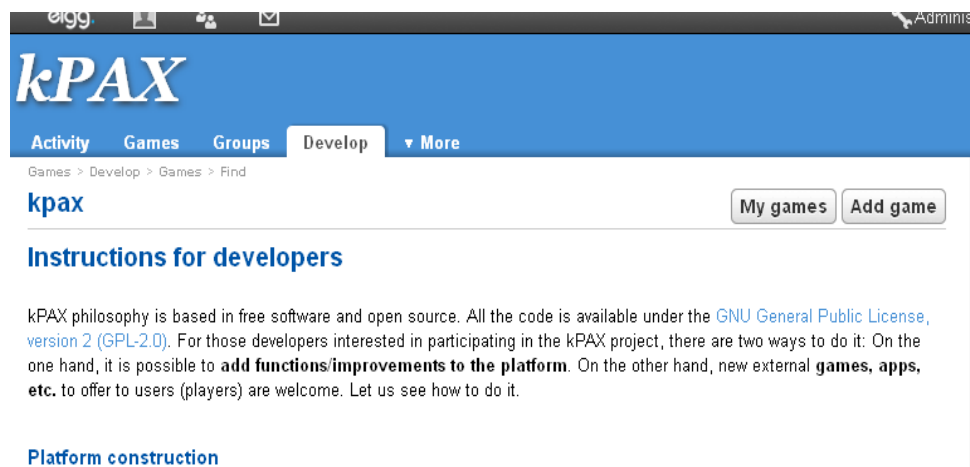


Figura 13: Incorporación de la pestaña Develop o Desarrolladores

Se observa que se ha agregado una nueva pestaña “Develop” con las instrucciones para instalar la plataforma de juegos serios y se agregan dos botones de “My games” y “Add game”.

Add game

Game name (*)

Description

Word count: 1 p

Logo (square image url 160px x 160px)

Game banner (image url)

Youtube Id video

Game category

Category 1

Creation date

01-06-2015 15:18:04

Available platform

Android

Skills (*)

Skill 1

Related tags, comma separated

Images

In certain many images as you want put the URLs separated by a space

Word count: 1 p

Security

In order to connect your game to NPAX, it is necessary to create a specific public/private key pair. It can be done using `openssl`. You only need to run the command:

```
openssl req -out requestGame.csr -new -newkey rsa:2048 -nodes -keyout private.key
```

Once you have run it, you should have two text files: the private key (.pem), which is only for your eyes, and the certificate request file (.csr), which you must include below. NPAX will create and store a certificate in your game's information sheet.

Certificate request file (.csr) (*)

Examinar... No se ha seleccionado ningún archivo.

Send my game!

Figura 14: Formulario modificado de agregar un nuevo juego

Al intentar guardar un juego se generaba un error, mediante los logs de Apache se pudo saber cual era ese error: ... [client 127.0.0.1] PHP Fatal error: Call to undefined function string_to_url_array() in E:\wamp\www\elgg-1.8.14\mod\kpax\actions\kpax\save.php on line 13, referer: <http://localhost/elgg-1.8.14/kpax/add>. Función que fue necesario crearla.

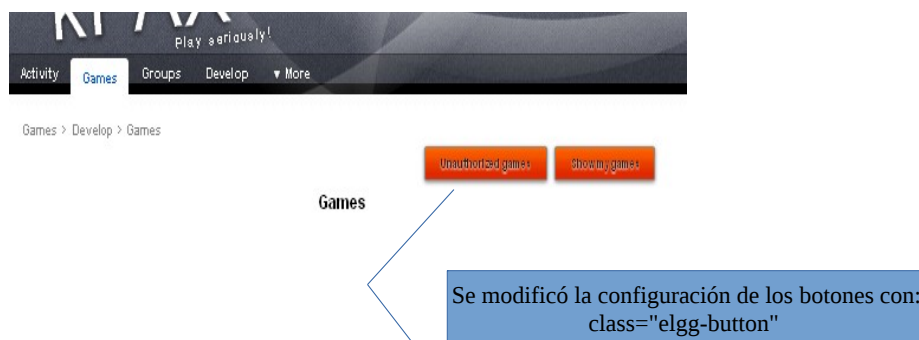


Figura 15: Pestaña Game

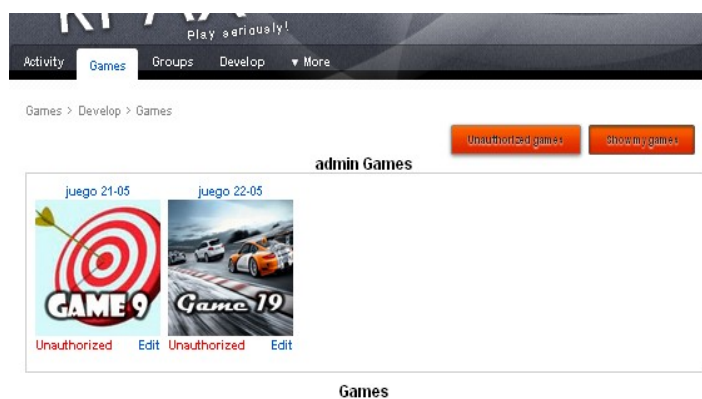


Figura 16: Pestaña Game – Show my games

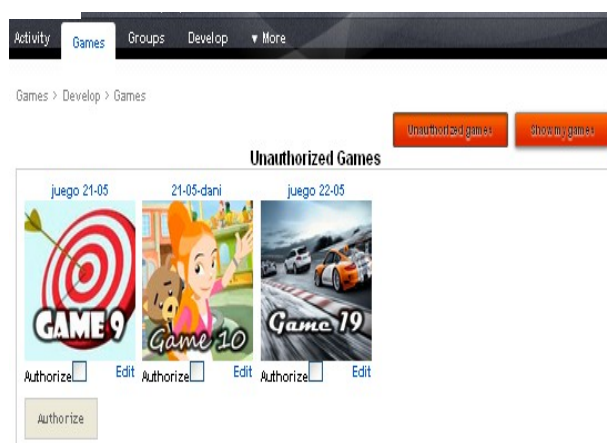


Figura 17: Pestaña Game – Unauthorized games

Se modificó la función “edit” en el ejemplo vemos como se modifica el juego 06, colocando un vídeo y como luego se refleja en la vista del juego.

Games > Develop > Games > edit product

edit product

Game name (*)
juego 21-05

Description
dia 21

Word count: 2 p

Logo (square image url 150px x 150px)
<http://localhost/images/game09.jpg>

Game banner (image url)

Youtube id video
ZFHB4tJHf4

Game categories
Category 1

Creation date
21-05-2015 20:24:00

Available platforms
Android

Skills (*)
Skill 1


Related tags, comma separated

Administration Settings Log out

save product

<http://localhost/elgg-1.8.14/kpax/view/263/juego-21-05>

Images
Select images to delete them



Insert as many images as you want, put the URLs separated by a space

<http://localhost/images/game19.jpg> <http://localhost/images/game22.jpg> <http://localhost/images/game23.jpg>

Word count: 3

Security
In order to connect your game to kPAK, it is necessary to create a specific public/private key pair. It can be done using openssl. You only need to run the command:

```
openssl req -out requestUser.csr -new -newkey rsa:2048 -nodes -keyout private.key
```

Once you have run it, you should have two text files: the private key (.pem), which is only for your eyes, and the certificate request file (.csr), which you must include below. kPAK will create and store a certificate in your game's information sheet.

Certificate request file [csr] (*)
Examinar... No se ha seleccionado ningún archivo.

Send my game!

kPAK
Play seriously!

Activity Games Groups Develop More

kPAK

GAME 9

DRIVECLUB: EL REGRESO DEL CONDUCTOR

10/10


Game 19

0 0 0 0 0 0

Description
dia 21

Other games from the same developer

juego 22-05



De
adr
Lat
Ga

Muestra las distintas imagenes

Figura 18: Pestaña Game – Edit

Modificaciones importantes

El análisis del código de las distintas funciones, se realizó con la herramienta Melg.

Un ejemplo es lo que muestra la siguiente imagen, donde se muestra que parte se modificó y cual no fue necesario actualizarlo.

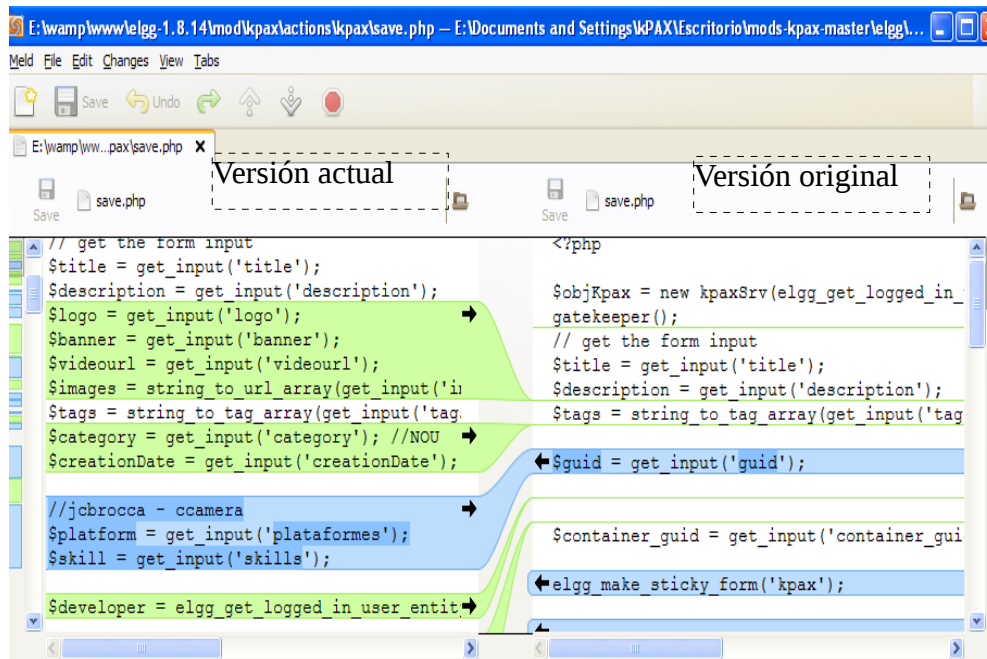


Figura 19: save.php - Imagen obtenida con Melg.org

/actions/kpax/save.php:

Se modifica para permitir incorporar al juego los valores de skill y platform. Se realiza una categorización sobre los posibles errores de almacenamiento, que antes estaban agrupados bajo una sola denominación.

/languages/:

Se actualizaron los diccionarios en relación a las modificaciones en otros archivos.

/lib/kpax.php:

Se agrupan en una sola función las acciones para obtener la categoría, plataforma y las habilidades de un juego.

/lib/kpaxSrv.php:

Se agregan las funciones para obtener plataformas y habilidades, a la vez que se modifica la función addGame().

/pages/kpax/my_dev_games.php:

Fue nuevamente codificado con características similares a la forma de visualizar los juegos en la pestaña Games.

/pages/kpax/play.php:

Modificaciones sobre las distintas listas de juego.

/pages/kpax/view.php:

Se modifica la acción que permite incorporar las imágenes por defecto cuando al obtener un juego, éste no contiene un logo o un banner. Se reemplazó \$page_owner → name por \$page_owner → username Se evita que en la sección otros juegos del mismo desarrollador se muestre el mismo juego que se está visualizando.

/views/default/forms/kpax/save.php:

Se cambia el código para obtener los valores de category, skill y platform desde un menú desplegable.

kpax/games_list.php:

Se modifica la referencia a las imágenes por defecto utilizando la función elgg_get_site_url() en lugar de la mención directa al servidor. También, la referencia incorrecta a la edición de un juego. No se muestra la posibilidad de editar juegos por quién no sea el desarrollador, y se cambia la visualización de los botones utilizando para ello las clases definidas en Elgg.

Integración del módulo Perfil completo del usuario

El módulo fue entregado por correo electrónico por el equipo de tutores, se ha procedido a la instalación del mismo, que consiste de los siguiente pasos:

Actualización del core

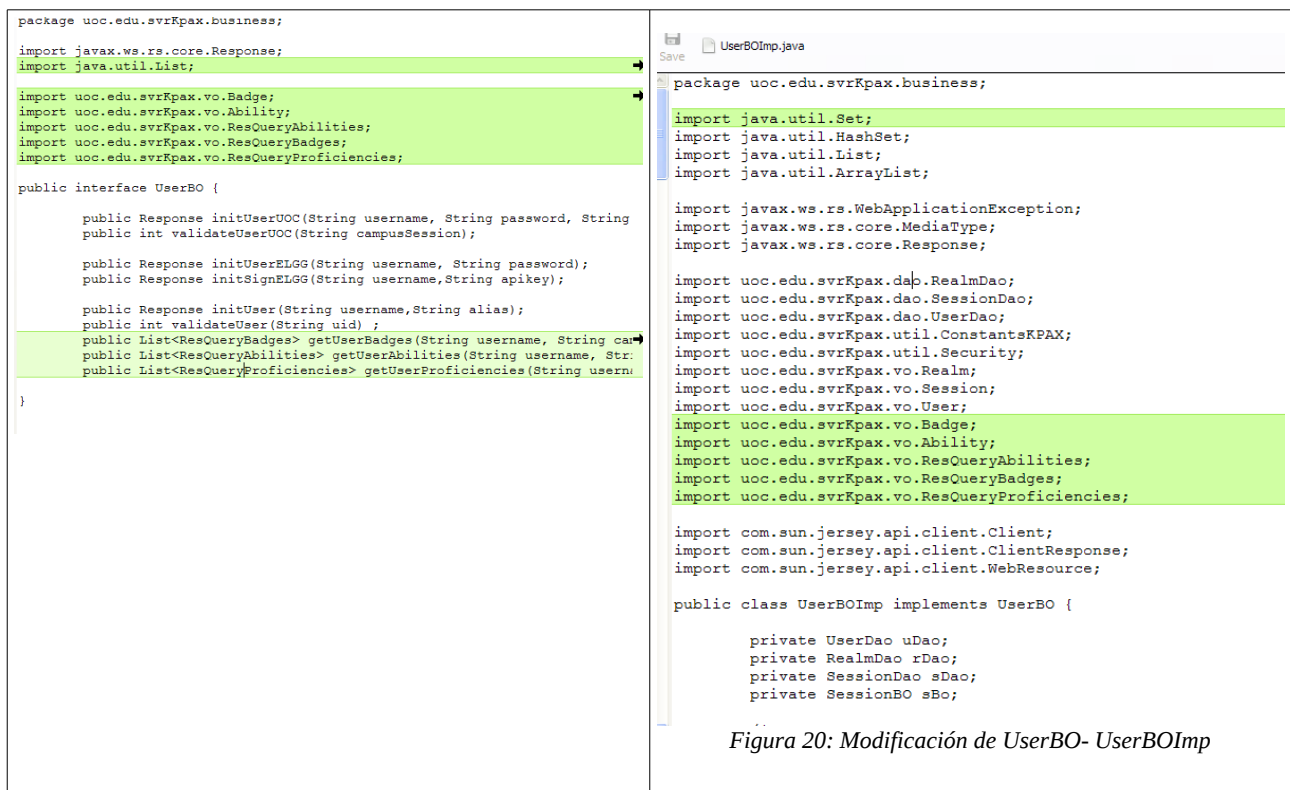
A) Se utiliza la herramienta Melg para detectar que parte de los archivos van a cambiarse y cuales se quedarán sin modificar.

La primer carpeta a comparar es: <E://kpax/src/main/java/uoc/edu/svrKpax/util>. No se detectaron cambios, puesto que los datos a modificar son los mismo para el localhost de la máquina virtual de trabajo.

Fue necesario agregar en la carpeta `E://kpax/src/main/java/uoc/edu/svrKpax/business` las implementaciones en java (BO y BOImp) correspondientes a:

Ability, Badge, Proficiency, Performance, Platform, PlatformGame, Award, AwardImagen, UserAward, UserAbility, UserBadge.

El cambio mas importante se presentó en UserBO y UserBOImp:



B) Al comparar el archivo `kpax.sql` surge las primeras diferencias por las bases de datos que son

incorporadas: Ability, Badge, Proficiency, Performance, Platform, PlatformGame, Award, AwardImagen.

Por lo tanto se incorpora al archivo la creación de las nuevas tablas con sus correspondientes registros.

C) El archivo `srvKpax-ds.xml` no fue necesario modificarlo.

Actualización de los módulos

Se actualizan los diccionarios de la carpeta: `/languages/` para mantener la relación a las modificaciones en otros archivos.

Para evitar conflictos ante un eventual cambio de versión de Elgg se modificó la carpeta `E:\wamp\www\elgg-1.8.14_graphics\kpaxicons` y se la incorporado al módulo que corresponde:

En el módulo: `kpmybadges_widget`

`E:\wamp\www\elgg-1.8.14\mod\kpmybadges_widget\graphics\kpaxicons`

y en el módulo: `kpmyabilities_widget`

`E:\wamp\www\elgg-1.8.14\mod\kpmyabilities_widget\graphics\kpaxicons`

Por último, inician los servicio y se activa los siguiente plugins: activar `kpmypage`, activar `kpmybadges_widget` y activar `kpmyabilities_widget`.



Figura 21: Plugins activados

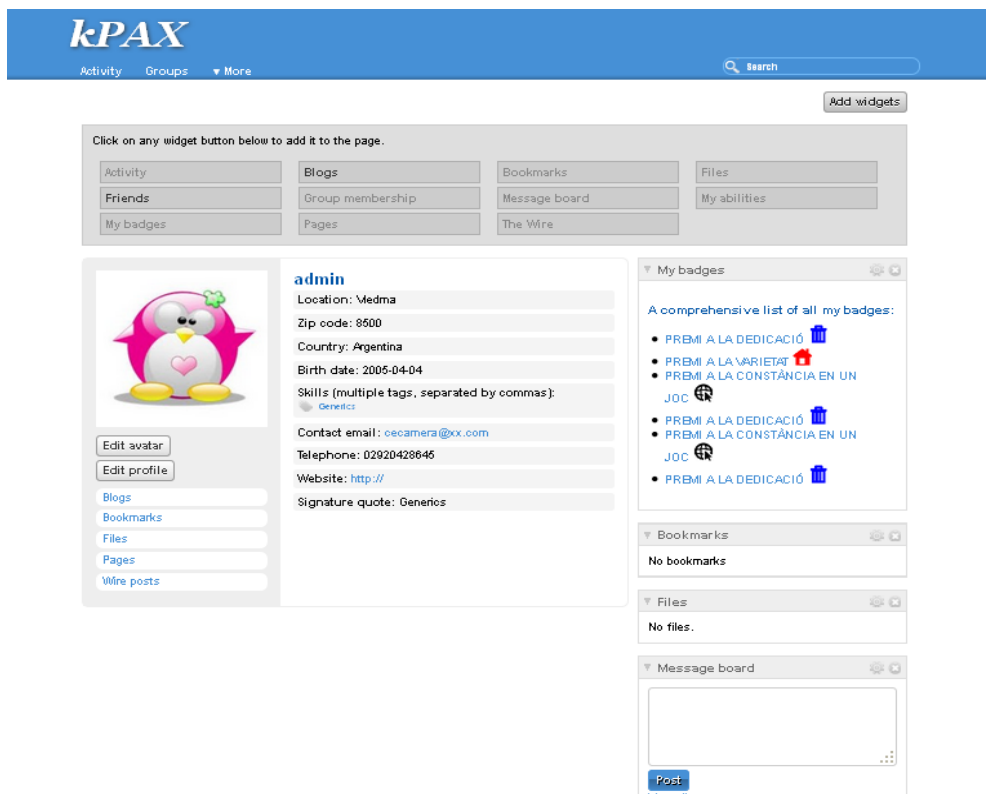


Figura 22: El módulo instalado y funcionando

Aplicando el tema:

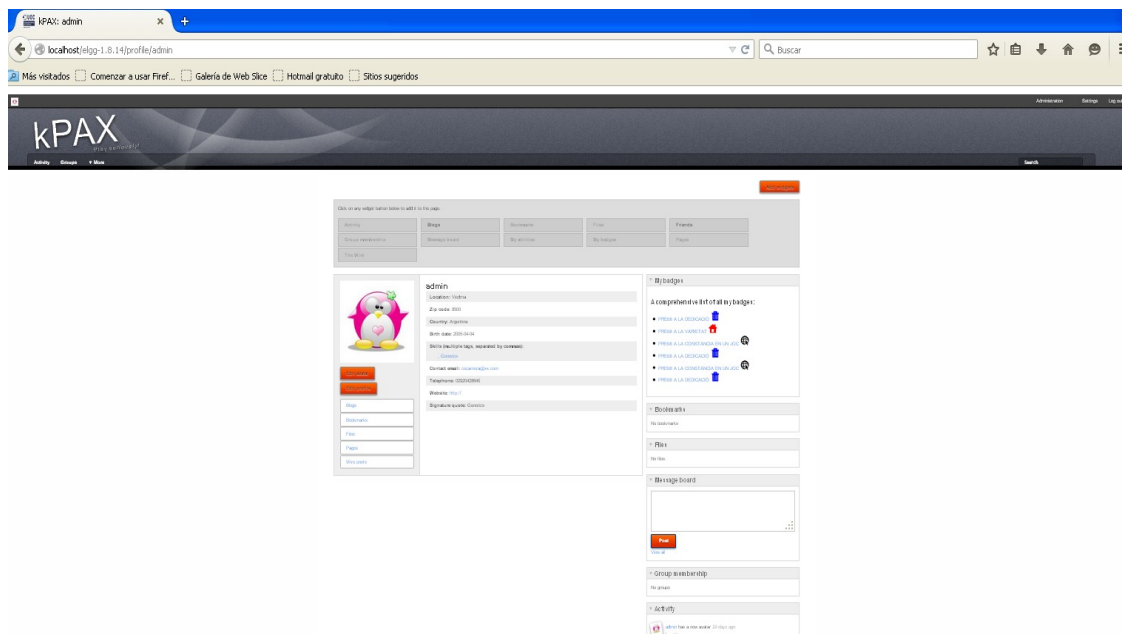


Figura 23: Plugin theme activado

Conclusiones

Como resultado del proyecto emprendido se concluye que se han podido lograr los objetivos planteados al inicio de las actividades.

Se ha estudiado en profundidad tanto la arquitectura kPAX como el conocimiento integral de los módulos, para obtener una descripción detallada del estado inicial del sistema que fue el punto de partida del proyecto.

Dado que el objetivo del mismo se basa en el análisis de los módulos “Desarrolladores” y “Perfil completo del usuario” se ha implementado una metodología de trabajo acordada con el tutor sobre una grilla de encuentros virtuales, permitiendo la organización de las tareas y verificar los requisitos que se deberían cumplir a lo largo del proyecto.

En forma conjunta con el tutor, se decidió incorporar primero al módulo “Desarrolladores” y una vez resuelta la integración, el tutor enviaría el segundo módulo.

Debido a la falta de experiencia en esta actividad, fue necesario profundizar en las funcionalidades de Elgg, entendiendo conjuntamente la interacción con el servidor Apache, MySQL, php, el plugins principal Kpax y el servicio web, tanto en lo que se refiere al código java como a su despliegue en el servidor de aplicaciones JBoss con todos sus componentes.

Los tiempos estimados para esta actividad se vio retrasada con respecto al el esperado inicialmente, debido al desconocimiento sobre el grado de complejidad con que han sido desarrollados los plugins y las dificultades de funcionamiento de la primer tarea a desarrollar.

Finalmente, se ha detectado la falta de algunas funciones que aún no estaban implementadas y que se tuvieron que rehacer o programar.

Esta nueva versión fue incorporada a GitHub para ser evaluada y consensuada con los tutores, para luego recibir el nuevo módulo.

El segundo módulo a incorporar fue más fácil, por supuesto, al contar de la experiencia de haber ya integrado el primero de ellos, se pudo encontrar una lógica de trabajo y hacer un seguimiento en pasos para fusionar módulos futuros. Fue muy importante encontrar una herramienta como Melg, que ha facilitado el trabajo de comparación de archivos.

En futuros proyectos, queda abierta la posibilidad de integrar a la tabla Game, con las propuestas tablas del trabajo de Farrerons, como por ejemplo; Badge, Award, etc.

Rescato lo aprendido en el trascurso del postgrado, ya que las asignaturas cursadas anteriormente han sido de mucha utilidad para poder llevar a cabo este proyecto final con éxito.

Mi experiencia con el software libre resultó muy positiva, la decisión de cursar este máster estuvo motivada por el deseo de profundizar en su filosofía y aplicación, ya que profesionalmente venía utilizando diferentes distribuciones de Linux, tanto en el ambiente universitario como privado.

Referencias bibliográficas

- [1] «A Learning Theory for the Digital Age». [En línea]. Disponible en: http://www.itdl.org/Journal/Jan_05/article01.htm. [Accedido: 28-ene-2015].
- [2] «Abstracts_ES3.pdf». .
- [3] «Aparador d'innovació - L'espai on difondre els projectes d'innovació de la UOC - Programa d'Innovació de la UOC». [En línea]. Disponible en: http://www.innovauoc.org/showcase/index.php?content=load_proyecto&id=112. [Accedido: 27-ene-2015].
- [4] «LAMP (software bundle) - Wikipedia, the free encyclopedia». .
- [5] «Introducción a Servicios REST - Arquitectura Java». [En línea]. Disponible en: <http://www.arquitecturajava.com/servicios-rest/>. [Accedido: 29-ene-2015].
- [6] «Spring Framework 4.1 GA is here!». [En línea]. Disponible en: <https://spring.io/blog/2014/09/04/spring-framework-4-1-ga-is-here>. [Accedido: 29-ene-2015].
- [7] «Home - Maven User - Codehaus». [En línea]. Disponible en: <http://docs.codehaus.org/display/MAVENUSER/Home>. [Accedido: 29-ene-2015].
- [8] «memoria PFC Sergio Melgar.pdf». .
- [9] «jfarreronsTFM0114memoria.pdf». .
- [10] «Meld». [En línea]. Disponible en: <http://meldmerge.org/>. [Accedido: 17-abr-2015].
- [11] «Eclipse - The Eclipse Foundation open source community website.» [En línea]. Disponible en: <https://eclipse.org/>. [Accedido: 03-jun-2015].
- [12] «Maven – Welcome to Apache Maven». [En línea]. Disponible en: <https://maven.apache.org/>. [Accedido: 03-jun-2015].
- [13] «WampServer, the web development platform on Windows - Apache, MySQL, PHP». [En línea]. Disponible en: <http://www.wampserver.com/en/>. [Accedido: 03-jun-2015].
- [14] «MySQL :: MySQL Workbench». [En línea]. Disponible en: <https://www.mysql.com/products/workbench/>. [Accedido: 03-jun-2015].
- [15] «GitHub». [En línea]. Disponible en: <https://github.com/>. [Accedido: 03-jun-2015].