



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



**TFG del Grado en Ingeniería
Informática**

UBUMonitor

**Monitorización de alumnos
en la plataforma Moodle.**



Presentado por Yi Peng Ji
en Universidad de Burgos — 25 de junio
de 2019

Tutor: Dr.D. Raúl Marticorena Sánchez
y Dr.D. Carlos Pardo Aguilar

Índice general

| | |
|---|-----------|
| Índice general | I |
| Índice de figuras | III |
| Índice de tablas | IV |
| Apéndice A Plan de Proyecto Software | 1 |
| A.1. Introducción | 1 |
| A.2. Planificación temporal | 1 |
| A.3. Estudio de viabilidad | 5 |
| Apéndice B Especificación de Requisitos | 9 |
| B.1. Introducción | 9 |
| B.2. Objetivos generales | 9 |
| B.3. Catalogo de requisitos | 9 |
| B.4. Especificación de requisitos | 9 |
| Apéndice C Especificación de diseño | 11 |
| C.1. Introducción | 11 |
| C.2. Diseño de datos | 11 |
| C.3. Diseño procedimental | 11 |
| C.4. Diseño arquitectónico | 11 |
| Apéndice D Documentación técnica de programación | 13 |
| D.1. Introducción | 13 |
| D.2. Estructura de directorios | 13 |
| D.3. Manual del programador | 15 |

| | |
|--|-----------|
| D.4. Compilación, instalación y ejecución del proyecto | 19 |
| D.5. Pruebas del sistema | 19 |
| Apéndice E Documentación de usuario | 21 |
| E.1. Introducción | 21 |
| E.2. Requisitos de usuarios | 21 |
| E.3. Instalación | 21 |
| E.4. Manual del usuario | 21 |
| E.5. Visualización de gráficas | 24 |
| Bibliografía | 35 |

Índice de figuras

| | |
|--|----|
| D.1. Menú Help y seleccionar New Software... | 16 |
| D.2. Instalación de la herramienta. | 17 |
| D.3. Habilitar servicios web. | 18 |
| D.4. Habilitar servicios web parte 2. | 19 |
| | |
| E.1. Primera pantalla de inicio de sesión. | 22 |
| E.2. Pantalla de selección de curso | 23 |
| E.3. Ventana emergente pidiendo la contraseña antigua. | 24 |
| E.4. Pantalla inicial. | 25 |
| E.5. Listado de usuario matriculados en el curso. | 26 |
| E.6. Listado de calificaciones. | 27 |
| E.7. Gráfico de líneas de las calificaciones. | 28 |
| E.8. Gráfico radar de las calificaciones. | 29 |
| E.9. Gráfico de BoxPlot general de las calificaciones. | 29 |
| E.10. Gráfico BoxPlot de grupo de las calificaciones. | 30 |
| E.11. Tabla de calificaciones de tres personas y las medias del grupo y general. | 30 |
| E.12. Opciones para las gráfica de registros. | 31 |
| E.13. Listado de componentes ordenados alfabéticamente. | 32 |
| E.14. Listado de componente y evento ordenados alfabéticamente por componente. | 33 |
| E.15. Gráfica de barras apiladas con dos participantes y tres compo- nentes agrupados por día de la semana. | 34 |

Índice de tablas

| | |
|---|---|
| A.1. Sueldos anuales, mensuales y de 5 meses. | 5 |
| A.2. Dependencias de Java. | 6 |
| A.3. Dependencias de JavaScript. | 6 |

Apéndice A

Plan de Proyecto Software

A.1. Introducción

En este apartado se comentará la planificación temporal de proyecto junto con su viabilidad económica y legal.

A.2. Planificación temporal

El desarrollo de la aplicación esta planteado usando Scrum como metodología de gestión del proyecto. Aunque cabe destacar que está adaptado para trabajar con una persona a diferencia de los 4-8 usuales y también con reuniones semanales en vez de diarias. A continuación se muestran las características que se han seguido de la filosofía ágil:

- Desarrollo iterativo o incremental mediante interacciones (*sprints*).
- La duración de los *sprints* han sido de uno a cuatro semanas y cada una de ellas contiene las tareas o *issues* realizadas.
- Reuniones semanales de más de una hora comentando el avance del desarrollo y presentando partes del producto final.
- También se planifica las siguientes tareas a realizar durante las reuniones.

Sprint 0 (14/01/2019 - 08/02/2019)

Durante la primera semana se buscó reuniones con tres docentes de la Universidad (José Manuel Galán Ordax, Bruno Baruque Zanón y Raúl Marticorena Sánchez) sobre posibles proyectos que tuvieran en mente.

Después de hablar con los tres en esa semana, hubo varios de reflexión sobre la difícil elección del trabajo. Finalmente se elige el proyecto de Raúl Marticorena.

En la siguiente semana se prepara la primera reunión comentando en profundidad los detalles del trabajo y los objetivos finales que se espera.

En este *sprint* también se prepara las instalaciones de las herramientas de necesarias para el desarrollo. Se ha usado el manual del programador aportado por el autor de UBUGrades 2.0 como medida de apoyo.

Sprint 1 (08/02/2019 - 15/02/2019)

Buscar la manera de guardar los datos extraídos del servicio web en ficheros locales cifrados usando la contraseña de acceso a Moodle como clave. Se decide usar el algoritmo de encriptación **Blowfish** [9] que permite claves de 32 hasta 448 bits. Cabe destacar que se han descartado los algoritmos de cifrado que usan tamaños específicos de claves. Un ejemplo de ellos es el algoritmo *Advanced Encryption Standard* (AES) [7] que necesita claves de cifrado de 128, 192 o 256 bits.

Se acuerda empezar de nuevo el trabajo de UBULogs al no funcionar para la nueva versión de Moodle de la Universidad usando una alternativa diferente.

También se arregla un error del anterior proyecto el cual trataba al nombre de usuario como si fuera un correo.

Sprint 2 (15/02/2019 - 13/03/2019)

En este *sprint* se busca la forma de iniciar sesión con el usuario y contraseña de Moodle y así poder descargar los registros del curso. Inicialmente se intentó usar solo el paquete **java.net** [3] para realizar las conexiones al no ser una de dependencia externa. Sin embargo los resultados fueron infructuosos y finalmente se empleó Jsoup [4], una librería muy famosa de web scraping.

Además se decide modificar completamente las clases encargadas de recoger información de los servicios web de Moodle.

Sprint 3 (13/03/2019 - 21/03/2019)

Los objetivos fueron arreglar los errores encontrados en el anterior *sprint* y cambiar el paquete model modificando las clases y añadiendo todos los tipos disponibles de actividades y recursos del curso.

Sprint 4 (21/03/2019 - 26/03/2019)

Centrado exclusivamente en recoger los datos de los servicios web y transformarlo en objetos de Java. La información que nos interesa son los usuarios matriculados, los contenidos (actividades y recursos), las calificaciones, los roles y grupos del curso.

Todo esto se extrae de los JSON que devuelve el servicio web de Moodle, se han creado métodos que recogen los campos más útiles del JSON.

Sprint 5 (26/03/2019 - 03/04/2019)

Buscado la forma de encontrar cómo actualizar los datos, tanto de las calificaciones como de los registros del curso.

En cuanto a los registros del curso se descubre una forma de descargar los datos por días a través de un parámetro de la URL. Los tiempos de actualización de datos baja considerablemente con este método al descargar registros parciales en vez de completos.

También se refactoriza códigos de anteriores *sprints* y **Carlos Pardo Aguilar** se incorpora como segundo tutor.

Se acuerda en qué tipo de gráfica debe mostrar los registros del curso, barras apiladas.

Sprint 6 (03/04/2019 - 26/04/2019)

Los objetivos son arreglar los errores de programación encontrados. También aparece un bug de los servicios web con la función **gradereport_user_get_grade_items** que devuelve un mensaje de error si la **Retroalimentación** está oculta en el calificador.

Después de varias propuestas se llega el consenso de usar la otra función del servicio web **gradereport_user_get_grades_table** mucho más incómoda de trabajar. Este inconveniente ha supuesto un retraso de una semana en el proyecto.

Se añadió un fichero de configuración para guardar los nombres de usuario y el servidor de Moodle en la pantalla de inicio de sesión.

Sprint 7 (26/04/2019 - 11/05/2019)

Durante este proceso se implementó la agrupación de los registros por usuarios, componente y distintos tipos de tiempo (días, meses, años, etc) y finalmente contándolos. Se calcula también la media de todos los participantes según el número de registros.

Con todos estos datos se crean métodos que transforma en formato específico de Chart.js.

Sprint 8 (11/05/2019 - 18/05/2019)

Durante este *sprint* se ha buscado una manera de añadir fácilmente nuevos tipos de registros según el componente y evento. Finalmente se ha creado varias herramientas en Python que busca los diferentes tipos que hay a partir de los registros en CSV y los guarda en un fichero JSON. Con este fichero se ha creado otra herramienta para generar las enumeraciones Java de los componentes y eventos junto con las traducciones. Todo esto se comenta con más detalles en el manual del programador (D.3).

Sprint 9 (18/05/2019 - 23/05/2019)

Los objetivos fueron empezar la documentación de la memoria y probar si se puede añadir la opción de agrupar por componente y evento en los registros. Para esto se creó una nueva rama con resultados satisfactorios.

También se añadieron las fotografías de los participantes del curso.

Sprint 10 (23/05/2019 - 08/06/2019)

En este *sprint* se añadieron varias mejoras de interacción con el usuario:

- Valores predefinido en el campo *host* de la pantalla de inicio.
- Mostrar la categoría del curso al en la pantalla de selección del curso.
- Añadir una opción de ajustar el valor máximo del eje Y en el gráfico de barras.

Sprint 11 (08/06/2019 - 22/06/2019)

Los objetivos fueron completar la memoria, documentación de los métodos de Java, arreglar varios errores y refactorización del código arreglando la gran mayoría de *code smells* detectados por Sonarqube y Codacy.

Sprint 12 (22/06/2019 - 30/06/2019)

Completar los apartados restantes del anexo y su revisión final.

A.3. Estudio de viabilidad**Viabilidad económica**

Desarrollado por una sola persona durante 5 meses. Calculando usando salarios medios brutos de 19.485€ para un programador junior [8]. Empleando la calculadora de sueldos netos de Cinco Días [5] nos da los resultados en costes anuales. Calculamos también el sueldo mensual y el coste en 5 meses:

| Concepto | Coste anual | Coste mensual | Coste 5 meses |
|-------------------------|--------------------|-------------------|-------------------|
| Sueldo bruto | 19.485,00 € | 1.623,75 € | 8.118,75 € |
| Retenciones por IRPF | 2.221,30 € | 185,11 € | 925,54 € |
| Cuotas a la Seg. Social | 1.237,30 € | 103,11 € | 515,54 € |
| Sueldo neto | 16.026,40 € | 1.335,53 € | 6.677,67 € |

Tabla A.1: Sueldos anuales, mensuales y de 5 meses.

En principio el producto no está pensado venderlo, la intención es que sea de código libre y gratuito. Los potenciales clientes serían las Universidades que usen Moodle, por lo que si podríamos cobrar por crear versiones específicas de la aplicación para cada una de las Universidades. Otra forma de ingresos sería a partir de donaciones de la comunidad.

Viabilidad legal

Se mantiene la licencia de MIT al ser una de las más permisivas para productos de código libre. Permite a la gente hacer casi todo lo que quieran con el proyecto, incluyendo crear y distribuir versiones de código cerrado.

En el desarrollo del proyecto se han empleado varias librerías externas, algunas de ellas nuevas y otras que estaban en el anterior proyecto se han quitado en este desarrollo.

En definitiva, se han usado varias librerías de Java, JavaScript y un paquete de iconos.

Dependencias de Java

En la siguiente tabla se muestran las dependencias de Java junto con su versión y licencia.

| Dependencia | Versión | Licencia |
|------------------------|----------|--------------|
| Apache Commons CSV | 1.6 | Apache-2.0 |
| Apache Commons Math | 3.6.1 | Apache-2.0 |
| Apache Log4j | 1.2.17 | Apache-2.0 |
| Gson | 2.8.5 | Apache-2.0 |
| JSON In Java | 20180813 | MIT |
| JSoup Java HTML Parser | 1.11.3 | MIT |
| SLF4J API Module | 1.7.26 | MIT |
| SLF4J LOG4J 12 Binding | 1.7.26 | MIT |
| ThreeTen Extra | 1.5.0 | BSD 3-Clause |

Tabla A.2: Dependencias de Java.

Dependencias de JavaScript

En esta otra tabla se muestra las dependencias de JavaScript. Cabe destacar que Google Charts no es de código libre, no permite descargar y usarlo en un fichero local sin conexión [2].

| Dependencia | Versión | Licencia |
|---------------|-------------|----------|
| Chart.js | 2.8.0 | MIT |
| color-hash | 1.0.3 | MIT |
| Google Charts | Desconocida | Privada |
| html2canvas | 0.4.1 | MIT |

Tabla A.3: Dependencias de JavaScript.

Paquete de iconos

Se ha usado el paquete de iconos Font Awesome [1] que también se emplea en Moodle.

Los iconos de Font Awesome están sacados de **fa2png** versión **4.7.0** y licencia **SIL OFL 1.1**.

Apéndice B

Especificación de Requisitos

- B.1. Introducción
- B.2. Objetivos generales
- B.3. Catalogo de requisitos
- B.4. Especificación de requisitos

Apéndice C

Especificación de diseño

- C.1. Introducción
- C.2. Diseño de datos
- C.3. Diseño procedimental
- C.4. Diseño arquitectónico

Apéndice D

Documentación técnica de programación

D.1. Introducción

En esta sección del anexo se explicarán cómo están organizados los paquetes del proyecto; un manual para el programador sobre las herramientas que debe instalar para seguir el desarrollo; su compilación, instalación y ejecución del proyecto y las pruebas realizadas en la aplicación.

D.2. Estructura de directorios

El repositorio del proyecto está distribuido de la siguiente forma:

- /doc/: los ficheros javadoc del proyecto.
- /latex/: los elementos necesarios para generar la memoria y el anexo junto con sus PDF.
- /lib/: librerías externas de Java.
- /lib/src: el código fuente de las librerías externas de Java.
- /lib/docs: los javadoc de las librerías externas de Java.
- /python/: varios archivos de utilidad para los registros del curso:

APÉNDICE D. DOCUMENTACIÓN TÉCNICA DE PROGRAMACIÓN

- **Componentes y eventos.json**: fichero dividido en tres niveles que son Componentes, Eventos y Descripciones de los registros del curso.
 - **Contains checker.ipynb**: funciones de utilidad que comprueba en las Descripciones de los registros si existe o no un carácter o varios.
 - **Generador de código componentes y eventos.ipynb**: funciones que generan código Java automáticamente a partir del fichero JSON mencionado.
 - **Rastreador de componentes y eventos.ipynb**: programa de utilidad que escanea los registros del curso en formato CSV y con eso crea un fichero JSON con todos los Componentes, Eventos y Descripciones encontrados.
-
- `/resources/`: recursos de apoyo para el programa.
 - `/resources/css/`: ficheros CSS que modifica la visualización y colores de la aplicación.
 - `/resources/graphics/`: ficheros HTML para dibujar las gráficas.
 - `/resources/graphics/lib`: librerías externas de JavaScript usado para las gráficas.
 - `/resources/img/`: todas las imágenes de la aplicación.
 - `/resources/messages/`: ficheros de internacionalización de textos.
 - `/resources/view/`: se encuentra los ficheros FXML.
 - `/src/`: código fuente del desarrollo.
 - `/src/controllers/`: paquete encargado del flujo de la ejecución.
 - `/src/controllers/datasets/`: dirige las generaciones de los datos en formato válido para Chart.js.
 - `/src/controllers/ubugrades/`: encargado de la generación de los cursos, usuarios matriculados, calificaciones, etc. Toda la parte de los servicios web.
 - `/src/controllers/ubulogs/`: gestiona la descarga de los registros y su posterior *parseo*.

- `/src/controllers/ubulogs/logtypes/`: las clases encargadas de gestionar los ids de las Descripciones del registro.
- `/src/model/`: representación de los datos, su lógica de negocio.
- `/src/model/mod`: todas las clases de los tipos de módulos del curso.
- `/src/persistence/`: encriptación y desencriptación de archivos de objetos Serializados.
- `/src/webservice/**`: paquete y subpaquetes encargados de llamar a las funciones de Moodle y recibir las respuestas.

D.3. Manual del programador

Este manual sirve de referencia para personas que tomen el proyecto en el futuro. Se explicarán como construir el entorno de desarrollo, qué es lo que necesitan instalar.

Instalación de Java

La aplicación está desarrollada en Java 8, por lo tanto para trabajar en ella necesitará tener el JDK 8. Durante el proyecto se ha trabajado con la **versión 1.8.0_201** pero sería recomendable trabajar con la última disponible que se encuentra en:

<https://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

Instalación de Eclipse

Durante el desarrollo se ha empleado la versión **Oxygen.1a Release (4.7.1a)** de Eclipse aunque no habría muchos problemas usar una versión más reciente. Se puede descargar en:

<https://www.eclipse.org/downloads/>

Instalación de e(fx)clipse

e(fx)clipse es una herramienta adicional de Eclipse que facilita el desarrollo de aplicaciones que usen JavaFX. Para instalar e(fx)clipse realizamos los siguientes pasos:

APÉNDICE D. DOCUMENTACIÓN TÉCNICA DE PROGRAMACIÓN

1. Seleccionamos el menú **Help** de la barra de herramientas de Eclipse.
2. Seleccionamos el elemento del menú **Install New Software...**

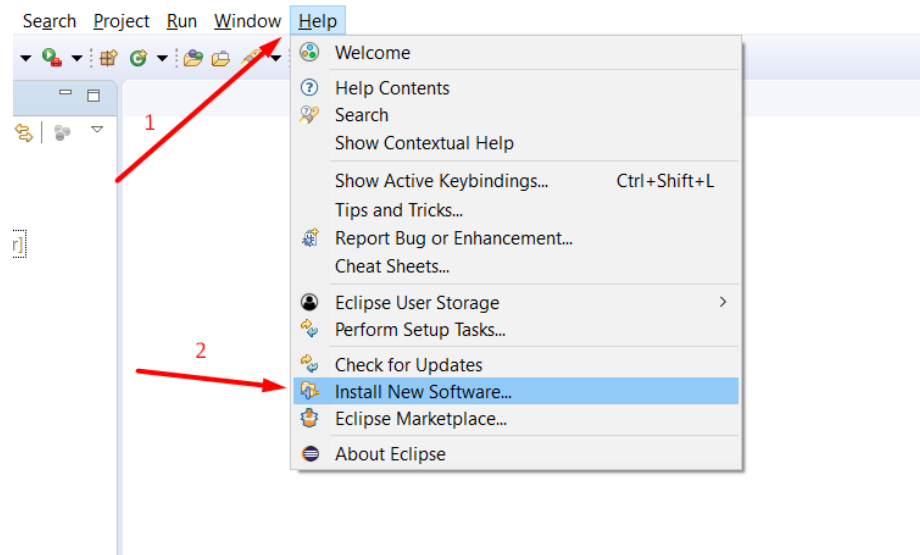


Figura D.1: Menú **Help** y seleccionar **Install New Software...**

3. Seleccionamos la opción **–All Available Sites–** de la lista desplegable de **Work with:**.
4. Escribimos **e(fx)clipse** en el cuadro de texto.
5. Marcamos solo la casilla que pone **e(fx)clipse - IDE**.
6. Pulsamos el botón **Next** y debería descargar e instalarse.

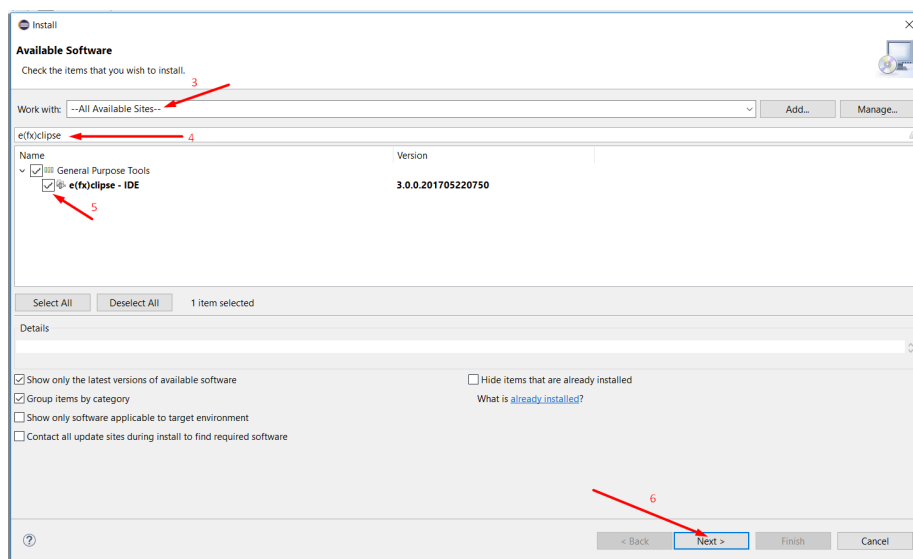


Figura D.2: Instalación de la herramienta.

Instalación de Scene Builder

La herramienta del Scene Builder (2.0) se puede conseguir desde:

<http://www.oracle.com/technetwork/java/javase/downloads/javafxscenebuilder-1x-archive-2199384.html>

Instalación de Moodle

Para instalar Moodle en el equipo personal, se puede descargar desde la página oficial con todo incluido para su funcionamiento:

<https://download.moodle.org/windows/>

Una vez descargado y descomprimido la carpeta, ejecutamos **StartMoodle.exe** que inicia la arranque del servidor.

Para acceder al sitio web de Moodle se usa el localhost: <http://localhost>. En la página web, la primera vez iniciará el proceso de instalación del servidor. Se sigue las instrucciones del instalador.

Por defecto no están activados los servicios web de Moodle. Para habilitarlo seguimos los siguientes pasos después de iniciar sesión con la cuenta del administrador:

1. Pulsamos el botón **Administración del sitio**.

APÉNDICE D. DOCUMENTACIÓN TÉCNICA DE PROGRAMACIÓN

2. Entramos dentro de **Mobile Settings**.

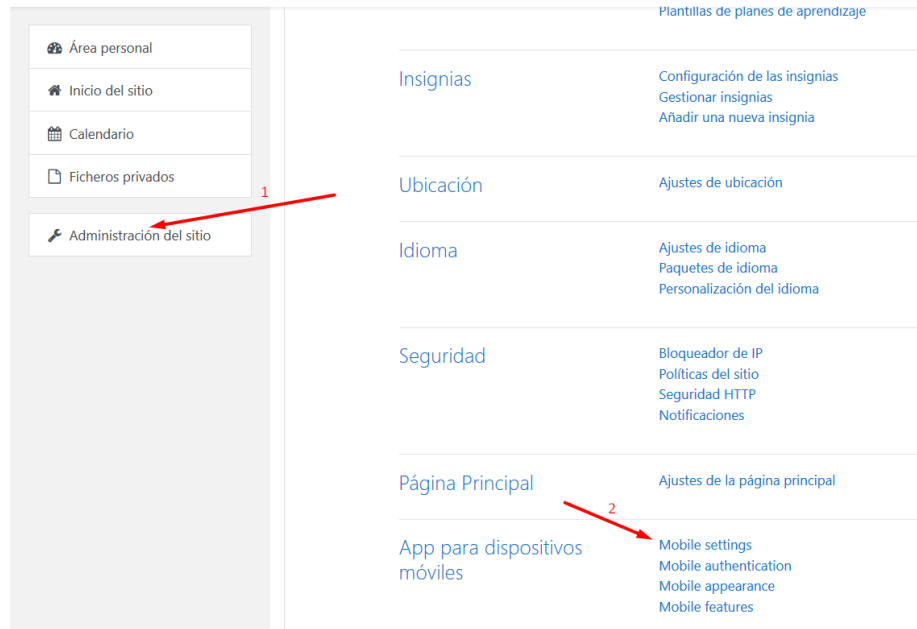


Figura D.3: Habilitar servicios web.

3. Marcamos la casilla de **Habilitar servicio web para dispositivos móviles**.

4. Pulsamos en **Guardar cambios**

D.4. COMPILACIÓN, INSTALACIÓN Y EJECUCIÓN DEL PROYECTO

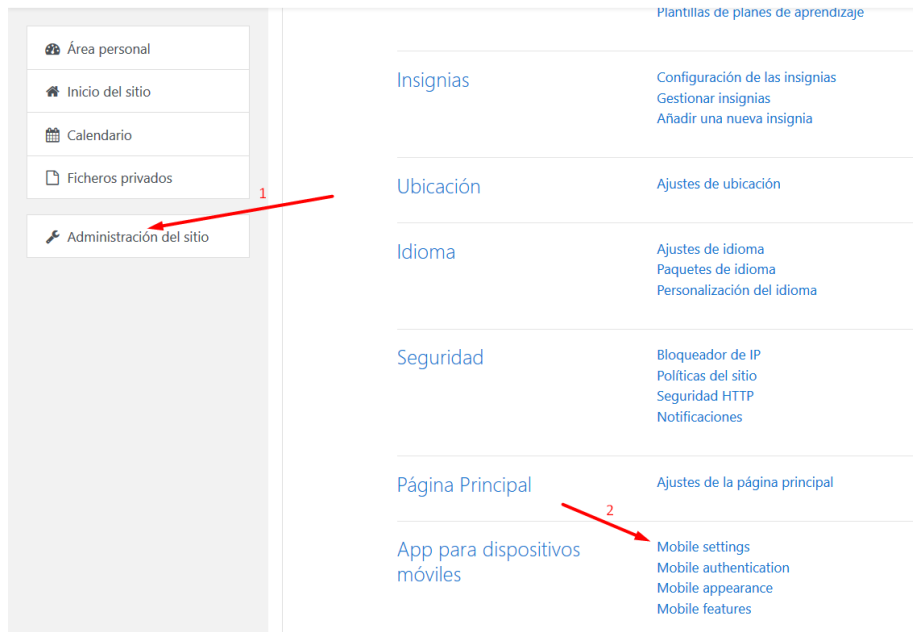


Figura D.4: Habilitar servicios web parte 2.

Instalación de Jupyter Notebook

Según la página oficial de Jupyter Notebook [6], lo más recomendable es instalarlo a través de Anaconda en:

<https://www.anaconda.com/distribution/#download-section>.

La versión recomendable que se debe descargar es el que incluye Python 3.7 o superior debido a que los diccionarios están ordenados por orden de inserción.

D.4. Compilación, instalación y ejecución del proyecto

Lo primero de todo, hay que descargar todo el ficheros fuentes.

D.5. Pruebas del sistema

Apéndice E

Documentación de usuario

E.1. Introducción

En este manual se explica los requerimientos de la aplicación, su ejecución en el ordenador e instrucciones de cómo usarlo correctamente.

E.2. Requisitos de usuarios

- Tener instalado **Java 8**.
- Obtener el jar ejecutable de la última versión disponible en:
<https://github.com/yjx0003/UBUMonitor/releases/latest>
- Debe disponer conexión a Internet.
- Una cuenta de Moodle con acceso a asignaturas con rol de Profesor.

E.3. Instalación

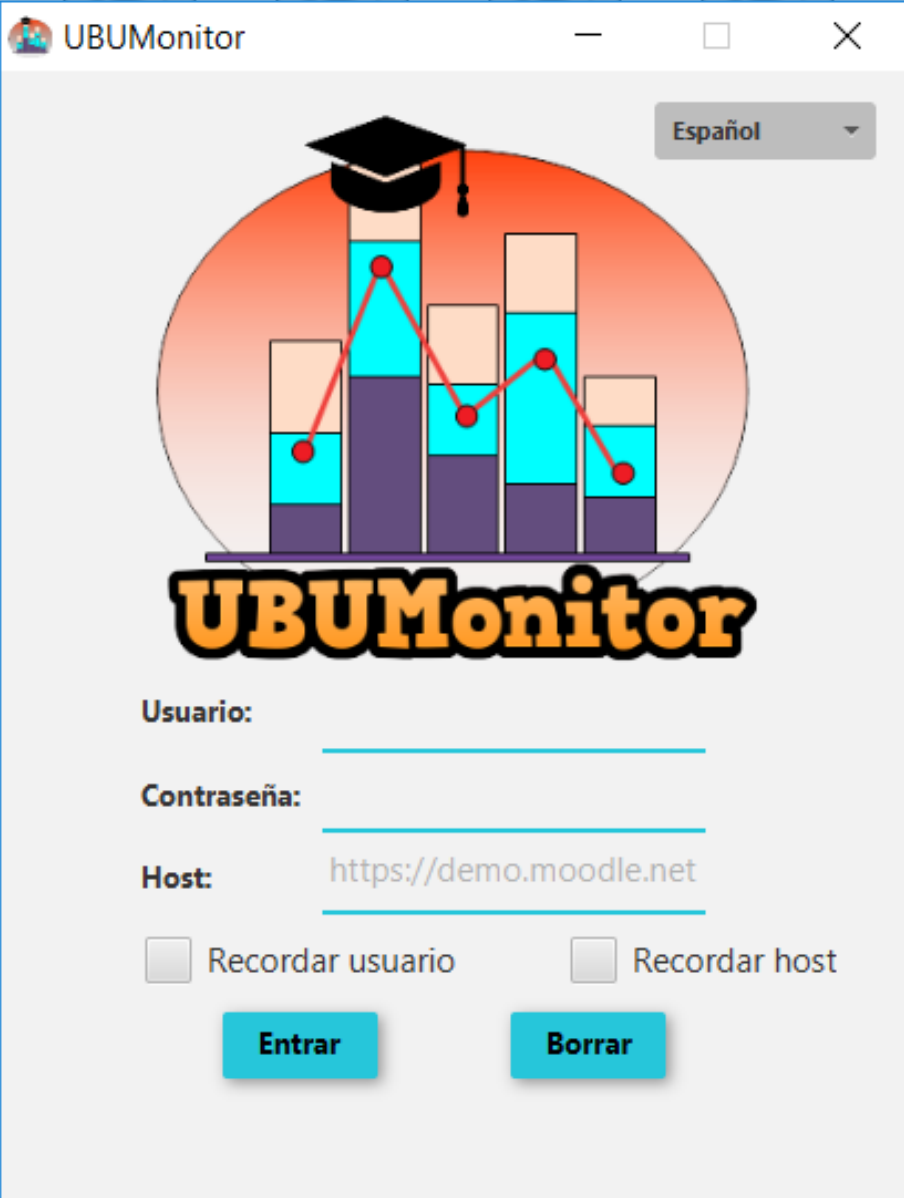
La aplicación no necesita instalar nada más a parte de poseer Java 8.

E.4. Manual del usuario

Después de ejecutar el jar previamente descargado, inicialmente nos mostrará una pantalla de inicio. Este manual de usuario también se encuentra en:

Inicio de sesión

En esta primera pantalla es la de inicio de sesión. Ahí introducimos las credenciales de usuario, contraseña y en el campos host la URL de la página web de la plataforma Moodle.



UBUMonitor

Español

UBUMonitor

Usuario: _____

Contraseña: _____

Host: https://demo.moodle.net

☐ Recordar usuario ☐ Recordar host

Entrar **Borrar**

Figura E.1: Primera pantalla de inicio de sesión.

Da la opción de recordar el usuario y host para los siguientes usos de la

aplicación. Esa información se guarda en un fichero **config.properties** si marcamos una o ambas opciones.

En la esquina superior derecha nos permite cambiar el idioma de la aplicación y el botón **Borrar** para eliminar el texto de los tres campos.

Después de introducir las credenciales pulsamos el botón de **Entrar**. Si los datos son válidos cargará la siguiente pantalla y si no mostrará un mensaje de error.

Elección del curso

En esta pantalla muestra un listado de cursos que está matriculado el usuario y permite seleccionar uno de ellos y acceder a la siguiente pantalla. Cuando se elige uno de los curso cambia la información de fecha de la última actualización en local. Esta fecha se muestra en función del fichero local que este guardado en el directorio **cache** con la opción de marcar o no la casilla de actualizar los datos. En caso contrario indica el texto **Nunca** y se deshabilita marca la casilla.



Figura E.2: Pantalla de selección de curso

Cuando se haya seleccionado un curso y pulsamos en el botón **Entrar**, si la casilla de actualizar los datos está seleccionado, carga fichero en local o crea en caso de que no exista y actualiza los datos del curso (usuarios matriculados, calificaciones, registros, etc...). El fichero local se **cifra** usando la contraseña de Moodle como clave. Este proceso puede tardar varios minutos si los registros son muy grandes. Si no está marcado descifra el fichero y lo carga de forma instantánea.

Carga la siguiente después de finalizar cualquiera de los dos procesos.

Contraseña modificada

En caso de que se haya modificado la contraseña de Moodle, muestra una ventana emergente de que se ha detectado un cambio en la contraseña y que pide la anterior que tuvieras en la fecha señalada.

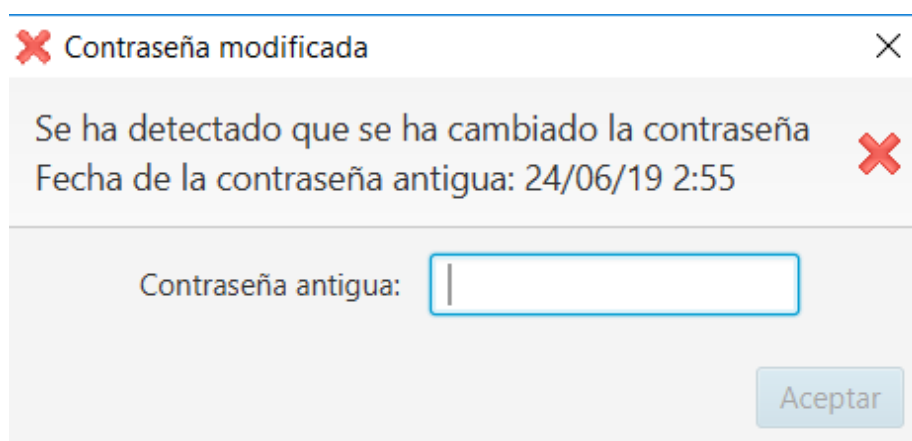
Una ventana emergente con un título que dice 'Contraseña modificada' precedido por un icono de 'X' roja. El cuerpo de la ventana contiene el mensaje 'Se ha detectado que se ha cambiado la contraseña' y 'Fecha de la contraseña antigua: 24/06/19 2:55', también precedido por un icono de 'X' roja. Debajo de este mensaje hay un campo de texto etiquetado 'Contraseña antigua:' con un cursor visible. En la esquina inferior derecha hay un botón gris con el texto 'Aceptar'.

Figura E.3: Ventana emergente pidiendo la contraseña antigua.

Cuando se haya introducido correctamente la contraseña antigua, el fichero en cache local se guarda con la nueva.

E.5. Visualización de gráficas

Ventana principal de visualización de gráficas del calificador y de los registros.

En esta ventana muestra los diferentes menús existentes, la lista de usuarios matriculados y las elecciones de diferentes gráficos.



Figura E.4: Pantalla inicial.

Barra de herramientas

En la parte superior de la ventana muestra una barra de herramientas con diferentes opciones:

■ Archivo

- Cambiar asignatura: vuelve a la ventana de Elección del curso
- Guardar gráfico como...: Permite guardar en formato PNG el gráfico actual
- mostrado en la aplicación.
- Guardar todo como...: Permite guardar en formato HTML todos los gráficos.
- Cerrar sesión: Cierra la sesión y vuelve a la pantalla de inicio.
- Salir: Cierra la aplicación.

■ Editar

- Borrar selección: Quita la selección de todos los listados disponibles.

■ Ayuda

- Acerca de la aplicación: Abre en la ventana del navegador el proyecto de GitHub.

También muestra los datos del usuario actual, el curso seleccionado y la página de la plataforma de Moodle (host).

Listado de usuarios matriculados

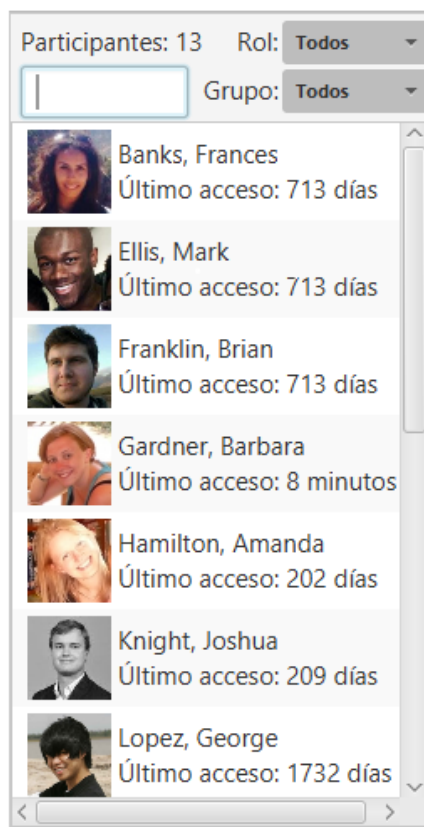


Figura E.5: Listado de usuario matriculados en el curso.

En este apartado se muestra el listado de los usuarios matriculados en el curso junto con su fotografía y el tiempo desde su último acceso a la plataforma de Moodle ya sea en minutos horas o días. Se toma como referencia la hora del sistema. Es de selección múltiple, elegir varios usuarios mientras se mantiene pulsando la tecla Control.

También da la opción de aplicar tres filtros juntos de forma conjunta:

- Campo de texto: filtrar el listado por nombre y/o apellidos.
- Selector de rol: seleccionar el rol por el que se quiere filtrar.
- Selector de grupo: grupo por el que se quiere filtrar.

Gráficas de calificaciones

▪ Listado de elementos del calificador (Grade Items)

- Listado de elementos del calificador (Grade Items).

En la pestaña de calificaciones muestra la jerarquía del calificador junto con sus filtros. Es de selección múltiple. En cualquier momento se puede desplegar y hacer más grande esta zona para ver mejor los nombres.

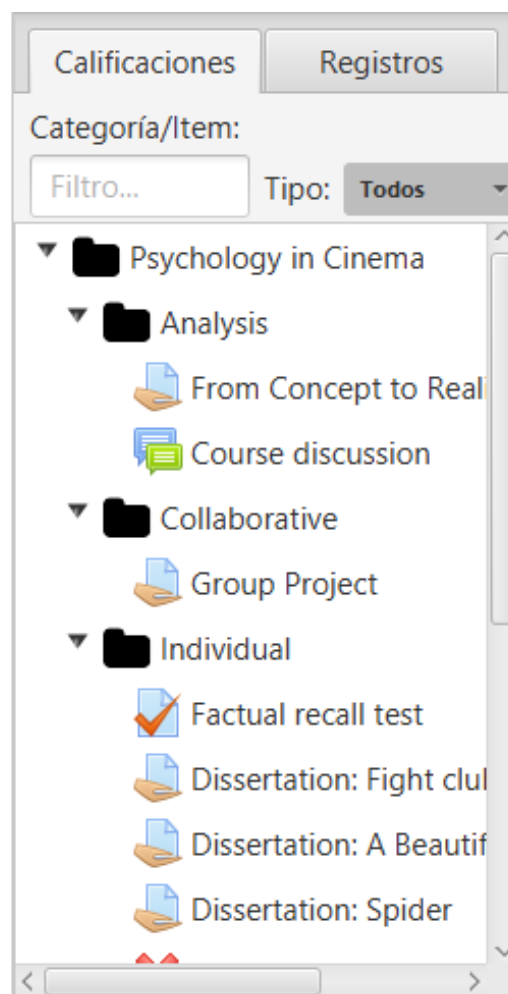


Figura E.6: Listado de calificaciones.

Existe dos tipos de filtros que se usan de modo conjunto:

- Campo de texto: por nombre del elemento del calificador.

- o Selector de tipo: según el tipo que sea (foro, tarea, cuestionario, et...)

También tiene varias pestañas de ocultar la leyenda, media general y media del grupo.

■ Gráfico de líneas

Muestra un gráfico de líneas donde se representan las notas de los alumnos para cada elemento del calificador seleccionado.

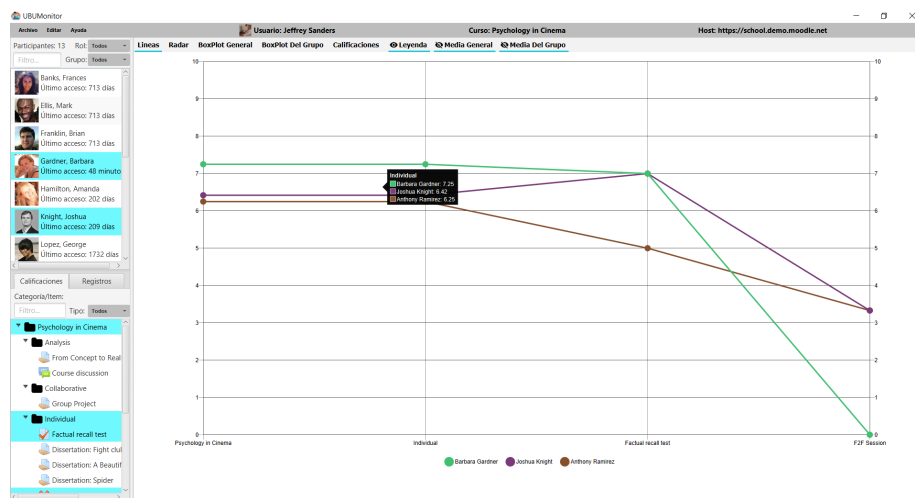


Figura E.7: Gráfico de líneas de las calificaciones.

■ Gráfico radar

Gráfico de tipo radar o también conocido como diagrama de araña. Muy útil para comparar dos participantes del curso.

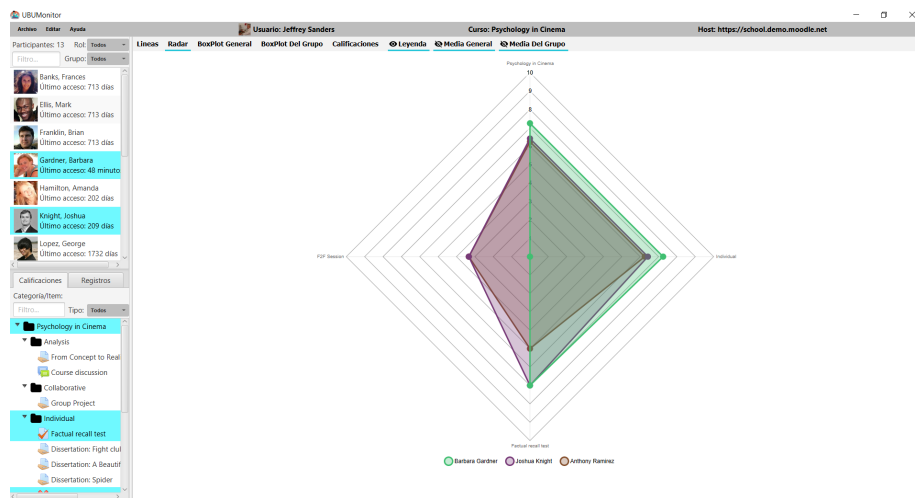


Figura E.8: Gráfico radar de las calificaciones.

■ Gráfico BoxPlot general

Muestra la información de un boxplot o diagrama de caja y bigotes mediante un gráfico de líneas. En este gráfico representa también los máximos, mínimos, la mediana así como los cuartiles primero y tercero.

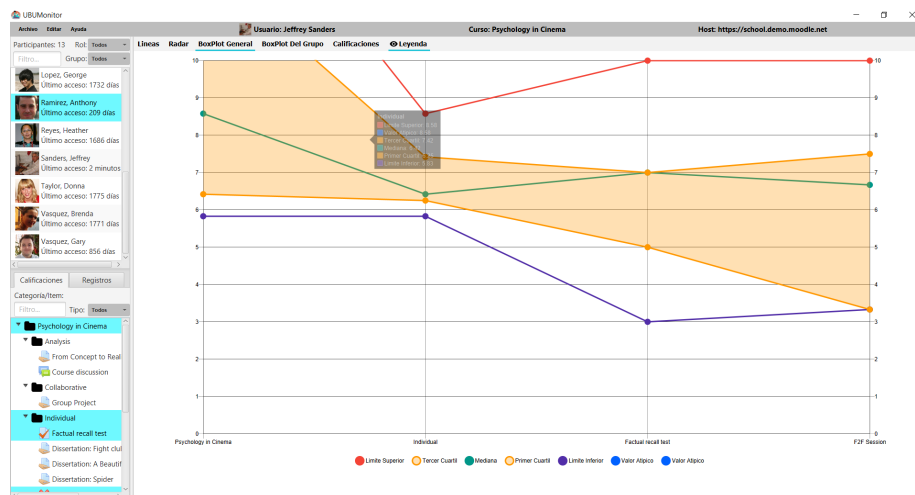


Figura E.9: Gráfico de BoxPlot general de las calificaciones.

■ Gráfico BoxPlot grupo

Muestra la misma información que el gráfico anterior pero para un grupo concreto. Para que este gráfico sea visible debemos seleccionar primero un grupo.

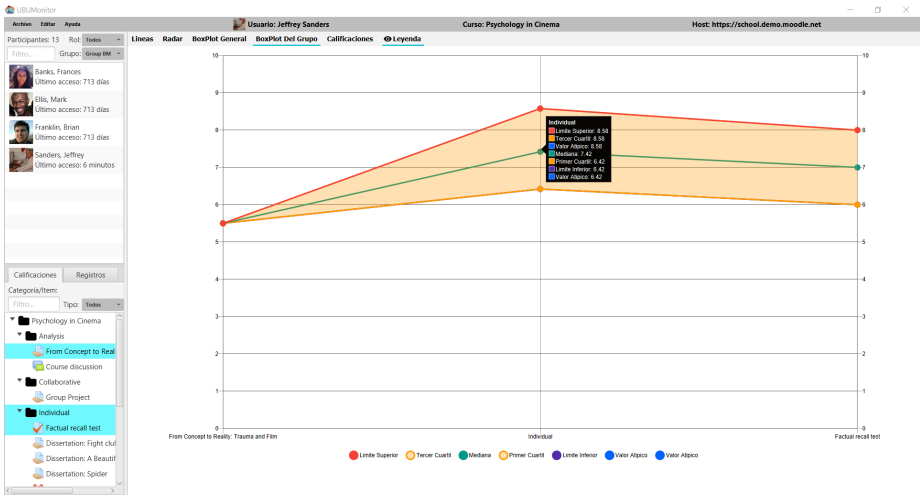


Figura E.10: Gráfico BoxPlot de grupo de las calificaciones.

■ **Tabla de calificaciones**

Muestra una tabla con las calificaciones así como la media general y la media de cada uno de los grupos del curso.

| Lineas | Radar | BoxPlot General | BoxPlot Del Grupo | Calificaciones |
|----------------------|-------|-----------------|-------------------|--|
| | | | | Notas |
| | | | | From Concept to Reality: Trauma and Film |
| | | | | Individual |
| | | | | Factual recall test |
| Mark Ellis | | | | 55.0/100.0 |
| Brian Franklin | | | | 77.0/120.0 |
| Barbara Cardiner | | | | 103.0/120.0 |
| Media | | | | 87.0/120.0 |
| Media Grupo Group FC | | | | 6.89/10 |
| Media Grupo Group IM | | | | 6.55/10 |
| Media Grupo Group S | | | | 7.17/10 |

Figura E.11: Tabla de calificaciones de tres personas y las medias del grupo y general.

Gráficas de los registros

■ **Opciones de los registros**

Al pulsar la pestaña de Registros aparece varias opciones en la pantalla inferior.

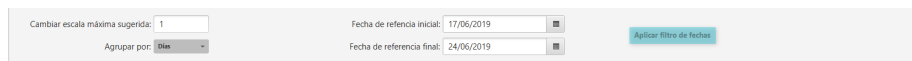
The image shows a horizontal filter bar with a light gray background. On the left, there is a text label 'Cambiar escala máxima sugerida:' followed by a text input field containing the number '1'. Below this is a label 'Agrupar por:' followed by a dropdown menu showing 'Día'. In the center, there are two date pickers. The first is labeled 'Fecha de referencia inicial:' and shows '17/06/2019'. The second is labeled 'Fecha de referencia final:' and shows '24/06/2019'. On the right side of the bar is a blue button with white text that says 'Aplicar filtro de fechas'.

Figura E.12: Opciones para las gráfica de registros.

- Cambiar escala máxima sugerida: permite modificar la escala máxima del eje Y. Se actualiza los valores al cambiar los componentes y/o eventos seleccionados o las otras opciones. Aporta información útil también al mostrar que existe un usuario con esa escala.
- Agrupar por: agrupa los registros en diferentes formas de tiempo (día, mes, año, etc...).
- Fecha de referencia inicial: la fecha de inicio orientativo que se muestra al agrupar.
- Fecha de referencia final: fecha límite orientativo que se muestra al agrupar.

Hay que tener en cuenta que la fecha de referencia inicial y final no toman todos los datos al realizar las agrupaciones. Por ejemplo al agrupar por meses y una fecha inicial del 17/06/2019 a 24/06/2019 no mostrará los registros entre esos dos días si no que mostrara todas las de junio. También las agrupaciones por Hora y Día de la semana no usa el selector de fechas, por lo tanto se deshabilitan.

■ **Componente**

Lista los registros del curso en función del Componente que sea. Solo se visualizan componentes que hayan generado los usuarios matriculados en el curso actualmente.

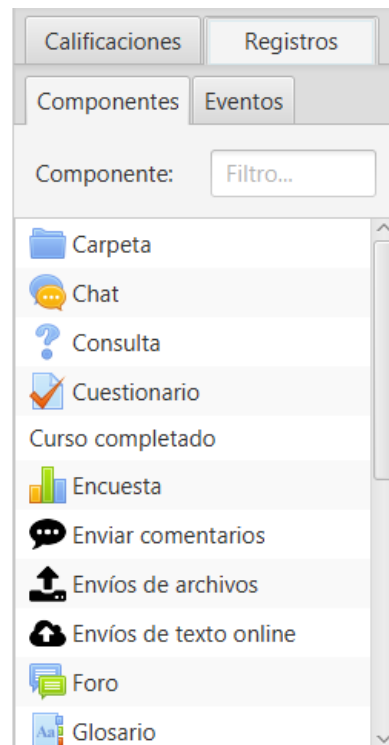


Figura E.13: Listado de componentes ordenados alfabéticamente.

■ Evento

Lista los registros del curso en función del Componente y Evento que sea. Se muestran solo los componentes y eventos generados por los usuarios actuales matriculados en el curso.

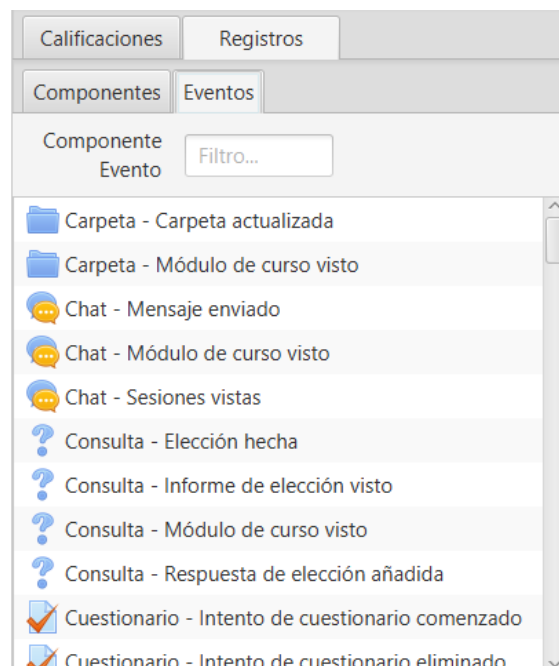


Figura E.14: Listado de componente y evento ordenados alfabéticamente por componente.

■ Gráfica de barras apiladas

Para ambos tipos de subpestañas (Componentes y Eventos) la gráfica que se usa para mostrar los registros del curso son barras apiladas. Muestra en barras cada uno de los usuarios seleccionados con los componentes y/o eventos apilados con diferentes colores cada uno. También se apilan las líneas que indican el valor medio de los usuarios filtrados en ese momento.

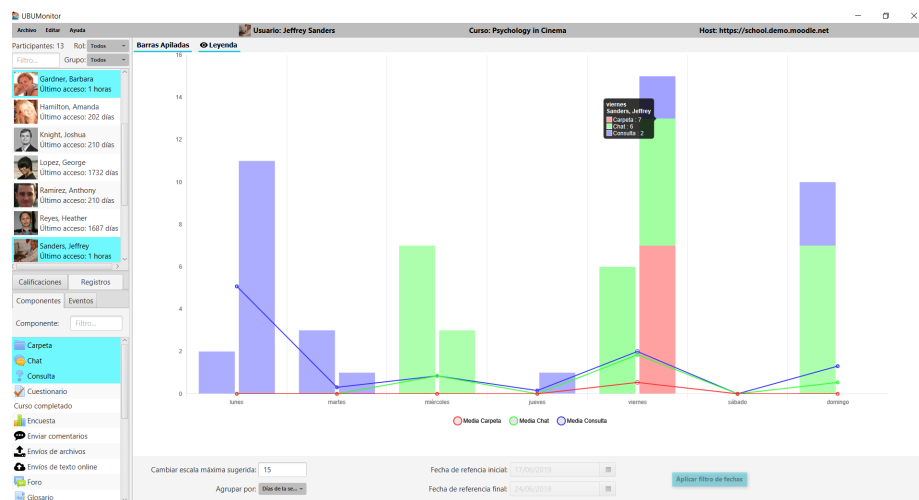


Figura E.15: Gráfica de barras apiladas con dos participantes y tres componentes agrupados por día de la semana.

Bibliografía

- [1] *Font Awesome*. URL: <https://fontawesome.com/> (visitado 23-06-2019).
- [2] *Frequently Asked Questions / Charts*. en. URL: <https://developers.google.com/chart/interactive/faq?cs=1#offline> (visitado 23-06-2019).
- [3] *java.net (Java Platform SE 8)*. URL: <https://docs.oracle.com/javase/8/docs/api/java/net/package-summary.html> (visitado 22-06-2019).
- [4] *jsoup Java HTML Parser, with best of DOM, CSS, and jquery*. URL: <https://jsoup.org/> (visitado 22-06-2019).
- [5] Ediciones El País. *Calculadora sueldo neto*. es. URL: <https://cincodias.elpais.com/herramientas/calculadora-sueldo-neto/> (visitado 23-06-2019).
- [6] *Project Jupyter*. URL: <https://www.jupyter.org> (visitado 25-06-2019).
- [7] Margaret Rouse. *What is Advanced Encryption Standard (AES)? - Definition from WhatIs.com*. en. Marzo de 2017. URL: <https://searchsecurity.techtarget.com/definition/Advanced-Encryption-Standard> (visitado 22-06-2019).
- [8] *Salarios para empleos de Programador/a junior en España / Indeed.es*. Junio de 2019. URL: <https://www.indeed.es/salaries/Programador/a-junior-Salaries?period=yearly> (visitado 23-06-2019).
- [9] *Schneier on Security: The Blowfish Encryption Algorithm*. URL: <https://www.schneier.com/academic/blowfish/> (visitado 18-06-2019).