

## Capítulo 7

# Manual de usuario

Las instrucciones de uso del sistema se detallan a continuación.

### 7.1. Introducción

EvalCourse es un sistema software que ayuda al docente en la difícil tarea de evaluar competencias genéricas del trabajo de los alumnos en un entorno LMS (Learning Management System). Las consultas son escritas en SASQL, un Lenguaje de Consulta de Apoyo a la Evaluación, y el trabajo de los alumnos se tomará de lo ocurrido en la plataforma virtual Moodle.

Medir las competencias genéricas es una tarea difícil; difícil en el sentido de que es poco escalable y que es subjetiva. Intentar pensar en cada uno de los alumnos que un profesor tiene y que para cada uno evalúe su capacidad de liderazgo, de trabajo en equipo, habilidades para gestionar el tiempo y organizarse, etc. Todo esto no es abarcable si se tiene que realizar manualmente y más aún si se tiene una clase con muchos alumnos, o incluso más de una clase. Además, siempre nos quedará la duda de si realmente estamos en lo cierto en nuestras suposiciones o si tenía mayor habilidad para alguna de las competencias mencionadas. Implicaría conocer muy bien a cada uno de tus alumnos para poder emitir un juicio de valor certero.

Con EvalCourse intentamos salvar esa barrera. El trabajo que los estudiantes realizan en la plataforma virtual de aprendizaje, Moodle en este caso, almacena mucha cantidad de información sobre todo lo que allí se hace, y lo mejor de todo es que se queda grabado en los registros de manera automática, a la misma vez que los alumnos interactúan con la plataforma.

Por todo ello, este software resulta de gran interés. En una consulta simple el profesor indica sobre qué indicadores desea obtener información, el sistema le da una respuesta a la petición con la información que se ha recogido de la plataforma. En ese momento la pelota cae en el tejado del docente, dejándole la labor de analizar los resultados para observar y comprobar qué ha hecho cada estudiante e interpretar los resultados según los parámetros que haya prefijado.

### 7.2. Características

Las principales funcionalidades del sistema son:

- Consulta a herramientas existentes en el LMS:

- Actividades.
  - Foros.
  - Talleres.
  - Campus.
  - Wikis
- Para cada herramienta podremos conocer:
    - Hitos.
    - Participación.
    - Interacción.
    - Accesos.
- Ficheros salida con la información pedida:
    - Fichero del desempeño de los estudiantes en formato *xml*.
    - Fichero del desempeño de los estudiantes en formato *csv*.
    - Gráfico del desempeño de los estudiantes en formato *Gephi*.
    - Gráfico del desempeño de los estudiantes en formato *html* (diagrama de barras, sectores o actividad temporal).

### 7.3. Requisitos previos

Requisitos software necesarios:

- Aplicación RCP.
- Pentaho. Sólo si se quiere editar alguna transformación, sin ella también obtendremos los informes.
- Gephi. Necesario para visualizar gráficos en formato *Gephi*.

### 7.4. Utilización

Lo primero que tendremos que hacer será elegir la procedencia del curso, es decir, si consultaremos usando la base de datos de Moodle o la copia de seguridad.

#### Configuración curso procedente de la base de datos

Para ello habrá que indicar en la consulta el id del curso del que se quiere obtener información y configurar el acceso a la base de datos. Este se consigue mediante la definición de los parámetros de conexión en el fichero *dbconexion.csv* 7.1, que se encuentra en la carpeta raíz de Eclipse. Un ejemplo de fichero se ve en el fragmento de código 7.2.

Listing 7.1: Esquema del archivo dbconexion.csv

```
1 name_of_the_connection:db_server:db_engine:db_mode:db_name:db_port:
2   db_username:encrypted_password
```

Listing 7.2: Ejemplo de fichero dbconexion.csv

```
1 mudel:localhost:MYSQL:Native:moodle:3306:
2   moodle:Encrypted 2e7fad9850be9c1869b2fad76e9b487ee
```

- *name\_of\_the\_connection*: Nombre de la conexión de la base de datos. El usuario del programa puede poner el nombre que quiera.
- *db\_server*: Nombre del servidor que contiene la base de datos. Podrá ser una dirección IP o localhost.
- *db\_engine*: Motor de base de datos. La Aplicación solo ha sido probada con el sistema MYSQL.
- *db\_mode*: Modo de conexión. Por defecto 'Native'.
- *db\_name*: Nombre de la base de datos que contiene la base de datos de Moodle.
- *db\_port*: Puerto por el que nos conectaremos a la base de datos.
- *db\_username*: Usuario de la base de datos.
- *encrypted\_password*: Encrypted + la clave encriptada.

## Configuración curso procedente de la copia de seguridad

Para realizar la copia de seguridad a un curso deberemos loguearnos en Moodle como en cualquier ocasión. En la barra lateral izquierda habrá una opción para realizar una copia de seguridad (figura 7.2). Si pinchamos ahí se nos abrirá una ventana donde deberemos ponerle un tick a aquellas opciones que queramos que nuestra copia de seguridad tenga, en concreto deberemos seleccionar además de las opciones ya marcadas por defecto la opción *Incluir archivos log de cursos* (figura 7.1) y alternativamente podremos hacer anónima la información de los usuarios si estamos realizando un estudio que será publicado y debemos salvaguardar los datos personales de los alumnos. Pinchamos en siguiente y dejaremos el resto de opciones por defecto tal y como vienen hasta poder descargar la copia de seguridad en nuestro equipo.

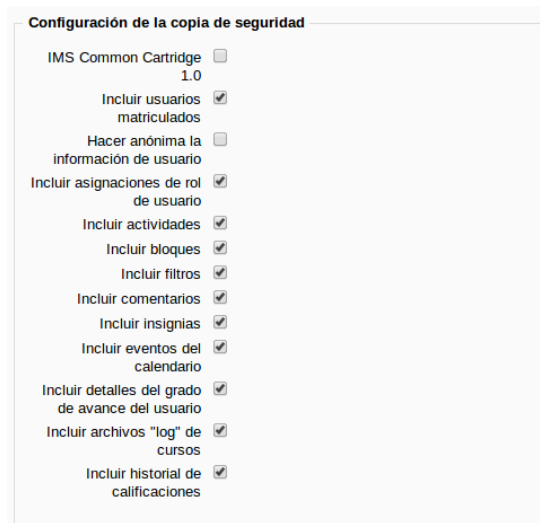


Figura 7.1: Parámetros a seleccionar.



Figura 7.2: Copia de seguridad.

Una vez tengamos el curso localizado de cualquiera de las dos formas, ejecutamos la aplicación Eclipse y creamos un proyecto EvalCourse. Para ello pinchamos en *New* → *Project* → *EvalCourse Project* (figura 7.3) y en la siguiente ventana, le ponemos un nombre (figura 7.4).

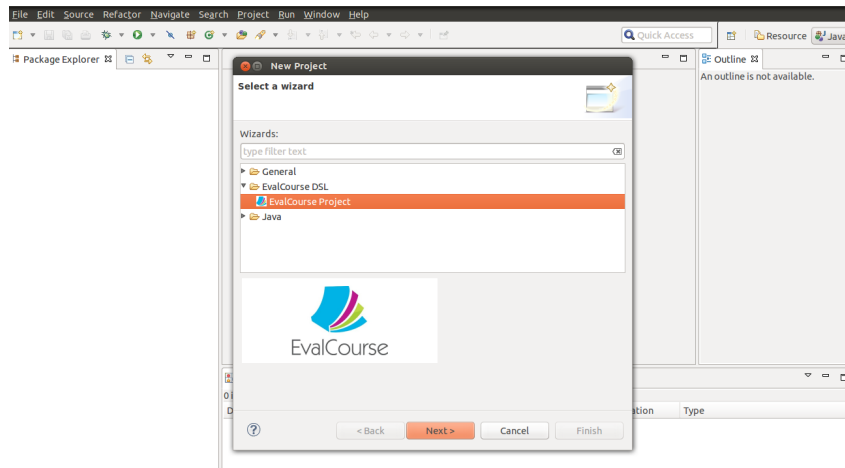


Figura 7.3: Crear proyecto EvalCourse.

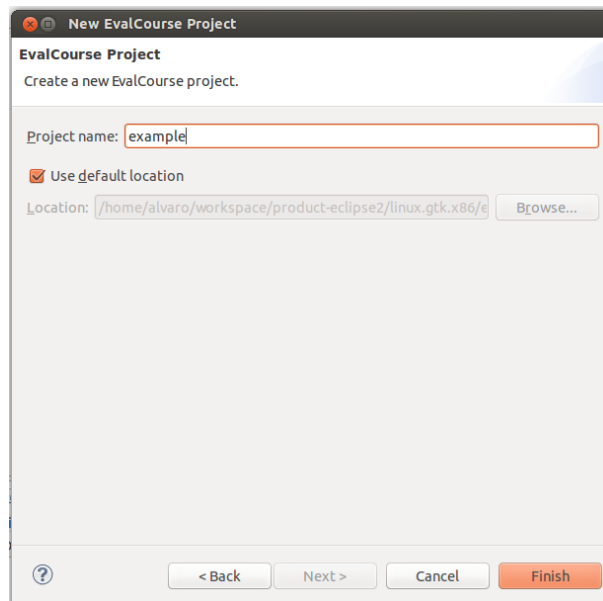


Figura 7.4: Nombrar proyecto.

El aspecto que observaremos en la estructura de directorios será (figura 7.5).

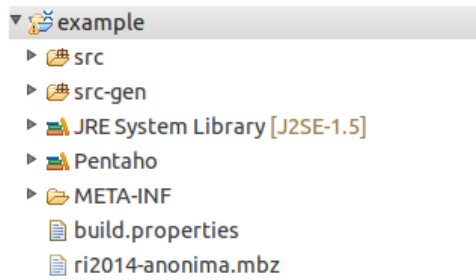


Figura 7.5: Proyecto vacío.

Abriremos el fichero *process.evc* dentro de la carpeta *src*, donde escribiremos nuestra consulta. Hasta que no la escribamos, el sistema nos marcará un error en el archivo, pues éste debe contener una consulta válida. Escribiremos, por tanto, nuestra consulta usando el patrón que se explica en 7.3 (figura 7.6). Si cometemos un fallo una vez hayamos escrito algo, el sistema nos informará debidamente del error (figura 7.7).

```

1 Evidence name_of_the_indicator :
2     get students
3     show milestones | participation | access | interaction
4     in assignment | workshop | forum | campus | wiki [list of ids]
5     from course id\_course | backup.

```

Listing 7.3: Sintáxis del lenguaje y palabras reservadas

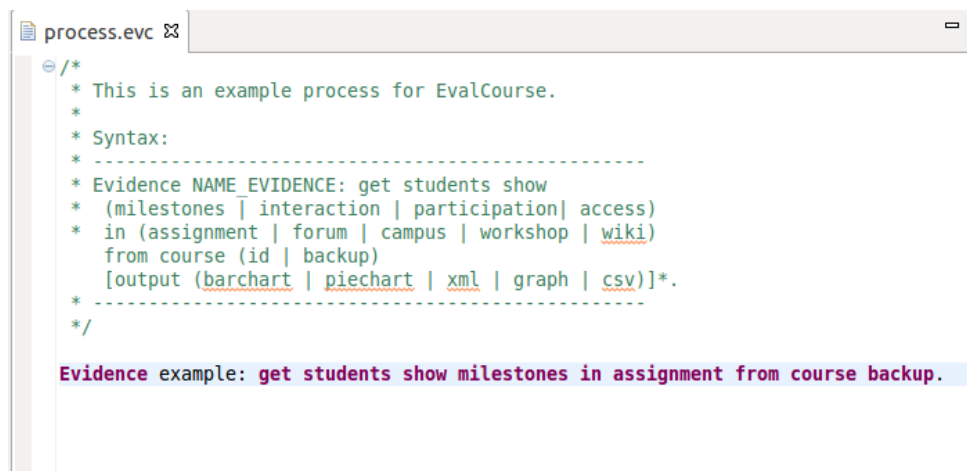


Figura 7.6: Escribir consulta.

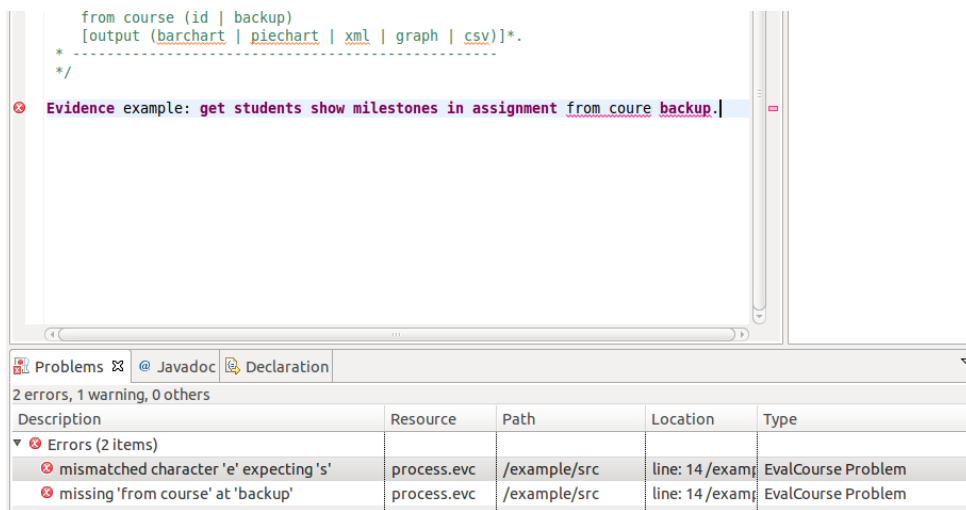


Figura 7.7: Aspecto de la consulta tras informar de un error.

Una vez escrita la consulta sólo tendremos que ejecutar el proyecto pulsando en Run, como se observa en (figura 7.8).

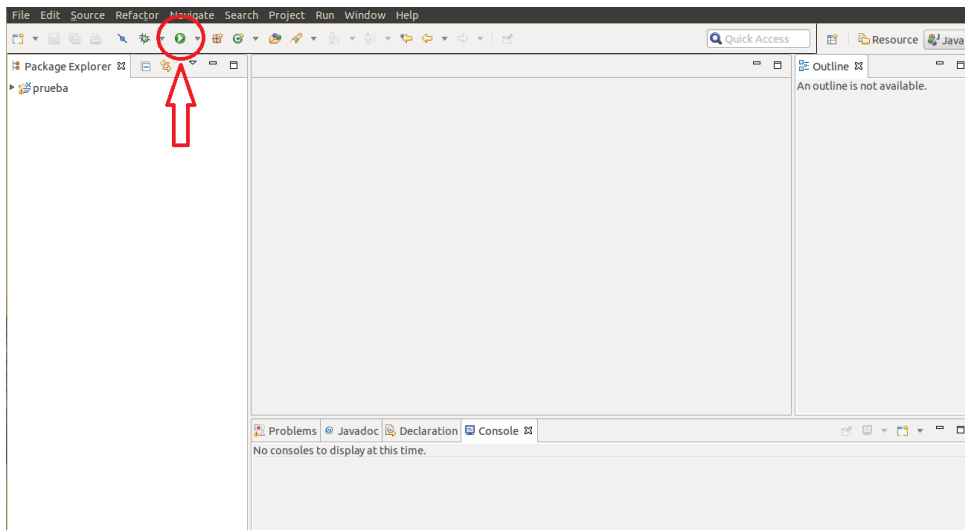


Figura 7.8: Ejecutar proyecto.

La primera vez que ejecutemos un proyecto deberemos indicar el punto de arranque. Para ello seleccionamos en la primera ventana *Java Application* (figura 7.9) y en la siguiente seleccionar la clase *Transformation.java* (figura 7.10).

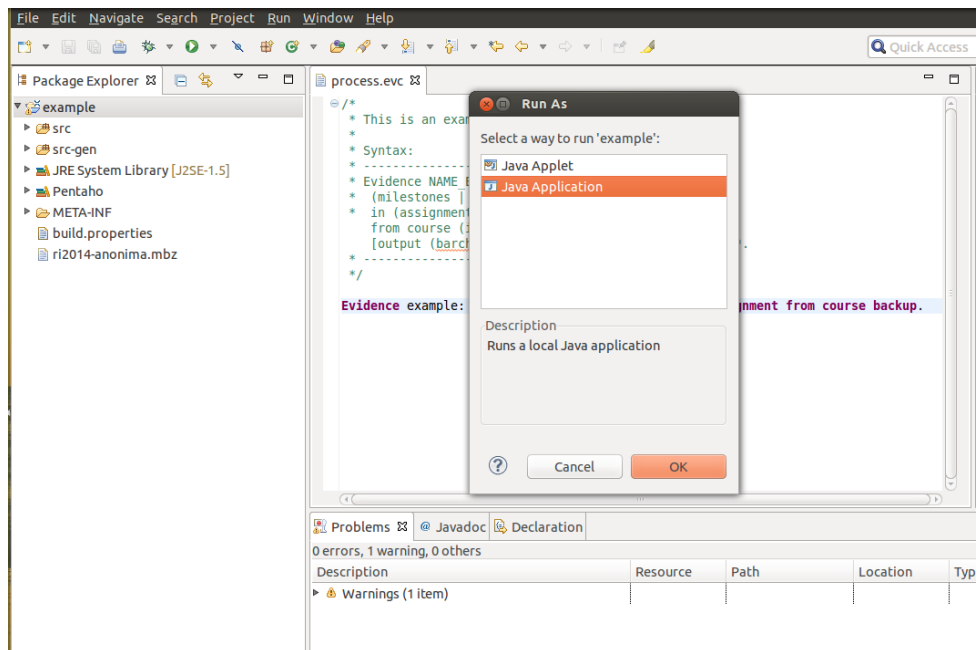


Figura 7.9: Indicar punto de arranque I.

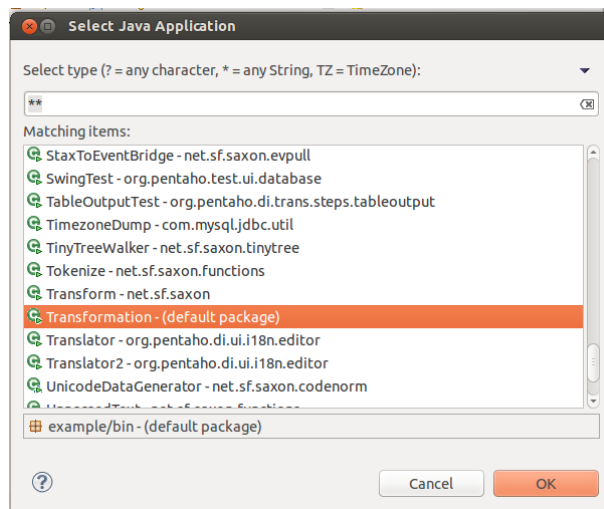


Figura 7.10: Indicar punto de arranque II.

Observaremos mientras se ejecuta la información de lo que va ocurriendo en la consola (figura 7.11) y una vez concluido el proceso, si abrimos la carpeta Output, observaremos los ficheros generados (figura 7.12).



```
Problems @ Javadoc Declaration Console x
<terminated> Transformation [Java Application] /usr/lib/jvm/java-6-openjdk-i386/bin/java (23/07/2014 16:32:25)
INFO 23-07 16:32:38,663 - Unzip - Finished job entry [Unzip file] (result=[true])
INFO 23-07 16:32:38,663 - Unzip - Job execution finished
INFO 23-07 16:32:39,183 - example - Dispatching started for transformation [example]
INFO 23-07 16:32:39,306 - example - Opening output stream in encoding: UTF-8
INFO 23-07 16:32:39,787 - Searching-module-users - Finished processing (I=9, O=0, R=0, W=9, U=0, E=0)
INFO 23-07 16:32:40,193 - Searching-submissions - Finished processing (I=32, O=0, R=0, W=32, U=0, E=0)
INFO 23-07 16:32:40,339 - Path-transformation - Finished processing (I=0, O=0, R=32, W=32, U=0, E=0)
INFO 23-07 16:32:40,620 - Searching-limit-Submissions - Finished processing (I=24, O=0, R=32, W=24, U=0, E=0)
INFO 23-07 16:32:40,621 - onTime - Finished processing (I=0, O=0, R=24, W=24, U=0, E=0)
INFO 23-07 16:32:40,627 - Sort-id - Finished processing (I=0, O=0, R=24, W=24, U=0, E=0)
INFO 23-07 16:32:40,640 - milestones2 - Finished processing (I=0, O=0, R=24, W=12, U=0, E=0)
INFO 23-07 16:32:40,642 - max - Finished processing (I=0, O=0, R=6, W=1, U=0, E=0)
INFO 23-07 16:32:40,944 - Join-Rows - Finished processing (I=0, O=0, R=7, W=6, U=0, E=0)
INFO 23-07 16:32:40,946 - percentage - Finished processing (I=0, O=0, R=6, W=6, U=0, E=0)
INFO 23-07 16:32:41,203 - MergeId - Finished processing (I=0, O=0, R=15, W=12, U=0, E=0)
INFO 23-07 16:32:41,207 - example-csv - Finished processing (I=0, O=7, R=6, W=6, U=0, E=0)
INFO 23-07 16:32:41,208 - example - Finished processing (I=0, O=6, R=6, W=6, U=0, E=0)
... Proceso finalizado
-----
Evalcourse: Gephi transformation started
-----
INFO 23-07 16:32:41,565 - tog - Dispatching started for transformation [tog]
INFO 23-07 16:32:41,698 - Read XML - Finished processing (I=1, O=0, R=0, W=1, U=0, E=0)
INFO 23-07 16:32:41,723 - Read XSL - Finished processing (I=1, O=0, R=0, W=1, U=0, E=0)
INFO 23-07 16:32:42,026 - ADD - Finished processing (I=0, O=0, R=2, W=1, U=0, E=0)
INFO 23-07 16:32:42,737 - XSL Transformation - Finished processing (I=0, O=0, R=1, W=1, U=0, E=0)
INFO 23-07 16:32:42,739 - Text file output - Finished processing (I=0, O=1, R=1, W=1, U=0, E=0)
... Proceso Gephi finalizado
-----
Evalcourse: Transformations completed
-----
```

Figura 7.11: Información de la consola.

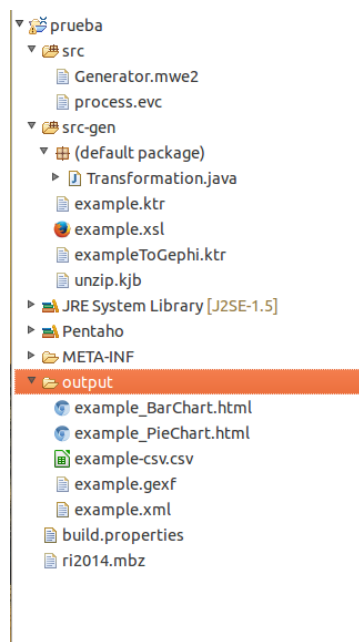


Figura 7.12: Información de la consola.



## Capítulo 8

# Manual de instalación y explotación

Las instrucciones de instalación y explotación del sistema se detallan a continuación.

### 8.1. Introducción

El objetivo principal de EvalCourse es la obtención de indicadores objetivos que mostrarán el grado de desempeño de los alumnos en las competencias genéricas del trabajo desarrollado en un LMS.

El ámbito de aplicación del software EvalCourse es la docencia y la investigación. Los profesores la usarán para que les sirva de ayuda en la dura tarea de evaluar y los investigadores para encontrar posibles nuevos parámetros con los que evaluar determinadas competencias, en definitiva, innovar en el campo de la evaluación.

### 8.2. Requisitos previos

Requisitos software necesarios:

- Aplicación RCP.
- Pentaho. Sólo si se quiere editar alguna transformación, sin ella también obtendremos los informes.
- Gephi. Necesario para visualizar gráficos en formato *Gephi*.

### 8.3. Procedimientos de instalación

- Vaya a la página del repositorio y descargue el proyecto: <https://www.assembla.com/spaces/evalcourse/documents>.
- Descargue la versión para su sistema operativo.

#### Enlaces de interés

Las aplicaciones las puede descargar de los siguientes enlaces:

- EvalCourse: <https://www.assembla.com/spaces/evalcourse/documents>
- Pentaho: <http://www.pentaho.com>
- Gephi: <https://gephi.org/>