



**ESCUELA SUPERIOR DE INGENIERÍA  
INGENIERÍA INFORMÁTICA DE SISTEMAS**

PROYECTO DE FIN DE CARRERA

José Alberto García Pinteño

7 de septiembre de 2015





## ESCUELA SUPERIOR DE INGENIERÍA

### INGENIERÍA INFORMÁTICA DE SISTEMAS

#### PROYECTO DE FIN DE CARRERA

- Departamento: Ingeniería Informática
- Directores del proyecto: Antonio Balderas Alberico y Manuel Palomo Duarte
- Autor del proyecto: José Alberto García Pinteño

Puerto Real, 7 de septiembre de 2015

Fdo: José Alberto García Pinteño



## ***Agradecimientos***

*Me gustaria darle las gracias a mis padres, ya que sin ellos no podria haber llegado hasta aqui, a mi familia por todo su apoyo y a todos los profesores que se han esforzado por dejar un poco de su conocimiento en mi, en especial a Manuel Palomo, por toda su paciencia y apoyo.*

## Resumen

Los wikis son un sistema muy popular como ayuda a la docencia. Cuando el número de estudiantes y la cantidad de información almacenada en un wiki aumentan, evaluar el trabajo de cada estudiante resulta difícil. Los wikis mantienen un registro con las diferencias entre las revisiones consecutivas de los artículos, que pueden ser usadas para la evaluación del aprendizaje. Esta información puede evaluarse a lo largo de la vida del wiki para obtener datos sobre la actividad de los estudiantes.

AssessMediaWiki es una aplicación web de código abierto que, al conectarse a una instalación MediaWiki, proporciona procedimientos de autoevaluación, hetero evaluación y evaluación entre iguales, a la vez que mantiene información sobre esas evaluaciones. Los supervisores pueden obtener informes que ayudan en la evaluación de los estudiantes.

Aunque hay un gran número de extensiones para el sistema MediaWiki, no hemos encontrado ninguna que permitiera evaluar contribuciones individuales a un wiki. La mayoría de las aproximaciones solo ofrecen formas de evaluar una versión en particular de un artículo (normalmente la más reciente), siendo ineficaces en este caso. Por ello, para evaluar la calidad de las contribuciones creamos AssessMediaWiki.

AssessMediaWiki implementa como base dos roles de usuario distintos: supervisores y estudiantes. Los estudiantes pueden elegir entre distintas opciones: evaluar una revisión, comprobar sus propias aportaciones evaluadas y verificar las evaluaciones ya enviadas. Por otro lado, los supervisores tienen un mayor número de opciones, como modificar los parámetros de los programas o vigilar las evaluaciones que los alumnos vayan haciendo.

**Palabras clave:** AssessMediaWiki, MediaWiki, Wiki, software libre, evaluación

# Índice general

<b>1. Introducción</b>	<b>1</b>
1.1. Motivación . . . . .	1
1.2. Alcance . . . . .	1
1.3. Glosario de Términos . . . . .	1
1.4. Organización del documento . . . . .	2
<b>2. Planificación</b>	<b>3</b>
2.1. Metodología de desarrollo . . . . .	3
2.2. Planificación del proyecto . . . . .	3
2.3. Organización . . . . .	6
2.4. Riesgos . . . . .	6
<b>I Desarrollo</b>	<b>7</b>
<b>3. Requisitos del Sistema</b>	<b>9</b>
3.1. Situación actual . . . . .	9
3.1.1. Entorno Tecnológico . . . . .	9
3.1.2. Fortalezas y Debilidades . . . . .	9
3.2. Objetivos del Sistema . . . . .	9
3.3. Catálogo de Requisitos . . . . .	9
3.3.1. Requisitos funcionales . . . . .	9
3.3.2. Requisitos no funcionales . . . . .	10
3.3.3. Requisitos de información . . . . .	10
3.4. Alternativas de Solución . . . . .	10
3.5. Solución Propuesta . . . . .	10
<b>4. Análisis del Sistema</b>	<b>11</b>
4.1. Modelo Conceptual . . . . .	11
4.2. Modelo de Casos de Uso . . . . .	14
4.2.1. Actores . . . . .	14
4.2.2. Descripción de los casos de uso . . . . .	16
4.3. Modelo de Comportamiento . . . . .	23
4.4. Modelo de Interfaz de Usuario . . . . .	30

<b>5. Diseño del Sistema</b>	<b>39</b>
5.1. Arquitectura del Sistema . . . . .	39
5.1.1. Arquitectura Física . . . . .	39
5.1.2. Arquitectura Lógica . . . . .	39
5.2. Parametrización del software base . . . . .	41
5.3. Diseño Físico de Datos . . . . .	41
<b>6. Construcción del Sistema</b>	<b>43</b>
6.1. Entorno de Construcción . . . . .	43
6.2. Código Fuente . . . . .	43
6.3. Scripts de Base de datos . . . . .	44
<b>7. Pruebas del Sistema</b>	<b>51</b>
7.1. Estrategia . . . . .	51
7.2. Entorno de Pruebas . . . . .	51
7.3. Roles . . . . .	51
7.4. Niveles de Pruebas . . . . .	51
7.4.1. Pruebas Unitarias . . . . .	51
7.4.2. Pruebas de Integración . . . . .	51
7.4.3. Pruebas de Sistema . . . . .	52
7.4.4. Pruebas de Aceptación . . . . .	52
<b>II Epílogo</b>	<b>53</b>
<b>8. Manual de implantación y explotación</b>	<b>57</b>
8.1. Introducción . . . . .	57
8.2. Requisitos previos . . . . .	57
8.3. Inventario de componentes . . . . .	57
8.4. Procedimientos de instalación . . . . .	58
8.5. Pruebas de implantación . . . . .	59
8.6. Procedimientos de operación y nivel de servicio . . . . .	59
<b>9. Manual de usuario</b>	<b>61</b>
9.1. Introducción . . . . .	61
9.2. Características . . . . .	61
9.3. Requisitos previos . . . . .	61
9.4. Uso del sistema . . . . .	62
<b>10. Conclusiones</b>	<b>63</b>
10.1. Objetivos alcanzados . . . . .	63
10.2. Lecciones aprendidas . . . . .	63
10.3. Trabajo futuro . . . . .	63
<b>Bibliografía</b>	<b>65</b>
<b>Información sobre Licencia</b>	<b>67</b>

# Índice de figuras

2.1. Desarrollo en cascada.	3
2.2. Datos del diagrama de Gantt.	4
2.3. Diagrama de Gantt.	4
2.4. Diagrama de Gantt (girada).	5
4.1. Diagrama de la base de datos de AMW 1.0.	12
4.2. Diagrama de la base de datos de AMW 2.0.	13
4.3. Modelo de casos de uso.	15
4.4. Caso de uso: administrar roles.	23
4.5. Caso de uso: crear rol.	24
4.6. Caso de uso: editar permisos de rol.	24
4.7. Caso de uso: eliminar rol.	25
4.8. Caso de uso: asignar rol.	25
4.9. Caso de uso: administrar ejercicios de evaluación.	26
4.10. Caso de uso: crear ejercicio de evaluación.	26
4.11. Caso de uso: editar ejercicio de evaluación.	27
4.12. Caso de uso: eliminar ejercicio de evaluación.	27
4.13. Caso de uso: editar criterios de evaluación.	28
4.14. Caso de uso: hacer preasignaciones.	28
4.15. Caso de uso: procesar ráfagas.	29
4.16. Caso de uso: acceso a ayuda.	29
4.17. Pantalla de login.	30
4.18. Pantalla de inicio del usuario (sin la sala de debug, eso es debido a que el modo de desarrollo esta activado).	31
4.19. Pantalla de usuario sin revisiones.	31
4.20. Pantalla de metaevaluaciones.	32
4.21. Pantalla de listado de metaevaluaciones.	32
4.22. Pantalla de listado de alumnos.	33
4.23. Pantalla de parámetros.	34
4.24. Pantalla de configuración de roles.	35
4.25. Pantalla de configuración de ejercicios de evaluación.	36
4.26. Pantalla de ayuda extra.	37
5.1. Diagrama de la base de datos de AMW 2.0.	40
5.2. Diagrama de la base de datos de MediaWiki (disponible en la web de MediaWiki).	40
5.3. Diagrama de la base de datos de AMW 2.0.	41

6.1. Diagrama de la base de datos de AMW 1.0. . . . .	49
6.2. Diagrama de la base de datos de AMW 2.0. . . . .	50
9.1. Pantalla de ayuda extra. . . . .	62

# Índice de cuadros



# Capítulo 1

## Introducción

### 1.1. Motivación

Tras trabajar durante un breve periodo de tiempo en AssessMediaWiki gracias a la beca Icaro, se planteo la posibilidad de seguir trabajando sobre el mismo, pero esta vez para usarlo como proyecto de fin de carrera.

Tras esa experiencia, viendo todo lo que había prendido de ella (manejar nuevos lenguajes y herramientas de desarrollo, distintas formas de enfocar los problemas, etc), mi interés por aprender y explorar diversos métodos didácticos y el interés y respeto que tengo hacia el software libre y su comunidad, decidí continuar desarrollando AssessMediaWiki y aprovechar para usarlo como proyecto de fin de carrera, con la posterior motivación extra de intentar hacer publicaciones con él.

### 1.2. Alcance

Este proyecto esta orientado a actualizar los métodos docentes, añadiendo nuevas herramientas y formas de interactuar con los alumnos, empezando por una forma de trabajo en grupo a la que ya puedan estar acostumbrados y permitiéndoles tener la responsabilidad de evaluarse entre ellos (no todo el trabajo, solo las partes mas significativas y teniendo el docente la voz final) y a su vez poder ver el progreso individual en las tareas asignadas con la opción de poder obtener estadísticas gracias a programas como StatMediaWiki o CleverFigures, que complementan la experiencia docente 2.0

### 1.3. Glosario de Términos

- AMW - AssessMediaWiki
- Edición - Aportacion realizada por un alumno al MediaWiki
- Metaevaluación - evaluación realizada sobre una evaluación existente, para poder ver así la corrección de la evaluación existente

## **1.4. Organización del documento**

-Introducción: Resumen de la descripción del proyecto, objetivos y estructura del documento presente.

to do

# Capítulo 2

## Planificación

### 2.1. Metodología de desarrollo

Definición del proceso de desarrollo, ciclo de vida y metodología empleada durante la elaboración del proyecto. Las fases y/o iteraciones que proponga el método empleado deberán quedar recogidas en la planificación que se detalle más adelante.

Se ha optado por una metodología de desarrollo en cascada, como se aprecia en la siguiente imagen:

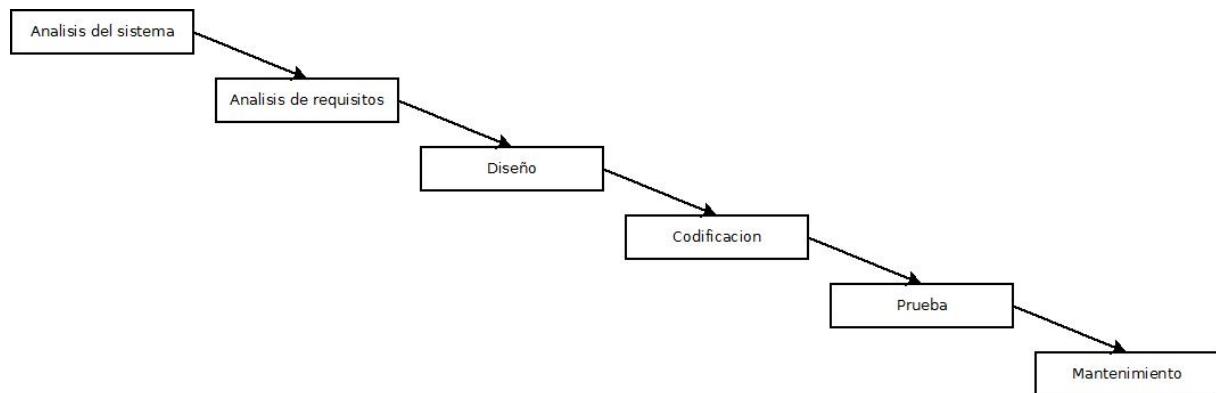


Figura 2.1: Desarrollo en cascada.

### 2.2. Planificación del proyecto

Estimación temporal y definición del calendario básico (hitos principales e iteraciones). Desarrollo de la planificación detallada, utilizando un diagrama de Gantt.

Se debe incluir una comparación cuantitativa del tiempo y el esfuerzo realmente invertido frente al estimado y planificado. Estos datos pueden recogerse del sistema de gestión de tareas empleado para el seguimiento del proyecto.

Fase	Proceso	Fecha Inicio	Duración	Fecha Fin	Completado	Pendiente	Realizado
Análisis	Análisis del sistema	15/01/2015	14	29/01/2015	100%	0,00	14,00
	Definición de nuevas funciones	30/01/2015	7	06/02/2015	100%	0,00	7,00
Desarrollo	Administración de roles	07/02/2015	7	14/02/2015	100%	0,00	7,00
	Asignación de roles	19/02/2015	10	01/03/2015	100%	0,00	10,00
	Ejercicios de evaluación	06/03/2015	7	13/03/2015	100%	0,00	7,00
	Ediciones por alumno	18/03/2015	7	25/03/2015	100%	0,00	7,00
	Evaluaciones por edición	30/03/2015	7	06/04/2015	100%	0,00	7,00
	Algoritmo de selección	11/04/2015	15	26/04/2015	100%	0,00	15,00
	Algoritmo de distribución	03/05/2015	15	18/05/2015	100%	0,00	15,00
	Algoritmo preasignaciones	25/05/2015	15	09/06/2015	100%	0,00	15,00
	Procesado ráfagas	16/06/2015	20	06/07/2015	100%	0,00	20,00
Testeo	Administración de roles	15/02/2015	3	18/02/2015	100%	0,00	3,00
	Asignación de roles	02/03/2015	3	05/03/2015	100%	0,00	3,00
	Ejercicios de evaluación	14/03/2015	3	17/03/2015	100%	0,00	3,00
	Ediciones por alumno	26/03/2015	3	29/03/2015	100%	0,00	3,00
	Evaluaciones por edición	07/04/2015	3	10/04/2015	100%	0,00	3,00
	Algoritmo de selección	27/04/2015	5	02/05/2015	100%	0,00	5,00
	Algoritmo de distribución	19/05/2015	5	24/05/2015	100%	0,00	5,00
	Algoritmo preasignaciones	10/06/2015	5	15/06/2015	100%	0,00	5,00
	Procesado ráfagas	07/07/2015	5	12/07/2015	100%	0,00	5,00
Documentación	Generado de gráficas	13/07/2015	10	23/07/2015	100%	0,00	10,00
	Creación de la documentación	24/07/2015	20	13/08/2015	100%	0,00	20,00
	Creación de diapositivas	14/08/2015	10	24/08/2015	100%	0,00	10,00

Figura 2.2: Datos del diagrama de Gantt.

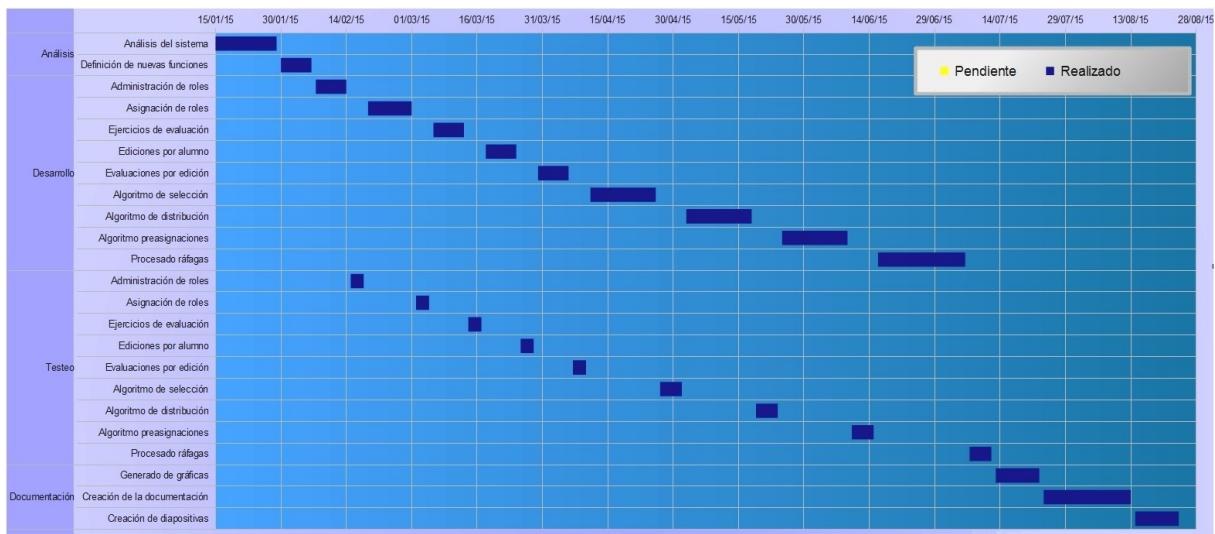


Figura 2.3: Diagrama de Gantt.

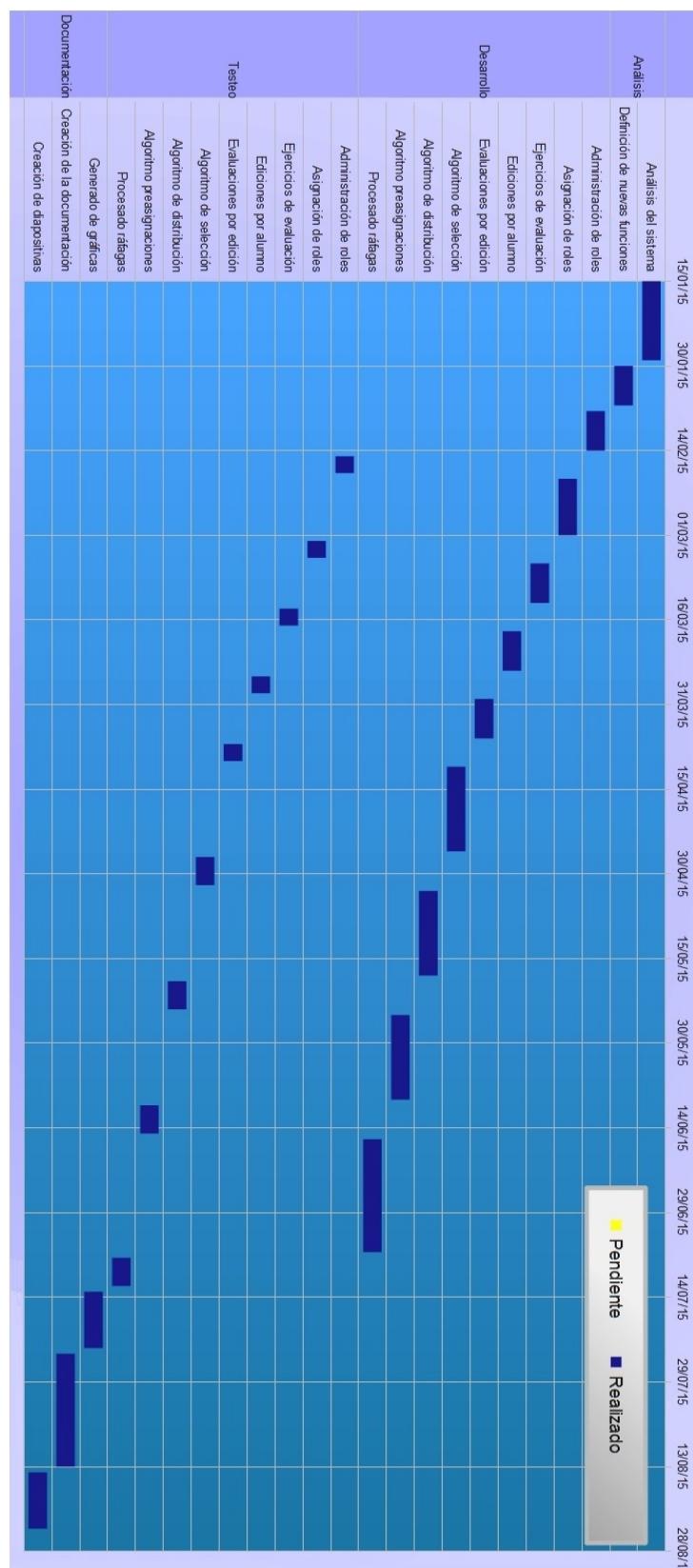


Figura 2.4: Diagrama de Gantt (girada).

## **2.3. Organización**

Relación de las personas (roles) involucradas en el proyecto y de cómo se estructuran las relaciones entre las mismas para ejecutar el proyecto. Relación de los recursos inventariables utilizados en el proyecto: equipamiento informático (hardware y software), herramientas empleadas, etc.

Para llevar a cabo este proyecto es necesario un MediaWiki, un servidor para alojar AssessMediaWiki y en caso de que el profesor no posea los conocimientos necesarios para la instalación, la colaboración de personal que facilite la instalación de AssessMediaWiki

## **2.4. Riesgos**

Enumeración de los riesgos del proyecto, indicando su posible impacto (efecto que la ocurrencia del citado riesgo tendría en el desarrollo del proyecto) y la probabilidad de ocurrencia. Una vez los riesgos son identificados y priorizados, hay que definir los planes necesarios para reducir los efectos del riesgo una vez se haya materializado o disminuir que este ocurra.

De momento hay un pequeño problema con los nombres de los roles creados en el sistema AssessMediaWiki y los ejercicios de evaluación, si los nombres llevan espacio no se podrán eliminar, y es posible que no puedan ser modificados, es un pequeño detalle que se puede arreglar en un trabajo futuro, para evitar este problema basta con usar barras bajas en lugar de espacios.

También puede considerarse un riesgo usar el framework ([CodeIgniter](#)) sin conocimiento previo, así como que en un futuro dicho framework deje de recibir soporte.

# Parte I

## Desarrollo



# Capítulo 3

## Requisitos del Sistema

### 3.1. Situación actual

En la versión actual (la que esta funcionando en estos momentos) de AssessMediaWiki nos encontramos con un algoritmo de selección que escoge una edición de forma aleatoria entre las  $n$  ediciones mas significativas (las de mayor tamaño), de forma que en el peor de los casos nos encontramos con la situación de que las  $n - 1$  ediciones mas significativas nunca serán seleccionadas.

Esta versión de AssessMediaWiki también carece de un procesado de ediciones para ciertos wiki-comportamientos (ráfagas, correcciones ortográficas, etc) y los roles de los usuarios se limitan a los de alumno y supervisor.

Estos situaciones han sido complementadas con esta nueva versión, añadiendo un algoritmo distinto de selección de ediciones, un sistema de administración de roles y el procesado de ráfagas

#### 3.1.1. Entorno Tecnológico

Para este proyecto sera necesario contar con un MediaWiki, normalmente alojado en un servidor, en el cual seria recomendable alojar el mismo software del proyecto.

#### 3.1.2. Fortalezas y Debilidades

Fortalezas: Herramienta muy útil para añadir nuevas posibilidades a los métodos docentes.//  
Debilidades: Es necesario un mínimo de conocimientos para su instalación, al igual que para la instalación del MediaWiki.

### 3.2. Objetivos del Sistema

El objetivo del sistema es añadir nuevas funciones al método docente y darle a los alumnos mas responsabilidades, así como analizar su participación y resultados.

### 3.3. Catálogo de Requisitos

#### 3.3.1. Requisitos funcionales

Listado de las nuevas funcionalidades que ofrece el sistema:

- Creación y edición de nuevos roles.
- Creación y edición de ejercicios de evaluación.
- Procesado de ráfagas.
- Preasignaciones de ediciones.

### **3.3.2. Requisitos no funcionales**

Descripción de otros requisitos (relacionados con la calidad del software) que el sistema deberá satisfacer: portabilidad, seguridad, estándares de obligado cumplimiento, accesibilidad, usabilidad, etc:

- MediaWiki.
- Servidor para hospedar tanto el MediaWiki como AssessMediaWiki.
- Conexión a internet.

### **3.3.3. Requisitos de información**

En esta sección se describen los requisitos de gestión de información (datos) que el sistema debe gestionar:

- IDs de los usuarios del MediaWiki.
- IDs de las ediciones creadas por los usuarios.
- Notas y comentarios asignados a ediciones evaluadas.
- Parámetros de configuración
- Roles creados por el administrador.
- Ejercicios de evaluación y sus configuraciones.

## **3.4. Alternativas de Solución**

Se presento la posibilidad de usar evalcomix como herramienta de evaluación alternativa.

## **3.5. Solución Propuesta**

Se descartó usar evalcomix debido a que se consideró mejor opción implementar un método propio que fuese escalable.

# Capítulo 4

## Análisis del Sistema

### 4.1. Modelo Conceptual

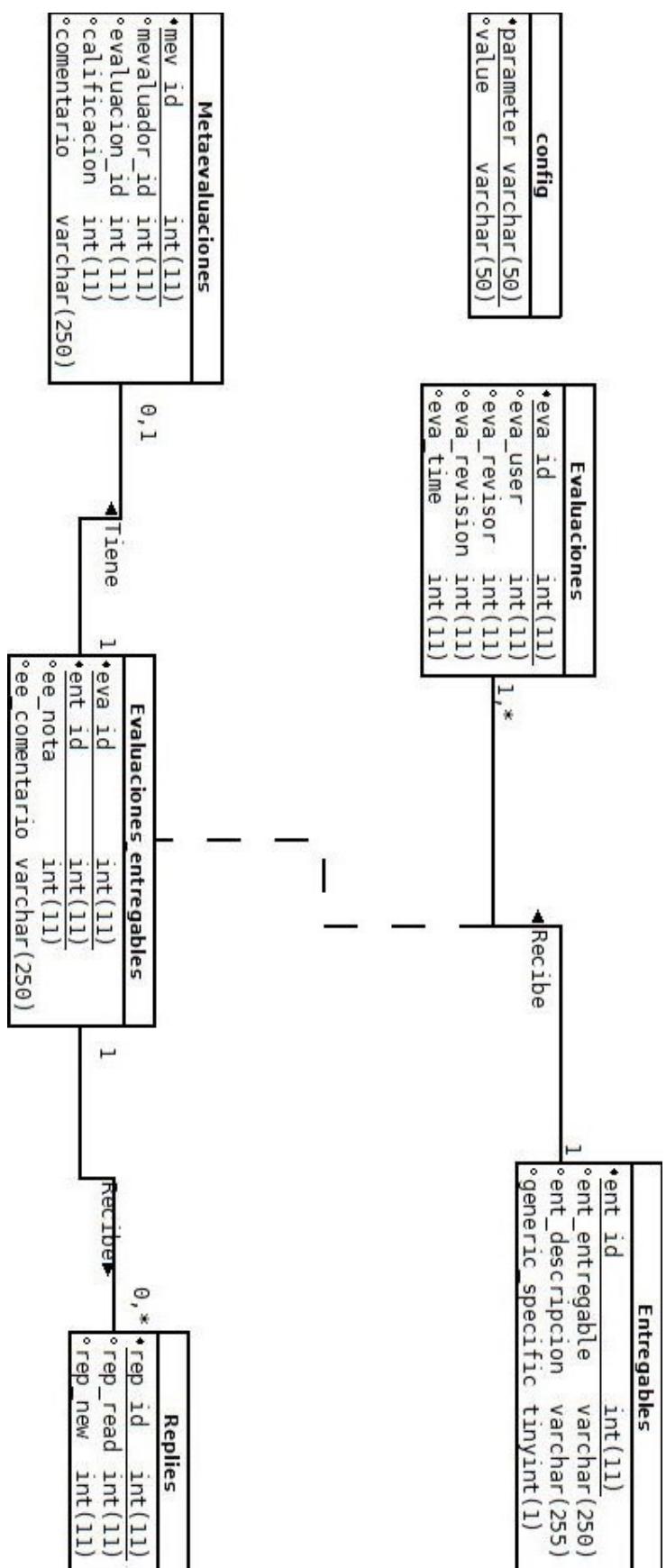


Figura 4.1: Diagrama de la base de datos de AMW 1.0.

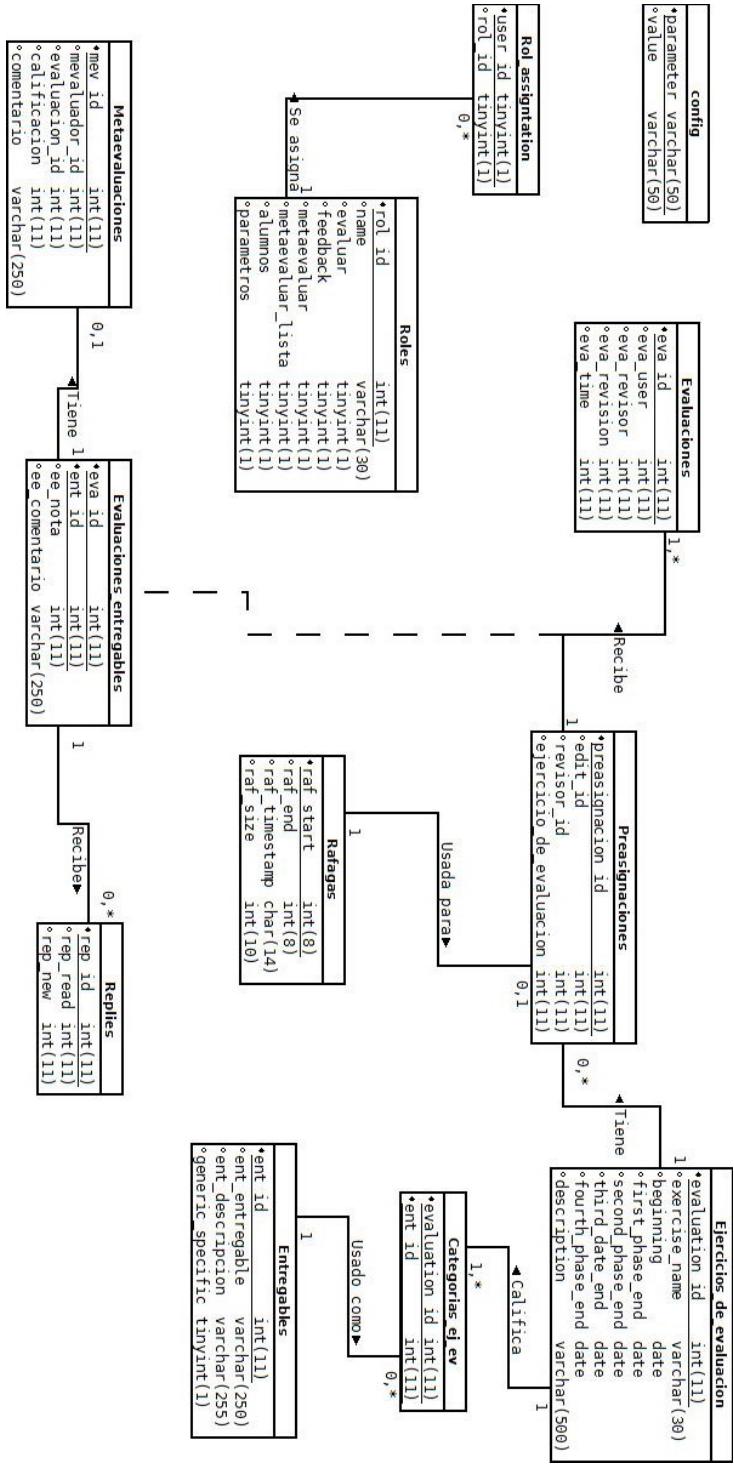


Figura 4.2: Diagrama de la base de datos de AMW 2.0.

## **4.2. Modelo de Casos de Uso**

### **4.2.1. Actores**

Los actores básicos son:

- Administrador/Profesor: El usuario supervisor del sistema
- Estudiante: Usuario que realizara las evaluaciones de las ediciones suyas o de sus compañeros

Pero hay que tener en cuenta que el administrador puede crear nuevos roles, lo que generara nuevos actores dependiendo de los permisos que se les conceda a dichos roles.

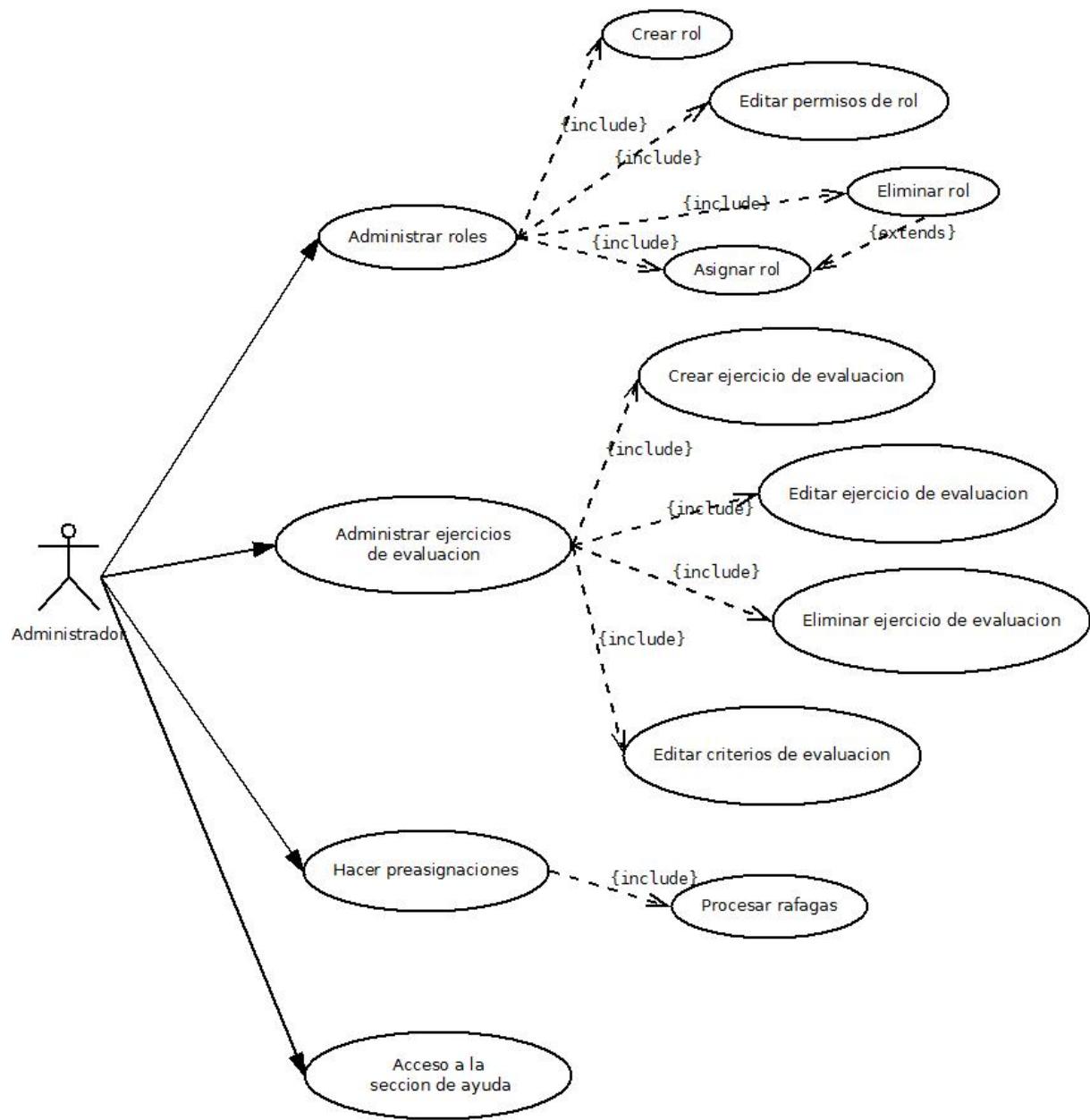


Figura 4.3: Modelo de casos de uso.

#### 4.2.2. Descripción de los casos de uso

##### Caso de uso: **Administrar roles**

Descripción: El usuario se dirige a la pagina de administración de roles.

Actores: Usuario, Sistema

Precondiciones: El usuario esta logeado y es administrador.

Postcondiciones: El sistema muestra la pagina de administración de roles.

Escenario principal:

- 1. El usuario hace clic en el enlace a la administración de roles.
- El sistema muestra la pagina de administración de roles.

Escenario alternativo 1:

- \*.\* En cualquier momento del caso de uso el usuario usa los botones de navegación (atrás, adelante, actualizar) y termina el proceso actual sin guardar los datos ni ejecutarse ninguna acción.

Escenario alternativo 2:

- \*.\* En cualquier momento del caso de uso el usuario viaja a otra sección de la web o a otra web y termina el proceso actual sin guardar los datos ni ejecutarse ninguna acción.

##### Caso de uso: **Crear rol**

Descripción: El usuario decide crear un nuevo rol.

Actores: Usuario, Sistema

Precondiciones: El usuario esta logeado y es administrador y se encuentra en la pagina de administración de roles.

Postcondiciones: Se introduce un nuevo rol en la base de datos del sistema.

Escenario principal:

- 1. El usuario rellena los datos para crear un nuevo rol en la pagina de administración de roles.
- 2. El usuario hace clic en crear rol.
- 3. El sistema comprueba que no existe ningún otro rol en el sistema con el nombre del nuevo rol a crear.
- 4. El sistema añade el nuevo rol a la base de datos del sistema.
- 5. El sistema actualiza la pagina de administración de roles, mostrando también el nuevo rol añadido.

Escenario alternativo 1:

- \*.\* En cualquier momento del caso de uso el usuario usa los botones de navegación (atrás, adelante, actualizar) y termina el proceso actual sin guardar los datos ni ejecutarse ninguna acción.

Escenario alternativo 2:

- \*.\* En cualquier momento del caso de uso el usuario viaja a otra sección de la web o a otra web y termina el proceso actual sin guardar los datos ni ejecutarse ninguna acción.

Escenario alternativo 3:

- 3.a El sistema detecta que el nuevo rol a introducir ya existe en el sistema.
- 4.a El sistema actualiza la pagina de administración de roles, sin añadir el nuevo rol que el usuario pretendía añadir.

#### Caso de uso: **Editar permisos de rol**

Descripción: El usuario decide editar los permisos de un rol.

Actores: Usuario, Sistema

Precondiciones: El usuario esta logeado y es administrador y se encuentra en la pagina de administración de roles.

Postcondiciones: Se modifican los permisos de un rol en la base de datos del sistema.

Escenario principal:

- 1. El usuario rellena los datos para modificar los permisos de un rol en la pagina de administración de roles.
- 2. El usuario hace clic en modificar rol.
- 3. El sistema modifica los permisos del rol en la base de datos del sistema.
- 4. El sistema actualiza la pagina de administración de roles, mostrando los cambios efectuados.

Escenario alternativo 1:

- \*.\* En cualquier momento del caso de uso el usuario usa los botones de navegación (atrás, adelante, actualizar) y termina el proceso actual sin guardar los datos ni ejecutarse ninguna acción.

Escenario alternativo 2:

- \*.\* En cualquier momento del caso de uso el usuario viaja a otra sección de la web o a otra web y termina el proceso actual sin guardar los datos ni ejecutarse ninguna acción.

#### Caso de uso: **Eliminar rol**

Descripción: El usuario decide eliminar rol.

Actores: Usuario, Sistema

Precondiciones: El usuario esta logeado y es administrador y se encuentra en la pagina de administración de roles.

Postcondiciones: Se elimina el rol seleccionado de la base de datos del sistema.

Escenario principal:

- 1. El sistema muestra una lista desplegable con los roles existentes en el sistema, exceptuando los de administrador y estudiante.
- 2. El usuario selecciona el rol que desea eliminar.
- 3. El usuario hace clic en el botón de eliminar rol.

- 4. El sistema elimina todas las asignaciones de usuarios al rol a eliminar.
- 5. El sistema elimina el rol seleccionado de la base de datos del sistema.
- 6. El sistema actualiza la pagina de administración de roles, mostrando los cambios efectuados.

Escenario alternativo 1:

- \*.\* En cualquier momento del caso de uso el usuario usa los botones de navegación (atrás, adelante, actualizar) y termina el proceso actual sin guardar los datos ni ejecutarse ninguna acción.

Escenario alternativo 2:

- \*.\* En cualquier momento del caso de uso el usuario viaja a otra sección de la web o a otra web y termina el proceso actual sin guardar los datos ni ejecutarse ninguna acción.

### Caso de uso: **Asignar rol**

Descripción: El usuario decide asignar un rol a un estudiante.

Actores: Usuario, Sistema

Precondiciones: El usuario esta logeado y es administrador y se encuentra en la pagina de administración de roles.

Postcondiciones: Se introduce un nuevo rol en la base de datos del sistema.

Escenario principal:

- 1. El sistema muestra una lista desplegable con los usuarios existentes en el sistema y otra con los roles existentes en el sistema.
- 2. El usuario selecciona un usuario y un rol de las listas.
- 3. El usuario hace clic en el botón de asignar rol.
- 4. El sistema comprueba que el usuario no tenia ningún rol asignado.
- 5. El sistema introduce los datos del usuario seleccionado y el rol que se le va a asignar.
- 6. El sistema actualiza la pagina de administración de roles, mostrando los cambios efectuados.

Escenario alternativo 1:

- \*.\* En cualquier momento del caso de uso el usuario usa los botones de navegación (atrás, adelante, actualizar) y termina el proceso actual sin guardar los datos ni ejecutarse ninguna acción.

Escenario alternativo 2:

- \*.\* En cualquier momento del caso de uso el usuario viaja a otra sección de la web o a otra web y termina el proceso actual sin guardar los datos ni ejecutarse ninguna acción.

Escenario alternativo 3:

- 4.a El sistema detecta que el usuario ya tiene un rol asignado.

- 5.a El sistema detecta que el nuevo rol que se le quiere asignar es el de estudiante.
- 7.a El sistema borra de la base de datos la asignación previa que tuviera el usuario seleccionado.
- 8.a El sistema actualiza la pagina de administración de roles, mostrando los cambios efectuados.

Escenario alternativo 4:

- 5.b El sistema detecta que el nuevo rol que se quiere asignar no es el de estudiante.
- 6.b El sistema actualiza la base de datos con los nuevos datos introducidos
- 7.b El sistema actualiza la pagina de administración de roles mostrando los cambios efectuados.

#### Caso de uso: **Administrar ejercicios de evaluación**

Descripción: El usuario se dirige a la pagina de administración de ejercicios de evaluación

Actores: Usuario, Sistema

Precondiciones: El usuario esta logeado y es administrador.

Postcondiciones: El sistema muestra la pagina de administración de ejercicios de evaluación

Escenario principal:

- 1. El usuario hace clic en el enlace a la administración de ejercicios de evaluación
- 2. El sistema muestra la pagina de administración de ejercicios de evaluación

Escenario alternativo 1:

- \*.\* En cualquier momento del caso de uso el usuario usa los botones de navegación (atrás, adelante, actualizar) y termina el proceso actual sin guardar los datos ni ejecutarse ninguna acción.

Escenario alternativo 2:

- \*.\* En cualquier momento del caso de uso el usuario viaja a otra sección de la web o a otra web y termina el proceso actual sin guardar los datos ni ejecutarse ninguna acción.

#### Caso de uso: **Crear ejercicio de evaluación**

Descripción: El usuario decide crear un nuevo ejercicio de evaluación

Actores: Usuario, Sistema

Precondiciones: El usuario esta logeado y es administrador y se encuentra en la pagina de administración de ejercicios de evaluación

Postcondiciones: Se introduce un nuevo ejercicio de evaluación en la base de datos del sistema.

Escenario principal:

- 1. El usuario rellena los datos para crear un nuevo ejercicio de evaluación en la pagina de administración de ejercicios de evaluación
- 2. El usuario hace clic en crear ejercicio de evaluación
- 3. El sistema comprueba que no existe ningún otro ejercicio de evaluación en el sistema con el nombre del nuevo ejercicio de evaluación a crear.

- 4. El sistema añade el nuevo ejercicio de evaluación a la base de datos del sistema.
- 5. El sistema actualiza la pagina de administración de ejercicios de evaluación, mostrando también el nuevo ejercicio de evaluación añadido.

Escenario alternativo 1:

- \*.\* En cualquier momento del caso de uso el usuario usa los botones de navegación (atrás, adelante, actualizar) y termina el proceso actual sin guardar los datos ni ejecutarse ninguna acción.

Escenario alternativo 2:

- \*.\* En cualquier momento del caso de uso el usuario viaja a otra sección de la web o a otra web y termina el proceso actual sin guardar los datos ni ejecutarse ninguna acción.

Escenario alternativo 3:

- 3.a El sistema detecta que el nuevo ejercicio de evaluación a introducir ya existe en el sistema.
- 4.a El sistema actualiza la pagina de administración de ejercicios de evaluación, sin añadir el nuevo ejercicio de evaluación que el usuario pretendía añadir.

#### Caso de uso: **Editar ejercicio de evaluación**

Descripción: El usuario decide editar un ejercicio de evaluación

Actores: Usuario, Sistema

Precondiciones: El usuario esta logeado y es administrador y se encuentra en la pagina de administración de ejercicios de evaluación

Postcondiciones: Se modifican un ejercicio de evaluación en la base de datos del sistema.

Escenario principal:

- 1. El usuario rellena los datos para modificar un ejercicio de evaluación en la pagina de administración de roles.
- 2. El usuario hace clic en modificar ejercicio de evaluación
- 4. El sistema modifica el ejercicio de evaluación en la base de datos del sistema.
- 5. El sistema actualiza la pagina de administración de ejercicios de evaluación, mostrando los cambios efectuados.

Escenario alternativo 1:

- \*.\* En cualquier momento del caso de uso el usuario usa los botones de navegación (atrás, adelante, actualizar) y termina el proceso actual sin guardar los datos ni ejecutarse ninguna acción.

Escenario alternativo 2:

- \*.\* En cualquier momento del caso de uso el usuario viaja a otra sección de la web o a otra web y termina el proceso actual sin guardar los datos ni ejecutarse ninguna acción.

### Caso de uso: **Eliminar ejercicio de evaluación**

Descripción: El usuario decide eliminar un ejercicio de evaluación

Actores: Usuario, Sistema

Precondiciones: El usuario esta logeado y es administrador y se encuentra en la pagina de administración de ejercicio de evaluación

Postcondiciones: Se elimina el ejercicio de evaluación seleccionado de la base de datos del sistema.

Escenario principal:

- 1. El sistema muestra una lista desplegable con los ejercicios de evaluación existentes en el sistema.
- 2. El usuario selecciona el ejercicio de evaluación que desea eliminar.
- 3. El usuario hace clic en el botón de eliminar ejercicio de evaluación
- 4. El sistema elimina el ejercicio de evaluación seleccionado de la base de datos del sistema.
- 5. El sistema actualiza la pagina de administración de ejercicios de evaluación, mostrando los cambios efectuados.

Escenario alternativo 1:

- \*.\* En cualquier momento del caso de uso el usuario usa los botones de navegación (atrás, adelante, actualizar) y termina el proceso actual sin guardar los datos ni ejecutarse ninguna acción.

Escenario alternativo 2:

- \*.\* En cualquier momento del caso de uso el usuario viaja a otra sección de la web o a otra web y termina el proceso actual sin guardar los datos ni ejecutarse ninguna acción.

### Caso de uso: **Editar criterios de evaluación**

Descripción: El usuario decide editar los criterios de evaluación de un ejercicio de evaluación

Actores: Usuario, Sistema

Precondiciones: El usuario esta logeado y es administrador y se encuentra en la pagina de administración de ejercicio de evaluación

Postcondiciones: Se modifican los criterios de evaluación de un ejercicio de evaluación en la base de datos del sistema.

Escenario principal:

- 1. El sistema muestra una tabla con los ejercicio de evaluación y criterios de evaluación existentes en el sistema.
- 2. El usuario modifica los criterios de evaluación de uno de los ejercicios de evaluación
- 3. El usuario hace clic en el botón de modificar criterios de evaluación
- 4. El sistema modifica los datos para el ejercicio de evaluación seleccionado.
- 5. El sistema actualiza la pagina de administración de roles, mostrando los cambios efectuados.

Escenario alternativo 1:

- \*.\* En cualquier momento del caso de uso el usuario usa los botones de navegación (atrás, adelante, actualizar) y termina el proceso actual sin guardar los datos ni ejecutarse ninguna acción.

Escenario alternativo 2:

- \*.\* En cualquier momento del caso de uso el usuario viaja a otra sección de la web o a otra web y termina el proceso actual sin guardar los datos ni ejecutarse ninguna acción.

Caso de uso: **Hacer Preasignaciones**

Descripción: El usuario decide realizar las preasignaciones para un ejercicio de evaluación

Actores: Usuario, Sistema

Precondiciones: El usuario esta logeado y es administrador y se encuentra en la pagina de administración de ejercicio de evaluación

Postcondiciones: Se realizan las preasignaciones para un ejercicio de evaluación

Escenario principal:

- 1. El sistema muestra una lista desplegable con los ejercicio de evaluación para los cuales aun no se han realizado preasignaciones existentes en el sistema.
- 2. El usuario selecciona uno de los ejercicios de evaluación
- 3. El usuario hace clic en el botón de hacer preasignaciones
- 4. El sistema realiza el procesado de ráfagas y posteriormente asigna las ediciones del ejercicio de evaluación a los Estudiantes según la configuración del sistema.
- 5. El sistema actualiza la pagina de administración de ejercicios de evaluación, mostrando los cambios efectuados.

Escenario alternativo 1:

- \*.\* En cualquier momento del caso de uso el usuario usa los botones de navegación (atrás, adelante, actualizar) y termina el proceso actual sin guardar los datos ni ejecutarse ninguna acción.

Escenario alternativo 2:

- \*.\* En cualquier momento del caso de uso el usuario viaja a otra sección de la web o a otra web y termina el proceso actual sin guardar los datos ni ejecutarse ninguna acción.

Caso de uso: **Procesar ráfagas**

Descripción: Se procesan las posibles ráfagas existentes en el sistema

Actores: Sistema

Precondiciones: El usuario selecciona la opción de hacer preasignaciones.

Postcondiciones: Se analizan las ediciones existentes y se crean las ráfagas que procedan.

Escenario principal:

- 1. El sistema busca en cada pagina todas las ediciones realizadas en el plazo establecido.
- 2. El sistema detecta que 2 ediciones consecutivas han sido realizadas en menos de el periodo de tiempo establecido por el mismo usuario.
- 3. El sistema crea una ráfaga o añade la ultima edición a una ráfaga existente.

### 4.3. Modelo de Comportamiento

En las siguientes imágenes podemos ver los modelos de comportamiento:

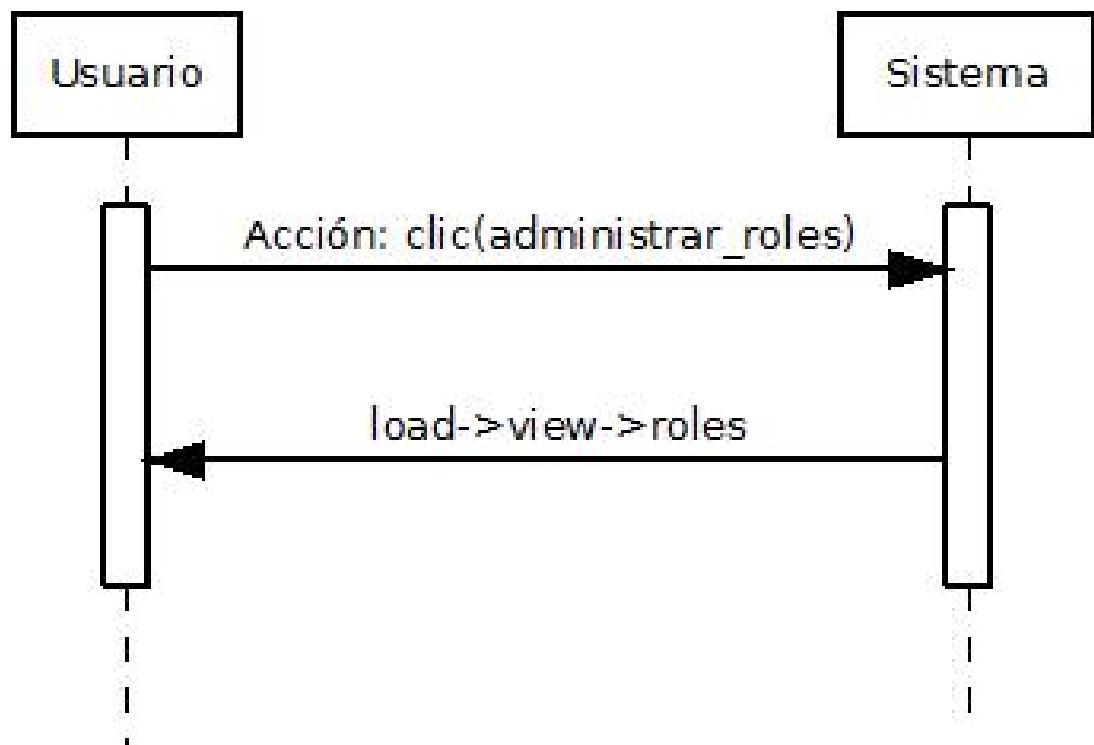


Figura 4.4: Caso de uso: administrar roles.

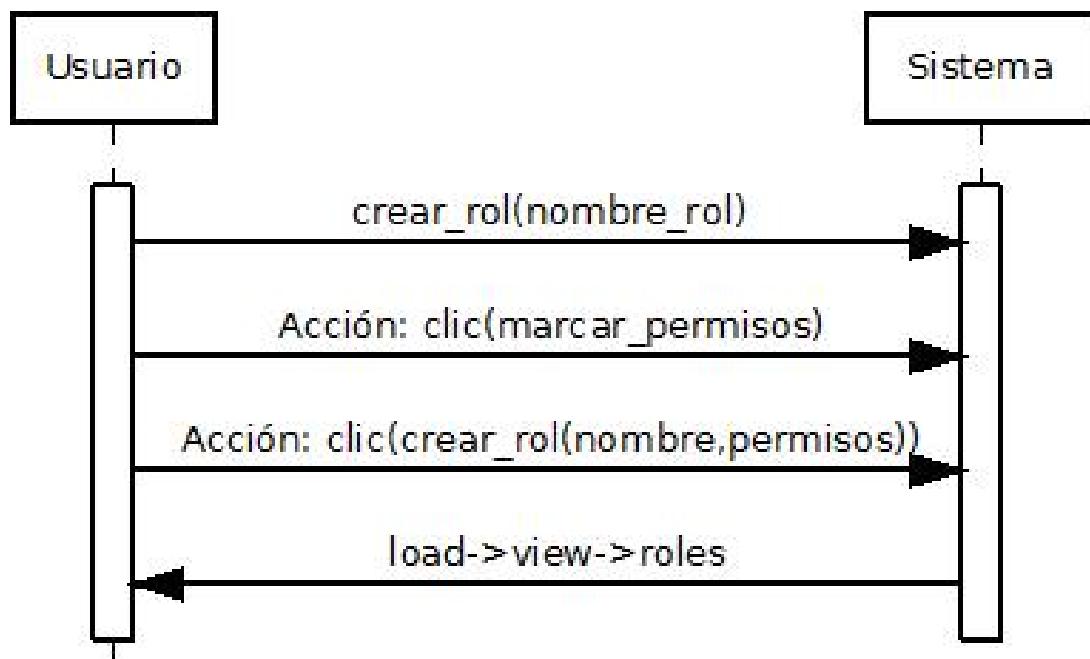


Figura 4.5: Caso de uso: crear rol.

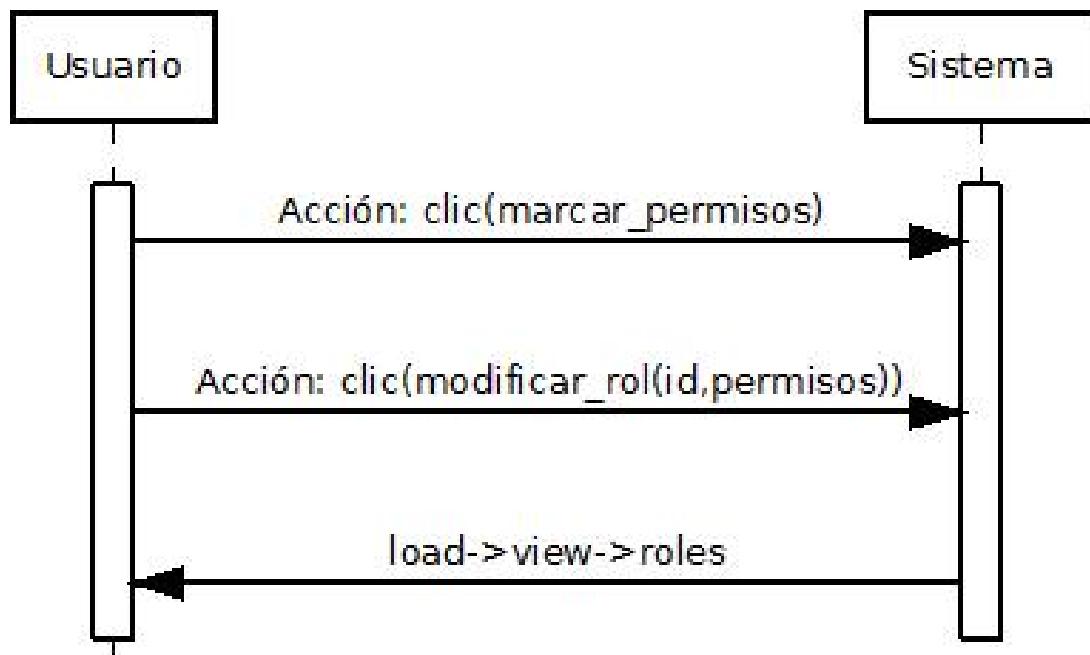


Figura 4.6: Caso de uso: editar permisos de rol.

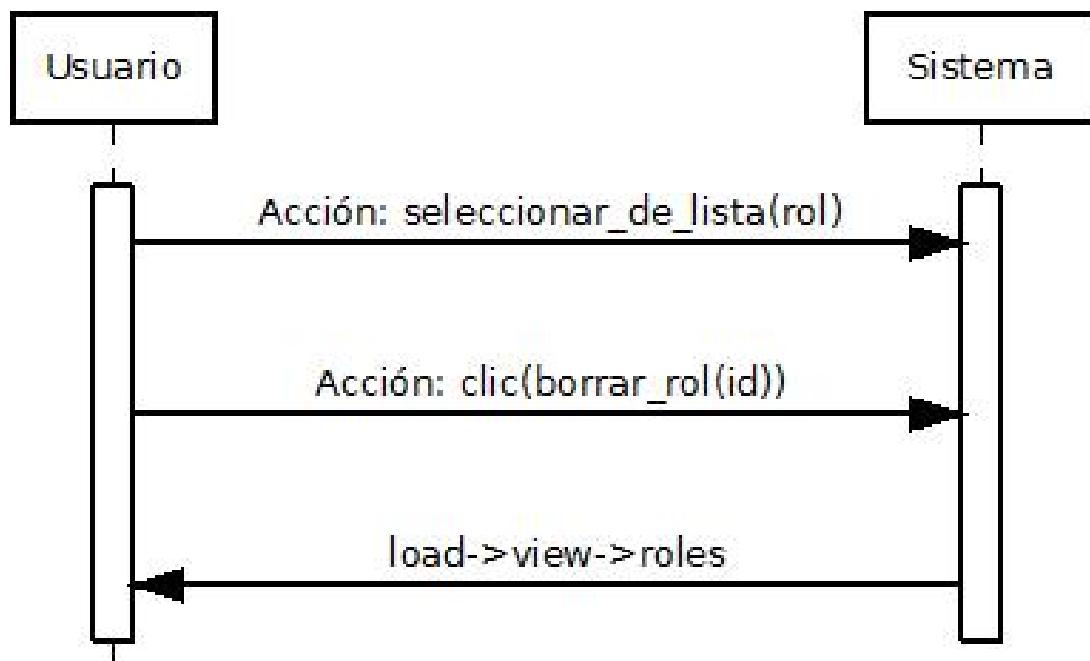


Figura 4.7: Caso de uso: eliminar rol.

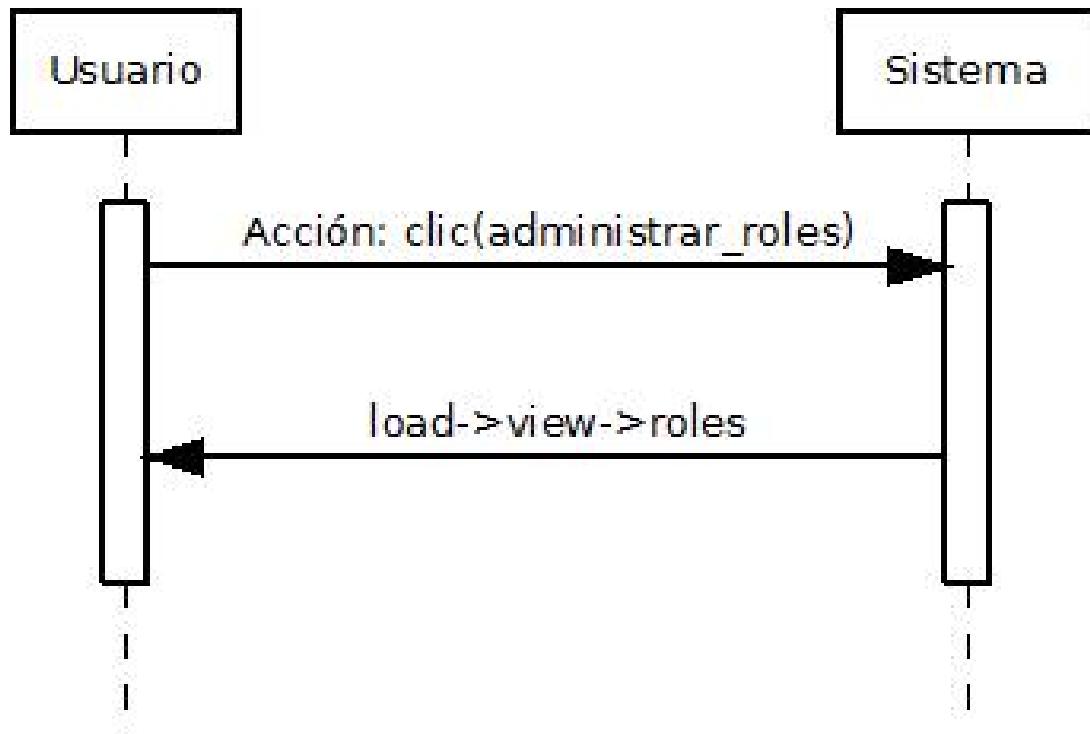


Figura 4.8: Caso de uso: asignar rol.

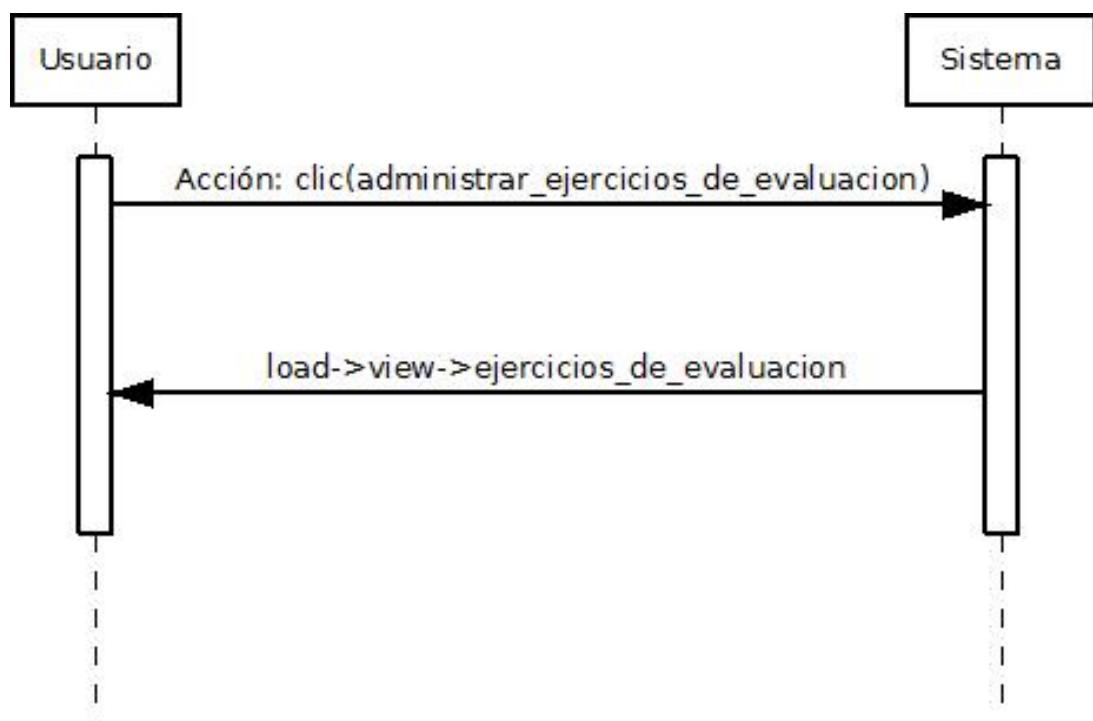


Figura 4.9: Caso de uso: administrar ejercicios de evaluación.

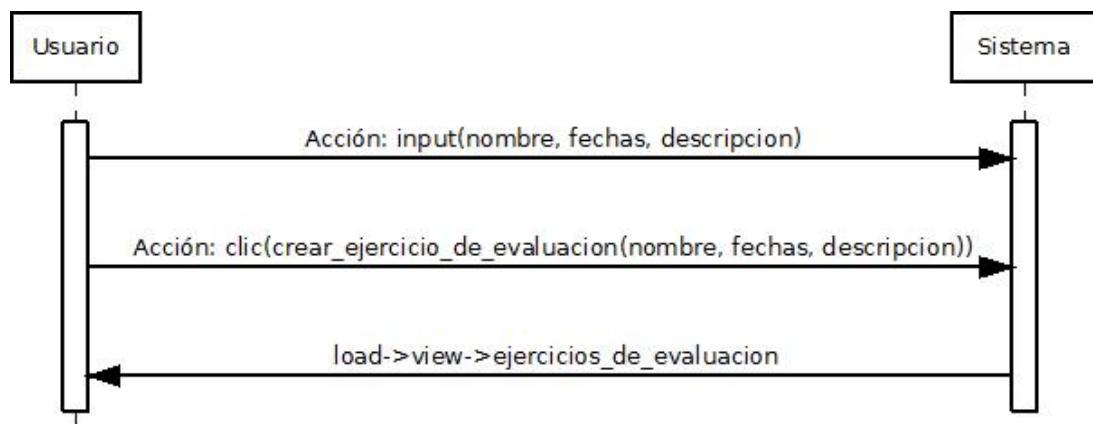


Figura 4.10: Caso de uso: crear ejercicio de evaluación.

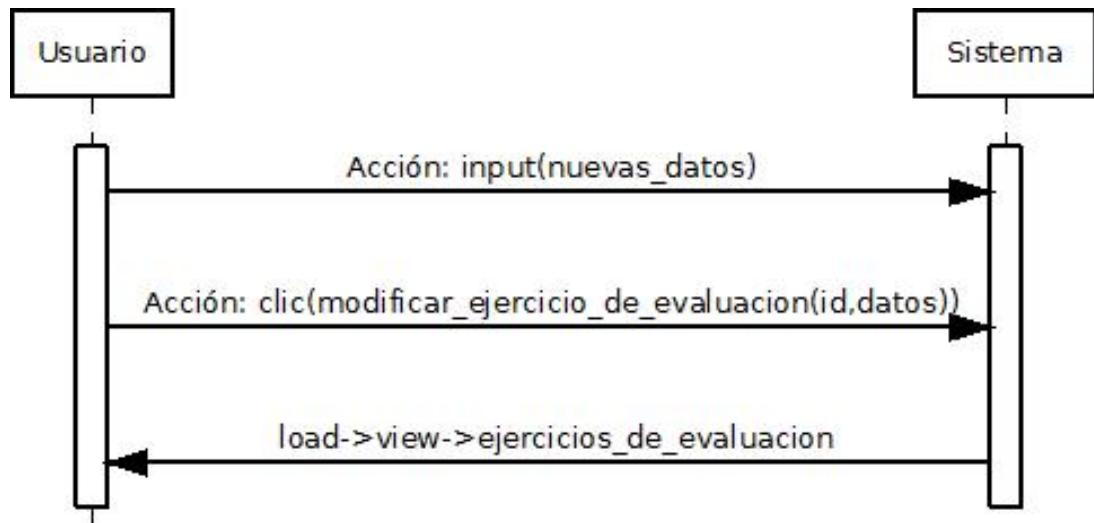


Figura 4.11: Caso de uso: editar ejercicio de evaluación.

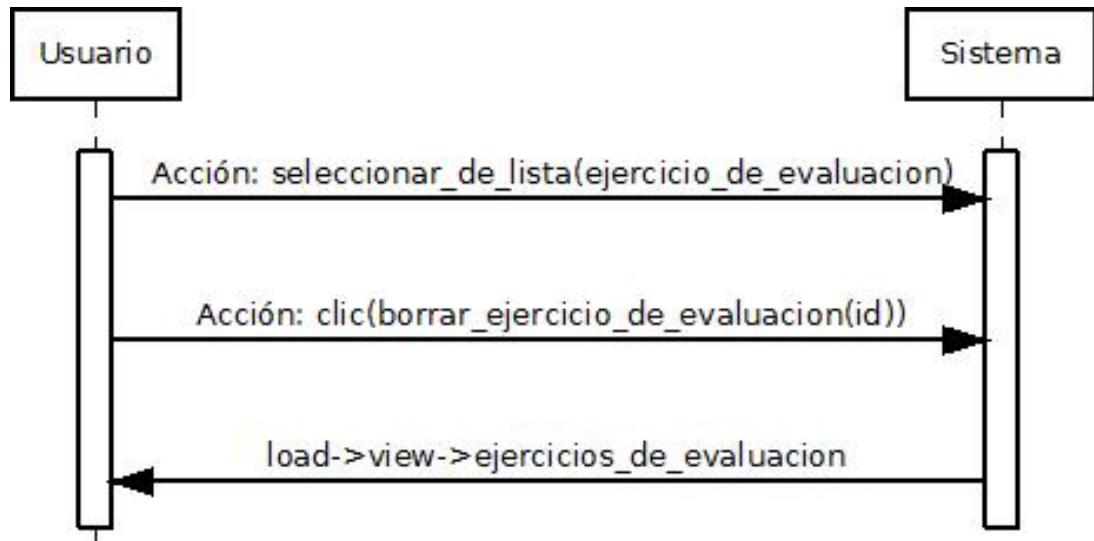


Figura 4.12: Caso de uso: eliminar ejercicio de evaluación.

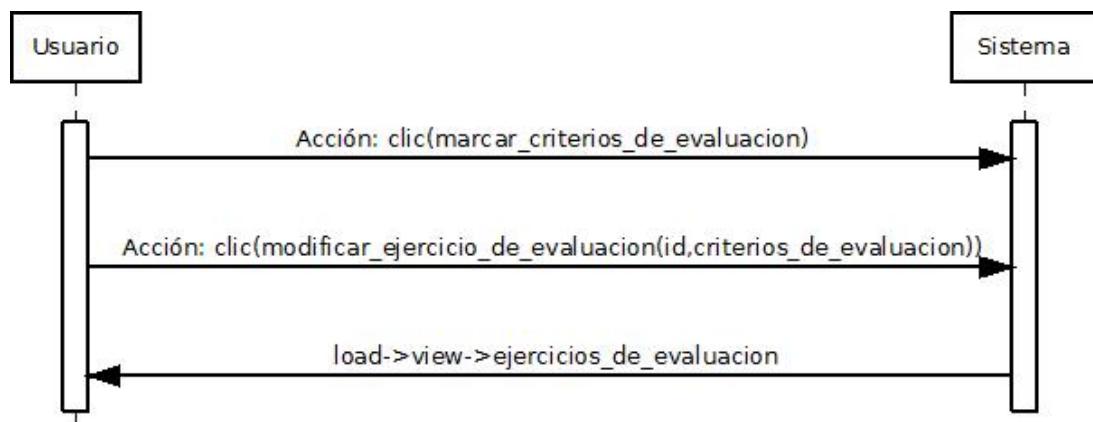


Figura 4.13: Caso de uso: editar criterios de evaluación.

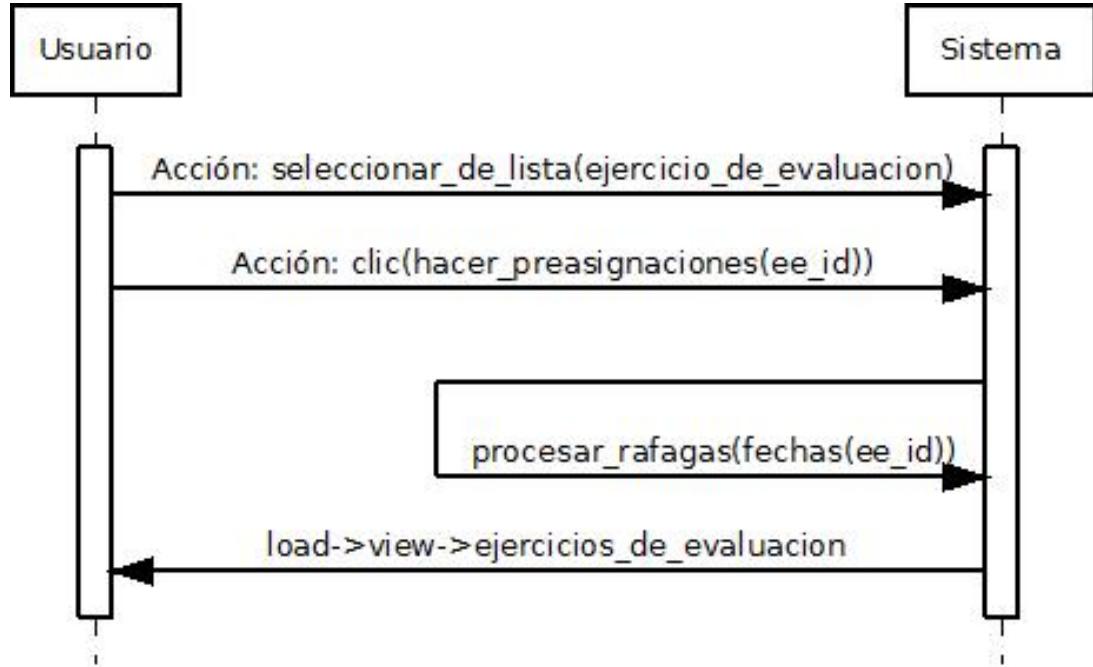


Figura 4.14: Caso de uso: hacer preasignaciones.

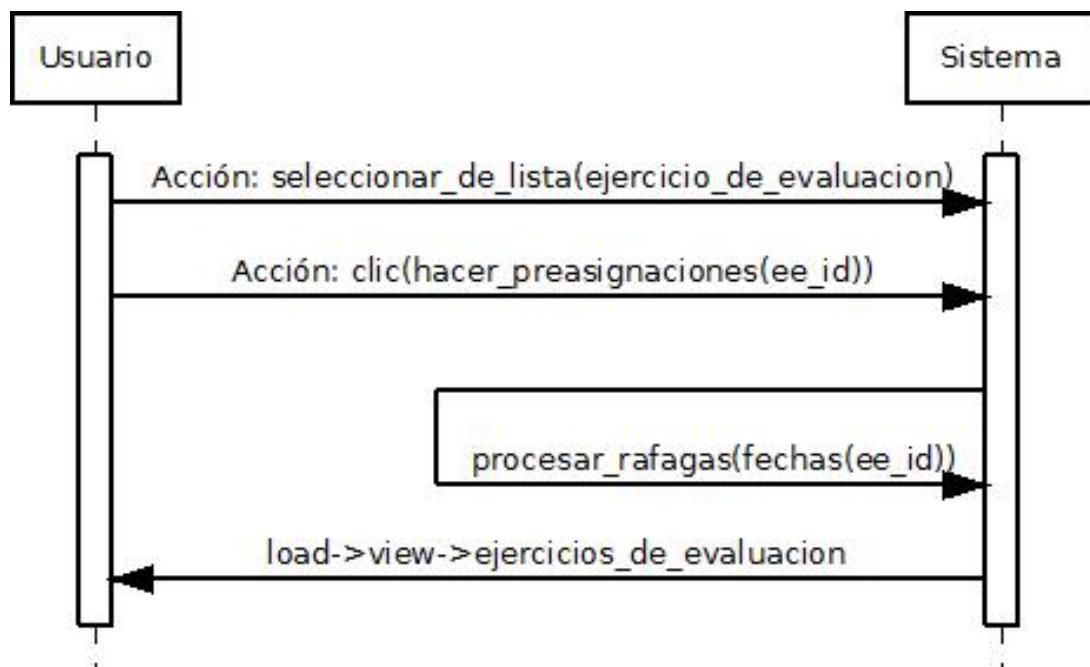


Figura 4.15: Caso de uso: procesar ráfagas.

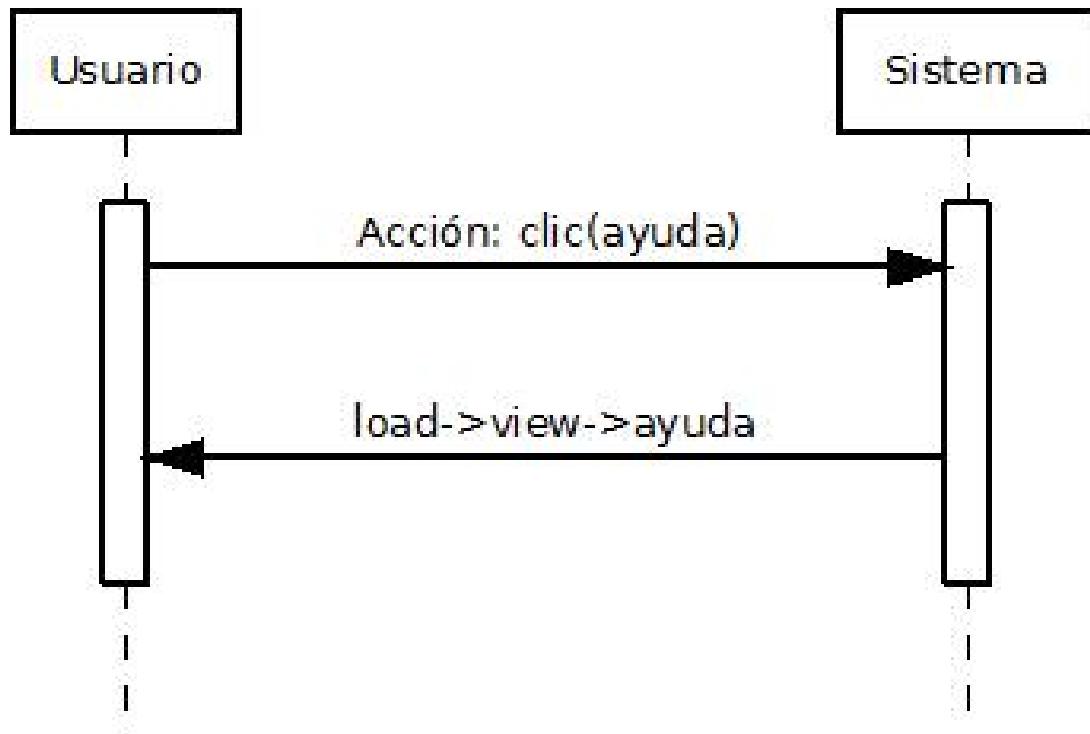


Figura 4.16: Caso de uso: acceso a ayuda.

#### 4.4. Modelo de Interfaz de Usuario

En esta sección se adjuntan capturas de pantalla de la interfaz del sistema:



The screenshot shows the login interface for the 'AssessMediaWiki' system. At the top left is a logo consisting of a magnifying glass icon with a checkmark inside it, followed by the text 'AssessMediaWiki'. Below the logo, a message says 'Please log in'. There are two input fields: 'Username:' and 'Password:', each preceded by a label and a small placeholder box. A single 'Log in' button is located below the password field.

Username:	<input type="text"/>
Password:	<input type="password"/>

Figura 4.17: Pantalla de login.

The screenshot shows the user interface for AssessMediaWiki. At the top, there is a logo consisting of a magnifying glass icon with a checkmark inside it, followed by the text "AssessMediaWiki". Below the logo is a navigation bar with links for "Assess", "My assessments", "Debug Room", and "Logout".

Below the navigation bar, a message states: "Hi **anom10**, there are 10 revisions pending for an assessment." It also says: "Here you are the criteria for the evaluation exercise, bellow that you'll find a link to the revision to assess." and "Aqui ira la descripcion del ejercicio de evaluacion".

It further notes: "You may grade the revision here below. [This is the url to assess.](#)".

A table is displayed with four rows, each representing a pending assessment:

Detected	Criterion	Grade	Description
<input type="checkbox"/>	<b>Test</b> qweqweqwe	0 <input type="button" value="▼"/>	<input type="text"/>
<input type="checkbox"/>	<b>Ortografia</b> asdasd	0 <input type="button" value="▼"/>	<input type="text"/>
<input type="checkbox"/>	<b>Probando</b> test2	0 <input type="button" value="▼"/>	<input type="text"/>
<input type="checkbox"/>	<b>Prueba</b> sfkstyk	0 <input type="button" value="▼"/>	<input type="text"/>

At the bottom left of the table area is a button labeled "Puntuar".

Figura 4.18: Pantalla de inicio del usuario (sin la sala de debug, eso es debido a que el modo de desarrollo esta activado).

The screenshot shows the user interface for AssessMediaWiki. At the top, there is a logo consisting of a magnifying glass icon with a checkmark inside it, followed by the text "AssessMediaWiki". Below the logo is a navigation bar with links for "Assess", "My assessments", "Metaevaluations", "All Metaevaluations", "Students", "Parameters", "Debug Room", and "Logout".

Below the navigation bar, a message states: "Hola anom1, en este momento no hay entradas para revisar. Inténtelo en otro momento."

Figura 4.19: Pantalla de usuario sin revisiones.



# AssessMediaWiki

Assess   My assessments   Metaevaluations   All Metaevaluations   Students   Parameters

Debug Room   Logout

Hi anom1 , there are no evaluations pending for a metaevaluation.

You may grade the evaluation of the revision here below. [This is the url assessed.](#)

Criterion	Grade	Description
Test	3	

Calification	Description
<input type="text"/>	<input type="text"/>

Figura 4.20: Pantalla de metaevaluaciones.



# AssessMediaWiki

Assess   My assessments   Metaevaluations   All Metaevaluations   Students   Parameters

Debug Room   Logout

## List of metaevaluations

There are a total of 13 metaevaluations done:

Metaevaluator	Metaevaluation grade (1-5)	Metaevaluation comments	URL	Criterion	Evaluation grade	Evaluation description	Evaluator
anom2	3		url	Test Ortografia	4 5		anom4
anom4	4	nice	url	Test Ortografia Probando	3 6 10	a b c	anom3
anom4	3	sep	url	Test Ortografia Probando	2 4 6	8 10 12	anom3

Figura 4.21: Pantalla de listado de metaevaluaciones.

The screenshot shows the main interface of the AssessMediaWiki application. At the top left is the logo 'AssessMediaWiki' with a magnifying glass icon. The top navigation bar includes links for 'Assess', 'My assessments', 'Metaevaluations', 'All Metaevaluations', 'Students', 'Parameters', 'Debug Room' (highlighted in red), and 'Logout'. Below the navigation is a section titled 'List of students' containing a table with 10 rows of student data. The columns are 'Students', 'Report', and 'Resume'. Each row contains a student name followed by a 'Report (CSV)' link under 'Report' and a 'Metaevaluations resume' link under 'Resume'.

Students	Report	Resume
anom1	<a href="#">Report (CSV)</a>	<a href="#">Metaevaluations resume</a>
anom10	<a href="#">Report (CSV)</a>	<a href="#">Metaevaluations resume</a>
anom100	<a href="#">Report (CSV)</a>	<a href="#">Metaevaluations resume</a>
anom101	<a href="#">Report (CSV)</a>	<a href="#">Metaevaluations resume</a>
anom102	<a href="#">Report (CSV)</a>	<a href="#">Metaevaluations resume</a>
anom103	<a href="#">Report (CSV)</a>	<a href="#">Metaevaluations resume</a>
anom104	<a href="#">Report (CSV)</a>	<a href="#">Metaevaluations resume</a>
anom105	<a href="#">Report (CSV)</a>	<a href="#">Metaevaluations resume</a>
anom106	<a href="#">Report (CSV)</a>	<a href="#">Metaevaluations resume</a>

Figura 4.22: Pantalla de listado de alumnos.



Assess	My assessments	Metaevaluations	All Metaevaluations	Students	Parameters	
Debug Room						Logout

## List of evaluation criteria

Add another criteria    Generate CSV

Criteria	Generic / Specific	Actions	
Test	Generic	Edit	Delete
Ortografia	Specific	Edit	Delete
Probando	Generic	Edit	Delete
Prueba	Specific	Edit	Delete

## General parameters

Category	<input type="text"/>
Start date	<input type="text" value="19900101010101"/> Format is yyyyymmddhhmmss
End date	<input type="text" value="20150101010101"/> Format is yyyyymmddhhmmss
Evals per student	<input type="text" value="10"/>
Meta-evals per student	<input type="text" value="0"/>
Evaluations per edition	<input type="text" value="3"/>
Each student must have a minimun of X editions evaluated	<input type="text" value="2"/>
Autoevaluation (1 = Allowed, 0 = Disallowed)	<input type="text" value="0"/>
Wiki URL	<input type="text" value="http://osi2.uca.es/wikihaskeII/"/>

## Evaluation exercises management

## Roles

## Extra help

Figura 4.23: Pantalla de parámetros.

 AssessMediaWiki

Assess My assessments Metaevaluations All Metaevaluations Students Parameters  
Debug Room Logout

### Rol permissions

Role name	Permissions to access to:	Ases	My assessments	Metaevaluations	All Metaevaluations	Students	Parameters	
Admin	<input checked="" type="checkbox"/>	<a href="#">Edit role permissions</a>						
Student	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<a href="#">Edit role permissions</a>
Test	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<a href="#">Edit role permissions</a>
Test_1_1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<a href="#">Edit role permissions</a>

### Create new role

Role name (without spaces)

Ases	<input checked="" type="checkbox"/>
My assessments	<input checked="" type="checkbox"/>
Metaevaluations	<input type="checkbox"/>
All Metaevaluations	<input type="checkbox"/>
Students	<input type="checkbox"/>
Parameters	<input type="checkbox"/>

[Create new role](#)

### Create new role

Role name (without spaces)

Ases	<input checked="" type="checkbox"/>
My assessments	<input checked="" type="checkbox"/>
Metaevaluations	<input type="checkbox"/>
All Metaevaluations	<input type="checkbox"/>
Students	<input type="checkbox"/>
Parameters	<input type="checkbox"/>

[Create new role](#)

### Delete role

Select role to delete

### Assign role

Select user to assign a role  Select role to assign  Assign role

### Users with roles

User	Role
anom1	Admin
anom2	Admin

Figura 4.24: Pantalla de configuración de roles.

**AssessMediaWiki**

Assess My assessments Metaevaluations All Metaevaluations Students Parameters  
Debug Room Logout

### Evaluation exercises

Evaluation exercise	Details						
Name	Beginning	End of exercise description phase	End of develop phase	End of evaluation phase	End of teacher corrections phase	Criteria for the students	Action
Preasignation_test	2009-01-01	2009-01-02	2010-12-30	2011-01-01	2011-01-02	Aqui ira la descripción del ejercicio	<a href="#">Edit exercise</a>
Prueba 2.0	2015-07-01	2015-07-02	2015-07-03	2015-07-04	2015-07-05	sdfg	<a href="#">Edit exercise</a>

### Make preasignations

Select evaluation exercise

### Delete preasignations

Delete preasignations from

### Criteria for evaluation exercises

Evaluation exercise	Test	Ortografía	Probando	Prueba	Action
Preasignation_test	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<a href="#">Edit evaluation exercise criteria</a>
Prueba 2.0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<a href="#">Edit evaluation exercise criteria</a>

### Delete preasignations

Delete preasignations from

### Criteria for evaluation exercises

Evaluation exercise	Test	Ortografía	Probando	Prueba	Action
Preasignation_test	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<a href="#">Edit evaluation exercise criteria</a>
Prueba 2.0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<a href="#">Edit evaluation exercise criteria</a>

### Create new evaluation exercise

Evaluation exercise name (without spaces)   
 Beginning date   
 End of exercise description phase   
 End of develop phase   
 End of evaluation phase   
 End of teacher corrections phase   
 Description

### Delete evaluation exercise

Select evaluation exercise to

Figura 4.25: Pantalla de configuración de ejercicios de evaluación.

The screenshot shows the main interface of AssessMediaWiki. At the top is a logo consisting of a magnifying glass icon with a checkmark inside it, followed by the text "AssessMediaWiki". Below the logo is a horizontal navigation bar with the following items: "Assess", "My assessments", "Metaevaluations", "All Metaevaluations", "Students", "Parameters", "Debug Room" (which is highlighted in green), and "Logout".

The **evaluations per students** indicates how many editions will evaluate each student, it's the main function of AssessMediaWiki.

The **metaevaluations** are evaluations of the evaluation of an edition made by another student, to allow metaevaluations they must be set in more than 0 and we have to define a role for metaevaluator students, allowing them to access to the metaevaluations section.

The amount of **evaluations per edition** indicates how many times one edition will be evaluated for several students and cannot be lower than 1, each edition has to receive at least one evaluation.

The **minimum of editions evaluated per student** indicates the amount of editions that would be evaluated for each student (always the most significant) after that minimum it will take the most significant one that haven't been selected yet, so if the minimum is 2 we can find students with 2 editions evaluated and students with 8 editions evaluated.

The only case in which we will find people with less editions evaluated than the specified value is the case that the student hasn't reached the minimum number of editions (students with one edition for example).

The **autoevaluation** allows students to evaluate their own editions, the only available values are 0 (off) and 1 (on).

Students for the evaluations of the **evaluation exercises** will be selected automatically, the only requisite is to have done any contribution to the wiki in the **develop phase**. If some student doesn't make any change during the **develop phase**, it would be impossible to assign her/him any edition of another student to evaluate it.

## Important

---

If we configure two **evaluation exercises** in which the **develop phase** has days in common for both the editions of those days will be evaluated in both exercises, they won't be differentiated.

We can find cases in which some editions have been evaluated twice by the same student, the chances of this event increases with a large amount of **evaluations per edition**, with a low amount is unusual but not impossible.

If the number of **evaluations per edition** is not divisor of the number of **evaluations per student** it will generate some editions with less evaluations than expected in order to no generate more **evaluations per student** than the settled in the configuration.

Example: if we define 10 **evaluations per student** and 3 **evaluations per edition** due to the algorithm selected it will cause that some editions get only 1 evaluation (10/3=3, in this case n=3), but we will never make students to do more than 10 evaluations.

Figura 4.26: Pantalla de ayuda extra.



# Capítulo 5

## Diseño del Sistema

### 5.1. Arquitectura del Sistema

La estructura del sistema esta basada en el esquema de modelo vista controlador, introducidos en [CodeIgniter](#) ademas de una base de datos propia y la base de datos del media wiki, de forma que encontraremos los archivos principales en las carpetas de vista, donde se genera la interfaz para el usuario, modelo, donde se encuentran las funciones y algoritmo para el funcionamiento interno, y controladores, donde hacemos que se comuniquen las vistas con los modelos.

#### 5.1.1. Arquitectura Física

Son necesarios para nuestro sistema un ordenador que actué de servidor (puede valer donde este montado el Media Wiki) y el ordenador propio de los usuarios, con los cuales interactuaran con el sistema.

En principio [AssessMediaWiki](#) esta pensado para ser independiente del sistema operativo, decisión que verse afectada en un futuro dependiendo de las necesidades de escalabilidad y las funciones que se quieran implementar.

#### 5.1.2. Arquitectura Lógica

La arquitectura del sistema se divide en las siguientes capas

**Modelo** En este grupo podemos encontrar todo el software que genera las funciones para obtener, generar datos e interactuar con las bases de datos.

**Vista** En este grupo podemos encontrar todo el software que genera la interfaz del usuario, siendo esta su funcionalidad exclusiva.

**Controlador** Aquí encontraremos el software que completa los campos de las vistas, comunicándolas con los modelos. Los controladores se encargan de mostrar los datos del sistema en las vistas, e interaccionar con los usuarios, ya sea modificando dichos datos o navegando entre las distintas vistas.

**Bases de datos** Nuestro sistema cuenta con dos bases de datos fundamentales, la base de datos del MediaWiki, donde se guarda toda la información de los alumnos y su aportación al Wiki, y la base de datos del AssessMediaWiki, donde almacenaremos la nueva información generada por la interacción de los usuarios.

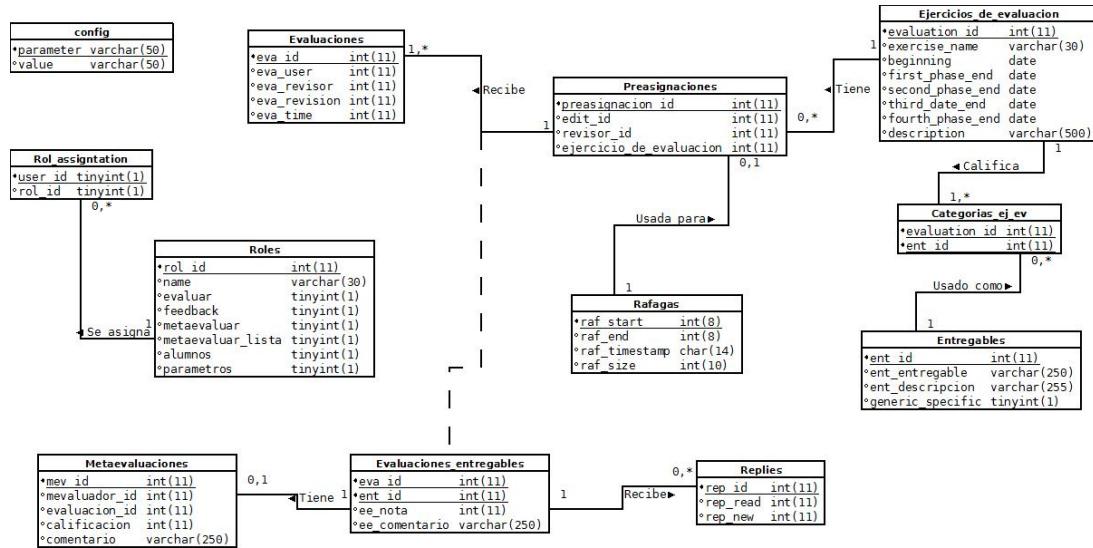


Figura 5.1: Diagrama de la base de datos de AMW 2.0.

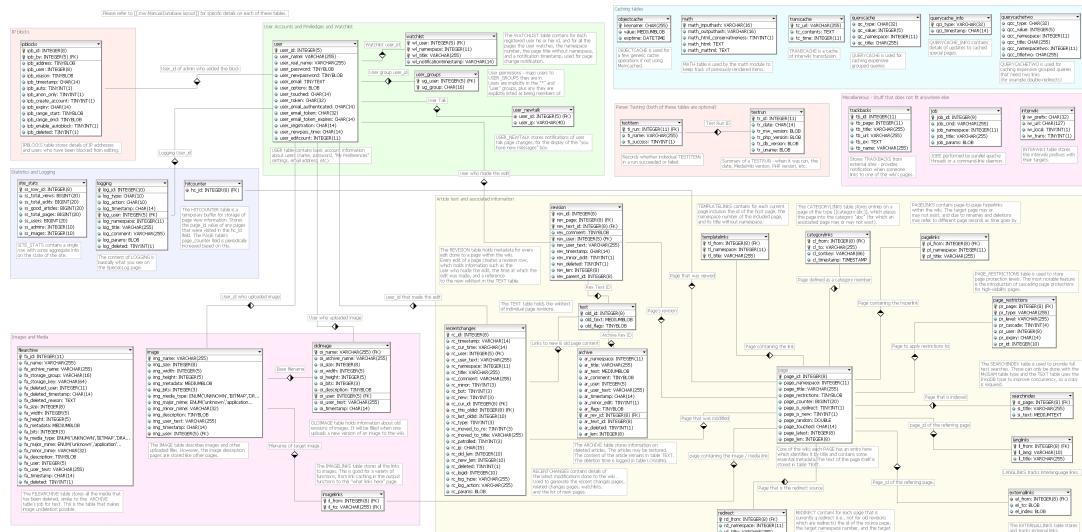


Figura 5.2: Diagrama de la base de datos de MediaWiki (disponible en la web de MediaWiki).

## 5.2. Parametrización del software base

Sera necesario editar el archivo de configuración para introducir los datos de nuestro MediaWiki, permitiendo así acceso a la base de datos y a generar las URLs para las evaluaciones (este proceso se explica detalladamente en el manual de instalación). También tenemos la opción de activar o desactivar el modo de desarrollo para hacer pruebas o cambios según se requiera.

## 5.3. Diseño Físico de Datos

En esta sección se define la estructura física de datos que utilizará el sistema, a partir del modelo de conceptual de clases, de manera que teniendo presente los requisitos establecidos para el sistema de información y las particularidades del entorno tecnológico, se consiga un acceso eficiente de los datos. La estructura física se compone de tablas, índices, procedimientos almacenados, secuencias y otros elementos dependientes del SGBD a utilizar.

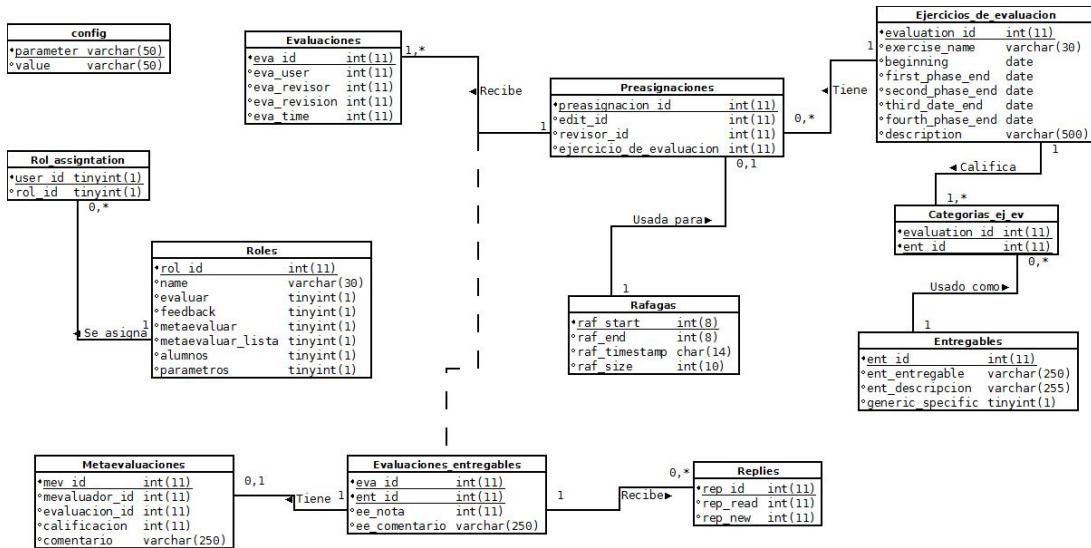


Figura 5.3: Diagrama de la base de datos de AMW 2.0.



# Capítulo 6

# Construcción del Sistema

Este capítulo trata sobre todos los aspectos relacionados con la implementación del sistema en código, haciendo uso de un determinado entorno tecnológico.

## 6.1. Entorno de Construcción

En esta sección se debe indicar el marco tecnológico utilizado para la construcción del sistema: entorno de desarrollo (IDE), lenguaje de programación, herramientas de ayuda a la construcción y despliegue, control de versiones, repositorio de componentes, integración continua, etc.

Para desarrollar este proyecto hemos usado:

- CodeIgniter [<http://www.codeigniter.com/>]
- PHP [<https://secure.php.net/>]
- HTML [<https://es.wikipedia.org/wiki/HTML>]
- MYSQL [<https://www.mysql.com/>]
- Forja Rediris [<https://forja.rediris.es/>]
- Tortoise SVN [<http://tortoisessvn.net/>]
- Firefox [<https://www.mozilla.org/es-ES/>]
- LaTeX [<http://www.latex-project.org/>]
- TeXstudio [<http://www.texstudio.org/>]
- Dia [<http://dia-installer.de/index.html.es>]

## 6.2. Código Fuente

Organización del código fuente, describiendo la utilidad de los diferentes ficheros y su distribución en paquetes o directorios. Asimismo, se incluirá algún extracto significativo de código fuente que sea de interés para ilustrar algún algoritmo o funcionalidad específica del sistema.

Se ha usado una estructura modelo vista controlador, de forma que al analizar inicialmente el sistema, pude ver que en la carpeta de vista guardaremos las interfaces, en la de modelos guardaremos los objetos donde se definen todas las funciones y en la carpeta de controladores guardaremos los controladores, que hacen posible que vistas y modelos interactúen entre si.

También me gustaría destacar el algoritmo de barajado aleatorio usado para la preasignación de ediciones a los alumnos.

Fuente: [Wikipedia: Algoritmo Fisher-Yates](#)

Otra cosa a destacar es que con el antiguo esquema de la base de datos, al realizarse una re-evaluación, esta no era tratada de forma especial, y contaba como si fuese una evaluación normal de una edición. Gracias a la nueva distribución de la base de datos se ha solventado ese problema, manteniendo a su vez la compatibilidad con otros sistemas como StatMediaWiki o CleverFigures

### 6.3. Scripts de Base de datos

El script para la creación de la base de datos, así como para introducir los datos y valores iniciales es el siguiente:

—  
—  
— Estructura de tabla para la tabla ‘config’  
—

```
CREATE TABLE IF NOT EXISTS ‘config‘ (  
‘parameter‘ varchar(50) NOT NULL,  
‘value‘ varchar(50) NOT NULL,  
PRIMARY KEY (‘parameter‘)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

—  
— Estructura de tabla para la tabla ‘entregables’  
—

```
CREATE TABLE IF NOT EXISTS ‘entregables‘ (  
‘ent_id‘ int(11) NOT NULL AUTO_INCREMENT,  
‘ent_entregable‘ varchar(250) NOT NULL,  
‘ent_description‘ varchar(255) NOT NULL,  
‘generic_specific‘ boolean NOT NULL DEFAULT false, -0 = generic, 1 = specific  
PRIMARY KEY (‘ent_id‘)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;
```

—  
— Estructura de tabla para la tabla ‘evaluaciones’  
—

```
CREATE TABLE IF NOT EXISTS ‘evaluaciones‘ (  
‘eva_id‘ int(11) NOT NULL AUTO_INCREMENT,  
‘eva_user‘ int(11) NOT NULL,  
‘eva_revisor‘ int(11) NOT NULL,  
‘eva_revision‘ int(11) NOT NULL,  
‘eva_time‘ int(11) NOT NULL,  
PRIMARY KEY (‘eva_id‘)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;
```

---

—  
— Estructura de tabla para la tabla ‘evaluaciones\_entregables’  
—

```
CREATE TABLE IF NOT EXISTS ‘evaluaciones_entregables‘ (  
‘eva_id‘ int(11) NOT NULL,  
‘ent_id‘ int(11) NOT NULL,  
‘ee_nota‘ int(11) NOT NULL,  
‘ee_comentario‘ varchar(250) NOT NULL,  
PRIMARY KEY (‘eva_id‘,‘ent_id‘)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

---

—  
— Estructura de tabla para la tabla ‘replies’  
—

```
CREATE TABLE IF NOT EXISTS ‘replies‘ (  
‘rep_id‘ int(11) NOT NULL AUTO_INCREMENT,  
‘rep_read‘ int(11) NOT NULL,  
‘rep_new‘ int(11) NOT NULL,  
PRIMARY KEY (‘rep_id‘)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;
```

---

—  
— Estructura de tabla para la tabla ‘metaevaluaciones’  
—

```
CREATE TABLE IF NOT EXISTS ‘metaevaluaciones‘ (
```

```
'mev_id' int(11) NOT NULL AUTO_INCREMENT,  
'mevaluador_id' int(11) NOT NULL,  
'evaluacion_id' int(11) NOT NULL,  
'calificacion' int(11) NOT NULL,  
'comentario' varchar(250) NOT NULL,  
PRIMARY KEY ('mev_id')  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;
```

---

– Estructura de tabla para la tabla ‘roles’

```
CREATE TABLE IF NOT EXISTS ‘roles‘ (  
'rol_id' int(11) NOT NULL AUTO_INCREMENT,  
'name' varchar(30) NOT NULL UNIQUE,  
'evaluar' boolean NOT NULL DEFAULT true,  
'feedback' boolean NOT NULL DEFAULT true,  
'metaevaluar' boolean NOT NULL DEFAULT false,  
'metaevaluar_lista' boolean NOT NULL DEFAULT false,  
'alumnos' boolean NOT NULL DEFAULT false,  
'parametros' boolean NOT NULL DEFAULT false,  
PRIMARY KEY ('rol_id')  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;
```

---

– Estructura de tabla para la tabla ‘rol\_assignment’

```
CREATE TABLE IF NOT EXISTS ‘rol_assignment‘ (  
'user_id' int(11) NOT NULL,  
'rol_id' int(11) NOT NULL,  
PRIMARY KEY ('user_id')  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;
```

---

– Estructura de tabla para la tabla ‘ejercicios\_de\_evaluacion’

```
CREATE TABLE IF NOT EXISTS ‘ejercicios_de_evaluacion‘ (  
'evaluation_id' int(11) NOT NULL AUTO_INCREMENT,  
'exercise_name' varchar(30) NOT NULL UNIQUE,
```

```
'beginning' date NOT NULL,  
'first_phase_end' date NOT NULL,  
'second_phase_end' date NOT NULL,  
'third_phase_end' date NOT NULL,  
'fourth_phase_end' date NOT NULL,  
'description' varchar(500) NOT NULL,  
PRIMARY KEY ('evaluation_id')  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;
```

---

—  
— Estructura de tabla para la tabla ‘preasignaciones’  
—

```
CREATE TABLE IF NOT EXISTS ‘preasignaciones’ (  
'preasignacion_id' int(11) NOT NULL AUTO_INCREMENT,  
'edit_id' int(11) NOT NULL,  
'revisor_id' int(11) NOT NULL,  
'ejercicio_de_evaluacion' int(11) NOT NULL,  
PRIMARY KEY ('preasignacion_id')  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;
```

---

—  
— Estructura de tabla para la tabla ‘categorias\_ej\_ev’  
—

```
CREATE TABLE IF NOT EXISTS ‘categorias_ej_ev’ (  
'evaluation_id' int(11) NOT NULL,  
'ent_id' int(11) NOT NULL,  
PRIMARY KEY ('evaluation_id','ent_id')  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;
```

---

—  
— Estructura de tabla para la tabla ‘rafagas’  
—

```
CREATE TABLE IF NOT EXISTS ‘rafagas’ (  
'raf_start' int(8) NOT NULL,  
'raf_end' int(8) NOT NULL,  
'raf_timestamp' char(14) NOT NULL,  
'raf_size' int(10) NOT NULL,  
PRIMARY KEY ('raf_start')
```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

---

- 
- Insercion de valores por defecto
  - A la vez que creamos la tabla añadimos los dos primeros usuarios, esto ha de hacerse como ultima accion
  - ya que si esta creado dara error y no se crearan las siguientes tablas, y una vez añadidos los usuarios
  - añadimos al primer usuario creado en la wiki como administrador.

— #TODO evitar ese error

```
INSERT INTO `roles`(`name`, `evaluar`, `feedback`, `metaevaluar`, `metaevaluar_lista`, `alumnos`,  
`parametros`)  
VALUES ('Admin',1,1,1,1,1);  
INSERT INTO `roles`(`name`, `evaluar`, `feedback`)  
VALUES ("Student",1,1);  
INSERT INTO `rol_assignment`(`user_id`, `rol_id`)  
VALUES (1,1);
```

---

La mayoría de funciones para manejar las bases de datos se encuentran en los modelos del sistema.

Como se ha mencionado anteriormente, al modificar la estructura de la base de datos del sistema (ver [Fig.6.1] y [Fig.6.2] a continuación) se ha conseguido solventar algunas debilidades de la versión anterior, así como implementar mejoras como las ráfagas y las preasignaciones, para las cuales es necesario que las bases de datos de AssessMediaWiki y MediaWiki se intercomuniquen, ese proceso se puede observar con mas detenimiento en el modelo de ejercicios de evaluación.

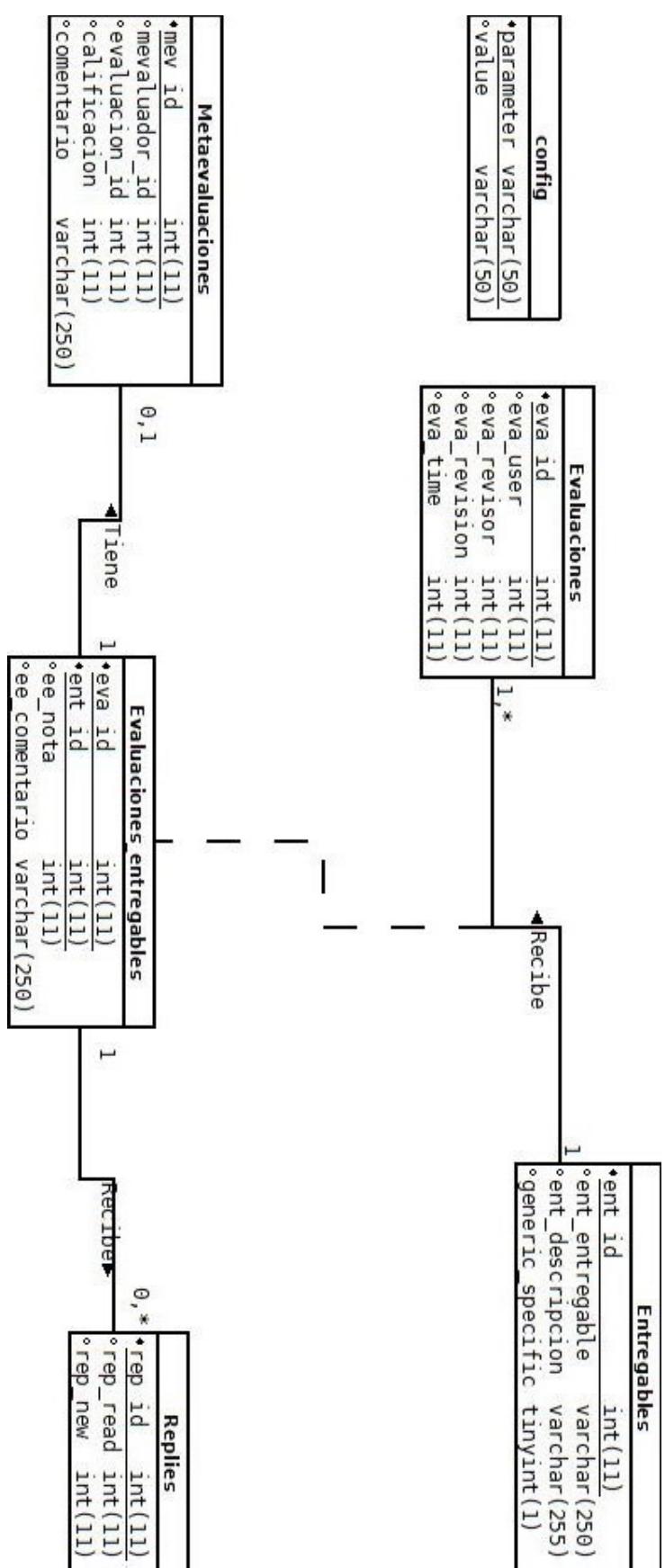


Figura 6.1: Diagrama de la base de datos de AMW 1.0.

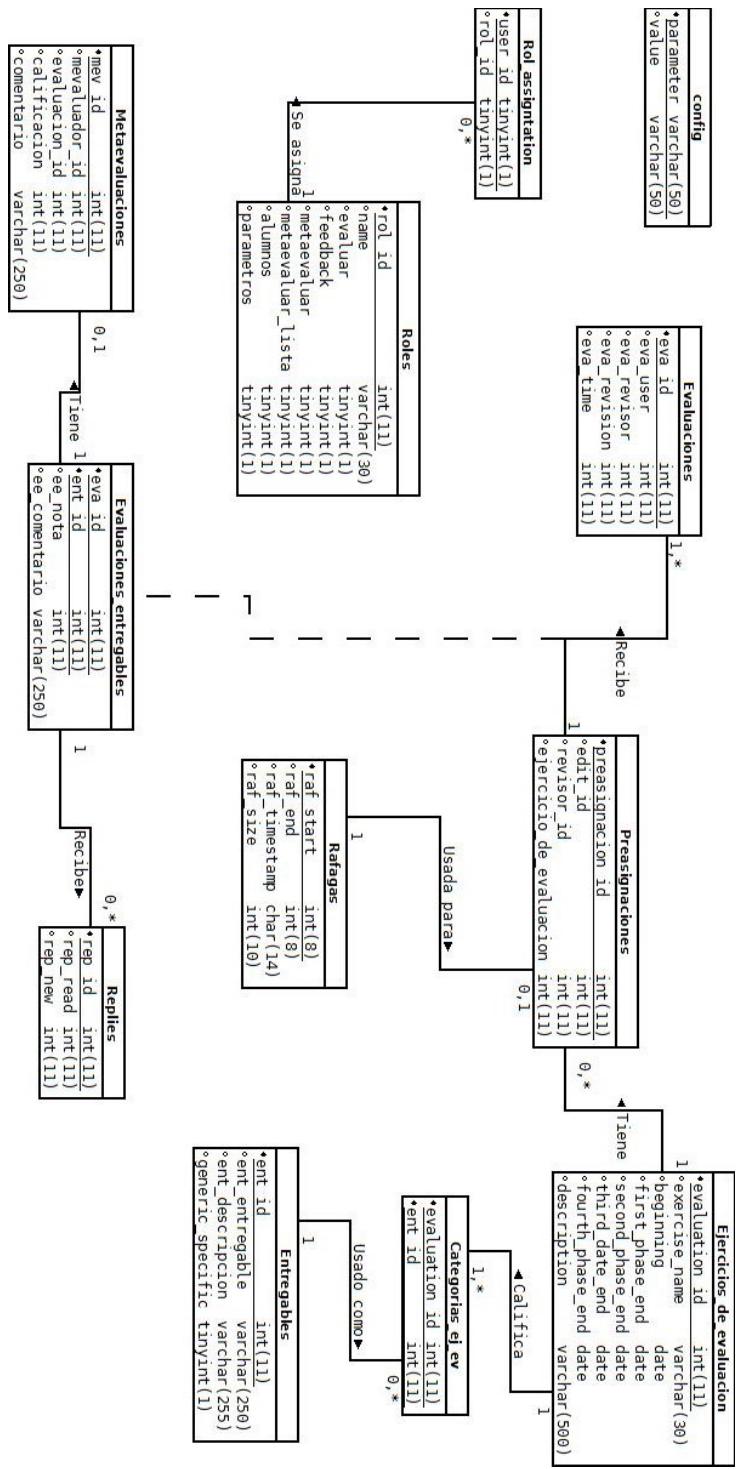


Figura 6.2: Diagrama de la base de datos de AMW 2.0.

# Capítulo 7

## Pruebas del Sistema

### 7.1. Estrategia

Para realizar las pruebas de sistema se ha creado un MediaWiki local con varios usuarios y ediciones.

### 7.2. Entorno de Pruebas

Para realizar pruebas en el sistema es necesario al menos un MediaWiki con usuarios y ediciones y que se haya creado la propia base de datos del sistema.

### 7.3. Roles

Describir en esa sección cuáles serán los perfiles y participantes necesarios para la ejecución de cada uno de los niveles de prueba.

Los roles de pruebas básicos son el de Estudiante y Profesor, pero dado que se pueden crear roles personalizables en el sistema es imposible hacer un listado total de los roles.

### 7.4. Niveles de Pruebas

En este sección se documentan los diferentes tipos de pruebas que se han llevado a cabo, ya sean manuales o automatizadas mediante algún software específico de pruebas.

Todas las pruebas de inserción de datos han sido realizadas manualmente, así como la corrección de ediciones y la creación y edición tanto de roles como de ejercicios de evaluación

#### 7.4.1. Pruebas de Sistema

Se creo un apartado: "sala de depurado (debug room)" para realizar pruebas preliminares introduciendo datos manualmente en el código en vez de hacerlas con la interfaz, lo cual agilizo mucho el proceso de realización de pruebas, permitiendo hacer varias pruebas de forma simultanea en el sistema.

#### **7.4.2. Pruebas de Aceptación**

El objetivo de estas pruebas es demostrar que el producto está listo para el paso a producción. Suelen ser las mismas pruebas que se realizaron anteriormente pero en el entorno de producción. En estas pruebas, es importante la participación del cliente final.

Al realizar las pruebas con la interfaz se vieron los mismos errores que en la sala de depurado, principalmente de comunicación con las bases de datos, una vez solventados estos problemas en la sala de depurado se comprobó que también hubiesen sido solventados en la interfaz del usuario.

## Parte II

### Epílogo



En esta última parte quedarán recogidas las conclusiones y los manuales necesarios para el manejo de la aplicación resultado del desarrollo. Si se ha realizado algún tipo de evaluación de la solución proporcionada, más allá de las pruebas del sistema, también deberá venir recogida en un capítulo separado dentro de esta parte. Pueden consultarse diversos tipos de evaluaciones sobre sistemas de información en [Hevner et al., 2004]: casos de estudio, análisis estático, análisis dinámico, simulación, experimento controlado, etc.



## Capítulo 8

# Manual de implantación y explotación

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

### 8.1. Introducción

AssessMediaWiki es una aplicación web de código abierto que, al conectarse a una instalación MediaWiki, proporciona procedimientos de autoevaluación, hetero evaluación y evaluación entre iguales, a la vez que mantiene información sobre esas evaluaciones. Los supervisores pueden obtener informes que ayudan en la evaluación de los estudiantes.

Aunque hay un gran número de extensiones para el sistema MediaWiki, no hemos encontrado ninguna que permitiera evaluar contribuciones individuales a un wiki. La mayoría de las aproximaciones solo ofrecen formas de evaluar una versión en particular de un artículo (normalmente la más reciente), siendo ineficaces en este caso. Por ello, para evaluar la calidad de las contribuciones creamos AssessMediaWiki.

AssessMediaWiki implementa como base dos roles de usuario distintos: supervisores y estudiantes. Los estudiantes pueden elegir entre distintas opciones: evaluar una revisión, comprobar sus propias aportaciones evaluadas y verificar las evaluaciones ya enviadas. Por otro lado, los supervisores tienen un mayor número de opciones, como modificar los parámetros de los programas o vigilar las evaluaciones que los alumnos vayan haciendo.

### 8.2. Requisitos previos

Para usar AssessMediaWiki es necesaria la previa presencia de un MediaWiki donde los alumnos vayan a trabajar

### 8.3. Inventario de componentes

Lista de los componentes hardware y software que se incluyen en la versión del producto:

- AssessMediaWiki 2.0
- Manual de instalación.

- Ayuda al usuario.

## 8.4. Procedimientos de instalación

### Descarga e instalación

AssessMediaWiki se puede descargar desde su [web oficial](#), dentro de la pestaña de [descargas](#). El contenido del archivo se debe descomprimir en la carpeta que deseé del servidor web (que permita ejecutar ficheros de lenguaje PHP). Por ejemplo, en el caso de los paquetes Xampp se denomina htdocs, y en otras instalaciones de Apache www.

### Configuración previa

Tras instalar AssesMediaWiki en nuestro equipo tendremos que entrar en los siguientes ficheros para hacer las siguientes modificaciones para que funcione adecuadamente:

htcdocs/assesmediawiki/applications/config/amw.php:

*config["database<sub>mw</sub>*

*config["username<sub>mw</sub>*

*config["password<sub>mw</sub>*

*config["usuariosadmin"] = array(1,2); en la array indicamos cuales usuarios de la wiki serán los administradores del AssesMediaWiki, en este caso serian el primero y el segundo en registrarse, los cuales saldrán por ese orden en la base de datos del wiki*

htcdocs/assesmediawiki/applications/config/database.php:

*db['default']['hostname'] = 'localhost'; localhost será el servidor donde se encuentran ubicadas las bases de datos, en caso de estar en el propio equipo lo dejaremos tal cual, en caso contrario lo cambiaremos por la dirección del servidor.*

*config["username<sub>mw</sub>*

*config["password<sub>mw</sub>*

*db['default']['database'] =' amw'; amw sería la base de datos propia de AssesMediaWiki, generada por el mismo programa. (En caso de que el AssesMediaWiki no cree su propia base de datos podemos generarla con la sentencia `mysql nombrebasededatos < fichero de texto`, pudiendo encontrar el fichero de texto en: `htcdocs/assesmediawiki/application/sql/estructura_amw` y siendo en este caso `iamw` el nombre de la base de datos).*

### Configuración

Tras haber configurado dichos archivos ejecutaremos el sistema gestor de bases de datos y el servidor web (en este orden). Por ejemplo, en el panel del control de Xampp o usando los scripts de /etc/init.d (o los comandos start, stop, reload y restart que lo sustituyen en “upstart”).

Tras hacer entrar al sistema como usuario administrador iremos a la pestaña de “Parameters”, donde modificaremos los siguientes datos:

Start date: fecha de inicio de evaluación de las entradas en la wiki.

End date: fecha de fin de evaluación de las entradas en la wiki.

Evals per student: numero de entradas que evaluara cada estudiante.

Meta-vals per student: numero de evaluaciones que evaluara cada metaevaluador.

Wiki URL: dirección URL del wiki en el que usaremos el AssesMediaWiki, es muy importante poner bien la dirección URL, ya que sin esta el AssesMediaWiki no puede funcionar correctamente.

## 8.5. Pruebas de implantación

Sería recomendable probar a crear y editar algún rol y ejercicio de evaluación y dentro del MediaWiki algún usuario y una edición de prueba para intentar corregirla.

## 8.6. Procedimientos de operación y nivel de servicio

Procedimientos necesarios para asegurar el correcto funcionamiento, rendimiento, disponibilidad y seguridad del sistema: back-ups, chequeo de logs, etc. También, es preciso indicar claramente aquellas actuaciones precisas necesarias para el mantenimiento preventivo del sistema y así prevenir posibles fallos en el mismo.

Para asegurar el correcto funcionamiento sera necesario tener el servidor siempre encendido y con conexión a internet, para que así sea posible acceder al sistema en cualquier momento y desde cualquier lugar.

Asimismo se recomienda la creación de una copia de seguridad periódicamente, para prevenir la perdida de datos ante algún error desconocido o problema con el servidor.



# Capítulo 9

## Manual de usuario

### 9.1. Introducción

AssessMediaWiki es una aplicación web de código abierto que, al conectarse a una instalación MediaWiki, proporciona procedimientos de autoevaluación, hetero evaluación y evaluación entre iguales, a la vez que mantiene información sobre esas evaluaciones. Los supervisores pueden obtener informes que ayudan en la evaluación de los estudiantes.

Aunque hay un gran número de extensiones para el sistema MediaWiki, no hemos encontrado ninguna que permitiera evaluar contribuciones individuales a un wiki. La mayoría de las aproximaciones solo ofrecen formas de evaluar una versión en particular de un artículo (normalmente la más reciente), siendo ineficaces en este caso. Por ello, para evaluar la calidad de las contribuciones creamos AssessMediaWiki.

AssessMediaWiki implementa como base dos roles de usuario distintos: supervisores y estudiantes. Los estudiantes pueden elegir entre distintas opciones: evaluar una revisión, comprobar sus propias aportaciones evaluadas y verificar las evaluaciones ya enviadas. Por otro lado, los supervisores tienen un mayor número de opciones, como modificar los parámetros de los programas o vigilar las evaluaciones que los alumnos vayan haciendo.

### 9.2. Características

Las principales funcionalidades del sistema son:

- Configurar ejercicios de evaluación.
- Evaluar ediciones al azar (dentro de las mas significativas).
- Generar CSV.

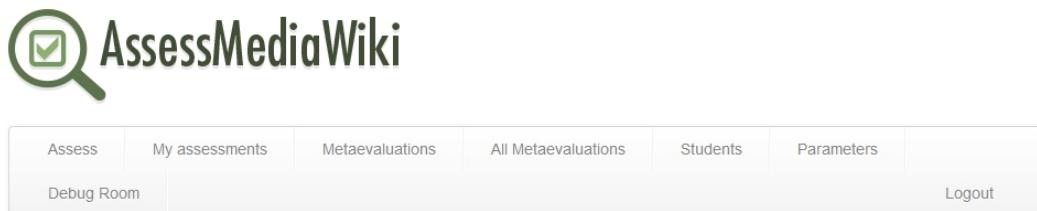
### 9.3. Requisitos previos

EL requisito principal es que debe haber un MediaWiki para que los alumnos trabajen sobre el.

## 9.4. Uso del sistema

Lo único necesario para poder usar el sistema es estar logeado en el MediaWiki existente, tras eso tan solo hay que seguir el manual de instalación y una vez se haya configurado el sistema podemos proceder a realizar pruebas del mismo, creando usuarios en el MediaWiki y realizando ediciones para posteriormente evaluarlas.

Si fuese necesario cualquier ayuda adicional, al final la sección de parámetros del sistema se encuentra un enlace a una sección de ayuda para el usuario, con información detallada de las opciones configurables del sistema con las que el docente deberá trabajar.



The **evaluations per student** indicates how many editions will evaluate each student, it's the main function of AssessMediaWiki.

The **metaevaluations** are evaluations of the evaluation of an edition made by another student, to allow metaevaluations they must be set in more than 0 and we have to define a role for metaevaluator students, allowing them to access to the metaevaluations section.

The amount of **evaluations per edition** indicates how many times one edition will be evaluated for several students and cannot be lower than 1, each edition has to receive at least one evaluation.

The **minimum of editions evaluated per student** indicates the amount of editions that would be evaluated for each student (always the most significant) after that minimum it will take the most significant one that hasn't been selected yet, so if the minimum is 2 we can find students with 2 editions evaluated and students with 8 editions evaluated.

The only case in which we will find people with less editions evaluated than the specified value is the case that the student hasn't reached the minimum number of editions (students with one edition for example).

The **autoevaluation** allows students to evaluate their own editions, the only available values are 0 (off) and 1 (on).

Students for the evaluations of the **evaluation exercises** will be selected automatically, the only requisite is to have done any contribution to the wiki in the **develop phase**. If some student doesn't make any change during the **develop phase**, it would be impossible to assign her/him any edition of another student to evaluate it.

### Important

If we configure two **evaluation exercises** in which the **develop phase** has days in common for both the editions of those days will be evaluated in both exercises, they won't be differentiated.

We can find cases in which some editions have been evaluated twice by the same student, the chances of this event increase with a large amount of **evaluations per edition**, with a low amount is unusual but not impossible.

If the number of **evaluations per edition** is not divisor of the number of **evaluations per student** it will generate some editions with less evaluations than expected in order to not generate more **evaluations per student** than the set in the configuration.

Example: if we define 10 **evaluations per student** and 3 **evaluations per edition** due to the algorithm selected it will cause that some editions get only 1 evaluation ( $10 \div 3 = 3$ , in this case  $n=3$ ), but we will never make students do more than 10 evaluations.

Figura 9.1: Pantalla de ayuda extra.

# Capítulo 10

## Conclusiones

### 10.1. Objetivos alcanzados

Este apartado debe resumir los objetivos generales y específicos alcanzados, relacionándolos con todo lo descrito en el capítulo de introducción.

Se ha podido ampliar las funcionalidades del sistema AssessMediaWiki, añadiendo mas herramientas para los docentes y mas funciones para los alumnos, con la posibilidad de darles responsabilidades personalizadas y haciendo que la experiencia docente sea mas interactiva entre ellos y con el profesor.

### 10.2. Lecciones aprendidas

A continuación, se detallan las buenas prácticas adquiridas, tanto tecnológicas como procedimentales, así como cualquier otro aspecto de interés.

Resumir cuantitativamente el tiempo y esfuerzo dedicados al proyecto a lo largo de su desarrollo que escribir un sencillo 'he trabajado mucho en este proyecto'. Tras unos nueve meses (a parte de experiencias previas) dedicándole a este proyecto una media de dos horas diarias (o al menos intentándolo) me he dado cuenta de que realmente lanzar una actualización o una nueva versión de algo lleva mucho tiempo y esfuerzo.

También he aprendido mucho sobre los framework, buscar recursos por internet y la ayuda de la comunidad en webs como StackOverflow

### 10.3. Trabajo futuro

En esta sección, se presentan las diversas áreas u oportunidades de mejora detectadas durante el desarrollo del proyecto y que podrán ser abarcadas en futuras versiones del software.

Como trabajo futuro quedan pendiente:

- Detección de mas wiki-comportamientos, como las ediciones de correcciones ortográficas y desplazamientos de texto.

- Realizar preasignaciones se realicen de forma automática (intentando mantener la independencia con el sistema operativo).
- Poder trabajar con AssessMediaWiki vía API.
- Integración como una extensión de MediaWiki.
- Botón de conflicto de interés / rechazar evaluación.

# Bibliografía

[CCOO, 2010] CCOO (2010). Tablas salariales 2010 IV Convenio Colectivo. <http://www.uca.es/sindicato/ccoo/documentos/tabla-salarial-pas-laboral-2010.pdf>.

[Hevner et al., 2004] Hevner, A. R., March, S. T., Park, J., and Ram, S. (2004). Design Science in Information Systems Research. *MIS Quarterly*, 28(1):75–105.



## Información sobre Licencia

AssessMediaWiki se desarrolla bajo los términos de la licencia GPL.

Este fichero es parte de la plantilla LaTeX para la realización de Proyectos Final de Grado, protegido bajo los términos de la licencia GFDL. Para más información, la licencia completa viene incluida en el fichero fdl-1.3.tex