# Actualización gráfica y funcional del proyecto “Rodolfo”.

**Estudiante:** David Guaita Tello
TFC-Technology Enhanced Learning
Ingeniería Técnica de Informática de Gestión.

**Consultor:** Manel Zaera Idiarte
30/05/2011

# 1 **Resumen**.

Lo que a continuación presentamos pretende ser una memoria descriptiva del trabajo realizado para mejorar la funcionalidad y aspecto gráfico del proyecto Rodolfo.

El proyecto Rodolfo es una herramienta online para ayudar en el proceso del aprendizaje de las matemáticas. Consiste pues, en un repositorio de fórmulas matemáticas y sus correspondientes locuciones para poder escuchar cómo se leen las mismas.

Al tratarse de una actualización partimos de una base que ya tiene ciertas funcionalidades. Y será nuestra tarea añadir más funcionalidades y mejorar las que ya están por lo que los objetivos del TFC serán los siguientes:

• Mejoras en la base de datos.

- Crea un workflow y los perfiles de usuario necesarios para la gestión del repositorio. Se definirán roles para los usuarios (visitante, usuario registrado, editor, administrador) que tendrán diferentes capacidades de acceso a la base de datos.

- Implementación de una taxonomía. Se añadirá la capacidad de etiquetar los objetos subidos al repositorio de acuerdo a una taxonomía predeterminada.

• Mejoras de aspectos visuales y multimedia de la aplicación.

- Integración de un editor visual. Se integrarán editores 'WYSIWYG' ya existentes decodificador LaTeX y MathML para facilitar la tarea de los usuarios.

- Implementación de un reproductor para los elementos de audio del repositorio.

En esta memoria mostramos los perfiles de usuarios (invitado, usuario, editor y administrador) que irán teniendo más privilegios en este mismo orden hasta llegar al administrador que tiene todos los privilegios. Este apartado requiere la creación de las tablas necesarias para contener a los usuarios, creación de interfaces para la administración de los mismos, creación de interfaces de “login” y realizar cambios en el código PHP existente para restringir las posibles acciones según privilegios de usuario.

La taxonomía la implementaremos a partir de la tabla existente con el mismo nombre, por lo que el trabajo en este punto consistirá en aplicar esta taxonomía a las fórmulas para que de forma obligatoria estén etiquetadas para pertenecer a un índice. Así mismo será necesaria la creación de las interfaces necesarias para administrar los índices.

Para la visualización y edición amigable de las fórmulas hemos optado por integrar el motor de visualización de fórmulas MathJax ya que este ha mostrado ser satisfactorio para los objetivos de este apartado. Esta solución permite que las fórmulas se introduzcan en código LaTeX y este mismo motor se encarga, a la hora de mostrar los datos, de ver la codificación tanto en LaTeX como en MathML.

Para la reproducción de locuciones Hemos optado por BS Sound, un Script muy ligero en JavaScript que permite transformar enlaces a un archivo de tipo MP3 en un reproductor sencillo. De esta forma los usuarios no tienen que descargar ni abrir el archivo fuera del navegador.

Las soluciones presentadas han completado con éxito los objetivos de TFC de una forma sencilla y utilizando código libre.

Es cierto que las soluciones aportadas pueden necesitar otras mejoras en el futuro como es lógico en el desarrollo de software, pero entendemos que las mejoras realizadas hasta el momento convierten el antiguo proyecto Rodolfo en una herramienta totalmente funcional.

# 2 Índice.
## 2.1 Índice de contenidos.


1 Resumen ..... 2
2 Índice ..... 4
2.1 Índice de Contenidos ..... 4
2.2 Índice de Ilustraciones ..... 5
2.3 Índice de Tablas ..... 5
3 Cuerpo de la Memoria ..... 6
3.1 Introducción ..... 6
3.1.1 Justificación del TFC y contexto en el que se desarrolla ..... 6
3.1.1.1 Estructura inicial de la Base de Datos ..... 7
3.1.1.1.1 Tabla “formulas”..... 7
3.1.1.1.1 Tabla “locucionsformules”..... 7
3.1.1.1.1 Tabla “taxonomia”..... 8
3.1.1.2 Funcionalidades iniciales ..... 8
3.1.1.2.1 Tratamiento de fórmulas ..... 8
3.1.1.2.1 Tratamiento de Locuciones ..... 8
3.1.1.2.1 Tratamiento de Taxonomía ..... 9
3.1.2 Objetivos del TFC ..... 9
3.1.3 Metodología seguida. ..... 9
3.1.4 Planificación del proyecto ..... 10
3.1.4.1 Tareas ..... 10
3.1.4.1.1 Fase 1: Preparación Inicial ..... 10
3.1.4.1.2 Fase 2: Reestructuración de la BBDD ..... 10
3.1.4.1.3 Fase 3: Reestructurar Formularios ..... 11
3.1.4.1.4 Fase 4: Reestructurar aspectos visuales y de accesibilidad a los datos ..... 11
3.1.4.1.5 Fase 5: Test ..... 11
3.1.4.1.6 Fase 6: Documentación y conclusiones finales ..... 11
3.1.4.2 Temporización ..... 11
3.1.4.2.1 Tabla ..... 12
3.1.4.2.2 Diagrama de Gantt ..... 13
3.2 Mejoras de la base de datos ..... 13
3.2.1 Sistema de Seguridad y privilegios ..... 13
3.2.1.1 Seguridad ..... 13
3.2.1.2 Script de Seguridad ..... 14
3.2.1.3 Tabla de Usuarios ..... 14
3.2.1.4 Privilegios pro tipo de usuario ..... 15
3.2.1.4.1 Usuarios de tipo Invitado ..... 15
3.2.1.4.2 Usuarios de tipo “Usuario básico o simple” ..... 19
3.2.1.4.3 Usuarios de tipo “Editor” ..... 20
3.2.1.4.4 Usuarios de tipo “Administrador” ..... 21
3.2.2 Workflow ..... 22
3.2.3 Taxonomía ..... 22
3.2.3.1 Cambios a nivel de Base de Datos ..... 22
3.2.3.2 Cambios a nivel de Código y Formularios ..... 23
3.2.4 Idiomas ..... 24
3.2.4.1 Cambios a nivel de Base de Datos ..... 24
3.2.4.2 Cambios a nivel de Código y Formularios ..... 24
3.3 Mejoras de aspectos visuales y multimedia ..... 25
3.3.1 Motor “MathJAX” ..... 25
3.3.1.1 Visualizador de fórmulas ..... 26
3.3.1.2 Editor de fórmulas ..... 27
3.3.1.3 Integración MathJax Alternativa ..... 29
3.3.2 Motor de reproducción de audio” ..... 30
3.3.2.1 Reproductor de locuciones ..... 30
3.3.2.1 Grabador de locuciones ..... 31

3.4 Conclusiones ... 31
4 Glosario ... 32
5 Bibliografía ... 34
6 Anexos ... 35
6.1 Requisitos del Sistema ... 35
6.2 Instrucciones de Instalación ... 35
6.2.1 Subida de Archivos por FTP y estructura de directorios ... 35
6.2.2 Archivos de configuración (Instrucciones) ... 35
6.2.3 Archivos de BBDD "*.sql" ... 36
6.2.4 Primer inicio de sesión y cambio de contraseña del Administrador ... 36
6.3 Esquemas del Workflow por tipos de Usuarios ... 37
6.3.1 Invitado ... 37
6.3.2 Usuario ... 38
6.3.3 Editor ... 39
6.3.4 Administrador ... 40

## 2.2 Índice de Ilustraciones.

Ilustración 1: Tabla formulas ... 7
Ilustración 2: Tabla de locuciones ... 8
Ilustración 3: Tabla de taxonomias ... 8
Ilustración 4: Diagrama de Gantt ... 13
Ilustración 5: Tabla de usuarios ... 14
Ilustración 6: Estado de la sesión ... 15
Ilustración 7: Inicio de sesión ... 15
Ilustración 8: Registro de usuario ... 16
Ilustración 9: Cambio de contraseña ... 16
Ilustración 10: Buscador de formulas por código LaTeX ... 16
Ilustración 11: Buscador por índice1 ... 17
Ilustración 12: Buscador por índice2 ... 17
Ilustración 13: Visualización de fórmulas ... 17
Ilustración 14: Visualización código LaTeX ... 17
Ilustración 15: Visualización de código MathML ... 18
Ilustración 16: Listado de louciones ... 18
Ilustración 17: Reproducción de locuniones ... 18
Ilustración 18: Añadir locuciones ... 19
Ilustración 19: Añadir fórmulas ... 19
Ilustración 20: Añadir índices ... 20
Ilustración 21: Panel del administrador ... 21
Ilustración 22: Tabla de fórmulas modificada ... 22
Ilustración 23: Inserción de índices ... 23
Ilustración 24: Eliminaión de índices ... 23
Ilustración 25: Búquedas por índices ... 23
Ilustración 26: Inserción de fórmulas con índices ... 24
Ilustración 27: Tabla de idiomas ... 24
Ilustración 28: Inserción de locuciones indicando el idioma ... 25
Ilustración 29: Editor de fórmulas amigable ... 29
Ilustración 30: Reproducción de audio ... 31
Ilustración 31: Workflow de "invitado" ... 37
Ilustración 32: Workflow de "usuario" ... 38
Ilustración 33: Workflow de "editor" ... 39
Ilustración 34: Workflow de "administrador" ... 40

## 2.3 Índice de Tablas.

Tabla 1: Temporización del proyecto ... 12

# 3 Cuerpo de la memoria.

## 3.1 Introducción.

El proyecto que estamos tratando es la continuación de un proyecto anterior y por lo tanto el objetivo de esta continuación es llevar a cabo una serie de mejoras y actualizaciones con respecto al anterior.

Así pues vamos a mejorar por un lado los aspectos estructurales de base de datos y la seguridad en el acceso a la misma y por otro lado vamos a mejorar los aspectos visuales, auditivos y de manejabilidad de la aplicación.

### 3.1.1 Justificación del TFC i contexto en el que se desarrolla.

Dado que este Trabajo Final de Carrera se basa en un proyecto anterior (Rodolfo), éste trata de aplicar unas mejoras en cuanto a sus funcionalidades y aspecto gráfico, por tanto, aceptaremos como válida la definición general y los objetivos presentados en la WEB del mismo proyecto (http://cimanet.uoc.edu/rdlf).

*"¿Qué es RODOLFO? Rodolfo es un repositorio de locuciones de fórmulas matemáticas, que nace de la necesidad de explicar a los estudiantes no presenciales el modo en que puede leerse una fórmula en voz alta.*

*El repositorio está dirigido a los profesores y diseñadores de contenidos docentes de matemáticas en formato digital.*

*Estos usuarios tendrán libre acceso para añadir nuevas fórmulas y locuciones, que luego podrán reutilizar fácilmente.*

*Rodolfo es multilingüe y de construcción colaborativa. Son sus usuarios los que deben hacer crecer con sus aportaciones, siempre bajo la supervisión de un administrador que garantizará la calidad.*

*El diseño del repositorio se ha basado en estándares. Para la transcripción de las fórmulas se ha elegido LaTeX. Además, y pensando en la integración en materiales web, se ha añadido una transcripción en MathML. Las fórmulas se describen con una versión adaptada y ampliada de la "Core Subject Taxonomy for Mathematical Sciences Education", que facilitará la clasificación y recuperación. Los ficheros de audio se graban en mp3.*

*Además de la transcripción en LaTeX y MathML, las fórmulas van acompañadas de la transcripción en texto en el idioma correspondiente. Una misma fórmula puede ir acompañada de varias locuciones en un mismo idioma, todas ellas correctas pero que se adaptan a diferentes estilos de lectura.*

*Se ha seguido por tanto el ideal de los contenidos abiertos y de la construcción colaborativa de los recursos docentes en la Internet, que son dos de los valores que el uso de las TIC ha incorporado a la enseñanza en entornos virtuales de aprendizaje.*

*Este proyecto se ha realizado en el marco de la primera convocatoria para proyectos de innovación docente del Vicerrectorado de innovación de la UOC."*

#### 3.1.1.1 Estructura inicial de la Base de Datos.

##### 3.1.1.1.1 Tabla “formulas”.

Contiene los datos de las fórmulas propiamente dichas incluyendo:

IdFormula: Campo auto numérico para identificación inequívoca de fórmulas.

LaTex: Campo que contiene la fórmula en codificación LaTex.

MathMLC: Campo que contiene la fórmula en codificación MathMLC (No utilizado).

MathMLP: Campo que contiene la fórmula en codificación MathMLP.

OpenMath: Campo que contiene la fórmula en codificación OpenMath.

| Name | Type | Length | Decimals | Allow Null |
|---|---|---|---|---|
| IdFormula | bigint | 20 | 0 | ☐ |
| LaTeX | text | 0 | 0 | ☑ |
| MathMLC | text | 0 | 0 | ☑ |
| MathMLP | text | 0 | 0 | ☑ |
| OpenMath | text | 0 | 0 | ☑ |

Ilustración 1:Tabla formulas.

##### 3.1.1.1.2 Tabla “locucionsformules”.

Contiene los datos de las locuciones de las fórmulas y las transcripciones de las mismas indicando el idioma en el que se introdujeron.

IdLocFormula: Campo auto numérico para identificación inequívoca de locuciones.

IdFormula: Campo que contiene el identificador del la fórmula a la que pertenece la locución.

Idioma: Campo que contiene el código del idioma en 2 caracteres para identificar en qué idioma está la locución.

Transcripcio: Campo que contiene la transcripción de la locución.

Arxiu: Contiene el archivo de la locución en formato .mp3.

Tipusarxiu: Contiene un identificador del tipo de archivo subido. (No está en uso)

TamanyArxiu: Contiene un indicador del tamaño de archivo subido. (No está en uso)

| Name | Type | Length | Decimals | Allow Null |
|---|---|---|---|---|
| IdLocFormula | bigint | 20 | 0 | ☐ |
| IdFormula | bigint | 20 | 0 | ☐ |
| Idioma | char | 2 | 0 | ☐ |
| Transcripcio | text | 0 | 0 | ☐ |
| Arxiu | mediumblob | 0 | 0 | ☑ |
| TipusArxiu | varchar | 30 | 0 | ☑ |
| TamanyArxiu | int | 11 | 0 | ☑ |

Ilustración 2: Tabla de locuciones.

#### 3.1.1.1.3 Tabla "taxonomia".

Contiene los datos propios de la clasificación por taxonomía. (No está en uso)

Index: Contiene la numeración de índice para la cosificación.

Name: contiene el nombre al que corresponde el índice.

| Name | Type | Length | Decimals | Allow Null |
|---|---|---|---|---|
| Index | varchar | 14 | 0 | ☐ |
| Name | varchar | 255 | 0 | ☐ |

Ilustración 3: Tabla de taxonomias.

### 3.1.1.2 Funcionalidades iniciales.

No encontramos un workflow definido y por tanto la aplicación da impresión de estar desorganizada y ser poco amigable con el usuario.

#### 3.1.1.2.1 Tratamiento de fórmulas.

- Se ha de introducir como texto plano para todos los tipos de codificación, cosa que dificulta la interacción del usuario.
- No tenemos la posibilidad de añadir una locución de forma directa a la fórmula que estamos dando de alta.
- No podemos editar fórmulas.
- No podemos eliminar fórmulas.

#### 3.1.1.2.2 Tratamiento de Locuciones.

- Hay que introducir el código del idioma como texto plano en lugar de elegir dicho idioma de un desplegable.
- Las locuciones no tienen un sistema propio e integrado a la Web para escucharlos.
- No podemos editar locuciones.

- No podemos eliminar locuciones.
-

#### 3.1.1.2.3 Tratamiento de Taxonomía.

- No se puede etiquetar una fórmula para que pertenezca a un índice.
- No podemos crear índices.
- No podemos eliminar índices.

### 3.1.2 Objetivos del TFC.

**• Mejoras en la base de datos**

- Creación de un Workflow que refleje las acciones lógicas que pueden realizar los usuarios a la hora de utilizar el repositorio de fórmulas y locuciones.
- Creación de perfiles de usuario necesarios para la gestión del repositorio. Se definirán roles para los usuarios (visitante, usuario registrado, editor, administrador) que tendrán diferentes capacidades de acceso a la base de datos.
- Implementación de una taxonomía. Se añadirá la capacidad de etiquetar los objetos subidos al repositorio de acuerdo a una taxonomía predeterminada.

**• Mejoras de aspectos visuales y multimedia de la aplicación**

- Integración de un editor visual.
- Se integrarán editores 'WYSIWYG' ya existentes decodificador LaTeX y MathML para facilitar la tarea de los usuarios.
- Implementación de un reproductor para los elementos de audio del repositorio.

Pretendemos que, para cumplir con estos objetivos, podamos aplicar código Open Source.

### 3.1.3 Metodología seguida.

El proyecto que nos ocupa pretende ser una actualización de un proyecto existente y por lo tanto no partimos de cero.

El primer paso para afrontar este proyecto consiste en realizar un análisis a fondo del proyecto base del que partimos.

Más adelante realizamos un análisis de posibles soluciones para cada una de las mejoras que pretendemos implementar.

Una vez elegidas las soluciones procederemos a aplicar cada solución propuesta teniendo siempre en cuenta la posibilidad de tener que retroceder y reconsiderar las posibles soluciones aplicadas.

### 3.1.4 Planificación del Proyecto.

Hemos planificado el proyecto dividiéndolo en tareas separadas por fases para poder afrontarlo de forma organizada.

#### 3.1.4.1 Tareas.

##### 3.1.4.1.1 Fase 1: Preparación Inicial.

- Análisis BBDD
- Análisis código PHP.
- Análisis de funcionalidad inicial.
- Análisis de soluciones.
- Programación de tareas.
- Documentación del plan de trabajo (PAC1)

##### 3.1.4.1.2 Fase 2: Reestructuración de la BBDD.

- Añadir Idiomas.
    - Creada una tabla “idiomas” que contiene un “ID” de idioma correspondiente a la codificación de idioma tipo “es = español” y “es= inglés”.
    - Esta tabla la usaremos también para obtener los datos para la elección de idioma a la hora de ver, editar o crear locuciones.

- Añadir Usuarios y Añadir Roles.
    - Creada una tabla “usuarios” que contiene los datos propios de un usuario y el nivel de privilegios del mismo.
    - Hemos puesto por defecto 5 niveles distintos de privilegios.

- Organización Taxonómica.
    - Creado un campo en la tabla “formulas” que contiene el ID del índice taxonómico de forma que podamos acceder a las fórmulas mediante esta tipo de índice.
    - Creado un campo en la tabla “taxonomía” que contiene el ID del índice taxonómico del que depende. De esta manera tendremos una organización de dependencia o categoría/subcategoria.

- Reorganizar Locuciones.
    - Modificada la Tabla de forma que guarda las rutas a los archivos mp3 de las locuciones en lugar de guardarlos de forma binaria en un campo de la Tabla.

#### 3.1.4.1.3 Fase 3: Reestructurar Formularios.

- Acciones permitidas según roles.
    - Creado un sistema de seguridad/login para exigir el inicio de sesión para según qué acciones.
    - El usuario de tipo Administrador podrá acceder a todo el contenido con el nivel más alto de permisos. Así pues, este usuario, podrá añadir, editar y eliminar tanto usuarios como fórmulas, taxonomías y locuciones.
    - El usuario de tipo Editor podrá añadir y modificar tanto fórmulas como locuciones.

- Añadir editores visuales WYSIWYG.
- Añadir organización taxonómica.
- Elección de idiomas con desplegables.
- Añadir acceso a locuciones desde formulario de fórmulas.
- Documentación del informe de evolución (PAC 2)

#### 3.1.4.1.4 Fase 4: Reestructurar aspectos visuales y de accesibilidad a los datos.

- Búsquedas amigables.
- Reproducción de locuciones integradas a la Web.
- Visualización de fórmulas amigables.
- Aplicación de un Workflow lógico y amigable.

#### 3.1.4.1.5 Fase 5: Test.

- Realización de los test necesarios para asegurar el buen funcionamiento de la aplicación.
- Documentación del informe de evolución (PAC 3)

#### 3.1.4.1.6 Fase 6: Documentación y conclusiones finales.

- Preparación de la documentación final.
- Preparación de la presentación.
- Entrega del proyecto. (Entrega final TFC).

### 3.1.4.2 Temporización.

Hemos procurado establecer la temporización dentro del calendario semestral de la asignatura. Por ello contemplamos las entregas parciales o informes que están repartidos en el mismo.

Como es normal la temporización es estimativa dado que siempre cabe la posibilidad de tener que alterar el tiempo que tomará realizar alguna tarea y el cambio de orden de las mismas.

#### 3.1.4.2.1 Tabla.

| Id | Nombre de tarea | Duración | Fecha inicio | Fecha fin |
|---|---|---|---|---|
| 1 | **PROYECTO** | 95,75 days? | 02/03/2011 16:00 | 10/06/2011 16:00 |
| 2 | **Fase 1: Preparación inicial.** | 17,25 days? | 02/03/2011 16:00 | 20/03/2011 16:00 |
| 3 | Análisis BBD | 4,25 days? | 02/03/2011 16:00 | 06/03/2011 14:00 |
| 4 | Análisis código PHP | 3 days? | 06/03/2011 11:00 | 08/03/2011 19:00 |
| 5 | Análisis de funcionalidad inicial | 4,75 days? | 09/03/2011 16:00 | 14/03/2011 20:00 |
| 6 | Análisis de soluciones | 5,75 days? | 13/03/2011 16:00 | 18/03/2011 18:00 |
| 7 | Programación de tareas | 3,5 days? | 17/03/2011 16:00 | 20/03/2011 15:00 |
| 8 | Documentación del plan de trabajo (PAC1) | 2 days? | 20/03/2011 8:00 | 20/03/2011 16:00 |
| 9 | **Fase 2: Reestructuración de la BBDD.** | 18 days? | 21/03/2011 16:00 | 08/04/2011 17:00 |
| 10 | Añadir Idiomas | 3,75 days? | 21/03/2011 16:00 | 24/03/2011 19:00 |
| 11 | Añadir Usuarios | 2,75 days? | 25/03/2011 15:00 | 27/03/2011 16:00 |
| 12 | Añadir Roles | 2,5 days? | 28/03/2011 16:00 | 30/03/2011 18:00 |
| 13 | Organización Taxonómica | 4 days? | 31/03/2011 16:00 | 04/04/2011 17:00 |
| 14 | Reorganizar Locuciones | 3,5 days? | 05/04/2011 16:00 | 08/04/2011 17:00 |
| 15 | **Fase 3: Reestructurar Formularios** | 19,25 days? | 10/04/2011 12:00 | 29/04/2011 18:00 |
| 16 | Acciones permitidas según roles | 3 days? | 10/04/2011 12:00 | 12/04/2011 20:00 |
| 17 | Documentación del  informe de evolución (PAC 2) | 1 day? | 13/04/2011 16:00 | 13/04/2011 20:00 |
| 18 | Añadir editores visuales WYSIWYG | 4 days? | 14/04/2011 16:00 | 18/04/2011 17:00 |
| 19 | Añadir organización taxonómica | 3,75 days? | 19/04/2011 16:00 | 22/04/2011 18:00 |
| 20 | Elección de idiomas con desplegables | 4 days? | 24/04/2011 12:00 | 27/04/2011 20:00 |
| 21 | Añadir acceso a lociones desde formulario de fórmulas | 1,75 days? | 28/04/2011 16:00 | 29/04/2011 18:00 |
| 22 | **Fase 4: Reestructurar aspectos visuales y de accesibilidad a los datos** | 18,5 days? | 01/05/2011 8:00 | 18/05/2011 20:00 |
| 23 | Búsquedas amigables | 4,75 days? | 01/05/2011 8:00 | 04/05/2011 19:00 |
| 24 | Reproducción de locuciones integradas a la Web | 5,75 days? | 05/05/2011 16:00 | 10/05/2011 20:00 |
| 25 | Visualización de fórmulas amigables | 5 days? | 11/05/2011 16:00 | 16/05/2011 17:00 |
| 26 | Aplicación de un Workflow lógico y amigable | 3 days? | 16/05/2011 16:00 | 18/05/2011 20:00 |
| 27 | **Fase 5: Testing y documentación** | 10,5 days? | 19/05/2011 16:00 | 29/05/2011 16:00 |
| 28 | Documentación preliminar TFC | 7,75 days? | 19/05/2011 16:00 | 26/05/2011 20:00 |
| 29 | Documentación del  informe de evolución (PAC 3) | 4,75 days? | 25/05/2011 16:00 | 29/05/2011 16:00 |
| 30 | **Fase 6: Documentación y conclusiones finales** | 10,25 days? | 30/05/2011 19:00 | 10/06/2011 16:00 |
| 31 | Preparación de la documentación final | 8 days? | 30/05/2011 19:00 | 07/06/2011 20:00 |
| 32 | Preparación de la presentación | 5,75 days? | 02/06/2011 16:00 | 07/06/2011 20:00 |
| 33 | Entrega del proyecto. (Entrega final TFC) | 2,25 days? | 08/06/2011 16:00 | 10/06/2011 16:00 |

Tabla 1: Temporización del proyecto.

### 3.1.4.2.2 Diagrama de Gantt

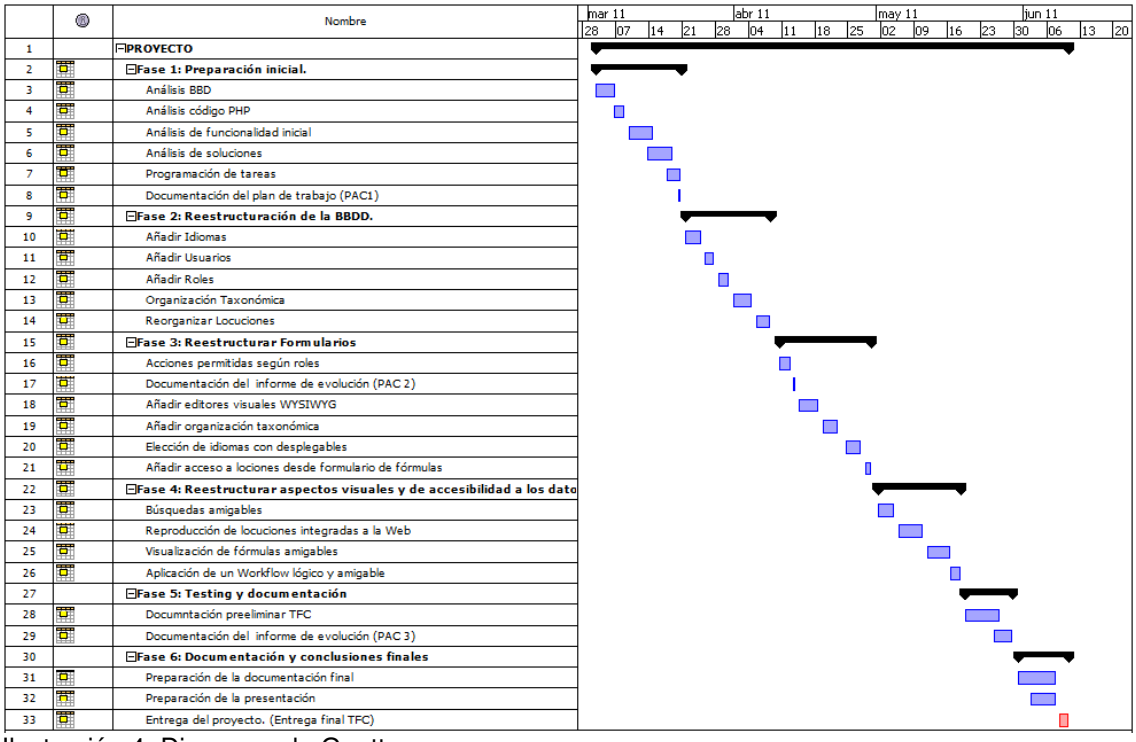

Ilustración 4: Diagrama de Gantt.

## 3.2 Mejoras de la base de datos.

En este apartado pretendemos mostrar todas las mejoras que principalmente necesitaban realizar cambios a nivel de base de datos. Esto implica la creación de un sistema de seguridad y un sistema para organizar las fórmulas de forma ordenada a través de una taxonomía.

### 3.2.1 Sistema de Seguridad y privilegios.

Esta es una parte fundamental en una aplicación web en la que se pretenda tener un repositorio de contenido y al mismo tiempo tener un control sobre qué usuarios podrán acceder al mismo o tener permisos de modificación y eliminación del contenido.

Como es normal se requerirá un panel de administración en el cual sólo puedan acceder los “super usuarios” para administrar los privilegios de los demás usuarios.

#### 3.2.1.1 Seguridad.

Para conseguir la integración de un sistema de seguridad de código abierto y a la vez sencillo, hemos optado por utilizar “PHP5 Login System v3.8.3”. Se trata de un sistema de seguridad en PHP creado por “Crisp Webdesigns”.

El sistema tiene la ventaja de sólo necesitar añadir una tabla a la base de datos para almacenar los usuarios y unas pocas líneas de código en los archivos actuales para poder restringir el acceso según los privilegios que tenga cada usuario.

Este sistema utiliza variables de sesión para almacenar el nombre de usuario y nivel de privilegios al iniciar sesión y los elimina al cerrar la misma.

La encriptación de las contraseñas almacenadas en la Base de Datos será en MD5.

#### 3.2.1.2 Script.

Para aplicar este sistema de seguridad a la aplicación web hemos optado por añadir unas líneas de código en el archivo "/connections/rodolfo.php", el cual, es llamado desde todas la páginas de nuestra aplicación.

```
include_once 'loginscript/include/processes.php';
$Login_Process = new Login_Process;
$Status = $Login_Process->welcome_note();
```

Estas líneas realizan una llamada al archivo principal del sistema de seguridad y luego hacen un chequeo del estado actual de la sesión.

Luego, en las secciones que queremos restringir según un perfil de usuarios, hemos de añadir lo siguiente:

- Permitir a usuarios registrado:

```
if($_SESSION['user_level'] >= 1) {    }
```

- Permitir a usuarios de tipo "Editor":

```
if($_SESSION['user_level'] >= 4) {    }
```

- Permitir a usuarios de tipo "Administrador":

```
if($_SESSION['user_level'] >= 5) {    }
```

#### 3.2.1.3 Tabla.

El sistema de seguridad requiere de una tabla nueva en nuestra B.D. para gestionar los usuarios y los privilegios de cada una de ellos.

| Name | Type | Length | Decimals | Allow Null | |
|---|---|---|---|---|---|
| ▶ userid | int | 25 | 0 | ☐ | 🔑 |
| first_name | varchar | 25 | 0 | ☐ | |
| last_name | varchar | 25 | 0 | ☐ | |
| email_address | varchar | 50 | 0 | ☐ | |
| username | varchar | 25 | 0 | ☐ | |
| password | varchar | 255 | 0 | ☐ | |
| info | varchar | 50 | 0 | ☐ | |
| last_loggedin | varchar | 100 | 0 | ☐ | |
| user_level | enum | 0 | 0 | ☐ | |
| forgot | varchar | 100 | 0 | ☑ | |
| status | enum | 0 | 0 | ☐ | |

Ilustración 5: Tabla de usuarios.

### 3.2.1.4 Privilegios por tipo de usuario.

Los privilegios de usuarios los hemos planteado de forma incremental, es decir, los usuarios de nivel superior heredan los privilegios inferiores y adquieren unos nuevos. De esta forma sucesiva hasta llegar al usuario de tipo "Administrador" que tiene todos los privilegios.

Los usuarios podrán ver en todo momento en qué estado se encuentra su sesión:

Ilustración 6: Estado de la sesión.

#### 3.2.1.4.1 Usuarios de tipo Invitado.

No se trata de un usuario registrado propiamente dicho pero así es como el sistema considerará a toda persona no registrada que acceda a la aplicación. Estos "usuarios" tendrán los siguientes derechos o privilegios:

- Iniciar Sesión: es la posibilidad de iniciar una sesión como un usuario del sistema propiamente dicho dentro del repositorio de usuarios almacenados en el sistema, eso sí, indicando siempre un nombre de usuario y una contraseña.

Ilustración 7: Inicio de sesión.

- Registrarse: es la posibilidad de crear una cuenta de usuario nueva, con lo cual tendremos más privilegios dentro de la aplicación. La cuenta de usuario que se crea por defecto será de tipo "Usuario" o "Usuario simple".

El registro se realiza introduciendo un nombre, unos apellidos, un Email de contacto, un nombre de usuario y una contraseña.

Ilustración 8: Registro de usuario.

- Recuperar Contraseña: es la posibilidad de recuperar las contraseñas de un usuario en caso de haberla olvidado. El sistema nos mandará un E-mail con la contraseña una vez hayamos introducido nuestro E-mail y nombre de usuario.

Ilustración 9: Cambio de contraseña

- Buscar Fórmulas por código: es la posibilidad de introducir código LaTeX en un campo de texto para realizar una búsqueda en la BBDD con respecto al mismo para que se nos muestren la coincidencias por pantalla

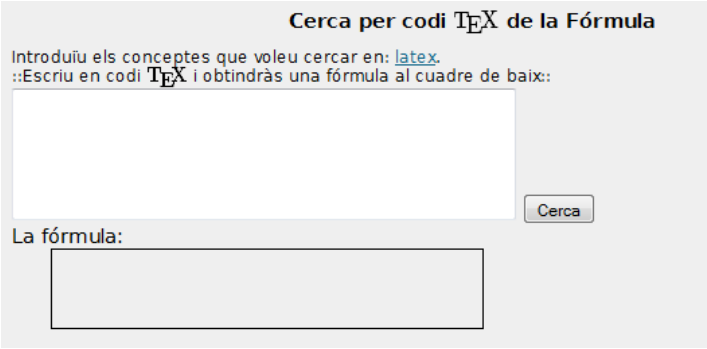


Ilustración 10: Buscador de formulas por código LaTeX.

- Búsqueda por Índice: es la posibilidad de elegir de un desplegable el índice que deseamos para que nos muestre todas las fórmulas que dependen de dicho Índice incluyendo las fórmulas pertenecientes a índices dependientes del índice seleccionado.

Ilustración 11: Buscador por índice1.

- Búsqueda por Nombre de Índice: es la posibilidad de introducir en un campo de texto el nombre del índice que deseamos. El sistema nos devolverá un listado de fórmulas ordenado por el índice al que pertenecen.

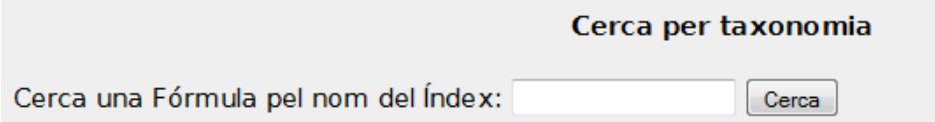


Ilustración 12: Buscador por índice2.

- Visualización de Fórmulas: es la posibilidad de visualizar las fórmulas buscadas en formato matemático regular

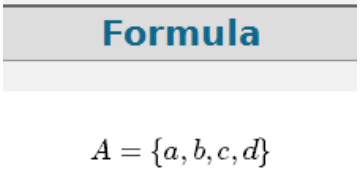


Ilustración 13: Visualización de fórmulas.

- Visualización código LaTeX de las Fórmula: es la posibilidad de ver el código LaTeX del que está hecha la fórmula matemática que vemos en pantalla. Esto se consigue pulsando sobre la fórmula deseada con el botón derecho del ratón, seleccionando “Format” y luego “TeX”. Seguidamente pulsamos otra vez sobre la fórmula con el botón derecho de ratón y seleccionamos “Show Source”. Esto abrirá una ventana nueva con el código deseado.

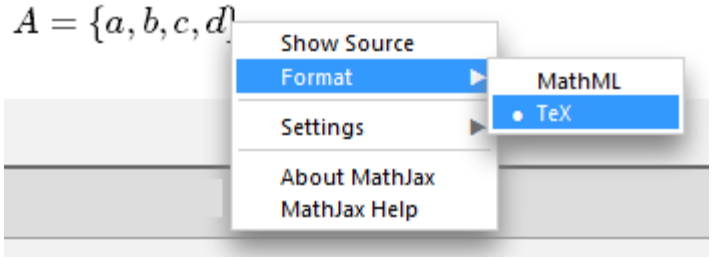


Ilustración 14: Visualización código LaTeX.

- Visualización código MathML de las Fórmula: es la posibilidad de ver el código LaTeX del que está hecha la fórmula matemática que vemos en pantalla. Esto se consigue pulsando sobre la fórmula deseada con el botón derecho del ratón, seleccionando “Format” y luego “MathML”. Seguidamente pulsamos otra vez sobre la fórmula con el botón derecho de ratón y seleccionamos “Show Source”. Esto abrirá una ventana nueva con el código deseado.

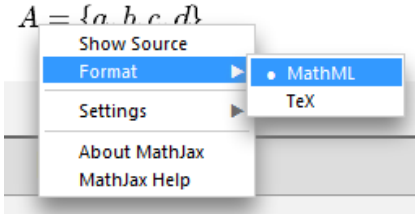


Ilustración 15: Visualización de código MathML

- Visualización de locuciones: es la posibilidad de ver las locuciones disponibles de una fórmula. Estas locuciones contendrán información del idioma en el que se grabaron, una transcripción de la locución y un reproductor de la locución propiamente dicha.

Ilustración 16: Listado de louciones.

- Escuchar locuciones: es la posibilidad de reproducir una locución.

Ilustración 17: Reproducción de locuniones.

#### 3.2.1.4.2 Usuarios de tipo “Usuario básico o simple”.

Se trata de un usuario registrado y es así como el sistema considerará a toda persona que haya creado una cuenta nueva y acceda a la aplicación con la misma. Estos “usuarios” tendrán los siguientes derechos o privilegios:

- Añadir Locuciones: es la posibilidad de introducir una locución nueva indicando el idioma, el fichero de la locución y una transcripción del contenido del archivo MP3.

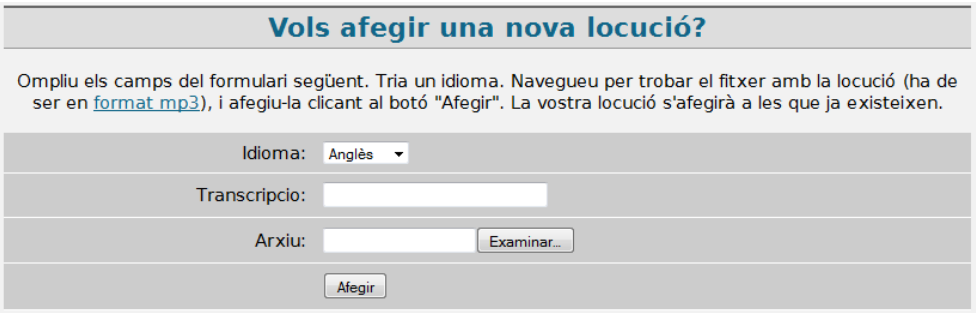


Ilustración 18: Añadir locuciones.

- Modificar sus Locuciones: es la posibilidad de modificar las locuciones que pertenecen a dicho usuario.

- Borrar sus Locuciones: es la posibilidad de borrar una locución que pertenezca a dicho usuario.

- Añadir Fórmulas: es la posibilidad de introducir una fórmula nueva indicando el índice al que pertenece y el código LaTeX del mismo.

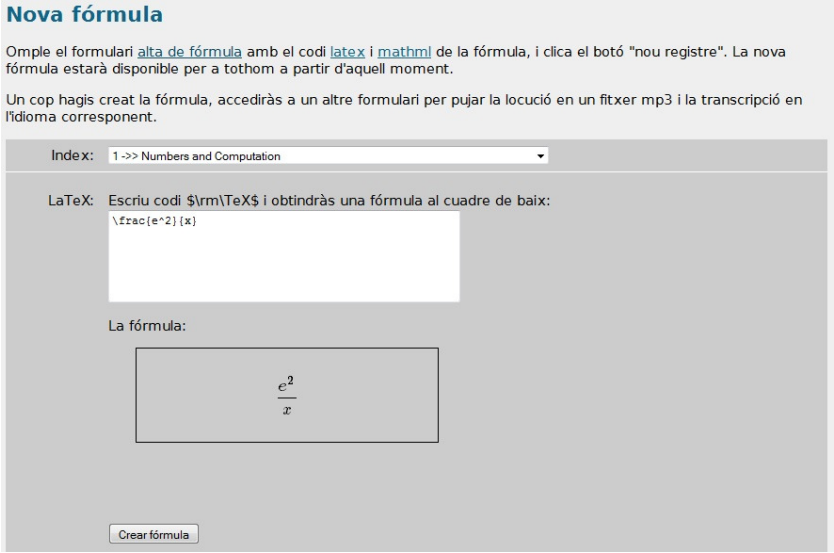


Ilustración 19: Añadir fórmulas.

- Modificar sus Fórmulas: es la posibilidad de modificar las fórmulas que pertenecen a dicho usuario.

- Borrar sus Fórmulas: es la posibilidad de borrar las fórmulas que pertenecen a dicho usuario.

#### 3.2.1.4.3 Usuarios de tipo "Editor".

Se trata de un usuario registrado y es así es como el sistema considerará a todo usuario al que un Administrador haya otorgado esos derechos o privilegios. Estos "usuarios" tendrán los siguientes derechos o privilegios:

- Modificar Locuciones: es la posibilidad de modificar las locuciones que pertenecen a cualquier usuario.

- Borrar Locuciones: es la posibilidad de borrar una locución que pertenezca a cualquier usuario.

- Modificar Fórmulas: es la posibilidad de modificar las fórmulas que pertenecen a cualquier usuario.

- Borrar Fórmulas: es la posibilidad de borrar las fórmulas que pertenecen a cualquier usuario.

- Añadir Índices: es la posibilidad de añadir índices nuevos indicando el índice padre de la misma, la numeración de este nuevo índice y título o nombre de dicho nuevo índice.

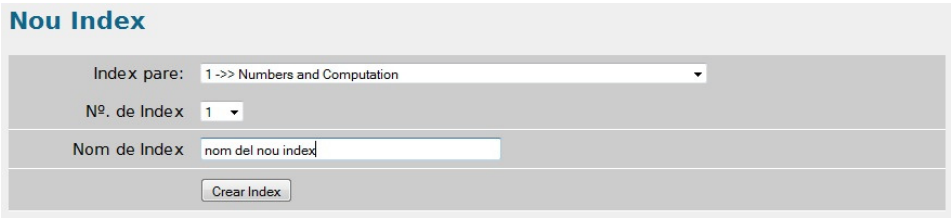


Ilustración 20: Añadir índices.

- Borrar Índices: es la posibilidad de eliminar un índice. Eso producirá la eliminación en cascada de las fórmulas y locuciones que dependan de dicho índice.

#### 3.2.1.4.4 Usuarios de tipo “Administrador”.

Se trata del usuario con el nivel más alto de privilegios (super usuario del sistema) y éste tendrá los siguientes derechos o privilegios:

- Añadir Usuarios: es la posibilidad de crear usuarios nuevos. Este proceso es análogo al registro de usuarios nuevos. Por lo tanto el usuario creado tendrá los privilegios de un usuario regular.

- Modificar datos de Usuario: es la posibilidad de modificar los datos de todos los usuarios.

- Modificar contraseña de Usuario: es la posibilidad de modificar las contraseñas de todos los usuarios del sistema.

- Activar / Suspender Usuario: es la posibilidad de suspender o activar un usuario. Como es normal un usuario suspendido no puede iniciar sesión en el sistema.

- Aprobar Usuarios nuevos: es la posibilidad de aprobar o activar un usuario recientemente creado.
  Por defecto un usuario recientemente creado por un invitado estará pendiente de activación por parte del administrador.

- Modificar Nivel de Privilegios de Usuario: es la posibilidad establecer los privilegios que tendrán los usuarios.

**Menu Administració: Admin**

Tornar a L'inici

**Usuaris Actius**

| Name: | Email Address: | Username: | Info: | Login: | Level: | Action | | |
|---|---|---|---|---|---|---|---|---|
| Administrador | | Admin | Administrador | 27/05/11 19:58:13 | 5 | | | |
| Editor | David.guaitat@gm | Editor | | 26/05/11 18:02:29 | 4 | | | |
| Usuari1 | | Usuari | | 27/05/11 19:55:04 | 1 | | | |

**Usuaris Suspés**

| Name: | Email Address: | Username: | Info: | Login: | Level: | Action |
|---|---|---|---|---|---|---|
| No Members to Display | | | | | | |

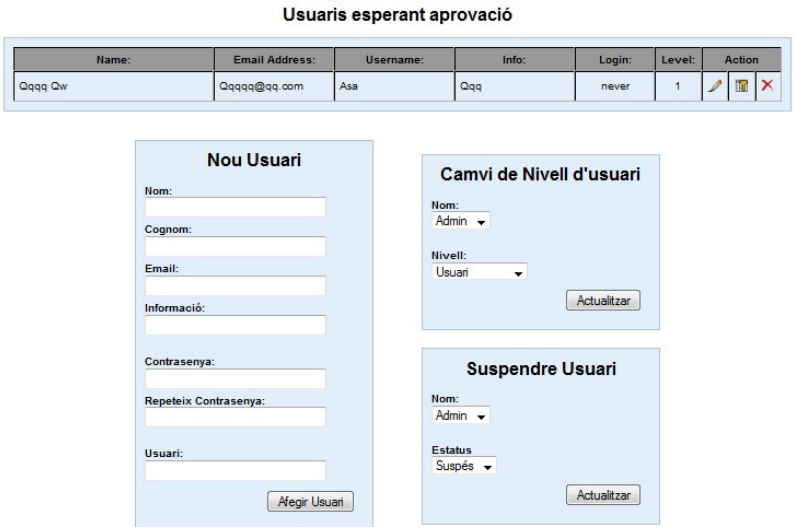

Ilustración 21: Panel del administrador.

### 3.2.2 Workflow.

Debido a que tenemos distintos niveles de privilegios, hemos optado por crear un esquema de Workflow para cada nivel.

Los esquemas de Workflow se pueden consultar en los anexos del apartado 6.4.

### 3.2.3 Taxonomía.

Debido a la gran cantidad de fórmulas que pueden almacenarse en la base de datos, nos encontramos con la necesidad de integrar un sistema de organización de las mismas. Esta mejora resulta de gran importancia para que los usuarios no se pierdan en listas interminables de fórmulas sin que sepan a qué campo de las matemáticas pertenecen.

#### 3.2.3.1 Cambios a nivel de Base de Datos.

Así pues, se han realizado las siguientes mejoras:

- Creado un campo en la tabla “formulas” para contener el ID del índice taxonómico de forma que podamos acceder a las fórmulas mediante este tipo de índice.

| Name | Type | Length | Decimals | Allow Null |
|---|---|---|---|---|
| IdFormula | bigint | 20 | 0 | ☐ |
| Index <--- Campo añadido | varchar | 14 | 0 | ☑ |
| LaTeX | text | 0 | 0 | ☑ |
| MathMLC | text | 0 | 0 | ☑ |
| MathMLP | text | 0 | 0 | ☑ |
| OpenMath | text | 0 | 0 | ☑ |

Ilustración 22: Tabla de fórmulas modificada.

- Creado un sistema de introducción de índices (dado que la tabla “taxonomía” ya existía de antemano) teniendo la posibilidad de elegir de qué otro índice

dependerá el que estamos introduciendo. De esta manera tenemos un sistema de índice-subíndices para poder organizar las fórmulas.

#### 3.2.3.2 Cambios a nivel de Código y Formularios.

Dado que en los formularios iniciales no había ningún elemento para etiquetar las fórmulas se han tenido que añadir los siguientes elementos:

- Creado un formulario para la introducción de nuevos índices para que luego los usuarios puedan hacer referencia a estos al crear, modificar o buscar fórmulas. Estos índices nuevos serán creados indicando de qué otro índice dependen o en su defecto serán una raíz nueva de índices.

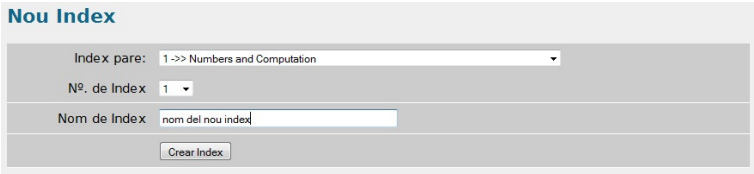


Ilustración 23: Inserción de índices.

- Creado un formulario para la eliminación de índices innecesarios.

Ilustración 24: Eliminaión de índices.

- Creado un campo desplegable con los índices para que los usuarios puedan filtrar las fórmulas por ese criterio.
  Esta búsqueda / filtro, dará como resultado todas las fórmulas que pertenezcan a dicho índice incluyendo las de los índices dependientes de forma ordenada.

- Creado un campo de texto para realizar búsquedas por nombre de índice. Esta búsqueda dará como resultado una lista de fórmulas cuyos nombres de índice coincidan con el texto introducido. El Listado estará ordenado por número de índice.

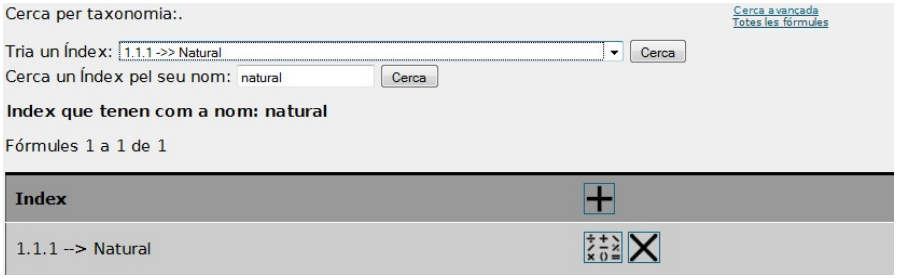


Ilustración 25: Búquedas por índices.

- Creado un campo desplegable tanto en el formulario de inserción como el de modificación de fórmulas de forma que puedan elegir el índice del que

dependen. Consideraremos que una fórmula siempre ha de pertenecer a un índice y así evitaremos crear contenido desordenado.

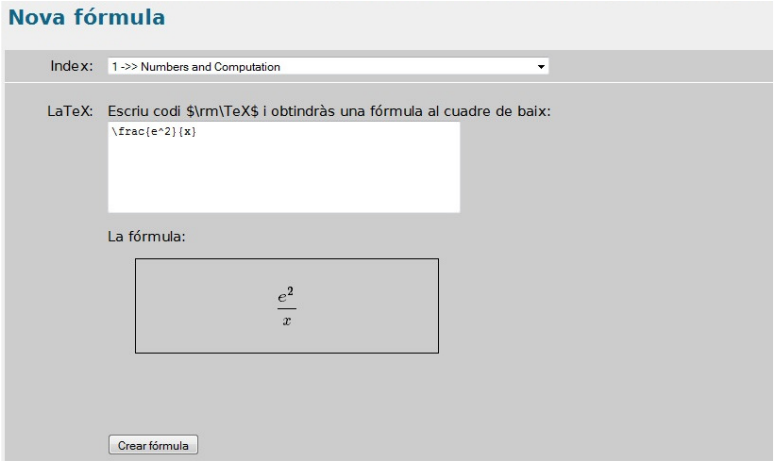

Ilustración 26: Inserción de fórmulas con índices.

### 3.2.4 Idiomas.

Este repositorio de fórmulas y locuciones de las mismas estaba pensado desde un principio para poder almacenar locuciones en varios idiomas pero, para la introducción de locuciones era necesario especificar en un campo de texto, el código del idioma en dos dígitos (es, en, ca,…). Dado que este sistema da pié a numerosos errores, hemos optado por realizar las siguientes mejoras:

#### 3.2.4.1 Cambios a nivel de Base de Datos.

- Creada una tabla “idiomas” que contenga un “ID” de idioma correspondiente a la codificación de idioma tipo “es = español” y “en= inglés”.

| Name | Type | Length | Decimals | Allow Null |
|---|---|---|---|---|
| Ididioma | char | 2 | 0 | ☐ |
| NomIdioma | text | 0 | 0 | ☐ |

Ilustración 27: Tabla de idiomas.

- Esta tabla la usaremos también para obtener los datos para la elección de idioma durante el proceso de ver editar o crear locuciones.

#### 3.2.4.2 Cambios a nivel de Código y Formularios.

- Añadido un desplegable de selección de idioma en los formularios de introducción y modificación de locuciones para que de forma obligatoria, una taxonomía, tenga que tener un idioma asociado.

Ilustración 28: Inserción de locuciones indicando el idioma.

- Creado un Script para asegurarse que de realizarse alguna modificación sobre la locución, ya sea de idioma o archivo, este guarde el nuevo archivo en la ruta que le corresponda según el idioma del mismo. De la misma manera una locución eliminada provocará la eliminación del archivo asignado a la misma para evitar la proliferación de archivos mp3 basura.

- Para evitar que se produzcan sobre-escrituras de archivos, todo archivo nuevo introducido obtendrá un nombre generado por el sistema consistiendo en: "Id del la fórmula_día-mes-año-hora-minuto-segundo.mp3" dando un resultado parecido a este "97_20-05-2011-11-09-22.mp3".

## 3.3 Mejoras de aspectos visuales y multimedia.

El otro objetivo principal que pretendemos alcanzar durante la actualización de "Rodolfo" es mejorar la relación usuario-aplicación.

Está claro que tener un repositorio de fórmulas matemáticas que sólo puedan verse en formato codificado ya sea LaTeX o MathML no termina de resultar práctico, dado que el objetivo de la aplicación es facilitar el aprendizaje de fórmulas matemáticas en cuanto a la forma en que estas han de ser leídas en varios idiomas.

Ahora que sabemos que no pretendemos aprender LaTeX o MathML, tenemos que encontrar una solución que nos permita tanto la visualización como edición de fórmulas en un formato puramente matemático.

Hemos de tener en cuenta que para la edición de fórmulas será necesario cierto conocimiento de codificación LaTeX pero pretendemos que el usuario pueda contar con la asistencia de un WYSIWYG que facilite la tarea.

Para la visualización de fórmulas pretendemos que una aplicación Open Source tome la codificación guardada en formato LaTeX y la transforme en formato matemático.

En lo que se refiere a las locuciones hemos optado por añadir un sencillo Script que facilite la reproducción de los mismos sin tener que descargar el archivo de audio.

### 3.3.1 Motor "MathJAX".

Según la propia página de los desarrolladores MathJax es:

*"Un motor de código abierto de visualización de JavaScript para LaTeX y MathML que funciona en todos los navegadores modernos. Fue diseñado con el objetivo de consolidar los recientes avances en las tecnologías web en una única plataforma*

*definitiva, las matemáticas en la web, el apoyo a los principales navegadores y sistemas operativos. No requiere configuración por parte del usuario (sin plugins para descargar o software para instalar), por lo que el autor de la página puede escribir documentos web, que incluyen las matemáticas y tener la confianza que los usuarios podrán verla de forma natural y sencilla. Uno simplemente incluye MathJax y algo de matemáticas en una página web, y MathJax hace el resto.*

*MathJax usa fuentes basadas en web (en los navegadores que lo soportan) para producir en alta calidad una composición tipográfica que escala e imprime a resolución completa (Los caracteres matemáticos se incluyen como imágenes). MathJax se puede utilizar con los lectores de pantalla y para la accesibilidad de los los discapacitados visuales. Con MathJax, las fórmulas matemáticas están basadas en texto en lugar de basado en imageesn, por lo que está disponible para los motores de búsqueda, lo que significa que de sus ecuaciones se pueden realizar búsquedas, al igual que el texto de sus páginas. MathJax permite a los autores de páginas escribir las fórmulas que usan la notación TeX y LaTeX, o MathML, un consorcio mundial para la representación de las matemáticas en formato XML. MathJax permite incluso convertir la notación TeX en MathML, por lo que puede hacerse más rápidamente por aquellos navegadores que admitne MathML de forma nativa, o para que pueda copiar y pegar en otros programas.*

*MathJax es modular, por lo que los componentes se cargan sólo cuando sea necesario, y se puede ampliar para incluir nuevas capacidades, según convenga. MathJax es altamente configurable, permitiendo a los autores personalizarlo para las necesidades especiales de sus sitios web. Por último, MathJax tiene una interfaz de programación de aplicaciones ricas (API) que puede ser utilizada para hacer que las matemáticas sean interactivas y dinámicas en sus páginas web".*

Para nuestro proyecto hemos considerado que por el hecho de que MathJax es capaz de traducir de la codificación LaTeX a MathML, no resulta necesario guardar en nuestra B.D. ambas codificaciones y por tanto hemos optado por introducir las fórmulas en formato LaTeX.

Después, durante la visualización de las fórmulas, podremos optar por visualizar tanto el código LaTeX como MathML si así lo requerimos.

#### 3.3.1.1 Visualizador de fórmulas

Para aplicar la visualización de fórmulas hemos tenido que añadir las siguientes líneas de código entre las etiquetas <head> </head> de las páginas que muestran fórmulas en codificación LaTeX:

```
<!-- MathJax  -->

<script type="text/x-mathjax-config">
  MathJax.Hub.Config({
    extensions: ["tex2jax.js"],
    jax: ["input/TeX","output/HTML-CSS"],
    tex2jax: {inlineMath: [["$","$"],["\\(","\\)"]]}
  });
</script>
<script type="text/javascript" src="mathjax/MathJax.js"></script>
```

```
<style>
h1 {text-align:center}
h2 {
  font-weight: bold;
  background-color: #DDDDDD;
  padding: .2em .5em;
  margin-top: 1.5em;
  border-top: 3px solid #666666;
  border-bottom: 2px solid #999999;
}
</style>
```

Estas líneas de código JavaScript permiten tomar como código a interpretar como fórmula matemática a todo lo que se encuentre entre "\[" y "\]". Luego hace una llamada al archivo principal que maneja las funcionalidades de "MathJax.js". Finalmente define una serie de parámetros de estilos que aplicará a la visualización de las fórmulas.

Como hemos comentado más arriba todo código LaTeX que pretendamos formatear hemos de ponerlo entre "\[" y "\]", por lo que todas las llamadas a este campo de la B.D. tendrán un formato parecido al siguiente:

```
\[<?php echo row_Recordset1['LaTeX']; ?>\]
```

Así pues la siguiente línea de código:

```
A=\left\{a, b, c, d\right\}
```

Se transformaría en: $A = \{a, b, c, d\}$

### 3.3.1.2 Editor de fórmulas

La disposición de un editor de fórmulas resulta más complejo debido a que la conversión de código LaTeX a formato matemático ha de realizarse de forma dinámica y no sólo al cargar el contenido de la página.

Así pues la opción que nos permite MathJax comienza por la introducción de las siguientes líneas de código entre las etiquetas <head> y </head> de las páginas que contengan un editor de fórmulas:

```
<script type="text/x-mathjax-config">
  MathJax.Hub.Config({
    extensions: ["tex2jax.js"],
    jax: ["input/TeX","output/HTML-CSS"]
  });
</script>
<script type="text/javascript" src="mathjax/MathJax.js"></script>

<style>
input {margin-top: .7em}
```

```
.output {
  border: 1px solid black;
  padding: 1em;
  width: auto;
  position: absolute; top: 0; left: 2em;
  min-width: 20em;
}
.box {position: relative}
</style>
```

Las siguientes líneas de código JavaScript se añaden dentro del cuerpo de la página, es decir entre las etiquetas <body> y </body>. Estas líneas son imprescindibles para que el código que vayamos introduciendo por un “textarea” se vaya interpretando o transformando en un formato matemático.

```
<script>
  //
  //  Use a closure to hide the local variables from the
  //  global namespace
  //
  (function () {
    var QUEUE = MathJax.Hub.queue;  // shorthand for the queue
    var math = null, box = null;    // the element jax for the math output, and the box it's in

    //
    //  Hide and show the box (so it doesn't flicker as much)
    //
    var HIDEBOX = function () {box.style.visibility = "hidden"}
    var SHOWBOX = function () {box.style.visibility = "visible"}

    //
    //  Get the element jax when MathJax has produced it.
    //
    QUEUE.Push(function () {
      math = MathJax.Hub.getAllJax("MathOutput")[0];
      box = document.getElementById("box");
      SHOWBOX(); // box is initially hidden so the braces don't show
    });

    //
    //  The onchange event handler that typesets the
    //  math entered by the user
    //
    window.UpdateMath = function (TeX) {
      QUEUE.Push(HIDEBOX,["Text",math,"\\displaystyle{"+TeX+"}"],SHOWBOX);
    }

    //
    //  IE doesn't fire onchange events for RETURN, so
    //   use onkeypress to do a blur (and refocus) to
    //   force the onchange to occur
    //
```

```
    if (MathJax.Hub.Browser.isMSIE) {
      window.MathInput.onkeypress = function () {
        if (window.event && window.event.keyCode === 13) {this.blur(); this.focus()}
      }
    }
  })();
</script>
```

Las siguientes líneas de código las pondremos donde nos interese en el cuerpo de la página.

```
<textarea name="LaTeX" id="MathInput" cols="50"  rows="5"
onkeyup="UpdateMath(this.value)" /></textarea>
```

Este “textarea” irá almacenando el código que vayamos introduciéndole y al mismo tiempo irá ejecutando la función JavaScript “UpdateMath(this.value)”, la cual, irá tomando el valor guardado y lo interpretará y transformará para su representación matemática.

```
<div class="box" id="box" style="visibility:hidden">
        <div id="MathOutput" class="output">$${}$$</div>
</div>
```

Estas líneas tienen la finalidad de identificar el lugar donde el motor MathJax ha de dar salida a la fórmula interpretada a partir del código introducido.

Como podemos observar en la siguiente figura el resultado consiste en un editor de fórmulas que facilita la tarea de los usuarios teniendo de forma inmediata el resultado del código introducido.

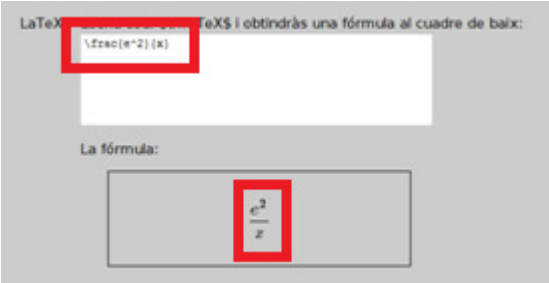


Ilustración 29: Editor de fórmulas amigable.

### 3.3.1.3 Integración MathJax Alternativa.

Según la documentación de MathJax, existe una forma alternativa de cargar el motor de visualización de fórmulas matemáticas que permitiría el no necesitar guardar los archivos del mismo en nuestro sitio web y facilitando las actualizaciones automáticas del mismo.

*“MathJax está ahora disponible como un servicio web de cdn.mathjax.org, Por lo que tenemos la posibilidad de obtener MathJax de allí sin necesidad de instalarlo en su propio servidor. El CDN es parte de una distribución "nube" de la red, por lo que está*

*cargo de servidores en todo el mundo. Eso significa que usted debe tener acceso a un servidor geográficamente cercano para una conexión rápida y fiable.*

*El CDN acoge la versión más actualizada de MathJax, así como la mayoría de versiones angteriores, así que usted puede vincularse a una versión de MathJax actualiza de MathJax, o puede quedarse con una de las versiones de lanzamiento para que sus páginas siempre utilicen la misma versión de MathJax.*

*La dirección URL que se utiliza para obtener MathJax determina la versión que estará utilizando. El CDN tiene la siguiente estructura de directorios:*

```
mathjax /        # Nombre-proyecto
  1.0 de última /
  1.1-beta /     # Temporal
  1.1 de última /   # La versión 1.1 con los parches ciritical
  ...
  más /       # La versión más reciente (1.1 más reciente, en este caso)
```

### 3.3.2 Motor de reproducción de audio.

Un reproductor de audio integrado en el proyecto Rodolfo puede parecer un detalle de poca importancia pero resulta muy útil, sobre todo por su comodidad para los usuarios del sistema.

#### 3.3.2.1 Reproductor de locuciones.

El sistema que hemos elegido para la reproducción de audio se trata de un Script desarrollado en Javascript llamado BSSound. Es un script sencillo que toma cualquier etiqueta de hipervínculo “<a> </a>” que apunte a un archivo de tipo “.mp3” y lo transforme en dos iconos preestablecidos “play” y “stop”. Pulsando sobre “play” el script llamará a dicho archivo evitando que se reproduzca de forma regular o descargue y en su lugar reproducirá dicho audio en la propia página en la que nos encontramos.

El código necesario para que se lleve a cabo ha de insertarse entre las etiquetas <head> y </head> de la página que contendrá los enlaces a archivos “.mp3”.

```
<!-- AUDIO -->
<link rel="stylesheet" href="estils/player.css" type="text/css" media="screen" />
<script type="text/javascript" src="js/jquery-1.3.2.min.js"></script>
<script type="text/javascript" src="js/jquery.pngFix.pack.js"></script>
<script type="text/javascript" src="js/jquery.bssound.js"></script>
<script type="text/javascript">
</script>
```

Estas líneas llaman a los archivos que conforman el motor de reproducción de audio y como resultado omtenemos dos botones: uno para iniciar la reproducción y otro para la pararla, tal y como podéis observar en la siguiente ilustración.

Ilustración 30: Reproducción de audio.

#### 3.3.2.2 Grabador de locuciones.

Durante el desarrollo de este TFC hemos estado investigando la posibilidad de tener un grabador de audio integrado en el proyecto Rodolfo ya que facilitaría en gran medida el trabajo de los usuarios a la hora de crear locuciones.

Este sistema requería que se generara un archivo mp3 temporal en el árbol de directorios de nuestra aplicación web, cosa que no hemos encontrado posible sin la utilización de algún servidor de streaming. Así mismo necesitábamos desarrollar la grabadora en Flash y por lo tanto la solución dejaba de ser open source.

Debido a que el desarrollo de un sistema de este tipo resulta complejo y extenso, lo hemos descartado y dejado como posible mejora como un proyecto en sí mismo.

## 3.4 Conclusiones

Este Proyecto ha supuesto el poner en práctica diversas aptitudes obtenidas a lo largo de los semestres de estudio con la UOC, lo cual, ha permitido llevarlo a cabo de forma satisfactoria.

A lo largo de las fases de trabajo planteadas desde el comienzo del proyecto hemos podido encontrar soluciones a las problemáticas encontradas y por tanto hemos conseguido alcanzar todos los objetivos que nos planteaba este TFC.

Los objetivos se han alcanzado buscando siempre el mejor resultado aplicando una solución lo más sencilla posible.

Podemos así decir que este proyecto está terminado pero evidentemente tenemos en cuenta que este proyecto no ha alcanzado su mayor potencial, sino que requerirá de nuevas actualizaciones para seguir ampliando su funcionalidad y utilidad para los estudiantes de matemáticas. Dentro de estas actualizaciones o mejoras pendientes podemos señalar el grabador de audio integrado en la aplicación Web.

# 4 Glosario

**Base de Datos**: Es un conjunto estructurado de datos que representa, entre otros, entidades y su interrelaciones, con integración y compartimentación de datos.

**BD**: Ver Base de Datos.

**BSSound**: es un script creado para poder reproducir audio mp3 a través de las páginas web.

**Código Libre**: es el término con el que se conoce al software distribuido y desarrollado libremente. El código abierto tiene un punto de vista más orientado a los beneficios prácticos de compartir el código que a las cuestiones morales y/o filosóficas las cuales destacan en el llamado software libre.

**Diagrama de Gantt**: es una popular herramienta gráfica cuyo objetivo es mostrar el tiempo de dedicación previsto para diferentes tareas o actividades a lo largo de un tiempo total determinado.

**Flujo de trabajo**: Es el estudio de los aspectos operacionales de una actividad de trabajo: cómo se estructuran las tareas, cómo se realizan, cuál es su orden correlativo, cómo se sincronizan, cómo fluye la información que soporta las tareas y cómo se le hace seguimiento al cumplimiento de las tareas.

**Hosting**: El alojamiento web (en inglés web hosting) es el servicio que provee a los usuarios de Internet un sistema para poder almacenar información, imágenes, vídeo, o cualquier contenido accesible vía web.

**JavaScript**: es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos,[3] basado en prototipos, imperativo, débilmente tipado y dinámico.

**LaTeX**: es un sistema de composición de textos, orientado especialmente a la creación de libros, documentos científicos y técnicos que contengan fórmulas matemáticas.

**Locución**: el texto hablado que se emite en los medios audiovisuales.

**MathJAX**: es una biblioteca javascript que permite visualizar fórmulas matemáticas en navegadores web, utilizando los lenguajes de marcado LaTeX o MathML.[

**MathML**: es un lenguaje de marcado basado en XML, cuyo objetivo es expresar notación matemática de forma que distintas máquinas puedan entenderla, para su uso en combinación con XHTML en páginas web, y para intercambio de información entre programas de tipo matemático en general.

**MP3**: es un formato de compresión de audio digital patentado que usa un algoritmo con pérdida para conseguir un menor tamaño de archivo. Es un formato de audio común usado para música tanto en ordenadores como en reproductores de audio portátil.

**Open Source**: Ver Código libre.

**Permiso**: Privilegio que se otorga a un usuario (o conjunto de usuarios) para realizar una operación determinada sobre un cierto objeto de la BD.

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

**Procedimiento Almacenado**: Acción o función definida por un usuario que proporciona un determinado servicio. Una vez ha sido creado, se guarda en la BD y pasa a ser tratado como un objeto más de ésta. La ejecución de un procedimiento puede devolver ninguno, uno o más valores.

**Rol**: Agrupación de privilegios sobre alguno de los componentes de una BD.
Script: Un script (cuya traducción literal es 'guión') o archivo de órdenes o archivo de procesamiento por lotes es un programa usualmente simple, que por lo regular se almacena en un archivo de texto plano.

**SGBD**: Ver Sistema de Gestión de Bases de Datos.

**Sistema de Gestión de Bases de Datos**: Software que gestiona y controla bases de datos. Sus principales funciones son las de facilitar la utilización simultánea a muchos usuarios de tipos diferentes, independizar al usuario del mundo físico y mantener la integridad de los datos.

**SQL**: Lenguaje pensado para describir, crear, actualizar y consultar bases de datos. Fue concebido por IBM a finales de los años setenta y estandarizado por ANSI e ISO en el año 1986 (el último estándar del SQL es de 1999). Actualmente lo utilizan casi todos los SGBD del mercado (incluso algunos SGBD no relacionales y algunos sistemas de ficheros).

**Streaming**: consiste en la distribución de audio o video por Internet. La palabra streaming se refiere a que se trata de una corriente continua (sin interrupción). El usuario puede escuchar o ver en el momento que quiera. Este tipo de tecnología permite que se almacenen en un búfer lo que se va escuchando o viendo. El streaming hace posible escuchar música o ver videos sin necesidad de ser descargados previamente.

**Taxonomía**: es, en su sentido más general, la ciencia de la clasificación.

**TFC**: Trabajo Final de Carrera.

**Workflow**: Ver flujo de trabajo.

## 5 Bibliografía.


- XHTML™ 1.1-Module-based XHTML-Second Edition. http://www.w3.org/TR/xhtml11/
- Luke Welling, Laura Thomson (2004) PHP and MySQL Web Development, (3rd Edition)
- Documentación de PHP: http://www.php.net/docs.php
- MySQL :: MySQL Documentation. http://dev.mysql.com/doc/
- Flanagan, David (2002). JavaScript: The Definitive Guide (4 ª Edición)
- Manual JavaScript: http://paginaweb2.com/cat-JavaScript-7
- Documentación de MathJax: http://www.mathjax.org/docs/1.1/index.html
- Documentación de Mathquill: http://laughinghan.github.com/mathquill/
- Documentación de Suim:. http://jp/en/tutorial.html

# 6 Anexos.

## 6.1 Requisitos del sistema.

-Hosting Web con servidor Linux o Microsoft

-PHP Versión 5 o superior

-Servidor de Base de Datos MySql

-Navegador Web (Internet Explorer 6 o superior, Mozilla Firefox 3 o superior, Google Crome 12 o superior).

## 6.2 Instrucciones de Instalación.

### 6.2.1 Subida de Archivos por FTP y estructura de directorios.

- Subir la carpeta Rodolfo a la raíz del sitio web mediante un cliente FTP (Filezilla, CuteFTP, ...)

- El código fuente del proyecto está disponible en el siguiente enlace: http://www.novopixel.es/rodolfo/rodolfo.zip

### 6.2.2 Archivos de configuración (Instrucciones).

Para la correcta funcionalidad de la aplicación es necesario configurar dos archivos ubicados en el interior de la carpeta. /rodolfo archivos:

- ./Connections/Rodolfo.php:

```
$hostname_rodolfo = "Nombre Servidor MySql";

$database_rodolfo = "Nombre BD";

$username_rodolfo = "Usuario BD";

$password_rodolfo = "Contraseña BD";
```

- ./loginscript/include/constants.php:

```
# Información BD

// Servidor BD (localhost)
define("DBHOST"," Nombre Servidor MySql ");
// Usuario BD
define("DBUSER"," Usuario BD ");
// Contraseña BD
define("DBPASS"," Contraseña BD ");
// Nombre BD
```

```
define("DBNAME"," Nombre BD ");
// Tabla USUARIOS
define("DBTBLE","cw_users");

# Información de ruta

// Ruta relativa del script de seguridad
define("Script_Path","/rodolfo/loginscript/");
// )
define("Script_URL","");

# Infomación del Sistema

// Nombre del sistema
define("Site_Name","Rodolfo");
// Nombre en Email del sistema
define("Email_From","Administrator");
// Email del webmaster
define("Email_Address","webmaster@ss.com");
// Dirección de "Dont reply"
define("Non_Reply","dontreply@ss.com");

# Información de Sesión y Cookies

// Tiempo de vida de la sesión
define("Session_Lifetime", 60*60);
// Nombre de Cookie
define("CKIEUS",'username');
define("CKIEPS","PASSWORDMD5");

# Configuración del sistema
// Requerir aprovación de usuarios
define("Admin_Approvial", true); // true o false
```

### 6.2.3 Archivos de BBDD "*.sql".

Dentro del paquete comprimido en .zip se encuentra un archivo "rodolfo.sql", el cual contiene los datos necesarios para realizar pruebas.

Será necesario crear una nueva BD en un servidor e importar los datos de este archivo.

- El archivo .sql está disponible en el siguiente enlace: http://www.novopixel.es/rodolfo/rodolfo.zip

### 6.2.4 Primer inicio de sesión y cambio de contraseña del Administrador.

Cuando entremos por primera vez al sistema tendremos que iniciar sesión como administrador (Usuario: admin; Contraseña: admin) y cambiar esta contraseña por motivos de seguridad.

Disponéis de una demo en: http://www.novopixel.es/rodolfo

## 6.3 Esquemas del Workflow por Usuarios

### 6.3.1 Invitado

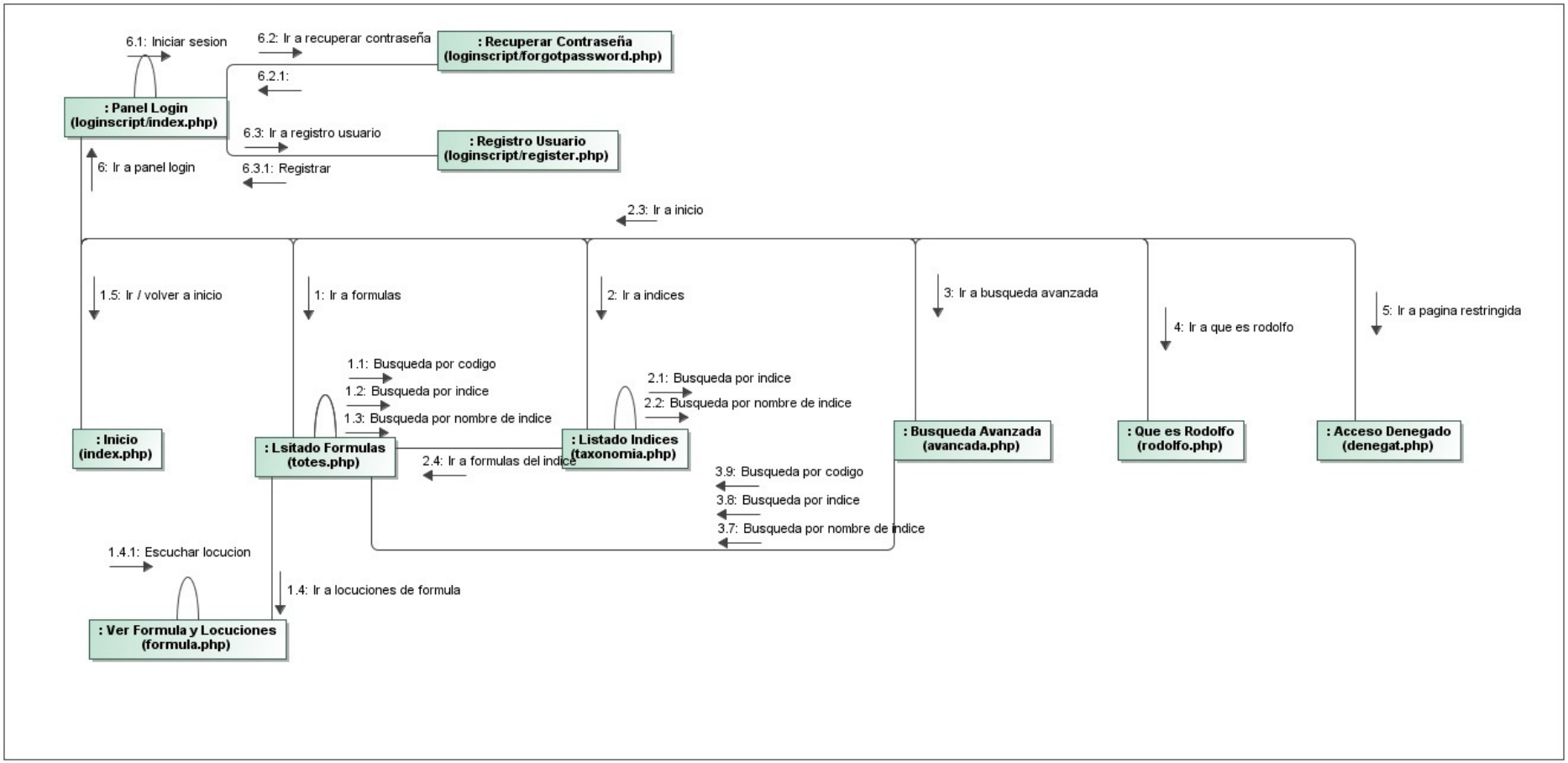


Ilustración 31: Workflow de “invitado”.

### 6.3.2 Usuario

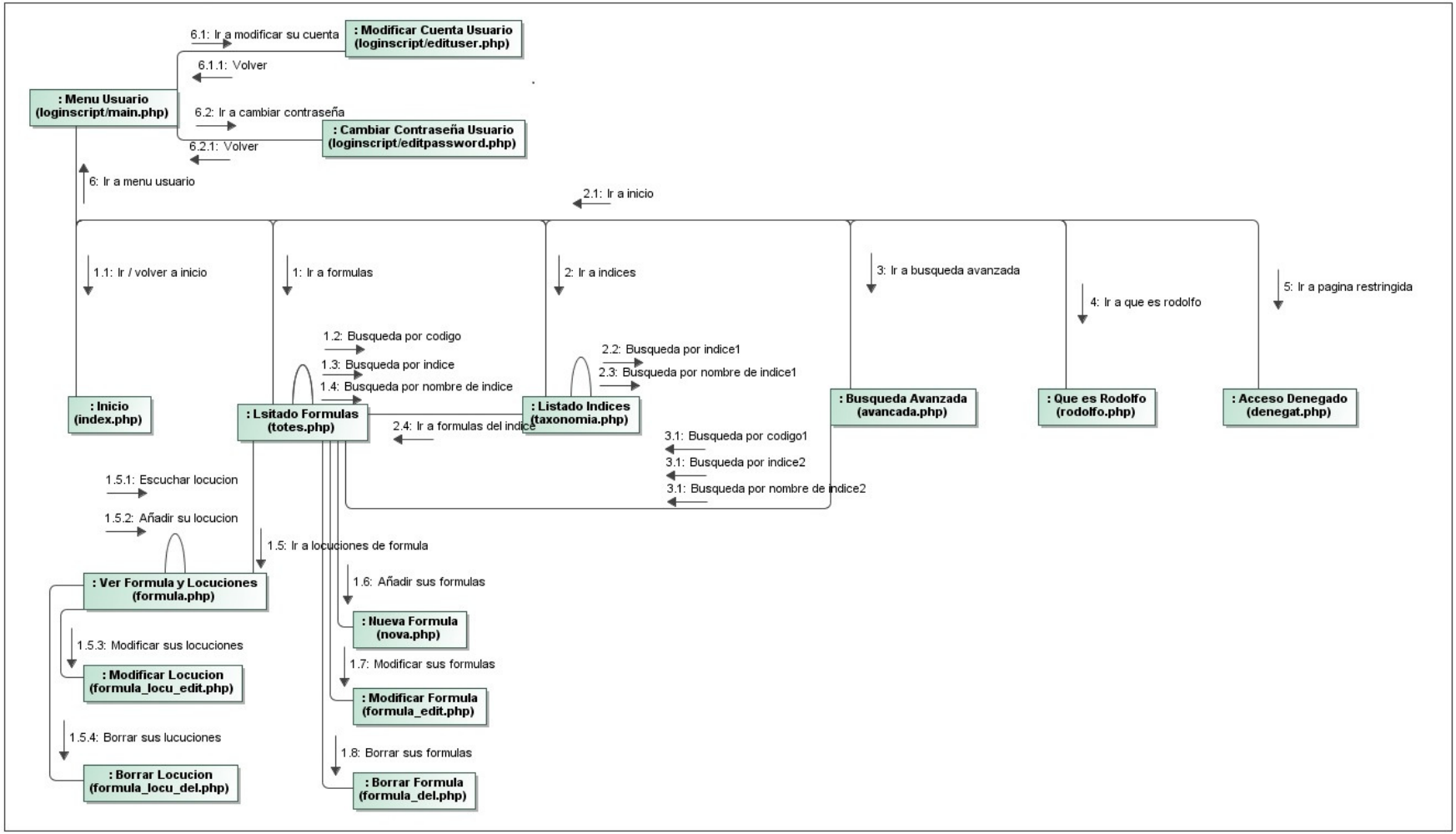


Ilustración 32: Workflow de “usuario”.

### 6.3.3 Editor

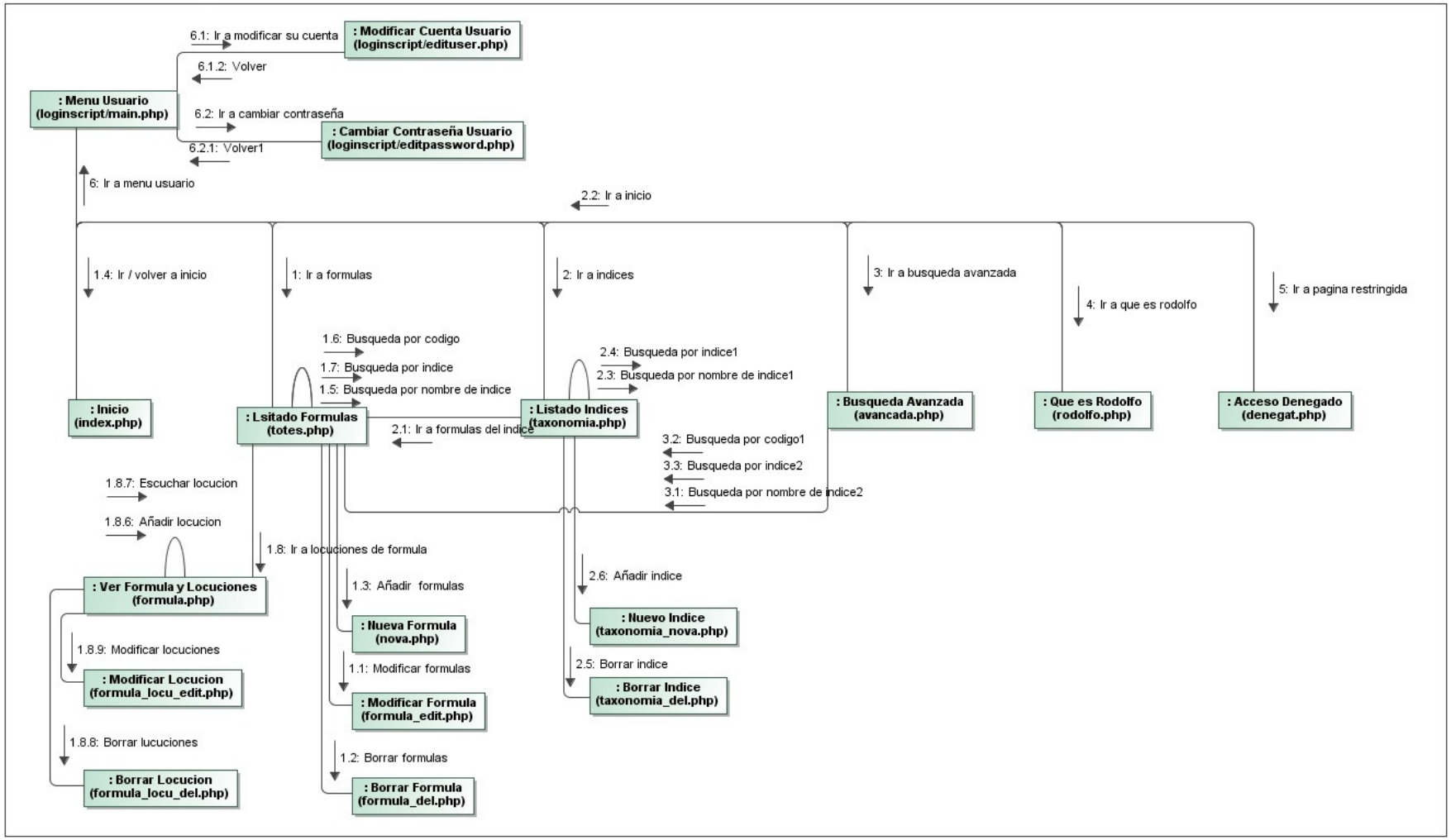


Ilustración 33: Workflow de “editor”.

### 6.3.4 Administrador

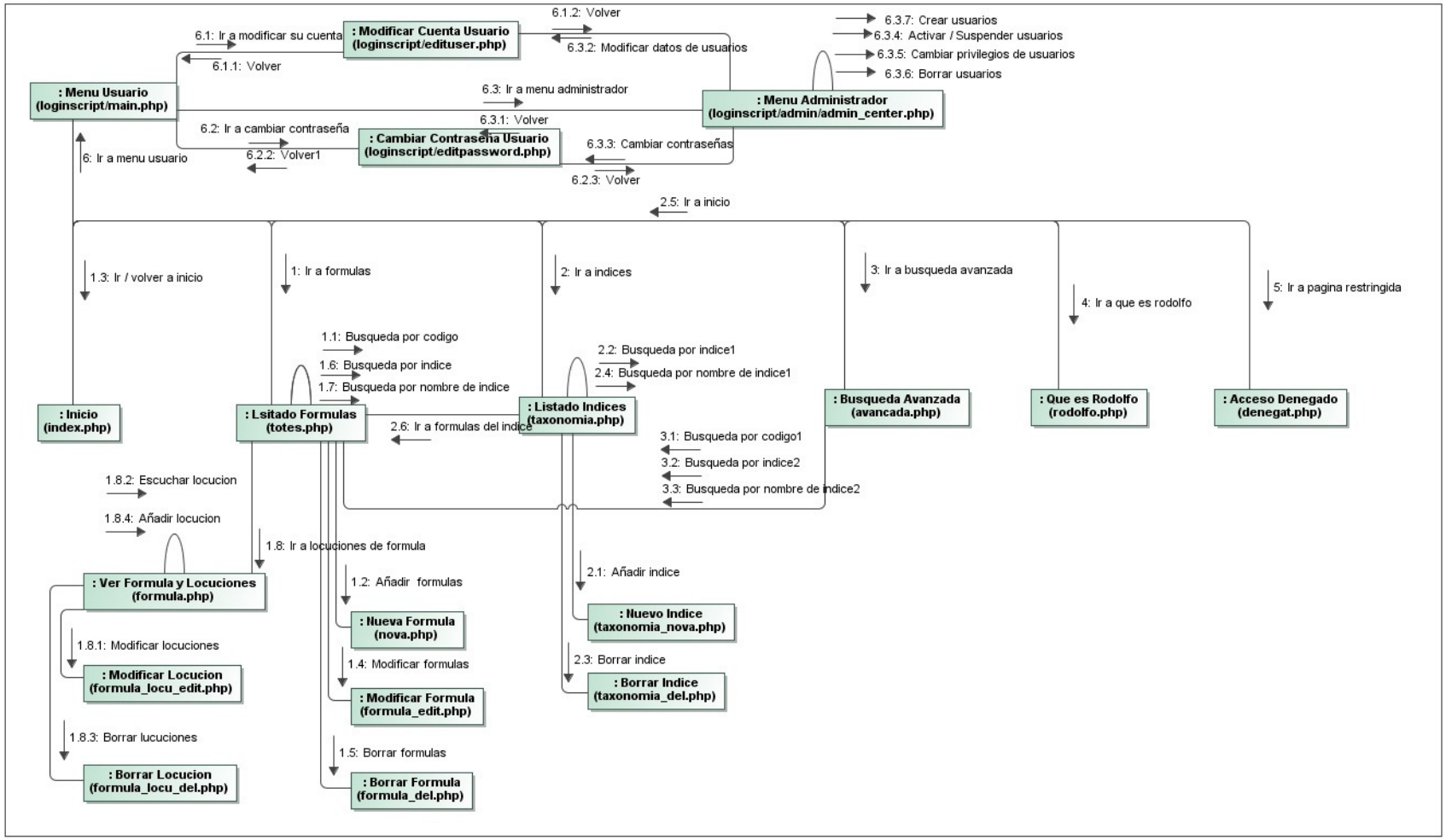


Ilustración 34: Workflow de “administrador”