



**Universidad**  
Zaragoza

# Trabajo Fin de Grado

## Student Moodle Application Asistente de escritorio para la plataforma Moodle

Autor/es

Diego Álvarez Arguedas

Director/es

Carlos Catalán Cantero

ESCUELA UNIVERSITARIA POLITÉCNICA DE TERUEL  
2018/2019

## Resumen

---

Este trabajo de fin de grado nace ante la idea de mejorar la efectividad de la plataforma Moodle, en especial para la Universidad de Zaragoza y el campus de Teruel, en lo referente a reducir el tiempo desde la publicación de un material docente y la obtención por parte del alumno de este. A pesar de que la plataforma ya cuenta con un sistema de notificaciones que realiza esta función, la carga de servidores y la cantidad tanto de alumnos del sistema.

Para el desarrollo de este trabajo se han tenido en cuenta diferentes soluciones tanto desde el lado del servidor (*plugin*) como desde la del alumno (aplicación). Así, tras un estudio preliminar, se procede al desarrollo de una aplicación. Esta aplicación utiliza la técnica de *web scraping*, que es poco afectado por cambios en el sistema. El principal objetivo del proyecto es desarrollar una aplicación que sincronice su contenido local (material docente y tareas) con la plataforma Moodle.

Como resultado del proyecto se obtiene una aplicación que cumple todos los requisitos establecidos en las primeras fases. Además, esta aplicación se ha realizado de forma modular de manera que permite un fácil mantenimiento y trabajo futuro.

Finalmente, se han hecho numerosas pruebas de uso y los resultados sitúan el tiempo necesario para la sincronización en tan solo 2-4 minutos. Además, las evaluaciones de los alumnos que la han probado son muy positivas.

**Palabras clave:** Moodle, NAS-TER, web scraping, material docente.

## Abstract

---

This end degree work is created with the idea of improving the effectiveness of the Moodle platform, specifically for the University of Zaragoza and the campus of Teruel, in relation to the reduction of time between the publication of a teaching material and the obtaining by the student. Although the platform already has a notification system that performs task, the load of servers, and both the high number of students and academic degrees in the system, causes a delay of up to 5 hours.

For the development of this project, different working have been considered from the server side (plugin), and also from the student side (application). Thus, after a preliminary study, it was proceeded to develop an application. This application employs the web scraping technique, which is little affected by changes in the system. The main goal of this application that synchronizes its local content (teaching material and tasks) with the Moodle platform.

As result of the project, the obtained application meets all the requirements that were established in the first phases. Moreover, this application has been performed in a modular way that allows an easy maintainability and work in the future.

Finally, many using tests have been performed and the experimental results establish the required synchronization time in only 2-4 minutes. In addition, the evaluations of the students who tested the application were very positive.

**Keywords:** Moodle, NAS-TER, web scraping, teaching material.

## Glosario

---

**Core:** parte de un programa o sistema que constituye una parte fundamental del mismo, el cual se ejecuta en modo privilegio.

**Drag & drop:** expresión que se refiere a la acción de mover (“agarrando”) un objeto virtual y arrastrándolo a una ubicación diferente o sobre otro objeto virtual.

**ERS:** especificación de requisitos de software, del inglés *software requirements specification* (SRS), es una descripción completa del comportamiento del sistema que va a desarrollar.

**IDE:** del inglés *integrated development enviroment*, se refiere a una aplicación software que proporciona unas instalaciones completas para el desarrollo software; normalmente consiste en un editor de código fuente, herramientas de automatización de compilación y un depurador.

**Plugin:** aplicación o programa informático que añade una funcionalidad adicional o una característica a un programa mayor.

**Quiz:** es un término utilizado para referirse a pruebas sencillas para evaluar un contenido.

**Scrum:** metodología de desarrollo ágil, caracterizada por tener una estrategia de desarrollo incremental y en el solapamiento de fases de desarrollo.

**Sprint:** periodo de tiempo que define un ciclo de trabajo en la metodología *Scrum*, normalmente entre 2 y 4 semanas.

**SystemTray:** también llamado *System tray* o *Status area*, es la es la parte de la interfaz de usuario que muestra iconos para las características del sistema y del programa que no tiene presencia en el escritorio.

**Widgets:** es una pequeña aplicación o programa, usualmente presentado en archivos o ficheros pequeños.

# ÍNDICE

---

1. Introducción .....	1
2. Análisis.....	2
2.1. Estado del arte .....	2
2.1.1. Plugins .....	3
2.1.2. Aplicaciones.....	3
2.2. Sistema planteado.....	4
2.2.1. Web <i>scraping</i> .....	5
2.2.2. Casos de uso.....	5
2.3. Especificación de Requisitos Software (ERS).....	6
2.3.1. Descripción general.....	6
2.3.2. ERS.....	7
3. Diseño.....	11
3.1. Metodología usada.....	11
3.2. Tecnologías usadas.....	12
3.2.1. Java.....	12
3.2.2. Herramientas utilizadas.....	12
3.3. Diseño estático .....	12
3.3.1. Diagrama de paquete: general.....	13
3.3.2. Diagrama de paquete: aplicacion .....	14
3.3.3. Diagrama de paquete: actualizador .....	16
3.3.4. Diagrama de paquete: sincronizacion.....	17
3.3.5. Diagrama de paquete: tools .....	18
3.4. Diseño dinámico.....	19
3.4.1. Diagrama de actividad: actualizar .....	19
3.4.2. Diagrama de actividad: proceso de actualización .....	20
3.4.3. Diagrama de actividad: cambio de frecuencia .....	21
3.4.4. Diagrama de actividad: cambio de idioma .....	22
3.4.5. Diagrama de actividad: nuevo perfil de usuario & editar perfil de usuario.....	23
3.4.6. Diagrama de actividad: sincronizar .....	25
3.4.7. Diagrama de actividad: sincronizar NAS-TER .....	27
3.4.8. Diagrama de actividad: sincronizar Moodle.....	28
3.5. Diseño de la interfaz.....	31
4. Pruebas y resultados .....	38
4.1. Sincronización NAS-TER.....	38

4.2. Sincronización Moodle .....	39
4.3. Discusión de resultados.....	41
5. Sistema de instalación y actualización .....	41
6. Conclusiones.....	42
7. Referencias.....	44
8. Anexos .....	46
8.1. Diagramas UML.....	46
8.2. Listado de código .....	53
8.3. Encuestas Beta-tester .....	54

Figura 1 - Moodle General Settings.....	2
Figura 2 - Visión sistema.....	4
Figura 3 - Diagrama Uso .....	6
Figura 4 - Inicio .....	7
Figura 5 - Nuevo usuario .....	8
Figura 6 - Ayuda.....	8
Figura 7 - Scrum.....	11
Figura 8 - Scrum solapado .....	11
Figura 9 - Diagrama general .....	13
Figura 10 - Diagrama de paquete: aplicacion.....	14
Figura 11 - Diagrama de paquete: actualizador .....	16
Figura 12 - Diagrama de paquete: sincronización.....	17
Figura 13 - Diagrama de paquete: tools.....	18
Figura 14 - Diagrama de actividad: actualizar .....	19
Figura 15 - Diagrama de actividad: proceso actualización.....	20
Figura 16 - Diagrama de actividad: cambio frecuencia .....	21
Figura 17 - Diagrama de actividad: cambiar idioma.....	22
Figura 18 - Diagrama de actividad: crear perfil usuario.....	23
Figura 19 - Diagrama de actividad: editar perfil usuario.....	24
Figura 20 - Diagrama de actividad: sincronizar 01 .....	25
Figura 21 - Diagrama de actividad: sincronizar 02 .....	26
Figura 22 - Diagrama de actividad: sincronizar NAS-TER .....	27
Figura 23 - Diagrama de actividad: sincronización Moodle 01 .....	28
Figura 24 - Diagrama de actividad: sincronización Moodle 02 .....	29
Figura 25 - Prototipo historial .....	31
Figura 26 - Aplicación historial .....	31
Figura 27 - Prototipo entrega .....	32
Figura 28 - Aplicación entrega.....	32
Figura 29 - Prototipo configuración .....	33
Figura 30 - Aplicación configuración .....	33
Figura 31 - Prototipo configuración 2 .....	34
Figura 32 - Aplicación configuración 2 .....	34
Figura 33 - Prototipo nuevo usuario .....	35
Figura 34 - Aplicación nuevo usuario .....	35
Figura 35 - Prototipo editar usuario.....	36
Figura 36 - Aplicación editar usuario.....	36
Figura 37 - Prototipo ayuda.....	37
Figura 38 - Aplicación ayuda .....	37
Figura 39 - tiempos NAS-TER.....	38
Figura 40 - Tiempos Moodle .....	39
Figura 41 - Tiempos Moodle 2.....	40
Figura 42 - Actualización disponible.....	41
Figura 43 - Actualización en proceso .....	42
Figura 44 - Diagrama de clase: aplicacion 01 .....	46
Figura 45 - Diagrama de clase: aplicacion 02 .....	47
Figura 46 - Diagrama de clase: aplicacion 03 .....	48
Figura 47 - Diagrama de clase: aplicacion 04 .....	48
Figura 48 - Diagrama de clase: actualizador 01 .....	49

Figura 49 - Diagrama de clase: actualizador 02 .....	49
Figura 50 - Diagrama de clase: sincronizacion 01 .....	50
Figura 51 - Diagrama de clase: sincronizacion 02 .....	50
Figura 52 - Diagrama de clase: sincronizacion 03 .....	51
Figura 53 - Diagram de clase: tools 01 .....	51
Figura 54 - Diagrama de clase tools 02 .....	51
Figura 55 - Diagrama de clase: tools 03 .....	52

# 1. INTRODUCCIÓN

---

Este proyecto se quiere plantear una mejora del sistema de Moodle del cual se tiene acceso como alumno del campus de Teruel, que forma parte de la Universidad de Zaragoza en adelante Unizar.

El proyecto comienza dado el sistema Moodle [18], el cual es una plataforma de aprendizaje diseñada para proporcionar tanto a educadores, administradores y alumnos un sistema para crear ambientes de aprendizaje. Dicho entorno es comúnmente usado con el fin de enlazar a alumnos y docentes, de manera que proporciona un medio de comunicación de material docente. Permite comunicación entre estos, funcionalidades de foros, encuestas, *wikis* y otras utilidades.

El quid de la cuestión viene de que el mismo sistema notifica tanto a alumnos como a docentes del nuevo material disponible, sin embargo, este proceso es dependiente de los recursos de terceros los cuales desencadenan retrasos en el mismo. Así pues, uno de los objetivos del proyecto es reducir el tiempo entre la publicación de un material por parte del docente y la obtención por parte del usuario.

También, otro aspecto que se quiere tratar es la necesidad de un sistema que permita al usuario disponer de *backups* del material docente, pues en la plataforma Moodle solo se almacena el material docente del año actual y los 3 anteriores. Y además, dicho sistema debe permitir mantener el material actualizado entre dos o más dispositivos del usuario. Para ello vamos a hacer uso de NAS-TER, el cual es un repositorio que ofrece en nuestro caso el campus de Teruel para que sus miembros puedan disponer de un sistema de almacenaje.

Como añadido del proyecto se quiere trabajar la entrega de tareas por parte del alumno en los cursos académicos, con el fin de reducir la necesidad de acceder a la plataforma, además de en un futuro otras funcionalidades.

Para este proyecto en el que se estudiaron diferentes soluciones como hacer un plugin o una aplicación, se ha elegido trabajar en una aplicación que hará uso de web *Scraping* que consiste en la técnica de obtención de información a partir del código html de la web.

Se eligió dicha técnica por ser independiente de la versión de la plataforma Moodle, con poco impacto en las actualizaciones de esta y sin precisar configuración y administración para su uso.

Así pues, vamos a ver en las siguientes secciones del documento una primera etapa de análisis en la que veremos las opciones existentes, un planteamiento de sistema y la redacción de una especificación de requisitos software, en adelante ERS. A continuación, una sección de diseño en la que veremos las metodologías y tecnologías que hicieron posible el proyecto, además de los diseños estático, dinámico y de interfaz. Siguiendo a esto, veremos una sección de prueba y resultado en el que estudiaremos los resultados obtenidos, así como, presentar a los beta-testers y sus evaluaciones. Por último, una sección de conclusiones y referencias.

## 2. ANÁLISIS

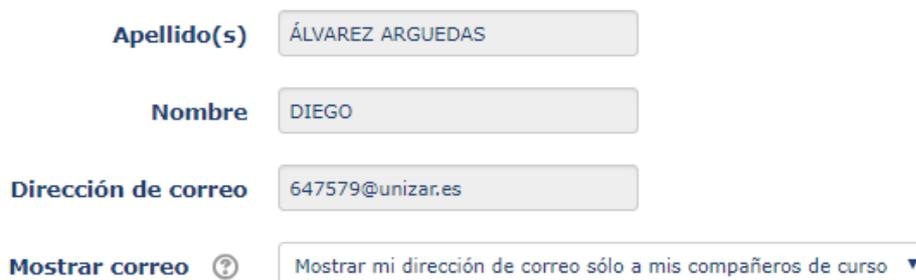
El principal problema que se quiere abordar es minimizar el tiempo entre la subida de un nuevo archivo/tarea por parte de un docente y la consciencia de que existe por parte del alumno.

Así como reducir la necesidad del alumno a revisar si tiene algún nuevo material docente o modificación de estos, en especial de las tareas o entregas de trabajos dirigidos; las cuales se quiere gestionar.

De forma innata Moodle ofrece un sistema de notificaciones, con relación a mensajes en foros. Sin embargo, esto está sujeto a:

- 1) El docente deberá en un foro informar de los cambios realizados en el curso ya sea una nueva entrega u otro material docente.
- 2) El usuario deberá configurar el correo al que desea recibir las notificaciones, por defecto el correo universitario, tendrá que cambiarlo o configurarlo para que este se enlace a otro correo.

### ▼ General



The image shows a screenshot of the Moodle 'General' settings page. It contains four input fields and a dropdown menu:

- Apellido(s)**: A text input field containing the value 'ÁLVAREZ ARGUEDAS'.
- Nombre**: A text input field containing the value 'DIEGO'.
- Dirección de correo**: A text input field containing the value '647579@unizar.es'.
- Mostrar correo**: A dropdown menu with a question mark icon on the left and the selected option 'Mostrar mi dirección de correo sólo a mis compañeros de curso'.

Figura 1 - Moodle General Settings

Este sistema plantea dos grandes problemas:

- 1) El docente debe ser quien redacte la notificación y no asegura llegar a todos los usuarios.
- 2) El servicio de Unizar gestiona la mensajería, y presenta un retraso de 0~5 horas dependiendo de la carga de trabajo del servidor.

### 2.1. Estado del arte

La plataforma Moodle sobre la que se planteó el proyecto fue la Versión 3.3, actualmente para el curso 2018-2019 es la versión 3.5 [26].

Dicha plataforma cuenta con una comunidad que desarrolla *plugins* para la misma plataforma, así como aplicaciones que trabajan con ella. Vamos a ver ambas vertientes existentes, siendo posibles caminos para el desarrollo del proyecto:

### 2.1.1. Plugins

La propia comunidad que desarrolla Moodle plantea una plataforma que gestiona los *plugins* [20], de la versión de estudio los siguientes *plugins* son las que tiene interés dado la proximidad a lo que se quiere plantear.

- Owncloud [23]  
Conecta repositorios privados de owncloud con Moodle, permitiendo creación de copias o espejos de los archivos. Esta más orientado hacia el docente, aunque el alumno podría usarlo.
- Office 365 [21]  
Proporciona acceso a Office 365 como repositorio, al vincular una cuenta Outlook (Hotmail); incluyendo OneDrive, Office Video, Office Delve, documentos *SharePoint* y archivos de grupo asociados a Moodle.
- Evernote [22]  
Permite descargar archivos y notas, además de añadir tags a los mismos de forma que podremos realizar búsquedas.

#### Pros respecto a la aplicación:

- Sincronización con Moodle más eficiente.
- Del lado del servidor, menos gasto de recursos para el usuario.

#### Contras respecto a la aplicación:

- Es necesaria cierta configuración por parte del usuario y/o de la administración Moodle.
- No gestión de entregas.
- No se detalla solución de ciertos problemas originados por los docentes.
- Solo Office 365 parece tener cierta utilidad a la hora de utilizar varios dispositivos.

### 2.1.2. Aplicaciones

Desde la perspectiva de aplicaciones podemos encontrar:

- Android [17]  
Aplicación que soporta varias versiones de Android, es una aplicación genérica que puede conectar a instancias de Moodle. Cumple las funcionalidades de la navegación web, usa el propio navegador para tareas complejas.
- iPhone/iPad (iOS) [31]  
Aplicación genérica que puede conectar a instancias de Moodle. Esta aplicación cumple las mismas funcionalidades que la navegación, destaca como novedad el poder poner en seguimiento los cursos.
- Computador [19]  
Aplicación propia de la plataforma Moodle puede conectar a instancias de Moodle. Esta aplicación cumple las mismas funcionalidades que la navegación web presentadas con el *layout* del sistema.

### Pros respecto a la aplicación:

- Interfaz más acorde a la imagen corporativa de Moodle.
- Cumplimiento de funcionalidades (*quiz*, *wiki*, mensajería, calendario).
- Algunas funcionalidades puntuales, seguimiento y estadísticas sobre el curso.

### Contras respecto a la aplicación:

- No está claro que reflejen los recursos web (o enlaces), que facilita los docentes.
- No queda claro la resolución de problemas humanos de los docentes (duplicado de archivos, mala tipificación).
- El material docente de los cursos parece ser presentado como un listado sin jerarquía de carpetas.
- No consideran el uso de varios dispositivos por parte del usuario, necesidad de terceros en el caso de la aplicación de computador.

Para el curso 2018-2019 la Universidad de Zaragoza ha puesto a disposición de los usuarios GAFE (*Google Apps For Education* – Aplicaciones Google para la educación) es una serie de aplicaciones que se complementan con un gestor documental Alfresco; sincronizará Alfresco con Moodle y el drive de la cuenta utilizada, permitiendo la gestión de documentos y entrega de trabajos, uso de calendario, mensajería y videoconferencias. Alfresco no gestiona algunos documentos como videos/fotos, binarios y ejecutables. Requiere peticiones en formulario para acceder. [6]

## 2.2. Sistema planteado

Vistas las posibles soluciones al sistema en los anteriores puntos, se va a desarrollar una aplicación, en la parte del usuario que haga de intermediario entre el dispositivo o dispositivos de este respecto a Moodle, encargado de sincronizar el material que los docentes ofrezcan (archivos, enlaces de ayuda y Tareas). Además, dicho sistema hará uso de NAS-TER de forma opcional para *backups* y sincronización del contenido entre ambos dispositivos. En la Figura 2 se muestra el diagrama de bloques del sistema.

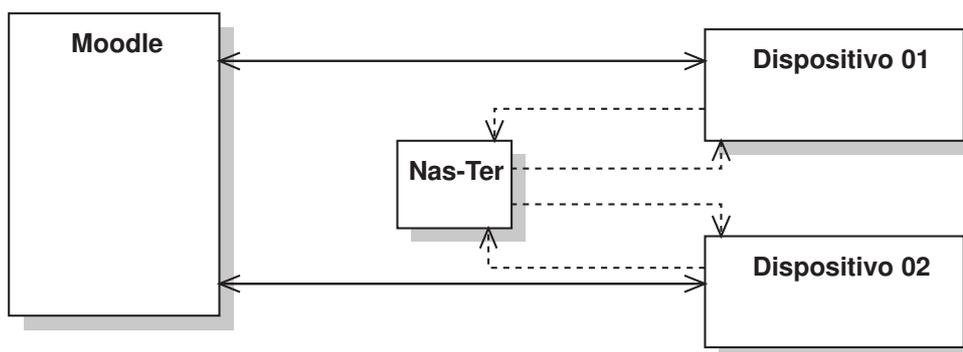


Figura 2 - Visión sistema

Se ha elegido hacer una aplicación porque a diferencia de los plugins no precisa de permisos administrativos del lado de la aplicación, además ya existe una gran cantidad y variedad de plugins que cuentan con una gran comunidad, que los desarrolla.

### 2.2.1. Web scraping

Para la cuestión de cómo obtener la información de Moodle, se ha elegido la técnica de *web scraping*, esta técnica consiste en que, mediante el uso de un software o aplicación, que simula la navegación humana en la web extrayendo la información relevante.

El *web scraping* está muy relacionado con la indexación de la web, la cual indexa la información de la web utilizando un robot y es una técnica universal adoptada por la mayoría de los motores de búsqueda. Sin embargo, el *web scraping* se enfoca más en la transformación de datos sin estructura en la web (como el formato HTML) en datos estructurados que pueden ser almacenados y analizados

El término *web scraping* también está relacionado con la automatización de tareas en la Web, la cual simula la navegación de un humano utilizando un software de computadora. Algunos de sus usos son la comparación de precios en tiendas, la monitorización de datos relacionados con el clima de cierta región, la detección de cambios en sitios webs.

Entre las diversas opciones existentes destacan:

- Jsoup [11]: proyecto que extrae los datos basándonos en *queries* sobre el html.
- Jtidy [7]: herramienta basada en lenguaje XML, se usa principalmente para validar contenidos HTML aunque se puede utilizar para extraer información.
- HtmlUnit [3]: proyecto que simula el navegador permitiendo generar eventos.

Entre las opciones existentes elegiremos la opción de Jsoup, debido al gran respaldo y comunidad que le da soporte, ofrece un rápido y sencillo aprendizaje [12] así como una versión online [10].

### 2.2.2. Casos de uso

Como se ha estado hablando en anteriores puntos del documento, el principal uso o funcionalidad es la sincronización con Moodle, además se quiere plantear el uso de NAS-TER como respaldo de la aplicación. Por consiguiente, deberemos crear un perfil de credenciales validadas respecto a estos sistemas. Así mismo cambiar la frecuencia de sincronización.

Además, se quiere proporcionar la capacidad de cambiar el idioma y actualizarse. Como resultado podemos ver el siguiente diagrama de casos de uso.

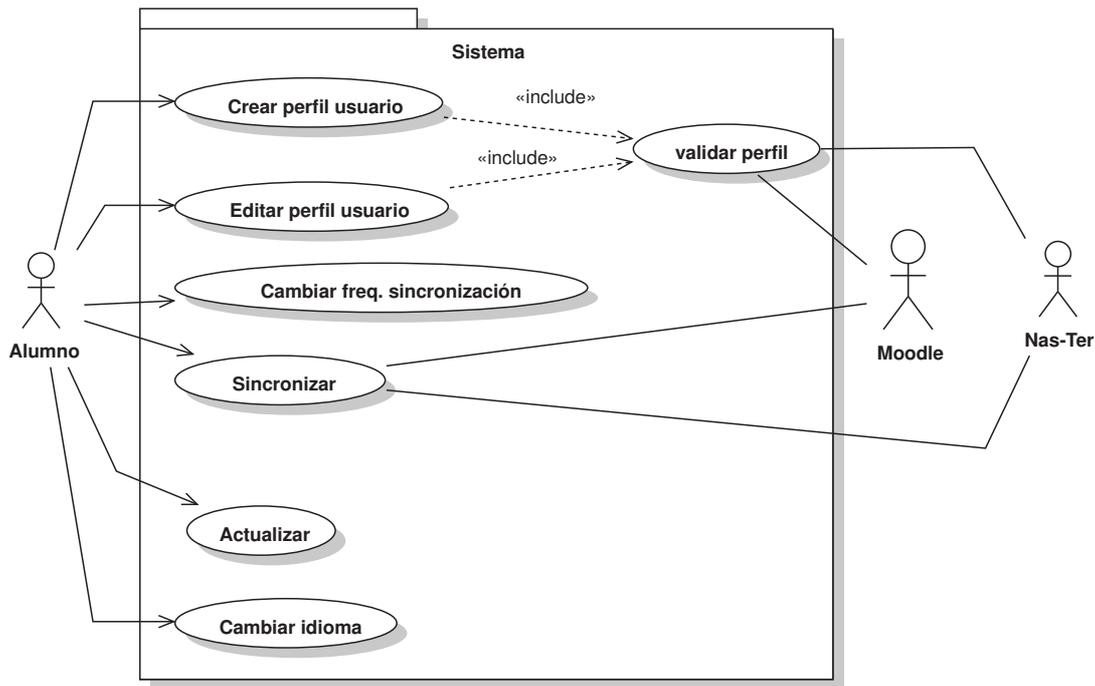


Figura 3 - Diagrama Uso

En la anterior figura vemos un diagrama de los casos de uso del sistema, en dicho sistema encontramos los siguientes actores:

- Alumno: desencadena sobre el sistema todas las funcionalidades.
- Moodle & NAS-TER: actores secundarios que ofrecen o gestionan información, pero que no actúan directamente sobre el sistema.

Sobre estos casos de uso hay que aclarar algunos casos de uso:

- Crear perfil de usuario & Editar perfil de usuario: ambos casos de uso consisten en validar el perfil de usuario contra Moodle y NAS-TER, sin embargo, esto no son acciones que hacen ambos sobre el sistema, sino que es el sistema el que actúa contra ellos.
- Sincronizar: es un caso de uso que consiste en que el sistema sincroniza el contenido Moodle o NAS-TER con el local o el sistema que gestiona la entrega de una tarea. En ambos casos es el sistema que manipula la información que ofrecen Moodle o NAS-TER.

## 2.3. Especificación de Requisitos Software (ERS)

### 2.3.1. Descripción general

#### 1) Perspectiva del producto

##### a. Interfaz de usuario

Distinguimos una única interfaz de usuario que será utilizada por el alumno. Dicha interfaz permitirá al alumno crear un perfil de usuario, así como su modificación; elegir y modificar la ruta del sistema de ficheros sobre la que la aplicación funcionará. El usuario podrá abrir documentos recientemente sincronizados, así como gestionar entregas.

## b. Interfaz Software

Existen varias interfaces software para dar soporte a escenarios distintos.

- Interfaz con Moodle, se encargará de gestionar el contenido asociado al alumno en la herramienta Moodle de Unizar, haciendo uso de *scraping*, para verlo reflejado en la aplicación.
- Interfaz con repositorio, contactará con el repositorio propio de la Unizar, en este caso NAS-TER, y mantendrá el contenido actualizado entre dos o más aplicaciones sujetas al mismo Alumno.

## 2) Características del usuario

La aplicación va destinada hacia un perfil de usuario que tenga algo de familiaridad con las aplicaciones de escritorio sin tener conocimiento técnico respecto al funcionamiento.

## 3) Restricciones

Se usará como lenguaje Java 8. Debido a que es un lenguaje que nos permite funcionar independientemente del S.O, además utilizaremos dicha versión por ser una versión relativamente actual con el fin de evitar problemas futuros de compatibilidad.

La aplicación se debe poder ejecutar en segundo plano.

## 2.3.2. ERS

### 1) Requisitos de Interfaz Externa

#### a. Interfaz de usuario

La primera vez que el alumno inicie la aplicación aparecerá la vista de la Figura 4, en la cual podrá crear un nuevo perfil de usuario Figura 5, o cambiar el idioma. Aunque en la parte superior existan más funcionalidades quedarán bloqueadas hasta que el perfil de usuario a excepción de la sección de ayuda presentada en la Figura 6.

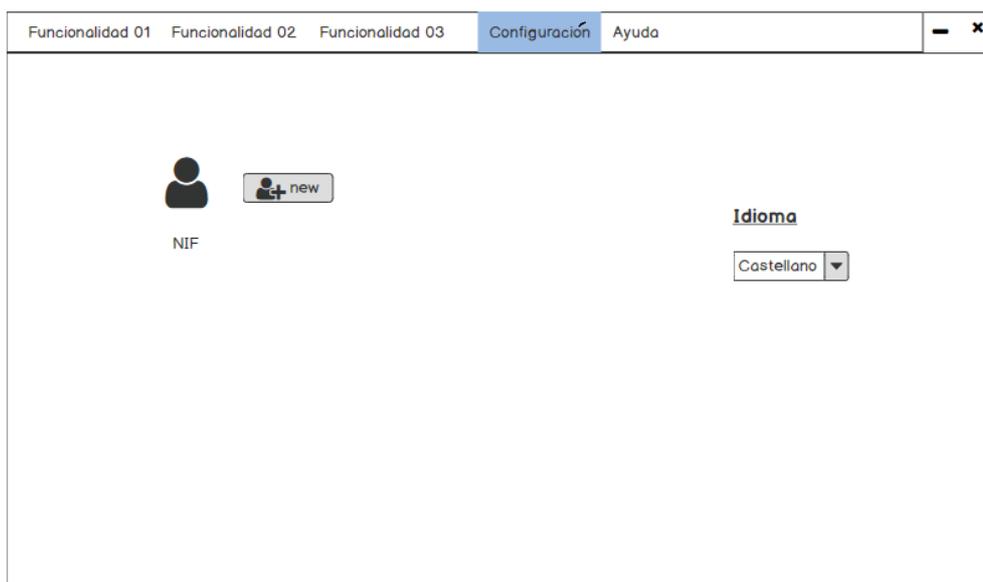


Figura 4 - Inicio

The screenshot shows a window titled "New User" with the following elements:

- NIF**: A text input field.
- Contraseña Moodle**: A text input field.
- Contraseña NAS-TER**: A text input field.
- Ruta de almacenamiento**: A text input field with a file selection icon on the right.
- Aceptar**: A button.
- Cancelar**: A button.

Figura 5 - Nuevo usuario

Para la creación de un nuevo perfil de usuario se pedirán los siguientes campos, ver figura 5:

- NIF: identificador proporcionado por la universidad, que representa al usuario.
- Contraseña administrativa: contraseña con la que el usuario accede a Moodle.
- Contraseña de correo: contraseña con la que el usuario accede al repositorio NAS-TER.
- Ruta: ruta local donde la aplicación reflejará el contenido Moodle.

Para la creación y aceptación del perfil, los datos serán validados correctamente e informado en caso contrario.

The screenshot shows a window titled "Ayuda" with the following elements:

- Menu Bar**: Funcionalidad 01, Funcionalidad 02, Funcionalidad 03, Configuración, Ayuda.
- Créditos**: Text label.
- Número de versión**: Text label.
- Actualizar**: Button.
- Link ayuda/wiki**: Text label.

Figura 6 - Ayuda

En la sección de ayuda Figura 6, el usuario podrá acceder a un link de ayuda o *wiki*, ver los créditos de la aplicación y además ver la versión de la aplicación así como actualizarla.

## **2) Requisitos funcionales**

### **a. Referentes a Moodle**

**RQM-01.** El programa gestionará el contenido al que el perfil Estudiante tiene acceso en Moodle tales como ficheros, enlaces, contenido multimedia.

**RQM-02.** El programa mantendrá un historial de los últimos contenidos gestionado en RQM-01 agrupado por materia.

### **b. Referentes a Seguridad**

**RQS-01.** El programa realizará todo intercambio de paquete de datos con algún cifrado de seguridad.

**RQS-02.** El programa almacenará los datos del Estudiante de forma segura.

### **c. Referentes a Funcionalidad**

**RQF-01.** El programa será capaz de gestionar entrega de documentos.

**RQF-02.** El programa será capaz de dar soporte a otras funcionalidades de Moodle.

**RQF-03.** El programa será capaz de actualizarse, tal como vemos en la Figura 6 - Ayuda en la sección 2.3.2.

**RQF-04.** El programa lanzará un proceso de actualización/comprobación hacia RQM-01 y RQD-01 cada cierta frecuencia ajustable por el usuario.

**RQF-05.** El programa contará con un tiempo mínimo que no podrá reducir referente a RQF-04.

**RQF-06.** En el caso de que el usuario ajuste la frecuencia, la siguiente actualización será conforme al ajuste y la hora actual.

**RQF-07.** El programa permitirá al usuario forzar una actualización.

## **3) Requisitos no funcionales**

**RQN-01.** El programa deberá crear una jerarquía de carpetas que recogerán el contenido de los cursos siguiendo el orden: Curso > Asignatura > Matrícula > Orden Moodle

**RQN-02.** En el caso de que el programa no pudiera conectar con el repositorio RQD-01 el programa seguirá sincronizándose con la instancia Moodle RQM-01.

**RQN-03.** En el caso de que el programa no pueda establecer conexión con la instancia Moodle RQM-01, se sincronizará con el repositorio RQD-01.

#### **4) Restricciones de diseño**

**RQD-01.** El programa será capaz de sincronizarse con el repositorio NAS-TER, mediante el protocolo Webdav.

**RQD-02.** Haciendo uso del repositorio, el programa guardará contenido de las materias del año actual. Sujeto a RQN-01.

**RQD-03.** El programa almacenará un único perfil de usuario, el cual se inicializará como se vio en la Figura 5 en la sección 2.3.2.

**RQD-04.** Para la creación y aceptación del perfil, los datos serán validados correctamente e informado en caso contrario.

**RQD-05.** El programa permitirá cambiar las credenciales del perfil de usuario, sujeto a RQD-04.

**RQD-06.** El programa permitirá cambiar la ruta sobre la que trabaja condicionado a que la nueva ruta disponga de espacio necesario para almacenar la totalidad de los ficheros creados por RQN-01.

**RQD-07.** El programa permitirá al usuario cambiar el idioma con el que funciona.

**RQD-08.** El programa será capaz de ejecutarse en segundo plano.

**RQD-09.** El programa hará uso de la técnica conocida como web *scraping* para cumplir RQM-01.

## 3. DISEÑO

### 3.1. Metodología usada

Para el desarrollo del proyecto se optó por elegir una metodología incremental, también llamada metodología ágil [4], en concreto una versión de *Scrum* [28] (un solo equipo de desarrollo compuesto por una persona).

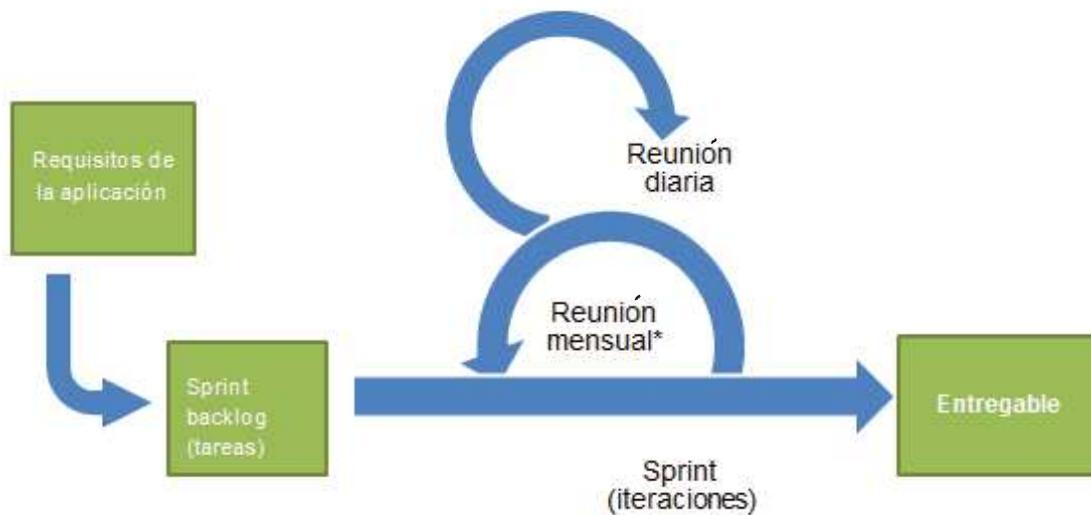


Figura 7 - Scrum

El principio de la metodología Scrum es a partir de unos requisitos de la aplicación, realizados al principio del proyecto, se van desarrollando *sprints* en los cuales se establecen unos objetivos y al finalizar el mismo da como resultado una iteración del proyecto. Sobre esta misma iteración se realiza otra iteración, así hasta tener un proyecto finalizado.

Esta metodología de trabajo permite trabajar los *sprints* de forma secuencial, secuencial con solapamientos o solapado. En nuestro caso secuencial con solapamiento, esto consiste en que cada sprint de trabajo que define unos puntos de trabajo o tareas a completar, al ser una única persona en desarrollo se permite la posibilidad de que a finales del sprint preparar el proyecto para el siguiente sprint; esta forma es la idea o se le conoce como secuencial con solapamiento, de la que podemos ver una idea en la siguiente figura.



Figura 8 - Scrum solapado

## 3.2. Tecnologías usadas

### 3.2.1. Java

Se ha elegido como tecnología de programación Java. Para este proyecto trabajaremos con la versión 1.8.0\_91 [16], debido a su filosofía WORA (“escribelo una vez, ejecútalo en cualquier sitio” del inglés “*write once, run anywhere*”) que permite codificar una aplicación independientemente del sistema, por su facilidad en el tratamiento de *threads*.

Además del lenguaje de programación Java se utiliza para complementar la parte de interfaz gráfica, JavaFx 2.0 [14] [15] siendo esta también una tecnología de Oracle.

### 3.2.2. Herramientas utilizadas

Para la realización de este proyecto se han utilizado las siguientes herramientas o IDE's

- **Netbeans** [30]  
Netbeans es un entorno de desarrollo integrado libre y gratuito orientado principalmente para el lenguaje de programación Java, para este proyecto se ha utilizado la versión 8.1 [25]
- **Balsamic Mockups** [2]  
Balsamic Mockups es una herramienta que permite el diseño de widgets o maquetas de interfaz utilizando un editor de *drag & drop*, para este proyecto se ha utilizado su servicio en nube Balsamic Cloud [1]
- **Scene Builder** [27]  
Scene Builder es una herramienta que permite del diseño de interfaces JavaFx mediante un editor de *drag & drop*, incluyendo la facilitación de hojas de estilo; para este proyecto se ha utilizado la versión 9.0.1.
- **Install4j** [24]  
Install4j es una aplicación multiplataforma que genera instaladores para distintos Sistemas Operativos a partir de un archivo Jar, para este proyecto se ha utilizado la versión 7.0.7 [5]

## 3.3. Diseño estático

Vamos a ver los diagramas de paquetes. Primero veremos diagrama general que mostrara todo el sistema existente, y después veremos cada sección del mismo explicando las clases y su función.



En la aplicación distinguiremos cuatro paquetes principales, que veremos en detalle más adelante, estos son:

- actualizador: paquete que contiene las clases orientadas a la actualización, respecto a versión, de la aplicación. Paquete extraído del sistema de actualización de [9] ver apartado 5.
- aplicacion: paquete que contiene el núcleo de la aplicación, arranque de interfaz, eventos y estructura de datos asociada.
- sincronizacion: paquete que gestiona la sincronización del sistema, contra Moodle y NAS-TER.
- tools: paquete que contiene una serie de herramientas o útiles para la aplicación, como un sistema de log, internalización (lenguaje) y almacén de datos.

### 3.3.2. Diagrama de paquete: aplicacion

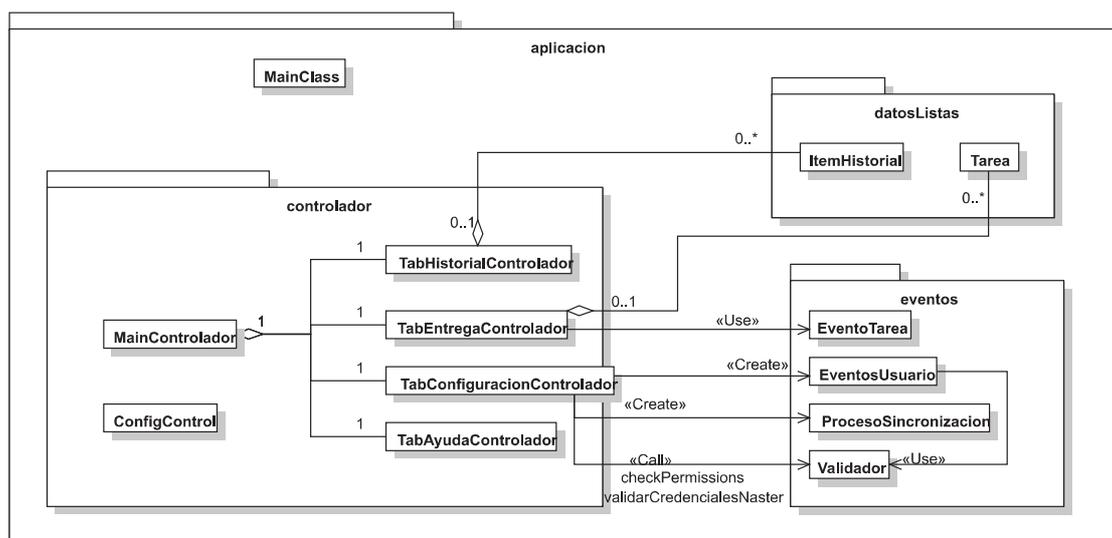


Figura 10 - Diagrama de paquete: aplicacion

En el paquete aplicacion, encontramos las siguientes clases:

- MainClass: clase de arranque del sistema, prepara y gestiona *SystemTray*, el lanzamiento del proceso de actualización, proporciona al controlador el recurso de idioma y hostService. Ver en el Anexo I Figura 44 - Diagrama de clase: aplicacion 01.
- MainControlador: clase de control de interfaz, la cual incluye a las clases control hijas, sirve de intermediario entre los otros controladores si estos generarán algún cambio en sus hermanas, así como intermediario de MainClass. Además, gestionará la interacción de otras clases con los controladores. Ver en el Anexo I Figura 45 - Diagrama de clase: aplicacion 02.
- ConfigControl: clase a la que pueden acceder los controladores, contiene cadenas de texto dependientes del idioma. Ver en el Anexo I Figura 45 - Diagrama de clase: aplicacion 02.

- **TabHistorialControlador**: controlador de la interfaz historial, crea, almacena y gestiona eventos asociados a **ItemHistorial**. Ver en el Anexo I Figura 45 - Diagrama de clase: aplicacion 02.
- **TabEntregaControlador**: controlador de la interfaz de tareas, presenta una tabla en la cual el sistema podrá añadir, archivar, guardar y cargar **Tarea**. Ver en el Anexo I Figura 45 - Diagrama de clase: aplicacion 02.
- **TabConfigController**: controlador de la interfaz de configuración, genera eventos de adición y edición de un perfil de usuario, cambio sobre el directorio de trabajo, uso de NAS-TER, cambio de idioma, cambio de frecuencia de actualización y actualización inmediata. Ver en el Anexo I Figura 45 - Diagrama de clase: aplicacion 02.
- **TabHelpController**: controlador de la interfaz de ayuda, informa de la versión y permite actualizar, proporciona links de ayuda así como información de contacto con el soporte. Ver en el Anexo I Figura 45 - Diagrama de clase: aplicacion 02.
- **ItemHistorial**: clase que representa un recurso Moodle descargado. Ver en el Anexo I Figura 46 - Diagrama de clase: aplicacion 03.
- **Tarea**: clase que representa una tarea Moodle y almacena información útil de la misma. Ver en el Anexo I Figura 46 - Diagrama de clase: aplicacion 03.
- **EventoTarea**: clase que representa el evento de gestionar la entrega de un fichero en una tarea Moodle. Ver en el Anexo I Figura 47 - Diagrama de clase: aplicacion 04.
- **EventoUsuario**: clase que representa el evento de crear o editar el perfil de usuario, informando de su resultado a **MainControlador**. Ver en el Anexo I Figura 47 - Diagrama de clase: aplicacion 04.
- **ProcesoSincronizacion**: clase que representa el evento de lanzar una sincronización sobre los cursos Moodle del año actual, así como la sincronización de estos sobre NAS-TER si fuera necesario; haciendo uso del paquete **sincronizacion**. Ver en el Anexo I Figura 47 - Diagrama de clase: aplicacion 04.
- **Validador**: clase que valida credenciales Moodle y NAS-TER, así como los permisos del directorio de trabajo. Ver en el Anexo I Figura 47 - Diagrama de clase: aplicacion 04.

### 3.3.3. Diagrama de paquete: actualizador

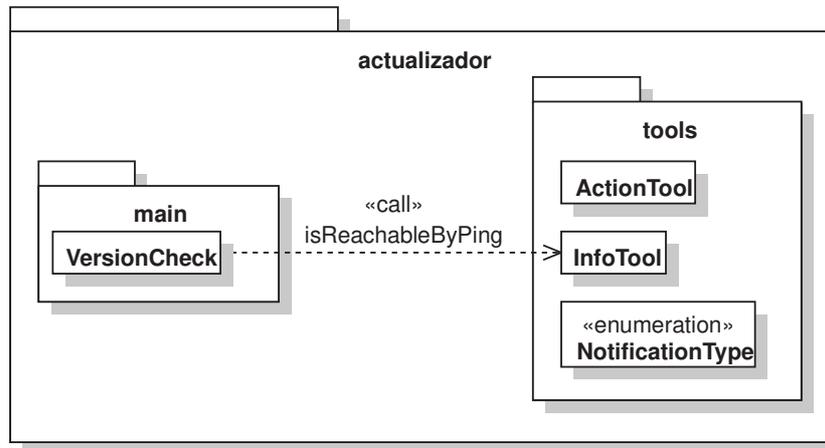


Figura 11 - Diagrama de paquete: actualizador

En el paquete actualizador, encontramos las siguientes clases:

- **VersionCheck**: clase que comprobará el estado del servidor de actualización y buscará el número de la última versión. Ver en el Anexo I Figura 48 - Diagrama de clase: actualizador 01.
- **ActionTool**: clase permite mostrar notificaciones al usuario. Ver en el Anexo I Figura 49 - Diagrama de clase: actualizador 02.
- **InfoTool**: proporciona información útil para la actualización (apenas usado). Ver en el Anexo I Figura 49 - Diagrama de clase: actualizador 02.
- **NotificationType**: sobre los tipos de notificación que se le pueden presentar al usuario. Ver en el Anexo I Figura 49 - Diagrama de clase: actualizador 02.

### 3.3.4. Diagrama de paquete: sincronizacion

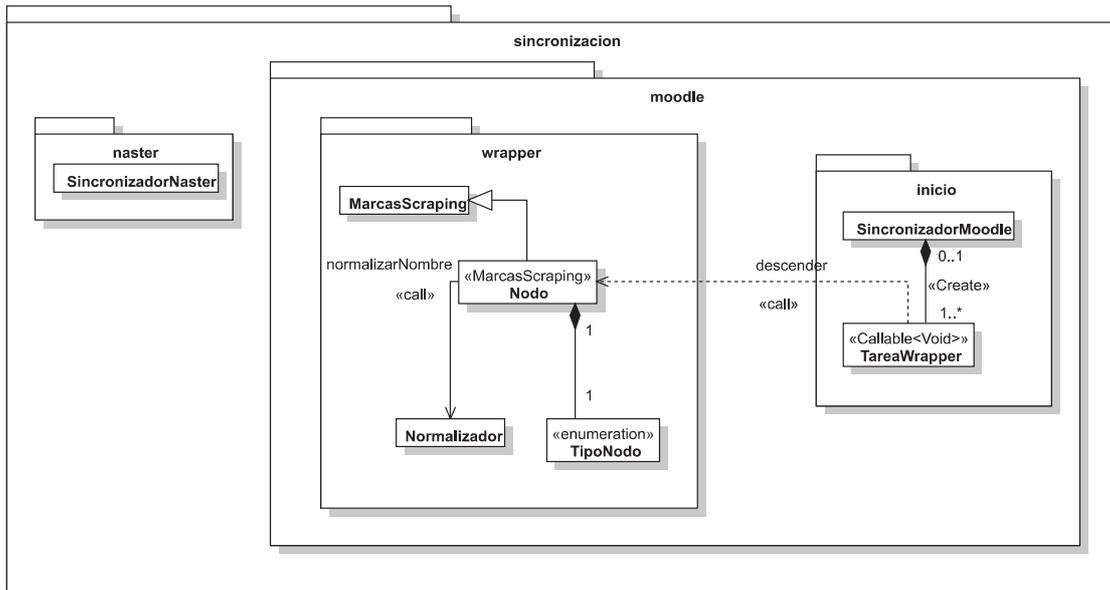


Figura 12 - Diagrama de paquete: sincronización

En el paquete sincronizacion, encontramos las siguientes clases:

- SincronizadorMoodle: clase que lanzará la sincronización sobre Moodle, autenticando en este, creando TareaWrapper por cada curso del año actual al que se tenga acceso. Ver en el Anexo I Figura 50 - Diagrama de clase: sincronizacion 01.
- TareaWrapper: clase utilizada para lanzar en *multithread* la sincronización, contiene un nodo que representa el curso. Ver en el Anexo I Figura 50 - Diagrama de clase: sincronizacion 01.
- MarcasScraping: clase con una serie de constantes utilizadas para el *scraping*. Ver en el Anexo I Figura 51 - Diagrama de clase: sincronizacion 02.
- Nodo: clase que conecta y procesa una dirección URL Moodle, generando un árbol de nodos hijos de los cuales cada uno representa un recurso Moodle. Una vez finalizado el árbol descargará o informará a MainControlador al respecto. Ver en el Anexo I Figura 51 - Diagrama de clase: sincronizacion 02.
- Normalizador: clase que normaliza el nombre de los Nodos, para ser utilizado en el Sistema de Ficheros. Ver en el Anexo I Figura 51 - Diagrama de clase: sincronizacion 02.
- TipoNodo: enum sobre el tipo de recurso Moodle que representa el Nodo. Ver en el Anexo I Figura 51 - Diagrama de clase: sincronizacion 02.
- SincronizadorNaster: clase que gestionará la sincronización sobre NAS-TER. Ver en el Anexo I Figura 52 - Diagrama de clase: sincronizacion 03.

### 3.3.5. Diagrama de paquete: tools

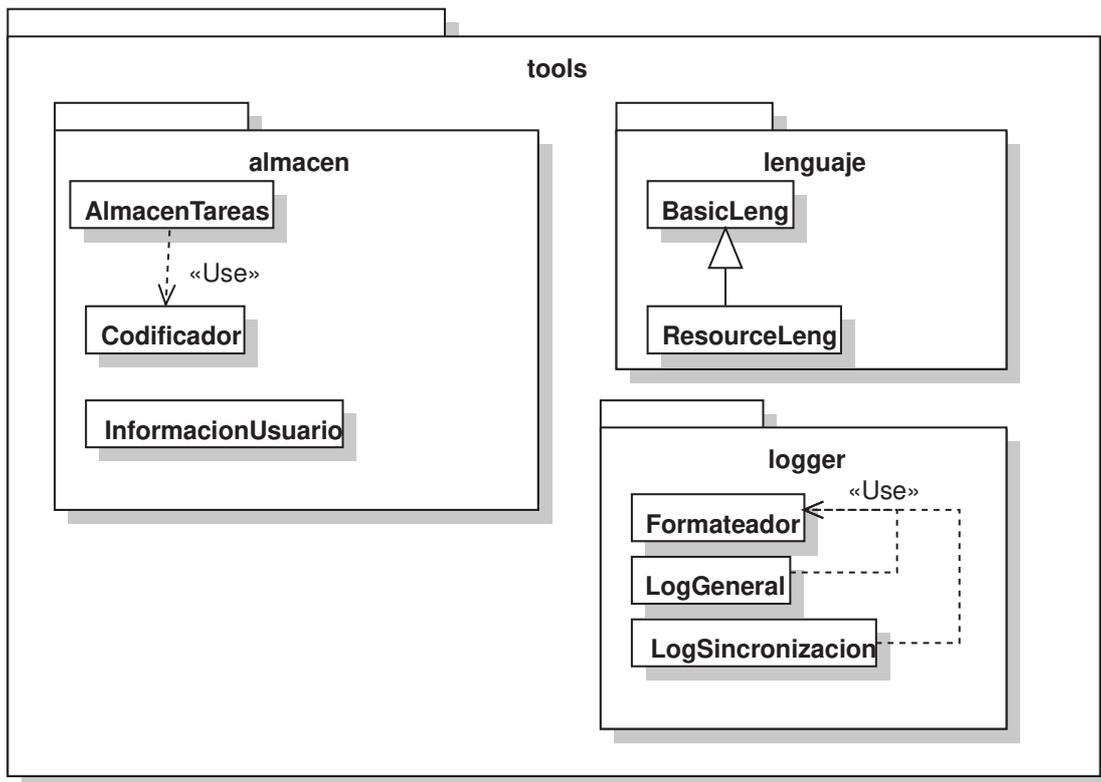


Figura 13 - Diagrama de paquete: tools

En el paquete tools, encontramos las siguientes clases:

- **AlmacenTareas**: clase que gestiona el guardado y recuperación de las tareas asociadas al perfil de usuario que tiene acceso en Moodle. Ver en el Anexo I Figura 55 - Diagrama de clase: tools 03.
- **Codificador**: clase utilizada para codificar/descodificar sobre un fichero una colección de Tarea, utilizado por AlmacenTareas. Ver en el Anexo I Figura 55 - Diagrama de clase: tools 03.
- **InformacionUsuario**: clase gestiona el almacenado y recuperación de los datos encriptados de un perfil de usuario. Ver en el Anexo I Figura 55 - Diagrama de clase: tools 03.
- **BasicLeng**: clase que contiene cadenas de texto constantes para la comunicación entre la aplicación y el sistema de actualización. Ver en el Anexo I Figura 53 - Diagrama de clase: tools 01.
- **ResourceLeng**: clase que contiene cadenas de texto constantes utilizadas para la internalización del sistema (utilizar diferentes lenguajes). Ver en el Anexo I Figura 53 - Diagrama de clase: tools 01.
- **Formateador**: clase que proporciona el formato/modelo para los logs. Ver en el Anexo I Figura 54 - Diagrama de clase tools 02.

- LogGeneral: clase que gestiona un archivo de Log, utilizada por la mayor parte de la aplicación. Ver en el Anexo I Figura 54 - Diagrama de clase tools 02.
- LogSincronizacion: clase que gestiona un archivo de Log, utilizada por las clases relacionadas con la sincronización. Ver en el Anexo I Figura 54 - Diagrama de clase tools 02.

## 3.4. Diseño dinámico

### 3.4.1. Diagrama de actividad: actualizar

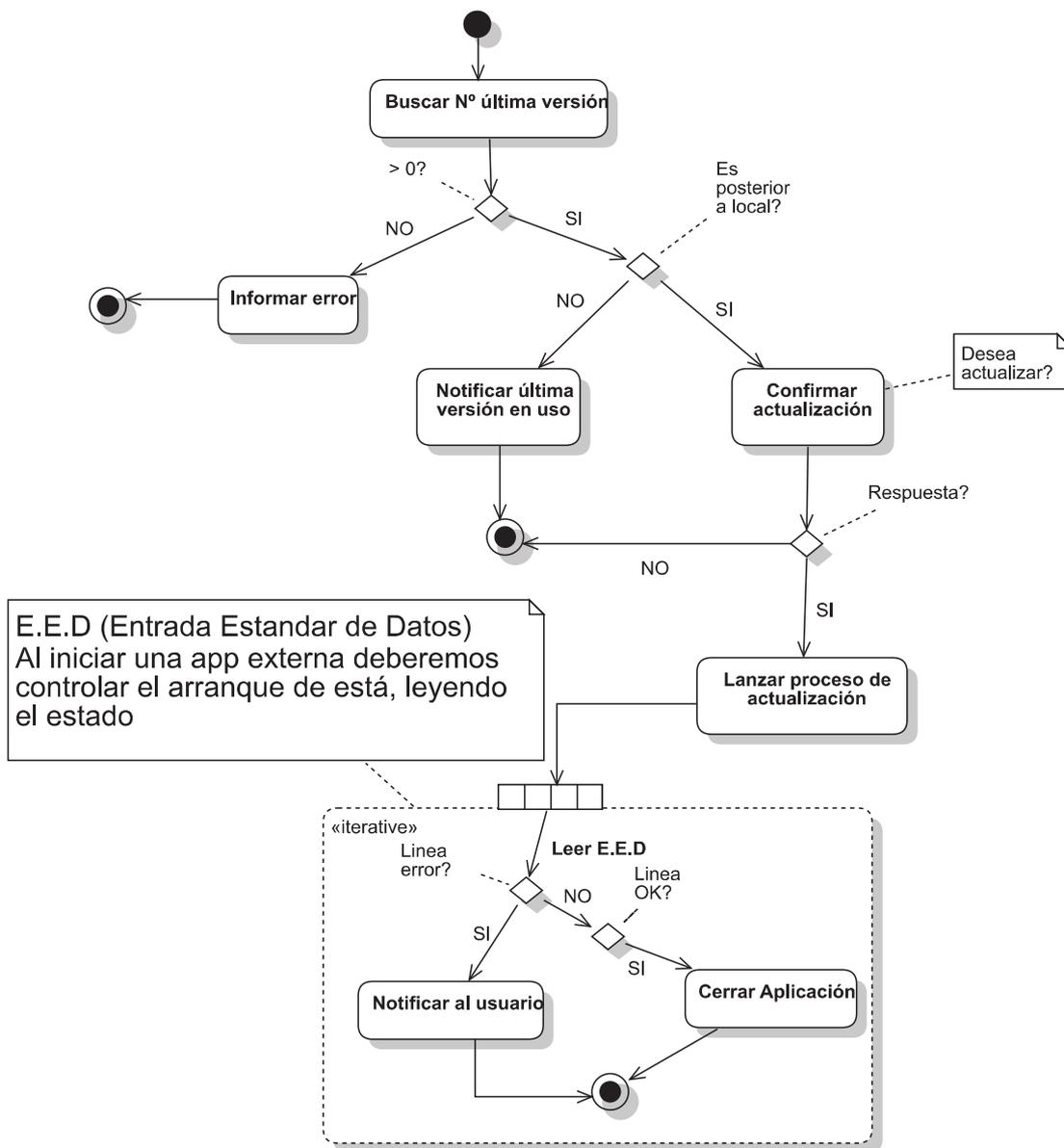


Figura 14 - Diagrama de actividad: actualizar

En el proceso de actualización, el usuario quiere actualizar el sistema, lo primero que ocurre es el sistema busca el **nº de la última actualización**, tras lo cual se **comprueba si es una versión posterior**, en caso de que así fuera se preguntaría **confirmación para actualizar**; en el caso que así se desee la aplicación **lanzará un proceso de actualización** (lanzando un .jar independiente)

entonces se leerá E.E.D para comprobar el arranque del mismo proceso, siendo un **caso de error** o un **caso correcto**.

### 3.4.2. Diagrama de actividad: proceso de actualización

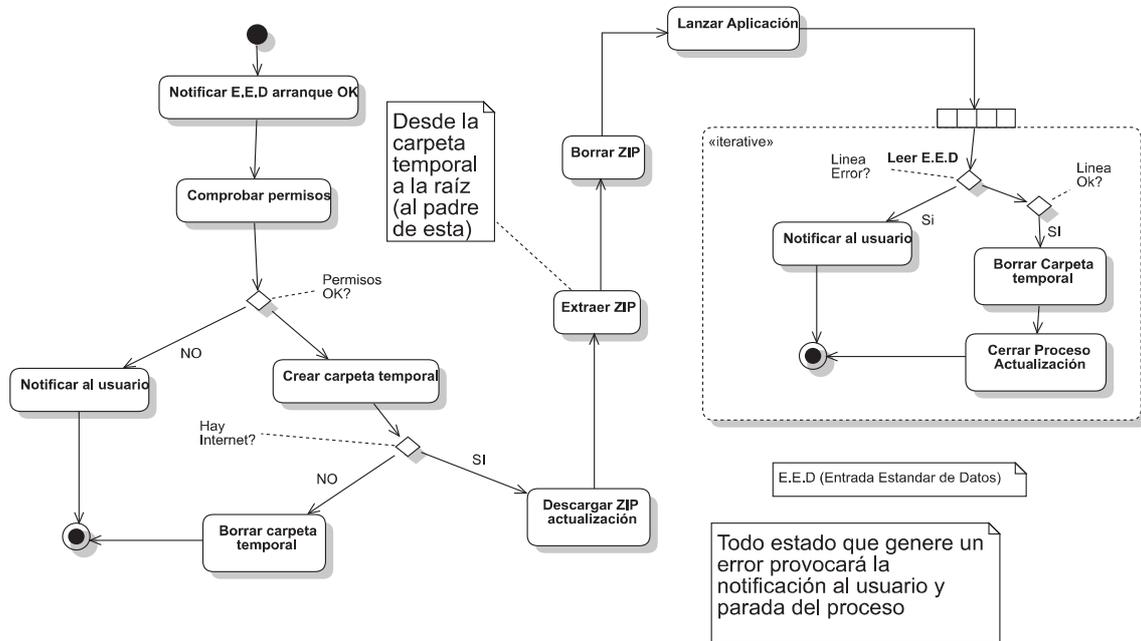


Figura 15 - Diagrama de actividad: proceso actualización

En el proceso de actualización, de la aplicación .jar externa al sistema, lo primero en ocurrir será **notificar en la E.E.D el correcto lanzamiento** tras lo cual se **comprobarán los permisos** de escritura/lectura sobre el directorio actual, de esta forma se **creará una carpeta temporal** y en la que si **disponemos de conexión** procederemos a **descargar la actualización**, descomprimir y reemplazar archivos; una vez hecho esto volveremos a **lanzar la aplicación** leyendo la E.E.D para comprobar el arranque actuando en **caso correcto** o en **caso de error**.

### 3.4.3. Diagrama de actividad: cambio de frecuencia

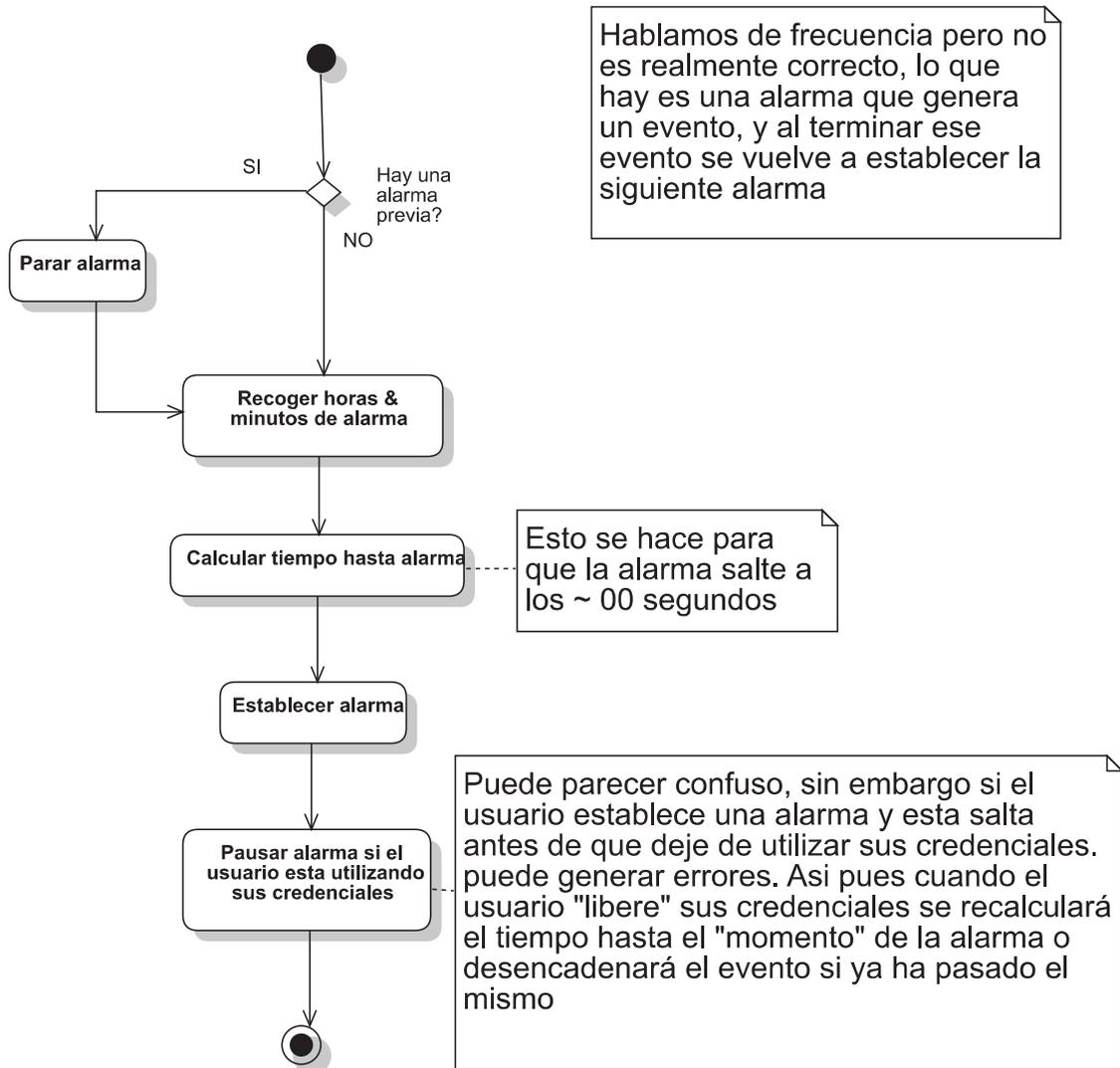


Figura 16 - Diagrama de actividad: cambio frecuencia

En el proceso de actualización, el usuario quiere actualizar el sistema, lo primero que ocurre es el sistema busca el **nº de la última actualización**, tras lo cual se **comprueba si es una versión posterior**, en caso de que así fuera se preguntaría **confirmación para actualizar**; en el caso que así se desee la aplicación **lanzaré un proceso de actualización** (lanzando un .jar independiente) entonces se leerá E.E.D para comprobar el arranque del mismo proceso, siendo un **caso de error** o un **caso correcto**.

### 3.4.4. Diagrama de actividad: cambio de idioma

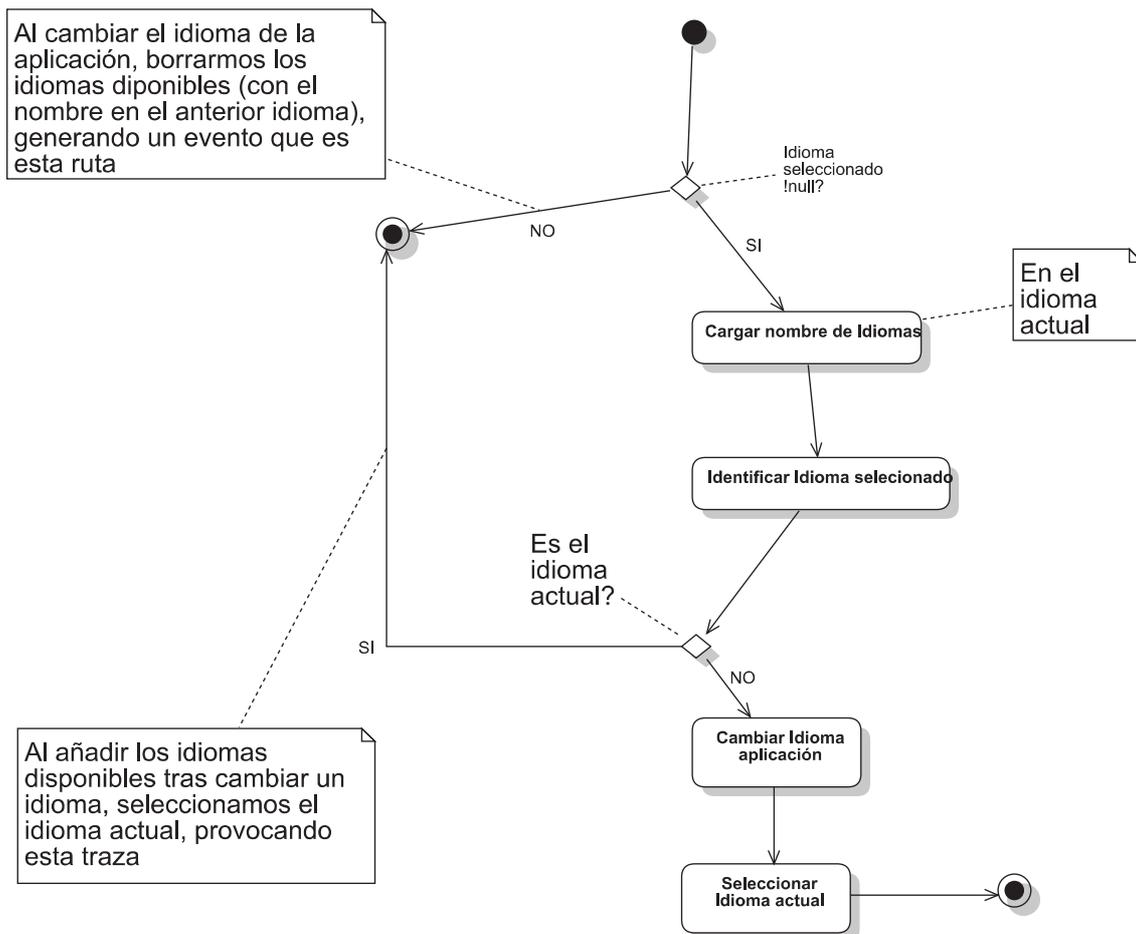


Figura 17 - Diagrama de actividad: cambiar idioma

En el proceso de cambiar el idioma, generado por un evento de selección de idioma en la aplicación, si la selección es válida lo primero será **cargar los idiomas** disponibles en el idioma actual lo que permitirá **identificar el idioma seleccionado**; si es un idioma distinto se **cambiará el idioma** del sistema y **elegiremos el idioma seleccionado** (en su propio idioma). Al elegir el idioma deseado en su idioma, se genera un evento redundante por eso la distinción de no ser el idioma actual.

### 3.4.5. Diagrama de actividad: nuevo perfil de usuario & editar perfil de usuario

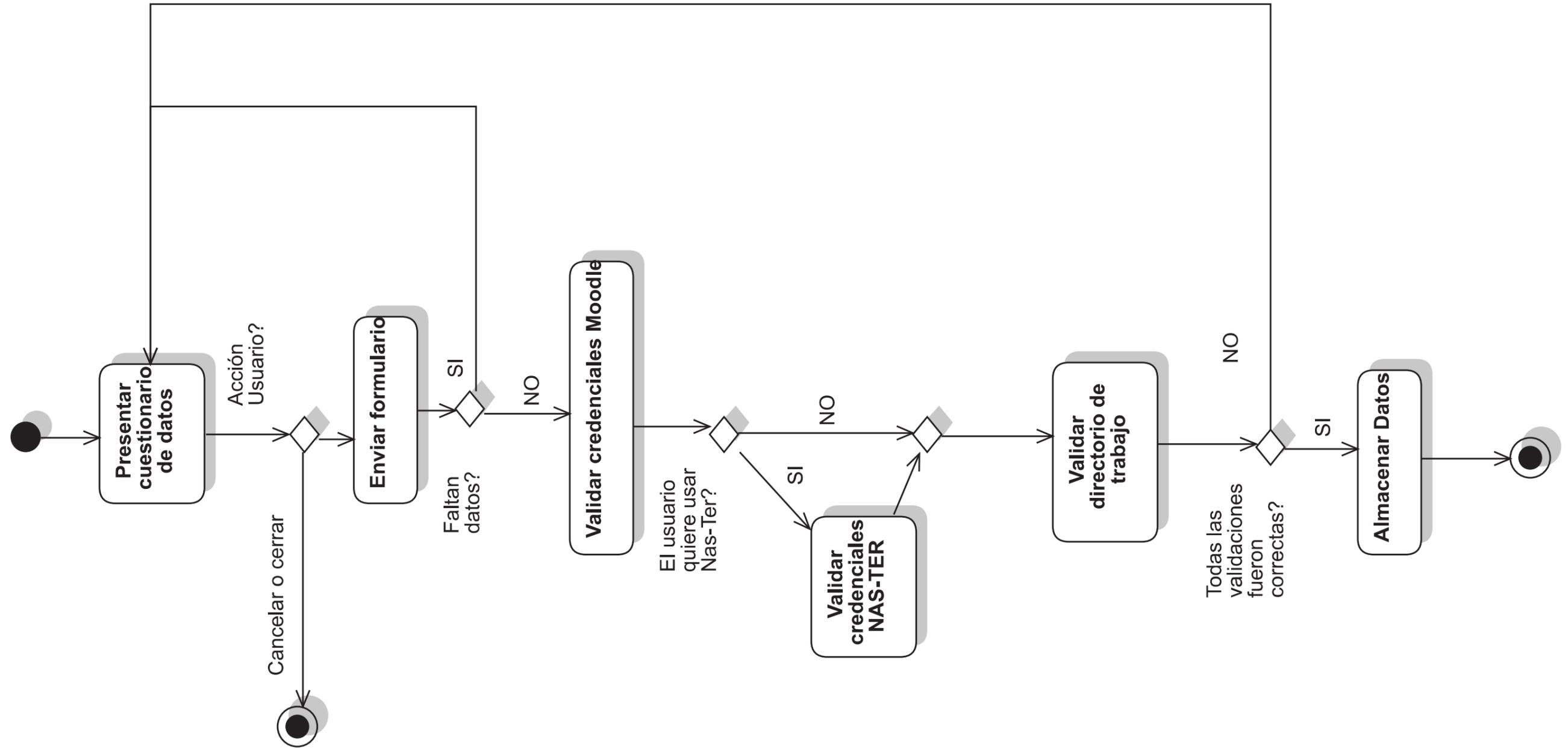


Figura 18 - Diagrama de actividad: crear perfil usuario

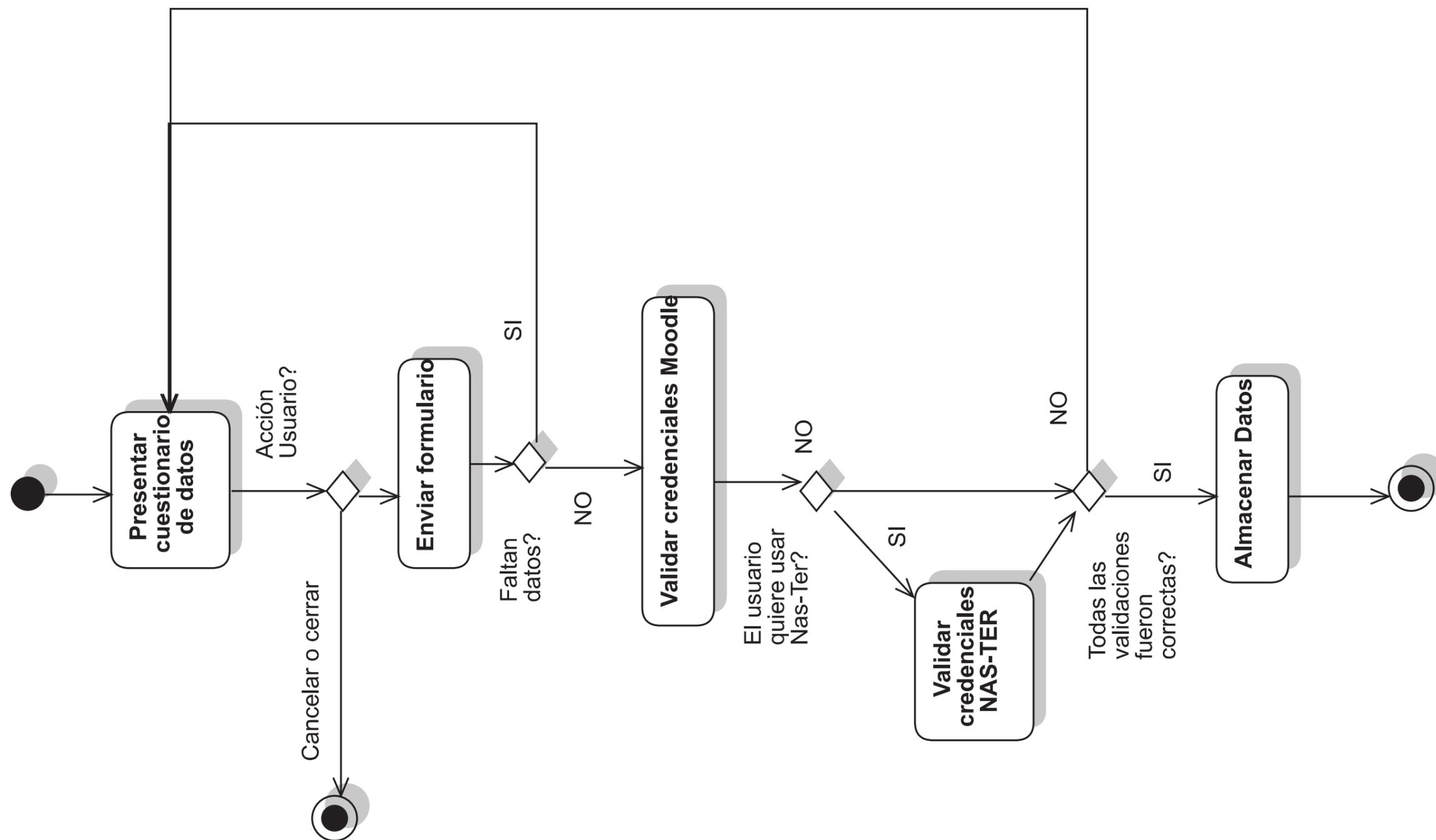


Figura 19 - Diagrama de actividad: editar perfil usuario

En los procesos de creación y edición de usuario se distinguen en el hecho de que en la creación de perfil durante las comprobaciones se comprueba los permisos de escritura/lectura en el directorio de trabajo. Así pues, dichos procesos consisten en **presentar un formulario** con los datos deseados, dicho formulario debe tener todos los **campos rellenos** y cuando ocurra esto se **comprobarán las credenciales Moodle** del usuario, si el usuario **quisiera usar NAS-TER se comprobarían** estas. En el caso de que todas las validaciones fueran correctas se procedería a **almacenar el perfil de usuario** resultante.

### 3.4.6. Diagrama de actividad: sincronizar

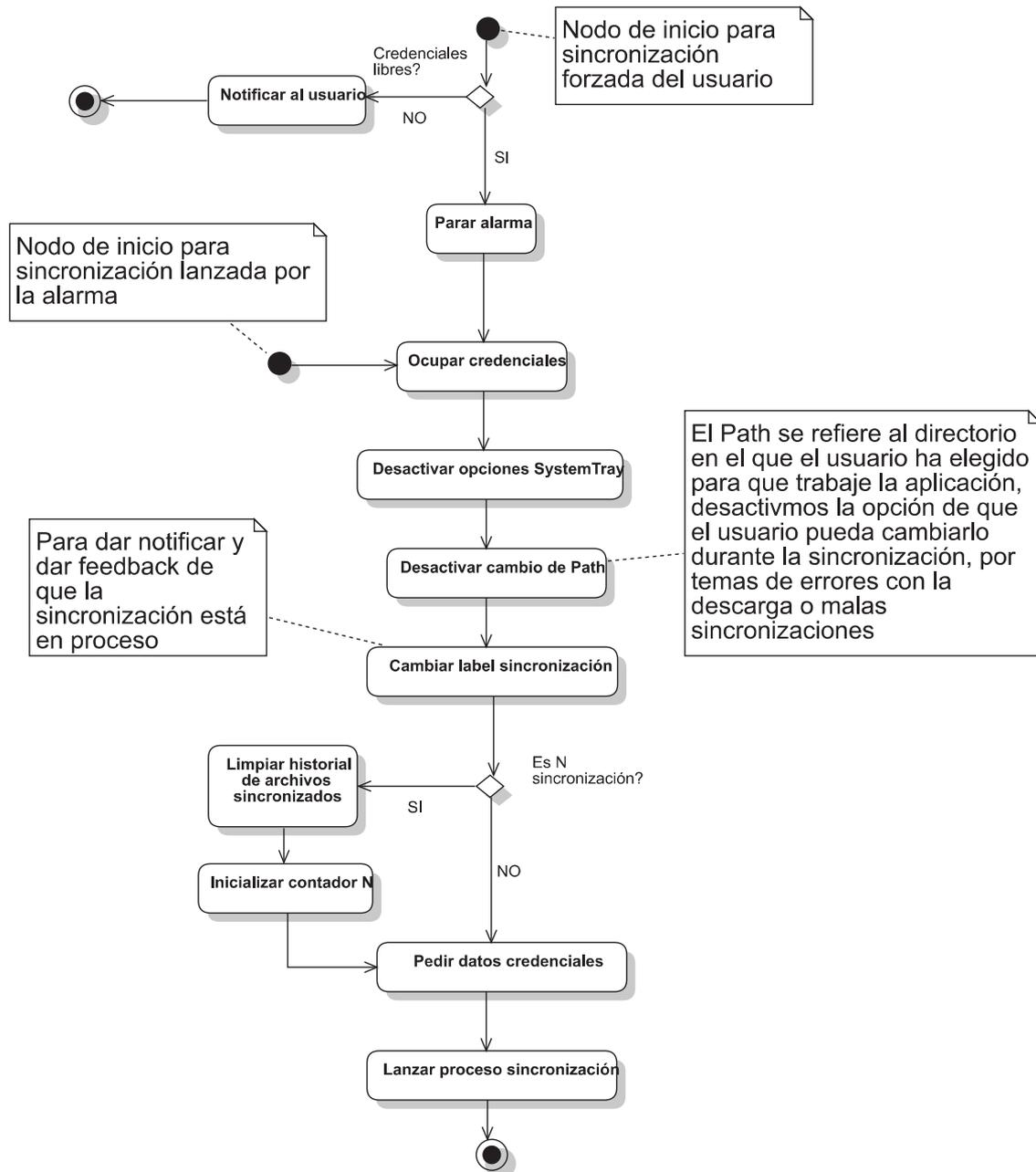


Figura 20 - Diagrama de actividad: sincronizar 01

En el proceso de actualización distinguimos dos nodos de inicio:

- Sincronización forzada del usuario: el usuario fuerza una sincronización, si las credenciales están libres se **parará la alarma** y se **ocuparán las credenciales**, en caso contrario se **notificará al usuario**.
- Sincronización lanzada por alarma: la alarma establecida que indica la frecuencia de sincronización indica un inicio de sincronización, **ocupando las credenciales**.

Una vez ocupadas las credenciales ambas actividades son iguales, se **desactivará opciones de SystemTray** para evitar fallos de ejecución y además se **desactivará el cambio de Path** para evitar fallos de sincronización. Se notificará el inicio de la sincronización **cambiando la label de sincronización**, si esta fuera la N sincronización **limpiaríamos el historial de archivos sincronizados e inicializaríamos el contador N**; sino el sistema **pedirá las credenciales** al almacenaje y se **lanzaría un proceso de sincronización**.

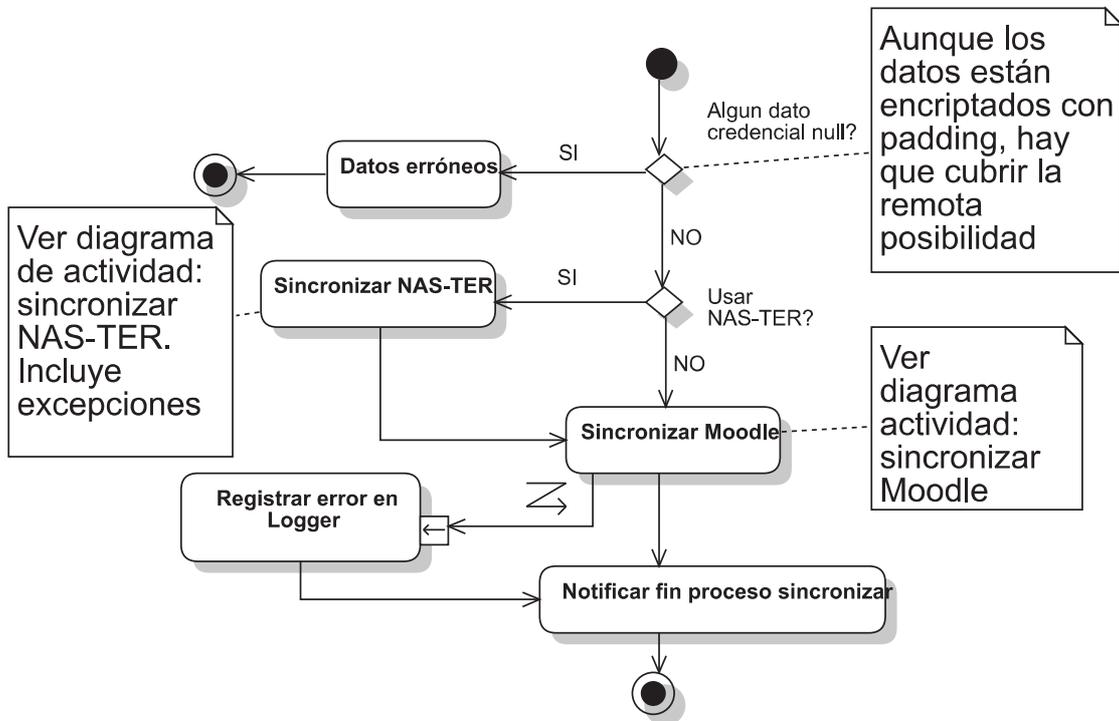


Figura 21 - Diagrama de actividad: sincronizar 02

A continuación, vamos a ver el proceso de sincronización. Este proceso es grande y complejo por lo que se ha dividido en varios diagramas. En este diagrama vemos la parte general en la que iniciamos el proceso y debemos **cargar los datos guardados** del sistema y si no hubiera error, primero **sincronizaríamos con NAS-TER** si el usuario desea utilizar el servicio; al finalizar o en caso de que no **sincronizaríamos con Moodle**. Y al acabar la sincronización con Moodle **notificaríamos fin del proceso** al usuario. Ambas sincronizaciones pueden verse interrumpidas y detenidas al producirse un fallo con lo que se **registrará el error**.

### 3.4.7. Diagrama de actividad: sincronizar NAS-TER

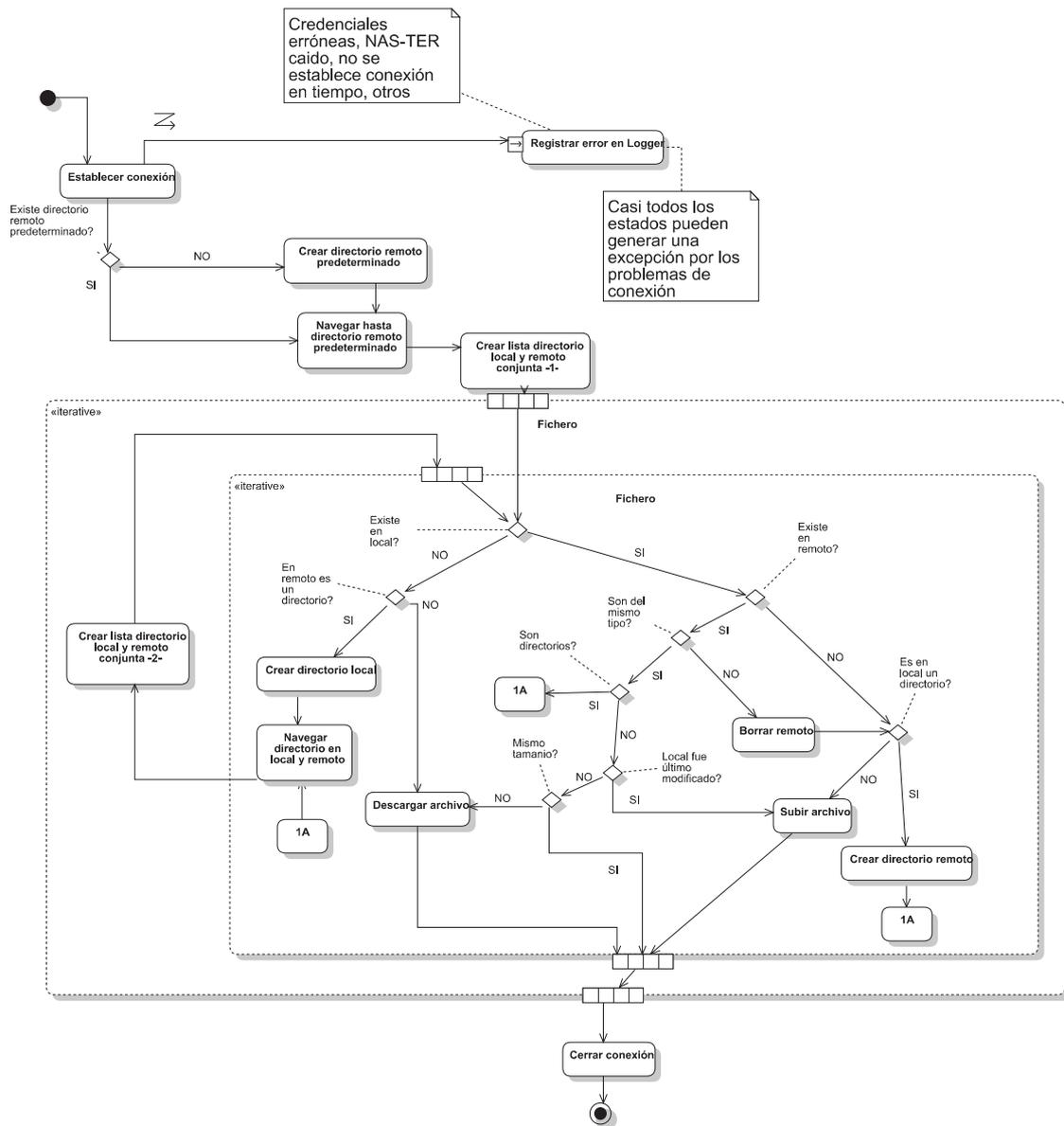


Figura 22 - Diagrama de actividad: sincronizar NAS-TER

En este diagrama vamos a ver la parte de la sincronización con NAS-TER, el sistema comenzará **conectando al servicio** al espacio del usuario y en caso de error **registraría el error**. Si la conexión fuera correcta primero comprobáramos la **existencia de un directorio defecto** donde trabajará la aplicación (de esta forma no desordenaríamos otros recursos usados por el usuario) creándolo si fuera necesario. A continuación, generaríamos una **lista del directorio de trabajo local y del directorio remoto (1)** e iterando sobre la lista en cada archivo actuaríamos con la siguiente prioridad:

- 1) Si NO existe en local y en remoto es un directorio creamos en local, navegamos en ambos y repetimos (1).
- 2) Si NO existe en local y en remoto NO es un directorio descargamos el archivo.

- 3) Si existe en local y NO en remoto, y es un directorio lo creamos en remoto, navegamos en ambos y repetimos (1).
- 4) Si existe en local y NO en remoto, y NO es un directorio lo subimos.
- 5) Si existe en ambos y NO son del mismo tipo borramos remoto y subimos local.
- 6) Si existe en ambos y son del mismo tipo, si son directorio navegamos en ambos y repetimos (1).
- 7) Si existe en ambos y son del mismo tipo, si local fue el último modificado subimos el archivo.
- 8) Si existe en ambos y son del mismo tipo, local NO fue último modificado y son de distinto tamaño descargamos el archivo.

### 3.4.8. Diagrama de actividad: sincronizar Moodle

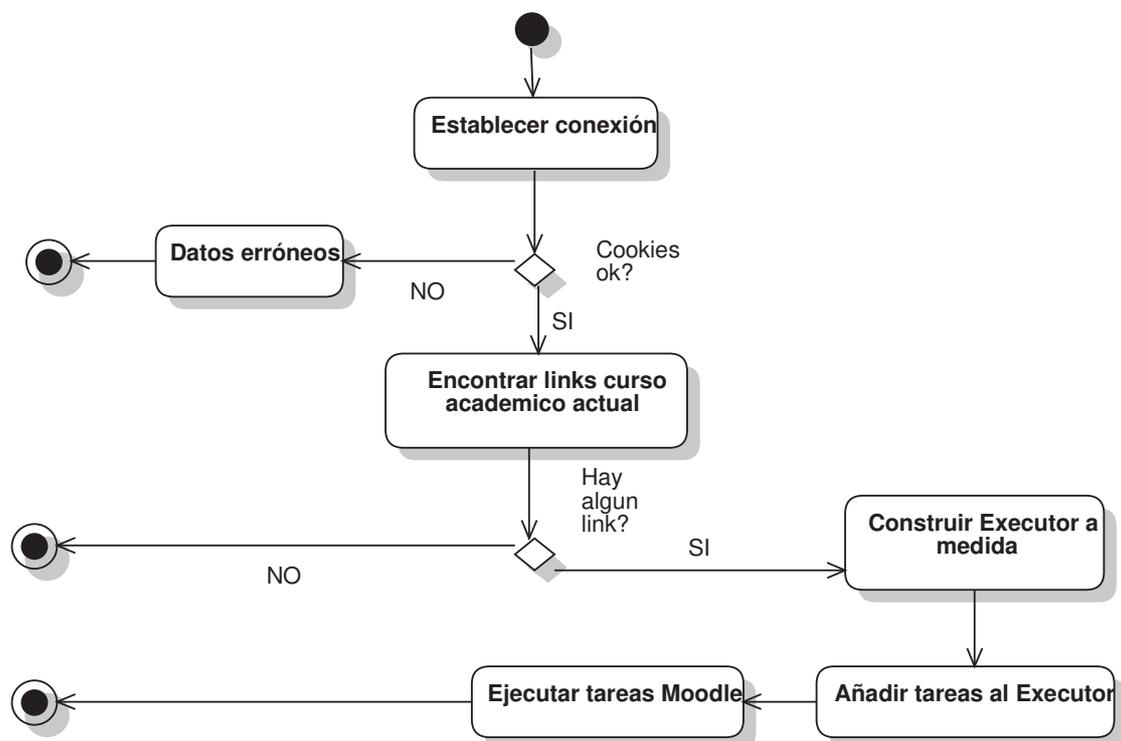


Figura 23 - Diagrama de actividad: sincronización Moodle 01

En este diagrama vamos a ver como la sincronización con Moodle inicia. Primero el sistema **establecerá conexión** con las credenciales almacenadas, y si la conexión fue correcta analizaremos la página actual en buscando los **links de los cursos académicos del año actual**; a continuación, **crearemos un Executor** por cada curso, y una **tarea asociada** al mismo y las ejecutaremos esperando la finalización en *multithread*.

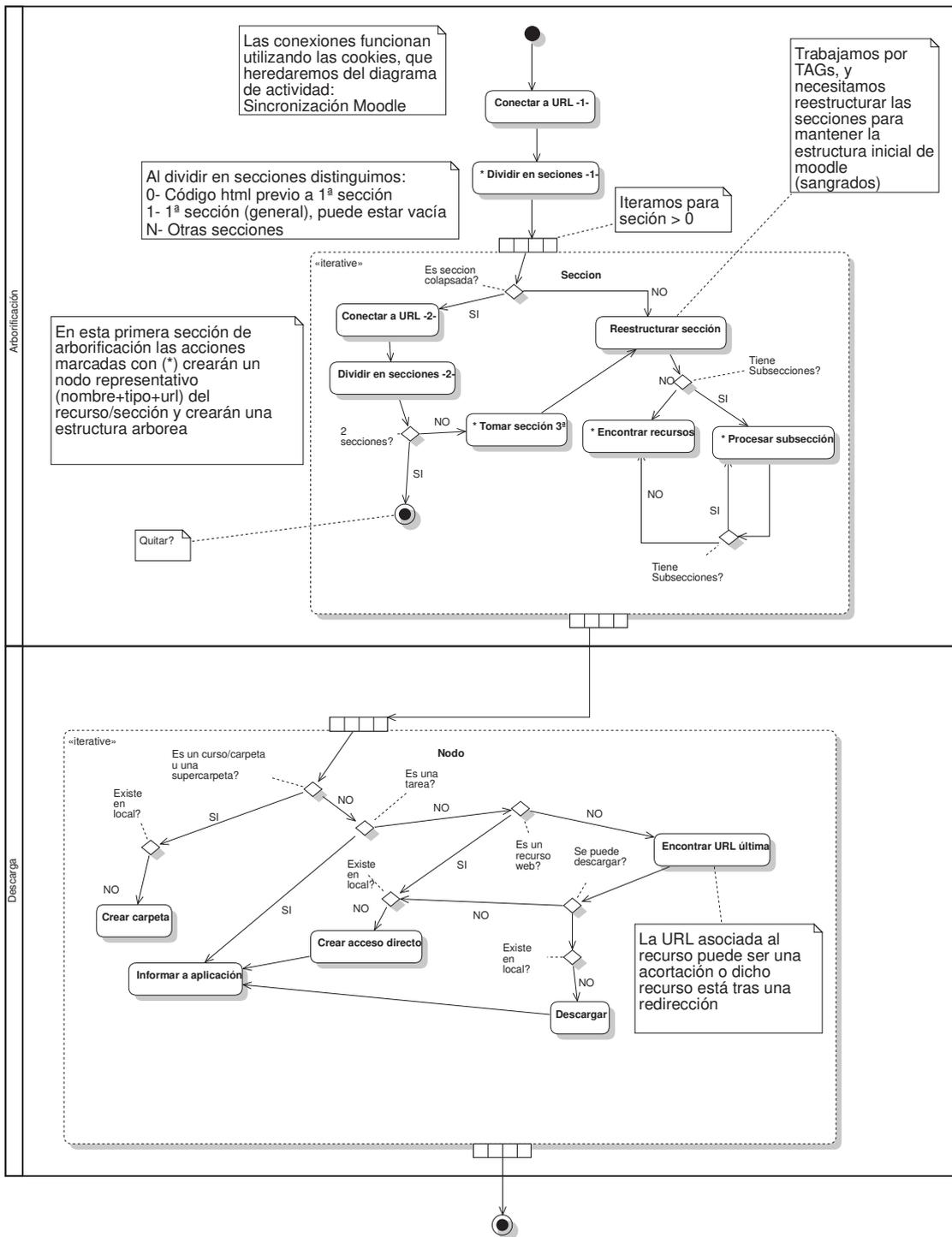


Figura 24 - Diagrama de actividad: sincronización Moodle 02

En este diagrama representamos el proceso de sincronización del sistema de cara a Moodle, podemos distinguir dos partes:

- Arborización: donde creamos una estructura arbórea para representar el contenido de un curso Moodle.
- Descarga: recorrido del árbol creado previamente y descarga si fuera pertinente.

Así pues, vamos a ver en que consiste la primera parte del proceso de wrapping que generará una estructura de nodos. Aclarar que los nodos son creados con cookies válidas que le permiten navegar por Moodle sin necesitar autenticación. El primer paso del nodo será **conectarse a la URL** asociada y **dividir el contenido html en secciones**, esto es posible debido a que Moodle está dividido en secciones de `<li id="section-` siendo estas:

- 0: cabecera de datos
- 1: sección general
- N: sección n.

Así pues, sobre cada sección >0 comprobaremos que no es una **sección colapsada**. Si lo fuera creamos un **nodo hijo (\*)** del nodo actual establecemos relación de hijo-padre conectamos a la URL asociada y tras dividir en secciones tomamos la sección 3 y la procesamos. En caso de que no sea una sección colapsada la procesaríamos.

Procesar una sección consiste en reestructurar su contenido basándonos en las etiquetas `<div>` & `<li>` para volver a montar la organización (sangrado) que el docente haya establecido para esta.

(1) Comprobaremos si sección **dispone de una subsección (\*)** y deberemos procesar esta (1') (\*); si la sección no dispone de una subsección **encontraremos los recursos** y los asociaremos al nodo padre (\*).

Las acciones marcadas con (\*) son acciones que cuyo resultado crean uno o varios nodos.

En la parte de la descarga, ya disponemos de un árbol de Nodos, cuya raíz representa un curso académico del año actual. Empezando por los nodos raíz recorreremos el árbol de forma que:

- 1) Si es el curso, sección o carpeta; crearemos una carpeta representativa si fuera necesario en el Sistema de Ficheros.
- 2) Si es una Tarea, informaremos a la aplicación de la información representativa de esta:
  - Curso asociado.
  - Nombre que el docente le da.
  - Fichero entrega, nombre del fichero entregado si lo tuviera.
  - Fecha límite de entrega.
  - Idioma en el que está la fecha.
  - Nota con la que corrigió el docente, si la tuviera.
  - Comentario/feedback del docente, si lo tuviera.
  - URL en la que se encuentra la tarea.
- 3) Si es un recurso web, crearíamos un acceso directo en el Sistema de Ficheros respetando la organización del curso.
- 4) Si fuera un archivo, encontraríamos el archivo asociado que pueda ser descargado y lo descargaríamos en el Sistema de Ficheros respetando la organización del curso.

Para los nodos de tipo no identificado, debido a mala tipificación del docente u otro problema, los trataríamos primero como un archivo 4). Si al tratarlo como un archivo no encontramos un archivo descargable, mala tipificación del docente, lo trataremos como un recurso web 3).

### 3.5. Diseño de la interfaz

Para el diseño del proyecto se ideó originalmente un prototipo de bajo nivel usando Balsamic Mockups [2], a continuación, vamos a ver la versión de bajo nivel y el resultado final con alguna observación.

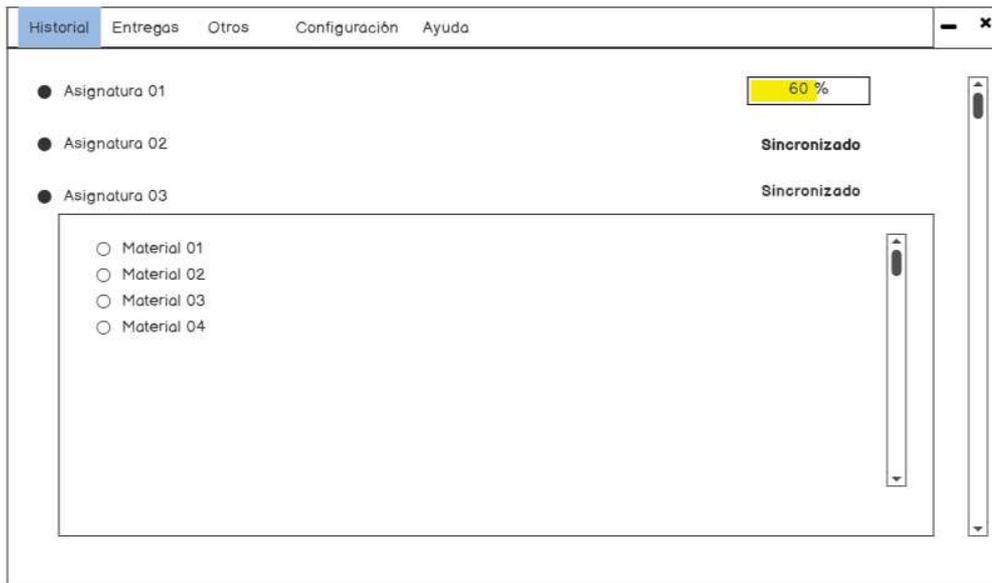


Figura 25 - Prototipo historial

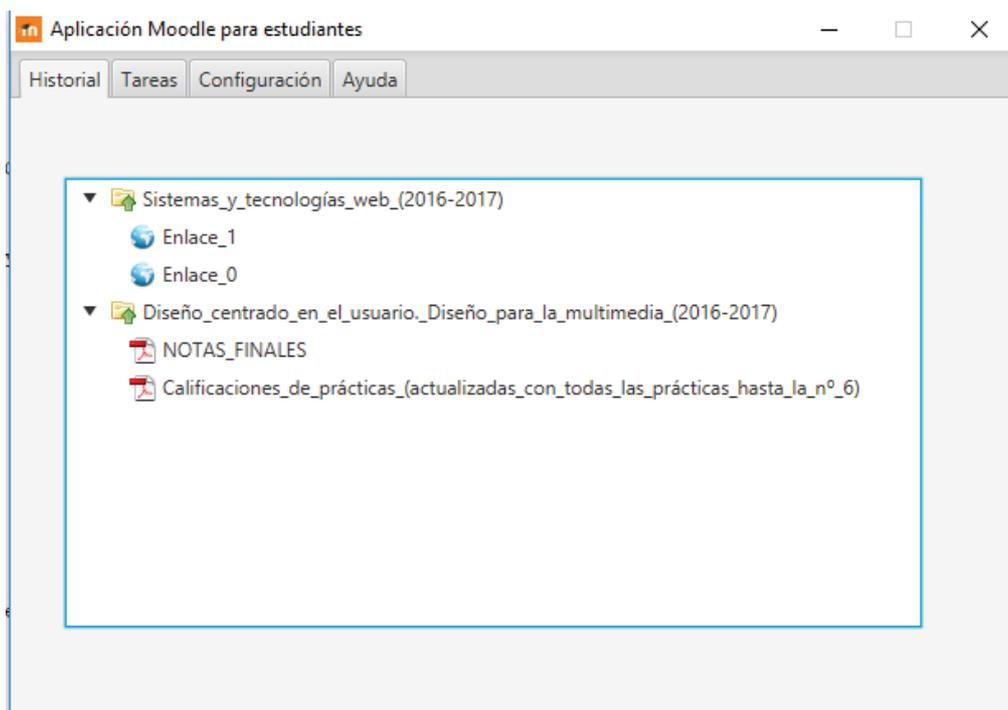


Figura 26 - Aplicación historial

La idea inicial era basarlo en un sistema parecido a Dropbox que indicara el estado y porcentaje de la sincronización. Sin embargo, más adelante se descubrió que los archivos tienen un tamaño muy reducido y además sabríamos de él instantes antes de su descarga.

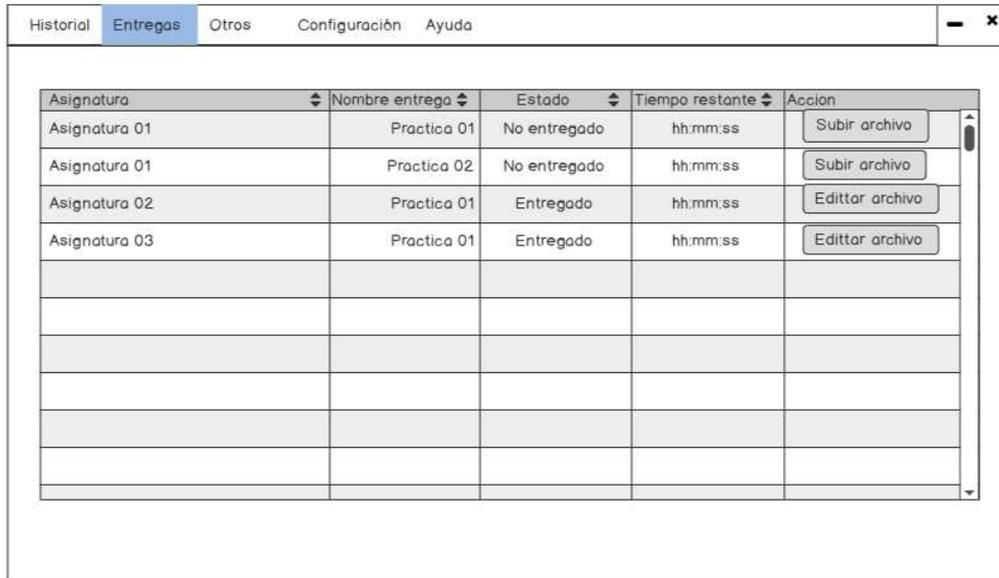


Figura 27 - Prototipo entrega

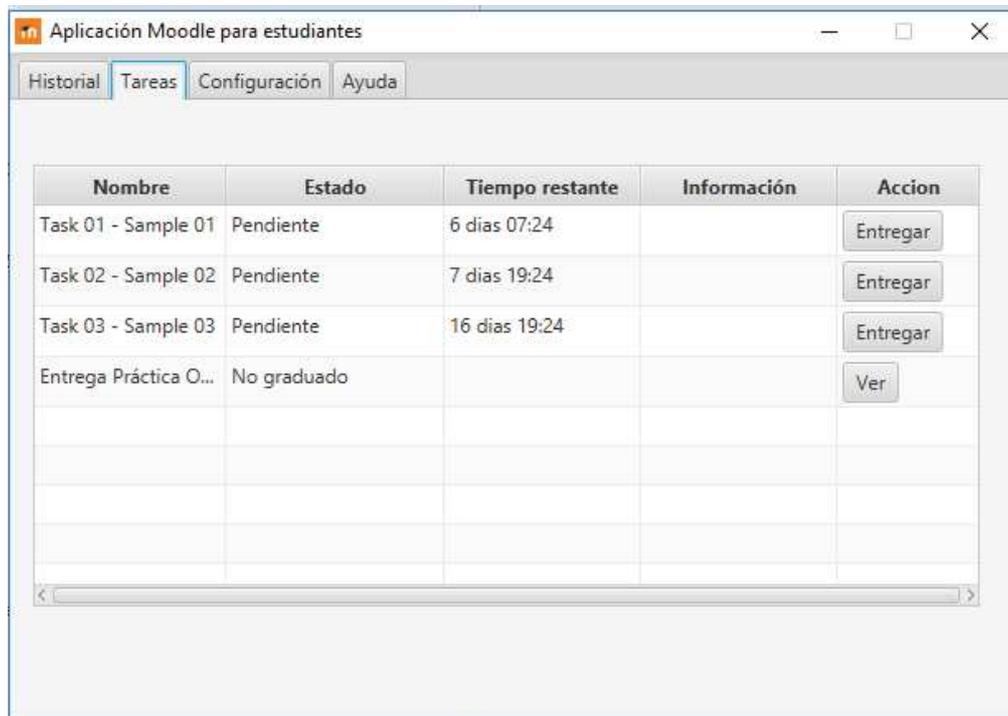


Figura 28 - Aplicación entrega

En las tareas se decidió juntar el nombre de la entrega y el curso en un solo campo; además se añadió un campo información en el que mostramos información relevante (Fichero entregado, nota, *feedback*)



Figura 29 - Prototipo configuración

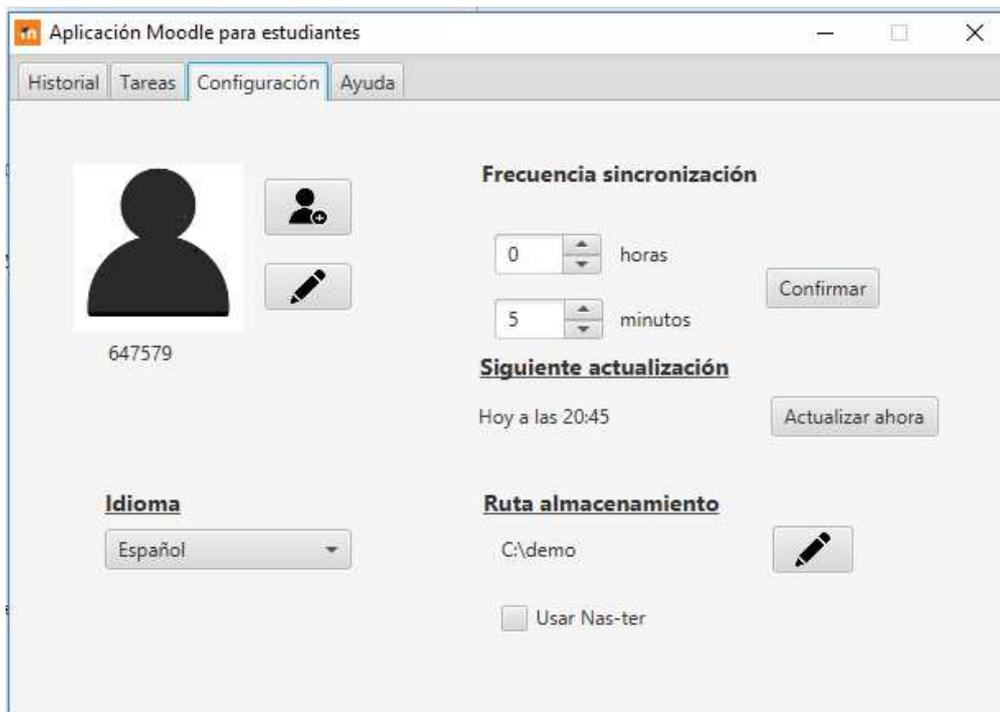


Figura 30 - Aplicación configuración

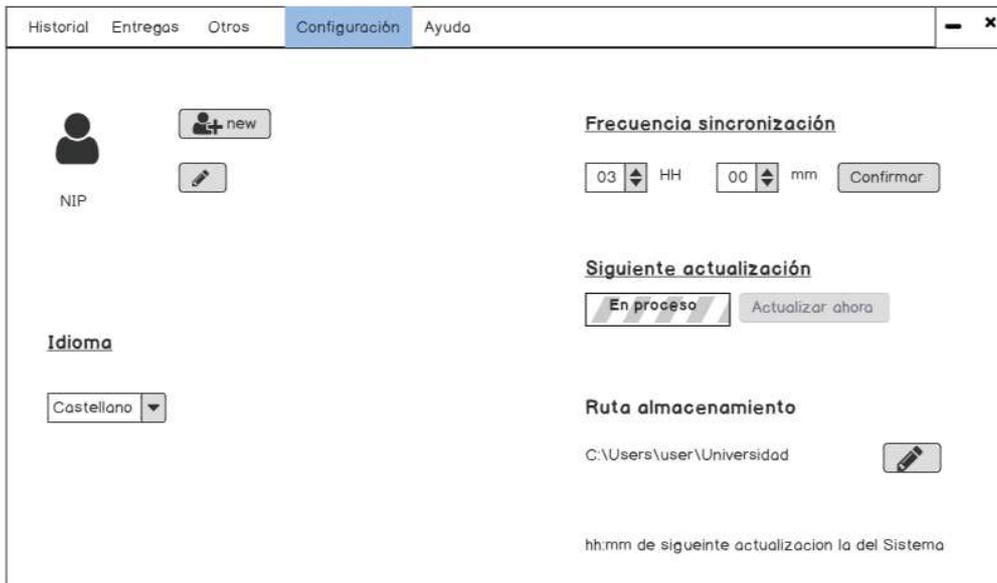


Figura 31 - Prototipo configuración 2

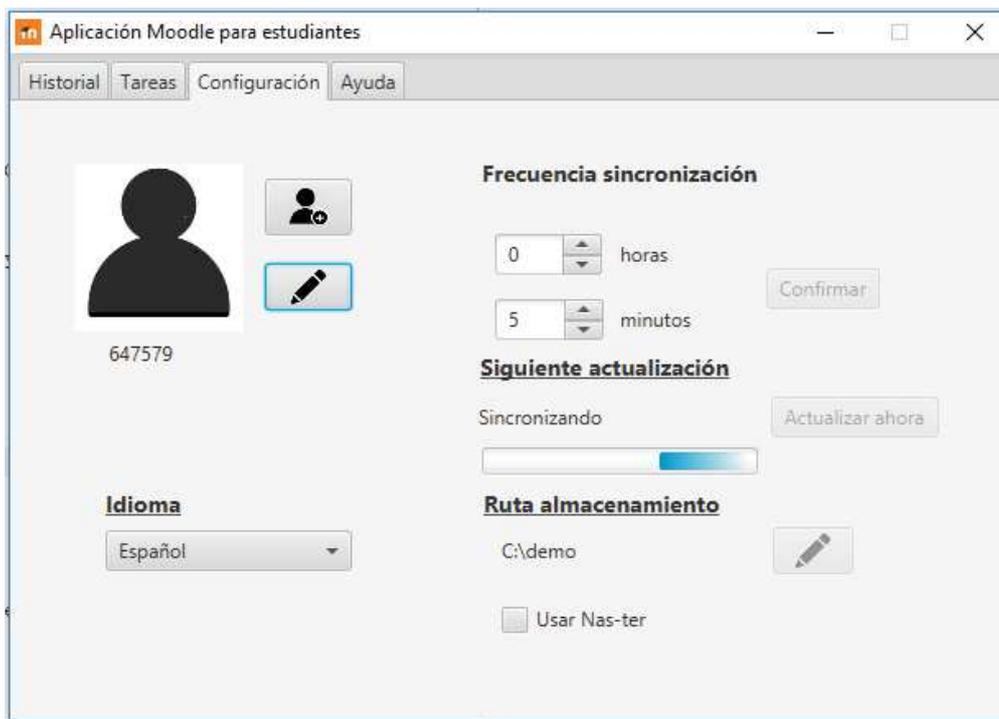


Figura 32 - Aplicación configuración 2

Las ventanas de configuración son casi idénticas. En un principio se quería marcar el estado de sincronización con una animación, pero debido a la inexperiencia se cambió a texto indicativo. Además, se da la opción de activar o desactivar el uso de NAS-TER.

Formulario de creación de usuario con los siguientes elementos:

- Título: **New User**
- Campo: **NIP** (caja de texto)
- Campo: **Contraseña Moodle** (caja de texto)
- Campo: **Contraseña NAS-TER** (caja de texto)
- Botones: **Aceptar** y **Cancelar**
- Campo: **Ruta de almacenamiento** (caja de texto con ícono de carpeta)

Figura 33 - Prototipo nuevo usuario

Captura de pantalla de la aplicación 'Nuevo usuario' con los siguientes elementos:

- Título: **Nuevo usuario**
- Campo: **NIP:** usuario
- Campo: **Moodle:** contraseña
- Campo: **NAS-TER:** contraseña
- Checkbox:  **Usar Nas-ter**
- Campo: **Descargando en:** directorio
- Botones: **Aceptar** y **Cancelar**

Figura 34 - Aplicación nuevo usuario

El uso de NAS-TER se cambió a opcional por el usuario, de modo que el formulario así lo refleja.

User 123456

<b>Contraseña Moodle</b> *****	<b>Contraseña NAS-TER</b> *****
<input type="button" value="Aceptar"/>	<input type="button" value="Cancelar"/>

Figura 35 - Prototipo editar usuario

Editar usuario

NIP:

Moodle:

NAS-TER:   Usar Nas-ter

Figura 36 - Aplicación editar usuario

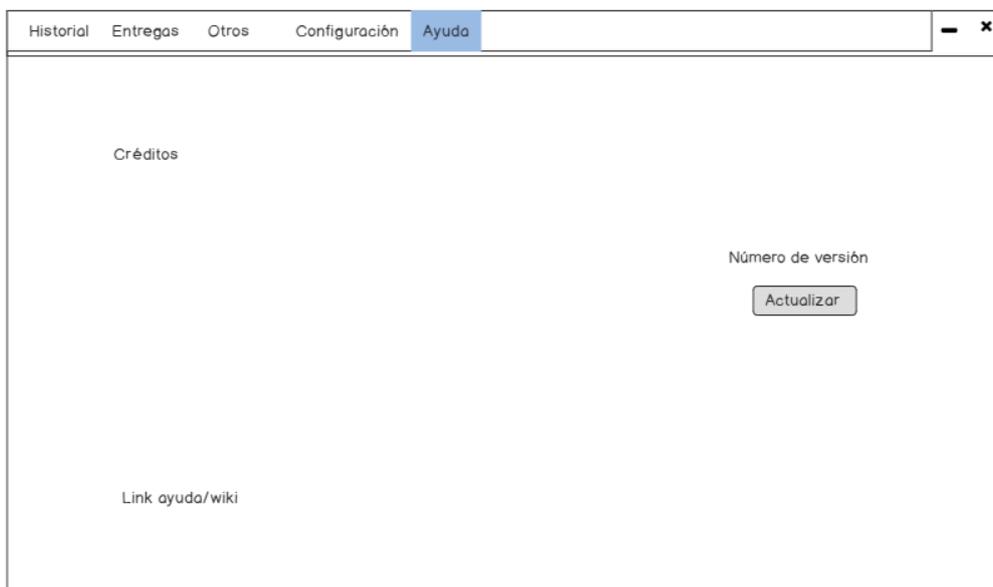


Figura 37 - Prototipo ayuda

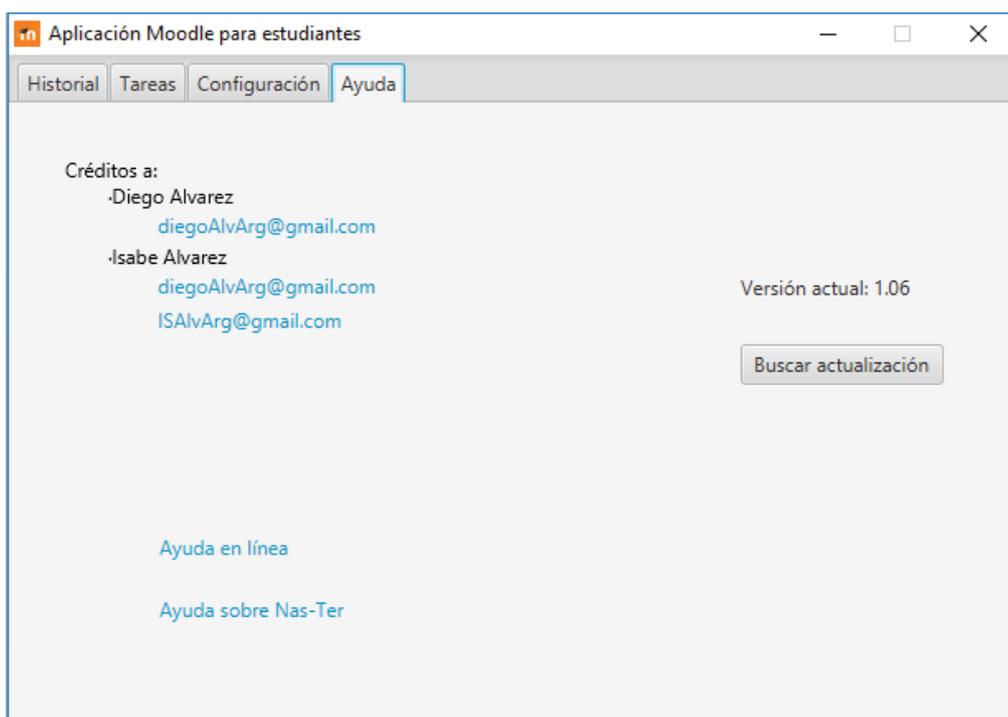


Figura 38 - Aplicación ayuda

## 4. PRUEBAS Y RESULTADOS

Todas las funcionalidades han sido probadas antes de su integración en el sistema.

Además, disponemos de 4 beta-tester de la aplicación, cuyas evaluaciones son:

- 542088 – Sergio Alloza García, ver Anexo III página 54
- 546784 – M<sup>a</sup> Luisa Rubio, ver Anexo III página 55.
- 698394 – Guillermo Morales, ver Anexo III página 56.
- 680723 – Laura Herrera, ver Anexo III página 57.

Las principales quejas o sugerencias de estas personas son:

- Mejorar el acabado visual.
- Disponer de una sincronización selectiva.
- Mejorar o facilitar más el proceso de creación de un perfil, así como disponer de imagen de perfil.

### 4.1. Sincronización NAS-TER

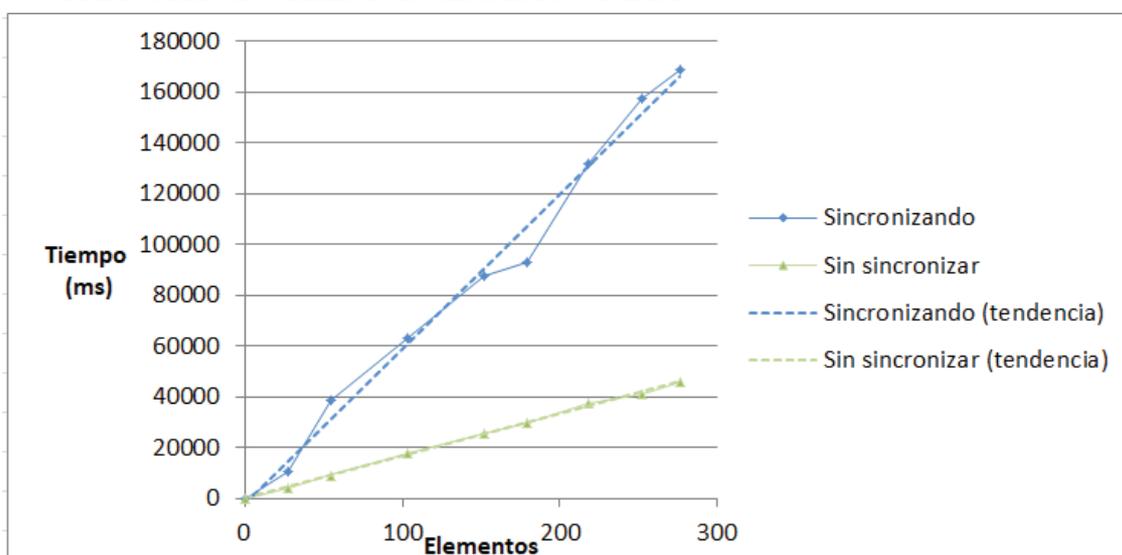


Figura 39 - tiempos NAS-TER

Elementos	Tiempo ms. (sincronizando)	Tiempo ms. (sin sincronizar)
0	0	0
28	10.262	4.110
55	38.825	9.051
103	63.094	17.496
152	87.698	25.224
179	92.945	29.610
218	131.732	37.147
252	157.431	41.225
276	168.604	45.924

En la figura y tabla anterior podemos ver la toma de tiempos para la parte de la sincronización con NAS-TER, siendo esta una sincronización secuencial en una franja de hasta 276 elementos. Se entiende por elemento la combinación de carpetas y archivos de las que el sistema guarda constancia en local. Podemos ver dos gráficas, las cuales son:

- Sincronizando: el elemento debe ser sincronizado (descargando en local o subiendo a remoto).
- Sin sincronizar: el elemento no debe ser sincronizado.

De estas gráficas generaremos las líneas de tendencia (lineal), y observando estas podemos apreciar una franja entre ambas, de esta franja podemos ver una fluctuación del proceso de sincronización con NAS-TER; el tiempo de este proceso dependerá del número de archivos y de si es necesario sincronizarlos o no.

## 4.2. Sincronización Moodle

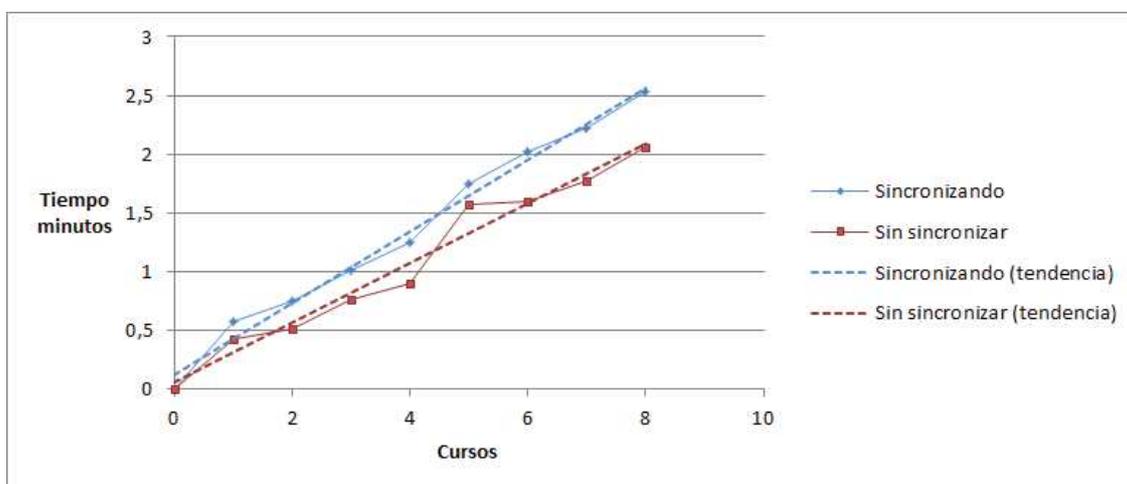


Figura 40 - Tiempos Moodle

Cursos	Tiempo min. (sincronizando)	Tiempo min. (sin sincronizar)
0	0	0
1	0'564	0'414
2	0'74	0'508
3	1'012	0'759
4	1'241	0'899
5	1'742	1'571
6	2'022	1'594
7	2'219	1'772
8	2'534	2'054

El estudio de tiempos para la sincronización Moodle presenta un par de apuntes a realizar: la sincronización Moodle se realiza en paralelo y esto lleva una disonancia, pues no todos los cursos son iguales en estructura o tardan el mismo tiempo. Siendo de la Figura 40 y tabla anterior:

- Sincronizando: el curso contiene elementos que deben ser sincronizados en local.

- Sin sincronizar: el curso no contiene elementos que deban ser sincronizados.

Podemos ver que la sincronización de dos cursos tarda entre 30.48'' y 44.4''. Sin embargo, la sincronización de ocho cursos que debería ser cuatro veces más tarda entre 2' 3'' y 2'32''; en vez de los 2' 1'' y 2' 57'' de la estimación.

Podemos decir que un curso, con una estimación de 30 elementos (entendiendo por elementos la combinación de carpetas, archivos y entregas), tardaría una franja entre 24.84'' y 33.84'' dependiendo de cada elemento deba ser sincronizado o no.

Así, pues se presenta en complementación la siguiente figura y tabla.

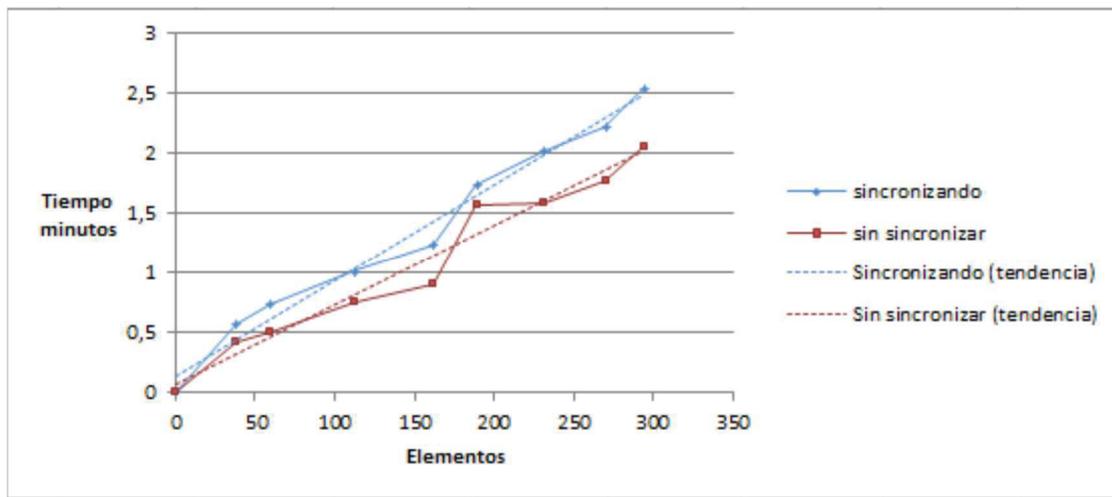


Figura 41 - Tiempos Moodle 2

Elementos	Tiempo min. (sincronizando)	Tiempo min. (sin sincronizar)
0	0	0
38	0'564	0'414
59	0'74	0'508
112	1'012	0'759
162	1'241	0'899
189	1'742	1'571
231	2'022	1'594
270	2'219	1'772
294	2'534	2'054

En la figura y tabla anteriores podemos ver la toma de tiempos para la sincronización Moodle teniendo en cuenta el número de elementos, entendiendo por elementos la combinación de carpetas, archivos y entregas. Podemos ver dos gráficas, las cuales son:

- Sincronizando: el elemento debe ser sincronizado en local.
- Sin sincronizar: el elemento no debe ser sincronizado en local.

Con una muestra de 294 elementos tarda un tiempo entre 2'03'' y 2'32'' a lo que saldría una equivalencia de un elemento con un tiempo entre 0.414'' y 0.517''. Sin embargo, podemos ver

el primer curso con 38 elementos (cerca de la estimación de la sección anterior a 30 elementos) tarda entre 24.84" y 33.84" que difiere de la previsión de 15.73" y 19.64".

### 4.3. Discusión de resultados

Como estudio de los anteriores apartados procedemos a la explicación en conjunto de esta. Recordando que el número de elementos en las mismas no son los mismos debido al cambio de elementos y que, además, los tiempos son para cursos ya finalizados; es pues que la conclusión debe ser tratada como una aproximación. Suponiendo que el alumno cuenta con 5 asignaturas (media de asignaturas en un cuatrimestre)

En el caso de querer usar NAS-TER, esta sincronización tardará un tiempo X, donde será un valor de tiempo entre 0' y ~1'32"

En la sincronización Moodle, esta tardará un tiempo Y, donde será un valor de tiempo entre ~1'34" y ~1'44"

## 5. SISTEMA DE INSTALACIÓN Y ACTUALIZACIÓN

Para mejorar la mantenibilidad de la aplicación se ha incorporado un sistema de actualización basado en el proyecto [JavaFXApplicationAutoUpdater](#) [9] del usuario [goxr3plus](#) [29] basándonos y siguiendo la licencia de uso [8].

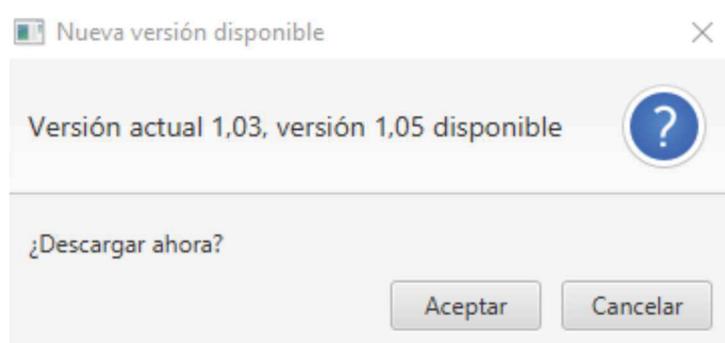


Figura 42 - Actualización disponible

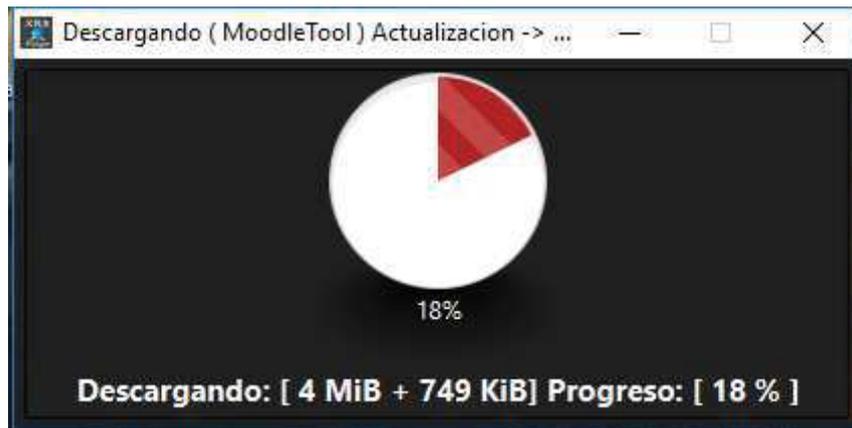


Figura 43 - Actualización en proceso

Además, para facilitar la distribución se ha generado un instalador a partir del proyecto compilado (.jar), gracias a la herramienta Install4j, que nos permite distribuir una versión Java con la que ejecutar el sistema correctamente y ejecutar la aplicación con permisos de administrador. Se recomienda utilizar el manual de uso [13]

## 6. CONCLUSIONES

Como resultado de este trabajo, cuyo objetivo era generar un sistema que minimizara el tiempo entre que un profesor generaba un material docente y el alumno era consciente de ello; se ha generado una aplicación que mantiene el equipo en el que está siendo ejecutado sincronizado con la plataforma Moodle con una frecuencia mínima de 5 min que tarda entre 0~3 min en sincronizar.

Se ha realizado de forma que es de fácil instalación y actualización, lo que permite a los usuarios sin conocimiento técnico su uso. Además, permite un sencillo mantenimiento.

Queda pendiente para un futuro trabajar en los siguientes aspectos:

- Mejorar el sistema de actualización. La herramienta Install4j permite adherir un sistema de actualización, que podría sustituir al actual de forma que no hagamos uso parcial o reconocimiento de terceros.
- Mejorar la velocidad de identificación de recursos Moodle utilizando la API Moodle en vez de *scraping*. O tal vez si comprometemos al buen uso de los docentes almacenar "imagen" de los cursos para identificar rápidamente los últimos recursos.
- Añadir otras funcionalidades Moodle como los *quizz*, *wikis*, foros, etc. Debido a que el curso sobre el que se utilizó para comprobar el funcionamiento de la aplicación no había ninguna de estas funcionalidades. Sin embargo, la aplicación está definida de forma modular por lo que sería fácil su implementación.
- Facilitar un mejor sistema de ayuda y manuales accesibles desde la ventana de ayudas. Actualmente no hay una web que contenga una guía debido a que no se dispone de ella, y que los elementos de la aplicación cuentan con *tooltips* por lo que no era una prioridad generar un manual de usuario.

- Cambiar los estilos (CSS) de forma que se asemeje más a la imagen corporativa de Moodle, destacar las notificaciones de SystemTray las cuales no parecen tener una fácil manipulación de estilos.
- Mejorar la accesibilidad de la aplicación, como es el tema de los idiomas poner imágenes de banderas, aclarar el proceso de creación de perfil de usuario.

## 7. REFERENCIAS

---

- [1] Balsamiq Cloud. *Balsamiq.cloud* [en línea] 2018. Disponible en: <https://balsamiq.cloud/>
- [2] Balsamiq. Rapid, effective and fun wireframing software. | Balsamiq. *Balsamiq.com* [en línea] 2018. Disponible en: <https://balsamiq.com/>
- [3] BOWLER, M., GUILLEMOT, M. and ASHOUR, A. 2018. HtmlUnit – Welcome to HtmlUnit. *Htmlunit.sourceforge.net* [en línea] 2018. Disponible en: <http://htmlunit.sourceforge.net/>
- [4] Desarrollo ágil de software. *Es.wikipedia.org* [en línea] 2018. Disponible en: [https://es.wikipedia.org/wiki/Desarrollo %C3%A1gil de software](https://es.wikipedia.org/wiki/Desarrollo_%C3%A1gil_de_software)
- [5] ej-technologies - Java APM, Java Profiler, Java Installer Builder. *Ej-technologies.com* [en línea] 2018. Disponible en: <https://www.ej-technologies.com/download/install4j/files>
- [6] Espacios Web colaborativos (GAFE) - Inicio | Servicio de Informática y Comunicaciones. *Sicuz.unizar.es* [en línea] 2018. Disponible en: <https://sicuz.unizar.es/correo-y-colaboracion/espacios-web-colaborativos-inicio/espacios-web-colaborativos-gafe-inicio>
- [7] GIUSTINA, F. 2018. JTidy - JTidy. *Jtidy.sourceforge.net* [en línea] 2018. Disponible en: <http://jtidy.sourceforge.net/>
- [8] goxr3plus/JavaFXApplicationAutoUpdater. *GitHub* [en línea] 2018. Disponible en: <https://github.com/goxr3plus/JavaFXApplicationAutoUpdater/blob/master/LICENSE>
- [9] goxr3plus/JavaFXApplicationAutoUpdater. *GitHub* [en línea] 2018. Disponible en: <https://github.com/goxr3plus/JavaFXApplicationAutoUpdater>
- [10] HEDLEY, J. 2018. Try jsoup online: Java HTML parser and CSS debugger. *Try.jsoup.org* [en línea] 2018. Disponible en: <https://try.jsoup.org/>
- [11] HEDLEY, J. 2018. jsoup Java HTML Parser, with best of DOM, CSS, and jquery. *Jsoup.org* [en línea] 2018. Disponible en: <https://jsoup.org/>
- [12] HEDLEY, J. 2018. Cookbook: jsoup Java HTML parser. *Jsoup.org* [en línea] 2018. Disponible en: <https://jsoup.org/cookbook/>
- [13] install4j Help. *Ej-technologies.com* [en línea] 2018. Disponible en: <https://www.ej-technologies.com/resources/install4j/help/doc/#install4j.welcome>
- [14] Installing JavaFX | JavaFX 2.0 Tutorials and Documentation. *Docs.oracle.com* [en línea] 2018. Disponible en: <https://docs.oracle.com/javafx/2/installation/jfxpub-installation.htm>
- [15] Java Archive Downloads - JavaFX 2. *Oracle.com* [en línea] 2018. Disponible en: <https://www.oracle.com/technetwork/java/javafx2-archive-download-1939373.html>
- [16] Java™ SE Development Kit 8, Update 91 Release Notes. *Oracle.com* [en línea] 2018. Disponible en: <https://www.oracle.com/technetwork/java/javase/8u91-relnotes-2949462.html>
- [17] Moodle - Apps on Google Play. *Play.google.com* [en línea] 2018. Disponible en: <https://play.google.com/store/apps/details?id=com.moodle.moodlemobile&hl=en>
- [18] Moodle - Open-source learning platform | Moodle.org. *Moodle.org* [en línea] 2018. Disponible en: <https://moodle.org/>

- [19] Moodle downloads: Moodle Desktop. *Download.moodle.org* [en línea] 2018. Disponible en: <https://download.moodle.org/desktop/>
- [20] Moodle plugins directory. *Moodle.org* [en línea] 2018. Disponible en: <https://moodle.org/plugins/?q=type:repository%20moodle-version:3.3>
- [21] Moodle plugins directory: Office 365 Repository. *Moodle.org* [en línea] 2018. Disponible en: [https://moodle.org/plugins/repository\\_office365](https://moodle.org/plugins/repository_office365)
- [22] Moodle plugins directory: Evernote. *Moodle.org* [en línea] 2018. Disponible en: [https://moodle.org/plugins/repository\\_evernote](https://moodle.org/plugins/repository_evernote)
- [23] Moodle plugins directory: ownCloud. *Moodle.org* [en línea] 2018. Disponible en: [https://moodle.org/plugins/repository\\_owncloud](https://moodle.org/plugins/repository_owncloud)
- [24] Multi-Platform Java Installer Builder - install4j. *Ej-technologies.com* [en línea] 2018. Disponible en: <https://www.ej-technologies.com/products/install4j/overview.html>
- [25] NetBeans IDE Download. *Netbeans.org* [en línea] 2018. Disponible en: <https://netbeans.org/downloads/8.1/>
- [26] Recursos | add.unizar.es. *Add.unizar.es* [en línea] 2018. Disponible en: <http://add.unizar.es/add/campusvirtual/recursos-para-moodle-2x>
- [27] Scene Builder - Gluon. *Gluon* [en línea] 2018. Disponible en: <https://gluonhq.com/products/scene-builder/>
- [28] Scrum (desarrollo de software). *Es.wikipedia.org* [en línea] 2018. Disponible en: [https://es.wikipedia.org/wiki/Scrum\\_\(desarrollo\\_de\\_software\)](https://es.wikipedia.org/wiki/Scrum_(desarrollo_de_software))
- [29] STUDIO, G. 2018. · GitHub. *GitHub* [en línea] 2018. Disponible en: <https://github.com/goxr3plus>
- [30] Welcome to NetBeans. *Netbeans.org* [en línea] 2018. Disponible en: <https://netbeans.org/>
- [31] Moodle. *App Store* [en línea] 2018. Disponible en: <https://itunes.apple.com/au/app/moodle-mobile/id633359593>

## 8. ANEXOS

### 8.1. Diagramas UML

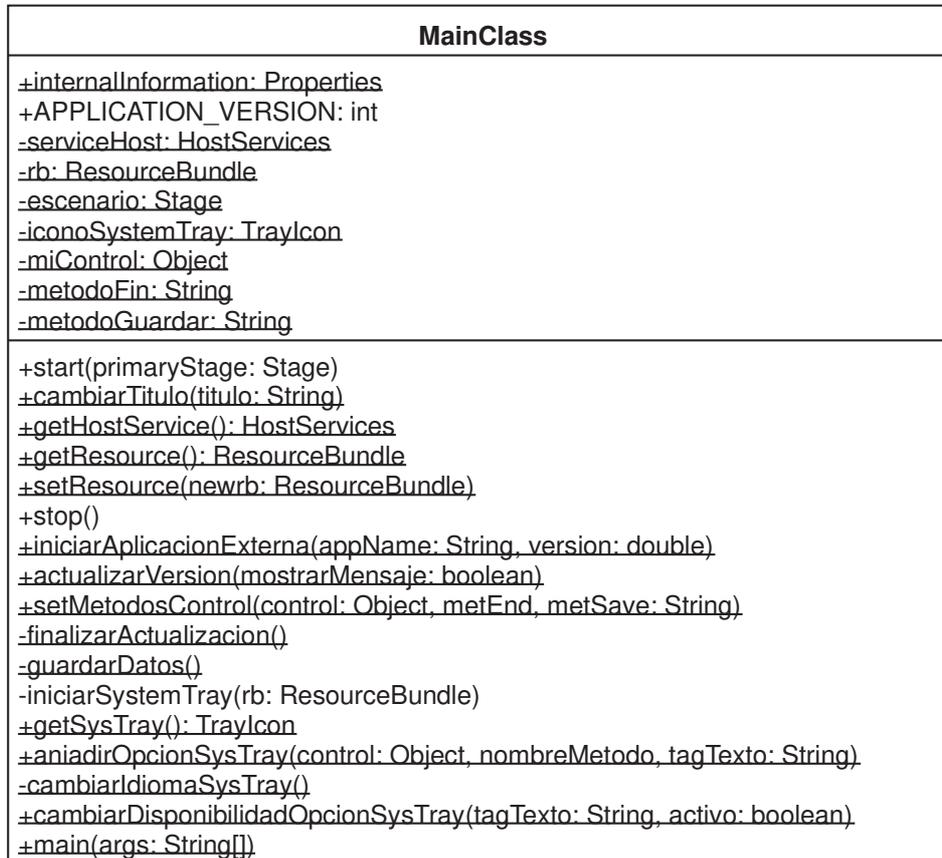


Figura 44 - Diagrama de clase: aplicacion 01

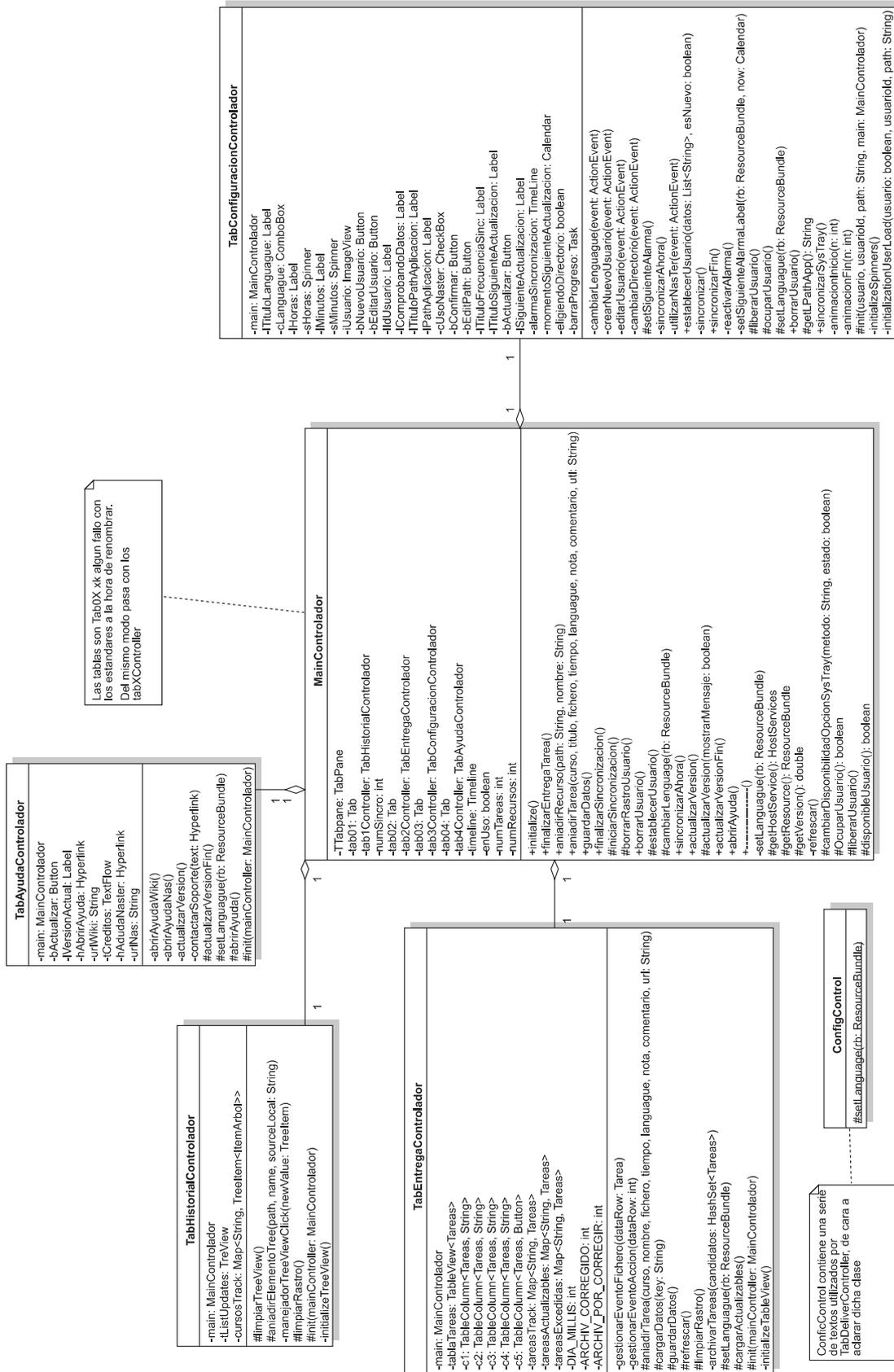


Figura 45 - Diagrama de clase: aplicacion 02

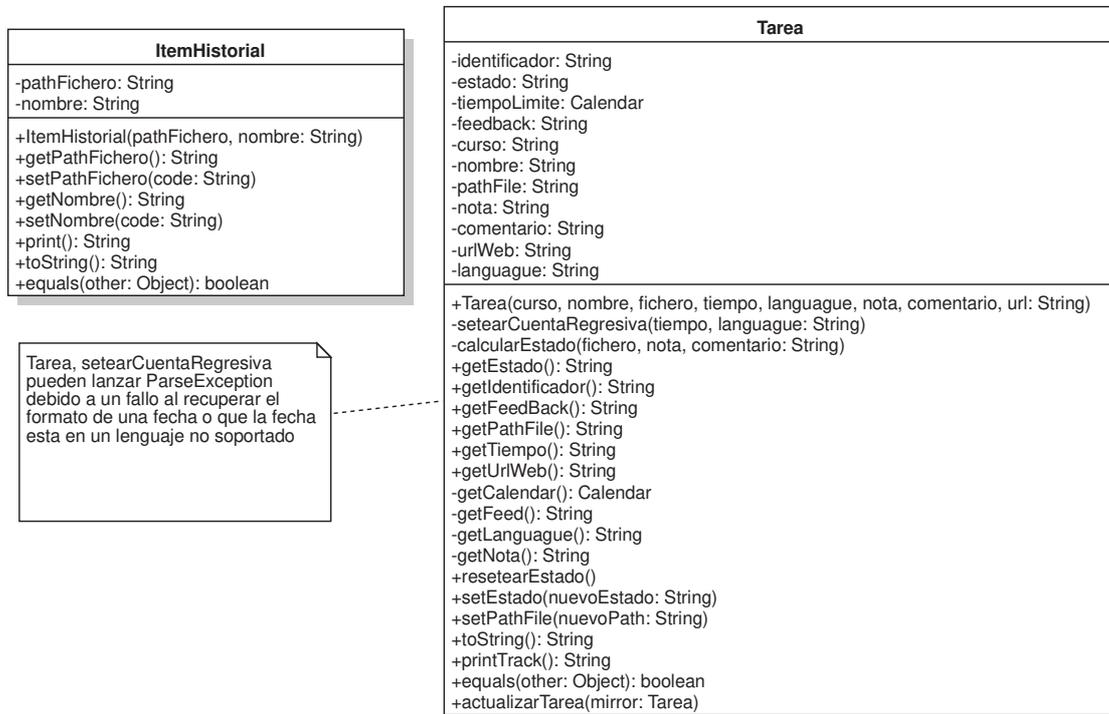


Figura 46 - Diagrama de clase: aplicacion 03

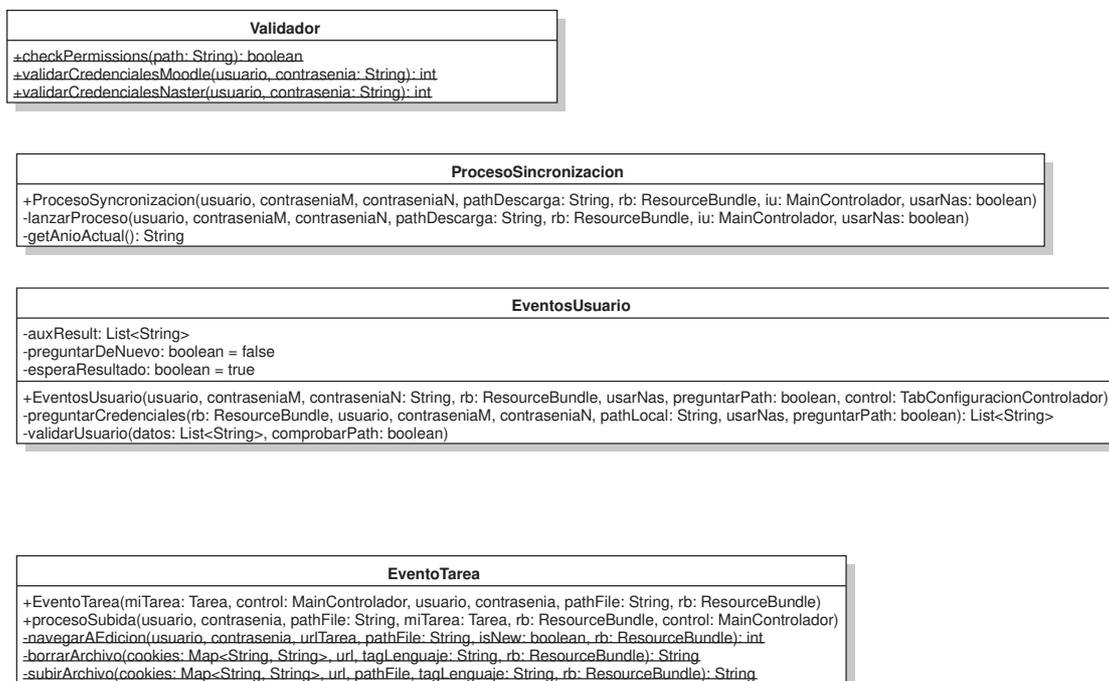


Figura 47 - Diagrama de clase: aplicacion 04



Figura 48 - Diagrama de clase: actualizador 01

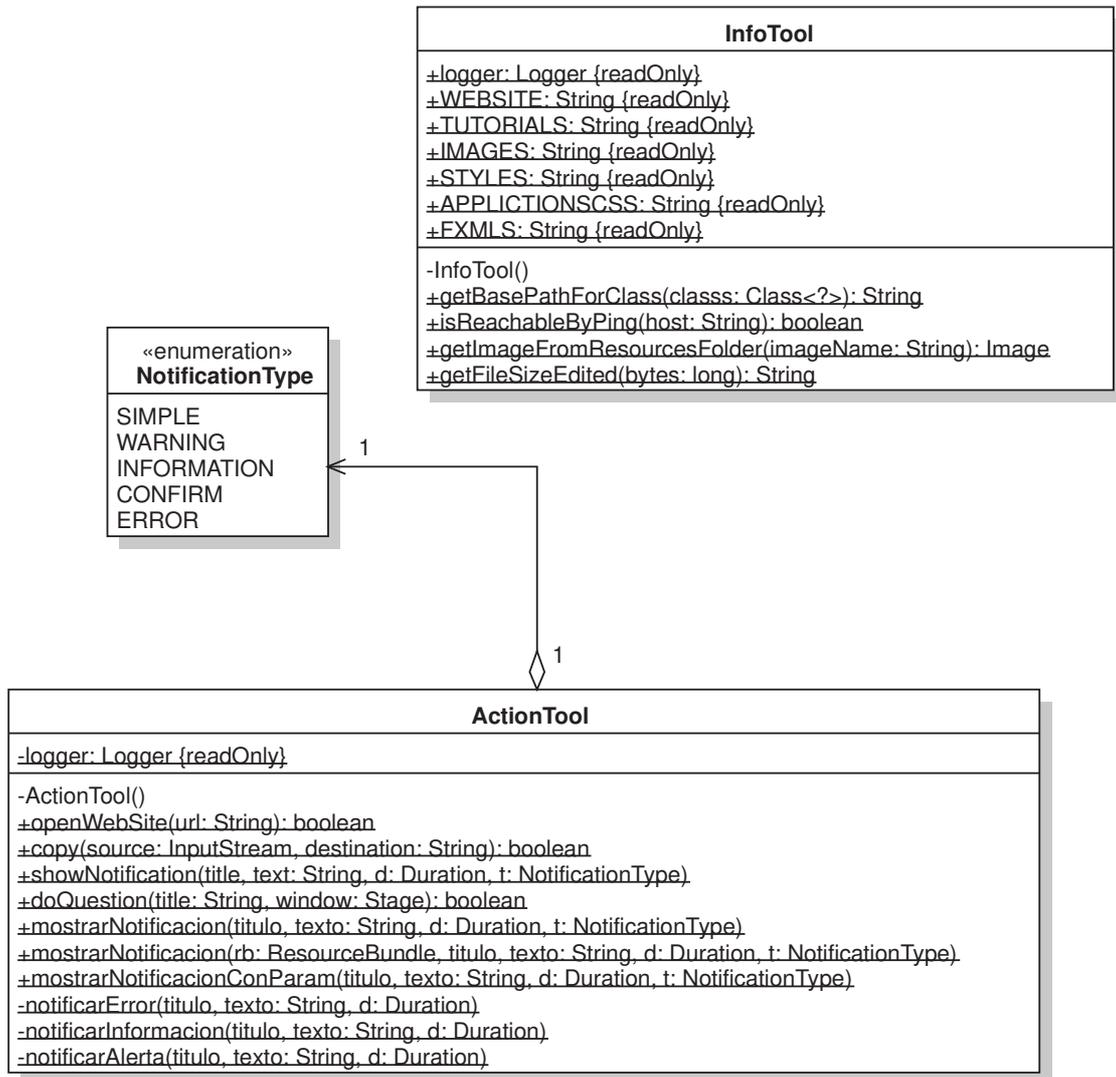


Figura 49 - Diagrama de clase: actualizador 02

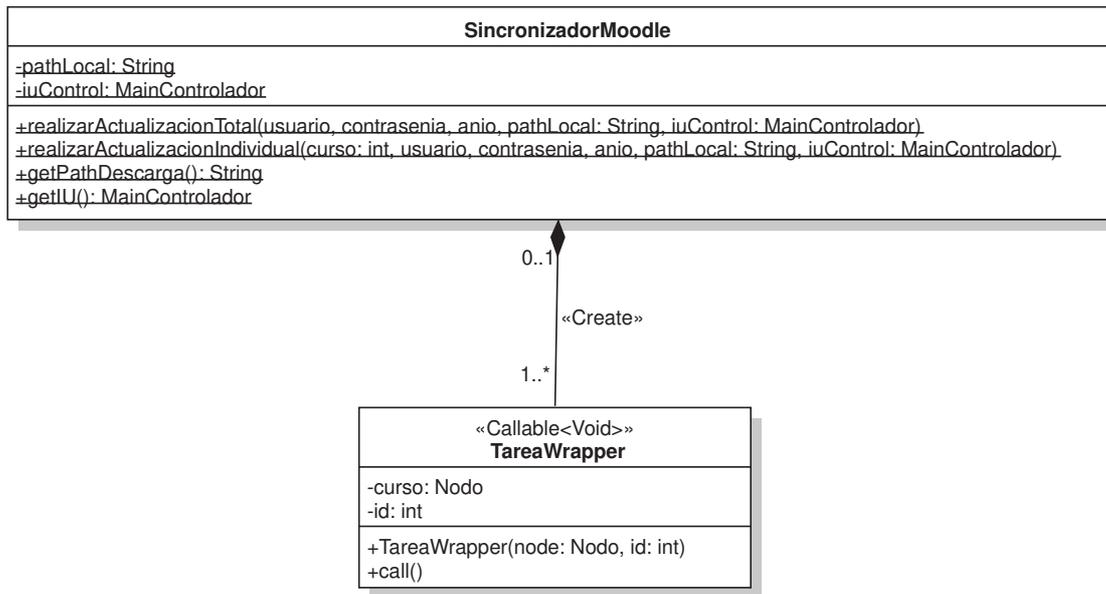


Figura 50 - Diagrama de clase: sincronizacion 01

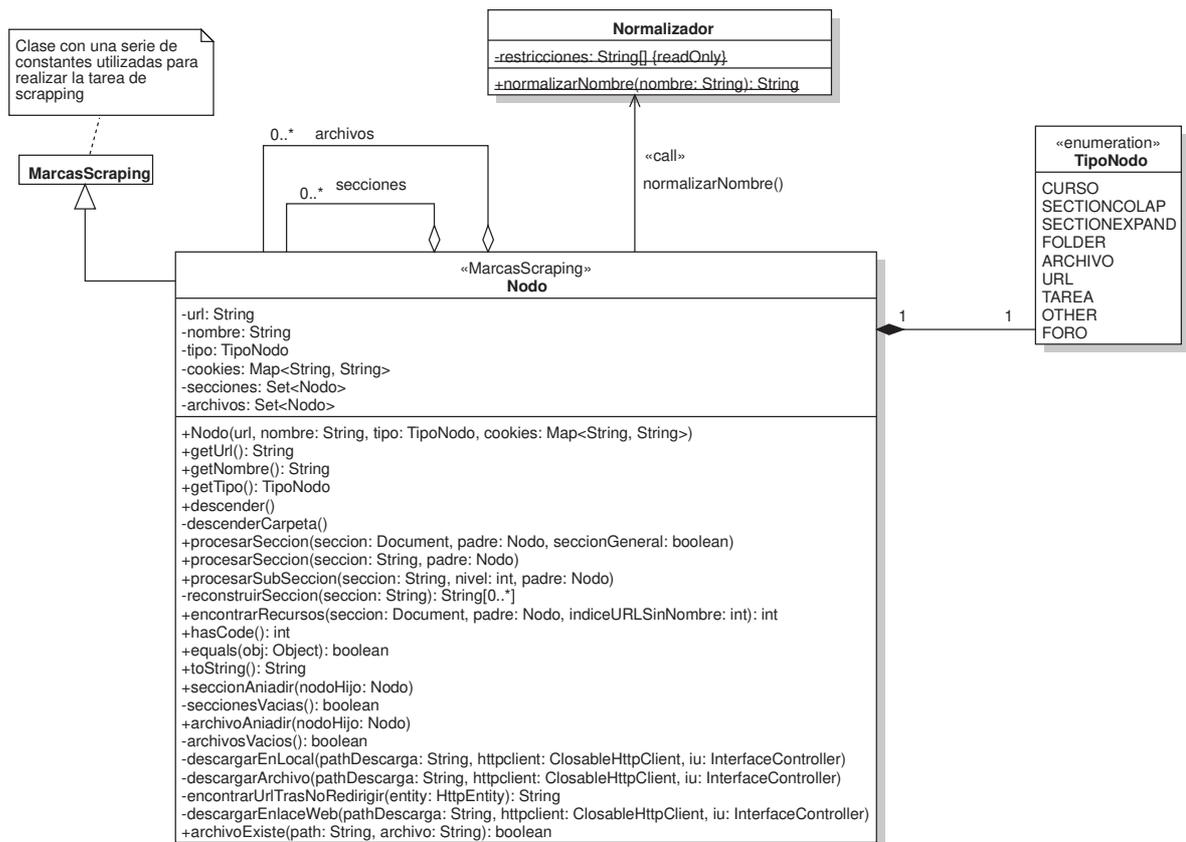


Figura 51 - Diagrama de clase: sincronizacion 02

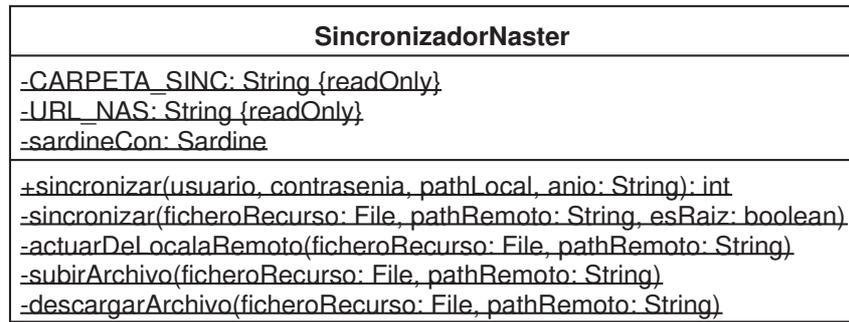


Figura 52 - Diagrama de clase: sincronizacion 03

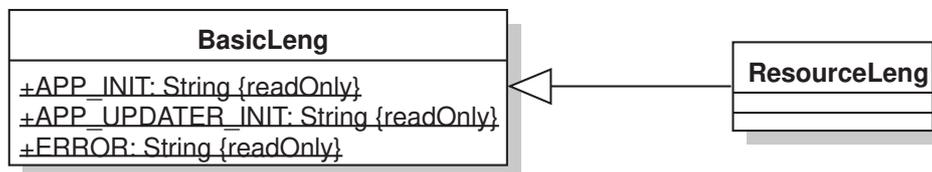


Figura 53 - Diagram de clase: tools 01

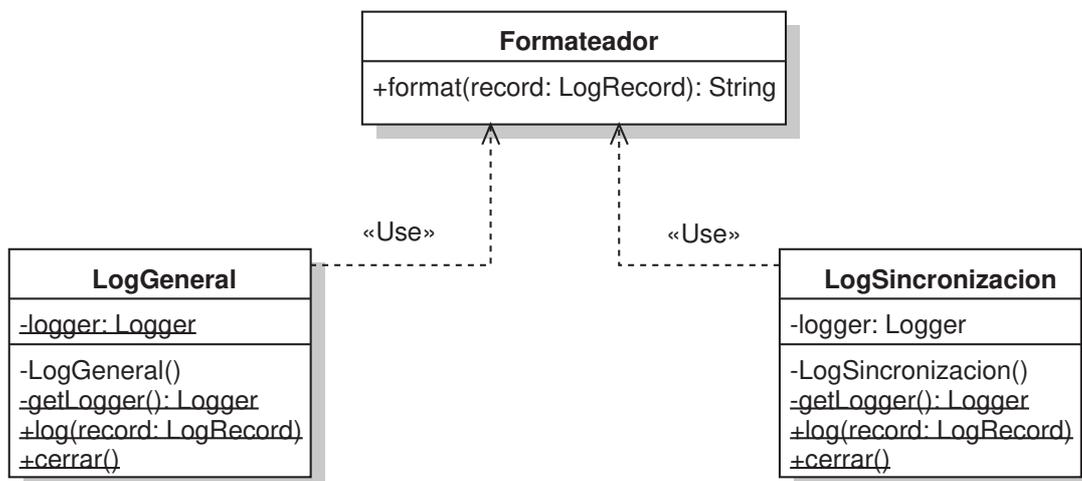


Figura 54 - Diagrama de clase tools 02

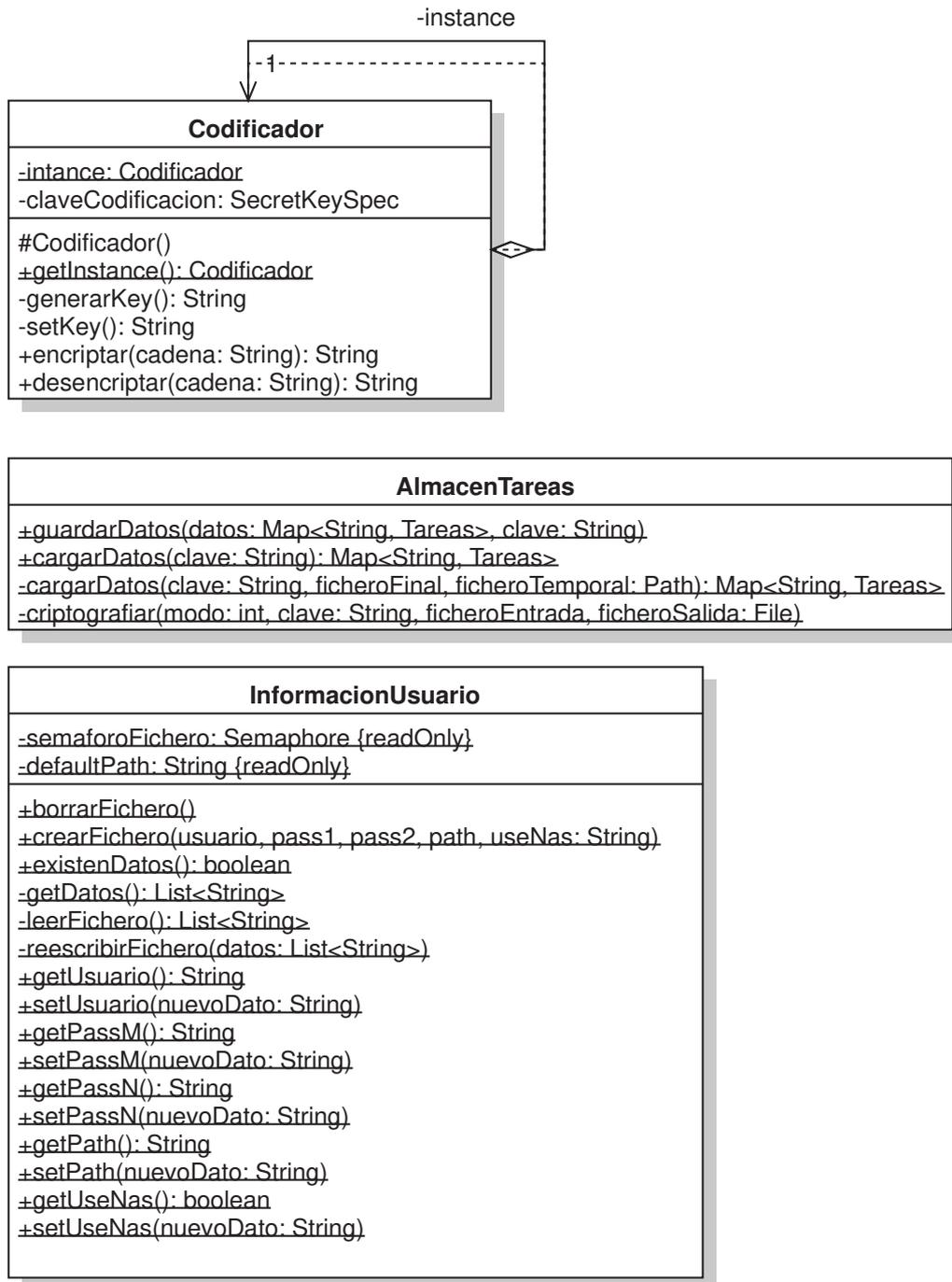


Figura 55 - Diagrama de clase: tools 03

## 8.2. Listado de código

El código de la aplicación entregada es la versión 1.06 que podemos encontrar en el siguiente enlace o QR.

- <http://bit.ly/2BuvnMA>



# Test evaluación

## 8.3. Encuestas Beta-tester

Fecha: 05 de diciembre de 2018

Datos evaluador			
Nombre:	Sergio		
Apellido 1:	Alloza	Apellido 2:	García
NIP	542088	Correo:	542088@unizar.es

Seleccione de las siguientes opciones una valoración de 1 a 5 siendo 1 muy en desacuerdo y 5 muy de acuerdo.

General	
La instalación ha sido sencilla	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input checked="" type="radio"/> 5
Es claro el uso de la aplicación	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input checked="" type="radio"/> 5
En un simple vistazo se puede identificar el uso de cada pestaña	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input checked="" type="radio"/> 4 <input type="radio"/> 5

Seleccione de las siguientes opciones una valoración de 1 a 5 siendo 1 muy en desacuerdo y 5 muy de acuerdo.

Específico	
Ha quedado claro cada elemento en la creación de un perfil	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input checked="" type="radio"/> 4 <input type="radio"/> 5
Ha quedado claro cada elemento en la edición de un perfil	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input checked="" type="radio"/> 4 <input type="radio"/> 5
Es fácil cambiar un elemento del perfil (nombre, contraseñas, ruta de almacenamiento)	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input checked="" type="radio"/> 4 <input type="radio"/> 5
Es capaz de identificar el significado de cada elemento en la tabla de entregas	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input checked="" type="radio"/> 5
Está conforme con el tiempo de arranque del programa	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input checked="" type="radio"/> 4 <input type="radio"/> 5
Está conforme con el tiempo de sincronización	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input checked="" type="radio"/> 4 <input type="radio"/> 5

Seleccione de las siguientes opciones una valoración de 1 a 5 siendo 1 muy en desacuerdo y 5 muy de acuerdo.

Opinión	
Utilizaría esta aplicación en el día a día	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input checked="" type="radio"/> 5
Recomendaría el uso de la aplicación a otros	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input checked="" type="radio"/> 5
Aplicación útil para usuarios Moodle, te ahorra tiempo navegando entre las carpetas de las asignaturas. Sencilla de usar.	

# Test evaluación

Fecha: 28 de noviembre de 2018

Datos evaluador			
Nombre:	Mª Luisa		
Apellido 1:	Rubio	Apellido 2:	Marzo
NIP	546784	Correo:	536784@unizar.es

Seleccione de las siguientes opciones una valoración de 1 a 5 siendo 1 muy en desacuerdo y 5 muy de acuerdo.

General	
La instalación ha sido sencilla	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input checked="" type="radio"/> 5
Es claro el uso de la aplicación	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input checked="" type="radio"/> 4 <input type="radio"/> 5
En un simple vistazo se puede identificar el uso de cada pestaña	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input checked="" type="radio"/> 4 <input type="radio"/> 5

Seleccione de las siguientes opciones una valoración de 1 a 5 siendo 1 muy en desacuerdo y 5 muy de acuerdo.

Específico	
Ha quedado claro cada elemento en la creación de un perfil	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input checked="" type="radio"/> 4 <input type="radio"/> 5
Ha quedado claro cada elemento en la edición de un perfil	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input checked="" type="radio"/> 4 <input type="radio"/> 5
Es fácil cambiar un elemento del perfil (nombre, contraseñas, ruta de almacenamiento)	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input checked="" type="radio"/> 4 <input type="radio"/> 5
Es capaz de identificar el significado de cada elemento en la tabla de entregas	<input type="radio"/> 1 <input type="radio"/> 2 <input checked="" type="radio"/> 3 <input type="radio"/> 4 <input type="radio"/> 5
Está conforme con el tiempo de arranque del programa	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input checked="" type="radio"/> 5
Está conforme con el tiempo de sincronización	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input checked="" type="radio"/> 5

Seleccione de las siguientes opciones una valoración de 1 a 5 siendo 1 muy en desacuerdo y 5 muy de acuerdo.

Opinión	
Utilizaría esta aplicación en el día a día	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input checked="" type="radio"/> 5
Recomendaría el uso de la aplicación a otros	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input checked="" type="radio"/> 5
En el inicio de la aplicación, pondría un título como: Inicio de sesión o dale aquí para iniciar. Ya que no es muy intuitivo al solo aparecer el icono. Cuando se solicita la ruta de la carpeta pondría una por defecto para el usuario, con la opción de modificarlo si quiere. Que haya un botón de cerrar sesión. En general lo veo muy útil.	

# Test evaluación

Fecha: 16 de diciembre de 2018

Datos evaluador			
Nombre:	Guillermo		
Apellido 1:	Morales	Apellido 2:	Blanco
NIP	698394	Correo:	698394@unizar.es

Seleccione de las siguientes opciones una valoración de 1 a 5 siendo 1 muy en desacuerdo y 5 muy de acuerdo.

General	
La instalación ha sido sencilla	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input checked="" type="radio"/> 4 <input type="radio"/> 5
Es claro el uso de la aplicación	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input checked="" type="radio"/> 4 <input type="radio"/> 5
En un simple vistazo se puede identificar el uso de cada pestaña	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input checked="" type="radio"/> 5

Seleccione de las siguientes opciones una valoración de 1 a 5 siendo 1 muy en desacuerdo y 5 muy de acuerdo.

Específico	
Ha quedado claro cada elemento en la creación de un perfil	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input checked="" type="radio"/> 5
Ha quedado claro cada elemento en la edición de un perfil	<input type="radio"/> 1 <input type="radio"/> 2 <input checked="" type="radio"/> 3 <input type="radio"/> 4 <input type="radio"/> 5
Es fácil cambiar un elemento del perfil (nombre, contraseñas, ruta de almacenamiento)	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input checked="" type="radio"/> 4 <input type="radio"/> 5
Es capaz de identificar el significado de cada elemento en la tabla de entregas	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input checked="" type="radio"/> 4 <input type="radio"/> 5
Está conforme con el tiempo de arranque del programa	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input checked="" type="radio"/> 4 <input type="radio"/> 5
Está conforme con el tiempo de sincronización	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input checked="" type="radio"/> 4 <input type="radio"/> 5

Seleccione de las siguientes opciones una valoración de 1 a 5 siendo 1 muy en desacuerdo y 5 muy de acuerdo.

Opinión	
Utilizaría esta aplicación en el día a día	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input checked="" type="radio"/> 4 <input type="radio"/> 5
Recomendaría el uso de la aplicación a otros	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input checked="" type="radio"/> 4 <input type="radio"/> 5
Opinión, observaciones, críticas	

# Test evaluación

Fecha: 19 de diciembre de 2018

Datos evaluador			
Nombre:	Laura		
Apellido 1:	Herrera	Apellido 2:	Ruiz
NIP	680723	Correo:	680723@unizar.es

Seleccione de las siguientes opciones una valoración de 1 a 5 siendo 1 muy en desacuerdo y 5 muy de acuerdo.

General	
La instalación ha sido sencilla	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input checked="" type="radio"/> 5
Es claro el uso de la aplicación	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input checked="" type="radio"/> 5
En un simple vistazo se puede identificar el uso de cada pestaña	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input checked="" type="radio"/> 5

Seleccione de las siguientes opciones una valoración de 1 a 5 siendo 1 muy en desacuerdo y 5 muy de acuerdo.

Específico	
Ha quedado claro cada elemento en la creación de un perfil	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input checked="" type="radio"/> 4 <input type="radio"/> 5
Ha quedado claro cada elemento en la edición de un perfil	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input checked="" type="radio"/> 4 <input type="radio"/> 5
Es fácil cambiar un elemento del perfil (nombre, contraseñas, ruta de almacenamiento)	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input checked="" type="radio"/> 5
Es capaz de identificar el significado de cada elemento en la tabla de entregas	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input checked="" type="radio"/> 5
Está conforme con el tiempo de arranque del programa	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input checked="" type="radio"/> 5
Está conforme con el tiempo de sincronización	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input checked="" type="radio"/> 5

Seleccione de las siguientes opciones una valoración de 1 a 5 siendo 1 muy en desacuerdo y 5 muy de acuerdo.

Opinión	
Utilizaría esta aplicación en el día a día	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input checked="" type="radio"/> 5
Recomendaría el uso de la aplicación a otros	<input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input checked="" type="radio"/> 5
La creación del perfil, podría estar mas claro. Una vez realizado validado el perfil, se podría cambiar el NIP por el nombre del usuario; asi como utilizar la imagen de perfil almacenada en Moodle.	