

Trabajo Fin de Grado

SLAM: un sistema para la gestión de los acuerdos de estudios en los planes de movilidad universitarios

SLAM: a System for Learning Agreement Management in Mobility Studies

Autor

Diego Grande Alagón

Director

Francisco Javier Fabra Caro

Grado en Ingeniería Informática

Escuela de Ingeniería y Arquitectura

2016

SLAM: un sistema para la gestión de los acuerdos de estudios en los planes de movilidad universitarios

Resumen

El proceso actual que realizan los estudiantes para poder participar en el programa de Erasmus requiere de la cumplimentación del Acuerdo de Estudios. Para que este documento pueda ser formalizado siguiendo el procedimiento establecido por la universidad, es necesario que anteriormente sea validado por el coordinador de movilidad.

En la actualidad, este proceso se basa en el intercambio de un fichero con las asignaturas a cursar en la universidad de destino y las que se convalidarán en la universidad de origen. Este intercambio se realiza con el coordinador de movilidad a través del correo electrónico, y es un proceso muy ineficiente desde una perspectiva de utilización de recursos, tanto humanos como temporales.

Debido a la necesidad de agilizar el proceso de creación, revisión y validación de los acuerdos de estudios, se ha desarrollado una plataforma a través de la cual, tanto los estudiantes como los coordinadores, pueden realizar una gestión completa de las distintas etapas que atraviesan los acuerdos.

El sistema desarrollado cubre a través de distintos módulos todos los aspectos involucrados en la gestión de los acuerdos de estudio, permitiendo administrar y gestionar la información relativa a los estudiantes, a los coordinadores, a las asignaturas, a las titulaciones y a las universidades de destino. Además, se permite la carga automática de titulaciones y asignaturas y se ofrece un servicio de comunicación entre usuarios para facilitar el intercambio de información.



DECLARACIÓN DE AUTORÍA Y ORIGINALIDAD

(Este documento debe acompañar al Trabajo Fin de Grado (TFG)/Trabajo Fin de Máster (TFM) cuando sea depositado para su evaluación).

D./D^a. Diego Grande Alagón

con nº de DNI 18053583D en aplicación de lo dispuesto en el art.

14 (Derechos de autor) del Acuerdo de 11 de septiembre de 2014, del Consejo

de Gobierno, por el que se aprueba el Reglamento de los TFG y TFM de la

Universidad de Zaragoza,

Declaro que el presente Trabajo de Fin de (Grado/Máster)
Grado _____, (Título del Trabajo)

SLAM: un sistema para la gestión de los acuerdos de estudios en los planes de
movilidad universitarios

es de mi autoría y es original, no habiéndose utilizado fuente sin ser citada
debidamente.

Zaragoza, 22 de Noviembre 2016

Fdo: Diego Grande Alagón

Índice

1. Introducción	5
1.1. Contexto	5
1.2. Motivación	6
1.3. Objetivos	6
1.4. Estado del arte	7
1.4. Organización	8
2. Análisis del Problema	10
2.1. Requisitos	10
2.2. Casos de uso	11
2.3. Tecnologías de desarrollo	13
3. Diseño e implementación	15
3.1. Diagrama de flujo de un Acuerdo de Estudios	15
3.2. Diseño arquitectural	16
3.3. Modelo de datos	17
3.4. Despliegue	18
3.5. Interfaz de servicios RESTful	18
3.6. Implementación	19
4. Evaluación y Pruebas	26
5. Conclusiones	27
5.1. Conclusiones	27
5.2. Trabajo futuro	27
5.3. Valoración Personal	28
6. Gestión del Proyecto	29
6.1. Gestión de la configuración	29
6.2. Planificación	29
6.3. Distribución Temporal	31
7. Bibliografía	32
8. Anexos	33
8.1. Anexo I: Manuales de Usuario	33
8.1.1. Manual de usuario Estudiante	33
8.1.2. Manual de usuario Coordinador	48
8.1.3. Manual de usuario Administrador	64
8.2. Anexo II: Manual de Despliegue	81
8.3. Anexo III: Requisitos del Sistema y Casos de Uso (por perfil de usuario)	82
8.4. Anexo IV: Detalles de diseño e implementación	88
8.5. Anexo V: Pruebas	94

1. Introducción

1.1. Contexto

El Acuerdo de Estudios o *Learning Agreement* es el documento que han de cumplimentar los estudiantes para poder participar en un programa de movilidad. En éste se registran las asignaturas que se desean cursar en la universidad destino y las correspondientes convalidaciones con las asignaturas de la universidad origen. A través de este documento el estudiante y las dos universidades implicadas dejan constancia de las características del intercambio.

El proceso desde que un estudiante cumplimenta por primera vez el documento, hasta que es validado por el coordinador, necesita en muchas ocasiones de varias reuniones, el intercambio de correos electrónicos con las correcciones oportunas y la realización de diferentes versiones hasta que el coordinador puede validarlo, un proceso totalmente ineficiente y que requiere de mucho tiempo por parte del estudiante y del coordinador. Es allí donde surge la necesidad de crear una aplicación que unifique en una sola plataforma todos estos aspectos, permitiendo gestionar las etapas de los acuerdos de estudios y los componentes que intervienen en su desarrollo de una forma más eficiente, ahorrando tiempo y trabajo a ambas partes.

En el presente trabajo se ha desarrollado la plataforma SLAM (Student Learning Agreement Management), una herramienta web cuyo principal objetivo es el de gestionar, elaborar y validar los acuerdos de estudio de una forma ágil y sencilla.

La aplicación permite el acceso a través de tres perfiles de usuario distintos (estudiante/coordinador/administrador), con una interfaz específica para cada uno de ellos y adaptable según el tamaño de pantalla donde se visualice.

Además del módulo de gestión de acuerdos de estudio, la aplicación dispone de los siguientes sistemas:

- Sistema de carga automática de titulaciones y asignaturas de la Universidad de Zaragoza
- Sistema para administrar los coordinadores y estudiantes registrados en la aplicación
- Sistema de administración de asignaturas tanto de la universidad de Zaragoza como de las universidades donde los estudiantes desean realizar el programa de movilidad
- Sistema de comunicación para enviar y recibir mensajes entre ambas partes de forma fácil y rápida.

1.2. Motivación

A continuación se presentan los motivos por los que se ha decidido desarrollar esta aplicación:

El desconocimiento de las tecnologías con las que se ha desarrollado el proyecto ha sido uno de los puntos clave, ya que pese a cursar asignaturas que enseñaban algunos de los conceptos desarrollados, las tecnologías con las que se ha trabajado no se habían visto antes. Este aspecto es de principal interés a nivel profesional, ya que permite aumentar la experiencia en nuevos campos y ampliar la rama de conocimientos en distintas tecnologías informáticas.

La gran demanda de desarrolladores con conocimientos en AngularJS o Node.js es otro motivo por el que se escogió trabajar en este proyecto. Actualmente son tecnologías muy usadas en las empresas debido a su flexibilidad, robustez y el gran soporte de desarrolladores que poseen.

Otro de los motivos ha sido desarrollar una aplicación que permite tanto al estudiante como al coordinador ahorrar mucho tiempo y esfuerzo. El diseño de la aplicación está principalmente enfocado para agilizar el proceso de creación y verificación de los Acuerdos.

Para finalizar, otro de los motivos importantes ha sido el hecho de no existir una aplicación de estas características, ya que como se verá más adelante, el resto de universidades españolas tienen una metodología de trabajo similar a la de la Universidad de Zaragoza.

1.3. Objetivos

A continuación se presentan los principales objetivos que se desean conseguir con el desarrollo de la aplicación:

1. Reducir el tiempo que invierten los estudiantes y los coordinadores en las diferentes etapas por las que pasa un Acuerdo de Estudios hasta que es cerrado.
2. Facilitar la comunicación entre estudiantes y coordinadores a través de un canal de comunicación exclusivo con notificaciones.
3. Almacenar la información de las titulaciones de la Universidad de Zaragoza y sus correspondientes asignaturas de forma automática a través de la técnica de Web Scraping.
4. Desarrollar un middleware de seguridad, que se encargue de comprobar el nivel de acceso del usuario que realiza la solicitud y que permita o deniegue el recurso solicitado.
5. Los alumnos y los coordinadores deben ser capaces de gestionar de forma clara y precisa todos los aspectos relacionados con los Acuerdos de Estudios.
6. Generar una aplicación que se adapte a cualquier tipo de pantalla.

1.4. Estado del arte

En esta sección se va a presentar el porfolio de documentos que utilizan distintas universidades españolas para gestionar los Acuerdos de Estudio, incluida la universidad de Zaragoza.

A		B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	S	
Escuela de Ingeniería y Arquitectura Universidad Zaragoza												PLAN DE RECONOCIMIENTO DE CREDITOS						FOTO
Apellidos:						Titulación / Grado:												
Nombre:						Destino:												
DNI:						Coordinador:												
NIA:						Curso:												
e-mail:						Estancia de uno/dos semestres:												
ASIGNATURAS A RECONOCER EN LA EIHA												ASIGNATURAS A CURSAR EN EL DESTINO						(3)
			(1)	(2)														
			U	T	M	E	E	N	O					U	E	N		
			r	I	A	C	T	N	T					r	S	C		
			s	P	T	R	S	A	A					s	e	C		
			o	R	S	A	A							o	.	T		
	Código	Nombre										Código	Nombre	Enlace				
17		Sistemas de información	3º	T	0	6						TL0-11006	Basics of Information and Knowledge Management	http://www.tut.fi/wwwppaat/opas201	0	4		
18												TL0-25456	Supply Chain Management	http://www.tut.fi/wwwppaat/opas201	0	4		
19		Algoritma para problemas difíciles	4º	O	0	6						MAT-72006	Advanced Algorithms and Data Structures	http://www.tut.fi/wwwppaat/opas201	0	7		
20		Informática Gráfica	4º	O	0	6						SGN-12006	Basic Course in Image and Video Processing	http://www.tut.fi/wwwppaat/opas201	0	5		
21																		
22		Subtotal Troncales/Obligatorias/Común				18									Subtotal		20	
23		Almacenes y minería de datos	4º	Op	0	6						SGN-43006	Knowledge Mining and Big Data	http://www.tut.fi/wwwppaat/opas201	0	5		
24		Tecnologías de la información en la empresa	4º	Op	0	6						TTA-11016	Basics of Industrial Management	http://www.tut.fi/wwwppaat/opas201	0	4		
25		Subtotal Optativas				12									Subtotal		9	
26																		
27																		
28																		
29																		
30		Subtotal Libre Elección/Trabajo de Fin de Grado				0									Subtotal		0	
31																		
32		TOTAL				30									TOTAL		29	
33																		


Figura 1. Acuerdo de Estudios (Universidad de Zaragoza)

La figura 1 muestra el formulario en Excel que deben cumplimentar los estudiantes de la Universidad de Zaragoza.

De manera representativa se han seleccionado los Acuerdos de Estudios de las universidades de Barcelona (figura 2) y Sevilla (figura 3).

U UNIVERSITAT DE BARCELONA B		
LEARNING AGREEMENT / CONTRACTE D'ESTUDIS / CONTRATO DE ESTUDIOS		
ACADEMIC YEAR ANY ACADEMIC ANO ACADEMICO	2006-07	
PERIOD/PERIODE/PERIODO		
FIELD OF STUDY I'ENSENYAMENT/I'ENSENANZA		
Name of student Nom de l'estudiant Nombre del estudiante		
Home institution Institució d'origen Institución de origen	Universitat de Barcelona (E BARCELONA 01)	
Host institution Institució de destinació Institución de destino	Centre UB/Centro UB/UB Centre:	
Country País	Espanya España Spain	
Country País		
Course unit code (if any) Codi de l'assignatura Código de la asignatura	Course unit title Nom de l'assignatura Nombre de la asignatura	Number of ECTS credits Nombre de crèdits ECTS Número de créditos ECTS
Student's signature Signatura de l'estudiant Firma del estudiante	Date Data Fecha	

Figura 2. Acuerdo de Estudios (Universidad de Barcelona)



**ACUERDO DE ESTUDIOS /
LEARNING AGREEMENT
ALUMNOS DE INTERCAMBIO /
EXCHANGE STUDENTS
UNIVERSIDAD DE SEVILLA /
UNIVERSITY OF SEVILLA**

FOTO

PROGRAMA / PROGRAMME	CURSO ACADÉMICO / ACADEMIC YEAR
MOVILIDAD INTERNACIONAL "UNIVERSIDAD DE SEVILLA" INTERNATIONAL MOBILITY "UNIVERSITY OF SEVILLA"	2011-2012
PERIODO DE ESTUDIOS / PERIOD OF STUDIES	
Desde / From: _____ Hasta / To: _____	

DATOS PERSONALES DEL ESTUDIANTE / STUDENT'S PERSONAL DATA

APELLIDOS Y NOMBRE / SURNAME, NAME	Nº DE PASAPORTE / ID / PASSPORT	SEXO / SEX	EDAD / AGE
DOMICILIO / ADDRESS		POBLACIÓN / CITY	
CP / PROVINCIA / POSTAL CODE / REGION	Teléfono / Telephone	Correo Electrónico / e-Mail	

DATOS DEL INTERCAMBIO / DATA OF THE EXCHANGE AGREEMENT

CENTRO DE ORIGEN / SENDING INSTITUTION
--

ABRUMBRAS EN CENTRO DESTINO / COURSE DET. TITLE AT THE HOST INSTITUTION	APELL. INICIAL / INITIAL YEAR / SEM.	Nº HORAS SEMESTRALES / SEMESTER HOURS PER WEEK	CREDITOS / CREDITS	ABRUMBRAS RECONOCIDAS EN CENTRO ORIGEN / EQUIVALENT COURSE DET. TITLE AT THE HOME INSTITUTION	CREDITOS RECONOCIDOS / CREDITS TO BE RECOGNIZED
TOTAL CREDITOS / TOTAL CREDITS				TOTAL CREDITOS RECONOCIDOS / TOTAL CREDITS TO BE RECOGNIZED	

CRÉDITOS MÍNIMOS A CURSAR / MINIMUM WORK LOAD IN CREDITS:

- Estancias de curso completo: 45 créditos/ Full academic year stay: 45 credits
- Estancias de un semestre: 20 créditos / One semester stay: 20 credits

EN CASO DE CURSAR UN NÚMERO DE CRÉDITOS INFERIOR / STUDENTS TAKING LESS CREDITS:

15 créditos/cursos complejos 15 créditos/estancias de 1 mes 10 créditos/estancias de 1-5 meses

EL ESTUDIANTE / STUDENT	INSTITUCIÓN DE ORIGEN / HOME INSTITUTION	INSTITUCIÓN DE ACOGIDA / HOST INSTITUTION
Fdo. / Signed	Fdo. / Signed Responsable: RR. Internacionales del Centro / International Relations Coordinator of the Faculty or School	Fdo. / Signed

Figura 3. Acuerdo de estudios (Universidad de Sevilla)

Como se puede observar la metodología seguida es similar en las tres universidades. Estos formularios deben ser revisados y firmados por los correspondientes coordinadores de estudios para que los estudiantes puedan formalizar su solicitud.

1.4. Organización

La memoria del proyecto está estructurada en diferentes secciones:

Sección 1. Introducción: formada por el contexto en el que se ha desarrollado la aplicación, los motivos por los que se ha decidido realizar este trabajo, los objetivos que se han de cumplir para darlo como satisfactorio y el estado del arte donde se presentan algunos ejemplos de acuerdos de estudios utilizados por universidades españolas.

Sección 2. Análisis de problema: donde se definen los requisitos de la aplicación, se representa los casos de uso y se analizan las tecnologías de desarrollo elegidas, así como las causas por las que han sido seleccionadas éstas y no otras.

Sección 3. Diseño e implementación: en primer lugar se muestra la arquitectura de la aplicación a alto nivel y el modelo de datos empleado, se indica como se ha realizado el despliegue de los distintos módulos a lo largo del proyecto, la interfaz de la API, como se ha realizado su implementación y la aplicación cliente.

Sección 4. Evaluación y pruebas: se definen las pruebas realizadas al sistema tanto en la API como en el cliente para asegurar el correcto funcionamiento.

Sección 5. Conclusiones: compuesta por la valoración personal del proyecto, las conclusiones finales y las posibles mejoras que podría tener la aplicación en el futuro.

Sección 6. Gestión del proyecto: formada por la gestión de configuración y la planificación.

Sección 7. Bibliografía

Sección 8. Anexos: documentos adjuntos a la memoria con los siguientes contenidos:

- Anexo 1. Manuales de usuario (estudiante/coordinador/administrador).
- Anexo 2. Manual donde se explica cómo realizar el despliegue de la aplicación.
- Anexo 3. Tablas de requisitos y casos de uso detallados según el perfil de Usuario.
- Anexo 4. Detalles de diseño e implementación.
- Anexo 5. Pruebas.

2. Análisis del Problema

2.1. Requisitos

A continuación se presentan los requisitos que debe satisfacer el sistema.

Requisitos funcionales

Código	Descripción
RF-1	Deberá permitir dar de alta a estudiantes
RF-2	Deberá permitir dar de alta a coordinadores
RF-3	Deberá permitir gestionar los Acuerdos de Estudios
RF-4	Deberá permitir gestionar las titulaciones de la Universidad de Zaragoza
RF-5	Deberá permitir gestionar las asignaturas de la Universidad de Zaragoza
RF-6	Deberá permitir gestionar las universidades de destino
RF-7	Deberá permitir gestionar las asignaturas de las universidades destino
RF-8	Permitirá gestionar los Estudiantes
RF-9	Deberá permitir gestionar los Usuarios
RF-10	Deberá permitir gestionar el perfil de usuario
RF-11	Deberá permitir gestionar el buzón de mensajes
RF-12	Deberá permitir el envío de mensajes entre estudiantes y sus coordinadores
RF-13	Deberá mostrar una notificación en caso de tener mensajes sin leer.
RF-14	Deberá permitir realizar web Scraping de la titulaciones y asignaturas de la Universidad de Zaragoza
RF-15	Deberá permitir filtrar y ordenar los datos de las titulaciones, asignaturas, universidades, acuerdos de estudio, estudiantes y usuarios en función de cualquiera de sus atributos.
RF-16	Generará y enviará a través del e-mail de forma autónoma las contraseñas de los usuarios

Requisitos no funcionales

Código	Descripción
RNF-1	Implementará una interfaz adaptativa
RNF-2	Será accesible a través de cualquier navegador moderno.
RNF-3	Implementará un sistema de seguridad basado en niveles de acceso, según el perfil de usuario con el que se acceda. Los perfiles de usuario serán los siguientes: administrador, coordinador y estudiante.
RNF-4	Deberá permitir acceder a los recursos de la aplicación a través de la interfaz de usuario y a través de llamadas a servicios web que cumplan con la arquitectura Rest.
RNF-5	Deberá ser robusta y escalable, por lo que el desarrollo de la Api será modular.
RNF-6	El front-end deberá estar desarrollado con Angular y Bootstrap
RNF-7	Almacenará únicamente el Hash de las contraseñas

En el [anexo 3](#) se pueden encontrar los requisitos de cada perfil de usuario con un nivel de detalle superior.

2.2. Casos de uso

A continuación se presenta el modelado de los requisitos funcionales del sistema mediante un diagrama de casos de uso.

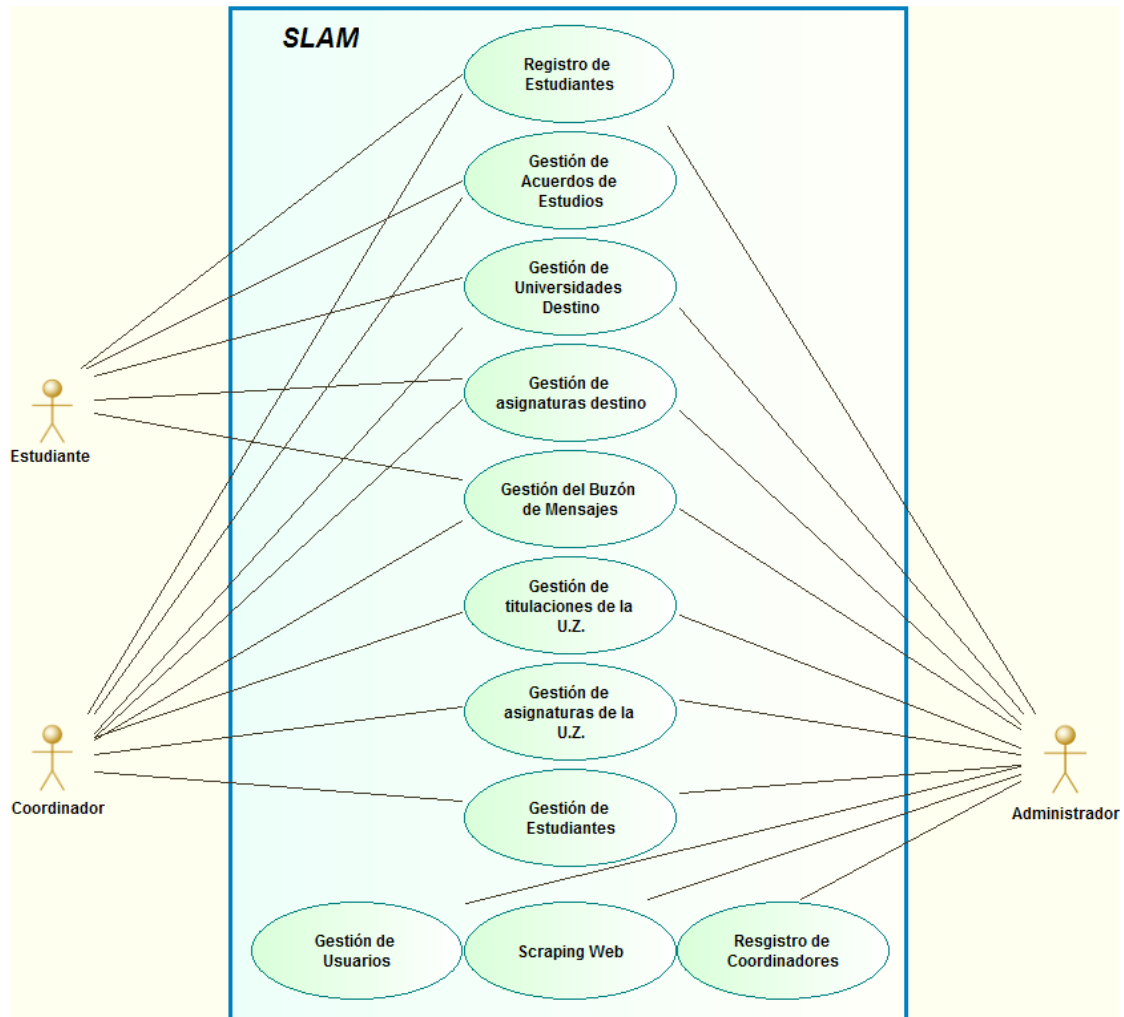


Figura 4. Diagrama casos de uso

El diagrama contiene tres actores:

- **Actor Estudiante:** actor primario cuyo principal cometido es el de gestionar su acuerdo de estudios para que el coordinador pueda revisarlo y aprobarlo.
- **Actor Coordinador:** actor primario encargado de revisar, evaluar y gestionar los acuerdos de estudios de los estudiantes.
- **Actor Administrador:** actor primario responsable de administrar diferentes componentes de la aplicación.

A continuación se detallan los casos de uso:

- **Registro de estudiantes:** el caso de uso permite dar de alta a estudiantes en la aplicación.
- **Gestión de Acuerdos de Estudio:** el caso de uso permite realizar diferentes operaciones sobre los acuerdos de estudio:
 - o **Estudiante:** puede crear y modificar su acuerdo de estudios.
 - o **Coordinador:** puede validar y evaluar el acuerdo de estudios de los estudiantes inscritos en su titulación.
- **Gestión de universidades destino:** el caso de uso permite realizar diferentes operaciones sobre las universidades destino:
 - o **Estudiante:** puede crear nuevas universidades.
 - o **Coordinador:** puede crear y editar el contenido de las universidades.
 - o **Administrador:** puede crear, borrar y editar el contenido de las universidades.
- **Gestión de asignaturas destino:** el caso de uso permite realizar diferentes operaciones sobre las asignaturas pertenecientes a las universidades de destino:
 - o **Estudiante:** puede crear asignaturas y editar aquellas que ha creado.
 - o **Coordinador:** puede crear y editar el contenido de las asignaturas.
 - o **Administrador:** puede crear, borrar y editar el contenido de las asignaturas.
- **Gestión de buzón de mensajes:** el caso de uso permite realizar diferentes operaciones sobre las conversaciones.
 - o **Estudiante:** puede comunicarse con su coordinador.
 - o **Coordinador:** puede comunicarse con los estudiantes de su titulación y gestionar las conversaciones, además de enviar mensajes de difusión.
 - o **Administrador:** puede comunicarse con los coordinadores y gestionar su conversaciones, además de enviar mensajes de difusión a los mismos.
- **Registro de coordinadores:** el caso de uso permite dar de alta en el sistema a los coordinadores.
- **Gestión de titulaciones de la Universidad de Zaragoza:** el caso de uso permite realizar diferentes operaciones sobre las titulaciones pertenecientes a la Universidad de Zaragoza.
 - o **Estudiante:** puede visualizar las titulaciones.
 - o **Coordinador:** puede visualizar todas las titulaciones pero únicamente puede editar el contenido de aquellas de las es coordinador.
 - o **Administrador:** puede crear, editar y borrar las titulaciones.
- **Gestión de las asignaturas de la Universidad de Zaragoza:** el caso de uso permite realizar diferentes operaciones sobre las asignaturas pertenecientes a las titulaciones.
 - o **Estudiante:** puede visualizar las asignaturas.
 - o **Coordinador:** puede visualizar todas las asignaturas, pero únicamente puede crear y editar el contenido de aquellas que pertenecen a la titulación de la cual es coordinador.
 - o **Administrador:** puede crear, editar y borrar las asignaturas.

- **Gestión de los estudiantes:** el caso de uso permite realizar diferentes operaciones sobre los estudiantes registrados en la aplicación.
 - o **Coordinador:** puede visualizar, editar y desactivar las cuentas de los estudiantes que registrados en su titulación.
 - o **Administrador:** puede crear, editar, borrar y desactivar los estudiantes.
- **Gestión de los usuarios:** el caso de uso permite crear, editar, borrar y desactivar las cuentas de los distintos usuarios. En caso de eliminar un usuario con perfil de estudiante los datos del estudiante asociado permanecerán visibles en la aplicación.
- **Scraping Web:** el caso de uso permite descargar, insertar y actualizar la información relativa a las titulaciones y asignaturas de la Universidad de Zaragoza.

En el [anexo 3](#) se pueden visualizar los diagramas de casos de uso en el que se presentan de forma detallada las funcionalidades de los tres actores.

2.3. Tecnologías de desarrollo

A continuación se describen y analizan las tecnologías con las que se ha trabajado en la aplicación.

Lenguaje de programación

- **JavaScript:** a pesar de ser un lenguaje interpretado que surgió con el objetivo de programar ciertas funcionalidades sobre páginas web, como resultado de la interacción con el usuario; en los últimos años se ha convertido en uno de los lenguajes más populares tanto para desarrollar en el front-end como en el back-end. Esto se debe principalmente a la necesidad de evolucionar que ha sufrido el lenguaje para adaptarse a los requerimientos cada vez más complejos de las aplicaciones web. Como consecuencia JavaScript ha alcanzado niveles de complejidad y de prestaciones iguales o superiores a los de otros lenguajes.

Tecnologías Back-end

- **Node.js:** es un entorno de ejecución de aplicaciones multiplataforma y de código abierto, desarrollado sobre el intérprete de JavaScript V8 (motor creado por Google) y sobre la que se ha implementado el servidor web que proporciona la API RESTful.
Pese a que Node.js se ejecuta sobre un único hilo, contiene la librería libuv para manejar eventos asíncronos, ésta le permite crear nuevos hilos de ejecución únicamente cuando se realizan operaciones de Entrada/Salida, las cuales se ejecutan de forma no bloqueante.
- **Express:** Es uno de los frameworks más importantes en Node.js, proporciona un conjunto de características adicionales al servidor (infraestructura MVC de la aplicación, routing de URLs...) para facilitar el desarrollo de la aplicación.
- **MongoDB:** Es un gestor de base de datos NoSQL orientado a documentos. Estos se almacenan en formato BSON (Binary JSON) y poseen un esquema dinámico, ya que cada documento de una misma colección puede tener un esquema distinto, esta flexibilidad permite una integración de los datos sencilla y rápida.

- **Mongoose:** Biblioteca para el modelado de datos de MongoDB. Permite crear esquemas donde se define la forma que tienen los documentos dentro de las colecciones y realizar la interacción con la Base de Datos a través de su API.

Tecnologías Front-end

- **AngularJS:** Es un framework MVC creado por Google que permite un desarrollo rápido y potente de aplicaciones web SP (Single-Page) basadas en JavaScript del lado del cliente. En la primera petición que se realiza al servidor, se ejecuta la carga de todas las plantillas HTML así como sus controladores, esto permite que en las siguientes peticiones la transferencia de información sea mínima (JSON con el recurso solicitado) por lo que la velocidad de carga es mucho mayor que si se tuviese que renderizar de nuevo toda la página.
- **HTML:** Lenguaje de marcas para el desarrollo de páginas web.
- **Bootstrap:** Es un framework CSS desarrollado inicialmente por Twitter que permite crear interfaces responsive y provee de librerías CSS que incluyen tipografías, botones, cuadros, menús y otros elementos que pueden ser utilizados en cualquier aplicación web. Proporciona un sistema de rejilla basado en la disposición del contenido de la aplicación dentro de columnas, las cuales automáticamente escalan al tamaño y posición adecuados dependiendo del tamaño de la pantalla.

La aplicación se ha desarrollado siguiendo la arquitectura stack MEAN (MongoDB - Express - AngularJS – Node.js), todas las tecnologías usadas se apoyan en el uso de JavaScript y JSON, lo que genera una gran homogeneidad en la plataforma.

La flexibilidad a la hora de modelar los esquemas de las colecciones en MongoDB ha facilitado el desarrollo de la aplicación en las distintas fases, permitiendo realizar constantes cambios en el modelo de datos, pese a que no sea lo más adecuado.

3. Diseño e implementación

3.1. Diagrama de flujo de un Acuerdo de Estudios

A continuación se presenta el diagrama de flujo donde se especifican las distintas fases por la que transcurre un Acuerdo de Estudios, desde que es creado por el alumno hasta que es cerrado.

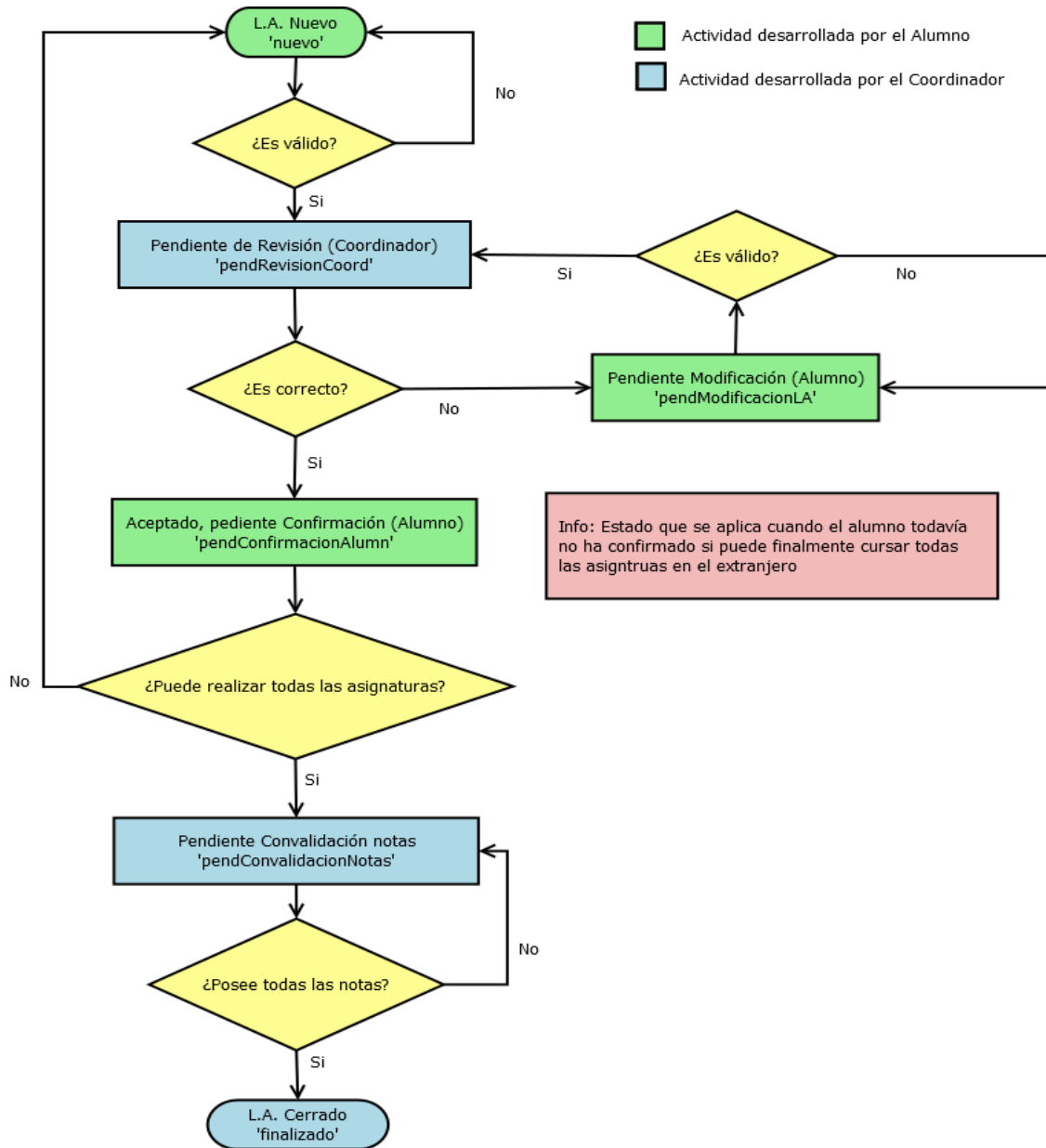


Figura 5. Diagrama de flujo Acuerdo de Estudios

3.2. Diseño arquitectural

A partir del contexto y los requisitos expuestos en los puntos anteriores, en la figura 6 se presenta la arquitectura cliente-servidor del sistema.

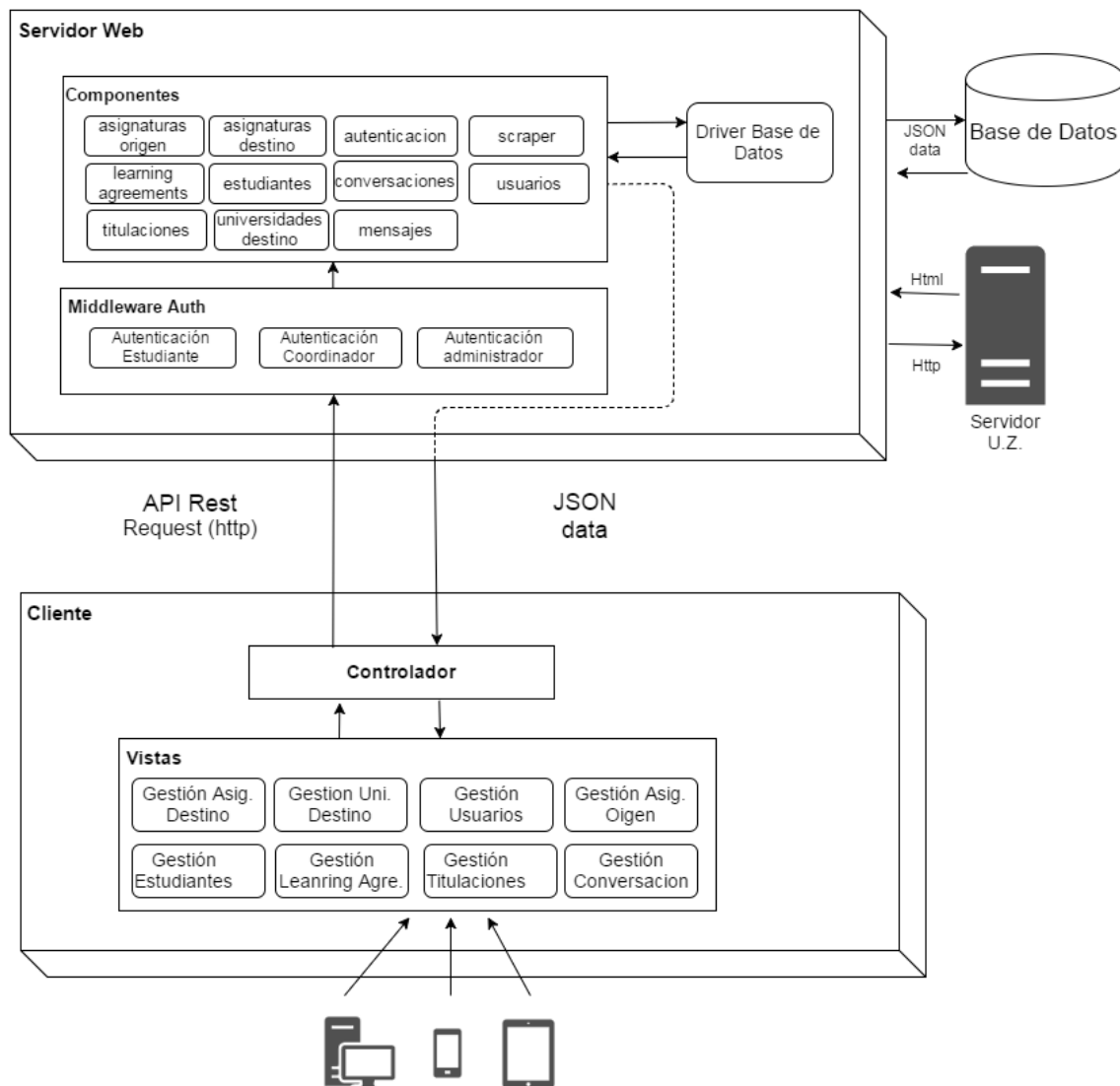


Figura 6. Arquitectura del sistema

Como se puede observar en el diagrama, el usuario accede a la aplicación a través de las vistas HTML. Cuando se produce algún evento (Ej. 'carga la info. del usuario'), el controlador de la vista correspondiente, realiza una petición http solicitando el recurso a la API.

El servidor es el encargado de recibir la petición y realizar el direccionamiento hacia el componente correcto. Una vez que el middleware comprueba que la petición tiene los permisos necesarios para acceder al recurso, el modulo correspondiente realiza las operaciones solicitadas y le devuelve a la aplicación cliente el recurso que había requerido.

Cuando controlador de la aplicación cliente recibe los datos en formato JSON, estos se cargan en la vista para que puedan ser visualizados.

3.3. Modelo de datos

En el siguiente diagrama se muestra el modelo de datos de la aplicación. En éste se representa la estructura lógica de la base de datos MongoDB con las relaciones que tienen las colecciones.

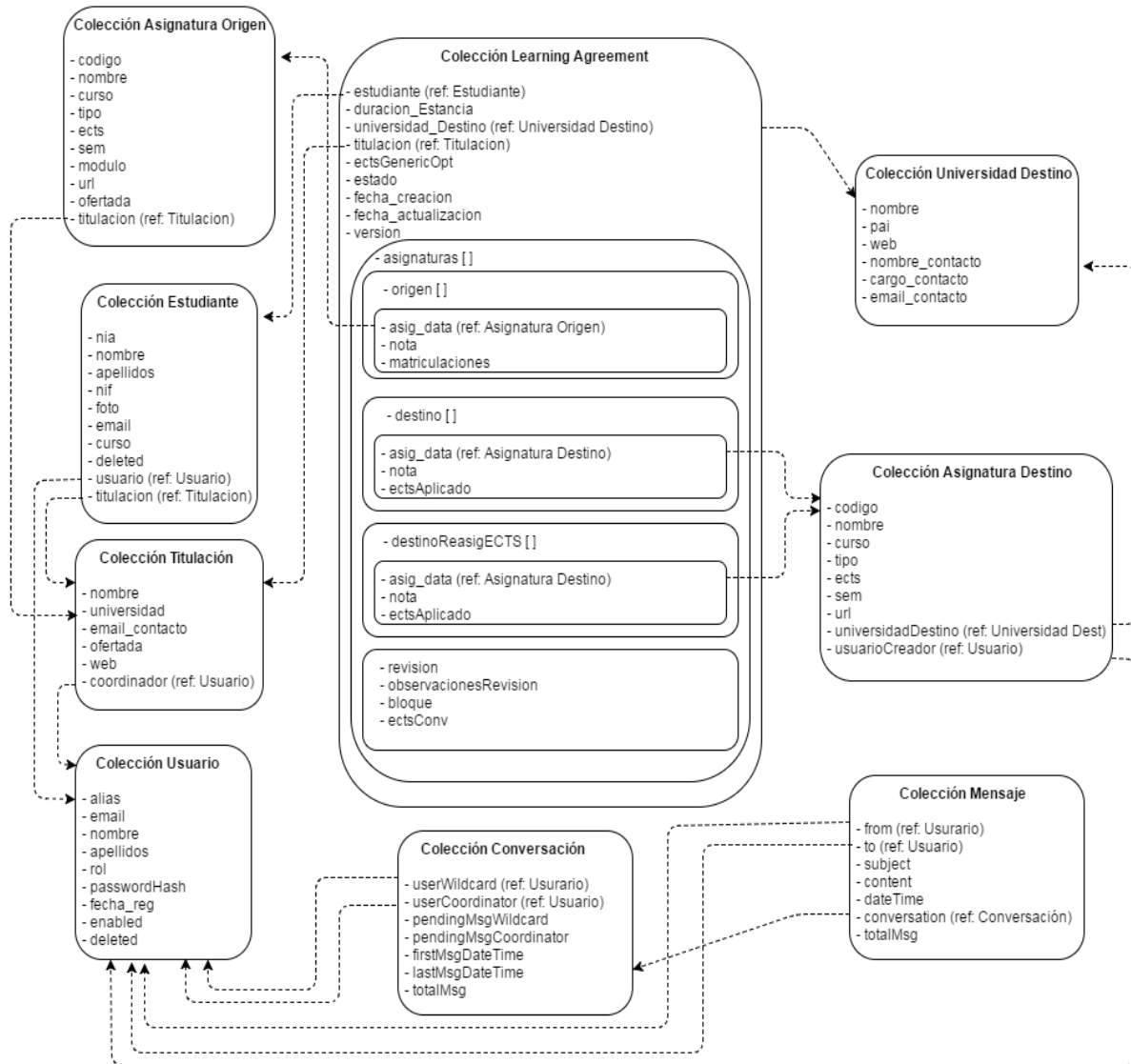


Figura 7. Modelo de datos

Cada documento de las colecciones se almacena en la base de datos con formato JSON, por lo que los atributos de las colecciones son campos <nombre-valor>.

3.4. Despliegue

El sistema se ha desarrollado en diferentes fases, siguiendo una metodología de trabajo iterativa e incremental.

El servidor web se ha desplegado sobre la versión 4.3.1 de Node.js y la aplicación cliente se ha creado haciendo uso del framework AngularJS versión 1.5.5.

Para el desarrollo de cada uno de los recursos que forman la API, en primer lugar se ha definido su modelo de datos en la aplicación, a continuación se ha creado para cada recurso un módulo de direccionamiento y un controlador donde se implementan sus funciones asociadas. Una vez el recurso está disponible y funcionando correctamente, se ha programado la interfaz del cliente y la estructura en AngularJS.

Durante todas las fases de desarrollo se han realizado las pruebas pertinentes para comprobar el correcto funcionamiento e integración.

En la primera fase se ha desarrollado el servicio de “Web Scraping”, para obtener la información de las titulaciones y asignaturas de la Universidad de Zaragoza.

En la segunda fase se ha desarrollado el acceso y la autenticación. Para ello se han implementado los módulos de usuarios, estudiantes, login y el middleware de autenticación por Token.

En la tercera fase se ha desarrollado el servicio de comunicación entre usuarios. Estos se componen de los módulos de Conversación y Mensajes.

En la cuarta fase se ha desarrollado el servicio de gestión de Learning Agreements, para ello se han implementado además del módulo de learning agreements, los módulos universidades destino y asignaturas destino.

Para finalizar se ha realizado el despliegue de la aplicación sobre una instancia Elastic Compute Cloud (EC2) T2.micro de Amazon.

El servidor en el que se ejecuta la aplicación está localizado en Oregon (US), monta un procesador Intel Xeon con 1 GB de RAM y se lanza sobre el sistema operativo Ubuntu 14.04.3.

El despliegue realizado sobre la máquina de Amazon se ha realizado únicamente para el desarrollo de pruebas del sistema.

3.5. Interfaz de servicios RESTful

Para la implementación de la API RESTful se han seguido los siguientes principios:

- 1- Todos los recursos son accesibles a través de identificadores únicos (URIs).
- 2- El envío y la recepción de datos se realiza en formato JSON.
- 3- Los recursos se manipulan a través de los métodos que provee Http (get, post, delete y put).

- 4- El servidor no almacena los estados de los clientes.
- 5- Todas las acciones que se realizan sobre un recurso envían una respuesta al cliente.

La API se compone de 11 recursos accesibles a través de URIs, que a su vez ofrecen distintos servicios en función de cual sea el recurso solicitado, formando un total de 75 servicios.

A continuación se presenta un ejemplo representativo de cómo se realizan algunas acciones sobre el recurso “estudiantes” y el resultado obtenido en función del tipo de petición Http que se realice. Todas las peticiones deben tener en el header el Token donde se guarda la información necesaria para poder realizarlas.

Uri del recurso: slam/estudiantes

Petición http	Descripción
Get	Petición al sistema que devuelve un listado de los estudiantes registrados en la aplicación.
Post	Registro de un nuevo estudiante en el sistema. <i>Body {"nombre": "Luis", "apellidos": "Cambra" ...}</i>

Uri del recurso: slam/estudiantes/:id

Petición http	Descripción				
Put	Actualiza la información del estudiante que tiene asociado el identificador parametrizado. <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Parámetro</th> <th>Descripción</th> </tr> </thead> <tbody> <tr> <td>Id</td> <td>Identificador de estudiante</td> </tr> </tbody> </table> <i>Body {"nombre": "Luis", "apellidos": "Alegre" ...}</i>	Parámetro	Descripción	Id	Identificador de estudiante
Parámetro	Descripción				
Id	Identificador de estudiante				
Delete	Elimina del sistema al usuario que tiene asociado el identificador parametrizado. <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Parámetro</th> <th>Descripción</th> </tr> </thead> <tbody> <tr> <td>Id</td> <td>Identificador de estudiante</td> </tr> </tbody> </table>	Parámetro	Descripción	Id	Identificador de estudiante
Parámetro	Descripción				
Id	Identificador de estudiante				

3.6. Implementación

A continuación se va a explicar cómo se ha realizado la implementación de las distintas funcionalidades de la aplicación tanto en el servidor como en el cliente. Analizando los principales módulos que lo forman y su función dentro del sistema.

En el [anexo 4](#) se encuentra un análisis más detallado del diseño y la implementación del sistema.

Servidor

En primer lugar se ha hecho uso de las funciones de middleware que proporciona el framework Express para implementar un sistema de direccionamiento por ficheros, el cual permite crear y gestionar las URIs de los recursos de una forma clara y organizada.

Se ha desarrollado un sistema de autenticación basado en token con un middleware donde se comprueba que el usuario que ha realizado la solicitud de un recurso tiene los privilegios adecuados para acceder al mismo.

El flujo de control es el siguiente:

1. El usuario accede a la aplicación.
2. Se valida si su nombre y contraseña son válidos y en caso afirmativo se crea un token con la información del usuario brindada por la base de datos. A continuación se le envía el token al usuario.
3. El usuario automáticamente a través de una librería de angular provee la información del token en el header de cada petición para acceder a las URIs.
4. Cada vez que el usuario realiza una petición a una URI, el middleware se encarga de comprobar la información del token y permite o bloquea el acceso al recurso, informando al usuario con la respuesta adecuada en formato JSON.

Existen tres niveles de privilegios:

1. Estudiante: nivel que se le otorga a los estudiantes.
2. Coordinador: nivel que se le otorga a los coordinadores, estos pueden acceder a todos los recursos propios y además a los recursos de los estudiantes
3. Administrador: nivel con más privilegios, ya que puede acceder a todos los recursos de la aplicación.

Los recursos disponibles a través de la API están formados por varios ficheros, en el primero se define el módulo donde se implementa su modelo de datos, en el segundo el controlador encargado realizar las operaciones necesarias para responder a la solicitud del usuario y por último el fichero de rutas donde se realiza el direccionamiento a la función del controlador asociada a la petición.

En los controladores se definen todas las operaciones que se realizan sobre los recursos. Siempre que se genere una petición al servidor, el controlador responde con un mensaje en formato JSON y con un código de status Http asociado al tipo de respuesta, si es correcta enviará un código de status 200, en caso contrario informará con el código de error correspondiente.

Se ha desarrollado el servicio de envío de mensajes al correo electrónico, que notifica al usuario las credenciales de acceso o le indica que debe realizar alguna operación sobre los Acuerdos de Estudios, para ello se ha creado una cuenta de Gmail asociada a la aplicación y se ha utilizado el módulo 'node-mailer' de Node.js.

Por motivos de seguridad, las cuentas de los coordinadores debe darlas de alta el administrador del sistema.

Base de datos

Cuando se lanza la aplicación en el servidor abre una conexión con la base de datos MongoDB, esta se realiza a través de la librería Mongoose. En caso de no existir creara una nueva base de datos vacía con el nombre indicado.

Si la conexión con la base de datos es satisfactoria, el sistema creara al usuario administrador en caso de no existir.

A continuación se muestra un ejemplo de cómo se define el esquema de la colección 'Asignatura Origen'.

```
//Esquema
var AsigOrg_Schema = mongoose.Schema({
  codigo: { type: String , required: true },
  nombre: { type: String , required: true},
  curso: { type: String },
  tipo: { type: String }, // {(OB)Obligat, (OP)Optativa, (LE)Libre elecc}
  ects: { type: Number },
  sem: { type: String },
  modulo: { type: String },
  url: { type: String },
  titulacion: {
    type: mongoose.Schema.Types.ObjectId,
    ref: 'Titulacion',
    index: true,
  },
  ofertada: { type: Boolean }
});

AsigOrg_Schema.index({ codigo: 1, titulacion: 1 }, { unique: true });
```

Fragmento de código 1. Esquema asignatura Origen

A través del esquema se pueden especificar las propiedades de los distintos campos, en el ejemplo de la figura anterior, el 'codigo y el 'nombre' son obligatorios y la 'titulación' hace referencia al identificador de un objeto de la colección Titulación.

Los tipos de datos soportados en los esquemas son: String, Number, Date, Buffer, Boolean, Mixed, Objectid y Array. Automáticamente MongoDB asigna a cada documento creado un id numérico, a través del cual se identifica de forma inequívoca.

Cliente

Se ha desarrollado una aplicación single-page (SPA) con el framework AngularJS MVC.

Cuando se accede a la aplicación por primera vez, Angular realiza una primera carga en el navegador del cliente de toda los módulos que forman la aplicación: plantillas HTML, controladores, CSS,... La primera petición tiene un coste temporal alto, sin embargo, al tener toda la lógica de la aplicación y los recursos de la misma ya cargados en el cliente, en las siguientes peticiones al servidor únicamente envía los datos del recurso solicitado, proporcionando un tráfico de red inferior, mayor velocidad de ejecución del sistema y liberando al servidor de carga de trabajo.

Para poder realizar la aplicación SPA se ha hecho uso del módulo de Angular 'UI-Router' que permite implementar el patrón 'Vista Compuesta', donde por ejemplo la barra de navegación representa una vista y el resto de la página otra, actualizándose la información de forma independiente a través de solicitudes asíncronas Ajax al servidor.

Para poder adaptar la aplicación a los distintos tamaños de pantalla se ha utilizado el framework Bootstrap v3.3.7, el cual además de generar un sistema de rejillas para implementar esta funcionalidad, provee de un conjunto de componentes que se han utilizado en el diseño de las interfaces. Pese a que existen algunas directivas creadas en angular para manipular los componentes de Bootstrap, también ha sido necesaria descargar la librería UI-Bootstrap desarrollada por el equipo de Angular que proporciona directivas nativas para manipular cualquier componente, además de añadir alguno al catálogo.

A continuación se muestra como se adapta la pantalla de inicio en función del tamaño de la misma.

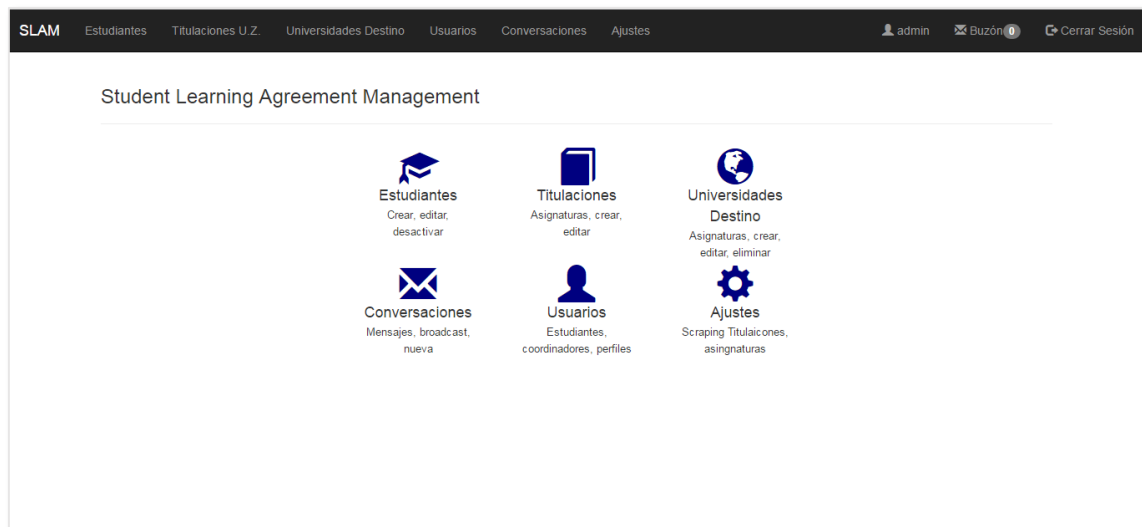


Figura 8. Vista menú principal (pantalla monitor)

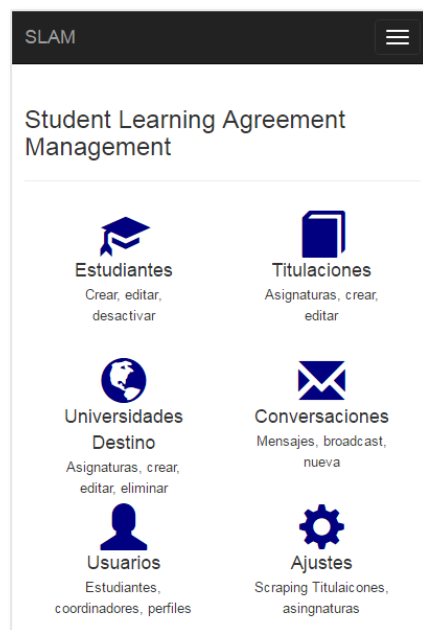


Figura 9. Vista menú principal (pantalla telef. móvil)

Uno de los aspectos fundamentales de la aplicación es el de guiar al estudiante en el proceso de creación de su Acuerdo de Estudios. Para ello se ha tomado la decisión de implementar un asistente que consta de tres ventanas modales en las que se seleccionan las asignaturas de origen y destino y en última instancia se verifica si puede añadir las al Acuerdo y en qué condiciones.

A continuación se van a presentar algunas vistas que forman parte de la interfaz de la aplicación.

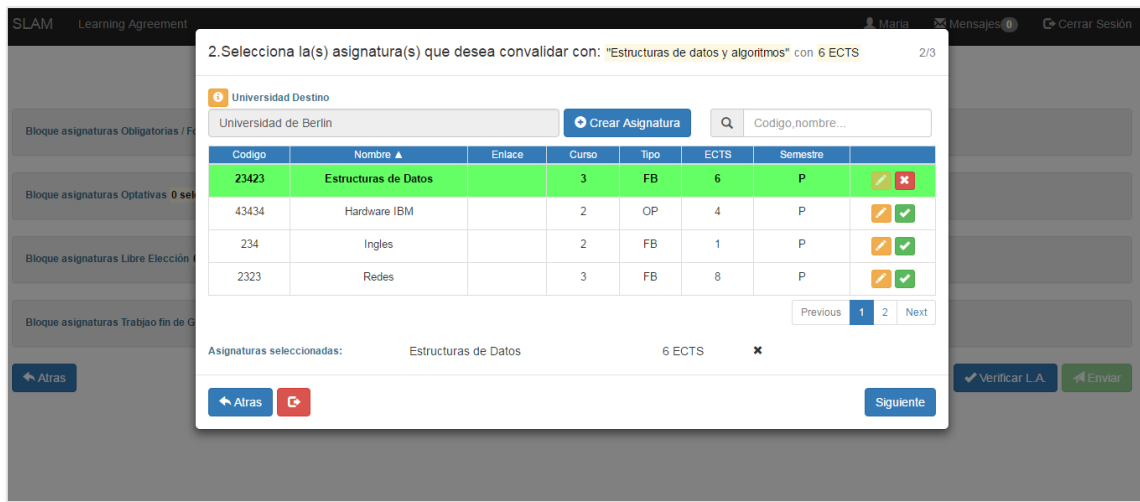


Figura 10. Interfaz asistente selección (2/3)

La figura 10 muestra la interfaz de una de las tres vistas que forma parte del asistente de selección de asignaturas.

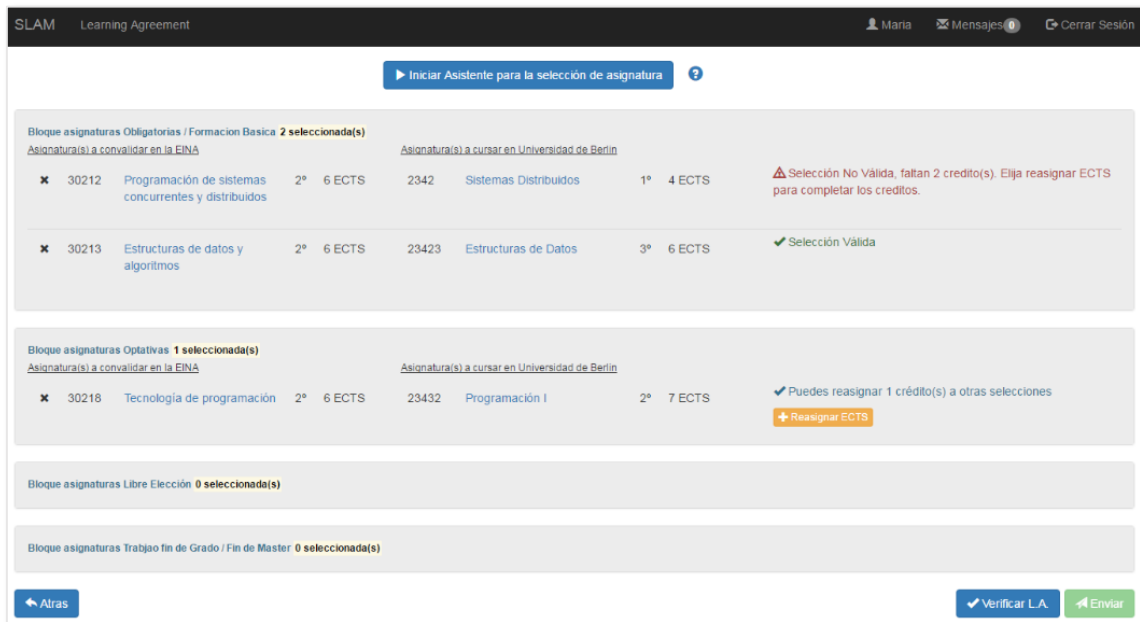


Figura 11. Interfaz creación Acuerdo de Estudios

La figura 11 muestra la interfaz donde se añaden los distintos bloques de asignaturas que forman el Acuerdo, antes de ser enviado para su revisión.

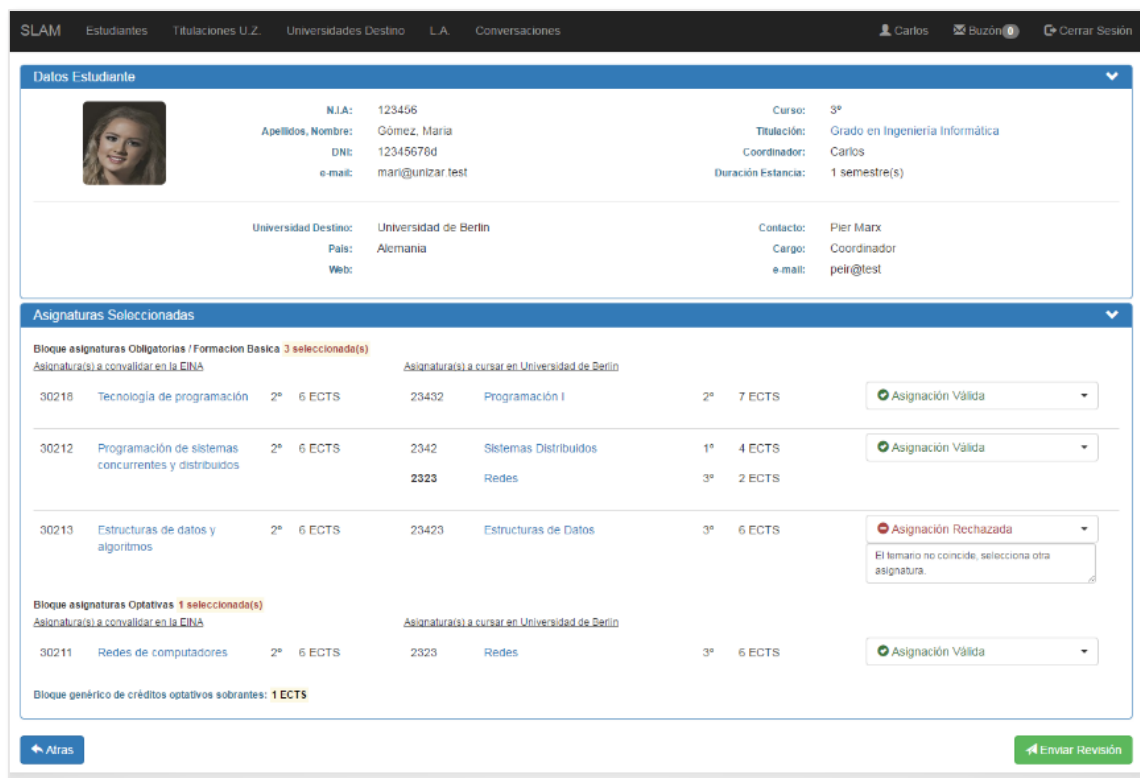


Figura 12. Interfaz revisión Acuerdo de Estudios

En la figura 12 se muestra la interfaz donde el coordinador realiza la revisión del Acuerdo de Estudios, anotando en los bloques rechazados el motivo.

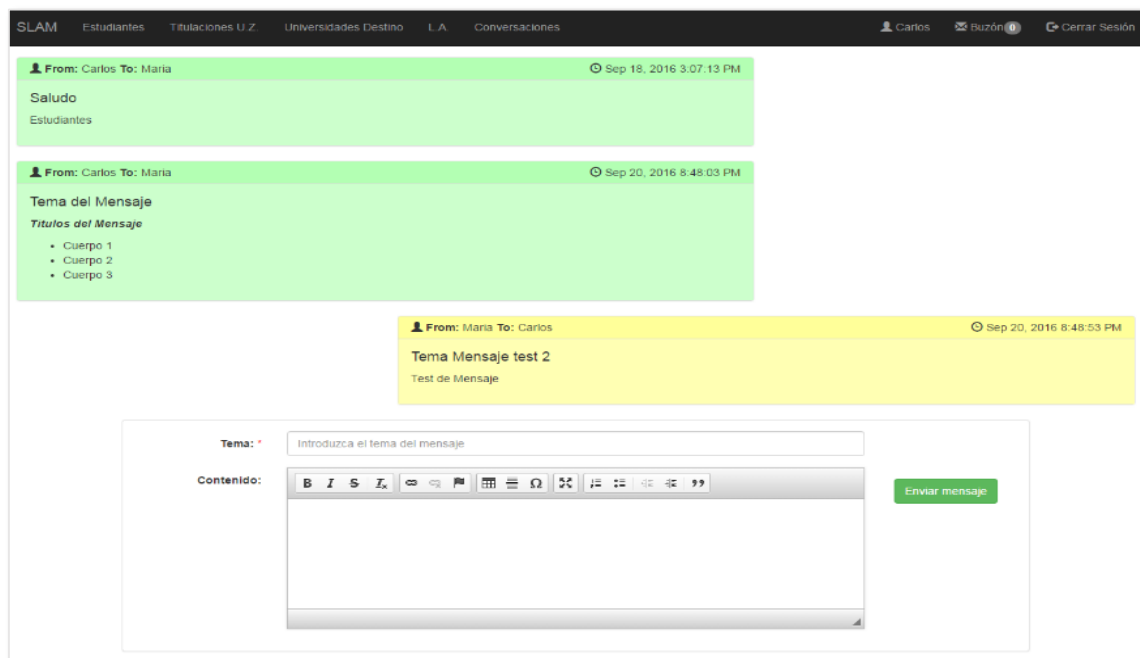


Figura 13. Interfaz de una conversación

En la figura 13 se muestra la interfaz de una conversación entre un estudiante y su coordinador.

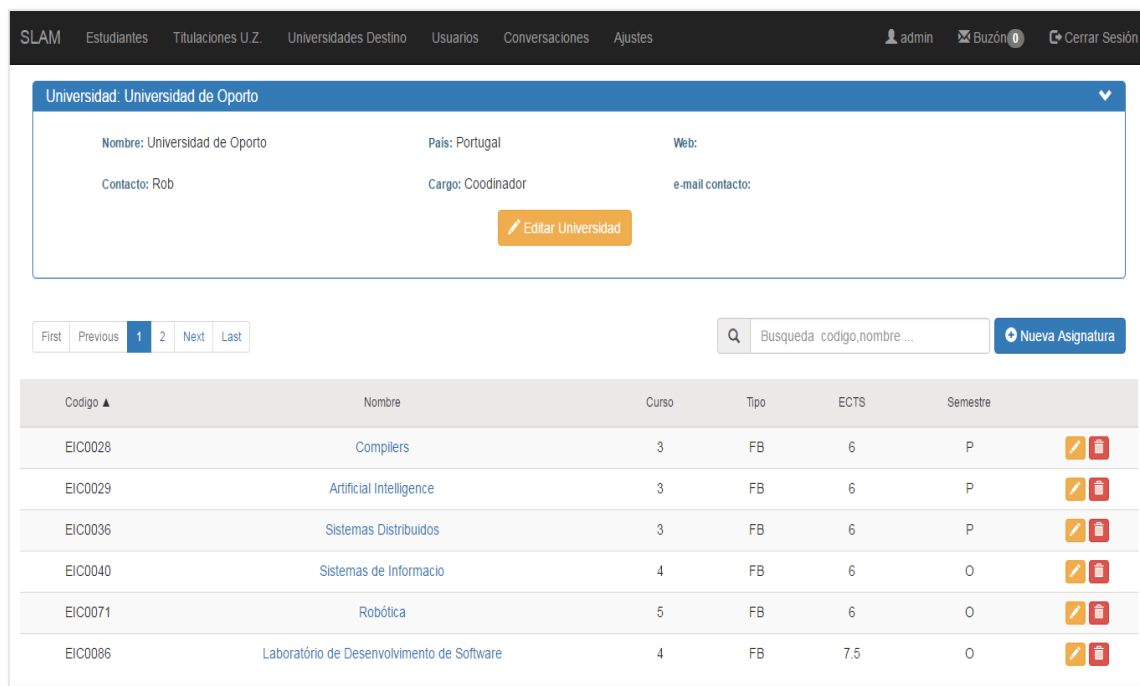


Figura 14. Interfaz gestión universidad destino

En la figura 14 se muestra la interfaz de gestión de una de las universidades de destino.

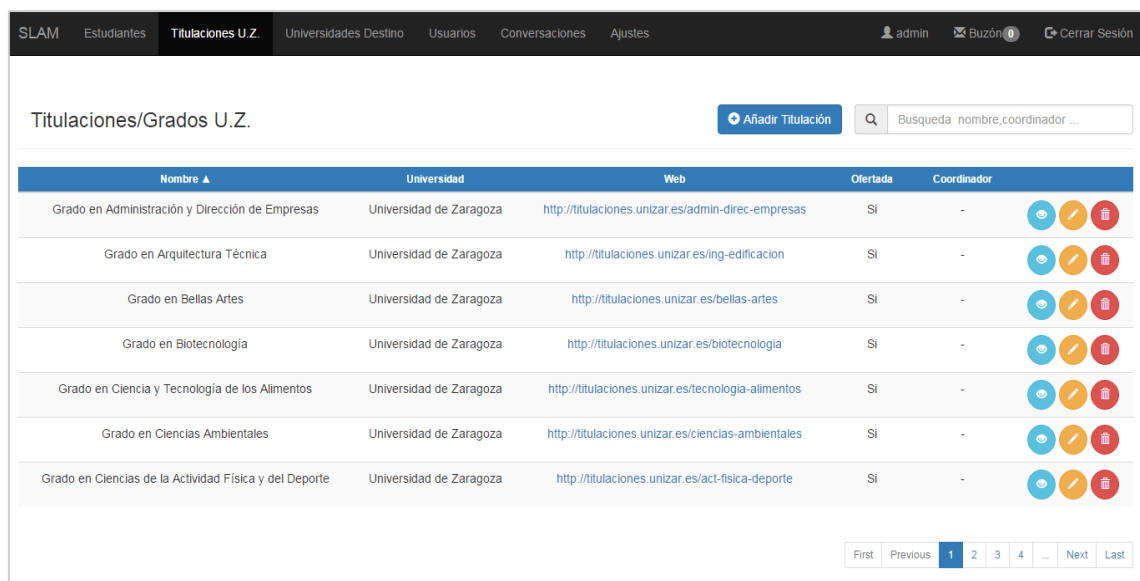


Figura 15. Interfaz listado titulaciones

Para finalizar, en la figura 15 se muestra la interfaz donde se lista la información de las titulaciones de la Universidad de Zaragoza.

4. Evaluación y Pruebas

La construcción de una aplicación que satisfaga las necesidades del cliente y cumpla con sus especificaciones es el principal objetivo de la verificación y validación.

Conforme se ha ido desarrollando el sistema, se han realizado pruebas de validación de carácter técnico (ausencia de errores) y funcionales, simulando un uso cotidiano de la aplicación para detectar fallos producidos por usuarios y asegurarse de que cumple con los requisitos establecidos.

Las pruebas realizadas sobre la API, se han ejecutado a través de los plugins de Google Chrome 'Postman' y 'Yarc'. En la realización de estas pruebas se ha comprobado que los usuarios tienen los privilegios suficientes para acceder a los recursos solicitados y que el funcionamiento de la API es el correcto. También se han realizado pruebas para comprobar el estado de los campos de la base de datos, no permitiendo insertar por ejemplo, usuarios con un email ya registrado.

En la API se ha utilizado la librería 'Winston' de Node.js, la cual registra en ficheros .log las distintas peticiones que llegan al servidor y el tipo de respuesta que generan. En caso de producirse algún error lo registra en un fichero distinto para facilitar su detección.

Una vez que se ha comprobado que la API está disponible, se han ejecutado las pruebas en el lado del cliente, estas se basan en comprobar los errores producidos por el usuario (campos obligatorios vacíos, introducir datos incorrectos...) y en comprobar el correcto funcionamiento de las distintas interfaces donde se ha de manipular la información introducida por el usuario y que genera un resultado. Un ejemplo de ello sería en la creación de los acuerdos de estudio y sus restricciones en la reasignación de créditos, ya que en función de las asignaturas que se han seleccionado se podrá realizar unas acciones u otras.

Para la realización de las pruebas se han utilizado los principales navegadores de escritorio (Mozilla Firefox, Google Chrome, Internet Explorer y Safari) y móviles (Chrome y Mozilla).

En el [anexo 5](#) se han detallado las pruebas realizadas y el resultado obtenido.

5. Conclusiones

5.1. Conclusiones

Una vez se ha comprobado que la aplicación satisface todos los requerimientos establecidos al comienzo del proyecto, se ha dado por concluida y se han obtenido las siguientes conclusiones.

El desarrollo modular tanto en el cliente como en el servidor ha permitido generar un código claro y fácilmente escalable, facilitando el proceso de integración de los distintos componentes que forman la aplicación.

Desarrollar la API con Node.js requiere programar con eficiencia cada una de las funciones que se realizan en el servidor, ya que aunque las operaciones de entrada/salida se ejecuten de forma aislada, las tareas que no lo son y tienen un alto coste operacional es necesario fragmentarlas cuando hagan un uso intensivo del core del procesador. Esto se debe a que si el core está ocupado ejecutando una tarea, cuando llegan otras peticiones al servidor estas se guardan en la cola de eventos que utiliza Node.js y hasta que el procesador no se libere no atenderá a las tareas almacenadas en dicha cola, produciendo un bloqueo en todas las solicitudes. En este sentido la única tarea que ha requerido de ser fragmentada ha sido la carga automática de las titulaciones y asignaturas.

Emplear una base de datos documental cuando existen muchas relaciones entre colecciones puede resultar algo tedioso, ya que al no disponer de funcionalidades nativas para manejar las relaciones, como el 'borrado en cascada' que si ofrecen las bases de datos relacionales, se deben ejecutar manualmente funciones para mantener la consistencia de los datos de las colecciones referenciadas, pese a esto, las prestaciones que ofrece MongoDB cuando se trabaja con una arquitectura MEAN superan a estas desventajas.

Definir al comienzo del proyecto los requisitos de la aplicación de una forma precisa y comprensible es indispensable si se quiere realizar una aplicación de esta envergadura, ya que interfiere directamente en el resto de etapas de desarrollo de la aplicación.

5.2. Trabajo futuro

Uno de los motivos por los que se ha realizado una programación modular tanto en el front-end como en el back-end ha sido para poder escalar las funcionalidades que ofrece la aplicación de una forma sencilla. A continuación se presentan algunas de las mejoras que se podrían llevar a cabo en futuros trabajos:

- Integración de la aplicación con la plataforma Sigma para poder automatizar el envío del Acuerdo de Estudios una vez haya sido aprobado por el coordinador. De esta forma el estudiante no tendrá que cumplimentar de nuevo el formulario oficial de la Universidad de Zaragoza.
- Utilizar el módulo Cluster de Node.js para aprovechar las ventajas de los sistemas multi-core y de esta forma balancear la carga de trabajo en distintos núcleos.

- Crear un módulo de estadísticas donde se almacenen las relaciones entre las asignaturas origen y destino y cuando fuesen a seleccionarlas los alumnos se les recomendará cuáles han elegido otros compañeros.
- Guardar copias de las versiones de los Acuerdos para impedir que el alumno pueda cometer los mismos errores pese a las indicaciones del coordinador.

5.3. Valoración Personal

Tras la realización del proyecto, puedo afirmar que se han alcanzado todas las metas personales que me propuse. Trabajar con tecnologías que desconocía ha sido quizás el mayor reto al que me he enfrentado durante el desarrollo, pero tras el impacto inicial y echando la vista atrás, considero que la elección de las mismas ha sido totalmente acertada, aunque en alguna ocasión pareciese lo contrario.

El diseño de la interfaz ha sido otro punto donde se ha invertido mucho tiempo y esfuerzo, ya que al no existir ninguna aplicación con unas características similares, no he tenido ninguna referencia donde apoyarme más allá de las indicaciones del coordinador. Pese a esto, considero que se ha logrado un alto nivel de usabilidad.

Para finalizar, me gustaría y espero que los coordinadores de movilidad de las distintas titulaciones de la Universidad de Zaragoza acaben utilizando este sistema, ya que sinceramente creo que puede ahorrarles esos factores que hoy en día tanto se valoran, como es el tiempo y el trabajo.

6. Gestión del Proyecto

6.1. Gestión de la configuración

Para la gestión de la configuración del proyecto se ha hecho uso de varias herramientas que han facilitado su realización. Todas las imágenes de diagramas, memoria e información de soporte se han mantenido en Dropbox, ya que es una herramienta ampliamente conocida.

El sistema de control de versiones se ha llevado a cabo a través de la aplicación Git y su repositorio Github donde se ha agregado al tutor del proyecto.

6.2. Planificación

La planificación del proyecto se ha realizado en distintas etapas marcadas por hitos, siguiendo un desarrollo incremental e iterativo.

En la primera etapa del proyecto se han estudiado y analizado las tecnologías implicadas. El resto de etapas están formadas por los módulos de Scraping, autenticación, comunicación y Learning Agreements.

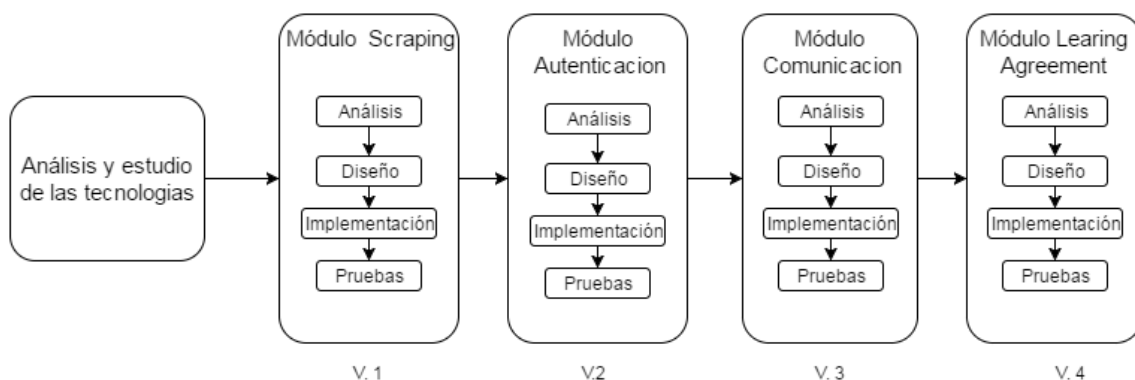


Figura 16. Etapas del desarrollo de la aplicación

Como se puede observar en la figura anterior, cada módulo se compone de una fase de análisis, una fase de diseño, una fase de implementación y por último las pruebas de verificación y validación.

A continuación se va a presentar un diagrama de Gantt donde se expone como se ha dividido temporalmente el desarrollo de las distintas etapas que forman el proyecto.

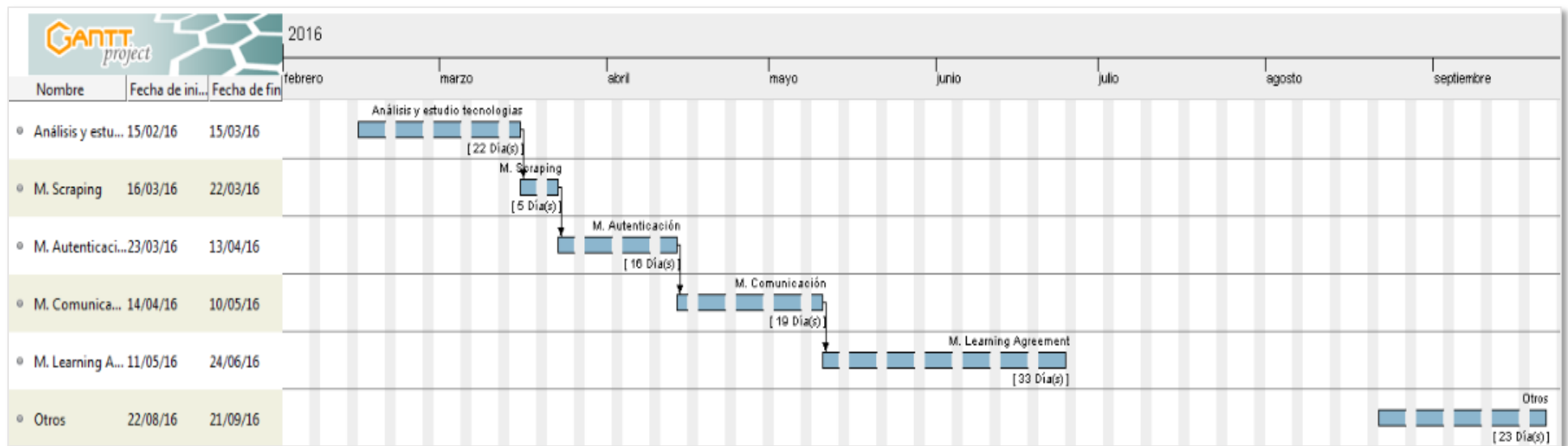


Figura 17. Diagrama Gantt

6.3. Distribución Temporal

Para finalizar, en las siguientes figuras se presenta la distribución temporal que han tenido las etapas del proyecto.

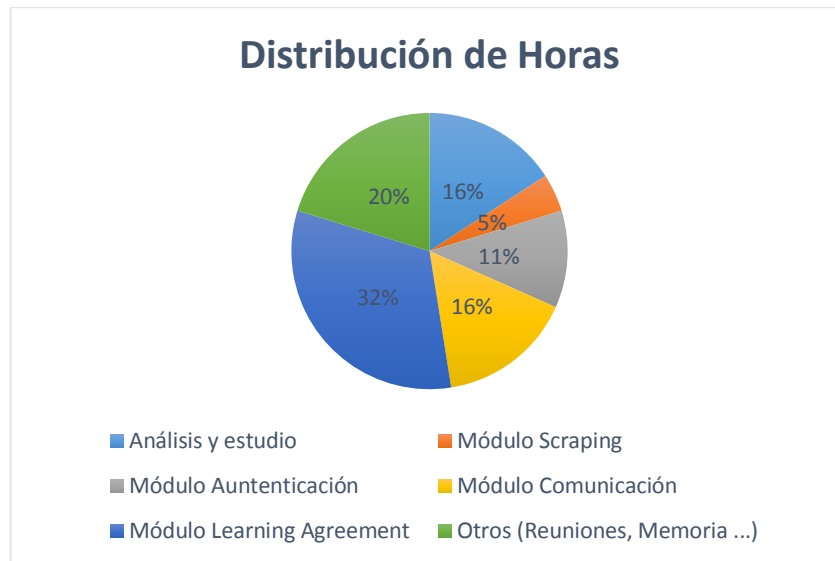


Figura 18. Gráfico distribución de horas

Como se puede observar, el módulo de Learning Agreements es el que más tiempo ha requerido, esto se debe a que el análisis, el diseño y la implementación de sus interfaces es más compleja en comparación con el resto de módulos, además de estar formado por muchas más vistas.

La memoria también ha sido un punto donde se han invertido muchas horas.

El módulo de Scraping es el que menos tiempo ha costado de desarrollar, esto se debe a que la implementación en el cliente es mínima.

Hito	Horas
Análisis y Estudio	82
Módulo Scraping	23
Módulo Autenticación	59
Módulo Comunicación	82
Módulo Learning Agreements	167
Otros	105
Total	518

7. Bibliografía

- Bootstrap. (n.d.). Noviembre de 2016, sitio web: <http://getbootstrap.com/>
- Angular-UI. (n.d.). Noviembre de 2016, sitio web: <https://angular-ui.github.io/>
- MongoDB. (n.d.). Noviembre de 2016, sitio web: <https://docs.mongodb.com>
- AngularJS. (n.d.). 12 de Noviembre de 2016, sitio web: <https://angularjs.org/>
- Manual Angular. (n.d.). Noviembre de 2016, sitio web: <http://www.desarrolloweb.com/>
- Express. (n.d.). Noviembre de 2016, sitio web: <http://expressjs.com/>
- Node.js. (n.d.). Noviembre de 2016, sitio web: <https://nodejs.org/en/>
- Cheerio. (n.d.). Noviembre de 2016, sitio web: <https://cheerio.js.org/>
- JSON Web Tokens. (n.d.). Noviembre de 2016, sitio web: <https://jwt.io/>
- Winston. (n.d.). Noviembre de 2016, sitio web: <https://www.npmjs.com/package/express-winston>
- jsPDF. (n.d.). Noviembre de 2016, sitio web: <https://parall.ax/products/jspdf>
- CKEditor. (n.d.). Noviembre de 2016, sitio web: <http://docs.ckeditor.com/>
- Bower. (n.d.). Noviembre de 2016, sitio web: <https://bower.io/>
- Wikipedia. (n.d.). Noviembre de 2016, sitio web: <https://es.wikipedia.org/>
- HTML Y JavaScript. (n.d.). Noviembre de 2016, sitio web: <http://www.w3schools.com/>
- AngularJS y JavaScript. (n.d.). Noviembre de 2016, sitio web: <https://www.codecademy.com/>

8. Anexos

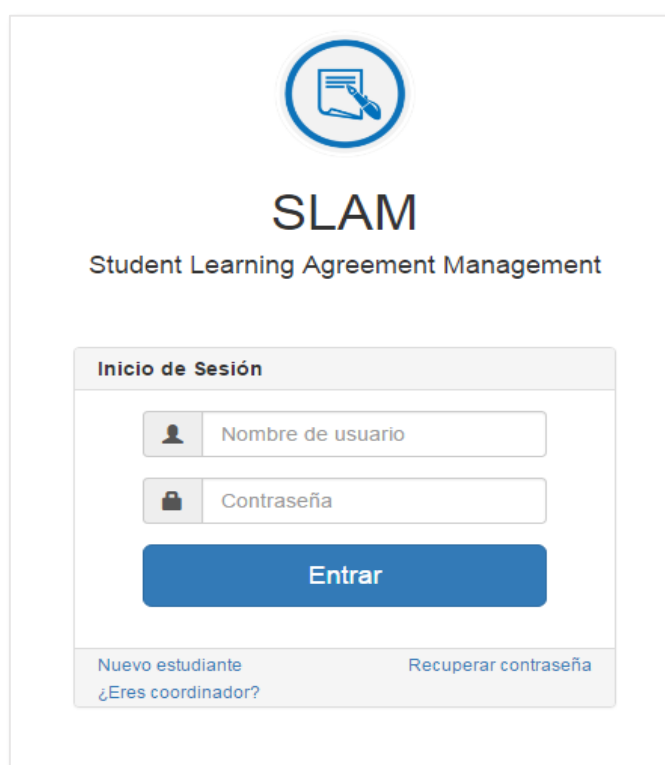
8.1. Anexo I: Manuales de Usuario

A continuación se presentan los distintos manuales de usuario (uno por cada perfil de acceso) de la aplicación.

8.1.1. Manual de usuario Estudiante

- *Acceso mediante Nombre Usuario/Contraseña*

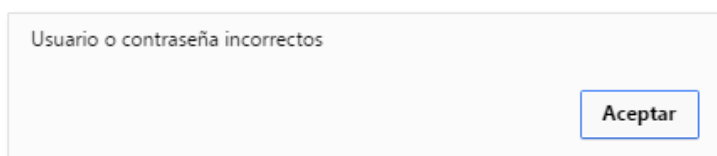
Inicialmente se muestra la pantalla de login, en la que se debe introducir un usuario y contraseña válidos para poder acceder a la pantalla de inicio.



La imagen muestra la interfaz de usuario para el inicio de sesión en SLAM. En la parte superior, hay un icono circular con un documento y un lápiz. Debajo, el título "SLAM" y el subtítulo "Student Learning Agreement Management". El formulario de inicio de sesión está titulado "Inicio de Sesión" y contiene dos campos de entrada: "Nombre de usuario" con un ícono de persona y "Contraseña" con un ícono de candado. Debajo de los campos hay un botón azul "Entrar". En la parte inferior del formulario, hay dos enlaces: "Nuevo estudiante ¿Eres coordinador?" y "Recuperar contraseña".

Figura 19. Pantalla login

En caso de producirse un error se mostrara un mensaje informativo y el estudiante podrá volver a introducir las credenciales.



La imagen muestra un mensaje de error de autenticación. El mensaje dice "Usuario o contraseña incorrectos" y hay un botón "Aceptar" a la derecha.

Figura 20. Error autenticación

- *Restablecer Contraseña*

El estudiante puede restablecer la contraseña insertando su nombre de usuario y su dirección de e-mail. El sistema le enviará las nuevas credenciales de acceso a la dirección facilitada.

The screenshot shows a web form titled "Recuperación de Contraseña". At the top, it says "Recupere su contraseña completando los datos del formulario". There are two input fields: "Nombre Usuario (login):" with the placeholder "Introduzca su nombre de acceso a la aplicación" and "Dirección de correo:" with the placeholder "Introduzca el e-mail". Below these is a reCAPTCHA widget with the text "No soy un robot" and a checkbox. At the bottom, there are two buttons: "Atras" (blue) and "Enviar" (green). A small note at the bottom states: "* La nueva contraseña se generará automáticamente en el servidor y será enviada a su dirección de correo."

Figura 21. Pantalla recuperación contraseña

- *Registro de nuevo estudiante*

Para registrarse en la aplicación el estudiante deberá cumplimentar el formulario de registro. En caso de olvidar algún campo obligatorio o con formato incorrecto el sistema le informara del error.

The screenshot shows a web form titled "Registro de nuevo estudiante". At the top, it says "Cree su cuenta completando los datos del formulario". The form contains several fields: "Nombre Usuario (login):*" with placeholder "Introduzca un nombre de acceso a la aplicación" and a note "Cadena alfanumerica sin espacios con un tamaño máximo de 10 caract."; "NIA: *" with placeholder "Introduzca el NIA"; "Apellido(s):*" with placeholder "Introduzca los apellidos"; "Nombre: *" with placeholder "Introduzca el nombre"; "NIF: *" with placeholder "Introduzca el NIF"; "Curso: *" with a dropdown menu "Seleccione un cui"; "Imagen de perfil:" with a file upload button; "E-mail: *" with placeholder "Introduzca el e-mail"; and "Titulación: *" with a dropdown menu "Seleccione una titulación". At the bottom, there is a reCAPTCHA widget with "No soy un robot" and a checkbox. A note at the bottom states: "* La contraseña se generará automáticamente en el servidor y será enviada a la dirección de correo electrónico facilitada". There are two buttons at the bottom: "Atras" (blue) and "Crear Cuenta" (green).

Figura 22. Registro de estudiante

En caso de que el 'Nombre de Usuario', el 'NIA' o el 'E-mail' estén ya registrados en la aplicación por otro estudiante se mostrara un mensaje de error. El sistema le enviará las credenciales de acceso al e-mail facilitado.

El estudiante solo tendrá disponible su titulación si existe un coordinador registrado que la gestione.

- *Inicio*

En la pantalla de inicio se muestran las principales funcionalidades de la aplicación.

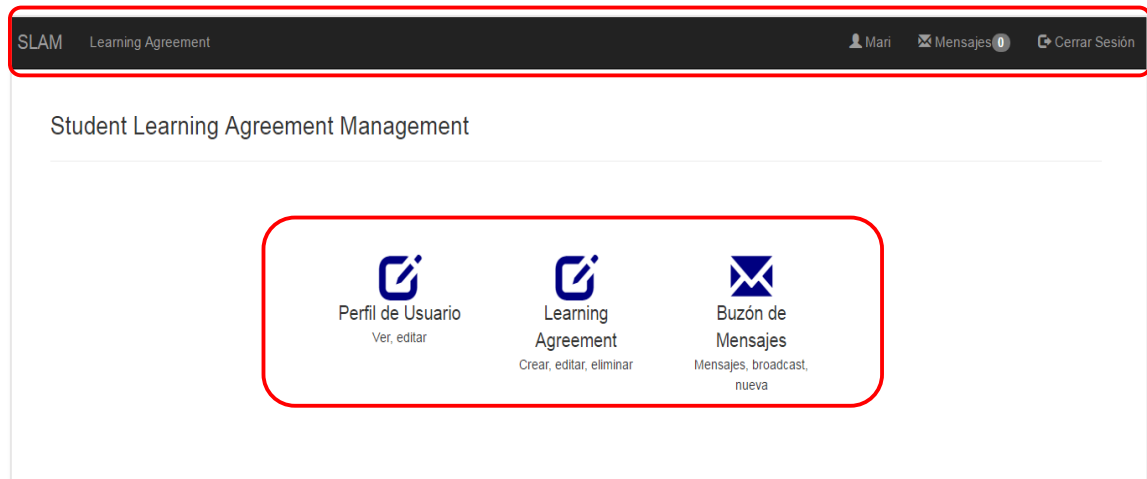


Figura 23. Pantalla de inicio

A continuación, se definen los recuadros señalados en la figura anterior:

- Barra de navegación, compuesta por los botones:
 - SLAM: enlace a la pantalla principal.
 - Learning Agreement: enlace al Learning Agreement.
 - Perfil: enlace al perfil de estudiante.
 - Mensajes: enlace a la conversación con el coordinador.
 - Número de mensajes pendientes de lectura.
 - Cerrar Sesión: cierra la sesión del usuario y sale del sistema.
- Botones principales
 - Perfil de Usuario: enlace al perfil de Estudiante.
 - Learning Agreement: enlace al Learning Agreement.
 - Buzón de Mensajes: enlace a la conversación con el coordinador.

- *Perfil del estudiante*

Pantalla en la que se visualiza la información asociada al estudiante.

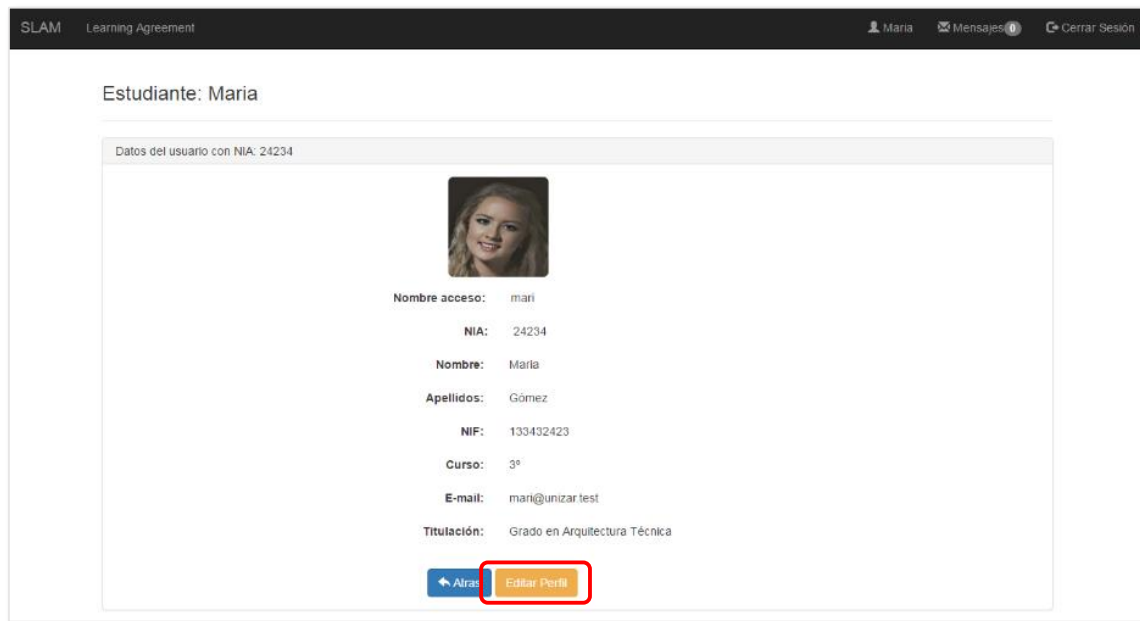


Figura 24. Perfil de estudiante

Si pulsa sobre el botón “Editar Perfil” accederá a la pantalla de edición de perfil del estudiante.

- *Edición de Perfil*

Pantalla en la que se muestra un formulario donde se pueden editar los datos asociados al estudiante. Una vez han sido modificados solo es necesario pulsar sobre el botón “Actualizar” para guardar los cambios.

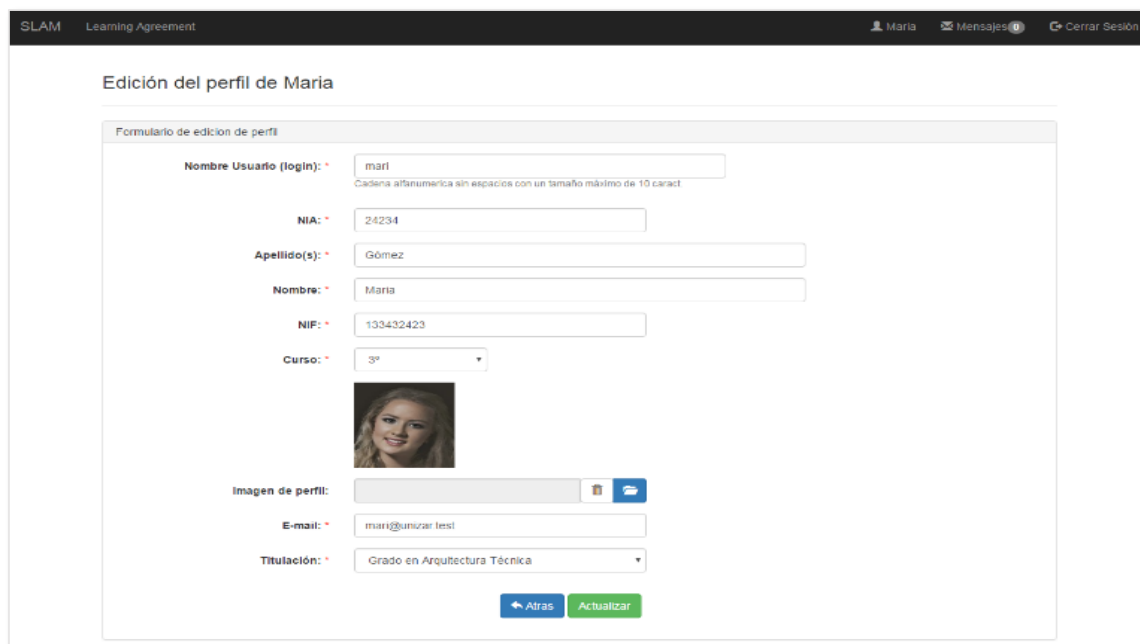


Figura 25. Pantalla de edición de perfil

Si algún campo obligatorio está vacío o no contiene el formato adecuado el sistema mostrara un mensaje informando del error.

- *Etapa 1: Creación de Learning Agreement (Pantalla 1/2)*

En la primera pantalla que forma la gestión del Learning Agreement, el estudiante debe seleccionar el periodo de intercambio (por defecto 'un semestre') y la universidad de destino.

Figura 26. Pantalla creación acuerdo de estudios (1/2)

En caso de que la universidad de destino no se encuentre registrada en la aplicación, el estudiante puede crear una nueva pulsando sobre el botón 'Registrar nueva Universidad'. Una vez se haya seleccionado la universidad el botón de 'Continuar' se habilitara.

- *Creación de Universidad de Destino*

En esta pantalla se registra una nueva universidad de destino

Figura 27. Creación universidad Destino

En caso de que el nombre de la universidad ya este registrado en la aplicación se mostrará el siguiente mensaje de error.

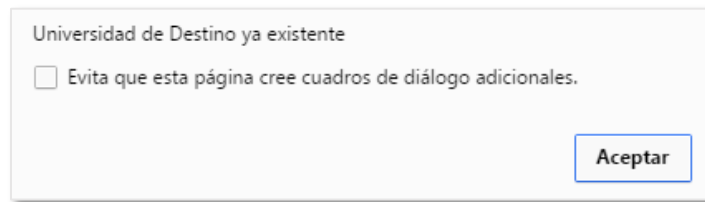


Figura 28. Mensaje error Universidad

- *Creación de Learning Agreement (Pantalla 2/2)*

En esta segunda pantalla es donde se añaden todas las asignaturas que se quieren cursar en el extranjero y aquellas con las que se quieren convalidar en la Universidad de Zaragoza.

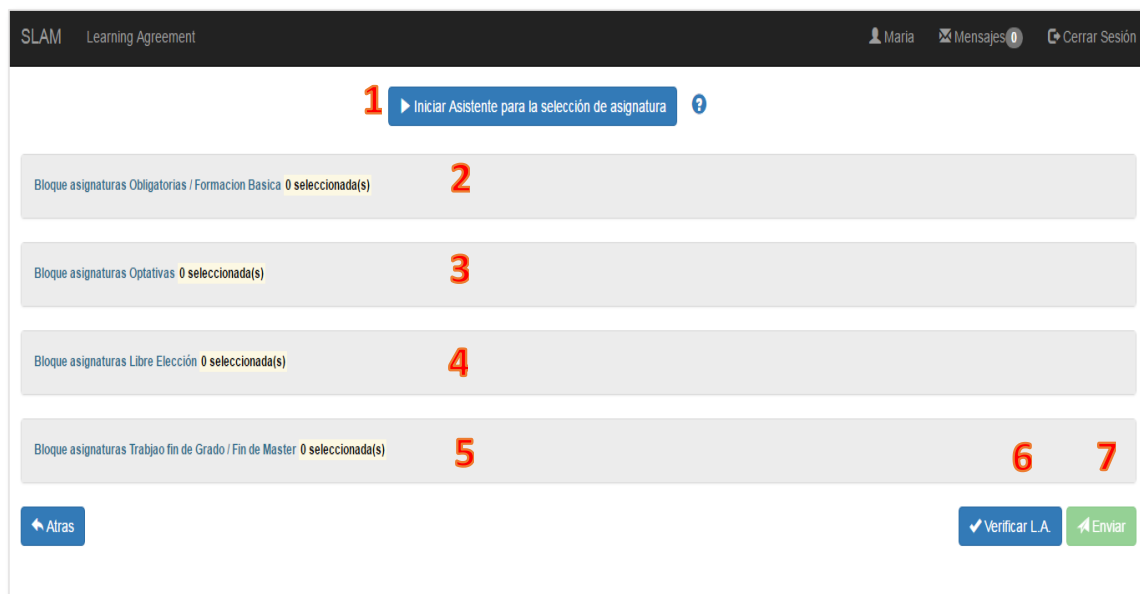


Figura 29. Pantalla creación acuerdo de estudios (2/2)

Funcionalidades:

- 1: inicio del asistente para la selección de asignaturas
- 2, 3, 4, 5: bloques donde se insertaran las asignaturas seleccionadas
- 6: verificación del Acuerdo de Estudios
- 7: envío del Acuerdo al coordinador

- *Asistente selección de asignaturas (1/3)*

A continuación se puede observar cuales son las combinaciones correctas de selecciones:

EQUIVALENCIAS VÁLIDAS			
Tipo de equivalencia de uno a uno			
ASIGNATURAS AQUÍ	ECTS	ECTS	ASIGNATURAS ALLÍ
asignatura A	6	6	asignatura X
Tipo de equivalencia de uno a Varios			
ASIGNATURAS AQUÍ	ECTS	ECTS	ASIGNATURAS ALLÍ
asignatura A	12	6	asignatura X
		3	asignatura Y
		3	asignatura Z
asignatura A	6	12	asignatura X
asignatura B	3		
asignatura C	3		

EQUIVALENCIAS NO VÁLIDAS			
Tipo de equivalencia de VARIOS a VARIOS (es imposible evaluar las asignaturas de aquí)			
ASIGNATURAS AQUÍ	ECTS	ECTS	ASIGNATURAS ALLÍ
asignatura A	12	6	asignatura X
asignatura B	3	6	asignatura Y
		3	asignatura Z

En la figura 30 se muestra la interfaz para seleccionar la asignatura(s) que se desea convalidar de la universidad de Zaragoza.



Figura 30. Interfaz asistente (1/3)

Por defecto se cargan las asignaturas de la titulación del estudiante, pero se pueden seleccionar otras. Para poder avanzar a la siguiente pantalla es necesario seleccionar por lo menos una asignatura.

En caso de que las asignaturas de destino no superen el número de créditos de la asignatura de origen, se pueden reasignar créditos sobrantes de otros bloques en la pantalla de creación del acuerdo (figura 29), una vez se haya insertado el nuevo bloque.

- *Asistente selección de asignaturas (2/3)*

En esta pantalla el estudiante puede seleccionar la(s) asignaturas que quiere realizar en la universidad de destino, en relación a la asignatura(s) seleccionada en la pantalla del asistente (1/3) (figura 30).

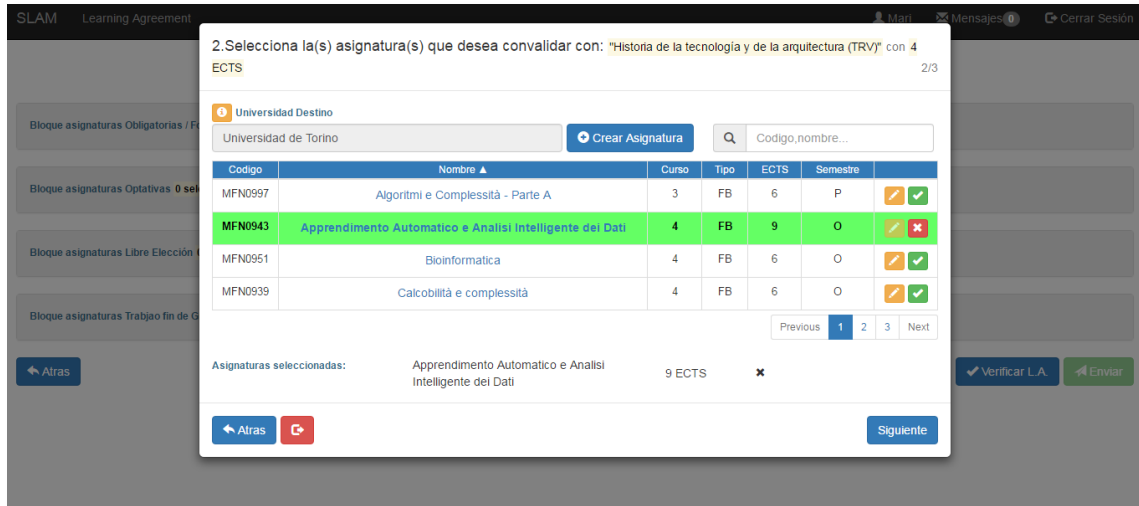


Figura 31. Interfaz asistente (2/3)

Las asignaturas pueden ser editadas siempre y cuando el estudiante sea el que las haya creado y no otro.

En caso de que la asignatura no esté registrada en el sistema puede crearla pulsando sobre el botón 'Crear Asignatura'.

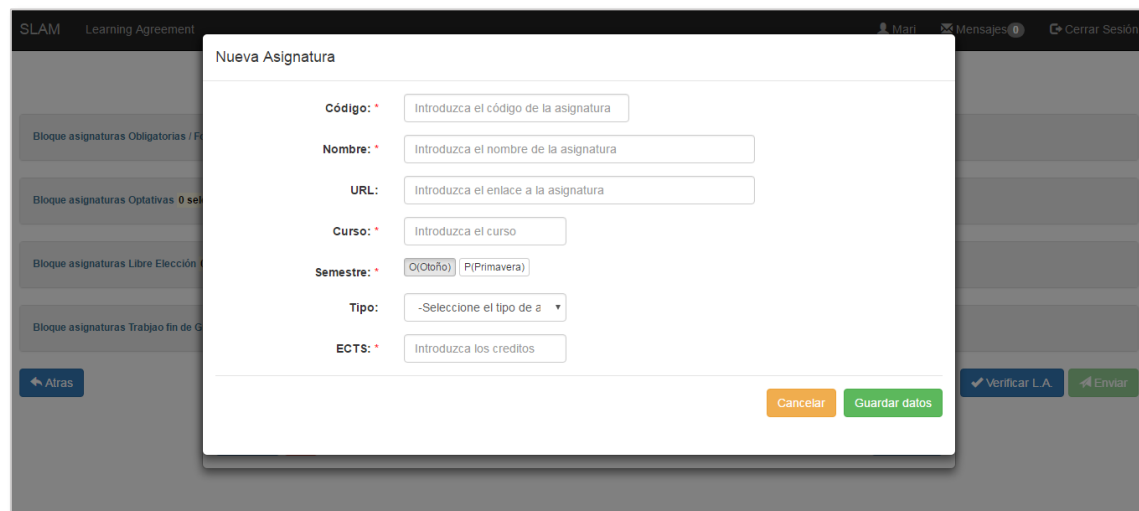


Figura 32. Interfaz crear asignatura destino

Si el código de la asignatura creada ya se encuentra registrada en la universidad, el sistema devolverá un error.

- *Asistente selección de asignaturas (3/3)*

En la última pantalla del asistente se muestra un resumen de las asignaturas seleccionadas.

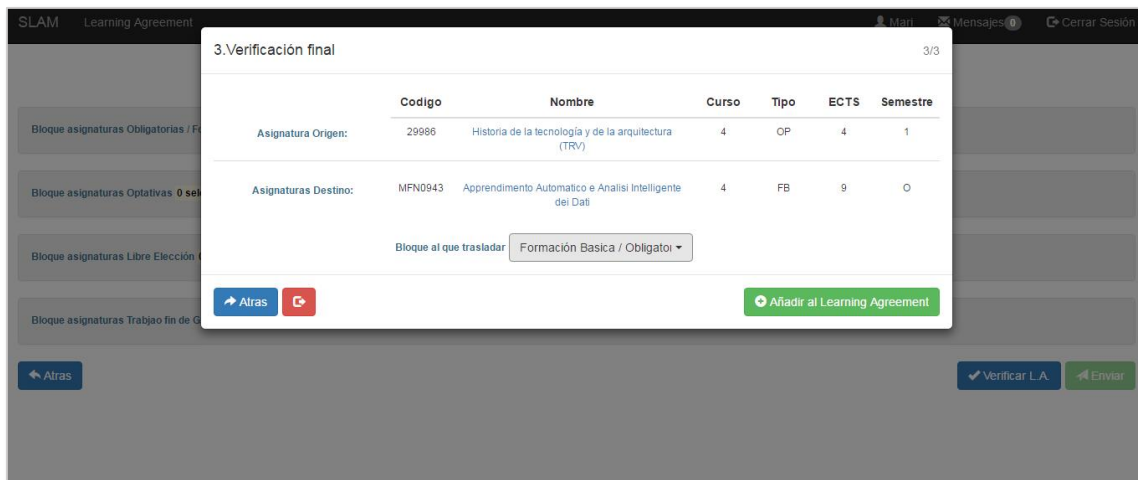


Figura 33. Pantalla asistente (3/3)

Para poder añadir la selección a el acuerdo de estudios es necesario indicar el bloque al que se quiere enviar (Formación Básica/ Libre Elec....).

En caso de que el número de créditos de las asignaturas destino sea inferior a las de las asignaturas de origen se mostrara el siguiente mensaje informativo.

ⓘ ¡Atención! El número de créditos de la asignatura a convalidar en la EINA debe ser superior a la suma de los créditos de las asignaturas a cursar en el extranjero. Puedes volver a atras y añadir otras asignaturas 'Destino' o en el Learning Agreement deberas añadir 1 creditos sobrantes de otros emparejamientos para poder realizar la validación.

Figura 34. Mensaje información

- *Reasignación de créditos*

Existen tres situaciones que se pueden dar cuando se añade una selección de asignaturas a un bloque. En la siguiente figura se puede observar un ejemplo:



Figura 35. Pantalla acuerdo de estudios

Como se puede observar se pueden presentar las siguientes situaciones:

1. Los créditos de origen y destino son iguales: Selección válida
2. Los créditos de origen son menores que los créditos de destino: Selección no válida.
3. Los créditos de origen son mayores que los créditos de destino: Selección válida y con posibilidad de reasignar créditos.

Si se pulsa sobre reasignar créditos aparecerá la siguiente pantalla:

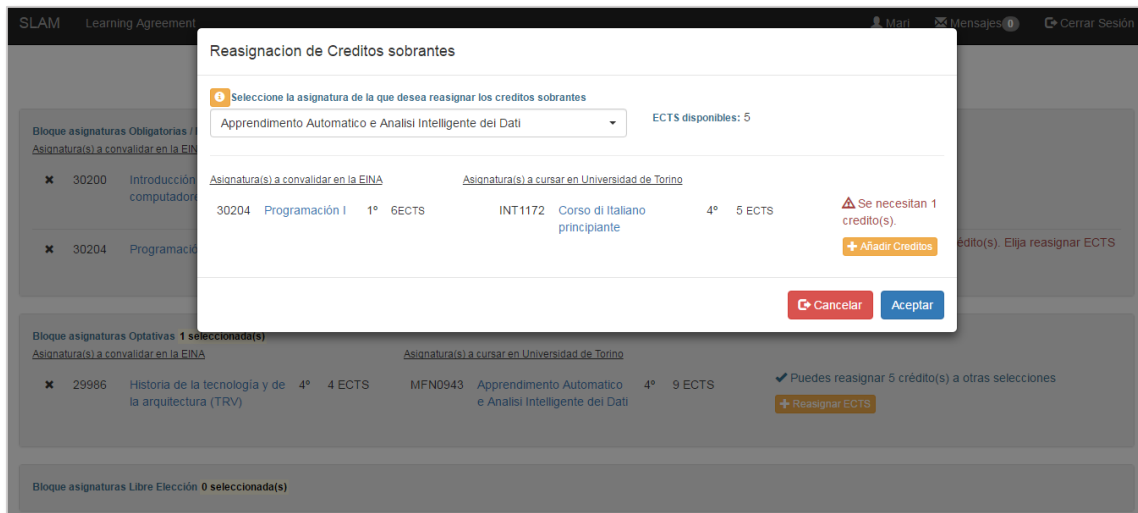


Figura 36. Reasignación de créditos

En figura anterior se muestran las asignaturas con déficit de créditos en las cuales se pueden reasignar. Para añadir los créditos se debe pulsar sobre el botón 'Añadir créditos' y seleccionar el número.

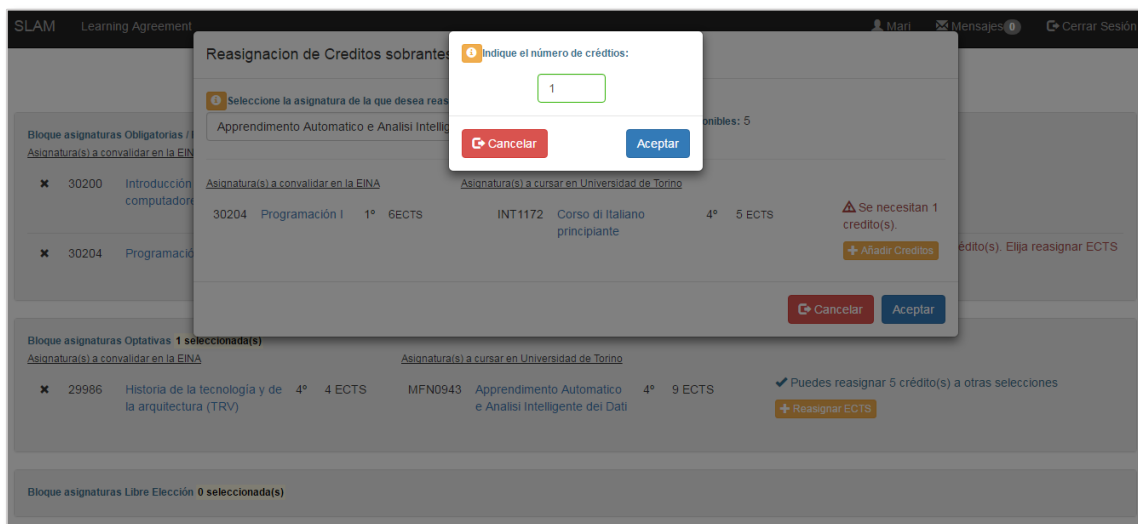


Figura 37. Pantalla selección del número de créditos

En caso de que se inserte un número no válido el sistema informara al estudiante del error.

Una vez se ha seleccionado el número de créditos automáticamente aparecerá la asignatura añadida al bloque.

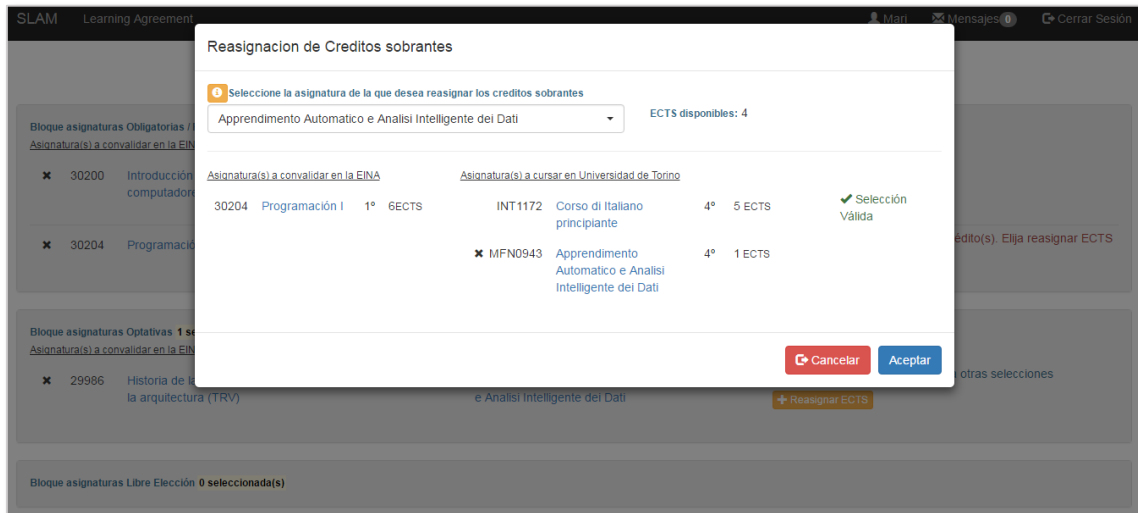


Figura 38. Pantalla con créditos de asignatura reasignada

Si se pulsa aceptar se reasignaran los nuevos créditos y el acuerdo de estudios actualizara su contenido automáticamente.

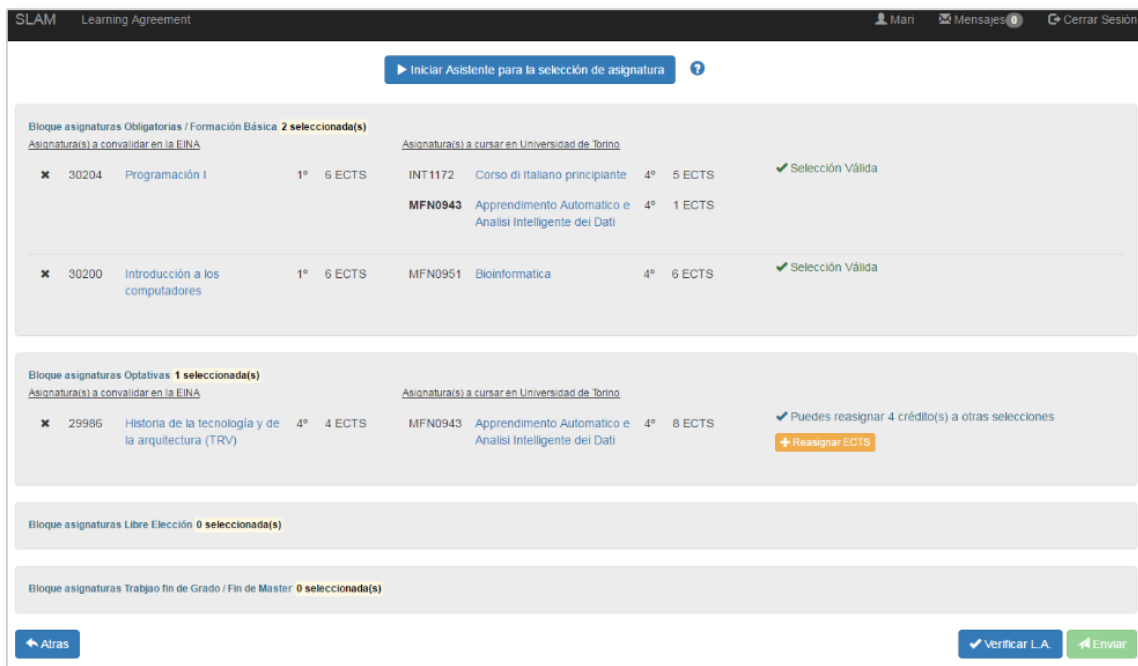


Figura 39. Acuerdo de estudios correcto

- *Envío del Learning Agreement*

Para poder enviar el acuerdo de estudios y que sea revisado por el coordinador, es necesario que sea verificado. Para ello, todas las selecciones deben ser correctas. En caso contrario se mostrara un mensaje de error.

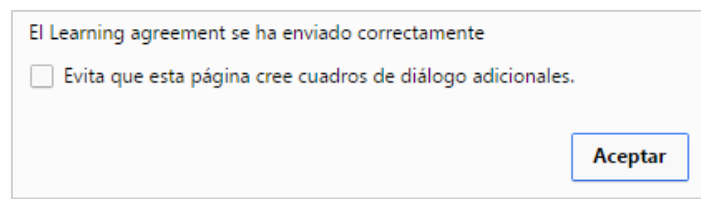
✘ El Learning Agreement contiene uno o mas bloques de asignaciones incorrectos (déficit de créditos). Por favor revise los bloques para que pueda ser verificado correctamente.

Figura 40. Error verificación de acuerdo

En caso de que el acuerdo sea válido

1. Se mostrara un mensaje de información.
2. Se creara un bloque genérico de créditos optativos sobrantes (en el caso de existan). Si se desea eliminar este bloque solo es necesario pulsar 'x'.
3. Se habilitara el botón de envío.

En caso de enviar el acuerdo, se mostrara el mensaje de la figura siguiente:



El Learning agreement se ha enviado correctamente

Evita que esta página cree cuadros de diálogo adicionales.

Aceptar

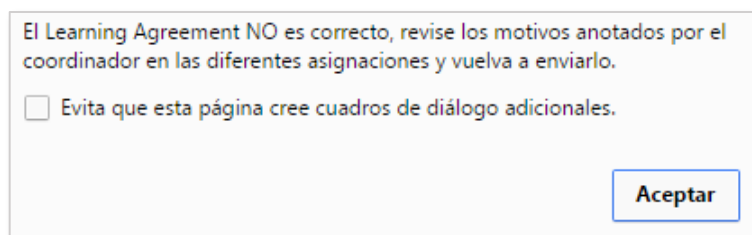
Figura 41. Mensaje envío correcto

- *Etapa 2: Modificación o Aprobación del Learning Agreement*

Una vez el coordinador ha realizado la revisión, se comunica al estudiante a través del e-mail que su acuerdo ha sido aprobado o rechazado.

Acuerdo de estudios rechazado:

En caso de que el coordinador lo haya rechazado cuando se acceda al acuerdo aparecerá el siguiente mensaje:



El Learning Agreement NO es correcto, revise los motivos anotados por el coordinador en las diferentes asignaciones y vuelva a enviarlo.

Evita que esta página cree cuadros de diálogo adicionales.

Aceptar

Figura 42. Notificación L.A. no válido

Una vez dentro del acuerdo, se podrá visualizar los mensajes adjuntados a cada selección indicado el motivo del rechazo.

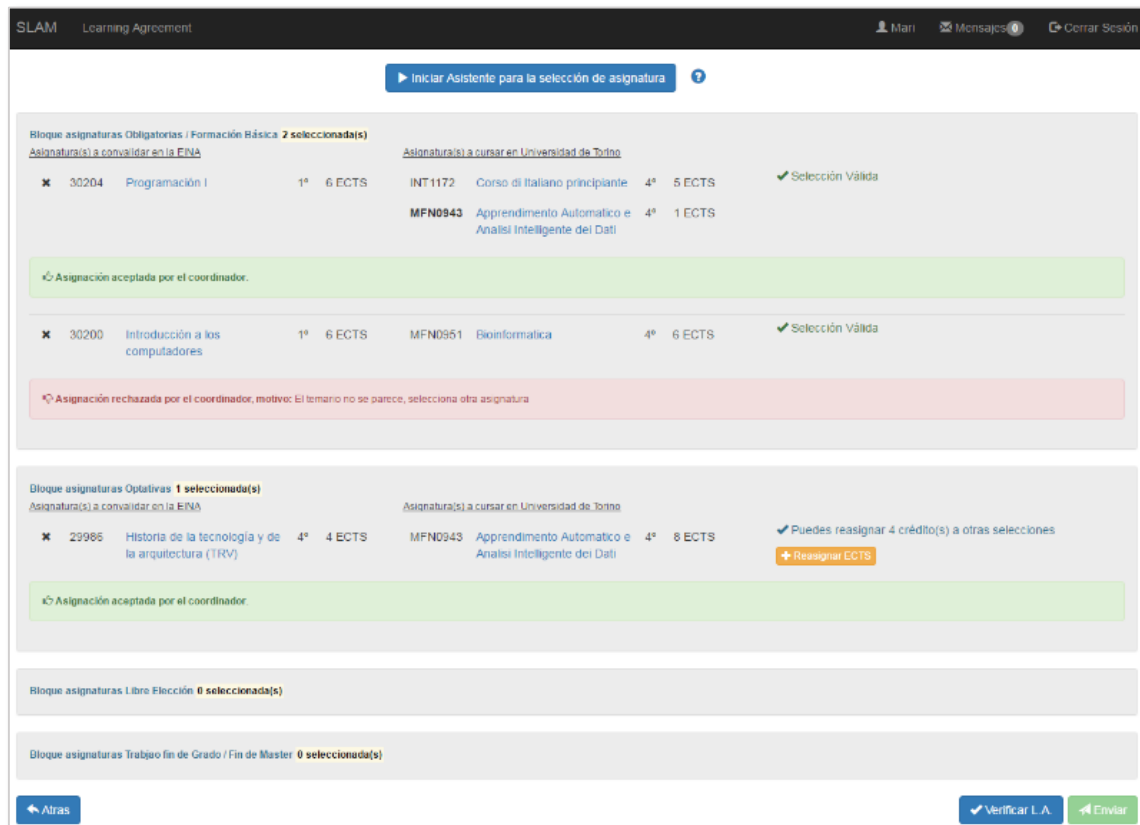


Figura 43. Learning Agreement rechazado

Para poder reenviarlo deberá corregir el bloque rechazado según las indicaciones.

Acuerdo de estudios aprobado:

En caso de que el acuerdo haya sido aprobado por el coordinador cuando acceda a visualizarlo se mostrara el siguiente mensaje:

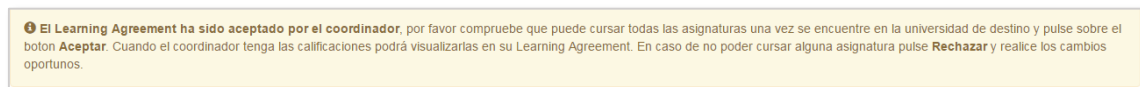



Figura 44. Acuerdo aprobado

Una vez que el estudiante se encuentre en la universidad de destino debe confirmar o rechazar el acuerdo.

SLAM Learning Agreement Man Mensajes Cerrar Sesión

El Learning Agreement ha sido aceptado por el coordinador. por favor compruebe que puede cursar todas las asignaturas una vez se encuentre en la universidad de destino y pulse sobre el botón **Aceptar**. Cuando el coordinador tenga las calificaciones podrá visualizarlas en su Learning Agreement. En caso de no poder cursar alguna asignatura pulse **Rechazar** y realice los cambios oportunos.

Datos Learning Agreement

	N.I.A: 12344	Curso: 1º
Apellidos, Nombre: Cambrá, Mari	DNI: 12345678d	Titulación: Grado en Ingeniería Informática
e-mail: mari@test	Coordinador: Manuel	Duración Estancia: 1 semestre(s)
Universidad Destino: Universidad de Torino	Pais: Italia	Contacto: Cargos: e-mail:

Bloque asignaturas Obligatorias / Formación Basica 2 seleccionada(s)

Asignatura(s) a convalidar en la EINA				Asignatura(s) a cursar en Universidad de Torino			
30200	Introducción a los computadores	1º	6 ECTS	MFN0951	Bioinformatica	4º	6 ECTS
30204	Programación I	1º	6 ECTS	INT1172	Curso di Italiano principiante	4º	5 ECTS
				MFN0943	Apprendimento Automatico e Analisi Intelligente dei Dati	4º	1 ECTS

Bloque asignaturas Optativas 1 seleccionada(s)

Asignatura(s) a convalidar en la EINA				Asignatura(s) a cursar en Universidad de Torino			
29986	Historia de la tecnología y de la arquitectura (TRV)	4º	4 ECTS	MFN0943	Apprendimento Automatico e Analisi Intelligente dei Dati	4º	6 ECTS

Bloque genérico de créditos optativos sobrantes: **4 ECTS**

Rechazar Aceptar Descargar

Figura 45. Confirmación acuerdo

En caso de rechazarlo, deberá realizar las modificaciones oportunas volviendo a la etapa 1 de creación de acuerdo de estudios. Por el contrario, si confirma el acuerdo pulsando sobre el botón 'aceptar', se le notificara al coordinador, a través de un e-mail generado por el sistema, que puede insertar las calificaciones de las asignaturas cuando disponga de ellas.

Si se pulsa sobre el botón de 'descargar', automáticamente se descargará una copia del Learning Agreement aprobado por el coordinador en formato PDF.


- *Etapa 3: Finalización Learning Agreement*

Cuando el estudiante recibe un e-mail notificándole que ya tiene disponibles las calificaciones, si accede a su acuerdo de estudios se mostrara una figura similar a la siguiente.

SLAM Learning Agreement Mari Mensajes 0 Cerrar Sesión

El Learning Agreement esta cerrado. Puede visualizar las calificaciones obtenidas en la universidad de destino y sus correspondientes convalidaciones en la Universidad de Zaragoza.

Datos Learning Agreement

	N.I.A: 12344	Curso: 1º
Apellidos, Nombre: Cambra, Mari	Titulación: Grado en Ingeniería Informática	Coordinador: Manuel
DNI: 12345678d	Duración Estancia: 1 semestre(s)	
e-mail: mari@test		

Universidad Destino: Universidad de Torino	Contacto:
País: Italia	Cargo:
Web:	e-mail:

Bloque asignaturas Obligatorias / Formación Basica 2 seleccionada(s)

Asignatura a convalidar en la EINA	NOTA	Asignatura(s) a cursar en Universidad de Torino	NOTA
30204 Programación I	5	INT1172 Corso di Italiano principiante	6
		MFN0943 Apprendimento Automatico e Analisi Intelligente dei Dati	7

Bloque asignaturas Opcionales 1 seleccionada(s)

Asignatura a convalidar en la EINA	NOTA	Asignatura(s) a cursar en Universidad de Torino	NOTA
30200 Introducción a los computadores	3	MFN0951 Bioinformatica	3

Bloque asignaturas Opcionales 1 seleccionada(s)

Asignatura a convalidar en la EINA	NOTA	Asignatura(s) a cursar en Universidad de Torino	NOTA
29986 Historia de la tecnología y de la arquitectura (TRV)	7	MFN0943 Apprendimento Automatico e Analisi Intelligente dei Dati	7

Bloque genérico de créditos optativos sobrantes: 4 ECTS

[Descargar](#)

Figura 46. Calificaciones Learning Agreement

Una vez que el estudiante puede visualizar las calificaciones el acuerdo de estudios se da por finalizado, dando la opción de descargar en formato PDF una copia del mismo.

- *Buzón de mensajes*

Cuando el estudiante accede a su 'Buzón de mensajes' se visualiza la siguiente figura:

SLAM Learning Agreement Mari Mensajes 0 Cerrar Sesión

From: Mari To: Manuel Sep 22, 2016 12:01:14 AM

Nuevo Registro
Desde este buzón de mensajes podran comunicarse

From: Manuel To: Mari Sep 22, 2016 2:17:20 AM

Saludo Mari
Bienvenida a la aplicación
En caso de tener alguna duda puedes consultarme a través de este buzón.

Tema: * Introdúzca el tema del mensaje

Contenido: [Enviar mensaje](#)

(Rich text editor toolbar with icons for Bold, Italic, Underline, Text Color, Background Color, Bulleted List, Numbered List, Link, Unlink, Undo, Redo, etc.)

Figura 47. Buzón mensajes

El mensaje de 'Nuevo registro' es generado por el sistema y enviado al coordinador cuando el estudiante se da de alta. A través de este buzón se pueden preguntar dudas al coordinador de movilidad, insertando el tema y contenido en el editor de envío.

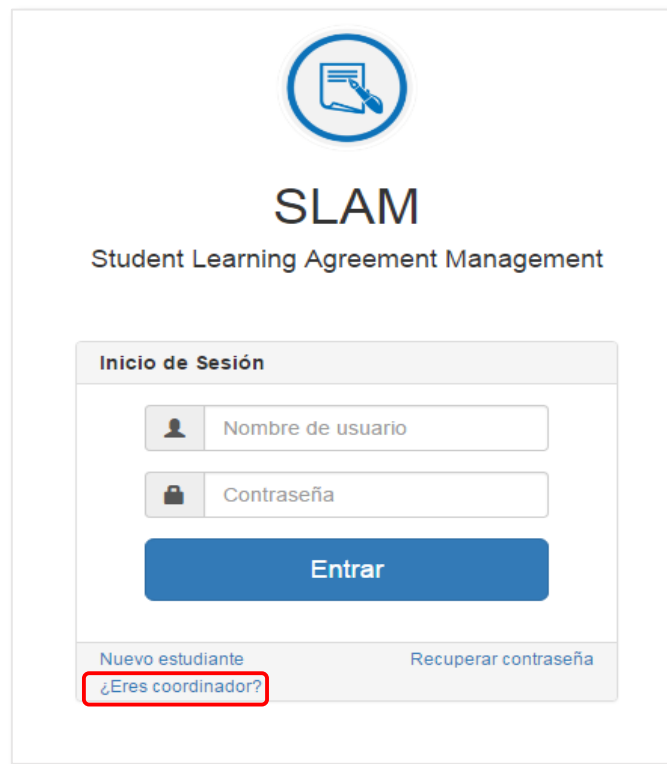
Los mensajes de color verde son aquellos enviados por el estudiante.

En caso de tener mensajes sin leer se notificará en la barra de navegación su número, que se pondrá a cero una vez se acceda al buzón.

8.1.2. Manual de usuario Coordinador

- *Acceso mediante Nombre Usuario/Contraseña*

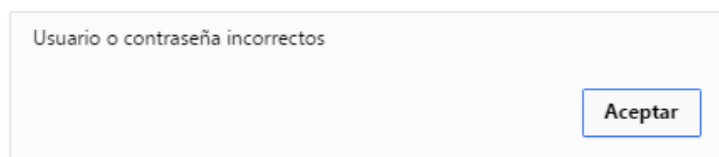
Inicialmente se muestra la pantalla de Login, en la que se debe introducir un usuario y contraseña válidos para poder acceder a la pantalla de inicio.



The screenshot shows the login interface for SLAM (Student Learning Agreement Management). At the top, there is a circular logo with a document and a pen. Below the logo, the text 'SLAM' is displayed in a large font, followed by 'Student Learning Agreement Management' in a smaller font. The main content area is titled 'Inicio de Sesión' and contains a login form. The form has two input fields: 'Nombre de usuario' (with a person icon) and 'Contraseña' (with a lock icon). Below these fields is a blue 'Entrar' button. At the bottom of the form, there are three links: 'Nuevo estudiante', 'Recuperar contraseña', and '¿Eres coordinador?'. The '¿Eres coordinador?' link is highlighted with a red rectangular box.

Figura 48. Pantalla login

En caso de producirse un error se mostrara un mensaje informativo y el coordinador podrá volver a introducir las credenciales.



The screenshot shows an error message box with the text 'Usuario o contraseña incorrectos'. In the bottom right corner of the box, there is a button labeled 'Aceptar'.

Figura 49. Error autenticación

En el caso de que no esté dado de alta en el sistema, si se pulsa sobre el enlace marcado en rojo de la figura 48, aparece la siguiente pantalla (figura 50), con las indicaciones necesarias para crear una cuenta de coordinador.

Bienvenido a la aplicación SLAM

Registrarse como coordinador

Para crear su cuenta de coordinador y gestionar los learning agreements de sus alumnos, siga las siguientes indicaciones.

- Envíe un e-mail a la dirección de correo **siam.uz.app@gmail.com** con los siguientes datos:
 1. **Nombre**
 2. **Apellidos**
 3. **Dirección de correo**
 4. **El nombre de las titulaciones de las cuales eres el coordinador de Erasmus.**
- Una vez que el administrador le haya dado de alta en el sistema, se le enviarán las credenciales de acceso para que pueda acceder a la aplicación.

[Atras](#)

Figura 50. Crear cuenta coordinador

- [Restablecer Contraseña](#)

El coordinador puede restablecer la contraseña insertando su nombre de usuario y su dirección de e-mail. El sistema le enviará las nuevas credenciales de acceso a la dirección facilitada.

Recuperación de Contraseña

Recupere su contraseña completando los datos del formulario

Nombre Usuario (login):

Dirección de correo:

No soy un robot 
reCAPTCHA
Privacidad - Condiciones

* La nueva contraseña se generará automáticamente en el servidor y será enviada a su dirección de correo.

[Atras](#) [Enviar](#)

Figura 51. Pantalla recuperación contraseña

- Inicio

En la pantalla de inicio se muestran las principales funcionalidades de la aplicación.

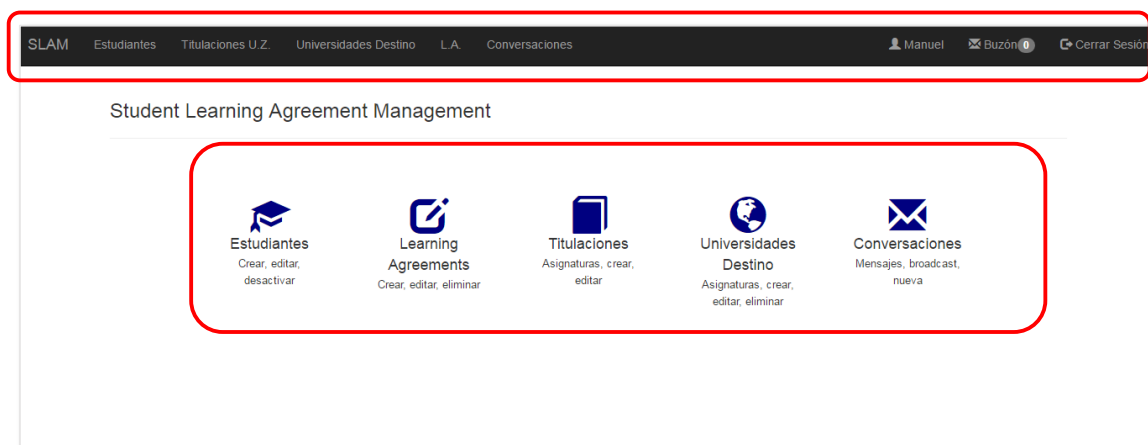


Figura 52. Pantalla de inicio

A continuación, se definen los recuadros señalados en la figura anterior:

- Barra de navegación, compuesta por los enlaces:
 - SLAM: enlace a la pantalla principal.
 - Estudiantes: enlace a la pantalla de gestión de estudiantes
 - Titulaciones: enlace a la pantalla de gestión de titulaciones de la U.Z.
 - Universidades Destino: enlace a la pantalla de gestión de las universidades de destino
 - Learning Agreement: enlace a pantalla de gestión de Learning Agreements.
 - Conversaciones: enlace a la gestión de las conversaciones.
 - Perfil: enlace al perfil de coordinador.
 - Mensajes: enlace a la gestión de las conversaciones.
 - Número de mensajes pendientes de lectura.
 - Cerrar Sesión: cierra la sesión del usuario y sale del sistema.
- Botones principales
 - Estudiantes: enlace a la pantalla de gestión de estudiantes
 - Titulaciones: enlace a la pantalla de gestión de titulaciones de la U.Z.
 - Universidades Destino: enlace a la pantalla de gestión de las universidades de destino
 - Learning Agreement: enlace a pantalla de gestión de Learning Agreements.
 - Conversaciones: enlace a la gestión de las conversaciones.

- *Gestión de Estudiantes*

En la pantalla de gestión de estudiantes (figura 53) se visualizan los estudiantes registrados en la(s) titulaciones gestionadas por el coordinador

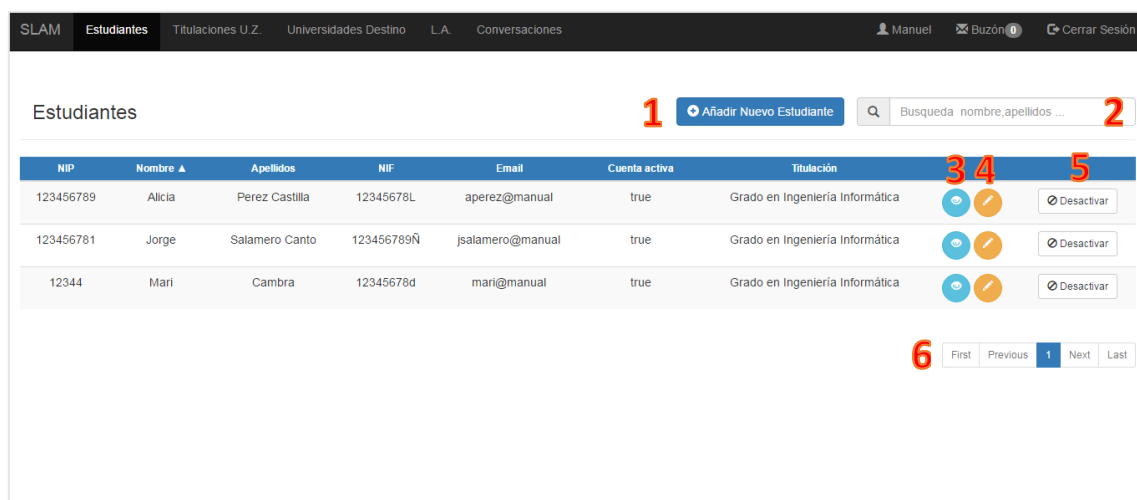


Figura 53. Gestión de estudiantes

Están disponibles las siguientes opciones:

1. Dar de alta un nuevo estudiante
2. Realizar búsquedas por NIP, nombre ...
3. Visualizar los datos de perfil del estudiante
4. Editar los datos de perfil del estudiante
5. Desactivar la cuenta del estudiante (No podrá acceder a la aplicación)
6. Paginación tabla

- *Registro nuevo estudiante*

Para registrar un nuevo estudiante es necesario completar los datos del formulario que se presenta en la siguiente figura.

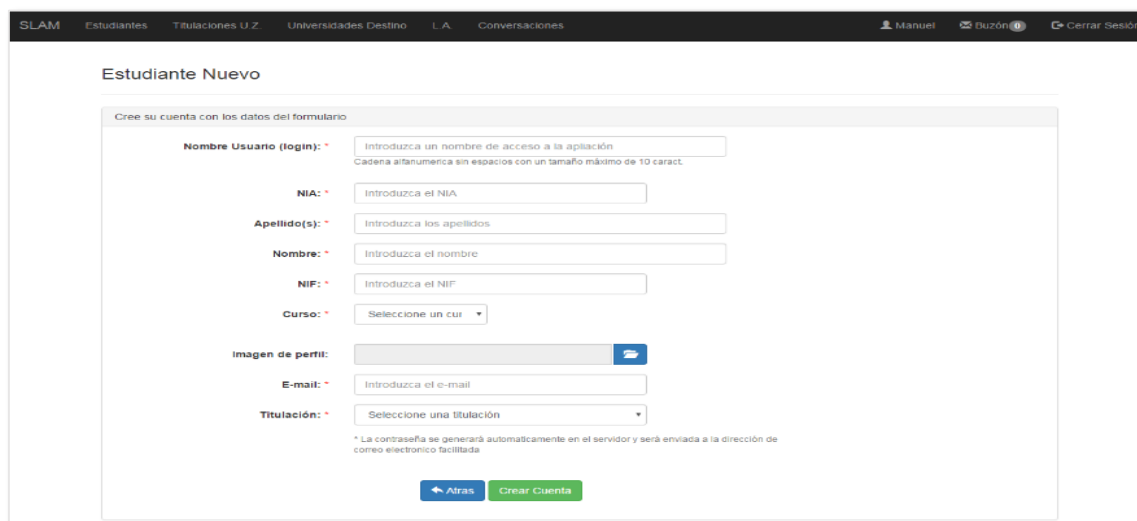


Figura 54. Interfaz registro nuevo estudiante

- *Visualización del perfil de estudiante*

Se visualiza los datos del estudiante, si se pulsa sobre el botón 'editar perfil' se enlazará con el formulario de edición (figura 56)

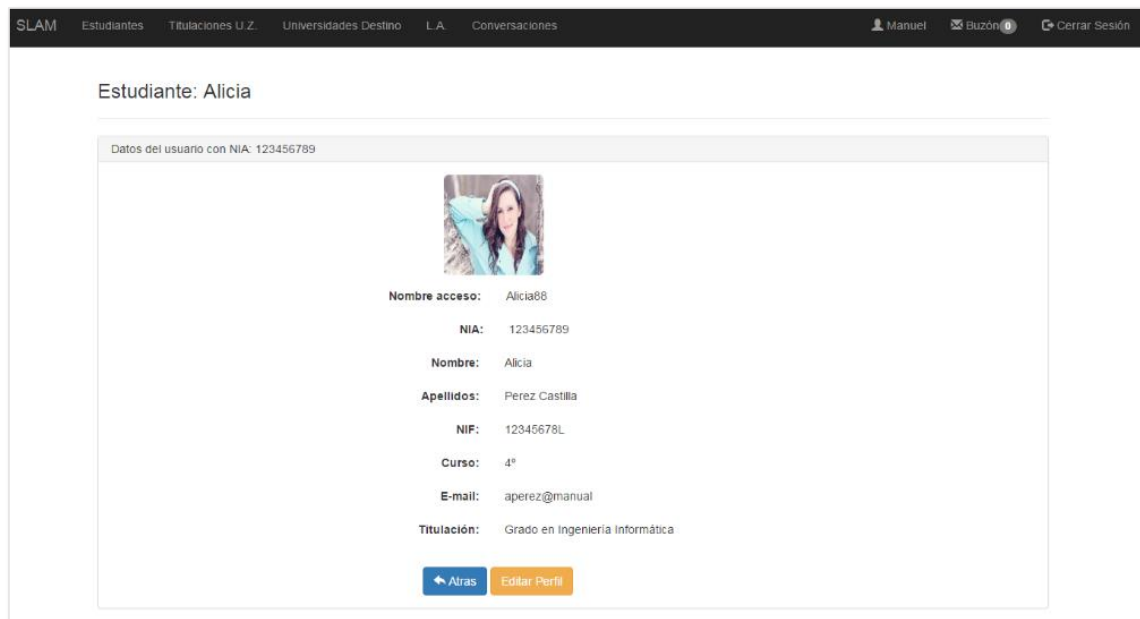


Figura 55. Interfaz visualización perfil estudiante

- *Edición del perfil de estudiante*

En la siguiente figura se visualiza un formulario con la información precargada del estudiante. Tras realizar las modificaciones se debe pulsar sobre el botón 'Actualizar' para que los cambios queden registrados en la aplicación.

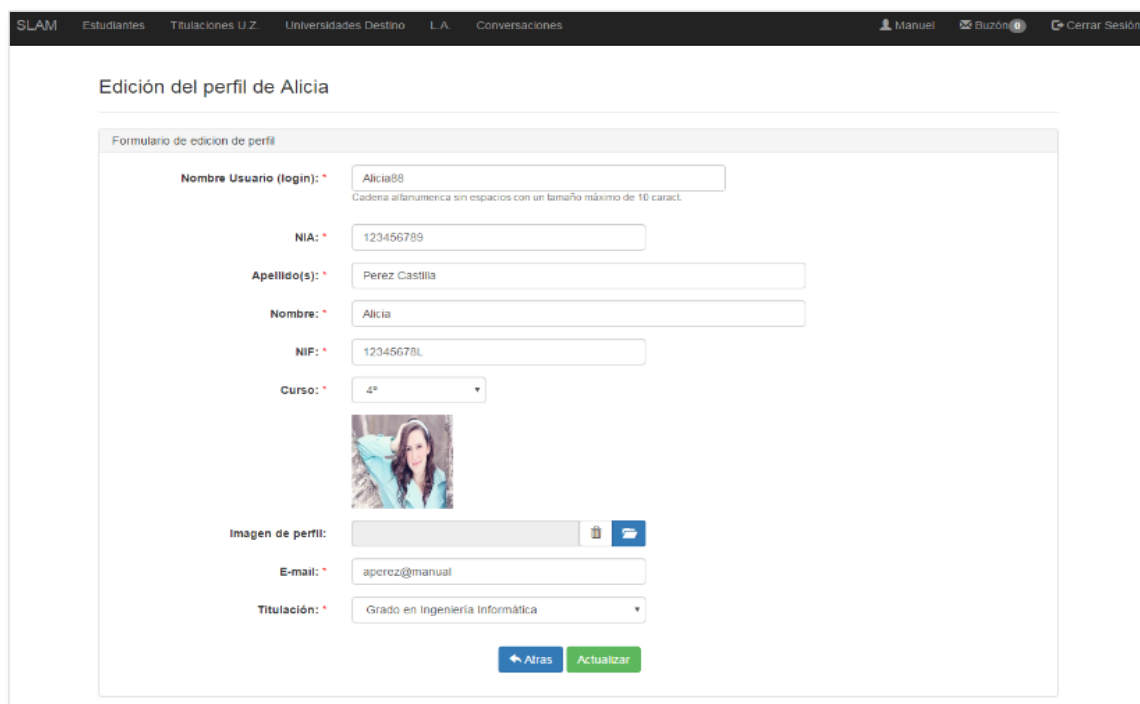


Figura 56. Interfaz edición de perfil

- *Gestión de titulaciones*

En la pantalla de gestión de titulaciones (figura 57) se visualizan las titulaciones de la Universidad de Zaragoza.

Figura 57. Interfaz gestión de titulaciones

Están disponibles las siguientes opciones:

1. Realizar búsquedas por universidad, nombre ...
2. Visualizar los datos completos de la titulación y las asignaturas que la componen.
3. Paginación tabla.

- *Visualización información titulación*

En la siguiente figura se presentan los datos asociados la titulación y un listado de las asignaturas que la componen.

Figura 58. Interfaz datos titulación

A continuación se presentan las siguientes opciones de la interfaz:

1. Paginación tabla.
2. Búsqueda de asignaturas por código, nombre...
3. Enlace al formulario de inserción de una nueva asignatura.
4. Enlace a el formulario de edición de asignatura.

En caso de intentar crear o editar una asignatura de una titulación, de la cual no se es el coordinador, se mostrara el siguiente mensaje de error:

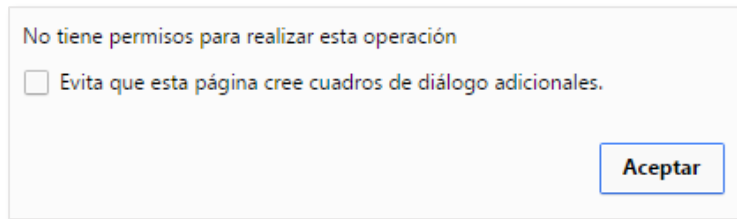


Figura 59. Mensaje error de permisos

- [Registrar nueva asignatura](#)

Para añadir una nueva asignatura a la titulación es necesario cumplimentar los datos del formulario que se presenta a continuación.

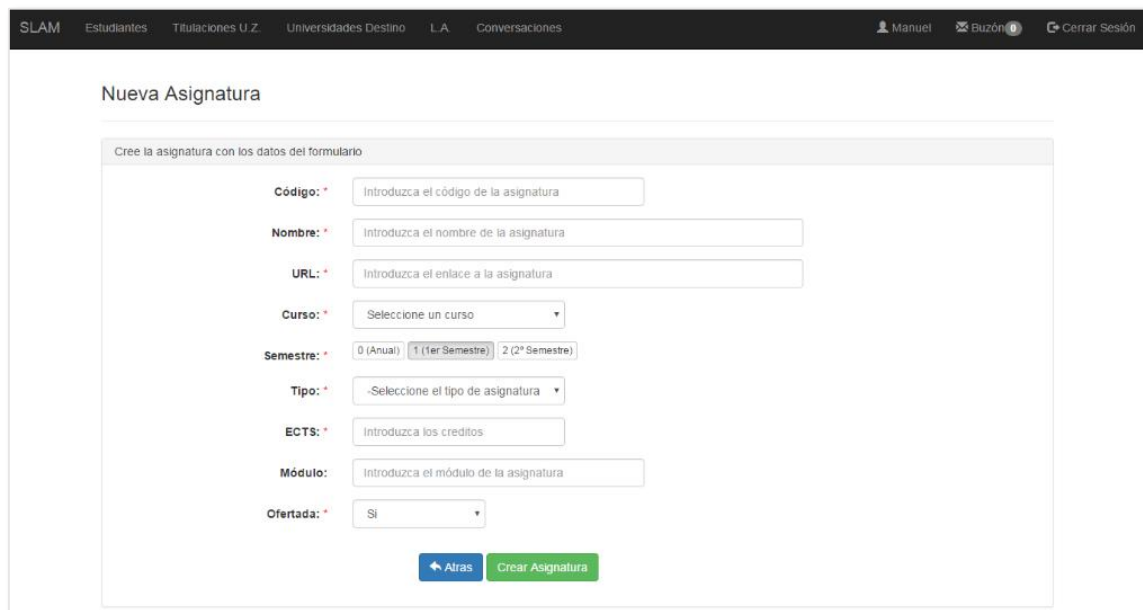
Una captura de pantalla de una interfaz web para crear una nueva asignatura. El título es "Nueva Asignatura". Debajo del título hay un recuadro con el texto "Cree la asignatura con los datos del formulario". El formulario contiene los siguientes campos: "Código:" con un campo de texto; "Nombre:" con un campo de texto; "URL:" con un campo de texto; "Curso:" con un menú desplegable que muestra "Seleccione un curso"; "Semestre:" con botones para "0 (Anual)", "1 (1er Semestre)" (seleccionado) y "2 (2º Semestre)"; "Tipo:" con un menú desplegable que muestra "-Seleccione el tipo de asignatura"; "ECTS:" con un campo de texto; "Módulo:" con un campo de texto; y "Ofertada:" con un menú desplegable que muestra "Si". En la parte inferior del formulario hay dos botones: "← Atras" (azul) y "Crear Asignatura" (verde).

Figura 60. Interfaz creación nueva asignatura

- [Edición de asignatura](#)

En la siguiente figura se visualiza un formulario con la información precargada de la asignatura. Tras realizar las modificaciones se debe pulsar sobre el botón 'Actualizar' para que los cambios queden registrados en la aplicación.

SLAM Estudiantes Titulaciones U.Z. Universidades Destino L.A. Conversaciones Manuel Buzón Cerrar Sesión

Editando la Asignatura: Técnicas creativas para la presentación de proyectos (TRV)

Formulario de edición de asignatura

Código: * 29985

Nombre: * Técnicas creativas para la presentación de proyectos (TRV)

URL: * http://titulaciones.unizar.es/guias16/index.php?asignatura=2998

Curso: * 4º

Semestre: * 0 (Anual) 1 (1er Semestre) 2 (2º Semestre)

Tipo: * Optativa

ECTS: * 4

Módulo: 111

Ofertada: * Si

Altras Actualizar Asignatura

Figura 61. Interfaz edición de asignatura

- *Gestión de Universidades de destino*

En la pantalla de gestión de universidades (figura 62) se visualizan las universidades de destino registradas en la aplicación.

SLAM Estudiantes Titulaciones U.Z. Universidades Destino L.A. Conversaciones Manuel Buzón Cerrar Sesión

Universidades Destino

1 Añadir Universidad Destino 2 Busqueda nombre, país ...

Nombre ▲	País	Web	Contacto	Cargo	E-mail	3 4
Danmarks Tekniske Universitet	Dinamarca	http://www.dtu.dk/	-	-	-	3 4
Luleå University of Technology	Suecia	http://www.ltu.se/?l=en	-	-	-	3 4
Universidad de Oporto	Portugal	-	Rob	Coordiador	-	3 4
Universidad de Torino	Italia	-	-	-	-	3 4

5 First Previous 1 Next Last

Figura 62. Interfaz gestión universidades destino

Están disponibles las siguientes opciones:

1. Crear nueva universidad de destino
2. Realizar búsquedas por país, nombre ...
3. Visualizar los datos de la universidad y sus asignaturas
4. Editar los datos de la universidad
5. Paginación tabla

- *Registro nueva universidad*

Para registrar una nueva universidad de destino es necesario completar los datos del formulario que se presenta en la siguiente figura.

SLAM Estudiantes Titulaciones U.Z. Universidades Destino L.A. Conversaciones Manuel Buzón Cerrar Sesión

Nueva Universidad

Cree la universidad con los datos del formulario

Nombre: *

Pais: *

Web:

Nombre contacto:

Cargo contacto:

E-mail de contacto:

[← Atras](#) [Crear Universidad](#)

Figura 63. Interfaz registro nueva universidad

- *Visualización información universidad*

En la siguiente figura se presentan los datos asociados la titulación y un listado de las asignaturas que la componen.

SLAM Estudiantes Titulaciones U.Z. Universidades Destino L.A. Conversaciones Manuel Buzón Cerrar Sesión

Universidad: Danmarks Tekniske Universitet

Nombre: Danmarks Tekniske Universitet Pais: Dinamarca Web: http://www.dtu.dk/

Contacto: Cargo: e-mail contacto:

1 [✎ Editar Universidad](#)

2

3 [+ Nueva Asignatura](#)

5 [First](#) [Previous](#) [1](#) [2](#) [Next](#) [Last](#)

Codigo	Nombre	URL	Curso	Tipo	ECTS	Semestre	4
2158	Concurrent programming	http://www.kurser.dtu.dk/02158.aspx?menulanguage=en-gb	BSc	FB	5	O	✎
2220	Distributed systems	http://www.kurser.dtu.dk/02220.aspx?menulanguage=en-gb	MSc	FB	7.5	P	✎
2223	Fundamentals of modern embedded systems	http://www.kurser.dtu.dk/02223.aspx?menulanguage=en-gb	MSc	FB	7.5	O	✎
2228	Fault-tolerant systems	http://www.kurser.dtu.dk/02228.aspx?menulanguage=en-gb	MSc	FB	7.5	O	✎
2244	Language based security	http://www.kurser.dtu.dk/02244.aspx?menulanguage=en-gb	MSc	FB	7.5	P	✎
2807	Computational tools for big data	http://www.kurser.dtu.dk/02807.aspx?menulanguage=en-gb	MSc	FB	5	O	✎

Figura 64. Interfaz datos universidad

Están disponibles las siguientes opciones:

1. Editar la información de la universidad
2. Realizar búsquedas por código, nombre ...
3. Registrar una nueva asignatura asociada a la universidad
4. Editar los datos de la asignatura
5. Paginación tabla

- *Edición de los datos de la universidad*

En la siguiente figura se visualiza un formulario con la información precargada de la universidad. Tras realizar las modificaciones se debe pulsar sobre el botón 'Actualizar' para que los cambios queden registrados en la aplicación.

SLAM Estudiantes Titulaciones U.Z. Universidades Destino L.A. Conversaciones Manuel Buzón Cerrar Sesión

Editando la Universidad: Danmarks Tekniske Universitet

Formulario de edición de universidad

Nombre: * Danmarks Tekniske Universitet

País: * Dinamarca

Web: http://www.dtu.dk/

Nombre contacto: Introduzca el nombre contacto

Cargo contacto: Introduzca el cargo del contacto

E-mail de contacto: Introduzca el email

Atras Actualizar

Figura 65. Interfaz edición datos universidad

- *Registro nueva asignatura de destino*

Para registrar una nueva asignatura de destino es necesario completar los datos del formulario que se presenta en la siguiente figura.

SLAM Estudiantes Titulaciones U.Z. Universidades Destino L.A. Conversaciones Manuel Buzón Cerrar Sesión

Nueva Asignatura

Cree la asignatura con los datos del formulario

Código: * Introduzca el código de la asignatura

Nombre: * Introduzca el nombre de la asignatura

URL: Introduzca el enlace a la asignatura

Curso: * Introduzca el curso

Semestre: * O(Otoño) P(Primavera)

Tipo: -Seleccione el tipo de asignatura

ECTS: * Introduzca los creditos

Atras Crear Asignatura

Figura 66. Interfaz registro nueva asignatura destino

- *Edición de los datos de una asignatura de destino*

En la siguiente figura se visualiza un formulario con la información precargada de la asignatura. Tras realizar las modificaciones se debe pulsar sobre el botón 'Actualizar' para que los cambios queden registrados en la aplicación.

Figura 67. Interfaz edición datos asignatura destino

- *Gestión de Learning Agreements*

En la pantalla de gestión de Learning Agreements (figura 68) se visualizan los acuerdos de estudios.

Estado	NIP	Estudiante	Titulación	Universidad Destino	Fecha Creación	Fecha última actualización	Version
Revisión Pendiente	123456781	Jorge Salamero Canto	Grado en Ingeniería Informática	Universidad de Torino	Sep 21, 2016 10:00:28 PM	Sep 21, 2016 10:00:28 PM	1
Pendiente estudiante	123456789	Alicia Perez Castilla	Grado en Ingeniería Informática	Universidad de Torino	Sep 21, 2016 9:46:29 PM	Sep 21, 2016 9:46:29 PM	0
Cerrado	12344	Mari Cambra	Grado en Ingeniería Informática	Universidad de Torino	Sep 22, 2016 1:55:36 AM	Sep 22, 2016 1:55:36 AM	1

Figura 68. Interfaz gestión Learning Agreements

Están disponibles las siguientes opciones:

1. Realizar búsquedas por NIP, nombre ...
2. Estado del acuerdo:
 - a. Revisión pendiente: tiene pendiente la revisión de un L.A.
 - b. Pendiente estudiante: El L.A. está en una etapa en la que debe ser revisado o confirmado por el alumno.
 - c. Evaluación pendiente: tiene pendiente la inserción de las calificaciones de las asignaturas que forman el L.A.
 - d. Cerrado: El L.A. se ha calificado y está cerrado.
3. Visualizar el acuerdo de estudios
4. Borrar el acuerdo de estudios
5. Paginación tabla

- *Etapa 1: Revisión Learning Agreement*

Una vez el estudiante envíe su Learning Agreement, el coordinador podrá visualizar la siguiente figura:

The screenshot shows the SLAM system interface. At the top, there are navigation tabs: SLAM, Estudios, Titulaciones U.Z., Universidades Destino, L.A., and Conversaciones. The user is logged in as 'Manuel'.

1 Datos Estudiante: This section displays the student's profile. It includes a profile picture, a name 'Salameo Canto, Jorge', and contact information: N.I.A. 123456781, DNI: 123456789Q, and e-mail: jsalameo@manuel. It also shows the student's current level (4º), degree (Grado en Ingeniería Informática), coordinator (Manuel), and duration (1 semestre(s)). Below this, it lists the destination university (Universidad de Torino) and its contact information (País: Italia, Contacto: e-mail).

2 Asignaturas Seleccionadas: This section shows a table of selected courses. The table is organized into two main groups: 'Bloque asignaturas Obligatorias / Formación Básica 6 seleccionada(s)' and 'Bloque asignaturas Opcionales 2 seleccionada(s)'. Each row represents a course assignment, showing the course code, name, origin level, origin ECTS, destination code, destination name, destination level, destination ECTS, and a dropdown menu for the assignment status.

3 The dropdown menu for the assignment status is highlighted. It has three options: 'Asignación Válida' (green checkmark), 'Asignación Rechazada' (red X), and 'Revisión pendiente' (grey). The 'Asignación Rechazada' option is currently selected, and a text box below it contains the reason: 'Temario no equivalente'.

4 The 'Enviar Revisión' button is highlighted at the bottom right of the interface.

Figura 69. Interfaz revisión acuerdo de estudios

En la figura anterior se presentan las siguientes opciones y características:

1. Datos del estudiante: información asociada al estudiante.
2. Bloque asignaturas: presentación de los bloques con las asignaturas de origen y sus respectivas asignaturas de destino.
3. Revisión: desplegable con las siguientes opciones
 - a. **Asignación válida:** la selección (asignatura(s) origen / asignatura(s) destino) es válida.
 - b. **Asignación rechazada:** la selección (asignatura(s) origen / asignatura(s) destino) no es correcta. En este caso se mostrara un cuadro de texto donde se inserta el motivo.
4. Envío de la revisión al estudiante. En caso de no haber realizado alguna revisión aparecerá el siguiente mensaje:

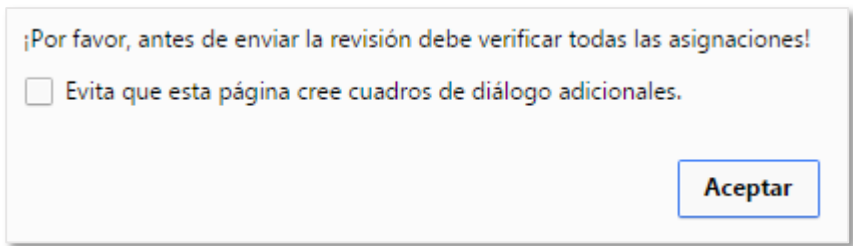


Figura 70. Mensaje error envío L.A.

- *Etapa 2: Calificaciones Learning Agreement*

Cuando se disponga de las calificaciones de las asignaturas, deberán ser insertadas a través de la interfaz que aparece en la siguiente figura:

The screenshot shows the 'Datos Learning Agreement' page for a student named Pérez Castilla, Aikía. The page displays a table of courses with their respective grades. The table is organized into two main sections: 'Bloque asignaturas Obligatorias / Formación Básica' and 'Bloque asignaturas Optativas'. Each row represents a course with its code, name, level, ECTS credits, and the grade entered by the student. The grades for the first two courses in the first section are 6 and 5, while the others are empty. The second section also shows empty grade boxes for its courses.

Asignatura a cursar en la EIMA		NOTA	Asignatura(s) a cursar en Universidad de Torino		NOTA
30235	Procesadores comerciales	6	MFND973	Modelazione Grafica	6
30262	Videogames	5	MFND976	Sistemi di realtà virtuale	5
30234	Informática gráfica		MFND943	Apprendimento Automatico e Analisi Intelligente dei Dati	
30233	Recuperación de información		MFND909	Calcolabilità e complessità	
30227	Seguridad informática		MFND996	Blockchain	
30256	Sistemas y tecnologías web		MFND643	Tecnologie web	
30264	Comercio electrónico		INF0002	Servizi web	

Bloque genérico de créditos optativos sobrantes: 9 ECTS

Enviar evaluación

Figura 71. Inserción calificaciones

Si se pulsa sobre el botón ‘Enviar evaluación’ y no se han calificado todas las asignaturas aparecerá el siguiente mensaje de error.

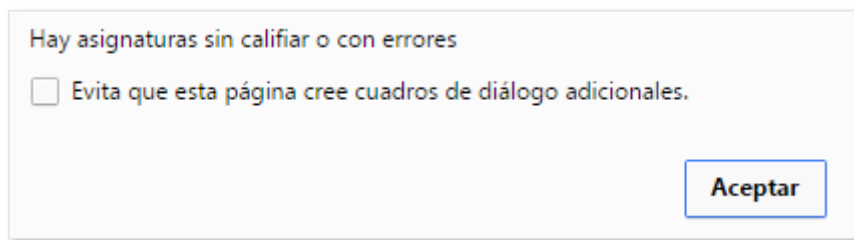


Figura 72. Mensaje error envío evaluación

En caso de que el envío sea satisfactorio el estado del Learning Agreement pasara a ‘cerrado’ y podrá ser descargado en formato PDF.

Bloque asignaturas Obligatorias / Formación Básica 5 seleccionada(s)	
Asignatura a convalidar en la EINA	Asignatura(s) a cursar en Universidad de Torino
30235 Procesadores comerciales 3º 6 ECTS 6	MFN0973 Modellazione Grafica 4º 9 ECTS 8
Asignatura a convalidar en la EINA	Asignatura(s) a cursar en Universidad de Torino
30262 Videjuegos 4º 6 ECTS 5	MFN0978 Sistemi di realtà virtuale 4º 9 ECTS 5
Asignatura a convalidar en la EINA	Asignatura(s) a cursar en Universidad de Torino
30234 Informática gráfica 4º 6 ECTS 9	MFN0943 Apprendimento Automatico e Analisi Intelligente dei Dati 4º 9 ECTS 10
Asignatura a convalidar en la EINA	Asignatura(s) a cursar en Universidad de Torino
30233 Recuperación de información 4º 6 ECTS 8	MFN0939 Calcolabilità e complessità 4º 6 ECTS 8
Asignatura a convalidar en la EINA	Asignatura(s) a cursar en Universidad de Torino
30227 Seguridad informática 4º 6 ECTS 3	MFN0996 Sicurezza 4º 6 ECTS 3
Bloque asignaturas Optativas 2 seleccionada(s)	
Asignatura a convalidar en la EINA	Asignatura(s) a cursar en Universidad de Torino
30256 Sistemas y tecnologías web 4º 6 ECTS 5	MFN0643 Tecnologie web 4º 6 ECTS 5
Asignatura a convalidar en la EINA	Asignatura(s) a cursar en Universidad de Torino
30264 Comercio electrónico 4º 6 ECTS 5	INF0002 Servizi web 3º 6 ECTS 5

Bloque genérico de créditos optativos sobrantes: 9 ECTS

Figura 73. Interfaz acuerdo de estudios cerrado

En caso de pulsar sobre el botón ‘Modificar evaluaciones’ el Learning Agreement mostrara de nuevo la figura 71.

- *Gestión de Conversaciones*

En la pantalla de gestión de conversaciones se visualiza un listado con la información de las conversaciones establecidas.

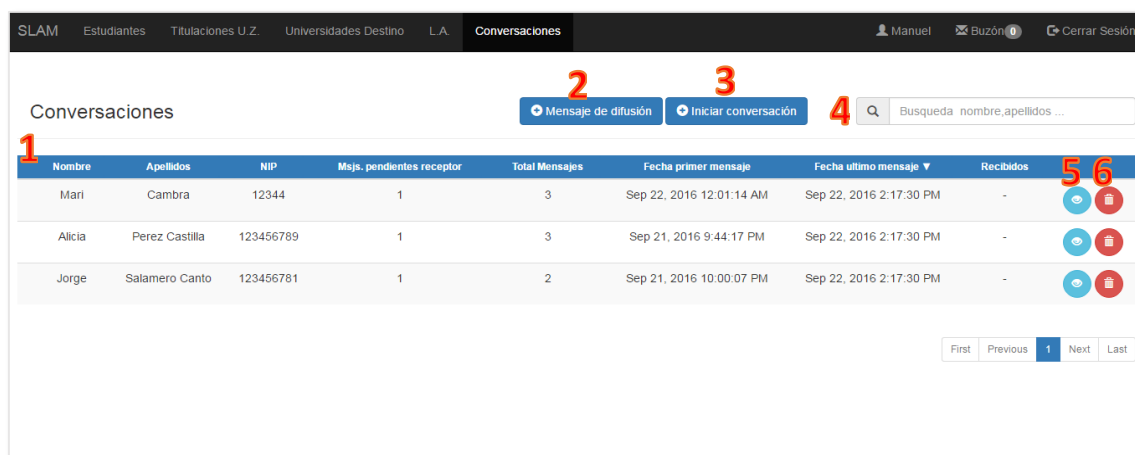


Figura 74. Interfaz gestión conversaciones

En la figura 74 se presentan las siguientes opciones y características:

1. Listado de las conversaciones.
2. Mensaje de difusión: permite realizar el envío de un mensaje a todos los estudiantes registrados en la titulación(es) gestionadas por el coordinador de movilidad.
3. Iniciar conversación: Inicia una conversación con un estudiante.
4. Realizar búsquedas por NIP, nombre ...
5. Acceder a la conversación.
6. Eliminar conversación.

- *Mensaje de Difusión*

A través de la interfaz que se muestra en la siguiente figura, se realiza el envío de un mensaje a todos los estudiantes pertenecientes a alguna de las titulaciones gestionadas por el coordinador.

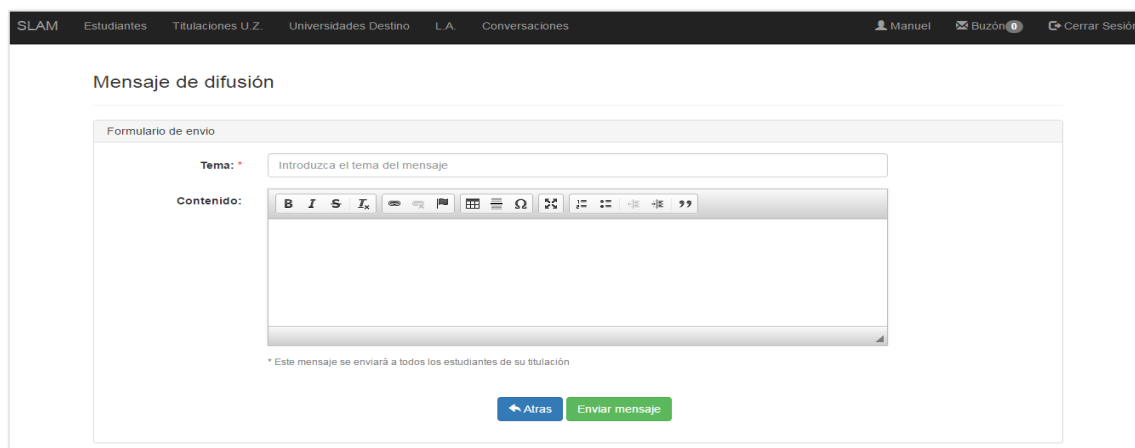


Figura 75. Interfaz envío mensaje de difusión

- *Iniciar una conversación*

A través de la interfaz que se muestra en la siguiente figura, se puede iniciar una conversación en caso de que no exista con un estudiante.

Figura 76. Interfaz inicio de conversación

Cuando el coordinador seleccione el estudiante al que desee enviar el mensaje automáticamente se habilitaran el resto de campos.

En caso de que seleccione un estudiante con una conversación ya en curso, se mostrara el siguiente mensaje:

Figura 77. Mensaje conversación ya existente

- *Visualización y envío de mensajes*

A través de la interfaz que se muestra en la siguiente figura, se pueden visualizar los mensajes enviados y recibidos de la conversación, así como realizar el envío de un nuevo mensaje.

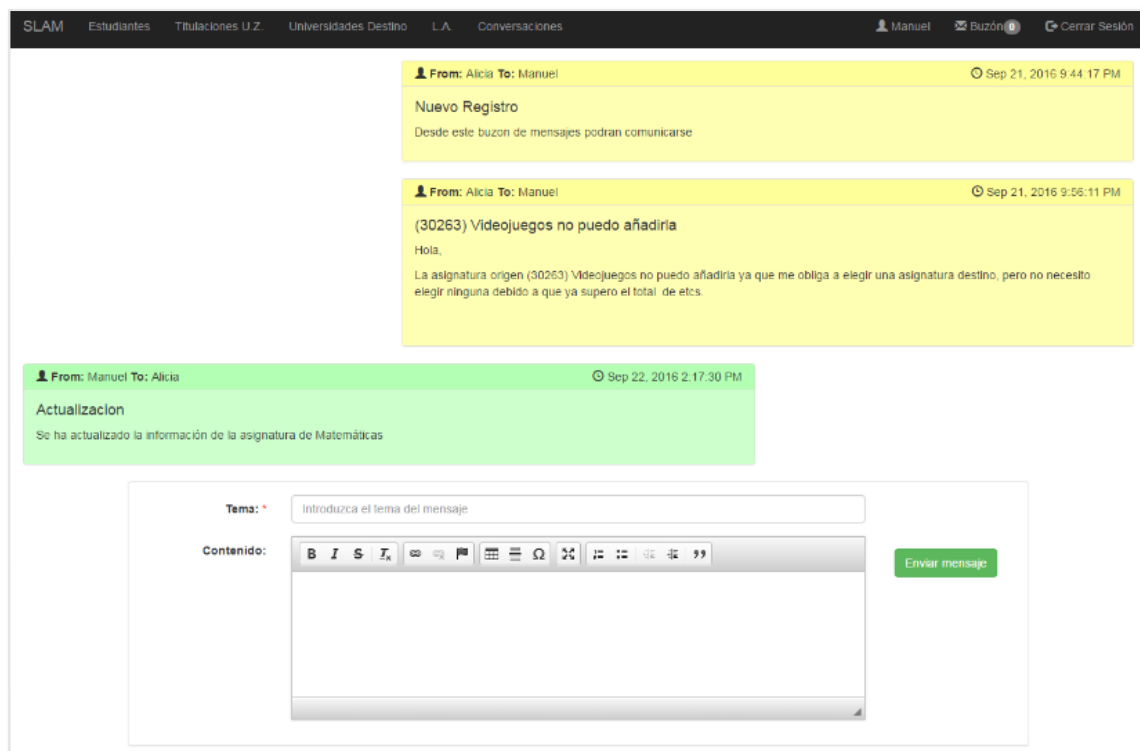


Figura 78. Interfaz visualización conversación

Los mensajes mostrados en color verde son los enviados por el coordinador.

8.1.3. Manual de usuario Administrador

- *Aceso mediante Nombre Usuario/Contraseña*

Inicialmente se muestra la pantalla de login, en la que se debe introducir un usuario y contraseña válidos para poder acceder a la pantalla de inicio.

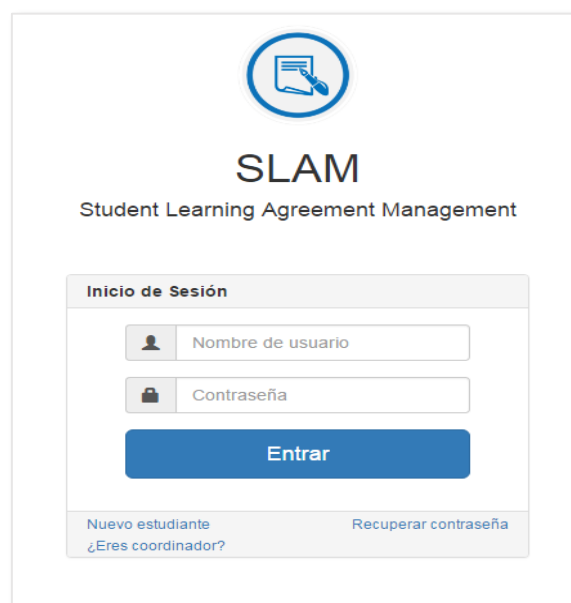


Figura 79. Pantalla login

En caso de producirse un error se mostrara un mensaje informativo y el administrador podrá volver a introducir las credenciales.

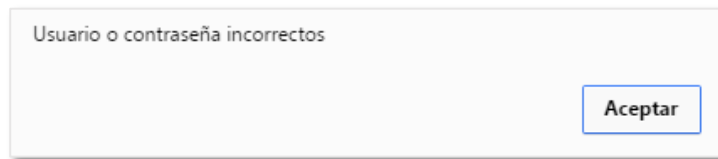


Figura 80. Error autenticación

- *Inicio*

En la pantalla de inicio se muestran las principales funcionalidades de la aplicación.

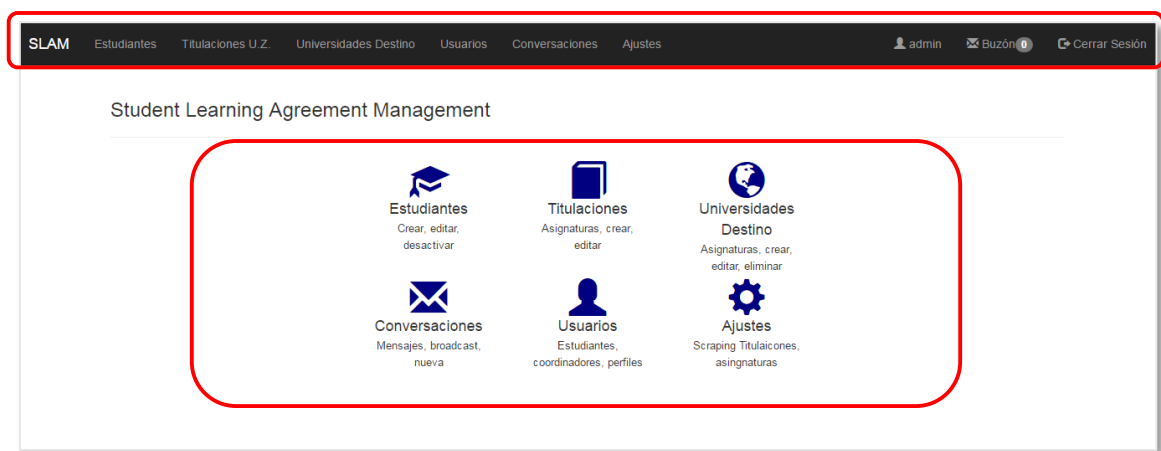


Figura 81. Pantalla de inicio

A continuación, se definen los recuadros señalados en la figura anterior:

- Barra de navegación, compuesta por los enlaces:
 - SLAM: enlace a la pantalla principal.
 - Estudiantes: enlace a la pantalla de gestión de estudiantes
 - Titulaciones: enlace a la pantalla de gestión de titulaciones de la U.Z.
 - Universidades Destino: enlace a la pantalla de gestión de las universidades de destino
 - Usuarios: enlace a la pantalla de gestión de usuarios
 - Conversaciones: enlace a la gestión de las conversaciones.
 - Ajustes: enlace a la sección de ajustes.
 - Perfil: enlace al perfil de administrador.
 - Mensajes: enlace a la gestión de las conversaciones.
 - Número de mensajes pendientes de lectura.
 - Cerrar Sesión: cierra la sesión del usuario y sale del sistema.
- Botones principales
 - Estudiantes: enlace a la pantalla de gestión de estudiantes
 - Titulaciones: enlace a la pantalla de gestión de titulaciones de la U.Z.

- Universidades Destino: enlace a la pantalla de gestión de las universidades de destino
- Usuarios: enlace a la pantalla de gestión de usuarios
- Conversaciones: enlace a la pantalla de gestión de las conversaciones.
- Ajustes: enlace a la sección de ajustes.

- *Gestión de Estudiantes*

En la pantalla de gestión de estudiantes (figura 82) se visualizan los estudiantes registrados en la aplicación.

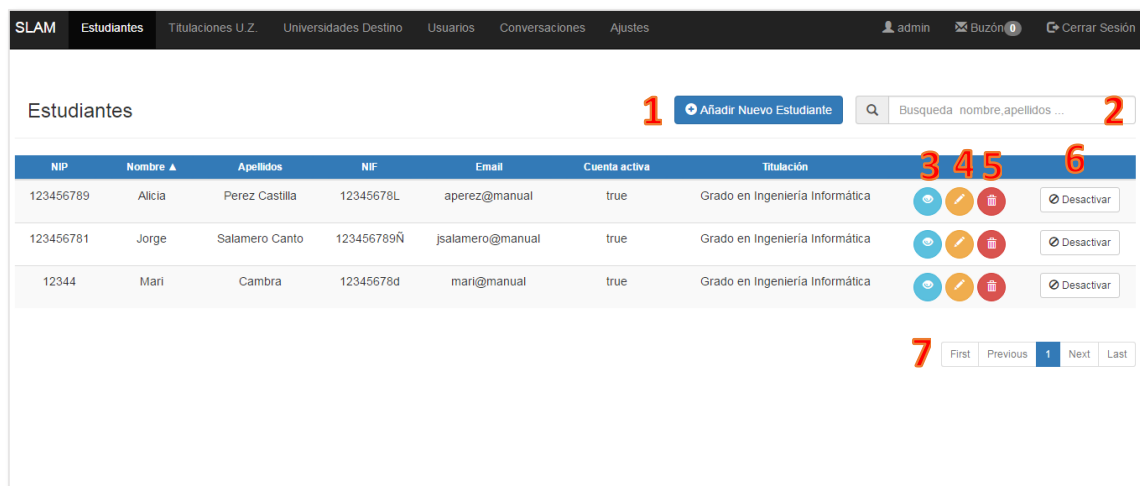


Figura 82. Gestión de estudiantes

Están disponibles las siguientes opciones:

1. Dar de alta un nuevo estudiante.
2. Realizar búsquedas por NIP, nombre ...
3. Visualizar los datos de perfil del estudiante.
4. Editar los datos de perfil del estudiante.
5. Borrar el estudiante.
6. Desactivar la cuenta del estudiante (No podrá acceder a la aplicación).
7. Paginación tabla.

- *Registro nuevo estudiante*

Para registrar un nuevo estudiante es necesario completar los datos del formulario que se presenta en la siguiente figura.

Figura 83. Interfaz registro nuevo estudiante

- *Visualización del perfil de estudiante*

Se visualiza los datos del estudiante, si se pulsa sobre el botón 'editar perfil' se enlazara con el formulario de edición (figura 85)

Figura 84. Interfaz visualización perfil estudiante

- *Edición del perfil de estudiante*

En la siguiente figura se visualiza un formulario con la información precargada del estudiante. Tras realizar las modificaciones se debe pulsar sobre el botón 'Actualizar' para que los cambios queden registrados en la aplicación.

Figura 85. Interfaz edición de perfil

- *Gestión de titulaciones*

En la pantalla de gestión de titulaciones (figura 86) se visualizan las titulaciones de la Universidad de Zaragoza.

Nombre ▲	Universidad	Web	Ofertada	Coordinador
Grado en Administración y Dirección de Empresas	Universidad de Zaragoza	http://titulaciones.unizar.es/admin-direc-empresas	Si	-
Grado en Arquitectura Técnica	Universidad de Zaragoza	http://titulaciones.unizar.es/ing-edificacion	Si	-
Grado en Bellas Artes	Universidad de Zaragoza	http://titulaciones.unizar.es/bellas-artes	Si	-
Grado en Biotecnología	Universidad de Zaragoza	http://titulaciones.unizar.es/biotecnologia	Si	-
Grado en Ciencia y Tecnología de los Alimentos	Universidad de Zaragoza	http://titulaciones.unizar.es/tecnologia-alimentos	Si	-
Grado en Ciencias Ambientales	Universidad de Zaragoza	http://titulaciones.unizar.es/ciencias-ambientales	Si	-
Grado en Ciencias de la Actividad Física y del Deporte	Universidad de Zaragoza	http://titulaciones.unizar.es/act-fisica-deporte	Si	-

Figura 86. Interfaz gestión de titulaciones

Están disponibles las siguientes opciones:

1. Crear nueva titulación.
2. Realizar búsquedas por universidad, nombre ...
3. Visualizar los datos completos de la titulación y las asignaturas que la componen.

4. Editar titulación
5. Borrar titulación
6. Paginación tabla

- *Crear titulación*

Para crear una nueva titulación es necesario completar los datos del formulario que se muestra en la siguiente figura.

SLAM Estudiantes Titulaciones U.Z. Universidades Destino Usuarios Conversaciones Ajustes admin Buzón 0 Cerrar Sesión

Titulación Nueva

Cree la titulación con los datos del formulario

Nombre: * Introduzca el nombre

Universidad: * Universidad de Zaragoza

Ofertada: * Si

E-mail de contacto: Introduzca el email

Web: Introduzca la web

Teléfono: Introduzca el teléfono

Coordinador de la titulación:

Atras Crear Titulación

Figura 87. Creación nueva titulación

- *Editar titulación*

En la siguiente figura se visualiza un formulario con la información precargada de la titulación. Tras realizar las modificaciones se debe pulsar sobre el botón 'Actualizar' para que los cambios queden registrados en la aplicación.

SLAM Estudiantes Titulaciones U.Z. Universidades Destino Usuarios Conversaciones Ajustes admin Buzón 0 Cerrar Sesión

Editando la titulación: Grado en Arquitectura Técnica

Formulario de edicion de titulación

Nombre: * Grado en Arquitectura Técnica

Universidad: * Universidad de Zaragoza

Ofertada: * Si

E-mail de contacto: Introduzca el email

Web: http://titulaciones.unizar.es/ing-edificacion

Coordinador de la titulación:

Atras Actualizar

Figura 88. Interfaz edición de titulación

- *Visualización información titulación*

En la siguiente figura se presentan los datos asociados a la titulación y un listado de las asignaturas que la componen.

The screenshot shows the SLAM web interface for the 'Grado en Arquitectura Técnica' degree. The page includes a header with navigation links (SLAM, Estudiantes, Titulaciones U.Z., Universidades Destino, Usuarios, Conversaciones, Ajustes) and user information (admin, Buzón, Cerrar Sesión). The main content area displays the degree details, including the university (Universidad de Zaragoza), name (Grado en Arquitectura Técnica), and contact information. A red '1' highlights the 'Editar Titulación' button. Below the details is a pagination control (red '2') and a search bar (red '3'). A red '4' highlights the 'Nueva Asignatura' button. A red '5' highlights the '56' count of subjects. A red '6' highlights the edit/delete icons for each subject in the table below.

Codigo ▲	Nombre	Curso	Tipo	ECTS	Semestre	Modulo	Oferida	
24900	Idioma moderno Inglés B1	4	OB	2	0	0	Si	✓ [X]
28600	Matemática aplicada a la edificación I	1	FB	6	1	99	Si	✓ [X]
28601	Física I: mecánica general	1	FB	6	1	99	Si	✓ [X]
28602	Fundamentos de los materiales de construcción	1	FB	6	1	100	Si	✓ [X]
28603	Expresión gráfica aplicada a la edificación	1	FB	6	1	99	Si	✓ [X]
28604	Historia de la construcción	1	OB	6	1	100	Si	✓ [X]

Figura 89. Interfaz datos titulación

A continuación se presentan las siguientes opciones de la interfaz:

1. Editar titulación
2. Paginación tabla.
3. Búsqueda de asignaturas por código, nombre...
4. Crear nueva asignatura
5. Editar asignatura
6. Borrar asignatura

- *Registrar nueva asignatura*

Para añadir una nueva asignatura a la titulación es necesario cumplimentar los datos del formulario que se presenta a continuación.

Figura 90. Interfaz creación nueva asignatura

- *Edición de asignatura*

En la siguiente figura se visualiza un formulario con la información precargada de la asignatura. Tras realizar las modificaciones se debe pulsar sobre el botón 'Actualizar' para que los cambios queden registrados en la aplicación.

Figura 91. Interfaz edición de asignatura

- *Gestión de Universidades de destino*

En la pantalla de gestión de universidades (figura 92) se visualizan las universidades de destino registradas en la aplicación.

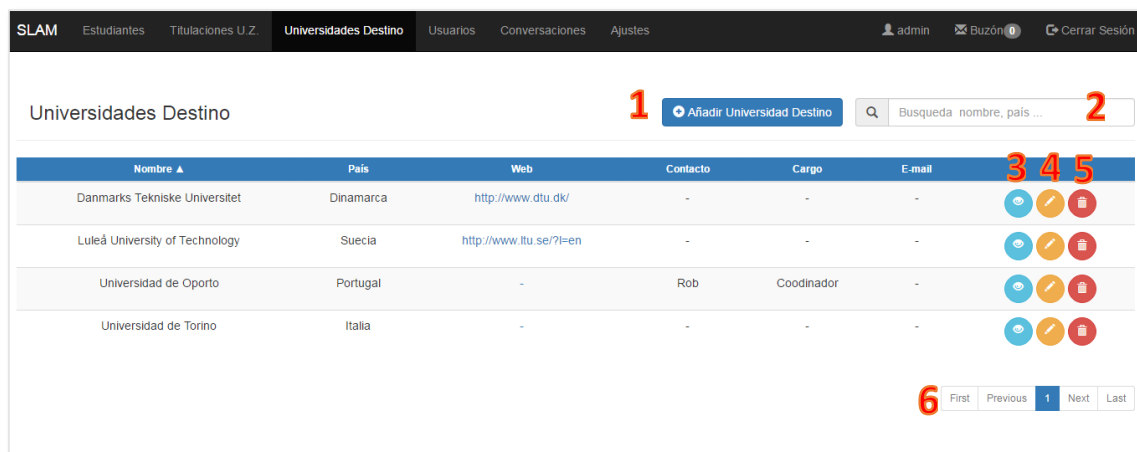


Figura 92. Interfaz gestión universidades destino

Están disponibles las siguientes opciones:

1. Crear nueva universidad de destino
2. Realizar búsquedas por país, nombre ...
3. Visualizar los datos de la universidad y sus asignaturas
4. Editar los datos de la universidad
5. Borrar la universidad de destino
6. Paginación tabla

- [Registro nueva universidad](#)

Para registrar una nueva universidad de destino es necesario completar los datos del formulario que se presenta en la siguiente figura.

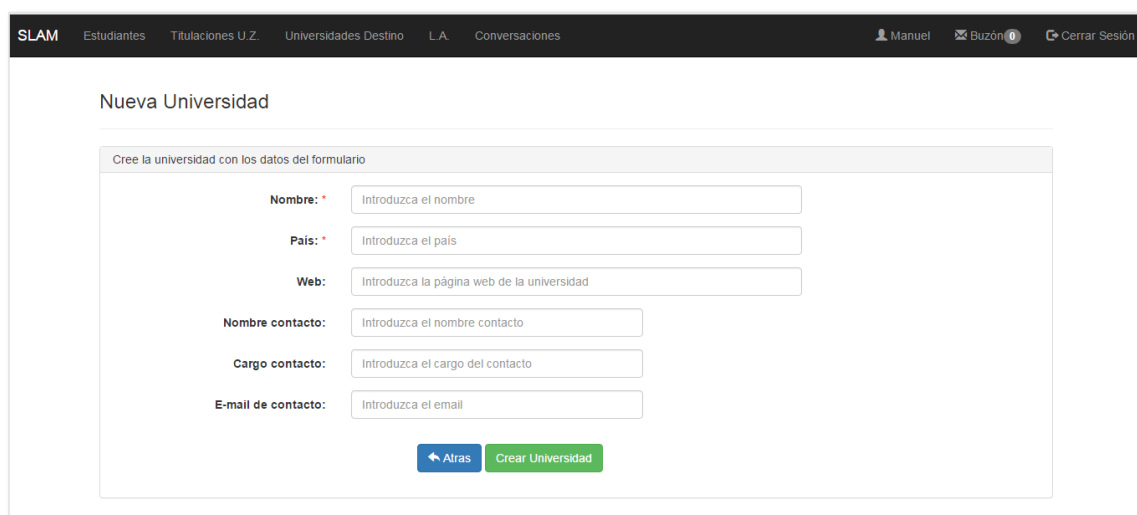


Figura 93. Interfaz registro nueva universidad

- [Visualización información universidad](#)

En la siguiente figura se presentan los datos asociados a la universidad y un listado de las asignaturas que la componen.

SLAM Estudiantes Titulaciones U.Z. Universidades Destino Usuarios Conversaciones Ajustes admin Buzón Cerrar Sesión

Universidad: Universidad de Oporto

Nombre: Universidad de Oporto País: Portugal Web:

Contacto: Rob Cargo: Coordinador e-mail contacto:

1 Editar Universidad

6 First Previous 1 2 Next Last 2 Busqueda código,nombre ... 3 Nueva Asignatura

Código	Nombre	Curso	Tipo	ECTS	Semestre	45
EIC0028	Compilers	3	FB	6	P	✓
EIC0029	Artificial Intelligence	3	FB	6	P	✓
EIC0036	Sistemas Distribuidos	3	FB	6	P	✓
EIC0040	Sistemas de Informacio	4	FB	6	O	✓
EIC0071	Robótica	5	FB	6	O	✓
EIC0086	Laboratório de Desenvolvimento de Software	4	FB	7.5	O	✓

Figura 94. Interfaz datos universidad

Están disponibles las siguientes opciones:

1. Editar la información de la universidad
2. Realizar búsquedas por país, nombre ...
3. Registrar una nueva asignatura asociada a la universidad
4. Editar los datos de la asignatura
5. Borrar asignatura
6. Paginación tabla

- *Edición de los datos de la universidad*

En la siguiente figura se visualiza un formulario con la información precargada de la universidad. Tras realizar las modificaciones se debe pulsar sobre el botón 'Actualizar' para que los cambios queden registrados en la aplicación.

SLAM Estudiantes Titulaciones U.Z. Universidades Destino L.A. Conversaciones Manuel Buzón Cerrar Sesión

Editando la Universidad: Danmarks Tekniske Universitet

Formulario de edición de universidad

Nombre: * Danmarks Tekniske Universitet

País: * Dinamarca

Web: http://www.dtu.dk/

Nombre contacto: Introduzca el nombre contacto

Cargo contacto: Introduzca el cargo del contacto

E-mail de contacto: Introduzca el email

Atras Actualizar

Figura 95. Interfaz edición datos universidad

- *Registro nueva asignatura de destino*

Para registrar una nueva asignatura de destino es necesario completar los datos del formulario que se presenta en la siguiente figura.

The screenshot shows the 'Nueva Asignatura' (New Subject) registration form. The form is titled 'Cree la asignatura con los datos del formulario' (Create the subject with the data from the form). It contains the following fields:

- Código:** Introduce el código de la asignatura
- Nombre:** Introduce el nombre de la asignatura
- URL:** Introduce el enlace a la asignatura
- Curso:** Introduce el curso
- Semestre:** Radio buttons for 'O(Ototoño)' and 'P(Primavera)'
- Tipo:** Dropdown menu with '-Seleccione el tipo de asignatura'
- ECTS:** Introduce los creditos

At the bottom of the form are two buttons: 'Atras' (Back) and 'Crear Asignatura' (Create Subject).

Figura 96. Interfaz registro nueva asignatura destino

- *Edición de los datos de una asignatura de destino*

En la siguiente figura se visualiza un formulario con la información precargada de la asignatura. Tras realizar las modificaciones se debe pulsar sobre el botón 'Actualizar' para que los cambios queden registrados en la aplicación.

The screenshot shows the 'Editando la Asignatura' (Editing the Subject) form. The form is titled 'Formulario de edición de asignatura' (Subject editing form). It contains the following pre-filled fields:

- Código:** 2158
- Nombre:** Concurrent programming
- URL:** http://www.kurser.dtu.dk/02158.aspx?menulanguage=en-gb
- Curso:** BSc
- Semestre:** Radio buttons for 'O(Ototoño)' and 'P(Primavera)'
- Tipo:** Dropdown menu with 'Formación Básica'
- ECTS:** 5

At the bottom of the form are two buttons: 'Atras' (Back) and 'Actualizar' (Update).

Figura 97. Interfaz edición datos asignatura destino

- *Gestión de Usuarios*

En la pantalla de gestión de usuarios (figura 98) se visualizan los usuarios de la aplicación.

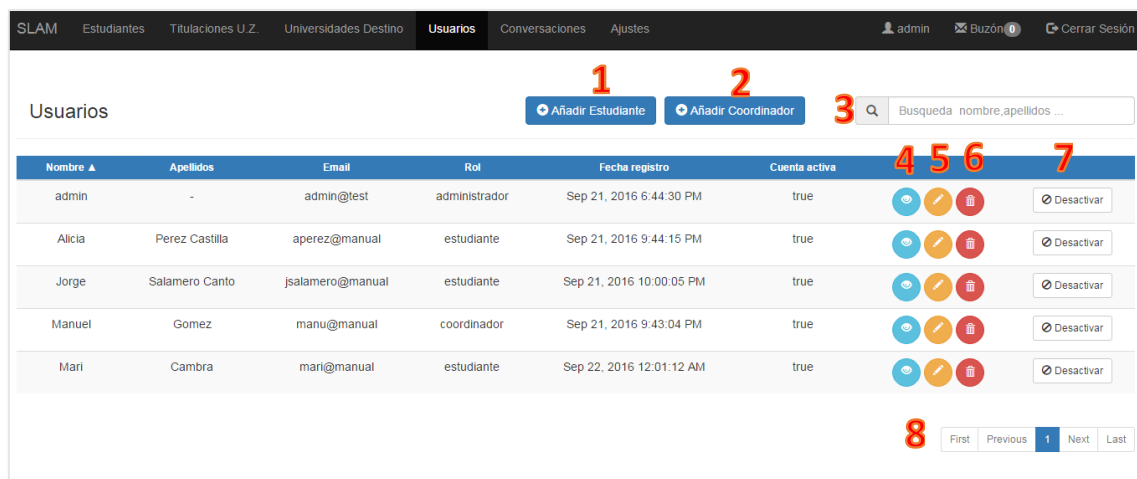


Figura 98. Gestión de usuarios

Están disponibles las siguientes opciones:

1. Dar de alta un nuevo estudiante.
2. Dar de alta un nuevo coordinador
3. Realizar búsquedas por rol, nombre ...
4. Visualizar los datos de perfil del usuario.
5. Editar los datos de perfil del usuario.
6. Borrar la cuenta del usuario.
7. Desactivar la cuenta del usuario (No podrá acceder a la aplicación).
8. Paginación tabla.

- *Registro nuevo estudiante*

Para registrar un nuevo estudiante es necesario completar los datos del formulario que se presenta en la siguiente figura.

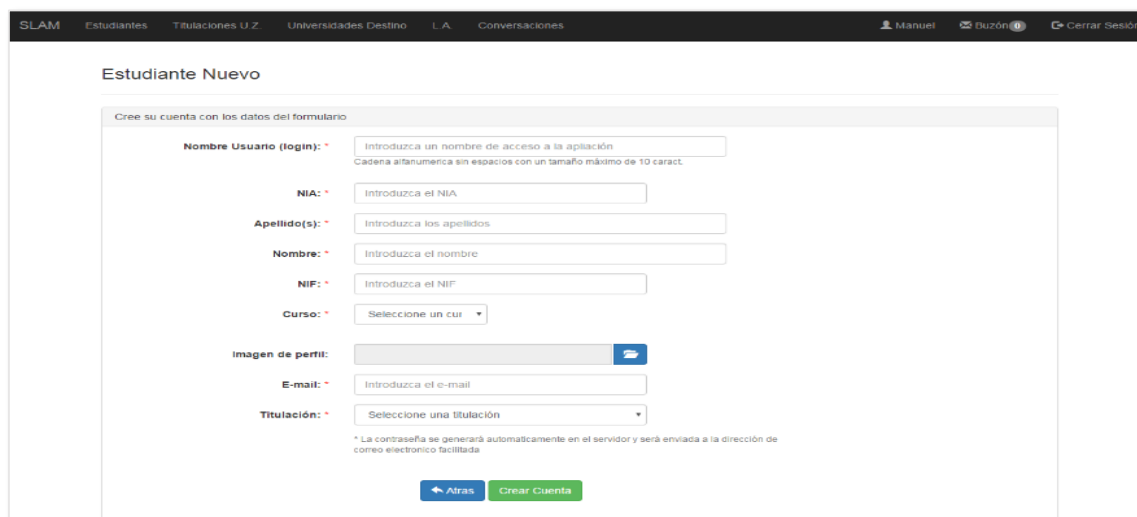


Figura 99. Interfaz registro nuevo estudiante

- *Registro nuevo coordinador*

Para registrar un nuevo coordinador es necesario completar los datos del formulario que se presenta en la siguiente figura.

The screenshot shows the 'Nuevo Coordinador' registration page. At the top, there is a navigation bar with 'SLAM' and menu items: 'Estudiantes', 'Titulaciones U.Z.', 'Universidades Destino', 'Usuarios', 'Conversaciones', and 'Ajustes'. On the right, there are user icons for 'admin', 'Buzón', and 'Cerrar Sesión'. The main content area is titled 'Nuevo Coordinador' and contains a form with the following fields: 'Nombre: *' (text input), 'Apellido(s): *' (text input), 'E-mail: *' (text input), 'Nombre Usuario (login):' (text input with value 'Generado automáticamente'), and 'Titulación(s):' (dropdown menu with value '---Seleccione la/s titulación---'). At the bottom of the form are two buttons: 'Atras' (blue) and 'Crear Cuenta' (green).

Figura 100. Interfaz registro nuevo coordinador

- *Visualización del perfil de usuario*

Se visualizan los datos del usuario, si se pulsa sobre el botón 'editar perfil' se enlazara con el formulario de edición (figura 102)

The screenshot shows the user profile page for 'Alicia'. The navigation bar is identical to the previous figure. The main content area is titled 'Usuario: Alicia' and contains a box with the heading 'Datos del usuario: Alicia'. The profile information is displayed as follows: 'Nombre acceso: Alicia88', 'Nombre: Alicia', 'Apellidos: Perez Castilla', 'e-mail: aperez@manual', 'Rol: estudiante', and 'Fecha creación: Sep 21, 2016'. At the bottom of the profile box are two buttons: 'Atras' (blue) and 'Editar Perfil' (orange).

Figura 101. Interfaz visualización perfil usuario

- *Edición del perfil de usuario*

En la siguiente figura se visualiza un formulario con la información precargada del usuario. Tras realizar las modificaciones se debe pulsar sobre el botón 'Actualizar' para que los cambios queden registrados en la aplicación.

Edición del perfil de Alicia

Formulario de edición de perfil

Nombre Usuario (login): * Alicia88

Nombre: * Alicia

Apellido(s): * Perez Castilla

E-mail: * aperez@manual

Rol: estudiante

Altras Actualizar

Figura 102. Interfaz edición de perfil de usuario

- *Gestión de Conversaciones*

En la pantalla de gestión de conversaciones se visualiza un listado con la información de las conversaciones establecidas.

Conversaciones

Mensaje de difusión Iniciar conversación 4

Busqueda nombre,apellidos ...

1	Nombre	Apellidos	Titulación	Msjs. Pendientes Coord.	Total Mensajes	Fecha primer mensaje	Fecha ultimo mensaje	Recibidos	5 6
	Manuel	Gomez	Grado en Ingeniería Informática	1	1	Sep 22, 2016 3:26:51 PM	Sep 22, 2016 3:26:51 PM	0	

First Previous 1 Next Last

Figura 103. Interfaz gestión conversaciones

En la figura anterior se presentan las siguientes opciones y características:

1. Listado de las conversaciones.
2. Mensaje de difusión: permite realizar el envío de un mensaje a todos los coordinadores registrados en la aplicación.
3. Iniciar conversación: Inicia una conversación con un coordinador.
4. Realizar búsquedas por apellidos, nombre ...
5. Acceder a la conversación.
6. Eliminar conversación.

- *Mensaje de Difusión*

A través de la interfaz que se muestra en la siguiente figura, se realiza el envío de un mensaje a todos los coordinadores.

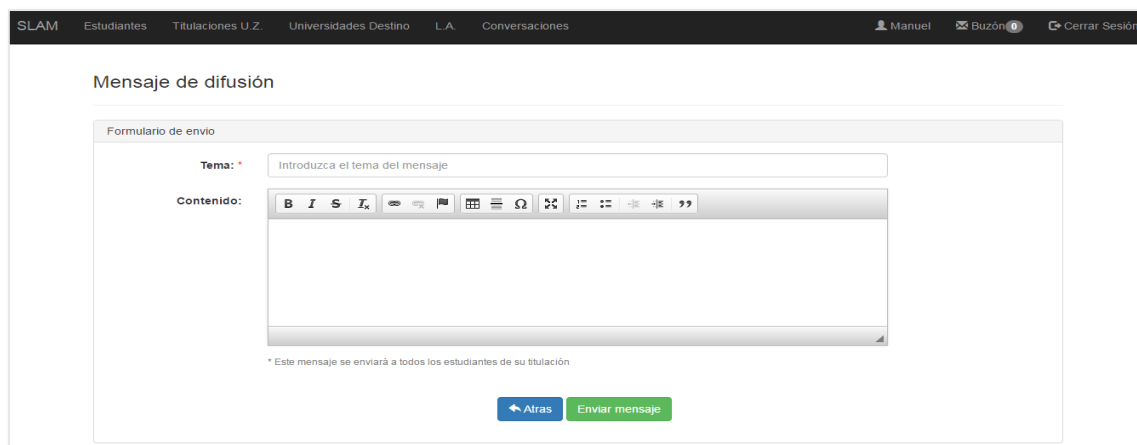


Figura 104. Interfaz envío mensaje de difusión

- *Iniciar una conversación*

A través de la interfaz que se muestra en la siguiente figura, se puede iniciar una conversación, en caso de que no exista, con un coordinador.

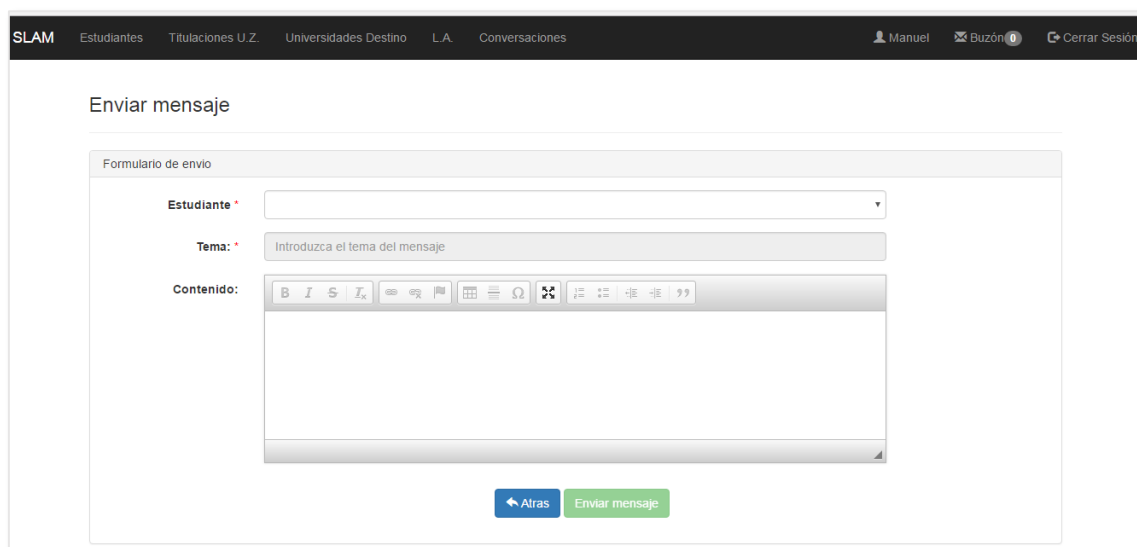


Figura 105. Interfaz inicio de conversación

Cuando se seleccione el coordinador al que desee enviar el mensaje automáticamente se habilitaran el resto de campos.

En caso de que seleccione un coordinador con una conversación ya en curso, se mostrara el siguiente mensaje:

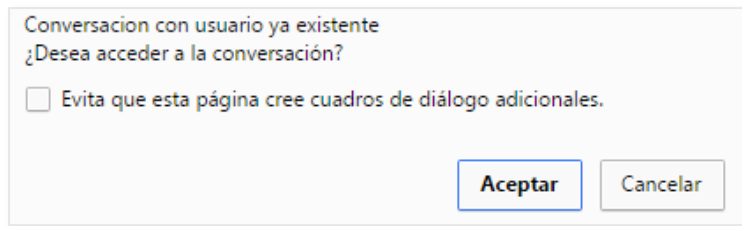


Figura 106. Mensaje conversación ya existente

- *Visualización y envío de mensajes*

A través de la interfaz que se muestra en la siguiente figura, se pueden visualizar los mensajes enviados y recibidos de la conversación, así como realizar el envío de un nuevo mensaje.

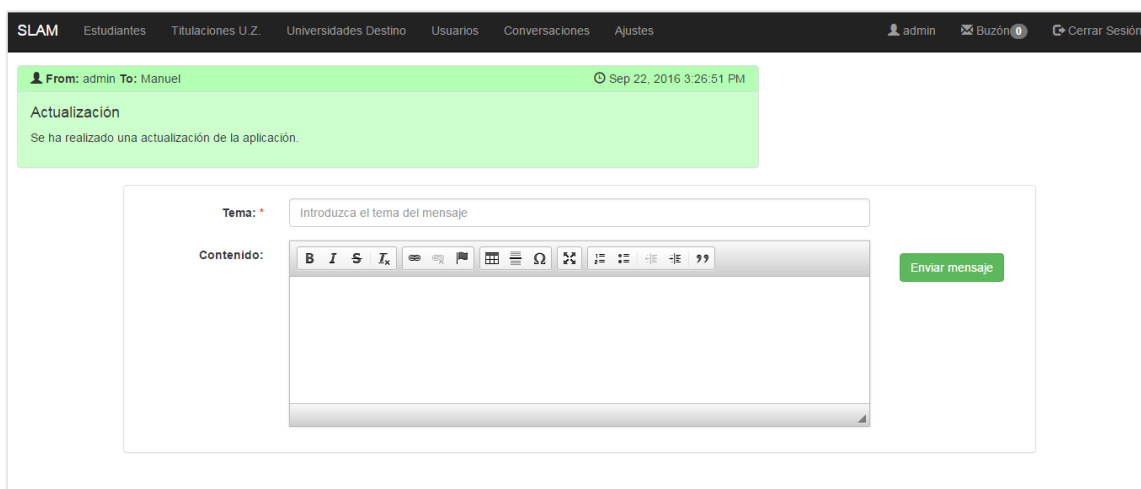


Figura 107 .Interfaz conversación

Los mensajes mostrados en color verde son los enviados por el administrador.

- *Ajustes*

En la pantalla de ajustes se visualizan las operaciones de carga automática de las titulaciones y asignaturas de la universidad de Zaragoza.

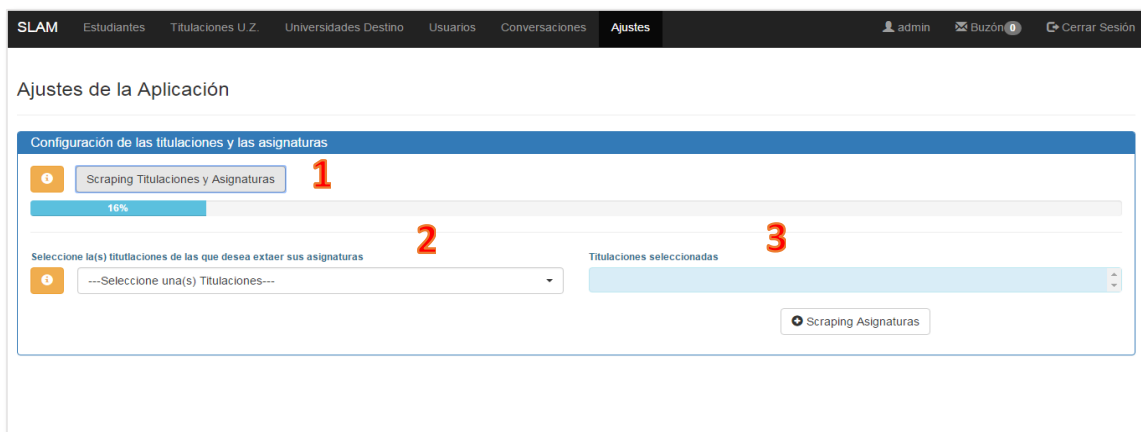


Figura 108. Interfaz ajustes

Las operaciones disponibles son las siguientes:

1. Scraping de titulaciones y asignaturas: cuando se pulsa sobre este botón, el sistema descarga y registra todas las titulaciones y asignaturas de la Universidad de Zaragoza. En caso de que la titulación o asignatura ya se encuentre en el sistema, actualizara su información.
2. Scraping de una titulación: posibilidad de seleccionar una o más titulaciones para actualizar su contenido y el contenido de sus asignaturas, descargando su información de la Universidad de Zaragoza.

8.2. Anexo II: Manual de Despliegue

Servidor

Para ejecutar la aplicación se necesita instalar el servidor web Node.js, que se puede descargar e instalar siguiendo los pasos descritos en su página web <https://nodejs.org/es/download/> y la base de datos MongoDB, que también se puede descargar e instalar a través de su página <https://docs.mongodb.com/>. En el desarrollo de este proyecto se ha usado la versión v4.5.0 de Node.js y la versión 3.2.3 de MongoDB.

Para descargar el código de la aplicación, es necesario acceder al repositorio de GitHub a través del enlace <https://github.com/DiegoGrandeAlagon/SLAM> y pulsar sobre el botón 'clone or download' situado en la parte superior derecha de la pantalla. Una vez se ha descargado el fichero y tras descomprimirlo, se podrá visualizar la carpeta /SLAM con el código de la aplicación.

Otra opción para obtener el código del programa es la de ejecutar el comando `git clone https://github.com/DiegoGrandeAlagon/SLAM.git` que descarga el contenido del repositorio, para ello imprescindible tener instalado Git.

Antes de ejecutar la aplicación se deben realizar dos modificaciones en el código:

- En el fichero `/SLAM/public/modules/core/config/routes_config.js` es necesario añadir la ip del servidor web a la variable declarada en la línea 7.
- En el fichero `/SLAM/public/modules/autenticación/controllers/autenticación_controller.js` en la línea 10 es necesario añadir la clave pública que genera Google al registrar el dominio del servidor para habilitar el recaptcha en las vistas de registro de usuario y recuperación de contraseña. Para obtener esta clave, se debe registrar el dominio del servidor en la dirección <https://www.google.com/recaptcha/intro/index.html> accesible a través de la cuenta de Gmail asociada a la aplicación.

Finalmente, para lanzar la aplicación SLAM únicamente es necesario posicionarse dentro de la carpeta del proyecto 'SLAM' y ejecutar el comando `node server.js`.

Cuando se ejecuta por primera vez la aplicación, el sistema crea un usuario administrador con las siguientes credenciales de acceso:

- Nombre de usuario: *admin*
- Password: *admin@slam*

Cliente

Para poder acceder a la aplicación a través del navegador web únicamente es necesario apuntar a la URL `http://hostName_server/` o `http://ip_server/` y se cargará la pantalla de Login.

8.3. Anexo III: Requisitos del Sistema y Casos de Uso (por perfil de usuario)

A continuación se presentan todos los requisitos de la aplicación y los casos de uso en función del rol que tiene cada usuario.

Requisitos funcionales **Estudiante**

Código	Descripción
RF-1	Deberá permitir darse de alta en la aplicación
RF-2	Deberá permitir autenticarse
RF-3	Deberá permitir restablecer la cuenta en caso de extraviar la contraseña
RF-4	Deberá permitir crear universidades destino
RF-5	Deberá permitir crear y editar asignaturas destino creadas por él.
RF-6	Deberá permitir gestionar su Acuerdo de Estudios
RF-7	Deberá permitir enviar y recibir mensajes
RF-8	Deberá permitir editar perfil

Requisitos no funcionales **Estudiante**

Código	Descripción
RFN-1	La interfaz deberá asistir al estudiante en el proceso de creación de su Acuerdo de Estudios
RFN-2	Podrá registrarse en la aplicación únicamente si el coordinador de la titulación a la que pertenece está también dado de alta en el sistema.
RFN-3	Solo permitirá enviar y recibir mensajes del coordinador de la titulación a la que pertenezca

Requisitos funcionales **Coordinador**

Código	Descripción
RF-1	Deberá permitir autenticarse
RF-2	Deberá permitir restablecer la cuenta en caso de extraviar la contraseña
RF-3	Deberá permitir crear y editar universidades destino
RF-4	Deberá permitir crear y editar asignaturas destino.
RF-5	Deberá permitir gestionar Acuerdos de Estudios
RF-6	Deberá permitir enviar y recibir mensajes
RF-7	Deberá permitir enviar mensajes de difusión a estudiantes
RF-8	Deberá permitir editar el perfil de usuario
RF-9	Deberá permitir dar de alta y editar cuenta de estudiantes
RF-10	Deberá permitir visualizar titulaciones U.Z.
RF-11	Deberá permitir crear y editar asignaturas de la titulación de la que sea coordinador
RF-12	Deberá permitir habilitar o deshabilitar cuenta de estudiantes

Requisitos no funcionales **Coordinador**

Código	Descripción
RFN-1	La interfaz deberá asistir al coordinador en el proceso de verificación y evaluación de los Acuerdo de Estudios
RFN-2	Únicamente podrá dar de alta nuevos coordinadores el administrador del sistema
RFN-3	Solo permitirá enviar y recibir mensajes de estudiantes que pertenezcan a las titulaciones gestionadas por él y al administrador.

Requisitos funcionales **Administrador**

Código	Descripción
RF-1	Deberá permitir autenticarse
RF-3	Deberá permitir crear, editar y borrar universidades destino
RF-4	Deberá permitir crear, editar y borrar asignaturas destino.
RF-5	Deberá permitir enviar y recibir mensajes de coordinadores
RF-6	Deberá permitir enviar mensajes de difusión a coordinadores
RF-8	Deberá permitir dar de alta y editar cuentas de usuario (coordinador/estudiante)
RF-9	Deberá permitir crear, editar y borrar titulaciones U.Z.
RF-10	Deberá permitir crear, editar y borrar asignaturas de la titulación
RF-11	Deberá permitir habilitar o deshabilitar cuenta de usuario
RF-12	Deberá permitir realizar web Scraping de la titulaciones y asignaturas de la Universidad de Zaragoza

Requisitos no funcionales **Administrador**

Código	Descripción
RFN-1	Solo permitirá enviar y recibir mensajes de coordinadores.

Casos de Uso

A continuación se presentan los diagramas de caso de uso en el que se especifica de forma detallada el comportamiento del sistema en función del rol que tiene cada actor.

Diagrama de casos de uso con rol estudiante

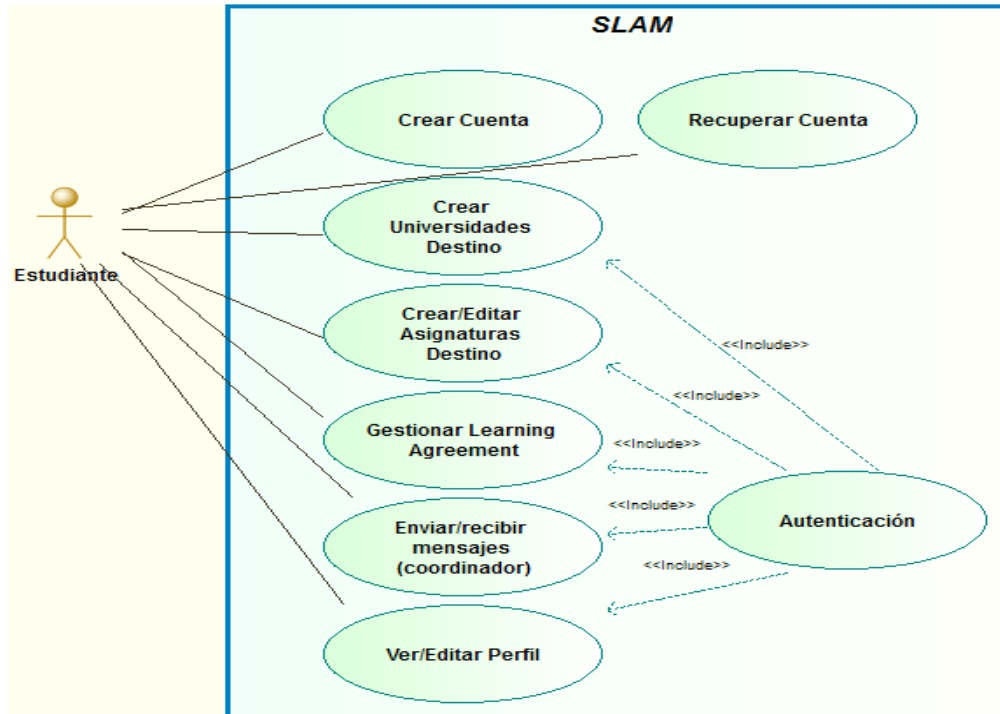


Figura 109. Casos de uso estudiante

Existe un único actor que tiene el rol de estudiante

- **Actor Estudiante:** actor primario que interactúa con la aplicación para realizar las diferentes funcionalidades contenidas en los casos de uso

A continuación se detallan los casos de uso:

- **Crear cuenta:** el caso de uso permite al estudiante darse de alta en la aplicación.
- **Recuperar cuenta:** el caso de uso permite al estudiante restablecer la contraseña de acceso.
- **Autenticación:** el caso de uso permite al estudiante acceder con sus credenciales a la aplicación.
- **Crear universidad destino:** el caso de uso permite al estudiante registrar nuevas universidades de destino. Debe estar autenticado en la aplicación.
- **Crear/editar asignaturas destino:** el caso de uso permite al estudiante crear y editar asignaturas de universidades destino siempre que las haya creado él. Debe estar autenticado en la aplicación.
- **Gestionar learning agreement:** el caso de uso permite gestionar las distintas etapas del Acuerdo de Estudios. Debe estar autenticado en la aplicación.
- **Enviar/recibir mensajes:** el caso de uso permite enviar y recibir mensajes del coordinador de su titulación. Debe estar autenticado en la aplicación.

- **Ver/editar perfil:** el caso de uso permite visualizar y editar los datos de perfil. Debe estar autenticado en la aplicación.

Diagrama de casos de uso con rol coordinador

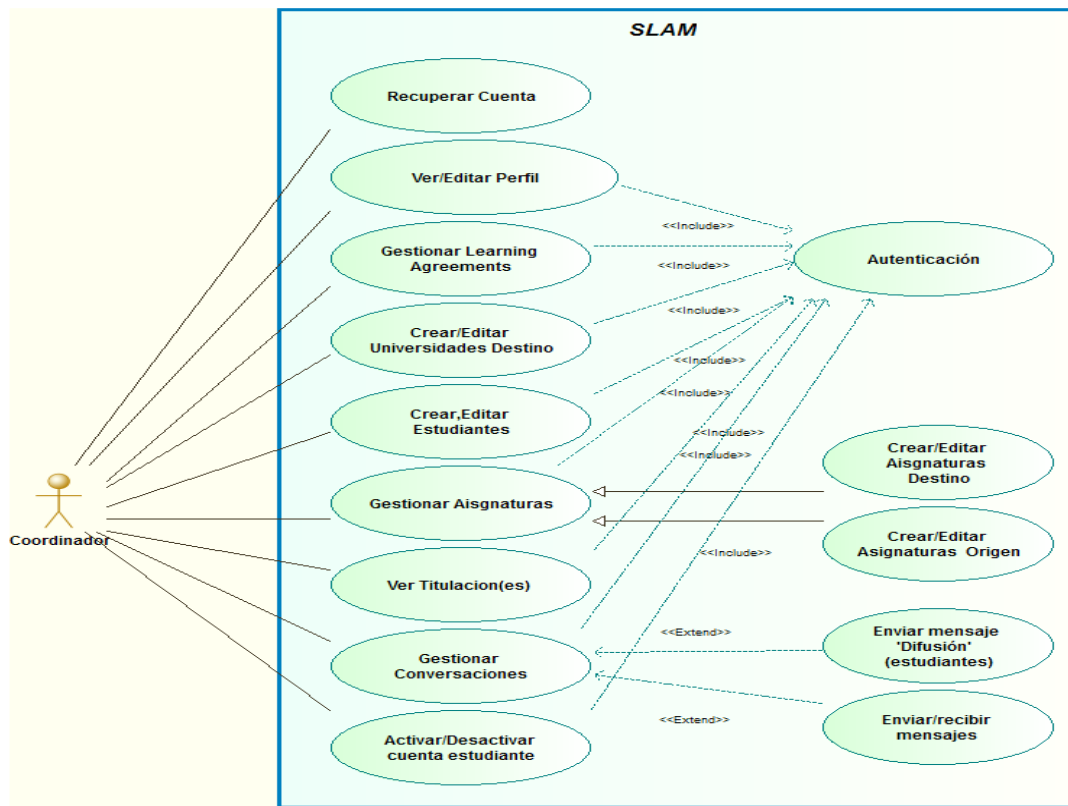


Figura 110. Casos de uso coordinador

Existe un único actor que tiene el rol de coordinador

- **Actor coordinador:** actor primario que interactúa con la aplicación para realizar las diferentes funcionalidades contenidas en los casos de uso

A continuación se detallan los casos de uso:

- **Recuperar cuenta:** el caso de uso permite al coordinador restablecer la contraseña de acceso.
- **Autenticación:** el caso de uso permite al coordinador acceder con sus credenciales a la aplicación.
- **Ver/editar perfil:** el caso de uso permite visualizar y editar los datos de perfil. Debe estar autenticado en la aplicación.
- **Gestionar learning agreement:** el caso de uso permite gestionar las distintas etapas del Acuerdo de Estudios de los estudiantes registrados en su titulación. Debe estar autenticado en la aplicación.
- **Crear/editar universidades destino:** el caso de uso permite al coordinador crear y editar universidades destino. Debe estar autenticado en la aplicación.
- **Crear/editar estudiantes:** el caso de uso permite al coordinador dar de alta y editar a estudiantes. Debe estar autenticado en la aplicación.

- **Gestionar asignaturas:** el caso de uso es permite realizar la gestión de asignaturas. Debe estar autenticado en la aplicación. Se compone de los casos de uso hijos:
 - o **Crear/editar asignaturas destino:** el caso de uso permite al coordinador crear y editar asignaturas de universidades destino siempre que las haya creado él.
 - o **Crear/editar asignaturas origen:** el caso de uso permite al coordinador crear y editar asignaturas de universidades de Zaragoza.
- **Visualizar titulaciones:** el caso de uso permite al coordinador visualizar los datos de las titulaciones. Debe estar autenticado en la aplicación.
- **Activar/desactivar cuenta estudiante:** el caso de uso permite al coordinador habilitar o deshabilitar las cuentas de los estudiantes. Debe estar autenticado en la aplicación.
- **Gestionar conversaciones:** el caso de uso es permite realizar la gestión de las conversaciones. Debe estar autenticado en la aplicación. De este caso de uso se extienden los siguientes:
 - o **Enviar mensaje de difusión:** el caso de uso permite al coordinador enviar mensajes de difusión a sus estudiantes.
 - o **Enviar/recibir mensajes:** el caso de uso permite enviar y recibir mensajes con los estudiantes y con el administrador

Diagrama de casos de uso con rol administrador

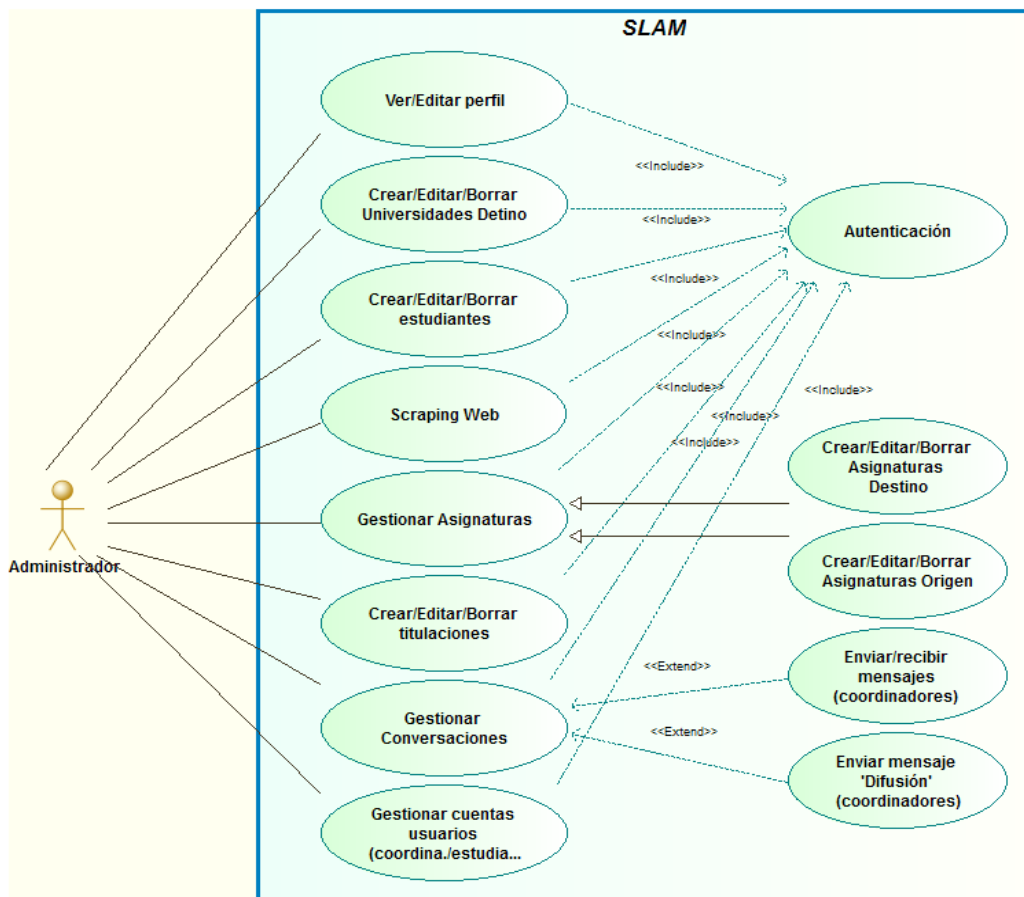


Figura 111. Casos de uso administrador

Existe un único actor que tiene el rol de administrador

- **Actor administrador:** actor primario que interactúa con la aplicación para realizar las diferentes funcionalidades contenidas en los casos de uso

A continuación se detallan los casos de uso:

- **Autenticación:** el caso de uso permite al administrador acceder con sus credenciales a la aplicación.
- **Ver/editar perfil:** el caso de uso permite visualizar y editar los datos del perfil. Debe estar autenticado en la aplicación.
- **Crear/editar/borrar universidades destino:** el caso de uso permite al administrador crear, editar y borrar universidades destino. Debe estar autenticado en la aplicación.
- **Crear/editar/borrar estudiantes:** el caso de uso permite al administrador dar de alta, editar y borrar a estudiantes. Debe estar autenticado en la aplicación.
- **Scraping Web:** el caso de uso permite al administrador descargar y almacenar la información de las titulaciones y asignaturas de la universidad de Zaragoza.
- **Gestionar asignaturas:** el caso de uso es permite realizar la gestión de asignaturas. Debe estar autenticado en la aplicación. Se compone de los casos de uso hijos:
 - o **Crear/editar/borrar asignaturas destino:** el caso de uso permite al administrador crear, editar y borrar asignaturas de universidades destino.
 - o **Crear/editar/borrar asignaturas origen:** el caso de uso permite al administrador crear, editar y borrar asignaturas de universidades de Zaragoza.
- **Crear/editar/borrar titulaciones:** el caso de uso permite al administrador crear, editar y borrar los datos de las titulaciones. Debe estar autenticado en la aplicación.
- **Gestionar conversaciones:** el caso de uso es permite realizar la gestión de las conversaciones. Debe estar autenticado en la aplicación. De este caso de uso se extienden los siguientes:
 - o **Enviar mensaje de difusión:** el caso de uso permite al administrador enviar mensajes de difusión a los coordinadores.
 - o **Enviar/recibir mensajes:** el caso de uso permite enviar y recibir mensajes con los coordinadores.
- **Gestionar cuentas de usuarios y estudiantes** el caso de uso permite al administrador gestionar las cuentas de los usuarios registrados en la aplicación. Debe estar autenticado en la aplicación.

8.4. Anexo IV: Detalles de diseño e implementación

En la siguiente figura se presenta el diagrama de arquitecturas detallando las tecnologías empleadas y el patrón MVC tanto en cliente como en el servidor.

Servidor

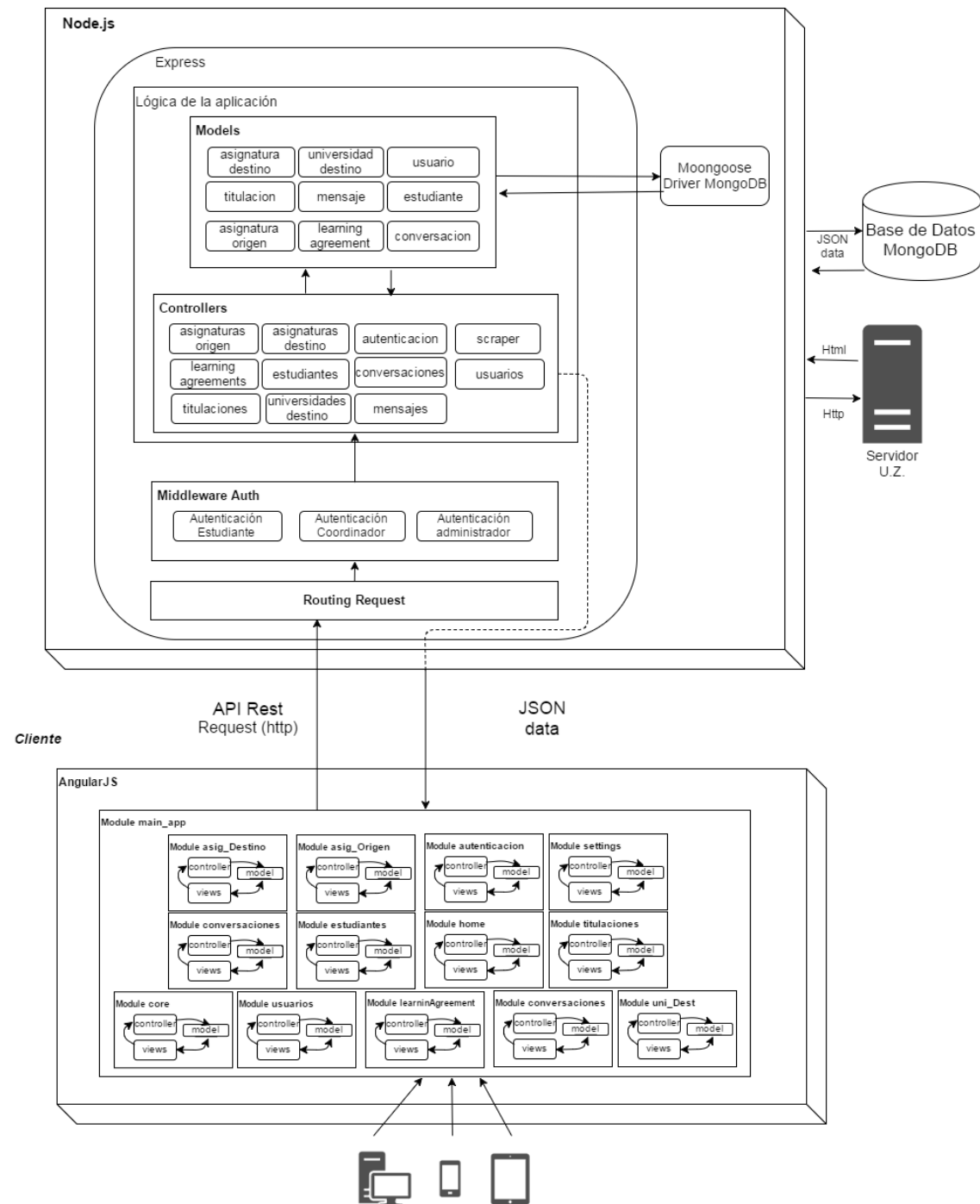


Figura 112. Diagrama arquitectura detallada

A continuación se va a analizar cómo están diseñadas e implementadas las principales características y funcionalidades del sistema.

Servidor web: API RESTful y MongoDB

- Estructura de directorios:

La estructura de directorios implantada en el servidor tiene como objetivo realizar una programación modular siguiendo el patrón Modelo-Vista-Controlador, permitiendo ordenar y definir de forma clara las operaciones que se realizan en los módulos que forman cada recurso de la API. Dicha estructura se compone de los siguientes directorios:

- **/config:** contiene los ficheros donde se declaran las variables de configuración de la aplicación. En el primero se almacena la URL desde donde se descarga la información de la Universidad de Zaragoza y en el segundo la clave para firmar el Token de autenticación.
 - **/controllers:** ficheros donde se declaran las funciones que ejecutan operaciones sobre los recursos solicitados y donde se responde a dichas solicitudes. Cada recurso tiene su propio controlador.
 - **/middleware:** contiene el fichero donde se analiza cada solicitud con el fin de verificar que el token se encuentra en la cabecera de la solicitud, su estado es el correcto y que posee los privilegios suficientes para acceder al recurso.
 - **/models:** contiene los ficheros donde se realiza el modelado de los datos a través de la librería Mongoose, en cada fichero se declara el esquema que tendrán las colecciones en la base de datos, además de exportar su modelo.
 - **/routes:** contiene los ficheros donde se realiza el enrutamiento de las peticiones que recibe la API. Cada fichero monta las rutas asociadas a un recurso.
 - **/services:** contiene los ficheros donde se declaran las funciones para la creación de Tokens, envío de emails y generación de hash.
- Detalles de implementación de las principales funcionalidades del sistema:
- *Web Scraping:*

Para desarrollar la funcionalidad de la aplicación que extrae los datos de las titulaciones y asignaturas de la Universidad de Zaragoza se ha utilizado la librería Cheerio, un módulo de Node.js que implementa una serie de funciones para localizar y extraer información de ficheros HTML.

En este caso se ha realizado una petición a la página web de la universidad '<http://titulaciones.unizar.es/listado.php?tipo=1>' para obtener el listado de las titulaciones y guardar su nombre, universidad y URL en la base de datos.

Para obtener los datos de las asignaturas se ha procedido de la misma manera pero en este caso las peticiones se han realizado a las URLs formadas por la URL de la titulación + '/cuadro_asignaturas.html'. De esta forma se extraen automáticamente los datos sin necesidad de registrar manualmente cada URL.

En los modelos de datos de las titulaciones y las asignaturas se ha declarado un campo Booleano 'ofertada', el cual antes de realizar la operación de scraping se pone a 'false' en las asignaturas y titulaciones ya guardadas en la base de datos (si las hay) y tras realizar la extracción de datos se pone a 'true' en las nuevas asignaturas y titulaciones

guardadas, de esta forma se pueden distinguir las que se ofertan en el curso actual de aquellas que han dejado de hacerlo.

- *Autenticación por tokens*

La autenticación por tokens permite verificar en cada una de las peticiones a la API la identidad del usuario que la realiza, sin necesidad de almacenar en el servidor ninguna información de estado asociada al usuario.

Se ha utilizado la librería jwt-simple de Node.js que permite a través de tokens controlar el acceso de los usuarios a los recursos del sistema.

Cada token se compone de tres partes:

Header: formado por el tipo y el algoritmo utilizado para la firma digital

Payload: formado por el identificador de usuario que ha accedido al sistema, las fechas de creación y caducidad del token y el rol del usuario.

Signature: firma digital que se utiliza para comprobar que el token no ha sido alterado y que se ha emitido en el servidor. La firma está formada por la función hash de los anteriores componentes (Header y Payload) codificados en Base64 y la clave secreta almacenada en uno de los ficheros de la carpeta /conf. Con estos datos se calcula la clave HMAC-SHA-256.

A continuación se muestra un fragmento de código de la función del middleware donde se comprueba si la petición que le ha llegado a la API contiene el token, si el usuario existe en la BBDD y si tiene los privilegios suficientes para realizar la petición (En este caso administrador).

```

//Comprueba que el token es correcto y que solo se puede continuar con la
//ejecucion de la siguiente funcion si el rol es administrador
exports.authLevelAdmin = function(req, res, next) {
  if (!req.headers.authorization) {
    return res
      .status(403)
      .send({ message: "Tu petición no tiene cabecera de
autorización" });
  }

  var token = req.headers.authorization.split(" ")[1];
  var payload = jwt.decode(token, config.TOKEN_SECRET);

  if (payload.exp <= moment().unix()) {
    return res
      .status(401)
      .send({ message: "El token ha expirado" });
  }
  //Comprobacion si existe el usuario y si el rol es el correcto
  verifyUser(payload.sub, function(econtrado) {
    if (econtrado) {
      if (payload.rol === 'administrador') {
        req.rol = payload.rol;
        next();
      } else {
        return res.status(401).send({ message: "No tiene permisos"
});
      }
    } else {
      return res.status(401).send({ message: "Usuario no encontrado"
});
    }
  });
}

```

Fragmento de código 2. Función middleware autenticación

- *Módulo de comunicación*

Este módulo tiene como finalidad implementar un buzón de mensajes a través del cual los usuarios de la aplicación puedan comunicarse. Para ello se han declarado los modelos de datos ‘conversaciones’ y ‘mensajes’ con sus respectivos controladores.

En el controlador de mensajes se han declarado las funciones ‘newMessag()’ y ‘sendMulticast()’. La primera permite el envío de mensajes entre usuario de forma individual, creando una conversación entre pares, en caso de no existir, o añadiendo el mensaje a la conversación existente. La segunda función permite a los coordinadores enviar un mensaje de difusión a todos los estudiantes que pertenecen a su titulación, o en caso de que el remitente sea el administrador de la aplicación, enviará un mensaje de difusión a todos los coordinadores.

- *Módulo de Gestión de Learning Agreements*

El módulo de learning agreements está formado por su modelo de datos y el controlador donde se declaran todas las operaciones que se pueden realizar sobre él.

En el esquema del modelo de datos se ha declarado un campo llamado *'asignaturas'*, donde se almacenan las asignaturas a convalidar y aquellas cursadas en la universidad de destino. A continuación se puede visualizar su estructura:

```
asignaturas: [{
  origen: [{
    asig_data: {
      type: mongoose.Schema.Types.ObjectId,
      ref: 'AsignaturaOrg',
    },
    nota: { type: String, },
    matriculaciones: { type: Number, },
  }],
  destino: [{
    asig_data: {
      type: mongoose.Schema.Types.ObjectId,
      ref: 'AsignaturaDest',
    },
    nota: { type: String, },
    ectsAplicado: { type: Number, }
  }],
  destinoReasigECTS: [{
    asig_data: {
      type: mongoose.Schema.Types.ObjectId,
      ref: 'AsignaturaDest',
    },
    nota: { type: String, },
    ectsAplicado: { type: Number, }
  }],
  revision: {
    type: String,
    enum: ['pendiente', 'valida', 'rechazada'],
    required: true,
    default: 'pendiente'
  },
  observacionesRevision: { type: String, default: '' },
  bloque: { type: String, enum: ['OB', 'OP', 'LE', 'TFG'], },
  ectsConv: { type: Number }
}],
```

Fragmento de código 3. Bloque Asignaturas (Modelo Learning Agreement)

Como se puede observar en el fragmento de código anterior, está formado por un array de bloques de asignatura(s) a convalidar en la Universidad de Zaragoza. Cada bloque se compone a su vez de:

- Origen: array con la asignatura(s) de la universidad de Zaragoza (a convalidar)
- Destino: array con las asignatura(s) destino
- DestinoReasigECTS: array con las asignaturas(s) destino reasignadas de otros bloques

Además en cada uno de los bloques se declara el campo observación, el tipo y los créditos totales de las asignaturas de destino.

Cliente: AngularJS

- Estructura de directorios:

La estructura de directorios que tiene cada módulo en AngularJS es la siguiente:

- **/assets:** contiene imágenes, CSS y librerías específicas del módulo.
- **/config:** contiene el fichero donde se configuran las rutas de las peticiones.
- **/controllers:** contiene el fichero donde se implementa la funcionalidad asociada a las vistas, gestionando los eventos, inicializando variables....
- **/services:** directorio que contiene los ficheros donde se declaran las factorías y los servicios, estos permiten almacenar datos que serán accesibles desde distintos controladores.
- **/views:** contiene las plantillas HTML donde se cargara la información.

- Detalles de implementación:

Para el desarrollo de las vistas, se ha modificado el estilo CSS de los documentos HTML, ya que aunque Bootstrap permite hacer un diseño de las interfaces adaptativo al tamaño de pantalla, en varias ocasiones se han tenido que corregir las posiciones de los elementos y retocar sus estilos. Un ejemplo de ello son los iconos que se muestran en la ventana de inicio, los cuales son componentes de que provee Bootstrap y que han sido modificados.

Para poder dar formato a los mensajes enviados entre usuarios se ha añadido el editor de texto CKEditor al módulo de Conversaciones.

Cuando el coordinador ha confirmado la validez del acuerdo de estudios o se han insertado las calificaciones, se habilita la opción de descargar el documento en formato PDF, para ello se ha utilizado la librería jsPDF de Node.js.

Durante todo el proceso de creación del acuerdo de estudios, en el controlador `learningAgreements_controller.js` es donde se realizan todas las operaciones para asegurar la correcta propuesta del acuerdo. El usuario recibe continuamente 'ayudas' a través de alertas, o deshabilitando elementos cuando ya no están disponibles.

Cada vez que se añade un bloque de asignaturas al acuerdo de estudios, automáticamente el sistema lo guarda en la base de datos, permitiendo al estudiante crear su acuerdo de estudios sin perder la información en caso de que salga del sistema o acceda a otra sección de la aplicación.

8.5. Anexo V: Pruebas

A continuación se presentan las distintas pruebas realizadas en el servidor durante el desarrollo de la API y las pruebas realizadas en el cliente.

Pruebas Servidor

Prueba	Código respuesta	Mensaje servidor
Solicitar un recurso sin cabecera de autorización (Token)	403	Tu petición no tiene cabecera de autorización
Solicitar un recurso sin privilegios suficientes	401	No tiene permisos para acceder a esta sección
Solicitar un recurso con privilegios	200	JSON con la información del recurso solicitado
Solicitar una operación sobre un recurso existente (listar, borrar, actualizar ...)	200	JSON con la información del recurso solicitado
Solicitar una operación sobre un recurso no existente	404	Recurso no existente
Solicitar una operación sobre un recurso con un id no valido	400	Error al intentar encontrar el recurso
Actualizar la información de un usuario o estudiante cuyo nombre de usuario ya está registrado en la BD	404	Nombre de Usuario (login) ya registrado
Registrar un nuevo recurso (titulación, estudiante ...)	200	JSON con la información del recurso creado
Registrar un usuario o estudiante ya registrado en la BD	404	Nombre de Usuario, NIA o e-mail ya registrados
Registrar una (titulación/ asignaturaOrg/ UniversidadDest/ asignaturaDest) ya existente	404	(titulación/ asignaturaOrg/ UniversidadDest/ asignaturaDest) ya existente

Pruebas Cliente

Para la presentación de las pruebas en el cliente se han seleccionado ejemplos representativos en el caso de formularios.

- Prueba 1. Acceso con credenciales incorrectas

En caso de intentar acceder a la aplicación con un nombre de usuario o contraseña incorrectos se muestra el siguiente mensaje de error.

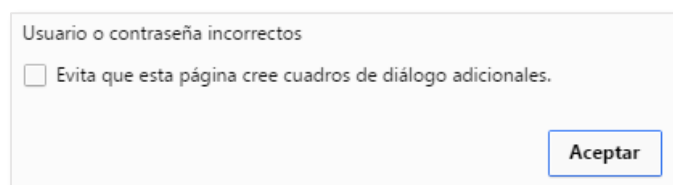


Figura 113. Error de autenticación

- Prueba 2. Cumplimentación errónea de formularios

En la figura siguiente se puede observar como el sistema notifica al usuario las irregularidades encontradas en el formulario.

Nombre Usuario (login): * Oscar_11 * Campo no valido.
Cadena alfanumerica sin espacios con un tamaño máximo de 10 caract.

NIA: * Introduzca el NIA * Campo obligatorio.

Apellido(s): Grande

Nombre: Oscar

NIF: 123423434d

Curso: 2º

Imagen de perfil: DSC03602.JPG

E-mail: * osca@ * Formato incorrecto.

Titulación: Grado en Arquitectura Técnica

* La contraseña se generará automáticamente en el servidor y será enviada a la dirección de correo electrónico facilitada

Atras Crear Cuenta

Figura 114. Formulario de registro incorrecto

Errores notificados: campo no valido, campo obligatorio y formato incorrecto.

En caso de que el formulario sea incorrecto no permite su envío.

- Prueba 3. Registro de estudiante ya existente en el sistema.

En caso de que se registre un nuevo estudiante y el nombre de usuario, NIA o e-mail ya estén en uso se informara del error al estudiante.

Nombre de usuario o estudiante ya existente

Evita que esta página cree cuadros de diálogo adicionales.

Aceptar

Figura 115. Error estudiante y existente

De forma análoga sucederá cuando edite su perfil de estudiante.

- Prueba 4. Edición de asignaturas Destino

Los estudiantes únicamente pueden editar la información asociada a las asignaturas de destino que ellos han creado. En caso de intentar editar una asignatura creada por otro alumno o coordinador se mostrara el siguiente mensaje:

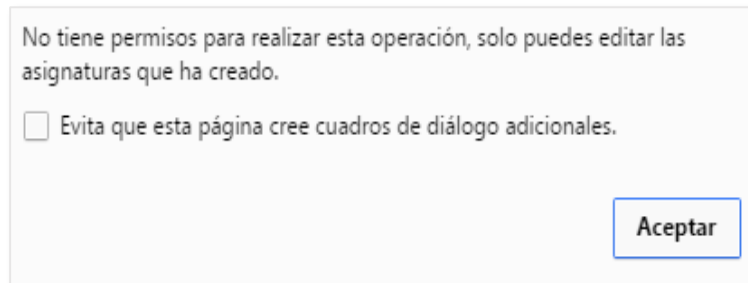


Figura 116. Mensaje error de permisos

- Prueba 5. Reasignación de créditos

En caso de que el estudiante intente reasignar a otro bloque un número de créditos sobrantes erróneo, se le informará por pantalla y no le permitirá realizar la operación.

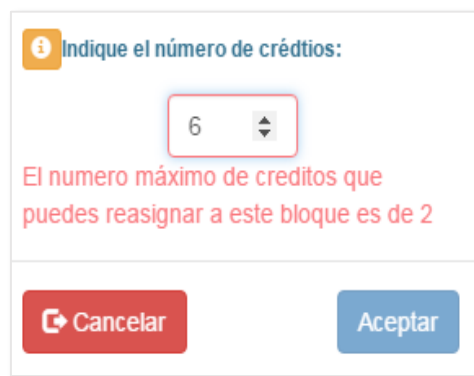


Figura 117. Error de reasignación de créditos

- Prueba 6. Envío del Acuerdos de Estudio

El Acuerdo de estudios es verificado antes de enviarlo al coordinador, mostrando un error en caso de error o habilitando el botón de envío en caso contrario.

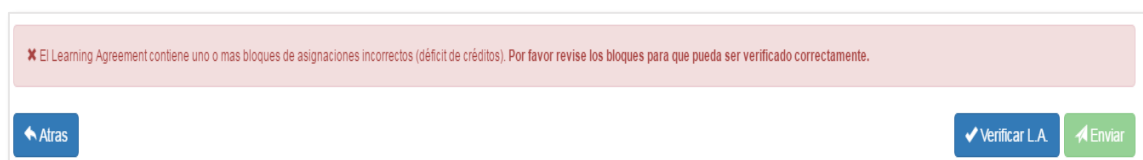


Figura 118. Mensaje error en el Acuerdo de Estudios

- Prueba 7. Revisión de los bloques que forman el Acuerdo de Estudios

En caso de existan bloque sin revisar se informara al coordinador con el siguiente mensaje de error:

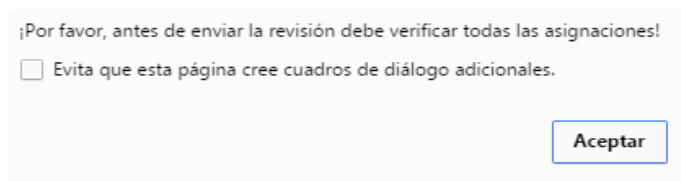


Figura 119. Error revisión de bloques

- Prueba 8. Calificación de asignaturas

En caso de que el coordinador olvide asignar alguna calificación cuando realiza el registro de notas se le informara con el siguiente mensaje:

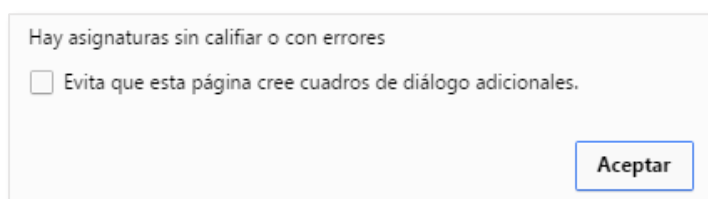


Figura 120. Error calificaciones

- Prueba 9. Inserción de una nueva asignatura Origen

Un coordinador únicamente podrá insertar una nueva asignatura en una titulación en el caso de que sea el coordinador de la misma. En caso contrario muestra el siguiente mensaje:

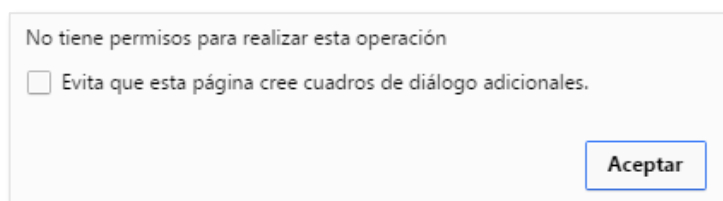


Figura 121. Error insertando asignatura

- Prueba 10. Iniciar conversación ya existente

Si un coordinador o un administrador intentan iniciar una conversación con un usuario y ésta ya existe, el sistema informara del error con el siguiente mensaje:

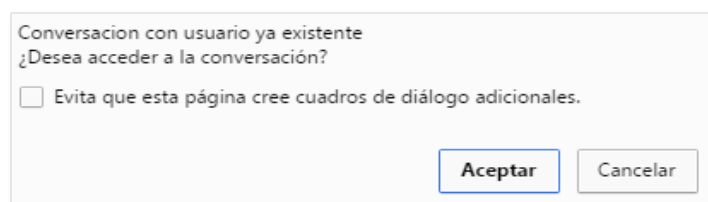


Figura 122. Error creando conversación

Para finalizar, se han realizado pruebas con datos de estudiantes que han participado con anterioridad en los programas de movilidad y se han simulado las distintas etapas que deben atravesar sus acuerdos de estudios hasta que son finalizados.