UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Facultat d'Informàtica de Barcelona

Plataforma informativa sobre el rendimiento académico del estudiante

Grado en ingeniería Informática
Tecnologías de la información

AUTOR

ANGEL RUBIÑO FERNANDEZ

DIRECTORES

SILVIA LLORENTE VIEJO
MARC ALIER FORMENT

30 de Junio de 2017

RESUMEN

Hoy en día disponemos de muchas herramientas para dar soporte y facilitar el aprendizaje de los estudiantes, ya sea en educación obligatoria o en estudios posteriores. Algunas de estas novedades han sido la utilización de pizarras electrónicas, complementar los libros de textos con el uso de tabletas o portátiles, la creación de espacios virtuales que permiten que se impartan las lecciones a distancia, habilitar foros donde los alumnos pueden interactuar entre ellos, agendas electrónicas, etc.

Un aspecto muy importante en la educación consiste en guiar y procurar que el alumno asimile los conceptos esperados, intentando ofrecerle todo el soporte posible y solucionando los problemas que puedan surgirle. En estudios obligatorios (educación primaria y ESO) esta práctica es más sencilla de llevar a cabo, ya que el número de alumnos por curso oscila entre los 25-30 y se puede hacer una tutoría individual suficiente. En otro tipo de cursos el número de participantes es mayor, en la fase inicial universitaria el número de estudiantes por clase asciende hasta los 60-80, dificultando la tutoría. Esto no solo afecta a estudios universitarios, en la educación a distancia se encuentra la misma problemática, ya que resulta más complejo interaccionar y empatizar con los alumnos.

Las instituciones docentes y su profesorado disponen de un amplio abanico de herramientas analíticas que permiten recolectar información de los alumnos y su comportamiento, siendo así capaces de adaptar el contenido del curso o enfatizar un temario en consecuencia, pero a pesar de esto, el flujo de información y soporte ofrecido al alumno no es suficiente en algunos casos.

Resulta evidente la necesidad de ciertas medidas para mejorar el soporte a la metodología de estudio del alumno. Este proyecto pretende expandir la funcionalidad ofrecida por los sistemas de gestión de aprendizaje (*Learning management systems*) dotándolos de capacidad analítica a partir de la información estadística generada por estas. Para hacer esto posible, vamos a utilizar diferentes algoritmos basados en inteligencia artificial, concretamente perteneciente al subcampo conocido como Machine Learning.

El resultado del proyecto muestra la viabilidad de la aplicación de esta tecnología en la docencia, obteniendo unos resultados prometedores y con información realmente útil para el alumno.

RESUM

Avui en dia disposem de moltes eines destinades a oferir suport i facilitar el procés d'aprenentatge del estudiants, tant en l'educació obligatòria com en estudis posteriors. Algunes d'aquestes millores son la utilització de pissarres electròniques, complementar els llibres de text amb l'ús de tauletes o portàtils, la creació d'espais virtuals que permetin impartir de les classes a distancia, habilitar fòrums on els estudiants puguin interactuar entre ells, agendes electròniques, etc.

Un aspecte molt important en l'educació consisteix en guiar a l'alumne i procurar que assimili tots els conceptes esperats, intentant oferir tot el suport possible així com solucionant els problemes que puguin sorgir-li. En estudis obligatoris (educació primària i ESO), aquesta es mes senzilla, ja que el nombre d'alumnes per classe oscil·la entre els 25-30, i es pot fer una tutoria individual suficient. En altre tipus de cursos el nombre de participants es major, a la fase inicial universitària, l'ocupació dels grups ascendeix fins el 60-80, dificultant la tutoria. Aquest problema no afecta tan sols a las universitats, en l'educació a distancia trobem la mateixa problemàtica, ja que resulta més complex interaccionar amb els alumnes.

Les institucions docents i el seu professorat disposen d'un ampli ventall d'eines analítiques al seu abast que permeten recol·lectar informació dels alumnes i el seu comportament, podent així adaptar el contingut del curs o emfatitzar un temari en conseqüència. A pesar d'aquestes solucions existents, el flux d'informació i suport ofert als alumnes no es suficients en alguns casos.

Resulta evident la necessitat de certes mesures per millorar el suport a la metodologia d'estudi de l'alumne. Aquest projecte pretén expandir les funcionalitats ofertes pels sistemes de gestió de l'aprenentatge (*Learning management Systems*) afegint-li capacitats analítiques a partir de la informació estadística generada per aquests. Per fer això possible, s'utilitzaran diversos algorismes basats en intel·ligència artificial, concretament del subcamp conegut com Machine Learning.

El resultat del projecte mostra la viabilitat de l'aplicació d'aquesta tecnologia en la docència, obtenint un resultats prometedors juntament amb informació de veritable utilitat pels estudiants.

ABSTRACT

Nowadays there are a lot of tools whose aim is to support the student's learning process. Some of these features are the use of electronic blackboards, the combination of books with laptops or tablets, virtual environments to help non presential courses, the use of forums where the students can discuss different course topics, online scheduling, etc.

One of the main aspects to consider when teaching, is to support the student offering him guidance and help him with the possible eventualities. This guidance is easier to do in the compulsory school, because the number of students per class goes from 25 to 30, and you can dedicate enough time on each student. That situation is much different in the university, where in the first year, the enrolled students increases twice as much as in school, making it very difficult to keep track of his progress. These difficulties are even worst for the distance courses, where it is more complex to empathize with the student.

The educational institutions have at their disposal a wide variety of analytics tools that allow them to collect student's performance data and use them to adapt the course content in consequence, for example, emphasizing the topics where the performance has been lower. Even though the existence of these tools, the communication with the student is not enough.

The aim of this project is to increase the coverage offered by the learning management systems, providing them with analytical functionalities that will be applied to the statistical information generated by these platforms. To make this possible, I'm going to use different techniques based on artificial intelligence, specifically, a variety of algorithms studied by the machine learning field.

The results obtained by the project have proven the viability of using this technology in the education. The information generated by the investigation is truly helpful for the student and opens the door to an entire range of possible future improvements.

Índice de contenidos

Índice de tablas

Índice de figuras

1. Formulación del problema

Hoy en día disponemos de muchas herramientas para dar soporte y facilitar el aprendizaje de los estudiantes, ya sea en educación obligatoria o en estudios posteriores. Algunas de estas novedades han sido la utilización de pizarras electrónicas, complementar los libros de textos con el uso de tabletas o portátiles, la creación de espacios virtuales que permiten que se impartan las lecciones a distancia, habilitar foros donde los alumnos pueden interactuar entre ellos, agendas electrónicas, etc.

Un aspecto muy importante en la educación consiste en guiar y procurar que el alumno asimile los conceptos esperados, intentando ofrecerle todo el soporte posible y solucionando los problemas que puedan surgirle. En estudios obligatorios (educación primaria y ESO) esta práctica es más sencilla de llevar a cabo, ya que el número de alumnos por curso oscila entre los 25-30 y se puede hacer una tutoría individual suficiente. En otro tipo de cursos el número de participantes es mayor, en la fase inicial universitaria el número de estudiantes por clase asciende hasta los 60-80, dificultando la tutoría. Esto no solo afecta a estudios universitarios, en la educación a distancia se encuentra la misma problemática ya que resulta más complejo interaccionar y empatizar con los alumnos.

Volviendo a la fase universitaria, según un estudio llevado a cabo por el Ministerio de Educación, Cultura y Deportes, uno de cada cinco universitarios (19%) abandonan los estudios en su primer año de carrera, siendo tan solo un 7,1% los que cambian de carrera. Se han realizado diversas investigaciones sobre las causas de abandono universitario[1], en muchas de ellas, se señala como un factor importante las malas estrategias y actividades de estudio llevadas a cabo por parte del estudiante.

Las instituciones docentes y su profesorado disponen de un amplio abanico de herramientas analíticas que permiten recolectar información de los alumnos y su comportamiento, siendo así capaces de adaptar el contenido del curso o enfatizar un temario en consecuencia, pero a pesar de esto, el flujo de información y soporte ofrecido al alumno no es suficiente en algunos casos.

Así como los docentes disponen de estos recursos suplementarios, por parte del alumno no se encuentra ninguna herramienta similar que poder utilizar como ayuda, la cual pueda ir informando al alumno de su nivel de rendimiento actual, temarios o actividades donde es necesario poner especial énfasis e incluso alertas sobre el estado de las diferentes materias.

Resulta evidente la necesidad de ciertas medidas para mejorar el soporte a la metodología de estudio del alumno. Este proyecto pretende expandir la funcionalidad ofrecida por los sistemas de gestión de aprendizaje (Learning management system) dotándolos de capacidad analítica a partir de la información estadística generada por estas.

Los sistemas de gestión de aprendizaje, a los cuales me voy a referir a partir de ahora como LMS, son espacios virtuales orientados a facilitar el aprendizaje a distancia, tanto para instituciones educativas como entornos empresariales.

El ámbito educativo es quizás el caso de uso por el que más se las conoce debido al uso de esta tecnología, cada vez mayor, tanto en educación obligatoria como en estudios posteriores. Aun así, tampoco hay que olvidar el entorno empresarial, donde muchas empresas lo utilizan para ofrecer formaciones a sus empleados mejorando así su capacitación.

Este proyecto se centrará en el campo educativo. El objetivo consiste en aumentar el soporte informativo ofrecido a los estudiantes que utilizan dichas plataformas. Para hacer esto posible, vamos a utilizar diferentes mecanismos basados en inteligencia artificial. Machine Learning es un subcampo de las ciencias de la computación que estudia diferentes técnicas con las cuales un ordenador pueda aprender a partir del procesado de información y realizar predicciones en consecuencia. Una definición menos técnica, utilizando las palabras de Arthur Samuel (uno de los pioneros en este campo), dice que consiste en “dotar a los ordenadores con la habilidad de aprender sin haber sido programados explícitamente para ello”.

A través de la recolección de la información generada por los LMS y su posterior procesado, utilizando algoritmos de aprendizaje, se pueden extraer diferentes parámetros de rendimiento del usuario con el fin de mantenerle informado en la medida de lo posible de su rendimiento actual y del posible resultado final si este se mantuviera, siendo así capaz el estudiante de modificar sus hábitos de estudio con antelación.

A partir de las ideas mencionadas anteriormente, los objetivos del proyecto son:

- Analizar los factores relacionados con el rendimiento académico y elaborar un sistema predictivo.
- Ampliar la funcionalidad de los sistemas de gestión de aprendizaje (LMS) ofreciendo diversos datos estadísticos que puedan ser utilizados por los estudiantes.
- Crear una plataforma sencilla donde poder mostrar los resultados.
- Privacidad. Estos datos pretenden ser de utilidad para el alumno, en ningún caso han de poder ser consultados por terceras personas, incluidos los docentes.
- Actualizado. La información mostrada ha de actualizarse diariamente.
- Simplicidad tanto en el diseño de la plataforma donde se mostrarán los datos como el formato de estos. Evitar un exceso de información difícil de interpretar por los usuarios, los datos serán concisos y se mostrarán ordenadamente.
- Analizar el resultado final y valorar la utilidad de la información ofrecida.

2. Contexto

La idea de este proyecto surgió a raíz del análisis de la situación académica de muchos alumnos en su primer año universitario, donde se produce un gran cambio respecto los estudios anteriormente realizados, que en muchos casos es difícil de gestionar. Se observó, como se ha mencionado anteriormente, que muchos de los problemas derivaban de malos hábitos de trabajo, por ello se consideró la utilidad de una herramienta que pueda advertir a los estudiantes de su rendimiento actual (entre otra información) y que estos dispongan de margen para corregirlo.

2.1 Actores implicados

Como se puede deducir de la explicación anterior, el producto va dirigido para los **estudiantes**, focalizado sobre todo en grados universitarios, donde es más difícil llevar a cabo un seguimiento por parte del tutor. El objetivo principal de los estudiantes consiste en desarrollar nuevos conocimientos (y en segunda instancia, aprobar las diversas materias), esta herramienta les dará soporte a lo largo de este proceso.

En cuanto a quien usará esta solución, encontramos dos agentes principales, los **estudiantes** (por razones obvias) y las **universidades**, ya que serán estas últimas las que implantarán la herramienta en sus servidores y ofrecerán sus servicios a los estudiantes.

Por último, entre los beneficiarios del uso de esta herramienta, encontramos a la **universidad,** ya que un incremento en el nivel de aprendizaje (y aprobado) de los alumnos tiene también repercusiones positivas para la universidad, mejorando su reputación e incrementando potencialmente el número de solicitantes anuales. Los **estudiantes** también se beneficiarán de ella, les ayudará a organizarse mejor en su día a día y a tener una mayor visibilidad de su estado actual. En el grupo de entidades que pueden beneficiarse del resultado del proyecto, encontramos también al **Estado** como un agente indirecto**.** En el caso de las universidades públicas, parte de los estudios están financiados por el estado, cuando un estudiante abandona la carrera, en muchos casos, esta inversión se considera perdida. No estamos hablando de ahorro, pero si de un incremento de la utilidad de la inversión.

3. Estado del arte

Este proyecto abarca el sector de los sistemas de gestión del aprendizaje y machine learning, en ambos sectores encontramos diferentes estudios y proyectos interesantes. Actualmente la investigación en el campo de la inteligencia artificial está en auge, debido a que la tecnología disponible ya es suficiente para ello y el sin fin de posibles aplicaciones que tiene este campo.

El sector educativo es uno de los muchos sectores donde poder aplicar esta tecnología. Debido a la poca explotación de la inteligencia artificial hasta la fecha, encontramos pocos estudios con un objetivo parecido y en todos los casos son muy recientes (no anteriores al 2015). Aun así, esto no implica que no se disponga de referencias suficiente.
En la mayoría de estudios consultados, el proyecto está enfocado en beneficiar al profesor o institución educativa, mejorando las capacidades analíticas de sus plataformas (o a través de la creación de nuevas herramientas) para poder detectar el mal rendimiento de los alumnos y ayudarles con antelación. Este es el caso de *Demonstration of a Moodle student monitoring web application [2],* un paper realizado en colaboración por el Instituto de Investigación en Informática de Toulouse(IRIT) y por Andil, empresa dedicada al sector de las tecnologías de la información y de la formación, llevando a cabo trabajos estrechamente relacionados con el tema que estamos tratando, como son el e-learning y los LMS.

La idea es muy similar a la planteada en este proyecto, se trata de crear una aplicación web que utiliza diferentes técnicas de machine learning y minería de datos para monitorizar el progreso de los estudiantes en los cursos realizados en las plataformas online y poder así avisarles antes de que fallen. El objetivo de este paper se basa en aumentar el rango de herramientas disponibles para los docentes y que estos sean los que ayuden a los alumnos. Acorde con el paper, el resultado final ha sido muy satisfactorio entre los usuarios, y en sus siguientes iteraciones se plantea la mejora de los mecanismos predictivos, comparando el resultado de diferentes algoritmos.

Otro estudio al que me gustaría hace referencia es *Indicators of Good Student Performance in Moodle Activity Data [3],* realizado por miembros de Insight Centre, University College Dublin, Ireland. Para que la plataforma que se desea elaborar sea de verdadera utilidad es muy importante la fiabilidad del algoritmo predictivo, para ello, se requiere que los datos de entrada recibidos por el algoritmo sean lo más precisos posibles. Este paper realiza un análisis de la actividad de los estudiantes en Moodle y pretende comprobar tres hipótesis iniciales: Las entregas tempranas son una buena señal, un alto nivel de actividad en la plataforma es indicador de buenos resultados, y, por último, que la actividad por las tardes es mejor que la diurna. El resultado final confirma la validez de las tres hipótesis.

Para acabar, hacer mención (y agradecer la amabilidad recibida por parte de sus participantes) al estudio *OU Analyse: Analysing At-Risk Students at The Open University [4]*. Es un paper, que al igual que ejemplos anteriores, pretende extraer patrones, elaborar y comprobar la validez de diferentes modelos predictivos para detectar con antelación el fracaso de los alumnos y poder ayudarles. Adicionalmente, al igual que la idea propuesta en este proyecto, han elaborado un dashboard bastante completo, pero enfocado de cara al profesorado, donde poder mostrar los resultados de las predicciones, filtrar los resultados según algún criterio (situación demográfica, por ejemplo), poder seguir el rendimiento individual de un alumno, etc.

A diferencia de la mayoría de estudios consultados previamente, han publicado los datos de Moodle que utilizaron en sus pruebas, un dataset muy completo tanto por el número de muestras como por la cantidad de variables que disponen. Del mismo modo que los dos proyectos anteriores, los resultados finales resultaron muy satisfactorios, actualmente se está trabajando en un paper final más adecuado. El factor distintivo de este paper es que fue utilizado en cursos reales, donde partieron con una precisión del 50%, enviando informes semanales, y finalmente obtuvieron una fiabilidad cercana al 90%.

En todos los proyectos referenciados, se consigue con éxito elaborar modelos predictivos sobre el rendimiento académico, por ello voy a utilizar algunas de las variables e hipótesis planteadas hasta el momento. A diferencia de la plataforma de Andil o la de The Open Univeristy , yo no pretendo que el profesorado disponga de acceso a la información, sino dejar que sea el alumno el que tomé las decisiones pertinentes a partir de la información facilitada, evitando así también que el profesor pueda ver su objetividad afectada a raíz de los datos recolectados.

4. Alcance

El objetivo principal del proyecto es elaborar un sistema predictivo que sea útil para el alumno, por ello, la plataforma que se va a desarrollar será sencilla y tan solo pretende dar un ejemplo de la posible información susceptible de ser mostrada. En un futuro, cada institución adaptará la solución a sus portales. Las dos ramas principales del proyecto serán:

- Un sistema predictivo que dados unos parámetros de entrada calcule el resultado final del curso (aprobado o suspendido), estos datos de entrada serán extraídos de la información recopilada por Moodle. A su vez, podemos dividir la rama en los modulos siguientes:
  - Modulo que extraiga la información de Moodle y la adapte a un formato entendible para posteriormente almacenarlos en una base de datos.
  - Modulo que se encarga de aplicar los diferentes algoritmos predictivos a partir de la información almacenada en la base de datos. El resultado final se guardará en otra tabla utilizada por el portal web.
- Un portal web, con un formato de tablero (comúnmente conocidos por su denominación en inglés, dashboards), donde se muestra el rendimiento del alumno para cada una de las asignaturas que está realizando.

4.1 Obstáculos

En cuanto a las posibles dificultades (potencialmente bloqueantes) que plantea este proyecto, encontramos:

- **Obtención de datos:** Dada la naturaleza de la información que se pretende utilizar, uno de los principales obstáculos a la hora de realizar el proyecto va a ser la obtención de datos reales y en cantidad suficiente para poder entrenar el modelo. Para entrenar un modelo predictivo (que sea capaz de predecir) se le han de introducir datos reales de un curso ya finalizado, sabiendo el resultado final, para que este puede extraer patrones. Por ley, las instituciones docentes no pueden hacer público los datos sobre rendimiento ni actividad de los participantes, por ello, un punto crítico va ser la obtención de este tipo de información. Al contar con tiempo limitado para realizar el estudio, no se contempla la creación de un curso donde los alumnos acepten que sus datos sean utilizados anónimamente, se requeriría de un cuatrimestre entero antes de poder empezar a trabajar.

- **Tecnología muy reciente:** Como se ha mencionado en apartados anteriores, el machine learning es un campo todavía muy reciente, donde las herramientas disponibles han sido poco utilizadas hasta el momento y no hay un gran soporte comunitario. Este podría ser uno de los inconvenientes a lo largo de la fase de implementación.

5. Metodología

En muchas empresas se está apostando por aplicar metodologías de trabajo Agile (o SCRUM, que no deja de ser un caso específico de metodologías de trabajo agiles), los principios de las cuales dan para un proyecto entero. Una de las principales ventajas es que evita el problema de la metodología en cascada, donde para empezar una fase del proyecto se requiere que la anterior esté completada, imposibilitando un flujo de trabajo regular y paralelo entre diferentes equipos.

Para el desarrollo de este proyecto, al tratarse de un trabajo de final de grado realizado individualmente, se va a apostar por una metodología en cascada, ya que en ningún momento nadie se va a encontrar bloqueado porque no esté finalizada la tarea de otro compañero. Por lo tanto, se van a ir realizando las diferentes tareas de forma secuencial, las fases que vamos a seguir están simplificadas en tres pasos.

- **Análisis de requerimientos y diseño:** En esta fase, se analizará que se ha de llevar a cabo, y se diseñará el sistema

- **Implementación:** Una vez se ha realizado el diseño del sistema, se procederá a la implementación del mismo.

- **Verificación:** En esta última fase, se realizarán diferentes pruebas sobre el sistema elaborado y se verificará que cumple con lo inicialmente acordado.

Al no colaborar con ninguna empresa más allá de la Universidad Politécnica de Catalunya, a lo largo del proyecto tan solo se mantendrá comunicación con la directora del trabajo y el codirector mediante correo electrónico y reuniones en persona, cada dos semanas (sujetas a modificaciones según la disponibilidad de los implicados). Del mismo modo que las decisiones relevantes sobre el proyecto se consensuarán entre los tres.

Para facilitar la comunicación y la organización durante el proyecto, se van a utilizar dos herramientas principales. *Bitbucket* en cuanto al sistema de control de versiones de código. *Trello* para facilitar la gestión del trabajo, tanto como herramienta de organización personal como de soporte para otorgar visibilidad al director y codirector del estado actual del proyecto, que tareas se están llevando a cabo, cuando se tiene previsto finalizarlas, etc.

Adicionalmente, dentro de cada una de las fases de desarrollo que se han mencionado anteriormente, se utilizará una metodología de trabajo basada en agile llamada Kanban. Kanban consiste en dividir el trabajo en tareas, lo más simples posibles, y representarlas en una tabla (se suele recomendar que sea en formato físico) organizada en tres columnas, “To Do, In Progress, Done”, pueden existir variaciones según las necesidades del proyecto. La utilización de Kanban sirve para aumentar la trazabilidad del proyecto, los siguientes pasos a realizar, etc. En el caso particular de este proyecto, se utilizará tanto una tabla física (para uso del desarrollador) como formato digital a través de *Trello*, para facilitar la trazabilidad de los profesores.

6. Descripción de las tareas

6.1 Tareas

El marco temporal en el que se va a desarrollar el proyecto está comprendido entre febrero de 2016 y junio del mismo año. La idea del proyecto se planteó los meses previos a su matriculación conjuntamente con el codirector y directora. Tiene prevista su finalización la última semana de junio, con la presentación y defensa del mismo.

A continuación, se detallan las diferentes tareas del proyecto junto a las subtareas que forman parte de cada una de ellas, también se indicará la estimación temporal.

Tarea	**Horas**
Inicio del proyecto	**48**
1.Tareas administrativas y puesta en marcha	8
2. Aprendizaje y familiarización	40
Gestión del proyecto	**75**

1. Definición del alcance	24,5
2. Planificación temporal	8,25
3. Gestión económica y sostenible	9,25
4. Presentación preliminar	6,25
5. Pliego de condiciones	8,5
6. Presentación oral y documento final	18,25
Implementación sistema predictivo	**140**
1. Análisis de requisitos, búsqueda y pretratado de datos e instalación de software necesario	40
2. Implementación del sistema predictivo	80
3. Realización de pruebas, comparación entre algoritmos y mejoras	20
Implementación de la plataforma web	**125**
1.Análisis de requisitos	5
2.Diseño	10
3. Aprendizaje de tecnologías y lenguajes de programación necesarios	30
4. Implementación de la plataforma	60
5. Realización de tests y mejoras	20
Documentación	**90**
1. Escribir documentación	60
2. Preparación de la presentación	30
TOTAL	**478**

Tabla 1: Definición y estimación de las tareas del proyecto

6.2 Planificación y Dependencias

El proyecto lo realizará una única persona que irá llevando a cabo las tareas de forma secuencial, en ese sentido este proyecto no tiene grandes problemas de paralelismo, ya que en ningún momento una persona se verá bloqueada porque un apartado no esté listo.

Dicho esto, a raíz de la metodología de trabajo que se va a llevar a cabo (cascada), van a existir dependencias obligadas que están determinadas por el orden de realización escogido. Dentro de cada tarea, encontramos las dependencias que se pueden deducir de cualquier trabajo, antes de la implementación hay que investigar y documentarse sobre la materia. Ejemplos de este tipo son el aprendizaje y familiarización previo al proyecto o el análisis de requisitos presente tanto en el sistema predictivo como en la plataforma web.

La parte de gestión del proyecto (segunda tarea), no tiene dependencias más allá de las tareas administrativas iniciales y la familiarización con el tema del trabajo. A lo largo de este proceso (mediados de febrero, finales de marzo) se pretende también combinarlo con tareas del sistema predictivo, que tendrá una duración de dos meses.

En lo referente a los dos bloques relevantes de cara a la funcionalidad (sistema predictivo y plataforma web) no existe una dependencia real entre ellos, ya que se puede desarrollar la plataforma web sin necesidad de que funcione correctamente (o incluso que no esté implementado) el sistema predictivo. Aun así, como se ha comentado en más de una ocasión en apartados anteriores, el principal objetivo del proyecto es elaborar un sistema predictivo, por lo tanto, sin que funcione el primer componente no tiene sentido realizar el segundo. Así pues, todo y no existir una dependencia real, se ha decidido que el sistema predictivo sea un requerimiento de cara a iniciar la implementación de la plataforma web.

La documentación del proyecto se va a ir realizando a lo largo de todo el proceso, paralelamente a las tareas de implementación. Por ello, esta tarea corresponde prácticamente con la fecha inicial y final del proyecto.

6.3 Diagrama de Gantt

A continuación, se presenta el diagrama de Gantt, una forma rápida y visual de entender la planificación del proyecto. Se han utilizado cinco colores, uno por cada tarea de la Tabla 1. El color azul corresponde al inicio del proyecto, el naranja a la gestión del proyecto, los colores verde y negro al sistema predictivo y plataforma web respectivamente, y, por último, el rojo para la documentación.

Tal y como se ha explicado en el apartado anterior, en la primera fase del proyecto, se realizará paralelamente la gestión del proyecto y el sistema predictivo. Una vez finalizados ambos bloques, se procederá a la implementación de la plataforma web. Dentro de este bloque se pueden observar dos tareas en paralelo, corresponden al *Aprendizaje de tecnologías y lenguajes de programación necesarios* e *Implementación de la plataforma*, se ha considerado que el proceso de aprendizaje tendrá lugar a la par que el proceso de implementación. Por último, la documentación final se llevará a cabo a lo largo de todo el proceso, acabando con la presentación y defensa del proyecto.

Tarea	Fecha inicio	Fecha fin
Inicio del proyecto	1 de febrero 2017	10 de febrero 2017
Gestión del proyecto	21 de febrero 2017	27 de marzo 2017
Implementación sistema predictivo	10 de febrero 2017	13 de abril 2017
Implementación de la plataforma web	14 de abril 2017	1 de junio 2017
Documentación	21 de febrero 2017	26 de junio 2017

Tabla 2: Fecha de realización de tareas

Nombre	Duración	Inicio	Fin	Predecesoras	Recursos
Tareas administrativas y puesta en marcha	8d	01/02/2017	06/02/2017		
Aprendizaje y familiarización	15d	01/02/2017	10/02/2017		
Definición del alcance	12d	21/02/2017	28/02/2017	1	
Planificación temporal	8d	01/03/2017	06/03/2017	3	
Gestión económica y sostenible	10d	07/03/2017	13/03/2017	4	
Presentación preliminar	10d	14/03/2017	20/03/2017	5	
Pliego de condiciones	10d	21/03/2017	27/03/2017	6	
Presentación oral y documento final	10d	21/03/2017	27/03/2017	6	
Análisis de requisitos, búsqueda y pretratado de datos e instalaci	15d	10/02/2017	21/02/2017	1	
Implementación del sistema predictivo	60d	21/02/2017	04/04/2017	9	
Realización de pruebas, comparación entre algoritmos y mejoras	15d	04/04/2017	13/04/2017	10	
Análisis de requisitos	4d	14/04/2017	17/04/2017	11	
Diseño	5d	18/04/2017	20/04/2017	12	
Aprendizaje de tecnologías y lenguajes de programación necesar	50d	21/04/2017	25/05/2017	13	
Implementación de la plataforma	50d	21/04/2017	25/05/2017	13	
Realización de tests y mejoras	10d	26/05/2017	01/06/2017	15	
Escribir documentación	158d	21/02/2017	09/06/2017		
Preparación de la presentación	22d	12/06/2017	26/06/2017	8,16,17	

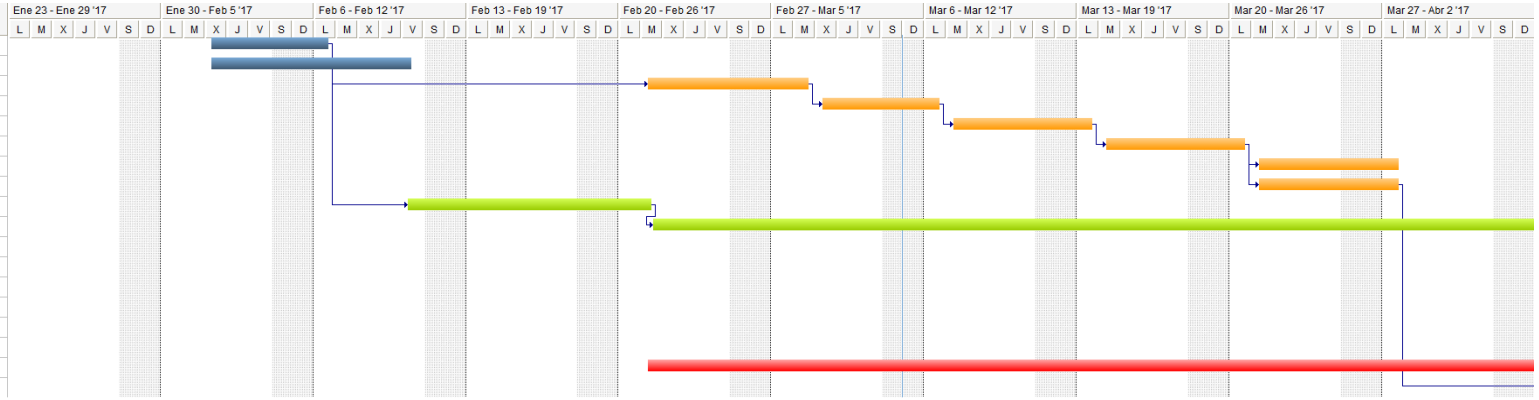

Figura 1: Diagrama de Gantt Enero-Marzo

Nombre	Duración	Inicio	Fin	Predecesoras	Recursos
Tareas administrativas y puesta en marcha	8d	01/02/2017	06/02/2017		
Aprendizaje y familiarización	15d	01/02/2017	10/02/2017		
Definición del alcance	12d	21/02/2017	28/02/2017	1	
Planificación temporal	8d	01/03/2017	06/03/2017	3	
Gestión económica y sostenible	10d	07/03/2017	13/03/2017	4	
Presentación preliminar	10d	14/03/2017	20/03/2017	5	
Pliego de condiciones	10d	21/03/2017	27/03/2017	6	
Presentación oral y documento final	10d	21/03/2017	27/03/2017	6	
Análisis de requisitos, búsqueda y pretratado de datos e instalaci	15d	10/02/2017	21/02/2017	1	
Implementación del sistema predictivo	60d	21/02/2017	04/04/2017	9	
Realización de pruebas, comparación entre algoritmos y mejoras	15d	04/04/2017	13/04/2017	10	
Análisis de requisitos	4d	14/04/2017	17/04/2017	11	
Diseño	5d	18/04/2017	20/04/2017	12	
Aprendizaje de tecnologías y lenguajes de programación necesar	50d	21/04/2017	25/05/2017	13	
Implementación de la plataforma	50d	21/04/2017	25/05/2017	13	
Realización de tests y mejoras	10d	26/05/2017	01/06/2017	15	
Escribir documentación	158d	21/02/2017	09/06/2017		
Preparación de la presentación	22d	12/06/2017	26/06/2017	8,16,17	

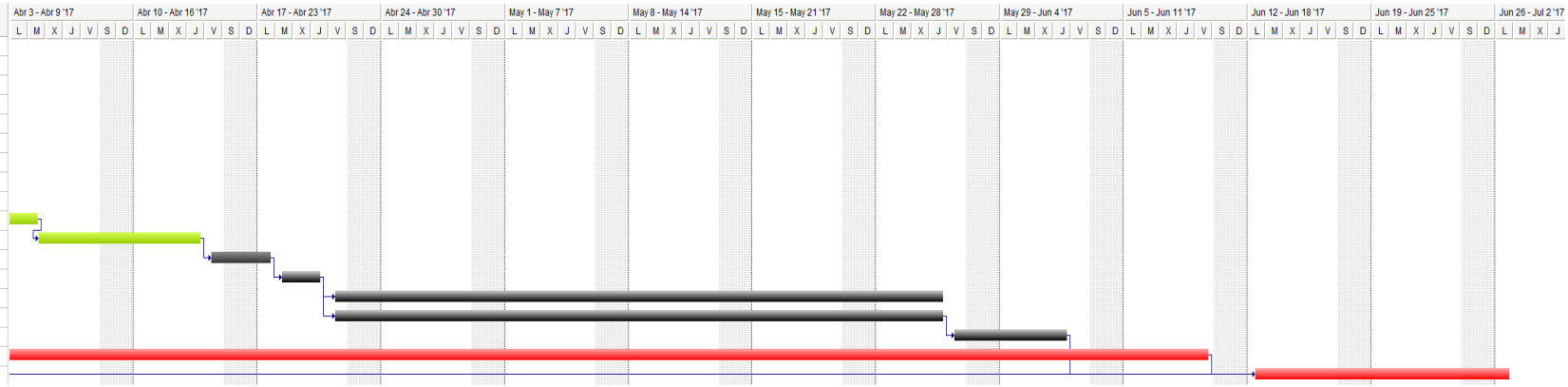

Figura 2: Diagrama de Gantt Abril-Junio

6.4 Recursos

En este apartado se explican los recursos utilizados en la realización del proyecto. Se han divido entre recursos físicos (o hardware) y software. En este apartado no se va a profundizar en la definición de los recursos humanos, ya que, al tratarse de un proyecto de final de grado sin ninguna colaboración con empresas externas, tan solo lo va a llevar a cabo una persona junto al soporte de la directora y el codirector. De todos modos, en el apartado 8.1 se especifican los diferentes roles y los costes asociados. Vamos a empezar por los recursos físicos:

- Portátil Asus K55VM con Windows 7 Professional y Ubuntu 14.04 LTS. 8GB de memoria RAM y procesador Intel Core i7-3610QM

- Servidor dedicado, alquilado a la compañía francesa Kimsufi. El sistema operativo es Ubuntu Server 14.04 "Trusty Tahr" LTS (64bits), procesador Intel 4G Atom N2800 y un disco SSD de 40 GB.

- Conexión a Internet. no hay que olvidar este elemento, esencial para todas las tareas realizadas y a través de la cual se accede a muchos de las herramientas de software que se describirán a continuación.

En cuanto a los elementos software utilizados encontramos tanto administrativos como de soporte a la programación, algunos de ellos se han mencionado en apartados anteriores.

- **Trello**. Herramienta orientada a la administración de tareas. Con un formato muy simple que facilita la organización y aumenta la visibilidad de todos los miembros del equipo. Se utilizará principalmente para la coordinación entre el alumno y director.

- **Bitbucket**. Software de control de versiones donde se almacenará el código del sistema predictivo y de la plataforma web, en repositorios distintos.

- **Microsoft Word**. Software de escritura con el que se redactará la documentación del proyecto.

- **Sublime Text 2**. Editor de texto utilizado para la programación de código.

- **Learning Locker**. Base de datos (o repositorio) que sirve como herramienta de almacenamiento y rastreo de información para los sistemas de e-learning. Se utilizará para almacenar la información sobre la actividad de los alumnos.

- **Moodle**. Plataforma de creación de cursos virtuales. Es con diferencia la más utilizada, en este proyecto se utilizará durante la fase de testeo para comprobar el correcto funcionamiento del algoritmo predictivo y la plataforma web.

- **MySQL**. Base de datos, se utilizará para almacenar toda la información utilizada por la plataforma web y el sistema predictivo, como nombres de usuarios y contraseñas.

- **Gantter**. Herramienta software online para el diseño de diagramas de Gantt. Se utilizará para la creación del esquema de Gantt del proyecto.

7. Alternativas y plan de acción

Dadas las características de este proyecto, tiempo limitado (5 meses) y realización individual, es oportuno considerar los puntos críticos del plan de trabajo establecido y elaborar alternativas en caso de no poder cumplir con lo estimado, garantizando una entrega de valor en junio.

La fase inicial y gestión del proyecto no presentan grandes dificultades, en todo este proceso además se contará con el soporte del profesorado de Gestión de Proyectos (GEP). Una vez más, el foco potencial de problemas reside en el sistema predictivo, a continuación, se analizan los motivos.

Un elemento crítico (aunque actualmente solucionado) son la obtención de datos válidos para el estudio. Tal y como se ha explicado brevemente en apartados anteriores, para que un sistema predictivo sea capaz de elaborar predicciones coherentes se ha de haber entrenado previamente. Este proceso de entrenamiento consiste en introducirle al sistema los datos de un curso ya finalizado, de modo que pueda sacar patrones de conducta entre los alumnos que aprueban y los que suspenden. Actualmente después de un proceso de búsqueda intensivo, se consiguió un dataset válido que utilizar, en caso que la información de este no sea válida para el propósito del proyecto va a resultar muy difícil conseguir otro, debido a las estrictas políticas de privacidad a la que está sometida este tipo de información. Aun así, las pruebas realizadas sobre el dataset no muestran ningún problema.

Otro factor importante es el relativo poco soporte y soluciones tecnológicas (api's) que existen en el campo de los algoritmos predictivos. Por lo tanto, el proceso de elaboración del sistema predictivo podría alargarse más de lo estimado. Si se produce esta situación, se ha tomado la decisión de simplificar el diseño y la complejidad de la plataforma web. En el peor de los casos, se puede elaborar una simple página web donde aparezcan las asignaturas del estudiante y su rendimiento actual, eliminando toda funcionalidad adicional.

Por último, en cuanto a la documentación no se prevé ningún tipo de improviso, el proceso de redacción se irá realizando a la par que la implementación, sumado al echo que a través del curso de GEP se documenta una parte importante del proyecto.

8. Identificación y estimación de los costes

En los apartados previos se ha explicado las tareas que formarán parte del proyecto, las horas de dedicación, recursos utilizados, etc. Todo esto tienes unos costes asociados, que vamos a dividir en recursos humanos, hardware y software. Las estimaciones siguientes se han calculado a partir de las tareas definidas en el Gantt.

8.1 Recursos humanos

Los roles que se van a considerar para el proyecto son jefe de proyecto, analista, diseñador, programador y tester. La distinción entre programador y tester viene dada porque en múltiples empresas, el responsable de comprobar el correcto funcionamiento del producto no puede ser el mismo que lo ha elaborado.

Como se ha explicado anteriormente, el estudio lo va a llevar a cabo una única persona, por lo tanto, esta asumirá todos los roles. Para hacer la estimación lo más ajustada a la realidad posible, se van a considerar diferentes sueldos según el rol, a pesar de que el salario estipulado para estudiantes es de 8€/h.

Rol	Salario (€/hora)
Jefe de proyecto	20
Analista	15
Diseñador	15
Programador	12
Tester	12

Tabla 3: Estimación salarial según el rol

El salario ha sido extraído a partir de las experiencias laborales del autor. En la siguiente tabla se puede ver desglosado por tareas el coste total del proyecto asociado a recursos humanos teniendo en cuenta los salarios de la tabla 1.

Tarea	Responsable	Horas	Coste estimado
Inicio del proyecto	**-**	**48**	**640**
1.Tareas administrativas y puesta en marcha	Jefe de proyecto	8	160
2. Aprendizaje y familiarización	Programador	40	480
Gestión del proyecto	**-**	**75**	**1500**
1. Definición del alcance	Jefe de proyecto	24,5	490
2. Planificación temporal	Jefe de proyecto	8,25	165
3. Gestión económica y sostenible	Jefe de proyecto	9,25	185

4. Presentación preliminar	Jefe de proyecto	6,25	125
5. Pliego de condiciones	Jefe de proyecto	8,5	170
6. Presentación oral y documento final	Jefe de proyecto	18,25	365
Implementación sistema predictivo	**-**	**140**	**1800**
1. Análisis de requisitos, búsqueda y pretratado de datos e instalación de software necesario	Analista y Programador	40	600
2. Implementación del sistema predictivo	Programador	80	960
3. Realización de pruebas, comparación entre algoritmos y mejoras	Tester	20	240
Implementación de la plataforma web	**-**	**125**	**1545**
1.Análisis de requisitos	Analista	5	75
2.Diseño	Diseñador	10	150
3. Aprendizaje de tecnologías y lenguajes de programación necesarios	Programador	30	360
4. Implementación de la plataforma	Programador	60	720
5. Realización de tests y mejoras	Tester	20	240
Documentación		**90**	**1800**
1. Escribir documentación	Jefe de proyecto	60	1200
2. Preparación de la presentación	Jefe de proyecto	30	600
TOTAL	**-**	**478**	**7285**

Tabla 4: Estimación de costes total en recursos humanos

Las tareas de la tabla anterior han sido extraídas del apartado *Tareas,* que posteriormente se utilizaron en la definición del diagrama de Gantt. Es una forma rápida de ver el coste total del proyecto asociado a los recursos humanos, así como de poder observar que partes son las más costosas económicamente.

8.2 Hardware

Dentro de los recursos materiales que se van a utilizar en el proyecto, se va a realizar una distinción entre hardware y software. Este apartado profundiza en el primero de ellos.

Producto	Precio €	Unidades	Vida útil	Amortización por hora (€/h)	Horas estimadas	Amortización total (€)
Asus K55VM	700	1	4 años	0,0851	880	74,90
Servidor Kimsufi alquilado	49,95	1	5 meses	0,0567	880	49,95
TOTAL						124,85

Tabla 5: Costes totales de hardware

Los resultados de la tabla 3 se han obtenido a partir de las siguientes suposiciones. El año 2017 dispone de 257 días laborables, para el caso del portátil Asus K55VM se ha supuesto que los próximos 3 años también dispondrán de dicha cantidad. La jornada laboral es de 8h, por lo tanto, para calcular la amortización por hora se han dividido las horas totales de uso (257 días/año * 8h al día) al precio total.

El servidor alquilado a la empresa Kimsufi es un caso particular, ya que el precio real del servidor es mucho mayor al indicado en la tabla. El valor que aparece son los cinco meses de alquiler (9,99€/mes). Este servidor se ha alquilado para la realización del proyecto, por lo tanto, el coste de amortización total ha de ser equivalente al precio invertido en él.

Las horas estimadas para ambos recursos (880h) se han calculado contando un uso continuo de 8h por día laborable a lo largo de los cinco meses de duración del proyecto.

8.3 Software

El software utilizado durante la realización del trabajo, especificado en el apartado *Recursos*, dispone de un período de prueba suficiente para realizar las tareas sin necesidad de pagar por él o es gratuito. El editor de texto Microsoft Word no es gratuito, pero su coste viene incluido en el precio de compra del Asus K55VM (adquirido con una distribución oficial de Microsoft Windows). Por lo tanto, en este apartado no se considera ningún coste económico asociado al proyecto.

8.4 Costes generales indirectos

A parte de los costes derivados de los recursos humanos, hardware y software, encontramos también algunos gastos indirectos que no suelen participar activamente en el desarrollo del proyecto. Dentro de este grupo, podemos encontrar gastos en transporte, electricidad, agua, internet, etc. En el caso particular de este proyecto no existen costes debido a alquiler. En la siguiente tabla se pueden ver todos estos costes en detalle (suponiendo un período de duración de 5 meses para el proyecto).

Producto	Precio (€/mes)	Coste estimado (€)
Transporte	10	50
Electricidad	40	90
Agua	15	75
Gas	15	75
Internet	50	250
TOTAL		540

Tabla 6: Costes indirectos totales

8.5 Contingencia

En todo proyecto surgen variaciones durante el proceso de desarrollo, algunas de estas situaciones pueden repercutir en un incremento económico respecto al valor inicial estimado en la fase de planificación. Es aconsejable tener en cuenta este tipo de situaciones y calcular un margen de error asumible, este margen de error reservado recibe el nombre de contingencia. La contingencia que vamos a reservar para el proyecto es un 15% de los costes directos (recursos humanos y materiales) e indirectos.

Producto	Margen de error (%)	Precio inicial (€)	Coste de contingencia (€)
Recursos humanos	15	7285	1092,75
Recursos materiales	15	124,85	18,73
Recursos indirectos	15	540	81
TOTAL			1192,48

Tabla 7: Costes de contingencia

8.6 Imprevistos

En esta sección se van a analizar los posibles imprevistos que puedan surgir, teniendo en cuenta cuan probable es su aparición y el impacto económico que estos supondrían.

Dadas las particularidades del proyecto, un imprevisto a tener en cuenta es el fallo de alguno de los ordenadores utilizados, tanto el portátil como el servidor alquilado. Al trabajar con el código y la documentación en línea, no supondría ninguna pérdida de trabajo. En caso de que falle el portátil, el tiempo de reparación podría ser asumible dependiendo de la gravedad, en cuyo caso seguimos disponiendo del servidor para realizar las tareas. Por el contrario, si no nos podemos permitir tanto tiempo en repararlo, se consideraría la posibilidad de comprar uno nuevo. Estimamos la posibilidad de que falle el portátil en un 5%.

Si fallará el servidor alquilado, el servicio técnico resolvería la incidencia en un tiempo asumible (por contrato) o en el peor de los casos se podría alquilar uno adicional durante un mes. De todos modos, tanto la conectividad a internet como el subministro eléctrico es redundante, y garantizan un SLA del 99,9%. Estimamos la posibilidad de que falle el servidor en un 1%.

En proyectos de esta índole es muy común que el tiempo de desarrollo sea superior al estimado, por ello también hay que tener en cuenta el sueldo de los trabajadores en ese tiempo extra de implementación. Vamos a suponer un incremento en el tiempo de proyecto de dos semanas (10 días), con una probabilidad del 10%

Producto	Probabilidad	Precio (€)	Coste (€)
Asus K55VM	5%	700	35
Servidor alquilado Kimsufi	1%	9,99	0,01
Retraso en la implementación (10 días)	10%	960	96
TOTAL			131,01

Tabla 8: Costes de imprevistos

8.7 Presupuesto final

Para acabar con el presupuesto, a modo de resumen de todos los apartados anteriores, en la siguiente tabla se puede observar el presupuesto total destinado al proyecto, teniendo en cuenta la estimación inicial, la contingencia y los imprevistos.

Concepto	Coste (€)
Recursos humanos	7285
Recursos materiales	124,85
Costes indirectos	540
Contingencia	1192,48
Imprevistos	131,01
TOTAL	9273.34

Tabla 9: Presupuesto final

8.8 Control de gestión

Este apartado pretende establecer las medidas para controlar que el presupuesto no difiera en exceso respecto lo estimado, y en caso de que esto ocurra, ser capaces de detectarlo rápidamente. Gastos constantes y derivados de la realización del proyecto tan solo encontramos los recursos humanos y el alquiler del servidor Kimsufi, los demás gastos continuarían estando presentes aun si no se llevara a cabo el proyecto (todos los gastos indirectos antes mencionados, por ejemplo). El portátil ya ha sido pagado en su totalidad y no estamos utilizando ningún software de pago temporal, así pues, vamos a centrarnos en los dos recursos susceptibles de aumentar nuestro gasto económico.

En lo referente al servidor alquilado, en la estimación de costes se ha contado el alquiler completo desde el comienzo del proyecto hasta el último mes del mismo. Por lo tanto, no se prevé la necesidad de ampliar la suscripción.

Los recursos humanos sí que tienen una influencia directa respecto el estado actual del proyecto, una desviación temporal supondrá un incremento en el gasto de los trabajadores, por eso se requiere un proceso de control efectivo para detectar estas situaciones lo antes posible.

La estrategia que se va a llevar a cabo consiste en mantener dos reuniones mensuales con la directora del proyecto (sin una duración predefinida), con un doble propósito, mantenerla informada del estado actual del proyecto y poder comparar el estado actual respecto el estimado. En el peor de los casos, se produciría una desviación de dos semanas respecto al plan establecido sin que los responsables lo detectarán. En cuyo caso, se utilizaría la contingencia mencionada en apartados anteriores (15%), pensada para cubrir dos semanas adicionales de suelo de los trabajadores.

9. Learning management systems

9.1 Definición y objetivo

Los Learning Management Systems(LMS), son un software instalado normalmente en servidores web que integran un conjunto de herramientas destinadas para la enseñanza o aprendizaje[1]. Se emplea para crear, administrar, distribuir y gestionar actividades formativas virtuales. Los principales usuarios de este tipo de software se dividen en dos grupos, por un lado, encontramos a los organizadores o responsables del curso, como podría ser un profesor. Y por el otro lado, los alumnos de los diferentes cursos.

Como se puede deducir, el objetivo principal de un LMS consiste en crear un entorno virtual dedicado al aprendizaje, pero esta definición no consigue plasmar todas las funcionalidades que esto implica. Desde el punto de vista del docente, ofrece funcionalidades que favorecen la organización y seguimiento del curso, por ejemplo, permite un mayor rango de monitorización de los estudiantes, siendo el administrador del curso capaz de detectar partes del temario donde el rendimiento ha sido menor y poder actuar en consecuencia. En lo referente al usuario, encontramos una gran facilidad para la difusión de materiales didácticos y herramientas de comunicación y colaboración, ya sea entre alumno y profesor o entre alumnos.

Agrupando todas estas funcionalidades, se puede concluir que entre las principales funciones que debe cumplir un LMS, se encuentra tanto la **administración** de los usuarios como del propio espacio virtual (creación y organización de recursos, contenidos y eventos); **la evaluación** y seguimiento del proceso de aprendizaje, con todo lo que ello puede implicar, como la generación de los informes de avances; y, por último, un aspecto muy importante como es la **comunicación entre los participante**, a través de foros de discusión y videoconferencias, entre otros.

9.2 Orígenes y evolución

Como muchas de las tecnologías actualmente utilizadas, la difusión y usó de los learning management systems aumentó y evolucionó (se podría decir que paralelamente) a raíz de la invención de Internet. Hasta entonces, la manera de intercambiar conocimiento e interaccionar entre personas a través de la Web se basaba en la creación de páginas HTML, correo electrónico y foros de discusión.

La rápida expansión de internet, que afectó a diversos sectores de la sociedad, también influenció al ámbito educativo, ya que la explotación didáctica de la web permitió ampliar la oferta educativa y el acceso a la educación. Otro punto que también considero importante en esta influencia de internet en la educación es la reducción de costes, tanto temporales como económicos. Una vez el material didáctico es elaborado y publicado, este puede ser utilizado repetidas veces por una gran variedad de personas.

[1] Clarenc, C. A.; S. M. Castro, C. López de Lenz, M. E. Moreno y N. B. Tosco (Diciembre, 2013). Analizamos 19 plataformas de e-Learning: Investigación colaborativa sobre LMS. Página 29

A medida que su uso fue creciendo, empezó a surgir un problema muy común en el mundo de la tecnología, cada entidad desarrollaba su propia solución, tanto de forma propietaria como industrial (a través del uso de software de terceros). Esto dificultó, o en muchos casos imposibilitó, la interoperabilidad entre plataformas, por lo tanto, se decidió crear un estándar con el fin de poder controlar estos inconvenientes. Dos de los factores más influyentes que permitieron iniciar este proceso hacia la estandarización fueron la interoperabilidad y manejabilidad.

Otra vez más, los motivos de ahorro económico y temporal fueron los impulsores de la siguiente etapa. Abaratar los costes de producción de cursos, reducir el tiempo requerido para su desarrollo, facilitar su gestión y simplificar su actualización. Conceptos como **Interoperabilidad, reusabilidad o efectividad en los costos** tienen una gran importancia en el origen de gran parte del software existente. Todo esto se podría resumir en aprovechar el esfuerzo realizado por otra persona, aumentando la productividad propia

Otra característica importante es la **Manejabilidad**, no es tanto un problema, sino mas bien una funcionalidad nueva que permite aumentar la utilidad de este tipo de plataformas. Se trata de registrar la actividad entre el usuario y el contenido, "*saber cuántas veces o cuánto tiempo los usuarios visitaron el curso, o conocer el resultado de una evaluación. ¿Puede el contenido conocer el nombre de quien lo está leyendo, o saber si es la primera vez que lo recorre, o en qué página el usuario lo dejó en su última visita? Estas propiedades se consiguen definiendo una forma de comunicación estándar entre la plataforma y los contenidos*"[2].

9.3 Características de un Learning management system

A continuación, se listan algunas características que deberían cumplir los sistemas de gestión del aprendizaje.[3]

- Interactividad

Este aspecto tiene que ver con la comunicación asíncrona y bidireccional entre emisor y receptor, dotándola de cierto grado de libertad, siendo el receptor capaz de elegir que parte del mensaje le resulta de interés a la vez que el emisor establece el grado de interactividad del mensaje. Un LMS ha de ofrecer la suficiente interactividad para que el alumno sea el protagonista de su propio aprendizaje.

[2] Cita extraída de http://www.e-abclearning.com/queesscorm, empresa con más de 10 años de experiencia a nivel nacional e internacional en **soluciones de e-learning**.
[3] Parte de las características explicadas en este apartado son una combinación personal junto con las mencionadas en Clarenc, C. A.; S. M. Castro, C. López de Lenz, M. E. Moreno y N. B. Tosco (Diciembre, 2013). Analizamos 19 plataformas de e-Learning: Investigación colaborativa sobre LMS. Página 37

- Flexibilidad

La flexibilidad, aplicada al campo de las LMS, hace referencia a la independencia de la plataforma respecto a los planes de estudio, puede adaptarse tanto a la organización que va destinada como al objetivo pedagógico.

- Escalabilidad

Se refiere a la propiedad de aumentar la capacidad de trabajo sin que esto repercuta en el correcto funcionamiento del servicio. Aplicado a un LMS, consiste en la adaptabilidad de la plataforma respecto a un aumento de sus funcionalidades, del número de cursos publicados, de la cantidad de usuarios soportados, etc.

- Reusabilidad

Como se ha hecho mención anteriormente, es importante que este tipo de plataformas estén reguladas bajo un estándar. Beneficiando tanto al creador del curso como el receptor.

Que un LMS sea reusable significa que se puede aprovechar el trabajo realizado por terceras personas, aumentando el abanico de material didáctico para los usuarios, a la vez que reduciendo costes de creación para el autor.

- Accesibilidad

Es quizás la característica más importante de un LMS, se basa en eliminar las limitaciones para acceder a la plataforma, tener la información disponible a cualquier hora y en cualquier lugar, tan solo se requiere un ordenador y conexión a Internet.

Todas las características descritas anteriormente derivan en el beneficio final de una **reducción de costes**, ya que entre otras cosas permite una distribución de la enseñanza de forma económica, en cualquier lugar y momento, posibilita que los profesores y alumnos administren eficientemente su tiempo y horario, mejora el aprendizaje en los estudiantes, reduce los tiempos de enseñanza, así como los costes de producción de contenidos.

9.4 SCORM

Actualmente, el tipo de contenido susceptibles de ser presentado en un learning management system es muy extenso, pueden almacenar textos, gráficos, imágenes, animaciones, multimedia, enlaces a otros recursos o cualquier otro tipo de material que pueda visualizarse en un navegador web. Estos contenidos, a su vez, están compuestos por módulos, actividades, encuestas, evaluaciones, etc.

SCORM es considerado el estándar de facto que regula el software de creación de los LMS. Es decir, un conjunto de reglas de programación que facilitan la comunicación entre los diferentes softwares de e-learning existentes. SCORM estipula como se comunican los LMS con los diferentes contenidos, siendo importante no confundir la funcionalidad de SCORM con reglas de diseño, es un estándar puramente técnico. De hecho, SCORM no es un estándar como tal, fue creado por la agencia **"Advance Distributed Learning"** (un grupo de investigación financiado por el departamento de defensa de Estados Unidos) como unificación de la multitud de estándares existentes, cada uno de los cuales solucionaba una problemática diferente.

Las siglas de SCORM hacen referencia a "**Sharable Content Object Reference Model**", la traducción literal sería Objetos de contenido compartido. Este nombre proviene de la idea de crear unidades de material educativo online que puedan ser compartidas entre plataformas, SCORM define como crear estos objetos de contenido compartido (también denominados SCOs).

9.5 Futuro del aprendizaje online

xAPI, también conocido como Tin Can API, es la evolución de SCORM y el estándar utilizado en este proyecto. Si antes he mencionado que Internet fue el impulsor de las plataformas e-learning y consecuentemente de SCORM, en el caso de xAPI se podría decir que su gran impulsor han sido lo que denominamos Activity Providers.

Entendemos por Activity Providers los sistemas capaces de generar experiencias de aprendizaje. Hasta el auge de la tecnología móvil, este tipo de sistemas se basaban en los navegadores web, pero poco a poco empezó a evidenciarse que actualmente el aprendizaje no se realiza únicamente en los LMS tradicionales accedidos a través del navegador. Algunos ejemplos de estas nuevas situaciones de aprendizaje son la ya mencionada Mobile learning, simulaciones, mundos virtuales, videojuegos, actividades que tienen lugar en el mundo real, etc. xAPI elimina esta limitación.

El proyecto para desarrollar xAPI fue promovido, al igual que SCORM, por la agencia Advance Distributed Learning (ADL) ante la necesidad de modernizar SCORM. En 2011, el contrato para desarrollar xAPI fue otorgado a Rustic Software, que iniciaron el proyecto con el nombre de Tin Can. La primera versión fue lanzada en abril del 2012.

xAPI permite compartir información sobre el rendimiento de las personas, es la respuesta de la ADL ante la necesidad de recolección y rastreo de información. Es importante remarcar y entender que xAPI es una extensión de SCORM, no un reemplazo, tan solo reemplaza (potencialmente) los protocolos de comunicación de los datos, los demás componentes de SCORM tales como el empaquetado y entrega de los SCO's no son competencia de xAPI.

Otro aspecto importante de xAPI y determinante a la hora de escogerlo para la realización de este proyecto, es que permite descentralizar el contenido de los learning management systems y enviar sentencias fuera de estos. Estas sentencias o reportes se envían a bases de datos externas denominadas Learning Record Store (LRS).

10. xAPI

En este apartado vamos a profundizar en el estándar xAPI, uno de los ejes centrales en los que está basado este proyecto. Empezaremos definiendo el formato de la información recopilada y transmitida por este, así como sus características principales y las posibles alternativas existentes.

10.1 Sentencias xAPI

xAPI se basa en la especificación "Activity Stream". Esta especificación fue una colaboración entre grandes entidades corporativas, entre las cuales se encontraban Google, Facebook o Microsoft, con el fin de intercambiar experiencias sociales utilizando un formato estándar. Uno de los aspectos a destacar de Activity Stream es que es legible tanto para ordenadores como para el ser humano.

Las sentencias xAPI a las que hemos hecho mención anteriormente están escritas en Javascript Object Notation Language, similar al XML. xAPI captura este flujo de datos y lo traduce en un formato entendible.

Activity Stream incluye tres componentes principales asociados a una experiencia de aprendizaje, como son el actor, verbo y objeto (o actividad). A continuación, se pueden observar algunos ejemplos[4] (son en inglés, ya que el formato de los logs es en este idioma).

- John Connor attempted "The War of 1812, Part 1"
- John Connor watched "The Battle of New Orleans Video"
- John Connor attempted "The War of 1812, Assessment"
- John Connor answered "Question 1"
- John Connor completed "The War of 1812, Assessment"
- John Connor scored "90%" on "The War of 1812, Assessment"

[4] Ejemplo extraído de Berking, P., & Gallagher, S. (2011). Choosing a learning management system. Advanced Distributed Learning (ADL) Co-Laboratories,(2.4). Página 8

Como vemos en los ejemplos anteriores, encontramos un actor (John Connor), un verbo (attempted, watched, answered, etc.) y un objeto (The War of 1812, Question 1, etc.)

10.2 Facilidades ofrecidas por xAPI

Antes se ha señalado a xAPI como la siguiente generación de estándares que regulan los learning management systems. A continuación, se definen algunas de las ventajas que ofrece respecto los modelos anteriores.

- Soporte para diversos tipos de contenidos

Soporte para recolectar información procedente de fuentes muy variadas, desde entornos de inmersión virtual (juegos o mundos virtuales) hasta actividades del mundo real. Esto aumenta el rango de contenidos a incluir en las experiencias de aprendizaje tratadas por xAPI.

- Implementación simplificada

Los modelos de datos utilizados, interpretables por los humanos, usan un esquema universal (JSON).

- Contenido portable

El contenido ya no ha de ser entregado a un LMS ni es necesario presentarlo en un navegador web. Se puede acceder al contenido a través de cualquier herramienta o sistema que el usuario desee, pudiendo incluir o no un LMS.

- Acceso mejorado a los datos

Los datos recolectados no son dependientes de la sesión, se almacenan en un learning record store(LRS) en un formato entendible por el hombre “sujeto-verbo-objeto”. Posteriormente, estos datos pueden ser solicitados con el fin de realizar data mining y analítica.

- Soporte para escenarios offline

Este es uno de los aspectos más importantes de xAPI. Consiste en la recolección de información en dispositivos móviles, estén conectados o no. Actualmente, la generación de datos ocurre en cualquier momento y lugar, por ello, en caso de no disponer de conexión en ese momento, la información se almacena localmente para posteriormente ser enviada una vez se disponga de conectividad.

- Creación de eventos

Por último, comentar que los eventos capturados por xAPI son configurables, es decir, xAPI permite que el propio usuario defina los posibles eventos que pueden ocurrir. Esto significa que la cantidad de escenarios que pueden ser analizados no tiene límites. Algunos ejemplos de casos reales podrían ser,

    - Interacción de un alumno con un archivo de vídeo
        - Tiempo invertido por el alumno observando el vídeo
        - En qué punto del vídeo se pausó (en caso que así fuera) y por cuanto tiempo.
        - Veces que se volvió a reproducir el vídeo, y que partes
    - Realización de una simulación de vuelo
    - Detectar cuando un alumno escribe en un foro

10.3 Learning Record Store

Los learning record stores, a los cuales hemos hecho referencia en apartados anteriores, son sistemas donde se almacena lo que se denomina como “Experiencias de Aprendizaje”, los datos recopilados por xAPI. Antes de la aparición de xAPI, la función de los LRS la cumplían los LMS, de hecho, uno de los cambios introducidos por xAPI es la descentralización (previamente mencionada) de la información, de modo que para poder implementar xAPI no se requiere un LMS.

Del mismo modo que ocurre con xAPI y SCORM, un LRS es un servicio de almacenamiento en la nube que tan solo trata con experiencias de aprendizaje, en ningún caso es un sustitutivo de las funciones llevadas a cabo hasta ahora por los LMS.

Las funcionalidades de estos productos van más allá del simple almacenamiento de sentencias, permiten almacenar experiencias de aprendizaje complejas, que abarcan desde actividades del mundo real hasta eventos de aplicaciones móviles o de rendimiento laboral. La información almacenada puede ser compartida con otros sistemas de reporte analítico o para dar soporte a otros servicios web.

10.4 Alternativas a xAPI

A fecha de junio de 2016 (actualmente septiembre de 2016), xAPI ha sido adoptado por alrededor de 170 organizaciones. Si se quiere captar información que va más allá de la facilitada por SCORM y dirigirla fuera del LMS, xAPI es la mejor opción por el gran soporte que ofrece, pero no es la única opción.

Uno de los problemas que se le atribuyen a xAPI es precisamente debido a este soporte generalizado que ofrece para multitud de casos de uso, donde no se garantiza la interoperabilidad entre los eventos capturados por los diferentes learning management systems. ¿Cómo comunica el contenido al LMS (o al LRS) que he completado un curso? En realidad, podría usar "completed" o "finished" pero entonces no existe interoperabilidad, no hay reglas que tengan que seguir obligatoriamente el contenido y el LMS.

Así pues, teniendo en cuenta los potenciales problemas que esto supone, encontramos **cmi5**. Se podría definir Cmi5 como un conjunto de reglas extras para xAPI, es decir, un conjunto de criterios a seguir por todos los learning management systems, de modo que la comunicación para informar de los eventos de aprendizaje tenga una misma notación, garantizando la interoperabilidad. También se habla de cmi5 como un "perfil" para usar xAPI con los sistemas tradicionales de LMS, entendiendo por perfil un conjunto especifico de reglas y documentación para implementar xAPI en un contexto especifico, ofreciendo un vocabulario particular, creado específicamente para este contexto o a partir de otro vocabulario ya existente.

Cmi5 fue un proyecto originalmente desarrollado por la AICC (Aviation Industry Computer-Based Training Committee) en 2010. Al igual que xAPI, su objetivo era reemplazar SCORM por una solución más robusta y con mayores funcionalidades. En 2012 cmi5 se encontraba en un estado avanzado de desarrollo, próximo a su finalización, pero el proyecto Tin Can API (xAPI) ya estaba completado, y esto decantó la popularización hacia este último.

Al haber cierta similitud entre el propósito de cmi5 y xAPI, ADL (impulsores de xAPI) y AICC acordaron cooperar en un "perfil para xAPI" que cubriera casos de uso más específicos, provocando en 2012 el reinicio del proyecto. En 2014 la AICC disolvió formalmente el proyecto cmi5 y transfirió las competencias a ADL. A día de hoy, han sido lanzadas dos versiones de cmi5, Sandstone (mayo 2015), con el objetivo de recolectar feedback de los usuarios y Quartz (junio 2016), la primera versión lista para implementación.

Dada la naturaleza de este proyecto, dónde el estudio se va a basar únicamente en datos de la plataforma de Moodle y datos facilitados por otras Universidades, no se ha creído necesario la utilización de cmi5, siendo también un factor muy influyente en esta decisión el poco soporte del que dispone así como ejemplos reales.

11. Machine Learning

A lo largo de este apartado se va a presentar la otra gran tecnología sobre la que se basa el proyecto. Machine learning es un subcampo de las ciencias de la computación, de hecho, es una disciplina que combina conceptos de la inteligencia artificial, probabilidad y estadística, teoría de la información y filosofía, entre otros campos. Hoy en día, incluso para los investigadores del campo, no existe una definición ampliamente aceptada sobre machine learning. Para ello, vamos a utilizar las definiciones de dos remarcados científicos en este campo.

Según Arthur Samuel (1959), uno de los pioneros, consiste en dotar a los ordenadores con la habilidad de aprender sin haber sido programados explícitamente para ello. Samuel, estaba interesado en el ajedrez, pero no era muy buen jugador, por ello, realizó un programa que jugara un número elevado de partidas por él. A partir de todas estas experiencias, el programa fue observando con que posiciones en el tablero se conseguían victorias y con cuáles no. Al final, el programa aprendió a diferenciar entre las buenas y malas posiciones de cada ficha, llegando incluso a jugar mejor que el propio Arthur Samuel.

La segunda definición[5], mas técnica y actual, la realizó Tom Mitchell (1998), investigador de las ciencias de la computación, conocido por sus aportaciones al campo del machine learning, inteligencia artificial y a la neurociencia cognitiva. Según Mitchell, un programa de ordenador se dice que aprende de la experiencia E, con respecto a una tarea T y una medida de rendimiento P, si el rendimiento sobre T, medido por P, mejora con la experiencia E.

Como se ha comentado en más de una ocasión a lo largo de este documento, el rango de casos de uso que abarca el machine learning es prácticamente ilimitado, pudiéndose aplicar en cualquiera de los sectores existentes. Algunos de estos casos de uso son las predicciones en transacciones financieras, diagnóstico médico, marketing (personalización de los anuncios), detección de fraude, búsquedas online, interpretación del lenguaje (escrito y oral), vehículos inteligentes, etc.

Por último, en cuanto a definiciones sobre machine learning, me gustaría aportar una última definición personal definiendo el machine learning como el estudio de algoritmos que mejoran automáticamente a través de las experiencias. En los apartados siguientes se van a ir presentando los distintos algoritmos utilizados en el proyecto.

[5] Definición extraída de Mitchell, T.M., 1997. Machine Learning, Annual Review Of Computer Science. McGraw-Hill, Inc., New York, NY, USA. doi:10.1145/242224.242229. Página 2

11.1 Principales tipos de estrategias

En cuanto al tipo de metodología utilizada por los diferentes algoritmos existentes, se podría realizar una clasificación general entre dos tipos principales. Uno de ellos son los algoritmos basados en **aprendizaje supervisado**. La base de estos algoritmos consiste en la realización de una fase inicial de entrenamiento con el objetivo de crear un modelo capaz de llevar a cabo predicciones. Posteriormente, las consultas de las diferentes instancias se realizarán sobre este modelo.

En la denominada fase de entrenamiento, se utilizan un conjunto de datos formados por dos pares de objetos. Por un lado, encontramos los diferentes atributos de los parámetros de entrada, y el otro objeto es el resultado generado por esos datos. En el caso particular de este proyecto, el primer objeto serían los valores de rendimiento del alumno, y el segundo sería su resultado al final del curso (aprobado o suspenso). Por lo tanto, para entrenar un modelo predictivo, se requiere que estos datos sean concluyentes, es decir, que se sepan los datos de entrada y el resultado que estos producen.

En esta fase de entrenamiento, el objetivo consiste en crear una función capaz de predecir el resultado a partir de unos datos de entrada. Cuanto mayor sea el número de datos utilizados para entrenar el modelo, mejor se ajustará la función a estos. Una vez tenemos el modelo (o lo que es equivalente, la función ha sido ajustada), podemos realizar consultas sobre este.

El otro tipo de algoritmo utilizado en el proyecto, es el **aprendizaje basado en instancias** (también referido como aprendizaje basado en memoria). El aprendizaje en este tipo de algoritmos consiste simplemente en almacenar los datos que se utilizan para entrenar los modelos en el caso del aprendizaje supervisado. Es importante remarcar que en este tipo de algoritmos no existe un modelo. Cuando se recibe una nueva consulta, se busca entre los datos almacenados aquellas instancias cuyos atributos sean similares a los de la instancia consultada. El resultado retornado será el valor que comparta con la mayoría de ellas.

Este tipo de metodología permite utilizar datos específicos para cada tipo de instancia en vez de entrenar un modelo para casos generales. De hecho, es la practica más habitual cuando se utilizan este tipo de algoritmos, se intentan utilizar datos específicos para el tipo de consulta, mejorando así el resultado obtenido.

A su vez, aunque este tipo de algoritmos parezcan a priori más sencillos que los anteriores, una fase muy importante consiste en la clasificación de los datos, es decir, el proceso por el cual se determinan que datos son “similares” a la instancia sobre la que se desea realizar la predicción. En muchas ocasiones, esta fase de clasificación puede resultar computacionalmente costosa.

Otro inconveniente que encontramos es debido a que estos algoritmos a la hora de estipular si dos instancias son similares tienen en cuenta el valor de todos los atributos por igual. Imposibilitando la priorización de unos sobre otros.

11.2 K-Nearest Neighbour Learning

Este algoritmo pertenece a la familia del aprendizaje basado en instancias. Como hemos mencionado en el apartado anterior, uno de los puntos más importantes dentro de esta metodología consiste en determinar que instancias son parecidas entre si. Para ello, este algoritmo asume que todas las instancias corresponden a puntos en un espacio n-dimensional, $\mathbb{R}^n$. Los "vecinos más cercanos" de una instancia se determinan utilizando por defecto la distancia Euclidiana.

Una definición más precisa del parágrafo anterior consiste en describir una instancia x como $<a_1(x), a_2(x),...,a_n(x)>$, siendo $a_r(x)$ el valor del atributo r para la instancia x. La distancia entre dos instancias x_i y x_j se define como $d(x_i, x_j)$, siendo su representación matemática:

$$d(x_i, x_j) \equiv \sqrt{\sum_{r=1}^{n} (a_r(x_i) - a_r(x_j))^2}$$

Figura 3: Formula de la distancia Euclidiana[6]

La distancia Euclidiana es un caso específico de la distancia Minkowski, la cual definimos como

$$d(\mathbf{x}, \mathbf{y}) = \left(\sum_{i=0}^{n-1} |x_i - y_i|^p \right)^{1/p}$$

Figura 4: Formula de la distancia Minkowski[7]

Algunos de los casos especiales que podemos encontrar son la distnacia Manhattan (p=1), distancia Euclidiana (p=2) o la distancia Chebyshev (p tiende a infinito).

[6] Formula extraída de la página 232 de Mitchell, T.M., 1997. Machine Learning, Annual Review Of Computer Science. McGraw-Hill, Inc., New York, NY, USA. doi:10.1145/242224.242229

[7] Formula extraída de https://wikimedia.org/api/rest_v1/media/math/render/svg/33aa1151bd324808aeb7d7bd1262f6b8c515ec14

11.3 Random forest

Por el contrario, Random forest es un tipo de algoritmo basado en aprendizaje supervisado. En este caso, la función de aprendizaje se representa a través de un árbol de decisión. Antes de explicar en qué consiste exactamente el random forest hay que empezar por los arboles de decisión, ya que los random forest son una variación de estos.

Un árbol de decisión se puede traducir como un conjunto de simples reglas if-then. Es uno de los métodos más populares dentro de los algoritmos de inferencia inductiva (a través de juicios particulares se estipulan reglas universales), utilizado en un amplio rango de tareas, desde diagnóstico médico hasta determinar el riesgo a la hora de conceder préstamos, por ejemplo.

Estos árboles clasifican las instancias ordenándolas desde la raíz del árbol hasta las hojas, las cuales determinan la clasificación final de la instancia (o resultado final). Dicho de otro modo, todos los nodos desde la raíz hasta la hoja son los diferentes atributos de la instancia, la hoja es el resultado final. En la siguiente imagen se representa la idea que acabamos de explicar.

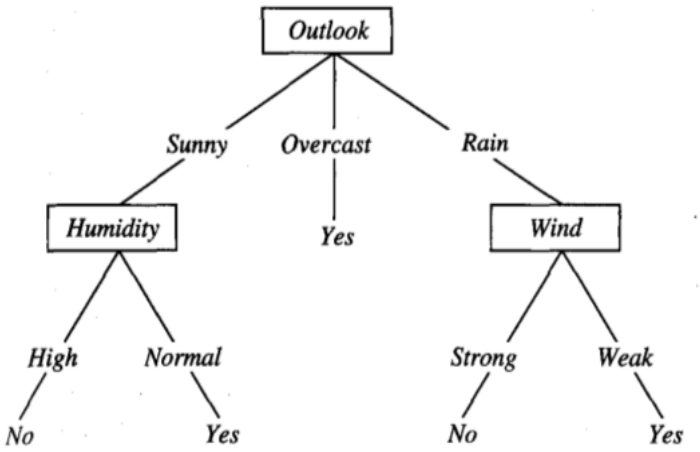

Figura 5: Ejemplo árbol de decisión para el caso de uso "Jugar a tenis" [8]

La imagen anterior corresponde a un árbol de decisión utilizado para determinar el concepto "Jugar a tenis". Como vemos, los atributos utilizados por las instancias son el pronóstico temporal, la humedad y el viento. Cada uno de estos atributos tiene un peso determinado sobre el resultado final, a partir de sus valores obtendremos el resultado final, en este caso si se jugará a tenis o no.

[8] Ejemplo extraído de la figura 3.1 Mitchell, T.M., 1997. Machine Learning, Annual Review Of Computer Science. McGraw-Hill, Inc., New York, NY, USA. doi:10.1145/242224.242229

Una vez entendido como funciona un árbol de decisión simple, random forest es una combinación de distintos arboles de decisión. La idea de random forest consiste en crear diferentes arboles de decisión a partir de *n* grupos de datos distintos. Para generar un random forest, los pasos a seguir son,

1. Creamos *n* grupos de estudiantes a partir del dataset, cada uno de estos grupos se asignará a la creación de un árbol diferente.
2. Para cada árbol, se escoge uno de los dos parámetros a tener en cuenta, en nuestro caso, suma de clicks y días de margen. Supongamos que escogemos la suma de clicks, entonces nuestro árbol intentará determinar el valor de suma de clicks que mejor divide al grupo de estudiantes con el que está tratando, todos los alumnos con un valor inferior a ese se marcan como suspendidos (rama izquierda), sino van a la rama derecha (aprobados).
3. Hacemos lo mismo que en el apartado 2, pero para la otra variable, los días de margen a la hora de entregar una práctica. A diferencia del apartado 2, en este caso ya sabemos que los alumnos que han ido a la rama izquierda en la primera clasificación han sido marcados como suspendidos, por lo tanto, no tiene sentido continuar por esa línea y tan solo seguimos desarrollando el árbol por la rama derecha.
4. Una vez no se dispone de más variables que utilizar para clasificar, el árbol se considera entrenado.

Cada uno de los arboles generados realiza una predicción sobre la instancia de manera individual, el resultado de esta predicción se denomina “voto”. La predicción devuelta por random forest es la clasificación que haya obtenido el mayor número de votos entre todos los árboles.

11.4 Redes neuronales

Este último algoritmo del que vamos a hablar, al igual que en el caso del random forest, es ampliamente utilizado, está focalizado en la interpretación de datos complejos generados por el mundo real. Entre los casos de uso más destacados encontramos el reconocimiento escrito, vocal e interpretación de imágenes.

Como se puede deducir del nombre, la idea de este algoritmo consiste en imitar el modelo de comunicación que sigue el cerebro humano, donde encontramos una compleja red de neuronas interconectadas entre si. Explicándolo de forma sencilla, se puede definir una neurona como una unidad simple que recibe unos datos de entrada y produce una salida. Continuando con la explicación anterior, la actividad de una neurona es promovida por las conexiones que dispone con otras neuronas, por lo tanto, los datos de entrada que recibe cada neurona pueden ser a su vez datos de salida de otras neuronas. Así pues, la idea que trata de imitar este algoritmo consiste en utilizar pequeñas unidades de cálculo cuyos datos de salida sean utilizados por otras unidades.

Antes de explicar cómo funcionan las redes neuronales vamos a centrarnos en el funcionamiento de una unidad individual y, después, lo extenderemos a casos más complejos. Las unidades que forman las redes neuronales, referidas como perceptrons, utilizan como datos de entrada un vector donde se encuentran los diferentes atributos de cada instancia, realizan una combinación lineal de los elementos de este vector y, finalmente devuelven un resultado. El funcionamiento del algoritmo depende de la relación entre los datos (lineales o no lineales).

Para la fase de entrenamiento vamos a distinguir entre dos tipos de metodologías, dependiendo de si los datos que estamos tratando son linealmente separables o no. En el primero de los casos, el más sencillo, partimos de unos datos linealmente separables donde el resultado devuelto por los perceptrons viene determinado por una función denominada *función de activación*, siguiendo un criterio similar al siguiente

$$o(x_1, \ldots, x_n) = \begin{cases} 1 \text{ if } w_0 + w_1 x_1 + w_2 x_2 + \cdots + w_n x_n > 0 \\ -1 \text{ otherwise} \end{cases}$$

Figura 6: Definición de la función de activación[9]

Donde x_1 corresponde al primer atributo de la instancia y w_1 al peso (o relevancia) de ese atributo, utilizado para indicar su contribución al resultado final. Adicionalmente, en este caso, al tratarse de unos datos linealmente separables, observamos la existencia de w_0, el cual actúa como threshold, es decir, su valor se utiliza para clasificar el resultado de la predicción, si la suma de los pesos de los atributos es mayor que w_0 se devuelve 1, en caso contrario -1. El caso específico de la anterior función de activación se denomina *Binary step*, es la más básica, la cual deriva de *Rectified Linear Unit* (relu).

[9] Formula extraída de la página 86 de Mitchell, T.M., 1997. Machine Learning, Annual Review Of Computer Science. McGraw-Hill, Inc., New York, NY, USA. doi:10.1145/242224.242229

Durante esta fase de entrenamiento se realizan diferentes iteraciones, normalmente el valor de los pesos de cada atributo se asigna inicialmente de forma aleatoria, por lo que en las primeras iteraciones el resultado de la predicción no es fiable. Los pesos de cada atributo se van modificando cuando una instancia es clasificada erróneamente. Este proceso se repite tantas veces como sea necesario hasta que el perceptron clasifica correctamente todas las instancias.

La actualización de los pesos en cada iteración viene determinada acorde a la regla

$$w_i \leftarrow w_i + \Delta w_i$$

Figura 7: Definición de la actualización de los pesos/iteración[10]

Donde

$$\Delta w_i = \eta(t - o)x_i$$

Figura 8: Definición del valor añadido al peso/iteración[10]

En este caso, *t* representa el output que le corresponde a la instancia, *o* al output generado por el perceptron y ŋ es una constante positiva denominada *learning rate.* El objetivo de esta constante consiste en moderar el grado de variación de los pesos en cada una de las iteraciones. Así pues, como podemos observar, si el output generado por el percepton es igual al esperado, *t-o = 0*, no se actualizará el peso del atributo en la próxima iteración, en cambio, en caso de que sea diferente al esperado, ŋ evitará un exceso de variación respecto al valor anterior. ŋ suele tener un valor similar a 0.01.

Si los datos con lo que se va a trabajar no son linealmente dependientes, los pasos a seguir son ligeramente diferentes. La idea es muy similar, se inicializan los diferentes pesos de cada atributo con valores aleatorios y se procede a realizar la predicción, pero en este caso, el proceso de entrenamiento no se repite hasta que se consigue una clasificación perfecta, ya que la naturaleza de los datos no lo permite. Por lo tanto, se realizan tantas iteraciones como se consideren necesarias hasta que el error asociado a la red neuronal tiene un valor suficientemente aceptable. Este error se calcula acorde a la siguiente formula.

$$E(\vec{w}) \equiv \frac{1}{2}\sum_{d \in D}(t_d - o_d)^2$$

Figura 9: Calculo del error de una red neuronal[10]

[10] Formulas extraídas de la página 88 y 89 de Mitchell, T.M., 1997. Machine Learning, Annual Review Of Computer Science. McGraw-Hill, Inc., New York, NY, USA. doi:10.1145/242224.242229

Donde D corresponde al conjunto de los ejemplos (instancias) utilizados durante el proceso de entrenaminto, siendo t_d y o_d el output de una instancia concreta dentro del conjunto, igual que en la figura 8, *t* representa el output de la instancia y *o* el calculado por la red neuronal. Una vez calculado el error presente en cada iteración, para ajustar los pesos se utiliza el denominado algoritmo *gradient descent,* para entenderlo resulta útil visualizar la siguiente gráfica.

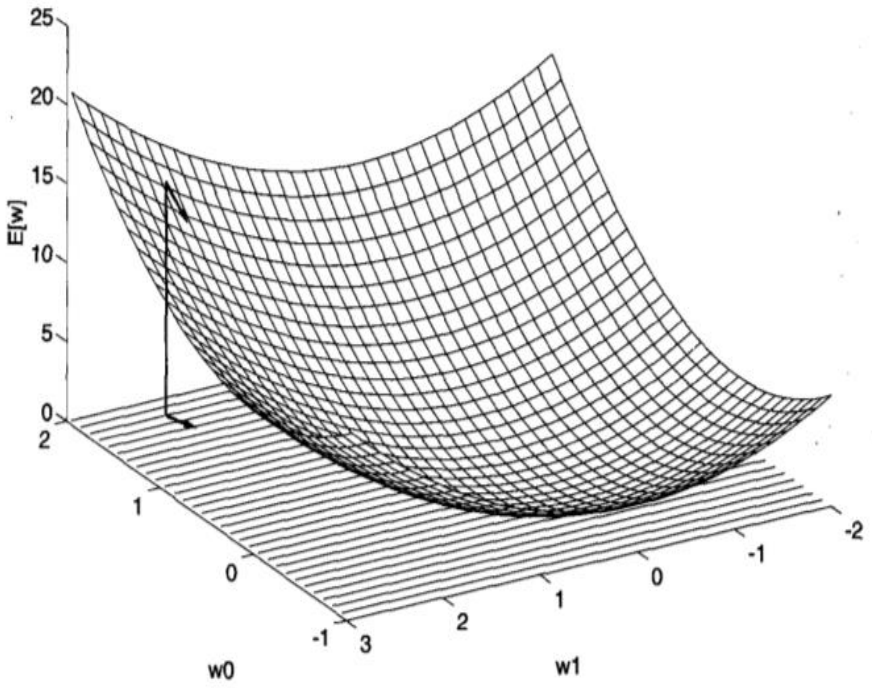

Figura 10: Gráfica donde las ordenadas corresponden al error de la red neuronal y las coordenadas representan los pesos de los atributos[11]

El plano anterior corresponde a todos los pesos posibles para los atributos w_0 y w_1 junto con su repercusión en el error asociado. El algoritmo gradient descent, por lo tanto, busca el valor de los pesos de cada atributo que en conjunto consigue minimizar el valor de error E.

En este caso, como se puede deducir, la función de activación no es tan simple como la mostrada para los casos en que los datos son linealmente separables. Encontramos una gran variedad de funciones utilizadas, entre las más destacadas, la tangente hiperbólica (tanH), logística (sigmoide), identidad, o la antes mencionada relu.

[11] Gráfica extraída de la figura 4.4 de Mitchell, T.M., 1997. Machine Learning, Annual Review Of Computer Science. McGraw-Hill, Inc., New York, NY, USA. doi:10.1145/242224.242229

Una vez explicadas las bases de las redes neuronales, en todos los ejemplos anteriores, tanto si los datos son linealmente separables o no, estamos planteando una toma de decisión lineal, es decir, si sucede X entonces Z. Pero como hemos mencionado anteriormente, las redes neuronales se aplican a casos de uso complejos, donde el resultado de una predicción puede estar formado por una gran variedad de decisiones no lineales. La siguiente imagen representa la estructura básica de una red neuronal.

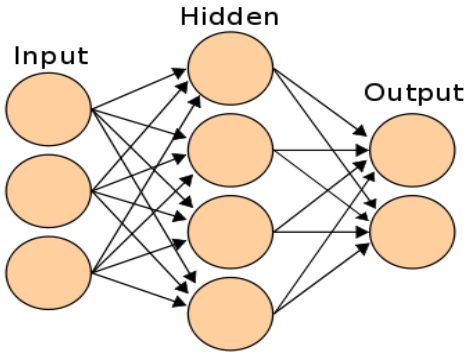

Figura 11: Ejemplo de una estructura para una red neuronal[12]

Como se puede observar, hay una primera capa (Input) que recibe el vector de atributos de la instancia inicial. Esta primera capa realiza una clasificación (o predicción) tal y como hemos explicado anteriormente. El resultado generado por esta es procesado por la siguiente capa, denominada hidden layer, ya que su existencia es transparente a nosotros.

La hidden layer, al igual que la inicial, realiza una nueva clasificación a partir de los datos facilitados por la capa anterior. Su resultado, a su vez, puede servir como datos de entrada para una nueva capa intermedia (no hay número máximo de hidden layers) o para la capa final (output). Finalmente, el resultado (nodo) cuyo valor sea más alto, será la clasificación final para la instancia que se esté procesando.

[12] Imagen obtenida de https://en.wikipedia.org/wiki/Artificial_neural_network#/media/File:Artificial_neural_network.svg

Una forma de entender la explicación anterior es a través de un ejemplo práctico, para ello vamos a utilizar el reconocimiento vocal. Supongamos que tenemos una red capaz de distinguir entre diez posibles vocales, todas ellas utilizadas en pablaras “h-d”, tales como “hid, had, head, hood”, etc.

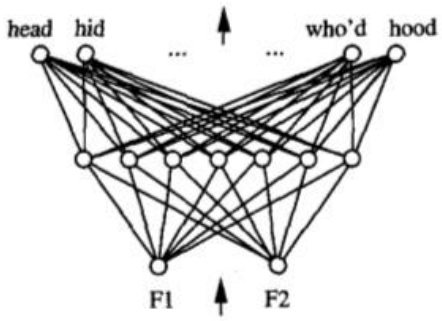

Figura 12:Ejemplo de caso de uso de una red neuronal para el reconocimiento de palabras[13]

En la imagen anterior vemos como la red neuronal recibe dos datos de entrada, F1 y F2, obtenidos a partir del análisis espectral de un sonido. En esta primera capa, cada uno de los nodos clasificará los sonidos dependiendo de la vocal que utilizan, esta clasificación inicial se realizará a nivel general, es decir, se trata de una primera clasificación según si utilizan el sonido “i” o el sonido “o”.

En este caso, la hidden layer ya dispondrá entonces de información sobre el tipo de vocal contenida en la palabra, y procederá a realizar una clasificación más específica, dentro del sonido “i” distinguirá si se trata de un “ea” o de una “i”.

Por último, como se ha mencionado antes, el nodo de la output layer cuya puntuación sea más alta corresponderá con la palabra emitida.

[13] Ejemplo extraído de la figura 3.1 Mitchell, T.M., 1997. Machine Learning, Annual Review Of Computer Science. McGraw-Hill, Inc., New York, NY, USA. doi:10.1145/242224.242229

12. Sistema predictivo

En este apartado se explica la topología del sistema predictivo, su funcionamiento, todos los elementos que forman parte de él, cómo se ha procedido al cálculo del rendimiento, así como el resultado obtenido por los diferentes algoritmos mencionados anteriormente.

12.1 Topología del sistema

Tal y como se comentó en el apartado de recursos utilizados, a lo largo del proyecto se ha trabajado con un portátil personal y un servidor alquilado. En el portátil personal se encuentra alojado el curso de Moodle sobre el que se han ido realizando las distintas pruebas. Por su parte, el servidor dispone del portal web y las distintas bases de datos.

Cuando un alumno realiza una actividad en el curso pertinente de Moodle, se crea un registro sobre dicha actividad (de forma transparente al usuario) denominado Moodle Logstore. Para poder adaptar estos registros al formato xAPI y enviarlos al LRS seleccionado utilizamos **Logstore xAPI.**

Logstore xAPI es un plugin oficial de Moodle que genera sentencias xAPI a partir de la información contenida en el Logstore. Posteriormente, estas sentencias xAPI se envían al LRS que se haya especificado. Una vez instalado el plugin en el servidor donde se encuentre Moodle, cualquier evento almacenado en el Logstore será automáticamente procesado por Logstore xAPI y enviado al LRS.

Por su parte, el LRS escogido ha sido Learning Locker. Es el LRS de referencia, cuya utilización es recomendable si se va a trabajar con xAPI. El uso que se le va a dar durante el proyecto va a ser meramente el de una base de datos de sentencias xAPI, pero el rango de funcionalidades ofrecidas por Learning Locker es mucho más amplio.

Cada día a las 00:00 se ha programado una cron task en el servidor que ejecuta un script en PHP para recoger todas las sentencias almacenadas en Learning Locker del día anterior. Una vez disponemos de ellas se lleva a cabo el procesado de cada una de las sentencias y la extracción de la información utilizada en las predicciones. La extracción de las sentencias almacenadas en Learning Locker se hace a través de **TinCanPHP,** una librería escrita en PHP que facilita el acceso y la recolección de las sentencias.

Por último, una vez disponemos de los parámetros de rendimiento de cada alumno actualizados a la fecha actual, se procede a ejecutar un script en Python que implementa el algoritmo predictivo. Este script utiliza la librería **Scikit-learn**. Dicha librería integra un amplio abanico de algoritmos de machine learning, intenta focalizarse en ofrecer soluciones de machine learning a usuarios no especializados en la materia a través de un lenguaje de alto nivel, hace especial énfasis en la facilidad de uso, así como el buen rendimiento obtenido y la cantidad de documentación disponible. No dispone de muchas dependencias, se distribuye bajo la licencia de BSD, tanto para uso académico como comercial. El código fuente como los binarios y la documentación pueden descargarse de http://scikit-learn.sourceforge.net.

Se ha escogido Scikit-learn ya que es una solución open source con más de 500 colaboradores, mientras que otras soluciones parecidas eran de uso comercial. En cuanto al rango de funcionalidades ofrecidas, es la librería que implementa un mayor número de algoritmos (un total de 9), mientras que otras soluciones se centran en un único algoritmo. Por último, para acabar de tomar la decisión, se consultó con compañeros del departamento de Computer Science de la UPC que trabajan con esta tecnología en su día a día. Algunas de las otras opciones existentes son GraphLab (comercial), Pylearn2 o Shogun.

12.2 Formato sentencias almacenadas en Learning Locker

Como se ha descrito en la topología del sistema, cada día a las 00:00 se ejecuta un script que recopila todas las sentencias almacenadas en nuestro LRS y las procesa. A continuación, vamos a detallar las partes más relevantes de dichos registros. Para ello, vamos a utilizar como ejemplo el statement "Angel viewed Curso primera prueba".

El primer campo que vamos a analizar es el actor. Actor es el objeto que define quien lleva a cabo la acción, es el primer objeto que aparece en el registro.

```
"actor": {
  "objectType": "Agent",
  "name": "Angel",
  "account": {
    "name": "2",
    "homePage": "http://127.0.0.1"
  }
}
```

Figura 13: Definición en JSON del campo actor

El tipo de objeto (propiedad objectType) puede ser un Agente, como en este caso, haciendo referencia a un único individuo, o un grupo, en cuyo caso se habla de una agrupación de agentes y dispondría de una propiedad adicional denominada member dónde se indican los miembros que forman parte del grupo. La propiedad nombre es simplemente el nombre completo del agente. Aunque no se explicite en este ejemplo, es muy común encontrar también el campo mbox, que actúa como identificador adicional del actor, tiene un formato mailto:email address.

El siguiente campo es el verbo, el cual define la acción entre el actor y la actividad. xAPI no especifica ningún verbo en particular, en vez de eso, deja crear verbos adicionales de modo que los usuarios puedan adaptar las sentencias para que tengan significado en cada caso particular. Aun así, hay una lista predefinida, con ciertas limitaciones, incapaz de capturar todos los eventos posibles, pero puede resultar útil para casos de uso que no requieren de vocabulario específico.

Este objeto dispone solo de dos campos, el id, que es una url donde podemos consultar información del verbo (idioma en que se muestra y significado) y la representación del verbo (dispaly), tal como aparecerá escrito en el statement.

```
"verb": {
    "id": "http://id.tincanapi.com/verb/viewed",
    "display": {
        "en": "viewed"
    }
}
```

Figura 14: Definición en JSON del campo verbo

El objeto que vamos a describir a continuación es el context. Es un campo opcional que ofrece información contextual sobre la sentencia, es decir, aumenta el significado de la sentencia aportando información extra. Un ejemplo del tipo de información que ofrece este campo podría ser el equipo con el que está trabajando el actor de la acción (en caso de ser un investigador) o la altitud con la que se intentó un escenario de simulación de vuelo, etc.

En este caso, en los resultados guardados por Learning Locker encontramos:

```
"contextActivities": {
        "category": [
            {
                "objectType": "Activity",
                "id": "http://moodle.org",
                "definition": {
                    "type": "http://id.tincanapi.com/activitytype/source",
                    "name": {
                        "en": "Moodle"
                    },
                    "description": {
                        "en": "Moodle is a open source learning platform designed to provide educators, administrators and learners with a single robust, secure and integrated system to create personalised learning environments."
                    }
                }
            }
        ],
```

Figura 15: Definición en JSON del campo contextActivities

ContextActivity es el objeto que nos muestra esta información adicional. Dentro de ContextActivity el primer campo que nos encontramos és Category, como su nombre indica, se utiliza para categorizar la sentencia, aplicado al ejemplo que estamos utilizando sería "Angel viewed Curso primera prueba y la sentencia fue capturada a través de Moodle". Nos aporta información adicional sobre el statement, en este caso dónde ha sido capturado el evento.

El otro campo que encontramos en ContextActivity es Grouping, dónde se especifican otras actividades con las que está relacionado nuestro statement.


```
"grouping": [
  {
    "objectType": "Activity",
    "id": "http://127.0.0.1",
    "definition": {
      "type": "http://id.tincanapi.com/activitytype/site",
      "name": {
        "en": "Proyecto final Machine Learning"
      },
      "description": {
        "en": "Proyecto final Machine Learning"
      }
```


Figura 16: Definición en JSON del campo grouping

En este caso nos indica que "Curso primera prueba" es un curso que pertenece al sitio Proyecto final Machine Learning (nombre con el que se definió el espacio virtual de Moodle utilizado en el proyecto).


```
"timestamp": "2016-12-15T11:17:04+01:00",
  "object": {
    "objectType": "Activity",
    "id": "http://127.0.0.1/course/view.php?id=2",
    "definition": {
      "type": "http://lrs.learninglocker.net/define/type/moodle/course",
      "name": {
        "en": "Curso primera prueba"
      },
      "description": {
        "en": "Este curso es una pruba para ver la informaci\u00f3n que se envia a Learning locker cuando completas assignements, cuando lees/abres material del curso..."
      }
```


Figura 17: Definición en JSON del campo objeto

En el fragmento de json de la figura 17 encontramos el timestamp y el objeto (o Activity, como me voy a referir a él para evitar confusiones) de la sentencia. El timestamp es un objeto creado por el LRS que indica el momento en que tuvo lugar el evento.

La Activity define la "cosa" sobre la que tuvo lugar la acción. Una activity puede ser una actividad, otro actor o incluso un substatement.[14]

- The Object is an Activity: "Jeff wrote an essay about hiking."
- The Object is an Agent: "Nellie interviewed Jeff."
- The Object is a SubStatement or Statement Reference (different implementations, but similar when human-read): "Nellie commented on 'Jeff wrote an essay about hiking.'"

En el caso de nuestra sentencia el objeto es *Curso primera prueba*. A parte del tipo de objeto, también encontramos una descripción del objeto (extraida de Moodle), el nombre, y el tipo (curso, asignment, quiz, chat, etc.)

Por último, vamos a hablar del campo **extensión,** que se ha omitido en los apartados anteriores para evitar un exceso de información. **Extension** es un campo que forma parte de la definición de una actividad, similar a los contextActivities, pero en vez de aplicar sobre el statement lo hace sobre la Activity. A continuación, en la figura 18, se muestra el objeto extension que referencia a *Curso primera prueba*.

[14] Ejemplo extraído de https://github.com/adlnet/xAPI-Spec/blob/master/xAPI-Data.md, sección 2.2.4

```
"extensions": {
        "http://lrs.learninglocker.net/define/extensions/moodle_course": {
          "id": "2",
          "category": "1",
          "sortorder": "10001",
          "fullname": "Curso primera prueba",
          "shortname": "Primera prueba",
          "idnumber": "1234",
          "summary": "<p>Este curso es una pruba para ver la informaci\u00f3n que se envia a Learning locker cuando completas assignements, cuando lees/abres material del curso...<br></p>",
          "summaryformat": "1",
          "format": "weeks",
          "showgrades": "1",
          "newsitems": "5",
          "startdate": "1478991600",
          "marker": "0",
          "maxbytes": "0",
          "legacyfiles": "0",
          "showreports": "0",
          "visible": "1",
          "visibleold": "1",
          "groupmode": "0",
          "groupmodeforce": "0",
          "defaultgroupingid": "0",
          "lang": "",
          "calendartype": "",
          "theme": "",
          "timecreated": "1479055517",
          "timemodified": "1479055517",
          "requested": "0",
          "enablecompletion": "1",
          "completionnotify": "0",
          "cacherev": "1481559069",
          "type": "course",
          "url": "http://127.0.0.1/course/view.php?id=2"
        }
      }
```


Figura 18: Definición en JSON del campo extensions

Como se puede observar en la figura anterior hay una gran cantidad de información adicional sobre el curso, la fecha de creación, la de la última modificación, si el curso aparece visible en Moodle, tipo de objeto, etc. Este tipo de información puede ser utilizada en un sin fin de aplicaciones, en el caso particular de este proyecto se utilizan algunas propiedades de este campo.

12.3 Hipótesis utilizadas en la predicción

Para poder determinar el rendimiento de un estudiante son necesarios unos indicadores con los cuales poder medir este parámetro, para ello, me he basado en el paper "*Indicators of Good Student Performance in Moodle Activity Data*" realizado por Ewa Młynarska, Derek Greene y Pádraig Cunningham de Insight Centre, University College Dublin, Ireland.

En este artículo de investigación se analizan una gran cantidad de datos sobre la actividad de los alumnos en cursos de Moodle con el fin de determinar si existen indicadores tempranos sobre el buen o mal rendimiento de los estudiantes. Las tres hipótesis en las que se basan son:

1. Entregar una práctica (o entregable) con un margen amplio respecto la fecha límite es un indicador de los alumnos que consiguen buenas cualificaciones finales.
2. Un alto nivel de interactividad en el curso (de Moodle en este caso) antes de las entregas de las diferentes tares es un indicador de buen rendimiento.
3. La actividad nocturna (o por la tarde) es un mejor indicador de rendimiento que la actividad diurna.

Uno de los problemas que se ha encontrado durante el desarrollo del proyecto ha sido la dificultad para conseguir datos válidos y reales sobre estudiantes que hayan realizado cursos utilizando plataformas virtuales debido principalmente a las estrictas políticas de privacidad a las que están sujetos estos datos. La información contenida en el dataset utilizado es bastante extensa y con ella se han podido utilizar en el proyecto dos de las tres hipótesis anteriormente descritas para calcular el rendimiento de los estudiantes.

12.4 Estructura del dataset

El dataset utilizado para el proyecto es "Kuzilek, J., Hlosta, M., Herrmannova, D., Zdrahal, Z. and Wolff, A. OU Analyse: Analysing At-Risk Students at The Open University. Learning Analytics Review, no. LAK15-1, March 2015, ISSN: 2057-7494.".

En este apartado vamos a listar las diversas tablas presentes en el dataset utilizado. Los tipos de datos son los correspondientes a MySQL, software utilizado para la creación de todas las bases de datos del proyecto.

- courses

Los diferentes cursos sobre los que se recopilaron los datos.

Nombre	Tipo	Definición
code_module	VARCHAR	Nombre del módulo (o asignatura), ejerce de identificador
code_presentation	VARCHAR	Año y cuatrimestre en que tiene lugar el curso, siendo 2017B (febrero) o 2017J(octubre)
module_presentation_length	INTEGER	Longitud del curso en días.

Tabla 10: Definición de la tabla courses del dataset utilizado en el proyecto

- assessments

Las prácticas presentes en cada uno de los cursos

Nombre	Tipo	Definición
code_module	VARCHAR	Nombre del módulo (o asignatura), ejerce de identificador
code_presentation	VARCHAR	Año y cuatrimestre en que tiene lugar el curso, siendo 2017B (febrero) o 2017J(octubre)
id_assessment	INTEGER	Identificador de la práctica

assessment_type	VARCHAR	Tipo de entregable, se distinguen Tutor Marked Assessment (TMA), Computer Marked Assessment (CMA) and Final Exam (Exam).
date	INTEGER	Información sobre la fecha final de entrega, calculada como el número de días desde el inicio del module-presentation. La fecha inicial tiene valor 0
weight	FLOAT	Valor en % de la práctica

Tabla 11: Definición de la tabla assessments del dataset utilizado en el proyecto

- vle

Los materiales del curso, tales como páginas web, archivos pdf's, etc.

Nombre	Tipo	Definición
id_site	INTEGER	Identificador del material
code_module	VARCHAR	Nombre del modulo (o asignatura), ejerce de identificador
code_presentation	VARCHAR	Año y cuatrimestre en que tiene lugar el curso, siendo 2017B (febrero) o 2017J(octubre)
activity_type	VARCHAR	El rol asociado a este material
week_from	INTEGER	Semana desde la que se planea utilizar el material
week_to	INTEGER	Semana hasta la que se planea utilizar el material.

Tabla 12: Definición de la tabla vle del dataset utilizado en el proyecto

- studentInfo

La información de los alumnos sobre los que se han recopilado los datos

Nombre	Tipo	Definición
code_module	VARCHAR	Nombre del módulo (o asignatura), ejerce de identificador
code_presentation	VARCHAR	Año y cuatrimestre en que tiene lugar el curso, siendo 2017B (febrero) o 2017J(octubre)
id_student	INTEGER	Identificador del estudiante
gender	VARCHAR	Genero del estudiante
region	VARCHAR	Región geográfica a la que pertenece el estudiante
Highest_education	VARCHAR	Nivel máximo educativo del estudiante cuando se registró en el curso
imd_band	VARCHAR	Índice de *Multiple Depravation* basado en la región donde vive el alumno. Es un estudio que llevó a cabo el gobierno de UK
age_band	VARCHAR	Franja de edad del estudiante
num_of_prev_attempts	INTEGER	Número de intentos previos en la asignatura (code_module)
studied_credits	INTEGER	Número total de créditos cursados este curso-cuatrimestre
disability	VARCHAR	Indicador de minusvalía
final_result	VARCHAR	Resultado final del curso

Tabla 13: Definición de la tabla studentInfo del dataset utilizado en el proyecto

- studentAssessment

Tabla que almacena la interacción del estudiante con el material del curso

Nombre	Tipo	Definición
id_assessment	INTEGER	Identificador de la práctica
id_student	INTEGER	Identificador del estudiante
date_submitted	INTEGER	Fecha en la que se entregó, calculada como el número de días desde el inicio del curso
is_banked	INTEGER	Indicador sobre si el entregable ha sido almacenado de una convocatoria anterior (en caso de que se esté repitiendo)
score	FLOAT	Nota obtenida en el entregable

Tabla 14: Definición de la tabla studentAssessment del dataset utilizado en el proyecto

- studentVle

Tabla que almacena la interacción del estudiante con el material del curso

Nombre	Tipo	Definición
code_module	VARCHAR	Nombre del módulo (o asignatura), ejerce de identificador
code_presentation	VARCHAR	Año y cuatrimestre en que tiene lugar el curso, siendo 2017B (febrero) o 2017J(octubre)
id_student	INTEGER	Identificador del estudiante
Id_site	INTEGER	Identificador del material
date	INTEGER	Fecha en la que el alumno interaccionó con el material, calculada como el número de días desde el inicio del curso
sum_click	FLOAT	Número de interacciones del alumno con el material para ese mismo día.

Tabla 15: Definición de la tabla studentVle del dataset utilizado en el proyecto

En la siguiente figura podemos observar el diagrama de UML de la base de datos que se ha creado a partir de los datos del dataset.

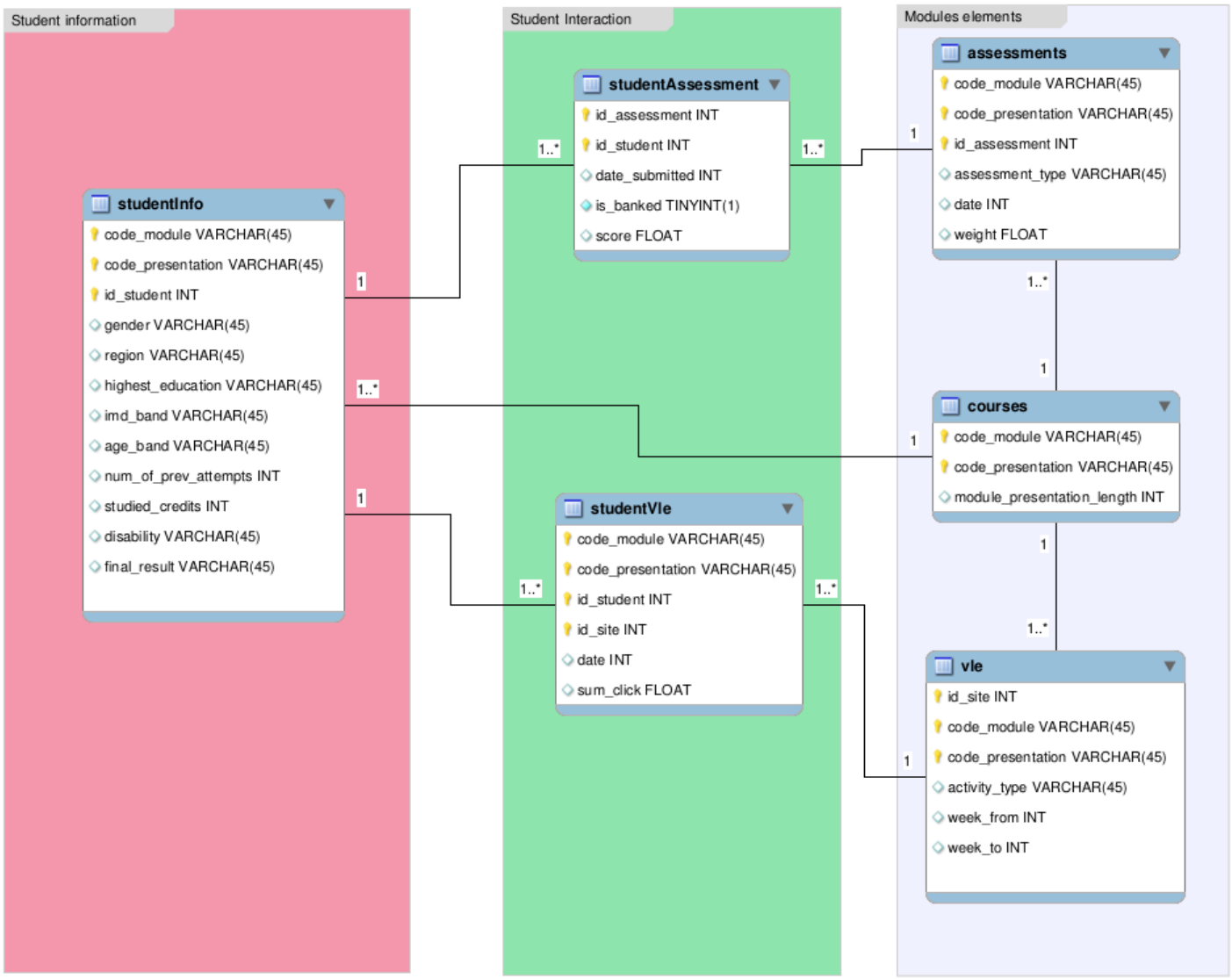

Figura 19:Diagrama UML de la base de datos del dataset

12.5 Pretratado del dataset

Antes de explicar cómo se ha procedido a la obtención de la información necesaria para la predicción a partir de los registros de Learning Locker hay que especificar algunas consideraciones relevantes sobre la información del dataset utilizado para entrenar el modelo de machine learning.

Primero, mencionar que los diferentes algoritmos de machine learning otorgan diferentes pesos a los valores de entrada dependiendo de su influencia en el resultado final, por eso se han omitido los assessments de tipo Final Exam, ya que la influencia de la nota del examen final tendría un peso demasiado elevado y tampoco es un factor que se considere en ninguna de nuestras hipótesis, unido al hecho de que la idea de esta solución consiste en otorgar un soporte continuado al alumno, el examen final suele tener lugar al final del curso.

El segundo aspecto a mencionar es sobre las fechas de entrega. A la hora de tratar este tipo de datos se observó que algunos valores excedían el intervalo válido (fecha de entrega mayor que fecha final del entregable), produciendo valores negativos en nuestros cálculos. Ante esta situación, se consultó con el responsable del dataset y nos informó que algunos de los assessments se habían entregado de forma excepcional vía correo electrónico una vez finalizada la fecha limite o, en otras situaciones, se podía solicitar una extensión de dos semanas. Dado que esto correspondía a casos aislados y la mayoría de los valores del dataset eran de entregas realizadas a través de la plataforma web dentro de las fechas indicadas, se consideró no utilizar estos casos para el entrenamiento del modelo.

Por último, se ha modificado el campo sum_clicks de la tabla studentVle. Este atributo indica el número de veces que el alumno interaccionó con un material en un mismo día. Como posteriormente veremos, para entrenar el modelo se realizará la media de la suma total de las interacciones, pero al tratarse de cursos ya finalizados esta media tendrá un valor elevado. Si se dejarán estos valores tal cual, el resultado de las predicciones daría valores incorrectos hasta que el curso no avanzará lo suficiente, ya que al principio habría una gran diferencia entre dichos valores. Por eso, se ha decidido normalizar este campo aplicando la denominada normalización N1. La idea es sencilla, consiste en coger el valor máximo y dividir todas las entradas por él.

Una vez comentadas las consideraciones a tener en cuenta, las sentencias sql que se utilizaron para obtener la información del dataset han sido,

```
SELECT distinct s.code_module, s.code_presentation, s.id_student,avg(a.date-
sa.date_submitted),s.final_result
FROM studentInfo s inner join assessments a ON a.code_module = s.code_module and
a.code_presentation = s.code_presentation and a.assessment_type <> 'Exam' inner
join studentAssessment sa on sa.id_assessment = a.id_assessment and sa.id_student
= s.id_student and sa.is_banked = '0' and a.date >= sa.date_submitted
WHERE s.code_module in ('AAA','CCC')
GROUP BY s.code_module, s.code_presentation, s.id_student, s.final_result;
```

Figura 20: Sentencia SQL para obtener los días de media en entregar una práctica

Con esta sentencia se obtiene la información del estudiante con los días de media que tarda en entregar una práctica. Como se puede observar, no tenemos en cuenta los casos donde la nota de la práctica es heredada de un cuatrimestre anterior (`sa.is_banked = '0'`), a la vez que también filtramos las fechas de entrega para evitar que sean negativas, tal como hemos comentado. `a.date >= sa.date_submitted`.

Una vez obtenido los días de media falta el número de interacciones, para ello se utilizó la siguiente sentencia,

```
SELECT distinct s.code_module, s.code_presentation, s.id_student,sum(sv.sum_click),s.final_result
FROM studentInfo s inner join vle v ON v.code_module = s.code_module and v.code_presentation = s.code_presentation inner join studentVle sv on sv.id_site = v.id_site and sv.id_student = s.id_student
WHERE s.code_module in ('AAA','CCC')
GROUP BY s.code_module, s.code_presentation, s.id_student,s.final_result;
```

Figura 21: Sentencia SQL para obtener el número de interacciones de los alumnos con el material

De este modo obtenemos la suma total de interacciones que ha realizado el alumno a lo largo de todo el curso, posteriormente, sobre este valor se aplicó la normalización N1 antes mencionada.

A partir de las dos consultas anteriores ya disponemos de toda la información necesaria para entrenar el modelo. El número de días de media que tarda el alumno en entregar las prácticas, el número de interacciones normalizado y, por último, el resultado del curso (aprobado o suspendido).

12.6 Calculo de rendimiento

Como he mencionado anteriormente, para calcular el rendimiento de un estudiante a partir del dataset facilitado por el Knowledge Media Institute, The Open Univeristy. United Kingdom, utilizo las hipótesis:

1. Entregar una práctica (o entregable) con un margen amplio respecto la fecha límite es un indicador de los alumnos que consiguen buenas cualificaciones finales.
2. Un nivel alto de actividad en el curso (de Moodle en este caso) antes de la entrega de las diferentes tares es un indicador de buen rendimiento.

Con la información de los registros de actividad almacenados en Learning Locker resulta sencillo obtener la hora a la que ha tenido lugar la actividad del usuario (objeto timestamp), permitiendo por lo tanto utilizar también la tercera hipótesis, *La actividad nocturna (o por la tarde) es un mejor indicador de rendimiento que la actividad diurna*, pero como se ha mencionado anteriormente, se necesita entrenar un modelo para poder hacer predicciones a partir de este, y el dataset con el que se ha trabajado no dispone de dicha información, imposibilitando la utilización de la tercera hipótesis en el estudio.

En el apartado anterior hemos explicado cómo se obtuvo la información necesaria para entrenar el modelo, a continuación, vamos a explicar cómo obtener la misma información, pero esta vez, extraída a partir de los registros almacenados en Learning Locker.

Para la obtención de la información necesaria se ha programado una tarea diaria (cron task) a las 00:00 en el servidor, que ejecuta un script en php. Este script utiliza la librería TinCanPHP para obtener todo el registro de eventos almacenados en Learning Locker del día anterior. Posteriormente, se procede al tratado de cada log y se actualiza la base de datos del servidor en consecuencia.

La información que hemos de obtener de cada registro es el nombre del alumno, el curso en el que ha realizado la actividad y detectar si se trata de un entregable o tan solo ha interaccionado con algún material, en el caso de ser un entregable se requiere también los días restantes hasta la fecha límite. El nombre del estudiante es fácil de obtener a través de métodos facilitados por la librería antes mencionada. Para detectar el curso en el que ha tenido lugar la acción hace falta recorrer el campo “context”. Como se ha explicado antes, este campo añade información adicional sobre el statement, en este caso, se encuentra especificado el curso. En el apéndice 1 se explica en mayor detalle este proceso.

Una vez obtenido el alumno y el curso, la información que falta por obtener es si se trata de una interacción que no tiene ningún entregable asociado, o, por el contrario, si se ha completado una práctica. Antes de explicar cómo se ha procedido para obtener esta información hace falta explicar la forma en que Moodle divide los diferentes tipos de materiales que dispone.

Encontramos una división principal entre recursos y actividades. Los recursos son materiales para el alumno en los que no se espera que este realice ninguna actividad asociada. Por el contrario, las actividades están formadas por eventos en que el estudiante tiene que realizar algún ejercicio, ejemplos de actividades son foros, encuestas, chats, entregables, etc. Para que los resultados del estudio se asemejaran lo máximo posible a los del dataset utilizado se ha considerado como assessment (nombre con el cual aparecen en el dataset) únicamente a las actividades identificadas como entregables.

Una vez explicitado que tipo de contenido se va a considerar como entregable, para identificar si el evento que se está tratando corresponde a una entrega o no se tienen en cuenta dos factores. Por un lado, se comprueba que el verbo del statement sea “completed”, verbo que Moodle asocia a la acción de añadir un fichero o responder a un entregable. Por otro lado, se verifica que en la definición de la activity (o objeto de la acción), en la descripción del tipo, aparezca “/moodle/assign”, identificador que reciben este tipo de actividades. Si se cumplen ambas condiciones se procede a calcular el número de días que ha tardado el alumno en completar la entrega. Para ello, utilizamos los valores de "duedate" y “time modified” que aparecen en el atributo “extension” del campo objeto, estos corresponden a la fecha límite y la fecha de entrega. Este proceso se encuentra detallado en el apéndice 2.

Por último, se decidió que cualquier tipo de interacción, tanto completar un entregable como leer un documento se consideraría como un "click", y, por lo tanto, se incrementa en una unidad el campo correspondiente en la base de datos para ambos casos.

12.7 Resultados de los algoritmos predictivos

Una vez se han generados los distintos scripts que implementan cada uno de los tres algoritmos predictivos, a la hora de entrenar los modelos se ha utilizado un subconjunto de los estudiantes del dataset. Para determinar la precisión de cada algoritmo se han realizado predicciones sobre el resto de estudiantes que no han sido utilizados para entrenar el modelo.

En este apartado vamos a analizar los resultados obtenidos por los diferentes algoritmos predictivos explicados en el apartado *13 Machine Learning*. La precisión obtenida por cada uno de ellos ha sido **78,03%** con k nearest neighbours, **72,72%** para el caso del random forest, y, por último, **80,22%** para las redes neuronales.

12.7.1 K-Nearest Neighbour

Este algoritmo es uno de los que mejor rendimiento ha ofrecido. Aparentemente, la idea detrás del mismo es la más sencilla de entender. Los parámetros que podemos configurar de este algoritmo son el número de vecinos cercanos a tener en cuenta y la función a utilizar para calcular la distancia. Para la búsqueda de los valores óptimos se han comprobado todas las posibilidades y escogido los valores con mejor resultado.

La distancia a la que un elemento se puede considerar vecino viene determinada por la formula Minkowski.

$$d(\mathbf{x}, \mathbf{y}) = \left(\sum_{i=0}^{n-1} |x_i - y_i|^p \right)^{1/p}$$

Figura 22: Formula de la distancia Minkowski[15]

Se han ido comprobando los diferentes valores de *p*, y el que mejor resultado ofrecido es para $p = 2$, que corresponde con la distancia Euclidiana. En la gráfica de la página siguiente se puede observar la variabilidad del rendimiento según el valor de *p*.

[15] Formula extraída de https://wikimedia.org/api/rest_v1/media/math/render/svg/33aa1151bd324808aeb7d7bd1262f6b8c515ec14

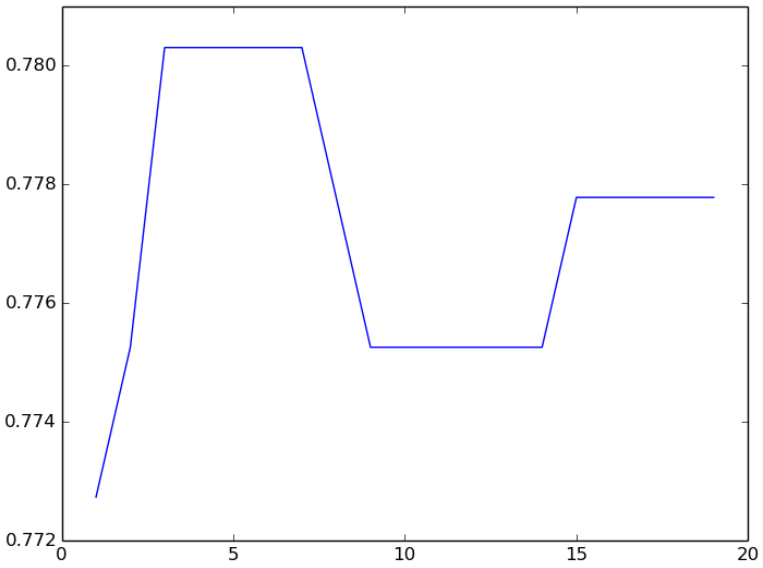

Figura 23:Gráfica que muestra la relación entre el rendimiento obtenido por K-nearest neighbours (abscisa) y el valor de p utilizado en la fórmula de Minkowski (ordenadas)

El otro parámetro configurable es el número de vecinos a tener en cuenta. Para ello, se ha utilizado el mismo mecanismo que en el caso anterior. El valor obtenido ha sido 22.

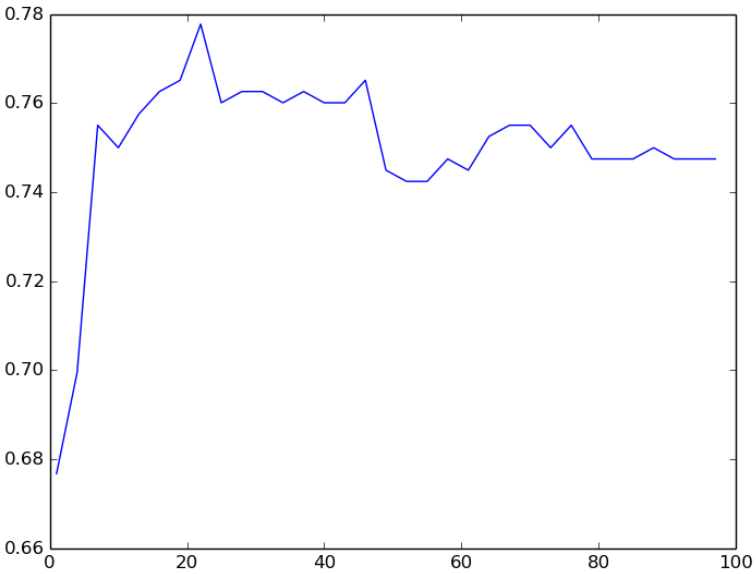

Figura 24: Gráfica que muestra la relación entre el rendimiento obtenido por K-nearest neighbours *(abscisa)y el número de vecinos a tener en cuenta por el algoritmo (ordenadas)*

Por lo tanto, en el caso de k nearest neighbours, para determinar si un alumno aprueba o suspende se tienen en cuenta las 22 instancias más parecidas utilizando la distancia Euclidiana.

12.7.2 Random forest

Por su parte, el random forest ha sido el algoritmo predictivo cuya precisión es la menor de los tres. En este caso, el único parámetro configurable por Scikit-learn es el número de árboles de decisión utilizado por el algoritmo. La metodología que se ha seguido es la misma que para el caso de k nearest neighbours. Se ha probado a ejecutar el algoritmo desde 1 hasta 1000 árboles, incrementando en 50 el valor para cada iteración. Los resultados obtenidos se muestran en la siguiente gráfica, siendo el eje de abscisas el número de árboles de decisión utilizado y las ordenadas la precisión obtenida por el algoritmo.

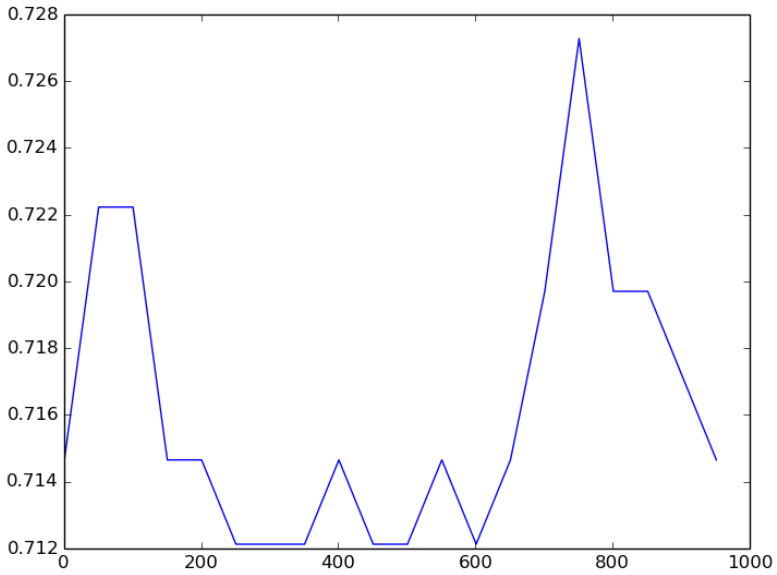

Figura 25: Gráfica que muestra la relación entre el rendimiento obtenido por random forest *(abscisa)y el número de árboles de decisión utilizados (ordenadas)*

12.7.3 Redes neuronales

Por último, las redes neuronales han sido el algoritmo con el que se ha obtenido mejor resultado. La cantidad de parámetros de configuración disponibles para este caso es muy elevada, ya que hay una gran variedad de factores a tener en cuenta. En nuestro caso particular, tan solo hemos tenido en cuenta el valor del *learning rate* (ajustado a 0.001, valor recomendado en muchos estudios), el número de iteraciones a ejecutar durante la fase de entrenamiento (1.000), el número de hidden layers así como de neuronas en cada una de ellas. Por defecto, la función de activación utilizada es la relu.

Para determinar que valores se ajustan mejor a nuestros datos, al igual que en los casos anteriores, hemos generado diferentes modelos modificando el número de hidden layers y de neuronas en cada una de ellas. En la siguiente gráfica se puede observar la variación de rendimiento acorde con el número de hidden layers.

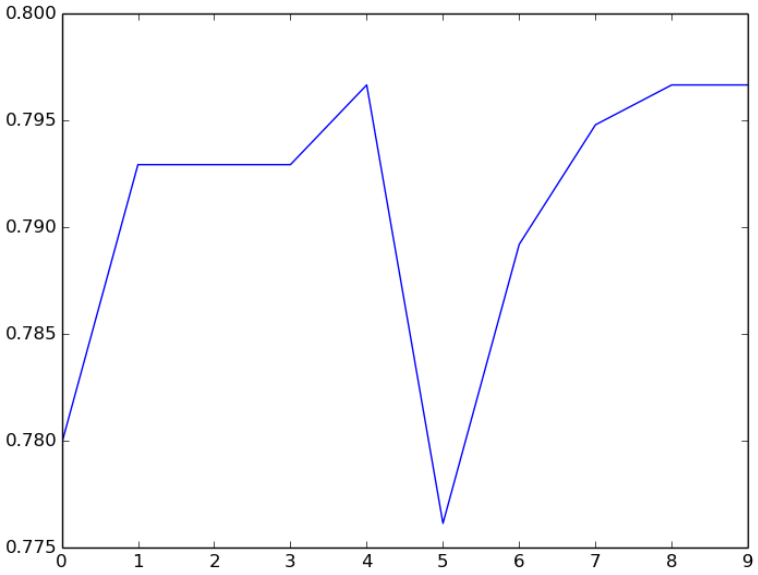

Figura 26: Gráfica que muestra la relación entre el rendimiento obtenido por la red neuronal *(abscisa)y el número de hidden layers utilizadas (ordenadas)*

En este caso, el número de hidden layers que mejor resultado da es cuatro u ocho.

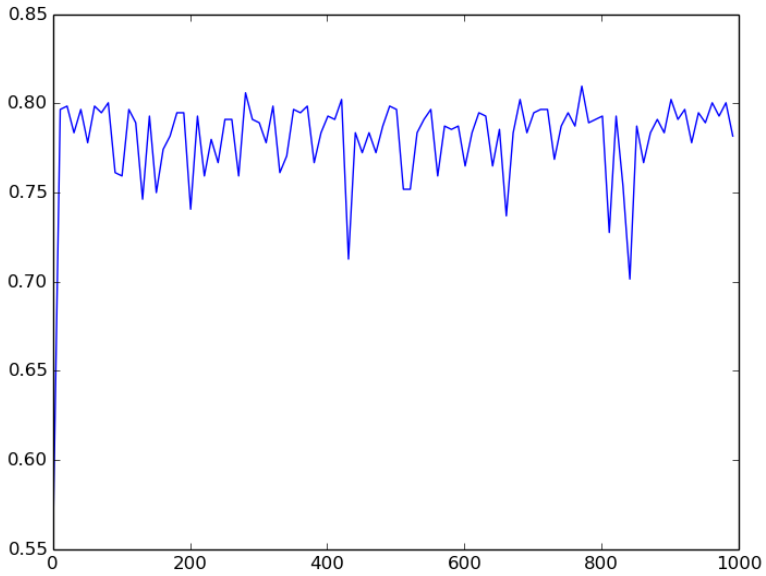

Figura 27: Gráfica que muestra la relación entre el rendimiento obtenido por la red neuronal *(abscisa) y el número de neuronas por cada capa (ordenadas)*

Y para las neuronas por cada capa, 771.

12.8 Análisis de los resultados

Después de las pruebas con los diferentes algoritmos se ha conseguido una precisión del 80,22% a través de las redes neuronales, dicho de otra manera, de cada 10 posibles predicciones, 2 son erróneas. Es un valor que a priori puede parecer aceptable, pero de cara a su utilización en casos reales considero que debería mejorarse. A continuación se detallan algunas de las causas que están limitando (o influenciando) estos resultados.

La primera de las causas es debido a la naturaleza de los datos. Aun y haber demostrado la validez de las dos hipótesis utilizadas en el estudio *"Indicators of Good Student Performance in Moodle Activity Data*, el dataset que hemos usado para entrenar los modelos no acaba de reflejar el mismo comportamiento.

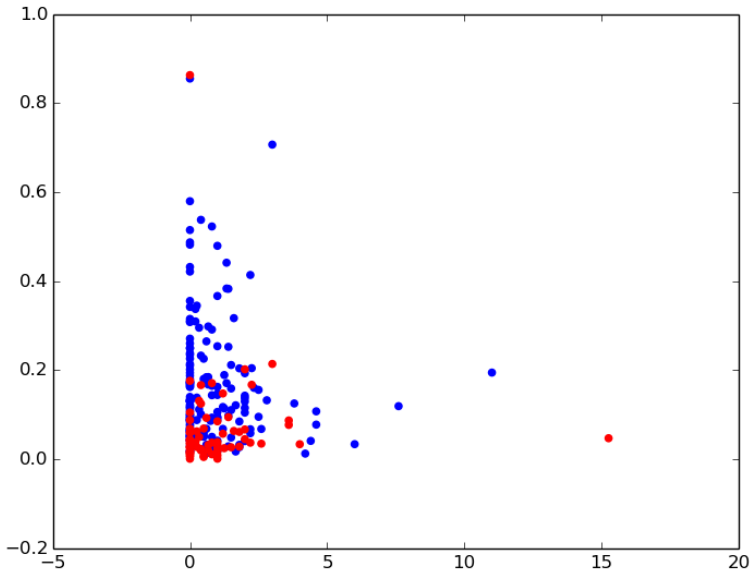

Figura 28: Gráfica que muestra la correlación entre los alumnos que aprueban y suspenden.Teniendo en cuenta el número de interacciones (abscisa) y los días de margencon los que se realiza una entrega (ordenada)

En la gráfica anterior se muestra la correlación entre los datos. Como vemos, sí que sigue ligeramente el comportamiento descrito por las hipótesis anteriores, la mayoría de puntos rojos (alumnos que han suspendido) están acumulados en la esquina inferior izquierda, donde el número de interacciones y margen entre entregas es cercano a 0. Aun así, todo y poder intuir dicho comportamiento, observamos que la distancia entre alumnos que suspenden y aprueban es muy cercana, en muchos casos se sobreponen, aumentando la complejidad a la hora de realizar predicciones.

Otro factor importante a tener en cuenta es que los algoritmos predictivos tan solo están recibiendo dos atributos de entrada por cada instancia (número de interacciones y margen entre entregas). En la mayoría de casos de uso reales donde se utiliza machine learning, la cantidad de atributos a tener en cuenta por el algoritmo es mucho mayor.

Así pues, a modo de resumen de este bloque del proyecto, se ha obtenido un porcentaje de acierto del 80,22%, valor que verifica la validez de las hipótesis utilizadas, de no ser así, el valor de la precisión obtenida distaría mucho.

Uno de los factores más influyentes, todo y haber sido solucionado, ha sido la dificultad por obtener unos datos reales con los que entrenar los modelos. Ya que el comportamiento de los estudiantes es diferente según el país, según la carrera, e incluso, dentro de cada carrera, según el centro donde se imparta. Por ello, para mejorar el resultado obtenido en este estudio, cada centro debería recopilar la información académica de los alumnos (con su consentimiento), realizar un análisis sobre la misma y extraer hipótesis a través de sus patrones de comportamiento, y, por último, elaborar un sistema predictivo acorde a estas hipótesis.

13. Plataforma Web

A lo largo de estos apartados se van a presentar los distintos elementos que forman parte de la plataforma web, tanto funcionales como de diseño. La mayor parte de las universidades (por no decir la totalidad) disponen de un espacio virtual propio donde el alumno puede consultar información sobre sus asignaturas matriculadas, próximos eventos, gestión de expediente, bolsa de trabajo, etc. Por eso, se ha decidido que carece de sentido elaborar un espacio virtual completo e independiente para mostrar los datos de rendimiento, ya que muchas de las funcionalidades susceptibles de ser mostrada en dicha plataforma ya están presentes en los espacios virtuales actuales y consideramos que es contraproducente para el estudiante ofrecerle un espacio adicional, independiente de los ya disponibles, en vez de centralizar todo en un único recurso. Por ello, se ha decidido plantear el diseño de esta plataforma como un prototipo del tipo de información adicional que pueden añadir las universidades a sus plataformas ya existentes, a la vez que nos sirve para enseñar de manera visual los resultados obtenidos por este proyecto.

En los siguientes dos subapartados vamos a comentar los diferentes requisitos, funcionales y no funcionales, que se espera de nuestra plataforma a la hora de interactuar con el usuario.

13.1 Requerimientos funcionales

Los requisitos funcionales hacen referencia a las funcionalidades que ofrecerá nuestro sistema. El orden en que se van a enumerar intenta seguir el flujo de actuación de un usuario cualquiera. Muchos de los requisitos enumerados se espera que estén presentes en los espacios virtuales de las universidades.

A. El usuario podrá conectarse a su espacio virtual a través del portal web

Cuando un usuario se conecte a la página web el servidor se ha de encargar de responder con la página HTML y elementos necesarios para mostrar la página inicial al usuario. Dicha página inicial mostrará información básica de la plataforma y ofrecerá la posibilidad de hacer login.

B. El usuario podrá enviar la información necesaria para autenticarse

Una vez el usuario ha seleccionado la opción de autenticarse, se mostrará una nueva interficie con un formulario donde el usuario será capaz de introducir el nombre de usuario y contraseña para posteriormente enviar los datos al servidor.

C. El sistema redirigirá al usuario a su espacio personal

Si los datos introducidos por el usuario en el anterior punto son correctos, el servidor se encargará de buscar toda la información relativa a este para posteriormente redirigirle a una nueva página donde aparecerá toda la información correspondiente. El orden de aparición de la información es, alertas (en el caso que exista alguna), estado actual de las asignaturas (aprobada o suspendida), rendimiento del estudiante respecto al total de los matriculados, tablón de mensajes y lista de tareas.

D. El usuario será capaz de consultar las alertas

La interficie ofrecerá al usuario la posibilidad de consultar sus alertas, clicando sobre ellas se le redirigirá a su tablón de mensajes.

E. Editar la lista de tareas

La interficie permite al usuario que organice las tareas relativas al listado de tareas. Las funciones principales que podrá llevar a cabo son, añadir una nueva tarea, marcar una tarea existente como completada y borrar una tarea.

13.2 Requerimientos no funcionales

A continuación, se listan los requerimientos no funcionales que hacen referencia a las características de funcionamiento del sistema, también denominados atributos de calidad.

A. Disponibilidad

El servicio ha de estar disponible para ser utilizado por los usuarios en todo momento.

B. Tiempo de respuesta

Cuando el usuario interacciona con el sistema, introduciendo datos en el formulario o editando su lista de tareas, el sistema ha de responder en un tiempo inferior a 5 y 2 segundos correspondientemente.

C. Usabilidad

El diseño de la interficie de cada página ha de seguir unos criterios de usabilidad básicos para facilitar la interacción del usuario con la plataforma, independientemente del tipo de usuario.

D. Seguridad

La información estará protegida contra accesos no autorizados utilizando mecanismos de validación que puedan garantizar que cada usuario tiene acceso solamente a la información relacionada con sus asignaturas.

Por cuestiones de seguridad los datos no deben viajar al servidor en texto plano, por ello se usarán mecanismos de encriptación, del mismo modo que dicha información se encontrará almacenada en las bases de datos de forma cifrada.

E. Confiabilidad

El sistema ha de ser robusto a fallos, tanto técnicos como derivados de la interacción con el usuario, como por ejemplo la introducción de una contraseña errónea.

F. Mantenimiento

El sistema ha de ser revisado periódicamente para detectar lo antes posible la existencia de un error, así como la realización de diferentes tareas de mejora del sistema.

G. Escalabilidad

El sistema ha de ser fácilmente escalable para hacer frente a un aumento del tráfico sin que el servicio resulte perjudicado.

Como se ha mencionado, esta solución está pensada para que la adopten las instituciones docentes en sus servidores, por lo que muchos de los puntos descritos anteriormente, tales como la seguridad, mantenimiento y escalabilidad, se entienden como competencias suyas y se da por supuesto su cumplimiento.

13.3 Casos de uso

En el siguiente diagrama de casos de uso (figura 29), se ilustran los requerimientos funcionales que se han explicado anteriormente, suponiendo que el usuario no ha iniciado sesión en la plataforma.

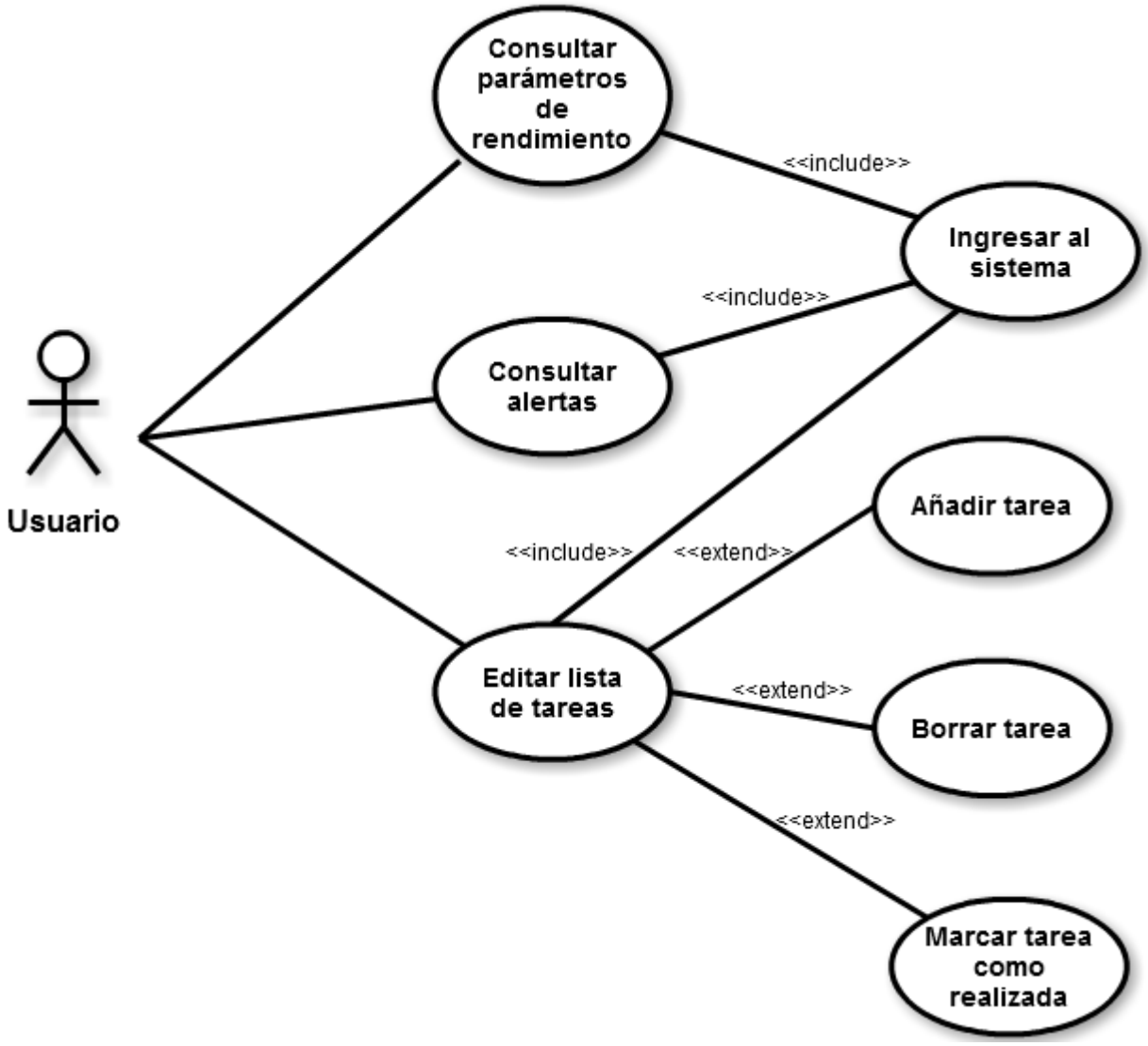

Figura 29: Casos de uso plataforma web

13.4 Diseño de la arquitectura

El patrón de diseño que se ha utilizado en la elaboración del portal web es el conocido como modelo vista controlador (MVC). Es una filosofía ampliamente conocida en el diseño de software, cuyos principios también se pueden extrapolar a la elaboración de un servicio web. Este patrón divide el sistema en tres niveles de abstracción o capas. E**l modelo**, encargado de almacenar los datos e interaccionar con ellos, **la vista,** encargada de mostrar la información al usuario (en este caso las páginas web) donde estos podrán interaccionar, y, por último, **el controlador** es la capa intermedia entre vista y modelo, recibe las peticiones de la vista, las traduce al modelo y retorna la respuesta correspondiente de nuevo a la vista. La vista se implementa en el *front-end*, mientras que el modelo y controlador forman parte del *back-end.*

Como se ha visto en el apartado anterior, interacciones (o peticiones) por parte del usuario que requieran de la aplicación de este principio encontramos el inicio de sesión y la edición de las tareas presentes en la lista de *todo's*. El resto de procesos son transparentes al usuario. La estructura que se ha seguido para respetar el patrón mvc se representa en las siguientes imágenes.

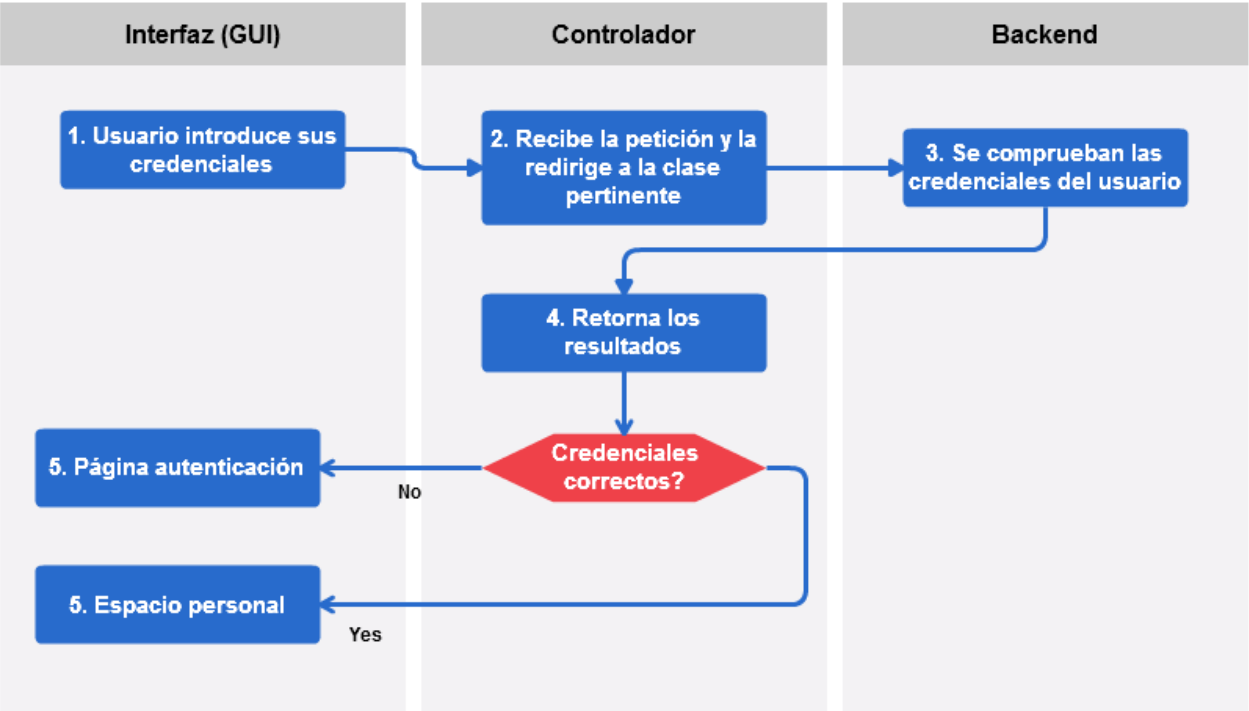

Figura 30: Diagrama de flujo inicio de sesión

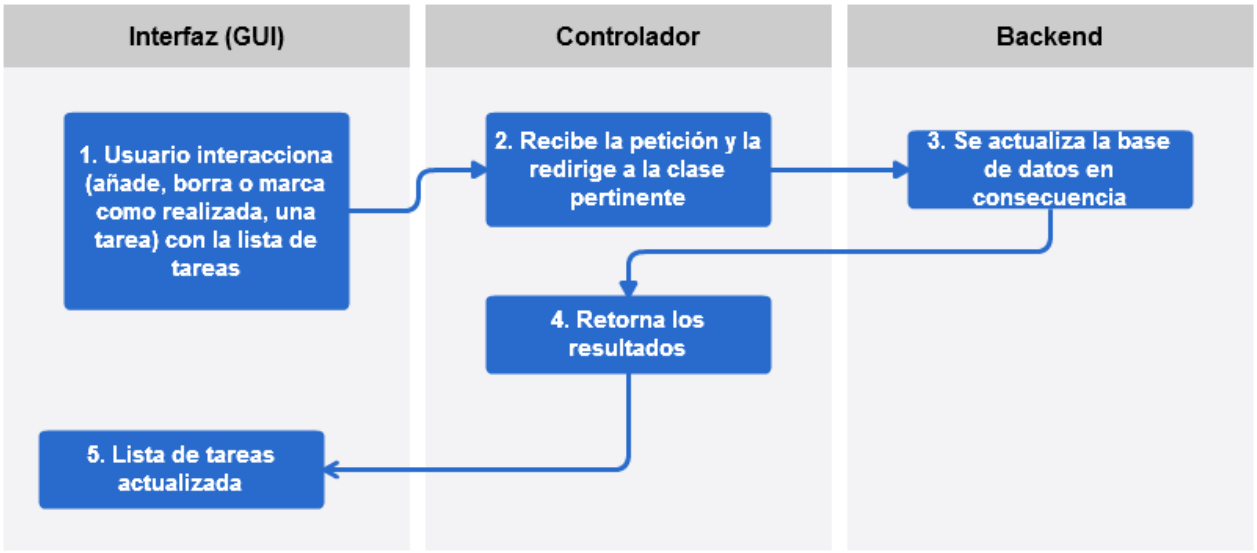

Figura 31: Diagrama de flujo de la interacción con la lista de tareas

Como se puede observar, la vista (referida como interfaz en las imágenes) está formada por una serie de ficheros como son la página principal, la de login, y, por último, el espacio personal del usuario. Cuando el usuario realiza una petición, esta es recibida por la clase controlador, que es la encargada de determinar qué acciones son necesarias así como de la comunicación con la base de datos. Una vez el controlador recibe la respuesta, se encarga de transferirla de nuevo a la vista.

13.5 Diseño de la interfaz

En este apartado se va a analizar el diseño de la plataforma web, comentando las diferentes páginas que forman parte de él y las opciones disponibles en cada una de ellas. Se va a seguir el orden lógico descrito en los casos de uso, desde que el usuario inicia sesión hasta que interacciona con su espacio personal.

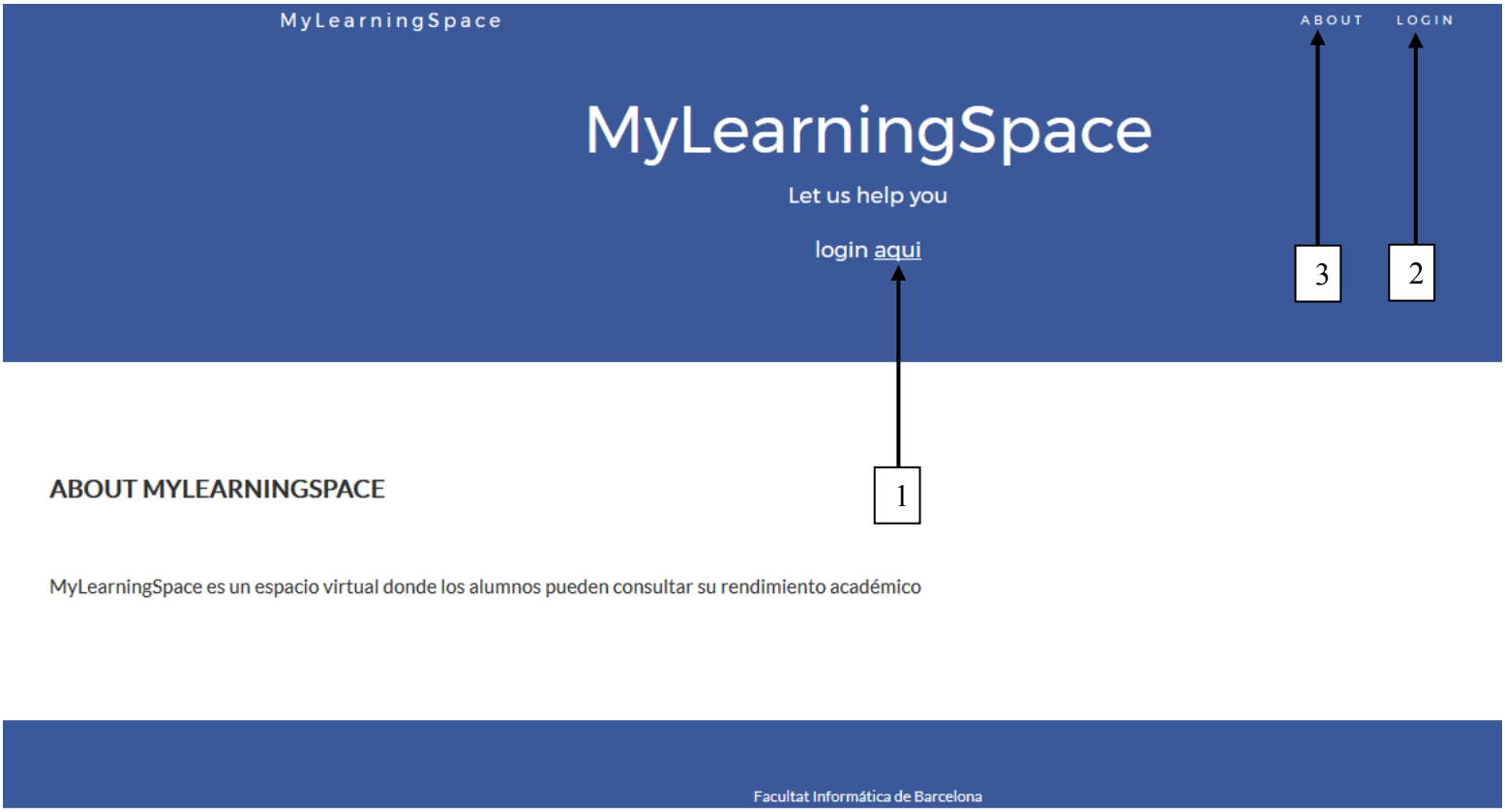

Figura 32: Diseño de la página principal del portal web

La imagen anterior corresponde a la página principal de la plataforma web. En ella podemos observar una breve descripción del objetivo de la plataforma, así como la posibilidad de iniciar sesión. Para ello, el usuario dispone de dos opciones, marcada con 1 y 2.

La opción about, 3, redirige a la sección correspondiente de la página. En el caso de acceder a la plataforma a través de un ordenador portátil o de sobremesa, el usuario ya es capaz de visualizar la totalidad del contenido sin necesidad de seleccionar la opción, como ocurre en la imagen. Esta opción está pensada para el uso a través de dispositivos móviles, donde el tamaño de la pantalla no permite una visualización completa.

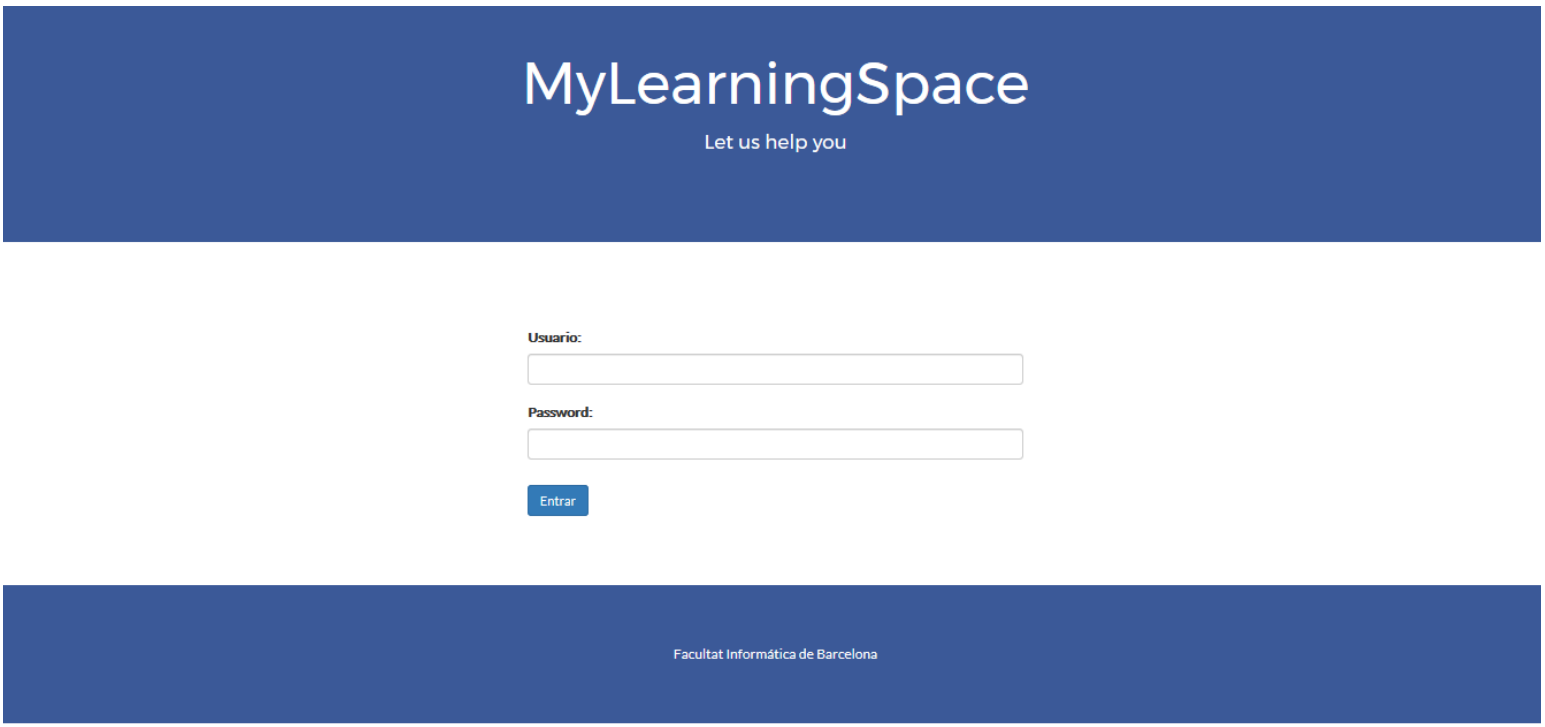

Figura 33: Diseño de la página de login de la plataforma web

Una vez seleccionada la opción de iniciar sesión, el sistema nos redirige a la página de la Imagen 33. Dispone de un diseño sencillo, donde podemos indicar el nombre de usuario y la correspondiente contraseña. En caso de ser incorrecta, se muestra un aviso al usuario como el de la siguiente Imagen.

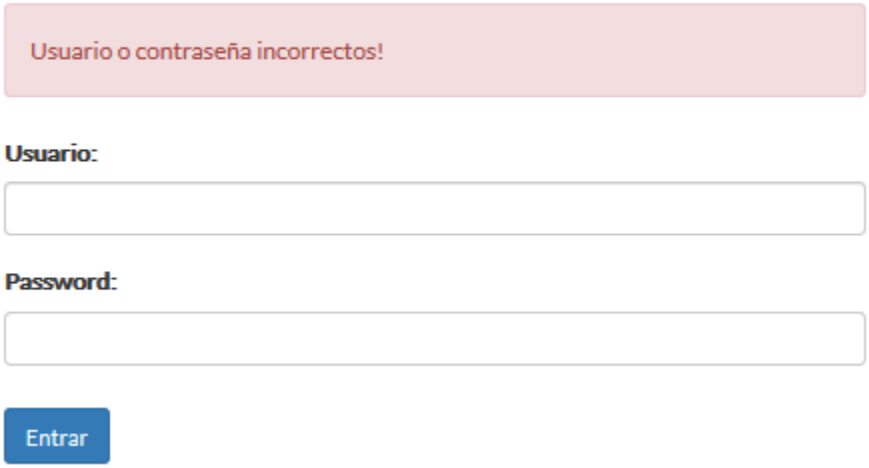

Figura 34: Mensaje de error en caso de login incorrecto

Antes de comentar el diseño del espacio personal del estudiante, una vez ya haya iniciado sesión correctamente, vamos a explicar las diferentes secciones que forman parte de él. La página tiene tres secciones principales:

- **Overview**: Es el primer apartado, aparece cada asignatura en la que está matriculado el alumno junto a su estado actual, aprobada o suspendida.

- **Performance**: Esta sección muestra de forma visual el rendimiento del alumno respecto a todos los estudiantes de la asignatura. El objetivo de esta sección también consiste en motivar al estudiante utilizando la competitividad por querer estar entre las mejores posiciones de la clase, puede que no suponga una gran motivación extra, pero en distintas asignaturas se ha demostrado la validez de este tipo de técnicas, como el juego de EDA por ejemplo.

- **Management**: Por último, en esta sección hay un tablón de mensajes (tan solo editable por la plataforma) donde se muestra información adicional para el alumno. También hay una lista de tareas para que el alumno pueda crear anotaciones.

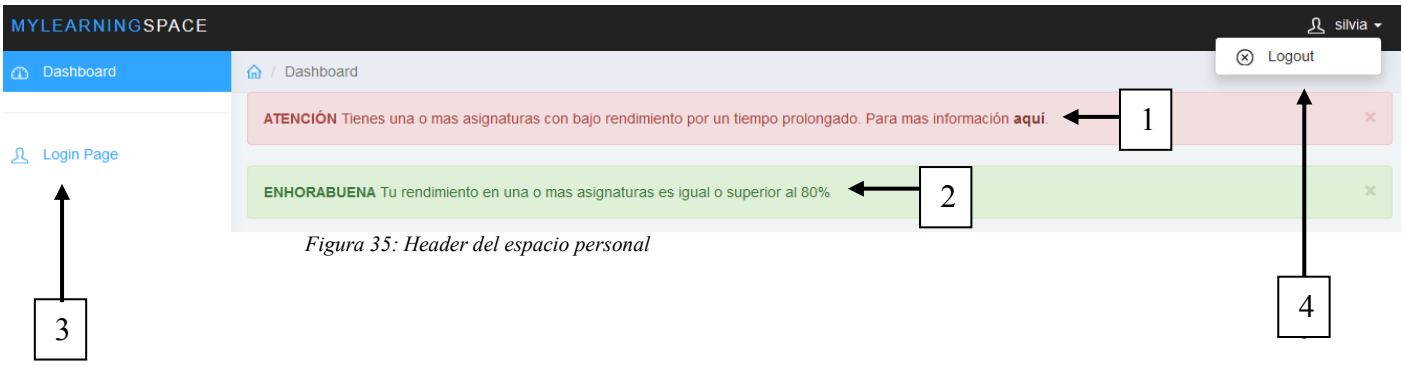

Figura 35: Header del espacio personal

La imagen anterior corresponde a la cabecera del espacio personal del estudiante. Como se puede observar, se muestran notificaciones que pueden resultar del interés del estudiante, tanto por su importancia, en el caso de bajo rendimiento 1, o, meramente informativa, si está manteniendo un buen nivel en alguna asignatura 2. De nuevo, la interfaz dispone de dos mecanismos para cerrar sesión, a través de 3 o 4, para que sea cómodo tanto a través del ordenador como del dispositivo móvil, donde no aparece la opción 3.

Figura 36: Sección de overview del espacio personal del estudiante

Tal y como se ha comentado anteriormente, en la imagen 36 podemos observar la primera sección del espacio personal del estudiante, el overview. De este modo, el alumno puede hacerse rápidamente una idea de cuál es el estado de las asignaturas. Como se muestra en la imagen, hay tres colores posibles para indicar el estado de una asignatura, estos son verde si aprobará, amarillo si está marcada como suspendida pero lleva marcada como tal durante un período inferior a 15 días, y rojo en caso contrario.

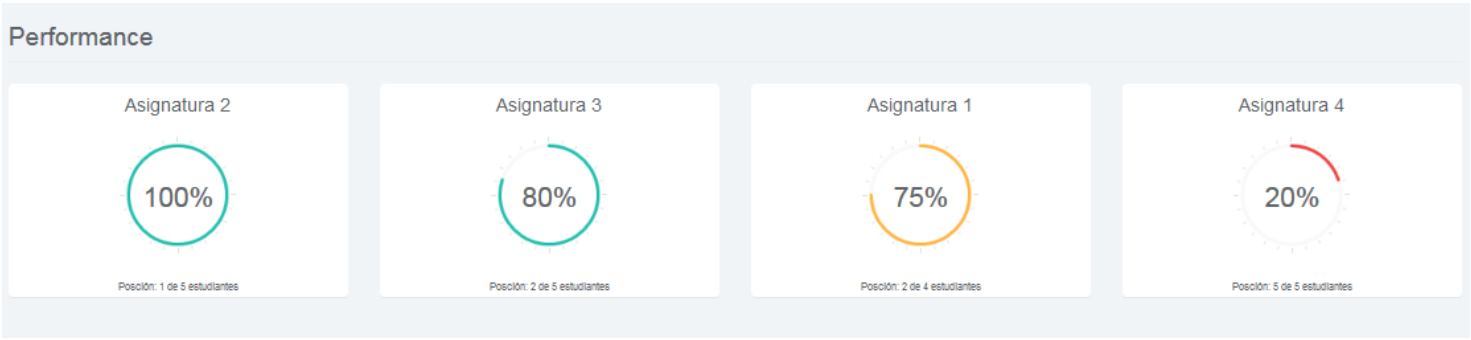

Figura 37: Sección de perfomance del espacio personal del estudiante

La imagen superior corresponde a la sección de performance. Como se ha comentado en la introducción del apartado, el objetivo de esta sección consiste en mostrar el rendimiento del usuario. Dicho rendimiento se calcula a partir de los parámetros de todos los estudiantes de la asignatura, siendo el estudiante con mejor promedio el que ocupará la primera posición.

Para mostrar la información utilizamos dos recursos, a través de gráficas circulares con distintos colores dependiendo de su rendimiento, y, por debajo de estas se indica la posición exacta que ocupa el estudiante entre todos los matriculados.

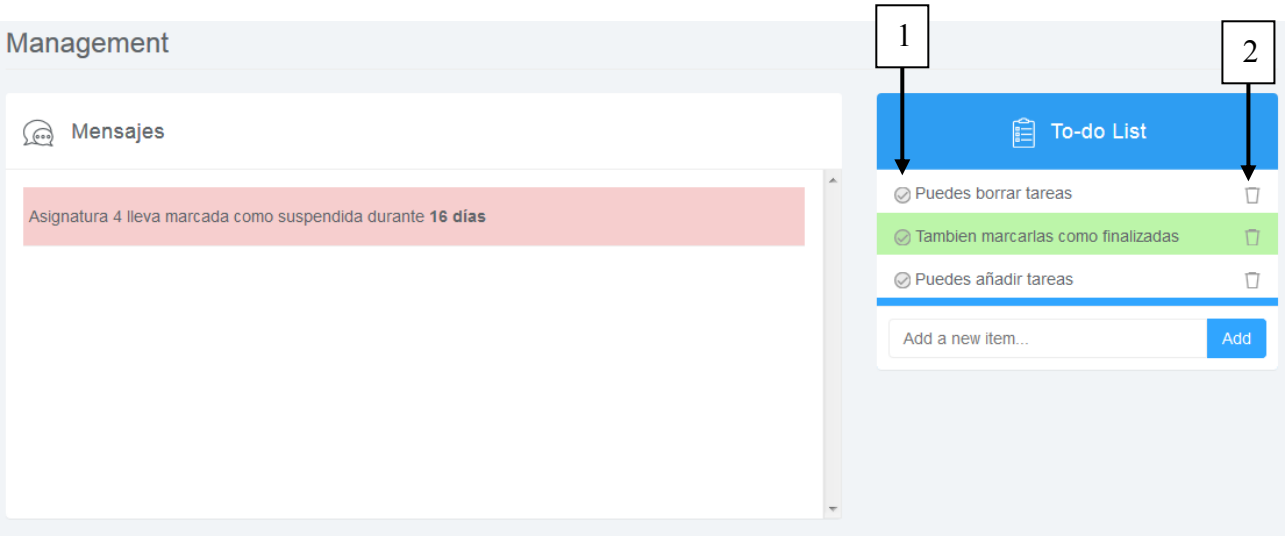

Figura 38: Sección de management del espacio personal del estudiante

Por último, encontramos la sección de *management*. La finalidad de este apartado, como su nombre indica, está orientada a aspectos más administrativos. En la imagen 38 se puede observar el diseño de la sección de *management*. En la parte izquierda se encuentra el tablón de mensajes, donde el usuario puede consultar sus notificaciones. En el caso del ejemplo, se informa al estudiante que la *asignatura 4* lleva 16 días consecutivos marcada como suspendida.

En la parte derecha encontramos una lista de tareas, de este modo el usuario puede añadir tareas o recordatorios que le resulten de interés y le faciliten la organización. Una vez se ha creado una tarea, el usuario puede marcarla como realizada, mediante el botón 1, la cual pasará a estar subrayada en verde o eliminarla mediante el botón de la papelera 2.

13.6 Implementación del portal web

Una vez explicado el diseño, en este apartado vamos a profundizar en cómo se ha llevado a cabo la implementación, tanto las tecnologías como las herramientas utilizadas. Vamos a empezar por los lenguajes de programación utilizados y posteriormente las definiciones de las tablas de la base de datos.

13.6.1 Lenguajes de programación

Para explicar este apartado vamos a dividirlo entre el front-end y back-end. En cuanto al front-end, como lenguajes utilizados, se ha optado por Bootstrap, CSS y JavaScript. Hoy en día, una gran parte de los accesos a las diferentes páginas webs se hace a través de dispositivos móviles, acentuándose sobre todo en la gente joven, por ello, una aplicación como la presentada en este proyecto consideramos que es esencial que sea *mobile responsive*.

En el back-end se ha decidido utilizar PHP y MySQL como gestor de las bases de datos. Ambas tecnologías han sido escogidas por el gran soporte y estabilidad que ofrecen, unido al hecho que ya se contaba con experiencia previa combinando ambas tecnologías, donde existe una fuerte sinergia.

13.6.2 Base de datos

Como hemos mencionado en el apartado anterior, el gestor de bases de datos escogido ha sido MySQL. En este apartado vamos a presentar las distintas tablas que forman parte de la plataforma web.

- user

Esta tabla tan solo sirve para identificar a los usuarios con una cuenta activa en la plataforma web, así como su contraseña, la cual se almacena cifrada mediante la función *password_hash* de php.

Nombre	Tipo	Definición
username	VARCHAR	Nombre de usuario
pwd	CHAR	Contraseña cifrada

Tabla 16: Definición de la tabla user del servidor web

- CourseStudent

Esta tabla es donde se almacena toda la información referente al rendimiento del estudiante. El contenido de esta tabla se actualiza diariamente a través del script en php cuya ejecución ha sido programada mediante una cron task (como se explicó en el apartado *14.1 Topología del sistema*). Parte de la información contenida en esta tabla se muestra en el portal web. De cara a facilitar la comprensión de los elementos de la tabla, dada una práctica, vamos a definir los días de margen como los días resultantes de restar la fecha límite de entrega – fecha en la que se ha entregado la práctica.

Nombre	Tipo	Definición
course	VARCHAR	Nombre del curso
student	VARCHAR	Nombre del usuario
avg_days	FLOAT	Días de margen de media
sum_days	INT	Días totales de margen, teniendo en cuenta todas las prácticas
num_assessments	INT	El número de prácticas entregadas hasta el momento
sum_click	FLOAT	El número de interacciones con los elementos del curso (prácticas, pdf's, foro, etc.) normalizado dividiendo total_clicks/max(total_clicks)

total_clicks	INT	El número total de interacciones con los elementos del curso (prácticas, pdf's, foro, etc.).
performance	FLOAT	Valor de rendimiento calculado a partir de (avg_days + sum_click)/ 2
pass	INT	Indica si el alumno aprueba o suspende la asignatura (1 y 0 respectivamente). Resultado obtenido a partir del algoritmo predictivo.
consecutive_days	INT	Días consecutivos que lleva marcada la asignatura como suspendida.

Tabla 17: Definición de la tabla CourseStudent del servidor web

El contenido de avg_days, sum_days, num_assessments y sum_click se actualiza a través del procesado de las sentencias de Learning Locker. Por otra parte, el valor de pass es el resultado del algoritmo predictivo. Al igual que se hizo con el dataset, el campo sum_click está normalizado. Para más información sobre la obtención de los datos contenidos en esta tabla consultar el apartado *14.6 Calculo de rendimiento.*

- TodoList

Esta tabla se utiliza para almacenar todas las tareas que cada usuario declara en su lista de *Todo's* presente en el espacio personal.

Nombre	Tipo	Definición
id	INT	Identificador de la tarea. Este campo se autoincrementa para cada tarea
username	VARCHAR	Nombre del usuario
description	CHAR	Descripción de la tarea
done	INT	Si la tarea ha sido marcada como completada o no (1 y 0 respectivamente)

Tabla 18: Definición de la tabla TodoList del servidor web

Las tres tablas anteriores se pueden ver representadas en el siguiente diagrama UML.

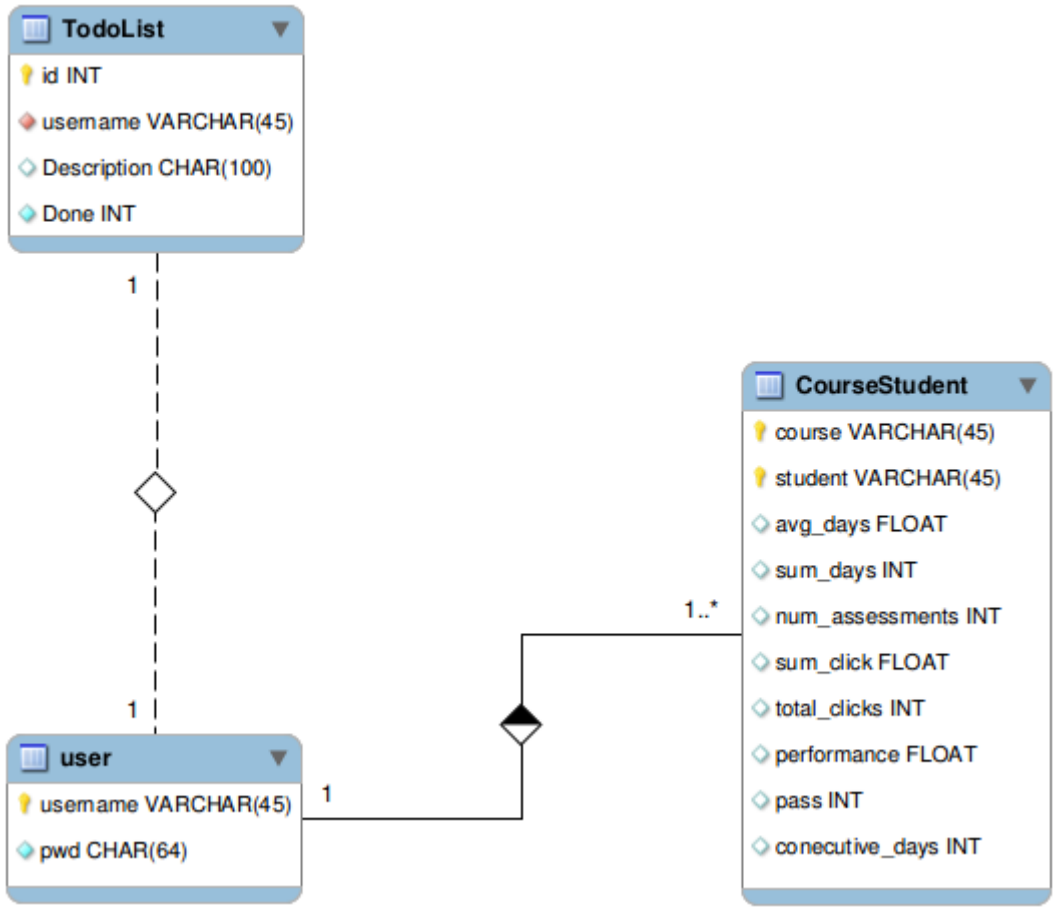

Figura 39: Diagrama UML base de datos portal web

13.7 Posibles funcionalidades adicionales

A lo largo de los distintos apartados de la sección 15 se han ido presentando los elementos que han formado parte en el diseño y desarrollo de la plataforma web. En este último apartado vamos a comentar posibles funcionalidades adicionales que por diversas causas no han podido ser añadidas a la solución actual, tanto por razones temporales como por limitación de recursos.

Como hemos comentado al inicio de la sección, esta plataforma web no pretende ser un espacio independiente de los que ya tienen a su disposición los estudiantes, sino un añadido a las soluciones que les ofrecen las universidades (por ejemplo, *el racó*). Algunas de las funcionalidades que vamos a listar a continuación podrían llevarse a cabo una vez las diversas universidades adoptaran la solución propuesta por este proyecto, de hecho, muchas de las mejoras están pensadas teniendo en cuenta las funcionalidades del *racó* (espacio virtual del estudiante ofrecido por la facultad de informática de Barcelona).

- **Mejorar el sistema de alertas.** Nuestro prototipo tan solo alerta al estudiante en caso de bajo rendimiento en una asignatura por un tiempo prolongado, también notificándole en caso de buen rendimiento. Muchos espacios virtuales ya ofrecen un calendario donde el usuario puede añadir tareas, como funcionalidad adicional se podría combinar el calendario con un sistema de alertas que informe al usuario cuando una práctica (o evento que el usuario haya marcado de interés) esté próxima a la fecha límite, o, por el contrario, cuando esté próxima a comenzar, de forma que el usuario tenga tiempo para organizarse.

- **Tendencias.** Una funcionalidad típica a la par que útil cuando se habla de rendimiento (tanto laboral como educativo) es poder visualizar la tendencia que se está siguiendo. Poder ver si el rendimiento ha incrementado respecto fechas anteriores, ver la evolución del rendimiento global de los matriculados, etc.

- **Aplicar mecanismos basados en la gamificación.** La Gamificación es una técnica de aprendizaje que traslada la mecánica de los juegos al ámbito educativo-profesional con el fin de conseguir mejores resultados, ya sea para facilitar la adquisición de conocimientos, mejorar alguna habilidad, o bien recompensar acciones concretas, entre otros muchos objetivos. Hoy en día, se está aplicando esta metodología a cada vez más sectores, algunas de las técnicas más utilizadas y susceptibles de ser incluidas en el espacio virtual son,

    - Acumulación de punto asignando valores cuantitativos a la realización de determinadas actividades.
    - Obtención de premios a medida que el usuario va completando objetivos, dichos premios son simbólicos.
    - Clasificar a los usuarios (pudiendo utilizar un nombre ficticio), en función de los puntos u objetivos.
    - Desafíos o retos que los usuarios puedan realizar y les otorguen puntos adicionales.

14. Leyes y regulaciones

La solución planteada por este proyecto utiliza y almacena información sobre el rendimiento académico del estudiante. Como ya se ha mencionado a lo largo de este informe, esta información está pensada para poder ser accedida tan solo por el propio estudiante, ya que es un mecanismo para darle soporte durante el proceso de aprendizaje, ningún profesor ni otro alumno tendrá acceso a ella.

A lo largo del proceso de desarrollo del proyecto se ha utilizado un dataset público y anónimo, por lo cual no se ha vulnerado ninguna ley vigente. En el futuro, si algún centro desea adoptar esta solución deberá asegurarse de cumplir los requisitos que aplican sobre estos datos.

Los requisitos a los que hacemos referencia son la Ley Orgánica de protección de datos 15/1999, que como es bien conocido, contiene las reglas por las que se rige la difusión de datos. Según el art. 1 de la norma "*tiene por objeto garantizar y proteger, en lo que concierne al tratamiento de los datos personales, las libertades públicas y los derechos fundamentales de las personas físicas, y especialmente de su honor e intimidad personal y familiar*".

Esta ley también dispone de apartados específicos para el trato de la información del estudiante por parte de las instituciones universitarias. La Disposición adicional vigésimo primera de la Ley Orgánica 4/2007, que se refiere a la protección de datos de carácter personal, estipula que "*Las universidades deberán adoptar las medidas de índole técnica y organizativa necesarias que garanticen la seguridad de los datos de carácter personal y eviten su alteración, tratamiento o acceso no autorizados*".

15. Sostenibilidad y compromiso social

15.1 Impacto económico

Como se comentó a lo largo del apartado *8.Identificación y estimación de los costes,* la mayor parte del presupuesto del proyecto está destinada a los recursos humanos. El resto de los gastos, materiales e indirectos, son constantes en cualquier proyecto de software, por lo tanto, el proyecto es totalmente viable económicamente, pudiendo reducir el tiempo de elaboración proporcionalmente al número de empleados que se disponga en plantilla.

Los modelos predictivos son un campo de estudio muy reciente y todavía por explotar, por lo que en caso de implantación de esta solución en las universidades, se requerirán diversas actualizaciones a lo largo de su vida útil. El coste de este mantenimiento es muy reducido y puede incluso llevarlo a cabo los diferentes docentes adjuntos a la universidad. En el caso de la Facultat de informática de Barcelona de la UPC, el departamento de Computer Science dispone de estudiantes y doctores investigando sobre esta tecnología.

No se prevé un ahorro económico para las instituciones que adopten esta solución, pero a su vez tampoco se derivan gastos adicionales. La mayoría de las universidades (por no decir la totalidad) disponen de servidores donde están alojados el correo, página web, portal, etc. Por lo tanto, de la forma en que se ha planteado este proyecto, podría ser una extensión de uno de estos servicios (Racó, en el caso de la FIB) y no sería necesaria la adquisición de ningún servidor adicional.

15.2 Impacto social

En la sección *1.Formulación del problema,* se ha explicado de forma extendida cuales han sido las motivaciones que han servido de bases para promover este proyecto. Actualmente, la situación financiera en España es delicada, al igual que en la educación pública, por ello todo dinero invertido en ella debe ser aprovechado. Aun así, el eje central del proyecto consiste en ayudar a los estudiantes. Como se ha mencionado en apartados previos, una de las principales causas del abandono universitario es debido a los malos hábitos de trabajo. La idea de este proyecto es ayudar a los alumnos en el proceso de aprobado y que puedan centrarse en el aprendizaje.

Esta herramienta por si sola no mejorará la situación actual de forma automática, en última instancia el responsable del resultado académico es el propio estudiante. Pero sí servirá para los alumnos que tengan dificultades para organizarse y su bajo rendimiento no provenga de falta de voluntad e interés.

En ningún caso se ha de interpretar la elaboración de este proyecto como una crítica a la atención que reciben los alumnos por parte de los docentes, no se puede realizar una tutoría continuada y completa a cada uno de los estudiantes. Por ello, creemos que existe una necesidad real en cuanto a este tipo de soluciones, ya que en toda asignatura existe una franja de alumnos que han suspendido con una nota que oscila entre el 4 y 4,9, pudiendo haber aprobado con una mejor organización.

Ningún colectivo se verá perjudicado por este proyecto, no se están restando competencias a los docentes, tan solo se ofrece un soporte adicional a los alumnos.

15.3 Impacto ambiental

Teniendo en cuenta tan solo la herramienta que se a desarrollado en el proyecto, se podría decir que desde el punto de vista ambiental el coste de implantación es elevado, ya que se requiere de servidores donde alojar el sistema predictivo y bases de datos con toda la información del estudiante. Pero como se ha mencionado en múltiples ocasiones, la idea del proyecto es que las diferentes universidades (y otros centros docentes) adopten esta solución. Al no ser una herramienta muy costosa en calculo computacional, estos centros no requerirán de nuevos servidores para poder utilizarla, sino que podrán incluir el sistema predictivo en los servidores que ya dispongan en ese momento. Por lo tanto, no se generará consumo eléctrico adicional (y en caso de que así fuera, se puede considerar negligible) ni se requerirá la producción de nuevos servidores.

La realización del proyecto por parte de una empresa, con fines comerciales, no supondría una mejora en el aspecto ambiental, en todo caso el consumo e impacto sería mayor. Los recursos materiales utilizados en las empresas se adquieren específicamente para este tipo de labores, a diferencia de este proyecto, donde el portátil utilizado se utiliza para tareas del día a día ajenas al trabajo. También hay que tener en cuenta que la empresa no asignaría un único empleado al desarrollo, aumentando el número de recursos utilizados de manera proporcional a los empleados que trabajen en él (en cuyo caso, también se ha de mencionar que el tiempo invertido en el producto sería potencialmente menor al actual).

Para aprovechar el trabajo realizado por otras entidades y evitar invertir tiempo en algo ya investigado por otras personas, se contemplará la utilización de librerías externas que ofrecen un gran soporte para los sistemas predictivos. Además, como se especificó en el apartado *Estado del arte,* los datos utilizados en el estudio y las hipótesis que se van a comprobar provienen de otros estudios, evitando así la duplicidad de trabajo (y consecuentemente de recursos).

Por último, mencionar que nuestra intención es que los resultados de este estudio sean públicos, tanto el documento descriptivo como el código. Por lo tanto, que puedan utilizarlo otras personas, mejorarlo e incluso adoptarlo para diferentes casos de uso, ya que un modelo predictivo se puede entrenar para todo tipo de situaciones.

15.4 Matriz de sostenibilidad

Este apartado muestra de forma gráfica las puntuaciones otorgadas a cada uno de los apartados anteriores. La matriz de sostenibilidad es un recurso utilizado en diferentes proyectos y que aplica a la Puesta a Punto del Proyecto, PPP (fase de planificación y desarrollo) además de la vida útil y los riesgos que puedan surgir a lo largo de esta. Debido a las limitaciones temporales del proyecto no se puede considerar la fase de implantación, pero se tendrá en cuenta el coste de su vida útil como si así fuera.

	PPP	Vida Útil	Riesgos
Ambiental	Consumo del diseño	Huella Ecológica	Riesgos ambientales
	7	16	-1
Económico	Factura	Plan de viabilidad	Riesgos económicos
	9	8	-1
Social	Impacto personal	Impacto social	Riesgos sociales
	7	15	0
Rango Sostenibilidad	23	39	-2
	60 sobre 90		

Tabla 19: Matriz de sostenibilidad

La mayoría de la puntuación indicada en la tabla puede encontrar su justificación en los apartados referentes al curso de gestión de proyectos. Aun así, me gustaría dedicarle unas líneas a hablar del impacto personal que ha tenido en mí este proyecto.

A menudo, muchos alumnos tienden a focalizar las críticas de sus malos resultados académicos en el profesorado, y siendo justificado en algunas situaciones, no siempre considero que sea el caso. Hay una gran variedad de factores que influyen en el proceso de aprendizaje de los estudiantes, pero como opinión personal, creo que el problema principal radica en la incorrecta interpretación del significado "aprender". Hoy en día, es muy común por parte de los docentes y los alumnos confundir los buenos resultados en los exámenes con aprender, cada vez más, el sistema educativo está enseñando a los estudiantes a aprobar exámenes antes que ofrecerles conocimientos realmente útiles. A lo largo de la carrera, una práctica muy habitual por parte de los alumnos, consiste en estudiar los exámenes anteriores y aprender a hacerlos igual en vez de preocuparse por entender la base del conocimiento. Me gustaría remarcar que esto es tanto problema de los docentes como de los estudiantes.

Esta herramienta no pretende (ni tampoco es capaz) de cambiar esta situación, pero sí que puede ayudar a los estudiantes a gestionarse mejor y conseguir el tan ansiado aprobado a través de una tutoría continuada (y limitada) que intente advertir al alumno de la importancia de la realización del trabajo diario y no solo en períodos próximos a exámenes.

16. Conclusiones

A lo largo de todo el proceso de trabajo, que ha tenido una duración de cuatro meses, se han conseguido cumplir los objetivos propuestos sin la necesidad de modificar la planificación inicial. Se ha elaborado un sistema predictivo cuya precisión es del 80%, y se dispone de una plataforma web que permite hacernos una idea de la utilidad que puede tener este tipo de información.

En todo momento se definió un alcance realista, siendo evidente que la solución final es tan solo un prototipo que dista mucho de un producto final listo para su uso real, aun así, el sistema es totalmente funcional y robusto. Realizando un breve análisis, no encontramos ninguna solución con un objetivo similar, por lo tanto, creemos que este tipo de soluciones tiene un futuro muy prometedor todavía por investigar y que puede resultar realmente útil tanto para los estudiantes como las instituciones docentes.

A pesar de los buenos resultados obtenidos queda mucho trabajo por hacer, a partes iguales entre una mejora de la precisión del sistema predictivo y añadir funcionalidades adicionales a las plataformas web de cada universidad. En los siguientes apartados vamos a analizar el transcurso del proyecto, así como todos los cambios que requiere la plataforma a corto y largo plazo.

16.1 Revisión de compromisos

Los objetivos iniciales del proyecto, de forma general, consistían en la elaboración de un sistema predictivo y una plataforma web sencilla donde poder mostrar los resultados. A lo largo de todo este documento se ha explicado el proceso de realización, así como los resultados obtenidos. Adicionalmente, también se han respetado los objetivos segundarios, como la **privacidad de los datos**, tan solo puede acceder a ellos el propio estudiante, mantener el **sistema actualizado**, las predicciones se elaboran diariamente, y, por último, el diseño de la plataforma web es **sencillo** a la par que **útil**.

En todo proyecto de software es muy común la aparición de imprevistos que provoquen modificaciones en la planificación inicial. Dada la naturaleza de este proyecto, los factores condicionantes estaban claramente identificados, la dificultad para obtener datos sobre estudiantes que poder utilizar, y el aparentemente poco soporte comunitario que existe en las diversas librerías y herramientas basadas en machine learning. Aun así, ninguno de estos posibles riesgos ha afectado al proyecto, por lo que la planificación inicial (explicitada en el apartado *6.2 Planificación y Dependencias*) ha seguido vigente a lo largo del proyecto. Previo al inicio del proyecto ya se obtuvieron datos de alumnos y se comprobó su validez, del mismo modo que las librerías utilizadas para la elaboración de los algoritmos predictivos han demostrado ser muy estables y con un mayor soporte del inicialmente mencionado.

En cuanto a los costes económicos, factor importante en cualquier proyecto didáctico o empresarial, tampoco se ha requerido ninguna inversión adicional. En los distintos escenarios contemplados, la causa principal que podía hacer aumentar el presupuesto dependía de las variaciones temporales estimadas en la planificación, ya que eso supondría más horas de trabajo de los "empleados", aumentando el coste en salarios.

16.2 Trabajo futuro

En el apartado *14.8 Análisis de los resultados*, se realiza un análisis de los resultados obtenidos por el sistema predictivo, centrándose sobre todo en los factores condicionantes que han afectado al proyecto. A modo de resumen, se señalan como elementos influyentes, la utilización de tan solo dos hipótesis para la predicción, ya que en la gran mayoría de casos de uso el número de variables de entrada para los algoritmos de *machine learning* es considerablemente superior. También hay que tener en cuenta que el dataset utilizado para realizar las predicciones pertenece a alumnos de otras regiones geográficas, donde sus patrones de comportamiento o metodologías de trabajo pueden distar mucho de las de otros países.

Por lo tanto, los siguientes pasos a seguir para mejorar tanto los resultados como la utilidad de las predicciones consiste en analizar los patrones de comportamiento de los alumnos para elaborar **nuevas hipótesis**. En el caso del paper *[4]OU Analyse: Analysing at-risk students at The Open University,* señalan que los resultados de sus algoritmos predictivos mejoraron considerablemente al tener en cuenta información adicional tales como datos demográficos o estadísticas derivadas de la interacción de los estudiantes en el foro de la universidad. Utilizar las mismas hipótesis que otros estudios puede ofrecer resultados razonables (como en este proyecto), pero la metodología de trabajo y comportamiento de los estudiantes puede variar mucho dependiendo del país o del tipo de carrera, incluso dentro de cada carrera, según el centro docente donde se realicen los estudios. Por lo tanto, el primer paso para mejorar los resultados, como ya hemos mencionado al inicio del parágrafo, consiste en la elaboración de nuevas hipótesis a partir del análisis de la información generada por los estudiantes de la **misma universidad** donde se aplicará la solución.

En cuanto a la plataforma web, el factor diferencial de este proyecto frente a otras soluciones parecidas, existen un gran nombre de funcionalidades susceptibles de ser añadidas, algunas de las cuales ya han sido comentadas en el apartado *15.7 Posibles funcionalidades adicionales*. Es muy importante no incluir excesivas funcionalidades, que, en última instancia sobrecarguen de información al alumno y provoquen un efecto opuesto al deseado.

Muchos de los estudios que se han consultado durante la elaboración del proyecto tenían como destinatarios de la información a las propias instituciones docentes o profesores, los cuales serían los encargados de advertir a los estudiantes llamándoles o enviando notificaciones. En cambio, esta solución apuesta porque el usuario sea el encargado de autogestionarse como considere oportuno a partir de la información ofrecida.

En el prototipo que se ha elaborado nos centramos en ofrecer de forma rápida y clara un resumen del estado de cada asignatura, así como una breve comparación entre el rendimiento de los distintos matriculados en cada una de ellas. En las siguientes iteraciones, el objetivo consiste en aumentar el rango de información ofrecida sin que esta resulte excesiva. Algunas de las posibles propuestas se enumeran a continuación.

- Una idea muy interesante, extraida de *[4]OU Analyse: Analysing at-risk students at The Open University,* propone no centrarse tan solo en el resultado final del curso, si no también en avisar al alumno si su rendimiento actual indica que el resultado de la siguiente práctica que ha de entregar esta "en peligro".

- Como hemos comentado en apartados anteriores, creo que las técnicas de gamificación (utilización de mecanismos de los juegos) tienen una utilidad real aplicadas al campo pedagógico. En el caso particular de la universidad en la que se está desarrollando este proyecto, encontramos un claro ejemplo de estas mecánicas en la asignatura de *Estructuras de datos y algoritmos*, dónde se realiza una práctica sobre algoritmos de recorrido de grafos utilizando como ejemplo distintos juegos (pacman, tron legacy, dragon ball, etc.), para posteriormente llevar a cabo una competición entre los algoritmos elaborados por todos los estudiantes. Elementos como los avatares, reputación, rankings, niveles, sistemas de realimentación, reglas, etc.

- Por último, para poder incrementar el rango de funcionalidades en general, sería útil empezar a combinar la información de la plataforma web propuesta en este proyecto con los espacios virtuales de las universidades.

Apéndice 1: Obtención del nombre del curso a partir de Learning Locker

Este apéndice muestra a través de un ejemplo como obtener el nombre del curso (o asignatura) al que está asociado un evento.


```
"grouping": [
        {
          "objectType": "Activity",
          "id": "http://127.0.0.1",
          "definition": {
            "type": "http://id.tincanapi.com/activitytype/site",
            "name": {
              "en": "Proyecto final Machine Learning"
            },
            "description": {
              "en": "Proyecto final Machine Learning"
            }
        {
          "objectType": "Activity",
          "id": "http://127.0.0.1/course/view.php?id=2",
          "definition": {
            "type": "http://lrs.learninglocker.net/define/type/moodle/course",
            "name": {
              "en": "Curso primera prueba"
            },
            "description": {
              "en": "Este curso es una pruba para ver la informaci\u00f3n que se envia a Learning locker cuando completas assignements, cuando lees/abres material del curso..."
            },
```


Figura 40: Fragmento de JSON para obtener el nombre del curso

Como se especificó en el apartado *14.2 Formato sentencia almacenadas en Learning Locker*, el campo Context dispone de un atributo llamado grouping, donde se muestran las relaciones que tiene el statement con otras actividades. En este caso encontramos la información del curso al que pertenece, para ello, se han de utilizar dos atributos. Por una parte, hemos de asegurarnos que el atributo type del objeto objectType corresponde a /moodle/course. Una vez se ha identificado que este objeto hace referencia a un curso, tan solo hemos de consultar el nombre del mismo. Para cada evento almacenado en Learning Locker, solo puede haber un objectType /moodle/course.

Apéndice 2: Identificación de eventos en Learning Locker

Este apéndice muestra a través de un ejemplo como identificar si un evento corresponde a la entrega de una práctica, en cuyo caso, además se tendrá que obtener los días de margen respecto a la fecha límite.

```
   "verb": {
      "id": "http://adlnet.gov/expapi/verbs/completed",
      "display": {
         "en": "completed"
      }
"object": {
      "objectType": "Activity",
      "id": "http://127.0.0.1/mod/assign/view.php?id=7",
      "definition": {
         "type": "http://lrs.learninglocker.net/define/type/moodle/assign",
         "name": {
            "en": "Assignment_1"
         },
         "description": {
            "en": "A module"
         }
}
```

Figura 41: Fragmento de JSON para detectar si se trata de un entregable

El primer paso consiste en identificar si estamos tratando con un evento más o se trata de la entrega de una práctica. Para ello, tan solo se ha de comprobar que el verbo de la sentencia sea “completed” y que el tipo de objeto sobre el que tiene lugar la sentencia sea /moodle/assign. En caso de que ambos criterios se cumplan, tendremos que obtener también los días de margen que ha tardado el alumno en completarlo. Vamos a utilizar el siguiente fragmento para ello.

```
"extensions": {
        "http://lrs.learninglocker.net/define/extensions/moodle_module": {
          "id": "1",
          "course": "2",
          "name": "Assignment_1",
          "intro": "",
          "introformat": "1",
          "alwaysshowdescription": "1",
          "nosubmissions": "0",
          "submissiondrafts": "0",
          "sendnotifications": "0",
          "sendlatenotifications": "0",
          "duedate": "1481670000",
          "allowsubmissionsfromdate": "1481497200",
          "grade": "100",
          "timemodified": "1481558647",
          "requiresubmissionstatement": "0",
          "completionsubmit": "0",
          "cutoffdate": "0",
          "teamsubmission": "0",
          "requireallteammemberssubmit": "0",
          "teamsubmissiongroupingid": "0",
          "blindmarking": "0",
          "revealidentities": "0",
          "attemptreopenmethod": "none",
          "maxattempts": "-1",
          "markingworkflow": "0",
          "markingallocation": "0",
          "sendstudentnotifications": "1",
          "preventsubmissionnotingroup": "0",
          "type": "assign",
          "url": "http://127.0.0.1/mod/assign/view.php?id=7"
```


Figura 42: Fragmento de JSON para obtener la fecha final y la de entrega

Para calcular los días de margen, se utiliza el atributo duedate, que como se puede deducir, corresponde con la fecha límite permitida para realizar la entrega junto al atributo timemodified. Este último atributo se actualiza cada vez que se interacciona con el material (adjuntar un fichero, contestar una pregunta, marcar la entrega como completada, etc.). Originalmente se pensó en utilizar el objeto timestamp facilitado por Learning Locker, pero como se mencionó en su momento, este objeto lo añade el propio LRS, y la fecha y hora corresponden al de la zona horaria en la que este configurado. Para evitar posibles errores derivados de esto, se decidió utilizar el atributo timemodified.

Agradecimientos

Cada año, en los premios Óscar, resulta evidente que los discursos de agradecimiento citando a todos los participantes de las películas son imposibles a la par que aburridos. Como no pretendo provocar ese efecto entre los lectores (si no lo he hecho ya), voy a ser breve.

Todas las personas y animales que me han ayudado a lo largo de este proceso recibirán mi agradecimiento en persona, a excepción de uno, va por ti, Peter.

Referencias Bibliográficas

[1] Bethencourt Benítez, J.T., Cabrera Pérez, L., Hernández Cabrera, J.A., Álvarez Pérez, P.R., González Afonso, M., 2008. Variables psicológicas y educativas en el abandono universitario. Electronic journal of research in educational psychology 6, 603–622.

[2] Bovo, A., Sanchez, S., Duthen, Y., Héguy, O., 2013. Demonstration of a Moodle student monitoring web application. Proceedings of the 6th International Conference on Educational Data Mining.

[3] Młynarska, E., Greene, D., Cunningham, P., 2016. Indicators of Good Student Performance in Moodle Activity Data.

[4] Kuzilek, J., Hlosta, M., Herrmannova, D., Zdrahal, Z. and Wolff, A. OU Analyse: Analysing At-Risk Students at The Open University. Learning Analytics Review, no. LAK15-1, March 2015, ISSN: 2057-7494.

[5] Baker, R.S., Lindrum, D., Lindrum, M.J., Perkowski, D., 2015. Analyzing Early At-Risk Factors in Higher Education e- Learning Courses. Proceedings of the 8th International Conference on Educational Data Mining 150–155.

[6] Berking, P., & Gallagher, S. (2011). Choosing a learning management system. *Advanced Distributed Learning (ADL) Co-Laboratories,(2.4)*.

[7] Mitchell, T.M., 1997. Machine Learning, Annual Review Of Computer Science. McGraw-Hill, Inc., New York, NY, USA. doi:10.1145/242224.242229

[8] Clarenc, C. A.; S. M. Castro, C. López de Lenz, M. E. Moreno y N. B. Tosco (Diciembre, 2013). Analizamos 19 plataformas de e-Learning: Investigación colaborativa sobre LMS. Grupo GEIPITE, Congreso Virtual Mundial de e-Learning. Sitio web: www.congresoelearning.org

Webgrafías

[9] Scikit-learn.org. *scikit-learn: machine learning in Python — scikit-learn 0.18.1 documentation*. [online] Available at: http://scikit-learn.org/stable/index.html [Accedido el 21 de febrero de 2017]

[10] Adlnet.gov.*SCORM - ADL Net*. [online] Available at: https://www.adlnet.gov/adl-research/scorm/. [Accedido el 2 de febrero de 2017]

[11] Adlnet.gov. *xAPI Technical Specifications - ADL Net*. [online] Available at: https://www.adlnet.gov/adl-research/performance-tracking-analysis/experience-api/xapi-technical-specifications/. [Accedido el 4 de febrero de 2017]

[12] Aicc.github.io. *The cmi5 Project*. [online] Available at: http://aicc.github.io/CMI-5_Spec_Current/ [Accedido el 2 de febrero de 2017]

[13] Software, R. *Home*. [online] SCORM. Available at: https://scorm.com/. [Accedido el 2 de febrero de 2017]

[14] Hughan, K., Miller, B. and Martin, T. *Tin Can API - Programmable E-learning and Experience Tracking*. [online] Experienceapi.com. Available at: http://experienceapi.com/ [Accedido el 2 de febrero de 2017]

[15] E-abclearning.com. *e-ABC es e-Learning Sin Límites*. [online] Available at: http://www.e-abclearning.com/index.php. [Accedido el 4 de febrero de 2017]