



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



**TFG del Grado en Ingeniería
Informática**

**Evaluación de calidad de
cursos implementados en la
versión de Moodle 4.x**



Presentado por Bilal Azar El Mourabit
en Universidad de Burgos — 29 de mayo
de 2024

Tutor: Carlos López Nozal
Tutor: Raúl Marticorena Sánchez



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



D. Carlos López Nozal, profesor del departamento de Ingeniería Informática, área de Lenguajes y Sistemas Informáticos.

D. Raúl Marticorena Sánchez, profesor del departamento de Ingeniería Informática, área de Lenguajes y Sistemas Informáticos.

Expone:

Que el alumno D. Bilal Azar El Mourabit, con DNI dni, ha realizado el Trabajo final de Grado en Ingeniería Informática titulado título de TFG.

Y que dicho trabajo ha sido realizado por el alumno bajo la dirección del que suscribe, en virtud de lo cual se autoriza su presentación y defensa.

En Burgos, 29 de mayo de 2024

Vº. Bº. del Tutor:

Vº. Bº. del co-tutor:

Carlos López Nozal

Raúl Marticorena Sánchez

Resumen

En estos últimos años hemos sido testigos del crecimiento de la docencia en línea en múltiples instituciones académicas y empresas. Según estadísticas [1] el elearning ha crecido un 900 % desde el año 2000. Las universidades, las empresas, los centros de formación vial, las academias y muchas más empresas e instituciones están optando por este tipo de educación, por ser más barata, más rápida y la preferida por los estudiantes. La docencia en línea es la nueva forma de educar y requiere de marcos de calidad que ayuden a conseguir los mejores resultados en la educación y la formación a través de internet.

En este proyecto se desarrolla una herramienta para medir la calidad de los cursos de Moodle. Para ello nos apoyamos en el estándar de calidad Quality Reference Framework for the quality of Moocs [12]. Además partimos de una versión anterior que funciona con las versiones 3.x de Moodle, por lo que una de las primeras ideas es hacer funcionar esta herramienta con versiones 4.x.

Para lograr este objetivo nos servimos de la api de Moodle [23] que ofrece una cantidad de información suficiente para generar informes de calidad interesantes y útiles para poder aprovechar al máximo las capacidades de Moodle.

Descriptores

moodle,calidad,springboot,maven,docker,cursos de moodle,framework de calidad.

Abstract

In recent years we have witnessed the growth of online teaching in many academic institutions and companies. According to statistics [1] e-learning has grown by 900 % since 2000. Universities, companies, road training centers, academies and many more companies and organizations are now using e-learning, academies and many more companies and institutions are opting for this type of education, as it is cheaper, faster and preferred by students. Online education is the new way of education and requires quality frameworks that help achieve the best results in education and formation through the Internet.

Translated with DeepL.com (free version)

In this project we develop a tool to measure the quality of Moodle courses. For this purpose we rely on the quality standard Quality Reference Framework for the quality of Moocs [12]. In addition, we start from an earlier version that works with Moodle 3.x versions, so one of the first ideas is to make this tool work with 4.x versions.

In order to achieve this goal we use the Moodle api [23] which offers a sufficient amount of information to generate interesting and useful quality reports to make the most of Moodle's capabilities.

Keywords

moodle, quality, springboot, maven, docker, moodle courses, quality framework.

Índice general

Índice general	iii
Índice de figuras	v
Índice de tablas	vi
1. Introducción	1
2. Objetivos del proyecto	3
3. Conceptos teóricos	5
3.1. Definiciones básicas	5
3.2. Marco de referencia de calidad para MOOCs	6
3.3. Buenas prácticas en la docencia en línea	9
3.4. Plan de calidad para cursos e-Learning	12
4. Técnicas y herramientas	17
4.1. Diseño dirigido por el dominio	17
4.2. Modelo-Vista-Controlador	18
4.3. Backend	18
4.4. Frontend	19
4.5. Modelo de desarrollo software	19
5. Aspectos relevantes del desarrollo del proyecto	21
5.1. Arreglar el fallo con las nuevas versiones de Moodle	21
5.2. Arreglar la portabilidad del proyecto con ejecutable .war	22
5.3. Implementación de Docker en el proyecto	23
5.4. Nueva regla: Definición de fechas y descripción del curso	23

5.5. Implementación de reglas de cuestionarios	24
5.6. Implementación de funcionalidad de exportación de informe en Excel	27
6. Trabajos relacionados	29
7. Conclusiones y Líneas de trabajo futuras	31
Bibliografía	33

Índice de figuras

3.1. Fases del desarrollo del Marco de referencia de calidad para MOOCs	8
3.2. Fórmula de índice de facilidad de estadísticas de reporte de Moodle	11
3.3. Fórmula de índice de discriminación de estadísticas de reporte de Moodle	11

Índice de tablas

3.1. Porcentaje de índices de facilidad con sus dificultades	12
3.2. Objetivos de calidad en base a referencias teóricas. Leyenda: Responsabilidad: R=Responsable X=Involucrado Perspectivas: P=Pedagógica T=Tecnológica E=Estratégica	12
3.2. Objetivos de calidad en base a referencias teóricas. Leyenda: Responsabilidad: R=Responsable X=Involucrado Perspectivas: P=Pedagógica T=Tecnológica E=Estratégica	13
3.2. Objetivos de calidad en base a referencias teóricas. Leyenda: Responsabilidad: R=Responsable X=Involucrado Perspectivas: P=Pedagógica T=Tecnológica E=Estratégica	14
3.2. Objetivos de calidad en base a referencias teóricas. Leyenda: Responsabilidad: R=Responsable X=Involucrado Perspectivas: P=Pedagógica T=Tecnológica E=Estratégica	15

1. Introducción

eLearningQA es una herramienta que trabaja sobre Moodle y que permite a sus usuarios visualizar informes sobre sus cursos de Moodle. Esta herramienta proporciona un valor importante para los diseñadores de los cursos, para profesores que administren, o para cualquier gestor de cursos en general. Para el desarrollo correcto de esta aplicación, hemos investigado sobre la calidad de cursos en línea masivos y abiertos (MOOC).

Por otro lado, hemos aplicado los conocimientos adquiridos durante el transcurso del grado por la Universidad de Burgos, para hacer un proceso de desarrollo acorde a las buenas prácticas de desarrollo software, a las metodologías ágiles y otros conceptos importantes.

Además, nos hemos basado en estándares de calidad definidos con el fin de crear marcos de calidad para que profesores y diseñadores tengan una guía para desarrollar sus cursos. Con estos marcos disponemos de una guía para desarrollar una aplicación basada en conceptos estudiados y formalizados por profesionales de la enseñanza.

El objetivo de calidad de eLearningQA es conseguir el aprovechamiento máximo de los recursos a disposición de una institución educativa con el fin de que sus estudiantes y profesores consigan el mayor desarrollo intelectual y personal posible, dentro los tiempos marcados.

Los documentos en los que nos hemos basado y han definido el progreso de la aplicación son: el estándar de calidad para moocs [12] que nos ha dado la base para hacer un desarrollo ordenado y razonado. Este marco define distintas dimensiones (fases, perspectivas y roles) que abarcan el proceso de la enseñanza a distancia desde que se comienza el análisis para la creación de un curso hasta que el curso termina. El estudio de modelos de calidad de elearning [26] es otro documento que define claramente la calidad

en la docencia en línea y da ejemplos de modelos de elearning de calidad. En este documento pueden ver los resultados positivos en docencia en línea de varias instituciones académicas y los sistemas de educación de los que disponen para lograr esos resultados.

Este proyecto ha pasado por varias versiones anteriormente. Con las nuevas versiones de Moodle hemos encontrado problemas de compatibilidad por lo que en esta nueva versión hemos habilitado la aplicación para que pueda funcionar con las versiones 4.x de Moodle, además hemos solucionado problemas de portabilidad y aumentado funcionalidad.

En este documento, se explican todos los aspectos relacionados con el proyecto, tanto la parte práctica como la teórica. Mostraremos el proceso de desarrollo, los aspectos técnicos y las contingencias encontradas durante el mismo, además, se presentan conclusiones con los aspectos importantes a mencionar.

2. Objetivos del proyecto

Los objetivos del proyecto se pueden dividir en dos. Los objetivos funcionales, entendidos como aquellos que comprenden las funcionalidades que deseamos que tenga la aplicación desarrollada, y los objetivos técnicos, entendidos como las características técnicas que debe tener la aplicación.

En cuanto a los objetivos funcionales, queremos que la aplicación permita a las personas que tengan permisos de gestión en cursos de Moodle, poder ver un informe detallado sobre las distintas fases del curso, entre las que encontramos:

- Diseño
- Implementación
- Realización
- Evaluación

Cada apartado de estos tiene unas métricas y cada una de estas tienen una puntuación que media en un computo global. Todas estas notas se muestran. Con esto se ve reflejada la calidad del curso.

Por otro lado, queremos que la aplicación sugiera al usuario cambios a realizar para mejorar, estos cambios se basan en las métricas mencionadas anteriormente. Además, se mostrara una gráfica de evolución temporal de la calidad del curso. Esto se hará en base tres roles:

- Diseñador
- Facilitador

- Proveedor

Y tres perspectivas:

- Pedagógica
- Tecnológica
- Estratégica

Además la aplicación debe tener un manual para que el usuario la utilice como guía para utilizar para la aplicación. También debe tener apartados para el contacto con los administradores de la página y un apartado para conocer la gestora de la página.

También queremos realizar el mantenimiento de la versión existente[16] de la aplicación actualizando aquellas dependencias que sean necesarias y programando la compatibilidad con Moodle 4.x, para que se puedan evaluar cursos que estén en estas versiones.

En cuanto a los objetivos técnicos, queremos que el proyecto siga un desarrollo basado en metodologías ágiles, dividiendo los avances en sprints, en los que se realizan una serie de tareas y se hacen reuniones en la fecha límite en cada uno para ver presentar los avances y recibir retrospectivas. Para seguir este proceso se utilizan herramientas como Zube y las issues de GitHub que permiten que todos los componentes del equipo puedan monitorizar los cambios y avances que se estén realizando.

Paralelamente, queremos que la aplicación este basada en tecnologías escalables y fácilmente mantenibles. Para esto, se utiliza SpringBoot con Maven, que proporciona la capacidad de desarrollar y desplegar una aplicación rápidamente, además de proporcionar una configuración automática y las ventajas de la inyección de dependencias. Adicionalmente, queremos utilizar patrones de diseño para resolver los problemas que puedan surgir desarrollando la aplicación, siguiendo así unas buenas prácticas.

3. Conceptos teóricos

A continuación explicaremos algunos conceptos teóricos, es importante recalcar que los conceptos que explicamos aquí están apoyados en la explicación de los mismo que encontramos en el TFG anterior a este. ¹.

3.1. Definiciones básicas

- **Administración de la calidad:** es el conjunto de actividades para definir un proceso de mejora continua: aseguramiento de calidad, planificación de calidad y control de calidad.
- **Aseguramiento de la calidad:** establecimiento de un marco de trabajo, de procedimientos y estándares que llevan a un curso de alta calidad.
- **Planificación de la calidad:** selección de los procedimientos y estándares del marco de trabajo y adaptación para un curso concreto y un software específico.
- **Control de calidad:** procedimientos y estándares que guían al equipo de trabajo.
- **Medición:** el proceso por el cuál se asignan números o símbolos a los atributos las entidades del curso en línea, de tal forma que los caracteriza de manera clara a través de reglas o consultas sobre LMS.
- **E-learning:** proceso de enseñanza y aprendizaje impartido por medios electrónicos y digitales.

¹Créditos al proyecto de Roberto Arasti Blanco: eLearningQA

- **LMS:** sistema de gestión de aprendizaje que permite gestionar la enseñanza a individuos y grupos de personas, como por ejemplo Moodle, Edmodo o Blackboard.
- **Moodle:** es LMS creado por Martin Dougiamas en 2002 y destaca por ser el mas extendido en el mundo y por ser altamente personalizable.
- **API de Moodle:** conjunto de definiciones y protocolos que permite a software de terceros interactuar con el software de Moodle. Estas definiciones las encontramos en la documentación oficial[23].
- **MOOC:** Massive Open Online Courses o Cursos En Línea Gratuitos Masivos, hacen referencia a cursos basados en e-learning que cada vez están más extendidos en los centro educativos de todo el mundo.
- **Gateway:** Puerta de enlace entre datos, archivos y funcionalidades gestionadas por el servidor y los usuarios [9].

3.2. Marco de referencia de calidad para MOOCs

Uno de los documentos más importantes para el desarrollo de este proyecto es el Marco de referencia de calidad para MOOCs[12] expedido por la Alianza Europea para la calidad de los MOOCs. Para la creación de este marco se obtuvo retroalimentación de diez mil personas, todas ellas participantes de algún curso en línea, tanto como diseñador, educador, proveedor o consumidor. Con toda la información obtenida de las opiniones de las distintas personas involucradas en el proceso se concluyeron unas visiones generales para que las personas encargadas de la gestión de algún curso en línea puedan seguir.

En primer lugar hay que entender que existen 3 dimensiones a la hora de crear cursos en línea de calidad: las fases, las perspectivas y los roles.

Fases

En este apartado encontramos 5 fases que definen el proceso previo a la creación del curso hasta la consumición del curso por parte del alumnado.

1. **Análisis:** Se comprenden tareas como la identificación y descripción de casos de uso y requerimientos, así como los límites y las necesidades.

Llevado a un caso concreto podríamos pensar en la identificación de los estudiantes objetivo y su nivel académico o la planificación temporal y financiera. El estudio de la necesidad de cuestionarios y sus objetivos.

2. **Diseño:** Se comprenden tareas de conceptualización y diseño de los cursos. Llevado a un caso concreto podríamos pensar en la organización de los conceptos y los roles o la definición de los objetivos de aprendizaje. El diseño de cuestionarios y actividades que animen la interacción.
3. **Implementación:** Se comprenden las tareas de implementación de una versión inicial del MOOC y las pruebas del mismo. Llevado a un caso concreto podemos pensar que una vez creado el curso se implementan contenidos técnicos o recursos y se prueban. Implementación de los cuestionarios diseñados y asegurar su correcto mantenimiento.
4. **Realización:** Se comprenden las tareas de puesta en funcionamiento del MOOC así como el soporte. Llevado a un caso concreto podríamos pensar en la administración del curso o la revista de niveles de competencia. Interacción del estudiantado con los cuestionarios del curso.
5. **Evaluación:** En esta fase se comprenden todas las tareas de evaluación del resto de fases. Llevado a un caso concreto podríamos pensar en la evaluación del plan previo, la evaluación de cursos o las mejoras y la optimización de un curso.

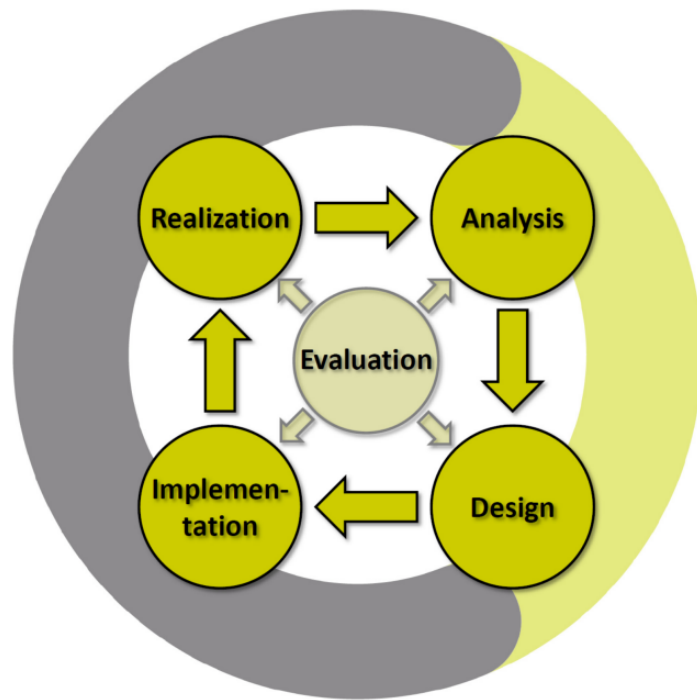


Figura 3.1: Fases del desarrollo del Marco de referencia de calidad para MOOCs

Perspectivas

Las perspectivas se pueden entender como las características objetivo de cada uno de los procesos de las fases.

1. **Pedagógicas**
2. **Tecnológicas**
3. **Estratégicas**

Ilustremos esto con un ejemplo, en la fase de implementación, en el proceso de implementación de contenido, tenemos dos perspectivas principales: tecnológicas y pedagógicas, ya que solo se engloban aquellas tareas de implementación de los contenidos académicos y de soporte de ese contenido.

Roles

En esta dimensión encontramos a aquellos actores que tengan alguna responsabilidad o estén involucrados en los procesos de las fases.

1. **Diseñador:** Se engloban los expertos en contenido, autores de contenido, diseñadores gráficos y expertos en plataformas LMS, así como todos los individuos que puedan contribuir al diseño del MOOC.
2. **Facilitador:** Se engloban los expertos pedagógicos como tutores, moderadores y profesores, así como todos los individuos que puedan contribuir al proceso de aprendizaje del MOOC.
3. **Proveedor:** Se engloban los proveedores técnicos como programadores, soporte técnico, desarrolladores de software así como todos los individuos que puedan contribuir a la toma de decisiones que lleven a la entrega del software del MOOC.

3.3. Buenas prácticas en la docencia en línea

En la docencia en línea al igual que en el desarrollo software existen unas buenas prácticas que es conveniente seguir por el equipo educativo. Ilustraremos al lector con varias buenas prácticas basadas en la experiencia de alumnos de UBUVirtual[19]. Además nos hemos basado en conceptos del libro *Strategies for Effective Online Teaching and Learning*: [27].

Retroalimentaciones a tiempo

El personal docente debe proporcionar retroalimentaciones a los estudiantes a tiempo, de esta forma se conseguirá que el alumno siga una cronología acorde a la del curso y pueda llevar una mejor organización de sus objetivos.

Recursos actualizados

Los recursos que el equipo docente ponga a disposición del alumnado, que pueden ser enlaces, libros, apuntes, vídeos, cuestionarios, simuladores y otros, deben estar actualizados. Para ello se deben hacer comprobaciones regularmente, de esta forma el alumnado tendrá todas las herramientas necesarias a su disposición sin incidencias.

Seguimiento del alumnado

Un seguimiento continuo de la evolución del alumnado puede dar al equipo docente información clave para definir el avance del curso con el mayor éxito posible. Para realizar un buen seguimiento el docente se puede servir de cuestionarios, actividades, gráficas y otras herramientas que ofrecen los LMS.

Priorizar la comunicación y las estrategias de interacción.

El principal problema de la docencia en línea es la comunicación. En la docencia presencial, los alumnos ven diariamente a sus profesores y viceversa, esto facilita la comunicación y la capacidad del profesor para analizar el impacto en vivo que generan los contenidos impartidos en sus alumnos.

Sin embargo, en la docencia en línea la información que el profesor puede obtener de las comunicaciones es más limitada, por lo que tiene que crear nuevas formas de comunicarse, que proporcionen la mayor cantidad de información posible para impartir una docencia de mayor calidad.

Esto se tendrá que realizar mediante el uso de herramientas que todos conocemos como foros, tableros de anuncios, grupos o correos. Pero estas herramientas tan usadas no proporcionan una línea de comunicación continua y duradera en un espacio temporal como lo puede hacer conversación presencial, por lo que una muy buena práctica sería proporcionar formas de conversar entre el equipo docente y el alumnado. Esto se puede hacer implementado chats, o haciendo llamadas por medios como Skype o Teams. Además se pueden hacer reuniones semanales para discutir con los alumnos el avance del curso y los aspectos a mejorar por ambas partes.

Cuestionarios

Los cuestionarios son una herramienta muy útil para conocer una variedad de información sobre el transcurso del curso. Los cuestionarios nos ofrecen información sobre la actitud del alumnado hacia el curso, el nivel académico de los estudiantes, la evolución y otros datos. Además los LMS, como norma general, ofrecen esta funcionalidad de una forma muy completa, en el caso concreto de Moodle, existen distintas estadísticas que se pueden revisar. Ilustraremos esto con ejemplos de la documentación oficial[8]. Estas estadísticas proporcionan al personal académico la capacidad de detectar preguntas que están mal planteadas, entender como responden los alumnos

a las preguntas y asegurar que las variantes aleatorias no afecten en exceso a los resultados:

1. **Índice de facilidad:** indica el porcentaje medio de la puntuación de los estudiantes en una pregunta, lo que se puede interpretar como dificultad de la pregunta. La tabla 3.1 muestra los intervalos de los índices de facilidad y su interpretación.

$$F_i = 100 \frac{\bar{x}_i - x_i(\min)}{x_i(\max) - x_i(\min)}$$

Figura 3.2: Fórmula de índice de facilidad de estadísticas de reporte de Moodle

2. **Calificación aleatoria estimada:** muestra el resultado medio de una pregunta con respuestas aleatorias, interesante para evitar el fraude, ya que si el porcentaje es 0 % significa que los alumnos nunca podrán aprobar el cuestionario respondiendo a las preguntas aleatoriamente.
3. **Índice discriminatorio:** Este índice se basa en la idea de que si un alumno ha respondido bien al resto de preguntas del cuestionario, debe responder bien a la pregunta.

$$D_i = 100r(x_i, T) = 100 \frac{C(x_i, T)}{\sqrt{V(x_i)V(T)}}$$

Figura 3.3: Fórmula de índice de discriminación de estadísticas de reporte de Moodle

4. **Índice de participación:** Este índice muestra la participación que están teniendo los cuestionarios de un curso comparando los estudiantes de un curso y la participación de un cuestionario.

Índice de facilidad	Interpretación
5 % o menos	Extremadamente difícil
6 % - 10 %	Muy difícil
11 % - 20 %	Difícil
21 % - 34 %	Moderadamente difícil
35 % - 65 %	Correcta para el estudiante promedio
66 % - 80 %	Bastante fácil
81 % - 89 %	Fácil
90 % - 94 %	Muy fácil
95 % - 100 %	Extremadamente fácil

Tabla 3.1: Porcentaje de índices de facilidad con sus dificultades

3.4. Plan de calidad para cursos e-Learning

Ya hemos visto los conceptos básicos en los que hemos basado los avances realizados en el proyecto, ahora combinaremos todos los conceptos explicados para obtener un plan de calidad claro en el que se basan las reglas implementadas de la aplicación. En este plan combinamos las distintas dimensiones del marco de referencia de calidad para MOOCs[12] junto con las buenas prácticas definidas. Como resultado obtenemos una tabla 3.4 en la que encontramos los objetivos de calidad que deseamos, la responsabilidad para los roles, las perspectivas de cada uno de los objetivos y el proceso del marco de calidad para MOOCs.

Tabla 3.2: Objetivos de calidad en base a referencias teóricas.

Leyenda:

Responsabilidad: R=Responsable X=Involucrado

Perspectivas: P=Pedagógica T=Tecnológica E=Estratégica

Consulta	Perspectivas	Diseñador	Facilitador	Proveedor	Proceso
Las opciones de progreso del estudiante están activadas	P + T	R	X		D-5
Se proporcionan contenidos en diferentes formatos	P + T	R	X	X	D-4

Tabla 3.2: Objetivos de calidad en base a referencias teóricas.

Leyenda:

Responsabilidad: R=Responsable X=Involucrado**Perspectivas:** P=Pedagógica T=Tecnológica E=Estratégica

Consulta	Perspectivas	Diseñador	Facilitador	Proveedor	Proceso
El curso tiene grupos	P	R	X	X	D-5
El curso tiene actividades grupales	P	R	X	X	D-3
Los estudiantes pueden ver las condiciones necesarias para completar una actividad	P	R	X	X	
Todas las actividades tienen la misma nota máxima en el calificador	P	R	X	X	
El curso tiene fechas y descripción definidas	P + E	R	X	X	
Las preguntas de los cuestionarios tienen una calificación aleatoria adecuada	P	R	X		
Las preguntas de los cuestionarios tienen retroalimentación	P	R	X	X	
Las preguntas de opción múltiple puntúan con una calificación aleatoria estimada de cero	P	R	X	X	

Tabla 3.2: Objetivos de calidad en base a referencias teóricas.

Leyenda:

Responsabilidad: R=Responsable X=Involucrado**Perspectivas:** P=Pedagógica T=Tecnológica E=Estratégica

Consulta	Perspectivas	Diseñador	Facilitador	Proveedor	Proceso
Los recursos están actualizados	P + T	R	X	X	I-1
Fechas de apertura y cierre de tareas son correctas	P + T	X	R	X	R-2
Se detallan los criterios de evaluación (rúbricas, ejemplos)	P + T	R	X	X	R-3
El calificador no tiene demasiado anidamiento	P + E	R	X	X	
Los alumnos están divididos en grupos	T + E	X		R	I-6
El profesor responde en los foros dentro del límite de 48 horas lectivas desde que se plantea la duda	P + T + E	X	R	X	R-2
Se ofrece retroalimentación de las tareas	P + T + E	X	R	X	R-2
Las tareas están calificadas	P + T	X	R	X	R-2
El calificador muestra cómo ponderan las diferentes tareas	P + T + E	X	R	X	R-2
El tiempo de los cuestionarios está bien ajustado	P	R	X	X	

Tabla 3.2: Objetivos de calidad en base a referencias teóricas.

Leyenda:

Responsabilidad: R=Responsable X=Involucrado**Perspectivas:** P=Pedagógica T=Tecnológica E=Estratégica

Consulta	Perspectivas	Diseñador	Facilitador	Proveedor	Proceso
Los cuestionarios tienen una dificultad estimada entre unos valores umbrales	P	R	X	X	
Las preguntas de los cuestionarios son buenas discriminando	P	R	X	X	
Los índices de facilidad de las preguntas son adecuados	P + T + E	X	R	X	R-2
Los cuestionarios tienen una participación adecuada	P + T + E	X	R	X	R-2
Las preguntas de los cuestionarios tienen un índice de discriminación adecuado	P + T + E	X	R	X	R-2
La mayoría de alumnos responden a los feedbacks	P + T + E	X	X	R	E-2
Se utilizan encuestas de opinión	P + T + E	X	X	R	E-2

4. Técnicas y herramientas

Para dar inicio a este capítulo, hay que entender la estructura que tiene y como funciona internamente la aplicación. Para ello hay que pensar en como debe funcionar esta aplicación y como interactúa con el usuario.

En primer lugar, hay que recordar que eLearningQA interactúa con una API que le proporciona la mayoría de los datos, esos datos se procesan, se calculan los resultados de las reglas y se muestran. La aplicación debe estar preparada para crecer indefinidamente, ya que una de las barreras existentes actualmente para el cálculo de reglas es la falta de datos, por lo que a medida de que se puedan obtener más datos la aplicación crecerá más.

Por otro lado, existe una lógica de negocio bastante clara basada en conceptos teóricos de una variedad de estudios y marcos, por lo que es un software que gira entorno a la lógica de negocio. Adicionalmente, nos interesa que sea una aplicación fácil de mantener, ya que hay que tener en cuenta que los costos de mantenimiento son los más altos en el ciclo de vida del producto software [3].

Con esta definición se puede deducir que existen dos patrones interesantes para el funcionamiento de la aplicación.

4.1. Diseño dirigido por el dominio

Domain Driven Design o Diseño Dirigido por Dominio es una técnica de diseño software que se centra en el análisis y diseño del dominio del problema. El producto estará basada en la lógica de negocio, por lo que el dominio debe estar muy bien definido.

Puesto que la aplicación aquí desarrollada se basa en la lógica de negocio, esta opción puede parecer muy atractiva, por que además es fácil de mantener y permite escalabilidad. Sin embargo, no es la mejor opción debido a que esta aplicación ya ha tenido versiones anteriores lo que supone un alto incremento en la complejidad para la implementación de este patrón.

4.2. Modelo-Vista-Controlador

La estructura de este proyecto sigue el patrón Modelo-Vista-Controlador[6]. Este patrón arquitectónico divide el software en tres componentes:

- **Modelo:** En este componente se encuentran los datos y toda la lógica relativa a su obtención.
- **Vista:** En la vista se encuentra la interfaz gráfica y todos aquellos componentes que interactúen directamente con el usuario.
- **Controlador:** Este componente es el intermediario entre el modelo y la vista, este es el que gestiona las peticiones que se reciben de la vista y decide que datos mostrar y como, es decir, es el que contiene la lógica de negocio.

Con el patrón MVC se pueden construir páginas web escalables mantenibles y fáciles de expandir. Además es una opción atractiva dado su facilidad de implementación y su clara división entre Frontend y Backend, lo que facilita la separación de preocupaciones.

4.3. Backend

Una vez entendida la arquitectura del proyecto se puede profundizar a cada una de las partes que lo componen.

Empezando con el Backend, esta aplicación está programada en Java utilizando Spring como framework y Maven para el control de dependencias. Además cuenta con contenerización con Docker.

SpringBoot y Maven

Spring Framework proporciona un modelo de configuración y programación intuitivo y fácil de usar. El objetivo de Spring Framework[18] es realizar todas las configuraciones internas necesarias para que los equipos

de desarrollo solo tengan que encargarse de desarrollar la lógica de negocio y no dediquen excesivo tiempo a la configuración del proyecto.

Uno de los proyectos que ofrece Spring es SpringBoot[17] que permite crear stand-alone y ejecutarlas rápidamente. Además cuenta con servidores web embebidos lo que facilita enormemente el desarrollo de una aplicación web.

Para proveer de SpringBoot y de todas las dependencias que necesita esta aplicación se utiliza Maven que proporciona un fichero xml llamado POM que permite añadir todas las dependencias necesarias al proyecto fácilmente.

Docker

Docker [5] es una plataforma de desarrollo que permite desacoplar las aplicaciones de la infraestructura de forma que se puede enviar y desplegar aplicaciones rápidamente. Docker se basa en la contenerización [7].

La contenerización consiste en que las aplicaciones son empaquetadas con aquellas bibliotecas del sistema operativo necesarias para que se ejecute, de esta forma se pueden desplegar aplicaciones en cualquier sitio de forma rápida y ligera.

4.4. Frontend

El frontend se entiende como todos los componentes que interactúan directamente con el usuario y la lógica en ellos. Actualmente, existe una gran variedad de opciones para implementar el frontend de una aplicación, tales como, Angular [2], React [14] o Vue.js [21].

En el caso de eLearningQA, se utiliza JSP (Java Server Pages) [13]. Las páginas JSP son una mezcla de código HTML, XML y Java que son convertidas a un servlet el cual genera la vista. Esta ,actualmente, no es la mejor opción para el desarrollo frontend, dado que existen opciones más sencillas de implementar y con más capacidades como las nombradas anteriormente. Sin embargo, para hacer interfaces básicas y rápidamente es una buena opción.

4.5. Modelo de desarrollo software

El modelo de desarrollo software es un enfoque sistemático que describe todas las actividades del ciclo de vida del desarrollo software. Existen varios

enfoques [4] pero el que se ha aplicado en el desarrollo de este proyecto ha sido el Modelo ágil.

Modelo ágil

Este modelo se basa en la adaptabilidad, la colaboración y el desarrollo incremental. Además con este modelo fomenta la entrega rápida del software cosa muy valorada en este proyecto.

El modelo ágil consiste en la división del trabajo en iteraciones cortas llamadas sprint. Antes de cada sprint se asigna una cantidad de trabajo a cada participante en orden de prioridad. Cuando comienza el sprint se realiza el trabajo y se realizan reuniones diarias con el equipo para hacer un seguimiento de los avances. A medida que se terminan funcionalidades se integran y se realizan pruebas. Cuando termina el sprint se realiza una revisión de todas las tareas completadas y todos los miembros del equipo hacen sus comentarios sobre el sprint.

5. Aspectos relevantes del desarrollo del proyecto

5.1. Arreglar el fallo con las nuevas versiones de Moodle

Al menos el 60 % de los costes de un software va dirigido al mantenimiento, ya sea correctivo o adaptativo a nuevas funcionalidades [3]. Esta aplicación necesitaba mantenimiento y actualización. Generalmente en el mantenimiento del software nos encontramos con problemas constantemente y es deber de los ingenieros solucionarlos, además de documentar esos problemas y los pasos que se han realizado para solucionarlos, para que en el futuro si vuelven a fallar las mismas cosas, otros desarrolladores tengan una guía de como solucionar dicho error, facilitando el trabajo en el futuro. Un buen método para hacer esto es el conocido como proceso ACR (Análisis de causa raíz).[24].

Planteamiento del problema

Al inicio de este proyecto la aplicación fallaba con las versiones de Moodle superiores a la 4.0 . La información que obtuvimos era poca, recibíamos un puntero a nulo en algún punto del código.

Análisis y obtención de datos

Para solucionar el problema, se ha estudiado los puntos en los que se generaban los errores, y se han obtenido datos de todas las trazas de los errores. Esto se ha conseguido gracias a un análisis de la aplicación minucioso

con la función de debug del IDE. Además se ha probado la aplicación con la versión 3.9 de Moodle, en este caso la aplicación funcionaba correctamente.

Esto lleva a pensar que algún dato de la API ha cambiado, generando un nulo al intentar mapear dicho dato en la aplicación.

Tras más pruebas pudimos detectar que este error se debía a la variable visible de algunas respuestas de la API de Moodle. Esta variable ha cambiado de tipo desde la versión 3.10 hasta las nuevas versiones, provocando fallos en el mapeo. Esto debido que la aplicación dispone de un modelo que representa los datos que se reciben desde la API en formato JSON. Estos datos se guardan en atributos en las distintas clases del modelo. El cambio de tipo de entero a booleano en uno de los datos ha generado una incompatibilidad con varias clases del modelo.

Solución

La solución elegida ha sido interceptar el dato antes de mapearlo en nuestra aplicación. Esto consiste en recibir los datos en formato JSON de la API localizar los tipos que nos generar conflicto y convertirlos manualmente al tipo que nos interesa, seguidamente podremos vincularlo al atributo de la clase. De esta forma la aplicación funciona correctamente tanto para las versiones anteriores como las nuevas. Es una solución que se asemeja a un microservicio de interceptación [11]

Esta interceptación se podía hacer de varias formas, nosotros hemos optado por hacerla en el mismo servicio, pero existen formas para hacerlo en la clase que hace la llamada a la API[20].

5.2. Arreglar la portabilidad del proyecto con ejecutable .war

Una de las ventajas de Spring Boot es la capacidad de crear y configurar aplicaciones portables rápidamente. Sin embargo la portabilidad de nuestra aplicación fallaba, por lo que no estábamos aprovechando esas ventajas que proporciona Spring Boot al máximo.

Planteamiento del problema

El problema encontrado es que al ejecutar el comando: "java -jar prototipo-0.4-SNAPSHOT.war", el error que recibíamos era un puntero a nulo en varios

lugares del código. Este error no permitía mostrar páginas más que las de manual de usuario. y `.about`". Como dato importante, el ejecutable si que funcionaba cuando estaba en el target que genera Maven al ejecutar el comando `"mvn install"`.

Análisis y obtención de datos

Tras analizar el código hemos podido descubrir que los nulos proceden de todas las instrucciones que utilizaban alguna ruta predeterminada, por lo que podemos suponer que el problema se originaba de la obtención de ficheros mediante rutas.

Solución

Para solucionar este problema hemos decidido aprovechar la carpeta de recursos que la extensión de Spring Boot empaqueta con nuestro `.war` [15]. De esta forma se asegura que todos los ficheros que utiliza el código estén dentro del `.war` y alcanzables en todo momento.

Por otro lado, hemos cambiado la forma de obtener los fichero en el código utilizando los `InputStream`[10]

5.3. Implementación de Docker en el proyecto

Una de las nuevas características de este proyecto es la capacidad de ser desplegado con Docker [5]. Con Docker la aplicación puede ser desplegada con rapidez y eficiencia en numerosos entornos, además abre la posibilidad de dividir el frontend y el backend y ser igual de rápido y eficiente que ahora. Esto permitirá tener una interfaz mas completa e interesante de lo que pueden ofrecer las hojas JSP.

5.4. Nueva regla: Definición de fechas y descripción del curso

En el marco de calidad QRF [12] uno de los estándares de calidad es la correcta definición del curso en la fase de diseño, esto incluye especificar una fecha de inicio y una fecha de fin. Esto ha definido un requisito funcional para una nueva funcionalidad en la aplicación.

Para la implementación de esta regla se utiliza la API que proporciona información general sobre cada curso. Entre esta información se encuentra la descripción del curso, que si se encuentra como un String vacío se interpreta que no tiene y las fechas que se devuelven en formato "timestamp", que si es 0 significa que no están definidas. Con estos datos, se ha programado esta regla que permite a los profesores saber si están cumpliendo estas sencillas condiciones para tener un curso de calidad.

5.5. Implementación de reglas de cuestionarios

Una de las principales herramientas que permiten acreditar la comprensión de los contenidos impartidos en un curso son los cuestionarios [25]. Desde la perspectiva estratégica los cuestionarios guían a los involucrados en el proceso educativo mostrando la adaptación y el desarrollo de los estudiantes en el curso y sus contenidos.

Dado que los cuestionarios son tan importantes, también es importante que estén bien hechos y con un mínimo de calidad para asegurar el máximo aprovechamiento de la herramienta. Por ello, se han definido reglas en este proyecto que permiten al usuario ver varios datos importantes para la calidad de cuestionarios.

En cuanto a la implementación de reglas de este tipo, esta aplicación utiliza datos que proporciona la Api de Moodle [23] y reportes estadísticos [8] que tienen a su disposición los profesores y tutores con permisos en su organización. Para la obtención de los reportes estadísticos se ha utilizado web scraping [22]. Además, para todas las reglas, sin incluir la regla de calificación aleatorio se utilizan los cuestionarios visibles y con fecha de finalización inferior a la fecha actual o sin fecha de finalización definida.

Nueva regla: Los cuestionario tiene una participación mínima

La finalidad de los cuestionarios es que el facilitador pueda ver si sus alumnos están entendiendo los contenidos impartidos. Si los alumnos no realizan estos cuestionarios el facilitador no puede aprovechar esta herramienta. Por ende, para poder sacar conclusiones de los resultados de los cuestionarios es importante que un mínimo de alumnos participe en ellos.

Con estas bases se ha diseñado e implementado una nueva regla en el proyecto. En esta nueva funcionalidad se compara los estudiantes de un curso con los participantes de un cuestionario finalizado. De esta comparación se obtiene un porcentaje y si está por debajo del valor definido en el fichero de configuración se considera que el cuestionario tiene una baja participación.

El porcentaje de participación en caso de que sea baja se muestra al usuario para que pueda tomar las medidas convenientes para aumentar esa cifra.

Nueva regla: Los cuestionarios tienen un índice de facilidad correcto

El índice de facilidad es la puntuación media de los alumnos en una pregunta de un cuestionario. Esta estadística que proporciona Moodle en su reporte estadístico de cuestionarios ayuda a entender a los facilitadores y diseñadores la dificultad de los cuestionarios. Según la documentación de Moodle respecto a las estadísticas de cuestionarios [8] los valores adecuados para el estudiante medio se encuentran entre el 35 % y 65 %.

Para la implementación de esta regla, se lee del archivo de estadísticas de cuestionarios el índice de facilidad de cada pregunta y se realiza una media con todos los índices obtenidos para obtener un índice de facilidad global del cuestionario. Si el índice de facilidad del cuestionario está fuera del intervalo estipulado se considera que el cuestionario no está bien planteado. En caso de que el cuestionario este mal planteado se muestran las preguntas que tienen un índice de facilidad incorrecto. Para que el facilitador sepa donde tiene que hacer cambios para mejorar la calidad del cuestionario.

Esta regla da una guía clara a los facilitadores e incluso a los diseñadores para entender el nivel al que están los estudiantes y la posible necesidad de adaptar los recursos del curso a las necesidades de los estudiantes.

Nueva regla: Los cuestionarios tienen un índice de de calificación aleatoria adecuado

El índice de calificación aleatoria es la media de calificación que el estudiante obtendría realizando la pregunta en cuestión aleatoriamente. Este índice solo esta disponible para preguntas de opción múltiple. El objetivo de esta estadística es que los profesores sepan cuál podría ser la calificación máxima de un estudiante sin que tenga conocimiento alguno

sobre la pregunta. Según la documentación oficial de Moodle [8] los valores correctos se encuentran por debajo del 40 % en cada pregunta.

Para implementar esta regla se lee el índice del archivo de reporte estadístico de cada cuestionario y se realiza una media con todas las preguntas contadas y se obtiene un porcentaje. Si este porcentaje supera un valor definido se considera que está mal planteado, por lo que sería necesario corregir algunas preguntas para considerarse correcto.

Nueva regla: Los cuestionarios tienen un índice de discriminación adecuado

El índice de discriminación relaciona la puntuación en una pregunta con las puntuaciones del resto de las preguntas. Este índice indica la efectividad de la pregunta para diferenciar a los estudiantes más hábiles en la materia de los que no lo son. Según la documentación oficial [8] una buena discriminación se encuentra por encima del 50 %.

Para implementar esta regla se accede nuevamente al fichero de estadísticas de cuestionarios y se lee el índice de cada pregunta, con los índices de todas las preguntas se hace una media y se comprueba que esa media este por encima de un valor definido. Si esta por debajo se considera que el cuestionario esta mal planteado y requiere revisión. Con el fin de localizar las preguntas que requieren revisión, se muestran las preguntas con sus respectivos índices.

Con este índice los facilitadores podrán identificar aquellas preguntas de los cuestionarios que no se adaptan correctamente al nivel de los estudiantes.

Diferencias entre versiones inferiores y superiores a 4.0 de Moodle

Como se ha mencionado anteriormente, para la implementación de la mayoría de los cuestionarios ha sido necesaria la lectura de un fichero de reporte estadístico mediante web scraping.

Sin embargo, uno de los problemas encontrado a la hora de implementar las funcionalidades ha sido que existen dos formatos para el fichero estadístico mencionado. En las versiones inferiores a la 4.0 de Moodle el fichero JSON está compuesto por arrays, sin claves que identifiquen que indica cada valor. A partir de la versión 4.0 el JSON tiene una mejor estructura con claves que identifican cada dato que se muestra.

Para la resolución de este conflicto ha sido necesario realizar un proceso intermedio en el que se identifica la versión del Moodle con el que se está trabajando, este dato se ha obtenido con la API de Moodle[23]. Conociendo la versión se realiza la operación de obtención de índices de una forma u otra.

Con esto se logra que los usuarios de cualquier versión puedan acceder a las funcionalidades implementadas, de esta forma se obtiene un mayor alcance de usuarios.

5.6. Implementación de funcionalidad de exportación de informe en Excel

Hasta el momento, en la aplicación, para visualizar un informe ha sido necesario conectarse, generar un informe y visualizarlo en el momento. No existía la capacidad de guardar el informe de ningún modo, lo que suponía una dificultad para el usuario hacer un seguimiento completo en el tiempo.

Para poder dar la opción de llevar un seguimiento de todos los informes generados existen dos opciones: que los informes sean mostrados por la aplicación o que el usuario pueda tener uno o varios archivos con los informes que desee. Para la primera opción sería necesario guardar los informes en una base de datos o en una carpeta con los informes guardados en archivos tipo JSON, Excel o incluso imagen. Sin embargo, esta opción aumentaría mucho la complejidad y el peso de la aplicación. Por lo que resulta más atractiva la segunda opción.

En virtud de lo cual, se optó por implementar la funcionalidad de exportación de informes a Excel formato ".xlsx". Para esta implementación, se ha diseñado una plantilla con las reglas y sus descripciones. A continuación, se ha implementado una función que rellena dicha plantilla con los datos del informe del curso en cuestión. Hay que puntualizar que solo se genera un Excel para cada curso, es decir, no se puede generar un Excel con los informes de varios cursos. Esta función es llamada desde la gateway que se utiliza desde la vista para obtener el Excel completo.

6. Trabajos relacionados

Este apartado sería parecido a un estado del arte de una tesis o tesina. En un trabajo final grado no parece obligada su presencia, aunque se puede dejar a juicio del tutor el incluir un pequeño resumen comentado de los trabajos y proyectos ya realizados en el campo del proyecto en curso.

7. Conclusiones y Líneas de trabajo futuras

Todo proyecto debe incluir las conclusiones que se derivan de su desarrollo. Éstas pueden ser de diferente índole, dependiendo de la tipología del proyecto, pero normalmente van a estar presentes un conjunto de conclusiones relacionadas con los resultados del proyecto y un conjunto de conclusiones técnicas. Además, resulta muy útil realizar un informe crítico indicando cómo se puede mejorar el proyecto, o cómo se puede continuar trabajando en la línea del proyecto realizado.

Bibliografía

- [1] 79+ staggering online learning statistics revealed!????(2024).
- [2] Angular.
- [3] Budgeting for software maintenance - baytech consulting.
- [4] Diferentes modelos de desarrollo de software: Una guía completa.
- [5] Docker: Accelerated container application development.
- [6] El patrón modelo-vista-controlador: Arquitectura y frameworks explicados.
- [7] En qué consiste la contenerización | ibm.
- [8] Estadísticas del reporte del examen - moodledocs. [Online; Accedido 12-Abril-2024].
- [9] Gestión de api - wikipedia, la enciclopedia libre.
- [10] InputStream (java platform se 8).
- [11] Interceptor microservice.
- [12] (pdf) quality reference framework (qrf) for the quality of moocs. [Online; Accedido 09-Abril-2024].
- [13] Qué es jsp.
- [14] React.
- [15] Read a file from resources in spring boot.

- [16] Robertoarastiblanco/elearningqa: Degree final project 2021-22 university of burgos. [Online; Accedido 10-Abril-2024].
- [17] Spring boot.
- [18] Spring framework.
- [19] Ubuvirtual - aula virtual de la universidad de burgos. [Online; Accedido 12-Abril-2024].
- [20] Using the spring resttemplate interceptor | baeldung. [Online; Accedido 17-Abril-2024].
- [21] Vue.js - the progressive javascript framework | vue.js.
- [22] Web scraping: ¿qué es y para qué se utiliza esta técnica? - ionos española.
- [23] Web service api functions - moodledocs. [Online; Accedido 09-Abril-2024].
- [24] ¿qué es el análisis de causa raíz? | kanban tool. [Online; Accedido 17-Abril-2024].
- [25] Julio Cabero-Almenara and Antonio Palacios-Rodríguez. The evaluation of virtual education: E-activities. *RIED-Revista Iberoamericana de Educacion a Distancia*, 24:169–188, 2 2021.
- [26] Alonso Lucía, Virginia Varela, Paragarino Rodés, Huertas Esther, Nativitat Hidalgo, and Lanzo Cabrera. Modelos de calidad del elearning.
- [27] Juan Mavo Navarro and Breeda Mcgrath. *Strategies for Effective Online Teaching and Learning: Practices and Techniques With a Proven Track of Success in Online Education*. 09 2021. [Online; Accedido 17-Abril-2024].