



UNIVERSIDAD
COMPLUTENSE
MADRID

Proyecto de Innovación

Convocatoria 2023/2024

Nº de proyecto 185

Maduración de la Corrección Automática Parcial de Actividades/Prácticas de los estudiantes

Responsable del proyecto: Iván García-Magariño García

Facultad de Informática

Departamento de Ingeniería del Software e Inteligencia Artificial

1. Objetivos propuestos en la presentación del proyecto

Los principales objetivos de este proyecto que se presentaron fueron los siguientes:

1. Realizar debate y recolectar opiniones entre profesores y estudiantes acerca de qué asignaturas y actividades serían adecuadas para la autoevaluación y corrección con soporte automatizado: Se planea hacer una automatización responsable, ética, aprobada por los actores principales y adecuada desde el punto de vista académica, basándose en las opiniones de profesores y estudiantes recogidos por medio de debates y encuestas. Esto persigue resolver el problema de usar la corrección automática sólo para aspectos que realmente se pueden evaluar de forma automática.
2. Proporcionar un mecanismo de autoevaluación a los estudiantes: De esta manera, se persigue que los estudiantes puedan obtener un mecanismo de evaluación inmediato para orientar su proceso de aprendizaje. Se busca centrarse en ejercicios y actividades prácticas de asignaturas que todavía no incorporan esa automatización y sea adecuado. Se persigue resolver el problema de la espera de los estudiantes en recibir su corrección que no le permite conocer sus errores de forma inmediata.
3. Facilitar la corrección por parte de los profesores: Se persigue que los profesores puedan disponer de herramientas en las que puedan formalizar sus prácticas y ejercicios, de tal forma que estas puedan indicar de forma automática si las soluciones prácticas funcionan correctamente en aquellas actividades y asignaturas que tengan sentido. Por tanto, este objetivo resuelve la sobrecarga de trabajo de los profesores en la corrección de aspectos de prácticas/actividades que se podrían automatizar.
4. Proporcionar la(s) herramienta(s) con ejemplos como un recurso educativo en abierto: Esto permitirá que otros profesores de otras universidades de diversos países puedan beneficiarse para hacer sus propios diseños. Se pretenda dar una difusión internacional por medio de al menos un artículo científico internacional y una web que describa y oriente el uso de esta herramienta. Este objetivo responde a la necesidad de disponer más herramientas de automatización de prácticas para dar soporte a la automatización de más tipos de ejercicios.

El PID del curso pasado tenía objetivos similares, y los cumplimos, automatizando prácticas de Fundamentos de Programación (FP) en tres grupos. Este proyecto se ha seguido con los mismos objetivos, pero usándolo en un dominio diferente, dado que se ha trabajado en los contextos de programación orientada a objetos en Tecnología de la Programación 2 (TP2) y en el desarrollo de aplicaciones web.

2. Objetivos alcanzados

Los principales objetivos alcanzados fueron los siguientes:

1. Realizar debate y recolectar opiniones entre profesores y estudiantes acerca de qué asignaturas y actividades serían adecuadas para la autoevaluación y corrección con soporte automatizado: Se recolectaron las opiniones de profesores acerca sobre qué otras asignaturas y contextos se podría aplicar además en FP. Se llegó a la conclusión que se podría aplicar para prácticas de programación orientada a objetos y desarrollo web.
2. Proporcionar un mecanismo de autoevaluación a los estudiantes: Se proporcionó un mecanismo de autoevaluación en TP2 y se presentó los resultados en un congreso (véase Anexo I).

3. Facilitar la corrección por parte de los profesores: Se facilitó la corrección a un profesor en TP2, y se trabajó sobre la posibilidad de aplicarlo en una asignatura sobre desarrollo de aplicaciones web a otro profesor.
4. Proporcionar la herramienta con ejemplos como un recurso educativo en abierto: Ahora mismo se dispone del código fuente de la herramienta en el repositorio de GitHub [ivangmg32/UnitJudge](https://github.com/ivangmg32/UnitJudge)¹ (todavía en privado por motivos de seguridad). Además, se ha publicado un nuevo artículo en este curso citado en el Anexo I.

3. Metodología empleada en el proyecto

En este proyecto, principalmente se ha explorado el uso del juez UnitJudge en diversas materias relacionadas con la programación y desarrollo de aplicaciones web, para valorar la utilidad en poder usarse en diversos escenarios. En concreto, en el artículo citado en Anexo I se puede ver las principales conclusiones sobre los usos de UnitJudge en TP2 y desarrollo de aplicaciones web.

Además, la metodología docente se ha centrado en un aprendizaje más autónomo por parte de los estudiantes por medio de unas actividades con corrección automática. Más concretamente, el juez en línea UnitJudge ha permitido que estudiantes de TP2 puedan probar sus ejercicios de forma autónoma. Además, se ha diseñado una manera en la que los estudiantes podrían autoevaluarse prácticas de desarrollo de aplicaciones web.

Por otra parte, la metodología docente ha buscado aumentar la productividad y calidad docente de los profesores, por medio de la descarga de trabajo en la comprobación de aspectos que se puedan automatizar. También es cierto que se requiere una inversión de tiempo para poder adaptar apropiadamente las prácticas y ejercicios para que puedan comprobarse por el juez. En el caso de desarrollo de aplicaciones web, todavía se requeriría que las prácticas cambiaran de estilo para que realmente pudieran ser autoevaluadas por los estudiantes. Dichas instrucciones deberían ser mucho más concretas en lo que se pide al estudiante, para que posteriormente se pudiera automatizar la corrección observando esos aspectos concretos.

4. Recursos humanos

En este proyecto docente, han trabajado todos los profesores indicados en la solicitud del mismo. La participación de los profesores ha sido clave para poder explorar otras áreas temáticas y asignaturas donde se ha podido aplicar el juez. En concreto, se ha observado que en programación orientada a objetos y en desarrollo de aplicaciones web, se ha podido encontrar una aplicación clara. Por el contrario, en otras asignaturas ha sido más difícil encontrar su aplicación, entre otros motivos por usar otro lenguaje de programación distinto a los soportados (e.g. Python) o por poder usar otro juez al tratarse de prácticas sencillas y cortas (e.g. en fundamentos de algoritmia).

5. Desarrollo de las actividades

En este apartado, se resumen los aspectos más relevantes que se concluyeron de cada tarea planeada en el proyecto dividido en fases:

FASE F1: Debate y recolección de opiniones de estudiantes y profesores

¹ <https://github.com/ivangmg32/UnitJudge> (última vez accedido el 22/05/2023) (por ahora está en privado y no pueden acceder los que no son colaboradores)

T1.1 Debate entre profesores sobre aspectos clave del proyecto y organización de dinámicas con estudiantes: Entre todos los profesores acordamos que TP2 y Aplicaciones Web (AW) serían las dos asignaturas que tendrían más sentido aplicar el juez.

T1.2 Debate entre algunos grupos de estudiantes: A los estudiantes se les preguntó cómo veían usar un juez para evaluar prácticas centradas en ser evaluadas por tests sobre funciones, y lo vieron adecuado.

T1.3 Elaboración de encuestas a los estudiantes: En este proyecto finalmente se decidió analizar los datos de uso del juez, en vez de preguntar a los estudiantes, evitando así sobrecargar a los estudiantes. El artículo citado en el Anexo I muestra cómo se evaluó el juez basándose en la ratio de estudiantes que fueron capaz de pasar el juez.

T1.4 Recogida y análisis de datos de las encuestas: Se usó la base de datos de UnitJudge para recoger los datos que posteriormente fueron analizados.

FASE F2: Preparación de automatización de prácticas

T2.1 Elección de actividades a automatizar sus prácticas: Se decidió automatizar varios ejercicios de TP2 y tomar como ejemplo una práctica de AW para diseñar una posible evaluación.

T2.2 Especificación de las características de sistemas para automatizar dichas prácticas: Para especificar los tests unitarios dentro de UnitJudge, se usó framework JUnit para Java y la librería de Selenium para acceder a páginas web de los estudiantes.

T2.3 Búsqueda de herramientas existentes para su automatización: En la búsqueda, se encontraron trabajos que usaban test unitarios para evaluación de prácticas, pero no se encontró jueces en línea basados en test unitarios en los que los estudiantes no necesitaran saber implementar test unitarios.

T2.4 Desarrollo de algoritmos para automatización de prácticas en las partes que no se puedan automatizar con las herramientas existentes: Se vio conveniente seguir usando el juez UnitJudge en línea que se desarrolló el curso pasado, para que pudiera probar distintas funciones dentro de una práctica sin tener que usar la entrada y salida global de toda la práctica

T2.5 Preparación de sistema o instrucciones para facilitar el uso por los profesores: Se establecieron una técnica de cómo usar UnitJudge en programación orientada objetos y AW, que se incluyó de forma resumida en el artículo del Anexo I.

FASE F3: Experimentación de la automatización de corrección de prácticas

T3.1 Preparación de prácticas autocorregibles siguiendo la aproximación de este proyecto: Se prepararon los tests y se probaron las soluciones de las prácticas sobre ejercicios de TP2.

T3.2 Realización de las prácticas en las asignaturas correspondientes: Se realizaron los ejercicios de TP2 y se planteó cómo se podría aplicar en AW.

T3.3 Preparación y realización de encuestas a los estudiantes: Finalmente, se realizaron análisis de los datos de uso en vez de encuestas.

T3.4 Análisis de los resultados obtenidos en los ejercicios: Finalmente, el 80% de los estudiantes que intentaron un ejercicio de TP2 acabaron consiguiendo pasar el juez.

FASE F4: Difusión de los resultados

T4.1 Difusión de los aspectos principales del proyecto y los resultados por medio de página web y foros: Por ahora, tenemos la página web de UnitJudge que hemos usado para los estudiantes y profesores. Se promocionó dicha herramienta entre los estudiantes por medio de publicación en foros de avisos de asignatura, los que enviaron notificaciones a los correos de los estudiantes.

T4.2 Escritura de artículo científico internacional sobre el proyecto y los resultados obtenidos para su publicación en congreso: Se publicó un artículo de cómo expandir el uso del juez en línea UnitJudge para ejercicios de programación orientada a objetos en java y en prácticas de aplicaciones web. Este artículo obtuvo el premio “Best presentation awards” y se cita en el Anexo I.

T4.3 Proporcionar la herramienta desarrollada para la corrección automática en abierto y ejemplos de recursos educativos realizados con esta herramienta: Además de la web y el repositorio de GitHub privado, ahora tenemos una técnica de cómo usar UnitJudge para evaluar prácticas de aplicaciones web.

Anexo I: Artículo presentado en congreso internacional

Artículo: García-Magariño, I., Bravo-Agapito, J., & López Fernández, M, (2024) Towards Expanding the Use of the Online Judge UnitJudge for Java Programming Exercises and Web Development Practices in Computer Science Education. In: ICELET 2024. XVIII International Conference on e-Learning and e-Teaching (Otorgado con **premio “Best presentation awards”** en dicho congreso)