

# Estrategias de control y seguimiento activo de proyectos de desarrollo de software

Miguel Matey Sanz, Carlos Granell Canut, Ramón Alberto Mollineda Cárdenas  
Departament de Llenguatges i Sistemes Informàtics  
Universitat Jaume I  
Castelló de la Plana  
{matey, canut, mollined}@uji.es

## Resumen

Se propone un sistema de acciones de control y seguimiento activo de proyectos de desarrollo de software por equipos. Ha sido concebido en el marco de una acción de coordinación de dos asignaturas del cuarto curso de un Grado en Ingeniería Informática, cuya piedra angular es un proyecto de desarrollo ágil dirigido por pruebas de aceptación y diseño evolutivo. Los estudiantes afrontan múltiples complejidades: escenario realista (p. ej. aplicación de movilidad), metodología de desarrollo (ATDD), requisitos técnicos (p. ej. pruebas ejecutables, arquitectura desacoplada), conexión a servicios públicos (p. ej. *geocoding*, cálculo de rutas) y uso de tecnologías modernas de desarrollo multiplataforma, integración y despliegue. A pesar de contar con recursos de dos asignaturas y un itinerario bien definido, el riesgo de abandono era significativo. Para reforzar los mecanismos de control, se creó un sistema con tarjetas de apercebimiento, dos sesiones dedicadas a la defensa de entregas parciales, y adaptaciones de la metodología *GitFlow* y de un estándar de *commits* (*Conventional commits*) para un seguimiento activo del proyecto y de la contribución individual de los estudiantes. Como resultado, se han observado mejoras en la tasa de presentados en primera convocatoria, en la calidad de los proyectos, y en el incremento en formación medido a partir de encuestas pre y post.

## Abstract

A system of active control and monitoring actions for software development projects by teams is proposed. It has been designed within the framework of a coordination action of two subjects from the fourth year of a Degree in Computer Engineering, the cornerstone of which is an agile development project guided by acceptance tests and evolutionary design. Students face multiple complexities: realistic scenario (e.g., mobility application), development methodology (ATDD), technical requirements (e.g., executable tests, decoupled architecture), connection to public services (e.g., *geocoding*, route calculation) and the use of modern

technologies for cross-platform development, integration and deployment. Despite having resources from two subjects and a well-defined itinerary, the risk of dropping out was significant. To reinforce the control mechanisms, a system was created with warning cards, two sessions dedicated to the defence of partial deliverable, and adaptations of the *GitFlow* methodology and a commit standard (*Conventional commits*) to actively monitor the project and the individual contribution of the students. As a result, improvements have been observed in the rate of projects completed in the first call, the project quality, and the increase in training measured from pre and post surveys.

## Palabras clave

Coordinación de asignaturas, proyecto de desarrollo de software, control y seguimiento de proyectos, *GitFlow*, *Conventional commits*.

## 1. Introducción

La formación universitaria convencional consta de un conjunto de asignaturas o materias, concebidas como partes de un todo (p. ej. grado, ingeniería, etc.), aunque estas suelen impartirse como dominios estancos, con ausencia o carencia de actividades prácticas concebidas para integrarlas.

Las oportunidades de integración de contenidos y competencias suelen aparecer en forma de estancias en práctica y de trabajos de fin de grado en la fase final de los estudios. Sin embargo, la cantidad y complejidad de dominios de conocimientos, y el tiempo transcurrido desde el momento en que mayoritariamente fueron impartidos hasta que pueden ser integrados, limita el potencial formativo de estos ejercicios.

La coordinación entre asignaturas surge como una estrategia efectiva para mitigar estas deficiencias estructurales, permitiendo a los estudiantes integrar teoría y práctica de materias afines desde etapas tempranas de la formación. Como se ha acreditado en trabajos previos [4, 5], estos esfuerzos contribuyen poderosa-

mente a la motivación de los estudiantes y a un aprendizaje más significativo.

El aprendizaje basado en proyectos es una de las prácticas de coordinación más populares y efectivas, porque establece un objetivo común a las asignaturas implicadas que solo puede lograrse con el concurso de las competencias de todas ellas.

El escenario más habitual suele ser un proyecto de una complejidad metodológica y técnica superior a experiencias previas, que requerirá por parte del alumnado, generalmente organizado en equipos, una dedicación regular durante varios meses. Los riesgos asociados son claros: dificultad para autoorganizarse, para asimilar e integrar nuevas tecnologías, para lograr dinámicas sostenibles de trabajo, etc.

Al margen de la consistencia de la propuesta, el éxito de esta práctica depende fundamentalmente de la efectividad de las estrategias de control y seguimiento del trabajo del alumnado. Estas deberían incluir pautas e hitos claros que fomenten desarrollos sostenibles, evaluación periódica del progreso, identificación temprana de errores y de problemas en la coordinación interna de los equipos, retroalimentación oportuna, e incentivos para aprender de errores, corregir desviaciones y revertir tendencias negativas.

Este trabajo es una evolución de un sistema de acciones de control y seguimiento de proyectos de desarrollo de software, en el marco de una experiencia de coordinación de dos asignaturas del cuarto curso de un Grado en Ingeniería Informática presentada en las JENUI 2022 [6], tras una experiencia inicial en las JENUI 2021 [5]. Esta se basa en la combinación del paradigma ágil Desarrollo Dirigido por Pruebas de Aceptación (ATDD, de *Acceptance Test Driven Development*) [7], y de estrategias de diseño evolutivo que dan soporte progresivo a necesidades de usuarios definidas mediante pruebas de aceptación ejecutables.

Esta acción de coordinación propone al alumnado el desarrollo de un proyecto de software realista que cubre objetivos formativos de ambas asignaturas siguiendo un itinerario único de sesiones prácticas, entregas y actividades de evaluación. No obstante, la dificultad de la propuesta solía provocar dos efectos opuestos. Había equipos que la entendían como una oportunidad para lograr un salto cualitativo en su formación y superar dos asignaturas concentrando esfuerzos, mientras otros eran incapaces de coordinarse y abandonaban en primera convocatoria.

Con el fin de mejorar los mecanismos de control y seguimiento del proyecto y del trabajo de los equipos, se incorporaron nuevas acciones como tarjetas de apercebimiento, sesiones dedicadas a la defensa de entregas parciales, y la aplicación de estándares de uso de Git (*GitFlow*) y de mensajes de confirmación de cambios (*Conventional commits*), con dos propósitos funda-

mentales: 1) ayudar a los equipos a organizar su trabajo interno, y 2) facilitar al profesorado un control detallado del progreso de los proyectos y de la contribución individual de cada estudiante en el trabajo de equipo.

Tras una implantación progresiva de estos cambios durante dos cursos, se han observado tasas elevadas de proyectos presentados en primera convocatoria, y mejoras en la calidad de los proyectos, y en la formación percibida a partir de encuestas pre y post.

El artículo está organizado como sigue: la sección 2 hace una revisión de trabajos relacionados, la sección 3 introduce la acción de coordinación, la sección 4 presenta la nueva estrategia de control y seguimiento, cuyos resultados son analizados en la sección 5. La sección 6 ofrece las conclusiones.

## 2. Trabajos relacionados

Como parte de las metodologías de aprendizaje basadas en proyectos (ABP), es crucial identificar riesgos potenciales, diseñar planes de prevención, y actuar oportunamente para que los resultados formativos sean satisfactorios. Esto cobra mayor importancia cuando el ABP se realiza en grupo.

Estas necesidades han sido constatadas en estudios previos. En [3] se identifican algunos riesgos potenciales de proyectos realizados en grupo, entre ellos, comunicación deficiente y mala planificación de tareas. Para la prevención, los autores proponen establecer desde un inicio reglas claras sobre mecánicas de grupo (p. ej. roles, disponibilidad, etc.) y el uso de herramientas de planificación de proyectos. Como parte del seguimiento, proponen informes semanales con el progreso del proyecto en relación a la planificación. Berná et al. [2] proponen un marco de coevaluación donde los estudiantes se evalúan entre ellos cada dos semanas durante el desarrollo del proyecto. Estas evaluaciones permiten al profesorado analizar el rendimiento de los estudiantes, pudiendo identificar aquellos con valoraciones bajas y actuar en caso de ser necesario.

Las propuestas anteriores se basan en informes que los estudiantes deben realizar en paralelo al desarrollo del proyecto, lo cual, además de su naturaleza subjetiva, podría distraerlos del objetivo principal. Estas estrategias desaprovechan el potencial de los sistemas de control de versiones (VCS, de *Version Control System*) como Git<sup>1</sup>, conjuntamente con servicios de hospedaje de repositorios en la nube como GitHub<sup>2</sup>, los cuales, además de ser potentes herramientas colaborativas de versionado de código necesarias en la formación de los estudiantes [9], pueden emplearse por parte del profesorado para el seguimiento de los proyectos.

El uso de un VCS como herramienta docente de control de proyectos no es novedoso. Andrade et al. [1] emplearon un VCS para monitorizar la evolución de

<sup>1</sup> <https://git-scm.com>

<sup>2</sup> <https://github.com>

los estudiantes a lo largo de las sesiones de prácticas, donde los cambios en los repositorios locales de los estudiantes se almacenaban automáticamente cada cierta frecuencia o de forma manual. Riesco *et al.* [8] también utilizaron un VCS con el objetivo de evaluar versiones intermedias del proyecto final para detectar posibles errores o estudiantes que colaborasen menos de lo adecuado. Más recientemente, Volkova *et al.* [11] presentaron un cuadro de mando integral para realizar el seguimiento de sus estudiantes mediante datos obtenidos de GitHub y Taiga<sup>3</sup>, generando métricas de interés como los porcentajes de líneas de código o tareas asignadas a cada estudiante.

A diferencia de las experiencias revisadas, en este trabajo se propone el uso de un VCS a través de buenas prácticas mediante los estándares de *GitFlow*<sup>4</sup> y *Conventional commits*<sup>5</sup> adaptados a los requisitos de las asignaturas coordinadas, para realizar un seguimiento de la participación de los estudiantes tanto en términos del volumen de trabajo individual como del grado de ajuste a la metodología de desarrollo ATDD.

### 3. Contexto: coordinación de asignaturas basada en proyecto

En el curso 2021/2022, pusimos en marcha una iniciativa de coordinación de dos asignaturas [6] basada en un proyecto común y en un sistema de evaluación unificada del proyecto. Estas asignaturas son Diseño de Software y Paradigmas de Software del itinerario de Ingeniería de Software del Grado en Ingeniería Informática de la Universitat Jaume I, las cuales se imparten en el primer cuatrimestre del cuarto curso. Mientras la primera cubre el estudio de patrones de diseño, la segunda introduce el paradigma ATDD. La acción de coordinación aprovecha el rol del diseño evolutivo en la metodología ATDD, la cual promueve soluciones que progresan en paralelo a las necesidades funcionales especificadas mediante pruebas de aceptación.

La propuesta técnica consistió en el desarrollo de una aplicación de movilidad para el cálculo y visualización de rutas entre dos lugares de interés, incluyendo trayecto, duración y coste estimado. La aplicación debía conectarse a servicios públicos abiertos (*free open API*) de *geocoding* y cálculo de rutas.

A continuación, se describe un ejemplo de la sinergia entre los objetivos formativos de ambas asignaturas. Una de las historias de usuario consistía en el cálculo del coste asociado de una ruta con un vehículo. Siguiendo ATDD, los escenarios y las pruebas fueron definidas (antes del código) de acuerdo al patrón *Given* (estado inicial), *When* (datos de prueba y ejecución de prueba), *Then* (resultado esperado y estado final). Una

prueba vinculada a la historia antes descrita debía validar el cálculo del coste asociado a la ruta considerando una distancia (p. ej. 100 km), un vehículo (p. ej. diésel con consumo de 5 L/100), y el precio actual de un litro de diésel obtenido en tiempo real de una API pública de precios de carburantes. Dada la imposibilidad de predecir el precio en tiempo real, la validación se planteó en términos de un rango verosímil. Una vez implementada y ejecutada la prueba contra un método sin lógica, y comprobado el fallo de la prueba, se debía extender el diseño de cada componente de software necesario para pasar la prueba. Estos incluían clases para interactuar con la API y recuperar el dato de interés (precio del litro de diésel) del fichero de intercambio de datos recibido desde la API.

Actualmente, el proyecto representa el 70 % de la nota en ambas asignaturas. El 30 % restante se obtiene a través de seminarios en Paradigmas de Software, y de seminarios (20 %) y actividades previas a las clases de teoría (10 %) en el caso de Diseño de Software.

#### 3.1. Mecanismos iniciales de coordinación y control

La primera edición de esta iniciativa en 2021/2022 incorporó una serie de mecanismos de coordinación, relacionados a continuación, que han servido de base para el despliegue progresivo de nuevas acciones durante 2022/2023 y 2023/2024 (foco de este trabajo).

Entre los mecanismos de coordinación originales podemos destacar un proyecto único a partir de los objetivos formativos de ambas asignaturas, la integración de laboratorio con profesorado común, un itinerario compartido con hitos y entregas parciales comunes, nota de proyecto compartida, sesión conjunta de defensa de proyectos frente al profesorado de ambas asignaturas, coordinación de sesiones de seminarios evitando actividades redundantes, incluyendo una sesión conjunta dedicada a defender el patrón de arquitectura elegido, y canales de comunicación periódica entre el profesorado de ambas asignaturas.

El control se limitaba a dos entregas parciales evaluadas con nota numérica por el profesorado de prácticas, para proporcionar retroalimentación inmediata. El seguimiento basado en dos entregas parciales se considera un mecanismo pasivo, porque deja a la iniciativa de los alumnos las acciones de corrección y mejora. Así, no existía forma de conocer el nivel de comprensión de las deficiencias señaladas, ni el valor de las acciones correctivas, si es que existían.

### 4. Mecanismos de control activo

Para superar las limitaciones anteriores se adoptaron algunas acciones menores de alineación entre ambas

<sup>3</sup> <https://taiga.io/es>

<sup>4</sup> <https://nvie.com/posts/a-successful-git-branching-model/>

<sup>5</sup> <https://www.conventionalcommits.org/en/v1.0.0/>

asignaturas (p. ej. mismo peso del proyecto en la nota final), conjuntamente con nuevos mecanismos de control activo que incluyen, por un lado, mejoras en la metodología docente de seguimiento y evaluación del proyecto y, por otro lado, el uso de metodologías de desarrollo de software colaborativo habitualmente utilizadas en entornos profesionales.

#### 4.1. Seguimiento y evaluación formativa

En cursos anteriores, los equipos de estudiantes debían realizar dos entregas parciales en las que debían especificar aspectos como el alcance funcional del proyecto, la planificación temporal inicial, el código de pruebas de aceptación ejecutables, etc. Ambas entregas eran evaluadas y se les asignaba una nota como parte de la evaluación continua (2 puntos). Esto tenía dos efectos negativos: 1) errores en etapas tempranas del desarrollo (algo muy común) tenían impacto en la nota final, y 2) se daba pie al conformismo y a una falsa sensación de seguridad.

Para mitigar estos efectos, se ha migrado de una evaluación sumativa a una evaluación formativa durante el período de desarrollo del proyecto, donde el resultado principal de las entregas parciales no es una nota numérica, sino una retroalimentación que los estudiantes deben aprovechar para mejorar su proyecto. De esta forma, los errores tempranos no solo no penalizan la nota final, sino que pueden ser corregidos a tiempo, ofreciendo a los estudiantes oportunidad y motivación para aprender. Junto con la retroalimentación, se implementa un sistema de tarjetas de apercebimiento, de forma que las entregas se califican en una de las siguientes categorías:

- *Superada*: la entrega cumple especificaciones técnicas y criterios de calidad. Los cambios y mejoras indicadas en la retroalimentación, si existen, son menores y deberán ser abordados para continuar con el desarrollo del proyecto.
- *Tarjeta amarilla*: la entrega presenta deficiencias relevantes que deben corregirse de inmediato. Los cambios y mejoras indicados deberán ser abordados, y entregar una nueva versión para su evaluación en el plazo de una semana. La nueva entrega vuelve a ser calificada y recibe nuevo informe de retroalimentación. Si las deficiencias han sido corregidas, la entrega se marca como superada; en caso contrario, la entrega mantiene la *tarjeta amarilla*.
- *Tarjeta roja*: la nueva entrega reitera deficiencias relevantes ya apercebidas con una tarjeta amarilla en la entrega anterior, lo cual indica que la dinámica de progreso del proyecto es negativa. Esta calificación conlleva el suspenso automático del proyecto. Cabe resaltar que no es posible obtener una *tarjeta roja* directa (requiere de una *tarjeta amarilla* previa).

Este sistema de calificación permite a los equipos ser conscientes del valor de las entregas parciales, a la vez que les otorga margen para solucionar las deficiencias sin sufrir penalización, al poder eliminar las tarjetas amarillas si se demuestra que las deficiencias se han resuelto satisfactoriamente. El sistema también fomenta que los estudiantes progresen y continúen focalizados en el objetivo principal (aplicar cambios y mejoras señalados) sin tener que estar pendientes de una nota numérica, como podría ocurrir con un sistema de calificación tradicional, lo que podría conducir a conformismo y desmotivación.

Adicionalmente a la evaluación formativa de las entregas, ambas asignaturas se coordinan en dos sesiones conjuntas donde los equipos defienden las entregas. Ambas actividades se organizan como sigue:

- *1.ª sesión* (tras la primera entrega parcial). Los alumnos justifican las tecnologías seleccionadas para el desarrollo, presentan un prototipo exploratorio con dichas tecnologías (*spike*), un breve análisis de los posibles riesgos que podrían suceder a lo largo del proyecto (medio plazo), así como de los inmediatos que podrían comprometer la siguiente entrega parcial (corto plazo), acompañado de acciones de mitigación de los riesgos identificados.
- *2.ª sesión* (tras la segunda entrega parcial). Los alumnos presentan una planificación inicial en iteraciones, la implementación de las pruebas de aceptación de las historias de usuario de la primera iteración, la justificación de los patrones de diseño seleccionados, una demo del estado actual del proyecto, y las contribuciones de cada estudiante dentro del grupo hasta el momento.

Al ser programados justo una semana después de las entregas, ambas sesiones de presentación y defensa de los proyectos permiten un seguimiento tanto a nivel individual como de equipo, constituyendo una acción de refuerzo de los mecanismos de control. Esta decisión asume que la obligación de participar en una actividad presencial de defensa del trabajo del equipo, exige un mayor compromiso individual que presentar un informe escrito a título colectivo. Además, la participación en estas sesiones se puntúa dentro de la evaluación continua, atendiendo a criterios relacionados con competencias transversales como el análisis crítico, la toma de decisiones, la resolución de problemas, la comunicación de ideas y el trabajo de equipo.

Por otra parte, el carácter grupal de estas sesiones permite a cada equipo conocer el progreso de otros equipos y compararse con ellos, ofreciendo oportunidades de identificar, por ejemplo, aspectos interesantes e inspiradores, así como deficiencias a evitar. Esta práctica puede incentivar la competitividad y la motivación entre los equipos, ya que los estudiantes descubren conocimientos y experiencias prácticas que podrían hacerlos más productivos en el desarrollo de sus

proyectos, de forma similar al aprendizaje mediante la contribución a proyectos de software existentes [10].

## 4.2. Metodologías de desarrollo colaborativo

En un contexto del desarrollo de software, el uso de un VCS es una práctica casi obligatoria, más cuando el desarrollo involucra a varias personas. Estos sistemas se complementan con servicios de hospedaje de repositorios. Esta dupla, además de mantener un registro de cambios, permite a cada persona trabajar de forma independiente y en paralelo en entornos locales, y luego integrar sus avances en entornos centralizados.

El uso de estas herramientas suele dar lugar a problemas de integración de cambios de los desarrollos locales (*merge conflicts*), lo que puede derivar en un auténtico caos.

Para prevenir y reducir estos problemas, se han diseñado normas que promueven un uso ordenado de un VCS. Por ejemplo, una metodología conocida en entornos profesionales es *GitFlow*, la cual define cómo emplear las ramas en un repositorio Git: ramas principal, de despliegues, de revisión, de funcionalidad, etc.

Dado que *GitFlow* es una metodología que cubre caústicas propias de escenarios profesionales, se propone una versión adaptada a las asignaturas coordinadas que contempla el uso de tres tipos de ramas:

- *Rama principal*: contiene versiones estables.
- *Rama de iteración*: cada iteración del desarrollo emplea una rama. Cuando la funcionalidad de la iteración está completa, la rama se fusiona con la principal, dando lugar a una versión estable.
- *Rama de historia*: cada historia de usuario, dentro de una iteración, crea una rama. Cuando la funcionalidad de la historia está completa, la rama se fusiona con la rama de su iteración.

Esta metodología permite que el equipo pueda trabajar paralelamente en varias historias minimizando el riesgo de conflictos durante el desarrollo. La figura 1 muestra un ejemplo de uso de esta metodología. A los estudiantes se les propuso el uso de Git como VCS y GitHub como servicio de hospedaje.

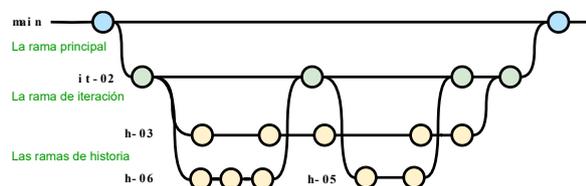


Figura 1: Ejemplo de uso de la adaptación de la metodología *GitFlow* para repositorios Git.

Otro aspecto clave en el uso de un VCS son los mensajes de confirmación de cambios (*commits*), los cuales permiten describir los cambios que se introducen en el repositorio, constituyendo así un “diario de desarrollo” que facilita el seguimiento del proyecto.

En el desarrollo colaborativo es importante el uso de un lenguaje común para que las descripciones de los cambios mantengan una estructura común. Una notación ampliamente usada es la de *Conventional commits*, la cual define cómo redactar mensajes para indicar el tipo de cambio o el ámbito, entre otros detalles.

En nuestro caso, adaptamos dicha especificación a nuestras necesidades, instando a los estudiantes a redactar sus mensajes de confirmación con el formato “<tipo>( <ámbito>): <descripción>”, donde el significado de cada elemento es el siguiente:

- <tipo>: indica el tipo de cambio descrito por el mensaje, por ejemplo, *test* (pruebas), *feat* (funcionalidades), *fix* (solución de errores), *refactor* (optimización de código), etc.
- <ámbito>: indica en qué iteración e historia se producen los cambios.
- <descripción>: breve descripción del cambio.

Un ejemplo de mensaje de confirmación de cambios que contiene pruebas de aceptación de la historia de usuario “h02” en la iteración “it01” podría ser:

<test>(it01h02): tests de aceptación

El uso conjunto de *GitFlow* y de normas para la escritura de mensajes permite a los estudiantes familiarizarse con estándares del desarrollo de software, mientras facilita al profesorado monitorizar el progreso de cada equipo y, sobre todo, verificar el uso correcto de la metodología ATDD. Así, sería posible identificar y evaluar versiones con pruebas de aceptación ejecutables antes del código objeto de evaluación.

## 5. Resultados

La evaluación de los mecanismos de control se realiza en tres áreas de análisis. En primer lugar, se comparan la tasa de proyectos presentados en la primera convocatoria, la ratio de aprobados y la nota media del proyecto con los índices de cursos anteriores. En otro análisis, se mide el incremento de aprendizaje de los estudiantes mediante una encuesta de dominio de competencias y herramientas. Finalmente, se presentan los resultados de una encuesta de valoración de la experiencia del alumnado con el proyecto y las asignaturas.

Los datos y el código usados en los análisis están disponibles en un repositorio público en GitHub<sup>6</sup>.

### 5.1. Análisis de las acciones de control

La figura 2 ilustra la trayectoria de los 8 equipos del curso 2023/2024 a través de las entregas parciales. La

<sup>6</sup> <https://github.com/matey97/jenui2024>

parte superior de la columna “1.ª entrega” muestra las marcas de colores de tres equipos (3, 5 y 6) que superaron el ejercicio directamente, mientras en su parte inferior se ordenan las marcas de cinco equipos (1, 2, 4, 7 y 8) que recibieron una *tarjeta amarilla*, aunque luego introdujeron las mejoras necesarias para superar la entrega. La “2.ª entrega” fue superada de forma directa por cinco equipos representados en su parte superior (2, 3, 4, 6 y 8), dos más que en la primera entrega, mientras que tres equipos recibieron apercibimientos (1, 5 y 7). De ellos, solo el 5 y el 7 superaron finalmente la segunda entrega.

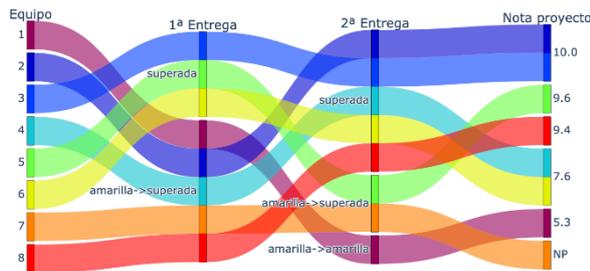


Figura 2: Resultados de entregas parciales y notas finales del proyecto. Las columnas centrales ilustran las entregas parciales. La parte superior agrupa marcas con el color de los equipos que superaron las entregas de forma directa. En la parte inferior, marcas con el color de los equipos que fueron apercibidos.

Destacan los casos de los equipos 2, 5 y 8, los que, pese a recibir una *tarjeta amarilla*, obtuvieron muy buenas calificaciones finales fruto del carácter formativo de los mecanismos de control activo y de retroalimentación inmediata. Por otra parte, los grupos 1 y 7 obtuvieron *tarjeta amarilla* en ambas entregas, lo que acabó reflejándose en el resultado final del proyecto.

El cuadro 1 resume los datos de estudiantes matriculados, presentados, aprobados en la primera convocatoria de 2021/2022, 2022/2023 y 2023/2024, así como la nota media de los proyectos en cada caso.

	2021/2022	2022/2023	2023/2024
Matriculados	27	22	27
Presentados	25 (92,6 %)	17 (77,3 %)	23 (85,2 %)
Aprobados	19 (76,0 %)	11 (64,7 %)	23 (100 %)
Nota media	7,02	5,24	8,48

Cuadro 1: Estadísticas por curso.

La tasa de presentados en 2023/2024 es del 85,2 %, superior a la del curso anterior (77,3 %) aunque inferior a la de 2021/2022 (92,6 %). Sin embargo, la tasa de aprobados (sobre presentados) del curso actual del 100% es muy superior al 64,7% y al 76% de los cursos

previos. Además, se observan diferencias entre las notas medias, también a favor del curso actual.

La comparación de las tasas de aprobados sugiere un efecto muy positivo de las acciones de control en la calidad de los proyectos. Sin embargo, no se observa una mejora en la tasa de alumnos presentados en primera convocatoria del curso 2023/2024, a pesar del bajo número de estudiantes no presentados (4), cantidad que suele depender de circunstancias específicas de cada curso. Este margen suele constar de uno o dos equipos con miembros con vínculo laboral, cuya escasa dedicación termina lastrando al equipo y comprometiendo la viabilidad del proyecto. Estos equipos atípicos podrían considerarse muestras ruidosas, pues la asignatura está concebida para asistencia regular.

### 5.2. Medición del incremento de la formación del estudiantado

Al iniciar y finalizar las asignaturas, una mayoría de estudiantes (18/27) respondieron una encuesta donde se les preguntaba por el grado de conocimiento autopercibido (1 - “No lo conozco”, 5 - “Lo domino”) de una serie de técnicas, herramientas y metodologías relacionadas con las asignaturas. Estas respuestas se emplearon para calcular el incremento en el dominio de aquellos recursos que los estudiantes indicaron que habían utilizado en el desarrollo del proyecto. Este incremento fue medido promediando las diferencias entre las encuestas final e inicial, en una escala de 1 a 5.

La figura 3 muestra incrementos generalizados en todas las técnicas y herramientas, pero destacan con más de dos puntos absolutos de ganancia el empleo de servicios *serverless* (2,61), pruebas de aceptación (2,53), desarrollo web moderno (2,50), consumo de APIs de terceros (2,41), convenciones de mensajes de confirmación (2,22), estándares para el uso de Git (2,13) y despliegue de servidores en la nube (2).

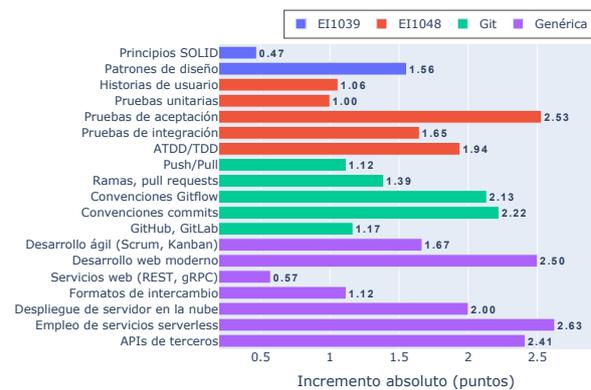


Figura 3: Incremento de formación en técnicas y herramientas empleadas en el proyecto.

Estos resultados indican la adquisición eficaz de las competencias específicas de las asignaturas (pruebas

de aceptación), de competencias tecnológicas (*serverless*, desarrollo web moderno, etc.) y de competencias metodológicas (convenciones en *commits* y Git).

También se observan incrementos notables (entre 1 y 2 puntos) en otras competencias específicas: ATDD (1,94), pruebas de integración (1,65) y patrones de diseño (1,56). Por otro lado, las técnicas/herramientas con incrementos más modestos (< 1 punto) son el uso de los principios SOLID (0,47) y el desarrollo de servicios web (0,57), lo cual puede deberse a un conocimiento adquirido previamente (SOLID) y a la aplicación residual en el proyecto (servicios web).

La figura 4 compara, únicamente para las técnicas y herramientas comunes, los incrementos obtenidos en el curso actual 2023/2024 respecto al curso 2021/2022 [6], último período del que se tienen mediciones, cuando no existían los mecanismos de control y seguimiento introducidos en este trabajo.

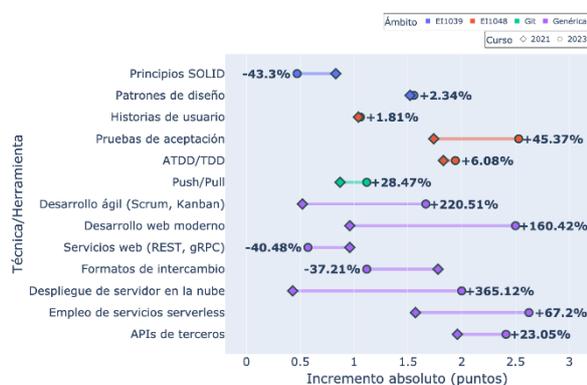


Figura 4: Comparativa del incremento de formación reportados en los cursos 2021/2022 y 2023/2024.

Los incrementos reportados en el curso actual son generalmente superiores a los del curso 2021/2022, a excepción del uso de principios SOLID (-43,30%), desarrollo de servicios web (-40,48%) y uso de formatos de intercambio (-37,21%). Se observan incrementos llamativos en competencias tecnológicas y metodológicas, como el despliegue de servidores en la nube (365,12%), el desarrollo ágil (220,51%) y el desarrollo web moderno (160,42%). Sin embargo, no se observa un incremento destacable en la adquisición de las competencias específicas, a excepción del aumento de las pruebas de aceptación (45,37%) y la disminución ya mencionada en el uso de principios SOLID.

En resumen, los resultados cuantitativos expuestos anteriormente parecen avalar los mecanismos implantados en este curso para el control y seguimiento de proyectos, ya que el incremento en la adquisición de competencias es generalizado, y la tasa de aprobados supera ampliamente a las de ediciones anteriores, manteniendo los mismos niveles de calidad y exigencia de los proyectos en los cursos analizados.

### 5.3. Evaluación de la experiencia

Tras finalizar la asignatura, más de la mitad de los estudiantes (14/27) respondieron a una encuesta para valorar su experiencia durante la realización del proyecto conjunto. Esta encuesta, presentada en [5] y también utilizada en [6], contiene 23 ítems relacionados con el perfil formativo del proyecto, experiencias de diseño y experiencias con ATDD. Además, en esta edición, se han añadido cinco nuevos ítems relacionados con el trabajo colaborativo:

24. La carga de trabajo se ha repartido equitativamente entre los miembros del equipo.
25. La comunicación entre los miembros del equipo ha sido efectiva.
26. No han surgido problemas entre los miembros del equipo.
27. El uso de la metodología propuesta para el uso de Git ha sido de utilidad y ha permitido trabajar paralelamente.
28. El uso de la metodología propuesta en la escritura de *commits* ha sido de utilidad.

La figura 5 recoge los resultados de las encuestas del presente curso y del 2021/2022 (últimos históricos disponibles), donde cada barra indica el valor medio de cada ítem. Como se observa, las puntuaciones medias de las cuestiones son muy similares entre cursos. Las mayores diferencias se aprecian en los ítems 5 (estimulante tener libertad de elección de tecnologías) y 8 (uso no forzado de patrones de diseño) a favor del curso 2021/2022, y en los ítems 13 (diseñar clases dependientes de abstracciones) y 19 (escribir pruebas facilita la colaboración) a favor del presente curso. También se han calculado medias de los ítems de cada ámbito: la valoración del perfil formativo del proyecto ha disminuido de 4,66 a 4,51, mientras que ha aumentado en experiencias de diseño (3,87 a 3,92) y ATDD (4,02 a 4,12). Estas puntuaciones indican que los estudiantes valoran la acción de coordinación.



Figura 5: Puntuación media por ítems.

En cuanto a los ítems añadidos en esta edición, destacan el 27 y 28, con medias de 4,35 y 4,64 respectivamente, lo que indica que los estudiantes valoran el uso de las metodologías de desarrollo colaborativo.

Por otra parte, la similitud de las valoraciones de los ítems comunes entre cursos diferentes ofrece garantías

razonables sobre la consistencia de la encuesta y el valor de las respuestas como criterio diagnóstico para introducir mejoras docentes.

Como parte de la encuesta, los estudiantes también tenían la posibilidad de expresar sus comentarios y opiniones. Los comentarios positivos se centraron en el realismo del proyecto propuesto y la libertad en el uso de tecnologías. Como comentarios negativos, algunos estudiantes indicaron que les habría gustado tener más tiempo para desarrollar el proyecto.

## 6. Conclusiones

Este trabajo presenta una evolución de una acción de coordinación de dos asignaturas de un Grado en Ingeniería Informática, a través de un sistema de acciones de control y seguimiento activo de proyectos de software concebido para mitigar riesgos de desmotivación y abandono. La propuesta introduce cambios en la metodología docente mediante la implantación de la evaluación formativa durante el desarrollo del proyecto, y adapta tecnologías y estándares de desarrollo colaborativo que promueven un uso ordenado de un VCS y un control preciso del trabajo individual. Resultados basados en la comparación de las notas medias de los proyectos con las de cursos anteriores sugieren un efecto positivo de las acciones implementadas, mientras se mantiene una tasa elevada de proyectos presentados. El valor formativo de la experiencia quedó también acreditado mediante dos encuestas que cubren tanto tecnologías como metodologías involucradas en el proyecto de software.

## Referencias

- [1] Diego Andrade, Emilio J. Padrón y Basilio B. Fraguera. Monitorización del trabajo en prácticas usando un sistema de control de versiones. En *Actas de las XVI Jornadas sobre la Enseñanza Universitaria de la Informática*, pp. 507-510, Santiago de Compostela, 2010.
- [2] José Vicente Berná Martínez, Carlos José Villagrà Arnedo, Rafael Molina Carmona, Faraón Llorens Largo, Francisco José Gallego Durán, Sergio Javier Viudes Carbonell, Pedro José Ponde de León Amador, Mireia Luisa Sempere Tortosa, María Dolores Sáez Fernández, María Pilar Escobar Esteban, Juan Antonio Gil Martínez-Abarca, David Gil Méndez, Antonio Ríos Vila, Javier Ortiz Zamora y Miguel Ángel Lozano Ortega. ¿La evaluación evoluciona? Una experiencia de coevaluación en ABP. En *Actas de las XXVIII Jornadas sobre la Enseñanza Universitaria de la Informática*, pp. 263-270, A Coruña, 2022.
- [3] Óscar Cánovas Reverte y Félix Jesús García Clemente. Prevención y seguimiento de factores limitantes del trabajo en equipo en experiencias ABP. En *Actas de las XXII Jornadas sobre la Enseñanza Universitaria de la Informática*, pp. 73-80, Almería, 2016.
- [4] María Ferré, Carlos García-Barroso, Montse García-Famoso, David Sánchez y Aida Valls. Mejora de la formación en el diseño y desarrollo de software a partir de la coordinación de distintas asignaturas. En *Actas de las XXV Jornadas sobre la Enseñanza Universitaria de Informática*, pp. 23-30, Murcia, 2019.
- [5] Alberto González Pérez, Ramón A. Mollineda Cárdenas, David Llorens Piñana. Aprendizaje basado en metodologías ágiles centradas en diseño evolutivo dirigido por pruebas de aceptación. En *Actas de las XXVII Jornadas sobre la Enseñanza Universitaria de Informática*, pp. 99-106, Valencia, 2021.
- [6] Alberto González Pérez, Carlos Granell Canut, y Ramón A. Mollineda Cárdenas. Coordinación de asignaturas dirigida por un proyecto de desarrollo ágil con evaluación unificada. En *Actas de las XXVIII Jornadas sobre la Enseñanza Universitaria de la Informática*, pp. 127-134, La Coruña, 2022.
- [7] Ken Pugh. *Lean-Agile Acceptance Test-Driven Development: Better Software Through Collaboration*. Addison-Wesley, 2011.
- [8] Miguel Riesco Albizu y M. Ángeles Díaz Fondón. Uso de sistemas de control de versiones en el seguimiento continuo del trabajo del alumno. En *Actas de las XVI Jornadas sobre la Enseñanza Universitaria de la Informática*, pp. 527-530, Santiago de Compostela, 2010.
- [9] Fran J. Ruiz-Bertol y Francisco Javier Zaragoza-Soria. El Control de Versiones en el aprendizaje de la Ingeniería Informática: Un enfoque práctico. En *Actas de las XIII Jornadas sobre la Enseñanza Universitaria de la Informática*, pp. 447-454, Teruel, 2007.
- [10] Diomidis Spinellis. Why computing students should contribute to open source software projects. *Communications of the ACM*, vol. 64, no. 7, pp. 36-38, 2021.
- [11] Alejandra Volkova, Carles Farré, Xavier Franch, José Daniel Conejos y Aleix Linares. Implantación de un cuadro de mando para el seguimiento del progreso de proyectos software desarrollados por equipos de estudiantes. En *Actas de las XXVIII Jornadas sobre la Enseñanza Universitaria de la Informática*, pp. 383-386, A Coruña, 2022.

# Estudio sobre la actitud hacia la Inteligencia Artificial en la educación preuniversitaria

Rafael Herrero-Álvarez, Coromoto León, Gara Miranda  
Luz Marina Moreno de Antonio, Casiano Rodríguez León

Departamento de Ingeniería Informática y de Sistemas

Universidad de La Laguna

38270 San Cristóbal de La Laguna

rherrero@ull.edu.es, cleon@ull.edu.es, gmiranda@ull.edu.es,

lmmoreno@ull.edu.es, crguezl@ull.edu.es

## Resumen

A pesar del creciente uso de herramientas basadas en Inteligencia Artificial, a menudo se pasa por alto la comprensión de sus conceptos subyacentes, centrandose la atención exclusivamente en el uso. En este trabajo se aborda un estudio de la percepción que tienen los estudiantes preuniversitarios antes y después de realizar una actividad enfocada a la creación de modelos, el entrenamiento supervisado y la problemática de la existencia de sesgos en estos sistemas, haciendo uso del instrumento denominado 'Actitud hacia la Inteligencia Artificial'. La actividad se llevó a cabo en el marco de la Olimpiada de Pensamiento Computacional, un evento cuyo objetivo es mejorar la percepción de las Ciencias de la Computación entre los más jóvenes. Participaron un total de 82 estudiantes de Educación Secundaria, entre 13 y 16 años, de los cuales 38 eran chicas, 40 chicos y 4 no binarios. Mientras que los chicos no creen que la Inteligencia Artificial pueda destruir a la humanidad sino que es posible que traiga beneficios, las chicas muestran una mayor desconfianza. El análisis revela un cambio en la percepción hacia la Inteligencia Artificial, tendiendo a ser más crítico y negativo tras realizar la actividad.

## Abstract

Despite the growing use of tools based on Artificial Intelligence, there is often an oversight in understanding its underlying concepts, focusing exclusively on its use. This work addresses a study of the perception pre-university students have before and after conducting an activity in which concepts such as model creation, supervised training, or the existence of biases in these systems were explored using the Attitude Towards Artificial Intelligence instrument. The activity was carried out within the framework of the Computational

Thinking Olympiad, an event aimed at improving the perception of Computer Sciences among youngsters. A total of 82 secondary education students, aged between 13 and 16, participated in the study, of which 38 were girls, 40 boys, and 4 non-binary. The results show a greater distrust towards Artificial Intelligence among girls compared to boys, while the latter do not believe it can destroy humanity and also think that it could bring benefits. The analysis reveals a change in perception towards Artificial Intelligence, tending to be more critical and negative after conducting the activity.

## Palabras clave

Inteligencia artificial, estudios preuniversitarios, Educación Secundaria, programación.

## 1. Introducción

La Inteligencia Artificial (IA), entendiéndose esta como un sistema basado en máquinas diseñado para funcionar con diversos niveles de autonomía y que puede mostrar capacidad de adaptación tras su despliegue y que, para objetivos explícitos o implícitos, infiere, a partir de la entrada que recibe, cómo generar salidas tales como predicciones, contenidos, recomendaciones o decisiones que pueden influir en entornos físicos o virtuales [1], ha experimentado un enorme crecimiento en los últimos años, especialmente con el desarrollo de herramientas de Procesamiento de Lenguaje Natural (NLP) o de IA generativa como ChatGPT [10]. La IA está revolucionando y transformando la estructura laboral y económica, planteando desafíos como la pérdida de empleo, o la necesidad de reentrenamiento continuo, haciendo necesario el planteamiento de nuevos modelos económicos y sociales, impactando especialmente en el ámbito de la medicina o el transporte [4].

Algunos autores consideran que la implementación de IA en el ámbito de la educación cambiará y mejorará los sistemas educativos, transformándola y revolucionando los métodos de enseñanza con plataformas de aprendizaje personalizadas donde los estudiantes aprenderán a su propio ritmo, tutores de IA que asistirán a los estudiantes de manera individualizada, juegos personalizados que se adaptarán a las necesidades educativas de cada uno o con sistemas de evaluación automática de las calificaciones y el rendimiento de los estudiantes [11].

A pesar de ser un tema de actualidad, existe en la sociedad un gran desconocimiento sobre IA, por lo que algunos autores destacan la necesidad de una formación generalizada para facilitar la creación de regulaciones pertinentes, ya que esto podría obstaculizar la implementación de normativas éticas y efectivas, de forma que no solo permitiese a la sociedad comprender mejor estas tecnologías, sino que también fomentaría un diálogo informado para la formulación de políticas [16], como el reciente reglamento aprobado a este respecto en la Unión Europea [3].

Con el objetivo de mejorar los conocimientos sobre IA en estudiantes preuniversitarios, Touretzky [15] propone cinco conceptos que todo joven debe conocer:

1. Los ordenadores pueden percibir el mundo gracias al uso de sensores.
2. Se hace uso de modelos en estos sistemas que son representaciones del mundo y les sirve para la toma de decisiones.
3. Los ordenadores pueden aprender de los datos.
4. Que los agentes interactúen de manera cómoda con los humanos supone un desafío importante para los desarrolladores.
5. Los sistemas de IA pueden impactar en la sociedad tanto de manera positiva como negativa.

Teniendo en cuenta lo descrito anteriormente, en este trabajo se recoge la descripción de un evento en el que se llevaron a cabo actividades en las que se trabajan conceptos de IA en estudiantes de Educación Secundaria, con el objetivo de ver si existía un cambio en la actitud que presentan frente a ella.

El resto del trabajo se organiza de la siguiente manera: en el apartado 2 se presenta la metodología que se ha seguido, los resultados del estudio se recogen en el apartado 3, y, finalmente, las discusiones y conclusiones se presentan en el apartado 4.

## 2. Metodología

La metodología de este trabajo se ha centrado en la realización de actividades con estudiantes preuniversitarios. En dichas actividades se ponen en práctica diferentes conceptos de los sistemas de IA. Se que-

ría comprobar si la actitud de los estudiantes hacia la IA se veía influenciada por la actividad, para ello se utilizó el instrumento ‘Actitud hacia la Inteligencia Artificial’ (*Attitude Towards Artificial Intelligence - ATAI*) [13]. Se contó con dos grupos uno experimental y otro de control que no realizó las actividades de IA, los cuales se organizaron siguiendo razones de logística.

La recogida de datos se llevó a cabo en el marco de la VI Olimpiada de Pensamiento Computacional organizada en la Universidad de La Laguna (ULL) en diciembre de 2023 [5]. Este evento, organizado por el Aula Cultural de Pensamiento Computacional de la ULL, busca promover las Ciencias de la Computación entre los más jóvenes y, especialmente, entre las chicas, debido al escaso interés que sienten por estas [8, 9]. La olimpiada se organiza como una práctica de Aprendizaje-Servicio de la asignatura de Lenguajes y Paradigmas de Programación del tercer curso del Grado en Ingeniería Informática [5, 6, 12], donde los estudiantes universitarios deben diseñar una actividad en la que expliquen diferentes conceptos de programación, como condicionales o variables, de manera desenchufada para estudiantes de la Educación Primaria, es decir, que no se hace uso de computadoras, o enchufada para los de la Educación Secundaria, y luego llevarla a cabo el día de la olimpiada.

En la realización de la práctica participaron un total de 73 universitarios organizados en grupos de 3 personas. Cada grupo de estudiantes realizó un conjunto de ejercicios en los que se aplicaron conceptos relacionados con el pensamiento computacional: secuencias, condicionales, bucles, variables, funciones e IA. Para cada uno de estos conceptos se contó con 3 actividades alternativas para primaria y 2 para secundaria. Para primaria se realizaron 15 ejercicios olímpicos desenchufados o analógicos agrupados en 3 itinerarios, mientras que para los estudiantes de secundaria se organizaron 8 ejercicios enchufados agrupados en 2 itinerarios (Figura 1) que se llevaron a cabo en el Centro de Cálculo [7]. Las actividades de IA de cada itinerario fueron: “Explorando la Inteligencia Artificial: reconociendo casas” y “Labyrinth”.

### 2.1. Participantes

De los 203 asistentes a la VI Olimpiada de Pensamiento Computacional, 82 eran estudiantes de secundaria siendo 38 chicas, 40 chicos y 4 no binarios. Estos se dividieron en dos grupos uno experimental con 26 estudiantes y otro de control con 56, siguiendo razones de logística. El grupo experimental realizó actividades relacionadas con IA mientras que el de control realizó otras actividades de los itinerarios. La distribución completa se recoge en el Cuadro 1, junto con la cantidad de estudiantes según el género.

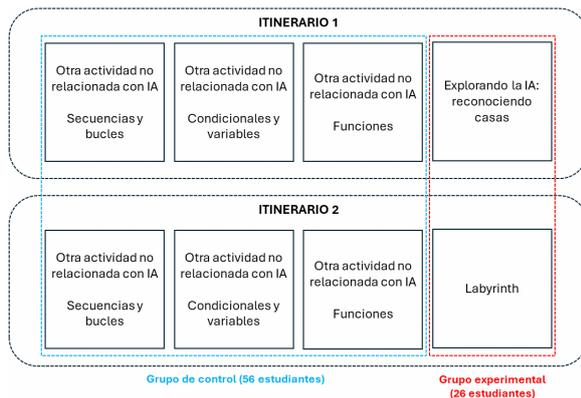


Figura 1: Diseño del experimento.

Grupo de control			Grupo experimental		
Chicas	Chicos	No binarios	Chicas	Chicos	No binarios
26	27	3	12	13	1
Edad media			Edad media		
15,04	15,15	15,33	14,42	14,61	14
TOTAL: 56 estudiantes			TOTAL: 26 estudiantes		

Cuadro 1: Descripción de la muestra de estudiantes de Educación Secundaria participantes en la VI Olimpiada de Pensamiento Computacional.

## 2.2. Actividades

Dentro de las actividades enfocadas a los participantes de Educación Secundaria se diseñaron ocho propuestas, divididas en dos itinerarios de cuatro actividades, con una duración de 40 minutos cada una. Cada participante realizó únicamente un itinerario.

Las actividades donde se presentaron los diferentes conceptos de IA fueron las siguientes:

- **Explorando la Inteligencia Artificial: reconociendo casas:** en esta actividad los estudiantes exploran el mundo de la IA junto con el reconocimiento de imágenes. Para ello, es necesario utilizar ordenadores con algún programa de dibujo y conexión a Internet. La sesión comienza con una breve explicación sobre la IA y cómo es posible que las máquinas aprendan de los ejemplos que se le proporcionan, animándole a que sean ellos los que propongan diferentes categorías y diferencien entre distintos tipos de dibujos.

Posteriormente, se introduce la herramienta *Teachable Machine*<sup>1</sup>, desarrollada por Google y la cual permite crear y entrenar modelos de aprendizaje automático para reconocer imágenes o sonidos de una manera sencilla y para la que no es necesario contar con conocimientos previos sobre programación o IA. Usando esta herramienta, los

estudiantes tendrán la oportunidad de entrenar un modelo que es capaz de reconocer diferentes elementos de imágenes. En este caso, se proporcionan los propios dibujos realizados en la sesión, los cuales deberán tener elementos distintivos como árboles, ventanas, etc.

La actividad concluye con una reflexión sobre el uso de este tipo de sistemas y herramientas, y cómo se deben entrenar para evitar los sesgos.

En la Figura 2 se muestra un ejemplo del entrenamiento del modelo, donde se aprecia un sesgo de IA, ya que se identifica un pino como una casa.

- **Labyrinth:** la actividad propuesta consiste en utilizar la herramienta *Machine Learning for Kids*<sup>2</sup> para que los estudiantes creen y entrenen una IA capaz de identificar diversos movimientos a partir de imágenes captadas por una cámara. Utilizando la herramienta, deben desarrollar un proyecto que categorice imágenes de manos realizando diferentes movimientos, como señalar arriba, abajo, izquierda o derecha, según las distintas formas y posiciones. Posteriormente, deben aplicar este modelo en un proyecto en *Scratch*<sup>3</sup>, concretamente un laberinto donde la IA identifica el movimiento correspondiente a la imagen captada, guiando a un personaje a través del laberinto hasta alcanzar la meta.

En la actividad, se fomenta el pensamiento analítico y lógico al descomponer el proceso en etapas para diseñar un sistema de reconocimiento, poniendo el foco en la posibilidad de que la máquina mejore su capacidad de identificación con el autoaprendizaje al recibir más información, según las imágenes que le proporcionen los estudiantes.

En la Figura 3 se muestra el código a desarrollar para obtener la solución al laberinto, teniendo en cuenta que es necesario utilizar la plataforma *Machine Learning for Kids* para entrenar el modelo.

## 2.3. Actitud hacia la Inteligencia Artificial (ATAI)

El instrumento ‘Actitud hacia la Inteligencia Artificial’, (*Attitude Towards Artificial Intelligence - ATAI*) [13] permite conocer la actitud de una persona hacia la Inteligencia Artificial. Se trata de cinco preguntas en una escala de Likert de 11 puntos donde un 0 indica “en total desacuerdo” y un 10 “totalmente de acuerdo”. Previamente, contestaban también sobre la edad que tenían y su género. Las respuestas de los estudiantes se recogieron utilizando un formulario en línea con las siguientes preguntas en español:

P1 Temo a la Inteligencia Artificial

<sup>2</sup><https://machinelearningforkids.co.uk/>

<sup>3</sup><https://scratch.mit.edu/>

<sup>1</sup><https://teachablemachine.withgoogle.com/>

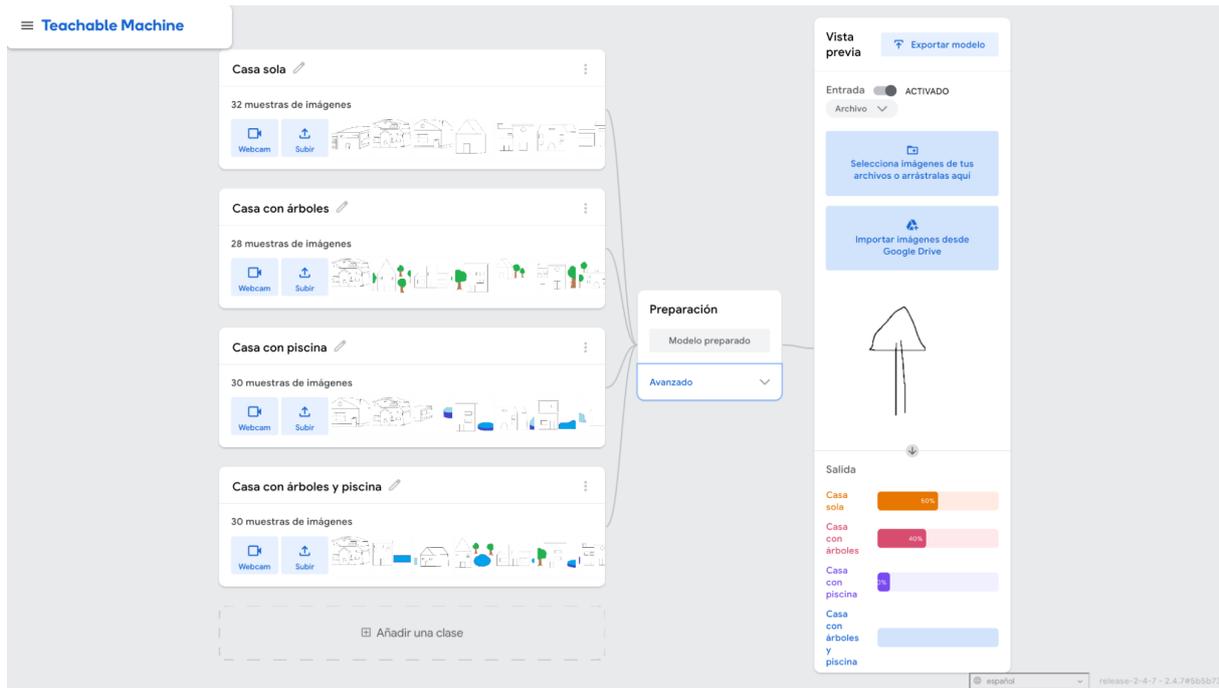


Figura 2: Actividad *Explorando la Inteligencia Artificial: reconociendo casas*.

- P2 Confío en la Inteligencia Artificial
- P3 La Inteligencia Artificial destruirá a la humanidad
- P4 La Inteligencia Artificial beneficiará a la humanidad
- P5 La Inteligencia Artificial provocará muchas pérdidas de empleo

## 2.4. Análisis de los datos

Para llevar a cabo el estudio de los resultados se realizó de manera previa una limpieza de los datos, donde se comprobó que no existiesen datos duplicados o que no se correspondiesen con la actividad realizada, tras el cual no se eliminó ningún dato.

Una vez finalizado el preprocesado de los datos, se llevó a cabo un análisis de Alfa de Cronbach para asegurar la fiabilidad de las muestras recogidas [2], además del cálculo de la t de Student [14] para determinar la existencia de diferencias significativas en las muestras, entre el grupo de control y el experimental, además de entre chicas y chicos, si el nivel de significancia es del 95 % ( $p < 0,05$ ). Además, se realizó un estudio de la puntuación media obtenida en cada una de las preguntas.

## 3. Resultados

En esta sección se describen los resultados de la evaluación de la actitud hacia la IA de los estudiantes.

	P1	P2	P3	P4	P5
Diferencias control y expr.	0,646	<b>0,0383</b>	0,3952	0,8932	0,6804
Diferencias entre chicas y chicos	0,9326	0,1659	<b>0,012</b>	0,1155	0,0714

Cuadro 2: P-valores obtenidos al estudiar las diferencias significativas entre los grupos de control y experimental y entre chicas y chicos.

Ambos grupos cumplieron el instrumento ATAI después de finalizar la actividad.

Con el objetivo de estudiar la fiabilidad de la muestra obtenida, se realizó un estudio del Alfa de Cronbach con toda la muestra y se obtuvo un valor  $\alpha = 0,5512$ . Al realizar un estudio sobre la existencia de diferencias significativas con la t de Student y un valor del 95 % ( $p < 0,05$ ), se aprecian diferencias entre el grupo de control y el experimental, en concreto en la pregunta **P2**, con un p-valor de 0,0383, tal y como se recoge en el Cuadro 2. Sin embargo, no se aprecian diferencias en el género entre chicas y no binarios, y chicos y no binarios, por lo que se ha decidido eliminar dicha muestra del estudio. En cuanto a diferencias entre chicas y chicos, sí existen diferencias significativas, en concreto para la pregunta **P3**, con un valor de 0,012.

Como se observa en la Figura 4a, atendiendo a los resultados de las chicas, el temor que tienen a la IA (**P1**) aumenta tras haber realizado la actividad en más de 1 punto. En cuanto a la confianza (**P2**), esta dismi-

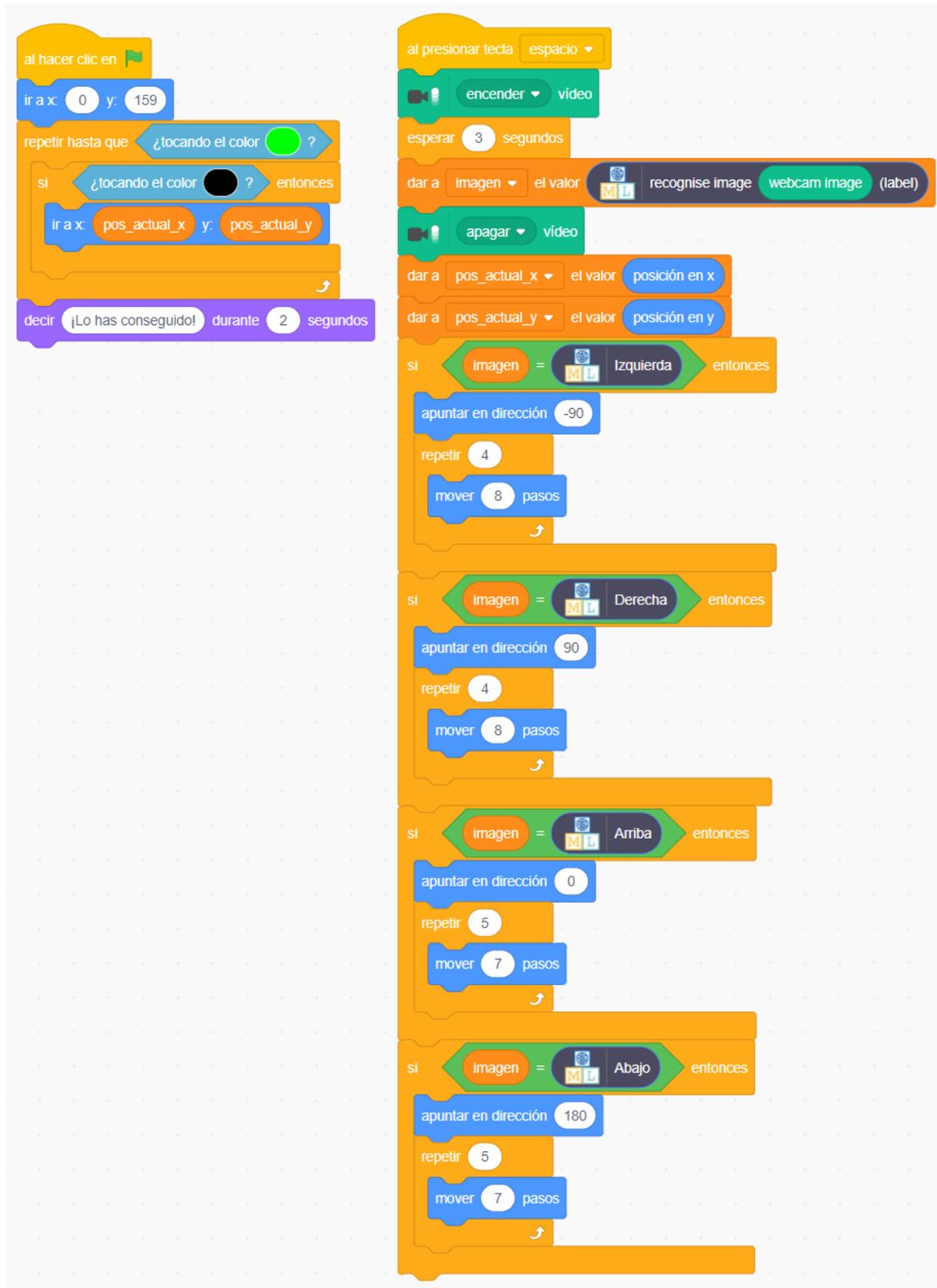
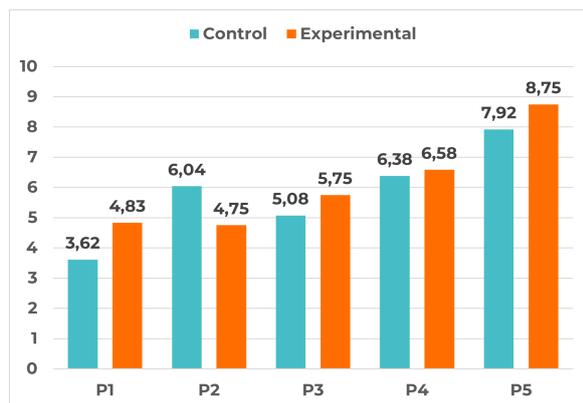
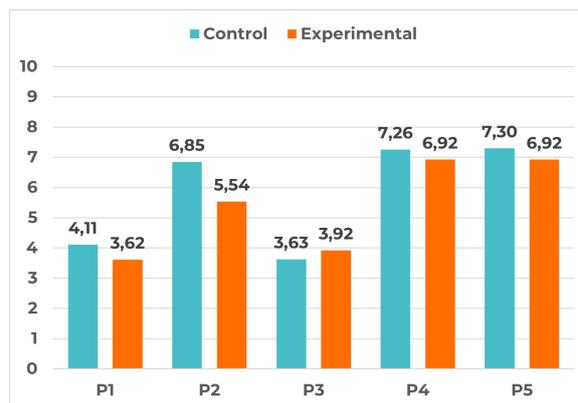


Figura 3: Actividad *Labyrinth*.



(a) Resultados para chicas.



(b) Resultados para chicos.

Figura 4: Puntuaciones medias obtenidas en el ATAI, diferenciando entre el grupo control y experimental.

nuye, mientras que la percepción que tienen sobre si llegará a destruir la humanidad (P3) aumenta, a la vez que creen más que traerán beneficios a la sociedad (P4) y también aumenta la idea que tienen sobre si provocará muchas pérdidas de empleo (P5).

En la Figura 4b se recogen las puntuaciones medias de los chicos al ATAI antes y después de la actividad, y en ella se puede observar como disminuye el temor que tienen a la IA (P1), al igual que la confianza que le tienen (P2). En cuanto a la percepción sobre si destruirá la humanidad (P3), esta aumenta, a la par que disminuye los beneficios que creen que traerá a la humanidad (P4), pero también disminuye el pensamiento sobre las pérdidas de empleo que provocará (P5).

Tal y como se recoge en la Figura 5, existen diferencias en la actitud que los estudiantes muestran sobre la IA, respecto al género. En el caso del temor que le tienen (P1), en las chicas aumenta, pero en los chicos disminuye, mientras que la confianza (P2) en ambos géneros disminuye al realizar la actividad. Respecto a si creen que destruirá la humanidad (P3), aumenta en ambos casos, pero en los beneficios que pueden dar a la humanidad (P4), la percepción aumenta en el caso de las chicas, pero disminuye en el de los chicos. Por último, sobre si provocará pérdidas de empleo (P5), la puntuación de las chicas aumenta, pero la de los chicos disminuye.

#### 4. Discusión y conclusiones

Este trabajo recoge los resultados y el análisis del cambio de la actitud hacia la Inteligencia Artificial por parte de estudiantes preuniversitarios tras realizar actividades en la que se introducen conceptos como el entrenamiento supervisado, algoritmos de clasificación o creación de modelos.

Los resultados muestran una gran disparidad entre los dos grupos, el de control que realizó otro tipo de

actividades, y el experimental que sí realizó actividades en las que se introducen conceptos de IA. Por un lado, el grupo de control tiene una opinión más positiva ya que le temen menos y su confianza es mayor que los del grupo experimental que sí trabajaron estos conceptos de IA. Esto también se da cuando se preguntan sobre si creen que destruirá a la humanidad, puesto que la actitud del grupo experimental a este respecto aumenta, mientras que creen que sí que traerá beneficios a la humanidad, pero no hay variación relevante en este aspecto.

El estudio respecto al género deja claro las diferencias que existen entre chicas y chicos en la percepción de este ámbito, ya que los resultados son completamente diferentes, donde ellas tienen un temor mucho mayor que ellos, al igual que una confianza menor, ya que también piensan que provocará muchas pérdidas de empleo.

Este trabajo pretende mostrar a los jóvenes cómo funcionan los sistemas de Inteligencia Artificial, y que no se conviertan en simple usuarios. Esta visión, donde entrenan un modelo y conocen los posibles sesgos que se pueden generar, hacen que se vuelvan más críticos y negativos con este tipo de sistemas, por lo que es necesario educar en este campo para que lo conozcan lo suficiente y sean capaces de emitir sus propios juicios fundamentados, realizando actividades donde se trabajen estos conceptos.

Dentro de esta experiencia, es posible que el tipo de actividades que realizaron los estudiantes influya en su visión sobre la Inteligencia Artificial, por lo que se plantea como trabajo futuro una mayor variedad de actividades en las que se trabaje también, por ejemplo, el aprendizaje no supervisado o por refuerzo.

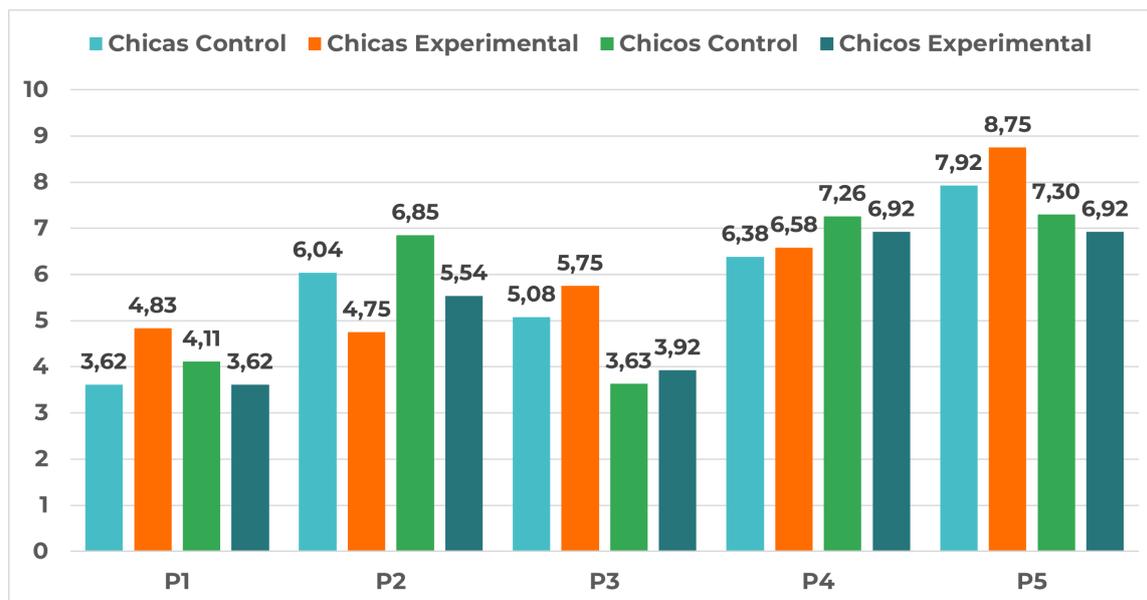


Figura 5: Puntuaciones medias obtenidas en el ATAI, diferenciando entre los grupos de control y experimental y el género.

## Agradecimientos

Este trabajo se ha realizado con el apoyo institucional de la Universidad de La Laguna y del Vicerrectorado de Innovación Docente, Calidad y Campus Anchieta en el marco del Proyecto de Innovación y Transferencia Educativa (PITE) del 2023-2024 denominado “Aprendizaje Servicio y Pensamiento Computacional”.

Este trabajo ha sido financiado parcialmente por el Consejo Social de la Universidad de La Laguna y por el Cabildo de Tenerife a través del proyecto “Piensa Computacionalmente 2023/24 (REF 23120270)”.

El trabajo de Rafael Herrero-Álvarez ha sido cofinanciado por la Agencia Canaria de Investigación, Innovación y Sociedad de la Información de la Consejería de Universidades, Ciencia e Innovación y Cultura y por el Fondo Social Europeo Plus (FSE+) Programa Operativo Integrado de Canarias 2021-2027, Eje 3 Tema Prioritario 74 (85 %).

## Referencias

- [1] EU Artificial Intelligence Act. Artículo 3: Definiciones | Ley de Inteligencia Artificial de la UE.
- [2] Lee J. Cronbach. Coefficient alpha and the internal structure of tests. *Psychometrika*, 16(3):297–334, Sep 1951.
- [3] Consejo Europeo de la Unión Europea. Inteligencia artificial - Consilium, 12 2023.
- [4] Yuval Noah Harari. Reboot for the ai revolution. *Nature (London)*, 550(7676):324–327, Oct 19, 2017.
- [5] Rafael Herrero-Álvarez, Coromoto León, Israel López-Plata, y Gara Miranda. Experiencia de aprendizaje-servicio en una asignatura de programación del grado en ingeniería informática. En *XXIX Jornadas sobre Enseñanza Universitaria de la Informática (JENUI)*, pp. 197–204, 2023.
- [6] Rafael Herrero-Álvarez, Coromoto León, Israel López-Plata, y Luz Marina Moreno. Propuesta de actividades y guías de evaluación para estudiar paradigmas de programación. En *Avances en Tecnologías, Innovación y Desafíos de la Educación Superior (ATIDES 2022)*, pp. 237–252, 2022.
- [7] Rafael Herrero-Álvarez, Coromoto León, Gara Miranda, y Luz Marina Moreno de Antonio. Actas de la VI Olimpiada de Pensamiento Computacional, enero 2024.
- [8] Rafael Herrero-Álvarez, Gara Miranda, Coromoto León, y Eduardo Segredo. Engaging primary and secondary school students in computer science through computational thinking training. *IEEE Transactions on Emerging Topics in Computing*, 11(1):56–69, 2023.
- [9] Rafael Herrero-Álvarez, Coromoto León, Gara Miranda, y Eduardo Segredo. Training future engineers: Integrating Computational Thinking and effective learning methodologies into education. *Computer Applications in Engineering Education*, 2 2024.

- [10] Partha Pratim Ray. Chatgpt: A comprehensive review on background, applications, key challenges, bias, ethics, limitations and future scope. *Internet of Things and Cyber-Physical Systems*, 3:121–154, 2023.
- [11] Francisco José Ruiz Rey. *La inteligencia artificial en entornos educativos*, pp. 454–474. Docencia, ciencia y humanidades: hacia una enseñanza integral en la universidad del siglo XXI. Dykinson S.L., 2021.
- [12] Casiano Rodríguez-León, Coromoto León, Gara Miranda, Eduardo Segredo, y Carlos Segura. *Prácticas de laboratorio en Ruby para lenguajes y paradigmas de programación*, pp. 548–579. Innovación docente en la educación superior: una recopilación de experiencias prácticas aplicadas. Vicerrectorado de Calidad Institucional e Innovación Educativa. Universidad de La Laguna, 2013.
- [13] Cornelia Sindermann, Peng Sha, Min Zhou, Jennifer Wernicke, Helena S. Schmitt, Mei Li, Rayna Sariyska, Maria Stavrou, Benjamin Becker, y Christian Montag. Assessing the attitude towards artificial intelligence: Introduction of a short measure in german, chinese, and english language. *KI. Künstliche Intelligenz (Oldenbourg)*, 35(1):109–118, Mar 1, 2021.
- [14] Student. The probable error of a mean. *Biometrika*, 6:1–25, 1908.
- [15] David Touretzky, Christina Gardner-McCune, Fred Martin, y Deborah Seehorn. Envisioning ai for k-12: What should every child know about ai? *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(1):9795–9799, Jul 17, 2019.
- [16] James M. White y Rolf Lidskog. Ignorance and the regulation of artificial intelligence. *Journal of risk research*, 25(4):488–500, Apr 24, 2022.

# Explorando metodologías de evaluación por pares offline y online en Ingeniería Informática

Sergio Santander-Jiménez<sup>1</sup>, Miguel A. Vega-Rodríguez<sup>1</sup>, José M. Granado-Criado<sup>1</sup>,  
Álvaro Rubio-Largo<sup>2</sup>, Juan A. Gómez-Pulido<sup>1</sup>, César Gómez-Martín<sup>1</sup>,  
Arturo Durán-Domínguez<sup>1</sup>

1) Departamento de Tecnología de los Computadores y de las Comunicaciones

2) Departamento de Ingeniería de Sistemas Informáticos y Telemáticos

Universidad de Extremadura

10003 Cáceres

sesaji@unex.es, mavega@unex.es, granado@unex.es,

arl@unex.es, jangomez@unex.es, cesar@unex.es, arduan@unex.es

## Resumen

La evaluación por pares representa una de las técnicas clave para involucrar al estudiantado en el proceso de enseñanza-aprendizaje. No obstante, su implementación conlleva en el contexto actual nuevos retos y cuestiones a abordar, de acuerdo a la disponibilidad tecnológica y la heterogeneidad de los estudiantes. Este trabajo investiga la aplicación de estrategias de evaluación por pares en siete asignaturas de Ingeniería Informática. Esta investigación se ejecutó durante cinco cursos académicos en dos fases, cubriendo la transición offline a online motivada por el COVID. La experiencia offline muestra cómo una implementación sencilla y no ambigua de las tareas y rúbricas permite obtener correlaciones estadísticas entre las calificaciones asignadas por los estudiantes y las del profesor. Además, se destaca cómo la asistencia a clase, el perfil de los estudiantes y otros factores externos pueden afectar al rendimiento. Por su parte, el cambio a metodologías online no provocó desviaciones significativas en asignaturas con experiencia previa en evaluación por pares. En base a esta experiencia, se aporta una discusión sobre el impacto de las metodologías offline y online conforme a cuatro aspectos: confianza en la evaluación, resolución de dudas, repercusión en el tiempo de clase y manejo de los grupos de impartición.

## Abstract

Peer assessment represents a key technique to involve students in the teaching-learning process. However, the implementation of peer assessment nowadays entails new challenges and issues to be addressed, in accordance with technological advances and students' heterogeneity. This work investigates the application of

peer assessment strategies to seven Computer Engineering courses. This research was conducted throughout five academic years and it was structured in two phases, which covered the offline-to-online shift imposed by COVID. The offline experience shows that a simple, non-ambiguous implementation of tasks and rubrics leads to statistical correlations between peer grading and lecturer grading. Moreover, it is highlighted how class attendance, student profiles, and other external factors can impact performance. On the other side, the shift to online methodologies did not motivate significant deviations in courses with previous experience in peer assessment. In light of this experience, we contribute with a discussion about the impact of adopting offline or online methodologies from four perspectives: confidence in the assessment, doubt solving, influence in class time, and group management.

## Palabras clave

Evaluación por pares, metodologías offline y online, análisis de rendimiento, experiencias docentes.

## 1. Introducción

A lo largo de los años, se han propuesto diversos métodos de evaluación para incentivar la participación del alumnado en las metodologías de enseñanza-aprendizaje, con vistas a potenciar su pensamiento crítico y contribuir a la adquisición exitosa de competencias. Entre ellos, la evaluación por pares ha jugado un papel clave en el ámbito universitario [6], la educación secundaria [7] y otras áreas [2]. Al permitir a los estudiantes actuar como evaluadores, esta aproximación permite desarrollar sus habilidades comunicativas y de

gestión de tareas, reflexionar y repasar puntos importantes del temario y alcanzar un mejor entendimiento de los criterios de evaluación [1]. Para el profesor, la evaluación por pares contribuye a alcanzar una experiencia docente más gratificante, resultado de un mejor entendimiento de la asignatura por los estudiantes.

Existen múltiples estudios que dan cuenta de aplicaciones de la evaluación por pares en distintos contextos. Podemos destacar trabajos como [10], donde se empleó la evaluación por pares para mejorar el desarrollo de competencias comunicativas para estudiantes de ingeniería. En [12], se estudiaron estrategias de evaluación por pares en el ámbito del procesamiento digital de la señal, señalando sus beneficios para mejorar el rendimiento de los estudiantes. Estos métodos fueron estudiados en [4] para la evaluación de proyectos de ingeniería por equipos (equivalentes a Proyectos Fin de Estudios), como herramienta para examinar el rendimiento individual de cada miembro del equipo. Por su parte, la experiencia publicada en [9] incide en la valía que supone la evaluación por pares en el caso de una asignatura de procesamiento de lenguaje natural.

No obstante, la implementación de la evaluación por pares representa una tarea compleja hoy en día. Esto es debido a que este proceso puede verse afectado por distintos factores, como el estado físico y anímico de los estudiantes, desconfianza a la hora de jugar el papel de evaluador, posible ambigüedad en la interpretación de las rúbricas y otras variables relacionadas con la profesionalidad de las evaluaciones y su integridad [8]. De hecho, los cambios metodológicos sufridos en los entornos académicos, especialmente el giro offline a online, han enfatizado el impacto de estos factores, trayendo consigo nuevos desafíos tecnológicos [5]. En consecuencia, es necesario estudiar con profundidad el desarrollo de estrategias para una implementación correcta y precisa de la evaluación por pares.

Este trabajo presenta una experiencia docente orientada a investigar estrategias de evaluación por pares offline (presencial) y online (virtual) en diferentes asignaturas del ámbito de la Ingeniería Informática y las Tecnologías de la Información, a lo largo de cinco cursos académicos. Con ello pretendemos ir un paso más allá con respecto a los trabajos relacionados, abarcando múltiples asignaturas sobre un margen temporal que cubre los años pre y post-pandemia. Este trabajo fue estructurado en dos fases. La fase 1, de cuatro años de duración, se centró en la realización de evaluación por pares presencial en dos asignaturas: Arquitectura de Computadores y Sistemas en Tiempo Real. En la fase 2, realizada en el curso 2021/2022, se investigó la integración de evaluación por pares online en las dos asignaturas anteriores y cinco adicionales: Análisis y Diseño de Algoritmos, Diseño y Administración de Bases de Datos, Documentación Científica e Infor-

mática, Domótica y Estructura de Computadores. Mediante el análisis de los resultados obtenidos por evaluación por pares en un rango amplio de asignaturas, las cuales abarcan distintas etapas del grado, se pretende responder a varias cuestiones de investigación:

1. ¿La evaluación por pares representa una herramienta precisa para evaluar a los estudiantes de Ingeniería Informática?
2. ¿Cuáles son los factores más importantes que influyen en el rendimiento bajo evaluación por pares, para distintas promociones de estudiantes y asignaturas?
3. ¿Afecta la transición offline a online en la fiabilidad del proceso de evaluación por pares? ¿Se dan desviaciones al migrar de un escenario a otro?
4. ¿Cómo impactan las metodologías offline y online al desarrollo de las clases y al proceso de evaluación en sí?

Este artículo está organizado del siguiente modo. La Sección 2 detalla en profundidad las fases del estudio y explica las metodologías adoptadas para realizar evaluación por pares en las asignaturas consideradas. La Sección 3 presenta y analiza los resultados obtenidos, discutiendo las cuestiones de investigación identificadas en este trabajo. Finalmente, la Sección 4 presenta las conclusiones de esta experiencia docente y marca posibles líneas de trabajo futuro.

## 2. Metodologías de evaluación

En esta sección describimos las metodologías aplicadas en esta experiencia docente. La figura 1 muestra la cronología de las dos fases del estudio.

### 2.1. Fase 1: evaluación offline

La primera fase se centró en la aplicación en clase de evaluación por pares presencial durante cuatro cursos académicos. Las asignaturas objeto de estudio fueron Arquitectura de Computadores (asignatura obligatoria de segundo semestre, impartida en el tercer año del grado) y Sistemas en Tiempo Real (asignatura obligatoria de primer semestre, impartida en el cuarto año).

Las actividades de evaluación por pares fueron implementadas como parte de las tareas de la evaluación continua, contribuyendo 2 puntos sobre 10. Se emplearon para ello preguntas de respuesta corta y problemas. Entre cuatro y seis actividades de evaluación fueron realizadas durante el semestre. La instrucción de los estudiantes se realizó presencialmente en clase, explicándoles el modo de proceder junto con los conceptos necesarios (evaluación por pares, revisor, rúbrica, sistema de calificación, etc.) y poniendo énfasis en la importancia de realizar una evaluación profesional.

		1 <sup>er</sup> semestre	2 <sup>o</sup> semestre
Fase 1: Evaluación offline	2016/2017	Sistemas en Tiempo Real	Arquitectura de Computadores
	2017/2018	Sistemas en Tiempo Real	Arquitectura de Computadores
	2018/2019	Sistemas en Tiempo Real	Arquitectura de Computadores
	2019/2020	Sistemas en Tiempo Real	COVID-19
	2020/2021	COVID-19	Arquitectura de Computadores
Fase 2: Evaluación online	2021/2022	Análisis y Diseño de Algoritmos	Estructura de Computadores
		Diseño y Admón. Bases de Datos	Arquitectura de Computadores
		Sistemas en Tiempo Real	Domótica Documentación Científica e Informática

Figura 1: Organización temporal de la experiencia docente, asignaturas y fases del estudio.

En cada actividad, los estudiantes debían resolver varias cuestiones propuestas por el profesor en clase, relativas a los contenidos impartidos de una actividad a otra. Al acabar, el profesor designaba de manera aleatoria los revisores responsables de evaluar a sus compañeros, utilizando un modelo de revisión no ciego. La evaluación se veía apoyada por una rúbrica mostrada en clase que contenía una explicación detallada de las respuestas correctas. Además, se mostraban las diapositivas de los apuntes que contenían dichas respuestas, con objeto de ayudar a los estudiantes a localizar en el material de la asignatura las temáticas preguntadas.

Para evitar ambigüedad en la asignación de calificaciones, se definió la siguiente escala de puntuación:

- Correcto: asignada cuando la respuesta del estudiante cubría todos los aspectos de la rúbrica.
- Regular+: se asignaba esta nota cuando la respuesta del estudiante era mayormente correcta pero no incluía algún punto aislado de la rúbrica.
- Regular: nota intermedia, aplicada cuando la respuesta del estudiante era parcialmente correcta.
- Regular-: se asignaba esta nota cuando la respuesta del estudiante no era correcta pero se mencionaba algún aspecto especificado en la rúbrica.
- Mal: asignada cuando el estudiante no respondía o su respuesta era completamente errónea.

Esta metodología fue aplicada a Arquitectura de Computadores durante los cursos 2016/2017 (con 10 estudiantes), 2017/2018 (23 estudiantes), 2018/2019 (17 estudiantes) y 2020/2021 (34 estudiantes). En Sistemas en Tiempo Real, estas actividades fueron reali-

zadas durante los cursos 2016/2017 (18 estudiantes), 2017/2018 (21 estudiantes), 2018/2019 (21 estudiantes) y 2019/2020 (18 estudiantes). Las actividades del segundo semestre del curso 2019/2020 y del primero de 2020/2021 fueron canceladas debido a la pandemia.

## 2.2. Fase 2: evaluación online

La segunda fase exploró la implementación de la evaluación por pares en entornos online durante el curso académico 2021/2022, motivada por las circunstancias derivadas de la pandemia de COVID-19. Junto con Arquitectura de Computadores y Sistemas en Tiempo Real, se consideraron otras cinco asignaturas: Análisis y Diseño de Algoritmos, Estructura de Computadores, Diseño y Administración de Bases de Datos, Domótica y Documentación Científica e Informática. Análisis y Diseño de Algoritmos y Estructura de Computadores son asignaturas obligatorias de segundo año, impartidas en el primer y segundo semestre respectivamente. Por su parte, Diseño y Administración de Bases de Datos es una asignatura obligatoria de primer semestre impartida en el tercer año del grado. Finalmente, Domótica y Documentación Científica e Informática son asignaturas optativas de segundo semestre.

Siguiendo la línea de la primera fase, las actividades de evaluación por pares fueron integradas en estas asignaturas como parte de las tareas de la evaluación continua. Al menos una actividad de evaluación por pares fue realizada en cada asignatura. Se dio libertad a los profesores para elegir la metodología y plataforma online a utilizar para la gestión de envíos y revisiones, de acuerdo a las características de sus asignaturas y los grupos de impartición (considerando, por ejemplo, el número de alumnos matriculados y el perfil de los mismos). Más en detalle, Sistemas en Tiempo Real implementó un proceso de evaluación doblemente ciego online combinando Tareas de Moodle [3] con hojas de Excel contenedoras de las rúbricas, involucrando a 17 estudiantes. En Arquitectura de Computadores, se realizó evaluación por pares no ciegos con 24 estudiantes a través de un entorno personalizado en la nube.

Las cinco asignaturas restantes emplearon la herramienta Taller de Moodle [6] para implementar la evaluación por pares. Utilizando esta herramienta, los estudiantes reciben dos calificaciones en función de la corrección del trabajo enviado y la precisión del estudiante como revisor. Los pesos de estas dos calificaciones sobre la nota de la actividad fueron establecidos libremente por los profesores, en ventanas del 80%-90% para la calificación del envío y 10%-20% para la calificación de la evaluación. Esta última fue calculada según la desviación observada con respecto a la media de todos los revisores del trabajo asignado (incluyendo la nota del profesor). El sistema fue configurado para asignar evaluadores aleatoriamente entre los estudian-

tes que participaron en la actividad. En lo que respecta a las rúbricas, estas fueron integradas en la interfaz de la página de la fase de evaluación del Taller.

Al utilizar la herramienta Taller, surgieron varias cuestiones en la instrucción de los estudiantes:

1. Algunos estudiantes tenían dificultades para localizar determinados elementos de la evaluación en la interfaz de usuario, particularmente el *feedback* de los revisores. Para evitar este problema, se proporcionó una explicación detallada y visual de la interfaz de la herramienta Taller en cada una de las asignaturas.
2. Algunos estudiantes no justificaban las notas asignadas a sus compañeros. Para intentar asegurar la recepción de *feedback* útil, se configuró la herramienta Taller para hacer obligatorios los comentarios de retroalimentación. En caso de encontrarse revisiones ambiguas o erróneas, se asignaban a las mismas un peso de cero en la evaluación.

Las actividades de evaluación por pares fueron realizadas mediante un modelo simple ciego en Análisis y Diseño de Algoritmos (con 151 estudiantes), Diseño y Administración de Bases de Datos (33 estudiantes) y Documentación Científica e Informática (19 estudiantes). Por su parte, Domótica optó por un modelo doblemente ciego (con 9 estudiantes). Finalmente, Estructura de Computadores incorporó actividades bajo los dos modelos (con 106 estudiantes).

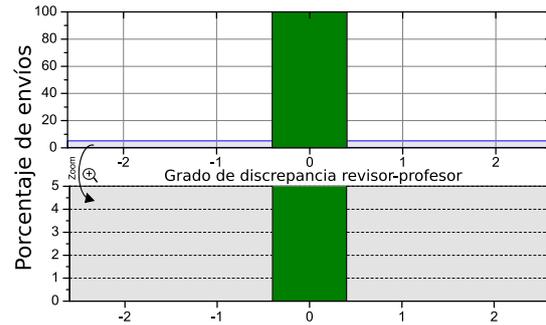
### 3. Análisis de resultados

En esta sección evaluamos los resultados de cada una de las fases de este estudio, a fin de responder las cuestiones de investigación planteadas.

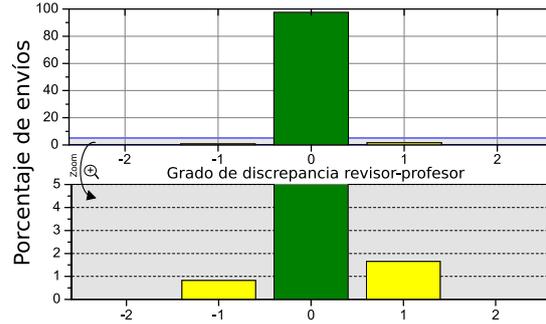
#### 3.1. Resultados de la fase 1

En primer lugar, validaremos la precisión de las evaluaciones realizadas durante la primera fase, examinando las diferencias entre las calificaciones dadas por los revisores y las notas que hubiera asignado el profesor en su lugar. Las figuras 2 y 3 presentan las diferencias observadas en Arquitectura de Computadores y Sistemas en Tiempo Real, respectivamente.

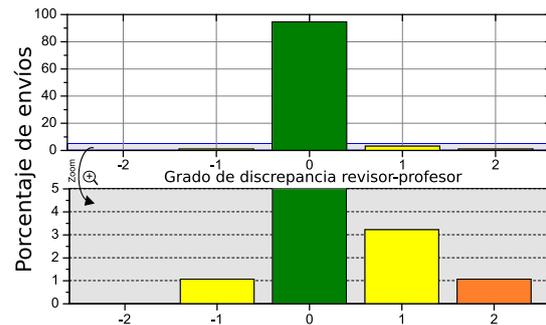
En el caso de Arquitectura de Computadores, la figura 2 denota una fuerte similitud entre las calificaciones de los revisores y del profesor. Estas notas coincidieron en su totalidad (diferencia en la escala de puntuación = 0) en más del 90,50% del total de las evaluaciones realizadas durante esta fase offline. En desglose por curso académico, no se observaron desviaciones en ninguna de las actividades para el curso 2016/2017, para 51 muestras de evaluación. En el curso 2017/2018 (121 muestras), no se dieron desviacio-



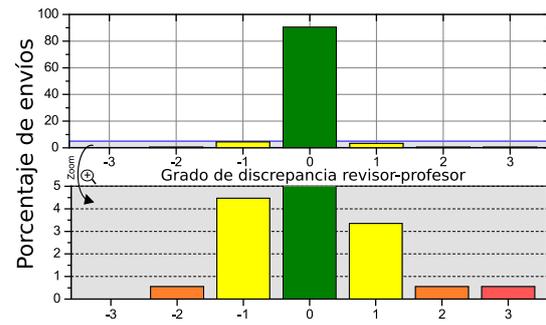
(a) 2016/2017



(b) 2017/2018



(c) 2018/2019



(d) 2020/2021

Figura 2: Precisión de la evaluación en Arquitectura de Computadores (grado 0 = coincidencia profesor-revisor, mientras que valores positivos o negativos denotan que los revisores asignaron escalas de puntuación mayor o menor de la que correspondía).

nes en un 97,52% de los envíos, existiendo un 1,65% de los revisores que asignaron notas más altas y un

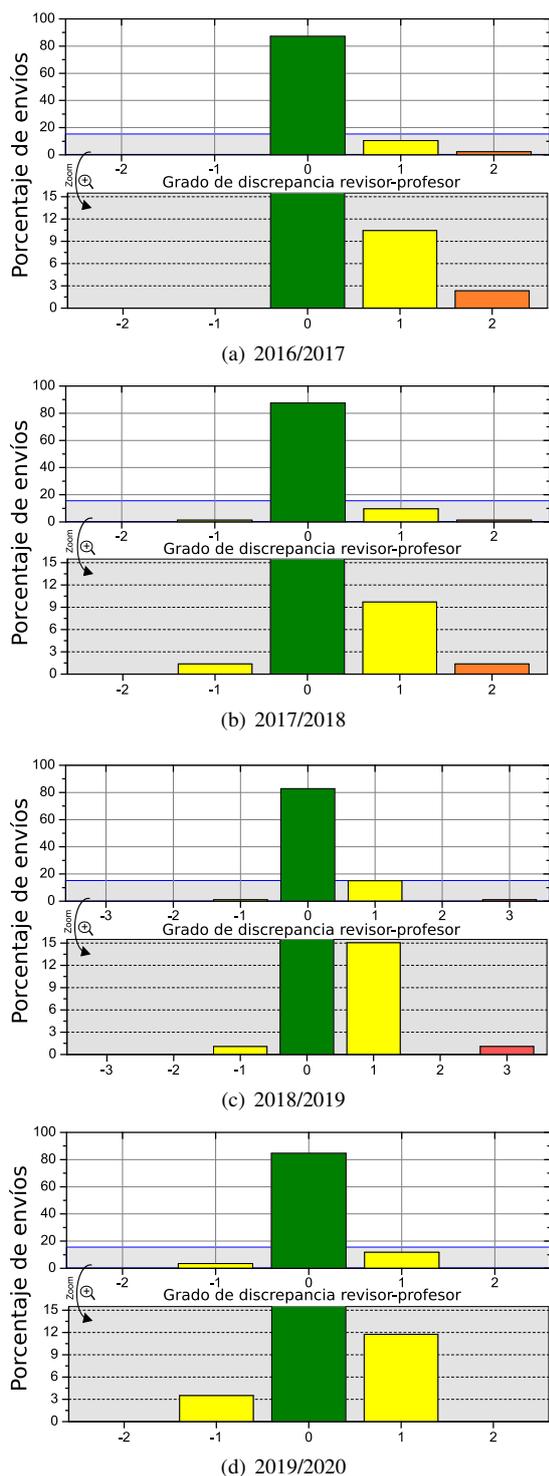


Figura 3: Precisión de la evaluación en Sistemas en Tiempo Real (grado 0 = coincidencia profesor–revisor, mientras que valores positivos o negativos denotan que los revisores asignaron escalas de puntuación mayor o menor de la que correspondía).

0,83 % que dieron notas más bajas. En 2018/2019 (93 muestras), un 94,62 % de las evaluaciones coincidieron

Curso	Test Mann-Whitney		Coeficiente Spearman	
	Valor p	Diferen. sign.?	Valor	Correlación?
2016/2017	1,000	No	1,000	Sí
2017/2018	0,946	No	0,986	Sí
2018/2019	0,896	No	0,993	Sí
2020/2021	0,993	No	0,947	Sí

Cuadro 1: Fase 1 – Comparación de calificaciones revisor-profesor para Arquitectura de Computadores

Curso	Test Mann-Whitney		Coeficiente Spearman	
	Valor p	Diferen. sign.?	Valor	Correlación?
2016/2017	0,221	No	0,866	Sí
2017/2018	0,449	No	0,915	Sí
2018/2019	0,303	No	0,900	Sí
2019/2020	0,339	No	0,822	Sí

Cuadro 2: Fase 1 – Comparación de calificaciones revisor-profesor para Sistemas en Tiempo Real

con la nota que el profesor hubiera asignado, observándose notas más altas en un 4,30 % de las evaluaciones y notas más bajas en un 1,08 %. Finalmente, el curso post-pandemia 2020/2021 (179 muestras) muestra una mayor tasa de variabilidad, existiendo desviaciones en un 9,50 % de las evaluaciones.

Para Sistemas en Tiempo Real, la figura 3 muestra un aumento ligero de las discrepancias entre las notas de los evaluadores y las del profesor. A pesar de ello, la tendencia general apunta a una coincidencia entre revisores y profesor en más del 82,80 % de las evaluaciones realizadas durante la fase offline. En 2016/2017 (86 muestras), un 87,21 % de los envíos recibieron la nota esperada mientras que un 12,79 % consiguieron una nota mayor. Es en el curso 2017/2018 (72 muestras) donde se observó la mayor tasa de acierto en las notas asignadas en esta asignatura, con un 87,50 %. En 2018/2019 (93 muestras), se observaron desviaciones en un 17,20 % de las evaluaciones. Para 2019/2020 (85 muestras), un 84,71 % de los trabajos fueron evaluados con precisión, observándose notas más altas y bajas en porcentajes de 11,76 % y 3,53 %, respectivamente.

Con objeto de estudiar estadísticamente los resultados y verificar si existe una correlación entre las calificaciones de los revisores y del profesor, se han analizado las muestras utilizando el test de Mann-Whitney y el coeficiente de correlación de Spearman [11]. Estos tests no paramétricos se llevaron a cabo considerando un intervalo de confianza del 95 %. El cuadro 1 muestra los resultados de estos tests para Arquitectura de Computadores, mientras que los resultados de Sistemas en Tiempo Real vienen dados en el cuadro 2.

Puede observarse en el cuadro 1 que los valores p de Mann-Whitney para Arquitectura de Computadores fueron mayores que 0,05 en todos los cursos académicos, denotando pues que no se aprecian diferencias estadísticamente significativas entre las calificaciones de los revisores y las que hubiera asignado el profesor.

Arquitectura de Computadores		
Curso	Nota media	Asistencia a clase
2016/2017	8,009	85,000 %
2017/2018	8,528	87,681 %
2018/2019	8,486	91,176 %
2020/2021	6,939	87,745 %
Sistemas en Tiempo Real		
2016/2017	8,733	95,556 %
2017/2018	7,991	85,714 %
2018/2019	8,198	87,619 %
2019/2020	8,520	94,444 %

Cuadro 3: Fase 1 – Comparación de notas medias bajo evaluación por pares y porcentajes de asistencia a clase

A su vez, el test de Spearman apunta a coeficientes de correlación por encima de 0,94, con valores p siempre inferiores a 0,05. Ello denota la existencia de una correlación estadística que sugiere la bondad del proceso de evaluación por pares. Estas ideas también pueden verificarse en Sistemas en Tiempo Real, tal y como muestra el cuadro 2. En concreto, Mann-Whitney indica que no existen diferencias significativas (valores  $p > 0,05$ ), con coeficientes de correlación entre 0,82 y 0,92. De hecho, los valores p de Spearman fueron también inferiores a 0,05, indicando correlación estadística. Por tanto, estos tests revelan que la evaluación por pares offline representó en estas asignaturas una herramienta precisa para evaluar a los estudiantes.

A continuación, estudiamos el rendimiento de los estudiantes a lo largo de la experiencia offline. El cuadro 3 muestra las calificaciones medias obtenidas por asignatura y curso académico, junto con las tasas de asistencia a clase. En Arquitectura de Computadores, las notas más altas (en torno a 8,5 sobre 10) se obtuvieron en los cursos 2017/2018 y 2018/2019. En ambos cursos, la asistencia a clase fue alta (87,68 % en 2017/2018 y 91,18 % en 2018/2019). En comparación, se observaron notas más bajas (8,0) en 2016/2017, curso en el que se verificó la asistencia más baja (85,00 %). Los peores resultados se dieron en el curso 2020/2021, con una nota media de 6,9. Es necesario destacar que, a pesar de que este curso presentó tasas de asistencia similares a 2017/2018, el efecto psicológico de la pandemia pudo hacer mella en los estudiantes, resultando en un empeoramiento de las notas. De hecho, el análisis estadístico de los resultados apunta a la gran importancia del contexto epidemiológico, al darse diferencias estadísticamente significativas ( $p$  de Mann-Whitney = 0,02) entre el curso 2020/2021 y los cursos 2017/2018 y 2018/2019.

Para Sistemas en Tiempo Real, se dieron las mejores notas en los cursos 2016/2017 (8,7) y 2019/2020 (8,5). En estos dos cursos se verificó la mayor tasa de asistencia a clase, por encima del 94,44 %. Por el contrario, las peores notas (por debajo de 8,0) se dieron en el curso con peor asistencia a clase, el 2017/2018. Por

Asignatura	Nota revisor	Nota profesor	Diferencia nota
Análisis y Diseño Algoritmos	6,310	6,140	0,170
Arquitectura de Computadores	7,712	7,643	0,069
Diseño y Admón. Bases Datos	5,250	5,870	-0,620
Doc. Científica e Informática	9,180	7,350	1,830
Domótica	8,400	7,430	0,970
Estructura de Computadores	6,620	5,585	1,035
Sistemas en Tiempo Real	9,456	9,250	0,206

Cuadro 4: Fase 2 – Comparación de calificaciones medias bajo evaluación por pares online

su parte, el curso 2018/2019 representó un escenario intermedio, con una nota media de 8,2 y una asistencia a clase del 87,62 %. Estos resultados sugieren que, bajo circunstancias que no afecten anímicamente a los estudiantes, una asistencia a clase alta representa un factor de importancia para ayudar a obtener mejor rendimiento. En este sentido, se observaron diferencias estadísticamente significativas ( $p$  de Mann-Whitney = 0,04) entre el curso con mejor tasa de asistencia (2016/2017) y los cursos 2017/2018 y 2018/2019.

Otro aspecto interesante a examinar es el rendimiento de una misma promoción de estudiantes a medida que progresa en el grado. Los estudiantes de Arquitectura de Computadores 2016/2017 que se matricularon en Sistemas en Tiempo Real 2017/2018 mostraron en ambas asignaturas la peor tasa de asistencia a clase y, a su vez, las peores notas (sin contar la excepcionalidad del curso 2020/2021). La clase 2017/2018 de Arquitectura, que estudió Sistemas en Tiempo Real en 2018/2019, mantuvo mejores tasas de asistencia y mejor rendimiento que la promoción anterior. La última promoción considerada en esta fase (Arquitectura de Computadores 2018/2019 - Sistemas en Tiempo Real 2019/2020) mostró la mejor tasa de asistencia e incluso superó en Sistemas en Tiempo Real los resultados del año anterior. En conclusión, durante la experiencia offline, puede afirmarse que los estudiantes mostraron, de una asignatura a otra, patrones similares de comportamiento y rendimiento bajo evaluación por pares.

### 3.2. Resultados de la fase 2

En el caso de las evaluaciones online, nos centraremos en estudiar posibles desviaciones entre las notas asignadas por los revisores y las del profesor para así determinar las implicaciones del cambio realizado. Los resultados obtenidos para las siete asignaturas de esta segunda fase se muestran en el cuadro 4.

Puede observarse que las asignaturas con experiencia previa en evaluación por pares, Arquitectura de Computadores y Sistemas en Tiempo Real, consiguieron mantener en las evaluaciones online el acuerdo revisor-profesor mostrado en cursos pasados. En concreto, las diferencias observadas fueron de 0,07 para

Arquitectura de Computadores y 0,21 para Sistemas en Tiempo Real. Esto sugiere que es posible mantener un grado similar de precisión al realizar el cambio a evaluaciones online en asignaturas con fuerte énfasis en el uso de evaluación por pares a lo largo de los años.

Para las asignaturas restantes, las cuales emplearon evaluación por pares por primera vez en esta fase, se puede observar un mayor grado de discrepancia entre los revisores y el profesor. La excepción viene dada por Análisis y Diseño de Algoritmos, donde se dio una desviación de tan solo 0,17 puntos. Por su parte, Diseño y Administración de Bases de Datos verificó una diferencia de -0,62. Ambas asignaturas fueron impartidas por el mismo profesor, aplicándose la misma metodología para la implementación de las actividades. Es por ello que las diferencias aquí mostradas pueden deberse al perfil particular de los estudiantes que cursaron las asignaturas. Para Estructura de Computadores, Domótica y Documentación Científica e Informática, la discrepancia entre revisores y el profesor fue igual o superior a 0,97 puntos. Es digno señalar además que, en Estructura de Computadores, no se apreciaron diferencias significativas (valor *p* de Mann-Whitney del 0,19) al emplear distintos grados de anonimato en las actividades de evaluación (ciegos simples y dobles).

Un aspecto interesante a destacar es el hecho de que se utilizó el Taller de Moodle como medio de implementar las evaluaciones en las asignaturas que mostraron, en término general, las mayores desviaciones. Por su parte, Arquitectura de Computadores y Sistemas en Tiempo Real optaron por implementaciones más sencillas y personalizadas. Si bien el Taller de Moodle proporciona una solución integral al proceso de evaluación, esta herramienta incorpora elementos que pueden introducir cierta variabilidad en las calificaciones (por ejemplo, la separación entre calificación del envío y calificación de la evaluación). También es preciso considerar que las asignaturas que usaron esta herramienta implementaron por primera vez evaluación por pares en esta fase 2. Esta combinación de hechos motiva la casuística que dio lugar a las diferencias observadas.

### 3.3. Impacto de las metodologías offline y online

A la hora de elegir entre estrategias de evaluación por pares offline u online, nuestra experiencia apunta a que el profesor debe considerar una serie de aspectos de distinta naturaleza que pueden influir en el proceso. Cada aproximación presenta una serie de ventajas e inconvenientes de acuerdo a los siguientes aspectos:

- *Confianza en la evaluación*: en nuestro estudio, los estudiantes, mediante comunicación personal al profesor, mostraron una mayor confianza en el trabajo de los revisores cuando se realizaron

evaluaciones offline en clase. Al realizarse presencialmente, los estudiantes podían confirmar visualmente que los revisores estaban efectivamente realizando su trabajo, evaluando el trabajo mediante las rúbricas. Por el contrario, las evaluaciones online dieron lugar a un grado menor de confianza y una mayor incertidumbre sobre el proceso de evaluación, al perderse la referencia visual (los estudiantes no sabían si los revisores estaban siguiendo las indicaciones de la rúbrica, cuánto tiempo dedicaban a la evaluación, etc.).

- *Resolución de dudas*: las dudas que surgían durante la evaluación (sobre los trabajos a evaluar, las rúbricas, etc.) se resolvían in situ de una manera rápida y efectiva en el caso de las evaluaciones offline, al estar el profesor físicamente en clase. En la alternativa online, la resolución de dudas tendía a ser llevada a cabo mediante comunicación escrita (por ejemplo, e-mails), lo cual tenía una repercusión temporal en comparación a la comunicación oral.
- *Impacto en el tiempo de clase*: las evaluaciones offline requerían invertir cierto tiempo de la clase en la asignación y evaluación de las tareas para cada actividad. Esto implicó que, los días en que se realizaba evaluación por pares, se impartía un menor número de contenidos al haber menos tiempo de clase disponible. Por su parte, las metodologías online tuvieron un impacto menor en el tiempo de clase, al poderse diseñar las actividades para ser realizadas fuera del horario de clase.
- *Gestión de grupos*: en asignaturas con un número elevado de estudiantes, la aproximación online representó una estrategia más eficiente dada la flexibilidad de las plataformas tecnológicas de evaluación. Por ejemplo, la asignación de evaluadores a cada envío podía realizarse automáticamente según unos parámetros de configuración establecidos por el profesor. El manejo de grupos grandes tiende a ser más laborioso en el caso offline.

## 4. Conclusiones

Este trabajo ha examinado la evaluación por pares en sus vertientes offline y online para dar soporte a los procesos de evaluación en el ámbito de la Ingeniería Informática y las Tecnologías de la Información. En esta experiencia docente, de cinco cursos académicos, se consideraron un total de siete asignaturas de distinta naturaleza que abarcan distintas etapas del desarrollo del grado. Este estudio se encuadró en un contexto dinámico, cubriendo los años pre y post-pandemia, que permitió evaluar distintas aproximaciones en dos fases: una primera fase centrada en la evaluación por pares presencial en clase y una segunda fase donde las acti-

vidades de evaluación se realizaron de manera online.

Los resultados de la fase 1 mostraron que la evaluación por pares representa una herramienta útil y precisa para apoyar el proceso de enseñanza-aprendizaje. En concreto, se dieron correlaciones estadísticas entre las calificaciones otorgadas por los revisores y las notas que hubiera asignado el profesor en todos los escenarios estudiados. Los resultados también apuntaron a que la asistencia a clase y otros factores externos (por ejemplo, el contexto epidemiológico) repercutían en el rendimiento observado para una misma asignatura. Es más, a medida que los estudiantes progresaban en sus estudios, su rendimiento bajo evaluación por pares tendía a seguir patrones similares para distintas asignaturas. Los resultados de la fase 2 indicaron que el cambio de evaluaciones presenciales a online no introducía diferencias marcables en la precisión de los revisores, para asignaturas con experiencia previa en evaluación por pares. Finalmente, se han identificado beneficios y limitaciones de las alternativas offline y online según cuatro aspectos: confianza en la evaluación, resolución de dudas, impacto en clase y gestión de grupos.

Nuestras líneas de trabajo futuro están destinadas a investigar otros sistemas de evaluación y estrategias de aprendizaje en Ingeniería Informática. En particular, examinaremos el impacto de adoptar estrategias como la gamificación y el aprendizaje basado en proyectos para mejorar el interés en la asignatura y la participación de los estudiantes. Se estudiarán también distintas herramientas como el uso de recursos interactivos H5P y tecnologías como ExeLearning para verificar su utilidad en asignaturas de naturaleza heterogénea.

## Agradecimientos

Este trabajo, respaldado por la AEI (Agencia Estatal de Investigación, España), el MICIU (Ministerio de Ciencia, Innovación y Universidades, España) y el FEDER (Fondo Europeo de Desarrollo Regional, Unión Europea), es parte del proyecto PID2022-137275NA-I00 (proyecto X-BIO) financiado por MICIU/AEI/10.13039/501100011033 y por FEDER, UE.

## Referencias

- [1] Chie Adachi, Joanna Hong-Meng Tai y Phillip Dawson. Academics' perceptions of the benefits and challenges of self and peer assessment in higher education. *Assessment & Evaluation in Higher Education*, 43(2):294–306, 2018.
- [2] Ching-Yi Chang, De-Chih Lee, Kai-Yu Tang y Gwo-Jen Hwang. Effect sizes and research directions of peer assessments: From an integrated perspective of meta-analysis and co-citation network. *Computers & Education*, 164:104123, 2021.
- [3] Martin Dougiamas y Peter C. Taylor. Moodle: Using learning communities to create an open source course management system. En *2003 World Conference on Educational Multimedia, Hypermedia and Telecommunications*, pp. 1–8. Chesapeake, VA, USA, 2003.
- [4] Wilhelm A. Friess y Andrew J. Goupee. Using Continuous Peer Evaluation in Team-Based Engineering Capstone Projects: A Case Study. *IEEE Transactions on Education*, 63(2):82–87, 2020.
- [5] Francisco José García-Peñalvo, Alfredo Corell, Víctor Abella-García y Mario Grande de Prado. Recommendations for Mandatory Online Assessment in Higher Education during the COVID-19 Pandemic. En *Radical Solutions for Education in a Crisis Context: COVID-19 as an Opportunity for Global Learning*, pp. 85–98. Springer Singapore, Singapore, 2021.
- [6] María del Carmen Iglesias Pérez, Juan Vidal-Puga y Margarita Rosa Pino Juste. The role of self and peer assessment in Higher Education. *Studies in Higher Education*, 47(3):683–692, 2022.
- [7] Laura Ketonen, Markus Hähkiöniemi, Pasi Nieminen y Jouni Viiri. Pathways Through Peer Assessment: Implementing Peer Assessment in a Lower Secondary Physics. *International Journal of Science and Mathematics Education*, 18:1465–1484, 2020.
- [8] Hongxia Li, ChengLing Zhao, Taotao Long, Yan Huang y Fengfang Shu. Exploring the reliability and its influencing factors of peer assessment in massive open online courses. *British Journal of Educational Technology*, 52(6):2263–2277, 2021.
- [9] Meixiu Lu y Ming Ming Chiu. Do Teamwork Guidelines Improve Peer Assessment Accuracy or Attitudes During Collaborative Learning? *IEEE Transactions on Education*, 65(4):485–492, 2022.
- [10] Sasha Nikolic, David Stirling y Montserrat Ros. Formative assessment to develop oral communication competency using YouTube: self- and peer assessment in engineering. *European Journal of Engineering Education*, 43(4):538–551, 2018.
- [11] David J. Sheskin. *Handbook of Parametric and Nonparametric Statistical Procedures*. 5th edition. Chapman & Hall/CRC, NY, USA, 2011.
- [12] Tarun Yellamraju, Alejandra J. Magana y Mireille Boutin. Investigating Students' Habits of Mind in a Course on Digital Signal Processing. *IEEE Transactions on Education*, 62(4):312–324, 2019.