

DESARROLLO DE HABILIDADES EN PROGRAMACIÓN A TRAVÉS DE PROYECTOS INDIVIDUALES: POTENCIANDO LA AUTONOMÍA Y LA CREATIVIDAD DEL ESTUDIANTE

Belén Vega-Márquez; Beatriz Pontes-Balanza; José María Luna-Romera

Departamento de Lenguajes y Sistemas Informáticos. Universidad de Sevilla

1. INTRODUCCIÓN

La educación en programación informática ha experimentado una transformación radical en los últimos años. Esta evolución ha llevado a la reevaluación de las estrategias pedagógicas tradicionales en favor de enfoques más centrados en el estudiante y orientados hacia el desarrollo de habilidades prácticas. En este contexto dinámico y en constante cambio, la práctica de fomentar el aprendizaje activo y el desarrollo de habilidades en materias de programación mediante proyectos individuales se ha destacado como una estrategia no solo altamente efectiva, sino también esencial para preparar a los estudiantes para los desafíos de la era digital.

En este contexto de cambio constante, el enfoque en la realización de proyectos individuales se destaca como una estrategia pedagógica que no solo satisface las necesidades actuales de la industria, sino que también apoya a que los estudiantes asuman un papel activo en su propio aprendizaje (Rubio-Sánchez et al., 2014). A través de proyectos individuales, los estudiantes se sumergen en el mundo de la programación de manera práctica y significativa, lo que les permite aplicar los conceptos teóricos en un contexto real y experimentar los desafíos de la creación de software funcional y soluciones innovadoras.

Este capítulo se centrará en explorar a fondo la práctica de fomentar el aprendizaje activo y el desarrollo de habilidades en programación mediante proyectos individuales. Analizaremos cómo esta metodología pedagógica no solo promueve la adquisición de habilidades técnicas, sino también la autonomía, la creatividad y la resolución de problemas, que son habilidades cruciales para el éxito en la industria tecnológica y en la vida cotidiana. Además, exploraremos ejemplos concretos, mejores prácticas y resultados de investigaciones que respaldan la eficacia de este enfoque, brindando una visión holística de su impacto en la formación de los estudiantes.

En última instancia, este capítulo busca arrojar luz sobre cómo el enfoque en proyectos individuales en la enseñanza de la programación puede empoderar a los estudiantes, preparándolos no solo para ser programadores competentes, sino también para ser pensadores críticos, solucionadores de problemas y creadores innovadores en un mundo impulsado por la tecnología.

1.1. Contexto

La asignatura en la que se enmarca la buena práctica presentada es Fundamentos de Programación, una asignatura anual de 12 créditos que se imparte en el primer curso de los tres grados de Ingeniería Informática en la Escuela Técnica Superior de Ingeniería Informática. En cada cuatrimestre se imparte un lenguaje de programación diferente: Python en el primero; y Java en el segundo. Además, para cada cuatrimestre se realiza un trabajo de curso personal e individualizado. Cada uno de estos proyectos se tiene en cuenta dentro de la evaluación continua de la asignatura, aportando un punto por cuatrimestre a la nota final.

Los estudiantes tienen la oportunidad de elegir un proyecto de programación de su interés, dentro de ciertos parámetros definidos por el profesor. En concreto, cada alumno podrá escoger a principios de cada cuatrimestre la temática con la que trabajará durante el desarrollo de su proyecto. A partir de ese momento los estudiantes trabajan de forma autónoma para planificar, diseñar e implementar su proyecto a lo largo del curso, siguiendo las pautas indicadas por el profesor. Cabe destacar que el proyecto a desarrollar por el alumno no se trabajará en horario de clase, donde se trabajarán con proyectos similares en complejidad, pero no así en su tamaño.

El profesor desempeña un papel fundamental en esta práctica. En primer lugar, brinda orientación a los estudiantes tanto al inicio del proyecto como durante su desarrollo, ayudándoles a definir sus objetivos y establecer un plan de trabajo. Proporciona recursos, materiales y herramientas necesarias para el desarrollo del proyecto. Durante el proceso, el profesor actúa como un facilitador, ofreciendo apoyo técnico, resolviendo dudas y brindando retroalimentación constructiva a medida que los estudiantes avanzan en sus proyectos. Para ello, los alumnos hacen uso del horario de tutorías del profesor, así como consultas por correo electrónico o en el propio horario de clase, si la planificación de la sesión así lo permite. Además, el profesor podrá enviar feedback al alumno de su trabajo realizado en distintos hitos a través de GitHub Classroom, una de las herramientas utilizadas.

1.2. Fomento de la autonomía y la creatividad

Esta práctica tiene como objetivo potenciar la autonomía y la creatividad de los estudiantes, así como el trabajo continuo del alumno a lo largo del cuatrimestre. Les permite tomar decisiones en cuanto a los datos a utilizar, el diseño de los tipos de datos a implementar y sus funcionalidades, entre otros aspectos. Esto promueve la capacidad de resolución de problemas, el pensamiento crítico y la toma de decisiones informadas. Los estudiantes también tienen la libertad de experimentar y proponer soluciones innovadoras, lo que fomenta la creatividad en el ámbito de la programación. Por otro lado, al desarrollarse de forma incremental durante la evolución de la asignatura, y coincidiendo con los contenidos de la misma, evita el abandono temprano del alumno, ya que hace que éste se encuentre más involucrado y motivado en el desarrollo de un proyecto de su propio interés y elección.

1.3. Evaluación del proyecto

La evaluación se realiza de manera integral, teniendo en cuenta diferentes aspectos del proyecto, como la calidad del código, la funcionalidad, la usabilidad, la originalidad y la presentación final. El profesor evalúa el proyecto en etapas clave, proporcionando retroalimentación continua para guiar a los estudiantes y ayudarles a mejorar su trabajo. En concreto, se realizan tres entregas evaluables, e igualmente distanciadas, a lo largo de cada cuatrimestre. En cada una de ellas, el alumno recibe feedback del profesor, donde se le indican tanto los posibles fallos que haya podido cometer como posibles mejoras o ampliaciones a realizar. Al final del cuatrimestre, el profesor emitirá una nota numérica del trabajo del alumno, junto con una evaluación APTA o NO APTA de su proyecto, que le permitirá continuar con la evaluación continua o, en caso de ser NO APTA, tener que recurrir a la evaluación final.

La evaluación de los estudiantes está basada en múltiples aspectos clave. En primer lugar, se busca que los estudiantes adquieran habilidades prácticas en programación, mejoren su capacidad para diseñar y desarrollar proyectos de software y fortalezcan su pensamiento lógico y habilidades de resolución de problemas. Adicionalmente, se promueve la creatividad y autonomía, desafiándolos a desarrollar proyectos únicos y personalizados al aplicar el contenido de la asignatura en sus trabajos. Se fomenta el trabajo en equipo, alentando a los estudiantes a buscar asesoramiento y colaboración con sus compañeros cuando sea necesario. Además, el curso actual ha demostrado un mayor nivel de implicación y motivación por parte de los alumnos en comparación con otros cursos en los que no se implementaron estas prácticas. Se espera que este compromiso se refleje en los porcentajes de aprobación de la asignatura.

1.4. Herramientas utilizadas

En el contexto de este proyecto se utilizan diversas herramientas que facilitan la colaboración, la comunicación y la gestión de los proyectos. Entre estas herramientas se encuentran GitHub, GitHub Classroom, Discord, BlackBoard y Codeshare. La combinación de estas herramientas proporciona un entorno digital integral que respalda la práctica docente, fomenta la colaboración entre los estudiantes y el profesor, y facilita la gestión eficiente de los proyectos individuales en el ámbito de la programación.

1.4.1. Github

GitHub se utiliza como un repositorio de código compartido, permitiendo a los estudiantes almacenar, organizar y colaborar en el desarrollo de sus proyectos de manera eficiente. GitHub Classroom proporciona una plataforma integrada para la asignación y entrega de proyectos, lo que facilita el seguimiento y la evaluación por parte del profesor. Ambas herramientas han sido utilizadas con anterioridad en entornos educativos (Anbaroğlu, 2021; Feliciano et al., 2016; Zagalsky et al., 2015). En nuestra asignatura, se integra además en los entornos de programación utilizados: Visual Studio Code y Eclipse, proporcionando utilidades que permiten que los alumnos puedan compartir sus proyectos con el profesor, así como recibir comentarios de éste. Es, por tanto, el medio que utilizamos para realizar las entregas y seguimiento de los trabajos.

1.4.2. Discord

Discord se emplea como un canal de comunicación en línea, permitiendo a los estudiantes interactuar con sus compañeros y con el profesor en tiempo real (Uong et al., 2022). Además, Discord puede ser utilizado para organizar reuniones virtuales, brindar apoyo técnico y fomentar la colaboración entre los participantes (Ardiyansah et al., 2021). En nuestro contexto, hemos creado un canal compartido por todos los alumnos de cada grupo de la asignatura. Aunque la idea inicial era que los alumnos pudiesen interactuar entre ellos en cuanto al contenido de la asignatura, hemos observado que la interacción que han llevado a cabo ha sido fundamentalmente de aspectos de gestión y organización.

1.4.3. Blackboard

BlackBoard es la plataforma de gestión del aprendizaje utilizada actualmente en la Universidad de Sevilla. Proporciona un entorno virtual completo donde los estudiantes pueden acceder a materiales de clase, realizar actividades y mantenerse al tanto de los plazos (Uziak et al., 2018). En el contexto del trabajo de curso, BlackBoard se ha utilizado fundamentalmente para compartir con los alumnos información sobre los requisitos y plazos de entrega de los proyectos individuales, así como para informarles de sus notas en dichos proyectos durante la última semana de cada cuatrimestre.

1.4.4. Codeshare

Codeshare es una herramienta en línea que permite a los estudiantes compartir y colaborar en tiempo real en la edición de código. Facilita la programación conjunta, la revisión de código y la resolución de problemas en equipo, promoviendo así el aprendizaje interactivo y la mejora continua de las habilidades de programación (Barphe et al., 2021). Esta herramienta es utilizada principalmente durante las horas de clase presencial, donde los alumnos pueden compartir parte de su código desde sus equipos de forma que el resto de los participantes del aula pudieran aportar sus impresiones, pudiendo incluso realizar modificaciones sobre el código compartido.

En resumen, esta práctica docente brinda a los estudiantes la oportunidad de desarrollar habilidades en programación a través de proyectos individuales, fomentando su autonomía, creatividad y capacidad para resolver problemas. El profesor desempeña un papel de guía y facilitador, proporcionando apoyo y retroalimentación para ayudar a los estudiantes a alcanzar sus objetivos. Al finalizar, los estudiantes habrán adquirido conocimientos prácticos en programación y habrán desarrollado proyectos que reflejen su creatividad y capacidad de solución de problemas. Hemos observado que, además de las ventajas ya mencionadas, el desarrollo de esta práctica hace que se incremente la motivación del alumno por la asignatura, mejorando así el porcentaje de éxito y superación.

2. RESULTADOS

2.1 Mejoras del estudiante

Desde su implementación en la asignatura de Fundamentos de Programación, la aplicación de esta estrategia pedagógica ha revelado impactos notables y beneficios tangibles en el proceso de aprendizaje de los estudiantes. A través de un enfoque centrado en proyectos individuales, los resultados han demostrado de manera convincente que esta metodología va más allá de simplemente impartir conocimientos y, en su lugar, nutre el crecimiento integral de los estudiantes. En cuanto a los beneficios al aprendizaje de los estudiantes, hemos identificado los siguientes aspectos:

1. Desarrollo de habilidades técnicas: Uno de los logros más destacados de esta práctica ha sido la adquisición de habilidades de programación de alto nivel por parte de los estudiantes. A través de la ejecución de proyectos individuales, los estudiantes no solo han adquirido un conocimiento sólido de los conceptos fundamentales de programación, sino que también han desarrollado la capacidad de aplicar estos conceptos en situaciones prácticas y resolver problemas de manera efectiva. Esta inmersión en el proceso de desarrollo de software ha permitido a los estudiantes dominar la lógica de programación,

mejorar sus habilidades de depuración y fortalecer su comprensión de algoritmos, preparándolos para enfrentar desafíos más complejos en el futuro.

2. Autonomía y creatividad: Además de impulsar el desarrollo de habilidades técnicas, la implementación de proyectos individuales ha dado lugar a un aumento significativo en la creatividad y la autonomía de los estudiantes. Al permitirles elegir proyectos que les interesen personalmente y definir soluciones únicas para problemas específicos, esta metodología ha liberado la creatividad latente de los estudiantes. Se han observado resultados sorprendentes en forma de proyectos innovadores que van más allá de las expectativas del plan de estudios. Al mismo tiempo, esta autonomía fomenta la toma de decisiones independientes, la gestión del tiempo y la responsabilidad personal, habilidades cruciales en el mundo profesional.

3. Aprendizaje activo y mejora de la motivación: Otro beneficio clave ha sido el aumento de la motivación y el compromiso de los estudiantes. La oportunidad de trabajar en proyectos que les apasionan y que tienen un impacto real en su aprendizaje ha llevado a una mayor participación en las clases y una actitud más positiva hacia la programación. Los estudiantes se han vuelto más intrínsecamente motivados para aprender y han mostrado un mayor nivel de persistencia en la resolución de problemas, ya que se dan cuenta de la relevancia y la aplicabilidad de lo que están estudiando en su vida cotidiana y futura carrera.

2.2 Repercusiones para la Asignatura, el Departamento y el Título

La implementación de la práctica "Desarrollo de Habilidades en Programación a través de Proyectos Individuales: Potenciando la Autonomía y la Creatividad del Estudiante" en la asignatura de Fundamentos de Programación ha tenido un impacto significativo en varios aspectos clave, que repercuten en la calidad de la asignatura, el departamento académico y el título en general.

2.2.1 Calidad de la Asignatura

Esta innovadora práctica ha elevado la calidad de la asignatura de Fundamentos de Programación a un nivel superior. Proporciona una experiencia de aprendizaje más activa y práctica para los estudiantes, lo que les permite no solo comprender los conceptos teóricos, sino también aplicarlos en proyectos reales. Este enfoque activo y centrado en proyectos contribuye a una mayor satisfacción de los estudiantes, ya que ven cómo sus habilidades se traducen en resultados tangibles. Además, se espera que esta mejora en la calidad de la asignatura se refleje en mejores resultados académicos y, a largo plazo, en una mayor reputación de la asignatura en el ámbito académico.

2.2.2 Retención de la Asignatura

La adopción de proyectos individuales en la enseñanza de la programación no solo impacta en la calidad de la asignatura, sino que también tiene un efecto positivo en la retención estudiantil. La experiencia de aprendizaje más atractiva y personalizada proporcionada por esta práctica puede aumentar la motivación de los estudiantes y su compromiso con el programa en general. Los estudiantes que se sienten más conectados

con su educación tienen una mayor probabilidad de continuar sus estudios y completar su título, lo que contribuye a una mayor tasa de retención en el departamento y, por ende, al éxito del programa.

2.2.3 Habilidades de Resolución de Problemas

La realización de proyectos individuales no solo se trata de aprender programación, sino que también impulsa el desarrollo de habilidades de resolución de problemas. Los estudiantes se enfrentan a desafíos concretos y deben desarrollar la capacidad de analizar, planificar y buscar soluciones efectivas de manera independiente. Cada vez que se plantea una mejora por parte del profesor, el estudiante asume la responsabilidad de identificar y abordar las deficiencias en su proyecto, lo que contribuye significativamente a su capacidad de resolución de problemas, una habilidad crucial en cualquier campo y en la vida cotidiana.

2.2.4 Estimulación de la Creatividad e Innovación

Una de las ventajas más destacadas de esta práctica es su capacidad para estimular la creatividad y la innovación en los estudiantes. La posibilidad de elegir temas de proyectos que les interesen y diseñar soluciones únicas requiere que los estudiantes exploren nuevas ideas y busquen enfoques innovadores para resolver problemas. La autonomía para definir los ejercicios planteados también contribuye al desarrollo de su pensamiento creativo e innovador. Esta experiencia no solo los prepara para ser programadores competentes, sino también para ser pensadores críticos y solucionadores de problemas creativos en un mundo en constante evolución.

2.3 Encuestas utilizadas para medir la calidad de la propuesta

Para poder comprobar si el uso de las herramientas digitales había supuesto una mejora en el cumplimiento y ejecución de los proyectos, se hicieron unas encuestas al iniciar y finalizar el curso, tanto para Github como para Discord.

En el caso de Github, la mayoría de alumnos no conocían inicialmente qué era ni para qué servía, sin embargo, al finalizar el curso la mayoría de alumnos que contestaron a la encuesta respondieron que preferían utilizar Github frente a otras herramientas como por ejemplo Enseñanza Virtual para realizar las entregas, véase la Figura 1.

12. ¿Te parece más cómodo utilizar Github que enseñanza virtual para subir tus proyectos informáticos?

[Más detalles](#)

Información

| | |
|-----------|---|
| ● Sí | 9 |
| ● No | 4 |
| ● Regular | 3 |



Figura 27. Cuestionario final sobre el uso de Github.

Por otro lado, con respecto a Discord, se puede decir que los resultados no fueron tan satisfactorios. Muchos alumnos conocían Discord porque lo utilizaban para comunicarse con amigos en juegos online, pero preferían otro tipo de herramientas más tradicionales de

20. ¿Has usado más Discord que otro canales de comunicación para comunicarte con tus compañeros en aspectos relacionados con la asignatura?

[Más detalles](#)

Información

| | |
|------|----|
| ● Sí | 3 |
| ● No | 13 |



comunicación como Whatsapp para hablar de aspectos relacionados con los estudios, tal como se observa en la Figura 2. Lo que sí es importante recalcar es que al final del curso, la proporción de alumnos que utilizaban Discord para asuntos relacionados con los estudios era mayor que al principio.

Figura 28. Uso de Discord frente a otras herramientas.

3. CONCLUSIONES

Consideramos que la aplicación de la buena práctica presentada aquí ha demostrado ser altamente efectiva para promover el aprendizaje activo, el desarrollo de habilidades y la motivación de los estudiantes. Además, las herramientas utilizadas juegan un papel fundamental en la facilitación de la colaboración, la comunicación y la gestión de los proyectos individuales. En su conjunto, consideramos que esta práctica ha generado

resultados muy positivos, tanto en términos de adquisición de conocimientos, habilidades y motivación de los estudiantes, como en sus resultados de evaluación.

En este estudio, hemos destacado la importancia de un seguimiento efectivo de los trabajos en el entorno educativo, particularmente en el campo de la programación. Hemos explorado cómo los sistemas de comunicaciones tradicionales pueden evolucionar y adaptarse para fomentar una mayor autonomía y creatividad por parte del alumno. Además, hemos subrayado que los conocimientos adquiridos en programación no solo se limitan a la adquisición de habilidades técnicas, sino que también se traducen en un mayor empoderamiento del estudiante para abordar desafíos complejos y resolver problemas de manera innovadora. Estas conclusiones destacan la importancia de diseñar estrategias pedagógicas que fomenten la independencia, la creatividad y la adquisición de conocimientos prácticos en el ámbito de la programación, lo que, a su vez, contribuye a la formación de profesionales altamente competentes en el campo de la informática y la tecnología.

En un enfoque innovador hacia la mejora del aprendizaje, se planea como trabajo futuro implementar un sistema de autoevaluación y evaluación por pares en relación al desarrollo de las prácticas de curso. Los estudiantes serán desafiados a reflexionar críticamente sobre su propio progreso, así como evaluar el trabajo de sus compañeros. A través de esta iniciativa, se espera que los estudiantes adquieran una comprensión más completa de la materia estudiada, y mejoren sus capacidades de autoevaluación y crítica constructiva.

REFERENCIAS

- Anbaroğlu, B. (2021). A collaborative GIS programming course using GitHub Classroom. *Transactions in GIS*, 25(6), 3132–3158. <https://doi.org/10.1111/TGIS.12810>
- Ardiyansah, T. Y., Batubara, R. W., & Auliya, P. K. (2021). Using Discord to facilitate students in teaching learning process during COVID-19 outbreak. *Journal of English Teaching, Literature, and Applied Linguistics*, 5(1), 76. <https://doi.org/10.30587/JETLAL.V5I1.2528>
- Barphe, S. S., Lokare, V. T., Sutar, S. R., & Kiwelekar, A. W. (2021). Effective Online Tools for Teaching Java Programming Course on an Online Platform. *Lecture notes in networks and systems*, 2, 615–622. https://doi.org/10.1007/978-981-16-6369-7_56
- Feliciano, J., Storey, M.-A., & Zagalsky, A. (2016). Student Experiences Using GitHub in Software Engineering Courses: A Case Study. *Proceedings - 2016 38th International Conference on Software Engineering Companion, ICSE'16*, 422-431. <https://doi.org/10.1145/2889160.2889195>

- Rubio-Sánchez, M., Kinnunen, P., Pareja-Flores, C., & Velázquez-Iturbide, J. Á. (2014). Student perception and usage of an automated programming assessment tool. *Computers in Human Behavior*, *31*, 453–460. <https://doi.org/10.1016/j.chb.2013.04.001>
- Uong, T. G. T., Nguyen, D. K., & Nguyen, H. N. (2022). Teachers' Feedback on Using Discord as an Online Learning Platform. *International Journal of TESOL & Education*, *2*(4), 84–104. <https://doi.org/10.54855/IJTE.22246>
- Uziak, J., Oladiran, M. T., Lorencowicz, E., & Becker, K. (2018). Students' and instructor's perspective on the use of Blackboard Platform for delivering an engineering course. *Electronic Journal of E-Learning*, *16*(1), 1–15. <https://doi.org/10.34190/EJEL.16.1.2367>
- Zagalsky, A., Feliciano, J., Storey, M. A., Zhao, Y., & Wang, W. (2015). The emergence of GitHub as a collaborative platform for education. *Proceedings - 2015 ACM International Conference on Computer-Supported Cooperative Work and Social Computing, CSCW 2015*, 1906–1917. <https://doi.org/10.1145/2675133.2675284>