

EVALUACIÓN EN LA DISCIPLINA DE LA INFORMÁTICA

Trabajo Final de Máster

Máster Universitario en Formación del Profesorado de Educación
Secundaria Obligatoria y Bachillerato, Formación Profesional y
Enseñanzas de Idiomas

Universidad Miguel Hernández de Elche
Ricardo Candela Castro
Curso académico 2023/2024

ÍNDICE

1. Resumen y palabras clave.....	3
2. Introducción.....	4
3. Revisión bibliográfica.....	6
3.1 Metodología.....	6
3.1.1 Definición de términos de búsqueda y selección de las bases de datos.....	6
3.1.2 Criterios de selección de los artículos.....	7
3.1.3 Revisión de los artículos y extracción de los datos.....	7
3.2 Resultados.....	7
3.2.1 Descripción general de los resultados.....	8
3.2.2 Métodos e instrumentos encontrados.....	9
3.3 Discusión de los resultados.....	11
4. Propuesta.....	13
5. Conclusiones.....	15
6. Referencias.....	16
7. Anexos.....	19

I. Resumen y palabras clave

Resumen

En esta revisión bibliográfica se investigará sobre la evaluación en el ámbito de la enseñanza y el aprendizaje de la informática. Conocer las experiencias en los centros, implementada por los docentes, permitirá obtener una amplia visión del panorama actual de la evaluación en esta disciplina y así tratar de determinar en qué contextos determinados pueden ser útiles estas herramientas.

Se prestará especial atención a aquellas técnicas que se estén utilizando que permitan una evaluación automática, facilitando la tarea docente e incorporando objetividad a la evaluación. Al incluir la informática técnicas tan dispares como la programación, las bases de datos, etc, estas técnicas de evaluación pueden ser muy interesantes.

Además, se realizará una propuesta de un método e instrumento de evaluación automatizado, en la cual se especifiquen nivel, asignatura o módulo, competencias o contenidos a evaluar, actividades a evaluar y técnica elegida de evaluación, con el objetivo de desarrollar una herramienta que proporcione una evaluación lo más objetiva y formadora posible.

Abstract

This literature review will investigate the assessment in the field of teaching and learning computer science. Understanding the experiences in schools, implemented by teachers, will provide a broad view of the current landscape of assessment in this discipline and thus try to determine in which specific contexts these tools can be useful.

Special attention will be given to those techniques that are being used to allow automatic assessment, facilitating the teaching task and incorporating objectivity into the evaluation. Since computer science includes such diverse techniques as programming, databases, etc., these assessment techniques can be very interesting.

Additionally, a proposal for an automated assessment method and instrument will be made, specifying the level, subject or module, competencies or contents to be assessed, activities to be evaluated, and the chosen assessment technique, with the aim of developing a tool that provides the most objective and formative assessment possible.

Palabras clave: competencias, metodología, instrumentos, pensamiento computacional, programación

2. Introducción

En la actualidad, la especialidad de Informática en los contextos de educación secundaria obligatoria y Bachillerato se encuentra en creciente cambio. La introducción en la sociedad de las TIC ha provocado una respuesta por parte de la Administración para responder a las nuevas necesidades, modificando contenidos y competencias de las asignaturas que se imparten en esta especialidad.

En el caso de las familias profesionales de Informática, a pesar de no haber sufrido cambios recientemente en cuanto a contenidos, objetivos y resultados de aprendizaje, es responsabilidad del cuerpo docente trasladar esas nuevas necesidades de la sociedad para proporcionar una educación de calidad.

En este contexto de la educación, resulta fundamental la evaluación del estudiantado, para medir así el aprendizaje y la adquisición de competencias. La efectividad de la evaluación no sólo informa del progreso del estudiantado, sino también contribuye a mejorar las metodologías aplicadas por el personal docente y adaptar el currículo de manera iterativa para satisfacer las necesidades educativas de los grupos.

Con respecto a la evaluación en la asignaturas de la especialidad de Informática, existen diversos métodos que se pueden emplear, que van desde una evaluación más tradicional, por medio de entrega de trabajos prácticos o cuestionarios teóricos, hasta un enfoque más moderno, en el cual se utilizan listas de comprobación o incluso métodos de corrección automáticos.

La elección de los métodos e instrumentos de evaluación dependerá enteramente del contexto sobre los que se quiera realizar dicha evaluación. Debido al carácter fundamentalmente práctico de la mayoría de asignaturas y/o módulos de la especialidad, estos métodos e instrumentos seleccionados estarán centrados principalmente en evaluar el desempeño del estudiantado y la demostración de la adquisición de las competencias en tareas y actividades en las que pongan en práctica los conocimientos adquiridos.

Por otro lado, existe el caso de módulos más teóricos, o incluso contenidos dentro de módulos o asignaturas más prácticas, que requerirán de métodos e instrumentos de evaluación quizá algo más tradicionales, como la elaboración de trabajos de investigación de manera individual o en grupo o los ya mencionados cuestionarios teóricos.

El principal objetivo de este TFM consiste en tratar de revisar qué métodos e instrumentos de evaluación existen para evaluar las asignaturas de Informática, tanto en la ESO y Bachiller como en Ciclos formativos, y comprobar su popularidad y adecuación a los contenidos o competencias a evaluar.

Además, se tratará de encontrar diferencias entre los distintos niveles educativos para comprobar cómo varía la evaluación en función de los diferentes factores que influyen en estos niveles educativos.

Por otro lado, se comprobará si en la actualidad se están llevando a cabo evaluaciones mediante métodos de corrección automáticos y existe constancia de ello. En el caso de encontrar resultados, se indagará en las asignaturas, módulos o partes de estos que los implementen, para intentar responder a por qué se emplean en estos en concreto.

A priori, actividades prácticas en las que el producto parcial o final sea un fragmento de código o similar son las que más se prestan a introducir estos métodos de corrección automáticos, pero conviene revisar si es así o existen otras posibilidades.

Por último, se intentará realizar una propuesta de introducción, en aquellas asignaturas, módulos o partes de ellos en los que no se haya encontrado en la literatura uso alguno de estos sistemas de corrección automáticos, de un método o instrumento automatizado que permita evaluar dichos contenidos o competencias de una manera rápida, eficiente y objetiva.

Para concretar estos objetivos, las preguntas que se van a intentar responder en esta revisión bibliográfica son las siguientes:

- ¿Qué métodos de evaluación existen actualmente para la evaluación de la asignatura de Informática? ¿Cuáles son los más populares o utilizados?
- ¿Cómo de efectivos/eficaces son en comparación con los otros métodos?
- ¿Existen diferencias entre los métodos de evaluación utilizados según el nivel de estudios a evaluar? (ESO, BACH, FP)
- ¿Varían las técnicas de evaluación en función de las competencias o contenidos a evaluar?
- ¿Se están empleando técnicas de evaluación automáticas en alguna asignatura de la especialidad? Si es así, ¿en qué asignaturas y/o competencias se utilizan? ¿Por qué en estas?
- ¿Podrían aplicarse las técnicas de evaluación automática a otras áreas del ámbito de la Informática en las que no estén utilizándose?

3. Revisión bibliográfica

3.1 Metodología

Para realizar esta revisión bibliográfica, se siguieron los siguientes pasos: (1) se plantearon las preguntas a responder; (2) los términos de búsqueda se definieron, y se seleccionaron las bases de datos apropiadas; (3) se seleccionaron los artículos que fueran relevantes y pudieran ser útiles para la revisión; (4) los artículos se analizaron en busca de información que contribuyera a responder a las preguntas planteadas; (5) se extrajeron los datos para poder responder a las preguntas.

3.1.1 Definición de términos de búsqueda y selección de las bases de datos

En esta revisión se han consultado las siguientes bases de datos: ERIC, Dialnet, Google Scholar, Science Direct y IJCSSES. La selección de estas bases de datos incluye publicaciones sobre investigación en educación e investigación en ciencias computacionales.

Los términos utilizados en las búsquedas fueron los siguientes:

- Primera consulta: “informática” Y (“secundaria” O “educación” O “primaria” O “bachillerato”) Y “evaluación”
- Segunda consulta: “informática” Y (“secundaria” O “educación” O “primaria” O “bachillerato”) Y (“evaluación” O “test” O “prueba” O “examen” O “medida”)
- Tercera consulta: “informática” Y “evaluación”
- Cuarta consulta: (“pensamiento computacional” O “informática”) Y (“secundaria” O “educación” O “primaria” O “bachillerato” O “superior”) Y (“evaluación” O “test” O “prueba” O “examen” O “medida”)

En todas las consultas se tradujeron al inglés también los términos de búsqueda, para tratar de obtener la mayor cantidad de resultados posibles.

Tras realizar la primera consulta, se descubrió que a menudo se empleaban palabras como test, examen, prueba o medidas como sinónimos, por lo que en la segunda consulta se incluyeron también estos términos. Las dos primeras consultas, sin embargo, no contemplaban otros sistemas educativos diferentes a los españoles, por la posibilidad de llamarse de forma distinta los niveles educativos en cada sistema. Por ello, se decidió hacer una búsqueda más amplia eliminando el término relativo al nivel educativo. A pesar de que esta consulta recogía muchos resultados que estaban fuera del alcance de esta revisión, se decidió conservar los resultados y filtrarlos, para asegurar que no se excluyeran artículos que utilizaran otra terminología.

Por último, ante la dificultad de encontrar suficientes resultados que fueran realmente relevantes para esta revisión, se incluyó, para los niveles educativos, la educación superior. Además, el término “pensamiento computacional” se añadió acompañando a “informática”, debido a que es una de las competencias más trabajadas

e importantes de la disciplina. Esta inclusión dio sus frutos, y arrojó más resultados relevantes, ampliando el número de artículos de la revisión.

3.1.2 Criterios de selección de los artículos

Para seleccionar los artículos, se siguieron los siguientes criterios:

- Se trató de que los resultados fueran actuales, con la excepción de algunos casos, en los que los resultados se pueden considerar algo obsoletos (más de cinco años) pero abordan temas interesantes, por lo que se decidió conservarlos.
- Se estableció la siguiente prioridad a la hora de la selección, en función del país de origen, dada la relevancia de la procedencia de estos por los diferentes contextos educativos: (1) artículos nacionales; (2) artículos publicados sobre países europeos; (3) artículos publicados del resto del mundo.
- En cuanto al idioma del artículo, inicialmente se escogieron artículos en español, pero se hizo extensivo el criterio a español e inglés, para ampliar el número de resultados.
- El artículo explica qué se está evaluando.
- El artículo presenta una herramienta o método para evaluar en el contexto de la Informática.

3.1.3 Revisión de los artículos y extracción de los datos

Una primera selección se realizó atendiendo a los títulos y resúmenes de los artículos para comprobar su adecuación al tema de la revisión. A continuación, se aplicaron los criterios de selección para eliminar los artículos que no cumplieran con ellos.

Los artículos que se seleccionaron, se leyeron de nuevo más en profundidad para extraer los datos que fueran relevantes para la revisión. Entre estos datos se consideraron:

- País en el que se desarrolló el artículo o estudio
- Nombre y forma de la herramienta o método de evaluación
- Contenido o competencia a evaluar
- Nivel o etapa educativa en la que se enmarca el artículo

3.2 Resultados

La presentación de los resultados se dividió en dos puntos principales: descripción de los resultados seleccionados y presentación de los diferentes métodos e instrumentos encontrados.

3.2.1 Descripción general de los resultados

Se puede hacer una distinción clara de todos los resultados obtenidos, atendiendo al nivel educativo, distinguiendo si el estudio o artículo se centra en educación primaria o secundaria o si, por el contrario, está más enfocado en niveles educativos superiores.

Los que se centran en niveles superiores principalmente son propuestas de evaluación innovadora o evaluación automática más que una reflexión o análisis de los mecanismos de evaluación utilizados.

Se encuentra una gran variedad de métodos y herramientas entre los artículos analizados para la evaluación de la informática en primaria y secundaria.

Rowe et al. (2021) utilizan puzzles interactivos para medir el grado de pensamiento computacional alcanzado por el alumnado comprendido entre los cursos de 3º de primaria a 2º de la ESO: incluyen descomposición de problemas y reconocimiento de patrones, capacidad de abstracción y diseño de algoritmos.

Otros, como Fagerlund et al. (2020) utilizan el análisis de proyectos de Scratch en estudiantes de 4º de primaria, evaluando patrones seguidos por el estudiantado y los procesos de desarrollo de código para aplicaciones de animación y tratamiento de datos. Allsop (2019) utiliza también aplicaciones como Scratch y Alice, para la creación de juegos en alumnado de 1º de la ESO, y propone un modelo de evaluación múltiple.

Aparecen también menciones a aplicaciones gamificadas y gamificación, como presentan Morales y Marrero (2020), que utilizan elementos como límites de tiempo, puntuaciones y tablas de clasificación con el objetivo de mejorar la motivación y el rendimiento del alumnado. Andreu (2022) destaca el uso de herramientas digitales como Quizizz y Kahoot! para evaluar la motivación y el rendimiento en la informática en entornos gamificados, y apunta que la novedad de estas herramientas incrementa la motivación y en rendimiento, pero puede reducir su efectividad a largo plazo, al perder este aspecto novedoso.

Babazadeh y Negrini (2022), en su revisión sistemática de la evaluación del pensamiento computacional en la educación K-12 en Europa, destacan herramientas de evaluación automáticas como Dr. Scratch para el análisis de proyectos de Scratch, que evalúan habilidades de programación a través de la implementación de algoritmos de evaluación automática.

Con respecto a la educación superior, aparecen por un lado artículos que se centran en el análisis de la evaluación y, por otro, propuestas de innovación de la evaluación, tanto supervisada por el docente como automatizadas.

Petri y Von Wangenheim (2017) revisan la literatura sobre la evaluación de juegos educativos en la informática, encontrando modelos de evaluación utilizados como MEEGA y EGameFlow, e instrumentos como cuestionarios y entrevistas para medir factores como el aprendizaje y la motivación en el alumnado.

En lo relativo a la evaluación automatizada en educación superior, Combéfis (2022) discute técnicas y herramientas de evaluación automática de código, como análisis estáticos y dinámicos, y el uso de inteligencia artificial para evaluar aspectos del código como sintaxis, plagio y rendimiento. Del Gobbo et al. (2023) investigan la evaluación automática de preguntas abiertas, utilizando técnicas de procesamiento de lenguaje natural y aprendizaje automático para proporcionar retroalimentación al modelo y así mejorar la calidad de la evaluación.

Por último, en cuanto a las propuestas de evaluación en niveles superiores, se encuentra una propuesta de evaluación alternativa a la más tradicional que se estaba llevando a cabo en una asignatura universitaria por Viudes y Fernández (2011). Esta propuesta promueve el uso de portfolios y trabajos prácticos en el dispositivo, para favorecer una evaluación integral y continua del aprendizaje.

Izquierdo et al. (2018) destacan la automatización de la corrección de tareas prácticas de programación a través de herramientas TIC, agilizando el proceso y mejorando la eficiencia y precisión en la evaluación.

Del Río et al. (2014) implementan pruebas unitarias automatizadas con JUnit para realizar una evaluación más coherente y rápida, priorizando la evaluación, además del contenido, del funcionamiento de los proyectos evaluados.

3.2.2 Métodos e instrumentos encontrados

Después del análisis de todos los artículos, se pueden clasificar los principales métodos e instrumentos que se han encontrado de la siguiente manera:

- Pruebas/tareas: la mayoría de los artículos consultados utilizan pruebas o tareas para evaluar las competencias de las asignaturas de informática, y en concreto el pensamiento computacional en muchos casos.

Se han encontrado test concretos diseñados específicamente para evaluar las habilidades de programación, como el Test de Computación Creativa con Bloques Visuales. Esta prueba comprende 40 ítems, en los cuales el alumnado va respondiendo a preguntas relacionadas con secuencias, bucles, sentencias condicionales, ejecución en paralelo, coordinación, manejo de eventos y entrada de teclado.

Otra prueba utilizada es el test de pensamiento computacional, dirigida a alumnado de secundaria (Roman-Gonzalez et al., 2017), en la que se deben resolver 28 tareas, como por ejemplo, se proporciona una secuencia de instrucciones y se debe decidir el número de veces a ejecutar la secuencia para mover un personaje entre dos puntos en una cuadrícula.

Otras tareas que se han encontrado para evaluar en informática son las tareas Bebras. Estas son unas tareas utilizadas en el International Challenge on Informatics and Computational Thinking. El objetivo del desafío es aumentar el

compromiso de los alumnos con la informática y promover el desarrollo del pensamiento computacional mediante la resolución de problemas atractivos y de la vida real (Chiazzese et al., 2018).

Por otro lado, se han utilizado las tareas de code.org para evaluar la informática. La plataforma ofrece diferentes cursos en línea para estudiantes con tareas de evaluación para resolver.

En otros casos, los investigadores utilizaron tareas creadas con diferentes lenguajes de programación, como pueden ser Scratch, Alice o Java.

En algunos casos, como en el estudio de Bryndová y Mališů (2020), se utilizaron tareas de robótica educativa que, por ejemplo, utilizaban los robots Ozobot EVO y BIT.

- Cuestionarios: los cuestionarios relacionados con el PC también se utilizan a menudo para evaluar las habilidades en informática y pensamiento computacional. Salas, J. (2010) emplea cuestionarios en la evaluación en el contexto de una aplicación informática implantada para evaluar el trabajo en equipo.

Petri, G. y Von Wangenheim, C. (2017) encontraron en su revisión sistemática sobre la gamificación en la educación de la informática que la mayoría de los estudios que analizaron utilizaban cuestionarios para recolectar datos y así poder realizar la evaluación.

También Babazadeh, M. y Negrini, L. (2022) descubrieron en el análisis de su revisión sistemática sobre el pensamiento computacional en la informática en educación obligatoria en Europa, que los cuestionarios son una de las herramientas evaluadoras más utilizadas. Los cuestionarios presentes en los estudios que analizaron estaban en su mayoría desarrollados por el profesorado, y destacaban dos de ellos que utilizaban una escala Likert de 5 puntos, yendo de 1 = "Totalmente en desacuerdo" - 5 = "Totalmente de acuerdo".

En su revisión sistemática sobre propuestas de gamificación en diferentes disciplinas educativas, en la disciplina de informática, Andreu, J. (2022) menciona los cuestionarios como una de las principales herramientas utilizadas, destacando además una correlación significativa entre la realización de estos cuestionarios y la motivación del alumnado.

- Observaciones: en algunos estudios, las habilidades propias de la informática y relacionadas con el pensamiento computacional se han evaluado a través de la observación del alumnado en la resolución de diferentes tareas. Por ejemplo, Allsop (2019) recopiló datos sobre habilidades de pensamiento computacional, y observó "el lenguaje que los niños y niñas usaban para sus explicaciones y

discusiones en grupo, los gestos, el contexto de sus relaciones con el profesor, compañeros y tecnología en el entorno de aula".

- Entrevistas: para la evaluación en informática y del pensamiento computacional, en algunos estudios se solicitó al alumnado que creara un producto (en este caso una historia codificada) para que posteriormente explicaran las historias que habían codificado en una entrevista con el profesorado responsable. (Price & Price-Mohr, 2018).
- Análisis de productos: otro método empleado en la evaluación de las habilidades del alumnado en la disciplina de informática es analizar los proyectos generados por el alumnado. Un ejemplo de esto son los proyectos creados en Scratch. Los productos pueden ser analizados manualmente o automáticamente con Dr. Scratch, por ejemplo. Dr. Scratch es una herramienta de análisis en línea que puede evaluar las habilidades de pensamiento computacional de un proyecto en Scratch basado en el número de bucles, sprites, bloques y otros ítems utilizados en el proyecto, y calcular una puntuación de habilidades computacionales de acuerdo a ello. Sin embargo, como indican Moreno-León y Robles (2015), Dr. Scratch tiene algunas limitaciones, ya que no puede detectar si el programa funciona como se pretende, es decir: un proyecto con los bloques, bucles, etc. apropiados podría obtener una alta puntuación en habilidades computacionales, aunque podría carecer de la funcionalidad esperada.

3.3 Discusión de los resultados

A partir de los resultados obtenidos, se concluye que existe una gran variedad de herramientas de evaluación para las asignaturas y las competencias de la disciplina informática (Çoban, & Korkmaz, 2021) y se extrae que las herramientas encontradas se pueden categorizar en cinco grupos principales: observación, pruebas y tareas, cuestionarios, entrevistas y análisis de productos.

La mayor parte de los artículos emplean únicamente una herramienta de evaluación, y se centran principalmente en la evaluación de algoritmos y habilidades de programación (Tang et al. 2020), dejando fuera otras posibles competencias o contenidos también relevantes en esta disciplina.

La necesidad de herramientas que permitan evaluar todas las facetas de la informática, y no solamente estas habilidades de pensamiento computacional ha sido mencionada anteriormente por Basu, Rutstein, Xu, Wang y Shear, 2021.

Tras analizar las herramientas mostradas en los estudios, se extrae de este análisis que no existe una concreción clara de a qué niveles en concreto se deben evaluar unas competencias u otras, y que grado de competencia ha de alcanzar el

alumnado en cada uno de los niveles, dado que se utilizan herramientas similares para alumnado de diferentes niveles.

De entre la literatura obtenida, se observa que en niveles educativos inferiores se utilizan métodos e instrumentos de evaluación más tradicionales, y a medida que aumenta la edad del alumnado, se trata de innovar más, con el objetivo de motivar y alcanzar una evaluación más formadora.

Es en los niveles educativos más altos, a partir de secundaria y, sobretudo, en educación superior y universitaria, donde se encuentra un mayor interés por automatizar las tareas de evaluación, opción más atractiva por dos motivos principales: la presencia de código y algoritmos es más acusada y se presta a este tipo de herramientas, y los grupos suelen ser más numerosos, facilitando así la tarea del profesorado.

Por otro lado, se ha de destacar que no se ha podido obtener información acerca de las familias profesionales de Informática, a pesar de ser un nivel educativo de transición entre la etapa de educación obligatoria y los estudios superiores, siendo un escenario ideal para poder implementar estrategias de evaluación automatizada.



4. Propuesta

Tras la revisión de la bibliografía encontrada, resulta evidente la falta de literatura referida a la formación profesional, por lo que se justifica que la propuesta se sitúe en este nivel educativo.

Además, este nivel educativo se aproxima más a la educación superior, nivel en el cual se han encontrado la mayoría de propuestas innovadoras tratando de objetivar y automatizar los procesos evaluadores.

El contexto en el cual se va a desarrollar la propuesta es en la asignatura de Administración de Sistemas Gestores de Bases de Datos, módulo que se encuentra dentro de los módulos impartidos en el segundo curso del ciclo formativo superior de Administración de Sistemas Informáticos en Red. Esta asignatura, debido a sus características, y los contenidos que comprende, se presta a proponer y utilizar una herramienta de corrección automática.

El producto a evaluar será un proyecto final que englobe todos los contenidos de bases de datos prácticos en un solo archivo con extensión .sql, que será la entrada para la herramienta desarrollada, de manera que se pueda determinar la puntuación del alumnado.

Este proyecto final, servirá a modo de refuerzo y culminación de la parte práctica del módulo, y la presencia de código, se presta a poder realizar estas correcciones.

El instrumento utilizado para evaluar será la propia herramienta desarrollada. En primer lugar, se van a definir los criterios para evaluar la correcta realización del proyecto final. Los contenidos que abordará este proyecto final serán:

- Creación de bases de datos.
- Creación de tablas con sus campos y claves primarias y secundarias, dentro de la base de datos previamente creada, con sus relaciones correctamente creadas.
- Creación de índices que favorezcan la consulta posterior de las tablas creadas.
- Diseño e implementación de rutinas que pueblen la base de datos y las tablas de los apartados anteriores.
- Configuración de permisos y roles de usuario para la base de datos y las tablas, de acuerdo con unos criterios previamente establecidos.
- Establecimiento de eventos y disparadores que actuarán según se produzcan las acciones especificadas en dichos ítems.

Se comprobará que cada uno de los apartados mencionados previamente se encuentre en el archivo entregado por el alumnado y, además, se atenderá a la sintaxis, obteniendo la máxima puntuación por apartado en los casos en los que la sintaxis esté perfecta.

Esta sería una primera fase en la que se podría aplicar la automatización para la evaluación en este módulo en concreto. Sin embargo, esta evaluación sería incompleta,

ya que se centra en la calidad del código, pero no se incluye el funcionamiento tras la ejecución del código.

Por esto, en una segunda fase para abordar la evaluación se centraría en la comprobación del funcionamiento de la base de datos creada a través de la ejecución del fichero, mediante pruebas unitarias automatizadas.

Es decir, utilizando como ejemplo la creación de la base de datos, se probaría, por un lado, que la sintaxis para la creación de esa base de datos es la adecuada y, por otro, que esa base de datos, una vez ejecutada su sentencia de creación, sea accesible y se pueda utilizar.

Para la creación del algoritmo de corrección, se va a trazar una rúbrica, que valore todos los elementos que comprenderá el producto de manera individual, y así poder traducir los diferentes grados de consecución de los objetivos del proyecto a decisiones que el algoritmo de evaluación tomará.

Dicha rúbrica se compartiría con el alumnado al presentar el proyecto, de manera que la evaluación sea lo más formadora posible, y conozcan así desde el primer momento cómo se les va a evaluar.

La rúbrica comprenderá una evaluación cualitativa, definiendo los objetivos en distintos niveles de adquisición, y una cuantitativa, en la que el grado que se haya alcanzado en cada uno de los objetivos se traducirá en una puntuación numérica sobre el total de la puntuación del proyecto.

5. Conclusiones

En esta revisión se ha tratado de responder a preguntas clave sobre los métodos de evaluación existentes que se emplean en las asignaturas y módulos de la disciplina de la informática, su efectividad y la implementación de técnicas que realicen esta evaluación de forma automatizada.

Se han identificado diversidad de métodos e instrumentos de evaluación, tales como tareas, cuestionarios, observación, entrevistas y análisis de productos. Se ha observado que estos métodos intentan adaptarse a los diferentes niveles educativos y tipos de contenido, no siempre ajustándose de forma clara a las competencias específicas de cada nivel.

Hay un creciente interés por el uso de técnicas de evaluación automatizadas, motivado por el ritmo exponencial al que se demandan las nuevas tecnologías y la competencia en su manejo, lo que motiva una constante actualización de currículos para dar cabida a estas necesidades. Se diferencian los niveles educativos de primaria y secundaria de los superiores en que, en la educación obligatoria, se ha encontrado que esta evaluación automatizada se potencia a través de aplicaciones ya existentes en el mercado que permiten este tipo de evaluación; sin embargo, en la educación superior, destacan las propuestas creadas específicamente para las necesidades de las materias en las que se aplican, utilizando técnicas avanzadas como inteligencia artificial o implementación de pruebas unitarias.

A pesar de la abundante literatura sobre los niveles educativos de educación primaria, secundaria y estudios universitarios, se ha identificado una gran carencia de estudios relacionados con la formación profesional. Esto es lo que ha motivado que la propuesta realizada a partir de la revisión bibliográfica se haya enmarcado en este nivel educativo, con la intención de aprovechar ese hueco y proponer la evaluación automatizada para un módulo de un ciclo formativo superior que, por su proximidad a la etapa universitaria, posee unos contenidos más maduros y complejos, que posibilitan y justifican la aplicación de estas herramientas automáticas.

Los hallazgos de la revisión pueden servir como guía para docentes de la especialidad de informática para la aplicación de diferentes procedimientos de evaluación, trasladando al aula nuevas formas de evaluar las competencias.

Para investigaciones futuras, se propone realizar una revisión sistemática, con el objetivo de conseguir una investigación más exhaustiva de las últimas tendencias en educación en informática, justificada por el ritmo vertiginoso al que cambia la demanda del mercado. Además, se ha descrito una primera fase teórica de la propuesta. Se recomienda desarrollar esta propuesta más en profundidad, quizá como un proyecto aislado, y explorar sus dos vertientes o fases, tanto la comprobación de la corrección del código como la comprobación de su funcionamiento.

6. Referencias

- Allsop, Y. (2019). Assessing computational thinking process using a multiple evaluation approach. *International Journal Of Child-computer Interaction*, 19, 30-55. <https://doi.org/10.1016/j.ijcci.2018.10.004>
- Andreu, J. M. P. (2022). *Revisión sistemática sobre la evaluación de propuestas de gamificación en siete disciplinas educativas*. Dialnet. <https://dialnet.unirioja.es/servlet/articulo?codigo=8136411>
- Babazadeh, M., & Negrini, L. (2022). How is computational thinking assessed in European K-12 education? A systematic review. *International Journal Of Computer Science Education In Schools*, 5(4), 3-19. <https://doi.org/10.21585/ijcses.v5i4.138>
- Basu, S., Rutstein, D. W., Xu, Y., Wang, H., & Shear, L. (2021). A principled approach to designing computational thinking concepts and practices assessments for upper elementary grades. *Computer Science Education*, 31(2), 169-198. <https://doi.org/10.1080/08993408.2020.1866939>
- Bryndová, L., & Malíšů, P. (2020). Assessing the Current Level of the Computational Thinking Within the Primary and Lower Secondary School Students Using Educational Robotics Tasks. *2020 the 4th International Conference on Education and Multimedia Technology*, (pp. 239–43). ICEMT 2020. New York, NY, USA: Association for Computing Machinery. <https://doi.org/10.1145/3416797.3416819>
- Çoban, E., & Korkmaz, Ö. (2021). An alternative approach for measuring computational thinking: Performance-based platform. *Thinking Skills and Creativity*, 42, 100929. <https://doi.org/10.1016/j.TSC.2021.100929>
- Combéfis, S. (2022). Automated Code Assessment for Education: Review, Classification and Perspectives on Techniques and Tools. *Software*, 1(1), 3-30. <https://doi.org/10.3390/software1010002>
- Del Gobbo, E., Guarino, A., Cafarelli, B., Grilli, L., & Limone, P. (2023). Automatic evaluation of open-ended questions for online learning. A systematic mapping. *Studies In Educational Evaluation*, 77, 101258. <https://doi.org/10.1016/j.stueduc.2023.101258>
- Del Río, A. C., Rodríguez, M. G., Fernández, N. G., & Rodríguez, B. M. G. (2014). *Uso de JUnit para evaluación en laboratorio de Estructuras de Datos*. Dialnet. <https://dialnet.unirioja.es/servlet/articulo?codigo=7368151>
- Fagerlund, J., Häkkinen, P., Vesisenaho, M., & Viiri, J. (2020). Assessing 4th Grade Students' Computational Thinking through Scratch Programming Projects. *Informatics In Education*, 19(4), 611-640. <https://doi.org/10.15388/infedu.2020.27>

- Izquierdo, C. E., Plata, I. L., Batista, M. B. M., & Vega, J. M. M. (2018). *Automatización de los procesos de corrección y autoevaluación de prácticas en asignaturas con contenidos de programación mediante herramientas TIC*. Dialnet. <https://dialnet.unirioja.es/servlet/articulo?codigo=6509302>
- Monter, I. B., Muñoz, J. J. G., & Gómez, N. A. (2018). *Evaluación de la calidad del aprendizaje de la Informática. Una experiencia desde la gestión académica*. <https://www.redalyc.org/journal/4780/478055151006/html/>
- Morales, D. V., & Marrero, L. G. (2020). *Aplicación de software gamificada en la asignatura Teleinformática*. Dialnet. <https://dialnet.unirioja.es/servlet/articulo?codigo=8573499>
- Moreno-León, J., & Robles, G. (2015). Analyze Your Scratch Projects with Dr. Scratch and Assess Your Computational Thinking Skills. *Proceedings of the 7th international Scratch conference (Scratch2015AMS)*, (pp. 12–15). Amsterdam, Netherlands.
- Petri, G., & Von Wangenheim, C. G. (2017). How games for computing education are evaluated? A systematic literature review. *Computers And Education/Computers & Education*, 107, 68-90. <https://doi.org/10.1016/j.compedu.2017.01.004>
- Price, C. B., & Price-Mohr, R. M. (2018). An Evaluation of Primary School Children Coding Using a Text-Based Language (Java). *Computers In The Schools*, 35(4), 284-301. <https://doi.org/10.1080/07380569.2018.1531613>
- Román-González, M., Moreno-León, J. & Robles, G. (2017). Complementary Tools for Computational Thinking Assessment. *Proceedings of the International Conference on Computational Thinking Education*, (pp. 154–59). https://www.researchgate.net/publication/318469859_Complementary_Tools_for_Computational_Thinking_Assessment
- Rowe, E., Asbell-Clarke, J., Almeda, M. V., Gasca, S., Edwards, T., Bardar, E., Shute, V., & Ventura, M. (2021). Interactive Assessments of CT (IACT): Digital Interactive Logic Puzzles to Assess Computational Thinking in Grades 3–8. *International Journal Of Computer Science Education In Schools*, 5(2), 28-73. <https://doi.org/10.21585/ijcses.v5i1.149>
- Salas, J. A. S. (2010). *Una Eina Informàtica d'Autoavaluació del Treball Equip en Projectes de Desenvolupament de Software*. Dialnet. <https://dialnet.unirioja.es/servlet/articulo?codigo=6399546>
- Tang, X., Yin, Y., Lin, Q., Hadad, R., & Zhai, X. (2020). Assessing computational thinking: A systematic review of empirical studies. *Computers And Education/Computers & Education*, 148, 103798. <https://doi.org/10.1016/j.compedu.2019.103798>

Viudes, M. V. L., & Fernández, S. I. M. (2011). *Propuesta de innovación a la hora de evaluar en la asignatura Modelos y Simulación*. Dialnet.
<https://dialnet.unirioja.es/servlet/articulo?codigo=4216247>



7. Anexos

Tabla I. Rúbrica para la propuesta de evaluación automática

	Experto	Avanzado	Intermedio	Principiante	Ponderación
	4	3	2	1	
Creación de la base de datos	Crea la base de datos correctamente y otorga un nombre descriptivo a esta.	Crea la base de datos correctamente.	La sintaxis de creación de la base de datos es incorrecta.	La sintaxis de creación de la base de datos es incorrecta.	10%
Creación de tablas con sus campos y claves primarias y secundarias	Crea las tablas y sus claves primarias y secundarias casi a la perfección, para optimizar el rendimiento de ellas.	Crea las tablas de manera adecuada, aunque falla en la asignación de algunas claves primarias o secundarias.	Crea las tablas, descuidando las claves primarias o secundarias.	La sintaxis de creación de las tablas y claves es incorrecta.	10%
Creación de relaciones entre tablas	Realiza las relaciones entre tablas de la manera más eficiente posible.	Relaciona las tablas de manera aceptable, aunque alguna tabla está relacionada incorrectamente.	Relaciona las tablas, aunque no sigue un criterio claro para hacerlo.	Existen tablas que no se relacionan.	10%
Creación de índices que favorezcan la consulta posterior de las tablas creadas	Todos los índices se crean para optimizar el rendimiento de la base de datos al máximo.	Crea todos los índices, aunque alguno de ellos resulta redundante o no aprovecha todas las posibilidades de estos.	Crea alguno de los índices, y pueden existir fallos en algunos de los creados.	La sintaxis de la creación de los índices es incorrecta o no se realiza.	15%
Diseño e implementación de rutinas que pueblen la base de datos y las tablas	Realiza un buen diseño de rutinas y una implementación eficiente de ellas.	Diseña unas buenas rutinas pero comete algún fallo en la implementación de ellas.	Diseña o implementa de manera parcial las rutinas requeridas por la base de datos.	No diseña o implementa las rutinas requeridas.	15%

Configuración de permisos y roles de usuario para la base de datos y las tablas	Configura todos los permisos y roles de usuario para el adecuado control de accesos.	Configura permisos y roles de usuario de manera parcial.	No configura o bien los permisos o bien los roles de usuario, y comete fallos en la configuración.	No configura permisos ni roles de usuario.	20%
Configuración de eventos y disparadores	Configura eventos y disparadores para tener contempladas la gran mayoría de casuísticas objetivo.	Configura eventos y disparadores, aunque comete fallos en la configuración o descuida los más importantes.	No configura o bien eventos o bien disparadores, y comete fallos en la configuración.	No configura eventos ni disparadores.	20%

