

# Enseñanza y Aprendizaje de Ingeniería de Computadores

Revista de Experiencias  
Docentes en Ingeniería de  
Computadores

Número 9, Diciembre 2019



**Edita:** Departamento de  
Arquitectura y Tecnología de  
Computadores



**Universidad de Granada**



**ENSEÑANZA Y APRENDIZAJE DE INGENIERÍA DE COMPUTADORES**  
**Revista de Experiencias Docentes en Ingeniería de Computadores**

TEACHING AND LEARNING COMPUTER ENGINEERING  
Journal of Educational Experiences on Computer Engineering

**Número 9, Año 2019**

**Comité Editorial:**

Miembros de la Comisión Docente del Departamento de Arquitectura y Tecnología de Computadores de la Universidad de Granada:

Mancia Anguita López	Alberto Guillén Perales
José Luis Bernier Villamor	Luis Javier Herrera Maldonado
Pedro A. Castillo Valdivieso	Gonzalo Olivares Ruiz
Miguel Damas Hermoso	Julio Ortega Lopera
Javier Diaz Alonso	Begoña del Pino Prieto
Antonio Díaz García	Beatriz Prieto Campos
F. Javier Fernández Baldomero	Alberto Prieto Espinosa
Francisco Gómez Mula	Manuel Rodríguez Álvarez
Jesús González Peñalver	Fernando Rojas Ruiz

Colaboradores externos de otras Universidades:

Sergio A. Cuenca Asensi (Universidad de Alicante)  
Domingo Benítez Díaz (Universidad de Las Palmas de Gran Canaria)  
Guillermo Botella Juan (Universidad Complutense de Madrid)  
José Carlos Cabaleiro Domínguez (Universidad de Santiago de Compostela)  
Jesús Carretero Pérez (Universidad Carlos III)  
Francisco Charte Ojeda (Universidad de Jaén)  
Anton Civit Balcells (Universidad de Sevilla)  
Ramón Doallo Biempica (Universidad de A Coruña)  
José Manuel García Carrasco (Universidad de Murcia)  
Consolación Gil Montoya (Universidad de Almería)  
José Ignacio Hidalgo Pérez (Universidad Complutense de Madrid)  
Juan Antonio Holgado Terriza (Dept. LSI, Universidad de Granada)  
Pedro López (Universidad Politécnica de Valencia)  
Diego R. Llanos Ferraris (Universidad de Valladolid)  
Joaquín Olivares Bueno (Universidad de Córdoba)  
Francisco J. Quiles Flor (Universidad de Castilla-La Mancha)  
Enrique S. Quintana Ortí (Universidad Jaime I)  
Dolores I. Rexachs del Rosario (Universidad Autónoma de Barcelona)  
Antonio Jesús Rivera Rivas (Universidad de Jaén)  
Goïuria Sagardui Mendieta (Universidad de Mondragón)  
Manuel Ujaldón Martínez (Universidad de Málaga)  
Miguel Ángel Vega Rodríguez (Universidad de Extremadura)  
Víctor Viñals Yúfera (Universidad de Zaragoza)

**ISSN:** 2173-8688, **ISSN versión electrónica:** 2695-5261

**Depósito Legal:** GR-899/2011

**Edita:** Departamento de Arquitectura y Tecnología de Computadores

**Imprime:** Copicentro Editorial

© Se pueden copiar, distribuir y comunicar públicamente contenidos de esta publicación bajo las condiciones siguientes (<http://creativecommons.org/licenses/by-nc-nd/3.0/es/>):

**Reconocimiento** – Debe reconocer los créditos de la obra de la manera especificada por el autor o el licenciador (pero no de una manera que sugiera que tiene su apoyo o apoyan el uso que hace de su obra).

**No comercial** – No puede utilizar esta obra para fines comerciales.

**Sin obras derivadas** – No se puede alterar, transformar o generar una obra derivada a partir de esta obra.

Printed in Spain

Impresa en España

# Índice

Editorial .....	1
La hoja de ruta de la ingeniería de computadores al final de la ley de Moore y el escalado de Dennard <i>J. Ortega, M. Anguita, A. Prieto, A. Cañas, M. Damas, A.F. Díaz, J. Fernández, J. González</i> .....	5
Modelado matemático, simulación y control de una mini motocicleta autónoma con rueda de reacción <i>J.C. Saavedra, A. Olivares, G. Olivares</i> .....	29
Sistema Empotrado Distribuido para el Control de Accesos – RFIDoors <i>D. García, I. Aguilera, A.A. Del Barrio, G. Botella</i> .....	43
Uso de la realidad aumentada para la mejora de la visión espacial del alumnado <i>A.J. Moreno, O. Baños</i> .....	53
Aplicaciones de Realidad Virtual y Realidad Aumentada como soporte a la enseñanza del Dibujo Técnico <i>Y. Alvarado, N. Jofré, M. Rosas, R. Guerrero</i> .....	65
Desarrollo de un repositorio de certificación basado en Blockchain para la evaluación de una asignatura de grado <i>P.A. Castillo, P. García-Sánchez, M.C. Aguilar-Luzón, J. García, E. Herrera-Viedma</i> .....	75
Estudio del paradigma Internet de las Cosas y tecnologías inalámbricas como arquitectura base de captación de datos en las fases de ejecución y mantenimiento de los proyectos de construcción <i>M. Martínez-Rojas, G. Corpas, A. Vitiello, G. Acampora, J.M. Soto-Hidalgo</i> .....	87
Teaching C/C ++ programming using a multimedia system with videos <i>J. Galindo, P. Galindo, J.M. Rodríguez</i> .....	97

Aplicación de técnicas de aprendizaje activo a la enseñanza de la programación de ordenadores <i>E. Gibaja, M. Luque, A. Zafra</i> .....	107
Modelo para fomentar el aprendizaje activo por medio de Designación Thinking en las Plataformas LMS con base en enfoque ágil <i>Cynthia López Valerio</i> .....	119
Instrucciones para autores .....	129

## Editorial

Presentamos aquí el noveno número de la revista Enseñanza y Aprendizaje de Ingeniería de Computadores que, desde 2011 viene editando anualmente el Departamento de Arquitectura y Tecnología de Computadores de la Universidad de Granada. De los diez artículos que incluye el número, cuatro son versiones extendidas de las Jornadas Andaluzas de Informática (JAI) de 2019, celebradas en Septiembre en Canillas de Aceituno (Málaga). Agradecemos su contribución a los autores de estos diez trabajos, pertenecientes a cinco Universidades españolas (Universidades de Córdoba, Cádiz, Granada, Málaga, Politécnica de Madrid) y tres extranjeras, respectivamente de Argentina (Universidad Nacional de San Luis), Costa Rica (Universidad Latina de Costa Rica), e Italia (Universidad Federico II de Nápoles).

El número se abre con “La hoja de ruta de la ingeniería de computadores al final de la ley de Moore y el escalado de Dennard”, de Julio Ortega, Mancia Anguita, Alberto Prieto, Antonio Cañas, Miguel Damas, Antonio F. Díaz, Javier Fernández, y Jesús González, de la Universidad de Granada. En este artículo se realiza un análisis de la evolución que previsiblemente seguirá la ingeniería de computadores a partir de esta segunda década del siglo XXI que se inicia en breve, una vez se aproximan el final de la ley Moore y del escalado de Dennard. Se justifican algunos de los cambios que deberían establecerse en la enseñanza superior de esta disciplina, al tiempo que se reflexiona sobre los límites teóricos que se pueden establecer para la computación, y las expectativas que ofrece la computación cuántica.

En “Modelado matemático, simulación y control de una mini motocicleta autónoma con rueda de reacción”, Juan Carlos Saavedra, Alberto Olivares, y Gonzalo Olivares de la Universidad de Granada proponen un modelo para la simulación y el control de una mini motocicleta autónoma estabilizada con una rueda de reacción. El sistema se ha desarrollado a partir de un diseño CAD, que se integra en *Simulink* junto con los módulos de control, y se emplea la enseñanza del diseño de sistemas de control, incluyendo elementos mecánicos, sensores, actuadores, así como la dinámica de contacto de las ruedas con el suelo.

El artículo “Sistema Empotrado Distribuido para el Control de Accesos – RFIDoors”, de Daniel García Moreno, Iván Aguilera Calle, Alberto A. del Barrio, y Guillermo Botella, de la Universidad Complutense de Madrid, continúa la línea de implementación de sistemas empotrados distribuidos compuestos por elementos de fácil adquisición y de bajo coste, como Raspberry Pi, en este caso para un sistema de identificación por radiofrecuencia (RFID), con el objetivo de controlar y gestionar la autenticación para la

apertura y cierre de puertas. Además, el sistema incluye un servidor y una aplicación para la parte administrativa y operativa del sistema.

A continuación siguen dos contribuciones que ilustran algunos aspectos del uso de la realidad aumentada en docencia. Se trata de “Uso de la realidad aumentada para la mejora de la visión espacial del alumnado” de Antonio José Moreno Garrido y Oresti Baños Legrán, de la Universidad de Granada, y “Aplicaciones de Realidad Virtual y Realidad Aumentada como soporte a la enseñanza del Dibujo Técnico” de Yoselie Alvarado, Nicolás Jofré, María Rosas, y Roberto Guerrero, de la Universidad Nacional de San Luis en Argentina. Mientras que en el primero de ellos se analiza si el uso de la realidad aumentada podría ser una herramienta útil para mejorar la visión espacial del alumno, en el segundo se considera su uso en la docencia de una asignatura.

La tecnología *Blockchain* se puede entender como una base de datos distribuida en Internet, sin un control central, y una serie de características que transformarán áreas tan importantes como el gobierno, la economía, y el funcionamiento de las organizaciones. Otro ámbito en el que, indudablemente, esta tecnología va a tener una incidencia importante es la educación. En el artículo “Desarrollo de un repositorio de certificación basado en *Blockchain* para la evaluación de una asignatura de grado”, Pedro A. Castillo, Pablo García-Sánchez, M. Carmen Aguilar-Luzón, Julia García Cabello, y Enrique Herrera-Viedma, de la Universidad de Granada, ponen de manifiesto una aplicación de *blockchain* en la docencia universitaria.

Este número de “Enseñanza y Aprendizaje de Ingeniería de Computadores” concluye con cuatro artículos correspondientes a ampliaciones de otras tantas contribuciones a las IV Jornadas Andaluzas de Informática (JAI), celebradas este Septiembre de 2019.

En el primero de ellos, titulado “Estudio del paradigma Internet de las Cosas y tecnologías inalámbricas como arquitectura base de captación de datos en las fases de ejecución y mantenimiento de los proyectos de construcción”, María Martínez-Rojas, Gregorio Corpas, Autilia Vitiello, Giovanni Acampora, y José Manuel Soto-Hidalgo respectivamente de las Universidades de Málaga, Córdoba, y Federico II de Nápoles (Italia), ponen de manifiesto la vigencia del paradigma de Internet de las Cosas en el desarrollo de proyectos, en este caso en el ámbito de la construcción, para mejorar la adquisición de datos y gestión de la información. La adquisición y compartición de datos mediante dispositivos interconectados permite disponer de ellos en tiempo real y que, posteriormente, puedan ser gestionados por sistemas que proporcionen el soporte inteligente del proceso de toma de decisiones.

A continuación, en “Teaching C/C ++ programming using a multimedia system with videos”, José Galindo, Patricia Galindo, y José María Rodríguez Corral, de la Universidad de Cádiz, describen una aproximación basada en vídeos para acelerar el aprendizaje del lenguaje de programación C/C++, ampliamente utilizado para enseñar programación a los estudiantes de ingeniería industrial en los primeros cursos de su carrera.

Finalmente, los artículos “Aplicación de técnicas de aprendizaje activo a la enseñanza de la programación de ordenadores”, de Eva Gibaja, María Luque, y Amelia Zafra, de la Universidad de Córdoba, y “Modelo para fomentar el aprendizaje activo por medio de Designación Thinking en las Plataformas LMS con base en enfoque ágil”, de Cynthia López Valerio, de la Universidad Latina de Costa Rica, inciden en la temática del

aprendizaje activo. En el primero de los dos se propone un programa de prácticas para la materia Programación que se imparte en el Grado en Ingeniería Informática de la Universidad de Córdoba, enfocado a que el estudiante se implique en su propio proceso de aprendizaje, experimentando con problemas de complejidad media que le acerquen al mundo real. En el segundo de estos dos artículos finales se propone un modelo para implementar *e-learning* a partir de las bases de la taxonomía de Bloom en una plataforma LMS basada en *Moodle*, y siguiendo un enfoque ágil que contribuya a disminuir el tiempo de aprendizaje y aumentar la satisfacción del estudiante.

Para terminar, agradecemos a los revisores su trabajo y sus comentarios y sugerencias, que han contribuido a la mejora de los artículos presentados, así como la labor de difusión de la revista que realizan algunos de los miembros del Comité Editorial. Finalmente, reiteramos nuestro agradecimiento a los autores de los artículos presentados, sin cuyo esfuerzo y confianza no hubiera sido posible elaborar este noveno número de la revista.

El Comité Editorial



## La hoja de ruta de la ingeniería de computadores al final de la ley de Moore y el escalado de Dennard

Julio Ortega, Mancia Anguita, Alberto Prieto, Antonio Cañas, Miguel Damas,  
Antonio F. Díaz, Javier Fernández, Jesús González

Departamento de Arquitectura y Tecnología de Computadores  
E.T.S.I.I.T., Universidad de Granada

[jortega, manguita, aprieto, acanas, mdamas, afdiaz, jfernand, jesusgonzalez}@ugr.es](mailto:{jortega, manguita, aprieto, acanas, mdamas, afdiaz, jfernand, jesusgonzalez}@ugr.es)

**Resumen.** En el presente trabajo se hace una revisión sobre la situación de la ingeniería de computadores al inicio de la década de los 2020 con objeto de perfilar algunos de los cambios que deberían establecerse en la enseñanza superior de esta disciplina. Se considera la gran relevancia del control del consumo energético y de las aplicaciones relacionadas con clasificación y optimización que requieren cantidades ingentes de datos (*big data*) y tiempos de respuesta difícilmente alcanzables utilizando las técnicas tradicionales de la ingeniería de computadores, y dada la reducción del ritmo que marca la ley de Moore y el final del escalado de Dennard. El artículo proporciona referencias bibliográficas recientes sobre la situación de la ingeniería de computadores, e identifica los nuevos requisitos de las interfaces presentes en la jerarquía de capas propia de los sistemas de cómputo, fundamentalmente los relacionados con la seguridad, el consumo energético, y el aprovechamiento del paralelismo heterogéneo. También se reflexiona sobre los límites teóricos que se pueden establecer para la computación y las expectativas que ofrece la computación cuántica.

Palabras clave: Arquitectura de Computadores, Arquitecturas de uso específico (DSA, *Domain-Specific Architectures*), Ataques *meltdown* y *spectre*, Computación energéticamente eficiente, Computación cuántica, Ingeniería de computadores, Paralelismo heterogéneo.

**Abstract.** This paper reviews the state of Computer Engineering at the beginning of the 2020s in order to outline some of the changes that should be established in higher education in this discipline. It is considered the great relevance of controlling energy consumption and applications related to classification and optimization that require huge amounts of data (*big data*) and response times difficult to achieve using traditional techniques of computer engineering, and given the reduction of the improvement rate set by *Moore's law* and the end of *Dennard scaling*. The article also provides recent bibliographical references on the situation of Computer Engineering, and identifies the new requirements of the interfaces present in the hierarchy of layers of computer systems, mainly those related to security, energy consumption, and the use of heterogeneous parallelism. It also reflects on the theoretical limits that can be established for computation and the expectations that quantum computation offers.

**Keywords:** Computer Architecture, Computer Engineering, Domain-Specific Architecture (DSA), Heterogeneous parallel computing, *Meltdown* and *Spectre* Attacks, Power-aware computing, Quantum computing.

## 1 Introducción

En 2005, algunos de los autores de este artículo publicaron un libro de texto de Arquitectura de Computadores [1], que se ha venido utilizando en diversas asignaturas del ámbito de la Ingeniería de Computadores. Han transcurrido casi quince años desde 2005, y la Ingeniería de Computadores ha experimentado cambios en los paradigmas que marcan los principios cuantitativos y cualitativos para el desarrollo de nuevos computadores, que deben tenerse en cuenta tanto en el diseño de nuevos planes de estudio para las carreras relacionados con la Ingeniería de Computadores, como en la elaboración de nuevos libros de texto para asignaturas en dicho ámbito. Así, en este artículo también se proponen los contenidos que debería incluir una posible nueva edición de un libro de texto de Arquitectura de Computadores para que cubra los contenidos relevantes de las asignaturas relacionadas de Ingeniería de Computadores en los próximos años. Para realizar esta propuesta se han considerado iniciativas que marcan los nuevos *roadmaps* a seguir, como es el caso de la última hoja de ruta de semiconductores publicada por la ITRS (*International Technology Roadmap for Semiconductors 2.0*), [www.itrs2.net](http://www.itrs2.net). Hay que tener en cuenta que el número de transistores incluidos en un circuito integrado seguramente seguirá creciendo, durante algunos años más, según lo establecido por la ley de Moore. Sin embargo, el problema aparece en el rendimiento que se extrae al añadir más transistores. Así, cuantas más etapas tenga un cauce, mayor es el número de instrucciones que se necesitan para mantenerlo lleno y maximizar el rendimiento de los transistores añadidos. El número de instrucciones disponibles para entrar en el cauce depende del número de instrucciones de salto que aparezcan en el código, aunque mediante el procesamiento especulativo se pueden seguir introduciendo instrucciones en el cauce tras una instrucción de salto condicional no resuelto (aunque no todas las instrucciones que se empiecen a procesar tengan que finalizar y actualizar sus resultados). A medida que los cauces son más “profundos”, desperdician una mayor cantidad del trabajo que realizan en sus etapas dado que se pueden introducir inútilmente más instrucciones ubicadas en las alternativas de las instrucciones de salto condicional que no hay que tomar. Hacia 2003, los procesadores superaban consumos de 200W por circuito integrado, marcando un umbral de potencia a partir del cual es más cara la tecnología para *enfriar* el procesador que la del resto de elementos del computador. De hecho, la frecuencia de reloj del procesador se mantuvo alrededor de los 2 GHz durante un periodo de tiempo considerable en esos años.

La Figura 1, elaborada con datos obtenidos de [2], muestra que las prestaciones de ejecución del *benchmark* SPECint respecto al VAX 11/780 pasaron de crecer un 52% anual entre 1986 y 2003, a un 12% entre 2011 y 2015, y *sólo* un 3.5% desde entonces.

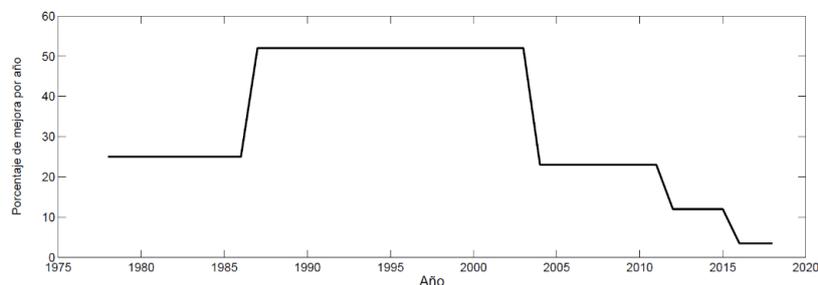


Figura 1. Evolución temporal de las prestaciones de ejecución del *benchmark* SPECint respecto al VAX 11/780 (datos obtenidos de [2]).

Además, desde aproximadamente 2004, y precisamente por la continua reducción del tamaño de los transistores, no se cumple el denominado escalado de Dennard, según el cual, para un área de silicio dada, la densidad de potencia instantánea se mantendría constante al incrementar el número de transistores. Desde alrededor de 2004 se viene observando una reducción considerable del porcentaje de mejora anual, habiéndose pasado de un 52% a un 23%. Es cada vez más difícil obtener mejoras de prestaciones basadas en aumentar la eficiencia del paralelismo entre instrucciones (ILP) en un núcleo de procesador segmentado de uso general. Además, si la frecuencia de reloj sigue creciendo en cauces cada vez más complejos, se hace más difícil conseguir que las salidas de una etapa del cauce puedan llegar, en un ciclo de reloj, a las entradas de la siguiente etapa. Una alternativa para abordar esta interacción entre aumento de la frecuencia de reloj y núcleos con cauces cada vez más complejos, ha sido la integración de más núcleos por circuito integrado, aprovechando el aumento del número de transistores disponibles. No obstante, la posible mejora de prestaciones asociada al aprovechamiento del procesamiento paralelo en estas arquitecturas de microprocesador multi-núcleo deja de ser transparente al usuario, que debe paralelizar convenientemente las aplicaciones para optimizar el uso de la microarquitectura multi-núcleo correspondiente.

Tras esta introducción, la Sección 2 describe los criterios a partir de los que se definirán las hojas de ruta para el desarrollo de las arquitecturas del computador y de los distintos elementos de la ingeniería de computadores. Precisamente la Sección 3 describe la aproximación al estudio del computador en términos de la jerarquía de capas entre las que actúan las distintas interfaces características del computador, al tiempo que se presentan los agentes que determinan la evolución de los computadores. La Sección 4 se dedica a los requisitos para las interfaces a desarrollar en un futuro próximo. Finalmente, la Sección 5 analiza los límites físicos para las prestaciones de los computadores y la Sección 6 introduce las principales características de los computadores cuánticos, que actualmente constituyen una alternativa contemplada con gran interés, para aproximar las prestaciones de los computadores a sus límites físicos, más allá de lo que se prevé conseguir mediante la tecnología electrónica utilizada actualmente. Para terminar, la Sección 7 del artículo discute una propuesta de contenidos para una asignatura de Arquitectura de Computadores (y también para el libro de texto correspondiente) a partir de las conclusiones alcanzadas a lo largo del artículo, y resumidas en la Sección 8.

## 2 Las nuevas hojas de ruta para el siglo XXI

En los próximos años, tras el agotamiento de la ley de Moore y el escalado de Dennard, el aumento de prestaciones en las arquitecturas no va a provenir fundamentalmente de la disponibilidad de arquitecturas con velocidades pico que se doblan cada dieciocho o veinticuatro meses, sino que habrá que prestar más atención al uso que se hace de las arquitecturas de cómputo para mejorar las prestaciones en las aplicaciones más demandadas en cada momento. Ha sido el desarrollo de las tecnologías de diseño e implementación hardware, así como la popularización de cierto tipo de aplicaciones, relacionadas con la inteligencia artificial y el *big data*, las que han impulsado el desarrollo de una tendencia en la mejora de las prestaciones basada en el uso de arquitecturas de dominio específico (DSA, *Domain-Specific Architectures*). En estas aplicaciones, gran parte del tiempo de ejecución corresponde al procesamiento de núcleos de código, o *kernels*, que actualmente implementan sobre todo tareas relacionadas con problemas de clasificación, optimización, aprendizaje, agrupamiento (*clustering*), etc, que pueden usarse en diferentes aplicaciones, con distinto nivel de granularidad. Teniendo en cuenta la ley de Amdahl, la aproximación más eficiente para acelerar la aplicación es, precisamente, buscar formulaciones de la misma que permitan aprovechar los tipos de paralelismo posibles. Así, las nuevas hojas de ruta para conseguir arquitecturas de cómputo cada vez más eficientes deben basarse en el aprovechamiento del paralelismo heterogéneo, en sus diversos niveles y para las aplicaciones más demandadas. Así, en cada momento habría que identificar los principales núcleos de procesamiento en los que la aplicación estaría involucrada la mayor parte del tiempo. Se identificarían elementos de cómputo que, al encontrarse repetidos, incluso en aplicaciones de dominio específico (elementos neuronales en redes neuronales profundas, o módulos de funciones comunes en metaheurísticas de optimización) contribuirían a reducir el coste de diseño y fabricación de las arquitecturas.

En relación con el consumo energético, las arquitecturas de uso específico, o de dominio específico (DSA) [2] permiten disponer de hardware eficiente para la ejecución de la aplicación considerada [3]. Por ejemplo, se intentan reducir las distancias entre las que hay que mover datos, incluyendo en la microarquitectura memorias controladas por software en lugar de cachés, cuyo hardware de control las hace menos eficientes energéticamente. No obstante, el diseño de DSA implica que el ingeniero de computadores debe ser consciente del proceso de optimización del hardware para la aplicación en cuestión. Es decir, no solo debe conocer la aplicación a fondo, sino también las partes de la arquitectura y la microarquitectura a optimizar: (1) el tamaño, la ubicación, y la gestión de las memorias, (2) incluir recursos hardware optimizados para los elementos de la microarquitectura que son muy frecuentemente utilizados (unidades aritmético-lógicas específicas), (3) implementar el paralelismo que precisa el dominio, (4) usar el lenguaje de programación específico para el dominio, y (5) usar el tamaño de datos ajustado a los tipos característicos del dominio.

Esta nueva forma de definir tendencias evolutivas, hojas de ruta, o *roadmaps*, en las arquitecturas de cómputo [4], constituye un cambio de paradigma del que estamos siendo testigos en esta segunda década del siglo XXI, y según la opinión de algunos

ingenieros de computadores (conferencia de Hennessy y Patterson al recibir el premio Turing de la ACM de 2017 [5]), va a ocasionar un nuevo resurgimiento de alternativas y posibilidades para la Ingeniería de Computadores. La forma usual de prefijar objetivos de mejora que se evalúan a través de conjuntos de *benchmarks* de uso general, ha dado paso a la necesidad de reenfocar las evaluaciones a aplicaciones específicas pero muy demandadas, que requieren mayores necesidades de cómputo, tanto en tiempo como en almacenamiento y eficiencia de consumo energético. Las nuevas propuestas de la Ingeniería de Computadores compiten no ya utilizando medidas como la velocidad pico, sino comparando tiempos de ejecución, consumo de energía, o necesidades de almacenamiento en versiones específicas hardware/software de instancias de aplicaciones reales muy demandadas socio-económicamente. Esta situación condiciona el tipo de recursos que se deben incluir en la arquitectura de la plataforma a considerar y, previsiblemente, a corto y medio plazo, las tendencias dominantes buscarán una redefinición de los niveles de capas. Se trata de aprovechar más eficazmente el paralelismo, e idear nuevos procedimientos para la optimización de arquitecturas orientadas a la implementación eficiente, en cuanto a velocidad, consumo energético, y aprovechamiento del paralelismo en todos sus niveles. Los nuevos conceptos deben proyectarse para las aplicaciones frecuentemente demandadas, rentabilizando la implementación de arquitecturas de dominio específico y disminuyendo los costos hardware/software no recurrentes de ingeniería, sin olvidar que el diseño de jerarquías de memoria debe ser energéticamente eficiente.

Con este panorama, el diseño de las interfaces que permitan el aprovechamiento eficiente del hardware, no solo sigue siendo fundamental, sino que debe incorporar elementos que den respuesta a las nuevas circunstancias resultantes de la situación actual de la ley de Moore y el escalado de Dennard.

### 3 Jerarquía de capas e ingeniería de computadores

La aproximación al estudio de un sistema complejo, como es el caso de un computador, se lleva a cabo dividiéndolo en partes que interactúan entre sí de acuerdo con un orden jerárquico. En un computador es usual definir un conjunto de capas o niveles de abstracción que se suceden unas a otras de forma que la del nivel inferior, con un menor nivel de abstracción por estar más próxima al nivel de las tecnologías físicas utilizadas para implementar el computador, genera elementos utilizables por la capa del nivel superior, y así sucesivamente hasta el nivel de aplicación, que considera programas implementados con lenguajes de alto nivel. Así, en un computador tendremos las capas de dispositivo, de lógica, de unidad funcional, de microarquitectura, de arquitectura del conjunto de instrucciones, arquitectura, de API (interfaz de programación de aplicaciones), de lenguaje, y de algoritmo [6]. Por otro lado, las distintas capas o niveles se pueden agrupar para definir bloques de elementos que interactúan y permiten entender, entre otras cosas, la evolución temporal de los computadores. Al estructurar las funciones del computador en capas es posible diseñarlo y analizarlo a través de subsistemas dotados de cierta independencia.

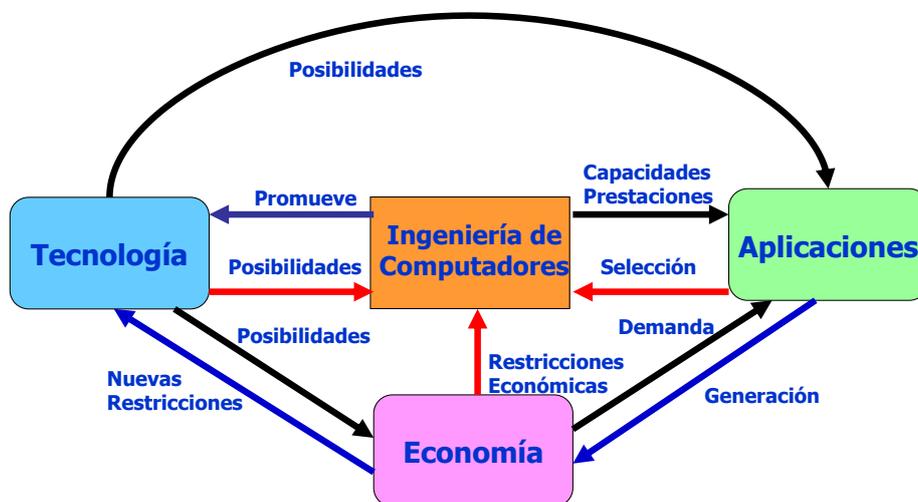


Figura 2. Interacciones entre los agentes que intervienen en la evolución de los computadores

Precisamente, la Figura 2 muestra la interrelación entre tecnología, aplicaciones, economía, e ingeniería de computadores. Así, las capas correspondientes a las tecnologías físicas que se utilizan para implementar el hardware del computador definen el bloque de *tecnología* de la Figura 2. También están las capas que atañen a las aplicaciones y al diseño de algoritmos y programas en lenguajes de alto nivel que se ejecutan en el computador para abordar dichas aplicaciones (en la Figura 2 se utiliza el término *aplicaciones*), y las capas que definen lo que se conoce como *ingeniería de computadores*, o interfaz entre el hardware y los distintos niveles de abstracción del software (capas de dispositivo, de lógica, de unidad funcional, de microarquitectura, de arquitectura del conjunto de instrucciones, arquitectura, de API). Por otra parte, la relevancia y la demanda social de las aplicaciones determinan la intervención de los factores económicos en la evolución de los computadores (bloque *economía* en la Figura 2). En el diagrama de la Figura 2 también se ilustra la interrelación entre tecnología, aplicaciones, economía, e ingeniería, que condiciona el desarrollo de estos sistemas. Las aproximaciones que se propongan en un futuro para mantener el crecimiento de prestaciones de las arquitecturas se pueden contemplar también considerando en qué capa, o capas, la nueva propuesta ocasiona una mayor disrupción, como se hace en [6]. Así, la denominada aproximación “*más Moore*”, que busca continuar mejorando gracias al aumento del número de transistores incluidos en los circuitos integrados, afecta fundamentalmente a la capa de dispositivo. La aproximación de “*cambios ocultos*” busca mejoras a partir de innovaciones en la capa de microarquitectura y capas inferiores, y la aproximación de “*cambios en la arquitectura*” propone modificaciones desde la capa del conjunto de instrucciones a las más próximas al hardware. Finalmente, los computadores “*no von Neumann*” implican cambios disruptivos prácticamente en todas las capas. Aquellas capas en las que la evolución de la Ingeniería de Computadores pueda producir cambios, deben abarcar los contenidos cuyo estudio deben incluirse en un libro de texto de ingeniería de

computadores, para así proporcionar los elementos que definen el espíritu de la materia, y permitir al ingeniero mantenerse activo en su contribución al progreso futuro de la disciplina.

Los fundamentos de la ingeniería de computadores parten de conceptos propios de las áreas de computación, electrónica, matemáticas y física, y deben tener en cuenta las posibilidades de aplicación de las tecnologías emergentes que, si bien pueden ser accesibles comercialmente, y se les reconoce como posibles promotores de transformaciones sociales futuras relevantes, suelen tener consecuencias que, en el momento considerado, no están claramente definidas entre los contenidos y la orientación de la ingeniería de computadores, y precisan, por tanto, un análisis más profundo [7]. Por tanto, un ingeniero competente debe ser capaz de identificar las tecnologías emergentes que contribuyen al crecimiento de la disciplina de ingeniería de computadores en un mundo en continua transformación. De hecho, los ingenieros de computadores estarán pronto involucrados en el diseño de computadores para cuya implementación se usarán tecnologías ópticas, cuánticas y/o biológicas, y/o en el desarrollo de sistemas robóticos avanzados para la nueva industria 4.0, con técnicas de realidad virtual, de análisis de datos o *big data*, que afectarán considerablemente a su trabajo.

Así, aunque los cambios en la Ingeniería de Computadores de los últimos años han venido tanto desde los niveles más próximos al hardware, tal y como se ha comentado antes en relación con el final del escalado de Dennard y el estancamiento de las prestaciones de los cauces segmentados, y con la generalización de microarquitecturas multinúcleo y la aparición de coprocesadores orientados al aumento del flujo de operaciones (throughput), como es el caso de las GPU, también se han propuesto mejoras en las capas más altas, entre los que están los cambios relacionados con el aprendizaje automático y las redes neuronales profundas [3], que permiten reemplazar el software complejo, comúnmente denominado “*Software 2.0*”, gracias a que los modelos de computación que se utilizan se basan en la interconexión de conjuntos de elementos neuronales similares, y relativamente simples de programar.

#### **4 Las interfaces de computador del siglo XXI**

Más arriba se ha definido la Ingeniería de Computadores como la interfaz entre el hardware y los distintos niveles de abstracción del software. Por tanto, como disciplina, incluye el diseño, evaluación y optimización de las interfaces entre las capas de algoritmos, descritos en términos de lenguajes de alto nivel, y las relacionadas con las tecnologías utilizadas para implementar el hardware, y permiten aislar los detalles de las capas de niveles más bajos y mayor complejidad. En este contexto, el repertorio de instrucciones máquina es la parte del computador visible para el programador y el compilador, y sus prestaciones se suelen estimar utilizando conjuntos de programas de prueba o *benchmarks* ejecutados en distintos tipos de máquinas: computadores de sobremesa, servidores, y computadores móviles personales y computadores empujados en aplicaciones. En muchas aplicaciones, la capacidad de procesamiento

con datos en coma flotante es menos importante que la necesidad de memoria [3], dado que una disminución de la memoria necesaria supone una reducción importante del coste energético de la aplicación. Desde esta perspectiva, el tamaño del código es importante. No obstante, repertorios no RISC como el 80x86 han mantenido una posición competitiva frente a los RISC para mantener la compatibilidad con códigos para PC y gracias a que el elevado número de transistores disponible ha permitido a los ingenieros implementar procesadores CISC con microarquitecturas capaces de traducir dinámicamente los códigos máquina CISC a secuencias de micro-operaciones de tipo RISC.

Hasta el momento actual, el hardware se ha implementado fundamentalmente a través de las tecnologías electrónica y electromagnética, pero los requisitos de prestaciones para nuevas aplicaciones pueden requerir cambios en las tecnologías hardware que se venían utilizando. Así, aplicaciones propias del campo de la inteligencia artificial han promovido el desarrollo de arquitecturas de dominio específico para *deep learning* [3], y la demanda de aplicaciones que no son fácilmente abordables mediante computadores electrónicos *clásicos*, está promoviendo el uso de nuevas tecnologías [7], como en el caso de los computadores cuánticos. La evolución tecnológica ha hecho aflorar nuevos requisitos de prestaciones relacionados con la eficiencia energética, la programabilidad, la fiabilidad, y la seguridad, más allá de la velocidad de cómputo [8].

Estas circunstancias también han generado nuevas exigencias de diseño para las interfaces que, más que ocultar detalles de una capa a las capas superiores, de menor complejidad, deben tener acceso a elementos de capas de abstracción diferentes para tenerlos en cuenta de cara a conseguir un comportamiento más optimizado en cuanto a las prestaciones. Se precisa, por tanto, un replanteamiento del diseño de las interfaces para que, además de permanecer estables durante periodos de tiempo considerable, proporcionen tolerancia a fallos, verificabilidad, seguridad para los datos, y velocidad de procesamiento, sobre transistores menos fiables, y con requisitos de energía más exigentes. De hecho, la arquitectura del computador define un conjunto de interfaces (el repertorio de instrucciones y la memoria virtual) que han evolucionado lentamente a lo largo de décadas, pero fueron definidas cuando la memoria era un recurso muy escaso, no había restricciones en cuanto al consumo energético, y existía poca preocupación por la seguridad. De hecho, la programabilidad (diseño y desarrollo de software ajustado a los requisitos de velocidad, consumo, fiabilidad y seguridad), que se apoyaba en técnicas de ingeniería de software como eran la modularidad y el ocultamiento de información para incrementar la productividad del programador a expensas de la eficiencia energética y la velocidad, debe buscar nuevas alternativas de optimización entre capas de distinto nivel de abstracción, el soporte hardware de la programabilidad (por ejemplo, la memoria *transaccional* busca simplificar la paralelización y la sincronización de códigos multihebra), o el diseño de lenguajes de dominio específico y script dinámico (por ejemplo, Javascript y Python).

Los repertorios de instrucciones actuales son poco eficientes a la hora de comunicar información crítica del software de alto nivel al hardware. De hecho, no se puede especificar fácilmente cuándo un programa necesita eficiencia energética, un nivel elevado de seguridad, calidad de servicio (QoS), etc. En cambio, es el hardware el que

debe recabar esa información por sí mismo, como se puede ver en el caso de la ejecución eficiente del paralelismo ILP en los procesadores superescalares. Por otra parte, hay ciertos requisitos que los repertorios máquina y las microarquitecturas disponibles actualmente no satisfacen [8]. Entre ellos están la seguridad, la eficiencia energética, y el aprovechamiento del paralelismo heterogéneo. A continuación se resumen algunos aspectos relacionados con el diseño de interfaces que proporcionan dichos requisitos entre las capas que se comunican a través de la interfaz correspondiente.

#### 4.1 Interfaces para la seguridad

La seguridad del computador se basa fundamentalmente en aislar las distintas áreas de memoria, separando direcciones de memoria del núcleo del sistema operativo que se marcan como no accesibles para el usuario, y separando zonas de memoria para procesos de usuarios diferentes. Sin embargo, ataques como *Meltdown* [9] o *Spectre* [10] aprovechan ciertos efectos colaterales de la ejecución desordenada (“*out-of-order*”, *o-o-o*) para permitir la lectura de posiciones arbitrarias de la memoria del núcleo, incluyendo claves. Este tipo de ataques aprovechan elementos incluidos en la microarquitectura (para mejorar sus prestaciones sin afectarla funcionalmente) y rompen el esquema de protección de memoria al permitir que cualquier usuario pueda leer todo el espacio del núcleo del sistema operativo del computador.

La separación entre procesos de usuario y de núcleo se suele implementar a través de un bit del procesador que establece si puede accederse a una determinada página del núcleo del sistema. Para ello, el bit de acceso se activa (se pone a uno, por ejemplo) cuando se ejecuta código del núcleo del sistema y se desactiva para permitir el acceso a páginas de procesos de usuario. Este hardware facilita a los sistemas operativos ubicar el núcleo en el espacio de direcciones de cada proceso y conmutar eficientemente entre espacio de usuario y núcleo, por ejemplo para la gestión de interrupciones. Los ataques *Spectre* y *Meltdown* explotan los recursos incluidos en la microarquitectura del procesador para mejorar sus prestaciones, creando “*canales de fuga de información*” (*side channels*) al extender la especificación funcional del repertorio de instrucciones que constituye la arquitectura del computador. Ahora, el conjunto de instrucciones utilizado como base para la portabilidad entre máquinas diferentes no es suficiente para poder asumir con total garantía propiedades y características de seguridad, y es preciso además disponer de la información de prestaciones que se obtiene al conocer detalles de funcionamiento de la microarquitectura. Por tanto, las especificaciones funcionales y de prestaciones constituyen una nueva definición de la interfaz entre aplicaciones y máquina. Surge una nueva era para la arquitectura de computadores, en la que todavía es más importante conocer las características de la microarquitectura y su incidencia en las prestaciones de la máquina. Hay que tener en cuenta que una CPU superescalar con procesamiento especulativo, por ejemplo, está diseñada para mantener la corrección funcional y deshacer los cálculos realizados especulativamente de forma innecesaria. Un ataque *Spectre* hace que el procesador ejecute especulativamente secuencias de instrucciones que no se hubieran ejecutado en la parte no especulativa del programa. A través de la ejecución de instrucciones especulativas se pueden inducir pérdidas de

información desde el espacio de memoria de la víctima del ataque. Por ejemplo, si se ejecuta una secuencia especulativa de instrucciones que incluye instrucciones de acceso a memoria, puede haber bloques de memoria que den lugar a fallos de memoria. Marcos de esos bloques se podrían introducir en la caché, a pesar de que la correspondiente excepción de memoria impediría que finalicen las instrucciones de memoria ejecutadas especulativamente (el proceso en ejecución acabaría). Sin embargo, aunque la secuencia de instrucciones especulativa no se complete, y los datos en la memoria principal no se modifiquen finalmente, podrían haberse introducido marcos de memoria en caché, que posiblemente generarían diferencias temporales entre accesos a memoria debido a *caminos encubiertos* que se pueden “generar” en la microarquitectura. Por ejemplo, instrucciones como FLUSH+RELOAD aprovechan el carácter compartido e inclusivo de la caché de último nivel. Así, instrucciones del tipo FLUSH forzarían la actualización de la línea de caché correspondiente en la memoria principal. Luego, al medir el tiempo necesario para volver a cargar el dato mediante una nueva instrucción de carga (RELOAD), el proceso atacante puede determinar si otro proceso cargó un marco de bloque en caché. Se podría identificar el proceso según haya sido el marco de bloque implicado, y diseñar códigos para implementar ataques, cambiando el estado de la microarquitectura, a pesar de que la ejecución del código se puede llevar a cabo sin cambios en los resultados. Tendríamos que, analizando únicamente el código máquina, no se podría prever *todo* lo que puede ocurrir en la microarquitectura, si incluimos en *ese todo*, por ejemplo, las cuestiones relacionadas con la seguridad de los códigos.

## 4.2 Interfaces para mejorar el consumo energético

El consumo de energía ha pasado a ser un factor muy importante en el diseño de las microarquitecturas, especialmente las de los procesadores embebidos, donde las restricciones relacionadas con la reducción del consumo siempre han sido esenciales. Además, la evolución hacia arquitecturas de uso general multi-núcleo ha sido una consecuencia de la necesidad de limitar la potencia por unidad de superficie disipada por el microprocesador, que podría haber llegado a valores inaceptablemente altos si se hubiera mantenido asociada la mejora de prestaciones a microarquitecturas superescalares cada vez más complejas tendentes a terminar más instrucciones por ciclo. Para entender la influencia de la tecnología en la potencia disipada de un circuito integrado CMOS se puede utilizar la expresión (1), en la que el primer término se debe al consumo dinámico de carga y descarga de la capacidad de salida,  $C$ , de una puerta lógica en un circuito integrado con un coeficiente de actividad,  $A$  que representa la fracción de puertas del circuito que conmutan en cada ciclo, a una tensión determinada,  $V$ , y una frecuencia de reloj dada,  $f$ . El segundo término de (1) se debe a la corriente entre fuente de alimentación y tierra,  $I_{cortoc}$ , durante el tiempo,  $t$ , en el que conmuta la puerta. El tercer y último término es la potencia consumida debido a la corriente de pérdidas,  $I_{leak}$ , independiente del estado de la puerta lógica. Es posible consumir menos potencia utilizando una frecuencia de reloj más baja o reduciendo la tensión de alimentación (con el consiguiente efecto en la velocidad del circuito).

$$Potencia = ACV^2f + tAVI_{cortoc} + VI_{leak} \quad (1)$$

Hay que tener en cuenta que la frecuencia máxima a la que puede funcionar el circuito está relacionada con la tensión de la alimentación (2):

$$f_{max} \sim \frac{(V - V_{umbral})^2}{V} \quad (2)$$

Obsérvese que para incrementar la velocidad de procesamiento aumentando la frecuencia del procesador, habría que reducir la tensión umbral,  $V_{umbral}$ , de (2), junto con la tensión  $V$ , para mantener constante la diferencia  $(V - V_{umbral})$ , lo que aumentaría la corriente de pérdidas, que crece exponencialmente al reducirse la tensión umbral:

$$I_{leak} \sim e^{-\frac{qV_{umbral}}{kT}} \quad (3)$$

Por tanto, el incremento de la frecuencia máxima de los microprocesadores al reducir la tensión de alimentación y la tensión umbral está limitado por el aumento de la corriente de pérdidas. Un procesador puede consumir menos potencia que otro para ejecutar un programa, pero si tarda más tiempo puede ocurrir que al final la energía consumida sea mayor. Además de la energía consumida por el procesador o procesadores, existen otros factores que afectan al consumo energético en el computador y están relacionados con el funcionamiento de la memoria, los buses y los sistemas de interconexión, los dispositivos de entrada/salida, etc. Por tanto, la computación eficiente desde el punto de vista energético implica técnicas basadas no sólo en el diseño de la microarquitectura, sino también en los niveles de sistema computador y sistema operativo. En [11] se describen, con las correspondientes referencias, algunas técnicas y aproximaciones para reducir el consumo energético, y se muestra la conveniencia de incorporar e intensificar los contenidos relacionados con el consumo energético en las asignaturas de arquitectura y tecnología de computadores de grado, especialmente en los estudios de Ingeniería de Computadores.

### 4.3 Interfaces para el aprovechamiento del paralelismo heterogéneo

En cuanto al desarrollo y soporte de códigos paralelos, se necesitan interfaces que faciliten el trabajo del programador, permitiendo la expresión del paralelismo a alto nivel, la localidad, las dependencias de cómputo, los patrones de comunicación, y los efectos colaterales y las claves compartidas. Mediante este tipo de interfaces se puede utilizar un hardware más simple, con primitivas de comunicación y sincronización más eficientes a la hora de reducir la transferencia de datos.

Por otra parte, el paralelismo heterogéneo también necesita nuevas interfaces, dado que las aplicaciones deben programarse utilizando modelos de paralelismo y memoria diferentes y portables entre hardware con características específicas y diferentes, y capaces de aprovechar con cierta rapidez, las nuevas características del hardware.

También se necesitan interfaces que permitan identificar más claramente los datos de larga duración y las dependencias entre datos y código, para que el hardware y el software puedan poner de manifiesto, y de forma dinámica, los caminos críticos. La gestión de datos es más compleja en el caso de aplicaciones de *big data*, donde hay que orquestar los datos entre varios sistemas de tamaños considerables. También se necesitan interfaces capaces de especificar los límites de granularidad fina para la protección entre módulos de una aplicación y tener en cuenta criterios de seguridad como algo esencial. Algunas partes de la aplicación pueden tolerar fallos, o pueden asumir fallos con el objetivo de mejorar las prestaciones de velocidad de procesamiento. Todas estas interfaces pueden beneficiarse de mecanismos hardware apropiados como el seguimiento de flujos de información (*information-flow tracking*), bloques de recuperación transaccional (*transactional recovery blocks*), etc.

## 5 Los límites físicos de la computación

Existen límites para las prestaciones de un computador según sus características físicas. Consideremos, para empezar, la velocidad de computación máxima. Independientemente de que se sepa diseñar un supercomputador capaz de alcanzar una velocidad dada, no es inmediato pensar que haya un límite superior para la velocidad de cómputo que pueda alcanzar un computador, ni tampoco es trivial estimar dicho límite. Se podría pensar que para una masa y dimensiones físicas dadas, siempre es posible diseñar arquitecturas de cómputo y disponer de tecnologías físicas nuevas que permitan ejecutar un número cada vez mayor de operaciones por unidad de tiempo. No es trivial pensar que la velocidad de un computador no pueda crecer de forma ilimitada (suponiendo que se dispone de la energía necesaria). Igualmente, puede parecer extraño que haya un límite para el tamaño máximo de la memoria, y del paralelismo. En [12] se analizan los límites que la energía impone a la velocidad de cómputo, los que determina la entropía en el espacio de memoria, y los que el tamaño del computador establece para la paralelización. Para ello, se define un computador que denominaremos *computador esencial* con un volumen de un litro y una masa de un kilogramo. Disponer de estimaciones de los límites de la capacidad de cómputo puede ayudar en el análisis de las posibilidades que una tecnología determinada tiene para mejorar las características de los computadores actuales, y permite identificar tendencias prometedoras para la industria que, en cada momento, van a definir el paradigma dominante en la Ingeniería de Computadores. No existen garantías de que puedan alcanzarse las prestaciones máximas estimadas, pero, al menos teóricamente, surgirán tecnologías novedosas que, como la computación cuántica, mostrarán nuevas y prometedoras alternativas para seguir acercándose a ellas.

El límite máximo en la velocidad de un computador se podría estimar a partir del tiempo mínimo necesario,  $\Delta t$ , para realizar una operación lógica elemental. Por otra parte, la expresión del principio de incertidumbre de Heisenberg en términos de la incertidumbre en la energía y del tiempo,  $\Delta E$  y  $\Delta t$ , establece una cota que relaciona ambas magnitudes de incertidumbre  $\Delta E \times \Delta t \geq h/4$ , donde  $h$  es la constante de Plank. Dado que la velocidad máxima alcanzable sería, por definición  $1/\Delta t$ , tendríamos que la cota superior para dicha

velocidad sería  $(4 \times \Delta E)/h$ , operaciones por unidad de tiempo. En un computador de un kilogramo, si identificamos  $\Delta E$  con el producto de la masa y la velocidad de la luz al cuadrado  $\Delta E = m \times c^2 = 8.9874 \times 10^{16}$  J y la velocidad de cómputo máxima sería igual a  $5.4258 \times 10^{50}$  operaciones por segundo (la constante de Planck es igual a  $h = 6.6261 \times 10^{-34}$  J×s y  $c$  es la velocidad de la luz). Para estimar esta cota superior de la velocidad de cómputo, se ha tenido que identificar la energía del computador con la incertidumbre en la energía,  $\Delta E$ , y el tiempo necesario para completar una operación, con la incertidumbre en el tiempo,  $\Delta t$ . Esas identificaciones basadas en relacionar energía y procesamiento de la información, se han realizado en trabajos como [13, 14] y suponen que se es capaz de invertir toda la energía del computador en operaciones que ocasionan cambios en los estados de los bits del computador. Así, el dispositivo que realiza una operación podrá experimentar una variación de energía,  $\Delta E$ , durante el tiempo,  $\Delta t$ , que tarda en realizarla. En el caso límite que se está considerando, la mecánica cuántica establece que un sistema puede pasar de un estado *ortogonal* a otro (esto es, pasar de un estado a otro distinguible del de partida) con una diferencia en la energía igual a  $\Delta E$ , en un tiempo mínimo  $\Delta t \geq h/(4 \times \Delta E)$ . Precisamente, en [15] se muestra que un sistema cuántico con energía media  $E$ , necesita un tiempo como mínimo igual a  $h/(4 \times E)$  para evolucionar a un *estado ortogonal*. En la estimación de la velocidad máxima del *computador esencial* descrito no interviene el tipo de arquitectura del mismo, a pesar de que se podría mejorar la velocidad mediante el procesamiento paralelo. Así, si la energía,  $E$ , se distribuye entre  $N$  puertas lógicas o procesadores, cada uno de esos elementos podría realizar operaciones a una velocidad máxima igual a  $(4 \times E)/(h \times N)$ , es decir  $N$  veces menor: si la energía se distribuye entre más puertas (más procesamiento paralelo), los elementos que intervienen podrían ser más lentos. Así, el grado de paralelización de la computación a realizar puede hacer posible una distribución más eficiente de la energía entre los distintos elementos del computador. La diferencia entre la velocidad máxima del *portátil esencial* (con las dimensiones consideradas, el *computador esencial* de un kilogramo que hemos indicado podría ser un portátil) y la que alcanza un computador real de las mismas dimensiones se debe a dos causas fundamentales. En primer lugar, la mayor parte de la energía de un computador corresponde a la masa de las partículas que constituyen el computador, de forma que solo una pequeña parte de la masa está disponible para configurar la lógica de procesamiento. Además, en los computadores actuales se utilizan muchas partículas para configurar una puerta lógica o almacenar un bit (todas las partículas que constituyen los transistores de las puertas lógicas o los biestables). Precisamente, los computadores convencionales actuales aprovechan la redundancia para funcionar de manera fiable. En un computador cuántico, como se verá más adelante, no haría falta redundancia para realizar las operaciones que podrían implementarse gracias a puertas definidas por una única partícula, que proporcionaría las velocidades máximas de procesamiento, pero también hay que introducir elementos que proporcionen tolerancia a los errores inevitables en la computación cuántica, como por ejemplo los debidos a la *decoherencia*, concepto que se presentará en la Sección 6.

Las leyes de la naturaleza también marcan un límite máximo para la memoria de un computador. Este límite está relacionado con el número de estados físicos diferentes,  $\mathcal{W}$ , a los que el computador puede acceder. Este número de estados accesibles,  $\mathcal{W}$ , está relacionado con la entropía termodinámica del sistema,  $S$ , a través de la expresión

correspondiente a la ley del Boltzmann,  $S=k_B \times \ln W$ , donde  $k_B$  es la constante de Boltzmann ( $k_B=1.38065 \times 10^{-23}$  J/K). Así, un sistema con  $W$  estados accesibles podrá almacenar el equivalente a  $b = \log_2 W$  bits y, a partir de la anteriormente indicada ley de Boltzmann, se tendría que  $b=S(E)/(k_B \times \ln 2)$ , donde  $S(E)$  es la entropía termodinámica del sistema, con un valor de energía esperada,  $E$ . A partir de la expresión previa,  $(4 \times E)/h$ , para el número máximo de operaciones por segundo, se tiene que el número de operaciones por segundo por bit es  $(4 \times E)/h/b = ((4 \times E)/h) \times (k_B \times \ln 2)/S(E)$ . Por lo tanto, para obtener una estimación del espacio de memoria máximo del que podrá disponer el *computador esencial* habrá que estimar el valor de la entropía máxima de dicho computador confinado en un volumen de un litro y de una masa de un kilogramo. Para realizar ese cálculo de forma precisa se necesitaría un conocimiento completo de la dinámica de las partículas elementales presentes en el sistema, si bien se puede obtener una estimación modelando el volumen ocupado por el computador como un conjunto de modos de partículas elementales con una energía media total igual a  $E$ . También se pueden obtener distintas aproximaciones para la entropía máxima considerando los estados de funcionamiento del computador, tal y como se puede ver en [12], donde se estiman valores de  $b=2.13 \times 10^{31}$  bits. Este tamaño de memoria estimado para el *computador esencial* es muchísimo mayor que el que encontramos en los computadores portátiles actuales (del orden de los  $10^{10}$  bits) debido a que los computadores reales actuales utilizan muchos grados de libertad para almacenar un bit, por razones de controlabilidad y estabilidad. En un *computador esencial* un bit solo necesitaría un grado de libertad, pero para conseguirlo se tendría que poder transformar toda la materia en energía y la memoria sería un plasma de millones de grados Kelvin.

Tras este análisis de los límites físicos de la computación [12-15], a continuación se presentarán las principales características de la computación cuántica, que se encuentra aún en lo que puede considerarse su infancia, pero para la que existe una clara demanda en el ámbito científico y tecnológico y por ser una alternativa para resolver problemas que no se pueden abordar con computadores clásicos, constituyendo una opción prometedora para el acercamiento a los límites físicos de la velocidad del computador.

## 6 La computación cuántica

El interés por la computación cuántica surge al considerar problemas cuya complejidad los hace prácticamente irresolubles mediante *algoritmos clásicos*, pero que pueden procesarse, en tiempos relativamente pequeños, a través de los denominados *algoritmos cuánticos*. Por ejemplo, la factorización de números con una cantidad moderadamente alta de dígitos es uno de los problemas complejos para los que la computación cuántica constituye una buena alternativa. Además, puesto que este algoritmo se utiliza comúnmente en el encriptado de información, la computación cuántica tendría una incidencia considerable en las aplicaciones de ciberseguridad.

En lugar de utilizar bits, los computadores cuánticos trabajan con los denominados *qubits* como unidades de información, y aprovechan las propiedades de superposición y entrelazado (*entanglement*) cuántico. A diferencia de lo que ocurre con un bit, que

puede estar únicamente en uno de sus dos posibles estados, 0 ó 1, un *qubit* puede estar en uno de sus dos *estados base ortogonales*,  $|0\rangle$  y  $|1\rangle$ , o en una superposición de estados base que se expresa como una combinación lineal de los mismos,  $|\Psi\rangle = a|0\rangle + b|1\rangle$ , donde  $a$  y  $b$  son números complejos y se denominan amplitudes de probabilidad, que verifican  $|a|^2 + |b|^2 = 1$ . La medida de un *qubit* proyecta el estado de dicho *qubit* sobre uno de los dos estados base  $|0\rangle$  y  $|1\rangle$  con probabilidades  $|a|^2$  y  $|b|^2$ , respectivamente. Así, mientras que un sistema digital clásico con  $n$  bits puede encontrarse en uno de los posibles  $2^n$  estados, en un sistema cuántico se tendría una superposición de esos  $2^n$  estados, que se podrían almacenar, y con los que podría operar al mismo tiempo. Esta es, precisamente, la base del aumento de velocidad exponencial que podría proporcionar un computador cuántico con respecto a uno *clásico* digital.

Puerta	Matriz	Símbolo
Identidad	$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$	
Pauli-Y	$\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$	
Pauli-X	$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$	
Pauli-Z	$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$	
CNOT	$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	

Figura 3. Algunas puertas cuánticas y sus matrices correspondientes [19].

Los algoritmos cuánticos se describen en términos de circuitos constituidos por puertas (cuánticas) que operan con *qubits*, y cuyo comportamiento se puede expresar a partir de matrices unitarias  $2^n \times 2^n$ , para el caso de puertas que actúen sobre  $n$  *qubits*. La Figura 3 muestra algunas puertas cuánticas y sus matrices para el caso de dos *qubits* ([19]). En este caso se trata de matrices de  $2 \times 2$ . Por ejemplo, al aplicar la puerta Pauli-X sobre dos *qubits*  $|0\rangle$  y  $|1\rangle$  se obtiene  $|\Psi_1\rangle = 0|0\rangle + 1|1\rangle$  y  $|\Psi_2\rangle = 1|0\rangle + 0|1\rangle$ , ó  $|\Psi_1\rangle = |1\rangle$  y  $|\Psi_2\rangle = |0\rangle$ . La puerta CNOT o NOT controlada de la Figura 3, es una puerta de 2 *qubits* (la matriz es  $4 \times 4$ ) que realiza la operación X sobre el *qubit* de la entrada superior de la puerta, cuando el *qubit* de control (el de la entrada inferior) es  $|1\rangle$ . Es posible encontrar conjuntos universales de puertas cuánticas, a partir de las que se puede implementar cualquier operación cuántica. La medida del estado de una puerta cuántica *colapsa* el estado cuántico de la puerta al valor que se obtiene tras la medida (valor clásico, no cuántico). Por tanto, no es posible determinar experimentalmente, con exactitud, el estado de superposición del *qubit*.

El funcionamiento de las tecnologías cuánticas actuales se ve afectado frecuentemente por errores debidos a la denominada *decoherencia*. Es decir, se producen pérdidas de información asociadas a la interacción del sistema de cómputo cuántico que ocasionan tasas de error de alrededor de 0.01 en las puertas cuánticas. Se trata de tasas bastante mayores que las tasas de  $10^{-12}$  y  $10^{-15}$  que se requieren en los algoritmos cuánticos complejos [16]. Esto significa que debe existir una capa que se encargue de implementar procedimientos para la corrección de errores cuánticos (Quantum Error Correction Code, QECC) y mecanismos de tolerancia a fallos. En los circuitos de corrección cuánticos (QEC) la información se protege mediante un QECC específico que codifica un *qubit* lógico a través de varios *qubits* físicos imperfectos. Además, el sistema se está monitorizando constantemente para detectar posibles errores y corregir su efecto. La codificación se lleva a cabo a través del entrelazado de *qubits* y la detección de errores mediante medidas de comprobación de paridad (ESM, *Error Syndrome Measurements*). Estas ESM permiten realizar la comprobación de paridad sin una medida directa de los *qubits*, por lo que mantiene la coherencia de los estados de dichos *qubits*.

Uno de los QECC más populares es el código de superficie (*surface code*, SC) [17] que se puede implementar mediante una estructura 2D de *qubits* de dos tipos: *qubits de datos* (en los que se almacena la información) y ciertos *qubits* de apoyo denominados *qubits ancilla*, entre los que hay interacción únicamente entre *qubits* vecinos. Así, cada *qubit ancilla* interactúa con cuatro *qubits* de datos para realizar las comprobaciones de paridad entre los *qubits* de datos. A través de los *qubits* de datos y los *ancilla*, indicados anteriormente, se pueden discretizar e identificar los errores cuánticos continuos como errores de cambio de bit (X), o de cambio de fase (Z). Este proceso de identificación de los errores se implementa a través de circuitos de electrónica clásica.

Los computadores cuánticos, dado el uso que se proyecta para ellos, deben ser considerados de dominio específico, y presentan problemas no resueltos satisfactoriamente hasta el momento. Como se ha comentado anteriormente, uno de dichos problemas es la decoherencia producida por la interacción con el entorno no cuántico y que ocasiona errores de un factor de alrededor de 0.01 en las puertas cuánticas. Por lo tanto, los sistemas cuánticos deben incluir elementos de detección y corrección de errores cuya infraestructura debe escalar con el aumento de *qubits*. Actualmente hay más de 50 algoritmos cuánticos básicos, de los que el que puede considerarse más representativo es el algoritmo de Shor, que es el primero que se aplicó a problemas reales y requiere entre millones y miles de millones de *qubits* para factorizar números competitivamente grandes en relación con otros algoritmos de factorización. En la computación cuántica pueden combinarse varios *qubits* para generar un nuevo estado que es la composición de los  $2^n$  estados posibles. El estado entrelazado del *qubit* no puede descomponerse en estados separados (*entanglement*), y cuando se aplica una operación cuántica sobre los *qubits* combinados, la operación se lleva a cabo sobre los  $2^n$  estados posibles al mismo tiempo. La ganancia exponencial de velocidad se alcanzaría a través de la aplicación de un conjunto universal de operaciones cuánticas sobre *qubits* superpuestos y con entrelazado (*entanglement*).

Un computador cuántico incluye tanto elementos de computación clásica o convencional como componentes de computación cuántica. El desarrollo de un computador cuántico implica resolver retos importantes desde el punto de vista de la ingeniería de computadores, además del desarrollo de dispositivos cuánticos [18]. Desde la perspectiva de la Ingeniería de Computadores, los computadores cuánticos son computadores convencionales dotados de un coprocesador o unidad de tratamiento de datos cuántica. La interfaz del procesador cuántico con el exterior se hace a través de unidades del computador convencional, apoyándose también en unidades de este tipo para compilación de programas, almacenamiento permanente de programas y datos, etc.

De igual forma que para abordar el estudio de un computador digital convencional se recurre a una descripción del mismo a través de una serie de niveles superpuestos, en un computador cuántico también se definen una serie de capas [18, 19] que deben estar presentes y haberse desarrollado convenientemente. Estas capas van desde la de descripción de alto nivel del algoritmo cuántico (realizada mediante el correspondiente lenguaje de programación cuántico de alto nivel), hasta la capa de operaciones físicas del procesador cuántico. Las dos capas superiores incluyen los algoritmos y los compiladores que hay que desarrollar para explotar el hardware cuántico subyacente, que define la capa inferior, en la que se asume que los *qubits* y las operaciones cuánticas no sufren ningún tipo de error. En [18] se consideran distintas alternativas para implementar la capa inferior hardware en un computador cuántico. Sin embargo, en el caso de la computación cuántica, un circuito actúa sobre la superposición de los estados posibles de los *qubits*, y por ello, el talón de Aquiles de la computación cuántica es la fragilidad de los *qubits*, debido a la *decoherencia*. Como se ha comentado anteriormente, un computador cuántico debe incluir sistemas de corrección de errores cuánticos (QEC) y de tolerancia a fallos (FT). Concretamente, debajo de la capa de alto nivel, existe una capa de compilación que convierte los algoritmos cuánticos en su versión tolerante a fallos mediante el correspondiente código QEC y genera una serie de instrucciones incluidas dentro del repertorio de instrucciones cuántico (QISA), análogo al repertorio ISA en un computador digital. La infraestructura de compilación consta de un compilador que genera la lógica clásica y otro que produce los circuitos cuánticos. La capa QISA es la que separa el hardware y el software (el compilador traduce instrucciones lógicas a instrucciones físicas pertenecientes al conjunto QISA para el que el hardware cuántico proporciona soporte). Como ejemplo de instrucciones del conjunto QISA están la inicialización, las medidas de magnitudes, y otras puertas cuánticas, como la CNOT.

El bloque de ejecución cuántica (QEX) es el que se encarga de ejecutar las instrucciones QISA generadas por la infraestructura de compilación. La interfaz cuántico-clásica aplica las señales eléctricas sobre las que actúa el chip cuántico y es responsable de las conversiones entre el plano analógico de *qubits* y las capas digitales de la pila del sistema. La unidad de control cuántica (QCU) decodifica las instrucciones del QISA, y realiza las operaciones cuánticas, el control de realimentación, y el control de errores. La Figura 4 proporciona un esquema de la distribución de las capas mencionadas.

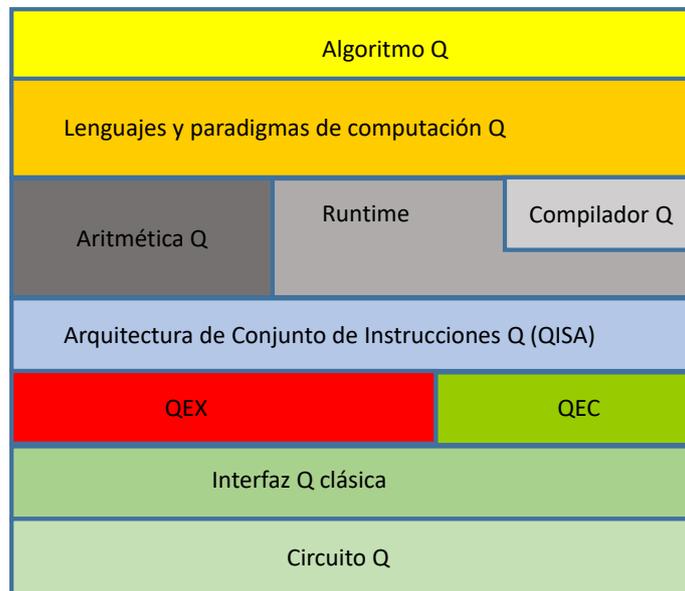


Figura 4. Organización de capas de un computador cuántico [18].

Uno de los retos que plantea el desarrollo de circuitos para los distintos módulos de los computadores cuánticos es el de aumentar el número de *qubits* por circuito integrado que pueden entrelazarse, al tiempo que se incrementa su tiempo de vida y su fidelidad, escalabilidad, y flexibilidad. La idea básica de las técnicas de control de errores cuánticos consiste en utilizar varios *qubits* imperfectos para componer unidades más fiables que se denominan *qubits* lógicos basados en códigos de corrección de error cuánticos específicos, como por ejemplo los denominados *surface codes*, en los que los *qubits* se organizan en una malla 2D regular que solo permite las interacciones más próximas y puede tolerar una tasa de error de hasta 0,01 (1%)

## 7 Una asignatura de Arquitectura de Computadores para 2020

En esta sección se propone un conjunto de contenidos esenciales para Arquitectura de Computadores, que deberían abordarse, con la extensión adecuada, en una o varias asignaturas del grado Ingeniería Informática y, por supuesto en grados o especialidades de Ingeniería de Computadores. A partir de esos contenidos se puede editar un libro de texto actualizado como referencia unificadora de la materia a impartir en cada una de las asignaturas relacionadas con Arquitectura de Computadores, muy útil en particular cuando se utilizan metodologías de docencia invertida. En dicho libro debería exponerse el cuerpo teórico aceptado para la materia, al tiempo que se reflejan los paradigmas del área de conocimiento surgidos de las prácticas usuales en el ámbito y de las líneas de investigación científica y tecnológica propias. No se propone una

distribución específica de los contenidos que se indican a continuación en asignaturas, ni tampoco se hacen estimaciones de tiempos de impartición dado que estos dependerían de la titulación de que se trate en cada caso y, por tanto, cambiarían según el contexto definido por los ámbitos y niveles de detalle que abordan el resto de asignaturas de la titulación considerada.

Para establecer los contenidos, se ha tenido en cuenta que, como se ha mostrado a lo largo del artículo, la ingeniería y la computación han experimentado cambios muy rápidos y, además existe la creencia de que se seguirá manteniendo esa tónica en el futuro, a velocidades incluso mayores. Por tanto, la formación de los ingenieros de computadores debe incluir los conocimientos fundamentales que les permitan adaptarse ágilmente a las tecnologías emergentes, e incluso ser capaces de identificar empresas y proyectos que, muy probablemente fracasarían por su dificultad para adaptarse a los inminentes cambios tecnológicos. Además, el ingeniero de computadores debe ser consciente de las consecuencias positivas y negativas del desarrollo de las tecnologías emergentes. De hecho, incluso en la introducción del área de conocimiento de *Arquitectura y Organización* en [20] se indica que los profesionales informáticos no pueden contemplar el computador como una caja negra que ejecuta programas por arte de magia, sino que también deben conocer el hardware en el que se basa la computación, sus componentes funcionales y los efectos de su interacción, para desarrollar programas con las prestaciones requeridas. También se debe disponer de un conocimiento exhaustivo de la interfaz que proporciona la arquitectura a las capas software superiores, y ser consciente de que la viabilidad de la ejecución de una aplicación en un computador no solo está relacionada con restricciones en el tiempo de ejecución y memoria, sino también con el consumo de energía.

En el libro de *Arquitectura de Computadores* de 2005 [1], el paralelismo se contempla como el paradigma que definía la evolución de la *Arquitectura de Computadores* (que traduce en capacidades reales y prestaciones, las posibilidades que ofrece la tecnología). El aprovechamiento de los distintos tipos y niveles de paralelismo sigue siendo una línea esencial para la mejora de prestaciones de las aplicaciones procesadas en un computador, y las alternativas según las que se contempla su uso cambian a medida que lo hacen la tecnología y las aplicaciones más demandadas. Por tanto, es importante disponer de procedimientos de optimización de aplicaciones en términos de módulos elementales, o de paralelismo heterogéneo. Además de estas consideraciones previas, al elaborar el contenido de un curso de *Arquitectura de Computadores* y su correspondiente libro de texto, también se deben tener en cuenta las últimas recomendaciones del IEEE y la ACM para el *Computer Engineering Curricula* de 2005 [20], y la guía docente de la asignatura en el grado de Ingeniería Informática de la UGR [21]. Igualmente, también es relevante revisar los contenidos de las ediciones más recientes de libros de texto de referencia en la materia, como [2]. Según esto, una propuesta de contenidos para *Arquitectura de Computadores* y, consecuentemente, de posibles capítulos a incluir en el correspondiente libro de texto es la siguiente:

1. Fundamentos cuantitativos de *Arquitectura de Computadores*.
2. Repertorios de instrucciones e Interfaces
3. Evaluación de prestaciones.

4. Programación paralela y jerarquía de memoria.
5. Procesadores con paralelismo entre instrucciones (ILP)
6. Procesadores con paralelismo a nivel de hebra (TLP)
7. Procesadores con paralelismo de datos (DLP): GPUs. Procesadores Vectoriales, Procesadores SIMD.
8. Computadores de dominio o uso específico
9. Tecnologías hardware y perspectivas de la evolución de los computadores

Para empezar, se abordarían los aspectos relacionados con la caracterización cuantitativa del computador, describiendo las magnitudes utilizadas para analizar sus prestaciones, tanto en cuanto a velocidad de procesamiento, como en potencia instantánea, o en energía consumida, e incluyendo información sobre los distintos tipos de computadores según las aplicaciones que ejecutan más eficientemente y los mercados que los demandan. También se abordaría la diferencia entre retardo o latencia y ancho de banda.

A continuación se introduciría el concepto de Arquitectura del Computador desde distintas perspectivas. Por un lado como repertorio de instrucciones máquina que definen una interfaz entre el hardware y el software del computador, o como diseño y organización del hardware para satisfacer los requisitos funcionales del computador.

El tercer capítulo presentaría conceptos relacionados con el procesamiento y la programación paralela, y describiría las magnitudes utilizadas para evaluar las mejoras que proporciona el paralelismo (ganancia de velocidad, eficiencia, etc.) junto con sus limitaciones. A través del estudio de las leyes de Amdahl y Gustafson se introducen límites para la ganancia de velocidad, que permiten estimar las mejoras alcanzables en función de los perfiles de paralelismo de las aplicaciones consideradas. Indudablemente, como se ha indicado más arriba, el paralelismo, sus alternativas, implementaciones, y formas de aprovechamiento, seguirán siendo parte esencial de las asignaturas de Ingeniería de Computadores. Así, independientemente de cómo se organicen los contenidos, deberían incluirse temas dedicados al paralelismo de instrucciones (ILP), de datos (DLP), y de hebras (TLP), y debería mantenerse el estudio los computadores paralelos MIMD considerando las estructuras de interconexión entre procesadores memorias y dispositivos de E/S, las clasificaciones de multiprocesadores y multicomputadores según tengan una jerarquía de memoria compartida o distribuida, y los conceptos de coherencia y consistencia de memoria junto con su relación con la sincronización y comunicación en el multiprocesador.

También es esencial el estudio de las interfaces presentes en el computador desde el punto de vista de la garantía de seguridad que ofrecen (por ejemplo frente a ataques como *Meltdown* o *Spectre*) dado el funcionamiento de las microarquitecturas actuales con paralelismo ILP, que incorporan, entre otros, recursos de procesamiento especulativo para la mejora de rendimiento de velocidad y consumo energético. Finalmente, el estudio de APIs que faciliten el aprovechamiento de arquitecturas heterogéneas resulta esencial, junto con el de las tecnologías físicas que constituirán la base de computadores cuya implementación empieza a considerarse en la actualidad. Un ejemplo lo tenemos en entornos como *OpenCL*, que permiten la programación

paralela independiente de la plataforma a través de programas que se procesan en un *host*, que envía funciones, denominadas *kernels* o núcleos, para que sean ejecutadas en otros dispositivos, definidos previamente como dispositivos *OpenCL* (*OpenCL devices*) que pueden implementar arquitecturas diferentes (CPU o GPU). De esta forma se pueden definir plataformas paralelas heterogéneas para las que deben abordarse problemas complejos de distribución de carga, no sólo para optimizar la ganancia de velocidad sino también el consumo de energía. No obstante, junto a estos entornos alternativos y algo más específicos, como el mencionado *OpenCL*, también existen otros, como *OpenMP* muy extendidos, y con implementaciones eficientes realizadas por, prácticamente, todos los fabricantes. Además, podría utilizarse en prácticamente todas las asignaturas de Arquitectura de Computadores.

Una aproximación que siempre se contempla para desarrollar computadores más eficientes que los disponibles en cada momento, es la de buscar nuevas tecnologías físicas que superen las limitaciones que se observan en las utilizadas y validadas en el presente. En [7] se describen algunos de los dispositivos alternativos a los de tecnología CMOS, y en [18, 19] se referencian tecnologías y dispositivos que se están considerando para conformar la capa de tecnología física en el diseño de computadores cuánticos. Dado que en el futuro es previsible que el desarrollo de dispositivos implementados en nuevas tecnologías físicas sea una alternativa muy frecuente a la hora de plantear nuevas mejoras en Ingeniería de Computadores, es conveniente mantener en los currícula, asignaturas dedicadas a las tecnologías físicas que se están considerando y a las que se prevé utilizar para conformar los niveles hardware de los computadores futuros.

## 8 Conclusiones

Al aproximarse el agotamiento de la ley de Moore han aparecido nuevos paradigmas para marcar la hoja de ruta del desarrollo de la Ingeniería de Computadores, y como consecuencia, se inducirán cambios tanto en los contenidos, como en su distribución entre las asignaturas de las nuevas titulaciones que habrá que plantear. En la mayoría de los casos, en lugar de definir conjuntos de programas de prueba que caractericen el uso más frecuente de un computador y usar magnitudes como las velocidades pico para comparar arquitecturas, se pueden utilizar directamente las aplicaciones reales consideradas, cuyos códigos implementan clasificadores y/o aproximadores universales como las redes neuronales profundas, y también metaheurísticas de optimización, como las basadas en algoritmos evolutivos. Si las aplicaciones de interés se implementan a partir de modelos neuronales, o metaheurísticas evolutivas, pueden ser bastante efectivas para obtener medidas de prestaciones realistas. Además, dado que poseen estructuras computacionales basadas en la interconexión de elementos de cómputo relativamente sencillos y parecidos entre sí, su uso contribuye a evitar el problema de la recurrencia en Ingeniería: gran parte de los procedimientos y alternativas de optimización que se han analizado y desarrollado para unas aplicaciones, se pueden aprovechar en otras diferentes para las que, sin embargo, también se utilizan aproximadores o clasificadores basados en redes neuronales profundas, o

metaheurísticas. No es casual el reciente interés, en el ámbito de la Ingeniería de Computadores, por el desarrollo de aplicaciones realistas a través de redes neuronales profundas y en su implementación eficiente a través de arquitecturas de uso específico que, por otra parte permiten un aprovechamiento eficiente del paralelismo implícito en la aplicación.

También despiertan un enorme interés aproximaciones, en principio prometedoras, con objeto de abordar problemas que caen fuera del alcance de la computación *clásica*. De hecho, en la Sección 6 de este artículo proporcionamos una aproximación a la computación cuántica. Por un lado, los computadores cuánticos vienen despertando un interés considerable debido a las posibilidades que ciertos algoritmos cuánticos ofrecerían en aplicaciones inabordables mediante algoritmos y computadores clásicos, gracias al paralelismo implícito que podría aprovecharse a través de conceptos de computación cuántica que, desde hace bastante tiempo se consideran una solución adecuada para superar las prestaciones que podrían proporcionar los computadores electrónicos debido a los límites que la minimización de los transistores impondría en su ritmo de mejora de prestaciones. Por otro lado, la integración de computadores cuánticos dentro del desarrollo de la ingeniería de computadores los sitúa más bien como coprocesadores para los que habrá que desarrollar interfaces que los ubiquen dentro de los computadores existentes. Nuestro propósito al referir aquí algunos detalles de la computación cuántica no es sugerir que se incluyan asignaturas dedicadas a la implementación de computadores cuánticos en los planes de estudios actuales para estudios de grado de Informática o Ingeniería de Computadores. Se ha buscado únicamente poner de manifiesto el interés de impartir contenidos relacionados con las tecnologías emergentes que puedan ser útiles para el desarrollo de computadores cada vez más potentes, al tiempo que, en asignaturas que abordan perspectivas de la Ingeniería de Computadores, se ilustra una vez más la importancia del estudio de interfaces para integrar y programar nuevas aplicaciones en los computadores al uso. Si bien es relevante abordar conceptos generales relacionados con la necesidad de interfaces eficientes y la integración de nuevas tecnologías para mantener niveles adecuados de mejora en las Arquitecturas de Computadores, el estado actual de desarrollo de los computadores cuánticos no aconseja la inclusión de asignaturas dedicadas a la computación cuántica en estudios de grado de Ingeniería de Computadores. Por esa misma razón, tampoco encontramos necesario introducir temas relacionados con los computadores cuánticos en un libro de texto sobre Arquitectura de Computadores. La perspectiva, y los problemas que se abordarían desde dicha perspectiva serían similares a los que habría que considerar a la hora de estudiar, en cada momento, otras aproximaciones al desarrollo de coprocesadores de dominio específico que ofrezcan prestaciones elevadas en ámbitos de aplicación muy demandados. Un ejemplo de esta situación lo constituye la computación neuromórfica, y el uso de coprocesadores para redes neuronales profundas, que están suscitando un interés considerable en la actualidad.

**Agradecimientos.** Este artículo ha sido financiado por el proyecto PGC2018 - 098813 – B – C - 31 (Ministerio de Economía y Competitividad” y fondos FEDER).

## 9 Referencias

1. Ortega, J.; Anguita, M.; Prieto, A.:”Arquitectura de Computadores”. Ed. Thomson-Paraninfo, 2005.
2. Hennessy, J.L.; Patterson, D. A.: “Computer Architecture: A quantitative approach (Sixth Edition)”. Morgan Kaufmann, 2019.
3. Jouppi, N.P. et al.:”In-Datacenter performance analysis of a Tensor Processing Unit”. 44<sup>th</sup> Symp. On Computer Architecture (ISCA). Junio, 2017.
4. Chien, A.: “Computer Architecture: disruption from above”. Communications of the ACM, Vol. 81, No.9, pp.5, Septiembre, 2018.
5. Hennessy, J; Patterson, D.:”2017 Turing Award Lecture” (ISCA 2018). <https://www.acm.org/hennessy-patterson-turing-lecture>.
6. Conte, T.M.; DeBenedictis, E.P.; Gargini, P. A.; Track, E.: “Rebooting Computing: The Road Ahead”. Computer, pp. 20-29. Enero, 2017.
7. Bourianoff, G.; Brewer, J.E.; Cavin, R.; Hutch, J. A.; Zhirnov; V.:”Boolean Logic and alternative information-processing devices”. IEEE Computer, pp. 38-46, 2008.
8. 21st Century Computer Architecture. A community whitepaper, <https://cra.org/ccc/wp-content/uploads/sites/2/2015/05/21stcenturyarchitecturewhitepaper.pdf>, 2012.
9. Lipp, M., et al.: “Meltdown”, 2018.
10. Kocher, P., et al.:”Spectre attacks: exploiting speculative execution”, 2018.
11. Díaz García, A.F.; et al. Consumo de energía y asinaturas de arquitectura y tecnología de computadores. Enseñanza y Aprendizaje de Ingeniería de Computadores, 7: 79-92 (2017). [<http://hdl.handle.net/10481/47374>]
12. Lloyd, S.:”Ultimate physical limits to computation”. arXiv: quant-ph/9908043v3, 14 Feb 2000
13. Landauer, R.:”Irreversibility and heat generation in the computing process”. IBM J., pp.183-191, 1961.
14. Plenio, M. B.; Vitelli, V.: ”The physics of forgetting: Landauer’s erasure principle and information theory”. Contemporary Physics, Vol.42, No.1, pp.25-60, 2001.
15. Margolus, N.; Levitin, L.B.:”The maximum speed of dynamical evolution”. Physica D: Nonlinear Phenomena 120 (1-2), pp.188-195, 1998.
16. Xu.; et al.:”A heterogeneous quantum computer architecture”, ACM CF’16, pp.323-330, 2016.
17. Fowler, et al.:”Surface code: Towards practical large-scale quantum computation. Phys. Rev. A., 86(3):032324, 2012.
18. Jones, N.C., et al.: “Layered architecture for quantum computing”. arXiv:10105022v3, 2012.
19. Almudever, C.G.; et al.:”The engineering challenges in Quantum Computing”. IEEE 2017 Design, Automation and Test in Europe (DATE), pp.837-845. 2017.
20. IEEE/ACM Computer Engineering Curricula 2016: <http://www.acm.org/education/curricula-recommendations>.
21. Guías docentes del Grado en Ingeniería Informática de la Universidad de Granada: [http://grados.ugr.es/informatica/pages/infoacademica/guias\\_docentes/guiasdocentes\\_curso\\_actual](http://grados.ugr.es/informatica/pages/infoacademica/guias_docentes/guiasdocentes_curso_actual).



## **Modelado matemático, simulación y control de una mini motocicleta autónoma con rueda de reacción**

J.C. Saavedra, A. Olivares, G. Olivares

Departamento de Arquitectura y Tecnología de Computadores  
Universidad de Granada. Granada, Spain

Juancarlosst97@correo.ugr.es, a.olivares@gmail.com, gonzalo@ugr.es

**Resumen.** Se presenta el modelado, la simulación y el control de una mini motocicleta autónoma, estabilizada mediante rueda de reacción, que se utilizará para la enseñanza del diseño de sistemas de control. El modelo se construye a partir de un diseño CAD para posteriormente ser integrado en Simulink, junto con los módulos de control. Se realiza la modelización del sistema completo, incluyendo elementos mecánicos, sensores, actuadores, así como la dinámica de contacto de las ruedas con el suelo, consiguiéndose un comportamiento muy similar al de la motocicleta física. A partir del modelo matemático (ecuaciones diferenciales, funciones de transferencia y ecuación de estado), se diseñan un controlador PI para la velocidad, y varios controladores para la inclinación (PID, LQR y LQI). Los controladores han sido probados también en la motocicleta física.

**Palabras Clave:** Motocicleta autónoma, Control Digital, Control PID, control LQR, LQI, Simscape, Aprendizaje Experimental.

**Abstract.** In this paper, the modelling and subsequent control of an autonomous mini motorcycle, which will be used to teach about control systems, is presented. The model is initially built from a CAD design and then integrated into Simulink, together with the control modules. The modelling of the complete system, including the mechanical parts, sensors, actuators and wheels behaviour with the ground is carried out, to achieve the same behaviour as the physical motorcycle. From the mathematical model (differential equations, transfer functions and state equation), a PI controller is designed for speed, and several controllers for inclination (PID, LQR and LQI). The controllers have also been tested on the physical motorcycle.

**Keywords:** Autonomous Motorcycle, Digital-Control, PID controller, State space controller, LQR control, Simscape Multibody Simulation, Experimental Learning.

## 1 Introducción

El principal objetivo de este artículo es diseñar un modelo virtual de una motocicleta auto-balanceada que sea lo más fiel posible a una mini motocicleta física, suministrada en el *Arduino Engineering kit*, con el fin de evaluar si este sistema es idóneo para utilizarse en las clases prácticas de las materias de control impartidas en la Universidad de Granada. El uso de modelos virtuales en la enseñanza de sistemas de control tiene varias ventajas (bajo coste, eliminación de coste mantenimiento, posibilidad de trabajar con grupos numerosos de prácticas y de que el alumno pueda trabajar desde casa, etc), pero también el inconveniente de que a veces el resultado se puede alejar de la realidad.

Representaremos el modelo de la motocicleta en un entorno gráfico denominado “*Simscape Multibody*” incluido en las librerías de las últimas versiones de Simulink. Este método de simulación gráfica tiene la ventaja de permitir la verificación del comportamiento del controlador y ver el resultado del movimiento de sistemas mecatrónicos tal y como se produciría en los sistemas reales, incorporando también módulos virtuales de controladores.

Para la creación del modelo mecánico se realiza inicialmente el diseño gráfico con un software CAD (en nuestro caso Solidworks), y posteriormente se exporta a Simulink. A continuación es necesario realizar el diseño del controlador, y las modificaciones necesarias en el modelo para conseguir un sistema que sea fiel a la motocicleta física.

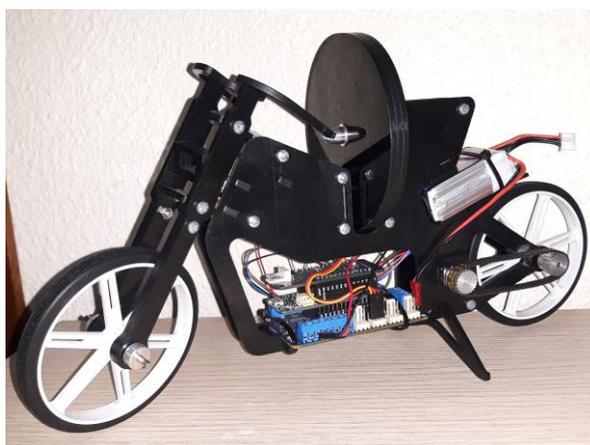
A lo largo de este documento, se muestra cómo tratar las ecuaciones que rigen el comportamiento de la motocicleta para realizar un controlador PI de velocidad y dos controladores diferentes para mantener la inclinación de la motocicleta deseada, siendo el primer controlador de inclinación un PID y el segundo de ellos un controlador LQI en el espacio de estados. Para comprobar los resultados, los controladores son probados también sobre la motocicleta física, lo que nos permitirá sugerir varias mejoras en su diseño.

El modelo de la motocicleta puede ser utilizado para aumentar el interés de los estudiantes, ya que es muy llamativo y motivador realizar simulaciones del controlador diseñado y ver su comportamiento en un modelo que representa de manera bastante fiel al modelo real. El trabajo realizado pretende servir de base para la futura construcción de una motocicleta autónoma de tamaño real.

Son muchos los autores que han realizado un estudio de este problema, a partir de un modelado matemático [1-8]. En nuestro caso, pretendemos usar un modelo lo más sencillo posible, para poder usarse en la enseñanza prácticas de asignaturas de control.

## 2 Motocicleta física.

La motocicleta empleada cuenta con todos los componentes necesarios que permiten su conducción de forma autónoma. Dicha motocicleta está compuesta por una estructura de ABS cortada con fresadora. Para conseguir el movimiento hacia adelante y hacia atrás se emplea un motor DC que lleva acoplado un encoder, lo que nos permitirá regular su velocidad. Para conseguir que la moto se equilibre se utiliza un motor DC de alta velocidad acoplado a una rueda de reacción, cuya velocidad angular es medida mediante un tacómetro.



**Figura 1.** Fotografía de la mini-motocicleta real

La tarjeta de control es del tipo *Arduino MKR1000*, que incorpora comunicación WiFi. Para el control de motores y servos se usa la tarjeta *MKR Motor Shield*.

Se emplea además una IMU (Unidad de Medición Inercial) *Bosch BNO055* para medir la inclinación de la motocicleta, que integra acelerómetro ejes, giróscopo y magnetómetro de 3 ejes.

Para mover la rueda de reacción, que va a estabilizar la motocicleta, se emplea un motor DC de alta velocidad. La velocidad de giro de este motor se mide con dos sensores de efecto Hall (*SL353HT de Honeywell*) y unos pequeños imanes que giran solidarios al eje del motor.

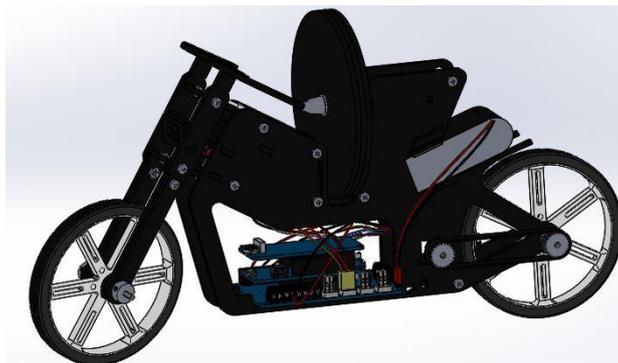
El motor empleado para realizar el movimiento de avance y retroceso de la motocicleta es también un motor DC que lleva acoplado una caja de engranajes para conseguir un mayor par motor y adecuar su velocidad a la requerida para el movimiento de la motocicleta. Este motor lleva acoplado un encoder en cuadratura para medir el ángulo girado y la velocidad angular del motor (derivando el ángulo). Conocida la velocidad angular del motor (y de la rueda) y el radio de la rueda, se obtiene la velocidad lineal de la motocicleta.

### 3 Modelado gráfico

El primer paso es realizar el proceso de modelado gráfico. Para crear un modelo que sea una representación realista del sistema se ha empleado el software de diseño *Solidworks*. Una vez que el modelado ha sido realizado, éste es exportado a un software de simulación que permite simular el comportamiento motriz del modelo creado anteriormente. Dicho software es “*Simscape Multibody*”, una extensión de *Simulink*.

#### 3.1 Diseño CAD

Inicialmente se diseñó un modelo CAD muy realista, pero a la vez demasiado complejo para ser modelado en Simulink, ya que incluía una gran cantidad de elementos mecánicos, con excesivos grados de libertad. Esto provoca que la potencia de cálculo necesaria sea excesiva y lenta de ejecutar en un computador convencional. Es fundamental crear cada parte con las mismas dimensiones, la misma masa, y la misma densidad para que así, las propiedades físicas del modelo sean las mismas que las de la motocicleta física.



**Figura 2.** Modelo inicial

A partir de este primer modelo se construyó otro modelo, que conserva las mismas características físicas que el inicial, pero cuya complejidad se ve reducida. Para conseguir esta reducción de piezas se emplea el comando “unir” disponible en *Solidworks*. Este comando permite, a partir de dos sólidos que estén en contacto, crear un único nuevo sólido, que posee la combinación de las características físicas que poseían los sólidos que lo forman. Gracias a esto, el modelo de la motocicleta queda reducido a siete elementos fundamentales. En la figura 3 se muestra el modelo gráfico simplificado.

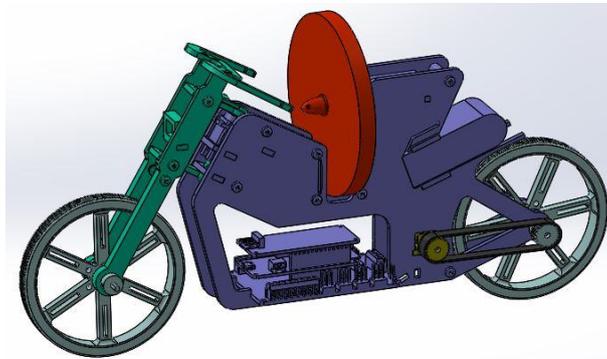


Figura 3. Modelo gráfico simplificado

### 3.2 Exportación del diseño CAD a Simulink

Este modelo puede ser importado a Simscape Multibody, para ello es necesario utilizar el elemento “*Simscape Multibody Link Inventor Plug-In*”. Una vez instalado este complemento, es necesario abrir el modelo en *Solidworks*, ir a *Simscape Multibody Link* y seleccionar la opción de exportar a *Simscape Multibody*. En la figura 4 se puede ver el diagrama de bloques del sistema.

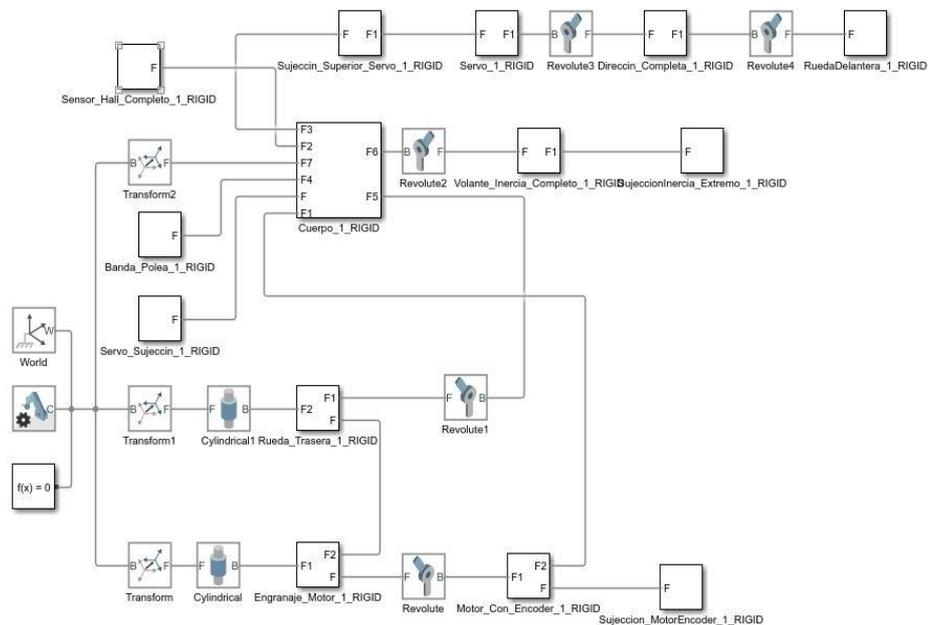
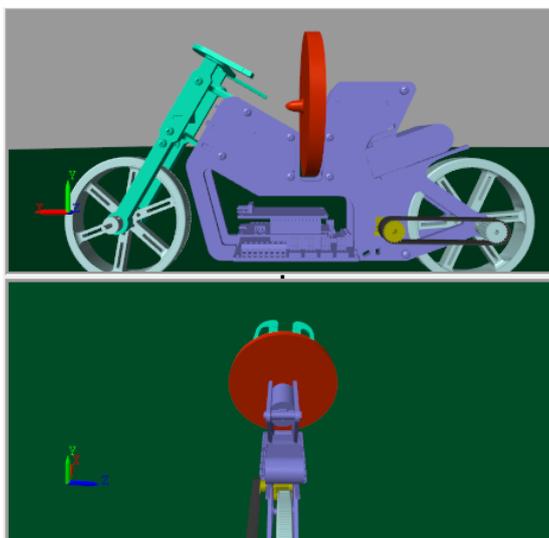


Figura 4. Diagrama de bloques *Simscape* del modelo simplificado de la motocicleta

Para modelar el contacto entre las ruedas y el suelo, se emplea una librería adicional, denominada “*Simscape Multibody Contact Forces Library*” que hay que añadirla manualmente. Esta librería permite simular el comportamiento que se produce entre cualquier tipo de superficie cuando una de ellas está fija y otra de ellas se mueve. En el caso de este modelo, se emplea el tipo *Sphere to Plane Force*. Mediante esta forma de contacto se simula la forma en la que una esfera se mueve a lo largo de un plano.



**Figura 1.** Modelo en Simscape Multibody

#### 4 Modelo matemático

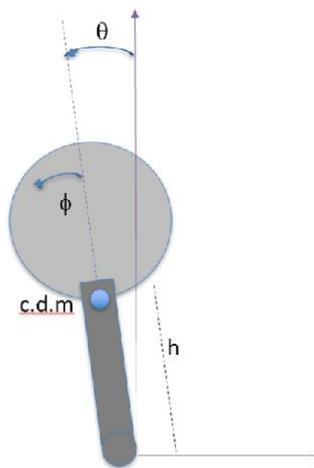
Vamos a considerar que la motocicleta sólo se estabiliza mediante la rueda de reacción. Inicialmente consideramos que la motocicleta se mueve en línea recta, y el ángulo del manillar no influye en la estabilidad, ni en la inclinación. Otro modelo que tienen en cuenta la inclinación producida por la variación de dicho ángulo se puede consultar en [2].

Tal y como se aprecia en la figura 6, la inclinación con respecto a la vertical se mide con el ángulo  $\theta$  y la rueda de reacción gira con un ángulo  $\phi$ . La distancia entre el suelo y el centro de masas, la denominamos  $h$ , y  $m$  es la masa de la motocicleta.

La Energía cinética de rotación viene dada por la expresión:

$$T = \frac{1}{2} I_M \dot{\theta}^2 + \frac{1}{2} J \dot{\phi}^2$$

Donde  $I_M = mh^2$  es el momento de inercia de la motocicleta con respecto a la vertical, y  $J$  es el momento de inercia de la rueda de inercia, gobernada por un motor DC de alta velocidad.



**Figura 6.** Esquema de la motocicleta

La energía potencial es:  $V = mgh \cos \theta$

El lagrangiano será:

$$L = T - V = \frac{1}{2} I_M \dot{\theta}^2 + \frac{1}{2} J \dot{\phi}^2 - mgh \cos \theta = \frac{1}{2} mh^2 \dot{\theta}^2 + \frac{1}{2} J \dot{\phi}^2 - mgh \cos \theta$$

Resolviendo las ecuaciones de Euler-Lagrange, se obtienen las ecuaciones diferenciales del sistema:

$$\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\theta}} \right) - \frac{\partial L}{\partial \theta} = -\tau \quad \rightarrow \quad mh^2 \ddot{\theta} - mgh \sin \theta = -\tau$$

$$\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\phi}} \right) - \frac{\partial L}{\partial \phi} = \tau \quad \rightarrow \quad J \ddot{\phi} = \tau$$

donde  $\tau$  es el par del motor que mueve la rueda de reacción.

Linealizando se obtiene la función de transferencia:

$$G(s) = \frac{\theta(s)}{\tau(s)} = -\frac{1/mh^2}{s^2 - \frac{g}{h}}$$

y la ecuación de estado: 
$$\begin{pmatrix} \dot{\theta} \\ \ddot{\theta} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ \frac{g}{h} & 0 \end{pmatrix} \begin{pmatrix} \theta \\ \dot{\theta} \end{pmatrix} + \begin{pmatrix} 0 \\ -\frac{1}{mh^2} \end{pmatrix} \tau$$

Si incorporamos el motor, de constante K (y consideramos que la constante de par y la constante electromotriz son iguales), tendremos finalmente:

$$\begin{aligned} \ddot{\theta} &= \frac{g}{h} \theta + \frac{K^2}{mh^2 R} \dot{\phi} - \frac{K}{mh^2 R} V \\ \ddot{\phi} &= -\frac{K^2}{RJ} \phi + \frac{K}{RJ} V \end{aligned}$$

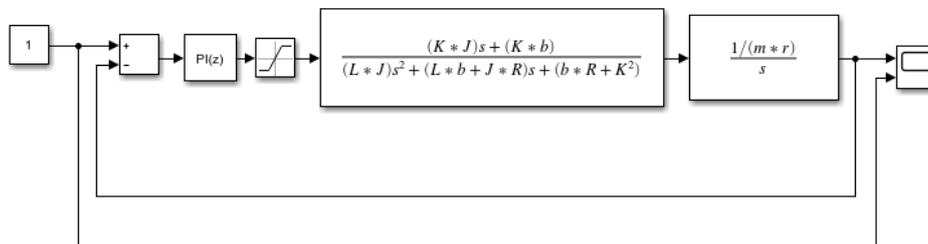
$$\begin{pmatrix} \dot{\theta} \\ \ddot{\theta} \\ \ddot{\phi} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ \frac{g}{h} & 0 & \frac{K^2}{mh^2 R} \\ 0 & 0 & -\frac{K^2}{RJ} \end{pmatrix} \begin{pmatrix} \theta \\ \dot{\theta} \\ \phi \end{pmatrix} + \begin{pmatrix} 0 \\ -\frac{K}{mh^2 R} \\ \frac{K}{RJ} \end{pmatrix} V$$

**Ecuación 1.** Ecuación de estados para la motocicleta.

A partir de la función de transferencia y de la ecuación de estados podremos diseñar varios tipos de controladores (PD, LQR, LQI, etc.), como veremos más adelante.

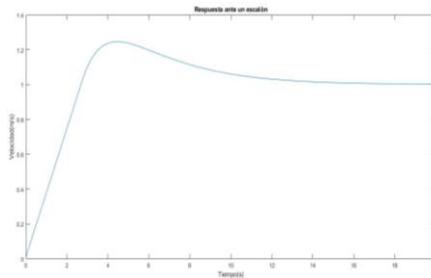
#### 4.2 Controlador PI para el control de la velocidad de avance

Para realizar un buen control de la velocidad de la motocicleta es necesaria una respuesta progresiva y sin oscilaciones, con el fin de evitar grandes desequilibrios en la motocicleta. A partir de los parámetros del motor de avance podemos calcular su función de transferencia y diseñar el controlador Proporcional-Integral discreto correspondiente. El ajuste de las constantes proporcional e integral se ha realizado automáticamente mediante la opción “Auto Tune” del módulo PID de *Simulink*. La figura 7 muestra el lazo del control de velocidad.



**Figura 7.** Diagrama de bloques para el control de velocidad

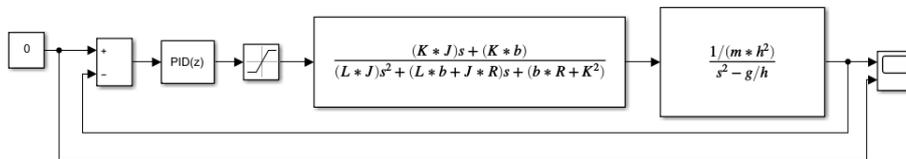
A continuación, el controlador PI de velocidad puede ser testeado. El resultado de introducir una entrada de tipo escalón se presenta en la figura 8. Los parámetros del controlador utilizados son:  $K_p=35.24$  y  $K_i=3$ .



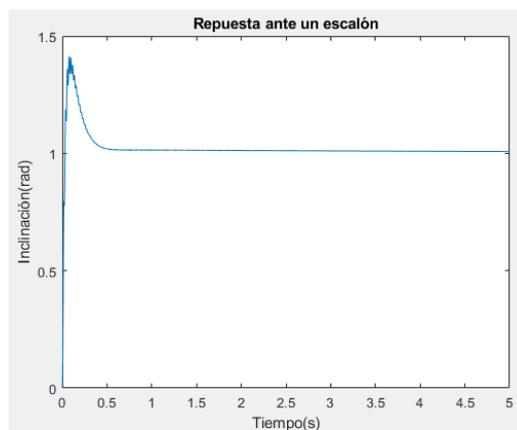
**Figura 8.** Respuesta del controlador PI de velocidad

### 4.3 Controlador PID de inclinación.

A continuación se diseñó un controlador PID de la inclinación de la motocicleta. Para esto, se ha empleado el diagrama de bloques que se presenta en la figura 9. Se ha tenido en cuenta la función de transferencia del motor de la rueda de reacción, junto con la función de transferencia que relaciona el par con la inclinación.



**Figura 9.** Control de inclinación en bucle cerrado



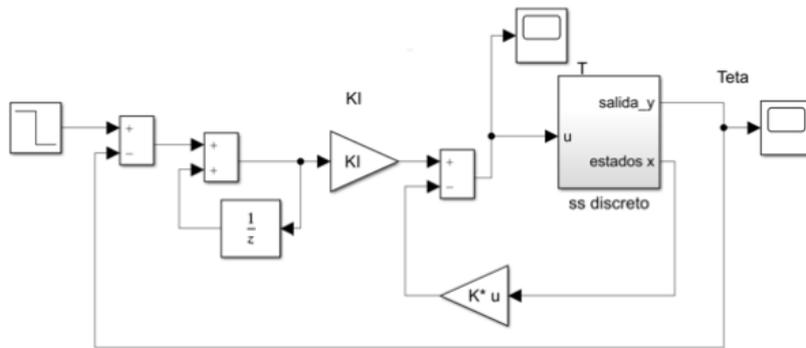
**Figura 10.** Respuesta ante un escalón unitario.

Para este controlador, el valor de las constantes del controlador son  $K_p=307.80$ ,  $K_i=1148.85$ ;  $K_d=19.164$ ;  $N=191.52$ .

Un video de la simulación puede verse en [9].

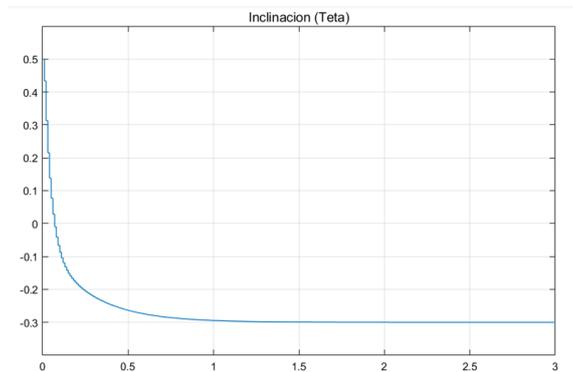
#### 4.4 Controlador en el espacio de estados para la inclinación.

Además del controlador PID de la inclinación se diseñó un controlador LQR en el espacio de estados, a partir de la Ecuación 1 que obtuvimos en el apartado 4.1. La respuesta de este controlador mantiene un error permanente ante un cambio de consigna diferente a la verticalidad. Este problema se ha corregido utilizando la configuración LQI, que incorpora un integrador discreto, como se puede apreciar en la figura 11:



**Figura 11.** Control LQI de la inclinación de la motocicleta

Los resultados obtenidos sobre el modelo Simscape de la motocicleta, ante un cambio de la consigna de inclinación desde 0.5 radianes hasta -0.3 radianes, se pueden ver en la Figura 12.



**Figura 12.** Respuesta del controlador LQI

Un vídeo de la simulación con la motocicleta virtual en movimiento puede verse en [10].

## **5. Control en tiempo real de la motocicleta física.**

Para finalizar, los controladores diseñados se han adaptado en *Simulink* para poder ejecutarse en tiempo real sobre la tarjeta *Arduino MKR1000*. Ello nos permitirá comprobar o no, si el modelo físico es apropiado para ser usado en el laboratorio de prácticas, con los alumnos.

### **5.1 Consideraciones previas**

Es posible programar la tarjeta MKR1000 directamente desde *Simulink*, usando la librería *Simulink Support Package for Arduino Hardware*, lo que permite no tener que realizar grandes cambios en los controladores diseñados previamente. Sin embargo, al diagrama de bloques detallado en apartados anteriores se le deben añadir los elementos necesarios para poder leer los diferentes sensores de los que dispone la motocicleta y los necesarios para manejar los actuadores.

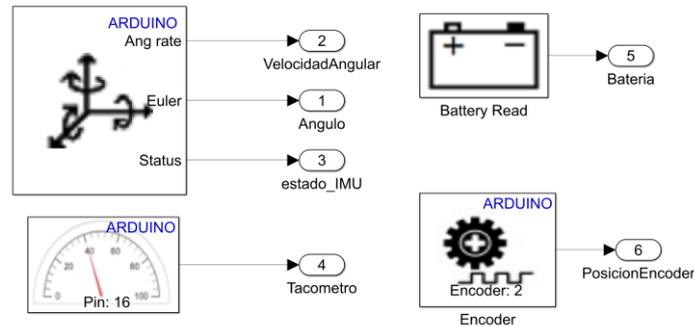
Otra de las ventajas de la tarjeta MKR1000 es que dispone de comunicación WiFi, lo que nos permite visualizar los datos obtenidos por los sensores de la motocicleta, y cambiar las consignas en tiempo real, sin necesidad de usar cable para conexión USB. Así podemos supervisar el control y modificar la velocidad y la trayectoria de la motocicleta a distancia.

### **5.2 Lectura de sensores.**

Los sensores empleados en la motocicleta son el tacómetro, que permite medir la velocidad angular de la rueda de reacción, el encoder, que se encarga de determinar la velocidad de giro del motor de avance, y la IMU, que es la unidad encargada de indicar la inclinación de la motocicleta. Todos estos elementos (ver Fig.13) se suministran en el *Arduino Engineering Kit* para ser incorporados en la librerías de *Simulink*.

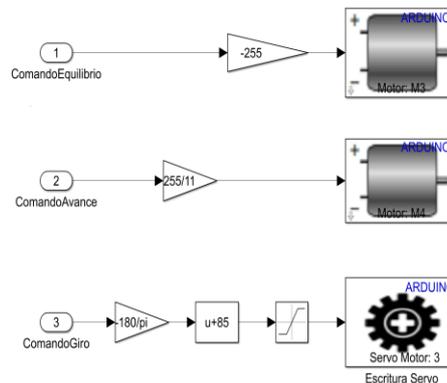
### **5.3 Actuadores.**

Para que el modelo funcione correctamente, es importante también realizar una correcta escritura de las variables calculadas por los controladores. En el caso de los motores de avance y de la rueda de reacción se trata de una señal PWM y en el caso del servomotor de dirección se requiere indicar el ángulo de giro, acotado según las restricciones del servo.



**Figura 13.** Módulos *Simulink* para la lectura de sensores

El bloque para los motores DC que incluye la librería “*Simulink Support for Arduino MKR Motor Carrier*” acepta valores que van entre 255 y -255, siendo estos los máximos valores de giro en un sentido y en el contrario. En el caso del servomotor de dirección, su control se realiza introduciendo directamente el ángulo que se desea girar, teniendo un rango comprendido entre 0 y 180 grados.



**Figura 13.** Módulos *Simulink* para los actuadores

#### 5.4 Control de la motocicleta física.

Los controladores diseñados en las secciones anteriores han sido trasladados a la motocicleta física sin cambiar sus parámetros. En la figura 14 se puede ver el esquema general de control de la motocicleta en tiempo real, realizado mediante *Simulink*.

No se ha realizado el cambio ya que, como se comprobó en el modelo virtual, su comportamiento era el deseado. Sin embargo, al realizar el control en la motocicleta física los resultados cambian eventualmente, y al cabo de unos segundos el sistema se

desestabiliza. Ello se debe a varias razones, relacionadas con el diseño de la motocicleta física, que exponemos a continuación.

La primera razón se debe al comportamiento del motor de la rueda de reacción. Cuando a este motor se le exigen cambios en la dirección de giro (funcionamiento fundamental para mantener la verticalidad) se produce un sobrecalentamiento excesivo del mismo. Esto provoca que sea necesario detener el sistema antes de que el motor pueda dañarse. Además, la medida de la velocidad angular de la rueda de reacción tiene muy poca precisión, ya que incorpora un tacómetro excesivamente simple. Sería conveniente incluir un motor con encoder incorporado.

Otro motivo que influye drásticamente en el control es el continuo cambio del punto de equilibrio del sistema. Es decir, al realizar diferentes pruebas, el punto de equilibrio varía ya que, fundamentalmente, la posición de la batería no es constante, y la estructura de plástico no es suficientemente rígida. Por todo ello es necesario realizar una recalibración de la IMU cada vez que se inicia de nuevo el proceso de control.

Por último, la IMU que el sistema incorpora en el kit no cuenta con las prestaciones óptimas necesarias para obtener una medida fiable de la inclinación. Esta IMU realiza el filtrado de la señal internamente con un procesador poco potente, lo que limita la frecuencia de muestreo, haciendo así que las medidas de la inclinación no sean precisas. En un próximo diseño, para medir la inclinación incorporaremos algoritmos de fusión de datos del acelerómetro y giróscopo que utilicen filtros de Kalman, con más potencia de cálculo, y en los que las matrices de covarianza  $Q$ ,  $R$  puedan ajustarse.

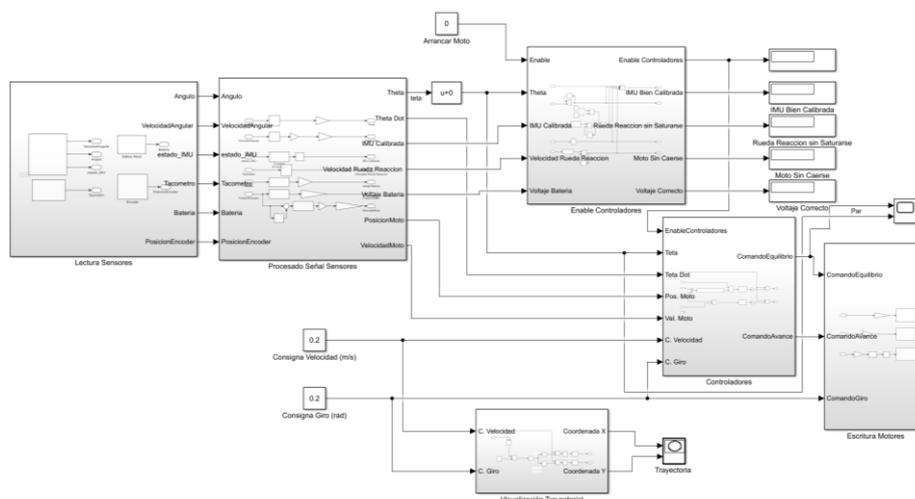


Figura 14. Diagrama Simulink del control en tiempo real de la motocicleta

## 6 Conclusiones

Se ha obtenido el modelo matemático y el modelado gráfico dinámico de una mini-motocicleta autónoma, incluyendo las partes mecánicas, sensores, actuadores y el contacto de las ruedas con el suelo. Esta es una metodología que permite diseñar controladores para sistemas mecatrónicos y visualizar los resultados antes de su implementación física. Este modelo virtual está preparado para poder usarse en el laboratorio de prácticas de mecatrónica, si bien requiere ejecutarse en un computador con cierta potencia de cálculo. Consideramos por otro lado, que aunque es una iniciativa muy interesante, esta primera versión de la motocicleta incluida en el *Arduino Engineering kit* no es idónea para usarse por parte de los alumnos, y requiere ser rediseñada, cambiando el motor de la rueda de reacción y la IMU. Sugerimos también incorporar una estructura metálica para aumentar la rigidez del sistema.

## Referencias

- [1] Adam Owczarkowski, P. K. (2015). Mathematical Modeling of the Bicycle Robot with the Reaction Wheel. *Journal of Automation, Mobile Robotics & Intelligent Systems*, 9(3), 3-8.
- [2] Aström, K. J., Klein, R. E., & Lennartsson, A. (2005). Bicycle Dynamics and Control. *Control Systems Magazine*, 25(4), 26-47.
- [3] Mavros, G. (2010). Enhanced motorcycle roll stability by use of reaction wheel actuator. . *Proceedings of the 10th International Symposium on Advanced Vehicle Control*, 532-537.
- [4] Lychek Keo, K. Y. (2011). Experimental Results for Stabilizing of a Bicycle with a Flywheel Balancer. *IEEE International Conference on Robotics and Automation*, 6150-6155.
- [5] Lei Guo, Q. L. (2009). Design of Linear Quadratic Optimal Controller for Bicycle Robot. *Proceedings of the IEEE International Conference on Automation and Logistics*, 1968-1972.
- [6] Kanjanawanishkul, K. (2015). LQR and MPC controller design and comparison for a stationary self-balancing bicycle robot with a reaction wheel. *Kybernetika*, 51(1), 173-191.
- [7] Bravo M., D. A., Rengifo R., C. F., & Díaz O., J. F. Control of a Robotic Bicycle. (2018).
- [8] C. Tapken, P. Glosekoetter, A.Olivares, G.Olivares “Modelling and Controlling the Kinetic and Dynamic of a Bicycle”. *Enseñanza y Aprendizaje de Ingeniería de Computadores* (2018). ISSN 2173-8688.
- [9] Saavedra Trujillo, J. C. (5 de Junio de 2019). Video de simulación PID. Obtenido de <https://www.youtube.com/watch?v=ecCUvxZ6i-I>
- [10] Saavedra Trujillo, J. C. (6 de Junio de 2019). Video de simulación LQR. Obtenido de <https://www.youtube.com/watch?v=WaIshBh2qtc>

## **Sistema Empotrado Distribuido para el Control de Accesos - RFIDoors**

Daniel García Moreno, Iván Aguilera Calle, Alberto A. Del Barrio, Guillermo Botella

Departamento de Arquitectura de Computadores y Automática, Facultad de Informática  
Universidad Complutense de Madrid  
Madrid, España  
{daniel10, ivanag01, abarriog, gbotella}@ucm.es

**Resumen.** Con el paso del tiempo se ha ido ampliando la utilización de sistemas con identificación por radiofrecuencia (RFID) en los distintos ámbitos de la sociedad actual. En este trabajo se presenta la implementación de un sistema empotrado distribuido compuesto por elementos de fácil adquisición y de bajo coste como la Raspberry Pi, los módulos RFID o los sensores de ultrasonidos, cuyo objetivo es controlar y gestionar un sistema de autenticación para la apertura y cierre de puertas. Como complemento, este sistema consta además de un servidor y una aplicación para la parte administrativa y operativa del sistema.

**Palabras Clave:** RFID, Sistema Empotrado Distribuido, Bajo Coste, UART, Accesos, Raspberry Pi.

**Abstract.** Nowadays, the use of the systems with radio frequency identification (RFID) is becoming widespread in different scenarios of society. This paper presents the implementation of a Distributed Embedded System composed of low-cost components such as Raspberry Pi, RFID modules, ultrasound sensors and others, whose objective is to manage an authentication system for the opening and closing of doors. Furthermore, this system incorporates a server and an application for the administrative and operative part of the system.

**Keywords:** RFID, Distributed Embedded System, Low Cost, UART, Access, Raspberry Pi.

## 1 Introducción

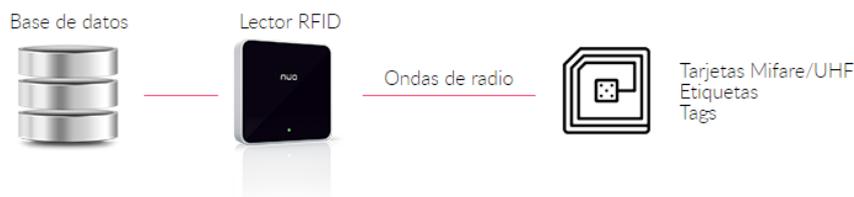
La asignatura Sistemas Empotrados Distribuidos (SED) es una materia que se ubica en el Máster de Ingeniería Informática de la Universidad Complutense de Madrid (UCM) [1], implantado desde el curso 2013/2014. Con el fin de potenciar las habilidades en el ámbito laboral, la asignatura de SED basa gran parte de su nota en la realización de múltiples prácticas y proyectos que permitan a los alumnos mejorar sus capacidades en el área de la *Computación Ubicua* y el *Internet of Things* (IoT) [2-7].

En este artículo se describe la implementación de un sistema empotrado distribuido cuyo objetivo es controlar el acceso a una sala a partir de una puerta motorizada. El control de accesos se realiza a partir de llaves electrónicas, empleando para ello la tecnología Radio Frequency Identification (RFID). El sistema desarrollado también permite llevar un control de las personas que han accedido por la puerta.

El resto del artículo se organiza de la siguiente manera: la Sección 2 presenta un estudio de diversos sistemas de control de acceso y de la tecnología RFID; en la Sección 3 se presenta la arquitectura del proyecto y en la 4 la maqueta realizada para implementar la propuesta; la Sección 5 ofrece nuestras conclusiones y posibles líneas de trabajo futuro.

## 2 Estado del Arte

La tecnología RFID permite identificar o reconocer una etiqueta, aka *tag*, la cual se puede encontrar ubicada en cualquier sitio, como por ejemplo: en la ropa, en un vehículo, en productos de una fábrica o almacén o incluso en las propias personas o animales. La tecnología RFID [17-20] se ha ido expandiendo en los últimos años, debido principalmente a su bajo coste de económico y a la flexibilidad e información y características que pueden aportar frente a los sistemas de control de acceso convencionales como son las tarjetas con bandas magnéticas, códigos de barras, o el reconocimiento basado en vídeo [8-11], redes neuronales [12-13] u otros filtros con mayor complejidad [14-16].



**Figura 1.** Sistema RFID [20].

Un sistema genérico que utilice la tecnología RFID de forma básica debe tener al menos la infraestructura mostrada en la Figura 1. Las etiquetas RFID tienen integrado un chip, el cual es capaz de responder o de emitir una señal como respuesta a las peticiones que realiza la antena o lector RFID. Por último, el lector RFID suele

consultar y/o realizar modificaciones sobre una Base de Datos del sistema con el fin de dotar de información y/o lógica a la funcionalidad del sistema.

Las etiquetas RFID se encargan de almacenar la información de identificación mediante un código único que se implanta en el chip durante la fabricación [17-19]. Los tags tienen un chip y una antena impresa, los cuales sirven para enviar la información al lector, como se puede apreciar en la Figura 2.

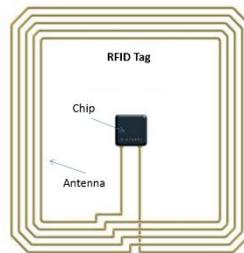


Figura 2. Esquema Tag RFID [17].

La antena del tag recoge la energía transmitida por el lector RFID y la canaliza al chip, por lo que cuanto mayor es el área de la antena de la etiqueta, mayor es la energía que transmite al chip, y también mayor es la distancia o alcance. En la Figura 3, se puede observar el cálculo de la inductancia de la antena del tag y su relación con las dimensiones de la etiqueta. La inductancia es la magnitud física que indica la capacidad para generar un flujo de inducción magnética, cuya unidad de medida es el Henry,  $L$ .

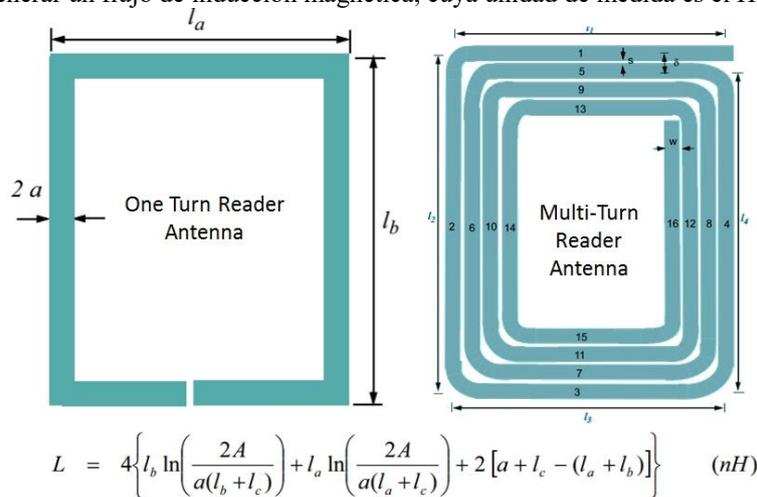


Figura 3. Calcula de Inductancia tag RFID [17].

## 2.2 Casos de uso de la tecnología RFID

La forma y los tamaños de los tags varían según cual sea el entorno en el que se estén utilizando, por lo que existe una diversidad abundante de formas y materiales de tags RFID, como puede observarse en la Figura 4.



**Figura 4.** Tipos de tags RFID [21].

Algunos de los sistemas de identificación mediante tecnología RFID más utilizados son los siguientes:

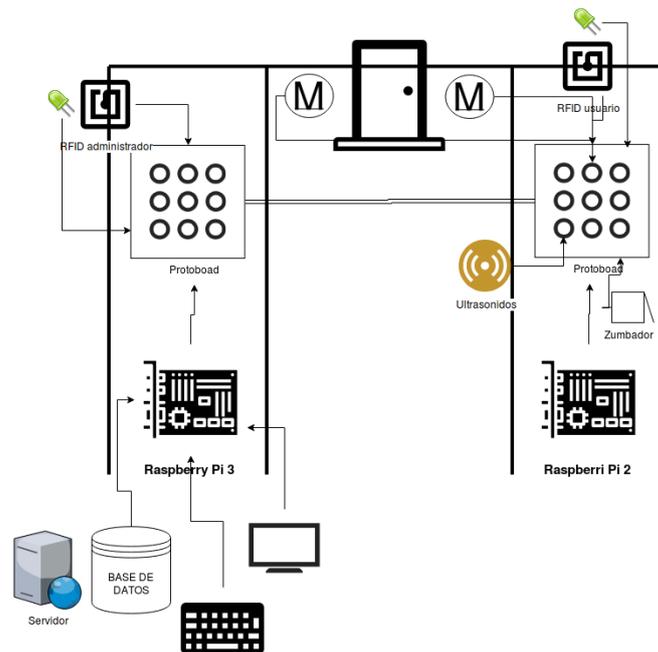
- Identificación de animales con tecnología RFID: se fabrican con materiales que no sean tóxicos y se comprime la etiqueta en una cápsula que no sea nociva para el organismo, ya que normalmente suelen insertarse en la parte inferior de la piel o incluso en el estómago. Los *Ear Tag* son muy utilizados en este contexto, por ejemplo.
- Sistemas de control de accesos de empleados. Los *Pocket Tag* y los *Key Tag* son muy utilizados en este entorno, por ejemplo.
- Sistemas de control de accesos a las habitaciones de hoteles. En este caso frecuentemente se encuentran tarjetas de plástico con RFID, que son un caso particular de *Pocket Tag*.

### 3 Arquitectura del Sistema

Como puede observarse en la Figura 5, los componentes principales del proyecto RFIDoors son dos Raspberry Pi conectadas vía UART. El papel que juega cada una de las placas está muy diferenciado:

- En primer lugar tenemos una Raspberry Pi 2 modelo B, que se encarga del control de los diferentes componentes del sistema que provocan el control de las puertas (motores, sensores de ultrasonido, lector de tarjetas, leds...). En otras palabras, esta placa interactuará de manera directa con el usuario.
- En segundo lugar, disponemos de una Raspberry Pi 3 modelo B, cuya función es contener la base de datos que contiene a los usuarios y al servidor. Esta placa estará constantemente a la escucha de las peticiones que vía UART le solicite la otra placa. Esta placa interactuará de manera más directa con el administrador del sistema, disponiendo de otro lector de tarjetas de uso restringido únicamente al administrador, cuyo objetivo es el poder dar de alta nuevas tarjetas. Además, funcionará como servidor para poder ejecutar la aplicación web con la que el administrador podrá dar de alta, de baja y consultar usuarios y accesos.

**DIAGRAMA DEL SISTEMA RFIDoors**



**Figura 5.** Arquitectura del sistema.

El funcionamiento básico del sistema es el siguiente:

- Las llaves electrónicas llevarán almacenadas un ID, el cual es único para cada tarjeta RFID. Para leer las tarjetas, hemos utilizado dos chips RFID RC522,

puesto que no necesitamos detectar tarjetas a más de dos o tres centímetros. Uno de los lectores será para el usuario y el otro para el administrador.

- Al acercar una tarjeta RFID al lector, este ID se envía de la Raspberry que está conectada con el lector RFID y a la Raspberry administrativa, con el objetivo de consultar su existencia en la base de datos. Si este identificador existe, la placa que ha recibido la solicitud de consulta respondería a la placa que ha obtenido el ID, para posteriormente proceder a la apertura de las puertas.

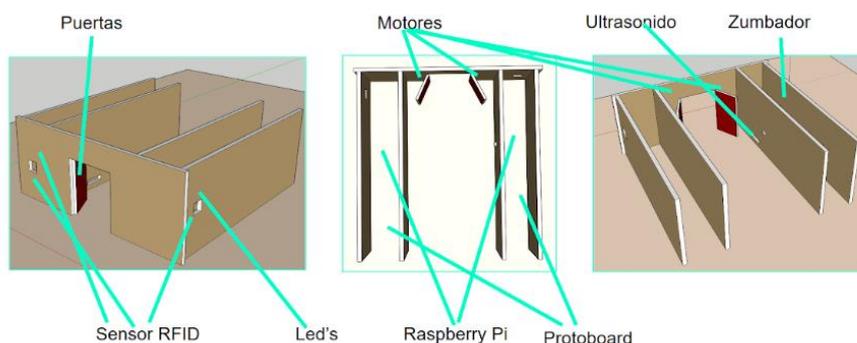
Una vez comprobada la validez de las tarjetas en la base de datos, el objetivo es que las puertas se abran y cierren de manera automática. Para tal fin, se ha utilizado dos servomotores (se trata de una puerta doble) SG90. Un servomotor rota los grados que se le indiquen mediante modulación del ancho de pulso (PWM). En el caso de estos servomotores, tienen un campo de acción de 180°, pero dado la estructura de la maqueta desarrollada para este proyecto y la ligereza de los materiales utilizados, solo se requiere giros de 90°, y este modelo de servo nos lo permite hacer con la suficiente fuerza como para poder abrir y cerrar las puertas. Los motores están conectados a la Raspberry Pi 2, encargada del control de todos los componentes.

Dado que los motores se encargan de abrir las puertas, es deseable que se cierren automáticamente, ya sea porque la persona que quería entrar ya ha pasado, o porque ha transcurrido un periodo de seguridad en el que nadie ha accedido (en nuestro sistema esto está configurado a 15 segundos). Para poder implementar esta funcionalidad, se ha utilizado un sensor de ultrasonidos, en concreto el modelo HC-SR04. El lector de ultrasonidos está formado por dos cilindros metálicos. Uno de ellos emite una señal que al rebotar en una pared es captada por el otro cilindro. Partiendo de este funcionamiento, lo que se ha hecho es capturar el instante de tiempo en el que la señal es lanzada por el sensor y el instante de tiempo en la misma ha sido recogida. Una vez obtenidos estos tiempos a partir de la fórmula matemática que indica que la distancia recorrida es igual al tiempo transcurrido por la velocidad entre dos ( $D = T \cdot V / 2$ ), calculamos la distancia sobre la que está el sensor de una pared. Si una puerta está colocada sobre un pasillo, el uso de este sensor nos ofrece la funcionalidad de permitir detectar el paso personas por delante de él. En cuanto se detecte una distancia inferior a la longitud del pasillo, la Raspberry Pi 2 enviaría a los motores una orden para cerrar las puertas. El rango de funcionamiento del sensor abarca desde los 2 cm hasta los 3 metros, por lo que para una maqueta pequeña el sensor es totalmente válido.

Para mejorar la usabilidad del sistema de cara al usuario, se han utilizado leds de colores colocados sobre el lector de tarjeta con el objetivo de que el usuario conozca si la tarjeta acercada es válida para poder acceder (led verde), o por el contrario, no es válida para acceder (led rojo). Acompañando a los leds, se ha utilizado un zumbador para poder emitir secuencias de pitidos en caso de acercar una tarjeta no válida o un pitido largo para avisar de que la tarjeta es válida y se va a proceder a la apertura de las puertas.

## 4 Prueba de Concepto

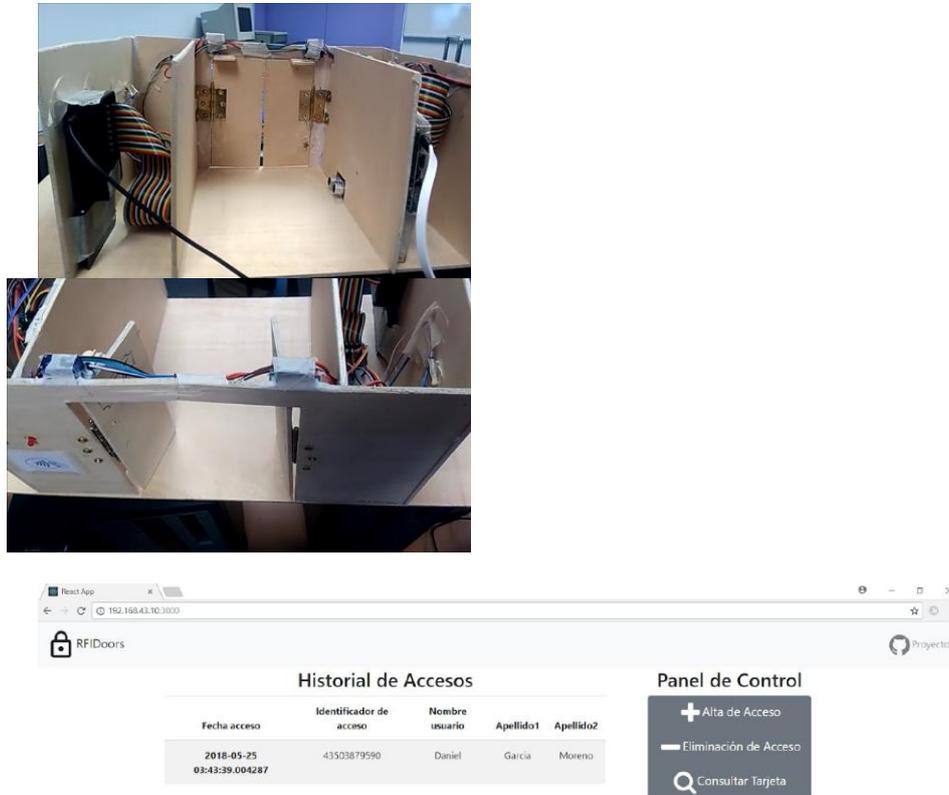
Para comprobar el correcto funcionamiento del proyecto, se construyó una maqueta a partir de tabloncillos de contrachapado que simulan una puerta, y tras ella, un pasillo. De forma paralela, se han construyeron dos pasillos ocultos que contienen toda la maquinaria del proyecto. En la Figura 6 podemos ver cómo sería la maqueta, así como la disposición final de los elementos anteriormente mostrados en la Figura 5. Como se puede observar, el pasillo central se encuentra tras atravesar la puerta. Los dos pasillos ocultos pueden apreciarse en la vista cenital mostrada en la Figura 6.



**Figura 6.** Boceto maqueta RFIDoors.

En el pasillo de la izquierda se encuentra la Raspberry Pi 3, es decir, la placa que contiene la base de datos, que actúa como servidor para mover la aplicación web de gestión del administrador, y que escuchará las peticiones que vía UART reciba de la otra placa para realizar consultas en la base de datos. Además, hay que recordar que gestiona un lector RFID para dar de alta nuevas tarjetas o consultar la información de otras, acompañado de los leds correspondientes que avisan del funcionamiento del lector.

En el pasillo de la derecha, se encuentra localizada la Raspberry Pi 2, la cual se encarga del control de los componentes que interactuarán con el usuario. Entre el pasillo de la derecha y el pasillo central se encuentra un agujero en la pared tras la puerta, donde se encuentra localizado el sensor de ultrasonido, de tal modo que nada más atravesar la puerta, el sensor capta el acceso para posteriormente cerrar la puerta. Los servomotores se encuentran en el pasillo central encima de las puertas, y se conectan a la Raspberry Pi 2 a través de un agujero en la pared del pasillo derecho. La UART conecta las dos placas, y su cableado transcurre por encima de las puertas. Las protoboard se encuentran una en cada pasillo lateral, con el objetivo de poder tener los cables de conexión de los componentes más ordenados, y realizar una conexión con las Raspberry más limpia. Finalmente, en la Figura 7 pueden verse un par de vistas de la maqueta con los dispositivos embebidos, así como una de las funcionalidades de la aplicación web del administrador.



**Figura 7.** Maqueta RFIDoors y aplicación web del administrador.

## 5 Conclusiones

En este artículo se ha presentado RFIDoors, un sistema empotrado distribuido para gestionar el acceso a una habitación por medio de tarjetas basadas en la tecnología RFID. Por medio de componentes de bajo coste y alta accesibilidad, como las placas Raspberry Pi, sensores de ultrasonido, servomotores, etc. se ha construido una maqueta como prueba de concepto y con resultado altamente satisfactorio. Además de gestionar el acceso, se ha conseguido el cierre automático de las puertas y se ha desarrollado una aplicación que permite a un administrador gestionar el acceso.

Como líneas de trabajo futuras, el futuro el prototipo planteado podría escalarse con más nodos e incluso pensar en la utilización de placas de más bajo coste y rendimiento para la gestión de los sensores, como puede ser la Raspberry Pi Zero o incluso NodeMCU.

## Referencias

1. Máster en Ingeniería Informática de la Universidad Complutense de Madrid, <http://informatica.ucm.es/estudios/2018-19/master-ingenieriainformatica>
2. D. Lora et al., "Sistema de Seguridad Basado en una Plataforma Heterogénea Distribuida", *Enseñanza y Aprendizaje de Ingeniería de Computadores*, 5: 29-38 (2015).
3. F. Parrales et al. "Una Orquesta Sinfónica como Ejemplo de Aplicación de un Sistema Empotrado Distribuido", *Enseñanza y Aprendizaje de Ingeniería de Computadores*, 5: 115-124 (2015).
4. I.M. Laclaustra et al. "Sistema Domótico Distribuido para Controlar el Riego y el Aire Acondicionado en el Hogar", *Enseñanza y Aprendizaje de Ingeniería de Computadores*, 6: 87-102 (2016).
5. H. Ivanov et al. "Bomberman modo multijugador", *Enseñanza y Aprendizaje de Ingeniería de Computadores*, 7: 53-68 (2017).
6. J. Martín et al., 2016. "A distributed HW-SW platform for fireworks", *Proceedings of the Summer Computer Simulation Conference*. Montreal (Canada), article 17, 7 pages.
7. M. Hernández et al. "Clúster de Computación Científica de Bajo Coste y Consumo", *Enseñanza y Aprendizaje de Ingeniería de Computadores*, 8: 85-95 (2018).
8. D.G. Fernández et al., "HEVC optimization based on human perception for real-time environments", *Multimed Tools Appl* (2018). <https://doi.org/10.1007/s11042-018-7033-y>
9. D. G. Fernández et al., "Complexity reduction in the HEVC/H265 standard based on smooth region classification. *Digital Signal Processing*, 73: 24-39 (2018).
10. D. Rodríguez et al. 2018. "Data hiding algorithm for HEVC using intra-coded frames", *Proceedings of the 50th Computer Simulation Conference*. Society for Computer Simulation International, Bordeaux (France), Article 2, 12 pages.
11. D. Rodríguez et al. "Intra-Steganography: Hiding Data in High-Resolution Videos," *2018 IEEE/ACM 22nd International Symposium on Distributed Simulation and Real Time Applications (DS-RT)*, Madrid, 2018, pp. 1-8.
12. M. S. Kim et al., "Efficient Mitchell's Approximate Log Multipliers for Convolutional Neural Networks", *IEEE Trans. Computers* 68(5): 660-675 (2019).
13. Min Soo Kim et al., "Low-power implementation of Mitchell's approximate logarithmic multiplication for convolutional neural networks", *ASP-DAC 2018*: 617-622.
14. A. A. Del Barrio et al., "A Distributed Clustered Architecture to Tackle Delay Variations in Datapath Synthesis," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 35, no. 3, pp. 419-432, 2016.
15. A. A. Del Barrio et al., "A Partial Carry-Save On-the-Fly Correction Multispeculative Multiplier," *IEEE Transactions on Computers*, vol. 65, no. 11, pp. 3251-3264, 2016.
16. A. A. Del Barrio et al., "A Combined Arithmetic-High-Level Synthesis Solution to Deploy Partial Carry-Save Radix-8 Booth Multipliers in Datapaths," in *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 66, no. 2, pp. 742-755, Feb. 2019.
17. A. Kalnoskas, "How do RFID tags and reader antennas work?," *Analog IC Tips*, 25-Apr-2017. [Online]. Available: <https://www.analogictips.com/rfid-tag-and-reader-antennas/>.
18. D. P. Rose et al., "Adhesive RFID Sensor Patch for Monitoring of Sweat Electrolytes," in *IEEE Transactions on Biomedical Engineering*, vol. 62, no. 6, pp. 1457-1465, June 2015.
19. D. He et al., "An Analysis of RFID Authentication Schemes for Internet of Things in Healthcare Environment Using Elliptic Curve Cryptography," in *IEEE Internet of Things Journal*, vol. 2, no. 1, pp. 72-83, Feb. 2015.
20. A. Donaire and R. Tan, "Sistema de identificación por radiofrecuencia para competiciones deportivas", 2018. [Online]. Available: <https://eprints.ucm.es/54982/>
21. J.D. Irawan et al., "RFID and IOT for Attendance Monitoring System", *MATEC Web of Conferences* 164, 01020 (2018).



## Uso de la realidad aumentada para la mejora de la visión espacial del alumnado.

Antonio José Moreno Garrido y Oresti Baños Legrán

Departamento de Arquitectura y Tecnología de Computadores, Universidad de Granada.

antonyamg@correo.ugr.es & antoniojose.amg@gmail.com  
oresti@ugr.es

**Abstract.** Spatial visualization is usually not trained in class explicitly. Thus, in turn, some students can show deficiencies and development problems for this skill. This work proposes the use of augmented reality to compensate such deficiencies. The work analyzes various augmented reality online platforms and uses one of them in a real classroom. A preliminary study of the spatial vision level of the students before and after the use of the proposed system is shown, also accounting for the perceptions (usability, possible obstacles to its use, motivation, etc.) of teachers and students on the use of augmented reality.

**Keywords:** Augmented Reality; Education; Spatial Visualization; Educational Innovation; Educational Technology.

**Resumen.** La realidad aumentada es una tecnología que está cada vez más presente en la sociedad y la cual se emplea paulatinamente con mayor frecuencia para diversos fines. En este trabajo se tratará de acotar su aplicación al ámbito educativo y concretamente al de la ayuda a la mejora de la visión espacial. Esto es debido a que dicha competencia, en general, es una habilidad que se le demanda al alumno, pero la cual no se trabaja de forma específica. Esta situación da lugar a dificultades en cuanto al dominio de esta habilidad y a diferencias de nivel entre alumnos de un mismo curso y grupo. Con la finalidad de intentar dar solución a estos aspectos, se llevará a cabo una breve investigación en la que se intentará responder si el uso de la realidad aumentada podría ser una herramienta a tener en cuenta a la hora de conformar un aporte positivo en cuanto a la mejora de la visión espacial del alumno. Para ello, se realizará un análisis de las plataformas de realidad aumentada disponibles actualmente en el mercado, se llevará a cabo una implementación didáctica en el aula y, finalmente, se recabarán y analizarán todos los datos recogidos en cuanto a nivel de visión espacial del alumnado. Además se recopilarán y tratarán datos sobre percepciones, tanto del propio alumnado, como del profesorado.

**Palabras clave:** Realidad Aumentada; Educación; Visión Espacial; Innovación Educativa; Tecnología Educativa.

## **1 Introducción**

### **1.1 Dificultad detectada y solución propuesta**

La competencia de visión espacial es una habilidad que es demandada al alumno y que, por tanto, necesita dominar con cierto nivel y soltura para poder dar respuesta a las diferentes cuestiones y retos que se le plantean a lo largo del periodo educativo. El principal problema que se plantea es que, aunque se demanda al alumno esta capacidad, en muy pocas ocasiones se trabaja de forma específica con la finalidad de desarrollarla progresivamente. De esta manera, se hace previsible que pueda generarse una cierta dificultad de visión espacial en el alumnado, pudiéndose dar también una diferencia de nivel entre alumnos de un mismo curso y grupo. Dependiendo de la causa que lo motive, estas dificultades mencionadas pueden traer consigo una posible frustración, pérdida de interés, segregación y discriminación dentro del aula, etc.

Puesto que la tecnología está cada vez más presente en la sociedad actual, se ha decidido emplear y aprovechar la realidad aumentada (de ahora en adelante abreviada, según sus siglas en inglés, como AR) como herramienta de ayuda a la mejora de la visión espacial. Debido a que esta capacidad trae consigo un carácter ciertamente gráfico, el uso de la AR es una tecnología que se considera muy idónea para su uso en el aula y que se presupone que puede tener una gran vinculación con el carácter espacio-visual que se está queriendo alcanzar. Además, se conjetura que puede darse una gran aceptación por parte del alumnado y del profesorado, debido al dinamismo y carácter novedoso que engloba a esta herramienta tecnológica. Como finalidad última, lo que se pretende conseguir es que el alumno disponga de una herramienta capaz de cooperar con la obtención de una visión y concepción general de elementos en el espacio, que le permita identificar y comprender sus propios errores, que le proporcione una amplia fuente de información complementaria, un ahorro de material a emplear, entre otros muchos aspectos.

En cuanto a la secuencia a seguir en el trabajo, se plantea, en primer lugar, hacer un análisis y selección de la plataforma de AR más idónea para su uso en el aula. Posteriormente y tras haber elaborado material gráfico propio, se procederá a la implementación didáctica. En esta puesta en práctica se dará una breve introducción al alumnado sobre qué es la AR, se medirá cual es el nivel de visión espacial del mismo, se procederá a la utilización de AR y, por último, se recogerá la percepción del alumnado y del profesorado acerca del uso de la AR en el aula.

### **1.2 Estado del arte**

Derivado de la consulta de diversas publicaciones sobre estudios llevados a cabo en base al empleo de la realidad aumentada en el aula, se definen diversas ideas a destacar. En primer lugar, según estudios realizados [4], la simple utilización de la AR aporta al usuario conocimientos sobre el manejo de dispositivos y conocimiento sobre qué es y las diferentes finalidades que puede tener esta tecnología. También es de destacar que el uso de la AR en el aula no está acotado específicamente a una etapa educativa concreta, sino que su empleo es factible desde niveles básicos como preescolar, hasta niveles más elevados como el universitario [1, 3]. Relacionado con este

empleo en educación enfocado al alumno, hay que resaltar la motivación que genera el uso de la AR en el alumnado, el complemento y enriquecimiento que puede ofrecer a apuntes y/o libros de texto utilizados, además del desarrollo del pensamiento científico que genera en el alumno [6]. Todos estos aspectos influirán, en mayor o menor medida, en el aumento del rendimiento del alumnado [2].

Por otro lado, a la hora de la puesta en práctica de la AR en el aula, son destacables diversos factores. El primero de ellos es la necesidad de formación previa. Ésta es de vital relevancia para el docente, el cual debe conocer la AR de manera destacable, estar al tanto de sus pros, sus contras, del amplio abanico de aplicaciones que puede tener en educación y de los problemas que pueden darse a la hora de su utilización [1, 7, 9]. Además, deberá tener el conocimiento suficiente para aportar al alumnado una introducción a esta tecnología y brindar unas breves nociones para su utilización en clase.

Como segundo factor, nos encontramos con los recursos. Para el uso de la AR es necesario un terminal móvil (ya sea smartphone o tablet), además de un punto de acceso a internet. Estos requisitos suponen una amplia inversión inicial y un posterior gasto en mantenimiento, lo cual genera una posible barrera en el uso de la AR [3, 8].

Por último, como tercer factor a tener en cuenta, nos encontramos con la limitación de uso. Este agente limitante es referido al empleo indiscriminado de la AR. Está demostrado que este tipo de utilización no es beneficioso para el alumno, sino que comienza a ser negativo cuando su empleo no es justificado. Es decir, el uso de esta herramienta tecnológica es el medio para intentar transmitir ciertos conocimientos y no el fin. No se persigue la utilización de la AR porque sí, sino que se persigue transmitir ciertos conocimientos con una herramienta nueva que permita sacar mayor partido a las explicaciones que brinda el docente y que facilite y amplifique la retención de esos contenidos por parte del alumnado. Por tanto, el uso infundado de esta tecnología trae consigo que el alumno pierda el interés por su utilización y que descuide su atención en clase [1, 2, 5].

## 2 Selección de plataforma de realidad aumentada

Con la finalidad de seleccionar la plataforma de AR más idónea y que brindara mayores facilidades de uso a la hora de la implementación didáctica en el aula, se llevó a cabo un muestreo. Se partió de 10 plataformas de AR existentes en el mercado. Se tuvieron en cuenta ciertos aspectos y, una vez definidos, se llevó a cabo un proceso de selección por descarte. Dichos ítems utilizados para la elección de la plataforma de AR son los siguientes: uso transversal, buscando que la plataforma no esté acotada únicamente a una materia específica. Creación de proyectos propios, pudiendo llevar a cabo un uso más personalizado, existiendo total libertad para creación de contenidos y no uno definido y limitado por la propia plataforma. Facilidad de uso, buscando que su utilización sea extensiva y la experiencia se centre en los contenidos que se desea transmitir y no en hacer funcionar la app de AR. Gratuidad, siendo un aspecto muy a destacar a la hora de buscar que la AR se use de forma masiva y que todo el mundo pueda tener acceso a la misma, dejando a un lado, en la medida de lo

posible, el factor económico. Existencia de app de visualización, la cual esté disponible para todo tipo de terminales móviles (smartphone y tablet) y sea compatible con los sistemas operativos utilizados en la actualidad (Android e iOS). Una vez definidos todos los ítems tenidos en cuenta en el proceso de selección, se determina como plataforma de AR elegida la de la casa comercial Aumentaty, cuya app móvil se denomina Scope. A continuación se disponen dos tablas (tabla 1 y 2) en las que queda recogido el análisis, descarte y selección final de la plataforma utilizada. Consultar [10] para conocer más detalles acerca de la selección realizada.

**Tabla 1.**Muestreo inicial de plataformas de realidad aumentada.

Muestreo de plataformas de AR								
Plataforma	Uso transv.	Proyectos propios	Fácil uso	Gratuidad	App visualiz.	Smartphone y tablet	Android e iOS	Selección de Apps
Anatomy 4D	x	x	-	-	-	-	-	NO
Augment	✓	✓	✓	✓*	✓	✓	✓	SÍ
Aumentaty	✓	✓	✓	✓**	✓	✓	✓	SÍ
Chromville	x	x	-	-	-	-	-	NO
Elements 4D	x	x	-	-	-	-	-	NO
HP Reveal	✓	✓	✓	✓	✓	✓	✓	SÍ
Layar	✓	✓	✓	✓	✓	✓	✓	SÍ
Quiver	x	x	-	-	-	-	-	NO
Vuforia	✓	✓	x	✓*	✓	✓	✓	SÍ
Zookazam	x	x	-	-	-	-	-	NO

\*Gratis durante un determinado periodo de prueba.

\*\* Versión gratuita aunque con algunas limitaciones con respecto a la versión de pago.

**Tabla 2.**Selección de plataforma de realidad aumentada para implementación didáctica en el aula.

Selección de plataforma de AR								
Plataforma	Sistema operativo				App gratuita	Gratuidad		Uso de App en interv. en el aula
	Android		iOS			Periodo Prueba	Total	
	Móvil	Tablet	Móvil	Tablet				
Augment	✓	✓	✓	✓	✓	✓	x	NO
<b>Aumentaty</b>	✓	✓	✓	✓	✓	✓	✓*	<b>SÍ</b>
HP Reveal	✓	✓	✓	✓	✓	✓	✓	NO
Layar	✓	✓	✓	✓	✓	✓	x	NO
Vuforia	✓	✓	✓	✓	✓	✓	x	NO

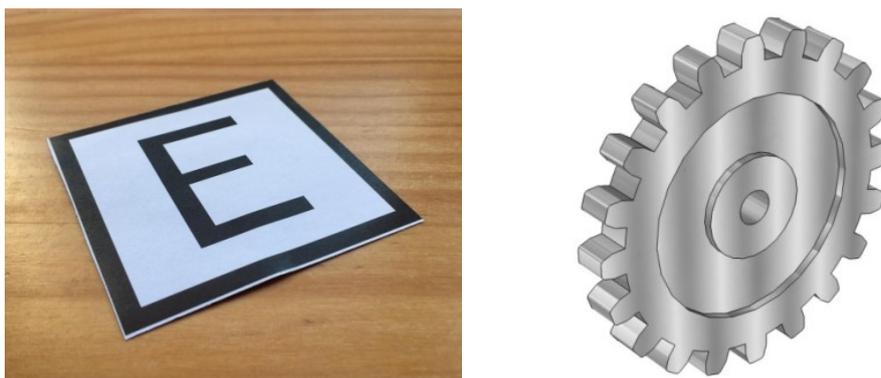
\*Con ciertas limitaciones respecto a la versión de pago.

### 3 Funcionamiento de plataforma de realidad aumentada

#### 3.1 Preparación de material necesario

Previo a la utilización de la app de AR, es necesario atender a la elaboración de cierto material para el correcto funcionamiento de esta herramienta. De forma concreta, es necesaria la elaboración de una imagen rastreable por la app móvil (denominada

marcador) y del propio contenido con el que se quiere trabajar (en este caso y relacionado con el desarrollo de la visión espacial, se utilizarán modelos digitales en 3 dimensiones). Una vez creado el marcador y el modelo 3D (ver Fig. 1), se importará dicho material a la plataforma de AR y se vinculará entre sí. Por tanto, cada marcador corresponderá a un modelo 3D concreto. Cada vinculación marcador-modelo 3D se denomina proyecto de AR.



**Fig.1.** (A la izquierda) imagen denominada marcador, la cual es rastreable y reconocida por la app de AR. (A la derecha) modelo digital en 3D de un engranaje metálico. Éste, entre otros, fue empleado en la implementación práctica en el aula.

### 3.2 Utilización de app de realidad aumentada

Para utilizar la app de visualización de AR, simplemente hay que realizar una búsqueda del proyecto de AR creado. Una vez encontrado, se procede a su descarga en el terminal móvil y a continuación ya se puede comenzar con el uso de esta herramienta (ver Fig. 2).



**Fig.2.** Ejemplo de funcionamiento de app de AR (Scope) utilizada en la sesión práctica llevada a cabo en el aula.

## 4 Puesta en práctica en el aula

Para la implementación práctica de la AR en el aula se tomó una muestra de 59 alumnos de 3º de ESO del IES Ilíberis (localidad de Atarfe, provincia de Granada, España). Además se contó con la participación de 3 docentes del mismo centro educativo. Una vez definidos los diferentes agentes intervinientes en el proceso, se procede al seguimiento de una cronología fijada previamente. Ésta consiste en una introducción de la AR al alumnado, una medición de su nivel de visión espacial, utilización de AR y, por último, una consulta al alumnado y al profesorado para conocer su percepción acerca del uso de esta herramienta tecnológica. Cada uno de estos 4 pasos seguidos se definen a continuación con mayor detalle. Como información complementaria, hay que mencionar que esta puesta en práctica en el aula se llevó a cabo durante el periodo lectivo de la asignatura de Tecnología.

### 4.1 Introducción a la realidad aumentada

En primer lugar, se procedió a explicar al alumnado en qué consistía el termino realidad aumentada, cuáles son los ámbitos en los que actualmente se utiliza esta tecnología (marketing, industria, automoción, turismo, educación...), además de mostrar algunos casos de la vida cotidiana en los que la AR está presente y no nos percatamos de ello. Un claro ejemplo son los filtros utilizados por las apps de redes sociales o la información vial que algunos vehículos proyectan en su parabrisas.

### 4.2 Medición de nivel de visión espacial

Previo a cualquier toma de contacto con AR, se procedió a la realización de un test de visión espacial (modelo contenido en [10]). En éste se encuentran cuestiones en las que el alumno debe relacionar cuerpos geométricos con sus vistas (alzado, planta y perfil), vincular volúmenes con su desplegado e identificar figuras tras haber variado la posición inicial de la misma. Todo ello con la finalidad de fijar un nivel del alumnado e intentar confirmar que existen algunos alumnos con ciertas dificultades en cuanto a visión espacial, además de identificar si está presente una diferencia de nivel destacable entre alumnos. Para la posterior recopilación y análisis de los resultados obtenidos, se fijó una valoración mínima de 0 puntos y una máxima de 10.

### 4.3 Experiencia con realidad aumentada

En primer lugar, se procedió a realizar una breve explicación sobre el funcionamiento de la app utilizada (en este caso, Scope). A continuación se procedió a la búsqueda y la descarga de diferentes proyectos de AR creados para esta sesión. Éstos contenían respuestas al test de visión espacial realizado previamente, además de modelos 3D de diferente índole. Para comenzar, la elaboración de modelos 3D correspondientes al test de visión espacial permitía al alumno conocer sus errores cometidos, además de tener la oportunidad, por sí mismo, de poder identificar y comprender en qué se había equivocado. Por otra parte, se dispuso de otros modelos 3D que podrían ser utilizados como complemento al libro de texto que utilizan normalmente, co-

mo complemento a la explicación del docente, como ejemplo de herramienta para que el alumnado pueda crear sus propios modelos 3D y proyectos de AR, etc. En definitiva, tener una primera toma de contacto con esta tecnología, para poder mostrar al alumnado y al profesorado las diferentes aplicaciones y beneficios que podría tener en el ámbito educativo.

#### 4.4 Obtención de percepciones sobre el uso de la realidad aumentada

Debido a la limitación de tiempo existente para la elaboración de la presente investigación, la cual deriva de un Trabajo Fin de Máster [10], se ha tenido únicamente la oportunidad de tener una primera toma de contacto con la AR. Más allá de recoger datos de carácter cuantitativo, para los que sería necesario un estudio más prolongado en el tiempo, se llevó a cabo la obtención de datos de carácter cualitativo. En definitiva, se procedió a conocer cuál era la percepción del alumnado y profesorado acerca del uso de la AR tras haberla usado en la sesión práctica. Para ello, se realizaron preguntas tipo test (tests disponibles en [10]), estando presentes cuestiones como la posibilidad de acceso a internet y a terminales móviles, motivación a la hora de usar AR, ya sea fuera y/o dentro del aula, aporte positivo del uso de esta tecnología, dificultad a la hora de su uso, necesidad de formación previa en el caso de los docentes, etc.

## 5 Resultados

### 5.1 Nivel de visión espacial del alumnado

Tras la realización del test de nivel de visión espacial llevado a cabo por el alumnado, se obtuvieron los siguientes resultados. Se parte de una valoración mínima de 0 puntos y una máxima de 10. El 73% del alumnado (43 alumnos) obtuvo una calificación de 5 o más puntos, mientras que el 27% (16 alumnos) restante obtuvo una nota inferior a 5 puntos. En cuanto a calificaciones extremas, se han dado casos de notas mínimas de 1 punto y máximas de 9 puntos.

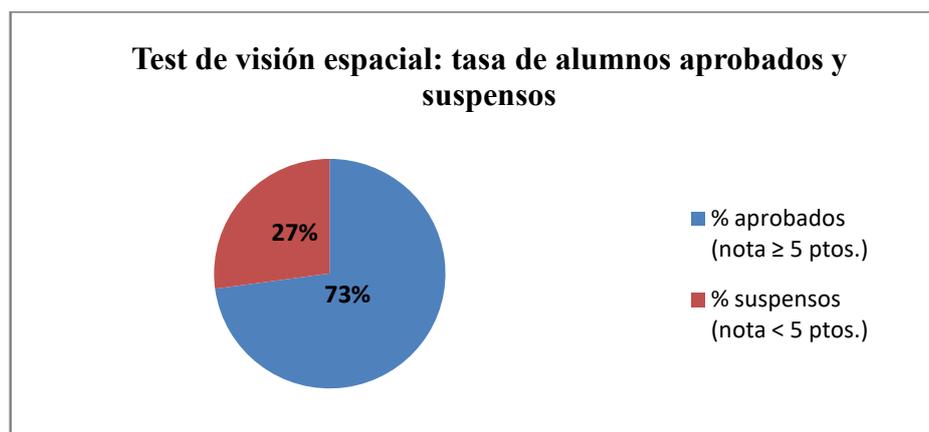


Fig.3. Resultados de test de visión espacial expresado en porcentaje de alumnos.

En cuanto a distribución del alumnado según nota, el 10% (6 alumnos) obtuvo una nota inferior a 4 puntos, mientras que la mayor parte, un 81% (48 alumnos), logró una calificación entre 4 y 7 puntos. Por último, el 9% restante (5 alumnos) consiguió un resultado superior a 7 puntos.

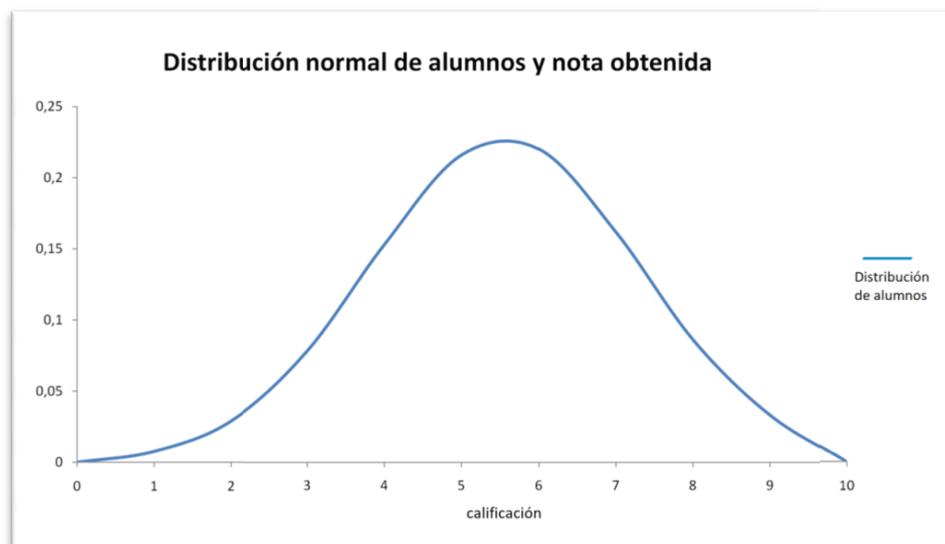
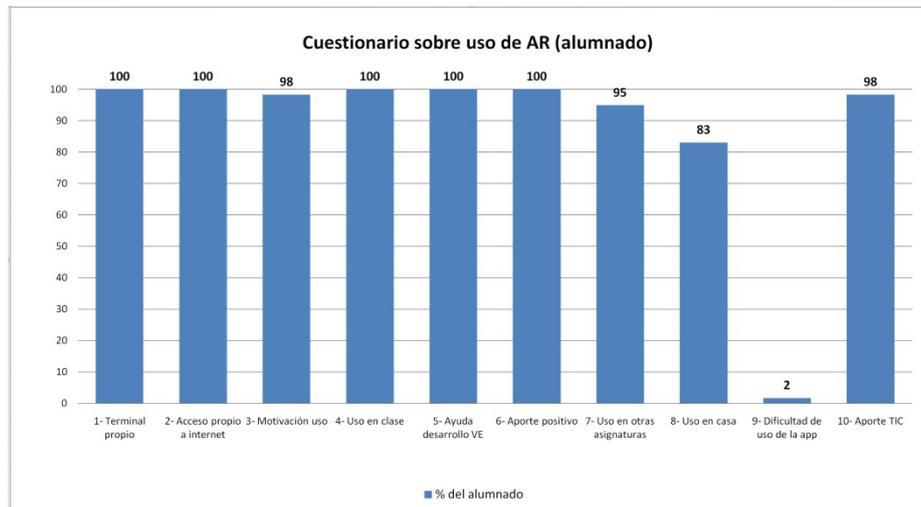


Fig.4. Gráfico de distribución normal de alumnos con respecto a nota obtenida.

## 5.2 Percepciones sobre el uso de realidad aumentada

### Percepciones del alumnado

Tras la sesión práctica con AR llevada a cabo, la opinión conformada por el alumnado participante (59 alumnos) acerca del uso de esta herramienta tecnológica fue la siguiente (ver Fig.5): todos y cada uno de los alumnos poseían terminal móvil propio y acceso a internet (ya sea a través de internet móvil y/o wifi). Además, un punto a destacar es el de la motivación existente a la hora de su uso, pero también de su uso en clase. Todos consideran que la AR les brinda una gran ayuda en el desarrollo de la habilidad de visión espacial, además de ser un aporte positivo como complemento al libro de texto o como elemento para ayudar a comprender contenidos de cierta dificultad. En general, prácticamente todos (un 95%) considera útil el uso de la AR en otras asignaturas, además de en la de Tecnología. Incluso un 83% está dispuesto a usarlo en casa a la hora de estudiar y/o para repasar y comprender lo explicado en clase por el docente. Prácticamente nadie tuvo problema alguno en manejar la app de AR y, por último, casi la totalidad del alumnado (un 98%) cree que el simple uso de la AR les proporciona también un conocimiento en el manejo de terminales móviles, de internet y competencia sobre realidad aumentada y las distintas posibilidades que ofrece y podría ofrecer en el futuro en diferentes ámbitos.

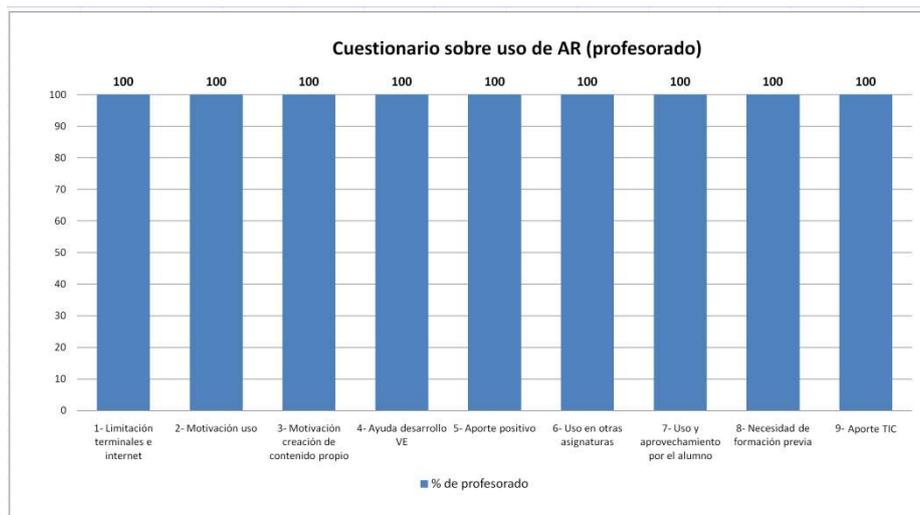


**Fig.5.** Resultados obtenidos tras la realización del cuestionario específico dirigido al alumnado participante (59 alumnos).

### Percepciones del profesorado

Con respecto al profesorado participante (3 profesores), tras todo el proceso previo de presentación de la AR y experiencia práctica llevada a cabo, se obtuvieron las siguientes percepciones (ver Fig.6): todos y cada uno de ellos fueron coincidentes en todas las respuestas arrojadas hacia las cuestiones planteadas. Para comenzar, todos creen que la necesidad de terminales móviles y de un punto de acceso a internet podría ser, en determinadas ocasiones, una barrera a la hora de utilizar la tecnología de AR. Esto radica en que pueden darse casos en los que el alumnado no posea terminales propios, que el centro educativo no pueda proporcionarlos o, también, porque pueden existir centros con una localización remota en los que el acceso a internet sea inviable. Todo esto dificultaría de forma destacable el uso de la AR. Por otro lado, es destacable la motivación que tienen los docentes para usar esta herramienta tecnológica, además de estar dispuestos a crear incluso contenido propio para personalizar y sacar mayor provecho de sus clases impartidas. Además, se da el caso de coincidir con el alumnado en diversos aspectos. Éstos son: opinión favorable de la AR acerca de su ayuda a la mejora de la visión espacial del alumnado. Herramienta con aporte positivo (como complemento al libro de texto y para ayudar en explicaciones de cierta dificultad). Utilidad a la hora de aplicarlo en otras asignaturas además de en la de Tecnología. Estimación positiva en cuanto a la mejora de la competencia TIC del alumno con el simple uso de esta tecnología, reforzando el conocimiento y destreza del uso de dispositivos móviles, manejo de internet y conocimiento de la AR. En general, los docentes también destacan que la mayor parte del alumnado sacaría provecho de esta tecnología, lo cual sería un elemento a tener en cuenta en una potencial mejora académica. Por último, también destacan de manera notable la necesidad de

una formación al profesorado previo al uso de esta herramienta, ya que, al perseguir su uso de forma extensiva, muchos docentes pueden no haber tenido nunca contacto con esta tecnología, desconociéndola totalmente. Además, si se tiene un conocimiento sobre AR, también se podrá sacar mayor rendimiento a las clases impartidas y resolver, de manera más rápida, posibles problemas técnicos que puedan surgir en su utilización.



**Fig.6.** Resultados obtenidos tras la realización del cuestionario específico dirigido al profesorado participante (3 docentes).

## 6 Conclusiones

Tras la breve investigación realizada, se han podido confirmar diferentes hipótesis planteadas inicialmente. De forma concreta, se ha confirmado la existencia de un porcentaje de alumnos (un 27%. Valor a tener bastante en cuenta) que tiene ciertas dificultades en cuanto a visión espacial. Además, se ha ratificado la existencia de una desigualdad de nivel entre alumnos de un mismo curso y grupo.

En cuanto al uso de AR, queda presente que la necesidad de terminales móviles y de acceso de internet puede ser una traba para utilizar esta tecnología. Aunque, para una primera toma de contacto, al estar cada vez más extendida la presencia del teléfono móvil y de internet en la sociedad de hoy en día, podría ser una opción que el alumnado utilizara sus propios terminales. Al menos para una prueba inicial que pueda hacer ver, tanto a éstos, como al profesorado, los diferentes aspectos positivos y negativos que puede ofrecer el uso de esta herramienta. Por otro lado, se puede ver también que existe una disposición y motivación por los alumnos y docentes para usar AR y, relacionado con la mejora de visión espacial, ambas partes intervinientes consideran que contribuiría a desarrollar esta competencia. La ayuda que esta herramienta puede proporcionar en este aspecto se podría extrapolar también a otros ámbitos y

materias. Esto, sumado a la posibilidad de aplicación de la AR a diferentes niveles académicos, abre de forma considerable el abanico de uso de esta tecnología en educación, además de potenciarla y de desarrollar cada vez más la AR y sus futuras aplicaciones. Por tanto, respaldado por diversos estudios realizados y por la percepción positiva que se tiene de esta tecnología, se podría afirmar que la AR puede ofrecer una potencialidad muy a tener en cuenta en cuanto al desarrollo, mejora y cambio del modelo educativo clásico hasta ahora establecido. Al fin y al cabo, es innegable que la revolución tecnológica que hemos y estamos sufriendo ha supuesto un innegable aporte positivo en nuestro día a día. Por tanto, sin dejar de lado el actual arquetipo educativo, ¿por qué no aprovechar ese avance tecnológico para complementar ese modelo existente? ¿Por qué no utilizar diferentes fórmulas para mejorar la formación y el éxito de nuestros alumnos y futuros ciudadanos? A todo esto habría que sumarle el compromiso con el medio ambiente que nos puede brindar la AR. Ésta nos ofrece la posibilidad de poder disponer, de forma rápida y sencilla, de cientos de maquetas tridimensionales e información de diferente índole. Pero además, a esta idea hay que agregarle el gasto, tanto material, como energético, que nos podemos ahorrar al usar AR. Todo ello con respecto a la elaboración, transporte y necesidad de espacio que supondría el uso de maquetas físicas. Por tanto, con la utilización de AR, todo se reduciría a la simple tenencia de un terminal móvil.

## Agradecimientos

Para la realización de este trabajo se ha contado con la participación de Manuel Solera Iglesias, profesor del IES Ilíberis (localidad de Atarfe, Granada, España). Ha brindado una inestimable colaboración y ayuda a la hora de llevar a cabo la experiencia práctica con AR en el aula. Habiendo conformado, por tanto, una de las piezas clave para la realización de la presente investigación.

## Referencias

1. Cupitra-García, A. & Duque-Bedoya, E.: Profesores aumentados en el contexto de la realidad aumentada: una reflexión sobre su uso pedagógico. *El Ágora USB* 18(1), 245–255 (2018).
2. De la Gala, K. & Vera, J.: Uso de la realidad aumentada mejora los niveles de comprensión lectora en estudiantes de quinto grado del nivel primario. *CITIE 2018: Congreso Internacional de Tendencias e Innovación Educativa*. Arequipa (Perú).
3. Gavilanes, W., Abásolo, M. & Cuji, B. Resumen de revisiones sobre realidad aumentada en educación. *Espacios* 39(15), 14-31 (2018).
4. Gutiérrez, R., Duque, E., Chaparro, R. & Rojas, N.: Aprendizaje de los conceptos básicos de realidad aumentada por medio del juego Pokémon Go y sus posibilidades como herramienta de mediación educativa en Latinoamérica. *Información tecnológica* 29(1), 49-58 (2018).
5. Henao, A., Bolívar, G., Gómez, C., Restrepo, S., Velásquez, C. & Gómez, L.: Aplicaciones de realidad aumentada en educación para mejorar los procesos de enseñanza-aprendizaje: una revisión sistemática. *Espacios* 39(49), 3-17 (2018).

6. León, F., Duque, E. & Escobar, P.: Estrategias de formulación de preguntas de calidad mediadas por realidad aumentada para el fortalecimiento del pensamiento científico. *Revista mexicana de investigación educativa* 23(78), 791-815 (2018).
7. Pérez, S. & Robles, B.: Objetos de realidad aumentada: percepciones del alumnado de Pedagogía. *Píxel-Bit. Revista de Medios y Educación* (53), 207-220 (2018).
8. Prendes, C.: Realidad aumentada y educación: análisis de experiencias prácticas. *Píxel-Bit. Revista de Medios y Educación* (46), 187-203 (2015).
9. Toledo, P. & Sánchez, J.: Realidad aumentada en educación primaria: efectos sobre el aprendizaje. *RELATEC: Revista Latinoamericana de Tecnología Educativa* 16(1), 79-92 (2017).
10. Moreno-Garrido, A.J.: Realidad aumentada como herramienta en el aula. Ayuda a la mejora de la visión espacial. Trabajo Fin de Máster, MAES, Universidad de Granada (2019).

## **Aplicaciones de Realidad Virtual y Realidad Aumentada como soporte a la enseñanza del Dibujo Técnico**

Yoselie Alvarado, Nicolás Jofré, María Rosas, Roberto Guerrero

Laboratorio de Computación Gráfica, Dpto de Informática  
FCFMyN, Universidad Nacional de San Luis  
San Luis, Argentina  
Ciudad, País  
{ymalvarado, npasinetti, mvrosas, rag}@unsl.edu.ar

**Resumen.** La popularización y expansión de Internet y las Nuevas Tecnologías de la Información y la Comunicación han modificado la vida de las personas en diversos aspectos. La educación no se ha mantenido al margen de esta revolución tecnológica. En este sentido, la Realidad Virtual (RV) y la Realidad Aumentada (RA) son alternativas prometedoras para dar soporte a la enseñanza. Estas tecnologías se proyectan en el campo de la educación superior como una tendencia que podría transformar significativamente las prácticas educativas. El presente trabajo reporta una implementación y uso de aplicaciones de RV y RA en un curso vinculado al dibujo técnico en la Universidad Nacional de San Luis (UNSL). Las aplicaciones desarrolladas incorporan gráficos generados por computadora con el fin de facilitar la transmisión y comprensión de conceptos complejos mediante información visual. Los resultados preliminares muestran un aumento en la cantidad de alumnos que aprobaron los exámenes en primera instancia.

**Palabras Clave:** Computación Gráfica, Realidad Virtual, Realidad Aumentada, Dibujo Técnico, Educación Superior.

**Abstract.** Internet popularization and expansion and the New Information and Communication Technologies have changed people's life in several aspects. Education has not been kept apart from this technological revolution. In this sense, Virtual Reality (RV) and Augmented Reality (AR) are promising alternatives to support teaching. These technologies are trending in the field of higher education that could significantly transform educational practices. This work reports the implementation and use of RV and RA applications in a course related to technical drawing at the National University of San Luis (NUSL). Developed applications involve computer-generated graphics in order to facilitate the transference of skills and know-how of complex concepts through visual information. Preliminary results show an increase in the number of students who passed the exams in first instance.

**Keywords:** Computer Graphics, Virtual Reality, Augmented Reality, Technical Drawing, Higher Education.

## 1 Introducción

La Realidad Virtual y la Realidad Aumentada existen como nuevas tecnologías desde los años ochenta y desde entonces se han mantenido en constante evolución, aunque siempre relacionadas con el mundo de los videojuegos y del entretenimiento. El renacimiento de los gadgets y su popularización ha permitido la migración de los mismos hacia otras aplicaciones como la educación, permitiendo integrar la tecnología en las aulas [1, 2].

Las realidades alternativas son mundos que se disfrutan a través de la RV y la RA, los cuales son creados desde cero, ya sea como mundos completamente ficticios o como una adaptación de entornos reales. Esto implica que, en esencia, se pueden crear infinitos mundos: las realidades alternativas ofrecen infinitas posibilidades [3, 4].

Los entornos de Realidad Virtual ofrecen un nuevo tipo de interacción multimodal en la cual las personas pueden sentir de diversas maneras el mundo virtual. Podría decirse que la RV es una técnica basada en gráficos por computadora, que involucra interacción hombre-máquina basada en tecnología multimedia y los sentidos del usuario. La RV alcanza a terrenos de amplia interdisciplinariedad. La RA por su parte es una tecnología que mezcla el mundo real con el mundo virtual. En ella se superponen diferentes capas de información al mundo físico ofreciendo una nueva experiencia de la realidad. Esta experiencia ofrece multitud de posibilidades de aplicación, ya que no solo se puede acceder a la información sino que se puede interactuar con ella [5, 6].

Por otro lado, es un hecho cada vez más evidente que las prácticas convencionales de enseñanza a menudo experimentan dificultades para mantener a los estudiantes motivados y comprometidos [7]. Debido a ello, en el ámbito educativo es cada vez más frecuente la incorporación de tecnologías que enriquecen los procesos de enseñanza y aprendizaje [8, 9].

Los videojuegos son muy exitosos en mantener durante horas altos niveles de motivación y compromiso a través de una serie de tareas, sin aparente pérdida de enfoque [10, 11]. La comunidad académica está interesada en explorar métodos que puedan generar una participación activa del estudiante, es por ello que ha mostrado un gran interés en adoptar realidades alternativas como un medio para aumentar la participación de los estudiantes y, a su vez, favorecer aprendizajes significativos [12].

La gran ventaja y beneficio que aporta la RV en educación es el ampliar las posibilidades de aprendizaje de los estudiantes. No obstante, no se debe pasar por alto la importancia del hecho que los estudiantes estén en contacto con la realidad que les rodea [13]. La RA es una alternativa válida que permite conectar el mundo real con contenidos digitales. Esta característica permite al usuario reforzar el aprendizaje de los contenidos educativos mediante su asociación con el mundo real [14, 15].

Las realidades alternativas permiten salir del aula y aprender de lo que se ve. Se trata de un concepto diferente de aprendizaje basado en el descubrimiento. Del mismo modo son herramientas muy útiles en una clase ya que favorecen y profundizan la integración de los conceptos mediante la visualización de modelos 3D [16]. Una de las claves de su uso en educación es que abren un mundo de posibilidades sin siquiera moverse del escritorio [17].

La RV y la RA encajan a la perfección como herramientas de apoyo a la formación de profesionales, ya que posibilitan mejorar la enseñanza y los procesos de aprendizaje logrados en conjunto con los enfoques educativos tradicionales [18, 19]. De esta manera es posible una educación de calidad, una enseñanza y aprendizaje a otro nivel, independientemente de la institución educativa, el entorno social, o incluso del docente asignado. La tecnología y los recursos, cada vez más asequibles, pueden ser un camino evidente hacia la equidad y calidad educativa mitigando muchas de las deficiencias actuales [20].

En función de lo expuesto, este trabajo ilustra los resultados obtenidos a partir de la exploración de las sinergias existentes entre la Realidad Virtual y la Realidad Aumentada, y su aplicación en la enseñanza universitaria.

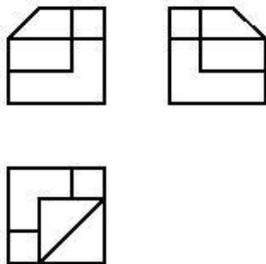
Las secciones 2 y 3 describen los conceptos de la asignatura que fueron incorporados en este ensayo y la aplicación de cada uno de los sistemas de realidad alternativa construidos. La sección 4 presenta los resultados y discusión. Finalmente, la sección 5 presenta las conclusiones correspondientes.

## 2 Sistemas de representación

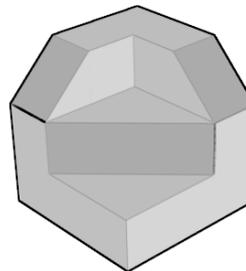
La asignatura Sistemas de Representación tiene por objetivo la enseñanza de diferentes técnicas de dibujo para representar objetos tridimensionales en un espacio bidimensional. De acuerdo al marco de aplicación, los sistemas de representación se pueden clasificar como [21, 22]:

**Sistemas de medida.** Sus dibujos se enfocan en la representación de las dimensiones y posición de los objetos, de forma que sea posible realizar mediciones a partir del dibujo. Los sistemas de medida no incluyen una representación de la profundidad de los objetos, por lo tanto las formas de los objetos suelen ser obtenidas mediante la composición (mental) de diferentes representaciones desde varios puntos de vistas; dicha reconstrucción es una tarea que debe realizar el observador. Algunos ejemplos de sistemas de medidas son Planos Acotados, Sistema Triédrico, entre otros.

**Sistemas representativos.** Involucran la representación de las formas de los objetos desde un único punto de vista, ya que el dibujo permite visualizar la profundidad de los objetos. Los dibujos de sistemas representativos suelen ser difíciles de realizar y usualmente no es posible realizar medidas a partir del dibujo. Algunos ejemplos de sistemas representativos son Isométrico, Caballera, Militar, entre otros.



**Figura 1.** Sistema Triédrico.



**Figura 2.** Sistema Isométrico.

A fin de ejemplificar, la Figura 1 muestra la representación mediante el Sistema Triédrico de un objeto tridimensional. Este involucra la vista de Alzada, Perfil y Planta. De manera similar, la Figura 2 ilustra la representación del mismo objeto mediante el sistema de representación Isométrico.

### 3 Caso de estudio

La experiencia fue realizada por los docentes de la asignatura Sistemas de Representación, la cual se dicta para las carreras de Ingeniería en Computación, Ingeniería en Informática y Tecnicatura Universitaria en Energías Renovables de la Facultad de Ciencias Físico Matemáticas y Naturales (FCFMyN) en la UNSL. Los alumnos de la asignatura se encuentran cursando segundo y tercer año de sus respectivas carreras.

El experimento consistió en el desarrollo de una aplicación de RV y una de RA, las cuales fueron realizadas por integrantes de la asignatura que a la vez pertenecen al Laboratorio de Computación Gráfica, FCFMyN, UNSL. Dichas aplicaciones fueron usadas durante las clases prácticas del ciclo lectivo 2017 con el fin de incentivar el desarrollo de la capacidad espacial de los estudiantes y, de esta manera, facilitar la integración de los conceptos involucrados en la resolución de los ejercicios propuestos en los prácticos de la asignatura.

#### 3.1 Experiencia con Realidad Virtual

El uso de la aplicación de Realidad Virtual consistió en la visualización de los modelos tridimensionales de objetos correspondientes a los primeros prácticos de la asignatura Sistemas de Representación.

En la realización de los ejercicios de los prácticos, se les proporcionó a los alumnos la vista Isométrica de un objeto 3D a partir de la cual debían obtener el correspondiente Sistema Triédrico; es decir, las vistas de Alzada, Perfil y Planta. En resumen, la tarea consistió en la transformación desde el Sistema Isométrico al Sistema Triédrico.

Para completar esta actividad el alumno requiere, en su imaginario, girar el objeto tridimensional para visualizar y dibujar cada una de las vistas solicitadas. La aplicación de RV tiene el fin de asistir y verificar este proceso.

Inicialmente, la aplicación ya dispone de los modelos tridimensionales correspondientes al práctico, tal como se muestra en la Figura 3. Mediante el uso del mouse, el alumno debe decidir qué modelo desea manipular. Una vez seleccionado el modelo, es posible girarlo en los tres ejes mediante el uso del mouse, visualizándolo en un modo libre. Ver Figura 4.

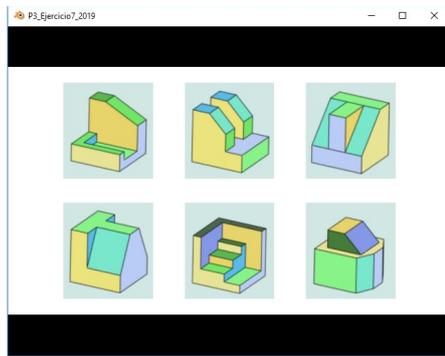


Figura 3. Menú.

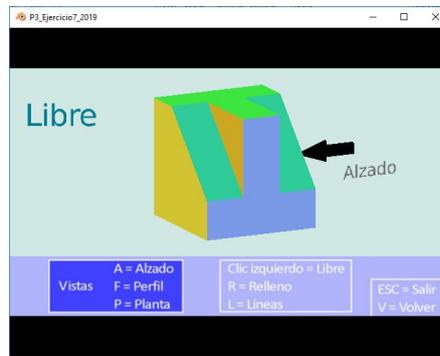


Figura 4. Modo Libre.

Una vez completado el Sistema Triédrico, el alumno puede verificar su resolución. Para ello la aplicación permite visualizar las tres vistas mediante las teclas **A** (Alzada), **F** (Perfil), **P** (Planta). La Figura 5 ejemplifica la visualización de la vista de Perfil en la aplicación.

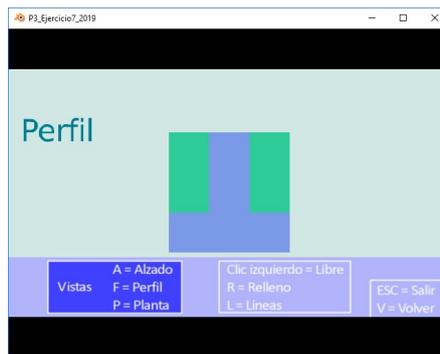


Figura 5. Vista de Perfil.

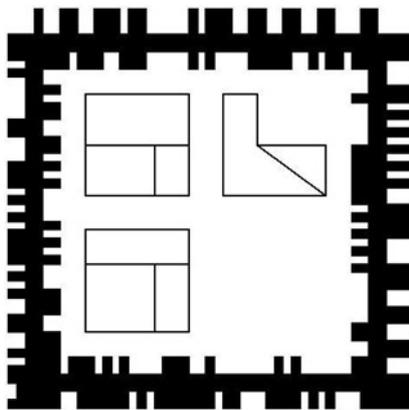
El sistema de Realidad Virtual desarrollado es una aplicación de escritorio realizada para el sistema operativo Microsoft Windows. Tanto los modelos 3D como el sistema en sí mismo fueron realizados con la herramienta Blender 3D [23].

### 3.2 Experiencia con Realidad Aumentada

La Realidad Aumentada favoreció el proceso de reconstrucción de objetos tridimensionales, es decir, el sentido inverso al desarrollado con la RV: desde el Sistema Triédrico al Sistema Isométrico. Esta actividad corresponde a los últimos prácticos de la asignatura Sistemas de Representación, en los cuales la complejidad para los estudiantes es mayor.

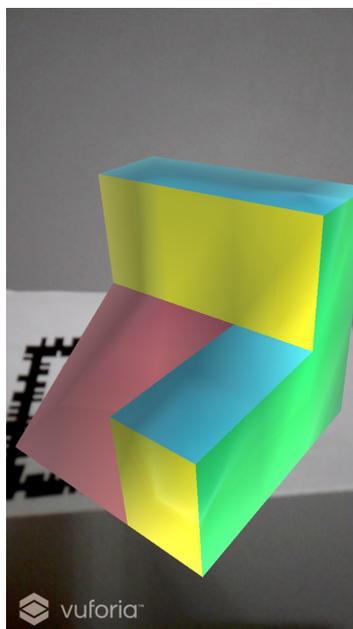
En esta ocasión, se les facilitaron a los alumnos únicamente las vistas bidimensionales que corresponden a la Alzada, Perfil y Planta del Sistema Triédrico (ver Figura 1) con las cuales se debe reconstruir la figura tridimensional correspondiente. El alumno debe intentar reconstruir, en su imaginación, el objeto tridimensional que corresponde a las tres vistas suministradas, para luego dibujar el correspondiente Sistema Isométrico (ver Figura 2). La aplicación de RA permite verificar este proceso.

Inicialmente, la aplicación dispone de todos los modelos 3D correspondientes a los Sistemas Triédricos de los prácticos. Posteriormente, a cada Sistema Triédrico se le asoció un patrón único de manera que las imágenes resultantes pudieran ser marcadores de RA. Cada marcador fue vinculado en la aplicación de RA con su correspondiente modelo tridimensional. La Figura 6 muestra uno de los marcadores construidos para su uso con RA.



**Figura 6.** Marcador de RA.

Para el uso de la RA el alumno debe seleccionar el marcador que contenga el Sistema Triédrico a verificar y visualizarlo mediante la cámara de su celular. Como resultado la aplicación muestra el modelo 3D correspondiente, tal como lo ilustra la Figura 7.



**Figura 7.** Aplicación de RA.

El sistema de RA desarrollado es una aplicación móvil realizada para el sistema operativo Android, la cual puede ser descargada desde la tienda Google Play [24]. Para su desarrollo se utilizó la herramienta Unity [25] con la incorporación de la librería Vuforia [26] para RA. Por su parte, los modelos 3D utilizados fueron realizados en Blender 3D y los marcadores fueron construidos con la herramienta de diseño asistido por computadora LibreCad [27].

#### **4 Resultados y Discusión**

Con el fin de valorar el impacto de la experiencia se registraron los resultados de dos evaluaciones realizadas durante el ciclo lectivo 2017 de la asignatura. Las evaluaciones involucraron el desarrollo de aspectos teóricos y prácticos. Sus resoluciones se realizaron en papel y también mediante el uso de la herramienta LibreCad en computadora.

**Primera evaluación.** Involucra los conceptos vinculados a los Sistemas de Medida, especialmente el Sistema Triédrico, y su proceso de dibujado utilizando la aplicación LibreCad. Esta evaluación incluyó ejercicios similares a los practicados con la aplicación de Realidad Virtual.

**Segunda evaluación.** Involucra los conceptos relacionados a los Sistemas Representativos y su correspondiente plasmado. Esta evaluación incluyó ejercicios consistentes en la transformación desde el Sistema Triédrico a diversos Sistemas Representativos; algunos de ellos similares a los utilizados con la aplicación de Realidad Aumentada.

A modo de poder contrastar los resultados obtenidos, se utilizó como referencia los datos de rendimiento académico correspondientes al ciclo de cursado 2016, durante el cual no se utilizó ningún tipo de herramienta de soporte.

La Tabla 1 muestra los porcentajes de aprobación de cada uno de los exámenes durante los ciclos 2016 y 2017.

**Tabla 1.** Porcentajes de aprobación.

Ciclo lectivo	Primera evaluación	Segunda evaluación
2016	66%	55%
2017	78%	60%

En función de los valores de la Tabla 1, la mejora lograda por la experiencia respecto al año anterior se muestra en la Tabla 2.

**Tabla 2.** Porcentajes de mejora.

	Primera evaluación	Segunda evaluación
Porcentaje mejora	15%	8%

De forma similar, se recolectaron los datos correspondientes a los alumnos que abandonaron la asignatura antes de terminar el dictado del curso. La Tabla 3 muestra los porcentajes de deserción correspondientes.

**Tabla 3.** Porcentajes de deserción.

Ciclo lectivo	Deserción
2016	10%
2017	8%

De acuerdo a los datos que se ilustran en la Tabla 3, el porcentaje de deserción disminuyó notablemente respecto al año anterior.

## 5 Conclusiones

En este artículo se describe el desarrollo e implementación de herramientas de Realidad Virtual y Realidad Aumentada para dar soporte a la enseñanza de conceptos relacionados al dibujo técnico. Las aplicaciones fueron construidas a partir de los ejercicios prácticos que se dictan en la asignatura Sistemas de Representación. El fin de las mismas fue auxiliar a los alumnos en aquellos procesos de aprendizaje cognitivos que involucran el imaginario tridimensional. Para facilitar la instalación y el uso de las aplicaciones, las mismas fueron desarrolladas para los sistemas operativos de mayor popularidad entre los alumnos: Microsoft Windows y Android.

Este trabajo intenta mostrar mediante la experiencia los beneficios que se pueden obtener al complementar la enseñanza tradicional con prácticas novedosas de transmisión de conocimiento. Sin embargo, hay varios aspectos para mejorar alrededor de los sistemas desarrollados, su aplicación y su evaluación.

De acuerdo con los resultados obtenidos, se observa que: durante el ciclo lectivo 2017 se logró una mejora del 15% en los resultados de la primera evaluación y una mejora del 8% en la segunda evaluación. Lo dicho evidenciaría que la Realidad Virtual es más efectiva que la Realidad Aumentada al momento de ser utilizada como herramienta de apoyo en la enseñanza de conceptos espaciales. Sin embargo, se debe tener en cuenta que los conceptos involucrados y el proceso cognitivo requerido en la segunda evaluación son mucho más elevados que en la primera evaluación; por lo tanto es esperable que los resultados de la primera evaluación sean mejores. Las experiencias se repitieron de manera similar durante el ciclo lectivo 2018 donde los resultados obtenidos fueron semejantes a los del ciclo 2017. Por consiguiente, se ha decidido incorporarlas en forma permanente como herramientas de soporte. Resta un análisis que permita depurar las aplicaciones a modo de lograr una propuesta superadora.

En cuanto al aprendizaje a largo plazo, consideramos que es necesario realizar un seguimiento de los alumnos en asignaturas posteriores que involucren temáticas afines, por ejemplo en la asignatura posterior Computación Gráfica. Los resultados también muestran que el porcentaje de deserción de alumnos en la asignatura disminuyó respecto al año anterior. Si bien este valor es un buen indicador del potencial de las herramientas utilizadas, se requiere de un análisis más detallado en futuras evaluaciones debido a que los factores que afectan el parámetro de deserción suelen provenir de diversas fuentes.

## Referencias

1. S. Kirkley and J. Kirkley, "Creating next generation blended learning environments using mixed reality, video games and simulations", vol. 49, pp. 42–53, 2004.
2. A. P. Altomari, "Realidad virtual y realidad aumentada en la educación, una instantánea nacional e internacional", *ECONOMA CREATIVA.*, vol. 0, no. 7, pp. 34–65, 2017.
3. I. de A. Souza-Concilio and B. A. Pacheco, "The development of augmented reality systems in informatics higher education", *Procedia Computer Science*, vol. 25, pp. 179 – 188, 2013. 2013 International Conference on Virtual and Augmented Reality in Education.

4. L. Dieker, et al., "Mixed reality environments in teacher education: Development and future applications", Online in real time: using web 2.0 for distance education in rural special education, p. 116, 2016.
5. I. Marquez, "La simulación como aprendizaje: educación y mundos virtuales", in II Congreso Internacional de Comunicación, vol. 3, 2010.
6. Y. tian Ye, "Design and implementation of digital art teaching system based on interactive virtual technology," International Journal of Emerging Technologies in Learning (iJET), vol. 11, no. 11, pp. 49–54, 2016.
7. H.-H. Liou, S. J. H. Yang, S. Y. Chen, and W. Tarn, "The influences of the 2d image-based augmented reality and virtual reality on student learning", Journal of Educational Technology Society, vol. 20, no. 3, pp. 110–121, 2017.
8. C. Prendes Espinosa, "Realidad aumentada y educación: análisis de experiencias prácticas", Pixel-Bit. Revista de Medios y Educación, no 46, pp. 187-203, 2015.
9. P. Chen, X. Liu, W. Cheng, and R. Huang, "A review of using augmented reality in education from 2011 to 2016," in Innovations in Smart Learning, pp. 13–18, Springer, 2017.
10. C. Steinkuehler, "Videogames as a serious medium", On the Horizon, vol. 24, no. 2, pp. 175–178, 2016.
11. A. N. Pfannenstiel, "Videogames in the class-room: student discussion leader presentations", On the Horizon, vol. 24, no. 1, pp. 100–103, 2016.
12. L. M. B. Sierra, R. S. Gutierrez, and C. L. Garzn-Castro, "Second life as a support element for learning electronic related subjects: A real case", Computers Education, vol. 58, no. 1, pp. 291–302, 2012.
13. G. Jin and S. Nakayama, "Experiential learning through virtual reality: Safety instruction for engineering technology students," vol. 30, pp. 16–23, 09 2013.
14. H.-K. Wu, et al., "Current status, opportunities and challenges of augmented reality in education", Computers Education, vol. 62, pp. 41–49, 2013.
15. M. Bower, C. Howe, N. McCredie, A. Robinson, and D. Grover, "Augmented reality in education - cases, places, and potentials", in 2013 IEEE 63rd Annual Conference International Council for Education Media (ICEM), pp. 1–11, Oct 2013.
16. Z. Merchant, E. T. Goetz, L. Cifuentes, W. Keeney-Kennicutt, and T. J. Davis, "Effectiveness of virtual reality-based instruction on students' learning outcomes in k-12 and higher education: A meta-analysis", Computers Education, vol. 70, pp. 29–40, 2014.
17. J. D. la Torre Cantero et al., "Entorno de aprendizaje ubicuo con realidad aumentada y tabletas para estimular la comprensión del espacio tridimensional", Revista de Educación a Distancia, vol. 0, no. 37, 2015.
18. De La Torre Cantero, JL. Aplicación de tecnologías gráficas avanzadas como elemento de apoyo en los procesos de enseñanza-aprendizaje del dibujo, diseño y artes plásticas, 2013.
19. A. Alvarez-Marin, et al., "Realidad Aumentada como Apoyo a la Formación de Ingenieros Industriales", Formación universitaria, vol. 10, pp. 31–42, 2017.
20. E. Durall Gazulla, et al., "Perspectivas tecnológicas: educación superior en Iberoamérica 2012-2017", 2012.
21. E. Lamadrid and J. Gonzalez, Sistemas de representación y Dibujo Técnico. Textos universitarios, Ediciones de la Universidad de Oviedo, 2007.
22. C. OPI and I. RUIZ, Dibujo técnico II. Ediciones Paraninfo, S.A, 2016.
23. <https://www.blender.org/>.
24. <https://play.google.com/store/apps/details?id=com.UNSL.SR>.
25. <https://unity3d.com/es>.
26. <https://www.vuforia.com/>.
27. <http://librecad.org>.

## Desarrollo de un repositorio de certificación basado en Blockchain para la evaluación de una asignatura de grado

Pedro A. Castillo<sup>1</sup>, Pablo García-Sánchez<sup>2</sup>, M. Carmen Aguilar-Luzón<sup>3</sup>, Julia García Cabello<sup>4</sup>, and Enrique Herrera-Viedma<sup>5</sup>

<sup>1</sup> Department of Computer Architecture and Technology, CITIC  
University of Granada, Spain

`pacv@ugr.es`

<sup>2</sup> Department of Computer Science and Engineering. ESI  
University of Cádiz, Spain

`pablo.garciasanchez@uca.es`

<sup>3</sup> Department of Social Psychology  
University of Granada, Spain

`maguilarluzon@ugr.es`

<sup>4</sup> Department of Applied Mathematics  
University of Granada, Spain

`cabello@ugr.es`

<sup>5</sup> Department of Computer Science and Artificial Intelligence  
University of Granada, Spain

`viedma@decsai.ugr.es`

**Abstract.** La tecnología Blockchain se puede entender como una base de datos distribuida en Internet, sin un control central, con una serie de características que transformarán la gobernanza, la economía, el funcionamiento de las organizaciones y, por supuesto, la educación. En Blockchain la información se almacena en forma de bloques, con un alto nivel de seguridad que evita la manipulación de la información almacenada. Debido a las oportunidades que ofrece en el mundo digital, han surgido varias aplicaciones piloto para utilizar esta tecnología en la educación. Por ejemplo, mediante el uso de la cadena de bloques, se han creado varios proyectos para proporcionar a los estudiantes diplomas digitales o certificados de estudios. En este trabajo se propone la creación de un repositorio de certificación basado en Blockchain que pueda ser utilizado por una institución educativa para incorporar información sobre la carrera y trayectoria que han desarrollado los estudiantes, tales como asignaturas cursadas, competencias y habilidades obtenidas, logros, cursos o estancias realizadas. Concretamente se propone una aplicación distribuida para almacenar y certificar las calificaciones obtenidas por un estudiante en una asignatura dada. Mediante el uso de la tecnología de Blockchain garantizamos la transparencia y la confianza, facilitando así a los formadores y empleadores en el futuro, de forma abierta y certificada, comprobar qué objetivos ha cumplido cada estudiante.

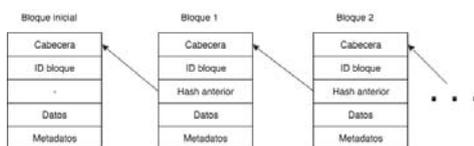
**Palabras clave:** Blockchain, Repositorio de Certificación, Docencia Grados Universitarios, Evaluación, Transparencia

**Abstract.** Blockchain is a distributed database, with no central control, spread across many computers that could transform governance, the economy, the functioning of organisations, and of course education. Information is stored in form of blocks, protected by complex algorithms that are hard to hack and cannot be manipulated. Due to the opportunities it offers in the digital world, several pilot applications have emerged in order to use this technology in education. For example, using blockchain, several projects have been created to provide students with digital diplomas and certificates of studies. The idea is to implement a distributed registry in which information about the career and path that a student has been doing (subjects, competencies, achievements, courses, grades, stays) is incorporated. By using blockchain technology we ensure transparency and trust, thus making it easier for trainers and employers in the future, in an open and certified way, to see what objectives each student has met. The management of these digital titles would allow to link them with the complete register of competences and abilities surpassed by each student in any institution.

**Keywords:** Blockchain, Certification Repository, University Education, Evaluation, Transparency

## 1 Introducción

La tecnología Blockchain (BC) está basada en una red descentralizada y segura que puede facilitar la transmisión y el almacenamiento de datos [1, 2]. BC es una tecnología basada en una topología peer-to-peer (P2P) que permite a diferentes tipos de usuarios crear y mantener un registro inalterable de transacciones, cada una de ellas sellada y vinculada a la anterior, de forma que, cada vez que se añade información, se convierte en otro bloque en la cadena [3] (ver Figura 1). Tal y como está desarrollada, es una tecnología de una sola escritura que permite añadir infinitos bloques de datos, lo que la convierte en un registro verificable y auditable de todas y cada una de las transacciones.



**Fig. 1.** Esquema de la cadena de bloques en una red blockchain.

En los últimos años ha despertado un gran interés y se espera que suponga un gran avance tecnológico a nivel empresarial y de investigación, como lo fue la inteligencia artificial, la IoT y la VR. En realidad, representa un nuevo paradigma

para la forma en que se comparte la información, lo que permitirá el crecimiento de estas últimas tecnologías.

En realidad, esta tecnología no se espera que reemplace a las bases de datos relacionales corporativas, pero sí abre nuevas puertas para el almacenamiento de datos dentro y fuera de las empresas de una forma global. Permitirá el flujo de activos digitales de forma segura, lo que añadirá valor a muchos procesos industriales y aplicaciones tecnológicas.

Hasta el momento, diversas industrias han comenzado a realizar implementaciones usando esta tecnología de cadena de bloques para el almacenamiento de registros y transacciones básicas, pero sobre todo hay expectativas de nuevas aplicaciones en el mundo real en los próximos años, sobre todo en industrias como la logística y el transporte, corporaciones financieras, asistencia sanitaria y hasta pagos móviles.

Sin embargo, la clave para una innovación más efectiva pasa por estandarizar la tecnología e integrar múltiples plataformas y herramientas de BC dentro de una cadena de valor. Esto proporcionaría a las industrias plataformas para desarrollar ideas y utilizar el sistema para crear las aplicaciones tecnológicas más avanzadas para su uso en todas las empresas.

Inicialmente BC se asoció a Bitcoin principalmente, y se usó para transferencias financieras debido a la seguridad que garantiza [4, 5]. Sin embargo, no hay razón para que no se aplique a otras estructuras de datos que necesiten registros ordenados, e incluso utilizarla como parte de un sistema de almacenamiento y recuperación de información y documentos. Es más, recientemente se han desarrollado algunos proyectos innovadores para aplicar esta tecnología en el ámbito educativo [6].

En cualquier caso, diversos expertos destacan el estado incipiente de desarrollo de esta tecnología [7, 8], por lo que no se espera que a corto plazo aparezcan despliegues totalmente operativos y a gran escala que reporten beneficios directos a las empresas. Aunque sí supondrá, a corto-medio plazo, un medio de garantizar transparencia y la capacidad de crear redes de comunicación seguras y en tiempo real entre socios de todo el mundo para dar soporte a todo tipo de servicios.

En este trabajo proponemos la creación de un sistema de certificación basado en la tecnología BC para incorporar información sobre la trayectoria que han desarrollado los estudiantes durante su vida académica, tales como asignaturas cursadas, competencias y habilidades obtenidas o cursos realizados, y posteriormente durante su vida laboral sobre logros o proyectos desarrollados.

El resto del artículo está estructurado como sigue: En la Sección 2 se presenta un análisis de las aplicaciones en diferentes ámbitos en las que se ha propuesto usar BC. A continuación, en la Sección 3 se muestra el funcionamiento general de Hyperledger Fabric como software de desarrollo para implementar la red BC, y a continuación se detalla la arquitectura del prototipo propuesto, así como el proceso de creación, configuración y puesta en marcha del mismo. Por último, se presentan una serie de conclusiones y trabajos futuros (Sección 4).

## 2 ¿Cómo sacar partido de la tecnología blockchain en el ámbito educativo?

Debido a lo prometedor de BC, esta tecnología ha sido utilizada en los últimos años, además de para desarrollar criptomonedas, en campos como los sistemas de tarjetas de crédito [9] o bancarios [10], el comercio internacional [11], el registro de derechos de autor [12], en sistemas de transparencia [13] o para mantener registros de personas de manera segura en ámbitos como la salud [14, 15].

Esta tecnología puede ser la base que dé soporte en el ámbito educativo [16, 17], facilitando a formadores la herramienta para certificar conocimientos, competencias y grados adquiridos, al tiempo que facilite a los empleadores la información requerida para determinar si el CV de un candidato es adecuado y si las cualificaciones requeridas son válidas.

Así, varias instituciones educativas han desarrollado programas piloto para emitir diplomas a sus estudiantes mediante la tecnología BC. Como ejemplo destacaremos el proyecto "Blockcerts Wallet" del MIT [18] que facilita a los estudiantes mantener y compartir sus propios registros oficiales directamente con otras personas o instituciones de una manera segura y a prueba de manipulaciones. Proyectos similares se han puesto en marcha en las universidades de Nicosia (Chipre) o en la University College London [19].

Por otro lado, dentro del proyecto OpenBlockchain [20, 21] de la Knowledge Media Institute (KMI) de la Open University se han venido planteando ideas sobre cómo gestionar certificados para dar soporte al aprendizaje personalizado, sin llegar a poner en marcha un producto general.

Finalmente, hay iniciativas privadas, como el proyecto de Sony Global Education, para crear bases de datos centralizadas de credenciales y logros que den soporte a una población de estudiantes y trabajadores con mayor movilidad a nivel internacional, que necesitan un sistema que no esté sujeto a pérdidas o a fraude. [22].

En cualquier caso, aún muchas instituciones educativas ven la tecnología del BC como algo experimental, con la que podrían perder cierto control (centralizado) sobre sus registros y datos. Además, surgen problemas relativos a la privacidad y el acceso a ciertos datos sensibles, lo que unido a la complejidad actual de la tecnología y a la falta de estándares reales limita la posible entrada de la tecnología en el ámbito educativo.

En este trabajo se propone un sistema de certificación basado en BC, a modo de proyecto piloto, para gestionar la información sobre los logros, competencias y habilidades obtenidas por estudiantes durante el desarrollo de una asignatura.

## 3 Prototipo desarrollado

En esta sección se lleva a cabo una recopilación de las plataformas y herramientas de desarrollo principales para desplegar redes BC, ya sean públicas o privadas (subsección 3.1).

A continuación, se detalla la arquitectura y funcionamiento general de Hyperledger Fabric como software de desarrollo para implementar la red BC en este trabajo (subsección 3.2).

Finalmente, se detalla la arquitectura del prototipo propuesto en este trabajo, así como el proceso de creación, configuración y puesta en marcha del mismo (subsección 3.3).

### 3.1 Plataformas y herramientas de desarrollo

Mientras que algunos grupos de la industria trabajan en la estandarización de versiones de software de BC, también hay muchas empresas que están trabajando en sus propias versiones de esta tecnología.

Así, existen diversas plataformas con diferente nivel de aceptación entre los investigadores y desarrolladores [23]:

- El Proyecto Hyperledger de la Fundación Linux es una colaboración de código abierto entre diversas empresas líderes de la industria en los campos de las finanzas, la banca, e IoT en un esfuerzo por obtener un estándar. La principal herramienta del proyecto es Hyperledger Fabric.
- Ethereum es una plataforma abierta que permite la construcción, despliegue y uso de aplicaciones descentralizadas basadas en BC.
- Multichain es una plataforma para la creación e implementación de BC privados, ya sea dentro o entre organizaciones.
- IBM Bluemix Blockchain es parte del catálogo de servicios de Bluemix. Está construido sobre HyperLedger.
- Azure BaaS (Blockchain as a Service): parte del catálogo de servicios de Microsoft Azure Cloud. Ofrece diversas opciones, desde la creación de prototipos de desarrollo rápido hasta aplicaciones empresariales.

Cualquiera de estas plataformas/herramientas permiten desarrollar, bien redes BC públicas que pueden ser gestionadas de forma autónoma para intercambiar información entre los nodos sin necesidad de una administración centralizada, o bien para crear una red BC privada. Esto último permite a las empresas crear y administrar sus propias redes transaccionales que pueden utilizarse entre empresas socias o exclusivamente dentro de una misma empresa.

### 3.2 Arquitectura de Hyperledger Fabric

En nuestro prototipo proponemos usar Hyperledger Fabric como herramienta de desarrollo de la BC, ya que su arquitectura ofrece las siguientes ventajas:

- Flexibilidad en el desarrollo de las aplicaciones de cadena de bloques, de forma que diferentes conjuntos de nodos pueden responder a diferentes peticiones, lo que mejora la robustez de la red ante la posible caída de algún nodo.
- Escalabilidad en la ejecución de peticiones de actualización o de obtención de datos en la cadena.

- Confidencialidad en el tratamiento de datos y ejecución de transacciones con requisitos de confidencialidad determinados.
- Modularidad de la arquitectura, lo que permite integrar diferentes implementaciones de consenso.

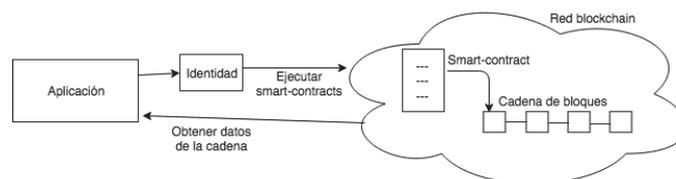
La arquitectura del sistema de Hyperledger Fabric consiste en un sistema distribuido entre muchos nodos que se comunican entre sí, y en los que se ejecutan aplicaciones, se almacenan datos y se ejecutan transacciones (asociadas a smart-contracts o contratos inteligentes).

El funcionamiento de un BC desarrollado sobre Hyperledger Fabric se basa en la ejecución de aplicaciones y smart-contracts para actualizar un almacén de datos (ledger o libro mayor), usando certificados digitales (emitidos por una autoridad certificadora), que permiten a las aplicaciones interactuar con la red BC.

En el prototipo desarrollado se hace uso del sistema de desarrollo de Hyperledger Fabric para ejecutar smart-contracts que realizan peticiones al almacén de datos, concretamente peticiones de información (lectura) y de actualización (escritura).

Las herramientas incluidas en la plataforma facilitan las labores iniciales de configuración del BC: crear la cadena de bloques inicial, lanzar el contenedor de smart-contracts, gestionar las autoridades de certificación, y distribuirlo todo entre los nodos de la red.

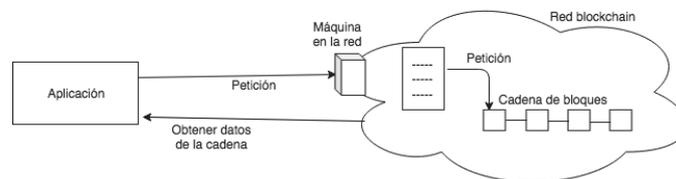
De esta forma, cada nodo en la red BC aloja una copia del almacén de datos junto con la aplicación que permite hacer consultas al mismo invocando smart-contracts que consultan el valor más reciente del almacén de datos y lo devuelven al usuario a través de la aplicación. La Figura 2 muestra el funcionamiento básico así como la estructura de una red blockchain.



**Fig. 2.** Estructura básica de una red blockchain.

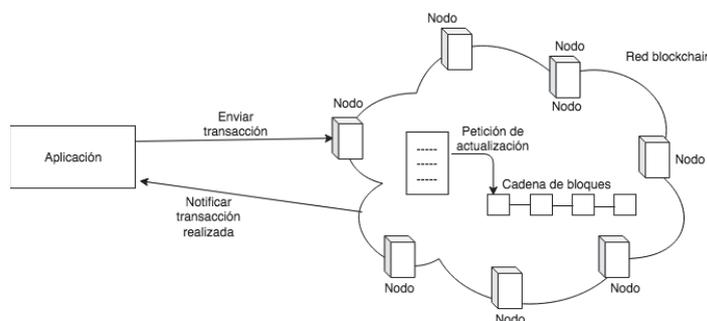
Las consultas más comunes en una red BC se utilizan para extraer (leer) los valores actuales de los datos en el almacén, su estado actual, representado como un conjunto de pares clave-valor. Este tipo de consultas se denomina "transacción", en las que se ejecuta un contrato inteligente para consultar cierto dato en la copia del libro mayor del nodo al que se ha realizado la consulta, que devuelve el resultado a la aplicación. En la Figura 3 se muestra el proceso seguido en la ejecución de una petición de lectura de datos a la red blockchain.

Por otro lado, para actualizar la BC (operaciones de escritura para añadir o modificar un dato), la aplicación envía una transacción a la red BC y una



**Fig. 3.** Proceso seguido en la ejecución de una petición de datos a la red blockchain.

vez validada, la aplicación recibe una notificación de que la transacción se ha realizado correctamente, quedando los datos actualizados. La Figura 4 muestra el proceso seguido en la ejecución de una transacción de actualización de los datos de la red blockchain.



**Fig. 4.** Proceso seguido en la ejecución de una transacción de actualización de los datos de la red blockchain.

### 3.3 Configuración y despliegue del sistema propuesto

La arquitectura del blockchain desarrollado se basa en implementar un "smart-contract" llamado *edu Smart Contract* (*eduSC*) que incluye varias funciones (ver la Tabla 1).

En el momento de instanciarlo en la red blockchain se ejecuta el método "Init", que crea las estructuras de datos y llama a la función "initLedger" para inicializar el almacén de datos.

Por otro lado, el método "Invoke" se ejecuta como resultado de una solicitud al smart-contract; recibe además el nombre de la función específica a ejecutar junto con los argumentos necesarios, y ejecuta la función especificada.

El resto de funciones incluidas en el smart-contract *eduSC* permiten crear un nuevo registro en el almacén de datos (función "create"), solicitar la información relativa a un registro (función "queryOne"), o todos los registros a la vez (función "queryAll").

Cada registro almacenado en la red blockchain consta de los siguientes campos (cadenas de caracteres): tipo, nombre completo, curso, nota de teoría, nota de prácticas.

Función	Descripción
Init	Crea las estructuras de datos básicas.
initLedger	Inicializa el almacén de datos al inicio.
Invoke	Recibe las solicitudes al smart-contract y ejecuta la función que se le especifica.
create	Crea un nuevo registro en el almacén de datos.
queryAll	Solicita la información relativa a un registro.
queryOne	Solicita todos los registros a la vez.

**Table 1.** Funciones incluidas en el smart-contract *eduSC*.

El despliegue de la red blockchain descrita se ha realizado en un sistema Ubuntu Server 16.04.6 64bits, siguiendo las instrucciones de instalación y configuración de Hyperledger Fabric dadas en la documentación de su página <sup>1</sup> en GitHub.

Una vez la red blockchain está en ejecución en los nodos (ver Figura 5), quedan habilitadas varias funciones de acceso mediante llamadas al smart-contract *eduSC*.

Para ello se han desarrollado varios programas en node.js<sup>2</sup> que se pueden utilizar desde la shell de Linux. Sin embargo, y para facilitar el acceso a la plataforma blockchain y a los datos almacenados en ella se ha habilitado un servicio web creado en node.js (ver Figura 6).

De esta forma, a través de un navegador web se puede acceder a la plataforma blockchain para solicitar la información de un registro específico (ver Figura 7) o a todos los registros almacenados (ver Figura 8).

```

pedro ~/fabric/fabric-samples/edu $ ./startFabric.sh node
# don't rewrite paths for Windows Git Bash users
export MSYS_NO_PATHCONV=1

docker-compose -f docker-compose.yml down
Removing network net_basic
WARNING: Network net_basic not found.

docker-compose -f docker-compose.yml up -d ca.example.com orderer.example.com
peer0.org1.example.com couchdb
Creating network "net_basic" with the default driver
Creating orderer.example.com ... done
Creating couchdb ... done
Creating ca.example.com ... done
Creating peer0.org1.example.com ... done

# wait for Hyperledger Fabric to start
# incase of errors when running later commands, issue export FABRIC_START_TIMEOUT=
<larger number>
export FABRIC_START_TIMEOUT=10
#echo ${FABRIC_START_TIMEOUT}
sleep ${FABRIC_START_TIMEOUT}

# Create the channel
docker exec -e "CORE_PEER_LOCALMSPID=Org1MSP" -e "CORE_PEER_MSPCONFIGPATH=/etc
/hyperledger/msp/users/Admin@org1.example.com/msp" peer0.org1.example.com peer
channel create -o orderer.example.com:7050 -c mychannel -f /etc/hyperledger
/configtx/channel.tx
2019-03-02 18:54:38.659 UTC [channelCmd] InitCmdFactory -> INFO 001 Endorser and
orderer connections initialized
2019-03-02 18:54:38.777 UTC [channelCmd] InitCmdFactory -> INFO 002 Endorser and
orderer connections initialized
2019-03-02 18:54:38.997 UTC [main] main -> INFO 003 Exiting....
# Join peer0.org1.example.com to the channel.

```

**Fig. 5.** Creación de la red blockchain.

<sup>1</sup> <https://hyperledger.github.io/composer/latest/installing/installing-prereqs.html>

<sup>2</sup> <https://nodejs.org>

```

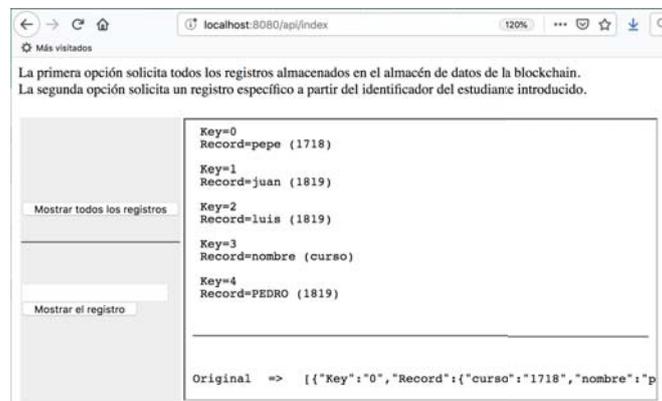
pedro ~/fabric/fabric-samples/edu $ node server.js
Store path: /home/pedro/fabric/fabric-samples/edu/hfc-key-store
Running on http://localhost:8080
Successfully loaded user1 from persistence
Requesting the item # 3
Query has completed, checking results
Response is
{"curso":"curso","nombre":"nombre","practicas":"practicas","teoria":"teoria","tipo":"estudiante"}

Successfully loaded user1 from persistence
Query has completed, checking results
Response is [{"Key":"0","Record":
{"curso":"1718","nombre":"pepe","practicas":"3","teoria":"6","tipo":"estudiante"}},
{"Key":"1","Record":
{"curso":"1819","nombre":"juan","practicas":"4","teoria":"5","tipo":"estudiante"}},
{"Key":"2","Record":
{"curso":"1819","nombre":"luis","practicas":"2","teoria":"3","tipo":"estudiante"}},
{"Key":"3","Record":
{"curso":"curso","nombre":"nombre","practicas":"practicas","teoria":"teoria","tipo":"estudiante"}}, {"Key":"4","Record":
{"curso":"1819","nombre":"PEDRO","practicas":"P3.0","teoria":"T3.0","tipo":"estudiante"}}]
    
```

Fig. 6. Ejecución del servicio web creado en node.js para facilitar el acceso a la plataforma blockchain.



Fig. 7. Acceso al servicio web habilitado para solicitar la información de un registro almacenado en la plataforma blockchain.



**Fig. 8.** Acceso al servicio web habilitado para solicitar todos los registros almacenados en la plataforma blockchain.

## 4 Conclusiones

Aún cuando BC sigue siendo una tecnología en pleno desarrollo, desde hace un tiempo se aprecian beneficios en aplicaciones reales. Específicamente en el ámbito educativo esta tecnología tendrá aplicaciones en el mundo de la educación a nivel individual, institucional, grupal, nacional e internacional. De hecho puede llegar a ser la base sobre la que desarrollar herramientas que ayuden a resolver las limitaciones de los sistemas de acreditación. Así, tener un sistema sencillo que registre la historia académica de un estudiante, antes y durante su vida laboral, no sólo puede ayudar a combatir la falsedad en los CVs, sino que también puede servir para potenciar el aprendizaje a medida.

En este sentido, en este trabajo se lleva a cabo una revisión de los diferentes usos de una tecnología prometedora como es BC en el ámbito educativo. Además, se propone una arquitectura de datos a modo de repositorio digital en el que almacenar diferentes registros del historial académico de los estudiantes a lo largo de su vida formativa o laboral.

BC facilitará la creación de registros digitales seguros y verificables de las cualificaciones de los estudiantes, obtenidas durante la carrera, así como de la experiencia y las competencias sociales adquiridas a lo largo de su vida.

Además, mediante el uso de contratos inteligentes, las aplicaciones distribuidas permitirán a los estudiantes tener un mayor control sobre su educación individual, ofreciendo un acceso flexible a recursos, contenidos y cursos en función de los éxitos o fracasos previos, así como a sus logros.

Como trabajo futuro planteamos ampliar el prototipo desarrollado con dos objetivos: dar cabida a todo el curriculum del estudiante, y no solo a los logros realizados en una asignatura, y gestionarlo mediante "smart contracts" que se resolverían una vez el estudiante adquiera ciertos conocimientos o competencias.

## Acknowledgements

Este trabajo ha sido financiado en parte por el Ministerio español de Economía y Competitividad en el marco del proyecto TIN2017-85727-C4-2-P (UGR-DeepBio).

## References

1. Pilkington, M.: Blockchain technology: Principles and applications. Research Handbook on Digital Transformations, edited by F. Xavier Olleros and Majlinda Zhegu. Edward Elgar. Available at SSRN: <https://ssrn.com/abstract=2662660> (2016)
2. The promise of blockchain is a world without middlemen. Harvard Business Review (6/3/2017). <https://hbr.org/2017/03/the-promise-of-blockchain-is-a-world-without-middlemen>, accessed: 2019-02-21
3. Dwyer, G.P.: The economics of bitcoin and similar private digital currencies. Journal of Financial Stability 17, 81 – 91 (2015), <http://www.sciencedirect.com/science/article/pii/S1572308914001259>, special Issue: Instead of the Fed: Past and Present Alternatives to the Federal Reserve System
4. Back, A., Corallo, M., Dashjr, L., Friedenback, M., Maxwell, G., Miller, A., Poelstra, A., Timon, J., Wuille, P.: Enabling blockchain innovations with pegged sidechains. Blockstream 2014. <http://www.blockstream.com/sidechains.pdf>, accessed: 2019-02-21
5. Wright, A., De Filippi, P.: Decentralized blockchain technology and the rise of lex cryptographia. SSRN Electronic Journal (01 2015)
6. Bartolomé-Pina, A., Bellver-Torlà, C., Castañeda-Quintero, L., Adell-Segura, J.: Blockchain en educación: Introducción y crítica al estado de la cuestión. EDUTEC. Revista Electrónica de Tecnología Educativa, número 61, pp. 1-14. ISSN 1135-9250 (2017)
7. Mearian, L.: Blockchain: The complete guide. Computerworld Jan 29, 2019. <https://www.computerworld.com/article/3191077/security/blockchain-the-complete-guide.html>, accessed: 2019-02-21
8. Blockchain and the internet of things: the iot blockchain opportunity and challenge. i-scoop, February 2018. <https://www.i-scoop.eu/blockchain-distributed-ledger-technology/blockchain-iot/>, accessed: 2019-02-21
9. Arnold, M.: Visa invita a las entidades a probar su nuevo sistema de pagos bancarios basado en la tecnología del ‘bitcoin’. Expansion, Sept 2016. <http://www.expansion.com/economia-digital/innovacion/2016/09/15/57d190a322601d456d8b45cb.html>, accessed: 2019-02-22
10. Allison, I.: Bank of england: Central banks looking at ‘hybrid systems’ using bitcoin’s blockchain technology. International Business Time, Jul 2017. <https://www.ibtimes.co.uk/bank-england-central-banks-looking-hybrid-systems-using-bitcoins-blockchain-technology-1511195>, accessed: 2019-02-22
11. Morales, T.: Cadena de bloques. cómo vender 25 toneladas de atún en 140 minutos gracias a blockchain. Retina - El Pais Economía, Noviembre 2017. [https://retina.elpais.com/retina/2017/11/27/innovacion/1511764546\\_361146.html](https://retina.elpais.com/retina/2017/11/27/innovacion/1511764546_361146.html), accessed: 2019-02-22

12. Behrens, A.: Berkleeice's open music initiative to host music and rights management blockchain events. Blockchain News, February 2018. <https://www.the-blockchain.com/2018/02/09/berkleeeices-open-music-initiative-host-music-rights-management-blockchain-events/>, accessed: 2019-02-22
13. Goswami, D.: Blockchain and the internet of things: the iot blockchain opportunity and challenge. Blog of Government Innovators Network, Harvard Kennedy School, Ash Center for Democratic Governance and Innovation, June 2016. <https://www.innovations.harvard.edu/blog/unchaining-blockchain-ultimate-transparency-tool>, accessed: 2019-02-22
14. Pérez, I.: Así puede ser el futuro del blockchain en la salud según las grandes empresas. Criptonoticias, Agosto 2016. <https://www.criptonoticias.com/aplicaciones/futuro-blockchain-salud-grandes-empresas/>, accessed: 2019-02-22
15. Go, C.: Blockchain and identity technologies. Medium, March 2018. <https://medium.com/@cathyguo.up/blockchain-and-identity-technologies-ac89df29dbf2>, accessed: 2019-02-22
16. Grech, A., Camilleri, A.: Blockchain in education. A. Inamorato dos Santos Ed. EUR 28778 EN; ISBN 978-92-79-73497-7 (2017)
17. Bartolomé, A., Lindín, C.: Posibilidades del blockchain en educación - blockchain possibilities in education. Education in the Knowledge Society (EKS) 19(4), 81–93 (2018)
18. Schmidt, P.: Blockcerts — an open infrastructure for academic credentials on the blockchain. GitHub 2016. <https://github.com/BlockTechCert/BTCert>, accessed: 2019-02-23
19. Koulaidis, V.: La tecnología blockchain como herramienta pedagógica. En A. Bartolomé, J.M. Moral-Ferrer: Blockchain en Educación. Cadenas rompiendo moldes, pp. 81-86. Barcelona: LMI. Colección Transmedia XXI (2018)
20. Sharples, M., Domingue, J.: The blockchain and kudos: A distributed system for educational record, reputation and reward. In: Verbert, K., Sharples, M., Klobočar, T. (eds.) Adaptive and Adaptable Learning. pp. 490–496. Springer International Publishing, Cham (2016)
21. KMI: Researching the potential of blockchains badges. Knowledge Media Institute - Open University, 2018. <https://blockchain.open.ac.uk/>, accessed: 2019-02-23
22. Clark, D.: 10 ways blockchain could be used in education. OEB Insights, Sept 2016. <https://oeb.global/oeb-insights/10-ways-blockchain-could-be-used-in-education/>, accessed: 2019-02-23
23. Purkayastha, S.: Eight blockchain platforms for rapid prototyping. radiostud, Sept 2018. <https://radiostud.io/eight-blockchain-platforms-comparison/>, accessed: 2019-02-24

## Estudio del paradigma Internet de las Cosas y tecnologías inalámbricas como arquitectura base de captación de datos en las fases de ejecución y mantenimiento de los proyectos de construcción

María Martínez-Rojas<sup>1</sup>, Gregorio Corpas<sup>2</sup>, Autilia Vitiello<sup>3</sup>, Giovanni Acampora<sup>3</sup>, and José Manuel Soto-Hidalgo<sup>2</sup>

<sup>1</sup> Universidad de Málaga mmrojas@uma.es

<sup>2</sup> Universidad de Córdoba {i02coprg, jmsoto}@uco.es

<sup>3</sup> Universidad Federico II, Nápoles  
{autilia.vitiello, giovanni.acampora}@unina.it

**Abstract.** El paradigma de Internet de las Cosas está afianzándose como tecnología para mejorar la adquisición de datos y gestión de la información en el ámbito de la construcción. En general se consolida como tecnología emergente en todas las fases del ciclo de vida de los proyectos y en concreto, por su aplicabilidad y relevancia, en las fases de ejecución y mantenimiento. El sistema de captación y compartición de datos mediante dispositivos interconectados permite la adquisición de éstos en tiempo real que, posteriormente, pueden ser gestionados por sistemas para ofrecer un soporte inteligente en el proceso de toma de decisiones. El objetivo de este trabajo es realizar un estudio sobre las propuestas existentes en la literatura que utilicen estos enfoques y explorar las necesidades en las fases de ejecución y mantenimiento del ciclo de vida de los proyectos de construcción.

**Keywords:** Internet de las Cosas · Ciclo de vida proyectos Construcción · Fase de ejecución · Fase de mantenimiento

### 1 Introducción

Las Tecnologías de la Información y la Comunicación (TIC) han tardado en adaptarse a la industria de la construcción por diversos motivos aunque principalmente todos ellos están relacionados con las características propias del sector [7]. Es decir, la temporalidad de los proyectos [38], el carácter tradicional [32], la simultaneidad de trabajos [2] o la singularidad de cada proyecto de construcción [30], etc. son los principales elementos limitantes de la integración de las TIC en este sector. Sin embargo, en la última década se ha experimentado una transformación importante en relación al uso de las nuevas tecnologías para dar soporte a la gestión de los proyectos de construcción [11, 1, 24].

El ciclo de vida de los proyectos de construcción contempla fases de diversa índole: (i) identificación de requisitos, (ii) planificación del proyecto, (iii) diseño (iv) ejecución, (v) mantenimiento y (vi) demolición. Dentro de estas fases, la que ocupa el periodo de

tiempo más largo del ciclo de vida de un edificio es la fase de mantenimiento [42]. Desde la perspectiva de la gestión, el objetivo en cada una de estas fases es diferente y requiere de diferentes herramientas. Por ejemplo, en las primeras etapas del proyecto es esencial la coordinación y colaboración entre las distintas partes involucradas, mientras que en la fase de ejecución el objetivo principal se centra en el control de costes, plazos, calidad y seguridad [34]. En la fase de mantenimiento es fundamental el control de diversos factores como la calidad del aire, el rendimiento energético o el propio mantenimiento del edificio, entre otros. Sin embargo, todas estas fases comparten la necesidad de un adecuado proceso de adquisición de datos así como técnicas que permitan gestionar dichos datos adecuadamente. En este sentido, la diversidad de sensores existentes así como la comunicación entre éstos son los que facilitan la captura de datos mientras que las diferentes técnicas de inteligencia artificial son las que posibilitan extraer información a partir de éstos para gestionar los datos y poder dar un soporte inteligente.

Este aspecto se ha visto potenciado por el paradigma conocido como Internet de las Cosas (IoT, por sus siglas en inglés) el cual describe un sistema de dispositivos de computación interrelacionados e identificados unívocamente que tienen la capacidad de transferir datos a través de una red, sin requerir de interacciones humanas [17]. Recientemente, este paradigma está siendo considerado por los investigadores en el ámbito de la construcción, en especial, en la fase de ejecución y mantenimiento mediante la sensorización de espacios físicos [39]. Internet de las Cosas aporta, entre otros aspectos, destacables beneficios en el ciclo de vida de la construcción, distinguiendo entre otros la reducción de costes, un incremento de la seguridad durante la ejecución de las obras, diseños más inteligentes y conocimiento sobre el entorno en tiempo real [23].

Por consiguiente, la integración de tecnologías de captura de datos en fases del ciclo de vida de un proyecto de construcción con Internet de las Cosas plantea un reto interesante que puede proporcionar importantes beneficios para la toma de decisiones y gestión de las fases del ciclo de vida de un proyecto de construcción en tiempo real. Estos aspectos han motivado el presente trabajo, el cual radica en realizar un estudio preliminar de las propuestas existentes en la literatura sobre Internet de las Cosas en relación al uso de sensores para facilitar el seguimiento y toma de decisiones en las distintas fases del ciclo de vida de un proyecto de construcción, en concreto en la fase de ejecución y mantenimiento.

El trabajo se estructura como se detalla a continuación. En la sección 2 se presentan, de modo general, las distintas fases del ciclo de vida de un proyecto de construcción junto con las posibles necesidades tecnológicas de cada fase. En la sección 3 se presentan las tecnologías potenciales, desde el punto de vista de Internet de las Cosas mientras que en la sección 4 se realiza un análisis de aquellos trabajos más relevantes existentes en la literatura que integran dichas tecnologías aplicadas a las distintas fases. Finalmente, en la sección 5 se reflejan las principales conclusiones y futuros trabajos.

## **2 Fases del ciclo de vida en el ámbito de la construcción**

La aparición de nuevos paradigmas en el sector de la construcción puede proporcionar ventajas importantes para diferentes tareas importantes en el ciclo de vida de los proyectos de construcción. En concreto, nos centraremos en la fase de ejecución y manten-

imiento, que son las que pueden obtener más beneficios de estas tecnologías. Por tanto, en esta sección, se analizan y detallan las diversas tareas para, posteriormente, analizar las tecnologías que se emplean en la actualidad para integrar y gestionar los datos en tiempo real.

1. **Fase de ejecución.** Durante la ejecución de los proyectos son de interés las siguientes tareas:
  - Monitorización de recursos y materiales. La monitorización y el seguimiento de los equipos y materiales puede aportar importantes beneficios para la ejecución del proceso, consiguiendo una reducción en los plazos así como en los costes de la obra [37].
  - Comunicación y colaboración. La gestión durante el proceso de construcción podría mejorarse utilizando la comunicación en tiempo real y la colaboración con dispositivos que integren tecnologías BIM e IoT [12].
  - Rendimiento del proceso de construcción y monitorización del progreso. Este aspecto puede aprovechar estas tecnologías en varios sentidos. En primer lugar, para conocer el estado actual del proceso de construcción en tiempo real. Por otro lado, para sincronizar la planificación de la obra en base a la monitorización del proceso [33].
  - Gestión de la Seguridad y Salud. Esta tarea es fundamental en las obras de construcción ya que la accidentalidad en este sector es muy elevado en comparación al de otros sectores. En este sentido, el uso de tecnologías que permitan monitorizar o detectar posibles riesgos en tiempo real puede resultar de mucho interés para evitar o reducir el número de accidentes [29].
2. **Fase de Mantenimiento.** En la fase de mantenimiento del edificio resultan interesantes las siguientes tareas:
  - Gestión de la ocupación de espacios. La monitorización y control en tiempo real de los espacios de un edificio puede resultar útil en varios sentidos. Por ejemplo, para adaptar las condiciones climáticas del espacio tales como ventilación, calefacción o refrigeración [18].
  - Mantenimiento. Permite el acceso a datos en tiempo real del comportamiento del edificio y la verificación de la mantenibilidad [16].
  - Gestión de la energía. Permite controlar el consumo de energía en tiempo real así como adaptarlo a las necesidades y características reales de cada edificio en cada momento del día. Gracias a este control se pueden disminuir los costes, disminuir el consumo de energía o crear valor con la información en tiempo real [35].
  - Respuesta a desastres y emergencias. Conocer cualquier alteración en las condiciones normales del edificio puede resultar de interés para detectar cualquier situación de emergencia, como por ejemplo, incendios, accidentes, etc. Además, puede ayudar a la hora de gestionar la evacuación mediante alternativas flexibles y adaptadas a la situación del edificio en el momento del incidente [10].

Una vez detalladas las tareas fundamentales que pueden beneficiarse de estas tecnologías, la siguiente sección detalla las principales tecnologías utilizadas y reflejadas en la literatura.

### 3 Tecnologías para la adquisición de datos y toma de decisiones inteligente en el ciclo de vida de proyectos de construcción

En esta sección se detallan, algunas de las tecnologías que se han identificado en la literatura como relevantes para la adquisición de datos durante el ciclo de vida de un proyecto de construcción y partícipes en sistemas IoT.

Por un lado, toda propuesta basada en el paradigma de Internet de las Cosas requiere de una infraestructura que proporcione soporte al sistema de dispositivos de computación interrelacionados. Para ello, éstos tienen que tener la capacidad de transferir datos de manera automática e identificarse de forma unívoca. La Figura 1 ilustra, de manera general, un esquema común de infraestructura base con los distintos elementos que forman parte de ésta. En el nivel inferior se encuentran los sensores, como elementos de captación de datos y medición de condiciones físicas. En el nivel intermedio se encuentran los elementos de transmisión, en este caso redes e internet y en el nivel superior, elementos de cómputo.

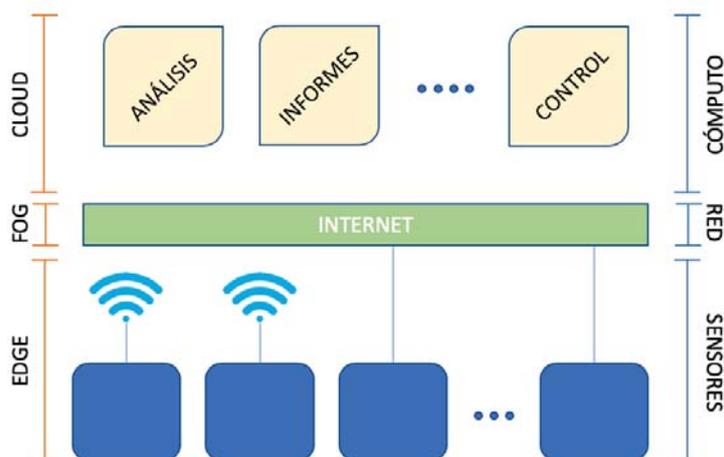


Fig. 1. Arquitectura IoT

Esta infraestructura permite ubicuidad de los dispositivos y/o elementos que la forman. En general, para gestionar la comunicación entre los elementos de la red, se suele hacer uso de protocolos de comunicación. Ejemplo de protocolo, por sus características de seguridad y interoperabilidad y generalidad es el protocolo MQTT (Message Queuing Telemetry Transport) [13], el cuál simplifica las tareas de envío/recepción de mensajes, asegurando ambos mediante unas métricas de QoS (calidad de servicio). Los principales actores involucrados en la comunicación son los siguientes:

- Broker: Servicio encargado de recibir y redirigir los mensajes a los demás actores involucrados.

- Publicador: Servicio cliente que envía mensajes a un topic.
- Subscriptor: Servicio cliente que recibe mensajes de uno o varios topics.

La Figura 2 ilustra un esquema de comunicación del protocolo MQTT.

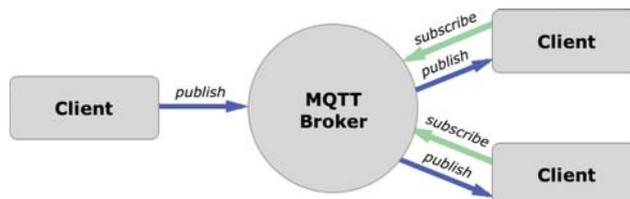


Fig. 2. Comunicación en el protocolo MQTT

En este sentido, y considerando la infraestructura IoT como base, podemos considerar como tecnologías de captación de datos: sensores, RFID y GPS. Los sensores pueden medir las condiciones físicas del mundo, como la posición, la ocupación, la aceleración, la velocidad, el movimiento, la temperatura, etc. Como se detallará a continuación, se han aplicado ampliamente para monitorear parámetros que incluyen el estado ambiental, el comportamiento de los ocupantes y el uso de energía en el interior de los edificios [22]. Mientras que la Identificación por Radiofrecuencia (RFID, por sus siglas en inglés) puede, mediante campos electromagnéticos, identificar y rastrear automáticamente las etiquetas adheridas a los objetos. Finalmente, el Sistema de Posicionamiento Global (GPS, por sus siglas en inglés) permite determinar la posición de personas u objetos con gran precisión.

#### 4 Análisis de tecnologías utilizadas en el ciclo de vida de la construcción

En esta sección se detallan algunas de las propuestas encontradas en la literatura dentro del marco del objetivo de este trabajo y se exploran las tecnologías más utilizadas. También, se detallan las fases del ciclo de vida, en concreto ejecución y mantenimiento, así como el objetivo específico para el que han sido diseñadas (gestión de la seguridad y salud, control, seguimiento del proceso o equipos/materiales).

Respecto a las tecnologías más utilizadas, dentro de los elementos de un sistema IoT, se consideran principalmente aquellas basadas en comunicación inalámbrica e internet así como distintos sensores, identificación por radio frecuencia (RFID) o sistemas de posicionamiento global (GPS). Los sensores se pueden aplicar por ejemplo para monitorear parámetros como el estado ambiental, el comportamiento de los ocupantes y el uso de energía en el interior de los edificios. La tecnología RFID se puede utilizar durante la fase de ejecución del edificio para tareas de seguimiento de material y para la gestión de la seguridad y salud, por ejemplo. Mientras que el GPS se puede utilizar para la localización de materiales y personas en tiempo real.

La Tabla 1 organiza varias propuestas de autores según lo indicado anteriormente, en primer lugar, se distinguen las tecnologías (sensores, RFID y GPS) y, en segundo lugar, se distingue si el objetivo de la propuesta se centra en la fase de ejecución o de mantenimiento. Dentro de la fase de ejecución, se han identificado tres actividades importantes en la gestión de los proyectos: seguimiento de materiales, control y gestión de la seguridad y salud. Tal y como se puede observar en la Tabla, la tecnología más utilizada es la de RFID, aunque la utilización de sensores está ganando mayor aceptación en los últimos años. En relación a las fases de la obra, la mayoría de las propuestas se centran en la fase de ejecución y, en concreto, para facilitar y mejorar la gestión de la seguridad durante el proceso de ejecución. Dentro de esta tarea, que es muy relevante en el sector de la construcción debido al alto índice de accidentalidad, las propuestas se centran principalmente en la identificación de riesgos y en alertar a los trabajadores para evitar los accidentes [8, 6, 36, 23]. De manera similar, el control de algunas actividades importantes como la planificación [9] o el estado de las estructuras de hormigón [16] han llamado la atención de la comunidad investigadora. En la fase de mantenimiento, las propuestas se centran en la gestión de la eficiencia energética [35], planificación en situaciones de emergencias [10] and estimación de ocupación para adaptar las condiciones climáticas [18].

Como se puede observar en la Tabla 1, las propuestas dedicadas a la gestión de la seguridad utilizan principalmente la tecnología de RFID, mientras que las propuesta centradas en la fase de mantenimiento hacen uso de sensores (temperatura, luz, humedad, etc). Todas las propuestas presentan tendencias del uso de estas tecnologías en las fases de los proyectos de construcción que, en conjunto con el despliegue bajo infraestructura IOT, plantean resultados prometedores.

## 5 Conclusiones

El ciclo de vida del edificio incluye la identificación de requisitos, la planificación del proyecto, el diseño y la ingeniería, la construcción, las operaciones y el mantenimiento y la demolición. Aunque la industria de la construcción siempre ha tenido un carácter muy tradicional en el que intervienen muchas empresas pequeñas y en el que la tecnología no ha estado muy presente, hoy en día nuevos enfoques como el de Internet de las Cosas están jugando un papel fundamental para mejorar el proceso de toma de decisiones. Sin embargo, se han encontrado algunas limitaciones. Por ejemplo, durante la fase de construcción la mayoría de las propuestas proponen un marco de trabajo específico y en ocasiones realizado con prototipos o escenarios. En este sentido, es necesario desarrollar propuestas en proyectos reales para explorar la escalabilidad y confiabilidad de éstas [26].

Además, cuestiones como el engorroso diseño del proceso, la conversión manual de datos, la confiabilidad de los datos recopilados por el sensor para el cálculo de mecanismos aún deben ser simplificados y abordados [33]. No obstante, algunos estudios de investigación han tenido éxito en caso de uso específico, como por ejemplo, la visualización, el funcionamiento de la grúa, el seguimiento de la ubicación y la advertencia de riesgo [45].

**Table 1.** Propuestas según tecnologías y fases del ciclo de vida de los proyectos

	Tecnologías		Fases			Mantenimiento
	Sensores	RFID GPS	Seguimiento	Ejecución materiales Control	Seguridad	
[20]		x		x		
[19]	x					x
[40]		x			x	
[8]		x			x	
[45]		x			x	
[47]				x		
[37]		x	x			
[9]		x		x		
[46]		x			x	
[5]		x			x	
[31]		x			x	
[25]		x			x	
[41]			x		x	
[16]	x			x		
[21]						x
[35]			x			x
[36]		x			x	
[14]		x	x	x	x	
[44]			x		x	
[18]	x					x
[27]	x				x	
[43]	x				x	
[23]		x			x	
[29]				x		
[3]	x				x	
[4]	x				x	
[28]		x		x		
[15]		x			x	
[10]			x			x

## References

1. Ahuja, V., Yang, J., Shankar, R.: Benefits of collaborative ict adoption for building project management. *Construction Innovation* **9**(3), 323–340 (2009)
2. Ansah, R.H., Sorooshian, S., Mustafa, S.B.: Lean construction: an effective approach for project management. *ARNP Journal of Engineering and Applied Sciences* **11**(3), 1607–1612 (2016)
3. Antwi-Afari, M.F., Li, H.: Fall risk assessment of construction workers based on biomechanical gait stability parameters using wearable insole pressure system. *Advanced Engineering Informatics* **38**, 683 – 694 (2018)
4. Awolusi, I., Marks, E., Hallowell, M.: Wearable technology for personalized construction safety monitoring and trending: Review of applicable devices. *Automation in Construction* **85**, 96 – 106 (2018)

5. Barro-Torres, S., Fernández-Caramés, T.M., Pérez-Iglesias, H.J., Escudero, C.J.: Real-time personal protective equipment monitoring system. *Computer Communications* **36**(1), 42 – 50 (2012)
6. Carbonari, A., Giretti, A., Naticchia, B.: A proactive system for real-time safety management in construction sites. *Automation in Construction* **20**(6), 686 – 698 (2011), selected papers from the 26th ISARC 2009
7. Cárcel-Carrasco, F.J., Peñalvo-López, E.: El uso de las tic para la formación en smart meters a los jefes de obra de edificación. un proyecto europeo. In: libro de resúmenes del 1er congreso internacional online del uso de las tic en la sociedad, la educación y la empresa. vol. 5, p. 43. *3Ciencias* (2016)
8. Chae, S., Yoshida, T.: Application of rfid technology to prevention of collision accident with heavy equipment. *Automation in Construction* **19**(3), 368 – 374 (2010), 25th International Symposium on Automation and Robotics in Construction
9. Cheng, M.Y., Chen, J.C.: Integrating barcode and gis for monitoring construction progress. *Automation in Construction* **11**(1), 23 – 33 (2002)
10. Chou, J.S., Cheng, M.Y., Hsieh, Y.M., Yang, I.T., Hsu, H.T.: Optimal path planning in real time for dynamic building fire rescue operations using wireless sensors and visual guidance. *Automation in Construction* **99**, 1 – 17 (2019)
11. Chung, J.K., Kumaraswamy, M.M., Palaneeswaran, E.: Improving megaproject briefing through enhanced collaboration with ict. *Automation in construction* **18**(7), 966–974 (2009)
12. Dave, B., Buda, A., Nurminen, A., Främbling, K.: A framework for integrating bim and iot through open standards. *Automation in Construction* **95**, 35–45 (2018)
13. Dizdarević, J., Carpio, F., Jukan, A., Masip-Bruin, X.: A survey of communication protocols for internet of things and related challenges of fog and cloud computing integration. *ACM Computing Surveys (CSUR)* **51**(6), 116 (2019)
14. Fang, Y., Cho, Y.K., Zhang, S., Perez, E.: Case study of bim and cloud-enabled real-time rfid indoor localization for construction management applications. *Journal of Construction Engineering and Management* **142**(7), 05016003 (2016)
15. de Gabriel, J.M.G., Fernández-Madrugal, J.A., López-Arquillos, A., Rubio-Romero, J.C.: Monitoring harness use in construction with ble beacons. *Measurement* **131**, 329 – 340 (2019)
16. Ham, S., Popovics, J.S.: Application of contactless ultrasound toward automated inspection of concrete structures. *Automation in Construction* **58**, 155 – 164 (2015)
17. Hassan, Q.: *Internet of Things A to Z: Technologies and Applications*. Wiley (2018)
18. Huang, Q., Mao, C.: Occupancy estimation in smart building using hybrid co2/light wireless sensor network. *Journal of Applied Sciences and Arts* **1**(2), 5 (2017)
19. Jang, W.S., Healy, W.M., Skibniewski, M.J.: Wireless sensor networks as part of a web-based building environmental monitoring system. *Automation in Construction* **17**(6), 729 – 736 (2008)
20. Jaselskis, E.J., El-Misalami, T.: Implementing radio frequency identification in the construction process. *Journal of Construction Engineering and Management* **129**(6), 680–688 (2003)
21. Jia, M., Srinivasan, R.S.: Occupant behavior modeling for smart buildings: A critical review of data acquisition technologies and modeling methodologies. In: 2015 Winter Simulation Conference (WSC). pp. 3345–3355 (Dec 2015)
22. Jia, M., Srinivasan, R.S.: Occupant behavior modeling for smart buildings: A critical review of data acquisition technologies and modeling methodologies. In: Proceedings of the 2015 Winter Simulation Conference. pp. 3345–3355. IEEE Press (2015)
23. Kanan, R., Elhassan, O., Bensalem, R.: An iot-based autonomous system for workers' safety in construction sites with real-time alarming, monitoring, and positioning strategies. *Automation in Construction* **88**, 73 – 86 (2018)

24. Kasim, N.: Ict implementation for materials management in construction projects: case studies. *Journal of Construction Engineering and Project Management* **1**(1), 31–36 (2011)
25. Kelm, A., Laußat, L., Meins-Becker, A., Platz, D., Khazaei, M.J., Costin, A.M., Helmus, M., Teizer, J.: Mobile passive radio frequency identification (rfid) portal for automated and rapid control of personal protective equipment (ppe) on construction sites. *Automation in Construction* **36**, 38 – 52 (2013)
26. Kim, K., Cho, Y., Zhang, S.: Integrating work sequences and temporary structures into safety planning: Automated scaffolding-related safety hazard identification and prevention in bim. *Automation in Construction* **70**, 128 – 142 (2016)
27. Lee, W., Lin, K.Y., Seto, E., Migliaccio, G.C.: Wearable sensors for monitoring on-duty and off-duty worker physiological status and activities in construction. *Automation in Construction* **83**, 341 – 353 (2017)
28. Li, C.Z., Xue, F., Li, X., Hong, J., Shen, G.Q.: An internet of things-enabled bim platform for on-site assembly services in prefabricated construction. *Automation in Construction* **89**, 146 – 161 (2018)
29. Louis, J., Dunston, P.S.: Integrating iot into operational workflows for real-time and automated decision-making in repetitive construction operations. *Automation in Construction* **94**, 317 – 327 (2018)
30. Lu, Y., Li, Y., Skibniewski, M., Wu, Z., Wang, R., Le, Y.: Information and communication technology applications in architecture, engineering, and construction organizations: A 15-year review. *Journal of Management in Engineering* **31**(1) (2014)
31. Marks, E.D., Teizer, J.: Method for testing proximity detection and alert technology for safe construction equipment operation. *Construction Management and Economics* **31**(6), 636–646 (2013)
32. Martínez-Rojas, M., Marín, N., Vila, M.A.: The role of information technologies to address data handling in construction project management. *Journal of Computing in Civil Engineering* **30**(4), 04015064 (2016)
33. Matthews, J., Love, P.E., Heinemann, S., Chandler, R., Rumsey, C., Olatunji, O.: Real time progress management: Re-engineering processes for cloud-based bim in construction. *Automation in Construction* **58**, 38 – 47 (2015)
34. M.Martínez-Rojas, N.Marín, C.Molina, M.Vila: Cost analysis in construction projects using fuzzy olap cubes. In: 2015 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE). pp. 1–8 (Aug 2015). <https://doi.org/10.1109/FUZZ-IEEE.2015.7338048>
35. Pan, J., Jain, R., Paul, S., Vu, T., Saifullah, A., Sha, M.: An internet of things framework for smart energy in buildings: designs, prototype, and experiments. *IEEE Internet of Things Journal* **2**(6), 527–537 (2015)
36. Park, J., Marks, E., Cho, Y.K., Suryanto, W.: Performance test of wireless technologies for personnel and equipment proximity sensing in work zones. *Journal of Construction Engineering and Management* **142**(1) (2016)
37. Sardroud, J.M.: Influence of rfid technology on automated management of construction materials and components. *Scientia Iranica* **19**(3), 381 – 392 (2012)
38. Shokri, S., Ahn, S., Lee, S., Haas, C.T., Haas, R.C.G.: Current status of interface management in construction: Drivers and effects of systematic interface management. *Journal of Construction Engineering and Management* **142**(2), 04015070 (2016). [https://doi.org/10.1061/\(ASCE\)CO.1943-7862.0001035](https://doi.org/10.1061/(ASCE)CO.1943-7862.0001035)
39. Solanki, A., Nayyar, A.: Green Building Management and Smart Automation. *Advances in Civil and Industrial Engineering*, IGI Global (2019)
40. Teizer, J., Allread, B.S., Fullerton, C.E., Hinze, J.: Autonomous pro-active real-time construction worker and equipment operator proximity safety alert system. *Automation in Construction* **19**(5), 630 – 640 (2010), *building Information Modeling and Collaborative Working Environments*

41. Teizer, J., Cheng, T.: Proximity hazard indicator for workers-on-foot near miss interactions with construction equipment and geo-referenced hazard areas. *Automation in Construction* **60**, 58 – 73 (2015)
42. Vähä, P., Heikkilä, T., Kilpeläinen, P., Järviluoma, M., Gambao, E.: Extending automation of building construction — survey on potential sensor technologies and robotic applications. *Automation in Construction* **36**, 168 – 178 (2013)
43. Valero, E., Sivanathan, A., Bosché, F., Abdel-Wahab, M.: Analysis of construction trade worker body motions using a wearable and wireless motion sensor network. *Automation in Construction* **83**, 48 – 55 (2017)
44. Wang, J., Razavi, S.N.: Low false alarm rate model for unsafe-proximity detection in construction. *Journal of Computing in Civil Engineering* **30**(2), 04015005 (2016)
45. Wu, W., Yang, H., Chew, D.A., Hua Yang, S., Gibb, A.G., Li, Q.: Towards an autonomous real-time tracking system of near-miss accidents on construction sites. *Automation in Construction* **19**(2), 134 – 141 (2010)
46. Yang, H., Chew, D.A., Wu, W., Zhou, Z., Li, Q.: Design and implementation of an identification system in construction site safety for proactive accident prevention. *Accident Analysis & Prevention* **48**, 193 – 203 (2012), intelligent Speed Adaptation + Construction Projects
47. Yang, J., Cheng, T., Teizer, J., Vela, P., Shi, Z.: A performance evaluation of vision and radio frequency tracking methods for interacting workforce. *Advanced Engineering Informatics* **25**(4), 736 – 747 (2011), special Section: Advances and Challenges in Computing in Civil and Building Engineering

## Teaching C/C ++ programming using a multimedia system with videos

José Galindo<sup>1</sup>, Patricia Galindo<sup>1</sup>, José María Rodríguez Corral<sup>2</sup>,

<sup>1</sup> Dpto. Lenguajes y Ciencias de la Computación, University of Málaga, Spain.

<sup>2</sup> Dpto. Dpto. Ingeniería Informática, University of Cádiz, Spain.

**Abstract.** The C programming language is widely used in computer and industrial engineering. Because of that, such programming language is also widely used as a language to teach programming to industrial engineering students. In Spain, many universities use this language compulsory in the first year, or even in higher courses. Our experience shows that learning computer programming in four months is an arduous task, but curricula require it. Such learning process is also tough by the fact that many students cannot attend classes regularly and, even if they attend, sometimes the class is no longer at the level they require. In this work we develop a series of files in "presentation" format (.ppsx) and videos that allow students to see several explanations about the most complicated programming C topics: functions, arrays, structures, strings, arrays with structures, etc. This multimedia material includes explanations (voice-over), and animations with examples. Students can watch and listen to the explanations whenever and wherever they want (tablet, PC, phone...). Surveys made to students reveal that it is also interesting for students who regularly attend classes, and they prefer to use this course material only at home, outside of regular classes.

**Keywords:** programming languages, multimedia teaching, autonomous learning, C/C++ language.

### 1 Introduction

It has been noted that students frequently experiment difficulties when learning traditional programming languages (such as C, C++ and Java) and abstract programming concepts [5][7][10].

In order to make the creation of new applications easier for people without programming skills, recent software developments include programming tools that hide much of the complexity of traditional languages. For example, a Visual Programming Language [3] could simplify many programming tasks being suitable also for learning purposes even in specific contexts, such as robot programming [13]. On the other hand, authoring tools allow a person to develop an application simply by linking together objects, such as a picture, a sound or a text paragraph. Authors can create useful and attractive graphics applications by simply describing the object's relationship among themselves and by sequencing them in an appropriate order. This

could be suitable for novice programmers [6], including those students in the first years whose university degrees are not directly related to the computer science discipline.

Basic programming concepts are easier to learn in a visual programming languages, but commercial and industrial applications are usually made using traditional languages. On the other hand, some kind of visual languages, such as Visual Basic or Visual C#, need to program with traditional language techniques in order to be efficiently and accurately used. The relevance of traditional languages is proved by the fact that these languages are very common in the universities, in many degrees such as computer sciences, telecommunications engineering, industrial engineering and so on. In fact, the C language is used, along with the assembly language, for programming microcontrollers, and the Arduino language [11] is based on C++. Likewise, Matlab (MATrix LABoratory) is a software very used in the field of scientific computing in universities, research institutes and industries around the world. It can be used for representation of graphics, mathematical computation, modeling and simulation, data analysis and processing and development of algorithms.

Particularly, in Spain there are a lot of subjects in first courses of different degrees in which their first goal is to learn to program using a traditional programming language (such as C, C++, java, Python, Visual C#, Matlab...). In addition, the majority of these subjects are taught in 15 weeks, at a rate of about four and a half hours of class per week. In this context, the students have to face the following main problems [4]:

- The high level of demand, especially in technical degrees, which are the majority.
- The scarce time to learn the theoretical concepts and mainly, to acquire the ability to put them into practice while programming with certain complexity: The exams include the most intricate problems seen in class, but only one or two weeks are devoted to such kind of problems.
- Numerous repeating students who do not attend all classes due to incompatibility of their schedules.
- Little practical teaching time to practice: Normally, students receive, each week, 3 hours of theoretical classes (on the blackboard) and 1.5 hours of practical classes. It is notorious that students must practice more outside university hours and teachers warn them to do so by providing some proposed exercise relationships (with some of them solved).
- Shortage of time to study and practice important concepts and exam exercises. The period of classes is followed by only 1 or 2 weeks before the exam.
- Students who already have certain knowledge have to wait until the rest of their classmates are at the same level, in order to advance together.
- Losing a class might mean being left behind or having to invest a lot of time and effort in catching up.
- To learn computer programming, students need to grasp valid basic concepts during their introductory classes as these forms a strong background for more advanced programming exercises. The more doubts the student has in previous concepts, the more effort he will have to invest in order to advance.

- Besides, a negative perception of the subject is added in many degrees excluding computer science. As beginners, the students believe that learning programming is cumbersome [14].

Moreover, teachers also have to face some problems although, of course, these inconveniences depend on different factors and the teaching context:

- High number of students per group, which prevents giving all classes in a practical way.
- Heterogeneous groups, where some start from scratch but others already have a good level. This aspect, together with the previous point, prevents giving proper treatment to each student according to his or her level.
- Need to repeat the same class over and over again: This may be because the same teacher educates several groups, but it is also referred to repetitions from one quarter or year to the following.
- The teaching quality might vary depending on personal factors of the teacher such as fatigue, noise, multiple repetitions, and distractors, among others.

Of course, it is not easy to solve all these problems for teachers and students in a comfortable and simple way. Some teachers have opted to record their classes on video and uploaded them on the Internet so that their students have access to them 24 hours. Only this solution already solves some of the problems, although it poses other challenges, such as the appearance of the blackboard, which should look good, or the quality of the video (regarding environmental noise, pronunciation...). All these problems are solved by professionally prepared videos or, as we have opted in our case, by preparing different classes in Microsoft Power Point format files (.ppsx files). Among other advantages, the quality of image and sound is excellent. Furthermore, the student has completely control over the explanations rhythm.

A study indicated that the native digital students are drawn to visual media and creative tasks [12]. Therefore, the integration of multimedia as a reflection tool in learning is crucial in order to maintain students' motivation and engagement in the programming class.

In this paper, we present our conclusions after making several C programming chapters in multimedia format for the subject 'Fundamentals of Computer Science' for Industrial Engineering students at the University of Malaga (Spain). Our main conclusion is that students perceive that the multimedia approach as motivating and engaging. In section 2, the methodology used is presented. Subsequently, in section 3, the results of the surveys carried out on the students are commented. In section 4, the advantages and disadvantages of our proposal are discussed and, finally, the document is ended with some conclusions and future lines.

## 2 Methodology

Learning to program is intensely meticulous for novice undergraduates who do not have any background on computer programming [9]. At the moment, current teaching about theoretical concepts is based on 'chalk and talk' method and textbooks and this does not always work well. Many teachers have substituted the chalk with static

presentations but, in essence, the methodology is very similar to those classes taught thousands of years ago. Of course, practical classes are very different.

Chansilp and Oliver [2] suggest that the graphic representations of algorithms used in most textbooks are abstract visualizations and are not sufficient for learners to develop the logical thinking required in programming courses. The students usually only depend on the lectures and the textbook to develop their programming comprehension. However, just as Annamalai and Salam [1] said, the lecturers only go through the lesson once. Hence, the students have no lesson repetition, while this is crucial to learn and understand better. Students' problems with learning C programming are mostly based on a lack of understanding of conceptual and mental models [8].

Our proposal is based on providing the student with a set of files with the main explanations of several programming parts (such as control statements, inputs and outputs, functions, arrays, strings, structures, searching and sorting algorithms...). The general characteristics of this system are:

- Pace of learning: the student can stop and restart the explanations whenever he wants, as well as repeat them as many times as he wishes.
- Animations and sound: There are animations and sounds that make the experience much more attractive. This, joined with the teacher's voice-over, tries to focus the student's attention. For example, in a visually attractive way the student sees how the value of a variable changes in different statements, as the program is running.
- Step by step explanations: Theoretical and practical concepts are linked together and are explained with examples whose complexity is growing. For example, Fig. 1 shows a program that uses a simple array on the left, and the behavior of the program is explained to the right with animated graphics and tables.
- Sample programs: Complete programs, fragments or example functions are included, explaining their execution step by step. For example, Fig. 2 shows a frame of the animation explaining the selection sort algorithm. Step by step it is explained how the elements of the array are exchanged among themselves as shown by the arrows. It can be seen in the bottom part how at the end of the algorithm the array is ordered. At the end of each topic, various programs with fewer explanations are provided, inviting the student to check, understand and modify them.
- Complex concepts: The most complex concepts are explained in several ways because we have detected that different students understand better the ideas using different techniques for each of them. For example, in Fig. 3, an explanation about passing parameters by reference is shown (using the C++ style), which is an issue that is often extremely difficult to understand by beginner students (even more using the traditional C style).

**Arrays Unidimensionales:  
Declaración y Acceso**

```

...
const unsigned TAMA = 100;
typedef array<int, TAMA> TVector;
...
int main()
{
    TVector V;
    int i, suma = 0;

    for(i=0; i<TAMA; i++)
        V[i] = 1;

    for(i=0; i<TAMA; i++)
        suma = suma + V[i];
    // ¿Cuánto vale la suma?
    ...
}
    
```

V[0]	V[1]	...	V[98]	V[99]
1	1	...	1	1

Valor i	Instrucción a ejecutar
0	suma = suma + V[0]
1	suma = suma + V[1]
2	suma = suma + V[2]
...	...
99	suma = suma + V[99]
100	→ FIN del bucle

Fig. 1. Example of slide explaining a simple array. Source: own construction.

**Ordenación por "Selección": Ejemplo**

```

for (i=0; i<N-1; i++)
{ // Ordenar elemento en la posición i:
    PosMin = i;

    for (j=i+1; j<N; j++)
        if (V[j]<V[PosMin])
            PosMin = j;

    Swap(V[PosMin], V[i]);
}
    
```

N=6 Elementos

	7	3	5	9	4	7
i=0	3	7	5	9	4	7
i=1	3	4	5	9	7	7
i=2	3	4	5	9	7	7
i=3	3	4	5	7	9	7
i=4	3	4	5	7	7	9

Fig. 2. Example of slide explaining the selection sort algorithm. The comment in green says: Sort element on i position. Source: own construction.

**Paso de Argumentos Por REFERENCIA**  
**Ejemplo en C++**

En C++

• **Ejemplo:** ¿Qué salida produce el siguiente programa para distintas entradas?

```

// Probar paso de arg. por referencia en C++
#include <iostream>
using namespace std;

void f2 (int& x, char& ch){
    x = x + 3;
    if (ch == 'A')
        ch = 'P';
    else
        ch = 'K';
}

int main(){
    int x, y=1;
    char letra='A';
    cout << "\n- Número: ";
    cin >> x;
    y = y + x;
    f2(y,letra);
    cout << "- Valores: "<<x<<" , "<<y<<" y "<<letra<<".\n";
    f2(x,letra);
    cout << "- Valores: "<<x<<" , "<<y<<" y "<<letra<<".\n";
    return 0;
}
    
```

Vars. de f2():

	x	ch
Vars. de main():	x	letra
	3	'A'
	4	

**Ejemplo de Ejecución:**

- Número: 3

Fig. 3. Example of slide explaining the pass of parameters by reference using C++ style.  
Source: own construction.

### 3 Surveys

In order to inform the students about this multimedia material developed, it has been used in one class per course, for several years. After that class, a survey has been passed and they are advised that all the material is uploaded and available via web. In addition, students are asked to fill in an anonymous and optional survey each time they use the material (using the platform [www.surveymonkey.com](http://www.surveymonkey.com)). The students are informed that their opinion is extremely important in order to improve the material and to broaden the topics covered.

The most interesting questions and results of these surveys are as follows:

1. Do you find this multimedia format interesting and useful to learn?
  - Average response: 4.6 of 5 (standard deviation: 0.3).
2. Do you think the explanations are adequate?
  - Average response: 4 of 5 (standard deviation: 0.6).
3. Do you think the examples used are appropriate?
  - Average response: 4.2 of 5 (standard deviation: 0.2).
4. Have you repeated or will you repeat all or some parts of the explanations?
  - Three answers: No (0%); All (16.7%); Some parts (83.7%).
5. Do you think it is appropriate to use this material in class or it is better to leave it only for individual study? Five answers:

- a) I prefer only the teacher's explanations in class (33.3%);
  - b) In class, teacher explanations should be mixed with little use of multimedia material (50%);
  - c) The ideal option is to combine both methods equally (16.7%);
  - d) The ideal option is to use mainly the multimedia material, and the teacher would only answer questions or add some comments (0%);
  - e) I prefer only multimedia material in class (0%);
6. Evaluate the speed of the explanations, in general:
- Three answers: Too fast (16.7%); Good rhythm (83.3%); Too slow (0%).
7. Would you like to have many topics of all your subjects in this format?
- Average response: 4.3 of 5 (standard deviation: 0.3).

The surveys basically tell us that students appreciate this material, that they have used it and that they will continue using it to study. They also reflect that they do not like that these material are used in class, relegating the teacher's task to solve doubts. In class, students prefer the immediacy and spontaneity of a teacher and the ability to raise their hands and ask at any time a specific question that may arise. That is something that, until now, is not easy to incorporate into an automatic education system.

One of the most important aspects in the development of this type of material is the speed of the explanations. Surveys say that, although most students value the pace of explanations positively, there is a minority who notice it too quickly, and nobody values it as too slow. In our case, we must assess whether to make the explanations a little slower. But this must be done carefully, first because they could become excessively slow and boring and secondly because we must take into account that the student can repeat the explanation whenever he wants. At any time the student can stop the explanation and go back, which facilitates the understanding for those students who notice that the explanations go faster than they would like.

#### **4 Advantages and disadvantages of our approach**

In this section, the strengths and limitations of this approach are considered. It is highly difficult to solve simply and quickly all the problems listed in the introduction. However, the material we propose resolves or reduces these problems because of the following aspects:

- It enables the students to learn at their own pace. Students can repeat the explanations whenever they want, wherever they like and as much as they need. This means that if a student misses a class, he or she does not have to be burdened by that. Besides, students can even go ahead of the teacher's explanations, organizing better their available time and fitting it to their schedules. This is especially interesting in repeating students or students with previous knowledge, who do not have to miss the first weeks until their classmates reach their level.

- Students can solve their doubts going through this material and listening to the exact part or lesson that interests them.

On the other hand, teachers have noticed that students who use this material advance better and faster, with less doubt and mistakes.

The available formats (presentation and video) allow students to choose the format they prefer:

- The advantage of presentation files is that presentations are automatically stopped at certain points. This invites the student to take a break to reflect or read something, and move on whenever he or she wants. The drawback is that you cannot go back to a specific point to repeat just from the desired time but you can go back (or advance) by blocks or pages.
- One of the advantages of videos is that the student can go back the video or advance to the exact second from which he or she wants to restart the explanation. Another additional advantage is that videos can be viewed online, from any device, without the need to download it.

As the student survey shows, they prefer not to use this material in class or use it in a timely manner. This is because they prefer to interact with the teacher and the possibility of asking questions just when they arise, something that does not allow the system that we present here, but that could be combined with non-contact tutoring schedules, both offline tutoring (using email or other messaging applications), and online tutoring (using some chat or video calls).

Non-contact tutoring is essential in case of distance teaching or in the case of students who cannot attend classes or tutoring for any reason (incompatibility of schedules, illness ...).

In addition, these tutoring schedules can be individual or in groups (using forums or group video calls). One advantage of group interactions is that they can occur without the direct participation of the faculty, who can play a supervisory role and resolve doubts when no one else does. In this case, the students can help each other solving doubts. The great advantage of this is that a climate of group work and participation enhances learning. For some reason, there are some groups that like this kind of interaction between themselves and other groups in which members do not participate or do very little. However, the teacher's role in favoring that participation can be decisive. In any case, the multimedia material favors the interaction between students: students use it in groups, they recommend others a certain part where something concrete is explained, etc.

To capture the student's attention, different techniques have been used. For example, put an error intentionally, to surprise the student. There are other techniques that we still need to explore, such as asking the student to find an error, telling an anecdote related to the exercise that is being solved, or even a joke related to computer programming.

## **5 Conclusions and future lines**

The introductory programming subjects usually suppose that students do not have previous concepts, but in only 15 weeks, they have to reach a quite high level of

programming (exercises of arrays, structures, nested datatypes, nested loops...). According to our experience, students find difficult to reach an adequate level in such a short time. It would be desirable for these subjects to be annual because, even if the syllabus and the class hours were not extended, the fact of having more time to mature the concepts and practice would make these subjects easier to learn. However, the current structure of the curricula in Spain has led to the subjects to be quarterly instead of annual, which makes our suggestion difficult to be taken into account.

Given this fact, any material or idea that improves learning is well received. In this paper, a multimedia material to learn to program in C language is exposed. It solves or reduces the problems described in the introduction. On the other hand, according to the surveys made to students, this material is well accepted. Although it can be improved, students applaud and use it. It must be highlighted the last question of our survey, in which the students reflect their desire to have this multimedia files for all their subjects.

In the future, it might not exist face-to-face classes and students might have all classes in electronic format (presentations, videos, software, exercises, games...), so that only a few practical classes and classes to solve doubts would be given. Even these classes could also be taught by chat or videoconference. However, that is not our goal when presenting this multimedia material. We think that the relationship between teacher and student is overriding and universities should not ignore the figure of the teacher and much less replace him with electronic means. Nonetheless, this system can reduce the need of teachers and, above all, make it easier for students to access explanations when they wish and not when the schedule stipulates they have to.

As future lines of expansion of this work, it is proposed to introduce more interactivity between the multimedia documents and students. Moreover, it is suggested to translate these files into English to reach a wider audience through the Internet and for exchange students.

**Acknowledgments.** The authors would like to express their gratitude to the University of Málaga, supporting this work under the PIE17-175 project entitled “Autoaprendizaje de programación de ordenadores”.

## References

1. S. Annamalai, & S.N.A. Salam. A Multimedia Approach towards Learning C Programming: A Discussion from Novice Learners' Perspective. *Journal of Telecommunication, Electronic and Computer Engineering*, Vol. 9 No. 2-12 (2017)
2. K. Chansilp & R. Oliver. Using multimedia to develop students' programming concepts. *Proc. of EDU-COM 2002*, 91--101 (2003)
3. M. Erwig, K. Smeltzer, & X. Wang. What is a visual language? *J. Vis. Lang., Comput.*, vol. 38, pp. 9--17, (2017)
4. J. Galindo and P. Galindo. Teaching computer programming for industrial engineering without teacher. *7th Teaching & Education Conference* (2019)
5. A. Gomes & A. J. Mendes. Learning to program-Difficulties and solutions. *Proc. Int. Conf. Eng. Educ. (ICEE)*, Coimbra, Portugal, (2007)

6. M. S. Horn, E. T. Solovey, R. J. Crouser, & R. J. Jacob. Comparing the use of tangible and graphical programming languages for informal science education. Proc. 27th Conf. Hum. Factors Comput. Syst. (CHI), 975—984. Boston, MA, USA, (2009)
7. E. Lahtinen, K. Ala-Mutka, & H. Järvinen. A study of the difficulties of novice programmers. Proc. 10th Annu. Conf. Innov. Technol. Comput. Sci. Educ. (ITiCSE), Caparica, Portugal, 14--18 (2005)
8. M.Y. Law, C.S. Lee, & Y.T. Yu. Learning motivation in e-learning facilitated computer programming courses. Computers & Education, Vol. 55, no. 1, 218--228 (2010)
9. M. J. Lee & A. J. Koo. Personifying programming tool feedback improves novice programmers' learning. Proc. of the seventh international workshop on Computing education research, 109—116 (2011)
10. I. Milne & G. Rowe. Difficulties in learning and teaching programming-Views of students and tutors. Educ. Inf. Technol., vol. 7, no. 1, 55--66 (2002)
11. S. Monk. Programming Arduino: Getting Started with Sketches. McGraw Hill (2012)
12. S. Naz, S.H. Shirazi, T. Iqbal, D. Irfan, M. Junaid & Y. Naseer. Learning Programming through Multimedia and Dry-run. Research Journal of Applied Sciences, Engineering and Technology, Vol. 7, no.21, 4455--4463 (2014)
13. J.M. Rodríguez Corral, I. Ruíz-Rube, A. Civit Balcells, J.M. Mota-Macías, A. Morgado-Estévez & J.M. Dodero. Study on the Suitability of Visual Languages for Non-Expert Robot Programmers. IEEE Access 7:17535--17550 (2019)
14. D. Weragama & J. Reye. Analysing student programs in the PHP intelligent tutoring system. Int. Journal of Artificial Intelligence in Education, Vol. 24, no. 2, 162--188 (2014)

## Aplicación de técnicas de aprendizaje activo a la enseñanza de la programación de ordenadores

Eva Gibaja<sup>1</sup>[0000-0002-0184-8789], María Luque<sup>1</sup>[0000-0001-7735-8340], and  
Amelia Zafra<sup>1</sup>[0000-0003-3868-6143]

Universidad de Córdoba, Departamento de Informática y Análisis Numérico,  
Córdoba, España  
{egibaja, mluque, azafra}@uco.es

**Resumen** El alumno tiende a ser un elemento pasivo en el proceso de enseñanza-aprendizaje, limitándose a recibir lecciones magistrales. En este trabajo realizamos una propuesta para el programa de prácticas de la materia Programación, impartida en el Grado en Ingeniería Informática de la Universidad de Córdoba, enfocado a que el alumno se implique en su propio proceso de aprendizaje experimentando con problemas de complejidad media, que le acercan al mundo real. La propuesta se basa en el aprendizaje activo y pretende que el alumno adquiera las competencias de la materia mediante la experimentación con casos prácticos.

**Keywords:** Aprendizaje activo · Programación de ordenadores · Ingeniería informática.

### 1. Introducción

La enseñanza tradicional es un modelo aún presente en las aulas universitarias. Está basado en un concepto de transmisión de la enseñanza donde el profesorado imparte clases magistrales en las que intenta enseñar a su alumnado los aspectos teóricos de la materia. Desde el punto de vista de los alumnos, supone un aprendizaje fundamentalmente memorístico-conceptual, ya que suelen ser agentes pasivos en su proceso de aprendizaje en el aula. Además, al menos en sus aspectos teóricos, supone un acercamiento muy superficial a la forma en que se construye el conocimiento científico. A pesar de que, en una primera lectura, podría parecer que este tipo de enseñanza es descartable por completo, esta afirmación no es del todo acertada. Primero, porque una clase magistral sobre conocimientos específicos, impartida por el profesorado, es un excelente recurso para que los alumnos conozcan los aspectos centrales de cada tema. Por otro lado, en aulas masificadas resulta imposible poder llevar a cabo otro método de enseñanza.

Sin embargo, una enseñanza basada exclusivamente en clases magistrales puede llegar a considerarse, en cierto modo, obsoleta, dado que los estudiantes disponen de un gran número de fuentes de información alternativas a la mera transmisión de conocimientos. Por tanto, en la medida de lo posible, hay que incorporar otras actividades en las que el alumnado sea agente activo de su proceso

de aprendizaje. Como docentes debemos facilitar que los estudiantes desarrollen sus aptitudes para aprender y producir conocimiento, comprometiéndole de una manera activa en los complejos procesos de búsqueda, comprensión e implicación en su futuro laboral y personal [11]. Es fundamental la motivación y enseñarles a que quieran y puedan continuar aprendiendo al abandonar las aulas. Así, los alumnos deben *aprender a aprender*, sobre todo, en un campo como la Informática, cuyo desarrollo es vertiginoso. Esto implica que, cuando los estudiantes que ahora están en el aula se conviertan en profesionales, es probable que los instrumentos de que dispongan en el ejercicio de su actividad y las técnicas que empleen sean diferentes a las que se les haya podido transmitir. Lo que ahora importa, no es tanto poseer una información determinada, sino haber adquirido la capacidad para descubrir y saber encontrar esa información. No vale cualquier tipo de enseñanza, sino aquella que facilita y estimula el aprendizaje de las competencias humanas consideradas valiosas [10].

El término *aprendizaje activo* [2], hace referencia a un conjunto de métodos de aprendizaje que enfocan la responsabilidad del aprendizaje en los estudiantes. Estos métodos pretenden involucrar al estudiante en tareas en las que, además de actuar, reflexione sobre la acción que desarrolla [8]. Todos ellos incluyen, entre otros aspectos: a) la implicación de los estudiantes en algo más que la escucha pasiva, b) el énfasis en el desarrollo de habilidades en los estudiantes y c) la realización de tareas que requieren procesos de pensamiento de cierta complejidad [9]. El aprendizaje activo hace referencia a que el estudiante se identifica con un proceso en el que investiga. No es un proceso en el que se busca una culminación concreta, sino que es el propio estudiante quien va a volver a aprender y reconstruir como herramienta y estrategia a lo largo del aprendizaje. De ahí, que se necesite un sujeto que indague y sienta curiosidad por lo que es desconocido (*learning by doing* [6]). El papel que debe desempeñar el alumno es un papel activo, autónomo, estratégico, reflexivo, cooperativo y responsable [7]. Esta estrategia ha sido utilizada con éxito tanto en otros estudios de Grado ligados a la ingeniería [14] [5], como de Ingeniería Informática [13] [4] [3] [12].

De forma más particular, en este trabajo se proponen un conjunto de actividades inspiradas en el aprendizaje activo para el desarrollo del programa práctico de la materia *Programación* impartida en el título de Grado en Ingeniería Informática de la Universidad de Córdoba. Esta materia abarca dos asignaturas denominadas *Introducción a la Programación (IP)* y *Metodología de la Programación (MP)*.

La necesidad de aplicar técnicas de aprendizaje activo surgió al detectar un bajo rendimiento de los estudiantes al finalizar el curso. A la hora de preparar los exámenes, es común la falta de consultas bibliográficas y dejar todo el estudio para el final, lo que implica una baja asimilación de los conceptos de la asignatura, una escasa asociación entre los contenidos de ésta y la dificultad de llevar los conceptos de la teoría a la práctica. Por ello, mediante el uso de aprendizaje activo, pretendemos plantear a los alumnos el desarrollo de tareas de complejidad media, basadas en el mundo real, que les permitan reflexionar

sobre los contenidos teóricos de la materia y aplicarlos en la resolución de dicha tarea.

En la Sección 2 se presenta una introducción al contexto de la materia Programación dentro del plan de estudios. A continuación, las Secciones 3 y 4 describen las propuestas presentadas y, finalmente, la Sección 5 presenta un conjunto de conclusiones.

## 2. Contexto académico

El título de Grado en Ingeniería Informática, en el ámbito internacional se corresponde con las titulaciones denominadas de forma genérica *Computer Science*, *Computer Engineering*, *Information Systems*, *Information Technology* y *Software Engineering*, según las recomendaciones de ACM [1].

En la Universidad de Córdoba, el objetivo general del título es la formación de profesionales en el ámbito de la Ingeniería Técnica en Informática con una base amplia y generalista de conocimiento en Ingeniería Informática y una formación que garantice la adquisición de los conocimientos específicos de las especialidades de Ingeniería del Software, Ingeniería de Computadores y Computación. Además, el título proporciona la capacidad de aplicación de dichos conocimientos a las actividades propias de la profesión del Ingeniero Técnico en Informática. El Grado consta de 4 cursos, el primero de ellos constituye el *módulo de formación básica* con contenidos de Informática, Matemáticas, Física y Empresa. Dentro de este módulo se encuentra la materia *Programación*, que abarca dos asignaturas: *Introducción a la Programación* y *Metodología de la Programación*.

## 3. Programa práctico para IP

*Introducción a la Programación* es impartida en el primer cuatrimestre con un total de 6 ECTS. La Tabla 1 muestra un resumen de las competencias y contenidos de la asignatura. El lenguaje de programación utilizado es C. Para el desarrollo del programa práctico, se proponen dos actividades. La primera de las actividades propuestas está basada en el álgebra lineal que hay detrás de los buscadores de internet y la segunda consiste en el desarrollo del juego de la vida. La Tabla 2 muestra un esquema de los contenidos teóricos de la asignatura cubiertos por cada una de las propuestas de prácticas presentadas.

### 3.1. Actividad 1: Clasificación de páginas Web

Supongamos que todas las páginas web accesibles desde cualquier navegador se pueden clasificar en cinco grupos diferenciados, en función del contenido de éstas; y que se tiene la matriz de probabilidad de la ecuación (1), donde un elemento  $M(i, j)$ , indica la probabilidad de pasar de una página del grupo  $j$ , a una página del grupo  $i$ . Por ejemplo, el elemento  $M(2, 3)$  vale 0,20, lo que indica que la probabilidad de pasar de una página del grupo 3 a otra del grupo 2 es de

Tabla 1: Contenidos impartidos en la asignatura IP

COMPETENCIAS
CB2: Que los estudiantes sepan aplicar sus conocimientos a su trabajo o vocación de una forma profesional y posean las competencias que suelen demostrarse por medio de la elaboración y defensa de argumentos y la resolución de problemas en el campo de la Ingeniería Informática
CB4: Que los estudiantes puedan transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado
CEB3: Capacidad para comprender y dominar los conceptos básicos de matemática discreta, lógica, algorítmica y complejidad computacional, y su aplicación para la resolución de problemas propios de la ingeniería
CEB4: Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería
CEB5: Conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la ingeniería
CONTENIDOS
Introducción a la informática
Algoritmos y Programas
Fundamentos de programación: Tipos de datos, operadores, expresiones, estructuras de control
Autodocumentación y estilos de programación
Tipos de datos compuestos: estructuras, arrays, cadenas
Programación estructurada y modular

Tabla 2: Contenidos de IP practicados en cada propuesta

	Algorítmica	Tipos de datos	Operadores expresiones	Estructuras control	Documentación	Arrays cadenas	Funciones
Páginas Web	√	√	√	√	√	√	√
Juego vida	√	√	√	√	√	√	√

0,20. La suma de los elementos de cualquier columna ha de ser 1 (a este tipo de matrices se les denomina *estocásticas*).

$$\mathbf{M} = \begin{pmatrix} 0,09 & 0,08 & 0,10 & 0,16 & 0,12 \\ 0,11 & 0,12 & 0,20 & 0,04 & 0,08 \\ 0,09 & 0,08 & 0,09 & 0,10 & 0,15 \\ 0,20 & 0,15 & 0,19 & 0,15 & 0,10 \\ 0,51 & 0,57 & 0,42 & 0,55 & 0,55 \end{pmatrix} \tag{1}$$

Para representar la página en la que estamos ubicados en un momento dado, se utiliza un vector de estados, de cinco elementos, en el cual todos los elementos son 0, excepto el elemento correspondiente al grupo al que pertenece la página en la que estamos ubicados, que vale 1. Por ejemplo, si estamos inicialmente en una página perteneciente al grupo 2, el vector de estados  $\mathbf{v}_0$  será:

$$\mathbf{v}_0 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{2}$$

Si, por ejemplo, estamos en una página del grupo 2, y accedemos a otra página, podemos obtener un nuevo vector  $\mathbf{v}_1$  que nos indica la probabilidad de

estar en una página perteneciente a cualquier grupo, de la siguiente forma:

$$\mathbf{v}_1 = \mathbf{M} * \mathbf{v}_0 = \begin{pmatrix} 0,08 \\ 0,12 \\ 0,08 \\ 0,15 \\ 0,57 \end{pmatrix} \quad (3)$$

Generalizando, después de cambiar de página  $n$  veces, se podría obtener el vector de estados  $\mathbf{v}_n$ , donde  $\mathbf{M}^n$  es la potencia  $n$ -ésima de la matriz  $\mathbf{M}$ , como:

$$\mathbf{v}_n = \mathbf{M}^n * \mathbf{v}_0 \quad (4)$$

### Propuesta de trabajo

Implementar un programa en C, en el que se obtenga la potencia  $n$ -ésima de la matriz  $\mathbf{M}$ , así como el vector de estados  $\mathbf{v}_n$ , partiendo de una página inicial dada por el usuario (se indicará a qué grupo pertenece, valor comprendido entre 1 y 5). La matriz se inicializará en el momento de su declaración con los valores de la matriz de ejemplo (1). Se implementará una función que multiplique dos matrices, de cualquier dimensión (siempre que se puedan multiplicar), y otra función, que invocando a la anterior, obtenga la potencia  $n$ -ésima de una matriz cuadrada. También se implementará una función para visualizar la matriz  $\mathbf{M}^n$ .

### 3.2. Actividad 2: El juego de la vida

Consideremos una población de  $K$  insectos en una matriz de dimensiones  $M \times N$ , de modo que en cada celda de la matriz hay, como máximo, un insecto. Por lo tanto, cada insecto tiene, como máximo, 8 insectos vecinos. La población está en desarrollo continuamente debido a los nacimientos y muertes que se producen siguiendo las siguientes reglas de evolución:

- Aquellos insectos que tienen 0, 1, 4, 5, 6, 7 u 8 vecinos mueren.
- Los insectos que tienen 2 o 3 vecinos sobreviven.
- En las celdas vacías con 2 o 3 vecinos nace un nuevo insecto.
- Los insectos que nacen o mueren no afectan a las reglas hasta que se ha completado un ciclo evolutivo, entendiéndose por éste un ciclo en el que se ha decidido la supervivencia o muerte de los insectos (vivos al comenzar el ciclo) de acuerdo con las reglas mencionadas. Así, al calcular los nuevos estados de las celdas de la matriz las modificaciones se deben hacer sobre una matriz auxiliar que, finalmente, se copiará en la matriz original.

### Propuesta de trabajo

Escribir una función *Evolucion* que simule la evolución de la población y que:

1. Reciba como entrada los enteros positivos  $K$ ,  $N$  y  $M$ , un *array* de estructuras con las coordenadas  $x$  e  $y$  de las celdas en las que se encuentran los  $K$  insectos iniciales de la población y un entero positivo  $L$  que indica cuántos ciclos se van a simular.

2. Muestre por pantalla el estado inicial de la población, marcando con un asterisco (\*) las celdas ocupadas por insectos.
3. Simule la evolución de la población durante los  $L$  ciclos evolutivos y represente en la pantalla el estado de la población en cada ciclo representándolo como en el siguiente ejemplo:
  - Con una matriz de  $10 \times 10$ , 3 insectos colocados inicialmente en las posiciones  $(0, 0)$ ,  $(0, 1)$  y  $(1, 0)$  y 3 ciclos de evolución tendríamos la configuración inicial de la Tabla 3a y las configuraciones de las Tablas 3b, 3c y 3d para la primera, segunda y tercera evolución respectivamente.

Tabla 3: Evolución de la población

(a) Configuración inicial	(b) Primera iteración																																																																																										
<table style="margin: auto; border-collapse: collapse;"> <tr><td style="border-bottom: 1px solid black;">0</td><td style="border-bottom: 1px solid black;">1</td><td style="border-bottom: 1px solid black;">2</td><td style="border-bottom: 1px solid black;">3</td><td style="border-bottom: 1px solid black;">4</td><td style="border-bottom: 1px solid black;">5</td><td style="border-bottom: 1px solid black;">6</td><td style="border-bottom: 1px solid black;">7</td><td style="border-bottom: 1px solid black;">8</td><td style="border-bottom: 1px solid black;">9</td></tr> <tr><td>*</td><td>*</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td>*</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> </table>	0	1	2	3	4	5	6	7	8	9	*	*									*										<table style="margin: auto; border-collapse: collapse;"> <tr><td style="border-bottom: 1px solid black;">0</td><td style="border-bottom: 1px solid black;">1</td><td style="border-bottom: 1px solid black;">2</td><td style="border-bottom: 1px solid black;">3</td><td style="border-bottom: 1px solid black;">4</td><td style="border-bottom: 1px solid black;">5</td><td style="border-bottom: 1px solid black;">6</td><td style="border-bottom: 1px solid black;">7</td><td style="border-bottom: 1px solid black;">8</td><td style="border-bottom: 1px solid black;">9</td></tr> <tr><td>*</td><td>*</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>*</td></tr> <tr><td>*</td><td>*</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>*</td></tr> </table>	0	1	2	3	4	5	6	7	8	9	*	*								*	*	*								*																														
0	1	2	3	4	5	6	7	8	9																																																																																		
*	*																																																																																										
*																																																																																											
0	1	2	3	4	5	6	7	8	9																																																																																		
*	*								*																																																																																		
*	*								*																																																																																		
<hr style="width: 100%;"/> <table style="margin: auto; border-collapse: collapse;"> <tr><td style="border-bottom: 1px solid black;">0</td><td style="border-bottom: 1px solid black;">1</td><td style="border-bottom: 1px solid black;">2</td><td style="border-bottom: 1px solid black;">3</td><td style="border-bottom: 1px solid black;">4</td><td style="border-bottom: 1px solid black;">5</td><td style="border-bottom: 1px solid black;">6</td><td style="border-bottom: 1px solid black;">7</td><td style="border-bottom: 1px solid black;">8</td><td style="border-bottom: 1px solid black;">9</td></tr> <tr><td></td><td>*</td><td></td><td></td><td></td><td></td><td></td><td>*</td><td></td><td></td></tr> <tr><td>*</td><td>*</td><td></td><td></td><td></td><td></td><td></td><td>*</td><td>*</td><td></td></tr> <tr><td>*</td><td>*</td><td></td><td></td><td></td><td></td><td></td><td>*</td><td></td><td>*</td></tr> </table>	0	1	2	3	4	5	6	7	8	9		*						*			*	*						*	*		*	*						*		*	<hr style="width: 100%;"/> <table style="margin: auto; border-collapse: collapse;"> <tr><td style="border-bottom: 1px solid black;">0</td><td style="border-bottom: 1px solid black;">1</td><td style="border-bottom: 1px solid black;">2</td><td style="border-bottom: 1px solid black;">3</td><td style="border-bottom: 1px solid black;">4</td><td style="border-bottom: 1px solid black;">5</td><td style="border-bottom: 1px solid black;">6</td><td style="border-bottom: 1px solid black;">7</td><td style="border-bottom: 1px solid black;">8</td><td style="border-bottom: 1px solid black;">9</td></tr> <tr><td></td><td></td><td>*</td><td></td><td></td><td></td><td></td><td>*</td><td>*</td><td></td></tr> <tr><td></td><td></td><td>*</td><td>*</td><td></td><td></td><td></td><td>*</td><td>*</td><td></td></tr> <tr><td></td><td></td><td>*</td><td>*</td><td></td><td></td><td></td><td>*</td><td>*</td><td>*</td></tr> <tr><td></td><td></td><td>*</td><td>*</td><td></td><td></td><td></td><td>*</td><td>*</td><td>*</td></tr> </table>	0	1	2	3	4	5	6	7	8	9			*					*	*				*	*				*	*				*	*				*	*	*			*	*				*	*	*
0	1	2	3	4	5	6	7	8	9																																																																																		
	*						*																																																																																				
*	*						*	*																																																																																			
*	*						*		*																																																																																		
0	1	2	3	4	5	6	7	8	9																																																																																		
		*					*	*																																																																																			
		*	*				*	*																																																																																			
		*	*				*	*	*																																																																																		
		*	*				*	*	*																																																																																		
<hr style="width: 100%;"/> <table style="margin: auto; border-collapse: collapse;"> <tr><td style="border-bottom: 1px solid black;">0</td><td style="border-bottom: 1px solid black;">1</td><td style="border-bottom: 1px solid black;">2</td><td style="border-bottom: 1px solid black;">3</td><td style="border-bottom: 1px solid black;">4</td><td style="border-bottom: 1px solid black;">5</td><td style="border-bottom: 1px solid black;">6</td><td style="border-bottom: 1px solid black;">7</td><td style="border-bottom: 1px solid black;">8</td><td style="border-bottom: 1px solid black;">9</td></tr> <tr><td>*</td><td>*</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td>*</td><td>*</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>*</td></tr> </table>	0	1	2	3	4	5	6	7	8	9	*	*									*	*								*	<hr style="width: 100%;"/> <table style="margin: auto; border-collapse: collapse;"> <tr><td style="border-bottom: 1px solid black;">0</td><td style="border-bottom: 1px solid black;">1</td><td style="border-bottom: 1px solid black;">2</td><td style="border-bottom: 1px solid black;">3</td><td style="border-bottom: 1px solid black;">4</td><td style="border-bottom: 1px solid black;">5</td><td style="border-bottom: 1px solid black;">6</td><td style="border-bottom: 1px solid black;">7</td><td style="border-bottom: 1px solid black;">8</td><td style="border-bottom: 1px solid black;">9</td></tr> <tr><td>*</td><td>*</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>*</td></tr> <tr><td>*</td><td>*</td><td></td><td></td><td></td><td></td><td></td><td>*</td><td>*</td><td>*</td></tr> <tr><td>*</td><td>*</td><td></td><td></td><td></td><td></td><td></td><td>*</td><td>*</td><td>*</td></tr> </table>	0	1	2	3	4	5	6	7	8	9	*	*								*	*	*						*	*	*	*	*						*	*	*																				
0	1	2	3	4	5	6	7	8	9																																																																																		
*	*																																																																																										
*	*								*																																																																																		
0	1	2	3	4	5	6	7	8	9																																																																																		
*	*								*																																																																																		
*	*						*	*	*																																																																																		
*	*						*	*	*																																																																																		

Para ello serán necesarias, al menos, las siguientes funciones auxiliares:

- *Pinta*, que recibe la matriz con los insectos y los datos necesarios para pintarla por pantalla utilizando asteriscos, tal y como muestra la Tabla 3d.
- *CuentaVecinos*, que recibe como parámetro la matriz de insectos, las coordenadas de una posición dentro de la matriz y los datos necesarios para devolver el número de insectos vecinos a esa posición. Hay que considerar la matriz circular, es decir, si tenemos una matriz de  $4 \times 4$  y estamos inspeccionando la casilla  $(3, 3)$  (esquina inferior derecha), los vecinos serán las casillas  $(2, 2)$ ,  $(2, 3)$ ,  $(2, 0)$ ,  $(3, 2)$ ,  $(3, 0)$ ,  $(0, 2)$ ,  $(0, 3)$  y  $(0, 0)$  (véase Tabla 4).

Tabla 4: Vecinos de la casilla (3,3)

0,0	0,1	0,2	0,3
1,0	1,1	1,2	1,3
2,0	2,1	2,2	2,3
3,0	3,1	3,2	<b>3,3</b>

- *ProximaGeneracion*, que recibe la matriz de insectos y los datos necesarios para devolver, en la misma matriz, el estado de la matriz después de una generación donde se han aplicado las reglas del juego de la vida (internamente, la función utilizará una matriz auxiliar).

La práctica deberá completarse con la implementación de un programa completo que haga uso de la función *Evolución* para la simulación de una población de insectos.

#### 4. Programa práctico para MP

*Metodología de la Programación* es impartida en el segundo cuatrimestre con un total de 6 ECTS. La Tabla 5 muestra un resumen de las competencias y contenidos de la asignatura. El lenguaje de programación utilizado es C. Para el desarrollo del programa práctico, se proponen tres actividades. La primera y la segunda propuestas son una aplicación de los contenidos de la asignatura a un problema de gestión de datos complejos. Finalmente, la tercera consiste en un problema de gestión de colas para ventanillas de atención al público. La Tabla 6 muestra un esquema de los contenidos teóricos de la asignatura cubiertos por cada una de las propuestas de prácticas presentadas.

Tabla 5: Contenidos impartidos en la asignatura MP

COMPETENCIAS
CB4: Que los estudiantes puedan transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado
CU2: Conocer y perfeccionar el nivel de usuario en el ámbito de las TIC
CEB4: Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería
CEB5: Conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la ingeniería
CONTENIDOS
Punteros
Ficheros: Texto y Binarios
Estructura de un programa en tiempo de ejecución
Memoria dinámica
Recursividad
Estructuras lineales dinámicas de datos: Listas, Pilas, Colas
Algoritmos básicos de búsqueda y ordenación y su complejidad algorítmica
Aspectos metodológicos de la programación: Documentación y Pruebas
Herramientas: generación automática de proyectos, documentación, bibliotecas, depuradores

Tabla 6: Contenidos de MP practicados en cada propuesta

	Memoria	Ficheros	Listas	Pilas colas	Punteros	Make	Argumentos <i>main</i>	Bibliot.	Orden
Conejos	✓	✓	✓		✓	✓	✓	✓	✓
Polígonos	✓	✓	✓	✓	✓	✓	✓	✓	✓
Colas	✓		✓	✓	✓		✓		✓

#### 4.1. Actividad 1: Gestión de una granja de conejos

Una empresa ganadera, dedicada a la cría y venta de conejos, dispone de un fichero binario, denominado *conejos.bin*, con los datos de los conejos criados en el año 2019. Los registros del fichero poseen la siguiente estructura:

- *Código* del conejo (cadena de 6 caracteres).
- *Edad* del conejo (en días) de tipo entero.
- *Peso* del conejo (en gramos) de tipo *double*.

La empresa ha detectado, que debido al peso, no todos los conejos pueden ser vendidos. Así, los conejos que no sobrepasan un determinado peso, no son deseados por los consumidores, y los conejos que sobrepasan un determinado peso tienen un exceso de grasa y no son muy aptos para el consumo.

##### Propuesta de trabajo

- Implementar un programa en C, que realice secuencialmente las siguientes operaciones:
  1. Paso de los registros del fichero a una lista doblemente enlazada.
  2. Cálculo del peso medio,  $m$ , de los conejos y de la desviación típica de los pesos,  $s$ , utilizando la lista doblemente enlazada.
  3. Eliminación de la lista doble de aquellos conejos cuyo peso sea inferior a  $m - 1,5 * s$ , y cuyo peso sea superior a  $m + 1,5 * s$ .
  4. Paso de los elementos de la lista resultante a un vector dinámico.
  5. Ordenación de los elementos del vector en orden decreciente del peso.
  6. Almacenamiento de los elementos del vector en un fichero de texto.
- Se usarán los siguientes archivos:
  - Un archivo *principal.c*, que contendrá la función *main*.
  - Los archivos *ficheros.c*, *listas.c*, y *vectores.c*, que contendrán las funciones correspondientes a ficheros, listas y vectores respectivamente. Cada uno de estos tres archivos llevará su correspondiente archivo de cabecera (*.h*).
  - Los tipos de datos para la lista y el vector, se implementarán en un archivo de cabecera denominado *tipos.h*.
- Se creará un fichero *makefile* con las siguientes características,
  - Se creará una biblioteca, *conejos.lib* con los archivos *ficheros.o*, *listas.o*, y *vectores.o*.
  - El ejecutable se creará a partir de la biblioteca y el archivo *principal.o*.
- Los nombres de los archivos de entrada y salida (*conejos.bin* y *conejosSeleccionados.txt*) se introducirán en línea de órdenes.

## 4.2. Actividad 2: Gestión de un fichero con polígonos

Se tiene un fichero de texto que almacena la información necesaria para representar polígonos en el plano cartesiano. Cada línea del fichero representa un polígono de la siguiente forma:

- Un número entero,  $n$ , seguido de un espacio en blanco, que indica el número de lados del polígono.
- A continuación,  $2 * n$  enteros separados por espacios en blanco y que representan las coordenadas  $(x, y)$  de cada punto del polígono (véase Archivo 1.1).

Archivo 1.1: Ejemplo de fichero de polígonos

```
9 185 4 142 1 113 12 71 74 75 115 97 149 167 172 183 169 231 132
4 217 4 192 33 201 37 221 28
2 10 297 8 308
8 200 315 170 297 145 298 108 352 120 381 126 387 187 392 212 348
9 195 0 194 0 190 2 188 4 186 12 198 21 198 21 201 21 205 17
```

### Propuesta de trabajo

Realizar un programa que implemente una serie de funciones para procesar y visualizar listas de polígonos. El programa realizará las siguientes operaciones:

1. Leer de un fichero de texto los polígonos y almacenarlos en dos listas (Figura 1):
  - *listaTriangulos*: contendrá los polígonos que tengan tres lados. El resultado se mostrará por pantalla.
  - *listaGeneral*: contendrá los polígonos que no tengan tres lados. En este caso se realizará una inserción ordenada por el número de lados, que podrá ser ascendente o descendente. Este apartado se implementará utilizando punteros a funciones. El resultado se mostrará por pantalla.
2. Para la lista *listaTriángulos*:
  - Calcular el perímetro de cada uno de los elementos de la lista.
  - Borrar de la lista aquellos polígonos cuyo perímetro sea inferior a 400. El resultado se mostrará por pantalla.
3. Mostrar los polígonos de *listaGeneral* de  $N$  lados ( $N$  se leerá desde el teclado).

Requisitos y recomendaciones:

- El programa se ejecutará secuencialmente, sin menús.
- El programa se estructurará en 7 ficheros:
  1. *main.c*: contiene el programa principal, *main*.
  2. *listas.c*: funciones relacionadas con las listas.
  3. *listas.h*: fichero de cabecera para *listas.c*.
  4. *ficheros.c*: funciones relacionadas con la lectura de fichero de polígonos.
  5. *ficheros.h*: fichero de cabecera para *ficheros.c*.
  6. *miscelanea.c*: para el resto de las funciones.
  7. *miscelanea.h*: fichero de cabecera para *miscelanea.c*.

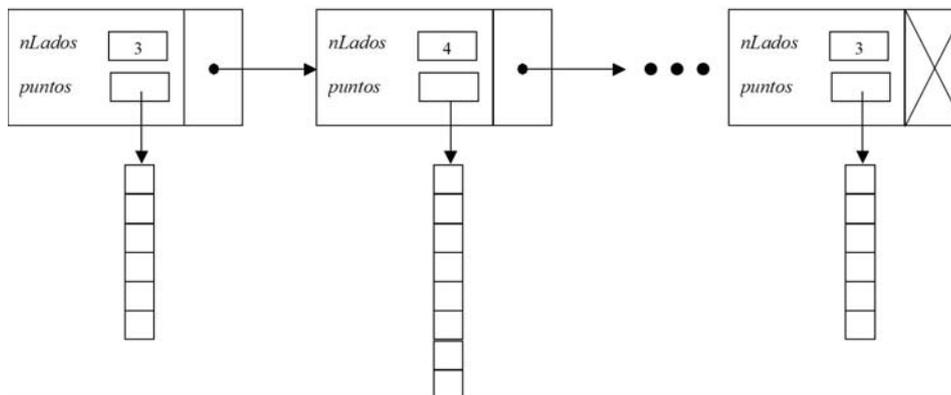


Figura 1: Ejemplo de lista de polígonos

- Se utilizará un fichero *makefile*.
  - Se desarrollará una biblioteca a partir de los ficheros *listas.o*, *ficheros.o* y *miscelanea.o*.
  - El ejecutable se creará a partir de *main.o* y de la biblioteca.
- El nombre del fichero de polígonos y el orden de inserción en *listaGeneral* (ascendente o descendente) se introducirá como un parámetro en la llamada al programa. Es decir, si el programa se llama *poligonos*, la llamada sería:
  - *poligonos ficheroPoligonos.txt a (orden ascendente)*
  - *poligonos ficheroPoligonos.txt d (orden descendente)*
- Se implementará una función para comprobar que la llamada al programa se ha realizado correctamente. Esto es, que el número de argumentos es el correcto y que el fichero de polígonos existe. En caso contrario, terminará la ejecución del programa y se mostrará al usuario un mensaje de error:
  - *Error: en el numero de argumentos. Sintaxis correcta: poligonos fichero-Poligonos.txt a | d*
  - *Error: el fichero de polígonos <ficheroPoligonos.txt> no existe*

#### 4.3. Actividad 3: Gestión de colas

Normalmente cuando accedemos a un servicio atendido por varias ventanillas (estación de tren, estación de autobuses, etc.), nos colocamos en la ventanilla cuya cola tiene menos personas esperando y después suele suceder que esa cola es atendida más lentamente que las demás porque hay uno o varios usuarios que requieren más tiempo de atención que los demás.

##### Propuesta de trabajo

Simular el funcionamiento de tres ventanillas (*A*, *B*, *C*), siguiendo tres estrategias distintas a la hora de seleccionar la cola en la cual se ubica el usuario, para evaluar cual es la mejor y ver las diferencias entre ellas. A continuación se detallan los pasos a seguir:

1. Introducir por teclado el número de personas,  $n$ , que vamos a utilizar para simular el funcionamiento.
2. Reservar un vector dinámico de tipo entero de  $n$  elementos, en el cual cada elemento será un número aleatorio comprendido entre 1 y 5. Este número será el tiempo de atención que requiere cada usuario.
3. Recorrer el vector desde el elemento 1 hasta el  $n$ , asignando cada elemento a la cola que le corresponda, siguiendo una de las tres estrategias que se detallarán más adelante. Cada elemento de la cola, tendrá tres campos:
  - Carácter: indica la cola en la que está ( $A$ ,  $B$  o  $C$ ).
  - Entero: indica su tiempo de atención (valor obtenido en el paso 2).
  - Entero: indica el tiempo de permanencia en la cola. Se obtendrá sumando el tiempo de atención de dicha persona a los tiempos de atención de todas las personas que le preceden en su cola.
4. Una vez que las colas están llenas, después del paso 3, se irán vaciando simultáneamente en orden creciente de los tiempos de permanencia, y los elementos se almacenarán en una lista, llamada *salida*, en la cual se almacenarán todas las personas. Las colas serán liberadas una vez que se vacíen.
5. Una vez completa la lista de salida, se visualizarán sus elementos en orden creciente y se calculará el tiempo medio de permanencia de todas las personas que componen esta lista.

Para simplificar el problema, supondremos que hasta que las personas no se han distribuido en las tres ventanillas, no serán atendidas y no saldrán de su cola. Las tres estrategias a seguir para seleccionar una ventanilla serán las siguientes:

1. *Estrategia 1.* Seleccionar la cola con menos personas.
2. *Estrategia 2.* Seleccionar aleatoriamente una de las tres colas.
3. *Estrategia 3.* Seleccionar aquella cola en la cual la suma de los tiempos de atención de los individuos que ya están en ella es el menor.

Para poder simular el funcionamiento de las tres estrategias, usando los mismos datos, los pasos 1 y 2 se harán solo una vez, y los pasos 3, 4 y 5 se repetirán tres veces, una vez para cada tipo de estrategia.

## 5. Conclusiones

Este trabajo ha presentado una serie de propuestas para aprendizaje activo de la materia *Programación* basadas en plantear al alumno problemas de complejidad media del mundo real que deben ser resueltos mediante el análisis, diseño e implementación de un programa informático. Al experimentar con casos prácticos reales, los alumnos deben ser capaces de reflexionar e interiorizar los conceptos de la asignatura y tomar una actitud activa en su proceso de aprendizaje. Así, la aplicación del aprendizaje activo para el desarrollo del programa práctico de la materia ha resultado muy adecuado para desarrollar las competencias esperadas en un futuro ingeniero y promover el trabajo autónomo del alumno.

## Referencias

1. The ACM-IEEE Joint Task Force for Computing Curricula 2005: Computing Curricula 2005 (CC2005): The Overview Report (2005)
2. Bonwell, C.C., Eison, J.A.: Active Learning: Creating Excitement in the Classroom. 1991 ASHE-ERIC Higher Education Reports. ERIC (1991)
3. García-Peñalvo, F.J., Moreno García, M.N., Bravo Martín, S., Conde González, M.Á.: Aprendizaje basado en problemas para la parte práctica de la materia ingeniería del software. <https://gredos.usal.es/jspui/handle/10366/81695> (2010)
4. González, A., Rodríguez, M., Olmos, S.: Aprendizaje activo en ingeniería técnica informática, esp. gestión. sistemas informáticos. In: Actas de las I Jornadas de Innovación Educativa de la Escuela Politécnica Superior de Zamora. pp. 627–640 (2006)
5. Jair E. Rocha, Carlos A. Arango, J.L.: Aprendizaje activo en ingeniería industrial. In: Eleventh LACCEI Latin American and Caribbean Conference for Engineering and Technology (LACCEI'2013). Cancun, Mexico (aug 2013)
6. Jovanovic, B., Nyarko, Y.: Learning by doing and the choice of technology. *Econometrica* pp. 1299–1310 (1996)
7. March, A.F.: Metodologías activas para la formación de competencias. *Educatio siglo XXI* **24**, 35–56 (2006)
8. Mestre, M.J.O., Palao, C.G., Peris, M.F., Navarro, M.B.: Aprendizaje activo y desempeño del estudiante: diseño de un curso de dirección de la producción (active learning methods and student performance: A design of a production management course). *WPOM-Working Papers on Operations Management* **3**(2), 84–100 (2012), ISSN: 1989-9068
9. Navarro, L.P.: Aprendizaje activo en el aula universitaria: el caso del aprendizaje basado en problemas. *Miscelánea Comillas. Revista de Ciencias Humanas y Sociales* **64**(124), 173–196 (2006), ISSN: 2341-085X
10. Nutall, G.: The cultural myths and realities of classrooms. *Teaching and learning: a Reacher College Record* pp. 895–934 (2005)
11. Pérez, A., Soto, E., Sola, M., Serván, M.: In: AKAL, E. (ed.) *Aprender en la Universidad. El sentido del Cambio en el EEES, Espacio Europeo de Educación Superior*, vol. 1 (2009)
12. Rojas Ruiz, F.: Estrategia de aprendizaje activo y cooperativo para periféricos y dispositivos de interfaz humana en el grado de ingeniería informática. *Enseñanza y Aprendizaje de Ingeniería de Computadores* (3), 69–76 (2013), ISSN: 2173-8688
13. Spinel, S.C., Ortiz, J.C.R.: Prácticas docentes que promueven el aprendizaje activo en ingeniería civil. *Revista de ingeniería* (18), 48–55 (2003), ISSN 0121-4993
14. Trujillo Suárez, C.A., González Agudelo, E.M.: Aprendizaje activo en cursos básicos de ingeniería: un ejemplo en la enseñanza de dinámica. *Uni-pluri/versidad* **10**(2), 65–67 (2010), ISSN: 2665-2730

## **Modelo para fomentar el aprendizaje activo en las Plataformas LMS con base en Design Thinking y la Taxonomía de Bloom con un enfoque ágil**

Cynthia López Valerio  
Facultad de Tecnologías de la Información y las Comunicaciones,  
Universidad Latina de Costa Rica  
cynthia.lopez@ulatina.cr

**Resumen.** Se plantea un modelo que ayuda a fomentar el aprendizaje activo en las plataformas LMS para este caso particular Moodle. Como principal herramienta se utiliza el Design Thinking o Diseño del Pensamiento combinado con los verbos de la taxonomía pues con ello ofrece mayor claridad al estudiante de como abordar un problema o caso y todo lo vincula a través de un enfoque ágil.

**Palabras Clave:** Design Thinking, Aprendizaje activo, Taxonomía de Bloom, Enfoque Ágil, Plataformas LMS, Modelo.

**Abstract.** A model that helps to promote active learning in the LMS platforms for this particular Moodle case is proposed. Design Thinking or Design of Thinking combined with the verbs of the taxonomy is used as the main tool, since it offers greater clarity to the student on how to approach a problem or case and links everything through an agile approach.

**Keywords:** Design Thinking, Active Learning, Bloom Taxonomy, Agile Approach, LMS Platforms, Model

### **1 Introducción**

La revolución de la educación ha venido a cambiar la forma tradicional en que vemos las clases por un enfoque digital donde el principal actor es la tecnología y las formas variantes que esta incluye dentro de plataformas interactivas o clases virtuales. El aprendizaje en línea o e-learning, tiene un uso intensivo de la web y el incremento de los dispositivos móviles, lo cual ha facilitado el desarrollo de nuevas plataformas, formas de educación y metodologías de aprendizaje que combinan las ya existentes con las activas como aula invertida, Design Thinking, rubricas, entre otras. Fomentar el aprendizaje colaborativo e interactivo es unas de las características más importantes y de los cuales los estudiantes siempre son los más beneficiados.

En este trabajo se propone un modelo para desarrollar e-learning a partir de las bases de la Taxonomía de Bloom, combinado con el Design Thinking en una plataforma LMS para este caso Moodle, siguiendo un enfoque ágil que contribuya a disminuir el tiempo y a aumentar la satisfacción del estudiante. Está fundamentada en la necesidad existente de propuestas que posibiliten crear una visión holística que contraste los métodos formales del aprendizaje activo con las metodologías ágiles y el pensamiento creativo, logrando ser eficaces en la identificación de las necesidades de los estudiantes.

El documento está estructurado de la siguiente forma: en la siguiente sección se describen los elementos del marco teórico en los que se sustenta el modelo propuesto. A continuación, se describe el Modelo y luego se discuten los resultados alcanzados con su aplicación en un curso real. Finalmente se presentan las conclusiones y los trabajos futuros.

## **2 Marco teórico – metodológico**

### **2.1 El e-learning**

Según la conceptualización amplia de educación a distancia, el e-learning, es educación a distancia, al basarse en un diálogo didáctico mediado entre el profesor (institución) y el estudiante que, ubicado en espacio diferente al de aquel, aprende de forma independiente y también colaborativa (García Aretio, 2001).

### **2.2 SCRUM**

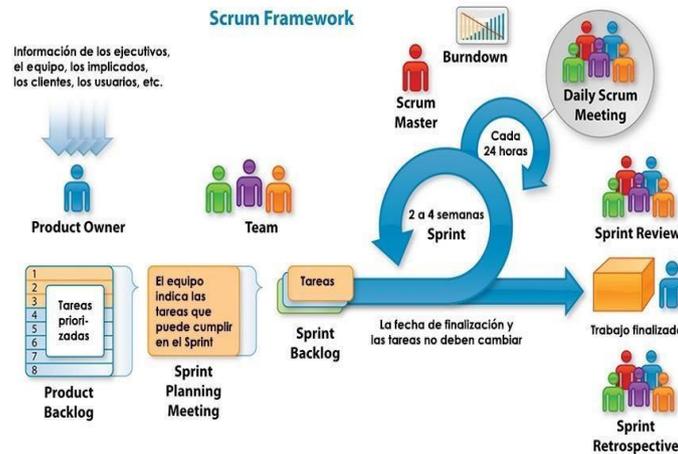
Considerado como el método ágil más popular, Scrum es un framework iterativo e incremental para proyectos y productos o desarrollo de aplicaciones [7], que se desarrolla en ciclos de trabajo llamados Sprints. Scrum no es un proceso o técnica para construir productos; sino como ilustra la Figura 2, es un marco de trabajo dentro del cual se pueden emplear varias técnicas y procesos [8].

Scrum consta de 3 elementos fundamentales para su desarrollo, el Scrum Team, los Eventos de Scrum, y los Artefactos de Scrum.

#### **Scrum Team**

*Dueño del Producto (Product Owner):* Es el único responsable (persona) de gestionar el Product Backlog, y sus decisiones deben ser respetadas por la organización.

*El Equipo (The Team):* Los equipos son autogestionados, auto-organizados y multifuncionales, y son los responsables de encontrar la manera de convertir el Product Backlog en un incremento de funcionalidad al interior de cada iteración gestionando su propio trabajo a ser realizado. Los miembros del equipo son responsables del éxito de cada iteración y del proyecto en conjunto [8]. El tamaño óptimo de El Equipo debe estar entre 3 y 9 personas, que es un grupo lo suficientemente pequeño como para permanecer ágil, y lo suficientemente grande como para completar una cantidad de trabajo significativa.



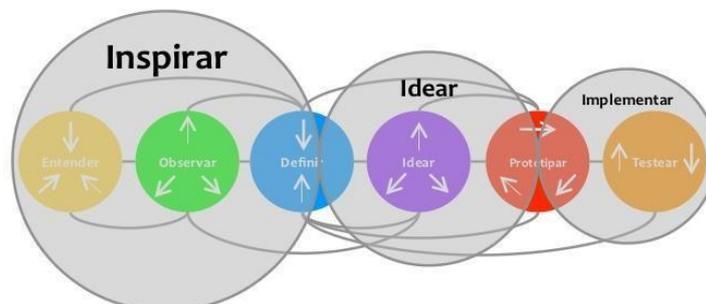
**Figura 1:** Marco de trabajo de SCRUM. Fuente: [7].

*Scrum Master:* El Scrum Master es el responsable de asegurar que Scrum se aplique y sea entendido de la manera adecuada y está al servicio del Scrum Team para garantizar que éste trabaje ajustado a la teoría, prácticas y reglas de Scrum. Además de maximizar el valor creado por el Scrum Team por medio de modificaciones de dichas interacciones [8].

### 2.3 Design Thinking

El Design Thinking es un método de trabajo que sirve para generar ideas disruptivas tomando en cuenta siempre las necesidades explícitas y latentes del cliente o usuario final [2]. Se trata de un cambio del pensar a la acción [3] y a diferencia de otros enfoques, el Design Thinking invita al cliente a tener un rol más activo en el diseño del producto o servicio. De esta forma se garantiza que cada cambio que se haga a la idea sea basado en cumplir sus necesidades. Esto implica que los involucrados dialoguen mutuamente en empatía, con lo que espera recibir (el cliente) y lo que se espera diseñar (equipo de proyecto).

Mediante este proceso iterativo, los equipos pueden adquirir una nueva percepción a partir de la observación continua y la elaboración de prototipos, y en ocasiones pueden llegar a replantearse el problema de una manera completamente nueva. La riqueza del método se muestra en la etapa de validación ya que es en este punto donde el cliente, mediante una participación activa, evalúa los prototipos exhibidos y acepta aquellos aspectos positivos del mismo; además, rechaza aquellos aspectos negativos y propone cambios que enriquecen el diseño. Como resultado final, se obtiene un producto/servicio que satisface las necesidades latentes del cliente de manera eficaz.



**Figura 2:** Proceso del Design Thinking. Fuente: [2].

## 2.4 Plataforma LMS y Moodle

Las plataformas LMS (*Learning Management System*) es un sistema de gestión de aprendizaje online, que permite administrar, distribuir, monitorear, evaluar y apoyar las diferentes actividades previamente diseñadas y programadas dentro de un proceso de formación completamente virtual (*eLearning*), o de formación semi-presencial (*Blended Learning*).

Su conceptualización está orientada a que sea fácilmente accesibles, amigables, intuitivos y flexibles, permitiendo ser utilizados tanto por los administradores, coordinadores y formadores, mientras se disponga de conexión a Internet. Por otro lado, también potencian de forma destacable la interacción online entre todos los agentes implicados dentro de un proceso de aprendizaje con componente *online*.

## 2.5 Moodle

El Moodle es un sistema para el Manejo del Aprendizaje en línea gratuito, que les permite a los educadores la creación de sus propios sitios web privados, llenos de cursos dinámicos que extienden el aprendizaje, en cualquier momento, en cualquier sitio.

## 2.6 Aprendizaje Activo

Es un aprendizaje basado en la implicación, motivación, atención y trabajo constante del alumno, es decir, el estudiante NO es un pasivo que se limita a escuchar al profesor y a tomar apuntes, sino que es un sujeto activo que es responsable directo de su aprendizaje.

Sin embargo, según lo planteado por [2] esto no significa que el profesor pase a un segundo plano, por el contrario, el rol del profesor en este proceso adquiere una gran relevancia, puesto que es el encargado de guiar y orientar a los alumnos para alcanzar ciertos objetivos de aprendizaje específicos.

En un extensivo estudio llevado a cabo por la consultoría Gallup en el que participaron cerca de 80.000 estudiantes, los investigadores descubrieron que un aumento del 1% en su índice de interacción equivalía a un aumento del 6% en los resultados académicos de los alumnos y hasta un 8% de aumento en sus resultados en el área de matemáticas. Esto pone de manifiesto la relación que existe entre la participación de los alumnos y sus resultados.

### 3 Métodos

El presente trabajo está enfocado en definir un modelo que fomente el aprendizaje activo a través de plataformas LMS, pero no con un enfoque tradicional, más bien facilitando que sea ágil, y que a través de la Taxonomía de Bloom se logren utilizar los verbos apropiadamente.

La plataforma LMS es el primer componente del modelo, el cual ofrece el entorno base en el cual el estudiante va a interactuar con las actividades y los recursos. Las primeras están orientadas a las acciones que se plantean para cumplir con los objetivos que se definieron en un curso por ejemplo una tarea, un foro, una encuesta, un cuestionario, entre otros. Con respecto a los recursos son todas las herramientas que tiene la plataforma para poder apoyarse en la realización de las acciones definidas. Estas pueden ser Archivo, libro, página, entre otras.

El segundo componente del modelo es el Aprendizaje Activo, el cual facilita una interacción directa con el estudiante, donde esta toma un papel protagónico y relevante en la formación que recibe, ya que participa activamente en todo el proceso y los resultados finales inciden hacia él.

El tercer componente es el Design Thinking, el cual es un método que le facilita al estudiante a través del diseño de su pensamiento poder construir los conocimientos e ideas que se plantea a lo largo de su aprendizaje.

Es un método para generar ideas innovadoras que centra su eficacia en entender y dar solución a las necesidades reales de los usuarios. Proviene de la forma en la que trabajan los diseñadores de producto. De ahí su nombre, que en español se traduce de forma literal como "Pensamiento de Diseño".

Se empezó a desarrollar de forma teórica en la Universidad de Stanford en California (EEUU) a partir de los años 70, y su primera aplicabilidad con fines lucrativos como "Design Thinking" la llevó a cabo la consultoría de diseño IDEO, siendo hoy en día su principal precursora, según [2].

El cuarto y último componente es el Enfoque Ágil el cual facilita mediante un proceso iterativo, la elaboración de prototipos los cuales a través de un método ordenado de ejecución ayuda a los equipos de personas a poder obtener resultados más eficientes en el corto plazo y cumplir con los entregables.

Los cambios que se utilizan para aprovechar e innovar en el aula la potencialidad de los estudiantes universitarios, son necesarias para que por medio de las tecnologías educativas que componen el modelo se produzcan grandes ventajas competitivas.

Los métodos principalmente van a facilitar el uso del modelo y determinan como se puede cerrar la brecha con respecto a los modelos tradicionales de aprendizaje.

## 4 Propuesta de solución

El modelo propuesto que se plantea se basa en la combinación de metodologías ágiles y activas como Aula Invertida, Design Thinking. Primeramente, mediante esta interrelación de actividades se ha logrado plantear la propuesta de solución que facilita la realización entre las herramientas indicadas de forma activa y práctica. Por ejemplo, la Ingeniería de Requerimientos, inicialmente se establece una lluvia de ideas, luego mediante el prototipado del Design Thinking se recopila la información y luego de esto se transfiere a una herramienta que está en la Plataforma LMS (Moodle), Para una mejor comprensión, se agrupan las actividades a realizar en tres procesos fundamentales basado en mejores prácticas como muestra la Figura 4.

El primero explica como a través de la Taxonomía de Bloom y sus múltiples clasificaciones podemos Aplicar, A analizar, Evaluar y Crear. Luego la interacción con los verbos en cada uno de los clasificadores anteriores.

### TAXONOMIA DE BLOOM Y DESIGN THINKING, FOMENTANDO EL APRENDIZAJE ACTIVO



Figura 3: Estructura del Modelo Propuesta. Fuente: Elaboración Propia.

A continuación, serán descritos cada uno de los procesos propuestos:

*Definir y Priorizar tareas:* En este proceso mediante entrevistas con el cliente se describen sus necesidades que luego son transformadas en requerimientos de software. Es importante en esta etapa identificar elementos propios del entorno, así como características de los usuarios potenciales del software a desarrollar. Estos elementos contribuirán a ejecutar satisfactoriamente el proceso siguiente.

En resumen, se propone identificar las necesidades del usuario y comprender el problema (Entender).

Seguidamente este listado de necesidades identificadas es transformado a requerimientos, los cuales serán redactados de una forma clara y precisa evitando ambigüedades. La lista con los requerimientos obtenidos, es priorizada teniendo en cuenta cada una de las tareas propuestas, las cuales conforman el producto de trabajo comúnmente conocido como Product Backlog.

*Diseño y Prototipado:* Se puede aseverar que este es uno de los procesos más importantes, ya que luego de contar con la lista de los requerimientos, éstos se deben comenzar a diseñar. La etapa de diseño es una etapa que se compone de varias actividades importantes. En primer lugar, es necesario clasificar los requerimientos identificados, así como otorgarles un grado de prioridad como parte de su proceso de evaluación.

Normalmente para el diseño no existe un patrón a seguir ni una forma específica de realizarlo. En el caso de la guía que se propone y aprovechando las ventajas del Design Thinking se utilizan inicialmente bocetos en papel los que luego son transformados en mockups, que representan el diseño de las funcionalidades y son utilizados para la demostración, evaluación del diseño, promoción, y para otros fines.

Es importante recalcar que cuando se obtienen los requerimientos, éstos se orientan a verse en pantallas con el objetivo de visualizar la forma en cómo se verán los diseños, identificando datos de entrada que luego serán implementados en el front end del software.

Luego estos bocetos son transformados haciendo uso de herramientas que facilitan el proceso de prototipado.

Los prototipos generados con esta herramienta son fáciles de entender y muy atractivos para el usuario. Este último elemento ayuda mucho durante la validación de los requerimientos. La salida de este proceso constituye el punto de partida para la implementación de los prototipos por el Analista-Programador en la tecnología seleccionada.

Con la ejecución de cada uno de los procesos descritos se garantiza la mejora continua como parte del conjunto de acciones dirigidas a obtener productos de software con calidad.

## **5 Resultados y discusión**

Luego de aplicar el modelo a un conjunto de estudiantes de investigación para un proyecto terminado de forma continua se visualizaron una serie de resultados interesantes de acuerdo con lo planteado en la propuesta.

Inicialmente se establece implementar el uso de una plataforma de libre distribución como herramienta tecnológica para reducir en la medida de lo posible la brecha que provoca la transformación digital. Asimismo, aplicar estrategias y resolver los problemas que se presenten cuando se utiliza el contenido de las prácticas educativas establecidas y diseñadas a través del Pensamiento de Diseño.

Es necesario ir desarrollando habilidades y actitudes en el uso del modelo para cumplir con la estructura y cada uno de sus componentes.

El principal resultado es establecer una guía que facilite la integración de los diferentes componentes.

Para demostrar la efectividad del modelo se realizó un plan piloto en el Área de toma de requerimientos, como parte del desarrollo de cada uno de los procesos se identificaron inicialmente las necesidades del cliente que se transformaron en un total de 45 requerimientos funcionales.

Se planificaron reuniones con frecuencia semanal con la siguiente dinámica:

1. Inicialmente se determina mediante una lluvia de ideas la lista de actividades que se requieren. *Comprensión y Empatía.*
2. Luego se prioriza las actividades para determinar los requerimientos que, según cronograma se abordarán primero y cuales después. *Se plantea un caso o problema.*
3. Se empieza a diseñar el prototipo de como ese requerimiento se vería, mediante los bocetos en papel. *Se establece una posible solución.*
4. Ese prototipo se valida con todos los participantes y los acuerdos se documentan mediante una minuta. *Diseña la solución.*
5. El boceto aprobado, es desarrollado en la herramienta Balsamiq Mockup por el analista y el prototipo resultando es entregado al programador para su implementación y evaluación. *Revisa y evalúa la solución.*
6. El programador presenta el prototipo funcional del requerimiento.
7. Informe resumen de estado del desarrollo.

Para dar cumplimiento al proceso de Control y seguimiento y con ello al punto 7 de las reuniones con el equipo de trabajo, se asignó un especialista en QA quien les dio seguimiento a los requerimientos. Este seguimiento se inicia con las listas de chequeo de lo que debe contener cada una de las tareas descritas y los responsables de éstas. Este informe resumen contiene los principales resultados de los avances, lo pendiente y en lo que se está trabajando. En el Apéndice 1 se muestran algunas imágenes como parte de la implementación de la guía.

Como principales resultados alcanzados se tiene un aumento de la productividad, la usabilidad y la mantenibilidad de las interfaces de usuarios resultantes. En cuanto al aumento de la productividad puede mencionarse que, con el diseño de los mockups, además de requerir poco tiempo, se posibilita recoger de una forma atractiva para el usuario los elementos del diseño que dan cumplimiento a cada requerimiento planteado. Así se disminuyen la probabilidad de ocurrencia de peticiones de cambio durante el ciclo de vida del proyecto, que al final se reflejan en inversión de tiempo.

Se aprecia también una mejora considerable en la usabilidad resultante, pues al aplicar en la guía elementos del diseño centrado en el usuario y del diseño de la experiencia de usuario, se crea un estado emocional placentero en la interacción con

el sistema. Por ejemplo, realizar un análisis de tareas permitió que la interfaz que se modele sea lo más similar posible a como se realizan en la práctica las tareas que se informatizaron.

Con respecto a la mantenibilidad, se obtienen resultados alentadores pues las interfaces que se obtienen son menos complejas y más modulares.

## 6 Conclusiones y Trabajo futuro

La revisión de los principales elementos asociados a la IR, así como de SCRUM, Design Thinking y Kanban permitió valorar la importancia y necesidad de su integración en el desarrollo de software ágil para la implantación exitosa y satisfaccón de los usuarios finales. Además, se pudo constatar que, si bien las metodologías tradicionales proponen un proceso formal para la IR, resulta complejo adoptar estas prácticas en proyectos que se desarrollen siguiendo un enfoque ágil.

A partir de los elementos teóricos estudiados se elaboró una guía metodológica para la Ingeniería de Requerimientos siguiendo un enfoque ágil. Durante su definición se tuvo en cuenta el resultado de la revisión realizada y como elementos más importantes que la guía incluye se encuentran los procesos a realizar, así como una propuesta de tecnologías a utilizar para alcanzar mejores resultados.

Para demostrar su nivel de aplicabilidad se describieron los resultados alcanzados luego de su aplicación en el desarrollo de un proyecto real, que demuestra mejoras en la productividad, usabilidad y mantenibilidad del sistema desarrollado.

## Referencias

1. Micheli, P., Wilner, S. J., Bhatti, S. H., Mura, M., & Beverland, M. B. 2018. Doing Design Thinking: Conceptual Review, Synthesis, and Research Agenda. *Journal of Product Innovation Management*. Brown, T. 2008.
2. Design Thinking. *Harvard Business Review*, 86(6), p 84-92. Brown, T., & Wyatt, J. 2010.
3. Building creative competence in globally distributed courses through design thinking. *Revista Comunicar*, 19(37), 27-34. Bartolomé Pina, Antonio, Blended learning. *Conceptos básicos*. Pixel-Bit. *Revista de Medios y Educación* 2004.
4. Ibáñez, J. S., de Benito Crosetti, B., Garcies, A. P., & Cervera, M. G. Blended learning, más allá de la clase presencial. *RIED. Revista Iberoamericana de Educación a Distancia*, 21(1), 195-213, 2018.
5. <https://www.goconqr.com/es/ensenar/aprendizaje-activo/> consultado el 20 de Octubre del 2019.
6. <http://www.designthinking.es/inicio/> consultado el 11 de Noviembre del 2019.
7. K. Schwaber and J. Sutherland, "La guía de Scrum", *Scrumguides. Org*, vol. 1, p. 21, 2013
8. Trujillo, H. M., Chávez, M. M. P., & Zermeno, M. G.G. Uso de plataformas de libre distribución (LMS) para educación básica. *Revista Iberoamericana*, 2016.
9. Design thinking for social innovation. *Development Outreach*, 12(1), 29-43. Steinbeck, R. 2011.



## **Instrucciones para Autores**

*Enseñanza y Aprendizaje de Ingeniería de Computadores (Teaching and Learning Computer Engineering)* es una revista de Experiencias Docentes en Ingeniería de Computadores que edita el Departamento de Arquitectura y Tecnología de Computadores de la Universidad de Granada, se publica anualmente, y se difunde tanto en papel como electrónicamente, a través del repositorio institucional de la Universidad de Granada (<http://digibug.ugr.es/>).

Los artículos remitidos para su evaluación pueden estar escritos en castellano o inglés, incluyendo un resumen y palabras clave en inglés en caso de que estén escritos en castellano, y deben seguir el formato descrito en la dirección web:

[http://atc.ugr.es/pages/actividades\\_extension/](http://atc.ugr.es/pages/actividades_extension/)

El correspondiente fichero .pdf debe enviarse a la dirección de correo electrónico [jortega@ugr.es](mailto:jortega@ugr.es) o [mdamas@ugr.es](mailto:mdamas@ugr.es)

Los artículos deben abordar, tanto contenidos relacionados con la docencia universitaria en general, como con la docencia de asignaturas específicas impartidas por las áreas de conocimiento involucradas en estudios relacionados con la Ingeniería de Computadores, y también pueden aspectos relativos a las competencias profesionales y la incidencia de estos estudios en el tejido socio-económico de nuestro entorno.

En particular, se anima a antiguos alumnos de los estudios de Informática y a estudiantes de grado y posgrado a que envíen colaboraciones relacionadas con sus experiencias al cursar asignaturas relacionadas con la Ingeniería de Computadores, sugerencias, propuestas de mejora, etc.



# Teaching and Learning Computer Engineering

Journal of Educational  
Experiences on Computer  
Engineering

December 2019, Number 9

