

Recurso Didáctico Basado en Juegos para el Refuerzo del Aprendizaje de Programación en Ingeniería para Dispositivos Móviles

A. J. Sierra, T. Ariza, F. J. Fernández-Jiménez, J. Muñoz-Calle, A. Molina, A. Martín-Rodríguez

Departamento de Ingeniería Telemática

Universidad de Sevilla

Sevilla, España

{antonio,mater,fjf,javi}@trajano.us.es; alfonso7193@gmail.com; alvmarrod@alum.us.es

Abstract—Los recursos didácticos son componentes esenciales en la educación, y se han mejorado y evolucionado mediante las tecnologías de la información y la comunicación de uso cotidiano, tales como los dispositivos móviles. La gran cantidad de tiempo dedicado a este dispositivo hace que sea un medio bastante atractivo para proponer aplicaciones que sirvan como refuerzo a los contenidos teóricos impartidos en las asignaturas. En este artículo se muestra la implementación de un recurso didáctico basado en juegos para el refuerzo del aprendizaje de Programación en Ingeniería para dispositivos móviles Android. Este recurso sirve como apoyo a la enseñanza del lenguaje de programación Java de una forma intuitiva y cómoda.

Keywords— *Programming languages; Engineering Teaching; Recursos educativos; Aplicaciones móviles; Game-based learning*

I. INTRODUCCIÓN

Los recursos educativos utilizados en el aprendizaje están constantemente cambiando para mejorar y adaptarse a los nuevos medios técnicos. Estos recursos son especialmente atractivos para el alumno si se realizan mediante juegos. El interés en la metodología de aprendizaje basada en juegos (GBL, Game-Based Learning) de ordenador ha ido creciendo a medida que la investigación continúa demostrando su eficacia para la educación [1][10][13]. Los juegos representan una herramienta eficaz para complementar los métodos convencionales de enseñanza. La mayoría de los estudiantes han usado juegos de ordenador por lo que les resulta una herramienta cercana. El uso de juegos permite capturar el interés del estudiante, estimulando su curiosidad y creándole la impresión de controlar su propio aprendizaje. Los estudiantes participan en el proceso de aprendizaje, fortaleciendo sus habilidades cognitivas y mejorando su nivel de satisfacción con un cierto sentido de logro. Ello conlleva un intrínseco aumento de la motivación del estudiante, facilitándole el aprendizaje [3][8].

En ingeniería, los conocimientos en programación son necesarios para múltiples asignaturas, por lo que son impartidos en los primeros cursos. El dominio de la programación necesita un alto nivel de abstracción, sin conexión directa con las tareas del mundo real, lo que representa un obstáculo para muchos estudiantes. Además, la

programación implica la adquisición de habilidades prácticas tras un trabajo de análisis, diseño, escritura de código y experimentación [24], esfuerzo que supone un nivel de dificultad importante. El uso de los juegos para el aprendizaje de la programación se basa en la visualización del código del programa, ayudando a los estudiantes a entender la idea de los paradigmas de programación. Ejemplos de este tipo de juegos son los mostrados en [2][7][22].

En los últimos años, las mejores prestaciones de los terminales móviles, con el consiguiente aumento del mercado de aplicaciones, ha provocado un incremento exponencial del número de usuarios móviles, superando en la actualidad al de usuarios fijos. Con objeto de aprovechar este interés de los alumnos en las aplicaciones móviles, la aplicación mostrada en el artículo ha sido inicialmente desarrollada para trabajar sobre sistemas Android, el cual representa en torno al 80% del mercado móvil [27].

El recurso docente realizado en este trabajo sirve como soporte para las asignaturas de Fundamentos de Programación de primer curso en el Grado de Ingeniería de Tecnologías de Telecomunicación (GITT) implantado bajo el Espacio Europeo de Educación Superior (EEES) [23].

En este artículo se muestra la implementación de un recurso didáctico basado en Juegos para facilitar el aprendizaje de Programación en Ingeniería de una forma intuitiva y cómoda. El jugador deberá diseñar programas con el objetivo de completar diferentes “misiones” que aumentan progresivamente su dificultad. Una vez escrito el código, este será ejecutado por pantalla, línea a línea, mostrando su funcionamiento visualmente. Con ello, el usuario aprende las bases de un lenguaje de programación y se acostumbra a diseñar programas de una forma estructurada. La herramienta ha sido desarrollada para facilitar el aprendizaje de la Programación en lenguaje Java impartido en la asignatura de Fundamentos de Programación II (FP2) de primer curso, y además ha sido diseñada para que pueda ser fácilmente adaptada a otros lenguajes.

En los siguientes apartados, en primer lugar, se expone una panorámica del aprendizaje basado en juegos. A continuación, se detallan los objetivos de la aplicación desarrollada. Después se describe el recurso docente para el refuerzo del aprendizaje. Finalmente, se presentarán las conclusiones.

II. PANORÁMICA DEL APRENDIZAJE BASADO EN JUEGOS

Los juegos de ordenador han demostrado ser una herramienta eficaz para complementar los métodos convencionales de enseñanza. En este apartado se realiza, en primer lugar una panorámica del aprendizaje basado en juegos y posteriormente se muestra diferentes propuestas realizadas para el uso de juegos en la programación.

A. Aprendizaje Basado en Juegos

En el aprendizaje basado en juegos (GBL, Games Based Learning) [18][19] los jugadores son implicados en diversas actividades de aprendizaje, usualmente por medio de videos educativos o de juegos serios. Estos juegos están diseñados para promover la participación activa y la interacción como centro de la experiencia, persiguiendo una mejora en determinadas capacidades o conocimientos del jugador en lugar de en su mero entretenimiento. Los alumnos adquieren nuevos conocimientos y habilidades complejas a través del juego, siendo este una herramienta de apoyo al proceso de aprendizaje. Por este motivo, cada vez se aprecia una mayor intensidad en el uso de juegos de ordenador para la formación en la educación superior, empresas e industrias.

El uso de juegos para mejorar el aprendizaje en programación ha sido explorado en la literatura reciente [15]. Son muchos los argumentos pedagógicos para apoyar el uso de juegos en la enseñanza en general, y en la programación en particular. Hirumi [10] afirma que la eficacia de los juegos en la educación se debe a su capacidad para “enganchar” a los estudiantes. Los juegos representan un “potente entorno de aprendizaje”, motivando a los estudiantes a través de los factores de diversión, retos y retroalimentación instantánea dentro de un entorno virtual visual e interactivo. En los juegos de ordenador, los jugadores participan en procesos tales como la anticipación, el pensamiento proactivo, la organización de la información y la generación de vías de solución alternativas.

El aprendizaje basado en juegos puede ser combinado con otras metodologías de aprendizaje, tales como el aprendizaje basado en la colaboración (CBL) [26], el basado en problemas (PBL) [34, 35] o el basado en proyectos (PBL) [4].

B. Juegos para el aprendizaje

La tecnología del juego ha sido usada extensamente para apoyar el aprendizaje de los lenguajes de programación. Uno de los enfoques preferidos es el uso de las actividades relacionadas con el diseño del programa. Los estudiantes crean fragmentos de código que controlan personajes y objetos en el entorno del juego. El resultado visual de la tarea realizada logra que el estudiante rápidamente se enganche con el aprendizaje de la programación. Podemos citar múltiples ejemplos de estos juegos, tales como GameMaker [21], Alice [2], Microsoft's Kodu [16] or Scratch [20]. En otro ejemplo, Chen y Cheng [6] utilizan el desarrollo de videojuegos como la actividad principal de su curso de introducción a la programación. Otros estudios han explorado los beneficios de los juegos digitales en los cursos de programación. En muchos de estos juegos los estudiantes no escriben programas, utilizando los videojuegos como meros contenedores de contenidos teóricos acerca de la programación, y como un medio para aumentar la motivación

de los estudiantes. Sin embargo, la falta de programación activa es una limitación de los beneficios que pueden lograr los juegos, ya que es una actividad esencial para aprender a programar. Por ejemplo, [5] propone que el jugador realice diferentes misiones, directamente relacionadas con los conceptos de programación Java.

El uso de un entorno de juego proporciona un aumento significativo de la motivación de los estudiantes. En [15], los autores proponen dos minijuegos para mejorar el aprendizaje de Java mediante el rellenado de espacios en blanco. En [11] se propone un juego para el aprendizaje de lenguaje C mediante rompecabezas, crucigramas y juegos de tiro. De forma similar, [9] plantea un juego de conceptos para el aprendizaje de lenguaje C++. También hay juegos educativos donde los estudiantes necesitan escribir pequeños programas para seguir adelante en el juego o alcanzar una meta específica. En [17] los autores presentan un juego en 3D donde los avatares de los jugadores son controlados usando un subconjunto de instrucciones del lenguaje Logo. Del mismo modo, uno de los mini-juegos propuestos en [5] invita a los jugadores a introducir comandos para guiar al personaje principal a un punto de destino utilizando un lenguaje de programación sencillo. Un enfoque híbrido se describe en [12], en el que se pide a los estudiantes programar un algoritmo con el fin de alcanzar una determinada meta. El juego ejecuta este algoritmo y le da al estudiante una puntuación final, basado en su eficacia.

III. OBJETIVO DEL RECURSO

El objetivo de este trabajo es el desarrollo de un recurso didáctico basado en juegos como refuerzo a los contenidos de programación sobre dispositivos móviles. Este recurso didáctico enseñará los elementos básicos de un lenguaje de programación, utilizando las características propias de los dispositivos móviles para facilitar el aprendizaje.

Aprender un lenguaje de programación, sin tener conocimientos previos de la materia, es difícil. En los grados de ingeniería, la asignatura de programación suele resultar difícil para muchos alumnos, que no han tenido contacto con un lenguaje de programación anteriormente. El objetivo de este recurso docente es ayudar a los alumnos en las primeras fases del aprendizaje, enseñándoles a pensar de forma estructurada y aprendiendo a utilizar bucles y condicione. Para ello el uso de una aplicación interactiva, es el mejor medio para la asimilación de los contenidos.

En este trabajo se implementa un videojuego que aporta un conocimiento básico sobre un lenguaje de programación. En el videojuego, el jugador aprenderá algunas nociones acerca de la programación, y la sintaxis del lenguaje de programación Java.

Para delimitar los conceptos a asimilar de tal forma que sean abordables por el alumno, el videojuego se divide en niveles. Cada nivel tiene un determinado objetivo que se debe cumplir. En cada nivel, el jugador debe escribir unas líneas de código que controlen a su personaje y lleven a este a superar el nivel. Cada nivel tiene componentes de azar y repetición, diseñados para que el usuario incluya en su código estructuras de control.

Otro de los objetivos al desarrollar el videojuego es que el usuario se abstraiga de la sintaxis concreta del lenguaje de programación. Para conseguir este objetivo, el código se escribe utilizando un teclado especial. Este teclado estará compuesto por los diferentes elementos que pueden ser necesarios al programar, y el código se escribe automáticamente, mostrando la sintaxis correcta del lenguaje.

Otro objetivo es la ejecución paso a paso y visual. Para ello, una vez que el código esté escrito, el videojuego mostrará de forma gráfica el desarrollo del mismo. Se mostrará el movimiento del personaje y cómo se recorren los bucles.

El recurso docente se ha diseñado con un objetivo básico, debe ser atractivo estéticamente y fácil de usar. Utilizarlo no debe aportar una dificultad añadida a la de aprender un lenguaje de programación.

Como objetivos específicos en la transmisión de conocimientos del lenguaje de programación Java el videojuego cubre los siguientes conceptos:

- La estructura de clases de Java.
- Objetos y variables.
- Estructuras de control.

Como objetivo final, el diseño de la aplicación debe permitir soporte para varios lenguajes de programación. En esta primera versión se ha incluido el soporte para el lenguaje de programación Java sobre dispositivos móviles Android.

IV. RECURSO DOCENTE PARA EL REFUERZO DEL APRENDIZAJE

En este trabajo se ha desarrollado un videojuego destinado a usuarios con conocimientos mínimos o inexistentes de un lenguaje de programación, que es fácil de jugar, y no deja de lado la parte didáctica. En primer lugar se proporciona una descripción general del recurso desarrollado. A continuación se describe el funcionamiento del mismo, y el desglose en niveles.

A. Funcionalidad del Recurso Desarrollado

En esta sección se va a explicar cómo se utiliza la aplicación desarrollada como recurso didáctico. Se hará un recorrido por las distintas ventanas que el usuario visitará en el transcurso del videojuego, explicando cómo se utilizan y su función en el progreso del juego. Para hacer progresos, el jugador deberá superar niveles (o “misiones”) que, a su vez, se dividen en subniveles (o “etapas”). Según se van superando estos niveles o etapas el jugador va avanzando en la historia, desbloqueando nuevos elementos y consiguiendo experiencia, con la cual podrá desbloquear nuevos niveles. Para superar los niveles, el jugador visitará las distintas actividades de la aplicación. A continuación se explican las actividades de la aplicación.

- Mapa: La actividad principal del videojuego es una vista del mapa. En ella se muestra una ciudad y la lista de misiones que se pueden hacer en ella. Al pulsar en una misión de la lista, aparecerán detalles de la misma, con un botón para ir a ella. Si el jugador ya había entrado en un nivel pero no lo ha completado, se muestra un botón para ir a la última etapa visitada. Las

misiones no desaparecerán una vez realizadas; seguirán presentes para que el jugador pueda volver a jugarlas cuando desee. Aunque las misiones nuevas sí tienen una distinción: aparece un signo de especial (la exclamación) sobre su botón. Una vez que el jugador accede a ese nivel, aunque no lo supere, el signo de exclamación desaparece. Esta actividad también tiene enlaces a las opciones (que permite reiniciar el juego y cambiar de idioma) y al libro de apuntes. Esta actividad se implementa en *MapaActivity*.

- Libro de apuntes: Conforme se avanza en el videojuego se van obteniendo lecciones acerca de distintos elementos de Java, enseñando al jugador a utilizarlos en el juego. Todas estas lecciones son guardadas en el libro de apuntes y pueden consultarse en cualquier momento. Esta actividad se implementa en *ApuntesActivity*.
- Introducción: Cada nivel empieza con dos personajes hablando, a modo de introducción y como contexto para el nivel. El diálogo tiene dos funciones: guiará al jugador acerca de qué código tiene que escribir y avanzará en la trama del videojuego. Esta actividad se implementa en *IntroduccionActivity*.
- Resumen del nivel: En esta actividad se muestra el tablero en el que se realiza la misión y los objetivos (principal y secundario) de la misma. Esta actividad se implementa en *ResumenMisionActivity*.
- Escribir código: El siguiente paso es escribir un fragmento de código necesario para cumplir la misión. El código se escribirá utilizando un teclado especial, en el cual las teclas escriben de forma automática el código utilizando la sintaxis adecuada. Al iniciar el nivel aparece un globo de texto con información del nuevo elemento de programación desbloqueado y lo añade al libro de apuntes. Muchos niveles también incluyen unas líneas preliminares, para guiar la escritura de código la primera vez que se utiliza cada elemento. Esta actividad se implementa en *CrearCodigoActivity*.
- Ver ejecución: Por último se realiza la ejecución del código. En esta actividad se ejecuta una línea de código cada vez que se pulsa la pantalla, mostrando de forma gráfica qué hace el código. Si el jugador le da al botón “Atrás” (al botón en pantalla o al botón físico) irá de vuelta a la actividad “*CrearCodigoActivity*”, donde podrá modificar lo que ha escrito. Esta actividad se implementa en *ResolverCodigoActivity*.

La interfaz de usuario de estas actividades se muestra en la Fig. 1, Fig. 2 y Fig. 3.

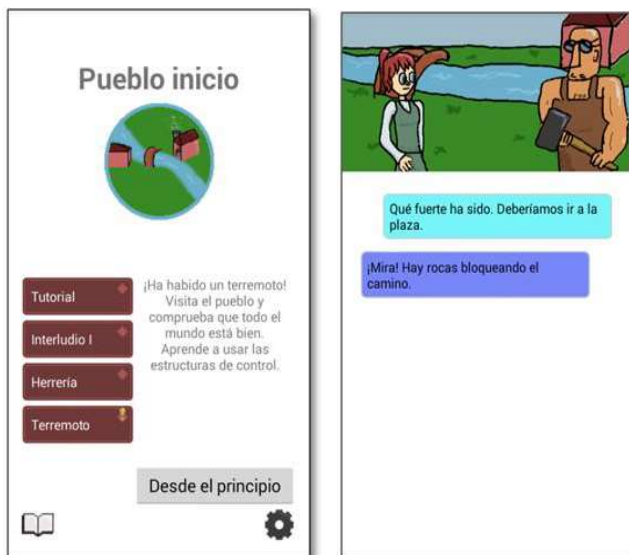


Fig. 1. MapaActivity y IntroduccionActivity.

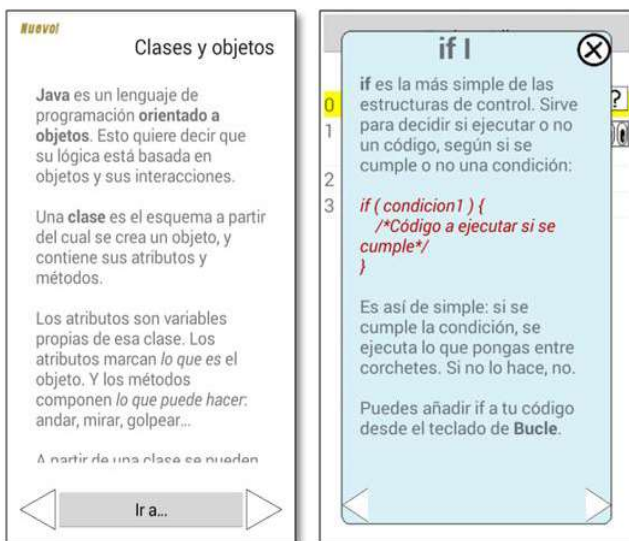


Fig. 2. ApuntesActivity y texto de ayuda.

B. Desglose de niveles

Los diferentes niveles que se han desarrollado para el videojuego y qué herramienta de programación se utilizan en cada una se describen en esta sección.

- Tutorial. Este nivel es necesario para explicar el funcionamiento de la aplicación. Esta parte enseña cómo se utiliza el teclado.
- Interludio I. Este segundo nivel sirve como introducción al lenguaje de programación Java. Explica la existencia de clases y objetos.
- Herrería. Desbloquea algunos métodos nuevos, como “mirar”, aumentando las acciones que puede realizar el personaje.

- Terremoto. Se introducen las sentencias de control. Progresivamente se enseñan “if”, “else” y “else if”.
- Interludio II. Clase de programación que introduce las variables y ejemplifica su uso con un pequeño nivel.
- Asalta Caminos. Se introduce por primera vez un bucle. Se utilizan dos, en dos etapas distintas: while y do-while.
- Asalta Caminos II. No desbloquea ninguna tecla nueva, sino que se pone a prueba la habilidad del usuario en la utilización de bucles y condicionales.
- La historia de Boole. Se enseña la noción de las excepciones. Este nivel también se utiliza para avanzar en la historia del videojuego.
- Torre. Excepciones y Try-catch.

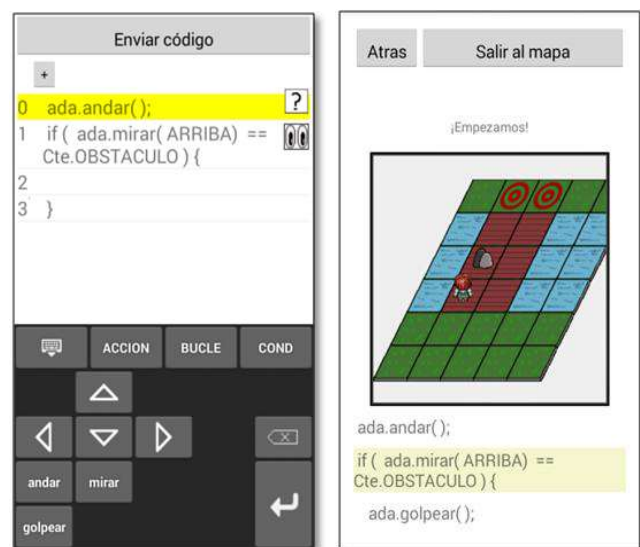


Fig. 3. CrearCodigoActivity y ResolverCodigoActivity.

V. VALIDACIÓN Y VALORACIÓN DEL RECURSO DOCENTE

Para la validación y valoración del recurso docente se ha realizado un cuestionario a los alumnos de Fundamentos de Programación II del GITT. Los objetivos han sido por un lado conocer el tiempo que un alumno utiliza el dispositivo móvil al día, y por otra parte, la opinión de los alumnos con respecto al videojuego. Las preguntas y los resultados obtenidos son los siguientes.

Pregunta 1: ¿Piensa que el uso del móvil (dispositivo móvil o tablet) es una herramienta apropiada para el aprendizaje de los Lenguajes de Programación?

SI	NO
100%	0

Pregunta 2: ¿Cuántas horas (y minutos) al día dedica a su dispositivo móvil? ¿Cuántas horas y minutos dedica al día al PC?

Móvil

Medio	Desviación Estándar
3,796875	1,97319274

PC

Medio	Desviación Estándar
3,625	1,40830868

Pregunta 3: ¿Ha usado algún juego o videojuego para el aprendizaje de algún Lenguaje de Programación?

SI	NO	NS/NC/Nulo
0%	81,25	18,75

Pregunta 4: ¿Cree que el video juego es **fácil de usar**?
Valore de 1 a 5

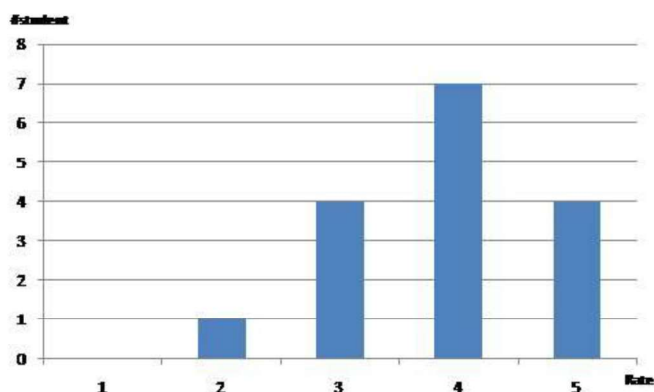


Fig. 4. Resultado de la pregunta 4.

Pregunta 5: ¿Cree que el video juego es **útil**? Valore de 1 a 5

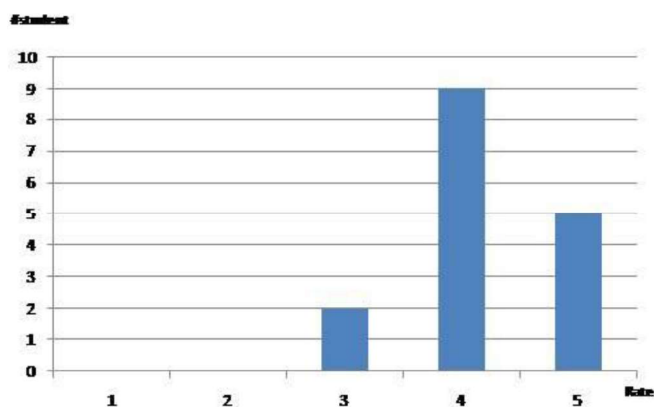


Fig. 5. Resultado de la pregunta 5.

Pregunta 6: ¿Cree que el video juego le puede ayudar a aprobar la asignatura? Valore de 1 a 5

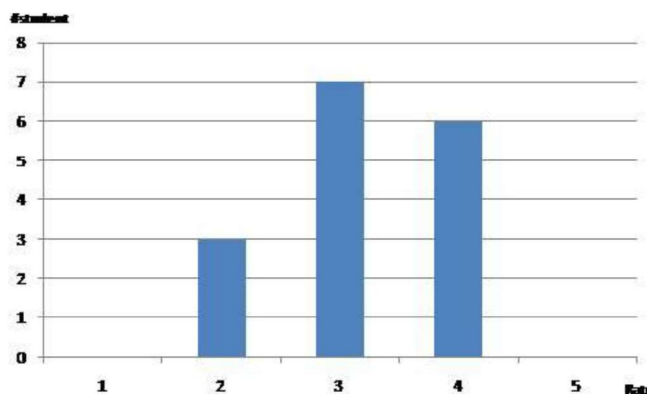


Fig. 6. Resultado de la pregunta 6.

Pregunta 7: ¿Cree que el video juego complementa los contenidos de la asignatura de fp2? Valore de 1 a 5.

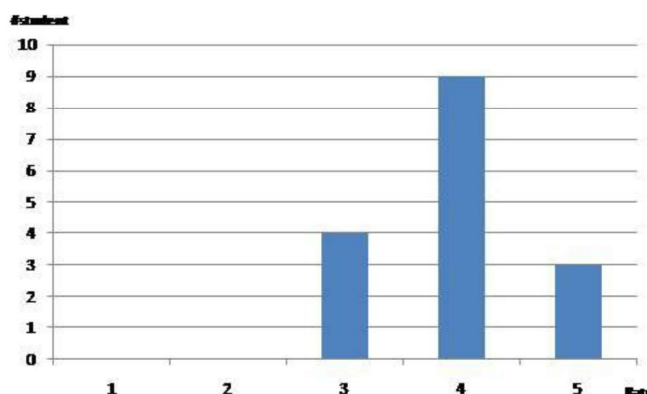


Fig. 7. Resultado de la pregunta 7.

En estos resultados se muestra la opinión del alumno e indican que este recurso puede ayudar a reforzar los contenidos de la asignatura de programación, mediante un videojuego fácil de usar sobre dispositivos móviles.

Conclusiones

En este trabajo se ha desarrollado un nuevo recurso docente para el refuerzo del aprendizaje del lenguaje de programación Java sobre dispositivos móviles basados en Android. Este recurso es un videojuego para aprender a programar en el lenguaje de programación Java, y que, aunque sea fácil de usar, no deja de lado la parte didáctica. Este recurso enseña los conceptos básicos de un lenguaje de programación, y está destinado a usuarios con conocimientos mínimos o inexistentes del mismo.

Agradecimientos

Este trabajo ha sido financiado por la Universidad de Sevilla en el II Plan Propio de Docencia.

References

- [1] E. Adams, *Fundamentals of Game Design*. Second Edition. New Riders : Berkeley, CA, 2010.
- [2] Alice Team of Carnegie Mellon University, "Alice 3", www.alice.org (accessed on 18/03/2016).
- [3] S. Bergin & R. Reilly, The influence of motivation and comfort-level on learning to program. In *Proceedings of the 17th Workshop on Psychology of Programming – PPIG'05*, 2005, 293–304.
- [4] S. Boss & J. Krauss, *Reinventing project-based learning: Your field guide to real-world projects in the digital age*. Eugene, OR: International Society for Technology in Education, 2007.
- [5] M. Chang, "Web-Based Multiplayer Online Role Playing Game (MORPG) for Assessing Students' Java Programming Knowledge and Skills," in *2010 Third IEEE International Conference on Digital*.
- [6] W.-K. Chen and Y. C. Cheng, "Teaching Object-Oriented Programming Laboratory With Computer Game Programming," *IEEE Transactions on Education*, vol. 50, no. 3, pp. 197–203, Aug. 2007.
- [7] S. Cooper, W. Dann, and R. Pausch, "Teaching object-first in introductory computer science" *Proc. of the 34th SIGCSE technical symposium on Computer science education*, pp. 191-195, ACM Press, 2003, ISBN 1-58113-648-X.
- [8] R. Garris, R. Ahlers, & J. Driskell, Games, motivation and learning: A research and practice model. *Simulation and Gaming*, 33(4), 441–467, 2002.
- [9] S. H. A. Hamid, and Fung L. Y., "Learn Programming by Using Mobile Edutainment Game Approach," in *2007 First IEEE International Workshop on Digital Game and Intelligent Toy Enhanced Learning (DIGTEL'07)*, 2007, pp. 170–172. N. Masso and L. Grace, "Shapemaker: A game-based introduction to programming," in *2011 16th International Conference on Computer Games (CGAMES)*, 2011, pp. 168–171.
- [10] A. Hirumi & C. Stapleton, Integrating Fundamental ID Tasks with Game Development Processes to Optimize Game-Based Learning. In C. Miller (ed). *Games: Their Purpose and Potential in Education* (pp. 127-160). New York: Springer Publishing, 2008.
- [11] R. Ibrahim, J. Semarak, K. Lumpur, and A. Jaafar, "Using educational games in learning introductory programming: A pilot study on students' perceptions," in *2010 International Symposium on Information Technology*, 2010, pp. 1–5.
- [12] H. C. Jiau, J. C. Chen, and K.-F. Ssu, "Enhancing Self-Motivation in Learning Programming Using Game-Based Simulation and Metrics," *IEEE Transactions on Education*, vol. 52, no. 4, pp. 555–562, Nov. 2009.
- [13] D. Kumar, Pedagogical dimensions of game playing. *ACM Intelligence Magazine*, 10 (1), 2000 .
- [14] M. Kölling, "Teaching Object Orientation with the Blue Environment", *Journal of Object-Oriented Programming*, 2004, Vol. 12 No. 2, pp.14-23.
- [15] T. Mitamura, Y. Suzuki, and T. Oohori, "Serious games for learning programming languages," in *2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2012, pp. 1812–1817.
- [16] M. B. MacLaurin, "The design of kodu: a tiny visual programming language for children on the Xbox 360," *ACM Sigplan Notices*, vol. 46, no. 1, pp. 241–245, 2011.
- [17] I. Paliokas, C. Arapidis, and M. Mpimpitsos, "PlayLOGO 3D: A 3D Interactive Video Game for Early Programming Education: Let LOGO Be a Game," in *2011 Third International Conference on Games and Virtual Worlds for Serious Applications*, 2011, pp. 24–31.
- [18] H. Pillay, An investigation of cognitive processes engaged in by recreational computer game players: Implications for skills of the future. *Journal of Research on Technology in Education*, 34 (3): 336-351, 2002.
- [19] A. E. Rais, S. Sulaiman, and S. M. Syed-Mohamad, "Game-based approach and its feasibility to support the learning of object-oriented concepts and programming," in *2011 Malaysian Conference in Software Engineering*, 2011, pp. 307–312.
- [20] M. Resnick, J. Maloney, A. Monroy-Hernández, N. Rusk, E. Eastmond, K. Brennan, A. Millner, E. Rosenbaum, J. Silver, B. Silverman, and Y. Kafai, "Scratch: Programming for all," *Communications of the ACM*, vol. 52, no. 11, pp. 60–67, 2009.
- [21] J. Robertson and J. Good, "Story creation in virtual game worlds," *Communications of the ACM*, vol. 48, no. 1, pp. 61–65, 2005.
- [22] D. Sanders, and B.D. Jerroo, "A tool for introducing object-oriented programming", *SIGCSE Technicak Symposium on Computer Science education*, 2003, pp. 201-204.
- [23] A. J. Sierra, T. Ariza, F.J. Fernández-Jiménez, "Establishment EHEA for telecommunication technologies engineering degree", *Global Engineering Education Conference (EDUCON)*, 2012, pp 1-6.
- [24] A. J. Sierra, T. Ariza, F.J. Fernández-Jiménez, *PBL in Programming Subjects at Engineering*, *Bulletin of the IEEE Technical Committee on Learning Technology*, Volume 15, Number 2, April 2013.
- [25] A. J. Sierra, T. Ariza, F.J. Fernández-Jiménez, G. Madinabeitia, J. Muñoz-Calle, A. Martínez-Gotor, *PBL e-assessment Client/Server Tools at Programming Subjects for Engineering*, *ICERI2014 Abstracts* ISBN: 978-84-617-2485-7, *Proceedings* ISBN: 978-84-617-2484-0. ICERI2014.
- [26] R.E. Slavin, *Cooperative Learning*. *Review of Educational Research*, Vol. 50, No. 2, 315-342, 1980.
- [27] G. Twohig, "Mobile Software Statistics 2015", <https://mobiforge.com/research-analysis/mobile-software-statistics-2015> (accessed on 25/04/2016).