

Usabilidad de un simulador para la enseñanza de la Programación de Sistemas

Enrique Ayala Franco, Víctor Hugo Menéndez Domínguez, Luis Fernando Curi Quintal,
María Enriqueta Castellanos Bolaños

Facultad de Matemáticas, Universidad Autónoma de Yucatán
Periférico Norte Tablaje 13615 C.P. 97119, Mérida, Yucatán, México

{enrique.ayala, mdoming, cquintal, enriqueta.c}@correo.uady.mx

Abstract

Simulators, as educational technology tools, facilitate significant learning in students, since they can perform, in a safe and unlimited way, processes or modeled phenomena for their study, meanwhile conditions of execution of the factors can be modified in a controlled manner. Simulators have been used as tools for teaching Computer Science courses, especially to analyze processes inside the computer or between computers, allowing understanding abstract concepts alternatively. In addition, they have served as a platform for the development of practices in computer architecture design or in computer programming with low-level languages. The objective of the present work is to analyze the degree of utility and usability of a simulator, developed by the professor of the course and his students, for teaching systems programming. Usability was measured by two standardized questionnaires applied to students. Results indicate a good acceptance of the simulator used as a teaching tool in subjects related to system programming.

Resumen

Los simuladores, utilizados como herramientas de tecnología educativa, facilitan el aprendizaje significativo de los estudiantes al ofrecer escenarios en donde se puede practicar, de forma segura e ilimitada, la realización de procesos o fenómenos modelados para su estudio, pudiendo variar las condiciones de ejecución de éstos de forma controlada. El uso de simuladores para la enseñanza de las ciencias de la computación ha tenido su espacio de desarrollo principalmente en el análisis de los procesos que ocurren al interior de la computadora o entre computadoras, lo cual ha permitido el entendimiento de conceptos difíciles de comprender de otra forma. También, han servido como plataforma para el desarrollo de prácticas de diseño de arquitecturas de cómputo o de programación de computadoras en lenguajes de bajo nivel. El objetivo del presente trabajo es analizar el grado de utilidad y usabilidad de un simulador, desarrollado por el profesor titular de la asignatura y sus estudiantes, para la enseñanza de la programación de sistemas. Para medir la usabilidad se emplearon dos cuestionarios estandarizados. Los resultados indican una buena aceptación del simulador como herramienta didáctica que puede ser empleada en asignaturas relacionadas con la programación de sistemas.

Keywords and phrases: usabilidad, simulador, TAM, SUS, tecnología educativa.

2010 Mathematics Subject Classification: 68N20, 68N25, 68U20, 97Q60, 68T35 (soft)

1 Introducción

La Tecnología Educativa (TE) es un campo de estudio que se encarga del abordaje de todos los recursos instruccionales y audiovisuales en actividades del proceso de enseñanza-aprendizaje; por tal motivo, el número de herramientas tecnológicas se ha multiplicado exponencialmente (actividades digitales de aprendizaje, portafolios, elaboración de blogs, entre otros), diseñadas para dinamizar los entornos escolares y promover la adquisición de nuevas competencias [21].

En el contexto de la educación basada en la teoría constructivista, algunas de las herramientas tecnológicas más prácticas y completas para lograr los objetivos de aprendizaje, fundamentadas en métodos experienciales, son los laboratorios remotos y los simuladores. El aprender-haciendo es un enfoque práctico del aprendizaje en donde los estudiantes son sometidos a ambientes procurando que sean lo más realistas posibles, para que desarrollen muchas de las habilidades necesarias en la era digital [7].

El uso de simuladores en la educación, en diferentes dominios y niveles del conocimiento, ha facilitado el desarrollo de habilidades cognitivas de los estudiantes durante su proceso formativo. La simulación se ha empleado para replicar, emular, sustituir o amplificar una experiencia o proceso real, controlando las variables de los fenómenos que han sido modelados, permitiendo la práctica segura, repetida y deliberada de las actividades de aprendizaje [18]. De esta forma, los simuladores facilitan la construcción de escenarios ideales para dotar al aprendiz de un recurso didáctico fundamentado principalmente en las siguientes características: 1) su papel motivacional, debido a que permite la representación de fenómenos de estudio los cuales captan la atención del estudiante, 2) facilitador de aprendizaje, ya que el estudiante interactúa con el simulador favoreciendo la aprehensión de conocimientos a través del descubrimiento, y 3) su papel reforzador, ya que permite al usuario la aplicación de los conocimientos adquiridos [17].

Como software educativo, un simulador debe contar con algunos de los siguientes rasgos: tiene una finalidad didáctica, emplea la computadora como apoyo para la realización de actividades, es interactivo, facilita la personalización de las tareas permitiendo que los alumnos avancen a su propio ritmo, y es fácil de usar [10].

Con el propósito de facilitar la comprensión de los conceptos vistos en la asignatura “Programación de Sistemas” y la realización de prácticas para el desarrollo de habilidades de diseño y programación de software de base, el profesor titular del curso ha creado diversas aplicaciones de software, con el apoyo de sus estudiantes pertenecientes a la Licenciatura en Ciencias de la Computación de la Universidad Autónoma de Yucatán. Se implementó un simulador de un sistema operativo y herramientas de software de base; dicho software se describe en [4].

El objetivo del presente trabajo es determinar el grado de usabilidad y utilidad del simulador detallado en [4] al utilizarse como recurso didáctico en la enseñanza de la programación de sistemas o temáticas relacionadas.

La usabilidad, de acuerdo con la norma ISO 9241-11, se define como: “el grado de eficacia, eficiencia y satisfacción con la que un entorno o sistema permite alcanzar unos objetivos específicos a los usuarios que utilizan dicho entorno”. En este sentido, la usabilidad es un factor clave que determina la calidad del software y un indicador importante de la aceptación del mismo acorde a los propósitos para los que fue diseñado.

2 Antecedentes

En el ámbito de la enseñanza de las ciencias computacionales, Alnoukari, Shafaamry y Aytouni [2], presentan un conjunto de simuladores utilizados en la enseñanza de algunas disciplinas del área: ingeniería de software, administración de proyectos de software, arquitectura de hardware de computadoras, redes de computadoras, entre otras. Destacan la importancia del software de simulación para comprender fenómenos o procesos que de otra forma sería muy complicado analizar por su naturaleza o por las dificultades para ser observados, por ejemplo, lo que ocurre al interior de una computadora o los procesos de comunicación en una red de computadoras. Se observa que, en el campo de la arquitectura de computadoras, se han desarrollado simuladores para su enseñanza, principalmente para entender las bases de la microprogramación y el manejo de ejecución de instrucciones en microcircuitos.

En el área de redes de computadoras y telecomunicaciones, la empresa Cisco Systems ha desarrollado simuladores como apoyo para la impartición de cursos de la Academia Cisco, disponibles sólo para alumnos matriculados en instituciones participantes en su programa. El simulador *Packet Tracer* es el más destacado por su flexibilidad y capacidad de diseño y emulación de complejos escenarios de redes de computadoras, que facilitan la práctica y comprensión de los conceptos del área [15].

También en esta misma área, Cabero-Almenara y Costas [10] describen su experiencia en el desarrollo de un simulador para la enseñanza de la asignatura “Sistemas Microinformáticos y Redes”. Diseñan el recurso con un enfoque constructivista y evalúan sus resultados aplicando encuestas de diseño propio para conocer la opinión de los alumnos y determinar la incidencia en el proceso enseñanza-aprendizaje.

En la asignatura “Programación de Sistemas” se estudia el desarrollo de software para la gestión de recursos de cómputo que incluyen sistemas operativos, compiladores, traductores y ensambladores, entre otros. La enseñanza de esta disciplina es tema de la presente investigación, en donde se han diseñado simuladores con algunas funcionalidades de los sistemas operativos [19][11] o para emular sistemas de cómputo con capacidades de ejecución de programas escritos en lenguajes de bajo nivel [16][14].

En la literatura existen pocos trabajos sobre el desarrollo de sistemas simuladores como apoyo en la enseñanza de asignaturas de programación de sistemas o similares, que abarquen tanto la arquitectura de computadoras, el diseño y uso de lenguajes de programación de bajo nivel y la simulación de las principales funciones de los sistemas operativos. Aunque es relativamente común el uso de simuladores y herramientas de software en educación superior y en la enseñanza de las ciencias computacionales, hay pocas investigaciones que muestren la aceptación de estos recursos por parte de los estudiantes, pues no siempre se realizan estudios que analicen la utilidad y efectividad de las herramientas para determinar su pertinencia en el proceso formativo.

En el trabajo de Ayala et al. [4], se presenta la descripción detallada de un software simulador de un sistema operativo y herramientas de programación de sistemas, específicamente para traducir código ensamblador a código máquina y ejecutar los programas generados, con opciones para analizar el estado de los procesos y aplicar varias técnicas de control de procesos y manejo de memoria. Consideramos que la propuesta resulta novedosa pues abarca varios aspectos del área de las ciencias computacionales y cubre necesidades concretas

detectadas en asignaturas relacionadas con la programación de sistemas; por ejemplo, el entendimiento del lenguaje máquina y la arquitectura de la computadora, el proceso de desarrollo de lenguajes de programación y el funcionamiento del sistema operativo. El sistema propuesto cuenta con herramientas que permiten observar y analizar estos procesos para una mejor comprensión. Adicionalmente, dado que tampoco se encontraron trabajos relacionados, el presente estudio contribuye con la descripción del análisis de usabilidad de un sistema simulador específico para la enseñanza de la programación de sistemas, con lo cual se espera contar con elementos para evaluar su incorporación como tecnología computacional en la enseñanza, y con el propósito de hacer que esta práctica se realice de manera más efectiva.

3 Descripción del simulador

3.1 Arquitectura

En la Figura 1, se muestran los principales elementos y módulos incorporados en el simulador. El editor y ensamblador facilitan la escritura de programas para generar código ejecutable para la máquina virtual. Se destaca que la computadora virtual originalmente fue desarrollada de manera independiente para ejecutar un solo programa en lenguaje máquina, almacenado en su memoria de forma estática, pero con la adición del módulo del sistema operativo, ahora es posible el multiprocesamiento y la selección de diferentes algoritmos de gestión de memoria y de planificación de procesos. Además, el simulador cuenta con módulos para el monitoreo y análisis de los procesos y recursos de la máquina virtual.

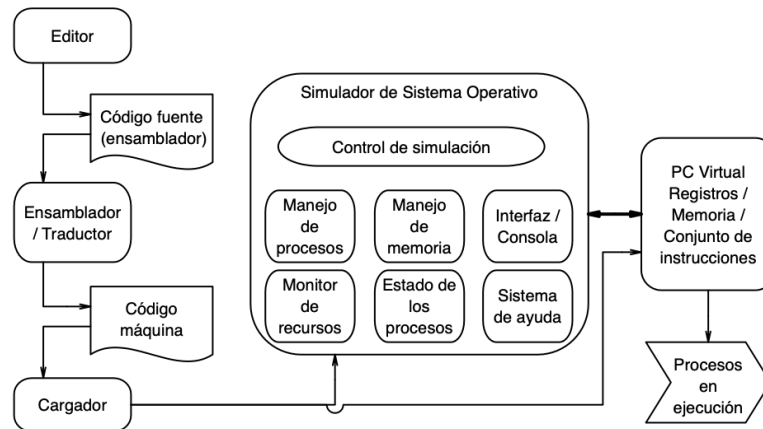


Figura 1 Arquitectura del sistema simulador.

3.2 Descripción de funcionalidades

A continuación, se presenta una descripción general de las principales funcionalidades del sistema empleado para contextualizar su aplicación. El diseño de la interfaz de usuario del simulador contiene dos opciones o áreas principales: la generación de programas y la simulación, en donde se controla la forma de ejecución de los programas. En la Figura 2 se observan las herramientas para la edición y manejo del código fuente ensamblador, la generación de ejecutables (traducción a código máquina), la gestión de los programas en lenguaje máquina y la carga de ejecutables para ser usados en el simulador del sistema

operativo. En la Figura 3, se muestra un ejemplo de un programa escrito en el lenguaje ensamblador del simulador siendo editado.

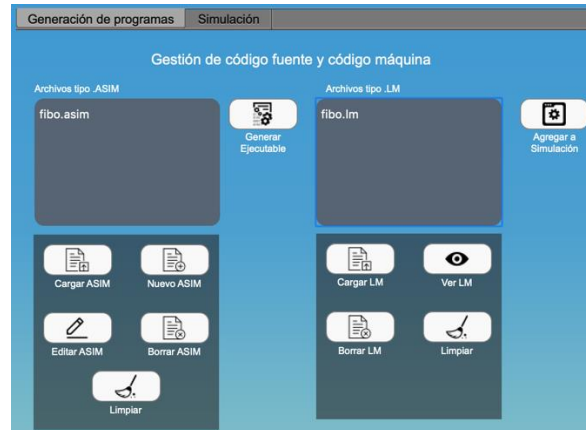


Figura 2 Módulo de edición y gestión de programas en ensamblador y código máquina.

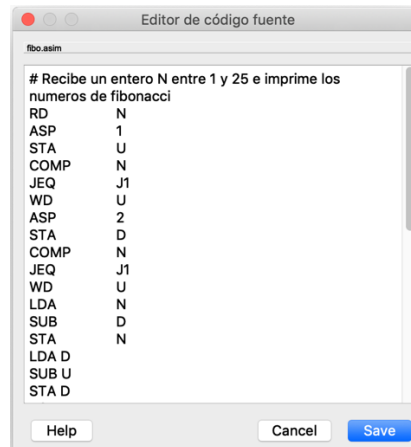


Figura 3 Edición de programa en lenguaje ensamblador.

La segunda área del sistema se muestra en la Figura 4. Cuenta con opciones para gestionar los programas a ejecutar en la simulación. Para ello, se deben crear los procesos (cargar a memoria y agregarlos a las estructuras internas del sistema operativo), ver los resultados en la consola del proceso y analizar la ejecución de los programas mediante la ventana de estado y la ventana de memoria. El módulo de control de simulación permite seleccionar algoritmos de calendarización de procesos y gestión de memoria, además de controlar la velocidad de ejecución, la cantidad de memoria y el modo de depuración (la salida en consola de cada instrucción ejecutada).

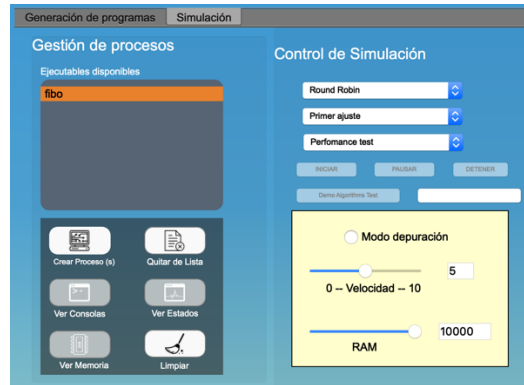


Figura 4 Módulo de simulación del sistema operativo.

Para iniciar la simulación y ejecución de programas, una vez elegidas las opciones disponibles, se presiona el botón Iniciar y se presentan los resultados en la ventana de la consola, además se pueden usar las herramientas para analizar detalladamente el estado de los procesos y la memoria, como se aprecia en el ejemplo mostrado en la Figura 5.

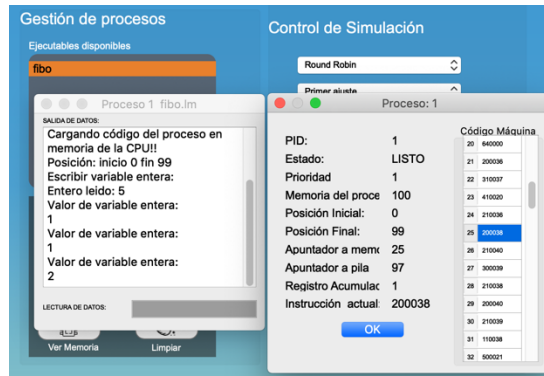


Figura 5 Simulación en curso, ventana de consola y estado de un proceso.

4 Metodología

Como se mencionó previamente, la usabilidad de un sistema considera aspectos cualitativos asociados a la percepción subjetiva de los usuarios potenciales de tal sistema, como la facilidad, utilidad. Estos aspectos o atributos no pueden ser medidos de manera directa, sino que se utilizan herramientas como encuestas o entrevistas. En este sentido, utilizamos dos pruebas ampliamente utilizadas en la literatura para valorar la usabilidad percibida de la propuesta mediante un experimento, y de esta manera analizar el grado de utilidad y usabilidad del simulador.

El experimento de usabilidad se desarrolló con alumnos de la asignatura “Programación de Sistemas” del séptimo semestre de la Licenciatura en Ciencias de la Computación de la Universidad Autónoma de Yucatán, en el semestre agosto-diciembre de 2018. Previamente, al inicio del curso, se revisaron los temas sobre conceptos de programación con lenguaje máquina y ensamblador; también se describieron las tareas de edición de programas en ensamblador, generación de código máquina, carga y ejecución de programas, con apoyo del simulador del sistema operativo. En una sesión de clases, se realizó una actividad en donde

se requería el uso del programa simulador. Después de completar la actividad, se solicitó a los participantes que contestaran cuestionarios estandarizados, para recoger sus opiniones respecto a la usabilidad y utilidad del software utilizado. En la Figura 6 se muestra el flujo general de las actividades realizadas durante el experimento.

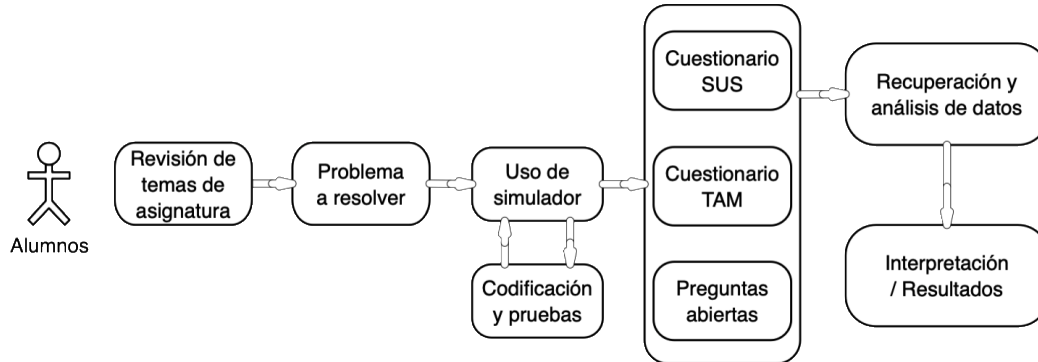


Figura 6 Flujo de actividades realizadas en el experimento.

4.1 Participantes

El grupo estuvo conformado por 8 alumnos (3 mujeres y 5 hombres) de entre 20 y 23 años, era la primera vez que cursaban la asignatura y utilizaban el software simulador. En cuanto a cursos previos, que contienen temas importantes relacionados a la asignatura “Programación de Sistemas”, solamente uno de los participantes no había cursado “Sistemas Operativos”. Seis de los participantes ya habían cursado la asignatura “Compiladores”, y dos la estaban cursando en ese mismo semestre. Todos los participantes tenían conocimientos a nivel intermedio de programación.

4.2 Actividad desarrollada

El diseño de la actividad consistió en la elaboración de un programa simple en lenguaje ensamblador (propio del simulador). La actividad fue: *elaborar un programa en lenguaje ensamblador que lea 3 números y determine el mayor de ellos, luego verificar su correcto funcionamiento y sus resultados.*

Primeramente, se les solicitó que intentaran hacer el programa sin apoyo del simulador, y posteriormente que lo utilizaran. Aunque el programa asignado es muy básico, sí requiere que los alumnos tengan conocimiento de la arquitectura de la máquina virtual implementada en el simulador y del conjunto de instrucciones del lenguaje ensamblador, dicha información se encuentra disponible en la misma aplicación. Además, para poder escribir, ejecutar y verificar el programa, se necesitan utilizar las funcionalidades disponibles en la herramienta, como son: edición, guardar archivo, traducir a lenguaje máquina, cargar al simulador el código máquina o ejecutable, preparar la simulación, ejecutar el programa, ver resultados en la consola del sistema, analizar el estado del proceso, ejecutar y revisar paso a paso las instrucciones, y consultar la ayuda del sistema para ver la arquitectura disponible y códigos de instrucciones a utilizar. Por lo tanto, la actividad diseñada es útil para probar gran parte del sistema e ilustra el uso de algunas de las herramientas básicas elaboradas en la programación de sistemas.

4.3 Instrumentos de recolección y análisis de datos

Los instrumentos para la recolección de datos y su análisis posterior han sido utilizados ampliamente para evaluar aspectos de usabilidad y utilidad de sistemas de software en general. Específicamente se utilizaron los instrumentos referidos como Escala de Usabilidad de Sistemas (SUS) y Modelo de Aceptación de Tecnología (TAM), los cuales se describen a continuación.

4.4 SUS (System Usability Scale)

El cuestionario SUS introducido por Brooke [8][9], ha mostrado ser efectivo para obtener una medida de usabilidad percibida y facilidad de aprendizaje de un sistema de software, es confiable a pesar de tamaños de muestra pequeños; por ejemplo, en [22] utilizaron muestras de 8 a 12 usuarios con buenos resultados, Sauro & Lewis [23] discuten ampliamente la validez de resultados estadísticos con muestras menores a 10 participantes, incluso entre 2 y 5 usuarios. En [24] se proporcionan reglas para determinar la probabilidad de descubrimiento de problemas en el software, para varios tamaños de muestra y probabilidades de ocurrencia; se explica que con tamaños de muestra entre 5 y 10, los estudios probablemente descubrirán suficientes problemas para mejorar los productos de software, por lo que el uso de la prueba de usabilidad en estos casos sigue siendo válida. En [5] y [20] se ha mostrado que SUS puede ser utilizado con una amplia gama de sistemas y tipos de tecnologías, permitiendo discriminar e identificar sistemas con bajo o alto nivel de usabilidad.

El cuestionario contiene 10 preguntas (5 positivas y 5 negativas), a contestar mediante escala Likert, de cinco opciones de elección, en el rango de “totalmente en desacuerdo” a “totalmente de acuerdo”. El resultado total varía de 0 a 100, esta puntuación total es la única que debe ser tomada como referencia, es decir, las puntuaciones de elementos individuales no son significativas por sí solas [8]. El procedimiento para normalizar el resultado y adjetivarlo mediante un rango de categorías se explica en [5][6]. Los posibles valores de las preguntas van de 0 a 4. A las puntuaciones de las preguntas impares se les resta 1, mientras que a los valores de las preguntas pares se resta de 5 la puntuación aportada. Se suman todas las puntuaciones obtenidas y se multiplica por un factor de 2.5. Este valor total se ubica en una escala de clasificación para su interpretación, como se observa en la Figura 7, en general se considera aceptable un valor mayor de 70.

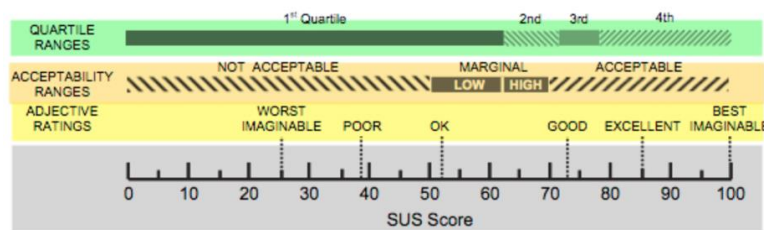


Figura 7 Escala SUS. Fuente: Bangor et al. [6].

Tópicos del cuestionario SUS

1. Creo que me gustará usar con frecuencia este sistema.
2. Encontré el sistema innecesariamente complejo.
3. Pensé que era fácil utilizar el sistema.
4. Creo que necesitaría del apoyo de un experto para usar el sistema.
5. Encontré las diversas posibilidades del sistema bastante bien integradas.

6. Pensé que había demasiada inconsistencia en el sistema.
7. Imagino que la mayoría de las personas aprenderían muy rápidamente a utilizar el sistema.
8. Encontré el sistema muy incómodo para usarlo.
9. Me sentí muy confiado en el manejo del sistema.
10. Necesito aprender muchas cosas antes de manejar el sistema.

4.5 TAM (Technology Acceptance Model)

El cuestionario basado en el modelo de aceptación de tecnología, TAM, mide la percepción de los usuarios respecto a la facilidad de uso y la utilidad de un sistema [13]. La aceptación y uso de una aplicación esta fuertemente determinada por estos factores, por lo que la adopción de un sistema o tecnología en la educación se ve influenciada por las creencias de los usuarios sobre el beneficio que le aportará a la realización de sus tareas y al mismo tiempo, qué tan sencillo resulta realizar su trabajo con la nueva herramienta. Diversas investigaciones indican que las actitudes, creencias e intenciones de los estudiantes hacia la tecnología, así como sus interrelaciones estas condicionadas por la percepción de facilidad de uso de esta [3]. La confiabilidad y validez de TAM han sido estudiada ampliamente y es soportada por varios trabajos como se cita en [1][25], así como su posible uso con diferentes poblaciones de usuarios y con distintos tipos de sistemas. TAM es probablemente uno de los modelos más ampliamente citados en el campo de la aceptación de tecnología y durante las últimas décadas ha consolidado su soporte empírico [26].

El instrumento contiene doce preguntas, las respuestas se presentan mediante una escala Likert de siete opciones que van de “Extremadamente improbable” a “Extremadamente probable”. En general, las primeras seis preguntas dan cuenta de la utilidad percibida, y las seis restantes, de la facilidad de uso percibida. Se calcula la media del grupo, para cada una de las preguntas, y se analizan los resultados.

Tópicos del cuestionario TAM

1. Usar este sistema me ayudaría a hacer mis tareas más rápido.
2. Usar este sistema mejoraría el desempeño de mi trabajo.
3. Usar este sistema incrementaría mi productividad.
4. Usar este sistema aumentaría la efectividad de mi trabajo.
5. Usar este sistema me facilitaría la realización de mi trabajo.
6. Encontraría este sistema útil en mi trabajo.
7. Aprender a utilizar este sistema sería fácil para mi.
8. Encontraría este sistema fácil para hacer lo que tengo que hacer.
9. Mi interacción con este sistema sería clara y entendible.
10. Encuentro este sistema flexible para interactuar con él.
11. Sería fácil para mi llegar a ser un experto en el uso de este sistema.
12. Encuentro a este sistema fácil de utilizar.

Además de los cuestionarios TAM y SUS, se pidió a los estudiantes que contestaran otras tres preguntas en formato libre para conocer directamente la opinión respecto a la utilidad y usabilidad del sistema y obtener recomendaciones o sugerencias para su mejora.

4.6 Procedimiento

Los cuestionarios se prepararon y quedaron disponibles para contestar en una plataforma en línea. La sesión de trabajo se llevó a cabo en el centro de cómputo, con disponibilidad de computadoras para todos los alumnos, con la aplicación previamente instalada. Después de completar la actividad, en aproximadamente 40 minutos, se solicitó a los estudiantes que respondieran los cuestionarios. Los resultados se almacenaron en una hoja de cálculo y se realizaron los cálculos para obtener las puntuaciones y su interpretación.

5 Resultados

5.1 Resultados del cuestionario SUS

Los resultados de los participantes se observan en la Figura 8. El promedio del grupo fue de 74.4 con una desviación estándar de 15.16. Este resultado indica que la aplicación es Aceptable, y la ubica en una categoría de Buena, sin llegar a ser Excelente.

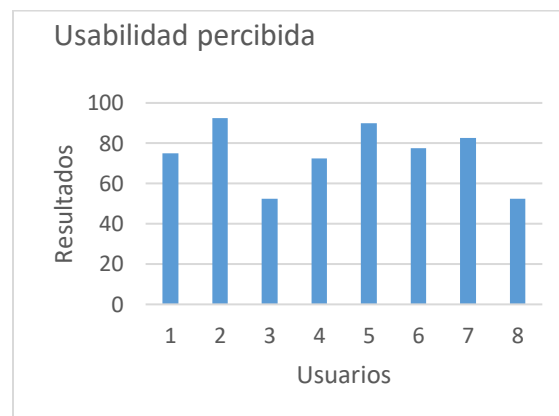


Figura 8 Resultados individuales del cuestionario SUS.

La prueba de normalidad Kolmogorov-Smirnov ($D=0.19741$, $p\text{-value}=0.85874$) establece que los datos no difieren de forma significativa con respecto a la media y se distribuyen normalmente. Los valores de la asimetría y la curtosis confirman lo anterior al estar cercanos a cero (Tabla 1). La variabilidad de los resultados de utilidad de SUS se puede apreciar gráficamente en una gráfica de cajas y bigotes (Figura 9).

Un análisis a detalle de cada reactivo (Tabla 2) permite identificar claramente que el 76% de los sujetos está de acuerdo en utilizar frecuentemente la propuesta y el 51% consideran que el sistema no es muy complejo y es fácil de usar (50%). El 50% de los participantes cree que no necesitaría soporte técnico, además de que tienen la percepción de que las funciones del sistema están bien integradas (100%) lo que hace que no haya inconsistencias en su utilización (76%). El 100% de los sujetos está de acuerdo en que aprender a utilizar el simulador fue rápido y están completamente en desacuerdo que resulte incómodo en utilizar (88%). El 100% de los usuarios se ha sentido muy seguro utilizando la aplicación y no considera que necesite aprender muchas cosas antes de usarla (26%).

Tabla 1. Estadísticas básicas de SUS.

Medida	Valor
Muestra	8
Media	74.38
Mediana	76.25
Desviación estándar	15.16
Mínimo	52.5
Máximo	92.5
Primer cuartil	57.5
Tercer cuartil	88.13
Rango intercuartílico	30.63
Outlier	Ninguno
Asimetría	-0.58
Curtosis	-0.81

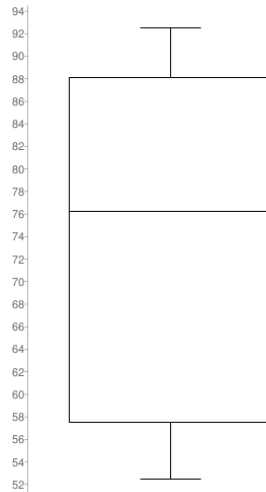


Figura 9 Gráfico de cajas y bigotes del cuestionario SUS.

En cuanto al reactivo positivo con mayor calificación (el 5, con 75%) fue “Me sentí muy confiado en el manejo del sistema.”. En contraste, el reactivo negativo con la mayor puntuación corresponde a “Encontré el sistema muy incómodo para usarlo” (el 1, con 88%). De lo anterior podemos concluir que el sistema presenta un alto grado de involucramiento con el usuario al fomentar la confianza y la comodidad del usuario al utilizar el simulador.

Tabla 2. Distribución de las respuestas de los sujetos en la encuesta SUS.

Reactivos SUS	Valores				
	1	2	3	4	5
1. Creo que me gustará usar con frecuencia este sistema.	0%	0%	25%	38%	38%
2. Encontré el sistema innecesariamente complejo.	38%	13%	25%	13%	13%
3. Pensé que era fácil utilizar el sistema.	13%	13%	25%	25%	25%
4. Creo que necesitaría del apoyo de un experto para usar el sistema.	25%	25%	25%	25%	0%
5. Encontré las diversas posibilidades del sistema bastante bien integradas.	0%	0%	0%	38%	63%
6. Pensé que había demasiada inconsistencia en el sistema.	38%	38%	0%	25%	0
7. Imagino que la mayoría de las personas aprenderían muy rápidamente a utilizar el sistema.	0%	0%	0%	75%	25%
8. Encontré el sistema muy incómodo para usarlo.	88%	0%	13%	0	0
9. Me sentí muy confiado en el manejo del sistema.	0%	0%	0%	25%	75%
10. Necesito aprender muchas cosas antes de manejar el sistema.	13%	13%	50%	13%	13%

5.2 Resultados del cuestionario TAM

Los resultados derivados del cuestionario TAM se observan en la Tabla 3. La media para la utilidad percibida fue de 5.94, y para la facilidad de uso de 5.85. Esto valores reflejan una buena aceptación del software evaluado.

Tabla 3. Resultados generales TAM

UTILIDAD	MEDIA	DESVEST
1 Trabajar más rápido	6.38	0.74
2 Desempeño del trabajo	6.00	0.93
3 Incremento de productividad	5.63	1.06
4 Efectividad	5.75	0.89
5 Hacer más fácil el trabajo	5.88	0.83
6 Utilidad	6.00	1.20
MEDIA DE UTILIDAD	5.94	
FACILIDAD DE USO	MEDIA	DESVEST
7 Fácil de aprender	5.75	0.71
8 Controlable	5.75	1.04
9 Claro y entendible	5.88	0.64
10 Flexible	5.75	1.16
11 Fácil para ser experto	5.75	1.04
12 Fácil de utilizar	6.25	0.71
MEDIA DE FACILIDAD DE USO	5.85	

La prueba de normalidad Kolmogorov-Smirnov ($D=0.17542$, $p\text{-value}=0.93251$) establece que los datos no difieren de forma significativa con respecto a la media y se distribuyen normalmente. Los valores de la asimetría y la curtosis (Tabla 4) confirman lo anterior al estar cercanos a cero. La variabilidad de los resultados de TAM se puede apreciar gráficamente en una gráfica de cajas y bigotes (Figura 10).

Tabla 4. Estadísticas básicas de TAM.

Medida	Valor
Muestra	8
Media	70.75
Mediana	71
Desviación estándar	7.11
Mínimo	60
Máximo	83
Primer cuartil	66
Tercer cuartil	75.75
Rango intercuartílico	9.75
Outlier	Ninguno
Asimetría	0.34
Curtosis	0.27

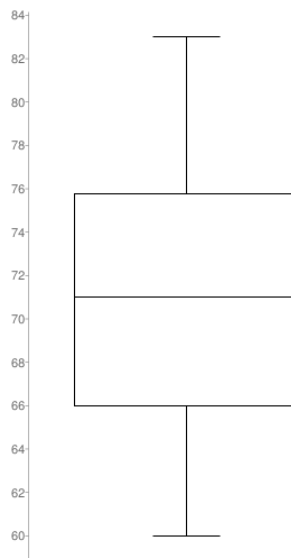


Figura 10 Gráfico de cajas y bigotes del cuestionario TAM.

Un análisis a detalle de cada reactivo (Tabla 5) permite identificar claramente que el 88% de los sujetos está de acuerdo en que la propuesta le permite realizar las tareas más rápido y el 63% consideran que el sistema mejora su desempeño e incrementa su productividad (50%). El 51% de los participantes cree que aumentaría su efectividad y facilita la realización de su trabajo (63), además que lo consideran útil (63%). El 63% de los sujetos está de acuerdo en que aprender a utilizar el sistema fue fácil y es sencillo de controlar. El 76% consideran que la interacción con el sistema es clara, entendible y es flexible (51%), además que es fácil de

utilizar (88%), logrando que puedan convertirse en expertos de manera pronta (63%). En cuanto al reactivo positivo con mayor calificación (el 6, con 63%) fue “Mi interacción con este sistema sería clara y entendible”. De lo anterior podemos concluir que el sistema presenta un alto grado de involucramiento con el usuario al fomentar la confianza y la comodidad del usuario al utilizar el simulador.

Tabla 5. Distribución de las respuestas de los sujetos en la encuesta TAM.

Reactivos TAM	Valores						
	1	2	3	4	5	6	7
1. Usar este sistema me ayudaría a hacer mis tareas más rápido.	0%	0%	0%	0%	13%	38%	50%
2. Usar este sistema mejoraría el desempeño de mi trabajo.	0%	0%	0%	0%	38%	25%	38%
3. Usar este sistema incrementaría mi productividad.	0%	0%	0%	13%	38%	25%	25%
4. Usar este sistema aumentaría la efectividad de mi trabajo.	0%	0%	0%	0%	50%	25%	26%
5. Usar este sistema me facilitaría la realización de mi trabajo.	0%	0%	0%	0%	38%	38%	25%
6. Encontraría este sistema útil en mi trabajo.	0%	0%	0%	13%	25%	13%	50%
7. Aprender a utilizar este sistema sería fácil para mi.	0%	0%	0%	0%	38%	50%	13%
8. Encontraría este sistema fácil para hacer lo que tengo que hacer.	0%	0%	0%	13%	25%	38%	25%
9. Mi interacción con este sistema sería clara y entendible.	0%	0%	0%	0%	25%	63%	13%
10. Encuentro este sistema flexible para interactuar con él.	0%	0%	0%	13%	38%	13%	38%
11. Sería fácil para mi llegar a ser un experto en el uso de este sistema.	0%	0%	0%	13%	25%	38%	25%
12. Encuentro a este sistema fácil de utilizar.	0%	0%	0%	0%	13%	50%	38%

5.3 Preguntas abiertas

En las preguntas abiertas se resumen los comentarios vertidos por los alumnos:

Utilidad: Muy útil para mostrar el proceso de traducción de ensamblador a lenguaje máquina y para mostrar cómo un sistema operativo administra los procesos; muy útil para entender mejor los conceptos vistos en clase al ver el seguimiento del proceso mediante la ventana de estado; sirve para entender la lógica de la programación en lenguaje ensamblador y los conceptos del manejo de instrucciones en lenguaje máquina; es eficiente y bueno como herramienta para reforzar el conocimiento y para madurar el manejo de comandos e instrucciones. Es una herramienta bastante sencilla que nos ofrece trabajar con un lenguaje de ensamblador bastante corto pero detallado, muy bueno para comenzar a trabajar y comprender esos campos, permitiendo ver detalle a detalle el proceso, lo que nos permite encontrar errores en nuestros ejercicios.

Facilidad de uso: Fácil de usar, pero requiere una pequeña tutoría, no se requiere mucho conocimiento técnico para entender el funcionamiento del sistema y tiene una pequeña guía,

es fácil de aprender y se puede usar para ejercicios básicos, es entendible. Es un poco confuso de usar en un principio, hasta que se ubican las herramientas a usar y se vuelve mucho más fácil.

Mejoras: Se detectó un problema en la visualización del tiempo de animación que siempre iniciaba en 5, en un caso no se iniciaban los cálculos (posiblemente debido a errores en la codificación del programa), simplificar las instrucciones pues para programas simples se requieren muchas líneas de código, incluir cálculo de raíz cuadrada en las instrucciones del lenguaje. Mejorar la sección de ayuda para se acceda más fácilmente la descripción del lenguaje ensamblador para no recurrir a documentos externos al sistema, y además ampliar la ayuda general para saber cómo funcionan los elementos de las pestañas con mayor detalle.

6 Conclusiones

Si bien el estudio tiene limitantes, dadas las características de la tarea asignada, el uso parcial del sistema simulador y lo reducido del grupo, fue posible obtener información relevante para conocer la percepción de los alumnos respecto a la integración de esta herramienta tecnológica en el aula y para su posible mejora.

La utilidad percibida indica el grado en que el alumno cree que la herramienta puede ayudarle a realizar sus tareas y mejorar su rendimiento académico. De forma general, con los resultados obtenidos se puede establecer que los alumnos aceptan la herramienta, la consideran útil y de fácil manejo. En la mayoría de los participantes la tendencia fue calificar alto los elementos presentados y externar comentarios positivos de la herramienta, aunque sí se detectaron algunos errores de funcionamiento y se hacen sugerencias para mejorar el sistema.

Los comentarios de los alumnos coinciden con los resultados de los cuestionarios TAM y SUS, al considerar el sistema simulador aceptable en cuanto a utilidad y facilidad de uso para ser empleado como instrumento de enseñanza-aprendizaje en el curso de “Programación de Sistemas”, y posiblemente, en otras asignaturas relacionadas, como “Compiladores”, “Sistemas Operativos”, “Arquitectura de Computadoras” o “Fundamentos de Programación”, pues facilita la comprensión de sus conceptos básicos y refuerza el desarrollo de habilidades requeridas en la programación en general.

7 Trabajo futuro

Como trabajo futuro, derivado de los resultados y comentarios obtenidos, se considera una revisión y mejora de la herramienta y su posterior evaluación, tomando en cuenta un mayor número de usuarios y rediseñando las actividades de aprendizaje, así como comparándolo con otros sistemas similares. En este sentido, el sistema simulador se sigue actualizando con la ayuda de alumnos interesados en el tema y bajo la supervisión de los autores de este trabajo. Así mismo, se están tomando en cuenta los comentarios vertidos en las encuestas corrigiendo aquellos aspectos que lo requieran. El propósito es utilizar el simulador de manera regular como herramienta de apoyo en la enseñanza de asignaturas con temas afines a los tratados en la programación de sistemas. Por tal motivo, se contará con más grupos de alumnos en donde se podrán realizar nuevas pruebas para confirmar la tendencia en los resultados reportados en este momento. Dichas pruebas serán rediseñadas para elaborar programas en ensamblador más complejos y ejecutarlos de manera que se pueda evaluar la totalidad de las funcionalidades de las herramientas implementadas en el sistema; con ello, se tendrán mejores bases para determinar su validez y transferibilidad en otros contextos.

Referencias

- [1] Adams, D. A., Nelson, R. R., & Todd, P. A. (1992). Perceived usefulness, ease of use, and usage of information technology: A replication. *MIS Quarterly*, 16(2), 227–247.
- [2] Alnoukari, M., Shafaamry, M. & Aytouni, K. (2013). Simulation for Computer Sciences Education. *Communications of the ACS*, 6(1).
- [3] Arteaga-Sánchez, R., Duarte-Hueros, A. & García-Ordaz, M. (2013). E-learning and University of Huelva: a study of WebCT and the technological acceptance model. *Campus-Wide Information Systems*, 30(2), 135-160. doi: 10.1108/10650741311306318
- [4] Ayala, E., Madera, F. & Basto, L. (2017). Operating System Simulator to Translate Assembler Code to Machine Code. *International Journal of Computer Trends and Technology (IJCTT)*, 51(1):58-63.
- [5] Bangor, A., Kortum, P. T., & Miller, J. T. (2008). An empirical evaluation of the System Usability Scale. *International Journal of Human-Computer Interaction*, 24(6), 574-594.
- [6] Bangor, A., Kortum, P. T., & Miller, J. T. (2009). Determining what individual SUS scores mean: Adding an adjective rating scale. *Journal of Usability Studies*. 4(3), 114–123.
- [7] Bates, A. W. (2015). *Teaching in a digital age: Guidelines for designing teaching and learning*. Vancouver BC: Tony Bates Associates Ltd.
- [8] Brooke, J. (1996). SUS: A ‘quick and dirty’ usability scale. In P. Jordan, B. Thomas, & B. Weerdmeester (Eds.), *Usability Evaluation in Industry* (pp. 189–194). London, UK: Taylor & Francis.
- [9] Brooke, J. (2013). SUS: A retrospective. *Journal of Usability Studies*, 8(2), 29–40.
- [10] Cabero-Almenara, J., & Costas, J. (2016). La utilización de simuladores para la formación de los alumnos. *Prisma Social*, (17), 343–372.
- [11] Cahya, S. (2009). Designing Operating System Simulator: A Learning Tool. 2009 11th International Conference on Computer Modelling and Simulation, Cambridge, pp. 156- 160, 2009. doi: 10.1109/UKSIM.2009.92
- [12] Cuesta, I.I., Abella, V. & Alegre, J.M (2014). Evaluación del módulo de cuestionarios del entorno de trabajo UBUVirtual mediante el modelo de aceptación tecnológica. *Profesorado: Revista de currículum y formación del profesorado*. 18(1).
- [13] Davis, F. D. (1989). Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology. *MIS Quarterly*, 13(3), 319–340. <https://doi.org/10.2307/249008>
- [14] Deitel, F. & Deitel, O. (2004). *Como Programar en C/C++ y Java*. (4ª Ed.), México: Pearson.
- [15] Janitor, J., Jakab, F., & Kniewald, K. (2010). Visual Learning Tools for Teaching/Learning Computer Networks: Cisco Networking Academy and Packet Tracer. 2010 Sixth International Conference on Networking and Services.
- [16] Leland, B. (1988). *Software de Sistemas. Introducción a la Programación de Sistemas*. Addison Wesley.

- [17] Navarro, R. E., & Santillán, A. G (2009). Un modelo didáctico basado en el diseño de simuladores: el caso de la matemática financiera. *Ide@s CONCYTEG*. 4(46), 14.
- [18] Ruiz, J.M. (2008). La Simulación como Instrumento de Aprendizaje. Recuperado de: http://fp.atxuri.net/escenarios/Simulacion_como_Instrumento_de_Aprendizaje.pdf.
- [19] Saraswat P.K. & Gupta, P. (2006). Design and Implementation of a Process Scheduler Simulator and an Improved Process Scheduling Algorithm for Multimedia Operating Systems. *International Conference on Advanced Computing and Communications*, Surathkal, pp. 513-517.
- [20] Sauro, J. (2011). *A practical guide to the System Usability Scale*. Denver, CO: Measuring Usability LLC.
- [21] Torres, P. C., & Cobo, J. K. (2017). Tecnología educativa y su papel en el logro de los fines de la educación. *Educere*, 21(68), 31–40.
- [22] Tullis, T. S., & Stetson, J. N. (2004). A comparison of questionnaires for assessing website usability. Paper presented at the Usability Professionals Association Annual Conference, June. UPA, Minneapolis, MN.
- [23] Sauro, J., & Lewis, J. R. (2016). *Quantifying the user experience: Practical statistics for user research*, 2nd. Ed. Elsevier, USA.
- [24] Lewis, J. R. (2014). Usability: Lessons Learned ... and Yet to Be Learned. *International Journal of Human-Computer Interaction*, 30(9), 663–684.
- [25] Wallace, L. & Sheetz, S. (2014). The adoption of software measures: A technology acceptance model (TAM) perspective. *Information & Management*, 51 (2), 249-259
- [26] Hamed, T. (2018) A review of technology acceptance and adoption models and theories. *Procedia Manufacturing*, 22, pp. 960-967