



Agentes tutores inteligentes: Apoyo educativo dentro de plataformas virtuales.

Claudia Rosado Guzmán
Daniel Sánchez Guzmán

Universidad del SABES
Instituto Politécnico Nacional

En la actualidad se ha visto un crecimiento considerable del uso de sistemas virtuales de educación, de manera genérica lo que se conoce como: Sistema de Administración de Aprendizaje ó Learning Management System (LMS) en inglés. Estos sistemas han demostrado sus beneficios dentro de las instituciones educativas como son flexibilidad, instrucción asíncrona y a distancia. De igual manera estos sistemas han sido implementados en diversos niveles de educación, principalmente dentro de los niveles de bachillerato hasta nivel de posgrado. Una situación característica de estos sistemas es el relacionado con la cobertura educativa, sí bien es conocido que estas herramientas permiten atender a una cantidad de usuarios alta, entre cientos y miles; también es importante mencionar que la tasa de crecimiento de personal docente que atiende a esa cantidad de usuarios se ve rebasada por el número de estudiantes matriculados, lo que acarrea con ello un problema



en la relación estudiante/docente. Para reducir este déficit de atención se presenta la propuesta del uso de Agentes Tutores Inteligentes, esta herramienta actúa como una capa intermedia entre el docente y el estudiante, apoyando en actividades de instrucción y retroalimentación al estudiante; en otras investigaciones se ha demostrado la eficiencia del empleo de este tipo de herramientas, pero se continúa con el desarrollo y aplicación dentro de diferentes unidades de aprendizaje y dentro de diferentes niveles educativos.

INTRODUCCIÓN

En la actualidad la educación en línea es un paradigma que está adquiriendo un papel muy importante, muchas instituciones y centros educativos tienden a adoptar estrategias que permitan ofrecer una mayor y mejor cobertura de sus planes y programas de estudio a un número mayor de población (Sánchez, 2009), esto con el objetivo de acelerar el Proceso Enseñanza-Aprendizaje (PEA). Aunado a la lenta implementación de este paradigma, por cuestiones de infraestructura en los países en vías de desarrollo, existe también la barrera sobre el concepto clásico de la enseñanza presencial en donde un grupo de alumnos se presenta al aula a recibir clases, el maestro expone su tema y se discuten problemas relacionados a este, de los cuales la mayoría de las veces no se contrasta el concepto adquirido contra otras fuentes de información o no se realiza la experimentación de ciertos fenómenos físicos; llegando a limitar y acarrear consigo el problema de no tener una retroalimentación más enriquecedora que permita al estudiante validar los conceptos adquiridos y a la vez mejore su formación académica.

El cambio de paradigma no tiene gran aceptación en las personas y organizaciones aún y a pesar de los logros que las aplicaciones de la tecnología ha demostrado. Entre las fuentes individuales de resistencia al cambio se hallan los hábitos adquiridos y convertidos en rutinas que no facilitan sino anquilosan, la realización reflexionada de actividades con significado. La resistencia organizacional apunta a diversas inercias estructurales de mecanismos contruidos ancestralmente que si bien producen estabilidad, son anacrónicos a las demandas de tiempos culturales diferentes. La inercia de una institución o grupo se da porque sus normas actúan restringiendo el cambio, sin enfrentarlo y repensarlo por lo que también, contribuyen a la resistencia nombrada (Fainholc, 2008). Desde la enseñanza presencial convencional también se puede describir un proceso de evolución convergente, desde que las TIC's y sobre todo las redes se utilizan como un medio de distribución de la enseñanza. Por ello, parece necesario reflexionar sobre los elementos y las relaciones que se establecen y que entran en juego en éstas nuevas modalidades de enseñanza-aprendizaje (Sánchez, 2009). Esto supone nuevos entornos, y requiere nuevos enfoques para entenderlos, diseñarlos e implementarlos (Salinas, 2002).

Uno de los mayores problemas con los que se enfrentan los docentes en las aulas es en el estudio de las ciencias exactas, como lo son Matemáticas, Física y Química. Estos estudios son más tediosos para los alumnos visto desde su punto de vista, y parte fundamental son las



bases que los alumnos han adquirido en la formación básica y media superior, esto ya que los alumnos consideran, o se tiene la creencia que las disciplinas antes citadas no tienen ya aplicación práctica (Sánchez, 2009). Puesto que todos los estudiantes no son iguales, ellos aprenden de maneras diferentes. Si los profesores asociaran el éxito con lo que aprenden los estudiantes, deben de prepararse para adoptar estilos de instrucción que coincidan con la manera en la que los alumnos aprenden. Revisando los fundamentos para la práctica educativa – las teorías de aprendizaje – puede derivarse una amplia gama de potenciales estructuras interactivas que deben reforzar el proceso de aprendizaje, ya sea por interacción física simple o a través de un compromiso cognoscitivo más complejo e implícito (Orellana, 2002).

Todo lo anterior nos da un marco general básico y teórico sobre el cual se puede desarrollar un conjunto de herramientas de aplicación de nueva generación, que permitirá mostrar en el presente trabajo como es que la aplicación de la Inteligencia Artificial en el campo educativo puede mostrarnos avances significativos en el proceso de aprendizaje de los estudiantes. El presente trabajo se compone de los siguientes apartados: Introducción, se presenta un marco básico de trabajo y de descripción del entorno común de la educación en virtual y a distancia, presentando los actores y factores generales actuales en esta modalidad educativa. Agentes Tutores Inteligentes, se describe lo que es un agente inteligente, como la inteligencia artificial ha contribuido en el desarrollo de sistemas educativos que apoyen de manera positiva el aprendizaje y al final se presentan diferentes implementaciones actuales de estos sistemas. Caso Práctico, en este apartado se presenta una aplicación práctica utilizando agentes tutores inteligentes para la enseñanza de la unidad de aprendizaje correspondiente al Sistema Internacional de Unidades, describiendo de manera general su creación y puesta en marcha. Conclusiones, es importante mencionar cual es el estado actual que tiene este tipo de aplicaciones en el ámbito educativo dentro de México y que se está desarrollando de manera paralela dentro de diferentes grupos de investigación e instituciones.

Palabras clave

Niveles Educativos . **Software Educativo**
Educación Virtual Educación Virtual y a Distancia
Agentes tutores inteligentes

1. Agentes tutores inteligentes

El término agente tiene diferentes significados y definiciones en distintos ámbitos y comunidades, en este trabajo se hará referencia como agente a un sistema informático, situado en un entorno determinado dentro del cual actúa de forma autónoma y flexible para cumplir sus objetivos (Les; Cumming y Finch, 1999). Específicamente, un agente es un personaje simulado por computadora y que tiene características humanas como la capacidad de aprender de la experiencia y del conocimiento previo, emociones (considerando que se pueden identificar pero no mostrar), autonomía y otras características personales que facilitan el aprendizaje. Estas características pueden ser expresadas mediante texto, gráficos, íconos, voz, animación ó multimedia (Chou; Chan y Lin, 2003). La forma de implementarlos y mantenerlos no incluye mucho coste y son de fácil adaptación. Los agentes inteligentes surgen como una opción más para poder ser aplicados dentro del ámbito educativo, es de fácil implementación y los docentes no deben preocuparse tanto por cuestiones técnicas sino de contenido de los programas académicos, el desarrollo de estos tutores no es reciente pero en la actualidad se cuenta con la capacidad de procesamiento suficiente para poderlos operar.

Los agentes inteligentes de software no tienen un desarrollo reciente, se está hablando de los años cincuenta cuando varios investigadores entre los que se encuentran Rosenblatt y Samuel (Kenneth, 2008) comenzaron a explorar el desarrollo de software inteligente. Fogel y otros colaboradores incursionaron un trabajo pionero para ese entonces, basado en el uso de evolución simulada como un mecanismo más flexible y abierto, logrando crear agentes que exhibían un comportamiento inteligente y complejo; modificando los ambientes de manera dinámica (Fogel, Owens, Walsh, 1966). Un factor importante que impacto en el desarrollo de estas herramientas fue la capacidad de procesamiento de los ordenadores de aquella época, en años recientes se ha tenido un crecimiento exponencial en la capacidad de cálculo y procesamiento lo que ha permitido que el estudio y aplicación de los agentes inteligentes tenga un crecimiento mayor (Sánchez, 2009).

2.1 Implementaciones de agentes tutores inteligentes

Se expondrán diferentes implementaciones de agentes. Apoyándose en las mismas se tendrá un panorama general de lo que es un agente inteligente para concluir con la implementación que enmarcará el agente a aplicar dentro del presente trabajo.

- Agente MuBot. “El término agente es usado para representar dos conceptos ortogonales. El primero es que el agente debe de tener la habilidad de ejecutarse de manera autónoma. La segunda tiene que ver con la habilidad del agente para tener un razonamiento orientado al dominio al que pertenece”. Este punto de vista lo propone Sankar Virdhagriswaran (Virdhagriswaran, 1999).

- Agente AIMA. “Un agente es cualquier cosa que puede ser visto como un perceptor de su ambiente a través de sensores y actuar dentro de su ambiente utilizando actuadores.” (Russell, Norving, 1995). AIMA es el acrónimo de: “Artificial Intelligence: a Modern Approach”, libro de Inteligencia Artificial (IA), bastante utilizado en más de doscientas universidades en 1995. Los autores estaban interesados en las técnicas de IA para incorporar agentes. Claramente la definición de AIMA dependía en extremo sobre el medio ambiente y sobre lo que sensar y actuar significaba. Si se definiera el ambiente como cualquier cosa que proveyera entradas y aceptara respuestas, entonces cualquier programa que recibiera entradas y entregara resultados a través de los periféricos de salida, se puede considerar un agente. Por lo tanto si se quiere tener una diferencia marcada entre lo que es un agente y lo que es un programa, se debe de restringir y delimitar muy bien lo que es el ambiente, sensar y actuar.

- Agente Maes. “Los agentes autónomos son sistemas computacionales que habitan en un medio ambiente complejo y dinámico, perciben y actúan de manera autónoma dentro de este medio ambiente, y como consecuencia pueden realizar un conjunto de metas o tareas para los cuales han sido creados” (Maes, 1995). Pattie Maes del laboratorio de medios del MIT es una de las pioneras en la investigación de agentes. Ella agrega un elemento crucial en la definición de un agente: “los agentes deben de actuar de manera autónoma así como realizar ciertas metas.” También los ambientes son restringidos a ser complejos y dinámicos, lo que no está claro es el

papel que estos elementos juegan dentro de los programas y sus respectivas restricciones.

- Agente KidSim. “Se define agente como una entidad de software persistente creada para un propósito en específico. ‘Persistente’ es el contexto que distingue al agente de subrutinas; los agentes tienen sus ideas para poder realizar diferentes tareas, así como su propia planificación. El ‘Propósito específico’ distingue al agente de las aplicaciones multifunción esto ya que los agentes son más pequeños.” (Smith, Cypher, Spohrer, 1994).

Existen múltiples ejemplos de sistemas tutores inteligentes, algunos desarrollados en el mundo universitario y otros creados con fines comerciales. Entre los primeros, uno de los más conocidos es el sistema Andes, desarrollado por el equipo de Kart VanLehn de la Universidad de Pittsburg (VanLehn, K., Lynch, C., Schulze, K., Shapiro, J. A., Shelby, R., Taylor, L., Treacy, D., Weinstein, A., Wintersgill, M., 2005). El sistema se encarga de guiar a los estudiantes mientras estos resuelven problemas y ejercicios. Cuando el estudiante pide ayuda en medio de un ejercicio, el sistema aporta pistas para avanzar en la solución o indica qué ha fallado en algún paso anterior.

- Agente CTAT. “Cognitive Tutor Authoring Tools (CTAT), es un conjunto de herramientas que permiten desarrollar agentes tutores inteligentes enfocados a la educación; este tipo de tutores están siendo creados por un grupo de investigadores en Carnegie Mellon University, se trabaja el desarrollo de dos tipos de tutores principales: Tutores de Ejemplo de Traza (Example-Tracing Tutors) y Tutores Cognitivos (Cognitive Tutors)” (PSLC, 2009).

CTAT es el conjunto de herramientas que ha aportado una flexibilidad de adecuación dentro de diferentes experimentos educativos, esto debido a que el grupo de investigación que trabaja en este proyecto da una flexibilidad a CTAT, enmarcándolo como un software de soporte para diferentes tipos de investigaciones, de igual manera CTAT tiene fundamentos similares a Andes ya que han sido desarrollados dentro de la misma universidad.

A continuación se describirá el caso práctico donde el software CTAT en conjunto con la herramienta Adobe Flash Professional CS5® permiten aplicar este tipo de agentes tutores inteligentes aplicados a la educación.



3. Caso práctico

En años recientes se ha incrementado la aplicación de estos agentes tutores inteligentes en la educación. Han sido muchos los modelos cognitivos, de software y de interacción dando como resultado una cantidad importante de aplicaciones integradas. En el presente apartado se describirá de manera general las herramientas software que se utilizaron para crear un tutor que tendrá la función de instruir en el aprendizaje del tema: Conversión de Unidades dentro del Sistema Internacional de Unidades (SI). Al final del apartado presentara una propuesta de aplicación de este tutor dentro de una estrategia de aprendizaje.

3.1 Cognitive Tutor Authoring Tools (CTAT)

Herramientas de Autoría para Tutores Cognitivos de su traducción al Español (CTAT), es una herramienta que ha sido diseñada y en la actualidad se mantiene en constante desarrollo para poder elaborar tutores inteligentes, el equipo de investigación actualmente se encuentra en la universidad Carnegie Mellon (Carnegie Mellon University – CMU) a través del Instituto de Interacción Hombre-Computadora (Human Computer Interaction Institute – HCII) dirigido por Michael Bett y colaboradores, así como la participación de John R. Anderson y colaboradores del Departamento de Psicología de la CMU del cual se fundamento la mayoría del desarrollo y modelado de los procesos cognitivos para ser aplicados en la instrucción utilizando agentes tutores inteligentes, el resultado de estas investigaciones dentro del campo de ciencias cognitivas dio como resultado un modelo teórico de aprendizaje conocido como Adaptive Control of Thought – Rational (ACT-R) o Control del Pensamiento Adaptativo – Racional de su traducción al Español, este modelo ha permitido modelar diferentes procesos cognitivos de manera teórica, apoyándose principalmente en experimentos de psicología y analizando el comportamiento del cerebro a través del estudio de imágenes de resonancia magnética cuando los individuos se presentan dentro de un ejercicio de aprendizaje.

Este software permite capturar y retroalimentar al alumno respecto a una unidad de aprendizaje en particular, esto es, si queremos que el alumno aprenda a sumar el tutor es instruido dentro de los diferentes escenarios para este pueda hacer una retroalimentación efectiva. El modelo propuesto por Anderson (2007) sirve al tutor inteligente para que este vea todas las posibilidades y al profesor para diseñar la secuencia que debe de seguir un problema determinado, de ahí se

tienen que describir los pasos que el alumno aplicará una vez que interactúe con el tutor; estos pasos son evaluados por el tutor y en caso de que al alumno se le presente un problema el tutor tiene la posibilidad de brindar "tips" o "hints" para que el alumno avance en la resolución del problema y logre avanzar en su aprendizaje.

La propuesta del grupo de investigación en CMU es la de crear una herramienta que sirva de investigación para que los diferentes investigadores educativos puedan obtener información dentro de sus experimentos y generar la información correspondiente para analizar el proceso enseñanza aprendizaje. A continuación se describirá la manera de crear un tutor en particular y se presentará una propuesta de aplicación en una secuencia de aprendizaje.

3.2. Descripción de funcionamiento del tutor inteligente

Como interfaz gráfica se utilizó el software Adobe Flash Professional CS5© (AFP), permitiendo crear una interfaz amigable y de fácil interacción con el estudiante, el núcleo del tutor está programado a través de LISP que es el software sobre el cual se ejecuta ACT-R y la manera en la que la comunicación se lleva a cabo entre ACT-R y Adobe Flash Professional CS5© es a través de CTAT, este se instala a través de un componente en AFP. Es importante mencionar que este tipo de herramientas están logrando tener resultados que ponen a prueba diferentes aspectos del proceso enseñanza-aprendizaje, así como la validación de teorías e investigación educativa.

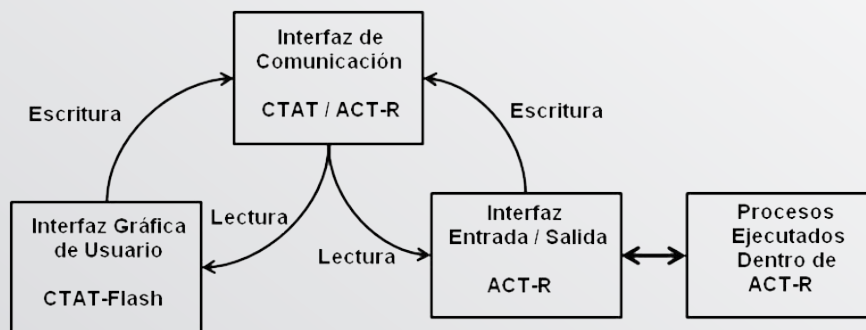


Figura 1.
Interfaz entre
Adobe Flash Professional CS5©
y ACT-R para intercambio de información.



La interfaz gráfica se visualiza utilizando el reproductor Adobe Flash Player®, que es un software de aplicación libre de distribución y uno de los reproductores más utilizados en el mundo. Aquí se establece la comunicación con el software CTAT, el cual a su vez establece la comunicación con ACT-R (ver Figura 1). Cabe mencionar que se debe de llevar a cabo un proceso de integración de las herramientas; una de las ventajas al respecto del software CTAT es la modularidad que este presenta, ya que contiene extensiones que son agregadas a Adobe Flash Professional CS5® a través de Adobe Extensión Manager®, permitiendo por lo tanto hacer estándar los componentes de los tutores inteligentes y poderlos diseñar de una manera más sencilla; concentrando la atención en el diseño de la actividad y los contenidos a ser presentados ante el alumno, permitiendo que los profesores se enfoquen más en la parte didáctica y del proceso de aprendizaje.

La siguiente figura muestra la interfaz gráfica del software CTAT, en ella se pueden apreciar las diferentes áreas de las que se compone. El área de mayor tamaño es dónde se van describiendo las secuencias o pasos de un problema en particular, dentro de cada uno de estos pasos se pueden definir los errores más comunes por parte de los estudiantes e instruir al tutor para que de aviso al usuario de que la acción que acaba de realizar es incorrecta.

Es importante aclarar que este software no es visualizado por el alumno, el alumno sólo interactúa con la interfaz gráfica del tutor, CTAT se ejecuta de manera paralela al tutor y permanece en segundo plano, apoyando al software en AFP, en dónde el estudiante realmente es enfrentado con la actividad de aprendizaje.

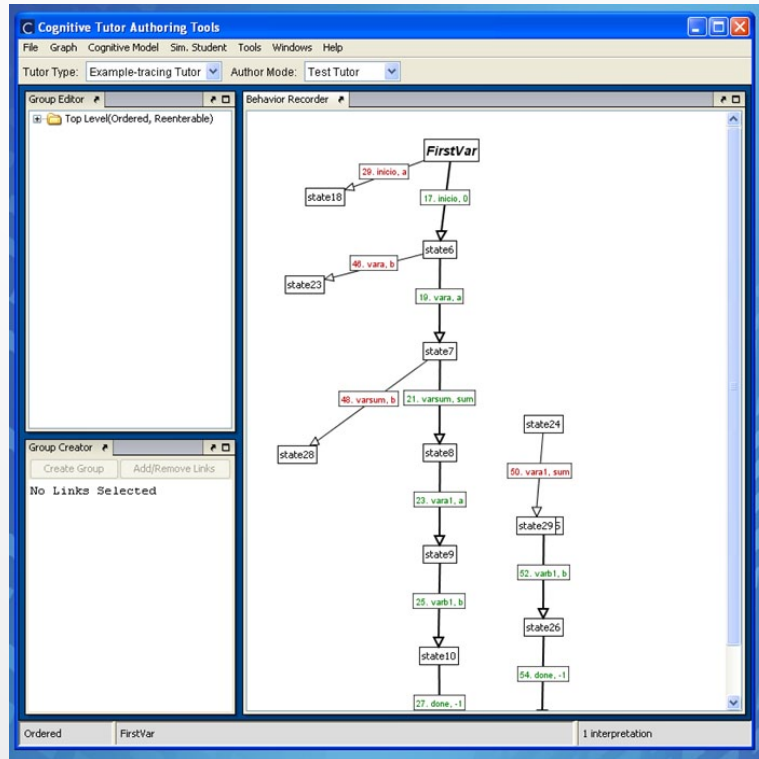


Figura 2.
Pantalla de software CTAT.
Se presentan los estados o pasos de una actividad.
El texto en verde corresponde a un paso correcto y el texto
en negro a un posible error que comenta del alumno.

3.3 Creación del tutor inteligente

Uno de los problemas de aprendizaje en el área de física se presenta cuando el alumno tiene que trabajar con el manejo de unidades de medición; en la mayoría de los currículos a nivel bachillerato los primeros temas que son abordados tienen que ver con el manejo de unidades y su relación con el Sistema Internacional de Unidades (SI), así como la relación de estas unidades respecto a otros sistemas, como son el Sistema Inglés o el MKS (Sánchez, 2009). El tutor que aquí se presenta para tiene por objetivo desarrollar la habilidad en los estudiantes de hacer la manipulación y conversión correspondiente además de permitir llevar al alumno dentro del problema, retroalimentarlo si existiera alguna situación incorrecta y por último si no hubiera posibilidad

e avance respecto al aprendizaje ofrecer la respuesta correcta. Mencionado lo anterior se comenzará con la descripción para la creación del tutor que sirva a los estudiantes a desarrollar sus habilidades en el manejo de los sistemas de unidades.

3.3.1 Diseño y creación de interfaz gráfica

Para comenzar se utilizará el componente CommShell de CTAT, el cual es integrado en AFP y servirá como interfaz de usuario, la siguiente figura muestra el entorno de AFP con el respectivo CommShell en blanco, sin información esta plantilla viene incluida en CTAT y tiene la ventaja de permitir una manipulación aceptable, incluso para alguien que tenga nulos conocimientos de computación. La siguiente figura muestra un ejemplo de la plantilla.

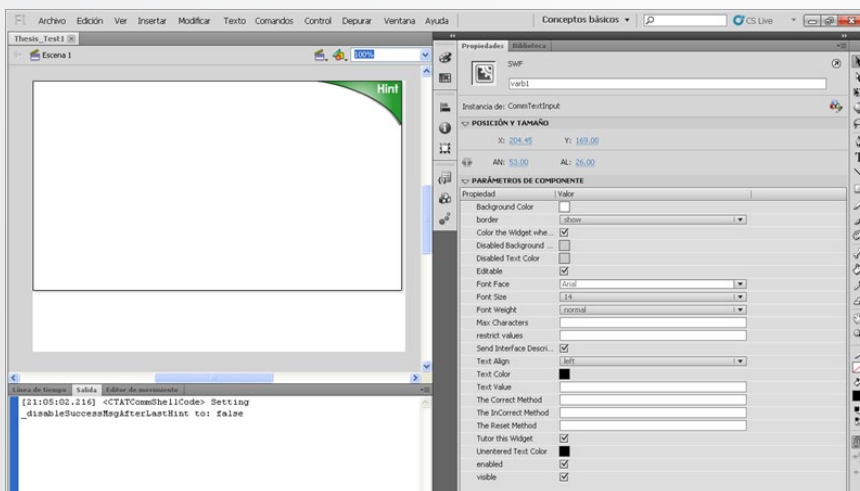


Figura 3.
Pantalla de
Adobe Flash Professional©
con área de trabajo de CommShell.

Después se continúa con la creación de cuadros de texto con las instrucciones correspondientes a la actividad; este texto se puede agregar utilizando la herramienta de texto de AFP, la elección y colocación de contenidos depende de cada diseñador de contenidos y el objetivo que se quiere lograr. La siguiente figura muestra el texto creado para el tutor.

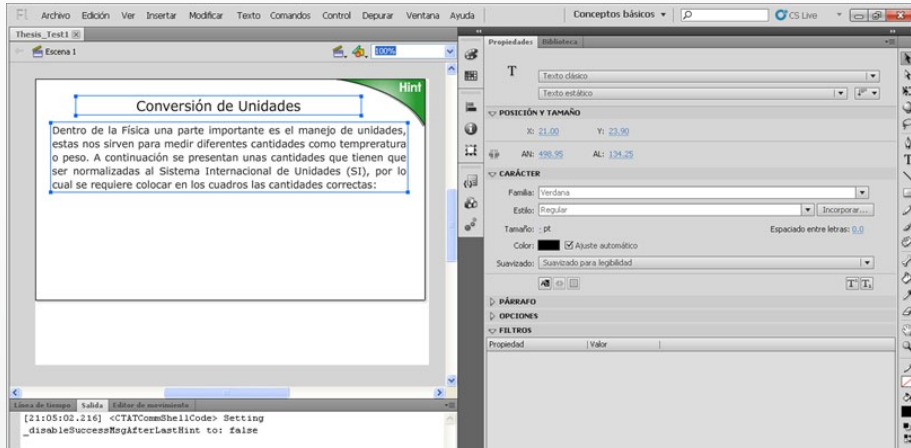


Figura 4.
Asignación de texto
en la interfaz de usuario del tutor.

La edición de texto es muy similar a un procesador de texto y permite tener diferentes estilos de letra, de tal manera que la presentación visual sea atractiva para el estudiante. Una vez terminado el texto para poder informar al usuario final que es lo que debe de llevar a cabo, se procede con crear cuadros de texto, en dónde el usuario colocará los valores correspondientes solicitados una vez que el tutor este en funcionamiento. La siguiente figura muestra la asignación de cuadros de texto y las cajas de texto con la información correspondiente.

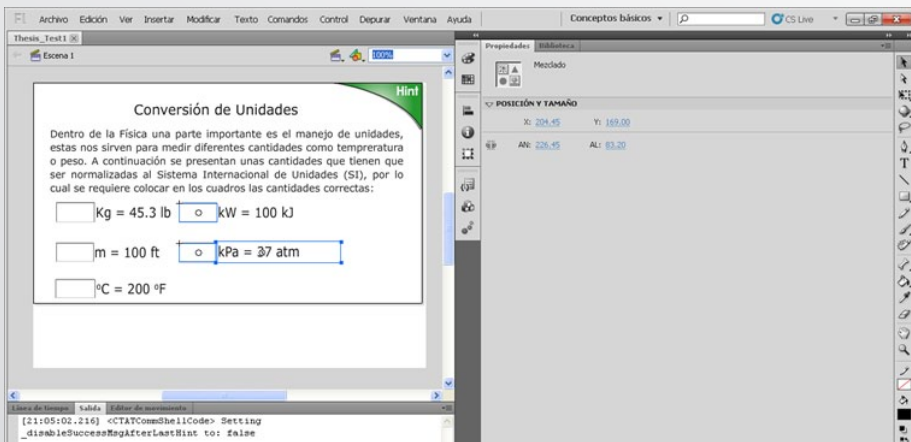


Figura 5.
Inserción de cuadros de texto
para captura de resultados en el tutor.

Es importante ver la parte del diseño, esto es más dejado a la imaginación y creatividad que el diseñador de contenidos considere pertinente, para el presente tutor se contempla un esquema de fácil comprensión, dando énfasis en las instrucciones para que el alumno interprete de una manera lo más clara posible lo que se le pide a resolver, ya que el colocar mucho texto en una actividad de aprendizaje tiende generar pérdida de interés. Por último se debe de crear un botón que permita determinar al tutor que la actividad ha sido terminada por parte del usuario, para eso se coloca un botón de termino; la siguiente figura muestra este botón.

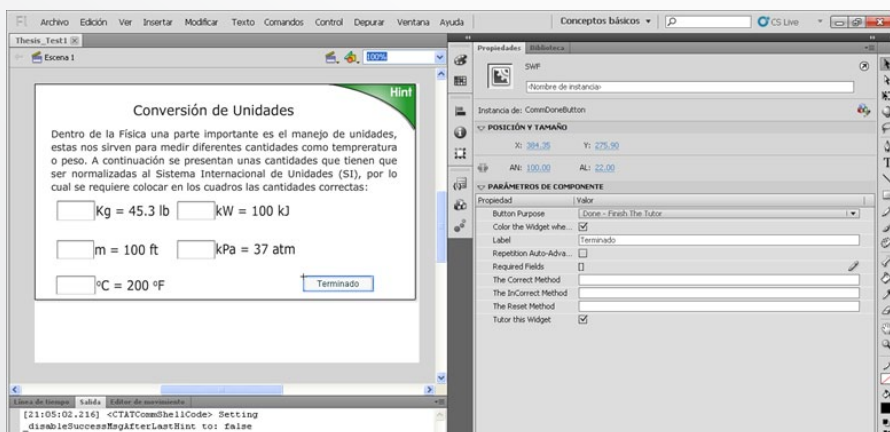


Figura 6.
Pantalla que contiene el botón de término de la actividad y detiene la ejecución del tutor.

Se debe de mencionar que el presente tutor sólo utiliza texto y cuadros de texto para la interacción con el alumno, la herramienta CTAT contiene más posibilidades de diseño con el empleo de imágenes y animaciones así como otras actividades que permiten diseñar actividades más integrales y de mayor dinamismo, el objetivo de trabajar con el presente tutor se debe a que este sirva como ejemplo de aplicación sencilla y demostrar que no se requieren muchos recursos para poder presentar la facilidad de implementación.

Hasta el momento se tiene la parte de creación de interfaz gráfica de usuario lo que permitirá al usuario poder interactuar con el tutor inteligente, a continuación se describirá como se genera la

instrucción al tutor en base a la interfaz de usuario y se le colocan los mensajes de ayuda para el alumno, además de determinar cuándo una operación es incorrecta y cuando no.

3.3.2 Entrenamiento del tutor inteligente

Para comenzar con la instrucción del tutor inteligente se debe de tener los dos programas en ejecución, tanto CTAT como Adobe Flash Professional CS5© (AFP); esto permite tener la comunicación correspondiente entre ellos y uno permitirá que el siguiente pueda agregar los pasos o eventos a analizar, la siguiente figura muestra cómo los dos programas están en ejecución.

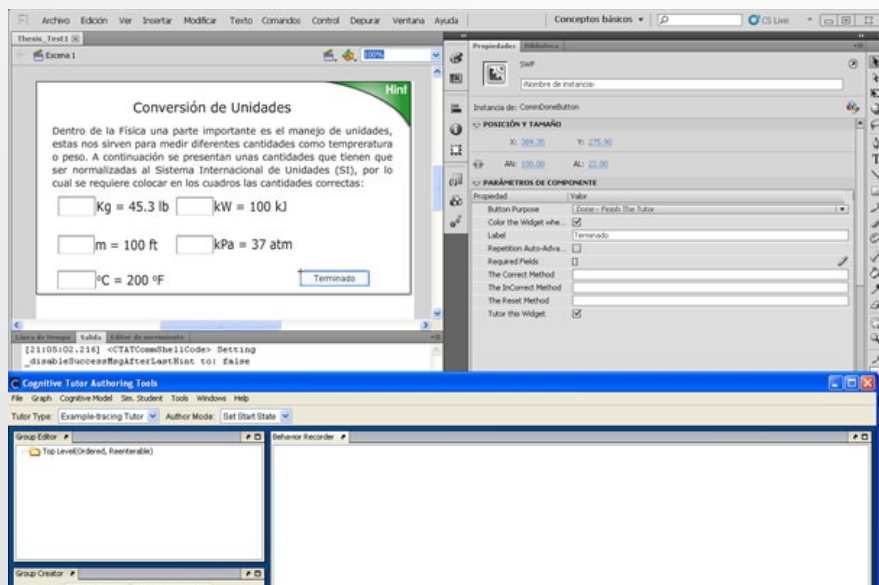


Figura 7.
Pantalla que los dos
programas en ejecución:
CTAT y AFP.

CTAT permite registrar los eventos que se llevan a cabo dentro de la interfaz en AFP, para iniciar este proceso se debe de primero crear un nuevo estado en CTAT que registre los eventos de la interfaz AFP, esto se llevará a cabo cuando la interfaz se ejecute y se comiencen a agregar datos dentro de los cuadros de texto.

Como regla de operación básica en CTAT, este considera que todos los eventos que se lleven a cabo por primera vez serán los correctos. La siguiente figura muestra como CTAT captura el dato del primer cuadro de texto y este se considera como un dato correcto.

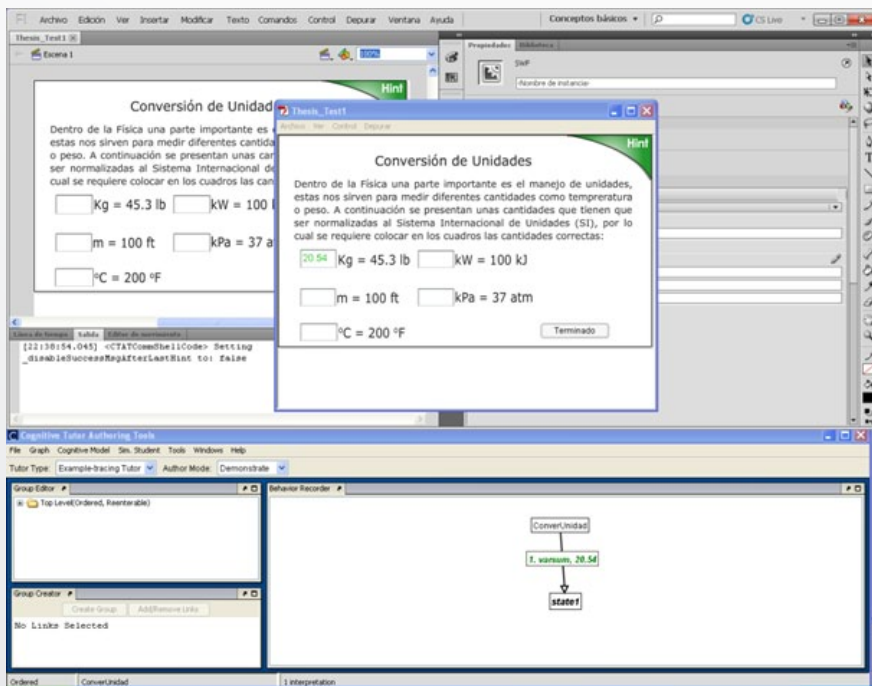


Figura 8.

Captura de información correcta en la interfaz de usuario para entrenamiento del tutor, el texto en color verde se considera como información correcta por parte del tutor.

Toda la secuencia de pasos ira generando un conjunto de estados, esto se conoce como traza, por ejemplo todas las respuestas correctas se capturan dentro de la interfaz gráfica y CTAT registra estos eventos, generando un conjunto de estados que recordará el tutor para poder apoyar en la retroalimentación de la instrucción y preparando el escenario para que los estudiantes puedan interactuar con el tutor.

Una vez completada la traza de la actividad se procede a oprimir el botón con la leyenda: "Terminado", esto avisa al tutor que la actividad. Con los valores correctos se ha concluido el

entrenamiento y se procede a detener el software y las comunicaciones con ACT-R. La siguiente figura muestra el llenado de las casillas del tutor así como la conclusión de la traza, esto se identifica con la leyenda “Done” por parte del tutor.

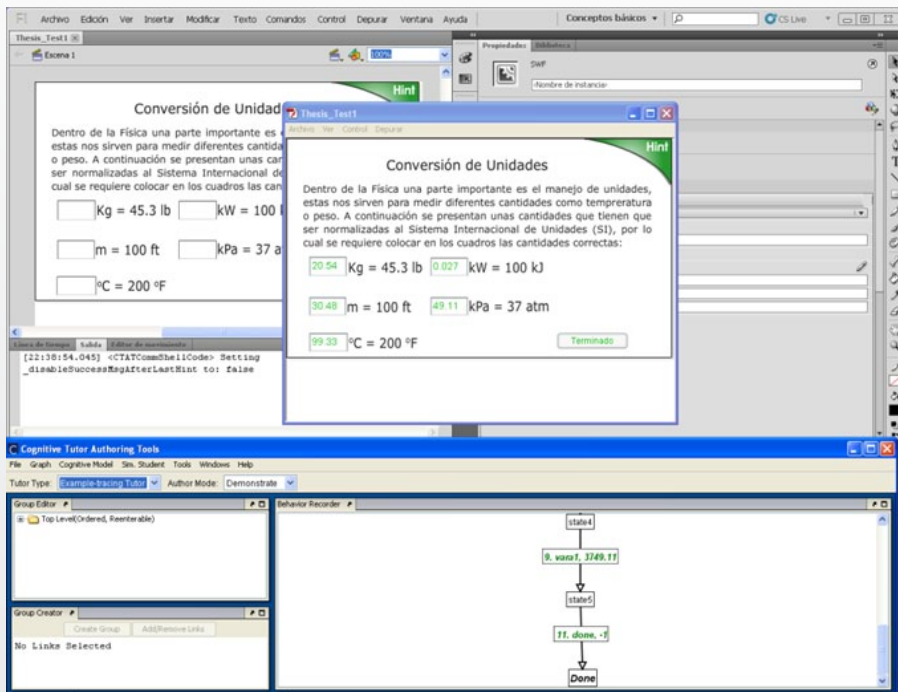


Figura 9.
Traza completa de entrenamiento para el tutor inteligente, una vez concluidos los pasos se procede a dar click en el botón: **Terminado**.

Hasta este momento el tutor ya tiene las instrucciones necesarias para poder validar la actividad, sabe cuáles son los valores correctos y procede a guardarlos en la memoria “procedural” de ACT-R, a partir de ahora se debe de entrenar al tutor cuando un valor capturado es incorrecto, para que este a su vez pueda dar aviso al usuario y poder retroalimentarle, la descripción de asignar los errores al tutor se hará para el primer valor, el resto del entrenamiento opera de manera similar.

La siguiente figura muestra cómo es que se le asigna al tutor el valor incorrecto de un paso, se puede ver que la acción que se captura es incorrecta al ser una letra y esto se le debe de indicar al tutor, a manera de referencia el texto cambia de color verde a rojo

para hacer la indicación visual de que este se trata de un valor incorrecto.

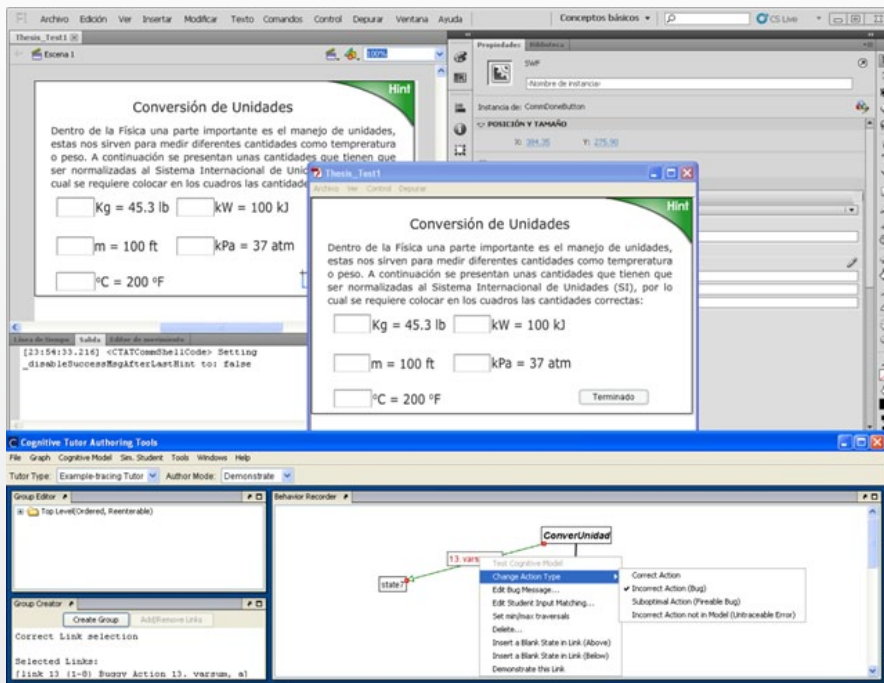


Figura 10.
Determinación de un error al momento
de ser capturado por el usuario.

Este tipo de opciones son útiles para que el estudiante tenga un apoyo dentro de la operación y funcionamiento del tutor y no cometa errores de captura dentro de la herramienta.

Para poder dar retroalimentación al estudiante se tiene una ventana donde se capturan los mensajes que darán soporte al alumno y pueda continuar con la actividad. Estos mensajes aparecerán cuando se da click en el botón: "hint", se tiene la opción de colocar tres mensaje de ayuda, se pueden agregar más mensajes dependiendo de la complejidad de la operación o actividad a realizar, pero una recomendación es colocar tres, dos que den pista fáciles de comprender y asociar y una última indicando el valor correcto y a ser colocado en la interfaz de usuario (LearnLab, 2009), esto con la finalidad de que no se detenga el trabajo con el tutor. Para ingresar estos mensajes de apoyo estos se encuentran dentro del estado correcto, la siguiente

figura muestra la ventana con los mensajes capturados de ayuda.

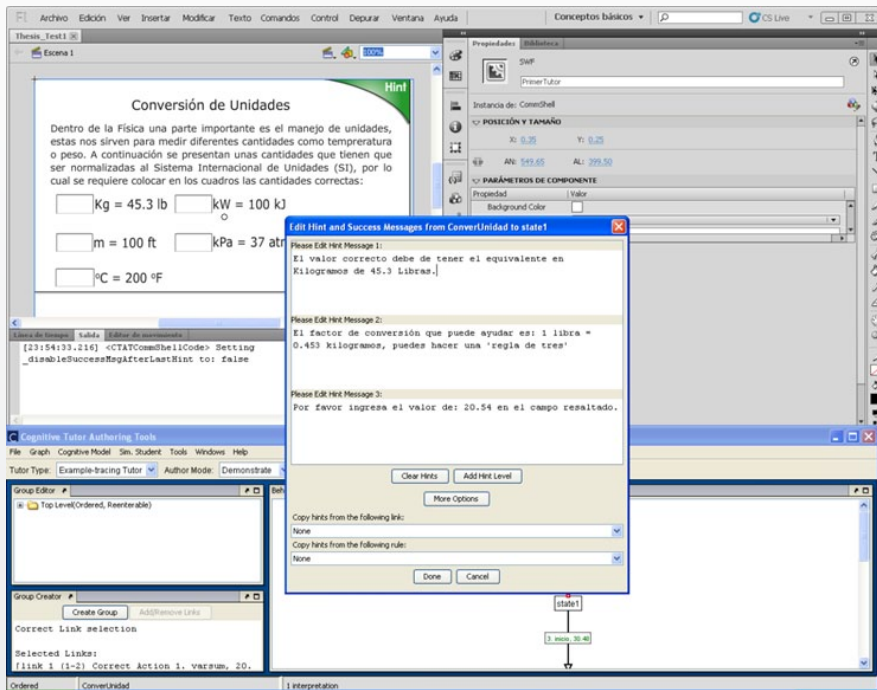


Figura 11.
Modificación del mensaje de error para notificar al usuario de una acción incorrecta.

Se puede observar en la figura anterior que los mensajes que se colocan dan una idea de cuáles son los datos que debe de contener la caja de texto, sí el alumno tiene problemas para poder determinar qué valor debe de ir, el tutor al final muestra cuál es ese valor. Aquí se debe de mencionar que todos los intentos y pasos llevados a cabo por el alumno son registrados para poder tener un análisis de la resolución de problemas para futuros trabajos de investigación.

3.3.3 Validación del tutor inteligente

El realizar las pruebas de la mayoría del software que se desarrolla implica riesgos y se debe de tener plena consciencia de que hacer una prueba en particular no es sólo el ejecutar las instrucciones, sino colocarse en el lugar del usuario final, realizar los pasos más

probables que este usuario realizaría, dentro del presente experimento se considera que el usuario final es un estudiante de primer año del sistema de bachillerato en el sistema educativo mexicano, lo que equivaldría en otros países a referirse a un grado 10 u 11, de igual manera este tipo de tutores se puede extender a niveles superiores de licenciatura o ingeniería. Así se tiene una idea de lo que esto implica respecto al tutor y como las pruebas deben de ir orientadas a este tipo de usuario. A continuación se muestra y describe una alternativa de llevar a cabo las pruebas del tutor y como emplear la herramienta como un recurso didáctico que mejorará el proceso enseñanza aprendizaje.

Ya terminada la creación del tutor en Adobe Flash Professional CS5© así como la conclusión y validación de las trazas en CTAT, se debe de indicar al software que se utilizará en modo de prueba para que ahora se envíe la información a la interfaz de usuario proveniente de CTAT y ACT-R. La siguiente figura muestra el software ejecutándose de manera paralela (APF, CTAT y ACT-R), al iniciar no muestra nada anormal en su comportamiento.

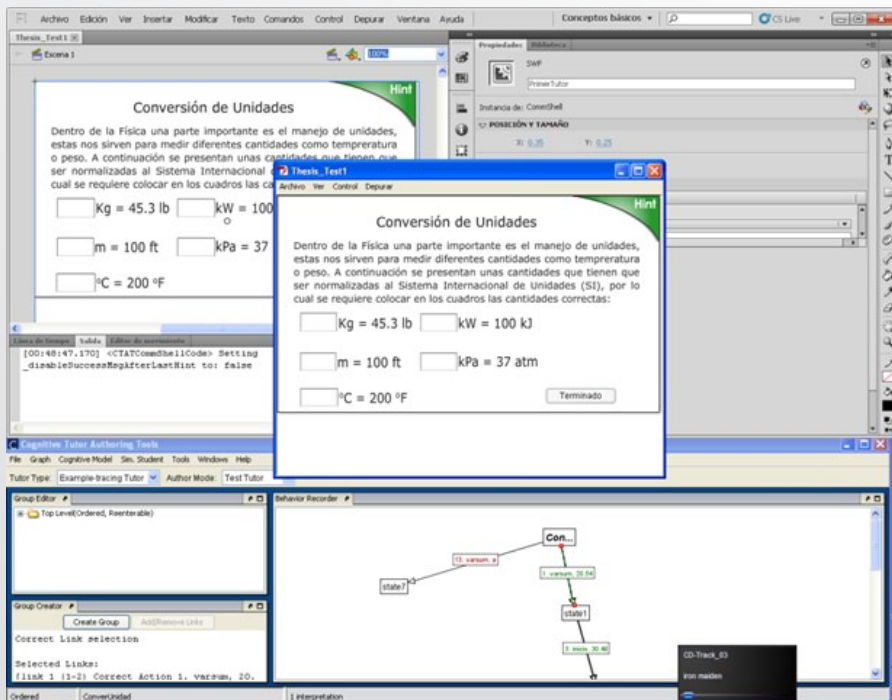


Figura 12.
Inicialización del tutor
para pruebas de evaluación.

En la imagen anterior se presenta todo el conjunto de aplicaciones que son ejecutadas de manera concurrente, esto puede causar confusión, pero en la práctica sólo se muestra el tutor ejecutándose de manera aislada, el resto de las aplicaciones no se muestran a los estudiantes.

De este paso en adelante, el profesor comienza a capturar los datos dentro de los cuadros de texto correspondientes a la resolución del problema; dentro del proceso de captura se recomienda cometer errores de manera intencionada para verificar el comportamiento del tutor al momento de requerir retroalimentación el estudiante. La siguiente figura muestra el comportamiento del tutor cuando le es solicitada la ayuda a través de dar clic en el botón "hint".

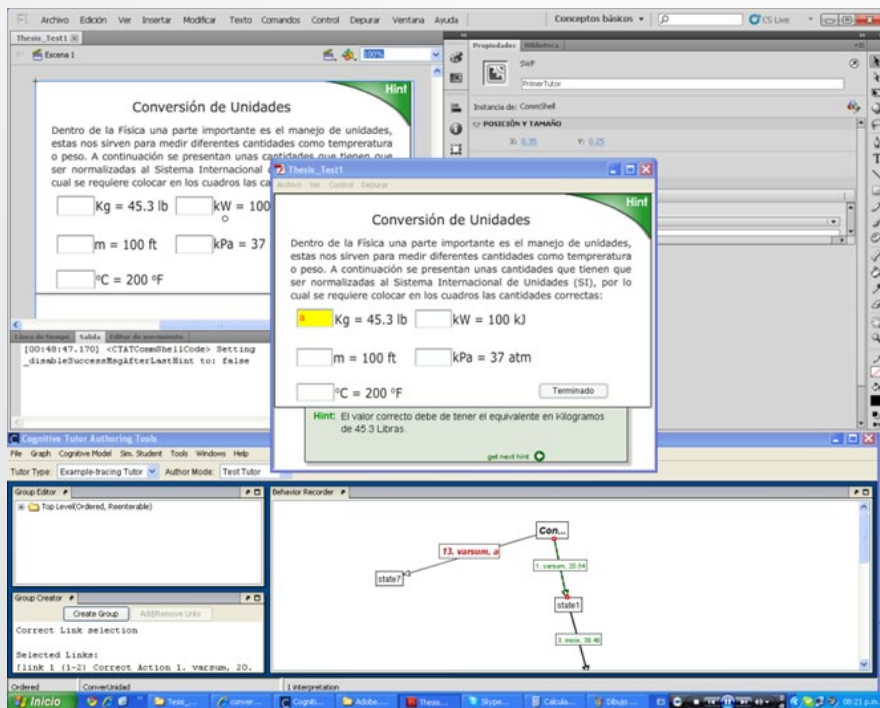


Figura 13. Despliegue de la ayuda por parte del tutor para dar pistas al usuario de la actividad a realizar.

En caso de existir inconsistencias de comportamiento por parte del tutor este se puede volver a entrenar, ya sea desde el inicio del tutor o en los estados donde se presenten los errores de



comportamiento. El profesor que diseñe la actividad con el tutor debe de aportar su experiencia para poder presentar un tutor lo suficientemente competente y que realmente apoye al estudiante dentro de la instrucción. El tutor debe de ser visto como un apoyo al aprendizaje más no como un reemplazo del profesor, este debe de tener la capacidad de transmitir el conocimiento del profesor a través de la ayuda y las guías puestas para tal efecto.

Para concluir este apartado, es importante mencionar que este tipo de tutores se pueden exportar a ambientes Web, pueden convivir con entornos de aprendizaje en línea, virtual y mixto; a manera de ejemplo, este tipo de tutores se puede insertar dentro de una actividad a desarrollar por el estudiante mediante un ambiente virtual de aprendizaje, por ejemplo MOODLE, la convivencia entre ambos sistemas permite tener un conjunto de tutores desarrollados y que pueden ser definidos de manera optima, sin comprometer la calidad de la instrucción.

3.4 Propuesta de Secuencia Didáctica

Como ejemplo de aplicación a continuación se propone una secuencia didáctica donde el tutor puede ser incluido para el apoyo de actividades docentes. El tutor que se presenta en este trabajo se enfoca al aprendizaje por parte del estudiante del manejo y la manipulación de conversión de unidades de medida dentro de los diferentes sistemas que existen en diferentes partes del mundo.

En la primera etapa de la secuencia se propone que el profesor exponga utilizando los recursos en el aula y mediante una estrategia de aprendizaje el tema de los sistemas de unidades y la relación que se guarda entre ellos, el profesor presentará los diferentes tipos de unidades aplicadas al tiempo, masa, peso, temperatura, etc. Y mostrará las unidades dentro de los diferentes sistemas de referencia; una vez que se presentan los ejemplos más representativos se presentan las fórmulas que permiten la conversión entre las diferentes unidades y dentro de los diferentes sistemas, se realiza una serie de ejercicios en los cuales el profesor describe como se lleva a cabo cada operación y solicita que los estudiantes resuelvan de manera colaborativa los ejercicios



propuestos, se recomienda dejar ejercicios de práctica. Todo lo anterior debe realizarse dentro de una hora u hora y media de clase, con el objetivo de que el estudiante tenga los conocimientos básicos del tema. La siguiente clase el profesor presentará a los estudiantes la interacción con el tutor y la manera de manipularlo, se propone que el estudiante se enfrente con tutores diferentes a los relacionados con el tema propuesto en clase, el objetivo de esta actividad es que el estudiante se familiarice con el manejo del tutor; de igual manera se requiere una clase y un aula con computadoras para que los estudiantes puedan llevar a cabo el entrenamiento del tutor, un par de alumnos por computadora es una efectiva estrategia ya que entre pares se tiene una mejor confianza de trabajo y apoyo entre los mismos estudiantes.

Por último una tercera sesión servirá para presentar a los alumnos los tutores correspondientes al tema visto, para este caso el relacionado con la conversión de unidades, esto será de gran apoyo ya que los estudiantes al ser preparados en el manejo de los tutores no tendrán dificultad de operación de los mismos. Los resultados en otros experimentos han demostrado que los estudiantes mejoran su rendimiento respecto al aprendizaje en la resolución de problemas, de manera cualitativa los estudiantes en general comentan que sienten una mayor confianza de operar el tutor y después sienten mayor seguridad al resolver problemas sin el mismo ya que la ayuda del tutor les sirvió para comprender mejor el proceso que estaban desarrollando (Hernández 2012).

Conclusiones

Se ha presentado de manera general una herramienta aplicada a la enseñanza, novedosa y con muchas aplicaciones por ser llevadas a cabo. El trabajo que se encuentra por delante es mucho y se tienen diferentes proyectos derivados del presente trabajo. Actualmente se están desarrollando tutores a nivel bachillerato y se comenzará en breve la creación de proyectos a nivel licenciatura. Es importante mencionar que esta área de trabajo educativo no tiene mucha presencia en países en vías de desarrollo, esto por el lento desarrollo tecnológico que presentan estos países, pero se puede ver en este trabajo que en la actualidad esto ya no es limitante o dependiente de la tecnología, sólo es necesaria una conexión a internet y el entusiasmo por parte de los profesores de querer iniciar un camino de trabajo y desarrollo de materiales que sirva a sus estudiantes, mejorando con esto la calidad en la educación.

Una de las ventajas respecto al software es que CTAT es de libre acceso, eliminando posibles licencias y facilitando la inclusión de la misma dentro de la labor docente. El software Adobe Flash Professional CS5© requiere el pago de una licencia, pero CTAT también puede ser programado utilizando el lenguaje de programación Java™, el cual es un lenguaje de programación orientado a objetos y de libre distribución. El objetivo de presentar CTAT utilizando Flash Professional CS5 es el demostrar que este tipo de herramientas son accesibles y no se requiere un conocimiento amplio en los campos de programación, inteligencia artificial y sistemas computacionales en general.

CTAT y la mayoría del software desarrollado utilizando inteligencia artificial aplicada a la educación se ha desarrollado en países de habla inglesa, se debe de comenzar a desarrollar este tipo de herramientas dentro del contexto latinoamericano o hispano-parlante para reducir la brecha tecnológica respecto de los países más avanzados. Se ha visto que la tecnología se tiene, sólo hace falta la iniciativa por parte de las instituciones, los investigadores, los profesores y los estudiantes para poder generar más proyectos de investigación.

Agradecimientos

Claudia Rosado Guzmán agradece a la Universidad del SABES el apoyo brindado para la conclusión de sus estudios de maestría. Daniel Sánchez Guzmán agradece a la Secretaría de Investigación y Posgrado del Instituto Politécnico Nacional a través de los recursos derivados del proyecto SIP-2012-0714 y al Consejo Nacional de Ciencia y Tecnología de México por la beca del Sistema Nacional de Investigadores, para la generación del presente artículo.

Claudia Rosado Guzmán

Maestría en Ciencias en Física Educativa.

Universidad del SABES.

Profesor Tutor.



Daniel Sánchez Guzmán

Doctorado en Tecnología Avanzada.

Centro de Investigación en Ciencia Aplicada y Tecnología Avanzada
Unidad Legaria del Instituto Politécnico Nacional.

Jefe del Departamento de Estudios de Posgrado de CICATA – Legaria.



REFERENCIAS

Anderson J. R., Bothell D., Lebiere C., Matessa M. (1998). An integrated theory of list memory. *Journal of Memory and Language*, No. 38, pp. 341-380.

Anderson, J. R. (2007). *How Can the Human Mind Occur in the Physical Universe?* Oxford University Press, Inc. Pp. 39-80. ISBN 978019-532425-9.

Chou C., Chan T. y Lin C. (2003). Redefining the learning companion: the past, present, and future of educational agents. *Computers and Education*, No. 40, pp. 255-269. Elsevier. ISSN: 0360-1315.

Fainholc B. (2008). De Cómo las TIC's Podrían Colaborar en la Innovación Socio-Tecnológico-Educativa, *RIED*, Vol. 11, No. 1, pp. 53-79.

Fogel L., Owens A. y Walsh M. (1966). *Artificial Intelligence through Simulated Evolution*. New York, John Wiley & Sons, pp. 115-180.

Hernández G. F. (2012). *Agentes Tutores Inteligentes Aplicado a la Enseñanza de Circuitos Eléctricos en Estudiantes de Bachillerato*. Tesis de Maestría, Centro de Investigación en Ciencia Aplicada y Tecnología Avanzada Unidad Legaria del Instituto Politécnico Nacional, pp. 60-94. México.

Kenneth A. (2008). Evolving Intelligent Agents: 50th Year Question. *IEEE Computational Intelligence Magazine*, pp. 12-17.

LearnLab. (2009). Pittsburgh Science of Learning Center. <http://www.learnlab.org/>. Consultado en Marzo de 2012.

Les J., Cumming G. y Finch S. (1999). Agent systems for diversity in human learning. *Proceedings of AI-ED 99, World Conference on Artificial Intelligence in Education*. Le Mans, France, pp. 13-20.

Maes P., (1995). Artificial Life meets Entertainment: Lifelike Autonomous Agents. *Communications of the ACM*, Vol. 38, No. 11, pp. 108-114.

Newell A., Simon H. A. (1972). *Human problem solving*. Englewood Cliffs, NJ, Prentice Hall, pp. 112-134.

Orellana N. (2002). *Estilos de Aprendizaje y Utilización de las TIC en la Enseñanza Superior*. Unidad de Tecnología Educativa. Dpto. MIDE. Universidad de Valencia, pp. 235-248.

Russell S., Norvig P. (1995). *Artificial Intelligence: A Modern Approach*. Englewood Cliffs, NJ, Prentice Hall, pp. 448-487.

Salinas J. (2002). *La integración de las TIC en las instituciones de educación superior como proyectos de innovación educativa*. Universidad de las Islas Baleares, pp. 361-344.

Sánchez G. D. (2009). *Agentes Tutores Inteligentes: Diseño e Implementación para la Enseñanza de la Física*. Tesis de Doctorado, Centro de Investigación en Ciencia Aplicada y Tecnología Avanzada Unidad Legaria del Instituto Politécnico Nacional, pp. 8-10, 18-24. México.

Smith D. C., Cypher A., Spohrer J. (1994). KidSim: Programming Agents Without a Programming Language. *Communications of the ACM*, Vol. 37, No. 7, pp. 55-67.

VanLehn K., Lynch C., Schulze K., Shapiro J. A., Shelby R., Taylor L., Treacy D., Weinstein A., Wintersgill M. (2005). The Andes Physics Tutoring System: Lessons Learned. *International Journal of Artificial Intelligence in Education*, No. 15, pp. 147-204.

Vardharisgaran (1999). Mubot Agent a cognition system approach. <http://www.crystaliz.com/logicware/mubot.html> (Visitado en Febrero 2009)