

El uso de tutoriales interactivos en ambientes educativos: un caso práctico

Alicia B. Paur¹; Zulema B. Rosanigo²; Pedro Bramati³; Alfredo Ortega⁴; José P.Cerra⁵

Facultad de Ingeniería – Sede Trelew – U.N.P.S.J.B. Te-Fax (02965) 42 84 02

Resumen

En este artículo se expone un caso práctico, surgido de la necesidad de hallar herramientas alternativas para mejorar el proceso de enseñanza-aprendizaje, que cumplan con los requisitos de calidad de software educativo y contemplen aspectos pedagógicos en etapas tempranas del ciclo de vida del software.

Se trata del producto resultante del proyecto de investigación PI N° 383 en el área de informática educativa denominado: "*Construcción de tutoriales basados en componentes reusables*".

La herramienta en cuestión, brinda al docente de cualquier área, la posibilidad de crear tutoriales como complemento ideal para el desarrollo de su actividad áulica. Al mismo tiempo, el alumno al interactuar con esos tutoriales, puede analizar tantas veces como lo desee el problema y la solución propuesta, observando el comportamiento, incrementando así su capacidad de asimilación y fijación de conceptos.

Palabras claves

Tutorial interactivo – framework – software educativo

¹ Analista Programador Universitario - Investigador Cat. V – Profesor adjunto D.S.E. - apaur2@hotmail.com

² Magister en Ingeniería de Software - Ingeniera Civil – Analista Programador Universitario – Investigador Cat. III - Profesor Asociado D. E. brosanigo@hotmail.com

³ Ingeniero Civil – Investigador Cat. IV - Profesor Titular D. S.E.. bramati@infovia.com.ar

⁴ Licenciado en Informática cachito@ortega.net.ar

⁵ Alumno de Analista Programador Universitario – cerraquza@infovia.com.ar

Introducción

En el proyecto de investigación Nro. 383 nos planteamos como objetivo facilitar la generación de material didáctico en general, y de tutoriales del tipo “enseñanza paso a paso” en particular, que cumplan con los requisitos de calidad de software educativo (SE). Para ello, nos propusimos diseñar una herramienta basada en componentes reusables que permita construir y ejecutar tutoriales interactivos en cualquier dominio del conocimiento, de forma tal, que esa tarea resulte sencilla y simple para el docente o usuario que lo utilice.

Según Galvis Panqueva (1994), desarrollar ambientes de enseñanza-aprendizaje basados en computador sin tener un buen sustento teórico respecto al aprendizaje humano y a las características del computador como medio de enseñanza, pueden replicar, indiscriminadamente, las estrategias de enseñanza-aprendizaje que se conocen y además desaprovechar algunas características útiles del computador. Hace falta entender y aplicar teorías de aprendizaje humano que den sustento al diseño de ambientes de aprendizaje efectivos.

En el desarrollo de todo software, es imprescindible valerse de los métodos, procedimientos y herramientas, que provee la ingeniería de software para construir programas de calidad, en especial, en el SE, deben contemplarse además las pautas que las teorías educativas y de la comunicación nos marcan para lograr nuestro objetivo final, obtener SE de calidad. Esto nos induce a estudiar las distintas teorías educativas y captar de cada una de ellas, las ventajas y características que las hacen aplicables a las diferentes situaciones áulicas que deseamos mejorar.

Uno de los resultados emergentes del proyecto de investigación es: TUTGEN (Tutorial Generator), un software para creación de tutoriales interactivos. Esta herramienta está diseñada para permitir al docente, construir tutoriales del tipo “enseñanza paso a paso” a partir de algunos componentes primarios o básicos -desarrollados especialmente para el dominio de aplicación- el tutorial así creado, a su vez se convierte en un nuevo componente del dominio y por lo tanto puede ser usado para la creación de otros, adaptándose en forma inteligente al nuevo contexto. Ha sido especializado en el área de las construcciones geométricas y es aplicado en la materia Sistemas de Representación del ciclo básico de Ingeniería.

TUTGEN permite al profesor: crear los tutoriales que apoyarán, guiarán y facilitarán la capacitación del alumno, de ser necesario podrá usar otros tutoriales previamente creados como pasos intermedios del nuevo, usando todas las herramientas que estarán a su disposición (cuadros de texto para ingresar conceptos teóricos, recursos visuales, etc.).

La herramienta “aprende” a dominar los tutoriales creados y permite al alumno: interactuar con ellos, ingresando los datos iniciales, variándolos con simplemente “arrastrar” las figuras, reiterar las acciones a partir de cualquier paso, guardar su propio ejemplo con los datos de entrada elegidos, permitiendo así, su entrenamiento para la adquisición de la habilidad pretendida por el profesor, en este caso en particular, en el área de de las construcciones geométricas.

En este artículo se presentan brevemente las características de las principales teorías del aprendizaje, también se describen distintos tipos de Software Educativo, cada uno con la particularidad de adaptarse a un tipo de realidad áulica y de proceso enseñanza-aprendizaje en particular, muchas veces complementándose entre ellos. A continuación se expone un caso práctico, surgido de la necesidad de hallar herramientas alternativas para mejorar el proceso de enseñanza-aprendizaje, que cumplan con los requisitos de calidad de software educativo.

Marco teórico

Según Salcedo Lagos (2002) las aproximaciones al fenómeno del aprendizaje oscilan entre dos extremos: del conductismo al cognitivismo y viceversa.

En un extremo, la teoría del conductismo lo único que ve del sujeto que aprende son las condiciones externas que favorecen su aprendizaje. Habla de un modelo de "caja negra" donde lo fundamental es la programación en pequeños pasos que llevan al logro del objetivo esperado, lo que se manifiesta por la respuesta del sujeto y su reforzamiento.

En el otro, para la teoría del cognitivismo lo que cuenta es el sujeto con todo su campo vital, su estructura cognitiva y las expectativas que tiene. Se habla de un modelo de "caja traslúcida" donde lo que cuenta es el sujeto dentro de su entorno psicológico y social.

Como consecuencia de esto el papel del profesor oscila entre ser un instructor y expositor de clases magistrales, máximo depositario de la información, sugiriendo ejercicios a realizar para reforzar los aprendizajes (conductismo), a ser facilitador y guía de aprendizajes que ayuda a los alumnos a construir conocimientos, pieza clave en el triángulo: alumno-profesor-tecnología (cognitivismo).

El papel del alumno varía entre ser un actor pasivo en el proceso de enseñanza-aprendizaje y un memorizador del conocimiento transmitido por el profesor y los libros, aplicándolos para contestar preguntas y realizar ejercicios que le ayudaran a asimilar los contenidos (conductismo), a ser protagonista activo y centro de las actividades de enseñanza – aprendizaje, gestor de un aprendizaje significativo para construir nuevos conocimientos, con la ayuda de nuevas tecnologías y a menudo en ámbitos colaborativos.

En medio de los enfoques anteriores, es posible encontrar una combinación de ambos, donde no se ignora por completo al sujeto, pero tampoco se lo hace eje principal.

A continuación damos una breve enumeración de las características de las teorías educativas más relevantes y sus principales representantes.

Conductismo (Skinner 1958)

- Formación de reflejos condicionados mediante mecanismos de estímulo- respuesta
- Ensayo y error con refuerzos y repetición.
- Ley del efecto o del resultado de la acción.
- Memorización mecánica.

Aprendizaje significativo (Ausubel 1983)

- Relación con las estructuras cognitivas previas y funcionalidad
- Utilización de organizadores previos.
- Diferenciación-reconciliación integradora que genera una memorización comprensiva

Cognitivismo (Gardner 1987)

- Consideración de diversas etapas en el proceso de aprendizaje.
- Consideración de las interacciones: estudiante - sistema simbólico de los medios.

Constructivismo (Piaget 1989)

- Construcción del propio conocimiento mediante la interacción constante con el medio.
- Equilibrio - desequilibrio - reequilibrio: adaptación y construcción de nuevos esquemas de conocimiento
- Atención al desarrollo cognitivo.

Las teorías no son ni buenas ni malas *per sé*, en determinadas circunstancias, se aplica mejor el conductismo y en otras, el constructivismo o el aprendizaje significativo. Nuestro deber como docentes es aprovechar de cada teoría lo que mejor convenga a la situación, al sujeto, y al tipo de aprendizaje a lograr.

Software educativo

Marques Graells define al software educativo como los programas para computadora creados con la finalidad específica de ser utilizados como medio didáctico, es decir, para facilitar los procesos de enseñanza y de aprendizaje, imitando la labor tutorial personalizada que realizan los profesores y presentando modelos de representación del conocimiento en consonancia con los procesos cognitivos que desarrollan los alumnos.

Un programa de este tipo debe satisfacer algunos criterios para valorar su utilidad y calidad como recurso educativo:

- El contenido debe estar adaptado al nivel de conocimientos de los alumnos y ser pertinente con relación al currículum escolar.
- Debe poseer una clara estructuración de la información con el objeto de facilitar su procesamiento por parte del usuario.
- Ha de definir una estrategia didáctica para mostrar y explicar el contenido, que respete la teoría educativa utilizada por el educador.
- Es fundamental que se contemplen los aspectos pedagógicos en etapas tempranas del desarrollo de un SE.

Podemos distinguir los siguientes tipos de software educativos:

- tutoriales,
- de ejercicios y prácticas,
- de recuperación de información o Bases de Datos,
- simuladores,
- juegos,
- micromundos
- sistemas expertos

Los tutoriales resultan particularmente útiles cuando se requiere alta motivación, retroalimentación inmediata, ritmo propio y secuencia controlable por el usuario parcial o totalmente.

Los sistemas de ejercitación y práctica permiten reforzar la aplicación y retroalimentación por medio de ejercicios tradicionales.

Los sistemas de recuperación de información o Bases de Datos, presentan datos organizados en un entorno estático mediante unos criterios que facilitan su exploración y consulta selectiva para resolver problemas, analizar y relacionar datos, comprobar hipótesis, extraer conclusiones.

Los simuladores pretenden apoyar el aprendizaje por medio de experimentos, de forma que el estudiante descubra conceptos en un micromundo semejante a una situación real. El alumno es agente activo que ejercita aprendizajes inductivo y deductivo mediante la toma de decisiones y adquisición de experiencia en situaciones imposibles de lograr desde la realidad, facilitando el aprendizaje por descubrimiento.

Los juegos educativos, son similares a los simuladores porque apoyan el aprendizaje semejando situaciones, no necesariamente reales pero generalmente más excitantes o entretenidas.

Los micromundos exploratorios, emplean un lenguaje de programación simple que se usa para interactuar en un micromundo, donde se llega a la solución de forma estructurada, con división de problemas en subproblemas, por ejemplo Logo.

Los sistemas expertos son capaces de representar y razonar acerca de algún dominio rico en conocimientos, con el ánimo de resolver problemas y dar consejo a quienes no son expertos en la materia. Cuentan con una base de conocimientos construida en base a la experiencia humana, y a partir de ella y con reglas de alto nivel es capaz de hallar o juzgar la solución a algo.

Los entornos tutoriales interactivos

Tienen como objetivo enseñar un determinado contenido académico a través de la interacción del usuario con el programa informático. Según las orientaciones pedagógicas adoptadas en su diseño, los tutoriales dirigen y orientan, en mayor o menor medida, el aprendizaje de los alumnos.

Son muy útiles en aquellas áreas del conocimiento donde se presentan problemas cuya solución requiere seguir determinado procedimiento o proceso paso a paso o adquirir destrezas. Frecuentemente en un libro, por ejemplo, el alumno se encuentra con la resolución impresa del problema en un solo paso, con todos los procesos intermedios obviados; un tutorial interactivo posibilita al alumno indagar todas las etapas intermedias hasta lograr el conocimiento o adquirir la destreza buscada.

Estos programas dirigen el aprendizaje de los alumnos mediante una teoría subyacente conductista de la enseñanza, guían los aprendizajes y comparan los resultados de los alumnos contra patrones, generando muchas veces nuevas ejercitaciones de refuerzo, si en la evaluación no se alcanzaron los objetivos de aprendizaje. Se han desarrollado modelos cognitivistas, donde se usa información parcial, y el alumno debe buscar el resto de la información para la resolución de un problema dado.

Un caso práctico

A continuación se describirá una herramienta desarrollada para facilitar la tarea de crear tutoriales en un dominio del conocimiento como recurso didáctico, que colaboren y potencien los procesos de aprendizaje en la educación superior y sirvan de apoyo al proceso de enseñanza-aprendizaje.

Motivaciones y objetivos

En el desarrollo de la currícula de una carrera universitaria suelen existir tiempos diferenciados en cuanto a sus potencialidades: *tiempos de adquisición de conocimientos*, por un lado, y *tiempos de aplicación de conocimientos*, por el otro. Muchas veces, no existen correlatividades inmediatas de ejecución de los mismos, y además, hay períodos intermedios entre ambos, que pueden llegar a ser excesivamente extensos, ya sea por problemas particulares en el diseño propio de la currícula, o por causas debidas a un prolongado período de permanencia de los alumnos dentro de la carrera, por lo que el intervalo entre el *tiempo de adquisición* y el de *aplicación de conocimientos* puede ser mucho más amplio todavía.

El transcurrir del tiempo, sin aplicación de las habilidades oportunamente desarrolladas, disminuye, limita o borra sistemáticamente todo tipo de conocimiento adquirido. Cuanto mayor es el mismo, mayor es la deficiencia.

Este problema ha sido detectado en varias áreas de la carrera de Ingeniería. Por ejemplo, tomemos como base de análisis la temática referida al problema que se presenta cuando se desea realizar *empalmes de curvas planas*, que es tarea particularmente específica de la asignatura Sistemas de Representación (Unidad III del Programa Analítico de la asignatura), de la carrera de Ingeniería Civil.

Para esta situación, el intervalo que media entre el *tiempo de adquisición* y el *tiempo de aplicación de conocimientos*, es en el mejor de los casos de dos años de duración, porque la utilización pragmática de los contenidos oportunamente incorporados, se manifiesta en las cátedras: Construcción de Edificios e Instalaciones, Arquitectura y Urbanismo, Construcciones Metálicas y de Madera, etc., que son materias que están insertas en el último período formativo de la currícula (cuarto o quinto año), y siendo que Sistemas de Representación lo está en el segundo cuatrimestre del segundo año de la carrera. En el medio, no hay ninguna asignatura en la que realmente se realice alguna aplicación práctica de las habilidades que los alumnos han adquirido con anterioridad.

En este estado de situación, si llegara a ser necesario resolver algún problema puntual que involucre aquellos conocimientos, implicaría necesariamente realizar una lectura obligada de bibliografía especializada en el tema para recuperar parte o toda la masa de información adquirida en su momento.

Los desarrollos temáticos bibliográficos a los que nos referimos son obviamente, representaciones estáticas, por lo que la interpretación de las construcciones y/o los procesos constructivos, se convierten muchas veces en un camino bastante dificultoso y tedioso de seguir y de entender.

La representación gráfica impresa, es en general un esquema que indica de manera resaltada el resultado final del problema, y con trazos menos significativos el proceso constructivo que llevó a obtener el fin propuesto originalmente. Hay agregados un conjunto de letras y números que individualizan diferentes partes constitutivas del procedimiento como una manera de mejorar el entendimiento del lector, todo este código simbólico usado está apoyado paralelamente con una explicación teórica del desarrollo intrínseco en sí.

La percepción visual es una de las capacidades de asimilación de conocimientos mayores con las que cuenta el alumno, pero si además, a ella se le agrega colorido y movimientos entonces se superlativiza esta capacidad. Y si supletoriamente, se utiliza algún instrumento o herramienta que permita al estudiante avanzar y retroceder cuantas veces lo crea necesario, desarmando y rearmando el camino crítico primariamente inducido, las posibilidades de incremento de captación son aún mayores.

Ante esta situación, se consideró oportuno desarrollar un tutorial dinámico que, de una forma amena y rápida, proporcione los conocimientos necesarios para el aprendizaje de los conceptos básicos referidos a, en este caso particular, *empalmes entre curvas planas*.

Puesto que el esfuerzo de desarrollo y programación de un SE es grande y la problemática presentada en la temática descrita ocurre también en muchas otras áreas, surgió la idea de diseñar una herramienta que permita al docente, a partir de algunos componentes -desarrollados especialmente para un dominio de aplicación- construir tutoriales del tipo enseñanza “paso a paso”, para que el alumno pueda ejercitarse tanto como lo necesite, dando la posibilidad de servir como herramienta de consulta para cualquier tipo de usuario: alumno, docente o usuario externo.

Nos propusimos como meta desarrollar un framework que permita generar tutoriales en cualquier dominio del conocimiento. Es decir, una aplicación semicompleta que aporta la estructura y funcionalidad de los componentes comunes, establece las interacciones entre los objetos intervinientes y define las abstracciones fundamentales y sus interfaces. En el framework quedan especificados los aspectos de los tutoriales que se mantienen estables en todos los dominios y se proveen mecanismos para poder expresar las variaciones que sean necesarias. Para poder generar tutoriales en un dominio del conocimiento en particular, es necesario extender al framework con las clases de objetos que representan los datos y acciones del dominio específico. Una vez que se han desarrollado esos componentes, quedan integrados al framework y se convierte en una aplicación lista para generar tutoriales en ese dominio del conocimiento.

Hacia un framework para generación de tutoriales

Comenzamos estudiando y analizando las características y comportamientos de tutoriales (del tipo paso a paso) en diferentes áreas, tomando en cuenta qué comportamientos son compartidos y cuáles son específicos. De esta manera, determinamos los aspectos de los tutoriales que se mantienen estables en todos los dominios y conforman el framework, y definimos además, los mecanismos necesarios para poder expresar las variaciones y características particulares.

Se tuvieron en cuenta tanto los aspectos metodológicos como los pedagógicos subyacentes en las teorías educativas que todo tutorial de calidad educativa debe contemplar.

Entre los aspectos comunes encontramos que en los tutoriales de este tipo, se pretende enseñar mediante un ejemplo, a resolver un problema, mostrando a partir de un conjunto de datos iniciales, la forma de llegar al resultado detallando todos los pasos intermedios necesarios. Lo que se pretende mostrar son las transformaciones necesarias a realizar sobre los datos iniciales, para producir la salida. Cada una de estas transformaciones es un paso del tutorial y habrá que realizarlas en un cierto orden.

Cada paso también tiene sus datos de partida y produce un resultado. Puede verse como un esquema de caja negra, donde las entradas del paso i se encuentran entre las salidas de los pasos anteriores y/o los datos iniciales, en forma directa o mediante operaciones sobre ellos. Como cada uno de ellos representa un problema en particular, debe contemplar la explicación correspondiente.

Otra característica común es la interacción que debe mantener con el usuario para permitirle avanzar, retroceder y reejecutar tantas veces como sea necesario para reforzar su aprendizaje.

Cada tutorial debe poder guardarse, reutilizarse y representarse tanto en forma independiente o como paso intermedio de otro tutorial.

Para poder crear, enseñar o mostrar un tutorial, se requiere de un panel o pizarra en la cual se pueda editar, mover o borrar y de una barra de herramientas que provea los elementos necesarios.

La aplicación debe contemplar dos modalidades de uso bien definidas: creación de un tutorial y uso de un tutorial.

El trabajo de creación de un tutorial mediante la herramienta, debe resultar al docente “tan natural” como si lo desarrollara manualmente, y en general, mucho más sencillo.

En modalidad de ejecución de un tutorial, puede ser practicado individualmente tantas veces como se quiera, en forma completa o parcial, permitiendo volver atrás cuando el alumno lo requiera. Cada paso debe ir acompañado de la explicación correspondiente.

Los aspectos específicos de cada dominio, están dados por las clases de objetos que pueden ser datos y los que pueden ser transformaciones. Estas clases presentan una interfaz común de comportamiento entre los diferentes dominios y las resuelven específicamente, por ejemplo cualquier objeto debe poder representarse en la pizarra, la forma de hacerlo dependerá del objeto mismo.

Teniendo en cuenta todos estos aspectos, diseñamos y desarrollamos el framework al que denominamos TUTGEN proveyéndolo de la inteligencia necesaria para adaptarse a cambios en sus datos de entrada y al reuso de un mismo tutorial en diferente contextos. Finalmente lo especializamos en el dominio de las construcciones geométricas y se está utilizando en la cátedra de Sistemas de Representación del ciclo básico de Ingeniería.

Con esta herramienta, el docente realiza una actividad gráfico-visual con intervención y accionamiento directo sobre la computadora, de la misma manera que la explicación teórico-constructiva se estuviese realizando en el pizarrón, por lo que está reemplazando la pizarra, la tiza, el compás y la regla o escuadra, por elementos gráficos especiales que desarrollan las mismas

actividades y tienen las mismas funciones que los instrumentos descriptos precedentemente. Se facilita la actividad explicativa del docente, al tener la posibilidad de creación de tutoriales a medida y al poder ejecutarlos en cualquier dirección y de manera reiterativa si es así requerido, brindando un complemento ideal para el desarrollo de la actividad áulica, permitiendo disminuir y agilizar los tiempos de dictado sin afectar contenidos. Estos tutoriales creados, van conformando una biblioteca, pueden ser usados como componentes en otro tutorial y pueden ser consultados por los alumnos en cualquier momento, en el aula o fuera de ella, durante el *tiempo de adquisición* o durante el *tiempo de aplicación del conocimiento*.

Descripción del software

TutGen está diseñado para permitir al docente construir tutoriales del tipo “enseñanza paso a paso” a partir de algunos componentes primarios o básicos desarrollados especialmente para el dominio de aplicación. El tutorial así creado, a su vez se convierte en un nuevo componente del dominio y por lo tanto puede ser usado para la creación de otros, debiéndose adaptar en forma inteligente al nuevo contexto.

Permite *al profesor*: crear los tutoriales que apoyarán, guiarán y facilitarán la capacitación del alumno, de ser necesario podrá usar otros tutoriales previamente creados como pasos intermedios del nuevo, usando todas las herramientas que estarán a su disposición (cuadros de texto para ingresar conceptos teóricos, recursos visuales, etc.).

La herramienta “aprende” a dominar los tutoriales creados y permite *al alumno*: interactuar con ellos, ingresando los datos iniciales, variándolos con simplemente “arrastrar” los elementos, avanzar o retroceder cuantas veces quiera, guardar su propio ejemplo con los datos de entrada elegidos, permitiendo así, su entrenamiento para la adquisición de la habilidad pretendida por el profesor, en este caso en particular, en el área de la geometría.

Contempla dos modalidades de uso bien definidas: *creación* de un tutorial y *aprendizaje* de un tutorial.

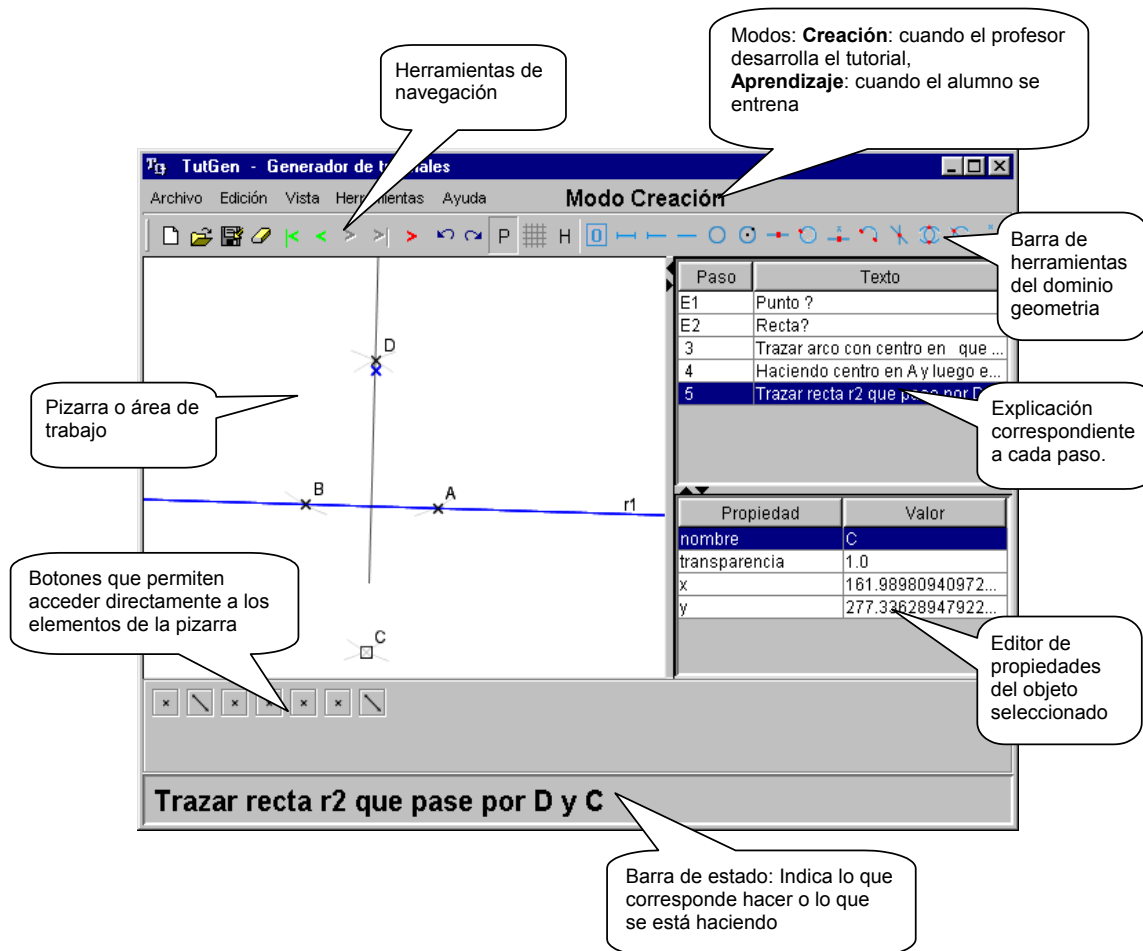
Modo Creación

En este modo el docente cuenta con una barra de herramientas que le permite elegir los datos de entradas y las acciones de transformación para lograr los resultados, una pizarra donde expone los pasos del tutorial, un panel de descripción de cada paso y un editor de las propiedades de los objetos utilizados. La barra de herramientas se adapta automáticamente al dominio para el cual se especializó el framework.

Para crear un tutorial, se debe comenzar indicando los elementos que son datos de entrada del problema (Ejemplo: un punto, una recta). Luego, se indican las sucesivas transformaciones que se necesitan realizar para llegar al resultado esperado (Pasos). Finalmente, se indican cuáles son los elementos que conforman el resultado (Salida). Mediante un botón se cambia de estado, entrada, paso o salida, y la barra de herramientas se actualiza para exponer los elementos correspondiente a cada estado.

A medida que se va agregando una entrada o un paso de transformación, el panel de pasos se va actualizando con la explicación por defecto prevista para ese componente. Esta explicación puede ser cambiada por el docente si así lo desea. Es importante su claridad porque será el texto que irá viendo el alumno durante la ejecución, en modo aprendizaje.

El docente, creador del tutorial, deberá también ingresar el título, el enunciado del problema y la explicación teórica del mismo. Con estos datos, TutGen crea un archivo html que se exhibirá en el panel de explicación en Modo Aprendizaje y que también puede ser consultado fuera de la herramienta, con el navegador.



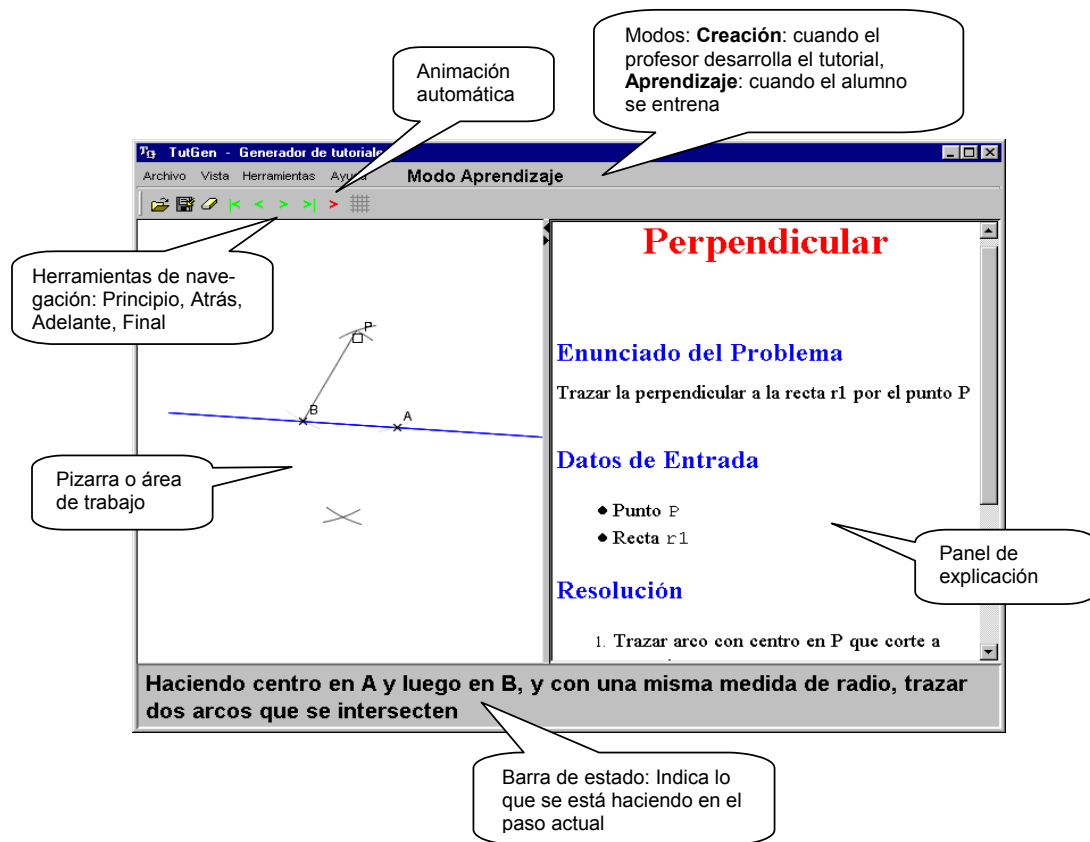
Y para que este tutorial pueda ser utilizado por el alumno para su aprendizaje o por el mismo docente para componer un nuevo tutorial que lo tenga como paso intermedio, el tutorial deberá guardarse. Es el proceso de grabación el que se encargará de almacenar la información necesaria que permita que este tutorial sea utilizado en nuevos contextos en forma inteligente.

Modo Aprendizaje

En modo Aprendizaje, el alumno selecciona el tutorial que desea a través del ítem abrir del menú archivo o del botón equivalente. Una vez realizada esta operación, TutGen solicita los datos y luego va ejecutando el tutorial en cámara lenta, esto es, va mostrando la resolución paso a paso del problema con la rapidez elegida por el alumno, acompañándolo de la explicación correspondiente.

El alumno cuenta con una barra de herramientas que le permiten navegar por el tutorial, avanzar o retroceder todas las veces que lo requiera, ir al principio o al final en un solo paso, correrlo en forma animada y automática. La forma automática recorre sucesivamente los distintos escalones de ejecución: entrada de datos, pasos de ejecución, salida o resultado final, de manera autónoma, desarrollando todo el tutorial completo de principio a fin, en forma animada, cuya velocidad y demora entre paso y paso, puede modificarse en el menú de configuración.

También puede variar los datos y ver cómo es la resolución para ese caso.



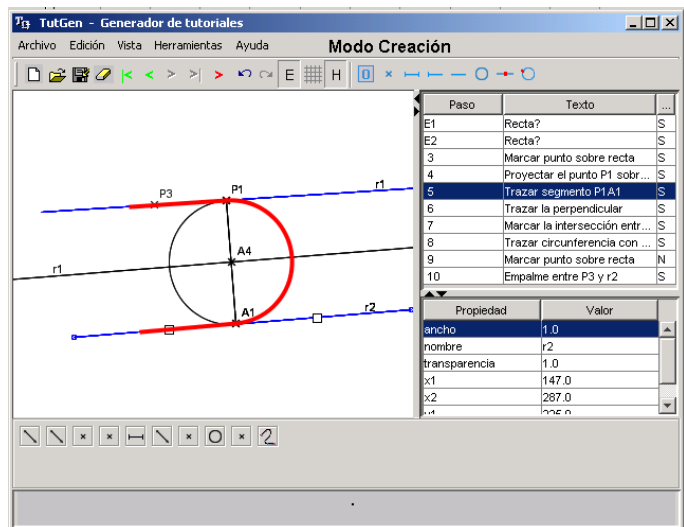
Un ejemplo

El ejemplo explicitado a continuación, corresponde a la creación del tutorial “empalme entre dos rectas paralelas y una circunferencia”.

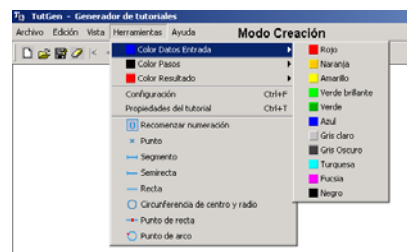
Se puede observar en la pizarra, el tipo de código simbólico utilizado para distinguir entre datos de entradas, pasos y salida, reflejada en colores azul, gris oscuro y rojo respectivamente.

En el Panel de Descripción de Pasos, aparece resaltado lo que es el paso N° 5: Trazar segmento P_1A_1 . Y en el Panel de Propiedades, aparece resaltada una de las características de las propiedades de la recta dato r_2 : el ancho, cuyo valor asignado es igual a 1,0.

Las tonalidades cromáticas pueden ser variadas a gusto del usuario, tanto para la entrada de datos, como para los pasos de ejecución y la salida del resultado, entrando por la barra de menú: Herramientas y direccionando en el sentido optado: Color Datos de Entrada, Color Pasos, Color Resultado.



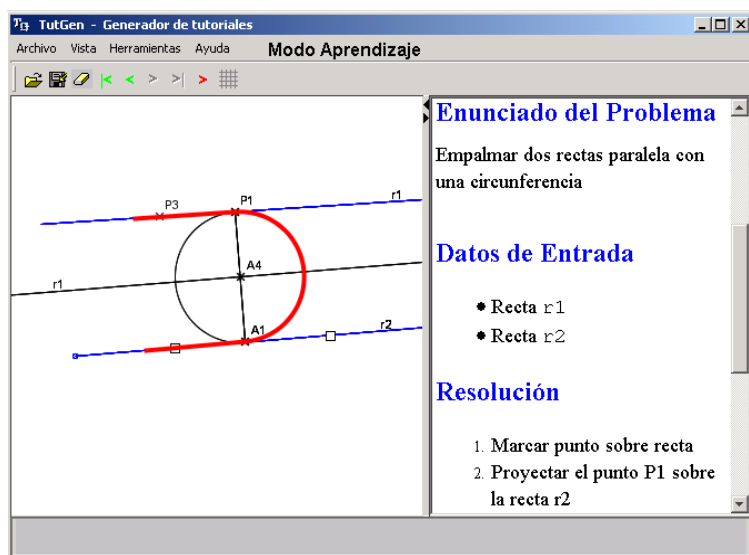
El menú desplegable correspondiente a Herramientas, varía su tamaño dependiendo del punto de ubicación en que nos encontremos en el desarrollo del Tutorial. Hay funciones que son específicas de los Pasos de ejecución del Tutorial, por lo que no es menester que aparezcan ni en la Entrada de Datos ni en la Salida. Estas mismas funciones aparecen como botones en la barra de herramientas del dominio.



Existe la posibilidad de variar el tiempo de ejecución del tutorial como así también la velocidad de animación de cada elemento. Para ello, desde la barra del menú Herramientas, se puede acceder a la Configuración del sistema, en donde se puede modificar la velocidad de animación.

Accediendo al ítem de menú Propiedades del tutorial, se ingresa el título, el enunciado del problema y la explicación teórica del mismo. Con estos datos TUTGEN arma un archivo html de explicación para el alumno.

En la pantalla gráfica siguiente se ve el mismo tutorial en modo aprendizaje, en la etapa final de la animación.



En el panel derecho aparece la explicación paso a paso del problema, armada en forma automática por TutGen con los datos que se le suministró: título, enunciado, explicación teórica y con los elementos que se fueron seleccionando como entradas y los que se fueron seleccionando como pasos de transformación.

El esquema general de ese archivo html de explicación es: Título (no se ve en este caso), explicación conceptual, enunciado del problema, datos de entrada, resolución y resultado.

Conclusiones

Entendemos que la utilización de tutoriales interactivos, es una alternativa apropiada para ser considerada en la planificación de las actividades docentes de las materias de cualquier carrera universitaria.

Con la implementación y uso de este tipo de herramienta, se resuelve en principio, una de las dificultades que se plantea en el aprendizaje tradicional, al facilitar al alumno la visualización de todo un proceso constructivo o desarrollo, y al mismo tiempo, respetar sus tiempos de aprendizaje.

No sólo se potencia la capacidad de asimilación y comprensión de los alumnos, sino que se facilita la actividad explicativa del docente, al tener la posibilidad de creación de tutoriales a medida y al poder ejecutarlos en cualquier dirección y de manera reiterativa si es así requerido, brindando un complemento ideal para el desarrollo de la actividad áulica, permitiendo disminuir y agilizar los tiempos de dictado sin afectar contenidos.

Futuros trabajos

Continuando con esta línea de investigación y desarrollo, nuestro próximo trabajo está orientado a complementar al generador de tutoriales con un entrenador genérico e inteligente, que interprete de cada tutorial los diferentes caminos que conducen a la solución, los reconozca como válidos sin necesidad de que el docente deba expresarlos uno por uno, y permita asistir al alumno en sus errores guiándolo hacia la solución correcta, sin necesidad de la permanente presencia del docente, a la vez que le brinda un mecanismo de autoevaluación.

Bibliografía

ACM Special Interest Group on Computer-Human Interaction Curriculum. ACM SIGCHI: Curricula for Computer-Human Interaction Communications of ACM

Galvis Panqueva A (1994) "Ingeniería de Software Educativo". Ediciones Uniandes.

Gardner H. (1987): "La nueva ciencia de la mente: Historia de la psicología cognitiva". Barcelona. Paidós.

Kristof, Ray –Satran, Amy.(1998) "Diseño interactivo". Ediciones Anaya Multimedia 1998.

Mullet K., Sano D. (1995) "Designing Visual Interfaces: Communication Oriented Techniques", SunSoft Press (Prentice-Hall).

Piaget J.(1989): La construcción de lo real en el niño. Crítica. Grijalbo.

Roberts D., Johnson R. (1997) "Patterns for Evolving Frameworks", PLOP-3 (Pattern Language Oriented Programming).

Rosanigo, Z. B., Paur, A.B., Bramati, P. Metodología de desarrollo de software educativo. Actas de VI Congreso Internacional de Ingeniería Informática ICIEY2K Fac. de Ingeniería, U.B.A. - Buenos Aires – 2000.

Salcedo Lagos P. (2002) Revista Ingeniería Informática, edición 6, <http://www.inf.udec.cl/revista/edicion6/psalcedo.htm>

Skinner B. F., (1958): Teaching Machines, Science, publicado en 1958