Herramientas Didácticas para el Dictado de Informática Teórica

Ing. Maria Alejandra Paz Menvielle
Universidad Tecnológica Nacional / Facultad Regional Córdoba

Ing. Marcelo Marciszack
Universidad Tecnológica Nacional / Facultad Regional Córdoba

Ramiro Perez
Universidad Tecnológica Nacional / Facultad Regional Córdoba

Marina Cardenas
Universidad Tecnológica Nacional / Facultad Regional Córdoba

Abstract

Los contenidos de Informática Teórica en carreras de informática no deberían presentar dificultades en cuanto a su transposición didáctica, ya que constituyen, el núcleo central de estudio de su disciplina. Para que los estudiantes logren un aprendizaje significativo de los conceptos de Informática Teórica, se ha elaborado como estrategia la creación de un conjunto integrado de herramientas didácticas accesibles desde la web, las cuales son utilizadas en los procesos de enseñanza, aprendizaje, posibilitando de esta manera que los estudiantes fijen los contenidos curriculares de teoría de autómatas, gramáticas formales, traducción de lenguajes y compiladores, realizando prácticas efectivas de los mismos, permitiendo además que sea el propio alumno que evalúe el progreso de su aprendizaje mediante el uso de un test de autoevaluación.
Por lo tanto, la estrategia es que el estudiante disponga de un conjunto integrado de herramientas y materiales de estudio, en ambiente web, sentando bases sólidas de conocimientos teóricos, con una gran ejercitación y aplicabilidad de los mismos, permitiendo autoevaluar sus conocimientos teóricos y prácticos a través de simuladores y test de autoevaluación.

Palabras Clave: Informática Teórica, Máquinas Abstractas, Gramáticas Formales, Autómatas, Estrategias didácticas

Introducción

Los contenidos de Informática Teórica, incluidos en el núcleo común de las carreras con informática tales como lenguajes y gramáticas formales, teoría de autómatas, traducción de lenguajes, máquinas computadoras teóricas, y compiladores son difíciles de relacionar por los estudiantes, por su naturaleza matemática y fuertemente abstracta.
Además los estudiantes efectúan prácticas de diseño de máquinas abstractas, sin contar con herramientas para la experimentación en computadora y exploración de contenidos, que les permita en forma clara aplicarlos en la resolución de problemas reales. Esta dificultad manifiesta de relacionar la teoría y sus posibles aplicaciones, se ha advertido durante las evaluaciones de los mismos.
Para poder asimilar en forma completa estos contenidos, consideramos que se deben realizar prácticas efectivas de los mismos utilizando simuladores de máquinas abstractas, estudiando su funcionamiento y los algoritmos de las herramientas respectivas. Es necesario además que el educando pueda explorar estos conocimientos y además que, mediante un simulador de evaluaciones de contenidos teóricos (el que incluye la generación de un

instrumento de evaluación automatizado que basado en un repositorio de preguntas y respuestas en ambiente web), le sea posible generar en forma aleatoria un test de autoevaluación. La importancia de una completa asimilación de estos contenidos, radica en que los mismos no sólo tienen efectiva aplicabilidad en la construcción de compiladores, sino en temas tan diversos dentro de la disciplina informática como son la ingeniería de requerimientos o programación de dispositivos móviles y embebidos.

La dificultad manifiesta de relacionar la teoría y sus posibles aplicaciones, se ha advertido durante el dictado de los contenidos y se pone en evidencia en el proceso de evaluación.

Existe una gran cantidad de bibliografía y herramientas de software para el estudio de estos temas, pero las mismas los abordan en forma parcializada, sin integración adecuada, con diferentes nomenclaturas y con definiciones dispares, por lo que no resultan didácticamente útiles para llevar adelante el proceso de enseñanza-aprendizaje, sin permitir la correcta transposición de los contenidos teóricos en aplicaciones de la realidad.

En este proyecto, se busca además, lograr profundo conocimiento teórico y práctico, de las técnicas y herramientas conceptuales de la Informática Teórica relacionadas con la construcción de compiladores, e involucrar a alumnos en el estudio y la investigación de temas fundamentales de su carrera.

Los aspectos teóricos, para el abordaje de la temática planteada, están muy difundidos y son bastante generalizados: sobre los contenidos de informática Teórica, es completo lo publicado en [1] [2] [3] [4] [5], pero los mismos no se ajustan en lo didáctico, para ser transferido en forma directa a los alumnos en una carrera de grado en informática.

Siguiendo los lineamientos de un proyecto anterior a éste, tendrá como misión fundamental continuar lo realizado en la fase previa a través de la revisión, reformulación y ampliación el conjunto de herramientas desarrolladas, valiéndonos de los resultados obtenidos y de los cambios introducidos en la Cátedra, para mejorarlas y así propender a lograr en los alumnos aprendizaje significativo. Se utilizarán técnicas y metodologías actuales que posibiliten a los miembros del proyectos incrementar sus habilidades en diseño, desarrollo y pruebas de los productos construidos, basándose el la bibliografía mencionada en [6] [7] [8].

En resumen, en este trabajo se propone construir un conjunto de herramientas disponibles para ejecutar desde la web “simuladores” que le permitan al alumno realizar una práctica efectiva, utilizando estos simuladores de máquinas abstractas, estudiando y comprendiendo el funcionamiento de los programas fuente de los mismos, y en todo momento poder autoevaluar sus aprendizajes.

Elementos del Trabajo y metodología

Nos basamos en el Modelado UML, y la programación extrema (XP) como metodología de desarrollo, con amplia aplicación del modelo incremental; en la construcción de los simuladores adoptamos un lenguaje de programación de última generación, Java, utilizado por la mayoría de los estudiantes de las carreras de informática.

Para abordar los contenidos de manera de facilitar el aprendizaje de los alumnos, se define la arquitectura y el modelo orientado a objetos vinculado a los simuladores; como también la construcción del sitio Web donde se encuentran disponibles los simuladores para que los alumnos realicen las prácticas; conjuntamente con material teórico,

práctico y de evaluación sobre los contenidos de la asignatura.

Para la construcción de los simuladores, se han desarrollado herramientas didácticas para poder ejemplificar y visualizar los conceptos teóricos involucrados en informática teórica. Estas herramientas web, permiten ver los resultados gráficamente en un applet, expresado como XML, o bien de manera web.

Los simuladores de gramáticas permiten definirlas mediante los alfabetos de símbolos terminales y no terminales, el axioma y el conjunto de producciones; los usuarios podrán cambiar los componentes que define la gramática, y visualizarlas en un grafo, en un XML o en notación formal.

El simulador permite identificar el tipo de gramática, identifica símbolos superfluos terminales y no terminales, símbolos inaccesibles, permitiendo limpiar la gramática o indicar si se encuentra en una Forma Normal. En los casos de gramáticas regulares y dependientes del contexto, se generará el autómata correspondiente.

En los simuladores para autómatas nos enfocamos en los autómatas finitos y los autómatas a pila, que son los de más relevancia en la etapa de análisis dentro de un compilador.

Estos simuladores permiten explorar, pudiendo ingresar y modificar los componentes que los definen, los alfabetos (de entrada, de salida, y de pila) el conjunto de estados, el conjunto de estados finales, el símbolo inicial, y el conjunto de transiciones.

En ambos casos procuramos poder reconocer si una palabra pertenece a un lenguaje generado por una gramática y si puede ser aceptado por un autómata.

También podremos pasar un autómata a una gramática, de acuerdo a su tipo, y también realizar el proceso inverso; de ésta manera se entiende el isomorfismo entre las gramáticas y los autómatas.

Al poder los alumnos visualizar una aplicación práctica de los contenidos, y poder simular la ejecución de autómatas finitos, con pila, gramáticas formales y su isomorfismo, de manera que ellos mismos evalúen las soluciones a los problemas planteados en las clases prácticas, es de esperar que los conocimientos y habilidades adquiridas, sean realmente significativas y eleven su desempeño en el cursado de Informática Teórica

Resultados

Para la realización de este trabajo se utilizó el método científico, con la aplicación de un desarrollo experimental, el cuál comenzó con la recolección del material sobre este temática que se encuentra publicado, continuando con la selección y clasificación del mismo.

Se procedió a la unificación de la nomenclatura y simbología utilizada.

Se estableció la estrategia didáctica para abordar los contenidos de manera de facilitar el aprendizaje de los alumnos incluyendo la estructura del sitio web se incluyen todos los contenidos teóricos, ejercitación práctica y las herramientas de simulación para que los alumnos realicen las prácticas, y la evaluación de sus aprendizajes.

Detallaremos a continuación las características esenciales de los productos construidos:

<u>Sitio Web</u>

A través de él se puede acceder en forma integrada a todos los contenidos teóricos, conjuntamente con la correspondiente ejercitación práctica con ejercicios resueltos y a resolver por los alumnos. Conjuntamente

con los links para acceder a los simuladores y a los tests de evaluación.

Figura 1: Pantalla de acceso Sitio Web

El sitio web puede ser accedido desde la dirección www.institucional.frc.utn.edu.ar/sistemas/ghd

Herramientas de Simulación

Para la ejercitación práctica de los contenidos, se desarrollaron las siguientes herramientas:

- Generadores de gramáticas formales con su correspondiente ejercitación práctica.
- Simuladores de autómatas finitos, con pila, linealmente acotados y máquina de Turing, para que los alumnos evalúen las soluciones a los problemas planteados en las clases prácticas.
- Generador de analizadores léxicos a partir de una gramática regular que muestre la generación automática de código desde una especificación formal y las distintas formas de tratar con el no determinismo del autómata finito obtenido: conversión a autómata finito determinista, búsqueda en árboles generados por estados posibles y procesamiento en paralelo de los distintos caminos deterministas generados por el no determinismo del autómata.

La siguiente es una pantalla que muestra uno de los simuladores (Autómatas Finitos), con los que los alumnos pueden realizar simulaciones y ejercitar.

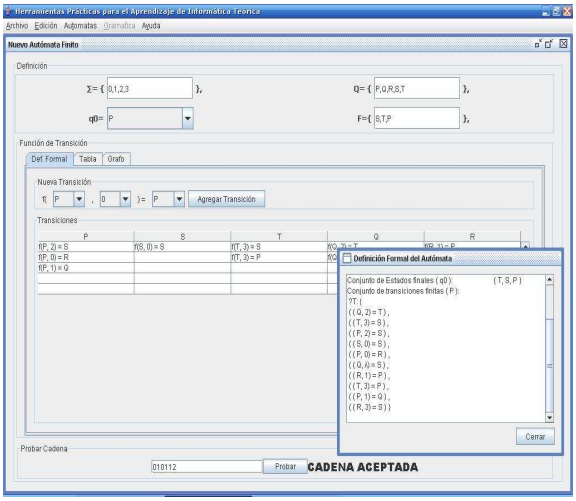

Figura 2: Simulador de Autómatas Finitos

Test de Autoevaluación

Si bien existen un gran cantidad de herramientas para confeccionar cuestionarios de evaluación de aprendizajes (incluidas en plataformas para e-learning, incluso gran cantidad de ellas de software libre), se construyó una herramienta de manera de poder integrarla al sitio y que cumpliera con funcionalidades adicionales como el de poder generar en forma aleatoria el instrumento de evaluación con las siguientes particularidades: definir cantidad de preguntas por cada unidad temática y sub-temas a incluir en el cuestionario de evaluación, selección aleatoria de los diferentes enunciados para una misma pregunta, selección aleatoria de opciones correctas e incorrectas dentro de un conjunto de opciones dentro de cada pregunta, cantidad de opciones correctas que puede tener una pregunta y permitir valorizaciones diferenciadas para cada pregunta.

Esto permite entre otras cosas la imposibilidad de que un instrumento resulte igual a otro ya practicado por el alumno.

A continuación se muestran algunas pantallas generadas por la herramienta de autoevaluación.

Para la Generación del Instrumento

En esta pantalla, el Docente, luego de haber ingresado el conjunto de preguntas, con sus diferentes enunciados y opciones de respuestas correctas e incorrectas, puede configurar diversos parámetros, tales como: selección de contenidos de las unidades temáticas y dentro de estas los sub-temas a incluir en el cuestionario en forma conjunta con la cantidad de preguntas, cantidad de opciones por preguntas, y opciones de valorización de las mismas.

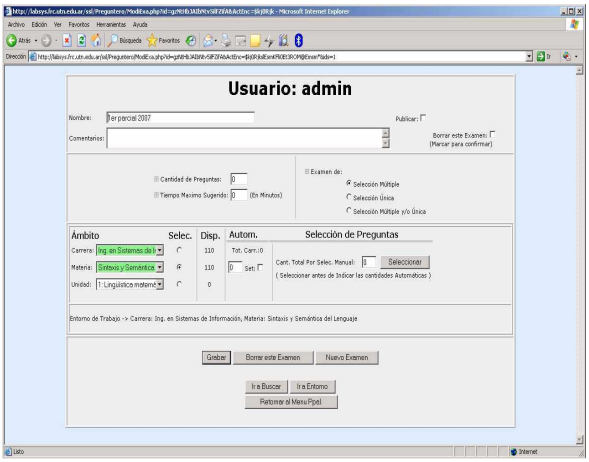

Figura 3: Pantalla de configuración

A continuación se muestra parcialmente la visualización de un cuestionario generado por la herramienta de autoevaluación en donde se evidencia la simplicidad para responder las preguntas por parte del alumno.

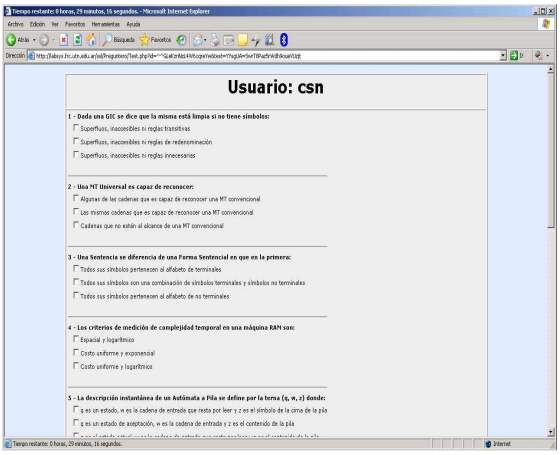

Figura 4: Instrumento generado

Implementación

Estas herramientas, aunque aún en proceso de desarrollo y continuo agregado de nuevas funcionalidades, durante el ciclo 2008 han sido puestas en práctica sobre un conjunto de alumnos, mas precisamente en la comisión 2K4 en el dictado de la asignatura Sintaxis y Semántica del Lenguaje de la carrera de grado de Ingeniería en Sistemas de Información en la Facultad Regional Córdoba de la Universidad Tecnológica Nacional, estando disponibles desde un primer momento los contenidos teóricos, la ejercitación práctica, las herramientas construidas, y el test de autoevaluación. El sitio web, alojado en la intranet de la Facultad Regional Córdoba, a modo de prueba piloto para este conjunto de alumnos, se hará efectivo para todas las comisiones en ciclos lectivos posteriores.

Si bien, no ha estado completamente implementado al inicio del ciclo lectivo, ya que durante el desarrollo del dictado de la asignatura se han introducido cambios y nuevas funcionalidades, se evidencia que el resultado obtenido ha sido satisfactorio, desde las siguientes dimensiones de análisis: la primera, que resulta subjetiva ya que no existen parámetros que de medición que la justifiquen, es que se ha reflejado con respecto a ciclos anteriores que los alumnos se encuentran mas motivados al poder verificar con los simuladores sus ejercitaciones, y la

segunda que si es verificable, ya que aplicar la misma metodología de evaluación, durante el mismo período y sobre idénticos contenidos, al efectuar una comparación de los rendimientos académicos con ciclos lectivos anteriores, el promedio de las calificaciones obtenidas por los estudiantes se ha visto incrementado en algo superior a un punto en una escala de diez.

Discusión

El presente proyecto se encuentra consolidado dentro de la línea de investigación que comenzó con la fase I desarrollado en el Dpto. de Sistemas de Información de la UTN FRC, el cual tuvo como objetivo la construcción del sitio web en su etapa inicial y el desarrollo de los primeros simuladores. En esta fase II se está perfeccionando el sitio y se están desarrollando mayores funcionalidades y nuevos simuladores.

Conclusión

Las herramientas desarrolladas junto con los contenidos teóricos y prácticos de la asignatura están puestos a disposición de los alumnos en un sitio web, se han comenzado a utilizar en forma satisfactoria ya que los alumnos han evidenciado aplicación práctica de los contenidos teóricos, han podido verificar los resultados de la ejecución de los prácticos, y han podido autoevaluar sus aprendizajes previamente a la instancia de evaluación.

Las pruebas realizadas nos permitieron detectar correcciones a realizar de manera de introducir los cambios necesarios para facilitar la comprensión por parte de los alumnos. En esta segunda fase se introducirán todas las modificaciones observadas en las pruebas y se completarán las herramientas de simulación con el agregado de pantallas de gráficos que permitan en forma visual observar la ejecución de las mismas.

Para completar la utilización de este conjunto de herramientas, y como futuro trabajo, se realizará una medición adicional, en donde se aplicará los test de evaluaciones como instrumento predictivo, permitiendo detectar tópicos de dificultad de manera de efectuar correcciones en el dictado y producir de esta manera una mejora en el proceso de enseñanza-aprendizaje.

Agradecimientos

A la Agencia Córdoba Ciencia y a las autoridades de la Facultad Regional Córdoba de la Universidad tecnológica Nacional, el subsidio y apoyo para la realización del proyecto.

Referencias

[1] J. E. Hopcroft / Rajeev Motwani / J.D. Ullman (2008), Teoría de Autómatas, Lenguajes y Computación, PEARSON Addison-Wesley

[2] Kenneth Louden (2004), Lenguajes de Programación, Thomson Internacional

[3] Pascual J. Iranzo (2005), Lógica Simbólica para Informáticos, Alfaomega Group Editor

[4] John C. Martin (2004), Lenguajes Formales y Teoría de la Computación, McGraw-Hill

[5] P. Isasi / P. Martínez / D. Borrajo (1997), Lenguajes, Gramáticas y Autómatas, Addison-Wesley

[6] Borsotti Carlos (2006), Temas de Metodología de la investigación, Ed MNO Y DAVILA

[7] Pressman Roger (2005), Ingenieria del software: un enfoque práctico, McGraw Hill

[8] Norma ISO 9001:2000, Calidad en el desarrollo de software.

Datos de Contacto

Ing. Maria Alejandra Paz Menvielle. Universidad Tecnológica Nacional Facultad Regional Córdoba. pazmenvielle@yahoo.com.ar

Ing. Marcelo Marciszack Universidad Tecnológica Nacional Facultad Regional Córdoba. marciszack@gmail.com

Ramiro Perez. Universidad Tecnológica Nacional Facultad Regional Córdoba. ramipez@gmail.com

Marina Cardenas. Universidad Tecnológica Nacional Facultad Regional Córdoba. angelaesmeralda@gmail.com