

Jornadas Académicas en Facultad de Ingeniería de la ORT

Por Virginia Matos

La Facultad de Ingeniería de la Universidad ORT organizó el 17 de setiembre las 11 Jornadas Estudiantiles, 4^{tas} Jornadas Académicas y 3^{eras} Jornadas Universidad - Empresa. Durante estas jornadas, estudiantes de las diferentes carreras de la facultad presentaron trabajos y proyectos, y tuvo lugar un intenso intercambio académico entre investigadores y profesionales que trabajan en temas relacionados con la informática, la electrónica y las telecomunicaciones. En este marco, la facultad llevó a cabo el 8^{vo} Concurso de Programación, cuyos equipos ganadores representaron a la Universidad ORT en la Final Sudamericana del "ACM International Collegiate Programming Contest", organizado por ACM (Association for Computing Machinery), que tuvo lugar en Buenos Aires el 15 de noviembre.

La Facultad de Ingeniería y Vincular, la Oficina de Vinculación Universidad Empresa de la Universidad ORT, invitaron a las principales selectoras de personal, asociaciones y organizaciones vinculadas al sector TIC, a participar de estas Jornadas. Desde hace tres años esta es una instancia de vinculación entre el sector académico y la industria, que brinda la posibilidad a los estudiantes y graduados de tomar contacto con las empresas, presentar su perfil y acceder a oportunidades laborales, contribuyendo al fortalecimiento del vínculo entre la universidad y la empresa, establecido desde diversas acciones en busca de mayor innovación y oportunidades. Participaron las organizaciones: Advice, Arcanus, ARTech, Asociación de Jóvenes Emprendedores, Asociación de Ingenieros del Uruguay, Cámara Uruguaya de Tecnologías de la Información, Dovat, Carriquiry y Asociados, Escuela Militar de Aeronáutica, IEEE, Infocorp, Isbel, Manpower, McGraw Hill, Microsoft, Pearson, Quanam, TATA Consultancy Services y Uy Lug.

Lo que sigue es un breve informe acerca de algunos de los trabajos presentados en las Jornadas Estudiantiles de la Facultad de Ingeniería (JEFI).

Videjuegos con software libre

Martín Sagastume y Ruben Darío Duarte, estudiantes del segundo año de Licenciatura en Sistemas e Ingeniería en Sistemas respectivamente, presentaron una introducción al tema del desarrollo de videjuegos de tres dimensiones (3D) en Java, utilizando software libre, o sea software a cuya conformación se puede acceder, modificar, mejorar y distribuir libremente sin pagar una licencia.



Al visitar el GameLab de la Universidad ORT, donde vieron que había material con muchas posibilidades para desarrollar videjuegos, como el motor de juego jMonkey, decidieron investigar el campo y luego presentar los resultados de su estudio.

Un motor de juego o *Game Engine* consiste en una serie de rutinas de programación que permiten el diseño, la creación y la representación de un videjuego. La analogía con el motor de un automóvil es ilustrativa: el motor debajo del capot no es visible pero le da una funcionalidad al automóvil que es la de transportar. En un juego las texturas y los modelos 3D serían la carrocería, pintura y exteriores, y del mismo modo que los automóviles no se mueven sin un motor, el arte y los guiones del juego no funcionan sin un motor de juego.

El motor jMonkey que utilizaron los estudiantes en esta presentación es una novedad de bastante reciente



aparición que recién ahora se está ganando un lugar en el mercado. En la charla mostraron videos acerca de la evolución de los videojuegos y también videos de juegos que actualmente están siendo desarrollados por empresas utilizando el motor jMonkey.

Altimetro digital

Gonzalo Díaz y Nicolás Cremona son estudiantes de la carrera Ingeniería en Electrónica. Motivados por la carrera de piloto que está cursando Gonzalo, que les permitió conocer y experimentar con los instrumentos que actualmente portan los aviones ligeros de nuestro país, y habiéndose familiarizado con los microcontroladores a través de la asignatura curricular "Microcontroladores y Microprocesadores", desarrollaron un altímetro digital como trabajo curricular obligatorio de la asignatura "Instrumentación y Medidas".

El microcontrolador es un circuito integrado capaz de ser programado por un usuario para que realice las funciones que este desee. En este caso, los estudiantes lo programaron para que mida el valor de presión atmosférica, valor que -mediante una operación matemática- es transformado en un valor de altura que luego es enviado hacia una pantalla LCD.

Al igual que los altímetros convencionales empleados actualmente, el diseñado por Díaz y Cremona basa su funcionamiento en la medición de presión para determinar la altura del avión. Pero mientras que los altímetros convencionales, analógicos, actualmente usados en la mayoría de los aviones ligeros de nuestro país, transforman el valor medido de presión en movimiento (la deflexión de una aguja en el panel visor del instrumento), el altímetro digital mide la presión con un sensor más sofisticado y transforma el valor medido en un valor de voltaje eléctrico que muestra en la pantalla LCD. Esto permite mejorar la exactitud en la determinación de la altura, ofreciendo una mejor visualización de la información y facilitando al piloto las calibraciones necesarias previas a cada vuelo.

Sin ocultar su orgullo por el logro alcanzado, los estudiantes señalaron que, mientras el error máximo de medición de

altura admitido por la aviación civil en Uruguay es de 60 pies, el altímetro diseñado por ellos tuvo un error final de 12 pies. El funcionamiento del aparato fue probado en un vuelo, resultando todo un éxito.

Generador de aplicaciones para dispositivos móviles

Ignacio Galiazi, Virginia Jamaro y María Eugenia Queirolo, estudiantes de Ingeniería en Sistemas presentaron el resultado de su proyecto final de la carrera. Este es un generador de aplicaciones para dispositivos móviles, celulares, iphones, ipods, u otros que tengan conexión a Internet. El producto desarrollado permite, por ejemplo, que los médicos cuenten con una aplicación en sus celulares que les permita ingresar al sistema del hospital donde trabajan y ver la historia clínica de sus pacientes desde cualquier lugar.

En el caso de los vendedores de una empresa que distribuye productos, estos deben recorrer los locales de los clientes para levantar los pedidos. Si dispusieran de la aplicación mencionada podrían hacerlo a través de un dispositivo móvil, sin necesidad de realizar esa recorrida. También permite realizar transacciones. De esta forma la empresa se expande, no queda confinada en sus límites físicos sino que cualquier integrante de la misma que esté fuera de la empresa puede estar interactuando con la misma. Si los funcionarios quieren enviarse mensajes de texto entre sí, pueden hacerlo desde el sitio de la empresa al que están conectados, seleccionando al funcionario al cual quieren enviarle el mensaje.

Lo novedoso del proyecto es que los estudiantes no desarrollaron la aplicación para los médicos o para los vendedores sino que diseñaron un generador para cualquier tipo de aplicación tanto para las actividades de una farmacia, un hospital o un banco, por ejemplo. A partir de una base de datos existente y del generador, un analista de sistemas arma la aplicación adecuada a cada necesidad particular, la que en sólo dos horas puede estar funcionando en los dispositivos móviles.





Por un tema de tiempos los autores del trabajo aún no han podido lanzar el producto al mercado, pero la idea es hacerlo en un futuro cercano.

Asistente para Deducción Natural

Bruno Bianchi cursa el cuarto año de Ingeniería en Sistemas. El proyecto que presentó en las JEFI fue el resultado de un trabajo de investigación desarrollado durante el primer semestre de este año. Se trata de una herramienta de software ideada por la cátedra de Teoría de la Computación, para brindar apoyo didáctico e interactivo al dictado del curso curricular de Lógica que se imparte en el segundo semestre de la carrera. Puntualmente, el sistema llamado 'Asistente DN', busca asistir a los alumnos en el aprendizaje de la Deducción Natural (DN).

El sistema de Deducción Natural busca "arritmetizar" el razonamiento, es decir, definir reglas de cálculo cuya aplicación permita derivar nuevo conocimiento (conclusión) a partir de conocimiento verdadero (premisas iniciales) y garantice la preservación de la verdad de dichas premisas a lo largo del proceso.

Una demostración de la validez de cierta afirmación, obtenida a partir de otras, escrita en este sistema es una 'demostración formal' en la que en cada paso se aplican las reglas definidas en el sistema. Estas demostraciones realizadas por una computadora son extremadamente largas y su lectura es complicada y tediosa para los seres humanos. Cuando los matemáticos se comunican entre sí realizan 'demostraciones informales', en las que utilizan argumentos suficientemente rigurosos como para convencer a otros matemáticos de que están en lo cierto. Aunque en realidad son demostraciones muy rigurosas, no van paso a paso como en las formales, permitiendo que se siga la idea central de los argumentos más fácilmente.

Una ventaja de las demostraciones formales es que tienen

una forma mecánica de verificar que todos los pasos son correctos, dando una seguridad del 100% de que el argumento es correcto.

Un programa de DN ayuda al usuario que quiere demostrar un teorema, y que tiene en mente una demostración informal, a escribir la demostración formal. Como esto es muy tedioso y es muy fácil cometer errores, es importante la ayuda del programa. Este colabora de dos maneras: la primera, se asegura de que todos los pasos que uno toma en la demostración son correctos; la segunda, busca automáticamente demostraciones formales para ciertos pasos de la prueba. En general, buscar demostraciones formales puede ser muy difícil hasta para una computadora, pero en ciertos casos, las computadoras pueden encontrarlas más rápidamente que los seres humanos. O sea que, mientras el usuario escribe la demostración formal en el programa, en algunos momentos puede pedirle al programa que escriba otras partes.

Ya existen programas que hacen esto. La idea se usa cada vez más en aplicaciones reales, para demostrar que un programa determinado hace exactamente lo que uno quiere. Por ejemplo, si el programa toma decisiones sobre cómo volar un avión, o si toma decisiones de cuándo es que un marcapaso tiene que hacer una cosa u otra, uno quiere saber que funciona en el 100% de los casos, ya que un 99.99% no es suficiente.

Los profesores de la cátedra querían contar con un sistema que tuviera la funcionalidad de Asistente DN, para utilizarlo como herramienta de apoyo didáctico. La tarea de Bianchi consistió en identificar las funcionalidades del sistema a desarrollar y, después de investigar productos similares desarrollados por otras universidades, materializar todos estos requerimientos en un producto de *software* concreto, para lo cual también fue preciso investigar ciertas tecnologías utilizadas en la interfaz gráfica, por ejemplo. En comparación con aplicaciones similares para uso profesional, este sistema reúne una gran variedad de recursos didácticos y una interfaz amena e intuitiva.