

Sistema web para la enseñanza de algoritmos y programación usando Jigsaw

LEIBNITZ ROJAS

Univesidad Nacional Mayor de San Marcos

leiparov@gmail.com

Resumen

Los estudios muestran que en muchas universidades del mundo, aún existen problemas cuando se trata de enseñar cursos relacionados a programación y algoritmos. Muchos estudiantes repiten las materias y otros simplemente abandonan en mitad de semestre. Por otro lado, existen muchas investigaciones respecto a cómo mejorar los problemas de aprendizaje de los estudiantes y no necesariamente en temas de programación. Muchos autores han aplicado diversas técnicas de aprendizaje colaborativo obteniendo resultados notables en sus alumnos. El objetivo del presente trabajo es desarrollar un sistema web para la enseñanza de algoritmos y programación a través de una técnica de aprendizaje colaborativo, el mismo que permitirá a los estudiantes desarrollar ejercicios y problemas de forma colaborativa.

I. INTRODUCCIÓN

A pesar de que la programación es el corazón de las ciencias de la computación, y por ende, la mayoría de las carreras de computación tienen cursos de programación, los resultados son desalentadores pues existen muchos estudios multi institucionales que indican que hay serias deficiencias en el aprendizaje de alumnos que han pasado uno o más cursos de programación (McCracken y cols., 2001; Lister y cols., 2004; Tenenberg y cols., 2005). Algunas instituciones han logrado mejorar los cursos de programación adoptando el Python como primer lenguaje de programación. Así lo indica Nikula, Sajaniemi, Tedre, y Wray (2007).

Según Knobelsdorf, Kreitz, y Böhne (2014), los altos ratios de fracasos en los cursos de introducción a la teoría de las ciencias de la computación son un problema comun en las universidades de Alemania, Europa, y Norte-América, pues los alumnos tiene dificultades con lo contenidos que por naturaleza son abstractos y teóricos. (Knobelsdorf y cols., 2014) plantean en su investigación ciertas modificaciones a la pedagogía de un curso dictado

en la Universidad de Postdam, Alemania, las mismas que fueron motivadas por un enfoque de aprendizaje cognitivo.

Existen diversas técnicas para desarrollar el aprendizaje colaborativo en un aula de clase y una de ellas, muy conocida, es la técnica de Jigsaw o técnica de Rompecabezas. Esta técnica fue creada en (1978) por Aronson et al. y actualmente es una de las más importantes para fomentar la cooperación y discusión entre miembros de una comunidad de aprendizaje y es usada frecuentemente en ambientes face-to-face y en situaciones de aprendizaje en línea (Blocher, 2005).

Hoy en día, muchos estudiantes tienen dificultades para llevar con éxito los cursos de algoritmos y programación, problema que se evidencia en el porcentaje de alumnos que desaprueban los exámenes, que desaprueban el curso o que simplemente se retiran a mitad de ciclo.

El problema descrito en el párrafo anterior se justifica con el hecho de que la alta tasa de fracaso de los estudiantes de programación ha

sido durante muchos años un tema polémico para las instituciones de aprendizaje con reportes de ratios de fracasos alrededor de 26 % a 40 %. (Sheard y Hagan, 1998; Truong, Bancroft, y Roe, 2003; Lang, McKay, y Lewis, 2007; Han y Beheshti, 2010).

Además, la últimas investigaciones sobre el problema reflejan que éste aún persiste. Los altos porcentajes de fracasos en cursos introductorios de programación son un problema común en universidades en Alemania, Europa, y Norte América, ya que los alumnos tiene problemas para entender los contenidos de tales cursos debido a su abstracción y naturaleza teórica (Knobelsdorf y cols., 2014).

En lo que respecta al uso de técnicas de aprendizaje colaborativo tenemos que Martínez y Camacho (2011) presentaron el diseño, implementación y evaluación de una estrategia de enseñanza basada en aprendizaje colaborativo para introducir el tema de álgebra relacional en un curso de base de datos. La estrategia fue evaluada desde las perspectiva del alumno y del profesor, y se encontró que entre el 78 % y el 92 % de los estudiantes consideraron que el trabajo en grupo enriqueció su aprendizaje, dando soporte al uso del aprendizaje colaborativo; y recientemente, Cliburn (2014) desarrolló el curso de Estructura de Datos a través del aprendizaje basado en equipos y el aprendizaje tradicional con el fin de comparar resultados en las evaluaciones de los estudiantes, y, aunque no encontró diferencias significativas entre ambas secciones de alumnos, aún continúa usando el aprendizaje en equipos debido a la alta satisfacción que los alumnos muestran en comparación con el método de enseñanza tradicional.

II. ESTADO DEL ARTE

LearnCS

LearnCS! es un entorno de programación creado específicamente para el uso de estu-

diantes de primer año de la carrera de ciencias de la computación. Este programa elimina la necesidad de los alumnos de tener que preocuparse por un editor de texto, comandos linux y proceso de compilación. *LearnCS!* provee un entorno web en el cual los alumno pueden escribir, ejecutar y depurar programas usando un interfaz familiar y amigable. El compilador de C embebido que posee el sistema permite al alumno ejecutar sus programas con simplemente hacer click en un botón y obtener los resultados de ejecución de su código fuente (Lipman, 2014).

LearnCS! fue creado para proporcionar un ambiente de aprendizaje para estudiantes de informática de primer año. Sus principales objetivos son proporcionar asistencia útil al alumno en la construcción de un modelo mental de la máquina nociónal de C a través de la visualización detallada de la memoria de *LearnCS!* y sus mensajes de error integrados en las instalaciones de depuración, y para proporcionar que producen consejos que son útiles para el principiante para localizar y corregir errores de sintaxis (Lipman, 2014).

Jigsaw

Jigsaw es una técnica de aprendizaje colaborativo y recientemente ha sido aplicada por Buhr, Heflin, White, y Pinheiro (2014) a un grupo de estudiantes de medicina de la Universidad de Duke tal y como se describe en el artículo *"Using the Jigsaw Cooperative Learning Method to Teach Medical Students About Long Term and Postacute Care"*. En este estudio se desarrolló una experiencia colaborativa usando la técnica de Jigsaw para exponer a los estudiantes sobre el cuidado intensivo a largo plazo (LTPAC - Long Term and Post Acute Care) y así lograr que ellos conozcan las herramientas y roles del personal involucrado en este tipo de cuidados de pacientes en un asilo de ancianos. Para alcanzar este objetivo, pequeños grupos de estudiantes de medicina realizaron entrevistas al personal

de LTPAC sobre sus respectivos roles. Estos grupos posteriormente fueron reorganizados en nuevos grupos conteniendo un estudiante de cada grupo original más un moderador y cada estudiante en los nuevos grupos tuvo que explicar sobre el rol del profesional LTPAC al cual había entrevistado (Buhr y cols., 2014).

Ideone API - Sphere Engine

Sphere Engine, antes conocida como Ideone API, permite a los usuarios ejecutar código en múltiples lenguajes de manera online. Ideone es un compilador y una herramienta de depuración que soporta más de 60 lenguajes de programación. A través de la API, los desarrolladores pueden crear sus propias aplicaciones con fines educativos, personales o de negocios. Sphere Engine es un servicio web el cual puede ser accedido a través de protocolo SOAP.

Esta API posee las siguientes funcionalidades:

1. Permite subir código fuente y compartirlo con otros usuarios.
2. Permite ejecutar el código fuente con una data inicial en el lado servidor y en más de 60 lenguajes de programación diferentes.
3. Permite descargar los resultados obtenidos en la ejecución del código fuente (salida, errores, información de compilación, tiempo de ejecución, uso de memoria, etc).

III. APOORTE PRÁCTICO

Sistema Jigsaw Coding

El Sistema Jigsaw Coding tiene como característica principal el permitir a los usuarios de perfil alumno poder resolver problemas de programación que el docente asigne para la sesión jigsaw. Este requisito está descrito en el caso de uso **Resolver problema** que se presenta en la Figura 1.

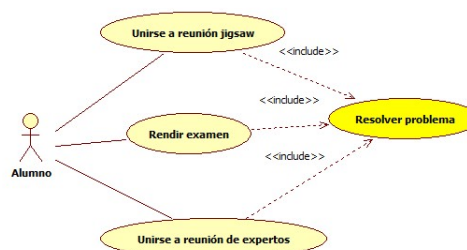


Figura 1: Caso de uso Resolver problema

Como se observa en la Figura 1, el caso de uso Resolver problema se encuentra como parte de los casos de uso Unirse a reunión de expertos, Unirse a reunión jigsaw y Rendir examen, y esto es porque, en cada una de las fases de la sesión jigsaw que se elabora en el sistema, se asignan problemas de algoritmos y programación que los alumnos deben desarrollar ya sea de manera grupal, o individual cuando se trata de un examen. El desarrollo de estos problemas se realiza en un editor de código fuente y luego se compila para obtener los resultados de la solución elaborada.

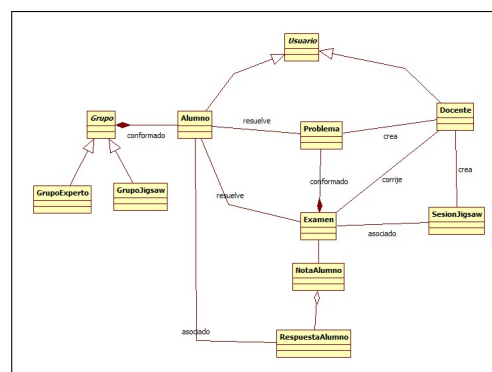


Figura 2: Diagrama de Clases

Por otro lado, tal como se muestra en el diagrama de clases de la Figura 2, el Sistema Jigsaw Coding permite el acceso a Usuario lo cuales pueden ser de dos tipos: Docente o Alumno. El docente puede crear problemas y además es el encargado de crear las sesiones jigsaw. El docente también es quien crea los exámenes, los cuales están conformados por un conjunto de problemas y cada examen

es resuelto por los alumnos, quienes para tal objetivo, deben resolver los problemas que componen cada examen. Por otro lado, los alumnos también resuelven problemas cada vez que se encuentran dentro de un grupo de expertos o un grupo jigsaw. Naturalmente, los exámenes son evaluados por el docente y éste les asigna una nota a cada una de las respuestas que los alumnos envía al terminar su examen.

La vista de desarrollo muestra el sistema desde la perspectiva del programador y se ocupa de la gestión del software a implementar. Esto es, en esta vista se describe cómo estará dividido el sistema Jigsaw Coding en paquetes y las dependencias que hay entre ellos. Ver [Figura 3](#).

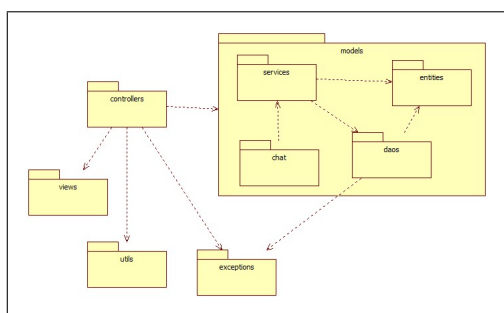


Figura 3: Diagrama de Paquetes

■ controllers

Este paquete contiene todas las clases Controladores que sirven para gestionar el ruteo de páginas del sistema.

■ models

En este paquete se encuentran las clases de Servicios, Entidades y Acceso a Datos que son requeridas para el sistema.

■ views

Este paquete contiene todas las plantillas (*.scala.html) que permitirán renderizar las páginas web del sistema.

Para la evaluación de la calidad del sistema se presentaron métricas de calidad basadas en la norma ISO-9126, la cual establece un estándar internacional para la evaluación de la

calidad de los productos software (ISO/IEC, 2003).

1. Entendibilidad: ¿Qué porcentaje de los usuarios consideran que las funcionalidades del sistema son fáciles o muy fáciles de entender?
2. Portabilidad: ¿En cuántos navegadores web puede usarse el sistema sin tener problemas?
3. Eficiencia: ¿Cuál es el promedio de tiempo de respuesta del Sistema Jigsaw Coding?

IV. CASO DE ESTUDIO

El caso de estudio para la aplicación del Sistema Jigsaw Coding consta principalmente de estudiantes en cuyas carreras llevan o han llevado cursos relacionados a Programación y Algoritmos.

En esta ocasión, se tomará en cuenta a seis estudiantes a quienes se les hará partícipes de una sesión de clase jigsaw a través del sistema desarrollado en esta tesis.

En este caso de estudio se formarán 3 grupos expertos los cuales desarrollarán problemas de estructuras selectivas y repetitivas (if - for - while). De esa forma, cuando se llegue a la reunión jigsaw, los grupos jigsaw contarán con expertos en problemas sobre los 3 tipos de estructuras.

Para este caso de estudio se plante la resolución de los siguientes problemas de estructuras selectivas y estructuras repetitivas los cuales podrán ser desarrollados usando cualquiera de los 3 lenguajes que el Sistema Jigsaw Coding permite usar (Java, C++ o Python).

Para la fase de reunión de expertos y reunión jigsaw se usarán los siguientes problemas:

1. **Par o impar.** Elaborar un programa que lea un entero y determine si es un número par o impar.
2. **Múltiplos de 7.** Elaborar un programa que imprima los primeros 10 números múltiplos de 7 usando la estructura repetitiva `for`.
3. **Números descendentes.** Elaborar un programa que imprima de forma descendente los primeros 20 números pares usando una estructura repetitiva `while`.

Finalmente, para la fase de evaluación se combinará problemas de estructuras selectivas y repetitivas:

1. **Número mayor.** Elaborar un programa que lea dos números enteros y muestre el número mayor.
2. **n múltiplos de a.** Elaborar un programa que muestre los n primeros múltiplos de a, donde n y a son ingresados por teclado.
3. **De número a texto.** Elaborar un programa que lea un número del 1 al 5 y lo muestre de forma escrita. Ejemplo 1 → uno. Si es un número distinto a 1,2,3,4 o 5 que muestre el mensaje: Número equivocado.

4. **Asteriscos.** Elaborar un programa que imprima la siguiente serie de asteriscos:

```
*
**
***
****
*****
```

V. RESULTADOS

Durante la fase de expertos se plantearon los problemas *Múltiplos de 7*, *Par o impar*, *Números descendentes* y a continuación se presentan las soluciones de los alumnos.

GE_DÁT_ECM → Múltiplos de 7.

Elaborar un programa que imprima los primeros 10 números múltiplos de 7 usando la estructura repetitiva `for`.

```
#include <iostream>
using namespace std;
int main () {
    int i;
    cout<<"Los 10 primeros mutiplos de 7 son:"
    <<endl;
    for(i=1;i<=10;i++){
        cout<<i*7<<endl;
    }
}
```

GE_KMM_DEG → Par o impar.

Elaborar un programa que lea un entero y determine si es un número par o impar.

```
import java.util.Scanner;

class Ejercicio01 {
    public static void main (String args[] ){
        Scanner sc = new Scanner(System.in);

        System.out.print("Introduzca un numero entero:");
        int n = sc.nextInt();

        if(n%2==0) System.out.print("El numero es par");
        else System.out.print("El numero es impar");
    }
}
```

GE_SMM_MRB → Números descendentes.

Elaborar un programa que imprima de forma descendente los primeros 20 números pares usando una estructura repetitiva `while`.

```
#include <iostream>
using namespace std;
int main () {
    int var = 40;
    while(var > 1){
        var = var - 2;
        cout<<var<<endl;
    }
    return 0;
}
```

Cuando la fase de expertos finalizó, cada alumno fue reagrupado para dar inicio a la reunión *jigsaw* y en ella cada grupo resolvió los 3 problemas: *Múltiplos de 7*, *Par o impar*,

Números descendentes.

Por otro parte, luego de aplicar las métricas definidas para el sistema, se obtuvieron los siguientes resultados:

- El Sistema es muy fácil de entender para el 66 % de los usuarios del caso de estudio.
- El Sistema es compatible y funciona correctamente en 4 navegadores web: Chrome, Internet Explorer, Ópera y Safari.
- El Sistema posee un tiempo promedio de respuesta de 1.3 segundos para las funcionalidades más importantes.

VI. CONCLUSIONES

- Se desarrolló los antecedentes del problema y de la técnica. Así mismo, se planteó la justificación de y se definieron cuáles serían los objetivos y alcances de este trabajo de investigación.
- Se detalló algunos conceptos teóricos sobre aprendizaje colaborativo y cuáles son sus beneficios al aplicarlo en los estudiantes.
- Se plasmó el Estado del arte de las algunas herramientas y técnicas que permiten el aprendizaje colaborativo.
- Se describió todo el desarrollo del Sistema Jigsaw Coding, para el cual se usó las mejores prácticas de RUP. Además, también se detalló el uso del sistema y se definieron 3 métricas de calidad que posteriormente serían aplicadas al sistema.
- Se presentó el caso de estudio sobre el cual se aplicaría el Sistema Jigsaw Coding. Se contó con la participación de 6 estudiantes universitarios y la sesión jigsaw comprendió temas de estructuras selectivas y repetitivas.
- Se aplicaron al Sistema Jigsaw Coding las métricas de calidad definidas y se presentaron los resultados obtenidos.

REFERENCIAS

- Blocher, J. M. (2005). Increasing learner interaction: Using jigsaw online. *Educational Media International*, 42(3), 269–278. Descargado 2014-04-17, de <http://www.editlib.org/p/97880/>
- Buhr, G. T., Heflin, M. T., White, H. K., y Pinheiro, S. O. (2014). Using the jigsaw cooperative learning method to teach medical students about long-term and postacute care. *Journal of the American Medical Directors Association*, 15(6), 429 - 434. Descargado de <http://www.sciencedirect.com/science/article/pii/S1525861014000905>
- Cliburn, D. C. (2014, mayo). Team-based learning in a data structures course. *J. Comput. Sci. Coll.*, 29(5), 194–201. Descargado 2014-05-02, de <http://dl.acm.org/citation.cfm?id=2600623.2600662>
- Han, J., y Beheshti, M. (2010, abril). Enhancement of computer science introductory courses with mentored pair programming. *J. Comput. Sci. Coll.*, 25(4), 149–155. Descargado 2014-05-07, de <http://dl.acm.org/citation.cfm?id=1734797.1734826>
- ISO/IEC. (2003). *ISO/IEC TR 9126-3: Software engineering - product quality - part 3 : internal metrics*. ISO/IEC.
- Knobelsdorf, M., Kreitz, C., y Böhne, S. (2014). Teaching theoretical computer science using a cognitive apprenticeship approach. En *Proceedings of the 45th ACM technical symposium on computer science education* (p. 67–72). New York, NY, USA: ACM. Descargado 2014-05-07, de <http://doi.acm.org/10.1145/2538862.2538944>
- Lang, C., McKay, J., y Lewis, S. (2007). Seven factors that influence ICT student achievement. En *Proceedings of the 12th annual SIGCSE conference on innovation and technology in computer science education* (p. 221–225). New York, NY, USA: ACM. Descargado 2014-05-07, de <http://doi.acm.org/10.1145/1268784.1268849>

- Lipman, D. (2014, junio). LearnCS!: a new, browser-based c programming environment for CS1. *J. Comput. Sci. Coll.*, 29(6), 144–150. Descargado 2014-05-02, de <http://dl.acm.org/citation.cfm?id=2602724.2602752>
- Lister, R., Adams, E. S., Fitzgerald, S., Fone, W., Hamer, J., Lindholm, M., ... Thomas, L. (2004). A multi-national study of reading and tracing skills in novice programmers. En *Working group reports from ITiCSE on innovation and technology in computer science education* (p. 119–150). New York, NY, USA: ACM. Descargado 2014-05-03, de <http://doi.acm.org/10.1145/1044550.1041673>
- Martinez, A., y Camacho, A. (2011). A cooperative learning-based strategy for teaching relational algebra. En *Proceedings of the 16th annual joint conference on innovation and technology in computer science education* (p. 263–267). New York, NY, USA: ACM. Descargado 2014-05-01, de <http://doi.acm.org/10.1145/1999747.1999821>
- McCracken, M., Almstrum, V., Diaz, D., Guzdial, M., Hagan, D., Kolikant, Y. B.-D., ... Wilusz, T. (2001). A multi-national, multi-institutional study of assessment of programming skills of first-year CS students. En *Working group reports from ITiCSE on innovation and technology in computer science education* (p. 125–180). New York, NY, USA: ACM. Descargado 2014-05-03, de <http://doi.acm.org/10.1145/572133.572137>
- Nikula, U., Sajaniemi, J., Tedre, M., y Wray, S. (2007). Python and roles of variables in introductory programming: Experiences from three educational institutions. *JITE*, 6, 199–214. Descargado 2014-05-07, de <http://www.jite.org/documents/Vol6/JITEv6p199-214Nikula269.pdf>
- Sheard, J., y Hagan, D. (1998). Our failing students: A study of a repeat group. En *Proceedings of the 6th annual conference on the teaching of computing and the 3rd annual conference on integrating technology into computer science education: Changing the delivery of computer science education* (p. 223–227). New York, NY, USA: ACM. Descargado 2014-05-07, de <http://doi.acm.org/10.1145/282991.283550>
- Tenenberg, J., Fincher, S., Blaha, K., Bouvier, D., yi Chen, T., Chinn, D., ... Vandegrift, T. (2005). Students designing software: a multi-national, multi-institutional study. *Informatics in Education*.
- Truong, N., Bancroft, P., y Roe, P. (2003). A web based environment for learning to program. En *Proceedings of the 26th australasian computer science conference - volume 16* (p. 255–264). Darlinghurst, Australia, Australia: Australian Computer Society, Inc. Descargado 2014-05-07, de <http://dl.acm.org/citation.cfm?id=783106.783135>