



FACULTAD DE INGENIERÍA DE SISTEMAS E INFORMÁTICA

ESCUELA DE INGENIERÍA DE SOFTWARE

Leibnitz Pavel Rojas Bustamante

SISTEMA WEB DE TIEMPO REAL PARA EL APRENDIZAJE COLABORATIVO

Tesis de Ingeniería

Lima, 9 de octubre de 2014

Leibnitz Pavel Rojas Bustamante

SISTEMA WEB DE TIEMPO REAL PARA EL APRENDIZAJE COLABORATIVO

“Tesis presentada a la Universidad Nacional Mayor de San Marcos, Lima, Perú, para obtener el Título de Ingeniero de Software”.

Asesora: Mg. Lenis Wong Portillo

UNMSM - LIMA
9 de octubre de 2014

© Leibnitz Rojas, 2014.
Todos los derechos reservados.

Este trabajo esta dedicado a mis
padres Arturo y María.

Agradecimientos

A la profesora Lenis Wong por su orientación, consejos y revisiones del presente trabajo.

A mis colegas y amigos de la escuela de Ingeniería de Software por sus observaciones y porque en todo momento me incentivaron para que culmine este trabajo.

A los profesores de la UNMSM, principalmente a la profesora Lenis Wong por sus observaciones teóricas que me sirvieron de mucho.

A todas aquellas personas que indirectamente me ayudaron para culminar este trabajo y que muchas veces constituyen un invalorable apoyo.

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.

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, el mismo que permitirá a los estudiantes desarrollar temas a través de técnicas de aprendizaje colaborativo como la técnica de Jigsaw y la técnica de Investigación en pares.

Abstract

Studies show that in many universities around the world, there are still problems when it comes to teaching courses related to programming and algorithms. Many students repeat the courses and others just leave in the middle of the semester.

There are many studies on how to improve the learning problems of students and not necessarily on programming topics. Many authors have applied various techniques of collaborative learning obtaining remarkable results in their students.

The aim of this work is to develop a web system for teaching programming and algorithms, the same will enable students to develop issues through collaborative learning techniques like Jigsaw and Pair Research.

Índice general

Índice general	IX
Índice de figuras	X
Índice de tablas	XI
1. Introducción	1
1.1. Antecedentes	1
1.1.1. Antecedentes del problema	1
1.1.2. Antecedentes de la técnica	2
1.2. Definición del Problema	4
1.3. Justificación	4
1.3.1. Justificación práctica	4
1.3.2. Justificación teórica	5
1.4. Objetivos	5
1.4.1. General	5
1.4.2. Específicos	5
1.5. Alcances	5
1.6. Estructura de la Tesis	6
2. Marco Teórico	7
2.1. Aprendizaje Colaborativo	7
2.1.1. Definición	7
2.1.2. Elementos en el aprendizaje colaborativo	8
2.1.3. Beneficios del aprendizaje colaborativo	8
2.2. Técnicas de Aprendizaje Colaborativo	9
2.2.1. La técnica de Jigsaw	9
2.2.2. Programación en pares	11
2.2.3. El estudio de Casos	12
2.3. Tecnologías web	12
2.3.1. AJAX	12

2.3.2.	Polling	14
2.3.3.	HTTP Long-Polling	15
2.3.4.	WebSockets	15
2.4.	Glosario	16
3.	Estado del Arte	18
3.1.	Herramientas para el aprendizaje colaborativo	18
3.1.1.	LearnCS	18
3.1.2.	Google Docs	20
3.1.3.	Sistema Web para la enseñanza de Casos de Uso em- pleando la Técnica de Aprendizaje Cooperativo de Rom- pecabezas.	22
3.1.4.	CodeBunk	24
3.1.5.	Análisis comparativo	25
3.2.	Técnicas para el aprendizaje colaborativo	27
3.2.1.	Jigsaw	27
3.2.2.	Pair Research	29
3.2.3.	Análisis comparativo	31
3.3.	Frameworks para aplicaciones web de tiempo real	32
3.3.1.	GoogleDrive Realtime API	32
3.3.2.	Ideone API - Sphere Engine	33
3.3.3.	Análisis comparativo	34
4.	Aporte práctico	35
4.1.	Las mejores prácticas de RUP	36
4.2.	Implementación de un sistema web de tiempo real para el aprendizaje colaborativo	38
4.3.	38
A.	Casos de Uso	39
A.1.	Introducción	39
A.1.1.	Propósito	39
A.1.2.	Alcance	39
A.1.3.	Definiciones, acrónimos y abreviaciones	39
A.2.	Catálogo de actores	40
A.3.	Diagramas de casos de uso	41
A.3.1.	Casos de uso - Docente	41
A.3.2.	Casos de uso - Alumno	42
A.4.	Especificaciones de Casos de Uso	43
A.4.1.	Registrar Alumno	43
A.4.2.	Crear Problema	44

A.4.3. Crear grupo experto	45
A.4.4. Asignar alumnos a grupo experto	46
A.4.5. Crear sesión jigsaw	47
A.4.6. Modificar sesión jigsaw	48
A.4.7. Crear evaluación	49
A.4.8. Atender consulta de alumno	50
A.4.9. Participar en reunión de expertos	51
A.4.10. Resolver problema	52
A.4.11. Consultar al grupo	53
A.4.12. Participar en reunión jigsaw	54
A.4.13. Consultar al docente	55
A.4.14. Rendir evaluación	56
B. Especificaciones suplementarias	57
B.1. Introducción	57
B.2. Funcionalidad	57
B.3. Fiabilidad o confiabilidad	57
B.4. Usabilidad	58
B.5. Eficiencia o Performance	58
B.6. Mantenibilidad	58
B.7. Portabilidad	58
C. Documento de Arquitectura de Software	59
C.1. Introducción	59
C.1.1. Propósito	59
C.1.2. Alcance	59
C.2. Vista lógica	61
C.3. Vista de procesos	62
C.4. Vista de desarrollo	63
C.5. Vista física	64
C.6. Vista de escenarios	64
Referencias	65

Índice de figuras

2.1. Grupos originales en la técnica Jigsaw	10
2.2. Grupos de expertos	11
2.3. Regreso a los originales	11
2.4. Tecnologías agrupadas bajo el concepto de AJAX	13
2.5. Polling	14
2.6. Long Polling	15
2.7. Websockets	16
3.1. LearnCS	19
3.2. LearnCS	20
3.3. Google Docs	21
3.4. CodeBunk	24
3.5. Jigsaw Paso 1	27
3.6. Jigsaw Paso 2	28
3.7. Pair Research	30
3.8. Ideone	33
4.1. RUP	36
4.2. CRONOGRAMA	38
A.1. Diagrama de actores	40
A.2. Casos de uso - Docente	41
A.3. Casos de uso - Alumno	42
C.1. Modelo de arquitectura 4+1	60
C.2. Diagrama de Paquetes	63
C.3. Diagrama de Despliegue	64

Índice de tablas

3.1. Herramientas para el aprendizaje colaborativo	25
3.2. Técnicas de aprendizaje colaborativo	31
3.3. Frameworks para aplicaciones web de tiempo real	34
4.1. Artefactos del proceso de desarrollo	37
A.1. Actores	40

Capítulo 1

Introducción

1.1. Antecedentes

1.1.1. Antecedentes del problema

La idea del aprendizaje colaborativo(AC) empezó a ser de interés para los profesores de colegios americanos allá por el año 1980, pero la primera idea básica fue desarrollada en los años 1950 a 1960 por un grupo de profesores e investigadores británicos ([Bruffee, 1984](#)). Después de estudiar la interacción entre estudiantes de medicina y su profesores de física, M.L.J Abercrombie concluyó que los estudiantes de medicina que aprendieron a realizar diagnósticos como un grupo alcanzaron un buen juicio médico, más rápido que aquellos que trabajaron individualmente. Bruffee además plantea que su primer encuentro con la creencia de AC fue cuando se encontró con las conclusiones de un grupo de investigadores que pensaban que el AC se deriva de un ataque contra los estilos de enseñanza autoritarios.

El aprendizaje basado en proyectos colaborativos con equipos distribuidos está siendo revolucionado por los rápidos avances tecnológicos que existen hoy en día. Tanto profesores, alumnos e información para las clases deben ser reunidas en un entorno virtual para reducir las barreras geográficas y temporales de cada uno de los miembros de cada equipo de aprendizaje ([Wang, 2002](#)). Wei Wang, en el 2002 propuso en su tesis Computer-Supported Virtual Collaborative Learning and Assessment Framework for Distributed Learning Environment un marco de trabajo para el aprendizaje cooperativo de equipos distribuidos y con ello diseñó e implementó un Sistema de Soporte para la Enseñanza y Aprendizaje Colaborativo (CLASS por sus siglas en inglés: Collaborative Learning Assessment Support System).

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; Tenenbergs 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).

Así mismo, en el año 2010, en la Universidad Pinar del Río se vió la necesidad de elaborar una herramienta de software que sirviera de apoyo para la enseñanza del curso de Inteligencia Artificial y así, Salao Bravo, J. R en su tesis “Estudio de las técnicas de Inteligencia Artificial mediante el apoyo de un software Educativo” implementó un sistema web con el objetivo de potenciar el estudio de las técnicas, funcionamiento y aplicaciones de la Inteligencia Artificial (Salao Bravo, 2010).

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 NorteAmérica, pues los alumnos tiene dificultades con lo contenidos que por naturaleza son abstractos y teóricos. (Knobelsdorf y cols., 2014) plantea 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.

1.1.2. Antecedentes de la técnica

Según (Laal & Laal, 2012), el aprendizaje colaborativo es un enfoque educacional de enseñanza y aprendizaje que involucra grupos de estudiantes trabajando juntos para resolver un problema, completar una tarea, o crear un producto y también significa aprender a través del trabajo en conjunto en lugar de aprender por uno mismo (Barkley, Cross, & Major, 2012).

El aprendizaje cooperativo o aprendizaje colaborativo es una técnica de enseñanza muy conocida y que se ha aplicado con una gran variedad de materias y un amplio espectro de las poblaciones (L. L. Beck & Chizhik, 2008).

Azizinezhad, Hashemi, y Darvishi (2013) realizaron un estudio para investigar los efectos del aprendizaje colaborativo en el aprendizaje del idioma

inglés como lengua extranjera para los alumnos. En dicho estudio se concluyó que los alumnos fueron capaces de mostrar mejores y significativas competencias lingüísticas, competencia discursiva, competencias estratégicas y competencias de comunicación no verbal que el resto de alumnos. En un entorno de aprendizaje cooperativo, hubo muchas tareas interactivas, que de forma natural, estimularon las habilidades sociales, lingüísticas y cognitivas de los estudiantes. Las actividades cooperativas tendían a integrar la adquisición de aquellas habilidades, y crear potentes oportunidades de aprendizaje.

Existen diversas técnicas para desarrollar el aprendizaje colaborativo en un aula de clase y una de ellas, muy conocida, es la técnica de Jigsaw. Esta técnica fue creada en (1978) por Aronson et al. y actualmente es una de las más importantes técnicas 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). De acuerdo con (Aronson, Blaney, Stephin, Sikes, & Snapp, 1978), usualmente en un Jigsaw el contenido se divide en 5 a 6 subtemas y a cada alumno se le asigna la tarea de estudiar a detalle su respectivo subtema. Los alumnos repasan en grupo el subtema para convertirse en “expertos”. Al final de esta fase, los grupos de expertos se dispersan y se forman nuevos grupos llamados “grupos jigsaw o grupos rompecabezas”. Dentro del nuevo grupo, a cada alumno se le pide que informe sobre su subtema a los demás, y así, al final, todos los grupos obtienen una visión completa de los contenidos.

Según los creadores de la técnica Jigsaw (Aronson y cols., 1978) ,ésta es particularmente apropiada cuando el tópico de estudio es fácil de fragmentar en sub tópicos, y/o en aquellos contextos donde es particularmente importante trabajar sobre la responsabilidad individual. Sin embargo, cuando se diseña un Jigsaw Online, hay aspectos críticos que se deben tomar seriamente en consideración: el tamaño de la población objetivo, las restricciones de tiempo y la necesidad de un sistema de comunicación bien estructurado (Persico D., 2008).

La técnica de Jigsaw ha sido usada en los procesos educacionales en países de todos los continentes y puede mejorar el rendimiento de los alumnos y estudiantes a través del aprendizaje colaborativo (Maftei & Maftei, 2011). Así mismo, Kilic (2008) sostiene que el aprendizaje colaborativo es el proceso de aprendizaje de aquellos que no conocen mucho sobre un tema trabajando en conjunto con aquellos que sí lo conocen, y esto es un concepto que continuamente atrae a muchos docentes; Según Kilic (2008), el aprendizaje colaborativo es un proceso que se enfoca en desarrollar a los estudiantes

social e intelectualmente. Además, varias investigaciones han mostrado que especialmente en primaria, secundaria y universidad, la técnica Jigsaw es efectiva en el proceso de aprendizaje de cursos teóricos, en el desarrollo de pensamiento crítico de los estudiantes y en sus habilidades de comunicación.

La técnica de rompecabezas o técnica de Jigsaw, fue implementada en un sistema web en el año 2013 en la Universidad Pontificia Católica del Perú con el fin de automatizar los procesos que se requiere aplicar dicha técnica al aprendizaje colaborativo. A través de ese sistema los alumnos pudieron aprender conceptos sobre Casos de Uso de una manera diferente a una clase tradicional (Pinzás & Yatsen, 2013).

1.2. Definición del Problema

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.

1.3. Justificación

1.3.1. Justificación práctica

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 & Hagan, 1998; Truong, Bancroft, & Roe, 2003; Lang, McKay, & Lewis, 2007; Han & Beheshti, 2010).

Las 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 tienen problemas para entender los contenidos de tales cursos debido a su abstracción y naturaleza teórica (Knobelsdorf y cols., 2014). Además, este problema también aún es vigente en universidades del Perú y sin ir muy lejos, también está presente en la Facultad de Ingeniería de Sistemas e Informática de la Universidad Nacional Mayor de San Marcos.

1.3.2. Justificación teórica

Martinez y Camacho (2011) presentaron el diseño, implementación y evaluación de una estrategia de enseñanza basada en aprendizaje cooperativo 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.

1.4. Objetivos

1.4.1. General

Desarrollar un sistema web de tiempo real para promover el aprendizaje colaborativo de los estudiantes a través de la técnica de Jigsaw y enfocándolo específicamente a la enseñanza de cursos de algoritmos y programación.

1.4.2. Específicos

- Investigar sobre diferentes técnicas y herramientas informáticas para el aprendizaje colaborativo.
- Investigar sobre frameworks que sirven para realizar aplicativos web de tiempo real.
- Definir 3 métricas de calidad para el desarrollo del sistema.
- Aplicar el sistema web a la enseñanza de algoritmos y programación en la Facultad de Ingeniería de Sistemas e Informática en la UNMSM.

1.5. Alcances

La presente tesis tendrá los siguientes alcances:

- Se investigará sobre las diferentes técnicas y herramientas informáticas usadas para el aprendizaje colaborativo.
- Se investigará sobre los diversos frameworks para aplicaciones web de tiempo real.
- Se evaluará el sistema a desarrollar en base a las métricas definidas.
- Se implementará el sistema en la FISI.

1.6. Estructura de la Tesis

La presente tesis está organizada en X capítulos que a continuación se explican brevemente.

En el Capítulo 2 se describe el marco teórico, donde se explican los conceptos fundamentales sobre Aprendizaje cooperativo, los mismo que son abordados a lo largo de toda esta investigación. Así mismo, también se detallan los conceptos principales para entender el funcionamiento de sistemas web de tiempo real.

En el Capítulo 3 se describe el estado del arte, donde se describen y analizan algunas de las técnicas existentes para el desarrollo del aprendizaje colaborativo, así mismo se presentan herramientas para el aprendizaje colaborativo y finalmente se describen algunos frameworks que permiten realizar aplicaciones web de tiempo real que ayudarán en la implementación del sistema propuesto en esta tesis.

Capítulo 2

Marco Teórico

En el presente capítulo se define con mayor detalle los conceptos relevantes que abarca esta tesis. El marco teórico incluye una breve descripción de lo que significa Aprendizaje colaborativo presentando además algunos de sus elementos y beneficios. Así mismo, también se describe algunas de las técnicas más conocidas para el aprendizaje colaborativo. Finalmente, se describe algunos conceptos sobre tecnologías para aplicaciones web, conceptos que serán de ayuda para la implementación de la presente tesis.

2.1. Aprendizaje Colaborativo

2.1.1. Definición

Existen diversas formas de definir lo que es Aprendizaje Colaborativo; Macgregor (1990) dice que la enseñanza y el aprendizaje colaborativo son un enfoque educacional que involucra grupos de estudiantes trabajando juntos para resolver un problema, completar una tarea o crear un producto y Gerlach (1994) sostiene que el aprendizaje cooperativo está basado en la idea de que el aprendizaje es un acto social natural en el cual los participantes conversan entre sí mismos y que es a través de la comunicación y la charla donde realmente ocurre el aprendizaje.

El aprendizaje colaborativo es un término para describir una variedad de enfoques educacionales que implican reunir el esfuerzo intelectual de los estudiantes, o estudiantes y profesores juntos. Usualmente los estudiantes están trabajando en grupos de dos o más, buscando entender, solucionar problemas o crear productos. Las actividades de aprendizaje cooperativo son variadas, pero la mayoría se centran en la exploración del estudiante o la

aplicación de los materiales de curso, no simplemente en la presentación de un tema por parte del profesor (Smith & MacGregor, 1992); además, el aprendizaje colaborativo tiene como principal característica una estructura que permite a los estudiantes comunicarse entre sí, y es ahí donde ocurre el aprendizaje (Golub y cols., 1988).

2.1.2. Elementos en el aprendizaje colaborativo

(Johnson, Johnson, & Holubec, 1984) plantea 5 elementos básicos en el aprendizaje colaborativo. El aprendizaje colaborativo no es simplemente para los estudiantes el hecho de trabajar en grupo y de acuerdo con su investigación, un ejercicio de aprendizaje sólo califica como colaborativo si están presentes los siguientes elementos:

- *La interdependencia positiva.* Los miembros del equipo están obligados a confiar en los demás para alcanzar un objetivo. Si uno de los miembros del equipo falla al realizar su parte, todos sufren las consecuencias. Los miembros del equipo necesitan creer que están unidos con los demás de una forma que aseguren el éxito en conjunto.
- *La interacción “cara a cara” o simultánea.* Los miembros del equipo se tienen que ayudar y alentar entre sí para aprender. Ellos deben de explicar qué entendieron y así compartir su conocimiento.
- *La responsabilidad individual.* Todos los estudiantes de un grupo son responsables de hacer su parte del trabajo.
- *Habilidades sociales.* Los estudiantes deben ser alentados y ayudados a desarrollar y practicar la confianza de equipo, liderazgo, toma de decisiones, comunicación, y manejo de conflictos.
- *Autoevaluación de grupo.* Los miembros del equipo tienen que fijarse objetivos, revisar periódicamente qué están haciendo bien como equipo, e identificar cambios por hacer con el fin de mejorar la efectividad a futuro.

2.1.3. Beneficios del aprendizaje colaborativo

Numerosos beneficios han sido descritos para el aprendizaje cooperativo (Panitz, 1999). Una buena forma de organizarlos es colocándolos en categorías. Johnson y Johnson (1989); Panitz (1999) hicieron una lista de más de 50 beneficios para el aprendizaje cooperativo, algunos de los cuales se presentan a continuación:

1. Beneficios sociales

- a) Ayuda a desarrollar un sistema de apoyo social para los estudiantes.
- b) Lleva a construir un entendimiento de la diversidad entre los estudiantes y el personal.
- c) Establece un entorno positivo para modelar y practicar la cooperación y el trabajo en equipo.
- d) Desarrolla comunidades de aprendizaje.

2. Beneficios psicológicos

- a) La instrucción centrada en los estudiantes aumenta la autoestima de los mismos.
- b) La cooperación reduce la ansiedad.
- c) El aprendizaje cooperativo desarrolla actitudes positivas hacia los profesores.

3. Beneficios académicos

- a) El aprendizaje cooperativo promueve habilidades de pensamiento crítico.
- b) Envuelve a los estudiantes activamente en el proceso de aprendizaje.
- c) Los resultados de clase son mejorados.
- d) El aprendizaje cooperativo modela técnicas apropiadas para la resolución de problemas.
- e) Grandes conferencias pueden ser personalizadas.
- f) El aprendizaje es especialmente útil para motivar a los estudiantes en un plan de estudios específico.

2.2. Técnicas de Aprendizaje Colaborativo

2.2.1. La técnica de Jigsaw

La técnica de Jigsaw, que fue introducida por [Aronson y cols. \(1978\)](#) para mejorar la cooperación en pares y crear solidaridad en equipo entre los estudiantes a través de la división de tareas, involucra a cada estudiante en un grupo a asumir responsabilidades en el aprendizaje. En consecuencia,

los estudiantes trabajan en dos diferentes grupos: el grupo de expertos y el grupo jigsaw.

Los objetivos de esta técnica son:

- Estructurar las interacciones entre los alumnos, mediante equipos de trabajo.
- Lograr que los alumnos dependan unos de otros para lograr sus objetivos.

La secuencia de pasos que conforman esta técnica son los siguientes ¹:

1. El docente debe tener preparada la división del tema a tratar en cinco o seis documentos, los cuales se repartirán a los alumnos siguiendo un orden. Cada uno de ellos será necesario para aprender la totalidad del tema, y por lo tanto, todos ellos formarán la unidad temática completa.
2. Se divide a los alumnos en grupos de cinco o seis (según el número de documentos elaborados) y dentro de cada grupo cada miembro recibirá un número de 1 a 5 (ó 6). Ver figura 2.1

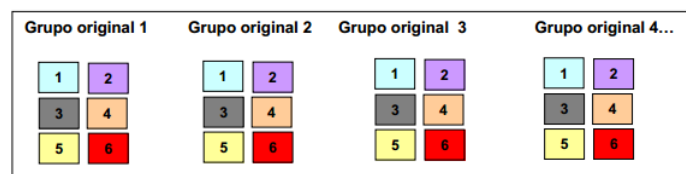


Figura 2.1: Grupos originales en la técnica Jigsaw

A los estudiantes con el número 1 se les reparte el mismo documento, que será diferente al resto de los compañeros y que puede corresponderse a la primera parte del tema de estudio. A los alumnos con el número 2 se les reparte otro documento y así sucesivamente.

La primera fase será, por tanto, que los alumnos preparen su documento de forma individual, que lo lean, que lo entienda, que lo aprendan y que recopilen las dudas que surjan.

3. Una vez que ya ha finalizado el tiempo estimado para la preparación individual del documento, comienza la segunda fase que se denomina "Reunión de expertos". En este momento todos los alumnos con el

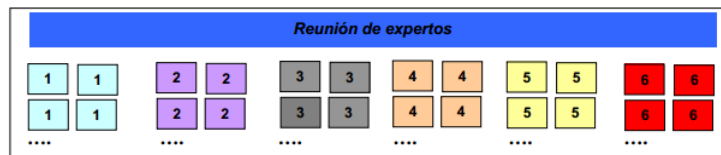


Figura 2.2: Grupos de expertos

mismo número se reúnen para debatir y comentar sobre el documento que les fue asignado. Ver figura 2.2

4. Finalizada las reuniones de expertos, llega la tercera fase, que supone el regreso al grupo original y, cada alumno explicará al resto de sus compañeros el documentos que ha estado preparando. Ver figura 2.3

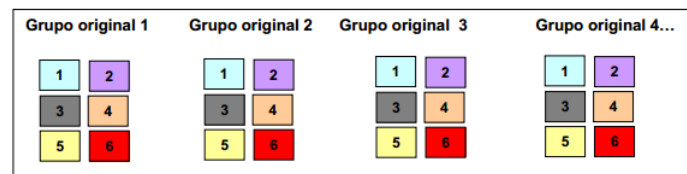


Figura 2.3: Regreso a los originales

5. La última fase, consiste en evaluar el aprendizaje logrado y la eficacia de la técnica individualmente. Para ello, el docente prepara un test sobre todo el material que ha sido trabajado durante la sesión de clase.

2.2.2. Programación en pares

Una técnica educacional que tiene elementos en común con el aprendizaje cooperativo es la programación en pares. En esta forma de colaboración, dos programadores trabajan lado a lado en un computador. En cualquier momento, un miembro del equipo (“the driver”) está escribiendo en el computador o transcribiendo algún diseño elaborado. El otro integrante del equipo (“the navigator”) está observando activamente el trabajo del primero, buscando defectos, pensando en otras alternativas de solución, haciendo preguntas, etc. Los roles de “driver” y “navigator” son intercambiados periódicamente entre ambos miembros del equipo.

La programación en pares fue originalmente popularizada como parte de la metodología de desarrollo de software XP (K. Beck, 2000). Así mismo,

¹(Servicio de innovación educativa - Universidad Politécnica de Madrid, 2008)

resultados de investigaciones muestran que los programadores en pares producen código de mayor calidad en mitad de tiempo que lo programadores individuales (L. A. Williams, 2000; L. Williams, Kessler, Cunningham, & Jeffries, 2000). La técnica de programación en pares también has mostrado ser efectiva para estudiantes de programación, logrando mejorar el aprendizaje en los alumnos(McDowell, Werner, Bullock, & Fernald, 2002).

2.2.3. El estudio de Casos

Es una técnica de aprendizaje cooperativo, la cual está basada en un enfoque de estudio de problemas. (Winter & McGhie-Richmond, 2005). Durante un Estudio de Casos, los estudiantes reciben materiales que describen una situación en concreto y se les pide analizarla tratando de identificar los puntos fuertes y débiles del caso, así como también se les pide reflexionar sobre posibles soluciones que pudiesen añadir a las ya presentadas en el problema. El factor más importante del Estudio de Casos es que éstos están basados en problemas y/o situaciones reales (Winter & McGhie-Richmond, 2005).

2.3. Tecnologías web

2.3.1. AJAX

El término AJAX es un acrónimo de Asynchronous JavaScript + XML que puede traducirse como “JavaScript asíncrono más XML”. El término AJAX se presentó por primera vez en el artículo “Ajax, A new Approach to Web Applications” publicado por Jesse James Garret en el año 2005. En realidad, en dicho artículo se dice que AJAX no es una tecnología en sí mismo sino un conjunto de varias tecnologías que se unen de formas nuevas y sorprendentes (Holzner, 2006).

AJAX está compuesto por los siguiente:

- La presentación en el navegador está basada en HTML y CSS.
- Document Object Model (DOM), para la interacción y manipulación dinámica de la presentación.
- XML, XSLT y JSON, para el intercambio y la manipulación de información
- XMLHttpRequest, para el intercambio asíncrono de información.

- JavaScript, para unir todas las tecnologías antes mencionadas.

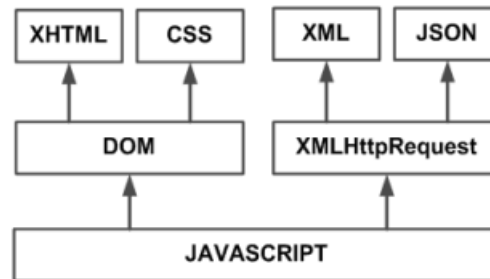


Figura 2.4: Tecnologías agrupadas bajo el concepto de AJAX

JavaScript es un lenguaje que la mayoría de navegadores soporta y que permite manipular datos entre la vista y el servidor. A continuación se indica cómo es que AJAX funciona:(Holzner, 2006)

1. En el navegador se escribe código en JavaScript con el cual se puede obtener información del servidor las veces que sean necesarias.
2. Cuando más data es requerida, el JavaScript usa el objeto XMLHttpRequest para enviar la petición al servidor sin necesidad de refrescar toda la página web.
3. La data que retorna desde el servidor puede estar en formato XML o puede ser sólo texto plano. Lo importante es que el JavaScript en el navegador puede leer la información recibida y mostrarla en la vista.

¿Qué se puede hacer con AJAX?

AJAX ha estado presente desde 1998 y muchas aplicaciones como Microsoft's Outlook Web Access la han usado. Sin embargo, recién a partir del año 2005 cuando AJAX tomó sitio en el campo del desarrollo web después de que fuese utilizado en aplicaciones como Google Suggest y Google Maps.(Holzner, 2006)

Desde entonces, AJAX ha permitido realizar diferentes tipos de aplicaciones web como por ejemplo:

- *Búsqueda en tiempo real.* Una de las mejores cosas que se puede realizar usando AJAX son las búsquedas en vivo, donde el usuario obtiene lo que busca mientras va escribiendo lo que necesita buscar.

- *Obtener la respuesta con autocompletado.* Muy parecido a las búsquedas en vivo son las aplicaciones que tratan de adivinar la palabra que estamos escribiendo obteniendo una lista de palabras similares del servidor que son puestas a nuestra vista.
- *Chatear con amigos.* Puesto que AJAX no necesita refrescar toda la página web, es una buena herramienta para realizar programas de chat donde muchos usuarios conversan al mismo tiempo.
- *Dragging and Dropping.*
- *Juegos con AJAX.*
- *Menús pop up.* Se puede obtener data desde el servidor tan pronto como sea necesario usando Ajax.

2.3.2. Polling

Después del posicionamiento de AJAX, no pasó mucho tiempo para tratar de lograr que los eventos en el navegador salieran fuera de la ecuación y se pudiera automatizar el proceso de obtener nueva información desde el servidor. Fue entonces que los desarrolladores establecieron un intervalo de actualización para revisar actualizaciones cada n segundos (Lengstorf & Leggetter, 2013). Ver figura 2.5

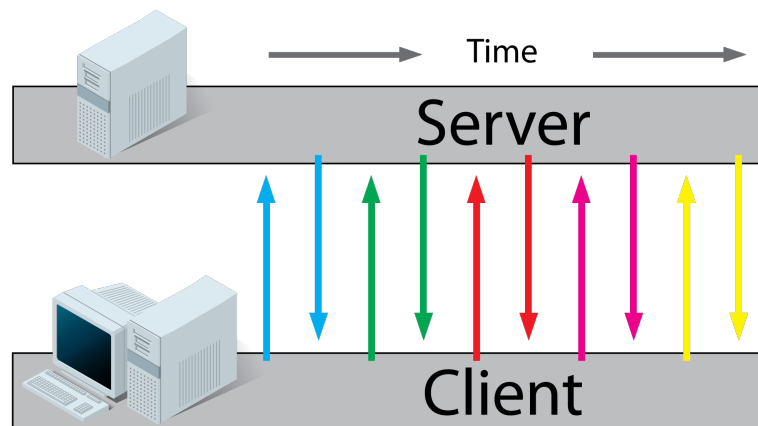


Figura 2.5: Polling
A través del Polling se envía peticiones HTTP para comprobar si existe nueva información.

2.3.3. HTTP Long-Polling

El siguiente paso en la evolución del tiempo real es el HTTP *long-polling*, el cual consiste en abrir una petición HTTP por un periodo de tiempo para escuchar respuestas del servidor. Si hubiese nueva data, el servidor la enviaría y se cerraría la petición; de otro modo, la petición es cerrada después de un intervalo de tiempo límite y se abre una nueva petición (Lengstorf & Leggetter, 2013). Ver figura 2.6

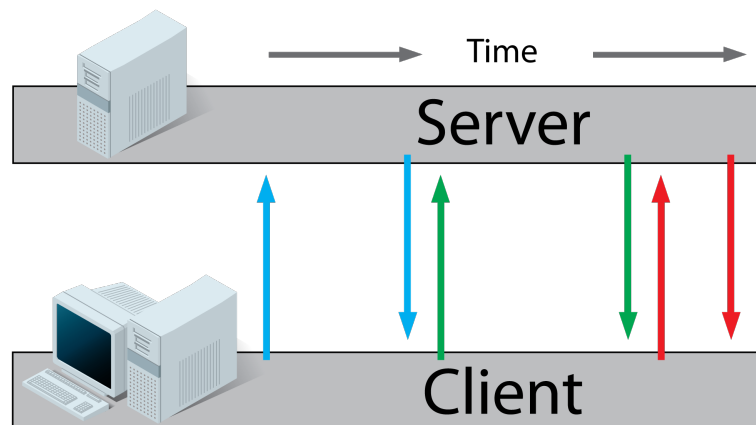


Figura 2.6: Long Polling

Gracias al HTTP Long-polling, se mantiene abierta una petición HTTP por un periodo de tiempo para comprobar si existe nueva información.

2.3.4. WebSockets

Web sockets (Ver figura 2.7) es una tecnología que proporciona un canal de comunicación bidireccional y full-duplex sobre una única conexión TCP en los navegadores y servidores web. A comparación de HTTP, WebSocket puede reducir efectivamente el tráfico de red innecesario, a través de una forma estandarizada para que el servidor envíe contenido al navegador sin que éste sea solicitado.(Cheng, Zhu, Li, & Zhou, 2013)

¿Por qué usar Websockets?

Existen muchas razones que no conllevan a utilizar websockets cuando desarrollemos una aplicación web. Algunas de ellas se mencionan a continuación:

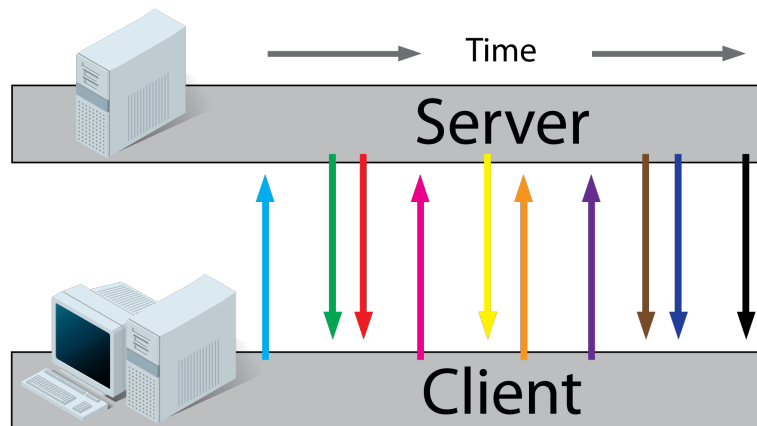


Figura 2.7: Websockets
Una nueva tecnología para la comunicación bidireccional.

- Websockets permite que la comunicación en tiempo real sea mucho más eficiente. Desde luego, siempre será posible usar polling sobre HTTP para recibir las notificaciones desde el servidor. Sin embargo, con el uso de Websockets se ahorra ancho de banda, cpu, y latencia.
- La comunicación entre el cliente y el servidor se vuelve más sencilla.

Los websockets tiene una gran acogida de parte de la comunidad de desarrolladores, lo cual se ve reflejado en la variedad de implementaciones de websockets como Apache mod_pywebsocket, Jetty, Socket.IO, entre otros.

2.4. Glosario

1. **Framework.** La palabra inglesa "framework" (marco de trabajo) define, en términos generales, un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular que sirve como referencia, para enfrentar y resolver nuevos problemas de índole similar.
2. **Librería.** En informática, una librería es un conjunto de implementaciones funcionales, codificadas en un lenguaje de programación, que ofrece una interfaz bien definida para la funcionalidad que se invoca.
3. **Socket.** Designa un concepto abstracto por el cual dos programas (posiblemente situados en computadoras distintas) pueden intercambiar cualquier flujo de datos, generalmente de manera fiable y ordenada.

4. **Tiempo real.** Un sistema en tiempo real (STR) es aquel sistema digital que interactúa activamente con un entorno con dinámica conocida en relación con sus entradas, salidas y restricciones temporales, para darle un correcto funcionamiento de acuerdo con los conceptos de predictibilidad, estabilidad, controlabilidad y alcanzabilidad
5. **CSCL.** Computer-Supported Collaborative Learning. Aprendizaje colaborativo apoyado por computador.
6. **API.** Application Programming Interface. Es el conjunto de funciones y procedimientos que ofrece cierta librería para ser utilizado por otro software como una capa de abstracción. Representa la capacidad de comunicación entre componentes de un software.

Capítulo 3

Estado del Arte

En este capítulo se desarrollo el estado del arte de las herramientas para el aprendizaje colaborativo como son LearnCS, GoogleDocs, entre otros, las mismas que son analizadas en un cuadro comparativo. Además, también se describe el estado del arte de 2 técnicas de aprendizaje colaborativo que serán usadas para la presente tesis. Finalmente, se desarrolla el estado del arte de algunos frameworks y APIs que serán utilizadas para la implementación del sistema web para el aprendizaje colaborativo.

3.1. Herramientas para el aprendizaje colaborativo

3.1.1. LearnCS

LearnCS! es un entorno de programación creado específicamente para el uso de estudiantes 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).

En muchos cursos de programación de primer ciclo, el concepto de depurar un programa es enseñado a finales del curso. En cambio, a través de *LearnCS!*, los alumnos pueden establecer puntos de corte en el programa y empezar a depurar paso a paso su código fuente. Además, el sistema brinda

la opción de ver la representación de la memoria y así visualizar en detalle el estado de su programa. Cuando se llaman a funciones, los alumnos aprenden cómo los argumentos y variables locales son colocados en la pila de la memoria y cómo las variables son reservadas en memoria (Lipman, 2014).

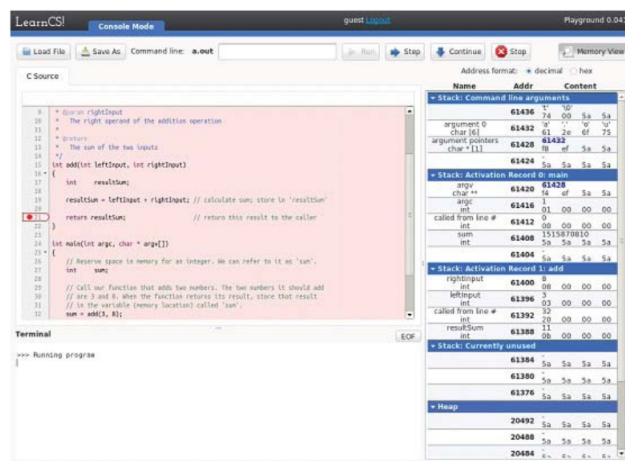


Figura 3.1: Interfaz del sistema LearnCS! (Lipman, 2014)

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 nocional 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).

LearnCS! se ejecuta en un navegador web y tal como se muestra en la Figura 3.1, ofrece lo siguiente:

1. (Panel superior-izquierdo) Un área de la pantalla dedicada a la edición del programa que se está desarrollando.
2. (Panel derecho) Una representación de la vista de la memoria.
3. (Panel inferior) Una zona de “salida” que contiene una “terminal” para la entrada y salida de texto.
4. (Panel inferior-derecho) Opcionalmente, como se muestra en la figura 3.2, se muestra un área gráfica que permite desarrollar programas más interactivos y más interesantes para los alumnos.

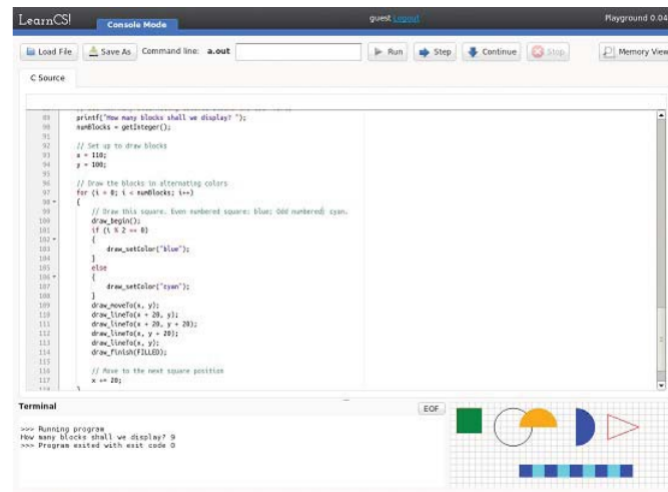


Figura 3.2: Área gráfica para programas más interactivos en LearnCS! (Lipman, 2014)

3.1.2. Google Docs

Google Docs es un conjunto de herramientas online que permiten elaborar documentos, hojas de cálculo, dibujos y diapositivas de manera colaborativa usando solamente un navegador web. Esta herramienta es un gestor de documentos pues a través de ella se pueden subir a la nube todo tipo de archivos y ordenarlos en carpetas así como compartirlos con otros usuarios (Google, 2014).

Para acceder a esta herramienta de colaboración, en primer lugar se debe tener una cuenta de gmail y con ello se obtiene el acceso a Google Docs a través del siguiente link <http://docs.google.com/>. Las funcionalidades más resaltantes que posee GoogleDocs son las siguientes: (Google, 2014)

1. *Crear documentos básicos desde cero.* Con GoogleDocs es posible realizar tareas que corresponden al manejo de documentos de oficina, como crear documentos de texto, hojas de cálculo, presentaciones de diapositivas, y añadirles a todos ellos imágenes, comentarios o fórmulas, entre otras muchas cosas más.
2. *Subir archivos ya creados.* Existe la opción de subir archivos que ya están creados y para ellos, Google Docs acepta la mayoría de formatos de archivos comunes como DOC, XLS, ODT, CSV, PPT, PDF, etc.

3. *Editar un documento.* GoogleDocs posee una barra de herramientas que nos permite aplicar negrita, subrayar, cambiar la fuente, color, etc.
4. *Compartir documentos.* Cuando se edita un documento, existe la opción de invitar a otros usuarios, enviar el documento por correo electrónico, publicarlo como página web, ver quién tiene acceso, etc.
5. *Chatear en tiempo real con otros usuarios.*
6. *Exportar documentos.* Google Docs permite descargar nuestros documentos en el escritorio en archivos Word, Excel, OpenOffice, RTF, PDF, HTML o ZIP.



Figura 3.3: GoogleDocs (Google, 2014)

3.1.3. Sistema Web para la enseñanza de Casos de Uso empleando la Técnica de Aprendizaje Cooperativo de Rompecabezas.

Este sistema es producto de una tesis de grado implementada en la Pontificia Universidad Católica del Perú. El sistema pretende dar soporte a las tres fases que comprende una clase en la cual se emplea la técnica de Jigsaw para lo cual, se construyeron los módulos de Planificación, Ejecución y Evaluación.

El módulo de Planificación permite realizar el diseño de cada sesión de clase. Ahí se plantean los datos de la sesión que serán la base de los objetivos y resultados esperados que permitirán medir el progreso académico de los alumnos.

El módulo de Ejecución se encarga de llevar a cabo la ejecución de una sesión de clase basada en la técnica de Jigsaw. Permite el desarrollo paso a paso desde la lectura de materiales, documentos y casos hasta la diagramación de la solución que brinden cada uno de los grupos Jigsaw y Expertos. En este módulo se cuenta con foros de discusión que permiten la comunicación entre los miembros de cada grupo.

Por último se tiene el módulo de Evaluación, en el cual se elaboran preguntas y exámenes que luego el profesor aplica a sus alumnos. Estos exámenes son calificados manualmente o de forma automática por el propio sistema.

El sistema fue desarrollado usando una arquitectura Modelo-Vista-Controlador. Se usó Java como plataforma de desarrollo y MySQL como motor de base de datos. A continuación se indican los frameworks utilizados en el sistema:

- Framework J2EE
- Struts 2
- MyBatis
- Librerías AJAX: JQuery, DojoToolKit

El sistema desarrollado consta de 3 módulos que se detallan a continuación:

Planificación

Este módulo posee las siguientes funcionalidades:

1. Crear dinámica de Curso
2. Consultar información del curso
3. Consultar configuración de dinámica
4. Consultar información de dinámica
5. Actualizar mensaje interno

Ejecución

Este módulo se encarga de llevar a cabo el desarrollo de una sesión basada en la técnica de aprendizaje colaborativo de rompecabezas. Permite la lectura de materiales y documentos, hasta la elaboración colaborativa de la solución a los diferentes problemas propuestos en los diferentes grupos Jigsaw y Expertos.

El módulo cuenta con herramientas de chat para la comunicación online, así como también de un foro para la discusión de los temas.

Este módulo posee las siguientes funcionalidades:

1. Actualizar publicación en foro
2. Ingresar sesión cooperativa
3. Controlar estado de la dinámica
4. Elaborar casos de uso,.

Evaluación

En esta parte del sistema es donde se elaboran las preguntas y exámenes por parte del docente y posteriormente se aplican a los alumnos. El sistema permite que estos exámenes sean evaluados manualmente o de forma automática.

Este módulo posee las siguientes funcionalidades:

1. Consultar configuración de evaluación.
2. Consultar corrección de evaluación.
3. Consultar calificación de evaluación.

4. Crear evaluación.
5. Rendir evaluación.
6. Realizar calificación.

Fuente (Pinzás & Yatsen, 2013)

3.1.4. CodeBunk

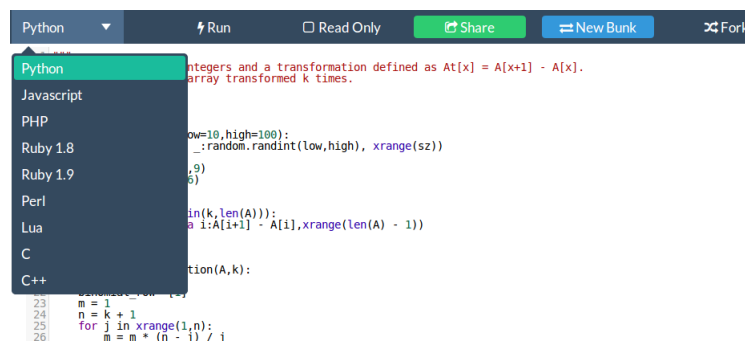


Figura 3.4: Interfaz gráfica de Codebunk (CodeBunk, 2014)

Es una plataforma que permite codificar y compilar en diferentes lenguajes de programación de forma colaborativa y en tiempo real. Algunas de sus funcionalidades son las siguientes:

1. Posee un editor colaborativo que soporta 14 lenguajes de programación, tiene coloración de sintaxis según el lenguaje y brinda un indentado inteligente.
2. Compilar y Ejecutar. La plataforma permite compilar y ejecutar código en Python, Java, C, C++, Ruby, Javascript, entre otros.
3. Audio y Video Chat.
4. Code playback. Permite repetir la historia de los cambios realizados en el código.
5. Equipos. Permite crear equipos de trabajo e invitar a otros usuarios a colaborar en el desarrollo de un programa.

3.1.5. Análisis comparativo

Las cuatro herramientas para el aprendizaje colaborativo que se han estudiado hasta el momento son resumidas y comparadas en el siguiente cuadro, en el cual se podrá apreciar las características que nos ofrece cada una de las cuatro aplicaciones presentadas anteriormente.

Tabla 3.1: Herramientas para el aprendizaje colaborativo

CARACTERÍSTICAS	LearnCS	Google Docs	Sistema PUCP	Code Bunk
◊ Permite escribir y compilar código fuente	✓	✗	✗	✓
◊ Posee un representación de la memoria del computador	✓	✗	✗	✗
◊ Muestra mensajes de error	✓	✗	✗	✓
◊ Es un aplicativo web	✓	✓	✓	✓
◊ Permite crear y editar documentos de texto	✗	✓	✗	✗
◊ Permite crear y editar hojas de cálculo	✗	✓	✗	✗
◊ Permite crear y editar diapositivas	✗	✓	✗	✗
◊ Compartir documentos con otros usuarios	✗	✓	✗	✓
◊ Chat de tiempo real	✗	✓	✗	✓
◊ Permite diseñar una sesión de clase jigsaw	✗	✗	✓	✗
◊ Permite tomar y calificar evaluaciones	✗	✗	✓	✗
◊ Brinda un foro de discusión	✗	✓	✓	✗
◊ Permite elaborar casos de uso	✗	✗	✓	✗
◊ Permite desarrollar código fuente entre varios usuarios conectados al mismo tiempo	✗	✗	✗	✓
◊ Coloración de sintaxis de código fuente	✓	✗	✗	✓
◊ Indentado inteligente	✗	✗	✗	✓
◊ Visualización de resultados de ejecución	✗	✗	✗	✓
◊ Historial de revisiones de código	✗	✓	✗	✓
◊ Creación de equipos colaborativos	✗	✓	✗	✓

◇ Juez virtual de programación	X	X	X	X
◇ Banco de preguntas de programación	X	X	X	X

3.2. Técnicas para el aprendizaje colaborativo

3.2.1. 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](#)).

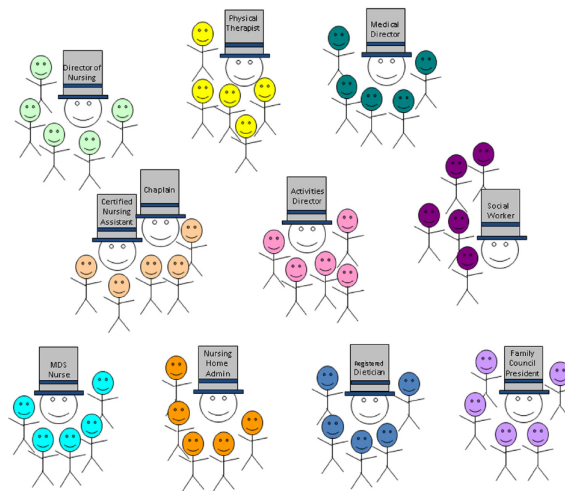


Figura 3.5: Jigsaw Paso 1
Fase de reunión de grupos de expertos ([Buhr y cols., 2014](#))

Los objetivos del ejercicio jigsaw realizado fueron para definir LTPAC, identificar cuáles son los alcances de este tipo de servicios que se brindan a los pacientes y describir los roles de los miembros del equipo que laboraba

en el asilo de ancianos. Para esto, 10 miembros claves del equipo LTPAC fueron identificados y un total de 50 estudiantes fueron divididos en 10 grupos expertos. A cada grupo experto le fue asignado de 30 a 45 minutos para encontrar y entrevistar a 1 miembro del equipo LTPAC (Ver figura 3.5). Cada estudiante en el grupo tenía que convertirse en un experto sobre el rol del miembro que le fue asignado. Después de las entrevistas, los estudiantes tuvieron un tiempo de 15 minutos para debatir en sus grupos de expertos sobre los aspectos más resaltantes del rol del profesional al que entrevistaron. Finalmente, los grupos fueron reorganizados conteniendo un estudiante experto de cada grupo original así como un moderador (Ver figura 3.6). Los nuevos grupos Jigsaw se reunieron por 1 hora durante la cual cada estudiante expuso los resultados de la entrevista que había realizado al profesional LTPAC del asilo (Buhr y cols., 2014).

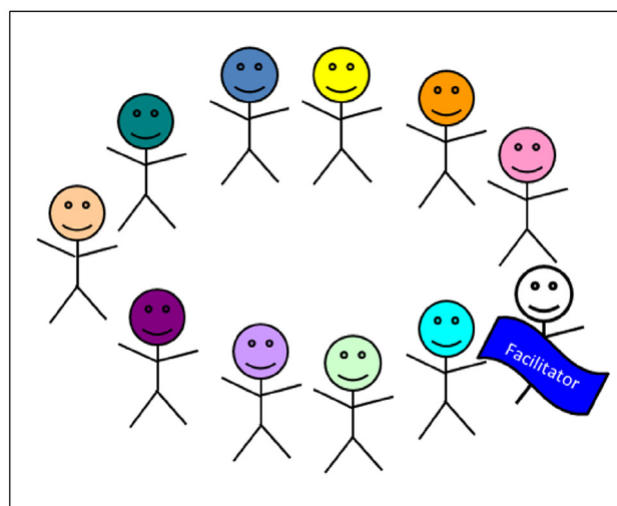


Figura 3.6: Jigsaw Paso 2
Fase de reunión de grupos Jigsaw dirigidos por un moderador (Buhr y cols., 2014)

La efectividad de esta experiencia de aprendizaje fue medida a través de los comentarios por escrito de los estudiantes y el personal LTPAC así como también a través de una encuesta con una escala de 5 puntos donde 1 reflejaba malo y 5 significaba bueno. Los estudiantes también fueron evaluados con un examen de conocimientos sobre el tema tratado en la actividad desarrollada en el asilo. Los resultados de este examen indicaron que los estudiantes aprendieron los roles del personal LTPAC de forma satisfactoria. En general, el ejercicio jigsaw fue bien recibido por todos los participantes y mostró ser

un medio eficaz para introducir a los estudiantes de medicina en el mundo del cuidado de ancianos (Buhr y cols., 2014).

3.2.2. Pair Research

En el estudio “*Pair Research: Matching People for Colaboration, Learning, and Productivity*” elaborado por Miller, Zhang, Gilbert, y Gerber (2014) se describe una nueva forma de interacción denominada “*Pair Research (Investigación en pares)*” que permite incrementar la productividad, el aprendizaje y la colaboración entre los diferentes grupos de investigación. A lo largo del artículo, los autores detallan un forma de cómo elaborar los grupos y además presentan los resultados obtenidos, los cuales mostraron que los miembros de los equipos usaban la investigación en pares en diferentes maneras incluyendo programación en pares, lluvia de ideas, recolección de data y análisis.

Pair Research es una generalización de la programación en pares. Cada semana, los miembros de los grupos son emparejados guiados por algoritmo de emparejamiento. Cada pareja se reúne por una a dos horas de las cuales la mitad del tiempo es dedicado a trabajar en el proyecto de la otra persona. El trabajo puede ser cualquier actividad relacionada a programación, pruebas, diseño, recolección de datos y análisis, lluvia de ideas y asesoría. La siguiente semana, se forman nuevas parejas y el proceso se repite.

Miller y cols. (2014) elaboraron un prototipo de hoja de cálculo para gestionar la investigación en pares de tal forma que cada semana, el sistema automáticamente hacía recordar a los miembros del grupo que envíen en qué necesitaban ayuda y en qué aspecto cada uno podía ayudar al resto para luego realizar el respectivo emparejamiento y notificar a cada miembro quien sería su siguiente pareja. Lo principal del prototipo elaborado (Ver figura 3.7) es la matriz de preferencias para el emparejamiento de personas. Cada semana, los miembros de cada grupo especificaban la ayuda que necesitaban en esa semana como por ejemplo: “depurar programa en Django” o “revisar un informe”. Otros miembros luego completaban la matriz de preferencias de acuerdo a si podían o estaban interesados en ayudar en algún tema en específico; Para esto, ellos tenían que colocar un número según si preferencia siendo 1 el número que refleja mayor interés, -1 el mayor desinterés y 0 indicaba neutro (Miller y cols., 2014)

La matriz de preferencias sirve para influir en la asignación de parejas

Pair Research pool

File Edit View Insert Format Data Tools Help Paired Research All changes saved in Drive

fx LID

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	
1	Paired Research					what help can you provide?													
2	for the week of 13-May-2013					fill out the column below your name, using: 1 if you can help that person													
3						-1 if you definitely can't help													
4						0.1 - 0.9 if you're in between													
5	In the ready column, "now" means you're available for paired research this week; otherwise put the date when you'll be available for paired research again.	ready	group	can pair with	what help do you need?	Tish Schenk tischenk@gmail.com	Kenya Manner kenyam@gmail.com	Hulu Rey hulu@gmail.com	Rolando Schenk rolanda@gmail.com	Gaynell Manner gaynell@gmail.com	Mandie Alba chowdh@gmail.com	Judy Chowdhury chowdh@gmail.com	Laveta Conigle conigle@gmail.com	Reanna Syring reanna@gmail.com	Jackie Granger buesing@gmail.com	Lorita Resnick resnick@gmail.com	Asha Lang asha@gmail.com	Tama Sax tamasax@gmail.com	
6	Tish Schenk tischenk@gmail.com	now	GSU	any group	brainstorming on life photo visualization														
7	Kenya Manner kenyam@gmail.com	now	GSU	any group						0.5								-1	
8	Hulu Rey hulu@gmail.com	now	LID	any group															
9	Rolando Schenk rolanda@gmail.com	now	LID	any group	thinking about online video learning					0.5					0.7	1			
10	Gaynell Manner gaynell@gmail.com	now	LID	any group	thesing									0.7					
11	Mandie Chowdhury chowdh@gmail.com	now	LID	any group	in-depth idea bouncing					1			0.8						
12	Judy Conigle conigle@gmail.com	now	USW	any group															
13	Laveta Syring laveta@gmail.com	now	USW	any group	pilot study														
14	Reanna Granger reanna@gmail.com	now	USW	my group only	thesing														
15	Jackie Buesing buesing@gmail.com	now	USW	my group only															
16	Lorita Resnick resnick@gmail.com	20-May-2013	USW	any group															
17	Asha Lang asha@gmail.com	now	USW	any group															
18	Tama Sax tamasax@gmail.com	now	USW	any group	writing feedback												0.7		

Figura 3.7: Pair Research
Hoja de excel utilizada para la matriz de preferencias y emparejamiento (Miller y cols., 2014)

de tal forma que aquellos con preferencia mutua positiva sean siempre emparejados mientras que aquellos con preferencia mutua negativa nunca sean emparejados y además se empareje de forma aleatoria a los miembros con preferencia mutua neutral. Para generar las vinculaciones, el sistema utiliza las preferencias ingresadas para formar un grafo ponderado cuyos nodos representan a los miembros del grupo y el grafo contiene un arco entre dos nodos si y solamente si ambos miembros tienen preferencias no negativas. El peso de una arista entre dos miembros es el promedio de sus preferencias mutuas más un plus siempre que ambos miembros no hayan sido emparejados recientemente. Por otro lado, para el caso de aquellos miembros con preferencias neutrales, el sistema agrega una perturbación aleatoria para hacer la asignación entre dichos miembros del grupo (Miller y cols., 2014).

La finalidad de la investigación en parejas o investigación en pares fue mejorar la productividad, el aprendizaje informal y la colaboración. En cuanto a la productividad, los resultados del estudio mostraron que los miembros se sintieron más motivados al trabajar usando esta técnica y realizaron satisfactoriamente sus trabajos asignados. Respecto al aprendizaje, se observó mejoras sobre herramientas específicas, habilidades, trabajos prácticos, aprendizaje sobre los demás miembros de grupo. Finalmente, en cuanto a colaboración, los integrantes de los grupos lograron interactuar con más personas de lo que usualmente lo harían lo cual incrementó sus oportunidades para colaboraciones futuras (Miller y cols., 2014).

3.2.3. Análisis comparativo

En el siguiente cuadro se resumen las características principales que nos ofrece cada una de las dos técnicas de aprendizaje colaborativo que han sido presentadas en los apartados anteriores.

Tabla 3.2: Técnicas de aprendizaje colaborativo

CARACTERÍSTICAS	JIGSAW	PAIR RESEARCH
◊ Se aprende del compañero	✓	✓
◊ Permite las revisiones de trabajo constantes	✗	✓
◊ Desarrolla la creatividad	✓	✓
◊ Mejora la comunicación del estudiante	✓	✓
◊ Favorece el aprendizaje del tema a tratar	✓	✓
◊ Existe un método para generar los emparejamientos	✗	✓
◊ Se requiere la dirección de un profesor	✓	✗
◊ Permite evitar conflictos interpersonales	✗	✓
◊ Favorece la integración con nuevas personas	✓	✗
◊ El alumno elige con quién trabajar	✗	✓
◊ Los grupos se forman según estilos de aprendizaje	✗	✗

3.3. Frameworks para aplicaciones web de tiempo real

3.3.1. GoogleDrive Realtime API

La API de GoogleDrive en tiempo real ofrece el trabajo colaborativo como un servicio para los archivos en Google Drive a través del uso de las transformaciones operativas. El API es una biblioteca JavaScript que ofrece a los desarrolladores objetos de colaboración, eventos y métodos para la creación de aplicaciones en las cuales se puedan realizar tareas colaborativas.

Esta API permite a los desarrolladores diseñar un modelo de datos común que se ve como un modelo local en memoria. Se pueden escribir código para manipular listas, arrays, matrices, maps, y objetos javascript propios del desarrollador. Cada vez que se modifique un modelo de datos, éste automáticamente cambiará para todos los usuarios presentes en el documento.

La API está basada en la misma tecnología de colaboración usada por GoogleDocs y por ello, cada vez que un modelo de datos es modificado, el cambio es aplicado inmediatamente a la copia local del documento y luego, la API envía una representación del cambio al servidor de tal forma que el cambio es guardado en el documento y comunicado a los demás colaboradores.

La API en tiempo real se encarga de todos los aspectos de la transmisión de datos, el almacenamiento y la resolución de conflictos cuando varios usuarios están editando un archivo. En general, la API nos brinda lo siguiente:

1. Funciones para cargar y trabajar documentos en tiempo real.
2. Objetos contruídos colaborativamente(cadenas, listas, y mapas).
3. La capacidad de crear objetos propios que puedan personalizarse.
4. Eventos para la detección de cambios en el modelo de colaboración y detección del ingreso o salida de colaboradores.

La API en tiempo real de GoogleDrive proporciona todas las herramientas que se necesitan para crear una aplicación colaborativa que no necesita correr en nuestro propio servidor.

3.3.2. 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).

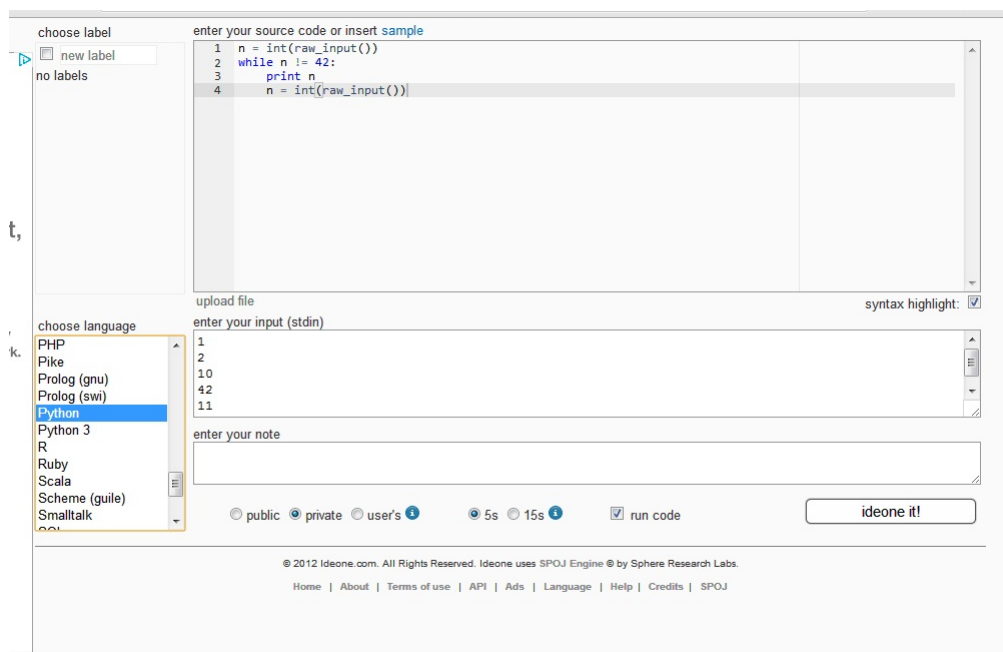


Figura 3.8: Interfaz gráfica de Ideone (Ideone, 2014)

3.3.3. Análisis comparativo

Tanto la API de GoogleDrive como la API de Ideone ofrecen características importantes para la elaboración de aplicaciones web de tiempo real, las mismas que se detallan en el siguiente cuadro comparativo.

Tabla 3.3: Frameworks para aplicaciones web de tiempo real

CARACTERÍSTICAS	Google Drive Realtime API	Ideone API - Sphere Engine
◇ Se implementa a través de Javascript	✓	✗
◇ Creación y edición de listas, arrays, matrices, maps, objetos javascript personalizados	✓	✗
◇ Detección de cambios	✓	✗
◇ Detección de ingreso o salida de colaboradores	✓	✗
◇ Ejecutar código en múltiples lenguajes de manera online	✗	✓
◇ Servicio web vía protocolo SOAP	✗	✓
◇ Permite cargar código fuente	✗	✓
◇ Permite ejecutar código fuente	✗	✓
◇ Permite subir data de prueba inicial	✗	✓
◇ Permite descargar los resultados obtenidos en la ejecución	✗	✓

Capítulo 4

Aporte práctico

En el capítulo 3 se presentaron algunas técnicas y herramientas que sirven para el aprendizaje colaborativo; No obstante, para objetivos de esta tesis, se requiere una herramienta que integre ciertas características de las que ya poseen las mencionadas en el capítulo anterior. Ver Tabla [3.1](#).

El sistema que se pretende desarrollar estará basado en el proceso para la elaboración de sesiones de clase usando la técnica de Jigsaw con la particularidad de que en esta ocasión la generación de grupos expertos no será siempre de forma aleatoria sino que también se podrá realizar basándose en información del estudiante respecto a su estilo de aprendizaje y preferencias sobre el tema a desarrollar y los miembros posibles para su grupo. Esta última característica no forma parte de la estructura tradicional de la técnica de Jigsaw y la técnica de Pair Research como se puede ver en la tabla [3.2](#).

El sistema a desarrollar también tendrá la particularidad de que permitirá a los estudiantes trabajar de forma colaborativa en la elaboración de archivos de código fuente, los cuales serán producto de las respuestas que dichos alumnos darán a los problemas que les plantee el profesor durante la sesión de clase jigsaw. Por ende, para lograr este objetivo y tener dicha funcionalidad en el sistema se usará los frameworks y APIs descritas en el capítulo anterior. Ver tabla [3.3](#).

Finalmente, par el desarrollo del sistema web que se plantea en esta tesis se va a considerar como proceso de desarrollo a las mejores prácticas de RUP para garantizar un producto software de calidad.

4.1. Las mejores prácticas de RUP

EL Proceso Unificado Rational(RUP) es un proceso de ingeniería de software que provee un enfoque para la asignación de tareas y responsabilidades durante el desarrollo de un software. Tiene como objetivo asegurar la producción de un producto software de alta calidad que satisfaga los requerimientos de los usuarios finales dentro de un tiempo y presupuesto establecido (IBM, 2014). Ver figura 4.1

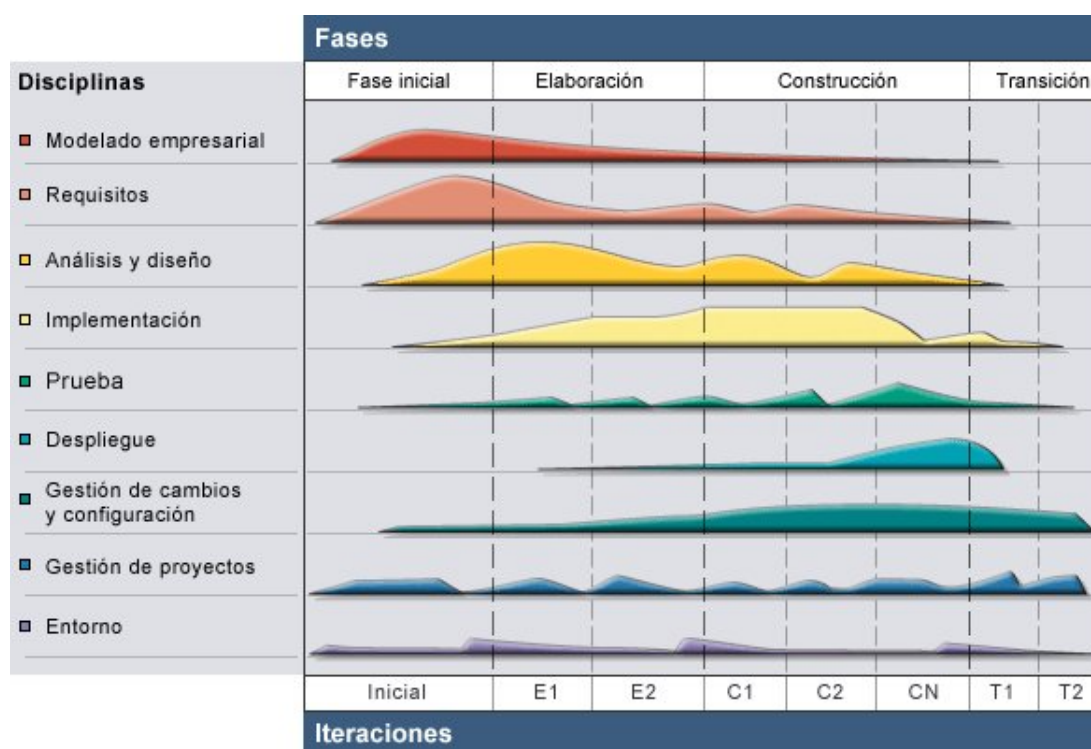


Figura 4.1: Proceso de desarrollo de software - RUP (Rational Method Composer, 2014)

Así mismo, RUP también es una guía para usar de manera efectiva el Lenguaje Unificado de Modelado (UML) que no es más que un lenguaje estándar que permite comunicar claramente los requerimientos, arquitecturas y diseños (IBM, 2014).

Es por ello que el desarrollo del sistema propuesto por esta tesis estará guiado por la buenas prácticas de RUP, estableciéndose iteraciones semanales en las cuales se irá desarrollando cada una de las fases en las que

RUP divide el ciclo de desarrollo de software: Inicio, Elaboración, Construcción y Transición. Los artefactos que serán entregados durante el proceso de desarrollo del sistema web de tiempo real para el aprendizaje colaborativo propuesto en esta tesis se mencionan en la Tabla 4.1. Dichos artefactos se encuentran agrupados según la disciplina a la que pertenecen.

Tabla 4.1: Artefactos del proceso de desarrollo

DISCIPLINA RUP	ARTEFACTO
Requisitos	<ul style="list-style-type: none"> • Modelo de caso de uso • Especificaciones de casos de uso • Especificaciones suplementarias
Análisis y diseño	<ul style="list-style-type: none"> • Diagrama de clases • Modelo de datos • Prototipo de interfaz de usuario • Documento de arquitectura de software
Implementación	<ul style="list-style-type: none"> • Código fuente • Sistema web desplegado
Prueba	<ul style="list-style-type: none"> • Casos de prueba • Resultados de prueba

Para el desarrollo del sistema web, se seguirá el siguiente cronograma, el mismo que se está orientado a seguir las actividades y tareas que plantea RUP. Se indica también las fechas en las cuales se estará aplicando el sistema al caso de estudio.

CRONOGRAMA DE TRABAJO				
Semana	Fecha	ACTIVIDAD	FASE RUP	Artefactos
S5	11/09/2014	ITERACIÓN 1	INICIO, ELABORACIÓN	Modelo de caso de uso, Especificación de casos de uso, Especificaciones suplementarias
S6	18/09/2014	ITERACIÓN 2	INICIO, ELABORACIÓN	Diagrama de clases, Prototipos de interfaz de usuario
S7	25/09/2014	ITERACIÓN 3	ELABORACIÓN	Documento de arquitectura, Modelo de base de datos
S8	2/10/2014	ITERACIÓN 4	CONSTRUCCIÓN	Código fuente: CU001, CU002, CU003
S9	9/10/2014	ITERACIÓN 5	CONSTRUCCIÓN	Código fuente: CU004, CU005, CU006
S10	16/10/2014	ITERACIÓN 7	CONSTRUCCIÓN	Código fuente: CU007, CU008, CU009
S11	23/10/2014	ITERACIÓN 8	ELABORACIÓN, CONSTRUCCIÓN	Código fuente: CU010, CU011, CU012 Casos de prueba
S12	30/10/2014	ITERACIÓN 9	TRANSICIÓN	Código fuente: Revisión CU001 - CU012 Casos de prueba
S13	6/11/2014	Caso de Estudio y Pruebas		
S14	13/11/2014	Caso de Estudio y Pruebas		
S15	20/11/2014	Conclusiones y Trabajo Futuro - Referencias		
S16	27/11/2014	Actualización del Documento de Tesis, Artículo		
S17	4/12/2014	Presentación ante jurado		

Figura 4.2: Cronograma de trabajo

4.2. Implementación de un sistema web de tiempo real para el aprendizaje colaborativo

4.3.

Apéndice A

Casos de Uso

A.1. Introducción

A.1.1. Propósito

El propósito de este documento es describir de una forma clara y concreta cada uno de los casos de uso definidos para el sistema web a implementar.

A.1.2. Alcance

Los casos de uso presentados en este documento representan los requerimientos que se desean implementar en el sistema web para el aprendizaje colaborativo.

A.1.3. Definiciones, acrónimos y abreviaciones

1. CU: Caso de uso
2. SAC: Sistema para el aprendizaje colaborativo

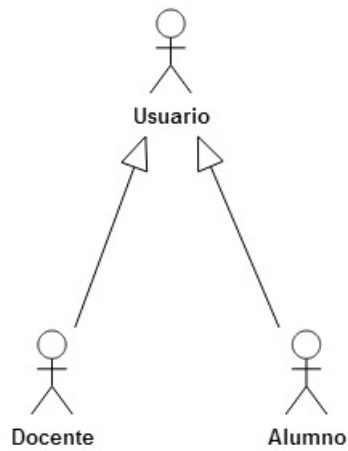


Figura A.1: Diagrama de actores

A.2. Catálogo de actores

En el la figura A.1 se puede ver los actores que participan en el SAC y en la Tabla A.1 se encuentra una breve descripción de cada uno de los actores.

Tabla A.1: Actores

ACTOR	DESCRIPCIÓN
Usuario	Persona que usará el sistema web de tiempo real para el aprendizaje colaborativo.
Docente	Es la persona responsable de crear y dirigir las sesiones de clase que serán aplicadas a los alumnos. Además, el docente es el responsable de las evaluaciones que rendirán los alumnos una vez terminada cada sesión de clase.
Alumno	Es la persona que será instruida en temas de algoritmos y programación a través de cada sesión diseñada por el docente.

A.3. Diagramas de casos de uso

A.3.1. Casos de uso - Docente

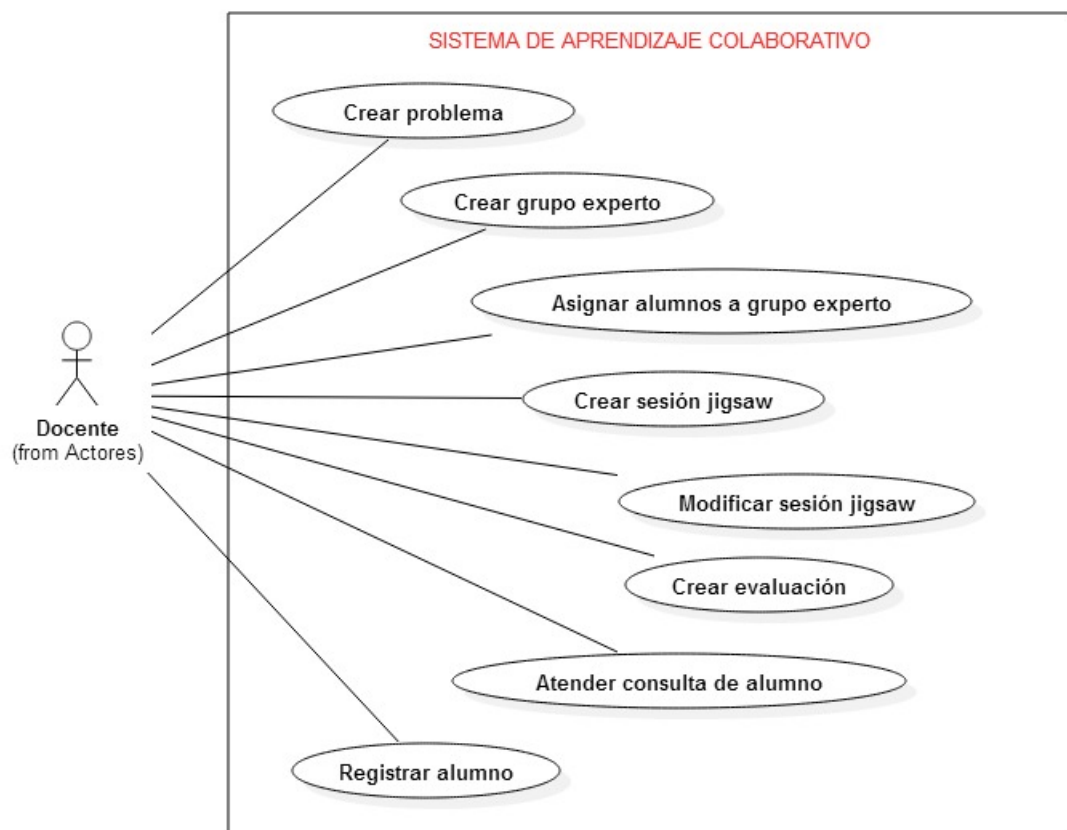


Figura A.2: Casos de uso - Docente

A.3.2. Casos de uso - Alumno

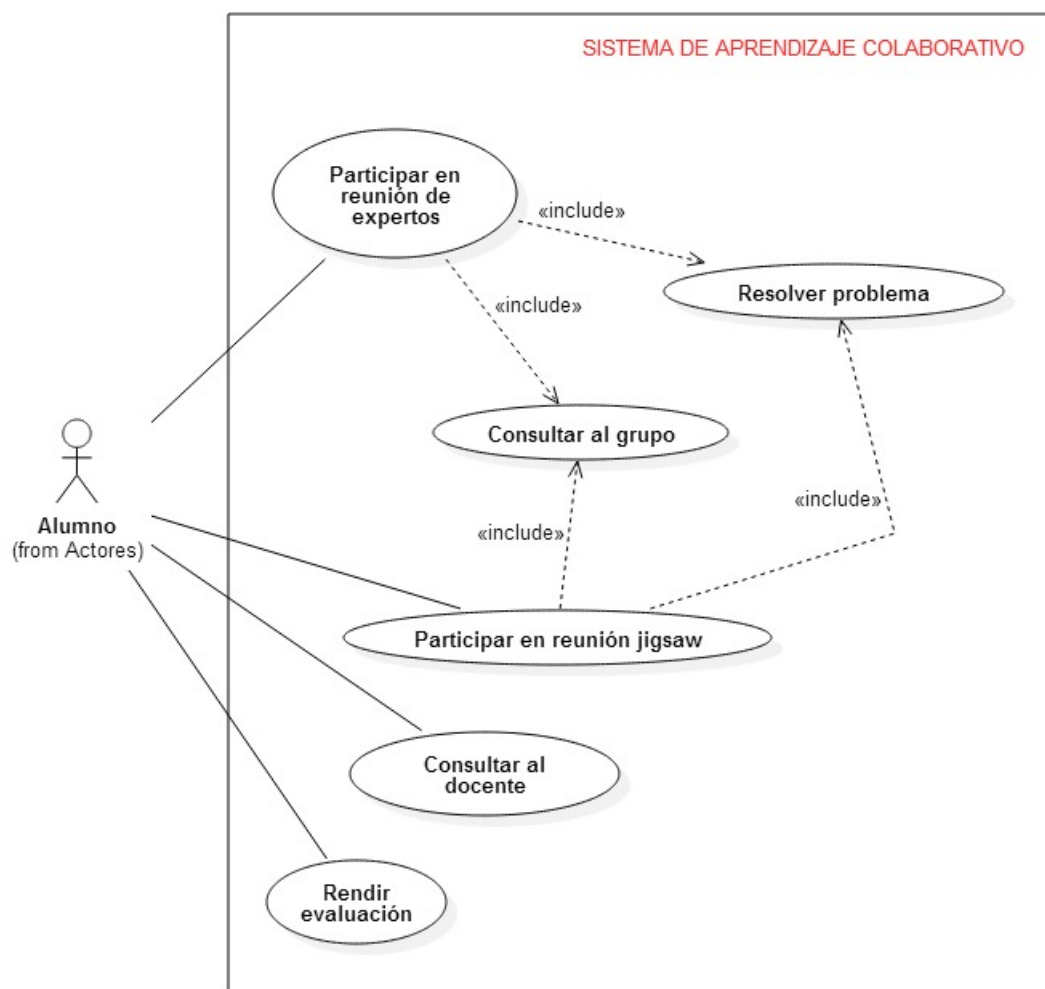


Figura A.3: Casos de uso - Alumno

A.4. Especificaciones de Casos de Uso

A.4.1. Registrar Alumno

Código	CUS-1
Nombre	REGISTRAR ALUMNO
Descripción	El caso de uso inicia cuando el docente selecciona la opción registrar nuevo alumno. Luego completa la información del alumno y el caso de uso termina cuando se presiona la opción Guardar
Actores	Docente
Precondiciones	El docente debe estar logueado en el sistema.
Postcondiciones	Ninguna
Flujo básico	<ol style="list-style-type: none">1. El docente selecciona la opción para registrar un nuevo alumno.2. El sistema muestra el formulario para completar los datos del alumno.3. El docente completa los campos DNI, Apellido Paterno, Apellido Materno, Nombres, Email, Sexo, Email, Contraseña, Repetir Contraseña.4. El sistema valida el formato del DNI, apellidos, nombres, email.5. El sistema valida que el email no esté registrado en el sistema.6. El sistema verificar que los campos contraseña y repetir contraseña sean idénticos.7. El sistema registra al nuevo alumno en el sistema.
Flujo alternativo	Ninguno

A.4.2. Crear Problema

Código	CUS-2
Nombre	CREAR PROBLEMA
Descripción	El caso de uso inicia cuando el usuario accede al sistema y elige la opción crear problema. Luego el usuario redacta el enunciado y demás observaciones sobre el problema y finaliza el caso de uso
Actores	Docente
Precondiciones	El usuario de ser un Docente y estar logueado en el sistema
Postcondiciones	El docente verá en la lista de problemas el problema recién creado
Flujo básico	<ol style="list-style-type: none">1. El docente elige la opción NUEVO PROBLEMA2. El sistema solicita al docente ingresar el título y enunciado del problema3. El docente escribe el título y enunciado del problema4. El docente selecciona la opción GUARDAR5. El sistema guarda el nuevo problema
Flujo alternativo	Ninguno

A.4.3. Crear grupo experto

Código	CUS-3
Nombre	CREAR GRUPO EXPERTO
Descripción	El caso de uso inicia cuando el usuario accede al sistema y elige la opción crear grupo experto. Luego completa la información sobre el grupo experto a crear y finalmente el caso de uso termina cuando el sistema crea experto.
Actores	Docente
Precondiciones	El usuario de ser un Docente y estar logueado en el sistema.
Postcondiciones	El docente verá en el listado de grupos expertos el nuevo grupo creado
Flujo básico	<ol style="list-style-type: none">1. El docente elige la opción NUEVO GRUPO EXPERTO.2. El docente digita el nombre para el grupo experto3. El docente indica el máximo de integrantes que se pueden incluir.4. El docente agrega al grupo experto a los alumnos que se encuentren disponibles.5. El docente selecciona la opción FINALIZAR.6. El sistema guarda la información de los grupos expertos.
Flujo alternativo	Ninguno

A.4.4. Asignar alumnos a grupo experto

Código	CUS-4
Nombre	ASIGNAR ALUMNOS A GRUPO EXPERTO
Descripción	El caso de uso inicia cuando el docente selecciona la opción editar integrantes para un determinado grupo experto, luego agrega a los alumnos y cuando presiona la opción guardar, el caso de uso termina.
Actores	Docente
Precondiciones	El usuario de ser un Docente y estar logueado en el sistema. Deben existir alumnos registrados.
Postcondiciones	Ninguna
Flujo básico	<ol style="list-style-type: none">1. El docente selecciona la opción Editar integrantes en la lista de grupos expertos creados.2. El docente arrastra a los alumnos desde la lista de disponibles hacia la lista de alumnos para el grupo experto.3. El sistema valida que el número de alumnos seleccionados no exceda el máximo de integrantes permitidos para el grupo experto.4. El docente presiona la opción Guardar.5. El sistema guarda la lista de integrantes del grupo experto.
Flujo alternativo	Ninguno

A.4.5. Crear sesión jigsaw

Código	CUS-5
Nombre	CREAR SESIÓN JIGSAW
Descripción	El caso de uso inicia cuando el usuario accede al sistema y elige la opción de crear una nueva sesión de clase jigsaw; luego ingresa los datos necesarios de la sesión de clase y el caso de uso termina cuando la nueva sesión es creada satisfactoriamente en el sistema
Actores	Docente
Precondiciones	El usuario debe ser un Docente y estar logueado en el sistema. Los grupos expertos deben haber sido creados
Postcondiciones	El docente verá en el listado de clases la nueva sesión creada
Flujo básico	<ol style="list-style-type: none">1. El docente abre un formulario para crear una nueva sesión de clase jigsaw.2. El sistema solicita los campos requeridos para crear la nueva sesión jigsaw.3. El docente rellena los campos Curso y Tema.4. El docente asigna la duración en minutos para la reunión de expertos e indica la fecha y hora de inicio.5. El docente asigna la duración en minutos para la reunión jigsaw e indica la fecha y hora de inicio.6. El docente indica el número total de grupos expertos a incluir en la sesión.7. El docente asigna un problema a cada Grupo Experto.8. El docente selecciona la opción Guardar.9. El sistema graba la información de la nueva sesión jigsaw.
Flujo alternativo	Ninguno

A.4.6. Modificar sesión jigsaw

Código	CUS-6
Nombre	MODIFICAR SESIÓN JIGSAW
Descripción	El caso de uso inicia cuando el usuario accede al sistema y selecciona la opción modificar sesión jigsaw. El usuario realiza los cambios que requiera y luego finaliza el caso de uso
Actores	Docente
Precondiciones	El usuario debe ser un Docente y debe estar logueado en el sistema. Debe existir una sesión jigsaw en el sistema
Postcondiciones	Ninguna
Flujo básico	<ol style="list-style-type: none">1. El docente selecciona una sesión jigsaw creada y luego elige la opción editar.2. El sistema muestra la información de la sesión jigsaw seleccionada.3. El docente puede cambiar los problemas asignados a los grupos expertos así como también el tiempo y fecha de inicio de las reuniones de expertos y reuniones jigsaw.4. El docente selecciona la opción FINALIZAR.5. El sistema guarda los cambios realizados a la sesión jigsaw
Flujo alternativo	Ninguno

A.4.7. Crear evaluación

Código	CUS-7
Nombre	CREAR EVALUACIÓN
Descripción	El caso de uso inicia cuando el usuario accede al sistema y elige la opción Crear Evaluación. Luego el usuario formula las preguntas o problemas y agrega el puntaje de cada una de ellas. El caso de uso termina cuando el usuario selecciona guardar la nueva evaluación
Actores	Docente
Precondiciones	El usuario debe ser un docente y debe estar logueado en el sistema. Deben existir problemas creados
Postcondiciones	El docente podrá ver una nueva evaluación en su listado de evaluaciones
Flujo básico	<ol style="list-style-type: none">1. El docente selecciona la opción CREAR EVALUACIÓN.2. El sistema muestra el formulario con los campos requeridos para crear una evaluación: fecha de inicio, hora de inicio, duración en minutos, número de problemas, puntaje por problema.3. El sistema solicita definir los problemas que serán incluidos en la evaluación.4. El docente selecciona los problemas que serán usados para la evaluación.5. El docente selecciona la opción FINALIZAR.6. El sistema guarda la nueva evaluación
Flujo alternativo	Ninguno

A.4.8. Atender consulta de alumno

Código	CUS-8
Nombre	ATENDER CONSULTA DE ALUMNO
Descripción	El caso de uso inicia cuando se notifica al docente que uno o más alumnos le quieren consultar algo. El docente activa la ventana de conversación y responde a la consulta del alumno. El caso de uso finaliza cuando se cierra la ventana de conversación entre docente y alumno.
Actores	Docente
Precondiciones	El docente debe estar logueado en el sistema y algún alumno debe haber realizado una consulta al docente
Postcondiciones	Ninguna
Flujo básico	<ol style="list-style-type: none">1. El sistema notifica al docente que un alumno requiere consultarle sobre algún tema o asunto de la sesión jigsaw.2. El docente abre la ventana de conversación con el alumno.3. El docente responde a la consulta del alumno.4. El docente cierra la ventana de conversación.
Flujo alternativo	Ninguno

A.4.9. Participar en reunión de expertos

Código	CUS-9
Nombre	PARTICIPAR EN REUNIÓN DE EXPERTOS
Descripción	El caso de uso inicia cuando el alumno selecciona la opción participar en reunión de expertos. El alumno dispondrá de un tiempo fijado por el docente para debatir y desarrollar el problema planteado de forma colaborativa con los demás miembros del grupo experto. El caso de uso finaliza cuando culmina el tiempo asignado para la reunión de expertos.
Actores	Alumno
Precondiciones	Debe existir una sesión jigsaw creada por el docente y el alumno debe estar logueado en el sistema y ser parte de un grupo experto
Postcondiciones	Ninguna
Flujo básico	<ol style="list-style-type: none">1. El alumno selecciona en una de las sesiones jigsaw disponibles la opción Reunión de Expertos.2. El sistema mostrará al alumno la información del problema asignado(Tema, Enunciado, Tiempo disponible), un editor de código fuente para el trabajo colaborativo y una lista con los miembros presentes en la reunión.3. El alumno resuelve el problema en el editor.4. El alumno usa el chat grupal para consultar a los demás miembros de su grupo.5. El sistema informará el término de la reunión con 2 minutos de anticipación.6. El sistema finalizará la reunión de expertos.
Flujo alternativo	Ninguno

A.4.10. Resolver problema

Código	CUS-10
Nombre	RESOLVER PROBLEMA
Descripción	El caso de uso inicia cuando el alumno ingresa a una reunión en la cual desarrollará el problema planteado discutiendo con los demás miembros del grupo y elaborando el código fuente de la solución
Actores	Alumno
Precondiciones	El alumno debe haber ingresado a una reunión de expertos o reunión jigsaw
Postcondiciones	Ninguna
Flujo básico	<ol style="list-style-type: none">1. El alumno escribe la solución al problema en el editor de código fuente.2. El alumno ingresa los datos de prueba en el panel correspondiente.3. El alumno selecciona la opción RUN para compilar su código fuente y ver los resultados de su solución
Flujo alternativo	Ninguno

A.4.11. Consultar al grupo

Código	CUS-11
Nombre	CONSULTAR AL GRUPO
Descripción	El caso de uso inicia cuando el alumno abre la ventana de chat grupal para hacer alguna consulta
Actores	Alumno
Precondiciones	El alumno debe haber ingresado a una reunión de expertos o reunión jigsaw
Postcondiciones	Ninguna
Flujo básico	<ol style="list-style-type: none">1. El alumno abre la ventana de conversación.2. El alumno escribe y envía su consulta.3. El sistema envía la consulta a los demás miembros del grupo.4. El alumno cierra la ventana de conversación.
Flujo alternativo	

A.4.12. Participar en reunión jigsaw

Código	CUS-12
Nombre	PARTICIPAR EN REUNIÓN JIGSAW
Descripción	El caso de uso inicia cuando el alumno selecciona la opción participar en reunión jigsaw. El alumno dispondrá de un tiempo fijado por el docente para debatir y desarrollar el problema planteado de forma colaborativa con los demás miembros del grupo experto. El caso de uso finaliza cuando culmina el tiempo asignado para la reunión de expertos.
Actores	Alumno
Precondiciones	Debe existir una sesión jigsaw creada por el docente y el alumno debe estar logueado en el sistema y ser parte de un grupo jigsaw.
Postcondiciones	Ninguna
Flujo básico	<ol style="list-style-type: none">1. El alumno selecciona en una de las sesiones jigsaw disponibles la opción Reunión Jigsaw.2. El sistema mostrará al alumno la información de los problemas a desarrollar(Tema, Enunciado, Tiempo disponible), un editor de código fuente para el trabajo colaborativo y una lista con los miembros presentes en la reunión.3. El alumno resuelve el problema que le tocó en su grupo experto.4. El alumno usa el chat grupal para consultar y responder a las interrogantes de los demás miembros de su grupo.5. El sistema informará el término de la reunión con 3 minutos de anticipación.6. El sistema finalizará la reunión jigsaw.
Flujo alternativo	Ninguno

A.4.13. Consultar al docente

Código	CUS-13
Nombre	CONSULTAR AL DOCENTE
Descripción	El caso de uso inicia cuando el alumno abre la ventana de consulta al docente
Actores	Alumno
Precondiciones	El alumno debe haber ingresado a una reunión de expertos o reunión jigsaw
Postcondiciones	Ninguna
Flujo básico	<ol style="list-style-type: none">1. El alumno abre la ventana de conversación.2. El alumno escribe y envía su consulta.3. El sistema envía la consulta al docente.4. El alumno cierra la ventana de conversación.
Flujo alternativo	Ninguno

A.4.14. Rendir evaluación

Código	CUS-14
Nombre	RENDIR EVALUACIÓN
Descripción	El caso de uso inicia cuando el alumno selecciona la opción rendir evaluación y el sistema le muestra el examen creado por el docente. El alumno desarrolla las preguntas planteadas en el tiempo asignado y cuando selecciona la opción FINALIZAR, el caso de uso termina
Actores	Alumno
Precondiciones	El alumno debe estar logueado en el sistema y debe tener asignado algún examen
Postcondiciones	Ninguna
Flujo básico	<ol style="list-style-type: none">1. El alumno selecciona la opción RENDIR EVALUACIÓN.2. El sistema muestra la información del examen(Número de preguntas, puntaje por pregunta, tiempo total para el examen, enunciados de preguntas).3. El alumno responde a las preguntas.4. El alumno selecciona la opción finalizar.5. El sistema guarda las respuestas y culmina la evaluación
Flujo alternativo	Ninguno

Apéndice B

Especificaciones suplementarias

B.1. Introducción

El propósito de este documento es definir los requerimientos del Sistema web de aprendizaje colaborativo que no pueden ser capturados vía especificación de casos de uso. Estos requerimientos que en su mayoría corresponden a requerimientos no funcionales serán capturados y clasificados según la norma ISO-9126 que define atributos de calidad para productos software.

B.2. Funcionalidad

- El sistema no debe tener errores en la asignación de tiempos para cada reunión.
- El sistema debe validar correctamente la asignación de alumnos a los grupos expertos.
- El sistema debe limitar el acceso sólo a usuarios debidamente registrados.

B.3. Fiabilidad o confiabilidad

- El sistema debe funcionar las 24 horas del día.
- El sistema debe soportar una concurrencia de hasta al menos 50 usuarios.

- El sistema debe capturar cualquier error y notificarlo sin tener que dejar de funcionar.

B.4. Usabilidad

- El sistema debe ser intuitivo para el usuario.
- El sistema debe permitir al usuario que éste aprenda cómo usarlo sin ninguna ayuda adicional.
- El sistema debe ser amigable para el usuario en cuanto a interfaz (colores y tonos claros, letras claras y legibles).

B.5. Eficiencia o Performance

- El tiempo de respuesta del sistema no debe exceder de 1 segundo.

B.6. Mantenibilidad

- El sistema web debe permitir realizar actividades correctivas o de actualización sin afectar el correcto funcionamiento de sus distintos módulos.

B.7. Portabilidad

- El sistema debe funcionar en los navegadores Chrome y Mozilla Firefox en sus últimas versiones.
- El sistema debe ser multiplataforma.

Apéndice C

Documento de Arquitectura de Software

C.1. Introducción

C.1.1. Propósito

El presente documento de arquitectura de software tiene como finalidad describir el sistema web Jigsaw Coding desde diferentes perspectivas, las cuales serán de gran utilidad para el desarrollo de dicho sistema. La presentación de la arquitectura del sistema será presentada haciendo uso del modelo 4+1.

C.1.2. Alcance

Este documento arquitectónico aplica para describir todas las funcionalidades del sistema web a implementar. Así mismo, en este artefacto se encontrará de forma clara las 5 vistas que se describen en el modelo 4+1: Vista lógica, Vista de procesos, Vista de despliegue, Vista física y Vista de escenarios.

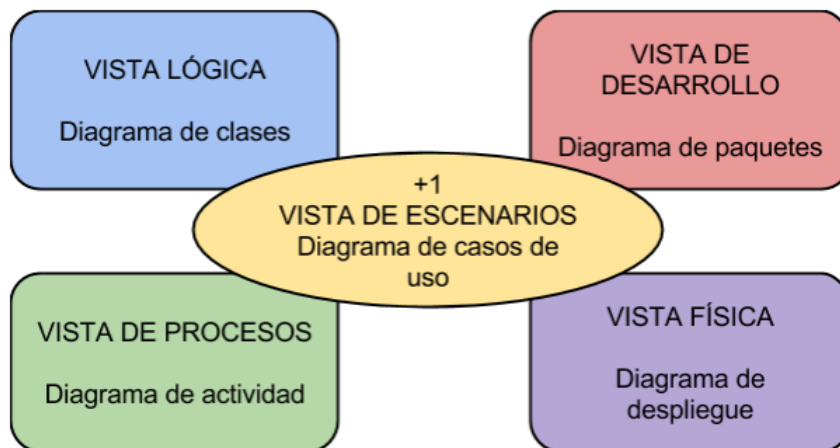


Figura C.1: Modelo de arquitectura 4+1

C.2. Vista lógica

C.3. Vista de procesos

C.4. Vista de desarrollo

La vista de despliegue 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 en paquetes y las dependencias que habrá entre ellos.

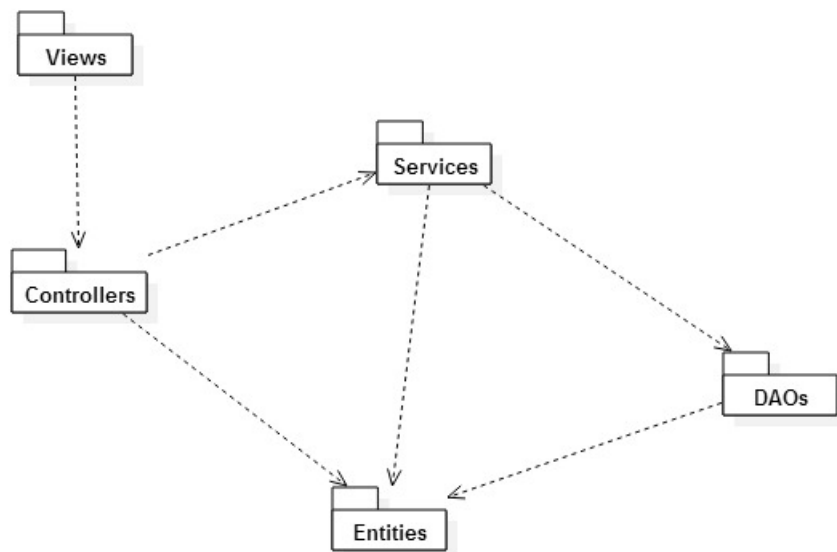


Figura C.2: Diagrama de Paquetes

C.5. Vista física

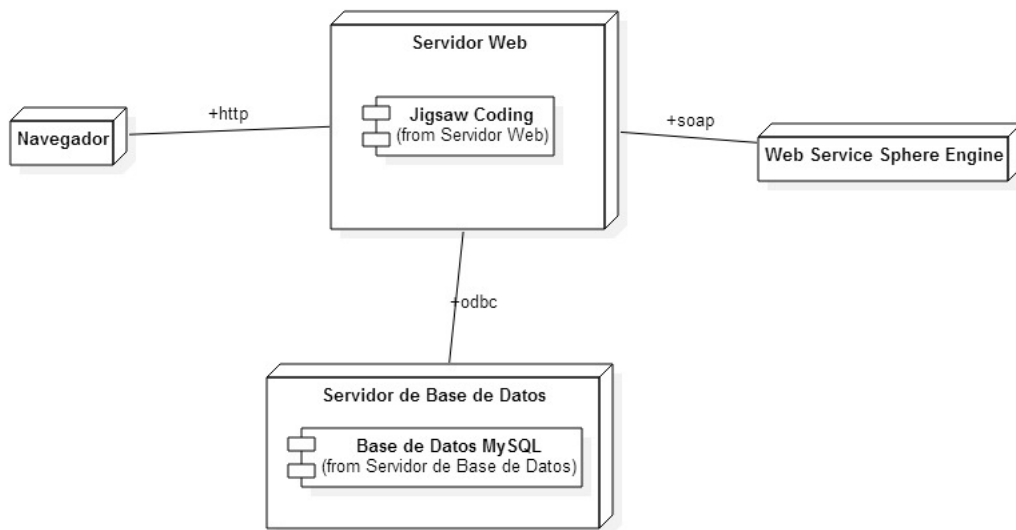


Figura C.3: Diagrama de Despliegue

C.6. Vista de escenarios

Referencias

- Aronson, E., Blaney, N., Stephin, C., Sikes, J., & Snapp, M. (1978). *The jigsaw classroom*. SAGE Publications.
- Azizinezhad, M., Hashemi, M., & Darvishi, S. (2013, octubre). Application of cooperative learning in EFL classes to enhance the students' language learning. *Procedia - Social and Behavioral Sciences*, 93, 138–141. Descargado 2014-04-12, de <http://www.sciencedirect.com/science/article/pii/S1877042813032692>
- Barkley, E. F., Cross, K. P., & Major, C. H. (2012). *Collaborative learning techniques: A handbook for college faculty*. John Wiley & Sons.
- Beck, K. (2000). *Extreme programming explained: Embrace change*. Addison-Wesley Professional.
- Beck, L. L., & Chizhik, A. W. (2008). An experimental study of cooperative learning in CS1. En *Proceedings of the 39th SIGCSE technical symposium on computer science education* (p. 205–209). New York, NY, USA: ACM. Descargado 2014-04-12, de <http://doi.acm.org/10.1145/1352135.1352208>
- 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/>
- Bruffee, K. A. (1984, noviembre). Collaborative learning and the “Conversation of mankind”. *College English*, 46(7), 635–652. Descargado 2014-04-17, de <http://www.jstor.org/discover/10.2307/376924?uid=3738800&uid=2&uid=4&sid=21103674393401>
- Buhr, G. T., Heflin, M. T., White, H. K., & 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>
- Cheng, D., Zhu, W., Li, D., & Zhou, Z. (2013). A new collaborative sketching method on web browser. En *Proceedings of the 11th asia pacific conference on computer human interaction* (p. 286–290). New York, NY,

- USA: ACM. Descargado 2014-05-03, de <http://doi.acm.org/10.1145/2525194.2525259>
- 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>
- CodeBunk. (2014). *CodeBunk: Online Collaborative Editor and Compiler*. <http://codebunk.com/>.
- Gerlach, J. M. (1994). Is this collaboration? En K. Bosworth & S. J. Hamilton (Eds.), *Collaborative learning: Underlying processes and effective techniques, new directions for teaching and learning* (pp. 5–14). San Francisco; USA: Jossey - Bass Publishing.
- Golub, J., y cols. (1988). *Focus on collaborative learning. classroom practices in teaching english, 1988*. ERIC.
- Google. (2014). *Google docs*. <http://docs.google.com/>.
- Han, J., & 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>
- Holzner, S. (2006). *Ajax for dummies*. Wiley Publishing.
- IBM. (2014). *Rational unified process. best practices for software development teams*. Descargado 2014-08-31, de http://www.ibm.com/developerworks/rational/library/content/03July/1000/1251/1251_bestpractices_TP026B.pdf
- Ideone. (2014). *Ideone*. <http://ideone.com/>.
- Johnson, D., & Johnson, R. (1989). *Cooperation and competition theory and research*. Edina, Minnesota; USA: Interaction Book Co.publishing.
- Johnson, D., Johnson, R., & Holubec, E. (1984). *Cooperation in the classroom* (Edina, Ed.). Minnesota; USA: Interaction Book Co. publishing.
- Kilic, D. (2008). The effect of the jigsaw technique on learning the concepts of the principles and methods of teaching. *World Applied Sciences Journal*, 4(1), 109–114.
- Knobelsdorf, M., Kreitz, C., & 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>
- Laal, M., & Laal, M. (2012). Collaborative learning: what is it? *Procedia - Social and Behavioral Sciences*, 31, 491–495. Descargado 2014-04-16, de <http://www.sciencedirect.com/science/article/pii/S1877042811030217>
- Lang, C., McKay, J., & 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>
- Lengstorf, J., & Leggetter, P. (2013). *Realtime web apps: HTML5 WebSocket, pusher, and the web's next big thing*. Apress.
- 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>
- Macgregor, J. (1990, junio). Collaborative learning: Shared inquiry as a process of reform. *New Directions for Teaching and Learning*, 1990(42), 19–30. Descargado 2014-05-01, de <http://onlinelibrary.wiley.com/doi/10.1002/tl.37219904204/abstract>
- Maftai, G., & Maftai, M. (2011). The strengthen knowledge of atomic physics using the “mosaic” method (the jigsaw method). *Procedia - Social and Behavioral Sciences*, 15, 1605–1610. Descargado 2014-04-15, de <http://www.sciencedirect.com/science/article/pii/S1877042811005179>
- Martinez, A., & 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., Guzdia, 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>
- McDowell, C., Werner, L., Bullock, H., & Fernald, J. (2002). The effects of pair-programming on performance in an introductory programming course. En *Proceedings of the 33rd SIGCSE technical symposium on computer science education* (p. 38–42). New York, NY, USA: ACM. Descargado 2014-04-16, de <http://doi.acm.org/10.1145/563340.563353>

- Miller, R. C., Zhang, H., Gilbert, E., & Gerber, E. (2014). Pair research: Matching people for collaboration, learning, and productivity. En *Proceedings of the 17th ACM conference on computer supported cooperative work; social computing* (p. 1043–1048). New York, NY, USA: ACM. Descargado 2014-05-02, de <http://doi.acm.org/10.1145/2531602.2531703>
- Nikula, U., Sajaniemi, J., Tedre, M., & 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>
- Panitz, T. (1999). Benefits of cooperative learning in relation to student motivation. En M. Theall (Ed.), *Motivation from within: Approaches for encouraging faculty and students to excel, new directions for teaching and learning*. San Francisco, CA; USA: Josey-Bass publishing.
- Persico D., S. L., Pozzi F. (2008). Fostering collaboration in CSDL. *Encyclopedia of Information and Communication Technology*.
- Pinzás, C., & Yatsen, G. (2013, noviembre). Desarrollo de un sistema web para la enseñanza de casos de uso empleando la técnica de aprendizaje cooperativo de rompecabezas. Descargado 2014-04-06, de <http://tesis.pucp.edu.pe/repositorio//handle/123456789/4929>
- Rational Method Composer. (2014). *RUP for small projects*. Descargado 2014-09, de http://cgrw01.cgr.go.cr/rup/RUP.es/SmallProjects/index.htm#core.base_rup/guidances/supportingmaterials/welcome_2BC5187F.html
- Salao Bravo, J. R. (2010). *Estudio de las técnicas de inteligencia artificial mediante el apoyo de un software educativo* (Thesis). Descargado 2014-04-07, de <http://dspace.esepoch.edu.ec/handle/123456789/90>
- Servicio de innovación educativa - Universidad Politécnica de Madrid. (2008). Aprendizaje cooperativo.
- Sheard, J., & 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>
- Smith, B. L., & MacGregor, J. T. (1992). What is collaborative learning? En A. Goodsell, M. Maher, V. Tinto, B. L. Smith, & M. J. T. (Eds.), *Collaborative learning: A sourcebook for higher education*. Pennsylvania State University, USA: National center on postsecondary teaching, learning, and assessment publishing.
- 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., & 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>
- Wang, W. (2002). *Computer-supported virtual collaborative learning and assessment framework for distributed learning environment* (Thesis, Massachusetts Institute of Technology). Descargado 2014-04-05, de <http://dspace.mit.edu/handle/1721.1/84815> (Thesis (S.M.)—Massachusetts Institute of Technology, Dept. of Civil and Environmental Engineering, 2002.)
- Williams, L., Kessler, R. R., Cunningham, W., & Jeffries, R. (2000, julio). Strengthening the case for pair programming. *IEEE Softw.*, 17(4), 19–25. Descargado 2014-05-03, de <http://dx.doi.org/10.1109/52.854064>
- Williams, L. A. (2000). *The collaborative software process* (Tesis Doctoral no publicada). Citeseer.
- Winter, E. C., & McGhie-Richmond, D. (2005). Using computer conferencing and case studies to enable collaboration between expert and novice teachers. *Journal of Computer Assisted Learning*, 21(2), 118–129. Descargado de <http://dx.doi.org/10.1111/j.1365-2729.2005.00119.x>