

CHILDPROGRAMMING-C: EXTENDIENDO CHILDPROGRAMMING DESDE LA PERSPECTIVA DE LA INGENIERÍA DE LA COLABORACIÓN



Universidad
del Cauca

ANA MARÍA CHIMUNJA GONZÁLEZ

ANEXOS

Director: Ph.D. César Alberto Collazos Ordoñez

Co-Director: Ph D. Julio Ariel Hurtado Alegría

**Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Maestría en Computación
Grupo de Investigación IDIS
Áreas de Investigación: Ingeniería de la Colaboración
Popayán, abril de 2018**

Contenido

Lista de tablas.....	4
Lista de figuras	6
1. ANEXO A - PATRONES Y THINKLETS DE COLABORACIÓN	7
1.1 Patrones de colaboración	7
1.2 Clasificación de thinklets según el patrón de colaboración	8
1.2.1 Patrón generación.....	8
1.2.2 Patrón reducción	10
1.2.3 Patrón clarificación	11
1.2.4 Patrón organización.....	12
1.2.5 Patrón evaluación.....	15
1.2.6 Patrón construcción de consenso	16
1.3 Ampliación de descripción de thinkles principales.....	18
2. ANEXO B: ESTUDIOS DE CASO PRELIMINAR	25
2.1 Diseño de estudio de caso preliminar 1	25
3. ANEXO C. ESTUDIO DE CASO: APLICACIÓN DEL MODELO CHILDPROGRAMMING -G EN EL DESARROLLO DE UNA APLICACIÓN SOFTWARE	35
3.1 Diseño estudio de caso 2.....	36
3.2 Ejecución del estudio de caso 2.....	37
3.3 Resultados del estudio de caso	39
4. ANEXO D – EVALUACIÓN DE CRITERIOS PARA IDENTIFICAR SI UNA ACTIVIDAD REALIZADA POR NIÑOS REQUIERE TRABAJO COLABORATIVO EN SU EJECUCIÓN.....	41
5. ANEXO E – SENSIBILIZANDO A LOS NIÑOS SOBRE LA IMPORTANCIA DEL TRABAJO COLABORATIVO ANTES DE INICIAR UN PROCESO DE DESARROLLO DE SOFTWARE	47
5.1 Desarrollo actividad 1.....	47
5.2 Desarrollo actividad 2.....	50
6. ANEXO F. ANALIZANDO LA CAPACIDAD DE APRENDIZAJE EN EL MODELO CHILDPROGRAMMING, COMO UNA SUBCARACTERÍSTICA DE LA USABILIDAD.....	53
6.1 ISO/IEC 25000 SQuaRE (System and Software Quality Requirements and Evaluation)	53

6.1.1	Usabilidad	54
6.1.1.1.1	Capacidad de aprendizaje	55
6.1.2	Definición de las métricas relacionadas con atributos de la facilidad de aprendizaje	56
7.	ANEXO G. TÉCNICAS DE APRENDIZAJE COLABORATIVO	65
8.	ANEXO H. DESCRIPCIÓN DEL MODELO CONCEPTUAL CHILDPROGRAMMING71	

Lista de tablas

Tabla 1. Patrones definidos en la Ingeniería de la Colaboración [2].....	7
Tabla 2. Patrón generación y thinklets relacionados [4][5].	10
Tabla 3. Patrón reducción y thinklets BroomWagon [4][5].....	11
Tabla 4. Patrón clarificación y thinklet Pin the tail on the donkey [4][5].....	12
Tabla 5. Patrón organización y thinklet concentration [4][5].	13
Tabla 6. Patrón organización y thinklet RichRelations [4][5].....	14
Tabla 7. Patrón organización y thinklet leafhopper [4][5].	14
Tabla 8. Patrón evaluación y thinklet StrawPoll [4][5].....	15
Tabla 9. Patrón Construcción de consenso y Crowbar [4][5].....	17
Tabla 10. Patrón Construcción de consenso y thinklet MoodRing [4][5].....	17
Tabla 11. Patrón Construcción de consenso y PointCounterPoint [4][5].....	18
Tabla 12. Thinklet onepage	19
Tabla 13. Thinklet Pin the Tail on the Donkey.	21
Tabla 14. Thinklet Concentration.	23
Tabla 15. Thinklet MoodRing	24
Tabla 16. Diseño del estudio de caso preliminar.....	26
Tabla 17. Descripción Actividad No. 1 tomada y adaptada de [6]	28
Tabla 18. Análisis de la observación respecto a lineamientos.....	31
Tabla 19. Diseño de estudio de caso 2	37
Tabla 20. Descripción Actividad No. 1	38
Tabla 21. Caracterización de expertos.	45
Tabla 22. Resultados de valoración de expertos.....	46
Tabla 23. Actividad 1 – análisis de videos.....	49
Tabla 24. Planeando estrategia - 1.	50
Tabla 25. Planeando estrategia – 2.	51
Tabla 26. Subcaracterísticas de Usabilidad ISO/IEC 25010 [8] [13].....	55
Tabla 27. Atributos y métricas relacionados con la capacidad de aprendizaje en aplicaciones software [6][7].....	58

Tabla 28. Atributos y métricas relacionados con la capacidad de aprendizaje en childprogramming	58
Tabla 29. Técnicas colaborativas de discusión	66
Tabla 30. Técnicas de enseñanza recíproca entre compañeros	68
Tabla 30. Técnicas colaborativas para resolver problemas.....	69
Tabla 32. Técnicas colaborativas de organización gráfica.	70
Tabla 33. Descripción de las clases del modelo Chilprogramming.....	76

Lista de figuras

Figura 1. Patrón generación y thinklets	8
Figura 2. Patrón reducción y thinklets	10
Figura 3. Patrón clarificación y thinklets	11
Figura 4. Patrón organización y thinklets	12
Figura 5. Patrón evaluación y thinklets	15
Figura 6. Patrón Construcción de consenso y thinklets.....	16
Figura 7. Plegable entregado. Tomado de [6]	29
Figura 8. Evidencia fotográfica - Actividad 1	30
Figura 8. Evidencia fotográfica - Actividad 2	30
Figura 10. Encuesta aplicada para la gamificación	32
Figura 11. Evidencia fotográfica análisis	38
Figura 12. Desarrollo de aplicaciones caso de estudio 2.	39
Figura 13. Niños participantes en presentación de videos.	49
Figura 14. Participación de los niños en actividades lúdicas.	50
Figura 15. Participación de los niños en actividades lúdicas.	51
Figura 16. Características del producto software[8]	54
Figura 12. Clasificación de técnicas colaborativas	65
Figura 18. Modelo conceptual del Childprogrammig-C.....	71

ANEXO A - PATRONES Y THINKLETS DE COLABORACIÓN

1.1 Patrones de colaboración

Los *patrones de colaboración* se definen en términos del “movimiento del grupo desde su estado inicial hasta su estado final” [1]. La siguiente tabla presenta los *patrones de colaboración* definidos en la Ingeniería de la Colaboración [1] y utilizados en el siguiente trabajo:

PATRONES DE COLABORACIÓN	DESCRIPCIÓN
Patrón generación	Es un patrón a partir del cual el grupo crea contenido. Consiste en pasar de tener pocos a muchos conceptos que son compartidos por el grupo.
Patrón reducción	El objetivo de este patrón es mantener sólo la información que cumple con un determinado criterio o criterios. Consiste en pasar de tener muchos conceptos a unos pocos que el grupo considere que requieren mayor atención.
Patrón Clarificación	El objetivo de este patrón es lograr el entendimiento común de conceptos manejados por el grupo. Consiste en pasar de tener un menor a un mayor conocimiento compartido de los conceptos, las palabras y frases usadas para expresarlos.
Patrón organización	Consiste en pasar de tener un menor a un mayor conocimiento de las relaciones entre los conceptos que el grupo esté considerando.
Patrón evaluación	Consiste en pasar de un menor a un mayor conocimiento del valor relativo de los conceptos bajo consideración. Este patrón tiene como efectos apoyar a la toma de decisiones y a la comunicación del grupo.
Patrón construcción de consenso	Moverse de tener pocos a muchos miembros del grupo quienes estarán dispuestos a comprometerse para un objetivo.

Tabla 1. Patrones definidos en la Ingeniería de la Colaboración [2]

1.2 Clasificación de thinklets según el patrón de colaboración

Los Thinklets de colaboración pueden clasificarse según al patrón de colaboración al que pertenecen [3]. En las siguientes figuras y tablas se presentan la clasificación y descripción de acuerdo a los estudios [4][5]:

1.2.1 Patrón generación

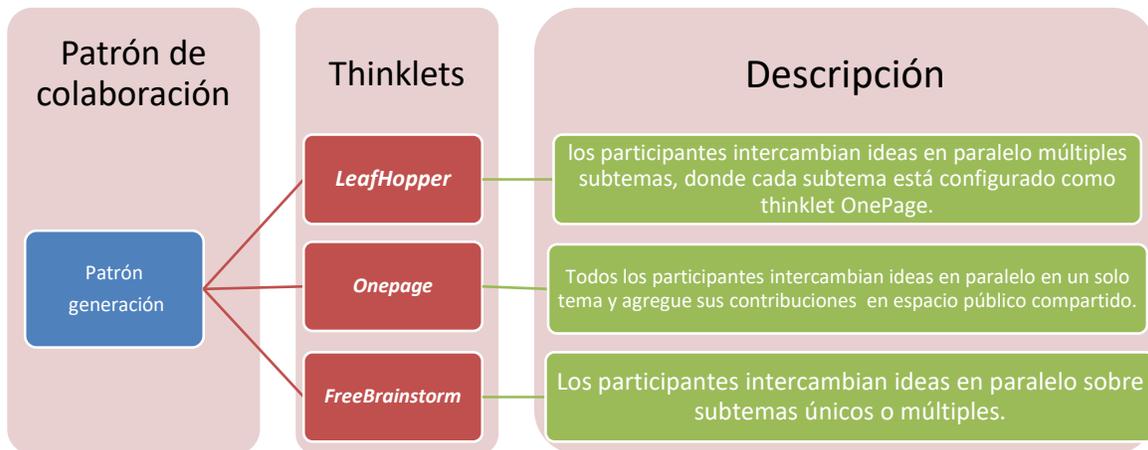


Figura 1. Patrón generación y thinklets

Patrón de colaboración: generación	
Subpatrones asociados:	<p>Reunir: coleccionar y compartir conceptos entre los miembros del grupo.</p> <p>Crear: Producir y compartir nuevas ideas que no fueron previamente conocidas por los miembros del grupo.</p> <p>Elaborar: Adicionar detalles a los conceptos que ya fueron compartidos por los miembros del grupo.</p> <ul style="list-style-type: none"> • Descomponer: caracterizar un concepto en términos de sus componentes o subcomponentes. • Expandir: adicionar detalles para una explicación más completa o para describir un concepto.

<p>Thinklet:</p>	<p>OnePage:</p> <p>Descripción [3]: Todos los participantes intercambian ideas en Paralelo en una un solo tema y agregue sus contribuciones a una sola, espacio público compartido.</p> <ol style="list-style-type: none"> 1. Permita que los participantes agreguen en paralelo cualquier número de contribuciones a la lista. 2. Permita que los participantes agreguen solo contribuciones que coincide con la especificación de contribución. 3. Asegúrese de que los participantes lean las contribuciones de otros para la inspiración. 	
	<p>Entradas:</p> <p>La pregunta de lluvia de ideas o sugerencia.</p>	<p>Salidas:</p> <p>Conjunto de comentarios en respuesta a la pregunta de lluvia de ideas o sugerencia.</p>
<p>Thinklet:</p>	<p>LeafHopper:</p> <p>Descripción [3]: los participantes intercambian ideas en paralelo múltiples subtemas, donde cada subtema está configurado como thinklet OnePage. Los participantes eligen el (los) tema (s) secundario (s) a lo cual contribuirán.</p> <ol style="list-style-type: none"> 1. Permita que los participantes agreguen en paralelo cualquier número de contribuciones a cualquier categoría. 2. Permita que los participantes agreguen solo contribuciones que son relevantes para las categorías en las que están participando. 3. Permita que los participantes agreguen solo contribuciones que coincide con la especificación de contribución. 4. Permita que los participantes cambien el enfoque de una categoría a categoría como lo dictan el interés y la inspiración. 5. Asegúrese de que los participantes lean las contribuciones de otros para la inspiración. 	

	Entradas: Un listado de tareas que deben ser abordadas por el equipo.	Salidas: Un conjunto de observaciones organizadas por temas.
Thinklet:	FreeBrainstorm Descripción: los participantes intercambian ideas en paralelo sobre subtemas únicos o múltiples. Después de cada contribución, los participantes intercambian las hojas de contribuciones al azar. Los participantes solo pueden ver las contribuciones en la página que están trabajando, después del intercambio pueden ver las contribuciones en la página siguiente que se recibe. 1. Permita que los participantes agreguen una contribución a cada página que reciben en paralelo. 2. Asegúrese de que los participantes intercambien páginas al azar después de cada contribución. 3. Permita que los participantes agreguen solo contribuciones que coincide con la especificación de contribución. 4. Asegúrese de que los participantes lean las contribuciones de otros para la inspiración.	
	Entradas: Un listado de temas para abordar.	Salidas: Conjunto de contribuciones dadas por los participantes.

Tabla 2. Patrón generación y thinklets relacionados [4] [5].

1.2.2 Patrón reducción

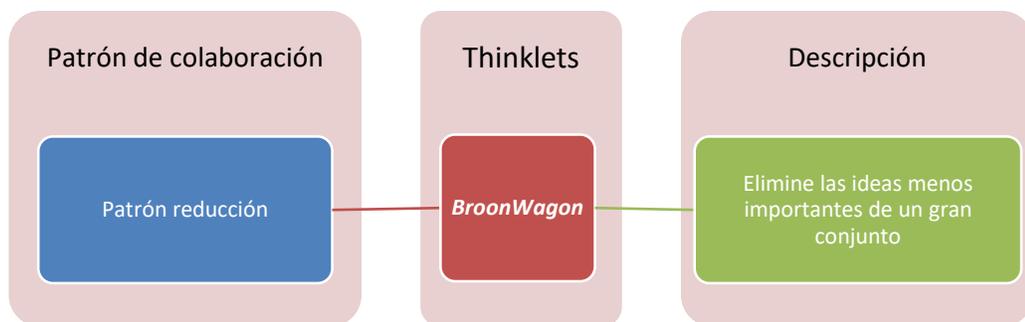


Figura 2. Patrón reducción y thinklets

Patrón de colaboración: Reducción		
Subpatrones asociados:	<p>Seleccionar: Escoger un subconjunto de conceptos existentes.</p> <p>Abstraer: Derivar conceptos más generales desde instancias específicas, en el conjunto existente.</p> <p>Resumir: Capturar la esencia de los conceptos, sin eliminar conceptos únicos.</p>	
Thinklets:	BroonWagon	
	<table border="1"> <tr> <td>Entradas: Un gran conjunto de elementos, por ejemplo, el resultado del libre intercambio de ideas</td> <td>Salidas: Un pequeño conjunto ordenado de elementos con los que está de acuerdo el equipo.</td> </tr> </table>	Entradas: Un gran conjunto de elementos, por ejemplo, el resultado del libre intercambio de ideas
Entradas: Un gran conjunto de elementos, por ejemplo, el resultado del libre intercambio de ideas	Salidas: Un pequeño conjunto ordenado de elementos con los que está de acuerdo el equipo.	

Tabla 3. Patrón reducción y thinklets BroonWagon [4] [5].

1.2.3 Patrón clarificación



Figura 3. Patrón clarificación y thinklets [5]

Patrón de colaboración: clarificación	
Subpatrones asociados:	Describir: proponer explicaciones y descripciones alternativas de un concepto.

Thinklet:	<i>Pin the tail on the donkey</i>	
	Descripción: hacer que un grupo identifique conceptos importantes que ameritan mayor deliberación	
	Entradas: Una gran cantidad de comentarios de los integrantes del equipo en reacción a las ideas, proposiciones, propuestas, etc.	Salidas: <ul style="list-style-type: none"> • comentarios claves para la actividad. • Entendimiento compartido sobre los comentarios claves.

Tabla 4. Patrón clarificación y thinklet Pin the tail on the donkey [4] [5].

1.2.4 Patrón organización

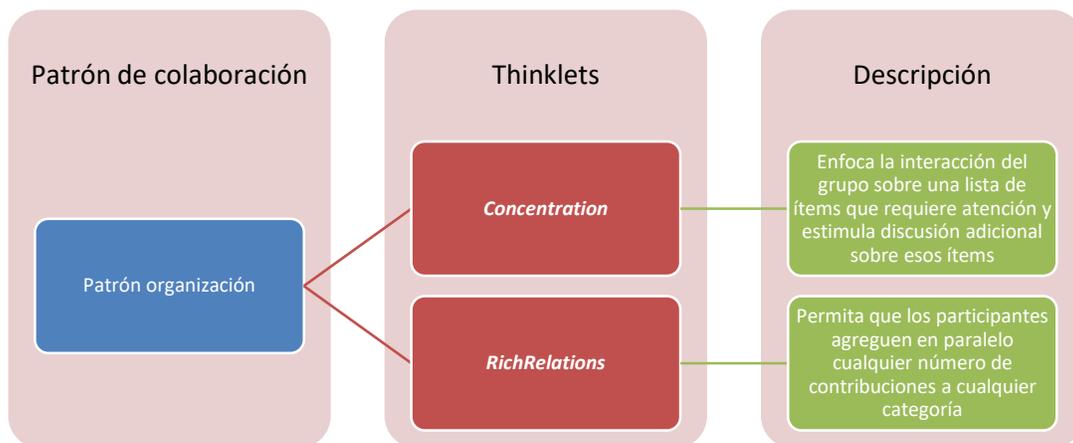


Figura 4. Patrón organización y thinklets [5]

Patrón de colaboración: organización	
Subpatrones asociados:	Estructurar: crear arreglos espaciales entre conceptos para representar sus relaciones conceptuales.

	Concentration	
Thinklets:	<p>En lugar de trabajar en paralelo, los participantes deciden conjuntamente qué acción se debe tomar; p.ej. organización conjunta, aclaración / reformulación conjunta, evaluación conjunta. Un participante se desempeña como director para el grupo, tomando la acción de acuerdo con los deseos del grupo. Si bien el trabajo paralelo puede ser más eficiente que el trabajo con el director, en algunos casos es más valioso garantizar la comprensión compartida y llegar a acuerdos mutuamente aceptables que ser más eficiente: eliminar la superposición entre las ideas para crear un conjunto único.</p>	
	<p>Entradas: Una lista aproximada de las ideas, que contenga solapamiento entre las ideas.</p>	<p>Salidas: Una lista limpia, es decir, una lista en la que las ideas duplicadas se han eliminado, ideas ambiguas han sido reformulado, y las ideas que abordan temas o conceptos similares se han combinado.</p>

Tabla 5. Patrón organización y thinklet concentration [4].

Patrón de colaboración: organización		
Subpatrones asociados:	Clasificar: ordenar conceptos en categorías.	
Thinklets:	RichRelations	
	<p>Descripción: Hacer que un grupo descubra categorías posibles en las que se pueden organizar varios conceptos existentes.</p>	
	<p>Entradas: Se realizan comentarios dentro de cada equipo de trabajo sobre la actividad y se escogen de</p>	<p>Salidas: Entrega de un conjunto de subtarefas organizadas</p>

	forma autónoma la forma de trabajar mediante una de lluvia de ideas.	dependientes o independientes de otras subtareas en la actividad.
--	--	---

Tabla 6. Patrón organización y thinklet RichRelations [4][6].

Patrón de colaboración: organización			
Subpatrones asociados:	Clasificar: ordenar conceptos en categorías.		
Thinklets:	<p style="text-align: center;">LeafHopper</p> <p>Descripción:</p> <ol style="list-style-type: none"> 1. Permita que los participantes agreguen en paralelo cualquier número de contribuciones a cualquier categoría. 2. Permita que los participantes agreguen solo contribuciones que son relevantes para las categorías en las que están metido. 3. Permita que los participantes agreguen solo contribuciones que coincide con la especificación de contribución. 4. Permita que los participantes cambien el enfoque de una categoría a categoría como lo dictan el interés y la inspiración. 5. Asegúrese de que los participantes lean las contribuciones de otros para la inspiración. <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%;">Entradas: Un listado de tareas que deben ser abordadas por el equipo.</td> <td style="width: 50%;">Salidas: Un conjunto de observaciones organizadas por temas.</td> </tr> </table>	Entradas: Un listado de tareas que deben ser abordadas por el equipo.	Salidas: Un conjunto de observaciones organizadas por temas.
Entradas: Un listado de tareas que deben ser abordadas por el equipo.	Salidas: Un conjunto de observaciones organizadas por temas.		

Tabla 7. Patrón organización y thinklet leafhopper [4][6].

1.2.5 Patrón evaluación

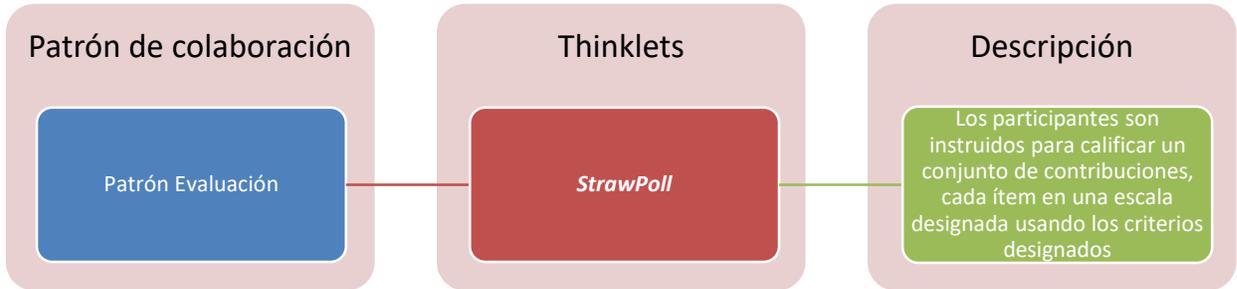


Figura 5. Patrón evaluación y thinklets

A continuación, se presenta la descripción más detallada de los patrones.

Patrón de colaboración: Evaluación		
Subpatrones asociados:	<p>Votar: evaluar la opinión del grupo, respecto a los conceptos.</p> <p>Categorizar: identificar un orden de preferencia entre los conceptos.</p> <p>Valorar: Especificar y detallar en el valor de los conceptos.</p>	
Thinklets:	<p style="text-align: center;">StrawPoll</p> <p>Descripción: El moderador publica una página de contribuciones no evaluadas. Los participantes son instruidos para calificar cada ítem en una escala designada usando los criterios designados. A los participantes se les dice que no están tomando una decisión, que solo tienen una idea de las opiniones del grupo para ayudar a enfocar la discusión posterior.</p>	
	<table border="1" style="width: 100%;"> <tr> <td style="width: 50%;">Entradas: Un conjunto de tareas para ser evaluadas.</td> <td style="width: 50%;">Salidas: <ul style="list-style-type: none"> • Un listado de las tareas evaluadas con su respectiva prioridad. </td> </tr> </table>	Entradas: Un conjunto de tareas para ser evaluadas.
Entradas: Un conjunto de tareas para ser evaluadas.	Salidas: <ul style="list-style-type: none"> • Un listado de las tareas evaluadas con su respectiva prioridad. 	

Tabla 8. Patrón evaluación y thinklet StrawPoll [4][6].

1.2.6 Patrón construcción de consenso

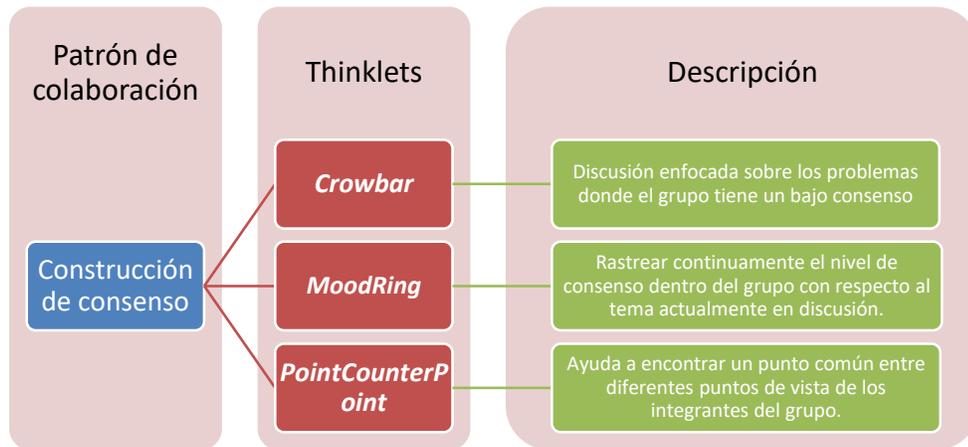


Figura 6. Patrón Construcción de consenso y thinklets

Patrón de colaboración: construcción de consenso			
Subpatrones asociados:	Medir: Evaluar el grado en que los participantes están dispuestos a comprometerse con una propuesta.		
Thinklets:	<p style="text-align: center;">Crowbar</p> <p>Descripción: discusión enfocada sobre los problemas donde el grupo tiene un bajo consenso. Después de una votación, el moderador llama la atención del grupo sobre los puntos con mayor desacuerdo. Los miembros del grupo discuten las razones por las cuales alguien podría darle una calificación baja a un artículo. La conversación resultante revela supuestos no cuestionados, información no compartida, conflictos de objetivos y otra información útil para avanzar hacia el consenso.</p> <table border="1" style="width: 100%;"> <tr> <td style="width: 50%;">Entradas: Entrega de puntos para evaluar.</td> <td style="width: 50%;">Salidas: <ul style="list-style-type: none"> Lista de priorización de las tareas y aplicación de las reglas en la actividad. </td> </tr> </table>	Entradas: Entrega de puntos para evaluar.	Salidas: <ul style="list-style-type: none"> Lista de priorización de las tareas y aplicación de las reglas en la actividad.
Entradas: Entrega de puntos para evaluar.	Salidas: <ul style="list-style-type: none"> Lista de priorización de las tareas y aplicación de las reglas en la actividad. 		

		<ul style="list-style-type: none"> • Comprensión de las reglas y socialización de diferencias de opiniones dentro del grupo.
--	--	---

Tabla 9. Patrón Construcción de consenso y Crowbar [4][6].

Patrón de colaboración: construcción de consenso			
Subpatrones asociados:	Medir: Evaluar el grado en que los participantes están dispuestos a comprometerse con una propuesta.		
Thinklets:	<p style="text-align: center;"><i>MoodRing</i></p> <p>Descripción: Para rastrear continuamente el nivel de consenso dentro del grupo con respecto al tema actualmente en discusión.</p> <table border="1" style="width: 100%;"> <tr> <td style="width: 50%;">Entradas: Entrega de guía de trabajo y de acuerdos o compromisos que los equipos deben adquirir en la actividad.</td> <td style="width: 50%;">Salidas: Una lista ordenada y consensuada de las tareas y normas a seguir en la actividad.</td> </tr> </table>	Entradas: Entrega de guía de trabajo y de acuerdos o compromisos que los equipos deben adquirir en la actividad.	Salidas: Una lista ordenada y consensuada de las tareas y normas a seguir en la actividad.
Entradas: Entrega de guía de trabajo y de acuerdos o compromisos que los equipos deben adquirir en la actividad.	Salidas: Una lista ordenada y consensuada de las tareas y normas a seguir en la actividad.		

Tabla 10. Patrón Construcción de consenso y thinklet MoodRing [4][6].

Patrón de colaboración: Construcción Consenso	
Subpatrones asociados:	<p>Diagnosticar: Buscar la comprensión de las causas de desacuerdos.</p> <p>Apoyar: Tratar de convencer a otros a aceptar y adoptar una posición.</p> <p>Resolver: Buscar la forma de superar las causas de desacuerdos.</p>

Thinklet:	PointCounterPoint	
	Descripción: Ayuda a encontrar un punto común entre diferentes puntos de vista de los integrantes del grupo.	
	Entradas: Un propuesta discutible.	Salidas: <ul style="list-style-type: none"> • Un conjunto de argumentos, y soluciones en torno a una propuesta discutible. • Movimiento de las posiciones extremas hacia un término medio.

Tabla 11. Patrón Construcción de consenso y PointCounterPoint [4][6].

1.3 Ampliación de descripción de thinkles principales

A continuación se presentan una descripción más amplia de los thinklets OnePage, Pin the Tail on the Donkey, Concentration, MoodRing ofrecida en [7].

OnePage	
Escoger este Thinklet...	<ul style="list-style-type: none"> - Para generar unos pocos comentarios sobre un tópico. - Cuando 5 o menos personas generan ideas juntas. - Cuando 6 o más personas generan ideas por menos de 10 minutos. - Cuando no es probable tener muchos comentarios sobre el asunto bajo discusión.
No escoger este Thinklet.	<ul style="list-style-type: none"> - Cuando usted espere más de 80 comentarios porque esto puede causar información sobrecargada. Considerar <i>FreeBrainstorm</i>. - Cuando 6 o más personas generan ideas. - Cuando el equipo debe dirigir más de un asunto a la vez.

Información general	Los miembros del equipo contribuyen con comentarios simultáneamente en la misma página o lista a la misma vez.
Entradas	La pregunta de lluvia de ideas o sugerencia.
Salidas	Conjunto de comentarios en respuesta a la pregunta de lluvia de ideas o sugerencia.
Pasos	<ul style="list-style-type: none"> - Comunicar a los participantes que tienen diferentes páginas (cada una de las cuales tiene un ítem). - Solicitar que hagan sus comentarios en las diferentes páginas. - Dejar que contribuyan con sus ideas hasta que ya no se tengan más comentarios o hasta que se llegue al límite de tiempo determinado previamente.
Características	<p>Es el más simple de todos los Thinklets. Es principalmente para grupos de 5 y más pequeños. Cuando el mismo grupo utiliza <i>OnePage</i>, todas las contribuciones se muestran sobre la página compartida.</p> <p>El problema evidente de este Thinklet es la sobrecarga de información, cuando en una página se tienen alrededor de 100 comentarios.</p>
Historia exitosa	<p>La primera vez que se ha programado una actividad al mismo momento y en diferente lugar, incluía personas sobre un velero en el mar. Inexplicablemente <i>OnePage</i> llegó a ser clave para el éxito. La actividad vinculó a personas en el velero junto con personas en tres laboratorios y dos universidades, así que ellos deberían desarrollar especificaciones para una red experimental. Los participantes se comunicaron por teléfono y radio, tuvieron un GroupSystem que los unía vía Internet.</p>

Tabla 12.Thinklet onepage

Pin the Tail on the Donkey	
Escoger este Thinklet...	<ul style="list-style-type: none"> - Cuando un grupo ha generado algunos comentarios (entre 100 y 400) sobre un conjunto de ideas, propósitos, planes, entre otros. - Para construir conocimiento compartido entre un grupo sobre algunos comentarios importantes y aspectos de discusión. - Para evitar ir con el grupo por cada comentario separadamente, pero enfocarse solamente por los más destacados.
Información general	Este Thinklet es apropiado cuando un grupo ha generado un largo número de comentarios, ideas, proposiciones, entre otros. Durante una plenaria de discusión es muy costoso considerar los comentarios individualmente dado que esto toma mucho tiempo. Con este Thinklet se puede dejar que los miembros del equipo definan sus principales contribuciones. Se puede ayudar a que la gente cree conocimiento compartido con respecto a sus propias apreciaciones.
Entradas	Una extensa cantidad de comentarios que fueron contribuidos por los miembros del grupo en reacción a ideas, proposiciones, propósitos, entre otros.
Salidas	<ul style="list-style-type: none"> - Principales comentarios identificados. - Conocimiento compartido sobre principales comentarios.
Pasos	<p>Informar a los participantes que se ha elaborado una extensa cantidad de aspectos y se han creado algunos comentarios.</p> <p>Solicitar que lean los diferentes comentarios y pongan anotaciones a los que consideren son claves, que cambien la percepción de lo que ya tenían, o que resuman mejor un número de comentarios. Sólo se podrá adicionar un número definido de anotaciones.</p> <p>Después de que el grupo ha ubicado sus anotaciones y ha leído los comentarios destacados, facilitar una discusión durante la cual se invite a las personas para que expliquen la razón por la que ciertos comentarios fueron claves.</p> <p>Continuar moderando la actividad hasta que se llegue al límite de tiempo previamente determinado o hasta que se estén dando contribuciones en la discusión por parte de algún integrante.</p>
Historial exitosas	En alguna ocasión se facilitó un taller con una gran compañía de Seguros de Vida Europea. La compañía tuvo que prepararse para la introducción de la moneda del Euro. El taller comenzó con un número de casos de estudio en los cuales los miembros del grupo tomaron una decisión sobre cómo convertir su moneda nacional a un precio en Euro.

	<p>Cada caso recibió extensos comentarios relacionados con los pro y los contra de varios procedimientos de la transición. En general el grupo recibió cerca de 150 comentarios sobre 5 casos.</p> <p>Se pide a los participantes que fijen cinco comentarios, uno en cada caso, los participantes lo hicieron en 10 minutos. En cada situación se fijaron entre 4 y 6 comentarios. La subsecuente plenaria de discusión les ayudo a todos los participantes a entender los principales argumentos en cada caso de estudio.</p> <p>El grupo usó ese conocimiento para seleccionar un procedimiento de transición para ir desde su moneda tradicional a Euros usando una simple Selección Múltiple en StrawPoll. El resto del taller se invirtió en la generación de aspectos que debían ser tratados antes de que el proceso de transición fuera implementado.</p>
<p>¿Por qué ese nombre?</p>	<p>Pin the Tail on the Donkey es un juego de niños que con frecuencia se da en los cumpleaños. Un dibujo de un burro que no tiene cola se ubica en la pared. Los niños están con los ojos vendados y se les da una cola y un alfiler. El propósito es tratar de colocar la cola en el lugar adecuado del burro. El propósito de este Thinklet es que deje a los participantes fijen las colas de mayor interés (comentarios) en los burros disponibles (ideas, propósitos, planes, etc.).</p>

Tabla 13. Thinklet Pin the Tail on the Donkey.

<p style="text-align: center;">Concentration</p>	
<p>Escoger este Thinklet...</p>	<ul style="list-style-type: none"> - Cuando un grupo ha generado una o más listas de ideas las cuales son redundantes, ambiguas o deben eliminarse. - Cuando es necesario limpiar una lista. Por ejemplo: eliminar redundancia, ambigüedad.
<p>No escoger este Thinklet...</p>	<ul style="list-style-type: none"> - •Si la siguiente actividad del grupo no requiere una lista limpia. Un PopCornSort, por ejemplo, puede hacerse un Concentration de antemano. Este Thinklet puede ser parte de BucketWalk seguido del PopCornSort.

Información general	Este Thinklet deja que usted guíe al grupo a través de proceso limpio y estructurado. Aún la lista más desordenada de lluvia de ideas puede limpiarse guiando al grupo para eliminar duplicados, combinando ideas y reescribiendo ideas no claras. Este Thinklet compromete a las personas en una actividad de limpieza que es motivante y efectiva a la vez. El enfoca la interacción del grupo sobre una lista de ítems que requiere atención y estimula discusión adicional sobre esos ítems.
Entradas	Una lista tosca (compleja) de ideas, la cual contiene ideas solapadas entre sí.
Salidas	Una lista limpia. Por ejemplo, una lista en la cual ideas duplicadas han sido removidas, ideas ambiguas se han reformulado e ideas que han direccionado aspectos similares o conceptos que han sido combinados.
Pasos	<ul style="list-style-type: none"> - Presentar a todos los evaluadores el listado de ítems. - Informar a los evaluadores que es posible que haya algunas ideas similares o que presenten ambigüedad. - Invitarlos a que identifiquen y seleccionen ideas similares o que presenten ambigüedad. - En caso de que algún participante haya identificado ideas similares, se deberá: <ul style="list-style-type: none"> - Invitarlo a que exprese la razón por la que considera que esas ideas son similares. - Decidir con el grupo si dichas ideas deben combinarse o alguna de ellas debe ser eliminada. <p>En caso de que algún participante haya identificado ideas que sean ambiguas:</p> <ul style="list-style-type: none"> - Invitarlo a que exprese qué es lo que no entiende de esa idea. - Preguntar si alguno de los demás participantes puede explicarla. - Invitar a que se proponga una idea alterna. <p>Continuar con este proceso hasta que el grupo no tenga más sugerencias sobre ideas que deben unirse, replantearse o eliminarse.</p>

Historias exitosas	Es uno de los Thinklets que se usan con más frecuencia para ordenar listas de ideas. Una historia que ilustra la efectividad del Thinklet viene de un grupo de oficiales de policía que trabajan en una ciudad importante. El grupo se reunió para determinar los criterios que iban a utilizar para establecer prioridades entre proyectos de crimen organizados. Ellos generaron ideas sobre 80 posibles criterios de selección en 20 minutos. Los siguientes 20 minutos se invirtieron en un juego de concentración. Durante este juego, los participantes disminuyeron la lista de posibles criterios a 35. Finalmente se escogieron los 8 más importantes los cuales fueron probados por un período de juicio en la fuerza policial.
¿Por qué ese nombre?	Concentration es el nombre de un juego común que se juega alrededor del mundo. En este juego, un conjunto de cartas contiene dos copias de un número de imágenes. Las cartas se colocan al revés en una mesa, cada jugador puede abrir dos cartas. Si las cartas tienen la misma imagen, el jugador las guarda y toma otras dos cartas. Si no son iguales, el jugador debe esperar otro turno para sacar nuevas cartas. Como en un juego de cartas, los participantes en un ejercicio de concentración pueden probar y encontrar ideas similares.

Tabla 14. Thinklet Concentration.

MoodRing	
Escoger este Thinklet	<ul style="list-style-type: none"> • Para seguir la trayectoria de consenso de un único tema en tiempo real. • Para saber cuándo es el momento de detener una conversación y tomar una decisión.
Información general	Los participantes registran sus opiniones en un único tópico, luego empieza una discusión. Dado que pueden observar algunas opiniones de los demás, algunas veces cambian su opinión o dirección, lo cual puede llevar a que cambien su voto. Los resultados se actualizan en tiempo real.
Pasos	<ul style="list-style-type: none"> - Solicitar a los participantes que registren su opinión sobre el tema de discusión. - Pedir a los participantes que hablen respecto del tema.

	<ul style="list-style-type: none">- Motivarlos a que expresen su nueva opinión sobre el tema, si escuchan cualquier cosa que cambie la idea que tenían inicialmente.- Se genera discusión hasta que se alcance alguna clase de consenso sobre este tema.- Continuar moderando la actividad hasta un tiempo previamente determinado o hasta que no se estén dando contribuciones en la discusión por parte de algún integrante.
--	--

Tabla 15.Thinklet MoodRing

2. ANEXO B: ESTUDIOS DE CASO PRELIMINAR

En este documento se presenta el diseño, la ejecución y los resultados obtenidos al realizar un estudio de caso exploratorio, con el objetivo de analizar las características de colaboración en niños de 10- 12 años de edad al intentar resolver un problema. Para realizar dicho estudio se tuvo en cuenta algunos lineamientos planteados en [8]. Cabe resaltar que este estudio de caso se realizó como una herramienta para identificar mejor el problema de investigación.

2.1 Diseño de estudio de caso preliminar 1

En la siguiente tabla se presenta el diseño del presente estudio de caso.

Pregunta de investigación
¿Qué características colaborativas se evidencian en los equipos de niños entre 10 y 12 años de edad al intentar resolver un problema utilizando el modelo ChildProgramming?
Objetivos del estudio
<ul style="list-style-type: none">• Analizar las características colaborativas de los niños al intentar resolver un problema utilizando el modelo ChildProgramming [9].• Caracterizar la población objeto de estudio.
Lugar
Institución educativa de Caloto – Cauca.
Selección del estudio y unidad de análisis
La unidad de análisis para este estudio de caso son los equipos de trabajo. El estudio de caso es de tipo embebido considerando 5 unidades de análisis con 24 sujetos de investigación. Las unidades de análisis seleccionadas se organizarán de acuerdo a grupos de trabajos previos de los estudiantes.

<p>Al ser un estudio de tipo revelatorio¹, permitirá analizar las características y comportamientos de los equipos de trabajo. La técnica de recolección de información seleccionada es la observación y se aplicará como instrumento una lista de chequeo, de igual manera, para realizar la caracterización de los niños se desarrollará una encuesta.</p>
<p>Criterios para identificar colaboración en equipos de trabajo.</p>
<p>En [10] se identifica cinco indicadores del éxito de procesos de aprendizaje basados en sistemas colaborativos, los cuales son: el uso de estrategias, la cooperación intra - grupo, revisión de criterios de éxito, el seguimiento y la actuación del grupo. Del mismo modo en el artículo “Indicadores de Cooperación en el Trabajo Grupal” [11] se seleccionan algunos indicadores de colaboración, que puedan ser usados para evaluar la forma en que distintos grupos de personas enfrentan una tarea, en la que requieran colaborar entre sí para resolverla. Los cinco indicadores son: uso de estrategias, cooperación intragrupal, revisión de criterios de éxito, monitoreo y proveer ayuda.</p> <p>A partir de las propuestas [12], [10] y [11], se tomaron como elementos esenciales para realizar actividades de aprendizaje colaborativo los siguientes aspectos:</p> <ul style="list-style-type: none"> - La igual participación. - La interdependencia positiva. - La responsabilidad individual. - Uso de estrategia. - El trabajo del líder.

Tabla 16. Diseño del estudio de caso preliminar

2.1.1 Ejecución del estudio de caso

La actividad 1 descrita en la tabla 2 se realizó durante dos sesiones, donde inicialmente se hizo una introducción de cómo se iba a trabajar y en qué consistía la actividad. Al conformar los equipos se presentaron las tareas a realizar y posteriormente los niños de forma libre construyeron una torre de forma poliédrica. A continuación, se presenta la descripción de actividad realizada, la cual fue propuesta para la validación inicial de ChildProgramming.

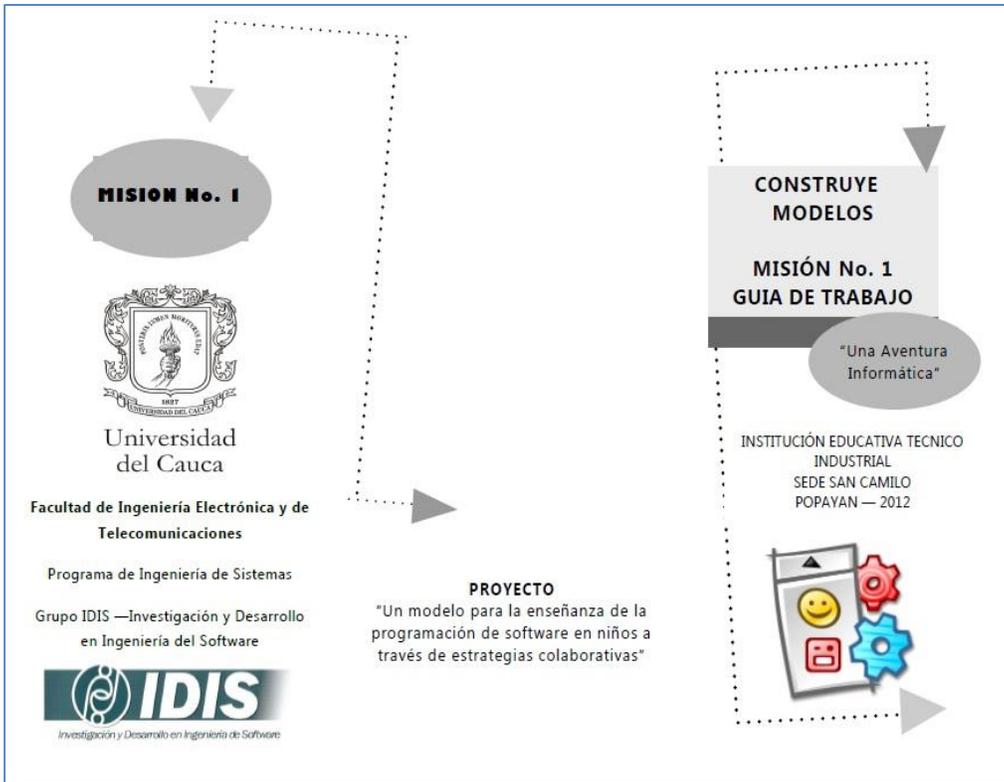
¹ Un estudio de caso estudio revelatorio existe cuando un investigador tiene la oportunidad de observar y analizar un fenómeno previamente inaccesible a la investigación científica.

DESCRIPCIÓN DE LA ACTIVIDAD 1	
Nombre de la Actividad:	Construye Modelos
Descripción:	Es un juego lúdico – colaborativo que hace parte de la misión. El objetivo es construir una estructura poliédrica en condiciones normales, con características similares al ejemplo entregado y contando con un factor de tiempo. El éxito de esta misión depende del grado de colaboración entre los integrantes de cada equipo por ello es necesario que coordinen muy bien el trabajo y manejen un único canal de comunicación de tal forma que se entiendan, comprendan y compartan sus ideas.
Participantes:	<ul style="list-style-type: none"> ✓ Niños de sexto grado de la Institución Educativa ✓ Docente del área de informática de la institución
Conformación de los Equipos:	<p>Para este estudio de caso se contó con un curso donde se organizaron grupos teniendo en cuenta el número de niños que lo conformaban así:</p> <ul style="list-style-type: none"> ▪ Grado 6°: 4 equipos de 5 estudiantes y un equipo de 4 estudiantes. <p>Se permitió que los niños organizarán sus equipos, pues ellos han trabajado desde años anteriores en actividades grupales, esto se considera un factor importante para identificar su comportamiento colaborativo e innato.</p>
Ambiente de Trabajo:	Se ejecutó la actividad en un ambiente amplio donde se pudieran dispersar los niños y no generar desorden, por ello el espacio adaptado fue el patio principal de la institución, el cual estaba dotado de varias mesas de trabajo, pues la actividad estaba pensada para conformar equipos y ubicarlos en cada mesa. La distribución de las mesas estaba organizada en orden de equipos, lo que garantizaría una mejor observación.
Materiales:	<p>Para el desarrollo de la actividad cada equipo contó con los siguientes materiales para realizar las construcciones.</p> <ul style="list-style-type: none"> - 24 espaguetis - 25 Masmelos por equipo - Una hoja de papel de color

	<ul style="list-style-type: none"> - Un lápiz - Un borrador - Un plegable con información de la actividad
Metodología:	<p>Una vez conformados los equipos se ubicaron en la mesa asignada la cual tenía sobre ella el material suficiente y necesario para desarrollar la actividad. Posteriormente se leyeron las instrucciones de la actividad, cada equipo debía repartirse el trabajo adecuadamente para obtener el logro de la meta, esta fue una actividad libre, los niños debían definir la mejor forma para construir la torre. Cada equipo debía asignar un nombre como su identidad y escribirlo en una hoja con sus respectivos integrantes.</p> <p>Durante el desarrollo los equipos se podían acercar a la estructura de ejemplo para tomar detalles de ella y ayudarse un poco para sus construcciones, paralelamente cada equipo era observando con el fin de obtener de ellos las características iniciales de comportamiento y organización. Finalmente, el equipo que iba terminando lo reportaba y se procedía a evaluar la estructura construida de forma detallada para dar la puntuación final al equipo.</p>
Entregables de la Actividad:	Torre con forma poliédrica construida a base de espaguetis y masmelos.
Unidades Generadas	La unidad generada para esta actividad fue una la torre poliédrica de palos y masmelos construida y entregada de forma tangible. (Una sola unidad)
Defecto	Estructuras o torres con imperfecciones de forma, apariencia y similitud con respecto al modelo presentado.

Tabla 17. Descripción Actividad No. 1 tomada y adaptada de [13]

A continuación, se presenta el plegable entregado a los niños para el desarrollo de la actividad y algunas evidencias fotográficas de la actividad realizada.



CONSTRUYE MODELOS

Es un juego lúdico colaborativo y es la primera misión que deben explorar, en la que tendrán que construir una estructura en un límite de tiempo. El éxito de esta misión depende del grado de colaboración entre los integrantes de cada equipo por ello es necesario que coordinen muy bien su trabajo y manejen un canal de comunicación entre ustedes de tal forma que se entiendan y compartan sus ideas.

META
Construir el modelo más preciso y exacto al expuesto en cuanto a la forma, orden y estabilidad.

MATERIALES
24 Palillos de madera de 25 cm
25 Masmelos
Una hoja de papel de color
1 lápiz y 1 borrador

TIEMPO
Tendrán 15 minutos para lograr construir y terminar la estructura propuesta.

PISTAS

- Cada equipo debe repartirse el trabajo adecuadamente para obtener el logro de la meta.
- Cada equipo debe tener un nombre como su identidad dentro de la aventura, escribirlo en una hoja con sus integrantes.
- Observar la estructura a construir detalladamente y determinar una estrategia. (Tiempo Máximo 3 minutos para la planeación).
- Iniciar la construcción de su propia torre.

PUNTUACION
Se dará la puntuación de acuerdo al desarrollo de la construcción y trabajo en equipo. Esta puntuación es acumulable para la siguiente etapa. (50 puntos máximo y 5 puntos mínimo).

DESCUENTOS

- Por cada palillo o masmelos adicional utilizado se restaran 3 puntos del puntaje total
- Por solicitar tiempo adicional (5 min) se restaran 2 puntos.

BUENA SUERTE !!
ADELANTE EN SU PRIMERA MISION

NOMBRE DEL EQUIPO

Vista General

Vista de lado

Vista de Arriba

Figura 7. Plegable entregado. Tomado de [13]



Figura 8. Evidencia fotográfica - Actividad 1

La actividad dos se desarrolló en tres sesiones la cual consistió en el desarrollo de una aplicación básica realizada en Scratch, como se muestra en la siguiente figura.



Figura 9. Evidencia fotográfica - Actividad 2

2.1.2 Resultados del estudio de caso

En este primer acercamiento se pudo apreciar que trabajar realmente de forma colaborativa no es fácil, no basta con disponer a un grupo de personas entorno a una actividad y esperar a que el aprendizaje llegue” [14], “es necesario estructurar actividades para alcanzar ese objetivo” [15]. Respecto a esta estructuración se considera fundamental hacer uso de la Ingeniería de la Colaboración.

A continuación se presenta el análisis general realizado de la observación del comportamiento colaborativo de los niños respecto a los seis lineamientos seleccionados de [12] [10] [11] :

Lineamientos para evaluar trabajo colaborativo tomado de [12] [10] [11]	Resultados en equipos de trabajo
Igual participación	No se cumple la igual participación, algunos niños asumen el liderazgo del desarrollo de las actividades, principalmente cuando los equipos son de 5 estudiantes y haciendo a un lado los aportes de sus compañeros.
Interdependencia positiva	Se pudo apreciar que todos los integrantes de los equipos desean ganar, a pesar de que no todos contribuyen a ello.
Responsabilidad individual	En los equipos no se definieron y asignaron responsabilidades a los integrantes.
Uso de estrategias	Al indagar a los niños por la estrategia diseñada para solucionar el problema, se pudo apreciar que no hay una definición clara de las estrategias.
Trabajo del líder	En los equipos de trabajo la elección del líder y su desempeño no fue una actividad importante para los integrantes.
Cooperación intra-grupal	Al no tener estrategias bien definidas, es difícil hacer evaluación de este lineamiento.
Revisión de criterios de éxito	Los criterios de éxito están dados en términos de lineamientos, límites y roles, los cuales deben definirse al inicio de la actividad, pero en los equipos de trabajo los límites y roles no fueron claros.

Tabla 18. Análisis de la observación respecto a lineamientos

2.1.3 Caracterización de la población

En este estudio de caso también se buscó identificar las características de los niños, relacionadas con conocimientos de programación de software, gusto por el trabajo colaborativo, experiencia en manejo de equipos de cómputo, entre otros aspectos. El instrumento utilizado fue la encuesta “Instrumentos para identificar las características individuales de los niños” [13].

A continuación, se presenta el formato diligenciado.

INSTRUMENTO PARA IDENTIFICAR LAS CARACTERÍSTICAS INDIVIDUALES DE LOS NIÑOS (Anexo H – MODELO CHILDPROGRAMMING)

Nombre: Jhon Esteban Martínez Edad: 11 Fecha: 16-3-2020

1. ¿Te gusta trabajar en equipo?
 - a. Si
 - b. No
 2. ¿Te gusta compartir con tus compañeros de clase?
 - a. Si
 - b. No
 3. ¿Qué tipo de actividades te gusta realizar en equipo?
 - a. Talleres, trabajos o tareas de clase
 - b. Juegos en el recreo
 - c. Proyectos de ciencia
 - d. Jornadas lúdicas (música, danza, teatro)
 4. ¿En el salón de clase cuáles son tus materias favoritas?
 - a. Español y Artística
 - b. Ciencias Naturales e Idiomas (Inglés)
 - c. Ciencias Sociales y Educación Religiosa (Religión)
 - d. Matemáticas e Informática
 - e. Todas las anteriores
 5. ¿En clase de informática te gusta trabajar de forma individual o en equipo?
 - a. Individual
 - b. En Equipo
 6. ¿Has tenido contacto con el computador últimamente?
 - a. Si
 - b. No
 7. ¿Identificas y sabes para que son las partes del computador?
 - a. Si
 - b. No
 8. ¿Te gustaría aprender a usar el computador para realizar tus propias actividades (tareas, juegos, proyectos)?
 - a. Si
 - b. No

¿Por algunas veces qué?
hay mucha falta
 9. ¿Cuando tienes contacto con el computador que te gusta hacer?
 - a. Jugar
 - b. Leer
 - c. Aprender – Investigar Tareas
 - d. Dibujar
 - e. Escuchar Música
 - f. Ver Películas
 - g. Ninguna
10. ¿Conoces el término programación?
 - a. Si
 - b. No.
 11. ¿Cuál de estas tareas requieren programación?
 - a. Hacer tareas y jugar
 - b. Consultar en internet
 - c. Ver Películas y escuchar música
 - d. Prender el Computador
 - e. Hacer tus propios juegos
 - f. Ninguna
 12. ¿Te gustaría aprender a hacer tus propios programas en el computador?
 - a. Si
 - b. No.

¿Por se necesita estudiar muchas cosas qué?
 13. ¿Te gusta jugar con cosas que te enseñen aunque no te diviertan?
 - a. Si
 - b. No
 14. ¿Qué te gustaría hacer cuando aprendas a programar?
 - a. Hacer juegos
 - b. Hacer programas de otro tipo (educativo por ejemplo)

¿Cuáles?
hacer dibujos

 - c. ninguno
 - d. Otros.

¿Cuáles?



Muchas Gracias por tu colaboración

Figura 10. Encuesta aplicada para la gamificación

El análisis de la encuesta permitió identificar las características de los niños, obteniendo los siguientes resultados: el 95% de los niños manifiestan que les gusta trabajar en equipo, también les agrada organizar equipos principalmente para desarrollar talleres, hacer tareas, y ejecutar proyectos. Se evidencia que en los últimos días el 50% ha tenido poco contacto con el computador, sin embargo, expresan deseo de querer aprender a usarlo mejor para realizar principalmente sus actividades escolares; del mismo modo, manifiestan que las principales actividades que desarrollan al tener un equipo son: jugar, realizar tareas y leer. Aunque el 38% manifiesta conocer el concepto de programación de software, al analizar

las respuestas de la pregunta tales como: ¿Cuáles tareas requieren programación?, son muy pocos los niños que identifican tareas que requieren realmente programación de software. El 95% manifiesta querer aprender a hacer sus propios programas en el computador, y muestran gran interés principalmente por desarrollar juegos y programas educativos.

Esta caracterización permitió identificar los perfiles de los niños, que están siendo capacitados en la construcción de algoritmos, y el manejo de Scratch 2 como herramienta de desarrollo de software.

Conclusiones

- En este estudio de caso se pudo apreciar que a pesar de que los niños eligieron sus compañeros de equipo, y se asumió que ya habían desarrollado actividades grupales, se evidenció que no siempre trabajan colaborativamente.
- Para los niños fue muy importante elegir compañeros de su mismo género.
- En los equipos de trabajo no se definieron roles ni se asignaron responsabilidades.
- Algunos niños se vieron opacados por sus compañeros, principalmente en grupos de cinco estudiantes.
- Al no haber estrategias definidas para el desarrollo de los objetivos algunos integrantes de los equipos entraron en controversia al querer defender sus soluciones.
- Organizar equipos y planear una actividad colaborativa no garantiza que los grupos logren una verdadera colaboración.
- Este tipo de actividades motiva a los niños a compartir espacios de conocimiento.
- Respecto al folleto de la actividad, se pudo apreciar que los niños no leen las indicaciones, la mayoría se limitó a observar y seguir el modelo gráfico plasmado en el plegable.

3. ANEXO C. ESTUDIO DE CASO: APLICACIÓN DEL MODELO CHILDPROGRAMMING -G EN EL DESARROLLO DE UNA APLICACIÓN SOFTWARE

Este anexo presenta la experiencia del estudio de caso al aplicar el modelo Child Programming -G [13], mediante la construcción y desarrollo de un algoritmo de la vida real por parte de niños que participaron en el estudio de caso. Mediante este estudio de caso se busca analizar las características colaborativas presentes en los niños al resolver problemas utilizando el modelo Child Programming y una herramienta de programación. La Ingeniería de la Colaboración es una aproximación sistemática para el diseño de procesos y tecnologías colaborativos reutilizables, que pretende guiar la ejecución de tareas colaborativas, de forma predecible y exitosa entre los miembros de un equipo de trabajo [16]. Para el desarrollo del presente caso de estudio se tuvo en cuenta los lineamientos propuestos en [8] y las primeras actividades propuestas en el método SCASE (Case Study based Analysis in Collaboration Engineering) [17].

La actividad principal consiste en construir una aplicación software dada por el docente, se asume que el éxito de las actividades plantadas depende del grado de colaboración entre los integrantes de cada equipo, por ello es necesario que los participantes coordinen muy bien su trabajo y manejen un único canal de comunicación de tal forma que se entiendan, comprendan y compartan sus ideas [13], la definición de la actividad fue adaptada de acuerdo a las características de la población objetivo, teniendo en cuenta el contexto identificado.

CONTEXTO:

La aplicación de la actividad se realizó con 24 estudiantes de grado sexto de una Institución Educativa, ubicada en zona rural del Municipio de Caloto - Cauca, los niños cuentan con una edad promedio de 11 años, la composición étnico-cultural de la población es homogénea, con mayoría de raza mestiza e indígena. La institución cuenta con elementos básicos para llevar a cabo el proceso educativo desde transición a educación media.

Respecto a recursos tecnológicos computacionales se cuenta con una sala de sistemas dotada de 8 equipos portátiles y 10 equipos de escritorio, la conexión a Internet es muy inestable y se presentan fluctuaciones del servicio de energía comúnmente.

3.1 Diseño estudio de caso 2

Pregunta de investigación
¿Qué características colaborativas se evidencian en los equipos de niños entre 10 y 12 años de edad de la Institución Educativa Huasano, al resolver problemas aplicando el modelo Child Programming?
Objetivos del estudio
Analizar las características colaborativas de los niños entre 10 y 12 años de edad, al intentar resolver un problema aplicando el modelo Child Programming.
Lugar
Institución educativa del municipio de Caloto.
Selección del Estudio y unidad de análisis
La unidad de análisis para este estudio de caso son los equipos de trabajo. El estudio de caso es de tipo embebido considerando 5 unidades de análisis con 24 sujetos de investigación. Las unidades de análisis seleccionadas se organizaron de acuerdo a grupos de trabajos previos de los estudiantes. Con el objetivo de evaluar las características y comportamientos de los sujetos dentro de sus equipos de trabajo el estudio es clasificado de tipo Revelatorio ² .
Criterios para identificar colaboración en equipos de trabajo.
A partir de las propuestas [12], [10] y [11] para el desarrollo de este estudio de caso se tomaron como punto de referencia los siguientes criterios: la igual participación, la

² Un estudio de caso estudio revelatorio existe cuando un investigador tiene la oportunidad de observar y analizar un fenómeno previamente inaccesible a la investigación científica.

interdependencia positiva, la responsabilidad individual, el uso de estrategia, la cooperación intragrupal y la revisión de criterios de éxito.

Tabla 19. Diseño de estudio de caso 2

3.2 Ejecución del estudio de caso 2

Este estudio de caso se ejecutó a en el mes de abril del 2017. La actividad se realizó durante 2 sesiones, cada una con una duración de 50 minutos, donde inicialmente se realizó una introducción de cómo se iba a trabajar y en qué consistía la actividad en general. Previo a la aplicación de la actividad presentada en la tabla 1, los estudiantes en clases de Tecnología e Informática recibieron información del modelo ChildProgramming y sus prácticas propuestas, de igual manera se han estado preparando a los estudiantes en la construcción de algoritmos y el entorno de programación Scratch.

Se organizaron 4 equipos de 5 estudiantes, y un equipo de 4 estudiantes, todos los estudiantes contaron con las mismas condiciones de trabajo y presentaron características similares en cuanto a conceptos lógicos, operacionales y formales del lenguaje. Es así como antes de iniciar la actividad todos los estudiantes recibieron orientaciones generales por parte de la docente – Investigadora.

A continuación se presenta la actividad a desarrollar [13]

DESCRIPCIÓN DE LA ACTIVIDAD	
Nombre de la Actividad:	Construye aplicaciones.
Descripción:	Cada grupo de niños recibe un guía del reto a resolver e inicia a desarrollarlo
Participantes:	<ul style="list-style-type: none"> ✓ Niños de sexto grado de una Institución Educativa del municipio de Caloto - Cauca. ✓ Docente del área de informática de la institución. ✓ Observadores/Investigadores.
Conformación de los Equipos:	Para este estudio de caso se contó con 1 curso, donde se organizó grupos teniendo en cuenta el número de niños que lo conformaban así: <ul style="list-style-type: none"> ▪ Grado 6°: 4 equipos de 5 estudiantes y un equipo de 4 estudiantes.
Ambiente de Trabajo:	Se ejecutó la actividad en un ambiente amplio donde se pudieran dispersar los niños y no generar desorden, el cual estaba dotado de varias mesas de

	trabajo (4) ya que la actividad estaba pensada para conformar equipos y ubicarlos en cada mesa. La distribución de las mesas estaba organizada en orden de equipos, lo que garantizaría una mejor observación, cada mesa contaba con los materiales necesarios para el desarrollo de la actividad.
Materiales:	Para el desarrollo de la actividad cada equipo contó con los siguientes materiales para realizar las construcciones. <ul style="list-style-type: none"> - Portátiles con Scratch instalado - Papel bond - Marcadores - Lapiceros - Guía de la misión a realizar
Metodología:	Una vez conformados los equipos se ubicaron en la mesa asignada la cual tenía sobre ella el material suficiente y necesario para desarrollar la actividad. Una vez leídas las instrucciones de la actividad, cada equipo debía repartirse el trabajo adecuadamente para obtener el logro de la meta. Se pidió que cada equipo se asignara un nombre como su identidad y lo escribieron en una hoja con sus respectivos integrantes.
Entregables de la Actividad:	Software realizado en scratch. Taskboard de los equipos.

Tabla 20. Descripción Actividad No. 1

En las siguientes figuras se muestra parte del diseño y desarrollo de la aplicación realizada por los estudiantes.



Figura 11. Evidencia fotográfica análisis



Figura 12. Desarrollo de aplicaciones caso de estudio 2.

3.3 Resultados del estudio de caso

Observaciones generales. De las actividades desarrolladas se observó:

- Elegir el líder de cada equipo causó dificultades en la mayoría de los equipos, sin embargo, hubo grupos que no realizaron dicha actividad.
- De seis equipos solo dos definieron el nombre para el equipo, los demás entregaron la hoja asignada para ello en blanco.
- Todos los equipos se demoraron por lo menos 4 minutos más de lo planeado para la actividad 1.
- Los niños leen muy poco las instrucciones dadas en el plegable, por lo cual muchas de las recomendaciones de trabajo en equipo plasmadas en el folleto no fueron tenidas en cuenta, la mayoría se limitan a observar solo las gráficas del folleto.
- Los equipos no definieron estrategias claras.
- Los grupos no asignaron responsabilidades claras, de acuerdo a las pistas dadas en el descriptivo del folleto y las orientaciones ofrecidas por la docente.
- Algunos niños fueron opacados por sus compañeros, principalmente en los equipos de cinco integrantes, por lo cual fue muy limitada su participación.
- En general los equipos mostraron agrado e interés por el desarrollo de las actividades.

ANEXO D – EVALUACIÓN DE CRITERIOS PARA IDENTIFICAR SI UNA ACTIVIDAD REALIZADA POR NIÑOS REQUIERE TRABAJO COLABORATIVO EN SU EJECUCIÓN.

Instrumento utilizado: A continuación, se presenta el instrumento utilizado para analizar los criterios que aplican para identificar si una actividad realizada por niños requiere trabajo colaborativo en su ejecución.



UNIVERSIDAD DEL CAUCA
GRUPO DE INVESTIGACION Y DESARROLLO EN
INGENIERIA DEL SOFTWARE
MAESTRIA EN COMPUTACIÓN



VALIDACIÓN POR PARTE DE EXPERTOS EN INGENIERIA DE LA COLABORACIÓN, DE CRITERIOS PARA IDENTIFICAR SI UNA ACTIVIDAD REALIZADA POR NIÑOS REQUIERE TRABAJO COLABORATIVO EN SU EJECUCIÓN.

Director: PhD Cesar Alberto Collazos

Estudiante: Ing. Ana María Chimunja González

Objetivo del instrumento: En el marco del trabajo de Investigación: “ChildProgramming – C: Extendiendo ChildProgramming³ desde la Ingeniería de la Colaboración”, se plantea la presente encuesta con la finalidad de que usted como experto en Ingeniería de la Colaboración, nos ayude a validar si los criterios presentados en la tabla 1, son aplicables para identificar si una actividad de un proceso de desarrollo de software ejecutada por un equipo de niños, requiere trabajo colaborativo en su ejecución. Gracias por sus valiosos aportes.

³ ChildProgramming es un modelo para el desarrollo de software llevado a cabo por niños, se compone de tres dimensiones [2]: la dimensión cognitiva, lúdica y colaborativa.

Datos del experto:

Nombre	
Sexo	
Institución	
Tiempo de Experiencia en la Ingeniería de la Colaboración	
Último título Académico	



UNIVERSIDAD DEL CAUCA
GRUPO DE INVESTIGACION Y DESARROLLO EN
INGENIERIA DEL SOFTWARE
MAESTRIA EN COMPUTACIÓN



Desarrollo:

- a) Los criterios del 1 al 4 presentados en la siguiente tabla han sido planteados en el estudio [1], con el objetivo de identificar si una actividad requiere trabajo colaborativo en su desarrollo, dichos criterios se han aplicado en equipos de adultos y en procesos colaborativos de evaluación de usabilidad. Al considerar la interdependencia positiva y la responsabilidad individual como aspectos fundamentales de los procesos colaborativos, en la presente investigación también se proponen los criterios 5 y 6.

A partir de su valiosa experiencia y conocimiento en el área de Ingeniería de la Colaboración, le pedimos nos ayude a verificar si los siguientes criterios son aplicables para identificar si una actividad llevada a cabo por niños, requiere trabajo colaborativo durante un proceso de desarrollo de software:

Id Criterio	Criterio para identificar si una actividad requiere trabajo colaborativo en su ejecución.	¿El criterio es aplicable en proceso de desarrollo de software por niños?		Observación
		Si	No	
1	La ejecución de la actividad y el plan de trabajo requieren incluir varias personas, las cuales pueden tener diferentes roles.			

2	Se requiere contar con personas que tienen un alto grado de experticia en un área específica de conocimiento, para la ejecución de las actividades.			
3	Es necesario compartir conocimiento, recursos e información con otras personas.			
4	Es necesario tener en cuenta los diferentes aportes, opiniones y puntos de vista de otros integrantes del grupo que está ejecutando la actividad.			
5	La actividad permite a los participantes asumir responsabilidades individuales.			
6	Es necesario definir una estrategia para la ejecución de la actividad.			

- c) Cómo experto, considera que hace falta algún criterio para evaluar si una actividad desarrollada por niños requiere trabajo colaborativo? Si su respuesta es afirmativa por favor indique ¿cuál?

Gracias por su valiosa colaboración en este proceso de investigación.

PARTICIPANTES: A continuación se presenta la caracterización de cada uno de los expertos participantes en la escueta presentada.

Nombre	Toni Granollers
Sexo	Masculino
Institución	Universidad de Lleida (España)
Tiempo de Experiencia en la Ingeniería de la Colaboración	10 años
Último título Académico	Doctor
Nombre	Gabriel Mauricio Ramírez Villegas
Sexo	M
Institución	Universidad del Cauca – Universidad Nacional Abierta y a Distancia

Tiempo de Experiencia en la Ingeniería de la Colaboración	10
Nombre	Andrés Solano
Sexo	M
Institución	Universidad Autónoma De Occidente
Tiempo de Experiencia en la Ingeniería de la Colaboración	5 Años
Último título Académico	Doctorado
Nombre	Jeferson Arango López
Sexo	Masculino
Institución	Universidad del Cauca – Universidad de Granada (España)
Tiempo de Experiencia en la Ingeniería de la Colaboración	2 años
Nombre	Luis Freddy Muñoz Sanabria
Sexo	Masculino
Institución	Fundación Universitaria De Popayán
Tiempo de Experiencia en la Ingeniería de la Colaboración	1 Año
Último título Académico	Doctorado En Ciencias De La Electrónica (Computación)
Nombre	Ángela Patricia Villareal Freire
Sexo	Femenino
Institución	Universidad del Cauca
Tiempo de Experiencia en la Ingeniería de la Colaboración	1 año
Último título Académico	Magíster en computación
Nombre	Andrés Felipe Aguirre Aguirre
Sexo	Masculino
Institución	Universidad del Cauca
Tiempo de Experiencia en la Ingeniería de la Colaboración	1 año
Último título Académico	M.Sc en Computación

Nombre	Libardo Pantoja
Sexo	M
Institución	Unicauca
Tiempo de Experiencia en la Ingeniería de la Colaboración	No tengo experiencia en este tema, pero conozco cosas de CSCW.
Último título Académico	Magister en Computación
Nombre	Alejandro Fernández
Sexo	Masculino
Institución	Universidad Nacional de La Plata (Argentina)
Tiempo de Experiencia en la Ingeniería de la Colaboración	15 años
Último título Académico	Ph.D.

Tabla 21. Caracterización de expertos.

Resultados de la aplicación de las encuestas:

A continuación, se presenta la valoración dada por cada uno de los expertos en Ingeniería de la Colaboración.

N°	Criterio para evaluar las actividades que requieren trabajo colaborativo (Aplicación en proceso de desarrollo de software por niños)	Aplicación en proceso de desarrollo de software por niños								
		Exp1	Exp2	Exp3	Exp4	Exp5	Exp6	Exp7	Exp8	Exp9
C1	1. La ejecución de la actividad y el plan de trabajo requieren incluir varias personas, las cuales pueden tener diferentes roles.	Si	Si	Si	Si	Si	Si	Si	Si	Si
C2	2. Se requiere contar con personas que	No	No	No	No	Si	No	Si	No	No

	tienen un alto grado de experticia en un área específica de conocimiento, para la ejecución de las actividades.									
C3	3. Es necesario compartir conocimiento, recursos e información con otras personas.	Si								
C4	4. Es necesario tener en cuenta los diferentes aportes, opiniones y puntos de vista de otros integrantes del grupo que está ejecutando la actividad.	Si								
C5	5. La actividad permite a los participantes asumir responsabilidades individuales	Si	No							
C6	6. Es necesario definir una estrategia para la ejecución de la actividad.	No	Si	No	Si	Si	Si	Si	Si	No

Tabla 22. Resultados de valoración de expertos

4. ANEXO E – SENSIBILIZANDO A LOS NIÑOS SOBRE LA IMPORTANCIA DEL TRABAJO COLABORATIVO ANTES DE INICIAR UN PROCESO DE DESARROLLO DE SOFTWARE

En la presente sección se muestra un conjunto de experiencias realizadas con veinte estudiantes entre diez y once años de edad en una Institución Educativa ubicada en la zona rural del municipio de Caloto – Cauca. Teniendo en cuenta las situaciones presentadas en los estudios de caso de los anexos 2 y 3, y con el ánimo de aportar significativamente al modelo Childprogramming-C, se realizó una búsqueda de estrategias que permitieran sensibilizar a los equipos de niños sobre la importancia del trabajo colaborativo con la ayuda de un equipo de docentes.

A pesar de encontrar en la literatura varias dinámicas para ser trabajadas en equipos, es muy escasa la información sobre estrategias para sensibilizar a los niños al trabajo colaborativo principalmente al iniciar un proceso de desarrollo de software; por lo cual se recurrió a construir un plan de actividades que involucra estrategias didácticas como dinámicas, presentación de videos, análisis de casos. A continuación, se describen brevemente dos de las actividades realizadas:

4.1 Desarrollo actividad 1

Análisis de la colaboración en situaciones presentadas en videos: para llevar a cabo la siguiente actividad se realizó una búsqueda y selección de videos relacionados con trabajo colaborativo disponibles en la web. En la tabla II se describe la actividad realizada.

Análisis de videos	
Objetivo	Sensibilizar a los niños sobre la importancia de la colaboración y sus características mediante el análisis de videos que promueven el trabajo en equipo.

Análisis de videos	
Materiales	Videos, video beam o pantalla de TV, hojas de papel, marcadores, lapiceros.
fuentes	https://www.youtube.com/watch?v=e2xNXr4oox8 https://www.youtube.com/watch?v=LAOICltn3MM https://www.youtube.com/watch?v=226iRoOSIWM https://www.youtube.com/watch?v=f5acUrUZIMs&t=4s https://www.youtube.com/watch?v=Q0rRUAMevUI https://www.youtube.com/watch?v=HiITPKoOsw https://www.youtube.com/watch?v=TiNUaHXsjBA https://www.youtube.com/watch?v=AlaqP8Yobol https://www.youtube.com/watch?v=e2xNXr4oox8
Desarrollo	<p>En diferentes jornadas de clase los niños observaron los videos relacionados con la importancia del trabajo colaborativo, y mediante la estrategia de lluvia de ideas se analizaron preguntas tales como:</p> <ul style="list-style-type: none"> - ¿Cuál es la meta de los integrantes del equipo presentado en los videos? - ¿Qué pasaría si los protagonistas de los videos no trabajan colaborativamente? - ¿Cuál es la responsabilidad de cada uno de los protagonistas del video? - ¿Cuál es la estrategia que utiliza el equipo para lograr el objetivo? - ¿Qué pasaría si un integrante del equipo abandona la misión? - ¿Qué papel he desempeñado usted cuando ha sido parte de los equipos de desarrollo de software de la clase? - ¿Cuál ha sido el mayor inconveniente para trabajo en un equipo?

Análisis de videos	
	Finalmente se resalta la importancia de la colaboración, la responsabilidad individual, la interdependencia positiva y la igual participación, como características fundamentales del trabajo colaborativo, que deben promoverse durante el desarrollo colaborativo, así mismo se muestra la importancia del líder.

Tabla 23. Actividad 1 – análisis de videos

Esta actividad se realizó en los momentos de motivación durante aproximadamente seis clases de Informática y Tecnología. La actividad propuesta permitió que los niños identificaran características del trabajo colaborativo, y además reflexionaran acerca de sus comportamientos como integrantes de los equipos organizados en las actividades de desarrollo de software previamente ejecutadas.

De igual manera, se contó con el apoyo interdisciplinar de otros docentes quienes en sus jornadas de clase también motivaron a los estudiantes a trabajar colaborativamente. La siguiente figura muestra alguna de las jornadas donde se presentaron los videos.



Figura 13. Niños participantes en presentación de videos.

4.2 Desarrollo actividad 2

Desarrollando estrategias. Consiste en plantear actividades lúdicas que le permitan a los niños identificar y analizar las características del trabajo colaborativo mientras juega. En la tabla III y IV y las fig. 4 y 5, se muestran ejemplos de las actividades realizadas.

Planteando estrategias – 1	
Objetivo	Sensibilizar a los niños sobre la importancia de la colaboración y sus características mediante el uso de dinámicas fuera del aula de clase.
Materiales	Vasos, listones, agua
fuelle	https://www.youtube.com/watch?v=A_XaSBEYi1E
Desarrollo	Los niños deben buscar la mejor estrategia para llevar un vaso con agua, a lo largo de un trayecto sin tocar la superficie del vaso, los únicos elementos que pueden tocar son listones que deben organizar estratégicamente para poder transportar el agua, y llenar otro vaso que está colocado a 10 metros al frente de cada equipo.

Tabla 24. Planeando estrategia - 1.



Figura 14. Participación de los niños en actividades lúdicas.

Planteando estrategias – 2	
Objetivo	Sensibilizar a los niños sobre la importancia de la colaboración y sus características mediante el uso de juegos fuera del aula de clase. Incentivar a los niños sobre la importancia de la definición de estrategias para el rendimiento de los equipos de trabajo.
Materiales	Botellas con huecos, vasos, agua.
fuentes	https://www.youtube.com/watch?v=ymlpS8ndF0E (minuto 2.45)
Desarrollo	Como se puede ver en la Figura 3 Los niños deben buscar la mejor estrategia para llenar con agua una botella que tiene muchos orificios. Un integrante del equipo se encarga de traer un vaso con agua, mientras los demás integrantes deben buscar la forma de tapar los huecos de la botella y evitar que el agua se salga. Al final gana el equipo que más rápido llene la botella.

Tabla 25. Planeando estrategia – 2.



Figura 15. Participación de los niños en actividades lúdicas.

Al finalizar cada actividad se realizó el análisis y reflexión de la participación de cada uno de los integrantes, además de los nuevos valores que los niños estaban aprendiendo. De estas actividades se pudo confirmar que los aspectos cognitivos efectivos y motivacionales

involucrados en el aprendizaje orientado a grupos tienen impacto en el proceso del mismo, al estar expuestos a diferentes puntos de vista y al observar un gran número de ejemplos, los estudiantes retan sus propias ideas y así motivan su aprendizaje, de este modo, la cooperación y el trabajo en equipo pueden soportar y fomentar los esfuerzos de aprendizaje individuales [18].

Las actividades propuestas para sensibilizar a los niños hacia la importancia del trabajo colaborativo, permitieron mejorar las características de la colaboración tales como: la interdependencia positiva, la igual participación, la responsabilidad individual, la definición de estrategias durante el desarrollo de software y el desempeño de los líderes de los equipos.

5. ANEXO F. ANALIZANDO LA CAPACIDAD DE APRENDIZAJE EN EL MODELO CHILDPROGRAMMING, COMO UNA SUBCARACTERÍSTICA DE LA USABILIDAD

En la investigación y consolidación de ChildProgramming se han realizado diferentes estudios de casos, donde los docentes han manifestado algunas dificultades en el momento de interpretar y aplicar el modelo en el entorno escolar. Aunque ChildProgramming no es un producto software, sino un proceso, en este documento se muestra la evaluación de la capacidad de aprendizaje que ofrece el modelo a sus usuarios finales, como una de las subcaracterísticas de la usabilidad propuestas en la norma ISO/IEC 25010. Para dicha evaluación se definieron métricas relacionadas con la familiaridad, el lenguaje común, la consistencia y la intuición, que a pesar de ser expuestas para la evaluación de productos software, en este trabajo se adaptan para evaluar un modelo de procesos, con el objetivo de identificar posibles mejoras en la estructuración y descripción del modelo ChildProgramming, para que sea más fácil de comprender y usar.

5.1 ISO/IEC 25000 SQuaRE (System and Software Quality Requirements and Evaluation)

La familia ISO/IEC 25000 es el resultado de la evolución de otras normas anteriores orientadas hacia la calidad de software, especialmente de las normas ISO/IEC 9126, que describe las particularidades de un modelo de calidad del producto software, e ISO/IEC 14598, que abordaba el proceso de evaluación de productos software [19].

Dentro de la familia ISO/IEC 25000 se encuentra el modelo de calidad del producto definido por la ISO/IEC 25010 [20] que representa la piedra angular en torno a la cual se establece

el sistema para la evaluación de la calidad de un producto, en este modelo se determinan las características de calidad que se van a tener en cuenta a la hora de evaluar las propiedades de un producto software determinado.

Como se puede apreciar en la siguiente figura, la calidad del producto software está compuesta por ocho características: adecuación funcional, eficiencia de desempeño, compatibilidad, usabilidad, fiabilidad, seguridad, mantenibilidad y portabilidad [20].



Figura 16. Características del producto software[20]

Dentro de este trabajo se seleccionó la usabilidad como la característica más adecuada, para analizarla y mejorar el modelo de procesos de ChildProgramming.

5.1.1 Usabilidad

La ISO/IEC 25010 define la usabilidad como la capacidad del producto software para ser entendido, aprendido, usado y resultar atractivo para el usuario, cuando se usa bajo determinadas condiciones [21]; la usabilidad se subdivide a su vez en las siguientes subcaracterísticas relacionadas con productos software [21].

Subcaracterística	Descripción
Capacidad para reconocer su adecuación.	Capacidad del producto que permite al usuario entender si el software es adecuado para sus necesidades.
Capacidad de aprendizaje.	Capacidad del producto que permite al usuario aprender su aplicación.
Capacidad para ser usado	Capacidad del producto que permite al usuario operarlo y controlarlo con facilidad.
Protección contra errores de usuario	Capacidad del sistema para proteger a los usuarios de hacer errores.
Estética de la interfaz de usuario	Capacidad de la interfaz de usuario de agrandar y satisfacer la interacción con el usuario.
Accesibilidad	Capacidad del producto que permite que sea utilizado por usuarios con determinadas características y discapacidades.

Tabla 26. Subcaracterísticas de Usabilidad ISO/IEC 25010 [20] [21]

5.1.1.1.1 Capacidad de aprendizaje

La capacidad o facilidad de aprendizaje, es una subcaracterística de la usabilidad que permite que un producto pueda ser usado por los usuarios con el fin de lograr aprender a usarlo con eficiencia, efectividad, libre de riesgos y satisfacción en el contexto de uso especificado [21], describe la capacidad para permitir a los usuarios realizar tareas en el primer intento, y la rapidez con la cual un usuario se adapta al sistema e inicia a utilizar servicios y disponer de la información que el sistema provea [22]. El objetivo de la capacidad de aprendizaje es minimizar el tiempo que se requiere desde el no conocimiento de una aplicación hasta su uso productivo [22] [23], logrando calidad de uso para los usuarios a través del tiempo [24].

Aunque todas las definiciones de capacidad de aprendizaje como subcaracterística de la usabilidad se focalizan en el uso de aplicaciones software, mediante esta propuesta se

busca contextualizar esta subcaracterística para evaluar la capacidad de aprendizaje que ofrece el modelo de procesos de ChildProgramming.

5.1.2 Definición de las métricas relacionadas con atributos de la facilidad de aprendizaje

El proceso de software es importante para el éxito de los proyectos de desarrollo de software [25] y debe ser utilizable porque [26][27] [28] [29]:

- ✓ Ayuda a los intérpretes a entender, comunicar y ejecutar el proceso.
- ✓ Facilita el seguimiento y control del proceso de ejecución.
- ✓ Facilita la evolución y reutilización del proceso.
- ✓ Apoya la gestión, evaluación y mejora del proceso.
- ✓ Apoya la ejecución automatizada del proceso de software (siempre que sea posible).
- ✓ Proporciona una base para la construcción de una cultura de Ingeniería de Software.

Dentro de los estudios de caso que se han realizado para aplicar el modelo Chilprogramming, se han recibido sugerencias por parte de los docentes para mejorar la usabilidad del modelo, de esta forma se ha considerado que el modelo ChildProgramming debe ser fácil de entender y aplicar para un docente, porque en todas las ocasiones no se podrá contar con el rol del investigador o experto que guíe el proceso.

Teniendo en cuenta las ventajas de tener un proceso usable y los problemas presentados por los docentes al aplicar el modelo ChildProgramming sin la asesoría de un experto, el equipo de investigadores del grupo IDIS organizó un grupo focal donde participaron seis docentes, para realizar un análisis cualitativo de las subcaracterísticas de la usabilidad de un producto software, con el objetivo de evaluar el modelo de procesos de ChildProgramming.

El perfil de los participantes fue heterogéneo, respecto a la edad participaron docentes entre 25 y 42 años, y teniendo en cuenta que en las instituciones educativas el área de Tecnología en Informática no siempre es orientada por docentes Ingenieros, se invitó a participar a docentes de otras áreas como, por ejemplo: Matemáticas, Emprendimiento e inglés, que

Intuitivo:	Indicador: nivel de intuición: porcentaje de metáforas de diseño que son adecuadas y bien utilizadas por el sistema luego de un tiempo prudente de entrenamiento sobre alguna de las tareas del modelo.	No necesariamente es aplicable porque se busca evaluar la capacidad de aprendizaje del modelo, sin tener un entrenamiento previo.
-------------------	--	---

Tabla 27. Atributos y métricas relacionados con la capacidad de aprendizaje en aplicaciones software [6][23].

Al identificar que la mayoría de atributos evaluados pueden ser adaptados en la evaluación de la capacidad de aprendizaje del modelo ChildProgramming se proponen los siguientes atributos y métricas.

Métrica
<p>Atributo: familiaridad del modelo: Relación del conocimiento que tiene el docente o usuario del modelo, y el conocimiento que necesita para interactuar con el modelo Childporgramming.</p> <p>Indicador: familiaridad: porcentaje de conceptos utilizados por el modelo que resultan comunes o convencionales para los docentes.</p>
<p>Atributo: Lenguaje Común: Los nombres y contenidos de fases, actividades, roles y prácticas utilizadas del modelo, resulta sencillo y natural para el docente.</p> <p>Indicador: naturalidad de contenido: porcentaje del contenido que maneja un lenguaje natural y común para el docente o usuario del modelo Childprogrammig.</p>
<p>Atributo: Consistente: Convención unificada para descripción de fases, roles y prácticas del modelo.</p> <p>Indicador: consistencia: cumplimiento de un nombrado unificado de componentes del modelo en todas las secciones del documento y gráficos representativos del modelo, así como en las guías de las prácticas lúdicas, colaborativas y ágiles.</p>

Tabla 28. Atributos y métricas relacionados con la capacidad de aprendizaje en childprogramming

Este conjunto de atributos fue valorado por expertos, quienes ayudaron a mejorar las métricas. Posteriormente se definió las escalas de medición para la familiaridad y el

lenguaje común, además de las posibles técnicas e instrumentos para llevar a cabo la evaluación.

5.1.3 Métrica definida para evaluar la familiaridad

Para la evaluación de la familiaridad del modelo se ha definido la métrica *conocimientos claves (CC)*: relacionada con los conocimientos que tiene el docente y el conocimiento que necesita para interactuar con el modelo.

Métrica: Valor de conocimientos utilizados por el modelo que resultan comunes o convencionales para los docentes.

$$A = \frac{\# \text{ de práctica conocidas por el docente}}{\# \text{ de prácticas del modelo}}$$

$$B = \frac{\# \text{ de roles conocidas para el usuario}}{\# \text{ de roles del modelo}}$$

$$C = \frac{\# \text{ de las actividades conocidas para el docente}}{\# \text{ de actividades definidas en las fases del modelo}}$$

$CC = \frac{\text{suma de puntuación de elementos aplicados}}{\text{Número de elementos Aplicados}}$
--

Escala de valoración para CC:

Nivel	Rango
Alta	0,8 - 1
Media	0,61 – 0,79
Baja	0 – 0,6

5.1.4 Métrica definida para evaluar el lenguaje común.

Esta métrica define que los nombres y contenidos de las fases, prácticas y roles propuestos en Childprogramming resulten sencillos y naturales para el docente.

Métrica: LC= promedio entre los conocimientos claves (CC) y la relación de conceptos que maneja un lenguaje natural para el docente (NC).

Los conceptos claves se definirán relacionados con: las prácticas del modelo (ágiles, cognitiva colaborativas), roles y actividades de las fases propuestas por el modelo.

$$A = \frac{\text{\# de conceptos claves de las prácticas comunes para el docente}}{\text{\# de concepto claves de prácticas del modelo}}$$

$$B = \frac{\text{\# de conceptos de roles comunes para el usuario}}{\text{\# de conceptos de roles del modelo}}$$

$$C = \frac{\text{\# de conceptos de actividades comunes para el usuario}}{\text{\# de conceptos definidos en las actividades de las fases}}$$

$$NC = \frac{\text{suma de puntuación de elementos aplicados}}{\text{Número de elementos Aplicados}}$$

Escala de valoración para NC:

Nivel	Rango
Alta	0,9 -1
Media	0,41 – 0,89
Baja	0 – 0,4

$$LC = \frac{CC + NC}{2}$$

Escala de valoración para LC:

Nivel	Rango
Alta	0,9 -1
Media	0,41 – 0,89
Baja	0 – 0,4

Análisis de consistencia y nivel de intuición

Al analizar la consistencia como el cumplimiento de un nombrado unificado de componentes del modelo en todas las secciones del documento y gráficos representativos del modelo, se pudo detectar que antes de evaluar y definir escalas de medición para esta métrica es necesario contar con más material didáctico elaborado para el soporte de las actividades de los docentes.

Evaluación de las métricas propuestas para la evaluación de capacidad de aprendizaje en el modelo childprogramming.

Evaluación de la familiaridad del modelo Childprogramming.

Técnica/ instrumento: para la evaluación de la métrica propuesta se utilizó como técnicas la entrevista y la encuesta. Se aplicó un cuestionario a una muestra de ocho usuarios docentes. El instrumento está disponible en:

(

https://docs.google.com/a/misena.edu.co/forms/d/1wpQ5QOf_mt2kAFARwDWfbThHf_RKPNOUOtbaVhGfDBE/viewform?pli=1).

Evaluación y Resultados: El cuestionario planteado se aplicó a una muestra de docentes en la ejecución de estudios de caso de la validación del modelo Childprogramming. La familiaridad evaluada como la relación del conocimiento que tiene el docente o usuario del modelo, y el conocimiento que necesita para interactuar con el modelo Childporgramming, tuvo un resultado de 0.64 escalafonado en un nivel medio. Esto indica que se debe utilizar

conceptos más comunes y convencionales para los docentes.

Evaluación del lenguaje común del modelo ChildProgramming

Técnica/ instrumento: para la evaluación de la métrica propuesta se utilizó como técnica de investigación verbal el grupo focal que "es un método empírico de investigación, rentable y rápido para obtener información cualitativa y realimentación (desde un grupo específico), el cual se puede utilizar en varias fases y tipos de investigación" [30], además permite: obtener realimentación de los participantes sobre preguntas de investigación o nuevos conceptos, [31] "obtener realimentación sobre la manera en que los modelos o conceptos son presentados o registrados" [32]

Evaluación y resultados: La evaluación de esta métrica se realizó con el mismo grupo de docentes que participaron en la evaluación de la métrica de familiaridad, se obtuvo un resultado de 0.577 en la escala, lo cual indica que se debe mejorar en aspectos relacionados con el lenguaje del modelo.

Para la evaluación de la consistencia del modelo se utilizó las técnicas de entrevista y grupos de discusión dirigida, indicando que en la documentación del modelo se cumple con los nombrados unificados de fases, roles, y prácticas ágiles, lúdicas y colaborativas, sin embargo, es necesario elaborar manuales más prácticos para los docentes que quieran implementar el modelo Childprogramming en sus aulas de clase.

CONCLUSIONES

Aunque ISO/IEC 25010 ofrece normas y estándares directos para la evaluación de usabilidad de productos software, este trabajo es un acercamiento para evaluar la capacidad de aprendizaje en un proceso que busca enseñar a desarrollar software.

Es común encontrar propuestas de modelos y metodologías para el desarrollo de procesos en diferentes áreas del conocimiento, sin embargo, muchas veces no se mide si realmente dichas propuestas son fáciles de aprender, comprender y por tanto usar, para un usuario final quien realmente pondrá en práctica un proceso.

Aunque Childprogramming se continuará formalizando y documentando como un modelo

de procesos desde la Ingeniería del Software y la mejora de procesos, este trabajo permitió identificar la necesidad de estructurar el modelo pensando también en el usuario final, por tal motivo se diseñaran guías prácticas y más didácticas que orienten a los docentes u otras personas que deseen aplicar el modelo a un grupo de niños para enseñar a programar.

Contar con usuarios como los docentes dentro de todo el proceso de Childprogramming, ha permitido a lo largo de los años consolidar un modelo de desarrollo de software que ha mejorado desde diferentes aspectos de la Ingeniería del software, la gamificación, la Ingeniería de la Colaboración, y continua en proceso de evolución para seguir contribuyendo al desarrollo del pensamiento computacional de los niños de una manera dinámica y metódica.

6. ANEXO G. TÉCNICAS DE APRENDIZAJE COLABORATIVO

A continuación se presenta un conjunto de 23 técnicas colaborativas que el docente podría utilizar desarrollar los procesos colaborativos, las técnicas se clasifican en [33].



Figura 17. Clasificación de técnicas colaborativas

TECNICAS COLABORATIVAS DE DISCUSIÓN		
Técnica colaborativa:	Es una técnica en la cual los estudiantes:	Es particularmente usado para:

1: Think-Pair-Share (Pensar para compartir)	Piensan individualmente durante unos minutos, y luego discuten y comparan sus respuestas con un compañero antes de hablar con toda la clase.	Preparar a los estudiantes para participar más completa y efectivamente en discusiones de toda la clase.
2: Round Robin	Generan ideas y hablan en orden, pasando de un estudiante al siguiente.	Estructurar las sesiones de brain-strooming y asegurarse que todos los estudiantes participen.
3: Buzz Groups	Discuten informalmente preguntas relacionadas con el curso en pequeños grupos de compañeros.	Generar mucha información e ideas en un corto período de tiempo para prepararse y mejorar las discusiones de toda la clase.
4: Talking Chips	Participan en una discusión grupal y entrega una ficha cada vez que hablen.	Asegurando una participación equitativa.
5: Three-Step Interview	Entrevistarse con un par e informar lo que aprenden.	Ayudar a los estudiantes a conectarse y mejorar las habilidades de comunicación.
6: Debates críticos	Asumen y discuten un problema con un compañero, que está en oposición a sus puntos de vista personales.	Desarrollar habilidades de pensamiento crítico y codificar a los estudiantes para desafiar sus suposiciones.

Tabla 29. Técnicas colaborativas de discusión

TÉCNICAS COLABORATIVAS DE ENSEÑANZA RECÍPROCA ENTRE COMPAÑEROS

Este CoLT	Es una técnica en la que los estudiantes:	Es particularmente usado para:
7: Tomar notas	Agrupar la información de sus notas individuales para crear una versión mejorada con la de su compañero.	Ayudar a los estudiantes a adquirir información faltante y corregir imprecisiones en sus notas y aprender a ser mejores tomadores de notas.
8: Celulas de aprendizaje Learning Cell	Aprenden a cuestionarse unos a otros utilizando preguntas que hayan desarrollado individualmente sobre una tarea de lectura u otra actividad de aprendizaje.	Involucrar activamente a los estudiantes para que piensen sobre el contenido y alentándolos a desafiarse entre sí para perseguir niveles más profundos de pensamiento.
9: Pecera Fishbowl	Se organizan grupos concéntricos los grupos exteriores escuchan y observan las discusiones de los grupos concéntricos más pequeño.	Brindar oportunidades para que los estudiantes observen modelos de procesos grupales en un escenario de discusión
10: Juego de roles	Asumir una identidad diferente y representar un escenario	Involucrar a los estudiantes en una actividad creativa que los ayude a "aprender haciendo".
11: Jigsaw (Rompecabezas)	Desarrollar el conocimiento de un tema determinado y luego se lo enseñó a otros.	Motivar a los estudiantes para que aprendan y procesen información lo suficientemente profunda como para enseñársela a sus compañeros

12: Equipos de prueba	Prepararse para una prueba en grupos de trabajo, tome la prueba individualmente y luego vuelva a tomar la prueba en sus grupos	Ayudar a los estudiantes a evaluar y mejorar su comprensión de la materia, ya que también se enseñan mutuamente estrategias para tomar evaluaciones.
-----------------------	--	--

Tabla 30. Técnicas de enseñanza recíproca entre compañeros

TÉCNICAS COLABORATIVAS PARA RESOLVER PROBLEMAS		
Técnica colaborativa:	Es una técnica en la cual los estudiantes:	Es particularmente usado para:
13: Solución de problemas de pensar en voz alta	Resolver problemas en voz alta para probar su razonamiento	Enfatizar el proceso de resolución de problemas (en lugar del producto) y ayudar a los estudiantes a identificar errores de lógica o proceso.
14: Enviar un problema	Tratar de resolver un problema como un grupo, y luego pasar el problema y la solución a un grupo cercano quienes harán lo mismo; el grupo final evaluará la solución.	Ayudar a los estudiantes a practicar juntos las habilidades de pensamiento requeridas para una efectiva resolución de problemas y para comparar y discriminante entre múltiples soluciones.
15: Caso de estudio	Revisar un estudio escrito de un escenario del mundo real y desarrollar una solución al dilema presentado en el caso	Presentar principios abstractos y teorías de manera que los estudiantes encuentran relevancia.
16: Solución estructurada de Problemas	Seguir un formato estructurado para resolver problemas	Dividiendo los procesos de resolución de problemas en pasos manejables para que los estudiantes no se sientan abrumados y para que ellos

		aprendan a identificar, analizar, y resolver problemas de una manera organizada.
17: Equipos analíticos	Asumir roles y tareas específicas para realizar al leer de forma crítica una tarea, escuchando una conferencia, o viendo un video.	Ayudando a los estudiantes a comprender las diferentes actividades construyendo un análisis crítico.
18: Grupos de Investigación	Plan, conducta, e informe sobre proyectos de investigación en profundidad	Enseñar a los estudiantes procesos de investigación y ayudándolos a obtener un conocimiento profundo acerca de áreas específicas

Tabla 31. Técnicas colaborativas para resolver problemas

TÉCNICAS COLABORATIVAS DE ORGANIZACIÓN GRÁFICA		
Técnica colaborativa:	Es una técnica en la cual los estudiantes:	Es particularmente usado para:
19: Afinidad grupal	Generar ideas, identificar temas en común, y luego ordenar y organizar las ideas en consecuencia.	Ayudar a los estudiantes a “descomprimir” un tema complicado e identificar y clasificar las partes que lo constituyen
20: cuadrícula de grupo	Se les da información y se les pide que los coloquen en las celdas en blanco de una cuadrícula según las rubricas de categoría.	Aclarar categorías conceptuales y desarrollar habilidades de clasificación
21: Matriz de equipo	Discriminar entre conceptos similares al notar y marcar en un gráfico la presencia o ausencia de características importantes y definitorias	Distinguir entre conceptos estrechamente relacionados

22: Cadenas de Secuencia	Analizar y representar gráficamente una serie de eventos, acciones, roles o decisiones.	Comprender los procesos, causas y efectos, y organizar la información en una progresión ordenada y coherente.
23: Webs de Palabras	Generar una lista de ideas y luego organícelas en una relación gráfica, Utilizando líneas o flechas para representar la conexión	Descifrar y representar la relación; como mapas, estos pueden mostrar tanto el destino como los sitios y las vistas en el camino

Tabla 32. Técnicas colaborativas de organización gráfica.

7. ANEXO H. DESCRIPCIÓN DEL MODELO CONCEPTUAL CHILDPROGRAMMING

En este anexo se describe el modelo conceptual de ChildProgramming, teniendo en cuenta sus componentes colaborativos, lúdicos y ágiles [13] [34].

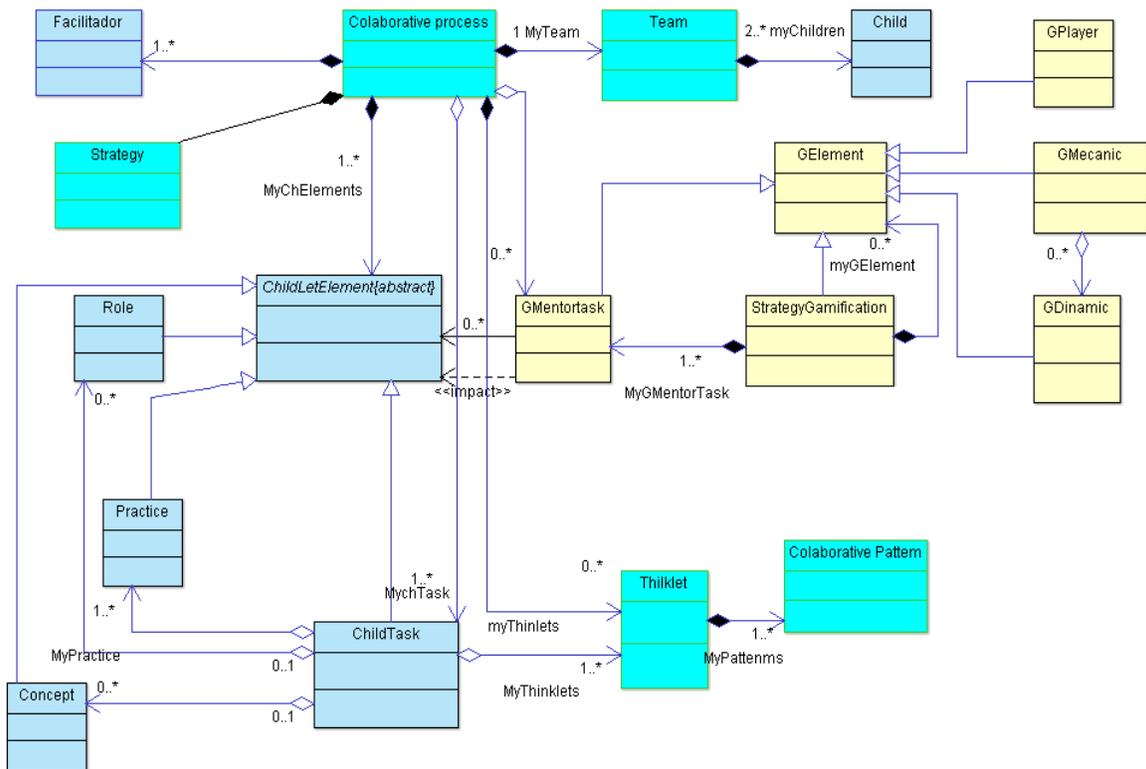


Figura 18. Modelo conceptual del Childprogrammig-C

A continuación, se presenta cada uno de los conceptos que hacen parte del modelo abstracto de ChildProgramming [13] [34].

CLASE	DESCRIPCIÓN
<div style="border: 1px solid black; padding: 5px; background-color: #e6f2ff;"> <p style="text-align: center; margin: 0;">Child</p> <ul style="list-style-type: none"> - Cod_Child : int - Name_Child : String - Grade_Child : String - Age_Child : int </div>	<p>Niño (Child): concepto que representa la información de un niño(a) que es integrante de un equipo de trabajo, identificado por el atributo <i>Cod_Child</i>. Este concepto permite la conformación de equipos (<i>Team</i>) y es la encargada de abstraer toda la información particular del niño como su nombre, grado y edad, con los atributos <i>Name_Child</i>, <i>Grade_Child</i>, <i>Age_Child</i>. <i>Child</i> representa al niño (alumno), actor principal en el proceso Child Programming.</p>
<div style="border: 1px solid black; padding: 5px; background-color: #e6f2ff;"> <p style="text-align: center; margin: 0;">Team</p> <ul style="list-style-type: none"> - Id_Team : int - Name_Team : String - Description_Team : String </div>	<p>Equipo (Team): clase que contiene la información de un equipo. Un equipo es identificado por el atributo <i>Id_Team</i>. Un Equipo es el que ejecuta un Proceso Colaborativo (<i>Collaborative Process</i>) y abstrae toda la información del equipo a través los atributos: <i>Name_Team</i> y <i>Description_Team</i>. Esta clase tiene una relación de composición fuerte con la clase <i>Collaborative Process</i>. <i>Team</i> representa el equipo al que pertenecen los niños en el proceso Child Programming.</p>
<div style="border: 1px solid black; padding: 5px; background-color: #e6ffe6;"> <p style="text-align: center; margin: 0;">Collaborative Process</p> <ul style="list-style-type: none"> - id_CollaborativeProcess : int - Name_Process : String - Goal_Process : String </div>	<p>Proceso Colaborativo (Collaborative Process): clase que abstrae toda la información referente al proceso colaborativo que sigue ChildProgramming, un proceso colaborativo se identifica con el atributo <i>Id_CollaborativeProcess</i> y abstrae toda la información del proceso a través de los atributos <i>Name_Process</i> y <i>Goal_Process</i>. <i>Collaborative Process</i> representa el proceso colaborativo en el que participan los grupos de estudiantes dentro de ChildProgramming y en el cual las mecánicas y dinámicas de juego se incorporan y visualizan para gamificar el entorno y para que los estudiantes las tengan en cuenta, lo que hace que se mejore el comportamiento y aumente el compromiso.</p>
<div style="border: 1px solid black; padding: 5px; background-color: #e6f2ff;"> <p style="text-align: center; margin: 0;">Facilitator</p> <ul style="list-style-type: none"> - Id_Facilitator : int - Description_Facilitator : String </div>	<p>Facilitador (Facilitator): clase que contiene la información de un facilitador. Un facilitador es la persona encargada de definir y conducir las actividades prácticas a realizar. Es identificado</p>

	<p>por el atributo <i>Id_Facilitator</i>. El facilitador y el equipo dan inicio al Proceso Colaborativo (<i>Collaborative Process</i>).</p>										
<table border="1" data-bbox="272 338 599 569"> <tr> <td colspan="2" style="text-align: center;">ChildLetElement</td> </tr> <tr> <td colspan="2" style="text-align: right;">(abstract)</td> </tr> <tr> <td>- Id_ChElement</td> <td>: int</td> </tr> <tr> <td>- Name_ChElement</td> <td>: String</td> </tr> <tr> <td>- Description_ChElement</td> <td>: String</td> </tr> </table>	ChildLetElement		(abstract)		- Id_ChElement	: int	- Name_ChElement	: String	- Description_ChElement	: String	<p>Elemento (ChildLetElement): esta es una clase abstracta donde se describen los atributos y métodos comunes para las clases heredadas (<i>Concept, Practice, ChildLet Task</i> y <i>Role</i>). Esta clase abstrae la información referente a los elementos que hacen parte del ChildLet a través del identificador único <i>Id_ChElement</i> y de los atributos <i>Name_ChElement, Descripción_ChElement</i>. Un elemento es considerado una unidad fundamental dentro del proceso, la unión de varias unidades determina el desarrollo del Proceso Colaborativo.</p>
ChildLetElement											
(abstract)											
- Id_ChElement	: int										
- Name_ChElement	: String										
- Description_ChElement	: String										
<table border="1" data-bbox="272 800 586 940"> <tr> <td colspan="2" style="text-align: center;">ChildLet Task</td> </tr> <tr> <td>- Id_ChTask</td> <td>: int</td> </tr> <tr> <td>- Inputs_Task</td> <td>: String</td> </tr> <tr> <td>- Outputs_Task</td> <td>: String</td> </tr> </table>	ChildLet Task		- Id_ChTask	: int	- Inputs_Task	: String	- Outputs_Task	: String	<p>Tarea (ChildLet Task): esta es una clase heredada de <i>ChildLetElement</i>. Describe atributos propios de la clase como: <i>Inputs_Task</i> y <i>Outputs_Task</i>, un <i>ChildLet Task</i> se identifica a través del atributo <i>Id_ChTask</i>. Esta clase abstrae toda la información de una tarea colaborativa propia del proceso. Una tarea especifica una funcionalidad o acción que se quiera realizar, para childprogramming las tareas definen las acciones colaborativas que los equipos deben realizar para cumplir una misión. Es una generalización de la clase <i>ChildLetElement</i>, una versión más específica de esta clase, además, tiene una asociación de agregación con la clase <i>Collaborative Process</i>.</p>		
ChildLet Task											
- Id_ChTask	: int										
- Inputs_Task	: String										
- Outputs_Task	: String										
<table border="1" data-bbox="272 1398 613 1577"> <tr> <td colspan="2" style="text-align: center;">Practice</td> </tr> <tr> <td>- Id_Practice</td> <td>: int</td> </tr> <tr> <td>- Inputs_Practice</td> <td>: String</td> </tr> <tr> <td>- Outputs_Practice</td> <td>: String</td> </tr> </table>	Practice		- Id_Practice	: int	- Inputs_Practice	: String	- Outputs_Practice	: String	<p>Práctica (Practice): clase heredada de <i>ChildLetElement</i>, una <i>Practice</i> se identifica con el atributo <i>Id_Practice</i> y describe atributos propios como: <i>Inputs_Practice</i> y <i>Outputs_Practice</i>. Esta clase abstrae toda la información relacionada con las prácticas (ágiles, colaborativas o cognitivas) que durante el proceso los niños tendrán en cuenta como un refuerzo aplicable para dar solución a las tareas. Es una generalización de la clase <i>ChildLetElement</i>, una versión más específica de esta clase, además, tiene una asociación de agregación con la clase <i>ChildLet Task</i>.</p>		
Practice											
- Id_Practice	: int										
- Inputs_Practice	: String										
- Outputs_Practice	: String										

<pre> class Concept - Id_Concept : int - Note_Concept : String </pre>	<p>Concepto (Concept): clase heredada de <i>ChildLetElement</i>, un Concept se identifica con el atributo <i>Id_Concept</i> y describe un atributo propio <i>Note_Concept</i>. Esta clase abstrae toda la información relacionada con los conceptos (de tipo colaborativo, ágil y cognitivo) que durante el proceso los niños tendrán en cuenta para solucionar sus tareas. Esta clase tiene una relación de agregación con la clase <i>ChildLet Task</i>. Esta clase es una generalización de la clase <i>ChildLetElement</i>, una versión más específica de esta clase, además, tiene una asociación de agregación con la clase <i>ChildLet Task</i>.</p>
<pre> class Role - Id_Role : int </pre>	<p>Rol (Role): clase heredada de <i>ChildLetElement</i>, un rol se identifica con el atributo <i>Id_Rol</i>. Esta clase abstrae toda la información relacionada con los roles que durante el proceso los niños desempeñaran teniendo en cuenta las tareas asignadas. Esta clase es una generalización de la clase <i>ChildLetElement</i>, una versión más específica, además, tiene una asociación de agregación con la clase <i>ChildLet Task</i>. <i>Rol</i> representa el comportamiento que asumirá el niño como parte del desarrollo de la tarea dentro del proceso Child Programming.</p>
<pre> class Collaborative Pattem - Id_CollaborativePattem : int - Name_Pattem : String - Description_Pattem : String </pre>	<p>Patrón Colaborativo (Collaborative Pattern): clase que contiene la información de un collaborative pattern, un patrón de colaboración se identifica con el atributo <i>Id_CollaborativePattern</i>. Esta clase tiene una asociación de composición fuerte con la clase <i>Collaborative Process</i>. <i>Collaborative Pattern</i> representa una guía de ejecución del proceso.</p>
<pre> class Thinklet - Id_Thinklet : int - Name_Thinklet : String - Description_Thinklet : String - Inputs_Thinklet : String - Outputs_Thinklet : String - Steps_Thinklet : String </pre>	<p>Thinklet: clase que contiene la información de un Thinklet, un Thinklet se identifica con el atributo único <i>Id_Thinklet</i>. Esta clase depende de la existencia de la clase <i>Collaborative Process</i> y abstrae toda la información del thinklet a través los atributos: <i>Name_Thinklet</i>, <i>Description_Thinklet</i>, <i>Inputs_Thinklet</i>, <i>Outputs_Thinklet</i>, <i>Steps_Thinklet</i>. Además, la clase thinklet tiene una relación de agregación con la clase <i>ChildLet Task</i>. <i>Thinklet</i> representa una unidad de construcción</p>

	<p>que especifica como un determinado patrón de colaboración debe utilizarse cuando el proceso está en ejecución.</p>										
<table border="1" data-bbox="289 430 587 623"> <tr> <td colspan="2" style="text-align: center;">GElement</td> </tr> <tr> <td>- id_GElement</td> <td>: int</td> </tr> <tr> <td>- Name</td> <td>: String</td> </tr> <tr> <td>- Description</td> <td>: String</td> </tr> <tr> <td>- Target</td> <td>: String</td> </tr> </table>	GElement		- id_GElement	: int	- Name	: String	- Description	: String	- Target	: String	<p>Elemento de juego o de la estrategia de gamificación (GElement): Clase que tiene la abstracción de los elementos de juego o los elementos de la estrategia de gamificación. Los elementos de juego que describe son las mecánicas de juego (GMecanica), las dinámicas de juego (GDinamica) y los tipos de jugadores (GJugadores). Los elementos de la estrategia de gamificación que describe son las tareas realizadas por el docente (GMentorTask) y la estrategia de gamificación (StrategyGamification). Está identificada con el atributo id_GElement y esta descrita con los atributos Nombre (Name), la descripción (Description) y el objetivo (Target).</p>
GElement											
- id_GElement	: int										
- Name	: String										
- Description	: String										
- Target	: String										
<table border="1" data-bbox="289 976 579 1136"> <tr> <td colspan="2" style="text-align: center;">GMentorTask</td> </tr> <tr> <td>- id_GMentorTask</td> <td>: int</td> </tr> <tr> <td>- Inputs_Task</td> <td>: String</td> </tr> <tr> <td>- Outputs_Task</td> <td>: String</td> </tr> </table>	GMentorTask		- id_GMentorTask	: int	- Inputs_Task	: String	- Outputs_Task	: String	<p>Tarea del mentor (GMentorTask): Concepto identificado con el atributo id_GMentorTask que representa la información de una tarea que el docente o mentor debe realizar en el proceso ChildProgramming-G. Esta se relaciona fuertemente con las estrategias de gamificación (StrategyGamification). Describe atributos propios de la clase como: Inputs_Task y Outputs_Task.</p>		
GMentorTask											
- id_GMentorTask	: int										
- Inputs_Task	: String										
- Outputs_Task	: String										
<table border="1" data-bbox="289 1354 583 1486"> <tr> <td colspan="2" style="text-align: center;">StrategyGamification</td> </tr> <tr> <td>- id_EstrategiaG</td> <td>: int</td> </tr> </table>	StrategyGamification		- id_EstrategiaG	: int	<p>Estrategia de gamificación (StrategyGamification): Contiene la información de una estrategia de gamificación. Esta clase hereda de los elementos de la gamificación y está compuesta por tareas que realiza el mentor (GMentorTask) y por elementos de la gamificación (GElement).</p>						
StrategyGamification											
- id_EstrategiaG	: int										
<table border="1" data-bbox="289 1621 574 1764"> <tr> <td colspan="2" style="text-align: center;">GJugador</td> </tr> <tr> <td>- id_GJugador</td> <td>: int</td> </tr> <tr> <td>- Type</td> <td>: String</td> </tr> </table>	GJugador		- id_GJugador	: int	- Type	: String	<p>Tipo de jugador (GJugador): Clase que representa la información de un tipo de jugador. Esta identificado con el atributo id_GJugador y abstrae la información del tipo de jugador con el atributo Tipo (Type) el cual puede ser Explorador, Triunfador, Socializador y Asesino. Los tipos de jugadores se utilizan para poder analizar, seleccionar y</p>				
GJugador											
- id_GJugador	: int										
- Type	: String										

	realizar una óptima estrategia de gamificación (StrategyGamification).				
<table border="1"> <tr> <td>GMecanica</td> </tr> <tr> <td>- id_GMecanica : int</td> </tr> <tr> <td>- utilidadChPG : String</td> </tr> <tr> <td> </td> </tr> </table>	GMecanica	- id_GMecanica : int	- utilidadChPG : String		<p>Mecánica de juego (GMecanica): Clase que abstrae toda la información referente a una mecánica de juego, la cual hace parte de los elementos de juego que forman y constituyen la gamificación y la estrategia de gamificación (StrategyGamification) que se va a aplicar en el proceso colaborativo (Collaborative Process) de ChildProgramming-G. Se identifica con el atributo id_GMecanica y abstrae toda la información de la mecánica de juego a través de los atributos heredados de la clase GElement. Además, se describe con el atributo propio utilidad (utilidadChPG). En el proceso ChildProgramming-G las mecánicas de juego generan las dinámicas de juego y en esta clase se agregan.</p>
GMecanica					
- id_GMecanica : int					
- utilidadChPG : String					
<table border="1"> <tr> <td>GDinamica</td> </tr> <tr> <td>- id_GDinamica : int</td> </tr> <tr> <td>- utilidadChPG : String</td> </tr> <tr> <td> </td> </tr> </table>	GDinamica	- id_GDinamica : int	- utilidadChPG : String		<p>Dinámica de juego (GDinamica): Clase que abstrae toda la información referente a una dinámica de juego, la cual hace parte de los elementos de juego que forman y constituyen la gamificación y la estrategia de gamificación (StrategyGamification) que se va a aplicar en el proceso colaborativo (Collaborative Process) de ChildProgramming-G. Se identifica con el atributo id_GDinamica y abstrae toda la información de la dinámica de juego a través de los atributos heredados de la clase GElement. Además, se describe con el atributo propio utilidad (utilidadChPG).</p>
GDinamica					
- id_GDinamica : int					
- utilidadChPG : String					

Tabla 33. Descripción de las clases del modelo Chilprogramming

REFERENCIAS

- [1] R. O. Briggs, R. O. Briggs, G. De Vreede, G. De Vreede, J. F. Nunamaker Jr., and J. F. Nunamaker Jr., "Collaboration Engineering with ThinkLets to Pursue Sustained Success with Group Support Systems.," *J. Manag. Inf. Syst.*, vol. 19, no. 4, pp. 31–64, 2003.
- [2] R. O. Briggs, A. P. Massey, and G.-J. de Vreede, "Collaboration Engineering: Foundations and Opportunities: Editorial to the Special Issue on the Journal of the Association of Information Systems Journal of the Association for Information Systems Collaboration Engineering: Foundations and Opportunities," no. March 2016, 2009.
- [3] G. L. Kolschoten and E. L. Santanen, "Reconceptualizing generate thinklets: The role of the modifier," *Proc. Annu. Hawaii Int. Conf. Syst. Sci.*, pp. 1–10, 2007.
- [4] G. M. de Vreede, Gert Jan, Luis Guerrero, "Groupware: Design, Implementation, and Use," vol. 3198, p. 2004, 2004.
- [5] G.-J. De Vreede and R. O. Briggs, "Collaboration Engineering: Designing Repeatable Processes for High-Value Collaborative Tasks," *Proc. 38th Annu. Hawaii Int. Conf. Syst. Sci.*, vol. 0, no. C, p. 17, 2005.
- [6] "Collaboration Engineering."
- [7] A. F. Solano, "Metodología para la evaluación colaborativa de la usabilidad de sistemas software interactivos," Universidad del Cauca, 2015.
- [8] P. Runeson and M. Höst, "Guidelines for conducting and reporting case study research in software engineering," *Empir. Softw. Eng.*, 2009.
- [9] J. Hurtado, C. Collazos, S. Cruz, and O. Rojas, "Child Programming: Una Estrategia de Aprendizaje y Construcción de Software Basada en la Lúdica, la Colaboración y la Agilidad," *Rev. Univ. RUTIC*, vol. 1, pp. 9–14, 2012.
- [10] C. A. Collazos *et al.*, "Evaluating collaborative learning processes using system-based measurement," *Educational Technology and Society*, vol. 10, no. 3. pp. 257–274, 2007.
- [11] L. Guerrero, R. Alarcón, and C. Collazos, "Indicadores de cooperación en el trabajo grupal," in *Memorias de la Conferencia Latinoamericana de Informática, CLEI 2000*, 2000, p. 10.
- [12] C. A. Collazos, "Diseño de Actividades de Aprendizaje Colaborativo Asistidas por Computador," *Rev. Educ. en Ing.*, vol. 9, no. 17, pp. 143–149, 2014.
- [13] S. T. Cruz and O. E. Rojas, "Un Modelo Para la Enseñanza de la Programación de Software en niños a través de Estrategias Colaborativas," Universidad del Cauca, 2013.
- [14] C. Collazos and J. Mendoza, "How to take advantage of ' cooperative learning ' in the classroom," *Educ. y Educ.*, vol. Volumen 9, no. 4128, pp. 61–76, 2006.
- [15] J. George, L. Gan, and B. Jessica, "Learning cooperative learning via cooperative learning," *Hawker Brownlow Education Learning*, 2007.
- [16] G. L. Kolschoten and G.-J. Vreede, "The Collaboration Engineering Approach for Designing," in *Groupware: Design, Implementation, and Use*, vol. 4715, 2007, pp. 95–110.
- [17] J. Hurtado and C. Collazos, "Analyzing and Evaluating Collaborative Processes using Case Studies in the Software Development Context," *Proc. XV Int. Conf. Hum. Comput. Interact.*, pp. 1–2, 2014.
- [18] P. Mendoza and P. Galvis, "Juegos Multiplayer: Juegos Colaborativos Para La Educación1," *Informática*

Educ., vol. 11, no. 2, pp. 223–239, 1998.

- [19] “Standardization, International Organization for Standardization, Ginebra.pdf.” .
- [20] ISO, “ISO/IEC CD 25010: Software Engineering – Software Product Quality Requirements and Evaluation (SQuaRE) – Software and Quality in Use Models (Resolution 937),” p. 42, 2008.
- [21] “I. 2. SQuaRE, «Software Product Quality Requirements and Evaluation – Quality Model.» ISO, 2012.,” p. 2012, 2012.
- [22] N. B. Moe and T. Dybå, “The use of an electronic process guide in a medium-sized software development company,” *Softw. Process Improv. Pract.*, vol. 11, no. 1, pp. 21–34, 2006.
- [23] T. Grossman, G. Fitzmaurice, and R. Attar, “A Survey of Software Learnability: Metrics, Methodologies and Guidelines,” *Proc. 27th Int. Conf. Hum. factors Comput. Syst.*, pp. 649–658, 2009.
- [24] N. BEVAN and M. MACLEOD, “Usability measurement in context,” *Behav. Inf. Technol.*, vol. 13, no. 1–2, pp. 132–145, 1994.
- [25] M. N. ri Mahrin, D. Carrington, and P. Strooper, “Investigating factors affecting the usability of software process descriptions,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 5007 LNCS, pp. 222–233, 2008.
- [26] B. Curtis, M. I. Kellner, and J. Over, “Process modeling,” *Commun. ACM*, vol. 35, no. 9, pp. 75–90, 1992.
- [27] “Kellner, M.I., Becker-Kornstaedt, U., Riddle, W.E., Tomal, J., Verlage, M.: Process guides: effective guidance for process participants. In: Proc. of the Fifth International Conference on the Software Process: Computer Supported Organizational Work, pp. 1,” vol. 25, p. 1998, 1998.
- [28] “Wang, Y., King, G.A.: Software engineering processes: principles and applications. CRC Press, Boca Raton (2000),” p. 2000, 2000.
- [29] L. J. Osterweil, “Software processes are software too,” *Proc. 9th Int. Conf. Softw. Eng.*, vol. 3, pp. 2–13, 1987.
- [30] J. Kontio, L. Lehtola, and J. Bragge, “Using the focus group method in software engineering: obtaining practitioner and user experiences,” *2004 Int. Symp. Empir. Softw. Eng. ISESE 2004*, pp. 271–280, 2004.
- [31] “Wohlin, C., Runeson, P., Höst, M., Ohlsson, M. C., Regnell, B., & Wesslén, A. (2012). Experimentation in software engineering . Springer Science & Business Media.,” p. 2012, 2012.
- [32] “Kim, M. (2011, May). An exploratory study of awareness interests about software modifications. In Proceedings of the 4th International Workshop on Cooperative and Human Aspects of Software Engineering (pp. 80-83). ACM.,” p. 2011, 2011.
- [33] B. E. F. Cross Patricia, Major Claire Howell, “Collaborative learning techniques : A Handbook for college faculty,” p. 2005, 2005.
- [34] A. A. Garcia and H. Orejuela, “ChildProgramming-G: Extendiendo ChildProgramming con Técnicas de Gamificación,” Universidad del Cauca Facultad, 2014.