



**UNIVERSIDAD ESTATAL DE MILAGRO  
FACULTAD CIENCIAS DE LA INGENIERÍA**

**TRABAJO DE TITULACIÓN DE GRADO PREVIO A LA OBTENCIÓN  
DEL TÍTULO DE INGENIERO EN SISTEMAS COMPUTACIONALES**

**PROPUESTA PRÁCTICA DEL EXAMEN DE GRADO O DE FIN DE  
CARRERA (DE CARÁCTER COMPLEXIVO)  
INVESTIGACIÓN DOCUMENTAL**

**TEMA: ANÁLISIS DE LA IMPORTANCIA DEL MODELADO UML EN  
EL DISEÑO DE SOFTWARE**

**AUTORES:**

- SALAVARRÍA ARISTEGA KEVIN JAVIER
- ROJANO LÁRRAGA CARLOS FERNANDO

**ACOMPañANTE:**

- MSC. MIRELLA AZUCENA CORREA PERALTA

**Milagro, Mayo 2018  
ECUADOR**

## DERECHOS DE AUTOR

Ingeniero.

Fabricio Guevara Viejó, PhD.

**RECTOR**

**Universidad Estatal de Milagro**

Presente.

Nosotros, **SALAVARRÍA ARISTEGA KEVIN JAVIER, ROJANO LÁRRAGA CARLOS FERNANDO** en calidad de autores y titulares de los derechos morales y patrimoniales de la propuesta práctica de la alternativa de Titulación – Examen Complexivo: Investigación Documental, modalidad presencial, mediante el presente documento, libre y voluntariamente procedo a hacer entrega de la Cesión de Derecho del Autor de la propuesta practica realizado como requisito previo para la obtención de mi Título de Grado, como aporte a la Temática **MODELADO UML BASADO EN EL DISEÑO DE SOFTWARE** del Grupo de Investigación **TICS Y DESARROLLO DE SOFTWARE** de conformidad con el Art. 114 del Código Orgánico de la Economía Social de los Conocimientos, Creatividad e Innovación, concedo a favor de la Universidad Estatal de Milagro una licencia gratuita, intransferible y no exclusiva para el uso no comercial de la obra, con fines estrictamente académicos. Conservamos a nuestro favor todos los derechos de autor sobre la obra, establecidos en la normativa citada.

Así mismo, autorizamos a la Universidad Estatal de Milagro para que realice la digitalización y publicación de esta propuesta practica en el repositorio virtual, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior.

Los autores declaran que la obra objeto de la presente autorización es original en su forma de expresión y no infringe el derecho de autor de terceros, asumiendo la responsabilidad por cualquier reclamación que pudiera presentarse por esta causa y liberando a la Universidad de toda responsabilidad.

Milagro, a los 24 días del mes de Mayo de 2018

*Kevin Salavarría*

Salavarría Aristega Kevin Javier  
CI: 0953872140

*Carlos Rojano L*

Rojano Lárraga Carlos Fernando  
CI: 0942096470

## APROBACIÓN DEL TUTOR DE LA INVESTIGACIÓN DOCUMENTAL

Yo, **CORREA PERALTA MIRELLA AZUCENA** en mi calidad de tutor de la Investigación Documental como Propuesta práctica del Examen de grado o de fin de carrera (de carácter complejo), elaborado por el estudiante **SALAVARRÍA ARISTEGA KEVIN JAVIER, ROJANO LÁRRAGA CARLOS FERNANDO**, cuyo título es **ANÁLISIS DE LA IMPORTANCIA DEL MODELADO UML EN EL DISEÑO DE SOFTWARE**, que aporta a la Línea de Investigación Ciclo De Vida De Un Proyecto De Software, Metodologías Y Plataformas previo a la obtención del Grado Ingeniero en Sistemas Computacionales; considero que el mismo reúne los requisitos y méritos necesarios en el campo metodológico y epistemológico, para ser sometido a la evaluación por parte del tribunal calificador que se designe, por lo que lo **APRUEBO**, a fin de que el trabajo sea habilitado para continuar con el proceso de titulación de la alternativa de Examen de grado o de fin de carrera (de carácter complejo) de la Universidad Estatal de Milagro.

En la ciudad de Milagro, a los 24 días del mes de Mayo de 2018.



---

Correa Peralta Mirella Azucena  
Tutor  
C.I.: 0919615906

## APROBACIÓN DEL TRIBUNAL CALIFICADOR

El tribunal calificador constituido por:

MIRELLA AZUCENA CORREA PERALTA

LISSETT MARGARITA ARÉVALO GAMBOA

DENIS DARÍO MENDOZA CABRERA

Luego de realizar la revisión de la Investigación Documental como propuesta practica, previo a la obtención del título (o grado académico) de INGENIERO EN SISTEMAS COMPUTACIONALES presentado por el /la señor (a/ita) CARLOS FERNANDO ROJANO LARRAGA

Con el título: ANALISIS DE LA IMPORTANCIA DEL MODELADO UML EN EL DISEÑO DEL SOFTWARE

Otorga a la presente Investigación Documental como propuesta práctica, las siguientes calificaciones:

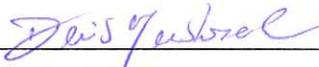
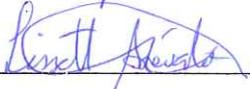
Investigación documental	[ 80 ]
Defensa oral	[ 18 ]
Total	[ 98 ]

Emite el siguiente veredicto: (aprobado/reprobado)

Aprobado

Fecha 24 de Mayo del 2018.

Para constancia de lo actuado firman:

	Nombres y Apellidos	Firma
Presidente	Mirella Azucena Correa Peralta	
Secretario /a	Denis Darío Mendoza Cabrera	
Integrante	Lissett Margarita Arévalo Gamboa	

## APROBACIÓN DEL TRIBUNAL CALIFICADOR

El tribunal calificador constituido por:

MIRELLA AZUCENA CORREA PERALTA

LISSETT MARGARITA ARÉVALO GAMBOA

DENIS DARÍO MENDOZA CABRERA

Luego de realizar la revisión de la Investigación Documental como propuesta practica, previo a la obtención del título (o grado académico) de INGENIERO EN SISTEMAS COMPUTACIONALES presentado por el /la señor (a/ita) KEVIN JAVIER SALAVARRIA ARISTEGA

Con el título: ANALISIS DE LA IMPORTANCIA DEL MODELADO UML EN EL DISEÑO DEL SOFTWARE

Otorga a la presente Investigación Documental como propuesta práctica, las siguientes calificaciones:

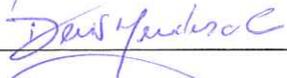
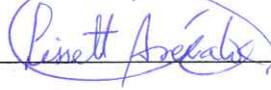
Investigación documental	[ 80 ]
Defensa oral	[ 18 ]
Total	[ 98 ]

Emite el siguiente veredicto: (aprobado/reprobado)

Aprobado

Fecha 24 de Mayo del 2018.

Para constancia de lo actuado firman:

Nombres y Apellidos		Firma
Presidente	Mirella Azucena Correa Peralta	
Secretario /a	Denis Darío Mendoza Cabrera	
Integrante	Lissett Margarita Arévalo Gamboa	

## **DEDICATORIA**

Dedico este trabajo a nuestro creador Dios, por darme la fortaleza y sabiduría que he necesitado para nunca desistir de mis sueños y metas. A mi Madre Elena María Aristega León y a mi padre Francisco Javier Salavarría Correa por ser mis ejemplos de lucha y perseverancia, por apoyarme siempre, estar en mis buenos y malos momentos.

A mi tutora Msc. Mirella Azucena Correa Peralta, por su paciencia y tiempo para la culminación de nuestros estudios profesionales y ayudar en el desarrollo de este trabajo.

**Salavarría Aristega Kevin Javier**

A Dios, por darme la oportunidad de llegar hasta este punto y haberme dado salud para finiquitar mis metas, brindarme bondad y amor.

A mis padres, por darme ese apoyo moral en todo instante, doy gracias por sus consejos, sus valores, por la alegría que me dan constantemente en la cual me ha permitido ser una persona de bien, pero más que nada, por su amor incondicional. A mi madre Marina que siempre ha estado ahí ayudándome, dándome consejos contándome sus experiencias, gracias a esas palabras pude llegar hasta aquí, mi padre Fernando que con su carácter fuerte me ayudó a tomar decisiones asertivas en el camino, siempre ahí dándome fuerzas para seguir adelante, a mis hermanos, familiares y amigos que supieron darme apoyo moral para seguir en pie, a mi novia Kerly que por sus ayuda incondicional pude seguir y terminar mis estudios.

A mi tutora Msc. Mirella Azucena Correa Peralta, por su gran apoyo y motivación para la culminación de nuestros estudios profesionales y por la elaboración de este documento.

**Carlos Fernando Rojano Lárraga**

## **AGRADECIMIENTO**

Este proyecto es resultado de esfuerzo y dedicación conjunto con los que conformamos el grupo de trabajo. Por eso agradezco a nuestro docente a lo largo de este tiempo quien ha puesto a prueba su capacidad y conocimiento para el desarrollo de este proyecto. A mi familia quien a lo largo de la vida me ha apoyado y motivado en mi formación académica.

**Salavarría Aristega Kevin Javier**

En primera instancia, agradezco a Dios por haberme dado, sabiduría, entendimiento e inteligencia, a mis padres por apoyarme en las buenas y en las malas; y el estar a mi lado ayudándome para mi es una satisfacción enorme.

Me gustaría agradecer a mi compañero de práctica en la UNEMI por su colaboración enorme para este documento, Particularmente me gustaría nombrar a mi tutora Correa Peralta Mirella Azucena.

Agradecerle por su cooperación y darle las gracias por todas las oportunidades que me ha dado durante la investigación, definitivamente me ha brindado todas las herramientas necesarias para completar mi trabajo documental.

Finalmente, mis amigos. Por haber estado ahí para apoyarme en los momentos difíciles.

¡Muchas gracias a todos!

**Carlos Fernando Rojano Lárraga**

## ÍNDICE GENERAL

DERECHOS DE AUTOR.....	II
APROBACIÓN DEL TUTOR DE LA INVESTIGACIÓN DOCUMENTAL .....	III
APROBACIÓN DEL TRIBUNAL CALIFICADOR .....	IV
DEDICATORIA.....	VI
AGRADECIMIENTO .....	VIII
ÍNDICE GENERAL.....	X
ÍNDICE DE FIGURAS .....	XI
RESUMEN .....	1
ABSTRACT .....	2
INTRODUCCIÓN.....	3
PROBLEMA DE INVESTIGACIÓN .....	4
MARCO TEÓRICO CONCEPTUAL.....	7
METODOLOGÍA.....	14
DESARROLLO DEL TEMA.....	16
CONCLUSIÓN .....	23
REFERENCIAS BIBLIOGRÁFICAS .....	24

## ÍNDICE DE FIGURAS

Figura 1. Elementos UML.....	8
Figura 2. Relaciones de UML.....	9
Figura 3. Ejemplo de un diagrama de estructural de clases UML.....	10
Figura 4. Metodología de la investigación. ....	15
Figura 5. Red general de coautoría que relaciona autores por colaboración a estudio UML en la base de datos de SCOPUS (2014-2017). ....	17
Figura 6. Red de coautoría con mayor cantidad de autores relacionados a estudio UML en la base de datos de SCOPUS (2014-2017). ....	18
Figura 7. Segunda red de coautoría con mayor cantidad de autores relacionados a estudio UML en la base de datos de SCOPUS (2014-2017). ....	19
Figura 8. Tercera red de coautoría con mayor cantidad de autores relacionados a estudio UML en la base de datos de SCOPUS (2014-2017). ....	20
Figura 9. Red de coautoría conformada por autores de publicaciones diferentes. ....	21
Figura 10. Red de co-ocurrencia de keywords. ....	22

# **TÍTULO DE LA INVESTIGACIÓN DOCUMENTAL: ANÁLISIS DE LA IMPORTANCIA DEL MODELADO UML EN EL DISEÑO DE SOFTWARE**

## **RESUMEN**

UML (Lenguaje Unificado de Modelado) permite modelar la estructura y diseño de un sistema, de esta forma se puede expresar de manera visual los elementos arquitectónicos del software, facilitando la conversión a código y disminuyendo errores. Por lo tanto el objetivo de esta investigación es analizar las publicaciones relacionadas a UML en la base de datos Scopus obteniendo como resultado varios grupos de autores relacionados por su colaboración y keywords más utilizadas en los artículos y su relación potencial que permitirán dar seguimiento a las publicaciones. Para el desarrollo de este análisis bibliométrico se realizó la búsqueda de información limitándola entre los años 2014 - 2017 y a las áreas de Computer Science y Engineering, dando como resultado 20 publicaciones las cuales fueron analizadas con la herramienta VOSviewer, de la cual se obtuvo que Atkinson C, Gerbig R, Tunjic C.V, Fritzsche M son los autores de 2 publicaciones que se encuentran relacionados, Ujhelyi Z, Bergmann G, Hegedüs Á, Horváth Á, Izsó B, Ráth I, Szatmári Z, Varró D forman la mayor cantidad de autores relacionados con un total 8 autores, la segunda mayor cantidad de autores relacionados contiene 7 autores, además las keywords relacionadas de las publicaciones científicas son: unified modeling language, modeling languages, computational linguistics y software engineering. Para la elaboración de este trabajo se utilizó la metodología documental, descriptiva que se usa para describir fenómenos que se esté investigando y se requiera analizar, donde se define las características de estos. El presente trabajo también presenta información que respalde la importancia de modelar con UML en el diseño del software como su empleo en el desarrollo de ontologías que facilitan el diseño del software, para modelar sistemas dinámicos, empleado por la ingeniería inversa como herramienta para obtener un modelo de un sistema ya existente para su posterior modificación.

**PALABRAS CLAVE:** Lenguaje de modelado, Ingeniería de software, Lingüística computacional, Lenguaje unificado de modelado (UML)

**TITLE OF DOCUMENTARY RESEARCH: ANALYSIS OF THE IMPORTANCE OF THE UML MODELING IN THE DESIGN OF SOFTWARE**

**ABSTRACT**

UML (Unified Modeling Language) allows you to model the structure and design of a system, in this way you can express the visual architectural elements of the software, facilitating the conversion to code and reducing errors. Therefore the objective of this research is to analyze the publications related to UML in the Scopus database, obtaining as a result several groups of authors related by their collaboration and keywords most used in the articles and their potential relationship that will allow to follow up the publications. For the development of this bibliometric analysis, the information search was carried out limiting it between the years 2014 - 2017 and the areas of Computer Science and Engineering, resulting in 20 publications which were analyzed with the VOSviewer tool, from which it was obtained that Atkinson C, Gerbig R, Tunjic CV, Fritzsche M are the authors of 2 publications that are related, Ujhelyi Z, Bergmann G, Hegedüs Á, Horváth Á, Izsó B, Ráth I, Szatmári Z, Váró D form the largest number of authors related to a total of 8 authors, the second largest number of related authors contains 7 authors, in addition the related keywords of the scientific publications are: unified modeling language, modeling languages, computational linguistics and software engineering. For the elaboration of this work we used the documentary, descriptive methodology that is used to describe phenomena that are being investigated and that need to be analyzed, where the characteristics of these are defined. The present work also presents information that supports the importance of modeling with UML in software design as its use in the development of ontologies that facilitate software design, to model dynamic systems, used by reverse engineering as a tool to obtain a model of an existing system for its subsequent modification.

**KEY WORDS:** Unified modeling language, Modeling languages, Computational linguistics y Software engineering,

## INTRODUCCIÓN

La ingeniería de software como disciplina de la ingeniería, brinda un conjunto de métodos, técnicas, herramientas y mediante una serie de actividades convierte los requisitos de los clientes en el software. El ciclo de vida del software consta de varios procedimientos, como es el diseño del software, donde se modela los requisitos de la arquitectura del software a desarrollarse, así como un arquitecto elabora un modelo para la construcción de un edificio, para esto se ayuda de UML.

UML (Lenguaje Unificado de Modelado) UML es un lenguaje estándar, que modela, muestra, elabora y documenta la estructura y diseño de un sistema, para modelar se debe conocer cómo y bajo qué reglas se acoplan los elementos, relaciones y diagramas que proporciona UML (Romero & Escalada, 2016).

Los diagramas UML utilizados para el diseño de un sistema suelen ser: diagramas de clases, objetos, colaboración, secuencia, de estado, actividad, componentes y de ejecución., además se pueden generar híbridos de estos, extenderlos y organizarlos, y su empleo va más allá de la ingeniería de software, llegando al punto de la reutilización de los modelos.

La presente investigación tiene como objetivo la identificación de información que respalde la importancia de modelado UML en el diseño del software, además de analizar los datos bibliométricos de los artículos científico que se obtuvieron de la base de datos de Scopus.

El presente trabajo está desarrollado en 5 capítulos, inicia con el capítulo 1 que contiene el planteamiento del problema donde se hace referencia a la situación en el que se enmarca el trabajo, los objetivos que son el propósito del trabajo y la justificación donde se describe las razones que condujeron a la selección del tema. Capítulo 2 comprende los fundamentos teóricos que permitan un entendimiento de la estructura y dinámica del objeto de estudio. Capítulo 3 contiene la metodología a seguir del investigador donde se realiza un análisis descriptivo. Capítulo 4 comprende la selección de información, el análisis del material y de su elaboración y por último el Capítulo 5 contiene las conclusiones del trabajo de investigación.

# CAPÍTULO 1

## PROBLEMA DE INVESTIGACIÓN

La implementación del lenguaje UML en la fase de diseño del ciclo de vida de desarrollo de software es un tema aún abierto, esto se debe a que el uso de este lenguaje para modelar no tiene límites, este adapta a todos los campos de la ingeniería, además se crean extensiones o perfiles que ayudan a solucionar problemas del modelado.

Los documentos científicos que se encuentran en las bases de datos como Scopus, proporcionan información que demuestran la importancia de modelar con UML, mostrando soluciones a algunos problemas, como:

- Una forma en que los analistas y diseñadores de los sistemas móviles puedan modelar una característica claves de estos sistemas como lo es la movilidad.
- El desarrollo de ontologías con el fin de mejorar las distintas técnicas relacionadas con el diseño del software.
- Modelar sistemas dinámicos que necesitan ser analizados para identificar comportamientos anormales.
- Las herramientas de modelado que pueda extender la meta-dato, usando un lenguaje que tenga integrado un mecanismo de extensión
- Se necesita modelar funciones para el diseño de artefactos, sistemas empresariales y sistemas naturales.

En la actualidad un problema que dificulta la búsqueda de información útil es la amplia cantidad de publicaciones en las bases de datos científicas, para lo cual un análisis bibliométrico sirve como herramienta para dar un seguimiento a las publicaciones de interés.

En el presente trabajo se enfoca en un análisis bibliométrico con énfasis en la relación de colaboración de los autores y las keywords más utilizadas en las publicaciones, que son los datos bibliométricos más adecuados a analizar y de esta manera dar seguimiento a estas publicaciones.

Para el desarrollo del análisis bibliométrico se utilizó el programa VOSviewer para crear redes de coautoría y co-ocurrencia, que facilitará dicho análisis, esto se debe a que las redes de coautoría la cual permite visualizar las relaciones que hay entre actores de publicaciones y las redes de co-ocurrencia permiten visualizar las relaciones potenciales de términos como las keywords de los artículos.

Como resultado se obtuvo una red general que muestra la relación de autores por colaboración y las palabras claves más utilizadas por los autores en sus respectivos artículos, las cuales son: counified modeling language, modeling languages, computational linguistics y software engineering, que permitirán realizar un seguimiento de las publicaciones.

## **Objetivos**

### **Objetivo general:**

- Analizar las publicaciones relacionadas a la importancia del modelado UML para el diseño de software.

### **Objetivos específicos:**

- Identificar la importancia del modelado UML en el diseño del software a partir de artículos publicados en el 2014 al 2017.
- Evaluar estudios relacionados a UML en la base de datos de Scopus del 2014 al 2017.
- Elaborar gráfico representativo de la información bibliográfica de estudios relacionados a UML.

## **Justificación**

El presente trabajo tiene como finalidad, la identificación de información que respalde la importancia del modelado UML en el diseño del software a partir de los artículos extraídos de la base de datos Scopus entre los años 2014 y 2017 y el respectivo análisis bibliométrico, que mediante la herramienta VOSviewer se obtendrá redes bibliométricas como las redes de coautoría y de co-ocurrencia, que comparen datos bibliográficos de las publicaciones, en este caso datos de los autores y keywords.

Aunque UML es un lenguaje estándar para el modelado utilizado en el diseño del software, el conocimiento de los desarrolladores de acerca de porque es importante modelar con dicho lenguaje y cuál es su alcance, suele ser escaso. Por lo que es útil la recopilación de información que demuestre lo establecido.

UML va más allá de la ingeniería de software, por ejemplo implementándose en el desarrollo de ontologías que faciliten el diseño del software o en el campo de la reingeniería de software, que por medio de la ingeniería inversa se obtiene un modelo en UML de un sistema ya existente.

Es necesario un análisis bibliométrico a los artículos extraídos de Scopus, esto se debe a la basta cantidad de publicaciones en las bases de datos científicas que dificultan la búsqueda de información útil, por lo que ayudará a dar un seguimiento a las publicaciones científicas relacionadas al tema consultado. El trabajo es importante pues se encuentra apalancado con la línea de investigación TICS y desarrollo de software de la Universidad Estatal de Milagro.

## **CAPÍTULO 2**

### **MARCO TEÓRICO CONCEPTUAL**

#### **Lenguaje de modelado unificado (UML)**

UML (Lenguaje Unificado de Modelado) es un lenguaje estándar para modelar, documentar la estructura y diseño de un sistema que acoplan elementos, objetos, relaciones y diagramas conceptuales al describir y especificar la estructura interna de los elementos arquitectónicos de software (Romero & Escalada, 2016), además permite la interpretación estructural y funcional de estos elementos, donde los actores aceptan como pauta para llegar a una interpretación del mismo (Arias Barragán, Fernando García, & Rivas Trujillo, 2013).

Es posible generar híbridos de los diagramas, extenderlos y organizarlos (Arias Barragan, 2016), para plasmar de manera gráfica un software brindando a los programadores una visión gráfica y facilitar el proceso de codificación y reducir errores a partir de los requisitos hasta el modelado de actividades de planificación y gestión de versiones (Ciencias, Ambiente, Río, La, & Periodo, 2015) con la finalidad de establecer una comunicación estándar para alcanzar un sistema más acoplable a las estructuras estáticas y dinámicas de los sistemas (Rodas, 2016).

#### **Elementos gráficos**

Los componentes básicos de UML son elementos, relaciones y diagramas.

#### **Elementos UML**

Se determinan como abstracciones, consideradas como unidades fundamentales al momento de modelar, incluye 4 elementos como se muestra en la Figura 1: estructurales, comportamiento, de agrupación y de notación (Alberto & Salvatierra, 2015).

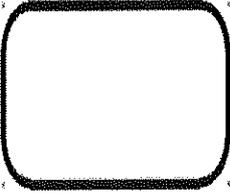
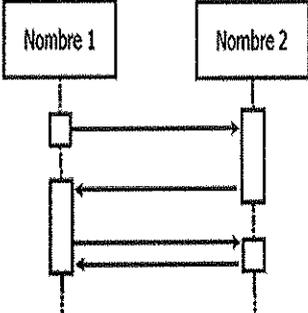
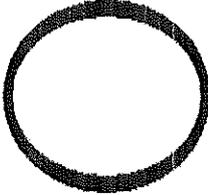
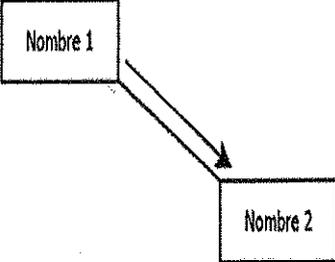
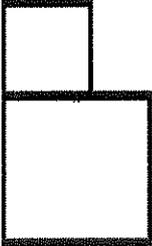
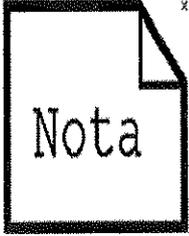
ELEMENTOS ESTRUCTURALES		ELEMENTOS DE COMPORTAMIENTO	
Clase		Estado	
Interfaz		Secuencia	
Caso de uso		Colaboración	
	AGRUPACIÓN		ANOTACIÓN
Paquete		Nota	

Figura 1. Elementos UML.

Adaptado de (Condori, 2016).

## Relaciones UML

Las relaciones de elementos en UML son: relaciones de generalización, de dependencia, de realización de asociación, de agregación y de composición, como se muestra en la Figura 2.

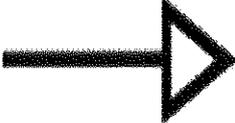
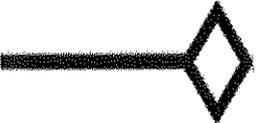
<i>Generalización</i>	
<i>Dependencia</i>	
<i>Realización</i>	
<i>Asociación</i>	
<i>Agregación</i>	
<i>Composición</i>	

Figura 2.Relaciones de UML.

Elaborado por los autores: Kevin Salavarría, Carlos Rojano.

### **Diagramas UML**

Los diagramas presentan de manera gráfica el sistema, estos se clasifican en: Diagramas Estructurales y Diagramas de comportamiento. En la Figura 3 se muestra un ejemplo de un diagrama de clases UML.

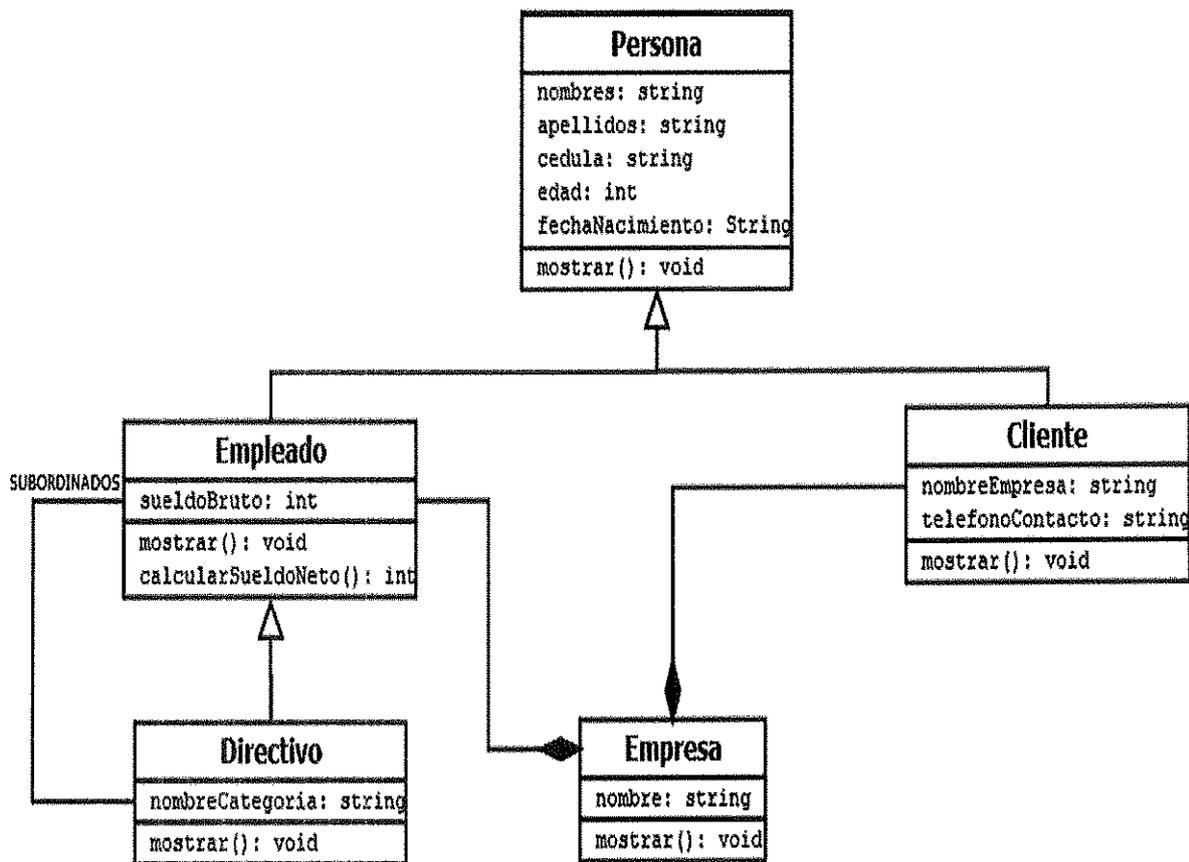


Figura 3. Ejemplo de un diagrama de estructural de clases UML.

Elaborado por los autores: Kevin Salavarría, Carlos Rojano.

## Fundamentos Teóricos

La demanda de los sistemas móviles va en aumento, por lo que es necesario que haya una forma en que los analistas y diseñadores de los sistemas móviles puedan especificar las características de los sistemas interactivos que los usuarios necesitan. El lenguaje unificado de modelado (UML), que aunque ofrece un conjunto de diagramas para modelar eficazmente el comportamiento de los sistemas, estos no funcionan de igual manera para modelar la movilidad. Una extensión de UML, m-TOPP-UML, muestra los aspectos de la movilidad en vistas y diagramas. Se implementó por primera vez la extensión m-TOPP-UML al seguimiento móvil de los sistemas de progreso del paciente, donde dio como resultado diagramas m-TOPP-UML que mostraron un análisis y diseño del sistema móvil (Sciences, Scientific, & Corp, 2014).

El desarrollo de ontologías para la ingeniería de software se lo realiza con el fin mejorar las distintas técnicas relacionadas con el diseño del software, una gran cantidad de técnicas de desarrollo de ontología usan el lenguaje UML, esto se debe a que este lenguaje puede alcanzar el mismo nivel de calidad que brinda otros lenguajes para el desarrollo de ontologías, además el uso de UML es óptimo para la conceptualización de ontologías livianas (Reynares, Laura, & Rosa, 2014).

Un modelo conceptual detalla el dominio que se analiza de forma autónoma a la tecnología, utilizando términos de dominio, consideraciones y ciertas reglas para elaborar. El modelo de dominio se diseña en UML usando los paradigmas de análisis orientados a objetos con lo que el modelo adopta y promueve la reutilización y el intercambio de los elementos. Para diseñar un software se puede tomar como base el modelo de dominio, esto se debe a que proporciona un entendimiento del entorno en el que el sistema va a funcionar (Chamanara, J; Konig-Ries, 2014).

Los sistemas dinámicos necesitan ser analizados para identificar comportamientos anormales, normalmente UML es utilizado para modelar sistemas dinámicos mediante diagramas de estados o máquinas de estados como suelen ser llamados, pero estos no pueden describir cómo se comportan realmente los sistemas, para lo cual los diagramas de estados de UML son la base para que se pueda aplicar un formalismo DEVS (especificación de sistema de eventos discretos), de esta manera se observa que tan importante es modelar con UML el diseño del software, sus aspectos dinámicos (Al-Fedaghi, 2014).

Las herramientas de modelado permiten observar vistas personalizadas de los sistemas empresariales a distintas partes interesadas, por lo tanto las herramientas de modelado deberían permitir ampliar y mejorar tanto los servicios e idioma. Esto es muy imposible ya que la meta-modelo encargada del idioma y servicio no es apto a extensiones. Aun así se puede obtener soluciones para extender la meta-modelo, estas extensiones suelen venir incorporados en lenguajes de modelado, pero otras deben implementar herramientas externas. Por lo que es importante usar UML como estándar de modelado, el cual viene integrada con un mecanismo de extensión en la notación UML 2, de esta manera la

herramienta de modelado no tendrá problemas de capacidad, facilitando el modelado de diseño del software que se encuentre en desarrollo (Atkinson, Gerbig, & Fritzsche, 2015).

Las herramientas de modelado multinivel ha tenido un gran acoge y su uso aumenta cada día, esto se debe a que la conversión de modelos sencillos a modelos multinivel permite mejorar la interoperabilidad, por lo que (Atkinson, Gerbig, & Vjekoslav, 2013) propusieron un enfoque para que lenguajes de conversión sean multinivel y que además se pueda establecer el contenido del modelo multinivel en el contenido de dos niveles tradicionales y viceversa, el enfoque usará los modelos de dos niveles elaborados con UML, este como lenguaje estándar permite limitar la complejidad de la conversión.

En la actualidad la aplicación de métodos de ingeniería inversa a sistemas de software son cada vez más común, estos métodos permiten reutilizar la infraestructura de un sistema existente para el desarrollo de un sistema nuevo. Aplicando el proceso de ingeniería inversa se obtiene información o un diseño, descomponiendo o reconstruyendo el código fuente o el sistema existente a diagramas UML. Un enfoque de ingeniería inversa puede convertir un documento en formato de relleno de formulario en un diagrama de clases UML, este enfoque de ingeniera inversa consta con las fases de captura, normalización, mapeo y el proceso de traducción que es la fase en la que se elaboran los diagramas de clase. La ingeniería inversa brinda información en notación UML, que es utilizada para el análisis de un sistema existente por la reingeniería de software, que es la que se encargara en modificar un sistema existente en uno más sostenible (Muhairat & Abdelqader, 2016).

Para elaborar un sistema web reutilizables tiene que modelar un MDWE reutilizable (Ingeniería Web Impulsada por Modelos), con el cual se modela el análisis, contenido, navegación y presentación del sistema web, luego estos modelos serán convertidos en códigos de implementación. Para desarrollar un MDWE reutilizable se usa UML y UWE (Ingeniería Web Basada en UML) en el modelado, los que permiten declarar artefactos MDWE en la fase de análisis y diseño, estos artefactos se almacenaron con su código de implementación y casos de prueba en un herramienta de caso que elaboraron (Elminir, Riad, & El-halawany, s. f.).

El modelado de funciones es fundamental en el diseño de artefactos, sistemas empresariales y sistemas naturales, por lo que para modelar funciones con UML, se lo debe realizar

mediante un perfil UML llamado lenguaje de modelado de funciones (FUEL), este perfil es el resultado de un análisis donde se determinaron las restricciones que tienen varios enfoques para modelar funciones con UML. FUEL permite modelar funciones, como interactúan entre sí y abstraer o atribuir funciones, fue elaborado principalmente para su empleo en el campo de la bioinformática, pero es idóneo para modelar sistemas de dominios cruzados, también permite el desarrollo de modelos nuevos, también colabora en la modificación de estructuras de modelos existentes (Burek, Loebe, & Herre, 2016).

El lenguaje unificado de modelado (UML) en el desarrollo del software y de aplicaciones web que usan XML para el intercambio estructurado de datos. Se volvió una parte fundamental en varias de sus fases independientes de la metodología que se use. Para describir la estructura de los contenidos de los documentos XML es necesario que se realicen modelos conceptuales que muestren la estructura de datos y relaciones de los esquemas XML. Para desarrollar los modelos conceptuales se usa los diagramas UML, donde se transforma los elementos y atributos XML a clases con sus características que facilita el entendimiento de los esquemas XML (Fouad & Mohamed, 2016).

La Arquitectura dirigida por modelos (MDA) propone utilizar modelos para desarrollar un software separando la estructura del diseño y de la plataforma de desarrollo. MDA realiza la conversión de modelos, del nivel CIM (Modelo Independiente de Cómputo) al nivel PIM (Modelo Independiente de Plataforma). Normalmente se usa UML como la mejor opción para detallar el nivel PIM. Un ejemplo es la conversión de modelos comerciales orientados a servicios, a modelos de diseño web donde se usa SoaML para modelar el nivel CIM y UML para modelar el nivel PIM, el resultado es una serie de modelos de diseños web de modelos comerciales orientados a servicios basados en UML (Rhazali, Y; Hadi, Y; Mouloudi, 2016).

Para modelar servicios web basados en WSDL (Lenguaje de descripción de servicios web), que es el utilizado por W3C para describir y cómo funcionan los servicios web, (Marcos, Esperanza; De Castro, M Valeria; Vela S, 2004), establecieron una extensión que permita modelar un servicio web usando diagramas de clases UML, esta reduce la complejidad de la conversión de diagramas de clases en código WSDL de manera automática.

## **CAPÍTULO 3**

### **METODOLOGÍA**

#### **Metodología documental**

Es un procedimiento de ámbito científico, donde se recolecta, estructura, analiza e interpreta información de un tema definido, este tipo de investigación conduce al desarrollo de conocimiento (Alfonzo, 1994).

Utilizando la metodología documental, se procedió a la recolección de información del tema de investigación procedente a partir de artículos científicos, en donde se indagó, analizó e interpretó información útil que permitió el desarrollo de conocimiento.

#### **Metodología descriptiva**

Sirve para describir cualquier tipo de fenómeno que se esté tratando y que se quiera analizar, determinando particularidades fundamentales de dichos fenómenos. La metodología descriptiva es apropiada para expresar con exactitud las dimensiones, donde ser determinara que se medirá y sobre que se recolectara la información por parte del investigador. (Hernandez Sampieri, Roberto; Fernández Collado, Carlos; Baptista Lucio, 2006).

#### **Análisis Bibliométrico**

El análisis bibliométrico permite realizar un análisis cuantitativo de todas las fuentes científicas, en el que se tienen en cuenta los elementos como autores, palabras claves, etc., utilizando tecnología para expresar sus resultados, con el fin de evaluar la producción científica de diferentes bases de datos (Solano Lopez, 2009)

Para el desarrollo del presente trabajo se empleó los siguientes pasos: definición del tema, identificación de las palabras claves, búsqueda de información, selección de la fuente de información, obtención de un repositorio de datos de Scopus, analizar las fuentes

bibliográficas, utilizar el programa VOSviewer para crear redes de coautoría y co-ocurrencia, desarrollo del tema y al final una conclusión, como se observa en la Figura 4.

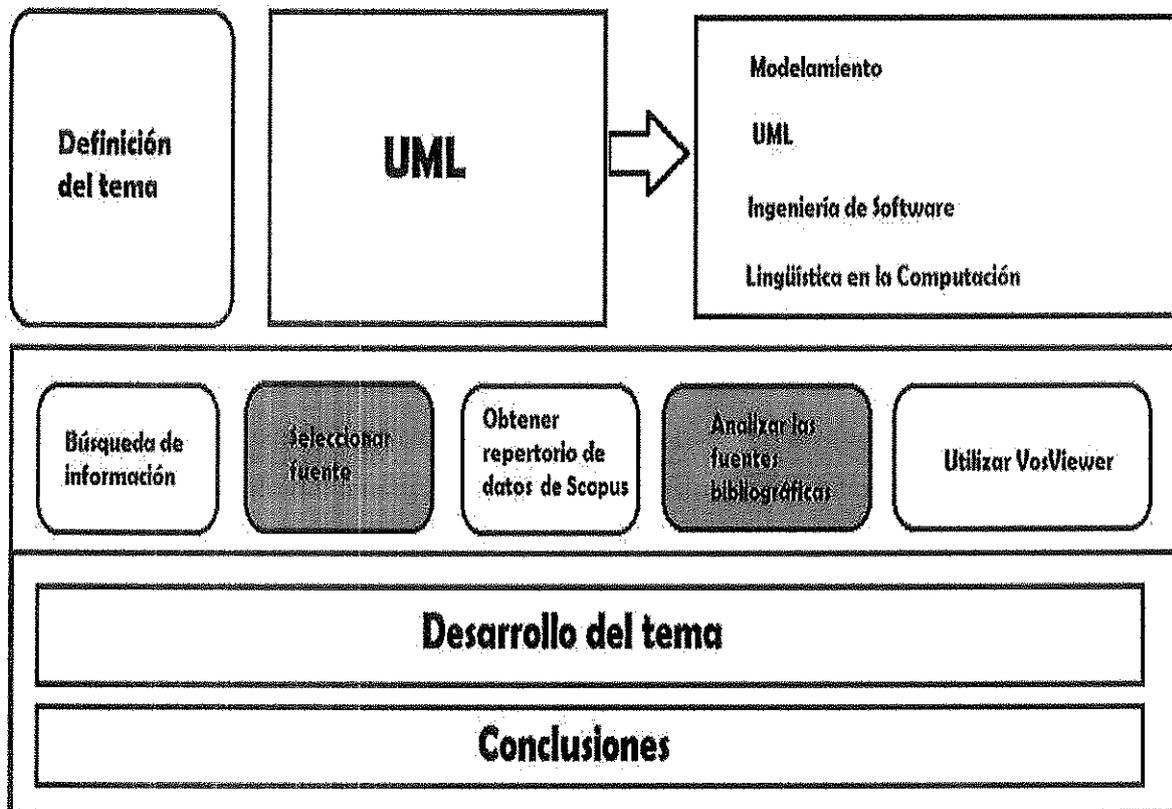


Figura 4. Metodología de la investigación.

Fuente: Elaborado por los autores: Kevin Salavarría, Carlos Rojano.

## CAPÍTULO 4

### DESARROLLO DEL TEMA

La búsqueda de información relacionada al tema, la importancia del modelado UML en el diseño del software, se desarrolló en la base de datos Scopus (<https://www.scopus.com>) limitando la información entre los años 2014 al 2017 y a las áreas de Computer Science y Engineering, de la cual se obtuvo un repositorio de datos de 20 publicaciones científicas, que mediante el programa VOSviewer se elabora redes de coautoría de autores y de co-ocurrencia de los keywords para su respectivo análisis.

El programa VOSviewer permite elaborar redes bibliométricas como las redes de citas, relaciones de coautoría, acoplamiento bibliográfico y co-ocurrencia. (VOSviewer, 2018).

Las redes de coautoría son un grupo de relaciones que permite agrupar individuos, el análisis de esta red permite visualizar las relaciones que hay entre actores de publicaciones y las redes de co-ocurrencia permiten visualizar las relaciones potenciales de términos, que se encuentren en un material escrito (González Alcaide, G; Valderrama Zuriá, JC; Aleixandre Benavent, R; Alonso Arroyo, A; de Granda Orive, J I; Villanueva Serrano, 2006).

En la Figura 5 se observa una red general de coautoría, que muestra la relación de colaboración de los autores de las publicaciones del repositorio de datos obtenido de Scopus, en dicha red se visualiza 19 grupos de autores relacionados.

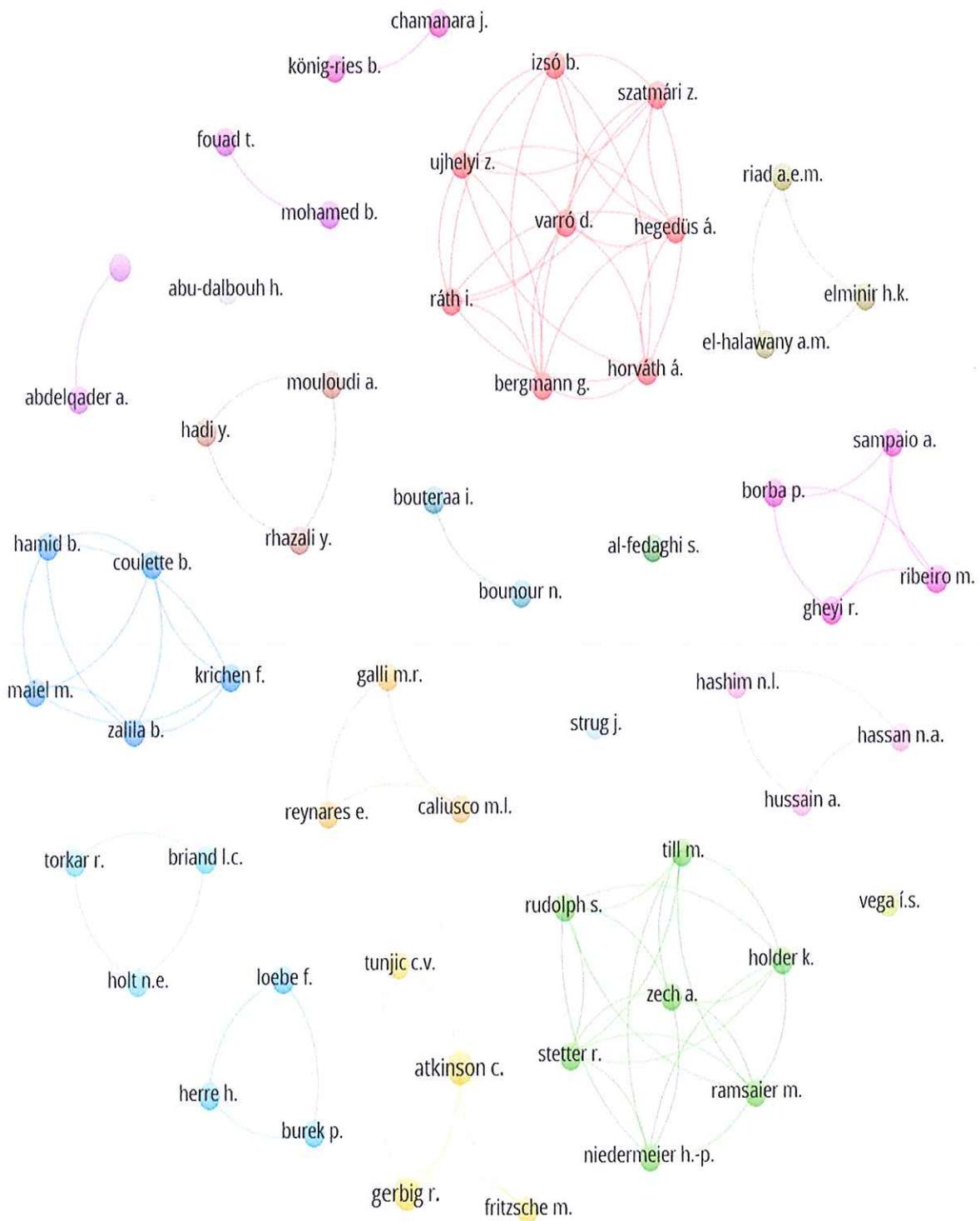


Figura 1. Red general de coautoría que relaciona autores por colaboración a estudio UML en la base de datos de SCOPUS (2014-2017).

Ujhelyi Z, Bergmann G, Hegedüs Á, Horváth Á, Izsó B, Ráth I, Szatmári Z, Varró D son los 8 miembros que conforman la red de coautoría con el mayor número de autores relacionados por su colaboración, donde todos ellos trabajaron en una sola publicación como se muestra en la Figura 6.

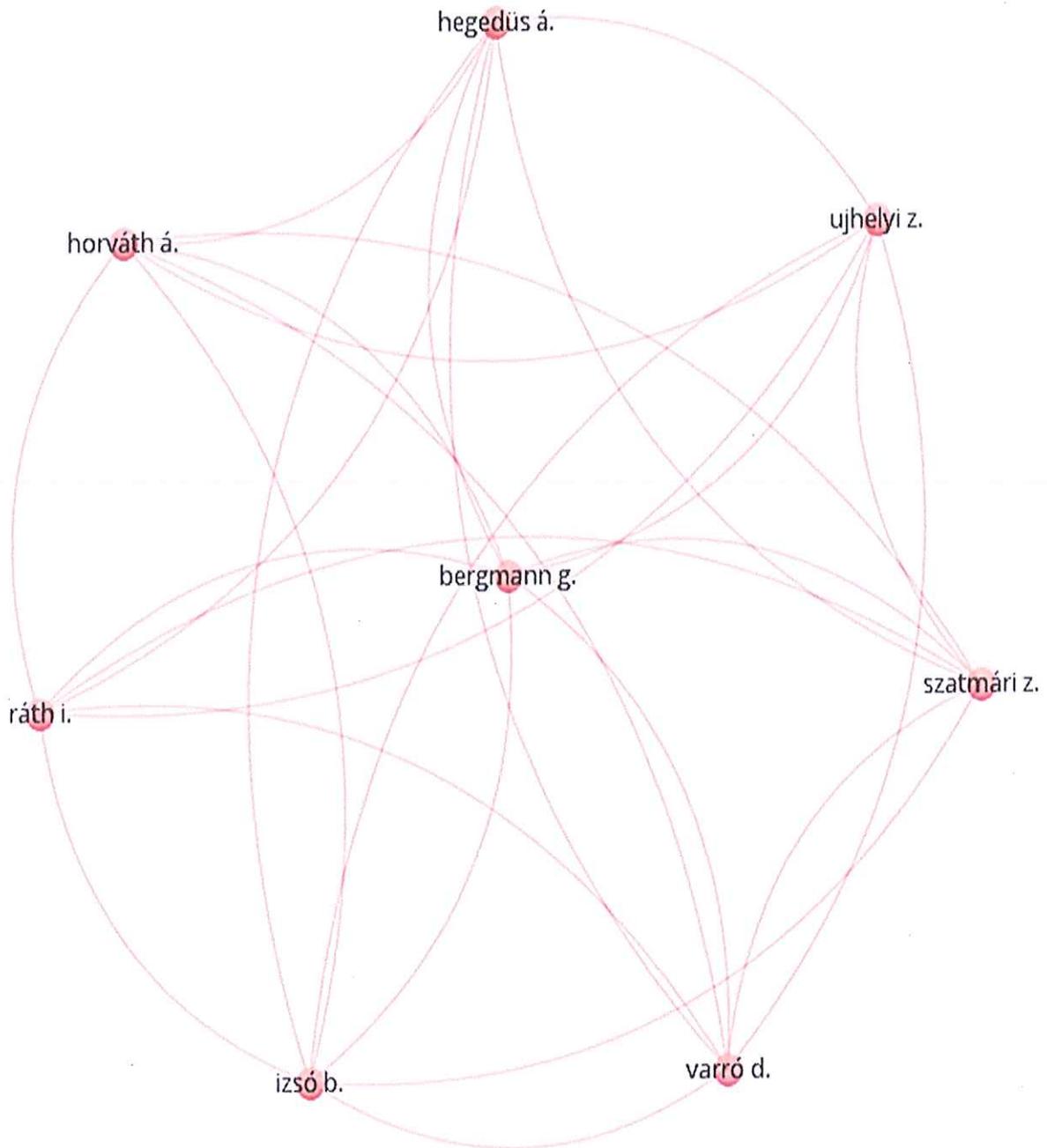


Figura 1. Red de coautoría con mayor cantidad de autores relacionados a estudio UML en la base de datos de SCOPUS (2014-2017).

Holder K, Zech A, Ramsaier M, Stetter R, Niedermeier H.P, Rudolph S., Till M. son los 7 miembros que conforman la segunda mayor red de coautoría donde se relacionan los autores por su colaboración, todos ellos trabajaron en una sola publicación, como se muestra en la Figura 7.

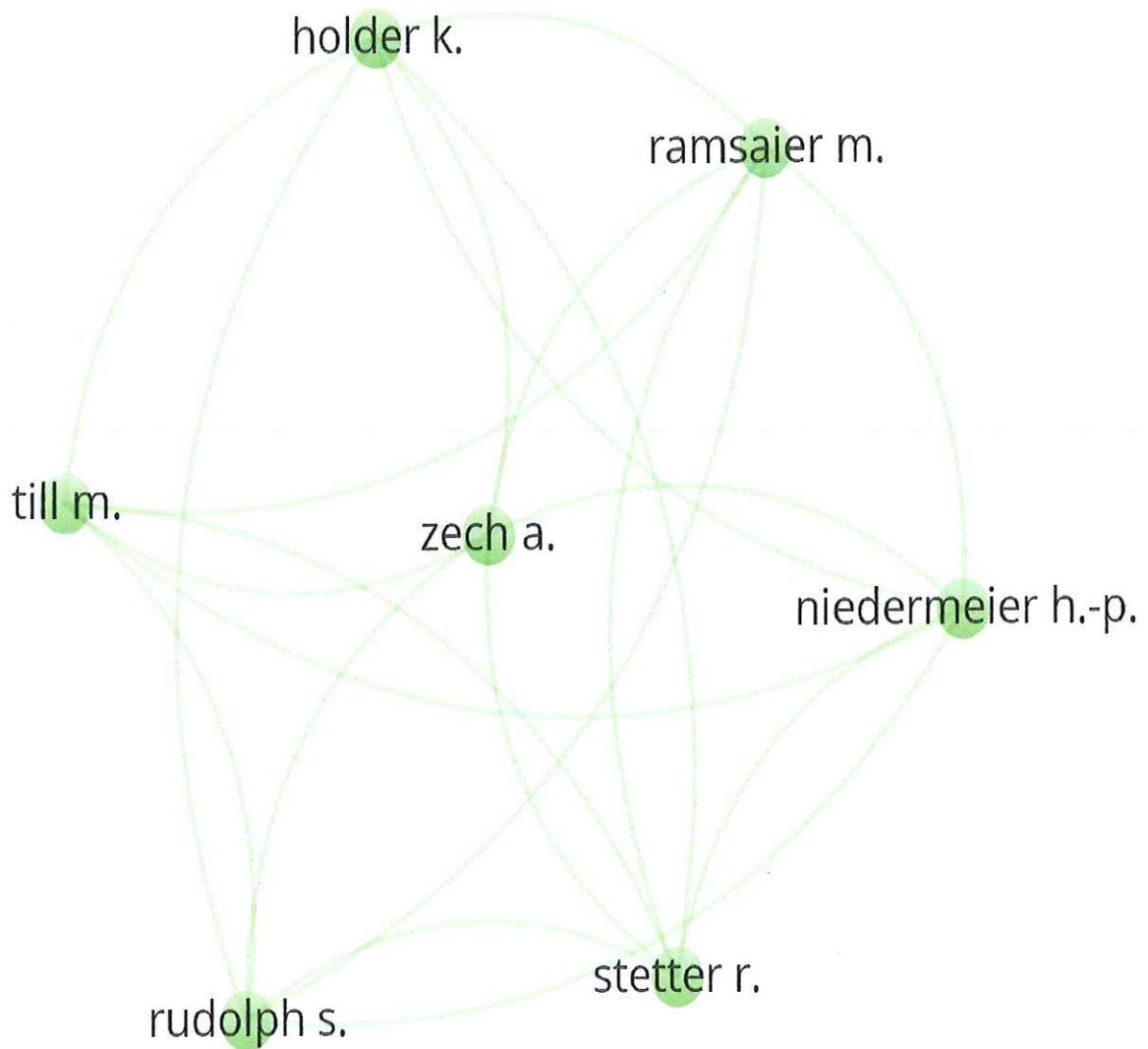


Figura 1. Segunda red de coautoría con mayor cantidad de autores relacionados a estudio UML en la base de datos de SCOPUS (2014-2017).

Krichen F, Hamid B, Zalila B, Jmaiel M, Coulette B son los 5 miembros que conforman la tercera mayor red de coautoría donde se relacionan los autores por su colaboración, todos estos trabajaron en una sola publicación, como se muestra en la Figura 8.

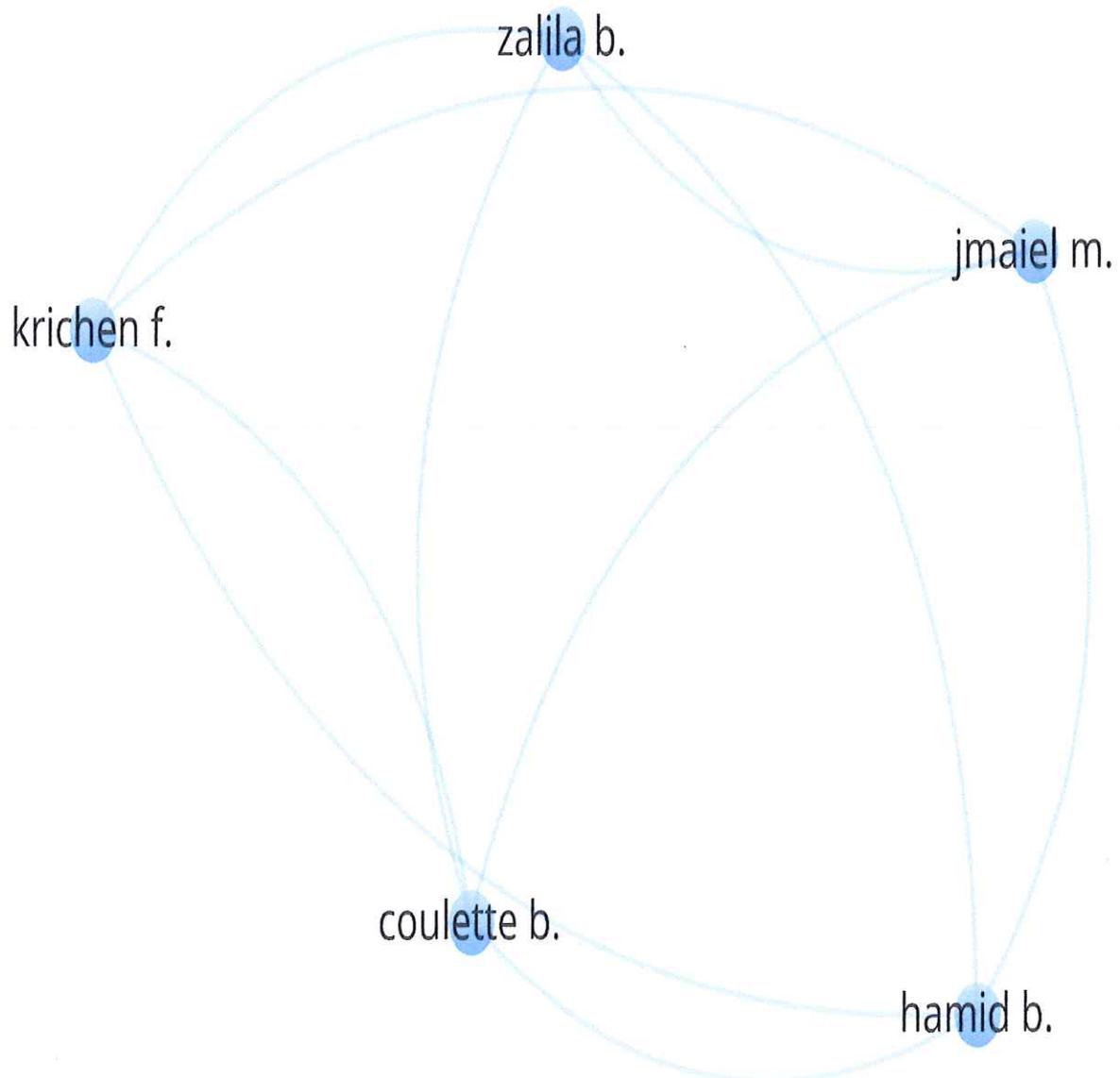


Figura 2. Tercera red de coautoría con mayor cantidad de autores relacionados a estudio UML en la base de datos de SCOPUS (2014-2017).

Como se muestra en la Figura 9; Atkinson C, Gerbig R, Tunjic C.V, Fritzsche M conforman la única red de coautoría donde los autores de 2 publicaciones se encuentran relacionados por su colaboración, en la primera publicación Atkinson C y Gerbig R trabajaron con Tunjic C.V y en la segunda publicación Atkinson C y Gerbig R trabajaron con Fritzsche M.

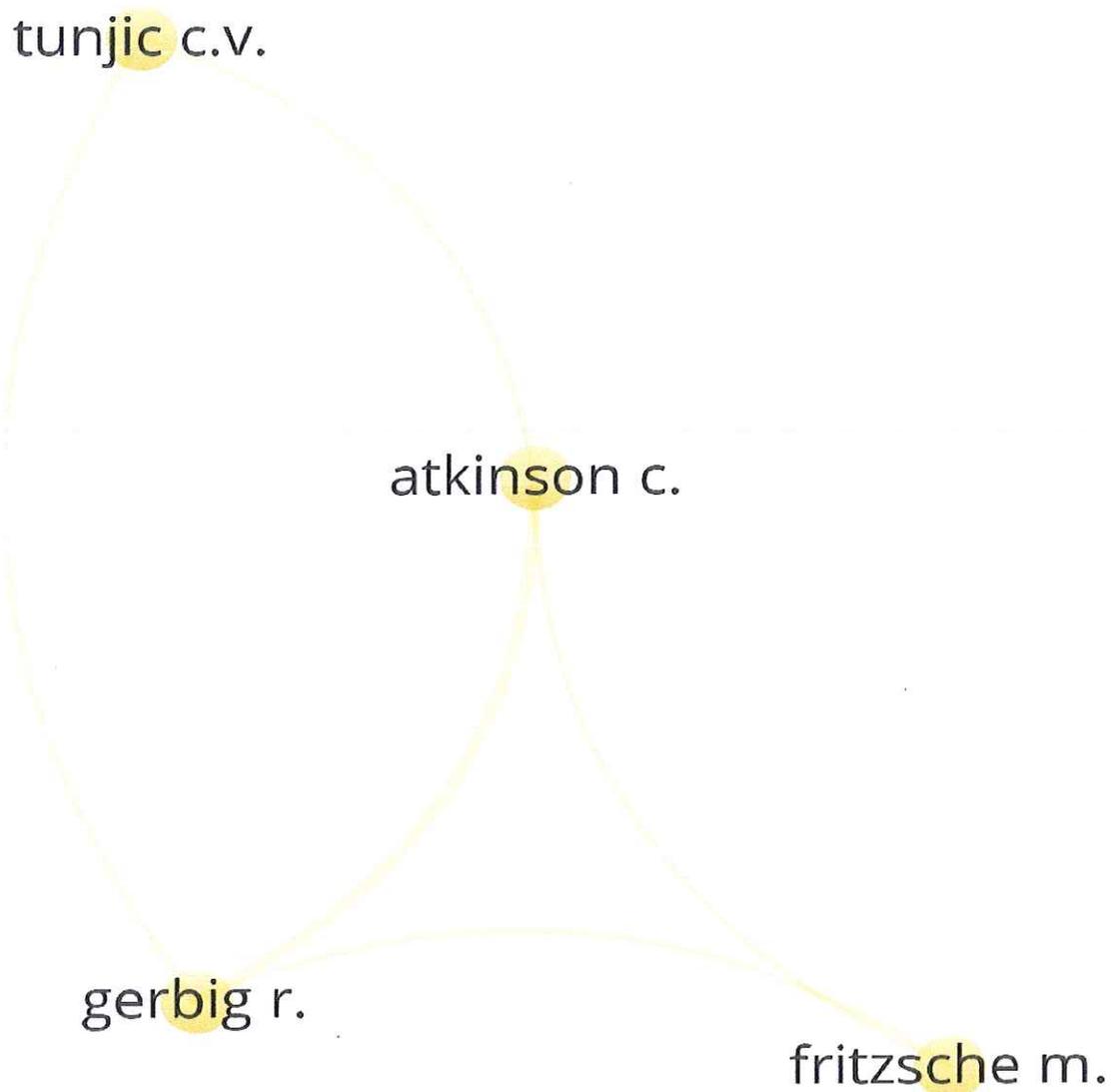


Figura 1. Red de coautoría conformada por autores de publicaciones diferentes.

Mediante el programa VOSviewer se elaboró una red de co-ocurrencia donde se tomó en cuenta las keywords de las publicaciones obtenidas de Scopus, en la cual se muestra las relaciones potenciales de los términos que más son utilizados por los autores, estos son: counified modeling language, modeling languages, computational linguistics y software engineering, como se muestra en la Figura 10.

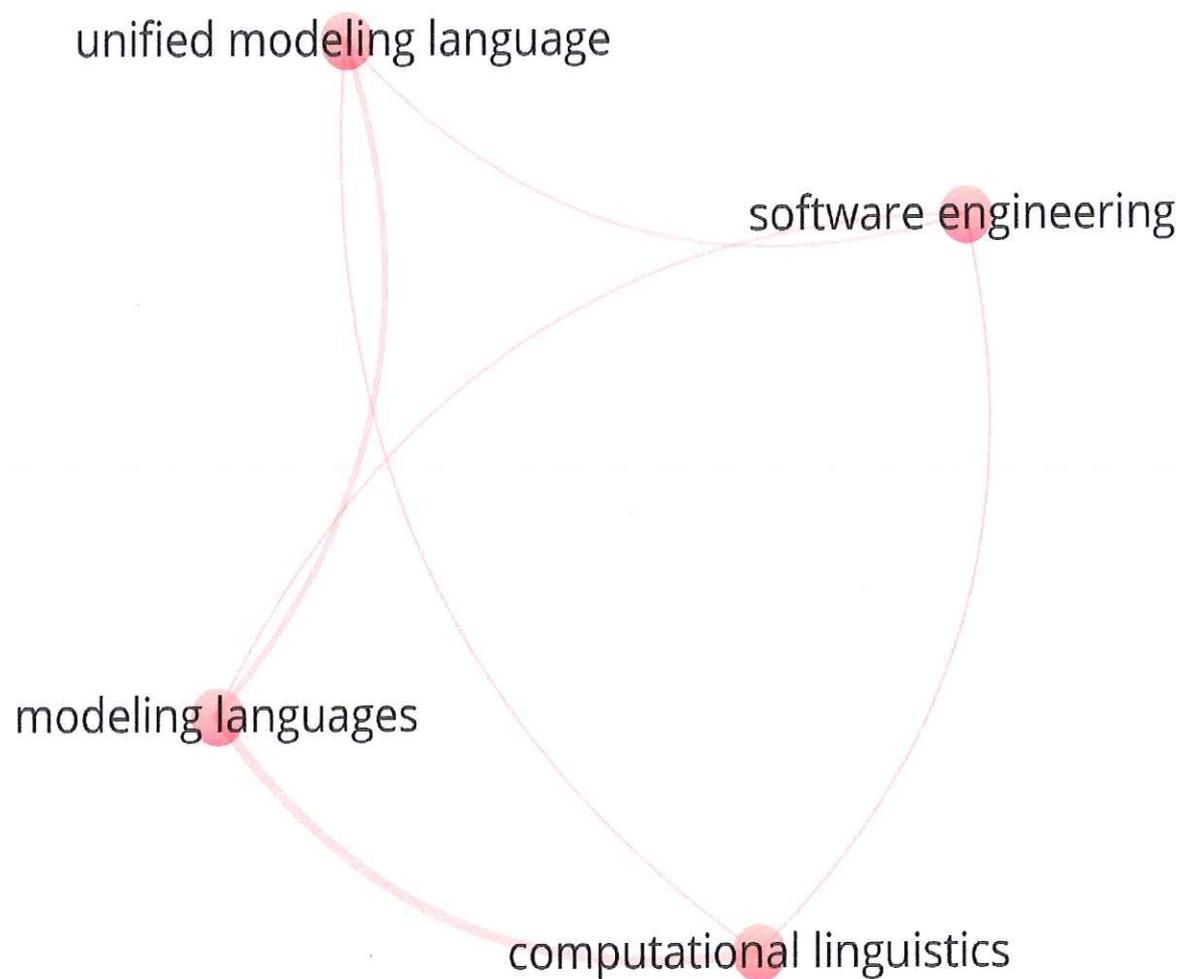


Figura 2. Red de co-ocurrencia de keywords.

## CAPÍTULO 5

### CONCLUSIÓN

Como conclusión se entiende que el lenguaje unificado de modelado (UML), permite modelar la estructura y diseño en muchas fases del ciclo de vida del desarrollo de un software. UML brinda un conjunto de diagramas, donde se generan híbridos de estos, además de extenderlos y realizar perfiles para modelar como son las funciones. Con UML en la fase de diseño del software, convierte los requisitos del cliente en modelos que plasman el diseño del software que serán utilizados como guías en la fase de codificación.

En la búsqueda de información que respalde que tan importante es modelar con UML, se concluyó que su uso es importante en el desarrollo de ontologías que facilitan el diseño del software, para modelar sistemas dinámicos y de esta manera identificar comportamientos anormales, empleado por la ingeniería inversa como herramienta para obtener un modelo de un sistema ya existente para su posterior modificación, para la elaboración de sistemas web reutilizables, modelar funciones, etc.

Mediante el análisis bibliométrico aplicado a las 20 publicaciones obtenidas de la base de datos de Scopus se obtuvo como resultado el análisis bibliométrico un total de 19 grupos de autores asociados por su colaboración, donde Atkinson C, Gerbig R, Tunjic C.V, Fritzsche M son los autores de 2 publicaciones relacionados con Ujhelyi Z, Bergmann G, Hegedüs Á, Horváth Á, Izsó B, Ráth I, Szatmári Z, Varró D.

Las palabras: unified modeling language, modeling languages, computational linguistics y software engineering, lo que permite identificar los estudios están enfocados en los resultados del software utilizando a la ingeniería de software como el punto práctico inicial al momento de obtener un herramienta informática.

Como conclusión general se encuentra que UML resuelve problemas que se presentan al momento de modelar ciertos aspectos como lo es movilidad en los sistemas móviles, para lo cual se encontró información donde se extendió UML, de esta manera dando solución a dicho problema.

## REFERENCIAS BIBLIOGRÁFICAS

- Al-Fedaghi, S. (2014). International Journal of Healthcare Information Systems and Informatics. Recuperado a partir de <https://www.scopus.com/record/display.uri?eid=2-s2.0-84928014984&doi=10.4018%2Fijhisi.2014010102&origin=inward&txGid=06a5b1cb48f0df6893498311df09e9da>
- Alberto, H., & Salvatierra, C. (2015). *ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL Facultad de Ingeniería en Electricidad y Computación INFORME DE PROYECTO DE GRADUACIÓN " INGENIERÍA INVERSA DE UN SOFTWARE PARA MODELAR PROCESOS DE NEGOCIO BASADO EN UNA NOTACION PROPIETARIA , UTILIZANDO NOTACIÓN*. Recuperado a partir de <https://www.dspace.espol.edu.ec/retrieve/101569/D-84599.pdf>
- Alfonzo, I. (1994). *Técnicas de investigación bibliográfica*. (C. Ediciones, Ed.). Caracas.
- Arias Barragan, L. A. (2016). Lenguaje de modelamiento unificado (UML) para modelamiento de embotelladora. *Scientia et Technica*, 21(1), 38–42. Recuperado a partir de <http://www.redalyc.org/articulo.oa?id=84950584006>
- Arias Barragán, L. A., Fernando García, D., & Rivas Trujillo, E. (2013). Modelo de gestión de redes de distribución mediante UML y BPL. *Tecnura*, 17(37). Recuperado a partir de <http://www.redalyc.org/articulo.oa?id=257028383011>
- Atkinson, C., Gerbig, R., & Fritzsche, M. (2015). A multi-level approach to modeling language extension in the Enterprise Systems Domain. *Information Systems*, 54, 289–307. <https://doi.org/10.1016/j.is.2015.01.003>
- Atkinson, C., Gerbig, R., & Vjekoslav, C. (2013). Enhancing classic transformation languages to support multi-level modeling. <https://doi.org/10.1007/s10270-013-0384-y>
- Burek, P., Loebe, F., & Herre, H. (2016). FueL : Representing function structure and function dependencies with a UML profile for function modeling, 11, 155–203. <https://doi.org/10.3233/AO-160167>
- Chamanara, J; Konig-Ries, B. (2014). Ecological Informatics. Recuperado a partir de <https://www.scopus.com/record/display.uri?eid=2-s2.0-85027933915&doi=10.1016%2Fj.ecoinf.2013.12.003&origin=inward&txGid=752d8075c9f825324006a2940dfd30ac>
- Ciencias, F. D. E., Ambiente, D. E. L., Río, D. E. L., La, S. Y. D. E., & Periodo, L. C. (2015). *ANÁLISIS Y DISEÑO DE UN SISTEMA DE INFORMACIÓN PARA EL CONTROL DE PROCESOS JUDICIALES EN LA ZONA REGISTRAL N° VII- SEDE HUARAZ*. Santiago Antunez De Mayolo. Recuperado a partir de <http://renati.sunedu.gob.pe/handle/sunedu/134233>
- Condori, M. (2016). Lenguaje Unificado de Modelado UML. Recuperado a partir de <https://maritzacondori.wordpress.com/2016/03/01/lenguaje-unificado-de-modelado-uml/>
- Elminir, H. K., Riad, A. E. M., & El-halawany, A. M. (s. f.). A Systematic Approach for Reusing Web System Using UML-based Web Engineering. <https://doi.org/10.3923/jse.2016.192.221>

- Fouad, T., & Mohamed, B. (2016). TRANSFORMING XML SCHEMA CONSTRAINING FACETS AND XML QUERIES TO OBJECT CONSTRAINT LANGUAGE, *87*(3), 469–478. Recuperado a partir de <https://www.scopus.com/inward/record.uri?eid=2-s2.0-84971473466&partnerID=40&md5=b55b6ad8b969d290ae38fdbfc0556ef6>
- González Alcaide, G; Valderrama Zuriá, JC; Aleixandre Benavemt, R; Alonso Arroyo, A; de Granda Orive, J I; Villanueva Serrano, S. (2006). Transtornos Adictivos. Recuperado a partir de <http://www.elsevier.es/es-revista-trastornos-adictivos-182-articulo-redes-coautoria-colaboracion-las-instituciones-13090606>
- Hernandez Sampieri, Roberto; Fernández Collado, Carlos; Baptista Lucio, P. (2006). *Metodología de la Investigación* (McGraw-Hill). Mexico.
- Marcos, Esperanza; De Castro, M Valeria; Vela S, B. (2004). e-Gnosis. Recuperado a partir de <http://www.redalyc.org/articulo.oa?id=73000207>
- Muhairat, M. I., & Abdelqader, A. (2016). A Mathematical Model to Verify a Reverse Engineering Approach Which Convert Form Fill Format into UML Class Diagram, *10*(5), 45–60. Recuperado a partir de <https://www.scopus.com/inward/record.uri?eid=2-s2.0-84971543390&doi=10.14257%2Fijseia.2016.10.5.05&partnerID=40&md5=2a6597aa4ac83a762fe8c8b6e5658249>
- Reynares, E., Laura, M., & Rosa, M. (2014). Expert Systems with Applications Approaching the feasibility of SBVR as modeling language for ontology development : An exploratory experiment. *Expert Systems With Applications*, *41*(4), 1576–1583. <https://doi.org/10.1016/j.eswa.2013.08.054>
- Rhazali, Y; Hadi, Y; Mouloudi, A. (2016). International Journal of Software Engineering and its Applications. Recuperado a partir de <https://www.scopus.com/record/display.uri?eid=2-s2.0-84964788276&doi=10.14257%2Fijseia.2016.10.4.13&origin=inward&txGid=9e87266319f73927059c73ac0904401c>
- Rodas, A. (2016). Creating an architecture using Unified Modeling Language ( UML ), (20), 15–23.
- Romero, N., & Escalada, R. De. (2016). Propuesta de Extensión de UML para Proceso de Conceptualización De Requisitos, *4*(2), 59–72.
- Sciences, A., Scientific, M., & Corp, P. (2014). m-TOPP-UML : An Extension to UML for the Modeling of Mobile Tracking on Patient Progress System Hussain Abu-Dalbouh Department of Computer Sciences , College Sciences and Arts in Unaizah , Al-Qassim University , Kingdom of Saudi Arabia. <https://doi.org/10.19026/rjaset.7.406>
- Solano Lopez, E. (2009). MediSur. Recuperado a partir de [http://scielo.sld.cu/scielo.php?pid=S1727-897X2009000400011&script=sci\\_abstract](http://scielo.sld.cu/scielo.php?pid=S1727-897X2009000400011&script=sci_abstract)
- VOSviewer. (2018). VOSviewer. Recuperado a partir de <http://www.vosviewer.com/>