

Urkund Analysis Result

Analysed Document: propuesta_version_urkund201849143918.doc (D38082239)
Submitted: 4/28/2018 12:52:00 AM
Submitted By: mcorreap@unemi.edu.ec
Significance: 1 %

Sources included in the report:

PAREDES.docx (D12455336)

Instances where selected sources appear:

1

INTRODUCCIÓN

En la actualidad el avance de los sistemas computacionales es cada vez más, por esta razón se debe lograr un mayor grado de acercamiento a las necesidades establecidas de la situación real. La Ingeniería de software juega un papel importante dentro de esta área de la ingeniería computacional, en donde ofrece metodologías, técnicas y herramientas para el diseño y desarrollo de software de alto rendimiento, desde la especificación de requisitos hasta el producto final. El diseño de software es parte de las etapas del ciclo de vida del software, ya que aquí se desarrollarán los distintos modelos del sistema a construir. Actualmente, pensar en el desarrollo del software sin pasar por la etapa del modelado es prácticamente irrealizable. Según CITATION Ast09 \l 3082 (Asteasuain, 2009) "Los lenguajes de modelado son herramientas para construir los modelos del sistema, un modelo es la abstracción de una entidad del mundo real, existen varios lenguajes para ello, el lenguaje de modelado UML es considerado un estándar dentro de la comunidad científica."

El presente trabajo está conformado por 5 capítulos: el capítulo 1, contiene el problema de investigación en donde se conocerá y delimitará el problema, se establecerán los objetivos de la investigación tanto el general como los específicos y la justificación del problema. El capítulo 2, describirá el marco teórico conceptual el cual comprende los antecedentes de la investigación y los fundamentos teóricos. El capítulo 3, dará a conocer la metodología usada. El capítulo 4, incluirá el desarrollo del tema en el cual se detallará lo elaborado, y por último el capítulo 5, alojará las conclusiones del proyecto.

En este trabajo se analizará publicaciones tanto de revistas, periódicos o documentos referente al tema de UML publicados en la base de datos SCIELO, se manejará el análisis en tres tipologías, en países, por periódicos y por áreas temáticas, representando en cada uno el nivel de relación con el tema de UML, tomando en consideración el año 2014, comparando a su vez, tres herramientas para el diseño de diagramas UML.

CAPÍTULO 1

PROBLEMA DE INVESTIGACIÓN

Planteamiento del Problema El lenguaje UML se creó con la finalidad de idear una expresión visual más sencilla dentro del complicado mundo del proceso del software, permitiendo así que cualquier tipo de usuario que desee entender un sistema consiga hacerlo. El propósito de UML es ayudar a puntualizar, documentar y representar modelos de sistemas de softwares. En la actualidad el uso de UML juega un papel importante al momento de modelar, dentro de la fase de diseño de elaboración de un software, es por eso que hoy en día para lograr un buen modelado o tener conocimiento se recurre a la variada cantidad de documentos que nos facilitan las distintas bases de datos bibliográficas. Sin duda alguna cada vez es mayor encontrar bases de datos científicas y el gran impacto que tienen dentro de la sociedad teniendo una amplia variedad de usuarios, las nuevas tecnologías nos brindan grandes oportunidades de difusión, obligando a tener un buen análisis adecuado de los documentos. Actualmente el análisis documental se enfoca al proceso de extraer las ideas representativas de un documento facilitando su acceso al original. En los últimos años, los métodos de análisis

suelen ser diversos para lograr interpretar el documento. El grado de profundidad de análisis va a depender de los aspectos que se den como son: • Los usuarios y las necesidades de la información que requieran. • Los servicios que brinden, sean empresas o bibliotecas. • Los diferentes documentos a indagar, como libros, revistas, etc. • Las Bases de datos y las oportunidades que brindan los sistemas de recuperación.

Al realizar un análisis documental del modelado UML en el diseño de un software dentro de una base de datos científica lo ideal sería obtener cuantiosa cantidad de información, ya que así se estaría aportando a una considerable recopilación de datos al momento de efectuar la investigación. Hoy en día uno de los problemas de mayor importancia, es el número de publicaciones que se dan en las bases de datos científicas acerca de UML, ya que se ha reducido gradualmente el porcentaje de publicación en los últimos años, dando como resultado acceder a poca información.

En el 2014 las publicaciones eran superiores que en el 2016 y 2017 acerca de temas relacionados con UML, en donde se tenía una amplia información encontrada en artículos, libros, tesis, etc.

Varias de las documentaciones que se lograba encontrar de UML, se basaban en el estudio de artículos, revistas científicas, libros o periódicos, en donde colaboraban al análisis documental de información al momento de hacer uso de UML o trabajar con este tema, cabe mencionar que gran parte de la recolección de información es tomada de varios autores que ayudan o que muestran sus publicaciones dentro de las bases de datos científicas.

Al contar con este tipo de información se estará al tanto de lo nuevo que va apareciendo sobre un determinado tema, sobre qué campo de investigación se encuentra, cuál es el autor, y que revistas publican más dentro de estas bases de datos. Se indagará en las publicaciones que se daban en el 2014 en la base de datos SCIELO acerca de UML, ya que actualmente es escasa la cantidad de información encontrada en SCIELO.

Objetivos Objetivo general: • Analizar publicaciones relacionadas a modelado UML para el diseño de software en la base de datos SCIELO en el 2014.

Objetivos específicos: • Realizar levantamiento documental relacionado al tema UML en la base de datos de SCIELO en el 2014.

• Comparar softwares que contribuyen al desarrollo UML. • Concluir los resultados obtenidos del proceso de análisis documental acerca de UML en SCIELO en el 2014

Justificación El trabajo investigativo documental a desarrollar tiene como finalidad el analizar las publicaciones relacionadas a UML en las bases de datos científicas, en este caso el estudio se enfocará en la base de datos SCIELO en el 2014, con el fin de hacer un análisis comparativo de la cantidad de información que se encontraba en dicho año y la poca información que se encuentra en la actualidad. Durante el proceso de este análisis permitirá conocer información relevante sobre las diferentes publicaciones en relación con países, periódicos, revistas y áreas temáticas, que se daban en el 2014. La base de datos SCIELO por su amplio desarrollo científico permite responder a las necesidades que requieren los países en especial América

Latina y el Caribe, contribuyendo a la toma de decisiones en los distintos niveles, promoviendo el acceso y uso de información científica y técnica en los diferentes países. El trabajo expuesto identificará los diferentes softwares que contribuyen al desarrollo UML, permitiendo una visión más amplia del tema. En resumen, se constatará como se ha reducido el número de publicaciones referentes a UML en SCIELO, perjudicando estar al día con softwares nuevos o mejorados que ayuden al modelo UML en el diseño de software. Desde el contexto de articulación para la justificación del trabajo que se presenta, está relacionado con el Plan Nacional de Desarrollo 2017-2021, línea de investigación de la Universidad Estatal de Milagro denominado tecnología de la información y comunicación del grupo de investigación TICs y desarrollo de software, además de contribuir documentalmente los contenidos recibidos en la carrera de ingeniería de sistemas.

CAPÍTULO 2

MARCO TEÓRICO CONCEPTUAL

Antecedentes de la investigación Los científicos de la computación son los responsables de las herramientas y los conceptos de software que permiten la creación de todo el software. Las ciencias de la computación centran en el proceso de cómputo a través de varias áreas de especialización, como la teoría de la computación, los algoritmos, las estructuras de datos, conceptos y lenguajes de programación, la arquitectura de computadores, los sistemas de información gerencial, la inteligencia artificial y la ingeniería de software. Unos de los importantes retos para las ciencias de la computación es el problema de la confiabilidad del software. Las técnicas actuales de desarrollo de software no afirman que un sistema de software marchará sin fallas en todas las condiciones. Cada vez más instituciones humanas dependan de los sistemas de computación, cada vez será más importante que los científicos de la computación encuentren formas para crear y desarrollar software en el cual podamos tener la seguridad y confianza. CITATION MarcadorDePosición1 \l 3082 (Beekman George, 1999) El diseño de software estimula el modelado, al modelar un sistema, el diseñador se concentra en el dominio del problema, ignorando transitoriamente detalles menos significativos para llegar a la solución. Esto proporciona la separación de la complejidad esencial de la complejidad accidental del problema. Como fué exclamado por Fred Brooks en "The Mythical Man-Month", en su libro de administración de proyectos de software "añadir recursos humanos a un proceso retrasado, lo hace demorarse aún más". CITATION JrF95 \l 3082 (Brooks, 1995) , esa separación es beneficiosa para la calidad final del sistema proyectado. La ingeniería de software asistida por computadora (CASE) es el nombre que se da al software que ayuda a las actividades del proceso del software como la ingeniería de requerimientos, el diseño, el desarrollo de programas y las pruebas. Las herramientas denominadas CASE contienen, diccionarios de datos, editores de diseño, compiladores, herramientas de construcción de sistemas, depuradores, etc., ayudando al proceso del software automatizando en varias de sus actividades, proporcionando información acerca del software en desarrollo. CITATION Ian05 \l 3082 (Ian Sommerville, 2005)

Figura 11 Sistema CASE Fuente: Obtenido del libro. CITATION Ken11 \l 3082 (Kenneth Kendall, Julie Kendall, 2011)

“El diseño de software es tanto el proceso de definición de la arquitectura, módulos, interfaces y otras características de un sistema como el resultado de un proceso.”CITATION Dan17 \l 3082 (Ramos, Noriega, Laínez & Durango, 2017)

Para Martin Fowler, consultor en el área de tecnología de objetos, relata en el prefacio de su libro “UML Distilled”, cuando comenzamos a elaborar el lenguaje de modelado unificado, esperábamos poder producir un medio estándar para expresar diseños que no solo reflejara las mejores prácticas de la industria, sino que también ayudaría a desmitificar el proceso de modelado de sistemas de software, creímos que la disponibilidad de un lenguaje de modelado estándar alentaría a más desarrolladores a modelar sus sistemas de software antes de construirlos. La adopción rápida y generalizada de UML demuestra que los beneficios del modelado son bien conocidos por la comunidad de desarrolladores; la creación del lenguaje UML fue en sí misma un proceso iterativo e incremental muy similar al modelado de un gran sistema de software, el resultado final es un estándar integrado y reflexivo de numerosas ideas y contribuciones hechas por cuantiosos individuos y compañías de la comunidad de objetos. CITATION Mar04 \l 3082 (Martin Fowler, Kendall Scott, 2004)

Fundamentos teóricos El Object Management Group (OMG) es un consorcio internacional sin fines de lucro y de membresía abierta para estándares tecnológicos, fundado en 1989. OMG controla la definición y el mantenimiento de las especificaciones de UML. Sus estándares son suscitados por proveedores, instituciones académicas, usuarios finales y agencias gubernamentales. El lenguaje de modelado unificado UML le ayuda a especificar, visualizar y documentar modelos de sistemas de software, incluyendo su estructura y diseño, de una manera que cumpla con todos estos requisitos. También puede usar UML para modelar y modelar negocios de otros sistemas que no son de software. CITATION Obj18 \l 12298 (Object Management Group, Inc, 2018)

Unified Modeling Language(UML) es el sucesor de la aparición de métodos de A (análisis) y DOO (diseño orientado a objetos) que aparecieron a finales de los años 80 y principios de los años 90. El UML pasó por un proceso de estandarización con el OMG (Object Management Group) y ahora es un estándar OMG, se designa a UML un lenguaje de modelado y no un método, los métodos radican de un proceso y del lenguaje de modelado. El lenguaje de modelado hace referencia a la notación (específicamente gráfica) que utilizan los métodos para expresar diseños y el proceso es su consejo sobre los pasos a seguir para hacer un diseño. CITATION Fos17 \l 3082 (Matias Fossati, 2017)

El lenguaje unificado de Modelado se plantea en 1997 como propuesta de estandarización para sistemas orientados a objetos; es un lenguaje visual estándar para la descripción, visualización de estructuras de sistemas basados en software, utilizado para especificar decisiones de análisis, diseño e implementación y en ocasiones para generar código de un lenguaje de programación. Cabe destacar que UML tiene elementos para representar sistemas de software de manera visual, pero no es una metodología para guiar un proceso de desarrollo de software, siendo solo una notación gráfica. Es aplicado desde sistemas técnicos, distribuidos, de tiempo real hasta en software de negocios e información.CITATION Bus \l 12298 (Guillermo Bustos, 2017)

UML permite una modelación de los componentes estáticos de una aplicación software tanto diagramas de casos de uso, diagramas de clase; y como del comportamiento dinámico de sus principales elementos durante el funcionamiento como diagramas de secuencias y de estados. Los diagramas de estado permiten la modelación de estados principales y eventos para una instancia de una clase, o un sistema como un todo, mientras que los diagramas de secuencia permiten modelar instancias de interacción entre actores u objetos de clase a través de mensajes para conocer lo que ocurre internamente entre actores e instancias que participan en un diagrama de estado de un sistema de software. CITATION Cri12 \l 12298 (Vidal, Schmal, Rivero & Villarroel, 2012)

UML surge como una herramienta para el diseño e implementación de una solución automatizada, que está en el modelo de gestión de cualquier sistema teniendo que contar con la documentación apropiada para su desarrollo y mantenimiento. UML constituye un lenguaje de propósito general siendo este una ventaja primordial; porque a través de este lenguaje se logra observar las iteraciones de los elementos que conforma un sistema de gestión como eslabones estructurales con funciones específicas facilitando el diseño de objetos y sus subrutinas. CITATION Lui17 \l 12298 (Arias, Rivas & León, 2017)

0: PAREDES.docx

54%

UML es definido como un lenguaje grafico para visualizar, especificar, construir y documentar los bloques de un sistema. Brinda una forma estándar de diagramar planos de un sistema,

abarcando las partes conceptuales y objetos concretos. Al ser un modelo universal es empleado en mayoría para la descripción de arquitecturas de componentes genéricos y para especificar la estructura interna de elementos arquitectónicos usando diagramas de clases y de secuencia. CITATION Lui13 \l 12298 (Arias, García & Trujillo, 2013)

UML es una herramienta estándar, el cual está dividido en bloques de elementos, relaciones y diagramas, las cuales permite mejorar el diseño de un software dando como resultado un software de mayor calidad, usando UML de manera interactiva los participantes en el proceso de diseño obtienen una mejor comprensión de los requerimientos y de los procesos que deben hacer. CITATION Rom16 \l 12298 (Natalia Romero, 2016)

GQM (Goal-Questions-Metrics) se ha venido usando en el proceso de evaluación de calidad de software, proporcionando una manera útil para definir mediciones tanto del proceso como de los resultados de un proyecto, las mismas que se realizan en laboratorio dedicado a probar software y componentes de software. Teniendo en cuenta las propiedades que ofrece el lenguaje UML, se propone usarlo para especificar los elementos que conforman el enfoque GQM y así garantizar una correcta comunicación entre el equipo de desarrolladores y el encargado de realizar pruebas usando un lenguaje común. La especificación de GQM como herramienta de pruebas usando UML se divide en dos etapas, en la que se representa la estructura usando artefactos de UML y la segunda etapa consiste en incluir los elementos del

perfil de pruebas de UML a los artefactos usados por los desarrolladores. CITATION Jos15 \l 12298 (Hernández, Verona & Pérez , 2015)

Al referirse al tema UML se lo debe tratar como un lenguaje de modelado, tal como lo describe su nombre un Lenguaje Unificado de Modelado, mas no como un proceso o un método. Está compuesto por una notación específica y por reglas semánticas relacionadas a la elaboración de sistemas de software. CITATION Geo \l 3082 (Sparks, 2001) En su última versión UML 2.0 define un total de trece diagramas, divididos en tres categorías principales: • La estructura de estructura: cuenta con seis tipos, incluyendo el diagrama de clase, objeto, componentes, paquete y diagrama de despliegue. • Tipos generales de comportamiento: cuenta con 3 tipos, incluyendo diagrama de casos de uso, de actividad y diagrama de estado. • Diferentes aspectos de interacciones: cuenta con 4 tipos, incluyendo diagrama de comportamiento, de comunicación, de tiempo y diagrama de descripción de interacción.

En UML el modelo dinámico, describe cómo modelar los aspectos dinámicos de los sistemas de software usando la notación y la semántica del UML. Hace uso de los diagramas de secuencia, los diagramas de actividad y las cartas de estados. En UML el modelo lógico es usado para modelar los elementos estructurales estáticos. Captura y define los objetos, entidades y bloques de construcción de un sistema. Es una vista estática de los objetos y las clases que cubren el espacio de análisis y diseño. Frecuentemente, un modelo de dominio es una vista pobre, de alto nivel de los objetos de negocio y de las entidades, y el modelo de clases es un modelo riguroso y enfocado al diseño. CITATION Geo \l 3082 (Sparks, 2001)

CAPÍTULO 3

METODOLOGÍA

Metodología Descriptiva Los estudios descriptivos buscan especificar las propiedades importantes de personas, grupos, comunidades o cualquier otro fenómeno que sea sometido a análisis. CITATION DAN76 \l 3082 (DANKHE, 1976), miden o evalúan diversos aspectos, dimensiones o componentes del fenómeno a investigar. Desde el punto de vista científico, describir es medir. Esto es, en un estudio descriptivo se selecciona una serie de cuestiones y se mide cada una de ellas independientemente, para así describir lo que se investiga. CITATION RHe06 \l 3082 (Hernández & Fernández, 2006)

Metodología Documental Recolecta información necesaria recurriendo a fuentes de datos en los que la información ya se encuentra registrada de forma organizada, tales como libros, revistas especializadas, investigaciones científicas, archivos, información documental. Establecido en el artículo de CITATION DAR14 \l 3082 (DAROSI, 2014), el análisis de estudio documental comprende medir, identificar y analizar la documentación bibliográfica; por lo cual, se realizó este trabajo empleando el método de análisis documental bibliográfico de publicaciones establecidas en la base de datos SCIELO referente al tema de UML.

Para el presente trabajo se formuló siete aspectos generales: definir y establecer ideas, búsqueda de fuentes de información, consulta de artículos en la base de datos SCIELO, seleccionar los artículos y contenido referente al tema UML, redactar y ordenar el documento,

desarrollar el trabajo comparando herramientas para desarrollar diagramas UML y finalmente una conclusión del trabajo final. (Figura 2)

Figura 22. Metodología usada en el desarrollo de la investigación.

Elaborado por: Irma Barahona & Jhonny Calle

CAPÍTULO 4 DESARROLLO DEL TEMA

El análisis de búsqueda de información relacionado a publicaciones en la base de datos SCIELO, respecto al tema de UML en el año 2014 se ejecutó con la siguiente línea de código UML AND IN: * AND JOURNAL_TITLE: * AND YEAR_CLUSTER:("2014") AND SUBJECT_AREA: * y se alcanza representar en las posteriores graficas en relación con países, periódicos o revistas y áreas temáticas. Como se observa en la gráfica, Brasil presenta mayor número de publicaciones, alcanzando 13 artículos en el año 2014, de las cuales 6 publicaciones corresponden a la revista Brazilian Journal of Microbiology. (Figura 3)

Figura 33. Representación por países.

La posterior gráfica, representa el mayor nivel de publicaciones de revistas o periódicos, dando como resultado a la revista Brazilian Journal of Microbiology con un total de 6 publicaciones y a Brazilian Archives of Biology con 4 publicaciones realizadas en el año 2014 en Brasil. (Figura 4)

Figura 44 Representación de publicaciones en revistas o periódicos.

Al analizar la gráfica acerca de las áreas temáticas en el tema de UML, se presenta al área de ingeniería aportando con 14 publicaciones en el año 2014, seguida de las Ciencias Biológicas con un numero de 8 publicaciones realizadas. (Figura 5)

Figura 55. Representación por áreas temáticas.

En la actualidad, el Análisis Orientado a Objeto y diseño (OOA&D) se ha transformado en un modelo de desarrollo de software absoluto. En el ciclo de vida de un proyecto existe un gran avance en los procesos de pensamiento, por lo que ahora es inverosímil pensar en encabezar el desarrollo de un software, sin haber elaborado primeramente el proceso de ingeniería de software obligatorio, incluyendo el modelo de la aplicación a futuro. Para realizar estos procesos existen varias herramientas basados en el modelado UML como lo es NetBeans UML, ArgoUML, StarUML, entre otras.; se realizará una síntesis para comparar las herramientas mencionadas. NetBeans es un IDE (Entorno Desarrollo Integrado), de código abierto, especialmente para el lenguaje de programación Java, da soporte a diversas tecnologías como PHP, Java, HTML, etc.; el plugin "PlantUML-NB" o además EasyUML instalado en NetBeans es una herramienta para modelado que brinda funciones para diseñar diagramas UML y generación de código en Java a partir de un diagrama. Podemos observar el uso de la herramienta de NetBeans versión 8.1 para la representación de un diagrama de clases UML acerca de la gestión de venta de una pizzería. (Figura 6)

Figura 66 Diagrama UML en NetBeans v8.1 ArgoUML es una herramienta que está evolucionando a un ritmo bastante rápido. Se basa en una base sólida que utiliza otros proyectos Open Source de alto perfil y estables como ANT (Another Neat Tool) para la construcción y ANTLER (Another Tool for Language Recognition). Actualmente cumple con la versión 1.3 del estándar UML. La arquitectura modular ArgoUML emplea una biblioteca de metamodelos conectables que permite la rápida adopción de nuevas versiones del estándar UML. CITATION Bri08 \l 3082 (Brian Sam-Bodden, Christopher M Judd, 2008) Se observa el presente diagrama de clases diseñado en la herramienta ArgoUML, para definir el almacenamiento de datos de empleados, clientes e información la empresa. (Figura 7)

Figura 77 Diagrama UML en la herramienta ArgoUML

StarUML es un marco de modelado UML que respalda el enfoque MDA (Model-Driven Architecture). Este marco se caracteriza por su flexibilidad y su funcionalidad extensible. Por lo tanto, StarUML permite agregar nuevas funciones para satisfacer los requisitos del usuario. CITATION Dan11 \l 3082 (Danny Weyns, 2011) En la posterior figura se plantea un diagrama de clases referente a una biblioteca y su gestión de libros, diseñado en la herramienta StarUML. (Figura 8)

Figura 88. Diagrama UML planteado en la herramienta StarUML

Cuadro Comparativo de herramientas CASE analizadas.

Herramientas CASE Ventajas Desventajas • NetBeans • Soporte para PHP. • Posee Templates y Wizards. • Instalación e actualización simple • No tiene editor de código HTML. • Escasos plugins para la plataforma • ArgoUML • Soporte de UML 1.4 • Exportación a seis formatos gráficos. • Multiplataforma • Poco amigable. • No tiene botón deshacer. • Se debe seleccionar una clase para elaborar un diagrama de secuencia • StarUML • Software libre. • Generar código a partir de un diagrama o viceversa. • Brinda plantillas de proyectos • No es multiplataforma. • Da problemas con la ingeniería inversa.

CAPÍTULO 5 CONCLUSIONES

El trabajo de análisis documental realizado en la base de datos SCIELO del 2014, dio como resultado varios puntos a ser analizados como: • La disminución de publicaciones en bases de datos científicas acerca del tema UML, tomando como referencia al año 2014 con 36 publicaciones, en el 2015 con 18 publicaciones, en el 2016 con 17 publicaciones y para el año 2017 tan solo con 6 publicaciones existentes. • Se evidencia que el país con mayor índice publicaciones hacia temas relacionados de UML en el año 2014 es Brasil con un 36%, contando con 13 artículos de los cuales 6 de ellos corresponden a la revista Brazilian Journal of Microbiolog, seguida de Colombia con un 17%, contando con 6 artículos y 3 de ellos pertenecen a la revista DYNA. • Al analizar temas de UML en revistas o periódicos se encuentra que la revista Brazilian Journal of Microbiology aporta con 6 divulgaciones y la revista Brazilian Archives of Biology con un aporte de 4 divulgaciones realizadas en el año 2014, siendo predominante el país de Brasil. • En áreas temáticas acerca del tema a redactar se obtiene resultados donde especifican que en el área de ingeniería se encuentran 14 publicaciones,

seguidas del área de Ciencias Biológicas con 8 aportaciones realizadas, todo esto en el periodo 2014. • Las comparaciones realizadas herramientas para el diseño UML se consiguió analizar NetBeansUML, ArgoUML y StarUML; siendo estas Open Source y adaptable al lenguaje de programación Java.

18

hdphoto1.wdp

hdphoto2.wdp

hdphoto3.wdp

hdphoto4.wdp

hdphoto5.wdp

hdphoto6.wdp

hdphoto7.wdp

Hit and source - focused comparison, Side by Side:

Left side: As student entered the text in the submitted document.

Right side: As the text appears in the source.

Instances from: PAREDES.docx

0: PAREDES.docx 54%

UML es definido como un lenguaje grafico para visualizar, especificar, construir y documentar los bloques de un sistema. Brinda una forma estándar de diagramar planos de un sistema,

0: PAREDES.docx 54%

UML) es un lenguaje gráfico para visualizar, especificar, construir y documentar los artefactos de un sistema de software intensivo. El UML ofrece una forma estándar de escribir los planos de un sistema,