

# INSTITUTO POLITÉCNICO NACIONAL ESCUELA SUPERIOR DE CÓMPUTO

Trabajo terminal

Sistema generador de documentos de casos de uso

"TESSERACT"

2018-B140

---

Presentan:

Jiménez Chávez Luis Gerardo

López Orozco Diego Efrain

Martínez Ibáñez Esteban Pablo

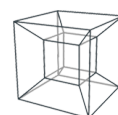
Olvera Neria Yamile Giselle

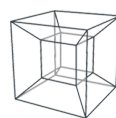
Directores: M. en C. José Jaime López Rabadán, M. en C. Hermes Francisco Montes Casiano  
Escuela Superior de Cómputo, IPN

4 de abril de 2019



|  |           |
|--|-----------|
| <b>1. Introducción</b>   | <b>5</b>  |
| 1.1. Problemática . . . . .  | 5         |
| 1.2. Propuesta . . . . .   | 6         |
| 1.3. Objetivos . . . . .   | 6         |
| 1.3.1. Objetivo General . . . . .  | 6         |
| 1.3.2. Objetivos Específicos . . . . .   | 6         |
| 1.4. Justificación . . . . .   | 7         |
| 1.5. Estructura del Documento . . . . .  | 7         |
| <b>2. Estado del Arte</b>  | <b>9</b>  |
| 2.1. Antecedentes . . . . .  | 9         |
| 2.1.1. UCD-Generator - Una aplicación LESSA para el diseño de casos de uso . . . . .                         | 10        |
| 2.1.2. Generación automatizada de diagramas de casos de uso a partir de requerimientos de usuarios . . . . . | 10        |
| <b>3. Marco teórico</b>  | <b>11</b> |
| 3.1. Ingeniería de software . . . . .  | 11        |
| <b>4. Análisis de mercado</b>  | <b>13</b> |
| 4.1. Situación y evolución del mercado . . . . .   | 13        |
| <b>5. Bibliografía</b>   | <b>15</b> |

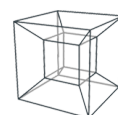




La etapa de mantenimiento de software requiere mayor tiempo y costo que sus fases complementarias, por lo que resulta ser también la etapa de mayor complejidad dentro del ciclo de vida de desarrollo de software. Se estima que aproximadamente dos tercios del costo total del software se dedican al mantenimiento [1]. Esta situación es causada por diversos problemas presentes durante las etapas precedentes, principalmente en la etapa de análisis, ya que es difícil contar con las bases sólidas de una documentación bien construida y estructurada que favorezca a la fase de mantenimiento. Específicamente, el proceso de documentación de los casos de uso requiere una gran cantidad de esfuerzos humanos y es habitualmente propenso a errores, generando un impacto negativo en el desarrollo e implementación del sistema [2].

### 1.1. Problemática

La obtención de requerimientos es crucial para la generación de casos de uso desde el punto de vista del analista [3]. La inadecuada especificación de requerimientos es una de las causas predominantes en el fracaso del desarrollo de los sistemas de software hoy en día [4]. Del mismo modo, es común que el equipo de análisis se enfrente a situaciones que dificultan y prolongan la tarea de documentar casos de uso, algunos de los problemas más comunes son: La falta de consistencia en la utilización de los nombres de los actores, reglas de negocio y mensajes, la incorrecta agrupación de casos de uso en gestiones determinadas, la confusión entre escenarios [5], la falta de adaptación a un estándar de escritura y redacción de los elementos del documento y la incorrecta descripción de derechos funcionales (permisos), todos esto, resultado de la falta de experiencia de los analistas y el gran esfuerzo humano que requiere obtener un producto final óptimo [6].



## 1.2. Propuesta

Se propone construir un sistema que asista a la generación de un documento de análisis basado en casos de uso, que permita gestionar:

Un catálogo de actores el cual explicará brevemente el objetivo del mismo, teniendo la siguiente estructura para definirlos: el nombre del actor, descripción del mismo y sus responsabilidades relacionadas con el sistema según aplique, con el fin de tener consistencia en la utilización de los nombres de los actores. Un catálogo de Reglas de negocio especificando los siguiente: Identificador y nombre de la regla de negocio, de que tipo es, el nivel, una descripción explicando en qué consiste dicha regla, con el fin tener un control al momento de usarlas en diferentes casos de uso. Un catálogo de Mensajes el cual explicará brevemente el objetivo del mismo, este catálogo documentará los mensajes de la siguiente manera: identificador y nombre del mensaje, el tipo de mensaje, propósito, la redacción del mismo y que parámetros deben cumplirse para que el mensaje aparezca esto ayudará a que el usuario pueda reutilizar mensajes en diferentes casos de uso evitando la confusión de los nombres de los mensajes. La agrupación de casos uso dividiéndolos por módulos. Un estándar de redacción y escritura definido para evitar confusiones en la descripción de los casos de uso. Generar de manera automatizada documentos de casos de uso es un desafío que propone la idea de transformar la escritura del lenguaje natural (comúnmente empleado en la elaboración de dichos documentos) a un lenguaje formal y específico. De concretarse este desafío, el tiempo que actualmente toma solucionar los problemas que se presentan durante la elaboración del análisis y su documentación por el personal de análisis será optimizado. Coadyuvará a adquirir experiencia al equipo para disminuir errores, su uso representará una reducción de tiempo (el cual no podrá ser medido debido al tiempo con el que se cuenta para desarrollar el proyecto) y en los recursos destinados al análisis. Lo anterior permitirá generar un documento de análisis con mayor estructura y consistencia.

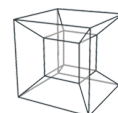
## 1.3. Objetivos

### 1.3.1. Objetivo General

Desarrollar un sistema que asista en la generación de la documentación de casos de uso de un proyecto de software con base en una plantilla predefinida a fin de contribuir en el proceso de su creación.

### 1.3.2. Objetivos Específicos

Generar un módulo de gestión de usuarios encargado del control de acceso y la administración de los usuarios, en este módulo se asignan los roles y permisos a los diferentes usuarios que se registren. Generar un módulo de gestión proyectos encargado de la administración de proyectos, a partir de este módulo se hará el registro, lectura, modificación, eliminación de los componentes necesarios para la documentación del proyecto; contendrá un apartado para la asignación de usuarios al proyecto. Generar un módulo de gestión de casos de uso encargado de la creación, lectura, modificación y eliminación de casos de uso así como la asociación de analistas. Generar un módulo de gestión de elementos encargado de la definición y registro de todos los elementos necesarios para la creación de casos de uso, con el objeto de registrarlos en una base de datos y poder reutilizarlos al momento de escribir un caso de uso. Generar un módulo de revisión y validación de elementos de casos de uso encargado de mostrar los elementos que los conforman para su



revisión y validación por usuarios permitidos. Generar un módulo de generación de documento de análisis encargado de la generación del documento final de casos de uso para el desarrollo de un sistema con base en la plantilla predefinida.

## 1.4. Justificación

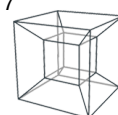
Un proyecto de software bien construido y formado es esencial para la competitividad de una organización, e incluso para su propia supervivencia, del mismo modo, la documentación es un elemento partícipe que determina la calidad del sistema dado que facilita su interpretación y comprensión, provee los antecedentes que permiten conocer cómo fué diseñado, que hace y cómo está operando, sirve de base para auditorias, elimina los riesgos de dependencia con respecto al personal, es fundamental para la capacitación de los usuarios del sistema facilitando la comunicación, provee antecedentes esenciales, concretos y permanentes para evaluar modificaciones a su funcionamiento y/o para decidir la sustitución de los mismos y aumenta la seguridad y eficiencia en su mantenimiento reduciendo su costo.

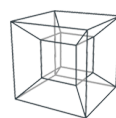
El proceso de construcción del documento no es sencillo, al analista le toma tiempo aprender y hacer de manera entendible la redacción, la inclusión de elementos del caso de uso y la especificación correcta de las trayectorias. La curva de aprendizaje es extensa y es común que una persona inexperta en el tema tenga complicaciones y retrasos al realizar el documento. Una herramienta web capaz de recolectar, almacenar y procesar los elementos que integran un proyecto para generar el documento de análisis será de gran apoyo para los analistas, reduciendo de manera considerable el tiempo, costo y gastos de dicho documento.

Éste proyecto de trabajo terminal se considera un trabajo terminal porque coadyuvará a formación de los autores en áreas de investigación, autoaprendizaje, y resolución de problemas, en la generación de este sistema se utilizarán conocimientos del área de Ingeniería de software, bases de datos, programación, tecnologías web, algoritmos y diseño orientado a objetos.

## 1.5. Estructura del Documento

En el capítulo ?? ... En el capítulo ?? ...







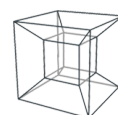
Es común que dentro del área de la ingeniería del software, se confundan los elementos: "Caso de uso" y "Diagrama de caso de uso", sin embargo es importante resaltar las diferencias para comprender el objetivo principal del proyecto terminal.

- Un caso de uso narra una historia detallada sobre cómo interactúa un usuario final (con cierto número de roles posibles) con el sistema en circunstancias específicas. La historia puede ser un texto narrativo, un lineamiento de tareas o interacciones, una descripción basada en un formato o una representación diagramática. Sin importar su forma, un caso de uso ilustra el software o sistema desde el punto de vista del usuario final [7].
- Un caso de uso es aquel que describe la interacción entre un actor y un sistema en forma de secuencia de acciones y circunstancias específicas.
- En cambio, un diagrama de casos de uso es el modelo de un grafo con dos tipos de nodos (Actor y caso de uso), el cual ilustra el comportamiento del caso de uso.

Por lo tanto, el diagrama no es más que una representación gráfica del caso de uso, un diagrama de casos de uso no reemplaza el documento de casos de uso. En el diagrama no se describe la interacción detallada del sistema con los actores.

Ahora bien, en la red hay una gran variedad de sistemas que permiten la generación de diagramas de casos de uso en UML, a partir del procesamiento del lenguaje natural sin embargo no hay herramientas comerciales o gratuitas que posibiliten la generación del documento con las especificaciones y la gestión de sus componentes.

### 2.1. Antecedentes



### 2.1.1. UCD-Generator - Una aplicación LESSA para el diseño de casos de uso

Las herramientas CASE convencionales requieren una comprensión completa del negocio, una gran cantidad de tiempo y esfuerzos adicionales por parte del analista del sistema durante el proceso de creación, organización, etiquetado y finalización de los diagramas de casos de uso. Es por esto que se diseñó un sistema que proporciona una manera rápida y confiable de generar diagramas de casos de uso para ahorrar tiempo y presupuesto tanto para el usuario como para el analista del sistema.

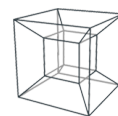
#### Objetivo

Este sistema presenta un enfoque basado en el procesamiento del lenguaje natural LESSA (Language Engineering System for semantic analysis) que se utiliza para comprender automáticamente el texto en lenguaje natural y extraer la información requerida. Esta información se utiliza para dibujar los diagramas de casos de uso. El usuario escribe sus preferencias basadas en la interfaz en inglés, en unos pocos párrafos y el sistema diseñado tiene una capacidad notable para analizar el script dado. Después del análisis compuesto y la extracción de información asociada, el sistema diseñado en realidad dibuja los diagramas de casos de uso.

### 2.1.2. Generación automatizada de diagramas de casos de uso a partir de requerimientos de usuarios

Con el estado actual de la tecnología de procesamiento de lenguaje natural (PNL), muchos investigadores han demostrado que automatizar el proceso de análisis de requisitos es posible, lo que ahorra una cantidad significativa de tiempo invertido por los analistas. Se han desarrollado numerosas herramientas semiautomáticas que ayudan al analista en este proceso. Sin embargo, una técnica comúnmente utilizada para usar la gramática en el texto obtenido como la base para identificar información útil, ha estado enfrentando problemas de escalabilidad debido a que el formato textual de los requisitos consiste en lenguaje natural no estructurado (NL).

Este proyecto aborda este problema utilizando una técnica probabilística para identificar actores y casos de uso. El resultado prometedor demuestra que las mejoras adicionales de este enfoque pueden automatizar completamente la fase de análisis, propone una metodología para la asistencia automática de análisis de requisitos a los analistas de software mediante la extracción de un diagrama de caso de uso del documento de requisitos del usuario. En contraste con investigaciones anteriores en esta dirección, este proyecto ha intentado con éxito extraer actores y usar casos utilizando un modelo de clasificación probabilística junto con una asistencia mínima de enfoque basado en reglas. Los casos de uso son nítidos y consistentes independientemente del tamaño del texto de los requisitos. Debido al pequeño tamaño de los datos utilizados, el rendimiento no se ha logrado precisar. Sin embargo, se pueden utilizar mejores modelos de clasificación con un conjunto de datos más grande que incluya otros dominios de software para mejorar los resultados. El desafío restante aquí se relaciona con abordar los requisitos no funcionales y también para incorporar funciones de inclusión y extensión al diagrama de casos de uso. Un gráfico bien diseñado.

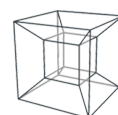


### 3.1. Ingeniería de software

El software se ha incrustado profundamente en casi todos los aspectos de nuestras vidas y, como consecuencia, el número de personas que tienen interés en las características y funciones que brinda una aplicación específica ha crecido en forma notable, por lo que debe hacerse un esfuerzo concertado para entender el problema antes de desarrollar una aplicación de software, por otro lado, los requerimientos de la tecnología de la información que demandan los individuos, negocios y gobiernos se hacen más complejos con cada año que pasa. En la actualidad, grandes equipos de personas crean programas de cómputo que antes eran elaborados por un solo individuo. El software sofisticado, que alguna vez se implementó en un ambiente de cómputo predecible y autocontenido, hoy en día se halla incrustado en el interior de todo, desde la electrónica de consumo hasta dispositivos médicos o sistemas de armamento. La complejidad de estos nuevos sistemas y productos basados en computadora demanda atención cuidadosa a las interacciones de todos los elementos del sistema.

Hacer ingeniería con el software en todas sus formas y a través de todos sus dominios de aplicación, se ha convertido en una necesidad tangible .

El análisis es el proceso clave de la construcción de modernas aplicaciones de sistemas de información y la base para el diseño y desarrollo de la aplicación de software robusto y vigoroso.





---

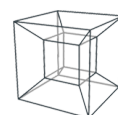
### Análisis de mercado

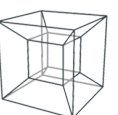
---

En este apartado se demuestra la viabilidad comercial del trabajo terminal TESSERACT, así mismo se realiza un estudio en donde se determina el campo en donde un sistema con las características del generador de documento de casos de uso podría generar un mayor impacto y aceptación por parte de los equipos de construcción de software. Cabe resaltar que TESSERACT no pretende ser comercializado por el momento.

#### 4.1. Situación y evolución del mercado

Dentro del proceso de desarrollo de software existen distintos perfiles o actores que trabajan en una fase en específico para : 1. Líder de proyecto 2. Líder de análisis 3. Líder de diseño





---

### Bibliografía

---

- [1] Rui, K. Butler, G. (2003, April 21). Refactoring use case models: the metamodel [Online]. Available: <https://dl.acm.org/citation.cfm?id=783140>
- [2] Shuang, L. Sun, L. (2014, September 19). Automatic early defects detection in use case documents [Online]. Available: <https://dl.acm.org/citation.cfm?id=2642969>
- [3] J. Lee. (1999, August). Analyzing user requirements use cases a goal driven approach. [Online]. Available: <https://ieeexplore.ieee.org/document/776956>
- [4] L. Julijana. (2007, August). "Information Systems Modeling with Use Cases" IEEE Computer [Online]. Available: <https://ieeexplore.ieee.org/document/4283759>
- [5] Jason Gorman, J. G. (2007, 9 marzo). 10 Common Use Case Pitfalls. Recuperado 23 abril, 2018, [Online]. Available: <http://codemanship.co.uk/parlezuml/blog/?postid=364>
- [6] Susan Lilly [2002, August]. "Use Case Pitfalls: Top 10 Problems from Real Projects Using Use Cases" [Online]. Available: <https://ieeexplore.ieee.org/document/787547>
- [7] Pressman, Roger. (2010). Desarrollo de casos de uso. En Ingeniería de Software. Un enfoque práctico / 7 ED.(777 páginas). USA: McGraw-Hill Interamericana.

