

ESTIMACIÓN DEL TAMAÑO DE UN SISTEMA MEDIANTE EL ANÁLISIS DE DIAGRAMAS UML Y LA APLICACIÓN DE MÉTODOS BASADOS EN PUNTOS DE FUNCIÓN

Nicolás Hernández, Julio Castillo, Marina Cardenas, Gonzalo Albarracín, José Victoria
Laboratorio de Investigación de Software/Dpto. Ingeniería en Sistemas de Información/Facultad
Regional Córdoba/Universidad Tecnológica Nacional
{damiannicolas05, dr.jotacastillo, ing.marinacardenas, galbarracin16, josefvictoria52}@gmail.com

RESUMEN

Este proyecto tiene por objetivo lograr estimaciones precisas del tamaño de un sistema mediante el uso de métodos de estimación que utilicen puntos de función, en las etapas tempranas de un proyecto. En la actualidad existen tres métodos que han ganado mucha relevancia entre los métodos de estimación por puntos de función, estos son COSMIC, IFPUG y NESMA. Para poder utilizar estos métodos se requiere información en las etapas iniciales de un proyecto, y por esta razón se analizarán los diagramas UML, y, en particular, los archivos creados por los diferentes programas que soportan estos diagramas y que brindan archivos con diferentes tipos de estándares tales como los archivos XMI, XML, JSON y SVG.

Palabras Clave Automatización, Mediciones, Estimaciones, COSMIC, IFPUG y NESMA, UML, XMI, XML, JSON, SVG, Puntos de función.

CONTEXTO

El objetivo de este trabajo es el de desarrollar una herramienta que permita realizar estimaciones precisas con la información disponible en las etapas iniciales del desarrollo de un proyecto de software. Dicho trabajo, se encuentra enmarcado dentro del proyecto de investigación homologado por la Secretaría de Ciencia y Tecnología (SCyT) de la UTN “Modelo para el procesamiento de textos estructurados Fase 2” (cód. SIECACO0008518), el cual está dentro del Grupo de Aprendizaje Automático, Lenguajes y Automatas (GA²LA [1]) conformado en la Universidad Tecnológica Nacional - Facultad Regional Córdoba (UTN- FRC), el cual tiene por objetivo investigar los temas relacionados

con las áreas de redes neuronales, gramáticas y sintaxis [2][3][4], y también el desarrollo de sistemas, su calidad y trazabilidad. En etapas tempranas se cuenta con información especificada principalmente en diagramas de UML (Lenguaje de Modelado Unificado), los cuales son generados con las herramientas disponibles en el mercado y que, por lo general, proveen como archivos de salidas en alguno de los formatos estándares mencionados anteriormente, o bien en algún formato propietario. Para poder obtener y analizar esta información se deben realizar análisis léxicos-sintácticos de estos archivos a los efectos de poder realizar el cálculo de los puntos de función.

El proyecto físicamente se desarrolla en el Laboratorio de Investigación de Software LIS del Dpto. de Ingeniería en Sistemas de Información de la Universidad Tecnológica Nacional Facultad Regional Córdoba (UTN-FRC), el cual está compuesto por doctores, doctorandos, ingenieros, docentes - investigadores, pasantes, becarios alumnos y tesis de posgrado.

1. INTRODUCCIÓN

La problemática que afecta a los proyectos de sistemas es el no poder contar con información suficiente para saber cuántos recursos asignar para lograr entregar dentro de los plazos acordados, con las funciones solicitadas y dentro del presupuesto establecido. Según la IFPUG (International Function Point User Group) el 45% de los proyectos exceden su presupuesto, 7% entregan fuera de cronograma y 56% entregan menos requerimientos [5]. Es por esto que las empresas deben poder predecir estas tres

variables con la mayor precisión posible con el fin de disminuir los riesgos y las pérdidas, para lograr el objetivo de entregar un sistema con la mayor calidad posible que cumpla con las especificaciones y requerimientos. Para poder realizar esto, es preciso recurrir a una tarea compleja que permite predecir tiempo, costo y esfuerzo: la estimación [6]. Los principales métodos de estimación pueden dividirse en dos grandes grupos, el primero se basa en experiencias adquiridas en proyectos anteriores y en el conocimiento que se tiene del dominio del negocio, y el segundo grupo incluye los métodos basados en algoritmos de costos. Este último grupo, como se mencionó anteriormente, es el que más relevancia ha obtenido con los métodos basados en puntos de función, los cuales tienen la ventaja de ser independiente tanto de los expertos como de los lenguajes de programación. Entre los más populares se pueden nombrar NESMA FPA [7], IFPUG-FPA [5] y COSMIC [8]. Dado que estos métodos requieren de mucho tiempo y esfuerzo para su realización, se ve la necesidad de automatizar el proceso de estimación. Como existe una gran variedad de software que gráfica los diagramas de UML, se planea desarrollar una herramienta que acepte diferentes tipos de archivos y que los procese automáticamente.

Para esto se seleccionaron, según diferentes características, 25 programas. Los cuales fueron estudiados para determinar si cumplían con las siguientes características:

- Soportar UML total o parcialmente.
- Ser gratuitos, de licencia libre (GNU, GPL, MIT, BSD, EPL, shareware), o contar mínimo con una versión de prueba o con suscripción con características reducidas.
- Los tipos de archivos debían ser XMI, XML, JSON, SVG y/o extensiones propias de cada software que fueran similares a los antes mencionados.

Los programas seleccionados fueron:

- BOULM.
- DIA.

- Modelio.
- NClass.
- ModelSphere.
- UMPLE.
- visual paradigm.
- WhiteStarUML.
- app.diagrams.net.
- UML Designer.
- ALTOVA UModel.
- Moqup.
- Creately.
- Visio.
- StarUML.
- JetUML.
- IBM Engineering Systems Design Rhapsody.
- Enterprice Architect.
- Diagramo.
- GenMyModel.

En la actualidad se ha conformado un banco de prueba con datos de proyectos, diagramas y duraciones. Cada diagrama ha sido generado con cada programa para estudiar y analizar sus salidas, como la salida XMI que se muestra en la siguiente figura.

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <XMI:PROJECT xmlns:XPD="http://www.staruml.com" version="1.1">
3 <XPD:HEADER>
4 <XPD:SUBUNITS>
5 </XPD:SUBUNITS>
6 <XPD:PROFILES>
7 <XPD:PROFILE><UMLStandard/></XPD:PROFILE>
8 </XPD:PROFILES>
9 </XPD:HEADER>
10 <XPD:BODY>
11 <XPD:OBJ name="DocumentElement" type="UMLProject" guid="QwLokAFQZScd3tU1VtQ5wAA">
12 <XPD:ATTR name="Title" type="string">>ipavc/</XPD:ATTR>
13 <XPD:ATTR name="#OwnedElements" type="integer">5</XPD:ATTR>
14 <XPD:OBJ name="OwnedElements[0]" type="UMLModel" guid="Nqy0dUteK2725UCgU1HgAA">
15 <XPD:ATTR name="Name" type="string">Use Case View/</XPD:ATTR>
16 <XPD:REF name="Namespace">QwLokAFQZScd3tU1VtQ5wAA/</XPD:REF>
17 <XPD:ATTR name="#OwnedDiagrams" type="integer">2</XPD:ATTR>
18 <XPD:OBJ name="OwnedDiagrams[0]" type="UMLUseCaseDiagram" guid="2/eF101f0qiyV1oc83mAA">
19 <XPD:ATTR name="Name" type="string">Diagrama de casos de uso esenciales/</XPD:ATTR>
20 <XPD:REF name="DiagramOwner">Nqy0dUteK2725UCgU1HgAA/</XPD:REF>
21 <XPD:OBJ name="DiagramView" type="UMLUseCaseDiagramView" guid="EecF7w0kv0xvVf00xtdAAA">
22 <XPD:REF name="Model">2/eF101f0qiyV1oc83mAA/</XPD:REF>
23 <XPD:REF name="Diagram">2/eF101f0qiyV1oc83mAA/</XPD:REF>
24 <XPD:ATTR name="#OwnedViews" type="integer">32</XPD:ATTR>
25 <XPD:OBJ name="OwnedViews[0]" type="UMLActorView" guid="vz0M9uXj0qPjn2XU5wAAAA">
26 <XPD:ATTR name="LineColor" type="string">1Maroon/</XPD:ATTR>
27 <XPD:ATTR name="FillColor" type="string">50009FFF/</XPD:ATTR>
28 <XPD:ATTR name="Left" type="integer">58/</XPD:ATTR>
29 <XPD:ATTR name="Top" type="integer">124/</XPD:ATTR>
30 <XPD:ATTR name="Width" type="integer">152/</XPD:ATTR>
31 <XPD:ATTR name="Height" type="integer">78/</XPD:ATTR>
32 <XPD:REF name="Model">4Tn0kKDYeXq1mba98RQAA/</XPD:REF>
33 <XPD:OBJ name="NameCompartment" type="UMLNameCompartmentView" guid="K6G3ZncdEqhjt09ypRzGAA">
34 <XPD:OBJ name="NameLabel" type="LabelView" guid="0Rg0m1UzE2kaYqP8fIwAA">
35 <XPD:ATTR name="Frontstyle" type="integer">1</XPD:ATTR>
36 <XPD:ATTR name="Text" type="string">Encargado de Liquidaci&#243;n/</XPD:ATTR>

```

Fig. 1 Ejemplo de salida en formato XMI.

Posteriormente estos diagramas de pruebas serán analizados por el software de estimación que se encuentra en desarrollo. Este subsistema de procesamiento de diagramas está basado en la web y utiliza Django como framework web en conjunción con el lenguaje Python. Se eligió Python dado que cuenta con la API Sax [9], la cual permite analizar los

datos XMI, XML, JSON, SVG, que luego permitirá que sean utilizados en el cómputo de los puntos de función al momento de realizar las estimaciones de costo y duración de proyectos software.

2. LÍNEAS DE INVESTIGACIÓN Y DESARROLLO

Este proyecto tiene por objetivo investigar temas en el área de las estimaciones basadas en puntos de función, y las metodologías utilizadas para contar dichos puntos, además de investigar sobre formatos de transferencia de datos basados en texto estructurado y su análisis sintáctico. En ese sentido, la línea de investigación asociada a este trabajo se relaciona principalmente con el área de Estimaciones de Software en Ingeniería de Software.

A su vez, este trabajo se encuentra dentro del proyecto de “Modelo para el procesamiento de textos estructurados Fase 2” (cód. SIECACO0008518) el cual está homologado por la Secretaría de Ciencia y Tecnología de la UTN y se enmarca principalmente en el área de la Lingüística Computacional e Ingeniería de Software.

3. RESULTADOS OBTENIDOS/ESPERADOS

Como resultados obtenidos, podemos mencionar que se logró confeccionar un banco de prueba con un gran volumen de información y una amplia variedad de gráficos UML de muestra, todos en archivos con los formatos de salida anteriormente mencionados. También se ha avanzado con el desarrollo del módulo de análisis de archivos y la interpretación de los formatos de datos.

Se espera que se logre completar el software con un módulo que, en base a los datos obtenidos del módulo de análisis, permita realizar las estimaciones que se irán mejorando hasta lograr el mayor grado de precisión.

4. FORMACIÓN DE RECURSOS HUMANOS

En el proyecto de investigación dentro del cual se encuentra el trabajo descrito en este artículo participan docentes investigadores ingenieros y alumnos del departamento de Ingeniería en Sistemas de Información de la UTN FRC.

Dicho proyecto sirve como marco para la realización de la tesis de Maestría en Ingeniería en Sistemas de Información del Ing. Nicolás Hernández, cuyo director es el Dr. Julio Castillo y su co-directora, la Mg. Marina E. Cardenas. Además ha permitido que estudiantes becarios realicen diferentes tareas permitiendo que incorporen experiencia desde el punto de vista científico.

5. BIBLIOGRAFÍA

- [1] Vázquez J., Castillo J., Constable L. y Cardenas M.. (2018). XX Workshop de Investigadores en Ciencias de la Computación, WICC 2018, Universidad Nacional del Nordeste.
- [2] Castillo J. y Cardenas M. (2010) Using Sentence Semantic Similarity Based on WordNet in Recognizing Textual Entailment. Iberamia 2010, LNCS, vol. 6433, pp. 366-375.
- [3] Castillo J., Cardenas M., Curti A., Casco O., Navarro M., Hernández N. y Velazco M. (2017) Desarrollo de sistemas de análisis de texto. XIX Workshop de Investigadores en Ciencias de la Computación (WICC 2017).
- [4] Cardenas M., Castillo J., Navarro M., Hernández N. y Velazco M. (2019) Herramientas para el desarrollo de sistemas de análisis de textos no estructurados. XXI Edición de Workshop de Investigadores en Ciencias de la Computación. WICC 2019, UN de San Juan, Argentina, 25 y 26 de abril de 2019.

[5] Bundschuh M. y Dekkers C. (2008) The IFPUG Function Point Counting Method. 10.1007/978-3-540-68188-5_12.

[6] PMI (2013) Project Management Institute, PMBOK-Project Management Body Of Knowledge, Quinta ed., Newton Square, Pennsylvania: PMI, 2013.

[7] NESMA. (2019) Function Point Analysis for software enhancement. Professional guide of the Nesma. <https://nesma.org/> Recuperado Febrero 2023, de <https://nesma.org/wp-content/uploads/2020/10/FPA-for-Software-Enhancement-v2.3-EN.pdf>

[8] COSMIC. (2020). Manual de medición de COSMIC. www.cosmic-sizing.org. Recuperado Febrero 2023, de <https://cosmic-sizing.org/wp-content/uploads/2020/05/Part-1-MM-Rules-v5.0-b-EN-ESPA-%C3%91OL-1.pdf>.

[9] XML.sax (2022) Soporte para analizadores SAX2 — documentación de Python - 3.10.6. doc.phyton.org. Recuperado 11 de agosto de 2022, de <https://docs.python.org/es/3/library/xml.sax.html>.