

بسم الله الرحمن الرحيم



**Sudan University of Science and Technology  
College of Graduate Studies**

# Enabling Use Case Function Points Estimation in Service –Based Systems

تمكين النقاط الوظيفية للتقدير في أنظمة الخدمات

**A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE  
REQUIREMENT FOR THE DEGREE OF MASTER IN COMPUTER  
SCIENCE SOFTWARE ENGINEERING TRACK**

***Prepared By:***

Nusiba Abdalrhman Ahmed

***Supervisor:***

Dr. Abdelgaffar Hamed Ahmed

**December 2013**

قال تعالي

﴿ مَا تَكُونُ فِي شَأْنٍ وَمَا تَتْلُو مِنْهُ مِنْ قُرْآنٍ وَلَا تَعْمَلُونَ مِنْ عَمَلٍ إِلَّا كُنَّا عَلَيْكُمْ شُهُودًا إِذْ تُفِيضُونَ فِيهِ وَمَا يَعْزُبُ عَنْ رَبِّكَ مِنْ مِثْقَالِ ذَرَّةٍ فِي الْأَرْضِ وَلَا فِي السَّمَاءِ وَلَا أَصْغَرَ مِنْ ذَلِكَ وَلَا أَكْبَرَ إِلَّا فِي كِتَابٍ مُبِينٍ ﴾

الاية ﴿61﴾ سورة يونس

# Abstract

Software complexity estimation is an important activity to software development life-cycle such as in terms of effort, cost and size. A lot of estimation models were proposed in the literature but they do not consider the new paradigm shift in software development service-oriented architecture (SOA). SOA is a new style of software architecture that has six principles like standard interface (contract) which allows building technology-independent applications that can interoperate. Function Point (FP) metrics were designed to consider functional requirements (complexity) instead of lines of code, which is enriched by Use Case Point (UCP), a software estimation technique proposed to estimate the software size and effort for software development projects of object-oriented systems. The estimation of SOA in terms of cost and effort for the design and code is important activity but considered hard to measure using the traditional FP because of the fact the six principles abstracts many measurable features not like in case of object oriented systems where it was possible to get different sort of metrics like Case Point (UCP) .In SOA service is not under the control of requestor, It means that there is no much information available for estimator. The basic assumption behind like UCP is that artifacts are like white box but in SOA artifacts in general are black-box. The complexity in service-based systems depends on many factors, this is the first step toward enabling UCP for service-based systems.The research has adopted them for estimating service-based systems complexity as a source of information for system under estimation in high level models.

# المخلص

تقدير التعقيد للبرمجيات هونشاط هام لتطوير دورة حياة البرمجيات من حيث الجهد والتكلفه والحجم . هنالك الكثير من نماذج التقدير التى اقترحت في السابق لكنها لاتعبر عن النقلة النوعيه الجديده في مجال تطوير البرمجيات مثل خدمات المنحنى الكائنى هو نمط جديد من هندسة البرمجيات لديها سته مبادئ مثل واجه قياسية التى تسمح للتطبيقات تكنولوجيا البناء المستقله ان تتفاعل مع بعضها البعض . تم تصميم النقاط الوظيفية لنظر في الاحتياجات الوظيفيه بدلا من اسطر التعليمات البرمجيه التى تطوره منها لاحقا استخدام الحالات الوظيفية وهي تقنية تقدير البرامج المقترحه لتقدير حجم البرمجيات والجهد المبزول لتطويرها من وجه نظر المنحنى الكائنى . مع ذلك لايزال يتم التحقق في اساليب التقدير الجيد لخدمات المنحنى .الفكرة الاساسية هي محاولة لإجاد نظره عالميه لربط تقدير التعقيد خدمات المنحنى من خلال استخدام الحالات الوظيفية مستعينه بنماذج لغة النمذجه الموحد . هما نموذج حالات الاستخدام ونموذج النشاطات.

# ***ACKNOWLEDGMENTS***

I would like to thank the members of my family who assisted me with this project: my mother ,who copy edited my report, and my father ,my brothers and sisters and who assisted in the construction of the display board .

A special not of thank s to DR: Abdelgaffar Hammed Ahmed of Department of software engineering Sudan University of Science & Technology, for compose , format my thesis document , for reviewing it , for encouragement, involvement, motivation throughout my Master's degree for his support and make it easier .

## Table of Contents

	<b>Page</b>
الاية	<b>ii</b>
<b>Abstract</b>	<b>iii</b>
المستخلص	<b>iv</b>
<b>Acknowledge GMENTS</b>	<b>v</b>
<b>List of Contents</b>	<b>vi</b>
<b>List of Contents</b>	<b>vii</b>
<b>List of Figures</b>	<b>vii</b>
<b>List of Table</b>	<b>viii</b>
<b>1.Chapter One: Introduction</b>	<b>x</b>
1.1. Introduction	<b>1</b>
1.2. Problem Definition	<b>2</b>
1.3. Research Objectives	<b>3</b>
<b>2.Chapter Two: Literature Review</b>	
2.1. Introduction	<b>4</b>
2.2. The Estimation of Complexity	<b>4</b>
2.3 Service	<b>4</b>
2.3.1 Core Web Services technologies	<b>4</b>
2.4. Service Oriented Architecture	<b>6</b>
2.4.1 Kinds of Services	<b>7</b>
2.4.3 Design Principles for Services	<b>7</b>
2.4.4. Benefits of SOA	<b>7</b>
2.5. Function Point	<b>8</b>
2.5.1 Benefits of Function Point Analysis	<b>8</b>
2.5.2 Function Point Counting Diagram	<b>8</b>
2.5.3Function Point Formula	<b>9</b>
2.6 Use Case Points (UCP)	<b>9</b>
2.6.1.1. Unadjusted Use Case Weight (UUCW)	<b>10</b>
<b>3. Chapter Three: Proposed Case study</b>	
3.1. Introduction	<b>14</b>
3.2. Methodology	<b>14</b>
3.3.The deferent between OOP&SOA	<b>15</b>

3.4 SOA with UML	<b>15</b>
3.5 UML Models for Estimation	<b>16</b>
3.5.1 Use Case Model	<b>17</b>
3.5.2 Activity Model	<b>17</b>
3.5.3 Weighting Activity Model Constructs	<b>17</b>
3.6 The Calculation of the Complexity	<b>19</b>
3.7 Case study1	<b>19</b>
3.8 The Calculation of UCP for SOA	<b>26</b>
3.8.1 Unadjusted services Weight	<b>27</b>
3.8.2 Unadjusted Actor Weight (UAW)	27
3.9 Case study2	<b>28</b>
3.9 .1 The Calculation of UCP for case studay2	<b>32</b>
3.10. Related Work	<b>34</b>
3.11 CONCLUSION	<b>35</b>
3.12 REFERNCES	<b>36</b>
3.12 REFERNCES	<b>37</b>

## List of Figures

<b>Figure No.</b>	<b>Name</b>	<b>Page</b>
Figure 2.1	Components of the services.	<b>5</b>
Figure 2.2	Function Point Counting Diagram	<b>5</b>
Figure 3.1	Course Manger Use Case Diagram	<b>21</b>
Figure3. 2	Course Manger Activity Diagram	<b>23</b>
Figure3.3	Course Manger Activity Diagram	<b>24</b>
Figure3.4	Course Manger Activity Diagram	<b>24</b>
Figure3.5	Course Manger Activity Diagram	<b>24</b>
Figure 3.6	Online Shopping System	<b>29</b>
Figure3.7	login activity diagram	<b>31</b>
Figure3.8	purchase products activity diagram	<b>32</b>



**List of Table  
Name**

<b>Table No</b>	<b>Name</b>	<b>Page</b>
Table 2.1	Use case Weight Classification	11
Table 2.2	Actor Weight Classification	12
Table 3.1	Mapping classifications SOA principles to UML Models	16
Table 3.2	Activity Weight Classification	18
Table 3.3	Classification of services in the case study1	25
Table 3.4	Unadjusted services Weight (USW)	26
Table 3.5	Unadjusted Actor Weight (UAW)	28
Table 3.6	Classification of services in the case study2	30
Table 3.7	Unadjusted services Weight (USW)	31
Table 3.8	Unadjusted Actor Weight (UAW)	32