

The Method for Cloud Service Recommendation Based on Requirements of Tenant

An Young Min[†] · Kang Tae Jun[†] · Park Joon Seok^{**} · Yeom Keun Hyuk^{***}

ABSTRACT

It is difficult to provide proper cloud services for cloud users, because the number of cloud services are increasing and the type of cloud services are diversifying. To overcome this problem, the concept of cloud service broker is presented to mediate cloud services between cloud providers and tenant. The most important role of cloud service broker is to finding cloud services that fulfill requirements of tenant. However, current existing cloud service broker conduct passive requirements analysis process with cloud service expert's assistance. In addition, the systematic functional and non-functional requirement analysis is insufficient. Therefore, we need the new methods for requirement analysis to find nearest service that matches with requirement of tenant. In this paper, we apply pairwise comparison from AHP method to analyze requirement automatically and systematically. It calculates score of service by comparing requirement with service specification, calculating importance rate, and so on.

Keywords : Cloud Service Broker, Service Requirement Analysis, Service Analysis, Cloud Service Recommendation

테넌트 요구사항 기반의 클라우드 서비스 추천 방법

안 영 민[†] · 강 태 준[†] · 박 준 석^{**} · 염 근 혁^{***}

요 약

클라우드 서비스가 다양해지고 그 수가 급격히 증가하면서 클라우드 서비스 사용자에게 적합한 서비스를 제공하기가 어려워지고 있다. 이러한 문제를 해결하기 위해 서비스 사용자와 제공자 사이에서 클라우드 서비스를 중개해주는 클라우드 서비스 브로커라는 개념이 제시되었다. 클라우드 서비스 브로커의 핵심적인 역할은 사용자의 요구사항에 적합한 서비스를 찾아주는 것이다. 그러나 현재 제시된 클라우드 서비스 브로커 대부분은 클라우드 서비스 전문가의 도움으로 사용자의 요구사항을 분석하는 수동적인 과정이 수행되고 있으며, 체계적으로 기능적, 비기능적 요구사항을 분석하는 과정이 미흡하다. 따라서 유사한 기능을 가지고 있는 클라우드 서비스 중 사용자의 요구사항의 중요도를 분석해서 가장 근접한 서비스의 선택을 지원해줄 수 있는 요구사항 분석 기법이 필요하다. 본 논문은 AHP(Analytic Hierarchy Process)의 쌍대비교법을 응용하여 클라우드 서비스에 대한 사용자의 요구사항을 분석하고 각 서비스의 명세에 따른 점수를 계산하여 그 결과에 따라 클라우드 서비스를 선택하는 방법을 제시한다.

키워드 : 클라우드 서비스 브로커, 서비스 요구사항 분석, 서비스 분석, 클라우드 서비스 추천

1. 서 론

시장조사 기관 가트너에 따르면[1], 최근 클라우드 서비스 사용자의 수요가 큰 폭으로 늘어났으며 클라우드 서비스의 도메인과 유형이 매우 다양해졌다. 이에 따라 클라우드 서

비스 사용자들이 원하는 서비스를 선택하는 데 어려움을 겪고 있다. 이 문제점을 해결하면서 클라우드 서비스의 사용을 확산시키기 위해 제안된 개념이 클라우드 서비스 브로커(Cloud Service Broker)[2]이다. 클라우드 서비스 브로커의 핵심 역할은 클라우드 서비스 제공자가 제공하는 서비스 정보와 사용자의 요구사항 정보를 이용하여 서비스 사용자에게 적합한 서비스를 제공해주는 것이다.

현재 운영되는 클라우드 서비스 브로커들은 대부분 사용자의 요구사항 수집과 분석을 수동적으로 진행하고 있다. 이로 인해 사용자가 클라우드 서비스 브로커를 통해 서비스를 제공받기까지 많은 비용과 시간이 소요되어 상당한 불편함이 있다. 따라서 사용자의 요구사항을 자동적으로 수집 및 분석할 수 있는 체계적인 접근법이 필요하다.

* 이 논문은 2013년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No. NRF-2013R1A2A2A01068256).

** 이 논문은 2014년도 한국정보처리학회 추계학술발표대회에서 '사용자 중심의 클라우드 서비스 중개를 위한 AHP 기반의 기능적 요구사항 분석 방법'의 제목으로 발표된 논문을 확장한 것임.

† 준 회원 : 부산대학교 전기전자컴퓨터공학과 석사과정

** 정 회원 : 부산대학교 물류혁신네트워킹연구소 연구교수

*** 정 회원 : 부산대학교 정보컴퓨터공학부 교수

Manuscript Received : December 31, 2014

First Revision : March 16, 2015

Accepted : March 19, 2015

* Corresponding Author : Yeom Keun Hyuk(yeom@pusan.ac.kr)

3. 요구사항 분석 방법

3.1 요구사항 계층화

테넌트의 요구사항에는 기능적 요구사항과 비기능적 요구사항이 있다. 기능적 요구사항은 서비스가 수행해야 할 동작들에 대한 요구사항들을 정의하고, 비기능적 요구사항은 기능 이외에 서비스가 지켜야 할 사항(품질, 제약사항 등)들을 정의하고 있다. 본 논문은 기능적 요구사항과 비기능적 요구사항 모두에 대해 분석을 수행하며 이를 Fig. 1과 같은 계층으로 나눈다.

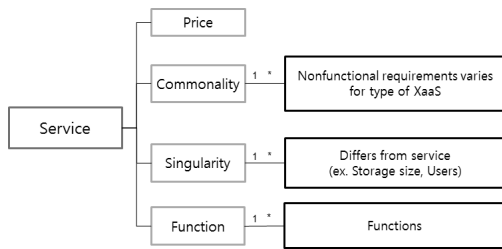


Fig. 1. Requirement Hierarchy Architecture

서비스의 계층은 크게 기능, 공통성, 특이성, 가격으로 구성된다. 기능은 기능적 요구사항을 포함하는 계층이고, 비기능적 요구사항이 3가지 계층으로 분류된다. 공통성은 XaaS로 대표되는 클라우드 유형의 공통적인 비기능성이고, 특이성은 테넌트가 찾고자 하는 서비스의 종류에 따라 달라지는 비기능성이다. 예를 들어, 웹하드 서비스가 필요하다면 저장 공간 용량을 비기능적 요소로 가지는 것과 같다. 가격은 다른 요소와 달리 모든 서비스에 대해 공통적이며 테넌트들이 다른 기능이나 비기능 요소보다 우선적으로 고려하는 중요한 항목이므로 별도로 분류하였다.

3.2 쌍대비교 행렬 및 중요도 계산

2.2절에서 제시한 쌍대비교를 기반으로 테넌트의 요구사항을 분석한다. 임의의 5개의 기능적 요구사항 R1, R2, R3, R4, R5에 대해 요구사항 분석을 수행한다고 가정하며 기능 R1~R5에 대한 쌍대비교 행렬의 예시는 Table 3과 같다.

Table 3. Example of Pairwise Comparison Matrix

	R1	R2	R3	R4	R5
R1	1	5	1	3	3
R2	1/5	1	1/5	1	1/7
R3	1	5	1	3	1
R4	1/3	1	1/3	1	1
R5	1/3	7	1	1	1

2.2절에서 언급한 것과 같이 하나의 요소값이 결정되면 반대 요소의 값도 결정되는 것이 쌍대비교이다. 그 예로 R1과 R4를 비교하면 3의 값을 갖지만, R4와 R1을 비교한 값은 그 역수인 1/3임을 알 수 있다.

쌍대비교 행렬을 구성한 후 각 항목의 중요도를 계산하는

과정을 수행한다. 본 논문에서는 쌍대비교 행렬의 고유벡터 [10]를 구하는 방법을 수행한다. 고유벡터는 쌍대비교 행렬 자신에 대해 행렬곱을 수행한 후 각 행의 합을 더하면 된다. Table 4는 고유벡터를 계산한 결과를 나타낸다.

Table 4. Eigenvector Calculation

	R1	R2	R3	R4	R5	Sum
R1	5	39	7	17	10.71	78.71
R2	0.98	5	1.08	3.34	2.09	12.49
R3	4.33	25	5	15	8.71	58.05
R4	1.53	12.33	2.2	5	3.48	24.54
R5	3.4	21.67	4.07	13	5	47.13

Table 4에서 각 행의 합을 구한 열을 고유벡터라고 할 수 있다. 구해진 고유벡터 역시 벡터이므로 같은 값을 곱하거나 나누어도 같은 성질을 유지한다. AHP에서는 고유벡터의 총합을 1로 하기 위해 모든 값을 더하여 각 행으로 나눈 결과값을 중요도로 정의한다. 다음 Table 5는 중요도 계산 전과 후를 나타낸다.

Table 5. Importance Rate Calculation

	Sum	Importance
R1	78.71	0.3563
R2	12.49	0.0565
R3	58.05	0.2627
R4	24.54	0.1111
R5	47.13	0.2133

divide each value with sum of eigenvector

Table 5에서 산출된 중요도값이 항목 R1~R5 각각의 중요도이다. 여기에는 기능적 요구사항에 대해서만 제시하였으나 공통성, 특이성의 하위 항목들에 대해서도 동일한 방법으로 중요도를 계산할 수 있다.

계산된 중요도는 기능 및 비기능에 따라 다른 용도로 처리된다. 기능 요소의 중요도는 이후 서비스 명세와의 비교 과정에서 기능 점수를 계산하기 위한 각 항목의 점수로 사용되고, 비기능 요소의 중요도는 서비스의 비기능 점수를 계산할 때의 가중치로 활용된다.

Table 5의 R1~R5는 기능적 요구사항으로서, Fig. 1의 기능 하위 계층에 있는 Functions에 있는 요소들 중 어느 기능이 더 중요한지를 나타내는 중요도값을 계산한 것이다. 반면 비기능에 해당하는 공통성과 특이성의 하위 항목은 각각 Fig. 1의 Commonality와 Singularity의 하위 계층의 요소들에 해당하는데, 이들은 각각 적어도 하나 이상의 요소가 존재하며 이들의 개별 항목에 대해서도 중요도를 계산한다. 이와 같이 최하위 계층에 대하여 각각의 쌍대비교를 수행한 후, 상위 계층에 대해 중요도를 계산해야 한다. 즉 기능 중에 어떤 기능이 더 중요한지와 비기능 중에 어떤 요소가 더 중요한지를 결정한 후, 다음 상위 계층인 기능, 공통성, 특이성, 가격 중 어느 항목이 더 중요한지를 결정한다. 각 계층의 중요도 계산 결과는 $weight_{FR}$, $weight_{comm}$, $weight_{singularity}$, $weight_{price}$ 로 나타내며 3.5절에서 최종 서비스 점수를 계산할 때 가중치로 사용된다.

3.3 기능적 요구사항 점수 계산

본 절에서는 3.2절에서 계산된 중요도를 활용하여 기능적 요구사항의 정량적 계산을 수행하는 과정을 제시한다.

기능적 요구사항 분석은 Fig. 2와 같이 어떤 서비스 A의 명세가 테넌트의 요구사항에 기술된 기능을 적어도 하나 이상 포함하고 있는 경우 두 기능 목록을 비교하여 기능적 요구사항 중 서비스 명세에 포함되는 것의 개수가 얼마인지를 파악하는 것이다.

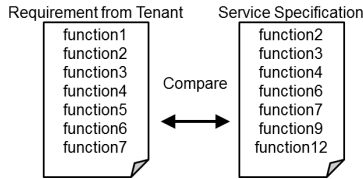


Fig. 2. Example of Function Comparison

Fig. 2의 예시에서 요구사항과 서비스 A의 명세에 모두 포함된 기능은 2, 3, 5, 6, 7의 총 5가지이다. 따라서 서비스 A는 테넌트의 기능적 요구사항 7개 중 5개를 만족하며 5가지 요구사항에 대한 중요도의 합이 서비스 A의 기능점수이다. 여기서는 이 점수의 값을 $score_{FR}$ 로 나타내며, 서비스 A에 대하여 Fig. 1의 최하위 계층 'Functions' 부분에 해당하는 점수 계산이 완료된다.

3.4 비기능적 요구사항 점수 계산

기능과 달리 비기능적 요구사항은 모든 비기능 요소에 대하여 비교하고 점수를 매긴다. 서비스 A가 SaaS라는 가정하에 Table 6과 같은 공통성(Response time, Fault rate 등)을 가진다.

Table 6. Commonality and Specification

Classification	Requirement	Comparison	Specification
Commonality	req_{resp}	Response Time	$spec_{resp}$
	req_{fault}	Fault Rate	$spec_{fault}$
	req_{avail}	Availability	$spec_{avail}$
	-	Usability	$spec_{usa}$
	req_{backup}	Backup	$spec_{backup}$
	-	Security	$spec_{sec}$

Table 6에 나타낸 바와 같이 테넌트의 비기능적 요구사항을 req_{NFR} 라 하고, 서비스의 명세 요소의 값을 $spec_{NFR}$ 라 정의한다. 이때 해당 비기능 요소의 점수는 Equation (1)~(3)을 사용하여 구할 수 있다.

$$nfrRatio = \frac{spec_{NFR}}{req_{NFR}} \quad (1)$$

$$score_{nfrLow} = \begin{cases} 1 & \text{if } nfrRatio \leq 1 \\ 2 - nfrRatio & \text{if } 1 < nfrRatio < 2 \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

$$score_{nfrHigh} = \begin{cases} 1 & \text{if } nfrRatio > 1 \\ nfrRatio & \text{if } nfrRatio \leq 1 \end{cases} \quad (3)$$

Equation (1)은 비기능 요소의 점수를 구하기 위한 값으로 비기능의 비율을 구한 값이다. 용량 등과 같은 비기능 요구사항은 값이 높을수록 테넌트의 요구사항에 더 적합한 경우(Equation (3))와 가격이나 응답 시간과 같이 값이 낮을수록 테넌트의 요구사항에 더 적합한 경우(Equation (2))의 두 가지가 있다. 여기서 산출되는 값을 해당 비기능 요소의 점수로 정의한다.

계산된 공통성 요소들을 사용하여 서비스 공통성의 점수를 산출할 수 있다. 3.2절에서는 Fig. 1의 기능 부분을 계산하였고, 3.3절은 각 비기능 계층들에 대해 점수를 계산하는 과정을 제시한 것이다. 기능적 요구사항의 경우는 3.2절에서 계산한 중요도를 각 요소의 점수로 사용하고 서비스 명세와 요구사항을 비교하여 기능의 포함 여부로 점수를 산출하였으나, 비기능의 경우 3.2절에서 계산한 중요도는 가중치로 사용하고 위의 Equation (1)~(3)으로 계산한 결과를 비기능 요소의 점수로 사용한다. 3.2절에서 계산한 비기능 요소의 중요도를 $weight_{NFR}$ 이라고 정의하고 Equation (4)와 같이 공통성의 점수를 계산할 수 있다.

$$score_{comm} = \sum_{i=1}^k (weight_i \cdot score_i) \quad (4)$$

where k 는 공통성 항목의 개수

이와 유사하게 특이성의 점수도 Equation (5)를 사용하여 구할 수 있다. 특이성은 공통성과 달리 항목의 다양성이 크며, 미리 정의되지 않은 서비스 분류에는 특이성 요소가 존재하지 않을 수도 있다.

$$score_{singularity} = \sum_{i=1}^k (weight_i \cdot score_i) \quad (5)$$

where k 는 특이성 항목의 개수

가격은 그 자체로 별도의 계층을 가지고 있으므로 하위 계층의 점수를 계산하지 않는다. 가격은 Equation (2)와 동일한 방식으로 계산하며 그 결과로 산출된 점수를 가격의 점수로 사용한다. 가격에 대한 점수는 $score_{price}$ 로 나타낸다.

3.5 서비스 점수 계산

최종적으로 테넌트의 요구사항을 어느 정도 만족하는지를 나타내는 서비스의 총점을 계산한다. 앞서 3.3, 3.4절에서 계산한 기능 점수 $score_{FR}$, 공통성의 점수 $score_{comm}$, 특이성의 점수 $score_{singularity}$, 가격의 점수 $score_{price}$ 를 사용한다. 이 네 가지의 항목들은 Fig. 1에 나타낸 계층도에서 상위 계층에 해당한다.

서비스의 총점은 Equation (6)과 같이 계산한다. Equation (6)의 $weight$ 들은 3.2절에서 기술했던 각 상위 계층에 대한 중요도이다.

$$score_{service} = weight_{FR} \cdot score_{FR} + weight_{singularity} \cdot score_{singularity} + weight_{comm} \cdot score_{comm} + weight_{price} \cdot score_{price} \quad (6)$$

4. 사례 및 검증

본 절에서는 사례연구로 하나의 요구사항에 대해 3개의 CRM 서비스를 비교하여 어떠한 과정으로 요구사항과 서비스가 분석되고 서비스 추천 결과로서 어떤 서비스가 선택되는지를 보여준다. 또한, 2절에서 제시한 기존 연구들과의 비교를 통해 본 연구의 활용성을 검증한다.

4.1 사례연구

1) 대상 서비스

비교 대상이 되는 클라우드 서비스는 다수의 서비스 리뷰 사이트를 통해 정보를 수집 및 분석하여 결정하였다. 그 결과 대표적인 클라우드 CRM 서비스들인 Salesforce[11], Nimble[12], Pipeline Deals[13]를 대상 서비스로 선정하였다.

2) 요구사항 분석

테넌트는 적합한 서비스를 추천받기 위해 요구사항을 갖는다. 테넌트의 서비스 기능적 및 비기능적 요구사항을 Table 7 과 같이 선별하였다. 각 요구사항은 사례연구 대상인 각 서비스에서 제공하는 서비스 명세에서 추출한 요소들로 구성하였다.

Table 7. Element of Tenant's Requirement

Classification	Elements
Functional Requirement	<ul style="list-style-type: none"> · Contract Management · Mobile Access · Email Integration · Time Tracking · Activity Management
Nonfunctional Requirement	Commonality <ul style="list-style-type: none"> · Response Time : 1500ms · Fault Rate : 5% · Availability : 99% over · Backup : Required, every 2weeks
	Singularity <ul style="list-style-type: none"> · Number of account : 10 · Number of deal : 5000 · Network Connection Limit : 300 · Storage Size : 30GB · 3rd Party Maximum : 1000
	Price <ul style="list-style-type: none"> · Measure : per month per user · Value : under 30 preferred

한편 Table 8에서는 대상 서비스인 Salesforce, Pipeline Deals, Nimble에 포함된 각 기능을 비교한다. 즉 표의 O, X 는 서비스의 해당 기능에 대한 포함 유무를 나타낸다. 기능 명세는 각 서비스의 공식 소개 자료를 통해 분석한 기능들로 나타내었다.

Table 8. Function Comparison Among CRM Services

Functions	Salesforce - Group	Salesforce	Pipeline Deals	Nimble
Contact Management	X	O	O	O
Social Connection	X	O	X	O
Sales Management	O	O	O	O
Office Integration	O	O	O	O
Email Integration	O	O	X	O
3rd Party Integration	X	O	O	O
Activity Management	O	O	O	O
Data Importing	X	X	O	X
Mobile Access	O	O	O	X
Time Tracking	O	O	O	O
Dashboard	O	O	X	X

Table 7의 기능적 요구사항들을 차례로 FR1, FR2, FR3, FR4, FR5에 대응시키고, Table 9와 같이 테넌트가 설정한 쌍대비교 행렬을 통해 중요도를 분석한다.

Table 9. Pairwise Comparison Matrix of Functional Equirement

	FR1	FR2	FR3	FR4	FR5
FR1	1	5	3	5	3
FR2	0.2	1	1/3	5	1
FR3	1/3	3	1	3	3
FR4	1/5	1/5	1/3	1	1/3
FR5	1/3	1	1/3	3	1

Table 9의 쌍대비교 행렬에 대하여 고유벡터를 구하고 중요도를 계산하면 Table 10과 같다.

Table 10. Importance Rate Calculation Result for Functionalities

	FR1	FR2	FR3	FR4	FR5	EV	IR
FR1	5	23	10.33	53	21.67	113	0.4594
FR2	1.84	5	3.27	15	5.27	30.38	0.1235
FR3	2.87	11.27	5	31.67	11	61.8	0.2512
FR4	0.66	2.73	1.44	5	2.47	12.31	0.05
FR5	1.58	5.27	3	13.67	5	28.51	0.1159

테넌트의 기능적 요구사항에 대하여 각 기능의 중요도를 계산한 결과로 각 기능은 Contract Management가 0.4594, Mobile Access는 0.1235, Office Integration은 0.2512, Time Tracking 0.05, Activity Management 0.1159의 중요도값을 가진다. 이 값들은 테넌트의 각 기능 요소의 점수가 되어 이후 대상 서비스들의 점수 계산 시 사용된다.

같은 과정을 비기능적 요구사항과 상위의 계층에 대해서도 수행하며 그 결과를 Table 11과 같이 정리할 수 있다.

Table 11. Importance Rate Conclusion

Classification	Elements	IR
Commonality	· Response Time	0.3555
	· Fault Rate	0.1070
	· Availability	0.0369
	· Backup	0.2566
	· Usability	0.0699
	· Security	0.1742
Singularity	· Number of account	0.3458
	· Number of deal	0.1304
	· Network Connection Limit	0.0591
	· Storage Size	0.3458
	· 3rd Party Maximum	0.1190
Upper Classes	· Functional Requirement	0.4716
	· Commonality	0.1638
	· Singularity	0.1070
	· Price	0.2576

Table 11의 계산된 중요도값은 이후 서비스 점수를 계산하기 위한 가중치로 사용된다.

3) 서비스 점수 계산

요구사항 분석이 완료되면 각 서비스의 점수를 계산하는 과정을 수행한다. 이때 계산된 서비스의 점수는 테넌트의 요구사항을 얼마나 만족하는지를 정량적으로 나타내는 수치이다. 본 논문에서는 Nimble 서비스에 대한 서비스 점수를 계산하는 것을 일례로 보여준다. 서비스 분석은 기능 점수, 비기능 점수를 구한 후 전체 서비스의 점수를 구하는 순서로 수행한다.

먼저 Nimble 서비스의 기능 명세와 사용자의 기능 요구사항을 비교하여 서로 일치하는 기능을 찾는다. Nimble 서비스의 기능은 사용자의 기능 요구사항 5개 중 4개를 포함하며, 이들은 각각 Contract Management, Office Integration, Time Tracking, Activity Management이다. 이들의 중요도값을 모두 더하면 0.8765이고, 이 값이 Nimble 서비스의 기능 점수가 된다.

다음으로 비기능 점수를 구한다. 비기능 점수는 3절에서 제시한 Equation (1)~(5)를 사용하여 계산하며, Nimble 서비스의 비기능 요소의 값은 Table 12와 같다.

Table 12. Nonfunctional Elements of Service Nimble CRM

Classification	Elements
Commonality	· Response Time : 1300ms
	· Fault Rate : 7%
	· Availability : 95%
	· Backup : Every 1week
	· Usability : Good
	· Security : Good
Singularity	· Number of account : unlimited
	· Number of deal : unlimited
	· Network Connection Limit : unlimited
	· Storage Size : 10GB
	· 3rd Party Maximum : unlimited
Price	· Measure : per month per user
	· Value : \$15

Table 12에서 ‘unlimited’로 표현한 부분은 실제 서비스 명세에서 제한을 두지 않는 항목이다. 구성된 각 비기능 요소에 대해서 Equation (1)~(5)를 적용하면 Table 13과 같은 결과를 얻을 수 있다.

Table 13. Score Calculated for Each Nonfunctionalities

Classification	Elements	Score
Commonality	· Response Time	1
	· Fault Rate	0.6
	· Availability	0.8
	· Backup	1
	· Usability	0.8
	· Security	0.8
Singularity	· Number of account	1
	· Number of deal	1
	· Network Connection Limit	1
	· Storage Size	0.33
	· 3rd Party Maximum	1
Price	· \$15/mo/user	1

비기능 요소는 특성상 경우에 따라 점수가 1이 될 가능성이 높다. 수집된 점수를 사용하여 Equation (4), (5)를 사용해 공통성과 특이성의 점수를 계산할 수 있으며 그 값은 각각 0.9011, 0.7684이다.

앞서 계산한 기능, 공통성, 특이성 점수와 가격 점수를 모두 사용하여 Nimble 서비스의 점수를 계산할 수 있다. Table 11에서 미리 상위 계층에 대한 중요도값을 계산하였으므로 이를 가중치로 사용한다. Equation (6)을 적용하면 전체 서비스 점수를 계산할 수 있으며 그 값은 0.7095이다.

이와 마찬가지로 나머지 서비스들에 대해서도 계산하면 Table 14와 같은 전체 서비스 점수를 얻을 수 있다.

Table 14. Score for Each Service

Service	Salesforce -Group	Salesforce	PipelineDeals	Nimble
Score	0.4685	0.6431	0.6590	0.7095

Salesforce는 4개의 플랜이 있으므로 이들에 대해 각각 점수를 구했으나 3개의 플랜의 점수가 같아 동일하게 표기하였다. 최종적으로 Table 7의 요구사항에 대해서 다른 서비스들을 비교한 결과 4개의 서비스 중 가장 적합한 서비스로 Nimble을 추천할 수 있다.

4.2 비교 및 평가

2절에서 제시한 기존 연구들과 본 논문에서 제시하는 방법을 서로 비교하였다. 기존 연구들 역시 테넌트의 요구사항을 분석하여 테넌트가 사용하고자 하는 서비스를 찾아준다는 점에서 유사하다. 성능의 평가를 위해서는 본 논문에서 접근한 방법과 다른 방법들에 대한 결과의 정확성을 비교해야 하

Table 15. Qualitative Evaluation of Several Studies

Studies	T. Han[2][3]	S. Sundareswaran[4]	L. Qu[5]	M. Godse[6]	G. Baranwal[7]	This Work
Supported Cloud Type	SaaS PaaS IaaS	IaaS	Not specified	SaaS	SaaS PaaS IaaS	SaaS PaaS IaaS
Supported Requirement Type	NFR Only	NFR Only	NFR Only	FR and NFR	NFR Only	FR and NFR
Appliance	Ontology	B+ Tree	Context and rank analysis	AHP	Metric Calculation	AHP and additional calculation
Hierarchy Existence	Yes	No	No	Yes	No	Yes
Characteristics	Needs professionalism of semantics	Complicated algorithms	Too many models for way of appliance data	Just simply applied AHP, not along with scoring	Can not consider changes in requirement for each circumstance	Limitation on number of pairwise comparison

지만, 다른 방법들은 지원하는 클라우드 서비스의 유형이 상이하고 비기능적 요구사항만 고려한 서비스 검색을 수행하기 때문에 동일한 기준에서의 비교가 불가능하다. 따라서 Table 15와 같이 정성적 평가를 통한 비교를 수행하였다.

또한 다양한 의사결정 도메인에 지속적으로 적용 및 활용되고 있는 AHP 의사결정 모델을 클라우드 서비스를 추천하는 데 적용하였으며, 본 논문에서 제안한 기법에 대한 적용 타당성을 확보하였다. 본 논문에서 제안한 기법은 AHP를 근간으로 하지만 클라우드 서비스의 유형에 따른 품질을 분석하여 계층으로 정의하였고, 클라우드 서비스를 선택 및 추천하는 일련의 프로세스를 제시하였다. 그리고 AHP를 기반으로 테넌트의 요구사항에 대한 정량적 가중치를 산출하여 클라우드 서비스를 점수화하는 방법을 제시하였다.

본 사례연구에서 요구사항에 가장 적합한 클라우드 서비스로 추천된 Nimble CRM 서비스의 경우 테넌트가 요구하는 5개의 기능 중 4개를 충족하고 있으므로 80%의 요구사항을 만족한다고 할 수 있다. 이와 같이 테넌트의 기능적 요구사항의 유무로만 클라우드 서비스를 추천한다면 5개의 기능을 모두 포함하는 Salesforce CRM 서비스가 선택되었을 것이다. 그러나 테넌트가 개인 선호도에 따라 요구한 상대적 중요도를 기반으로 기능, 공통성, 특이성, 가격의 전반적 가중치를 고려하여 계산하였고, 그 결과 보다 더 합리적으로 테넌트 맞춤형 서비스인 Nimble CRM 서비스가 선택되었다.

본 논문에서 제안한 쌍대비교 기반으로 요소별 가중치를 계산하는 수식에서 각 요소의 상대적인 중요성을 결정하기 위해 AHP에서 제시한 중요도 비교 기준값이 적용되었다. 또한 비기능성 점수를 계산할 때는 테넌트가 요구한 것과 서비스가 제시한 것의 차이를 비율로 계산하였다. 한국정보통신기술협회의에서 제시하는 소프트웨어 요구사항 품질평가항목 [14] 부분에서 제시된 요구사항 측정값의 범위는 0과 1 사이 값을 기준으로 제시하고 있다. 따라서 그 범위 값을 넘어서

는 경우, 보정이 가능한 경우에는 보정 처리하며 그렇지 않고 2배 이상을 넘는 경우에는 0으로 처리하도록 기준을 세워 비기능 점수 계산을 위한 Equation (1), (2), (3)을 제시하였다.

AHP 의사결정 모델을 도입함으로써 얻을 수 있는 장점은 확일적인 기준이 아니라 테넌트의 주관적인 상대적 기준에 따라 가중치가 설정된다는 것이다. 이로 인해 테넌트가 원하는 요구사항에 가장 적합한 서비스를 찾아줄 수 있는 가능성이 높으며, 본 논문은 이러한 AHP의 장점을 도입하여 제시한 기법이다.

본 논문에서 제안한 수학적 모델의 계산식의 기준은 AHP 모델 활용 및 TTA의 기준, 내부 설문 등을 통해 제시되었다. 수학적 모델에 대한 효용성은 제시한 사례연구를 별도로 다양한 사례에 적용하여 검증을 수행하고 있으며, 이에 대한 실제적인 판단은 서비스 제공 후 테넌트의 피드백을 통한 더 구체적인 검증이 수행되어야 할 것이며 향후 연구로 진행 중에 있다.

5. 결 론

본 논문에서는 기존의 클라우드 서비스 브로커들이 가지는 불편함과 기존 연구들의 문제점을 해결하기 위해 수학적 접근법을 제시하였다. 요구사항의 확장성 및 관리를 위해 계층별로 분류하고 각 계층에 포함된 요소들에 대한 중요도를 산출하는 AHP 기반의 요구사항 분석 방법을 제시하였다. 수학적인 의사결정 모델인 AHP를 근간으로 하는 계산된 중요도는 기능 요소의 점수나 비기능 요소의 가중치로 사용하며, 각 서비스의 요구사항 만족도를 계산하는 데 적용된다.

또한 CRM 도메인의 서비스들에 대한 사례연구를 수행하였다. 주어진 테넌트의 서비스 요구사항에 대해 다양한 서비스들을 분석하고, 요구사항에 가장 적합한 서비스를 선택

하는 과정을 제시하였다. 마지막으로 기존의 관련 연구들과의 정성적인 비교를 수행하였다.

앞으로는 서비스 수준 협약인 SLA(Service Level Agreement)와 연계하여 서비스의 품질 수준을 협상 및 보장하는 방법에 대한 연구를 수행할 것이다. 또한 클라우드 서비스 브로커에서 테넌트의 요구사항을 충족시키기 위해 서로 다른 클라우드 서비스들을 조합해주는 중개 기술을 연구 및 본 논문에서 제안한 방법에 대해 테넌트의 만족도를 평가하는 방법에 대한 연구를 수행할 것이다.

References

- [1] T. Shin, K. Seo, "Cloud computing based fusion service market," Cloud supporting center, White Paper, 2014.
- [2] F. Liu, J. Tong, J. Mao, R. Bohn, J. Messina, L. Badger, and D. Leaf, "NIST Cloud Computing Reference Architecture," National Institute of Standards and Technology, 2011.
- [3] T. L. Saaty, L. G. Vargas, "Models, Methods, Concepts & Applications of the Analytic Hierarchy Process," Springer Science & Business Media, 2001.
- [4] T. Han, K. Sim, "An Ontology-enhanced Cloud Service Discovery System," *International Multi Conference of Engineers and Computer Scientists*, Vol.1, pp.644-649, 2010.
- [5] J. Kang, K. Sim, "A Cloud Portal with a Cloud Service Search Engine," *International Conference on Information and Intelligent Computing*, Vol.18, pp.1-8, 2011.
- [6] S. Sundareswaran, A. Squicciarini, and D. Lin, "A Brokerage-Based Approach for Cloud Service Selection," *IEEE International Conference on Cloud Computing*, pp.558-565, 2012.
- [7] L. Qu, Y. Yang, M. Orgun, L. Liu, and A. Bouguettaya, "Context-aware Cloud Service Selection based on Comparison and Aggregation of User Subjective Assessment and Objective Performance Assessment," *IEEE International Conference of Web Services*, 2014.
- [8] M. Godse, S. Mulik, "An approach for selecting software-as-a-service product," *IEEE Cloud Computing*, pp.155-158, 2009.
- [9] G. Baranwal, D. P. Vidyarthi, "A framework for selection of best cloud service provider using ranked voting method," *IEEE International Advance Computing Conference*, pp.831-837, 2014.
- [10] A. Ishikaza, M. Lusti, "How to derive priorities in AHP: a comparative study," *Central European Journal of Operations Research*, Vol.14, Issue4, pp.387-400, 2006.
- [11] Salesforce [Internet], <http://www.salesforce.com/>
- [12] Nimble [Internet], <http://www.nimble.com/>
- [13] PipelineDeals [Internet], <http://www.pipelinedeals.com/>
- [14] Telecommunications Technology Association, "TTA Standard, Quality Measurement of Software Requirement," 2010.



안 영 민

e-mail : aym41@pusan.ac.kr

2014년 부산대학교 정보컴퓨터공학부 (학사)

2014년~현 재 부산대학교 전기전자컴퓨터 공학과 석사과정

관심분야: 클라우드 컴퓨팅, 온톨로지 추론 기법, 소프트웨어 아키텍처 등



강 태 준

e-mail : roxyeris@gmail.com

2013년 부산대학교 정보컴퓨터공학부 (학사)

2015년 부산대학교 전자전기컴퓨터공학과 (석사)

관심분야: 클라우드 컴퓨팅, 모바일 시스템 등



박 준 석

e-mail : pjs50@pusan.ac.kr

1999년 부경대학교 컴퓨터공학과(학사)

2002년 부산대학교 컴퓨터공학과(석사)

2012년 부산대학교 컴퓨터공학과(박사)

2012년~2014년 부산대학교 박사후 연구원

2014년~현 재 부산대학교 물류혁신네트워크 연구소 연구교수

관심분야: 소프트웨어 재사용, 모바일 컴퓨팅, 프로덕트 라인 공학, 서비스 지향 아키텍처, 소프트웨어 아키텍처, 유비쿼터스 컴퓨팅 등



염 근 혁

e-mail : yeom@pusan.ac.kr

1985년 서울대학교 계산통계학과(학사)

1992년 플로리다대학교(Univ. of Florida) 컴퓨터공학과(공학석사)

1995년 플로리다대학교(Univ. of Florida) 컴퓨터공학과(공학박사)

1985년~1988년 금성반도체 컴퓨터연구실 연구원

1988년~1990년 금성사 정보기기연구소 주임연구원

1995년~1996년 삼성SDS 정보기술연구소 책임연구원

1996년~현 재 부산대학교 정보컴퓨터공학부 교수

관심분야: 소프트웨어 재사용, 클라우드 컴퓨팅, 빅데이터 서비스, 센서 네트워크 기반 상황 인식 미들웨어, 적응형 소프트웨어 개발 등