ISSN:0975-9646

Deepika Puvvula et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 3 (3) , 2012,4225-4231

QoS Based Web Service Selection using Credibility Evaluation

Deepika Puvvula[#1], Sai Charan Dhatrika [*2], Alekhya Varikuti [#1]

[#1]*Department of Computer Science and Engineering,*
GITAM University,
Visakhapatnam, A.P., India

[*2] *Bharat Institute of Engg. & Tech.,*
R.R.Dist., Hyderabad, A.P., India

***Abstract*— The Software as a Service (SaaS) model, where software is delivered on-demand and priced on-use, has been made possible by the widespread adoption of fast Internet access, combined with the widespread acceptance of Service oriented architecture (SOA) based solutions. By integrating outsourced software into project development can be challenging or even risky. The trustworthiness of service providers is commonly measured by their reputations. There exist so many web services that share same functional properties(reputations); so it is often a challenging effort to select a credible web service based on their various history Quality of Service (QoS) records. In view of this challenge, a QoS based service selection method is considered based on credibility evaluation associated with negotiable and nonnegotiable QoS dimensions. More specifically, the historical empirical data, i.e., execution price, reputation and success rate of a service, are used for evaluation purpose.**

***Keywords*— QoS, service selection, credibility evaluation.**

I. INTRODUCTION

There exist so many software's that share similar functional properties, so it is often a challenging effort to select credible and optimal software based on their various history of Quality of Service(QoS) records. Service oriented architecture (SOA) [1] has emerged as a fundamental architectural model that supports the overall paradigm of service computing from architecture perspective, and the web service technology is a popular technique to realize the SOA model. To initiate a service-provisioning relationship, the client first identifies the service it desires, then arranges the permissions required for the service to be delivered to the point at which it wishes to access it, typically the client's own interface to the Internet. This may mean entering into one or more service provisioning relationships, possibly governed by formalized agreements. A client is exposed to two major risks when entering an electronic-service outsourcing relationship: First, the service may not meet some requirements necessary to deliver the value that the client expected to receive as a consequence of using the service. This will result in a cost to the client, either directly or in terms of lost revenue. Second, the client will usually have to make an initial investment to acquire or implement client software capable of using the service, or more generally to integrate the service into its IT infrastructure.

Web service is a software application accessible to the user over the web. Although the current web service architecture supports registry, discovery, and consumption of web services, how to effectively select a web service which satisfies a user's requirement remains to be a challenge, as there are so many services that share similar functionalities. QoS has been studied a lot and is applied effectively in service discovery [2], service selection [3] and service composition [4]. Generally, QoS could be utilized to discriminate multiple functional equivalent web services, and the best one would be selected and returned to the user. In most web service selection methods, it is often assumed that the QoS information offered by providers is fixed and trusted, which may not be practical in certain situations.

Web services encapsulate application functionality and information resources, and make them available through programmatic interfaces, as opposed to the interfaces typically provided by traditional Web applications which are intended for manual interactions. Since they are intended to be discovered and used by other applications across the Web, Web services need to be described and understood both in terms of functional capabilities and Quality of Service (QoS) properties. In the presence of multiple Web services with identical functionality, users will discriminate these alternatives based on their QoS. QoS is a broad concept that encompasses a number of nonfunctional properties such as price, availability, reliability, and reputation. These properties apply both to standalone Web services and to Web services composed of other Web services (i.e., composite Web services).

For example, the execution time of a service is undetermined until the service finishes its execution. So for the purpose of flexibility, the execution time of a service may be given with value range, e.g., [1s, 5s]. In this situation, if all the functional qualified candidates are within the same range of execution time, e.g., [1s, 5s] (here, only one criterion execution time is considered), then it would be difficult to evaluate all the functional qualified alternatives. Furthermore, the trustworthiness of the QoS information given by service providers may not be assured, as some providers are apt to provide inauthentic QoS information, in order to attract more potential end-users. In this situation, how to cope with these fraud cases is still a challenge.

In view of these challenges, a new criterion named credibility is used, to evaluate the actual quality of a service. For each functional qualified candidate, its credibility is calculated, and the candidate that achieves the largest credibility is selected finally.

II. Web Service Quality Model

In a Web environment, several Web services may provide similar functionality with different non functional property values (e.g., different prices). In the composition model, Web services will typically be grouped together in a single community. To differentiate the members of a community during service selection, their non-functional properties need to be considered. For this purpose, a Web service quality model based on a set of quality criteria(i.e., non-functional properties) are applicable to all Web services, for example, their pricing and reliability. Although the adopted quality model has a limited number of criteria, it is extensible: new criteria can be added without fundamentally altering the service selection techniques built on top of the model. In particular, it is possible to extend the quality model to integrate non-functional service.

The Quality Criteria for Services:

1. Execution Price: Given an operation (*op*) of a service (*s*), the execution price $q_{pr}(s, op)$ is the fee that a service requestor has to pay for invoking the operation (*op*).

2. Execution Duration: The execution duration $q_{du}(p)$ of an execution plan (*p*) is computed using the Critical Path Algorithm (CPA). The CPA is applied to the execution path W_e of execution plan *p*, as a project digraph. The critical path of a project digraph is a path from the initial state to the final state which has the longest total sum of weights labeling its nodes. A node corresponds to a task *t* in W_e, and its weight is the execution duration of the service operation invoked by *t*, that is: $q_{du}(sv_p(t), op(t))$, where $sv_p(t)$ is the service assigned to task *t* in plan *p*, and $op(t)$ denotes the operation invoked by task *t*. A task that belongs to the critical path is called a critical task, while a service assigned to a task that belongs to the critical path is called a critical service.

Figure 1 provides an example of a critical path. This figure depicts an execution path as a project digraph, and an associated execution plan p, where

p = { <t1; s1 >;<t2; s2 >;<t3; s3 >;<t4; s4 >;<t5; s5> }

For each service, its execution duration is shown next to it. There are two project paths in this project digraph, where project path 1 is < *t1; t4; t5* > and project path 2 is < *t2; t3; t4; t5* > . The execution time of project path 1 is 37 seconds and project path 2 is 62 seconds. The critical path is therefore path 2 and the execution duration of the plan is 62 seconds. Task t2, t3, t4 and t5 are critical tasks while services s2, s3, s4 and s5 are critical services.

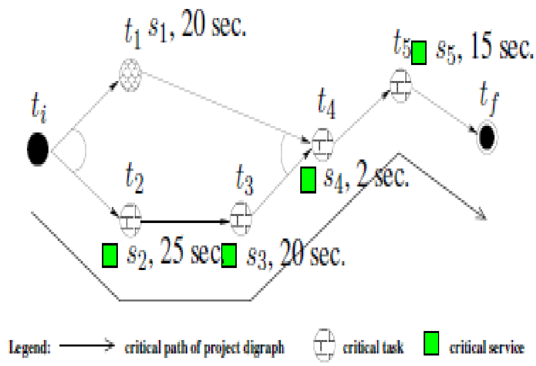

Figure 1 : Example of Critical Path

3. Reputation: The reputation $q_{rep}(p)$ of an execution plan (*p*) is the average of the reputations of the services that participate in *p*.

$$q_{rep}(s) = (\sum_{i=1}^{N} q_{rep}(S_i)) / N$$

4. Successful Execution Rate (Reliability)*:* $q_{rat}(s)$ is the probability that a request is correctly responded within maximum expected time frame indicated in web service description.

$$q_{rat}(s) = N_c(s) / k$$

$N_c(s)$ is the number of times the service has been successfully completed within maximum expected time frame.

k is total number of invocations.

5. Availability: $q_{av}(s)$ is the probability that the service is accessible.

$$q_{av}(s) = T_a(s) / \theta$$

$q_{av}(s)$ is the total amount of time(in sec) in which service (*s*) is available during last θ seconds.

III. QoS Specifications

There exist many software's that share similar functional properties; so it is often a challenging effort to select credible and optimal software based on their various history QoS records. The QoS dimensions could be classified into the following two categories, i.e., negotiable dimensions and nonnegotiable dimensions.

Negotiable dimensions: The value of a negotiable dimension may vary at runtime according to the service requestor's requirements.
Nonnegotiable dimensions: The value of a nonnegotiable dimension of a service is determined by its historical execution records and cannot be modified by the provider.
For instance, price is a negotiable dimension, as a user may accept a higher price for higher quality of service, e.g., less execution time or higher availability. On the contrary, the successful execution rate and reputation of the service are nonnegotiable, as their values cannot be determined by the service provider.
Suppose there are N nonnegotiable QoS dimensions, i.e., $q_n (1 \le n \le N)$ and M negotiable QoS dimensions, i.e., $q_m (N+1 \le m \le N+M)$. In service selection, QoS requirements are expressed in a range which is more accustomed to users than a precise value.

Definition 1(Service Request in QoS : RQ)
$RQ = (rq_1,..., rq_i,..., rq_I)$,where rq_i represents the constraints on q_i in service request. The values of rq_i are denoted by the range $rq_i = [q_i^{\min}, q_i^{\max}]$ $(1 \le i \le I)$, where $q_i^{\min}$ and $q_i^{\max}$ are lower and upper bounds of the closed interval.
For example, a user may expect the service price in the range of [10$,15$].

Definition 2 (Execution Log on QoS : EL)
The execution log is used to record the execution history of a service.
The execution records are formalized as follows:
EL = < nonneg, neg, timestap >
where *nonneg* $= (qv_1,....qv_n,....,qv_N)$, qv_N represents the value of nonnegotiable dimension of service execution,
neg = $(qv_{N+1},....qv_m,....,qv_{N+M})$, qv_m represents the value of negotiable dimension of service execution and the timestamp records the time when the execution happened.
For example, successful execution rate is a nonnegotiable dimension and its value in EL is true or false.
The considered QOS dimensions are: Price, Execution time, Reputation, Successful rate, Availability.
With the continuing proliferation of decision methods and their modifications, it is important to have an understanding of their comparative value. Each of the methods uses numeric techniques to help decision makers choose among a discrete set of alternative decisions. This is achieved on the basis of the impact of the alternatives on certain criteria and thereby on the overall utility of the decision maker(s). To select from different criteria, the Multi criteria decision making (MCDM) method is considered.

A. MCDM (Multi criteria decision making)
Multiple-criteria decision-making or multiple-criteria decision analysis is a sub-discipline of operations research that explicitly considers multiple criteria in decision-making environments. Cost or price is usually one of the main criteria. MCDM is concerned with structuring and solving decision and planning problems involving multiple criteria. The purpose is to support decision makers facing such problems. Typically, there does not exist a unique optimal solution for such problems and it is necessary to use decision maker's preferences to differentiate between solutions.
MCDM has number of problem solving techniques and one of them is TOPSIS (Technique for Order Preference by Similarity to the Ideal Solution).

B. TOPSIS (Technique for Order Preference by Similarity to the Ideal Solution)
TOPSIS is a technique for solving decision problems and is efficiency analysis in the field of Operations Research. The objective of this technique is to determine the relative advantages of alternatives. A situation that can be applied in the TOPSIS for example, the purchase of a car.
In this method two artificial alternatives are hypothesized:
Ideal alternative: the one which has the best level for all attributes considered.
Negative ideal alternative: the one which has the worst attribute values.
TOPSIS selects the alternative that is the closest to the ideal solution and farthest from negative ideal alternative.
The Euclidean distance approach is used to evaluate the relative closeness of alternatives to the ideal solution. Thus, the preference order of alternatives is yielded through comparing these relative distances.

IV. CREDIBILITY EVALUATION
It is difficult to determine the quality level of web service before execution and execution log offers information to evaluate a service quality. Credibility [5] is used for evaluating quality of service by mining its execution log. The credibility of a service is computed based on its historical execution log.

The selection method mainly consists of the following four steps:
Step 1: Data preparation. Get the constraints and preferences of QoS from the consumer and extract the QoS information from the execution log of each product service.
Step 2: Calculate the credibility of nonnegotiable QoS dimensions for each product service.
Step 3: Calculate the credibility of negotiable QoS dimensions for each product service.
Step 4: Software selection based on credibility evaluation. According to the calculation results in Step2 and Step3, synthesize the final credibility of each provider service, and select a service with largest credibility.

Step 1 : Data Preparation

The evaluation is based on the execution log, the QoS data should be extracted from execution logs of candidate services and put in matrix form according to Definition1. The user request play an important role in service selection and the QoS constraints are acquired based on Definition 2.

$$A = \begin{array}{c} \\ EL_1 \\ EL_2 \\ . \\ . \\ . \\ EL_K \end{array} \overset{\begin{array}{cccc} q_1 & q_2 & \dots\dots & q_I \end{array}}{\begin{bmatrix} x_{11} & x_{12} & \dots & x_{1I} \\ x_{21} & x_{22} & \dots & x_{2I} \\ . & & & \\ . & & & \\ . & & & \\ x_{K1} & x_{K2} & \dots & x_{KI} \end{bmatrix}}$$

$$RQ = (rq_1, rq_2, \dots, rq_i, \dots, rq_I)$$

$$W = (\omega_1, \omega_2, \dots, \omega_I)$$

where $EL_1, EL_2, \dots EL_K$ are pieces of execution logs on QoS in historical executions, x_{ij} is the value of execution log EL_i with respect to the QoS dimension q_j and ω_i is the weight of QoS dimension q_i.

Step 2 : Credibility of nonnegotiable QoS dimensions

For a nonnegotiable QoS dimension, the average value of execution records [6] describes its credibility. If the average value does not satisfy the user's QoS constraints, then the service will be abandoned.

For example, if successful execution rate in request constraint is 95% while the value of candidate service's success rate is 90%, then the service will be discarded.

The credibility of nonnegotiable QoS dimension is computed by:

$$C_{q_n} = \begin{cases} \left(\frac{1}{K} \sum_{x=1}^{K} q_{n,x} - q_{n,\min} \right) \Big/ \left(q_{n,\max} - q_{n,\min} \right), & q_{n,\max} \neq q_{n,\min} \\ 1 & , q_{n,\max} = q_{n,\min} \end{cases}$$

where $1 \le n \le N$, K represents the number of execution records, $q_{n,\max}$ and $q_{n,\min}$ represents the best value and worst value for q_n, .The value of C_{q_n} is in the range [0,1]. For example, $C_{rate} = T / K$, C_{rate} represents the credibility of successful execution rate of a candidate service and T denotes the number of times the service is successfully completed; K denotes the number of times the service has been called.

Step 3 : Credibility of negotiable QoS dimensions

The quantity of execution records satisfying the user's constraints over the negotiable dimensions and also the quality of the executions are considered to calculate the credibility of negotiable dimensions.

The MCDM(Multiple criteria decision making) technique is applied here. MCDM is concerned with structuring and solving decisions and planning problems involving multiple criteria. There are multiple conflicting criteria that need to be evaluated in making decisions. Here, TOPSIS (Technique for order performance by similarity to ideal solution) method is considered. The main principle of this method is that, the choosen alternative should be as close to the positive ideal solution (PIS) and as far from the negative ideal solution (NIS) as possible.

PIS is the user's largest satisfaction degree based on the service request (RQ).

NIS is the user's lowest satisfaction degree based on the service request (RQ).

With the derived PIS and NIS, the satisfaction degree (SD) of each candidate service could be calculated by $SD = (d^-)/(d^* + d^-)$

where, d^- represents the distance between the service's quality and NIS

d^* represents the distance between the service's quality and PIS.

The credibility of negotiable dimensions of a candidate service is calculated as follows :

i) To compute the quantity of executions which satisfy the user's QoS constraint. For each item of execution log i.e., EL_i with $1 \le i \le K$, *TF(* EL_i *)* is used to indicate whether a service can satisfy user's constraints on negotiable dimensions. The value of *TF(* EL_i *)* is true or false.

If *TF(* EL_i *)* is true, then mark this item of execution log and the value of $Count_i$ increases by one. The number of execution records that satisfy the request (RQ) is represented in $Count_i$.

$Count_i : C_i = Count_i / \max\{Count_1, \dots, Count_L\}$, where L represents the number of candidate services.

ii) The average of the marked items is calculated in the execution log. Suppose that AVG_i is expressed as the average value, then $AVG_i = (aq_{N+1}, \dots aq_m, \dots, aq_{N+M})$ and $aq_m = (\sum_{x=1}^{Count_i} A_{x,m}) / Count_i$, for *TF(* EL_i *)* is true and $Count_i \neq 0$.

According to the descriptions and calculations of *PIS, NIS* and AVG_i, the credibility of quality for service S_i can be calculated as,

$$Q_i = DIST\ (AVG_i(x), NIS\ (x) / (DIST\ (AVG_i, NIS\ ) + DIST\ (AVG_i, PIS\ )) - (2)$$

$$DIST\ (AVG_i, NIS\ ) = (\sum_1^M d\ (AVG_i(x), NIS\ (x))^2)^{1/2} - (3)$$

$$DIST\ (AVG_i, PIS\ ) = (\sum_1^M d\ (AVG_i(x), PIS\ (x))^2)^{1/2} - (4)$$

where *DIST(.,.)* is expressed as the scaled Euclidean distance between the points in M-dimensional space. Here, $d(AVG_i(x), NIS(x))$ and $d(AVG_i(x), PIS(x))$ are scaled Euclidean distances of AVG_i from *NIS* and *PIS* with respect to *x-th* dimension.

where ,

$$d\ (AVG_i(x), NIS\ (x)) = (AVG_i(x) - NIS\ (x)) / (PIS\ (x) - NIS\ (x))$$

$$d\ (AVG_i(x), PIS\ (x)) = (PIS\ (x) - AVG_i(x)) / (PIS\ (x) - NIS\ (x))$$

iii) The credibility of negotiable QoS dimensions of candidate service can be calculated by $S_i . C_i$ and Q_i which are already calculated. C_i denotes the quantity of executions that satisfy the user constraints. and Q_i represents the satisfaction degree of the quality.

The credibility of negotiable QoS dimensions can be calculated as,

$$C_{neg}\ (S_i) = \alpha C_i + \beta Q_i - (5)$$

where α and β are variants indicating quantity and quality.

Step 4 : Selection based on credibility evaluation

The final credibility of service S_i can be calculated by combining the credibility of negotiable QoS dimensions and nonnegotiable QoS dimensions. The preferences on QoS dimensions can be transformed to weights i.e., $W = (\omega_1, \omega ,...., \omega_I)$ with $\omega_1 + \omega_2 + ... + \omega_I = 1$.

The *Credibility(* S_i *)* is calculated for each service by,

Credibility(S_i *)*

$$= \sum_{n=1}^{N} w_n C_{q_n} (S_i)) + (\sum_{m=N+1}^{N+M} W_n) C_{neg}\ (S_i) - (6)$$

Among all the candidate services, the service with the highest value of credibility is selected.

V. EXPERIMENTAL RESULTS

We consider an application with three candidate services and apply the credibility evaluation method to select one service.

Step 1 : Data Preparation

We consider two nonnegotiable QoS dimensions, q_1= successful execution rate and q_2 = reputation and three negotiable QoS dimensions q_3= price, q_4 = availability and q_5= execution time. The user will give the QoS constraints and preferences on each Qos dimension.

Suppose the service request constraints on QoS and the weights are given in Table-I

Table-I QoS dimensions by user constraints and weights

dimension	q_1 (0-1)	q_2 (0-1)	q_3 ($)	q_4 (0-1)	q_5 (sec)
constraint	[0.8,1]	[0.85,1]	[10,15]	[0.85,1]	[3,5]
Weight	0.3	0.1	0.2	0.2	0.2

The QoS information is extracted from the service execution log of the three candidate service S_1, S_2, S_3 and if there are eight execution records for each candidate service and the QoS values are proposed as in Table-II.

Table-II QoS data of candidate services

Log Index	q_1	q_2	q_3	q_4	q_5
S1-1	T	0.9	10	0.8	5
S1-2	T	0.95	15	0.95	4.1
S1-3	T	0.8	12	0.9	4.5
S1-4	T	0.75	10	0.8	5.2
S1-5	T	0.9	12	0.85	4.2
S1-6	F	0.6	12	0.9	9
S1-7	T	0.95	15	0.95	4
S1-8	T	0.95	10	0.8	4.6

Log Index	q_1	q_2	q_3	q_4	q_5
S2-1	T	0.85	13	0.85	4.7
S2-2	T	0.9	15	0.9	4.4
S2-3	T	0.95	18	0.95	4.2
S2-4	T	0.75	13	0.85	5.2
S2-5	T	0.8	18	0.95	4
S2-6	T	0.95	15	0.85	4.4
S2-7	T	0.8	13	0.8	5.1
S2-8	F	0.6	15	0.9	10
S3-1	T	0.9	13	0.85	4.2
S3-2	T	0.9	11	0.8	5
S3-3	F	0.6	15	0.95	8
S3-4	T	0.8	13	0.85	4.3
S3-5	T	0.9	16	0.95	4.0
S3-6	T	0.8	15	0.9	5
S3-7	T	0.75	13	0.85	4.1
S3-8	F	0.6	11	0.8	10

Step 2 : Credibility of non negotiable QoS dimensions

The credibility of nonnegotiable QoS dimensions is calculated by (1). For example, for the constraint of reputation,

$Cq_2(S_1)$
=(0.9+0.95+0.8+0.75+0.9+0.6+0.95+0.95)/8
= 0.85

As the constraint on success rate in request *RQ* is [0.8,1] and 0.75 which does not satisfy the user constraint i.e., [0.8,1], and hence service S_3 will be discarded.

Step 3 : Credibility of negotiable QoS dimensions

To compute the credibility of negotiable QoS dimensions by using formula (2), (3) and (4), suppose α = 0.3 and β = 0.7.

PIS is the user's largest satisfaction and NIS is the user's lowest satisfaction.
PIS = (10, 1, 3) and NIS = (15, 0.85, 5).

Take Service S_1 for example, the calculation process is as follows: The number of execution records that satisfy the request (RQ) is represented in $Count_i$

The records that satisfy the user constraints are S1-2, S1-3, S1-5 and S1-7.

$Count_1$ = 4

$$C_1 = 4 / \max\{4,3\} = 1$$

To evaluate the quality of executions

$$AVG_1(q_3) = (15+12+15+12)/4 = 13.5$$

$$AVG_1(q_4) = (0.95+0.9+0.85+0.95)/4 = 0.9125$$

$$AVG_1(q_5) = (4.1+4.5+4.2+4)/4 = 4.2$$

$DIST(AVG_1, NIS)$ = 0.6504

$DIST(AVG_1, NIS)$ = 1.091

Finally, quality is calculated by,

$$Q_1 = DIST(AVG_1(x), NIS(x) / (DIST(AVG_1, NIS) + DIST(AVG_1, PIS))$$

= 0.6504 / (0.6504 + 1.091) = 0.374

The credibility of negotiable QoS dimensions can be calculated by (5) as,

$C_{neg}(S_1)$ = (0.3) 1.0 + (0.7) 0.374 = 0.5618

Step 4 : Selection based on Credibility Evaluation

The credibility of each candidate service can be calculated by (6) as,

Credibility(S_1) = (0.3) 0.875 + (0.1) 0.85 + (0.6) 0.56 = 0.6847

Credibility(S_2) = (0.3) 0.875 + (0.1) 0.825 + (0.6) 0.30075 = 0.4932

After credibility calculation, *Credibility(* S_1) > *Credibility(* S_2) and hence service S_1 is selected.

VI. CONCLUSION

A criterion named credibility is proposed to evaluate the quality of a service and the degree of satisfaction of a service requestor. As different users usually have different QoS requirements, it is inappropriate and unreasonable to compute the credibility of services based on the potentially unfair and subjective users' feedbacks. The calculation of services' credibility is mainly based on the requestors' special QoS requirements and the execution log, which is more credible to describe the trustworthiness of these services. TOPSIS method are adopted for service selection, i.e., a multi-criteria decision making problem. The web service selection method based on credibility evaluation is proved to be feasible and efficient. Firstly, the credibility of different QoS dimensions are computed, according to the correlation of some dimensions and the user's special service request. Then the aggregated credibility of each candidate service is calculated and the service with the highest credibility is chosen.

REFERENCES

[1] "Service-Oriented Architecture (SOA): concepts, technology, and design", Prentice Hall, 2004.
[2] Ziqiang Xu, Patrick Martin, Wendy Powley, Farhana H. Zulkernine, "Reputation-Enhanced QoS-based web Services Discovery", ICWS,pp.249-256, 2007.
[3] San-Yih Hwang, Ee-Peng Lim, Chien-Hsiang Lee, Cheng-Hung Chen, "Dynamic Web Service Selection for Reliable Web Service Composition", IEEE Trans. Services Computing, Vol. 1, No. 2,pp.104-116, April-June 2008.
[4] Liangzhao Zeng, Boualem Benatallah, Anne H. H. Ngu, Marlon Dumas, Jayant Kalagnanam, Henry Chang, "QoS-Aware Middleware for Web Services Composition", IEEE Trans. Software Eng., Vol.30, No.5, pp.311-327, May 2004.
[5] N. Limam, R Boutaba, "Assessing Software Service Quality andTrustworthiness at Selection Time", IEEE Tran. Software Engineering, Vol.99, No.1, 2010.
[6] L-. H. Vu, M. Hauswirth, and K. Aberer, "QoS-based Service Selection and Ranking with Trust and Reputation Management", Proceedings of OTM'05, R. Meersman and Z. Tari (Eds.), p.p. 466-483, 2005.

AUTHORS BIOGRAPHY

Deepika Puvvula is pursuing M.Tech in Software Engineering from GITAM UNIVERSITY, Visakhapatnam, A.P., INDIA. Her research areas include Software Quality Assurance, Software Reliability and Software Cost Estimation Techniques.

Sai Charan Dhatrika is working as Asst.Professor, in CSE Department, Bharat Institute of Engineering and Technology, Ibrahimpatnam, R.R.Dist., Hyderabad, A.P., India. He has received his M.Tech in Computer Science and Technology from GITAM UNIVERSITY, Visakhapatnam, A.P., INDIA. His research areas include Database security, Image processing and Software Quality Assurance.

Alekhya Varikuti is pursuing M.Tech in Software Engineering from GITAM UNIVERSITY, Visakhapatnam, A.P., INDIA. Her research areas include Software Testing, Software Quality Assurance, and Software Reliability.