ISSN 2320-5407 International Journal of Advanced Research (2014), Volume 2, Issue 4 ,1128-1136

ISSN NO. 2320-5407

Journal homepage: http://www.journalijar.com

INTERNATIONAL JOURNAL OF ADVANCED RESEARCH

RESEARCH ARTICLE

Better Ranking Of Qos Feedback System in Cloud Computing.

Takhellambam Martin Singh, MTech Md Nurul Hasan, MTech Subhranshu sekharTripathy, MTech

Syed Irfan Yaqoob,MTech

Department of computer science, Bharath University

Manuscript Info

Manuscript History:

Received: 12 February 2014
Final Accepted: 25 March 2014
Published Online: April 2014

Key words:
cloud service, ranking prediction, Quality-of-service, personalization

***Corresponding Author**

Takhellambam Martin Singh

Abstract

Cloud computing is shared configurable resources which are provided to computers and other devices as services and it is an internet-based computing. With the rapid growth of cloud computing, it is quickly becoming popular in recent years. Many leading industrial organizations such as Amazon, IBM and HP started to offer cloud services to various consumers. With the rising popularity of cloud computing, how to build high-quality cloud applications becomes required research problem. QoS rankings gives valuable information for selection of optimal cloud service from a set of functionally equivalent service candidates. To obtain QoS values, real-world invocations on the service candidates are usually required and it's time-consuming and expensive. To avoid this expensive and time consuming real world service invocations, a novel framework for ranking of cloud services is proposed by taking the advantage of the past service usage experiences of other consumers. QoS ranking To predict the QoS rankings directly, two personalized prediction approaches are proposed in this paper. Experiments are conducted employing real world QoS data and the results show that our approaches provides best result then any other competing approaches.

1. INTRODUCTION

"A cloud is a type of parallel and distributed systems consisting of a collection of inter connected and virtualised computers that are dynamically provisioned and presented as one are more unified computing resources based on service level agreements established through negotiation between the service provider and customer." With the rapid growth of cloud computing, it is quickly becoming popular in recent years. Many leading industrial organizations such as Amazon, IBM and HP started to offer cloud services to various consumers. Cloud applications deployed in the cloud environment are typically complex and large scale. With the rising popularity of cloud computing, how to build high-quality cloud applications becomes required research problem. Cloud application involve many cloud component which are communicating with each other through application programming interfaces, such as through web services like traditional component-based system. Fig. 1 shows an example of cloud applications. As shown in the figure, Cloud application1 is deployed in the cloud say some tourism website deployed in the cloud which provides various tourism services to customers. This cloud application is composed by a number of software components. Some of this components invoked other cloud services as shown in fig.1. These cloud services are provided and deployed in the cloud by some other organizations, can also be employed by other cloud applications (example: cloud application 2 and cloud application 3 as shown in Fig. 1). Since there are a number of cloud services which are functionally equivalent in the cloud, selection of optimal service is important. In this paper, the user who use the cloud service refer to cloud applications. In the context of a service invocation, server side refers to the cloud services and the user-side or client side refers to the cloud applications. Quality-of-service (QoS) are usually describe non functional performance of cloud services. Selection

of optimal cloud service from a set of services which are functionally equivalent, Quality-of-service values of cloud services gives valuable information in decision making. Software components are invoked locally in component-based systems, while cloud services are invoked remotely through internet connections in cloud applications. Thus performance of cloud services in client-side is greatly influenced by the unpredictable Internet connections. Therefore, different levels of quality for the same cloud service may receive by different cloud applications.
Thus different cloud applications required personalized cloud service Quality-of-service (QoS) ranking. To obtain Quality-of-service (QoS) values, real-world invocations on the service candidates are usually required and it's time-consuming and expensive. To avoid this expensive and time consuming real world service invocations, a personalized ranking prediction framework(Cloud Rank) is proposed to predict the Quality-of-service (QoS) ranking of a set of cloud services by taking the advantage of the past service usage experiences of other consumers without requiring additional real-world service invocations from the intended users.

1. RELATED WORK

QoS-Aware Web Service Recommendation by Collaborative Filtering

AUTHOR: Zibin Zheng Hao Ma, Michael R. Lyu, and Irwin King

With increasing presence and adoption of Web services on the World Wide Web, Quality-of-Service (QoS) is becoming important for describing nonfunctional characteristics of Web services. In this paper, we present a collaborative filtering approach for predicting QoS values of Web services and making Web service recommendation by taking advantages of past usage experiences of service users. We first propose a user-collaborative mechanism for past Web service QoS information collection from different service users. Then, based on the collected QoS data, a collaborative filtering approach is designed to predict Web service QoS values. Finally, a prototype called WSRec is implemented by Java language and deployed to the Internet for conducting real-world experiments. To study the QoS value prediction accuracy of our approach, 1.5 millions Web service invocation results are collected from 150 service users in 24 countries on 100 real-world Web services in 22 countries. The experimental results show that our algorithm achieves better prediction accuracy than other approaches. Our Web service QoS data set is publicly released for future research.

An Automatic Weighting Scheme for Collaborative Filtering

AUTHOR: Rong Jin, Joyce Y. Chai and Luo Si

Collaborative filtering identifies information interest of a particular user based on the information provided by other similar users. The memory-based approaches for collaborative filtering (e.g., Pearson correlation coefficient approach) identify the similarity between two users by comparing their ratings on a set of items. In these approaches, different items are weighted either equally or by some predefined functions. The impact of rating discrepancies among different users has not been taken into consideration. For example, an item that is highly favored by most users should have a smaller impact on the user-similarity than an item for which different types of users tend to give different ratings. Even though simple weighting methods such as variance weighting try to address this problem, empirical studies have shown that they are ineffective in improving the performance of collaborative filtering. In this paper, we present an optimization algorithm to automatically compute the weights for different items based on their ratings from training users. More specifically, the new weighting scheme will create a clustered distribution for user vectors in the item space by bringing users of similar interests' closer and separating users of different interests more distant. Empirical studies over two datasets have shown that our new weighting scheme substantially improves the performance of the Pearson correlation coefficient method for collaborative filtering.

Cumulated Gain-Based Evaluation of IR Techniques

AUTHOR: kalervo jarvelin and jaana kekalainen

Modern large retrieval environments tend to overwhelm their users by their large output. Since all documents are not of equal relevance to their users, highly relevant documents should be identified and ranked first for presentation. In order to develop IR techniques in this direction, it is necessary to develop evaluation approaches and methods that credit IR methods for their ability to retrieve highly relevant documents. This can be done by extending traditional evaluation methods, that is, recall and precision based on binary relevance judgments, to graded relevance judgments. Alternatively, novel measures based on graded relevance judgments may be developed. This article proposes several novel measures that compute the cumulative gain the user obtains by examining the retrieval result up to a given ranked position. The first one accumulates the relevance scores of retrieved documents along the ranked result list. The second one is similar but applies a discount factor to the relevance scores in order to devaluate late-retrieved documents. The third one computes the relative-to-the-ideal performance of IR techniques, based on the cumulative gain they are able to yield. These novel measures are defined and discussed and their use is demonstrated in a case study using TREC data: sample system run results for 20 queries in TREC-7. As a relevance base we used novel graded relevance judgments on a four-point scale. The test results indicate that the proposed measures credit IR methods for their ability to retrieve highly relevant documents and allow testing of statistical significance of effectiveness differences. The graphs based on the measures also provide insight into the performance IR techniques and allow interpretation, for example, from the user point of view.

Performance Analysis of Cloud Computing Services for Many-Tasks Scientific Computing

AUTHOR: Alexandru Iosup, Simon Ostermann, Nezih Yigitbasi ,Radu Prodan, Thomas Fahringer and Dick Epema

Cloud computing is an emerging commercial infrastructure paradigm that promises to eliminate the need for maintaining expensive computing facilities by companies and institutes alike. Through the use of virtualization and resource time-sharing, clouds serve with a single set of physical resources a large user base with different needs. Thus, clouds have the potential to provide to their owners the benefits of an economy of scale and, at the same time, become an alternative for scientists to clusters, grids, and parallel production environments. However, the current commercial clouds have been built to support web and small database workloads, which are very different from typical scientific computing workloads. Moreover, the use of virtualization and resource time-sharing may introduce significant performance penalties for the demanding scientific computing workloads. In this work we analyze the performance of cloud computing services for scientific computing workloads. We quantify the presence in real scientific computing workloads of Many-Task Computing (MTC) users, that is, of users who employ loosely coupled applications comprising many tasks to achieve their scientific goals. Then, we perform an empirical evaluation of the performance of four commercial cloud computing services including Amazon EC2, which is currently the largest commercial cloud. Last, we compare through trace-based simulation the performance characteristics and cost models of clouds and other scientific computing platforms, for general and MTC-based scientific computing workloads. Our results indicate that the current clouds need an order of magnitude in performance improvement to be useful to the scientific community, and show which improvements should be considered first to address this discrepancy between offer and demand.

1. PROPOSED SYSTEM

QoS rankings gives valuable information for selection of optimal cloud service from a set of functionally equivalent service candidates. To obtain QoS values, real-world invocations on the service candidates are usually required and it’s time-consuming and expensive. To avoid this expensive and time consuming real world service invocations, a novel framework for ranking of cloud services is proposed by taking the advantage of the past service usage experiences of other consumers. To predict the QoS rankings directly, two personalized QoS ranking prediction approaches are proposed. Our approach takes the advantage of the past usage experiences from other users for making personalized ranking prediction for the current user. Fig. 2 shows the system architecture of our CloudRank framework, which provides personalized QoS ranking prediction for cloud services. Cloud applications are the target

users of the CloudRank framework the cloud applications, that seek personalized cloud service ranking for making selection of optimal service. As shown in Fig. 2, a user can get the service ranking prediction of available cloud services from the CloudRank framework by providing observed QoS values of some cloud services.

2. ILLUSTRATION OF FEEDBACK SESSIONS

AUTHENTICATION (Login\Register)

If you are the new user going to access the network then they have to register first by providing necessary details. After successful completion of sign up process, the user has to login into the application by providing username and exact password. The user has to provide exact username and password which was provided at the time of registration, if login success means it will take up to main page else it will remain in the login page itself.

USER PRIVILEGES

View services and feedback

A user who has authorization can enter into cloud application and use that multiple services in it. Finally user can give his feedback.

CLOUD PROVIDER

Response time

Throughput and Failure rate

While using the multiple services how the **throughput** will satisfy the user. **Failure rate** means to calculate the lack of success of services that will be provided by cloud provider at the time of detection.

Ranking

By calculating similarity values between the current active users with other training users, the similar users can be identified. Finally, cloud provider to consider about user feedback and give the ranking.

Step 1:Rank the employed cloud services based on the observed QoS values. The values of the rank are from small to large where a smaller value indicates higher quality.

Step 2:For each service in the full service set I, calculate the sum of preference values with all other services. Larger value indicates more services are less preferred than. In other words, service I should be ranked in a higher position.

Step 3:Services are ranked from the highest position to the lowest position by picking the service that has the maximum value. The selected service is assigned a rank equal to 1 so that it will be ranked above all the other remaining services. The selected service t is then removed from I and the preference sum values of the remaining services are updated to remove the effects of the selected service t.

Step 4:Step3 treats the employed services in E and the non employed service in I _ E identically which may incorrectly rank the employed services. In this step, the initial service ranking is updated by correcting the rankings of the employed services in E. By replacing the ranking results with the corresponding correct ranking, our approach makes sure that the employed services in E are correctly ranked.

1. ASSOCIATED WORK

With the rapid growth of cloud computing, it is quickly becoming popular in recent years. A number of works have been carried out on cloud computing [8], [10],including performance analysis, market-oriented cloud computing, management tool, workload balance, dynamic selection, etc. Quality-of-service has been widely employed for presenting the nonfunctional characteristics of the software systems and services [19]. QoS of cloud services can be measured from either the client side (e.g., response time, throughput, etc.) or at the server side (e.g., price, availability, etc.). Based on the service QoS measures, various approaches have been proposed for service selection [3], [19], [20], which enables optimal service to be identified from a set of functionally similar or equivalent candidates. To provide QoS ranking information for the service selection approaches, this paper focuses on predicting QoS ranking of cloud services.

Collaborative filtering methods are widely adopted in recommender systems [5], [15]. A memory-based approach is one type of the most widely studied collaborative filtering approaches. The most analyzed examples of memory-based collaborative filtering include user-based approaches [4], [9], item-based approaches [7], [11], [16], and their fusion [13], [16], [17], [22], [23]. User-based and item-based approaches often use the vector similarity method [4] and the PCC method [15] as the similarity computation methods. Compared with vector similarity, PCC considers the differences in the user rating style when calculating the similarity The rating-based collaborative filtering approaches try to predict the missing QoS values in the user-item matrix as accurately as possible. However, in the ranking-oriented scenarios, accurate missing value prediction may not lead to accuracy ranking prediction. Therefore, ranking-oriented collaborative filtering approaches are becoming more attractive. Liu and Yang [12] propose a rankingoriented collaborative filtering approach to rank movies. Yang et al. [18] propose another ranking-oriented approach for ranking books in digital libraries. Different from these previous approaches [12], [18], our work provides a comprehensive study of how to provide accurate QoS ranking for cloud services, which is a new and urgently-required research problem.\ Currently, our CloudRank framework is mainly designed for cloud applications, because: 1) client-side QoS values of different users can be easily obtained in the cloud environment; and 2) there are a lot of redundant services abundantly available in the cloud, QoS ranking of candidate services becomes important when building cloud applications. The CloudRank framework can also be extended to other component-based applications, in case that the components are used by a number of users, and the past usage experiences of different users can be obtained

2. VII EXPERIMENTAL SETUP AND RESULT

Evaluation Metric

accurate as possible. Therefore, differences between the predicted values and the true values are usually employed to evaluate the prediction accuracy. Mean Absolute Error and Root-Mean Square Error (RMSE) metrics are two widely adopted evaluation metrics for rating-oriented approaches. MAE is defined as

$$MAE = \frac{\sum_{i,j} |r_{i,j} - \widehat{r}_{i,j}|}{N},$$

and RMSE is defined as

$$RMSE = \sqrt{\frac{\sum_{i,j} (r_{i,j} - \widehat{r}_{i,j})^2}{N}},$$

where ri;j denotes the expected QoS value of service j observed by user i, bri;j is the predicted QoS value, and N is the number of predicted values. However, since the object of this paper is to predict service QoS ranking instead of predicting QoS values, we employ the Normalized Discounted Cumulative Gain (NDCG) [2] metric, which is a popular metric for evaluating ranking results. Given an ideal service QoS ranking and a predicted QoS ranking, the NDCG value of the Top-K ranked services can be
calculated by

$$NDCG_k = \frac{DCG_k}{IDCG_k},$$

where DCGk and IDCGk are the discounted cumulative gain (DCG) values of the Top-K services of the predicted ranking and ideal ranking, respectively. The value of DCGk can be calculated by

$$DCG_k = rel_1 + \sum_{i=2}^{k} \frac{rel_i}{log_2 i},$$

where reli is the QoS value of the service at position i of the ranking. The premise of DCG is that high-quality service appearing lower in a ranking list should be penalized as the QoS value is reduced logarithmically proportional to the position of the result via dividing by log2i. The DCG value is accumulated from the top of the ranking to the bottom with the gain of each result discounted at lower ranks. The ideal rank achieves the highest gain among different rankings. The NDCGk value is on the interval of 0 to 1, where larger value stands for better ranking accuracy, indicating that the predicted ranking is closer to the ideal ranking. The value of k is in the interval of 1 to n, where n is the total number of cloud services.

Here, we first conduct a simulation-based performance evaluation on the cloudrank algorithms for processing QoS rankings prediction of available cloud services. As discussed, to obtain QoS values, real-world invocations on the service candidates are usually required and it’s time-consuming and expensive. To avoid this expensive and time consuming real world service invocations, a novel framework for ranking of cloud services is proposed by taking the advantage of the past service usage experiences of other consumers. To perform the necessity test we have to take a system, which should be having the features for hardware part Processor Pentium –IV Speed will be 1.1 GHz RAM 256 MB (min) Hard Disk must be20 GB Key Board Standard Windows Keyboard Two or Three Button Mouse Monitor SVGA need. The software configuration should be. Operating System on which simulation is going to be performed is the Windows XP. Programming Language we are using is C#. DATABASE, MYS, tool is Microsoft

Visual Studio .Net 2010 to perform the desire experiment first we have to collect data from the user.

If you are the new user going to access the network then they have to register first by providing necessary details. After successful completion of sign up process, the user has to login into the application by providing username and exact password. The user has to provide exact username and password which was provided at the time of registration, if login success means it will take up to main page else it will remain in the login page itself.

A user who has authorization can enter into cloud application and use that multiple services in it. Finally user can give his feedback. The length of time it takes to react to a given stimulus or event. At the cloud provider side the response time will be computed by cloud provider while user using the services. By calculating similarity values between the current active users with other training users, the similar users can be identified. Finally, cloud provider to consider about user feedback and give the ranking.

3. CONCLUSION

We proposes a QoS ranking prediction framework for cloud services by taking advantage of the past service usage experiences of other consumers. Our proposed framework requires no additional invocations of cloud services when making QoS ranking prediction. Two personalized QoS ranking prediction approaches are proposed to predict the QoS rankings directly. In this QoS ranking prediction framework for cloud services, this requires no additional service invocations when making QoS ranking. By taking advantage of the past usage experiences of other users, our ranking approach identifies and aggregates the preferences between pairs of services to produce a ranking of services. We propose two ranking prediction algorithms for computing the service ranking based on the cloud application designer's preferences.

4. REFERENCES

1. Z. Zheng, H. Ma, M.R. Lyu, and I. King, "QoS-Aware Web Service Recommendation by Collaborative Filtering," IEEE Trans. Service Computing, vol. 4, no. 2, pp. 140-152, Apr.-June 2011

2. M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R.H. Katz, A. Konwinski, G. Lee, D.A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "Above the Clouds: A Berkeley View of Cloud Computing," Technical Report EECS-2009-28, Univ. California, Berkeley, 2009.

3. G. Linden, B. Smith, and J. York, "Amazon.Com Recommendations: Item-to-Item Collaborative Filtering," IEEE Internet Computing, vol. 7, no. 1, pp. 76-80, Jan./Feb. 2003.

4. R. Jin, J.Y. Chai, and L. Si, "An Automatic Weighting Scheme for Collaborative Filtering," Proc. 27th Int'l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR '04), pp. 337-344, 2004.

5. K.J. arvelin and J. Kekalainen, "Cumulated Gain-Based Evaluation of IR Techniques," ACM Trans. Information Systems, vol. 20, no. 4, pp. 422-446, 2002.

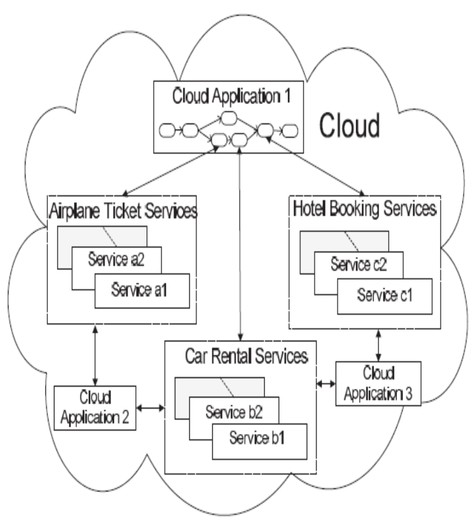
6. **.** J.S. Breese, D. Heckerman, and C. Kadie, "Empirical Analysis of Predictive Algorithms for Collaborative Filtering," Proc. 14th Ann. Conf. Uncertainty in Artificial Intelligence (UAI '98), pp. 43-52, 1998.

7. B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-Based Collaborative Filtering Recommendation Algorithms," Proc. 10th Int'l Conf. World Wide Web (WWW '01), pp. 285-295, 2001.

8. M. Deshpande and G. Karypis, "Item-Based Top-n Recommendation," ACM Trans. Information System, vol. 22, no. 1, pp. 143-177, 2004.

[9] R. Jin, J.Y. Chai, and L. Si, "An Automatic Weighting Scheme for Collaborative Filtering," Proc. 27th Int'l ACM SIGIR Conf. Research

and Development in Information Retrieval (SIGIR ’04), pp. 337-344, 2004
[10] T. Yu, Y. Zhang, and K.-J. Lin, “Efficient Algorithms for Web Services Selection with End-to-End QoS Constraints,” ACM Trans. Web, vol. 1, no. 1, pp. 1-26, 2007.
[11] Z. Zheng, Y. Zhang, and M.R. Lyu, “CloudRank: A QoS-Driven Component Ranking Framework for Cloud Computing,” Proc. Int’l Symp. Reliable Distributed Systems (SRDS ’10), pp. 184-193, 2010.
[12] Z. Zheng, H. Ma, M.R. Lyu, and I. King, “QoS-Aware Web Service Recommendation by Collaborative Filtering,” IEEE Trans. Service Computing, vol. 4, no. 2, pp. 140-152, Apr.-June 2011.
[13] Zibin Zheng, Xinmiao Wu, Yilei Zhang, Michael R. Lyu and Jianmin Wang, “QoS Ranking Prediction for Cloud Services,” IEEE Trans. Parallel and Distributed Systems, vol. 24, no. 6, June. 2013.
[14] P.A. Bonatti and P. Festa, “On Optimal Service Selection,” Proc. 14th Int’l Conf. World Wide Web (WWW ’05), pp. 530-538, 2005.
[15] J.S. Breese, D. Heckerman, and C. Kadie, “Empirical Analysis of Predictive Algorithms for Collaborative Filtering,” Proc. 14th Ann. Conf. Uncertainty in Artificial Intelligence (UAI ’98), pp. 43-52, 1998.
[16] M. Deshpande and G. Karypis, “Item-Based Top-n Recommendation,” ACM Trans. Information System, vol. 22, no. 1, pp. 143-177, 2004.
[17] R. Jin, J.Y. Chai, and L. Si, “An Automatic Weighting Scheme for Collaborative Filtering,” Proc. 27th Int’l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR ’04), pp. 337-344, 2004.
[18] G. Linden, B. Smith, and J. York, “Amazon.com Recommendations: Item-to-Item Collaborative Filtering,” IEEE Internet Computing, vol. 7, no. 1, pp. 76-80, Jan. /Feb. 2003.
[19] N.N. Liu and Q. Yang, “Eigenrank: A Ranking-Oriented Approach to Collaborative Filtering,” Proc. 31st Int’l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR ’08), pp. 83-90, 2008.
[20] H. Ma, I. King, and M.R. Lyu, “Effective Missing Data Prediction for Collaborative Filtering,” Proc. 30th Int’l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR ’07), pp. 39-46,



2007

Fig.1. Example

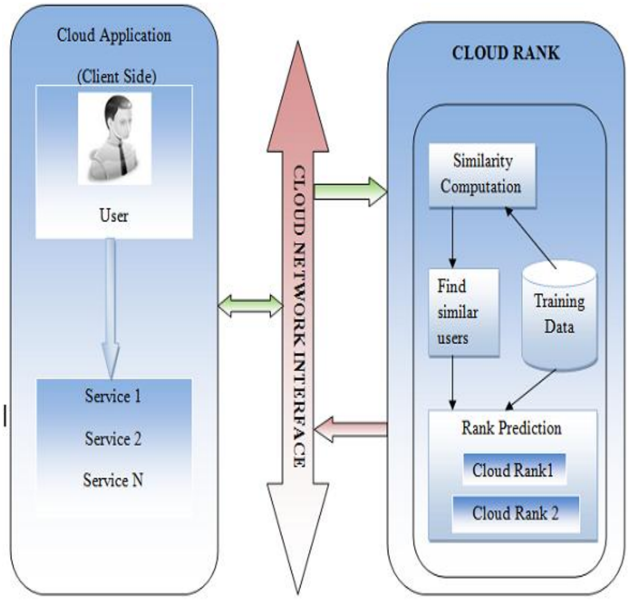


Fig.2.System architecture of CloudRank

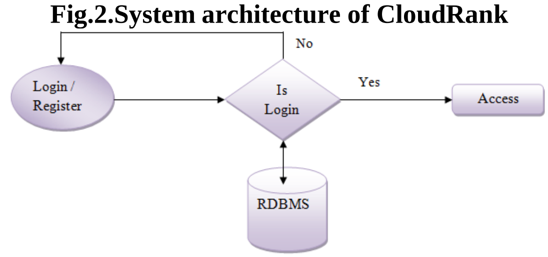


Fig.3. Authentication

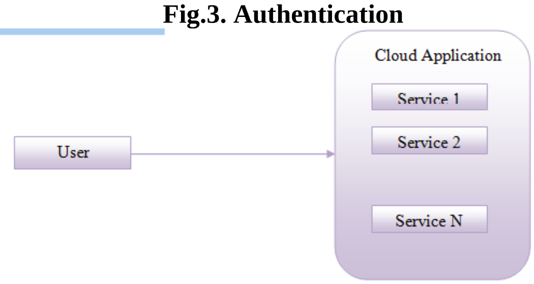
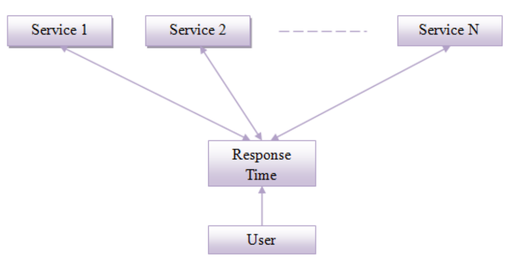


Fig.4. view services and feedback.



The length of time it takes to react to a given stimulus or event. The response time will be computed by cloud provider while user using the services.

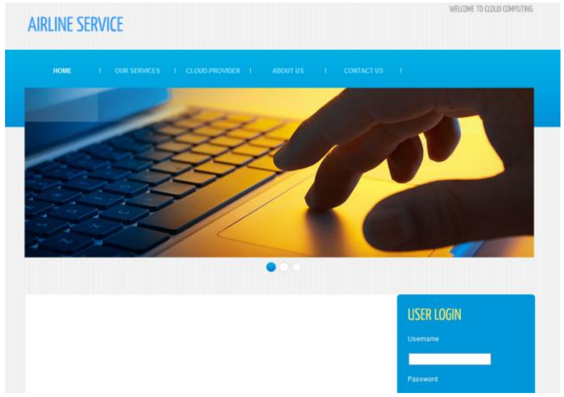


Fig.5. The design of the login page of the website

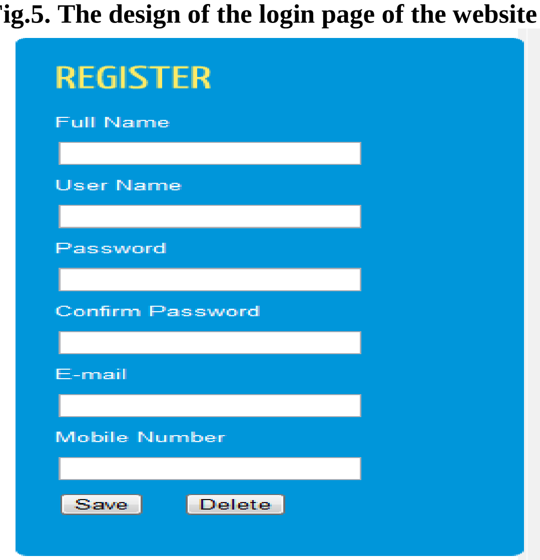


Fig.6. The design of the registration page.