Journal of Applied Mathematics and Computation (JAMC), 2018, 2(2), 42-49
http://www.hillpublisher.org/journal/jamc
ISSN Online:2576-0645
ISSN Print:2576-0653

Towards the Cloud Deployment of a Heterogeneous Service Oriented System

Festim Halili[1], Skofiar Kamberi[2], Merita Kasa Halili[3]

[1]State University of Tetovo Tetovo, Macedonia, Email: festim.halili@unite.edu.mk
[2]International Balkan University Skopje, Macedonia, Email: skofiarkamberi@gmail.com
[3]State University of Tetova Tetovo, Macedonia, Email: merita.kasa@unite.edu.mk

How to cite this paper: Halili, F., Kamberi, S., Halili, M.K. (2018) Towards the Cloud Deployment of a Heterogeneous Service Oriented System. *Journal of Applied Mathematics and Computation*, **2(2)**, 42-49.
http://dx.doi.org/10.26855/jamc.2018.02.002

Corresponding author: Skofiar Kamberi, International Balkan University Skopje, Macedonia,
Email: skofiarkamberi@gmail.com

Abstract

Nowadays cloud computing is gaining a lot of attention as a new technology, allowing us to get more storage and to meet expectations on some institutions. Based on our monitoring in our country there is a lack of implementation of the student evaluation systems in the Higher Educational Institutions, therefore we decided to develop a heterogeneous system composed of web services and deployed in IaaS cloud. Student evaluation system is a prototype which allows students to commit evaluation on instructors online. It is designed in such way that meets all the student requirements, faculty requirements and it is user friendly. We aim a heterogeneous, because it will be consisted of composite web services. Service composition indicates the development of complex applications using several web services that are coordinated. Furthermore, we have conducted analysis of deployment models and cloud types for this prototype system in the cloud computing system.

General Terms

Cloud Computing, Web Applications, Service Oriented

Keywords

Web Services, IaaS, Cloud Computing, SLA, System.

Introduction

The concept of web services (WS) has gained great importance during the past couple years. It has become essential to make applications broadly available in the World Wide Web. WS could be especially useful for the creation of dynamic e-business applications and to allow Java EE and NET technologies to interoperate. New WS standards have been developed through the cooperation of several corporations. [1] Numerous existing concepts such as business process management, security, directory services, routing and transactions are being adapted for WS. [2] Cloud Computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model is composed of five essential characteristics (On-demand self-service, Broad network access, Resource pooling, Rapid elasticity, Measured service), three service models (Software as a Service (SaaS). Platform as a Service (PaaS). Infrastructure as a Service (IaaS)) and four deployment models (Private cloud, Community cloud, Public cloud, Hybrid cloud). [3] [4] In our project we suggest of using Infrastructure as a Service as a model. We are going to give our research and arguments why we suggest implementing this model. Figure 1 is illustration of service models and representation of what each of them allows user to manage and what is managed by other parties.

Infrastructure is the foundation of cloud computing. It provides delivery of computing as a shared service reducing the investment cost, operational and maintenance of hardware. [5] Infrastructure should be reliable and flexible for easy implementation and operations of applications.

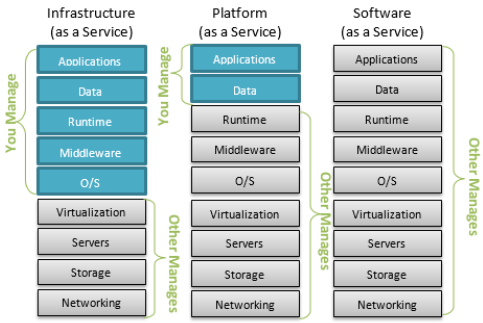

Figure 1. Service Models

Delivery of resources such as servers, storage and network components as a service. Lower total cost of ownership. Full scalability. Eliminate the need for administration and maintenance of hardware. Enterprise grades infrastructure for all subscribers. IaaS cloud offers resources such as images in a virtual-machine image -library, raw and file -based storage, firewalls, load balancers, IP addresses, virtual local networks and software bundles.
Examples of IaaS providers are Amazon cloud formation, amazon EC2, google compute engine, HP cloud, iland, joyent, oracle infrastructure as a service and rackspace cloud. [6]
The IaaS is categorized into [7]:
1) Computation as a Service (CaaS), in which virtual machine based servers are rented and charged per hour based on the virtual machine capacity – mainly CPU and RAM size, features of the virtual machine, OS and deployed software.
2) Data as a Service (DaaS), in which unlimited storage space is used to store the user's data regardless of its type, charged per GB for data size and data transfer.
Many applications simply don't run well in a pure multi-tenant server environment. Databases perform better on a dedicated server where they don't have to compete for input/output resources, and the situation is similar with web server applications. From hardware service (utility computing) point of view, there are few new aspects in cloud, the most prominent being the illusion of infinite computing resources and the ability to pay for use of computing resources on a short-term basis as needed. As consumers move towards adopting such a Service-Oriented Architecture, the quality and reliability of the services become important aspects. However, the demands of the service consumers vary significantly. It is not possible to fulfill all consumer expectations from the service provider perspective and hence a balance needs to be made via a negotiation process. At the end of the negotiation process, provider and consumer commit to an agreement. In SOA terms, this agreement is referred to as a SLA (Service Level Agreement). This SLA serves as the foundation for the expected level of service between the consumer and the provider. The QoS (Quality of Service) attributes that are generally part of an SLA (such as response time and throughput) however, change constantly and to enforce the agreement, these parameters need to be closely monitored.
Infrastructure as a Service (IaaS) is the delivery of hardware (server, storage and network), and associated software (operating systems virtualization technology, file system), as a service. It is an evolution of traditional hosting that does not require any long-term commitment and allows users to provision resources on demand. Unlike PaaS services, the IaaS provider does very little management other than keep the data center operational and users must deploy and manage the software services themselves just the way they would in their own data center. Amazon Web Services Elastic Compute Cloud (EC2) and Secure Storage Service (S3) are examples of IaaS offerings.
Infrastructure as a Service is a form of hosting. It includes network access, routing services and storage. The IaaS provider will generally provide the hardware and administrative services needed to store applications and a platform for running applications. Scaling of bandwidth, memory and storage are generally included, and vendors compete on the performance and pricing offered on their dynamic services. The service provider owns the equipment and is responsible for housing, running and maintaining it. IaaS can be purchased with either a contract or on a pay-as-you-go basis. However, most buyers consider the key benefit of IaaS to be the flexibility of the pricing, since you should only need to pay for the resources that your application delivery requires.
Characteristics and components of IaaS include:
• Utility computing service and billing model.
• Automation of administrative tasks.

• Dynamic scaling.
• Desktop virtualization.
• Policy-based services.
• Internet connectivity.

DEPLOYMENT MODELS

Public cloud or external cloud describes cloud computing in the traditional mainstream sense, whereby resources are dynamically provisioned on a fine-grained, self-service basis over the internet, via web applications/webservices from an off-site third-party provider who bill on fine-grained utility computing basis. The cloud infrastructure is made available to the general public or large industry group and is owned by an organization selling cloud services. Examples: Amazon Elastic-compute-cloud, Google AppEngine, Sun Cloud.
A community cloud may be established where several organizations have similar requirements and seek to share infrastructure so as to realize some of the benefits of the cloud computing. With the costs spread over fewer users than a public cloud (but more than a single tenant) this option is more expensive but may offer a higher level of privacy, security and/or policy compliance. Example of community cloud includes Google's "Gov Cloud".
The term "**Hybrid Cloud**" has been used to mean either two separate clouds joined together, or a combination of virtualized cloud server instances used together with real physical hardware. The most correct definition of the term "Hybrid Cloud" is probably the use of physical hardware and virtualized cloud server instances together to provide a single common service. This type of cloud uses a combination of public and private storage clouds. Hybrid clouds are often useful for backup functions allowing local data to be replicated to a public cloud.
A private cloud is a particular model of cloud computing that involves a distinct and secure cloud based environment in which only the specified client can operate. Compare to other models, private clouds will provide computing power as a service within virtualized environment using an underlying pool of physical computing resource. However, under the private cloud model, the cloud is only accessible by a single organization providing that organization with greater control and privacy. [8]

PROBLEM COMPLEXITY

A. Interface

Creating a user-friendly interface for this system was one of the goals. The system is created in such way that the user (in our case students) will not for a single step, feel lost. They have instruction on page for what they have to do. Design is simple and efficient at the same time. Figure 2, is illustration of how a table looks in the page. For each instructor there has to be created such table so that students will do evaluation on them.

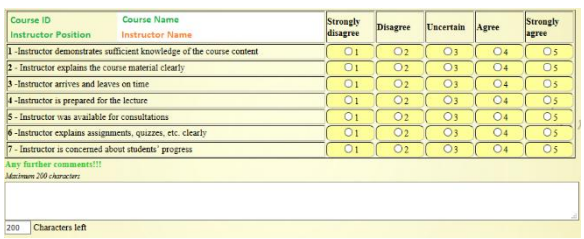

Figure 2. Instructor evaluation table

B. Web Services

In order our system to meet today requirements and today technological advancement we had to include web services. [9] We had to implement some Web Services in such way that it gives more flexibility on the software and it will allows us to reuse some of services many times. Web Services can be manually created but also there are some repositories for getting them online.
B.1 SOAP as an exchange protocol
The method of invoking web services is crucial; therefore the SOAP (Simple Object Access Protocol) is established to exchange messages between service providers and consumers. It is a structured XML message format for exchanging data in a distributed environment. It uses an underlying transport protocol (HTTP, SMTP etc) through binding. By the

time of writing this paper there are two version of SOAP: SOAP version 1.1 and SOAP version 1.2 which has brought some new benefits: It is cleaner, faster, it has better web integration and more it is versatile.
There are three main types of SOAP Nodes:

- SOAP Sender – Generates and transmits a SOAP message,
- SOAP Receiver – Receives and processes the SOAP message and it also may generate SOAP response, message or fault as a result, and
- SOAP Intermediary (Forwarding or active) – It is both, a SOAP receiver and a SOAP sender. It receives and processes the SOAP header blocks targeted at it and resends the SOAP message towards an SOAP receiver. This process is illustrated in the Figure 3 :

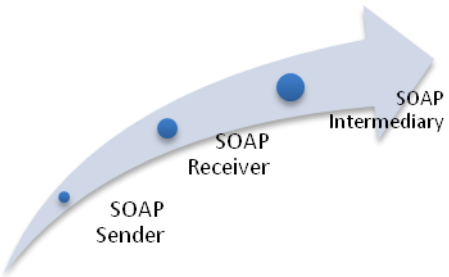

Figure 3. SOAP Nodes

The SOAP message has a structure, which is characterized with two SOAP-specific sub-elements within the overall SOAP Envelope (env:Envelope), namely a SOAP Header (env:Header) and a SOAP Body (env:Body).
To better percept the anatomy of the SOAP message, we will show an example from the development of a specific web service dedicated to an evaluation system. The following code represents a SOAP 1.2 version of a message, more particullary for the evaluation web service.
The main issue of SOAP Version 1.2 is to encapsulate remote procedure call functionality using the extensibility and versatility of XML. In the above case where the SOAP message is transmitted over the HTTP protocol, there is no header included (Refer to the above explanation) because the header is an optional element in the SOAP Envelope. The SOAP Envelope is declared using the namespace as seen in the code. In case the namespace is incorrect, a SOAP application will generate an error. Figure 7 is shown a pseudocode for a web method.

```
      POST /evaluationWS/Service.asmx HTTP/1.1
      Host: localhost
      Content-Type: application/soap+xml;
charset=utf-8
      Content-Length: length
      <?xml version="1.0" encoding="utf-8"?>
      <soap12:Envelope
      xmlns:xsi="http://www.w3.org/2001/XMLSch
      ema-instance"
      xmlns:xsd="http://www.w3.org/2001/XMLSch
      ema"
      xmlns:soap12="http://www.w3.org/2003/05/
      soap-envelope">
      <soap12:Body>
      <AddStudentToEvaluate
xmlns="http://tempuri.org/">
      <StudentID>int</StudentID>
      <ProfessorID>int</ProfessorID>
      <AdminID>int</AdminID>
      </ AddStudentToEvaluate >
      </soap12:Body>
      </soap12:Envelope>
```

Figure 7. Pseudocode for a web method

C. Database

In order to have the data all organized and have them all in one place we had to implement a well-structured database. We decided to use SQL Server as platform to create, maintain and save data. Database must be normalized still. It must go through normalization forms. We created tables for each faculty.

D. Cloud

After everything is designed, developed and tested, we now can think about deploying our system. With the trending of cloud computing, we thought that the best way for our system to deploy will be, to deploy it in cloud.

PROTOTYPE

This web application is designed in very user-friendly interface. The idea behind is to make users feel comfortable and at the same time to make them feel that their evaluation on instructors and university is safe in terms of their privacy. This web application will assure users that no private data will be collected by system. In order to prove this, we will do explanation later on. This web application is designed in different views. We created 2 different views, one for the administrator and the other one for users in this case users are students. Based on the privileges that each user has administrator or student it will be redirected to the corresponding page. These privileges are set upon LogIn page. The design of the application is done using ASP.Net, HTML and CSS. Concepts used in the design include: tables, paragraph, heading, divisions, images, textboxes, radio buttons, dropdown lists, labels, etc.

As we mentioned before this application is designed in two different views Administrator view is designed and programmed in such way that the administrator can get results from evaluation. Administrator view is available only for administrators which are predefined in database. Administrator can get general reports about university, individual instructor reports, individual department report and faculty level reports. Administrator also has permissions to save the corresponding reports as excel file. In the other hand, students don't have the same privileges as administrators do. Students can Log In using their email and their passwords. But this is subject to change depending on university needs. After students log in he/she is redirected to the first evaluation page which is general evaluation about university. Students must make a selection to proceed in the other page. They select their current academic year, semester and their department. Based on their selection they are redirected to the corresponding page.

In the log in page web services come into play. Every checking in email and passwords, checking if the student has already completed the evaluation is done through web services. We have created a web service to do these tasks, and then, we have just called it in my code in the final project.

There are three more webservices implemented into this system. There are two separate webservices consisted of two methods each. In this way in our system we are including composite webservices which makes our system heterogenous. We use SQL Server for implementing database. We use SQL Query to query data from and into database.

```
localhost2.Service1 obj = new localhost2.Service1();
DataSet ds = obj.fill();
DropDownList1.DataSource = ds;
DropDownList1.DataTextField = "ColumnName";
DropDownList1.DataValueField = "ColumnName";
DropDownList1.DataBind();
```

Figure 4. Pseudocode for triggering web method in our system prototype

CONCLUSIONS

We can conclude that student evaluation system must be included in every university in Republic of Macedonia. The reason being is that it will allow students to do evaluation online. User friendly interface will give students motivation to complete this evaluation process. As well as for administrator analyzing data from the database it is going to be done in more easy and efficient way. Getting data on different form is one of the advantages for this system. Our system can be as prototype for development of other same-like systems. Usage of web services and composition of them is an advantage why this system is efficient. We conclude that Infrastructure as a Service is going to be our model for deployment in cloud.

Results

Figure 5 illustrates the first page of our system. There is text box and radio buttons to select privileges for the user. Whenever users click on Log In Button, web service is triggered. Based on the information entered by the user, it will be redirected to corresponding page.

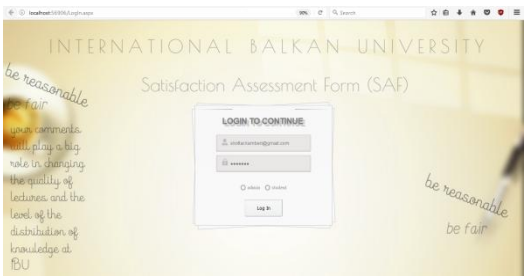

Figure 5. Log In page

Admin page (Admin panel) is shown on Figure 6. As we can see from the Figure 6, administrator has many option for analyzing data from the database. He can get individual report for each instructor, for each course, for each department and each faculty. He also can see how many students completed the evaluation, how many students are registered in total. How many students from each department have completed the evaluation also is information that only administrator can see. Another feature or advantage of our system is that data can be saved into excel file.

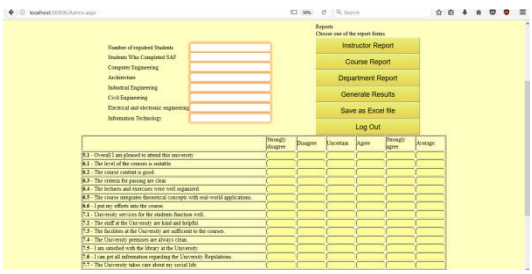

Figure 6. Admin Panel

After researching web services, seeing their structure, their benefits and their application in today's technology we decided to implement them in our system. We created two web services each containing two methods. First web service is used in the log in page. Whenever users try to log in into our system they are calling one of the two methods in our first web service. First method is related to admin log in information while second one is related to student log in information. When users enter data into the textboxes and selects one of the radio buttons on the log in page, that information is passed into web service, which then, does the comparison of entered information with data in the database. Figure 8 illustrates pseudocode for one of the web service (a web method).

```
[WebMethod]
public DataSet fill()
{
SqlConnection
 con.Open();
SqlCommand cmd = new SqlCommand(query con);
SqlDataAdapter sda = new SqlDataAdapter(cmd);
DataSet ds = new DataSet();
sda.Fill(ds);
con.Close();
return ds;
       }
```

Figure 8. pseudocode of a web method

Pseudocode which triggers this web method is shown in Figure 4. We are calling the service and then with data that we are getting from it, we are filling drop down list. Basically, drop down list is going to show us list of instructors that are in our database.

We recommend for my system to use IaaS (Infrastructure as a Service) as a model for implementing our system prototype into cloud. The reason why we choose IaaS over SaaS is that we want to have more control over the system. We want to manage and have control over application and data. While in SaaS data, application is control by vendor. In IaaS vendor is responsible over Servers, Storage, Networking and Virtualization. We found that these are enough for the vendor to take care of. IaaS gives me more balance between what we can manage and what vendor manages.

Table 1. Public Cloud vs Private Cloud

Public Cloud	**Private Cloud**
Publicly Shared Virtualized Resources	Privately Shared Virtualized Resources
Supports multiple customers	Cluster of dedicated customers
Supports connectivity over internet	Connectivity over internet, fiber and private network
Suited for less confidential information	Suited for secured confidential information and core systems

Another thing that is important while thinking of deploying in cloud is, choosing on which type of cloud to deploy. Comparison between private cloud or public cloud is shown on Table 1. Based on those results that we found, our recommendation for student evaluation system is to be deployed in private cloud. As we can see from Table 1, Private Cloud gives me more security over the data, it can be accessed by dedicated group of users. Another advantage of private cloud over public cloud is that it can be implemented over private network. Since, student evaluation system contains confidential information and data, private cloud is the best solution. So, our recommendation is to use private cloud for deployment of this system.

Conclusion

The process of development of this system prototype, introduction of one of the latest technologies (Web Services), simple and effective user interface, are one of the key areas that we discussed in our project.

We can conclude that student evaluation system must be included in every university in Republic of Macedonia. The reason being is that it will allow students to do evaluation online, it need less human resources to manage system and process of evaluation. User friendly interface will give students motivation to complete this evaluation process. As well as for administrator analyzing data from the database it is going to be done in more easy and efficient way. Getting data on different form is one of the advantages for this system. This system can be as prototype for development of other same-like systems. Usage of web services and composition of them is an advantage why this system prototype is efficient. Introduction of more web services will improve furthermore this system.

Future work

Deployment of student evaluation system in the cloud is one of the tasks that we seek to do in future. Normalizing database in all levels of normalization forms is also one of the priorities for future work. Introducing charts for different report is another feature that we tend to add in future. Implementing our system into some bigger university system is task that we look forward of achieving it in the future.

References

[1] F. Halili, A. Dika and K. M. Popovska, "Towards the Composition of Web Services and the Role of the Query Optimization," International Journal of Web Applications, vol. IV, no. 2, pp. 57-68, 2012.

[2] P. F. J. S. P. Abdaladhem Albreshne, "Web Services Technologies: State of the art," 2009.

[3] S.Satyanarayana, "CLOUD COMPUTING : SAAS," GESJ: Computer Science and Telecommunications, vol. 4, no. 36, pp. 75-79, 2012.

[4] S. M. Manish Godse, "An Approach for Selecting Software-as-a-Service (SaaS)," IEEE, no. 74, pp. 155-158, 2009.
[5] S. Bhardwaj, L. Jain and S. Jain, "CLOUD COMPUTING: A STUDY OF INFRASTRUCTURE AS A SERVICE (IAAS)," International Journal of Engineering and Information Technology, vol. 2, no. 1, pp. 60-63, 2010.
[6] S. Khurana and A. G. Verma, "Comparison of Cloud Computing Service Models: SaaS, PaaS, IaaS," InternatIonal Journal of electronIcs & communIcatIon technology, vol. 4, no. 3, pp. 29-32, 2013.
[7] R. G. Charan, T. Rao and P. Srinivas, "Deploying an Application on the Cloud," International Journal of Advanced Computer Science and Applications, vol. II, no. 5, pp. 119-125, 2011.
[8] C. Rao, M. Leelarani and R. Kumar, "Cloud: Computing Services And Deployment Models," International Journal Of Engineering And Computer Science, vol. 2, no. 13, pp. 3389-3392, 2013.
[9] R. Basiura, M. Batongbacal, B. Bohling, M. Clark, A. Eide, R. Eisenberg, K. Hoffman, B. Loesgen, C. Miller, M. Reynolds, B. Sempf and S. Sivakumar, Professional ASP.NET Web Services, Birmingham: Wrox Press Ltd, 2001.
[10] "International Conference on Internet Technologies & Society," Perth, 2010.
[11] F. Halili and M. Kasa, "Analysis and comparison of web service architectural styles, and business benefits of their use," International Conference in Information Systems and Technologies and their importance in, pp. 701-707, 2011.
[12] F. Halili , E. Rufati and I. Ninka, "Styles of Service Composition – Analysis and Comparison Methods," International conference on Computational Intelligence, Communication, no. 5, pp. 278-284.
[13] P. Patel, A. Ranabahu and A. Sheth, "Research Gate - Service Level Agreement in Cloud Computing," 19 September 2017. [Online]. Available: https://www.researchgate.net/publication/318017907_Service_Level_Agreement_in_Cloud_Computing. [Accessed 21 09 2017].
[14] K. Kadman, D. Rajwal, . D. Band, Z. Chougule and A. Yadav, "Study of SaaS and its Application in Cloud," IOSR Journal of Computer Engineering, vol. XIV, no. 5, pp. 65-69, 2013.
[15] A. Albreshne, P. Fuhrer and J. Pasq, "Web Services Orchestration and Composition," 2009.
[16] D. B. Claro, P. Albers and J. K. Ha, "WEB SERVICES COMPOSITION," Région du Pays de La Loire, 2006, p. Chapeter 8.
[17] ourkloud, "ourkloud," [Online]. Available: http://www.ourkloud.com.au/cloud/public/. [Accessed 16 03 2017].