

# AirBits: A Web Application Development Using Microsoft Azure

Sahil Taneja<sup>1</sup>, Mani Karthik<sup>2</sup>, Mohit Shukla<sup>3</sup>, Hitesh Kumar Sharma<sup>4</sup>  
<sup>1,2,3,4</sup>University of Petroleum & Energy Studies, Dehardun  
Uttarakhand, India  
<sup>1</sup>sahiltupes@gmail.com, <sup>4</sup>hkshitesh@gmail.com

## Abstract

Microsoft Azure is a public cloud computing platform. It provides various cloud computing services, which includes computing, data analytics, storage resources and network resources. End users or Client can select and choose from these computing services to build - innovate, maintain and scale new applications, or run/deploy the existing applications, in the cloud and Moreover, it provides a pay-per-use option which considered to be very cheap or reasonable for one to use cloud resources.

## KEYWORDS

JSP, Servlets, IaaS, Azure.

## 1. INTRODUCTION

Today many of the companies in the world are shifting to cloud. Many applications are deploying on Cloud. Because cloud is easy to maintain and can access everywhere. Cloud Computing is a technology which delivers computing resources to the clients on request. Cloud computing mainly delivers three types of services that are infrastructure, Platform, Software and Community. In IaaS (Infrastructure as a service) deliveries hardware requirements such as servers, Storage etc. to users on request and charge back them - based on their usage. Similarly In platform provides runtime compliers etc. and in SaaS (Software as a service) provides software where software is maintain by the provider. In our project we deploy the web application which will be deployed and maintain over Microsoft Azure. The Web Application will mainly developed using technologies JSP – Java Server Pages, Java Servlets, XML.

## 2. PROBLEM STATEMENT

In a nutshell, a Java Web-Based Application will be develop by using JSP, Servlets and XML along with it, this Java Web application will be deployed on cloud platform which is Microsoft Azure - Platform as a Service (PaaS). As running up the web apps - 24/7, it will be expensive for the customer as the person needs to handle the charges for the usage of infrastructure

resources such as CPUs, storage and networking to host and maintain the web apps and ensures that there is less downtime. So to overcome this, Client must migrate its web app towards Cloud technology which provides a pay-per-use model so it will cost him/her very less.

## 3. LITERATURE REVIEW

In early days web pages are static only web pages are written in html language only. In that days there is no data transmission between clients and server. Server just shows the static pages that client request. Slowly these static pages are changed to dynamic with languages like Servlets, JSP and XML. After introduction of these languages, many dynamic web pages and dynamic web applications have been developed. There are two major protocols used, GET and POST for dynamic behavior. Servlets are used to determine whether the request method is GET or POST. In simple terms servlets are used to developed dynamic web apps or web pages. First Internet came into existence. As internet came, access to the information has been standardized in the form of html pages accessed using a standard browser. About 90% of the information today is accessed using web-browsers. As the information was exponentially increasing day by day that needed large data sets to be processed in shortest possible time. Grid Computing Allowed computers from different administrative domains to run workloads in parallel. This form Distributed network of loosely coupled geographically dispersed computers. Working towards a common goal - Issues in Grid computing: Grid scaled well for all resources, but was not same for storage capacity. Grid had large anonymous computers (as the participating entities).Creating storage on anonymous, unregistered, unaudited computers includes the security risk. So the final Conclusion is that Grid is good for compute, but not for the storage part.

And now, finally Cloud Computing came into picture - For cloud computing internet development, majorly acted as catalyst. Laid foundation for a workable client-server model. As cloud computing came into existence, accessing data from anywhere is become easier. Many companies like Microsoft, Amazon and Google started

providing cloud computing services 3 deployment methods.

In a nutshell, a Java Web-Based Application will be developed by using JSP, Servlets and XML along with it, this Java Web application will be deployed on cloud platform which is Microsoft Azure - Platform as a Service (PaaS).

As running up the web apps - 24/7, it will be expensive for the customer as the person needs to handle the charges for the usage of infrastructure resources such as CPUs, storage and networking to host and maintain the web apps and ensures that there is less downtime. So to overcome this, Client must migrate its web app towards Cloud technology which provides a pay-per-use model so it will cost him/her very less.

#### 4. OBJECTIVES

Our objective is to build the Web Application using JSP, Servlets and XML Technologies, then to deploy the Web Application on Microsoft Azure – Public Cloud.

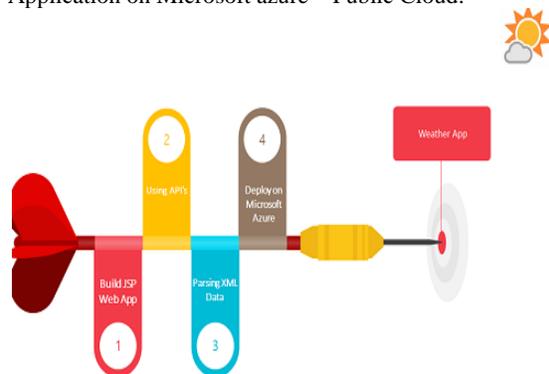


Fig 1: Objective Sequence

In above diagram we have shown our objectives in a sequential manner. It will start from building a web application and ending the processing after deploying the same on cloud.

#### 5. METHODOLOGY

A Java Web-Based Application will be developed by using JSP, Servlets and XML along with this, the Java Web application will be deployed on cloud platform which is Microsoft Azure - Platform as a Service (PaaS). Where, Client must migrate its web app towards Cloud technology which provides a pay-per-use model so it will cost him/her very less. The following steps shows the how we will deploy the created JSP application to Microsoft Azure through FTP (File Transfer Protocol):

### 1. Build the Web Application using JSP, Servlets and XML Technologies

#### 1.1 Setting up a Web Application Project

Web applications are distributed applications in a very efficient way, in other words they are computer programs or instructions that run or execute on more than one system and interact through a network zone or dedicated server. Java J2EE (Java 2 Platform, Enterprise Edition) provides support for web application through Servlets and JSPs

#### 1.2 Creating a Java Package and a Java Source File

First, we'll start off with creating a Java project. We can use the New Java Package wizard to create a Java package and provide the justified project name along with selection of Java Runtime Environment (JRE) and an appropriate Project Layout which determines whether there would be a separate folder for the source codes.

#### 1.3 Creating a Java Server Pages (JSP) File

Java J2EE (Java 2 Platform, Enterprise Edition) provides support for web application through Servlets and JSPs. Java J2EE - JSP technology enables rapid development of web-based applications that are server - and platform - independent. Java Server Pages technology provides the ability to add code snippets of servlet directly into a text-based document.

### 2. Deploying the Web Application Project on Microsoft Azure

After the completion of business logic of website, the jsp website will be deployed on Cloud platform, which is Microsoft Azure - Platform as a Service (PaaS). Microsoft Azure is a public cloud computing platform. It provides various cloud computing Services, which includes computing, data analytics, storage resources and network Resources. The web application will be deployed using FTP (File Transfer Protocol). Microsoft Azure provides a means to run/deploy Java web apps in the cloud using the Azure Portal's configuration UI, and the Azure Marketplace. It was created to allow user to easily on board any Java application on Azure.

The experiments were conducted on following configuration:

**Hardware** - RAM : 2 GB, HDD : 100GB, Processor: Intel Core i3/i5/i7

**Software** - OS : Windows XP/7/8/8.1/10 | Ubuntu, Local Server : Apache Tomcat

**Compiler** - Eclipse Neon 2.0 | Eclipse Mars - Java J2EE

**Web App** – Microsoft Azure Platform

### 5.1 Flowchart of Work Proposed

The figure 2 represents the flow chart of the proposed system.

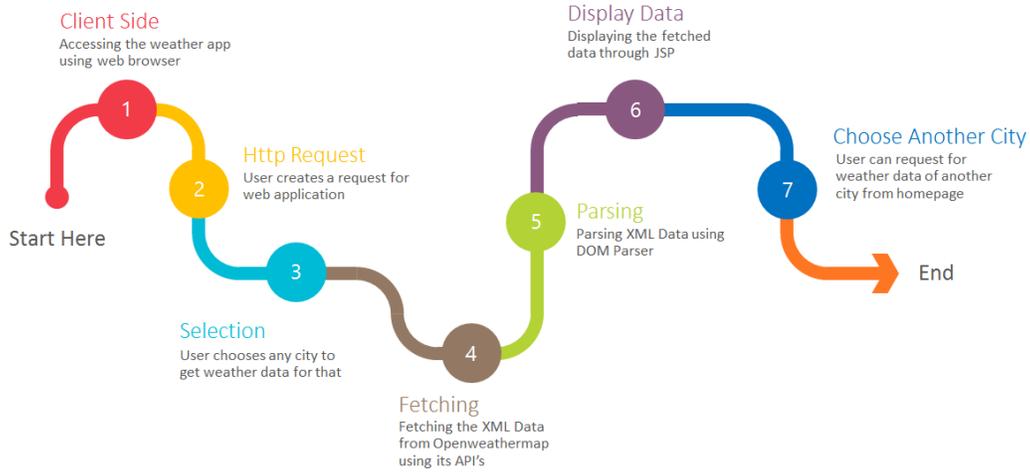


Fig 2: Flow Chart of Proposed System

### 5.2 Implementation

The figures 3 and 4 show the implementation process.

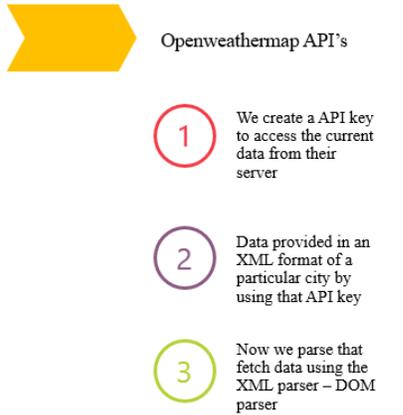


Fig 3: Openweathermap steps

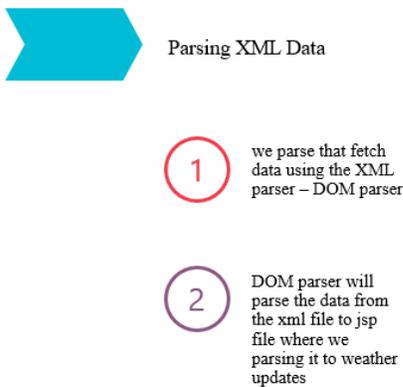


Fig 4: Parsing of XML data

### 6. CONCLUSION

Our weather app will revolutionize the way web applications are designed. In our project we have deploy the web application over Microsoft Azure. The java based weather application is developed using technologies like JSP – Java Server Pages, XML, CSS , Here we provide a user friendly interface which will cater user needs, specifically detailing the ongoing temperature for the desired location. SaaS is defined by delivery via Internet in addition with periodic payments. Hence we aim to design a self contained weather web application which gives us weather information obtained from Openweathermap service provider using its respective APIs

### 7. FUTURE WORK

Weather forecast: Weather forecasting is the application of science and technology to predict the state of the atmosphere for a given location. Openweathermap is a service provider who provide the API's to access the weather data of a particular city, we will use the suitable APIs for weather forecasting Login system connectivity through JDBC: we will use JSP for creating dynamic web application and through the JDBC drivers we connect to the database, and with this we will create sign up page along with database connectivity in java where we have created a new user sign up page and stored its record into database.

### 8. REFERENCES

[1] P. S. Yu, M. S. Chen, H. U. Heiss, S. H. Lee, 1992. On Workload Characterization of Relational

- Database Environments, IEEE Transactions on Software Engineering, Volume 18, Issue 4, pp. 347-355.
- [2] K. P. Brown, M. Metha, M. J. Carey, M. Livny, 1994. Towards Automated Performance Tuning for Complex Workloads, Proceedings of 20<sup>th</sup> VLDB Conference, Santiago, pp. 72-84.
- [3] <http://davidmlane.com/hyperstat/index.html>
- [4] S. Elnaffar, W. Powley, D. Benoit, P. Martin, 2003. Today's DBMSs: How Autonomic are They?, Proceedings of the 14<sup>th</sup> DEXA Workshop, Prague, pp. 651-654.
- [5] D. Menasec, Barbara, R. Dodge, 2001. Preserving QoS of E-Commerce Sites through Self-Tuning: A Performance Model Approach, Proceedings of 3<sup>rd</sup> ACM-EC Conference, Florida, pp. 224-234.
- [6] D. G. Benoit, 2000. Automated Diagnosis and Control of DBMS resources, EDBT Ph.D Workshop, Konstanz.
- [7] [www.dtc.umn.edu/disc/resources/debnath1.pdf](http://www.dtc.umn.edu/disc/resources/debnath1.pdf)
- [8] K. P. Brown, M. J. Carey, M. Livny, 1996. Goal-Oriented Buffer Management Revisited, Proceedings of ACM SIGMOD Conference, Montreal, 1996, pp. 353-364.
- [9] P. Martin, H. Y. Li, M. Zheng, K. Romanufa, W. Powley, 2002. Dynamic Reconfiguration Algorithm: Dynamically Tuning Multiple Buffer Pools", Proceedings of 11<sup>th</sup> DEXA conference, London, pp.92-101.
- [10] P. Martin, W. Powley, H. Y. Li, K. Romanufa, 2002. Managing Database Server Performance to Meet Qos Requirements in Electronic Commerce System", International Journal of Digital Libraries, Volume 8, Issue 1, pp. 316-324.
- [11] S. Duan, V. Thummala, S. Babu, 2009. Tuning Database Configuration Parameters with iTuned, VLDB '09, Lyon, France.
- [12] H. K. Sharma, A. Shastri, R. Biswas, 2012 Architecture of Automated Database Tuning Using SGA Parameters, Database Systems Journal, Volume 3, Issue 1, pp. 7-10.
- [13] A. G. Ganek, T. A. Corbi, 2003. The Dawning of the Autonomic Computing Era, IBM Systems Journal, Volume 42, Issue 1, pp. 5-18.
- [14] H. K. Sharma, A. Shastri, R. Biswas, 2012. A Framework for Automated Database Tuning Using Dynamic SGA Parameters and Basic Operating System Utilities, Database Systems Journal, Volume 3, Issue 4, pp. 25-32.