ISSN: 2278 – 1323
International Journal of Advanced Research in Computer Engineering & Technology
Volume 1, Issue 5, July 2012

ONLINE BUSINESS MANAGEMENT USING HYBRID CLOUD

Vanaja Gokul, Venkatesh.G, Vignesh.B and GowriShankar.M
Department of Computer Science & Engineering,
Sri Venkateswara College of Engineering,
Sriperumbudur, Chennai, India
vanajagokul@gmail.com

***Abstract*— Hybrid cloud combines the external public services and the internal private cloud providing services thereby enhancing the capabilities to secure, manage and govern the entire cloud spectrum. In this paper, online business management system is implemented using a hybrid cloud architecture which provides faster and greater ease of access to business transaction and management. The implemented system effectively favors the small scale, large scale, Business to Business (*B2B*) and Business to Consumers (*B2C*) by using E-commerce for management solution.**

***Index Terms*— Hybrid cloud, business management, E-commerce**

1. INTRODUCTION

1.1 Distributed Computing

Distributed computing is a field of computer science where multiple autonomous computers communicate through a computer network. The computers interact with each other in order to achieve a common goal.

Types of Distributed Computing

- **Autonomic computing**: Computer systems capable of self-management.
- **Client–server model**: Client–server computing refers to any distributed application that distinguishes service providers and service requesters.
- **Grid computing**: A form of distributed and parallel computing that forms the basic element for the large scale supply of cloud services [2].
- **Mainframe computer**: Powerful computers used mainly by large organizations for critical applications.
- **Utility computing**: The packaging of computing resources [3], such as computation and storage, as a metered service similar to a traditional public utility, such as electricity.
- **Peer-to-peer**: Distributed architecture without the need for central coordination, with both the suppliers and consumers of resources being available at the same time.
- **Cloud computing:** Cloud computing [9] is a general term for anything that involves delivering hosted services over the Internet.

1.2 Cloud Computing

Cloud is an emerging technology [8, 14] that uses the internet and central remote servers to maintain data and applications. Cloud computing allows consumers and businesses to use applications without installation and access their personal files at any computer with internet access. This technology allows for much more efficient computing by centralizing storage, memory, processing and bandwidth. It can also be defined as the delivery of computing as a service rather than a product, whereby shared resources, software, and information are provided to computers and other devices as a utility over a network.

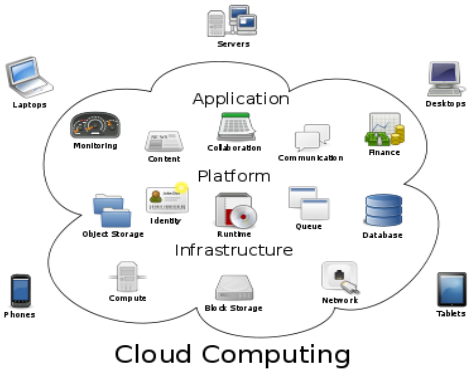

Figure 1.1 Sample Cloud Architecture

1.2.1 Service Models

As referred from [14], the three service models supported in cloud are,

- Infrastructure-as-a-Service (*IaaS*)

Manuscript received June, 2012.
***Vanaja Gokul**, Asst. Prof., Department of Computer Science & Engg., Sri Venkateswara College of Engineering, Sriperumbudur, India (email:vanajagokul@gmail.com).*
***Venkatesh**,**Vignesh and GowriShankar,** Students, Department of Computer Science & Engg., Sri Venkateswara College of Engineering, Sriperumbudur, India.*

- Platform-as-a-Service (*PaaS*)
- Software-as-a-Service (*SaaS*)

1.2.1.1 Infrastructure-as-a-Service (*IaaS*)

Cloud providers offer computers as physical or more often as virtual machines, raw (block) storage, firewalls, load balancers, and networks. *IaaS* providers supply these resources on demand from their large pools installed in data centres. Local area networks including IP addresses are part of the offer. Some examples of *IaaS* are

- Amazon S3/EC2,
- Microsoft Windows Azure,
- VMWare vCloud.

1.2.1.2 Platform-as-a-Service (*PaaS*)

In the *PaaS* model, cloud providers deliver a computing platform or solution stack typically including operating system, programming language execution environment, database, and web server. Application developers can develop and run their software solutions on a cloud platform without the cost and complexity of buying and managing the underlying hardware and software layers. With some *PaaS* offers, the underlying compute and storage resources scale automatically to match application demand such that the cloud user does not have to allocate resources manually. Some examples of *PaaS* are

- Google App Engine.
- Microsoft Azure Services Platform.
- ORACLE/AWS.

1.2.1.3 Software-as-a-Service (*SaaS*)

In the *SaaS* cloud model, the vendor supplies the hardware infrastructure, the software product and interacts with the user through a front-end portal. *SaaS* is a very broad market. Services can be anything from Web-based email to inventory control and database processing. Because the service provider hosts both the application and the data, the end user is free to use the service from any application.

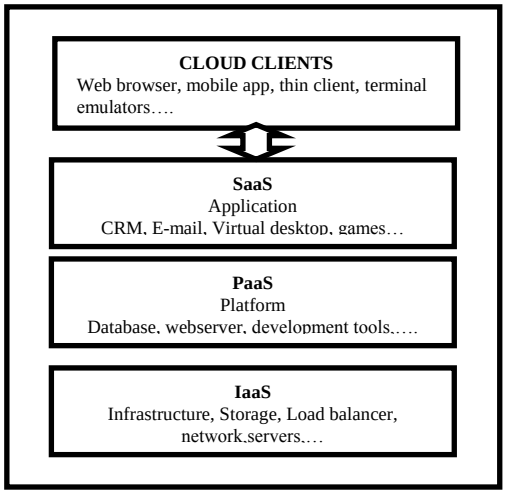

Figure 1.2 Service Models

1.2.2 Deployment Modes

As referred from [1, 14] the following are the deployment modes in cloud.

1.2.2.1 Private

Private cloud (also called internal cloud or corporate cloud) is a computing architecture **[11]** that provides hosted services to a limited number of people behind a security validation. Hence, the organizations themselves have to *buy, build, and manage them* and thereby is not accepted by many as an actual cloud computing and subsequently do not even reap benefits from major advantages of cloud computing.

1.2.2.2 Public

A public cloud is actually the known version of Cloud and is based on the original cloud computing model. Here the service provider manages the resources including the applications and data storage and makes it available to the general public over the Internet. Generally the Public cloud service providers operate on a *pay-per-usage* model but some also provide free services till a certain limit to attract more users.

1.2.2.3 Hybrid

Hybrid cloud as the name suggests is a composition of two or more clouds generally private and public. The individual cloud models remain independent entities but function together, thereby, offering the benefits of multiple deployment models. It sometimes is used in the reference of a platform where multiple cloud systems are connected and which offers the flexibility to move programs and data easily from one deployment system to another. Organizations may host critical applications on private clouds and applications with relatively less security concerns on the public cloud.

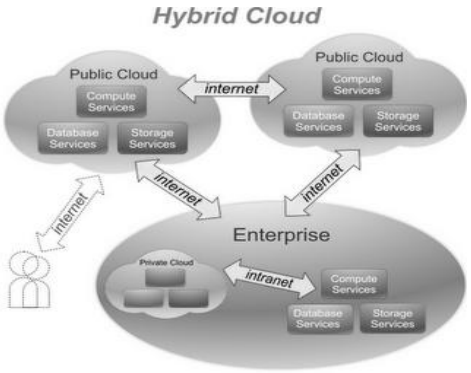

Figure 1.3 Hybrid Cloud Architecture

1.3 E-Commerce

E-commerce (**e-commerce**) or electronic **commerce**, subset of e-business, is the purchasing, selling, and exchanging of goods and services over computer networks (such as the Internet) through which transactions or terms of sale are performed electronically.

2. RELATED WORK AND MOTIVATION

Many of the previous work [3, 9] in the field of cloud computing have been in the areas of enhancing distributed computing, general explanation of the cloud technology, differences among similar technologies, security requirements and the future expectations in these emerging environments. The new methods can be experimented and used by the advent of global industries acting as cloud providers. This motivates the proposed implementation of hybrid cloud architecture.

2.1 Cloud-Scale Intelligent Infrastructure

Cloud computing is an emerging application platform that aims to share data, calculations and services among users. The methods to model it with the challenges like user interface, task distribution and coordination issues are explained and evaluated in [4].

2.2 Cloud-Based Infrastructure

As an example of cloud, a set of distributed parallel computers is considered to be working independently or dependently, but additively to serve the cumulative needs of a large number of customers requiring service. Quantitative methods of statistical inference on the quality of service (*QoS*) or conversely, Loss of Service (*LoS*), as commonly used customer satisfaction metrics of system reliability and security performance are reviewed in [7].

2.3 On Demand Cloud Services

The emergence of the cloud computing paradigm promises flexibility and adaptability through on-demand provisioning of compute resources. Different providers expose different interfaces to their compute resources utilizing varied architectures and implementation technologies [5].

2.4 Cloud Storage

Cloud storage provides a large paradigm shift from the past, where now the customers have to pay only for the utilization of the storage and transfer of data. Large enterprises use *CDNS* (Content Delivery Network System) and content-centering [6] for the transfer of data and storage of data. Some major advancement in cloud storage and *CDNS* are given in [10].

2.5 Hybrid Cloud Architectures

Hybrid architectures that combine traditionally provided *IT* services with web-based services [1, 12] will enable to harness the advantages of the computing paradigm, while minimizing the risks.

3. PROPOSED WORK

The proposed work of implementing an online business management tool using hybrid cloud architecture can be deployed and can be used in an on-demand basis.

3.1 Modules used

1. Setting up hybrid cloud infrastructures
2. Configuring (Application Programming Interface) *API*'s using *PHP*
3. Designing Application patterns and User Interface
4. Back end connectivity Modules
5. Deploying App to Local Cloud environment
6. Acquiring *DNS* from Cloud portal and Test run

Business application tool is built using *PHP* (whole application runs on a web browser) which provides a good back end connectivity and runtime evaluation support. Back end modules are coded using *SQL Azure* an alternative to *SQL* for developing cloud applications. The basic *API* workflow consists of the design of the web application built in *PHP*. Cloud app is a platform free website with cloud database as backend and it highlights the difference and various uses of private, public and Hybrid cloud infrastructures. End users access cloud based applications through a web browser or a light weight desktop or mobile application.

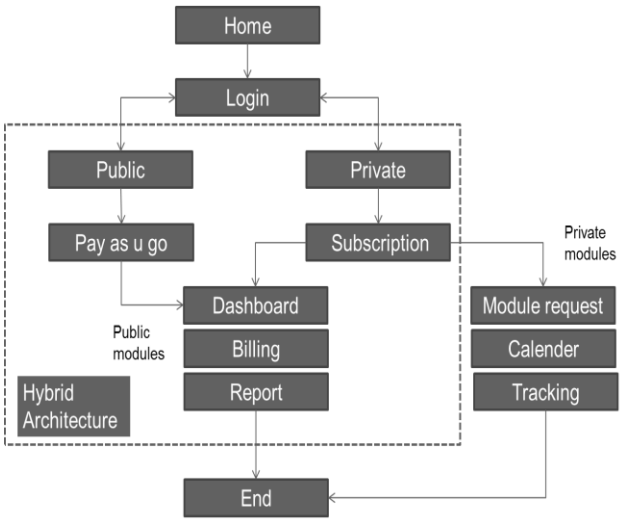

Figure 3.1: *API* Architecture

3.2 Application Provider (*SaaS*)

Online business application is built on cloud infrastructure and the system acts as an application provider or it acts a middleware. End users access the application through a PHP compatible browser, or it can also be incorporated as a standalone desktop application.

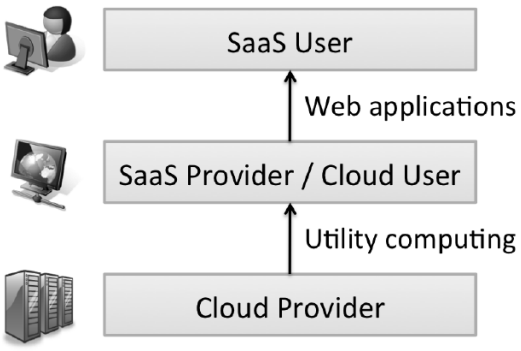

Figure 3.2 *SaaS* Provider

3.3 *SQL-AZURE* Overview

SQL Azure [13] is a highly scalable cloud database service built in *SQL* server technologies. *SQL Azure* database has built-in high-availability, failover, and redundancy. *SQL Azure* is a cloud version of *SQL* Server database engine. *SQL Azure* is an ideal database for transactional query-based business applications such as transaction processing, departmental or line of business solutions. *SQL Azure* allows creation of virtual *SQL* server on cloud servers. Virtual server consists of databases, user accounts etc. *SQL azure* is a high availability Server and it provides automatic, seamless fail-over.

SQL Azure can be accessed from Windows Azure applications, as well as on-premises applications hosted in customer's data centers. It provides high availability by storing multiple copies of databases, elastic scale and rapid provisioning. It exposes a subset of the full *SQL* Server functionality, including only a subset of the data types including string, numeric, date and boolean. It uses an *XML*-based format for data transfer. Like Microsoft *SQL* Server, *SQL Azure* uses *T-SQL* as the query language and Tabular Data Stream (*TDS*) as the protocol to access the service over internet.

3.3.1 Setting up Windows Azure Cloud Development Environment

Microsoft Windows Azure [13] provides *PaaS* to build and deploy the application into the cloud storage media. A unique App Key and client name is provided for every subscription. Windows Azure enables us to build and run highly available applications without focusing on the infrastructure. It provides automatic OS and service patching, built in network load balancing and resiliency to hardware failure. It supports a deployment model that enables to upgrade the application without downtime. In addition to service deployment and management tasks, the Windows Azure Platform Management Portal provides the user interface for provisioning *SQL Azure* servers and logins, configuring the firewall, and creating *SQL Azure* databases.

After provisioning a server, a tool such as *SQL* Server Management Studio or the Management Portal for *SQL Azure* must be used for database management tasks such as designing and editing tables, views, stored procedures, authoring and executing *Transact-SQL* queries.

3.4 Application Pattern Design

Microsoft Azure provides a wide range of options to design *API*'s using Cloud storage as the back end storage. The *SaaS* Application is designed using *PHP* for better compatibility and greater flexibility in creating dynamic web pages. The following describes the interface modules.

3.4.1 Registration

Registration is a three step process which contains collection of details and credit card validation. Registration page gives the introduction and modes of subscription for the users. A user can make use of both public and private modules by subscribing to the appropriate resources of his needs.

- PAY-AS-U-GO

Public user has the independence to select the amount of database or time of usage.

- SUBSCRIPTION

Private user can subscribe to the service in half yearly or annual subscription models.

3.4.2 Dashboard

The domain serves as the homepage of the application and it is a public domain in the hybrid architecture. The domain is responsible for posting news about pending payments, delay in the shipment of the item and keeps track of time usage and memory bytes used. The domain shows the order of the status of a particular item with its item code and the name of the user who ordered the item. It shows the invoice summary of the user giving descriptions of the invoice and the changes occurring in the invoice every month.

3.4.3 Customer

This domain keeps log of all the customer names in an alphabetical order. Details about the customers include email id, phone number, addresses. The domain is also responsible for getting the mode of payments for items purchased by the customer.

3.4.4 Billing

Invoice is a record of business transaction. There is typically an invoice for every order that a customer places. This invoice records all the information regarding the items purchased along with the customer details and also includes the cost incurred for using the service. The billing domain is used to generate reports which can be used to calculate the profit gained by the business.

3.4.5 Organizer

The organizer is a private module domain which is used to specify the different tasks and all the upcoming events to a particular customer.

3.4.6 Module Request and Feedback

Module request is an on-demand private module, which includes customer feedback and interaction.

3.4.7 Reports

A report is a private module that is used to generate dynamic graphs and tables by analyzing the invoice and customer statistics. Cloud usage and subscription details can also be viewed in the reports module.

3.5 Back End Connectivity Modules

The whole back end infrastructure is coded in windows azure cloud development environment. The *SQL Azure* query language is utilized for storage and performing data mining.

There are three major types of storage medium
- Tables
- Blobs
- Queues

3.5.1 Tables

The tables contain entities, and the entities contain properties. Entities act as classes and rows act as objects providing an object oriented approach to the data variables. The tables are scalable to billions of entities and terabytes of data, and may be partitioned across thousands of servers. The tables in Windows Azure Storage enforce no schema and the properties in a single entity can be of different types that can change over time.

Table Data Model

The following summarizes the data model for Windows Azure Table,

- Storage Account – An application must use a valid account to access Windows Azure Storage. A new account can be created via the Windows Azure portal web interface. The user will receive a 256-bit secret key once the account is created. This secret key is then used to authenticate user requests to the storage system.
- Table – contains a set of entities. An application may create many tables within a storage account.
- Entity (Row) – Entities (an entity is analogous to a "row") are the basic data items stored in a table. An entity contains a set of properties. Each table has two properties, namely the "PartitionKey and RowKey" that form the unique key for the entity.
- Property (Column) – This represents a single value in an entity. Property names are case sensitive. A rich type set is supported for property values.
- PartitionKey – The first key property of every table. The system uses this key to automatically distribute the table's entities over many storage nodes.
- RowKey – A second key property for the table. This is the unique ID of the entity within the partition it belongs to. The PartitionKey combined with the RowKey uniquely identifies an entity in a table.

Table 3.1 SQL table

Data Fields	NUMBER	NAME	PHONE
	001	Axvdjh	9822104534
	002	Bsdkdf	9884144559

Normal SQL tables are scaled vertically downwards where the first row of the table represents the data fields as shown in Table 3.1.

Table 3.2 SQL Azure Table

Property \ Entity		
Number	001	002
Name	Axvdjh	bsdkdf
Phone	9884144559	9822104534

SQL Azure tables are scaled horizontally where each row represents entity and column represents property.

3.5.2 Blob Storage

Binary large objects (*Blobs*) are just slabs of bytes. Windows Azure *Blob* enables applications to store large sets of these slabs of bytes, up to 50GB each in the cloud. Blob storage can be used to upload multimedia and document files; it acts as an online hard disk for the application. It supports a massively scalable blob system, where hot blobs will be served from many servers to scale out and meet the traffic needs of the application. Furthermore, the system is highly available and durable. One can always access the data from anywhere at any time, and the data is replicated at least 3 times for durability. In addition, strong consistency is provided to ensure that the object is immediately accessible once it is added or updated, a subsequent read will immediately see the changes made from a previously committed write.

Blob Data Model

1. Storage Account – All access to Windows Azure Storage is done through a storage account.

- This is the highest level of the namespace for accessing blobs

- An account can have many Blob Containers

2. Blob Container – A container provides a grouping of a set of blobs. The container name is scoped by the account.

- Sharing policies are set at the container level. Currently "Public READ" and "Private" are supported. When a container is "Public READ", all its contents can be read by anyone without requiring authentication.

- When a container is "Private", only the owner of the corresponding account can access the blobs in that container with authenticated access.

- Containers can also have metadata associated with them. Metadata is in the form of pairs, and they are up to 8KB in size per container.

- The ability to list all of the blobs within the container is also provided.

3.5.3 Queues

Queue allows decoupling of different parts of a cloud application, enabling cloud applications to be easily built with different technologies and easily scale with traffic needs.

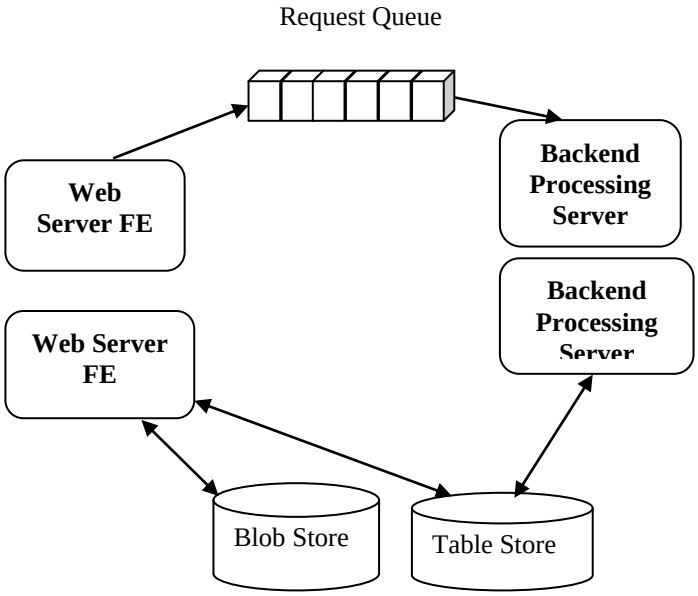

Figure 3.4 Queue usage in SQL Azure

3.6 Deployment and Test Run

In a traditional server hosting model, the deployment of an application to a host, patching the operating system, components and application code has to be taken care manually. Windows Azure handles deployment and update issues automatically. There are four major scenarios for deploying cloud application.

3.6.1 New Deployment

To deploy an application to Windows Azure, a new hosted service must be created via the Windows Azure Management Portal. A hosted service allows the application to select one of the six Windows Azure data centers around the world where the applications will be deployed. A *DNS* record is created which maps the custom domain name to the Windows Azure *DNS* name.

3.6.2 Configuration Change

Once an application is deployed and running, the roles can be reconfigured by modifying the CSCFG file in use.

3.6.3 Incremental Code Upgrade

If the application wants an incremental code upgrade to a role a new CSPKG file containing the new code is created and uploaded to Windows Azure via the Windows Azure Management Portal.

3.6.4 Major Upgrade

Windows Azure also offers a way to make a major new release of the application without incurring any downtime.

4. EXPERIMENT AND ANALYSIS

Registration

The first step of registration contains a form to fill up the basic user details as shown below,

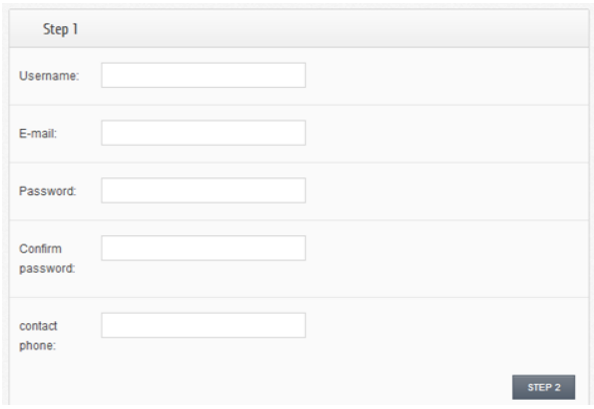

Figure 4.1 Registration (Step 1)

The second step of registration contains a form to get input about the basic details of organization as shown below,

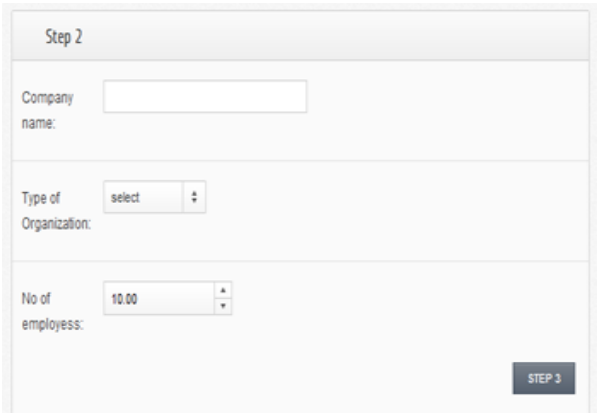

Figure 4.2 Registration (Step 2)

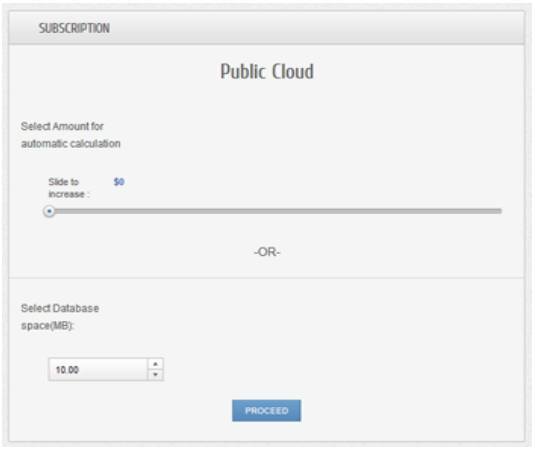

Figure 4.5 Subscription (Public)

Subscription

In the first step of subscription, user has to choose between public or private service.

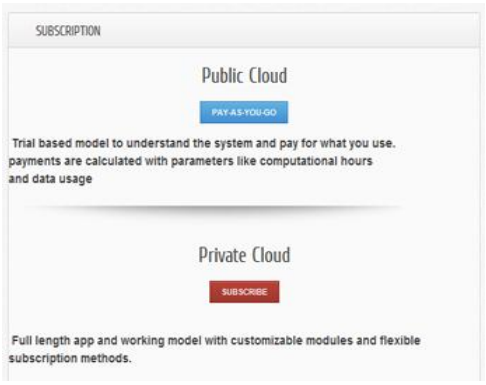

Figure 4.3 Subscription

The following snapshot shows the different subscription options for private service subscribers.

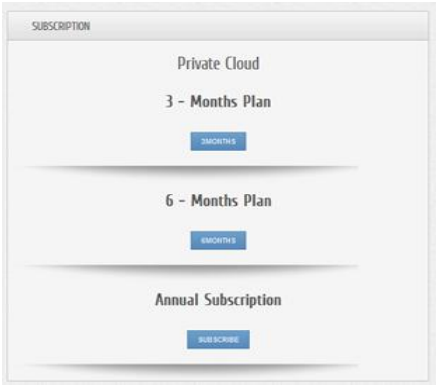

Figure 4.4 Subscription (Private)

The following snapshot shows the different subscription options for public service subscribers. Public user can either pay for the database size or a fixed amount can be specified.

Dashboard (Private)

Dashboard is the home page of the application. Private users have all the modules specified.

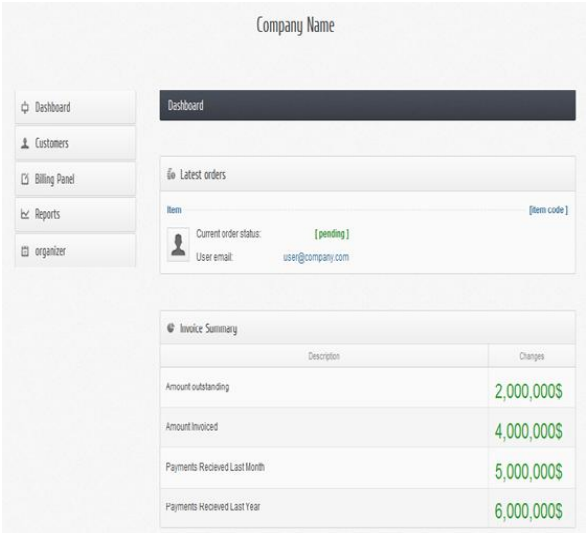

Figure 4.6 Dashboard (Private)

Dashboard (Public)

The following interface shows that the public users can access only three important modules in the dashboard.

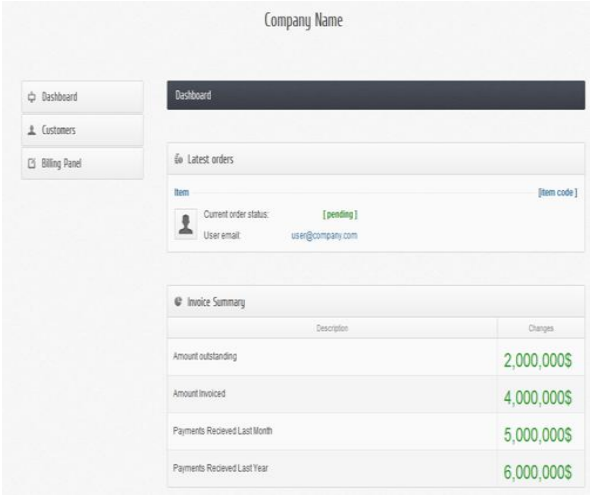

Figure 4.7 Dashboard (Public)

Billing Module

Billing panel is a hybrid module where new invoices and bills are added to the database.

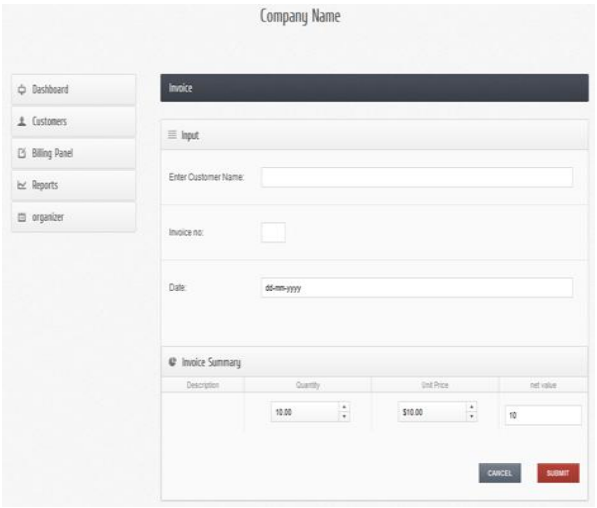

Figure 4.8 Billing

Reports Module

The screenshot below shows a report that is generated from the bills in the form of charts and graphs.

Figure 4.\9 Report

Organizer

Organizer is a private module that is used to schedule events and set reminders

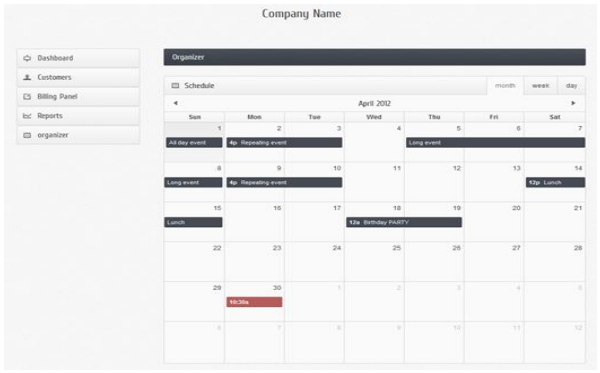

Figure 4.10 Organizer

Worker Role

The Worker Role is similar to a windows service. It starts up and will be running all the time. It is used to perform some maintenance work.

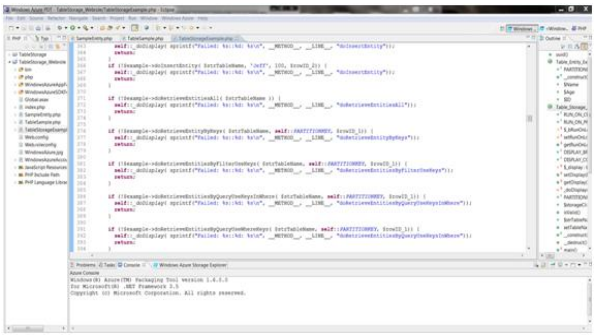

Figure 4.11 Worker Role

Packaging application

The following interface shows the packaging of the cloud application for deployment in local development environment.

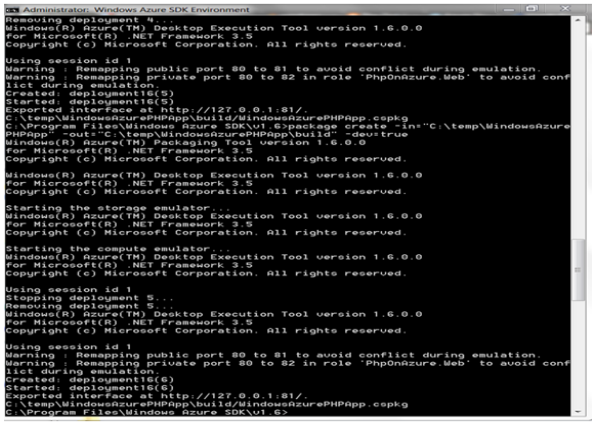

Figure 4.12 Packaging

APP Deployment

The screenshot below shows the sample Azure app deployment and worker roles maintaining the backbone of the application.

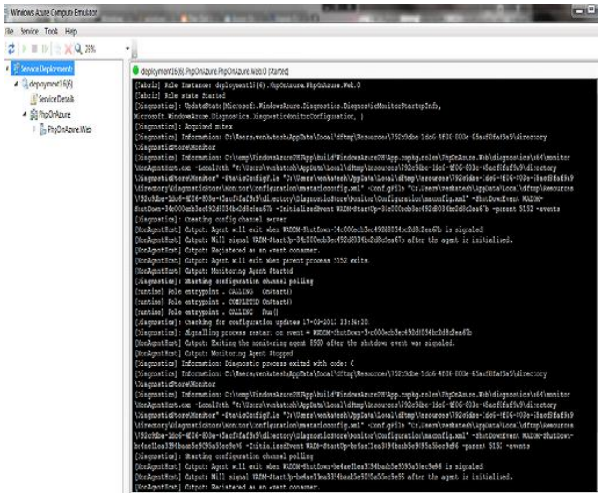

Figure 4.13 APP Deployment

5. CONCLUSION

In this proposed work the advantages of cloud computing is illustrated as a developing technology and its impact on E-commerce and the IT industry. The proposed work mainly concentrates on the implementation of the Hybrid cloud architecture.

6. REFERENCES

1. Georg Lackermair. "Hybrid cloud architectures for the online commerce". In Proceedings of Procedia Computer Science, 2011, Vol: 3, pp 550–555.

2. Aymerich, F. M., Fenu, G., Surcis, S. "An Approach to a Cloud Computing Network". In Proceedings of 1st International Conference on the Applications of Digital Information and Web Technologies, Ostrava, CZECH REPUBLIC, IEEE 2008, pp 120-125.

3. Banerjee, P. "An intelligent IT infrastructure for the future". In Proceedings of 15th International Symposium on High-Performance Computer Architecture, 2009.

4. Delic, K. A., and Riley, J. A.. "Enterprise Knowledge Clouds: Next Generation KM Systems". In Proceedings of International Conference on Information, Process, and Knowledge Management, Cancun, MEXICO. 2009, pp 49-53.

5. Dodda, R. T., Smith, C., and van Moorsel, A. "An Architecture for Cross-Cloud System Management". In Proceedings of 2nd International Conference on Contemporary Computing, Noida, INDIA, 2009, pp 556-567.

6. Erickson, J. S., Spence, S., Rhodes, M., Banks, D., Rutherford, J., Simpson, E., et al. (2009), "Content-Centered Collaboration Spaces in the Cloud". IEEE Internet Computing, volume 13 Issue 5, pp 34-42.

7. Grossman, R. L., Gu, Y. H., Sabala, M., and Zhang, W. Z. (2009), "Compute and storage clouds using wide area high performance networks". Future Generation Computer Systems: The International Journal of Grid Computing Theory Methods and Applications, volume 25 Issue 2, pp 179-183.

8. Lijun, M., Chan, W.K., and Tse, T.H. (2008), "A tale of clouds: Paradigm comparisons and some thoughts on research issues". In Proceedings of IEEE Asia-Pacific Services Computing Conference, APSCC'08, pp 464-469.

9. Praveena, K., and Betsy T. (2009), "Application of Cloud Computing in Academia". IUP Journal of Systems Management', volume 7 Issue 3, pp 50-54.

10. Open Grid Forum and Storage Networking Industry Association (2009), "Cloud Storage for Cloud Computing".

11. Rohit Maheshwari and Sunil Pathak, **"**A Proposed Secure Framework for Safe Data Transmission in Private Cloud"

12. www.gartner.com

13. www.windowsazure.com

14. http://en.wikipedia.org/wiki/Cloud_computing