Guidelines for Student's Laboratory Journal

The laboratory assignments are to be submitted by student in the form of journal. Journal consists of Certificate, table of contents, and handwritten write-up of each assignment (Title, Date of Completion, Objectives, Problem Statement, Software and Hardware requirements, Assessment grade/marks and assessor's sign, Theory- Concept in brief, algorithm, flowchart, test cases, Test Data Set(if applicable), mathematical model (if applicable), conclusion/analysis

Assignment

Title –

Objective –

Problem Statement –

Software and Hardware requirements –

Theory –

1. Introduction
2. diagram
3. Algo
4. Flow chart

Conclusion –

Assignment - 1

Title – Microsoft Azure

Objective – To learn about Microsoft Azure, a cloud computing platform and infrastructure, created by Microsoft

Problem Statement – Case study on Microsoft azure to learn about Microsoft Azure is a cloud computing platform and infrastructure, created by Microsoft, for building, deploying, and managing applications and services through a global network of Microsoft-managed data centers.

Software and Hardware requirements –

A computing device with a display having internet connectivity and capable to run web browser

Operating system – any

Web browser – any

Theory –

Introduction

- Microsoft Azure is a cloud computing service created by Microsoft for building, testing, deploying and managing applications and services through a global network of microsoft managed data centers. It provides Software as a Service (SaaS) platform as a Service (PaaS) and information as a Service (IaaS) and support many different programming languages, tools and frameworks, including both Microsoft specific and third-party software and systems.
- Azure was announced in October 2008, started with codename "Project Red Dog", and released in February 1, 2010 as "Windows Azure" before being renamed "Microsoft Azure" on March 25, 2014.

- Microsoft Azure uses a specialized operating system, called Microsoft Azure, to run its "febric layer": a cluster hosted at microsoft's data centers that manages computing and storage resources of computers and provisions the resources (or a subset of them) to applications running on top of microsoft Azure.
- Microsoft Azure has been described as a "cloud layer" on top of a number of Windows Server systems, which use Windows Server 2008 and a customized version of Hyper-V known as the microsoft Azure Hypervisor to provide virtualization of services.
- Scaling and reliability are controlled by the microsoft Azure Fabic controller, which ensures the services and environment do not fail if one or more of the servers fails within the microsoft data center, and which also provides the management of the user's web application such as memory allocation and load balancing.
- Azure provides an API built on REST, HTTP, and XML that allows a developer to interact with the services provided by microsoft Azure. Microsoft also provides a client-side managed class library that encapsulates the function of interacting with the services with microsoft Visual Studio, Git and Eclipse.

Diagram

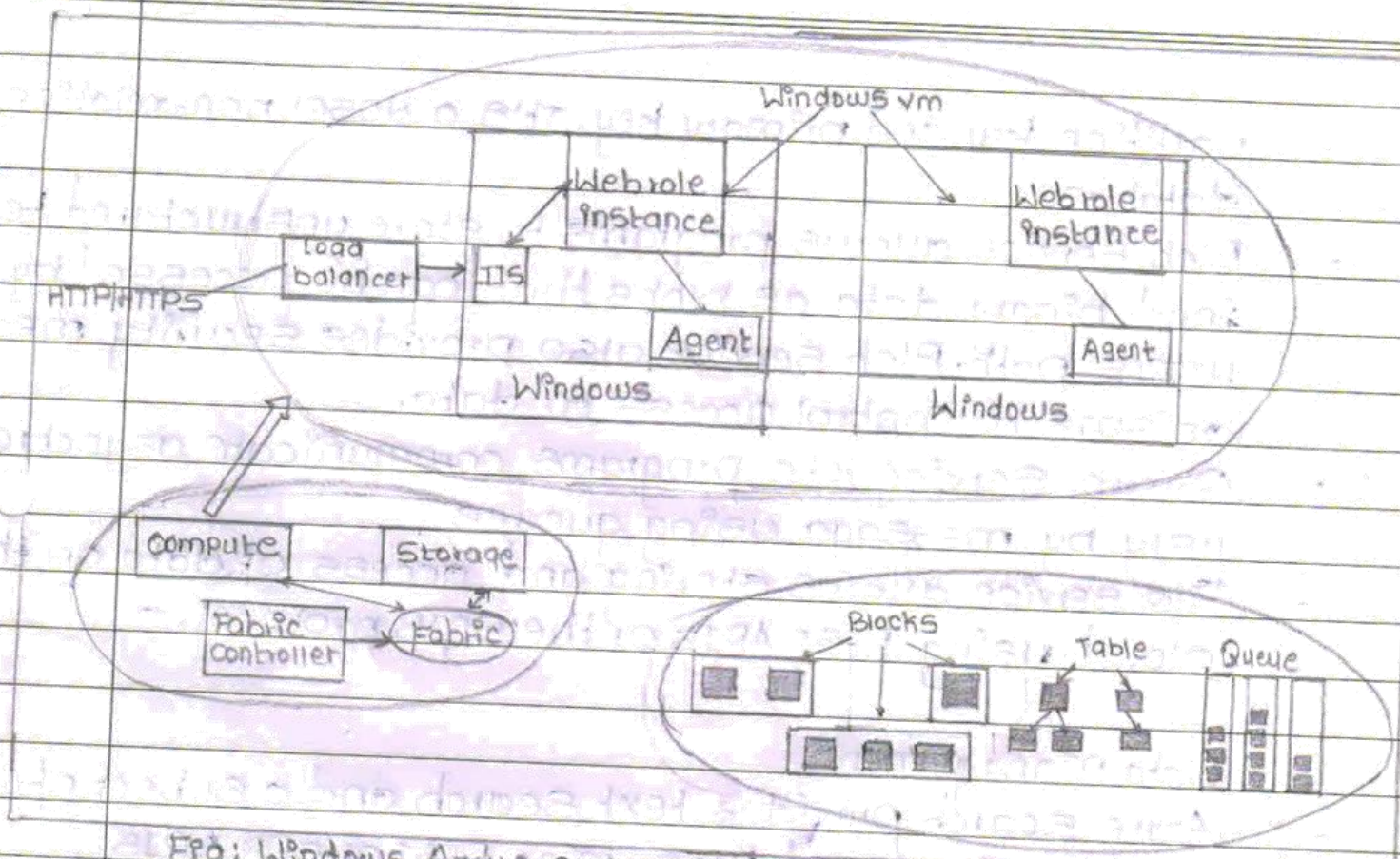


Fig: Windows Azure component Architecture.

Services:

Mobile Services:

- Mobile engagement collects real-time analytics that highlight users' behavior. It also provides push notifications to mobile devices.
- HockeyApp can be used to develop, distribute and beta-test mobile apps.

Storage Services:

- Storage services provides REST and SDK APIs for storing and accessing data on the cloud.
- Table Service lets programs store structured text in partitioned collections of entities that are accessed by

partition key and primary key. It's a NOSQL non-relational database.

- Blob Service allows programs to store unstructured text and binary data as blobs that can be accessed by a HTTP(S) path. Blob service also provides security mechanisms to control access to data.
- Queue Service lets programs communicate asynchronously by message using queues.
- File service allows storing and access of data on the cloud using REST APIs or the SMB protocol.

Data Management:

- Azure Search provides text search and a subset of OData's structured files using REST or SDK APIs.
- Cosmos DB is a NOSQL database service that implements a subset of file the SQL SELECT statement on JSON documents.
- Redis cache is a managed implementation of Redis.
- StorSimple manages storage task between on-premises devices and cloud storage.
- SQL database, formerly known as SQL Azure database, works to create, Scale and extended applications into the cloud using Microsoft SQL server technology. It is also Integrates with Active Directory and Microsoft System Center and Hadoop.
- SQL Data Warehouse is a data warehousing service designed to handle computational and data intensive queries on datasets exceeding 1TB.
- Azure data factory, is a data integration service that

allows creation of data-driven Workflows in the cloud for orchestrating and automating data movement and data transformation.

- Azure Data lake is scalable data storage and analytic service for big-data analytics Workloads that require developers to run massively parallel queries.
- Azure HDInsight is a big data relevant service that deploys Hortonworks Hadoop on microsoft Azure, and supports the creation of Hadoop clusters using linux with Ubuntu.
- Azure Stream Analytics is a serverless scalable event processing engine that enables users to develop and run real-time analytics on multiple streams of data from sources such as devices, sensors, Web sites, social media, and other applications.

Conclusion –

Executing application in the cloud offer many advantages over the traditional way of running program.

Through this case study we learned about Microsoft Azure a cloud computing platform and infrastructure, created by Microsoft, studied about its features, services and data management techniques.

Many companies use Microsoft azure cloud services for their business.

Assignment - 2

Title – Google App Engine

Objective – to install and configure Google App Engine

Problem Statement – Installation and configure Google App Engine

Software and Hardware requirements –

Hardware – pc desktop/laptop with i5 processor and 4 gb ram

Software-

Operating system – LINUX

Google cloud software development kit

Python 2.7 or above

Theory –

Introduction

Google App Engine is a web application hosting service. an application or service accessed over the Web, usually with a web browser: storefronts with

shopping carts, social networking sites, multiplayer games, mobile applications, survey

applications, project management, collaboration, publishing, and all the other things we're

discovering are good uses for the Web. App Engine can serve traditional website content too,

such as documents and images, but the environment is especially designed for real-time dynamic

applications

Google App Engine is designed to host applications with many simultaneous users. When an application can serve many simultaneous users without degrading

performance, we say it scales. Applications written for App Engine scale automatically. As more

people use the application, App Engine allocates more resources for the application and manages

the use of those resources. The application itself does not need to know anything about the

resources it is using.

Google App Engine:

It is a platform-as-a-service (PaaS) Cloud computing platform that is fully managed and uses inbuilt services to run your apps. You can start development almost instantly after downloading the software development kit (SDK). You can go on to the developer's guide right away when you click on the language you wish to develop your app in.
As soon as you have signed up for a Cloud account, you can build your app:

- With the template/HTML package in Go
- With Jinja2 and webapp2 in Python
- With Cloud SQL in PHP
- With Maven in Java

Generally Available Features

These are covered by the depreciation policy and the service-level agreement of the app engine. Any changes made to such a feature are backward-compatible and implementation of such a feature is usually stable. These include data storage, retrieval, and search; communications; process management; computation; app configuration and management.

- Data storage, retrieval, and search include features such as HRD migration tool, Google Cloud SQL, logs, datastore, dedicated Memcache, blobstore, Memcache and search.
- Communications include features such as XMPP. channel, URL fetch, mail, and Google Cloud Endpoints.
- Process management includes features like scheduled tasks and task queue
- Computation includes images.
- App management and configuration cover app identity, users, capabilities, traffic splitting, modules, SSL for custom domains, modules, remote access, and multitenancy.

Steps to install and configure google app engine:

1. Create a Google Cloud Platform project

2. Download google cloud SDK archive for your system depending upon the system os and architecture

3. Extract the archive to any location on your file system; preferably, your Home folder.

4. If you're having trouble getting the gcloud command to work, ensure your $PATH is defined appropriately. Use the install script to add Cloud SDK tools to your path. You will also be able to opt-in to command-completion for your bash shell and usage statistics collection during the installation process. Run the script using this command: ./google-cloud-sdk/install.sh

5. Restart your terminal for the changes to take effect.

6 Initialize the SDK

7 Use the gcloud init command to perform several common SDK setup tasks. These include authorizing the SDK tools to access Google Cloud Platform using your user account credentials and setting up the default SDK configuration.

8 Run core gcloud commands

Conclusion –

From the above lab practical we learned how to implement and configure google app engine using googles own google SDK and commands

Assignment - 3

Title – Apex programming Language application salesforce

Objective – to Create an Application using Apex programming Language in SalesForce.com

Problem Statement – Creating an Application in SalesForce.com using Apex programming Language

Software and Hardware requirements –

A computing device with a display having internet connectivity and capable to run web browser

Operating system – any

Web browser – any

Theory –

Introduction

What is Apex?

Apex is a proprietary language developed by the Salesforce.com. As per the official definition, Apex is a strongly typed, object-oriented programming language that allows developers to execute the flow and transaction control statements on the Force.com platform server in conjunction with calls to the Force.com API.

It has a Java-like syntax and acts like database stored procedures. It enables the developers to add business logic to most system events, including button clicks, related record updates, and Visual force pages. Apex code can be initiated by Web service requests and from triggers on objects. Apex is included in Performance Edition, Unlimited Edition, Enterprise Edition, and Developer Edition.

Apex Applications

We can use Apex when we want to −

 Create Web services with integrating other systems.

 Create email services for email blast or email setup.

 Perform complex validation over multiple objects at the same time and also custom validation implementation.

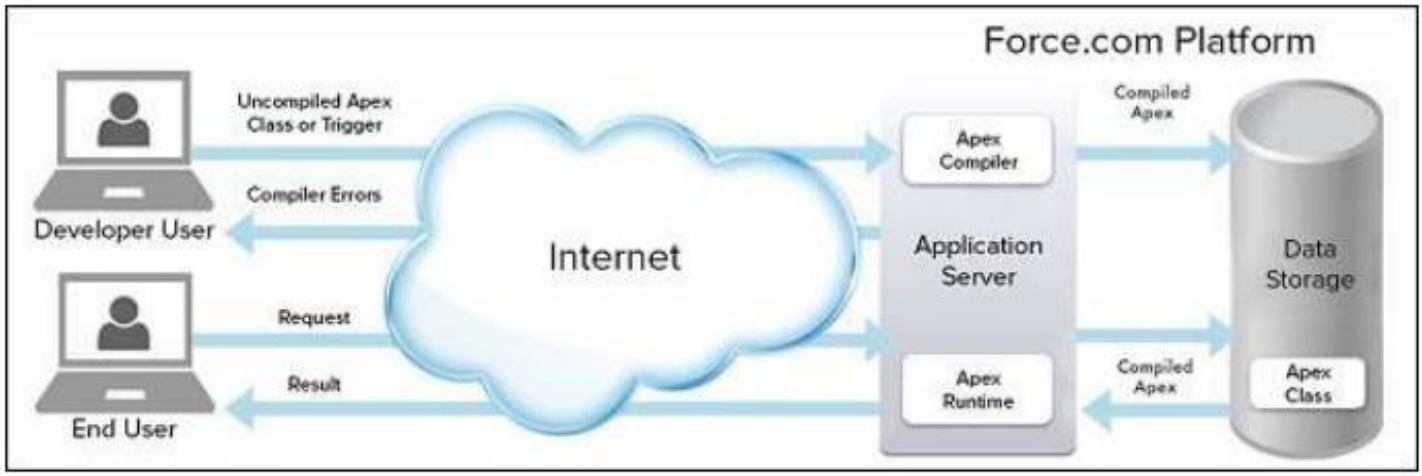
 Create complex business processes that are not supported by existing workflow functionality or flows.

 Create custom transactional logic (logic that occurs over the entire transaction, not just with a single record or object) like using the Database methods for updating the records.

 Perform some logic when a record is modified or modify the related object's record when there is some event which has caused the trigger to fire.

Working Structure of Apex

As shown in the diagram below (Reference: Salesforce Developer Documentation), Apex runs entirely on demand Force.com Platform



Flow of Actions

There are two sequence of actions when the developer saves the code and when an end user performs some action which invokes the Apex code as shown below

Developer Action

When a developer writes and saves Apex code to the platform, the platform application server first compiles the code into a set of instructions that can be understood by the Apex runtime interpreter, and then saves those instructions as metadata.

End User Action

When an end-user triggers the execution of Apex, by clicking a button or accessing a Visualforce page, the platform application server retrieves the compiled instructions from the metadata and sends them through the runtime interpreter before returning the result. The end-user observes no differences in execution time as compared to the standard application platform request.

Steps to create an Application in SalesForce.com using Apex programming Language\

1. Create new org: https://developer.salesforce.com/signup
2. After signup, logging using following URL https://login.salesforce.com/
3. Go to Developer Console for writing Program
4. Click on file then Select Apex class and write a code
5. Select tabs section (for creating GUI)
6. Open a pages
7. Write code in the Program Windows

Conclusion –

After performing the above lab practical we learned how to Create an Application in SalesForce.com using Apex programming Language and learned about apex architecture and its applications.

Assignment – 4

Title – Salesforce Cloud custom Application

Objective – to develop an Application using Salesforce Cloud

Problem Statement – Design and develop custom Application (Mini Project) using Salesforce Cloud

Software and Hardware requirements –

A computing device with a display having internet connectivity and capable to run web browser

Operating system – any

Web browser – any

Theory –

Introduction

Salesforce.com Inc. is an American cloud-based software company headquartered in San Francisco, California. Though the bulk of its revenue comes from a customer relationship management (CRM) product, Salesforce also sells a complementary suite of enterprise applications focused on customer service, marketing automation, analytics and application development.

Salesforce is the primary enterprise offering within the Salesforce platform. It provides companies with an interface for case management and task management, and a system for automatically routing and escalating important events. The Salesforce customer portal provides customers the ability to track their own cases, includes a social networking plug-in that enables the user to join the conversation about their company on social networking websites, provides analytical tools and other services including email alert, Google search, and access to customers' entitlement and contracts.

Lightning Platform

Lightning Platform (also known as Force.com) is a platform as a service (PaaS) that allows developers to create add-on applications that integrate into the main Salesforce.com application. These third-party applications are hosted on Salesforce.com's infrastructure.

Force.com applications are built using declarative tools, backed by Lightning and Apex (a proprietary Java-like programming language for Force.com) and Lightning and Visual force (a framework that includes an XML syntax typically used to generate HTML). The Force.com platform typically receives three complete releases a year. As the platform is provided as a service to its developers, every single development instance also receives all these updates.

Steps to Design and develop custom Application using Salesforce Cloud –

1. Click on Lightning Experience
2. Click on Setup and select Setup for current App.
3. Click on Create an Object
4. Click on Object Manager Tab next to Home Tab
5. Click on Create – Custom Object
6. New custom object page Open , Label as a-Comment, Plural label- comments
7. Give Record Name as –comment name, Data type- text
8. Select Allow Reports Check Box
9. Click on Save
10. Click on Home-Search Tabs in Quick search ,Select Custom Object-Click on New
11. For Object Select Comment, For Tab Style Select Any Icon
12. Click-Next-Next-Save
13. Search App Manager in Quick Search and select app manager
14. Enter name to app name, Click on Next-Next-Next.
15. Select Items (Contacts,Comment), Click on Next
16. Select Profiles (System Administrator) and move to selected profile.
17. Click on Save and Finish.
18. Click on App Launcher Symbol and Select Comment Box App
19. Tour the app
20. Try out mobile app
    - Select Chrome developer tools
    - Open Chrome-Right Click on Chrome page
    - Select Inspect
    - Click Toggle Device Mode Button to simulate your browser as a mobile device
21. To simulate the sales force mobile app in your browser, copy and paste in url from previous tab.Delete the part of the url immediately.

- Click on Left navigation bar
- Find comment object under recent and click on it
- Click new to create a comment

Conclusion – After performing the above lab practical we learned how to Design and develop custom Application using Salesforce Cloud, we also learned about the salesforce lightning platform.