

ASP.Net 4.0 SOA C#4.0 AJAX  
Design Patterns WCF  
PLINQ Web Services WWF  
WPF XML VB.Net NET 4.0  
Sql Server 2008

Microsoft   
**Visual Studio**  
2010

**Certifications**  
MCP, MCTS (ASP.Net, C#)

*Get the rewards*

respect, and recognition you deserve,

Become a

# Microsoft Certified

Technology Specialist  
(MCTS)

## Job Oriented Training Kit

SOA Architecture with WCF Implementation  
SQL SERVER 2008



**MIRACLE  
TECHNOLOGIES**

1st Floor, B-12, Behind DELITE  
Showroom, Adjacent to Metro Station,  
Near Gol Chakkar Sector-2, Noida

Ph. : +91-120-3058446  
+91-120-3058447  
Mobile : 9311305845/6/7

Mail ID :  
mcts@miracleindia.com  
www.miracleindia.com

**.NET Framework 4.0 builds incrementally on the new features added in .NET Framework 4.0. For example, feature sets in Windows Workflow Foundation (WF), Windows Communication Foundation (WCF), Windows Presentation Foundation (WPF) and Windows CardSpace. In addition, .NET Framework 4.0 contains a number of new features in several technology areas which have been added as new assemblies to avoid breaking changes. They include the following: :-**

- Deep integration of Language Integrated Query (LINQ) and data awareness. This new feature will let you write code written in LINQ-enabled languages to filter, enumerate, and create projections of several types of SQL data, collections, XML, and Datasets by using the same syntax. Support for Parallel LINQ for multi processor processing**

- ASP.NET AJAX lets you create more efficient, more interactive, and highly personalized Web experiences that work across all the most popular browsers.**

- New Web protocol support for building WCF services including AJAX, JSON, REST, POX, RSS, ATOM, and several new WS-\* standards.**

- Full tooling support in Visual Studio 2010 for WF, WCF, and WPF, including the new workflow-enabled services technology.**

- New classes in .NET Framework 4.0 base class library (BCL) that address many common customer requests. And better support to dynamic languages with Dynamic Language Runtime.**

## VS 2010 & Framework 4.0

### .New Features in Vs2010, Framework 4.0 & ASP.NET4.0

- Ø Dynamic Language Runtime
- Ø Breakpoint in XML (Import and Export)
- Ø Pinning
  - a) Project Pinning
  - b) Show Previous Debugged Data
  
- Ø Export Pin as xml.
- Ø Improved Intellisense.
  - a) Complete Text Search
  - b) Pascal Searching
  - c) Filtered Search
  
- Ø Starter Project Templates
  - a) Project Pinning
  - b) New User Interface in WPF
  
- Ø Multi-Targeting
- Ø Multiple Monitor Support
- Ø HTML / ASP.NET / JavaScript Code Snippets
- Ø Auto-Start ASP.NET Applications
  - a) Better Support for ASP.NET Applications
  
- Ø )Searching and Navigating Code in VS 2010
- Ø )WPF 4
- Ø Add Reference Dialog Improvements
  
- Ø SEO Improvements with ASP.NET 4.
  - a) MetaKeywords of Page Class
  - b) MetaDescription of Page Class
  
- Ø Built-in Charting Controls for ASP.NET and Windows Forms
- Ø Cleaner HTML Markup with ASP.NET 4 - Client Ids
  
- Ø C# 4.0
  - a) Optional Parameters
  - b) Named Arguments with ASP.NET MVC 2
  
- Ø Automatic Properties
- Ø New <%: %> Syntax for HTML Encoding Output using ASP.NET 4
- Ø JavaScript Intellisense Improvements with VS 2010
- Ø Box Selection
- Ø Multi-line Editing Support with VS 2010
  - a) Multiline Writing
  - b) Multiline Pasting

# Fundamentals

## .Net Framework + Language Basics + Windows Application

- Ø How .Net evolved & Why .Net?
- Ø What is DLL Hell?
- Ø What is Assembly? Why the need arise to create assembly when we had dll & exe's.
- Ø Is .Net a framework or architecture? Why?
- Ø Introduction to CLR , CTS & CLS
- Ø Managed v/s Unmanaged Code
- Ø Metadata
- Ø Types
- Ø NET Framework Class Library
- Ø Interface-Based Programming
- Ø Everything is an Object
- Ø ILDASM

## Components of CLR

- Ø Security
- Ø Class Loader
- Ø Garbage Collector
- Ø COM marshaler
- Ø Exception handler
- Ø Type Checker
- Ø Common Type System
- Ø Common Language Specification

## Memory Management

- Ø Garbage Collection Performance
- Ø Generations
- Ø Processes
- Ø Threads

## Implementing Dispose & Finalize

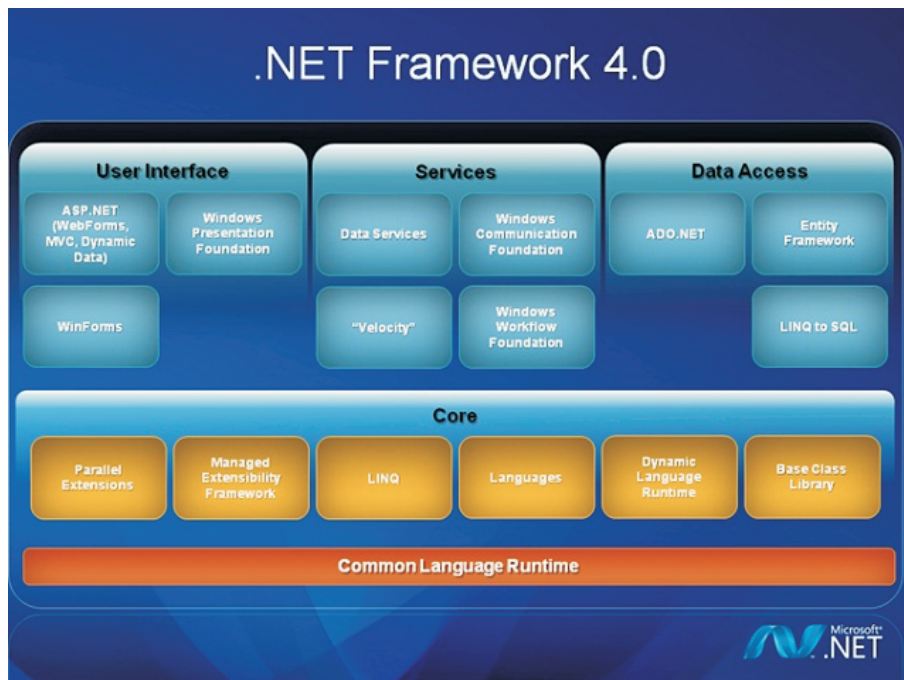
- Ø Demo

## A tour of Visual studio IDE.

- Ø Creating a new Project
- Ø Working on Windows Form
- Ø The Solution Explorer
- Ø The Properties Window and Dynamic Help
- Ø References in Visual Studio
- Ø The Solution Explorer

## Ø The Menu Structure of Visual Studio.Net

- Ø Demo



## C#

### Kick Starting OOPS & .Net

- Ø Encapsulation
- Ø Creating UDT
- Ø Class
- Ø Struct
- Ø Interface,
- Ø Enum
- Ø Events
- Ø Delegates
- Ø Abstraction
- Ø Access modifiers in .Net
  - Public
  - Private
  - Protected
  - Internal
  - Protected Internal

Lab: Using Access modifiers in your project

- Ø Polymorphism ( Runtime & Compile time )
  - Compile Time Polymorphism
  - Function Overloading
  - Operator Overloading
  - Runtime Polymorphism
  - Virtual Functions
  - Abstract Functions
  - Overriding

Lab: Implementing Polymorphism

- Ø Inheritance
  - Hiding
  - Overriding
  - Abstract Class

Lab: Implementing Inheritance

- Ø Interface Polymorphism
  - Implementing Interfaces
  - Concept of CTS

Lab: Implementing interfaces

### Error Handling & Event Logging

- Ø Try, Catch & Finally.
- Ø Creating Custom logs.
- Ø Writing to Event Logs.
- Ø Switch Statement.
- Ø Trace Listener, Text, XML, Eventlog listener
- Ø Debug Class

### Instrumentation

- Ø Logging Events.
- Ø Creating & Deleting Logs, Writing to Events Logs.
- Ø Debugging & Tracing, Debug Attributes.

## Windows Programming

### Events & Delegates

- Ø Understanding the publisher/subscriber model
- Ø What is an Event?
- Ø What is Delegate?
- Ø Creating Custom Delegates
- Ø Creating Custom Events
- Ø Creating your own Event Arguments
- Ø Covariance and Contravariance

- Ø Multicast Delegate
- Ø Custom Event Handling
- Why Events when we are having Delegate?
- Ø Demo
- Ø Lab : Implementing Event & Delegates

### **Threading**

- Ø Why Threading?
- Ø Benefits of Multicore Processor
- Ø Thread Start Delegate
- Ø Parameterized Thread Start Delegate
- Ø Create and Manage Multiple Threads

### **Synchronization**

- Monitor Class
- Lock
- Join
- InterLocked
- Reader Writer Lock
- Mutex
- Semaphore
- Ø Theard Priority
- Ø Using Data to thread
- Ø Stopping thread

### **Assemblies**

- Ø Private v/s Shared Assembly
- Ø Creating Shared Assembly
- Ø Assembly Versioning
- Ø Reading Assembly metadata
- Ø Global Assembly Cache
- Ø Delay Signing
- Ø Demo

Ø Lab: Creating strong named assemblies and registering in GAC

### **Collections**

- Ø Array List
- Ø Stack
- Ø Queue
- Ø Bit Array
- Ø Hash Table
- Ø Sorted List
- Ø IEnumerable and IEnumerator

### **Specialized**

- Ø String Collection
- Ø String Dictionary
- Ø Name Value Collection
- Ø List Dictionary
- Ø Ordered Dictionary
- Ø Hybrid Dictionary

### **Generics**

- Ø List<T>
- Ø Stack<T>
- Ø Queue<T>
- Ø Dictionary<T,T>
- Ø Create own Generics

### **File System**

- Ø System IO Namespace.
- Ø Stream & FileStream Class
- Ø StreamReader & StreamWriter
- Ø BinaryReader & BinaryWriter

- Ø File & FileInfo Class
- Ø Directory & DirectoryInfo Classes
- Ø DriveInfo Classes
- Ø Path Class
- Ø Memory Stream
- Ø Isolated Storage
- Ø Compression Stream
- Ø GZIP Stream
- Ø Encoding and Decoding
- Ø UTF 7, 8,16,32
- Ø Lab: Implementing all classes
- Ø Lab: Creating a tool using these classes

### **Windows Services**

- Ø What is a Windows Service
- Ø How to create a Service Project.
- Ø How to install Project for service.

### **Regular Expressions & Encoding & Decoding**

- Ø Use of Regular Expression for Pattern Matching
- Ø Extract Matched Data
- Ø Create a Regex Expression Evaluator
- Ø Encoding & Decoding
  - 1) Understand UTF Format
  - 2) UTF, 7, UTF-8, UTF-16, UTF-32

### **Graphics**

- Ø Set Location and Size of Controls Dynamically.
- Ø Pen Class
- Ø Solid, Liner & Gradient Brush Class
- Ø Hatch Brush and Hatch Style
- Ø Create and Save Images
- Ø Load Images and Forms
- Ø Pie Chart
- Ø Working with Images
- Ø Formatting Text
- Ø Add 2D Graphics with Text
- Lab
  - Ø Application using Brushes, Pen, FontFamily, Font class
  - Ø Shapes and Size

### **Mailing**

- Ø Process of Creating & Sending an E-mail Message
- Ø Mail Message Class
- Ø SMTP Client
- Ø Attach Files
- Ø HTML E-Mails
- Ø Linked Resource
- Ø Alternate View
- Ø Handle Mail Exceptions
- Ø Configure Credentials
- Ø Configure SSL

### **Instrumentation**

- Ø Logging Events
- Ø Using Microsoft Events and Logging
- Ø Creating & Deleting Events Logs
- Ø Writing to Events Logs
- Ø Debugging & Tracing
- Ø Debug Attributes
- Ø Listener Object, Overview of Processes
- Ø Process Class

## Interoperation

- Ø Using COM Objects
- Ø Import Type Libraries
- Ø Expose COM Components in .Net
- Ø Exposing .Net components to COM

## Wrapping Up

- Ø Interview Questions .Net framework and C#
- Ø Test of Proficiency

## Data Access in .Net - ADO.Net Concepts

- Ø What is ADO.Net
- Ø How to use ADO.Net
- Ø Modes of ADO.Net

## Data Providers

- Ø SQL Data Provider
- Ø Oracle Data Provider
- Ø Oledb Data Provider
- Ø Object Data Provider
- Ø Linq Data Provider

## Connected Architecture

- Ø \*\*\*Connection Classes
- Ø \*\*\*Command Classes
- Ø Demo
- Ø Lab: Accessing and Manipulating database using Datareader and Command Classes

## Disconnected Architecture

- Ø Data Set Objects & Data Table Objects
- Ø \*\*\*DataAdapter
- Ø Database manipulation using dataset
- Ø DataRelations
- Ø Creating dataset and DataTable using Code
- Ø ADO.Net and XML
- Ø Demo
- Ø Lab: Accessing and Manipulating database using DataAdapter and Dataset
- Ø Lab: Implementing DataRelations
- Ø Lab: Reading and Writing XML using DataSet

## ADO.NET

- Ø Advantage of ADO.NET
- Ø SQL BULK COPY CLASS
- Ø New Features in Data Tables Objects.
- Ø Batch Processing Multiple Updates.
- Ø Demo

## Continue with ADO .Net

- Stored Procedure
- Functions
- In and Out Parameter in Stored Procedure
- N-Tier Application Calling
- Calling With Generics
- TOOLS:
- Ø Lab: 2 tools to be created by student using ADO.Net

## Wrapping Up

- Ø Interview Questions on ADO.Net
- Ø Test of Proficiency

## ASP.NET

Overview of the ASP .Net architecture

- Ø What is the ASP.Net Platform?
- Ø What is the ASP.Net Framework?
- Ø Describe the ASP.Net Framework and its components.
- Ø .Net Framework Base Classes

### ASP .Net Pipeline Model

- Ø HTTP Application
- Ø HTTP Module
- Ø HTTP Handler
- Ø INET Info.exe
- Ø Writing ASP.Net Pages
- Ø Coding Issues Web v/s Windows
- Ø Page Class
- Ø Inline V/s Code Behind models

### Controls in ASP.Net

- Ø HTML Server Controls
- Ø Writing ASP.Net Pages
- Ø Coding Issues Web v/s Windows
- Ø Page Class
- Ø Using Code Behind

### Creating Controls

- Ø Building User Control
- Ø Building custom Controls
- Ø Demo
- Ø Lab: Creating User & Custom Controls & using in a simple Web Site

### CSS

- Ø Introducing Caching Style Navigation
- Ø Investigating Styles
- Ø Inline Stylesheet
- Ø Internal Stylesheet
- Ø External Stylesheet
- Ø Using Visual Studio 2008' tool for working with styles
- Ø Tables
- Ø Creating a Consistent Website
- Ø Investigating Styles
- Ø Using Visual Studio 2008's Tools for Working with Styles

### Theme and Master Pages

- Ø Why Theme?
- Ø Theme vs CSS
- Ø Priority of Themes
- Ø Master Pages Introductions
- Ø Master Pages vs Web User Controls
- Ø Contnet Place Holder
- Ø CSS/Theme with Master Page
- Ø Change Master page/ Theme dynamically at runtime
- Ø Find controls from Master Page

### Site Navigation

- Ø ASP .Net Website Navigation
- Ø Creating an XML Site Map File
- Ø Site Navigation Controls
- Ø Site Navigation API
- Ø SitemapPath
- Ø Menu Control

- Ø Tree View
- Ø Sitemap File

### **Web Parts**

- Ø Webpart Zone
- Ø Catalog Zone
- Ø Editor Zone
- Ø Webpart Manager

### **Server Controls & HTML Controls**

- Ø What are Server Controls?
- Ø The ASP.Net Page Model Revisited
- Ø Server Control are Event Driven
- Ø When should I use?
- Ø Controls in ASP.Net
- Ø HTML Server Controls
- Ø Demo
- Ø Lab: Using Asp.Net Controls

### **Validation Controls**

- Ø The Base Validator Class
- Ø Required Field Validator Control
- Ø Compare Validator Control
- Ø Range Validator Control
- Ø Regular Expression Validator Control
- Ø Custom Validator Control
- Ø Validation Summary Control
- Ø Cancelling Validation of a Page
- Ø Validation Groups
- Ø Demo
- Ø Lab: Creating a simple website using validation controls

### **ASP.Net Web Form Controls**

- Ø The Web Control Base Class
- Ø Using the Web Form Controls
- Ø The Auto post Back feature
- Ø List Controls( ListBox, ListView)
- Ø ASP.Net Data Grid Controls
- Ø Data Binding Concept
- Ø Adding Styles and using Templates
- Ø Demo
- Ø Lab: Implement List Controls

### **What's New with ASP.Net 2.0**

- Ø GridView Control
- Ø FormView
- Ø Wizard Control
- Ø Multiview Control
- Ø DetailView
- Ø Datasource Controls  
( XmlDataSource, ObjectDataSource, AccessDataSource, SqlDataSource)
- Ø Web Parts Control
- Ø Site Navigation
- Ø Miscellaneous
- Ø Partial class
- Ø Master Page Template
- Ø Themes and Skins
- Ø Demo
- Ø Lab: Creating Custom navigation using SiteMap and Database
- Ø Lab: Creating master pages and using in web application
- Ø Lab: Creating Themes and Skins

### **Head start with C# 3.0**

- Ø New features and enhancements
- Ø Anonymous methods
- Ø Lambda Expressions
- Ø Implicitly typed variables
- Ø Extension methods
- Ø Anonymous types
- Ø Initializers
  - Ø Collection Initializers
  - Ø Object Initializers

### **ASP.Net Web Form Objects**

State Management (Client objects)

- Ø Page-Level State
- Ø Cookies
- Ø Query String
- Ø View State
- Ø Hidden Fields
- Ø Cross Page Posting

State Management (Server objects)

- Ø Application
- Ø ASP.NET Session State
- Ø Storing Objects in Session State
- Ø Configuring Session State
- Ø Using Cookieless Session IDs
- Ø Cache
- Ø Demo
- Ø Lab: State management using above

### **Data Bound Controls**

- Ø GridView Control
- Ø Data List
- Ø Repeater
- Ø SQL Data Source
- Ø Access Data Source
- Ø Linq Data Source
- Ø Object Data Source
- Ø FormView
- Ø Details View
- Ø List View
- Ø Data Pager

Web Applications

- Ø Web Applications and Global.asax
- Ø Demo

### **XML Reader/Writer**

- Ø XML Tree
- Ø XML Reader
- Ø XML Writer
- Ø XML Attributes
- Ø XSLT

### **XPATH**

- Ø XPath Intro
- Ø XPath Nodes
- Ø XPath Syntax
- Ø XPath Operators

## Configuration

- Ø ASP.Net Configuration
- Ø Configuration Settings
- Ø Page Configurations
- Ø Application Settings
- Ø Tracing & Debugging

## Securing ASP.Net Applications

- Ø Working with User Profile
- Ø User Profile Basics
- Ø Defining User Profiles
- Ø Identifying Users
- Ø Set & Save a User Profile
- Ø IIS security Overview
- Ø The Types of Authentication and Authorization
- Ø Windows
- Ø Passport
- Ø Forms-Based

## Deployment

- Ø Publishing
  - 1) Copy Web Tool
  - 2) Allows pages to be Precompiled
  - 3) Allow pages to be not updatable
  - 4) Host Application on IIS
  - 5) Virtual Directory Concept

## Ø FTP

- 1) Introduction of FTP
- 2) File Transfer using FTP
- 3) File Zila and Cute FTP

# Distributed Programming

## Reflection

- Ø Understand Assemblies and Modules
- Ø Loading Assemblies Dynamically
- Ø Type Class
- Ø Assembly Class
- Ø Method Info
- Ø Property Info
- Ø Parameter Info
- Ø Field Info
- Ø Constructor Info
- Ø Application Domains
- Ø Get Private Fields and Members
- Ø Activator Class
- Ø Lab: Creating Tool using Reflection

## Web Services

- Ø Web Service Overview
- Ø Introduction to Protocols
- Ø Http-Get
- Ø Http-Post
- Ø SOAP Protocols
- Ø Web Method and Web Service Attributes
- Ø What is a Web Service?
- Ø Creating a Web Service
- Ø Describing Web Services
- Ø What is WSDL?
- Ø Web Service Proxies
- Ø Web Service Security

- Ø Lab: Creating and Exposing a web service
- Ø Lab: Implementing Web Service security

#### Wrapping Up

- Ø Interview Questions on Distributed Applications
- Ø Test of Proficiency

## Advanced Topics

### AJAX in .Net

- Ø Start With Java Script
- Ø Create Functions
- Ø Create Class
- Ø Create Interface
- Ø Var Keyword
- Ø Ajax Life Cycle
- Ø Installing Ajax Toolkit
- Ø Using AJAX controls
- Ø Using XMLHttpRequest Object

### Object Relational Modeling

- Ø LINQ
- Ø DLINQ
- Ø XLINQ
- Ø Demo
- Ø Lab: Implementing everything that you have learnt.

### Design Patterns

Design patterns is a reusable, high quality solution to a given requirement, task or recurring problem. Further it does not comprise of a complete solution that may be instantly converted to a code component, rather it provides a framework for how to solve a problem. Because design patterns consist of proven reusable architectural concepts, they are reliable and they speed up software development process.

Three types of Design patterns

- 1) Creational Patterns
- 2) Structural Patterns
- 3) Behavioural Patterns

- Ø Gang of Four
- Ø Factory
- Ø Abstract Factory
- Ø Façade
- Ø Singleton
- Ø Bridge
- Ø Proxy
- Ø Observer
- Ø Prototype

### SOLID Principle

Initial	Stands for	Concept
<b>S</b>	<b>SRP</b>	<a href="#">Single responsibility principle</a> the notion that an <b>object</b> should have only a single responsibility.
<b>O</b>	<b>OCP</b>	<a href="#">Open/closed principle</a> the notion that "software entities ... should be open for extension, but closed for modification".
<b>L</b>	<b>LSP</b>	<a href="#">Liskov substitution principle</a>
<b>I</b>	<b>ISP</b>	<a href="#">Interface segregation principle</a> "many client specific interfaces are better than one general purpose interface."
<b>D</b>	<b>DIP</b>	<a href="#">Dependency inversion principle</a> one should "Depend upon Abstractions. Do not depend upon concretions"

## **WCF - Overview**

Accelerate's Windows Communication Foundation (WCF) training teaches Microsoft's new framework for building distributed systems. It unifies and builds on the diverse set of existing distribution mechanisms, which include ASP.NET Web services and .NET Remoting. WCF enables developers to produce highly configurable, secure, reliable and transactional services using a single simplified programming model. And since WCF supports the WS-\* series of Web service standards, it enables simple interoperation with other platforms and technologies.

## **Hands-on/Lecture Ratio**

This Advanced C# training class is 70% hands-on, 30% lecture.

## **WCF Training Objectives**

### **Ø ABC of WCF**

- 1) Address
- 2) Binding
- 3) Contract

### **Ø Introduction to WCF**

- 1) Review the Challenges of Building Distributed Applications
- 2) Examine the Concept of Service Oriented Application
- 3) See a High Level Overview of Windows Communication Foundation
- 4) Building and Host your First WCF Service

### **Ø Hosting and Calling WCF Services**

- 1) Understand how to Host a WCF Service
- 2) Understand how Clients Communicate with Services
- 3) See how to Host and Call a Service using Multiple Bindings
- 4) See how to Configure Endpoint in Code and in Configuration Files

### **Ø Bindings and Contracts**

- 1) Understand how to Work with Bindings
- 2) Explore how to Create Service and Data Contracts
- 3) See the Implications of Modifying Service and Data Contracts

### **Ø Exceptions, Diagnostics and Message Patterns**

- 1) Learn how to Handle Exceptions in WCF Services
- 2) Explore One-Way and Duplex Communication

### **Ø Sessions and Transactions**

- 1) See how to use Sessions to Maintain State in WCF Services
- 2) Explore Options for Controlling the Lifetime of a Service Instance

### **Ø WCF Security**

- 1) Explore the Basics of WCF Security
- 2) See how to Authenticate Callers of a Service
- 3) See how to Authorize Callers Based on Roles

### **Ø Workflow Services**

- 1) Learn how to Expose a Workflow as a WCF Service
- 2) See how to Call a WCF Service from a Workflow
- 3) Understand how to Build a Stateful Workflow Service

## Windows Workflow Foundation - Overview

Windows Workflow Foundation is a ground-breaking addition to the core of the .NET Framework that allows you to orchestrate human and system interactions as a series of workflows that can be easily mapped, analyzed, adjusted, and implemented.

As business problems become more complex, the need for a workflow-based solution has never been more evident. WF provides a simple and consistent way to model and implement complex problems. As a developer, you focus on developing the business logic for individual workflow tasks.

### WF Training Objectives

#### Ø Introduction

- 1) What is Windows Workflow Foundation?
- 2) Creating Your First Workflow
- 3) Hosting a Workflow in a Windows Application

#### Ø Handling Conditions and Exceptions

- 1) Introducing Conditional Activities
- 2) If/Else Activity
- 3) While Activity
- 4) Using a Custom Activity
- 5) Handling Exceptions

#### Ø Flow Activities

- 1) Introducing Flow Activities
- 2) Delay Activity and Code Activity

#### Ø Creating State Machine Workflows

- 1) Introduction to State Machines
- 2) State Machine Activities
- 3) Create a State Machine Workflow

#### Ø Creating Custom Activities

- 1) Introducing Custom Activities
- 2) Creating a Simple Activity
- 3) Creating a More Complex Activity
- 4) Creating Composite Activities

#### Ø Workflows and Web Services

- 1) Workflow-Based Web Services
- 2) Build a Workflow Web Service
- 3) Invoke a Web Service in a Workflow

## Windows Presentation Foundation - Overview

Windows Presentation Foundation or WPF ("Avalon"), the new .NET technology from Microsoft for building rich Windows applications. It is part of .NET 3.0/3.5, previously called "WinFX" by Microsoft. WPF includes an XML-based markup language for defining program elements, Extensible Application Markup Language (XAML). WPF applications can be created using only code or a combination of code and XAML pages.

### WPF Training Outline

- Ø Introducing WPF
- 1) Understand the motivation behind WPF
- 2) Examine the various 'flavors' of WPF applications
- 3) Overview the services provided by WPF
- 4) Examine the core WPF assemblies and namespaces
- 5) Learn the syntax of XAML
- 6) Understand the XAML / code relationship

### Ø WPF Controls

- 1) Survey the core WPF control types
- 2) Review the WPF control programming model
- 3) Learn to position controls using layout managers
- 4) Understand the role of WPF control commands

### Ø WPF Graphical Rendering Services

- 1) Understand the scope of WPFs graphical rendering services
- 2) Work with the 2D & 3D
- 3) Work with Brushes and Pens
- 4) Apply graphical transformations
- 5) Understand the role of geometries and drawings

### Ø WPF Resource Management

- 1) Learn to manage binary resources
- 2) Understand the role of logical resources
- 3) Work with resources in XAML and procedural code
- 4) Work with resource dictionaries
- 5) Understand the resource lookup mechanism
- 6) Know the role of dynamic resources
- 7) Learn how to make use of system resources

### Ø WPF Styles

- 1) Learn how to define and apply WPF styles
- 2) Learn to limit where a style can be applied
- 3) Build new styles based on existing styles

### Ø WPF Animation Support

- 1) Understand the scope of WPF's animation services
- 2) Define animations in code and XAML
- 3) Work with linear interpolation animations

### Ø WPF Data Binding

- 1) Understand the WPF data binding mechanism
- 2) Make use of data binding operations in XAML and code

### Ø WPF Navigation XBAPs

- 1) Understand the role of navigational-based applications
- 2) Work with the NavigationWindow and Page types
- 3) Manage the history journal
- 4) Understand and create XBAP applications
- 5) Examine XBAP security issues

### Wrapping Up

- Ø Interview Questions on .Net 3.5
- Ø Test of Proficiency

### Sharepoint Server 2007

MOSS 2007 is an integrated suite of server capabilities that can help improve organizational effectiveness by providing comprehensive content management and enterprise search, accelerating shared business processes, and facilitating information-sharing across boundaries for better business insight.

### Ø Design Definition

- 1) The Good, the Bad, and the Ugly
- 2) Where design fits
- 3) People who have input to design
- 4) SharePoint Design considerations

### Ø Web Design Considerations

- 1) Goal-Oriented Design Driving Adoption
- 2) Storyboarding

### 3) Web Page Design Planning for Change

#### Ø General Concept Design

- 1) Getting Inspired Creating the Design
- 2) The Intranet Counterpart Continuing Education

#### Ø Communicating and Collaborating

- 1) Communication/Publishing Site definition
- 2) When to Use a Communication Site
- 3) Effectively Using a Communication/Publishing Site
- 4) Publishing Site Templates
- 5) Communication/Publishing Site Best Practices
- 6) Collaboration/Team Site definition

#### Ø SharePoint Designer Review

- 1) Customizing MOSS Sites
- 2) The Design View
- 3) Working with Master Pages
- 4) Working with CSS
- 5) Working with Web Parts and Web Part Zones

#### Ø Themes

- 1) SharePoint Themes
- 2) How SharePoint Themes Work
- 3) How to Create a Theme
- 4) How to Design a Theme
- 5) Tools for Creating Themes

#### Cascading Style Sheets with MOSS 2007

- 1) Introduction to Cascading Style Sheets
- 2) Working with CSS
- 3) How to Include CSS in Your MOSS Site
- 4) Converting Your Design to HTML and CSS

#### Ø Master Pages

- 1) What Is a Master Page?
- 2) The Master Page Structure
- 3) Using Master Pages with MOSS
- 4) Challenges with Master Pages in MOSS
- 5) The Great Customization Debate
- 6) Implementing a Minimal Master Page
- 7) Converting HTML Design to Master Pages

#### Ø Page Layouts

- 1) Definition
- 2) Internet Site Welcome Page Layout Example

#### Ø Working with Out-of-the-Box Web Parts

- 1) Using the Content Query Web Part
- 2) The Data View Web Part
- 3) The Content Editor Web Part
- 4) The Page Viewer Web Part
- 5) The XML Web Part
- 6) The Image Web Part

#### Ø Customizing Search

- 1) Accuracy vs. Relevancy
- 2) Making Better Information

## **SQL SERVER 2008 - Overview**

SQL Server 2008 delivers on Microsoft's Data Platform vision by helping your organization manage any data, any place, any time. SQL Server 2008 provides the highest levels of security, reliability, and scalability for your business-critical applications. To take advantage of new opportunities in today's fast-moving business world, companies need the ability to create and deploy data-driven solutions quickly. SQL Server 2008 reduces time and cost of management and development of applications. SQL Server 2008 enables data to be consumed from custom applications developed using Microsoft .NET and Visual Studio and from within service-oriented architectures (SOA).

## **SQL Server 2008 Training Objective**

### **Ø Getting Started with Databases and Transact-SQL in SQL Server 2008**

- 1) Overview of SQL Server 2008
- 2) Overview of SQL Server Databases
- 3) Overview and Syntax Elements of T-SQL
- 4) Working with T-SQL Scripts
- 5) Lab : Using SQL Server Management Studio

### **Ø Querying and Filtering Data**

- 1) Using the SELECT Statement
- 2) Filtering Data
- 3) Performance Considerations for Writing Queries
- 4) Lab : Querying and Filtering Data

### **Ø Grouping and Summarizing Data**

- 1) Summarizing Data by Using Aggregate Functions
- 2) Summarizing Grouped Data
- 3) Lab : Grouping and Summarizing Data

### **Ø Joining Data from Multiple Tables**

- 1) Querying Multiple Tables by Using Joins
- 2) Combining and Limiting Result Set
- 3) Lab : Joining Data from Multiple Tables

### **Ø Working with Subqueries**

- 1) Writing Basic Subqueries
- 2) Writing Correlated Subqueries
- 3) Comparing Subqueries with Joins and Temporary Tables
- 4) Using Common Table Expressions
- 5) Lab : Working with Subqueries

### **Ø Modifying Data in Tables**

- 1) Inserting Data into Tables
- 2) Deleting Data from Tables
- 3) Updating Data in Tables
- 4) Lab : Modifying Data in Tables

### **Ø Overview of Views**

- 1) Overview of Stored Procedures
- 2) Create Stored Procedures
- 3) Overview of Triggers
- 4) Creating Stored Procedures
- 5) Lab : Using Programming Objects for Data Retrieval

## Service Oriented Architecture

- Ø What is SOA
- Ø Why SOA?
- Ø SOA with WCF
- Ø Benefits of SOA

## Windows CardSpace

- Ø Introduction to CardSpace

## Layered Architecture (Ntier)

- Ø Project are completely based on Layered Architecture
- Ø Enterprises Library
  - Caching Application Block
  - Encryption Application Block
  - Data Application Block
  - Exception Handling Block
  - Security Application Block

## Silverlight

### **Graphics**

- Bitmap API
- Pixel Shaded Effects
- Hardware Composition
- Cached Composition
- Image Info, Opened Event

### **Animation**

- Easing
- Text Animation
- Point Animation

### **Media**

- H.264 Support
- RAW Audio and Video Support
- 3rd Party Codecs

### **Accessibility, Navigation & SEO**

- System Colors for High Contrast
- Navigation Framework & URI Mapping

### **New Controls & Layout**

- Additional Core Controls
- Silverlight Toolkit
- Multiselect Listbox
- File Save Dialog
- Layout controls
- Wrap Panel
- Dock Panel
- View Box
- Child Window

### **Input**

- Mouse Wheel Event
- Multitouch on Windows 7

### **Text**

- Compressed Fonts
- Caret Brush for TextBox

### **DataBinding**

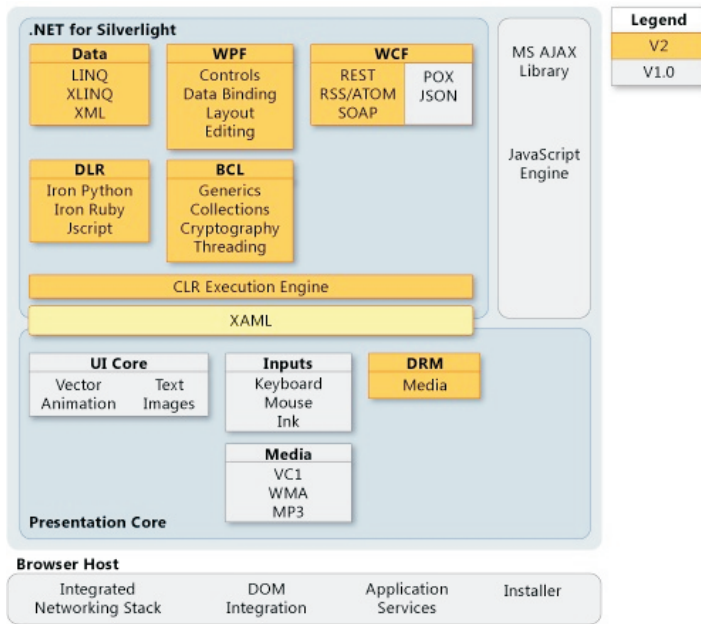
- Element-to-Element binding
- Data Validation & Exception UI

### **Styling**

- "Based On" styles support
- Resetting Styles
- Merged Resource Dictionaries

**Behaviors**

- Blend 3 – Intro
- Triggers & Actions
- RIA Services
- RIA Overview
- Getting Started with RIA

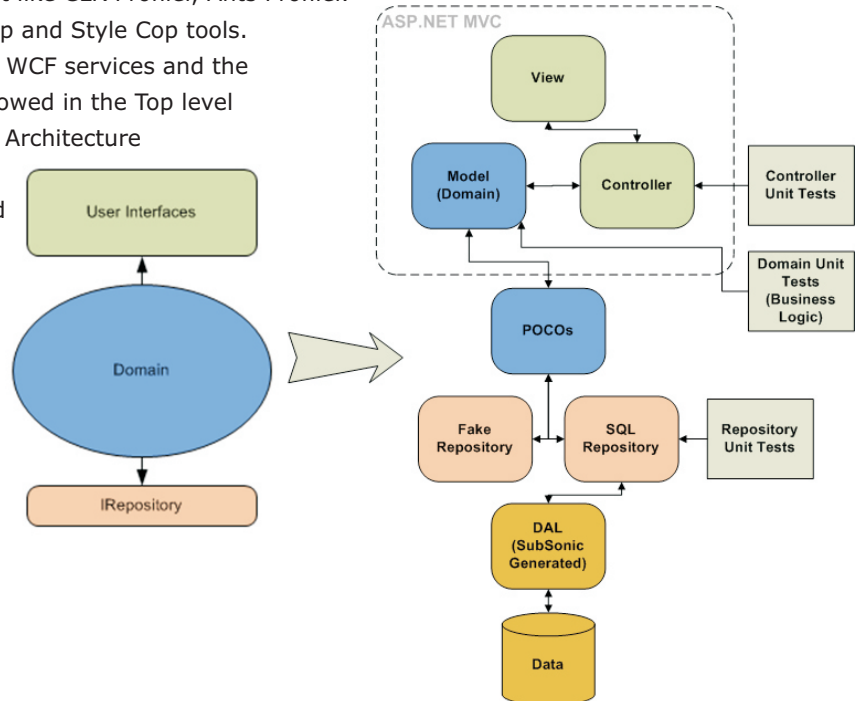


**Project Development Standards**

Project are following the complete life cycle (SDLC). Requirement gathering, Design, Coding with CMM Level5 Standard. Testing with the tools latest in the market like CLR Profiler, Ants Profiler.

Maintaining coding Standards with FX Cop and Style Cop tools. Project are having the High security with WCF services and the MVC or different custom architecture followed in the Top level companies project. And the Most trusted Architecture Service Oriented Architecture (SOA)

Implementation for better reusability and security under the guidance of Project Manager and Software Architect.



## MICROSOFT .NET SKILLS MEAN SUCCESS!

Microsoft's .NET technology redefines what computing means to IT professionals, businesses and end users alike. Microsoft .NET allows applications to communicate over the Internet and intranets regardless of operating system or programming language. This means users experience a highly interactive and individualized computing experience with the ability to share and manipulate data across multiple platforms. As Microsoft.NET gains momentum throughout the industry, IT professionals must start updating their skills now in order to master .NET. IT professionals who gain these new skills will lead the curve in understanding .NET and migrating their systems to .NET technologies. Along with these skills come increased job opportunities and financial rewards. Softcopy also available [microsoft@miracleindia.com](mailto:microsoft@miracleindia.com)

### Soft Copy available for

1. Core Java + Advance Java + J2EE + Struts + Hibernate
2. Dotnet - Visual Studio - 2008
2. Embedded System + Mobil Communication Application
3. TESTING
4. MAINFRAME
5. SOFT SKILLS COMMUNICATIONS
6. ORACLE
7. SAP
8. ERP
9. PHP
10. SEO
11. Project
12. SAS
13. Hardware Networking

[sun@miracleindia.com](mailto:sun@miracleindia.com)  
[microsoft@miracleindia.com](mailto:microsoft@miracleindia.com)  
[embedded@miracleindia.com](mailto:embedded@miracleindia.com)  
[testing@miracleindia.com](mailto:testing@miracleindia.com)  
[mainframe@miracleindia.com](mailto:mainframe@miracleindia.com)  
[skill@miracleindia.com](mailto:sun@miracleindia.com)  
[oracle@miracleindia.com](mailto:oracle@miracleindia.com)  
[sap@miracleindia.com](mailto:sap@miracleindia.com)  
[erp@miracleindia.com](mailto:erp@miracleindia.com)  
[php@miracleindia.com](mailto:php@miracleindia.com)  
[seo@miracleindia.com](mailto:seo@miracleindia.com)  
[project@miracleindia.com](mailto:project@miracleindia.com)  
[sas@miracleindia.com](mailto:sas@miracleindia.com)  
[networking@miracleindia.com](mailto:networking@miracleindia.com)