Training Plan for New Resources

# C# [Course: Ultimate C# Series]

* Development environment and basic coding principles
* Arrays, Methods, and Overloading **[Course: Ultimate C# Series: Part 1]**
* Basic and advanced Object Oriented Programming principle **[Course: Ultimate C# Series: Part 2]**
  + Classes
  + Association between classes
  + Inheritance
  + Polymorphism
  + Interfaces
  + Static Classes and Methods
  + Constant Classes
* Advanced Concepts **[Course: Ultimate C# Series: Part 3]**
  + Generics
  + Lambda Expression
  + Events and Delegates
  + Extension Methods
  + Implicit & explicit Casting
  + Templated classes and methods
  + LINQ
  + PLINK
  + Exception Handling
  + Async/Await
* Unit Tests **[Course: Unit Testing for CSharp Developers]**
  + Fundamentals of Unit Testing
  + Core Unit Testing Techniques
* Best coding practices
  + Course: C# Developers: Double Your Coding Speed
  + Course: Clean Coding and Refactoring
  + Clean Code by Robin C. Martin
  + [https://stackify.com/solid-design-principles](https://stackify.com/solid-design-principles/)

# Design Patterns [Course: The Ultimate Design Patterns]

* Memento Pattern
* State Pattern
* Iterator Pattern
* Strategy Pattern
* Template Method Pattern
* Command Pattern
* Observer Pattern
* Mediator Pattern
* Chain of Responsibility
* Visitor Pattern
* Composite Pattern
* Adapter Pattern
* Decorator Pattern
* Facade Pattern
* Flyweight Pattern
* Bridge Pattern
* Proxy Pattern
* Repository pattern

# WinForms

* Creation of Sample Application
* Placement of controls (Panels, buttons, radio, check buttons, date time, FLP, TLP)
* Layouting of controls
* Explore alignment, docking, anchoring and other control properties
* Events binding with a control
* Grids
* Sample exercises for practice
* Exploring DevExpress

# SQL Server [Course: Complete SQL Mastery]

* Introduction
* Basic Queries
* Retrieving Data from a Single and Multiple Tables
* Inserting, Updating, and Deleting Data
* Summarizing Data
* Writing Complex Query
* Views
* Stored Procedures
* Transactions and Concurrency
* Data Types
* Designing Databases
* Indexing for High Performance

# Entity Framework [Course: Entity Framework 6 in Depth]

* Building a Model Using Code-first Workflow
* Overriding Code-First Conventions
* Querying Data
* Loading Related Objects
* Changing Data
* Using Entity Framework in Applications

# NHibernate

* Setup and basic usage pattern
* Exploring Query Methods and Syntaxes
* Insert, Update, Delete Semantics
* Modeling Relationships in NHibernate
* Advanced Querying of child collections and views
* Querying Relationships and Best pattern for ISession Lifecycle
* Many to Many, View & Components
* Techniques of Effective Database-Driven Modeling
* Techniques of Effective Object-Driven Schemas
* SProcs, Interceptors, and NHibernate Query Analyzer
* Techniques for Modeling Inheritance in the database
* Working without a session

# WebApi

* RESTful Convention
* Building an API
* Testing an API
* Postman
* Data Transfer Objects
* AutoFac (Dependency Injection)
* Auto Mapper
* Repository pattern
* Builders and Workers concept
* Using Camel Notation
* IHTTPActionResult

# Smart Client

* [Composite Application Block](https://richnewman.wordpress.com/2007/08/05/dependency-injection-and-the-composite-application-block-introduction-to-cabscsf-part-5/)
* WorkItemController
* ControlledWorkItem
* [ModuleController](https://richnewman.wordpress.com/2007/10/27/introduction-to-smartparts-and-workspaces-introduction-to-cabscsf-part-15/)
* [Presenter](https://richnewman.wordpress.com/2007/11/24/workspace-types-introduction-to-the-cabscsf-part-17/)
* Outlining the Development Process
* Application Shell-Service and Infrastructure Service Design
* Workspace vs. UIExtensionSite
* Infrastructure Services
* Identifying WorkItems
* [Sub-WorkItem or WorkItem](https://richnewman.wordpress.com/2008/02/25/model-view-presenter-why-we-need-it-and-the-basic-pattern-introduction-to-cabscsf-part-23/)
* Business Entity–Driven Strategy
* Packaging WorkItems into Modules
* **Our Application Architecture**

# Git [Course: The Ultimate Git Course]

* Overview on source control
* Creating Snapshots
* Browsing History
* Branching
* Collaboration
* Rewriting History

# Azure DevOps

* Agile Planning and lean project management
* Boards
  + Work Items
  + Backlog
  + Sprints and Scrum
  + Queries and search
* Repositories
  + Branches
  + Pull Requests

# Elastic Stack (ELK)

* Document Management and Analyzers
* Defining the Index Structure
* Querying Elasticsearch
* Creating Kibana Visualizations

# Tools

* Power Commands for Visual Studio
* CodeMaid
* AutoMapper
* Resharper
* ApexSQL Complete
* WinMerge
* GrepWin
* Search Everything
* Viasfora
* LINQPAD
* PostMan

# Final Project

* All the learnt concepts will be utilized here

Courses available at [www.codewithmosh.com](http://www.codewithmosh.com)

NHibernate, WCF, WebAPI, Elastic Stack courses are only available locally.