# Agile Software Development Course

## This course is a unique collection of latest software development methods, includes **Object Oriented Design(OOD)**, **Design Pattern**, **Agile methods** with a detailed description of a complete software design for reusable programs in C++. Using a practical, problem solving approach, it shows how to develop an object-oriented application from the early stages of analysis, through the low level design and into the implementation.

**This course covers** : **principles of class design**, **analysis and design**, **patterns and paradigm**, **explains the principles of OOD**, one by one and then demonstrates them with numerous examples, completely worked-through designs, and case studies. Covers traps, pitfalls, and work around in the application of C++ and OOD and then shows how Agile methods can be used. Discusses the methods for designing and developing big software in detail.

# Course Outline:

## Source Control (1 hour):

* Git

## Agile Development (3 hour):

* Agile Practices
* Planning
* Testing
* Refactoring

## Agile Design (8 hour):

* SRP : The Single Responsibility Principle
* OCP : The Open-Close Principle
* LSP : The Liskov Substitution Principle
* DIP : The Dependency-Inversion Principle
* ISP : The Interface-Segregation Principle

## Project1: The Payroll Case Study (8 hour)

* Command And Active Object
* Template Method And Strategy: Inheritance VS Delegation
* Facade And Mediator
* Singleton And Monostate
* Null Object
* Implement Project

## Packaging Project1 (4 hour):

* Continuous Delivery
* Continuous Integration
* Principles Of Package Design
* Factory

## Project2 (8 hour):

* Composite
* Observer-Backing Into A Pattern
* Abstract Server, Adapter, And Bridge
* Proxy And Stairway to Heaven : Managing Third Party APIs

## Project3 (8 hour):

* Visitor
* State
* The ETS Framework