SETC Training Corporate Logo

# Le développement .NET de A à Z (NFCOMPFR, 9 jours)

## Description

Ce cours certifié SETC introduit les programmeurs aux concepts essentiels de la programmation orientée objet et à son implémentation. Le cours débute par une introduction aux principes du développement itératif de logiciels tel que proposé par le processus unifié de Rational (RUP). On poursuit ensuite avec une introduction aux pratiques de l'analyse et du design orientés objet tel qu'ils s'appliquent pour les méthodologies itératives. L'étape d'analyse et design du cycle de vie du développement logiciel complétée, on peut alors entreprendre l'implémentation des principes de l'orienté objet dans un environnement .NET. Ceci inclus le développement de classes, l'héritage, le polymorphisme, les classes abstraites, les constructeurs et les destructeurs. Le cours couvre également l'exploration de concepts clés dans la programmation .NET tels que les collections, les génériques, les fichiers, les flux et la programmation sur plusieurs fils d'exécution (multi-thread programming). Le cours se termine avec un survol du développement d'applications Web avec ASP .NET et le développement de bases de données avec ADO .NET.

## Tarifs

- Tarification: $6,600/person
- Rabais de 10% lorsque vous inscrivez 3 personnes.

## Plan de cours

Le Framework .NET

- Les niveaux d'application : applications à 1, 2 et 3 niveaux
- Chaînes de caractères, globalisation et conversions
- Débogage de
- Applications distribuées : Technologie et infrastructure
- Applications Web : Technologie et infrastructure.
- Versions et capacités de la structure .NET
- La réponse .NET
- Assemblages et espaces de noms
- L'environnement Visual Studio .NET

Analyse et conception orientées objet

- Le processus unifié Rational
- La phase de conception
- La phase d'élaboration
- La phase de construction
- La phase de transition

Analyse orientée objet

- Le magasin de vidéos heureux
- Identification des objets
- Identification des associations
- Le modèle de domaine.
- L'ébauche du modèle de domaine de la boutique vidéo heureuse
- Ajout d'attributs
- Ce qui fait un bon logiciel
- Qu'est-ce qu'un objet ?
- Les avantages de la programmation orientée objet
- Réutilisation des objets
- Objets et classes

Identifier les objets : Analyse orientée objet
Recueillir les exigences de l'application
Cas d'utilisation

## Conception orientée objet

Passer de l'analyse à la conception
Notation des classes UML
Généralisation
Agrégations
Classes de collection
Le diagramme de classe final
Le Rup et le Ood

## Création d'objets avec VB .NET

Un codage correct
Utilisation des classes
Création de classes
Propriétés des classes
Méthodes des objets
Constructeurs et destructeurs
Destructeurs
Données statiques
Un exemple complet d'objet

## Héritage et polymorphisme

Héritage
Polymorphisme
Fonctions et classes abstraites
Interfaces
Exemples courants d'héritage .NET

## Travailler avec des objets

Travailler avec des types de référence
Collections
Événements
Modèles de conception
Opérations sur les fichiers Io
Sérialisation
Programmation multithread

## Développement d'applications Web

Bases de l'Internet
Création de contrôles Web
Mise en cache du contenu Web
Tracer
Introduction aux services Web XML
Principes de base d'ASP.NET
Concepteur de formulaires Web
Principes de base des formulaires
Notions de base sur les contrôles
Création de formulaires
Contrôles de base
Contrôles de validation
Gestion des états

## Programmation de bases de données

Vue d'ensemble d'ADO .NET
Connexions
Données connectées : Déclarations SQL
Données connectées : Procédures stockées
Utilisation des transactions

Données déconnectées

Utilisation de la classe Dataset
Gestion des changements
Contraintes et relations
Mise à jour de la base de données
Liaison de données
Intégration XML

Configuration et déploiement

Configuration de l'application
Déploiement de l'application
Projets de configuration de Visual Studio .NET