.NET, développer avec des Design Patterns et des Frameworks

Cours Pratique de 5 jours - 35h
Réf : TAA - Prix 2024 : 2 970€ HT

Concevez des applications d'entreprise robustes et maintenables en utilisant les Design Pattern les plus éprouvés du génie logiciel. Appréhendez les différents frameworks de l'écosystème .NET et appuyez-vous sur leurs ressources/classes pour accélérer, fiabiliser et standardiser vos développements.

OBJECTIFS PÉDAGOGIQUES

À l'issue de la formation l'apprenant sera en mesure de :

Maîtriser l'injection de dépendances et le cycle de vie d'un objet

Mettre en œuvre le pattern command dans une architecture CQRS

Maîtriser l'accès aux données et l'application des transactions

Réalisation d'une API REST et d'une IHM avec AspNet Core

Un exercice "fil rouge" sera déroulé, chaque étape sera validée par des tests unitaires.

LE PROGRAMME

dernière mise à jour : 04/2023

1) Les problématiques du développement d'applications d'entreprise

- Les objectifs : productivité, scalabilité, évolutivité, testabilité.
- La séparation des responsabilités.
- L'application monolithique.
- L'architecture microservice.

Travaux pratiques : Prise en main de l'environnement de développement (Visual Studio).

2) Présentation de l'écosystème .NET

- Plusieurs langages : C#, VB.NET, F#.
- L'émergence de .NET Core, l'unification de .NET et de .NET Framework depuis .NET 5.
- Des frameworks : DependencyInjection, EntityFramework, ASP.NET, etc.

Travaux pratiques : Prise en main de l'environnement de développement.

3) Bonnes pratiques de conception et designs patterns

- Séparation des responsabilités avec facade.
- Injection de dépendances avec strategy.
- Interceptions avec proxy.
- Gestion du cycle de vie des composants avec singleton et prototype.
- Instanciation des composants de l'application avec factory.
- Implémenter une architecture orientée message avec command et observer.

Travaux pratiques : Mise en place de ces patterns avec Microsoft.Extensions.DependencyInjection.

4) Accès aux données et transactions

- Introduction aux différents types de bases de données (SQL, NoSQL).

PARTICIPANTS

Ce cours s'adresse aux développeurs.

PRÉREQUIS

Bonnes connaissances en programmation C# et du framework .NET. Expérience requise en développement d'applications .NET.

COMPÉTENCES DU FORMATEUR

Les experts qui animent la formation sont des spécialistes des matières abordées. Ils ont été validés par nos équipes pédagogiques tant sur le plan des connaissances métiers que sur celui de la pédagogie, et ce pour chaque cours qu'ils enseignent. Ils ont au minimum cinq à dix années d'expérience dans leur domaine et occupent ou ont occupé des postes à responsabilité en entreprise.

MODALITÉS D'ÉVALUATION

Le formateur évalue la progression pédagogique du participant tout au long de la formation au moyen de QCM, mises en situation, travaux pratiques…
Le participant complète également un test de positionnement en amont et en aval pour valider les compétences acquises.

MOYENS PÉDAGOGIQUES ET TECHNIQUES

• Les moyens pédagogiques et les méthodes d'enseignement utilisés sont principalement : aides audiovisuelles, documentation et support de cours, exercices pratiques d'application et corrigés des exercices pour les stages pratiques, études de cas ou présentation de cas réels pour les séminaires de formation.
• À l'issue de chaque stage ou séminaire, ORSYS fournit aux participants un questionnaire d'évaluation du cours qui est ensuite analysé par nos équipes pédagogiques.
• Une feuille d'émargement par demi-journée de présence est fournie en fin de formation ainsi qu'une attestation de fin de formation si le stagiaire a bien assisté à la totalité de la session.

MODALITÉS ET DÉLAIS D'ACCÈS

L'inscription doit être finalisée 24 heures avant le début de la formation.

ACCESSIBILITÉ AUX PERSONNES HANDICAPÉES

Vous avez un besoin spécifique d'accessibilité ? Contactez Mme FOSSE, référente handicap, à l'adresse suivante psh-accueil@orsys.fr pour étudier au mieux votre demande et sa faisabilité.

- Les principes ACID et la gestion des transactions.
- Le pattern unit of work.
- Frameworks de persistance (EntityFrameworkCore, NHibernate). mini-ORM (Dapper) : avantages/inconvénients Best pratices.

Travaux pratiques : Accès aux données d'une base relationnelle depuis une application C#, application des transactions.

5) API REST avec AspNet Core

- Les principes de conception d'une API REST (URI, mediatype, HATEOAS).
- Les bases du protocole HTTP.
- La réalisation d'API REST avec ASP.NET MVC.
- La sécurité : authentification par token avec OpenID Connect.

Travaux pratiques : Développement d'une API REST pour exposer l'application développée précédemment.

6) IHM avec AspNet Core

- Rappel du pattern MVC.
- Les vues Razor : accès au modèle, internationalisation, gestion des exceptions.
- Authentification par formulaire, sécurisation des routes et des vues, protection contre les attaques CSRF.

Travaux pratiques : Développement d'une IHM pour exposer l'application développée précédemment.

7) Industrialisation des développements

- L'intégration continue.
- La livraison continue.
- Création d'une image OCI avec Docker.
- Le déploiement sur l'orchestrateur Kubernetes.

Travaux pratiques : Création d'une image docker et études des descripteurs de déploiements Kubernetes.

LES DATES

CLASSE À DISTANCE
2025 : 17 mars, 30 juin, 08 sept., 01 déc.

PARIS
2025 : 10 mars, 23 juin, 01 sept., 24 nov.