

## Formation ASP.NET Core 6

### MVC Core avec C# 10 et Visual Studio 2022

S'inscrire

Durée 4 jours ( 28 heures )

#### Présentation

Avec .NET Core : Formez-vous sur LA toute dernière technologie Web de Microsoft cross-plateforme !

[ASP.NET Core 6](#) est sont la toutes dernières technologies open-source et multiplateformes proposées par Microsoft et soutenue par une importante communauté. Il fut récemment annoncé dans la [.NET Conf 2021](#) avec son lot de nouveautés et de prise en charge du support des toutes dernières technologies.

[Dotnet Core](#), est à la fois modulaire, cross-plateforme et pérenne dans le temps; cette technologie vous permettra de mettre en place des solutions robustes, maintenables et innovantes. Moderne, ce nouveau framework avec des performances incroyables, vous permettra d'appréhender [Docker et WebAssembly](#). (Retrouvez nos autres formations dédiées sur le sujet !) Cette formation a pour but de vous permettre d'appréhender les concepts de développement de site web et le développement d'application multiplateforme à l'aide de [.NET Core 6](#) et de vous présenter [toutes ses nouveautés](#). Nous aborderons les nouveautés, les principes fondamentaux et avancés de ce nouveau framework ainsi que les bonnes pratiques de développement, mais également la gestion de la migration des anciennes versions vers cette nouvelle mouture.

À l'issue de cette formation, à travers notre mise en pratique sous Visual Studio 2022 (17.0) et langage C# version 10.0, vous serez en mesure d'être autonomes afin d'amorcer sereinement le développement d'une nouvelle application ASP.NET MVC Core ou bien d'être en mesure de migrer une application existante vers ASP.NET MVC Core.

Dans cette formation, comme dans toutes nos formations que nous vous proposons nous utiliserons la [dernière version stable](#) ([ASP.NET Core 6](#) et [MVC Core 3](#) avec le nouveau [langage C# 10](#)).

#### Objectifs

- Maîtriser le fonctionnement de .NET Core
- Découvrir les fonctionnalités du framework ASP.NET Core et MVC Core
- Savoir développer une nouvelle application Web résiliente
- Savoir concevoir une application ASP.NET en utilisant les bonnes pratiques de développement
- Connaître les mécanismes avancés du framework et s'initier à Entity Framework Core (EF Core)
- Être capable de migrer un projet ancien ASP.NET MVC vers le nouveau ASP.NET MVC Core

#### Public visé

## Pré-requis

Connaissance en programmation orientée objet, dans l'idéal en environnement .NET.

## Programme de Notre Formation .NET Core 6 et MVC Core

### Jour 1 : Kick-off ASP.NET Core 3 & 5 & 6

#### Introduction

- Historique, contexte et objectifs
- Problématique actuelle du .NET Framework
- Qu'est-ce qu'ASP.NET Core ?
- Avantages et inconvénients d'ASP.NET Core
- Contraintes, quand choisir ASP.NET CORE 3 vs ASP.NET Framework
- Présentation rapide des nouveautés du langage C# 8 et environnements d'exploitation
- Support & Cycle de vie : [LTS vs Current Lifecycle](#)
- Les nouveautés de la version 2.1 & 2.2 & 3.0 & 3.1 & 5 & la future version 6
- Comment développer en mode cross-plateforme ?

#### Mise en place environnement de développement

- Travaux pratiques : Premier projet
  - Récupération des packages
  - Création en ligne de commande
  - Édition et debugging sous VS Code
  - Projet Visual Studio
- Éléments constitutifs d'un projet
- Retour sur d'importantes briques conceptuelles :
  - MVC (Modèle-vue-contrôleur)
  - Packages NuGet
  - Injection de dépendances
- Travaux pratiques : Mise en place des concepts du jour

### Jour 2 : Approfondissement

- Vision globale du cycle de développement d'un nouveau projet ASP.NET Core
- Accélérateurs Visual Studio (Scaffolding, etc.)
- Test de non régression, debuggig et performance
- View, Model, Pages, Razor & Tags helpers
- Travaux pratiques

- Routes & Controllers
  - Comment déclarer ses routes
    - Templates de routage : syntaxe et usage
    - Nouvelle notion d'area
    - Attributs
  - Filtres sur les requêtes
  - Url rewriter
- Travaux pratiques
- Retour sur l'injection de dépendances et la gestion des Options d'initialisation
  - Déclaration
  - Validation
- Travaux pratiques

## Jour 3 : Notions avancées

- Entity Framework (EF Core)
- Travaux pratiques
- Web API
- Authentification/Authorization/Identity Management
- Sécurité CORS (Cross Origin Resource Sharing)
- Récentes problématiques dues aux "breaking evolutions" de Google
  - Sécurité renforcée
  - Samesite cookie
- Travaux pratiques
- Caching
- Localization
- Middleware
- Background tasks
- Travaux pratiques

## Jour 4 : synthèse des notions acquises

- Utilisation en environnement cloud
  - Azure
  - AWS
- Présentation de différents systèmes basés sur ASP.NET Core
  - Blazor
  - Orchard Core
- Intégration à d'autres projets Open Sources
  - Exemple de Swagger
- Retour sur la migration d'Asp.Net 4 vers Asp.Net Core (outils, trucs & astuces)
- Travaux pratiques : Synthèse des éléments appris sous forme d'un projet réaliste

## Modules Complémentaires (+2 jours - uniquement en intraentreprise sur demande)

### Appliquer des styles

- Mise en page CSS
- Responsive design

- Intégration de Bootstrap 4.4.1

## Tests unitaires : bonnes pratiques

- Mise en œuvre des tests unitaires
- Stratégie de gestion d'exceptions

## Contrôle d'accès : identité et autorisation

- Personnalisation du User Model avec Identity
- Restreindre l'accès
- Paramétrage de la sécurité : tokens, cookies, durée de session
- Utiliser des fournisseurs externes : OAuth (Google, Facebook, Microsoft) & OpenID
- Les nouvelles classes Principal et Identity qui sont basées dans l'esprit sur des claims
- Authentification par tokens JWT
- Construire sa propre API
- Sécuriser son API
- Journalisation (Logging)
- Application Web résiliente :
  - Statefull
  - Gestion des états et des contrôles d'accès sécurisés

## SignalR : Fonctionnalités web en temps réel

- Créer un projet ASP.NET Core qui héberge le serveur et le client [SignalR](#)
- Créer un hub SignalR
- Configurer le projet pour utiliser SignalR
- Créer le code client SignalR
- Exécuter une application haute fréquence
- Introduction à gRPC

## Déploiement et intégration

- Déployer sur un serveur IIS
- Déployer avec un service Git (VSFS, TFS, GitHub)
- Déployer vers le cloud Azure
- Déployer en utilisant Docker
- Gérer la transformation de la configuration

## Migrer son code legacy

- Migration des versions antérieures vers ASP.NET Core 3.x
- Migrer une application vers .NET Core
- Migrer de ASP.NET MVC 5 à ASP.NET MVC Core
- Migrer de ASP.NET Web API 2 à ASP.NET MVC Core

## Sociétés concernées

Cette formation s'adresse aux entreprises, petites ou grandes, souhaitant former ses équipes à une nouvelle technologie informatique avancée.

## Méthodes pédagogiques

Stage Pratique : 60% Pratique, 40% Théorie. Support de la formation distribué au format numérique à tous les participants.

## Organisation

Le cours alterne les apports théoriques du formateur soutenus par des exemples et des séances de réflexions, et de travail en groupe.

## Validation

À la fin de la session, un questionnaire à choix multiple permet de vérifier l'acquisition correcte des compétences.

## Sanction

Une attestation sera remise à chaque stagiaire qui aura suivi la totalité de la formation.