

ASP.NET MVC Core, développement d'application web

5 j (35 heures)

Ref : ASCORE

Public

Développeurs web, chefs de projet techniques...

Pré-requis

Niveau : Avoir une bonne maîtrise du langage C# et avoir des connaissances en programmation Web (HTML et JavaScript)

Techniques (formations en classe virtuelle) : Vous devez disposer d'un ordinateur connecté à internet, d'un micro et d'une caméra

Moyens pédagogiques

Modalité : Formation présentielle ou Formation distancielle (classe virtuelle) - Inter / Intra - Groupes de 4 à 12 stagiaires

Méthodes : Présentation des concepts, discussion technique, démonstrations, exercices et TP

Matériel :

Présentiel : Un poste informatique par stagiaire connecté à internet, à une imprimante en réseau et au réseau informatique,

Les salles sont équipées d'un tableau interactif ou d'un vidéoprojecteur et d'un paperboard

Distanciel : Aelion met à disposition de chaque stagiaire

- Un PC équipé des outils et logiciels nécessaires à la formation auquel vous accédez via un outil de prise en main à distance

- Un accès à un outil de classe virtuelle (Meet)

Support de formation : Un support de formation sera remis à chaque stagiaire en fin de formation : plateforme collaborative intégrant le code source des exercices réalisés en formation, webographie, mémos

Modalités de suivi et d'évaluation

Questionnaire d'évaluation des pré-requis, suivi des connaissances tout au long de la formation, Evaluation des acquis en fin de formation

Questionnaire d'évaluation de la satisfaction en fin de stage, feuille de présence émargée par demi-journée par les stagiaires et le formateur, Attestation de fin de formation

Action Collective ATLAS

AELION est sélectionné par l'OPCO ATLAS dans le cadre des Actions Collectives Nationales (ACN) DEVELOPPEMENT.

Formation 100 % pris en charge par votre OPCO ATLAS (sous réserve de fonds disponibles et de validation de votre projet par votre conseiller-ère ATLAS).

Le Framework opensource ASP.NET Core est une réécriture du Framework propriétaire ASP.Net de Microsoft. Il prend définitivement sa suite à l'occasion de la version 5 et perd à cette occasion son suffixe Core. Il s'est imposé comme une technologie phare pour développer des sites web et des web services utilisables sur l'ensemble des plateformes existantes. A l'issue de la formation, vous serez capable de développer des applications Web basées sur le framework ASP.Net MVC Core, en appliquant notamment le modèle de programmation MVC (Modèle-Vue-Contrôleur) et les services Web Windows Azure.

Objectifs

- Comprendre les principes d'ASP Net MVC
- Acquérir les compétences pour créer des applications Web avec Visual Studio
- Comprendre comment mettre en oeuvre une méthode MVC pour développer en équipe
- Ecrire un service Web REST et l'appeler à partir d'une application MVC
- Tester et déboguer des applications Web ASP .NET Core
- Sécuriser une application web
- Déployer une application web

Programme détaillé

COMPRENDRE LES PRINCIPES D'ASP NET MVC

- Plateforme propriétaire.net
- Plateforme opensource .NET Core
- Framework ASP.NET Core
- Historique croisée des versions et des noms
- Langage C#
- Principes du MVC
- Principes Rest
- Protocole http

ACQUERIR LES COMPETENCES POUR CREER DES APPLICATIONS WEB AVEC VISUAL STUDIO

- Outils de développement
- Configuration d'un projet
- Structure d'un projet
- Gestion des dépendances et des tâches

COMPRENDRE COMMENT METTRE EN ŒUVRE UNE METHODE MVC POUR DEVELOPPER EN EQUIPE (1/3)

- ORM ENTITY FRAMEWORK CORE
- Configuration d'une base de données InMemory ou physique.
- Paradigmes DataFirst, CodeFirst, ModelFirst
- Mappage des entités
- Mappage des associations
- Gestion des changements
- Gestion des transactions
- Linq To Entities
- Linq To Objects

COMPRENDRE COMMENT METTRE EN œUVRE UNE METHODE MVC POUR DEVELOPPER EN EQUIPE (2/3)

Controleur MVC
Injection de dépendance
Actions et résultats d'action
Vue simple
HTML
CSS, LESS, SASS
JavaScript, JQuery
Bootstrap

COMPRENDRE COMMENT METTRE EN œUVRE UNE METHODE MVC POUR DEVELOPPER EN EQUIPE (3/3)

Vues, vues partielles, vues modèles
Vues composants
Syntaxe Razor
Conventions et ViewStart
Zones, vues partielles et layouts
Formulaire et Tag Helpers
Partage de données ViewData/ViewBag/TmpData

ÉCRIRE UN SERVICE WEB REST ET L'APPELER A PARTIR D'UNE APPLICATION MVC (1/2)

API REST
Contrôleur Rest
Synchrone vs asynchrone
Routage Middleware
CRUD avec les méthodes HTTP
Gestion des paramètres simples
Convertisseur JSON

ÉCRIRE UN SERVICE WEB REST ET L'APPELER A PARTIR D'UNE APPLICATION MVC (2/2)

Appel d'une API REST par un controleur MVC
Gestion des routes et de la méthodes HTTP
Passage de paramètres simples
Conversions des données JSON
Appel d'une API Rest depuis un Client Riche

TESTER ET DEBOGUER DES APPLICATIONS WEB ASP .NET CORE (1/2)

Gestion des erreurs
Codes d'erreurs

- Pages d'erreurs
- Contraintes et validation
- Les filtres d'exception
- Journalisation et niveaux de logs

TESTER ET DEBOGUER DES APPLICATIONS WEB ASP .NET CORE (2/2)

- Deboguer une application WEB
- Approche TDD et BDD
- Tests unitaires
- Test d'intégration
- Injection de dépendance
- Assertions et messages
- Tests paramétrés

SECURISER UNE APPLICATION WEB

- Configuration d'ASP IDENTITY
- Niveaux de verbosité d'authentification
- Personnalisation du User Model
- Construire et sécuriser sa propre API
- Sécuriser l'AIP avec les tokens JWT
- OpenID et OAuth 2 pour les fournisseurs externes (Google, Microsoft, ...)
- Configurer le Cross-Origin pour un Client Riche

DEPLOYER UNE APPLICATION WEB

- Deployer sur un serveur IIS
- Déployer vers le cloud Azure
- Déployer en utilisant Docker
- Azure App Services
- Compilation des pages et des vues à la publication
