











# Programme



**METHODES ET** MOYENS

**PEDAGOGIQUES** 

- Maitriser le langage C#, Asp.net, le Framework, Angular et Visual Studio.
- Etre capable en fin de session de concevoir une application client lourd, client léger. communiquant avec des bases de données, des flux XML et des services Web en utilisant la plateforme .Net
- Acquérir le savoir être du consultant

### Méthodes pédagogiques. Pour l'ensemble des stagiaires, le cours intégrera les suivantes

- Alternance d'exercices, cas pratiques, QCM et de notions théoriques, Projet Fil Rouge
- Evaluations

### Moyens pédagogiques

- AJC met à la disposition de chaque stagiaire un accès à notre plateforme à distance ainsi qu'éventuellement les logiciels utiles dans le cadre de chaque module
- Les supports de cours seront remis via notre la plate-forme de téléchargement Quest et/ou AJC Classroom

### Informations concernant les classes virtuelles

- Pour les formations en classe virtuelle, avec @JC CLASSROOM, vous profiterez des mêmes possibilités et interactions avec votre formateur que lors d'une formation présentielle : votre formation se déroulera en connexion continue 7h/7.
- Vous pourrez échanger directement avec le formateur et l'équipe pédagogique à travers notre système de visioconférence, mais aussi grâce aux forums et chats présents dans @JC CLASSROOM.
- Votre formateur sera à même de vérifier l'avancement de votre travail et de vous évaluer à l'aide d'exercices et de cas pratiques. Cela lui permettra de vous apporter un suivi pédagogique et des conseils personnalisés pendant toute la durée de la formation.
- Notre équipe technique vous enverra les modalités de connexion (accès, identifiants, dates, heures et numéro de la hotline) par mail dès votre inscription.
- Si vous rencontrez un problème de connexion, vous pourrez joindre à tout moment (avant ou même pendant la formation) notre hotline assistance technique au 01 82 83 72 41 ou par mail (hotline@ajc-formation.fr)



 Des notions d'algorithmie seraient un plus



• Scientifique ou toute personne en reconversion métier



Développeur C# Back-End, Développeur C# Front-End, Développeur FullStack, Développeur .NET, Développeur ASP.NET, ...



Présentiel et/ou Distanciel



Attestation de formation

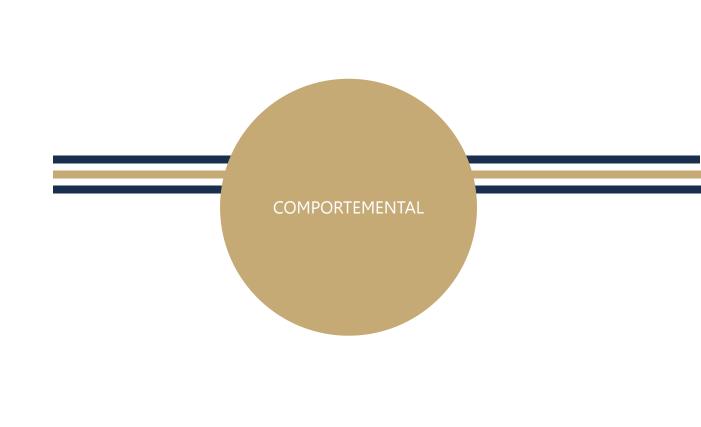
AJC FORMATION - 6 rue Rougemont 75009 - 01 81 51 64 85 - formonsnous@ajc-formation.tr S.A.R.L. au capital de 10 000 € - SIRET : 50096702100030 N° de déclaration d'activité 117542701 75 (ce numéro ne vaut pas agrément de l'Etat)

# Programme - Contenu pédagogique

COMPORTEMENTAL	RÔLE ET COMPORTEMENT DU CONSULTANT OBJECTIF « QUALITÉ » DE LA MISSION	2 jours
FONDAMENTAUX ET METHODES	ALGORITHMIE	3 jours
	OBJET UML	2 jours
	XML	2 jours
BASE DE DONNEES	SERVER SQL : TRANSACT SQL ET EXPLOITATION	3 jours
DEVELOPPEMENT .NET	LE DÉVELOPPEMENT .NET EN C# SOUS VISUAL STUDIO 2015/2017	5 jours
BASES DE DONNEES	L'ACCÈS AUX DONNÉES C# SOUS VISUAL STUDIO 2015/2017	5 jours
FONDAMENTAUX ET METHODES	DESIGN PATTERNS	2 jours
	AGILE SCRUM	2 jours
DEVELOPPEMENT .NET	INITIATION WEB AVEC HTML5, CSS3, JAVASCRIPT, RESPONSIVE, BOOTSTRAP	4 jours
	ANGULAR	5 jours
	CONCEPTION D'INTERFACES GRAPHIQUES WEB ASP .NET MVC SOUS VISUAL STUDIO 2015/2017	5 jours
	WEB SERVICES REST AVEC .NET	2 jours
	DÉVELOPPER DES APPLICATIONS WINDOWS AZURE ET DES WEB SERVICES	2 jours
	INTRODUCTION À ASP.NET CORE 2.1	2 jours
	MICROSOFT AZURE : FONDAMENTAUX	3 jours
	AZURE DEVOPS	4 jours
COMPORTEMENTAL	PRÉSENTER SES NOUVELLES COMPÉTENCES	1 jour
PROJET	PROJET FINAL & SOUTENANCE – C#	3 jours

**57 JOURS** 







# ROLE ET COMPORTEMENT DU CONSULTANT

### PROGRAMME DU MODULE

# Pourquoi s'intéresser aux comportements en tant que consultant?

- Qu'est-ce qu'un comportement ? Qu'est-ce qu'un rôle ?
- En quoi les comportements peuvent faire la différence ?
- Pourquoi choisit-on d'adopter un comportement? Le processus d'apprentissage d'un « savoir-être »

# Adopter la meilleure stratégie de coopération pour mieux travailler en équipe

- Comment agir pour des développer des relations positives et durables ?
- La théorie CRP

### Savoir communiquer et éviter les malentendus

- Pourquoi la communication passe-t-elle mal : les filtres, le cadre de référence ?
- Savoir utiliser l'écoute active : questionnement ouvert et reformulation
- Savoir convaincre: comment influencer positivement les échanges

# Comment faire évoluer ses comportements

- Qu'est-ce qui conditionne nos comportements?
- Sur quel levier agir pour ajouter des « cordes à son arc »

# Comprendre sa personnalité et mieux cerner celle des autres

 Savoir se situer et comprendre en quoi notre personnalité se traduit à travers

### nos comportements

 Situer les autres et comprendre leur mode de fonctionnement pour mieux coopérer

### Développer son intelligence émotionnelle pour modifier ses comportements

- Qu'est-ce que l'intelligence émotionnelle?
- En quoi notre QE est-il déterminant par rapport à nos comportements
- Apprendre à gérer son stress pour éviter les comportements inadaptés
  - Le stress : de quoi parle-t-on ?
  - Comment prévenir le stress et le gérer
    ?

### Appréhender le rôle des croyances et de l'éducation dans nos comportements

- Qu'est-ce qu'une croyance ?
- Pourquoi conditionnent-elles nos comportements?

# L'assertivité et l'empathie pour mieux travailler en équipe

- Qu'est-ce que l'assertivité ? Qu'est-ce que l'empathie ?
- La notion de respects des besoins et de gagnant-gagnant
- Savoir recadrer un comportement qui ne nous convient pas et renouer avec des relations positives

# 2 jours, 14 heures



PRESENTIEL et/ou DISTANCIEL

- La communication interne et externe au sein de l'entreprise
- Adapter et maîtriser les différents types de communication pour accroître son efficacité personnelle





# **ALGORITHMIE**

### PROGRAMME DU MODULE

Maîtriser les outils de l'algorithmique

(Schémas de programme, types et structures de données, modules)

Acquérir le s bases des méthodes de programmation structurée nécessaires à l'apprentissage de tout langage de programmation

Apprendre à raisonner sur un algorithme

Maitriser les notions de fichiers

Découvrir et mettre en œuvre la traduction d'un algorithme dans un langage de programmation

# 3 jours, 21 heures



PRESENTIEL et/ou DISTANCIEL

- Présenter les principes fondamentaux de la programmation et de l'algorithmique et expliquer les notions communes à tous les langages de programmation
- S'approprier les structures logiques et la démarche de résolution d'un problème de façon structurée et indépendante de toute contrainte matérielle ou logicielle
- Résoudre des problèmes plus ou moins complexes



# **OBJET UML**

### **Diagrammes d'interaction**

- Diagramme de Séquence : événement, message, synchrone, asynchrone axe temporel, cycle de vie etc.
- Diagramme de Communication :
- Idem diagramme séquence mais avec une représentation spatiale et non temporelle
- Diagramme de Timing
- Diagramme d'Activités : Actions, Activités, Transitions
- Couloirs d'activités etc.
- Diagramme d'Etat-Transitions :
- Comportements, états, événements, conditions, cycle de vie, automates etc.
- Améliorations des diagrammes existants

### Diagrammes d'implémentation

- Diagramme de Composants
- Diagramme de Déploiement : noeuds, artefacts etc.

### Nouveautés UML 2

- Diagramme de Structure
- Interne (Composite)
- Diagramme de Vue des
- Interactions

### Positionnement d'UML dans la démarche de développement

- Besoins et analyse
- Processus itératifs et incrémentaux
- Cycles de développement
- Niveaux d'abstractions
- Introduction aux templates de conception: Design Patterns
- Positionnement et utilisation d'UML dans les projets
- Méthodes existantes : RUP, XP etc.

Enjeux et avenir

# 2 jours, 14 heures



**PRESENTIEL** DISTANCIEL

### **OBJECTIFS**

 Apprendre à conceptualiser Objet

PROGRAMME DETAILLE

 Savoir lire et écrire des diagrammes UML



PROGRAMME DU MODULE

### Introduction

- "L'Objet": principaux concepts et avantages
- Historique et présentation d'UML
- Utilisation et avantages
- Les diagrammes UML : différentes vues d'une même réalité
- Présentation des outils utilisés sur le marché

### Objet

- Classes et objets
- Abstraction
- Instanciation
- Membres
- Encapsulation
- Visibilité
- Staticité
- Mémoire : pile, tas
- Création d'instances
- Point sur les différents langages objets
- Polymorphisme
- Associations, dépendance, réalisation
- Agrégation et composition
- Héritage : généralisation et spécialisation
- Classe abstraite, finale
- Interface
- Persistance: SGBDO, framework etc.
- Architectures distribuées : CORBA, J2EE, .NET

### Diagramme de cas d'utilisation

- Périmètre du système
- Notion d'acteurs
- Cas d'utilisation
- Relations

### Diagramme de classe

Représentation de tous les concepts



PROGRAMME DU MODULE

### $\mathsf{XML}$

### Du mainframe au Web

- Qu'est ce que le WWW?
- HTTP, Hypertexte et Hypermédia
- Applications Web, convention de nommage URL, navigateur et serveur Web
- Technologies (CGI, Perl, Activex,
- Applet, Com, Asp, Php, J2EE, .Net

### Langages HTML

- Structure d'un document HTML
- Notion d'élément et de balise HTML
- Eléments textes (paragraphes, titres...)
- Mise en page du texte, lisibilité, ergonomie
- Techniques de mise en page à l'aide de tableaux
- Insertion d'objets externes (Images, Applets, Contrôles ActiveX...)
- Quels formats d'images utiliser (GIF, JPEG, PNG)
- Liens hypertextes, création de menus
- Création de formulaires, insertion d'éléments (zones texte, boutons, listes...)
- Utilisation des frames

# Utilisation des « Cascading Style Sheets »

- Principe des CSS : séparation
- Contenu / contenant
- Structure d'une déclaration d'attributs de style
- Définition de classes
- Les bases d'une vision objet d'un document XML
- Sélecteurs spécifiques

Importation de déclarations CSS

### Introduction à XML

- Les bases d'XML
- Document XML bien formé
- Les éléments
- Les attributs
- Structuration d'un document XML

### La syntaxe des DTD

- Document XML valide
- Structuration des données
- Limites des DTD

### Les espaces de noms (namespaces)

- Règles d'écriture des espaces de noms
- Utilisation des espaces de noms
- Identifier les différents langages
- XML dans un document complexe

### XML Schéma

- Le langage XML Schéma
- La validation par les Schémas
- Modèle UML et XML Schéma
- Utilité des Schémas XML
- Création de Schémas XML
- Structuration des Schémas

# 2 jours, 14 heures



PRESENTIEL et/ou DISTANCIEL

- Apprendre à structurer les données
- Appliquer une grammaire pour organiser les flux de données
- Mettre en place des feuilles de style et de la transformation de données.



# XML (Suite)

### PROGRAMME DU MODULE

### XSL

- Le langage de programmation XSLT
- Cas d'utilisation de XSLT
- Utilisation de XPath avec XSLT
- Transformation de langages XML par XSLT
- Adresser et manipuler des éléments XML avec XPath
- Utilité de XPath
- Parcours des données avec XPath
- Requêtes XPath
- Les principes de base de XSLT
- XSLTvrai langage de prog
- Transformation et restructuration de documents XML
- Les extensions de XSLT
- Structuration de programmes XSLT
- Mise en page et formatage avancés
- Les éléments de base d'un document XSL
- Transformation XSL pdf

### **XQuery**

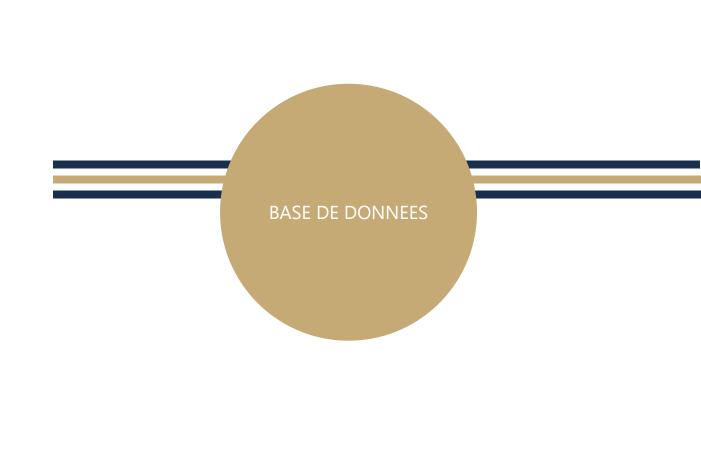
- Interroger un fichier de données
- XML comme une base relationnelle
- XPath 2, XSLT 2 et XQuery
- Utilité de XQuery

# 2 jours, 14 heures



PRESENTIEL et/ou DISTANCIEL

- Apprendre à structurer les données
- Appliquer une grammaire pour organiser les flux de données
- Mettre en place des feuilles de style et de la transformation de données.





# SERVER SQL:TRANSACT SQL ET EXPLOITATION

### PROGRAMME DU MODULE

### Présentation du langage

- Transact-SQL Langage de Programmation
- Transact-SQL Types d'instructions.
   Éléments de la syntaxe Transact-SQL
- Utilisation de la documentation en ligne de SQL Server

### Utilisation des outils de requête Transact-SQL

- Analyseur de requêtes SQL Server
- Utilisation de l'utilitaire osgl
- Exécution d'instructions
- Création et exécution de scripts

### Extractions de données

- Extraction de données à l'aide de l'instruction SELECT
- Filtrage des données
- Mise en forme des ensembles de résultats
- Traitement des requêtes

# Regroupement et synthèse de données

- Énumération des n premières (TOP) valeurs
- Utilisation de fonctions d'agrégation
- Présentation de la clause GROUP BY
- Génération de valeurs d'agrégation dans des ensembles de résultats
- Utilisation des clauses COMPUTE et COMPUTE BY
- Jointure de tables
- Utilisation d'alias pour les noms de table
- Combinaison de données provenant de plusieurs tables. jointure : inner join, left,

### join, right join

- Combinaison d'ensembles de résultats
- Exécution de requêtes sur plusieurs Tables

### Utilisation de sous-requêtes

- Présentation des sous-requêtes
- Utilisation d'une sous-requête en tant que table dérivée, en tant qu'expression, pour corréler des données
- Utilisation des clauses EXISTS et NOT EXISTS
- Utilisation de sous-requêtes

### Modification de données

- Utilisation de transactions
- Insertion et suppression de données
- Mise à jour et modification de données

### Définition de données

- Création, Modification, Suppression de tables
- Présentation, avantages des vues
- Création, Modification, Suppression de vues
- Création, Modification, Suppression d' index
- Création, Modification, Suppression de contraintes

### **Procédural**

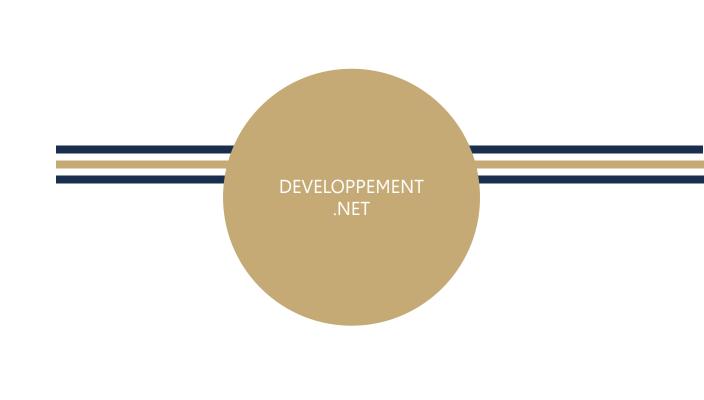
 Présentation des procédures stockées, des déclencheurs et des fonctions définies par l'utilisateur

# 3 jours, 21 heures



PRESENTIEL et/ou DISTANCIEL

- Manipuler les données sous SQL SERVER à l'aide du langage SQL
- Utilisation des procédures stockées et des déclencheurs





# LE DEVELOPPEMENT .NET EN C# SOUS VISUAL STUDIO 2015/2017

### PROGRAMME DU MODULE

# Introduction à C# et au .NET framework

- Introduction au .Net Framework
- Création de projets avec Visual Studio
- Écrire une application C#
- Construire une application graphique
- Documenter une application
- Exécuter et déboguer des applications avec Visual Studio

### Structure de programmation du C#

- Déclaration de variables et affectation de valeurs
- Utilisation d'expressions et d'opérateurs
- Création et utilisation des tableaux
- Instructions de décisions
- Instructions d'itérations

### Déclaration et appel de méthodes

- Définir et appeler des méthodes
- Spécification de paramètres optionnels et de paramètres de sortie
- Gestion d'exceptions

### Gestion des exceptions

- Soulever des exceptions
- Lire et écrire dans des fichiers
- Accéder au système de fichiers
- Lecture et écriture dans des fichiers en utilisant les flux

### Créer de nouveaux types de données

- Création et utilisation d'énumérations
- Création et utilisation de classes
- Création et utilisation de structures
- Comparaison des types références et

### types valeurs

### Encapsulation de données et de méthodes

- Contrôler la visibilité des membres
- Partager méthodes et données

# Héritage de classes et implémentation d'interfaces

- Utiliser l'héritage pour définir de nouveaux types références
- Définir et implémenter des interfaces
- Définir des classes abstraites

# Gestion de la durée de vie des objets et contrôler les ressources

- Introduction au Garbage Collection
- Gestion des ressources

# Encapsuler des données et définir des surcharges d'opérateurs

- Création et utilisation des propriétés
- Création et utilisation des indexeurs
- Surcharge d'opérateurs

### Découpage de méthodes et gestion d'évènements

- Déclaration et utilisation de délégué
- Utilisation des Lambda Expressions
- Gestion d'évènements

# 5 jours, 35 heures



PRESENTIEL et/ou DISTANCIEL

- Disposer d'une parfaite connaissance de la syntaxe C#
- Maîtriser la programmation orientée objet en C#
- Comprendre comment utiliser au mieux les fonctionnalités offertes par .Net Framework
- Savoir différencier les technologies d'accès aux données du .Net Framework



# LE DEVELOPPEMENT .NET EN C# SOUS VISUAL STUDIO 2015/2017 (Suite)

PROGRAMME DU MODULE

# Utilisation des collections et construction de types génériques

- Utilisation des collections
- Création et utilisation des types génériques
- Définir des interfaces génériques et comprendre la variance
- Utilisation de méthodes génériques et des délégués

# Construction et énumération de classes de collection personnalisées

- Implémenter une classe de collection personnalisée
- Ajouter un énumérateur à une classe de collection personnalisée

# Utilisation de LINQ pour interroger les données

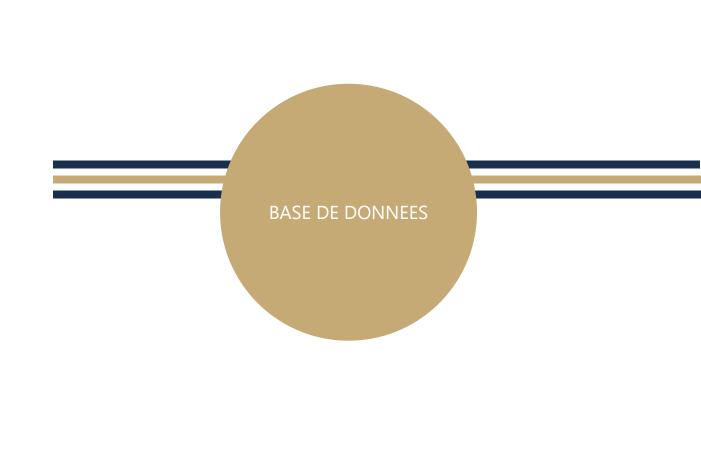
- Utilisation des méthodes d'extension LINQ et des opérateurs de requête
- Construction de requêtes et d'expressions LINQ dynamiques

# 5 jours, 35 heures



PRESENTIEL et/ou DISTANCIEL

- Disposer d'une parfaite connaissance de la syntaxe C#
- Maîtriser la programmation orientée objet en C#
- Comprendre comment utiliser au mieux les fonctionnalités offertes par .Net Framework
- Savoir différencier les technologies d'accès aux données du .Net Framework





# L'ACCES AUX DONNEES EN C# SOUS VISUAL STUDIO 2015/2017

### PROGRAMME DU MODULE

# Architecture des applications liées aux données

- Architecture Client-Serveur
- Architecture n-Tiers
- Architecture orientée services
- Technologies d'accès aux données .Net

### **ADO .NET**

- System.Data et fournisseurs d'accès aux données
- Architecture connectée et DbDataReader
- Architecture déconnectée et DbDataAdapter
- DataSet: typage, sérialisation

### **Entity Framework**

- Avantages d'Entity Framework (EF)
- Fichier de mapping et concepteur visuel
- Les approches du mapping : "database first", "model first" et "code first"
- Génération du schéma
- Gestion des EntityKey
- Gestion des relations
- Gestion de l'héritage
- Opérations CRUD et transactions
- Proxies et détection des modifications
- ObjectContext

### LINQ

- Méthodes d'extension, délégués et expressions Lambda
- LINQ To Objects
- LINQ To XML
- LINQ To Entities

### Mise à jour des données à travers Entity Framework

- Insertion, modification et suppression
- Scénario multi-utilisateurs

### Architecture orientée Services

- Les différents types de service
- Les services WCF
- Les services Cloud

### **Services REST**

- Avantages de l'approche REST
- Les services de données WCF
- Les services Web API

# 5 jours, 35 heures



PRESENTIEL et/ou DISTANCIEL

- Comprendre et savoir utiliser Entity Framework
- Savoir manipuler les données à l'aide de la syntaxe LINQ
- Être en mesure de concevoir des architectures orientées services pour l'accès aux données
- Disposer de l'ensemble des connaissances nécessaires au développement d'applications Net
- Être à même de tester, débugger et optimiser ses applications





# **DESIGN PATTERNS**

### PROGRAMME DU MODULE

### Introduction

### Historique

- Contexte d'utilisation
- Atouts et limites
- Le GOF (Gang Of For)

### **Anti Patterns**

- Erreurs courantes
- Couplage fort
- Effet spaguetti
- Manque de modularité
- Manque d'interface
- Copier-coller

### **Classification Patterns**

- Patterns créationnel
- Patterns structurel
- Patterns comportemental

### Patterns de création

- Fabrique abstraite
- Constructeur
- Méthode de fabrique
- prototype
- singleton.

### Patterns de structure

- Adaptateur
- Pont
- Composite
- Décorateur
- Façade
- Poids mouche
- Proxy

### Patterns de comportement

- Chaîne de responsabilité
- Commande
- Interpréteur
- Itérateur
- Médiateur
- Mémento
- Observateur
- Etat
- Stratégie
- Modèle de méthode
- Visiteur

### Exemples

- Singleton
- Factory
- AbstractFactory
- Composite
- Strategy
- FluentInterface
- Special Case

# 2 jours, 14 heures



PRESENTIEL et/ou DISTANCIEL

- Comprendre la valeur ajoutée des design patterns
- Mettre en œuvre les principes fondamentaux en conception objet
- Appliquer les patterns comportementaux
- Appliquer les patterns de construction
- Identifier les anti-pattern



# **AGILE SCRUM**

### PROGRAMME DU MODULE

### L'Agilité, c'est quoi?

- Les racines de l'Agilité (Manifeste Agile),
- Un cycle de vie différent

### La méthode Scrum

- Présentation de la méthode Scrum,
- L'équipe Scrum et ses rôles : le Product Owner, le ScrumMaster et l'équipe, le stakeholder,
- Les responsabilités de chaque rôle
- Les pratiques d'ingénierie agiles du logiciel

### Le Scrum Master en détail

- Un facilitateur et un garant de la méthode
- Responsabilités croisées entre Scrum Master et Product Owner
- Accompagnement de l'équipe de développement
- Accompagnement de l'organisation dans la mise en oeuvre de l'agilité
- Orientation solution / Servant Leadership

### Le Backlog de produit

- Identifier les Features et les Acteurs (Personas)
- Introduction aux «User Stories»
- Cycle de vie d'une User Story
- Rédiger les Users Stories à partir des Features
- Découper et détailler et prioriser les Users Stories dans le Backlog de produit
- Business Value et Effort quelles différences?

### Gestion du backlog et préparation

### du Sprint

- Techniques d'estimations des Users Stories
- Priorisation & plans (Périmètre fixé, Date fixée)
- Vélocité estimée & mesurée de l'équipe
- Définition de prêt et de fini

### Démarrage et déroulé du Sprint

- Planification de sprint
- Atelier: Session de planification de sprint: Création des tâches
- Le Daily Scrum : 3 questions quotidiennes
- Gestion des impondérables
- L'avancement du sprint
- Les tests dans un contexte Agile

### Fin du Sprint

- Livraison d'un incrément
- S'améliorer à intervalles réguliers
- La « sprint retrospective»

### Les outils du Scrum Master

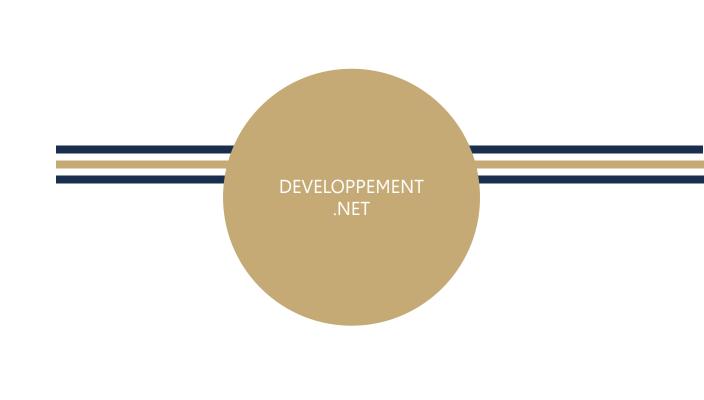
- Suivi du projet et gestion du temps
- Indicateurs et Key Performance Indicators (KPI)
- Suivre la qualité
- Comprendre les différents graphiques

# 2 jours, 14 heures



PRESENTIEL et/ou DISTANCIEL

- Maîtriser les composantes du rôle de Scrum Master
- Connaître les principes, méthodes et techniques Agiles
- Acquérir les compétences pour gérer et motiver les équipes
- Aider à la collaboration entre acteurs du projet
- Faciliter le travail de l'équipe et du PO
- Savoir animer les différentes cérémonies Scrum





# INITIATION WEB AVEC HTML5, CSS3, JAVASCRIPT, RESPONSIVE, BOOTSTRAP

### PROGRAMME DU MODULE

### Introduction protocole HTTP

- Requêtes et Réponse HTTP
- En tête HTTP
- Codes retour serveur
- Analyse avec F12

### Introduction langage HTML

- Contexte: web statique
- Balises HTML
- HTML et HTML 5
- Formulaire
- Audio et Vidéo
- Validation de champs

### **Introduction CSS**

- Contexte: ergonomie et habillage web statique
- Feuille de style externe, interne et inline
- Notion de cascade
- Notion de class
- Notion de id
- Notion de block
- Sizing et Positionning

### **Introduction Javascript**

- Contexte: web dynamique
- Spécification ECMA Script
- Les objets javascript
- Les objets du navigateur
- Validation des champs
- Gestion DOM: Document Object Model
- Gestion Evénements

# Introduction au Responsive Web Design

### • Ce qu'est le Responsive Web Design

- Les impacts dans la palette du web design
- Quels besoins le RWD vient-il satisfaire ?
- Activité séquentielle d'un device à l'autre
- Usage de devices différents selon le contexte, le moment de la journée
- Rapprochement mobile desktop: un seul développement

### Introduction BOOTSTRAP

- Notion de framework
- Augmenter la productivité et l'ergonomie des écrans web
- CSS et Javascript BOOTSTRAP
- Installation et mise en oeuvre

# 4 jours, 28 heures



PRESENTIEL et/ou DISTANCIEL

- S'initier aux technologies standards du Web
- Conception et mise en forme de pages web
- Mise en place de contenus dynamiques multicanaux



## **ANGULAR**

### PROGRAMME DU MODULE

### Rappels

- TypeScript
- Outillage

### **Présentation**

- Points négatifs AngularJS
- Points positifs Angular
- Points négatifs Angular
- Architecture

### Démarrer une application Angular

- Commencer un nouveau projet
- Angular-CLI
- Webpack

### Tests

- Jasmine
- Karma

### Composants

- Syntaxe des templates
- Propriétés
- Evénements
- Directives
- Composants

### **Composants Angular**

- ngStyle
- ngClass
- ngFor
- nglf
- ngSwitch
- ngNonBindable

### Injection de dépendances

Injecteurs

# 5 jours, 35 heures



PRESENTIEL et/ou DISTANCIEL

### **OBJECTIFS**

- Découvrir les fonctionnalités du framework Angular
- Savoir développer une nouvelle application Web
- Savoir concevoir une application Angular avec JavaScript et TypeScript
- Connaître les mécanismes avancés du framework

### Providers

- LES PIPES
- Création
- Utilisation

### **Communication avec une API Rest**

- Http
- Observables

### Router

- Configuration du Router
- Directive RouterOutlet
- Directive RouterLink
- Stratégies URL
- Paramètres URL
- Cycle de vie

### Gestion des formulaires

- Directives
- Validation
- SERVER-SIDE RENDERING
- Angular Universal



# CONCEPTION D'INTERFACES GRAPHIQUES WEB ASP.NET MVC SOUS VISUAL STUDIO 2015/2017

### PROGRAMME DU MODULE

### **Technologies Web .NET**

- Les modèles de développement Web
- Le choix du Framework : .NET Full vs .NET Core
- Les solutions d'hébergement : IIS, Azure, Kestrel
- Les outils : WebMatrix, Visual Studio

### Concevoir le modèle de données

- Création du modèle de données
- Connexion à la base de données par Entity Framework

### Concevoir les contrôleurs

- Contrôleurs et actions
- Filtres d'actions

### Concevoir les vues

- Syntaxe Razor
- HTML Helpers
- Réutilisation de code avec les vues partielles

### Tests unitaires et débogage

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

### Routage

- Le moteur de routage ASP .NET
- Implémentation d'un routage personnalisé

### **Appliquer les styles**

- Mise en page
- BootStrap

### Créer un site Web réactif

### Mise en cache des données

jQuery

### Sécurisation d'un site

- Authentification et autorisation
- Rôle et appartenance
- Windows Identity Framework

### Implémentation du Web API

- Architecture et mise en oeuvre
- Appel d'un service Web API côté client

### Déploiement d'un site

- Déploiement local
- Déploiement dans le Cloud

# 5 jours, 35 heures



PRESENTIEL et/ou DISTANCIEL

- Disposer des compétences nécessaires pour créer des applications Web avec Visual Studio 2015 ou 2017
- Être en mesure de développer des applications d'entreprises ASP .Net MVC en utilisant les technologies et outils du .NET Framework
- Savoir tester et déboguer des applications Web ASP .Net MVC
- Être capable d'écrire un service ASP .Net MVC Web API et l'appeler à partir du client
- Savoir intégrer aux applications Web ASP .Net des Frameworks JavaScript tels que BootStrap ou JQuery pour améliorer la qualité des interfaces clientes Web et mobiles



# WEB SERVICES REST AVEC .NET

Sécuriser un web service

- Niveaux de sécurité
- Gestion de l'authentification dans un web service
- Gestion des droits
- Signatures de messages
- Standards de sécurité disponibles

### PROGRAMME DU MODULE

### Comprendre le besoin

- Architecture orientée service (SOA) : composantes, technologies
- Présentation des Web Services (WS) : fonctionnement, intérêt, interopérabilité
- Technologies: protocole SOAP, Architecture REST
- Plates-formes à services web
- APIs .NET : WCF vs Web API

# Implémenter et interroger des services web REST

- Architecture REST: composantes, méthodes d'appel
- Gestion des erreurs
- API Web ASP.NET
- Développement de services RESTful
- Sérialisation Xml, JSON
- Envoi de paramètres, validations
- Type de réponse, format
- Déploiement d'un service RESTful
- Interrogation de web services REST

### Requêter un serveur avec OData

- OData: présentation du protocole, syntaxe
- Ecriture de requêtes avec OData et un client .NET
- Optimisation de volumes

### Déployer un service web

- Utilisation de IIS
- Gestion des logs : TraceListener vs Log4Net

# 2 jours, 14 heures



PRESENTIEL et/ou DISTANCIEL

### **OBJECTIFS**

 Savoir créer et consommer des services web REST avec .NET



# DÉVELOPPER DES APPLICATIONS WINDOWS AZURE ET DES WEB SERVICES

### PROGRAMME DU MODULE

# Vue d'ensemble des technologies de services et du cloud

- Éléments clés des applications distribuées
- Les données et les technologies d'accès aux données
- Technologies de Service
- Cloud Computing
- Exploration de l'application d'ateliers Blue Yonder Airlines

# Création et consommation de services Web API ASP.NET

- Que sont les services HTTP ?
- Création d'un service Web API ASP .NET
- Gestion des requêtes et réponses HTTP
- Hébergement et consommation de services Web API ASP .NET

# Extension et sécurisation des services Web API ASP .NET

- La gestion du pipeline Web API ASP .NET
- La gestion des réponses par l'API Web ASP .NET
- Création de services OData
- Implémentation de la sécurité dans les services Web API ASP .NET
- Injection de dépendances dans les contrôleurs

### Création de services WCF

- Avantages de la création de services avec WCF
- Création et mise en oeuvre d'un contrat
- Configuration et hébergement de

# 2 jours,14 heures



PRESENTIEL et/ou DISTANCIEL

### **OBJECTIFS**

- Comprendre les différentes formes de Cloud Computing et la place de Windows Azure dans l'offre actuelle
- Découvrir pourquoi les organisations veulent exécuter les applications dans le nuage Azure
- Se familiariser avec
   l'architecture de Windows Azure
- Explorer le SDK Azure et l'environnement de développement Dev Fabric
- Comprendre comment développer des applications pour Azure et pourquoi elles différent des applications .Net standards
- Explorer les capacités de stockage d'Azure
- Découvrir SQL Azure, la base de données relationnelle dans les nuages

### Hébergement de Services

Consommer des services WCF

services WCF

- Hébergement de services sur site
- Hébergement de services dans Windows Azure

### **Windows Azure Service Bus**

- Les relais de Windows Azure Service Bus
- Les files d'attente de Windows Azure Service Bus
- Les sujets de Windows Azure Service Bus

### Déploiement de Services

- Déploiement Web avec Visual Studio
- Création et déploiement des packages d'applications Web
- Outils ligne de commande pour les packages de déploiement Web
- Déploiement sur Windows Azure



# INTRODUCTION A ASP .NET CORE 2.1

L'objet HttpMessageResponse

**Utilisation avec Entity Framework** 

Intégration de Entity Framework Core

Migration du schéma de la base

Parcours rapide d'utilisation avec le

Framework client AngularJS

Exemple de mise en oeuvre

Objet Relationnal Mapping

Core 2.1

Mapping avancé

de données

Couche cliente

avec ASP .NET MVC

### PROGRAMME DU MODULE

### Rôle et architecture de .Net Core

- Les plateformes supportées
- Les outils disponibles
- Visual Studio (Windows)
- Visual Studio Code (Windows / Mac / Linux)
- Les outils CLI
- Tour d'horizon d'un produit encore incomplet
- Exemple de mise en oeuvre

# Architecture et mise en oeuvre de base

- Serveurs et nouveau modèle d'hébergement des applications Web
- Définition du pipeline des middlewares
- Le code de démarrage de l'application
- Traitement des fichiers statiques
- Structure d'une application Web et configuration
- Maintien d'état : données de requête HttpContext.ltems et middleware de session

### Routage et MVC

- Définition des routes et des contrôleurs
- Rappels de l'utilisation de Razor
- Nouveauté des vues : les Tag Helpers
- Tag Helpers et formulaires
- Layouts
- Injection de dépendance dans les vues

### Web API

- Evolutions importantes d'architecture de Web API
- Les controllers

# 2 jours, 14 heures



PRESENTIEL et/ou DISTANCIEL

- Comprendre l'architecture, les enjeux de la refonte et du portage du Framework .Net en sa version Core
- Comprendre les modifications que cela implique pour ASP.Net Core
- Mettre en oeuvre les nouveautés et déterminer si la migration vers cette nouvelle technologie est pertinente.



# MICROSOFT AZURE: FONDAMENTAUX

### PROGRAMME DU MODULE

**Azure** 

PowerShell Azure

Les outils de gestion Microsoft

■ SDK Azure et interface en ligne

de commande Azure Cross-Platform

### Présentation de Microsoft Azure

- Qu'est-ce que le Cloud Computing
- Qu'est-ce que Microsoft Azure
- Gérer Azure
- Gérer les abonnements et la facturation

### Sites Web et services Cloud

- Créer et configurer des sites Web
- Déployer et surveiller des sites Web
- Créer et déployer des services Cloud

### **Machines virtuelles**

- Créer et configurer des machines virtuelles
- Configurer des disques

### Réseaux virtuels

- Introduction au réseaux virtuels
- Création d'un réseau virtuel
- Implémentation

### Stockage dans le Cloud

- Comprendre ce qu'est le stockage dans le Cloud
- Création et gestion du stockage

### Les bases de données Microsoft Azure

- Comprendre les options de déploiement des bases de données relationnelles
- Créer et connecter des bases de données SOL

### **Active Directory Azure**

- Gérer les objets AD Azure
- Gérer l'authentification

# 3 jours, 21 heures



PRESENTIEL et/ou DISTANCIEL

- Décrire le Cloud Computing et les concepts de Microsoft Azure
- Créer et gérer des sites Web Microsoft Azure
- Créer et configurer des machines virtuelles dans Microsoft Azure
- Créer un réseau virtuel Microsoft Azure
- Configurer le stockage Microsoft Azure
- Déployer les bases de données dans Microsoft Azure
- Configurer Microsoft Azure Active Directory.



PROGRAMME DU MODULE

## **AZURE DEVOPS**

### Présentation générale

- DevOpset gestion du cycle de vie des applications (Application LifecycleManagement)
- Azure Boards
- Azure Pipelines
- Azure Repos
- Azure Artifacts
- Politique de licence

### Contrôle de versions

- Vue d'ensemble du contrôle de versions
- TFVC vs Git
- Visual Studio Team Explorer
- TFVC –espaces de travail locaux
- Atelier 1.1 : contrôler le code-source avec TFVC
- Git –référentiel local
- Liens avec les éléments de travail
- Stratégies d'archivage
- Branchement et fusion
- Créer un projet Visual Studio de bases de données

### **Gestion des Builds**

- Vue d'ensemble DevOps
- Pools d'agents
- Définition des Builds
- Résultats des Builds
- Pipeline as Code
- ClassicUI vs Yaml
- Les Artefacts
- Variables et groupes de variables
- Dimensionnement de l'infrastructure des pools d'agents

Intégration continue

### **Tests Techniques**

- Tests unitaires
- Présentation générale
- Mise en place des tests dans une build
- Test DrivenDevelopment
- Présentation et principes
- Présentation de librairies de Mock, d'assertion et de couverture de code
- BehaviorDrivenDevelopment
- Présentation de SpecFlow
- Tests IHM
- CodedUlou White

### Gestion des déploiements

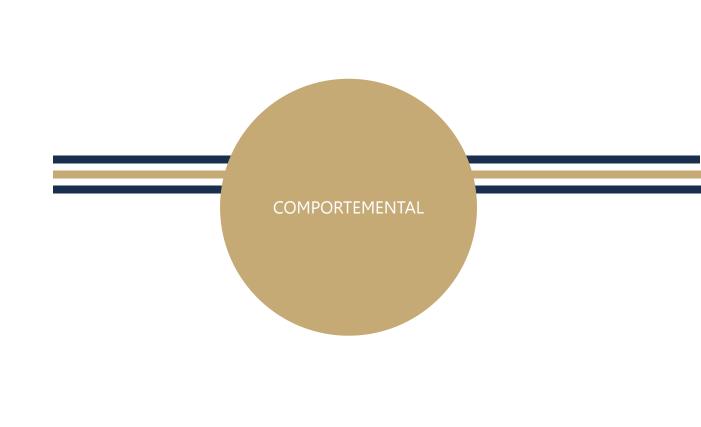
- CI/CD : Intégration Continue / Déploiement Continu
- Azure Release Pipelines
- Les groupes de déploiement
- Configuration d'un déploiement avec Yaml
- Bonnes pratiques Builds& Releases

# 4 jours, 28 heures



PRESENTIEL et/ou DISTANCIEL

- Gestion du Cycle de Vie des Applications (Application LifecycleManagement)
- Maîtrise des fonctionnalités de Azure Devops
- Gestion des référentiels de source : TFVC vs Git
- Builds& Intégration Continue
- Tests techniques
- Releases & Déploiement Continu





# PRESENTER SES NOUVELLES COMPETENCES

PROGRAMME DU MODULE

### Les bases de la communication

- Ecoute active
- Le questionnement
- Reformulation et feedback

# La communication verbale et non verbale

- Importance de la communication non verbale
- Savoir se présenter à l'oral
- Postures Attitudes Discours

### Les profils comportementaux

- Les 4 profils
- Auto évaluation
- Développer son adaptabilité relationnelle

### **Développer son Capital Talents**

- Définition d'un talent
- Talent vs points forts
- 5 stratégies pour gérer ses points faibles

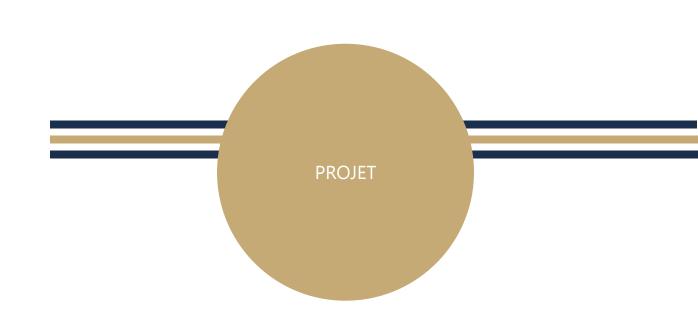
# 1 jour, 7 heures



PRESENTIEL et/ou DISTANCIEL

### **OBJECTIFS**

 Se présenter en entretien tout en mettant en valeur ses nouvelles compétences en les considérant acquises





# PROJET FINAL & SOUTENANCE C#

### PROGRAMME DU MODULE

### Déroulement du module

- Les stagiaires travaillent en toute autonomie, en binôme. Ils sont libres d'effectuer les choix adaptés, de développer les parties dont ils jugent avoir le plus besoin et d'apporter leurs propres solutions aux problèmes posés.
- Le formateur encadre les stagiaires par sa présence et répond aux questions. Il intervient pour épauler un binôme en difficulté ou pour faire le point à l'ensemble du groupe sur des notions non acquises. Il peut être amené à approfondir ou compléter certaines connaissances.

# 3 jours, 21 heures

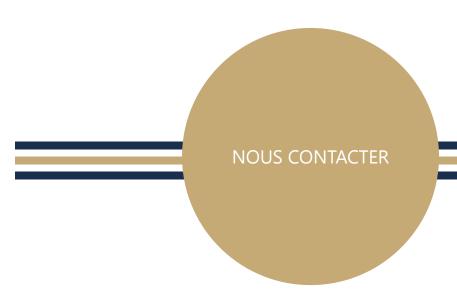


PRESENTIEL et/ou DISTANCIEL

### **OBJECTIFS**

 Mettre en application les acquis de la formation en complétant les mini projets réalisés dans tout le cursus





AJC FORMATION 01 81 51 64 85 formonsnous@ajc-formation.fr 6 rue ROUGEMONT 75009 PARIS



www.ajc-formation.fr www.ajc-classroom.fr

