

# Catalogue 2016 :

## Méthodologies

## Formation sur la Certification Scrum Master

Durée : 2 jours

Maîtrise des concepts fondamentaux liés à la planification, l'initiation et la conduite d'un projet Scrum

Toute personne intervenant au sein d'une équipe projet et souhaitant développer une méthodologie Scrum

### **Introduction à Scrum**

Historique de l'agilité

Principes fondamentaux de Scrum

### **Les 3 piliers de la théorie**

Transparence

Introspection

Adaptation

### **La science de Scrum**

L'optimisation de la production

### **Le rôle du Scrum Master**

Travail en équipe

Gestion de la connaissance

Techniques de motivation

Amélioration de la performance des équipes et des individus

### **Scrum dans la pratique**

Un outil interactif pour favoriser l'adoption de Scrum par les équipes : Scrum Game

### **Collaborer avec les équipes Scrum**

Atelier : réunion quotidienne d'enfer

La définition du mot "Terminé" (Done)

### **Création et gestion du "Product Backlog"**

Atelier : le Product Backlog

Atelier : le Planning Poker

### **Ecrire les User Stories et leurs tests de recette**

Atelier : création d'un projet

Atelier : remanier les User Stories

gestion de la Valeur Acquise (EVM Agile)

Minimum Marketable Feature (MMF)

### **Planification et estimation**

Techniques d'estimation

Priorisation et planification

Mise en pratique avec le Scrum Game de l'enfer

### **Rétrospective**

Atelier : L'art de la rétrospective

### **Conclusion**

## Formation sur la conception orientée objet

Durée : 3 jours

Maîtrise des concepts fondamentaux liés à l'apprentissage du développement Objet, à la notion d'Objet

Toute personne intervenant en tant que développeur, analyste programmeur, chef de projet

### La chaîne globale

Nécessité de paliers entre la pensée humaine et les séquences binaires 01010001...

Les différentes phases : sources, compilation, binaire

Le cas particulier d'un interpréteur

Ne pas tout réécrire (bibliothèques, éditions de liens, et exécutables)

### Algorithme

Les "atomes" pour s'exprimer

Apprendre à formuler un traitement (rien ne s'invente)

Utilisation d'un pseudo langage

Exemples d'algorithme bien conçu, "mal" conçu, et ... faux !

### Variables et types de données

Pourquoi typer les variables ?

Exemples de type (entier, réel, caractères...)

Exemples de problèmes liés aux types

Les opérateurs disponibles (+, -, /, \* / % ...)

Le confort des tableaux et des structures

### La "grammaire" d'un langage

Les blocs de programme (début ... fin)

Le test d'une expression (si ... alors ... sinon ...)

La boucle (tant que ...)

Les confort syntaxiques

### Un premier programme

Présentation de l'environnement de développement

Un premier programme simple en Java

Compilation et exécution

Avantage d'une interface de développement (IDE)

Présentation des acteurs majeurs pour les langages objet

### Structurer son programme

La nécessité d'utiliser des procédures ou des fonctions

Différences conceptuelles et syntaxiques entre les deux

Passer des arguments à une fonction (prototype, code retour)

Passage par valeur et par adresse dans une fonction

### Les bibliothèques

Ne pas réécrire ce qui existe déjà (librairies)

Écrire ses propres librairies

Comment les utiliser dans d'autres programmes ?

### L'importance de la présentation

Les bonnes habitudes pour anticiper les erreurs

Les commentaires utiles et efficaces

Les architectures de code (couches, séparation clients/serveur, etc.)

**L'approche Objet**

Rappel sur l'algorithme

Les objectifs du monde Objet, les classes et les objets

Travailler avec les attributs et les méthodes (fonctions de classes, constructeurs, destructeurs et accesseurs)

Comprendre et savoir utiliser efficacement l'encapsulation, l'instanciation

Traduction des concepts Objet en langage : les packages et les espaces de noms, les classes, les méthodes et leur visibilité, les attributs et leur visibilité, l'instanciation, l'appel de méthodes et la référence aux variables

Mise en pratique

**Analyse et conception**

Comprendre le besoin du client

Savoir imaginer une architecture logicielle adaptée

Imaginer les acteurs de l'application

**UML**

Spécification

Dans quel cas utiliser UML

Diagrammes de cas d'utilisation : que fait notre application ?

Diagrammes de classe : adapter ses connaissances objets aux diagrammes

Diagramme de séquence : modéliser les interactions de nos applications

Utilisation d'UML dans le monde réel

## Formation sur la gestion agile de projets avec SCRUM

Durée : 2 jours

Maîtrise des concepts fondamentaux liés à l'amélioration de la productivité des équipes projets avec la méthode Agile Scrum

Toute personne intervenant en tant que Chef de Projet, Directeur de Projet

### Introduction

Une approche agile de gestion de projet  
L'organisation projet traditionnelle  
Limites du cycle de vie en V  
Le manifeste Agile  
La place de Scrum dans le monde des méthodes Agiles

### Tour d'horizon

Naissance de Scrum  
Scrum en quelques mots  
Le cycle de vie de Scrum  
Les 3 Piliers de SCRUM  
Coût, délai, périmètre

### L'équipe

Le Scrum Master  
Le Product Owner  
Les développeurs  
L'environnement de travail

### Les Besoins : écriture du Product Backlog

Le Sprint 0  
Le Product Backlog  
Rédiger les User Stories et Epics  
Les erreurs fréquentes  
Les outils d'écriture d'User Stories

### Les tests pour les User Stories

La Pyramide de tests de Mike Cohn  
Les tests d'acceptation  
Les TDD et ATDD

### Priorisation du Product Backlog

La Priorisation  
Définir la Valeur Métier par le Priority Poker  
Priorisation par le modèle de Kano  
Priorisation à partir des Thèmes  
Priorisation par la Communication

### Estimation de la durée et planification d'un Sprint

Equipe, Product Backlog et tableau de Sprint Backlog  
La réunion de planification de Sprints  
Définition du but à atteindre et de la durée d'un Sprint  
Estimation des User Stories  
Définition des User Stories pour le prochain Sprint  
Le Découpage en tâches  
Les Users Stories techniques  
Les Defect Stories

**Planification des Releases**

Scrum et les Releases  
Planification d'une Release

**Vie d'un Sprint**

Check-list  
Scrum Meeting/Daily Scrum  
Sprint Review  
Rétrospective de Sprint

**Le suivi**

Planifier, itérer et assurer le suivi  
Estimer le Reste à Faire  
Connaître la valeur métier pour arrêter le projet  
La motivation de l'équipe

**La conduite du changement**

Comment mener le changement vers Scrum ?  
Etat des Lieux en France  
Stopper les idées reçues  
La résistance au changement  
Les 8 étapes selon John Kotter  
SWOT et SCRUM

**Contractualisation**

Réalité et Contradiction  
La Relation Client  
Les Appels d'Offre  
Le Plan Qualité Projet  
Mise en œuvre de la contractualisation  
Les différentes formes de contrats  
Exemple de contrat

## **Formation sur la gestion de projets avec Microsoft Project 2007**

Durée : 3 jours

Maîtrise des concepts fondamentaux liés aux fonctionnalités de Microsoft Project 2007

Toute personne intervenant en tant que Chef de Projet

### **Introduction**

Rappels sur la gestion de projets  
Présentation de Microsoft Project  
Manipulations particulières à Project  
Utilisation de l'aide  
Présentation des options

### **Création d'un plan de projet**

Saisie des données du projet  
Saisie du détail des tâches et de leurs durées  
Structuration du plan  
Définition des liaisons entre les différentes tâches  
Affectation des ressources et des coûts associés  
Résolution des problèmes de sur-utilisation  
Sauvegarde de la planification initiale  
Pilotage et suivi du projet  
Communication des informations aux différents intervenants  
Impression et sauvegarde du résultat final

### **Les éléments du projet**

Les tâches : champs utilisables, types de contraintes et familles de tâches  
La planification : objectifs, impacts, fractionnement des tâches  
Les structures hiérarchiques : objectifs et applications aux vues d'utilisation  
Le mode multi projets : enjeux, possibilités offertes par MS Project et la planification des projets insérés pour le suivi des éventuels sous-traitants  
Les calendriers : applications de nouveaux calendriers au projet  
Les ressources : affectation, tableau des disponibilités, accumulation des coûts, tableau d'utilisation et ressources matérielles, les bibliothèques de ressources  
Les coûts : heures supplémentaires, sur-utilisation et audit

### **Le pilotage et le suivi**

Audit des ressources : identification des ressources sur utilisées  
Le suivi de l'avancement du projet  
Les options de calcul affectant l'avancement  
Pilotage du projet par le travail  
Pilotage du projet par la durée  
Pilotage par la valeur acquise

### **Génération de rapports et exportation des données**

Identification des rapports clés  
Personnalisation des rapports  
Impression des rapports  
Import de données depuis une application externe  
Export de données vers d'autres applications

### **Démonstration de Project Server 2007 (EPM)**

Tour d'horizon des fonctionnalités offertes  
Interaction avec la suite Office  
Saisie d'informations par des interfaces Web  
Organisation de l'équipe projet autour d'EPM

## Formation sur la gestion de projets avec Microsoft Project 2010

Durée : 3 jours

Maîtrise des concepts fondamentaux liés à la méthodologie de planification et de suivi de projet

Toute personne intervenant en tant que Chef de Projet

### Introduction

La structure MSP Pro et MSP Serveur

Créer un projet

Gérer les affichages

Enregistrer un modèle

Modifier et créer des calendriers

Les options importantes pour le planning

### La création des projets

Créer des tâches par la durée

Créer des tâches par le travail

Les tâches récapitulatives

Les jalons

Les liens entre les tâches

Les tâches hamac

Les tâches répétitives

Affecter un calendrier à une tâche

Fractionner une tâche

Utiliser les échéances pour les dates butoirs

Utiliser les contraintes pour les tâches hors planning

### Les ressources

Créer les ressources (travail, matériel, coût)

Partager les ressources entre plusieurs projets

Modifier les détails des ressources (changement de coût, informations de compte Windows, etc.)

Gérer les calendriers des ressources (vacances, absences, etc.)

Gérer les coûts des ressources et les ressources de coûts

Affecter des ressources à temps plein

Calcul Durée=Travail/Nb de Ressources

Détecter et résoudre les situations de ressources sur-utilisées

Affecter des ressources supplémentaires

Faire un audit des ressources

Équilibrer et optimiser le projet manuellement

### Visualiser les informations clés des projets

Afficher le chemin critique du projet

Visualiser les coûts des tâches, des ressources et des projets

Saisir les heures supplémentaires

Quel impact des heures supplémentaires sur le planning et sur les coûts

Enregistrer le premier planning de référence

Visualiser le planning de référence

### Piloter l'avancement du projet

Les différentes approches pour saisir l'avancement de la réalisation du projet

Saisir l'avancement par la durée

Visualiser les dérapages

Saisir l'avancement par le travail

Visualiser l'utilisation des tâches et des ressources

Saisir plusieurs plannings de références après les recadrages des projets

**Personnaliser les données des projets**

Personnaliser les diagrammes

Insérer des commentaires et des graphiques dans les diagrammes

Personnaliser les champs avec des formules

Créer des tableaux de bord

Filtrer

Grouper

Trier

Personnaliser le « global.mpt », le modèle global pour MS Project Pro

**Travailler sur plusieurs projets**

Travailler en multi-projets

**Utiliser les données des projets hors de MS Project**

Créer des rapports depuis MS Project Pro

Copier et exporter le diagramme de Gantt

Exporter des données dans MS Excel

Créer des graphes de données projets dans MS Excel

Présenter les données des projets dans MS PowerPoint

## **Formation sur la gestion de projets avec Microsoft Project 2010/2013 (fonctionnalités avancées)**

Durée : 2 jours

Maîtrise des concepts fondamentaux liés à l'utilisation des fonctionnalités avancées

Toute personne intervenant en tant que Chef de Projet

### **Rappel des fondamentaux**

Définition du projet, programme et portefeuille

Tâches, ressources et livrables

Les jalons

Les phases

Le réseau PERT

Le GANTT

### **Elaborer son projet**

Rappel des différentes versions et limites MS Project

Focus sur MS Project server

Organiser le projet en lots de travaux (WBS)

Lister les tâches

Identifier les livrables (PBS)

Organiser les ressources (OBS)

Introduction au RACI

Organiser son projet avant de le mettre dans MS Project

### **La planification avec MS Project 2010/2013**

Rappel de l'interface et navigation

Le calendrier du projet

La date de début du projet

Description de la tâche

Focus sur les notions de planifications manuelles ou automatiques

Formatage du détail de la tâche

Focus sur les notions de durée et de travail (charges)

Liste des activités

Estimation de la durée des tâches

Notions d'antériorité

Les différents types de liens

Les marges (libre et totale)

Le chemin critique

### **Planifier les ressources sous MS Project**

Définition de la notion de ressources

Définition de la capacité

Equilibre charge capacité

Affectation des ressources

Surcapacité, nivellement et lissage

Focus sur le budget et l'allocation des coûts

Gestion des ressources dans un cadre multi-projets, programme ou portefeuille

**Piloter son ou ses projets avec MS Project**

Figier le planning prévisionnel  
Méthodes de suivi du projet  
Mesure de l'avancement du projet  
Les lignes isochrones  
Les différentes vues de suivi du projet  
Incidence d'une nouvelle planification  
Liens entres projets

**Tableaux de bords**

Le diagramme Temps-Temps  
Calcul de l'avancement global du projet  
La courbe en S  
Focus sur les notions de CBTP (Budget prévisionnel), CRTE (Coût réel du travail) et CBTE (Valeur acquise)  
Les rapports formatés  
Les rapports personnalisés

**La gestion multi-projet avec MS Project**

Utilisation d'une version professionnelle (possibilités et limites)  
Utilisation d'une version serveur, prérequis et fondamentaux à respecter  
Gestion des versions  
Gestion des droits (bonnes pratiques)  
Pilotage du projet  
Pilotage des projets (Multi-projets, programmes et portefeuilles)  
Focus sur les états d'avancement  
Les tableaux de bord  
Conclusion

## Formation sur la gestion des projets avec les méthodes agiles

Durée : 2 jours

Maîtrise des concepts fondamentaux liés aux différentes méthodes agiles disponibles, à l'amélioration de la productivité des équipes projets avec les méthodes agile

Toute personne intervenant en tant que Chef de Projet, analyste, Développeur

### Introduction

- Cartographie des méthodes agiles
- Extreme Programming
- Rational Unified Process
- Feature Driven Development
- Rapid Application Développement
- Dynamic systems development method
- SCRUM
- Le Manifeste Agile
- Quand utiliser Scrum
- Quelques Mythes sur Scrum
- Cadre et Théorie Scrum
- Scrum en Pratique

### Les Rôles SCRUM

- L'équipe Scrum
- Environnement Projet
- Product Owner
- SCRUM Master
- L'équipe de développement

### Les Artefacts Scrum

- Incréments
- Product Backlog
- Stories
- Product BurnUp Chart (hors Scrum)
- Product et Sprint BurnDown Charts

### Les évènements Scrum

- Nature des Evènements
- Planification de Sprint
- Planning Poker
- Daily Scrum
- Revue de Sprint
- Rétrospective de Sprint

## **Formation sur la gestion des projets avec Microsoft Project 2013**

Durée : 3 jours

Maîtrise des concepts fondamentaux liés à la méthodologie de planification et de suivi de projet

Toute personne intervenant en tant que Chef de Projet, membre d'une équipe Projet

### **Présentation**

Rappel sur quelques notions de méthodologie de projet

Présentation du logiciel

Présentation de la nouvelle interface 2013

Importer des données externes (Excel, SharePoint...)

### **Construction d'un planning prévisionnel**

Initialisation d'un projet

Création, modification et utilisation des calendriers

Définition des dates de début ou de fin

Tracer les séquences de tâches

Saisie et structuration des tâches : définition des durées, définition des liens, des contraintes

Surligner les liaisons entre tâches

Les Prédécesseurs pilotant

Les Successeurs entraînés

Les tâches récapitulatives

Les tâches répétitives

Définition du chemin critique

Afficher le chemin critique d'une séquence de tâches

Paramétrages du GANTT. Insérer / Masquer des colonnes

### **Enregistrement de la planification initiale**

Ensemble du projet

Par phase de projet

Planification intermédiaire

### **Les affichages.**

Utilisation des tâches

Utilisation des ressources

Graphe des ressources

Personnalisation

Création des affichages

Utilisation des affichages simples ou combinés

### **Suivi de projet par la durée**

Déroulement réel

« Avancer » un projet

Faire le suivi

Créer des dates de contraintes

Créer des échéances

Comparaison avec le planning initial

Les courbes d'avancement

Utilisation des diagrammes pour le suivi

Reporter une tâche

Fractionner une tâche

Afficher les marges totales

### **Les tables**

Changer de table  
Modifier une table  
Les colonnes « indicateurs »  
Les colonnes « textes »  
Formules de calculs

**Mise en forme du diagramme**

Echelle de temps  
Mise en forme des barres  
Styles des barres  
Le quadrillage

**Gestion des données**

Trier un planning  
Filtrer un planning  
Regrouper des données

**Impressions du planning, des calendriers**

Mise en page, en-tête/pied de page, légende  
Combiner affichage, table et impression  
Imprimer des rapports Project standards  
Imprimer des rapports graphiques  
Imprimer un rapport d'avancement

**Gestion des ressources**

Créations, paramétrage des ressources  
Calendriers des ressources  
Affectation des ressources  
Gestion du coût des ressources  
Partager des ressources communes à plusieurs projets  
Le planificateur d'équipe

**Les coûts**

Affectation des coûts fixes  
Etude et analyse des coûts

**Le Multi projets**

Créer un fichier de consolidations de projets  
Insérer un nouveau projet  
Lier des projets

**Imports/Exports**

Exporter vers Excel via le mappage  
Exporter vers Excel via les rapports

## **Formation sur la mise en place et l'animation d'un Project Management Office**

Durée : 2 jours

Maîtrise des concepts fondamentaux liés aux objectifs du PMO (optimisation du portefeuille des projets et succès des projets), au choix du niveau de PMO adapté aux besoins de l'entreprise

Toute personne intervenant en tant que Chef de Projet, Directeur de Projet, Directeur de PMO

### **Vendre en interne les bénéfices de la mise en place d'un PMO**

L'alignement stratégique des projets

L'optimisation de l'emploi des ressources

La maximisation du ROI pour l'entreprise

Le soutien à chacun des projets, pour apporter les résultats attendus dans les délais adaptés, au moindre coût

La définition du niveau du management de projet adapté à la maturité de l'entreprise

### **Mettre en place un PMO en 7 étapes**

Élaborer le CdC du projet de mise en œuvre du PMO

Choisir le type de PMO

Etablir la liste des services rendus sur un plan à deux ans : gouvernance (définition, exécution), pilotage du portefeuille de projets, mise sous contrôle des projets, arbitre des conflits entre projets et au sein des projets, développeur de compétences...

Définir la composition du PMO : rôles, responsabilités et compétences (dans les projets, dans le PMO, dans l'entreprise)

Etablir les processus clés du PMO : priorisation des projets, lancement d'un projet, allocation de ressources, validation aux jalons, sortie du portefeuille des projets

Créer le tableau de bord du PMO : définir les objectifs de contrôle, identifier les indicateurs pertinents en regard de ces objectifs, structurer le tableau de bord

Créer les premiers outils à mettre en place pour obtenir rapidement des gains de productivité visibles

### **Animer un PMO**

Définir le plan de montée en puissance sur le moyen terme

Mesurer la prestation apportée par le PMO et afficher les gains apportés à l'entreprise

Étendre les prestations du PMO par des expériences pilote

Faire de la formation et de l'accompagnement le fer de lance de la valeur perçue par l'entreprise

## Formation sur l'automatisation des tests logiciels

Durée : 3 jours

Maîtrise des concepts fondamentaux liés à l'automatisation des tests et la connaissance des différentes approches d'automatisation

Toute personne intervenant en tant qu'utilisateur, chef de projet, AMOA, AMOE

### Phase de rappels sur les tests

Définition du test

Le processus de test

Différents niveaux de test (unitaire, intégration, système, recette)

Différents types de test (statique, structurel, fonctionnel et non fonctionnels) et leurs techniques associées

Différents cycles de développement (V, itératif, Agile)

### Introduction à l'automatisation

Mythes et réalités

Que peut-on automatiser ?

Pourquoi automatiser ?

Dans quels types de cycle ?

ROI

### Gestion des tests unitaires

Environnement de test unitaire

Bouchon

Test statique et outillage

Tests structurels et outillage

Test fonctionnel et non fonctionnel

La présentation et l'utilisation de JUNIT3, ant et Cobertura

### Appréhender les tests systèmes

Environnement de tests systèmes

Les différentes stratégies de développement d'un simulateur

Bonnes pratiques

Qu'est ce qu'un framework de test ?

Test d'API/Web Service: TP JUNIT JUNIT 4, Web Service et DBUnit

Test d'IHM : spécificités, enregistrement rejeu, modularité, reconnaissance graphique

Présentation et utilisation de Selenium

### La programmation par mot clef

Concepts : automatisation systématique

Présentation de Robot Framework

Programmation par mot clef avec Robot Framework

### Comprendre le rapport de test

Générer un rapport de test

Le couplage avec un outil de gestion de test

La démonstration avec Testlink et Selenium

### Conclusion

Quelle compétence, quel profil ?

Choix d'un outil : mise en place, formation, manuel d'utilisation, bonnes pratiques

Coût et ROI

## **Formation sur l'élaboration d'un modèle conceptuel de données avec Merise**

Durée : 3 jours

Maîtrise des concepts fondamentaux liés à la modélisation d'application avec Merise

Toute personne intervenant en tant qu'analyste, développeur, concepteur

### **Historique**

#### **Appréhender l'approche Merise**

Comprendre le cycle de vie d'un projet  
Comprendre le cycle de décision et d'abstraction

#### **Présentation et positionnement des modèles Merise**

Modèles conceptuels, physiques, logiques et organisationnels

#### **Gestion du modèle conceptuel de communication (MCC)**

Acteurs internes et externes  
Gestion du flux d'informations échangés  
La validation du modèle

#### **Appréhender le modèle conceptuel de données (MCD)**

Domaines et règles de gestion  
Informations et entités  
Gestion des propriétés d'entités et des identifiants d'entités  
Les associations d'entités  
La définition des cardinalités

#### **Compréhension du modèle conceptuel de traitement (MCT)**

L'analyse des flux  
Définition des opérations, règles d'émission

#### **Modèle physique de données (MPD)**

La mise en œuvre avec génération MCD vers MPD  
Dé-normalisation d'un MPD

#### **Gestion du modèle organisationnel de traitements (MOT)**

Procédures et acteurs  
L'identification des phases et de leurs événements  
La définition des attributs des phases

## Formation sur les enjeux du test logiciel

Durée : 2 jours

Maîtrise des concepts fondamentaux liés aux enjeux et au métier de test

Toute personne intervenant dans les tests de logiciels

### **Cerner les enjeux du test**

Enjeux du test

Savoir ce qu'est un test ?

Identifier les objectifs des tests

La mesure de la qualité : une couverture ?

### **Appréhender les tests et les cycles de développement**

Test unitaire

Test d'intégration

Test système

Recette

Test et cycle de développement (cycle en V, itératif, mode agile, maintenance)

### **Compréhension des types de tests**

Tests statistiques (revue)

L'analyse statique (outil d'analyse de code)

L'analyse dynamique (cpu, couverture de code, etc...)

Les aspects liés au test fonctionnel et ses techniques

Le test non fonctionnel ou test technique

Le test exploratoire

Le test de non-régression

### **L'organisation des tests**

Gestion du rôle, équipe, indépendance des tests

Cerner la phase de planification : différentes stratégies, estimation des tests, exemple de stratégie (IEE829)

Cerner la phase d'analyse : spécifier et organiser ses tests (modèle de documentation)

Cerner la phase d'exécution : organisation des campagnes, environnements de test

Cerner le bilan : générer un rapport de test (modèle de document)

### **Les aspects liés à la recette**

Comprendre l'objectif

Comprendre la responsabilité MOA-MOE

### **Compréhension des environnements de test**

Framework de test

Simulateur, emulateur

Données de test

### **Outillage**

Compréhension des outils de gestion de test (panorama et démo outil Open Source)

Focus sur l'automatisation (panorama et démo outil Open Source)

Outil de test de charge (démonstration)

Identifier les avantages et désavantages des outils

Le choix et l'introduction d'un outil (démarche)

### **Conclusion**

## **Formation sur les fondamentaux liés à la conception logicielle et agilité**

Durée : 3 jours

Maîtrise des concepts fondamentaux liés aux fondamentaux de la conception logicielle

Toute personne intervenant en tant qu'architecte logiciel, chef de projet, développeur

**Comment allier qualité et productivité dans le projet de développement logiciel ?**

**Les activités d'ingénierie logicielle classiques (exigences, conception, codage, tests) et l'importance de la modélisation du métier.**

**Les fondamentaux de la programmation structurée/OO**

Séparation des dépendances

Limitation des dépendances

**Vue d'ensemble de la conception**

La conception dans le processus UP et l'approche MDA (Notions de PIM et de PSM)

Patterns d'architecture (couches, MVC)

**Une conception applicable à toute technologie (PIM)**

Une architecture MVC dans une vision "PIM"

La conception statique - bonnes pratiques : structuration en classes et en packages, limitation des dépendances, du bon usage de l'héritage

La conception dynamique - bonnes pratiques : définir la communication entre classes (requête/réponse et notification), du bon usage des interfaces, cycle de vie des objets

**La conception dans une technologie cible (PSM)**

La conception de la plate-forme technique : architecture matérielle et framework

La conception PSM du logiciel développé : conception des composants et de leur déploiement, bonnes pratiques et conception PSM des classes : prise en compte de la plate-forme technique (framework logiciel, distribution et persistance)

**Méthodes Agiles (e.g. SCRUM)**

L'utilisateur au cœur de la démarche

Accepter (et contrôler) les changements

Le caractère itératif de la méthode

La conception logicielle « agile », un « émergent design » maîtrisé

**Conclusion**

## **Formation sur les tests et recettes des applications informatiques**

Durée : 3 jours

Maîtrise des concepts fondamentaux liés à la méthodologie pour bâtir un plan de tests et élaborer des jeux d'essais

Toute personne intervenant en tant que Key User, chef de projet, AMOA, informaticien participant à la recette

### **Phase de rappels**

Comprendre les enjeux des tests

Différents niveaux de tests au sein des différents cycles de développement

Différents types de tests : fonctionnels et non fonctionnels

Processus de test

Comprendre la notion de couverture

### **Phase de préparation et suivi**

Comprendre ce qu'est une stratégie de test

Faire un zoom sur la stratégie basée risque

La définition de l'effort de test, des moyens à mettre en œuvre, des critères d'arrêts

Gestion de la mise en application sur un cas concret

Comprendre l'estimation des tests

Gestion du suivi qualité et planification

La mise en application

### **Comprendre la phase de conception**

Les revues

Qu'est ce qu'une exigence ?

Répertorier et organiser les exigences

La gestion d'un référentiel d'exigences

La mise en application avec un outil Open Source

Répertorier et organiser ses tests

La conception des tests : techniques de test fonctionnel (partition par équivalence, valeurs aux limites, table de condition, cas d'utilisation, etc...)

Mise en application avec un outil Open Source

Tests non fonctionnels

Cas particulier des tests d'intégration

Cas particulier des tests unitaires : revue de code

### **Comprendre la phase d'exécution**

La préparation et la maîtrise de son environnement de test et de ses données de test

La gestion des livraisons

L'exécution des tests unitaires et la mesure de la couverture de code

L'organisation des tests d'intégration : stratégie d'intégration et planification

L'organisation des campagnes de tests systèmes et d'acceptation

Gérer le suivi des tests et des anomalies

Quand arrêter les tests ?

La mise en application avec un outil Open Source

**Appréhender la recette**

Objectifs

Responsabilité MOA-MOE

Connaitre les critères de démarrage et les critères d'arrêt

Gestion de la recette opérationnelle (qualité de service, test de récupération, HA, etc...)

Gestion de la recette utilisateur (cas d'utilisation, Beta test)

**Bilan**

Générer des rapports de tests

Générer un bilan et améliorer son processus de test

La mise en application avec un outil Open Source

L'utilisation d'un outil

## Formation sur PRINCE2 (Foundation)

Durée : 3 jours

Maîtrise des concepts fondamentaux liés à la constitution, et l'application de la méthode de gestion de projets PRINCE2

Toute personne intervenant dans la gestion de projets

### **Introduction à la gestion de projet PRINCE2**

Historique

Tour d'horizon

PRINCE2 et la gestion de projet

### **Les thèmes**

Cas d'affaire

Organisation

Plans

Progression

Risques

Qualité

Changements

### **Les processus**

Diriger le projet

Elaborer le projet

Initialiser le projet

Contrôler une séquence

Gérer la livraison des produits

Gérer une limite de séquence

Clore le projet

### **Les principes**

Leçons tirées de l'expérience

Rôles et responsabilités définis

Justification continue pour l'entreprise

Management par séquences

Management par exception

Focalisation produit

Adaptation à l'environnement de projet

### **Mise en pratique de la méthodologie**

Etude de cas

### **Examen-test PRINCE2 Foundation**

Révision

Test blanc

### **Passage de l'examen PRINCE2 Foundation**

Examen délivré en salle et en français

## Formation sur PRINCE2 (Practitioner)

Durée : 2 jours

Maîtrise des concepts fondamentaux liés aux processus, aux composants et aux techniques de PRINCE2

Toute personne intervenant dans la gestion de projets

### Les éléments de PRINCE2

L'attribution des rôles selon les acteurs

Gestion de la gouvernance et de la matrice des responsabilités

Vue détaillée des produits de management

Vue détaillée des thèmes, processus et principes

Vue détaillée des techniques recommandées

Gestion des interactions entre les produits de management, des processus et de thèmes

L'adaptation et l'intégration des 4 éléments PRINCE2 sur un projet

L'adaptation des produits de management PRINCE2 à l'environnement projet

### La mise en pratique de la méthode

L'application des principes PRINCE2 aux projets

L'utilisation et l'adaptation des produits de management PRINCE2

L'utilisation des techniques recommandées

Gestion de la mise en pratique de l'élaboration, l'initialisation, le contrôle et la clôture d'un projet

PRINCE2 sur une étude de cas

### Examen-test PRINCE2 Praticien

Révision

Test blanc

**Gérer le passage de la certification PRINCE2 Practitioner**

## Formation sur UML : processus de développement

Durée : 5 jours

Maîtrise des concepts fondamentaux liés au formalisme d'UML 2.1 pour le développement d'un système d'information

Toute personne intervenant en tant que développeur, concepteur, architecte

### Phase d'introduction

Les objectifs et l'approche générale

Compréhension des notions de base objet en UML

Gestion de la vue d'ensemble de la méthode

Vue d'ensemble de la notation UML

### Gérer la modélisation du métier

La description des processus métier avec UML :

- Identification des processus métier et des rôles externes
- Identification des rôles internes
- Organisation interne du métier
- Description des processus : activity diagram

La modélisation conceptuelle des informations métier: quelques points clés

Etude de cas avec Enterprise Architect

### Appréhender l'expression des besoins

Les besoins fonctionnels : acteurs, cas d'utilisation et scénario (use case diagram et sequence diagram)

Les besoins non fonctionnels : approche FURPS

Comprendre la description détaillée des interfaces

Etude de cas avec Enterprise Architect

### Analyse

L'analyse dans le processus de développement :

- Objectifs: une préconception pour un logiciel maintenable et évolutif
- L'approche MDA (Model Driven Architecture)
- L'architecture logique en couches

Modélisation statique des classes : class diagram :

- Attribut
- Association
- Généralisation
- Structuration en packages
- Opération
- Étude de cas avec Enterprise Architect

Modélisation dynamique :

- Réalisation de cas d'utilisation : sequence diagram
- Cycle de vie d'objets : state machine diagram
- Étude de cas avec Enterprise Architect

**La conception**

La conception des composants du logiciel et de leur déploiement :

- Architecture physique : diagramme de déploiement
- Conception de composants

La conception des classes de conception :

- Affinement technique des classes d'analyse
- Conception d'interfaces

Etude de cas avec Enterprise Architect

**Compléments méthodologiques**

Compléments sur le "Unified Process" et les processus itératifs

Mettre en place UML dans l'entreprise