



Formations Objis

1^{er} semestre 2009



Table des matières

Introduction.....	3
Présentation OBJIS.....	4
Les 5 atouts Objis.....	4
Exemple de bilan pédagogique Objis	4
PARTIE 1 : Calendrier de formations.....	6
Janvier	6
Février.....	6
Mars.....	6
Avril.....	6
Mai.....	6
Juin.....	7
PARTIE 2 : Modules de formation	7
Catégorie 1 : gestion Vues / écrans web.....	7
Contexte d'utilisation :.....	7
Formation SPRING (5j).....	7
Formation STRUTS (3j).....	11
Formation AJAX/WEB2 (5j).....	13
Formation JSF (3j).....	15
Catégorie 2 : gestion Persistence / accès données.....	17
Contexte d'utilisation :.....	17
Formation HIBERNATE (3j).....	17
Formation EJB3 et Web Services (5j).....	19
Catégorie 3 : conception.....	21
Contexte d'utilisation	21
Formation DESIGN PATTERN (2j).....	23
Formation XML (4j).....	25
Catégorie 4 : administration serveur J2EE.....	27
Contexte d'utilisation	27
Formation TOMCAT (3j).....	27
Formation JONAS (5j).....	29
Catégorie 5 : tests et intégration continue.....	31
Contexte d'utilisation	31
Formation TESTS (2j).....	31
Formation SUBVERSION (1j).....	33
Formation MAVEN (3j).....	35
Catégorie 6 : remise à niveau.....	37
Contexte d'utilisation	37
Remise à niveau JAVA (2j).....	37
Remise à niveau J2EE (2j).....	37
PARTIE 3 : Les cours du soir Objis.....	38
Format et principe.....	38
Calendrier	39
Avantages.....	39
Coût	39

Introduction

La technologie Java s'impose de plus en plus dans le choix des décideurs pour la réalisation des applications clés de leurs système d'information. Malgré sa jeunesse (12 ans seulement), Java est aujourd'hui un acteur clé de l'industrie logicielle, tout comme ses principaux cocurrents que sont les technologies de développemnet.NET et PHP.



Objis a depuis ses débuts choisi de former dans un esprit d'excellence ses clients dans l'utilisation de cette technologie pour créer de la valeur. Nous étoffons cette année notre catalogue avec des formations aux outils d'intégration continue et de développement agile, pour un meilleur pilotage et suivi des projets. **Plus qu'un catalogue, ce document se veut un guide, catégorie par catégorie, pour le choix de technologies dont la maîtrise (grâce à Objis !) constituera un élément différentiateur sur la qualité de la conception et du développement de votre application.** Bien sûr Objis saura vous conseiller la meilleure formation en fonction de votre situation, votre projet. **Appelez-nous !**

Avec un taux de satisfaction client de 97%, Objis se positionne parmi les centres de formation à forte valeur ajoutée sur les technologies JAVA/J2EE. Nous tacherons de faire honneur à notre réputation en cette année 2009.

Nous avons constaté en 2008 une attente particulière chez nos clients SSII : faire évoluer le plus rapidement possible leur consultants développeurs vers un rôle de concepteur/architecte. Nous allons relever ce nouveau challenge avec des parcours spécifiques que nous vous conseillons. De plus, nous proposons un nouveau produit, qui permet la **formation continue des collaborateurs** sans affecter leur disponibilité chez vos clients : les **cours du soir Objis** (présentés en 3ème partie)

Toute l'équipe technique et pédagogique Objis vous souhaite une excellente année 2009...et des formations à forte valeur ajoutée avec Objis, le spécialiste de la formation JAVA.

Douglas MBIANDOU
Directeur technique Objis
Responsable de la formation

Présentation OBJIS

Objis est un centre de formation en informatique spécialiste du transfert de compétences JAVA Entreprise. Depuis 2005, nous améliorons la productivité des consultants techniques de nos clients, à différentes étapes de leur intervention:

- Conception,
- Développement,
- Tests
- Intégration continue
- Administration serveurs J2EE

Nous **fournissons** à nos clients les meilleures pratiques, contribuant ainsi à une meilleure productivité logicielle, pour des projets Systèmes d'information de petites, moyenne et grandes tailles.


Notre objectif : faire évoluer tout développeur vers un rôle de concepteur / Architecte logiciel. Lui fournir les bonnes pratiques et réflexes associés à un projet de qualité professionnelle et modulaire, robuste, maintenable, sécurisé, documenté.

Les 5 atouts Objis

- Audit technique de vos besoins
- 70% de travaux pratiques (TP).
- Suivi au quotidien du savoir-faire acquis.
- Bilan pédagogique individuel
- Clé USB avec outils et Corrigés TP

Exemple de bilan pédagogique Objis

Une des valeurs ajoutées d'Objis par rapport à d'autres centres de formation est notre bilan individuel de formation. Il vous sera remis pour chacun des collaborateurs ayant suivi une formation avec Objis. Ci-dessous un exemple avec une formation SPRING réalisée en 2008.

 Nous allons vous faire aimer Java		Formation	SPRING		
		Date	1-5 decembre (5j)		
		Formateur			
Stagiaire	Pré-requis	Assimilation Concepts SPRING	Réalisation Travaux pratiques SPRING	Capacité à mise en oeuvre SPRING en projet	Remarque Formateur
Charles XXX	4	5	5	5	+ Bonnes capacités d'abstraction + Comprend et met en oeuvre vite...et bien + Questions pertinentes
Jacques XXX	3	5	5	4	+ Curieux - A tendance à vouloir comprendre détails tt de suite
Ibrahim XXX	4	5	5	4	+ Curieux + Bonne comprehension bonnes pratiques - Attention méthodologie : appliquer tt de suite bonnes pratiques !
			2		Insuffisant
			3		Passable
			4		Satisfaisant
			5		Très satisfaisant

Nos références

Les clients suivants nous font confiance depuis 4 ans :

- **ACTIANE CONSULTING** (www.actiane.fr)
- **ARMEE de l'AIR** (www.defense.gouv.fr/air/)
- **AWAT** (www.awat.fr)
- **BANQUE CENTRALE de MAURITANIE** (www.bcm.mr)
- **BLUECHAM** (www.bluecham.net)
- **BULL** (www.bull.fr)
- **EUROPTIMAL** (www.europtimal.fr)
- **GENESIS** (www.genesis-groupe.com)
- **ANFA** (www.anfa-auto.fr)
- **GTM INGENIERIE** (www.gtm-ingenierie.fr)
- **IDYAL CONSULTING** (www.idyal.com)
- **IT VALUE** (www.itvalue.fr)
- **IVISION** (www.ivision.fr)
- **LAGARDETTE INFORMATIQUE** (www.laginfo.com)
- **LINAGORA** (www.linagora.com)
- **OCEANE CONSULTING** (www.oceaneconsulting.com)
- **OXYMEL** (www.oxymel.com)
- **SMART TRADING DIFFUSION** (www.cmoinscher.fr)
- **SOAT** (www.soat.fr)

PARTIE 1 : Calendrier de formations

Janvier

- Hibernate/JPA (12-14)
- Tomcat (26-28)

Février

- Spring (9-13)
- Ajax/Web2 (16-20)
- Maven (25-27)

Mars

- Struts (11-13)
- Ejb3/Webservices(16-20)
- UML(24-27)

Avril

- Spring(6-10)
- Hibernate(15-17)
- Tests(29-30)

Mai

- SPRING (11-15)
- Maven(18-20)
- UML(26-29)

Juin

- Subversion (4)
- Spring (8-12)
- Design Pattern (25-26)

PARTIE 2 : Modules de formation

Nous avons classé nos formations **selon les critères suivants leur catégorie d'utilisation :**

- **Catégorie 1: gestion Vues/écrans**
- **Catégorie 2: gestion Persistence / accès données**
- **Catégorie 3: conception**
- **Catégorie 4: administration serveur J2EE**
- **Catégorie 5: tests et intégration continue**
- **Catégorie 6: remise à niveau Java/J2ee**

Pour chaque catégorie, nous précisons le contexte d'utilisation

Catégorie 1 : gestion Vues / écrans web

Contexte d'utilisation :

Créez des écrans applicatifs à partir d'un des frameworks suivants :

Formation SPRING (5j)

Présentation

Suite aux 15 travaux pratiques que nous proposons dans cette formation Spring, vous serez prêts à mettre en oeuvre Spring dans tout type de projet java/j2ee. Suite à cette formation 'commando' résolument pratique (70%), vous maîtriserez le framework spring, qui représente le framework de développement java/j2ee le plus complet du marché.

Limitez les dépendances entre les différentes couches de vos applications. Injectez vos composants métiers, vos services, vos DAO sous forme de bean SPRING. Testez plus facilement vos applications et coupez efficacement Spring avec les frameworks modernes de présentation (Struts/JSF) , de persistence (Hibernate/Toplink/iBatis), de service Web (Axis), de messages MOM(ActiveMQ). Augmentez la sécurité, maintenabilité et l'évolutivité de vos applications !

Durée/Tarifs

5 jours

Inter : 2150 Euros HT

Intra : 7500 Euros HT (6 personnes)

Programme détaillé

- **Problématique du développement J2EE**
 - Les frameworks Java
 - Les conteneurs légers
 - Présentation du projet Spring Framework
 - Les modules de Spring
 - Pourquoi Spring simplifie le développement J2EE
- **Principe de Spring**
 - Histoire d'instanciation
 - Le design pattern Inversion de Contrôle
 - Présentation de l'AOP
 - La couche d'abstraction de Spring

- **Premiers pas**
 - Coder par interface
 - Faire un lookup
 - Implémenter une classe métier
 - Configurer l'loC
 - Synthèse
- **Fabriquer des beans avec Spring**
 - Rôle de l'interface BeanFactory
 - Méthodes de l'interface BeanFactory
- **Déclaration des beans**
 - Les façons d'instancier un bean
 - Méthodes de callback
- **Configurer les propriétés et les dépendances**
 - Propriétés simples
 - Propriétés de type collection
 - Associations entre beans
 - Méthodes d'injection
 - Auto-câblage
- **Notions avancés**
 - Beans abstraits
 - Héritage
 - Interfaces de callback
 - Injecter la BeanFactory dans un bean
 - Les post-processors
- **L'ApplicationContext**
 - Définition
 - Les différents type d'ApplicationContext
 - Les catalogues de messages
 - Injecter l'ApplicationContext dans un bean
 - Gestion évènementielle
 - Les post-processors
 - Interfaces de callback
- **Spring et la Persistance**
 - Qu'est-ce que la persistance ?
 - Définition
 - Développer une couche de persistance (développement JDBC, frameworks de type Data Mapper, frameworks de type ORM)
 - Les apports de Spring
- **Spring JDBC**
 - Les classes de la couche d'abstraction
 - Configurer l'loC
 - Exécuter des requêtes de lecture

Exécuter des requêtes d'écriture
Récupérer les valeurs des auto-incréments

- **iBatis**
 - Présentation du framework
 - Les classes de la couche d'abstraction
 - Configurer l'loC
- **Hibernate**
 - Présentation du framework
 - Les classes de la couche d'abstraction
 - Configurer l'loC
- **Gestion des transactions**
 - Concept de transaction
 - Gérer les transactions avec Spring
 - Transactions programmatiques
 - Transactions déclaratives
- **AOP, remoting et sécurité**
 - Spring AOP
 - Définitions et concepts
 - Créer des pointcuts avec Spring
 - Créer des advices avec Spring
 - Créer un aspect avec Spring
 - Stratégie de weaving
 - Créer des proxy AOP avec Spring
- **Techniques de remoting**
 - Présentation du remoting
 - Exposer un service avec RMI
 - Exposer un service avec HTTP invoker
 - Exposer un service avec Hessian / Burlap
- **Module de sécurité Acegi**
 - Présentation du module Acegi
 - Gestion de l'authentification
 - Sécuriser l'invocation des objets
- **Programmation Struts avec Spring**
 - Rappels Struts
 - Injecter les beans métiers dans les actions Struts
 - Charger l'ApplicationContext
 - Configuration Struts
 - Spring MVC
 - Principe
 - Configurer l'loC
 - Comparaison Struts / Spring MVC
- **Spring et la Couche Métier**

- La plate-forme J2EE
 - Présentation
 - Les apports de Spring
 - EJB
 - Présentation du modèle EJB
 - Principe des EJB
 - Programmation EJB avec Spring
 - JMS
 - Les middlewares orientés messages
 - Différences RPC / MOM
 - L'API JMS
 - Utiliser JMS avec Spring
 - Web Services
 - Principe des services web
 - Java et les services web
 - Programmer un service web avec Spring

Formation STRUTS (3j)

Présentation :

Suite aux 10 travaux pratiques que nous proposons, vous serez prêts à mettre en oeuvre Struts pour la réalisation, le test et l'évolution d'applications web multilingues. Résolument pratique (70%), cette formation permet aux développeurs de servlets/JSP d'augmenter significativement leur productivité.

Ce cours de trois jours vous apprendra les règles, trucs et astuces permettant d'exploiter au mieux toutes les possibilités de Struts. Vous commencerez par apprendre à mettre en place une infrastructure de navigation et de gestion des requêtes efficace, avant de l'enrichir avec les mécanismes avancés de Struts, comme les TagLibs d'internationalisation ou les modules, puis avec ses extensions comme Tiles ou Validator.

Vous allez apprendre à

- Concevoir et développer des applications Web en utilisant le modèle d'architecture MVC de Struts
- Configurer correctement vos applications Struts mono ou multi-modules
- Utiliser les composants, la logique et les bibliothèques de tags HTML pour écrire des JSP faciles à maintenir
- Exploiter les capacités d'internationalisation (i18n) du Framework Struts
- Créer un formulaire unique capable de produire de multiples actions
- Écrire du code de validation (client et serveur) facile à maintenir grâce au Validator
- Utiliser les DynaForms pour réduire la quantité de code nécessaire à la création de Form Beans
- Éviter les erreurs classiques d'utilisation de Struts grâce aux règles de bon usage élaborées sur des projets Struts réels
- Améliorer la maintenabilité de vos pages Web en factorisant les éléments communs grâce au Framework Tiles

Durée/Tarifs

3 jours

Inter : 1200 Euros HT

Intra : 4000 Euros HT (6 personnes)

Programme détaillé

- Introduction
 - Pourquoi utiliser Struts ?
 - Front Controller Pattern
 - Les composants de base de Struts
 - Cycle de vie d'une requête Struts
- Configuration et déploiement
 - Comment obtenir Struts
 - Contenu de la distribution Struts
 - Les fichiers de configuration
 - Utilisation de la documentation
- La bibliothèque de Tags HTML
 - Tags Struts : notions de base
 - Tags pour les formulaires

- Relations avec les ActionForms
- Afficher les messages d'erreur
- Options de création des liens
- Traiter les URL relatives
- Forwards et Forwarding ActionMappings
 - Objectifs, options et utilisation des forwards
 - Forwards internes vs. Redirections
 - Comment éviter les URL en dur dans vos JSP
 - Création dynamique d>ActionForwards
 - Utilisation des ActionMappings
- Bean Tags et i18n
 - Utilisation du tag Write avec un Bean
 - Utilisation du tag Message pour l'internationalisation des sorties
 - Autres Bean Tags
 - Tags de contrôle du flux
 - Tags imbriqués
- Les Actions en détail
 - Architecture des actions et multi-threading
 - Méthodes de la classe Action
 - Utilisation de Tokens pour éviter les envois multiples d'un formulaire
 - Options d>ActionMapping
 - Actions standard
 - Formulaire à actions multiples
- Les ActionForms en détail
 - Cycle de vie des ActionForms
 - Validation et ActionErrors
 - DynaActionForms
- Le Framework Validator
 - Composants Validator et fichiers de configuration
 - Les Validators intégrés
 - Syntaxe du fichier validation.xml
 - Génération de JavaScript
 - Formulaires multi-pages
- Tiles
 - Création de templates
 - Définitions de Tiles dans les JSP et tiles-defs.xml
 - Etendre les définitions
- Modules
 - Structure d'une application multi-modules
 - Redirection entre modules
 - SwitchAction
 - Utilisation de fichiers de configuration multiples
- Sujets avancés et évolutions
 - Sécurité avec Struts
 - Extension du Framework Struts
 - Configurer des Datasources
 - JSTL et la bibliothèque de Tags Struts-EL

Formation AJAX/WEB2 (5j)

Présentation:

Ajax est devenu une technologie incontournable du développement d'applications riches Web 2.0, mêlant ergonomie et rapidité. Durant ce cours, vous apprendrez la manipulation de ses composants, (Javascript, XML, CSS, le DOM) ainsi qu'à établir des échanges client-serveur avec l'objet XMLHttpRequest. Cette formation vous donnera les outils pour développer efficacement en Javascript, mettre en oeuvre et comparer différents frameworks Ajax : YUI, GWT, Dojo, jQuery, OpenLaszlo.

La sécurité constituera également un des atouts de cette formation dense.

Cette formation est destinée à tous ceux qui cherchent à créer des applications web 2.0 ou bien souhaitent choisir un framework afin de migrer leurs applications web actuelles vers une version orientée web 2.0.

Durée/Tarifs:

5 jours

Inter : 1850 Euros HT

Intra : 6500 Euros HT (6 personnes)

Programme détaillé:

- Introduction web Ajax/2.0
 - Concepts web 2.0,
Exemples d'applications web 2.0,
Présentation Ajax et technologies associées
Les objectifs d'Ajax.
Accessibilité
- Rappels XML
 - La structure XML.
XML bien formé et validé.
namespaces
validation XML : DTD et XSD
- Rappels Javascript
 - La programmation Objet.
Le type de variables.
Les tableaux.
Le parseur XML.
Ecmascript et l'émergence du langage ActionScript. Avantage d'E4X.
- JSON (Javascript Object Notation)
 - Avantages et inconvénients dans les échanges.
Objets, Arrays, syntaxe et parseur JSON..
- DOM
 - Importance du langage XHTML.
Les composantes du Dynamique HTML.
Le rôle du DOM dans la programmation XML.
Gestion dynamique de CSS.
Accès, modification, ajout des éléments et des attributs (AddChild, getAttributs...).

- XMLHttpRequest
 - La classe XMLHttpRequest.
Créer un objet XMLHttpRequest.
Utiliser un objet XMLHttpRequest.
Gestion des réponses (handle). Code de retour des serveurs HTTP. Développement d'un gestionnaire d'erreur.
Comment dialoguer en mode synchrone et asynchrone avec le serveur et
- TraitementXML en Javascript
 - Présentation XSLT.
Le langage Xpath.
Le format RSS.
- Technologies XML/Javascript coté client
 - XUL
XAML
XFORMS
HTML5
- Frameworks Ajax
 - YUI
 - Dojo
 - jQuery
- Ajax coté serveur
 - OpenLaszlo
 - Orbeon Presentation Server
 - GWT
- Sécurité
 - Attaques : XSS, SQL Injection
 - Audit de sécurité

Formation JSF (3j)

Présentation:

JSF est au développement d'écrans web ce que JPA est au développement d'une couche persistante : un standard. Ce nouveau Framework Web du monde Java rapproche la création d'applications intranet / Internet du développement rapide d'applications client / serveur (Delphi, PowerBuilder, Visual Basic) et permet de sérieux gains de productivité.

Tout en mettant l'accent sur les travaux pratiques et l'utilisation d'éditeurs Java / JSF, ce cours vous permettra de comprendre en détail le fonctionnement du Framework JSF afin de le faire répondre au mieux à vos besoins tout en découvrant d'où peuvent venir les problèmes de performance dans les applications JSF.

Durée/Tarifs:

3 jours

Inter : 1200 Euros HT

Intra : 4000 Euros HT (6 personnes)

Programme détaillé:

- JavaServer Faces, les bases
 - Historique
 - Pourquoi un nouveau Framework ?
 - Caractéristiques essentielles
 - Les implémentations : JSF-RI, MyFaces, IBM-JSF...
- Fonctionnement de JSF
 - Cycle de vie d'une page JSF
 - Règles de navigation
 - Managed et Backing Beans
 - Actions et gestion d'événements
 - TagLibs et JSF
- Configuration et déploiement
 - Le fichier faces-config.xml
 - Intégration dans une application Web (web.xml)
 - Compatibilité des moteurs de Servlets
- Utilisation des composants JSF
 - Affichage
 - Validation
 - Internationalisation
 - Tableaux de données
 - Convertisseurs
- Conception avancée et personnalisation du Framework
 - Bases de la création de composants JSF
 - Les Design Patterns dans JSF
 - Lien entre JSF et les autres couches de l'application
 - Utiliser Facelets comme View Technology
 - Templating avec Facelets

- Les enjeux de JSF
 - Les environnements de développement JSF
 - Les bibliothèques de composants
 - JSF et les performances
 - JSF et la sécurité

Catégorie 2 : gestion Persistence / accès données

Contexte d'utilisation :

Mettez en oeuvre une couche d'accès aux données à partir d'un des frameworks suivants

Objis SAS au capital de 49500 € - RCS Lyon – SIRET 482 075 066 0027 – APE 804 C – TVA Intra FR 24 482075066 – agrément : 82.69.08812.69,
425 cours Emile Zola – 69100 Villeurbanne – 04 78 29 37 26 - Tél : 04 78 29 37 26 – Fax : 09 56 22 89 49 – contact@objis.com

Formation HIBERNATE (3j)

Présentation:

A travers 10 travaux pratiques, maîtrisez l'utilisation d'Hibernate, framework de persistance le plus utilisé actuellement. Développez également un regard critique sur son efficacité et sa pérennité. Abordez les problèmes indissociables des couches de persistance : le découplage du code métier vis-à-vis du framework de persistance, le mapping avec une base existante, la gestion de cache, des transactions et de la concurrence d'accès.

Découvrez enfin la configuration d'Hibernate comme fournisseur de persistance JPA

Durée/Tarifs:

3 jours

Inter : 1200 Euros HT

Intra : 4000 Euros HT (6 personnes)

Programme détaillé:

- Persistance avec Hibernate
 - Processus de développement
Connexion à la base de données
Configuration d'Hibernate et Session Factory
Création, ajout et suppression d'objets
Session Hibernate
- Conception avec Hibernate
 - Design Patterns pour couche de persistance
Persistance du modèle objet
Gestion de l'héritage
Gestion des associations
- HQL
 - Clauses des requêtes
Les fonctions d'agrégation
Les requêtes polymorphes
Les sous requêtes
- Utilisation avancée d'Hibernate
 - Transactions
Exceptionse
Gestion du cache
Lazy-Loading
- Mapping objet relationnel
 - Ecriture du fichier de mapping
Génération du fichier par Xdoclet
Gestion de l'identifiant
Types de données et personnalisation

GTypes de mapping (Top-Down, Bottom-Up, Meet in the Middle)

Stratégies de mapping d'héritage (une table par classe, une table par hiérarchie...)

Stratégies de mapping d'association (par clé étrangère, par table de jointure...)

- Architectures techniques
 - Hibernate dans une application client / serveur
 - Hibernate dans une application Web
 - Hibernate dans une architecture distribuée
 - Limitations et interrogations
 - Pourquoi choisir Hibernate ?
 - Limites techniques
 - Limites stratégiques
 - Hibernate et EJB3

Formation EJB3 et Web Services (5j)

Présentation:

les EJB 2 disparaissent et sont remplacés par une double spécification regroupant les API EJB 3 et JPA. Cette nouvelle version des EJB apporte des modifications notables dans le modèle de développement et intègre la nouvelle API JPA (Java Persistence API) pour la gestion de la persistance. Cette dernière tire profit du succès des

Frameworks tels qu'Hibernate et Toplink.

Ce cours, focalisé sur la mise en oeuvre des EJB 3 et l'analyse de leurs nouveautés, vous permettra de comprendre comment mettre en oeuvre une application multi-niveaux en utilisant les EJB session pour votre couche de service et JPA pour votre couche d'accès aux données.

Durée/Tarifs:

5 jours

Inter : 1900 Euros HT

Intra : 6500 Euros HT (6 personnes)

Programme détaillé:

- Présentation des architectures multi-niveaux
 - La couche présentation
 - La couche service
 - La couche d'accès aux données
 - Services techniques associés à ces couches
- La relation conteneur / composant
 - Définition
 - Le proxy
- Les EJB session
 - Local ou distribué ?
 - Avec ou sans état ?
 - Démarcation transactionnelle
 - L'injection de dépendance
- JPA : la persistance des objets Java
 - Présentation du mapping objet / relationnel
 - Les méta-données du mapping
 - JPQL : un langage de requêtage objet
 - Le contexte de persistance
 - Optimisation : Lazy Loading ou JPQL
 - Gestion des Locks (optimistes ou pessimistes)
- La sécurité
 - Déclaratif
 - Programmatique
- Les EJB Messages : Message Driven Bean
 - Présentation des Middlewares orientés messages (JMS)
 - Développer un MDB
- Gestion des transactions
 - Stratégies de transaction et de verrouillage
 - Lien avec le SGBD
 - Démarcation transactionnelle

Transactions distribuées

- EJB et Web Services
 - Notion de Endpoint
 - Standardisation JAX-WS
 - Génération de code et déploiement
 - Impact sur le client

Catégorie 3 : conception

Contexte d'utilisation

Traduisez efficacement les besoins de vos clients et créez des applications robustes, maintenables

Formation UML (4j)

Présentation:

Les technologies Objet ont grandement contribué à "industrialiser" les réalisations informatiques. Développer "objet", c'est travailler sur la base de concepts stables et largement éprouvés, et c'est bénéficier d'une panoplie d'outils et de langages performants. Parmi ceux-ci, UML est devenu un standard incontournable des activités d'analyse, de conception et de spécifications fonctionnelles ou opérationnelles.

Avec Objis, montez en puissance sur UML et développez réflexes clés.

Durée/Tarifs:

4 jours

Inter : 1650 Euros HT

Intra : 5500 Euros HT (6 personnes)

Programme détaillé:

- Comprendre les principales notions de l'approche objet
 - Rappel sur les paradigmes de programmation (logique, impératif, objet...)
Les objets : identité, état et comportement. Analogie monde réel et systèmes d'Info.
Popularité, avantages de l'Orienté Objet.
Abstraction, encapsulation, classification. Classes et instances. Classes abstraites. Concept d'héritage.
Méthodes et envois de messages entre objets. Polymorphisme. Surcharge et redéfinition.
- Analyse et conception objets, introduction à l'UML
 - Pourquoi modéliser ? Appréhender le spectre de l'analyse et de la conception
Domaine métier et modélisation d'une solution informatique. Le modèle, un artefact central du processus projet.
Analyse et conception d'une solution informatique. Impacts des langages de programmation.
Evolution vers l'analyse/conception Objet. Avantages.
- Présentation générale d'UML
 - Historique, évolution et objectifs. Vues de l'architecte.
Le coeur de l'UML : les différents types de diagrammes. Différences entre diagrammes statiques et dynamiques.
Présentation de plusieurs démarches de modélisation.
Extensions UML : stéréotype, profils, contraintes, etc.
- Diagramme UML des cas d'utilisation
 - Capture et description des besoins fonctionnels de l'application
Principaux objectifs et utilisation. Décrire les fonctionnalités du système.
Eléments du diagramme : cas d'utilisation, acteurs et frontière du système.
Etapas de construction du modèle des cas d'utilisation.
Comment identifier les acteurs ? Comment décrire un cas d'utilisation ? Les scénarios.
Formats, pré-conditions, post-conditions, relations (utilisation, inclusion, extension).
- Diagrammes UML de modélisation statique
 - Montrer une vue d'ensemble du système, de ces éléments et de leurs relations
Diagramme de classe : son rôle et son utilisation. Exemples.
Comment identifier les classes utiles ?
Une classe en UML : nom, attributs et opérations. Visibilité (publique, privée, protégée).

Formalisme et notation.

Relations entre classes (association, généralisation, agrégation et composition).

Multiplicités, rôles, contraintes, etc.

Classes abstraites, interfaces, packages.

Diagramme d'objets et instances de classes.

Diagramme de structure composite.

- Diagrammes UML de modélisation dynamique
 - Montrer l'évolution du système et les interactions entre objets
Diagramme de séquence : interactions entre objets au cours du temps. Message (synchrone et asynchrone).
Diagramme global d'interactions : enchaînements possibles entre les scénarios identifiés sous forme de diagrammes de séquences.
Diagramme de communication : rôle des objets, interactions, concurrence de traitements...
Diagramme d'état transition : états possibles d'un objet et événements déclenchant les transitions.
Diagramme d'activité : flux des activités pour réaliser une opération, objets en charge de ces activités.
Diagramme de temps : variations d'une donnée au cours du temps.
Notations et exemples.
- L'architecture logicielle et matérielle du système.
 - Modèles d'architectures. Organisation en couches. Sous-systèmes.
Les paquetages et leurs relations.
Diagramme de composants : organisation du code en modules, dépendances.
Diagramme de déploiement : déploiement physique du système (machines, réseaux, etc.).
- Concepts avancés, outillage
 - Compléments de conception
Modèles de données. Mapping Objet/Relationnel. Autres modèles.
Conception des écrans d'IHM.
Répondre à des problèmes récurrents
Design Patterns (ex : singleton, adaptateur, proxy, façade...), leur rôle dans la conception.
Frameworks, réutilisation.
- Finalisation du système. Ateliers de modélisation
 - Génération de rapports, de code. Création de stéréotypes, etc.
Format XMI d'échange de modèles UML entre AGLs.
UML et les méthodes projet
Différentes approches. Intégration d'UML.
Introduction au Rational Unified Process (RUP), itérations, phases et activités.
Autres possibilités (XP, etc.).

Formation DESIGN PATTERN (2j)

Présentation:

La puissance du concept objet fait de la modélisation logicielle un métier de spécialistes dont seuls l'expérience et les retours terrain alimentent l'expertise. Avec Objis, développez des compétences opérationnelles sur le design des applications. Gagner en productivité grâce à l'utilisation des patterns. Les nombreux cas pratiques vous apprendront à modéliser et réaliser des composants et des applications évolutives et réutilisables, et à

comprendre les principaux patterns de conception.

Durée/Tarifs:

3 jours

Inter : 1200 Euros HT

Intra : 4000 Euros HT (6 personnes)

Programme détaillé:

- Présentation du design
 - Rappel des notions fondamentales de la programmation OO et d'UML. Les diagrammes de la notation UML. Ses apports pour la conception.
Les enjeux de la conception : accroître la réutilisation sans freiner les évolutions.
La réutilisation par l'héritage : avantages et inconvénients.
- Principes fondamentaux en conception objet
 - La stratégie d'évolution avec le principe d'ouverture/fermeture (OCP).
Une réutilisation efficace par l'héritage et les interfaces : le principe de substitution de Liskov (LSP)
Le concept de polymorphisme.
L'impact de la conception objet sur le cycle de vie des projets.
- Principes d'organisation en packages
 - Le package comme unité de conception avec les principes d'équivalence livraison/réutilisation (REP) et de réutilisation commune (CRP).
Le découpage des packages gr'ce au principe de fermeture commune (CCP).
L'organisation entre packages : principes des dépendances acycliques (ADP) et de relation dépendance/stabilité (SDP).
- Principes de construction des classes
 - La gestion raisonnée des dépendances avec l'inversion de dépendance (DIP).
La réduction de la complexité apparente par la séparation des interfaces (ISP).
La répartition des responsabilités avec le principe de GRASP.
- Principes des design patterns
 - Les principes techniques de la conception d'une application objet.
Comment réutiliser de l'expérience lors de la conception et du développement d'applications objet : les design patterns ou « patrons de conception » comme solutions logicielles.
Origine et portée des patterns.
Les avantages et les limites des design patterns.
Les design patterns comme réponse aux problèmes techniques.
Résoudre des problèmes récurrents et assurer la pérennité des développements.
- Les patterns fondateurs de Gamma et Gof
 - Le catalogue de patterns de la "bande des quatre".
Les objectifs et les avantages.
Isoler la création des objets de leur utilisation avec les patterns de création d'objets : fabrique, singleton et prototype.
Affiner l'affectation des responsabilités grâce aux patterns comportementaux : chaîne de responsabilité, patron de méthode et observateur.
Améliorer la structuration des classes avec les patterns de structure : adaptateur, façade et

composite.

Formation XML (4j)

Présentation:

XML est dans le monde des données ce que JAVA est dans le monde programmation : **un langage portable**. Devenue une technologie incontournable tant sur le poste utilisateur (Ajax/web2) que sur les serveurs de l'entreprise (configuration d'applications, stockage de base de données, web services...), XML n'est pas moins complexe. Le besoin de partage et d'échange d'informations entre utilisateurs et entre applications a accéléré l'usage dXML. Avec Objis, développez vos compétences XML (Conception, programmation) et découvrez comment développer en XML avec Java/PHP/.Net.

Objis SAS au capital de 49500 € - RCS Lyon – SIRET 482 075 066 0027 – APE 804 C – TVA Intra FR 24 482075066 – agrément : 82.69.08812.69, 425 cours Emile Zola – 69100 Villeurbanne – 04 78 29 37 26 - Tél : 04 78 29 37 26 – Fax : 09 56 22 89 49 – contact@objis.com

Durée/Tarifs:

4 jours

Inter : 1600 Euros HT

Intra : 4900 Euros HT (6 personnes)

Programme détaillé:

- L'écosystème XML
 - Rappel sur les langages structurés : SGML, HTML, XML et XHTML.
Problématique du contexte actuel. Pourquoi XML est incontournable.
XML et le schéma directeur du S.I. La structure XML (élément et modélisation).
Avantages et inconvénients de XML.
Rôle prédominant des offres bureautiques.
Les techniques XML (WML, SVG, VoiceXml) et les technologies XML orientées Métier (FixML, CML...).
- Les parseurs XML
 - Offre du marché.
Rôle des parseurs. Leurs performances, avantages et inconvénients.
Les parseurs API : type push (SAX), type pull (XMLPull ou StAX), Les représentations arborescentes (DOM, JDOM, ...), type « Data Binding » (Castor ou JAXB).
Les Data Island et l'intégration dans les pages Web.
Les accélérateurs XAN (XML Aware Network).
Les parseurs Java (JAXP), .Net, Flash (XML Parser).
Ajax et la lecture de données dans les pages HTML.
- Validation de documents
 - Conception
Structure et composants d'un document XML :
Les méta-données (XMP, IPTC, EXIF).
Document bien formé et document valide.
- Modèles de validation
 - Les DTDs, XML-schema (XSD), Relax-NG, Schematron. Syntaxe et mise en oeuvre.
Les DTD de l'édition (MARC, TEI). Schéma XML
Schémas XSD : structure, typage des données, outils de composition. Modélisation.
Les formats XSD, XDR, SOX, DCD.
Exemple d'une mise en oeuvre DocBook.
- Namespaces
 - Les Namespaces. Rôle, intégration, partage, création.
Exemple : appel à partir d'une page XHTML.
Transformation XSLT
Enjeux du langage XSLT sur la transformation XML.
Intégration dans le système d'information.
Le langage de transformation XSL-T 1.0 et 2.0 : construction d'arbres, restructuration, génération multiformats : XHTML, SVG, PDF.

- Publication de documents
 - Visualisation brute sous les navigateurs.
Mise en forme avec les feuilles de style CSS.
Rôle du XHTML sur l'accessibilité. Conseils techniques.
XPath 1.0 et 2.0 pour la navigation dans les données XML.
Les feuilles de style XSL-T : templates, structures. Modes de programmation.
Le langage XSL-FO . Conversion en format type papier PDF, RTF...
- Message et échanges XML
 - DOM et l'interface de programmation normalisée : Data Island, Ajax, JavaScript.
Les serveurs d'applications : architecture en .NET et J2EE.
Portail d'entreprise.
Flash : FLEX, échanges XML et services Web.
Transfert et sérialisation de messages : Rest, XML-RPC, SOAP, WSDL, UDDI.
Définition des Services Web.
- XML et les bases de données
 - Le stockage de documents XML et bases semi-structurées.
Data Island.
SGBDR aux bases de données natives XML Xindice, Tamino.
Langages d'interrogation XML : XPath, XQuery, XLink, XPointer, extensions du SQL.
XML et les bases Oracle et SQL Server 2005.
Interfaçage de SQL Server avec IIS pour un accès direct par URL.
- Sécurité des échanges XML
 - Les différents protocoles de sécurisation : HTTPS, XML Signature (Xml Dsig) : signature numérique, XML encryption, XML access control, WS-security.
Interfaces graphiques Les nouveaux langages de conception graphiques
Le langage XAML de Microsoft.
Le langage XUL de l'Open Source.
SMIL : les animations Web.
XForms, InfoPath : la technologie de formulaires électroniques.
Blogs et RSS (Really Simple Syndication), ATOM un format de syndication de contenu Web.
SVG : standard vectoriel.
- XML au coeur de la bureautique
 - RDF, un cadre de définition de méta-données.
Offre Office de Microsoft. Open XML.
XML Open Document Format (ODF).
Open XML vs OpenDocument.
XML compression.

Catégorie 4 : administration serveur J2EE

Contexte d'utilisation

Déployez efficacement vos livraisons J2EE (WAR, EAR) sur un des serveurs suivant :

Formation TOMCAT (3j)

Présentation:

Avec Objis, maîtrisez l'architecture du serveur Tomcat, son administration sous tous ses aspects: l'installation, la configuration ainsi que le suivi et le tuning.. Les participants mettront en oeuvre de façon pratique l'interfaçage de Tomcat avec les services connexes (machine virtuelle Java, serveur Apache, authentification, bases de données). Les problématiques concernant le déploiement d'applications et la sécurisation sont abordées. Enfin, les architectures avancées permettant l'équilibrage de charges et la tolérance aux fautes sont également étudiées.

Durée/Tarifs:

3 jours

Inter : 1200 Euros HT

Intra : 4000 Euros HT (6 personnes)

Programme détaillé:

- **Introduction et installation**
 - Le projet Tomcat : historique et différentes versions
Rappel sur les applications Webs en Java : Contenu statique, dynamique, Servlets et JSPs, positionnement de Tomcat dans la norme J2EE, le modèle MVC
Installation : Choix de la JVM et installation, Téléchargement distribution binaire et installation, Scripts de démarrage et d'arrêt, Parcours de la distribution et des applications fournies.
- **Configuration**
 - Le fichier server.xml : Structure hiérarchique, présentation des différents éléments (Server, Service, Connector, ...)
Tomcat et JNDI : Déclaration de ressources JNDI, exemple d'une DataSource JDBC
Tomcat et l'authentification : realms, authentification web
Tomcat et la session utilisateur : Rappel du mécanisme de session, sérialisation des sessions utilisateur, les différents manager disponibles
Mise en place des logs d'accès
Interface d'administration : Parcours de l'application d'administration
- **Déploiement d'applications**
 - Les fichiers .war : Structure, construction, le descripteur de déploiement web.xml, le mapping de servlet.
Le déploiement de tomcat : Retour sur les descripteurs de contexte, les paramètres de configuration influant sur le déploiement, déploiement au démarrage et déploiement à chaud, Tomcat et JMX
Tomcat manager : Déploiement local ou distant avec le manager, déchargement ou rechargement de contexte
Automatisation des déploiements : Présentation de l'outil Ant, Exécution de commande du manager via Ant
- **La sécurité**
 - Introduction : Les différentes sources de danger, les techniques basiques de protection
Applications : Restreindre les accès aux ressources des applications web, le fichier catalina.policy
Attaques externes : Les différents types d'attaque, le filtrage de requête et les valves
Sécurisation par SSL : Présentation de SSL, mise en place dans Tomcat

- **Optimisation et mesures des performances**
 - Introduction : Méthodologie pour la mesure des performances, panorama des outils OpenSource
L'outil JMeter : Présentation des concepts, installation, enregistrement de scénarios de test, présentation des différentes mesures et visualiseurs
Optimisation : Types d'optimisation, paramètres de configuration influant sur les performances.
- **Intégration avec le serveur Apache**
 - Introduction : présentation des différentes techniques, avantages et inconvénients.
Apache comme Proxy : Configuration des deux serveurs
Utilisation de mod_jk : Configuration des deux serveurs
- **Architecture avancée**
 - Introduction : Le clustering, différents types et objectifs, équilibrage de charge, scalability, tolérance aux fautes et continuité de service, le déploiement dans un cluster, le farming
Mode instances multiple : les variables d'environnement CATALINA_HOME, et CATALINA_BASE, architecture
Equilibrage de charge et tolérance aux fautes : Problématique, les différentes solutions, utilisation de proxy et de rewriting, Apache comme répartiteur de charge, l'application balancer.
Support pour le clustering : Réplication et persistance de session, problématique limitations et différentes techniques, le farming dans tomcat

Formation JONAS (5j)

Présentation:

Découvrez la valeur ajoutée de ce serveur d'application Open source certifié J2EE, et développé par un consortium autour de BULL, FRANCE TELECOM et l'INRIA. Le serveur d'applications Jonas, bien que moins renommé que certains autres serveurs d'applications Open Source, permet aux entreprises de développer en toute liberté leurs applications Internet de nouvelle génération. A l'issue de cette formation, les participants sauront mettre en oeuvre le serveur d'applications Jonas.

Durée/Tarifs:

5 jours

Inter : 1900 Euros HT

Intra : 6000 Euros HT (6 personnes)

Programme détaillé:

- Présentation et installation
 - Architecture du produit
 - Procédures d'installation
 - Procédures de démarrage
 - Procédures d'arrêt du serveur
 - Intégration avec Tomcat
 - Organisation des répertoires sur le serveur
- Configuration de base
 - Activation et désactivation des différents services (JavaMail, JMS, JTA...)
 - JNDI et les datasources
 - Mécanismes de connexion aux bases de données : JDBC, pool de connexions
 - Présentation de JDO
 - Mapping XML
- La console d'administration
 - Mise en place de la console d'administration
 - Opérations courantes d'administration : déploiement d'une application, visualisation des logs...
- Déploiement d'application
 - Déploiement d'une application Web (war)
 - Déploiement d'une application J2EE (ear)
- Mise en place des droits utilisateurs
 - Gestion des authentifications et autorisations
 - Mise en oeuvre des realm
 - Les différents modes de stockage
 - Les droits utilisateurs (fichier plat, base de données, LDAP)
 - Sécurité : JAAS, rôles, groupes de permissions
- Administration avancée
 - Automatisation des tâches récurrentes d'administration par la mise en place de scripts
 - Connexions interapplicatives avec JCA
- Interfaçage avec Apache HttpServer
 - Installation du connecteur mod_jk
- Les performances / disponibilités
 - Equilibrage de charge au niveau http
 - Mise en place d'un cluster avec deux noeuds pour le serveur d'applications
 - Surveillance
 - Tuning
 - Analyse des fichiers de logs
- Spécificités de Jonas

- Joram
- Genic
- Carol
- Jonathan
- Jeremie
- Jotm
- C-JDBC

Catégorie 5 : tests et intégration continue

Contexte d'utilisation

Mettez en oeuvre les meilleures pratiques de développement agile et de construction d'application :

Formation TESTS (2j)

Présentation:

L'objectif de qualité qui s'impose dans toutes les activités de l'entreprise devient une préoccupation centrale dans le développement logiciel où la recherche du zéro défaut se décline en grande partie sous forme de tests. Le principe Lean suggérant de supprimer les défauts dès leur apparition incite à tester le code dès qu'il est écrit.

Durée/Tarifs:

Objis SAS au capital de 49500 € - RCS Lyon – SIRET 482 075 066 0027 – APE 804 C – TVA Intra FR 24 482075066 – agrément : 82.69.08812.69,
425 cours Emile Zola – 69100 Villeurbanne – 04 78 29 37 26 - Tél : 04 78 29 37 26 – Fax : 09 56 22 89 49 – contact@objis.com

2 jours

Inter : 800 Euros HT

Intra : 2500 Euros HT (6 personnes)

Programme détaillé:

- Le test dans le processus de développement
 - Processus, qualité, tests
Typologie des tests
- Tests unitaires automatisés avec Junit
 - Le besoin d'un Framework de test
Le Framework Junit
Cas et suites de tests avec Junit
Alternatives (TestNG) et outillage complémentaire
- Bonnes pratiques associées aux tests unitaires
 - Bonnes pratiques associées à Junit
Liste de tests
Tests isolés
Granularité raisonnable
Refactoring
- Écrire du code testable
 - Composition plutôt qu'héritage
Éviter le code statique
Isoler les dépendances
Inversion of Control
Code hérité (legacy)
- Les Mock Objects
 - Quand les créer
Types de Mock
Types d'implémentations
Bibliothèques de Mocks
Styles de test unitaire
- Couverture des tests
 - Les axiomes sur la couverture des tests
Types de couverture
Combien de tests faut-il écrire ?
Stratégies de test
Tests de régression
Outils de couverture
- Quelques problèmes liés au test dans des situations particulières
 - Tests Web
Tests EJB
Tests en présence de bases de données
Tests d'interface utilisateur

Tests et multi-threading

Formation SUBVERSION (1j)

Présentation:

Avec Objis, maîtrisez l'administration de votre outil de gestion de sources ! Successeur de CVS, Subversion est aujourd'hui le logiciel de gestion de configuration le plus utilisé.

Durée/Tarifs:

1 jour

Inter : 500 Euros HT

Intra : 1500 Euros HT (6 personnes)

Programme détaillé:

Présentation

- Concepts de travail en équipe

- Divisions des tâches par domaine fonctionnel
Problème des accès aux ressources partagées
Définition de périmètres partageables
- Historique et futur de Subversion
Relation à CVS
Panorama rapide des SCM (Source Code Management) concurrents et fonctionnalités spécifiques à Subversion
- Architecture de Subversion
Architecture des serveurs Subversion
Protocoles d'accès aux serveurs ou accès local
Notion de dépôt (repository)
Structure d'un repository
- Panorama des clients Subversion et critères de choix
- Concepts basiques de Subversion
 - Définition d'un repository
Définition d'un changes et d'une révision
Gestion des répertoires et des fichiers
Nommage par son URL de chaque ressource
Management des méta-données
Création de branches et de tags
- Utilisation au jour le jour
 - Importation initiale de contenu préexistant
Récupération du contenu d'un repository
Propagation d'une modification
Synchronisations et conflits
Résolution des conflits (merges)
Retrouver l'auteur d'une modification
- Administration des repositories
 - Savoir créer un repository
Organiser la structure de son repository de façon adaptée
Utiliser les outils d'administration et de monitoring
Savoir exporter et importer un repository
Vérifier le contenu d'un repository
Techniques de réparation d'un repository en cas d'incident
Savoir effectuer des backups à chaud
Savoir scripter les événements disponibles du repository (hook scripts)
Application à l'obligation d'avoir un message de commit non-vide
Automatisation de l'envoi de mails lors des commits
Application à certains types de sauvegardes automatisées
- Administration des serveurs
 - Présentation de SVNServe et d'Apache mod_dav_svn
Critères à évaluer lors d'un choix
Configuration d'un Apache mod_dav_svn
Configuration d'un SVNServe

Ajout d'une protection SSL aux échanges

Formation MAVEN (3j)

Présentation:

Ce cours est destiné à tous ceux qui se posent les questions suivantes : "Quels sont les bénéfices réels de l'intégration continue ?", "Quels sont les apports de Maven par rapport à des outils plus classiques comme ANT ?", ou de manière plus pratique "Comment mettre en oeuvre Maven comme outil de construction ?", ou "Quelle intégration entre Maven et des outils comme Subversion, ou encore le serveur d'intégration continue type Continuum ? » .

Durée/Tarifs:

3 jours

Inter : 1200 Euros HT

Intra : 4200 Euros HT (6 personnes)

Programme détaillé:

- Introduction

*Objis SAS au capital de 49500 € - RCS Lyon – SIRET 482 075 066 0027 – APE 804 C – TVA Intra FR 24 482075066 – agrément : 82.69.08812.69,
425 cours Emile Zola – 69100 Villeurbanne – 04 78 29 37 26 - Tél : 04 78 29 37 26 – Fax : 09 56 22 89 49 – contact@objis.com*

- Définition et intégration dans le cycle de vie des projets
Naissance de Maven : du script shell à Maven
Règle du « Convention over configuration »
- Le POM (Project Object Model)
 - Mécanisme de gestion des objets par Maven
Détail de la fiche projet : le fichier pom.xml
Le « super POM » et les mécanismes d'héritage
Exploration de la structure des projets Maven
Les types de projets Maven (Simple, Web, d'Entreprise, ...)
Notion de propriétés et de filtre des ressources
- Repository Maven et coordinations
 - Mécanisme de localisation d'une librairie
Notion de repository et d'architecture organisationnelle
Le repository local et les repositories distants
Outillage pour la gestion du cache et de la sécurité avec Archiva
Mise en place de Maven
Pré-requis
Installation et présentation des différentes techniques de configuration
- Modèle de gestion des dépendances
 - Exploration du mécanisme de gestion des dépendances et de transitivité
Optimisation des dépendances
Le cycle de construction d'un projet
Les différents cycles de construction
Comprendre les phases et le rattachement des goals Maven aux différents cycles
Exécution de plugins
Maven : une plateforme d'exécution de plugins
Comment configurer les plugins dans le but d'enrichir le cycle de construction?
- Les profils
 - Définition et intérêt des profils
Configuration et mise en oeuvre
Présentation des bonnes pratiques
- Développement de plugins
 - Création d'un plugin Maven en Java et attachement au cycle de construction
Extension pour la création d'un type de projet
Debugging de l'exécution d'un plugin dans un IDE
Cas de l'utilisation de Groovy comme langage alternatif pour l'écriture de plugins
- Archetype et Assemblies
 - Définition et spectre des archetypes existants
 - Génération de livrable à la demande
- La gestion de configuration
 - Communication avec un gestionnaire de sources
Fabrication d'un livrable et détails des pré-requis

- L'exécution des tests.
 - Organisation Maven pour les tests d'intégration et fonctionnels
 - Mise en œuvre du debugging des tests dans Eclipse lancés par Maven
 - Cas d'exemple de tests d'intégration s'exécutant au sein d'un navigateur Web démarré par Maven
- Les rapports
 - Maven permet de générer le site Maven
 - génération de rapports : Javadoc, couverture de code, détection de bugs, normes dev.
 - Principe d'une gestion pro active
 - Techniques de déploiement du site
- Maven dans les équipes de développement
 - Les différents modes d'utilisation de Maven avec les IDE Eclipse
 - Cas d'utilisation du plugin m2eclipse
- Intégration continue
 - Problématiques et enjeux
 - Outillages et mise en pratique avec Hudson
- Limites, avantages et futures évolutions
 - Comparaison entre Maven avec le nouvel outil de build Gradle
 - Gestion d'un build incrémental

Catégorie 6 : remise à niveau

Contexte d'utilisation

Les formations ci-dessus nécessitent des compétences Java/J2ee.

Remise à niveau JAVA (2j)

Nous consulter

Remise à niveau J2EE (2j)

Nous consulter

PARTIE 3 : Les cours du soir Objis

Soucieux de ne pas arrêter la relation avec nos clients comme fournisseur de formations d'appoint, et conscient des besoins réguliers d'expertise des consultants de SSII, Objis propose en 2009 une offre 'agile' contribuant à la formation continue de vos collaborateurs.

Format et principe

Les cours du soir Objis se présentent ainsi :

- Chaque mois :
 - une technologie JAVA/J2EE choisie parmi les compétences réclamées par le marché
 - une semaine dédiée à la montée en compétences de un ou plusieurs de vos collaborateurs sur cette technologie
 - Durant la semaine dédiée, 2 heures de formation intensive tous les jours , de 18h30 à 20h30.

Calendrier

Calendrier des thèmes du premier semestre 2009:

- Février : SPRING (9-13)
- Mars : EJB3 (16-20)
- Avril : HIBERNATE (13-17)
- Mai : MAVEN (25-29)
- Juin : UML/DESIGN PATTERN (22-26)

Avantages

- Un **véritable plan de formation** continue pour tous vos collaborateurs
- L'essentiel sur chaque technologie clé du marché -
- 70% de pratique.
- Evolution rapide vers rôles architecte / Chefs de projets techniques

Coût

- Coût par mois de formation : de 2500 à 4000 HT jusqu'à 5 collaborateurs.