



Formation GWT

Plan de cours détaillé

Objectifs:

Développer les compétences clés et les réflexes permettant d'intégrer un projet impliquant le framework GWT

Connaissances préalables :

Connaissance développement JAVA. Voir notre cours OBJFO1.

Profils participants :

Développeurs, Chefs de projets

Durée : 3 jours

<p>Introduction</p> <ul style="list-style-type: none"> ✦ Historique : lancement JavaOne 2006 ✦ Fondateurs : Bruce Johnson ✦ Positionnement : Java vers Javascript ! ✦ Concurrents : Flex, Dojo, jQuery, Jsf, etc... ✦ Ecosystème et outils connexes ✦ Nouveautés GWT 1.6 	<p>Gestion de projet</p> <ul style="list-style-type: none"> ✦ Cycle de vie d'un projet GWT ✦ Création du squelette de l'application avec WebAppCreator ✦ Développement d'un projet GWT dans Eclipse ✦ Séparation des sources client / sources serveur ✦ Travail en mode hébergé ✦ Cycle de débogage du code client ✦ Déploiement du projet et compilation GWT
<p>Architecture Application GWT</p> <ul style="list-style-type: none"> ✦ webAppCreator ✦ HTML, CSS ✦ Module XML ✦ EntryPoint JAVA ✦ fichier nocache.js ✦ Internationalisation . 	<p>Outillage</p> <ul style="list-style-type: none"> ✦ Mode hébergé GWT & bouton refresh ✦ Compilation : GWTCompiler ✦ Internationalisation : i18nCreator ✦ Tests unitaires : junitCreator ✦ Benchmark : benchmarViewer ✦ Plugin Eclipse ✦ Plugin Maven 1 / Maven 2
<p>Développement interfaces riches</p> <ul style="list-style-type: none"> ✦ Widgets simples (Boutons, liens, images, cases à cocher, textes...) ✦ Widgets complexes : Menus, Dialog, widgets Composites ✦ Panels : VerticalPanel, HorizontalPanel, FormPanel... ✦ Gestion d'évènements Handlers GWT (ClickHandler, KeyboardHandler...) ✦ Formulaire Ajax 	<p>Accès couche service</p> <ul style="list-style-type: none"> ✦ Architectures asynchrones de type Ajax ✦ Communication avec serveur J2ee : GWT-RPC ✦ Appels du service depuis le code client ✦ Sérialisation d'objets ✦ Gestion des exceptions ✦ Utilisation de JSON (JavaScript Object Notation) ✦ Servlet RemoteServiceServlet ✦ Interfaces RemoteService, RemoteServiceAsync
<p>Design Patterns et GWT</p> <ul style="list-style-type: none"> ✦ GWT et les Design pattern (DP) ✦ Mise en oeuvre DP Commande dans GWT ✦ Mise en oeuvre DP MVC ✦ Mise en oeuvre DP Observateur ✦ Mise en oeuvre DP Stratégie ✦ Mise en oeuvre DP DTO 	<p>Accès couche service</p> <ul style="list-style-type: none"> ✦ Architectures asynchrones de type Ajax ✦ Communication avec serveur J2ee : GWT-RPC ✦ Appels du service depuis le code client ✦ Sérialisation d'objets ✦ Gestion des exceptions ✦ Utilisation de JSON (JavaScript Object Notation) ✦ Servlet RemoteServiceServlet ✦ Interfaces RemoteService, RemoteServiceAsync

Tests <ul style="list-style-type: none">✦ Tests techniques et Tests IHM✦ Intégration GWT et Junit✦ Classe GWTestCase✦ Refactoring code en vue de test✦ Test asynchrone et Timer	Intégration framework <ul style="list-style-type: none">✦ Intégration GWT et Spring✦ Intégration GWT et JPA/Hibernate✦ Intégration GWT et JSF
JSNI <ul style="list-style-type: none"><input type="checkbox"/> Présentation JSNI (JavaScript Native Interface)<input type="checkbox"/> Atouts et limitations<input type="checkbox"/> Intégration GWT / Ext JS / Dojo<input type="checkbox"/> Emulation JRE	GWT et Maven <ul style="list-style-type: none">✦ Installation plugin Maven GWT✦ <code>goal gwt:gwt</code>✦ Mode hébergé, mode web✦ Création WAR✦ Automatiser tester et déploiement