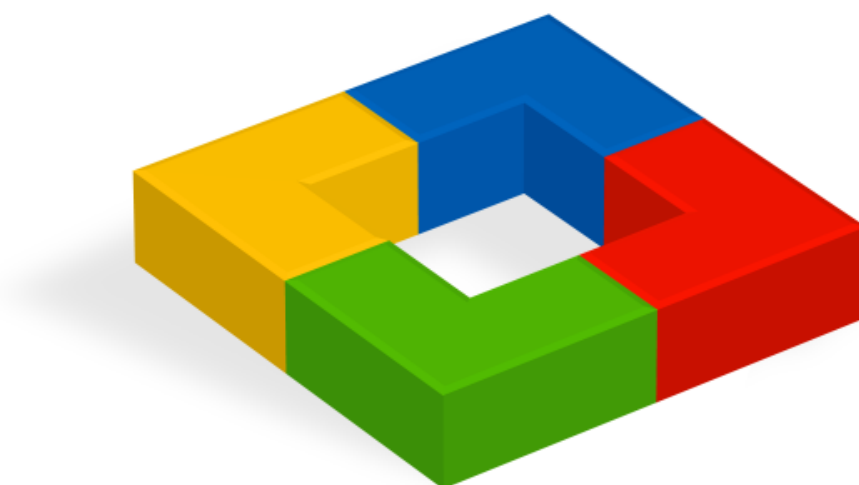


Guide de Prise en Main Rapide

CodeCharge Studio 2



EDITEUR :

YesSoftware, Inc.

6330 S. Eastern Ave., Suite 5

Las Vegas, NV 89119

Etats-Unis

Tél. : +1-888 241 7338

Fax : +1-866 312 8049

www.codecharge.com

DISTRIBUTEUR FRANCE :

KAPITEC SOFTWARE S. A.S.

57, Bld de l'Embouchure

31075 Toulouse Cedex 2

France

Tél. : +33-5 34 27 90 03

Fax : +33-5 34 27 90 04

www.kapitec.com

Version : 1.0

Date : 26/11/2003

1. Table des Matières

1.	Table des Matières	2
2.	Introduction	4
3.	Exemples d'Utilisation de CodeCharge Studio	6
4.	Qui Devrait Utiliser CodeCharge Studio ?	7
4.1.	Développeurs d'Applications Web	7
4.2.	Développeurs de Bases de Données	7
4.3.	Programmeurs	7
4.4.	Webmestres et Concepteurs	7
4.5.	Intégrateurs Systèmes et Revendeurs à Valeur Ajoutée	7
5.	Pré-Requis Système	8
5.1.	Environnement de Développement	8
5.2.	Bases de Données Supportées	8
5.3.	Environnement de Déploiement (Serveur)	8
6.	Comment utiliser CodeCharge Studio ?	10
7.	Moteur de Génération de Code	11
8.	Caractéristiques Clefs et Bénéfices de cette Technologie	13
8.1.	Productivité et Développement Rapide d'Applications	13
8.1.1.	Générateur d'Applications	14
8.1.2.	Environnement de Développement Intégré (IDE)	14
8.1.3.	Générateurs de Formulaires et Composants	15
8.1.4.	Générateur Visuel de Requêtes	16
8.1.5.	Gestion de Sécurité/Contrôle d'Accès	17
8.1.6.	Diagrammes de Site	17
8.1.7.	Support de Multiples Langages de Programmation	18
8.1.8.	Extensibilité	18
8.1.9.	Connexions à de Multiples Bases de Données	18
8.1.10.	Intégration à Microsoft FrontPage	19
8.1.11.	Séparation du Code de la Conception	19
8.1.12.	Support Important de Bases de Données	20
8.1.13.	Templates d'Applications (Solutions)	20
8.1.14.	Code sur Mesure et Actions	21
8.1.15.	Génération de Code Rapide et Pratique	22
8.1.16.	Documentation	22
9.	CodeCharge Studio Edition Personnel	23
10.	Interface Locale	24
11.	Roadmap Produit	25
12.	Plus d'Informations	26
12.1.	Liens Utiles	26
12.1.1.	Base de Connaissance	26
12.1.2.	Documentation	26
12.1.3.	Mise à Jour	26
12.1.4.	Tutorial	26
12.2.	Support	26
12.3.	Autres Questions	26
13.	Annexe - Aperçu de l'Implémentation de l'Architecture Java	27
13.1.	Flux de Contrôle de Servlet Java	27
13.2.	Modèle JSP et XML	28
13.3.	Approches	29
13.3.1.	Model-View-Controller (MVC)	29
13.3.2.	Pooling de Connexion	29
13.3.3.	Événements	29

13.3.4.	Couche Accès aux données	29
13.3.5.	Templates	30
13.3.6.	Modèle XML	30
13.4.	Fonctionnalités	30
13.4.1.	Enregistreur (Logger)	30
13.4.2.	Extension Source de Données JDBC	30
13.4.3.	Formatage - Validation de Données	30
13.4.4.	Passage de Paramètres	30
13.4.5.	Valeurs par Défaut et Expressions	30
13.4.6.	Construction Automatique de Requêtes SQL Basées sur des Valeurs de Paramètres	30
13.4.7.	Actions Pré-Définies	31
13.5.	Composants	31
13.5.1.	Contrôles Web	31
13.5.2.	Marqueurs	31
13.5.3.	ControllerServlet	31
13.5.4.	Classe Abstraite Action	31
13.5.5.	Classe Abstraite View	31
13.5.6.	Classe JDBCConnection	31

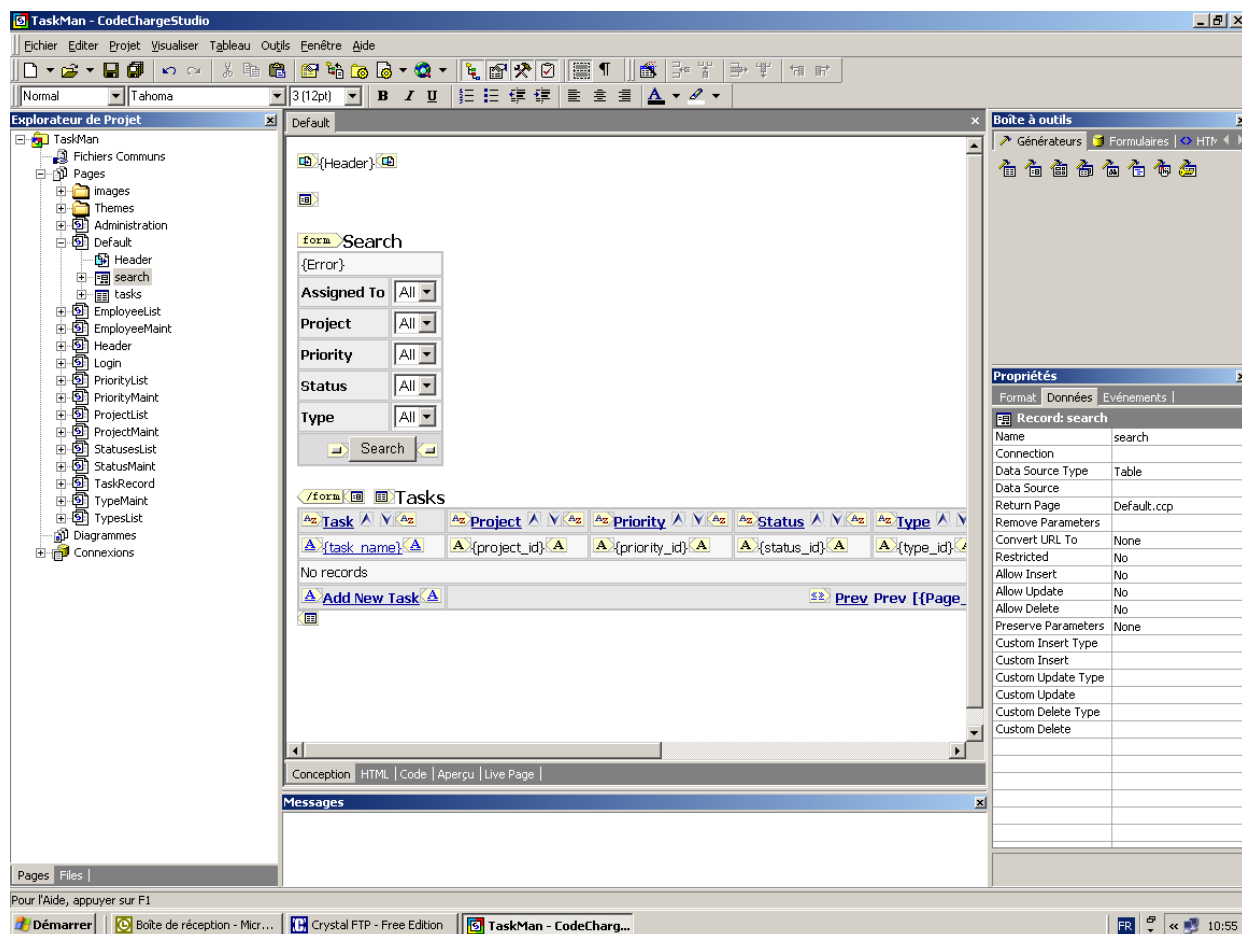
2. Introduction

Issu de la technologie CodeCharge, CodeCharge Studio 2 est un générateur visuel d'applications et un générateur de code, dédié au Développement Rapide d'Applications (RAD) pour le Web. C'est la meilleure solution de productivité aujourd'hui disponible sur le marché pour créer rapidement des applications Web puissantes, extensibles et sécurisées. Basé sur un moteur XSL sophistiqué, utilisant les formats de fichiers XML, CodeCharge Studio 2 offre de nouvelles possibilités aux développeurs Web : automatisation de la création de quasiment tout composant d'applications Web, et génération d'un code robuste côté serveur dans de nombreux langages de programmation (ASP.NET - C# et VB.Net -, ASP 3.0, PHP 4.0, Java Servlets 2.2, JSP 1.1, ColdFusion 4.01 et Perl 5).

Avec toute la puissance de la technologie et de l'environnement de développement de CodeCharge Studio, les utilisateurs peuvent tirer profit d'un modèle de données centrales qui sépare le HTML du code côté serveur. Le code généré peut ainsi être aisément modifié en utilisant l'éditeur de code interne, qui verrouille les blocs de code modifiés, ceci afin d'empêcher la réécriture ultérieure des modifications de l'utilisateur lors de nouvelles générations de code (l'édition dans les deux sens s'en trouve facilitée). CodeCharge Studio inclut la Gestion de Sécurité/Contrôle d'Accès basé sur le Rôle. CodeCharge Studio est également entièrement intégré à Microsoft FrontPage; cette fonctionnalité intéressera les développeurs qui utilisent FrontPage pour la conception graphique Web.

De puissantes caractéristiques visuelles, un environnement entièrement extensible, et le support de quasiment toutes les bases de données et technologies serveur les plus utilisées font de CodeCharge Studio 2 l'environnement de développement d'applications Web idéal.

Interface Utilisateur de CodeCharge Studio



3. Exemples d'Utilisation de CodeCharge Studio

CodeCharge Studio peut être utilisé pour créer une grande variété d'applications Web se connectant à des bases de données, en passant par les Répertoires d'Employés jusqu'à des Portails ou des Systèmes de Gestion de Contenu beaucoup plus complexes. Voici certains exemples sur la manière dont cette technologie a été utilisée :

- ◆ **IBM Australie** a utilisé CodeCharge pour construire un Intranet.
- ◆ **Gravitymax** a utilisé CodeCharge pour créer ses Systèmes de Gestion de Contenu et de Gestion de Relation Client (CRM) - <http://www.gravitymax.com.au>
- ◆ **UltraApps** a utilisé CodeCharge pour créer l'un de ses systèmes Web les plus connus de Localisation de Défauts et de Gestion de Problèmes, ainsi qu'un Portail pour des clubs et des organisations à but non-lucratif - <http://www.ultraapps.com>
- ◆ **Goodrich Corporation** a utilisé CodeCharge pour générer plus d'un million de lignes de code pour leurs applications de Gestion de Base de Connaissance, Gestion de Configuration, HelpDesk/Service Clients, Fiche de Données Fournisseurs, entre autres. Goodrich utilise également CodeCharge Studio pour créer une interface avec leurs clients afin de suivre la fiabilité et les performances de Goodrich.
- ◆ **Sony Corporation** a utilisé CodeCharge pour créer une interface Web avec leurs applications WebLogic.

4. Qui Devrait Utiliser CodeCharge Studio ?

CodeCharge Studio est parfaitement adapté aux groupes d'utilisateurs décrits ci-dessous.

4.1. Développeurs d'Applications Web

CodeCharge Studio fournit aux développeurs toutes les fonctionnalités nécessaires pour créer rapidement des applications Web puissantes, extensibles et sécurisées. Les développeurs peuvent se focaliser sur la programmation de la logique d'affaires indispensable derrière l'application, et non pas sur le framework applicatif initial, ce qui élimine ainsi les schismes qui se produisent souvent dans le processus de développement. Ce groupe d'utilisateurs comprend les développeurs en entreprise, les sociétés de services en informatique, les programmeurs indépendants et les consultants.

4.2. Développeurs de Bases de Données

Les développeurs de bases de données qui sont à l'aise avec les outils visuels de bases de données et de conception d'applications apprécieront le modèle de données centrales proposé par CodeCharge Studio. Quant aux programmeurs Microsoft Access, ils trouveront l'interface de CodeCharge Studio très familière.

4.3. Programmeurs

Les architectes de systèmes avancés et sophistiqués sont rarement familiers avec le développement d'applications Web. Visually Studio ou d'autres environnements de programmation Windows sont souvent trop complexes et ne proposent pas de solution simple pour comprendre le développement d'applications Web. CodeCharge Studio simplifie et accélère le processus de développement Web au travers d'un éventail de caractéristiques incluant des composants de glisser-déposer (*drag-and-drop*), des générateurs d'applications, et des exemples d'applications pré-construites.

4.4. Webmestres et Concepteurs

Les Webmestres et les concepteurs savent combien il est nécessaire de stocker le contenu Web dans une base de données et les avantages que cela procure, et plus encore lorsqu'ils sont amenés à maintenir des centaines de pages Web. On s'attend souvent à ce qu'ils produisent un contenu dynamique, tel qu'un répertoire d'employés ou un portail pour leur société. CodeCharge Studio adresse cette problématique et permet à n'importe qui de créer des applications se connectant à des bases de données sans avoir besoin de programmer beaucoup. Pour les concepteurs souhaitant conserver leurs outils de conception, CodeCharge Studio est parfaitement intégré à Microsoft FrontPage, et peut également être utilisé avec quasiment tout autre outil de conception.

4.5. Intégrateurs Systèmes et Revendeurs à Valeur Ajoutée

Les sociétés qui produisent, vendent ou maintiennent des applications logicielles trouveront en CodeCharge Studio un outil inestimable pour créer des interfaces Web pour leurs solutions. Par exemple, un fournisseur de logiciel CRM offrant une solution Windows peut maintenant facilement créer l'équivalent Web de son produit ou au minimum les parties Web de l'application et des fonctionnalités associées.

5. Pré-Requis Système

5.1. Environnement de Développement

CodeCharge Studio 2 est disponible pour les configurations suivantes :

- ◆ Processeur Intel Pentium ou équivalent
- ◆ 64 Mo RAM
- ◆ 30 Mo disque
- ◆ Ecran 256 couleurs/Résolution 800 x 600
- ◆ Windows 95, 98, ME, NT4, 2000 ou XP
- ◆ Microsoft Internet Explorer 5.5 ou version supérieure

5.2. Bases de Données Supportées

CodeCharge Studio peut se connecter à quasiment toute base de données conforme ODBC, en utilisant les drivers appropriés installés sur la machine de développement sur laquelle la licence CodeCharge Studio est installée.

Les bases de données supportées par CodeCharge Studio incluent :

- ◆ Microsoft SQL
- ◆ Oracle
- ◆ MySQL
- ◆ Microsoft Access
- ◆ DB2
- ◆ Informix
- ◆ Interbase
- ◆ mSQL
- ◆ Sybase
- ◆ PostgreSQL

Le code généré ne requiert pas de connectivité ODBC et se connecte à la base de données en utilisant la méthode appropriée pour l'environnement donné (par exemple ADO pour ASP, JDBC pour Java, PHPLib pour PHP, ou DBI pour Perl).

5.3. Environnement de Déploiement (Serveur)

Le code généré s'exécute sur tous les systèmes d'exploitation avec les serveurs d'applications capables d'exécuter les versions des langages de programmation suivants :

- ◆ ASP 3.0 (s'exécute sur des serveurs IIS et PWS avec VBScript 5)
- ◆ ASP.NET 1.0 (C# ou VB.Net)

- ◆ ColdFusion 4.5
- ◆ PHP 4.0
- ◆ JDK 1.3
- ◆ Perl 5

6. Comment utiliser CodeCharge Studio ?

Fonctionnellement et visuellement, CodeCharge Studio s'apparente aux outils RAD tels que Microsoft Access et Visual Studio, avec la capacité de maximiser la productivité des développeurs tout en minimisant le coût de développement. Alors que les outils visuels de développement sont traditionnellement utilisés pour développer des applications bureau ou client, CodeCharge Studio est parfaitement adapté au développement d'Applications Web sophistiquées.

CodeCharge Studio intègre de nombreuses caractéristiques très utiles aux développeurs dans le processus de construction d'applications Web. Cela passe par les Générateurs qui construisent les pages et les formulaires, les Composants de données *intelligentes* et les Contrôles qui peuvent être placés sur une page.

Les étapes ci-dessous montrent comment créer une application de base en utilisant le Générateur d'Applications, qui crée automatiquement des applications Web basées sur des tables de bases de données.

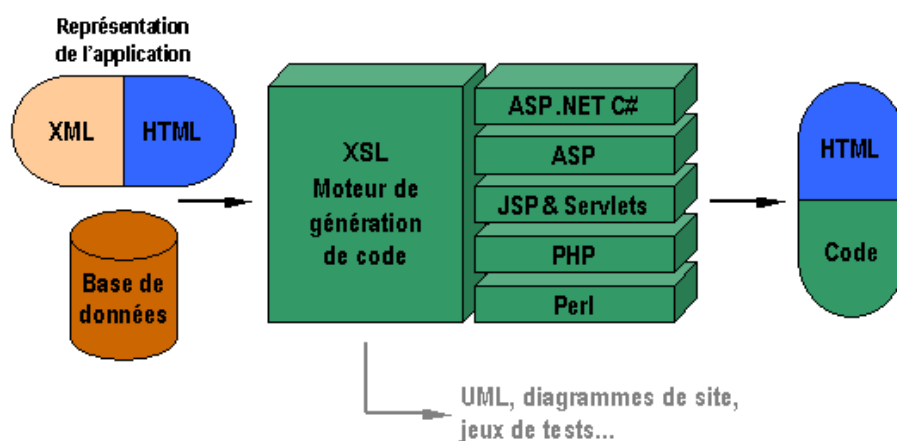
1. Démarrez CodeCharge Studio.
2. Démarrez un **Nouveau Projet**.
3. Nommez le **Projet** et sélectionnez **Générateur d'Applications**.
4. Spécifiez les paramètres du projet :
 - ⇒ le langage de programmation
 - ⇒ chemin d'accès au serveur Web...
5. Spécifiez le type de connexion base de données et établissez la connexion (alternativement, vous pouvez sélectionner un des exemples de bases de données : *Intranet* ou *Internet*)
6. Configurez l'authentification du site en spécifiant la table de la base de données et les colonnes qui contiennent les informations sur le login utilisateur et le mot de passe comme les groupes de sécurité.
7. Sélectionnez les tables de bases de données sur lesquelles l'application Web sera créée.
8. Spécifiez le type de page Web à créer pour chacune des tables de bases de données précédemment sélectionnées. Spécifiez si vous voulez créer des pages **Recherche & Grille** et **Enregistrement Maintenance** pour chaque table
9. Spécifiez un **thème** (style) à appliquer au site.
 - ⇒ Le thème définit les fontes et les couleurs pour l'application.
10. Le Générateur d'Applications crée alors automatiquement un projet avec des ensembles de pages qui constituent l'application Web. Vous pouvez alors publier les pages sur le serveur et les tester.
11. Si vous le souhaitez, vous pouvez faire des modifications à l'application créée en utilisant l'IDE (Environnement de Développement Intégré) de CodeCharge Studio.

7. Moteur de Génération de Code

Les utilisateurs de CodeCharge Studio créent des applications Web se connectant à une (ou des bases) de données en plaçant des formulaires de données *intelligentes* et des composants sur une page. Pendant le processus de conception d'une page, la représentation du projet et de toutes les pages afférentes sont sauvegardées dans un format XML en plus du code HTML, qui contient la conception de la page. Pendant la publication du projet, les fichiers HTML sont copiés sur le serveur alors que les fichiers XML sauvegardés contiennent le modèle du projet pour lequel CodeCharge Studio génère le code côté serveur via l'utilisation de templates de code XSL.

Cette architecture flexible permet aux utilisateurs de régénérer leurs applications Web dans n'importe quel langage de programmation et à n'importe quel moment. Par exemple, les utilisateurs peuvent générer une application C# et ensuite la régénérer en Java. Des utilisateurs plus expérimentés peuvent même créer leurs propres templates UML, qui généreront automatiquement la documentation et les diagrammes UML de leurs applications.

Les versions supérieures de CodeCharge Studio (en cours de développement) permettront la génération de ce type de documents, ainsi que les scripts de tests automatisés pour tester les applications générées.



Sur le serveur, le code généré est entièrement séparé du HTML. Lors de l'exécution, le code charge les fichiers HTML en mémoire, remplace le contenu statique par les valeurs provenant de la base de données, et les présente au navigateur au format HTML. Cette méthode consistant à séparer le code du contenu HTML permet aux concepteurs de procéder à des changements ultérieurs sur le site sans avoir à retoucher le code.

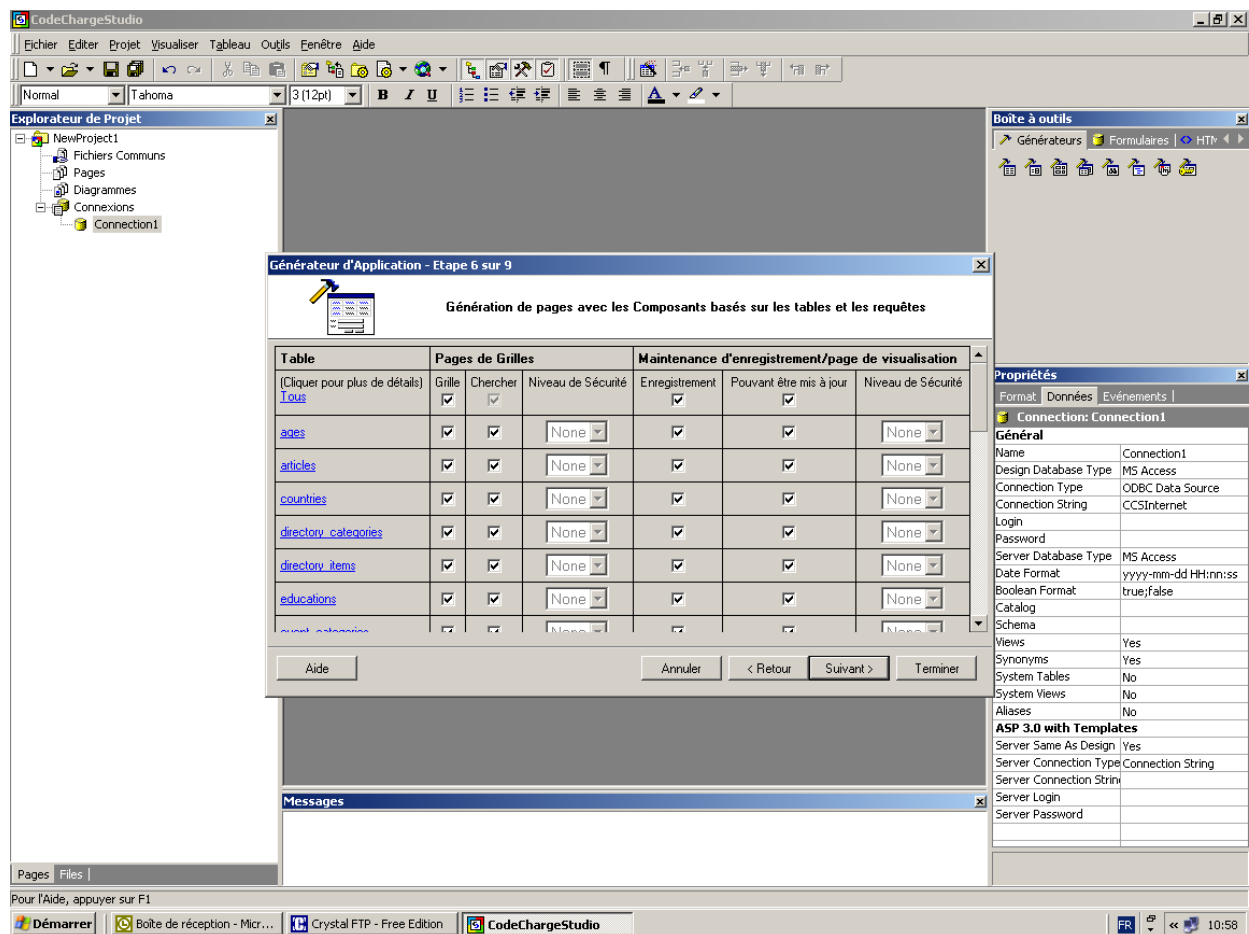
8. Caractéristiques Clefs et Bénéfices de cette Technologie

8.1. Productivité et Développement Rapide d'Applications

CodeCharge Studio génère un code orienté -objet structuré, facile à maintenir et à étendre. CodeCharge Studio rend la génération de code très pratique en éliminant la phase fastidieuse du processus de génération de code professionnel. Les développeurs peuvent facilement modifier n'importe quelle partie du code généré en utilisant un puissant éditeur de code inclus dans CodeCharge Studio. Le code ayant été modifié avec l'éditeur est verrouillé pour ne pas être réécrit lors des prochaines générations de code.

CodeCharge Studio offre une solution sans précédent aux utilisateurs pour construire rapidement de puissantes applications Web se connectant à des bases de données. En pointant et en cliquant sur l'interface, les utilisateurs peuvent sélectionner les composants, le langage de programmation, les schémas de présentation et les requêtes de bases de données (sans programmation ou sans avoir à comprendre complètement les technologies Web sous-jacentes). Ce qui rend CodeCharge Studio unique en son genre, c'est sa rapidité même. Par exemple, CodeCharge Studio rend possible la création d'un système simple de gestion de tâches en moins de cinq minutes. De nombreuses templates d'applications pré-construites sont incluses dans le produit et peuvent être adaptées aux besoins du développeur et étendues dans des systèmes Web riches en fonctionnalités.

8.1.1. Générateur d'Applications



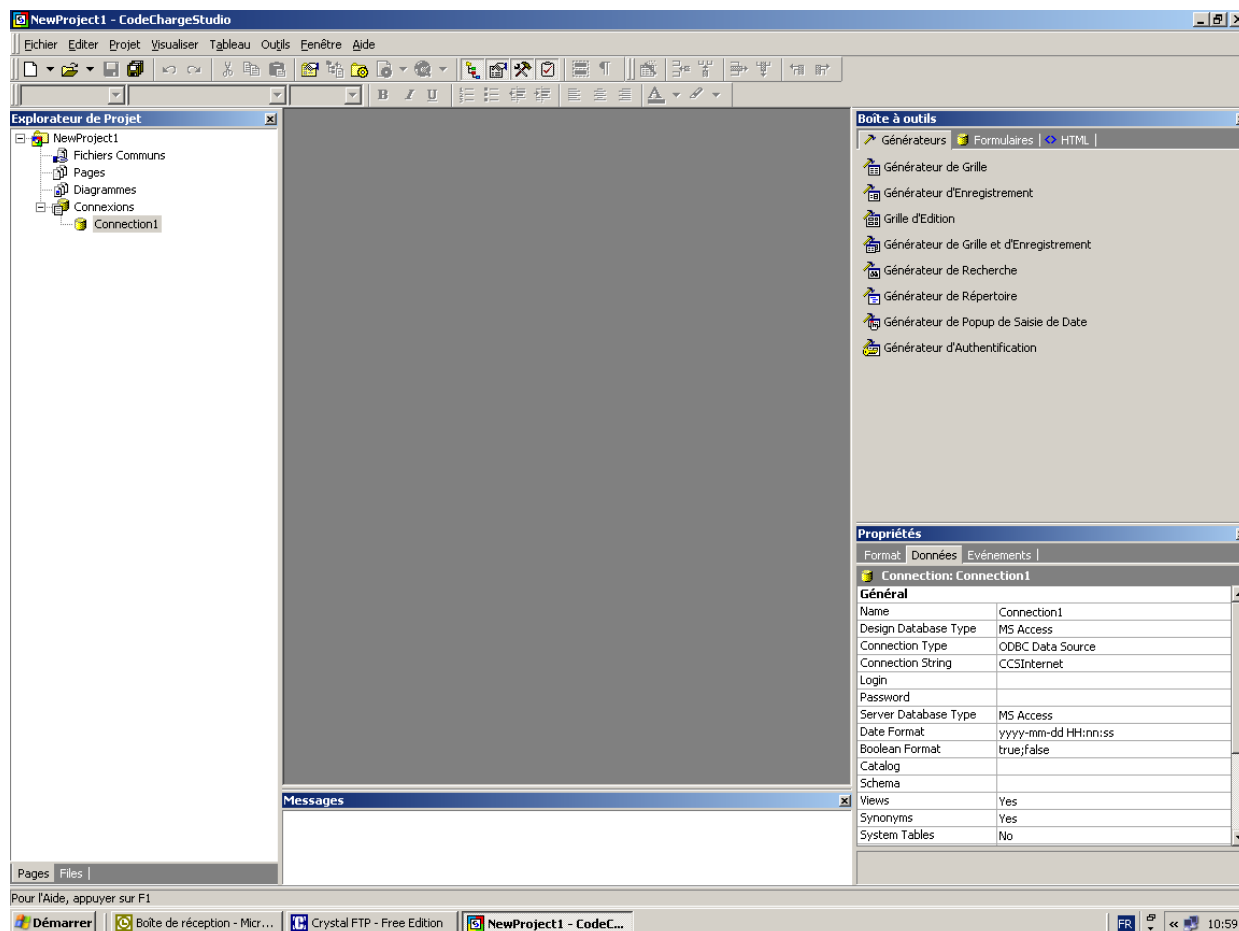
Le Générateur d'Applications est vraisemblablement l'outil le plus avancé d'automatisation de développement Web disponible pour les développeurs Web. En seulement quelques clics souris, les utilisateurs peuvent créer des applications Web complexes se connectant à des bases de données avec des dizaines, voire des centaines de pages Web pour la gestion et l'administration de données. Par exemple, un utilisateur avec une base de données d'employés, de projets et de tâches, peut utiliser le Générateur d'Applications pour créer automatiquement une page Liste des Tâches, une page Recherche d'Employés, une page Maintenance Employés et une page Maintenance Projet, et achever un projet applicatif complet en quelques minutes seulement.

8.1.2. Environnement de Développement Intégré (IDE)

CodeCharge Studio se caractérise par de puissants éditeurs HTML et de code, des composants de conception graphique HTML, un explorateur de projet, un navigateur de propriétés, des fenêtres d'outils, et bien d'autres fonctionnalités utiles aux développeurs professionnels. CodeCharge Studio permet aux utilisateurs de contrôler complètement le code généré au travers d'un éditeur de code complet et personnalisable, avec la mise en évidence de la coloration de syntaxe et avec une indentation pertinente, qui facilite la distinction des différents éléments du code source. Les développeurs peuvent modifier n'importe quelle partie du code sans que ce qui a été personnalisé (exemple coloration pour la syntaxe) ne soit écrasé/perdu lors de la génération de code. Cela signifie en clair que les utilisateurs ont un contrôle complet de leurs programmes et qu'ils n'ont pas besoin d'avoir recours à des éditeurs de code externes.

L'éditeur HTML intégré permet aux utilisateurs de concevoir (graphiquement) le site, ou de modifier les formulaires produits par le Générateur.

8.1.3. Générateurs de Formulaires et Composants



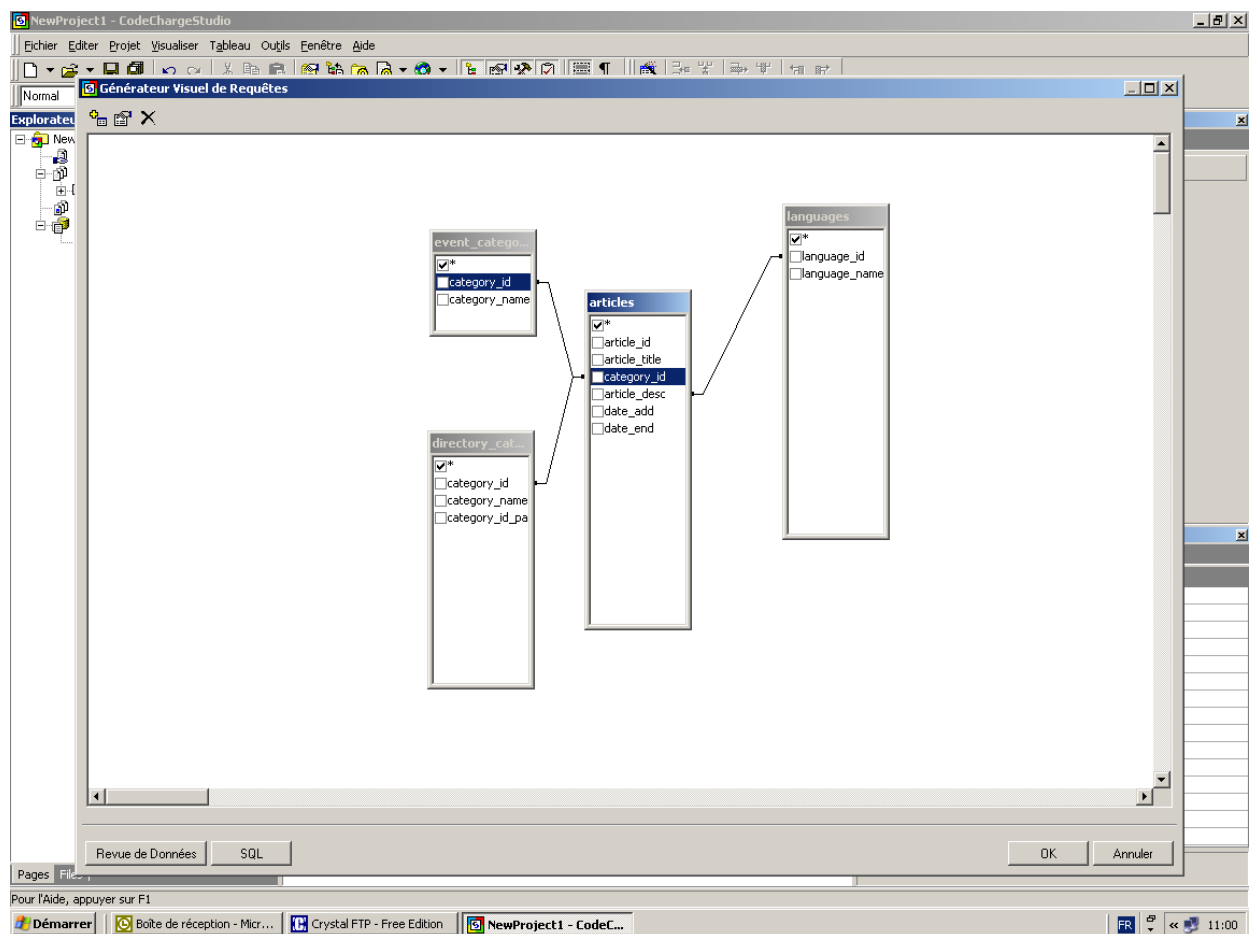
Les Générateurs de Composants sont les éléments clés de CodeCharge Studio : ils permettent aux utilisateurs d'assembler des pages Web (composants tels que Grille, Grille Editable, Enregistrement, Répertoire, Login, Recherche ou encore Menu). Un Générateur est un assistant dynamique qui crée le composant approprié sur la page en permettant au développeur de spécifier la table de la base de données et les champs qui sont utilisés pour l'affichage, la validation ou l'acceptation d'informations. Les Générateurs peuvent également être utilisés pour créer des composants avancés tels que le composant Upload de Fichier (utilisé pour charger des fichiers depuis une machine distante à un endroit indiqué sur le serveur Web). Ce composant peut être utilisé pour que les utilisateurs chargent des fichiers sur le serveur, par exemple dans le cas où les utilisateurs sont autorisés à soumettre des images.

CodeCharge Studio 2 introduit une nouveauté : le Générateur de Formulaire Répertoire. Ce générateur est spécialement dédié pour afficher un contenu de manière hiérarchique. Le contenu qui est affiché dans un Formulaire Répertoire est stocké de manière à ce qu'il y ait des éléments de niveau supérieur avec des niveaux inférieurs. Le Formulaire Répertoire implémente un *drill down* d'informations commençant sur les catégories parent et continuant sur des sous-catégories successives. Le Formulaire Répertoire produit des rendus visuels similaires au répertoire du site Yahoo ou de bien d'autres moteurs de recherche.

Yellow Pages Directory

Main		
Automotive Tires Rental More...	Education and Instruction Colleges and Universities K-12 More...	Health and Medicine Fitness Mental Health More...
Community Libraries Disabilities More...	Entertainment and Arts Entertainers Bars, Pubs, and Clubs More...	Home and Garden Lawn and Garden Appliances More...
Computers and Internet Computer Training Desktop Publishing More...	Food and Dining Catering Restaurants More...	Legal and Financial Taxes Arbitration and Mediation More...

8.1.4. Générateur Visuel de Requêtes



Les développeurs peuvent facilement choisir des tables et des procédures stockées ou du code SQL personnalisé comme source de données pour les formulaires de données. Un Générateur Visuel de Requêtes est fourni pour permettre la création visuelle de requêtes SQL. Par exemple, un utilisateur peut créer une requête qui affiche la liste des employés, avec les services et les fonctions associés. Les développeurs peuvent également spécifier des critères de filtrage pour par exemple n'afficher que les employés dont les noms correspondent aux critères de recherche saisis dans une autre page ou un autre formulaire.

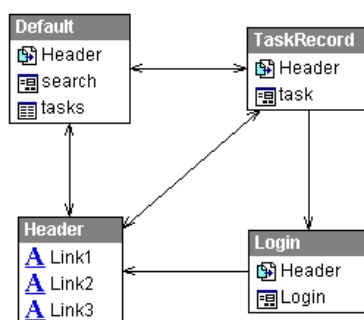
8.1.5. Gestion de Sécurité/Contrôle d'Accès



Une autre particularité de CodeCharge Studio est qu'il permet de protéger les applications Web générées en implémentant des fonctionnalités supplémentaires de sécurité empêchant les utilisateurs de faire des modifications externes aux formulaires et de soumettre des données invalides au serveur. CodeCharge Studio permet donc aux développeurs de définir des accès sécurisés basés sur des rôles sur chaque page ou chaque formulaire. Les utilisateurs Web qui n'ont pas les privilèges requis pour l'accès à une page sont automatiquement redirigés sur la page de connexion. Les utilisateurs qui sont autorisés à accéder à une page mais non autorisés à visualiser ou à mettre à jour un formulaire spécifique voient la page sans le formulaire concerné, et ne voient pas les boutons Insertion/Mise à Jour/Suppression requis pour mettre à jour le formulaire.

8.1.6. Diagrammes de Site

Pour n'importe quel projet existant dans CodeCharge Studio, les utilisateurs peuvent créer un ou plusieurs diagrammes de site : représentation visuelle de toutes les pages du projet ou seulement de certaines pages. Un diagramme de site peut être considéré comme un plan de pages au sein du projet avec des liens associés aux pages.



Les diagrammes de site étant visuels, cela les rend parfaitement adaptés pour créer la documentation du site ou encore des présentations concernant le site.

8.1.7. Support de Multiples Langues de Programmation

CodeCharge Studio génère du code dans tous les langages de programmation côté serveur, listés ci - après : C#, VB.Net (ASP.NET), VBScript (ASP), ColdFusion, Java (Servlets ou JSP), PHP et Perl. Parce que le modèle du projet est stocké dans un format XML, le langage de programmation peut être changé à tout moment, et le même projet peut être régénéré dans de nombreux langages.

8.1.8. Extensibilité

L'architecture de CodeCharge Studio est ouverte et entièrement extensible. La plupart des fonctionnalités sont implémentées en HTML, JavaScript, XML et XSL. Les utilisateurs peuvent personnaliser pratiquement n'importe quelle partie des fonctionnalités intégrées ou même développer leurs propres composants, comme des Générateurs, Composants, Actions, Thèmes, etc. Dans un avenir proche les utilisateurs pourront obtenir le SDK (Software Development Kit) de CodeCharge et alors développer leurs propres templates de génération de code.

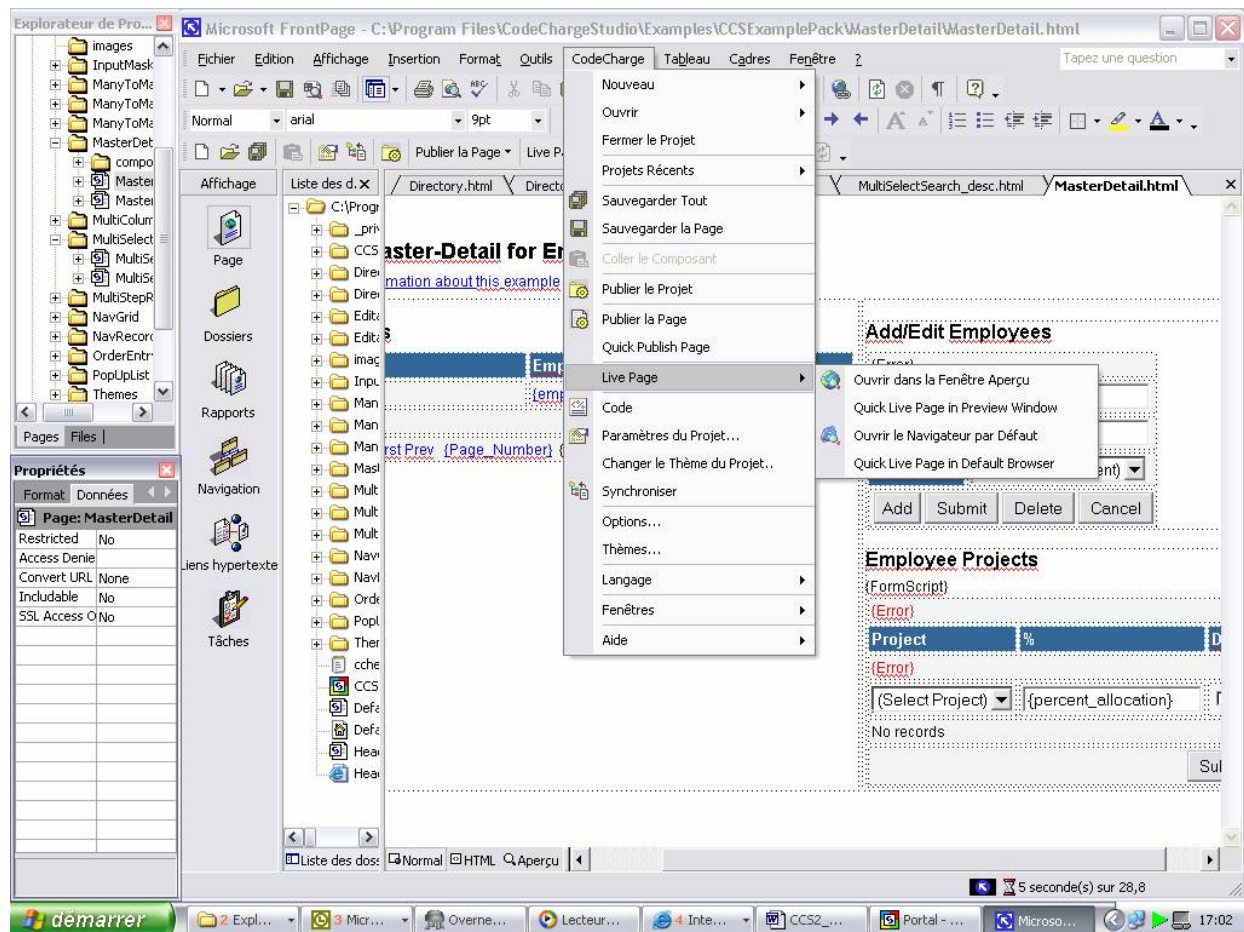
8.1.9. Connexions à de Multiples Bases de Données



CodeCharge Studio¹ supporte de multiples connexions à des bases de données permettant ainsi une application Web d'utiliser de nombreuses sources de données, pour par exemple valider l'accès utilisateur d'une base de données externes.

¹ CodeCharge Studio Edition Personnel permet la connexion à une SEULE base de données.

8.1.10. Intégration à Microsoft FrontPage

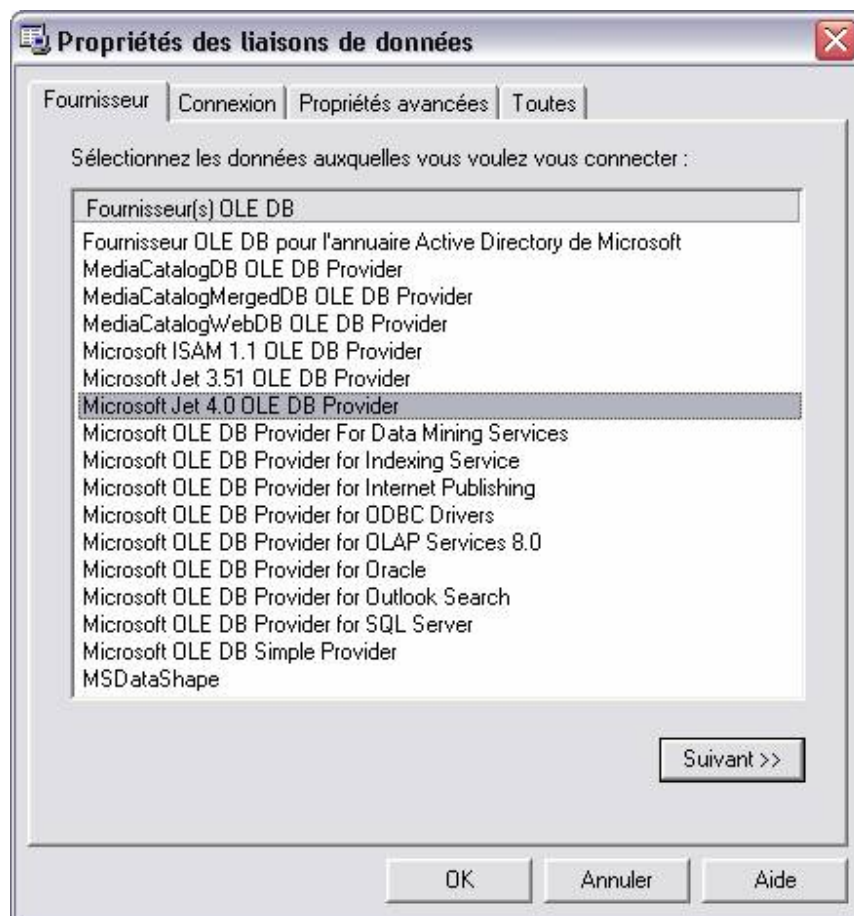


CodeCharge Studio peut être installé en tant que Add-In (module ajouté) à Microsoft FrontPage. Ce module étend les fonctionnalités de FrontPage en fournissant un ensemble de générateurs de code, un explorateur de projet, un navigateur de propriétés et un éditeur de code. Les utilisateurs de FrontPage peuvent alors utiliser toutes les fonctionnalités de CodeCharge Studio directement dans leur environnement de conception avec lequel ils sont familiarisés.

8.1.11. Séparation du Code de la Conception

CodeCharge Studio supporte complètement la séparation (sur option) du code de la conception en générant des fichiers code source (.asp, .php, .jsp, etc.), qui sont entièrement séparés des fichiers HTML correspondants (HTML templates). Cette approche permet aux équipes Web d'utiliser les meilleures pratiques : les concepteurs peuvent travailler séparément et en parallèle avec les développeurs. Beaucoup d'autres technologies obligent les développeurs à exécuter des tâches de conception, et les concepteurs à traiter le code. L'approche de CodeCharge Studio permet au développeur de produire des applications qui pourront être ultérieurement personnalisées en modifiant les fichiers templates HTML au sein de l'IDE de CodeCharge Studio ou en utilisant un environnement de conception approprié, et tout cela en laissant le code intact.

8.1.12. Support Important de Bases de Données

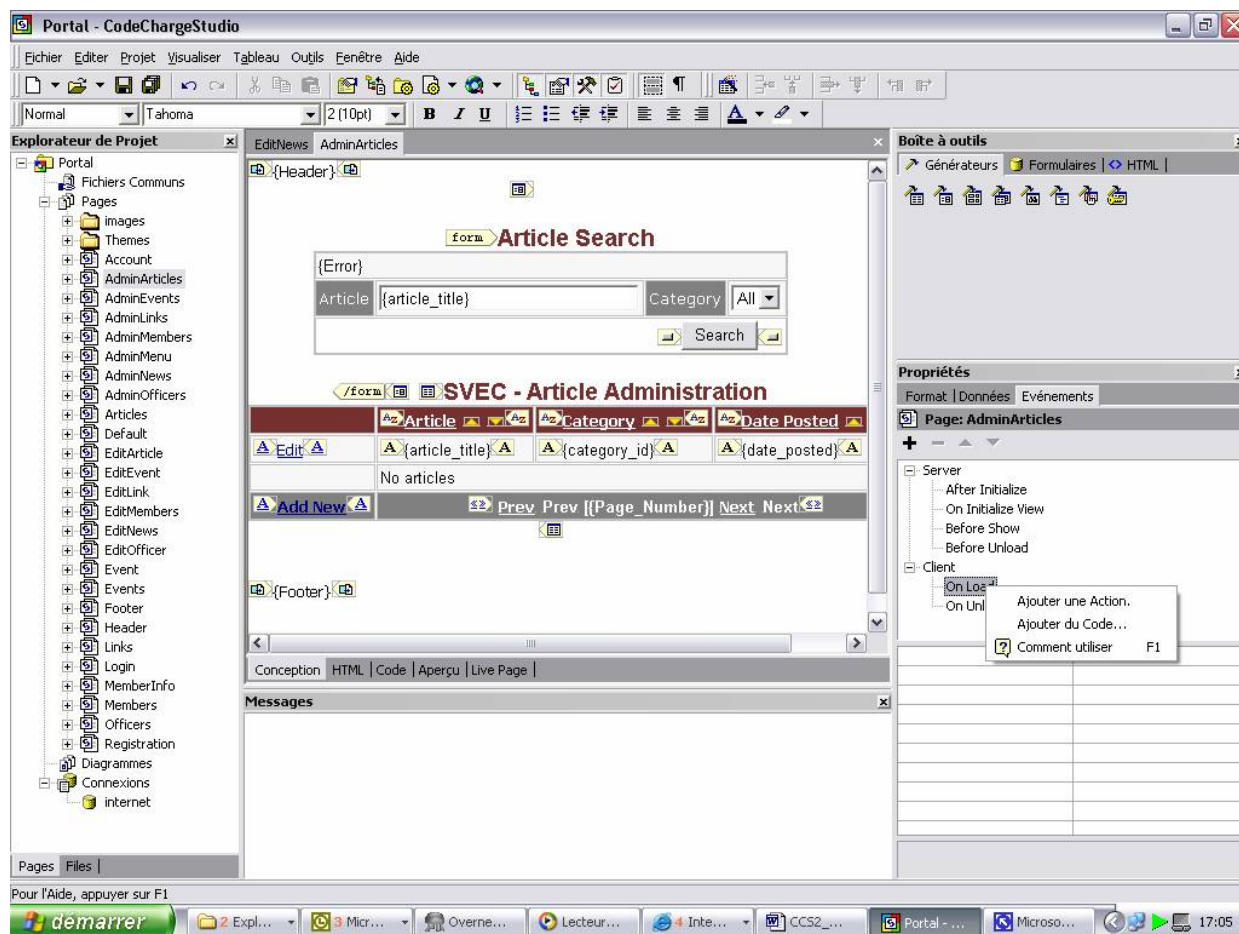


CodeCharge Studio supporte la plupart des bases de données existantes via JET et ODBC en mode conception, tout en utilisant JET, ODBC, JDBC, ADO, DBI et PHPLib dans les programmes générés pour se connecter aux bases de données en phase d'exécution. Par exemple, la connexion d'un projet JSP à une base de données Oracle peut être établie en utilisant le driver JDBC d'Oracle. CodeCharge Studio va jusqu'à implémenter la correction de bogues à la bibliothèque PHPLib pour une utilisation plus fiable du PHP avec Oracle.

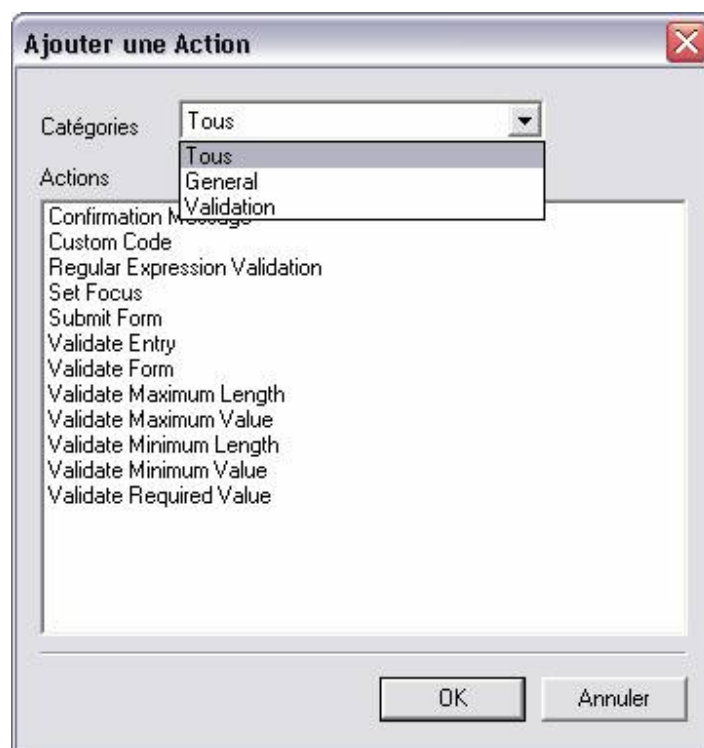
8.1.13. Templates d'Applications (Solutions)

CodeCharge Studio est fourni avec 5 templates/exemples d'applications Web pré-construites, qui peuvent servir de point de départ pour un développement d'applications Web. Ces templates sont prêtes à être personnalisées et adaptées aux besoins. Les templates fournies incluent des applications Intranet, telles que des systèmes de Gestion de Tâches ou Répertoire d'Employés. Les exemples d'application Internet inclut un Portail de Communauté en Ligne, un Forum de Discussion, et un Formulaire d'Enregistrement.

8.1.14. Code sur Mesure et Actions



Les applications générées avec CodeCharge Studio peuvent facilement être étendues en ajoutant du code personnalisé ou en utilisant n'importe laquelle des Actions pré-définies. Les actions sont des composants de code définissables par l'utilisateur qui insèrent des blocs de code dans les procédures d'événements. De nombreuses actions sont fournies avec le produit, mais les utilisateurs peuvent bien évidemment créer des Actions supplémentaires et les partager. Les actions se composent de code XML et XSL facilement modifiable.



Contrairement à l'édition manuelle de code qui ne peut pas toujours être récupérée sans l'utilisation d'un système de version ou une sauvegarde, les actions peuvent être régénérées à n'importe quel moment.

8.1.15. Génération de Code Rapide et Pratique

Un des plus grands défis pour rendre la génération de code pratique est qu'elle doit fournir aux développeurs une totale flexibilité pour modifier le code généré sans perdre les avantages liés à l'automatisation. La technologie mise au point par YesSoftware et CodeCharge Studio répondent parfaitement à ces contraintes et offrent aux utilisateurs une totale flexibilité avec le code généré. Les parties modifiées du code sont verrouillées pour ne pas être réécrites lors de générations ultérieures, ce qui a pour avantage de ne pas avoir besoin d'importer le code modifié dans le source de l'application.

Lors de la publication d'un projet ou d'une page, CodeCharge Studio génère du code source qui est alors copié sur le serveur Web pour pouvoir exécuter l'application. Les utilisateurs peuvent choisir et changer le langage de programmation à tout moment; par exemple, une application peut être générée en Active Server Pages (ASP) sur le poste Windows en local, puis être générée en Java (Servlet ou JSP) pour un déploiement final sur n'importe quel serveur d'application Java. Le code généré par CodeCharge Studio est très semblable au code écrit à la main et peut être très facilement amélioré en dehors de CodeCharge Studio par n'importe quel développeur. Aucun autre logiciel du marché ne génère un code professionnel aussi utilisable et pratique dans autant de langages que ceux supportés par CodeCharge Studio.

8.1.16. Documentation

Une importante documentation est fournie à la fois pour l'IDE et le code de programmation sous-jacent. Le Guide Utilisateur montre comment travailler avec l'IDE de CodeCharge Studio alors que le Composant Reference contient les informations de programmation sur les divers composants utilisés pour construire les applications Web. La documentation inclut également des tutoriaux qui expliquent étape par étape comment réaliser les différentes tâches.

9. CodeCharge Studio Edition Personnel

L'édition Personnel de CodeCharge Studio se différencie de CodeCharge Studio sur les points indiqués ci-dessous. Elle apporte 4 limitations :

- ◆ Création de 100 pages maximum pour un projet
- ◆ Une seule connexion base de données, c'est-à-dire à une seule et même base de données (alors que CodeCharge Studio supporte les connexions multiples)
- ◆ Utilisation d'une seule technologie (choisie au moment de la commande) : ASP, ColdFusion, Java, Perl, PHP ou .NET, ce qui ne permet pas de régénérer le projet dans un autre langage
- ◆ Pas de support des procédures stockées

10. Interface Locale

L'interface de CodeCharge Studio est disponible en 4 langues :

- ◆ Anglais
- ◆ Français
- ◆ Allemand
- ◆ Polonais

Par défaut, après son installation, CodeCharge Studio est lancé avec l'interface en anglais. Pour changer la langue, rien de plus simple ! Il suffit de sélectionner le Menu :

Tools à Language à Français

Pour que le changement soit effectif, il suffit de quitter CodeCharge Studio et de le relancer.

11. Roadmap Produit

Dans le futur CodeCharge Studio devrait pouvoir offrir ses possibilités de génération de code dans d'autres IDE et éditeurs Web, tels que Microsoft Visual Studio.NET, GoLive d'Adobe, NetObjects Fusion, et Dreamweaver MX de Macromedia. CodeCharge Studio est déjà étroitement intégré à Microsoft FrontPage, ce qui permet aux utilisateurs FrontPage de passer du développement de sites Web statiques à la création d'applications Web dynamiques.

Parmi les autres fonctionnalités prévues, il y a :

- ◆ Intégration de systèmes de contrôle de code source et de versions
- ◆ Authentication NT et LDAP
- ◆ Champ sécurisé
- ◆ Génération de scripts de test
- ◆ Un SDK (Software Development Kit) pour les développeurs souhaitant créer leurs propres solutions de génération de code basé sur la technologie CodeCharge
- ◆ De nouveaux Composants et Générateurs
- ◆ CRM, CMS et solutions de travail en équipe construites avec CodeCharge Studio

12. Plus d'Informations

12.1. Liens Utiles

12.1.1. Base de Connaissance

http://support.codecharge.com/kb_search.asp

12.1.2. Documentation

CodeCharge Studio : <http://docs.codecharge.com/studio/>

CodeCharge : http://support.codecharge.com/tutorials.asp?product_id=1

12.1.3. Mise à Jour

<http://support.codecharge.com/updates.asp>

12.1.4. Tutorial

<http://support.codecharge.com/tutorials.asp>

12.2. Support

Email : support@kapitec.com

Internet : http://support.codecharge.com/support_new.asp

et <http://www.kapitec.com/Services/fr/support.htm>

12.3. Autres Questions

Pour toute autre question, n'hésitez pas à nous contacter :

⇒ sales@kapitec.com

⇒ Tél. : +33-5 34 27 90 03

13. Annexe - Aperçu de l'Implémentation de l'Architecture Java

CodeCharge Studio fournit son propre framework qui compose le cœur des applications Web générées en Java. Le framework est basé sur MVC (Model-View-Controller) décrit à cette URL :

http://java.sun.com/blueprints/patterns/2ee_patterns/model_view_controller/

Le modèle utilisé dans les applications générées par CodeCharge consiste en des Contrôles Web implémentés comme des JavaBeans. La couche Contrôleur est construite à partir de ControllerServlet et des Actions de page. La Vue est représentée par le ViewerServlet (pour l'implémentation de Servlet) ou par des pages JSP (pour la version JSP).

Alors que d'autres frameworks similaires sont implémentés comme des environnements *fermés*, celui de CodeCharge est *ouvert* et complètement extensible : les développeurs peuvent aisément modifier les composants inclus (Contrôles Web) et en créer de nouveaux.

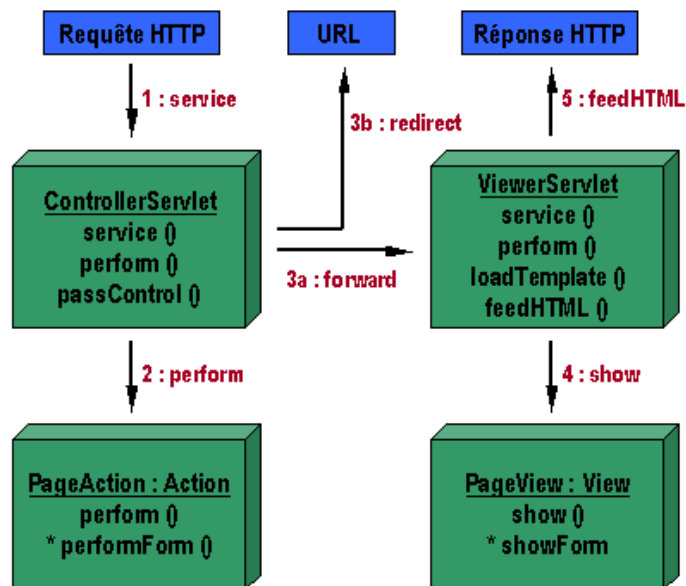
Du modèle de projet simple, CodeCharge Studio génère deux types de code Java, tous les deux supportant la séparation du code de la conception :

- ◆ Servlets Java avec les Templates HTML. Une fois sur le serveur, la Servlet change la Template HTML et remplace les composants HTML statiques par des données provenant de la base.
- ◆ Modèle JSP et XML.
- ◆ Les fichiers XML contiennent les paramètres de configuration pour chaque Contrôle Web inclus dans la page. Les programmes JSP utilisent ces paramètres pour produire le code HTML correspondant.

13.1. Flux de Contrôle de Servlet Java

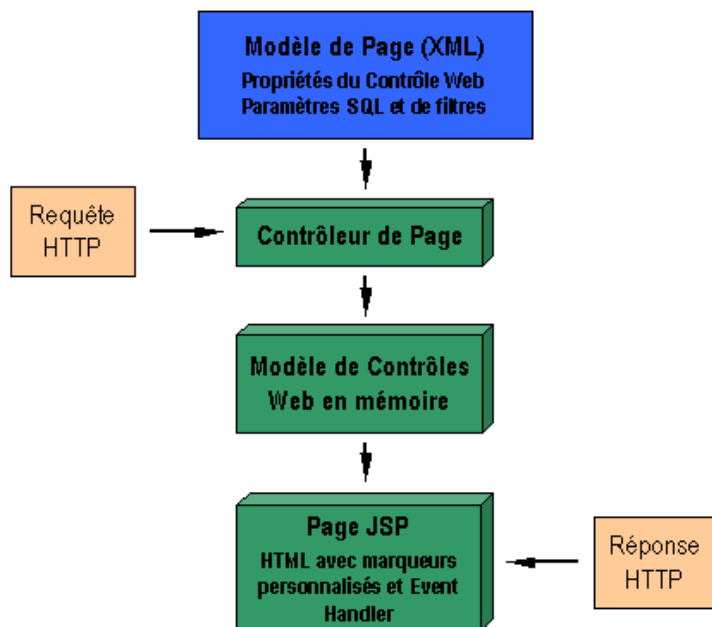
Le diagramme ci-dessous montre comment l'état du Composant (Modèle) passe d'Action à Requête de Visualisation. L'action de PageAction et la vue de PageView ont accès à la Template HTML chargée et aux objets HttpServletRequest et DBConnectionManager (connexion de *pooling*²).

² Mise en commun.



13.2. Modèle JSP et XML

Le diagramme ci-dessous montre comment le modèle de page stocké au format XML est traduit en page HTML (réponse HTTP) avec l'utilisation du Contrôleur de Page qui remplit et met à jour le Modèle objet en Mémoire. La page JSP lit le Modèle En Mémoire et produit les données au format HTML.



13.3. Approches

13.3.1. Model-View-Controller (MVC)

La conception du Framework est basé sur le Model-View-Controller (MVC).

13.3.2. Pooling de Connexion

Permet d'augmenter les performances en sauvegardant les connexions de base de données dans le *pool* (réserve). Au cas où certains drivers de serveurs ne supporteraient pas le *pooling*, le framework de CodeCharge offre ses propres classes de pooling.

13.3.3. Événements

Tous les Contrôles Web de CodeCharge utilisent les événements, ce qui permet aux utilisateurs d'étendre les fonctionnalités de l'application par l'implémentation de listeners d'événements.

13.3.4. Couche Accès aux données

Implémentée comme un JavaBean, sépare la logique d'affaires des données. Permet une connexion facilitée à d'autres Sources de Données.

13.3.5. Templates

Séparation complète de la conception du code de programmation.

13.3.6. Modèle XML

JSP utilise le modèle de représentation XML de Page, qui décrit toutes les propriétés des contrôles et la manière pour retrouver leurs données dans la base de données. En éditant les fichiers XML, les utilisateurs peuvent changer les propriétés, le comportement de la page et ses contrôles.

13.4. Fonctionnalités

13.4.1. Enregistreur (Logger)

Permet l'enregistrement de l'activité de l'application avec différents niveaux de verbosité. L'utilisateur peut facilement détecter les erreurs applicatives et l'endroit où elles se produisent. L'enregistreur peut être désactivé pour augmenter les performances, ou les résultats peuvent être redirigés dans un fichier.

13.4.2. Extension Source de Données JDBC

Permet l'utilisation de connexions pré-définies stockées dans JNDI. Les utilisateurs n'ont pas besoin de se tracasser avec le nom de la base de données, la classe du driver, le login et d'autres paramètres de connexion JDBC. Cela permet une intégration facilitée avec la plupart des serveurs d'applications.

13.4.3. Formatage - Validation de Données

Supporte des formats Date, Numérique, Booléen et bien d'autres formats par l'utilisation des modèles de format. Cela facilite l'implémentation de validations, telles que requis, unique, longueur min./max., valeur min./max., expression régulière, date, etc.

13.4.4. Passage de Paramètres

Permet des interactions entre les pages et les formulaires. Le développeur peut indiquer quels sont les paramètres à conserver (QueryString, POST, ALL) et ceux à enlever (par nom). Ceci est particulièrement utile pour préserver l'état du tri, le navigateur et les critères de recherche entre pages.

13.4.5. Valeurs par Défaut et Expressions

Les paramètres SQL peuvent être obtenus depuis plusieurs sources : Requête, Session, Application, CodeExpression (toute expression Java). Si le paramètre est vide, une valeur par défaut est appliquée.

13.4.6. Construction Automatique de Requêtes SQL Basées sur des Valeurs de Paramètres

Construit automatiquement les requêtes SQL appropriées qui peuvent être différentes si certains paramètres SQL ont des valeurs nulles. Cela permet d'économiser du temps lors de l'implémentation de requêtes SQL complexes.

13.4.7. Actions Pré-Définies

Brides de code semi-généré qui peuvent être utilisées pour stocker et rechercher des valeurs de session, validation de saisie de données, action de connexion et de déconnexion au système. Ceci permet de cacher la complexité du développement aux utilisateurs tout en fournissant une flexibilité totale et la possibilité de développer de nouvelles Actions pour le besoin d'un service.

13.5. Composants

13.5.1. Contrôles Web

Les Contrôles Web représentent des contrôles HTML et d'autres composants de niveaux plus élevés, tels que Grille, Enregistrement, Navigateur et Tri. Ils sont implémentés comme des JavaBeans et permettent de contrôler le comportement sur le serveur lors de l'exécution.

13.5.2. Marqueurs

Simplifie la structure des pages JSP et permet la réutilisabilité.

13.5.3. ControllerServlet

Décide comment envoyer des requêtes à une Action ou à une Vue basée sur une requête URL.

13.5.4. Classe Abstraite Action

Fournit la fonctionnalité de base pour Page Action. Simplifie le développement de nouvelles actions.

13.5.5. Classe Abstraite View

Fournit la fonctionnalité de base pour PageView. Stocke les réponses et les autres objets pour donner la Vue.

13.5.6. Classe JDBCConnection

Classe d'aide pour encapsuler le travail avec l'API JDBC et pour fournir l'interface pour les différentes Sources de Données (Connexion mise en commun - *Pooled* - ou connexion JDBC 2.0).