

ISSN 1677-8464

## CodeCharge: Gerador de Códigos para Aplicações Web

Edgard Henrique dos Santos<sup>1</sup>

Programas geradores de código são extremamente práticos quando se tem pouco tempo ou uma pequena equipe para o trabalho de desenvolvimento. Esta afirmação é verdadeira quando se imagina um aplicativo incompleto quanto ao uso, que gere código completo sem a necessidade de personalizações e dê uma série de opções para a escolha da linguagem de programação ou do banco de dados que se deseja utilizar.

Ampliando as situações de praticidade da afirmação pode-se ir além e pensar em uma ferramenta fictícia que a partir da análise do problema através de uma "conversa com o cliente" pudesse dar, sem a necessidade de retoques, a melhor solução possível. E talvez isso ocorra no futuro. Nada de equipes, apenas uma máquina conversando com o cliente.

Algumas ferramentas Computer Aided Software Engineering (CASE) possuem interfaces que possibilitam a geração de código, porém existem limitações que as deixam distantes da simples "conversa com o cliente". E CASE, como automação do desenvolvimento de software (MacClure, 1989), não tem a função de banir o trabalho de programação de código do processo de construção de um sistema. Além disso, alguns embutem paradigmas de desenvolvimento que podem exigir o treinamento de uma equipe, o que não as tornariam práticas segundo os termos da afirmação dada no início.

Um programa de fácil manipulação e que, especificamente para o desenvolvimento *Web*, apresenta bons resultados é o CodeCharge da Yes Software Corporation (Yes Software, 2003). CodeCharge não é um CASE segundo a definição de uma ferramenta de apoio ao processo de engenharia de software (Sakamoto, 1998). Também não é uma opção mágica e de uso irrestrito para a criação de sistemas, mas se apresenta como um gerador de códigos para aplicações *Web* que fornece opções de uso de diferentes linguagens de programação e acesso a bancos de dados.

A seguir será dada uma introdução sobre o CodeCharge e mostrado um exemplo prático de uso, com o intuito de propor aos desenvolvedores de aplicações *Web* o gerador de código como opção de escolha, além de algumas considerações sobre o produto.

### Características do CodeCharge

O CodeCharge é um a ferramenta Rapid Application Development (RAD) desenvolvida pela empresa norte-americana Yes Software Corporation que integra a função de geração de código para aplicativos de acesso a banco de dados via *Web*, em um ambiente Integrated Development Environment (IDE) bastante simples. É possível criar *Web* sites nas linguagens Active Server Pages (ASP) (Microsoft Corporation, 2003b), Java

<sup>1</sup> B.Sc. em Ciência da Computação, Técnico de Nível Superior II da Embrapa Informática Agropecuária, Caixa Postal 6041, Barão Geraldo - 13083-970 - Campinas, SP. (e-mail: edgard@cnpia.embrapa.br)

Server Pages (JSP) (Sun Microsystems, 2002a), PHP: Hypertext Preprocessor (PHP) (PHP Group, 2003a), Perl (O'Reilly Associates, 2003a), ColdFusion (Macromedia Inc., 2003), ASP.Net/C# (Microsoft Corporation, 2003c, 2003d) acessando qualquer banco de dados que possa ser conectado usando as tecnologias JET (Microsoft Corporation, 2003e), Open Database Connectivity (ODBC) (Microsoft Corporation, 2003f), Java Database Connectivity (JDBC) (Sun Microsystems, 2003b), ActiveX Data Objects (ADO) (Microsoft Corporation, 2003a), Data Base Interface (DBI) (PHP Group, 2003a) e PHPLib (PHP Group, 2003b).

Suporta o uso de código Structured Query Language (SQL) customizado, além do propriamente gerado pela aplicação, através do uso de formulários de tabelas, de registros, *login*, pesquisa e menus. O desenvolvedor pode especificar através desses formulários, a ação – baseada em funções SQL – e o que mostrar quando o banco de dados for acessado, além de determinar os níveis de autenticação de usuário na aplicação.

Além disso possui funcionalidades que auxiliam no trabalho de *design*, como uma interface para a geração de arquivos Cascading Style Sheet (CSS) e estilos Hypertext Markup Language (HTML) ou a aplicação de alguns existentes em um diretório contendo *templates*, a criação de diagramas de site, que possibilitam representar a estrutura da interface da aplicação *Web*.

Como principais características da versão 2.0 têm-se :

- **Banco de Dados:** qualquer banco de dados relacional, incluindo Oracle, Sybase, mySQL, Microsoft SQL Server e Microsoft Access que sejam passíveis de conexão via JET, ODBC, JDBC, ADO, DBI e PHPLib.
- **Linguagens:** ASP, ASP.Net/C#, ColdFusion, JSP, PHP, Perl.
- **Plataformas:** os códigos gerados rodam em qualquer versão do Windows, Unix e Linux.

#### • Requisitos de sistema:

- Processador Intel ou equivalente, 133+ MHz.
- 64 MB de memória RAM e 25MB de espaço no disco rígido.
- Windows 9x/NT/2000/XP, ADO 2.0, Internet Explorer 5.0+.
- Web Server capaz de executar ASP, ASP.Net/C#, ColdFusion, JSP, PHP ou Perl.

#### Exemplo de Uso

O exemplo a seguir mostra um aplicação *Web* simples, chamada Telefones, que permite a pesquisa e edição de uma tabela contendo informações de nomes e telefones. A aplicação foi criada a partir da versão 2.0 do CodeCharge disponível para *download* no site <http://www.codecharge.com>, que permite o uso por trinta dias para avaliação. O banco de dados utilizado é o MSAccess da Microsoft e a linguagem escolhida, JSP. O servidor de aplicação *Web* usado é o Jakarta TomCat 4.1.12 (Apache Software Foundation, 2002), executando em um Pentium 1,7 GHz com Windows 2000.

Os passos da criação do exemplo são expostos a seguir para mostrar a praticidade da criação de aplicações *Web* sem padronizações de código.

Ao solicitar a execução do CodeCharge é aberta uma caixa de diálogo que fornece as opções para abrir ou criar um novo site. Neste exemplo escolhe-se a primeira, clicando na opção "Create new site" como mostra a Fig. 1.

Na área "Properties", título "General", coloca-se o nome do projeto de "Telefones" e em seguida escolhe-se JSP1.1 como linguagem de programação para a geração das páginas. Entre as opções estão as linguagens ASP,C#, ColdFusion, JSP, PHP e Perl, que podem ser alteradas a qualquer momento do projeto (Fig. 2).

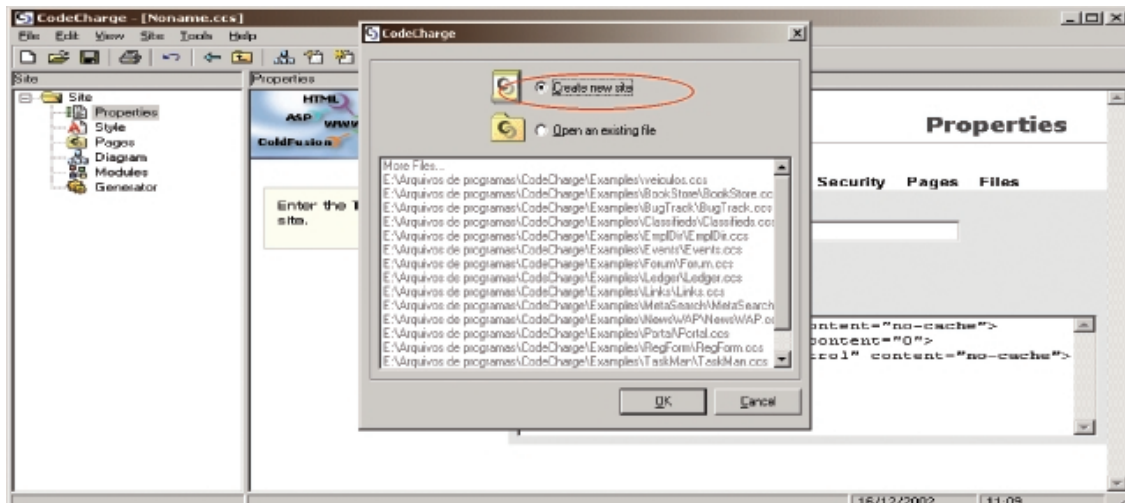
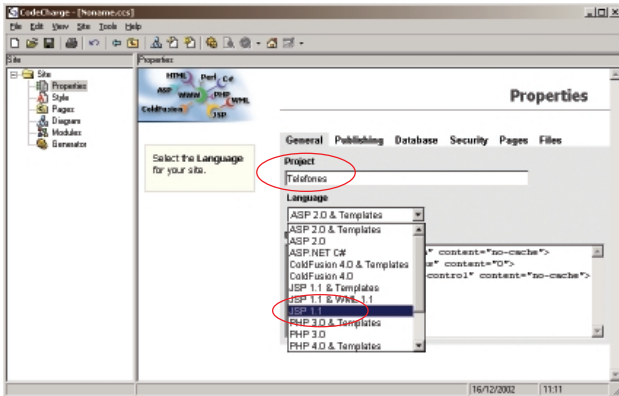
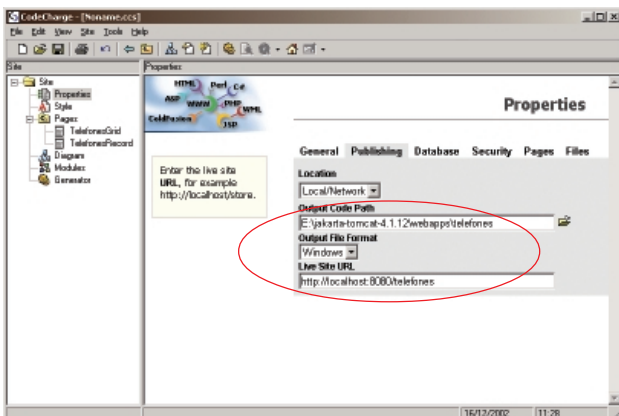


Fig. 1. Create new site - Criação de novo site.



**Fig. 2.** Properties/General - Propriedades gerais

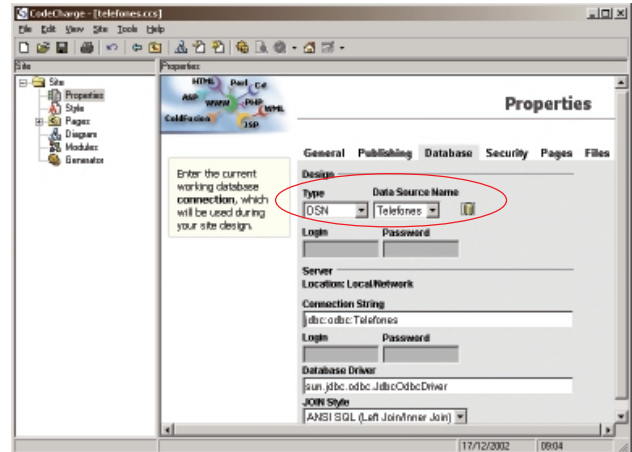
No título "Publishing" (Fig. 3) escolhe-se para "Location" a opção "Local/Network" para publicar as páginas via rede. Pode-se escolher a opção "FTP", caso se deseje publicar o site via File Transfer Protocol (FTP). O "Output Code Path" é o caminho físico para as páginas que serão geradas e executadas no servidor. Usou-se a pasta "webapps" contida no diretório do "Jakarta-TomCat" e dentro foi criada uma com o nome do projeto.



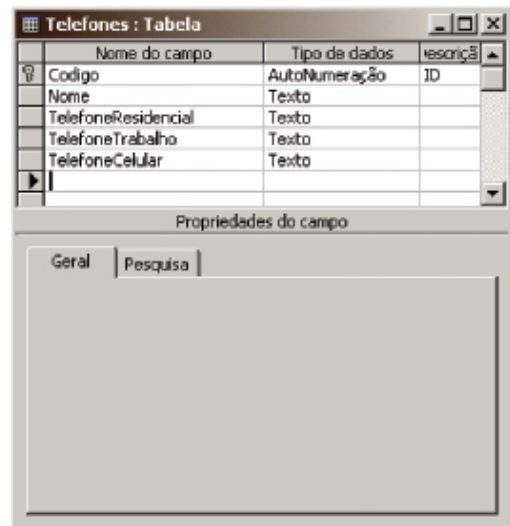
**Fig 3.** Properties/Publishing – Local de publicação das páginas.

O "Live Site URL" contém o caminho da aplicação a ser acessado pelo browser. No caso de uma publicação em uma máquina que não a local deve-se substituir o nome "localhost" pelo endereço Internet Protocol (IP) do servidor a ser usado. Usa-se o endereço local `http://localhost:8080/telefonos`, onde telefonos é a referência à pasta criada em "webapps".

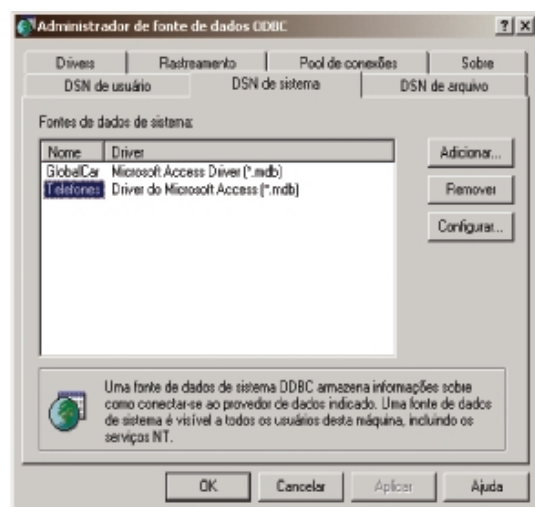
No título "Database" informa-se o banco de dados e a forma de conexão a ser usada (Fig. 4). No exemplo criou-se um banco de dados chamado "Telefonos" no SGBD Access, da Microsoft (Fig. 5), e associou-se à uma Data Source Name (DSN) a partir do administrador de ODBC do sistema operacional Windows (Fig. 6).



**Fig. 4.** Properties/Database – Configurações de acesso a bancos de dados.



**Fig. 5.** MSAccess Design.



**Fig. 6.** Administrador de fonte dados ODBC.

No campo de entrada "Type" escolhe-se DSN e procura-se o nome do banco de dados que foi associado ao DSN.

Usou-se a string de conexão “jdbc:odbc:Telefones” no campo “Connection String” e no campo “Database Driver” usou-se o driver “sun.jdbc.odbc.JdbcOdbcDriver”. Tanto a string de conexão quanto o driver variam de acordo com o SGBD que é escolhido para a aplicação e devem ser observadas, caso se deseje alterar o banco de dados da aplicação.

Na área “Style”(Fig. 7), botão “Load”, escolheu-se a folha de estilo (Fig. 8) correspondente às cores de borda de tabelas, texto e fundo que se quer para a aplicação. É possível criar uma folha de estilo personalizada ou escolher o item a ser editado no campo “Item” e modificar cores e fontes.

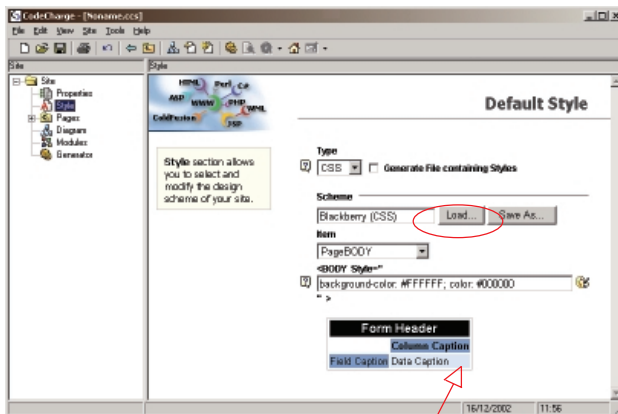


Fig. 7. Default Style - Estilo padrão.

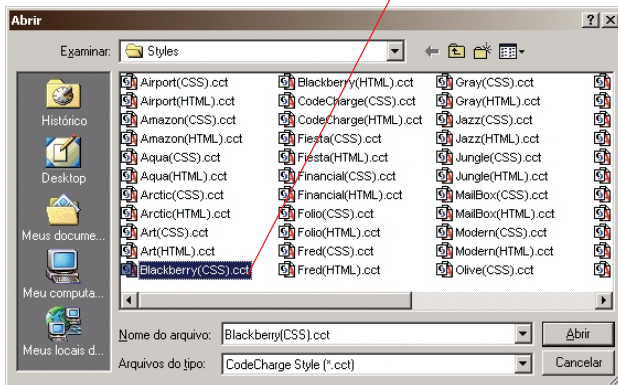


Fig. 8. Style – Folhas de Estilo.

Cria-se então as páginas para acesso ao banco de dados. Um página de consulta de telefones por nomes e uma outra que possibilita a edição e inclusão de novos nomes e telefones. Na área “Pages” pode-se escolher entre as opções “Add Page” e “New Page Wizard” (Fig. 9). Escolhe-se a segunda por conter alguns modelos práticos de navegação.

Selecionado o modelo “Grid and Edit” (Fig. 10), que contém uma seqüência para consulta e edição, escolheu-se o banco de dados e os campos que são mostrados na consulta (Fig. 11). Note que os campos da tabela que foi escolhida aparecem na caixa de diálogo sem a necessidade de digitá-los.

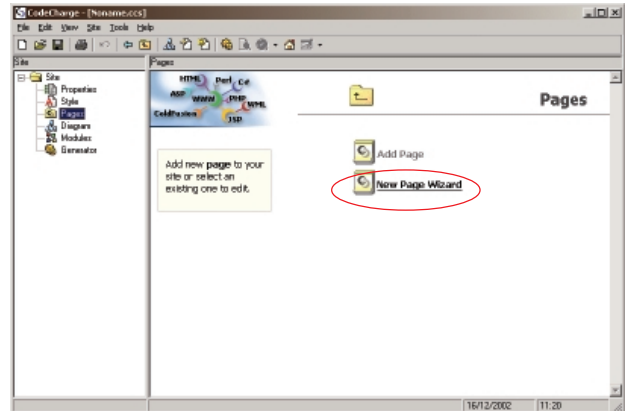


Fig. 9. Pages/New Page Wizard - Criação de nova página.

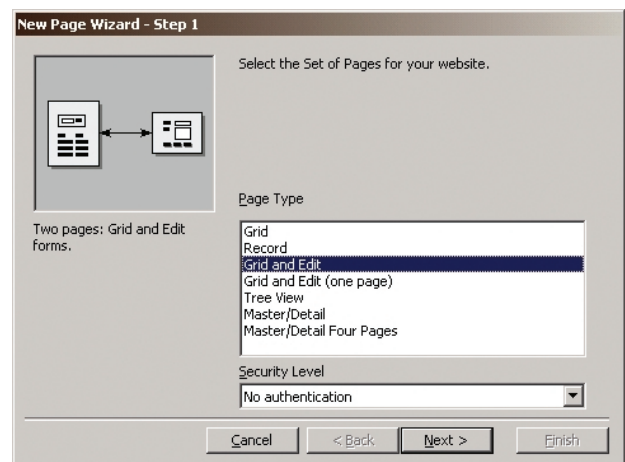


Fig. 10. New Page Wizard - Step 1.

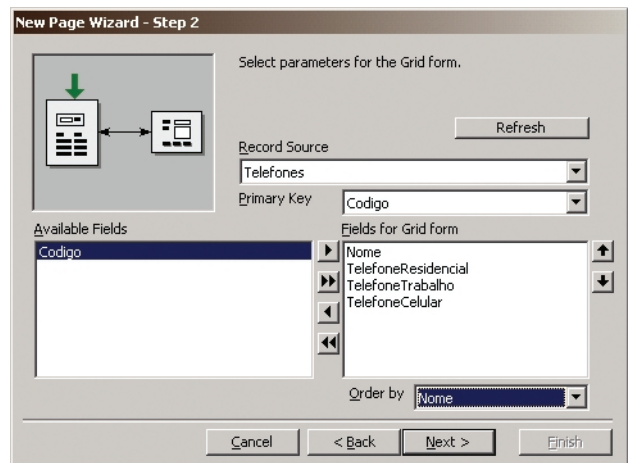


Fig. 11. New Page Wizard - Step 2.

Selecionou-se também os campos que são mostrados (Fig. 12) no formulário de edição. No caso do exemplo, todos os campos são incluídos para a página de edição.

O segundo formulário é o de consulta. Escolhe-se os campos e o tipo de busca (Fig. 13). No caso da escolha de um único campo, no exemplo “Nome”, o tipo de busca padrão é “And”, podendo ser alterado para “Or” em caso de adição de mais um campo.



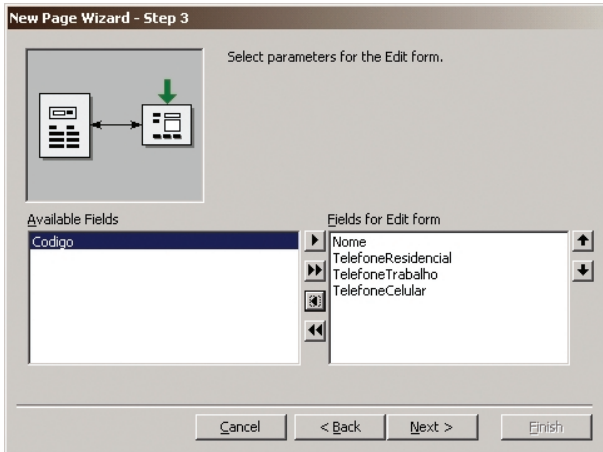


Fig. 12. New Page Wizard - Step 3.

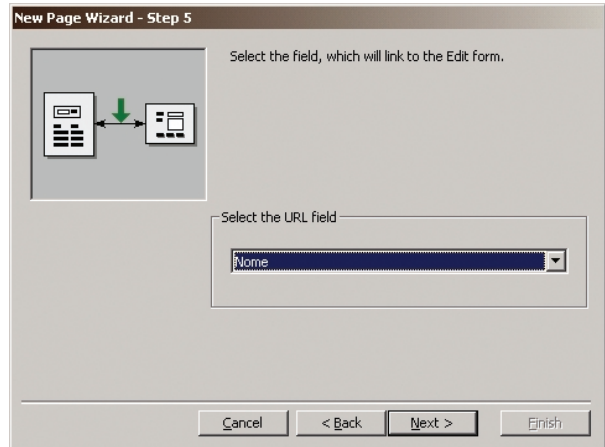


Fig. 14. New Page Wizard - Step .5.

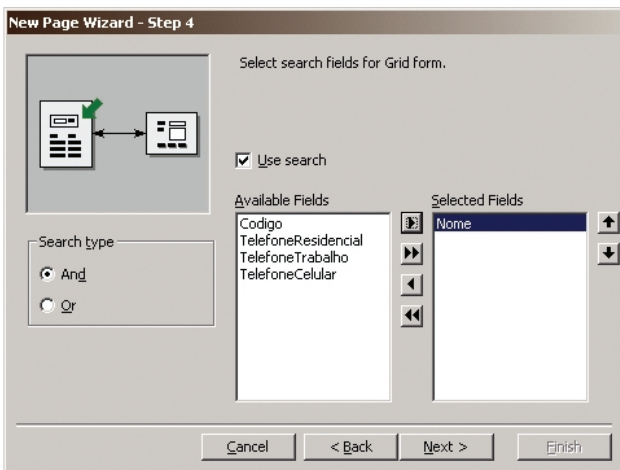


Fig. 13. New Page Wizard - Step 4.



Fig. 15. New Page Wizard - Step 6.

Logo após as escolhas feitas para os formulários (Fig.15) e solicitação de finalização, uma caixa de diálogo é mostrada (Fig. 16). As páginas são geradas logo após ser pressionado o botão "Finish".

Escolhe-se o campo que servirá de link para o formulário de edição (Fig. 14) e que pode posteriormente ser alterado para um botão. O número de registros que será mostrado por páginas pode ser modificado e no caso foram escolhidos 10. Podem ser adicionadas funcionalidades de ordenação por campos ou barra de rolagem (Fig. 15). Na mesma caixa de diálogo se determina os critérios de edição de campos. O formulário do exemplo permitirá todas as funções (inserção,deleção, cancelamento e regras de validação)



Fig. 16. New Page Wizard – Finished.

A página de busca pode ser acessada pelo endereço <http://localhost:8080/telefones/TelefonosGrid.jsp>. Os nomes das página JSP são criados automaticamente pelo CodeCharge e podem ser alterados

posteriormente. A página “TelefonosGrid.jsp” (Fig.17) apresenta a lista de nomes como links para uma página de edição de dados e um link para a página de inserção de dados (Fig. 18).

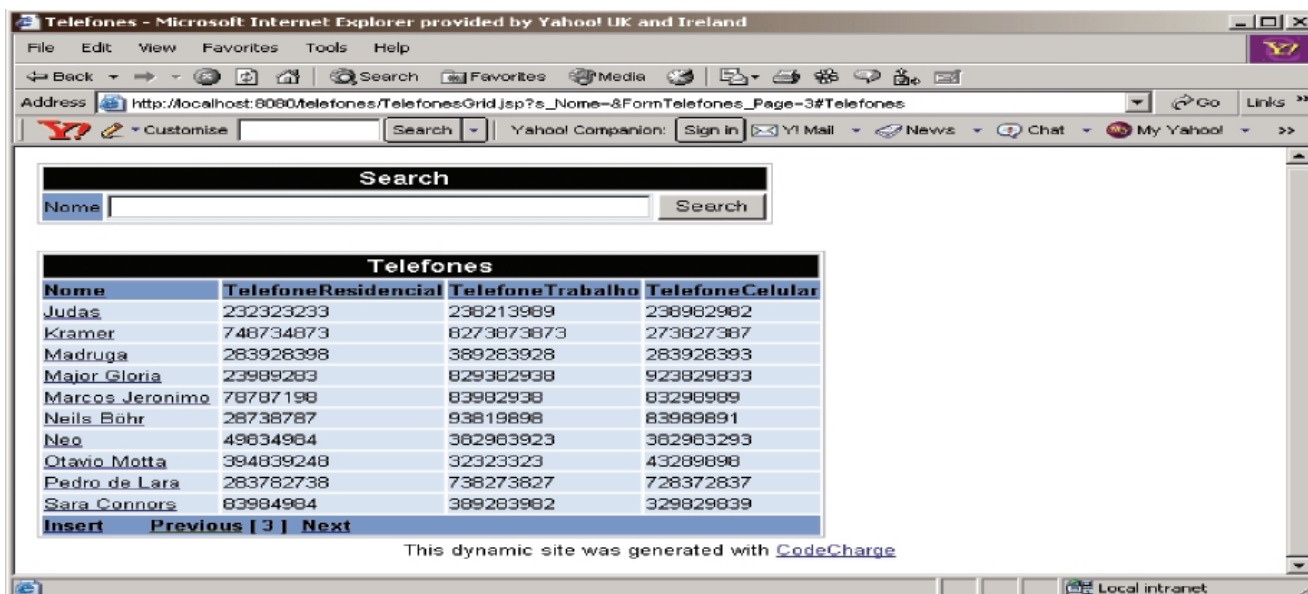


Fig. 17. Página de consulta TelefonosGrid.jsp.

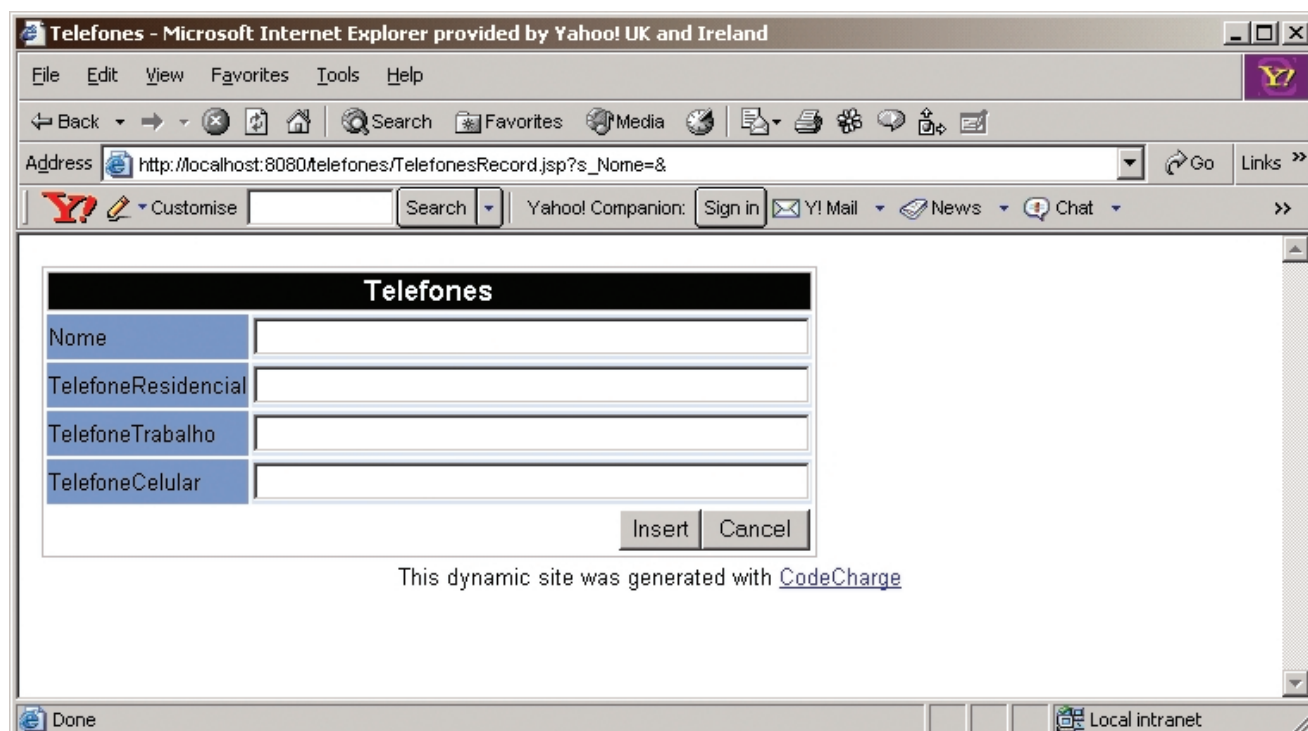


Fig. 18. Página de Inserção de dados TelefonosRecord.jsp.

## Considerações Finais

A redução do tempo de desenvolvimento proporcionada pelo CodeCharge, sua interface intuitiva e a variedade de tecnologias envolvidas que podem ser usadas sem a necessidade do conhecimento aprofundado de todas elas são as principais vantagens da ferramenta.

No entanto, padronizações que vão além de suas funcionalidades, como a inclusão de algoritmos mais complexos ou específicos, podem exigir um trabalho maior, pela necessidade de se analisar um programa gerado automaticamente, um problema que pode ser resolvido através do uso do CodeCharge Studio, uma evolução do CodeCharge, que apresenta uma interface mais avançada e completa e que facilita o processo de padronização. Assim como o CodeCharge, o Charge Studio foi testado por trinta dias, porém apresentou problemas na geração de código JSP que estão sendo resolvidos pela YesSoftware.

Mesmo com alguma necessidade de padronização o CodeCharge mostrou-se eficiente acessando o banco de dados do sistema Agritempo, desenvolvido em Oracle, gerando páginas JSP e futuramente poderá substituir a tecnologia de servlets usada atualmente.

Deve-se conhecer os limites do CodeCharge e da aplicação que se deseja desenvolver para que se possa ponderar os benefícios de seu uso. É um trabalho semelhante a escolhas de plataformas, métodos e arquitetura. Optar entre Cobol e Java para um determinado projeto, por exemplo, requer uma análise detalhada logo no início. Da mesma forma, optar por um gerador de códigos como o CodeCharge pode ser vantajoso, e isso vai depender de uma boa análise preliminar.

## Referências Bibliográficas

ACUCORP INC. **Acucorp, Inc. - the new face of Cobol.** Disponível em: <<http://www.cobol.com>>. Acesso em: 21 fev. 2003.

APACHE SOFTWARE FOUNDATION. **The Apache Jakarta Project:** The Jakarta site: Apache Tomcat. Disponível em: <<http://jakarta.apache.org/tomcat/>>. Acesso em: 24 set. 2002.

GIBSON, M. L. A guide to selecting CASE tools. **Datamation**, v. 34, n. 13, p. 65-66, jul. 1988.

MACROMEDIA INC. **Macromedia ColdFusion MX.** Disponível em: <<http://www.macromedia.com/software/coldfusion>>. Acesso em: Consultado em 21 de fevereiro de 2003.

MCCLURE, C. **CASE is software automation.** Upper Saddle River: Prentice-Hall, 1989. 290 p.

MICROSOFT CORPORATION. **Accessing data with ADO.NET.** Disponível em: <<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpguide/html/cpconaccessingdatawithadonet.asp>>. Acesso em: 21 fev. 2003a.

MICROSOFT CORPORATION. **Active Server Pages.** Disponível em: <<http://msdn.microsoft.com/library/default.asp?url=/nhp/Default.asp?contentid=28000522>>. Acesso em: 21 fev. 2003b.

MICROSOFT CORPORATION. **ASP.NET.** Disponível em: <<http://msdn.microsoft.com/library/default.asp?url=/nhp/Default.asp?contentid=28000440>>. Acesso em: 21 fev. 2003.

MICROSOFT CORPORATION. **C# Programming languages future features.** Disponível em: <[http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dv\\_vstechart/html/vbconCProgrammingLanguageFutureFeatures.asp](http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dv_vstechart/html/vbconCProgrammingLanguageFutureFeatures.asp)>. Acesso em 21 fev. 2003.

MICROSOFT CORPORATION. **Microsoft Jet database engine programmer's guide - introduction.** Disponível em: <<http://www.microsoft.com/technet/treeview/default.asp?url=/technet/prodtechnol/sql/proddocs/msjet/jetintro.asp>>. Acesso em: 21 fev. 2003c.

MICROSOFT CORPORATION. **ODBC.** Disponível em: <[http://msdn.microsoft.com/library/default.asp?url=/library/en-us/vccore98/HTML/\\_core\\_odbc.asp](http://msdn.microsoft.com/library/default.asp?url=/library/en-us/vccore98/HTML/_core_odbc.asp)>. Acesso em: 21 fev. 2003d.

ORACLE CORPORATION. **Oracle 8i - Internet - Oracle Technology Network.** Disponível em: <<http://otn.oracle.com/products/oracle8i/content.html>>. Acesso em: 24 set. 2002.

O'REILLY & ASSOCIATES. **Perl.com:** a short guide to DBI. Disponível em: <<http://www.perl.com/pub/a/1999/10/DBI.html>>. Acesso em: 21 set. 2003a.

O'REILLY & ASSOCIATES. **Perl.com - the source for Perl.** Disponível em: <<http://www.perl.com/>>. Acesso em: 21 set. 2003b.

PHP GROUP. **PHP - hypertext preprocessor.** Disponível em: <<http://www.php.net>>. Acesso em: 21 fev. 2003a.

PHP GROUP. **PHP base library.** Disponível em: <<http://phplib.sourceforge.net>>. Acesso em: 21 fev. 2003b.

SAKAMOTO, A. R. Ferramenta CASE – solução? **Artigos InfoChoose**, set. 1998. Disponível em: <<http://www.choose.com.br/infochoose/artigos/viewer.asp?n=2&a=1>>. Acesso em: 24 set. 2002.

SUN MICROSYSTEMS. **Java servlet technology**. Disponível em: <<http://java.sun.com/products/servlet/>>. Acesso em: 22 set. 2002a.

SUN MICROSYSTEMS. **The source for Java technology**. Disponível em: <<http://java.sun.com>>. Acesso em: 22 set. 2002b.

SYBASE INC. **Sybase**: e-business software integration with databases, mobile / wireless, financial services, solutions, enterprise portals and application servers. Disponível em: <<http://www.sybase.com/home>>. Acesso em: 21 fev. 2003.

YES SOFTWARE. **CodeCharge**: code generator for rapid web application development. Disponível em: <[http://www.codecharge.com/download/files/pdf/CodeCharge\\_DataSheet.pdf](http://www.codecharge.com/download/files/pdf/CodeCharge_DataSheet.pdf)>. Acesso em: 23 set. 2003.

#### Comunicado Técnico, 45

#### Embrapa Informática Agropecuária Área de Comunicação e Negócios (ACN)

Av. André Tosello, 209  
Cidade Universitária - "Zeferino Vaz"  
Barão Geraldo - Caixa Postal 6041  
13083-970 - Campinas, SP  
Telefone (19) 3789-5743 - Fax (19) 3289-9594  
e-mail: sac@cnptia.embrapa.br

**1ª edição**  
2002 - on-line  
Todos os direitos reservados

#### Comitê de Publicações

**Presidente:** José Ruy Porto de Carvalho  
**Membros efetivos:** Amarindo Fausto Soares, Ivanilde Dispatto, Luciana Alvim Santos Romani, Marcia Izabel Fugisawa Souza, Suzilei Almeida Carneiro  
**Suplentes:** Adriana Delfino dos Santos, Fábio Cesar da Silva, João Francisco Gonçalves Antunes, Maria Angélica de Andrade Leite, Moacir Pedroso Júnior

#### Expediente

**Supervisor editorial:** Ivanilde Dispatto  
**Normalização bibliográfica:** Marcia Izabel Fugisawa Souza  
**Capa:** Intermídia Publicações Científicas  
**Editoração Eletrônica:** Intermídia Publicações Científicas