

## **FERRAMENTA DE GESTÃO APLICADA AO GERENCIAMENTO DE EMPREENDIMENTOS NA ÁREA DA CONSTRUÇÃO CIVIL**

Rodrigo Henrich

Faculdades Integradas de Taquara – Faccat – Taquara – RS – Brasil  
rodrigohenrich@gmail.com

Prof. Dr. Marcelo Azambuja

Professor Orientador

Faculdades Integradas de Taquara – Faccat – Taquara – RS – Brasil  
azambuja@faccat.br

### **Resumo**

Este artigo apresenta os resultados de uma pesquisa experimental com o objetivo de desenvolver uma ferramenta de gerenciamento para empreendimentos na área da construção civil denominada SGCC (Sistema de Gerenciamento para Construção Civil). O desenvolvimento do sistema foi baseado em uma metodologia elaborada de forma a permitir o acompanhamento de todas as etapas de um empreendimento, permitindo uma maior agilidade dos participantes de um projeto em tomar decisões. O sistema foi desenvolvido utilizando tecnologias web, que permite uma maior facilidade de acesso. O desenvolvimento foi baseado em tecnologias e ferramentas gratuitas, desta forma o sistema será distribuído livremente para ser baixado e utilizado por qualquer pessoa ou empresa interessada.

**Palavras-chave:** empreendimento, metodologia de gerenciamento, web, construção civil.

## **MANAGEMENT TOOL APPLIED TO DEVELOPMENTS IN THE FIELD OF CONSTRUCTION**

### ***Abstract***

*This article presents the results of an experimental research which aims at developing a management tool for enterprises in the civil construction area called SGCC (Management System for Civil Construction). The development of the system was based on a methodology created to enable the monitoring of all stages of a project, allowing greater flexibility to participants to make a decision in a project. The system was developed using web technologies, which allow a greater easy access. The development was based on free software and technologies so that the system will be freely distributed to be used and downloaded by any person or company interested in it.*

**Key-words:** *enterprise, management methodology, web, civil construction.*

## 1. Introdução

Um projeto é formado por uma sequência lógica de processos, com um objetivo bem definido. As atividades devem ser coordenadas com o objetivo de gerar um produto ou serviço único. Kerzner (2006) nos diz que projeto é uma sequência de processos com objetivos, prazos, custos bem definidos, onde cada tarefa possui recursos específicos alocados e cada etapa do cronograma deve ser cuidadosamente monitorada por um gestor, para evitar falhas na execução.

Gerenciamento de projeto é o ato de planejar e controlar a execução das atividades ou processos que fazem parte do projeto, coordenando as atividades e garantindo a qualidade e o cumprimento dos prazos, a fim de gerar resultados satisfatórios para o produto ou serviço final.

Para que todos os processos do gerenciamento de projetos possam ocorrer de maneira sincronizada e com o menor grau de erro possível, a utilização de ferramentas informatizadas podem auxiliar nesse objetivo. Com essa finalidade, surgem os sistemas para gerenciamento de projetos. Existem hoje no mercado vários *softwares* de gerenciamento de projetos que atendem a projetos de uma forma genérica, oferecendo suporte ao gerenciamento de cronograma e recursos de pessoal para cada tarefa (CIRIACO, 2009). Como exemplo deste tipo de ferramenta, pode-se citar os Microsoft<sup>1</sup> Project, OpenProj<sup>2</sup>, dotProject<sup>3</sup> e openWorkbench<sup>4</sup>.

Este trabalho apresenta como proposta o desenvolvimento do sistema denominado SGCC - Sistema de Gestão para Construção Civil, implementado para plataforma web, utilizando linguagens voltadas para este fim específico, e onde foi priorizada a utilização de tecnologias livres e consolidadas no mercado de desenvolvimento para web. Seguindo o princípio de software livre a ferramenta desenvolvida neste projeto foi disponibilizada para ser baixada gratuitamente, permitindo que ela seja instalada, utilizada e modificada livremente.

O principal objetivo do sistema é possibilitar a utilização de recursos computacionais de gerenciamento de projetos na construção civil, permitindo que os usuários tenham acesso as informações do projeto de forma simplificada e eficiente, possam manter-se informados sobre o andamento geral de um empreendimento e sejam alertados sobre algum imprevisto o mais rapidamente possível, contornando um possível contratempo, como um atraso no andamento do cronograma, por exemplo. Para atender esse objetivo foi desenvolvida uma metodologia que especifica uma série de processos que permitem o acompanhamento da execução de um projeto.

---

<sup>1</sup> [www.microsoft.com.br](http://www.microsoft.com.br)

<sup>2</sup> [sourceforge.net/projects/openproj/](http://sourceforge.net/projects/openproj/)

<sup>3</sup> [www.dotproject.net/](http://www.dotproject.net/)

<sup>4</sup> [sourceforge.net/projects/openworkbench/](http://sourceforge.net/projects/openworkbench/)

A principal linguagem aplicada no desenvolvimento da aplicação foi a PHP<sup>5</sup>, empregando-se o paradigma orientado a objetos. Também foram utilizadas programações com JavaScript<sup>6</sup>, a partir do uso da biblioteca jQuery<sup>7</sup>. A interface com o usuário foi desenvolvida utilizando as linguagens de formatação HTML<sup>8</sup> e CSS<sup>9</sup>. Para a codificação foi utilizado o ambiente de desenvolvimento integrado (IDE) NetBeans<sup>10</sup>, em sua versão específica para codificação web.

O artigo está dividido nas seguintes seções: I) a seção 2 apresenta a metodologia proposta neste trabalho para gerenciar projetos na área de construção civil; II) A seção 3 apresenta as principais tecnologias utilizadas na execução do projeto; III) na seção 4 é apresentada a metodologia de desenvolvimento de *software* adotada no desenvolvimento deste trabalho; IV) na seção 5 são apresentados os resultados obtidos; V) a seção 6 apresenta as considerações finais relativas a pesquisa; VI) na seção 7 é apresentada a bibliografia pesquisada.

## **2. Proposta de metodologia para o acompanhamento das tarefas**

A primeira contribuição deste trabalho foi a criação de uma metodologia que permite acompanhar os processos de um típico projeto na área de construção civil. A Figura 1 ilustra a metodologia de trabalho proposta.

Foi especificada uma sequência de processos que devem ser seguidos para estruturar a empresa construtora dentro do sistema, bem como a forma como as tarefas serão acompanhadas.

---

<sup>5</sup> Hypertext Preprocessor

<sup>6</sup> Linguagem de scripts interpretada, utilizada para adicionar dinamismo a páginas web no lado cliente.

<sup>7</sup> <http://jquery.com/>

<sup>8</sup> HyperText Markup Language, que em português significa Linguagem de Marcação de Hipertexto.

<sup>9</sup> Cascading Style Sheets que em português significa folhas de estilo em cascata.

<sup>10</sup> Ambiente de desenvolvimento integrado, gratuito e de código de fonte aberto.

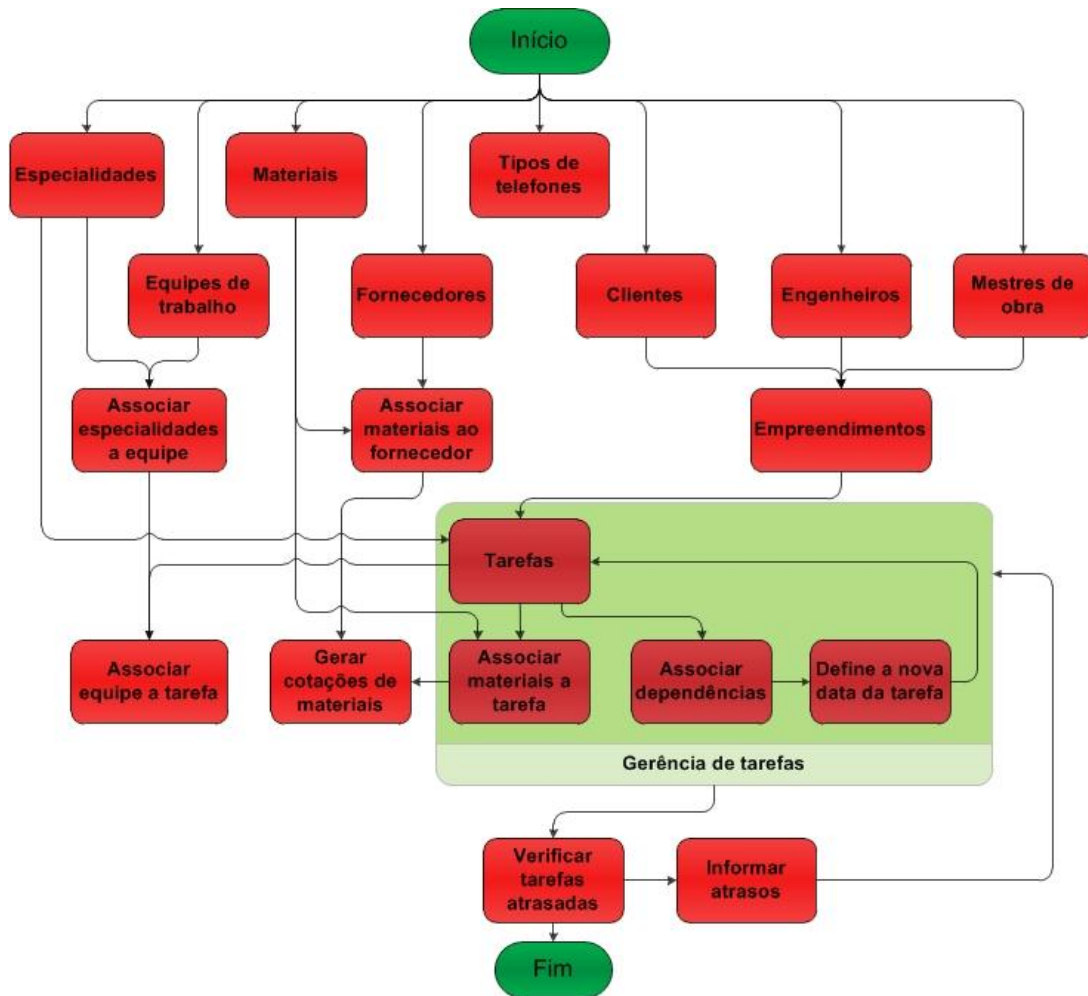


Figura 1: Diagrama com as etapas e sequenciamento da metodologia proposta para o sistema.

A partir da Figura 1, pode-se compreender que, ao acessar o sistema pela primeira vez, o usuário deverá cadastrar as informações básicas que serão utilizadas em vários processos. São elas:

- Especialidades: pode-se chamar também de tipo de tarefa, e é todo o tipo de atividade que será realizada durante uma construção, como por exemplo pode-se citar pintura, reboco, piso entre outros. As informações requeridas para esta informação são nome, uma unidade de medida e uma descrição;

- Materiais: são todo o tipo de materiais que são utilizados em uma obra, tais como areia, cimento, brita, cal, tinta, entre outros. Cada material é composto de um nome e uma unidade de medida;

- Tipos de telefones: inclui classes de telefones no sistema, e as informações necessárias ao funcionamento desse módulo é um nome e um campo para dizer se esse tipo de telefone pode ou não receber mensagens de texto. Caso sim, os telefones cadastrados com estes tipos serão utilizados no módulo de comunicação com os usuários.

A partir dos cadastros fundamentais, é possível inserir as informações que utilizam junções ou complementações desses cadastros fundamentais, tais como:

- Equipes de trabalho: para realizar as tarefas é necessário criar equipes, estas equipes devem ser cadastradas no sistema. As equipes podem ser internas, ou seja, equipes da empresa construtora, ou terceirizadas, neste caso requerem informações sobre a empresa responsável por esta equipe;

- Associar especialidade a equipe: nesse ponto são definidas quais são as especialidades de uma equipe cadastrada. Para que este processo possa ocorrer é preciso que existam especialidades e equipes cadastradas;

- Fornecedores: são empresas responsáveis pela venda dos materiais necessários na execução das tarefas.

- Associar materiais ao fornecedor: cada fornecedor pode trabalhar com um tipo diferente de material, assim tendo materiais cadastrados é possível associa-los ao um determinado fornecedor por meio deste processo;

- Clientes: são pessoas que contratarão a construtora para realizar seus empreendimentos;

- Engenheiros: são pessoas que serão responsabilizadas por um empreendimento;

- Mestre de obras: pessoa que está presente no canteiro de obras coordenando o andamento das atividades.

- Empreendimento: depois de definidos os clientes, engenheiros e mestres de obra é possível criar um empreendimento que possui, além de um cliente, um mestre de obras e um engenheiro responsável, um endereço, onde será executado o projeto;

- Tarefas: tendo um empreendimento, especialidades e equipes cadastradas pode-se incluir uma nova tarefa que será associada a um empreendimento; uma tarefa possui uma data e hora inicial, uma data e hora final, que gera um tempo de duração para esta tarefa;

- Associar equipe a tarefa: no próximo passo é preciso definir uma equipe que executará a tarefa, de acordo com a especialidade da tarefa, deve ser selecionada uma das equipes que atenda aquela especialidade;

- Associar materiais a tarefa, nesta etapa são listados todos os materiais e quantidades necessários no decorrer da tarefa;

- Gerar cotações de materiais: gera cotações que serão enviadas aos fornecedores, permitindo uma escolha pelo menor custo de cada material. Um material adicionado a tarefa, será cotado com todos os fornecedores que possuem este material associado a seu perfil;

- Associar dependências: as tarefas podem ter dependência entre si, por exemplo, não posso construir paredes sem ter realizado as fundações, pensando neste aspecto a metodologia permite adicionar essas dependências a tarefa, dizendo qual tarefa precisa estar pronta para que outra inicie;

- Define a nova data da tarefa: com a adição de requisitos, a tarefa não pode ser executada antes que a dependência esteja concluída, para garantir isso, caso seja necessário, a data da tarefa deverá ser alterada para que ela inicie após o término de sua dependência;

- Verificar tarefas atrasadas, com todas essas informações definidas as tarefas serão executadas e o percentual concluído delas deverá ser atualizado periodicamente por algum membro da equipe, o engenheiro ou mestre de obras poderão realizar esta operação. As tarefas que ultrapassarem a data de finalização e não estiverem prontas serão listadas e informadas aos participantes do empreendimento.

### 3. Referencial teórico

Nesta seção serão apresentadas as principais tecnologias que foram utilizadas para o desenvolvimento deste projeto de *software*. Ferramentas aplicadas ao gerenciamento da estrutura do projeto, modelagem do banco de dados, desenvolvimento da análise do sistema e desenvolvimento do sistema.

#### 3.1 Linguagem de programação PHP

PHP (HiperText PreProcessor) é uma linguagem de programação muito difundida em programação web. Permite a criação de páginas interativas, por meio de formulários, permitindo uma fácil interação com banco de dados (TOLEDO, 2011).

O PHP é uma linguagem interpretada, que é executada do lado servidor, o que permite que as páginas sejam geradas de modo muito rápido, e garante uma grande segurança para os dados que estão sendo submetidos por formulários web. Essa segurança é importante, pois garante a integridade e confiabilidade das informações que estão sendo tratadas pelo código PHP (TOLEDO, 2011). Oferece suporte aos bancos de dados mais comuns no mercado, tais como MySQL, PostgreSQL, InterBase, Oracle, entre outros. Além de suporte a diversos SGBDs<sup>11</sup>, o PHP pode ser instalado em vários sistemas operacionais, como Windows, Linux, OpenBSD, Solaris, entre outros, permitindo ao desenvolvedor escolher o servidor que melhor se ajuste as suas necessidades (WELLING, 2006).

PHP suporta o paradigma de programação orientado a objetos e estruturado, o que confirma a versatilidade desta linguagem. O PHP também permite desenvolver scripts que podem ser rodados via linha de comando (WALACE, 2006).

Para atender as requisições web ao PHP foi utilizado um dos mais populares servidores HTTP<sup>12</sup> gratuitos e de código fonte aberto do mercado, o Apache. (SOUZA, 2005).

---

<sup>11</sup> Sistema de Gerenciamento de Banco de Dados.

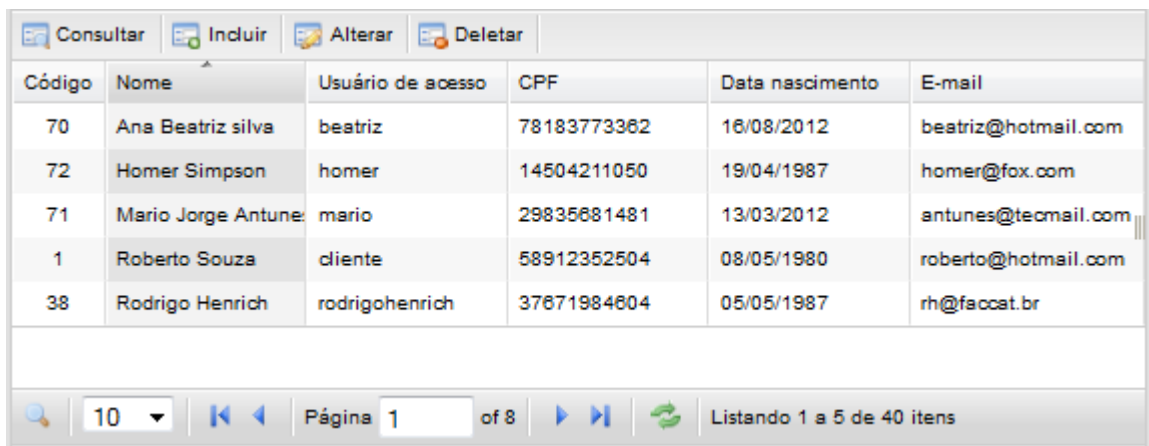
<sup>12</sup> Protocolo de Transferência de Hipertexto

### 3.2 Biblioteca JavaScript jQuery

Segundo Silva (2010a) JQuery é uma biblioteca JavaScript desenvolvida e disponibilizada gratuitamente por Jonh Resig. Sobre licença GPL<sup>13</sup>. O JQuery tem por objetivo facilitar e agilizar criação de *scripts* em linguagem JavaScript permitindo a adição de interatividade a páginas web com a manipulação de efeitos visuais, alteração do DOM<sup>14</sup> a qualquer momento, utilização de requisições Ajax<sup>15</sup> e uma série de recursos avançados.

JavaScript é uma linguagem de programação interpretada, leve, com recursos de orientação a objetos que permite a adição de recursos avançados a páginas web, é interpretado do lado cliente pelo navegador, permitindo que algumas informações sejam tratadas antes de serem submetidas ao servidor (SILVA, 2010b).

As funcionalidades do jQuery podem ser facilmente ampliadas com a utilização de *plugins* disponíveis para *download* gratuitamente. Um destes que está sendo utilizado neste projeto o FlexGrid<sup>16</sup>, que é um *plugin* jQuery para exibição de *grids*<sup>17</sup> permitindo uma série de recursos com ordenação, pesquisa, paginação, adição de botões, entre outras. Este *plugin* aparece em várias telas do sistema uma delas a listagem de clientes, que pode ser observada na Figura 2.



Código	Nome	Usuário de acesso	CPF	Data nascimento	E-mail
70	Ana Beatriz silva	beatriz	78183773362	16/08/2012	beatriz@hotmail.com
72	Homer Simpson	homer	14504211050	19/04/1987	homer@fox.com
71	Mario Jorge Antunes	mario	29835681481	13/03/2012	antunes@tecmail.com
1	Roberto Souza	cliente	58912352504	08/05/1980	roberto@hotmail.com
38	Rodrigo Henrich	rodrigohenrich	37671984604	05/05/1987	rh@faccat.br

10 | < << | Página 1 of 8 | >> > | Listando 1 a 5 de 40 itens

Figura 2: Listagem de clientes exemplo de funcionamento FlexGrid

<sup>13</sup> GPL – General Public Licence, pode ser consultado em <http://www.gnu.org/copyleft/gpl.html>

<sup>14</sup> DOM – Document Object Model.

<sup>15</sup> Ajax - Asynchronous Javascript and XML – Permite a realização de requisições para o servidor sem que seja necessário recarregar constantemente a página. (Pérez, 2008).

<sup>16</sup> <http://flexigrid.info/>

<sup>17</sup> São utilizados para apresentar as informações em formato tabular.

### 3.3 Ferramenta jQueryUI

Segundo Silva (2012) o jQueryUI<sup>18</sup> é uma biblioteca para a biblioteca jQuery, pois elas trabalham juntas, sendo o jQueryUI destinado ao enriquecimento da *interface* com o usuário, seguindo mesmo princípio do jQuery de com pouco código escrito obter resultados muito bons e agradáveis ao usuário.

Assim como o jQuery o princípio de funcionamento do jQueryUI é o de manipulação do DOM, permite a criação de caixas de diálogo, campos para inserção de datas, movimentação de objetos da página entre outros recursos. Na Figura 3, que apresenta a tela de inclusão de cliente do sistema, serve como exemplo de alguns recursos da aplicação da biblioteca jQueryUI. Na imagem pode-se ver um *datepicker*, que é um campo para seleção de uma data específica, ele esta disponível dentro do pacote de *widgets*<sup>19</sup> do jQueryUI, e sua utilização é amplamente customizável, sendo possível alterar o nome dos meses, dos dias da semana, definir qual o ano inicial, qual o formato oficial da data, enfim uma série de recursos de fácil aplicação em sistemas web.



Figura 3: Exemplo de funcionamento jQueryUI

<sup>18</sup> <http://jqueryui.com/>

<sup>19</sup> Widget é um componente de uma interface gráfica do usuário, o que inclui janelas, botões, menus, ícones, barras de rolagem entre outros componentes.



## 4. Metodologia de desenvolvimento

O SGCC foi desenvolvido com ferramentas e tecnologias baseadas em *software* livre, para reduzir os custos do projeto e garante que o sistema possa ser utilizado e distribuído livremente. Segundo Alecrim (2011), *software* livre está atrelado a quatro liberdades básicas definidas pela FSF<sup>20</sup>. A liberdade de nível 0 especifica que o usuário poderá executar o *software* para qualquer propósito; a liberdade de nível 1 garante ao usuário o direito de estudar e adaptar o código fonte<sup>21</sup> do programa; a liberdade de nível 2, permite distribuir cópias de forma a ajudar na distribuição do *software*. A liberdade de nível 3 permite ao usuário modificar e redistribuir o *software* desde que sejam seguidas as regras anteriores. Essas liberdades estão definidas nos termos de licença GPL<sup>22</sup> definidos pelo FSF.

O sistema foi desenvolvido utilizando uma metodologia rápida de desenvolvimento RAD<sup>23</sup>, este tipo de metodologia de desenvolvimento é baseada em pequenos ciclos de desenvolvimento que cumprem algumas fases: (PRESSMAN, 2002).

- a) Modelagem do negócio;
- b) Modelagem dos dados;
- c) Modelagem do processo;
- d) Geração da aplicação;
- e) Teste e entrega.

Na etapa de análise do SGCC foi utilizada a ferramenta StarUML<sup>24</sup>, que permite a modelagem UML<sup>25</sup>. Para o sistema foram criados os diagramas de Casos de Uso, Implementação e de Classes.

### 4.1 Paradigma de programação orientado a objetos

O SGCC foi desenvolvido utilizando a linguagem PHP com o paradigma de programação orientado a objetos. Dall’oglio (2007) nos diz que o paradigma orientado segue uma filosofia que se aproxima mais da vida real, utilizando objetos, daí o nome, orientada a objetos (DALL’OGLIO, 2007).

---

<sup>20</sup> FSF - Free Software Foundation, <http://www.fsf.org/>

<sup>21</sup> Código fonte – é um conjunto de instruções escritas em uma linguagem específica que formam um programa para executar determinada tarefa.

<sup>22</sup> GNU Public License, <http://www.gnu.org/licenses/gpl.html>

<sup>23</sup> Desenvolvimento Rápido de Aplicação.

<sup>24</sup> [staruml.sourceforge.net](http://staruml.sourceforge.net)

<sup>25</sup> Unified Modeling Language

Para poder trabalhar com orientação a objetos é importante entender conceitos como classes e objetos. Classe é como uma estrutura que define um novo tipo de dados, uma classe pode conter atributos e métodos. Um objeto, nada mais é do que uma instância, ou seja, a criação de um objeto de determinada classe, neste caso os atributos passam a definir o estado deste objeto e os métodos servem para modificar o estado deste objeto (RAMOS, et al., 2007).

Por exemplo, pode-se definir uma classe Produto, esta possui os atributos, tipo (define um tipo para o produto), descrição (descreve o produto) e preço (custo do produto). O próximo passo é criar um objeto do tipo Produto, é possível definir com esta mesma classe produtos como impressora, bicicleta, micro-ondas, geladeira, pois todos estes itens possuem atributos definidos na classe Produto, variando apenas os valores para cada produto diferente. Desta forma está caracterizado um objeto (DALL'OGGIO, 2007).

#### **4.2 Etapa de Análise de Desenvolvimento do Software**

O processo de análise se iniciou pelo levantamento de requisitos, que segundo Pressman (2002) é uma maneira de entender o que o cliente deseja do *software*, verificar o que é necessário para cumprir os objetivos, analisar o que pode ou não ser desenvolvido, obter soluções para conflitos e realizar a especificação de forma não ambígua. No SGCC a análise de requisitos partiu da análise do funcionamento do mercado da construção civil atual, que gerou a documentação de projeto de *software*, com base nesta documentação foram gerados os requisitos do sistema que foram analisados e originou a divisão do sistema em módulos de acordo com as funcionalidades, desta forma cada funcionalidade pode ser desenvolvida e testada individualmente como entregas individuais do projeto de *software*. Com base nestas informações os módulos foram listados e ordenados de acordo de acordo com seus pré-requisitos.

O resultado desse processo pode ser visto na Tabela 1, onde existe três colunas. Na primeira identificação, cada módulo listado recebeu um identificador, uma letra, de A até N, representando os quatorze módulos do sistema; a segunda coluna, módulo, o nome de módulo a ser desenvolvido; e a terceira coluna, pré-requisito, mostra como os módulos do sistema estão relacionados, ou seja mostra quais módulos devem estar prontos para que outro possa ser iniciado. Por exemplo, pode-se citar o gerenciamento de clientes, identificado pela letra F. Para que ele possa ser implementado, é preciso ter os módulos de gerenciamento de acesso ao sistema, identificado pela letra A, e o módulo de Gerenciamento de telefones para pessoa, identificado pela letra D.

Tabela 1: Relação dos módulos e seus requisitos do sistema

Identificador	Módulo	Pré-requisito
A	Gerenciamento de acesso ao sistema	-
B	Gerenciamento de especialidades	A
C	Gerenciamento de materiais	A
D	Gerenciamento de telefones para pessoa	A
E	Gerenciamento de telefones para empresa	A
F	Gerenciamento de clientes	A, D
G	Gerenciamento de engenheiros	A, D
H	Gerenciamento de equipes	A, B, E
I	Gerenciamento de fornecedor	A, C, E
J	Gerenciamento de mestre de obra	A
K	Envio de SMS	-
L	Gerenciamento de empreendimentos	A, F, G, H, I, J
M	Gerenciamento de tarefas de um empreendimento	L
N	Geração de cotação de materiais	M

O próximo passo da análise do sistema foi o desenvolvimento dos diagramas de casos de uso. Um exemplo de diagrama em alto nível pode ser visto na Figura 4, onde são mostrados todos os casos de uso disponíveis para o usuário com perfil de construtor.

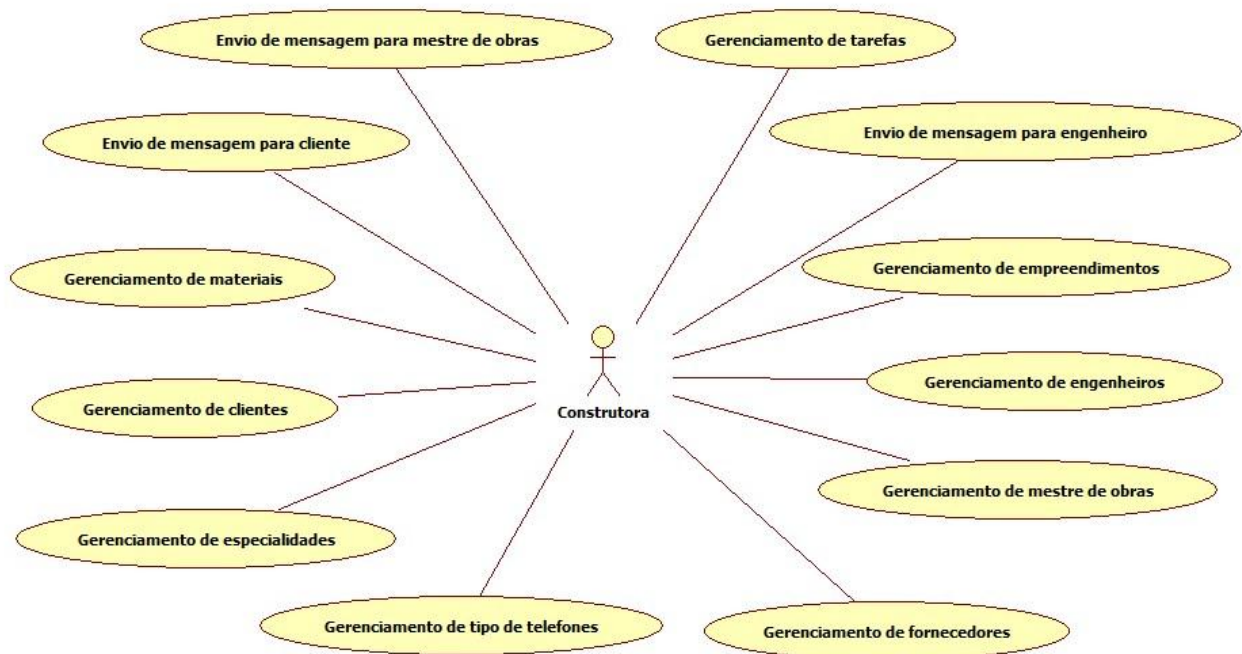


Figura 4: Diagrama de caso de uso do usuário construtor

### 4.3 Estrutura física do sistema

Fisicamente o projeto está alocado em um diretório raiz que foi dividido em três subdiretórios. O primeiro, *docs*, contém a documentação do projeto, como diagramas de banco, diagramas UML e toda a documentação que deu origem ao projeto; a pasta ferramenta guarda as ferramentas necessárias ao desenvolvimento do sistema.

A pasta *sistema* é onde efetivamente estão os códigos fontes do *software*. Nela está contido o arquivo de configuração do sistema, que possui informações como local, usuário e senha de acesso ao bando de dados, diretório de instalação da aplicação, além de conter mais duas subpastas. A primeira subpasta é chamada de *classes*, nela estão gravas todas as classes necessárias ao funcionamento do sistema. O outro diretório contido na pasta sistema chama-se *www* é o primeiro diretório que deve estar visível a nível web, pois é uma medida de segurança para impedir que ocorra violação de acesso ao código das classes e ao arquivo de configuração.

Na pasta *www* estão contidas todos os formulários de cadastro que serão chamados por requisição Ajax, além de conter mais cinco subdiretórios. O primeiro deles é o *jQuery*, que contém a biblioteca javaScript jQuery, dentro ainda existem mais uma pasta chamada *plugins*, que contém os *plugins* jQuery que são utilizados no sistema, como exemplo pode-se citar o flexgrid. Cada *plugin* contém arquivos javaScript e arquivos de folhas de estilo próprios que ficam dentro desta pasta.

Outra pasta que está contida dentro do diretório *www* é a jQueryUI, que possui arquivo de folha de estilo e javaScript próprio. Existe ainda uma pasta *js* com o arquivo *geral.js* onde está todo o código jQuery e javaScript utilizado dentro dos formulários do sistema. Dentro da pasta *tema* localizada no diretório *www* está contido o arquivo *estilo.CSS* que define algumas informações do *layout* do sistema e ainda uma pasta com imagens que são exibidas nas telas do sistema chamada *img*.

A pasta PHP localizada dentro do diretório *www* contém todos os arquivos com o código fonte PHP que são requisitados pelas chamadas Ajax para obter e gravar informações ao banco de dados por intermédio das classes do sistema. Na Figura 5 é possível ver um diagrama que mostra a estrutura de diretórios do sistema.

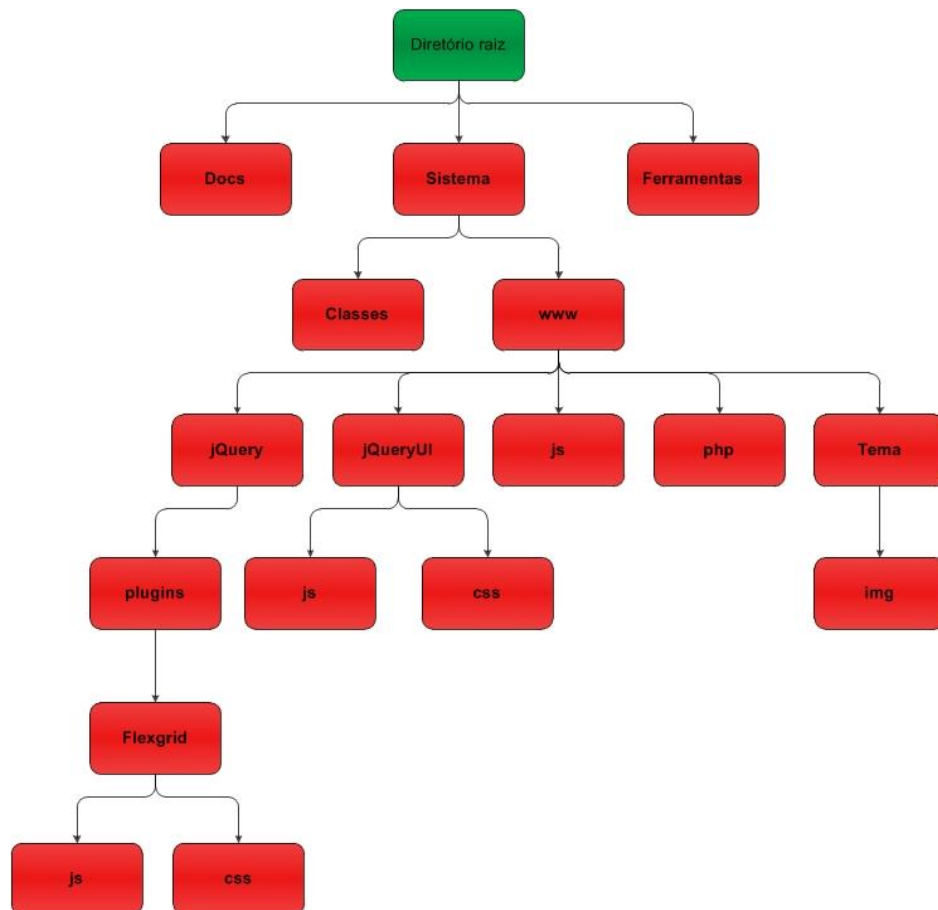


Figura 5: Diagrama ilustrando a estrutura de diretórios do sistema.

#### 4.4 Codificação

Para geração dos códigos fonte do sistema foi utilizado a IDE de desenvolvimento NetBeans. Segundo Netbeans (2012), esse é um ambiente integrado de desenvolvimento de código fonte aberto, que oferece suporte a diversas linguagens de programação como Java, C/C ++, PHP, JavaScript e Groovy.

Em se tratando de ferramentas de *software* web é importante pensar em compatibilidade do código gerado com os diversos navegadores disponíveis no bem como o funcionamento em diversas plataformas de sistemas operacionais. Pensando neste aspecto todas as bibliotecas e *plugins* utilizados foram escolhidos de forma cuidadosa para garantir essa compatibilidade, bem como toda a codificação gerada foi estruturada de forma a utilização do navegador com o qual o usuário esteja mais intimamente acostumado a trabalhar, evitando com isso uma necessidade de adaptações da estrutura de trabalho dos usuários do sistema.

## 4.5 Sistema para controle de modificações Subversion (SVN)

Rapcinsk (2011) diz que o SVN é uma ferramenta de código aberto que permite armazenar e controlar modificações de diversos tipos de arquivos, efetuando controle de acesso a repositórios mantendo um controle sobre as revisões dos arquivos. O SVN se aplica principalmente quando se tem vários desenvolvedores atuando sobre um mesmo projeto, mas pode ser utilizado simplesmente para controle de versão de arquivos pessoais (RAPCINSK, 2011).

Segundo Pressman (2002), controlar a modificação dos arquivos de código fonte do projeto é de fundamental importância, pois se as alterações não forem controladas adequadamente podem levar o projeto ao caos e torna-lo inviável.

Os documentos armazenados em um repositório são baixados para um computador local, por meio de um cliente de SVN, em uma operação chamada de *checkout*, onde é possível escolher a revisão que se deseja baixar. Os arquivos podem ser alterados localmente e as alterações submetidas ao repositório por meio de uma operação chamada de *commit*, cada uma destas operações gera uma revisão. Para receber modificações que estejam no repositório, basta efetuar o comando *update* localmente.

Para controlar as modificações dos arquivos do SGCC, foi criado um repositório SVN no portal xp-dev<sup>26</sup>, e para enviar e baixar as revisões dos arquivos para os servidores do xp-dev, foi utilizado o cliente SVN TortoiseSVN<sup>27</sup>. Desta forma é possível garantir a integridade dos arquivos sempre e obter qualquer revisão do projeto.

## 4.6 Funcionamento do sistema

Todo o sistema é baseado em requisições em formato Ajax, desta forma depois de carregado o arquivo principal do sistema todos os demais são adicionados a uma *div*<sup>28</sup> por meio de mecanismos executados pelo jQuery disparadas para o servidor. Como resultado destas requisições são obtidos os formulários em formato HTML, que serão mostrados dentro de uma *div*. Desta mesma forma são obtidos os dados para estes formulários e as informações que serão mostradas nos *grids*. Essas informações serão obtidas em formato JSON e posteriormente mostrados nos formulários e *grid* utilizando mecanismos da biblioteca JavaScript jQuery. Para enviar dados para o servidor utiliza-se o mesmo mecanismo, porém os dados são enviados em formato POST ou GET e

---

<sup>26</sup> xp-dev.com

<sup>27</sup> <http://tortoisesvn.net/>

<sup>28</sup> Define uma divisão em um arquivo HTML.

a resposta após os dados serem tratados pelo PHP é devolvida em formato JSON<sup>29</sup>, as mensagens de conformação ou erro retornadas pelo PHP são então mostradas pela biblioteca jQuery locais e formação específicos de acordo com o tipo de mensagem. Na Figura 6 é possível observar o funcionamento básico das requisições do sistema.



Figura 6: Exemplo de funcionamento de requisição Ajax no sistema.

#### 4.7 Tecnologias para mensagens do SGCC

Para melhorar a comunicação entre os participantes do sistema, algumas informações e ocorrências são comunicadas aos participantes do sistema via e-mail, bem como é possível que os participantes enviem suas próprias mensagens para os demais participantes por meio da consulta de empreendimento.

Visando melhorar a agilidade na leitura do correio eletrônico, o sistema envia mensagens SMS (Short Message Service) para os telefones móveis dos participantes comunicando que foi enviado um e-mail. Para que essa função esteja disponível foi utilizado um serviço de gateway SMS, fornecido pela empresa fastSMS<sup>30</sup>. Esse serviço permite enviar mensagens para o celular por meio de requisições PHP enviadas aos servidores da referida empresa. O serviço funciona na modalidade pré-paga.

O gateway da fastSMS permite o envio de mensagens de texto para telefones das principais operadoras de telefonia do Brasil, tais como: Amazonia Celular, Claro, Oi, Tim e Vivo. Garantindo disponibilidade de 24x7 em 99% do tempo. No site do fastSMS é possível acompanhar estatísticas de utilização do pacote de SMS, por período, o que facilita ao construtor definir qual o pacote se adequa melhor as suas necessidades.

<sup>29</sup> JavaScript Object Notation. Formato simples de dados compatível com uma série de linguagens como C, Java, JavaScript, Perl, Python e PHP. (JSON, 2012).

<sup>30</sup> <https://www.fastsms.com.br/>

## 4.8 Persistência das informações

O sistema utiliza o SGBD<sup>31</sup> MySQL para o armazenamento das informações. Segundo Milani (2007), o MySQL é um banco de dados bastante robusto que oferece características semelhantes a grande maioria dos SGBDs disponíveis no mercado, com a vantagem de possuir uma versão livre baseada na licença GNU-GPL.

Para permitir que outros sistemas de gerenciamento de banco de dados possam ser utilizados no SGCC, o sistema foi estruturado de forma que apenas uma classe interaja com o SGBD, sendo ela responsável por buscar e gravar as informações geradas pelos diversos objetos do sistema. Cada nova instância de um objeto recebe como parâmetro em seu método construtor uma cópia de uma instância do objeto banco.

Caso seja necessário migrar para outro SGBD, basta reescrever essa classe, alterando apenas as informações necessárias para que a classe interaja com a nova ferramenta de banco de dados.

## 5. Resultados obtidos

A pesquisa experimental gerou como resultado uma ferramenta de *software* web, baseada na metodologia de gerenciamento de *software* apresentada na seção 2, ferramenta essa denominada SGCC.

Durante o processo de instalação do sistema aqui proposta, serão cadastrados os dados de uma empresa construtora, que será a base para o funcionamento de todo o sistema.

Para simplificar a utilização do sistema, as telas de listagem de informações seguem um modelo semelhante ao apresentado na Figura 7. As telas de listagem de informações apresentam botões que permitem incluir, alterar, deletar e consultar detalhes da linha na lista. As opções disponíveis nessas telas são geradas de acordo com o perfil do usuário, sendo uma questão de segurança que alguns usuários não possam excluir determinadas informações. Na segunda barra é possível ver o nome das colunas e ordenar os dados clicando sobre uma delas de forma crescente ou decrescente. Também é possível ocultar e exibir colunas, marcado e desmarcando na lista de colunas.

É possível pesquisar entre os dados apresentados clicando na lupa no final do *grid*, alternar entre as páginas clicando nas setas e atualizar o conteúdo clicando no ícone com setas circulares.

---

<sup>31</sup> Sistema de Gerenciamento de Banco de Dados



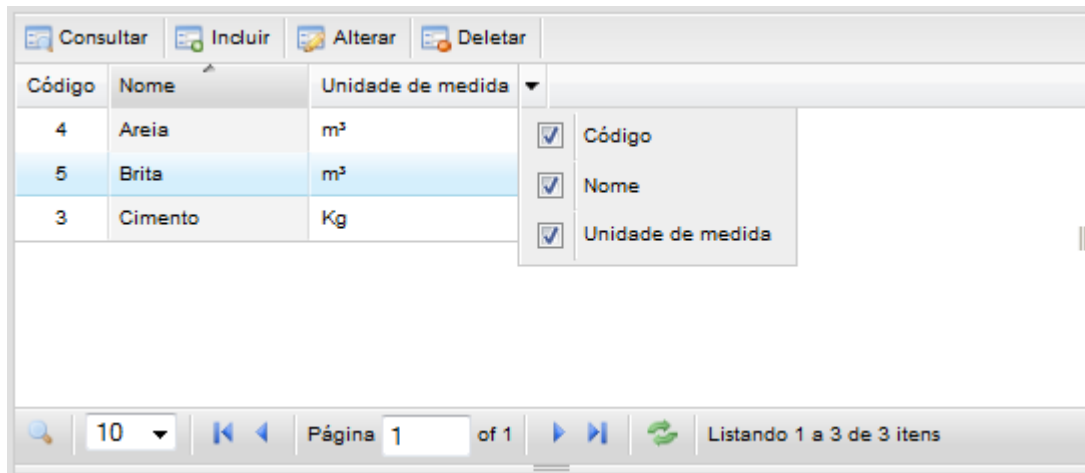


Figura 7: Tela de exemplo da interface do sistema.

Depois de instalada essa empresa construtora, terá um usuário e senha para acessar o sistema e poderá partir para o cadastramento de informações que são requisitos para a criação e gerenciamento dos empreendimentos da empresa, conforme o fluxo explicado na seção 2.

A Figura 8 apresenta a tela inicial do sistema, onde pode-se visualizar um menu na parte esquerda da tela, com todos os módulos disponíveis. No topo a direita é apresentado o nome da pessoa ou empresa que está relacionado com o usuário que fez acesso no sistema.

As telas iniciais para os demais perfis de acesso, cliente, engenheiro e mestre de obra, são iguais, mudando apenas as opções que são apresentadas no menu, de acordo com as permissões do perfil do usuário. O menu é gerado automaticamente de acordo com o perfil de usuário baseado nos relacionamentos entre as páginas e o perfil de usuário.

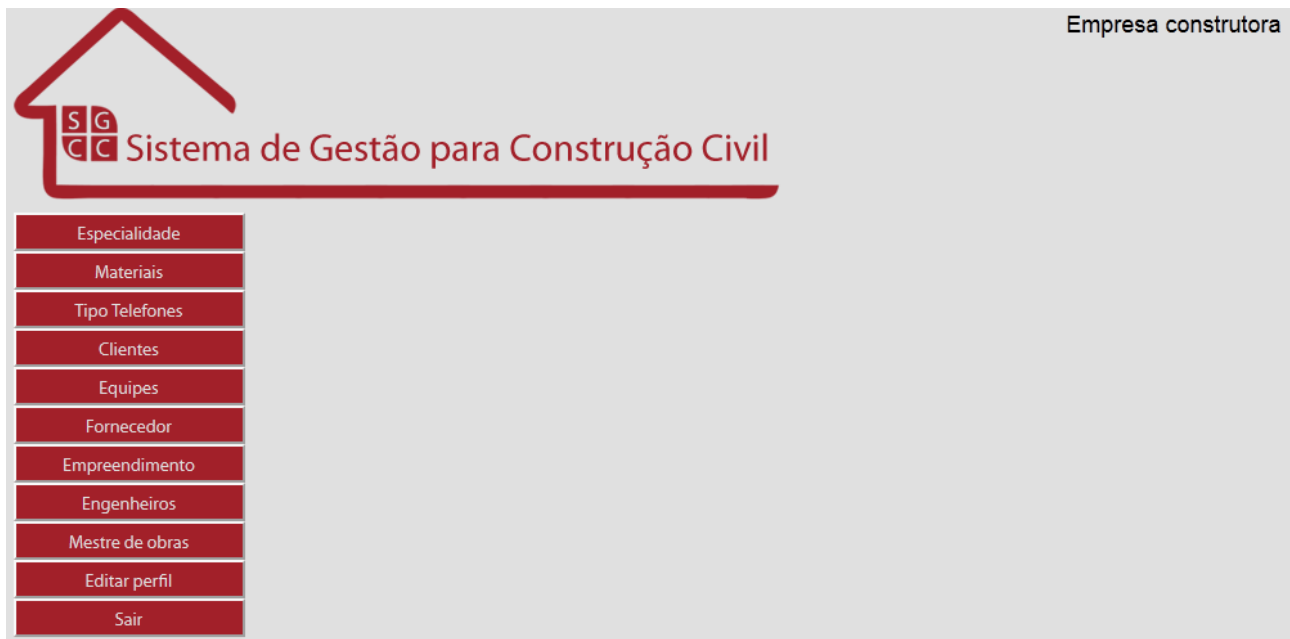


Figura 8: Tela inicial do sistema, ao acessar como construtora.

Com as informações básicas necessárias para o funcionamento do sistema cadastradas, o construtor pode partir para a criação dos empreendimentos. Na Figura 9 pode-se ver a tela de

inclusão de empreendimento. Para melhorar a distribuição das informações a tela foi dividida em abas, onde cada aba contém informações de determinada natureza. Na aba “Dados principais” aparecem informações como nome e endereço do empreendimento, na segunda aba deve ser selecionado um cliente para o empreendimento, na terceira se realiza a escolha de um engenheiro responsável e na quarta um mestre de obra. Ao iniciar o cadastro das informações as demais abas estarão desativadas, só sendo ativadas ao clicar no botão avançar, presente no final de cada aba.



The image shows a web form with a red header bar containing four tabs: "Dados principais", "Cliente", "Engenheiro", and "Mestre de Obras". The "Dados principais" tab is active. Below the tabs are the following fields:

- Nome:
- Rua:
- Número:
- Bairro:
- CEP:
- Estado:
- Município:
- Complemento:
- 

Figura 9: Tela de inclusão de empreendimento

As abas para seleção de cliente, engenheiro e mestre de obras se comportam de maneira muito semelhante. Um exemplo pode ser visto na Figura 10, onde é exibido a aba de seleção de cliente. A seleção é feita clicando em alguma linha da lista. Clicando no botão avançar, é dada sequência no processo. Quando todas essas informações estiverem definidas o empreendimento pode ser salvo e todas as informações serão submetidas via requisição Ajax para serem interpretadas e validadas no servidor, pelo PHP.

Código	Nome	Usuário de acesso	CPF	Data nascimento	E-mail
70	Ana Beatriz silva	beatriz	78183773362	16/08/2012	rodrigohenrich@gmail.com
50	Arnaldo Souza	arnaldo	94549023007	28/08/2012	rodrigohenrich@gmail.com
72	Homer Simpson	homer	14504211050	19/04/1987	rodrigohenrich@gmail.com
71	Mario Jorge Antunes	mario	29835681481	13/03/2012	rodrigohenrich@gmail.com
11	Natalia Alcântara	natalia	74024351800	18/07/2012	rodrigohenrich@gmail.com

10    Página 1 of 5    Listando 1 a 5 de 23 itens

Avançar

Figura 10: Tela de inclusão de cliente ao empreendimento.

Com um empreendimento cadastrado é possível incluir tarefas a este empreendimento. A Figura 11 mostra as duas primeiras etapas da inclusão de uma nova tarefa. Na Etapa 1 são informados os dados básicos como nome e uma descrição do que será realizado. Na Etapa 2 será selecionada a equipe que realizará a tarefa, esta equipe é selecionada com base na especialidade da tarefa, nesta etapa também é informado um tamanho da tarefa por unidade de medida, por exemplo quantos metros quadrados de reboco serão aplicados nessa tarefa.

**Dados principais**   Equipe   Datas   Materiais   Requisitos

Nome:

Descrição:

Avançar

Etapa 1

**Dados principais**   Equipe   Datas   Materiais   Requisitos

Especialidade:

Tamanho da tarefa em m²:

Código	Nome	Terceirizada
44	Equipe 10	Não

10    Página 1 of 1    Listando 1 a 1 de 1 itens

Avançar

Etapa 2

Figura 11: Etapas 1 e 2 do processo de inclusão de tarefa

A Figura 12 apresenta mais duas etapas do cadastro de tarefa, a definição das datas e hora inicial e final da tarefa (Etapa 3) e os materiais necessários para a execução da tarefa (Etapa 4).

The image shows two screenshots of a software interface for task management. The top navigation bar includes tabs for 'Dados principais', 'Equipe', 'Datas', 'Materiais', and 'Requisitos'.  
**Etapa 3:** Shows a form for task scheduling. It includes input fields for 'Data início' (17/10/2012), 'Hora início' (20 : 30), and 'Data final' (dd/mm/aaaa). Below these is a calendar for October 2012, with the 17th highlighted. A 'Salvar' button is present.  
**Etapa 4:** Shows a form for material selection. It includes a 'Materiais' dropdown menu, a 'Quantidade' input field (value: 1), and a 'Cotação' checkbox (checked). Below is a table of materials:

Código	Nome	Quant.	Unidade de medida	Cotação
4	Areia	3	m³	Sim
5	Brita	3	m³	Não
3	Cimento	60	Kg	Sim

At the bottom of Etapa 4, there is a pagination control showing 'Página 1 of 1' and 'Listando 1 a 3 de 3 itens'.

Figura 12: Etapas 3 e 4 do processo de inclusão de tarefa

A Figura 13 apresenta a última etapa do processo de inclusão de tarefa, onde serão definidos os pré-requisitos das tarefas. Ao incluir um pré-requisito a tarefa, suas datas inicial e final poderão ser modificadas para manter a consistência da relação termino inicio das tarefas.

The screenshot shows the 'Requisitos' tab in the software interface. It features a 'Selecione uma tarefa:' dropdown menu with 'Sem itens' selected. Below the dropdown are 'Adicionar' and 'Deletar' buttons. A table lists the task details:

Seq.	Nome	Data/Hora início	Data/Hora fim	Duração
0	Fundações da obra	2012-10-17 10:20:00	2012-10-18 10:20:00	1 d 0 H 0 M

At the bottom, there is a pagination control showing 'Página 1 of 1' and 'Listando 1 a 1 de 1 itens'.

Figura 13: Etapa final do cadastramento de tarefas

Esta forma de interface com o sistema tem como principal objetivo a simplicidade, permitindo aos usuários que não possuem grandes conhecimentos em gerenciamento de projetos que consigam incluir e gerenciar seus empreendimentos.

## 5.1 Compatibilidade com os padrões da W3C

Para garantir a compatibilidade do sistema com os navegadores disponíveis no mercado foi utilizado a ferramenta de validação<sup>32</sup> disponibilizada pelo site da W3C<sup>33</sup> (World Wide Web

<sup>32</sup> <http://validator.w3.org>

Consortium). A W3C é um consórcio onde várias organizações filiadas realizam um trabalho integrado para buscar a padronização em páginas web. A ferramenta de validação utilizada permite verificar a compatibilidade da codificação HTML do código do sistema com os padrões especificados pela W3C.

A Figura 14 mostra a distribuição de utilização dos navegadores, utilizando dados de Bright, (2012). Apesar do Internet Explorer dominar uma fatia grande de mercado, os demais navegadores tem um percentual significativo de usuários. Dessa forma é importante que os sistemas desenvolvidos para web sejam compatíveis com a grande maioria dos navegadores. O SGCC é compatível com o Internet Explorer, Firefox, Google Chrome, Safari e Opera, em suas versões mais atuais que oferecem suporte a especificação 5 do HTML.

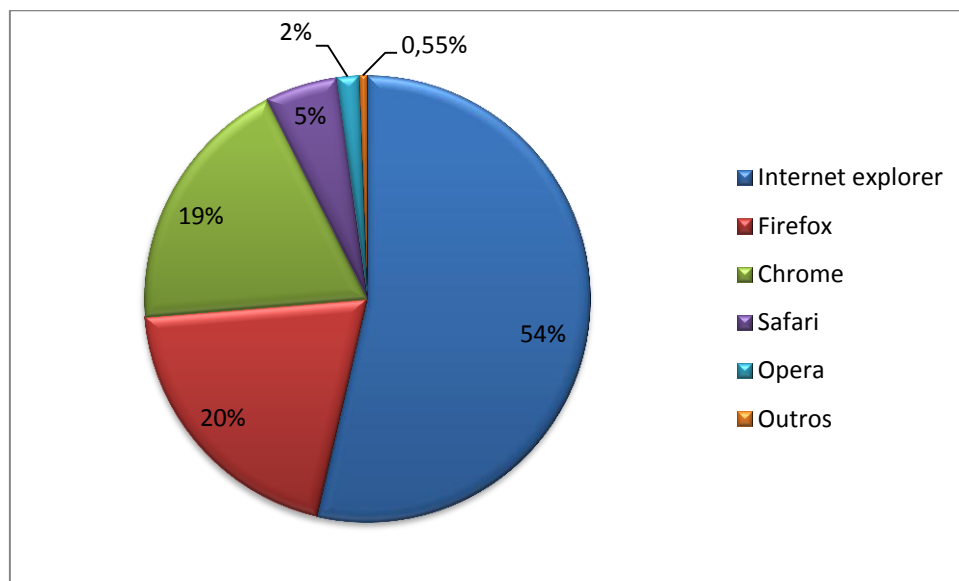


Figura 14: Gráfico ilustrando a utilização dos navegadores.  
Fonte: Adaptado de Bright (2012).

## 5.2 Publicação do *software*

Para que o sistema possa ser distribuído na internet foi criado um repositório para o projeto no sourceforge<sup>34</sup>. O sourceforge é um serviço de hospedagem de projetos de código fonte aberto, além de disponibilizar ferramentas para gerenciamento de código fonte, *bug tracker*<sup>35</sup>, fórum e wiki. Segundo sourceforge, (2012) possui atualmente 324.000 projetos hospedados gerando cerca de quatro milhões de *downloads* diários.

O sistema poder ser baixado dos servidores do sourceforge utilizando o endereço [sourceforge.net/p/sgecc](http://sourceforge.net/p/sgecc). O uso desta ferramenta permite manter um histórico de versões anteriores

<sup>33</sup> <http://www.w3c.br>

<sup>34</sup> <http://sourceforge.net/>

<sup>35</sup> Ferramenta para controle de falhas em *software*

para que possam ser baixadas, e para manter um histórico do desenvolvimento, além de permitir que sejam reportadas falhas ocorridas no sistema, bem como a troca de informações entre os usuários por meio do recurso de fórum.

## 6. Considerações finais

O SGCC é uma ferramenta de *software* web que atende ao gerenciamento de empreendimentos na área da construção civil. Esse sistema permite à construtora ter todas as informações referentes a um empreendimento em um mesmo local, tais como engenheiro, cliente, mestre de obras e endereço. Principalmente, permite um acompanhamento de todas as tarefas que serão executadas dentro de um empreendimento, com as informações sobre qual equipe está realizando os trabalhos, quais matérias primas então sendo empregadas em cada tarefa, bem como um acompanhamento do percentual concluído de cada tarefa. O sistema permite a comunicação entre os participantes do projeto, garantindo que as informações sejam entregues aos interessados em tempo hábil, e mantendo um entrosamento entre os envolvidos em um empreendimento.

O *software* está em uma versão totalmente funcional, atendendo a todos os requisitos especificados pela metodologia criada nesta pesquisa, permitindo o acompanhamento de todas as etapas de um empreendimento. Contudo, poderão ser implementados novos módulos, como por exemplo, adicionar funcionalidades para que os fornecedores possam gerenciar suas cotações pelo sistema, que hoje é realizado por e-mail. Outra funcionalidade que pode ser incluída como trabalho futuro é a inserção de um módulo para geração de gráficos com o sequenciamento das tarefas, o que permitiria uma melhor visualização do projeto, tais como um gráfico de gantt<sup>36</sup>. Também poderão ser incluídos relatórios de custo de empreendimento baseado nas informações das cotações, e também a geração de um memorial descritivo contendo fotos e dados sobre o projeto.

## 7. Referências

- ALECRIM, Emerson. **Software livre, código aberto e software gratuito: as diferenças**. Portal infowester, 2011.
- ALVES, Fábio José. **Desperdício na construção civil**. Artigo consultado em 2011.
- BRIGHT, Peter. **Peak Chrome? Google's browser falls as Firefox, Internet Explorer stay flat**. Arstechnica, 2012.
- CERNEV, Adrian K.; LEITE, Jaci C.. **Segurança na internet**. Acessado em 2011.
- SOUZA, Marcelo S.. **Análise do servidor web Apache em clusters OpenMosix com memória compartilhada distribuída**. Salvador. Centro Baiano de Computação de Alto Desempenho (CEBACAD), 2005.
- CHESWICK, William R.; BELLOVIN, Steven M.; RUBIN, Aviel D.. **Firewalls e segurança na internet 2º edição**. São Paulo. Artmed, 2003.

---

<sup>36</sup> Forma gráfica de representar as tarefas e seus relacionamentos (VARGAS, 2007).

- CIRIACO, Douglas. **Ferramentas para gerenciar projetos**. 2009.
- DALL’OGLIO, Pablo. **PHP Programando com Orientação a Objetos**. São Paulo. Novatec, 2007.
- DATE, C. J.. **An Introduction to database systems**. 8 ed. Boston. Person Education, 2004.
- ELMASRI, Rmez; NAVATHE, Shamkant B.. **Sistemas de banco de dados**. São Paulo. Pearson, 2006.
- FELIPINI, Dailton. **Segurança na internet**. Acessado em 2011.
- FIESP. Federação das Indústrias do Estado de São Paulo.
- JUNG, Carlos Fernando. **Metodologia aplicada a projetos de pesquisa: Sistemas de Informação & Ciência da Computação**. Taquara, 2009. 1 CD-ROM.
- KERZNER, Harold. **Gestão de projetos: as melhores práticas**. 2 ed. São Paulo. Artmed, 2006.
- KUROSE, James F.; ROSS, Keith W.. **Redes de computadores e a internet**. 3 ed. São Paulo. Pearson, 2006.
- LEMAY, Laura. **Aprenda a criar páginas web com HTML e XHTML em 21 dias**. São Paulo: Prentice Education do Brasil, 2002.
- MARTINS, Eliane. **O que é um Script**. 2008.
- MILANI, André. **MySQL – Guia do Programador**. Novatec, 2007.
- NETBEANS. **Portal oficial NetBeans** netbeans.org, Acesso em 2012.
- OHRT, Monte; ZMIEVSKI, Andrei. **Smarty – a ferramenta para compilar templates para PHP**. 2007.
- PEREIRA, Eliane Alves. **Diagnóstico dos problemas de gestão de obras habitacionais de interesse social em empresas privadas**. Belo Horizonte: Universidade Federal de Minas Gerais, 2008.
- PÉREZ, Javier Eguíluz. **Introducción a AJAX**. Livros web, 2008 (Acessado em 2012).
- JSON. www.json.org. (Acessado em 2012).
- PRESSMAN, Roger S. **Engenharia de Software 5ª edição**. Rio de Janeiro: McGraw-Hill, 2002.
- RAMOS, Ricardo, et al.. **PHP Para profissionais**. São Paulo. Digerati, 2007.
- RAPCINSK, Heitor. **Utilizando o subversion como controle de versão**. Grupo de usuários java, consultado em 2011.
- SCHIMITT, Chistopher. **CSS Cookbook**. São Paulo: Novatec, 2010.
- TOLEDO, Carlos Benedito Sica de. **PHP com tudo**. Rio de Janeiro. Ciência Moderna Ltda., 2011.
- SILVA, Maurício Samy. **JavaScript: Guia do programador**. São Paulo. Novatec, 2010b.
- SILVA, Maurício Samy. **jQuery – A Biblioteca do Programador javaScript**. 2 ed. Novatec 2010a.
- SILVA, Maurício Samy. **jQuery UI Componentes de interface rica para suas aplicações web**. Novatec, 2012.
- SOARES, Walace. **PHP 5: conceitos, programação e integração com banco de dados**. 2 ed. São Paulo: Erica, 2006.
- SOURCEFORGE. **Portal oficial do serviço** sourceforge.net. Acesso em 2012
- VARGAS, Ricardo Vianna. **Manual prático do plano de projeto utilizando o PMBOK Guide**. 3 ed. São Paulo: Abreu’s System Ltda, 2007.
- VARGAS, Ricardo Vianna. **Gerenciamento de projetos 6ª edição**. Rio de Janeiro. Brasport, 2008.
- Portal oficial PostgreSql. **www.postgresql.org.br**. 2011.
- WELLING, Luke. **PHP e MySQL: desenvolvimento web**. Rio de Janeiro: Elsevier, 2005.