

# WebSolutions4U – Cloud University Builder

**Bruno C. C. da Costa<sup>1</sup>, Isabella P. C. Moreira<sup>1</sup>, Julio C. P. Marques<sup>1</sup>, Raphaell A. B. Pereira<sup>1</sup>, Thiago V. de Aguiar<sup>1</sup>, Vinicius B. C. Chaves<sup>1</sup>**

Instituto de Matemática e Estatística – Universidade do Estado do Rio de Janeiro  
(UERJ)

Rua São Francisco Xavier, 524 - Maracanã - 20550-013  
Rio de Janeiro - RJ – Brasil

bruno-999@hotmail.com, {isabellapcm, raphaellalexandre}@gmail.com, {julinho\_boss, thiagovaguiar}@yahoo.com.br,  
vbaeta@globo.com

**Abstract.** *Researches in national and international academic web sites, showed a shortage of important components and lack of organization in most of the institute's web sites. Based on these researches, was proposed the creation of a practical solution to generate these web sites. Because it is an application that will be only responsible for the basic creation, and not for the maintenance, was opted for a web solution to be executed on the server. This way it is not necessary to install the application. This article shows the way that was treated all steps of development of this application, along with the technologies that was necessary for the development accomplishment.*

**Resumo.** *Pesquisas em alguns sites acadêmicos, tanto nacionais quanto internacionais, apontaram uma carência de componentes essenciais e falta de organização na maioria dos sites dos institutos. Com base nessas pesquisas, propôs-se a criação de uma solução prática para geração destes web sites. Por se tratar de uma aplicação que somente atuará na criação básica, ou seja, na criação e não, também, na manutenção, optou-se por uma solução web, de forma a ser executado no lado servidor, eliminando a necessidade de instalação do aplicativo. Este artigo mostra a maneira na qual foi tratada todos os passos de desenvolvimento deste aplicativo, juntamente com as tecnologias utilizadas e também os pontos principais que foram pesquisados para que tal desenvolvimento fosse realizado com sucesso.*

**Palavras-Chave:** *Web Site, Site Acadêmico, Desenvolvimento, Tecnologia.*

## **1. Introdução**

As instituições de ensino têm como função básica transmitir informações específicas de forma clara e objetiva. Uma maneira de se passar tais informações é por meio da internet, hoje em dia um meio tão popular quanto à televisão e o rádio. Com essa informação, um web site acadêmico seria a melhor maneira de prover tal interação entre aluno e instituto. Mas, diferente desses dois últimos, a internet necessita de uma formação específica, até mesmo para sites de pequeno porte, para que as informações estejam um mínimo, organizadas e para que de certa forma, ajude ainda mais o aluno.

Uma pesquisa para levantamento de requisitos, feita em sites acadêmicos, de instituições de nível superior, tanto nacionais quanto internacionais, apontou uma carência de alguns componentes básicos tais como: Links para notícias rápidas, calendário de eventos e uma falta de organizações no que se diz respeito ao layout da apresentação de seu conteúdo.

Para auxiliar na melhoria desses web sites, foi proposto o desenvolvimento de um aplicativo que gere estes sites de acordo com os módulos que o usuário irá escolher. Graduação, Mestrado e Doutorado são os módulos disponíveis para escolha. Através de um formulário com abas no qual ele irá informar, por exemplo, nome do site, título, tipo do layout, fará com que a geração seja rápida e objetiva. Dessa maneira, não há qualquer necessidade de que o usuário tenha o domínio dos conceitos de programação web.

## **2. Cloud University Builder (CUB)**

Cloud University Builder (CUB) é a ferramenta web que foi desenvolvida, como uma aplicação em nuvem, provendo facilidades para que usuários com qualquer nível de instrução em programação web possam criar sites acadêmicos de qualidade e de maneira simples.

Para se obter o máximo no que se refere à facilidade de uso, o aplicativo baseia-se em formulários onde o usuário preenche campos e seleciona opções pré-definidas como no caso do layout das páginas, cursos oferecidos, título e outros. No preenchimento total deste formulário, com base nas informações passadas pelo usuário, é gerado um pacote de arquivos correspondente que é disponibilizado ao usuário assim que a criação do site é confirmada. Todas as opções são pré-definidas e não há necessidade do usuário programar.

## **2.1 Diagramas**

Junto à criação do programa, foi desenvolvida uma série de diagramas e documentos para auxiliar na construção e arquitetura do software e oferecer um maior entendimento das funcionalidades do aplicativo.

### **2.1.1 Diagrama de classes**

Ao longo do projeto, vários diagramas de classes foram criados. Eles foram construídos com base em Engenharia Reversa, que tem como principal propriedade gerar diagramas a partir do código já pronto. Assim, conforme o código do programa foi sendo desenvolvido, o diagrama de classes também foi se transformando e evoluindo de forma a refletir tudo que foi sendo alterado e adicionado ao longo do desenvolvimento do projeto. A Engenharia Reversa ajudava de forma vital o entendimento de tudo que foi sendo adicionado ao programa.

## **3. Implementação da Abordagem**

De forma a atender melhor esta demanda, optou-se pela criação de uma ferramenta web desenvolvida com base nas bibliotecas do GWT (Google Web toolkit) acopladas na linguagem de programação JAVA, fazendo uso, também, de HTML e CSS. Também foi utilizado o SmartGWT para melhorar a interface gráfica.

### **3.1 GWT (Google Web Toolkit)**

GWT é um pacote do Google baseado em java que gerência códigos voltados para web e oferece uma série de classes e funções prontas para serem usadas para este propósito.

Sua principal função é simular as classes já existentes na biblioteca java, por intermédio de herança e, na compilação, traduzir os objetos gerados dessas classes para código JavaScript.

### **3.2 SmartGWT**

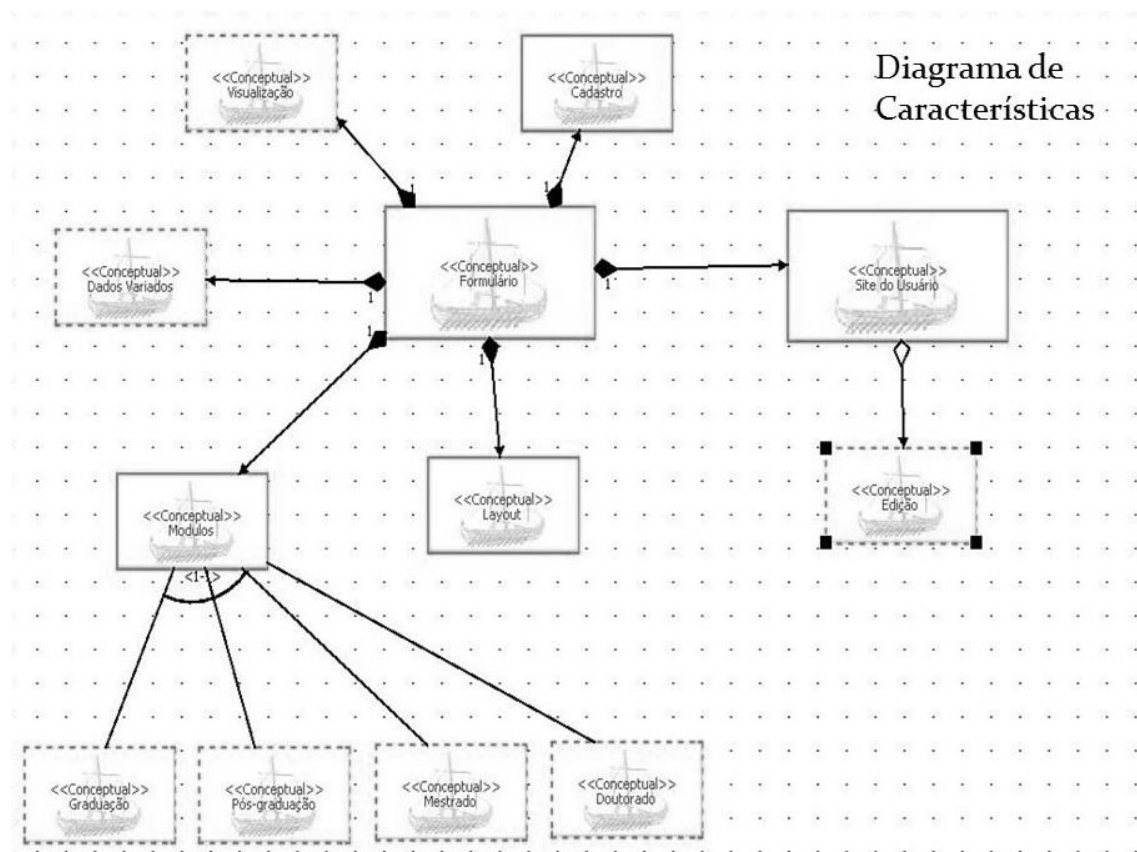
Smart é um framework GWT que permite não só utilizar a sua biblioteca de widgets abrangente, mas também usa estes widgets com o seu lado do servidor para o gerenciamento de dados.

SmartGWT foi utilizada nesta aplicação para melhorar a interface gráfica, já que somente o GWT não possuía uma interface muito satisfatória.

### 3.3 Modelos de Características

Foi criado em adição ao diagrama de classes, um modelo de características do domínio, isto é, um diagrama que represente todas as características de programas e aplicativos do domínio da aplicação. Este modelo representa todas as características obrigatórias e opcionais que devem conter em todas as aplicações desse mesmo domínio, que é o de geradores de sites. A notação utilizada no modelo foi Odyssey-Fex da Engenharia de Domínio CBD-Arch-DE.

O modelo ajudou para a compreensão do domínio do aplicativo que estava sendo desenvolvido e para um melhor entendimento sobre o que realmente precisava ser feito para que fosse alcançado o objetivo do projeto.



### 4. Trabalhos Relacionados

O KidWeb, é um trabalho que está diretamente relacionado com o Cloud University Builder. Esta ferramenta tem como objetivo a criação de sites para crianças, enquanto o CUB tem como foco os universitários. O KidWeb auxiliará as atividades desenvolvidas, com crianças da 3ª e 4ª séries do Ensino Fundamental.

## **4.1 Ferramentas do KidWeb**

### **4.1.1 Fine Artist**

O *Fine Artist* foi uma das ferramentas escolhidas para esse desenvolvimento. A escolha desta ferramenta se deu pela mesma adequar-se ao público alvo do projeto proposto, possuir ícones identificados com a realidade das crianças e proporcionar à mesma um ambiente de autoria. A ferramenta Fine Artist permite à criança criar histórias em quadrinhos, figuras animadas, cartazes e adesivos.

### **4.1.2 HotDog**

O *HotDog Junior* é um editor HTML para crianças em Inglês criado pela Sausage Software. Possui personagem próprio, um cachorro que representa as funções do editor e sua tela de abertura usa recursos de multimídia.

## **4.2 Funcionalidades**

As funcionalidades definidas para o editor foram baseadas no FrontPage, são elas: "Inserir Marcadores", "Formatar Parágrafo", "Recortar", "Copiar", "Colar", "Inserir Cor de Fundo", "Salvar", "Imprimir", "Visualizar Página" e "Ajuda". Estas foram selecionadas por serem as tarefas mais básicas que possibilitam a elaboração de websites.

O software desempenha funções básicas como: criar um novo documento HTML baseado no template escolhido, salvar este documento, inserir figuras, texto, hiperlinks, formatação e outros. A Figura 5 ilustra a interface do KidWeb com um modelo de template para elaborar história em quadrinhos.

## **4.3 Relacionamentos com CUB**

Pode-se perceber que o KidWeb e o CUB, embora tenham foco de clientes diferente, ambos possuem o mesmo objetivo que é a geração de um web site. Em ambos temos pré-definições de layout, se é usado HTML. Algumas funcionalidades do KidWeb são as propostas futuras de melhora no CUB, como inserção de figuras, hiperlinks entre outros.

## **5. Considerações Finais**

### **5.1 Conclusões**

Visto que o objetivo do projeto era um aplicativo que gerasse sites acadêmicos, o objetivo foi concluído. Acredita-se que o aplicativo será de boa ajuda ao se tratar de gerador de web site.

## **5.2 Limitações**

Apesar de algumas limitações, a estrutura projetada do aplicativo em si, facilitará muito na construção de um web site acadêmico, pois o usuário não vai precisar ter conhecimento técnico nenhum. O que pode ser feito para melhorar, é a implementação de mais funções dando mais liberdade ao usuário, fazendo com que o site fique mais completo.

## **5.3 Trabalhos Futuros**

Há algumas limitações no aplicativo, como as poucas opções de layout propostas para o usuário, e também não há a opção de escolha de cor. Os layouts propostos foram baseados a partir de uma pesquisa em que identificava a estrutura da maioria desses sites acadêmicos. Porém não abrange uma grande variedade devido à vasta quantidade de web sites existentes.

Visto esses pontos, a proposta futura é uma melhoria no que se diz respeito à forma em que essas opções são escolhidas, podendo assim dar mais liberdade ao usuário.

Outra questão a ser tratada futuramente é a questão de compactar os arquivos gerados pelo usuário, e a rodar o programa ao lado do servidor.

**References:**

- [1] Artigo “Representação de Variabilidades em Componentes de Negócio no Contexto da Engenharia de Domínio” de Regina F. Oliveira, Ana Paula T. B. Blois, Aline Vasconcelos e Cláudia Werner
- [2] GOOGLE, Google Web Toolkits. Disponível em <<http://code.google.com/intl/pt-BR/webtoolkit/overview.html>>
- [3] GOOGLE, Smartgwt. Disponível em <<http://code.google.com/p/smartgwt/>>
- [4] SANJIV JIVANS, Smartgw. Disponível em <[http://www.jroller.com/sjivan/entry/smartgwt\\_1\\_0\\_released](http://www.jroller.com/sjivan/entry/smartgwt_1_0_released)>
- [5] SBC, Sociedade Brasileira de Computação. Disponível em <<http://www.sbc.org.br/index.php?language=1&subject=60&content=downloads&id=26>>
- [6] BDBCOMP, Simone Creminácio, Disponível em: <<http://www.lbd.dcc.ufmg.br/bdbcomp/servlet/Autor?id=1748>>