



## Aplicação para Formatação Automática de Projetos

Caetano Finisterre Burjack da Silva<sup>1</sup>, Cláudio de Castro Monteiro<sup>2</sup>

<sup>1</sup>Graduando de Sistemas para Internet – IFTO. e-mail: caetano.burjack@gmail.com

<sup>2</sup>Professor Doutor do Instituto Federal de Educação, Ciência e Tecnologia de - IFTO. e-mail: ccm@ifto.edu.br

**Resumo:** Considerando o tempo e o trabalho gasto na formatação dos principais projetos acadêmicos, este projeto surge com o objetivo de criar uma aplicação *online* que permita ao acadêmico focar-se basicamente no conteúdo, deixando os detalhes da formatação para a aplicação aqui proposta. Desta forma, diferentemente do que ocorre com as ferramentas existentes, de uso corriqueiro, o usuário não precisará formatar cada parte do projeto acadêmico, apenas delimitá-las, para que a aplicação formate-as automaticamente e adequadamente, considerando as normas propostas pela ABNT (Associação Brasileira de Normas Técnicas), gerando ao final um arquivo em PDF. Outra característica, que adicionará ergonomia ao processo de confecção dos projetos acadêmicos, é o fato de tratar-se de uma aplicação *online*, que rodará diretamente do navegador, podendo ser acessada de qualquer lugar através da Internet.

**Palavras-chave:** acadêmico, ferramenta, formatação, internet, projetos

### 1. INTRODUÇÃO

Todo acadêmico, em algum momento durante sua formação, se vê obrigado a formatar seus projetos acadêmicos sob normas muito bem estabelecidas, especificadas pela ABNT (Associação Brasileira de Normas Técnicas). Tais normas são objeto de frustração para a maioria destes, já que esta tarefa é extremamente maçante e demanda muito tempo. Nesta ocasião, o acadêmico fica sujeito a mais dois trabalhos, além da pesquisa propriamente: compreender as normas da ABNT e aprender a usar a ferramenta de texto de sua escolha. Apesar dos inúmeros manuais e referências espalhadas pela Internet para auxiliar nesta tarefa, tanto no tocante as regras, quanto no uso das ferramentas, ainda faz-se necessário muito esforço para que a formatação fique a contento. Diante deste cenário surge um questionamento. Porque não, desenvolver uma aplicação que se distancie das existentes, por ser de fácil manuseio e específica para a formatação de projetos acadêmicos e que permita ao acadêmico, concentrar-se no conteúdo da pesquisa, sem necessariamente preocupar-se com detalhes da formatação?

Este trabalho tem o objetivo de responder a esta pergunta através da implementação de uma aplicação, *online*, capaz de receber o conteúdo da pesquisa e aplicar todas as regras cabíveis, segundo as normas impostas pela ABNT. Nesse sentido o usuário precisará apenas delimitar as partes do trabalho acadêmico, para que a aplicação formate-as adequadamente.

Toda a substância responsável por justificar o desenvolvimento desse projeto será retirado primeiramente das deficiências das aplicações existentes para formatação de textos mais usadas e das necessidades específicas da formatação de projetos acadêmicos.

O restante desse documento está dividido como segue. O capítulo 2 apresenta trabalhos relacionados. O capítulo 3 apresenta os materiais e os métodos utilizados. O capítulo 4 apresenta as características e os requisitos do projeto, bem como as ferramentas e o paradigma de programação empregados. Também é apresentado detalhes da modelagem e da análise de requisitos. O capítulo 5 apresenta detalhes da implementação do sistema. Por fim, o capítulo 6 apresenta a conclusão e sugestões de trabalhos futuros.

### 2. TRABALHOS RELACIONADOS

Existem algumas ferramentas, gratuitas ou pagas que possibilitam a formatação de projetos em geral, porém todas apresentam a limitação de não serem específicas para esta tarefa. Dentre as existentes, foram analisados três sistemas. Primeiro, o MSWord (Grover, 2007), da Microsoft,



processador de texto com interface gráfica que conta com recursos altamente profissionais para os mais variados tipos de texto. Segundo, o Write (Costa, 2007), que é um processador de texto similar ao da Microsoft, geralmente vinculado a distribuições Linux. E terceiro, o Latex (Griffiths, 1997), que funciona sobre a linguagem Tex, através de comandos de mais baixo nível para tornar possível a formatação. Estes são provavelmente os mais utilizados nos dias de hoje. Dessa forma, foi levantada uma série de dados que serviram de parâmetros para as fases posteriores. A tabela 1 mostra algumas comparações entre os sistemas analisados e o sistema proposto.

Tabela 1 - Comparação entre alguns sistemas para formatação de projetos.

<i>Software/</i> Características	MSWord	Write	Latex	Aplicação proposta
É exclusivo para formatação de projetos acadêmicos?	Não	Não	Não	Sim
Possui alguma ferramenta que auxilie o usuário na formatação de projetos acadêmicos, obedecendo as normas propostas pela ABNT?	Não	Não	Não	Sim
Possui interface <i>online</i> ?	Não	Não	Não	Sim
É gratuito?	Não	Sim	Sim	Sim

### 3. MATERIAL E MÉTODOS

#### 3.1 Materiais

NBR 14724 (ABNT, 2011): Norma da ABNT que dispõe das normas metodológicas para concepção de trabalhos acadêmicos;

Grails (Brown, 2009): *Framework* de alta produtividade para desenvolvimento de aplicações *web*;

Groovy (Judd, 2008): Linguagem de programação sobre a qual o Grails trabalha;

PostgreSQL: (Momjian, 2000) Sistema gerenciador de bancos de dados;

Netbeans (Oracle, 2012): Ambiente de desenvolvimento integrado, que oferece todas as ferramentas necessárias para criação de aplicativos.

Tex(Knuth, 1993): É um sistema desenvolvido para a impressão de texto científico e fórmulas matemáticas bastante popular para os acadêmicos de matemática, ciência da computação, economia, dentre outros da área de exatas. Geralmente não se usa tex puramente, mas sim através de macros.

Latex: Sistema composto por um conjunto de macros para Tex, utilizado para formatação de textos. O script feito em *l*átex possui a extensão “tex”.

MSWord: Processador de texto da Microsoft. Permite criar, editar e formatar os mais variados tipos de texto, e para isso conta com recursos altamente profissionais.

Write: Processador de texto, similar ao MSWord, porém gratuito. Geralmente está vinculado a distribuições linux.

#### 3.2 Métodos

A metodologia usada nesse trabalho é apresentada abaixo, dividida em quatro etapas.

1ª. Levantamento de requisitos:

- Pesquisa através de bibliotecas e Internet, de bibliografia relacionada ao tema;
- Análise das normas metodológicas da ABNT para trabalhos acadêmicos (NBR 14724);
- Análise das necessidades do usuário;
- Análise dos trabalhos relacionados.

2ª. Desenvolvimento de protótipos:

- Uso do *framework* Grails, e da linguagem de programação Groovy,
- Uso do PostgreSQL, como sistema gerenciador de banco de dados,

- c) Integração da aplicação com Latex, que atuará na formatação do trabalho acadêmico e concepção do PDF,
- d) Análise de ferramentas adicionais caso necessário,
- e) Concepção de protótipos.

3ª. Análises e Testes na fase de prototipação:

- a) Análise da integração com Latex,
- b) Análise das funcionalidades do sistema,
- c) Elaboração de questionários com o objetivo de medir a eficiência da aplicação proposta em relação às existentes de uso comum, abordando características como facilidade de uso, tempo gasto em determinadas tarefas e ergonomia de modo geral.
- d) Testes práticos com acadêmicos do IFTO e outros usuários em potencial. Serão definidas tarefas a serem realizadas tanto na aplicação aqui proposta quanto em outra aplicação escolhida pelo próprio acadêmico. Dessa forma será medida a vantagem de se usar a aplicação aqui proposta em relação a que o usuário normalmente usaria para confeccionar seus trabalhos acadêmicos. Ao final dos testes, os acadêmicos serão submetidos a questionários.
- e) Elaboração de relatório com base nos questionários.

4ª. Finalização do projeto e concepção da aplicação:

- a) Desenvolvimento da aplicação final,
- b) Desenvolvimento da documentação da aplicação.

A pesquisa se encontra atualmente na terceira etapa, portanto é importante salientar que caso as etapas subsequentes não sejam atingidas satisfatoriamente, poderá fazer-se necessário voltar a algumas fases anteriores.

## 4. PROPOSTA

Neste capítulo é apresentado, o projeto proposto, as atividades realizadas, os resultados obtidos até a atual fase do desenvolvimento, e o que se espera ao final do projeto.

### 4.1 Projeto

O projeto aqui proposto consiste no desenvolvimento de uma aplicação que possibilite ao usuário, criar, formatar e editar seus projetos acadêmicos, sem necessariamente conhecer as normas da ABNT. Espera-se uma aplicação *online*, que rode diretamente do navegador, que seja especialmente simples, intuitiva e específica para a formatação de projetos obedecendo às normas metodológicas impostas pela ABNT.

Para a concepção da aplicação foi seguido o paradigma de orientação a objetos, por apresentar vantagens quanto à abstração de dados, reutilização de código e extensão da aplicação.

A primeira etapa do projeto compreendeu o estudo analítico da Norma da ABNT, NBR 14724, que dispõe das normas metodológicas para concepção de trabalhos acadêmicos, a fim de se obter matéria para a fase de levantamento de requisitos.

Ainda na etapa inicial, foram definidas as ferramentas, bem como as linguagens que seriam utilizadas no processo produtivo do projeto. Como sistema gerenciador de bancos de dados foi escolhido o PostgreSQL. A programação foi implementada essencialmente com o framework de desenvolvimento ágil Grails, que roda sobre a linguagem de programação Groovy. Essa escolha baseou-se nas próprias características do Groovy\_ características estas, herdadas e melhoradas do Java \_, que além de permitir programação rápida e intuitiva, possui integração nativa com Java. Como ambiente Gráfico para desenvolvimento foi escolhido o Netbeans 7.1, por ser completo e gratuito. Foi também usado o sistema LATEX que consiste em um conjunto de macros para a linguagem Tex. O Latex é amplamente utilizado na produção de textos científicos, devido sua alta qualidade tipográfica. Outra característica importante, é que o Latex é um sistema gratuito e de código aberto, fato que influenciou na escolha. A camada de apresentação está sendo desenvolvida com HTML (Bowers, 2008), CSS (Bowers, 2008) e Javascript (Thau, 2001).

Na segunda etapa foram determinadas as necessidades da aplicação como um todo, envolvendo usuários, recursos e funcionalidades esperadas. Nessa fase foi adotado o modelo de desenvolvimento de *software* conhecido por Processo Unificado (Scott, 2003), onde se define os requisitos do usuário sistematicamente e os interconecta a fim de gerar o *software*. Dessa forma, foram considerados as necessidades do usuário e a norma da ABNT pertinente. Nessa mesma fase também foram analisados os *softwares* existentes, de uso corriqueiro, que permitem a criação de projetos acadêmicos, que são o MSWord, Write e Latex. Nesse processo se levantou os pontos fortes e fracos dos *softwares* em questão, e que serviram de base para a concepção do projeto aqui proposto.

A terceira fase envolveu a criação de protótipos e é a atual fase do projeto. As seções seguintes detalham as fases do projeto.

#### 4.2 Análise de Requisitos e Modelagem

Essa fase envolveu a análise do problema em questão, para obter a substância necessária ao entendimento do domínio do problema, funcionalidades e processos, e assim, refinar as especificações. Alguns problemas foram levantados, baseados em experiências de usuários com aplicações já existentes. Segue algumas abaixo:

- Dificuldade em formatar as referências do projeto,
- Dificuldades em formatar as notas de rodapé,
- Inconveniência em numerar imagens, tabelas e quadros.
- Dificuldade em produzir o sumário,
- Dificuldade em produzir índice remissivo, dentre várias outras.

Estas informações contribuíram para a criação da modelagem e mais tarde para os casos de uso. A principal ferramenta para a modelagem da aplicação foi a norma da ABNT, NBR 14724, de onde foram retiradas todas as classes de domínio necessárias para a aplicação aqui proposta. A figura 1 é parte da norma em questão.



Figura 1 - Estrutura de projetos acadêmicos retirado da NBR 14724

Cada elemento da figura 1 se transformou em um domínio, que será representado por uma tabela no banco de dados. Cada domínio por sua vez possui uma série de atributos, que compõem sua estrutura.





### 4.3 Casos de Uso

Nessa fase foram capturados os requisitos funcionais do sistema. Foram criados diagramas que representam funções do sistema, com base nas necessidades do usuário. A tabela 2 representa alguns dos casos de uso da aplicação, bem como sua descrição resumida.

Tabela 2 - Casos de uso básicos

Caso de Uso	Interação
Iniciar no Sistema	O usuário faz um cadastro, valida sua conta e inicia no sistema, com login e senha.
Iniciar um projeto	O usuário define o nome e o tipo do novo projeto. Assim feito, ele é redirecionado a pagina inicial do projeto, onde serão exibidos todas as partes do projeto. Dessa forma o usuário pode incluir as partes da forma que desejar.
Criar uma Capa	Considerando que o usuário esteja na pagina inicial do projeto, haverá um link “Criar” a frente de cada parte do projeto. Dessa forma o usuário clica sobre “Criar Capa” e é redirecionado a um formulário.
Excluir/Editar Capa	Considerando que o usuário esteja na pagina inicial do projeto, a frente da parte “Capa” do projeto haverá a opção de editar o projeto. Clicando sobre esta, o usuário será redirecionado a pagina de edição da capa, que também permite sua exclusão.
Gerar PDF	Considerando que o usuário esteja na pagina inicial do projeto, haverá no canto inferior direito da pagina um link para geração de pdf. Ao clicar, o usuário será direcionado a outra pagina onde poderá visualizar e baixar o pdf.

Excetuando os elementos textuais, todos os outros deverão ser apresentados ao usuário em forma de formulários. Neste sentido, o usuário precisará apenas preenchê-los, deixando o trabalho de formatação para a aplicação. A solução para os elementos textuais ficou a cargo de um editor de texto simples, implementando tags, similares as do HTML. Desta forma, ao invés do usuário formatar o texto, ele precisa apenas definir as partes deste, usando tags para isso, sem necessariamente conhecê-las, apenas clicando em botões autoexplicativos. A figura 2 representa um protótipo do que se planeja para o editor de texto da aplicação.





Figura 2 – Protótipo do editor de texto da aplicação.

## 5. IMPLEMENTAÇÃO DO SISTEMA

Esta é a fase em que se encontra atualmente a pesquisa. Toda a modelagem já foi criada e persistida no banco de dados. A interface gráfica para todos os elementos, bem como, telas de inserção, edição e exclusão de dados, já foi criada, exceto para os elementos textuais, constituídos pela introdução, desenvolvimento e conclusão. A figura 3 mostra a tela inicial do projeto, onde são exibidos todos os seus elementos de forma bem definida. Conforme a figura, para cada elemento único do projeto, como por exemplo, capa ou folha de rosto, é exibido a frente, um botão com a opção “criar”, que permite criar um novo elemento. Caso já tenha sido criado, os botões de opção “ver” e “editar” serão exibidos. Ao clicar sobre a opção “criar”, o usuário é direcionado a pagina de criação do elemento, que contém um formulário, como mostra a figura 4. A opção “ver” ainda está sendo implementada e pretende-se mostrar com esta, o elemento da forma que ficará em PDF. A opção “editar” direciona o usuário para um formulário semelhante ao de criação, onde poderá ser feito alterações ou a exclusão do elemento. Para elementos múltiplos, como por exemplo, banca, abreviaturas ou anexos, será exibido a frente, a opção “adicionar”, que permite criar o elemento. Os elementos criados são exibidos a frente em forma de lista (figura 3). Clicando sobre esses elementos, é exibida uma pagina que permite alterá-los ou excluí-los.

Toda a interface gráfica está sendo desenvolvido com Groovy, HTML, CSS e Javascript. O HTML é a linguagem de marcação utilizada para produzir paginas da web. O CSS é uma linguagem de estilo que aperfeiçoa o visual das paginas HTML. O Javascript adiciona algumas funcionalidades extras às paginas, que certamente não seriam conseguidas usando apenas HTML e CSS.

### 5.1 Processo de formatação

A formatação dos elementos do projeto e a geração do PDF se darão através do LATEX. Todo o conteúdo digitado nos formulários será armazenado no banco de dados e posteriormente embutidos em um script de extensão “tex”. Quando o usuário solicitar o PDF, seu script, conforme exemplo na figura 5, será executado pelo sistema LATEX que gerará o PDF formatado, seguindo as normas da ABNT. Todo esse processo será feito, ao olhos do usuário, a nível de navegador, portando, independe de qualquer outro *software* instalado em sua máquina para acontecer. Vale salientar que apesar desse processo já estar implementado, ainda faz-se necessário finalizar o editor de texto, para que seja possível gerar o PDF.



Figura 3 – Página inicial de uma monografia criada na aplicação proposta.

**Principal**

**Criar Capa**

**Instituicao** INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO TOCANTINS

**Faculdade**

**Curso** TECNOLOGIA EM SISTEMAS PARA INTERNET

**Autor** CAETANO FINISTERRE BURJACK DA SILVA

**Título** FORATA - Formatador Automático de Trabalhos Acadêmicos

**Subtítulo**

**Cidade** PALMAS

**Estado** TO

**Data** 2012 ▼

**Criar**

Figura 4 – Formulário de criação da capa do projeto



```
\begin{titlepage}
\begin{center}
\Large{\textsc{INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO TOCANTINS}} \\\
\textsc{TECNOLOGIA EM SISTEMAS PARA INTERNET}
}
\par\vfill
\Large{CAETANO FINISTERRE BURJACK DA SILVA}
\par\vfill
§ \bigskip
\Large{FORATA - Formatador Automático de Trabalhos Acadêmicos}
\par\vfill
\Large{PALMAS - TO\2012}
\end{center}
\end{titlepage}
```

Figura 5 – Código LATEX referente a capa do projeto exibida na figura 4.

## 6. CONCLUSÕES E TRABALHOS FUTUROS

Considerando a evolução do projeto, com base no que foi e no que há de ser feito, pretende-se concluí-lo na primeira quinzena de outubro.

Para a conclusão do projeto e concepção da aplicação da forma proposta, vale destacar o que ainda deverá ser feito:

- Materializar o conceito do editor de texto;
- Desenvolver o código javascript para conversão do conteúdo do editor em código látex;
- Desenvolver o *script* para executar o código látex e gerar o pdf;
- Desenvolver requisitos básicos de segurança para contas de usuário;
- Realizar outros ajustes comuns na fase de acabamento.

A partir desse projeto, uma gama enorme de possibilidades surge, primeiro pela complexidade do tema, segundo por necessidades que ainda podem ser descobertas. Dentre estas, duas valem enfatizar. A primeira seria desenvolver a interface para *desktop* da aplicação aqui proposta, com integração total com a mesma, possibilitando ao usuário trabalhar tanto *online* quanto *off-line*. A segunda seria a de desenvolver módulos para cada área acadêmica ou setores específicos, permitindo outros tipos de formatações além das impostas pela ABNT. A decisão agora cabe a quem encabeçar os projetos futuros.

Existem muitos editores de texto disponíveis nos dias de hoje, sejam pagos ou gratuitos. Porém na maioria dos casos estes foram concebidos para lidarem com o maior numero de necessidades possíveis do usuário, ponto que é positivo por um lado, e negativo por outro. Positivo por permitir ao usuário fazer o maior número de tarefas diferentes, e negativo por nem sempre desempenhar cada tarefa com a melhor qualidade possível ou até mesmo simplesmente com qualidade. Esse cenário foi o responsável pelo surgimento da pesquisa aqui proposta. Uma aplicação específica para a formatação de projetos, que fosse fácil e intuitiva, e porque não, *online*.

Considerando a atual fase do projeto, que é a de implementação, ainda há muito trabalho a ser feitos, porém, o que foi analisado e as conclusões tiradas, contribuirão em qualquer abordagem dessa natureza.

Por fim, pretende-se deixar como contribuição uma aplicação *online* de código aberto para a formatação de projetos que certamente será de grande valia aos acadêmicos do IFTO, bem como de outras universidades, que poderão usa-lo ou até mesmo continuá-lo.

## REFERÊNCIAS

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 14724**: Informação e documentação: Trabalhos Acadêmicos: Apresentação, 2011.





- Bowers, M.. **Profissional Padrões de projeto com CSS e HTML**. Alta Books, 1ed, 2008.
- Brown, J. e Rocher, G.. **The definitive guide to Grails**. Apress, 2ed, 2009.
- Costa, E. A.. **BrOffice.org – Da Teoria à Prática**. Brasport, 2007.
- Griffiths, D. F. and Higham, D. J.. **Learning Latex**. Society for Industrial and Applied Mathematics, 1997.
- Grover, C. **Word 2006: The Missing Manual**. O Reilly Media, Inc., 1<sup>th</sup> edition, 2007.
- Judd, C. M., Nusairat, J. F. e Shingler, J.. **Beginning Groovy and Grails: From Novice to Professional**. Apress, 2008.
- Knuth, D. E.. **TEX: The Program**. Addison-Wesley. Co, 1993.
- Momjian, B.. **PostgreSQL, Introduction and Concepts**. Addison-Wesley, 2000.
- Oracle. **Netbeans**. Disponível em: <[http://www.netbeans.org/index\\_pt\\_BR.html](http://www.netbeans.org/index_pt_BR.html)> Acesso em: 13 de julho de 2012.
- Scott, k. **O processo Unificado Explicado**. Bookman Editora, 1ed, 2003.
- Thau D.. **O livro de Javascript**. Ciência Moderna, 1ed, 2001.