

# ASCIIDOC – Livre-se de templates inúteis de documentação

29/07/2013 09:46

Quem segue o ABAPZombie há algum tempo deve saber que eu [abomino](#) a forma como os projetos costumam documentar aplicações criadas em ABAP. Acho que deixei bem claro o que eu não gosto, e dei diversas sugestões de melhorias para documentações [naquele post](#).

Neste ano, muita coisa mudou na minha vida profissional. Felizmente, fui parar em um lugar onde eu tenho abertura total para utilizar as ferramentas de desenvolvimento que eu e meus “*comparsas*” julgarmos serem as melhores. E na busca por uma ferramenta fácil e prática para documentar aplicações, eu descobri o [ASCIIDOC](#).

O **ASCIIDOC** é uma ferramenta gratuita, que interpreta arquivos de texto escritos em um determinado formato, e cria arquivos *bacanudos* em diversos formatos (html, pdf, epub...). A utilização é MUITO simples, e você consegue gerar arquivos bem profissionais sem se preocupar tanto com formatação em questão de minutos. Você pode ainda utilizar variáveis na hora de escrever o seu documento, para ajudar em mudanças futuras. Os templates são todos gerados com base em um CSS, portanto, você sempre escreverá o arquivo com o conteúdo do mesmo jeito, podendo criar diversos outputs diferentes.

Ok, eu sei que é mais fácil exemplificar. Vou resumir bem como a coisa toda funciona. Você abre um documento de texto, escreve isso aqui:

```
:toc2:
:numbered:
:blogname: ABAPZombie

Título do Documento sobre o {blogname}
=====

Este é um documento de exemplo.

Primeiro tópico
-----

Vamos falar sobre zumbis

Sub Tópico 1
~~~~~

Eles andam

Sub Tópico 2
~~~~~

Eles sangram

Sub Tópico 3
~~~~~

Eles fazem um barulho irritante

Segundo Tópico
-----

Vamos falar sobre pessoas que matam zumbis

Sub Tópico 1
~~~~~
```

Eles dão tiro de doze

Sub Tópico 2

Eles usam granadas

Depois, vai na linha de comando (Terminal ou Prompt de Comando), e roda o comando **"asciidoc <nomedoarquivo>.asc"**. Isso vai gerar um arquivo html com o template .css default, [igual a este aqui \(clique aqui para ver o documento gerado!\)](#). Simples, falaí? E você tem muitas opções para utilizar. Dê uma olhada nesta ["Cheatsheet" \("cola"\)](#), com os comandos mais utilizados.

Nas minhas andanças pela interwebs, eu descobri que o ASCIIDOC é utilizado por muita gente para escrever documentações técnicas, simplesmente porque ele é muito leve (um script escrito em python), possui comandos simples de serem aprendidos, e consegue gerar arquivos dos mais [diversos formatos](#).

Eu passei a utilizar há mais ou menos 2 meses, e agora eu não largo mais 😊 . O esquema de poder utilizar "variáveis" e criar as referências manualmente num *HTML-Style* é tudo que eu precisava. Os quadros [de alerta e de atenção](#) também me lembram livros de programação que li na época da faculdade, e guias de jogos estilo [gamefags](#).

A [documentação da ferramenta](#) é muito detalhada, e é feita com o próprio ASCIIDOC. Meio inception 😊 A [instalação](#) é bem simples, instalei aqui no meu OSX sem problemas. Se alguém instalar e tiver algum problema/dica, compartilhe nos comentários deste post.

Sei que é um pouco difícil sugerir uma mudança como essa "do nada" em projetos, mas faça um teste com o ASCIIDOC, e mostre para seus líderes. Quem sabe o cara não curte?

Enfim: usei, curti e resolvi compartilhar. Se você conhecer alguma outra alternativa legal para documentações, compartilhe também nos comentários.

Abraços!

## Comentários

**Fabio Pagoti** — 29/07/2013 10:12

Isso é bom hein. O resultado CSS é de muito bom gosto ainda por sinal.

O ASCIIDOC aparenta resolver muito bem o problema da ferramenta e estilo para documentação. Creio que um dos principais problemas referente a documentação não é apenas a ferramenta em si mas onde a etapa de documentar se encaixa no processo de desenvolvimento.

Como esta aplicação é instalada no cliente, não se resolve o problema de manter a documentação sempre atualizada pois entendo que cada um teria arquivos html/css individualmente nas máquina. Uma alternativa que já vi usarem é versionar arquivos de documentação usando o Git. Neste caso o problema seria resolvido e ainda ganha-se as vantagens (e complexidade) do git.

Claro que tudo tem um custo, a ferramenta passa a ser útil quando todos sabem usá-la bem. E neste caso, não vejo isso sendo feito de maneira eficiente sem o uso do Git. Logo os programadores deveriam saber além dos comandos do ASCIIDOC usar o Git em si. Acrescenta-se aí um repositório remoto privado e teremos um custo

financeiro que apesar de pequeno sempre tem que ser justificado para a mesma pessoa que não vê importância nem na etapa de documentação muitas vezes.

Falando em git, o github me vem a mente e agora eles tem uma Wiki criada para cada projeto, o que permite criar uma documentação centralizada na web sem tanto esforço (há ainda os comandos de formatação da Wiki). Acho esta abordagem mais simples para quem usa porém mais difícil de ser aceita por toda a cultura do GitHub.

A melhor dica que eu poderia dar referente a ferramentas de documentação é o uso do Confluence (Atlassian) atrelado ao Jira.

<https://www.atlassian.com/software/confluence/overview/team-collaboration-software>

O Jira é uma ferramenta de controle de chamados/stories/atividades/etc e o Confluence é uma ferramenta de Wiki muito bonita e incrivelmente fácil de ser usada. Para mim os produtos da Atlassian são os de maior qualidade considerando todos os softwares privados que já vi na vida (incluindo o R/3 e o Mac OSX). Quando a softwares abertos o WordPress é o primeiro na minha opinião.

Tudo da Atlassian é pago mas vale cada centavo. Com o Confluence não é preciso decorar comandos, tem-se uma documentação web em forma de Wiki e há todo um controle de mensagens entre usuários, além da integração com o próprio Jira, que é fantástico.

Para times que estão abertos a aprendizado contudo o ASCIIDOC pode trazer grandes benefícios.

Abraços!

**Mauricio Cruz** — 29/07/2013 10:35

Já ouvi falar do Confluence, mas nunca tive a oportunidade de experimentar, vou tentar dar uma olhada. Porém tanto ela como o ASCIIDOC são meras ferramentas, e eu concordo com você que a parte mais importante desse lance de documentações é a questão de como ela se encaixa no ciclo de desenvolvimento.

Eu já falei muito dessas paradas no post antigo “Documentações – O tempo perdido que não volta mais”, mas a perspectiva de mudança em larga escala é bem pequena.

Sei que o ASCIIDOC está me quebrando um galho enorme, e, além de ser free, achei muito simples a utilização.

Valeu pelos comentários 😊

Abs!