



Universidade de Brasília UnB
Faculdade UnB Gama FGA

Gerência de Configuração de Software
Relatório Final

Alunas: Ana Carolina Lopes da Silva

Elaine Cristina Meirelles Peronico

Sumário

1. ESPELHO POLÍTICO..... 3

2. FERRAMENTAS UTILIZADAS..... 3

3. RESULTADOS 4

4. DIFICULDADES 4

REFERÊNCIAS..... 5

1. Espelho Político

Website desenvolvido em *Ruby on Rails* durante as disciplinas de Métodos de Desenvolvimento de Software e Gestão de Portifólios e Projetos de Software da Universidade de Brasília.

Foi implementado com base no conceito de Dados Abertos, onde seu principal objetivo é mostrar, de forma transparente, para a população quais os políticos que realmente estão fazendo a diferença no Brasil. Isso é demonstrado através de gráficos com a quantidade e nomes de proposições de leis criadas por eles e rankings dos que mais elaboram essas proposições.

2. Ferramentas Utilizadas

- **GitHub:** Serviço de *Web Hosting* Compartilhado para projetos que usam o controle de versionamento Git. Possui funcionalidades de uma rede social como *feeds*, seguidores, wiki e um gráfico demonstrando os trabalhos dos desenvolvedores nas versões de seus projetos/repositórios.
- **Git:** Um sistema de controle de versão distribuído e um sistema de gerenciamento de código fonte, com ênfase em velocidade. Cada diretório de trabalho do Git é um repositório com um histórico completo e habilidade total de acompanhamento das revisões, não dependente de acesso a uma rede ou a um servidor central.
- **Jenkins:** Fornece serviços de integração contínua para desenvolvimento de software. É um sistema baseado em servidor que funciona em um *servlet container* como o Apache Tomcat. Ele suporta ferramentas de SCM incluindo AccuRev, CVS, Subversion, Git, entre outras.
- **Chef:** Ferramenta de gerência de configuração escrito em Ruby e Erlang. Utiliza uma DSL Ruby para criar *scripts* de ambiente. O chef é usado para simplificar a tarefa de configuração e manutenção de servidores de uma empresa e consegue descrever o que consiste a infra-estrutura do sistema e a maneira pela qual cada parte deve ser implantada, configurada e testada. Além disso, pode se integrar com plataformas baseadas em nuvem, como Rackspace, Internap, Amazon EC2, Cloud Platform Google, OpenStack, SoftLayer e Microsoft Azure, e automaticamente provisionar e configurar novas máquinas.

- **Vagrant:** Software que cria e configura ambientes de desenvolvimento virtuais. Pode ser visto como um *wrapper* de nível superior em torno de um software de virtualização como o VirtualBox, Vmware, KVM e Linux Containers (LXC), e em torno de software de gerenciamento de configuração, como Ansible, Chef, Salt e Puppet.

3. Resultados

O projeto Espelho Político não possuía nenhuma gerência de configuração. Sendo assim, o objetivo deste trabalho era iniciar essa gerência com ferramentas que visam facilitar o uso do ambiente de desenvolvimento do software e melhorar a qualidade do mesmo a partir da integração contínua.

Em virtude de dificuldades iniciais com a integração contínua, o primeiro passo completado foi a configuração da Vagrant, utilizada para manter em comum o ambiente de desenvolvimento para todos os desenvolvedores. O sistema operacional configurado na máquina virtual foi o Ubuntu 14.04 LTS, 64 bits, através da imagem base (conhecida como *box*) `ubuntu/trusty64`, encontrada em <<https://atlas.hashicorp.com/ubuntu/boxes/trusty64>>. Além disso, fez-se uso de *shell script provision* para a instalação dos pacotes necessários para o total e correto funcionamento do software.

Logo após, foi entregue a Vagrant o chef-solo, versão 12.3.0. Foram instalados diversos cookbooks, relacionados ao banco de dados e linguagem de programação. Além disso, foram feitas receitas e atributos.

Por último, foi implantada a integração contínua. Para isso foi utilizada a ferramenta Jenkins CI. Para a configuração do Jenkins foram utilizados plugins para o Github, repositório oficial do projeto, para o Git, utilizado para o controle de versão, além de plugins referentes a Ruby on Rails, linguagem utilizada no projeto.

4. Dificuldades

A dupla de integrantes desconhecia, inicialmente, a gerência de configuração em si, juntamente com as suas ferramentas de apoio. Fator que culminou em pequenas dificuldades de configuração e uso no começo do trabalho.

Outra dificuldade, que resultou na mudança da ferramenta de integração contínua, foi a de escrita do *script* de testes do arquivo `.travis.yml`, utilizado pela ferramenta Travis CI. Até o momento não foi descoberta a causa do erro.

REFERÊNCIAS

- GALE, Andy. **Getting started with chef: First steps with chef.** Disponível em: < <http://gettingstartedwithchef.com/first-steps-with-chef.html> >. Último acesso em junho de 2015.
- VAGRANT DOCS. **Getting started.** Disponível em: < <https://docs.vagrantup.com/v2/getting-started/index.html> >. Último acesso em junho de 2015.
- REYNOLDS, Rick. **Setting up Jenkins for GitHub, Rails & Rspec.** Disponível em: < <http://www.webascender.com/Blog/ID/522/Setting-up-Jenkins-for-GitHub-Rails-Rspec#.VYmjYend88o> >. Último acesso em junho de 2015.