

As tecnologias e as metodologias envolvidas nos ambientes de desenvolvimento e gestão colaborativa da MWEB-SIEXP (Módulo web de gestão dos dados experimentais da Embrapa)

Thiago de Oliveira Alves¹
Sérgio Aparecido Braga da Cruz²

Devido à armazenagem não estruturada e dispersa de dados originados por unidades pesquisadoras da Empresa Brasileira de Pesquisa Agropecuária (Embrapa), há a dificuldade no acesso aos dados resultantes da pesquisa e com isso uma duplicação de esforços. Devido a esse problema, originou-se um projeto liderado pela Embrapa Informática Agropecuária, denominado Gestão de dados experimentais da Embrapa (SIEXP), tendo como objetivo central, o estabelecimento do processo informatizado de aquisição, armazenamento, disponibilização e segurança dos dados de experimentos da Embrapa. Participam desse projeto as Unidades Piloto (Ups): Embrapa Arroz e Feijão, Embrapa Gado de Corte, Embrapa Agropecuária Oeste, Embrapa Clima Temperado, Embrapa Trigo, Embrapa Milho e Sorgo, Embrapa Semiárido, Embrapa Algodão, Embrapa Pecuária Sudeste, Embrapa Soja, Embrapa Mandioca e Fruticultura e Embrapa Suínos e Aves.

Membros de Processo e de Tecnologia da Informação (TI) dessas Unidades realizarão, durante o desenvolvimento do SIEXP, uma das primeiras iniciativas de trabalho colaborativo na Embrapa, com equipe geograficamente distribuída.

O MWEB-SIEXP será desenvolvido em uma plataforma web, usando as tecnologias baseadas na linguagem de programação Java, e contará ain-

¹ Faculdade de Tecnologia de Americana, thiago.alves@colaborador.embrapa.br

² Embrapa Informática Agropecuária, sergio.cruz@embrapa.br

da com três outros módulos: um para integração com dispositivos de medição (balanças, analisadores de umidade, termômetros, etc.), outro para integração com coletores de dados e o último, denominado “Migrador”, responsável pela migração de dados dispersos para a nova base centralizada do SIEXP.

Este trabalho tem como foco as tecnologias e metodologias envolvidas nos ambientes de desenvolvimento e gestão colaborativa do MWEB-SIEXP, apresentando e conceituando o papel de cada uma durante a execução do projeto. Realizado com base em metodologias ágeis (BECK et al., 2012), mais especificamente adaptadas do framework SCRUM (ARAÚJO, 2012), as quais implementam os princípios do manifesto ágil.

O SCRUM contém diversas siglas e alguns dos conceitos importantes são definidos no contexto dessa metodologia. *Product backlog* (HASMIK, 2012) é uma lista de requisitos que devem ser implementadas no projeto, neste caso, sendo dividida em três tipos: *described* (descrita), *prioritized* (priorizada) e *detailed* (detalhada); *Product owner* é o solicitante/proprietário do produto; *Sprint Backlog* é a definição de tarefas, criado a partir do *Product backlog*.

O processo de desenvolvimento do software será dividido em seis etapas que, de acordo com a metodologia adotada, deverá ser repetido ciclicamente até o término da implementação do software. Ao final de cada etapa é produzido um conjunto de artefatos que darão suporte ao desenvolvimento do software. Essas etapas estão sob o contexto de uma Sprint, com duração previamente estabelecida de 1 mês (período no qual é realizado o desenvolvimento e incremento do produto), resultando na implementação de uma parte funcional do software. Primeiramente ocorre o levantamento dos requisitos, realizado pelos responsáveis das unidades piloto, resultando em um *described product backlog*, o qual contém uma lista dos requisitos requeridos pelo *product owner* (solicitante). Em seguida há a priorização dos requisitos, definindo uma classificação segundo a importância do requisito para o solicitante, desse modo gerando um *prioritized product backlog*. Na terceira etapa é produzido um conjunto de histórias de usuário, resultando no *detailed product backlog*. As etapas seguintes são realizadas exclusivamente pela equipe de desenvolvimento, nas quais cada item de história de usuário determina a realização de uma ou várias tarefas de *Sprint Backlog* para que seja possível um compartilhamento de atividades mais efetivo. Essas tarefas podem ser de especi-

ficação técnica (“Controlar a quantidade de logins simultâneos para evitar uma exceção”) ou mais direcionado ao usuário (“Permissão de volta entre telas”).

A lista principal de todas as funcionalidades desejadas no produto será gerenciada por meio da ferramenta Gerenciador de Projetos de Desenvolvimento de Software (GPDS..., 2012), possibilitando o suporte à organização de execução do MWEB-SIEXP. Em seguida, há a execução das tarefas, com o auxílio de algumas ferramentas open source, como Eclipse IDE (Indigo), OpenJdk 6 (1.6.0_23), Apache Ant (1.8.4), Jboss AS 7 (7.1.1.Final), Log4J (1.2.17), PostGreSQL (9.1), PostGIS (1.5.3), postgresql-9.1-902.jdbc4 (9.1 Build 902), PrimeFaces (3.3.1), Selenium Java Client (2.24.1), Junit (4.10), Hibernate (4.0.Final) além do uso de linguagens de marcação, focado ao lado do cliente, HTML 5 e CSS 3. Para o gerenciamento de mudanças, optou-se pela utilização do Jenkins (KAWAGUCHI; MOLTER, 2012), provendo a integração contínua do software e de forma automatizada, aumentando a produtividade da equipe de desenvolvimento. Após a integração da parte do software finalizada no Jenkins, esta será implementada em um servidor de pré-homologação, tendo a finalidade de oferecer acesso direcionado à equipe de desenvolvimento, havendo a discussão e possível manutenção das tarefas apresentadas entre os membros. Caso não ocorra erros com a tarefa na etapa de pré-homologação, seguirá ao servidor de homologação, no qual possibilita-se o acesso e validação pelos usuários finais do software parcialmente implementado. Nesse momento ocorre a finalização do ciclo temporal do Sprint atual.

Este resumo tem como objetivo dar uma visão global da trajetória de desenvolvimento do MWEB-SIEXP, as ferramentas e as metodologias utilizadas, cujo projeto tem a previsão de término em 2015.

Durante a verificação das 6 etapas pertencentes ao Sprint, identificaram-se alguns fatores determinantes para a previsão de concretização destas, como: identificação, escolha e integração das ferramentas de desenvolvimento, caracterização dos papéis das pessoas envolvidas no sistema, conhecimento prévio das rotinas, identificando possíveis resultados e consequências. Esses fatores permitem o maior controle de tempo x esforço durante o cronograma proposto pelos gestores, havendo a possibilidade de servirem como referências importantes no desenvolvimento de futuros projetos.

Este trabalho foi concluído com a ajuda do pesquisador Leonardo R. Queiroz.

Referências

ARAÚJO, I. *Faq scrum: Scrum*. 2012. Disponível em: <<http://blog.myscrumhalf.com/2012/02/o-que-e-sprint-%E2%80%93-faq-scrum/>>. Acesso em: 26 set. 2012.

BECK, K.; GRENNING, J.; MARTIM, R. C.; BEEDLE, M.; HIGHSMITH; MELLOR, S.; BENNEKUM, A. van; HUNT, A.; SCHWABER, K.; COCKBURN, A.; JEFFRIES, R.; SUTHERLAND, J.; CUNNINGHAM, W.; KERN, J.; THOMAS, D.; FOWLER, M.; MARICK, B. **Manifesto for Agile software development**. 2012. Disponível em: <<http://agilemanifesto.org/iso/ptbr/>>. Acesso em: 1 out. 2012.

GPDS - Gerenciador de Projetos de Desenvolvimento de Software. 2012. Disponível em: <<https://www.gpds.cnptia.embrapa.br>>. Acesso em: 24 set. 2012.

HASMIK. **Artifact**: product backlog. 2008. Disponível em: <http://epf.eclipse.org/wikis/scrumpt/Scrum/workproducts/product_backlog_68345C16.htm> . Acesso em: 27 set. 2012.

KAWAGUCHI, K.; MOLTER, T. **Meet Jenkins**: Jenkins. 2012. Disponível em: <<https://wiki.jenkins-ci.org/display/JENKINS/Meet+Jenkins>>. Acesso em: 26 set. 2012.