

Utilizando a REST API e o Recebimento de E-mails do Redmine para a Integração com Ferramenta de Atendimento ao Usuário

Manuela Klanovicz Ferreira

Universidade Federal do Rio Grande do Sul – UFRGS, Centro de Processamento de Dados
Rua Ramiro Barcelos, 2574, 90035-003 Porto Alegre, RS

manuelakf@cpd.ufrgs.br

Tipo de Trabalho: *Relato de experiência*

Resumo: *O Redmine tornou-se a principal ferramenta de controle de projetos de software no CPD da UFRGS, sendo adotado com sucesso desde 2009. Com isso, surgiu a necessidade de integrá-lo com a ferramenta de atendimento ao usuário, permitindo a vinculação das tarefas do Redmine com os chamados dos usuários que as originaram. Este trabalho demonstra como as interfaces alternativas de REST API e recebimento de e-mails do Redmine podem ser utilizadas para a criação, consulta e alteração de tarefas, permitindo sua integração. As dificuldades constatadas durante a utilização da classe PHP sugerida na documentação da REST API para alteração de tarefas foram superadas com sua utilização combinada com a capacidade de recebimento de e-mails do Redmine. Apesar dos problemas, a classe continuou a ser utilizada na integração para a criação e consulta de tarefas devido à facilidade por ela proporcionada ao tratar tarefas como objetos e tornar transparente as requisições HTTP feita à REST API.*

Palavras-chave: *Redmine, REST API do Redmine, Recebimento de e-mails pelo Redmine, Integração*

1 INTRODUÇÃO

O planejamento e controle das etapas de projetos de software pode ser facilitado pelo uso de ferramentas de gerenciamento de projetos. O Redmine¹ é uma ferramenta web de código aberto para gerenciamento de projetos cuja a adoção entre as universidades vem crescendo, como pode ser constatado pelos trabalhos de Amaral et al (2015) e Moura e Nascimento (2009).

Desde agosto de 2009 o Redmine foi adotado como principal ferramenta de controle de projetos de software dentro do CPD da UFRGS. Aproveitando o fato de o Redmine possuir integração com ferramentas de controle de versão, também foi adotado o SVN para controlar as alterações de software. A adoção do Redmine foi um sucesso, visto que em março de 2010 o CPD já comemorava a tarefa número 1.000. Atualmente o Redmine utilizado pelo CPD da UFRGS possui as seguintes características:

- versão 3.1;
- proximadamente 9.000 tarefas;
- customizações no tema;
- plugins para autenticação LDAP por grupos;
- plugin do Agile;
- plugin CKEditor (formatação de texto).

A fim de permitir a vinculação de tarefas criadas no Redmine com as solicitações dos usuários que as originaram, foi desenvolvida uma integração entre a ferramenta de atendimento de usuários, Qualitor², e o Redmine. Conforme Figura 1, o objetivo da integração é permitir que a partir da ferramenta de atendimento de usuários seja possível:

¹ Redmine: <http://www.redmine.org>

² Qualitor: <http://www.qualitor.com.br/site/content/qualitorfree/?id=21>

- criar tarefas no Redmine, recebendo como retorno o identificador da tarefa criada;
- inserir notas em tarefas do Redmine tendo como parâmetro o identificador da tarefa;
- pesquisar automaticamente tarefas de um determinado tipo que estejam fechadas para encerrar o respectivo chamado vinculado na ferramenta de atendimento.

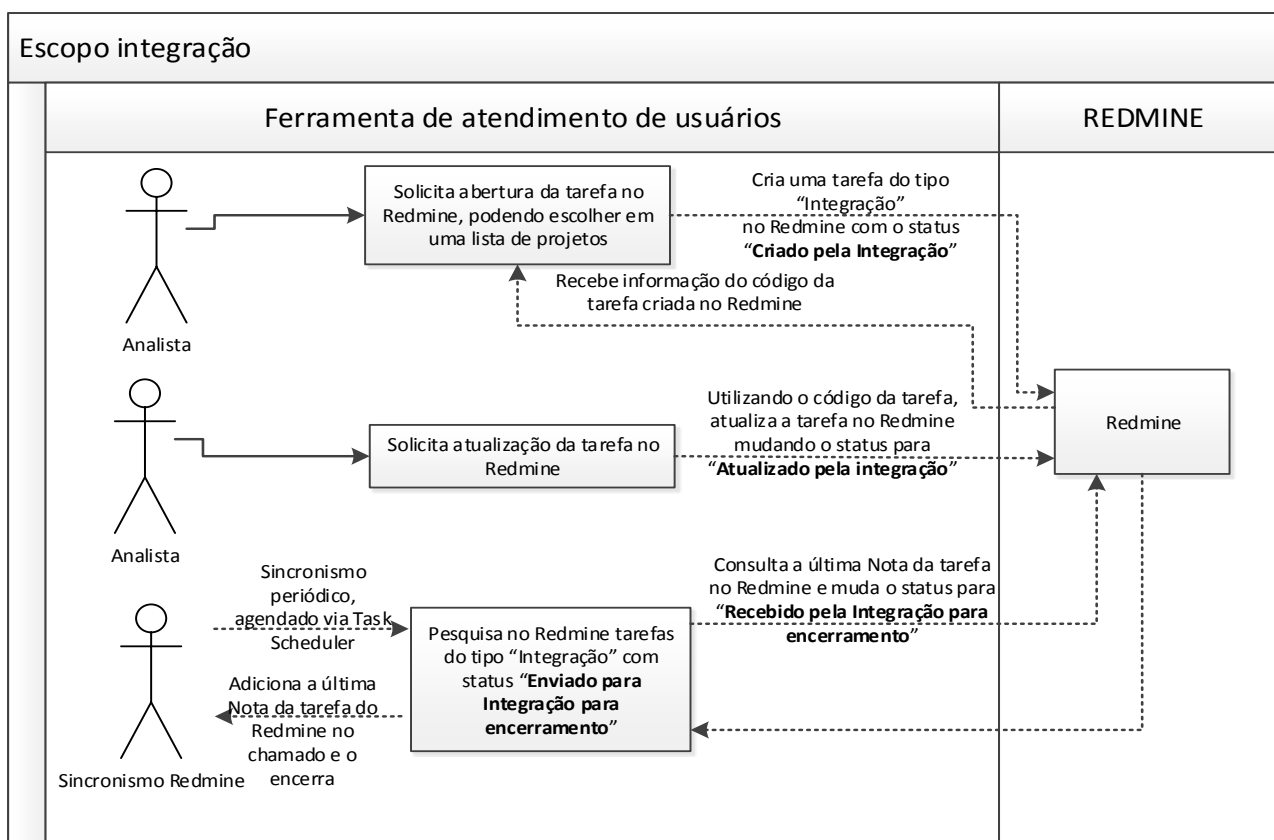


Figura 1 – Escopo da integração entre o Redmine e a ferramenta de atendimento aos usuários.

Para atender às necessidades de integração foi necessário utilizar a REST API combinada com a capacidade de recebimento de e-mails para a criação e alteração de tarefas do Redmine. Este trabalho descreve as facilidades e dificuldades encontradas ao longo deste processo. Na sessão 2.1 é descrita a utilização da REST API do Redmine e, na sessão 2.2, a utilização do recebimento de e-mails. Por fim, na sessão 3 são apresentadas as conclusões do trabalho.

2 INTEGRAÇÃO DO REDMINE

Inicialmente, tentou-se utilizar apenas a REST API do Redmine para permitir a integração com a ferramenta de atendimento ao usuário. Contudo devido a falhas na classe PHP disponibilizada para acesso à REST API, convencionou-se utilizar a REST API em conjunto com a capacidade de atualização de tarefas através do recebimento de e-mails pelo Redmine.

2.1 REST API do Redmine para tarefas

REST³ API é um padrão de comunicação entre sistemas que permite a consulta e alteração de informações em um sistema a partir de uma sintaxe de URL, com a informação sendo passada ou recebida por objetos como XML ou JSON via requisições HTTP. Desta forma, não é necessária uma linguagem de programação específica para utilizar a REST API de um sistema, pois qualquer linguagem de programação capaz de fazer requisições HTTP pode ser utilizada.

³ Representational State Transfer (REST): <https://pt.wikipedia.org/wiki/REST>

A REST API do Redmine⁴ permite a execução da maioria das operações relacionadas a tarefas, tais como: criação, modificação de conteúdo, inclusão de notas, modificação de status e atribuição. Além disso, na REST API do Redmine a autenticação é feita da mesma forma e provendo as mesmas autorizações em todos os projetos que o usuário teria ao acessar o Portal do Redmine, no qual é possível ativar a REST API em Configurações → Autenticação → “Habilitar a API REST”.

A ferramenta de atendimento ao usuário determinou que seria utilizado o PHP para realizar as requisições HTTP para acesso à REST API do Redmine. A documentação da REST API do Redmine provê uma classe PHP chamada ActiveResource.php (PHP ActiveResource, 2016) para facilitar a execução destas requisições. Utilizando-se esta classe, as tarefas do Redmine são tratadas como objetos e seus dados são tratados como atributos, sendo que toda a configuração e execução das requisições HTTP à REST API são feitas de forma transparente. A Figura 2 compara os comandos necessários para a criação e consulta de tarefas através da REST API do Redmine utilizando PHP com e sem a classe ActiveResource.php.

| Com a classe ActiveResource.php | Sem a classe ActiveResource.php |
|---|--|
| <pre>//criar tarefa \$issue = new Issue (array ('tracker_id' => 2, 'subject' => 'Tarefa criada com REST API', 'project_id' => '3', 'description' => 'Descrição tarefa criada'));</pre> | <pre>//criar tarefa \$params = "<issue>". "<tracker_id>2</tracker_id>". "<subject>Tarefa criada com REST API</subject>". "<project_id>3</project_id>". "<description>Descrição tarefa criada</description>". "</issue>"; \$ch = curl_init (); curl_setopt (\$ch, CURLOPT_URL, "https://redmine.br/issues.xml"); curl_setopt (\$ch, CURLOPT_POST, 1); curl_setopt (\$ch, CURLOPT_POSTFIELDS, \$params); \$res = curl_exec (\$ch);</pre> |
| <pre>//consultar tarefas com um determinado status \$issues = \$issue -> find ("all", array('status_id' => '3'));</pre> | <pre>//consultar tarefas com um certo status \$ch = curl_init (); curl_setopt (\$ch, CURLOPT_URL, "https://redmine.ufrgs.br/issues.xml?status_id=3"); \$res = curl_exec (\$ch);</pre> |

Figura 2 – Utilizando a REST API do Redmine no PHP: com e sem a classe ActiveResource.

Apesar da facilidade de utilização da REST API oferecida pela classe, foi detectado um problema: ao salvar as informações de um objeto, ou seja, ao enviar modificações feitas em um objeto instanciado localmente para serem registradas de fato no Redmine, a classe enviava todos os dados para o Redmine, não apenas os dados modificados desde o último salvamento. Isso fazia com que uma nota que já havia sido salva aparecesse repetida na tarefa do Redmine, mesmo ela sendo única dentro do objeto. A fim de contornar este problema, foi utilizada a interface de recebimento de e-mails do Redmine para realizar qualquer modificação em uma tarefa, sendo que a REST API permaneceu sendo utilizada apenas para a criação de uma nova tarefa, pois havia o retorno do identificador da tarefa, e para a consulta de informações das tarefas ou de uma lista de tarefas. A próxima sessão descreve como o Recebimento de e-mails pelo Redmine foi utilizado.

2.2 Recebimento de e-mails pelo Redmine

O recebimento de e-mails pelo Redmine⁵ permite a criação e modificação de tarefas caso o e-mail do remetente esteja autorizado no respectivo projeto do Redmine. Para ativar o recebimento de e-mails

⁴ Redmine REST API: http://www.redmine.org/projects/redmine/wiki/Rest_api

⁵ Redmine Receiving Emails: <http://www.redmine.org/projects/redmine/wiki/RedmineReceivingEmails>

é necessário habilitar esta funcionalidade em Configurações → E-mails recebidos → “Habilitar WS para e-mails de entrada” e configurar o servidor de e-mail para encaminhar e-mails para o Redmine através de uma requisição HTTP, conforme descrito em REDMINE Receiving Emails (2016).

A Figura 3 mostra dois exemplos do padrão de e-mail reconhecido pelo Redmine, onde para a criação de uma tarefa é necessário identificar o projeto através da linha destacada em negrito e, para atualizar a tarefa, é necessário informar o identificador numérico da tarefa precedido do caractere ‘#’ e entre colchetes, também destacado em negrito.

| Criando tarefas no Redmine por e-mail | Atualizando tarefas no Redmine por e-mail |
|--|---|
| Para: redmine@redmine.br Assunto: Título da tarefa que desejo criar | Para: redmine@redmine.br Assunto: RES[Projeto teste #0001] |
| Projeto: identificador-projeto | No corpo do e-mail vai a nota que se deseja inserir na tarefa |
| Esta tarefa está sendo criada com esta descrição. | |

Figura 3 – Criando e atualizando tarefas no Redmine por e-mail.

Ao criar uma tarefa, os e-mails listados em **Para** e **CC**, se registrados no projeto do Redmine, serão adicionados como **Observadores** na tarefa. Isso é possível apenas ao criar a tarefa, ao atualizar isso não funciona. Os arquivos anexados no e-mail serão automaticamente anexados na tarefa, desde que não excedam o tamanho máximo de arquivos no Redmine que se encontra configurado em Configurações → Geral → “Tamanho máximo do anexo”.

3 CONCLUSÃO

Este trabalho demonstra como interfaces alternativas da ferramenta Redmine podem ser utilizadas para a criação, consulta e alteração de tarefas, permitindo sua integração com outras ferramentas. Ao utilizar a classe ActiveResource.php, sugerida na documentação da REST API, foram encontrados problemas na alteração de tarefas. Entretanto, devido à facilidade oferecida pela classe ao tratar tarefas como objetos e tornar transparente as requisições HTTP, foi decidido manter a sua utilização para criação e consulta de tarefas. Para a alteração de tarefas foi implementada uma solução de contorno através da capacidade de recebimento de e-mails o que permitiu a efetiva integração do Redmine com a ferramenta de atendimento aos usuários.

REFERÊNCIAS

Amaral, M. M., Cassol, L. A., Ferreira, G. G. L., Pereira A. (2015). Workshop de Tecnologia da Informação e Comunicação das IFES – IX WTICIFES. Disponível em: <http://www.ixwticifes.ufba.br/modulos/submissao/Upload-275/66208.pdf>. Acesso em 16 mar. 2016.

Moura, M. H. D., Nascimento, H. A. D. (2009). Gerenciamento de Projetos com o Redmine. Workshop de Tecnologia da Informação e Comunicação das IFES – III WTICIFES. Disponível em: <http://www.iiwticifes.ufba.br/modulos/submissao/Upload/19688.pdf>. Acesso em 16 mar. 2016.

PHP ActiveResource (2016). Disponível em: <http://wiki.github.com/lux/phpactiveresource/>. Acesso em 16 mar. 2016.

REDMINE Receiving Emails (2016). Disponível em: <http://www.redmine.org/projects/redmine/wiki/RedmineReceivingEmails>. Acesso em 16 mar. 2016.