

Modelo Distribuído de Cooperação Multi-grupo através de uma REST API na Gestão de Informação do Site PET UFMA

Micael Machado Gomes¹, Antônio Moreira Pinto¹,
Gabriel Phelipe C. Freitas¹, Geraldo Braz Junior¹

¹Programa de Educação Tutorial do Ciência da Computação
Departamento de Informática - Universidade Federal do Maranhão (UFMA)
Av. dos Portugueses, 1966 Bacanga – 65.065-545 – Sao Luís – MA – Brasil

{micaellgoms, antoniomorara, gpcfreitas, ge.braz}@gmail.com

Abstract. *This paper discusses the redesign of the PET UFMA site to its implementation stages, according to the new architectural changes attributed to it. These range from the development of a new architecture based on a distributed model of cooperation between the PET Groups, to the construction of WordPress plugins that support such. All this is supported by the premise of sharing responsibility, which today belongs to one group, to all.*

Resumo. *Este trabalho aborda deste o reprojeto do site PET UFMA até suas fases de implementação, conforme as novas mudanças arquiteturas atribuídas à mesma. Estas vão desde a proposta de uma nova arquitetura baseada em um modelo distribuído de cooperação entre os Grupos PET, até a construção de plugins WordPress que dão suporte. Tudo isso sustentado pela premissa que consiste em dividir a responsabilidade, que hoje é de um grupo, à todos.*

1. Introdução

O site geral de grupos PET da universidade federal do maranhão, PET UFMA, tem como objetivo mostrar as ações do Programa de Educação Tutorial dentro e fora dos muros da Universidade, expondo como os grupos desenvolvem suas atividades e aplicam a tríade do programa: Ensino, Pesquisa e Extensão. No entanto, essa responsabilidade é de cada grupo, que é uma tarefa interna, feita sem um padrão, agregando dificuldades à coleta de informações a respeito de projetos e notícias relacionadas ao grupo. Assim, passou a ser necessário adotar uma abordagem eficiente para que a responsabilidade de criar conteúdo orgânico e dinâmico seja atribuída à todos os grupos, evitando sobrecarregar o grupo que detém gestão e a manutenção do site, o PET de Computação.

Considerando estas dificuldades o projeto de reformulação do site foi estipulado, propondo uma arquitetura de simples acesso, que possibilitasse aos outros grupos PET a capacidade de informar, sem o intermédio de um administrador, seus próprios dados. Além disso, um novo layout foi elaborado, levando em consideração requisitos de usabilidade, dedicados aos usuários que possuam baixa familiaridade em interações humano-computador, considerando que o Programa de Educação Tutorial atua em diversas áreas, que não são estritamente ligadas ao uso de computadores. Partindo dessa premissa, o WordPress continua adepto para realizar a função de centralizar a informação sobre os grupos, porém a proposta conta com uma mudança estrutural em seu Banco de Dados para que haja suporte ao modelo distribuído de coleta de dados em cada grupo.

2. Fundamentação Teórica

A principal função do reprojeto consiste na introdução de tecnologias que respeitem os requisitos de usabilidade, permitindo que uma maior quantidade de usuários esteja qualificada para utilizar o sistema. A versão anterior do site PET UFMA já obedecia tal especificação, pois o site era um *template* em WordPress, para que durante o processo de alimentação de conteúdo os moderadores não precisassem de conhecimento técnico em programação WEB. Desta forma, o WordPress continuou se mostrando a ferramenta mais adequada para gerenciar as publicações, levando em conta que seu Banco de Dados possui uma estrutura robusta, mas passível de mudanças, alvos do reprojeto, como a criação e gerenciamento de tabelas.

2.1. WordPress

O WordPress é um dos SGCW (Sistema de Gerenciamento de Conteúdo Web) [McKeever 2003] mais utilizados no mundo [Patel et al. 2011], sendo ele considerado ideal para projetos profissionais, gerando páginas web através de informações armazenadas em seu bancos de dados.

Em um blog WordPress, é possível publicar conteúdo a partir de *posts*, que de maneira prática são abstrações de alto nível para páginas web, tais como são publicadas em sites de notícias. Com isso é possível estabelecer acessibilidade e manutenibilidade, que é vital para que a produção de conteúdo abranja todos os grupos PET.

Também, no WordPress, é possível utilizar plugins. Um plugin pode ser definido como um software incremental responsável por disponibilizar funcionalidades não garantidas nativamente. Assim, conforme novas funções estavam se estabelecendo, novos plugins foram implementados e utilizados para satisfazê-las.

2.2. Conectando os Grupos com REST API

Como o site necessita informar a respeito de diversos grupos PET, quais tem suas informações contidas em domínios diferentes, é necessário centralizar as informações, para que elas sejam apresentadas ao mesmo tempo no domínio único PET UFMA. Levando isto em consideração foi levantada como requisito uma API.

Uma API (Interfaces de Programação de Aplicações) é uma camada de abstração responsável por intermediar a troca de informações entre dois dispositivos comunicantes com arquiteturas distintas, utilizando um protocolo conhecido. Tal que esta comunicação geralmente é apresentada de quatro formas: criar, ler, atualizar e deletar. Estas geralmente são conhecidas por seu acrônimo em inglês, CRUD. Durante a fase de reprojeto do site, a API escolhida foi a REST API [Masse 2011], uma tecnologia robusta amplamente utilizada no mercado.

Uma REST API funciona a partir do protocolo HTTP. Este protocolo executa as funções da interface a partir de rotas, por onde seus parâmetros são passados ao inserir uma chamada. Ao fazer esta chamada, a API retorna a função endereçada, devolvendo um arquivo .json ou funcionando apenas em back-end. Com isso, podemos garantir o compartilhamento de várias informações em domínios e arquiteturas diferentes através de uma linguagem padrão. À fins práticos, podemos, com a REST, buscar a informação de diferentes bancos de dados WordPress, desde que utilizando o plugin PET HOME, onde está contido o código fonte do roteamento.

3. Metodologia

O principal objetivo é utilizar as tecnologias apresentadas para gerar um site capaz de aplicar o modelo distribuído de coleta das informações nos grupos. Para isso foi necessário propor uma arquitetura que fosse capaz de transmitir os dados coletados para ao banco de dados.

A metodologia foi dividida em quatro etapas cascadas:

- Reprojeto, para validar os requisitos já consolidados no site antigo, além da elicitación de novos
- Implementación de plugins em cima de uma REST API, com acesso direto ao Banco de Dados
- Criação de um novo *template* moderno, funcional e responsivo, capaz de suportar o modelo
- Criação de Plugins, interfaces terminais, capazes de comunicar com o usuário

3.1. Reprojeto

O site em questão é um Sistema de Informação[de Oliveira et al. 2016], que apesar de ser uma solução ineficiente em relação ao problema destacado, passou pelo processo de Engenharia de Requisitos[Barbosa et al. 2009], tornando-se uma valiosa fonte de requisitos, fundamentais para nortear o reprojeto. Para o processo de elicitación dos requisitos, foi utilizada uma técnica tradicional, a *Análise de Documentos*[Pohl 2010], que se adequou ao projeto justamente por utilizar o principal artefato apurado, o site anterior ao reprojeto. Além disso, utilizando a mesma técnica, o documento de deliberações do 3º Encontro Maranhense de Grupos PET foi analisado, justamente por trazer em suas pautas a maneira com qual o programa se identifica, a partir de critérios objetivos elaborados democraticamente.

A partir da técnica de análise de documentos também foi elicitada a necessidade de adotar metodologias claras e consistentes de *Design*, para endossar os requisitos capturados. Para tal, foi utilizada a linguagem de Design, *Material Design*, que é incorporado no framework de front-end Materialize-css[Prabhu and Shenoy 2016] com o objetivo de tornar a disposição dos elementos mais agradável e acessível, garantindo um certo grau de fluidez e elegância. Além disso, foi utilizada a técnica de *Benchmark*, para que os requisitos de interface pudessem ser validados, usando como referência outros sites do Programa de Educação Tutorial de outros estados, como PET UFAL (Universidade Federal de Alagoas) e PET UFC (Universidade Federal do Ceará).

3.2. Arquitetura Proposta

O principal requisito do Sistema, em cima do que já foi apresentado, consiste em dividir a responsabilidade de gestão do conteúdo à todos os grupos PET, para assim garantir a independência aos grupos. Para tal, foi proposta uma arquitetura fundamentada em um modelo distribuído de cooperação, baseada em uma interface composta dos requisitos apurados, um usuário ficaria incumbido de inserir os dados. Com a finalidade de executar esta tarefa foi adotado o mecanismo de plugins, quais fazem parte do ecossistema de desenvolvimento WordPress, seguindo o modelo da Figura 1b.

Os plugins são integrados diretamente à UI WordPress, anexados à *Dashboard* da página de administrador, onde é possível visualizar e editar as informações. O PET

HOME se comporta como uma interface de gerenciamento dos dados gerais do grupo. Ele e o plugin PET MEMBERS possuem uma relação de inquilinismo, pois o HOME em sua instalação é responsável por criar as tabelas em seu banco de dados relacional, para que desta forma o plugin MEMBERS possa efetuar as operações de edição da informação dos integrantes.

3.3. Adaptação ao Banco de Dados

Na estrutura do WordPress foram realizadas mudanças estruturais, para que ele se adequasse a forma como os PETs são formados, com tutores e petianos. Para este objetivo ser alcançado é no mínimo necessário inserir as informações de cada petiano e tutor em nosso banco de dados. Essas informações abrangem desde o nome ao lattes do petiano, além de conter as informações do PET ao qual ele pertence, para que dessa forma os leitores tenham acesso aos dados do grupo.

Essa informação deve ser organizada de acordo com o padrão WordPress, utilizando a tabela que armazena os post do *template*. No entanto, durante a requisição, existe um processo de limpeza, para apresentar os dados armazenados, já que a tabela possui uma grande relação de informações. Para solucionar este problema, foi criada uma tabela que ficou responsável por armazenar as informações citadas anteriormente onde, aplicando a taxinomia do Banco, foi chamada de: `wp_custom_equipe`, representada na Figura 1a.

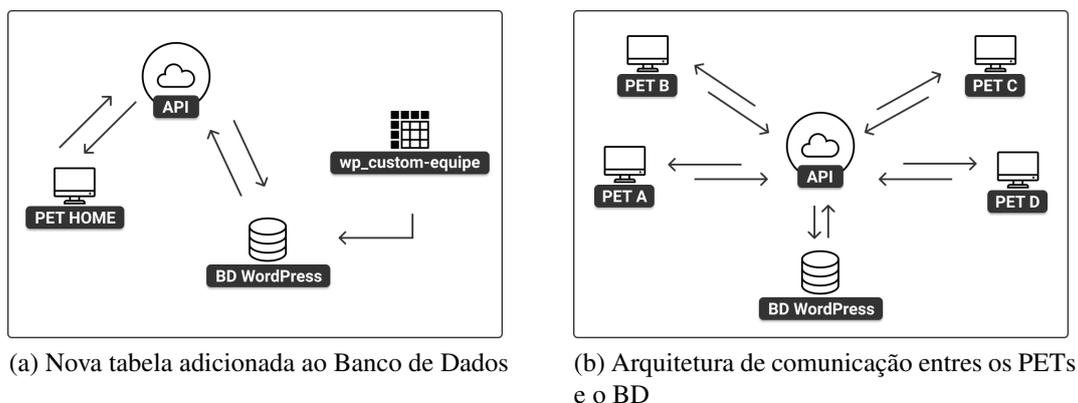


Figure 1. Arquitetura proposta

4. Resultados

Como produto final, foram implementados os dois plugins citados anteriormente: PET MEMBERS e o PET HOME, paralelamente ao desenvolvimento do *template*, ambos em funcionamento. O tema WordPress seguiu a estruturação padrão de páginas obrigatórias da comunidade, como a página de exibição de posts, aplicando os requisitos de interface levantados utilizando o *Material Design*, como demonstrado na Figura 2. Os plugins destacados contemplam todas as especificações técnicas necessárias para a efetivação da arquitetura propostas: criar uma tabela dentro do banco de dados do site para persistir as informações do PET, manter tais informações, com operações de inserção, deleção, atualização e recuperação.

O PET HOME é o plugin utilizado para realizar o cadastro das informações de cada PET. A ele também é atribuída a função de construir a tabela `wp-custom-equipe`

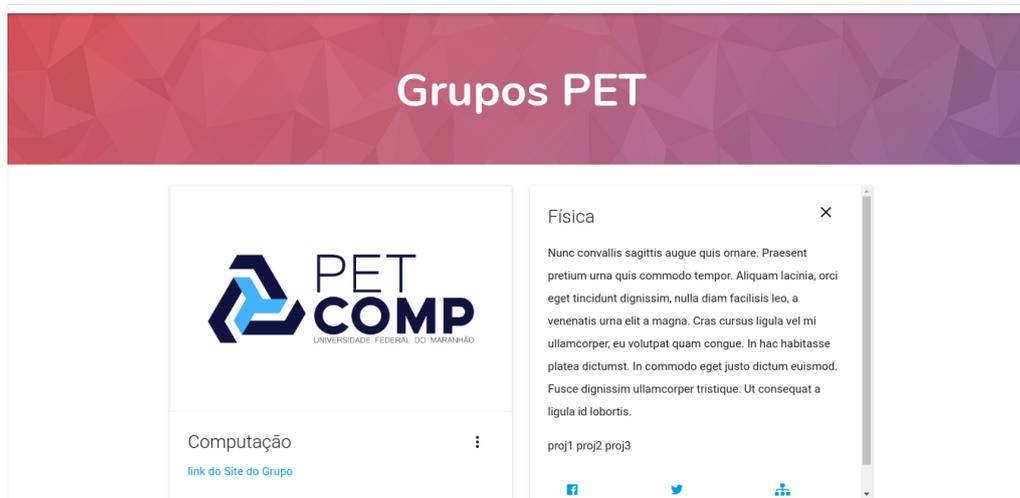


Figure 2. Página de Grupos, utilizando Cards na apresentação das informações

dentro do banco de dados, que é fundamental para que as informações de cada grupo seja propagada no SGCW. O cadastro das rotas da REST API também está dentro do código fonte do plugin, levando em consideração que somente há informações a serem recuperadas a partir do momento no qual um grupo está cadastrado. Sua interface requer as informações de nome do grupo, email, lista de projetos, logo, facebook, twitter e link da página web, assim como na Figura 3.

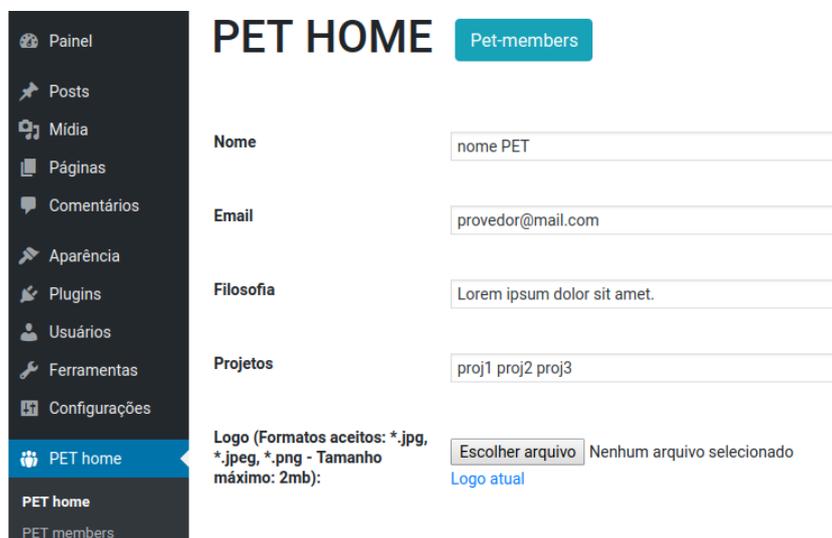


Figure 3. Interface do plugin PET HOME integrado com WordPress

O outro plugin, PET MEMBERS, tem como função realizar as operações de inserção, deleção e atualização dos dados dos membros do respectivo grupo PET. Assim é possível garantir, através de uma interface gráfica, três das quatro operações CRUD. Como a recuperação de informação funciona através do plugin PET HOME, é possível satisfazer todas as operações CRUD. Nem todas as operações foram estipuladas à API por questões de segurança, reconhecendo que um protocolo HTTP está livre para ser acessado em qualquer terminal.

Durante o final do reprojeto foram estabelecidas as seguintes métricas de avaliação: número de acessos durante o mês, para analisar as mudanças tiveram algum resultado quanto a visibilidade do site, e o fluxo de conteúdo publicado quantitativamente, com o objetivo de compreender se a publicação de conteúdo tornou-se acessível. Assim é possível concluir o futuro do reprojeto deve ser revisado ou não, de acordo com as novas necessidades dos usuários, tendo em perspectiva a manutenção do sistema.

5. Conclusão

Durante a execução do trabalho foi aplicado com sucesso o que foi levantado na fase de reprojeto, reaproveitando requisitos que na versão anterior do site haviam sido validados e elicitando novos. Após a reestruturação do site a nova arquitetura permitiu que os futuros requisitos possam ser satisfeitos criando plugins comunicantes com a API. Partindo deste ponto, pretende-se criar um terceiro plugin, capaz de cadastrar os eventos sediados por qualquer PET, de caráter local, regional e nacional.

O projeto está hospedado no link *pet.ufma.br* e no *GitHub*, onde ocorre o versionamento do software e é possível inspecionar o re-design. Ele está em sua fase final de desenvolvimento, na qual já apresenta o Programa de Educação Tutorial, tal como sua filosofia, história do programa e as publicações mais recentes dos grupos PET da UFMA.

Para o futuro está planejado o engajamento de integrantes de outros PETs, responsáveis por alimentar o site com informações. O PET de computação está elaborando um calendário para visitar os demais grupos, e dessa maneira auxiliar e capacitar os petianos a utilizar o sistema, tornando assim o site PET UFMA uma aplicação sustentável e independente de moderadores.

References

- Barbosa, G., Werneck, M., Assis, H., Fernandes, U., and Silva, I. (2009). Um processo de elicitação de requisitos com foco na seleção da técnica de elicitação. *VIII Simpósio Brasileiro de Qualidade de Software*, pages 159–173.
- de Oliveira, J. L., da Silva, A. F., Machado, B. N., Louzada, J. A., and Soares, P. M. (2016). Uma abordagem para engenharia de requisitos aplicada a sistemas de informação.
- Masse, M. (2011). *REST API Design Rulebook: Designing Consistent RESTful Web Service Interfaces*. ” O’Reilly Media, Inc.”.
- McKeever, S. (2003). Understanding web content management systems: evolution, lifecycle and market. *Industrial management & data systems*, 103(9):686–692.
- Patel, S. K., Rathod, V., and Prajapati, J. B. (2011). Performance analysis of content management systems-joomla, drupal and wordpress. *International Journal of Computer Applications*, 21(4):39–43.
- Pohl, K. (2010). *Requirements engineering: fundamentals, principles, and techniques*. Springer Publishing Company, Incorporated.
- Prabhu, A. and Shenoy, A. (2016). Introducing materialize. In *Introducing Materialize*, pages 1–9. Springer.