



# ERLANGMS: Uma Plataforma em Erlang/OTP para Modernização de Sistemas Legados através de uma Abordagem Orientada a Serviços na UnB

Everton Agilar<sup>1</sup>, Alysson Ribeiro<sup>1</sup>, Renato Ribeiro<sup>1</sup>

<sup>1</sup>Universidade de Brasília – Campus Universitário Darcy Ribeiro  
Caixa Postal – 70910-90 – Brasília – DF – Brasil

{evertonagilar,rcarauta,alyssondsr}@unb.br

**Resumo.** Nos últimos anos, a modernização dos sistemas legados da Universidade de Brasília (UnB) tem sido prioridade para o CPD/UnB. A Arquitetura Orientada a Serviços (SOA) surge como uma maneira de solucionar este problema, disponibilizando uma abstração de alto nível entre as aplicações e a camada de negócio ou serviço. Este artigo aborda este tema e descreve alguns resultados obtidos com o uso da plataforma ERLANGMS desenvolvido pelo CPD/UnB sob uma abordagem orientada a serviços que compreende um processo de modernização e um barramento aderente ao estilo arquitetural Representational State Transfer (REST). Mais especificamente, são discutidos os principais resultados alcançados, as quais destacam-se um barramento de serviços orientado a contrato de serviços; um kit de desenvolvimento (SDK) para criação de serviços; um processo denominado SMSOC para guiar as atividades de migração de sistemas legados; e alguns serviços de apoio como um proxy LDAP para unificar o login de usuários e um serviço de autenticação/autorização OAuth2 para os serviços em REST.

**1 Introdução** Os sistemas legados correspondem às aplicações que sustentam o funcionamento negocial de uma Instituição e consolidam a maior parte das informações corporativas [Bennett 1995, Bisbal et al. 1999]. Na Universidade de Brasília (UnB), há uma gama considerável de sistemas legados desenvolvido ao longo dos últimos 30 anos pelo CPD/UnB que consistem em um arcabouço de regras de negócios que são de vital importância para o pleno funcionamento da Instituição. Entretanto, com as sucessivas revisões nas regras de negócios para mantê-los alinhados com as necessidades e a obsolescência tecnológica desses sistemas, tornaram-se rígidos e inflexíveis, a ponto de serem de difícil manutenção e evolução.

De forma geral, os sistemas da UnB dividem-se em três áreas de negócio: área acadêmica, administrativa e de pessoal. A maioria desses sistemas foram construídos em diferentes linguagens de programação, arquiteturas e plataformas que não conversam entre si, a não ser, por meio do banco de dados. Durante muitos anos, a linguagem de programação VB foi a predominante. Os dois sistemas mais importantes escritos em VB são o *Sistema Acadêmico (SIGRA)* e o *Sistema de Pessoal (SIPES)*, sendo os demais sistemas escritos em VB.Net, C#, PHP, ASP e Java (a plataforma atual).

Neste cenário, as tradicionais práticas de manutenção deixam de atender às organizações, que buscam formas de reduzir os custos com a manutenção, maximizar a

integração entre os sistemas, torná-los mais flexíveis às mudanças de forma para prolongar sua vida útil e facilitar a evolução desses sistemas [Bennett 1995, Bisbal et al. 1999, Comella-Dorda et al. 2000].

Ao conduzir a modernização dos sistemas legados na UnB, optou-se por experimentar com a arquitetura orientada a serviços, particularmente seguindo o estilo arquitetural *REST*, já adotado em muitas Instituições devido ao aproveitamento da infraestrutura web existente e a facilidade para invocar serviços REST a partir de qualquer sistema [Fielding 2000, Kalin 2013]. Assim, foi criada a plataforma *ERLANGMS*, que compreende um barramento de serviços desenvolvido na linguagem *Erlang/OTP* para possibilitar a publicação de serviços; de um processo de modernização denominado *Software Modernization through Service Oriented Computing (SMSOC)*, para guiar os trabalhos de modernização e disponibilizar uma arquitetura de software padronizada para criação dos serviços.

Salienta-se que embora *SOA* seja um tema de crescente interesse por parte dos pesquisadores e da indústria, identificou-se a necessidade prévia de condução de um mapeamento sistemático para caracterizar a modernização de sistemas legados no contexto da manutenção de software [Agilar et al. ]. Desse modo, muitas decisões de design da plataforma proposta foram definidas a partir deste estudo prévio onde verificou-se que a maior parte das contribuições na literatura estão relacionados aos aspectos gerenciais da modernização de software (55,88% das publicações) e há poucos relatos de contribuições que descrevem (ou validem) técnicas ou ferramentas de modernização de software.

Este artigo apresenta a plataforma *ERLANGMS* e discute as experiências obtidas na implantação do barramento de serviços no CPD/UnB. Mais especificamente, são descritos os principais resultados alcançados, as quais destacam-se:

- Desenvolvimento de vários serviços web na linguagem Java para atender as demandas da UnB com o uso do processo de modernização SMSOC e o design de arquitetura subjacente, denominado de kit de desenvolvimento (SDK).
- Desenvolvimento de um serviço proxy LDAP ou Lightweight Directory Access Protocol para unificar o acesso dos usuários ao oferecer um "logon único" para os sistemas da UnB, como o Redmine, Webmail ou SEI, que se utilizam do protocolo LDAP para autenticação de usuários;
- Desenvolvimento de um serviço de autenticação OAuth2, integrado ao Sistema de Controle de Acesso (SCA) da UnB, que permite que os serviços possam ser consumidos pelos clientes utilizando a mesma infraestrutura de segurança já existente.

ATÉ AQUI

**2 Metodologia** Esta seção visa mostrar uma das funcionalidades do barramento de serviços da Universidade de Brasília, que é a implementação do protocolo OAuth 2.

Atualmente as aplicações presentes na Universidade de Brasília (UNB) não se comunicam, o que dificulta a disponibilização de serviços e o controle de acesso. O controle de acesso é feito utilizando o banco de dados. É definido o que o usuário pode acessar, armazenando a url do sistema e o as permissões de acesso a esta url.

Com a implantação do OAuth 2, como protocolo de autorização, visa facilitar o

controle de acesso as aplicação e facilitar a integração tanto para os sistemas da UNB quanto para sistemas externos que necessitem acessar informações.

O oauth 2 é um framework que veio com a ideia de se padronizar a maneira como as aplicações cliente acessam conteúdo restrito. O oauth 2 se baseia em tokens de acesso. Ao se autenticar, uma aplicação cliente recebe do servidor de autorização um token de acesso. A aplicação cliente pode acessar os recursos restritos, basta apenas enviar o token recebido pelo servidor e este valida ou não o acesso aos dados requisitados.

Outra importante característica do oauth 2 é a obrigatoriedade do uso de HTTPS para tráfego de informações entre o cliente e o servidor.

Com a implementação desse framework proporciona um controle de acesso das aplicações da UNB de maneira centralizada, além de possibilitar uma autenticação única e integrada dos sistemas internos e de algum sistema externo que precise de informações restritas da Universidade de Brasília.

Diversos serviços já estão disponibilizados como: a declaração de aluno, o catálogo de aluno, os cursos dos alunos, dentre outros. Todos os serviços utilizam o barramento como comunicação entre as aplicações cliente e os dados armazenados no servidor.

Os serviços seguem um padrão, chamado de catálogos, que fornecem a url de acesso, qual a linguagem do serviço, onde está o código do serviço, seu método HTTP de acesso e a forma de autorização.

Segue abaixo um exemplo de um dos serviços já disponibilizados.

### 3 References

#### References

- Agilar, E., de Almeida, R. B., and Canedo, E. D. A systematic mapping study on legacy system modernization.
- Bennett, K. (1995). Legacy systems: coping with success. *Software, IEEE*, 12(1):19–23.
- Bisbal, J., Lawless, D., Wu, B., and Grimson, J. (1999). Legacy information systems: Issues and directions. *IEEE software*, 1(5):103–111.
- Comella-Dorda, S., Wallnau, K., Seacord, R., and Robert, J. (2000). A survey of legacy system modernization approaches. Technical Report CMU/SEI-2000-TN-003, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA.
- Fielding, R. T. (2000). *Architectural styles and the design of network-based software architectures*. PhD thesis, University of California, Irvine.
- Kalin, M. (2013). *Java web services: up and running*. ” O’Reilly Media, Inc.”.