

Universidade Federal de Pernambuco

Graduação em Ciência da Computação

Centro de Informática

2017.1

Uma biblioteca Multi-Tenant para Flask **Proposta de Trabalho de Graduação**

Aluno: Bruno Resende Pinheiro (brp@cin.ufpe.br)

Orientador: Vinícius Cardoso Garcia (vcg@cin.ufpe.br)

RECIFE, 14 DE ABRIL DE 2017

Sumário

Contexto	3
Objetivo	5
Cronograma	6
Possíveis Avaliadores	7
Bibliografia.....	8
Assinaturas.....	9

Contexto

A Computação em Nuvem, também conhecida como *Cloud Computing*, é uma tecnologia definida pela entrega baseada no consumo de processamento computacional, armazenamento de dados e serviços através de uma plataforma Web disponibilizada pelo fornecedor contratado cuja cobrança é calculada de acordo com o consumo. Os serviços de computação em nuvem tem como vantagens por exemplo: reduzir custos de manutenção e execução de infraestruturas próprias como *datacenters*, aumentar escalabilidade do sistema, pagar apenas pelo recurso consumido [1].

Gigantes do setor começaram a investir em infraestrutura nas últimas décadas com o propósito de prover serviços de *Cloud* para empresas de menor porte e também consumidor final. Com isto consumidores tem mudado sua arquitetura de TI e reduzido custos [2].

A Computação na Nuvem pode ser dividida em três modelos de serviço: Software como Serviço (SaaS), Plataforma como Serviço (PaaS) e Infraestrutura como Serviço (IaaS). Além disso novos modelos foram criados (há controvérsias sobre estes) como o *Anything as a Service* (XaaS), ou seja, “tudo como um serviço” que inclui todos os “aaS” [3].

O modelo SaaS, Software as a Service, será a cerne deste trabalho, pode ser descrito como serviço online que fornece um software específico [4]. Aplicações de SaaS disponibilizam como vantagens uma arquitetura multi-inquilino, ou *Multi-Tenancy*, com customização para cada inquilino caso necessário. Além disso o modelo SaaS gerencia toda a necessidade de atualização de softwares e hardwares de forma que o consumidor terceira essa questão e apenas utiliza a aplicação sem a necessidade de cuidar da infraestrutura para o correto funcionamento da aplicação. Mais uma vantagem do SaaS é o baixo custo inicial, pois a cobrança é feita através de inscrições e não licenças como no modelo de compra de software convencional [5].

De acordo com o artigo publicado de Custódio e Junior [6], *Multi-Tenancy*, ou multi-inquilino, é uma abordagem organizacional do modelo SaaS. Algumas das características de aplicações que utilizam arquitetura *Multi-Tenancy* são:

- Isolamento das aplicações alocadas;
- Disponibilidade de configurabilidade para cada inquilino;
- Instância da aplicação e/ou banco de dados compartilhados.

Multi-Tenancy é uma arquitetura de sistema que permite vários clientes/empresas, também chamados de inquilinos, compartilhem serviços, aplicações, bancos de dados, entre outros. Desta forma abre uma possibilidade de redução de custos e personalização do sistema de acordo com o inquilino [7].

Um ambiente *Multi-Tenancy* provê benefícios como [8]:

- Maior utilização dos recursos de hardware;
- Manutenção da aplicação torna-se mais fácil de ser realizada e mais barata;
- Custos globais reduzidos, permitindo oferecer um serviço com menor custo quando comparado aos concorrentes.

Objetivo

Objetivos Gerais:

O objetivo deste trabalho é criar uma biblioteca *open source* que auxilie a conversão de projetos desenvolvidos em *Flask* em projetos *Multi-Tenant*. Consequentemente os desenvolvedores que utilizem o *framework* podem aplicar a arquitetura *Multi-Tenant* em seus projetos, de forma que os esforços não sejam concentrados na implementação ou implantação da arquitetura *Multi-Tenant* e sim nas regras de negócio dos sistemas.

Objetivos Específicos:

- Fazer uma análise sobre a arquitetura *Multi-Tenant*, de maneira a viabilizar a implementação dentro de suas restrições;
- Apresentar uma análise dos benefícios e malefícios da utilização dessa arquitetura;
- Implementar uma biblioteca *Multi-Tenant* para o framework Flask;

Cronograma

Atividades	Março	Abril	Maio	Junho
Revisão Bibliográfica	■ ■ ■ ■ ■ ■ ■ ■			
Preparação da Proposta do TG		■ ■ ■ ■		
Escrita do Embasamento Teórico			■ ■ ■ ■ ■ ■ ■ ■	
Desenvolvimento da Biblioteca			■ ■ ■ ■ ■ ■ ■ ■	
Escrita dos Resultados e Conclusão				■ ■ ■ ■ ■ ■ ■ ■
Elaboração de Apresentação				■ ■ ■ ■ ■ ■ ■ ■

Possíveis Avaliadores

Nelson Souto Rosa (nsr@cin.ufpe.br)

Kiev Santos da Gama (kiev@cin.ufpe.br)

Bibliografia

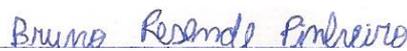
- [1] AMAZON WEB SERVICES. **O que é a computação em nuvem?**. Disponível em: <<https://aws.amazon.com/pt/what-is-cloud-computing/>>. Acesso em: 14 abr. 2017.
- [2] TAURION, Cezar. **Computação em nuvem: Transformando o mundo da Tecnologia da Informação**. 1 ed. Rio de Janeiro: Brasport, 2009. 3 p.
- [3] TECHTARGET. **Xaas (anything as a service)**. Disponível em: <<http://searchcloudcomputing.techtarget.com/definition/xaas-anything-as-a-service>>. Acesso em: 14 abr. 2017.
- [4] AMAZON WEB SERVICES. **Tipos de computação em nuvem**. Disponível em: <<https://aws.amazon.com/pt/types-of-cloud-computing/>>. Acesso em: 14 abr. 2017.
- [5] SALESFORCE. **Benefits of SaaS**. Disponível em: <<https://www.salesforce.com/saas/benefits-of-saas/>>. Acesso em: 14 abr. 2017.
- [6] CUSTÓDIO, Anderson Fernando; JUNIOR, Edson A. Oliveira. **Um exemplo de aplicação multi-tenancy com hibernate shards**.
- [7] IMASTERS. **Entendendo o modelo multi-tenancy**. Disponível em: <<https://imasters.com.br/artigo/19067/cloud/entendendo-o-modelo-multi-tenancy/?trace=1519021197&source=single>>. Acesso em: 14 abr. 2017.
- [8] BEZEMER, Cor-paul; ZAIMAN, Andy. **Multi-tenant saas applications: maintenance dream or nightmare?**.

Assinaturas



Vinicius Cardoso Garcia

(orientador)



Bruno Resende Pinheiro

(proponente)

Recife, Pernambuco. Brasil

14 de Abril de 2017