



UFC



SystagramRest - Um serviço Restful para Gerenciar Posts de Fotos, implementado com o Google API Engine

Desenvolvimento de Software para Nuvem - 2019.1 - UFC/MDCC

Professores: Fernando Trinta e Paulo Rego

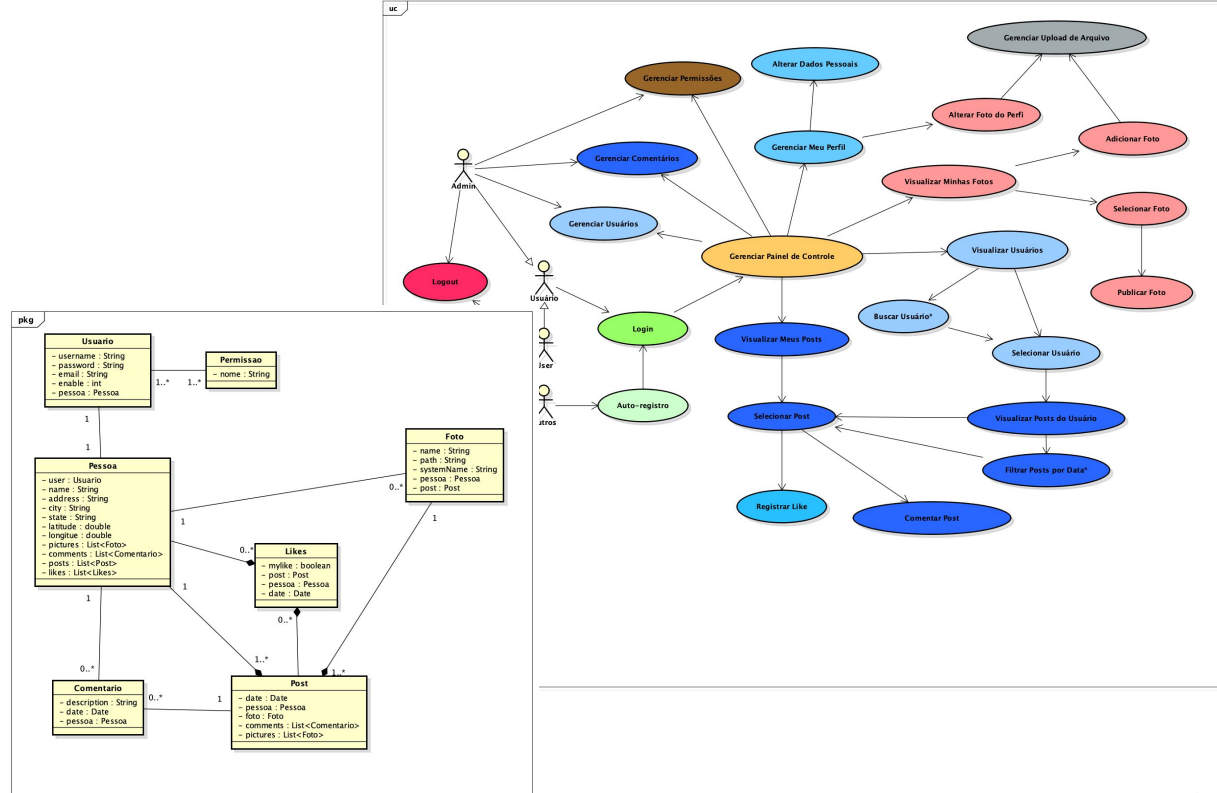
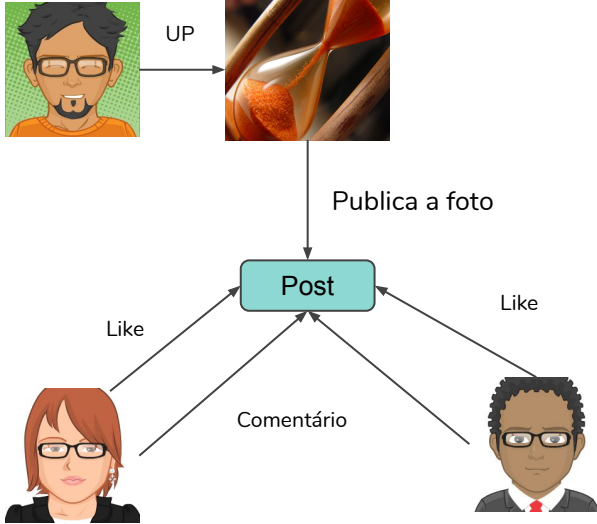
Alunos: Armando Soares e Juarez Meneses



Agenda

- ❑ O que é Systagram?
- ❑ Requisitos Funcionais
- ❑ Requisitos Não Funcionais
- ❑ Recursos publicados pelo serviço SystagramRest
- ❑ Arquitetura
- ❑ Protótipo da Aplicação Móvel
- ❑ Considerações Finais

O que é o Systagram?





Requisitos Funcionais - Histórias do Usuário

HU001 - O usuário se registra na Aplicação

HU002 - O usuário faz login na Aplicação

HU003 - O usuário faz logout da Aplicação

HU004 - O usuário gerencia seu perfil

HU005 - O usuário visualiza seu painel de controle

HU006 - O usuário visualiza outros usuários

HU007 - O usuário busca um usuário

HU008 - O usuário seleciona um usuário

HU009 - O usuário visualiza posts do usuário selecionado

HU010 - O usuário filtra posts de por data

HU011 - O usuário seleciona um post

HU012 - O usuário registra um like em post selecionado

HU013 - O usuário registra um comentário em post selecionado

HU014 - O usuário visualiza suas fotos

HU015 - O usuário faz upload de foto

HU016 - O usuário seleciona uma foto

HU017 - O usuário publica uma foto

HU018 - O usuário seleciona outro usuário

HU019 - O usuário visualiza perfil de outro usuário



Requisitos não funcionais

RNF001 - A aplicação deverá funcionar em um esquema de Servless hospedado funcionando no APP Engine em um esquema PaaS (Plataforma como Serviço) no Google Cloud usando o Google API Engine.

RNF002 - As informações dos usuários devem ser gravadas em uma instância de banco de dados relacional, criada pelo serviço Cloud SQL;

RNF003 - As fotos que serão enviadas devem ser armazenadas utilizando o serviço Cloud Storage;



Estrutura da Solução

Solução implementada sobre os Serviços do Google Cloud com o Google API Engine

- Aplicação Rest hospedada no APP Engine
 - Uso do Framework Restlet compatível com o Google API Engine.
 - Conjunto de serviços que serão consumidos pela aplicação móvel.
 - Componentes que se conectam com outros serviços do Google Cloud
- Cloud SQL
 - Serviço de Banco de Dados Relacional - MySql
- Cloud Storage
 - Serviço de Storage
- Clientes consomem os recursos rest da aplicação SystagramRest
- LoadBalance
 - Balanceamento de carga gerenciado automaticamente pelo APP Engine
- Auto Scaling
 - Esquema de elasticidade gerenciado pelo APP Engine.



Recursos publicados pelo Systagramrest

1. Inserir novo usuário.

```
curl -v --header "Content-Type: application/json" --request POST --data '{"username":"uteste2", "password":"uteste2"}'  
https://systagramgae.appspot.com/users/newUser
```

Dado um usuário novo de id=5 segue comando para **atualizar o usuário**:

```
curl -H 'Content-Type: application/json' -X PUT -d  
'{"username":"utesteUpdated","password":"utesteUpdated","enabled":true,"email":"utesteUpdated@gmail.com"}'  
https://systagramgae.appspot.com/users/5
```

Dado uma pessoa de id=2 segue comando para **atualizar a pessoa**:

```
curl -H 'Content-Type: application/json' -X PUT -d '{"name":"Pessoa Teste Updated","latitude":0.0,"longitude":0.0, "address":"Rua Teste Pessoa  
Updated", "city":"Teresina", "state":"PI", "cep":""}' https://systagramgae.appspot.com/person/2
```

2. Novo usuário se loga.

```
curl https://systagramgae.appspot.com/users/utesteUpdated@gmail.com/utesteUpdated
```

a. Usuário lista os usuários do sistema.

```
curl https://systagramgae.appspot.com/users
```



Recursos publicados pelo Systagramrest

3. Usuário de pessoa com id=2 insere fotos

```
curl -v -F 'img_mypicture=@/Users/armandosoaressousa/git/tsd/systagramgae/dragon-fly-1391358.jpg' https://systagramgae.appspot.com/upload/selected/picture/person/2
```

4. Listas as figuras do usuário.

```
curl https://systagramgae.appspot.com/upload/person/2/picture
```

4.1. Usuário publica a foto 32 e vira um novo post.

```
curl -v https://systagramgae.appspot.com/person/2/picture/32/post
```

4.2. Usuário publica a foto 33 e vira um novo post.

```
curl -v https://systagramgae.appspot.com/person/2/picture/33/post
```

5. Usuário lista seus posts.

```
curl -v https://systagramgae.appspot.com/person/2/post/listPosts
```

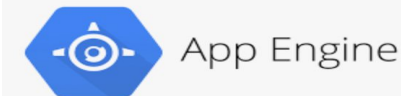
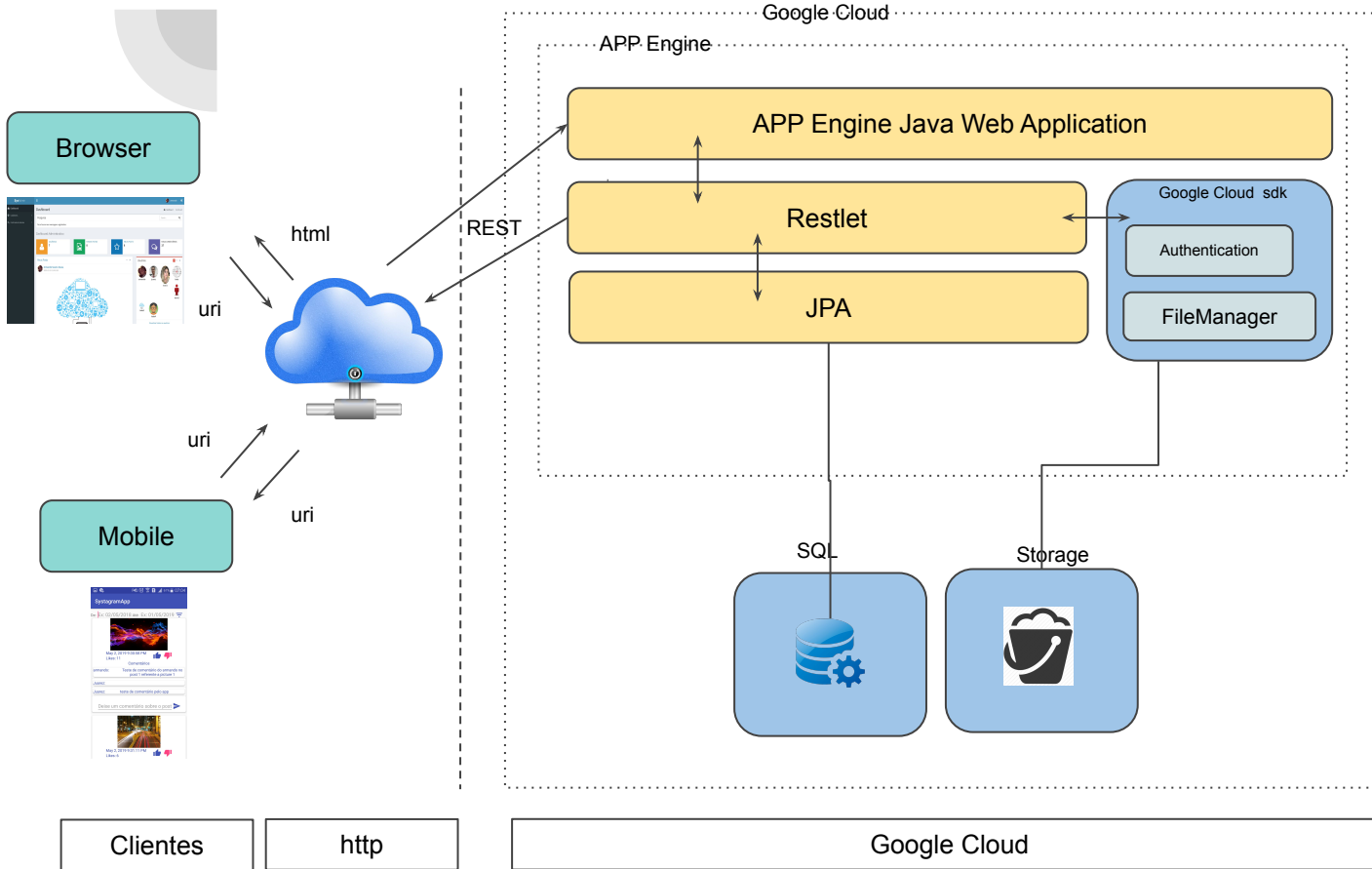
6. Usuário comenta o post 2.

```
curl -v --header "Content-Type: application/json" --request POST --data '{"description":"Teste de comentário no post 2 referente a picture 32"}' https://systagramgae.appspot.com/person/2/post/2/comment
```

7. Usuário dá um like no post 2.

```
curl -v --header "Content-Type: application/json" --request POST --data '{"description":"Teste de like no post 2", "mylike":"true"}' https://systagramgae.appspot.com/person/2/post/2/like
```


Arquitectura



Framework Restlet

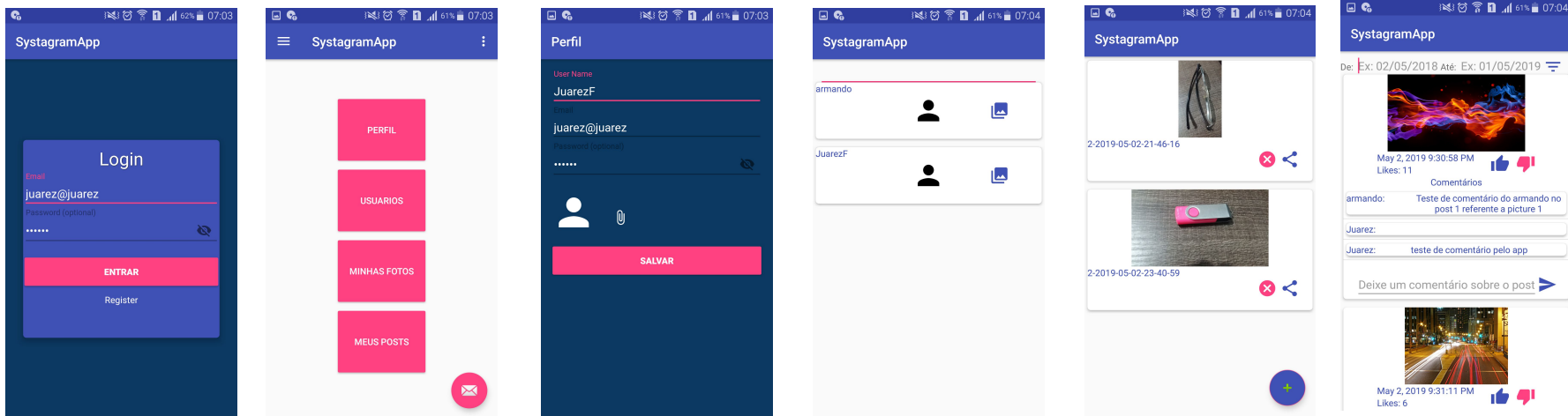


Tecnologias

Protótipo da Aplicação Móvel

Serviço Rest Systagram em produção:

<https://systagramgae.appspot.com>





Considerações Finais

- Criar aplicações para a nuvem não é trivial.
- Para criar aplicações na nuvem, no esquema PaaS, é preciso um conhecimento profundo da API disponibilizada pelo fornecedor.
 - Foi necessário trabalhar com um framework específico para o Google API Engine e usar o Google Cloud SDK para se comunicar com os outros serviços como o SQL e Storage.



Considerações Finais

- Foi identificada uma incompatibilidade com os frameworks mais novos para aplicações java como por exemplo, o Spring Boot e Jersey - que são muito usados na comunidade para aplicações web e serviços Rest - não funcionam muito bem com o Google API Engine. Com isso, foi necessário adotar o framework Restlet que é compatível com o Google API Engine.



Referências

Repositório de código do SystagramGAE App Web

<https://github.com/topicos-sistemas-distribuidos/systagramgae>



Dúvidas?





Contato



Armando Soares Sousa

armando@ufpi.edu.br

Juarez L. Meneses Filho

juarezmeneses@great.ufc.br

<http://www.mdcc.ufc.br>

Mestrado e Doutorado DC/UFC