

# Item 1. Estudo de caso

## Das Características Gerais 1.1

O sistema é um app mobile que guarda na nuvem dados de proprietários e de seus gatos. As perguntas e repostas são parametrizadas da seguinte forma:

Uma pergunta pode ser: Sim ou Não, múltipla escolha ou campo livre, caso seja uma pergunta de campo livre, a resposta pode ser estritamente um inteiro, uma string ou um campo ponto flutuante, a tela deve se comportar de forma apropriada dependendo do tipo de pergunta e dos parâmetros de resposta, por exemplo caso uma pergunta seja o peso do animal e o campo *tipo* 'textfree' o campo *possiveis* 'double' a tela deve ter uma caixa de texto que só permita um campo flutuante, assim como caso o campo *tipo* seja multiply uma tela de múltipla seleção com o conteúdo do array do campo *possiveis* para as possíveis seleções. A escolha do usuário ou resposta de campo aberto deve ser salva no campo *resposta* referente a resposta selecionada. Posterior a isso todo o objeto json referente ao usuário isto é o objeto do usuário, a lista de gatos e a lista de perguntas de cada gato com as respostas deve ser enviado para o servidor em seguida uma cópia deve ser mantida na estrutura chave valor do phonegap.

As perguntas estão separadas em categorias caso haja uma pergunta na lista de perguntas de algum gato sem resposta a categoria correspondente deve ser sinalizada com o número de perguntas sem respostas como mostra a figura 9 do anexo I.

## Das Características Técnicas – Requisitos não funcionais 1.2

O sistema é uma aplicação mobile que utiliza cordova phonegap com HTML5, CSS3 e JQuery mobile. A identidade visual e o protótipo com os dados fixos já estão criados e funcionando em uma single page no index.html com as "telas" separadas por *tags* sections (o formato preferencialmente deve ser mantido single page como recomenda o jquery mobile). O botão voltar do dispositivo deve obedecer ao comportamento de navegação entre as *tags* sections idênticos aos observados das pilhas de activits na navegação dos dispositivos android. Deverá ser observado tabulações e posicionamentos nas telas guardando posicionamentos e proporções obedecendo as imagens da documentação da identidade visual anexo I.

Todas as fronteiras de interfaces do front-end serão feitas através da função js *requisicao* que encapsula chamadas ajax entretanto, como os serviços ainda não estão prontos as chamadas retornarão objetos **mokados** previamente fornecidos através de funções jquery. As chamadas ajax trocarão objetos Json através de Restful. As estruturas dos objetos Json estão anexos no Anexo II os diagramas UML se encontram no anexoIII, as descrições dos casos de uso estão no item 2 desse documento.

Os dados retornados pelas URLs */login* ou */id* que irão conter todos os dados dos usuários a serem renderizados deverão ter uma cópia do objeto Json que foi recebido na estrutura chave valor do Phonegap assim como seu login, senha e id, para que seu estado senha mantido logado e os dados renderizados offline.

Posteriormente ao desenvolvimento utilizando os mokes será feito realizado a integração com os micro serviços que a esta época já deverão estar prontos.

O domínio da URL deverá ser parametrizada de modo que mudando em um único lugar sirva para toda a aplicação.

**OBS:** POSSÍVEIS CASOS DE TESTES NÃO MOKADOS MAS QUE FAÇAM PARTE DESSA MESMA ESTRUTURA DEVERÃO SER OBSERVADOS

**OBS2:** Não é possível mokar código de requisição restiful na função *requisicao ()* entretanto os fluxos alternativos devem ser previstos

**OBS3:** O sistema tem que manter seu estado off-line usando a estrutura chave valor do phonegap observar descrições de caso de uso

**OBS4:** O programador precisa pensar em novas estruturas mokadas caso for necessário, os mokes são apenas sugestões de como o desenvolvimento deve ser conduzido.

## Item 2. Descrição de Caso de uso

Caso de uso	1
URL	/login
method	get

### Fluxo Básico

É enviado objeto contendo email e senha figura 3 anexo I

É retornado código 200 e a mesma estrutura de objeto preenchida com o id preenchido

Então o objeto é salvo na estrutura chave valor do phonegape com o id, email e senha

Seguir o caso de uso número 4

### Fluxo alternativo 1

É enviado objeto contendo email e senha

É retornado o erro 403

Exibe mensagem de usuário ou senha inválidos

### Fluxo alternativo 2

É enviado objeto contendo email e senha

É retornado o erro 404

Exibe mensagem de servidor indisponível

### Fluxo alternativo 3

É enviado objeto contendo email e senha

É retornado o erro 500

Exibe mensagem de servidor indisponível

Caso de uso	2
URL	/new
method	post

Fluxo básico

As críticas mencionadas na figura 5 anexo I são realizadas

É enviado objeto usuário contendo email e senha figura 4 anexo 1 é enviado

É retornado código 201 e a mesma estrutura de objeto preenchida com o id preenchido é retornado

Então o objeto é salvo na estrutura chave valor do phonegape com o id, email e senha

Exibir mensagem de sucesso e tela de login aparece

Fluxo alternativo 3

É retornado o erro 403

Exibe mensagem de usuário já cadastrado

Fluxo alternativo 3

É enviado objeto contendo email e senha

É retornado o erro 404

Exibe mensagem de servidor indisponível

Fluxo alternativo 3

É enviado objeto contendo email e senha

É retornado o erro 500

Exibe mensagem de servidor indisponível problemas no servidor

Caso de uso	3
URL	/forgetpass/email
method	get

Fluxo básico

É retornado código 200

Sistema retorna para tela de login e exibe mensagem de que foi enviada uma nova senha para o email do usuário

Fluxo alternativo 1

É retornado o erro 403

Exibe mensagem de email não existe na base de dados

#### Fluxo alternativo 2

É retornado o erro 404

Exibe mensagem de servidor indisponível

#### Fluxo alternativo 3

É retornado o erro 500

Exibe mensagem de servidor indisponível problemas no servidor

Caso de uso	4
URL	/id
method	get

#### Fluxo Básico

É enviado objeto contendo email, senha e id

É retornado código 200 e os dados dos usuários e dados básicos dos gatos

Esse objeto deve ser guardado localmente na estrutura local do dispositivo

A lista de gatos com as fotos é exibida

#### Fluxo alternativo 2

É retornado o erro 404

Exibe mensagem de servidor indisponível

#### Fluxo alternativo 3

É retornado o erro 500

Exibe mensagem de servidor indisponível problemas no servidor

Caso de uso	5
URL	/id/cat <id do usuário> /<id do gato>
method	get

#### Fluxo Básico

Quando selecionado um determinado gato

É retornado código 200 e os dados dos usuários e todos os dados dele assim como todas suas perguntas tanto as já respondidas quanto as sem respostas

Esse objeto deve ser guardado localmente na estrutura local do dispositivo

A lista de categorias devem ser renderizada

As categorias com perguntas a serem respondidas devem ser sinalizadas como mostra a figura 9 no anexo I

Quando uma categoria for selecionada a lista de perguntas daquela categoria deve ser exibida

Quando uma pergunta for selecionada ele deve se comportar de acordo com o seu tipo

Fluxo alternativo 2

É retornado o erro 404

Exibe mensagem de servidor indisponível

Fluxo alternativo 3

É retornado o erro 500

Exibe mensagem de servidor indisponível problemas no servidor

Caso de uso	5
URL	/id/cat <id do usuário> /<id do gato>
method	put

Fluxo Básico

Quando uma pergunta for respondida

Todo o objeto deve ser armazenado localmente usando a estrutura do phonegap

O gato e a pergunta com a resposta deve ser enviado

Um código de sucesso 200 será recebido

Uma mensagem de sucesso deve ser exibida para o usuário

Fluxo alternativo 2

É retornado o erro 404

Exibe mensagem de servidor indisponível

Fluxo alternativo 3

É retornado o erro 500

Exibe mensagem de servidor indisponível problemas no servidor

Caso de uso	6
URL	/id/cat <id do usuário> /<id do gato>
method	post

## Fluxo Básico

Na lista de gatos quando clicado no símbolo mais deve ser aberta a tela descrita na figura 10 anexo I

**Atenção: o campo nome não está na imagem da figura 10 e deve ser adicionado, o campo foto deverá ser escolhido tanto através da galeria quanto através de uma nova foto tirada da câmera.**

O usuário poderá deverá preencher os campos referentes as informações básicas dos gatos.

Os campos nome, foto, sexo, cor, pelagem e procedência são obrigatórios

Os campos sexo, cor, pelagem e procedência são multivalorados e a lista de valores se encontra no anexo IV, esses campos devem ser exibidos em listas de multi seleção

Depois de preenchido o objeto json gato.

Um código de sucesso 201 será recebido

Uma mensagem de sucesso deve ser exibida para o usuário

## Fluxo alternativo 2

É retornado o erro 404

Exibe mensagem de servidor indisponível

## Fluxo alternativo 3

É retornado o erro 500

Exibe mensagem de servidor indisponível problemas no servidor

# ANEXO I

## IDENTIDADE VISUAL



figura1

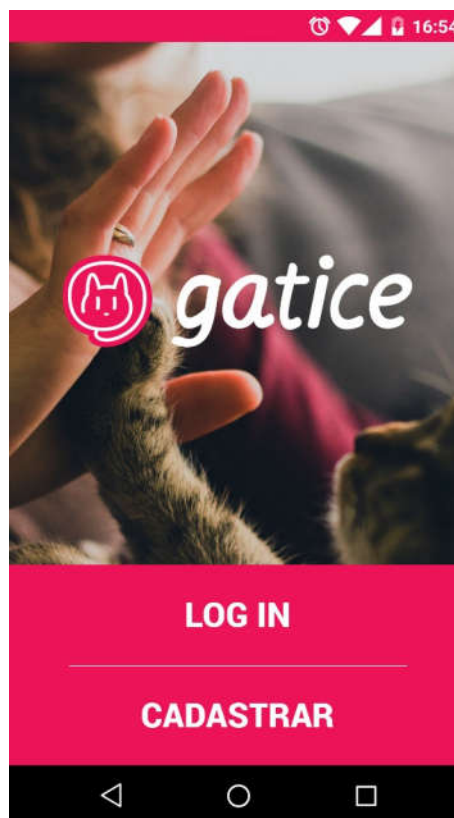


figura2

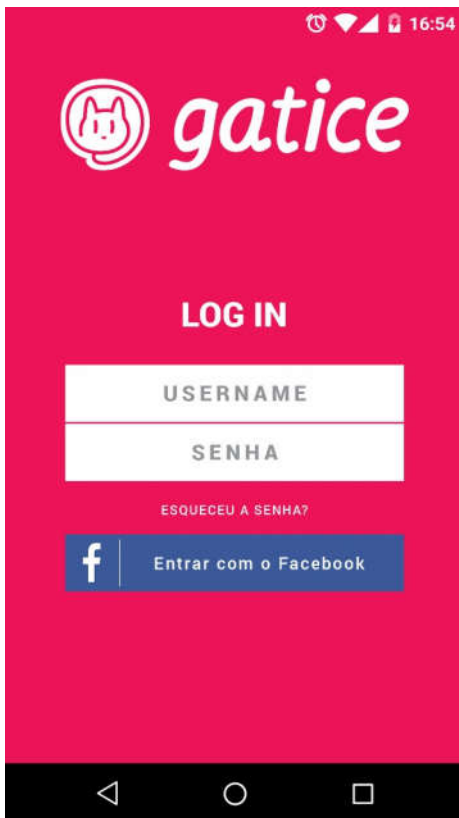


Figura3

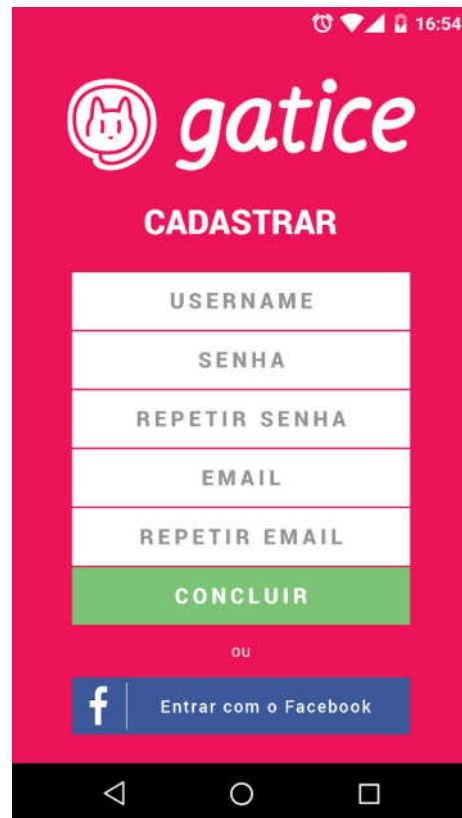


figura4



**gatice**

### CADASTRAR

GATOGATO123	USERNAME INDISPONÍVEL	✗
****	MÍNIMO DE 6 CARACTERES	✗
*****	SENHAS DIFERENTES	✗
GATOROSA_93@GMAIL	EMAIL JÁ ESTÁ SENDENDO USADO	✗
PATOBranco_93@GAT	EMAILS DIFERENTES	✗

**CONCLUIR**

OU

Entrar com o Facebook

Figura5

**gatice**

### CADASTRAR

GATOGATO123		✓
*****		✓
*****		✓
GATOROSA_93@GMAIL.COM		✓
GATOROSA_93@GMAIL.COM		✓

**CONCLUIR**

OU

Entrar com o Facebook

figura6

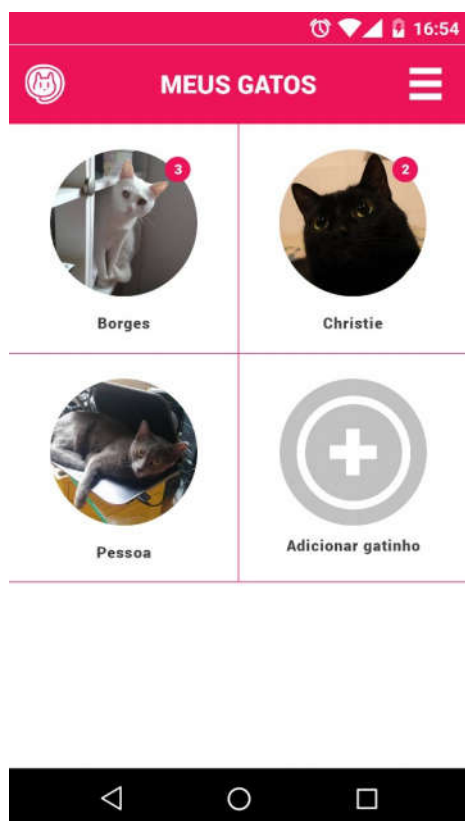


Figura7

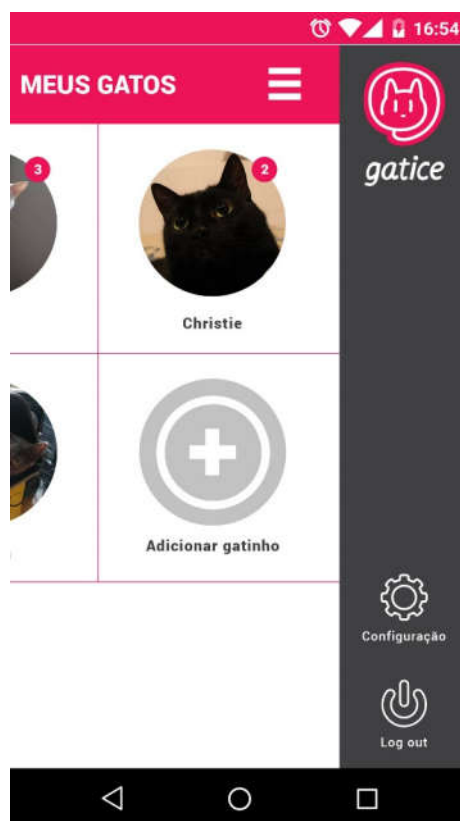


figura8

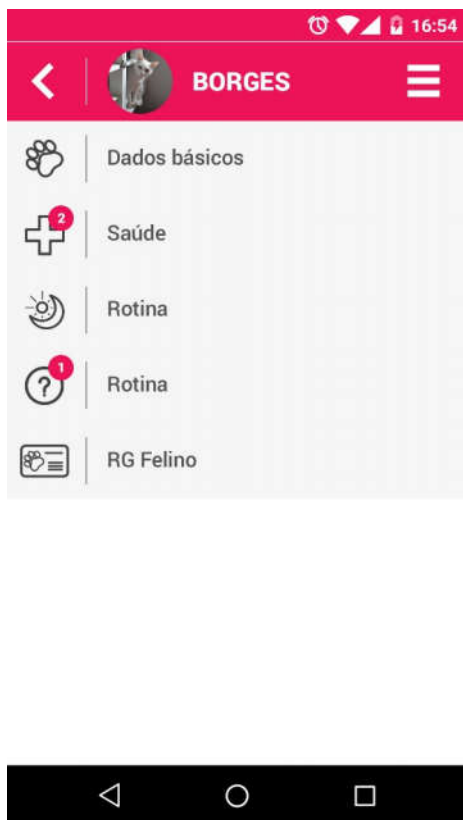


Figura9



figura10

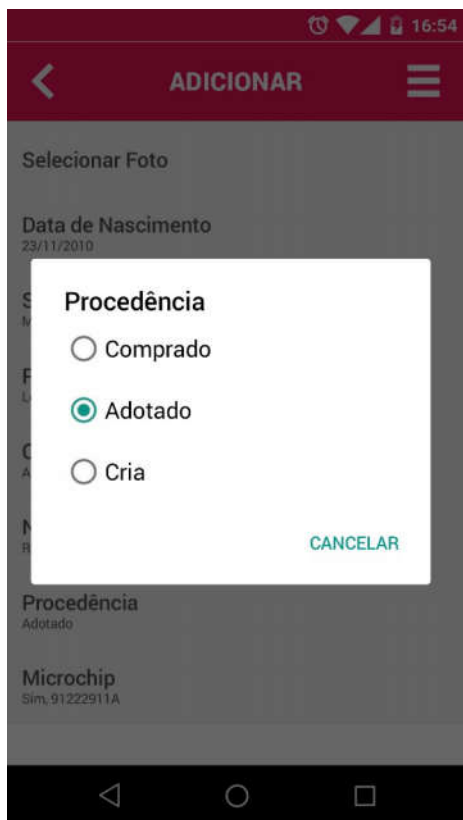


Figura11

## ANEXO II

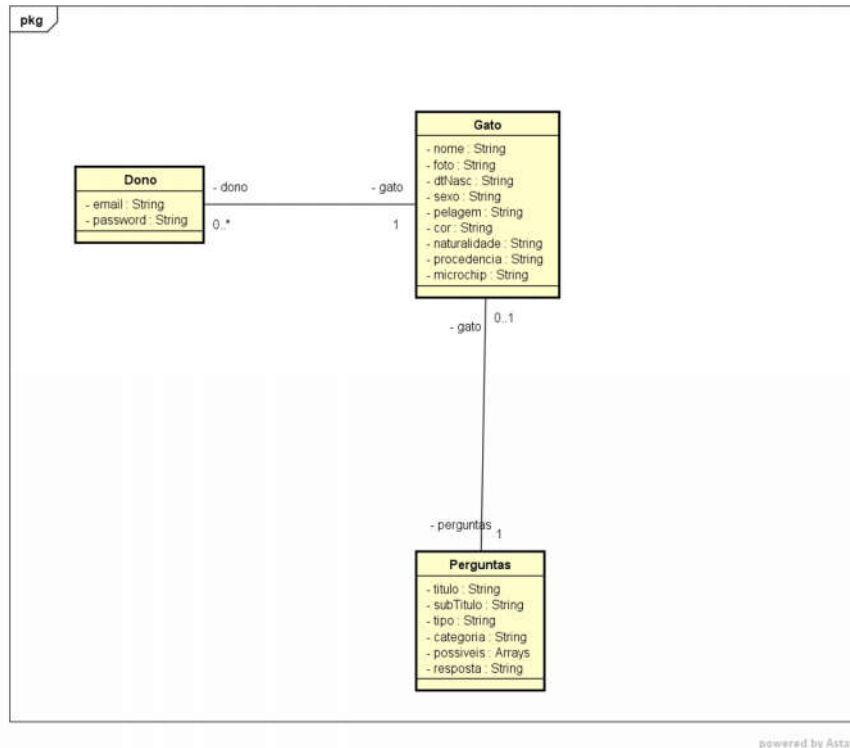
```
{
  id:",
  email:",
  senha:",
  gatos:[],
}
```

gato

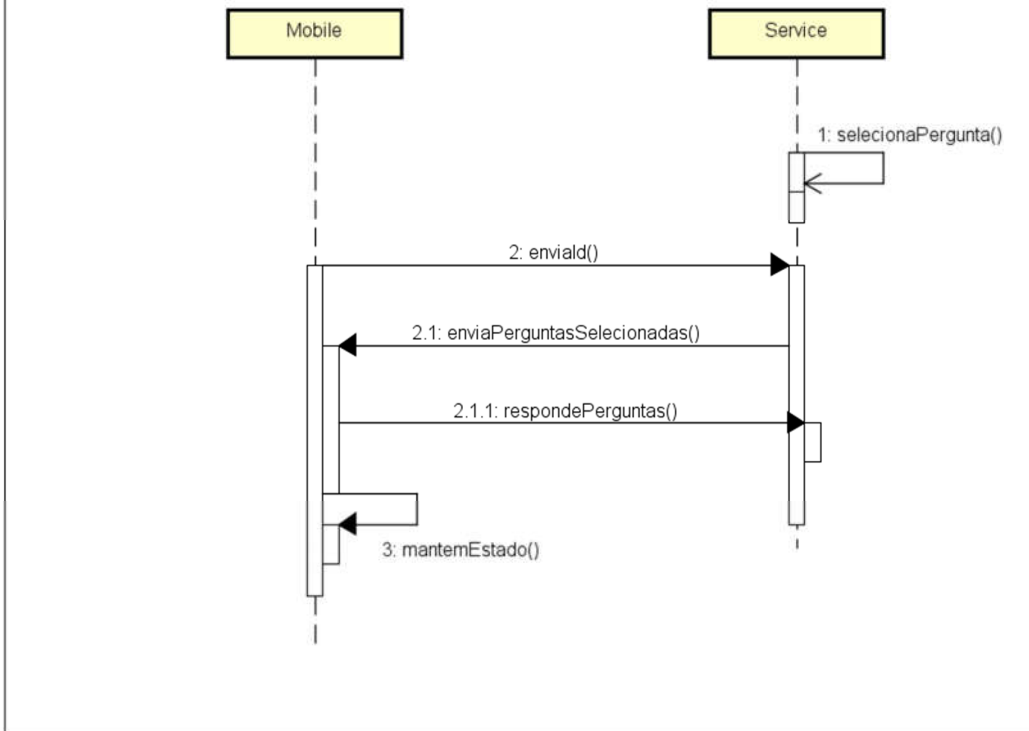
```
{
  id:",
  nome:",
  foto:",
```

```
    dtNasc:",
    sexo:",
    pelagem:",
    cor:",
    naturalidade:",
    procedencia:",
    microchip:",
    perguntas:[],
  }
  pergunta
  {
    id:",
    titulo:",
    valor:",
    subTitulo:",
    tipo:",
    categoria:",
    posiveis:[],
    resposta:",
  }
}
```

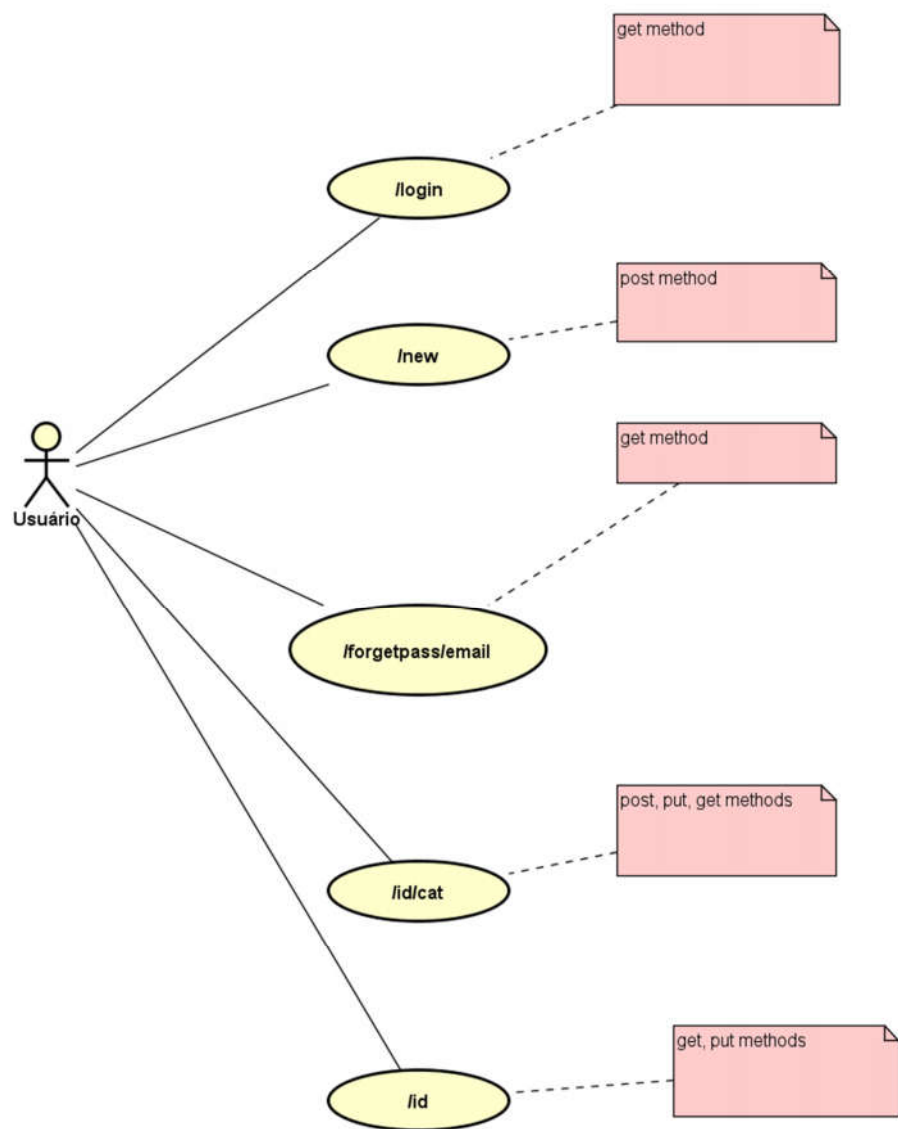
# ANEXO III



sd Sequence Diagram0



uc





# ANEXO IV

## Campos Multivalorados

### COR

- Branco
- Preto
- Cinza
- Amarelo
- Escaminha
- Tricolor
- Frajola
- Branco/amarelo
- Tigrado
- Tigrado/branco
- Preto/branco
- Cinza/branco
- Outra

### Sexo

- Masculino
- Feminino

### Pelagem

- Curto
- Longo
- Sem Pelo

### Procedência

- Adoção
- Compra
- Cria