

# Estudo da modelagem Pet Clinic

Conrado Boeira e Luiz Henrique Mosmann

21 de Agosto 2019

## 1 Introdução

Nesse trabalho, estudaremos o funcionamento da aplicação *Petclinic*, exibindo dois modelos e uma *checklist* para os modelos mencionados anteriormente.

## 2 Modelos

O modelo abaixo é um modelo de classes para a aplicação. Há 4 containers principais que são representados no modelo. Estes são: Model, que inclui a classes com Person que modelam as características básicas para pessoas; Vet, que modela especialmente um veterinário na clinica; Owner, que traz uma especificação de pessoa que é dona de um pet e traz uma classe para modelos o animal de estimação; e Visit, que tem uma classe que modelo a ação de ter uma visita ao veterinário marcada.

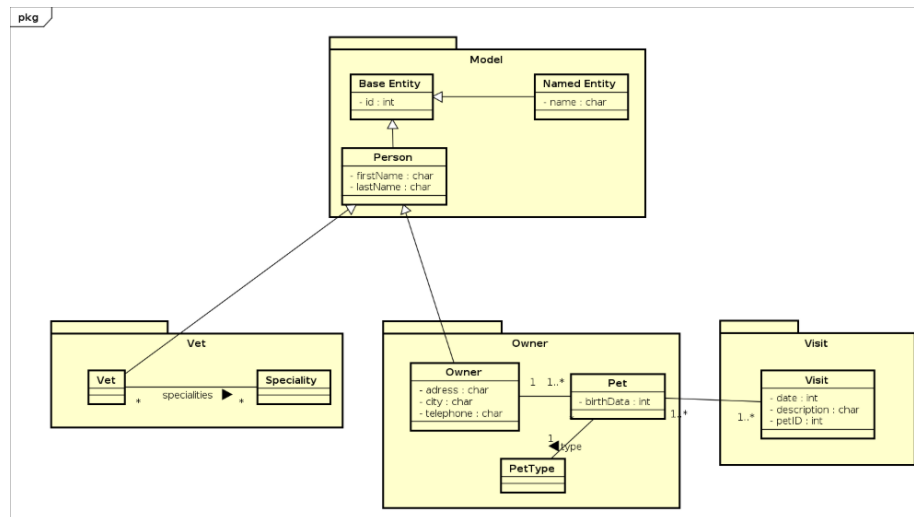
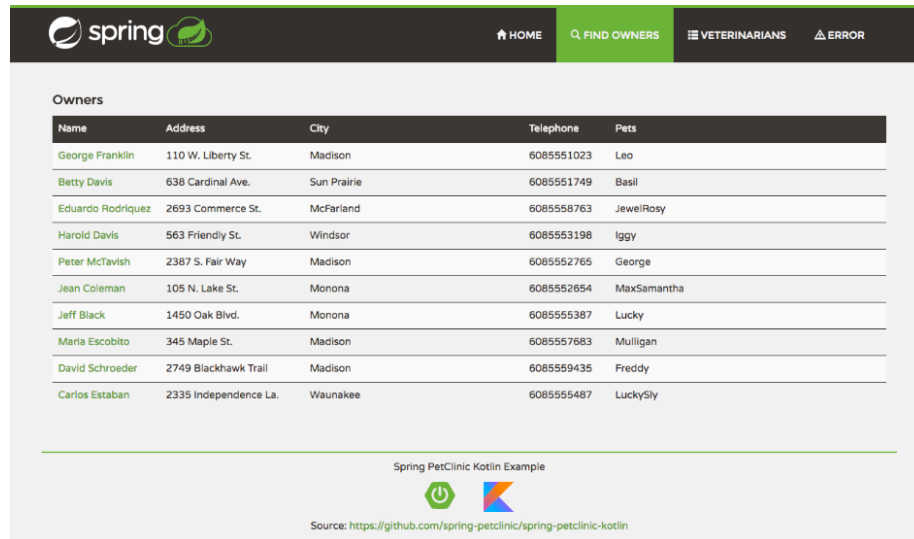


Figure 1: Modelo de classes

O modelo abaixo trata-se de um protótipo de interface da aplicação *Pet-Clinic*. No protótipo há uma tabela com as principais informações dos clientes, divididas em colunas, sendo elas: Nome, Endereço, Cidade, Telefone e os *Pets* do Cliente. Pode-se citar algumas vantagens do modelo proposto: antes mesmo do desenvolvimento, já se tem algo material para apresentar para o cliente e/ou time; processo constante de *feedback*; redução de riscos; redução de custos.



Name	Address	City	Telephone	Pets
George Franklin	110 W. Liberty St.	Madison	6085551023	Leo
Betty Davis	638 Cardinal Ave.	Sun Prairie	6085551749	Basil
Eduardo Rodriguez	2693 Commerce St.	McFarland	6085558763	JewelRosy
Harold Davis	563 Friendly St.	Windsor	6085553198	Iggy
Peter McTavish	2387 S. Fair Way	Madison	6085552765	George
Jean Coleman	105 N. Lake St.	Monona	6085552654	MaxSamantha
Jeff Black	1450 Oak Blvd.	Monona	6085555387	Lucky
Maria Escobito	345 Maple St.	Madison	6085557683	Mulligan
David Schroeder	2749 Blackhawk Trail	Madison	6085559435	Freddy
Carlos Estaban	2335 Independence La.	Waunakee	6085555487	LuckySly

Spring PetClinic Kotlin Example

Source: <https://github.com/spring-petclinic/spring-petclinic-kotlin>

Figure 2: Protótipo de interface

### 3 Checklist

A checklist para checar os modelos segue:

- Em modelos de classes, todas classes precisam ter nomes
- Todo modelo precisa ter claro o que está sendo modelado
- Todo modelo precisa disponibilizar uma maneira de altera-lo
- Em modelos de classes, todas as relações entre classes precisam esta explícitas
- Todo modelo precisa ter a data em que foi feito

### 4 Referências

[1] Kotlin version of the Spring PetClinic Sample Application. Acessado: 21-08-2019. URL: <https://github.com/spring-petclinic/spring-petclinic-kotlin>

- [2] Spring PetClinic Sample Application. Acessado: 21-08-2019. URL: <https://github.com/spring-projects/spring-petclinic>
- [3] Análise de modelos de PetClinic. Acessado: 21-08-2019.  
URL: [https://github.com/conradoboeira/ESOM-Analise\\_modelos/tree/master](https://github.com/conradoboeira/ESOM-Analise_modelos/tree/master)