petclinic.model

<<Entity>>
BaseEntity

- id: Integer

+ <init>(): void
+ getId(): Integer
+ setId(Integer): void
+ isNew(): boolean

<<Entity>>
Person

- firstName: String
- lastName: String

+ <init>(): void
+ getFirstName(): String
+ setFirstName(String): void
+ getLastName(): String
+ setLastName(String): void

<<Entity>>
NamedEntity

- name: String

+ <init>(): void
+ getName(): String
+ setName(String): void
+ toString(): String

petclinic.vet

<<Infrastructure>>
Vets

- vets: List

+ <init>(): void
+ getVetList(): List

<<Infrastructure>>
VetController

- vets: VetRepository

+ <init>(VetRepository): void
+ showVetList(Map): String
+ showResourcesVetList(): Vets

<<Application Service>>
PetClinicApplication

+ <init>(): void
+ main(String[]): void

<<Entity>>
Specialty

+ <init>(): void

<<Aggregate Root>>
Vet

- specialties: Set

+ <init>(): void
# getSpecialtiesInternal(): Set
# setSpecialtiesInternal(Set): void
+ getSpecialties(): List
+ getNrOfSpecialties(): int
+ addSpecialty(Specialty): void

<<Repository>>
VetRepository

+ findAll(): Collection

petclinic.owner

<<Aggregate Root>>
Owner

- address: String
- city: String
- telephone: String
- pets: Set

+ <init>(): void
+ getAddress(): String
+ setAddress(String): void
+ getCity(): String
+ setCity(String): void
+ getTelephone(): String
+ setTelephone(String): void
# getPetsInternal(): Set
# setPetsInternal(Set): void
+ getPets(): List
+ addPet(Pet): void
+ getPet(String): Pet
+ getPet(String, boolean): Pet
+ toString(): String

<<Entity>>
Pet

- birthDate: LocalDate
- type: PetType
- owner: Owner
- visits: Set

+ <init>(): void
+ setBirthDate(LocalDate): void
+ getBirthDate(): LocalDate
+ getType(): PetType
+ setType(PetType): void
+ getOwner(): Owner
# setOwner(Owner): void
# getVisitsInternal(): Set
# setVisitsInternal(Set): void
+ getVisits(): List
+ addVisit(Visit): void

<<Entity>>
PetType

+ <init>(): void

<<Service>>
PetValidator

- REQUIRED: String

+ <init>(): void
+ validate(Object, Errors): void
+ supports(Class): boolean

<<Repository>>
OwnerRepository

+ findByLastName(String): Collection
+ findById(Integer): Owner
+ save(Owner): void

<<Infrastructure>>
OwnerController

- VIEWS_OWNER_CREATE_OR_UPDATE_FORM: String
- owners: OwnerRepositoryng

+ <init>(OwnerRepository): void
+ setAllowedFields(WebDataBinder): void
+ initCreationForm(Map): String
+ processCreationForm(Owner, BindingResult): String
+ initFindForm(Map): String
+ processFindForm(Owner, BindingResult, Map): String
+ initUpdateOwnerForm(int, Model): String
+ processUpdateOwnerForm(Owner, BindingResult,int): String
+ showOwner(int): ModelAndView

<<Service>>
PetTypeFormatter

- pets: PetRepository

+ <init>(PetRepository): void
+ print(PetType, Locale): String
+ parse(String, Locale): PetType
+ print(Object, Locale): String
+ parse(String, Locale): Object

<<Infrastructure>>
VisitController

- visits: VisitRepository
- pets: PetRepository

+ <init>(VisitRepository ,PetRepository): void
+ setAllowedFields(WebDataBinder): void
+ loadPetWithVisit(int, Map): Visit
+ initNewVisitForm(int,Map): String
+ processNewVisitForm(Visit, BindingResult): String

<<Repository>>
PetRepository

+ findPetTypes(): List
+ findById(Integer): Pet
+ save(Pet): void

<<Infrastructure>>
PetController

- VIEWS_PETS_CREATE_OR_UPDATE_FORM: String
- pets: PetRepository
- owners: OwnerRepository

+ <init>(PetRepository, OwnerRepository): void
+ populatePetTypes(): Collection
+ findOwner(int): Owner
+ initOwnerBinder(WebDataBinder): void
+ initPetBinder(WebDataBinder): void
+ initCreationForm(Owner,ModelMap): String
+ processCreationForm(Owner, Pet, BindingResult, ModelMap): String
+ initUpdateForm(int,ModelMap): String
+ processUpdateForm(Pet, BindingResult, Owner, ModelMap): String

petclinic.visit

<<Domain Event>>
Visit

- date: LocalDate
- description: String
- petId: Integer

+ <init>(): void
+ getDate(): LocalDate
+ setDate(LocalDate): void
+ getDescription(): String
+ setDescription(String): void
+ getPetId(): Integer
+ setPetId(Integer): void

<<Repository>>
VisitRepository

+ save(Visit): void
+ findByPetId(Integer): List

petclinic.system

<<Infrastructure>>
WelcomeController

~ <init>(): void
+ welcome(): String

<<Infrastructure>>
CrashController

~ <init>(): void
+ triggerException(): String

<<Infrastructure>>
CacheConfiguration

~ <init>(): void
+ petclinicCacheConfigurationCustomizer(): JCacheManagerCustomizer
- cacheConfiguration(): Configuration
- lambda$petclinicCacheConfigurationCustomizer$0(CacheManager): void