Engenharia de Softwares Escaláveis

Design Patterns e Domain-Driven Design com Java

Agenda

Etapa 9:

- Assesment.
- Context Mapper.

Assesment

Competência 1 Desenvolver software aplicando design patterns

1. Qual é o papel dos Design Patterns na solução de problemas de software?
2. Qual é o papel das Fachadas na integração de contextos limitados?

Competência 2 Projetar softwares de forma estratégica, usando "bounded contexts", subdomínios e linguagem ubíqua

1. Como o Domain-Driven Design (DDD) auxilia na gestão da complexidade de projetos de software?
2. Explique a diferença entre design estratégico e tático.
3. Qual é a importância da Linguagem Ubíqua, mesmo em projetos que não usam DDD?
4. O que são contextos principais, de suporte e genéricos? Por que os contextos genéricos, em geral, são contratados externamente?

Competência 3 Projetar softwares de forma estratégica usando "context maps"

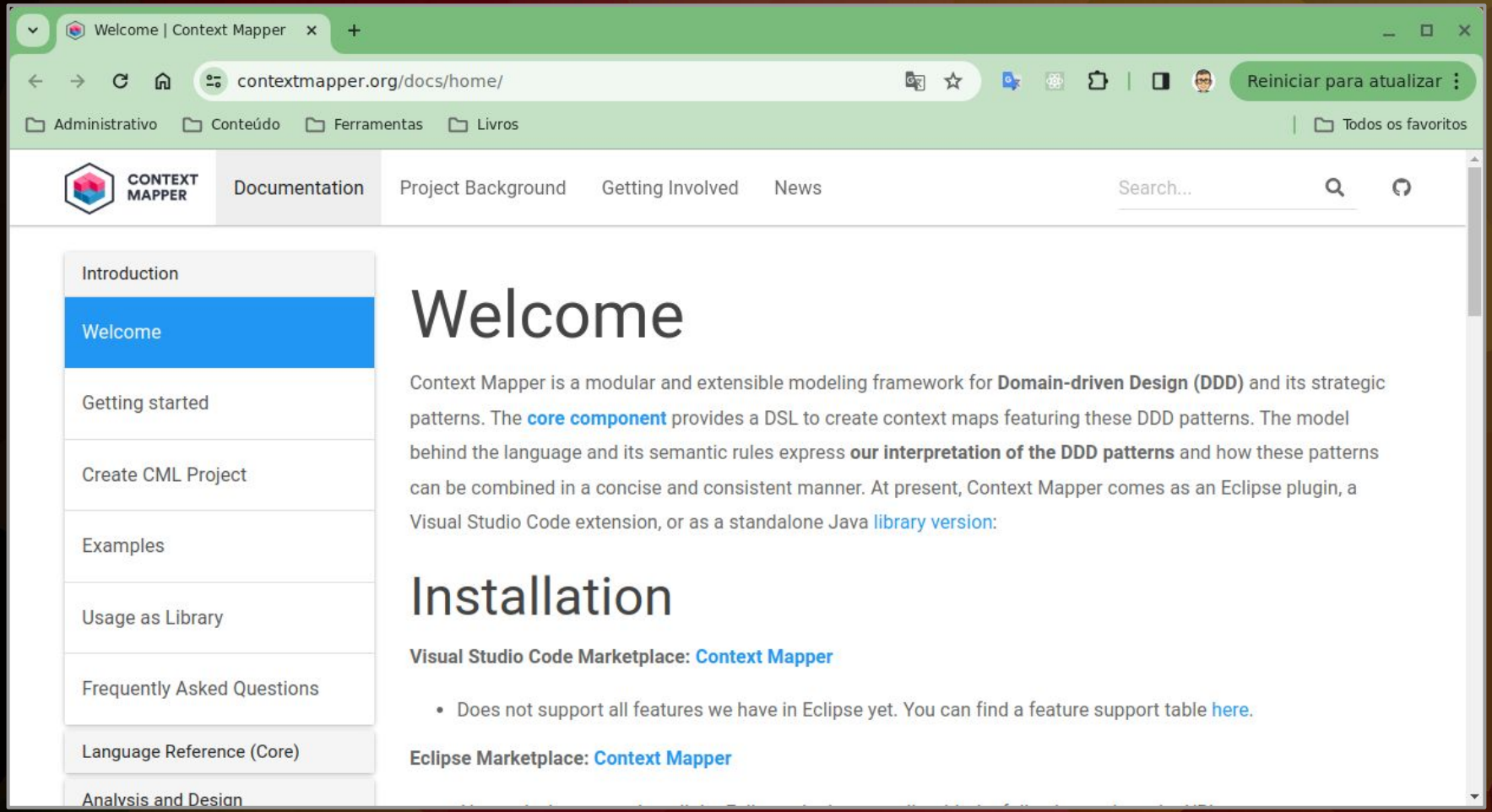
1. Desenhe o Mapa de Contexto do Projeto Pet Friends (use o modelo do TP3 com as melhorias que achar necessário) e classifique os contextos encontrados em “Principal”, “Genérico” e “Suporte” (neste cenário do AT o seu Bounded Context principal é o de Agendamento). Indique no mapa os tipos de relacionamentos.
2. Elabore uma lista com as estratégias a serem adotadas para as comunicações / integrações entre o contexto de agendamento e os contextos relacionados e aponte os tipos de comunicações entre contextos.

Competência 4

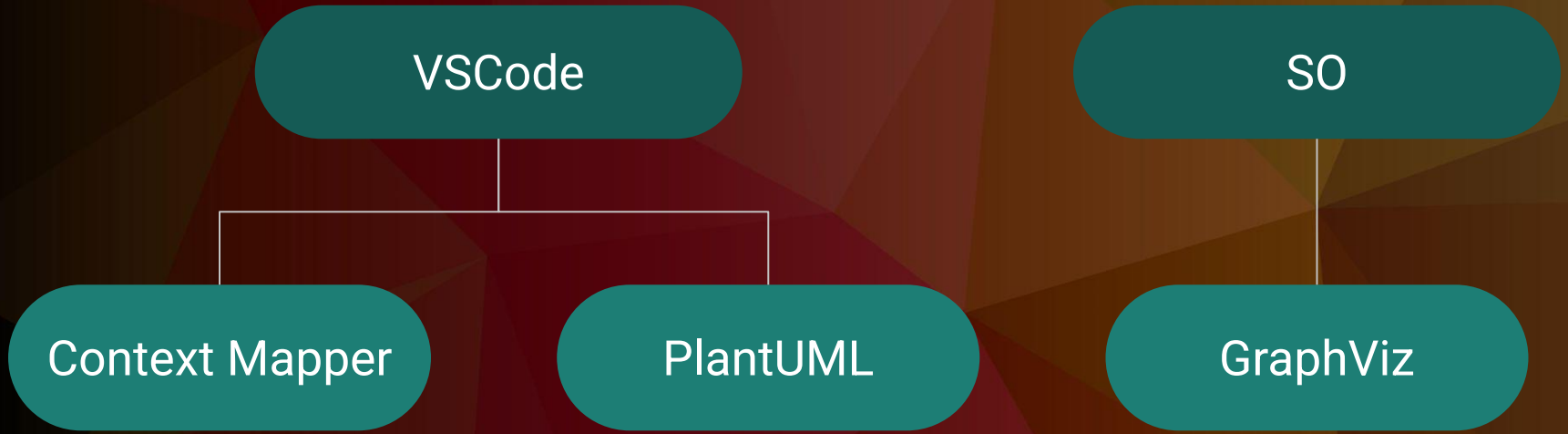
Projetar softwares usando "aggregates"

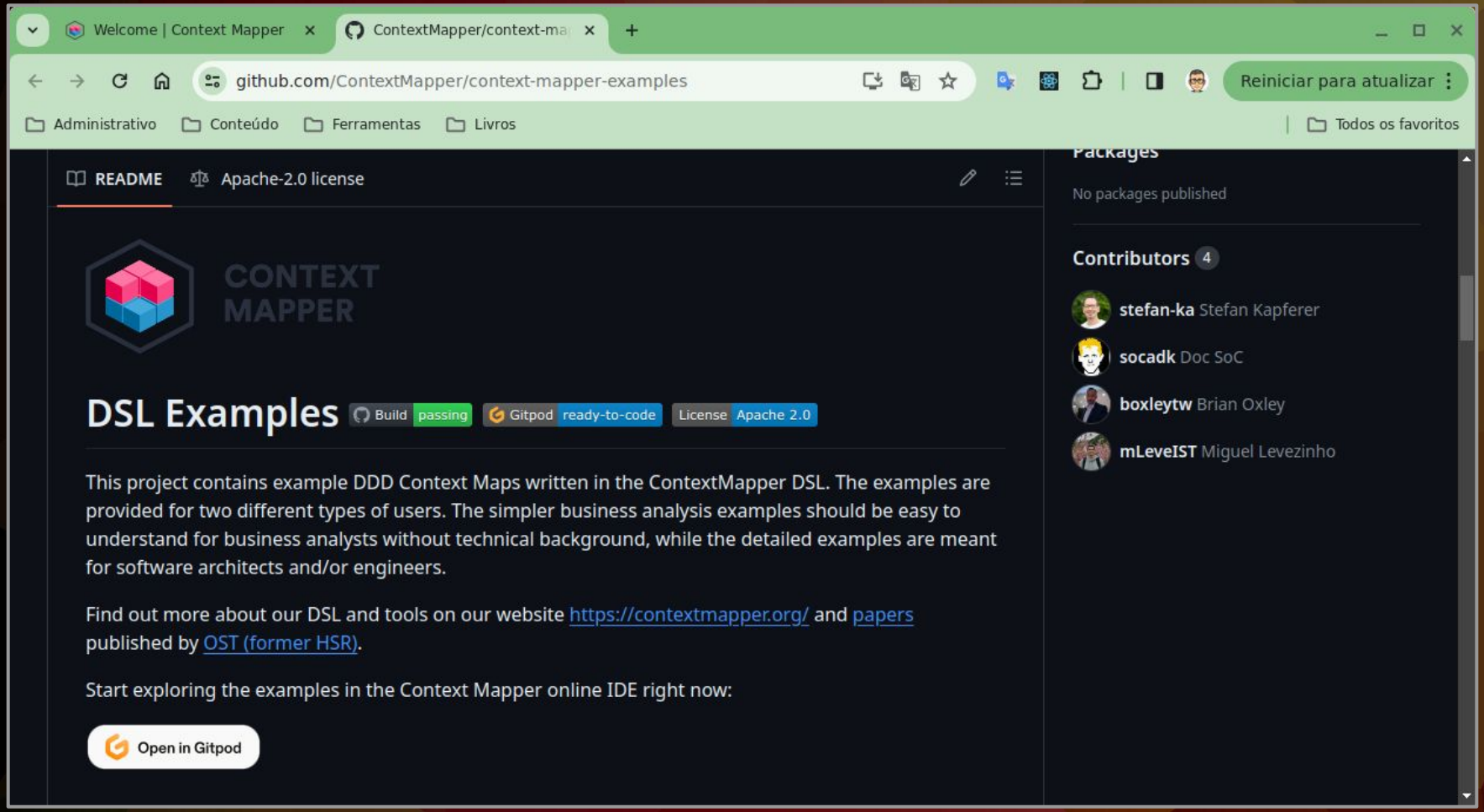
1. Explique a diferença entre Entidade e Objeto de Valor e dê um exemplo de cada usando o Pet Friends.
2. Modele os agregados que encontrar no contexto de Agendamento.

Context Mapper



https://contextmapper.org/docs/home/


VSCode
SO
Context Mapper
PlantUML
GraphViz



https://github.com/ContextMapper/context-mapper-examples

Insurance-Example-Stage-1.cml - context-mapper-examples-master - Visual Studio Code

Arquivo Editar Seleção Ver Acessar Executar Terminal Ajuda

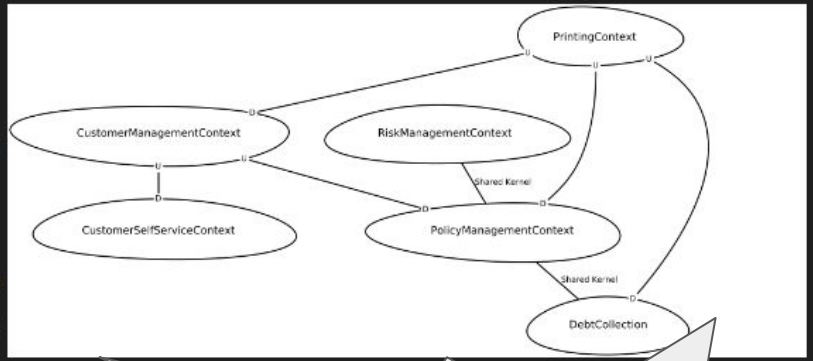
Insurance-Example-Stage-1.cml

bin > main > insurance-example > Insurance-Example-Stage-1.cml >

```
/* Example Context Map written with 'ContextMappe
ContextMap InsuranceContextMap {
    type = SYSTEM_LANDSCAPE
    state = TO_BE

    /* Add bounded contexts to this context map:
    contains CustomerManagementContext
    contains CustomerSelfServiceContext
    contains PrintingContext
    contains PolicyManagementContext
    contains RiskManagementContext
    contains DebtCollection

    /* Define the context relationships: */

    CustomerSelfServiceContext <- CustomerManagem

    CustomerManagementContext <- PrintingContext

    PrintingContext -> PolicyManagementContext

    RiskManagementContext <-> PolicyManagementCon

    PolicyManagementContext <- CustomerManagement

    DebtCollection <- PrintingContext

    PolicyManagementContext <-> DebtCollection
}

/* Bounded Context Definitions */
BoundedContext CustomerManagementContext implemen

BoundedContext CustomerSelfServiceContext impleme
```

Insurance-Example-Stage-1_ContextMap.png

src-gen > Insurance-Example-Stage-1_ContextMap.png



Botão direito do mouse.
Generate Graphical Context Map

PROBLEMAS 216 SAÍDA CONSOLE DE DEPURAÇÃO TERMINAL PORTAS

SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
SLF4J: Defaulting to no-operation (NOP) logger implementation

⊗ 0 ⚠ 154 ⓘ 62 0 Java: Ready Ln 7, Col 39 Tamanho da Tec ab: 4 UTF-8 LF Context Mapper DSL

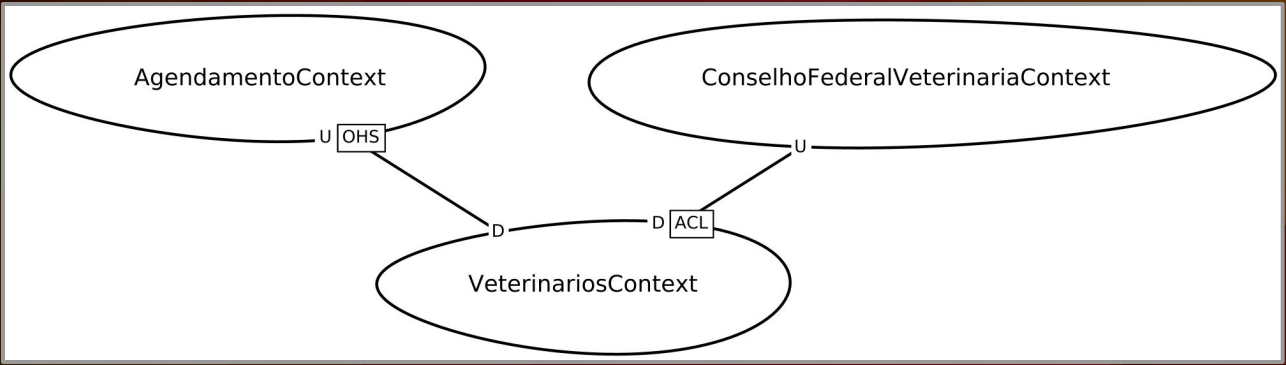
Pet Friends

1

PetFriends_1.cml

PetFriends_1.cml > PetFriends

```
1
2   ContextMap PetFriends {
3       contains AgendamentoContext
4       contains VeterinariosContext
5       contains ConselhoFederalVeterinariaContext
6
7       VeterinariosContext [D] <- [U, OHS] AgendamentoContext
8
9       VeterinariosContext [D, ACL] <- [U] ConselhoFederalVeterinariaContext
10  }
11  BoundedContext AgendamentoContext
12  BoundedContext VeterinariosContext
13  BoundedContext ConselhoFederalVeterinariaContext
```

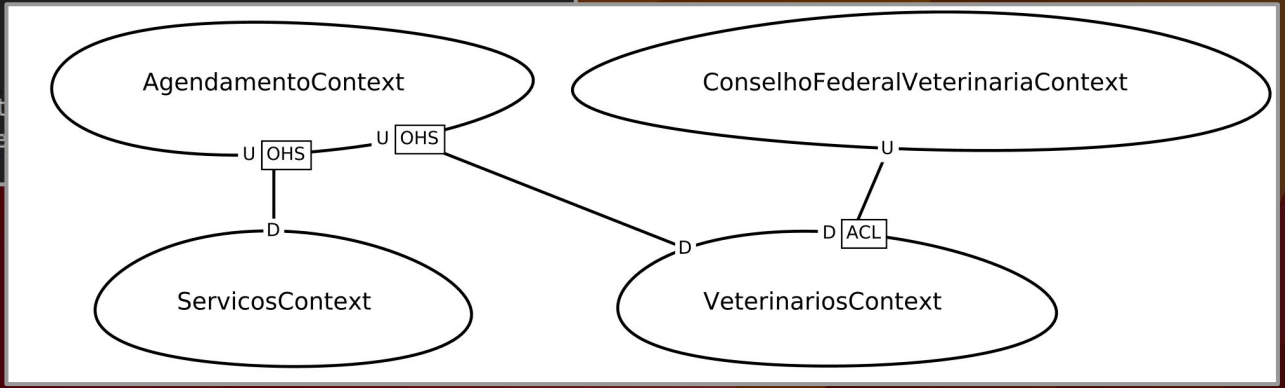


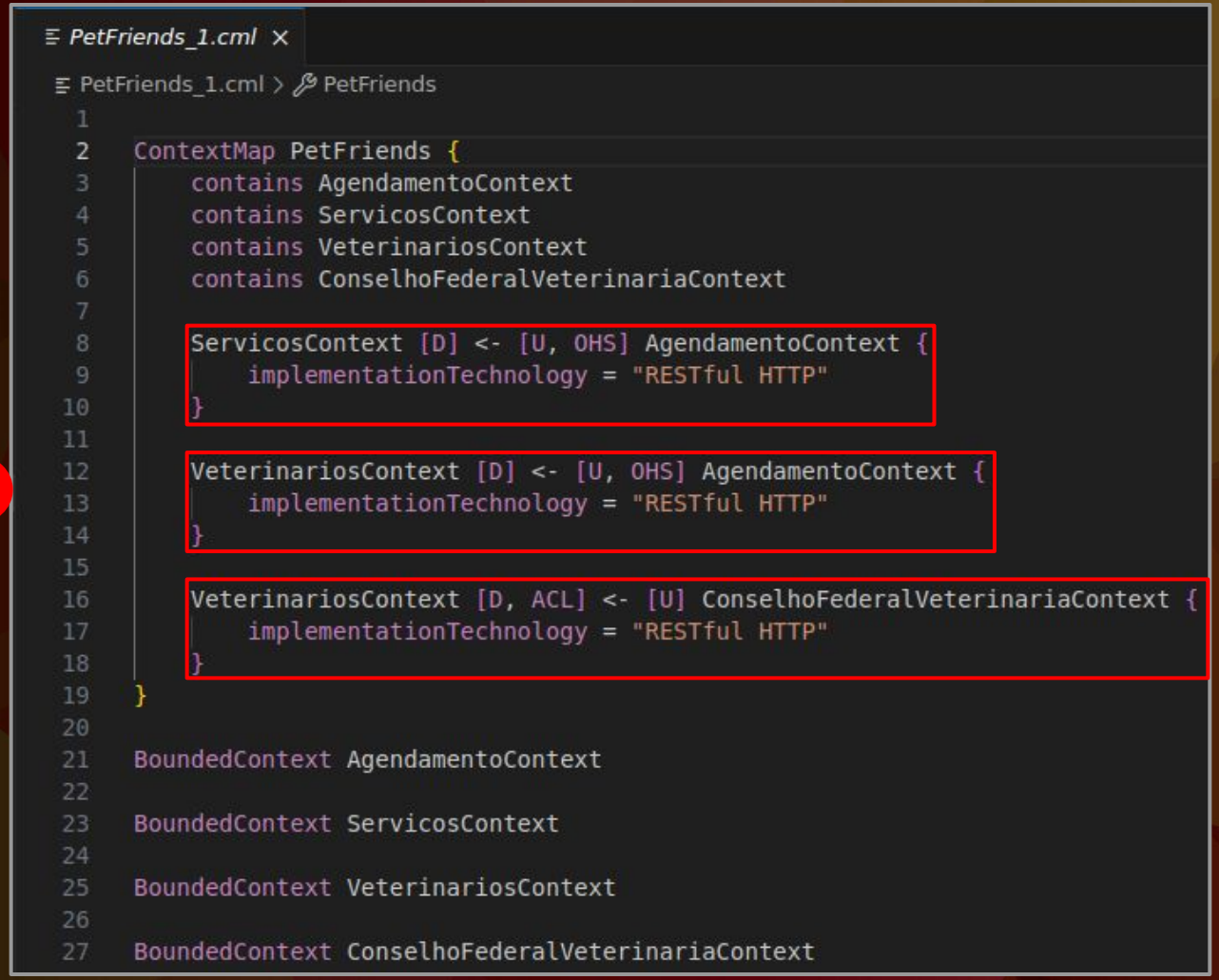
PetFriends_1.cml

PetFriends_1.cml > PetFriends

```
1
2   ContextMap PetFriends {
3       contains AgendamentoContext
4       contains ServicosContext
5       contains VeterinariosContext
6       contains ConselhoFederalVeterinariaContext
7
8       ServicosContext [D] <- [U, OHS] AgendamentoContext
9
10      VeterinariosContext [D] <- [U, OHS] AgendamentoContext
11
12      VeterinariosContext [D, ACL] <- [U] ConselhoFederalVeterinariaContext
13  }
14
15  BoundedContext AgendamentoContext
16  BoundedContext ServicosContext
17  BoundedContext VeterinariosContext
18  BoundedContext ConselhoFederalVete
```

2

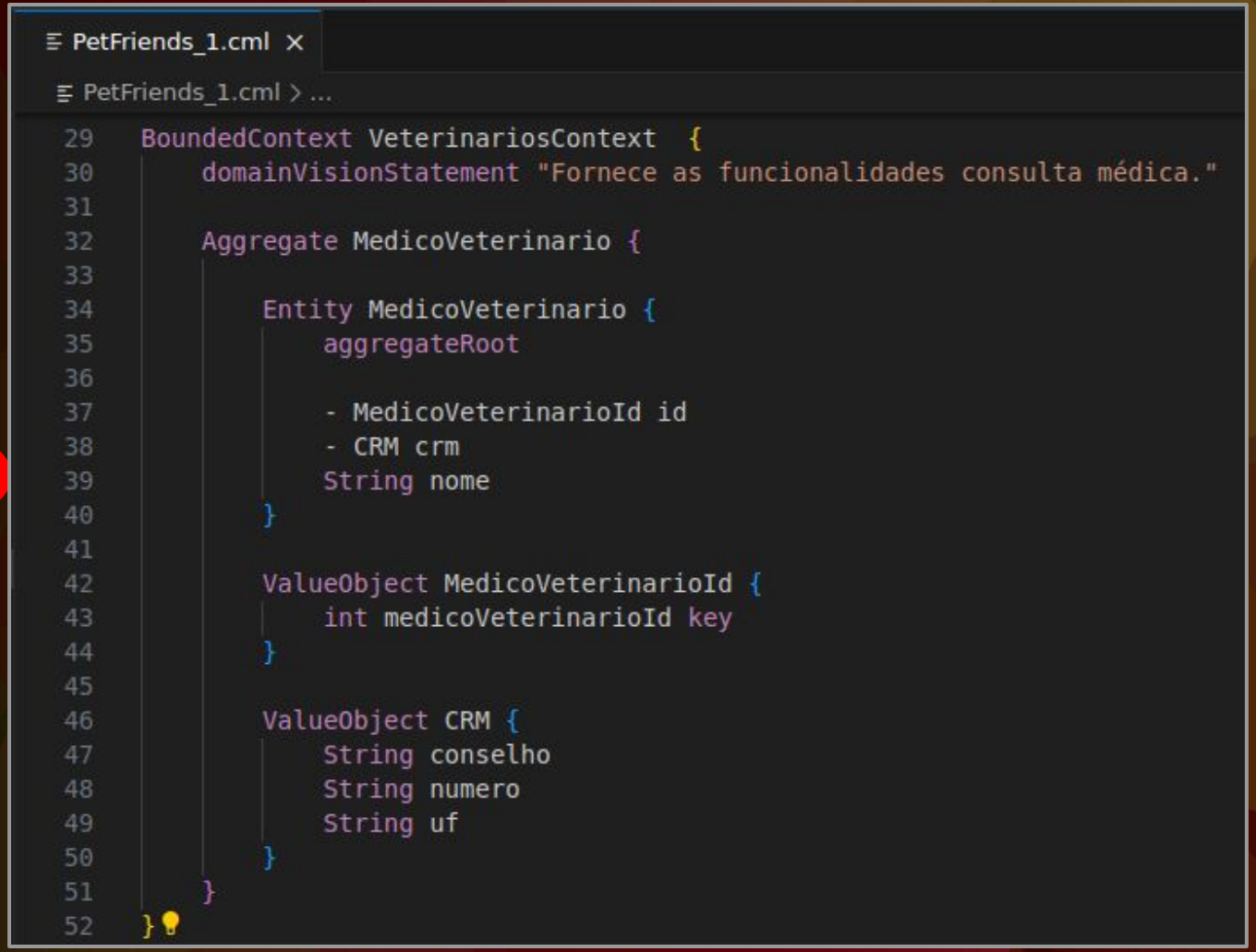



PetFriends_1.cml
PetFriends_1.cml > PetFriends
3
ContextMap PetFriends {
contains AgendamentoContext
contains ServicosContext
contains VeterinariosContext
contains ConselhoFederalVeterinariaContext
ServicosContext [D] <- [U, OHS] AgendamentoContext {
implementationTechnology = "RESTful HTTP"
}
VeterinariosContext [D] <- [U, OHS] AgendamentoContext {
implementationTechnology = "RESTful HTTP"
}
VeterinariosContext [D, ACL] <- [U] ConselhoFederalVeterinariaContext {
implementationTechnology = "RESTful HTTP"
}
}
BoundedContext AgendamentoContext
BoundedContext ServicosContext
BoundedContext VeterinariosContext
BoundedContext ConselhoFederalVeterinariaContext

PetFriends_1.cml

```
ContextMap PetFriends {…

BoundedContext AgendamentoContext {
    domainVisionStatement "Fornece as funcionalidades de agendamento para todos os contextos da empresa."
}

BoundedContext ServicosContext {
    domainVisionStatement "Fornece as funcionalidades de Banho, Tosa e Passeio."
}

BoundedContext VeterinariosContext  {
    domainVisionStatement "Fornece as funcionalidades consulta médica."
}

BoundedContext ConselhoFederalVeterinariaContext{
    domainVisionStatement "Fornece as funcionalidades de gerenciamento dos cadastros de médicos veterinários."
}
```

4


PetFriends_1.cml
PetFriends_1.cml > ...
5
BoundedContext VeterinariosContext {
domainVisionStatement "Fornece as funcionalidades consulta médica."
Aggregate MedicoVeterinario {
Entity MedicoVeterinario {
aggregateRoot
- MedicoVeterinarioId id
- CRM crm
String nome
}
ValueObject MedicoVeterinarioId {
int medicoVeterinarioId key
}
ValueObject CRM {
String conselho
String numero
String uf
}
}
}

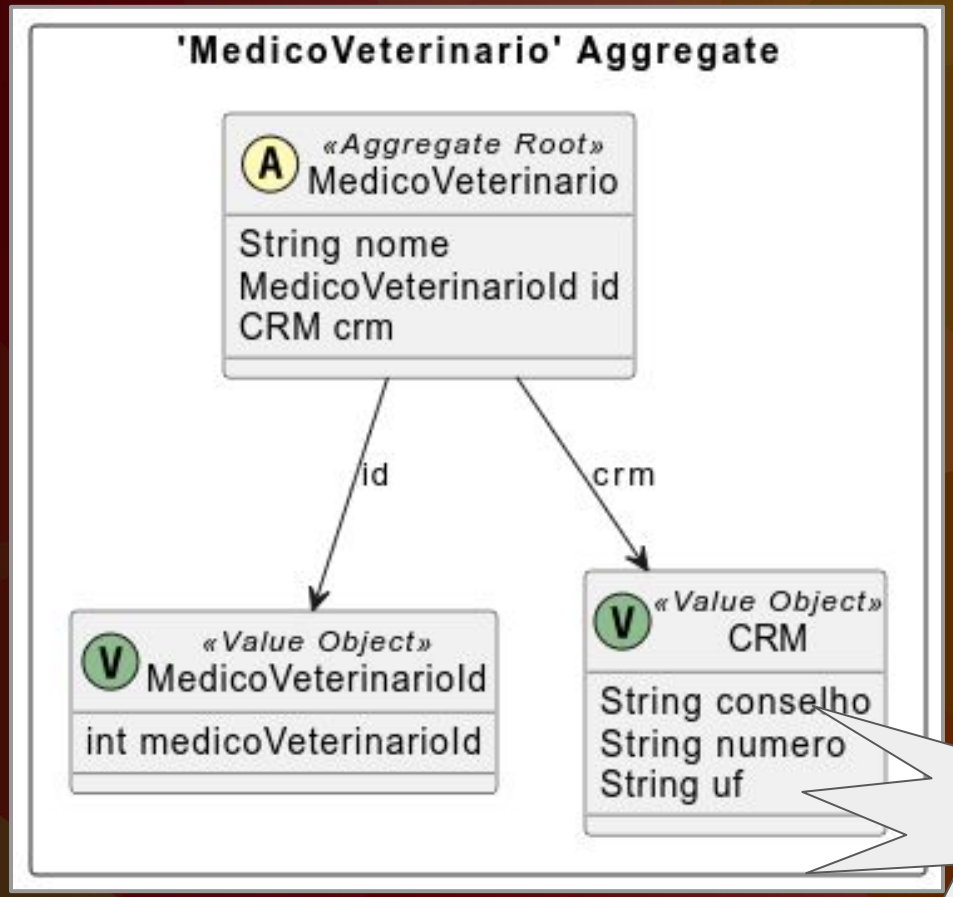
PetFriends_1.cml | PetFriends_1_BC_VeterinariosContext.puml 1

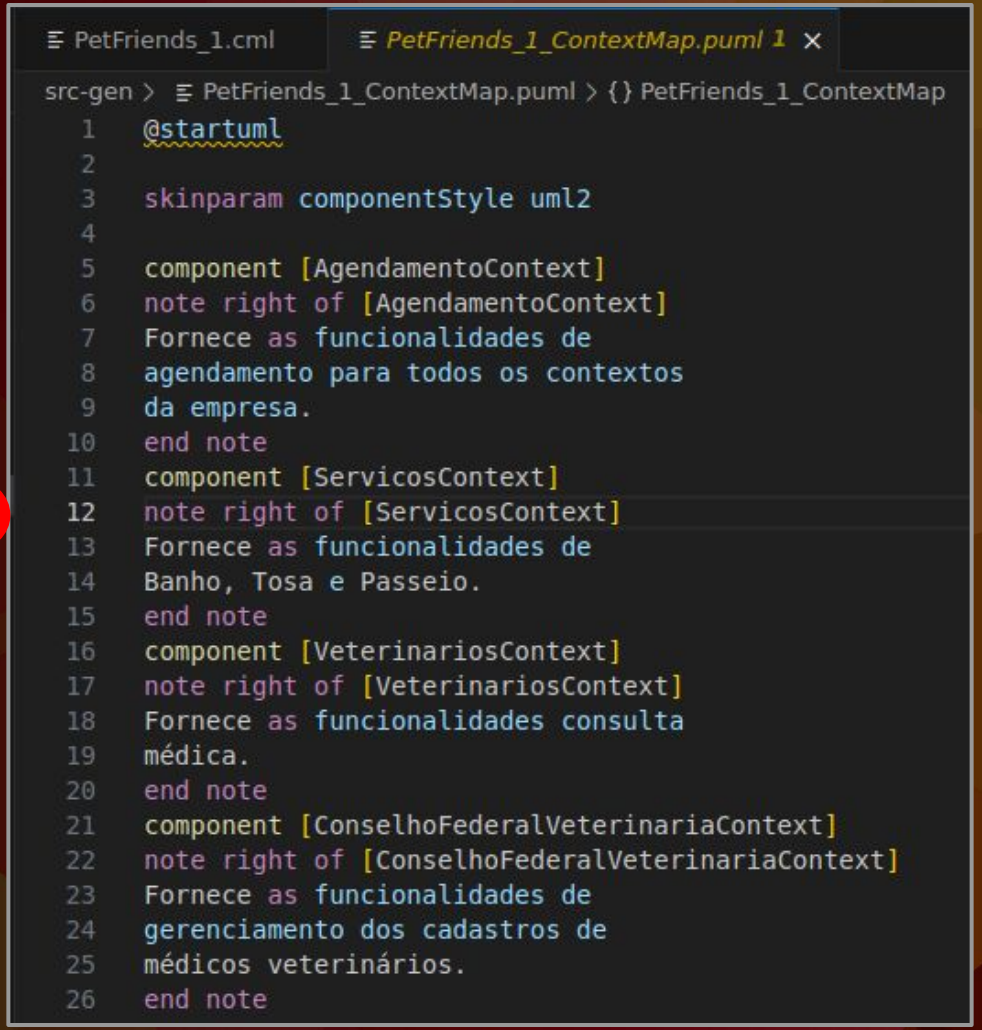
src-gen > PetFriends_1_BC_VeterinariosContext.puml > ...

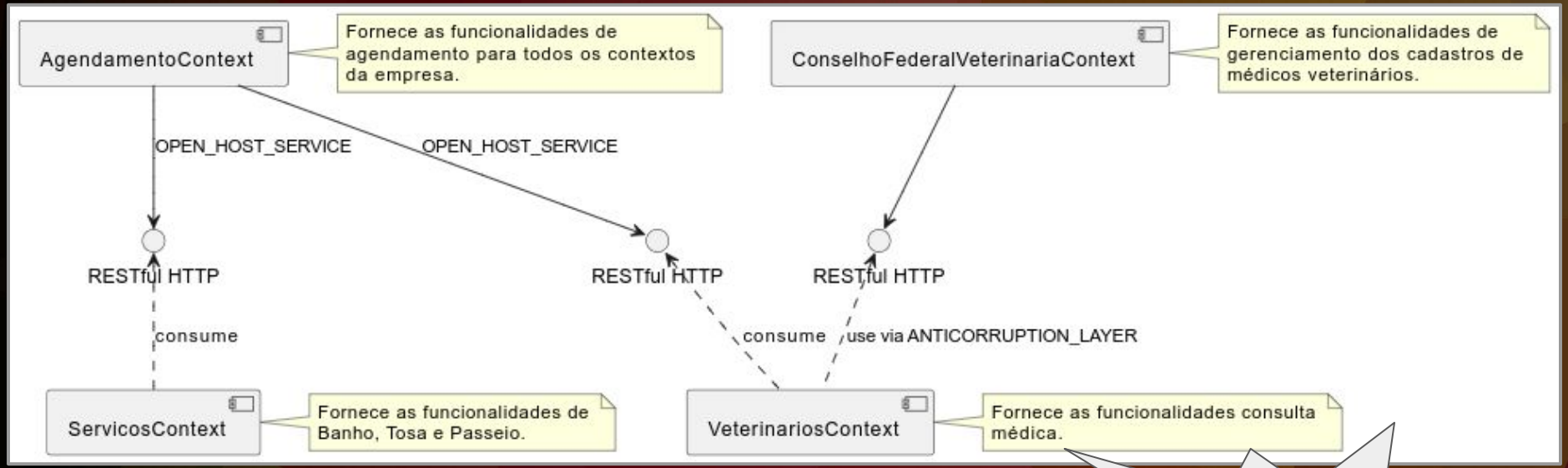
```
1   @startuml
2
3   skinparam componentStyle uml2
4
5   package "'MedicoVeterinario' Aggregate" <<Rectangle>> {
6       class MedicoVeterinario <<(A,#fffab8) Aggregate Root>> {
7           String nome
8           MedicoVeterinarioId id
9           CRM crm
10      }
11      class MedicoVeterinarioId <<(V,DarkSeaGreen) Value Object>> {
12          int medicoVeterinarioId
13      }
14      class CRM <<(V,DarkSeaGreen) Value Object>> {
15          String conselho
16          String numero
17          String uf
18      }
19  }
20  MedicoVeterinario --> MedicoVeterinarioId : id
21  MedicoVeterinario --> CRM : crm
22
23
24  @enduml
```

6

Botão direito do mouse. Generate PlantUML


'MedicoVeterinario' Aggregate
«Aggregate Root»
MedicoVeterinario
String nome
MedicoVeterinarioId id
CRM crm
id
crm
«Value Object»
MedicoVeterinarioId
int medicoVeterinarioId
«Value Object»
CRM
String conselho
String numero
String uf
ALT+D


PetFriends_1.cml
PetFriends_1_ContextMap.puml 1
src-gen > PetFriends_1_ContextMap.puml > {} PetFriends_1_ContextMap
@startuml
skinparam componentStyle uml2
component [AgendamentoContext]
note right of [AgendamentoContext]
Fornece as funcionalidades de
agendamento para todos os contextos
da empresa.
end note
component [ServicosContext]
note right of [ServicosContext]
Fornece as funcionalidades de
Banho, Tosa e Passeio.
end note
component [VeterinariosContext]
note right of [VeterinariosContext]
Fornece as funcionalidades consulta
médica.
end note
component [ConselhoFederalVeterinariaContext]
note right of [ConselhoFederalVeterinariaContext]
Fornece as funcionalidades de
gerenciamento dos cadastros de
médicos veterinários.
end note
7


AgendamentoContext
Fornece as funcionalidades de agendamento para todos os contextos da empresa.
ConselhoFederalVeterinariaContext
Fornece as funcionalidades de gerenciamento dos cadastros de médicos veterinários.
OPEN_HOST_SERVICE
OPEN_HOST_SERVICE
RESTful HTTP
RESTful HTTP
RESTful HTTP
consume
consume
use via ANTICORRUPTION_LAYER
ServicosContext
Fornece as funcionalidades de Banho, Tosa e Passeio.
VeterinariosContext
Fornece as funcionalidades consulta médica.

ALT+D