



UNIVERSIDADE FEDERAL DE OURO PRETO
DEPARTAMENTO DE COMPUTAÇÃO - DECOM
Professor: Tiago

SPRINT 01
Disciplina: BCC322 - Engenharia de Software 1

Aluno: Bernardo Cavanellas Biondini - 20.1.4112

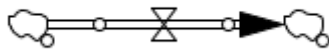
Ouro Preto
2022

O trabalho visa a construção de uma API que satisfaça as necessidades dos três sistemas apresentados pelo arquivo mdl. Para a primeira etapa, é preciso que seja identificado os casos de uso e critérios de aceitação as quais a API deve satisfazer, a criação de casos de teste a partir dos estudos de utilização da API e um diagrama UML que a represente.

CASOS DE USO

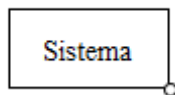
Foram identificados os seguintes casos de uso:

1) Fluxo isolado



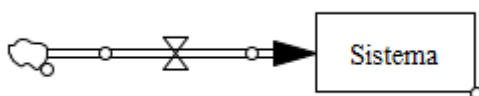
```
Modelo m = Modelo();  
Fluxo f1 = Fluxo("nome_fluxo", x)  
f1.conecta(NULL, NULL);  
m.adcFluxo(f1);  
m.execute();
```

2) Sistema isolado



```
Modelo m = Modelo();  
Sistema s1 = Sistema ("nome_sistema", x);  
m.adcSistema(s1);  
m.execute();
```

3) Sistema único com fluxo de entrada



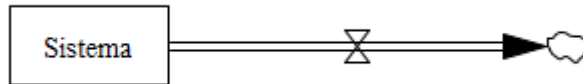
```
Modelo m = Modelo();  
Sistema s1 = Sistema ("nome_sistema", x);
```

```

Fluxo f1 = Fluxo("nome_fluxo", x)
f1.conecta(NULL, s1);
m.adcSistema(s1);
m.adcFluxo(f1);
m.execute();

```

4) Sistema único com fluxo de saída

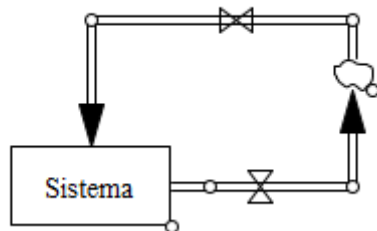


```

Modelo m = Modelo();
Sistema s1 = Sistema ("nome_sistema", x);
Fluxo f1 = Fluxo("nome_fluxo", x)
f1.conecta(s1, NULL);
m.adcSistema(s1);
m.adcFluxo(f1);
m.execute();

```

5) Sistema com fluxo de entrada e saída dele mesmo

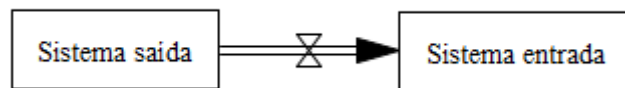


```

Modelo m = Modelo();
Sistema s1 = Sistema ("nome_sistema", x);
Fluxo f1 = Fluxo("nome_fluxo", x)
f1.conecta(s1, s1);
m.adcSistema(s1);
m.adcFluxo(f1);
m.execute();

```

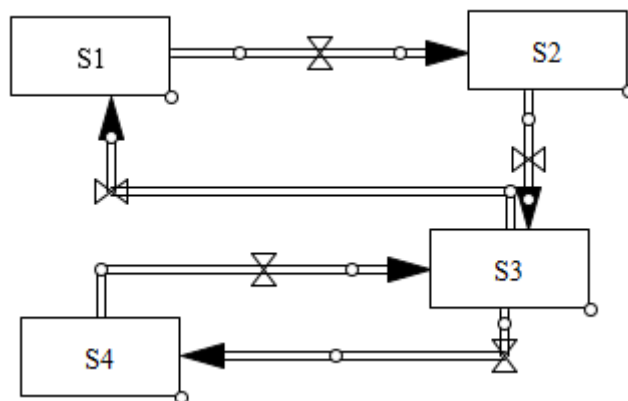
6) Sistema com fluxo de saída de um sistema e entrada para outro sistema



```

Modelo m = Modelo();
Sistema s1 = Sistema ("nome_sistema", x);
Sistema s2 = Sistema ("nome_sistema", x);
Fluxo f1 = Fluxo("nome_fluxo", x)
f1.conecta(s1, s2);
m.adcSistema(s1);
m.adcSistema(s2);
m.adcFluxo(f1);
m.execute();
  
```

7) Modelo com um conjunto de sistemas e fluxos de entrada e saída, como os casos acima citados



```

Modelo m = Modelo();
Sistema s1 = Sistema ("S1", x);
Sistema s2 = Sistema ("S2", x);
Sistema s3 = Sistema ("S3", x);
Sistema s4 = Sistema ("S4", x);
Fluxo f1 = Fluxo("nome_fluxo", x);
Fluxo f2 = Fluxo("nome_fluxo", x);
Fluxo f3 = Fluxo("nome_fluxo", x);
Fluxo f4 = Fluxo("nome_fluxo", x);
  
```

```

Fluxo f5 = Fluxo("nome_fluxo", x);
f1.conecta(s1, s2);
f2.conecta(s2, s3);
f3.conecta(s3, s4);
f4.conecta(s4, s3);
f5.conecta(s3, s1);
m.adcSistema(s1);
m.adcSistema(s2);
m.adcSistema(s3);
m.adcSistema(s4);
m.adcFluxo(f1);
m.adcFluxo(f2);
m.adcFluxo(f3);
m.adcFluxo(f4);
m.adcFluxo(f5);
m.execute();

```

DIAGRAMA UML

