

Instituto Federal de Educação, Ciência e Tecnologia do Espírito Santo – Campus
Serra

Bacharel em Sistemas de Informação

Igor Soares dos Santos

Italo Lourenço Trindade

TRABALHO 0 POO2

Serra / 2015

Igor Soares dos Santos

Italo Lourenço Trindade

TRABALHO 0 POO2

Trabalho apresentado ao Instituto Federal do Espírito Santo – Campus Serra como parte a das exigências da disciplina Programação Orientada a Objetos 2 do Curso de Bacharelado em Sistemas de Informação sob a orientação do professor Paulo como avaliação parcial para aprovação.

Serra / 2015

Sumário

Desenvolvimento	4
MiniMundo.....	4
Desenvolvimento	5
Chineses:	5
Japoneses:	6
Indianos:	6
Entrada de dados:	7
Exemplo:.....	7
Saída de dados:	7
Diagrama de Classe	8
MVC.....	9
Análise de dados	10

Desenvolvimento

Mini Mundo

Dois estudantes do IFES foram contratados por jogadores profissionais para desenvolver um simulador do jogo “Age of Empires” , com isso o simulador iria simular uma partida, uma guerra entre as civilizações.

O jogo basicamente consiste em colocar duas nações uma contra a outra, onde cada nação monta um conjunto de guerreiros para representa-la.

Os guerreiros são divididos em dois grupos os de defesa que são os defensores e os guerreiros de ataque que são os ofensores.

Para esse simulador do jogo, existem 3 possíveis nações a serem escolhidas:

- China: suas unidades de ataque, em geral, não são muito poderosas. As unidades de defesa têm propriedades bem interessantes.
- Japão: suas unidades de ataque são poderosas as de defesa nem tanto.
- Índia: suas unidades de ataque são medianas, mas possuem ótima defesa.

Basicamente serão feitas 4 filas de duelos:

- 1 fila de defesa e 1 de ataque para a nação1
- 1 fila de defesa e 1 de ataque para a nação2

A fila de ataque da nação1 ataca a fila de defesa da nação2.

A fila de ataque da nação2 ataca a fila de defesa da nação1.

A escolha da nação que ataca primeiro é feita por sorteio. Um ataque de uma nação implica em cada ofensor atacar 1 vez. Os ataques são feitos na fila de defensores da nação adversária.

Os ofensores atacam sucessivamente um defensor até que ele seja eliminado, uma vez que isso ocorra o próximo defensor da fila de defesa entrará em sua vez.

Depois que atacam os ofensores voltam para o final da fila de ofensores.

O jogo acaba se uma nação não tem mais ofensores ou defensores, ou seja, se um fila não tem mais guerreiros a nação perde.

Desenvolvimento

Primeiramente é necessário definir o que é um Guerreiro. Um Guerreiro é alguém que luta, podendo ser ofensor ou defensor e possui obrigatoriamente:

- Nome
- Idade
- Peso
- Energia: que deve ser inicializada em 100 no momento da criação do guerreiro.

Guerreiros morrem quando sua energia fica menor ou igual a 0.

A habilidade de atacar é definida no ofensor, mas o ofensor não sabe como atacar (sempre será um tipo de ofensor que terá essa habilidade).

A seguir apresentaremos os Guerreiros de cada nação:

Chineses:

Ofensores:

- 1) **Chun Ku:** os Chun Ku são arqueiros chineses. Retiram 5 pontos de qualquer defensor indiano e 10 pontos de qualquer defensor japonês.
- 2) **Gun Te:** os Gun te são guerreiros de grandes espadas. São especialmente bons contra a defesa japonesa, retirando 20 pontos de qualquer defensor. Quando atacam defensores indianos tiram 1 ponto mas morrem em seguida.
- 3) **Nok Tu:** os Nok Tu possuem grandes lanças. Retiram 5 pontos de qualquer defensor. Mas a cada ataque geram um Mangal de defesa que é colocado na fila de defensores dos chineses.

Defensores:

- 1) **Mangal de defesa:** é um boneco mecânico de defesa automática. Os mangais de defesa tiram 2 pontos de qualquer atacante.

2) **Montor do escudo:** os montores são guerreiros de grandes escudos cuja energia inicial é de 150 (é o único guerreiro que redefine esse valor). Quando os montores morrem eles levam consigo (matam) o guerreiro ofensor que os atacou.

3) **Mirk o conversor:** se atacados por Samurais os convertem em guerreiros Gun Te e colocam na fila de atacantes. Não sofrem qualquer dano de Samurais.

Japoneses:

Ofensores:

1) **Samurai:** guerreiros lendários japoneses. Qualquer defensor atacado perde 50 pontos, exceto Mirk o conversor.

2) **Ninja:** guerreiros sorrateiros japoneses. Qualquer defensor atacado perde 20 pontos.

Defensores:

1) **Tan tan:** os tan tan são guerreiros com escudos fixos nos braços. Quando morrem se transformam em ninjas.

Indianos:

Ofensores:

1) **Seak:** os seak são flexíveis unidades de ataque indianas, possuindo espada e arco. Os Seak retiram 25 pontos de qualquer guerreiro defensor atacado.

Defensores:

1) **Monge Leaf:** quando atacados por Ninjas ou Chun Kus recebem um escudo de ouro que os tornam inatacáveis por esses tipos de guerreiros ofensores, ou seja, ficam invulneráveis a Ninjas e Chun Kus.

2) **Monge Bomb:** quando atacados morrem, mas deixam o ofensor atacante com energia em 1 unidade.

3) **Monge Barrier:** quando morre coloca 2 monges barrier em seu lugar com metade da energia. Ou seja, 1 Monge Barrier de 100 se transforma em 2 de 50, esses 2 em 4 de 25, esses em 8 de 12 que por sua vez viram 16 de 6, depois 32 de 3, 64 de 1 e daí quando atacados morrem de vez.

O programa deverá ler 2 arquivos (nacao1.txt e nacao2.txt) e montar as filas de ofensores e defensores de cada nação.

Entrada de dados:

A entrada de dados de um arquivo de nação deverá ter o seguinte formato:

<nome da nação>

Ofensores:

<tipo do ofensor><nome do ofensor><idade 1 ><peso 1>

...

<tipo do ofensor n><nome do ofensor><idade n ><peso n>

Defensores:

<tipo do defensor><nome do defensor ><idade 1 ><peso 1>

...

<tipo do defensor n><nome do defensor ><idade n ><peso n>

Exemplo:

Japão

Ofensores:

1 NitTe 30 70

1 Fak 24 75

2 Full 23 77

1 Merc 55 80

Defensores:

1 Tark 60 50

1 Lan 40 30

Nesse caso foram criados 3 samurais (NitTe, Fak e Merc), 1 Ninja (Full) e 2 Tan tan (Tark e Lan)

Saída de dados:

A nação vencedora foi: <nome da nação> (Japão, Índia ou China)

A nação perdedora foi: <nome da nação> (Japão, Índia ou China)

Acabaram os guerreiros <categoria> (Ofensores ou defensores).

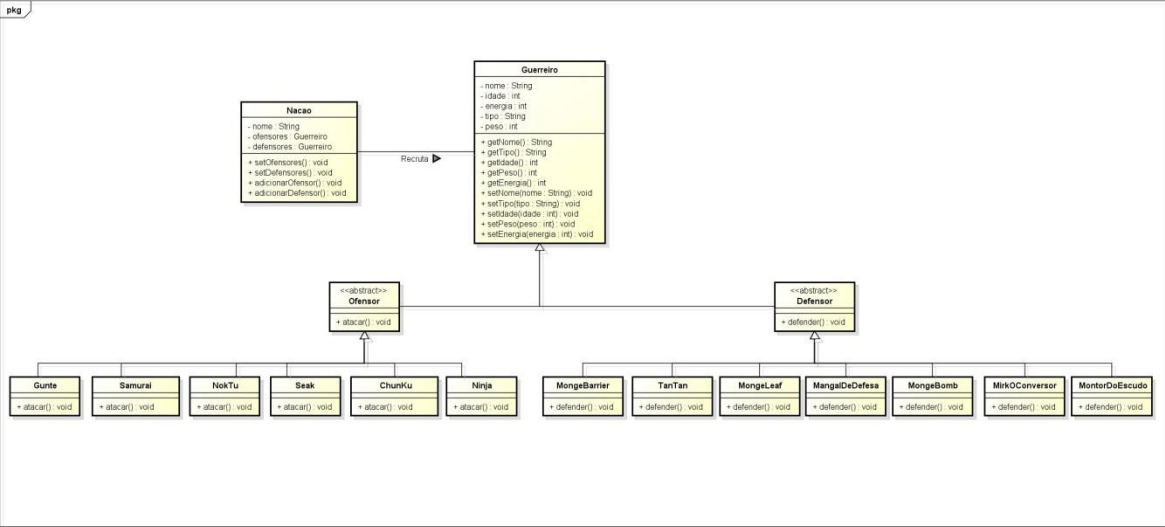
Exemplo:

A nação vencedora foi: Japão

A nação perdedora foi: Índia

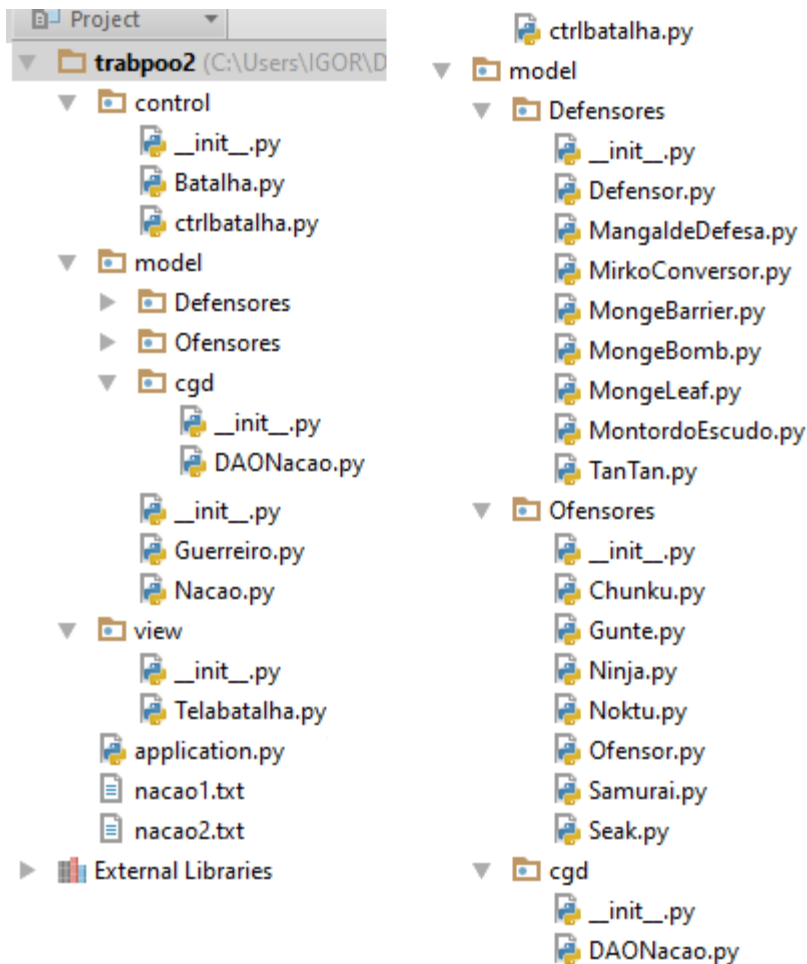
Acabaram os guerreiros Defensores.

Diagrama de Classe



powered by Astah

MVC



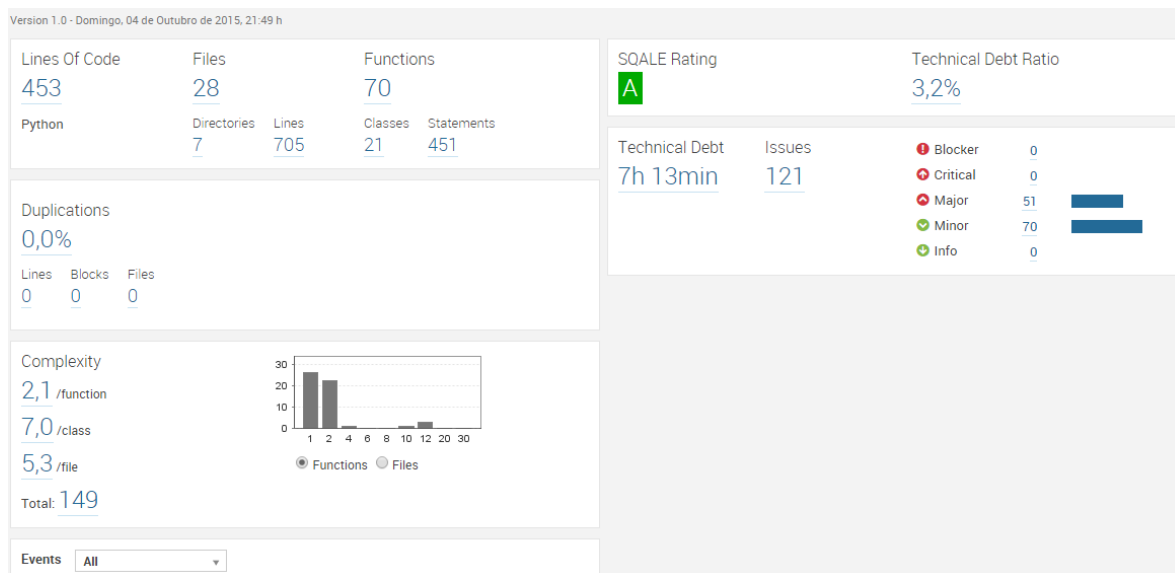
O padrão Model-View-Controller foi uma forma de organizar a aplicação com intuito de dividir em camadas, onde os objetos escondem as suas informações e como elas são manipuladas e apresentam uma simples interface para o seu uso.

Então criamos no modelo os pacotes Defensores e Ofensores onde encontra-se as classes de acordo com o diagrama de classe e o seu comportamento de como atacar e defender de cada guerreiro seja ele defensor ou ofensor, no cgd ficou a classe DAONacao com a responsabilidade de saber comunicar com o arquivo e comunicar com a classe Nacao. Já na camada do controle na classe CtrlBatalha ficou apenas com a responsabilidade de instanciar um objeto batalha onde a classe batalha possui o comportamento que vai se interagir com as classes DAONacao e TelaBatalha, sendo assim comunicando-se com o modelo e visão, portanto o encapsulamento do código acontece pois o modelo fica dividido em camadas e também consegue a separação dos conceitos e do código.

Análise de dados

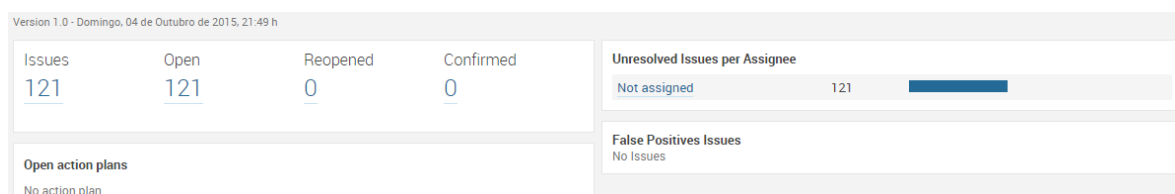
Resultados obtidos utilizando o SonarQube:

A imagem abaixo fornece várias métricas como número de linhas de código, número de classes, duplicações de código e complexidade ciclomática.



Agora são exibidas informações sobre dívida técnica, como por exemplo, a quantidade de problemas (Issues), ou seja, o número de regras que foram quebradas.

As issues são categorizadas quanto ao seu grau de severidade, indo de “Blocker” (mais grave) até “Info” (mais leve).



ISSUES

Project: Application Severity: All Status: All Assignee: Not assigned Resolution: Unresolved + More Criteria

Ordered by Update Date Found: 121

Minor Open 43 minutes

Rename this local variable "ofesoresDefesa" to ...
trabpoo2/control/Batalha.py

Minor Open 42 minutes

Rename this local variable "nacaoAtaque" to ...
trabpoo2/control/Batalha.py

Minor Open 42 minutes

Rename this local variable "proxAtaque" to ma...
trabpoo2/control/Batalha.py

Major Open 43 minutes

Remove those useless parentheses
trabpoo2/control/Batalha.py

Application

trabpoo2/control/Batalha.py

78 Lines of code 40min Debt 13 Issues

```

80 defensor.defender(ofensor, defensoresDefesa, ofensoresDefesa, ofensoresAtacantes)
81
82 print ofensor.getEnergia()
83
84 if(ofensor.getEnergia())>0:
85
86     nacaoAtaque.adicionarOfensor(ofensor)
87     ofensor = None
88
89     ofensor = nacaoAtaque.getLutador()

```

Remove those useless parentheses

Open Debt: 1min

A imagem abaixo mostra Time Machine

	04/10/2015	1.0	Issues	121
Lines of code	453		Blocker issues	0
Lines	705		Critical issues	0
Statements	451		Major issues	51
Files	28		Minor issues	70
Classes	21		Info issues	0
Functions	70		Technical Debt	7h 13min
Accessors				
	04/10/2015	1.0		
Comments (%)	0,0%		Complexity	149
Comment lines	0		Complexity /function	2,1
Public documented API (%)			Complexity /class	7,0
Public undocumented API			Complexity /file	5,3
	04/10/2015	-		

A imagem abaixo mostra o Compare

	Application
	1.0
	21:49
Lines of code	453
Complexity	149
Comments (%)	0,0%
Duplicated lines (%)	0,0%
Issues	121
Coverage	