



**Departamento de Engenharia Informática e de
Sistemas
Instituto Superior de Engenharia de Coimbra
Instituto Politécnico de Coimbra**

Licenciatura em Engenharia Informática

Curso Diurno

Ramo de Desenvolvimento de Aplicações

Unidade Curricular de Programação Avançada

Ano Letivo de 2019/2020

Meta 2

Trabalho prático

PLANET BOUND

José Paulo Ferreira Martins

Número de Aluno: 2016013630

Coimbra, 16 de junho de 2020

José Martins

Planet Bound

Relatório sobre trabalho prático para aprovação à disciplina Programação Avançada

Coimbra, 16 de junho de 2020

Índice

RESUMO ii

1. INTRODUÇÃO 1

2. FUNCIONALIDADES.....A

2.1. Funcionalidades ImplementadasA

2.2. Funcionalidade do menu “Cheats” para debuggingA

2.3. Funcionalidade do “Game Mode”A

2.4. Funcionalidade “Save/Load”A

3. Relacionado à meta 1.....B

3.1. Alterações relacionadas com a meta 1B

3.2. Necessidade de adicionar à meta 1B

3. INTERFACE GRÁFICAC

3.1. MVC.....C

3.2. Interação com o utilizadorC

3.3. Resultado finalC

3.4. Algumas opções tomadas G

4. CONCLUSÃO H

RESUMO

Este relatório serve para descrever e ajudar a compreender os métodos utilizados para a realização da segunda meta do trabalho pratico da disciplina de Programação Avançada.

Nesta segunda meta do trabalho pratico é pedida a implementação de uma interface gráfica, utilizando JAVA FX. Para isso teremos que usar a logica do programa desenvolvido na meta 1, podendo esse código estar suscetível a alterações.

1. INTRODUÇÃO

O trabalho realizado é a continuação da implementação realizada na meta 1. Para a realização desta meta a maioria do código da meta 1 foi reutilizado, sendo necessárias pequenas alterações.

A implementação da segunda meta foi sem dúvida muito menos trabalhosa e exigente dos alunos relativamente à meta 1, o que quero dizer com isto é que a quantidade de código menor e mais simples.

Para a realização do trabalho pratico da meta 2 foi necessário a implementação de toda a interface gráfica do jogo 'Planet Bound', essa interface gráfica foi implementada usando JAVAFX e um padrão de programação abordado nas aulas da Unidade curricular, o 'MVC'.

Neste Relatório é abordada a funcionalidades do programa, como foi implementado certos processos e decisões tomadas.

2. FUNCIONALIDADES

2.1. Funcionalidades Implementadas

Para a realização da meta 2 do trabalho pratico, foi necessário realizar pequenas alterações na parte da meta 1. As alterações realizadas foram a alteração de alguns objetos de classes que podiam ser 'private' referido pelo professor durante a defesa da meta 1.

Durante a fase de teste também me deparei com um erro de logica num dos estados relacionado com a finalização do jogo que teve que ser corrigido.

Todos as funcionalidades da meta 1 foram implementadas com uma interface gráfica totalmente funcional, é possível a qualquer momento guardar o estado atual do jogo ou carrega-lo. Foi adicionado um menu com cheats, para ajudar no debugging , e também 2 modos de jogo estra que consistem em aumentar o numero de artefactos necessários para ganhar e o outro um modo de sobrevivência em que o jogo só acaba quando se perder.

2.2. Funcionalidade do menu "Cheats" para debugging

Para auxiliar o debugging da aplicação foram implementados os seguintes menus:

- Fill Resources – enche os recursos ao máximo.
- Add Artefact – adiciona um artefacto.
- Add Drone – adiciona um mining drone.

2.3. Funcionalidade do "Game Mode"

Para tornar o jogo um pouco mais interessante decidi possuir 3 modos de jogo:

- Normal – coleciona 5 artefactos para ganhar.
- Hard – coleciona 100 artefactos para ganhar.
- Sobrevivência – o jogo só termina quando o jogador perder.

2.4. Funcionalidade "Save/Load"

Para a meta 2 era pedido que implementasse mos uma funcionalidade para permitir ao jogador guardar o estado atual e a possibilidade de carregar um estado previamente guardado.

Para permitir que o jogador aceda a estas funcionalidades optei por implementar um menu que se encontra disponível durante todo o jogo, dando a possibilidade do jogador gravar ou carregar durante qualquer parte do jogo.

3. Relacionado à meta 1

3.1. Alterações relacionadas com a meta 1

Durante a execução da meta 2 reparei em alguns problemas que não me tinha deparado na meta 1 e que precisavam de ser corrigidos. Um deles era o facto de que existia um erro de logica num dos estados, relacionado com quando o jogador ganhava o jogo. Este erro passava despercebido e era fácil de corrigir.

O segundo erro que encontrei tinha a ver com as compras depois de explorar o planeta, também era um erro simples e que se tornava menos perceptível quando testamos as funcionalidades numa interface de texto.

Também corriji o diagrama da máquina de estados pois como indicado pelo professor da defesa esta poderia levar a mal entendimentos.

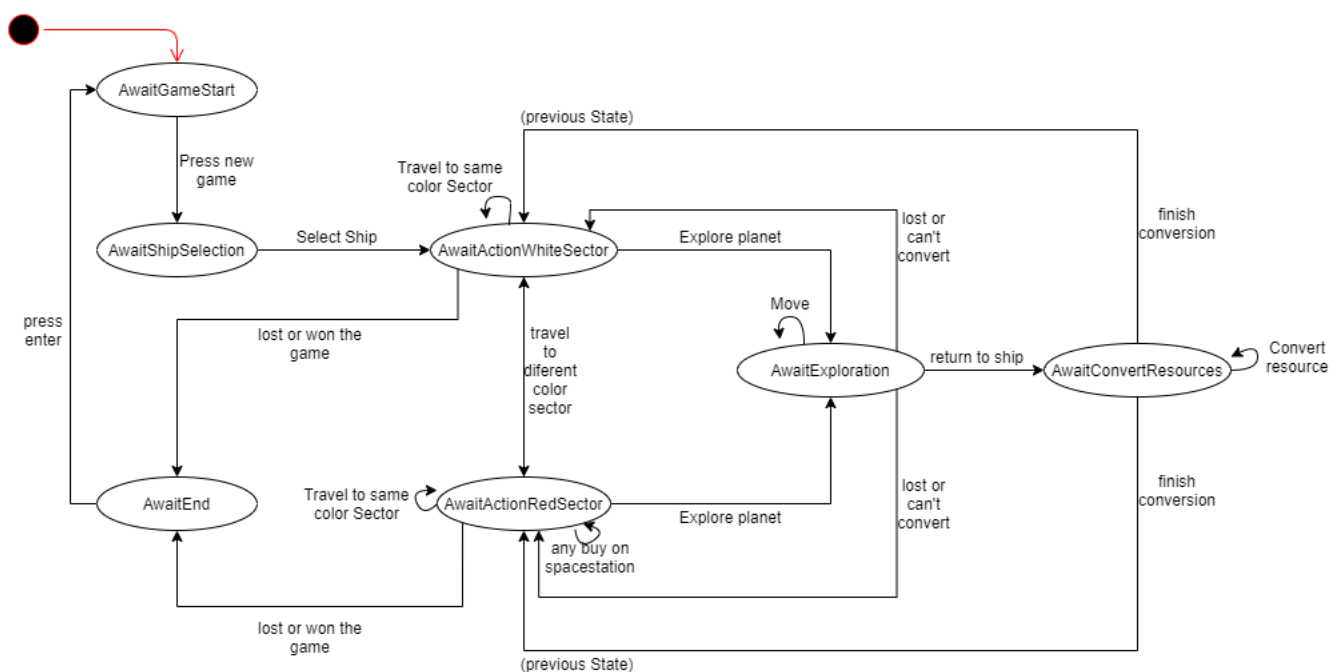


Figura 1 - Diagrama Maq.Estados

3.2. Necessidade de adicionar à meta 1

Para a realização da meta 2 tive que ajustar alguns métodos para aceder à informação dos dados, para poder mostrá-la nas Labels.

Na face de aterrar no planeta também tive que fazer alguns ajustes devido à maneira que usei para implementar essa mesma face na meta 1. Relacionado com esta mesma face também tive que implementar outra alteração para poder identificar o recurso que existia. Pois na altura tinha decido que era melhor não dar para identificar o recurso, mas na meta 2 achei que seria melhor essa possibilidade.

3. INTERFACE GRÁFICA

3.1. MVC

Para a implementação da meta 2 foi pedida a implementação de uma interface gráfica, para interagir com a mesma tinha que utilizar um padrão de design de software abordado nas aulas, conhecido como MVC (Model-view-controller) que consistem em separar a logica do programa das views (o que é apresentado ao utilizador).

Para poder implementar esta padrão primeiro criei as views e depois implementei um controller, que fica responsável por informar os dados (Model) das ações do utilizador e informar as views caso exista alterações nos dados. A parte do Model corresponde a toda a logica do programa, correspondendo ao trabalho implementado na meta 1.

3.2. Interação com o utilizador

Para poder jogar o utilizador apenas necessita do rato, em todas as situações não vi necessidade da utilização do teclado.

3.3. Resultado final

Neste ponto é apresentado como ficaram todas as view e o que estas representam.



Figura 2 - Barra menu

A barra de menu apresentada na figura 2 esta presente em todo o programa. As suas opções já se encontram descritas nos pontos 2.2, 2.3, 2.4.

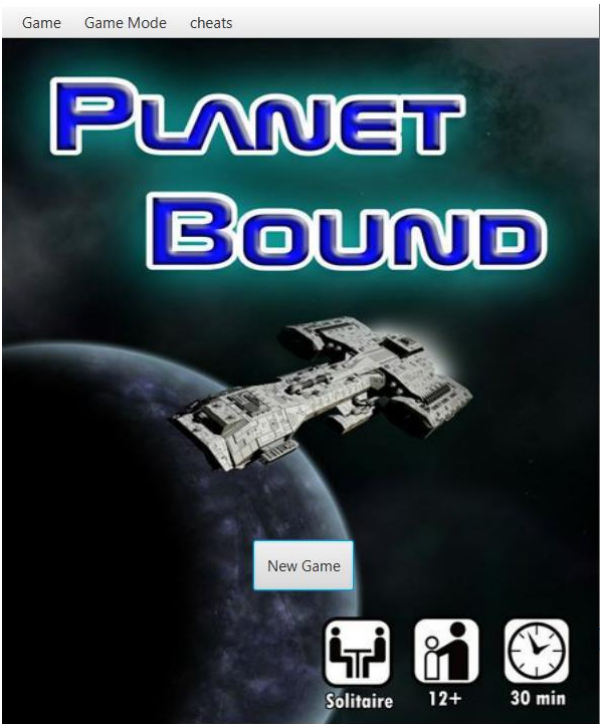


Figura 3 - Menu inicial

No menu apenas é possível iniciar um novo jogo, pelo facto do utilizador ter acesso à barra que já fornece uma serie de opções.

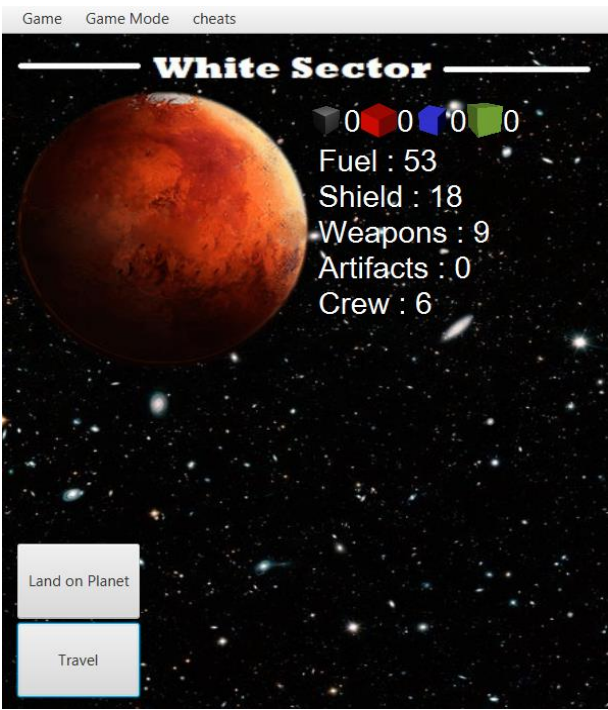


Figura 4 - White sector

A figura 4 representa a interação com o utilizador no sector branco, é mostrado ao utilizador todas as informações necessárias relacionadas com o jogo e as opções que este pode tomar.

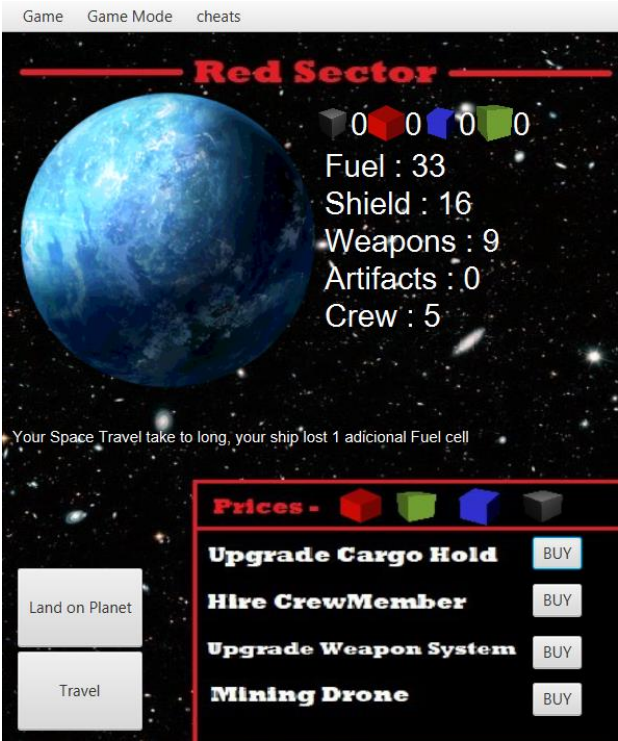


Figura 5 - Red Sector

Na figura 5 é representado a interação com o utilizador no sector vermelho, tal como no branco representado na figura 4. A única difference é que neste setor o utilizador pode realizar algumas ações extra.

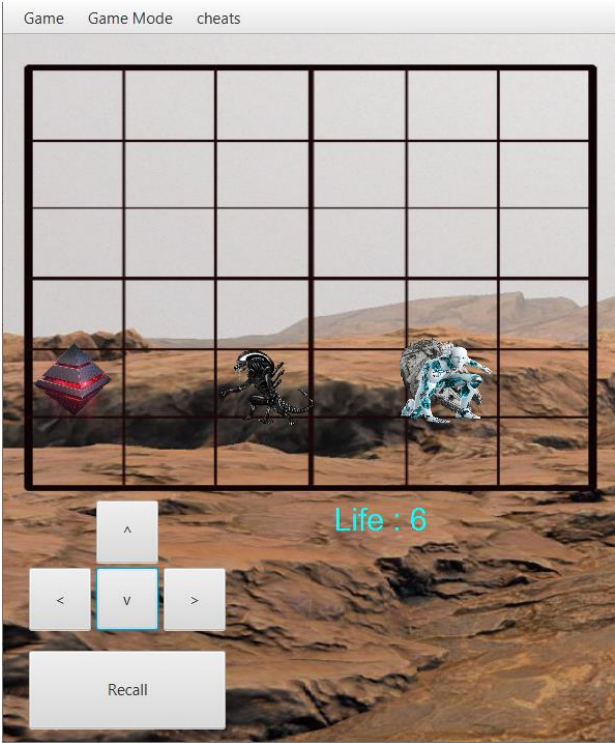


Figura 6 - Vista aterrar

A figura 6 representa a interação com o utilizador durante a exploração do planeta, para isso foi realizado algo simples em que o robô branco representa o jogador e o alien também é identificável. Os recursos são representados por cubos de diferentes cores, e na figura vemos um artefacto. O ponto de partida é representado por uma nave.

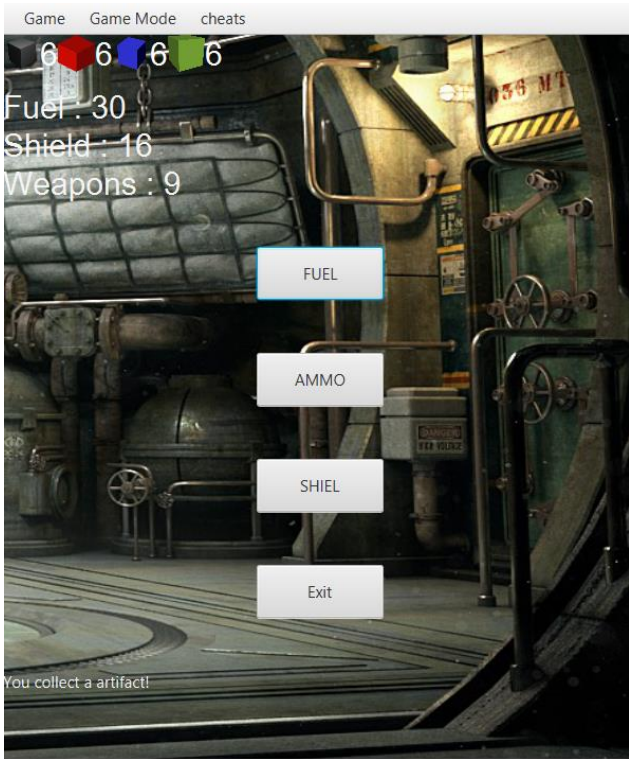


Figura 7 - vista para conversao de recursos

A figura 7 representa a conversão de recursos, que se torna bastante simples com apenas as opções possíveis.



Figura 8 - Fim do jogo

Na figura 8 esta representado um exemplo da visão final do jogo, também optei por algo simples que é dizer se ganhou ou perdeu o jogo e mostrar o número de artefactos coletados, que é um dos principais objetivos do jogo.

3.4. Algumas opções tomadas

As duas views para interação com o sector deve-se à maneira como optei por programar a máquina de estados na meta 1, que está relacionado com o facto de ter integrado as compras possíveis na estação espacial junto com o sector vermelho.

Na view para interação com o utilizador na face de exploração tinha a opção de desenhar o tabuleiro, mas preferi integrá-la na imagem de fundo visto que esta era estática.

Em ambas as views de interação com o sector, em vez de dizer a cor do planeta optei por mostrar uma imagem de um planeta com a cor correspondente, que foi algo que achei que ficaria mais engraçado e tornaria a vista destes menus mais apelativa.

No menu inicial optei por apresentar apenas uma opção pelo facto de que a barra de menu já possui uma variedade de opções.

4. CONCLUSÃO

Na minha opinião foi um trabalho porreiro de realizar, deu o seu trabalho, mas no final é bom poder jogar o jogo que fizemos.

A realização deste trabalho no geral ajudou-me a compreender melhor a utilização de máquinas de estados, apesar de que estas fazem parte da meta 1 acho importante voltar a referi-la porque percebo a sua importância e as facilidades que estas oferecem. Também consegui perceber melhor em que consiste o padrão MVC e a saber como implementá-lo.