

JAVA 1

SpacePigFighter

Authors : Vincent Reynaert, Nicolas Sobczak

Contents

1	Game presentation	1
1.1	What is it ?	1
1.2	Rules	1
1.2.1	Animal class	1
1.2.2	Animal specialAttack	2
1.2.3	Meteorites malus	2
1.2.4	Stuff choice	2
1.3	How it is thought/programmed	2
2	Story: what happens when you launch the game	3
2.1	Start the game	3
2.2	Part 1 of the game	3
2.3	Part 2 of the game	3
2.4	End	3
3	Development part	5
3.1	UML	5
3.2	Technical part: class description	5
3.2.1	Main	5
3.2.2	FileManagement	5
3.2.3	Player	6
3.2.4	Set the game	6
3.2.5	The Space :	6
3.2.6	The FightArea :	6
3.3	Encountered difficulties	6
3.3.1	Special action	6
3.3.2	Exception	6
4	Conclusion	7

Chapter 1

Game presentation

1.1 What is it ?

"Space Pig Fighter" is a game that is played in the terminal by 2 players. Each player is a space pig and have to beat the other one.

A game happens in 2 phases. The first one is a spacecraft battle. The second one is a melee battle. Each spacecraft has several characteristics. Each pig has several characteristics and some weapon.

1.2 Rules

1.2.1 Animal class

Here are the concept we chose :

Animal class	Life	Force	Resistance	Special attack
Bear	mid	mid	big	damageAnnulation
Chicken	low	big	mid	triple attack
Duck	big	mid	low	fly
Pig	mid	low	big	moreDamage
Tiger	mid	big	low	paralyze foe which can't attack next turn

Here are the exact values we chose :

Animal class	Life (hp)	Force	Resistance	Special attack
Bear	1000	110	40	damageAnnulation: nn
Chicken	800	130	20	triple attack: nn
Duck	1200	110	0	fly: nn
Pig	1000	90	40	moreDamage: nn
Tiger	1000	130	0	paralyze foe which can't attack next turn: nn

1.2.2 Animal specialAttack

- damageAnnulation
- moreDamage
- paralyze foe which can't attack next turn
- tripleAttack, 1 turn to charge
- fly, dodge current attack and attack next turn

1.2.3 Meteorites malus

Size	Malus
small	-20 hp
medium	-50 hp
big	-100 hp

1.2.4 Stuff choice

You have 2 skill points to share between offensif and defensif stuff. You may choose to boost your attack at the expense of the your defense or to boost your defense at the expense of the your attack. Unless you prefer to choose a well balanced build.

Build	Attack points	Defense points
Offensive	2	0
Well balanced	1	1
Defensive	0	2

Here are the bonus value of each stuff :

Build	Offensive stuff	Stats bonus	Defensive stuff	Stats bonus
Offensive	Axe	40	None	00
Well balanced	Sword	20	Helmet	20
Defensive	None	00	Shield	40

1.3 How it is thought/programmed

Each player plays when it is its turn.

Chapter 2

Story: what happens when you launch the game

Welcome players

2.1 Start the game

- player 1 is invited to choose his animal, enter animal's pseudo and color (pink by default), his spacecraft's color (gray by default).
- player 2 is invited to choose his animal, enter animal's pseudo and color (pink by default), his spacecraft's color (gray by default).

2.2 Part 1 of the game

- launch part1 of the game: space battle. You have to find the right location of the other player's spacecraft by entering a position. Each player try to guess turn by turn. You have to be careful, avoid meteorites ! Otherwise your pig's life will decrease.
- when a player find the other one's spacecraft, he climbs aboard and it's time for part 2 of the game.

2.3 Part 2 of the game

Players are welcomed to choose a stuff build in order to fight the other player.

1 Turn:

- Player 1 choose an action for his pig to do (choose between normal attack, special attack and scream)
- Player 2 choose an action for his pig to do (choose between normal attack, special attack and scream)
- Resolution

Game is over when a pig has no life point left. Since the resolution happens after both player's action, the result can be a draw.

2.4 End

At the end of the game, a file is created with the game summary written in it.

Chapter 3

Development part

Each player plays when it is its turn.

3.1 UML

Here is the UML diagram of the program:

<insert here: UML>

Blue classes are abstract classes.

Green classes are interface.

Pink classes are enumeration.

Yellow classes are the two main classes from the 2 different parts of the game.

3.2 Technical part: class description

<insert here: brief class description>

3.2.1 Main

This class contains the main functions:

- main : main function that calls all the following functions.
- playerCreation : function that create the 2 players.
- part_1 : function that runs game part 1.
- part_2 : function that runs game part 2.

3.2.2 FileManagement

This class contains all useful functions to save the game story in a file.

3.2.3 Player

We created a Player class that keeps all information about each player.

3.2.4 Set the game

- set Player class for each player.
- set Space class with 2 CubeEnvironment (1 for each player). Each CubeEnvironment is set with 3 meteorites and 1 spacecraft.
- set FightArea class with 2 pigs. Each pig is initialized with stuff selected by the player.

3.2.5 The Space :

It is composed by 2 CubeEnvironments created thanks to the 2 Players.

3.2.6 The FightArea :

It is composed by 2 Animals created thanks to the 2 Players.

3.3 Encountered difficulties

3.3.1 Special action

Special actions are very different. So we had to think our code so that it would be able to welcome each special action. (We may had to modify our code ?)

3.3.2 Exception

We created an exception.

Chapter 4

Conclusion

We think our project cover lots of different aspect of java language. Besides we enjoyed doing this game. That's why we may add a graphical interface in the future.

