<FlappyBird>

Programming 3 Final Project

<Yassine Haouet>

<BYI1SZ>

1. DESCRIPTION OF THE TASK

Flappy Bird is a game in which the player controls the bird, which moves persistently to the right. The player is tasked with navigating the bird through pairs of pipes that have equally sized gaps placed at random heights. The bird automatically descends and only ascends when the player presses the space key. Each successful pass through a pair of pipes awards the player one point. Colliding with a pipe or the ground ends the gameplay. The highest score is displayed at the end of the game.

2. STRUCTURAL DESCRIPTION

1.1 DESCRIPTION OF THE CLASSES

1.1.1 Bird

Responsibilities

This class represents the bird which is the main character of the game

Attributes

width	the width of the image
height	the height of the image
x	the position of the bird on the x-axis
y	the position of the bird on the y-axis
dy	the displacement of the bird
strImage	the image path
icoBird	the icon painted from the image
imgBird	the bird image
PAUSE	this is a constant. it refers to the pause duration of the bird thread (fly)

Methods

getX()	Getter of attribute X
getY()	Getter of attribute Y
getWidth()	Getter of attribute width
getHeight()	Getter of attribute height
getImgTube()	Getter of attribute imgBird
setX()	Setter of attribute X
setY()	Setter of attribute Y

1.1.2 Tube

Responsibilities

This class represents the tube which is the obstacle that the bird will collide with.

Attributes

width	the width of the tube image
height	the height of the tube image
x	the position of the tube on the x-axis
y	the position of the tube on the y-axis
strImage	the image path
icoTube	the icon painted from the tube image
imgTube	the tube image

Methods

getX()	Getter of attribute X
getY()	Getter of attribute Y
getWidth()	Getter of attribute width
getHeight()	Getter of attribute height
getImgTube()	Getter of attribute imgTube
setX()	Setter of attribute X
setY()	Setter of attribute Y

1.1.3 Chrono

Responsibilities

This class represents the thread responsible for the shifting of the background and the tubes

Attributes

PAUSE	It represents the thread pause duration. It is responsible for the smoothness of the movements.

Methods

Run()	Run () function of the thread.

1.1.4 Keyboard

Responsibilities

the Keyboard class is that it implements the KeyListener interface and it is responsible for reading the key inputs of the user and make the program act accordingly

Attributes

-

Methods

keyPressed()	The keyPressed() function is called once every time a key is pressed
keyReleased()	-
keyTyped()	-

1.1.5 Scene

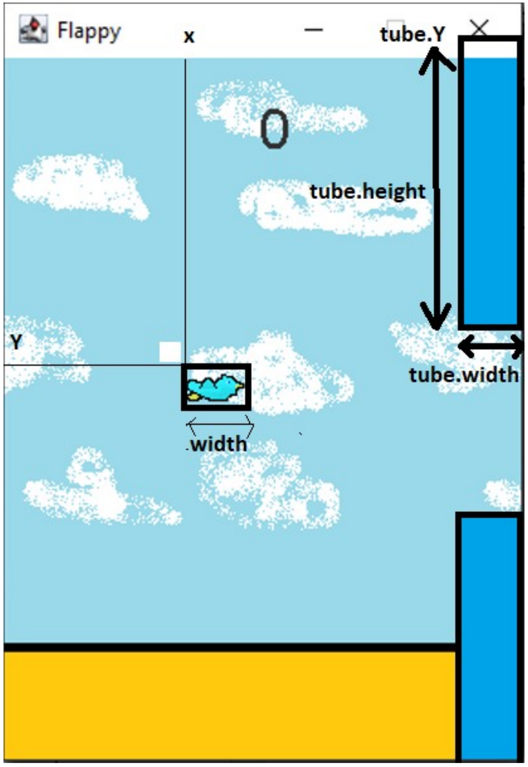
Responsibilities

This class represents the main class of the project. It extends JPanel and gathers all the components of the game

Attributes

icoBackground	The icon painted from the background image
imgBackground	The background image
ArrayList<Tube> tubes	The list of tubes : the scene consists of 6 pipes which will be shifting throughout the game.
bestScore	The highest score of the game
score	The player current score
font	The font of the text to be written on the screen
flappyBird	The bird character
BACKGROUND_WIDTH	The width of the background image
DISTANCE_BETWEEN_TUBES	Distance between two consecutive tubes
GAP_BETWEEN_UP_DOWN_TUBES	The gap length between up and down pipe
xBackground	To be used for the background shifting throughout the game
endOfGame	Boolean variable which indicates the end of the game when the bird hits an obstacle.
random	Random integer variable used for positioning the pipes.

Methods

tubeShifting(Graphics g)	This function is responsible for the shifting of the 6 pipes throughout the game. For example, for TubeUp1, its X coordinate keeps decreasing until it reaches -100. At that time, it gets shifted by 100 pixels after the TubeUp3
collideWith(Tube tube, Bird bird)	Tests if the bird collides with pipe  Flappy x tube.Y 0 tube.height Y tube.width width No collision as long as one of these conditions is met : Y>tube.Y+tube.height X+width<tube.X tube.X+tube.width<X
gameOver()	Game ends when a collision is detected
Score()	It updates the score when the bird passes successfully between the two pipes.
saveBestScore()	Save the best score into an external file
loadBestScore()	Loads the best score from the external file
paintComponent(Graphics g)	This function is responsible for painting the frame throughout the game. It is executed everytime repaint() is called inside the Chrono thread run() function.

1.1.6 Main

Responsibilities

This class is responsible

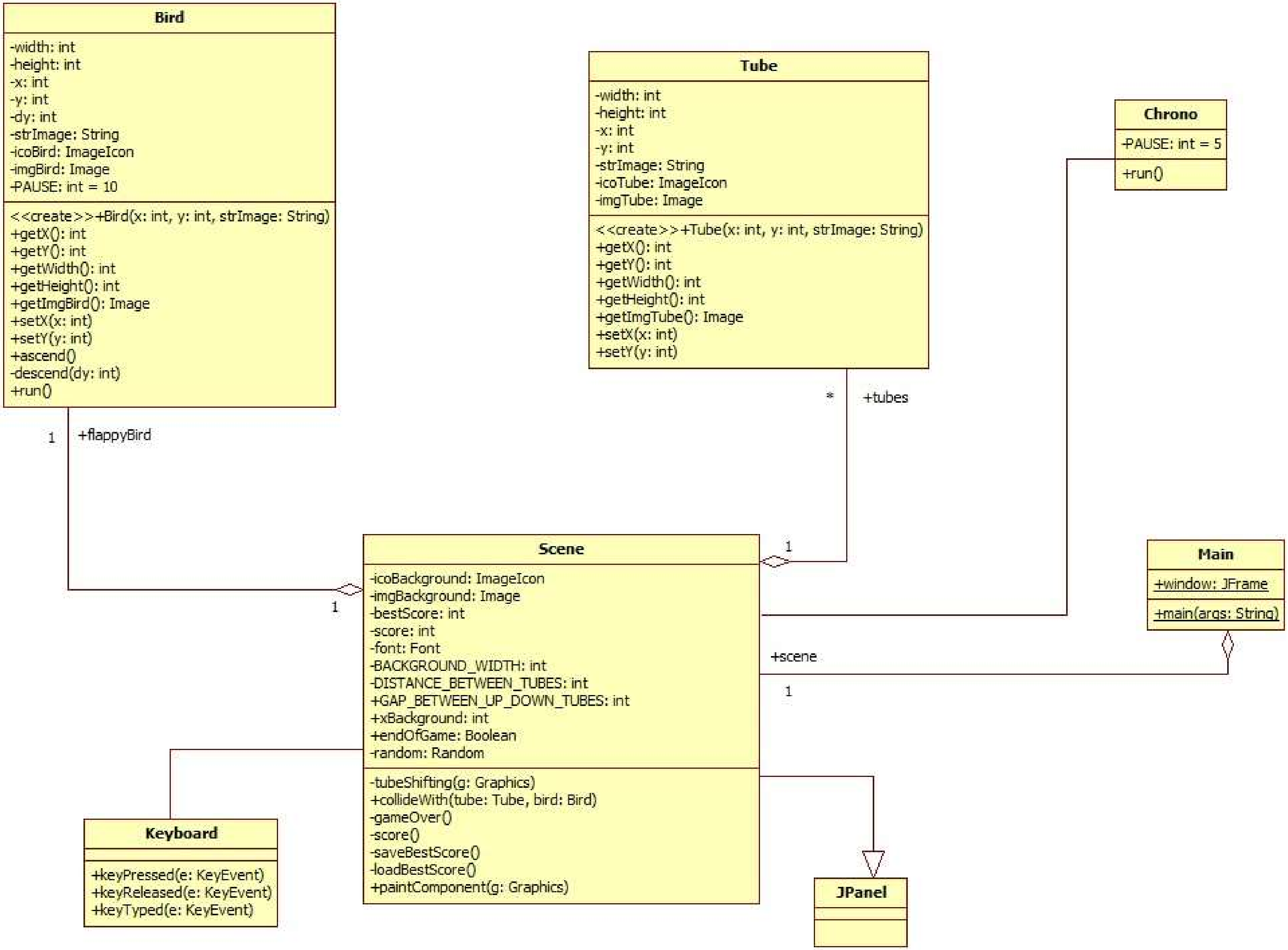
Attributes

window : JFrame	The frame in which the game will be played
scene : Scene	The game scene

Methods

-

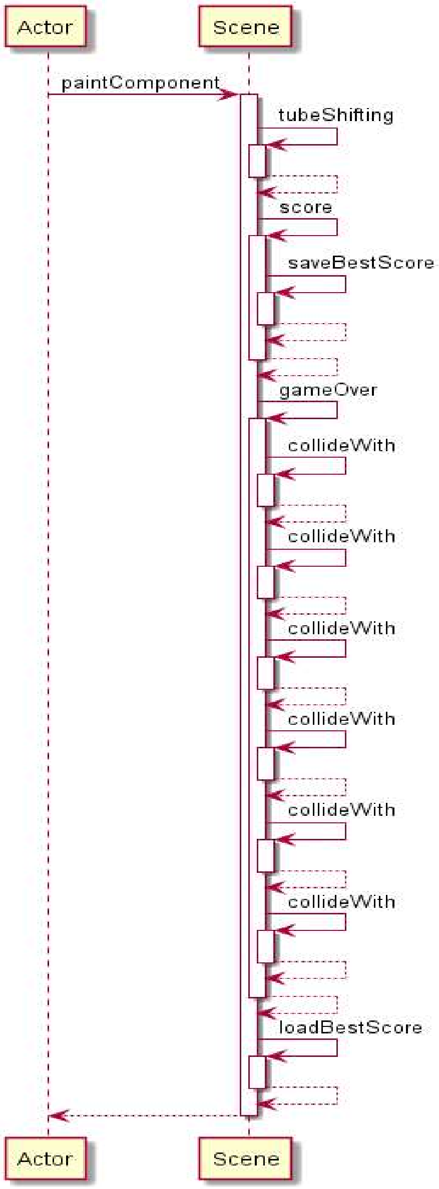
1.2 Class Diagram



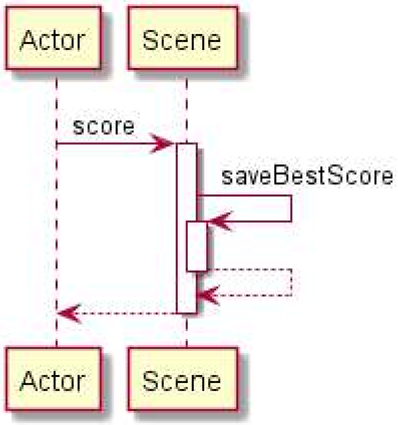
2. Behavioral description

2.1 Sequence Diagrams

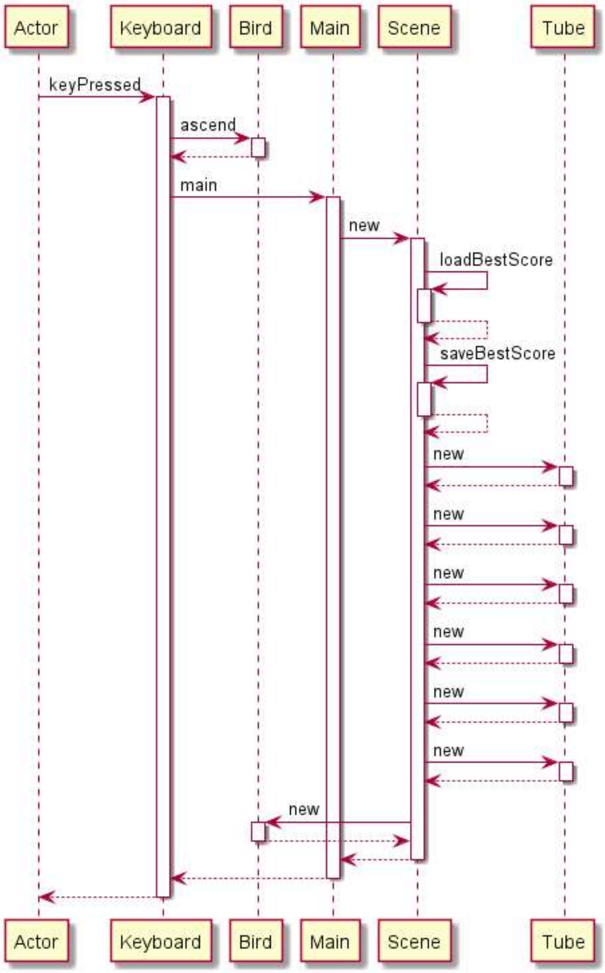
2.1.1 Paint the components



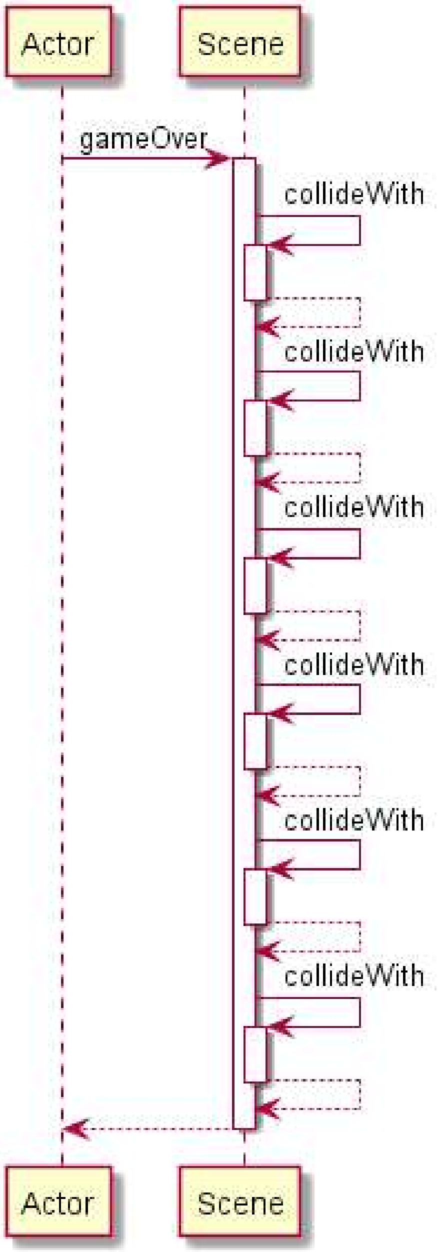
2.1.2 update the score



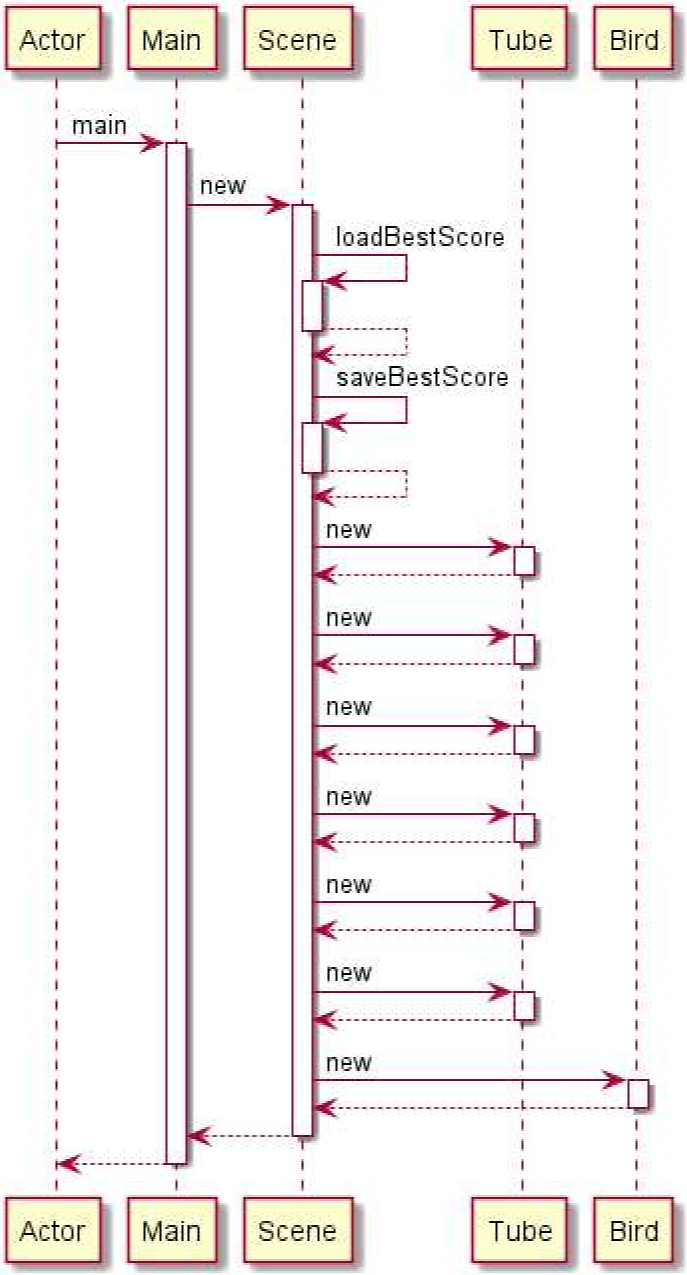
2.1.3 move the bird



2.1.4 the game is Over



2.1.5 Main execution



3. JUNIT TEST CASES

3.1.1 testSetX

```
@Test
	public void testSetX() {
		flappy.setX(200);
		Assertions.assertEquals(200, flappy.getX());
	}
```

Explanation : this test case verifies if if the X setter and getter work properly.

3.1.2 testSetY

```
@Test
	public void testSetY() {
		flappy.setY(200);
		Assertions.assertEquals(200, flappy.getY());
	}
```

Explanation : this test case verifies if if the Y setter and getter work properly.

3.1.3 detectCollideWhenCollideTest()

```
@Test
	public void detectCollideWhenCollideTest(){
		scene.flappyBird.setX(400);
		scene.flappyBird.setY(10);
		assertTrue(scene.collideWith(scene.tubes.get(0), scene.flappyBird));

		}
```

Explanation : this test case verifies if the method collideWith returns true when collision happened between the bird and the tube.

3.1.4 detectCollideWhenNotCollideTest()

```
@Test
	public void detectcollideWhenNotCollideTest(){
		scene.flappyBird.setX(150);
		scene.flappyBird.setY(150);
		assertFalse(scene.collideWith(scene.tubes.get(0), scene.flappyBird));

	}
```

Explanation : this test case verifies if the method collideWith returns false when no collision happened between the bird and the tube.