



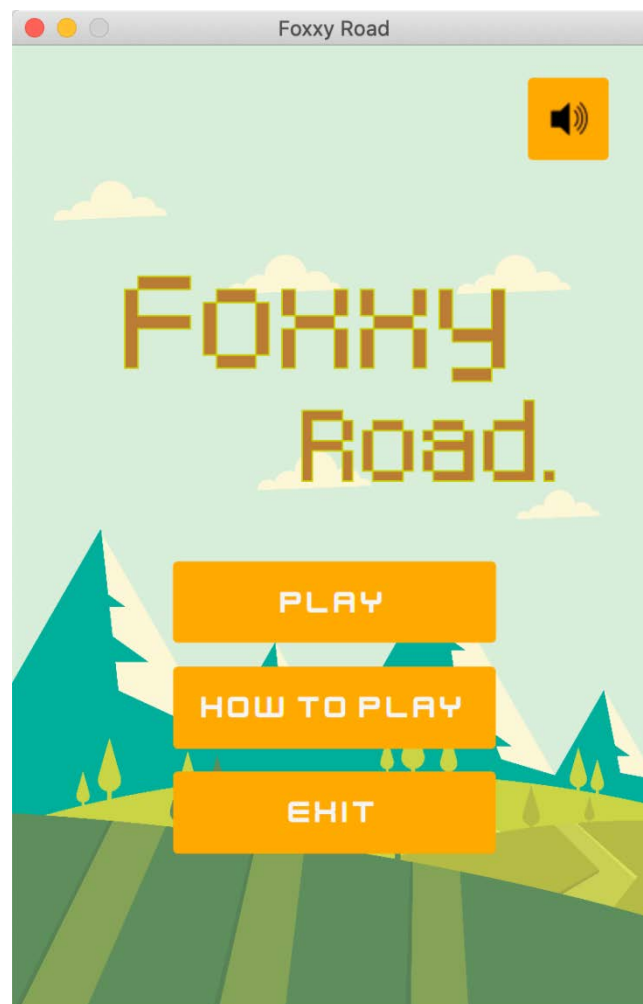
By Posawat Maileung 6131030721  
Pattanut Phetmune 6131035921

# ABOUT THE GAME

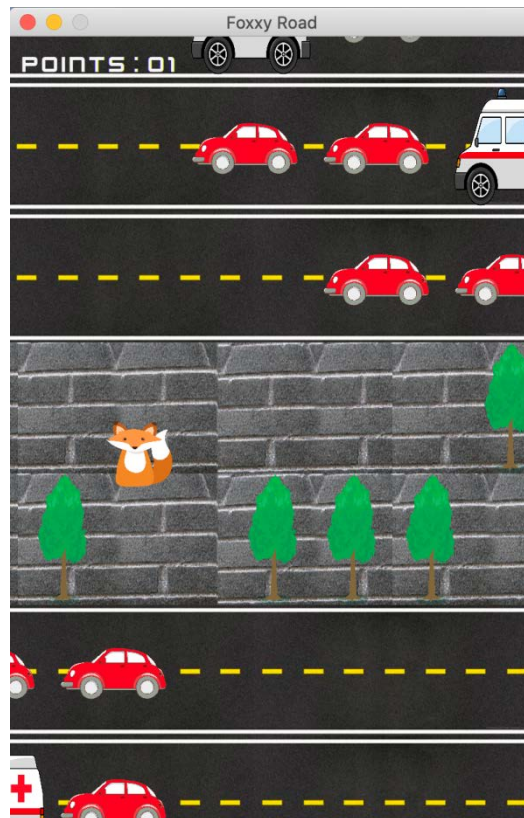
"Foxyxy road" is a 2D java game create by Java FX. The game is vertical game where the player controls, attempting to cross road which have many obstacle instance of car tree. The goal of this game is you need to across highest road as possible.

## GAME SCREENS

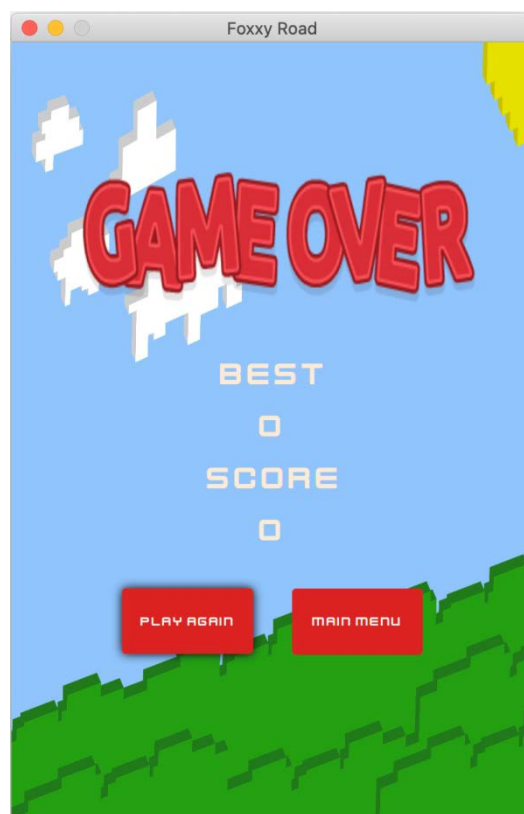
Main menu:



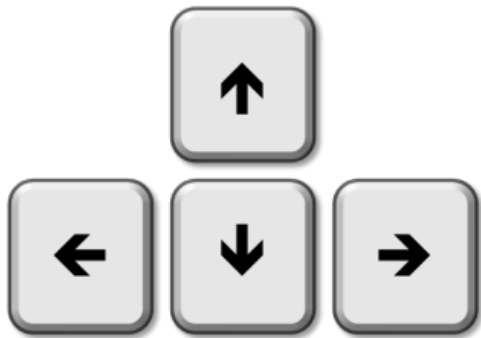
Gameplay Screen:



Game over Screen:



## CONTROLS



**Press ArrowUp to go forward.**

**Press ArrowDown to go backward.**

**Press ArrowLeft to go left.**

**Press ArrowRight to go right.**

## GAME MODELS

Player

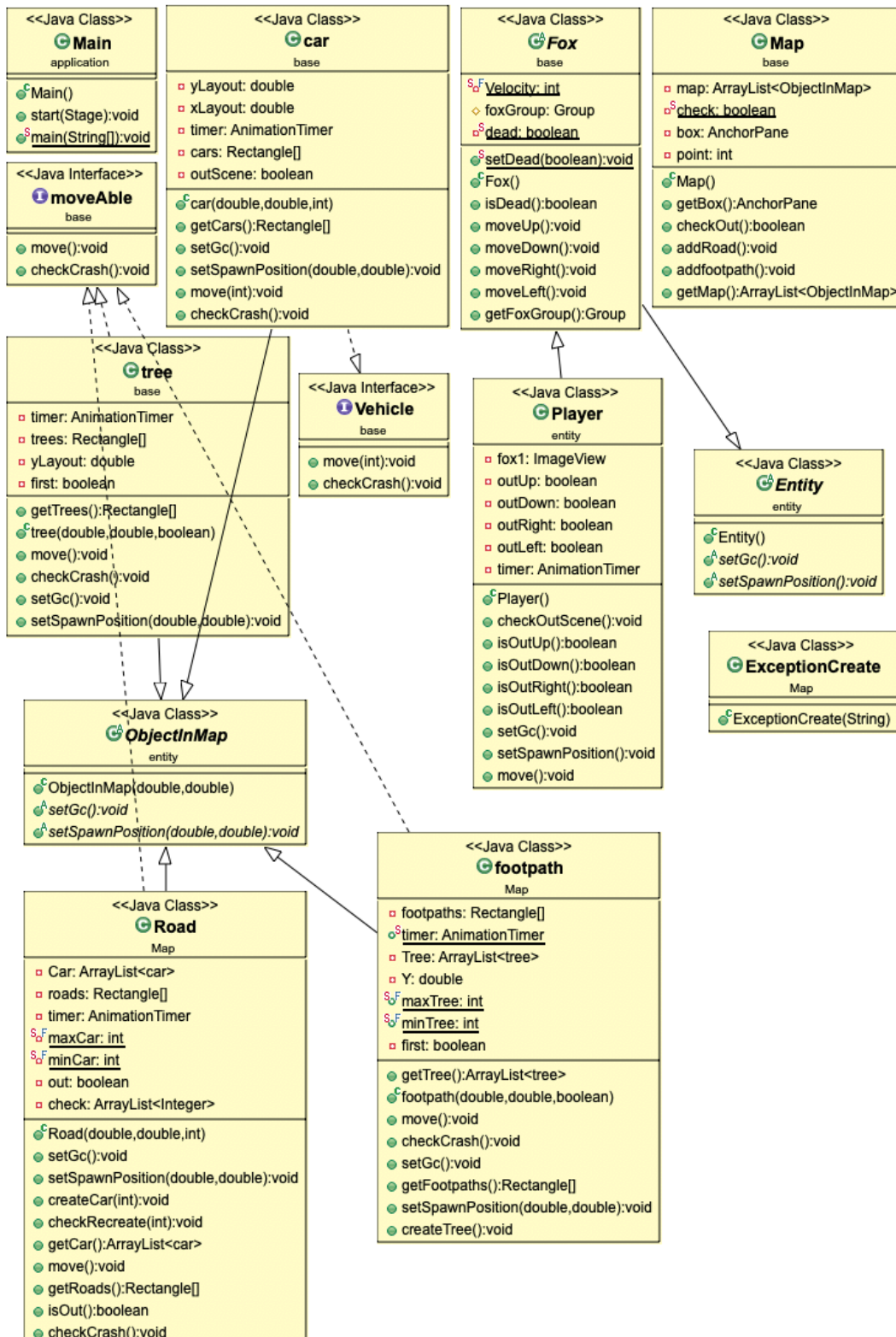


Obstacles



# Implementation Details

The UML diagram as shown:



<<Java Class>> <b>G mainMenu</b> gui2	
<ul style="list-style-type: none"> <li>playBtn: Buttons</li> <li>howToBtn: Buttons</li> <li>exitBtn: Buttons</li> <li>scenetitle1: Text</li> <li>scenetitle2: Text</li> <li>mainPane: AnchorPane</li> <li>buttonPane: VBox</li> <li><u>mainStage: Stage</u></li> <li><u>gameScene: Scene</u></li> <li>hT: howToPlay</li> </ul>	
<ul style="list-style-type: none"> <li>mainMenu()</li> <li>createMenu():void</li> <li>initializeMain():void</li> <li>initializeButton():void</li> <li>initializeTitle():void</li> <li>createBg():void</li> <li>getGameScene():Scene</li> <li>createBackgroundSound():void</li> <li><u>playBackgroundSound():void</u></li> <li><u>stopBackgroundSound():void</u></li> </ul>	

<<Java Class>> <b>G Buttons</b> gui2	
<ul style="list-style-type: none"> <li><u>BUTTON_HEIGHT: int</u></li> <li><u>BUTTON_WIDTH: int</u></li> </ul>	
<ul style="list-style-type: none"> <li>Buttons(String)</li> <li>setButtonsFont():void</li> <li>initializedButton():void</li> </ul>	

<<Java Class>> <b>GGameOver</b> gui2	
<ul style="list-style-type: none"> <li>playAgainBtn: Buttons</li> <li>mainMenuBtn: Buttons</li> <li>buttonPane: HBox</li> <li>mainPane: AnchorPane</li> <li><u>mainStage: Stage</u></li> <li><u>gameScene: Scene</u></li> <li>gameOver: ImageView</li> </ul>	
<ul style="list-style-type: none"> <li>GameOver()</li> <li>createMenu():void</li> <li>initializeMain():void</li> <li>initializeButton():void</li> <li>createBg():void</li> <li>getGameScene():Scene</li> <li>createHighestScoreLabel():void</li> <li>createCurrentScoreLabel():void</li> </ul>	

<<Java Class>> <b>GGame</b> gui2	
<ul style="list-style-type: none"> <li>gamePane: AnchorPane</li> <li>gameScene: Scene</li> <li><u>gameStage: Stage</u></li> <li><u>pointsLabel: PointsPane</u></li> <li><u>player: Player</u></li> <li>Timer: AnimationTimer</li> <li>Loop: AnimationTimer</li> <li>map: Map</li> <li>up: boolean</li> <li>down: boolean</li> <li>left: boolean</li> <li>right: boolean</li> </ul>	
<ul style="list-style-type: none"> <li>Game()</li> <li>playerMove():void</li> <li>newGame():void</li> <li>setSceneGame():void</li> <li>createLoop():void</li> <li>createKeyListener():void</li> <li>getGameScene():Scene</li> <li>Dead():void</li> </ul>	

<<Java Class>> <b>GGameSetting</b> gui2	
<ul style="list-style-type: none"> <li><u>Game_width: int</u></li> <li><u>Game_height: int</u></li> </ul>	
<ul style="list-style-type: none"> <li>GameSetting()</li> </ul>	

<<Java Class>> <b>GPointsPane</b> gui2	
<ul style="list-style-type: none"> <li><u>pointsLabel: Label</u></li> <li><u>points: int</u></li> <li><u>highestPoints: int</u></li> </ul>	
<ul style="list-style-type: none"> <li>PointsPane()</li> <li>setFont():void</li> <li><u>updateScore():void</u></li> <li><u>updateHighScore():void</u></li> <li><u>getPointsLabel():Label</u></li> <li><u>setPoints(int):void</u></li> </ul>	

<<Java Class>> <b>GhowToPlay</b> gui2	
<ul style="list-style-type: none"> <li>howToText: Text</li> <li>firstRules: Text</li> <li>secondRules: Text</li> <li>thirdRules: Text</li> <li>fourthRules: Text</li> <li>mainPane: AnchorPane</li> <li>textBox: VBox</li> <li>exitButt: Button</li> <li>gameScene: Scene</li> <li>subTextBox: HBox</li> <li>exit: boolean</li> </ul>	
<ul style="list-style-type: none"> <li>howToPlay()</li> <li>isExit():boolean</li> <li>initializeInfo():void</li> <li>createBg():void</li> <li>TextBox():void</li> <li>initializedButton():void</li> <li>getGameScene():Scene</li> <li>setCustomFont(Text):void</li> </ul>	

<<Java Class>> <b>GResloader</b> resloader	
<ul style="list-style-type: none"> <li><u>fox: ImageView</u></li> <li><u>gameOver: ImageView</u></li> <li><u>car: Image</u></li> <li><u>car2: Image</u></li> <li><u>road: Image</u></li> <li><u>exitButt: ImageView</u></li> <li><u>MAIN_BG: Image</u></li> <li><u>muteBtn: Image</u></li> <li><u>gameOver_BG: Image</u></li> <li><u>tree: Image</u></li> <li><u>footpath: Image</u></li> <li><u>backgroundGameSound: MediaPlayer</u></li> <li><u>jumpSound: AudioClip</u></li> </ul>	
<ul style="list-style-type: none"> <li>Resloader()</li> <li><u>load():void</u></li> </ul>	

## 1.Package application

### 1.1 Class Main

#### 1.1.1 Method

+ void start(Stage primaryStage)	Set primaryStage to mainStage Set Resizable false Show stage
+ void main(String[] args)	Entrance to the application

## 2. Package base

### 2.1 Class car extends Object implements Vehicle

#### 2.1.1 Field

- double xLayout	Xlayout of car.
- double yLayout	Ylayout of car.
- AnimationTimer timer	Timer for cars.
- Rectangle[] cars	Collect all rectangle for cars.
- boolean outScene	Check if car out of Scene.

#### 2.1.2 Constructor

+ car(double x,double y, int velo)	Initialize field.
------------------------------------	-------------------

#### 2.1.3 Method

+ Rectangle[] getCars()	Return cars.
-------------------------	--------------

+ void setSpawnPosition(double xAxis,double yAxis)	Set spawn position of cars.
+ void move()	Move cars in each animation timer cycle and check if out border.
+ void checkCrash()	Nothing.
+ boolean isOutScene	Return outScene.
+ void setGc()	Set graphic context for cars.

## 2.2 abstract class Fox extends Entity

### 2.2.1 Field

- static final int Velocity	Define Fox's velocity .
* Group foxGroup	Collect fox animation group.
- static boolean dead	Check if fox is dead or not.

### 2.2.2 Constructor

+ Fox()	Initialize field.
---------	-------------------

### 2.2.3 Method

+ void moveUp()	setLayoutY of fox by Layout-velocity.
+ void moveDown()	setLayoutY of fox by Layout+velocity.
+ void moveRight()	setLayoutX of fox by Layout+7.
+ void moveLeft()	setLayoutX of fox by Layout+7.
+ Group getFoxgroup()	Return foxGroup.



+ static void setDead(boolean dead)	Set is fox dead or not.
+ boolean isDead()	Return dead.

## 2.3 Class Map

### 2.3.1 Field

- ArrayList<ObjectInMap> map	Keep element in map.
- static boolean check	Check that elements which add to map are not out of layout.
- AnchorPane box	Pane that contain footpath and road.
- int point	Count total in each round.

### 2.3.2 Constructor

+ Map()	Initialize every field.  Generate every element in map and check is they are in a type of road or footpath.
---------	---

### 2.3.3 Method

+ AnchorPane getBox()	Return box.
+ boolean checkOut()	Check that map and footpath are not out of layout.
+ void addRoad()	Remove road that out of layout and add new one.

	Set point. Update score and high score.
+ void addfootpath()	Remove footpath that out of layout and add new one.  Set point.  Update score and high score.
+ ArrayList<ObjectInMap> getMap()	Return map.

## 2.4 Interface moveAble

### 2.4.1 Method

+ void move()	Movement of object.
+ void checkCrash()	Check is object crash.

## 2.5 Class tree extends ObjectInMap implements moveAble

### 2.5.1 Field

- AnimationTimer timer	Timer for tree.
- Rectangle[] trees	Collect all rectangle for tree.
- double yLayout	Position of tree in y axis.
- boolean first	Check that want to create trees or not.

### 2.5.2 Constructor

+ tree(double x,double y, boolean first)	Initialize every field.
--	-------------------------

### 2.5.3 Method

+ Rectangle[] getTrees()	Return tree.
+ void move()	Set spawn position of tree.
+ void checkCrash()	Nothing.
+ void setGc()	Set graphic context for tree.
+ void setSpawnPosition(double xAxis,double yAxis)	Set spawn position of tree.

## 2.4 Interface Vehicle

### 2.4.1 Method

+ void move(int velo)	Movement of object.
+ void checkCrash()	Check is object crash.

## 3. Package entity

### 3.1 Abstract class Entity

#### 3.1.1 Constructor

+ Entity()	Initialize methods.
------------	---------------------

#### 3.1.2 Method

+ abstract void setGc()	Set graphic context.
+ abstract void setSpawnPosition()	Set spawn location.

### 3.2 Abstract class ObjectInMap

### 3.1.1 Constructor

+ ObjectInMap (double xAxis,double yAxis)	Initialize methods.
---	---------------------

### 3.1.2 Method

+ abstract void setGc()	Set graphic context.
+ abstract void setSpawnPosition(double xAxis,double yAxis)	Set spawn location.

## 3.3 Class player extends Fox

### 3.3.1 Field

- ImageView fox1	Fox's image.
- boolean outUp	Check is fox go out of the scene.
- boolean outRight	Check is fox go out of the scene.
- boolean outLeft	Check is fox go out of the scene.
- boolean outDown	Check is fox go out of the scene.
- AnimationTimer timer	Timer for player.

### 3.3.2 Constructor

+ Player()	Initialize methods.
------------	---------------------

### 3.3.3 Methods

+ void checkOutScene()	Check if fox goes out the scene.
------------------------	----------------------------------

+ boolean isOutUp()	Return outUp.
+ boolean isOutDown()	Return outDown.
+ boolean isOutLeft	Return outLeft.
+ boolean isOutRight	Return outRight.
+ void setGc()	Set fox height = 50. Set fox width = 50.
+ void setSpawnPosition	Set layoutX and layoutY.
+ void move()	Set movement between fox and background.

## 4.Package gui2

### 4.1 Class Buttons extends Button

#### 4.1.1 Field

- static final int BUTTON_HEIGHT	Define button height
- static final int BUTTON_WIDTH	Define button width

#### 4.1.2 Constructor

+ Buttons(String text)	Initialize methods. Set text fill by using Color.WHITESMOKE. Set button background color.
------------------------	--

#### 4.1.3 Method

- void setButtonsFont()	Set button font.
- void initializeButton()	Initialize button.

## 4.2 Class Game

### 4.2.1 Field

- AnchorPane gamePane	Game's pane.
- Scene gameScene	Game's scene.
+ static Stage gameStage	Game's stage.
+ static PointsPane pointsLabel	Point pane.
+ static Player player	Player.
- AnimationTimer Timer	Timer for movement in game.
- AnimationTimer Loop	Loop for generate map.
- Map map	Map.
- boolean up	Check is it up.
- boolean down	Check is it down.
- boolean left	Check is it left.
- boolean right	Check is it right.

### 4.2.2 Constructor

+ Game()	Initialize every field and method.
----------	------------------------------------

### 4.2.3 Method

+ void playerMove()	Loop check that is player is not move
---------------------	---------------------------------------

	outside layout.
+ void newGame()	Initialize play , map , pointslabel and add them to gamepane. Play music.
+ void setSceneGame()	Set scene.
+ void createLoop()	Create map if it is road then generate car if not generate tree.
+ void createKeyListener()	Set keypress. Set jump sound.
+ Scene getGameScene()	Return gameScene.
+void Dead()	Stop game and go to gameoverPane. Stop music.

### 4.3 Class GameOver

#### 4.3.1 Field

- Buttons playAgainBtn	Define play again button.
- Buttons mainMenuBtn	Define main menu button.
- HBox buttonPane	Collect every button.
- AnchorPane mainPane	Define main pane.
+ static Stage mainStage	Define main stage.
+ static Scene gameScene	Define game scene.
- ImageView gameOver	Gameover picture.

#### 4.3.2 Constructor

+ GameOver()	Initialize method.  Initialize stage.  Set title,scene,resizeable.
--------------	--

#### 4.3.3 Method

+ void createMenu()	Initialize pane.  Initialize every element in pane.
- void initializeMain()	Set on action every button.
- void initializeButton()	Initialize play again and main menu button and add them to it's pane.
- void createBg()	Set mainpane background.
+ Scene getGameScene()	Return gameScene.
- void createHighestScoreLabel()	Create highscore label and set style.
- void createCurrentScoreLabel()	Create current score label and set style.

### 4.4 Class GameSetting

#### 4.4.1 Field

+ static final int Game_width	Define game width.
+ static final int Game_height	Define game height.

### 4.5 Class howToPlay

#### 4.5.1 Field



- Text howToText	How to play text.
- Text firstRules	First rule.
- Text secondRules	Second rule.
- Text thirdRules	Third rule.
- Text fourthRules	Fourth rule.
- AnchorPane mainPane	Main pane.
- VBox textBox	Collect every text.
- Button exitButt	Exit button.
- Scene gameScene	Game scene.
- HBox subTextBox	Sub text box.
- boolean exit	Check if it is exit.

#### 4.5.2 Constructor

+ howToPlay()	Initialize every field,method.
---------------	--------------------------------

#### 4.5.3 Method

+ boolean isExit()	Return exit.
+ void initializeInfo()	Initialize every text and add them to text box.
+ void createBg()	Set background image.
+ void TextBox()	Initialize and set text box.
+ void initializeButton()	Create exit button and set it.
+ Scene getGameScene()	Return gameScene.
+ void setCustomFont(Text x)	Set custom font.

## 4.6 Class mainMenu

### 4.6.1 Field

- Buttons playBtn	Play button.
- Buttons howToBtn	How to play button.
- Buttons exitBtn	Exit button.
-Text sceneTitle1	Scene title 1.
- Text sceneTitle2	Scene title 2.
- AnchorPane mainPane	Main pane.
- VBox buttonPane	Button pane.
- SoundButton soundBtn	Sound button.
+ static Stage mainStage	Main stage.
+ static Scene gameScene	Game scene.
- howToPlay hT	How to play object.

### 4.6.2 Constructor

+ mainMenu()	Initialize field and method.  Set scene,title  Set resizable = false
--------------	--

### 4.6.3 Method

+ void createMenu()	Initialize mainPane and method in menu.
+ void initailizeMain()	Set action in every button.
- void initializeButton()	Initialize play, howTo, exit button and

	add them to buttonPane.
- void initailizeTitle()	Initialize and set style title.
- void createBg()	Set background.
+ Scene getGameScene()	Return gameScene.
+ void createBackgroundSound()	Create background sound.
+ static void playBackgroundSound()	Play sound.
+ static void stopBackgroundSound()	stop sound.

## 4.7 Class PointsPane

### 4.7.1 Field

- static Label pointsLabel	Point label.
+ static int points	Point.
+ static int highestPoints	Highest point.

### 4.7.2 Constructor

+ PointsPane()	Create points label, set size, set style, set alignment, set padding and add text to label  Set points to 0  Set font.
----------------	--

### 4.7.3 Method

- void setFont()	Set font
------------------	----------

+ void updateScore()	Update text to current points
+ void updateHighScore()	Update highest points
+ Label getPointsLabel()	Getter for points label
+ void setPoints(int points)	Setter for points

## 5.Package Map

### 5.1 Class footpath extends ObjectInMap implements moveAble

#### 5.1.1 Field

- Rectangle[] footpaths	Collect all rectangle for footpath.
+ static AnimationTimer timer	Timer for move.
- ArrayList<tree> Tree	Keep every tree in one footpath.
- double Y	Y axis.
+static final int maxTree = 4	Max tree
+static final int minTree = 1	Min tree
- boolean first	Check that want to create trees or not.

#### 5.1.2 Constructor

+ footpath(double xAxis,double yAxis,boolean first)	Initialize field and method.
---	------------------------------

#### 5.1.3 Method

+ ArrayList<tree> getTree()	Return Tree.
+ void move()	Timer for move footpath.

+ void checkCrash()	Nothing.
+ void setGc()	Set graphic context for footpath.
+ Rectangle[] getFootpaths()	Return footpaths.
+ void setSpawnPosition(double xAxis,double yAxis)	Set spawn location.
+ void createTree()	Random amount of tree in each footpath and add them to list.

## 5.2 Class Road extends ObjectInMap implements moveAble

### 5.2.1 Field

- Rectangle[] roads	Collect all rectangle for road.
- boolean out	Is car out of road.
- ArrayList<car> Car	Keep every car in one road.
- AnimationTimer timer	Y axis.
+static final int maxCar = 15	Max tree
+static final int minCar = 10	Min tree
- ArrayList<Integer> check	Keep car x layout.

### 5.2.2 Constructor

+ Road(double xAxis,double yAxis,int velo)	Initialize field and method.
--	------------------------------

### 5.2.3 Method

+ ArrayList<car> getCar()	Return Car.
---------------------------	-------------

+ void move()	Timer for move road.
+ void checkCrash()	Nothing.
+ void setGc()	Set graphic context for road.
+ Rectangle[] getRoads()	Return footpaths.
+ void setSpawnPosition(double xAxis,double yAxis)	Set spawn location.
+ void createCar()	Random amount of car in each road and add them to list.
+ boolean isOut()	Return out.

### 5.3 Class ExceptionCreate extends Exception

#### 5.3.1 Method

+ ExceptionCreate(String e)	Throws exception
-----------------------------	------------------

## 6.Package resloader

### 6.1 Class Resloader

#### 6.1.1 Field

+ static ImageView fox	Fox image
+ static ImageView gameOver	Gameover text image
+ static ImageView exitButt	Exit button image
+ static Image car	Car1 image
+ static Image car2	Car2 image
+ static Image road	Road image

+ static Image MAIN_BG	Main background image
+ static Image muteBtn	Mute button background image
+ static Image gameOver_BG	Game over background image
+ static Image tree	Tree image
+ static Image footpath	Footpath image
+ static MediaPlayer backgroundGameSound	Background sound

### 6.1.2 Method

+ void Load()	Load all resources
---------------	--------------------