

Project Document : Animal Rescue

Pollawat Hongwimol 6030400021

Peeranuth Kehasukcharoen 6030416021

1. Introduction

ในปี คศ. 3019 บ้านเมืองเจริญขึ้น พื้นที่ป่าลดน้อยลง ประกอบด้วยมลพิษของเสียที่มนุษย์ทิ้งออกมา ทำให้สัตว์ป่าล้มตายลงไปจนเกือบหมด รัฐบาลในสมัยนั้นได้วางแผนการขนย้ายสัตว์ป่าที่ใกล้สูญพันธุ์จากในป่าเข้าไปในสวนสัตว์ที่เตรียมไว้โดยใช้เครื่องบินขนาดใหญ่ ในขณะที่กำลังขนย้ายได้มีกลุ่มโจรมาปล้น ซึ่งโจรกลุ่มนั้นต้องการนำสัตว์ป่าเหล่านั้นไปขาย เพราะชิ้นส่วนของมันแต่ละชิ้นเป็นของหายากและมีราคาแพง โจรได้ทำการปล่อยตอนเท่นเนอร์สัตว์ทั้งหมดลงจากเครื่องบินขนส่ง และจึงโดดร่มลงมาตาม ผู้เล่นจะได้รับบทเป็นพลทหารที่มีทักษะในการยิงปืนที่แม่นยำเป็นอย่างมาก มีภารกิจรับผิดชอบการขนย้ายสัตว์ป่าครั้งนี้ และคอยเฝ้าดูเหตุการณ์โจรกรรมครั้งนี้มาโดยตลอด ผู้เล่นมีหน้าที่กำจัดโจร และช่วยเหลือสัตว์ป่า ซึ่งเป็นภารกิจที่จะต้องทำให้สำเร็จ เพื่อรักษาสัตว์ป่าเหล่านั้นไว้ไม่ให้สูญพันธุ์ไป

ANIMAL RESCUE

ภาพที่ 1: Logo ของเกม Animal Rescue

เนื่องจากเกมนี้จำลองบทบาทการช่วยเหลือสัตว์จากการถูกโจรกรรมที่กล่าวมาข้างต้น เกมนี้จึงถูกตั้งชื่อว่า “Animal Rescue” และถูกออกแบบมาให้เป็นเกมที่มีรูปแบบ 8 บิต (8 bit) เพื่อให้คล้ายคลึงกับเกมส่วนใหญ่ที่ถูกสร้างมาในยุคแรก ๆ สัญลักษณ์ของเกมนี้จึงถูกสร้างออกมาในรูปแบบ 8 บิตด้วยเช่นกัน ดังภาพที่ 1 นอกเหนือจากนี้ ตัวอักษร “A” ตัวแรกของชื่อยังถูกนำมาตั้งเป็น Icon ของเกมอีกด้วย

2. How to play

ให้ผู้เล่นกดปุ่มบนคีย์บอร์ดโดยไม่สนใจตัวอักษรใหญ่-เล็ก (case-insensitive) เพื่อกำจัดวัตถุ (unit) ที่ตกลงมาจากบนฟ้า (ภาพที่ 2) ประกอบไปด้วย นายพราน (hunter) และ สัตว์ (animal) โดยวัตถุที่ตกลงมาจะมีตัวอักษรกำกับอยู่บนร่มพาราชูต (parachute) ซึ่งเมื่อผู้เล่นกำจัดนายพราน ผู้เล่นจะได้รับคะแนน (score) แต่ถ้าผู้เล่นกำจัดสัตว์ ผู้เล่นจะถูกหักคะแนน และผู้เล่นไม่สามารถกำจัดสัตว์มากเกินไปกว่าที่กำหนดได้ โดยมีรายละเอียดดังนี้

- เมื่อกำจัดนายพรานสำเร็จ ได้รับคะแนน 8 คะแนน
- เมื่อกำจัดสัตว์ จะถูกหักคะแนน 5 คะแนน
- เมื่อกำจัดสัตว์ครบ 5 ครั้ง จะถือว่าภารกิจล้มเหลว และจบเกม
- เมื่อนายพรานลงถึงพื้น จะถือว่าภารกิจล้มเหลว และจบเกม



ภาพที่ 2: units ประกอบไปदनายพราน (ซ้าย) และสัตว์(ขวา)



ภาพที่ 3: มุมด้านขวาบนสุดของหน้าจอในขณะที่เล่นเกม

อย่างไรก็ตาม นาฬิกาจับเวลา (timer) จะแสดงอยู่ด้านบนขวาของหน้าจอ พร้อมกับคะแนน และ จำนวนชีวิต (life points) จำนวน 5 ชีวิตในรูปของหัวใจจำนวน 5 ดวงตามลำดับ (ดูภาพที่ 3 ประกอบ) ซึ่งเมื่อผู้เล่นกำจัดสัตว์ จำนวนของหัวใจจะลดลงตามจำนวนครั้ง ผู้เล่นจะยังสามารถเล่นเกมได้ตอถึงแม้ว่าจะไม่มีจำนวนชีวิตเหลืออยู่ก็ตาม แต่ถ้าผู้เล่นกดปุ่มพลาดหรือตั้งใจเพื่อกำจัดสัตว์ในครั้งถัดไป เกมจะจบลงทันที

3. Application flow

เมื่อเข้าสู่ตัวเกม หน้าจอจะแสดงหน้าจอเกมขึ้นมาในลักษณะดังภาพที่ 4 ซึ่งประกอบไปด้วยสัญลักษณ์ (logo) ของเกม ปุ่มเริ่มเล่นเกม (play) ปุ่มเครดิต (credit) ปุ่มตั้งค่า (option) ปุ่มออกจากเกม (exit) และภาพพื้นหลัง (background) ซึ่งทั้งหมดถูกออกแบบมาให้อยู่ในรูปแบบ 8 bit



ภาพที่ 4: หน้าจอหลักเมื่อเริ่มต้นเข้าสู่ตัวเกม

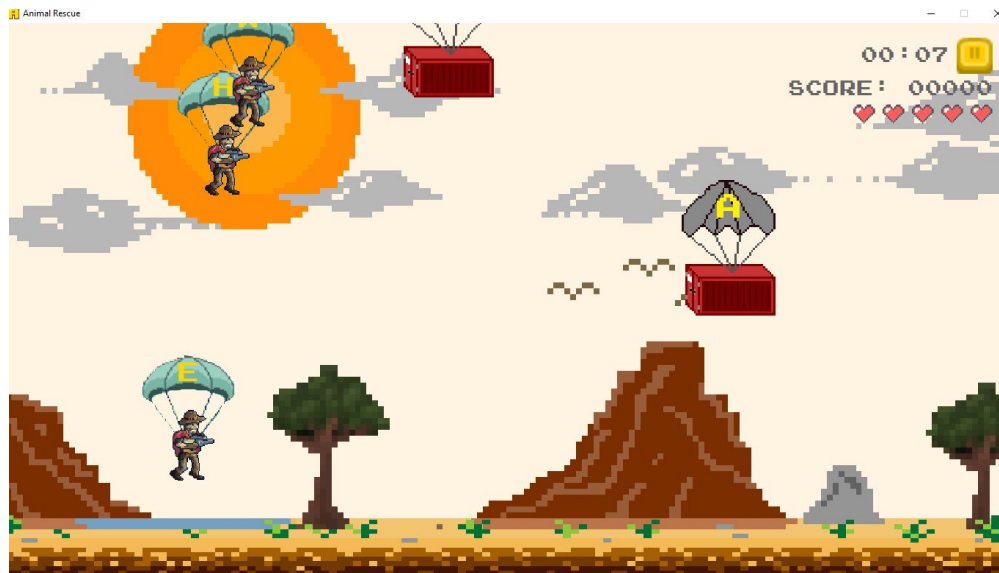
หลังจากเข้าสู่ตัวเกม การเล่นเกมนี้จะทำได้โดยการกดปุ่มเริ่มเล่นเกม จากนั้นจะมีหน้าต่างย่อย (subscene) เลื่อนเข้ามาจากทางด้านซ้ายดังภาพที่ 5 ซึ่งเป็นการแนะนำวิธีการเล่นเกม จากนั้นกดปุ่ม OK เพื่อทำการเริ่มเล่นเกม



ภาพที่ 5: หน้าต่างย่อยหลังจากกดปุ่มเริ่มเล่นเกม (play)

ในหน้าต่างย่อยของปู้มเริ่มเล่นเกมนั้นจะประกอบไปด้วยคำแนะนำในการเล่นเกควรรกดปู้มเพื่อกำจัดนายพราน และไม่ควรรกดปู้มเพื่อกำจัดสัตว์ อีกทั้งยังอธิบายคร่าว ๆ เกี่ยวกับจำนวนชีวิต ซึ่งเป็นฟังก์ชันหนึ่งในเกมอีกด้วย

หลังจากกดปู้ม OK ในหน้าต่างย่อยของปู้มเริ่มเล่นเกม เกมจะสลับหน้าจอมายังหน้าจอเล่นเกมดังที่แสดงให้เห็นดังภาพที่ 6 ซึ่งผู้เล่นจะสามารถเล่นเกมและเพลิดเพลินไปกับเกมอันสนุกสนานได้ที่หน้าจอนี้ ยิ่งไปกว่านั้น ผู้เล่นสามารถหยุดชั่วคราว (pause) และเล่นเกมต่อจากเดิม (unpause) ได้ที่ปู้มทางด้านขวาบน ถัดจากนาฬิกาจับเวลา



ภาพที่ 6: หน้าจอระหว่างการเล่นเกม (game stage)

ในส่วนของการพักชั่วคราว เมื่อผู้เล่นกดปู้มพักชั่วคราว เกมจะแสดงหน้าต่างย่อย ซึ่งภายในหน้าต่างย่อยจะมีปู้มเล่นใหม่อีกครั้ง (replay) ปู้มเมนู (menu) ซึ่งหมายถึงกลับไปยังหน้าจอดังที่แสดงในภาพที่ 4 ปู้มออกจากเกม (exit) และแถบปรับระดับความดังเสียง (volume) ดังภาพที่ 7



ภาพที่ 7: หน้าต่างย่อยหลังจากกดปู้มพักชั่วคราว (pause)

ภาพที่ 8 แสดงถึงหน้าจอเกมหลังจากเข้าสู่ตัวเกมและกดปุ่มเครดิต ซึ่งแสดงรูปภาพของผู้จัดทำทั้งสองคน ชื่อ-นามสกุล และรหัสประจำตัวนิสิต พร้อมข้อความติดตกทางด้านหลังว่า Withdrawn ในลักษณะเอียง ๆ เพื่อให้สอดคล้องกับชื่อทีม (Withdrawers) และเป็นส่วนหนึ่งในการนำชื่อทีมที่ผู้จัดทำได้ใช้เวลาค่อนข้างมากในการรังสรรค์ออกมา นำมาใส่ลงในตัวเกม



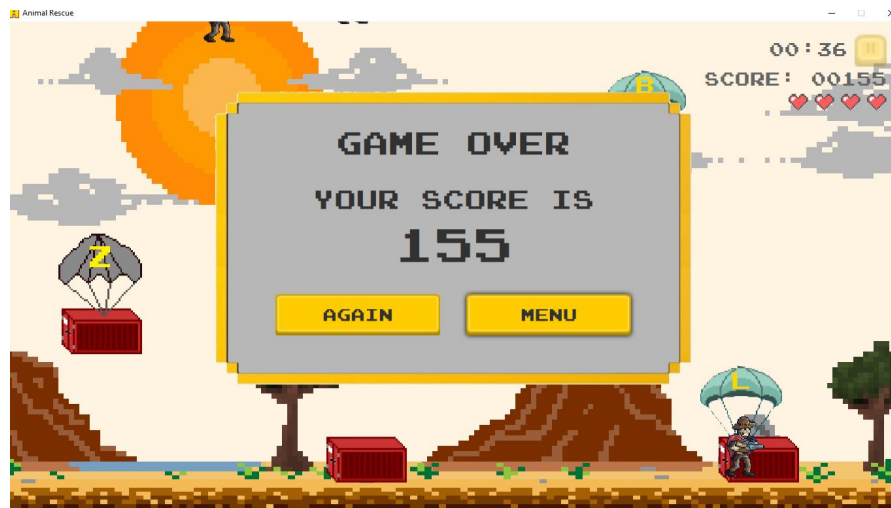
ภาพที่ 8: หน้าจอหลังจากกดปุ่มเครดิต (credit)



ภาพที่ 9: หน้าต่างย่อยหลังจากกดปุ่ม Option

ในส่วนของหน้าต่างย่อยของปุ่ม Option นั้น จะมีแถบที่สามารถเลื่อนได้ (slider bar) ดังภาพที่ 9 ซึ่งสามารถใช้ปรับระดับความดังของเสียงเพลงพื้นหลังภายในเกม รวมถึงเสียงเอฟเฟกต์ (effect sound) ภายในเกมได้ ซึ่งระดับความดังจะเริ่มต้นที่ 50 มีค่าน้อยที่สุด (minimum) เท่ากับ 0 และค่ามากที่สุด (maximum) เท่ากับ 100

หลังจากที่ผู้เล่นกดปุ่มเริ่มเล่นเกม ผู้เล่นจะเข้าสู่หน้าจอเล่นเกมดังที่แสดงให้เห็นในภาพที่ 6 จากนั้นผู้เล่นมีหน้าที่กดปุ่มเพื่อทำลายสิ่งๆที่ตกลงมาจากด้านบนของตัวเกม ดังข้อกำหนดใน [2. How to play](#) หลังจากนั้น เมื่อนายพรานลงถึงพื้นหรือจำนวนชีวิตเหลือต่ำกว่า 0 เกมจะจบลง และจะแสดงหน้าจอสรุปคะแนนดังภาพที่ 10 ซึ่งประกอบไปด้วยข้อความ “Game Over” และคะแนนที่สามารถทำได้ เป็นตัวเลขตามลำดับ ในบรรทัดถัดลงมา จะมีปุ่มเล่นเกมอีกครั้ง (again) และปุ่มกลับไปหน้าจอเมนู (menu) แสดงอยู่ เพื่อให้ผู้เล่นสามารถเลือกได้ว่าจะเล่นต่อหรือไม่



ภาพที่10: หน้าจอจบเกม (game over)

ไดอแกรม (diagram) ของโปรแกรมนี้แสดงให้เห็นดังภาพที่ 11 ด้านล่าง ประกอบไปด้วย 5 แพคเกจ (package) คือ application, draw, Exception, Input, logic และ sharedObject ซึ่งนับจำนวนคลาส (class) รวมกันทั้งหมดได้ 25 คลาส



- + (**public**)
(**protected**)
- (**private**)
static will be underlined.
abstract will be italic.

4.1 Class application. ButtonGenerator extends Button

4.1.1 Fields

- double BUTTON_WIDTH	ความกว้างของปุ่ม
- double BUTTON_HEIGHT	ความสูงของปุ่ม
- double FONT_SIZE	ขนาดของฟอนต์ด้านในปุ่ม
- String NORM_BUTTON_PATH	String url path ที่ระบุถึงภาพ Background ปุ่ม
- String PRESSED_BUTTON_PATH	String url path ที่ระบุถึงภาพ Background ปุ่ม
- String BUTTON_NORM_STYLE	String ที่ใช้ตั้ง style ของปุ่มตอนปกติ
- String BUTTON_ONPRESS_STYLE	String ที่ใช้ตั้ง style ของปุ่มตอนโดนกด

4.1.2 Constructor

+ ButtonGenerator(String name)	รับ parameter เป็น String Constructor มีหน้าที่ <ul style="list-style-type: none"> - ตั้งค่าด้านในปุ่ม ตาม parameter - ตั้งค่าขนาดปุ่มโดย setSize() - ตั้งค่า style เริ่มต้นของปุ่ม โดย setNormStyle() - ตั้งค่าการรับ input ต่างๆจาก mouse โดย setActionFromMouse()
--------------------------------	---

4.1.3 Methods

- setSize()	Method นี้มีหน้าที่ Set ค่าขนาดของปุ่มตามค่า BUTTON_WIDTH และ BUTTON_HEIGHT
- setNormStyle()	Method นี้มีหน้าที่ตั้งค่า Style ตาม BUTTON_NORM_STYLE และ เลื่อนตำแหน่งของปุ่มลงไป 7 และเปลี่ยนความสูงของปุ่มเป็น 63
- setPressedStyle()	Method นี้มีหน้าที่ตั้งค่า Style ตาม BUTTON_ONPRESS_STYLE และ เลื่อนตำแหน่งของปุ่มขึ้นไป 7 และเปลี่ยนความสูงของปุ่มเป็น 56
- setActionFromMouse()	Method นี้จะตั้งค่าการรับ input จากเมาส์ ดังนี้ <ul style="list-style-type: none"> - เมื่อเมาส์เลื่อนเข้ามาวางบนปุ่ม : สร้าง effect เจา และ เล่นเสียง enterd จาก class MediaManager - เมื่อเมาส์เลื่อนออกจากปุ่ม : หยุด effect เจา

	<ul style="list-style-type: none">- เมื่อกดเมาส์ : ถ้าเป็นการคลิกซ้ายให้ตั้ง style ปุ่มเป็นตอนโดนกด และ เล่นเสียง clicked จาก MediaManager- เมื่อปล่อยเมาส์ : ถ้าเป็นการปล่อยคลิกซ้ายให้ตั้ง style ปุ่มเป็นตอนปกติ
--	---

4.2 Class application. GameManager

4.2.1 Fields

- Stage mainStage	ค่าตัวแปรสำหรับ mainStage ของ game
- Scene game	ค่าตัวแปรสำหรับ Scene ของเกม
- AnchorPane root	Root ของ Scene
- GameTimer gameTimer	ค่าตัวแปร GameTimer
- HBox timerBox	Pane สำหรับแสดงเวลา
- Life life	ตัวแปร Life สำหรับแสดงชีวิตที่เหลือ
- PauseButton pauseButton	ตัวแปร PauseButton สำหรับปุ่ม pause
- ScoreCount scoreCount	ตัวแปร ScoreCount สำหรับแสดงคะแนน
- PauseSubscene pauseSubscene	ตัวแปร PauseSubScene สำหรับหน้า SubScene เวลา pause game
- AnimationTimer anime	ตัวแปร AnimationTimer สำหรับ Animation และ flow ของเกม
- GameScreen gScreen	ตัวแปร GameScreen สำหรับวาด graphic ต่างๆ
- GameLogic gLogic	ตัวแปร GameLogic สำหรับ flow ของเกม
- <u>int animalPrevSec</u>	ตัวแปรเก็บค่าวินาทีครั้งที่แล้วที่สร้างสัตว์
- <u>int hunterPrevSec</u>	ตัวแปรเก็บค่าวินาทีครั้งที่แล้วที่สร้างนายพราน
- <u>int accelPrevSec</u>	ตัวแปรเก็บค่าวินาทีที่แล้วที่เพิ่มความเร็วของ นายพราน และสัตว์
- <u>boolean isGameOver</u>	ตัวแปรที่เก็บค่าสถานะ game over
+ <u>boolean isPause</u>	ตัวแปรที่เก็บค่าสถานะการ pause game
- <u>String CURSOR_PATH</u>	String url Path ที่ระบุถึงภาพสำหรับ cursor

4.2.2 Constructor

+ GameManager()	Constructor มีหน้าที่ดังนี้ <ul style="list-style-type: none"> - Initialize Scene, Stage GameScreen, GameLogic โดยให้ขนาดเป็น 1366*768
-----------------	---

	<ul style="list-style-type: none"> - Set title เป็น “Animal Rescue” - Set icon เกม - สร้าง GameTimer โดย createTimer() - สร้าง PauseButton โดย createPauseButton() - สร้าง ScoreCount โดย createScoreCount() - สร้าง PauseSubScene โดย createSubScene() - สร้าง Life โดย createLife() - Set Style ของ cursor - Set การรับ input ต่างๆ โดย setKeyPress() - เริ่ม Flow เกม
--	--

4.2.3 Methods

+ Stage getMainStage()	mainStage's getter
- void createLife()	Initialize life และ ตั้งตำแหน่งของ life เป็นที่ <ul style="list-style-type: none"> - 30 ของ Anchor ขวา ของ Root - 110 ของ Anchor บน ของ Root
- void createPauseButton()	Initialize pauseButton และ ตั้งตำแหน่งของ pauseButton เป็นที่ <ul style="list-style-type: none"> - 30 ของ Anchor ขวา ของ Root - 20 ของ Anchor บน ของ Root
- void createTimer()	Initialize gameTimer และ timerBox โดยรับค่ามาจาก gameTimer ตั้งตำแหน่งของ timerBox เป็นที่ <ul style="list-style-type: none"> - 90 ของ Anchor ขวา ของ Root - 25 ของ Anchor บน ของ Root
- void createScoreCount()	Initialize scoreCount และ ตั้งตำแหน่งของ scoreCount เป็นที่ <ul style="list-style-type: none"> - 30 ของ Anchor ขวา ของ Root - 70 ของ Anchor บน ของ Root
- void createSubscene()	Initialize pauseSubScene และ <ul style="list-style-type: none"> - สร้างปุ่ม replay โดย ButtonGenerator โดย ตั้งให้เมื่อโดนกดจะซ่อน subScene และ เริ่มเกมใหม่ - สร้างปุ่ม menu โดย ButtonGenerator โดยตั้งให้เมื่อโดนกดจะแสดงหน้า Stage Menu ขึ้นมาและปิด Stage ของเกม - สร้างปุ่ม exit โดย ButtonGenerator โดย

	<p>เมื่อกดให้ปิด application พื้นที่</p> <ul style="list-style-type: none"> - เก็บปุ่มทั้ง 3 ปุ่มใน VBox ที่มีขนาด spacing เป็น 10 และวางไว้ที่ ตำแหน่งแกน x เป็น 400 แกน y เป็น 80 - สร้าง SliderBar แล้ววางที่ ตำแหน่งแกน x เป็น 120 แกน y เป็น 350 - สร้าง LabelGenerator คำว่า "PAUSE" โดยมีขนาด 50 และตั้งที่ ตำแหน่งแกน x เป็น 60 แกน y เป็น 80
- void showSubscene()	Show pauseSubScene โดยมี transition effect เลื่อนเข้ามาโดย method transitionIn() ของ pauseSubScene
- void hideSubscene()	ซ่อน pauseSubScene
- void createGameplay()	เริ่ม AnimationTimer เพื่อสร้าง flow ของ game เพื่อ update ค่าต่างๆ
- void updateEntities(int sec)	รับ parameter เวลาเป็นวินาทีเข้ามาเพื่อ <ul style="list-style-type: none"> - สร้างสัตว์ โดย createAnimal(int sec) - สร้างนายพราน โดย createHunters(int sec) - ถ้า เวลาหารด้วย 25 ลงตัว และไม่เท่ากับ 0 ให้เพิ่มอัตราการสร้าง FallableUnit - ถ้าเวลาเท่ากับ 59 ให้ตั้งค่า animalPrevSec, hunterPrevSec, accelPrevSec เป็น 0
- void createAnimal(int sec)	สร้างสัตว์ทุก 3 วินาที โดยสร้างจำนวนตามอัตราการสร้างของ FallableUnit
- void createHunters(int sec)	สร้างนายพรานทุก 2 วินาที โดยสร้างจำนวนตามอัตราการสร้างของ FallableUnit
- void accelerate(in sec)	เพิ่มความเร็วของ FallableUnit ทุก 5 วินาที
- void setKeyPress()	Set ค่าการรับ input ดังนี้ <ul style="list-style-type: none"> - มีการกดคีย์บอร์ด : ถ้าไม่ได้กดค้าง ให้ set triggered ใน IOManager เป็น true - ปลดคีย์บอร์ด : set ค่า pressed ของ IOManager เป็น false
- void pause()	Pause game หยุด flow เกมทุกอย่าง แล้วโชว์ pauseSubScene
- void unpause()	Resume game กลับมาดำเนิน flow เกมทุกอย่างต่อ

- void start()	เริ่ม flow game โดย method createGamePlay()
- void replay()	เริ่มเกมใหม่โดยตั้งค่าตัวแปรต่างๆที่เกี่ยวข้องกับ flow เกม กลับเป็นค่าเริ่มต้นทั้งหมด
+ boolean getGameOver()	isGameOver's getter
+ boolean setGameOver()	isGameOver's setter
- void gameOver()	<p>Method สำหรับเมื่อ game over โดยจะหยุด flow game ทั้งหมด และ disable ปุ่ม pause และแสดง หน้า SubScene game over</p> <p>โดยจะ Initialize PauseSubScene ใช้เป็น subSceneสำหรับหน้า game over โดยมีองค์ประกอบดังนี้</p> <ul style="list-style-type: none"> - LabelGenerator คำว่า "GAMEOVER" ขนาด 48 ตั้งที่ ตำแหน่ง แกน y เป็น 49 มี Alignment เป็น Pos.CENTER มีความกว้างเท่ากับความกว้าง SubScene - LabelGenerator คำว่า "Your Score Is" ขนาด 40 ตั้งที่ ตำแหน่ง แกน y เป็น 140 มี Alignment เป็น Pos.CENTER มีความกว้างเท่ากับความกว้าง SubScene - LabelGenerator แสดงคะแนนขนาด 72 ตั้งที่ ตำแหน่ง แกน y เป็น 189 มี Alignment เป็น Pos.CENTER มีความกว้างเท่ากับความกว้าง SubScene - ButtonGenerator ปุ่ม "Again" ตั้งที่ ตำแหน่งแกน x เป็น ความกว้างของ SubScene หาร 2 แล้วลบด้วย 272 แกน y เป็น 310 โดยตั้งให้เมื่อโดนกดปุ่มจะเริ่มเล่นเกม ใหม่ - ButtonGenerator ปุ่ม "Menu" ตั้งที่ ตำแหน่งแกน x เป็น ความกว้างของ SubScene หาร 2 แล้วบวกด้วย 20 แกน y เป็น 310 โดยตั้งให้เมื่อโดนกดปุ่มจะออกจาก Stage game และแสดง Stage menu
- void customCursor()	ตั้งค่าของ cursor ตามรูปจาก CURSOR_PATH

4.3 Class application. GameTimer

4.3.1 Fields

- HBox box	HBox สำหรับแสดงค่าเวลา
- Thread t	Thread สำหรับการนับเวลา
- Label colon	Label สำหรับ “:”
- int counter	ตัวแปรสำหรับนับเวลา
- <u>Label minute</u>	ตัวแปรสำหรับแสดงเวลาในหน่วยนาที
- <u>Label second</u>	ตัวแปรสำหรับแสดงตัวแปรในหน่วยวินาที
- String TIME_TEXT_STYLE	String สำหรับใช้แต่ง style ของ Label ที่แสดง

4.3.2 Constructor

+ GameTimer()	Initialize Hbox และ Label ทั้งหมด โดยใช้ method insertTimeBox()
---------------	---

4.3.3 Methods

+ void insertTimeBox()	Initialize Hbox ที่มี spacing เป็น 1.4 Label ต่างๆ โดยตั้งค่าเริ่มต้นเป็น “00 : 00” และตั้งค่า style ของ Label ทั้งหมดโดย setAllStyle()
+ void setAllStyle()	ตั้งค่า style ของ Label ทั้งหมดตาม TIME_TEXT_STYLE
+ HBox getTimerBox()	Getter ของ box
+ void start()	Initialize thread สำหรับนับเวลา โดยใช้ counter นับวินาที แล้วแสดงค่าเวลาโดยการเปลี่ยนตัวเลขที่ Label ที่ใช้แสดงเวลา
+ void pause()	หยุดการทำงานของ thread
+ void unpause()	Resume การทำงานของ thread
+ void reset()	Reset ค่าเวลากลับไปเป็นเวลาเริ่มต้น
+ <u>int getSecond()</u>	คืนเวลาที่ผ่านไปหน่วยวินาที

4.4 Class application. LabelGenerator extends Label

4.4.1 Constructor

+ LabelGenerator(String text, int size)	<p>รับ parameter เป็น String และ int และทำการตั้งค่าดังนี้</p> <ul style="list-style-type: none">- Set text ตาม String ที่รับเข้ามา- Warp text- ตั้ง Position เป็น TOP_CENTER- Set font ของ text เป็น Joystix Monospace และมีขนาดตามตัวแปร int ที่รับเข้ามา
---	--

4.5 Class application. Life extends HBox

4.5.1 Fields

+ <u>int</u> life	จำนวนชีวิต
- String LIFE_PATH	String ของ url ที่ที่ระบุถึงไฟล์ภาพของรูป life
- Thread thread	Thread สำหรับคอยเช็ค status ของ life

4.5.2 Constructor

+ Life()	Initialize โดย <ul style="list-style-type: none"> - ตั้ง spacing เป็น 10 - Start thread สำหรับ check status ของ life ทุก 20 ms โดยถ้าชีวิตเหลือต่ำกว่า 0 ให้ game Over ทันที
----------	--

4.5.3 Methods

+ void pause()	หยุดการทำงานของ thread
+ void unpause()	Resume การทำงานของ thread
+ <u>void</u> minusOne()	ลดจำนวนชีวิตลง 1
+ <u>void</u> reset()	Reset จำนวนชีวิตกลับเป็นค่าเริ่มต้น

4.6 Class application. Main extends Application

4.6.1 Fields

- <u>Stage menuStage</u>	Stage ของหน้า Menu
- <u>Stage gameStage</u>	Stage ของหน้า Game

4.6.2 Methods

+ void start(Stage primaryStage)	ทำการโหลด Font ทั้งหมด และแสดงหน้า Menu โดยใช้ loadFont() และ showMenuStage()
+ void main(String[] args)	Launch application
+ void showMenuStage()	Initialize menuStage โดยรับค่าจาก UIManager และแสดงหน้า Menu โดยให้ขยายขนาด Stage ไม่ได้พร้อมกับเล่นเพลงของหน้า menu
+ void showGameStage()	Initialize gameStage โดยรับค่าจาก GameManager และแสดงหน้า Game โดยให้ขยายขนาด Stage ไม่ได้พร้อมกับเล่นเพลงของหน้า game
+ void closeMenuStage()	ปิดหน้า Menu ลง
+ void closeGameStage()	ปิดหน้า Game ลง
- void loadFont()	Load font ที่ใช้ทั้งหมดในเกม

4.7 Class application. MediaManager

4.7.1 Fields

+ <u>double volume</u>	ตัวแปรเก็บค่าความดังของเสียงในเกม
- <u>String MAIN_PATH</u>	String url ที่ระบุถึงเพลงสำหรับหน้าเมนู
- <u>String GAME_PATH</u>	String url ที่ระบุถึงเพลงสำหรับหน้าเกม
- <u>String GET_SCORE</u>	String url ที่ระบุถึงเสียงเวลาได้รับคะแนน
- <u>String QUACK</u>	String url ที่ระบุถึงเสียงเวลาโดนลอบคะแนน
- <u>String CLICK</u>	String url ที่ระบุถึงเสียงเวลาคลิกปุ่ม
- <u>String ENTERED</u>	String url ที่ระบุถึงเสียงเวลาเลื่อนเมาส์ไปวางบนปุ่ม
- MediaPlayer mainPath	ตัวแปร MediaPlayer สำหรับเพลงหน้าเมนู
- MediaPlayer gamePath	ตัวแปร MediaPlayer สำหรับเพลงหน้าเกม
- AudioClip getScore	ตัวแปร AudioClip สำหรับเสียงเวลาได้รับคะแนน
- AudioClip quack	ตัวแปร AudioClip สำหรับเสียงเวลาโดนลอบคะแนน
- AudioClip click	ตัวแปร AudioClip สำหรับเสียงเวลาคลิกปุ่ม
- AudioClip entered	ตัวแปร AudioClip สำหรับเสียงเวลาเลื่อนเมาส์วางลงบนปุ่ม
+ <u>MediaManager instance</u>	ตัวแปร instance สำหรับใช้แบบ singleton

4.7.2 Constructor

+ MediaManager()	Initialize การเล่นของเพลงหน้าเมนูและหน้าเกม
------------------	---

4.7.3 Methods

+ void stopAll()	หยุดการเล่นเพลงหน้าเมนูและหน้าเกม
+ void update()	Update ความดังของเสียงตามค่าของ volume
+ void playMainPath()	หยุดเพลงทั้งหมดและเล่นเพลงหน้าเมนู
+ void playGamePath()	หยุดเพลงทั้งหมดและเล่นเพลงหน้าเกม
+ void pauseGamePath()	Pause เพลงหน้าเกม

+ void resumeGamePath()	Resume เพลงหน้าเกม
+ void replayGamePath()	ตั้งค่าเพลงในหน้าเกมให้กลับมาเริ่มต้นใหม่
+ void playClick()	เล่นเสียง clicked
+ void playEntered()	เล่นเสียง entered
+ void playGetScore()	เล่นเสียง getScore
+ void playQuack()	เล่นเสียง quack
+ <u>MediaManager getInstance()</u>	คืนตัวแปร instance

4.8 Class application. MenuSubscene extends Subscene

4.8.1 Fields

- boolean show	ตัวแปร boolean สำหรับแสดงว่า subScene แสดงอยู่หรือไม่ โดยมีค่าเริ่มต้นเป็น false
- <u>ButtonGenerator subSceneBtn</u>	ตัวแปร ButtonGenerator สำหรับปุ่มใน subScene
- <u>String BACKGROUND_PATH</u>	String url สำหรับระบุถึงภาพ background ของ หน้า subScene

4.8.2 Constructor

+ MenuSubscene()	Initialize SubScene ขนาดกว้าง 700 สูง 430.5 โดยมี root เป็น AnchorPane ที่ถูกตั้ง style เป็น BACKGROUND_PATH และหน้า SubScene มีตำแหน่งแกน x เป็น 1400 แกน y เป็น 245 โดยมีองค์ประกอบหน้า SubScene ดังนี้ <ul style="list-style-type: none"> - ปุ่ม ButtonGenerator “Back” โดยมี ความสูง 31.5 กว้าง 210 และตำแหน่งแกน x เป็น 450 แกน y เป็น 330
------------------	--

4.8.3 Methods

+ void transitionIn()	Show ตัว subscene โดยการเลื่อนเข้ามาในหน้า application โดยเลื่อน -875 และมีระยะเวลาการเล่น transition เป็น 300 ms
+ void transitionOut()	ซ่อนตัว subscene โดยเลื่อนออกจากหน้า application โดยเลื่อน 875 และมีระยะเวลาการเล่น transition เป็น 300 ms
+ Gette & Setter	<ul style="list-style-type: none"> - Getter ของ root - Getter ของ subSceneBtn - Getter ของ show

4.9 Class application. PauseButton extends Button

4.9.1 Fields

- double BUTTON_WIDTH	ความกว้างปุ่ม
+ <u>boolean isPause</u>	ตัวแปรเก็บสถานะการ pause
- String NORM_PAUSE_PATH	String url ที่ระบุถึงไฟล์ภาพปุ่ม pause ตอนปกติ
- String PRESSED_PAUSE_PATH	String url ที่ระบุถึงไฟล์ภาพปุ่ม pause ตอนโดนกด
- String NORM_PLAY_PATH	String url ที่ระบุถึงไฟล์ภาพปุ่ม play ตอนปกติ
- String PRESSED_PLAY_PATH	String url ที่ระบุถึงไฟล์ภาพปุ่ม play ตอนโดนกด
- String PAUSE_NORM_STYLE	Style ของปุ่ม pause ตอนปกติ
- String PAUSE_ONPRESS_STYLE	Style ของปุ่ม pause ตอนโดนกด
- String PLAY_NORM_STYLE	Style ของปุ่ม play ตอนปกติ
- String PLAY_ONPRESS_STYLE	Style ของปุ่ม play ตอนโดนกด

4.9.2 Constructor

+ PauseButton()	Initialize ปุ่มและ <ul style="list-style-type: none"> - ตั้งค่าขนาดปุ่มโดย setSize() - ตั้งค่า style ปุ่มเริ่มต้น โดย setNormStyle() - Set การรับ input ต่างๆ โดย setActionFromMouse()
-----------------	---

4.9.3 Methods

- void setSize()	ตั้งค่าความกว้างปุ่มตาม BUTTON_WIDTH
+ void restart()	เลิก pause เกมและตั้งทุกอย่างกลับเป็นค่า และstyleเริ่มต้น และเลิก disable ปุ่ม
- void setNormStyle()	ทำการตั้ง style ปุ่มเป็นแบบตอนปกติ โดย <ul style="list-style-type: none"> - ความสูงปุ่ม 49 - เลื่อนปุ่มขึ้น 10 - ถ้าเกม pause อยู่ตั้ง style เป็นปุ่ม play ตอนปกติ ถ้าไม่ pause ตั้ง style เป็นปุ่ม pause ตอนปกติ
- void setPressedStyle()	ทำการตั้ง style ปุ่มเป็นแบบตอนโดนกด โดย

	<ul style="list-style-type: none">- ความสูงปุ่ม 45- เลื่อนปุ่มลง 10- ถ้าเกม pause อยู่ตั้ง style เป็นปุ่ม play ตอนโดนกด ถ้าไม่ pause ตั้ง style เป็นปุ่ม pause ตอนโดนกด
- void setActionFromMouse()	<p>ตั้งค่าการรับ input ของปุ่มดังนี้</p> <ul style="list-style-type: none">- เมาส์เลื่อนมาวาง : สร้างเงาของปุ่ม- เมาส์เลื่อนออก : ลบเงาของปุ่ม- เมาส์กด : ตั้ง style ปุ่ม เป็นตอนโดนกด- เมาส์ปล่อย : ถ้าเกม pause อยู่ให้เล็ก และตั้งปุ่ม style เป็นตอนปกติ

4.10 Class application. PauseSubscene extends Subscene

4.10.1 Fields

- String BACKGROUND_PATH	String url ระบุถึงไฟล์ภาพ background ของ PauseSubScene
- AnchorPane root	PauseSubScene's root

4.10.2 Constructor

+ PauseSubscene()	Initialize SubScene โดยมี root เป็น AnchorPane และ <ul style="list-style-type: none"> - ตั้งขนาด subScene กว้าง 730 สูง 449 - ตั้ง style ของ SubScene ให้มี background เป็น ภาพตาม BACKGROUND_PATH - ตั้งตำแหน่งแกน x เป็น $(1366 - \text{ความกว้าง subScene})/2$ - ตั้งตำแหน่งแกน y เป็น $-(\text{ความสูง subScene})$
-------------------	--

4.10.3 Methods

+ void transitionIn()	สร้าง effect เลื่อนเข้ามาของ subScene โดยมีระยะเวลา transition เป็น 300 ms
+ void transitionOut()	สร้าง effect เลื่อนออกของ subScene โดยมีระยะเวลา transition เป็น 200 ms
+ Getter & Setter	Getter ของ root

4.11 Class application. SliderBar extends HBox

4.11.1 Fields

- <u>Slider slider</u>	SliderBar's slider
- <u>Label number</u>	SliderBar's number สำหรับแสดงค่าของตัวเลื่อน
- <u>String SLIDE_STYLE</u>	Style ของ slider
- <u>String LAB_STYLE</u>	Style ของ Label

4.11.2 Constructor

+ SliderBar()	Initialize Hbox โดย spacing เท่ากับ 10 และ <ul style="list-style-type: none"> - สร้าง label สำหรับคำว่า volume และ set style ตาม LAB_STYLE - สร้าง slider โดย createSlider() - สร้าง number โดย createLabel()
---------------	--

4.11.3 Methods

- void createSlider()	Initialize slider โดยมีค่าต่างๆดังนี้ <ul style="list-style-type: none"> - Min = 0, Max = 100 - ค่าเริ่มต้นเป็น MediaManager.volume*100 - ตั้งขนาด 296*16 - ตั้ง style ตาม SLIDE_STYLE
- void createLabel()	Initialize number โดยตั้งค่าดังนี้ <ul style="list-style-type: none"> - Style ตาม LAB_STYLE - Bind ค่ากับ slider - สร้าง thread มาคอย update ความดังเสียงในเกม
+ Getter & Setter	Getter ของ number

4.12 Class application. UIManager

4.12.1 Fields

- Stage mainStage	UIManager's Stage
- Scene mainScene	UIManager's Scene
- AnchorPane uiRoot	UIManager's Root
- MenuSubscene creditSubscene	ตัวแปร MenuSubScene ของหน้า credit
- MenuSubscene optionSubscene	ตัวแปร MenuSubScene ของหน้า option
- MenuSubscene playSubscene	ตัวแปร MenuSubScene ของหน้า play
- MenuSubscene curShowSubscene	ตัวแปร MenuSubScene สำหรับเก็บค่าตัว SubScene ที่แสดงอยู่
- MenuSubscene dummySubscene	ตัวแปร MenuSubScene สำหรับเก็บค่าตัว SubScene ที่ใช้เป็นตัวสลับไปมา
- String BACKGROUND_PATH	String url ที่ระบุถึงไฟล์ภาพ Background
- String LOGO_PATH	String url ที่ระบุถึงไฟล์ภาพ logo
- String CURSOR_PATH	String url ที่ระบุถึงไฟล์ภาพ cursor

4.12.2 Constructor

+ UIManager()	Initialize ตัวแปรต่างๆดังนี้ <ul style="list-style-type: none"> - Initialize uiRoot, mainScene และ mainStage โดยมีขนาด 1366*768 - ตั้งภาพ background โดย createBackground() - สร้างปุ่ม โดย createAllButton() - สร้าง logo โดย createLogo() - ตั้ง style cursor โดย customCursor() - สร้าง SubScene ต่างๆ โดย createSubScene()
---------------	--

4.12.3 Methods

- void createAllButtons()	สร้างปุ่มทั้งหมด โดย <ul style="list-style-type: none"> - ปุ่ม play โดย createPlayButton() - ปุ่ม credit โดย createCreditButton() - ปุ่ม option โดย
---------------------------	--

	createOptionButton() - ปุ่ม exit โดย createExitButton()
- void createPlayButton()	สร้างปุ่ม play โดย ButtonGenerator โดยตั้งค่าดังนี้ <ul style="list-style-type: none"> - Text เป็น "PLAY" - ตำแหน่ง แกน x ที่ 122.5 แกน y ที่ 315 - ตั้งการรับ input เมาส์คลิกให้แสดง subScene นี้เมื่อคลิกซ้ายและ ถ้าแสดงอยู่แล้วให้ซ่อน subScene
- void createCreditButton()	สร้างปุ่ม credit โดย ButtonGenerator โดยตั้งค่าดังนี้ <ul style="list-style-type: none"> - Text เป็น "CREDIT" - ตำแหน่ง แกน x ที่ 122.5 แกน y ที่ 385 - ตั้งการรับ input เมาส์คลิกให้แสดง subScene นี้เมื่อคลิกซ้ายและ ถ้าแสดงอยู่แล้วให้ซ่อน subScene
- void createOptionButton()	สร้างปุ่ม option โดย ButtonGenerator โดยตั้งค่าดังนี้ <ul style="list-style-type: none"> - Text เป็น "OPTION" - ตำแหน่ง แกน x ที่ 122.5 แกน y ที่ 455 - ตั้งการรับ input เมาส์คลิกให้แสดง subScene นี้เมื่อคลิกซ้ายและ ถ้าแสดงอยู่แล้วให้ซ่อน subScene
- void createExitButton()	สร้างปุ่ม exit โดย ButtonGenerator โดยตั้งค่าดังนี้ <ul style="list-style-type: none"> - Text เป็น "EXIT" - ตำแหน่ง แกน x ที่ 122.5 แกน y ที่ 525 - ตั้งการรับ input เมาส์คลิกซ้ายให้ออกเกมทันที
- void createBackground()	Set background ตาม BACKGROUND_PATH
- void createLogo()	สร้าง UIImageView ของ logo ตาม LOGO_PATH และตั้งตำแหน่งแกน x แกน y เป็น 100 และ 120 ตามลำดับ
- void createSubscene()	Initialize SubScene ทั้งหมดโดย <ul style="list-style-type: none"> - playSubscene โดย createPlaySubscene() - creditSubScene โดย createPlaySubscene() - optionSubScen โดย createOptionSubscene() - curShowSubScene และ dummySubScene ให้ initialize ตาม

	ปกติ
- void createPlaySubscene()	Initialize playSubScene โดย <ul style="list-style-type: none"> - Set background ตามภาพ play_scene.png - สร้าง Label คำว่า "Tutorial" ขนาด 36 มี Alignment เป็น Pos.Center ความกว้างตามขนาด subScene ตำแหน่งแกน y เป็น 30 - ตั้ง subSceneBtn เป็นคำว่า "ok" เมื่อโดนกดให้เริ่มเกม
- void createCreditSubscene()	Initialize creditSubScene โดย <ul style="list-style-type: none"> - Set background ตามภาพ credit_scene.png - ตั้ง subSceneBtn เมื่อโดนกดให้ซ่อน subScene
- void createOptionSubscene()	Initialize playSubScene โดย <ul style="list-style-type: none"> - สร้าง Label คำว่า "Option" ขนาด 36 มี Alignment เป็น Pos.Center ความกว้างตามขนาด subScene ตำแหน่งแกน y เป็น 30 - ตั้ง subSceneBtn เมื่อโดนกดให้ซ่อน subScene - สร้าง SliderBar ไว้ที่ตำแหน่ง แกน x ที่ 100 แกน y ที่ 180
- void customCursor()	ตั้ง style ของ cursor ตาม CURSOR_PATH
+ Getter & Setter	Getter ของ mainStage & mainScene

4.13 Class draw. GameScreen extends Canvas

4.13.1 Fields

- Random random	GameScreen's random สำหรับสุ่ม
- List<Character> huntersAlphabets	ArrayList ของตัวอักษรของนายพรานที่อยู่บนหน้าจอเกม
- List<Character> animalsAlphabets	ArrayList ของตัวอักษรของสัตว์ที่อยู่บนหน้าจอเกม
- Character[] ALPHABETS	อาเรย์เก็บตัวอักษรภาษาอังกฤษ A - Z

4.13.2 Constructor

+ GameScreen(double width, double height)	Initialize ค่าต่างๆ <ul style="list-style-type: none"> - Initialize Canvas ขนาดตาม parameter ที่รับเข้ามา - Initialize random - Initialize huntersAlphabets และ animalsAlphabets
---	---

4.13.3 Methods

+ int createRandomPosX()	สุ่มตำแหน่งในแกน x
+ int createRandomPosY()	สุ่มตำแหน่งในแกน y
+ String createAnimalsKey()	สุ่มสร้างตัวอักษรให้กับสัตว์ โดยจะไม่เอาตัวอักษรเดียวกับของนายพราน
+ String createHuntersKey()	สุ่มสร้างตัวอักษรให้กับนายพราน โดยจะไม่เอาตัวอักษรเดียวกับของสัตว์
+ void drawComponent()	เรียก method draw() ของตัวแปร IRenderable ใน container ของ IRenderableHolder ถ้ายังไม่ถูกทำลายและ visible
+ Getter & setter	Getter ของ huntersAlphabets และ animalsAlphabets

4.14 Class Exception. NegativeScoreException extends Exception

4.14.1 Constructor

+ NegativeScoreException()	Throw exception พร้อมข้อความ the score would not be negative!
----------------------------	---

4.15 Class Input. IOManager

4.15.1 Fields

- <u>String code</u>	ตัวแปรเก็บตัวอักษรที่โดนกด
- <u>boolean pressed</u>	ตัวแปรเก็บสถานะการ pressed
- <u>boolean triggered</u>	ตัวแปรเก็บสถานะการ triggered

4.15.2 Methods

+ <u>void postUpdate()</u>	ตั้งให้ triggered เป็น false
+ Getter & Setter	<ul style="list-style-type: none">- Getter & Setter ของ pressed- Getter & Setter ของ triggered ถ้าโดน trigger ให้ เก็บตัวแปร codeเป็นตัวอักษรที่กดด้วย- Getter ของ code

4.16 Class logic. Animals extends FallableUnit

4.16.1 Fields

+ <u>int WIDTH</u>	ความกว้างของรูปสัตว์
+ <u>int HEIGHT</u>	ความสูงของรูปสัตว์
- <u>int score</u>	คะแนนที่จะถูกหักเมื่อกำจัดสัตว์
- boolean flashing	ตัวแปรเก็บสถานะการกระพริบ
- int flashCount	จำนวนครั้งการกระพริบ
- int flashDuration	ระยะเวลาของการกระพริบ
- int showDuration	ระยะเวลาของการแสดงระหว่างการกระพริบ

4.16.2 Constructor

+ Animals(double x, double y, String key, double speed)	Initialize ตัวแปรทั้งหมดตาม parameters ที่รับเข้ามาและให้ค่า z ของสัตว์เป็น 8
---	---

4.16.3 Methods

+ void draw(GraphicsContext gc)	วาดภาพตาม key ของสัตว์
+ void update()	<ul style="list-style-type: none"> - เรียก method fall() - ถ้ากด keyboard ตรงกับ key ของสัตว์ให้ทำลายสัตว์และลบคะแนนและชีวิต - ถ้าสัตว์ตกลงถึงพื้นให้ speed เป็น 0 และให้กระพริบ - ถ้า game Over ให้ทำลายสัตว์

4.17 Class logic. Background implements IRenderable

4.17.1 Methods

+ int getZ()	คืนค่าเป็น -999
+ void draw(GraphicsContext gc)	วาดภาพตามภาพจาก IRenderableHolder.bg ที่ตำแหน่ง (0, 0) ขนาด 1366*768
+ boolean isVisible()	คืนค่า true
+ boolean isDestroyed()	คืนค่า false
+ void update()	

4.18 Class logic. FallableUnit (Abstract) extends Unit

4.18.1 Fields

- <u>int fallUnitSpawnRate</u>	อัตราการสร้างตัวละครที่เป็น FallableUnit
- <u>boolean isSpawnRateIncreased</u>	สถานะการเพิ่มอัตราการสร้างตัวละคร FallableUnit
- <u>double fallUnitSpeed</u>	ความเร็วของตัวละคร FallableUnit
# double speed	ความเร็วในการตก
# String key	ตัวแปรใช้เก็บค่าตัวอักษรของ Unit

4.18.2 Methods

# bool landOnGround()	คืนค่า true เมื่อความสูงของพื้นกับความสูงของ unit ห่างกันไม่เกิน 150 ไม่อย่างนั้นคืนค่า false
# void fall()	เพิ่มค่า y = speed
+ <u>void accelerate()</u>	fallUnitSpeed เพิ่มความเร็วเท่ากับ 5% ของความเร็ว fallUnitSpeed เดิม
+ <u>void increaseSpawnRate</u>	เพิ่ม fallableUnitSpawnRate 1 หน่วย แต่ fallableUnitSpawnRate ห้ามเกิน 5
+ <u>void resetValues()</u>	<ul style="list-style-type: none"> - กำหนดให้ fallUnitGenrate เป็น 1 - กำหนดให้ fallUnitSpeed เป็น 2 - กำหนดให้ isGenRateIncreased เป็น false
+ Getter & Setter	<ul style="list-style-type: none"> - <u>Getter</u> ของ fallUnitSpawnRate - <u>Getter & Setter</u> ของ isSpawnRateIncreased - <u>Getter</u> ของ fallableUnitSpeed - <u>Getter</u> ของ key

4.19 Class logic. Gamellogic

4.19.1 Field

- List<IRenderable> gameContainer	ตัวแปรที่ใช้เก็บ Object ที่เป็น IRenderable
-----------------------------------	---

4.19.2 Constructor

+ GameLogic()	<ul style="list-style-type: none"> - ประกาศ gameContainer เป็น ArrayList - เพิ่ม background และ ground ลงใน IRenderableHolder
---------------	---

4.19.3 Methods

+ addNewObj(IRenderable r)	เพิ่ม Object r ลงใน gameContainer และ IRenderableHolder
+ void clear()	ลบทุกสมาชิกใน gameContainer
+ logicUpdate()	ลบสมาชิกทุกตัวใน gameContainer ที่มีค่า isDestryed เป็น true และ update() ที่เหลือ

4.20 Class logic. Grounds extends Unit

4.20.1 Fields

+ <u>double</u> WIDTH	ค่าความกว้าง
+ <u>double</u> HEIGHT	ค่าความสูง

4.20.2 Constructor

+ Grounds()	<ul style="list-style-type: none"> - กำหนดค่า z เป็น -998 คืออยู่หลังสุดรองจากภาพ background - กำหนดค่า x เป็น 0 - กำหนดค่า y เป็น 668
-------------	---

4.20.3 Methods

+ void draw(GraphicsContext gc)	วาดภาพ ground ที่ตำแหน่ง x, y
+ bool isDestroyed()	คืนค่า false
+ bool isVisible()	ค่าค่า true
+ void update()	
+ <u>double</u> getY()	คืนค่า 668 (ค่าของ y)

4.21 Class logic. Hunters extends FallableUnit

4.21.1 Fields

- <u>int score</u>	คะแนนที่ได้จากการกำจัดนายพราน
+ <u>int WIDTH</u>	ความกว้างของนายพราน
+ <u>int HEIGHT</u>	ความสูงของนายพราน

4.21.2 Constructor

+ Hunters(double x, double y, String key, double speed)	Set ค่าตัวแปร x, y, key, speed และ ให้ z เป็น 9
---	---

4.21.3 Methods

+ void draw(GraphicsContext gc)	วาดภาพนายพรานซึ่งมีตัวอักษร key แปะอยู่บนร่ม
+ void update()	คอยรับค่า key ที่ใส่เข้ามาจาก IManager ถ้าตรงกับ key ของนายพราน ให้ทำลาย พร้อมกับ getScore และ เล่นเสียง

4.22 Class logic. scoreCount extends HBox

4.22.1 Fields

- Label scoreBox	คะแนนที่แสดงทางหน้าจอ
- int scoreCount	ตัวนับคะแนน
- <u>int temporaryScore</u>	ตัวนับการเปลี่ยนแปลงของคะแนน
- String TEXT_STYLE	String ที่ใช้ setStyle ของข้อความ

4.22.2 Constructor

+ ScoreCount()	กำหนด scoreCount เป็น 0 เรียก createBox()
----------------	--

4.22.3 Methods

- void createBox()	<ul style="list-style-type: none"> - สร้าง Label คำว่า "SCORE: " - สร้าง scoreBox โดยให้ค่าเริ่มต้นเป็น 00000 - add Label และ scoreBox ลงในคลาส
+ void update() throws NegativeScoreException()	<ul style="list-style-type: none"> - นำ scoreCount และ temporaryScore มาบวกกัน และให้ temporaryScore เป็น 0 - ให้ text ของ scoreBox เป็น scoreCount ที่เป็น String 5 หลัก นำหน้าด้วย 0
+ <u>void addScore(int score)</u>	เพิ่ม score ลงใน temporaryScore
+ void resetScore()	ให้ scoreCount มีค่าเป็น 0
+ int getScoreCount()	คืนค่า scoreCount

4.23 Class logic. Unit (Abstract) implements IRenderable

4.23.1 Fields

# double x	ตำแหน่งแนวแกน x
# double y	ตำแหน่งแนวแกน y
# int z	ความตื้น-ลึกของ
# boolean visible	เก็บค่าสถานะการแดง
# boolean destroyed	เก็บค่าสถานะการถูกทำลาย
# int UnitScore	คะแนนที่จะได้หรือถูกหักจาก unit

4.23.2 Constructor

# Unit()	set visible = true set destroyed = false
----------	---

4.23.3 Methods

+ int getZ()	คืนค่า z
+ boolean isVisible()	คืนสถานะ visible
+ boolean isDestroyed()	คืนสถานะ destroyed
+ int getUnitScore()	คืน unitScore

4.24 Interface sharedObject.IRenderable

4.24.1 Methods

+ int getZ()	
+ void update()	
+ boolean isVisible()	
+ boolean isDestroyed()	
+ void draw(GraphicsContext gc)	

4.25 Class sharedObject. IRenderableHolder

4.25.1 Fields

- List<IRenderable> container	List ที่เก็บตัวแปร IRenderable
- Comparator<IRenderable> comparator	ตัวแปรที่เก็บสถานะการเปรียบเทียบค่า z เพื่อจัดลำดับการ แสดงข้างหน้าข้างหลังบนหน้าจอ
- IRenderableHolder instance	เก็บค่าคงที่ของคลาส IRenderableHolder
+ Image a1, a2, a3, a4, a5, a6, a7, a8, a9, a10, a11, a12, a13, a14, a15, a16, a17, a18, a19, a20, a21, a22, a23, a24, a25, a26	รูปภาพของตู้ container ของสัตว์ ที่มีตัวอักษรภาษาอังกฤษ 1 ตัวอยู่บนร่ม จำนวน 26 รูปครบทุกตัวอักษร
+ Image bg1	รูปภาพพื้นหลังของเกม
+ Image s1, s2, s3, s4, s5, s6, s7, s8, s9, s10, s11, s12	รูปภาพของตู้ container ของสัตว์ที่ไม่มีตัวอักษรใด ๆ ทั้งหมด 12 สถานะตั้งแต่เริ่มทางจนถึงร่มหุบและหายไป
+ Image h1	รูปนายพราน
+ Image p1	รูปร่มของนายพราน
+ Image ground	รูปพื้นของเกมใน GameStage

4.25.2 Constructor

+ IRenderableHolder()	<ul style="list-style-type: none"> - ประกาศตัวแปร container ขึ้นมาเป็น ArrayList<IRenderable> - ประกาศตัวแปร comparator ให้เปรียบเทียบค่า z ของ Object IRenderable 2 ตัว โดยที่คืนค่าเป็น 1 เมื่อตัวแรกมีค่า z มากกว่าตัวที่สอง และคืนที่ -1 เมื่อตัวแรกมีค่าน้อยกว่าตัวที่สอง และถ้าทั้งสองตัวเป็นนายพรานหรือสัตว์ให้เรียงจาก Z ไป A เพื่อให้เวลา draw() ตัวอักษรที่น้อยกว่าจะอยู่ด้านหน้ากว่า
-----------------------	---

4.25.3 Methods

+ void add(IRenderable unit)	เพิ่ม Object IRenderable ลงใน container และทำการเรียงใหม่
+ void update()	เช็ค Object ใน container ทีละตัวว่า isDestroy() ถ้ามีสถานะเป็น true ให้ลบออกจาก container

+ <u>IRenderableHolder getInstance()</u>	คืนค่าตัวแปร instance
+ <u>ArrayList<IRenderable> getContainer()</u>	คืนค่าตัวแปร container เป็น ArrayList<IRenderable>
+ <u>void clear()</u>	ลบ Object ใน container ทุกตัวที่เป็นสมาชิกของ class Animals หรือ Hunters
- <u>void loadResource()</u>	LoadResource ของรูปภาพทุกรูปในคลาส IRenderableHolder โดยใช้ ClassLoader