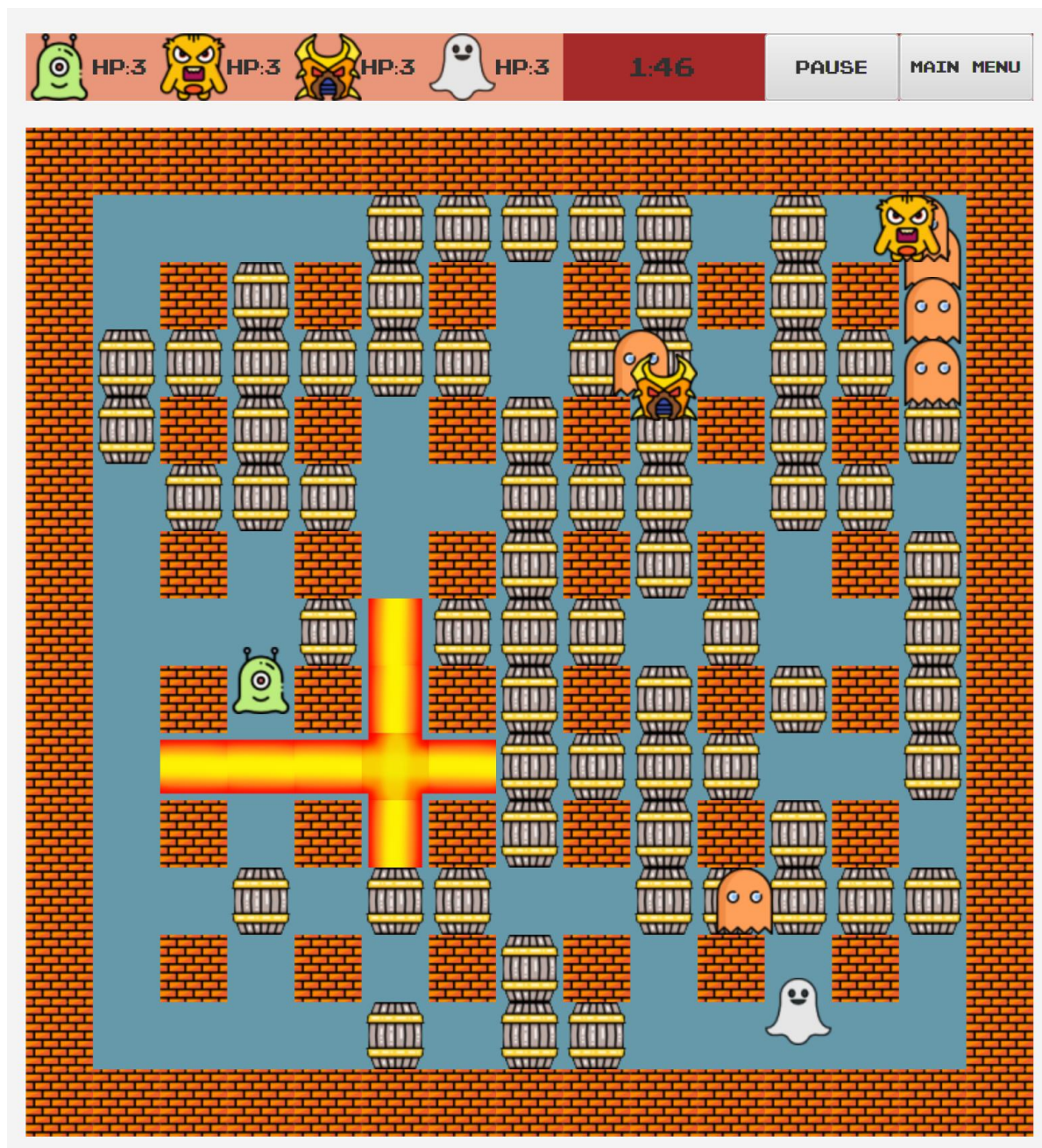


E.T. Bomber

เกม E.T. Bomber เป็นเกมที่เราต้องบังคับผู้เล่นไปวางระเบิดที่จุดต่างๆ เพื่อทำลายสิ่งกีดขวางและศัตรูที่มีความสามารถต่างๆ ซึ่งในขณะเดียวกันก็ต้องทำการหลบหลีกลูกน้องของศัตรูด้วย โดยในเกมจะมีไอเทมต่างๆ ค่อยเพิ่มความสามารถให้เรา โดยผู้เล่นสามารถบังคับตัวละครได้ด้วยปุ่ม W,A,S,D เพื่อเดินขึ้น ซ้าย ลง ขวา ตามลำดับ และกดปุ่ม Space เพื่อวางระเบิด



2. Implementation Details:

เพื่อความง่าย จะขอแบ่งข้อมูลต่างๆ ออกตาม package และอักษรจาก A-Z

สำหรับ Modifier ต่างๆ จะใช้สัญลักษณ์ดังนี้

public จะแทนด้วยเครื่องหมาย +

protected จะแทนด้วยเครื่องหมาย #

private จะแทนด้วยเครื่องหมาย -

static จะแทนด้วยการใช้ตัวหนา

ทุก Class เป็น public class

2.1 Package ai

2.1.1 Action

2.1.2 AIBase

2.1.3 AIStatusCheckList

2.1.4 Boss

2.1.5 CannotReachDestinationException

2.1.6 Minion

2.1.7 PathFinding

2.2 Package controller

2.2.1 Controller

2.2.2 GameController

2.2.3 InputInGame

2.2.4 LevelGenerator

2.2.5 ObjectInGame

2.2.6 StartPageController

2.3 Package gameobject

2.3.1 Destroyable

2.3.2 GameObject

2.3.3 Moveable

2.3.4 Obstacle

2.3.5 Wall

2.4 Package gui

2.4.1 CreditPage

2.4.2 GameField

2.4.3 GamePage

2.4.4 GameSummaryPage

2.4.5 HasButtonPage

2.4.6 InstructionPage

2.4.7 PlayerStatusBoard

2.4.8 ScoreBoard

2.4.9 StartPage

2.5 Package item

2.5.1 BombUpgradeItem

2.5.2 Item

2.5.3 LifeIncreaseItem

2.5.4 PowerUp

2.5.5 PowerUpgradeItem

2.5.6 Shield

2.5.7 SpeedUpgradeItem

2.6 Package main

2.6.1 Main

2.7 Package player

2.7.1 Player

2.7.2 PlayerBase

2.7.3 PlayerState

2.8 Package setting

2.8.1 Setting

2.9 Package weapon

2.9.1 Bomb

2.9.2 BombArea

2.9.3 Site

2.9.4 Weapon

2.1 Package ai

2.1.1 class Action

2.1.1.1 Field

- GameController gameController	Instance ของ GameController
- PathFinding pathFinding	Instance ของ PathFinding
- Random random	Instance ของ Random

2.1.1.2 Method

- boolean canMove(AIBase, ObjectInGame)	ตรวจสอบว่าจุดหมายนั้นสามารถเดินไปได้หรือไม่
+ void checkForBomb(AIBase)	ตรวจสอบว่าในระยะใกล้ AIBase นั้นมีระเบิดหรือไม่
+ void checkForWayAndItem(AIBase)	ตรวจสอบและเก็บว่าในระยะรอบตัวมี item หรือช่องทางเดินหรือไม่
+ void collectItem(AIBase)	ตรวจสอบว่าหาก $hp > 0$ และ ในระยะใกล้ๆ นั้นมี item หรือไม่ หากมีจะทำการตั้งเป้าหมายให้ไปเก็บ item นั้น
+ void dead(AIBase)	ตรวจสอบว่า AIBase นั้นมี hp เป็น 0 หรือไม่ ถ้าเป็นให้แก้ค่าใน AIStatusChecklist.isDead เป็น true
+ void escapeBomb(AIBase)	ตรวจสอบว่า หาก $hp > 0$ และมีระเบิดอยู่ในระยะใกล้ๆ หรือไม่ หากมีจะทำการเดินหนีหรือหาระยะปลอดภัยเพื่อหลบระเบิด
+ void goTo(AIBase)	ตรวจสอบว่าหาก $hp > 0$ จะทำการเดินไปยังจุดหมาย
- boolean isItem(AIBase, ObjectInGame)	ตรวจสอบว่าใน ObjectInGame นั้นเป็น item หรือไม่
+ void randomWalking(AIBase)	ตรวจสอบว่าหาก $hp > 0$

	และไม่มีจุดมุ่งหมายในการเดินให้ ทำการเดินแบบสุ่ม
+ void setGameController(GameController)	ทำการกำหนดค่าให้กับ gameController, pathFinding และ random
+ void spawnMinion(Boss)	ตรวจสอบว่าหาก hp > 0 จะทำการ spawn Minion ตามเวลา
+ void vanish(AIBase)	ตรวจสอบว่าหาก hp > 0 และเป็น Boss หรือ Minion ที่มี playerNumber เป็น 4 จะทำการหายตัวและแสดงตัวตามเวลา

2.1.2 abstract class AIBase extends PlayerBase

เป็น abstract class ที่จะถูก extends จาก Boss และ Minion

2.1.2.1 Field

# AIBase aiStatus	instance ของ AIBase
+ ObjectInGame[] objectAroundPlayer	เป็น Array ที่ใช้เก็บ ObjectInGame ที่อยู่รอบตัว
+ ObjectInGame[] objectInSightPlayer	เป็น Array ที่ใช้เก็บ ObjectInGame ที่สามารถมองเห็นได้
+ int[] objectInSightPlayer	เป็น Array ที่ใช้เก็บ ระยะที่ห่างจาก ObjectInGame ที่อยู่รอบตัว
# PlayerBase player	instance ของ ผู้เล่น

2.1.2.2 Constructor

AIBase(int, int, String, Pane, int, GameController, PlayerBase)	ทำการเรียก constructor ของ PlayerBase
---	---------------------------------------

2.1.2.3 Method

+ int[] calculatePosition(AIBase, int)	คืนค่าช่องตำแหน่งของ AIBase โดย int ที่ให้มามีค่าคือ 0 ถึง 8 เริ่มจากทิศบนวนทวนเข็มนาฬิกา และจบที่ตำแหน่งตรงกลาง
---	--

+ abstract void checkForAction()	ใช้สำหรับตรวจสอบว่า จะทำ Action ใด
# void checkPlayerAndObjectInSight()	ทำการเก็บค่าให้กับตัวแปร objectAroundPlayer, objectInSightPlayer และ objectInSightPlayer
# AIStatusCheckList getAIStatus()	คืนค่าเป็นตัวแปร aiStatus
+ PlayerBase getPlayer()	คืนค่าเป็นตัวแปร player
Override + boolean isDead()	ทำคืนค่า aiStatus.isDead()

2.1.3 class AIStatusCheckList

2.1.3.1 Field

+ boolean[] bombDirection	เก็บค่าว่ามีระเบิดในทิศทางต่างๆ รอบตัวหรือไม่ หากมีจะมีค่าเป็น True
+ boolean bombNearBy	เก็บค่าว่า รอบๆ ตัวมีระเบิดหรือไม่
+ int[] bombRange	เก็บค่าระยะของระเบิดห่างจากตัวที่ช่องตามทิศทางของตัวแปร bombDirection
+ boolean isDead	เก็บค่าว่า HP เป็น 0 หรือไม่
+ boolean isEscape	เก็บค่าว่า กำลังหลบระเบิดอยู่หรือไม่
+ boolean isEscapeComplete	เก็บค่าว่า หลบระเบิด สำเร็จหรือไม่
+ boolean isFinishMoving	เก็บค่าว่า เดินไปถึงเป้าหมาย หรือยัง
+ boolean isMakeAction	เก็บค่าว่า ได้ทำการหาตัวแล้วหรือไม่
+ boolean isMoving	เก็บค่าว่า กำลังเคลื่อนที่ อยู่หรือไม่
+ boolean isSpawnMinion	เก็บค่าว่าได้ spawn minion หรือไม่
+ boolean isVanish	เก็บค่าว่าในขณะนี้ต้องการหาตัวหรือไม่
+ boolean[] item	เก็บค่าว่ามี item ในทิศทางต่างๆ รอบตัวหรือไม่ หากมีจะมีค่าเป็น True

+ PlayerState moveDirection	เก็บค่าว่าจะเดินทางไปทิศทางไหน
+ int moveToX	เก็บค่าจุดเป้าหมายที่จะเดินไปในแกน x หาก เป็น -1 ให้หยุดนิ่ง หาก เป็น -2 ให้เดินแบบสุ่ม
+ int moveToY	เก็บค่าจุดเป้าหมายที่จะเดินไปในแกน y หาก เป็น -1 ให้หยุดนิ่ง หาก เป็น -2 ให้เดินแบบสุ่ม
+ long nextShowTime	เก็บค่าเวลาถัดไปที่จะแสดงตัว
+ long nextVanishTime	เก็บค่าเวลาถัดไปที่จะหายตัว
+ long showTime	เก็บค่า จะแสดงตัวเป็นระยะเวลากี่วินาที
+ long vanishTime	เก็บค่า จะหายตัวเป็นระยะเวลากี่วินาที
+ boolean[] ways	เก็บค่าว่ามีเส้นทางที่เดินได้ในทิศทางต่างๆ รอบตัวหรือไม่ หากมีจะมีค่าเป็น True

2.1.4 class Boss **extends** AIBase

2.1.4.1 Constructor

+ Boss(int, int, String, Pane, int, GameController, PlayerBase)	เรียก constructor ของ AIBase
---	------------------------------

2.1.4.2 Method

Override + void checkForAction()	ทำการตรวจสอบ Action ที่ Boss จะทำ
-------------------------------------	-----------------------------------

2.1.5 class CannotReachDestinateException **extends** Exception

2.1.6 class Minion **extends** AIBase

2.1.6.1 Constructor

Boss(int, int, String, Pane, int, GameController, PlayerBase)	เรียก constructor ของ AIBase
---	------------------------------

2.1.6.2 Method

+ void checkContactPlayer()	ทำการตรวจสอบว่าได้ชนกับ player หรือไม่ หากชนให้ทำการลดเลือด player และทำลายตัวเอง
Override + void checkForAction()	ทำการตรวจสอบ Action ที่ Minion จะทำ

2.1.7 PathFinding

เป็น class สำหรับหาเส้นทางที่ **Boss** จะใช้หนีระเบิดหรือเก็บ **item** ซึ่งทำงานโดยใช้ A star algorithm

2.1.7.1 Node (Inner class)

Node จะเก็บตำแหน่งที่อยู่ปัจจุบัน (x, y) และเก็บจุดเริ่มต้น, จุดปลายทาง และ **Node** พ่อของตัวเอง ซึ่งจะใช้ในการคำนวณระยะทางที่สั้นที่สุด

2.1.7.1.1 Field

- Node destinate	instance ของ Node ปลายทาง
- Node parentNode	instance ของ Node พ่อ
- Node startNode	instance ของ Node จุดเริ่มต้น
- int x	ตำแหน่งในแนวแกน x
- int y	ตำแหน่งในแนวแกน y

2.1.7.1.2 Constructor

+ Node(int, int, Node, Node, Node)	กำหนดค่าให้ x, y, parentNode, startNode และ destinate
------------------------------------	---

2.1.7.1.3 Method

Override + boolean equals(Object)	ตรวจสอบว่าเป็น Node เดียวกันหรือไม่ หาก x และ y เท่ากันให้คืนค่า true
- int getF()	คืนค่า F (อ้างอิงจาก AStar algorithm)
- Node getParent()	คืน parentNode
- int[] getXY()	คืนค่า x และ y ในรูป array ขนาด 2

- boolean isFinish()	ตรวจสอบว่า X และ y ปัจจุบันเท่ากับ X และ y ของ destinate หรือไม่
----------------------	--

2.1.7.2 Field

- GameController gameController	instance ของ GameController
---------------------------------	-----------------------------

2.1.7.3 Constructor

+ PathFinding(GameController)	กำหนดค่าให้กับ gameController
-------------------------------	-------------------------------

2.1.7.4 Method

- boolean checkContain(Node, List<Node>)	ตรวจสอบว่าใน List<Node> นั้นมี Node ที่จะหาหรือไม่ หากมีคืนค่า true
+ int[] findPath(int, int, int, int, Playerbase)	หาเส้นทางจากจุด (x1, y1) ไปที่จุด (x2, y2) โดยตรวจสอบว่าหาก PlayerBase ที่ใส่มาเป็น Boss ที่มีความสามารถเดินทะลุสิ่งกีดขวางได้ให้หาเส้นทางโดยไม่ต้องหลบสิ่งกีดขวาง
- void nodeSort(List<Node>)	เรียงลำดับ Node ใน List จากน้อยไปมากตามค่า F

2.2 Package controller

2.2.1 **abstract** class Controller

เป็น abstract class ที่จะถูก extends โดย GameController และ StartPageController

2.2.1.1 Field

# Controller otherController	เก็บ instance ของ อีก Controller นึง
# Scene scene	instance ของ Scene
# Stage stage	instance ของ Stage

2.2.1.2 Method

# abstract Scene createScene()	สร้าง Scene จากนั้นคืนค่า Scene ที่สร้าง
Getters and Setters	

2.2.2 class GameController **extends** Controller

2.2.2.1 Field

- Thread createBombThread	Thread ของการสุ่มระเบิด
- int currentNextThreadTime	เก็บค่าเวลาที่จะเปลี่ยนความถี่ในการสุ่มระเบิด
- AnimationTimer gameLoop	Instance ของ AnimationTimer ที่จะใช้ในการแสดงภาพในเกม
- GameObject[][] gameObjectArray	เก็บค่าของ item ในแต่ละช่อง
- GamePage gamePage	Instance ของ GamePage
- GameSummaryPage gameSummaryPage	Instance ของ GameSummaryPage
- InputInGame inputInGame	Instance ของ InputInGame
- boolean isPlaying	เก็บค่าว่า gameLoop กำลังเล่นอยู่หรือไม่
+ int level	เก็บค่า level ปัจจุบัน
- LevelGenerator levelGenerator	Instance ของ LevelGenerator

- List<Minion> minions	List ที่ใช้เก็บ minion
- List<PlayerBase> players	List ที่ใช้เก็บ Player และ Boss
- long remainingTime	เก็บค่าเวลาที่เหลือในด้านนั้น (เริ่มต้นที่ 150 วินาที)
- long startTime	เก็บค่าเวลาที่ gameLoop เริ่มทำงาน

2.2.2.2 Method

+ boolean canSetObject(int, int)	คืนค่าว่าที่ช่อง x และ y (0 - 14) สามารถวาง item ได้หรือไม่
- void checkGameFinish()	ตรวจสอบว่าจบ level แล้วหรือไม่ โดยจะจบก็ต่อเมื่อเงื่อนไขต่างๆ เป็นจริงดังนี้ 1. ผู้เล่น hp = 0 2. Boss อื่นๆ hp = 0 3. เวลาหมด
+ boolean checkMove(int, int, PlayerBase)	ตรวจสอบว่าที่ช่องตำแหน่ง x และ y สามารถเดินผ่านได้หรือไม่
- void checkPlayerGetItem(PlayerBase)	ตรวจสอบว่า PlayerBase ทำการเก็บ item
- void checklayerMoveAndSetState(PlayerBase)	ทำการรับค่าปุ่มกดคีย์บอร์ด และ กำหนดค่า currentPlayerState ของ Player
- void checkPlayerPlaceBomb(PlayerBase)	ทำการรับค่าว่าผู้เล่นได้ทำการวางระเบิดหรือไม่
- void checkRemainingTime()	ตรวจสอบเวลาที่เหลือ เพื่อแก้ไขอัตราการสุ่มระเบิด
+ int[] checkXYSpeed(int, int, PlayerBase)	ตรวจสอบและคืนค่าความเร็วของ PlayerBase ในทิศ x และ y
- void clearGame()	ทำการ reset เป็นค่าเริ่มต้นก่อนเริ่มเกม
- void createBackground()	ทำการสร้างพื้นหลัง
- void createGame()	ทำการสร้างเกมตามค่าที่สุ่มได้จาก LevelGenerator

- void createPlayer()	ทำการสร้าง Player และ Boss
Override # Scene createScene()	ทำการสร้าง GameScene และ เรียก createBackground(), createGame() และ createPlayer()
+ AnimationTimer gameLoop()	ทำการสร้าง gameLoop และคืนค่า
- void generateBombThread(int, int, long)	ทำการสร้าง createBombThread ตามอัตราที่กำหนดให้
+ GameObject getGameObject(int, int)	คืนค่าเป็น GameObject ที่ตำแหน่ง x, y
- void getItem(int, int, PlayerBase)	ทำการนำ item ที่ตำแหน่ง x, y ให้กับ PlayerBase
+ ObjectInGame getObjectInGame(int, int)	คืนค่าเป็น ObjectInGame ที่ตำแหน่ง x, y
+ void onRemoveScene()	ทำการเรียกเมื่อจะเปลี่ยน scene โดยจะ removeGame() และ นำ input ออก
- void removeGame()	ทำการลบข้อมูลและทำลาย ObjectInGame
- void removeItem(int, int)	นำ GameObject ที่ตำแหน่ง x, y ออก
+ void restartGame()	ทำการ clearGame() และ เริ่มใหม่
+ void setObjectInGame(int, int, GameObject)	นำ GameObject ใส่ลงในตำแหน่ง x, y
+ void setSummaryPageAppear(boolean)	แสดงหรือซ่อน gameSummaryPage
- void setTimer(long)	ตรวจและกำหนดค่าให้ remainingTime
Getters and Setters	G gamePage, G minions, G player, G remaning tim, G scene, G,S isPlaying, S startTime

2.2.3 class InputInGame

2.2.3.1 Field

- BitSet keyBoardBitSet	Instance ของ BitSet
- EventHandler<KeyEvent> keyOnPressHandle	ให้ set ค่าของ keyBoardBitSet ตามค่าของปุ่ม ordinal นั้นๆ ให้เป็น true
- EventHandler<KeyEvent> keyOnReleaseHandle	ให้ set ค่าของ keyBoardBitSet ตามค่าของปุ่ม ordinal นั้นๆ ให้เป็น false
- Scene scene	Instance ของ Scene

2.2.3.2 Constructor

+ InputInGame(Scene)	กำหนดค่าให้ scene
----------------------	-------------------

2.2.3.3 Method

+ void addListeners()	ทำการเอา keyOnPressHandle และ keyOnReleaseHandle เข้ามาใส่ให้กับ scene
+ void changeBitSet(KeyCode, boolean)	ทำการแก้ไขค่าตาม keyBoardBitSet ตามตำแหน่ง KeyCode ordinal ด้วยค่า boolean ที่ผ่านมา
+ void clearKeyboardCheck()	ให้ค่าทุกอย่างใน keyBoardBitSet เป็น false
+ boolean isKeyPress(KeyCode)	ทำการตรวจสอบว่ามีการกด KeyCode นั้นๆ อยู่หรือไม่
+ void removeListeners()	ทำการเอา keyOnPressHandle และ keyOnReleaseHandle ออกจาก scene

2.2.4 class LevelGenerator

2.2.4.1 Field

- final int emptyTileCount	จำนวนของช่องว่างที่จะมีในด่าน
- final int obstacleRate	จำนวน Obstacle ที่จะมีในด่าน
- Random random	Instance ของ Random

2.2.4.2 Method

+ ObjectInGame[][] generateLevel()	คืนค่าเป็น array ที่จะเก็บค่าว่าแต่ละช่องเป็น item ชนิดใด
------------------------------------	---

2.2.5 enum ObjectInGame

2.2.5.1 Variable

BOMB	object ชนิดนั้นคือ Bomb
BOMBUPGRADEITEM	object ชนิดนั้นคือ BombUpgradeItem
EMPTY	เป็นช่องเปล่า
LIFEINCREASEITEM	object ชนิดนั้นคือ LifeIncreaseItem
OBSTACLE	object ชนิดนั้นคือ Obstacle
POWERUPGRADEITEM	object ชนิดนั้นคือ PowerUpgradeItem
SHIELDITEM	object ชนิดนั้นคือ Shield
SPEEDUPGRADEITEM	object ชนิดนั้นคือ SpeedUpgradeItem
WALL	object ชนิดนั้นคือ Wall

2.2.6 class StartPageController **extends** Controller

2.2.6.1 Method

Override # Scene createScene()	ทำการสร้างและคืนค่าเป็น StartPage
-----------------------------------	-----------------------------------

2.3 Package gameobject

2.3.1 interface Destroyable

สำหรับ implements object ที่สามารถทำลายได้ หรือลบล้างได้ เช่น player item ต่างๆ obstacle เป็นต้น

2.3.1.1 Method

+ void onObjectIsDestroyed()	สำหรับลบ object เมื่อถูกทำลาย
------------------------------	-------------------------------

2.3.2 abstract class GameObject

จะถูก extends โดย object ต่างๆภายในเกมยกเว้น player

2.3.2.1 Field

# Image image	Instance ของ image
# ImageView imageView	Instance ของ imageView เป็นภาพของ object ที่จะสร้าง
# String imagePath	path ไปถึงรูปของ object ที่จะสร้าง
# Pane layer	Instance ของ layer
# int xPosition	ตำแหน่งในแกน x ของ object
# int yPosition	ตำแหน่งในแกน y ของ object

2.3.2.2 Constructor

+ GameObject(int xPosition, int yPosition, String imagePath, Pane layer)	กำหนดค่าของ xPosition, yPosition, imagePath, layer ตามค่าที่รับเข้ามา สร้าง imageView ตาม imagePath เรียก setImageShow(true) และ setPositionOnScreen()
--	--

2.3.2.3 Method

getter method of layer	
getter method of imageView	
+ void setPositionOnScreen()	ย้าย imageView ไปที่ xPosition, yPosition ของ object นั้นๆ

+ String toString()	คืนค่า imagePath
+ void setImageShow(boolean value)	ถ้า value == true ให้ layer เพิ่ม imageView ถ้า value == false ให้ layer ลบ imageView
all getter & setter method of xPosition, yPosition, ImagePath	

2.3.3 interface Moveable

จะถูก implements โดย object ที่สามารถเคลื่อนที่ได้ เช่น player minion

2.3.3.1 Field

+ final defaultMoveSpeed	ค่า speed เริ่มต้น มีค่าเท่ากับ 2
--------------------------	-----------------------------------

2.3.3.2 Method

+ void move()	สำหรับเคลื่อนที่ของ object นั้นๆ
---------------	----------------------------------

2.3.4 class Obstacle extends GameObject implements Destroyable

คือสิ่งที่กีดขวางผู้เล่นอยู่ มี item อยู่ข้างใน และ สามารถทำลายได้

2.3.4.1 Field

- GameController gameController	Instance ของ gameController
- Item itemInObstacle	Instance ของ itemInObstacle

2.3.4.2 Constructor

+ Obstacle(int xPosition, int yPosition, Pane layer, Item item, GameController gameController)	กำหนดค่าตาม constructor ของ super class โดยช่องของ imagePath ให้ใส่ “obstacle” กำหนดค่าของ Field ตามค่าที่ได้รับเข้ามา
--	---

2.3.4.3 Method

+ void onObjectIsDestroyed()	ใช้เมื่อถูกทำลาย โดยลบตัวเองออกจาก gameController แต่ถ้ามี item อยู่ ต้องเพิ่ม item นั้นลงใน gameController ด้วย
------------------------------	--

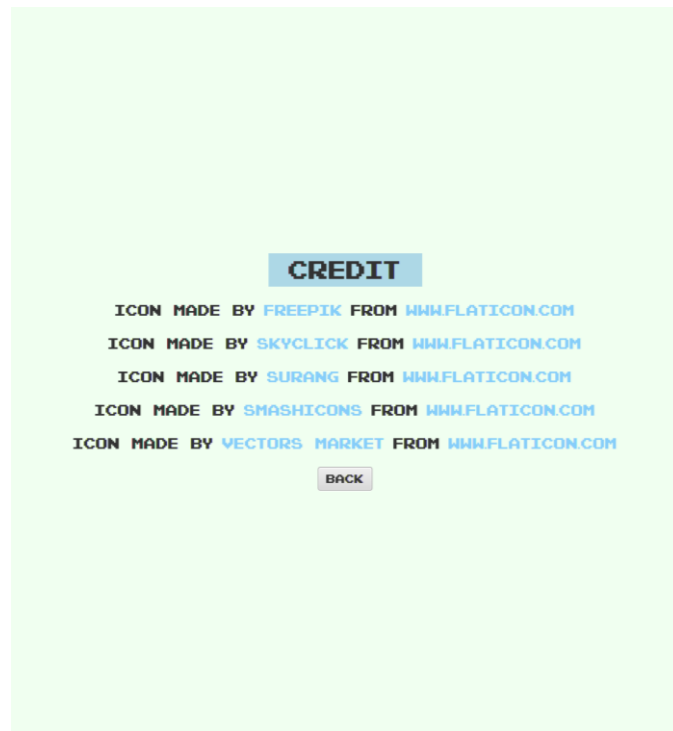
2.3.5 class Wall **extends** GameObject

2.3.5.1 Constructor

+ Wall(int xPosition, int yPosition, Pane layer)	กำหนดค่าตาม constructor ของ superclass โดยช่อง imagepath ให้ใส่ “wall”
+ Wall(int xPosition, int yPosition, String imagePath, Pane layer)	กำหนดค่าตาม constructor ของ superclass

2.4 package gui

2.4.1 Class CreditPage **extends** VBox **implements** HasButtonPage



2.4.1.1 Field

- Button backButton	Instance ของ Button
- StartPage startPage	Instance ของ StartPage

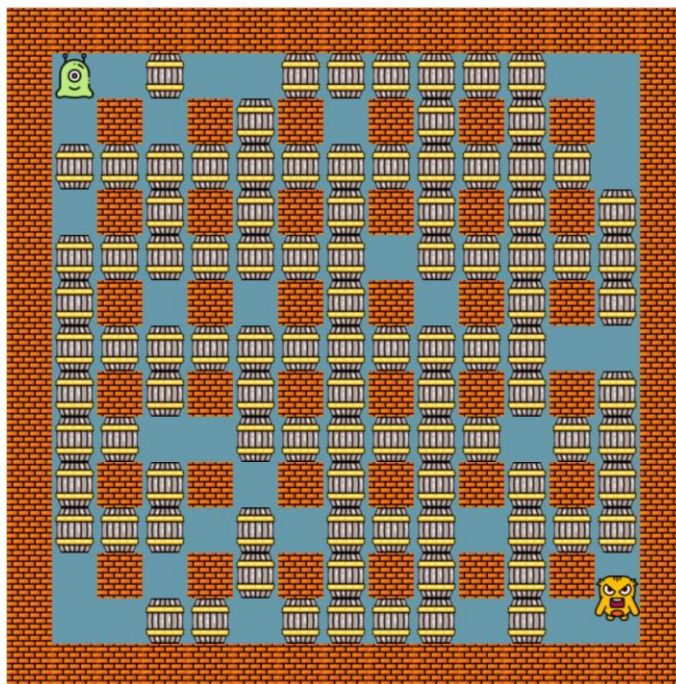
2.4.1.2 Constructor

+ CreditPage(StartPage)	กำหนดค่าให้กับ startPage และทำเรียก CreatePage()
-------------------------	--

2.4.1.3 Method

- Label CreateAuthorLabel(String)	ทำการสร้าง Label จาก String ที่ผ่านเข้ามา
- void CreatePage()	ทำการสร้าง Label ต่างๆ
Override + void setButtonAction()	กำหนด Action ให้กับ backButton ย้อนกลับไปหน้า startPage

2.4.2 class GameField **extends** StackPane



2.4.2.1 Field

- Pane itemLayer	Instance ของ Pane ที่จะใช้วาง item
- Pane playerLayer	Instance ของ Pane ที่จะใช้วาง PlayerBase

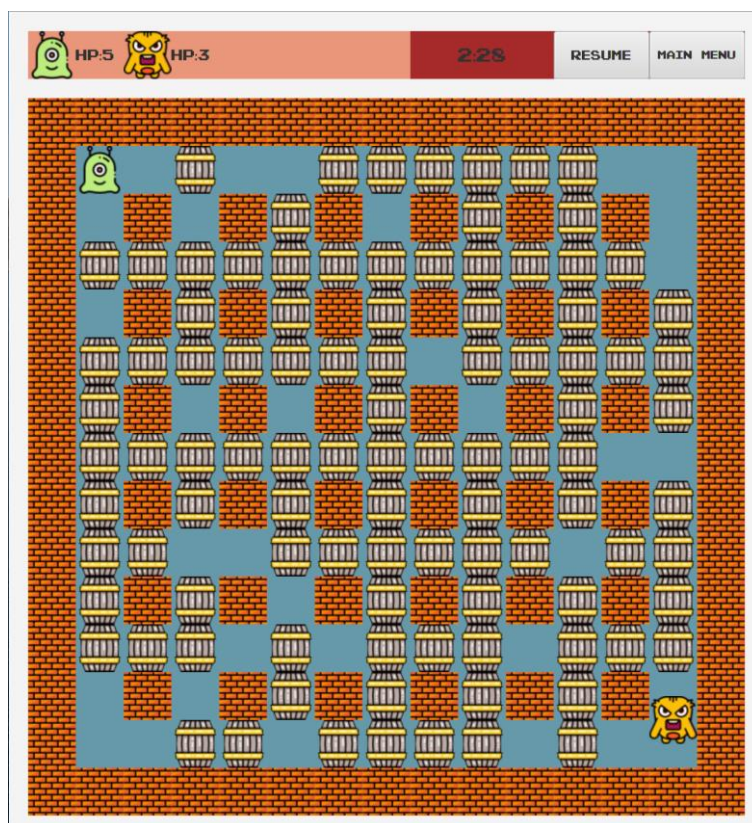
2.4.2.2 Constructor

+ GameField()	ทำการสร้าง itemLayer และ playerLayer
---------------	--------------------------------------

2.4.2.3 Method

Getters	
---------	--

2.4.3 class GamePage **extends** StackPane



2.4.3.1 Field

- GameField gameField	Instance ของ GameField
- ScoreBoard scoreBoard	Instance ของ ScoreBoard

2.4.3.2 Constructor

+ GamePage(GameController)	ทำการสร้าง gameField และ scoreBoard
----------------------------	-------------------------------------

2.4.3.3 Method

+ Pane getGameFieldItemPane()	คืนค่า itemLayer ของ gameField
+ Pane getGameFieldPlayerPane()	คืนค่า playerLayer ของ gameField
Getters	

2.4.4 class GameSummaryPage **extends** FlowPane **implements** HasButtonPage



2.4.4.1 Field

- Button continueButton	Instance ของปุ่มสำหรับไปด่านต่อไปหรือเริ่มใหม่
- GameController gameController	Instance ของ GameController
- Label mainLabel	Instance ของ Label ที่อยู่บรรทัดแรก
- button mainMenuButton	Instance ของปุ่มสำหรับไปหน้าแรก
- Label statusLabel	Instance ของ Label ที่อยู่บรรทัดสอง

2.4.4.2 Constructor

+ GameSummaryPage(GameController)	ทำการสร้างและกำหนดค่าให้ตัวแปร field
--------------------------------------	--------------------------------------

2.4.4.3 Method

Override + void setButtonAction()	ทำการกำหนด Action ให้กับ continueButton แล้ว mainMenuButton
+ void setText(boolean)	ทำการกำหนดข้อความที่จะกำหนดในหน้า

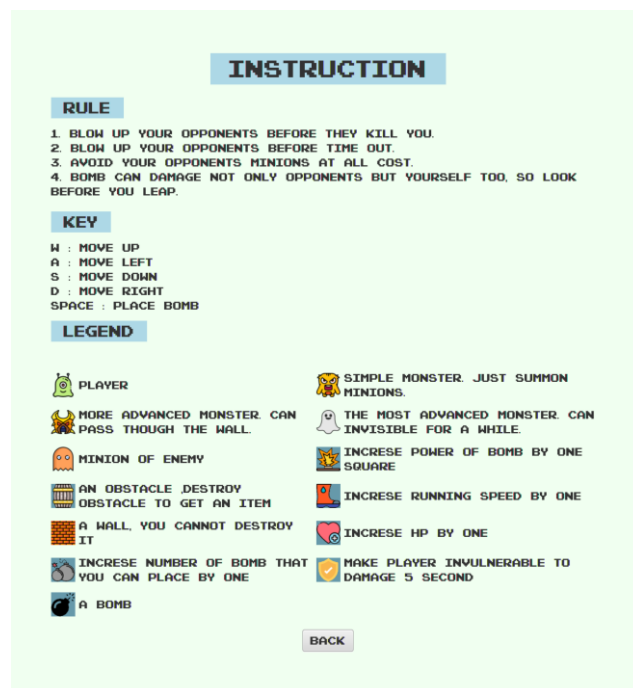
2.4.5 interface HasButtonPage

interface สำหรับ scene ที่มีปุ่ม ซึ่งจะถูกรับใช้โดย CreditPage, GameSummaryPage, InstructionPage, ScoreBoard และ StartPage

2.4.5.1 Method

void setButtonAction()	สำหรับกำหนด action ให้ Button
------------------------	-------------------------------

2.4.6 Class InstructionPage extends VBox implements HasButtonPage



2.4.6.1 Field

- Button backButton	Instance ของ Button ที่เมื่อกดจะกลับไปยัง startPage
- StartPage startPage	Instance ของ startPage

2.4.6.2 Constructor

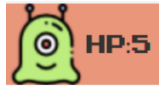
+ InstructionPage(StartPage)	ทำการสร้างและกำหนดค่าให้ตัวแปร field
------------------------------	--------------------------------------

2.4.6.3 Method

- ImageView createImageView(String)	ทำการสร้าง ImageView ตาม String path ที่หาค่ามา
--	---

- GridPane createLegendBox()	ทำการสร้าง GridPane ส่วน Legend
- void CreatePage()	ทำการสร้างหน้า
Override + void setButtonAction()	ทำการกำหนด Action ให้กับ backButton

2.4.7 class PlayerStatusBoard **extends** HBox



2.4.7.1 Field

- Label hpLabel	Instance ของ Label ที่ไว้แสดง HP
- PlayerBase player	Instance ของ PlayerBase ที่ PlayerStatusBoard นี้ เชื่อมด้วย
- ImageView playerImageView	Instance ของ ImageView

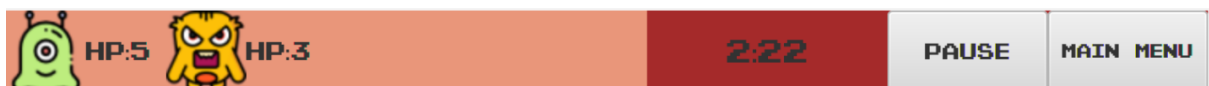
2.4.7.2 Constructor

+ PlayerStatusBoard()	ทำการสร้างและกำหนดค่าให้ตัวแปร field
-----------------------	--------------------------------------

2.4.7.3 Method

+ void linkToPlayer(PlayerBase)	ทำการเชื่อมกับ PlayerBase ที่ผ่านค่ามา
+ void updateStatus()	ทำการ update และ แสดง HP ของ PlayerBase ที่เชื่อมต่ออยู่

2.4.8 class ScoreBoard **extends** HBox **implements** HasButtonPage



2.4.8.1 Field

- Controller controller	Instance ของ Controller
- Button mainMenuButton	Instance ของ Button ที่เมื่อกดจะทำการกลับสู่หน้าแรก
- Button pauseButton	Instance ของ Button ที่เมื่อกดจะทำการหยุดหรือเริ่มเกมต่อ

- PlayerStatusBoard[] playerStatusBoards	Array ที่ใช้เก็บ PlayerStatusBoard
- Label timer	Instance ของ Label ที่ใช้แสดงเวลาที่เหลือในเกม

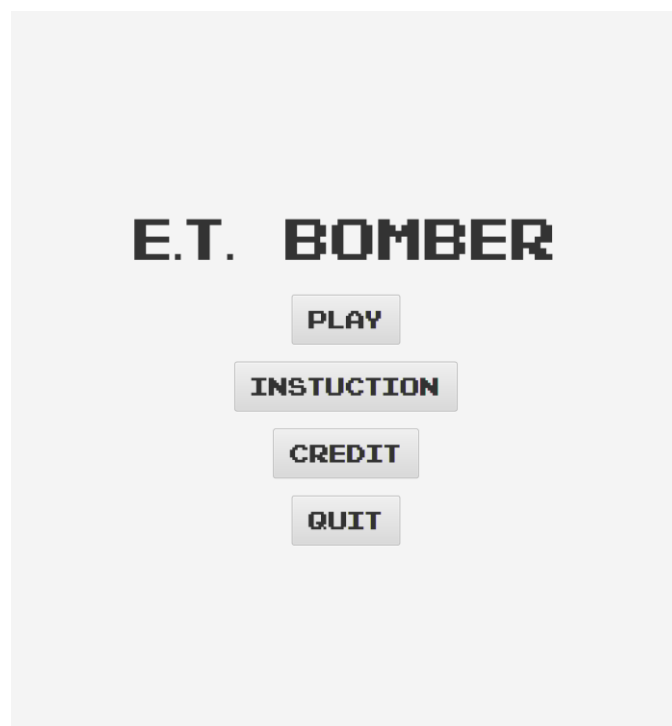
2.4.8.2 Constructor

+ ScoreBoard(Controller)	ทำการสร้างและกำหนดค่าให้ตัวแปร field
--------------------------	--------------------------------------

2.4.8.3 Method

+ void backToMainMenuAction()	ทำให้กลับไปสู่หน้าแรก
+ PlayerStatusBoard getPlayerStatusBoardViaIndex(int)	คืนค่า playerStatusBoards จาก index ที่ใส่ค่าให้
Override + void setButtonAction()	ทำการกำหนด Action ให้กับ Button
+ void setTimer(long)	กำหนดข้อความให้ timer
+ void updateStatus()	ทำการ update ข้อความใน playerStatusBoards

2.4.9 class StartPage extends StackPane implements HasButtonPage



2.4.9.1 Field

- Controller controller	Instance ของ Controller
- Button creditButton	Instance ของ Button ที่เมื่อกดจะเปลี่ยนไปยัง creditPage
- CreditPage creditPage	Instance ของ creditPage
- Button instuctionButton	Instance ของ Button ที่เมื่อกดจะเปลี่ยนไปยัง instuctionPage
- InstuctionPage instuctionPage	Instance ของ instuctionPage
- Button playButton	Instance ของ Button ที่เมื่อกดจะเปลี่ยนไปยัง GamePage
- Button quitButton	Instance ของ Button ที่เมื่อกดจะทำการออกจากเกม

2.4.9.2 Constructor

+ StartPage(Controller)	ทำการสร้างและกำหนดค่าให้ตัวแปร field
-------------------------	--------------------------------------

2.4.9.3 Method

Override + void setButtonAction()	ทำการกำหนด Action ให้ Button
+ void setPageAppear(VBox, boolean)	ทำการกำหนดว่าจะแสดงหรือซ่อน Page ใด

2.5 package item

2.5.1 class BombUpgradeItem **extends** Item **implements** PowerUp

เป็น Item ที่เมื่อ Player เก็บจะทำการวางระเบิดพร้อมกันเพิ่มได้ 1 ลูก

2.5.1.1 Constructor

+ BombUpgradeItem(int xPosition, int yPosition, Pane layer, GameController gameController)	กำหนดค่าตาม constructor ของ superclass โดยช่อง imagePath ให้ใส่ "bombItem"
--	--

2.5.1.2 Method

+ ObjectInGame getObjectInGame()	คืนค่า ObjectInGame.BOMBUPGRADEITEM
+ void onObjectIsDestroyed()	ลบภาพโดยใช้ setImageShow(false) และ ลบ object ออกจาก gameController
+ void onPlayerGetItem(PlayerBase player)	เรียกใช้เมื่อ player มาเก็บ item โดย จะเพิ่ม bombCount ของ player ขึ้นมา 1 พร้อมทั้งเรียก player.setCanUseWeapon()

2.5.2 **abstract** class Item **extends** GameObject **implements** Destroyable

2.5.2.1 Field

- GameController gameController	Instance ของ gameController
---------------------------------	-----------------------------

2.5.2.2 Constructor

+ Item(int xPosition, int yPosition, String imagePath, Pane layer, GameController gameController)	กำหนดค่าตาม constructor ของ superclass พร้อมทั้งเรียกกำหนดค่าของ gameController
---	---

2.5.2.3 Method

+ abstract ObjectInGame getObjectInGame()	ไว้ใช้คืนค่า ObjectInGame ของ object นั้นๆ
getter & setter method of gameController	

2.5.3 class LifeIncreaseItem **extends** Item **implements** PowerUp

เป็น Item ที่เมื่อ Player เก็บ จะเพิ่มเลือดของ Player 1 หน่วย

2.5.3.1 Constructor

+ LifeIncreaseItem(int xPosition, int yPosition, Pane layer, GameController gameController)	กำหนดค่าตาม constructor ของ superclass โดยช่อง imagePath ให้ใส่ "lifeItem"
---	--

2.5.3.2 Method

+ ObjectInGame getObjectInGame()	คืนค่า ObjectInGame.LIFEINCREASEITEM
+ void onObjectIsDestroyed()	ลบภาพโดยใช้ setImageShow(false) และ ลบ object ออกจาก gameController
+ void onPlayerGetItem(PlayerBase player)	เรียกใช้เมื่อ player มาเก็บ item โดย จะเพิ่ม Hp ของ player ขึ้น 1

2.5.4 **interface** PowerUp

จะถูก implements โดย object ที่สามารถเพิ่มพลังบางอย่างให้กับ Player ได้ เมื่อ Player เก็บ object นั้นๆ

2.5.4.1 Method

+ void onPlayerGetItem(PlayerBase player)	ใช้สำหรับเพิ่มพลังให้ player ตามชนิด ของ Item
---	---

2.5.5 class PowerUpgradeItem **extends** Item **implements** PowerUp

เป็น Item ที่เมื่อ Player เก็บ จะทำให้ระยะระเบิดของ Player นั้นเพิ่มขึ้น 1 ช่อง ทั้ง 4 ด้าน

2.5.5.1 Constructor

+ PowerUpgradeItem(int xPosition, int yPosition, Pane layer, GameController gameController)	กำหนดค่าตาม constructor ของ superclass โดยช่อง imagePath ให้ใส่ "powerItem"
---	---

2.5.5.2 Method

+ ObjectInGame getObjectInGame()	คืนค่า ObjectInGame.POWERUPGRADEITEM
+ void onObjectIsDestroyed()	ลบภาพโดยใช้ setImageShow(false) และ ลบ object ออกจาก gameController

+ void onPlayerGetItem(PlayerBase player)	เรียกใช้เมื่อ player มาเก็บ item โดย จะเพิ่ม BombRange ของ player ขึ้น 1
---	--

2.5.6 class Shield **extends** Item **implements** PowerUp

เป็น Item ที่เมื่อ Player เก็บ จะทำให้ Player เป็นอมตะ 5 วินาที

2.5.6.1 Constructor

+ Shield(int xPosition, int yPosition, Pane layer, GameController gameController)	กำหนดค่าตาม constructor ของ superclass โดยช่อง imagePath ให้ใส่ "shield"
---	--

2.5.6.2 Method

+ ObjectInGame getObjectInGame()	คืนค่า ObjectInGame.SHIELDITEM
+ void onObjectIsDestroyed()	ลบภาพโดยใช้ setImageShow(false) และ ลบ object ออกจาก gameController
+ void onPlayerGetItem(PlayerBase player)	เรียกใช้เมื่อ player มาเก็บ item โดย ให้เรียก player.setShield()

2.5.7 class SpeedUpgradeItem **extends** Item **implements** PowerUp

เป็น Item ที่เมื่อ Player เก็บ จะทำให้ Player เคลื่อนที่ไวยิ่งขึ้น

2.5.7.1 Constructor

+ SpeedUpgradeItem(int xPosition, int yPosition, Pane layer, GameController gameController)	กำหนดค่าตาม constructor ของ superclass โดยช่อง imagePath ให้ใส่ "speedItem"
---	---

2.5.7.2 Method

+ ObjectInGame getObjectInGame()	คืนค่า ObjectInGame.SPEEDUPGRADEITEM
+ void onObjectIsDestroyed()	ลบภาพโดยใช้ setImageShow(false) และ ลบ object ออกจาก gameController
+ void onPlayerGetItem(PlayerBase player)	เรียกใช้เมื่อ player มาเก็บ item โดย จะเพิ่ม Speed ของ player ขึ้น 1 และแนะนำควรตั้งขีดจำกัดของ Speed ไว้ให้มีค่าเท่ากับ 5 เพื่อความง่ายในการเล่น

2.6 package main

2.6.1 class Main **extends** Application

2.6.1.1 Field

- GameController gameController	Instance ของ GameController
- StartPageController startPageController	Instance ของ startPageController

2.6.1.2 Method

+ void main(String[] args)	เรียก launch(args)
+ void start(Stage primaryStage)	สร้าง startPageController และ gameController สร้าง Stage ที่ไม่สามารถเปลี่ยนขนาดได้ โดยใช้ Field ทั้งสองตัว
+ void stop()	หยุดโปรแกรม และออกจากโปรแกรม

2.7 package player

2.7.1 class Player **extends** PlayerBase

2.7.1.1 Constructor

+ Player(int, int, String, Pane, int, GameController)	เรียก Constructor ของ PlayerBase
---	----------------------------------

2.7.2 **abstract** class PlayerBase **extends** GameObject **implements** Moveable, Destroyable

2.7.2.1 Field

# int bombCount	เก็บค่าจำนวนระเบิดที่สามารถวางได้
# int bombRange	เก็บค่าความไกลของระเบิดเป็นช่อง
# boolean canUseWeapon	เก็บค่าว่าสามารถใช้อาวุธได้หรือไม่
# Queue<Bomb> countBomb	เก็บค่าว่าได้วางระเบิดไปตอนไหน
# PlayerState currentPlayerState	เก็บค่า PlayerState
# GameController gameController	instance ของ GameController
# int hp	เก็บค่า hp
# int playerNumber	เก็บค่าว่าเป็นผู้เล่นหมายเลขเท่าไร
- final long shieldDuration	เก็บค่าระยะเวลาของ item shield
# long shieldTime	เก็บค่าระยะเวลาที่ shield จะหมดไป
# int speed	เก็บค่าความเร็วในการเคลื่อนที่

2.7.2.2 Constructor

+ PlayerBase(int, int, String, Pane, int, GameController)	ทำการเรียก Constructor ของ GameObject จากนั้นจึงกำหนดค่าให้ playerNumber และ gameController และเรียก method setDefaultPlayer()
---	--

2.7.2.3 Method

+ boolean isDead()	คืนค่าว่า currentPlayerState เท่ากับ DEAD หรือไม่
+ boolean isHasShield()	คืนค่าว่า เวลาปัจจุบันน้อยกว่า shieldTime หรือไม่ หากน้อยกว่าแสดงว่ายังมี shield อยู่
Override + void move()	เคลื่อนที่ตามทิศทางต่างๆ ที่ค่า currentPlayerState เก็บไว้
Override + onObjectIsDestroyed()	ลบรูปออกจากเกม
# void setDefaultPlayer()	กำหนดค่าพื้นฐานให้กับ PlayerBase
+ void useWeapon()	ทำการวางระเบิด
Getters and Setters	

2.7.3 enum PlayerState

2.7.3.1 Variable

DEAD	player hp เท่ากับ 0
IDLE	ไม่มีการเคลื่อนที่
MOVEDOWN	เคลื่อนที่ลงด้านล่าง
MOVELEFT	เคลื่อนที่ไปด้านซ้าย
MOVERIGHT	เคลื่อนที่ไปด้านขวา
MOVEUP	เคลื่อนที่ขึ้นด้านบน

2.8 Package setting

2.8.1 Class Setting

2.8.1.1 Field

+ int GAME_TIME	จำนวนวินาทีในการเล่น 1 ด่าน
+ KeyCode MOVEUP_KEY	ปุ่มเดินขึ้น
+ KeyCode MOVEDOWN_KEY	ปุ่มเดินลง
+ KeyCode MOVELEFT_KEY	ปุ่มเดินไปทางซ้าย
+ KeyCode MOVERIGHT_KEY	ปุ่มเดินไปทางขวา
+ KeyCode PLACEBOMB_KEY	ปุ่มวางระเบิด
+ String PATH_TO_PLACEHOLDER	ข้อความที่บอก address ของ รูป placeholder
+ double SCENE_HEIGHT	ความสูงของหน้าต่างโปรแกรม
+ double SCENE_WIDTH	ความกว้างของหน้าต่างโปรแกรม

2.9 Package weapon

2.9.1 Class Bomb **extends** GameObject **implement** Weapon, Destroyable

2.9.1.1 Field

- int range	ระยะระเบิดของลูกระเบิด
- GameController gameController	Instance ของ gameController
- Thread explodeThread	Instance ของ explodeThread ใช้ในช่วงที่ก่อนระเบิด
- Thread bombAreaThread	Instance ของ bombAreaThread ใช้ใน ช่วงแสดงระยะของระเบิด
- BombArea area	Instance ของ ระยะระเบิด

2.9.1.2 Constructor

+ Bomb (int xPosition, int yPosition, Pane layer, int range, PlayerBase player, GameController gameController)	กำหนดค่า range จากค่าที่รับมา, กำหนดค่า gameController จากค่าที่รับมา, สร้าง explodeThread โดยให้ระยะเวลาก่อนระเบิดเป็น 1.5 วินาที สร้าง area หลังจากทีระเบิด และแสดงระยะระเบิด จากนั้นสร้าง bombAreaThread เพื่อค้ำระยะระเบิดไว้เป็นเวลา 0.5 วินาที โดยให้ bombAreaThread ทำงานหลังจาก explodeThread เท่านั้น ให้ thread ทั้งสองทำงานเมื่อสร้าง object ทันที
--	---

2.9.1.3 Method

+ boolean canMakeDamageToobject (GameObject targetobj)	ถ้า targetobj คือ Wall ให้คืนค่า false ไม่อย่างนั้นให้คืนค่า true
+ int getDamageRange()	คืนค่า range ของระเบิด
+ int getGameController()	คืนค่า gameController
+ ObjectInGame getObjectInGame()	คืนค่า ObjectInGame.BOMB

+ boolean isinRange(int x, int y, int xMin, int xMax, int yMin, int yMax)	รับพิกัด x , y เพื่อตรวจสอบว่า ค่า x, y นั้นอยู่ในระยะระเบิดหรือไม่ โดย xMin xMax yMin yMax คือระยะระเบิดในด้านต่างๆ
+ void makeDamageToObject()	ทำความเสียหายต่อ Object ในเกม ถ้า Object นั้นไม่ใช่ Wall โดยเรียกใช้ ((Destroyable) object).onObjectIsDestroyed() โดยใช้ Field boolean ของ area ในการตรวจสอบว่า มี object ในระยะระเบิดแต่ละด้านหรือไม่
+ void makeDamageToPlayer()	ใช้เพื่อทำความเสียหายต่อ Player และ Minion โดยรับพิกัด x,y ทั้ง 4 มุม ของ player นั้นๆ มาตรวจสอบว่า แต่ละมุมอยู่ในระยะของระเบิดหรือไม่หาระยะของระเบิดได้จาก Field boolean ของ area และใช้ isinRange method
+ void onObjectIsDestroyed()	ลบภาพของระเบิดออก ลบระเบิดใน array ของ gameController จากนั้น explodeThread.interrupt()

2.9.2 Class BombArea

Field ส่วนมากของ class นี้จะถูกกำหนดค่าใน class Bomb

2.9.2.1 Field

- int[] afterDestroy	เก็บระยะของระเบิดหลังจากทำลาย Object แล้ว
- ArrayList<ImageView> allImageViews	arraylist ที่เก็บภาพระยะระเบิดทั้งหมดไว้
- Bomb bomb	Instance ของ bomb
- boolean[] isCanShowTop	array ที่เก็บว่าช่องด้านบนของระเบิดแสดงระยะได้หรือไม่
- boolean[] isCanShowBot	array ที่เก็บว่าช่องด้านล่างของระเบิดแสดงระยะได้หรือไม่

- boolean[] isCanShowLeft	array ที่เก็บว่า ช่องด้านบนของระเบิดแสดงระยะได้หรือไม่
- boolean[] isCanShowRight	array ที่เก็บว่า ช่องด้านล่างของระเบิดแสดงระยะได้หรือไม่
- int range	เก็บระยะระเบิด
- int xPos	เก็บระยะ X ของ ระเบิด
- int yPos	เก็บระยะ y ของ ระเบิด

2.9.2.2 Constructor

+ BombArea(Bomb bomb)	กำหนดค่า xPos, yPos, bomb, range ตามเงื่อนไขข้างต้น จากนั้นให้ทุกๆ boolean[] มีช่องเท่ากับ range และ afterDestroy มี 4 ช่อง โดยให้แต่ละช่องมีค่าเท่ากับ -1
-----------------------	--

2.9.2.3 Method

+ Bomb getBomb()	คืนค่า bomb
+ int getRange()	คืนค่า range
all getter Method of boolean[] Field	คืนค่า boolean[] แต่ละตัว
+ void removeRange()	ให้ layer ลบรูปทั้งหมดที่อยู่ใน allImageViews
+ void setAfterDestroy()	ใช้ afterDestroy แก้ไขค่า boolean[] แต่ละตัว โดย ช่องที่ 0 แทนด้านบน ช่องที่ 1 แทนด้านล่าง ช่องที่ 2 แทนด้านซ้าย ช่องที่ 3 แทนด้านขวา
+ void setBomb(Bomb bomb)	กำหนดค่า bomb ตามที่รับค่ามา
+ void setIsCanShowRange()	กำหนดค่าของ boolean[] แต่ละตัวโดยใช้ setIsCanShowBySite(Site site)
+ boolean[] setIsCanShowBySite(Site site)	คืนค่า boolean ที่แสดงถึงระยะของระเบิด ว่าสามารถแสดงระยะได้หรือไม่ โดยถ้าเป็นช่องว่าง

	ให้ชื่อนั้นเป็น true ถ้ามี Object ขวาง ให้ชื่อนั้นเป็น false อีกทั้งยังให้กำหนดค่าให้ afterDestroy โดยให้ตัวเลขที่เป็น false ค่าแรก
+ void setRange(int range)	กำหนดค่า range ตามที่รับเข้ามา
+ void showRange()	แสดงระยะของระเบิดทั้ง 4 ด้าน โดยใช้ showRangeBySite(Site site) พร้อมทั้งแสดงระยะระเบิดที่ช่องที่ระเบิดวางอยู่
+ void showRangeBySite(Site site)	แสดงระยะของระเบิดโดยแบ่งกรณีตาม Site ที่รับเข้ามา

2.9.3 Enum Site

2.9.3.1 Variable

TOP	ด้านบนของระเบิด
BOT	ด้านล่างของระเบิด
LEFT	ด้านซ้ายของระเบิด
RIGHT	ด้านขวาของระเบิด

2.9.4 Interface Weapon

จะถูก implements โดย Bomb

2.9.4.1 Method

+ int getDamageRange();	สำหรับคืนค่า ระยะระเบิด
+ boolean canMakeDamageToobject(GameObject targetobj);	สำหรับตรวจสอบว่าสามารถทำความเสียหายต่อ targetobj ได้หรือไม่
+ void makeDamageToObject();	สำหรับทำความเสียหายต่อ Object
+ void makeDamageToPlayer();	สำหรับทำความเสียหายต่อ Player
+ abstract ObjectInGame getObjectInGame();	สำหรับคืนค่า ObjectInGame ของ Weapon นั้นๆ