

Bullet GOD Game Documentation

เกม Bullet GOD เป็นเกมแนว Platformer สำหรับการเล่น 2 คนที่จะให้ผู้เล่นควบคุมยามอวกาศคนละลำที่อยู่ซ้าย(ยานสีน้ำเงิน)และขวาของจอเกม(สีแดง) โดยเกมจะได้ออกแบบเป็น 5 ช่องในการยิงและเคลื่อนไหว ให้ผู้ทำลายของผู้เล่นอีกคนโดยการยิงให้โดนอีกฝั่ง เกมได้รับแรงบันดาลใจจากเกมอย่าง Rockman และ SpaceInvader โดยเกมจะเน้นไปที่จังหวะกดที่รวดเร็ว,ไหวพริบในการหลบหลีกและ การตอบสนองในการกดของผู้ โดยเฉพาะ(เรียกว่าเป็นเกม ที่เน้นไป APM(Action per minute) กี่ว่าได้) ซึ่งในเกมก็ยังมีระบบต่างๆ ดังนี้

1. ระบบเกี่ยวกับผู้ใช้
 - 1.1. หน้าจอแสดงผล
 - 1.2. ปุ่มแสดงผล
2. ระบบการเล่น
 - 2.1. การยิงและเคลื่อนที่
 - 2.2. กระสุนและไอเทม
 - 2.3. แทบการแสดงผลสถานะ
 - 2.4. การแสดงผล
3. Implement Details
 - 3.1. defaultProgressBar
 - 3.2. ButtonPress
 - 3.3. ButtonChageStage
 - 3.4. QuitButton
 - 3.5. Timer
 - 3.6. SpaceShip
 - 3.7. BombAble
 - 3.8. ResouceLoader
 - 3.9. GameMenu
 - 3.10. Main
 - 3.11. Bullet
 - 3.12. SpeedBullet
 - 3.13. Potion
 - 3.14. Event Manager

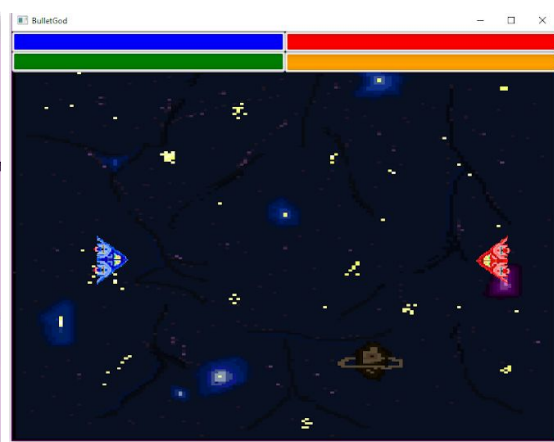
1. ระบบเกี่ยวกับผู้ใช้

1.1. หน้าจอแสดงผล

สำหรับระบบการแสดงผลจะแสดงผลเกมในรูปแบบเกมภาพ 8 bits(ซึ่งผู้พัฒนาทำการวาดและออกแบบเอง) และเน้นการตัดสีกันของภาพ



ภาพ 1.1 ภาพในเกม



ภาพ 1.2 ภาพในเกม



ภาพ 1.3 ภาพในเกม



ภาพ 1.4 ภาพในเกม

1.2 ปุ่มแสดงผล

สำหรับปุ่มกดจะใช้เป็นส่วนในการกดเพื่อไปยังหน้าต่างๆ ของเกม มีดังนี้

1. ปุ่ม Start

ทำการเริ่มเกม และไปยังหน้าจอเล่นเกมตามภาพ 1.2 เมื่อเอาเมาส์ไปใกล้จะมีขอบหนา และเมื่อเอาเมาส์ออกก็จะกลับไปเป็นเหมือนเดิม



2. ปุ่ม ReStart

จะปรากฏขึ้นมาหลังจากจบเกม ตามลำดับ ภาพ 1.3, 1.4 เมื่อเอาเมาส์ไปใกล้จะมีขอบหนา และเมื่อเอาเมาส์ออกก็จะกลับไปเป็นเหมือนเดิม



3. ปุ่ม Quit

ทำการปิดเกม ปุ่มจะอยู่ตามภาพ 1.1 , 1.3, 1.4 เมื่อเอาเมาส์ไปใกล้จะมีขอบหนาและเมื่อเอาเมาส์ออกก็จะกลับไปเป็นเหมือนเดิม



2.ระบบการเล่น

2.1 การยิงและการเคลื่อนที่

การยิงและการเคลื่อนที่นั้นจะทำการกดปุ่มโดยผู้เล่นจะมีปุ่มโดยแยกตามสีของยาน และสำหรับการเคลื่อนที่จะเคลื่อนที่ได้ทั้งหมด 5 ช่องการยิง และทุกการกดจะมีการกินพลังงานของยานตามแถบแสดงสถานะ (สีเขียวของยาน สีน้ำเงิน สีส้มของยานสีแดง) ภาพ 2.3.1 โดยมีการควบคุมดังนี้

2.1.1 ยานสีน้ำเงินทางซ้าย (ตามภาพ 1.2) จะมีการควบคุมคือ

2.1.1.1 การเคลื่อนไหว

ใช้ปุ่ม W ของ Keyborad ในการเคลื่อนที่ขึ้น
ใช้ปุ่ม S ของ Keyborad ในการเคลื่อนที่ลง

2.1.1.2 การยิง

ใช้ปุ่ม D ของ Keyborad ในการยิงกระสุนสีน้ำเงินออกไป

2.1.2 ยานสีแดงทางขวา (ตามภาพ 1.2) จะมีการควบคุมคือ

2.1.1.1 การเคลื่อนไหว

ใช้ปุ่มลูกศร UP(ข้างปุ่ม Ctrl ทางขวา)ของ Keyborad ในการเคลื่อนที่ขึ้น

ใช้ปุ่มลูกศร DOWN(ข้างปุ่ม Ctrl ทางขวา) ของ Keyborad ในการเคลื่อนที่ลง

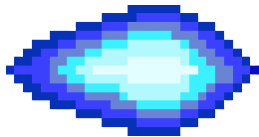
2.1.1.2 การยิง

ใช้ปุ่มลูกศร LEFT(ข้างปุ่ม Ctrl ทางขวา) ของ Keyborad ในการยิงกระสุนสีแดงออกไป

2.2 กระสุนและไอเทม

2.2.1 กระสุน

ในเกมกระสุนจะเกิดเมื่อมีการกดปุ่มยิงดังข้อ 2.1 ที่กล่าวมา สำหรับการสุ่มเมื่อยิงมาและกระทบกันจะเกิดการระเบิดโดยสีของระเบิดนั้นจะมีการเปลี่ยนแบบสุ่มและหายไปตามภาพ 2.2.4 ส่วนเมื่อกระทบกับยานสีตรงข้ามก็จะสร้างความเสียหายและจะมีการแสดงผลที่หน้าจอเป็นสี แล้วทำให้หลอดเลือดลดลง และเกิดภาพ 2.2.4



ภาพ 2.2.1 ภาพกระสุนสีน้ำเงิน

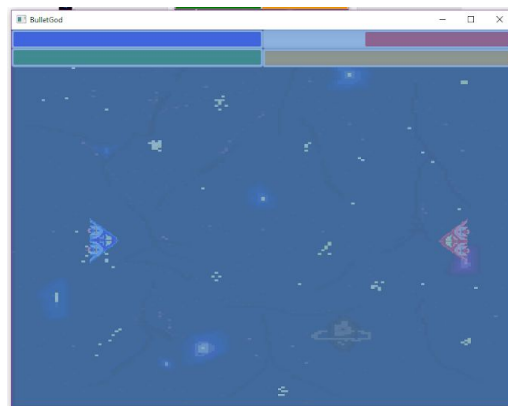


ภาพ 2.2.2 ภาพกระสุนสีแดง



ภาพ 2.2.3 ระเบิดภาพ

***สำหรับภาพ 2.2.3 นั้นเป็นเพียงตัวอย่างเนื่องจากการออกแบบให้สีที่เป็นแบบสุ่มและเน้นสีสัน



ภาพ 2.2.4 กระสุนกระทบจากยานสีฟ้าและทำความเสียหายกับยานสีแดง

6031050121 รวิภาส อภิกุลวณิช

6031052421 วรวิทย์พล ศรีพรม



ภาพ 2.2.5 เมื่อยานถูกยิงจนแถบเลือดหมด

2.2.2 ไอเทม

ในเกมนั้นไอเทมจะมีการออกแบบให้มีการเกิดแบบสุ่มใน 5 ช่อง โดยสุ่มจากการเคลื่อนที่และการยิง ของ ยานทั้ง 2 และสำหรับการเคลื่อนที่จะเคลื่อนที่ได้ทั้งหมด 5 ช่องการยิงและเมื่อถูกยิงจะมีผลพิเศษจากการระเบิดเป็นแบบ เดียวกับกระสุนดังภาพ 2.2.3 และ**ผลพิเศษนี้สามารถทำงานร่วมกันได้** ดังภาพ ภาพ 2.2.2.7, 2.2.2.8 ,2.2.2.9 และ 2.2.2.10 สำหรับไอเทมและผลพิเศษของไอเทมมีดังนี้ โดยไอเทมทุกชนิด จะเพิ่มพลังงานของผู้ที่เก็บได้

2.2.2.1 ดัง S

เมื่อยิงโดนจะมีผลพิเศษทำให้กระสุนที่ยิงจะถูกบีบอัดให้เล็กลงและยิงออกมาด้วยความเร็วสูง และยังสามารถ ทำความเสียหายได้เหมือนเดิม นอกจากนี้ยังความสามารถยิงกระสุนที่มีความถี่สูงขึ้นอีกด้วย



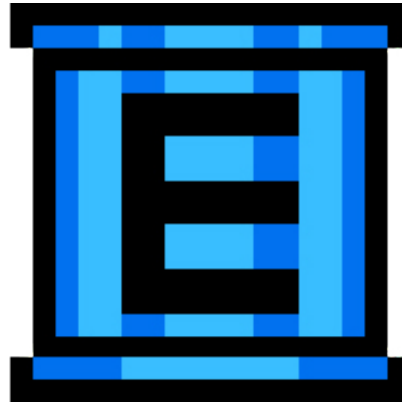
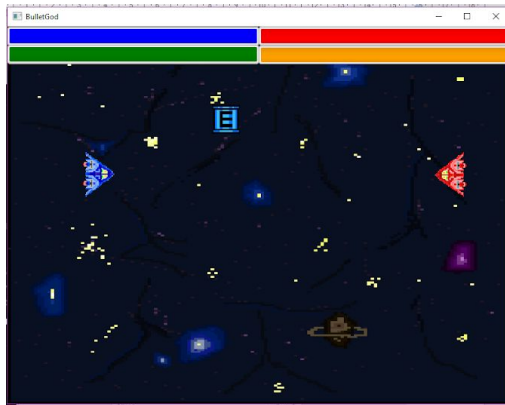
ภาพ 2.2.2.1 ดัง S ในฉาก



ภาพ 2.2.2.2 ยานสีแดงที่รับผลของ ดัง S

2.2.2.2 **ถึง E**

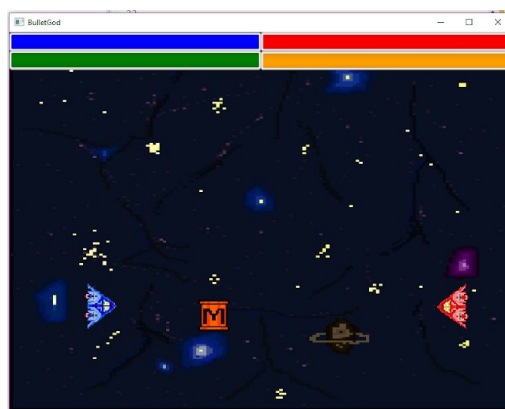
เมื่อยิงโดนจะมีผลพิเศษทำให้กระสุนที่ยิงจะถูกได้อย่างไม่จำกัด



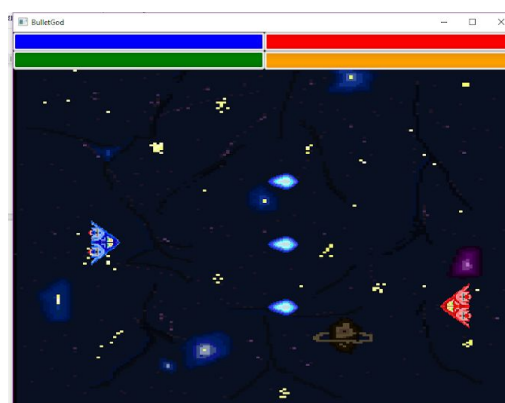
ภาพ 2.2.2.3 ถึง E ในฉาก

2.2.2.3 **ถึง M**

เมื่อยิงโดนจะมีผลพิเศษทำให้กระสุนที่ยิงออกมานั้นจะกระจายไปเป็นสามทาง ภาพ 2.2.2.5 และยังสามารถทำความเสียหายได้เหมือนเดิม



ภาพ 2.2.2.5 ถึง M ในฉาก



ภาพ 2.2.2.6 กระสุนที่รับผลของ ถึง M

ตัวอย่าง ผลพิเศษของไอเทมเมื่อรวมกัน



ภาพ 2.2.2.7 ถึง M กับ ถึง S รวมกันของยานสีแดง



ภาพ 2.2.2.8 ถึง M กับ ถึง E รวมกันของยานสีแดง



ภาพ 2.2.2.9 ถึง S กับ ถึง E รวมกันของยานสีแดง



ภาพ 2.2.2.10 ถึง M ถึง S และถึง E รวมกันของยานสีน้ำเงิน

2.3 แบบการแสดงสถานะ

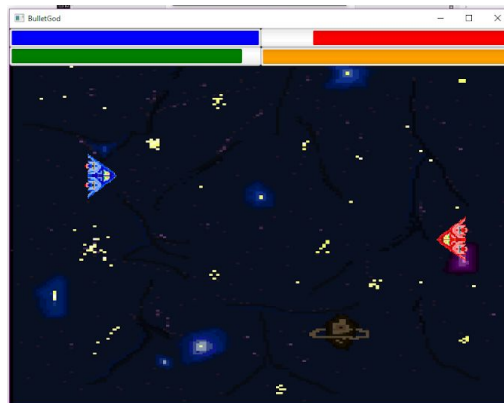
แสดงสถานะต่างๆ ของยานโดย แบ่งตามยานทั้งสองลำตามลำดับดังนี้

2.3.1 ยานสีฟ้า

จะใช้แถบการแสดงผลสถานะในแถบสีน้ำเงิน เป็นเลือดของยานเมื่อหมดจะมีการผลพิเศษขึ้นที่จอและตัดภาพไปยัง ภาพ และแถบสีเขียวเป็นพลังงานของยานที่ส่งผลต่อกระสุนและการเคลื่อนที่ ถ้าหมดจะไม่สามารถยิงหรือเคลื่อนที่ได้ เมื่อแถบหมดจะแสดงตามลำดับ

2.3.2 ยานสีแดง

จะใช้แถบการแสดงผลสถานะในแถบแดง เป็นเลือดของยานเมื่อหมดจะมีการผลพิเศษขึ้นที่จอและตัดภาพไปยัง ภาพ และแถบส้มเป็นพลังงานของยานที่ส่งผลต่อกระสุนและการเคลื่อนที่ ถ้าหมดจะไม่สามารถยิงหรือเคลื่อนที่ได้



ภาพ 2.3.1 แถบแสดงสถานะ

2.4 การแสดงผล

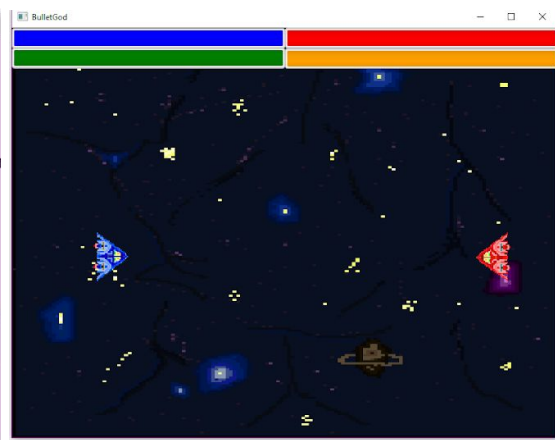
เมื่อโปรแกรมทำการเปิดขึ้นมาจะมีการเล่นเพลง(ผู้พัฒนาสร้างขึ้นเอง) ในระหว่างตลอดทั้งการเล่นเกม ตัวอย่างการแสดงผลตามลำดับของโปรแกรม

2.4.1 เมื่อเริ่มเกม กด -> Start -> หน้าเกม -> เมื่อแพ้ -> เปลี่ยนฉาก -> หน้าแสดงผู้ชนะ

****ในกรณีให้ Blue ชนะ****



1.Start



2.หน้าเกม



3.เปลี่ยนฉาก

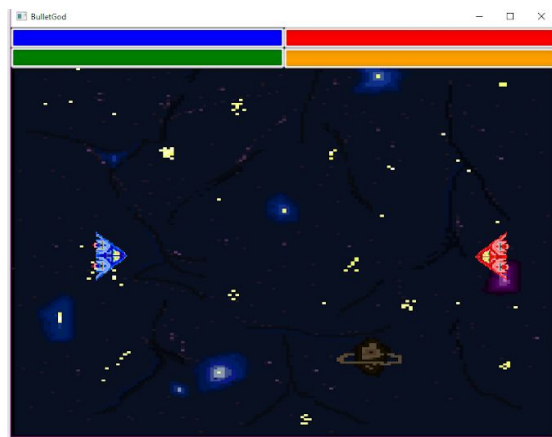


4. หน้าแสดงผู้ชนะ

2.4.2 เริ่มใหม่ หน้าแสดงผู้ชนะ -> ReStart -> หน้าเกม -> เมื่อแพ้ -> เปลี่ยนฉาก -> หน้าแสดงผู้ชนะ
ในกรณีให้ Red ชนะ



1.ReStart



2.หน้าเกม



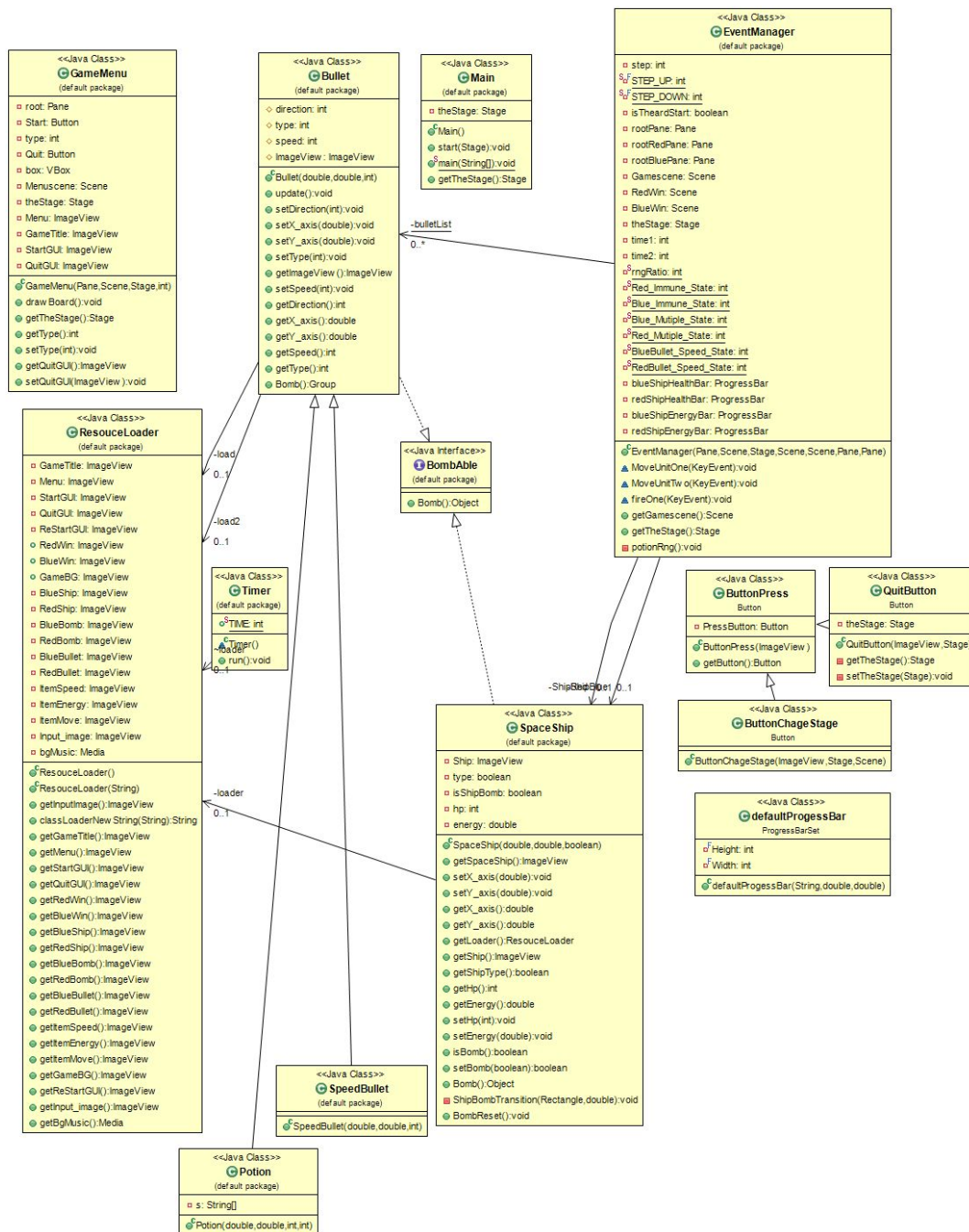
3.เมื่อแพ้



4. หน้าแสดงผู้ชนะ

3. Implement Detail

รูปแบบการจัดการ Class เป็นดังภาพ 3.1 และสำหรับ private และ ค่าอื่นๆอยู่ในภาพทั้งหมดแล้ว สำหรับ Class มีทั้งหมด 14 Class เป็น Class ปกติ 9 ,Inhetited 4 Class และมี Class interface 1 Class



3.1 UML

- + (public)
- # (protected)
- (private)

final will be Bold

static will be underlined.

abstract will be italic.

Package ProgressBarSet

3.1 Class defaultProgressBar extends ProgressBar

3.1.1 Field

- int Height	ประกาศตัวแปร Height เป็น final เก็บความสูงของ ProgressBar ที่จะสร้างเท่ากับ 30
- int Width	ประกาศตัวแปร Width เป็น final เก็บความยาวของ ProgressBar ที่จะสร้างเท่ากับ 400

3.1.2 Constructor

+ defaultProgressbar (String Style,double xLayout,double yLayout)	ทำการเริ่ม defaultProgressbar โดยการสร้าง ProgressBar ที่มี setProgress = 1.0 setPrefWidth , setPrefHeight ทำการ setStyle , setLayoutX , setLayoutY ตามตัวแปรที่ได้รับมา
-------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Package , Button

3.2 Class ButtonPress extends Button

3.2.1 Field

- Button PressButton	ประกาศตัวแปร Button
----------------------	---------------------

3.2.2 Constructor

+ ButtonPress(ImageView image)	ทำการเริ่ม defaultProgressbar โดยการสร้างรับ Imageview มา setGraphic เพื่อสร้างปุ่ม
--------------------------------	-------------------------------------------------------------------------------------

3.2.3 Methods

+ Button getButton()	ทำการคืนค่า Button
----------------------	--------------------

3.3 Class ButtonChageStage extends ButtonPress

3.3.1 Constructor

+ ButtonChageStage(ImageView image,Stage theStage,Scene scene)	ทำการเริ่ม ButtonChageStage สร้างปุ่มและ Add EventHandler ที่สามารถทำการเปลี่ยน Scene ของเกมได้
----------------------------------------------------------------	-------------------------------------------------------------------------------------------------

3.4 Class QuitButton extends ButtonPress

3.4.1 Field

- Stage theStage	ประกาศตัวแปร theStage
------------------	-----------------------

3.4.2 Constructor

+ QuitButton(ImageView image,Stage theStage)	ทำการเริ่ม QuitButton สร้างปุ่มและ Add EventHandlerที่สามารถปิดเกมได้
----------------------------------------------	-----------------------------------------------------------------------

3.4.3 Methods

- Stage getTheStage()	ทำการคืนค่า getTheStage
- Stage setTheStage(Stage theStage)	ทำการตั้งค่า TheStage

3.5 Class Timer extends Thread

3.5.1 Field

+ int TIME	ประกาศตัวแปร TIME
------------	-------------------

3.5.2 Methods

+ void run()	เป็นตัวกำหนดเวลาให้กับ Programและทำการรันเพลง
--------------	-----------------------------------------------

3.6 Class SpaceShip implements BombAble

3.6.1 Field

- ImageView Ship	ประกาศตัวแปร Ship
- boolean type	ประกาศตัวแปร type
- ResouceLoader loader	ประกาศตัวแปร loader
- boolean isShipBomb	ประกาศตัวแปร isShipBomb
- int hp	ประกาศตัวแปร hp
- double energy	ประกาศตัวแปร energy

3.6.2 Constructor

+ SpaceShip(double X_axis, double Y_axis, boolean ShipType)	ทำการเริ่ม SpaceShip สร้างรูปยานอวกาศตาม ShipType ที่รับ และทำการ set ตำแหน่งยาน ค่า hp ,ค่า energy และ ค่า isShipBomb
-------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------

3.6.3 Methods

+ Group Bomb()	คืนค่าเป็น Group ของ Rectangle ของระเบิดโดย เอฟเฟคจะใช้ Method ShipBombTransition เป็น ตัวช่วย และ ตั้งค่า isShipBomb เป็น True
- void ShipBombTransition(final Rectangle rect, final double millis)	ทำการจัดการกับ FadeTransition ของ Rectangle ในMethod Group Bomb()
+ void BombReset()	ทำ Reset ค่าของ Hp,Energy และตำแหน่ง X ,Y ของ Ship
+ getSpaceShip()	ทำการคืนค่า Ship
+ void setX_axis(double x_axis)	ทำการตั้งค่าตำแหน่ง X ของ Shipโดยรับค่าจาก จาก ภายนอก
+ void setY_axis(double y_axis)	ทำการตั้งค่าตำแหน่ง Y ของ Shipโดยรับค่าจาก จาก ภายนอก
+ double getX_axis()	ทำการคืนค่า ตำแหน่ง X ของ Ship
+ double getY_axis()	ทำการคืนค่า ตำแหน่ง Y ของ Ship
+ Getter & Setter method for every field except setter of Ship , type	

3.7 Class interface BombAble

3.7.1 Methods

+ Group Bomb()	
----------------	--

3.8 Class ResouceLoader

3.8.1 Field

- ImageView GameTitle, Menu, StartGUI, QuitGUI, ReStartGUI RedWin, BlueWin, GameBG BlueShip, RedShip, BlueBomb, RedBomb, BlueBullet, RedBullet, ItemSpeed, ItemEnergy, ItemMove Input_image	ประกาศตัวแปร GameTitle, Menu, StartGUI, QuitGUI, ReStartGUI RedWin, BlueWin, GameBG BlueShip, RedShip, BlueBomb, RedBomb, BlueBullet, RedBullet, ItemSpeed, ItemEnergy, ItemMove Input_image
- Media bgMusic	ประกาศตัวแปร bgMusic

3.8.2 Constructor

+ ResouceLoader()	ทำการสร้าง Classloader สำหรับทุกไฟล์ใน Program
+ ResouceLoader(String input)	ทำการสร้าง Classloader โดยการรับ Path

3.8.3 Methods

+ Stage getTheStage()	ทำการคืนค่า getTheStage
+ Stage setTheStage(Stage theStage)	ทำการตั้งค่า TheStage
+ Getter method for every field	

3.9 Class GameMenu

3.9.1 Field

- ImageView Menu, GameTitle, StartGUI, QuitGUI	ประกาศตัวแปร GameTitle, Menu, StartGUI,
------------------------------------------------	-----------------------------------------

- Pane root	ประกาศตัวแปร root
- Button Start Quit	ประกาศตัวแปร Start Quit
- VBox box	ประกาศตัวแปร box
- Scene Menuscene	ประกาศตัวแปร Menuscene
- Stage	ประกาศตัวแปร theStage

3.9.2 Constructor

+ GameMenu(Pane root, Scene scene, Stage theStage, int x)	ทำการสร้าง GameMenu
-----------------------------------------------------------	---------------------

3.9.3 Methods

+ void drawBoard()	ทำการสร้าง GUI ในเกม
+ Getter & Setter method for every field only type, theStage, QuitGUI	

3.10 Class Main

3.10.1 Methods

+ void start(Stage theStage)	ประกาศตัวแปร Time timer Pane PaneMenu, GamePane, Red, Blue สร้าง Scene RedWin/BlueWin เพื่อใช้ในการสร้าง Scene ของเกมที่เล่น และ สร้าง EventManger /GameMenu Start Program/ Add EventHandler เพื่อใช้ใน moveOne/moveTwo FireOne Start Thread of BGM
public static void main(String[] args)	
public Stage getTheStage()	

3.11 class Bullet implements BombAble

3.11.1 field

# protected int direction	ประกาศ int เพื่อใช้กำหนดทิศทางของกระสุน
---------------------------	-----------------------------------------

	(1 ไปทางขวา -1 ไปทางซ้าย 0 อยู่หนึ่งๆ)
# protected int type	ประกาศชนิดของกระสุน
# protected int speed =1	ประกาศ ความเร็วของกระสุน
#ImageView ImageView	ประกาศ ImageView ของภาพของกระสุน
- ResouceLoader load	Load resouce ของ RedBullet
- ResouceLoader load2	Load resouce ของ BlueBullet

3.11.2 Constuctor

+ Bullet(double X_axis, double Y_axis, int Direction)	ประกาศตำแหน่ง X,Y ของBullet และทิศทาง และ Set type =0 Set ภาพกระสุนให้เหมาะสมกับ เกม
-------------------------------------------------------	--------------------------------------------------------------------------------------

3.11.3 Method

+ void update()	update ตำแหน่งของกระสุนให้ขยับไปเพิ่ม + 10 * direction * speed
+ void setX_axis()	setX of ImageView
+ void setY_axis()	setY of ImageView
+ Group Bomb()	คืนค่าเป็น Group ของ Circle เอฟเฟคของระเบิด
+ Generate all getter setter	

3.12 class SpeedBullet extends Bulle

3.12 .1 Constuctor

+ SpeedBullet(double X_axis, double Y_axis, int Direction)	ทำการสร้างกระสุนความเร็วสูง
------------------------------------------------------------	-----------------------------

3.13 class Potion extends Bullet

3.13.1 field

- String s[]	ทำการเกิด Path ของ Item
--------------	-------------------------

3.13.2 Constuctor

+ Potion(double X_axis, double Y_axis, int Direction, int type)	ทำการสร้าง Potion 3 แบบตาม Type ที่รับมา
-----------------------------------------------------------------	------------------------------------------

3.14 Class Event Manager

3.14.1 Field

- SpaceShip ShipBlue,ShipRed	ประกาศตัวแปร ยานของผู้เล่นทั้ง2คน
-int step	ประกาศตัวแปร ที่ใช้นับระยะการขยับของยาน
<u>-int STEP_UP = -100</u>	
<u>-int STEP_DOWN = 100</u>	
-boolean isTheardStart	เช็คว่ Thread ที่ใช้ update กระสุน ทำงานหรือเปลว
-Pane rootPane,rootRedPane,rootBluePane	ประกาศPaneที่ใช้ ตอนจบเกม
-Scene Gamescene,RedWin,BlueWin	ประกาศ Scene ที่ใช้ตอนจบเกม
-Stage theStage	ประกาศ Stage theStage
-static ArrayList<Bullet> bulletList;	ประกาศ ArrayListที่ใช้เก็บ Bullet/Potion
-int time1,time2	ใช้เพื่อเช็คกระสุนเพื่อไม่ให้เกิด Mechinegun Effect
<u>-int rngRatio</u>	
<u>-int Red_Immune_State</u>	ประกาศตัวแปรใช้เก็บเวลาที่เหลือของสถานะ Energy Immune ของ RedShip
<u>-int Blue_Immune_State</u>	ของRedShip
<u>-int Blue_Mutiple_State</u>	ประกาศตัวแปรใช้เก็บเวลาที่เหลือของสถานะ Multiple Shot ของ BlueShip
<u>-int Red_Mutiple_State</u>	ของRedShip
<u>-int BlueBullet_Speed_State</u>	ประกาศตัวแปรใช้เก็บเวลาที่เหลือของสถานะ Multiple Shot ของ BlueShip
<u>-int RedBullet_Speed_State</u>	ของ RedShip
-ProgressBar blueShipHealthBar ,redShipHealthBar,blueShipEnergyBar,redShipEnergyBar	ประกาศ ProgressBar ของHp กับ Energy ของ ยานทั้ง 2

3.14.2 Constructor

+ EventManager(Pane rootPane, Scene scene, Stage theStage, Scene Red, Scene Blue, Pane RedPane, Pane BluePane)	สร้าง EventManager ด้วย Pane ที่ใช้เล่นเกม Pane/Scene ที่จะใช้ตอนจบเกม, Stage ที่ใช้ SetScene ตอนจบ

3.14.3 Method

-void MoveUnitOne(KeyEvent event)	รับปุ่ม W,S เพื่อเคลื่อนไหว BlueShip
-void MoveUnitTwo(KeyEvent event)	รับปุ่ม ขึ้น,ลง เพื่เคลื่อนไหว RedShip
-void fireOne(KeyEvent event)	รับปุ่ม D เพื่อให้ยาน BlueShipยิง รับปุ่ม ซ้าย เพื่อให้ยาน RedShip ยิง และUpdate สถานะกระสุน/ยาน/Potion ด้วย Thread
+ Scene getGamescene()	
+ Stage getTheStage()	
- void potionRng()	สุ่มเกิด Potion