

1. คำอธิบายเบื้องต้นในการเริ่มเล่น application นี้

Application นี้เป็นเกมยิงระเบิด ที่ได้สร้างสรรค์ขึ้นมาให้ผู้เล่นที่ต้องการจะทำลายป้อมปราการในการเข้าไปยึดปราสาทหลักที่ยังคงอยู่ที่สุดตลอดการของเกมนี้ ที่มีชื่อว่า “KingTower” โดยป้อมปราการแต่ละป้อมจะมีชื่อที่แตกต่างกัน และมีความทันท่วงทีที่แตกต่างกันในแต่ละสี ในส่วนของระเบิดจะมีให้เลือกมากถึง 3 สี ซึ่งจะมีคุณสมบัติที่แตกต่างกันออกไป ตัวเกมจะมีเวลาจำกัดให้ 60 วินาที และสามารถเลือกระดับความยากได้ 3 ระดับ ถ้าหากว่าพร้อมลุยแล้วล่ะก็สามารถกด Start เพื่อเริ่มเกมได้เลย

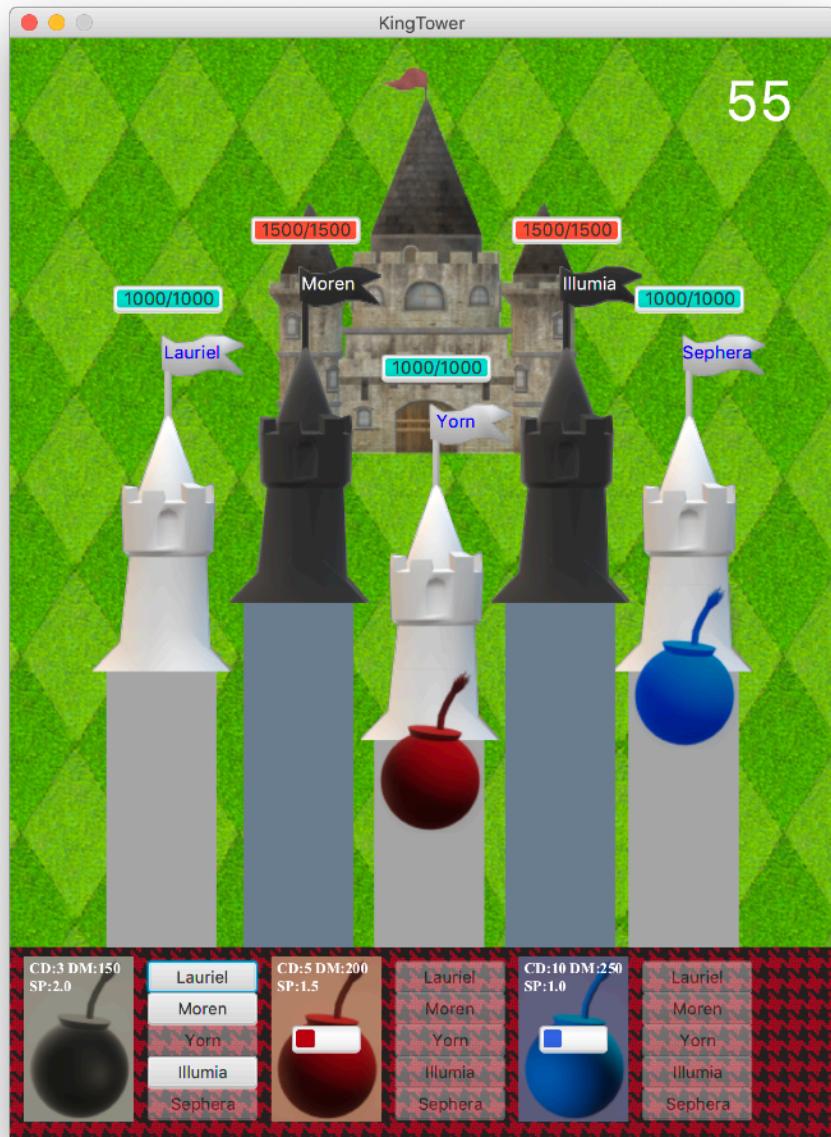




หน้าจอหลักของเกม จะประกอบไปด้วย ปุ่ม Start Game และปุ่ม Settings ซึ่งปุ่ม Settings นี้ จะมีไว้เพื่อย้ายไปหน้าเลือกระดับความยากง่ายของเกม 3 ระดับ คือ Easy Medium และ Hard (ค่าเริ่มต้นคือ Easy) ซึ่งจะมีผลต่อคุณลักษณะ Speed และ damage หลังจากเลือกระดับความยากง่ายเสร็จแล้ว สามารถกด Back เพื่อกลับมาหน้าจอหลัก หรือเลือกกด Start Game เพื่อเริ่มเล่นเกม



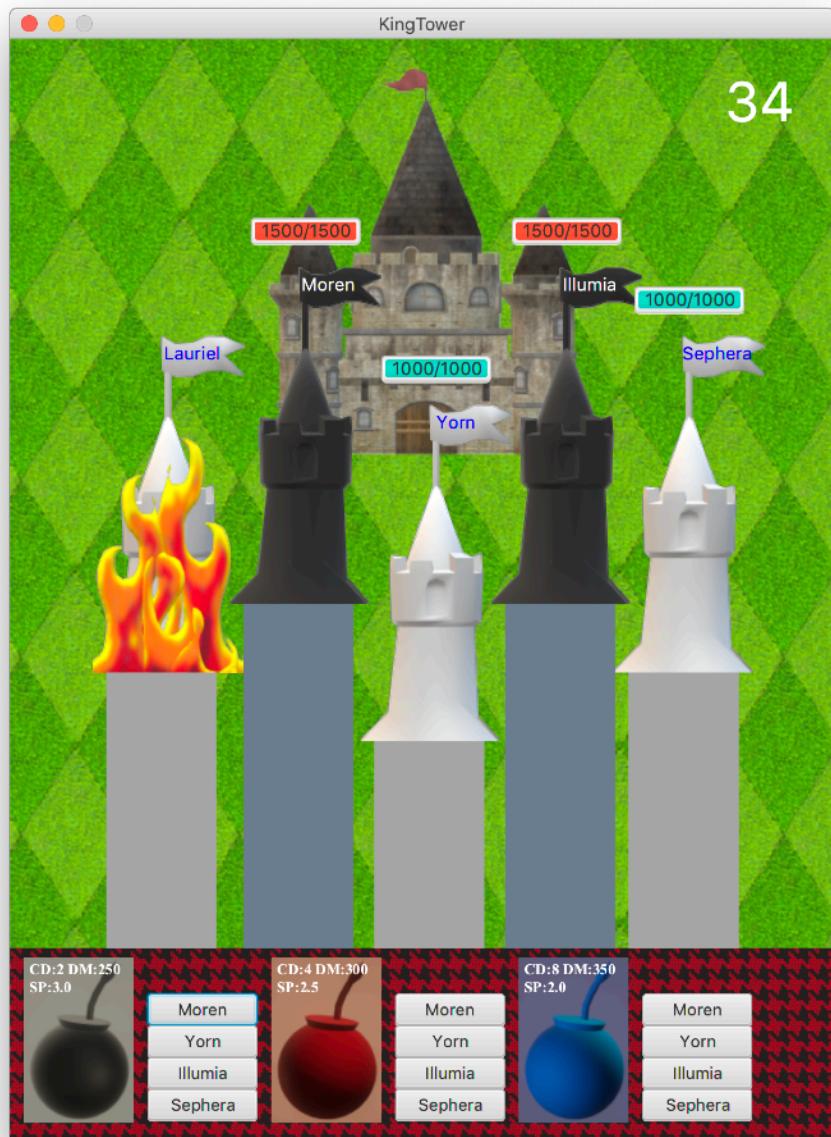
หน้าจอตัวเกม จะประกอบได้ด้วย ตัวนับเวลาถอยหลัง 60 วินาที ปราสาท 5 แห่ง และ King Tower หลัก และระเบิด 3 ชนิดที่มีปุ่มเลือกปราสาทที่ต้องการจะโจมตีอยู่ข้างๆ จำนวน 5 ปุ่ม ໃบแต่ละชนิดของระเบิดนั้นๆ



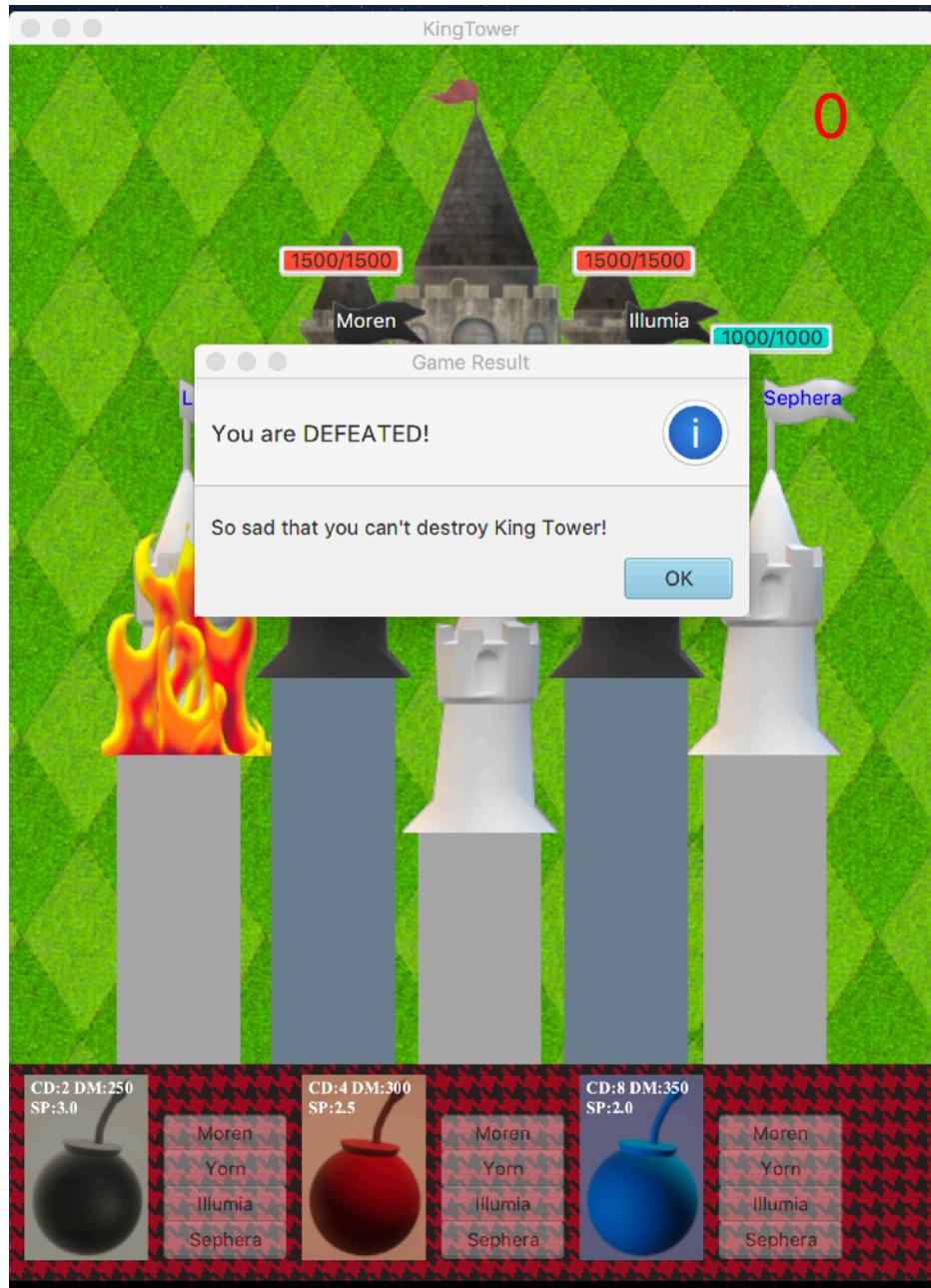
การเล่นเกมขั้นตอนแรกให้เลือกกดปราสาทที่ต้องการโจมตี หลังจากเลือกกดโจมตี ปราสาทจะแล้วระเบิดจะมีคูลดาวน์ในช่วงเวลาหนึ่ง ซึ่งทางเดินไปยังปราสาทจะมีลูกระเบิดได้เพียง 1 ลูกจะระเบิดต่อเส้นทางเดินเท่านั้น และเมื่อลูกระเบิดนั้นได้ระเบิดไปแล้ว จึงจะสามารถเลือกลูกระเบิดใหม่ลงทางเดินนั้นได้



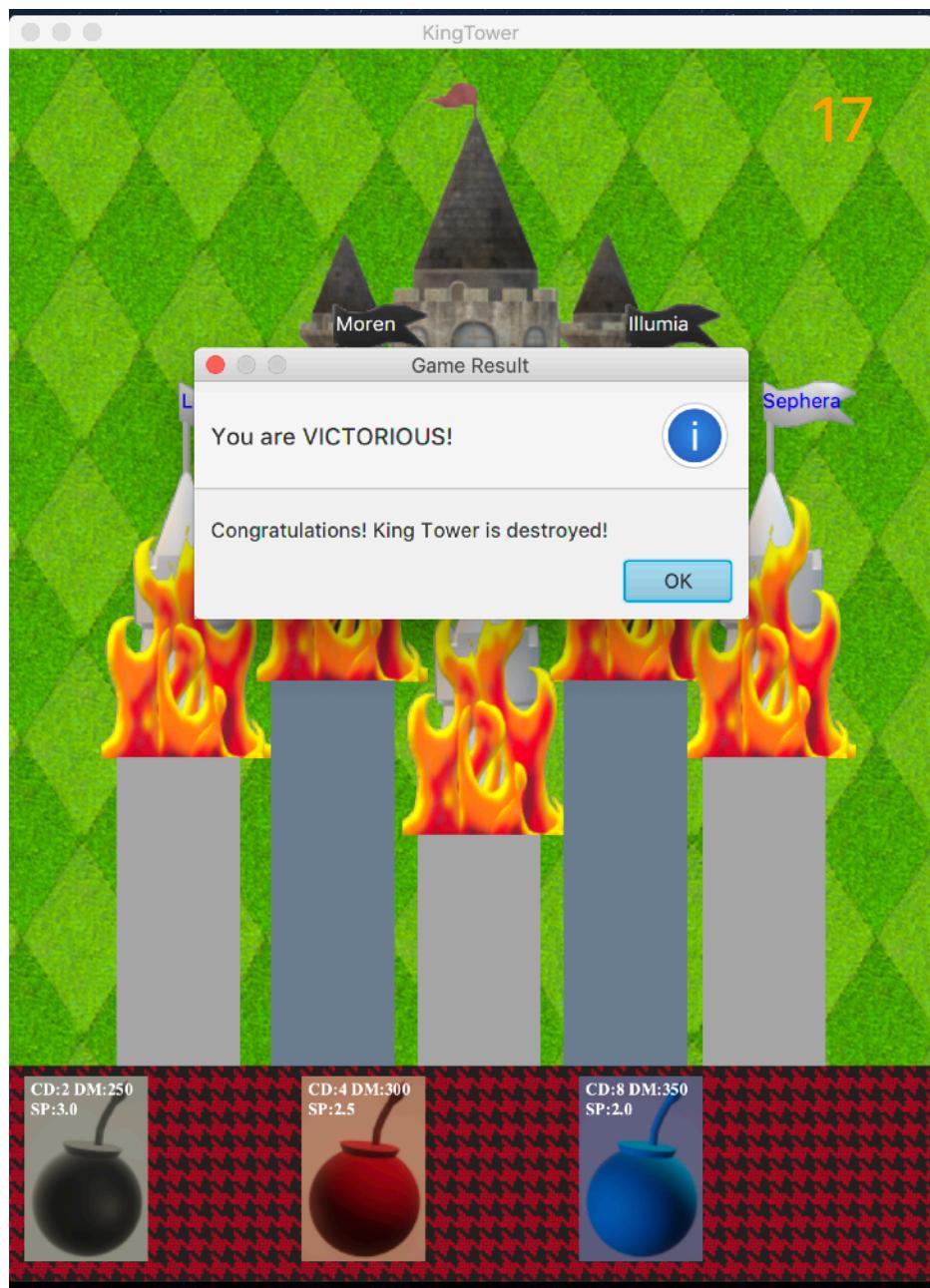
เมื่อลุกระเบิดก็ได้ทำการเลือกไป เดินถึงตัวปราสาทแล้ว ก็จะทำการระเบิดตามความ
แรงของระเบิดชนิดนั้นๆ และจะมีผลโจมต์โดยตรงกับความทันท่วงทายของปราสาท



เมื่อปราสาทถูกกำลังโจมตี ปราสาทนั้นได้ออก และปุ่มเลือกที่จะโจมตีปราสาทนั้นก็จะหายไปอย่างสมบูรณ์



หากไม่สามารถทำลายป้อมปราการทั้งหมดได้ภายในระยะเวลาที่กำหนด ตัวเกมจะแสดงหน้าต่างแจ้งเตือนว่า Time Up



หากสามารถทำลายป้อมปราการทั้ง 5 ป้อมได้สำเร็จภายในระยะเวลาที่กำหนด จะถือว่าชนะเกมและจะมีหน้าต่างแจ้งเตือนว่า Win

* Noted that Access Modifier Notations can be listed below

+ (public)

(protected)

- (private)

static will be underlined.

abstract will be italic.

2.1 CharacterPane extends HBox

2.1.1 Constructor

+ CharacterPane()	<ol style="list-style-type: none">1. Set the Sets the spacing of the InputField to 102. If Game Mode = 1 (Easy), then create three bomb as : name, damage, speed, maxhp, cd, imgCharacter, imgCard Grey, 250, 3, 100, 2, greyBomb, grey Red, 300, 2.5, 100, 4, redBomb, red Blue, 350, 2, 100, 8, blueBomb, blue3. If Game Mode = 2 (Medium), then create three bomb as : name, damage, speed, maxhp, cd, imgCharacter, imgCard Grey, 200, 2.5, 100, 3, greyBomb, grey Red, 250, 2, 100, 5, redBomb, red Blue, 300, 1.5, 100, 10, blueBomb, blue4. If Game Mode = 3 (Hard), then create three bomb as : name, damage, speed, maxhp, cd, imgCharacter, imgCard Grey, 150, 2, 100, 3, greyBomb, grey Red, 200, 1.5, 100, 5, redBomb, red Blue, 250, 1, 100, 10, blueBomb, blue5. Set all cool down progress bar as the same color of bomb
-------------------	---

2.2 GamePane extends Pane

2.2.1 Field

<u>- int currentTime</u>	To count down the time
<u>- long lastTimeTriggered</u>	
<u>- boolean win</u>	default is false
<u>- boolean timeOut</u>	default is false
<u>- AnimationTimer animationTimer</u>	
<u>- Canvas canvas</u>	size 600x800
<u>- GraphicsContext gc</u>	canvas.getGraphicConext2D();
<u>- CharacterPane cp</u>	
<u>- HPPane hp</u>	
<u>- GameScreen gs</u>	

2.2.2 Constructor

<u>+ GamePane()</u>	<ol style="list-style-type: none"> 1. Set size as 600x800 2. Create new GameScreen() as gs 3. Create new CharacterPane as cp 4. Add gs to this pane 5. Add canvas to this pane 6. Add hp to this pane 7. Add cp to this pane 8. Set cp at x=10, y=667
---------------------	---

2.2.3 Method

<u>+ void startTimer()</u>	<ol style="list-style-type: none"> 1. Set currentTime = 60 2. Set lastTimeTriggered = -1 3. Create new canvas and GraphicContext 4. Create new AnimationTimer() 5. Create + void handle(long now) { If lastTimeTriggered<0, set to now If now-lastTimeTriggered>=1000000000 If win, then draw as ORANGE and stop If currentTime = 60, unlock button (btn) If currentTime = 58, play music If currentTime = 0, draw as RED and set all btn disable, set all ready = 0 setTimeOut = true stop() timeOut = true } If 1<=currentTime<=3, then draw as YELLOW Otherwise draw as WHITE lastTimeTriggered = now currentTime- 6. Start animationTimer
----------------------------	--

- void drawCurrentTimeString(GraphicsContext gc2, Color color)	1. gc.setFill(color); 2. gc.setFont(new Font(40)); 3. gc.clearRect(512, 0, 90, 90); 4. gc.fillText(" " +currentTime, 520, 60);
+ Getter/Setter methods for every field	

2.3 Home extends Application

2.3.1 Method

+ void start(Stage primaryStage)	1. Create Pane 2. Create Canvas as 600x800 3. Draw main img 4. Create Scene 600x800 5. Create action close application 6. SetResizable = false 7. Create new GameLogic() 8. Create all button hyperlink 9. Set button click event 10. + void handle(long now) If time out, alert Defeated and stop else if win, alert Victorious and stop else logicUpdate() 11. Start animation 12. Play Home Sound
+ void main(String [] args)	An entry point of the application.

2.4 HPPane extends Pane

2.4.1 Field

- HPBar one	
- HPBar two	
- HPBar three	
- HPBar four	
- HPBar five	
- Label hpOne	
- Label hpTwo	
- Label hpThree	
- Label hpFour	
- Label hpFive	

2.4.2 Constructor

+ HPPane()	<ol style="list-style-type: none"> 1. Set size as 600x800 2. Create new Canvas at 600 140 3. Set canvas x=0, y=660 4. gc = getGraphicsContext2D 5. draw RenderableHolder.getBar() in gc at 0, 0 6. Set color and layout for all HpBar 7. Add name castle label 8. Call HPUpdate()
------------	---

2.4.3 Method

+ void HPUpdate()	<ol style="list-style-type: none"> 1. Initialise Thread 2. Thread sleep 100 milliseconds 3. set double(now Hp / max Hp) 4. If 3) = 0, then set Invisible 5. If Time Out or Win, break 6. Thread start
-------------------	---

2.5 Interface Drawable

2.5.1 Method

+ void draw(GraphicsContext gc)	
---------------------------------	--

2.6 GameScreen extends Canvas

2.6.1 Constructors

+GameScreen()	<ol style="list-style-type: none"> 1. Set size as 600x800 2. Draw img
---------------	---

2.6.2 Method

- void draw(GraphicsContext gc, Image img)	<ol style="list-style-type: none"> 1. Draw at 0,0
+ void paintComponent()	<ol style="list-style-type: none"> 1. For all entities 2. If entity visible and not being destroyed, then draw 3. If all castle is collapse(dead), setWin = true

2.7 BlackCastle extends Castle implements Drawable

2.7.1 Constructors

+ BlackCastle(int x, int y, int width, int height, String name)	1. super all and set name
---	---------------------------

2.8 WhiteCastle extends Castle implements Drawable

2.8.1 Constructors

+ WhiteCastle(int x, int y, int width, int height, String name)	1. super all and set name
---	---------------------------

2.9 ButtonLane extends Button

2.9.2 Field

- String name	
- String character	
- int ready	
- int lane	
- boolean dead	

2.9.2 Constructors

+ ButtonLane(String name, int lane, String character)	1. Super name 2. Set size 80, 20 3. Set ready = 0 4. Set disable = true 5. Set dead = false
---	---

2.9.3 Method

+Getter & Setter methods for every field	
--	--

2.10 Castle extends Entity implements Drawable

2.10.1 Field

- int hp	
- int maxhp	
- <u>int speed</u>	Set default as 10
- int activatePoint	
- int x, y	
- int width, height	
- HPBar hpbar	
- <u>int LIMIT</u>	Set default as 5000
- Image img	
- boolean dead	
- String name	
- <u>List<Castle> allCastle</u>	
- <u>List<Castle> activeCastle</u>	

2.10.2 Constructors

+ Castle(int x, int y, int width, int height, int maxhp, Image img)	<ol style="list-style-type: none"> 1. Set default hp = maxhp 2. Set dead = false 3. Set destroyed = false 4. Set hpbar x+5, y+140 5. Set name Castle
---	---

2.10.3 Method

+ void draw(GraphicsContext gc)	@Override
+ Getter & Setter methods for every field	

2.11 Character extends Entity implements Drawable

2.11.1 Field

- String name	
- int damage	
- double speed	
- int hp, maxhp	
- int cd	
- double x, y, length	
- Image imgCharacter	
- Image imgCard	
- Castle target	
- ButtonLane owner	

2.11.2 Constructors

+ Character(double x, double y, double length, String name, int damage, double speed, int maxhp, int cd, Image imgCharacter, Image imgCard)	1. Set destroyed = false
---	--------------------------

2.11.3 Method

+ boolean collideWith(Castle target)	return this.y - target.getY() <= 165
+ void draw(GraphicsContext gc)	@Override and setY(getY() - getSpeed());
+ void crash(Castle target)	<ol style="list-style-type: none"> Set new Hp after attack If hp<=0, set to 0 Set WhiteCastle and BlackCastle Fire image Set bomb sound effect Set destroyed = true Set owner.setDead(true)
+ Getter & Setter methods for every field	

2.12 CharacterCard extends HBox

2.12.1 Field

- List<ButtonLane> allbtn1	
- List<ButtonLane> allbtn2	
- List<ButtonLane> allbtn3	
- List<ButtonLane> allbtn4	
- List<ButtonLane> allbtn5	
- List<ButtonLane> all	
- String name	
- int damage	
- double speed	
- int hp, maxhp	
- int cd	
- Image imgCharacter	
- Image imgCard	
- ButtonLane one, two, three, four, five	
- CoolDownBar cb	
- List<Character> allCharacter	

2.12.2 Constructors

+ CharacterCard(String name, int damage, double speed, int maxhp, int cd, Image imgCharacter, Image imgCard)	<ol style="list-style-type: none"> 1. Set space 10 2. Set Alignment CENTER 3. Create new CoolDownBar as cb 4. Set cb width = 50 5. Create new Canvas at 80, 120 6. Add imgCard 7. Create new VBox as allbutton 8. allbutton alignment at CENTER 9. Set action on each tower button (one, two, three, four, five)
--	---

2.12.3 Method

+ Getter & Setter methods for every field	
---	--

2.13 CoolDownBar extends ProgressBar

2.13.1 Field

- CharacterCard owner	
- boolean ready	

2.13.2 Constructors

+ CoolDownBar(CharacterCard owner)	<ol style="list-style-type: none"> 1. ready = true 2. Width = 100 3. setVisible(false) 4. setProgress(0)
------------------------------------	--

2.13.3 Method

+ void update(ButtonLane one, ButtonLane two, ButtonLane three, ButtonLane four, ButtonLane five, List<ButtonLane> a, List<ButtonLane> all, int cd, String name)	<ol style="list-style-type: none"> 1. New Thread 2. Inside While(true) if time out or win, setVisible(false) and break loop 3. Then, sleep 10 milliseconds and set new progress at old+0.002*5/cd 4. For all, If name = itself and itself is dead, then minus 1 from isReady() 5. setDead(false) 6. If bomb progress bar >= 1, then set ready = true and set visible = false and minus isReady() by 1 for all button in its bomb type and set progress = 0. 7. For all button in pane if isReady() = 0, then loop for setDisable(false) 8. If isReady(), then break 9. Start thread
+ Getter & Setter methods for every field	

2.14 Class Entity (Abstract)

2.14.1 Field

- double x,y	
- double width, height	
- boolean visible, destroyed	
- boolean draw	

2.14.2 Constructors

- Entity()	1. visible = true 2. destroyed = false
------------	---

2.14.3 Method

+Getter & Setter methods for every field	
--	--

2.15 GameLogic

2.15.1 Field

- int gameMode	Set default = 1
- Castle c1	WhiteCastle 60, 210, 110, 250, "Lauriel"
- Castle c2	BlackCastle 160, 160, 110, 250, "Moren"
- Castle c3	WhiteCastle 255, 260, 110, 250, "Yorn"
- Castle c4	BlackCastle 350, 160, 110, 250, "Illumia"
- Castle c5	WhiteCastle 439, 210, 110, 250, "Sephera"
- Lane l1	70, 460, 80, 200, Color.DARKGREY
- Lane l2	170, 410, 80, 250, Color.SLATEGRAY
- Lane l3	265, 510, 80, 150, Color.DARKGREY
- Lane l4	360, 410, 80, 250, Color.SLATEGRAY
- Lane l5	450, 460, 80, 200, Color.DARKGREY

2.15.2 Constructors

- GameLogic()	1. Set all button disable = true 2. Add all castle and lane
---------------	--

2.15.3 Method

+ void logicUpdate()	<ol style="list-style-type: none"> 1. If is Character and not being destroyed and collide with target, then call crash() 2. If is WhiteCastle or BlackCastle and hp = 0, then setDead = true 3. If tower is down, then button also disable
+Getter & Setter methods for every field	

2.16 HPBar extends ProgressBar

2.16.1 Constructors

+ HPBar()	<ol style="list-style-type: none"> 1. Set Progress = 1 2. Set Width = 80
-----------	--

2.17 Lane extends Entity

2.17.1 Field

- List<Character> allcharacter	
- List<Lane> allLane	
- int x, y, width, height	
- Color color	

2.17.2 Constructors

+ Lane(int x, int y, int width, int height, Color color)	1. allLane.add(this)
--	----------------------

2.17.3 Method

+ void draw(GraphicsContext gc)	1. @Override Fill color and fillRect
+Getter & Setter methods for every field	

2.18 Interface IRenderable

2.18.1 Methods

+ boolean isDraw()	
+ void draw(GraphicContext gc)	
+ boolean isDestroyed()	
+ void setDestroyed(boolean destroy)	

+ boolean isVisible()	
-----------------------	--

+ setDraw(boolean draw)	
-------------------------	--

2.19 RenderableHolder

2.19.1 Fields

- <u>RenderableHolder instance</u>	
- <u>List<IRenderable> entities</u>	
- <u>Image blueBomb</u>	
- <u>Image redBomb</u>	
- <u>Image greyBomb</u>	
- <u>Image blue</u>	
- <u>Image red</u>	
- <u>Image grey</u>	
- <u>Image whiteCastle</u>	
- <u>Image whiteCastlefir</u>	
- <u>Image whiteCastlemorefir</u>	
- <u>Image blackCastle</u>	
- <u>Image blackCastlefir</u>	
- <u>Image blackCastlemorefir</u>	
- <u>Image bar</u>	
- <u>Image game</u>	
- <u>Image main</u>	
- <u>AudioClip homesound</u>	
- <u>AudioClip greysound</u>	
- <u>AudioClip redsound</u>	
- <u>AudioClipbluesound</u>	
- <u>AudioClip gamesound</u>	

2.19.2 Methods

static	loadResource()
+Getter & Setter methods for every field	