

# **Java Project Report**

on

**PB : GO**

Submitted by

**Kitipat Teachasupinnun 5930041921**

**Ravipas Plitnonkiat 5930430321**

**Subject**

**Course Programming Methodology Semester 1**

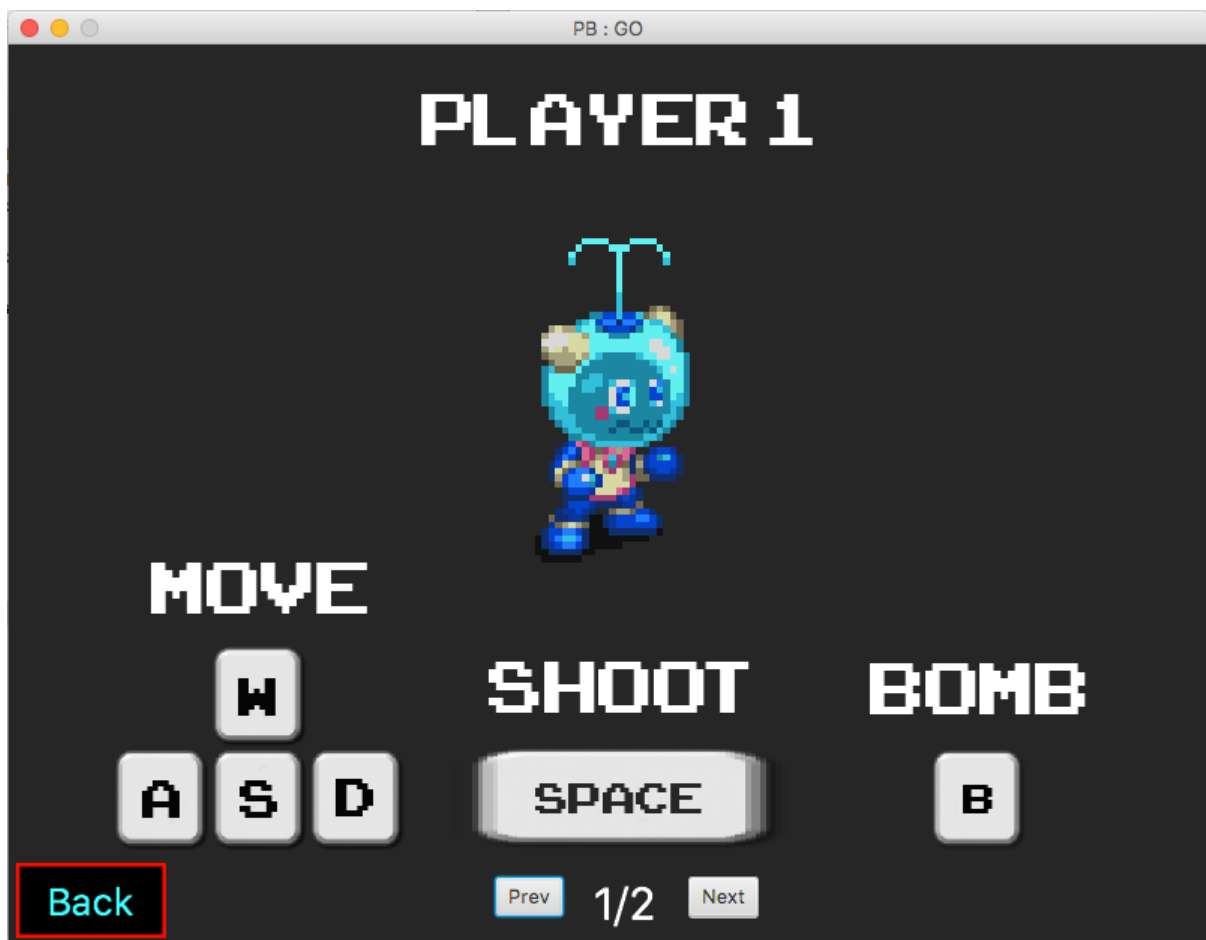


when start the program you will see mainmenu that have 3 button to click

When you click HowTo

- You will go to How to play screen that introduce you to know how to control your character.
- In this game need 2 player to play together so we have 2 page of Howto to teach you how to control each player character.

## How to play Player 1



your can control movement of player1 character by

- press A to go left
- press W to go up
- press S to go down
- press D to go right

your can command player1 to shoot by

- press SPACE BAR





your can command player1 to throw bomb by

- press B

## How to play Player 2



your can control movement of player1 character by

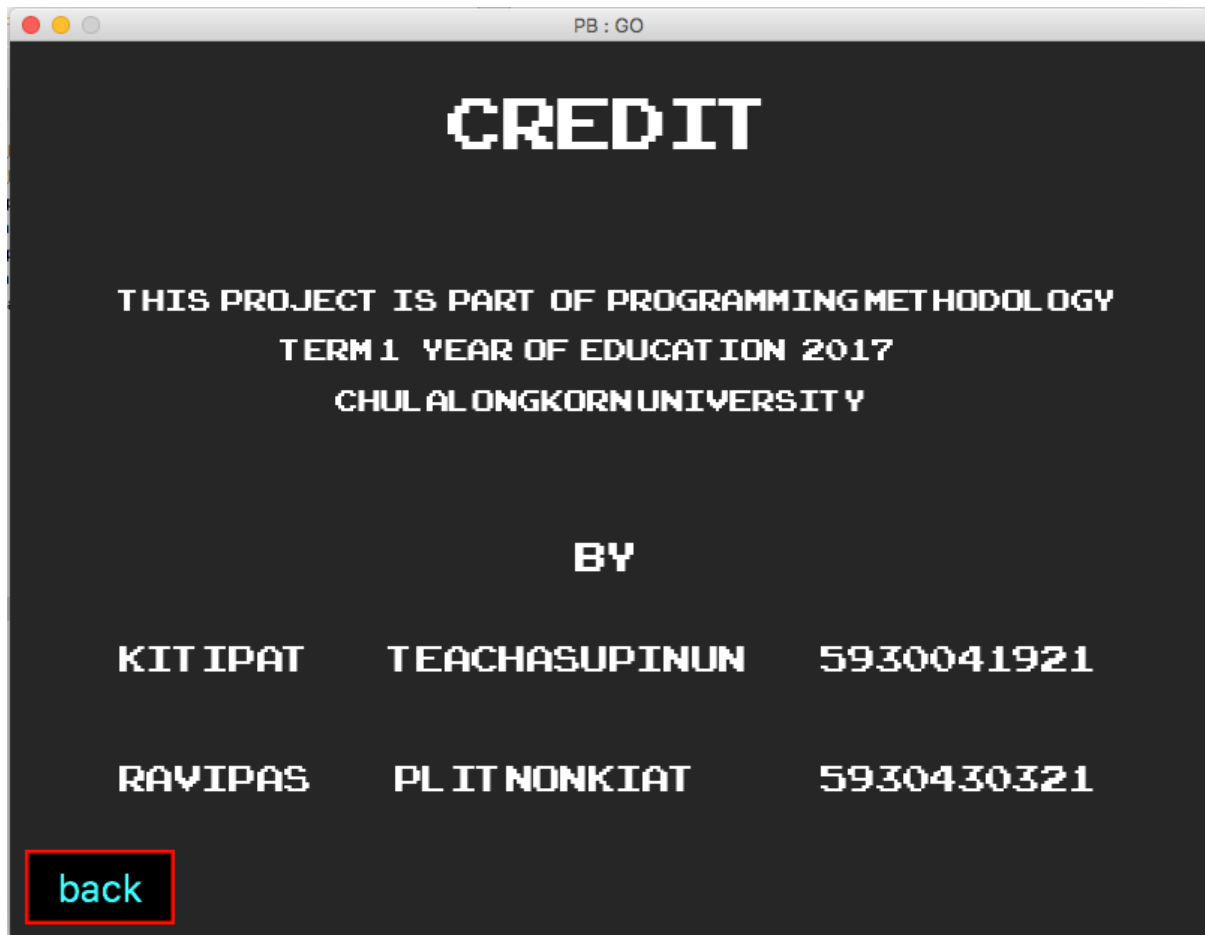
- press  to go left
- press  to go up
- press  to go down
- press  to go right

your can command player1 to shoot by

- press L

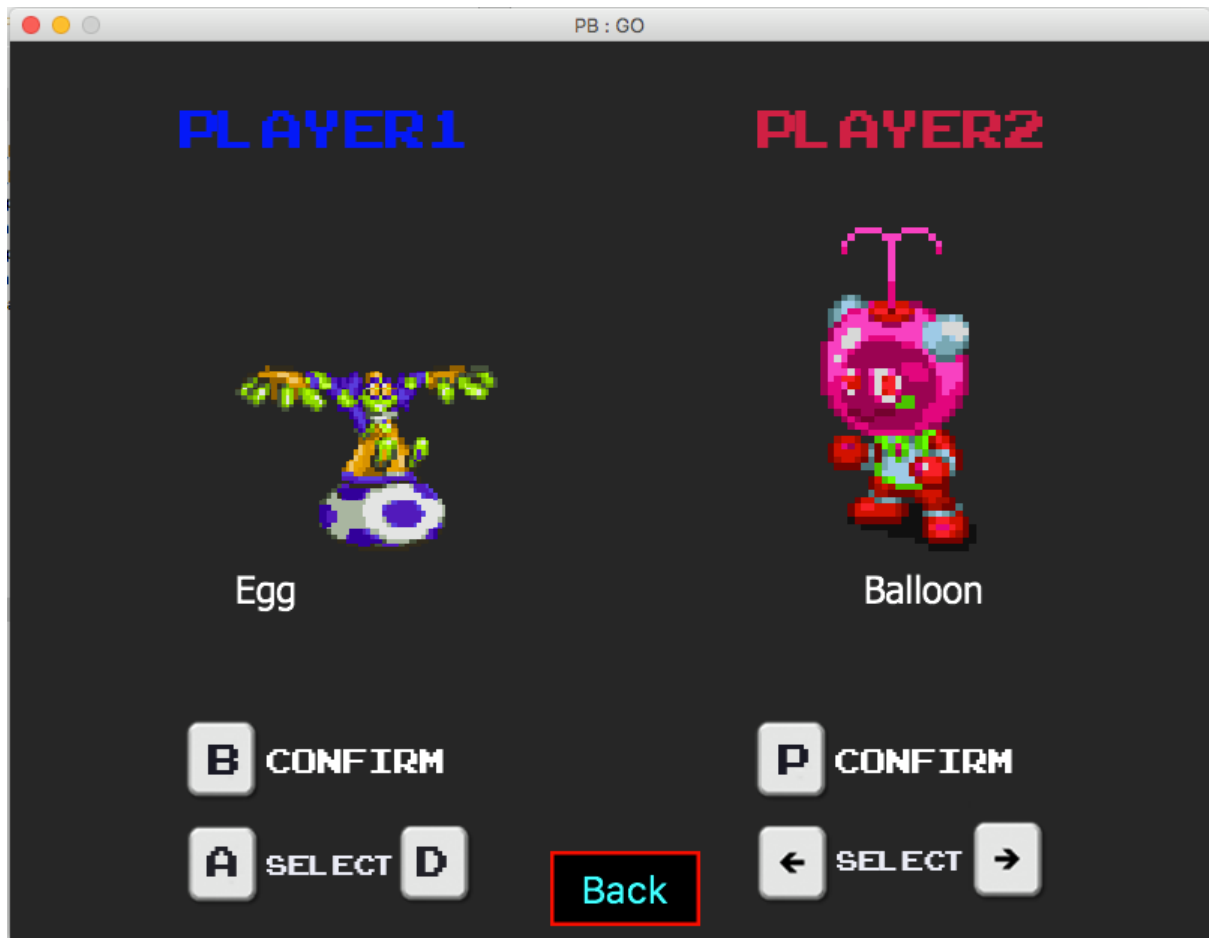
your can command player1 to throw bomb by

- press P



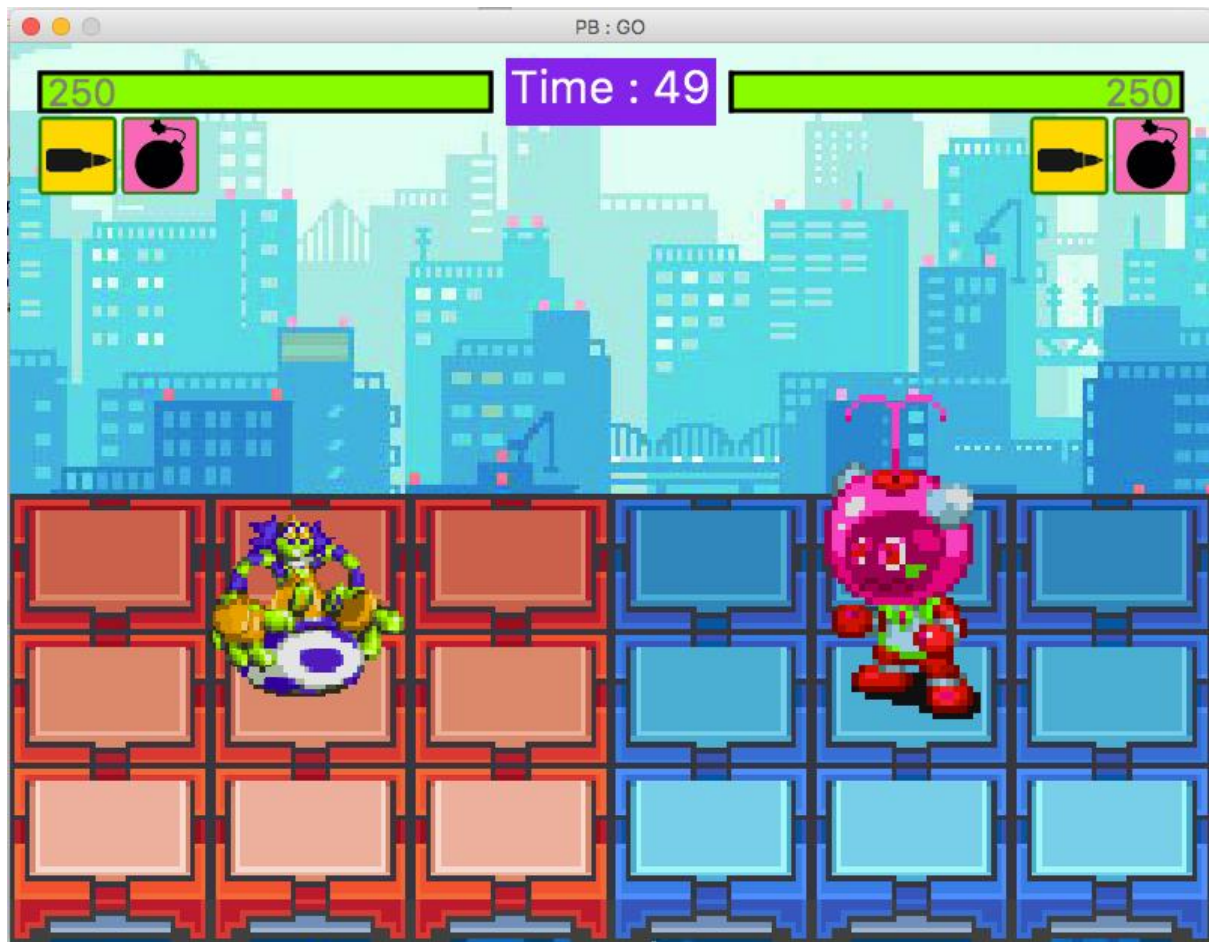
When you click Credit

- You will go to How to credit screen that show credit.



When you click Start

- You will go to How to CharacterSelect screen that let you choose your own character .
- Left side is for player 1 to choose character.
- right side id for player 2 to choose character.
- when you choose finish press your own confirm button to confirm your character.
- when two player click confirm go to gameScreen.



### GameScreen

-In this game we separate two player in other side.

-player 1 is left.

-player 2 is right.

-Goal of this game is you must shoot other player to reduce their health to 0 and you will win!!

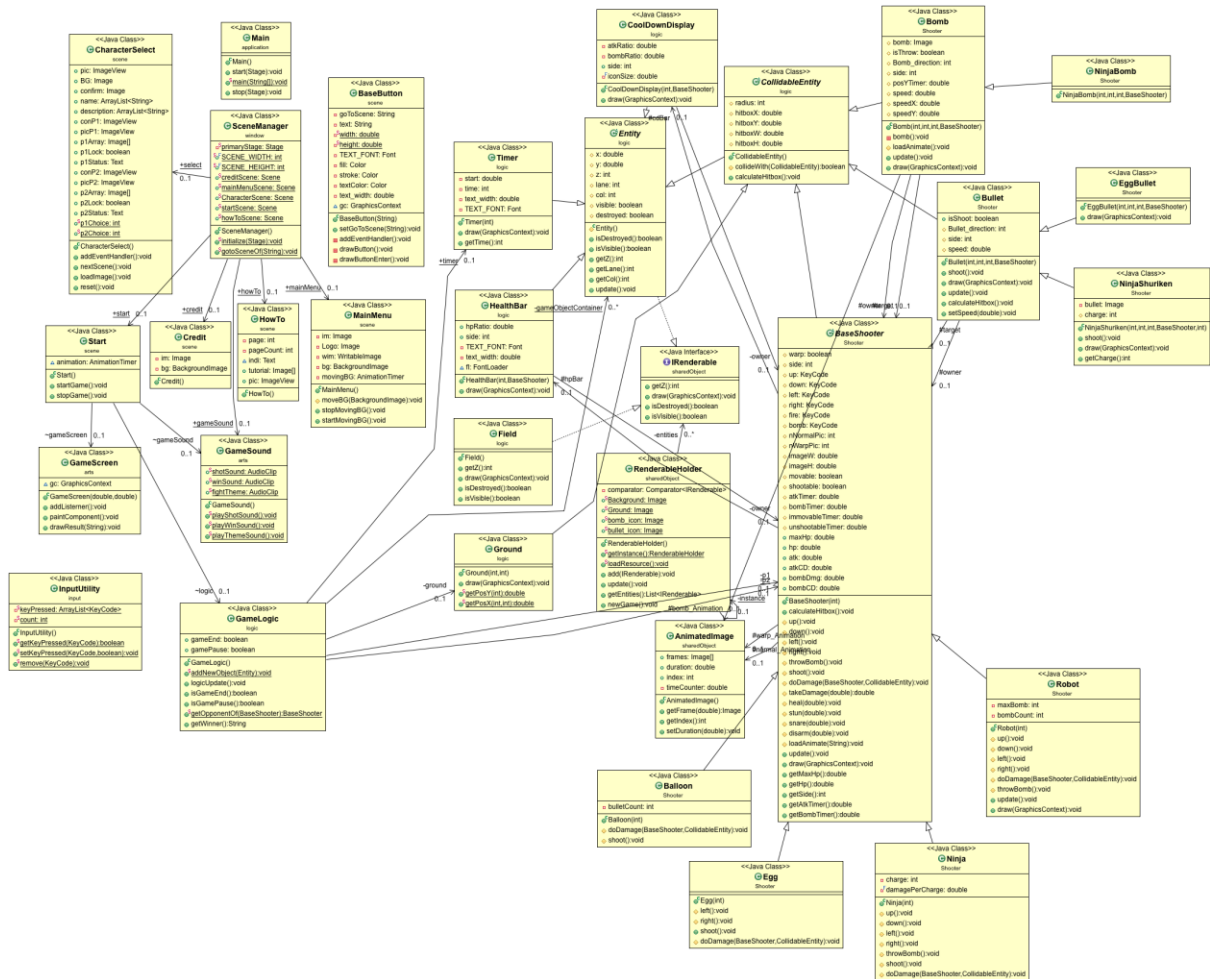
-but if health of each other is equal the match result is draw.



-when the game end .

-at the middle of gamescreen will show match result and you can go back to mainmenu and play again!!





An UML Diagram of Project.

## Class Detail :

### 2.1 Package application

#### 2.1.1 Class main

##### 2.1.1.1 Method

Void start	The main entry point for the JavaFX applications.
Void main	An entry point of the application

Void stop(stage primaryStage)	This method are going to execute before JavaFX application terminates.
-------------------------------	--

## 2.2 Package arts

### 2.2.1 Class GameScreen extends Canvas

#### 2.2.1.1 Field

GraphicsContext gc	A GraphicsContext .
--------------------	---------------------

#### 2.2.1.2 Constructor

GameScreen(double,double)	Initialize GameScreen.
---------------------------	------------------------

#### 2.2.1.3 Method

Void addListerner()	Addlisterner to receive key event.
Void paintComponent()	This method is used to draw draw all component in IRenderable entity.
Void drawResult(String winner)	<p>draw result of game match</p> <ul style="list-style-type: none"> <li>- if don't have any winner match result should be draw.</li> <li>- else draw a result that which player is a winner.</li> </ul>

## 2.2.2 Class GameSound

### 2.2.1.1 Field

<u>AudioClip shotSound</u>	AudioClip when you shoot.
<u>AudioClip winSound</u>	AudioClip when you win.
<u>AudioClip fightTheme</u>	AudioClip when you fighting.

### 2.2.1.2 Constructor

GameSound()	<ul style="list-style-type: none"><li>▪ Initialize shotSound.</li><li>▪ Initialize winSound.</li><li>▪ Initialize fightTheme.</li></ul>
-------------	---

### 2.2.1.3 Method

<u>Void playShotSound ()</u>	play Shot bullet sound.
<u>Void playWinSound()</u>	play sound when the game end.
<u>Void playThemeSound()</u>	play theme sound when playing game.

## 2.3 Package input

### 2.2.1 Class InputUtility

#### 2.2.1.1 Field

<u>ArrayList&lt;KeyCode&gt; keyPressed</u>	ArrayList that keep key that have been pressed
--	--

#### 2.2.1.3 Method

<u>Boolean getKeyPressed(KeyCode keycode)</u>	get the key that you have presses
<u>void setKeyPressed(KeyCode keycode,boolean pressed)</u>	set the key that you have pressed
<u>void remove(KeyCode keycode)</u>	remove the key in keyPressed

## 2.4 Package logic

### 2.2.1 Class CollidableEntity

#### 2.2.1.1 Field

double hitboxX	X position of hitbox
double hitboxY	Y position of hitbox
double hitboxW	hitbox width
double hitboxH	hitbox height

### 2.2.1.3 Method

boolean collideWith(CollidableEntity other)	-if this collide with other object return true  - else return false
void calculateHitbox()	calculate hitbox.

### 2.2.2 Class CoolDownDisplay extends Entity

#### 2.2.1.1 Field

Double atkRatio	percent of cooldown when attack is press
Double bombRatio	percent of cooldown when bomb is use
Int side	side of the player
BaseShooter owner	owner Character player
Double iconSize	iconSize

#### 2.2.1.2 Constructor

CoolDownDisplay(int side,BaseShooter owner)	- Initialize x position of cooldowndisplay - Initialize y position of cooldowndisplay - Initialize z of cooldowndisplay - Initialize side of cooldowndisplay - Initialize owner
---	---

### 2.2.1.3 Method

Void draw(GraphicsContext gc)	draw cooldowndisplay for each player
-------------------------------	--------------------------------------

## 2.2.2 Class Entity implements IRenderable

### 2.2.1.1 Field

Double x	x position of Entity
Double y	y position of Entity
Int z	z of Entity
Int lane	lane of player
Int col	column of player
Boolean destroyed	is object is destroyed
boolean visible	is object is visible

### 2.2.1.2 Constructor

Entity()	<ul style="list-style-type: none"><li>-Initialize isdestroyed = false</li><li>-Initialize visible = true</li></ul>
----------	--

### 2.2.1.3 Method

Boolean isDestroyed()	<ul style="list-style-type: none"><li>-if object is destroyed return true</li><li>-else return false</li></ul>
Boolean isVisible()	<ul style="list-style-type: none"><li>-if object is visible return true</li><li>- else return false</li></ul>
Int getZ()	return z position
Int getLane()	return lane
Int getCol()	return column
Void update()	update

## 2.2.2 Class Field implements IRenderable

### 2.2.1.3 Method

Int getZ()	return Z position
Void draw(GraphicsContext gc)	drawBackground
Boolean isDestroyed()	-if object is destroyed return true -else return false
Boolean isVisible()	-if object is visible return true - else return false

## 2.2.2 Class GameLogic

### 2.2.1.1 Field

List<Entity> gameObjectContainer	List that keed gameObject
Boolean gameEnd	is gameEnd
Boolean gamePause	is gamePausee
Timer timer	A timer
Ground ground	A ground
BaseShooter p1	A player 1
BaseShooter p2	A player 2



### 2.2.1.2 Constructor

GameLogic()	<ul style="list-style-type: none"><li>- Initialize gameObjectContainer</li><li>- Initialize field</li><li>- Initialize timer</li><li>- Initialize ground</li><li>- Initialize p1Choice</li><li>- Initialize p2Choice</li></ul> <p>if player1Choice equal 0</p> <p>-- Initialize p1 = balloon character</p> <p>if player1Choice equal 1</p> <p>-- Initialize p1 = egg character</p> <p>if player1Choice equal 2</p> <p>-- Initialize p1 = Ninja character</p> <p>if player1Choice equal 3</p> <p>-- Initialize p1 is robot character</p> <p>if player2Choice equal 0</p> <p>-- Initialize p2 = balloon character</p> <p>if player2Choice equal 1</p> <p>-- Initialize p2 = egg character</p> <p>if player2Choice equal 2</p> <p>-- Initialize p2 = Ninja character</p> <p>if player2Choice equal 3</p> <p>-- Initialize p2 is robot character</p> <p>add ground,p1,p2,timer to addNewObject</p> <p>set Volumn of game theme</p> <p>play game theme</p>
-------------	---

### 2.2.1.3 Method

Void addNewObject(Entity entity)	add new object to entity
Void logicUpdate()	update game logic  -if ESCAPE button is press stop game  -else update all item in Objectcontainer  -if one of each player health is 0 or time reach 0  game is end.
Boolean isGameEnd()	is Game end.
Boolean isGamePause()	is Game pause.
BaseShooter getOpponent(BaseShooter shooter)	return your opponent.
Sttring getWinner()	return who is winner.

### 2.2.2 Class Ground extends CollidableEntity

#### 2.2.1.2 Constructor

Ground (int x,int y)	<ul style="list-style-type: none"><li>▪ Initialize ground x position.</li><li>▪ Initialize ground y position.</li><li>▪ Initialize z =-20.</li></ul>
----------------------	--

### 2.2.1.3 Method

Void draw(GraphicsContext gc)	draw ground image
Double getPosY(int lane)	return ground Y position
Double getPosX(int col,int side)	return ground X position

### 2.2.2 Class HealthBar extends Entity

#### 2.2.1.1 Field

Double hpRatio	percent of healthbar
Int side	side of character
BaseShooter owner	character
Font TEXT_FONT	text Font
Double text_width	text width
FontLoader fl	font loader

### 2.2.1.2 Constructor

HealthBar(int side,BaseShooter sh)	<ul style="list-style-type: none"><li>- Initialize healthbar side</li><li>- Initialize x postion</li><li>- Initialize y position</li><li>- Initialize z position</li><li>- Initialize character</li></ul>
------------------------------------	---

### 2.2.1.3 Method

Void draw(GraphicsContext gc)	draw healthbar
-------------------------------	----------------

## 2.2.2 Class Timer extends Entity

### 2.2.1.1 Field

Double start	double starting time
Int time	int time
Double text_width	text width
Font TEXT_FONT	text font

### 2.2.1.2 Constructor

Timer(int countDown)	<ul style="list-style-type: none"><li>▪ Initialize z = 50</li><li>▪ Initialize starting time = System.nanoTime()</li><li>▪ Initialize time = cooldown</li></ul>
----------------------	---

### 2.2.1.3 Method

Void draw(GraphicsContext gc)	draw timer at the top center of game screen
Void getTime()	return time

## 2.2 Package scene

### 2.2.1 Class BaseButton extends Canvas

#### 2.2.1.1 Field

String goToScene	String that keed scenename
String text	text
<u>Double width</u>	button width
<u>Double height</u>	button height
Font TEXT_FONT	text font
Color fill	button color
Color stroke	button stroke
Color textColor	text color
GraphicsContext gc	GraphicsContext gc

### 2.2.1.2 Constructor

BaseButton(String text)	super(width,height)
-------------------------	---------------------

### 2.2.1.3 Method

Void setGoToScene(String goToScene)	set scene that want to go
Void addEventHandler()	addEvent to button -when mouse click go to other scene - when mouse on change button color - when mouse off chage button color
Void drawButton()	draw button
Void drawButtonEnter()	drawbutton when mouse enter

### 2.2.1 Class CharacterSelect extends AnchorPane

#### 2.2.1.1 Field

ImageView pic	Imageview picture
Image BG	Image background
Image confirm	Image when we confirm
ArrayList<String> name	ArrayList that keep name
ImageView conP1	ImageView conP1

ImageView picP1	ImageView picP1
Image[] p1Array	Image[] p1Array
Boolean p1Lock	Boolean p1Lock
Text p1Status	Text p1Status
ImageView conP2	ImageView conP2
ImageView picP2	ImageView picP2
Image[] p2Array	Image[] p2Array
Boolean p2Lock	Boolean p2Lock
Text p2Status	Text p2Status
Int p1Choice	Int p1Choice
Int p2Choice	Int p2Choice

#### 2.2.1.2 Constructor

CharacterSelect()	Initialize CharacterSelectScreen.
-------------------	-----------------------------------

### 2.2.1.3 Method

Void addEventHandler()	Addlistener to receive key event
Void nextScene()	go to next scene
Void loadImage()	loadImage

### 2.2.1 Class Credit extends BorderPane

#### 2.2.1.1 Field

Image im	A credit Image
BackgroundImage bg	Use for set an backGround.

#### 2.2.1.2 Constructor

Credit()	Initialize GameScreen.
----------	------------------------

### 2.2.1 Class HowTo extends AnchorPane

#### 2.2.1.1 Field

Int page	Number of total page.
Int pageCount	Number of current page.
Text indi	Text to show current page



Image[] tutorial	Collect an Image in tutorial.
ImageView pic	Use for show image.

#### 2.2.1.2 Constructor

Howto()	Initialize GameScreen.
---------	------------------------

### 2.2.1 Class MainMenu extends VBox

#### 2.2.1.1 Field

Image im	Full Back Ground image.
Image Logo	Image of logo.
WritableImage wim	Cut Back ground fit to screen.
BackgroundImage bg	use for change back ground
AnimationTimer movingBG	AnimationTimer for move back ground.

#### 2.2.1.2 Constructor

MainMenu()	Initialize GameScreen.
------------	------------------------

### 2.2.1.3 Method

Void moveBG(BackgroundImage bg)	change BackGround to new Back Ground
Void stopMovingBG()	Stop and movingBG animationTimer.
Void startMovingBG()	Start and movingBG animationTimer.

### 2.2.1 Class Start extends StackPane

#### 2.2.1.1 Field

GameLogic logic	Use for updateLogic.
GameScreen gameScreen	Use for drawing .
GameSound gameSound	sound service
AnimationTimer animation	<a href="#">loop in game</a>

#### 2.2.1.2 Constructor

Start()	Initialize GameScreen. Initialize GameLogic
---------	--

### 2.2.1.3 Method

Void startGame()	<a href="#">Start animation</a>
Void stopGame()	<a href="#">Stop animation</a>

## 2.2 sharedObject

### 2.2.2 Class AnimatedImage

#### 2.2.1.1 Field

Image[] frames	Array of animation.
Double duration	Interval between each Image.
Int index	Index of current frames.
Double timeCounter	Using in getFrames.

#### 2.2.1.3 Method

Image getFrame(double time)	if(SystemTime-timeCounter)sendNewframes.
Int getIndex()	return currentIndex.
Void setDuration(double duration)	set duration between each Image.

### 2.2.2 Class IRenderable

#### 2.2.1.3 Method

Int getZ()	For sorting and element.
Void draw(GraphicsContext gc)	For paint an graphic.
Boolean isDestroyed()	true if object no longer on the field.

Boolean isVisible()	true if object show on the field.
---------------------	-----------------------------------

## 2.2.2 Class RenderableHolder

### 2.2.1.1 Field

List<IRenderable> entities	collects all entity on gamescreen.
Comparator<IRenderable> comparator	Using for Sort entities.
<u>Image Background</u>	backGround for using inGame.
<u>Image Ground</u>	Ground in game;
<u>Image bomb_icon</u>	bomb icon
<u>Image bullet_icon</u>	bullet icon

### 2.2.1.2 Constructor

RenderableHolder()	<ul style="list-style-type: none"> <li>▪ Initialize entities</li> <li>▪ Initialize comparator</li> </ul>
--------------------	--

### 2.2.1.3 Method

<u>RenderableHolder getInstance()</u>	make Instance to this.
<u>Void loadResource()</u>	loadResource to container.

Void add(IRenderable entity)	add entity to entities.
Void update()	erase object if that object is destroyed.
List<IRenderable> getEntities()	return entities.
Void newGame()	clear data in entities.

### 2.2.2 Package Shooter

#### 2.2.2 Class Balloon extends BaseShooter

##### 2.2.1.1 Field

Int bulletCount	Count the bullet
-----------------	------------------

##### 2.2.1.2 Constructor

Balloon(int side)	Use super.
-------------------	------------

##### 2.2.1.3 Method

Void doDamage(BaseShooter other,CollidableEntity way)	Like super() but every third attack hit heal the character.
---	---

## 2.2.2 Class BaseShooter extends CollidableEntity

### 2.2.1.1 Field

AnimatedImage normal_Animation	Animation of character while standing still.
AnimatedImage warp_Animation	Animation of character while moving.
Boolean warp	check if while warp animation.
Int side	side of character.0 = left, 1 =right.
HealthBar hpBar	HealthBar of Character.
CoolDownDisplay cdBar	CoolDownDisplay of Character.
KeyCode up,down,left,right,fire,bomb	KeyCode for command.
Int nNormalPic	Number of frames in normal_animation
Inr nWarpPic	Number of frames in warp_animation
Double imageW	Width of Image.
Double imageH	Height of Image.
Boolean movable	False if CC.

Boolean shootable	False if CC.
Double atkTimer	For calculate, next attack.
Double immovableTimer	Duration of movement disable.
Double unshootableTimer	Duration of attack disable.
Double maxHp	maxHp of Character.
Double hp	current hp of Character.
Double atk	Attack Damage.
Double atkCD	Interval between each attack.
Double bombDmg	Bomb Damage.
Double bombCD	Interval Between each attack.

#### 2.2.1.2 Constructor

BaseShooter(int side)	Constructor of a class.
-----------------------	-------------------------

### 2.2.1.3 Method

Void calculateHitbox()	Calculate hitbox of character.
Void up()	Move up.
Void down()	Move down.
Void left()	Move left.
Void right()	Move right.
Void throwBomb()	if(curentTime-bombTimer>=bombCD) make new Bomb;
Void shoot()	if(curentTime-atkTimer>=atkCD) make new bullet;
Void doDamage(BaseShooter other ,CollidableEntity way)	Do damage to other .
Double takeDamage(double dmg)	Recieve for dmg.And return damage dealt.
Void heal(double amount)	Increase Hp for amount.
Void stun(double durationSecond)	For duration, character can't move or attack.
Void snare(double durationSecond)	For duration, character can't move.



Void disarm(double durationSecond)	For duration, character can't attack.
Void loadAnimate(String charName)	LoadImage to normal_animation and warp_animation.
Void update()	recieve and input from inputUtility.
Void draw(GraphicsContext gc)	show image frame by frame from normal_animation and warp_animation.
Double getMaxHp()	return maxHp.
Double getHp()	return hp;
Int getSide()	return side;
Double getAtkTimer()	return atkTimer.
Double getBombTimer()	return bombTimer;

## 2.2.2 Class Bomb extends CollidableEntity

### 2.2.1.1 Field

AnimatedImage bomb_Animation	Animation that collected explosion image.
Image bomb	Image of grenade.
Boolean isThrow	Boolean for bomb in the air.

Int Bomb_direction	Direction of bomb.
Int side	side of object.
BaseShooter owner	owner of this.
BaseShooter target	target fir this.
Double posYTimer	position of Y by time.
Double speed	Speed in line of movement.
Double speedX	Speed in X axis.
Double speedY	Speed in Y axis.

#### 2.2.1.2 Constructor

Bomb(int side,int lane ,int col,BaseShooter owner)	Consructor of an object.
--	--------------------------

### 2.2.1.3 Method

Void bomb()	set X, Y and isthrow = true.
Void loadAnimate()	LoadImage to bomb_animation.
Void update()	Move the bomb. When hit ground do damage to opponent.
Void draw(DraphicsContext gc)	Show picture of bomb ,when hit the ground Show explosion.

### 2.2.2 Class Bullet extends CollidableEntity

#### 2.2.1.1 Field

Boolean isShoot	Boolean for check that bullet in the air.
Int Bullet_direction	Direction of bullet.
Int side	Side of object.
BaseShooter owner	Owner of this.
BaseShooter target	Terget for this.
Double speed	Speed of bullet.

### 2.2.1.2 Constructor

Bullet(int side,int lane ,int col,BaseShooter owner)	Consructor of an object.
--	--------------------------

### 2.2.1.3 Method

Void shoot()	Set X ,Y and isShoot = true.
Void draw(GraphicsContext gc)	Show graphic of bullet.
Void update()	Move bullet equal to speed*bullet_direction.
Void calculateHitbox()	Calculate hitBox for this.
Void setspeed(double speed)	Set speed.

### 2.2.2 Class Egg extends BaseShooter

#### 2.2.1.2 Constructor

Egg(int side)	Consructor of an object.
---------------	--------------------------

### 2.2.1.3 Method

Void left()	Move left.
Void right()	Move right.
Void doDamage(BaseShooter other,CollidableEntity way)	doDamage to other.

## 2.2.2 Class EggBullet extends Bullet

### 2.2.1.2 Constructor

EggBullet(int side,int lane ,int col,BaseShooter owner)	Consructor of an object.
--	--------------------------

### 2.2.1.3 Method

Void draw(DraphicsContext gc)	Draw Image of Balloon_Bullet.
-------------------------------	-------------------------------

## 2.2.2 Class Ninja extends BaseShooter

### 2.2.1.1 Field

Int charge	Increase by 1 when move.
Double damagePerCharge	Use in doDamage.

### 2.2.1.2 Constructor

Ninja(int side)	Consructor of an object.
-----------------	--------------------------

### 2.2.1.3 Method

Void up()	move up and increase charge.
Void down()	move down and increase charge.
Void left()	move left and increase charge.
Void right()	move right and increase charge.
Void throwBomb()	if(pass the bombCD)throw a bomb.
Void shoot()	shoot a bullet.
Void doDamage(BaseShooter other,CollidbaleEntity way)	if(bomb)damage and snare target for 1 sec. if(bullet)doDamage plus charge*damagePerCharge.

### 2.2.2 Class NinjaBomb extends Bomb

#### 2.2.1.2 Constructor

NinjaBomb(int side,int lane ,int col,BaseShooter owner)	Consructor of an object.
--	--------------------------

## 2.2.2 Class NinjaShuriken extends Bullet

### 2.2.1.1 Field

Image bullet	Image of bullet.
Int charge	Charge when shoot.

### 2.2.1.2 Constructor

NinjaShuriken(int side,int lane ,int col,BaseShooter owner,int charge)	Constructor of an object.
--	---------------------------

### 2.2.1.3 Method

Void shoot()	set X ,Y and isShoot = true.
Void draw()	draw image of bullet.
Void getCharge()	return charge of this bullet.

## 2.2.2 Class Robot extends BaseShooter

### 2.2.1.2 Constructor

Robot(int side)	Constructor of an object.
-----------------	---------------------------

### 2.2.1.3 Method

Void up()	move up.
Void down ()	move down.
Void left()	move left.
Void right()	move right.

### 2.2.2 Package window

### 2.2.2 Class SceneManager

#### 2.2.1.1 Field

<u>Stage primaryStage</u>	primaryStage of Program.
<u>Int SCENE_WIDTH</u>	Width of scene.
<u>Int SCENE_HEIGHT</u>	Height of scene.
<u>GameSound gameSound</u>	Sound service .
<u>MainMenu mainMenu</u>	Initialized mainMenu Pane.
<u>Credit credit</u>	Initialized credit Pane.
<u>Howto howTo</u>	Initialized howTo Pane.



<u>CharacterSelect select</u>	Initialized select Pane.
<u>Start start</u>	Wait for Initialize.
<u>Scene creditScene</u>	Scene of credit.
<u>Scene mainMenuScene</u>	Scene of mainMenu.
<u>Scene CharacterScene</u>	Scene of Character Selector.
<u>Scene startScene</u>	Scene of game.
<u>Scene howToScene</u>	Scene of HowTo

### 2.2.1.3 Method

<u>Void initialize()</u>	Initialize and primaryStage.
<u>Void gotoSceneOf(String scene)</u>	goTo destination Scene.