

Shimada Bros

1.Overview

Shimada Bros is a game that 2 players need to fight against each other (PVP). This game is based on Overwatch game. This game has 2 character as Genji and Hanzo which come from Shimada Clan. When a game start there will be a screen (Shown in figure 1) with background music. After that, player can press ENTER to play. If clicked Play Button to play, then the game screen will change to loading screen that has shortcut tell player what you need to know before playing the game (Shown in figure 2). After that, game screen will change to battle screen that you can play the game (Shown in figure 3). When the game is end, it will change to play of the game screen that show the winner of the game (Shown in figure 4).



Figure 1



Figure 2

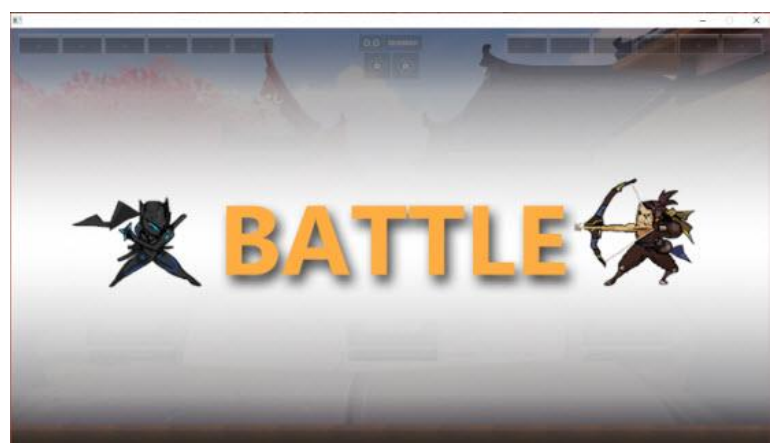




Figure 3



Figure 4

For the size of screen (1280x720)

Detail

1) Player

-There are 2 players that are Genji() and Hanzo()

-player with 0% hp first will lose.

1.1) Genji

Passive: can double jump by pressing W button twice

Weapon: sword (melee)

Move speed: fast

ATK: 30

HP: 200



1.2) Hanzo

Passive: use mouse to aim the target

Weapon: arrow (range)

Move speed: medium

ATK: 20

HP: 200



2) Platform

-Players can stand on platform

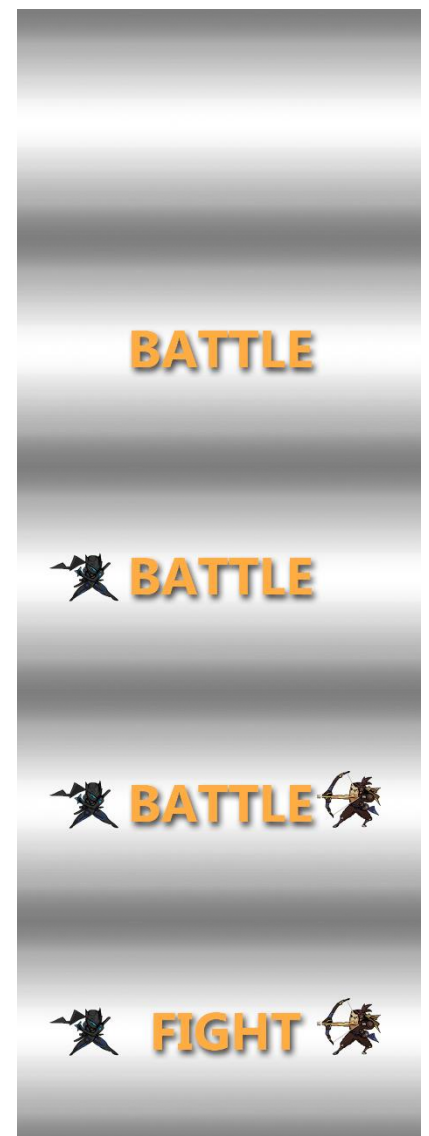
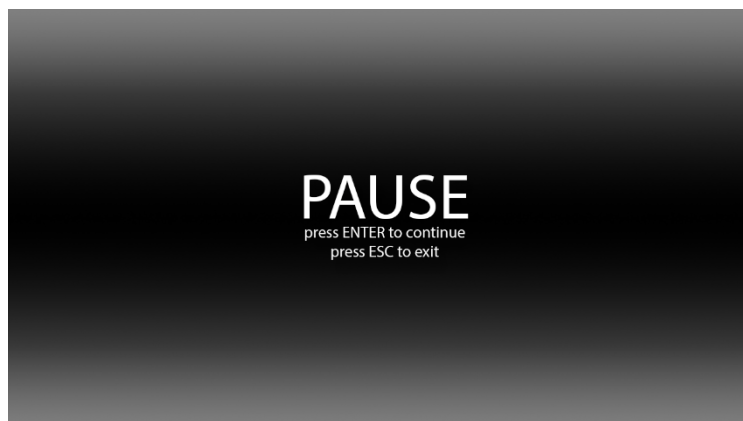


Resource

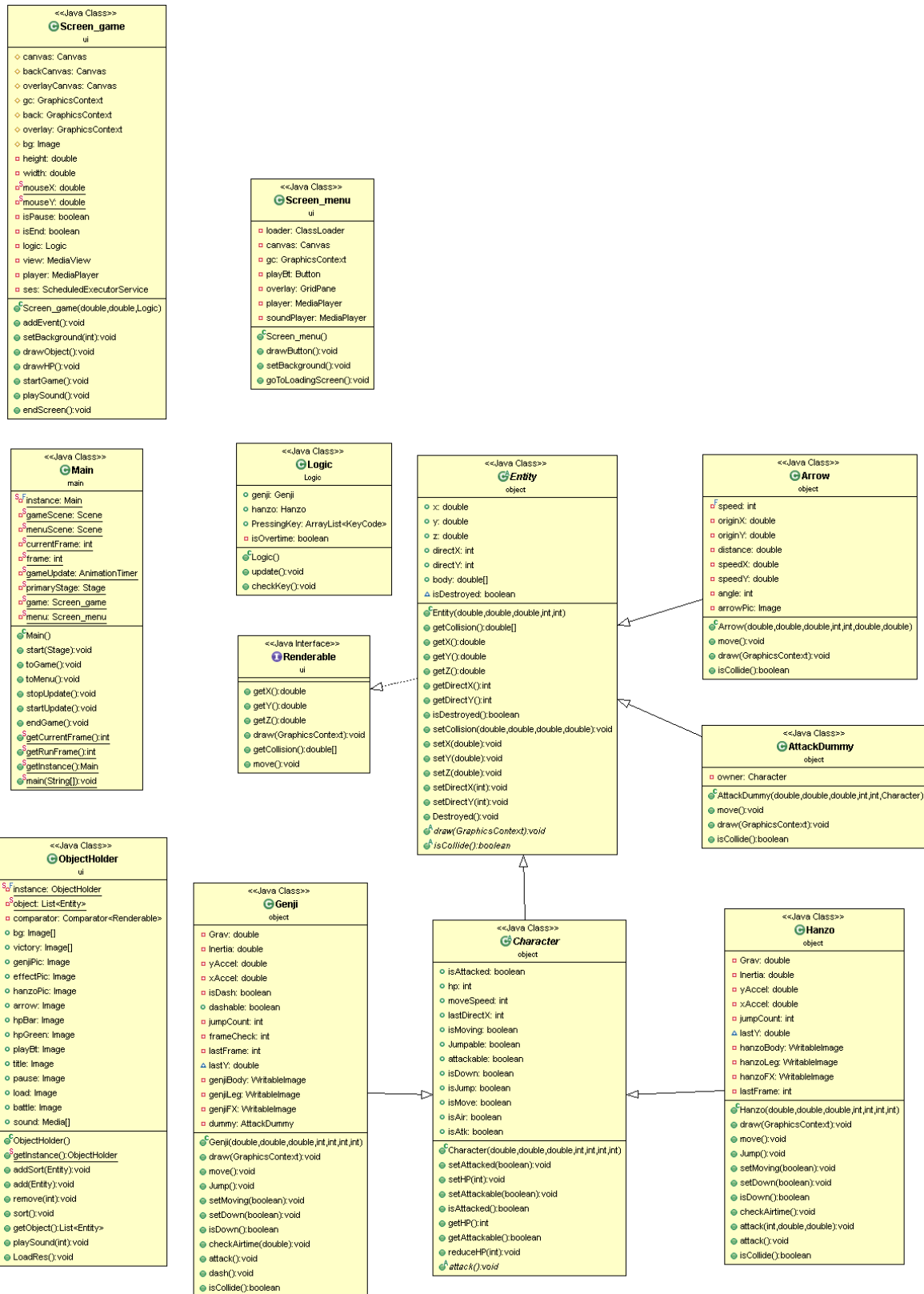


Genji
Victory

Hanzo
Victory



2.UML Class Diagram



3.Class Details

3.1 Package ui

3.1.1 Class Renderable

Class that need to draw image must implement this interface

Method

+double getX()	Return X
+double getY()	Return Y
+double getZ()	Priority of drawing
+void draw(GraphicsContext gc)	Implement drawing
+double[] getCollision()	Check collision of object
+void move()	Move object

3.1.2 Class ObjectHolder

This Class Hold all object

1) Field

-static final ObjectHolder instance	A Singleton object of RenderableHolder.
-static List<Entity> object	List of Renderable objects, this list must always be sorted.
-Comparator<Renderable> comparator	A Comparator to sort entities list by z value.
+Image[] bg	List of all background image.

+Image genjiPic, effectPic, HanzoPic, arrow, hpBar, hpGreen, playBt, title , pause	Picture of all object in game.
--	--------------------------------

2.) Constructor

ObjectHolder	Initialize - comparator should be compare base on z value of the Renderable object.
--------------	--

3.) Method

+static ObjectHolder getInstance()	Return instance
+void addSort(Entity obj)	Add object and sort to object list.
+void remove(int i)	Remove object in object list (by index).
+List<Entity> getObject()	Return object in list.
+void LoadRes()	Load all image and sound.
+void add(Entity obj)	Add obj to object list.

3.1.3 Class Screen_menu

This class is GUI of Main Menu

1.) Field

-ClassLoader loader	Classloader for load picture and sound.
-Canvas canvas	Canvas to draw.
-GraphicsContext gc	A graphics context represents a drawing destination.
-Button playBt	Play Button to start a game.
-GridPane overlay	A pane for show in screen.
-MediaPlayer player,soundplayer	Video of the Menu.

2.) Constructor

+Screen_menu	Initialize -size,alignment -setBackground() -drawButton() -setOnMouseReleased when click call method goToLoadingScreen.
--------------	--

3.) Method

+drawButton()	setMouseEnter and setOnMouseExited when mouse
---------------	--

	enter the button it will change color.
+void setBackground()	Insert Video Insert Background sound and set volume.
+goToLoadingScreen() (use Thread)	Stop video and sound Change screen to loadingscreen(use Main.toGame())

3.1.4 Screen_game

GUI of game screen

1.) Field

#Canvas canvas, backCanvas, overlayCanvas	A Canvas that contain -Background (no update) -Character (always update) -Pause (no update)
#GraphicsContext gc, back, overlay	A graphics context represents a drawing destination.
#Image bg	Background of screen.
-Double height, width	Width and Height of screen.
-static double mouseX	
-static double mouseY	

-Boolean clickCheck, isPause	Check when click and check game is pause.
-Logic logic	Bring logic class to game screen.
-MediaView view	
-MediaPlayer player	
-ScheduledExecutorService ses	

2.) Constructor

Screen_game(Double width, Double height, Logic logic)	Initialize -logic width and height. -clickCheck set to true.
---	--

3.) Method

+void addEvent()	-setOnKeyPressed(ENTER) when press ENTER key then screen will pause and pause image show up. When pause if press ESC key will return to main menu and game will reset. When pause if press Enter again it will unpause. -setOnKeyPress Player1:
------------------	--

	<p>When press W key character Genji will jump , press again will double jump.</p> <p>When press A key Genji will move left side.</p> <p>When press S key Genji will move down one floor.</p> <p>When press D key Genji will move right side.</p> <p>When press J key Genji will slash the target.</p> <p>Player2:</p> <p>When press UP character Hanzo will jump.</p> <p>When press DOWN Hanzo will move down one floor.</p> <p>When press LEFT Hanzo will move left side.</p> <p>When Press RIGHT Hanzo will move right side.</p> <p>When Left Click Hanzo will shot arrow to the way mouse clicked.</p>
+void setBackground()	Set background to screen
+void drawObject()	<p>-drawHP</p> <p>-draw everything in Renderable Holder</p>

+drawHP()	Use image hpBar to draw HP bar
+void startGame()	Start game
+void playSound()	startGame() play sound
+void endScreen()	Change screen to victory

3.2 Package object

3.2.1 Class Entity

Abstract Class of every object and implements Renderable class

1.) Field

+Double x,y,z	Position of object x,y,z
+Int directX,directY	Direction of object Standstill if directX = 0 or directY = 0 Rightside if directX is positive Leftside if directX is negative Downside if directY is positive Upside if directY is negative
+Double [] body	Contain corner point (topleft,topright,bottomleft, bottomright) for check the Collision.
+Boolean isDestroyed	Check object

2.) Constructor

Entity (double x, double y, double z , int dx , int dy)	Initialize -x,y,z,directX(dx),directY(dy) -body[4]
--	--

3.) Method

+Getters and Setters	Getters and Setters of x,y,z,directX,directY,body ,isDestroyed
+abstract void draw(gc)	
+abstract Boolean isCollide()	

3.2.2 Class Character

Abstract Class that extends Entity

1.) Field

+boolean isAttacked	Check that object can attack
+int hp, moveSpeed, lastDirectX	HP, moveSpeed , last direction before change direction
+boolean isMoving, Jumpable, attackable, isDown, isJump, isMove, isAir, isAtk	Check Condition as name of variable.

2.) Constructor

Character (double x, double y, double z, int dx, int dy, int hp, int moveSpeed)	Initialize -x, y, z, dx, dy, moveSpeed, dx = lastDirectX -set isDestroyed and isAttacked to false -set Jumpable and attackable to true
---	--

3.) Method

+Getters and Setters	isAttacked,HP
+void reduceHP(int dmg)	Remove hp with dmg if hp < 0 set HP = 0
+abstract void attack()	

3.2.3 Class Genji

This is one of a player class that extends from Character

1.) Field

-double Grav	Number of gravity in game =0.35
-double Inertia	Number of inertia (in Physics) =1
-double yAccel	Speed per update of y axis
-double xAccel	Speed per update of x axis

-int jumpCount, frameCheck, lastFrame	
+double lastY	Y position before y change
-WritableImage genjiBody, genjiLeg, genjiFX	Part of Genji picture
-AttackDummy dummy	

2.) Constructor

Genji (double x, double y, double z, int dx, int dy, int hp, int moveSpeed)	Initialize -x, y, z, dx , dy, hp , moveSpeed, lastY, lastFrame = 0 -set isJump, isMove, isDown, isAtk is false -set isAir is true -load image to genjiBody, genjiLeg and genjiFX -setCollision(x, x+50, y, +75)
---	--

3.) Method

+void draw(GraphicsContext gc)	Draw follow an action of character (more information in code)
+void move()	Initialize the condition that can move and move it

	(more information in code)
+void jump()	jumpCount increase by 1 and if jumpCount <= 2 then the object can jump and change yAccel =10
+Getters and Setters	Moving, Down
+void checkAirtime(double lastYAcc)	If directY = 0 then change isAir is false. If not change to true
+void attack()	If isAtk = false then frameCheck = 0 , isAtk = true and create new dummy to ObjectHolder
+boolean isCollide()	setCollision(x, x+50, y, y+75) check every object to every possible condition (Formula below)

if((first point of object corner □ -1 and second point of object corner □ 1) or (first point of object corner □ -1 and second point of object corner □ 1)){ if(third point of object corner □ -1 and fourth of object corner □ 1) or (third point of object corner □ -1 and fourth point of object corner □ 1)}

if object is arrow decrease hp 20 and that object destroyed then return true

3.2.4 Class Hanzo

This is one of a player class that extends from Character

1.) Field

-double Grav	Number of gravity in game =0.35
-double Inertia	Number of inertia (in Physics) =1
-double yAccel	Speed per update of y axis

-double xAccel	Speed per update of x axis
-int jumpCount, frameCheck, lastFrame	
+double lastY	Y position before y change
-WritableImage hanzoBody, hanzoLeg, hanzoFX	Part of Hanzo picture

2.) Construction

Genji (double x, double y, double z, int dx, int dy, int hp, int moveSpeed)	Initialize -x, y, z, dx , dy, hp , moveSpeed, lastY, lastFrame = 0 -set isJump, isMove, isDown, isAtk is false -set isAir is true -load image to hanzoBody, hanzoLeg and hanzoFX
---	---

3.) Method

+void draw(GraphicsContext gc)	Draw follow an action of character (more information in code)
+void move()	Initialize the condition that can move and move it

	(more information in code)
+void jump()	jumpCount increase by 1 and if jumpCount <= 1 then the object can jump and change yAccel =11
+Getters and Setters	Moving, Down
+void checkAirtime(double lastYAcc)	If directY = 0 then change isAir is false. If not change to true
+void attack()	create new arrow to ObjectHolder
+boolean isCollide()	setCollision(x, x+50, y, y+75) check every object to every possible condition (Formula below)

Formula same as Genji (hp reduce remake as 30)

3.2.5 Class AttackDummy

This class is a attack dummy for genji attack extends from Entity

1.) Field

-Character owner	character
------------------	-----------

2.) Constructor

AttackDummy(double x, double y, double z, int dx, int dy, Charater owner)	Initialize x, y, z, dx ,dy , owner
---	------------------------------------

3.) Method

+void move()	SetCollision(x, x+45, y, y+70) If owner directX or lastdirectX is 1 then x = owner(x) + 55 If owner directX is -1 then x = owner(x) - 60 Y = owner(y)
+abstract void draw(GraphicsContext gc)	
+boolean isCollide()	Return false

3.2.6 Class Arrow

This is arrow from hanzo attack and extends from Entity

1.) Field

-final int speed	15
-double originX, originY, distance, speedX, speedY	Origin x,y of arrow and speed x, y axis of speed
-int angle	Angle of arrow
-Image arrowPic	Image of arrow

2.) Constructor

+Arrow(double x, double y, double z, int dx, int dy, double ox, double oy)	Initialize -x, y, z, dx, dy , origin = ox and originY = oy -distance, speedX, speedY, angle
--	--

```
distance = Math.sqrt(Math.pow(x-originX,2)+Math.pow(y-originY,2));
```

```
speedX =((originX-x)/distance)*21;
```

```
speedY =((originY-y)/distance)*21;
```

```
angle =(int)Math.toDegrees(Math.acos((originX-x)/distance));
```

3.) Method

+void move()	Move x axis and y axis setCollision(x+25, x+75, y+25 , y+75) if object out of screen destroy this object
+void draw(GraphicsContext gc)	Draw object
+boolean isCollide()	Return false

3.3 Package Logic

This class is Logic of the game

3.1.1 Field

+Genji genji	Character in game (Genji)
+Hanzo hanzo	Character in game (Hanzo)

+ArrayList<KeyCode> PressingKey	Array to collect Keycode when pressingkey
------------------------------------	--

2.) Constructor

+Logic()	Genji(200, 500, 0, 0, 0, 200, 7); Hanzo(1000, 500, 0, 0, 0, 200, 6) PressingKey
----------	---

3.) Method

+void update()	-checkKey() -if all entity not destroyed then x.move() -remove all entity that destroyed -if one of those two character destroyed(die) then stopUpdate
+void checkKey()	Check all condition of PressingKey (more information in code)

3.4 Package main

Class Main – Run program extends Application

1.) Field

-static final Main instance	A singleton instance of this class
-static Scene gameScene	Scene of game screen
-static Scene menuScene	Scene of menu screen
-static int currentFrame	Current frame
-static int frame	Frame
-static AnimationTimer gameUpdate	Timer for game update
-static Stage primaryStage	For set primarystage
-static Scene_game game	For set game screen
-static Scene_menu menu	For set menu screen

2.) Method

+void start(Stage primaryStage)	Set scene and show
+void toGame()	currentFrame = 0 frame = 0 load resource from ObjectHolder new game gameUpdate = new AnimationTimer call startUpdate()
+void toMenu()	Load resource from ObjectHolder Set to menu screen
+void stopUpdate()	Stop game update
+void startUpdate()	Start game update
+Getter	currentFrame, frame, Instance

4.How to play

Each player need to attack each other (PVP) and whose hp reach 0% is loser.

Genji

W – jump and press again to double jump

A – move left side

S – move down

D – move right side

J – slash attack

Hanzo

Up – jump

Down – move down

Left – move left side

Right – move right side

Mouse Clicked to shoot

By

Supawit Kunopagarnwong 5831073021

Sorapon Pornsomchai 5831076021