TuaThaiKraiGorDai

Introduction

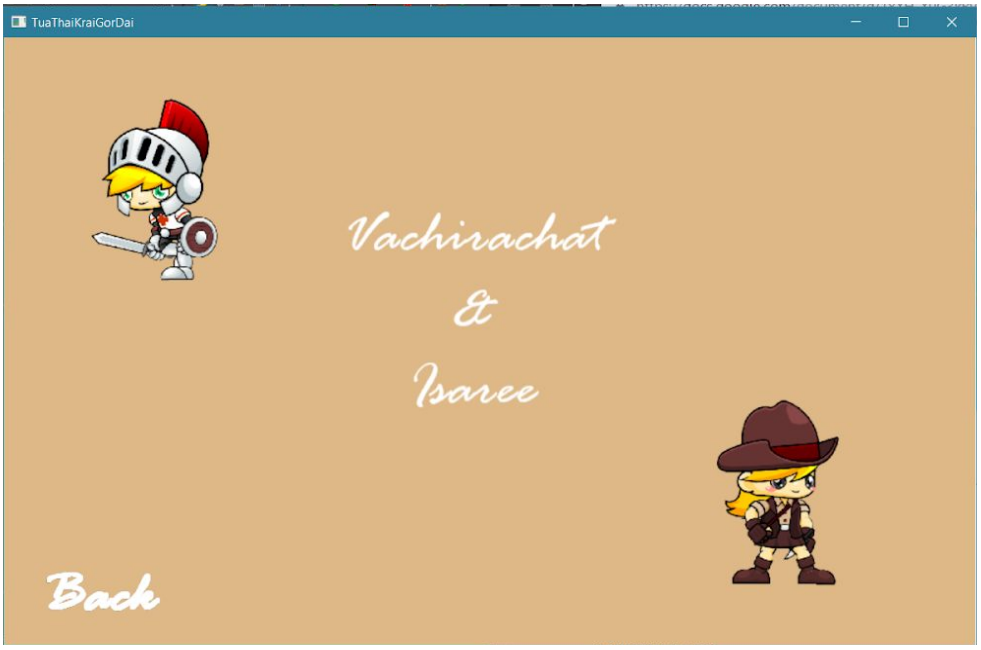
In this game, you are playing as a knight and a cowgirl whose duty is to save the world from the monsters. One day you found out that there were the monsters which came from the other planet and planned to take over the world, so you have to kill and stop them from whatever they're going to do.

Game Control

Start Window

To start the game, click on the play button.
To exit the game, click on exit button.
To see the name generator, click on credit button.

Credit Window



Click on back button(bottom left) to back to start window.

GameWindow (For two players)

- Main view is the view that shows everything in the current map. Background image, monsters, players and particles are shown in this area. Main view's area occupies the whole screen (including status bar area)
- Status bar is the place where the player's properties are shown. The following list are the properties that status bar shows.

- ● Current level
- ● Current HP and max HP
- ● Current experience points
- ● Skills and its cooldown

- Score is the place that showing current score.
- Knight and cowgirl are the player.

Game Control

Knight's control

- Move left: A
- Move right: D
- Move up: W
- Move down: S
- Use skill1: Q
- Use skill2: E

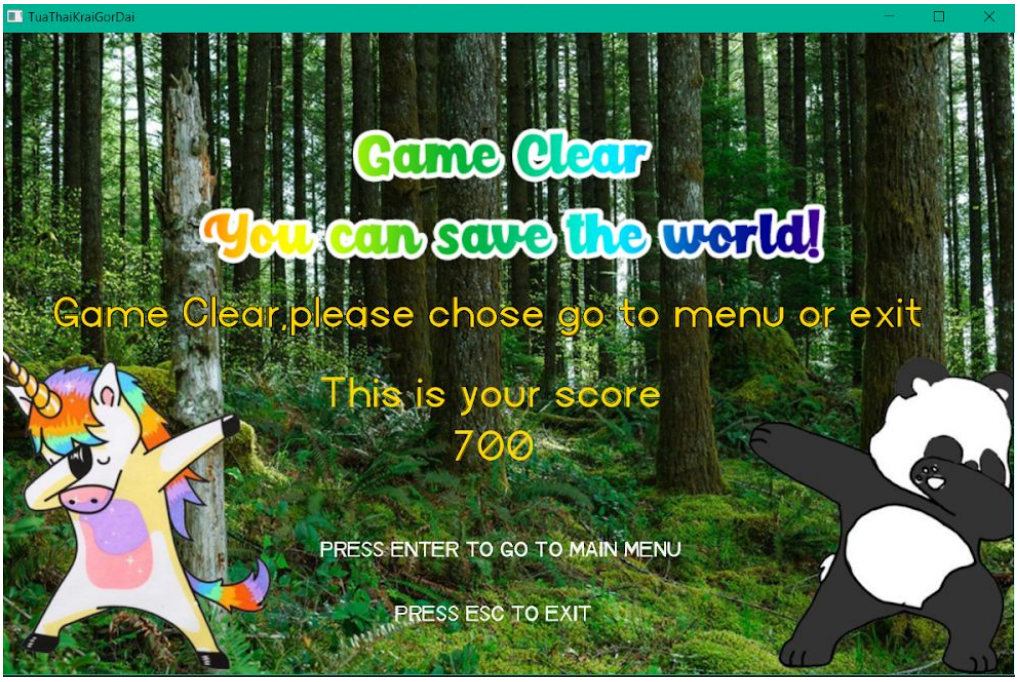
Cowgirl's control

- Move left: ←
- Move right: →
- Move up: ↑
- Move down: ↓
- Use skill1: N
- Use skill2: M

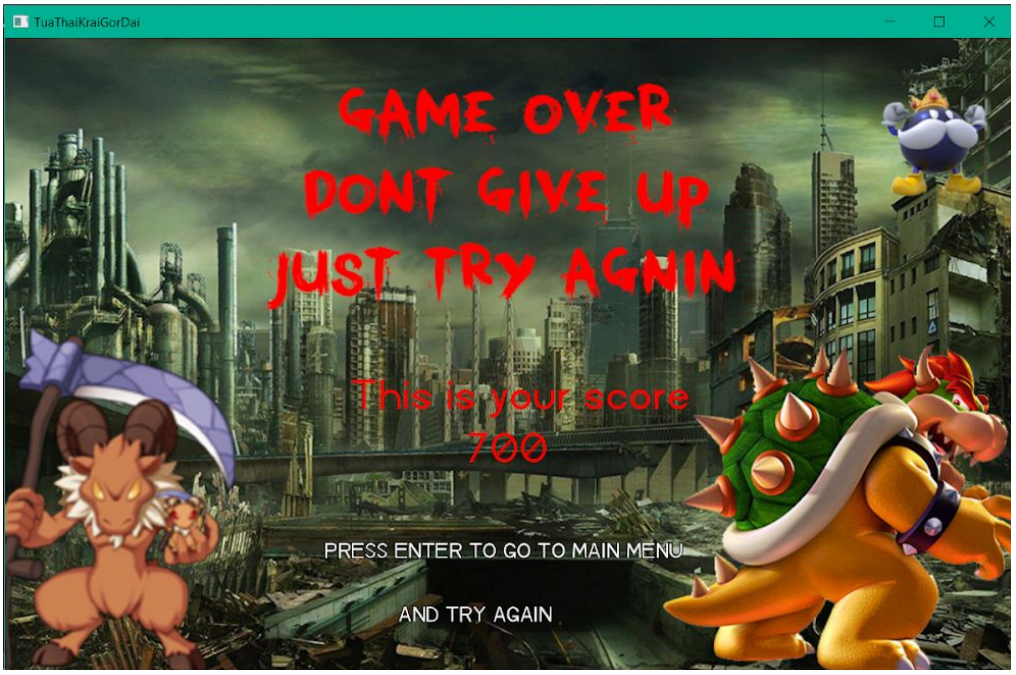
Monster

	Minion		Boss
	1. ninja		1. robot
	2. zombie		2. pumpkin
	3. cowboy		3. dino

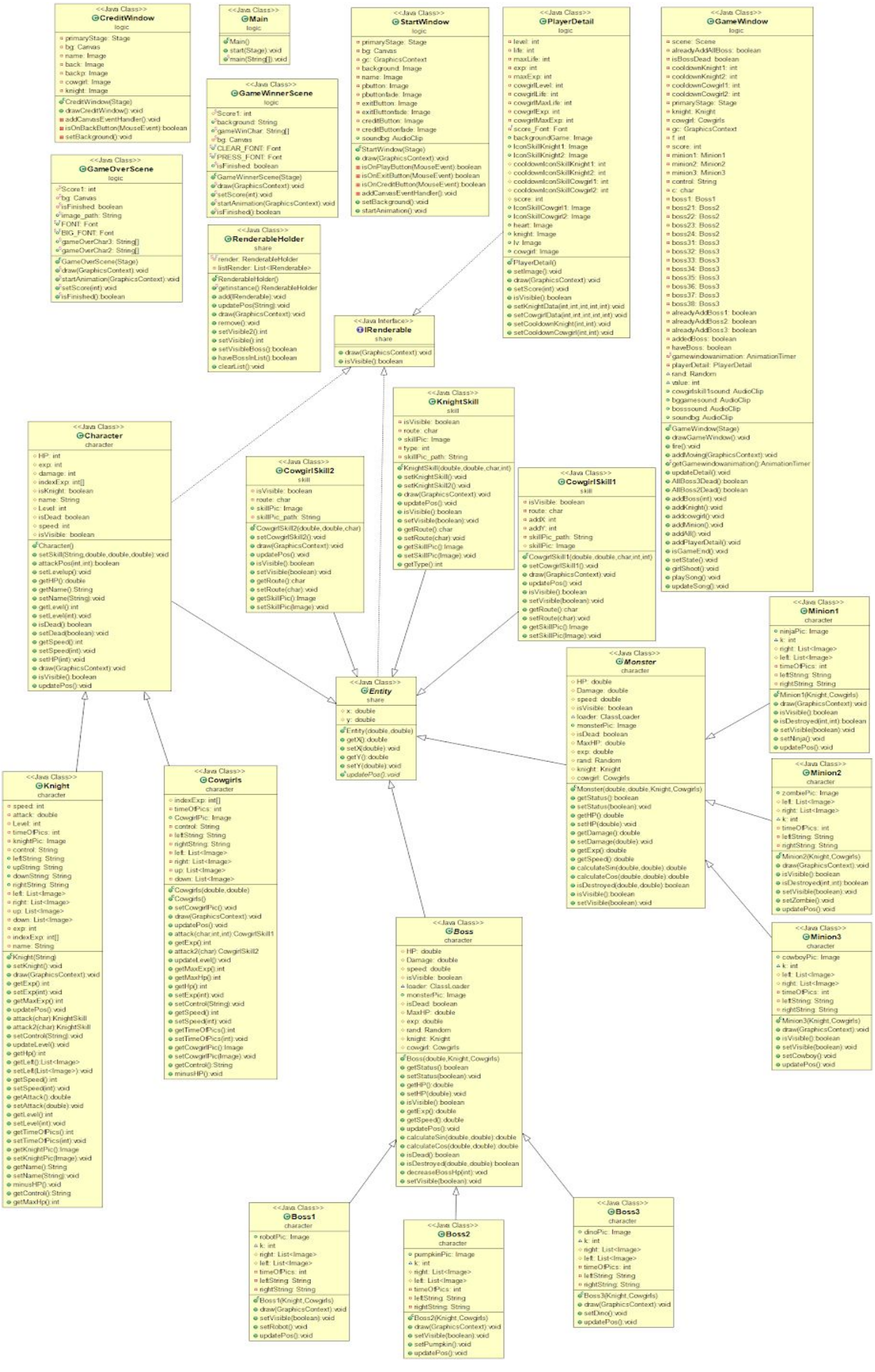
GameWinnerScene



GameOverScene



UML Diagram


CreditWindow
logic
primaryStage: Stage
bg: Canvas
name: Image
back: Image
backp: Image
cowgirl: Image
knight: Image
CreditWindow(Stage)
drawCreditWindow():void
addCanvasEventHandler():void
isOnBackButton(MouseEvent):boolean
setBackground():void
Main
logic
Main()
start(Stage):void
main(String[]):void
StartWindow
logic
primaryStage: Stage
bg: Canvas
gc: GraphicsContext
background: Image
name: Image
pbutton: Image
pbuttonfade: Image
exitButton: Image
exitButtonfade: Image
creditButton: Image
creditButtonfade: Image
soundbg: AudioClip
StartWindow(Stage)
draw(GraphicsContext):void
isOnPlayButton(MouseEvent):boolean
isOnExitButton(MouseEvent):boolean
isOnCreditButton(MouseEvent):boolean
addCanvasEventHandler():void
setBackground():void
startAnimation():void
PlayerDetail
logic
level: int
life: int
maxLife: int
exp: int
maxExp: int
cowgirlLevel: int
cowgirlLife: int
cowgirlMaxLife: int
cowgirlExp: int
cowgirlMaxExp: int
score_Font: Font
backgroundGame: Image
IconSkillKnight1: Image
IconSkillKnight2: Image
cooldownIconSkillKnight1: int
cooldownIconSkillKnight2: int
cooldownIconSkillCowgirl1: int
cooldownIconSkillCowgirl2: int
score: int
IconSkillCowgirl1: Image
IconSkillCowgirl2: Image
heart: Image
knight: Image
lv: Image
cowgirl: Image
PlayerDetail()
setImage():void
draw(GraphicsContext):void
setScore(int):void
isVisible():boolean
setKnightData(int,int,int,int,int):void
setCowgirlData(int,int,int,int,int):void
setCooldownKnight(int,int):void
setCooldownCowgirl(int,int):void
GameWindow
logic
scene: Scene
alreadyAddAllBoss: boolean
isBossDead: boolean
cooldownKnight1: int
cooldownKnight2: int
cooldownCowgirl1: int
cooldownCowgirl2: int
primaryStage: Stage
knight: Knight
cowgirl: Cowgirls
gc: GraphicsContext
f: int
score: int
minion1: Minion1
minion2: Minion2
minion3: Minion3
control: String
c: char
boss1: Boss1
boss21: Boss2
boss22: Boss2
boss23: Boss2
boss24: Boss2
boss31: Boss3
boss32: Boss3
boss33: Boss3
boss34: Boss3
boss35: Boss3
boss36: Boss3
boss37: Boss3
boss38: Boss3
alreadyAddBoss1: boolean
alreadyAddBoss2: boolean
alreadyAddBoss3: boolean
addedBoss: boolean
haveBoss: boolean
gamewindowanimation: AnimationTimer
playerDetail: PlayerDetail
rand: Random
value: int
cowgirlskill1sound: AudioClip
bggamesound: AudioClip
bosssound: AudioClip
soundbg: AudioClip
GameWindow(Stage)
drawGameWindow():void
fire():void
addMoving(GraphicsContext):void
getGamewindowanimation():AnimationTimer
updateDetail():void
AllBoss3Dead():boolean
AllBoss2Dead():boolean
addBoss(int):void
addKnight():void
addcowgirl():void
addMinion():void
addAll():void
addPlayerDetail():void
isGameEnd():void
setState():void
gitShoot():void
playSong():void
updateSong():void
GameWinnerScene
logic
Score1: int
background: String
gameWinChar: String[]
bg: Canvas
CLEAR_FONT: Font
PRESS_FONT: Font
isFinished: boolean
GameWinnerScene(Stage)
draw(GraphicsContext):void
setScore(int):void
startAnimation(GraphicsContext):void
isFinished():boolean
GameOverScene
logic
Score1: int
bg: Canvas
isFinished: boolean
image_path: String
FONT: Font
BIG_FONT: Font
gameOverChar3: String[]
gameOverChar2: String[]
GameOverScene(Stage)
draw(GraphicsContext):void
startAnimation(GraphicsContext):void
setScore(int):void
isFinished():boolean
RenderableHolder
share
render: RenderableHolder
listRender: List<IRenderable>
RenderableHolder()
getinstance():RenderableHolder
add(IRenderable):void
updatePos(String):void
draw(GraphicsContext):void
remove():void
setVisible2():int
setVisible():int
setVisibleBoss():boolean
haveBossInList():boolean
clearList():void
IRenderable
share
draw(GraphicsContext):void
isVisible():boolean
KnightSkill
skill
isVisible: boolean
route: char
skillPic: Image
type: int
skillPic_path: String
KnightSkill(double,double,char,int)
setKnightSkill():void
setKnightSkill2():void
draw(GraphicsContext):void
updatePos():void
isVisible():boolean
setVisible(boolean):void
getRoute():char
setRoute(char):void
getSkillPic():Image
setSkillPic(Image):void
getType():int
Character
character
HP: int
exp: int
damage: int
indexExp: int[]
isKnight: boolean
name: String
Level: int
isDead: boolean
speed: int
isVisible: boolean
Character()
setSkill(String,double,double,double):void
attackPos(int,int):boolean
setLevelup():void
getHP():double
getName():String
setName(String):void
getLevel():int
setLevel(int):void
isDead():boolean
setDead(boolean):void
getSpeed():int
setSpeed(int):void
setHP(int):void
draw(GraphicsContext):void
isVisible():boolean
updatePos():void
CowgirlSkill2
skill
isVisible: boolean
route: char
skillPic: Image
skillPic_path: String
CowgirlSkill2(double,double,char)
setCowgirlSkill2():void
draw(GraphicsContext):void
updatePos():void
isVisible():boolean
setVisible(boolean):void
getRoute():char
setRoute(char):void
getSkillPic():Image
setSkillPic(Image):void
CowgirlSkill1
skill
isVisible: boolean
route: char
addX: int
addY: int
skillPic_path: String
skillPic: Image
CowgirlSkill1(double,double,char,int,int)
setCowgirlSkill1():void
draw(GraphicsContext):void
updatePos():void
isVisible():boolean
setVisible(boolean):void
getRoute():char
setRoute(char):void
getSkillPic():Image
setSkillPic(Image):void
Entity
share
x: double
y: double
Entity(double,double)
getX():double
setX(double):void
getY():double
setY(double):void
updatePos():void
Monster
character
HP: double
Damage: double
speed: double
isVisible: boolean
loader: ClassLoader
monsterPic: Image
isDead: boolean
MaxHP: double
exp: double
rand: Random
knight: Knight
cowgirl: Cowgirls
Monster(double,double,Knight,Cowgirls)
getStatus():boolean
setStatus(boolean):void
getHP():double
setHP(double):void
getDamage():double
setDamage(double):void
getExp():double
getSpeed():double
calculateSin(double,double):double
calculateCos(double,double):double
isDestroyed(double,double):boolean
isVisible():boolean
setVisible(boolean):void
Minion1
character
ninjaPic: Image
k: int
right: List<Image>
left: List<Image>
timeOfPics: int
leftString: String
rightString: String
Minion1(Knight,Cowgirls)
draw(GraphicsContext):void
isVisible():boolean
isDestroyed(int,int):boolean
setVisible(boolean):void
setNinja():void
updatePos():void
Minion2
character
zombiePic: Image
left: List<Image>
right: List<Image>
k: int
timeOfPics: int
leftString: String
rightString: String
Minion2(Knight,Cowgirls)
draw(GraphicsContext):void
isVisible():boolean
isDestroyed(int,int):boolean
setVisible(boolean):void
setZombie():void
updatePos():void
Minion3
character
cowboyPic: Image
k: int
left: List<Image>
right: List<Image>
timeOfPics: int
leftString: String
rightString: String
Minion3(Knight,Cowgirls)
draw(GraphicsContext):void
isVisible():boolean
setVisible(boolean):void
setCowboy():void
updatePos():void
Knight
character
speed: int
attack: double
Level: int
timeOfPics: int
knightPic: Image
control: String
leftString: String
upString: String
downString: String
rightString: String
left: List<Image>
right: List<Image>
up: List<Image>
down: List<Image>
exp: int
indexExp: int[]
name: String
Knight(String)
setKnight():void
draw(GraphicsContext):void
getExp():int
setExp(int):void
getMaxExp():int
updatePos():void
attack(char):KnightSkill
attack2(char):KnightSkill
setControl(String):void
updateLevel():void
getHp():int
getLeft():List<Image>
setLeft(List<Image>):void
getSpeed():int
setSpeed(int):void
getAttack():double
setAttack(double):void
getLevel():int
setLevel(int):void
getTimeOfPics():int
setTimeOfPics(int):void
getKnightPic():Image
setKnightPic(Image):void
getName():String
setName(String):void
minusHP():void
getControl():String
getMaxHp():int
Cowgirls
character
indexExp: int[]
timeOfPics: int
CowgirlPic: Image
control: String
leftString: String
rightString: String
left: List<Image>
right: List<Image>
up: List<Image>
down: List<Image>
Cowgirls(double,double)
Cowgirls()
setCowgirlPic():void
draw(GraphicsContext):void
updatePos():void
attack(char,int,int):CowgirlSkill1
getExp():int
attack2(char):CowgirlSkill2
updateLevel():void
getMaxExp():int
getMaxHp():int
getHp():int
setExp(int):void
setControl(String):void
getSpeed():int
setSpeed(int):void
getTimeOfPics():int
setTimeOfPics(int):void
getCowgirlPic():Image
setCowgirlPic(Image):void
getControl():String
minusHP():void
Boss
character
HP: double
Damage: double
speed: double
isVisible: boolean
loader: ClassLoader
monsterPic: Image
isDead: boolean
MaxHP: double
exp: double
rand: Random
knight: Knight
cowgirl: Cowgirls
Boss(double,Knight,Cowgirls)
getStatus():boolean
setStatus(boolean):void
getHP():double
setHP(double):void
isVisible():boolean
getExp():double
getSpeed():double
updatePos():void
calculateSin(double,double):double
calculateCos(double,double):double
isDead():boolean
isDestroyed(double,double):boolean
decreaseBossHp(int):void
setVisible(boolean):void
Boss1
character
robotPic: Image
k: int
right: List<Image>
left: List<Image>
timeOfPics: int
leftString: String
rightString: String
Boss1(Knight,Cowgirls)
draw(GraphicsContext):void
setVisible(boolean):void
setRobot():void
updatePos():void
Boss2
character
pumpkinPic: Image
k: int
right: List<Image>
left: List<Image>
timeOfPics: int
leftString: String
rightString: String
Boss2(Knight,Cowgirls)
draw(GraphicsContext):void
setVisible(boolean):void
setPumpkin():void
updatePos():void
Boss3
character
dinoPic: Image
k: int
right: List<Image>
left: List<Image>
timeOfPics: int
leftString: String
rightString: String
Boss3(Knight,Cowgirls)
draw(GraphicsContext):void
setDino():void
updatePos():void

Class Detail

1. Package logic

1.1 Class Main extends Application

1.1.1 Method

+ void start(Stage primaryStage)	The main entry point for the JavaFX applications.
+ void main(String[] args)	An entry point of the application

1.2 Class StartWindow

1.2.1 Field

- Stage primaryStage	Stage from Main got from constructor
- Canvas bg	canvas of this screen
- GraphicsContext gc	GraphicsContext from the bg
- Image background	background of the screen
- Image name	Plate of Game's Name
- Image pbutton	Button for start game
- Image pbuttonfade	Button when mouse on start button
- Image exitButton	Button to exit game
- Image exitButtonfade	Button when mouse on exit button
- Image creditButton	Button to go to credit's page
- Image creditButtonfade	Button when mouse on credit button
+ AudioClip soundbg	sound of the bg that will be initialize in the constructor

1.2.2 Constructor

+ StartWindow(Stage primaryStage)	Initialize canvas for bg and gc. Also Initialize soundbg and then play it.

1.2.3 Method

+ void draw(GraphicsContext gc)	Starts background and effect
+ void setBackground()	set the background image on the canvas and the title of the game
- boolean isOnPlayButton (MouseEvent)	Return true when mouse curson is on start button
- boolean isOnExitButton (MouseEvent)	Return true when mouse curson is on exit button
- boolean isOnCreditButton (MouseEvent)	Return true when mouse curson is on credit button
- void addCanvasEventHandler()	Handle animation canvas when moving or clicking on button when clicking on button
+ void startAnimation()	Call the method draw to start

1.3 Class CreditWindow extends Canvas

1.3.1 Field

- Stage primaryStage	Stage from StartWindow
- Canvas bg	canvas of this screen
- Image name	Show name text
- Image back	Button for back to StartWindow
- Image backp	Button when mouse on back button
- Image cowgirl	Cowgirl’s image for decorate
- Image knight	Knight’s image for decorate

1.3.2 Constructor

+ CreditWindow(Stage primaryStage)	Initialize canvas for bg

1.3.3 Methods

+ void drawCreditWindow()	Starts background
- void addCanvasEventHandler()	Handle animation canvas when moving or clicking on button when clicking on button
- boolean isOnBackButton(MouseEvent)	Return true when mouse curson is on back button
- void setBackground()	set the background image on the canvas

1.4 Class PlayerDetail implements IRenderable

1.4.1 Field

- int level	knight's level
- int life	knight's life
- int maxLife	knight's maxlife
- int exp	knight's exp
- int maxExp	knight's maxexp
- int cowgirlLevel	Cowgirl's level
- int cowgirlLife	Cowgirl's life
- int cowgirlMaxLife	Cowgirl's maxlife
- int cowgirlExp	Cowgirl's exp
- int cowgirlMaxExp	Cowgirl's maxexp
- Font score_Font	Font for player detail
+ Image backgroundGame	background image
+ Image IconSkillKnight1	Image of the the skillknight1
+ Image IconSkillKnight2	Image of the the skillknight2
# int cooldownIconSkillKnight1	Count for cooldown time for the

	SkillKnight1
# int cooldownIconSkillKnight2	Count for cooldown time for the SkillKnight2
# int cooldownIconSkillCowgirl1	Count for cooldown time for the SkillCowgirl1
# int cooldownIconSkillCowgirl2	Count for cooldown time for the SkillCowgirl2
# int score	Counting current score
+ Image IconSkillCowgirl1	Image of the the skillcowgirl1
+ Image IconSkillCowgirl2	Image of the the skillcowgirl2
+ Image heart	Image of heart
+ Image knight	knight's icon
+ Image lv	Image of arrow
+ Image cowgirl	Cowgirl's icon

1.4.2 Methods

+ void setImage()	Initialize backgroundgame
+ void draw(GraphicsContext gc)	To show players detail
+ void setScore(int score)	setting the score for the parameter that given in this method for updating the screen
+ boolean isVisible()	PlayerDetail's always use to update the screen for every second so this method always return true.
+ void setKnightData(int maxExp,int exp,int level,int maxLife,int life)	setting the field for the parameter that given in this method for updating the screen about the knight's data.
+ void setCowgirlData(int maxExp,int exp,int level,int maxLife,int life)	setting the field for the parameter that given in this method for updating the screen about the cowgirl's data.
+ void setCooldownKnight(int cooldownKnight1,int	Initialize the image for the skillKnight. If the cooldown is not equal to zero.

cooldownKnight2)	this method should set the image with the cooldown (graycolor picture). If not , use the original one.
+ void setCooldownCowgirl(int cooldownKnight1,int cooldownKnight2)	Initialize the image for the skillCowgirl. If the cooldown is not equal to zero. this method should set the image with the cooldown (graycolor picture). If not , use the original one.

1.5 Class GameWindow extends Canvas

1.5.1 Field

- Scene scene	scene from Main of this game
- boolean alreadyAddAllBoss	Check that all boss is added
- boolean isBossDead	Check that boss is dead
- int cooldownKnight1	Count for cooldown time for the knightSkill1
- int cooldownKnight2	Count for cooldown time for the knightSkill2
- int cooldownCowgirl1	Count for cooldown time for the cowgirlSkill1
- int cooldownCowgirl2	Count for cooldown time for the cowgirlSkill2
- Stage primaryStage	primaryStage from Main of this game
- Knight knight	for initialize knight in the method addKnight
- Cowgirls cowgirl	for initialize cowgirl in the method addCowgirl
- GraphicsContext gc	GraphicsContext of the canvas
- int f	It counts the time to spawn monsters or items
- int score	For counting score
- Minion1 minion1	for initialize minion1 in the method

	addMinion
- Minion2 minion2	for initialize minion2 in the method addMinion
- Minion3 minion3	for initialize minion3 in the method addMinion
- String control	the string that contains the direction of the hero's direction
- char c	To tell the direction of the initial direction
- Boss1 boss1	for initialize Boss1 in the method addBoss
- Boss2 boss21,boss22,boss23,boss2	for initialize Boss2 in the method addBoss
- Boss3 boss31,boss32,boss33,boss34,boss35,boss36,boss37,boss38	for initialize Boss3 in the method addBoss
- boolean alreadyAddBoss1	Adding boss1 in the game
- boolean alreadyAddBoss2	Adding boss2 in the game
- boolean alreadyAddBoss3	Adding boss3 in the game
- boolean addedBoss	Adding boss in the game
- boolean haveBoss	Check that still have boss in list render or not
- AnimationTimer gamewindowanimation	use for running animation on this window
- PlayerDetail playerDetail	for initialize playerdetail in the method addPlayerDetail
- Random rand	for random the number of value
- int value	Value of minion
+ AudioClip cowgirlskill1sound	sound when the player press skill
+ AudioClip bggamesound	sound for the originalgame
+ AudioClip bosssound	Sound when the boss comes out

+ AudioClip soundbg	Sound for gameover or gamewinnerscene

1.5.3 Methods

+ void drawGameWindow()	-call add moving - set frame = 0 - create animation timer instance which will call updateDetail() upDatestate updateSong and isGameEnd()
+ void fire()	If cowgirlskill was pressed the bullets will come out in 8 directions
+ void addMoving (GraphicsContext gc)	set key event handlers and interactions with the game
+ void updateDetail()	Update frame count , spawn monster and add skills on the appropriate frame, remove and draw objects in RenderableHolder. Call methods in RenderableHolder, set exp and lv of hero , Call set Hero data on game screen , check skill's cooldown.
+ boolean AllBoss3Dead()	Check that all boss3 is dead
+ boolean AllBoss2Dead()	Check that all boss2 is dead
+ void addBoss(int countBoss)	Create new Boss instance and add it to RenderableHolder
+ void addKnight()	Create knight instance and add it to RenderableHolder
+ void addcowgirl()	Create cowgirl instance and add it to RenderableHolder
+ void addMinion()	Create new monster instance and add it to RenderableHolder
+ void addAll()	Call addPlayerDetail(), addKnight(), addcowgirl() and addMinion();
+ void addPlayerDetail()	Create new PlayerDetail instance and add it to RenderableHolder

+ void isGameEnd()	Check if game is end, draw GameOver canvas if game over or draw GameWinner canvas if defeated the Boss .
+ void setState()	Set monster speed , hero speed and monster amount according to the gametsate
+ void girlShoot()	Set cowgirlskill moving
+ void playSong()	Play songs
+ void updateSong()	Update the current song

1.6 Class GameWinnerScene

1.6.1 Field

# int Score1	score which you get while playing
+ String background	background of this Class
+ String[] gameWinChar	have the String “Game Clear, Please chose go to menu or exit” in the array of char to call each by each when the animation starts
- Canvas bg	canvas of this screen
- Stage primaryStage	Stage from previous window
- GraphicsContext gc	GraphicsContext from bg
+ Font CLEAR_FONT	Font of Score and Game Clear used in the method draw
+ Font PRESS_FONT	Font of the sentence “PRESS ENTER TO GO TO MAIN MENU” used in the method draw
+ boolean isFinished	true if the animation is finished

1.6.2 Constructor

+ GameWinnerScene(Stage primaryStage)	Initialize canvas for bg

1.6.3 Methods

+ void draw(GraphicsContext gc)	starts the animationTimer for drawing the background and start animations that will show “Game Clear, Please chose go to menu or exit” by each char. When it shows all the string , then show the sentence “PRESS ENTER TO GO TO MAIN MENU” and “PRESS ESC TO EXIT”
+ void setScore(int score)	Set your final score to show on the scene
+ void startAnimation (GraphicsContext gc)	call the method draw to start the animationTimer
+ boolean isFinished()	True if the AnimationTimer is finished all of the function in draw

1.7 Class GameOverScene

1.7.1 Field

# int Score1	score which you get while playing
- Canvas bg	canvas of this screen
- boolean isFinished	true if the animation is finished
+ String image_path	background of this Class
+ Font FONT	Font of the sentence “PRESS ENTER TO GO TO MAIN MENU AND TRY AGAIN”
+ Font BIG_FONT	Font of score and sentence “This is your score”
+ String[] gameOverChar3	have the String “PRESS ENTER TO GO TO MAIN MENU” in the array of char to call each by each when the animation starts
+ String[] gameOverChar2	have the String “AND TRY AGAIN” in the array of char to call each by each when the animation

	starts

1.7.2 Constructor

+ GameOverScene(Stage primaryStage)	Initialize canvas for bg

1.7.3 Methods

+ void draw(GraphicsContext gc)	starts the animationTimer for drawing the background and start animations that will show “PRESS ENTER TO GO TO MAIN MENU AND TRY AGAIN” by each char.
+ void startAnimation (GraphicsContext gc)	call the method draw to start the animationTimer
+ boolean isFinished()	True if the AnimationTimer is finished all of the function in draw

2. Package character

2.1 Class Character extends Entity

2.1.1 Field

# int HP	Hero’s hp.
# int exp	Hero’s exp.
# int damage	Hero’s damage
# int [] indexExp	Contain maxexp for each level
# boolean isKnight	Check that this is knight or not
# int Level	hero’s level. Set initial to 1.
# boolean isDead	Check if the life of Hero
# int speed	The speed of the hero
# boolean isVisible	Check that hero is visible or not

2.1.2 Constructor

+ Character()	- Initialize the position of the hero x = 100 and y = 100

	- Set initial hero’s exp to 0.

2.1.3 Methods

+ void setSkill(String skillName,double damage,double manaUse,double cooldown)	
+ void setLevelUp()	increase Level of hero by 1
+ void draw(GraphicsContext gc)	
+ boolean isVisible()	
+ void updatePos()	
Getters and setters for all field	In setHP(int hp) method, isVisible will be false if hp<=0

2.2 Class Knight extends Character

2.2.1 Field

- int speed	The speed of the knight
- int Level	Knight’s level. Set initial to 1.
- int timeOfPics	count the time the hero is drawn on the screen.
- Image knightPic	Image of the hero drawn on the screen.
- String control	String that contain chars which will indicate the direction knight’s moving.
- List<Image> left	List of images of knight walking to the left
- List<Image> right	List of images of knight walking to the right
- List<Image> up	List of images of knight walking up
- List<Image> down	List of images of knight walking down
- int exp	knight’s exp. Set initial to 0.

- int [] indexExp	Contain maxhp for each level

2.2.2 Constructor

+ Knight(String name)	- Initialize the position of knight - add all images of knight to the left right up down arraylist. - Initialize hp of knight - call setKnight()

2.2.3 Methods

+ void setKnight()	Set initial knightpic
+ void draw(GraphicsContext gc)	- Increase the time by 1.\ - if time >=30 set time = 0 - draw image of knight.
+ void updatePos()	Update current position of knight. The position will update according to the char in the string control and speed of knight. Also, set the knightpic according to the time the knight is drawn(to make knight walk) and the char in the string control (the direction of knight)
+ KnightSkill attack(char c)	Using knightskill1 to the monsters
+ KnightSkill attack2(char c)	Using knightskill2 to the monsters
+ void updateLevel()	If exp of hero exceeding maxexp increase level by 1 set exp to 0 and increase maxexp
+ void minusHP()	Decrease hero’s hp when monster attack
+ int getMaxHP()	Get max hp from 150+(Level*50)
Getters and setters for all field	

2.3 Class Cowgirls extends Character

2.3.1 Field

# int [] indexExp	Contain maxhp for each level

- int timeOfPics	count the time the hero is drawn on the screen.
+ Image CowgirlPic	Image of the hero drawn on the screen.
- String control	String that contain chars which will indicate the direction knight's moving.
- List<Image> left	List of images of cowgirl walking to the left
- List<Image> right	List of images of cowgirl walking to the right
- List<Image> up	List of images of cowgirl walking up
- List<Image> down	List of images of cowgirl walking down

2.3.2 Constructor

+ Cowgirls(String name)	- Initialize the position of cowgirl - add all images of cowgirl to the left right up down arraylist. - Initialize hp of cowgirl - call setCowgirlPic()

2.3.3 Methods

+ void setCowgirlPict()	Set initial cowgirlpic
+ void draw(GraphicsContext gc)	- Increase the time by 1.\ - if time >=30 set time = 0 - draw image of cowgirl.
+ void updatePos()	Update current position of cowgirl. The position will update according to the char in the string control and speed of cowgirl. Also, set the cowgirlpic according to the time the cowgirl is drawn(to make cowgirl walk) and the char in the string control (the direction of cowgirl)

+ CowgirlSkill1 attack(char c,int addX,int addY)	Using cowgirlskill1 to the monsters
+ CowgirlSkill2 attack2(char c)	Using cowgirlskill2 to the monsters
+ void updateLevel()	If exp of hero exceeding maxexp increase level by 1 set exp to 0 and increase maxexp
+ int getMaxHp()	Get max hp from 50+(Level*50)
+ void minusHP()	Decrease hero's hp when monster attack
Getters and setters for all field	

2.4 Class Monster extends Entity

2.4.1 Field

# double HP	Monster's hp
# double Damage	Monster's damage
# double speed	Speed of monster
# boolean isVisible	Tell that the monster is drawn or not
+ Image monsterPic	Image of monster
# boolean isDead	Check if monster is dead
# double MaxHP	Max hp of monster
# double exp	Give hero exp
# Random rand	Create random instance
# Knight knight	Instance of Hero Knight
# Cowgirls cowgirl	Instance of Hero Cowgirls

2.4.2 Constructor

+ Monster(double hp, double damage, Knight knight,Cowgirls cowgirl)	Set field to the corresponding parameters and set sponde monster at different position

2.4.3 Methods

+ boolean isVisible()	Check that monster is drawn on the screen or not.
+ boolean isDestroyed(int x,int y)	Set if position x and position y of knight/cowgirl/skill come nearer than 20 , set isVisible to false
+ double calculateSin(double charX,double charY)	Calculate sin value of the angle of the distance of hero and monster.
+ double calculateCos(double charX,double charY)	Calculate cos value of the angle of the distance of hero and monster.
Getters and setters for all field	

2.5 Class Minion1 extends Monster

2.5.1 Field

+ Image ninjaPic	Image of the minion drawn on the screen.
+ int k	Random number 0 or 1 If k=0, this monster will follow knight. If k=1, this monster will follow cowgirl
# List<Image> right	List of images of minion walking to the right
# List<Image> left	List of images of minion walking to the left
- int timeOfPics	count the time the minion is drawn on the screen.

2.5.2 Constructor

+ Minion1(Knight knight,Cowgirls cowgirl)	- add all images of minion to the left right arraylist. - check that if one player is dead then follow another one - call setNinja()

2.5.3 Methods

+ void draw(GraphicsContext gc)	- Increase the time by 1.\ - if time >=30 set time = 0 - draw image of minion.
+ boolean isVisible()	Check that monster is drawn on the screen or not.
+ boolean isDestroyed(int x,int y)	Set if position x and position y of knight/cowgirl/skill come nearer than 20 , set isVisible to false
+ void setVisible(boolean isVisible)	Set isVisible value
+ void setNinja()	Initialize monster image
+ void updatePos()	Set position of monster which will eventually move towards the hero.

2.6 Class Minion2 extends Monster

2.6.1 Field

+ Image zombiePic	Image of the minion drawn on the screen.
+ int k	Random number 0 or 1 If k=0, this monster will follow knight. If k=1, this monster will follow cowgirl
# List<Image> right	List of images of minion walking to the right
# List<Image> left	List of images of minion walking to the left
- int timeOfPics	count the time the minion is drawn on the screen.

2.6.2 Constructor

+ Minion2(Knight knight,Cowgirls cowgirl)	- add all images of minion to the left right arraylist. - check that if one player is dead then follow another one - call setZombie()

2.6.3 Methods

+ void draw(GraphicsContext gc)	- Increase the time by 1.\ - if time >=30 set time = 0 - draw image of minion.
+ boolean isVisible()	Check that monster is drawn on the screen or not.
+ boolean isDestroyed(int x,int y)	Set if position x and position y of knight/cowgirl/skill come nearer than 20 , set isVisible to false
+ void setVisible(boolean isVisible)	Set isVisible value
+ void setZombie()	Initialize monster image
+ void updatePos()	Set position of monster which will eventually move towards the hero.

2.7 Class Minion3 extends Monster

2.7.1 Field

+ Image cowboyPic	Image of the minion drawn on the screen.
+ int k	Random number 0 or 1 If k=0, this monster will follow knight. If k=1, this monster will follow cowgirl
# List<Image> right	List of images of minion walking to the right
# List<Image> left	List of images of minion walking to the left
- int timeOfPics	count the time the minion is drawn on the screen.

2.7.2 Constructor

+ Minion2(Knight knight,Cowgirls cowgirl)	- add all images of minion to the left right arraylist. - check that if one player is dead

	then follow another one - call setCowboy()

2.7.3 Methods

+ void draw(GraphicsContext gc)	- Increase the time by 1.\ - if time >=30 set time = 0 - draw image of minion.
+ boolean isVisible()	Check that monster is drawn on the screen or not.
+ boolean isDestroyed(int x,int y)	Set if position x and position y of knight/cowgirl/skill come nearer than 20 , set isVisible to false
+ void setVisible(boolean isVisible)	Set isVisible value
+ void setCowboy()	Initialize monster image
+ void updatePos()	Set position of monster which will eventually move towards the hero.

2.8 Class Boss extends Entity

2.8.1 Field

# double HP	Boss's hp
# double Damage	Boss's damage
# double speed	Speed of Boss
# boolean isVisible	Tell that the monster is drawn or not
+ Image monsterPic	Image of boss
# boolean isDead	Check if boss is dead
# double MaxHP	Max hp of นหห
# double exp	Give hero exp
# Random rand	Create random instance
# Knight knight	Instance of Hero Knight
# Cowgirls cowgirl	Instance of Hero Cowgirls

2.8.2 Constructor

+ Boss(double hp, double damage, Knight knight,Cowgirls cowgirl)	Set field to the corresponding parameters and set sponde monster at different position

2.8.3 Methods

+ boolean isVisible()	Check that monster is drawn on the screen or not.
+ boolean isDead()	If boss's hp < 0 , return true
+ void decreaseBossHp(int damage)	Decrease boss's hp when boss is attacked by knight/cowgirl
+ boolean isDestroyed(int x,int y)	Set if position x and position y of knight/cowgirl/skill come nearer than 20 , set isVisible to false
+ double calculateSin(double charX,double charY)	Calculate sin value of the angle of the distance of hero and monster.
+ double calculateCos(double charX,double charY)	Calculate cos value of the angle of the distance of hero and monster.
Getters and setters for all field	

2.9 Class Boss1 extends Boss

2.9.1 Field

+ Image robotPic	Image of the boss drawn on the screen.
+ int k	Random number 0 or 1 If k=0, this monster will follow knight. If k=1, this monster will follow cowgirl
# List<Image> right	List of images of boss walking to the right
# List<Image> left	List of images of boss walking to the left

- int timeOfPics	count the time the boss is drawn on the screen.

2.9.2 Constructor

+ Boss1(Knight knight,Cowgirls cowgirl)	- add all images of minion to the left right arraylist. - check that if one player is dead then follow another one - call setRobot()

2.9.3 Methods

+ void draw(GraphicsContext gc)	- Increase the time by 1.\ - if time >=30 set time = 0 - draw image of boss.
+ boolean isVisible()	Check that monster is drawn on the screen or not.
+ boolean isDestroyed(int x,int y)	Set if position x and position y of knight/cowgirl/skill come nearer than 20 , set isVisible to false
+ void setVisible(boolean isVisible)	Set isVisible value
+ void setRobot()	Initialize monster image
+ void updatePos()	Set position of monster which will eventually move towards the hero.

2.10 Class Boss2 extends Boss

2.10.1 Field

+ Image pumpkinPic	Image of the boss drawn on the screen.
+ int k	Random number 0 or 1 If k=0, this monster will follow knight. If k=1, this monster will follow cowgirl
# List<Image> right	List of images of boss walking to the right

# List<Image> left	List of images of boss walking to the left
- int timeOfPics	count the time the boss is drawn on the screen.

2.10.2 Constructor

+ Boss2(Knight knight,Cowgirls cowgirl)	- add all images of minion to the left right arraylist. - check that if one player is dead then follow another one - call setPumpkin()

2.10.3 Methods

+ void draw(GraphicsContext gc)	- Increase the time by 1.\ - if time >=30 set time = 0 - draw image of boss.
+ boolean isVisible()	Check that monster is drawn on the screen or not.
+ boolean isDestroyed(int x,int y)	Set if position x and position y of knight/cowgirl/skill come nearer than 20 , set isVisible to false
+ void setVisible(boolean isVisible)	Set isVisible value
+ void setPumpkin()	Initialize monster image
+ void updatePos()	Set position of monster which will eventually move towards the hero.

2.11 Class Boss3 extends Boss

2.11.1 Field

+ Image dinoPic	Image of the boss drawn on the screen.
+ int k	Random number 0 or 1 If k=0, this monster will follow knight. If k=1, this monster will follow cowgirl

# List<Image> right	List of images of boss walking to the right
# List<Image> left	List of images of boss walking to the left
- int timeOfPics	count the time the boss is drawn on the screen.

2.11.2 Constructor

+ Boss3(Knight knight,Cowgirls cowgirl)	- add all images of minion to the left right arraylist. - check that if one player is dead then follow another one - call setDino()

2.11.3 Methods

+ void draw(GraphicsContext gc)	- Increase the time by 1.\ - if time >=30 set time = 0 - draw image of boss.
+ boolean isVisible()	Check that monster is drawn on the screen or not.
+ boolean isDestroyed(int x,int y)	Set if position x and position y of knight/cowgirl/skill come nearer than 20 , set isVisible to false
+ void setVisible(boolean isVisible)	Set isVisible value
+ void setDino()	Initialize monster image
+ void updatePos()	Set position of monster which will eventually move towards the hero.

3. Package share

3.1 Class Entity

3.1.1 Field

# double x	Position in x axis of an entity.
# double y	Position in y axis of an entity.

3.1.2 Constructor

+ Entity(double x, double y)	Initialize x and y
+ Entity()	Default constructer for entities

3.1.3 Methods

+ abstract void updatePos()	An abstract method for an entities' position update
Getters and setters for all field	

3.2 Class IRenderable

3.2.1 Methods

+ void draw(GraphicsContext gc)	
+ boolean isVisible()	

3.3 Class RenderableHolder

3.3.1 Field

+ RenderableHolder()	Initialize object for this Class

3.3.2 Constructor

- RenderableHolder render	singleton of the RenderableHolder class
- List<IRenderable> listRender	List of all IRenderable item

3.3.3 Methods

+ RenderableHolder getinstance()	getter of render
+ void add(IRenderable i)	add IRenderable item into the listRender

+ void updatePos(String control)	Only the things that have movement (Surely have method updatePos) will call updatePos method
+ void draw(GraphicsContext gc)	Everything in the object are call method draw
+ void remove()	If the the things in the object is set isVisible to be false , it will be removed from the object
+ int setVisible2()	Set monsters is collide by skill from knight.
+ int setVisible	Set monsters is collide by skill from cowgirl.
+ boolean setVisibleBoss()	Set boss is collide by skill from hero.
+ void clearList()	Clear all the list in object and initialize the new one

4. Package skill

4.1 Class CowgirlSkill1 extends Entity

4.1.1 Field

- boolean isVisible	Tell that this entity is drawn on the screen or not
- char route	use to tell the direction skill's moving
+ Image skillPic	Image of the skill
- int addX	?
- int addY	?

4.1.2 Constructor

+ CowgirlSkill1(double x,double y,char route,int addX,int addY)	- Set field to the corresponding parameters - Initialize the position and the direction of the skill and set the picture of the skill

4.1.3 Methods

+ void setCowgirlSkill1()	Set the picture of the skill
+ void draw(GraphicsContext gc)	Draw the skill on the canvas
+ void updatePos()	Change the position of the skill according to the direction also set visible false if the position is out of the area.
Getters and setters for isVisible, route and SkillPic	

4.2 Class CowgirlSkill2 extends Entity

4.2.1 Field

- boolean isVisible	Tell that this entity is drawn on the screen or not
- char route	use to tell the direction skill's moving
+ Image skillPic	Image of the skill

4.2.2 Constructor

+ CowgirlSkill2(double x,double y,char route)	- Set field to the corresponding parameters - Initialize the position and the direction of the skill and set the picture of the skill

4.2.3 Methods

+ void setCowgirlSkill2()	Set the picture of the skill
+ void draw(GraphicsContext gc)	Draw the skill on the canvas
+ void updatePos()	Change the position of the skill according to the direction also set visible false if the position is out of the area.
Getters and setters for isVisible, route and SkillPic	

4.3 Class KnightSkill extends Entity

4.3.1 Field

- boolean isVisible	Tell that this entity is drawn on the screen or not
- char route	use to tell the direction skill's moving
+ Image skillPic	Image of the skill
- int type	Type of knight skill

4.3.2 Constructor

+ KnightSkill(double x,double y,char route,int type)	- Set field to the corresponding parameters - Initialize the position and the direction of the skill - If type=0, set the picture of the knightskill1. If type=1, set the picture of the knightskill2.

4.3.3 Methods

+ void setKnightSkill()	Set the picture of the skill1
+ void setKnightSkill2()	Set the picture of the skill2
+ void draw(GraphicsContext gc)	Draw the skill on the canvas
+ void updatePos()	Change the position of the skill according to the direction also set visible false if the position is out of the area.
Getters and setters for isVisible, route, type and SkillPic	